International Journal of Computer Science and Applications

Vol. 9 No. 1, pp. 116 - 151, 2012

TRUSTED INFORMATION SHARING USING SOA-BASED SOCIAL OVERLAY NETWORKS

Florian Skopik*

Safety and Security Department, AIT Austrian Institute of Technology
2444 Seibersdorf, Austria
florian.skopik@ait.ac.at – http://www.ait.ac.at/it-security

Daniel Schall, Schahram Dustdar

Distributed Systems Group, Vienna University of Technology
Argentinierstrasse 8/184-1, 1040 Vienna, Austria
lastname@infosys.tuwien.ac.at – http://www.infosys.tuwien.ac.at

Cross-enterprise collaboration has emerged as a key survival factor in today's global markets. Semantic Web technologies are the basis to establish enterprise interoperability including data mediation support and automatic composition of services. Capabilities of services are semantically described and reasoning techniques support the discovery and selection of services at run-time. In contrast to Semantic Web technologies that cover interactions between (technical) services, human collaboration emerges based on *social preferences*. Social networks have become a mass phenomenon and are increasingly used in businesses and professional environments. In a manner similar to service-oriented systems, they enable flexible discovery and dynamic collaboration between participants. In this paper, we discuss the concept of social overlays for Web service based collaboration infrastructures. This mechanism enables information flows between actors in order to allow for flexible group formation in highly dynamic large-scale networks. We present the implementation of a trusted information sharing framework and demonstrate how people adaptively share information according to the strength of social relations using SOA concepts. We evaluate technical concepts with in-depth experiments.

Keywords: Service-Oriented Enterprise Collaboration, Social Networks, Semantic Overlay, Trusted Information Sharing

1. Introduction

The rapid advancement of ICT-enabled infrastructure has fundamentally changed how businesses and companies operate. Global markets and the requirement for rapid innovation demand for alliances between individual companies. Such alliances are created on different scales ranging from short- to long-term. A long-term alliance is typically a merger of companies or individual organizational units. Short- to

*Parts of work presented in this paper were performed while the first author was with the Distributed Systems Group at the Vienna University of Technology.

mid-term alliances are commonly created to perform joint collaborations with the goal of fulfilling business objectives. Organizations have become *open enterprises systems* (OES) that offer capabilities as services. Capabilities can be discovered and composed to form new alliances. However, such systems do not only span automated interactions among (technical) services, but require humans actors to be in the loop. Today's Web applications facilitate interactive knowledge sharing, information exchange, content creation, and collaboration on the WWW. Even in business environments, *Web 2.0* tools increasingly provide users the free choice to interact or collaborate with each other in virtual communities. The Web becomes thereby a medium of interwoven human and service interactions. These principles have also changed models for computing on the Web by utilizing human manpower through crowdsourcing platforms (e.g., Amazon Mechanical Turk [Amazon.com, 2011]).

There are two obstacles hampering the establishment of seamless communications and collaborations across organizational boundaries: (i) the dynamic discovery and composition of resources and services, and (ii) flexible and context-aware interactions between people residing in different departments and companies. Here we address challenges related to human interactions in dynamic service-oriented systems. Semantic technologies and platforms [Berners-Lee et al., 2001] provide the means to automate the discovery and interactions of compositions. Semantically-enriched collaboration services provide the means for flexible interaction support. The technical composition layer of a service-oriented system (SOA) has received considerable attention in recent years from both the research community and industry. Considerably less attention was devoted to human aspects and interactions in such systems. For example, people use services to perform collaborations. We focus on *social aspects* in cross-organizational collaborations enabled by SOA. In order to take advantage of social preferences, we propose social network principles to overcome limited information flows in collaborative environments. Social interactions between network members allows to influence and control information flows.

Challenges and Approach Outline. In this work we address challenges related to the automated management of social networks based on interactions in cross-organizational collaborations. Major Objectives are:

- Top-down composition and interaction models are typically designed for long-term use. Dynamic environments that are short- to medium-lived such as open enterprise systems require *dynamic interaction models*. Flexible interactions with the purpose of communicating, coordinating, and collaborating need to be supported in a service-oriented manner.
- Theories found in social network analysis are promising candidate techniques to support flexible interactions. Since interactions take place dynamically, capturing the purpose and context of interactions to infer meaningful social relations remains challenging.
- Social network principles such as formation algorithms help to overcome

limited information exchange in separated collaborative networks through propagation of profile data. From the technical point of view, adaptive information flows need to be supported using services technology. Information needs to be discovered and exchanged based on the underlying social network.

- Traditional approaches to access rights management are based on manually assigned static user roles. In dynamic environments, the user is often not able to keep track of configurations such as dynamically changing roles.

The Semantic Web and related technologies have made important contributions to pave the way towards the effective interoperability of enterprise systems and infrastructures. Due to the proliferation of Web 2.0 collaboration principles and Semantic Web technologies, a combination of these approaches seems to be promising to create novel cross-enterprise collaboration systems. In the following we give an outline of our approach.

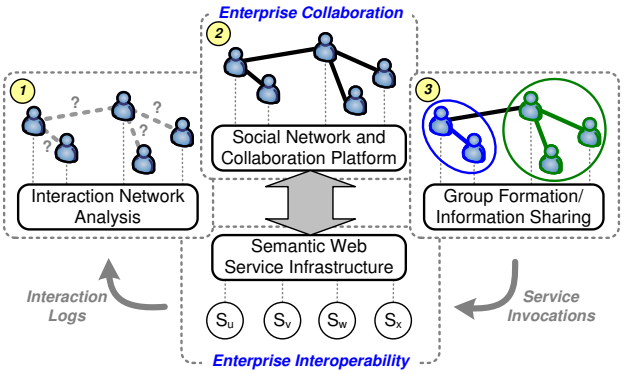


Fig. 1. Enterprise collaboration and interoperability through social and Semantic Web techniques.

Figure 1 illustrates the fundamental motivation of applying *and* combining Semantic Web methodologies with Web 2.0 concepts. We show two main building blocks (i) *Enterprise Interoperability* and (ii) *Enterprise Collaboration* to support a seamless service-oriented infrastructure for cross-organizational collaboration in open enterprise systems. The *Semantic Web Service Infrastructure* provides the means to enable efficient and dynamic interactions spanning humans that belong to different organizational units. Underneath, Web services build an abstraction mechanism for intra-organizational infrastructures and resources and therefore, are the ideal technical grounding to enable interactions across organizational boundaries. Observing interactions and collecting collaboration data (*Interaction Network Analysis*) helps to support humans in building up new relationships by recommending new partners or notifying about possibly interesting business opportunities. A *Social Network and Collaboration Platform* allows people to manage their personal contacts and interact with well-known collaboration partners in context of certain

projects. *Group Formation and Information Sharing Support* concepts applied in collaborative networks allow actors to discover unconnected members using profile information, to build alliances, and to dynamically establish reliable information flows in order to exchange information.

Our Contributions. In this paper we deal with:

- *Cross-Organizational Application Model.* Cross-organizational scenarios are supported considering social aspects of interacting humans on the Web and technological interoperability using Semantic Web concepts.
- *Group Formation.* Formation is typically based upon sophisticated member discovery techniques. Thus, enabling actors to share personal profiles and information in a trustworthy manner is a key concept of our work. We discuss a social trust based access control (TBAC) mechanism that accounts for dynamically changing trust relations.
- *Specification and Implementation.* We discuss the implementation of social overlay networks using today's Web technologies, including Semantic Web services, interaction mining techniques, public key infrastructures, and the Friend-Of-A-Friend (FOAF) ontology.
- *Sharing Framework for Service-oriented Collaboration Networks.* We discuss the prototype implementation of a sharing portal which resides on top of a social overlay network.
- *Evaluation and Discussion.* We evaluate proposed models and their application in virtual communities, and derive general findings for designing applications for socially-enhanced service-oriented environments.

Structure of this Work. The remainder of this paper is organized as follows. In Section 2 we outline our approach of linking Semantic Web paradigms with social network concepts, and introduce a motivating use case that highlights the application of social trust networks in human-centric flexible collaborations. Concepts for distributed social network management are further presented in Section 3. We specify and implement this system as shown in Section 4. Section 5 deals with the Trusted Information Sharing (TIS) Framework that resides on top of the created social overlay network. Then, we evaluate and discuss our work in Section 6. Section 7 deals with related work and Section 8 concludes the paper.

2. Social Overlays in Semantic SOA

Enterprise collaboration and interoperability services are going to become an invisible, pervasive, and self-adaptive knowledge and business utility for any industrial sector and domain. The goal is to enable rapid set-up, efficient management and effective operation of different forms of business collaborations, from the most traditionally supply chains to the most advanced and dynamic business ecosystems. Here, we discuss the *foundational concepts* of our proposed approach to *social overlay networks*, and discuss its application in a *science collaboration use case*.

2.1. *Foundational Building Blocks*

Figure 2 shows an overview of our layered approach to enable reliable and flexible formation of collaboration groups: (i) the *Semantic Service Layer* provides the technical infrastructure to semantically describe and host Web services in order to enable cross-organizational collaborations; (ii) the *Interaction Layer* provides the means of Web service-based human interactions; e.g., allows actors to communicate and collaborate with others using dedicated services from the bottom layer; (iii) the *Monitoring Layer*, observes interactions collected from various sources (i.e., interaction services); and (iv) the *Formation and Sharing Layer* discovers social relations gathered through mining of interactions and profile properties, and supports group formation based on evaluating network links.

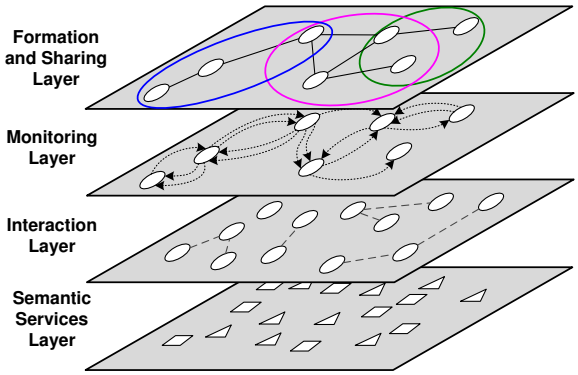


Fig. 2. Model for creating social overlay networks.

2.1.1. *Semantic Web Service Infrastructure*

In order to realize the vision of cross-organizational collaboration and interoperability, various multi-national research projects, such as within the EU Seventh Framework Program[a], are conducted. The COIN project[b], where our contributions of this paper are embedded, aims at developing a basic platform for future Web based cross-organizational collaborations. In the following, we discuss the architectural model of semantically-enriched social OESs and outline utilized major concepts on each layer.

