50 ADDRESSING AUTONOMY AND INTEROPERABILITY IN BREEDING ENVIRONMENTS

Toni Ruokolainen and Lea Kutvonen
Department of Computer Science
P.O. Box 68 (Gustaf Hällströmin katu 2 b)
FI-00014 UNIVERSITY OF HELSINKI
FINLAND

Toni.Ruokolainen@cs.Helsinki.FI, Lea.Kutvonen@cs.Helsinki.FI

Modern networked enterprises can preserve their competitive edge only by adopting a modern architecture for their computing systems, and thus, become agile with respect to changes in computing technologies, for pressures for modifications in external business processes, and for management of simultaneous and changing memberships in various business networks. Establishment of virtual enterprises is an essential trend and in progress of merging into research results of agent technologies and Service-oriented Computing. This paper describes the meta-information and agents needed for verifying interoperability during the breeding process for virtual enterprises. Taken together with operational time monitoring, the facilities provide type-safety across the bindings between members of the community. Behind the discussion is a set of prototype implementations of middleware facilities.

1. INTRODUCTION

Modern networked enterprises can preserve their competitive edge only by adopting a modern architecture for their computing systems, and thus, become agile for changes in computing technologies, for pressures for modifications in external business processes, and for management of simultaneous and changing memberships in various business networks. Establishment of virtual enterprises, i.e., loosely-coupled communities of autonomic business services is an essential trend and in progress of merging into research results of agent technologies and Service-oriented Computing.

The contribution of our research on this large field includes a contract-driven architecture (with supporting platform service implementations) where each business service is provided an enterprise level agent for a) managing virtual enterprise memberships, b) ensuring that interoperability requirements are met, c)

detecting and managing breaches, and d) negotiating (refining) the contract with other community members.

The contract is formed by the breeding environment facilities including potential member discovery and selection based on interoperability requirements. The contract itself is designed to cover a range of aspects from business considerations to technical accessibility. The emphasis has been on ensuring continuous interoperability while preserving autonomy of business services. In this paper we do not address the trustworthiness of the selection process; extensions on trust management between enterprise agents and business services are discussed separately (Ruohomaa et al., 2006; Kutvonen et al., 2006).

For us, interoperability means effective capability of mutual communication of information, request of processing and results, suggestions, and commitments. It covers various aspects: technical interoperability concerns with ability to transport messages between computational services; semantic interoperability concerns with shared understanding of message contents by the senders and the receivers both in terms of information representation and messaging sequences; pragmatic interoperability concerns with the willingness of partners for the actions committed for the collaboration. The technical and partially the semantic interoperability levels have been covered by many solutions; the pragmatic dimensions are still under work and of specific interest to us.

The requirement of using autonomic business services for the communities raises a need for global infrastructure that provides facilities for comparing service types while matching service offers (Ruokolainen and Kutvonen, 2006). The business services (or the computational counterparts) are developed independently, and the meta-information describing the collaborations (business network models) and the members (service offers) need common ontologies for making the breeding process and the operational time monitoring of interoperability possible (Kutvonen et al., 2005).

Especially related to the pragmatic interoperability, preservation of the business service autonomy raises new challenges. First, the business service must be supported by a computational implementation providing capability of performing requested services. However, it is often the case that not all of this capability is given out into all kinds of business networks. The computational services must be guarded by enterprise policies for use. In addition, the business service comes partially defined by its role (commitments, responsibilities, rights) in the community it participates. This aspect is governed by the contract. The local policies and the community wide contract may contradict at operational time, causing breaches that need to be managed. The breaches may be caused by the natural mismatch between the key issues for the enterprise and the community, and there is no need to try to remove this feature. The breaches may also be raised by operational time changes of enterprise policies — as part of their autonomy.

This paper describes the meta-information and agents needed for verifying interoperability during the breeding process for virtual enterprises. Taken together with operational time monitoring, the facilities provide type-safety across the bindings between members of the community. Behind the discussion is a set of prototype implementations of middleware facilities. Section 2 introduces our framework for automated business network management. Section 3 introduces our

concepts and mechanisms for overcoming autonomy-related problems in inter-enterprise collaboration.

2. A BUSINESS NETWORK MANAGEMENT FRAMEWORK

The web-Pilarcos framework proposes a federated model of inter-enterprise collaboration networks, or virtual enterprises, comprised of autonomic business services. The inter-enterprise business collaboration networks are called eCommunities and they are established dynamically to serve a certain business scenario or opportunity. A business service denotes a set of functionalities provided by an enterprise to its clientele and co-operators, and is governed by the enterprise's own business rules and policies, as well as by business contracts and regulatory systems controlling the business area.

