Managing Reputation in Contract-Based Distributed Systems*

Roberto Baldoni, Luca Doria, Giorgia Lodi, and Leonardo Querzoni

University of Rome "La Sapienza",
Via Ariosto 25, 00185, Rome, Italy
{baldoni,doria,lodi,querzoni}@dis.uniroma1.it

Abstract. In industry practice, bilateral agreements are established between providers and consumers of services in order to regulate their business relationships. In particular, Quality of Service (QoS) requirements are specified in those agreements in the form of legally binding contracts named *Service Level Agreements* (SLA). Meeting SLAs allows providers to be seen in the eyes of their clients, credible, reliable, and trustworthy. This contributes to augment their reputation that can be considered an important and competitive advantage for creating potentially new business opportunities.

In this paper we describe the design and evaluation of a framework that can be used for dynamically computing the reputation of a service provider as specified into SLAs. Specifically, our framework evaluates the reputation by taking into account two principal factors; namely, the run time behaviors of the providers within a specific service provision relationship, and the reputation of the providers in the rest of the system. A feedback-based approach is used to assess these two factors: through it consumers express a vote on the providers' behavior according to the actions the providers undertake.

We have carried out an evaluation that aimed at showing the feasibility of our approach. This paper discusses the principal results we have obtained by this evaluation.

Keywords: Service Level Agreements, reputation management, trust, voting mechanism, service providers, service consumers.

1 Introduction

Typically, bilateral agreements are established between providers and consumers of services. These agreements are used to regulate service provision relationships as they include all the rights and obligations the involved parties have to fulfill in order to successfully delivery and exploit the services.

It is a common industry practice for service providers to guarantee Quality of Service (QoS) requirements in the service provision relationship; these requirements are specified in the form of legally binding contracts termed Service Level

* This research is partially funded by the EU project CoMiFin (Communication Middleware for Financial Critical Infrastructures).

R. Meersman, T. Dillon, P. Herrero (Eds.): OTM 2009, Part I, LNCS 5870, pp. 760–772, 2009.

Agreements (SLAs). Meeting SLAs allows providers to be seen in the eyes of their clients, credible, reliable, and trustworthy, thus building a "reputation capital" which is a competitive advantage for them as it implies potentially new business opportunities. However, when providers fail in meeting their obligations, consumers can be led up to decrease their judgment on the reputation of the providers and, in the worse case, break business relationships because of their full untrustworthiness.

Augmenting and consolidating reputation become thus crucial requirements of many organizations participating in distributed IT systems. To motivate this statement we consider here an IT scenario that has recently gained particular attention: the protection of financial critical infrastructures [1]. Today, an increasing amount of financial services are provided also over public mediums such as Internet; this leads those services and the supporting IT infrastructures to be exposed to a variety of coordinated and massive Internet-based attacks and frauds that cannot be effectively faced by any single organization. Cross-domain interactions spanning different administrative organization boundaries are indeed in place in the financial context, and contribute to the born of the so-called *global financial ecosystem*. Protecting the financial ecosystem against the above Internet-based attacks requires global cooperation among ecosystem participating entities to monitor, share and react to emerging threats and attacks, and collectively offer the computational power to discover and contain those threats. For doing so, relationships among those entities are to be properly managed by establishing SLAs that regulate the sharing and monitoring capabilities of the cooperative environment. If those SLAs are effectively met, the financial entities can be seen reliable and trustworthy in their working environment, thus augmenting (or consolidating) their credibility (i.e., their reputation).

Therefore, we can state that reputation is an essential and strategic asset for many organizations and cannot be neglected. In particular, due to its importance, in this paper we argue that reputation has to be treated as other common QoS requirements such as service availability, and reliability and included into SLAs in the form of functions used for determining it.

The main contribution of this paper is then the design and evaluation of a framework that provides the necessary mechanisms for dynamically defining the reputation of service providers. In the reputation function evaluation, service consumers consider two main factors; namely, the run time behaviors the providers have with them in the service provision relationship, i.e., what we call *the local history* (see below), and the reputation of the providers in other business relationships. In particular, a feedback-based approach is used to assess these two factors: through it consumers express a vote on the providers' behavior according to the actions the providers undertake. These votes can be taken as input by other consumers, as specified into SLAs.