The COIN project aims at providing an open, self-adaptive integrative solution for *Enterprise Interoperability* and *Enterprise Collaboration*. Service orientation is a well-suited and widely adopted concept in collaboration scenarios, therefore, COIN utilizes state of the art SOA concepts, including Semantic Web technologies and

[a] `http://cordis.europa.eu/fp7`
[b] `http://www.coin-ip.eu`

Software-as-a-Service (SaaS) models (see [Gold et al., 2004] for more details). With respect to Enterprise Collaboration, COIN supports numerous features that focus on product development, production planning and manufacturing, and project management in networks of enterprises. As a fundamental aspect, human interactions exist in all forms and phases of virtual organizations and play a major role in the success of collaborations within open enterprise networks. Therefore, understanding human interactions and providing advanced support for efficient and effective interactions, is one of the key objectives in COIN's Enterprise Collaboration research track. The COIN Framework consists of (i) the Social Network and Collaboration Platform (SCP) that provides fundamental features that are required in (nearly) every collaboration scenario, and (ii) a Semantic Web Service Infrastructure (SSI) that allows extensions with services following the SaaS model from third party providers. The SCP is designed for and tightly coupled to a community portal that provides an effective way to configure and personalize the SCP for specific end-users by providing customized services and tools. Single sign-on- and security mechanisms span services and tools across layers. The SSI relies on Semantic Web technologies, implemented by the Web Service Modeling eXecution environment (WSMX)[c] [Haller et al., 2005] and is utilized to discover, bind, compose, and use third-party services at run time. Because of its extensibility and configurability, the COIN platform can be applied in a wide variety of different collaboration scenarios, ranging from traditional production planning to social campaigning and interest group formations in professional virtual communities. For enabling context-aware interactions, the following baseline components are of major interest (i) user data, including skills and interest profiles, (ii) context data, such as current ongoing activities and user preferences, (iii) integrated baseline services for communication and coordination (e.g., e-mail notifications, and instant messengers), (iv) the SCP as the platform to host extended human interaction services.

2.1.2. *Human Interaction Layer*

Open enterprise systems that allow to form virtual organizations pose additional challenges to human interaction support. Typically such virtual organizations are temporary alliances that form and dissolve again. Various actors from different physical organizations are involved collaborating and working on joint activities. Various artifacts need to be created in order to integrate common WSDL-based Web services into the Semantic Web infrastructure of WSMX [Zaremba and Vitvar, 2008]. We provide a basic description that acts as the underlying basis for the rest of this paper in the following:

- *Enterprise Collaboration Ontology:* A collection of predefined semantic concepts establishes data interoperability through transformation, mediation,

[c] `http://www.wsmx.org`

and reasoning. For that purpose the basic enterprise collaboration entities (see [Skopik et al., 2011] for details) and their relations are well defined in a baseline ontology.

- *Semantic Goals:* A client specifies the objective to be achieved in terms of a goal [Stollberg and Norton, 2007], and the system resolves this by automated detection, composition, and execution of Web services. This concept allows dynamic discovery based on functional as well as non-functional properties, and advanced composability of services and service instances respectively.
- *Grounding Descriptions:* Since WSMX functionalities operate on semantic descriptions of messages, non-semantic messages require transformations to semantic representations and vice versa (i.e., lowering and lifting scripts).
- *Semantically-enriched WSDL Interface:* Data types used by Web service interfaces (WSDL) need to be linked to corresponding grounding scripts that mediate data between standard SOAP messages and semantic goals (RDF).

We utilize Semantic Web technologies to cope with inherent dynamics of open enterprise systems and to keep the environment manageable. In particular, we use the WSMX [Haller et al., 2005] platform to enable

- *Cross-Organizational Abstraction.* Since members from various domains and organizations need to interact, we use Semantic Web Services as an abstraction from organizational structures in order to distribute communication facilities. Typically members of virtual communities use their organizations' resources and infrastructure; Web services resolve the need (semantic goal) of interaction to actual SOAP requests and additionally mediate between differing ontological concepts.
- *Context-aware Interaction Channel Selection.* Selecting appropriate communication, coordination, and collaboration service does not only depend on functional needs, but also on contextual constraints. For instance, the delivery of a message (described by a semantic goal) can be achieved through e-mail services, instant messaging, or postings in Internet forums. The appropriate channel can be selected based on user data (location, privacy rules) and messages (priority, size).

2.1.3. *Collaboration Monitoring Layer*

Interactions are observed and collected to determine social relations. We designed the system to manage relations by evaluating occurring interactions and therefore, unburden network participants – at least partly – from managing their relations manually. Logging invocations of collaboration services is the basis for advanced interaction analysis, and allows to infer social relations that are described by objectively measured metrics, such as average response times, availability, or reciprocity.

Formally, a virtual community is a special kind of social network, where the single actors participate to perform activities. A community is modeled as a directed graph, where vertices V represent the actors that are connected through edges E. A directed edge from actor u to v is denoted as e_{uv}. Activities A are a fundamental part of our model; thus, we describe the graph model of a community as $G = (V, E, A)$. The concept of an activity $a \in A$ is used to include a set of participants. Thus, in short, activities describe the collaboration boundaries and goals. Network members interact in scope of particular activities (i.e., to reach certain goals). Interactions are collected to determine (i) the center of interest of single network members by evaluating the frequency of used keywords [Schall and Dustdar, 2010; Skopik et al., 2010a], and (ii) the strength of a social relation by determining the similarity of the center of interests [Skopik et al., 2010d]. Since these techniques have been extensively discussed in previous work, we do not present a detailed description in this paper.

2.1.4. *Social Network Discovery and Information Sharing*

In our framework, an actor has several *passive* links, modeled as FOAF relations, that express business/personal contacts (typically emerged from previous collaborations), but not describing that interactions are performed along these links. An actor can *activate* these links by initiating a new collaboration, e.g., setting up a joint activity. However, due to resource constraints, members can only participate in a limited amount of concurrent activities, and thus, the number of simultaneously active links is limited. Hence, collaboration partners are discovered and selected carefully, considering required effort and received benefit.

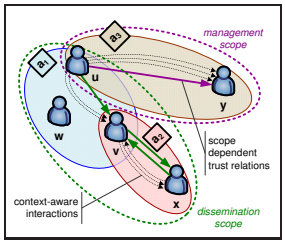
Direct relations are established to create a typical social network. Since single members usually build up strong relations to only a small amount of partners, reliable information flows through collaborative networks, such as exchanging expertise and interest profiles, are limited. Thus, the discovery layer allows actors to exchange business contacts by sharing and propagating (parts of) profiles over intermediate nodes. Each actor's connectivity to other community members is determined by issuing keyword-based queries [Schall and Skopik, 2010] denoted by the query context Q. The query context is described by a pool of keywords (e.g., describing certain expertise areas) picked from global taxonomies. Using logged interaction data (and additional manual ratings) the link weight from one actor to another is calculated using a *social trust* metric that is discussed in detail in the next section.

2.2. *The Science Collaboration Scenario*

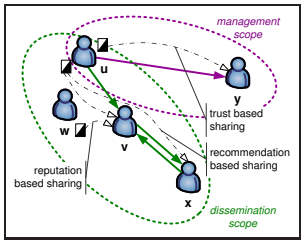
A typical environment for applying our concepts is a *science collaboration network*. It comprises scientists, members from national and international research labs, and experts from the industry. Collaboration is supported by modern service-oriented architectures that realize centralized people registries and profile management, communication services, and data sharing facilities. Network members collaborate to

address challenging research questions and to reach higher impact of scientific disseminations. They conduct joint project proposals, perform distributed software prototyping, and data analysis and visualization. Furthermore, certain participants can provide their support in a service-oriented manner. For instance, they offer document review services, or data analysis services, and interact through precisely predefined interfaces. We utilize the previously introduced Human-Provided Services (HPS) framework [Schall et al., 2008b] to embed humans acting as services using SOA concepts. This includes WSDL descriptions of interfaces, central registries, SOAP-based interactions, and sophisticated logging facilities.

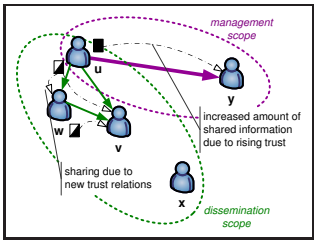
(a) Scenario resources and artifacts.



(b) On the emergence of trust.



(c) Information sharing.



(d) Evolution of relations.

Fig. 3. Information sharing upon dynamically changing trust relations in flexible collaborations.

2.2.1. *Emerging Trust Networks*

We demonstrated the (semi-)automatic flexible determination of trust in the above-mentioned service-oriented collaboration environment in detail earlier [Skopik et al., 2010a]. Briefly, our approach relies on the observation of fundamental interactions, such as SOAP-based communication, coordination or execution messages. People interact and use services when conducting activities. Figure 3(b) depicts this fundamental concept. Network members collaboratively perform activities of different types. These activities structure relevant contextual information, including involved actors, goals, temporal constraints, and assigned resources. So, we conclude that an activity holistically captures the context of interactions between participants [Schall et al., 2008a]. Several activity contexts are aggregated to uniform scopes, e.g., all activities of a specific type (activity scope), or all activities belonging to a certain project (project scope). Trust emerges from interactions and manual ratings of collaboration partners within those scopes. For instance, trust can rely on the responsiveness and reliability of collaboration partners, as well as on their collected experiences and skills. As shown in Figure 3(b), trust is represented by a

directed relation from one network member, e.g., u (the *trustor*) to another one v (the *trustee*), and relies on prior cooperative behavior in a given scope. These trust relations are determined by periodically analyzing and interpreting observed interactions and ratings of partners. For example, the collaboration of network members u, v, w, and x in different scientific dissemination activities a_1 and a_2, leads to the establishment of trust in one uniform 'dissemination scope'. Finally, a scale-free complex network emerges from cooperations in typical research collaborations as investigated by [Reka and Barabási, 2002].

2.2.2. *On Trusted Information Sharing*

In a science collaboration network scenario, understandably no member will share novel, yet unpublished, ideas carelessly. However, information sharing is essential to discover new collaboration opportunities. The challenge is to enable sensitive information sharing, that adapts and restricts the view on information with respect to changing trust relations. Therefore, we introduce the concept of *trusted information sharing*. This concept provides the means to share information, e.g., paper drafts, recently submitted papers, or project documentation, only with trusted network members who have demonstrated their reliable and dependable behavior before. In this case, trust reflects a probability measure of future collaboration successes, and therefore, potential benefits from collaborations.

As depicted in Figure 3(c), trusted information sharing is bound to trust scopes. For instance, if member u established trust in y in the management scope (because they jointly performed several project management activities successfully), y is allowed to access u's data about referees' contact details, planned future projects, and personal organizational details. However, no information dedicated to other scopes, such as scientific dissemination, is shared. Hence, information sharing is restricted to mandatory information in particular scopes.

As trust relations emerge dynamically based on interaction behavior of people, the amount of shared information is periodically adapted by the system and, in the optimal case, needs no further manual intervention of users. However, this approach works best in environments with flat (or practically no) hierarchies, where people may decide completely on their own about conditions for information sharing. In enterprise collaborations, with pre-determined communication paths and static role models, mechanisms that override trust-based sharing are required. But here, we focus on the depicted science collaboration network that consists of people with equal roles, rights and aims. We identified three fundamental trust concepts to enable trusted information sharing in the described environment:

Sharing based on Personal Trust Relations. Activity relevant artifacts are shared in a scope to different extent (views), according to the degree of trust between network members. For instance, in Figure 3(c) u grants access to y to information in the management scope.

