TOWARDS AUTOMATED SERVICE TRADING

Manuel Resinas, Pablo Fernandez, Rafael Corchuelo
ETS Ingenieria Informatica
University of Seville

Keywords: Service Oriented Computing, SLAs, Trading.

Abstract: The service-oriented architecture is a promising means to support outsourcing amongst real-time enterprises. In this context, SLAs (Service Level Agreements) are essential because they grant guarantees about how a service must be provided or consumed. We define service trading as the process of locating, selecting, negotiating, and creating SLAs. Although automating the service trading process is a key characteristic of a real-time enterprise, to the best of our knowledge it has not been completely addressed yet.
In this article, we propose a conceptual framework for automated service trading. Therefore, our goal is not to implement a concrete architecture but to develop a framework that can be used to define, compare and analyse the interoperability of different service trading architectures. The novel contributions of this paper are threefold: we identify the roles and interactions that are necessary to carry out this automated service trading, we motivate and introduce the idea of trading protocols, and we define the elements that are necessary to support an automated decision-making in service trading.

1 INTRODUCTION

In recent years, we have witnessed how new technologies are enabling the emergence of a new age of enterprises that quickly adapt to their ever-changing business environments but keep their costs under control. This is the main characteristic of what has been called *real-time enterprises*. Two elements are the key to achieve this vision: the management and analysis of the information collected by the enterprises related to their business environment, and the ability to use products or services offered by other enterprises as components for further innovation.

The service-oriented computing paradigm (Curbera et al., 2003) and the service-oriented architectures based on web services are the mechanims used to support this idea of using services offered by other companies as pieces of our systems. In this context, SLAs (Service Level Agreements) are a key point because they grant guarantees about how a service will be provided or consumed by establishing both functional and non-functional requirements that must be fulfilled by both parties during the service execution. Additionally, SLAs allow providers to deploy an automated provision of services based on the SLAs agreed with their customers (Ludwig et al., 2005).

We define service trading as the process of locating, selecting, negotiating, and creating SLAs. The service trading is a subprocess of a more general *contracting process* that was already defined in (Ludwig, 2003). Although there are infrastructures to provision SLAs and services (Ludwig et al., 2005) that agree with them automatically, there is little support, to the best of our knowledge, to tackle the service trading process, which is still mostly a human-centric process. However, automating the service trading process is a key characteristic of a real-time enterprise.

In this paper, we take the ideas exposed in (Ludwig, 2003) as a starting point and propose a conceptual framework for automated service trading. The framework is divided into six organisations[1] (discovery, information, selection, agreement making, binding, and trading), and each one cares for a specific subgoal in the whole service trading process. Our goal is not to implement a concrete architecture but

[1]We borrow this term from the GAIA methodology(Zambonelli et al., 2003). An organisation does not amount to a company or a department but to a number of agents that work together

Resinas M., Fernandez P. and Corchuelo R. (2006).
TOWARDS AUTOMATED SERVICE TRADING.
In *Proceedings of the International Conference on e-Business*, pages 38-45
DOI: 10.5220/0001428000380045

to develop a conceptual framework that can be used to define, compare and analyse the interoperability of different service trading architectures.

Our work advances the state of the art in service trading in the following. First, we clearly identify the roles that are necessary to carry out this service trading as well as the relationships between them. Second, we motivate and introduce the trading protocols, which are a specification of the global behaviour of the trading system from a temporal point of view. And third, we define the elements that are necessary for an automated decision-making in the service trading process. This bridges the two key elements of a real-time enterprise mentioned before: the management and analysis of the information collected by the enterprises as the basis of the decision-making, and the ability to use services offered by other enterprises through the automated creation of SLAs.

The structure of this article is the following. Next, in Section 2, we briefly introduce the organisational metaphor and we describe the conceptual framework. Section 3 compares our framework with other proposals developed by both the industry and the academy, and we conclude and enumerate future work in Section 4.