We have evaluated the feasibility of our approach in a scenario in which a service provider is involved in more than one SLA-based business relationship with many different service consumers. Depending on the impact of the two factors earlier mentioned, as specified by the SLAs, our evaluation shows that service consumers can dynamically and differently adapt their judgement on the provider reputation.

The rest of this paper is structured as follows. Section 2 presents a scenario of reference we have used in order to design our reputation management framework. Section 3 describes the framework that allows service providers and consumers to define reputation in SLA-based distributed systems. Section 4 discusses a case study we have identified in order to validate our framework, and presents a number of evaluations we have carried out for this purpose. Section 5 compares and contrasts our approach with relevant related work. Finally, Section 6 concludes the paper.

2 Reference Scenario

Figure 1 illustrates the scenario of reference we use in order to describe the principal functionalities of our reputation management framework.

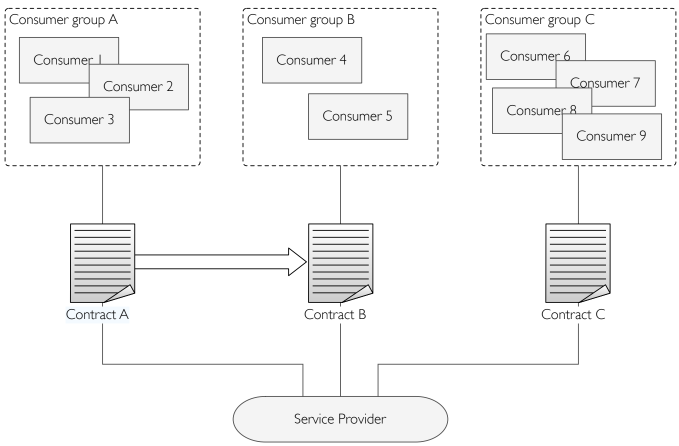

Fig. 1. Reference Scenario: an SLA-based distributed system

As shown in this Figure, we consider a distributed system in which a service provider is involved in a variety of business relationships for the provision of a specific service to different service consumers. The relationships are regulated through contracts, which we call SLAs from now on; each SLA defines different rights and obligations among the involved parties. Therefore, the distributed system can be conveniently thought of as consisting of a potentially large number of groups of consumers; these groups are formed based on the different SLA-based relationships established with the provider (the groups A, B, and C in Figure 1).

In particular, different SLAs may define different QoS requirements for which consumers are willing to pay. Included into these requirements, we propose to specify also the reputation of the provider in terms of an initial reputation value (see below) and the functions that are used to compute and update it according to the actions the provider undertake during the service provision.

In our scenario, the reputation defined into an SLA can be used as input for other SLAs, thus enabling links among SLAs and, consequently among groups of consumers. In Figure 1 this is represented by the arrow that links contract A with contract B for example. The binding among SLAs permits different consumers to take advantage of the knowledge on the provider reputation in other business relationships in order to enforce their trust in their counterpart. This knowledge can be crucial in some cases as it might be used for also determining how fast or slow can the reputation judgement towards service providers be changed. For example, if a service provider does not always behave correctly in the eyes of its clients, its reputation in other business relationships can count more and influence the final reputation evaluation.

Note that the scenario earlier described can be generalized to a distributed system with a large number of different service providers.

3 SLA-Based Reputation Framework

Owing to the reference scenario described in the previous section, our reputation management framework provides service consumers with a mean to compute the provider reputation. This is done by taking in input the SLA that regulates the business relationship among them, and that includes the rules for determining that reputation.

Thus, let p be a service provider involved in more than one SLA-based relationship with different consumers; we claim that in general its reputation is a function that can be defined as follows:

$$R_{p_j} = f(history_j, \{R_{p_i}, ..., R_{p_k}\} \setminus \{R_{p_j}\}) \quad (1)$$

$$\text{where} \quad i, j, k \in \{A, B, C, ...\}$$

To clarify the use of this function, we consider the specific scenario illustrated in Figure 1 and the reputation of the service provider from the point of view of the consumers in group B.

