Coping with global Information Systems

Requirements for a flexible SLA Discovery and Negotiation Infrastructure for the future Internet of Services

Sebastian Hudert, Torsten Eymann

Chair of Information Systems Management (BWL VII), University of Bayreuth

1 Introduction

Visions of 21st century's information systems show highly specialized digital services and resources, which interact continuously and with a global reach. Today's internet of mainly human interaction evolves to a global, socio-technical information infrastructure, where humans as well as software agents acting on their behalf continuously interact to exchange data and computational resources. Such infrastructures will possibly consist of millions of service providers (SP), consumers (SC) and a multitude of possible intermediaries like brokers, workflow orchestrators and others, thus forming a global economic environment. Electronic services and resources traded on a global scope will eventually realize the vision of an open and global Internet of Services (IoS) (Schroth and Janner, 2007, pp. 36-41).

For a broad adoption of this vision in a commercial context it is important to have a mechanism in place to guarantee quality of service (QoS) for each service invocation, even across enterprise boundaries. This becomes crucial when parts of mission critical workflows will be executed on external services. A very simple use case would be an engineering company purchasing basic storage and computation services from the Amazon Web Services (AWS)[1] as well as specific fluid simulation services from a specialized application service provider. Since such scenarios inherently lack the applicability of centralized QoS management, guarantees must be obtained in the form of bi- or even multi-lateral service level agreements (SLAs) assuring service quality across individual sites (Ludwig et al., 2003, pp. 43-59).

Representing qualitative guarantees placed on services, SCs can benefit from SLAs because they make nonfunctional properties of services predictable and subsequently the corresponding services dependable as needed in a business context. In order to support a comprehensive SLA-based management for such settings,

[1] http://aws.amazon.com/

the main phases of the SLA lifecycle should be directly supported by the underlying infrastructure: service discovery, negotiation, implementation, monitoring and post processing (such as rating the service or payment), which, as a set, form a fundamental basis of a comprehensive service management and governance overlay as needed in the envisioned settings.

The contribution of this paper is twofold. In section 2 we derive a detailed scenario model for the future IoS as we envision it as a first step to understand the challenges of next generation information systems (IS). This model will be deduced from current developments in distributed computing. Secondly, we conduct a comprehensive requirement analysis for technical infrastructures supporting the derived IoS vision in section 3. Because of their high importance in the overall lifecycle and due to space reasons, we will focus this analysis on the negotiation phase and partly on the discovery phase as far as needed for a flexible service negotiation (i. e. protocol.-generic service negotiations across market boundaries). Section 4 will finally give a short overview on current research projects and show which of the identified requirements are currently addressed and to what extent they are fulfilled up to now. We conclude our paper with a short section on gaps in the current research landscape with regard to the identified requirements and thus motivation for future research work in this area.

2 From SOAs, Grids and Clouds to the Internet of Services

In order to derive a model for the future IoS it is crucial to understand the paradigms that are currently employed in distributed IS and identify similarities and differences. Based on these developments a trend of ideas and paradigms can be extracted leading to a well-founded scenario definition for the IoS vision.

The technical paradigm underlying most of the developments in distributed IS in the recent years is the Service Orientation (SO). The main idea of SO is that every function offered by humans, organizational entities or computer systems, is viewed as an abstract service, which in turn can again be combined with other services to create more complex composite services. Very common basic services are for example storage or simple computing services as provided by AWS as opposed to very complex services such as a fluid simulation application.

Before going into more detail a set of related key concepts have to be defined and distinguished from each other, namely Service Oriented Computing (SOC), Service Oriented Architectures (SOA) and Service Oriented Systems (SOS). In order to distinguish these different perspectives on the very same vision a well-known concept in information systems research has to be presented: the distinction between the task layer and the actor layer (Ferstl and Sinz, 2006, pp. 52-60). The task layer comprises all abstract tasks and their combination to processes, whereas the actor layer contains all human or automated operators present in a given enterprise, which on their part can execute tasks assigned to them.

Applied to the SO realm a SOA therefore defines all services, and therefore abstract functionalities of a given system (task layer). Just as with traditional enterprise IS the operators providing the individual services can be both human and electronic services. The design paradigm concerned with the definition of such electronic services is called SOC. Finally the actual implementation of the automated services of a SOA is called SOS (actor layer).