Sharing based on Recommendations. In case of sparse trust networks, or low

connectivity of certain members, sharing based on personal relations only is limited. Second-hand opinions, called recommendations, are utilized to overcome this problem. For instance, u trusts v, and w trusts x because of successful previous collaborations in the dissemination scope. If these successes rely on the compatibility of each member's working style, there is a high probability that n_1 might establish trust to x upon future interactions (for transitive trust propagation see [Guha et al., 2004]). Hence, to facilitate the establishment of trust relations, u is encouraged to share pieces of information with the unknown member x. Sharing of data, such as parts from the personal profile, introduces u to x and supports the bootstrapping of future collaborations [Ziegler and Golbeck, 2007].

Sharing based on Reputation. If network members are trusted by several partners in the same scope, (i.e., they have more than one trustor), reputation can be determined. For instance, v is trusted by u and x. Therefore, network member w, who has not established trust in others yet, can rely on this reputation (inferred from single trust relations). So, w can either allow v to access parts of his personally managed information (passive sharing), or by pushing information towards v (active sharing).

2.2.3. *Evolution and Aging of Trust*

Since network members may change their interaction behavior over time, for instance, their goals and priorities shift or they start to develop interests in new fields, trust relations have to be altered too. However, trust relations can become even closer through successful long-term collaborations. These dynamics are indicated from Figure 3(c) to Figure 3(d). Here, trust from u to v has been increased, thus u grants v even more access to his personal files (see the filled document symbol). While trust from u in w and w in v has emerged due to closer collaboration, the relations from and to x have been removed.

Especially in our science collaboration scenario, it is inevitable to consider these trust dynamics. Imagine, someone suddenly shows unreliable behavior, e.g., does not answer support requests or does not fulfill his assigned activities any longer. In that case also the access to critical information has to be restricted. In this paper we discuss approaches to detect misleading behavior changes to guarantee timely updates of relations in a managed *Web of Trust*. However, not only existing relations are adapted, but new relations are introduced and outdated relations removed.

3. Social Network Management

This section discusses a framework enabling distributed profile management in large-scale Web-based open enterprise systems. Profiles are shared among members and evaluated to discover potential collaboration opportunities based on interest similarities, coverage of expertise needs, project participation, and organizational memberships.

3.1. *Architectural Overview and Design*

Since information sharing with mostly unknown individuals in large-scale environments is a delicate matter, our framework applies common security standards to encrypt sensitive information and therefore, enables selective sharing of information. We adopt one of the most popular encryption concepts, in particular *public key infrastructure* (PKI) [Adams and Lloyd, 1999] for that purpose.

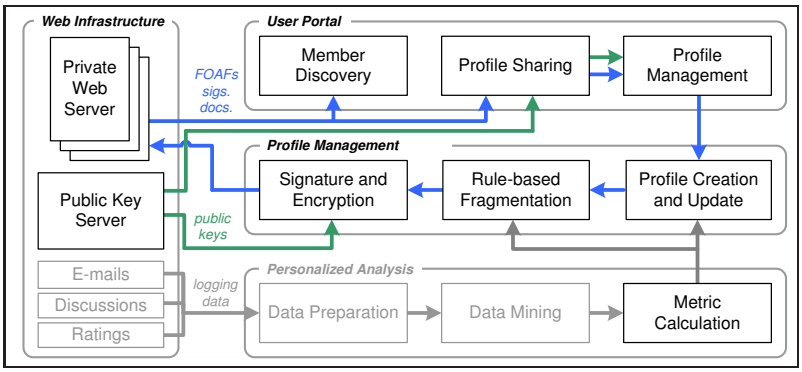


Fig. 4. Architecture supporting discovery in self-managed social networks.

The fundamental architecture of our framework is depicted in Figure 4. Basically, the left side consists of globally available components, such as various Web servers owned by individuals and organizations, public key servers, and collaboration tools hosted in a semantic Web services environment, including e-mail infrastructures, discussions forums, and rating platforms. The right side comprises distributed components that are replicated for each user (and groups of users forming closed communities respectively) to manage their profiles from their personal point of view. The architecture consists of the following three layers: (i) *Personalized Analysis* enables data aggregation from collaboration tools and data mining to determine collaboration relations. Basically, the strength of social relations is inferred by calculated various interaction and behavior metrics from mining e-mail data or Internet forum entries [Schall and Dustdar, 2010; Skopik et al., 2010a]. (ii) *Profile Management* includes features to semi-automatically create and update FOAF profiles with calculated metrics. Profiles are encrypted and valid signatures created so that only close collaboration partners can decrypt and use them for discovering actors. (iii) the *User Portal* hosts tools to discover potential partners and sharing and managing personal profiles.

3.2. *Automatic Social Trust Inference*

We believe that trust and reputation mechanisms are key to the success of open dynamic service-oriented environments. However, trust is emerging based on evidence, i.e., interaction behavior. Interactions, for example, may be categorized in terms of

success (e.g., failed or finished) and importance. Therefore, a key aspect of our approach is the monitoring and analysis of interactions to automatically determine trust. We argue that in large-scale SOA-based systems, only automatic trust determination is feasible. In particular, manually assigned ratings are time-intensive and suffer from several drawbacks, such as unfairness, discrimination or low incentives for humans to provide trust ratings.

Trust Definition. In contrast to a common security perspective, social trust refers to the interpretation of previous collaboration behavior [Skopik et al., 2010a] and the similarity of dynamically adapting interests [Golbeck, 2009; Skopik et al., 2010d]. Especially in collaborative environments, where users are exposed to higher risks than in common social network scenarios [Dwyer et al., 2007], and where business is at stake, considering social trust is essential to effectively guide interactions. Much research effort has been spent on defining and formalizing trust models (for instance, [Guha et al., 2004; Ziegler and Lausen, 2005]).

Here, we define trust as follows: *Trust reflects the expectation one actor has about another's future behavior to perform given activities dependably, securely, and reliably based on experiences collected from previous interactions.*

Interaction Metrics. In order to support the emergence of social relations, we utilize the following two metrics:

Interest Similarity isim. This metric determines the overlap of actor interests, which is an important measure to find motivated partners in the same interest area. We manage keywords used by actors u and v as interest profile vectors $\mathbf{p}_u$ and $\mathbf{p}_v$ respectively (see [Skopik et al., 2010d] for details), and determine the similarity of profiles through the cosine between their profile vectors (Eq. 1). The result is a value between 0 (no overlap) and 1 (full overlap).

$$isim(u,v) = cos(\mathbf{p}_u, \mathbf{p}_v) = \frac{\mathbf{p}_u \cdot \mathbf{p}_v}{|\mathbf{p}_u||\mathbf{p}_v|} \tag{1}$$

Reciprocity recpr. A typical social behavior metric is reciprocity [Falk and Fischbacher, 2006] that here reflects the ratio between obtained and provided support in a community. Let $REQ(u,v)$ be the set of u's sent support requests to v, and $RES(u,v)$ the set of u's provided responses to v's requests. Then we define reciprocity in $[-1,1]$ as in Eq. 2; hence, 0 reflects a balanced relation of mutual give and take.

$$recpr(u,v) = \frac{|RES(u,v)| - |REQ(u,v)|}{|RES(u,v)| + |REQ(u,v)|} \tag{2}$$

The actual strength (weight w respectively) of a social trust relation is determined by normalizing, combining and weighting these metrics whenever a discovery process is started, i.e., a query issued. While *isim* is a globally valid metric, *recpr* is bound to distinct contexts Q (e.g., expertise areas). In particular, interactions bound to all activities whose description match at least one of the query keywords issued for discovering neighbor nodes are considered when

calculating *recpr*.Currently we employ flat keyword-based matching only, however for more advanced ontology matching techniques see [Castano et al., 2006; Euzenat and Shvaiko, 2007]. Eq. 3 allows for the balancing between two cases: (i) *newcomer support* versus (ii) weighting of links of *well established* actors (based on evidence). The factor α can be adjusted based on the requirements for each case. For example, by setting $\alpha = 1$, newcomer support becomes more dominant since $isim_{uv}$ accounts for interest (profile) similarities. Whereas, the other case with $\alpha = 0$ puts stronger emphasis on already established links by accounting for the preference towards existing relations.

$$w^Q(u,v) = \alpha \cdot isim(u,v) + (1-\alpha) \cdot recpr^Q(u,v) \tag{3}$$

3.3. *TBAC - Trust based Access Control*

Trust Bases Access Control (TBAC) supports the discovery of collaboration partners and subsequently the formation of groups and networks in open enterprise systems using distributed profile information. The main idea is to allow actors to access the profiles of other network members based on the strength of social relations, e.g., social trust. In other words, only trustworthy partners are allowed to access, in particular read, someone's personal profile information. Key principles of the proposed approach are:

- *Self-managed Distributed Profiles.* Actors manage their personal profiles in a distributed manner, i.e., profiles are fully under control of the respective actors.
- *Public and Private Scopes.* Some profile information may be available public, for instance, expertise area and basic contact details in order to discover new collaboration partners. However, access to sensitive information, e.g., private contact details and friend relations, is restricted.
- *Social Trust-based Access Control.* Access to private fragments of profiles is granted based on strengths of social relations. For instance, close collaboration partners can read larger parts of an actor's profile. Social trust relies on interactions and an update of personal relations can be triggered by actors using logged information from the SOA infrastructure. Note, only logged interactions with personal involvement are used.
- *Public Key Infrastructure (PKI).* PKI is the means to enable public and private profile scopes and to address privacy concerns in open environments.

Transitive Access. As in the real world, information is not only shared between direct neighbors, but can traverse several intermediate nodes. Using this approach allows sharing of profiles along trusted paths even if actors are not directly connected in the social network. This spreading of information relies on the principle of recommendation and propagation of trust respectively [Guha et al., 2004]. Since

all involved parties are connected with a strong trust path, privacy is still maintained. Transitive access is an important concept to overcome inherent limitations of trust based discovery only.

4. Specification and Implementation of Social Overlay Networks

This section deals with the specification and implementation of the proposed social overlay model to realize dynamic discovery in semantically-enriched collaborative open enterprise systems.

4.1. *Adaptive Distributed Profile Management*

The mainly applied techniques are the Friend-Of-A-Friend (FOAF)[d] ontology, Public Key Infrastructure, in particular GnuPG[e], and Web-Of-Trust (WoT)[f] schemas.

4.1.1. *Friend-Of-A-Friend Profile Management*

Various concepts and protocols have been proposed to manage open social and collaborative networks in a distributed manner. The Friend-Of-A-Friend (FOAF) concept is one of the most popular ones on the Web.

```
<?xml version="1.0"?>
<rdf:RDF xmlns:foaf="http://xmlns.com/foaf/0.1/"
       xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
       xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
       xmlns:foaf="http://xmlns.com/foaf/0.1/"
       xmlns:dc="http://purl.org/dc/elements/1.1/"
       xmlns:wot="http://xmlns.com/wot/0.1/"
<foaf:Person rdf:ID="me">
 <foaf:name>Florian Skopik</foaf:name>
 <foaf:nick>florian</foaf:nick>
 <foaf:mbox_sha1sum>a4b378...</foaf:mbox_sha1sum>
 <wot:haskey rdf:nodeID="KeyFS" />
 <foaf:interest rdf:resource="http://..." />
 <foaf:currentProject>
  <foaf:Project>
   <dc:title>Implementation Module X</dc:title>
   <dc:description>WS, programming, java</dc:description>
   <dc:identifier rdf:resource="http://.../activity#4539"/>
  </foaf:Project>
 </foaf:currentProject>
 <foaf:knows>
  <foaf:Person>
   <foaf:mbox_sha1sum>1a4578...</foaf:mbox_sha1sum>
   <foaf:name>Daniel Schall</foaf:name>
  </foaf:Person>
 </foaf:knows>
</foaf:Person>
</rdf:RDF>
```

Listing 1. Example of public FOAF file.

