Aspects of Trusted and Secure Business-Oriented VO Management in Service Oriented Architectures

Adomas Svirskas[1*], Michael D. Wilson[1], Alvaro E. Arenas[1], Emil C. Lupu[2], Nilufer Tuptuk[2], David Chadwick[3], Pablo Giambiagi[4], Theo Dimitrakos[5], Bob Roberts[6]

[1]Central Laboratory of the Research Councils, Rutherford Appleton Laboratory, UK
{A.Svirskas, M.D.Wilson, A.E.arenas}@rl.ac.uk
[2]Imperial College London, UK
{E.C.Lupu, nt102}@doc.ic.ac.uk
[3]University of Kent, UK
D.W.Chadwick@kent.ac.uk
[4]Swedish Institute of Computer Science, Sweden
Pablo@sics.se
[5]British Telecom, UK
Theo.Dimitrakos@bt.com
[6]Kingston University London, UK
R.Roberts@kingston.ac.uk

Abstract

Virtual Enterprises or Organisations (VO) have been the focus of research for over a decade[1]. Although proprietary implementations of VO management tools exist, secure tools based on interoperating open standards are not yet available. The open standards on which to build them are just being released as reliable implementations. The requirements of VOs for trust and security are presented, which lead to an architecture for a secure VO management framework. The design of such a framework is analysed to show how the current open Web Service specifications could be used to implement it in practice. The need for the reliable and interoperable implementation of these essential Web Services specifications is advocated.

1. Introduction

For the purposes of this paper a VO is understood as a temporary or permanent coalition of geographically dispersed individuals, groups, organizational units or entire organizations that pool resources, capabilities and information to achieve common objectives. VO's can provide services and thus participate as a single entity in the formation of further VO's. This enables the creation of recursive structures with multiple layers of "virtual" value-added service providers [1].

Many researchers have operated for many years in both networks and project consortia or collaborations that are forms of VOs. Those who have participated in such research consortia will have experienced many of both the strengths and weaknesses of current VO's [10]. Such consortia can be established quickly and operated by stable member organizations easily as long as nothing goes wrong, and when the research products are public domain, so there is no required division of benefits. Once such VO's encounter problems with consequent liabilities, or result in benefits where financial reward needs to be allocated between partners then the VO can be bogged down in legal arguments between the members. To overcome these problems, legal mechanisms are employed such as the establishment of partnerships between organisations, or mutual joint ownership (common in the telecoms sector). However, these legal procedures consume time

[1] The term “Virtual Enterprise” was coined for a US Congress Report in 1989 [1].

and effort that is only justified by long term relationships, and are not appropriate for short term or dynamic virtual organizations required to respond rapidly to opportunities arising from changing market conditions.

Making the processes of creation, operation and dissolution rapidly responsive requires both the appropriate legal mechanisms, and dynamic management of the VO. Although several specifications and implementations of composable Web Services tools exist (e.g. [11]), secure tools for VO management (VOM) based on interoperating open standards are not yet available, let alone those that address the legal requirements too.

To open up the market for dynamic VO membership to the wider community who can present there businesses as Web services, requires the development of VOM tools that apply non-proprietary interoperable technology standards that can address VOM issues, including support for exchange of legally valid documents and legal procedures. This paper considers the technical requirements of such an IT infrastructure, as well as the legal requirements to allow financially profitable VO's to flourish, and the design options for VOM tools that might meet them.

2. VO Management

Within the academic research community, where the legal concerns and consequent trust and security requirements don't apply as strongly as in commercial sector, VOMS [12] has been developed as a VO membership service for European DataGrid based consortia to provide information on each user's relationship to the VO in terms of groups, roles and capabilities. However the requirements on VOMS only address membership, with limited security requirements including an expiration time for the single sign on to multiple VO's. It supports interoperability only based on the Globus Grid toolkit in which it is implemented. There are no provisions for legal contracts to limit liability or retain financial reward, and none to state and store agreements on quality of each service provided.

Although there are no identified VO management systems that address the range of requirements of security, interoperability and legal issues, there are many Web Services composition approaches ranging from robust orchestrators, less rigid choreographers, to those applying the flexibility of the Semantic Web [13] to address service discovery. However, the closest any get to addressing the legal requirements of the VO management process is in stating Service Level Agreements (SLA) for specific services to specify accounting and billing procedures and quality of service. The initial legal collaboration agreement to establish membership of a VO and address potential liabilities and reward distribution still remain paper documents outside the automated process based on templates such as those produced by the Alive project [14]. Such VO agreements can act as frameworks in which specific terms and conditions of automated SLAs operate, but they remain a non-automated long term necessity that limit the flexibility of creating dynamic VO's.