The properties of an eCommunity are described in a Business Network Model (BNM). (Kutvonen et al, 2005). The structure of an eCommunity is defined as a set of business roles and their inter-connections. Responsibilities for the participants in terms of descriptions for expected behaviours are declared on the one hand by the behavioural descriptions included in the role descriptions, and on the other hand by policies and business rules declared in the BNM. Non-functional properties, which declare quality-of-service, communication security, and trust-related requirements for the eCommunity, are also provided by the BNM.

A business role is described as a set of service types, a set of role composition rules, and role assignment rules. A service type (Ruokolainen and Kutvonen, 2006) is an abstract definition of service capabilities and behaviour which enables efficient interoperability validation for business services. Service type describes the service's externally visible behaviour in terms of an interface protocol (a bilateral process description), structures for the exchanged documents in terms of XML-Schema definitions and optionally semantic annotations embedded in document structures as references to common business ontologies. The services types are composed to with a set of intra-role coordination rules form business role functionality. The coordination rules relate communication actions in distinct service types with causal and temporal inter-dependencies. An optional set of role-specific assignment rules, that is, additional constraints for service attributes and service providers can also be given in the business role definition.

The eCommunities are established by utilising a breeding environment which comprises of service trading and community population services, and multilateral negotiations between the participants (Kutvonen et al., 2005). The breeding environment services, such as populators and type repositories, are not required from all sites, but can be provided as infrastructure services as a business on its own right. Negotiations are executed between business network management agents (NMA) of each enterprise to refine the contract-templates provided by the breeding environment. The NMAs act as representatives for the autonomous business services during the breeding process and operation of collaboration networks. The elements of the business network management environment are illustrated in Figure 1.

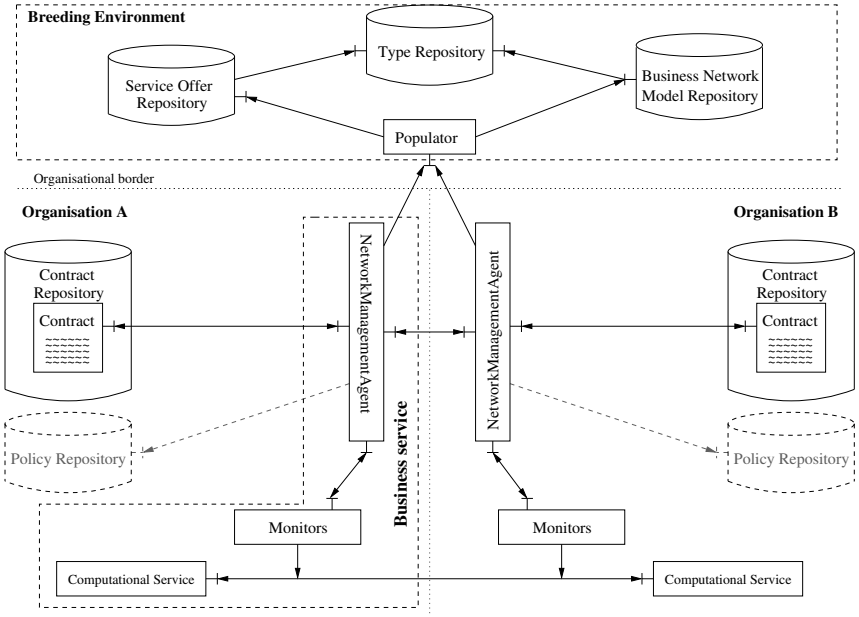

Figure 1 – An overview of the web-Pilarcos architecture. Arrows represent communication relationships, solid boxes are active agents and cylinders are information repositories.

The service trading mechanisms is provided by the service offer and type repositories which provide functionality similarly to the ODP trading and type repository functions (ODP, 1997; ODP, 1999). The type repositories are persistent storages of service typing information which are used as the primary means for achieving interoperation between business-services (Ruokolainen and Kutvonen, 2006). The service offer repositories are required to accept offers only from identifiable, tractable service providers. During publication of a service offer a conformance validation against the corresponding service type is initiated by the service offer repository; this is a necessary operation, in addition to runtime monitoring, for upholding type safety and interoperation in the system. Although the service offer repository itself thus becomes trustworthy, the service offers or service providers do not. A separate mechanism is designed for potential partners to collect experience information about provided services and pass around reputation information (Ruohomaa et al., 2006; Kutvonen et al., 2006)

The populator represents a breeding process phase where appropriate business service providers are selected for eCommunity roles. The populator function takes a Business Network Model and utilises service type and service offer repositories for fetching compatible business service providers for each eCommunity role. The populator selects the business services to an eCommunity on a basis of a constraint satisfaction process which considers the compatibility of the business service attribute values. When a set of compatible service offers have been found, populator returns the description of a populated eCommunity to the initiator of the population

process. Population processes are initiated by enterprises willing to establish business collaborations. Publicly available Business Network Model repositories are used to publish and discover appropriate Business Network Models (Kutvonen et al., 2005).

