A Collaborative Approval Process for Accessing Sensitive Data

Abstract: A collaborative environment to improve and streamline approval processes on-line is presented. A scenario for approving requests to access sensitive medical data records for research purposes from a medical data warehouse at a teaching hospital is used to illustrate the special requirements needed for collaborative approval processes. The architectural framework to support such processes is defined. It includes support for collaborative signatures, a service oriented architecture to integrate security mechanisms, policy-based control to address privacy issues, and categorization and anonymization of data along multiple dimensions to control access within a collaborative environment.

Keywords: security, privacy, digital signature, data anonymization, collaborative environment.

1 INTRODUCTION

Organizations around the world are now collecting large amounts of data about the services they provide to individuals. These data are processed, distributed and shared to better manage their operations and improve the quality of the services they provide. Frichman (2003) provides an excellent summary of the benefits and challenges of such information rich electronic commerce technology. There is great potential benefit for public and private research on that data to address issues that are in the public interest such as health, law enforcement and safety. However, as discussed in Peyton (2005) this must be balanced against the great potential to misuse such information. The data (for example health care data) is often of a sensitive or personal nature. The possible conclusions drawn may also be of a sensitive nature as to regards to public health and safety, so care must be taken that the data are not processed inappropriately or distributed without proper quality assessment. As a result, governments and organizations have been establishing laws, regulations, policies and oversight processes to control the manner in which such data are collected, accessed and distributed.

For example, many hospitals are creating data warehouses of electronic medical records which are potentially a valuable source of data for health research. Due to concerns over privacy, potential litigation, and appropriate use of the data for research there are cumbersome, manual and time-consuming processes in place that must be followed before one can obtain approval to access the data. Information technology could be used to improve and streamline the process, but the regulations and procedures are too complex to allow the process to be completely automated.

In this paper, we define a collaborative approval process to improve and streamline approval processes on-line in a manner that uses automation where possible, but which allows requestors and approvers to interact and discuss in a collaborative environment to create signed official requests and their formal approval.

We also present an architectural framework to support the collaborative approval process. The framework has been validated using a case study based on example scenarios drawn from an actual approval process that is used at a teaching hospital associated with our university.

Our main contributions to the collaborative approval process and associated architectural framework are:

- Identification of the requirements to be addressed by a collaboration approval process for accessing sensitive data. This is driven by what it means to have a signed official request and an information technology context for ensuring security and policy-based compliance with regulations.
- Definition of a two-step collaborative digital signature that allows requestor and approver to take legal responsibility and give final authority.
- Analysis of mechanisms for securing sensitive data including categorization along different dimensions in order to define policy-based control, anonymization of data and audit trails.
- Integration of a policy engine to control access that is automatic where possible, collaborative where needed, and allows for ultimate authority to reside with designated policy officers (approvers).
- Design of collaboration to leverage business process management (BPM) framework and a service oriented architecture (SOA). This enables flexible integration of collaborative approval into regular processes for data collection, access and dissemination. It also enables the collaboration environment to leverage services for privacy policies, security, and signatures.

Copyright © 2005 Inderscience Enterprises Ltd.

2 REQUIREMENTS FOR A COLLABORATIVE APPROVAL PROCESS

Organizations are collecting large amounts of data from individuals and are building data warehouses to support integrated, cross-functional analysis, research, reporting and decision-making. However, permission to view or query data contained in such a data warehouse should be granted only for legitimate institutional purposes.

In this section, we describe a scenario in the context of health care to motivate our approach. Consider the following situation (see Figure 1): a researcher wants to access some sensitive data from the hospital data warehouse for the purpose of research. This researcher fills in the request form to get permission to query data. But in many cases, her request cannot be granted automatically. More information may be needed, or there may be rules which prevent the request as currently formulated to be approved., or the institution may require an interview by a policy officer. In all of these cases there is a need for a collaborative process where the researcher can interact with a policy officer in order to get approval for the data required for her research.

To facilitate the collaborative approval process, we have created a collaboration environment where the researcher and policy officer can discuss and together create and sign a request that will allow the requested data access. There is a chat facility where the researcher and the policy officer can express their ideas and thoughts in messages. They can also view the researcher's requests, policy documents and other information together by shared web browser. They can create and modify the sensitive data request form collaboratively in a controlled fashion. The policy officer controls changing and modifying of the information, and can also transfer this control to the researcher (grant and revoke). They can verify and check policies for the request during the interaction. At the end of the interview, both of them have to sign the updated request form. The researcher signs the form to indicate that she is indeed making the request, and the policy officer signs the form to indicate that approval has been given. The requested is then validated one final time against the policies of the organization and the roles of the two collaborators and then the researcher is given access to the data.