[d] http://xmlns.com/foaf/spec/
[e] http://www.gnupg.org
[f] http://xmlns.com/wot/0.1/

It allows to model user properties, interests and relations with a well-known ontology. We apply FOAF to facilitate the discovery process used to find potential collaboration partners.

Listing 1 shows a simplified example of a public FOAF profile, containing basic personal properties (`name`, `nick`, `interest`) and social relations (`knows`). The Web of Trust (WoT) RDF ontology is used to integrate concept of a public key infrastructure into FOAF profiles, as demonstrated in Listing 2. The property `haskey` links a public key (`pubkeyAddress`), `hex_id`, and `fingerprint` to a `person`. Furthermore, a person's private key is used to sign the own FOAF profile and therefore, to guarantee for integrity and authenticity. Notice, the only guarantee regarding authenticity is that the FOAF signer is owner of the registered mail account that has been used to create the key pair.

```
<!-- restricted part of FOAF profile -->
<rdfs:seeAlso>
 <foaf:Document rdf:about="http://.../foaf-private.rdf.asc">
  <wot:encryptedTo>
   <wot:PubKey wot:hex_id="34c5a421b" />
  </wot:encryptedTo>
 </foaf:Document>
</rdfs:seeAlso>

<!-- digital signature for this file -->
<rdf:Description rdf:about="">
 <wot:assurance rdf:resource="foaf.rdf.asc" />
</rdf:Description>

<!-- public key of the owner/signer of this file -->
<wot:PubKey rdf:nodeID="KeyFS">
 <wot:hex_id>3756EA0B</wot:hex_id>
 <wot:length>1024</wot:length>
 <wot:fingerprint>03f4...</wot:fingerprint>
 <wot:pubkeyAddress rdf:resource="http://.../key.asc"/>
 <wot:identity>
  <wot:User>
   <foaf:name>Florian Skopik</foaf:name>
   <foaf:mbox_sha1sum>a4b378...</foaf:mbox_sha1sum>
  </wot:User>
 </wot:identity>
</wot:PubKey>
```

Listing 2. Signing FOAFs (`wot:assurance`) and linking encrypted content (`rdfs:seeAlso`).

Access to parts of a FOAF document may be restricted to certain users (whose public keys are used to encrypt those parts). We utilize this concept for (i) private information, such as private phone numbers or chat accounts that can only be decrypted and used by close neighbors (connected via `knows`), and (ii) personal ratings that are given either *explicitly* (manually) or *implicitly* (through data mining of e-mail logs, instant messaging (IM) logs, or Internet forums). We understand privacy as a major concern when applying mining techniques; hence, mining of metrics is performed from each actor's perspective (or at least limited to certain groups of experts). This means that data is not stored centrally but managed on the client side and private servers, such as e-mail servers and private Web forums.

Listing 3 depicts an example of encrypted private FOAF fragments. While users decide manually which parts of their profiles are shared globally and which are

restricted to neighbors only, relation metrics, e.g., derived from personal ratings, are managed automatically by the system. For that purpose, single ratings are stored in a dedicated document (`tipjar`) for each user. This document is processed by various evaluation tools and plugins that are fully under control of the users. Currently, we have three tools for (i) collecting manual ratings, (ii) analyzing Internet forums, and (iii) analyzing e-mail communication in order to assess collaboration performance of known partners and the strength of social ties based on past interactions. For that purpose, we adopt a rating ontology[g] for TV programs to store a personalized view of known people, expressed through manual ratings and mined metrics.

```
<rdf:RDF xmlns:foaf="http://xmlns.com/foaf/0.1/">
       xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
       xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
 <foaf:Person>
  <!-- mbox_sha1sum links to public FOAF profile -->
  <foaf:mbox_sha1sum>a4b378...</foaf:mbox_sha1sum>

  <!-- private contact details -->
  <foaf:mbox rdf:resource="mailto:skopik@....tuwien.ac.at"/>
  <foaf:phone>+43 xxxx xxxx</foaf:phone>

  <!-- private chat account -->
  <foaf:account>
   <foaf:OnlineAccount>
    <rdf:type rdf:resource="http://.../OnlineChatAccount" />
    <foaf:accountServiceHomepage rdf:resource="http://..../" />
    <foaf:accountName>florian_skopik</foaf:accountName>
   </foaf:OnlineAccount>
  </foaf:account>

  <!-- attach personalized ratings to known persons -->
  <foaf:knows>
   <foaf:Person>
    <foaf:mbox_sha1sum>1a4578...</foaf:mbox_sha1sum>
    <foaf:tipjar rdf:resource="http://..." rdfs:label="ratings"/>
   </foaf:Person>
  </foaf:knows>
 </foaf:Person>
</rdf:RDF>
```

Listing 3. Private fragment of a FOAF profile.

4.1.2. *Profile and Information Sharing*

The presented concepts enable the discovery of directly connected partners based on common properties, interests, ratings, and contextual constraints (such as projects), but still preserve their privacy. This means that profile owners encrypt sensitive parts of their profiles for their known neighbors, i.e., using their public keys. Since we do not only manage binary knows relations but also calculate the strengths of relations (e.g., social trust), the amount of shared information can be bound to certain strength levels. For instance, whenever one updates his profile, a rule-based system decides based on predefined link thresholds, who is allowed to read private FOAF fragments and encrypt files accordingly.

[g] `http://www.tvblob.com/ratings/#`

However, single members usually build up strong relations to only a small amount of partners. That hinders the discovery process. In order to overcome that hurdle, we allow propagation of information over several intermediate hubs along strong social paths. Enabling such flows of information enables actors to discover new potential collaboration partners. Technically, we allow actors to link *private* profile information of well connected partners as personally encrypted documents to their own profile. Restricted access is the basis for personalized and reliable sharing of information. We use once more the WoT ontology to link external documents to one's FOAF profile (see excerpt in Listing 4). A detailed implementation perspective regarding processing of XML data is out of scope of this paper, but has been investigated in detail in [Skopik et al., 2010b]. A semantically-enriched Web Service based environment allows to notify partners about updated profiles and send them links to encrypted documents. The receivers are able to validate these documents, i.e, verify the authenticity and consistency using the signer's public key and to decrypt information using their own private keys.

In the same manner, confidential information in scope of the science collaboration use case (see Section 2) can be linked to personal profiles and encrypted for particular collaboration partners. The Trusted Information Sharing framework, a Web-based application which enables convenient access to linked profile data by automating the attachment and extraction of XML-based information in the social overlay network, is introduced in Section 5.

```
1  <!-- link encrypted document -->
2  <foaf:Document rdf:about="http://.../foaf47.rdf">
3   <dc:title>Restricted Information</dc:title>
4   <wot:assurance>
5     <wot:Endorsement rdf:about="http://.../foaf47.rdf.asc">
6       <dc:title>signature of friend47 private profile</dc:title>
7       <wot:endorser rdf:nodeID="KeyFS"/>
8     </wot:Endorsement>
9   </wot:assurance>
10 </foaf:Document>
11
12 <!-- encryption information -->
13 <wot:EncryptedDocument rdf:about="http://.../foaf47.rdf.asc">
14  <dc:title>friend47 private profile</dc:title>
15  <wot:encryptedTo rdf:nodeID="KeyPartnerX"/>
16  <wot:encrypter rdf:nodeID="KeyFS"/>
17 </EncryptedDocument>
```

Listing 4. Linking encrypted documents in FOAF.

4.2. *Semantic Service Infrastructure*

WSMX (Web Service Modeling eXecution environment) [Haller et al., 2005] allows to describe and register Web services and thus, supports discovering, selecting, and invoking Web services at run-time in a semantic manner. The actual services are hosted elsewhere, but WSMX builds a semantic abstraction layer for these services by managing additionally required artifacts (as described in Section 2). The WSMX platform provides a WS entrypoint to submit semantic goals that need to

be fulfilled and the platform itself discovers the best suitable service based on (i) functional properties (FPs), i.e., supported concepts, such as messaging; and (ii) non-functional properties (NFPs), here, contextual constraints including organizational boundaries, people's location and working context.

4.2.1. *Registering Semantic Web Services*

The first step of registering a common Web service with a WSDL interface in WSMX is to annotate appropriate lowering- and lifting scripts. These XSLT scripts enable the transformation between SOAP messages and ontological representations. Listing 5 shows a small excerpt of a semantically-enriched WSDL file. Here, the complex data type `sendMessageKey` (and its corresponding response) have `loweringSchemaMapping` and `liftingSchemaMapping` respectively attached. Listing 6 shows a lowering script. Here, values of required semantic concepts to build an instance of type `sendMessageKey` are extracted from the enterprise collaboration ontology.

```
<xs:element name="sendMessageKey"
 sawsdl:loweringSchemaMapping="SendEmailMessage-lowering.xslt">
 <xs:complexType>
  <xs:sequence>
   <xs:element minOccurs="0" name="to" type="xs:string"/>
   <xs:element minOccurs="0" name="subject" type="xs:string"/>
   <xs:element minOccurs="0" name="body" type="xs:string"/>
   <xs:element minOccurs="0" name="key" type="xs:string"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>
<xs:element name="sendMessageKeyResponse"
 sawsdl:liftingSchemaMapping="SendEmailMessage-lifting.xslt">
 <xs:complexType>
  <!-- details omitted -->
 </xs:complexType>
</xs:element>
```

Listing 5. Schema mapping annotations in WSDL.

```
<xsl:template match="rdf:Description[rdf:type/@rdf:resource=
'http://www.coin-ip.eu/ontologies/ec#EmailServiceMessage']">
 <email:sendMessageKey>
  <xsl:for-each select="ecg:hasEmailAddress">
   <to><xsl:value-of select="."/></to>
  </xsl:for-each>
  <xsl:for-each select="ecg:hasSubject">
   <subject><xsl:value-of select="."/></subject>
  </xsl:for-each>
  <xsl:for-each select="ecg:hasContent">
   <body><xsl:value-of select="."/></body>
  </xsl:for-each>
  <xsl:for-each select="ecg:hasAuthenticationKey">
   <key><xsl:value-of select="."/></key>
  </xsl:for-each>
 </email:sendMessageKey>
</xsl:template>
</xsl:stylesheet>
```

Listing 6. Lowering script example.

