eChallenges e-2011 Conference Proceedings
Paul Cunningham and Miriam Cunningham (Eds)
IIMC International Information Management Corporation, 2011
ISBN: 978-1-905824-27-4

Closing the Gaps for e-Contracting in the Cloud

Bastian KOLLER[1], Peer HASSELMEYER[2], Philipp WIEDER[3]
[1]Höchstleistungsrechenzentrum Stuttgart, Universität Stuttgart,
Nobelstraße 19, 70771 Stuttgart, Germany
Tel: +49 711 68565891, Fax: +49 711 68565832, Email: koller@hlrs.de
[2]NEC Laboratories Europe, Kurfürstenanlage 36, 69115 Heidelberg, Germany
Tel: +49 6221 4342 250, Fax: +49 6221 4342 155, Email: Peer.Hasselmeyer@neclab.eu
[3]Service Computing Group/ITMC, TU Dortmund University,
August-Schmidt-Straße 12, 44227 Dortmund, Germany
Tel: +49 231 7552767, Fax: +49 231 7552763, Email: philipp.wieder@udo.edu

Abstract: Cloud computing is increasingly becoming a viable alternative for businesses large and small to outsource their service needs. Aspects of major importance in the business context are contracts between service providers and service consumers. Existing e-Contracting solutions mainly from the Grid domain can be partially re-used, but do not work well with the heterogeneity prevalent in the cloud. This paper introduces an architecture which allows the dynamic customization of service logic to adapt to various contexts and requirements. Customization is achieved via a plug-in like mechanism based on the dynamic integration of functionality "bricks". The architecture is applicable beyond the contracting sub-domain and is expected to improve the acceptance of the global service market place.

1. Introduction

e-Contracting, despite being a research topic for quite some time now, is increasingly creating interest among researches in the course of the on-going *cloud* evolution. Whilst significant results have already been created researching e-Contracting methods for *grids*, especially based on service-level agreements (SLAs) [1][2][3], a one-to-one transition of results towards cloud infrastructures is not possible. The reasons for this are manifold with the differences of business models and technology stacks being the most prominent. This has created a situation where on the one hand existing results are evaluated in the light of the new distributed computing paradigm, while on the other hand novel challenges related to e-Contracting in cloud-based ecosystems are currently targeted by researches.

We see that especially the flexibility of SLA handling during all SLA lifecycle phases needs to be increased to reflect the diversification of end-users with respect to their environments and expertise. With the growing diversity of cloud services and the penetration into everyday IT usage, different expertise levels have to be taken into account to maximize the user experience.

One example where flexibility is lacking is the integration of legal issues into electronic contracts. This is seen as a critical challenge to establish reliable and dynamic markets beyond the currently dominating "pay-as-you-go" model. With the hidden complexity of handling services in the cloud, legal parameters need to be taken into account automatically. A common scenario exemplifying the necessity of domain-specific legal support is the provisioning of data storage with location-based quality-of-service capabilities that has to adhere to various national and trans-national regulations and laws.

In this paper, we introduce one technological approach that helps to close some of these gaps leading to more generic e-Contracting solutions with domain-specific extension mechanisms.

2. Objectives

The objective of our contribution is to extend existing e-Contracting solutions, namely SLA Management frameworks, and to give an overview of shortcomings of current solutions, research fields which need to be urgently addressed, and to provide initial ideas on how these issues may be resolved. Even though technological bits are already available, the change from grids to clouds brings about several issues when trying to hide the complexity for the end user. Service-level agreements as underlying technology can be used; however, it is clear that their management will need to be supported with extended capabilities, taking into account the additional requirements which appear when working in the cloud.

These extended capabilities require new research topics which need to diverge from the classical research on SLAs yet must be based on existing solutions. The aim of this paper is to start from a real use case (using a pan-European cloud test-bed federated from individually managed sites) to see where the problematic areas with respect to e-Contracting are and what needs to be provided in terms of enhanced capabilities to provide a competitive cloud solution.