In this case, our framework takes in input the SLA_B, parses it, and determines the R_{p_B} that can be used by service consumers in group B in order to enforce their trust in the provider. To compute the reputation, two principal elements are taken into account: (i) the behavior of p in the SLA_B-based relationship: this is what we call *local history*; that is; the evaluation of the p's actions carried out in that specific relationship with consumers in group B, (ii) and the reputation that p has in group A, as SLA_B specifies that R_{p_A} is to be used (this is represented by the arrow in Figure 1). Note that, if more than one bind exists for SLA_B with other SLAs, the reputation function above is able to take into consideration all the reputation functions derived by the linked SLAs.

In the example of Figure 1 and using 3 above, we obtain that R_{p_B}=$f(history_B,$ $\{R_{p_A}\})$ where $R_{p_A} = f(history_A)$. In case of group A, no links with other SLAs are established and thus the reputation of p in group A is based on the local history, only. Hence, our framework is capable of differentiating the reputation of a same provider in different consumers' groups based on the different SLAs in the distributed system.

The local history is in turn a function that is computed, as in case of 3 above, by our framework as follows:

$$history_j = g(init\_rep\_value, [facts_1(p), ..., facts_k(p)]) \quad (2)$$

$$\text{where} \quad j \in \{A, B, C, ...\}$$

In other words, the local history is defined by considering an initial reputation value of the provider, which is included into the SLA, and an evaluation of the actions performed by p over the entire duration of the SLA (we call these evaluations *facts* in formula 2). These facts are the input of a feedback-based mechanism through which the different consumers belonging to a group can judge the provider's behavior based on the facts they have observed in that group.

The initial reputation is a value in the range of [0,1] and represents the initial credibility of the provider in the eyes of the different consumers. This credibility can be influenced by many factors: a previous business experience with the provider, the type of service the provider supply to the consumers, the level of QoS the provider guarantees to the consumers.

To make an example, we might think that some consumers have a previous long running business relationship with a specific provider so that their initial reputation judgement towards the provider can be high; that is, close to 1; in contrast, there can be other consumers that do not have any previous business experience with the provider so that they can be reluctant to consider the provider credible, and wish to evaluate its behavior during the duration of the contract in order to define its reputation.

The initial reputation value is then dynamically adjusted at run time by the local history function according to the judgements the consumers express on the provider p's actions. An example of a way to compute the reputation is described in the next section.

4 Adapting the Framework to an Example Scenario

In this section we show to the reader how the previously introduced framework can be adopted in a possible usage scenario, and which behaviour can be expected from it.

4.1 Scenario Description

Our possible usage scenario assumes a set of 310 clients distributed in 5 different usage groups. Clients of a group signed a contract with a service provider that can send continuous news updates on an argument of interests for the clients. All the five groups represent clients that signed similar contracts with different SLAs. Figure 2 reports this scenario. Note that, some SLAs are linked; for example,

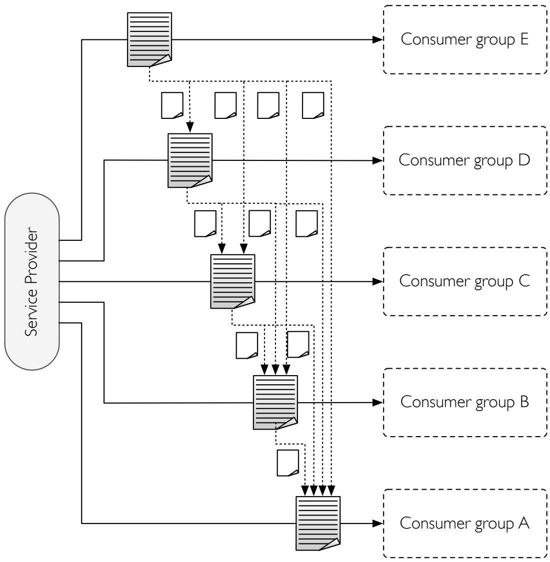

Fig. 2. An example scenario

users of group B are going to use reputation calculated at groups C, D and E and export their value to group A. Each group contains a different amount of clients: E has 10 members, D has 20 members, C has 40 members, B has 80 members, and, finally, A has 160 members.

