

Tools and infrastructure facilities for controlling nonfunctional properties in inter-enterprise collaborations

Lea Kutvonen

Department of Computer Science, University of Helsinki, Finland

Email: Lea.Kutvonen@cs.Helsinki.FI

Abstract

Management on inter-enterprise collaborations (business networks) still face grand challenges especially in terms of managing interoperability of nonfunctional properties (NFPs) at various levels (technical, semantic, pragmatic). This paper identifies three categories of nonfunctional properties (collaborative for business level, contractual for communication issues, extrafunctional for computational dependability) and describes how facilities are provided in the Pilarcos architecture for declaring, negotiating and enforcing agreed nonfunctional properties on collaborating business services and their joint activities.

1. Introduction

It has become commonly accepted that modern IT-supported business brings in business models that are networked in nature: business networks, extended enterprises, virtual enterprises, and supply chains are frequently discussed.

The computing architecture solutions, like service-oriented computing (SOA, SOC), (semantic) web services, (semantic) business process management, and e-contracting systems, all address some aspects of the collaboration management. However, the question of incorporating the management of nonfunctional properties - or indeed, even agreeing the scope and definition of nonfunctional properties - still remains open.

The Pilarcos architecture belongs to the group of SOA-based architectures for business service collaboration management, with special emphasis on providing business level concepts for managing application-based services and providing facilities for managing problems caused by business service autonomy. Therefore, the suggestions are in many cases be applicable to other approaches as well. The main difference of the Pilarcos architecture to many other solutions lies in the expectation

of the computing platform to be encapsulated within an enterprise. Although we consider model-driven service production techniques in other contexts, the design choice made for inter-enterprise business processes remains to be metadata for negotiation and monitoring, excluding an architecture decision on using such metadata for enactment of services.

In earlier WODPEC papers [1–4] we have shown, how Pilarcos architecture is basically relying on the RM-ODP concepts and functions [5–7] as well as hooks into the ODP enterprise language [8]. Those papers also explain our suggestions on adding concepts of business service and organisation-aware functions in the amendment process of RM-ODP.

This paper describes the key ideas used in the Pilarcos architecture in the area of nonfunctional properties (NFPs) of inter-enterprise collaborations, business services and supporting computing and communication facilities. At the technical level, the required concepts of policies (obligations, permissions and prohibitions) are already present in the RM-ODP terminology; however, in this paper we elaborate the use of those concepts in an inter-enterprise collaboration context, focusing on the different needs of modifying the collaborative behaviour (functionality) and negotiating and enforcing agreed qualities of the utilised functionalities.

Section 2 locates three categories of NFPs within the Pilarcos architecture, and identifies facilities that address their definition, negotiation, declaration and monitoring. The subsequent sections discuss the management phases in more detail: the collaboration strategies and goals (Section 3), eContract negotiation and decision-making (Section 4), using monitors to control the NFPs at operational time (Section 5), and using NFP declarations for managing communication platform usage (Section 6). We recall that the necessary metadata support is already in existence in the Pilarcos middleware repositories, and that the business architecture definition tools and decision-making support tools are conceptually supported by the present architecture.

2. Services and NFPs

The Pilarcos architecture describes an open, global ecosystem where new business networks can be established [9, 10]. In the open network, enterprises make available business services that they administer independently. Together with other enterprises they form task forces in which new kind of business scenarios are developed. Business networks are ad-hoc, loosely-coupled, eContract-governed collaborations with the objective of fulfilling a business scenario. Some new infrastructure services (B2B middleware services) are made available in the open network, such as services for business service discovery and selection, knowledge about existing business network models, and ontologies related to service types. In order to be able to participate this ecosystem, the enterprises have to have a private agent too, for supporting local decision-making and running the joint B2B middleware protocols with other partners in the ecosystem. The protocols are involved with eContract negotiation, monitoring, breach management, and reputation information distribution.