2 CONCEPTUAL FRAMEWORK

Several phases have been identified on the contracting process (Ludwig, 2003) (as it is shown in the left part of Figure 1): The first step that appears in the figure is outside the contracting process and it has been called preparation phase. This step involves the creation of the offer by the provider of the service and the analysis of its functional and non-functional requirements by the consumer. In the contracting process itself, first step is defined as information phase whose goal is to match service providers with potential consumers and vice versa. In the next phase, they may start a negotiation with those consumers or providers to find a mutually acceptable agreement. At the end of this phase the result is the creation of an agreement on the execution of a service between a provider and a consumer. In the fourth phase, both service provider and service consumer set up a deployment plan to make it possible to follow all terms established in the agreement settled in the previous phase. The last phase in the contracting process is the fulfilment phase. This phase involves the fulfilment of the obligations established in the agreement and in the monitoring of the whole process in order to ensure that both parties observe the agreement correctly.

The conceptual framework proposed in this article is focused on the information and negotiation phases. Therefore, on the one hand, input to the framework

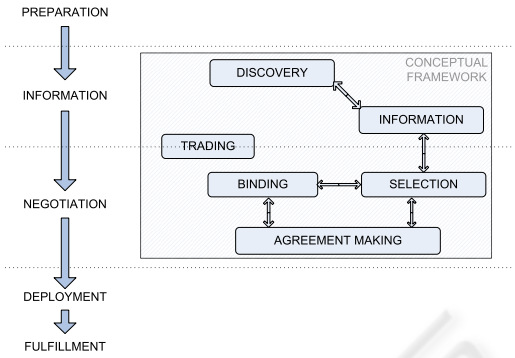

Figure 1: Conceptual framework.

consists in the requirements gathered during preparation phase. We call *user preferences* to this set of information developed in the preparation phase; additionally, it is worth pointing out that our idea of *user* is independent of the nature of the stakeholder, i.e. whether user stands for a service consumer or a service provider. On the other hand, the output of the framework is a established SLA. This SLA can be used to drive the further deployment and fullfilment phases.

In order to deal with the complexity of the service trading problem, we use the organisational metaphor (Zambonelli et al., 2003) where organisations are outlined developing a general architectural characterization of the problem. In so doing, our model is composed of six organisations (depicted in the central-right part of Figure 1) that interact amongst each other.

As an introduction to the conceptual framework, we can sketch the global behaviour of organisations as following: The discovery organisation performs the process of locating a set of potential providers or consumers according to a number of functional and non functional requirements; candidates discovered, are then passed to the information organisation in order to gather detailed information about the characteristics and preferences of each potential party. This information is subsequently used by the selection organisation to create and select a set of promising agreement proposals with other parties. Proposal selected are also analysed to decide whether we would start a negotiation process with other parties or produce a take-it-or-leave-it offer. This instructions are delegated to the agreement making organisation responsible to actually negotiate or propose the agreement to other party. During this procedure, agreement making organisation interact with the binding organisation by asking for approval to make or accept binding offers. In so doing, the responsiblity of the binding organisa-

tion can be seen as to determine when an offer may be accepted.

Finally, the main goal of the trading organisation is to specify a choreography that will regulate how the whole process is carried out, that is, it cares of starting the search for parties, submitting offers, waiting for responses, starting negotiations or sending binding offers. Additionally, trading organisation monitorise the market (making use of the discovery organisation) in order to decide when the agreement search should be started.

The remainder of this section describes each organisation detailing their goals and the roles that can be identify on them.

2.1 Trading

In general, service trading is a process whose details change from scenario to scenario depending on the type of parties involved and the temporal requirements to be met. In order to deal with these issues, it is necessary an orchestration of the different stages in the trading process.

The trading organisation focuses on the global behaviour from a temporal point of view. Its goal is the coordination of the remain layers so as to implement a trading protocol.

To understand what a Trading Protocol is, we use a simple real-life example: A public bidding where an institution looks for a service provider and devises a trading protocol that consists of the following stages: the announce of the bidding, a deadline for the submission of proposals, a period of resolution and, finally, the communication of results. The trading protocol also states temporal constraints for each stages.

Building on these ideas, the trading organisation should address the following issues: (i) A taxonomy of Trading Protocols, (ii) a repository with implementations for each Trading Protocols, (iii) the management of the life-cycle of the elements in the system, including a mechanism for the instantiation of the actors that implement the trading protocol, and (iv) a definition of the temporal parameters that will control the behaviour of other organisational. These parameters would be passed to the actors of those organisations.