One issue that falls outside the terms and conditions of an SLA, and therefore solely within the scope of the overall collaboration agreement, is that of trust, because when one can easily resort to the legal enforcement of binding terms and conditions, trust is less important. The terms and conditions of an SLA can be managed in terms of security and quality of service policies applied to each service. Trust becomes more crucial when the terms and conditions that users actually "live by" do not apply, or are broken because of unforeseen circumstances. In those cases different parties rely on the good will, and commonality of goals of those involved to resolve the issues that arise in unforeseen circumstances. The terms and conditions that are actually defined in contracts and SLA's may not be the same as those "lived by" in which case trust is also required to cover the gap between the two - and when it fails, lawyers must be employed. In order to manage the VO process, it is therefore necessary to understand the context of the terms and conditions, that is to say the broader business processes which are operating to require the existence of the VO, and that operate within it that define the foreseen, and therefore by exclusion the unforeseen, circumstances.

2.1 Relation between VOM and BPM

Virtual Organisations are closely related to business collaborations between the services, organisations and individuals involved and are intended to facilitate, directly or indirectly, business solutions in most cases. VO management process can be perceived just as a type of business collaboration or (process) that uses the same mechanisms as for "operational" business collaborations (or processes). The collaboration agreement of a VO specifies processes related to the administration of the VO itself, such as changes to the VO membership or the collaboration agreement.

While this opinion bases itself on the fact that VO management processes are fairly simple compared to "operational" business collaborations between business partners and this is certainly true, some clarifications need to be made. Firstly, we will explain the difference between the concepts of *collaborative* and *process* or, in other terms, between the *choreography* and *orchestration*.

Collaboration occurs between peers and takes form of message exchange between them, according to a defined set of rules. These are global "reactive" rules that declaratively prescribe normal/abnormal progress, common agreement of the outcomes and are used by each participant to determine which message exchange should happen next at any point of collaboration. A set of such rules collectively is referred to as choreography, which also provides a definition of the information formats being exchanged by all participants. Through the use of a global model, choreography ensures that contractual behaviour across multiple services can be achieved without complex wiring or complex wiring tools [8]. Recursive composition model that lets you build choreographies incrementally by combining existing choreographies, which is necessary to address complex inter-organization business processes [20].

Orchestration, on the other hand, specifies the behaviour of a participant in a choreography by defining a set of "active" rules that are executed to infer what to do next, once the rule is computed, the orchestration runtime executes the corresponding activity(ies). Orchestration assumes existence of an entity, which is the central point of control and governs overall workflow of activities. Choreography, on the contrary, is meant to be enacted by peers without an intermediary, at runtime, the choreography definition can be used to verify that everything is proceeding according to plan. Choreography can also be used to generate a public interface (e.g. abstract BPEL) that can be used to tie in internal activities to support the choreography [15].

Because choreography is less tightly coupled and needs to be transparent for inter organisational interoperation therefore it needs to be more robustly defined and standardised than orchestration [7]. W3C is currently standardising choreography [5], ensuring that there is liaison between the W3C Web Services Choreography working group and the OASIS WS BPEL Technical Committee to maintain alignment of the choreography specification with the needs of business process model execution.

Having clearly identified the differences between choreography and orchestration, we need to associate the VOM concepts with the more appropriate of the two former techniques. In some case VOM might take form of a centrally-controlled process with the VO Manager taking control according to the VO agreement, so it might look similar to orchestration. This is quite typical scenario for VOs in industry verticals, e.g. automotive, where a large vendor controls a supply chain VO, for example. However, a closer look into a more generic VO life cycle model and the fact that the collaboration agreement specifies events related to the administration of the VO (and, inherently messages to be exchanged) reveal that VOM is essentially a set of peer-to-peer interactions. The pairs of peers in this case mostly contain a VO Member on one side and a VO "enabling service" (management, monitoring, security, etc.) on the other, however peer-to-peer interaction occur between the enabling services themselves (e.g. VO Manager and Trust/Reputation service). In some cases, for example negotiation and need for consensus with regard to significant changes in VO operation, the interaction may involve multiple parties.