4.2.2. *Semantic Goal Description*

Listing 7 shows exemplarily a goal defined in WSML[h] for sending a notification via e-mail. For that purpose, NFPs are defined (here: type of discovery), as well as pre- and postconditions for invoking a capable Web service (e.g., defined recipient and message). The block `instance emailRequest` contains the actual parameters that are lowered to a SOAP message and sent to an Email Web service.

```
wsmlVariant _"http://www.wsmo.org/wsml/wsml-syntax/wsml-rule"
namespace { _"http://www.coin-ip.eu/goals/ec#",
disc _"http://wiki.wsmx.org/index.php?title=DiscoveryOntology#",
ec _"http://www.coin-ip.eu/ontologies/ec#",
ecp _"http://www.coin-ip.eu/ontologies/ecp#"}

goal MessageGoal
 importsOntology {
  ec#EnterpriseCollaborationOntology,
  ecp#EnterpriseCollaborationProcess
 }

capability MessageGoalCap
 nonFunctionalProperties
  disc#discoveryStrategy hasValue disc#NoPreFilter
  disc#discoveryStrategy hasValue disc#HeavyweightDiscovery
 endNonFunctionalProperties

sharedVariables {?x, ?z, ?y}

precondition MessageGoalPre
 definedBy
  ?x memberOf ec#EmailMessage and
  ?z memberOf ec#Individual and
  ?y memberOf ec#Individual.

postcondition MessageGoalPost
 definedBy
  ecp#messageSent(?z, ?x, ?y).

ontology EmailRequest
 importsOntology {
  ec#EnterpriseCollaborationOntology
 }

instance emailRequest memberOf ec#EmailMessage
 hasAuthenticationKey hasValue "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxx"
 hasEmailAddress hasValue "name@infosys.tuwien.ac.at"
 hasSubject hasValue "Notification about project opportunity"
 hasContent hasValue "Dear sir, according to your profile ..."
```

Listing 7. Semantic goal for e-mail message service.

5. The Trusted Information Sharing Framework

We describe our *Trusted Information Sharing* framework that has been first introduced in [Skopik et al., 2010b] and extend it to be used on top of the social overlay network in order to meet requirements of the science collaboration scenario discussed earlier in this work. We distinguish between two *modes of sharing*: (i) Activity-centric sharing accounts for the currently jointly processed activity of u

[h]Web service modeling language

and v. Therefore, information is shared to foster ongoing collaborations. (ii) Scope-centric sharing is about information sharing due to trust in a scope, but without accounting for a concrete activity. This kind of sharing is useful to facilitate future collaborations, i.e., the creation of new joint activities.

Besides the modes we distinguish two different *sharing styles*: (i) Active Sharing pushes information to actual or potential collaboration partners (depending on the sharing mode), e.g., a call for paper via notification and announcement services. (ii) Passive Sharing grants access to personal information when requested by other network members, e.g., when the collaboration network is searched for dissemination opportunities. We focus on the latter kind of sharing style that can be understood as a dynamic access control system.

5.1. *Architectural Overview*

The main components of our framework and their connections are depicted in Figure 5. The backend services comprise one or more *Information Repositories* that hold various kinds of information, encoded in XML and defined by XML Schemes (XSDs). A potential repository is further a set of FOAF profiles with linked external information. An *Information Catalog* enables users to link information from repositories to sharing scopes. Activities, as introduced in our motivating scenario, are managed by an *Activity Management Service* and act as the glue for multi-dimensional collaboration data. Especially trust relations that emerge from interactions between users during the execution of activities, are provided by the *Trust Network Provider* which extracts these data offline in periodic intervals from registered FOAF profiles. A *Sharing Rule Management Service* provides trust requirements

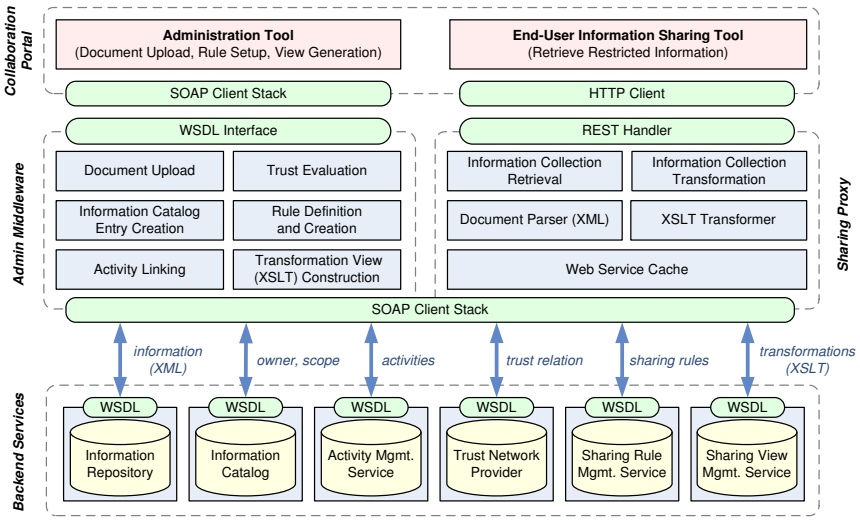


Fig. 5. Sharing framework overview.

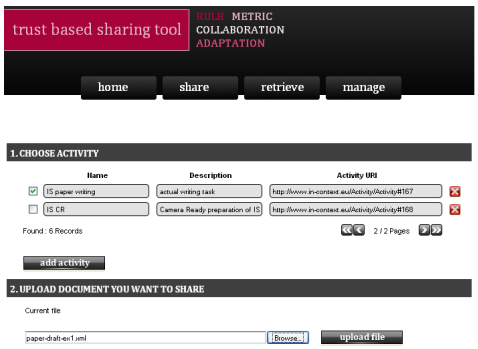
for information sharing, e.g., a minimum degree of personal trust or reputation, and the *Sharing View Management Service* stores transformation scripts (XSLTs) to customize the view on XML-encoded information.

The *Administration Middleware* is utilized by users to register potentially shared information in the platform. This component provides all features to enable the upload of XML documents, the creation of catalog entries to register the ownership of information and sharing in context of activities, the retrieval of interaction metrics and trust values of relations to information consumers, and the definition of sharing rules and restricted views on the uploaded documents. In the end-user collaboration portal an *Administration Tool* is provided, that communicates with the Administration Middleware. Users can register and unregister information, and create and modify their sharing rules. Furthermore, they have the ability to register new information types (XSDs).

The *Sharing Proxy* enables users to retrieve information from collaboration partners. This component utilizes all the aforementioned SOAP-based backend services and restricts information based on sharing views picked by evaluating sharing rules. Technically, this is realized by transforming XML data through applying XSLTs depending on trust relations between information owner and requester. Higher trust allows more details to be shared. Since the Sharing Proxy has to serve many concurrent requests and heavily relies on SOAP-based Web services in the backend, we integrated a Web service cache that buffers all responses from the backend. The configuration of cache update intervals is closely linked to the volatility of trust relations and, hence, to update and aging mechanisms discussed before. The *End-User Information Sharing Tool* provides a convenient user interface that enable people to browse the Web of Trust and retrieve information that is shared by collaboration partners. Each user of this tool retrieves his/her individually restricted view on shared information based on their personal relations.

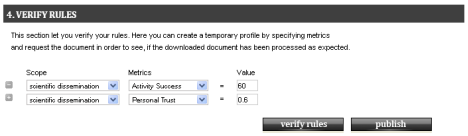
5.2. *Administration Tool*

Figure 6 shows the end-user's perspective of trusted information sharing. In the first step, as depicted in Figure 6(a) the user picks ongoing activities from a list where s/he wants to publish information. Second, the user uploads the actual document. The document content is modeled as an XML structure and follows a specific schema (XSD). In our example, the user shares a paper draft consisting of title, authors, abstract, keywords, and body, within a dissemination activity. After uploading the document, it is parsed in the administration middleware and all available XML tags are extracted. Then (Figure 6(b)), users are able to define sharing rules on these XML tags. All uploaded information is shown to others by default if no further restrictions are defined. Let us assume for the depicted example that the owner of the paper draft only wants close collaboration partners to see participating authors. Thus, after upload the user restricts access to the author section of the paper draft. A constraint is for instance that a certain requester of the document need to be



(a) Selection of activities for information sharing and upload of potentially shared information (XML document). TIS establishes a connection to registered FOAF profiles and adapts them accordingly.

(b) Specification of XML elements to share (=sharing view) and conditions of relation metrics (=sharing rules); Definition of individual views by dynamically creating and parameterizing transformation scripts (XSLTs).



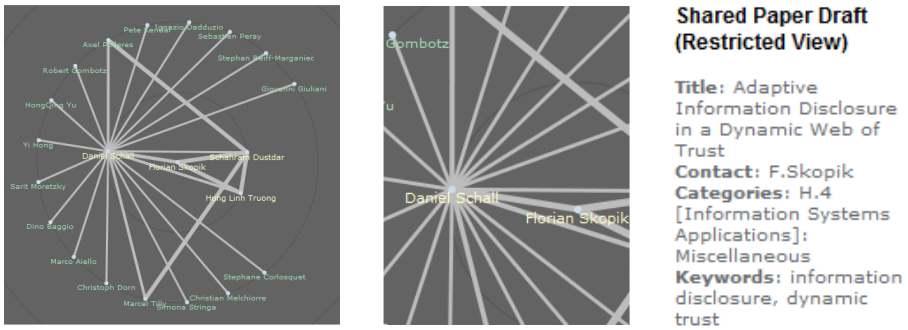
(c) Verification of rules by specifying synthetic relation data (metrics) and final publishing of all required artifacts (information, catalog entry, rules, and view).

Fig. 6. Set up sharing of document-based information with trustworthy collaboration partners.

personally trusted by the document owner with a value equal or higher than 0.5 ($\tau \in [0, 1]$). (Note, values and limits can be set upon best practices or suggestions from domain experts – see [Skopik et al., 2010c]). The specification of this rule produces two artifacts: (i) The *sharing view* is an XSLT that transforms the initial paper draft to a version with an omitted authors section. (ii) The *sharing rules* model constraints based on relation metrics for transforming the document. Thus, whenever someone who is not personally trusted with $\tau \geq 0.5$ requests the paper draft, s/he only receives a version without the authors section. Finally, as shown in Figure 6(c), the effects of configured rules can be verified by the document owner. For that purpose, artificial metric values can be specified and the document retrieved in its restricted version for the role of a document consumer, i.e., interested collaboration partner. Publishing the document means that the paper draft is stored in an information repository, a catalog entry is produced that links the document to certain activities, and generated views and rules are deployed in the respective backend services.

5.3. *End-User Information Sharing Tool*

The Web-based tool for exploring shared information is shown in Figure 7. The user is able to explore his/her network visualized as (undirected) graph. The collaboration network is established based on past interactions as discussed previously. The first view in Figure 7(a) shows a personalized view on the collaboration network (i.e., based on the member with most connections to other members). Users with just one single connection within the network are visualized in a different color. The link weight is proportional to the number of interactions between net-



(a) Browse network members. (b) Information discovery. (c) Shared view.

Fig. 7. Browsing shared information in a Web of trust.

work members, thus being a direct indicator for the level of trust. As the next step (see Figure 7(b)), a user explores shared information from another member. Again, the link weight and trust restrict how much information is shared between network members. An example for a shared (restricted) information view is shown by Figure 7(c). A detailed explanation on applied rules and transformations is given in the next section. We intended to design a lightweight tool for graph visualization and information sharing. The presented tool has been implemented on top of state-of-art Web toolkits and a JavaScript based visualization toolkit[i]. This has the advantage that collaboration networks can be visualized without requiring additional client side libraries or browser plugins.