According to the general agreement in the literature a service as applied in SOS exhibits the following characteristics (Foster, 2005, p. 814):

- A service is an individually addressable software component that provides some functionality to a service requester.
- Services are accessed over an electronic network, such as the Internet.
- As already said, individual services can be composed to higher-level complex services resulting in possible multiple levels of service complexity. Regularly this is achieved by combining several services to a (business) workflow.
- Services only advertise details, such as their capabilities, interfaces or accepted protocols that are needed to interact with them. Technical implementation details of the service are hidden from the SC.
- Regarding their interaction electronic services are loosely coupled. This means that their interactions are not hardcoded in each individual service, but every SC discovers and binds a given other service it interacts with at runtime.

Among others the main advantage (and thus reason for its massive adoption in research and industry) of SOS is that they are more flexible than traditional systems, because of the possibility of dynamic discovery and binding of service instances at runtime, also allowing such systems to reuse individual services in different business processes. Additionally SOS allow for the dynamic instantiation and removal of service instances to cope with load fluctuations. The new service instances just have to be registered to the discovery system and can immediately be invoked, thus realizing very powerful load balancing strategies.

In parallel to these mostly intra-organizationally used systems a paradigm for distributed coordination of electronic resources crossing organizational boundaries emerged: Grid Computing (GC). In the following we will elaborate why this concept marks an important step towards the IoS vision.

Some of the most recognized Grid researchers defined GC to be mainly concerned with "coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations" (Foster et al., 2001, p. 2).

These definitions already show the gist of what GC is all about:

- Grids are first and foremost systems that "coordinate [...] resources that are not subject to centralized control" (Foster, 2002, p. 2, ranging from computational, storage and network resources to code repositories (Foster et al., 2001, p. 7).

- In doing so Grids employ "standard, open, general-purpose protocols and interfaces" (Foster, 2002, p. 2) supporting the sharing process over the ubiquitous Internet.
- The final overall goal of GC is "to deliver nontrivial qualities of service", following the vision of the "utility of the combined system [being] significantly greater than that of the sum of its parts" (Foster, 2002, p. 3).

Building on GC and SO technologies, a powerful abstraction concept in distributed computing called Virtualization paved the way for the next big step towards the IoS. "Virtualization enables consistent resource access across multiple heterogeneous platforms [... and] also [the] mapping of multiple logical resource instances onto the same physical resource..." (Foster et al., 2002a, p. 40). Hence, virtualization does not only hide the actual implementation details of a service, but also allows for complete decoupling of the services from the actual physical resources they are deployed on. Based on these concepts the most recent trend in the ever-ongoing development in distributed IS was realized: Cloud Computing (CC).

Buyya defines CC as: "[...] a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers […] presented as one or more unified computing resources based on service level agreements established through negotiation between the service provider and consumers" (Buyya et al., 2009, p. 601). Following this rationale Liu and Orban also stress the fact that these services are remotely consumed on-demand (Liu and Orban, 2008, pp. 295-305).

CC therefore exposes the following characteristics:

- Clouds heavily build on virtualization technologies thus hiding the service implementation details and even the employed physical resources from the end user
- Cloud services are mostly offered by independent and external service providers
- Cloud resources are dynamically provisioned on demand
- Each CC offering is possibly based on pre-negotiated SLAs; therefore the main goal of each Cloud service is to meet certain QoS levels

Obviously GC and CC are very much related in terms of the employed technologies and goals. The main difference however is probably that clouds manage and schedule their resources centralized (e.g. at commercial data centers as with the AWS platform or IBM's CC offering[2]) as opposed to grids which offer access to decentralized resources with local policies (Begin, 2008, pp. 24-26). Apart from those conceptual differences both GC and CC combine a set of resources to execute demanding compute jobs or high storage needs.

[2] http://www.ibm.com/ibm/cloud/

All of the abovementioned developments in today's distributed IS point to the same vision of the future internet based on highly dynamic networks of composable services, offered and consumed on a global scope, ultimately leading to innovative business models and supporting the transition from value chains to value nets.