3. Methodology

The methodology of this paper is as follows. First we present and elaborate a use case for public clouds (cf. Section 4) and present their requirements towards (automated) e-Contracting. The main issues making federation of individual resource-offering sites hard are then introduced. A high level overview of the architecture for a system realizing dynamic context-adaptation is presented and the benefits of such a system for service providers and consumers are highlighted.

The main aim of the paper is to act as a position guideline for upcoming research issues, taking into account what is already addressed in activities, and where more work and effort is needed to establish a common, valuable and attractive cloud e-Contract management solution.

4. BonFIRE – Towards a European Cloud Test-Bed

To elaborate the problem space and to bring the potential solutions closer to reality, we have chosen to base our concepts on top of a currently running activity, settled within the Future Internet Research and Experimentation – FIRE initiative.

The aim of the BonFIRE project (*Building Service Testbeds on FIRE*) [4] is to provide a European cloud infrastructure for scientists and to deliver a ready-to-use test facility for research and development. For the first phase, five independent infrastructure providers across Europe contribute resources to a distributed cloud test-bed, which is built upon a software stack tailored by BonFIRE (cf. Figure 1).

Figure 1- The BonFIRE Infrastructure Map (EPCC - Edinburgh Parallel Computing Centre; HP - Hewlett Packard; IBBT - Interdisciplinary Institute for Broadband Technology; INRIA - French National Institute for Research in Computer Science and Control; USTUTT - High Performance Computing Center Stuttgart)

As this is a new kind of technological approach and a first attempt on a European level, expected and, of course, unexpected problem areas arise which need to be addressed before the test bed can be brought into operation.

Similar to grid technology, cloud software stacks show a broad range of diversification. While grids needed to support several different middleware suites, such as Globus, UNICORE or gLite, the cloud landscape now offers software stacks such as OpenNebula, Eucalyptus, Nimbus [5], Tashi [6], or proprietary implementations as developed by many commercial cloud providers. And even if they all claim to follow standards, interoperability is still not given as desired by cloud users.

On top of the implementation issues, non-functional aspects of cloud usage are an additional concern of interoperability. In the long term automated QoS handling (currently performed manually) is desired, which implies the handling of different kinds of topics directly linked to that. Diversity of legal regimes is an exemplary area. Laws may mandate that personally identifiable data may only be stored in particular countries or are subject to specific access rules not known in other locations. As a result, clients using cloud services can only use those providers who respect the required data privacy rules. Even in a pan-European test-bed such as the BonFIRE one, this aspect is a concern. In addition, individual testbed providers may deny access to their resources to certain groups of customers following their own policies or complying with the policies of the country they are operating from. Other aspects of non-functional heterogeneity are contract terms and conditions, and different quality-of-service notions, definitions and specifications.

Ideally, SLA establishment happens by interaction with a central BonFIRE instance that then has to take into account all the above mentioned requisites.

In the remainder of this paper, we will concentrate on the contractual problem space arising from moving to the cloud, always referring to the BonFIRE use case, as it shows best real requirements and issues.

5. Requirements

As a result of the evaluation of existing solutions with respect to potential cloud use cases (e.g. the BonFIRE one), several topics have been identified which need to be addressed to extend current SLA mechanisms to fill the gaps mentioned in the introduction. Here we list the three we think are most important to be fulfilled:

(1) The representation of **Business Objectives** and their integration into e-Contracting frameworks. Business (level) objectives need to be taken into account during the negotiation of a contract but also during the execution of cloud services. These objectives can include policies on the pricing model, the set of business partners, the set of services offered, and the quality-of-service provided.

(2) The **access mechanisms** to either (a) define an offered service in the cloud or (b) define the desired service in the cloud. This is mandatory due to the fact that simplicity is one of the properties of clouds. Neither end-users nor providers will be attracted to take up clouds, when there are no simple interfaces/portals to allow them to do their business in an easy way.