This organisation is composed of the following roles:

- The *Trading Protocol Selector* role analyses user preferences to decide which of the known trading protocols best suits to the temporal contraints especified on the preferences.
- The *Customer Manager* role is in charge of interacting with the environment in order to handle user preferences. Consequently, this role transmit the appropriate part of these preferences to the rest of the roles. Additionally, this role is the responsible for trigger the search based on market status and knowledge harvested from pervious searches. Once a given search is about to start, this role is asisted by the *Trading Protocol Selector* to invoke the approporiate actors that would develop the trading protocol used on the search.

2.2 Discovery

In the discovery organisation the main aim consist in locating potential parties demanding (or supplying) a service that other party provides (or needs). In order to deal with the issues related with discovery, we can identify the following requirements:

- An infrastructure for a taxonomy of services. Each demand or provision of service should be catalogued based on the taxonomy; the classifying criteria may include both functional and non-functional features.
- A method for registering new trading events. An example of such an event would be the fact that an entity is searching for service providers of a given type (as specified in the taxonomy).
- A way to store access points to the different actors that generate events.
- A method for subscribing to events. This is necessary to feed the knowledge base used to decide when an agreement search must be started or the results of a previous search must be refined. This information might also be useful for some negotiation strategies.
- A protocol to exchange and propagate events amongst organisations.
- An addressing specification to provide a mechanism to access and identify actors.

Roles in this organisation include:

- The *Discovery Service* role represents an abstraction of the discovery infrastructure that should be refined in further concrete models. Different infrastructures can be selected from a wide range of models: from a centralized paradigm to a distributed one.
- The *Market Mediator* role is in charge of adapting local knowledge model in a given party to the appropriate discovery infrastructure. This adaptation make independent the characteristics of market modelized by the discovery service to the rest of organisations.
- The *Tracker* role is the active part that make use of the discovery service to search for some particular trading event.

- The *Advertiser* role is the complementary role for *Tracker* generating trading events to be searched. This trading events, should be parametrized with the most significative user preferences that can be made public.

2.3 Information

The goal of this organisation is to manage the public information about the user preferences and the potential candidates found by the discovery organisation. The amount and type of information collected from each candidate may be different; however, at a conceptual level the information should include, at least, the public features about the service demanded/supplied. In addition, some information can be harvested from external sources, e. g., information about the reputation of the candidate.

Roles in this organisation include:

- The *Inquirer* role polls the different *Informant*s located by the Discovery organisations. In this process, this role can select different strategies of quering, depending on the intraction standard and the type of information needed to match user preferences.
- The *Informant* role is the responsible for publish all the public user preferences that can be usefull to other parties in order to evaluate the possibilities to become a business partner.

Both *Inquirer* and *Informant* must implement a compatible specification of a format to express functional and non-functional features of services and a procedure to query and to inspect services. In addition, the *Inquirer* must provide an integration of the service features format with the taxonomy of the discovery layer.

2.4 Selection

The aim of the selection organisation is to choose a set of candidate parties with whom a negotiation process can be started or to whom an agreement proposal can be submitted. The selection starts with a set of information about potential parties coming from several sources: information provided by the information organisation after an active search, agreement proposals received from other parties, and non-successful offers [2] coming from the binding organisation, so that they can be processed again.

The selection process is carried out by the following roles:

- The *Proposal builder* role creates agreement proposals based on the information gathered by the information organisation. Then it sends these agreements proposals to the *Proposal collector*.
- The *Proposal collector* receives the agreement proposals generated by the *Proposal builder* as well as the agreement proposals coming from other parties through the *Proponent* role and submits them to the *Proposal filter*. Optionally, it can kept them until an event occurs. For instance, they can be kept until the negotiation phase finishes.
- The *Proponent* role represents the party that is submitting us a proposal.
- The *Proposal filter* role is in charge of filtering the agreement proposals collected by the *Proposal collector* and the non-successful offers coming from the binding organisation. The filter criteria are not unique but, in most cases, they depend on the preferences given by the user and the status of the whole service trading process. After this process, several proposals are rejected and the others are sent to the *Proposal dispatcher* role.
- The *Proposal dispatcher* sends the proposal to the most appropriate *Agreement Maker*. One system may have several *Agreement Makers* with different characteristics and one of them may be better than the others for certain conditions. For instance one *Agreement Maker* can implement auction protocols, another one can implement bilateral negotiation protocols, and another one can implement just a take-it-or-leave-it protocol.