The key concept in the Pilarcos architecture is that of business service. A business service is a software-based (distributed) service, administered by a single authority. The providing enterprise has policies restricting its behaviour. The business service is in addition governed by an eContract in each collaboration it provides service for, the terms of scope/functionality, properties and accessibility of the service. There is no guarantee that discrepancies between eContracts, enterprise policies and service capabilities would not rise. Instead, a breach detection mechanism through monitoring is an essential part of the architecture.

As a reminder of the ecosystem support structures, Figure 1 illustrates the general architecture. Each enterprise has a private set of enterprise policies governing the services it provides. Each business service is supported by a constellation of software elements, and the constellation is guarded by a monitor. The monitor is responsible for running restricting rules from two sources, namely the eContract and the enterprise policy. The business network agents represent the enterprises and implement the necessary negotiation, control and management protocols between themselves. For example, when a breach is detected as a monitor detects a message contradicting its rules, it sends a notification to the local agent, that in turn may request breach recovery process to be started among the partners.

The NFPs under discussion are associated with a) inter-enterprise collaboration (i.e., business network) as a whole (in a recursive view, the collaboration itself can be viewed as a business service); b) the business service

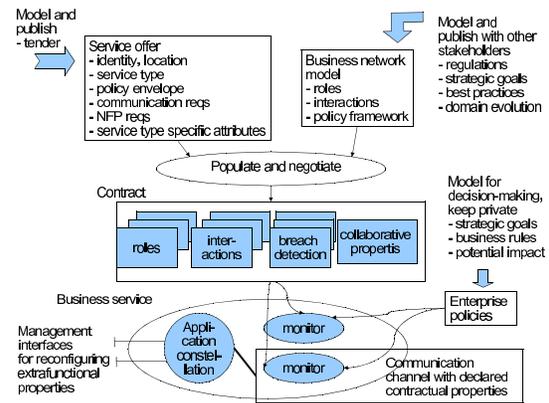


Figure 1. Pilarcos architecture view.

in a role in that collaboration; and c) the communication channel between the business services; and the computational services implementing the business service.

The associated categories of NFPs we name as collaborative, extra-functional, and contractual, as shown in Figure 1. The formalisation of the concepts is started in another paper in parallel with this [11]. The collaborative aspects address the pragmatic interoperability issues, i.e., processes and policies, and methods of decision-making on collaborations, e.g., risk, business value, trust and reputation. Again, there is need to define policies that are commonly understandable but dependent on all business domains involved.

The extra-functional properties of services address issues relevant for managing the dependability of a business service. The detailed structure of this category is dependent on the assumed computing platform for the business applications, because the aspects represent such features of the services that are expected to be dynamically managed.

The contractual aspects depend on the assumed communication platform: e.g. distribution transparency, privacy-preservation, transactionality.

We embed NFP management in inter-enterprise collaborations in the overall collaboration management processes as follows. First, business networks are modelled not only from the functional point of view but also declaring the NFPs of concern. Second, the architecture requires that when enterprises publish their service offers, they make claims on the NFPs as well. Third, the NFPs are covered in the eContracting process. Fourth, the monitoring systems detect NFP faults in service behaviour. Finally, we expect communication and computing platforms to be eventually restructured in such a way that the NFPs they support can be configured by declarative reconfiguration.

3. Designing collaborations

Traditionally, the establishment of business networks starts by negotiation of the joint objectives and goals, and collaborative definition of the joint processes, and definition of the methods of connecting individual computing elements to a coherent whole. This phase is supported by breeding environments where selection of partners, learning about their capabilities, and designing the joint business network model takes place. In this process the set of functionality is determined, as well as a set of business policies that must be adopted.

Although all this is necessary for the business network establishment, it is not necessary to perform the whole process independently for each business network. Neither is it necessary to repeat the whole process when partners have wishes to make changes to the collaboration goals, processes, or supporting applications or computing platforms.