(3) Flexibility in the **choice of protocols and underlying schemes**. Looking at all the existing solutions within the SLA domain, it is obvious that more flexibility will be needed when addressing the requirements of cloud end-users. Based on previous work by the authors [7][8], we will give an outlook on how flexibility can be reached to overcome the burden of current solutions of being stuck to a particular protocol and schema.

6. Technology Description

BonFIRE, as shown in the previous section, faces the issue that it has to combine five different test-bed facilities with heterogeneous resources to one unique test-bed. The resulting system is supposed to be accessible transparently for the end user, meaning that he should not have to deal with the peculiarities of the location and the business objectives of the provider chosen for a particular workload.

In this case, the aforementioned issues of current solutions are getting more and more obvious when trying to realize a cloud test-bed. In this section, we will give a short overview on what is currently possible exploiting state-of-the-art technology.

Covering topic (2) and partially (1), the results of the plugIT project [9] will be of interest for the realization of a “cloud-ready” SLA Management subsystem. The plugIT project is focusing on the development of a so-called IT-Socket, which allows for the alignment of business and IT similar to the provisioning of electricity via a socket to any device that can be plugged in. This is addressed by developing a Next Generation Modelling Framework which allows for the representation of infrastructure and services as machine interpretable models. Another development of the project is the so-called “Semantic Kernel” which enables semantic matchmaking of models (of different types).

The High Performance Computing Center Stuttgart (HLRS) is one of the use case stakeholders of this project and currently enhancing its Service Level Agreement Framework with the available outputs of the project. Due to this integration it is now possible for consumers to define their needs via an online application service which are then matched against the internal models of HLRS (cf. Figure 2).

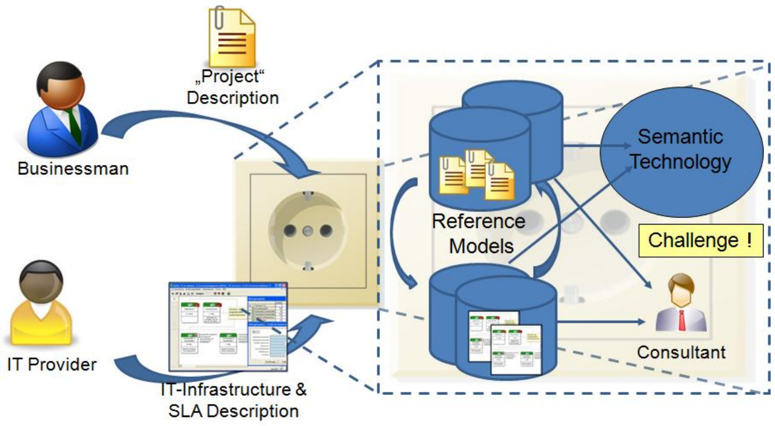

Figure 2- The plugIT Approach

While plugIT focused in the beginning mainly on the HPC infrastructure domain, the changes from grid to cloud have also affected the project's developments. By that, there is already an initial transfer of the concepts and results of this project into the cloud.

With respect to (3) also initial work has been performed [8]. But most mature e-Contracting approaches are still limited to the realization of a fixed, specific set of capabilities and do not support the use of several, context-dependent approaches.

The most prominent European project with focus on SLAs and Cloud Infrastructure-as-a-Service is SLA@SOI [10]. It provides an SLA framework comprising services from end-user to infrastructure levels. The core component of the framework is the *Service Level Agreement Manager* (SLAM), which is configured depending on the domain and on the application level (business, software, and infrastructure). Though the concepts of SLA@SOI target requirement (3) through, for example, a generic protocol engine for SLA negotiation and a generic SLA model, they affect mainly the infrastructure while the customization of use case scenarios still implies significant effort. Nevertheless, the results from SLA@SOI could serve as a foundation for the development of end-user-centric, added value services as proposed in the following sections.