The above scenario describes collaboration in synchronous mode. Alternately the researcher can collaborate in asynchronous mode. The researcher can interact in the collaboration space on her own; view policy documents and related information; modify her request form; submit the form with her signature; and leave. Later, a policy officer will view the researcher's request and decide if it is allowed or denied or perhaps request more information from the researcher.

To facilitate the collaborative approval process in our scenario, several functionalities and privacy and security features need to be provided. We will discuss the requirements for collaborative approval process in details as follows.

2.1 General Requirements of a Collaborative Environment

As a computer-supported cooperative work (CSCW), a collaborative environment should satisfy the specifications of functional and technical requirements for CSCW. Reinhard (1994) defined and described criteria and requirements for CSCW systems. Interaction, coordination, distribution, flexibility, visualization and data hiding were necessary criteria for a collaborative environment. Stiemerling (1999) recognized tailorability as an important feature in a CSCW system. Simone and Schmidt (1998) suggest that a CSCW system should provide awareness. We summarize the general requirements for a collaborative environment as follows:

- Interaction: the system should allow collaborative work among users by proceeding in asynchronous or synchronous mode.
- Coordination: the system should provide a means of communication within the group of users.
- Distribution: the system should enable people to interact from remote places.
- Visualization: the system should visualize public data collaboratively in the paradigm of what-you-see-is-what-I-see.
- Data Hiding: the system should contain a way of separating private data from public data.
- Flexibility: the system should be flexible in application field, functional fields and in technical field.
- Awareness: the system should provide users a way to be aware of what is happening in the environment.
- Tailorability: the system can compose modules from a set of predefined components and can add new functionalities to the system.

2.2 Additional requirements for a collaborative approval process

In a collaborative approval process, one user (requestor) is seeking permission for services or information that are restricted based on organizational policy, government regulation, or law, while another user (approver) is responsible for approving the request and for assisting the first user in understand the terms under which permission may be granted as well as for giving their approval.

One example would be when someone is attempting to gain access to information of a sensitive nature, like a medical research in a hospital. In most countries, personal information is protected by law, and individual organizations have their own policies and rules as well. It is essentially important to protect personal data. Access can be expanded as needed, but privacy, once violated, can seldom be repaired.

Beyond the common security threats that characterize networked systems, a collaborative approval process has several additional security and privacy requirements that are fundamental.

- Accessing sensitive data in compliance with policies, regulations, and law;
- Protecting privacy of shared content during the collaboration;
- Securing shared data exchange. Privacy and integrity should be ensured when documents flow among different parties.
- Online collaborative signature is required to ensure the identity of the signer and verify the integrity of the data.
- Audit trail and statistics for comparison, review, and audit.

In the following sections we will present a collaborative approval process that offers all these requirements and features.

3 SERVICES FOR COLLABORATIVE APPROVAL PROCESS

The collaborative approval process we have implemented is based on the concept of web services delivered within a service-oriented architecture. The researcher and policy officer use web browsers to interact with the system via a collaboration interface. The system composes various services distributed on Internet to achieve its functionalities. The collaboration service provides support for chat, shared editing and browsing. The online collaborative signature of the request combining with signature verification provides data security. In addition, the service-oriented architecture integrates a policy engine to enforce rules defined by the organization, as well as a data access service for accessing data once the request is approved.

3.1 Collaborative Digital Signature

The request and the collaboration between requestor and approver must be tied together with a collaborative digital signature. Since the services or information requested may be of a sensitive nature, responsibility for the request must be formally acknowledged and recorded with a digital signature by the requestor. Similarly, the approver must take responsibility for the advice they are providing and the approval they are granting with a digital signature as well. The two digital signatures must be bound together to the request.

Signing a document in an electronic form starts by calculating an unambiguous checksum of a determined length for the electronic document with the help of a hash function. This checksum is encrypted with the private key of the signing person where it is added to the document as the signature. To verify that this signature belongs to that person, the digital signature confirmed by checking the person's public key, which is freely available to any person. To make sure that the person who signed the document electronically is the one claimed, the checksum of the document is recalculated with the help of a hash function. Then the signed claimed person's public key is used to decrypt the digital signature. Identicalness of both checksums assures that it is the same person. The authenticity of the pair of keys is ensured by means of a certificate where certificates are electronic documents that associate the public key with the identification data of the key holder, which are signed electronically by certification authority. The strength of the digital signature technique depends on the strength of the public key infrastructure which is used in many of today's security services and it shows fairly strong cryptography, as defined in ISO 7498-2 (1989) and ISO/IEC 14888-3 (1999).

Collaborative digital signature (CDS) involves more than one person signing the same document in the same session. CDS requires the right order of signing. For example, in our medical research example, the researcher should sign the request first. Then, the policy officer confirms the request and signs it second. The existence of this double signature on the request's document authorizes the request for access to sensitive data. Notice that the document can not be changed after applying CDS where it is stored together with the CDS in a dedicated database for audit trail purposes. Furthermore, CDS requires the presence of all signers online where they sign the same document within the same web session in a predetermined order.