On a technical side it can easily be seen that all the currently applied concepts build on very similar infrastructure technologies. The majority of services employ Web Service standards for interface definition and communication and the Internet as a communication platform. On the other hand they differ slightly in the way the individual services are managed and used on a higher abstraction level. This is especially noticeable when looking at the applied invocation paradigms, the point of control within each invocation, the overall system configuration and service complexity as well as the scope of the employed systems. However, not building on different infrastructures but only using them differently should not prevent a development of consolidating and integrating of paradigms as a next step towards more powerful and efficient global systems. This is what many experts in research and industry envision as the next step in IS and also in business, the IoS.

This vision takes the main idea of SOC, the design of IS in terms of interoperable and composable services, one step further. It rigorously focuses on the goal of an Internet-based service economy similar to the real-world service sector. Digital services, but also services adduced by humans will be offered over digital service markets, purchased by respective customers and then combined with internal or other external services to business workflows of varying complexity. This way open and global business value networks can be implemented.

Such systems allow even small and medium enterprises to offer their potentially very specific services on a global scope and thus participate in the digital value chains. By concentrating on a company's core competence and by purchasing less critical services over the IoS new business models can emerge, following the rationale of business outsourcing and consolidation. This could ultimately lead to a higher rate of innovation in the digital service sector.

As can easily be seen this vision still builds on the aforementioned paradigms of GC and CC. The IoS explicitly focuses on the orchestration of a number of services from outside and inside a company to achieve higher utilities than the individual services would, just as GC. As proposed with the CC paradigm services will be invoked on demand and therefore have to be deployed on virtualized platforms to satisfy the QoS restrictions posed by the customers. In addition to the combination of these concepts the IoS focuses much more on new business models and the commercial application of the SO ideas. These will concern trading processes down to the level of an individual and potentially very simple service, and the subsequent charging based solely on its usage and delivered QoS. In such a system even very small and specialized companies can find a niche in the digital economy where they can compete with the ubiquitous international companies, which in turn have to face a much higher competition on a global market, ultimately leading to increasing service quality. (Theseus 2009)

Summarizing the IoS scenario model results in the following set of characteristics (Schroth and Janner, 2007, pp. 36-41):

- The IoS focuses on a (potentially huge) set of electronic services
- The services vary in complexity and therefore range from raw resources to very complex workflows
- These services will be employed in potentially mission-critical business processes and thus have to fulfill a (pre-negotiated) set of QoS guarantees
- SPs and SCs communicate over the global Internet to trade these services
- New business models will cope with the possibility of trading even individual services and charging them based on their actual usage
- The IoS will consist of a global set of SPs, SCs negotiating over digital services employing some mediating nodes such as service brokers and market makers

Arguably, other scenarios for the future internet are also possible, for example ones in which only very basic services, and thus SLAs, are traded, just as with current CC platforms. However, an IoS scenario more or less similar to our model seems the most likely, given the current and past developments in distributed IS.

3 Requirements for a Service Management Overlay in the IoS

As can easily be deducted from the last subsection, two of the main obstacles of our IoS vision from a commercial perspective are reliability and efficiency of the services traded and the infrastructure supporting this service economy. Being some of the traditional problems in computer science these issues become even more crucial in the IoS due to its global and cross-organizational character. In such settings the need for guaranteed reliability and service quality becomes more prominent, as no longer the question of who provides the service matters but only whether he is able to achieve the requested result. Such negotiated service quality guarantees are to be stated in potentially very fine-grained SLAs between SC and SP, acting as a signed contract governing the subsequent service invocation ((Buyya et al., 2009, p. 601). Based on such an SLA the actual execution of the service can be monitored in order to assess the compliance to the contract, eventually triggering some corrective measures in case of SLA violation.

Additionally current business trends force companies to implement highly flexible supply chains, allowing for a very dynamic adoption of their internal processes and thus for the dynamic integration with external ones. Current infrastructures do not yet support such a flexible service discovery and subsequent QoS negotiation, as would be needed for a reliable integration into an internal workflow.

Having these two perspectives on the IoS in mind a need for a decentralized QoS management mechanism, able to cope with cross-organizational settings is emerging. Consequently we will now derive a set of conceptual requirements for a service management overlay (SMO) that aims at supporting the flexible discovery and negotiation of services for SLA-based IoS settings.