We have separated the business network design phase from the network establishment phase. The business network models can be collaboratively designed, verified and validated for their suitability. These models also provide a common vocabulary for enterprises to use at the business network establishment negotiations: When a collaboration is being established, the pragmatic interoperability (processes and policies) is tested between partners. Thus, the business services forming a collaboration do not necessarily have a joint history in the breeding environment that would enforce interoperability but are just introduced to each other in a refining negotiation of the eContract.

It is beneficial to create rather abstract behaviour groups when designing business network models, to support evolution of collaboration styles. Each model can be further refined into alternative behaviours by choosing the guiding policy value at the eContract.

The ability to dynamic policy management is a strong tool: selecting policy languages and targets suitably, most business management needs related to strategies and business rules can be modelled and transformed to rules that can be monitored at runtime. Effectively, the introduction of different types of policies allows mapping business domain guidelines directly to B2B middleware facilities.

For defining the business network models, a design tool and environment is needed. The business network models comprise of business process models expressed in terms of roles (service requirements) and interactions between the roles (as in [8]). The NFPs are then added on to collaboration, role and interaction levels. We use a separate aspect language for each kind of property. As the business network model is designed, the model itself

has to be structured in such a way that it allows participating enterprises to use monitor the described features. In order to make the rules usable, the aspect languages need to be commonly implemented to allow transformations from the declarations made in the eContract to the actual monitoring rules that are used for detecting deviations from the business network model template or the more specific eContract instance.

The related business process modelling languages address mainly the functional properties of the workflows. The works on extending the languages to the direction of NFP support try to annotate the same language with additional concepts. Often, the approach is to execute or enact the model as such by a distributed workflow engine. Considering the present business process definition languages (e.g., BPMN, WS-BPEL, XPDL, WS-CDL; survey [12]) we do not deviate from the commonly used set of concepts for partners and interactions, but emphasize some special features that are relevant for service-orientation, NFP management, and evolution support on the service markets.

In the analysis of monitoring rule language requirements (e.g., RuleML [13]), policy languages (e.g., Ponder [14]), service level agreement languages (e.g., WSLA, WS-Policy; survey [15]), or eContract languages (e.g., BCA [16], [17]), we have seen that the expressive power of the already existing languages is sufficient. Support is found for major groups of a) obligation, prohibition, permission, b) time, duration, (partial) ordering of events, c) identity of business services, d) role and interaction names, policy names and value domains (these are scoped by the business network models), e) assignment rules to guide assigning business services to roles (fulfilment rules to monitor whether the assigned service keeps fulfilling the role requirements and the fulfilment rule criteria; both of these can address collaborative, contractual or extra-functional aspects as well as identities or properties of services as declared in their service types or properties of interactions), f) reputation of business services (not hosts, enterprises or software alone), g) service level and dependability, defined separately for each service type, h) binding service level, and i) expressions on collecting business value and business asset related information.

Architecture-wise the aspect language choice is not significant: the administrators of the system and business modellers need easy means for adjusting the collaborations and business services by technically reconfiguring the underlying monitor system. Therefore, it is essential to provide a good set of monitors and a suitable set of transformations from the management interfaces to the configuration parameters of the monitors.

The design of business network models is by nature

a distributed activity: The business network models are created in a unification process affected by all stakeholders, regulatory frameworks, and best practises [18], and the resulting model should even follow the regulations on that business domain or domains addressed. Therefore a common vocabulary is needed on-line for the designers to use, and strong guidance towards reuse of existing business process models is necessary. Therefore, we use the Pilarcos metainformation repositories (business network model repository, type repository) to provide a shared knowledge base for the modelling tools used in the enterprises.

The created models are published in an abstract (black box [19]) form, only revealing the obligated interactions, the NFP frameworks and breach management rules. This view is then to be refined by other design and configuration phases. The business network models are constructed by connecting together business processes that each have a single starting point, single termination point and one functional goal (which is essential for verification purposes too [20]). Connecting the processes together takes place by explicating which roles at each process model must collocate at a combined role. The new role inherits the service requirements from all these collocated roles. The business process models are annotated by criteria for assignment of business services and operational time criteria for not causing a breach. When the combined roles are created, annotations are added for restricting collocations, for example, to avoid legally invalid combinations of supervision relationships.