CDS should be differentiated from first double signature which extends the RSA scheme by having three keys instead of two: two private keys and one public key as presented in Shamir (1979), Boyd and Colin (1989). In this case, each party has one of the two private keys to sign with. Second, from the two-party signature scheme, in which each party holds a share of a decryption exponent and collaborates to compute the digital signature under a single corresponding public key that is know to both of them. Two-party signature scheme is usually used in the domain of server-aided password-based security as in Bellare and Sandhu (2001) and Nicolosi et al (2003). On the other hand, in CDS, each signature is a separate process where the signature assumes that previous signatures are actually part of the document being signed. CDS has a variety of applications including any document that needs more than one signature, especially if it requires the signatures to occur together within a specified time period. CDS can also be extended to more than two signatures

3.2 Security Mechanisms

A number of mechanisms must also be used to ensure the security of the data itself. Access to sensitive data must comply with privacy legislation. We use policy-based authorization to control access to the sensitive data. In order to do so effectively, it is useful to structure and categorize the data along different dimensions in what is usually

referred to as a “star schema”. . In many ways, dimensional modelling amounts to holding the fort against assaults on simplicity. Each dimension is a table in a data warehouse whose entries describe an attribute in a fact table Kimball and Ross (2002). Once all the fields describing a specific attribute are combined in one dimension, it becomes much easier to categorize the dimension’s fields according to privacy concerns. For example, a patient dimension in our medical research example would contain exclusively all identity information about a patient. Attributes like name, address, and postal code are clearly private and should not be used by researchers under any circumstances, whereas attributes like gender, date of birth, or city might be used with care under some circumstances.

Anonymization of data can be supported through the use of a policy engine to control access to the data warehouse. The policy engine refers to an access list that determines the accessibility of each dimension’s field. There are three possible sensitivities values: “identifiable”, “potential identifiable” and “anonymous”, which stand for different levels of sensitivities of the dimension’s field. In addition, there are expandable predefined polices to determine the data accessibility according to different factors: combination of certain fields for each dimension, requester type, purpose of use, and the possible action. The access list together with the predefined polices represent adequate resources for the policy engine to rely on in protecting sensitive data privacy. This is described in more detail in the next section (Policy Engine).

In order for a request to be confirmed so that data access service can fetch the required data, the request has to be signed. Collaborative digital signature complements the process of ensuring data integrity. Users have to be authenticated before they can request or approve data access. To provide strong authentication and protect identity, we use digital certificates issued and verified by a certificate authority to perform authentication on the Internet. An identification or authentication service failure will generate a system audit trail record.

Confidentiality of the data is one of the important security requirements in collaborative approval environments. We use SSL as a security mechanism for communication between the server and client, preventing others from capturing and viewing the data being exchanged. Web services support the transparent exchange of documents to facilitate business integration based on open standards. Such loosely coupled of web services provides significant benefits for their interoperation, but also imposes additional challenges to secure the interoperation Chang et al (2003). We use encrypted SOAP messages to communicate between the collaboration environment web site and web service providers. Moreover, to prevent internal or external thieves from viewing information they obtained without authorization, we use encryption service to encrypt databases storage for sensitive data.

An audit trail is also important. In a separate dedicated database, it records all the transactions between infrastructure components, user’s accesses to the system, and any other event during the collaborative approval process. Requests, their attached digital signatures, and the resulted data addresses are all recorded in the audit trail.

3.3 Policy Engine and Policy Language

The set of policies, regulations, and laws that may be relevant to a collaborative approval process can be quite complex, which can be challenging for humans to interpret and apply in a consistent manner. At the same, time access to data in an electronic environment can be ubiquitous. Services should be put in place to ensure that all access to data is mediated by a policy engine that enforces policy, regulations and laws in the form of business rules consistently. These services should be integrated into the collaborative approval process in such a manner that denied requests are explained and such that the researcher and policy officer can explore different ways of formulating requests to meet the needs of the requestor in a manner that complies with the policies enforced by the policy engine.

The Enterprise Privacy Authorization Language (EPAL) as described in Schunter (2003) is designed specifically to enable an organization to define and manage the enforcement of its privacy policies. EPAL is a XML based language.

We use an open source EPAL engine to act as a Policy Decision Point (PDP) for our collaboration approval process. It has a GUI based privacy administration application (PAP) by which a privacy officer or administrator can define EPAL policies. In EPAL the following categories are used: user, data, purpose and action. Below are two sample policies (summarized in English not EPAL). One will deny access to identifiable data, the other will allow access to anonymous data that is in a form that can be manipulated by the researcher, linked to other internal databases, but the researcher commits to destroying all copies of the data after one month.