The thoughts expressed above suggest that master/slave relationship between the VO Manager and the rest of the VO members is not suitable for VOM. Moreover, a single VO might consist of members belonging to different organisations, which do not share application and workflow implementation technologies and will not allow external control of their back-end applications [7]. These considerations and the relatively limited variety of VOM enabling collaborations (membership management, monitoring, trust provisioning) lead us to conclusion that VOM interactions should be choreographed, i.e. required collaborations between the parties can be defined declaratively and enacted at runtime. These choreographies together make VOM protocol, which exposes the common knowledge which the members of a VO need to share, while leaving the implementation of the protocol to the individual VO members. This approach adheres to public managed processes integration pattern [24].

2.1 The proposed Model

Service Oriented Computing frameworks allow for the creation, maintenance, and application of the service ensembles that VOs maintain. Key business functions are treated as services - that is globally identifiable and discoverable network-enabled entities that provide some capability through the exchange of messages over standardized extensible protocols that allow data-encapsulated cross-application invocations.

The TrustCoM IST project [17] is developing a framework for enabling secure collaborative business processing in on-demand created, self-managed, scaleable and highly dynamic VOs. The project is built on the convergence of emerging technologies such as Web Services and Grid. As part of this project we have done an analysis of Web Services Security Specifications as an enabling technology to build secure and trustworthy dynamic VOs

Within TrustCoM [17] we are using the term Enterprise Network (EN) to refer to a community of companies that have agreed in principle to cooperate in this way. The community - typically operating in a particular industry or application domain is bound by agreements, conventions and procedures that facilitate this cooperation. An EN is formed using "virtualised" view of its participating members, therefore all the internal details private to members are hidden behind Access Points, which map virtual resources to the actual ones. A Dynamic VO is cooperation within a subset of EN members. Specific objectives and market needs trigger the establishment and operation of the dynamic organisation. An EN provides the infrastructure to rapidly set up new VOs:

- The EN may be static but the VOs can be dynamic
- Participation in an EN shows disposition to create VOs and offers infrastructure support for creating VOs but EN is not a VO

Below in Figure 1 the interrelations between the outlined concepts are depicted. As we can see, an enterprise may participate in more that one VO at any given point of time, delegating appropriate resources (via virtualised services) and playing different roles, according to its policies and those of the VOs the enterprise is involved in.

To operate an EN and the virtual organisations within it appropriate services are needed to support the model. Firstly, infrastructure and services are needed to provide a framework for secure cooperation and to replace the trust inherent in operation within an integrated real organisation and between a real organisation and its customers. These infrastructure services are basically independent of the particular application domain. In essence, the VO management process is ensuring that the members of a VO play by the rules agreed by everyone involved and members' behaviour is observable. In order to define the necessary services for the VO management we need to identify some other key concepts. As indicated earlier, we can perceive a VO as composition and interaction of three main components:

- The collaboration agreement also called General VO Agreement (GVOA)
- The business collaboration
- The participants

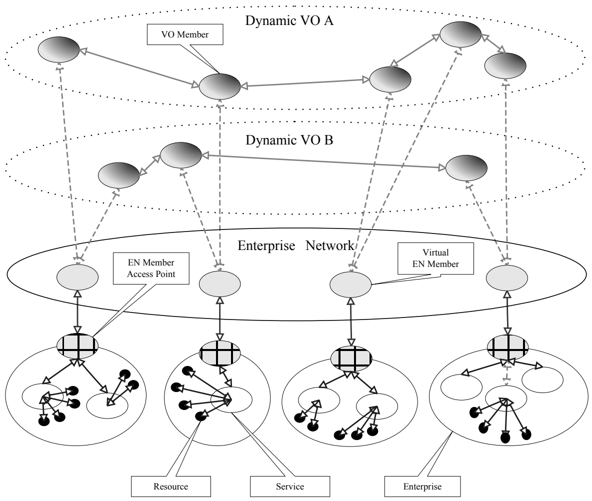

Figure 1. Relationship between the enterprises, enterprise networks and VOs (adapted from [22])

Participants can vary in size from individuals to entire organizations (real or virtual). The fundamental unit of business collaboration is the task; it is an operation, which is atomic from the VO's point of view, meaning that it can be performed entirely within one participant. Within a participant, it may be performed by decomposition into smaller subtasks (which comprise an orchestrated, workflow-like process), but that is hidden within the VO. A task may also need to receive input from or send output to other tasks (a part of choreography to define these interactions). Tasks are assigned to roles to be carried out. Any type of management is based on some agreements, therefore we will look into the collaboration agreement more closely, as depicted in Figure 2. A collaboration agreement, in turn, has three main components:

- The roles
- The policies
- The Service Level Agreements (SLA)

A collaboration agreement may be derived from a prior template that is then parameterized or customized appropriately by negotiation among the participants when the VO is created [26]. An agreement is a legal document that specifies certain legal aspects of a collaboration agreement, such as procedures for notarization, registration, etc. A collaboration agreement contains other aspects that are not necessarily legal documents, however.