The rules determining breaches are explicit, as well as agreement on what recovery process to use. For this purpose, a) multiple recovery business processes are defined and consistently viewed as a set of best practises definitions, and b) all business network models should be analysed to determine their recoverability style; some networks are not able to recover from breaches but need to be terminated, while others may recover from the loss of some members, and further some require a set of compensation actions to take place before either continuing operation or terminating.

Before publication, the business network models must be analysed and verified for properties like liveness, fairness, privacy-preservation (data flow sufficiency and minimality), termination of processes, and recoverability. For this purpose, existing business process verification tools are applicable, when each functional business network part is separately analysed.

Finally, we can summarise for the NFPs for collaborations as follows. For each role we associate a) business process composition rules describing how separate tasks from different processes should be joined to the

same performer or how avoiding joint responsibilities is enforced by regulatory systems (assignment rules); and b) breach rules explicitly listing situations and actions that are against the otherwise expected behaviour and NFPs. In addition, for each interaction we associate a) expected gain or loss in terms of identified assets assuming the interaction takes place normally or fails; b) required persistence of business service, i.e., business transaction properties and nonrepudiation properties; c) required service level, for which the correct terminology depends on the service type (and is declared in the published service type definition). Furthermore, overarching the whole eContract (relevant only for instances, not templates) we define invariants about the state of the business network, such as denial of delivering goods before payment is received by another partner.

4. Decision-making and eContracting

As the negotiation of the business network structure and goals have been factored to a separate step that results to an explicit, published model, the eContract negotiation between enterprises becomes more restricted in its scope. Effectively, the negotiation must result into a situation where it is ensured by static validation that interoperability at all levels exists between all parties, and that all parties are willing to participate the collaboration. Furthermore, the refining negotiation must select the policy values to be used for this particular collaboration and stored into the eContract.

The supporting facilities to be used here are as follows [9, 10]. First, the B2B middleware provides population of the business network followed by a generic negotiation protocol between the enterprise agents. The population process ensures that according to the claims in service offers the business services becoming members of the business network can be interoperable at all levels. Then, the proposed eContract draft is set to each enterprise to gather commitments of participation, or further refinements on the policies suggested. In the service offers, just ranges of policy values are announced.

The negotiation cycle ensures privacy of decision-making for each participant. In routine cases, it is possible for the enterprises' agent provide an automated response to the collaboration proposals: an explicit metapolicy guides the agent to pick routine rejections or commitments. Other situations can be recognised, for example, by uncertainty of the trustworthiness of the peers, uncertainty of the strategical benefit of the collaboration, or uncertainty of the acceptability of negative reputation effects caused by a refusal.

The enterprise policies carry responsibility of the following NFPs of the operation:

- meta-policies governing the decision-making; some collaboration proposals or business interactions are clearly either to be accepted as routine cases of normal operation, or clearly to be rejected because the proposal is of uninteresting or untrusted kind due to, for example, proposed partners, or business network model [21];
- decision-making policies in respect to reputation-based trust, risk and importance tolerances [22];
- privacy policy that may overrule any other decision-making reasoning in collaborative interactions; each service, information source and metainformation source must be protected [21];
- constraints for granting use of services;
- furthermore, the service type and business network models should include observable properties that are relevant for the business process re-engineering needs. Such an observable property is for example the satisfaction of clients after completing a session on a business service.

Both for the automated decision-making and for the support of human intervention, we propose to use an expert system to gather the relevant knowledge and to feed governing policies to the enterprise system, i.e., the relevant NFPs of the collaboration and its contributing services [21].