2.5 Agreement Making

The goal of the agreement making organisation is to provide a mechanism to create agreements, possibly through an automated negotiation process, that are acceptable to all the parties involved in them. Therefore, the result of this organisation is an agreement that specifies the terms under which the service shall be executed. This may include both functional and non-functional terms.

Consequently, the requirements of the agreement making organisation are: it must support an agreement format that must be understood by both parties and that allows them to identify the terms of the agreement; it must implement at least one protocol to create agreements and, optionally, to negotiate them; it must provide decision making mechanisms to evaluate the offers received and generate their own bids or counteroffers if necessary, and it must offer a way to create reliable and non-repudiable agreements.

The protocols to create agreements can range from a very simple form of communication such as the submission of an offer by one party and its acceptance o

[2] These offers are non-successful because either they were not good enough for us or the other party rejected them.

rejection by the other one, to a more complex form based on negotiation protocols. A negotiation protocol establishes the rules that govern the negotiation and the way the communication amongst the parties involved in the negotiation is carried out. The most common negotiation protocols are based on the submission of offers and can be categorised into auctions (Ströbel and Weinhardt, 2003) (e.g. English, Dutch or Vickrey) and bilateral negotiations. Bilateral negotiations involve the exchanging offers and counteroffers between the two parties carying out the negotiation (Sierra et al., 1997).

The decision-making mechanisms determine the way the parties involved in the negotiation process behave. There are four parts that sould be implemented in these decision-making mechanisms: (i) an offer evaluation, usually carried out through the definition of utility functions to each term of the agreement; (ii) a decision on which response shall be sent to the counterparty; (iii) a construction of a counteroffer if necessary (Faratin et al., 1998; Faratin et al., 2002; Karp et al., 2004), and (iv) a model of the world and of our opponents in order to improve our negotiation capabilities (Zeng and Sycara, 1998).

We have identified three roles in this organisation:

- The *Agreement Maker* is the role that implements our agreement creation mechanism. Therefore, it must understand an agreement format and support at least one protocol to create agreements. This role can act almost as a proxy if it just implements a take-it-or-leave-it agreement creation protocol. However, it can be very complex if it understand several negotiation protocols and has to create bids or counteroffers. There is no restriction on the number of *Counter-parties* that the *Agreement Maker* can be negotiating simultaneously.
- The *Counter-party* role represents the party that we are trying to reach an agreement with. This role must implement the same communication protocol and agreement format as the *Agreement Maker*.
- The *Notary* role must guarantee that the agreement created between the two parties is reliable and non-repudiable.

2.6 Binding

The goal of the binding organisation is to determine when a binding offer must be submitted and whether a binding offer that has been received should be accepted. In addition, this organisation must establish when these decisions are going to be made. For example, one option is to make the decision as the offers are received; another possibility is to make the decisions at some points in time that has been previously set. These points may be dynamically selected, depending on changing conditions of the environment such as the frequency of arrival of offers, or statically determined based on temporal constraints imposed by the *trading protocol*, or a combination of them both. Therefore, the responsibilities of this organisation are not only to determine whether a binding offer must be accepted or submitted but to establish when these decisions shall be made as well.

The decisions made in this organisations are based on several factors and they may vary depending on whether it is a service consumer who is making the decision or it is a service provider. Nevertheless, we can divide these factors in three groups: First, User preferences about the contents of the agreement. For instance, constraints about the values of the terms of the agreement or an utility function indicating the importance of these terms to the user. Second, user preferences about the agreement process. Some examples are the deadline and the eagerness to reach an agreement. Third, external factors that may prevent a party to commit to an agreement. For instance, the provider's capability to accept new agreements or the existence of dependencies amongst the agreements a service consumer wants to reach.