For the negotiation phase, the web-Pilarcos framework provides generic negotiation interfaces and meta-level protocols. The negotiations and eCommunity management during the operation of the community are handled by the NMAs (Kutvonen et al., 2005; Metso and Kutvonen, 2005).The collaboration management interfaces of NMAs provide functionality for example for renegotiating part of the collaboration contract, to query the status of the contract, and to control transitions between eCommunity epochs, that is, distinct phases of the collaboration (Metso and Kutvonen, 2005). NMAs utilise local contract and policy repositories which store information concerning the electronic contracts, business rules and policies effective in the corresponding virtual enterprise.

3. ADDRESSING AUTONOMY IN VE ENVIRONMENTS

Autonomy of business network participants manifests itself as degrees of freedom given to them during the virtual enterprise life-time. A modern VE support environment should allow freedom of: 1) design and implementation of services, 2) decisions concerning the operation of services, and 3) willingness to collaborate. Nonetheless, interoperability and autonomy are inter-dependent aspects of collaboration: autonomy allowed for VE participants must be compensated with mechanisms to guarantee interoperation in presence of the corresponding freedom. In the following we describe the mechanisms that are used to support autonomy in the web-Pilarcos framework during eCommunity breeding, negotiation and operation and provide mechanisms for solving interoperability problems induced by the corresponding aspects.

3.1 Autonomy during eCommunity breeding process

The freedom of implementation of computational services leads to technological heterogeneity. Most of the implementation technology-related barriers for inter-enterprise collaboration can be solved with use of appropriate middleware, such as Web Services (W3C-WS, 2006) as a technological unification layer. Provided with technical compatibility between the computational services, more challenging aspects of autonomy and interoperability between business services can be addressed.

The freedom of design, of both existing legacy systems and newly developed services, induces semantic interoperability problems which must be solved during the VE breeding process. In the web-Pilarcos framework semantics are considered with respect to the meaning of individual messages exchanged between business services as well as the behavioural aspects of the business services. Interoperability, that is behavioural compatibility and substitutability, of business services is validated between service types (Ruokolainen and Kutvonen, 2006; Kutvonen et al., 2005). Interoperability is propagated to the level of business services via utilising service type checking during service offer publication. Thus service offer and

service type repositories provide in concert the required functionality for validating semantic interoperability during community breeding.

Freedom of design and decisions concerning the operation of business services inflict both technical and pragmatic interoperability issues during selection and configuration of business services. Depending on the visibility and accessibility of service offer properties, business service attributes contribute to different phases of the breeding process. The information type (syntax and semantics) of an attribute is used as part of the service discovery criterion during the population process for finding compatible service offers. Values of publicly available service attributes affect service selection during the population of an eCommunity. Some business service attributes can be announced as private and their values are available only after prior commitments, i.e., during or after eCommunity negotiations. Private attributes express pragmatic aspects of interoperability related to local policies and business rules and provide service providers for context specific evaluation and expression of business service properties.

Table 1 provides a characterisation of the different business service attribute kinds that can be attached to service offers. Attributes are classified by visibility (public or private), type of the value (static or dynamic), and accessibility (how the actual attribute value can be obtained). An example of a typical usage of such attribute kind is also given. The "must negotiate"-kind of private attributes are used in situations where the service provider does not want to expose the attribute value before the client has at least identified himself (we presume that negotiations are held between identified partners). The service provider does not reveal a "must commit"-attribute value before a commitment to join the collaboration has been made by the client

Table 1 – Characterization of business service attribute kinds.

Visibility	Value type	Accessibility	Example
Public	Static	Published in the service offer	Service provider name
Public	Dynamic	Obtainable dynamically from service provider	Simple pricing information
Private	Dynamic	Must negotiate with the provider	Quality of service
Private	Dynamic	Must commit to contract before the value is exposed	Discount on the price of services

3.2 Negotiating the aspects of autonomy

Enterprises may express their willingness to collaborate during the negotiation phase of the breeding process. Negotiations are used primarily to come into conclusion about shared properties among autonomic agents; however, negotiation is also a mechanism to introduce autonomic decision procedures into the breeding process. An enterprise can keep its decision procedures and preferences private, since a negotiation about joining the collaboration leaves an option to decline the invitation to join the community. During the negotiations, partners agree about the private attribute values of business services discussed above, as well as granularity and permanence of collaboration commitments.