ID = A
Ruling = DENY
User = senior researcher & agree to terms
Data = identifiable
Purpose = research & linkage to internal db & consent for linkage
Action = data file manipulation & less than one month & file destroy after

ID = B
Ruling = ALLOW
User = senior researcher & agree to terms
Data = anonymous
Purpose = research & linkage to internal db & consent for linkage
Action = data file manipulation & less than one month & file destroy

Note that in the "data" category of policy definition, the only possible values are "identifiable", "potential identifiable" and "anonymous" which stand for different degrees of sensitivities, rather than specifying specific data fields.. There are two reasons for this.. First the actual data fields can change over time whereas policies should stay constant.. Second, if policies have to be written to each specific data field then we will have to be writing many, many policies. Instead a mapping table is used from actual data field to the categorization of sensitivity. For each request, the data fields in the request need to be categorized in terms of sensitivity based on the mapping table.

It is not realistic to expect, though, that all aspects of policy can be captured in a single set of rules so that an engine can arbitrate all decisions on its own. Typically, organizations and law require ultimate authority to reside with an individual or role within the organization. On the other hand, we want our policy engine to ensure that only authorized requests in compliance with policy are allowed.

Therefore, the policy engine for a collaborative approval process should have some policies that will allow access automatically in simple situations where approval from a policy officer is not required, but it should also have a policies that allows access when the request has been signed by both the requestor and a recognized authorized policy officer. There could be cases in which only certain policy officers are able to sign requests. We have extended EPAL in our system to allow for such situations.

According to a comparison of EPAL and XACML by Anne (2004), EPAL is a functional subset of XACML., especially its 4 category policy definition limitation and no override function support. In XACML, it provides a "Permit-overrides" rule-combining algorithm. XACML Committee draft (2004) defines, in the entire set of rules of the policy, if any rule evaluates to "Permit", then the result of the rule combination SHALL be "Permit". By using this, we can perfectly express the situation when privacy officer and user agree upon a request detail which should override other conditions inside PDP. We will be exploring the use of XACML in future work.

3 APPLICATION FRAMEWORK FOR COLLABORATIVE APPROVAL PROCESS

In this section, we discuss how the various services we have described were tied together into an application framework based on a service oriented architecture (SOA) to support the collaborative approval process. We also discuss how it is evolving from a standalone simple application framework to a business process framework in which the collaboration approval process can itself be embedded.

3.1 Collaboration User Interface

Figure 2-1 is the collaboration user interface where a researcher and a policy officer discuss, view and modify data access request form together. Figure 2-2 shows that there are four panels in this interface:

1. Instant messages panel
   The researcher and the policy officer can chat with each other in synchronous mode.
2. Shared Web browser panel
   The researcher and the policy officer can view shared information in the paradigm of what-you-see-is-what-I-see.
3. Collaborative form editing panel
   The researcher and the policy officer can create and modify the request form together in a kind of controlled way. The policy officer controls action during the collaboration and also can pass the control to the researcher. In the end of discussion, they have to sign in the same agreed form in two steps. First, the researcher signs the form, then the officer signs in the form that signed by the researcher. At last, the officer submits the form which has two signatures.
4. Policy response panel

The researcher and the policy officer can check and verify the data access policy at any time during the collaboration. The result of request is displayed in this panel. According to the policy response, they can adjust their form.

3.2 In the Context of a Service Oriented Architecture

The collaboration approval process takes place within the context of an enterprise computing environment. There are many other applications supported in this environment with which one might want to integrate the collaborative approval process. All of these applications should be leveraging a shared set of services for security, privacy, data access etc. in a service-oriented architecture (SOA).

According to W3C Working Group Note (2004), the service-oriented architecture defines the services of which the system is composed, describes the interactions that occur among the services to realize certain behaviors, and maps the services into one or more implementations in specific technologies. The primary benefit of SOA is the ability to compose services from other less complex services. It is also attractive for its tailorability, flexibility, reusability and extensibility. The services can be distributed over multiple servers from any geographical location in a transparent manner for users.

In Figure 3, we present a simple application architecture for the collaboration approval process environment. In this architecture, the client side user uses browser as its working environment and the services are distributed on the Internet. The working process and message transport is through Internet over HTTPS. The communication between collaborative interface and web services is encrypted SOAP message (Simple Object Access Protocol).

A typical collaboration approval process might involve the services as follows:

1. Collaboration interface retrieves the state of the current request that the researcher is making from request service.
2. The request is sent to policy service which returns a response indicating why it can not be approved as formulated.
3. The request is discussed and modified through interactions supported by the collaboration service (chat, linking of browsers).
4. After both researcher and officer agree on the new request a collaborative signature is enacted using the security service.
5. The signed request is then sent back to policy service which grants access.
6. The approved request is stored with the request service.
7. Researcher gets access to data through the data access service.