7. Developments

Based on the reflection of the state of the art, we have focused our work on topic (3) which, once solved, we expect to facilitate solving the other topics. Although "choice of protocols and underlying schemes" sounds like a low-level mechanism, it does have impact on the higher layers of business objectives and access mechanisms. In particular, a wide variety of protocols broadens the coverage of business objectives and access mechanisms. The brick nature allows this coverage to be dynamically increased and customized to particular application domains and environments.

To overcome the issue of missing flexibility, we propose to change the software stacks for the handling of partner selection and the establishment of electronic contracts (and beyond that for integrating several management subsystems of a cloud environment) by providing a static base implementation (the functional core) which is then enhanced by plugged-in intelligence.

Figure 3 shows a conceptual picture of how a realization of a cloud software stack (going beyond the domain of e-Contracting) may look like, integrating the above mentioned brick approach (labeled as "Cloud Bricks"). The platform will be used by both service providers and customers, facilitating both service consumption as well as service provisioning.

Even if this paper focuses only on the e-Contracting part of management systems, the approach is applicable to other areas as well, so that the overall system as shown in the figure brings benefits to many areas beyond e-Contracting.

One important point in this concept is that the choice of bricks depends on the context of the cloud use and the definitions and concrete requirements of the respective end user and service provider. Especially the context in terms of legal terms and business objectives can have a strong impact on the deployed e-Contracting (and other) subsystems.

What becomes obvious in this diagram, and what is closely related to topic (3), is that some sort of user guidance will be needed to ensure that users, even those with moderate or little expertise, are capable of using the system in an effective way. The tools are adapted to the role a user is playing. The tools used by a service provider look different from the tools used by a service consumer. Furthermore, the tools are adapted to the experience level of the user, allowing expert users more control while offering ease of use to intermediate users.

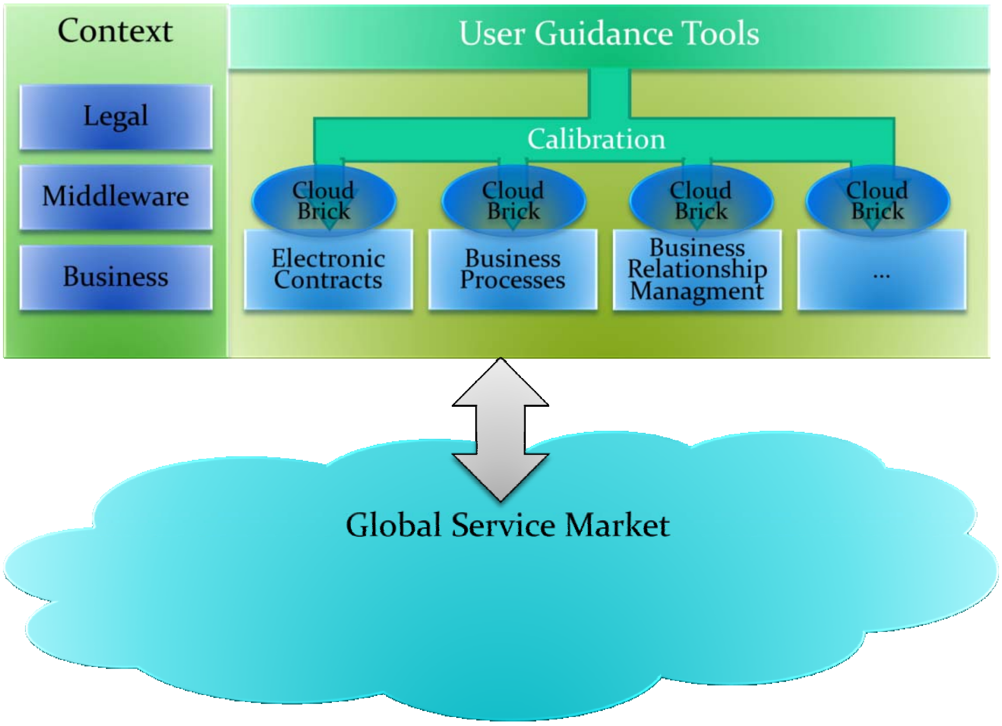

Figure 3 - Towards a Brick Supported Cloud Software Stack

Bricks are available to both service consumer and service provider. They are selected based on the requirements and constraints of the respective user (where the term "user" covers both service provider and consumer). Users therefore have to specify these requirements and constraints. Appropriate tools guide the users in specifying the requirements which then allow the platform to select matching bricks. The bricks then perform service selection, contract negotiation, and other service lifecycle tasks on behalf of the user under the guidance of the platform.