The participants of an eCommunity may commit to provide their business service functionality using three different time spans: the whole eCommunity life-time, epochs, and business service sessions. Individual actions or business transactions are considered too fine-grained to be used as objects of commitments. An epoch is a block of collaboration defined in the BNM where the set of roles and services is stable; an epoch change captures a major reorganisation of the collaboration structure, membership, and commitments (Kutvonen et al., 2005). Service sessions are the smallest behavioural structures that can be committed to in the web-Pilarcos framework. The session boundaries are prescribed by the service types and they are used to define "natural" modular units of business service behaviour. Each of these commitment types can be in turn pre-negotiable or re-negotiable. A pre-negotiated commitment can not be changed between the commitments points, whereas re-negotiable commitments allow enterprises to express changes in their willingness to collaborate and decisions concerning business service provision during operation of the eCommunity.

3.3 Addressing autonomy during operation of an eCommunity

Autonomy of participants during the operation of an eCommunity manifests itself as 1) local policy conflicts, which usually lead to, 2) contract breaches. Local policies, that is, organizational policies and business rules are declarative rules that allow enterprises to control how, when and in which context their business services can be used. Local policies modify behaviour of business services by requiring certain actions to be taken instead of the others, by prohibiting actions, or influence the way an action is taken.

Local policy conflicts occur typically when an enterprise has previously committed to deliver a certain business functionality after which the local policies have changed due to changes for example in business strategies, or alliances. If local policies are expressed in the corresponding service offers, conflicts between the local policies and eCommunity requirements can be identified during the breeding process. However, organisational policies and business rules are inherently dynamic entities subject to organizations' autonomic intentions and not even necessarily published outside the organisations. Contract breaches occur either due to conscious decisions made by participants (e.g. changes in local policies) or due to involuntary failures to deliver the required business functionality. Runtime monitoring of business service behaviour against local policies and eCommunity contract can be utilised to identify and deal with both local policy conflicts and contract breaches.

4. CONCLUSION

The B2B-middleware developed in web-Pilarcos project provides support for autonomously administered business services that collaborate in a loosely coupled eCommunities. The eCommunity establishment and maintenance does not need facilities for distributed enactment of business processes, but instead focuses on ensuring semantic and pragmatic interoperability.

In many projects (see survey for example in (Camarinha-Matos, 2003)), the breeding environment provides facilities for negotiating and modelling the

collaboration processes; the operational environment controls the enactment of the processes. Many of these virtual enterprise support environments use a shared abstract model to which all enterprises have to adapt their local services. In addition, the negotiations lean on an amount of human intervention for building the required trust relationships. In contrast to this, the approach in the web-Pilarcos project is a federated one and automation-oriented. Enterprises seek out partners that have services with which they are able to interoperate. The federated model provides the necessary flexibility and mechanisms to address pragmatic interoperability issues and autonomy. The automation is limited to routine decisions: strategic choices are left to humans while routine decisions concerning technical and semantic interoperability during eCommunity establishment or policy enforcement are provided with automated infrastructure facilities.

4. REFERENCES

1. Camarinha-Matos, L. M. (2003). Infrastructure for virtual organizations: where we are. In *Proceedings of ETFA'03 - 9th international conference on Emerging Technologies and Factory Automation*, Lisboan, Portugal.
2. Curbera, F., Goland, Y., Klein, J., Leyman, F., Roller, D., Thatte, S., and Weerawarana, S. (2002). *Business Process Execution Language for Web Services (BPEL4WS) 1.0.*
3. Kutvonen, L., Metso, J., and Ruokolainen, T. (2005). Inter-enterprise collaboration management in dynamic business networks. In *OTM Confederated International Conferences*, volume 3760 of *LNCS*. Springer-Verlag.
4. Kutvonen, L, Metso, J and Ruohomaa, S. (2006). From trading to eCommunity population: Responding to social and contractual challenges. Submitted manuscript.
5. Metso, J. and Kutvonen, L. (2005). Managing Virtual Organizations with Contracts. In *Workshop on Contract Architectures and Languages (CoALa2005)*, Enschede, The Netherlands.
6. ODP (1997). *ISO/IEC 13235: Information Technology. Open Distributed Processing - ODP Trading function.* ISO/IEC JTC1.
7. ODP (1999). *ISO/IEC 14769: Information technology - Open Distributed Processing - Type repository function*. ISO/IEC JTC1/SC7.
8. Ruohomaa, S, Viljanen, L and Kutvonen, L. (2006). Guarding enterprise collaborations with trust decisions – the TuBE approach. *The 1st International workshop on Interoperability Solutions to Trust, Security, Policies and QoS for Enhanced Enterprise Systems). In the frame of I-ESA 2006 conference.* Hermes-Science Publishers.
9. Ruokolainen, T. and Kutvonen, L. (2006). Service Typing in Collaborative Systems. To appear in proceedings of INTEROP-ESA 2006.
10. W3C-WS (2006).*Web Services Activity*. W3C. http://www.w3.org/2002/ws/.