This is a relatively straightforward application leveraging services in service oriented architecture. However, it fails to orient the collaborative approval process within the context of the overall work and processes that are taking place within the organization. There are many different points at which a researcher may want access to data as they do their work online, and hence many different points at which a collaborative approval process may be required, and there may be other processes other than simple data access which might benefit from a collaborative approval process.

A more sophisticated architecture is shown in Figure 4. A business process coordinator orchestrates and interweaves the processing of business processes or composite applications that are using a variety of services. Each service may be responsible for one or more steps in an overall business process. The process coordinator determines what business rules apply at the end of each step, and it determines what step comes next. It can also invoke such services independent of individual processes.

For example, this allows one to stipulate across an entire organization, that whenever a data request is made, the policy engine should be invoked to ensure regulatory compliance, and if it should fail, then it can be redirected to a collaborative approval process to resolve the issue. When we create processes for data collection and reporting as well as collaborative approval, all the relevant services can be interweaved in a similar fashion.

By orchestrating those services inside a process coordinator, it gives much greater flexibility to specify process sequences and construct composite application functionality.

4 RELATED WORK

Collaborative environments that support interactions over internet protocols, in particular web services, are becoming increasingly common. Leung (2000) introduced a Networked Intelligent Collaborative Environment (NetICE), which allowing multiple users to be connected to the same virtual space where they could communicate from anywhere at any time. Bidarra (2002) supported synchronous collaborative sessions in which development term members collaborate synchronously via Internet and can assign tasks to each other. Pan (2004) used web services to support collaborative Computer Aided Design (CAD) systems over the Internet in an environment that enabled designers to work together naturally and was easy to maintain.

Others have also looked at the integration of security, and policy-based mediation. Traoré (2003) presents a formal security model that combines access and information flow control in collaborative environment. The model allows the expression and enforcement of specific polices expressed by several principals for sensitive resources. Demchenko et al (2005) propose a flexible customer-driven security infrastructure for Open Collaborative Environment (OCE) to build a flexible, customer-driven security infrastructure for open collaborative applications. The architecture is based on web services and grid security technologies while combining with concepts of the generic authentication authorization and accounting (AAA) and role-based access control (RBAC) frameworks. The paper also provides a job-specific context for invocation of the basic OCE security services.

In our framework, we go further, embracing a service oriented architecture that facilitates the use of existing services by the collaborative environment for security, signature, data access, and privacy enforcement. In fact, our position is that collaboration should itself be a service that can be invoked in context. We also explored business process management and emerging standards such as BPEL as mechanisms for defining the way in which collaborative processes can be interweaved into other on-line processes.

In doing so, we have followed the path described by Erl (2004) in terms of two stages of web services. In the first stage, dispersed functional points within different applications are exposed as web services. It makes possible to invoke these functions from outside of an application. But in most of situation, a real complete business function often needs to involve several functional invocations and message procession/transformation. It requires aligning these individual services to fulfil a business purpose.

In the second stage of web services, context and transaction management is the first concern. Then, as a second concern, we look at when a business activity involves several web services invocations. In that case, a business process needs to be defined to structure the workflow and communication.

Gold-Bernstein and Ruh described that Business Process Management (BPM) is relevant and includes process modelling, automation, integration, workflow integration (for manual process), monitoring and optimization. The Business Process Execution Language for Web Service (BPEL4WS) is an emerging standard for modelling business

processes that enables portability of business model among tools and allows business as well as technical users to visualize business.

Our approach to policy-based control of access is based on Rouault and Clercq's (2004) policy based privacy framework with the three main components:

- Policy Administration Point (PAP) is the place where privacy policy be defined. It could be an application by which policy officers/administrator use to configure privacy policy.
- Policy Decision Point (PDP) is the application/service provides the policy based decision making. It checks the user request against pre-defined policy and gives back the verdict.
- Policy Enforcement Point (PEP) is the place where the policy verdict got enforced. PDP is only responsible for giving verdict (yes or no answer). It is up to PEP decides what exact operation needs to be done, like denying/permitting access, writing logs.

5 CONCLUSION AND FUTURE DIRECTIONS

Approval processes are well suited as an application area for collaborative environments. In an increasingly, complex and on-line society, the processes for regulating access and ensuring compliance can be improved by on-line collaboration. We have highlighted specific requirements and architectural approaches that arise in the context of a collaborative approval, including the need for secure data exchange, collaborative signature, and policy-based enforcement of legislation. In addressing these diverse requirements, a service oriented architecture supports the division of labor and flexible integration or services needed to fully support a collaborative approval process.

A future direction for our work is to investigate model-driven approaches integrated with Business Process management in order to build a collaborative approval process that can be configured for any request form. The current request is a hard-coded HTML form. In Dubinko (2003), the XForms standard is describe which uses XML to store data and also to define the form's presentation separate from the data as well as its routing.