This process is exemplarily shown in Figure 4 for the contracting subsystem. The user interface tool tailored to the needs of the user is assessing the contracting requirements of the user. The requirements are fed to the brick selection service which selects the appropriate brick from the brick warehouse. The brick is put into place in the contracting subsystem. During system execution, the selected brick is used for its particular purpose, here as a protocol engine for contract negotiation. Additional bricks customize other functionality of the contracting engine, including data and negotiation models.

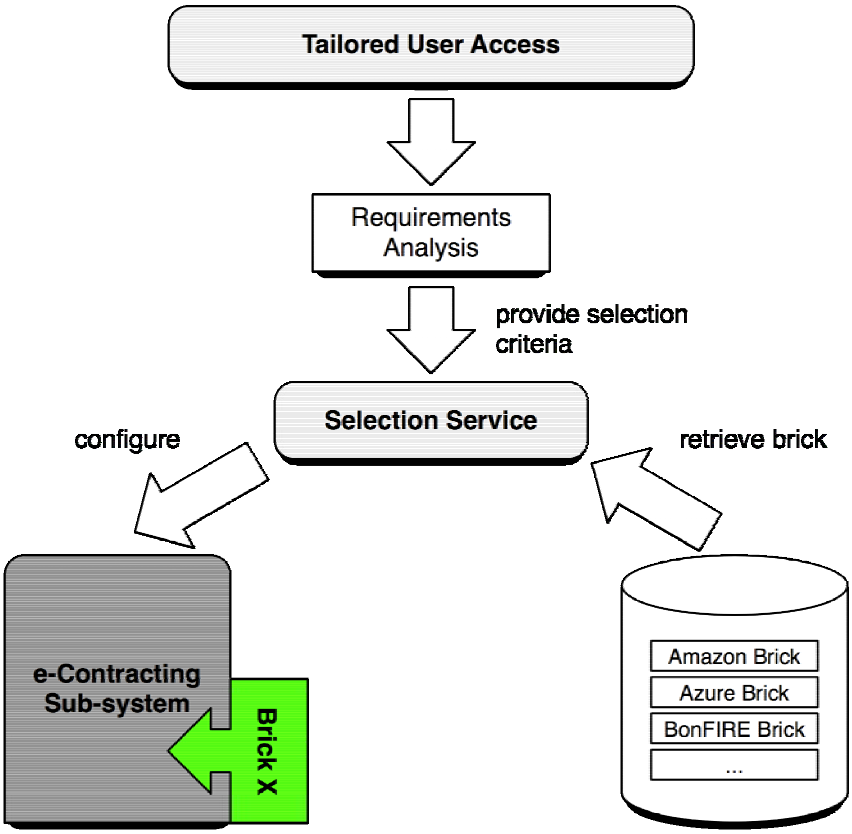

Figure 4 – Brick Operational Lifecycle

The brick approach enables mass-customization of service access by dynamically adapting access procedures to the requirements of service providers and users. The bricks can be acquired from various sources both commercial and free. The authors expect the emergence of a vibrant market place for such bricks. Different functional areas will have multiple implementations covering a wide range of customer requirements and provider policies. Software vendors will compete for the brick market based on price, performance, versatility, and reliability.

8. Business Benefits

The brick architecture has clear business benefits to not only the individual service provider and consumer, but to the whole society as business processes are made more efficient.