The decisions to join a collaboration balance between the risk of failure or loss of assets as a consequence of participation, and the potential benefits of participation. That is, the expert system should compute a three-value outcome (agree, disagree, call for human intervention) on whether a service or a collaboration is dependable and beneficial for the enterprise in a given context and situation [21]. A dependable service fulfils its business purpose and the use of the service does not involve intolerable risk of monetary loss or reputation loss, for example caused by delivery failures or unacceptable delivery delays. Semantically, the decision to join the collaboration means two things. From the service providers viewpoint, an outsourcing relationship to the rest of the collaboration community becomes effective. From the collaboration point of view, an in-sourcing relationship takes effect. We consider in-sourcing and outsourcing to have technically identical "clauses": three levels of interoperability and commitment to behaviour according to the eContract.

Computationally, the system computes values for risk and risk tolerance, both of which are vectors over a set of assets, such as monetary assets, reputation, fulfilment of purpose, and control of property [23]. For

the risk values, the essential input comes from reputation information, i.e., positive and negative recommendations by members of earlier collaborations. For the risk tolerance, the essential input is from the perceived importance of the tasks or business network. The starting values for the importance and loss scenarios should be created by an extensive risk analysis and strategic business analysis. When the risk vector is compared to the tolerance vector, the decision should be to a) agree, when no tolerance thresholds for acceptability are violated; b) disagree, when no tolerance vector values for disagreeability are violated; and or c) propagate to human decision-making, when any tolerance vector value gets classified differently from the other vector values, all vector values fall between acceptability and disagreeability thresholds, or the meta-policy classifies the case as nonautomatable. When the request is forwarded to human consideration with all the relevant information; the formulations and scope are yet to be detailed. The information should support the understanding the proposed collaboration, its business values and risks, trust on potential partners, privacy-preservation and so on. For the automated cases, the similar decision is based on a set of interoperability levels and NFPs.

In relation to other work (e.g., survey [24], eNegotiation [25], OMNI [26], [27]) and outsourcing management systems we emphasise a) use of predefined contract templates that capture not only business level or technical level issues, but both; b) running a multi-partner negotiation instead of bilateral negotiations; c) support of contract template evolution through the facilities for creating new business network models and policy variations; d) agility of business networks gained by operational time negotiations and renegotiations that is based on ontologies and abstract enough behaviour models created at design time; e) privacy of decision-making and using interoperability knowledge effectively for it; f) potential to use multiple negotiation protocols for different types of collaborations (auctioning systems, simple commitment protocols).

Besides the business network establishment phase, the same kind of decision-making takes place when entering significant tasks or business transactions within the collaboration, as discussed below.

5. Enforcing NFPs in collaborations

During the operation of the business network, the monitors governing the business services can proactively, actively or passively scan the messaging, reporting to enterprise level agents if the eContract is breached. Proactive monitoring holds the message till a decision has been made whether it is safe to send or

receive it, while active monitoring lets the message pass but reports breaches thus potentially causing breach recovery or termination of the collaboration. The passive monitoring just audits the events for later processing.

The breaches can mean failing to fulfil an obligation, or failing to provide the agreed quality level of the service; more formally, failing to provide the level of dependability expected. The concept of dependability, in terms of fulfilling the contracted aspects, can be concertised on two fields. There are general properties that can be set as service level expectations for any service, such as availability, timeliness, and privacy-preservation, or interaction relationship, such as nonrepudiation and immutability. In addition there are properties that are relevant for individual service types, each requiring a definition of value domain and metrics for defining the service levels relevant for the property. For example, reputation information (recommendation) can have a credibility property associated to it, determining how completely a recommendation from that source is assumed truthful. Another example is the traditional QoS levels with different metrics for data bandwidth and jitter in transfer.

The monitors receive rules from eContract and from their local policy repositories. These rules are not guaranteed to be nonconflicting, as they a) may address different issues and they b) can be changed after eContract establishment.

At detected breach situations, decisions are needed on whether the event is serious enough for terminating or leaving the business network. The same type of knowledge about the operational environment can be used, and again the expert system can make automated decisions or redirect the request for human intervention.

Transformations are required to map the rules on monitoring rules or state machines. Analysing the various monitoring needs, the required monitor techniques fall into a few simple cases:

- detectors of denied values or value combinations in message fields;
- detectors of nonacceptable ordering of messages, including failure to complete a task in time;
- detectors of series of messages jointly exceeding the acceptable limits calculated from the messages as they pass the monitor; and
- authorisers that hold the message while investigating whether the intended business transaction is to be trusted.