5.4. *Fundamental Mode of Operation*

We describe the interplay of the components to cover the fundamental use case of trustworthy sharing of a *particular* information (i.e., that is already referenced by an *uri*), of the owner u with the requester v. Let us assume, u explicitly publishes the *uri* of an XML file in a public area of the collaboration platform. User v wants to retrieve this information through the Sharing Proxy (see Figure 5), and view in his/her Browser. That is the point, where *trustworthiness* comes into play. The sequential interactions of the single components are depicted in Figure 8. The process starts with retrieving registered meta-data for the given information, including the owner and valid a scope of sharing. After that, joint scopes are requested from the Activity Management Service, i.e., the scopes of all currently running joint activities. Then, the sharing rules of the information owner are retrieved, as well as existing trust relations in the potential sharing scopes. The Sharing Proxy picks the sharing rule that results in the least restrictive information. This means sharing relies on the tightest available trust relation between owner and requester. According

[i]Visualizations for the Web: `http://thejit.org/`

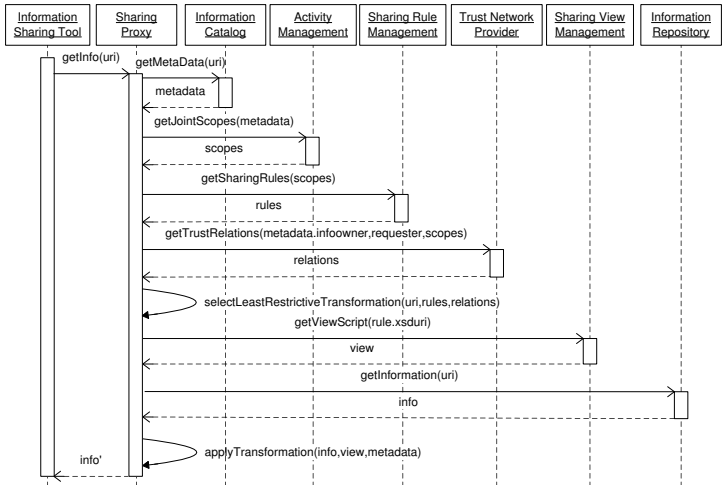


Fig. 8. Fundamental interactions of components when retrieving information.

to the picked rule, the corresponding XSLT script from the Sharing View Management Service is requested, as well as the initially requested information from the Information Repository. Finally, the initially requested information is transformed to its restricted view and delivered to the requester.

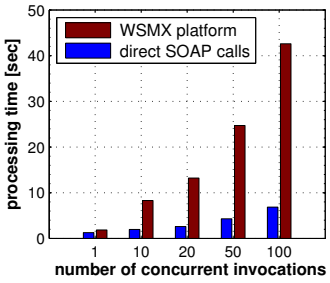
6. Evaluation and Discussion

This section deals with evaluation results regarding the whole system as well as discussions of essential findings. In particular, we demonstrate the performance of semantically-enriched service hosting with WSMX, discuss network formation processes using simulation, study member discovery processes through propagating distributed FOAF profiles, and discuss various design decisions with respect to PKI for FOAF.

6.1. *WSMX Performance Aspects*

The used WSMX setup consists of 38 different Web services, primarily communication services and document management services, 52 ontology parts, and 13 semantic goals (e.g., sending a message with a given content to a particular person). For the following experiments, WSMX and services (implemented using Axis2[j]) are hosted on a server with Intel Xeon 3.2GHz (quad), 10GB RAM, running Tomcat 6 with Axis2 1.4.1 on Ubuntu Linux. Furthermore we perform concurrent calls from a client simulation that runs on a Pentium 4 with 2GB on Windows XP, and is connected with the server through a local 100MBit Ethernet. Figure 9 compares the

[j] `http://ws.apache.org/axis2/`



WSMX step	time [ms]
lowering	250-400
service invoc.	> 130
lifting	150-250
single invoc.	time [ms]
SOAP (avg)	80-120
WSMX (avg)	450-750

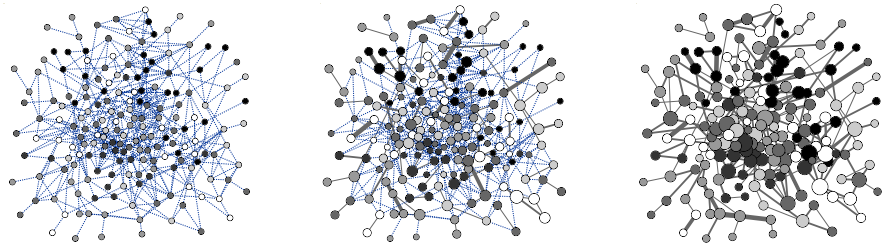
Fig. 9. WSMX performance comparison.

performance of WSMX with standard SOAP calls that invoke Web services directly for different numbers of concurrent calls.. Note, the additional overhead caused by WSMX is the difference between the two results, since after processing the semantic layer also WSMX invokes a particular WS via SOAP only. In our test environment, invoking a service via WSMX compared to invoking the same service directly takes approximately 5 times longer. The additional processing time is used for lowering a request (such as the goal in Listing 7) to a SOAP message, and, after invoking the service, lifting the response back to the semantic level. Although WSMX adds much additional overhead to service invocation, several advantages can be taken, including, dynamic discovery and selection of best suitable service instances (depending on NFPs), and establishing real cross-enterprise interoperability through data mediation on ontological level. Note, services can be distributed over several WSMX instances to distribute load and increase performance.

6.2. *Network Formation Simulation in SOA*

We use a Web service testbed to simulate the interaction behavior in SOA-based communities. The purpose of the Genesis2 framework [Juszczyk and Dustdar, 2010] (in short, G2) is to support software engineers in setting up testbeds for runtime evaluation of SOA-based concepts and implementations. It allows to establish environments consisting of services, clients, registries, and other SOA components, to program the structure and behavior of the whole testbed, and to steer the execution of test cases on-the-fly. G2's most distinct feature is its ability to generate real testbed instances (instead of just performing simulations) which allows engineers to integrate these testbeds into existing SOA environments and, based on these infrastructures, to perform realistic tests at runtime.

Experiment Setup. The created test environment consists of 200 autonomous services that simulate behavior in common flexible collaboration scenarios. Each service (called actor) has an interest/expertise profile assigned, consisting of 5 to 8 distinct keywords. Profiles may partly overlap. In order to bootstrap collaborations links between actors are predicted based on profile similarities. Typically, interest similarities are a reasonable grounding for future collaboration success and emerg-



(a) bootstrapping phase: only predicted links ($\alpha = 1$).

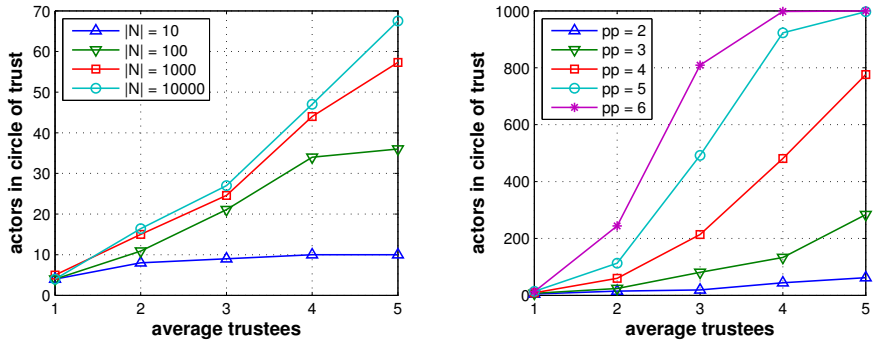
(b) formation phase: mix of predicted and emerged links ($\alpha = 0.5$).

(c) saturation phase: only emerged links ($\alpha = 0$).

Fig. 10. Network formation process visualization.

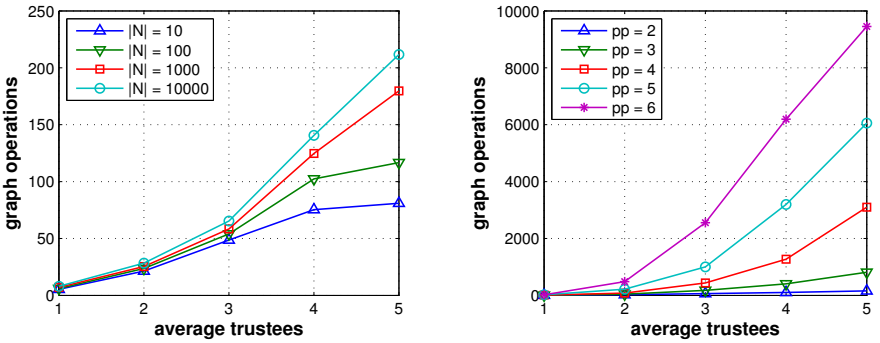
ing personal relations [Ziegler and Golbeck, 2007]. During the actual collaboration single actors interact by delegating tasks and requesting support from other members of the community; thus, in our simulation we let random members interact in fixed time intervals. Each interaction is tagged with a maximum of 3 keywords and sent to actors with matching interest profiles. We run different tests and vary the number of globally known tags, as well as the amount of occurring interactions. The results of these experiments enable us to study the formation process of typical medium scale Web-based communities. In particular we investigate the three phases of (i) bootstrapping, i.e., initiating the formation of a network; (ii) formation phase, i.e., setting up strong links between matching collaboration partners; (iii) saturation phase, i.e., cross-linking emerging small-scale communities with weak links. The aim of this experiment is to determine the effort in terms of monitoring and processing interactions until similar network structures (in the respective evolutionary phases) for different taxonomy complexities emerge. For instance, using less complex taxonomies consisting of only 10 keywords also requires less monitored interactions, since profiles and interaction contexts converge much faster than for more complex taxonomies.

Experiment Results. We study the network formation process of 200 unconnected actors for different environment setting. Depending on the complexity of the global taxonomy that determines interaction contexts, varying amounts of interactions are required in order to guarantee a feasible inference of social relations based on interest similarities. We let actors pick tags from a global taxonomy consisting of 10/20/50 keywords according to their interest profiles in order to annotate their interactions, e.g., express the expertise areas of support requests. In order to bootstrap a network formation process (see Figure 10(a)) links are predicted only (see dashed lines) based on actor profile overlaps [Skopik et al., 2010d]. Utilizing measured interaction metrics (here reciprocity cf. Eq. 2), social links are established based on evidence about reliable and dependable collaboration behavior. Note, the color of the nodes represent their (static) expertise areas, while their sizes reflect their degree of connectivity in the network. Figure 10(b) shows a network where



(a) varying network size $|N|$. (b) varying prop. path pp.

Fig. 11. Size of the circle of trust.



(a) varying network size $|N|$. (b) varying prop. path pp.

Fig. 12. Required graph operations.

most members have found at least one trustworthy (e.g., in terms of reciprocity) collaboration partner. Such social links are reflected by solid lines whereas their strengths reflect the level of cooperation. Still, most relations are predicted only (dashed lines). Finally, after a sufficient amount of interactions has been collected to reliably infer relations, a network consisting only of evidence-based relations is maintained in the saturation phase (Figure 10(c)).

We repeat this experiment to find out typically emerging network structures for varying taxonomy complexities (number of tags #tags) and different amounts of interactions (#ia). Table 1 reveals the details. The metrics are (i) number of connected components (nc), (ii) average number of network neighbors (nn), and (iii) network density (nd). Although an optimal connection is hard to determine, these graph metrics deem to be appropriate indicators [Romesburg, 2004] to describe and compare network structures. Note, the values in brackets int eh bootstrapping phase denote the given metrics if predicted links are treated as evidence-based links.