A role is a logical description of member's position within the VO and choreographed collaborations, a participant can assume multiple roles. Policies constrain roles and interactions between roles, define the tasks pertaining to these roles. An SLA is an aspect of the collaboration agreement, which specifies what the participants are obliged to provide in terms of the measurable characteristics of their services. SLAs are enforced with the help of monitors that observe information about services provided through their public interfaces.

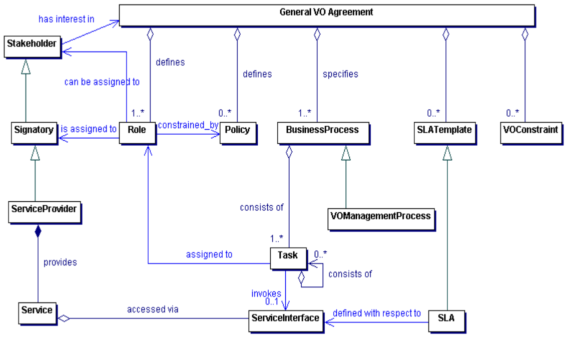

Figure 2. Static model of general VO agreements

Many, though not necessarily all, participants will provide services, either just to fellow participants or directly to the outside world as well. Services can only be accessed through their interfaces.

The following sections describe the infrastructure services and realization of the VO management.

3. Trust and security requirements of VO management

The development of a VO framework will become a reality only if the security concerns are adequately addressed and managed throughout the life cycle of the VO. VOs require a security system which can provide functionality necessary to minimise possible malicious activities and to enable authorised usage of the services provided by the VO partners. In order to specify the trust and security requirements for VOM, we need to take into account the processes, steps and organizational policies necessary for virtual organizations to be created across multiple administrative domains.

As an example we consider the case of a System Integrator, willing to form collaboration with other partners and suppliers on the development of a new product in the skill and labour-intensive industry sector. The product is delivered as a collaborative effort between service providers (e.g. suppliers) and the services orchestrator/integrator. Two key prerequisites for this collaborative activity to take place are:

- The ability for the integrator to enact various and, often radically different, contracts with its partners
- The ability to monitor and enforce such agreements throughout the engagement. By accepting the contractual obligations, each party will commit that all resources usage and services rendered are done in accordance to the terms agreed upon in the contract.

It is important to note, however, that the integrator, acting also as the orchestrator for the services rendered by its directly contracted or first level suppliers, may require having visibility of all contractual obligations that they in turn establish with their in turn subcontracted second level suppliers. As such, a recursive structure of service providers and aggregators may therefore exist for the purpose of establishing the VO, where each partner must integrate its business processes with other partners involved in order to deliver the final product. In this scenario, a service may be composed of a number of sub-services each executed by one or more party in the collaboration. The process, which happens in the initial phases of VO establishment, is depicted in Figure 3.

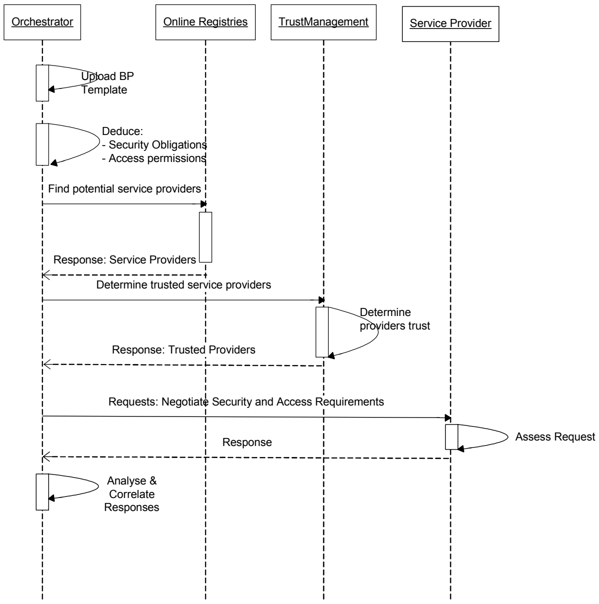
Orchestrator
Online Registries
TrustManagement
Service Provider
Upload BP Template
Deduce:
- Security Obligations
- Access permissions
Find potential service providers
Response: Service Providers
Determine trusted service providers
Determine providers trust
Response: Trusted Providers
Requests: Negotiate Security and Access Requirements
Assess Request
Response
Analyse & Correlate Responses