Using these techniques in various combinations in the application domain context, and utilising the business semantics building from the messages, a rigorous

set of constraints can be built. Especially, the business semantics and social requirements can be encoded.

6. Declarative management of NFPs of services and channels

The above discussion has focused on the collaboration related features. However, a lot of existing solutions appear on the area of controlling the NFPs of actual computational services and communication. As the present platforms do not allow us to actually (re)configure the service implementations as we would like, we emphasise the architecture capabilities of disseminating knowledge of the service capabilities (as legacy), for now.

The inter-enterprise architecture of Pilarcos is built on the assumption of strong service-orientation. Each element in the architecture is a service with an externally observable interface and a hidden implementation that is independent of other implementations. The services are manageable by contract-driven methods, which is the present trend [28]. That means that for each service type a set of properties (i.e., policies) is defined, and for each value available for the property there is a defined corresponding behaviour [29]. The basic behaviour of the service can thus be restricted, or a different implementation strategy can be adopted within the realisation of the service.

We apply this management method on business services and on bindings between services to support the required business interactions. For the realisation of the joint behaviour, the required policies must be mapped to a combination of business service policies and binding policies. For example, a requirement of privacy-preservation at the collaboration level maps to requirements for each involved business service not to forward the classified information out of the community, and to the requirement of using secured communication links between parties.

The interesting NFPs on business services can be divided to two categories, extra-functional and contractual. Extra-functional properties are involved with the dependability of the service and is closely related to the concept of service level. Examples of aspects in this category are persistence of the service (business transaction semantics, nonrepudiation), accuracy of the service (accurate and intact information, timeliness), and availability of the service. These properties are controllable by the service realisation provider by using the computational platform. Modifying these properties requires technical administrative authority over the service.

The contractual properties are focused on the role of the business service and how it relates to the busi-

ness network and its communicating peers. Example of aspects in this category are service level, cost, and restrictions on the acceptable clientele and time of access. These properties are controllable by the business service provider; modifying these properties requires business administrative authority over the service.

The concrete division is based on the way business services are mapped to technical solutions. We have defined a business service as a constellation of business applications providing some services, and a wrapping monitor that guards all the communication with the application constellation. The monitors follow policy rules set by the business administrators in the enterprise. Therefore, the application constellation is a "managed object" that is controlled by extra-functional properties set by the technical administration, while the business service as a whole is a "managed object" that is controlled by contractual properties by the business administration.

On binding management the situation is different as the endpoints of the binding are at different administrative domains. However, the eContract ensures that there is a joint vision on how the binding is structured and on what more primitive protocols are to be supported within to provide the expected level of service.

The situation is analogous to the business network construction itself, only on binding management the "business processes" deal with transporting messages from sender to receivers, authenticating the parties, encrypting the contents, and screening out communication trials at the monitors, for example. Again, when these processes are joint to a whole channel structure, a careful analysis of the correctness of the model is needed.

We call the method of joining the expected channel functionalities together aspect weaving. For each of the expected properties of the communication an aspect model is designed, detailing the roles and interactions required. Examples of the aspect models include a secure channel, authentication of peers, and transactional communication. Weaving these aspect models to the functional model can be done based on pattern matching and graph transformations [30]. However, although each individual transformation might be correct, the overall model may be distorted and become incorrect. Therefore, a feedback loop for the modeller is needed to force corrections and gathering more guidance for the weaving. Script based expert-system support for model weaving in more programming oriented context already exists and we find it profitable to combine these approaches.

As the NFPs of the bindings are not independent from each other, it is necessary to create an ontology of binding service levels to denote different combinations.

For each combination, the channel architecture (template) can be stored after verification and made available in the middleware repositories. When interoperability is ensured at the eContract establishment phase, a channel template can be found from the repository for each binding [31].