Requirements for the discovery phase:

- RD1 After the Discovery Phase all parties must have a common understanding of the protocol to be executed in the negotiation phase (Ludwig et al., 2006).
- RD2 This common understanding must be generated dynamically at runtime (Brandic et al., 2008, p. 4).
- RD3 In order to support global IoS settings a system of decentralized registries must be implemented offering service offer and demand information to potential transaction partners.
- RD4 These registries must be accessible over standard Internet protocols and must internally implement efficient data replication and update strategies.

It can easily be seen that RD3 and RD4 underline the need for a reliable discovery system on a global scope. It is reasonable to expect fully centralized registry systems insufficient and too much failure-prone for a global infrastructure as the IoS. On the other hand an open access over standard protocols is crucial for the emergence of a widely-used management layer for the actual service economy. Additionally most researchers predict the emergence of a number of individual service markets, together comprising the IoS economy. It is simply not reasonable and also not efficient to implement one particular market with only one particular protocol for all services traded in the IoS. On the other hand such a system of service markets demands discovery processes in which SCs and SPs need to discover the protocol used to negotiate a given service at runtime and adapt to it in order to take part in the particular market.

Requirements for the negotiation phase - Negotiation Object:

- RN11 Services (and thus SLAs) of different complexity must be negotiable (Neumann et al. 2008, pp. 325-347).
- RN12 It should be possible to state several possible service configurations in one offer (Ziegler et al., 2008, p. 11).
- RN13 Possible offers should be restrictable, incl. non-negotiable SLA terms (Ziegler et al., 2008, p. 11).

RN11 refers to the inherently varying complexity of services offered in an IoS setting. As can be derived from the scenario model above those services can range from simple resources to very complex business processes consequently also exposing very different sets of service attributes. The employed service and SLA description languages must be capable of representing this whole spectrum. In order to exploit the full potential of especially bargaining protocols it should be possible to state different acceptable SLA configurations within one offer, letting the opposing party chose one of them. This can heavily decrease the number of messages needed in a given negotiation. Finally the negotiators should be able to set some of the SLA attributes stated in an offer as fixed. This allows for the distinction of service properties that are under negotiation and those that aren't.

Negotiation Protocol / Setting:

- RN21 Different marketplaces and protocols even within one market are needed for different services to be traded (Neumann et al., 2008, pp. 325-347).
- RN22 Service requestors and consumers must be able to start the negotiation (Ziegler et al., 2008, p. 11).

Following the rationale from the sections above it is absolutely necessary that not only one particular protocol is used for all of the IoS but a set of different marketplaces offering different protocols will most likely emerge, each supporting a different negotiation setting or group of services. Additionally it is important that both the SC and SP can start a negotiation. This way not only a request for a service can be communicated but also a service offering can be actively proposed to potential SCs.

Negotiation Strategy / Participants:

- RN31 Software Agents should act as negotiators (Ludwig et al., 2006).
- RN32 Negotiators must be able to act on different markets, even simultaneously (Brandic et al., 2008).
- RN33 Intermediaries, such as auctioneers or brokers, should be present

It can easily be seen that complex and global settings like the IoS demand at least semi-automated service management. Thus software agents (Wooldridge, 1997, pp. 26-37) should be employed, especially for the discovery and negotiation phases as they particularly suite the software agent concept. As a result of the need for a number of different markets the agents have to be capable of taking part in more than one of them at a time. This way the full potential of the overall economy can be exploited. Finally a set of intermediaries will become necessary for management of the individual markets, registries etc.

Apart from these conceptual requirements infrastructures for the IoS of course also have to fulfill the same technical requirements as already applied for SOS, GC or CC, such as service orientation and thus a decentralized design based on messages and interfaces instead of internal components, scalability and robustness. Finally all developed mechanisms should build upon common standards for the definition of and communication between services as currently developed in standardization bodies, such as the Open Grid Forum (OGF)[3] or the Organization for the Advancement of Structured Information Standards (OASIS)[4].

4 Current Research Efforts towards flexible SLA Negotiations for the IoS

In this section we will give a short overview of related research projects also addressing the flexible negotiation of SLAs for the IoS. This will give an impression on the extent to which the abovementioned requirements are already fulfilled in current systems and also help identify gaps in the research landscape.