Figure 3. Sequence diagram for the identification and negotiation phase

From this example and from the earlier considerations we can identify the following (not exhaustive) trust and security related requirements:

- Ability to store different types of information about parties and to compute their reputations/trustworthiness based on this information and on trust/reputation algorithms. The actual algorithms and types of information that are used by the trust and reputation management system should be configurable, so that the system can be tailored to the domain of use.
- Ability to adapt policies to changes in the contract and the business processes.
- Ability to adapt policies in response to certain events in the overall execution environment: failures, external attacks, unauthorized access attempts, changes in other security policies, etc.
- Ability to adapt policies in response to variations of trust vested into both external clients and internal partners, as well as changes in the risk associated with the activities
- Dealing with intentional misrepresentations of trust in a party to (explicitly) damage reputation
- Secure storing of contract elements (including contract templates)

Of course, the basic security requirements such as identity, authentication, authorisation, message integrity, confidentiality, non-repudiation of origin, message traceability, preservation of privacy etc. must be satisfied.

4. The proposed VOM implementation

With the requirements stated above in mind, the next sections will outline the key services of VOM framework. We will focus on the VO Membership, Monitoring and Trust Management services, as they arguably have the biggest impact in VOM processes. We will deliberately leave the explanation of some other important services out of the scope of this paper due to the length limitations.

Figure 4 depicts the logical view of VOM technical architecture from the management perspective i.e. identifying the specific services, which comprise VO Management Service in general and, on the other hand, showing the interactive components exposed to the human users. Figure 4 also shows different types of interactions between the management services and management interface: request-response, event-based, reliable messaging, and shared use of registry/repository. At this level of detail we do not specify concrete protocols supporting the interactions and propose to encapsulate low-level technical intricacies into generic interfaces by defining abstraction layers for reliable messaging, event-based, communications etc. It is quite important to have certain degree of flexibility with regard to various technical implementation details because of frequent changes in underlying technologies, competing specifications, lack of standards etc. For example, there are two competing reliable messaging specifications in Web Services domain – WS-Relialibility and WS-ReliableMessaging [27].

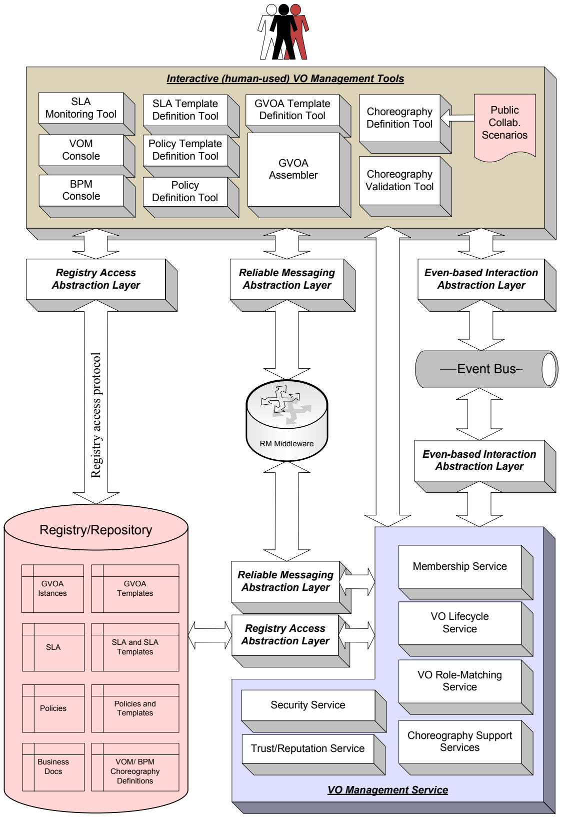
Interactive (human-used) VO Management Tools
SLA Monitoring Tool
VOM Console
BPM Console
SLA Template Definition Tool
Policy Template Definition Tool
Policy Definition Tool
GVOA Template Definition Tool
GVOA Assembler
Choreography Definition Tool
Choreography Validation Tool
Public Collab. Scenarios
Registry Access Abstraction Layer
Reliable Messaging Abstraction Layer
Even-based Interaction Abstraction Layer
Registry access protocol
Event Bus
RM Middleware
Even-based Interaction Abstraction Layer
Registry/Repository
GVOA Istances
GVOA Templates
SLA
SLA and SLA Templates
Policies
Policies and Templates
Business Docs
VOM/ BPM Choreography Definitions
Reliable Messaging Abstraction Layer
Registry Access Abstraction Layer
Membership Service
VO Lifecycle Service
VO Role-Matching Service
Security Service
Trust/Reputation Service
Choreography Support Services
VO Management Service

Figure 4. VO management technology architecture logical view - management perspective

The following sections will describe Membership, Monitoring and Trust Management services in greater level of detail.