This organisation is composed of three roles:

- The *Commit Handler* role has the final decision on whether to bind to an offer or not and it is also in charge of determining when these decision are made. To make these decisions it takes into account the user preferences about the contents of the agreement and the agreement process and it consults other roles about the feasibility of committing to an agreement.
- The *Capacity Planner* role analyses the provider's capability to provision a certain agreement and recommends the *Commit Handler* to commit or not to that agreement. This role is specific to the concrete deployment of the service provider.
- The *Demand Planner* is the role that plans the search of multiple agreements. For instance, a service consumer may want to reach agreements on different services, and there may exist dependencies between those services. These dependencies can be like "*I want either an agreement on all different services or no agreement at all*". The *Demand Planner* is in charge of enforcing these restrictions.

The most important decisions of the whole system take place in this organisation because it is where it is decided when to commit to an agreement. Therefore, in a real-time enterprise, the roles in this organisation are likely to be very complex and supported by additional components that manage and analyse the information collected by companies using methods such us data mining, forecasting or stochastic modelling.

3 RELATED WORK

In this section, we analyse related proposals from two points of view: First, we focus on different conceptual architectures or frameworks that address the interoperability of web services. Second, we give an overview of current technologies that can be used to develop different parts of the organisations identified in our framework.

Since web services and service-oriented computing paradigm were introduced, some architectural research has been developed. The comparative analysis carried out amongst them, focuses on automating the service trading; particularly, in Table 1 we study, for each of the proposals, if the organisational goals indentified in our conceptual framework are addresed.

Web Service Architecture (WSA)(W3C WSAG, 2004) is the reference architecture built by W3C. Due to the abstraction level of this conceptual architecture, SLAs automation creation is marginaly dealt and the only issues directly handled are discovery ones. However, in the last couple of years, an extended web service architecture (Weerawarana et al., 2005) have addressed Information problem in terms of standards for metadata interchange.

In the area of integration and virtual organisations, an evolved grid paradigm has emerged in the last years: the service grid. There is a wide range of on-going standarising work in this context. As part of this work, a conceptual architecture has been developed: the Open Grid Service Architecture (OGSA)(Globus Alliance, 2005). This approach tries to address a highly distributed scenario of collaborative stakeholders. Concerning our organisational goals, OGSA deals with all organisations that involves some kind of interaction (Discovery, Information and Agreement Making) in an explict way. Nevertheless, organisations centered in decision-making mechanisms (Selection and Binding) are not well defined and the needed elements have not been identified. However, in OGSA there are some references to the capacity planning issues.

The Semantic Web has influenced several research fields; particularly, semantic approaches have boosted several open research efforts in the web services field. One of the most active is the Web Service Modelling Ontology (WSMO) that comprises a group of especifications and systems for dealing with semantic web services. In particular, there is a conceptual architecture called WSMO-Full that describes the abstract background of WSMO. In this approach, interactions related to the information organisation are not clearly isolated and no further architectural element is outlined. However, there are some subtle references to additional information needs after a discovery phase. WSMO-Full(Preist, 2004) is more centered on decision making than other architectures and they explicitly propose a selection after the discovery of potential candidates. Nevertheless, it does not identify the issues related to the decision-making in the binding organisation. WSMO-Full supports the creation of agreements and defines contract agreement choreographies that are protocols for message interactions between at least one service requestor and at least one service provider. However, unlike our trading protocols they do not cover all phases of service trading but only the creation of agreements, and they are more focused on the messages exchanged rather than the temporal behaviour of the system.

Following the idea of semantic web services, a joint effort of several research groups in the area have developed a more general and abstract conceptual framework called Semantic Web Service Architecture (SWSA)(Burstein et al., 2005). SWSA completely covers discovery and agreement making. However, information issues are not addressed in a explicit architectural concept, although some higly related requirements are defined. Additionally, some matchmaking mechanisms are stated as part of their discovery phase. Nonetheless, this matchmaking is only about the advertised information of the service, while in our proposal, there is an additional selection carried out by the selection organisation that chooses amongst concrete agreement proposals instead of just advertised information.

We have analysed how different approaches differ based on the organisational outlining of our framework. Additionally, another significative difference amongst frameworks is the way they integrate the behaviour of service consumer and service provider into the overall architecture: on the one hand, WSMO-Full is aligned with our approach and the elements of its architecture are independent of the nature of the stakeholders, i.e. whether the are service consumers or service providers. On the other hand, SWSA strongly links the behaviour of active roles to the service consumer side while the more passive roles correspond to the service provider.

Several standards have emerged to enrich the basic web service stack. Table 2 shows a distribution of stadards over the conceptual organisations identified.