7. Conclusion

When designing the Pilarcos architecture, one of the main goals was to incorporate facilities for managing NFPs of business services on all interoperability levels with the same mechanisms as functional properties. Therefore, the meta-information made explicit in the architecture captures at each point the two sides into the same structures. This gives us a number of selected features to the architecture.

First, the business network models define the external business processes between partners' business services, declaring the framework for contracting essential business NFPs too. This design makes the business network models the essential "namespace roots" for the policy names and expressions, which means that we avoid the unreachable goal of an unified ontology of properties to be contracted and managed. This way it is clear that the enterprise policies and contract policies do not totally match, but this is not a problem. On one hand, the business network models and enterprise policies will evolve in time, causing mismatches eventually anyway. On the other hand, the monitoring mechanism is in place to detect operational time discrepancies, and we have therefore decided not to bother to do through discrepancy removal in the population or negotiation phases.

Second, the meta-information repositories give a common, evolvable knowledge base for defining new business models, property frameworks, and using that vocabulary while defining service types and making service offers. The additional work in publishing this information is not a huge overhead, since the design choices are anyway made in the local software engineering processes. Ability to browse and publish the ontology in the engineering process only increases the probability on interoperability between services.

In addition to the collaboration property management, we considered the more traditional computational NFPs, and found, that the management methods for these need to be different from the collaboration level properties. At the inter-enterprise collaboration level, it is essential to negotiate about properties using a shared vocabulary. At the level of managing computational services and communication channels the traditional management interfaces, used to address declared properties.