Membership management service

Based on the business process that needs to be enacted, a VO community membership plan needs is defined in advance, detailing the order within which a service joins the community and performs necessary actions. VO community membership evolves following the progress of the business process enactment: services that are needed for performing some task enter the VO community, those that have fulfilled their role leave and those which committed terminal violations are expelled and replaced, should the business process enactment continue. Naturally, during the VO operation new members may join the VO or existing members may leave the coalition. Membership privileges of those that have completed their tasks and those that have been expelled out need to be revoked. Each time a new member joins or leaves; the VO membership list needs to be updated. For example in case of a new member joining the VO, the VO Membership Management Service should:

- Be able to authenticate the new VO member
- Update the VO membership list
- Map the new member to a role (defining its obligations, permissions and prohibitions).

The member should configure its local security policies according to its obligations. This could include policies about data disclosure.

Furthermore, adaptation policies enable automatic updates to membership (e.g. expulsion due violation of SLA or security obligations), to role (e.g. an existing member also assuming a role that was previously assigned to a member that is now being expelled), or to policy (e.g. strengthening the performance thresholds so as to compensate for previous underperformance, or obligation to encrypt communication if intrusion is suspected, etc.).

4.2 Monitoring Service

One of the important aspects within a VO is the contractual "binding" between participants of the VO, therefore monitoring services are needed in order to observe participants' behaviour with respect performance. Such monitoring services are subsequently utilized by administrative business tasks to ensure proper business process enactment.

The term "monitoring" as it is commonly used in the context of SLA management, generally involves a range of functionalities that need to be distinguished. We will take a look mainly at contract-related monitoring, as other performance related issues, like execution of specific tasks, may be defined in a SLA.

Monitoring in general presumes that the parameters required by the SLO metrics are in some way accessible through existing instrumentation and their values can be monitored. Hence, in a concrete scenario one might have to consider a data provision instance that is closely coupled with the system environment that is affected by the SLO.

Three levels may be distinguished for monitoring and data provision:

- The Service level covers processes that run on the machine that hosts the service(s) offered to the VO. This ensures fast reaction to system- and service-specific events and is of most interest for taking preventive actions in order to avoid violations.
- The Enterprise Domain level. Sensitive data should not cross this point unless otherwise specified. A service provider may want to "neutralize" monitored data in such a way that no security policies of the company are compromised.
- The VO level. Most of the data aggregation and comparison will take place at this level. Monitoring services on VO level are “regular” services themselves.

The actual monitoring acquires the data from sources at any of these levels either (a) by requesting the data directly (pull-mode), i.e. by executing the appropriate operation of the data provision service or (b) by expecting the source to provision all required parameters on a regular or event-driven basis (push-mode). In most cases pulling data requires the least implementation effort for the service-provider, who needs to supply the expected operations, yet need not, like in push-mode, bother about reacting to possible changes in SLA management, which could e.g. require different scheduling of data provision or changing the event cause. Since pulling data will lead to a higher overhead in message exchange, the choice of mode will depend on implementation and efficiency considerations.

4.3 Trust and reputation management service

Members in a VO can monitor their interactions with respect to other entities within the VO to obtain first hand knowledge about their behaviour. As discussed earlier, outcomes of the interactions (experience) could be used to re-evaluate the trust level that one (subject) has in another entity (target) for a particular context. The combination can be carried out by the VO Trust Management service (a part of the VO Management Service) to combine individuals' views of trust relationships to compute a combined 'reputation' for each member in the VO. As illustrated in Figure 5, each member in the VO can provide the VO Trust Management Service with trust information (recommendations or opinions) about the members it is interacting with. The VO Trust Management Service could use this information to re-assess the trustworthiness of the entities in the VO. This information could be used to make business decisions based on the earlier behaviour of the participants.