The usage of a group reputation by another group, like group D using the calculated reputation by group E, can be seen as a sort of "service" offered by the latter to the former group. This service too can be regulated by a contract embedding various SLAs, among which, a reputation value can be present. This means that a group could repute differently reputation values provided by different groups. This "inter-group" contracts are shown in white in Figure 2.

Reputation as an SLA is defined within contracts by the two functions f and g introduced in section 3, that is the function that computes the reputation value (let us call it reputation function) and the history function. It is important to note that in each contract it is possible to introduce these functions in different ways in order to model different behaviours, but we just present one possible implementation of them. What is more, only for simplicity reasons we have used the same functions in all the contracts, but as said before we can imagine a more general situation where in each contract is introduced a different implementation of reputation and history functions.

Each client in the system maintains two different data structures for each subscribed service: the service providers reputation table and the groups reputation table. In the first table there is an entry for the service provider and a reputation number related to it; the second table contains an entry for each group whose reputation towards the service provider will be used by the client.

When a client in group x interacts with the service provider, it computes the following reputation function:

$$f_x = \frac{\sum_{i=1}^{n} R_i \cdot T_i + R_x}{n+1} \tag{3}$$

where R_i represents the reputation value provided by group i, $T_{i,x}$ is the reputation of group i with respect to clients in grup x (that is the value in the i-th entry of the groups reputation table), R_x is the current reputation of the service provider with respect to the client and, finally, n is the number of reputation values read from other groups. More specifically functions f for the different groups of our example scenario are defined as follows: $f_E = R_E$, $f_D = (R_E \cdot T_{E,D} + R_D)/2$, $f_C = (R_E \cdot T_{E,C} + R_D \cdot T_{D,C} + R_C)/3$, $f_B = (R_E \cdot T_{E,B} + R_D \cdot T_{D,B} + R_C \cdot T_{C,B} + R_B)/4$ and $f_B = (R_E \cdot T_{E,A} + R_D \cdot T_{D,A} + R_C \cdot T_{C,A} + R_B \cdot T_{B,A} + R_A)/5$.

The value returned by the function is the new reputation the client will assign to the service provider after each interaction with it. Note that, since interactions are assumed to be deterministic in this example scenario (e.g. we assume that all interactions with the service provider happens with the same order for all the clients in a group) and the method used to calculate f is deterministic, all clients in group x agree on the reputation value for the service provider.

After some interactions we suppose that a client can check through a set of facts the exact behaviour of the service provider. In the considered example the client could verify through a set of external information sources if the news sent since then by the service provider fulfills the requirements expressed in the contract. This can be modelled by defining the function g.

In our example scenario g is defined in such a way that the current reputation value for the service provider on a client is modified after each verification and increased if the verification confirms the expectations of the client ($R_x \geq 0.5$ and correct behaviour by the service provider or $R_x < 0.5$ and bad behaviour by the service provider), and reduced otherwise. The same type of update is executed on the reputation values for other groups too: if the reputation value read from a group confirms the verified behaviour of the service provider, the corresponding reputation value is increased, otherwise it is decreased. More specifically we used the following function:

$$g = (-1)^s \cdot K \cdot \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-0.5)^2}{2\sigma^2}} \tag{4}$$

where K is a constant real multiplicative factor, σ is the standard deviation and s is parameter that has value 0 if the verification confirms the reputation value or value 1 otherwise. In our example scenario we assumed $K = 0.005$ and $\sigma = 0.15$. The rationale behind this formula is that reputation values in the middle of the range are not useful to take decisions; therefore, the formula is designed to quickly polarize the reputation value either towards a large or small value and then maintain the reputation in this state despite possible sources of fluctuations.