References

- [1] L. Kutvonen, "Challenges for ODP-based infrastructure for managing dynamic B2B networks," in *WODPEC2004*.
- [2] —, "What applying of the odp viewpoints teaches us about tool-chains," in *WODPEC2006*.
- [3] —, "Using the odp reference model for enterprise architecture," in *WODPEC2007*.
- [4] L. Kutvonen and J. Metso, "Services, contracts, policies and ecommunities – relationship to ODP framework," in *WODPEC2005*.
- [5] *IS10746: Information Technology – Open Systems Interconnection, Data Management and Open Distributed Processing. Reference Model of Open Distributed Processing. Part 2: Foundations and Part 3: Architecture*, ISO/IEC JTC1, 1996.
- [6] *IS13235 ODP Trading function*.
- [7] *IS14746: ODP Type repository function*.
- [8] *DIS15414: ODP Enterprise language.*, 2002.
- [9] L. Kutvonen, J. Metso, and S. Ruohomaa, "From trading to eCommunity management: Responding to social and contractual challenges," *Information Systems Frontiers (ISF) - Special Issue on Enterprise Services Computing: Evolution and Challenges*, vol. 9, no. 2–3, pp. 181–194.
- [10] L. Kutvonen, T. Ruokolainen, and J. Metso, "Interoperability middleware for federated business services in web-Pilarcos," *Interat. Journal of Enterprise Information Systems, Special issue on Interoperability of Enterprise Systems and Applications*, vol. 3, no. 1, pp. 1–21.
- [11] T. Ruokolainen and L. Kutvonen, "Managing non-functional properties of inter-enterprise business service delivery," in *Non Functional Properties and Service Level Agreements in Service Oriented Computing Workshop (NFPSLA-SOC) (co-located with ICSSOC 2007)*.
- [12] W. van der Aalst, A. ter Hofstede, and M. Weske, "Business process management: A survey," in *Business Process Management, Proceedings of the First International Conference*. Springer Verlag, 2003.
- [13] B. N. Grosz and T. Poon, "SweetDeal: Representing agent contracts with exceptions using XML rules, ontologies and process descriptions," in *Proc. Intl. Conf. on the World Wide Web*, 2003.
- [14] N. Damianou, N. Dulay, E. Lupu, and M. Sloman, "The Ponder policy specification language," in *Workshop on Policies for Distributed Systems and Networks (Policy2001)*, ser. Lecture Notes in Computer Science, vol. 1995. HP Labs Bristol: Springer-Verlag, Jan.
- [15] T. Nurmela and L. Kutvonen, "Service level agreement management in federated virtual organizations."
- [16] Z. Milosevic, A. Jøsang, M. Patton, and T. Dimitrakos, "Discretionary enforcement of electronic contracts," in *EDOC2002*.
- [17] A. Goodchild, C. Herring, and Z. Milosevic, "Business contracts for B2B," in *Proceedings of the CAISE00 Workshop on Infrastructure for Dynamic Business-to-Business Service Outsourcing*, 2000.
- [18] A. Norta and P. Grefen, "Discovering Patterns for Inter-Organizational Business Collaboration," *International Journal of Cooperative Information Systems*, vol. 16, no. 3/4, pp. 507–544, Dec. 2007.
- [19] A. Norta, "Exploring dynamic inter-organisational business process collaboration," Ph.D. dissertation, 2007.
- [20] W. M. P. van der Aalst, H. M. W. Verbeek, and A. Kumar, "Xel/woflan: Verification of an xml/petri-net based language for inter-organisational workflows," in *Proceedings of the 6th Informs Conference on Information Systems and Technology (CIST-2001)*, 2001, pp. 30–45.
- [21] L. Kutvonen, "Automating decisions for inter-enterprise collaboration management," in *9th IFIP Working Conference on Virtual Enterprises (PROVE2008)*.
- [22] S. Ruohomaa and L. Kutvonen, "Making multi-dimensional trust decisions on inter-enterprise collaborations," in *3rd International Conference on Availability, Security and Reliability (ARES 2008)*.
- [23] S. Ruohomaa, L. Viljanen, and L. Kutvonen, "Guarding enterprise collaborations with trust decisions — the TuBE approach," in *Interoperability for Enterprise Software and Applications. Workshops EI2N, WSI, IS-TSPQ 2006 proceedings at I-ESA2006*, pp. 237–248.
- [24] B. Medjahed, B. Benatallah, A. Bouguettaya, A. H. H. Ngu, and A. K. Elmagarming, "Business-to-business interactions: issues and enabling technologies," *The VLDB Journal*, no. 12, pp. 59–85, 2003.
- [25] D. K. W. Chiu, S. C. Cheung, P. C. K. Hung, S. Y. Y. Chiu, and A. K. K. Chung, "Developing e-Negotiation support with a meta-modeling approach in a web services environment," *Decision Support Systems*, vol. 40, pp. 51–69, 2004, iSSN:0167-9236, Special issue: Web services and process management.
- [26] J. Vaquez-Salceda, V. Dignum, and F. Dignum, "Organizing multiagent systems," *Autonomous Agents and Multi-Agent Systems*, vol. 11, pp. 307–360, 2005.
- [27] R. Neisse, E. D. V. Pereira, L. Z. Granville, M. J. B. Almeida, and L. M. R. Tarouco, "A hierarchical policy-based architecture for integrated management of grids and networks," in *In Fifth IEEE International Workshop on Policies for Distributed Systems and Networks*, 2004, pp. 103–106.
- [28] H. Weigand, "Infrastructural services in a service-oriented architecture," in *2nd International Workshop on Interoperability Solutions to Trust, Security, Policies and QoS for Enhanced Enterprise Systems (IS-TSPQ)*, 2007.
- [29] G. Kiczales, J. Lamping, C. V. Lopes, C. Maeda, A. Mendhekar, and G. Murphy, "Open implementation design guidelines," in *International conference on Software engineering*, 1997, pp. 481 – 490.
- [30] C. Köllman, L. Kutvonen, P. Linnington, and A. Solberg, "An aspect-oriented approach to manage QoS dependability dimensions in model driven development," in *Model-Driven Enterprise Information Systems. Proceedings of the 3rd International Workshop on Model-Driven Enterprise Information Systems - MDEIS 2007*.
- [31] L. Kutvonen, *Trading services in open distributed environments*. University of Helsinki, 1998, PhD thesis. A-1998-2.