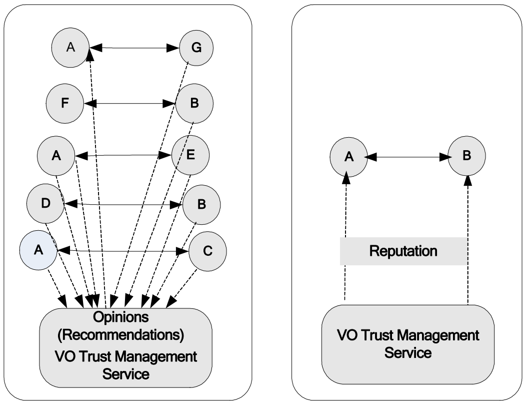

Figure 5. VO members' interactions with trust management service

During the operation phase, the VO Manager may need to bring in new members to meet the new demands of the VO or replace some of the old members. The Trust Management Service in this case can suggest a potential candidate (or advise about reputation of suggested one) based on the reputation information, available from the previous experience. VO members can register with the VO Trust Management Service to receive reputation information about a particular entity. This information could be used to avoid potential problems in later stages of member activity. VO Trust Management Service should be protected, i.e. only authorised entities can access the trust information.

Naturally, the trust-related information collected using different options, affect trust level of the VO members, as depicted in Figure 6.

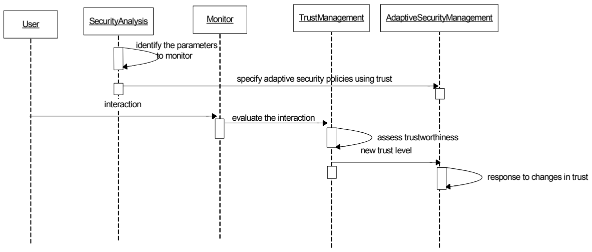
User
SecurityAnalysis
Monitor
TrustManagement
AdaptiveSecurityManagement
identify the parameters to monitor
specify adaptive security policies using trust
interaction
evaluate the interaction
assess trustworthiness
new trust level
response to changes in trust

Figure 6. Changes of trust levels resulting from the performance monitoring/feedback

5. Conclusions

The management of VO between organisations is currently available in insecure systems appropriate for non-commercial academic collaborations only, or by using proprietary solutions from single vendors that all organisations participating in the VO are required to use. This creates a very expensive entry cost for a VO which is only worthwhile if the VO survives for a long period providing substantial benefits. In industries such as aerospace and defence with fifteen years development cycles followed by another fifteen years of deployment for a single product this can be justified. However, most businesses require more dynamic cooperation between organisations. This can only be achieved when the existing IT provision of potentially cooperating organisations have the potential to interoperate. This will only come about when the run compatible implementations of open standards.

In the context of our work we use Web Services as the underlying technology to implement of models. The requirements for VO management described here can be met by a subset of the current WS* specifications, but they require secure, stable and interoperating implementations from a variety of IT vendors.

In particular, from the discussion earlier in this paper, we can identify the following areas, which need strong support of stable specifications and robust implementations:

- Service choreography. Being the primary vehicle to model, validate and facilitate peer-to-peer interactions between the business partners involved in VOs, choreography needs to be supported by standards and non-proprietary implementations. As it was mentioned earlier, W3C leads choreography standardisation, which resulted in Web Services Choreography Language Specification (WS-CDL) [8]. The first tools are beginning to appear to support WS-CDL [28] and hopefully the bigger vendors will follow.
- Reliable messaging support
- Event-based communication support
- Integration of the mentioned technologies with security support, policy management, QoS, etc.

Until these technologies are available the current limitations on automated VO formation and management will continue. This is therefore a plea to encourage IT vendors to produce secure, stable and interoperating implementations of the required WS* specifications.

6. Acknowledgements

The results presented here are partially funded by the European Commission under contract IST-2003-01945 through the project TrustCoM [17]. The authors would like to thank members of other organisations working in TrustCoM: SAP, ETH Zurich, European Microsoft Innovation Centre, Kings College London, University of Milano, University of Kent, BAE Systems, and IBM.

7. References

[1] Goldman, S.L., Nagel, R.N, Preiss, K. (1995), Agile Competitors and Virtual Organisations – Strategies for Enriching the Customer, New York: Van Nostrand Reinhold.

[2] Ritter, S. (2003) ebXML: The "Other" Web Services Architecture, http://it.sun.com/eventi/jc03/pdf/parallela_1/giorno29/02-ebxml.pdf, Java Conference 2003, May 2003, Milano, Italy (accessed February 2005)

[3] Capell, S. (2002). E-Business Standards First Steps Towards Scalable Interoperability. http://lists.ebxml.org/archives/ebxml-dev/200209/pdf00000.pdf

[4] Steve Vinoski, "WS-Nonexistent Standards," *IEEE Internet Computing* , vol. 8, no. 6, 2004, pp. 94-96.