Table 1. Characteristic network metrics in different evolutionary phases of a formation process.

phase	#tags/#ia	network metrics
bootstr.	10/0	$nc = 200$, $nn = 0(7.62)$, $nd = 0$
	20/0	$nc = 200$, $nn = 0(5.56)$, $nd = 0$
	50/0	$nc = 200$, $nn = 0(1.13)$, $nd = 0$
formation	10/1000	$nc = 99$, $nn = 1.12$, $nd = 0.006$
	20/3000	$nc = 109$, $nn = 0.84$, $nd = 0.005$
	50/5000	$nc = 101$, $nn = 0.98$, $nd = 0.005$
saturation	10/5000	$nc = 4$, $nn = 3.15$, $nd = 0.017$
	20/15000	$nc = 7$, $nn = 2.65$, $nd = 0.014$
	50/25000	$nc = 5$, $nn = 2.89$, $nd = 0.016$

6.3. *Member Discovery Simulations*

We create synthetic networks with fixed amounts of nodes and power-law distributed edges [Reka and Barabási, 2002] to evaluate the effects of propagating profile information. This means, encrypted parts of a FOAF profile are shared over multiple hops even between unconnected members, if there is a strong trust path between them. This concept of propagation [Guha et al., 2004] enables users to extend their *circles of trust* (i.e., all members that can be reached over a strong trust path without exceeding an upper limit of hops) and to discover previously unknown members therein. The complexity of a graph is described by the average outdegree of a node in the long tail of the degree distribution; in other words, the average number of trusted neighbors (trustees) for the bottom 90% of members. We pick random nodes from this set and run experiments for each one until we get stable average results.

The first set of experiments investigates the average size of the *circle of trust*, depending on the number of trustees for different network sizes N and propagation path lengths *pp*. For that purpose profiles of all neighbors of specified nodes in the network are retrieved recursively until the whole circle is discovered. Figure 11 show that for highly cross-linked graphs (i.e., $avgtrustees > 2$), only short *pp*s (max. 3 or 4 hops) are feasible. Otherwise, virtually all members are in one's *circle of trust*. A second set of experiments highlights the computational complexity of determining the *circle of trust*. While the size of the network does not considerably influence the number of required graph operations from each actor's perspective (at least for small *pp*), increasing *pp* in highly cross-linked graphs leads to exponential costs (Figure 12). Graph operations include retrieving referenced nodes and edges, as well as neighbors, predecessors and successors in the network model. Each of these operations means that finally distributed FOAF profiles need to be queried and retrieved from the Web.

6.4. *Processing Encrypted FOAF Profiles*

We shortly discuss the complexity and required steps to enable the discovery of collaboration partners based on FOAF profile sharing using the security concepts discussed in this paper. For that purpose, we distinguish between three different operations: (i) publishing profiles, (ii) discovering neighbors, i.e., retrieve their (en-

crypted) profiles, (iii) transitive discovery, i.e., propagation of profile information over one hop. Table 2 summarizes complexities in terms of *number of retrieved documents* (i.e., public/private FOAF fragments, signatures, public/private key files) and *number of required steps* (i.e., file retrieval, encryption, decryption, file update, file upload). Note, we do not measure absolute performance of the proposed profile management approach, because this heavily depends on the hosting environment and IT infrastructure. Symbol n denotes the number of direct neighbors; p the number of distinct private FOAF fragments.

Table 2. Comparison of profile management operations.

operation	#retrieved docs	#steps
FOAF profile publishing	$3 + n$	$3 + 3p + n$
neighbor discovery	$(2 + p) \cdot n$	$(3 + p) \cdot n$
transitive discovery	$2 + 2p + n + pn$	$3 + 3p + n + pn$

FOAF Profile Publishing. Updating an actor's own profile consists of profile retrieval and update of already existing public/private profile fragments, signing the public fragment with own private key, retrieving the neighbors' public keys, encrypting private fragments individually for strongly connected (e.g., trusted) neighbors, publish public and private fragments on the Web.

Neighbor Discovery. This operation discovers directly connected actors by evaluating their profiles, e.g., interests, project participation, organizational memberships. Evaluating neighbor profiles includes for each single neighbor to retrieve the public profile and public key to validate the signature, retrieval of linked private fragments, decryption of data with own private key.

Transitive Discovery. Transitive profile sharing enables the discovery of unconnected community members. For that purpose intermediate nodes mediate information by retrieving (encrypted) profiles from neighbors, and re-encrypt them for their own (trusted) neighbors. In particular the following steps are performed: retrieve published public/private FOAF fragments of one neighbor, get public key to verify signature, decrypt private fragment with own private key, get public key of other neighbor(s), re-encrypt private fragment, attach this fragment to own FOAF profile, re-sign and re-encrypt own FOAF fragments; optionally, notify interested neighbors about third-party profiles.

6.5. *End-to-End Information Sharing Performance*

The overall process of trusted information sharing involves several backend services. Communicating with and retrieving data from these Web services is time-intensive, especially if they are frequently utilized and/or large amounts of data are transferred (and processed by respective SOAP stacks). Besides the actual *Information Repository*, we identified the *Information Catalog*, *Sharing View Management Service* and *Sharing Rule Management Service* as the most data-intensive services. Therefore, we studied the overall performance when caching different kinds of data.

In particular, the *Sharing Proxy* implements the caching strategy of self-pruning cache objects as widely adopted [Goodman, 2002].

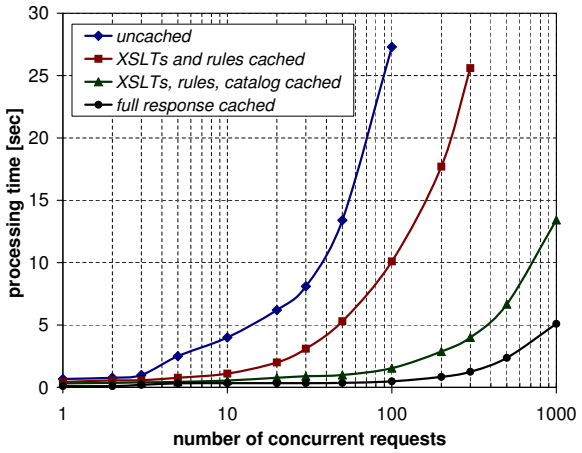


Fig. 13. Overall performance of the sharing framework.

Figure 13 depicts the required time of the *Sharing Proxy* to process different amounts of concurrent client requests. In detail, we measured the processing time (i) without any caching mechanisms, (ii) when caching only rarely changed sharing rules and associated sharing views (XSLTs), (iii) when caching rules, XSLTs, and catalog entries, (iv) for delivering the response only, i.e., providing the already transformed and cached information. The results show that with applying different caching strategies the overall performance can be significantly increased. However, depending on the domain's inherent trust dynamics, a trade-off between performance and up-to-dateness of cached data has to be carefully considered.

7. Background and Related Work

Cross-Organizational Collaborations. The concept of virtual communities is increasingly used to enable the collaboration between geographically distributed members belonging to various organizational units. Studies on distributed teams focus on human performance and interactions [Panteli and Davison, 2005] as well as *Enterprise 2.0* environments [Breslin et al., 2009]. Service-oriented architectures (SOA) have emerged as the defacto standard to design and implement open enterprise systems. They allow for loose coupling between single components and enable sophisticated discovery mechanisms based on functional (e.g., supported features) and non-functional (e.g., QoS) properties. Web service technology [Alonso et al., 2003] enables cross-organizational interactions in collaborative networks [Camarinha-Matos and Afsarmanesh, 2006].

Monitoring and Self-Organizing Systems. The problem of composition and adaptation is strongly related to organization and control. Self-* principles [Berns and Ghosh, 2009] provide the ability to manage systems autonomously and to dynamically adapt to changes in accordance with objectives and strategies. Self-organizing environments can adapt based on the context [Di Nitto et al., 2008]. Inspired by the principles of control systems, the autonomic computing paradigm aims at achieving dynamic adaption of the system based on the actual execution context [Leymann, 2006]. Enhanced flexibility of complex systems is introduced by establishing a cycle that feeds back environmental conditions to allow the system to adapt its behavior. This MAPE cycle [IBM, 2005] is considered as one of the core mechanisms to achieve adaptability through self-* properties. While autonomic computing allows for autonomous elements and applies these principles to distributed systems, current research efforts left the human element outside the loop. Based on the observed context of the environment, different adaptation strategies can be applied to guide interactions between actors, the parameters of those strategies, and actions to prevent inefficient use of resources and disruptions. In the context of multi agent systems (MAS), self-configuring social techniques were introduced in [Bryl and Giorgini, 2006].

Social Trust in service-oriented systems has become a very important research area. SOA-based infrastructures are typically distributed comprising a large number of available services and huge amounts of interaction logs. Therefore, trust in SOA has to be managed in an automatic manner [Malik and Bouguettaya, 2009]. Depending on the environment, trust may rely on the outcome of previous interactions [Mui et al., 2002] and interest similarity [Golbeck, 2009; Matsuo and Yamamoto, 2009]. In our approach, metrics express social behavior influenced by the context in which collaborations take place [Skopik et al., 2010a]. For instance, *reciprocity* [Falk and Fischbacher, 2006] is a concept describing that humans tend to establish a balance between provided support and obtained benefit from collaboration partners.

Social Platforms and Service Communities. Social networks have received tremendous attention recently from both research and academia. A large amount of information is exchanged online using social networking platforms. It becomes thus essential to adapt and influence the information exchange in an automated manner [Skopik et al., 2010b]. Selective dissemination of information (SDI) [Altinel and Franklin, 2000; Diao et al., 2004] is used filter unnecessary data by considering user profiles.

Social networks become more and more interlinked with enterprises and collaborative platforms [Breslin et al., 2009]. Semantically-enriched service platforms following the SOA paradigm such as WSMX [Haller et al., 2005] provide the means to discover and compose services in cross-organizational environments based on standardized languages (see WSMO [Lara et al., 2004]). These platforms not only enable interactions between technical services across boundaries, but also human

interactions on top of these services. The convergence of social interactions in flexible service-oriented environments makes it essential to extend well-established data formats for describing the structure of social networks such as FOAF with access control techniques.

The mechanisms for signing RDF graphs have been presented in [Giereth, 2005]. The combination of FOAF and SSL [Story et al., 2010] enables secure access to FOAF profiles. The embedding of access control mechanisms in FOAF has been illustrated in [Hollenbach et al., 2005; Kruk et al., 2006].

Controlling Information Disclosure. The interplay of trust and privacy has been studied in the areas of social networks [Dwyer et al., 2007] and electronic commerce [Metzger, 2004]. Especially the latter work concludes that trust is strongly related to information disclosure, and thus, privacy. As users increasingly disseminate their information on the Web, privacy concerns demand flexible information access control mechanisms [Mori et al., 2005]. In some recent articles on the Web, e.g., see [Dybwad, 2009; Kilner, 2009], the authors discuss to what extent shared personal information of (naive) users may be exploited. Marsh discusses in an article [Marsh, 2008] how trust models enhance information sharing among community members. Knowledge sharing behavior in virtual communities and the relation to trust has been investigated in [Hsu et al., 2007].