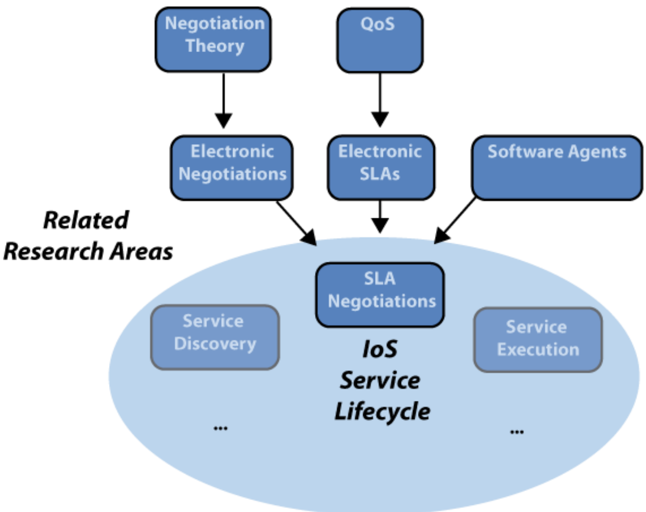

Figure 1: Related Research Fields

The commercial vision of the IoS heavily builds on Negotiation Theory and different negotiation protocols developed therein. Such negotiation protocols crucially define a negotiation's outcome by "determin[ing] the way offers and messages [...] are exchanged" (Bichler et al., 2003, pp. 311-335). The huge amount of different protocols developed up to date will form the conceptual basis of the service economy making up the IoS (it also underlines the central requirement RN21). As a next step these findings are ported to the digital world, forming the new research discipline on Electronic Negotiations (Bichler et al., 2003, pp. 311-335). On the one hand researchers came up with (semi-)formal description and characterization of given negotiation protocols, on the other hand increasing computing power allowed for the definition of new negotiation protocols, which would not work

[3] http://www.ogf.org/

[4] http://www.oasis-open.org/

efficiently with human negotiators, such as multi-attribute auctions. By doing so current approaches pay a lot of attention to RN21 and RN22 and also RN11, thus tackling a lot of the issues on the protocol side of negotiation infrastructures.

Additionally scientists constantly improved software agent technologies, finally allowing for the implementation of very complex bidding strategies in a fully automated fashion (Paurobally et al., 2007) as requested by RN31. International research projects such as CATNETS[5] or SORMA[6] bring the vision of agent-based service markets to life in Grid environments. ZIMORY[7] can be seen as an industrial implementation of a service market, though not really based on software agents. RN32 on the other hand is still not addressed appropriately as most agent implementations still only work with one protocol only.

QoS and its guarantees as stated in SLAs have risen after traditional distributed systems came to maturity and reliable, mission critical applications were executed on those infrastructures. Significant work was done in the area of SLA languages or architectures of SLA based systems by researchers such as Ludwig and Keller (2003) or Yarmolenko and Sakellariou (2007, pp. 1975-1990), addressing RN11, RN12 and RN13. An ever growing amount of research projects, such as SORMA or CoreGRID[8] already employ SLAs for resource management and thus have done the first step towards the IoS. However, they still lack a conceptual integration of the aforementioned negotiation protocols and service markets to finally implement a running service economy. Nevertheless the ongoing WS-Agreement (Andrieux et al., 2005) standardization effort at the OGF shows the growing interest for SLA-based infrastructures from both research and industry.

Surprisingly, there is little research done in combining the economic considerations concerning negotiations on the one hand (RN21) and QoS/SLA developments in SOC (RN11 and RN31) on the other hand. As the GC projects pioneered in combining digital resources on a large scale, mostly Grid projects stand out in terms of developed SLA negotiation mechanisms. However even those projects mainly focus on static and centralized architectures within which only one particular, and fixed, negotiation protocol is implemented, e. g. OntoGrid (Paurobally et al., 2007) using the (Iterated) ContractNet or NextGrid[5] the Discrete Offers Protocol). Hence those systems allow for the definition of service markets; however they lack possibilities to integrate those to a global service economy (RN21).