4.2 Preliminary Experimental Evaluation

Here we show some preliminary results obtained through an experimental study aimed at characterizing the behaviour of the two functions in the aforementioned scenario. The study was realized by a simple simulator that mimics the behaviour of our system by letting (i) the provider interact with all the clients, (ii) the clients calculate their reputation values and (iii) the clients adapt their values after validating the result of the previous interaction. A number of different interactions is realized in subsequent step, while the reputation values on the clients are observed. Note that our tests do not take into account all the problems involved with exchanging reputation values in a real distributed setting where clock are not perfectly synchronized, communications happening through network links can suffer from delays and failures, etc. While these implementation aspects are fundamental in order to make our system actually deployable, in this paper we focussed our attention on the potentialities offered in the proposed context by our framework, thus we limited our evaluation to more high level aspects.

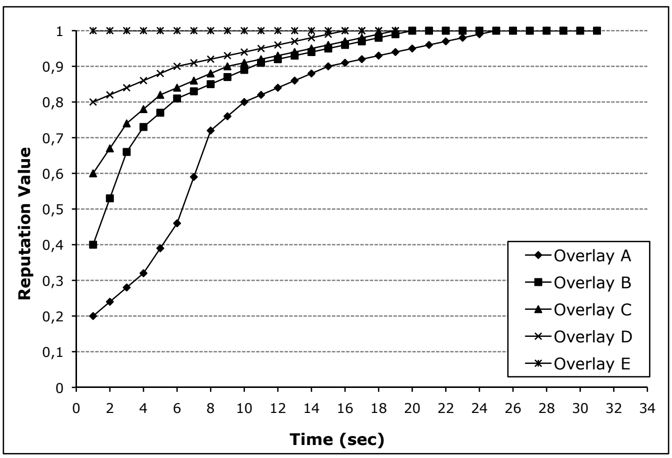

Fig. 3. Evolution of reputation when the provider behaves in a good manner

Figure 3 reports the behaviour of reputation values in a setting where the provider is supposed to always interact with clients completely fulfilling the requirements expressed in the contracts. Clients in different group start with different reputation levels ranging from 0.2 for group A to 1.0 for group E. The curves report the value of reputation calculated in different groups versus time, assuming that a new interaction with the provider happens every second. As the graph shows all reputation values converge toward value 1.0 driven by both the verification of the provider behaviour and the reputation values read from other groups.

Figure 4 shows the behaviour of reputation values in the opposite setting where the provider is supposed to always violate the SLAs defined in the contracts. As the graph shows, in this case all reputation values converge toward value 0.0 but do so with widely different behaviours. Clearly, clients in group E, starting from a reputation value 1.0, tend to maintain an higher reputation value. However, as time passes by, also the reputation of clients in E is driven toward 0 by the subsequent verification of the provider behaviour.