There exist several works in the area of recommender systems (e.g., [Walter et al., 2009]) that use personal trust to optimize item recommendation (that is usually based on collaborative filtering only). One of the more related use cases is the recommendation of documents [Hess et al., 2006]. However, while they use trust to improve document recommendations (e.g., to better match interests of users), we restrict access based on context-aware personal relations. Others focus on traditional trust-based access control mechanisms [Bhatti et al., 2005] that are based on more static role models and relations.

The technical realization of *trusted information sharing* in the introduced science collaboration network is related to *selective dissemination of information* (SDI) [Altinel and Franklin, 2000; Diao et al., 2004]. SDI deals with questions regarding which (parts of) data are shared with others, and mechanisms to disseminate data. We adopted concepts of SDI, such as the representation of information through XML, or mechanisms to process XML-based data.

8. Conclusion and Future Work

In this paper, we discussed the application of social network concepts in cross-enterprise collaboration scenarios. While creating dynamic profiles and flexibly discovering people and services is frequently used in typical recommender systems and on social platforms, the application in enterprise scenarios in form of overlay networks is a novelty. Especially, the combination with Semantic Web methodologies, such as semantic Web services, taxonomy-based context management and SOA to achieve data and service interoperability is a new aspect. We proposed an approach

to support human collaboration in different domains and organizations in a seamless manner; not only from a social perspective, but also from a technical one. In particular we studied and implemented concepts for trusted information sharing which is a key objective in modern collaboration systems.

Our future research includes the application of social overlay networks in real cross-enterprise scenarios. This will be done within the EU FP7 project COIN, where we will collect valuable information regarding the efficiency of the discovery process based on social network structures. Furthermore, we will study network dynamics such as member fluctuation and frequency of re-discovering partners; as well as the feasibility of our approach from a technical point of view, e.g., limits in managing FOAF profiles depending on profile change rates. There is also great potential for improvement and extensions respectively of the Semantic Web Services stack, especially in context of our socially-enhanced application domain.

Acknowledgments

This article is an extended version of [Skopik et al., 2011]. The authors like to thank all people who contributed to the original paper. Furthermore, the authors like to express their gratitude to Iwona Les for implementing various aspects of the Trusted Information Sharing Framework.

References

Adams, C. and Lloyd, S. (1999). *Understanding Public-Key Infrastructure: Concepts, Standards, and Deployment Considerations*. Macmillan Publishing.

Alonso, G., Casati, F., Kuno, H., and Machiraju, V. (2003). *Web Services - Concepts, Architectures and Applications*. Springer.

Altinel, M. and Franklin, M. J. (2000). Efficient filtering of xml documents for selective dissemination of information. In *Very Large Data Bases Conference*, pages 53–64.

Amazon.com (last access: Nov. 2011). Amazon mechanical turk. available online: http://www.mturk.com.

Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The semantic web. *Scientific American*.

Berns, A. and Ghosh, S. (2009). Dissecting self-* properties. In *International Conference on Self-Adaptive and Self-Organizing Systems*, pages 10–19. IEEE.

Bhatti, R., Bertino, E., and Ghafoor, A. (2005). A trust-based context-aware access control model for web-services. *Distributed and Parallel Databases*, 18(1):83–105.

Breslin, J., Passant, A., and Decker, S. (2009). Social web applications in enterprise. *The Social Semantic Web*, 48:251–267.

Bryl, V. and Giorgini, P. (2006). Self-configuring socio-technical systems: Redesign at runtime. *International Transactions on Systems Science and Applications*, 2(1):31–40.

Camarinha-Matos, L. M. and Afsarmanesh, H. (2006). Collaborative networks. In *PRO-LAMAT*, pages 26–40.

Castano, S., Ferrara, A., and Montanelli, S. (2006). Matching ontologies in open networked systems: Techniques and applications. *J. Data Semantics V*, pages 25–63.

Di Nitto, E., Ghezzi, C., Metzger, A., Papazoglou, M., and Pohl, K. (2008). A journey to highly dynamic, self-adaptive service-based applications. *Automated Software Engineering*.

Diao, Y., Rizvi, S., and Franklin, M. J. (2004). Towards an internet-scale xml dissemination service. In *Very Large Data Bases Conference*, pages 612–623.

Dwyer, C., Hiltz, S. R., and Passerini, K. (2007). Trust and privacy concern within social networking sites: A comparison of facebook and myspace. In *Americas Conference on Information Systems*.

Dybwad, B. (2009). Think twice: That facebook update could get you robbed. http://mashable.com/2009/08/27/facebook-burglary/.

Euzenat, J. and Shvaiko, P. (2007). *Ontology Matching*. Springer, Berlin.

Falk, A. and Fischbacher, U. (2006). A theory of reciprocity. *Games and Economic Behavior*, 54(2):293–315.

Giereth, M. (2005). On partial encryption of rdf-graphs. In *International Semantic Web Conference*, pages 308–322.

Golbeck, J. (2009). Trust and nuanced profile similarity in online social networks. *ACM Trans. on the Web*, 3(4).

Gold, N., Knight, C., Mohan, A., and Munro, M. (2004). Understanding service-oriented software. *IEEE Software*, 21(2):71–77.

Goodman, B. D. (2002). Accelerate your web services with caching. *IBM Advanced Internet Technology*.

Guha, R., Kumar, R., Raghavan, P., and Tomkins, A. (2004). Propagation of trust and distrust. In *International World Wide Web Conference*, pages 403–412.

Haller, A., Cimpian, E., Mocan, A., Oren, E., and Bussler, C. (2005). Wsmx - a semantic service-oriented architecture. In *International Conference on Web Services*, pages 321–328. IEEE.

Hess, C., Stein, K., and Schlieder, C. (2006). Trust-enhanced visibility for personalized document recommendations. In *ACM Symposium on Applied computing*, pages 1865–1869.

Hollenbach, J., Presbrey, J., and Berners-Lee, T. (2005). Using rdf metadata to enable access control on the social semantic web. In *International Symposium on Wearable Computers*.

Hsu, M.-H., Ju, T., Yen, C.-H., and Chang, C.-M. (2007). Knowledge sharing behavior in virtual communities: The relationship between trust, self-efficacy, and outcome expectations. *International Journal of Human-Computer Studies*, 65(2):153–169.

IBM (2005). An architectural blueprint for autonomic computing. *Whitepaper*.

Juszczyk, L. and Dustdar, S. (2010). Script-based generation of dynamic testbeds for soa. In *International Conference on Web Services*, pages 195–202. IEEE.

Kilner, R. (2009). Internet shopping for burglars on social networks (online, last access: Nov. 2011. http://www.insurancedaily.co.uk/2009/08/28/internet-shopping-for-burglars-on-social-networks/.

Kruk, S. R., Grzonkowski, S., Gzella, A., Woroniecki, T., and Choi, H.-C. (2006). D-foaf: Distributed identity management with access rights delegation. In *Asian Semantic Web Conference*, pages 140–154.

Lara, R., Roman, D., Polleres, A., and Fensel, D. (2004). A conceptual comparison of wsmo and owl-s. In *European Conference on Web Services*, pages 254–269. IEEE.

Leymann, F. (2006). Workflow-based coordination and cooperation in a service world. In *CoopIS, DOA, GADA, and ODBASE*, pages 2–16.

Malik, Z. and Bouguettaya, A. (2009). Reputation bootstrapping for trust establishment among web services. *Internet Computing*, 13(1):40–47.

Marsh, S. (2008). Information sharing is enhanced using trust models. *PerAda Magazine (Pervasive Adaptation)*.

Matsuo, Y. and Yamamoto, H. (2009). Community gravity: Measuring bidirectional ef-

fects by trust and rating on online social networks. In *International World Wide Web Conference*, pages 751–760.

Metzger, M. J. (2004). Privacy, trust, and disclosure: Exploring barriers to electronic commerce. *Journal on Computer-Mediated Communication*, 9(4).

Mori, J., Sugiyama, T., and Matsuo, Y. (2005). Real-world oriented information sharing using social networks. In *International Conference on Supporting Group Work*, pages 81–84.

Mui, L., Mohtashemi, M., and Halberstadt, A. (2002). A computational model of trust and reputation for e-businesses. In *Hawaii International Conference on System Sciences*, pages 188–.

Panteli, N. and Davison, R. (2005). The role of subgroups in the communication patterns of global virtual teams. *IEEE Trans. Prof. Com.*, 48(2):191–200.

Reka, A. and Barabási (2002). Statistical mechanics of complex networks. *Rev. Mod. Phys.*, 74:47–97.

Romesburg, H. C. (2004). *Cluster Analysis for Researchers*. Krieger Pub. Co.

Schall, D., Dorn, C., Dustdar, S., and Dadduzio, I. (2008a). Viecar - enabling self-adaptive collaboration services. In *Euromicro Conference on Software Engineering and Advanced Applications*, pages 285–292.

Schall, D. and Dustdar, S. (2010). Dynamic context-sensitive pagerank for expertise mining. In *Social Informatics*, pages 160–175. Springer.

Schall, D. and Skopik, F. (2010). Mining and composition of emergent collectives in mixed service-oriented systems. In *Conference on Commerce and Enterprise Computing*, pages 212–219. IEEE.

Schall, D., Truong, H.-L., and Dustdar, S. (2008b). Unifying human and software services in web-scale collaborations. *IEEE Internet Computing*, 12(3):62–68.

Skopik, F., Schall, D., and Dustdar, S. (2010a). Modeling and mining of dynamic trust in complex service-oriented systems. *Inf. Syst.*, 35:735–757.

Skopik, F., Schall, D., and Dustdar, S. (2010b). Trust-based adaptation in complex service-oriented systems. In *International Conference on Engineering of Complex Computer Systems*, pages 31–40. IEEE.

Skopik, F., Schall, D., and Dustdar, S. (2010c). Trustworthy interaction balancing in mixed service-oriented systems. In *ACM Symposium on Applied Computing*, pages 801–808.

Skopik, F., Schall, D., and Dustdar, S. (2011). Managing social overlay networks in semantic open enterprise systems. In *International Conference on Web Intelligence, Mining and Semantics*, pages 50–. ACM.

Skopik, F., Schall, D., Psaier, H., and Dustdar, S. (2010d). Social formation and interactions in evolving service-oriented communities. In *European Conference on Web Services*, pages 27–34. IEEE.

Stollberg, M. and Norton, B. (2007). A refined goal model for semantic web services. In *International Conference on Internet and Web Applications and Services*, pages 17–22.

Story, H., Harbulot, B., Jacobi, I., and Jones, M. (last access: 2010). Foaf+ssl: Restful authentication for the social web. http://esw.w3.org/Foaf%2Bssl.

Walter, F. E., Battiston, S., and Schweitzer, F. (2009). Personalised and dynamic trust in social networks. In *ACM Conference on Recommender Systems*, pages 197–204.

Zaremba, M. and Vitvar, T. (2008). Wsmx: A solution for b2b mediation and discovery scenarios. In *European Semantic Web Conference*, pages 884–889.

Ziegler, C.-N. and Golbeck, J. (2007). Investigating interactions of trust and interest similarity. *Decision Support Systems*, 43(2):460–475.

Ziegler, C.-N. and Lausen, G. (2005). Propagation models for trust and distrust in social networks. *Inf. Syst. Frontiers*, 7(4-5):337–358.