Concerning the discovery organisation, there are three specifications that can be used to implement its requirements: (i) UDDI can be used as a flexible repository that can be used to store the access points of elements and the taxonomies used by the discovery organisation. (ii) WS-Notification (Graham et al., 2005) can be used to subscribe and broker notification events. (iii) Lastly, WS-Adressing (Weerawarana et al., 2005) provides an specification of the references/locations of web services by means of a standardization of the concept of *endpoint references*.

There are a number of standards that deal with the exchange of service descriptions, from

Table 1: Comparative analysis of conceptual frameworks.

Phases Architectures	Discovery	Information	Selection	Agreement Making	Binding	Trading
WSA (W3C WSAG, 2004)	+	∼				
OGSA (Globus Alliance, 2005)	+	+		+	∼	
WSMO-Full (Preist, 2004)	+	∼	+	+		
SWSA (Burstein et al., 2005)	+	∼		+		
Ours	+	+	+	+	+	+

Table 2: Related standards.

Discovery	Information	Agreement Making	Trading
UDDI	WS-MetadataExchange	WS-Agreement	WS-CDL
WS-Notification	WS-InspectionLanguage	FIPA Protocols	BPEL
WS-Addressing	WS-Agreement		

both a functional and a non-functional point of view and they can be used in the implementation of the information organisation. For instance, WS-MetadataExchange (Weerawarana et al., 2005) and WS-InspectionLanguage. Alternatively, WS-Agreement (Andrieux et al., 2004a) uses a template-driven procedure, and those templates can be seen as a mean of expressing the preferences of a given party.

The most significant specification that covers most aspects included in the agreement making organisation is WS-Agreement (Andrieux et al., 2004a). It allows to specify the structure of an agreement document, so that it must be used together with one or several domain-specific vocabularies to give the proper semantic to the terms of the agreement. Furthermore, it defines a protocol and a web service-based interface to create, represent and allow the monitoring of agreements.

However, WS-Agreement just defines a take-it-or-leave-it protocol. To use more complex negotiation protocols, other specifications must be implemented. For instance, WS-AgreementNegotiation[3] (Andrieux et al., 2004b), which builds on WS-Agreement and specifies a bilateral negotiation protocol, or the negotiation protocols defined by FIPA (FIPA,).

Concerning the trading organisation, depending on the complexity of the trading protocol used, different approaches are possible. For complex coordinations, there are workflow standard such as BPEL (Weerawarana et al., 2005) or choreography languages such as WS-CDL(Group, 2003). In the case of simple cases, an alternative to implement Trading Protocols would be the specification of ad-hoc elements in the concrete architecture build upon the conceptual framework.

[3]Still in a very early stage of development

4 CONCLUSIONS AND FUTURE WORK

This paper focuses on the problem of service trading. In this context, our main aim is to achieve an effective background for the development of automated discovering, selection and negotiation of SLAs. To this end, a conceptual framework is developed and compared with related conceptual approaches.

The main contributions of this article are:

- A decomposition of the automated service trading problem outlining a set of abstract roles and organisations. Unlike other proposals, which are centered in the interactions between parties, we also identify the neccesary elements for the automated decision making.
- The conceptual framework presented aims to define and compare different trading architectures. Furthermore, the conceptual background developed can be used to analyse potential interoperability amongst architectures.
- We introduce the concept of Trading Protocol as a method for defining the temporal features and behavioral stages of trading scenarios. These protocols drive the choreography of the different elements and allow a temporal match procedure among SLA demands/offers of stakeholders.

Additionally, it is worth pointing out that an automation of the service trading process, shall benefit not only a cross-organisational scenario but also an intra-organisational (integration) one: SLAs have been associated traditionally with cross-organisational transactions where a company must enforce a certain level of service to their partners; however, real-time enterprises paradigm, in most cases, is

built upon the integration of a complex organisation; in this context SLAs are starting to be an important issue to be addressed amongst the subsystems involved. In this integration scenario, the rationalisation of the usage of resources inside the organisation argues for SLAs to be managed as automatically as possible. In so doing, new promising fields such as the Service Grid (Globus Alliance, 2005) are aligned with an hybrid scenario cross/intra-organisational where SLAs are comparatively important.