6 ACKNOWLEDGMENT

We wish to acknowledge the support, encouragement, discussion of scenarios, and review of our work provided by Marcus Bornfreund and Alistair Forster of the University of Ottawa Technology Law program, and Alan Forster of the Ottawa Hospital. This research was supported by funds from ORNEC (Ontario Research Network for Electronic Commerce) and by NSERC (National Sciences and Engineering Research Council of Canada) as well as a generous equipment grant from CFI (Canada Foundation for Innovation), OIT (Ontario Innovation Trust), and IBM Canada.

7 REFERENCES

1 Anderson A, *"A Comparison of EPAL and XACML"*, 2004, Sun Microsystems. http://research.sun.com/projects/xacml/CompareEPALandXACML.html

2 Bellare M, and Sandhu R, *"The Security of Practical Two-Party RSA Signature Schemes"*, Manuscript, 2001.

3 Bidarra R, Kranendonk N, Noort A, and Bronsvoort WF, (2002), *"A collaborative framework for integrated part and assembly modelling"*, Journal of Computing and Information Science in Engineering, vol. 2, no. 4, pp. 256-264.

4 Boyd, and Colin. *"Digital Multisignatures"*, Cryptography and Coding. H.J.Beker and F.C.Piper Eds., Oxford University Press, 1989, pp241-246. http://sky.fit.qut.edu.au/~boydc/papers/ima89.pdf

5 Chang S, Chen Q, and Hsu M, *"Managing security policy in a large distributed web services environment"*, Proceedings of the 27th Annual International Computer Software and Applications Conference, 2003.

6 Demchenko Y, Gommans L, de Laat C, Oudenaarde B, Tokmakoff A, Snijders M, and van Buuren R, *"Security Architecture for Open Collaborative Environment"*, Accepted paper for EGC2005 Conference February 14-16, 2005, http://staff.science.uva.nl/~demch/papers/egc2005-oce-security-architecture-ydemchenko-01.pdf

7 Dubinko M, *"XForms Essentials"*, 2003, O'Reilly.

8 Erl T, *"Service-Oriented Architecture: A field Guide to Integrating XML and Web Services"*, 2004.

9 *"eXtensible Access Control Markup Language (XACML)"*, Version 2.0, Committee draft 02, 30 Sep 2004.

10 Frichman, R.G., Cronin, M.J., *"Information-Rich Commerce at a Crossroads: Business and Technology Adoption Requirements"*, Communications of the ACM Sept. 2003, Vol. 46, No. 9.

11 Gold-Bernstein B, and Ruh W, *"Enterprise Integration"*, 2005, Addison-Wesley Oracle.

12 ISO 7498-2. *Information processing systems- Open Systems Interconnection – Basic Reference Model – Part 2: "Security Architecture"*, International Organization for Standardization JTC 1, 1989.

13 ISO/IEC 14888-3, *Information Technology – Security Techniques – Digital Signatures with appendix – Part 3*, International Organization for Standardization JTC 1/SC 27, 1999.

14 Kimball R and Ross M, *"The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling"*, Second Edition, 2002.

15 Leung WH, Goudeaux, K., Panichpapiboon, S., Sy-Bor Wang, Tsuhan Chen, *"Networked Intelligent Collaborative Environment (NetICE)"*, Multimedia and Expo, 2000. ICME 2000. IEEE International Conference on Volume 3, 30 July-2 Aug. 2000 Page(s):1645 - 1648 vol.3.

16 Nicolosi A, Krohn M, Dodis Y, and Mazières D, *"Proactive two-party signatures for user authentication"*, in Proceedings of the 10th Annual Network and Distributed System Security Symposium, pages 233-248, February 2003.

17 Pan Y, Xu D, Chen C, and Zhang Y, *"Using Web services implementing collaborative design for CAD systems"*, Services Computing, 2004. (SCC 2004). Proceedings, 2004 IEEE International Conference on 15-18 Sept. 2004 Page(s):475 – 478.

18 Peyton L, and Hu J, *"Protecting Privacy while Addressing Identity Theft"*, MCETECH 2005, Montreal, Canada, 2005.

19 W. Reinhard, J. Schweitzer and G. Volksen, *"CSCW Tools: Concepts and Architectures"*, IEEE Computer, Volume 27, Issue 5, May 1994 Page(s):28 – 36.

20 Rouault J and De Clercq J, *"Identity Management Architectures"*, July 2004, HP Dev Resource Central.

21 Schunter M., Powell C., The Enterprise Privacy Authorization Language (EPAL), IBM, June, 2003. http://www.zurich.ibm.com/security/enterprise-privacy/epal/

22 Shamir, A., *"How to Share a Secret"*, Comm. ACM 22, 11, 612-613, 1979.

23 Simone, C. and Schmidt, K., , 1998, *"Taking the distributed nature of cooperative work seriously "*, Proceedings of 6th Euromicro Workshop on Parallel and Distributed Processing, Madrid, Spain, IEEE Press, pp. 295-301.