[5] Web Services Choreography Description Language Version 1, http://www.w3.org/TR/2004/WD-ws-cdl-10-20041217/ (accessed February 2005)

[6] Security in a Web Services World: A Proposed Architecture and Roadmap (2002) http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnwssecur/html/securitywhitepaper.asp (accessed February 2005)

[7] Papazoglou, M.P., Dubray J.J. (2004), A Survey of Web Service Technologies, University of Trento Technical Report #DIT-04-058, Trento, Italy

[8] W3C's WS-CDL Targets Peer-to-Peer Web Services Collaboration, http://www.intldeveloper.co.uk/news/business+news/w3c+targets+web+services.asp

[9] Lehmann, (2005) M. Deploying Large-Scale Interoperable Web Services Infrastructures, Web Services Journal, Jan, 2005

[10] Wildeman L, Alliances and networks: the next generation, International Journal of Technology Management, 15: 1/2, pp. 96-108 1998.

[11] Ferguson, D.F., Storey, T., Lovering, B. and Shewchuk, J. (2003). Secure, Reliable, Transacted Web Services http://www-106.ibm.com/developerworks/webservices/library/ws-securtrans/ (accessed February 2005)

[12] VOMS - http://hep-project-grid-scg.web.cern.ch/hep-project-grid-scg/voms.html

[13] Biplav Srivastava and Jana Koehler2003, Web Service Composition – current solutions and open problems. ICAPS 2003 workshop on planning for Web services, Trento, Italy.

[14] E. Weitzenboeck, 2002, VE Model Contracts, Deliverable D17a from the Alive IST project IST-2000-25459 - http://www.vive-ig.net/projects/alive/models.html

[15]Dubray, J.J. WS-CDL – Choreography Description Language- http://www.ebpml.org/ws_-_cdl.htm

[16] Chiusano, J., M. and Booz, A., H. (2003). Web Services Security and More: The Global XML Web Services Architecture (GXA) http://www.developer.com/services/article.php/2171031 (accessed February 2005)

[17] Dimitrakos, T. et al, 2004. TrustCoM - A Trust and Contract Management Framework enabling Secure Collaborations in Dynamic Virtual Organisations. ERCIM News No. 59, pp 59-60, Sophia Antipolis, France. http://www.ercim.org/publication/Ercim_News/enw59/dimitrakos2.html (accessed December 2004).

[18] The Anatomy of the Grid: Enabling Scalable Virtual Organizations. I. Foster, C. Kesselman, S. Tuecke. International J. Supercomputer Applications, 15(3), 2001.

[19] WSRF, The WS-Resource Framework http://www.globus.org/wsrf/

[20] Afshar, M., Hilderbrad, H., Kavantzas, N. Shaffer, D. Surpur, A. (2004) Process-centric realization of SOA: BPEL moves into the limelight Web Services Journal, Nov,2004 http://www.findarticles.com/p/articles/mi_m0MLV/is_11_4/ai_n7071401/pg_1

[21] EbXML. (2003). The ebXML Framework, http://www.ebxml.org (accessed February 2005)

[22] A Svirskas, R. Roberts, "An architecture based on ebXML and Peer-to-Peer technologies and its application for dynamic virtual enterprises of European SMEs", XML Europe 2004, Amsterdam, 19-21 April

[23]Leymann, F., Roller, D., and Schmidt, M.-T. (2002). Web services and business process management. IBM Systems Journal, Volume 41, Number 2, 2002, p. 199.

[24] Adams, H. et al. (2002), Best practices for Web services: Part 4, A Managed Public and Private Process Application Pattern Scenario http://www-106.ibm.com/developerworks/library/ws-best4/?n-ws-12122

[25] Brown, A. et al. (2003). Using Service-Oriented Architecture and Component-Based Development to Build Web Service Applications. A Rational Software Whitepaper from IBM. TP032, April 2003, http://www-106.ibm.com/developerworks/rational/library/content/03July/2000/2169/2169.pdf (accessed February 2005)

[26] Keoh, S.L. Lupu, E., Sloman, M.: PEACE: A Policy-Based Establishment of Ad-hoc Communities. ACSAC 2004: 386-395

[27] Chappel, D., 2003, Will the Real Reliable Messaging Please Stand Up?, http://xml.coverpages.org/Chappell-WSRelSOAP.pdf

[28] Pi4 Technologies - a vendor of WS-CDL open source tool. http://www.pi4tech.com