Further work to be done in this field is to develop a reference architecture making use of the conceptual framework presented. This architecture has been outlined in (Resinas et al., pear) although some refinement and implementation must still be developed.

REFERENCES

Andrieux, A., Czajkowski, K., Dan, A., Keahey, K., Ludwig, H., Pruyne, J., Rofrano, J., Tuecke, S., and Xu, M. (2004a). WS-Agreement specification.

Andrieux, A., Czajkowski, K., Dan, A., Keahey, K., Ludwig, H., Pruyne, J., Rofrano, J., Tuecke, S., and Xu, M. (2004b). WS-AgreementNegotiation specification, draft.

Burstein, M., Bussler, C., Zaremba, M., Finin, T., Huhns, M. N., Paolucci, M., Sheth, A. P., and Williams, S. (2005). A semantic web services architecture. *IEEE Internet Computing*, 9(5):72–81.

Curbera, F., Khalaf, R., Mukhi, N., Tai, S., and Weerawarana, S. (2003). The next step in web services. *Commun. ACM*, 46(10):29–34.

Faratin, P., Sierra, C., and Jennings, N. R. (1998). Negotiation decision functions for autonomous agents. *Int. Journal of Robotics and Autonomous Systems*, 24(3-4):159–182.

Faratin, P., Sierra, C., and Jennings, N. R. (2002). Using similarity criteria to make trade-offs in automated negotiations. *Artificial Intelligence*, 142:205–237.

FIPA. FIPA Contract Net Interaction Protocol Specification. `http://www.fipa.org/specs/pesspecs.tar.gz`.

Globus Alliance, . (2005). Open grid service architecture. `http://www.globus.org/ogsa/`.

Graham, S. G., Hull, D., and Murray, B. (2005). WS-BaseNotification Specification. `http://docs.oasis-open.org/wsn/wsn-ws_base_notification-1.3-spec-pr-01.pdf`.

Group, X. P. W. (2003). Soap specification. `http://www.w3.org/TR/soap12-part0/`.

Karp, A. H., Wu, R., Chen, K.-Y., and Zhang, A. (2004). A game tree strategy for automated negotiation. In *ACM Conference On Electronic Commerce*, pages 228–229.

Ludwig, H. (2003). A conceptual framework for building e-contracting infraestructure. In Corchuelo, R., Wrembel, R., and Ruiz-Cortes, A., editors, *Technologies Supporting Business Solutions*, chapter 1. Nova Publishing.

Ludwig, H., Gimple, H., Dan, A., and Kearney, R. D. (2005). Template-based automated service provisioning-supporting the agreement-driven service life-cycle. Technical Report rc23660, IBM Research Division.

Preist, C. (2004). A conceptual architecture for semantic web services. In McIlraith, S. A., Plexousakis, D., and van Harmelen, F., editors, *International Semantic Web Conference*, volume 3298 of *Lecture Notes in Computer Science*, pages 395–409. Springer.

Resinas, M., Fernandez, P., and Corchuelo, R. (2006 (to appear)). Automatic creation of agreements in a service-oriented scenario. In Columbus, F., editor, *Progress in Computer Network Research*. Nova Science Publishers.

Sierra, C., Faratin, P., and Jennings, N. R. (1997). A service-oriented negotiation model between autonomous agents. In *Proc. of the 8th European Workshop On Modelling Auton. Agents in a Multi-Agent World*, pages 17–35. Springer-Verlag.

Ströbel, M. and Weinhardt, C. (2003). Montreal taxonomy for electronic negotiation, the. *Group Decision and Negotiation*, 12(2):143–164.

W3C WSAG, . (2004). Web services architecture. `http://www.w3.org/TR/ws-arch/`.

Weerawarana, S., Curbera, F., Leymann, F., Storey, T., and F.Ferguson, D. (2005). *Web Services Platform Architecture*. Prentice Hall.

Zambonelli, F., Jennings, N. R., and Wooldridge, M. (2003). Developing multiagent systems: The gaia methodology. *ACM Trans. Softw. Eng. Methodol.*, 12(3):317–370.

Zeng, D. and Sycara, K. (1998). Bayesian learning in negotiation. *Int. J. Hum.-Comput. Stud.*, 48(1):125–141.