24 Stiemerling, Oliver; Hinken, Ralph; Cremers, and Armin B., *"Distributed Component-Based Tailorability for CSCW Applications, in: Proceedings of the ISADS '99"*, IEEE Press, Tokyo, Mar. 20-23, 1999, pp. 345-352.

25 Traoré I, and Khan S, *"A Protection scheme for Collaborative Environments"*, March 2003 Proceedings of the 2003 ACM symposium on Applied computing.

26 *"Web Services Architecture"*, W3C Working Group Note 11 February 2004. http://www.w3.org/TR/ws-arch

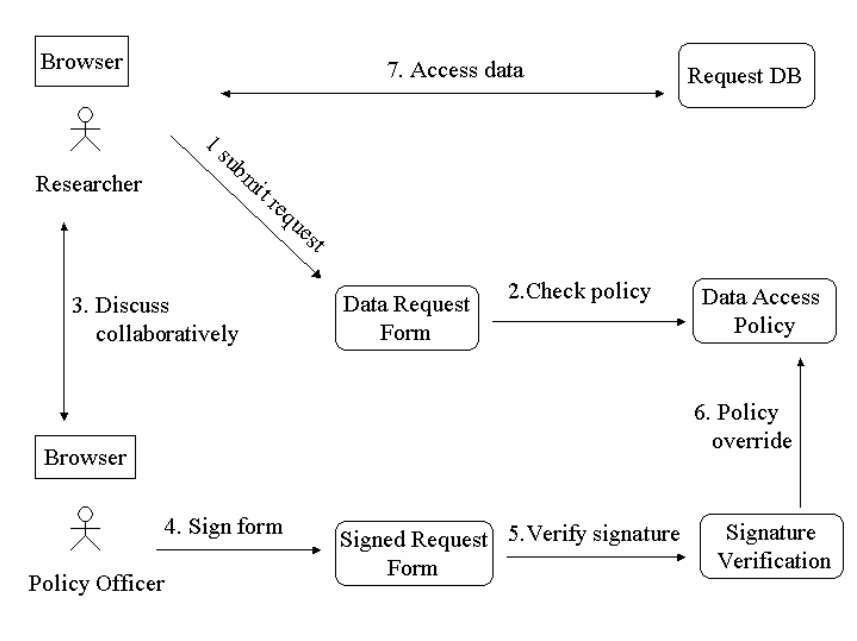

Figure 1 Collaborative approval process scenario

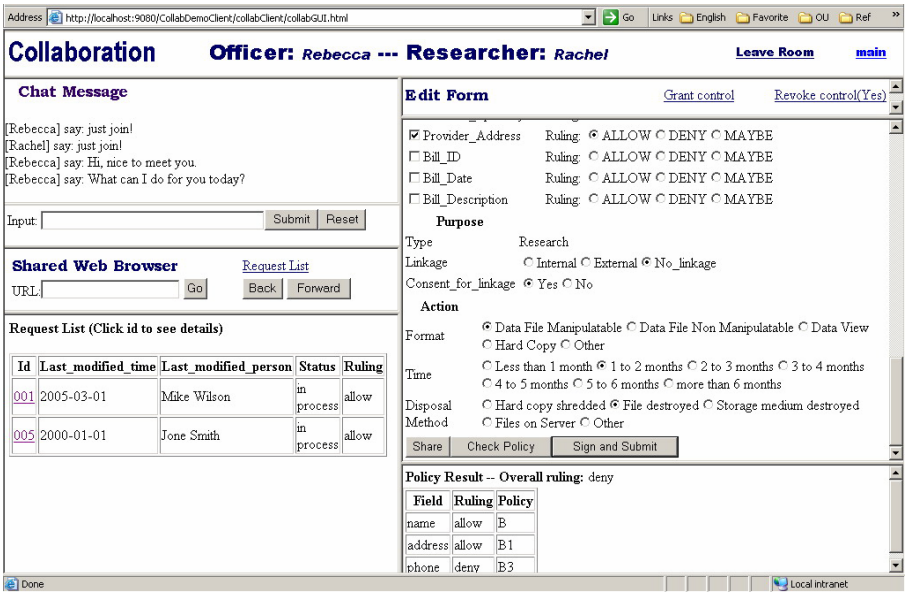

Figure 2-1 Collaboration User Interface

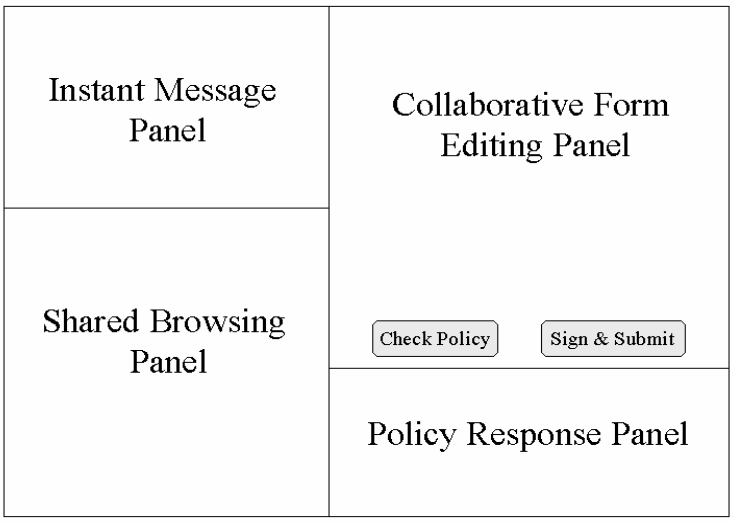

Figure 2-2 Four panels in the collaboration user interface

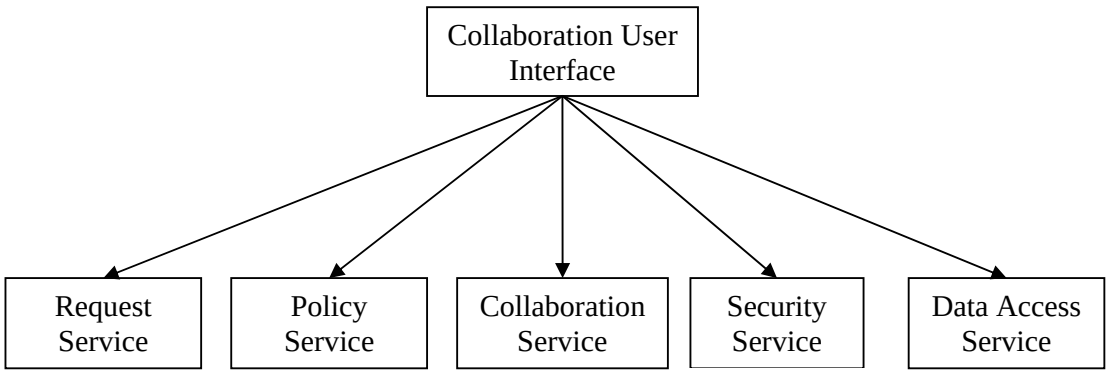

Figure 3 – Simple Application

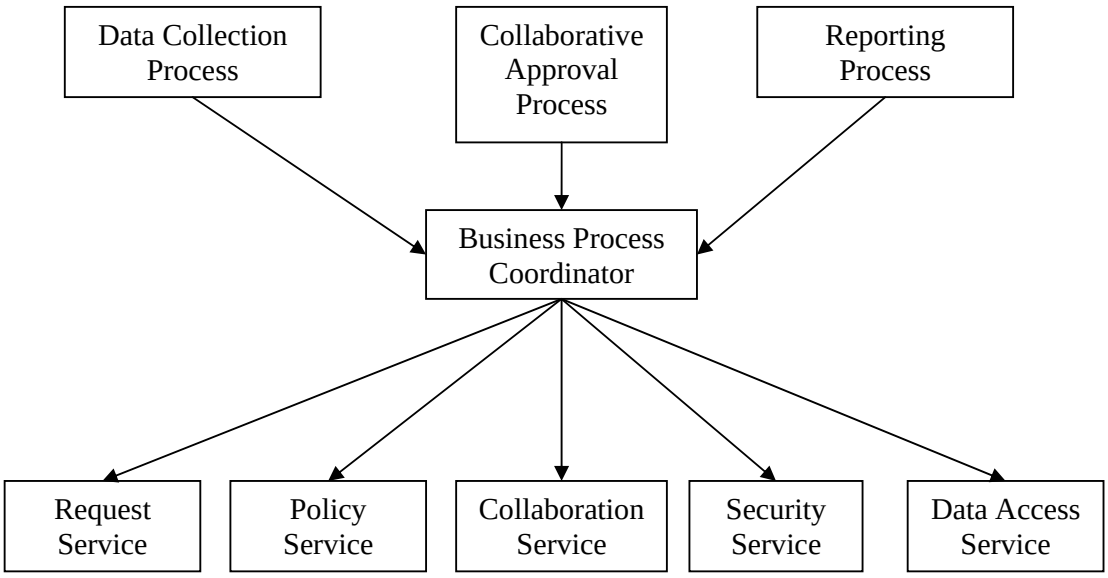

Figure 4 – A Framework for Business Processes