Service providers benefit from a brick approach as they can extend the set of potential customers. As their services can be customized by bricks, the services become attractive to a larger audience. Without a brick platform, the service provider needs to focus on providing its services to a particular set of customers using a particular set of business processes and protocols. With pluggable processes, protocols and data representations, the service provider focuses on the core service logic only and let other experts realize the logic to adapt services to various customer sets operating in various contexts (e.g. legal domains) and under different requirements (e.g. urgent service selection).

Service consumers benefit from such a platform as the set of potential business partners, i.e. providers of a particular service, increases with the number of usable protocols and processes. The enlargement of the service provider base is expected to result in increased competition among business participants which in turn is expected to decrease prices and/or increase quality of service for customers. In addition, selection of appropriate providers is (at least partially) automated, reducing costs and effort traditionally spent on selecting providers.

In addition, the brick architecture facilitates participation in the global service marketplace. The entry barrier to conducting electronic business is significantly lowered by having domain experts realize bricks that are providing appropriate functionality. Service creators can focus on their services ignoring aspects not directly related to the service logic. The set of potential business partners is therefore increased not only by improving interoperability, but also by increasing the overall number of market participants.

Taking the three aspects described above together, they should drive down overall service usage costs freeing up that money for other purposes that could benefit society at large.

9. Conclusions

The paper shows that there needs to be a shift from the all too common approach of working on already addressed issues over and over again towards focused work on closing the remaining gaps. We showed how a technology transfer from one domain (e.g. plugIT concepts from the semantics domain) can be used to close some of the gaps.

We emphasized that the current solutions need to be made more “open” in terms of allowing self-adaptation and self-calibration based on the defined needs of the users, no matter whether they are service consumers or service providers. We introduced the Cloud Bricks architecture which makes use of bricks that dynamically add concrete implementations for particular abstract processes to service implementations. Such a platform is expected to have impact on the electronic service market place, increasing its size and driving down costs.

References

[1] P. Wieder and R. Yahyapour, “Service Level Agreements in Grids”, In: Phillip A. Laplante (Ed.) Encyclopedia of Software Engineering, Taylor & Francis, 2010.

[2] P. Balakrishnan, S. Selvi, and G. Britto, “Service Level Agreement Based Grid Scheduling”. In: Proceedings of IEEE International Conference on Web Services 2008 (ICWS '08), pp. 203 – 210, 2008.

[3] L. Burchard, B. Linnert, F. Heine, M. Hovestadt, O. Kao, and A. Keller, “A Quality-of-Service Architecture for Future Grid Computing Applications”, In: Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium, 2005.

[4] The BonFIRE Project – Building Service Testbeds on FIRE, Last accessed May 25, 2011: http://www.bonfire-project.eu.

[5] P. Sempolinski and D. Thain, “A Comparison and Critique of Eucalyptus, OpenNebula and Nimbus”, In: Proceedings of the IEEE Second International Conference on Cloud Computing Technology and Science 2010 (CloudCom), pp. 417 – 426, 2010.

[6] Tashi, Last visited May 25, 2011: http://incubator.apache.org/tashi/.

[7] B. Koller, “Towards Optimal Creation of Service Level Agreements”, In: Proceedings of the eChallenges 2009 Conference (eChallenges 2009), Istanbul, Turkey, October 2009). IIMC International Information Management Corporation, 2009. ISBN: 978-1-905824-13-7.

[8] P. Wieder, P. Hasselmeyer, and B. Koller. “Towards Service Level Management in Cloud“, In: Proceedings of the eChallenges 2010 Conference (eChallenges 2010). IIMC International Information Management Corporation, 2010.

[9] Avellino et. al – D4.1.2 - 2nd System Architecture and Design, Deliverable of the plugIT project, 2010.

[10] W. Theilmann, R. Yahyapour, and J. Butler, Multi-level SLA Management for Service-Oriented Infrastructure, In: Towards a Service-Based Internet, Proceedings oft he First European ServiceWave Conference 2008, Springer, 2008. ISBN 978-3-540-89896-2.