Although a common understanding states the need for flexible negotiations, only a few research efforts incorporate the mere possibility of different protocols in SLA negotiations. Ludwig et al. (2006) and Brandic et al. (2008) being two of the most prominent examples. However those frameworks still lack important flexibility by restricting the negotiation protocols to a small and fixed set (this is contradic-

[5] http://www.catnets.org
[6] http://sorma-project.org
[7] http://www.zimory.com/
[8] http://www.coregrid.net

tory to RD1 and RD2) and by building on static, centralized architectures without appropriate discovery mechanisms(as demanded by RD3 and RD4).

5 Conclusion and Future Research

In this paper we developed a very detailed scenario model of the future IoS as it is anticipated in the research community. Based on this scenario we derived a set of conceptual requirements for infrastructures supporting service discovery and negotiation in future IoS settings. In doing so, our paper provides a roadmap for future research towards a global IoS. Although a lot has been done in the past years it is still a long way towards global service economies. Especially the centralized service market architectures currently employed prevent an efficient global IoS. Future research work will have to address this issue and also develop appropriate discovery mechanisms able to cope with such scenarios.

References

Andrieux A, Czajkowski K, Dan A, Keahey K, Ludwig H, Nakata T, Pruyne J, Rofrano J, Tuecke S, Xu M (2005), Web Services Agreement Specification Draft, Version 09/2005

Begin M-E (2008), An EGEE comparative study: Grids and Clouds, evolution of revolution?, Technical Report, CERN

Bichler M, Kersten G, Strecker S (2003), Towards a Structured Design of Electronic Negotiations, Group Decision and Negotiation, Vol. 12

Brandic I, Venugopal S, Mattess M, Buyya R (2008), Towards a Meta-Negotiation Architecture for SLA-aware Grid Services, Technical Report, University of Melbourne

Buyya R, Yeo CS, Venugopal S, Broberg J, Brandic I (2009), Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility, Future Gen. Comp. Sys., Vol. 25

Ferstl OK, Sinz EJ (2006), Grundlagen der Wirtschaftsinformatik. Oldenburg Wissenschaftsverlag, München

Foster I (2002), What is the Grid? A Three Point Checklist, GRIDtoday, Vol. 1, No. 6

Foster I, Kesselman C, Tuecke S (2001), The Anatomy of the Grid: Enabling Scalable Virtual Organizations, IJSA, Vol. 15

Foster I, Kesselman C, Nick JM, Tuecke S (2002a), Grid Services for Distributed System Integration, IEEE Computer Society, Vol. 35

Foster I, (2005), Service-oriented Science, Science, Vol. 308, 814-817

Liu H, Orban D (2008), GridBatch: Cloud Computing for Large-Scale Data-Intensive Batch Applications, CCGrid08, 295-305

Ludwig H, Keller A, Dan A, King R, Franck R (2003), A Service Level Agreement Language for Dynamic Electronic Services, JECR, Vol. 3

Ludwig A, Braun P, Kowalczyk R, Franczyk B (2006), A Framework for Automated Negotiation of Service Level Agreements in Services Grids LNCS, Proc. of the Workshop on Web Service Choreography and Orchestration for Business Process Management, Vol. 3812/2006

Neumann D, Stoesser J, Weinhardt C, Nimis J (2008), A Framework for Commercial Grids - Economic and Technical Challenges, JOGC, Vol. 6, 325-347

Paurobally S, Tamma V, Wooldrdige M (2007), A Framework for Web service negotiation, ACM Trans. Auton. Adapt. Syst., ACM, Vol. 2, Nr. 14

Schroth C, Janner T (2007), Web 2.0 and SOA: Converging Concepts Enabling the Internet of Services, IT Professional, IEEE Computer Society, Vol. 9, 36-41

Theseus (2009), TEXO – Business Webs im Internet der Dienste, http://theseus-programm.de/anwendungsszenarien/texo/default.aspx, Abruf am 14. 09. 09

Ziegler W, Waeldrich O, Wieder P, Nakata T, Parkin M (2008), Considerations for negotiation and monitoring of Service Level Agreements, Research Report, CoreGRID Project

Wooldridge M (1997), Agent-based software engineering IEEE Proceedings Software Engineering, Vol. 144, 26-37

Yarmolenko V, Sakellariou R (2007), Towards increased expressiveness in service level agreements, Research Articles Concurr. Comput., Pract. Exper., John Wiley and Sons Ltd., Vol. 19