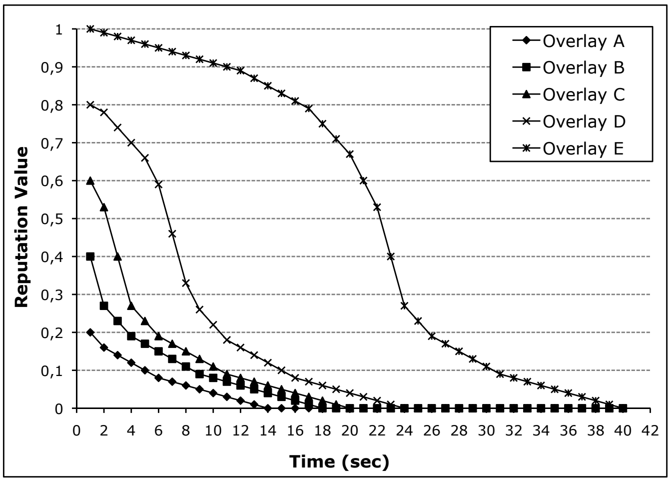

Fig. 4. Evolution of reputation when the provider behaves in a bad manner

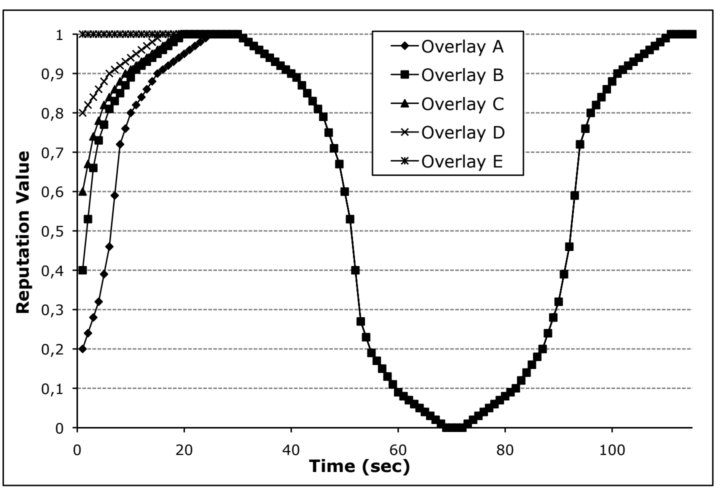

Fig. 5. Evolution of reputation when the provider behaves alternatively in a good and bad manner

More interestingly, Figure 5 shows the results obtained in a setting where the provider behaves alternatively in good and bad ways. The graph shows how the reputation values adapts to the provider behaviour in a consistent way: all groups agree on a same reputation value during the whole execution of the test, showing how the proposed adaptation of the framework to the scenario of interest is able to produce a shared picture of the provider reputation, regardless of the specific contracts signed with different clients.

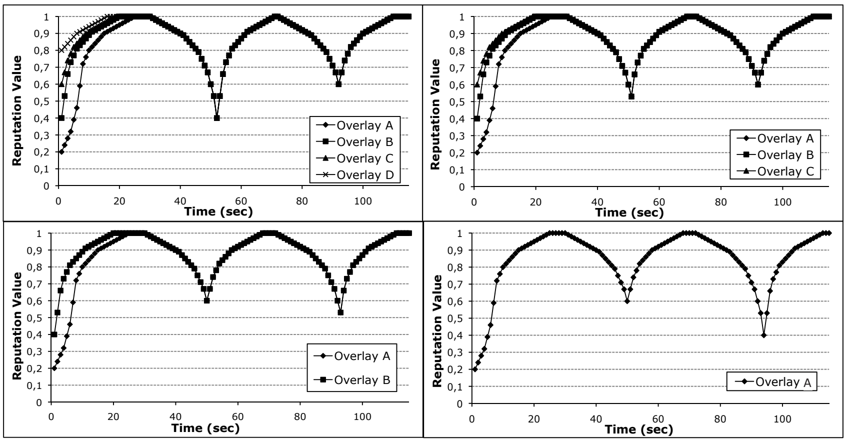

Fig. 6. Evolution of reputation towards group E (up/left), group D (up/right), group C (bottom/left) and group B (bottom/right)

We also observed, considering the same setting, how reputation among groups vary during the test. Figure 6 shows in four different graphs the average reputation of groups E, D, C and B respectively (reputation of group A is not considered in our example case). Clearly the reputation of groups show cuspids whenever that group changes its opinion with respect to the behaviour of the provider, i.e. when its reputation crosses the 0.5 "border" value. The time at which this crossing happens has an impact on the minimum value of the group reputation: the sooner is the better from this point of view as other groups will not be forced to decrease too much the reputation value.

5 Related Work

We have identified two principal macro research areas in which our work can be located; namely, reputation management systems and contract(SLA)-based distributed systems.

In this section we compare and contrast a number of works that fall into these two categories in order.

Reputation management is a body of research that has been recently widely investigated in the literature. This is proven by a large number of works that can be cited in the context of this paper [3], [8], [7], [5], [6], [4], [9]. For the sake of conciseness, we do not describe and contrast all of them in detail; rather we discuss here those works we believe are the closest to ours and from which we have derived a number of design principles and recommendations.

An important and successful example of reputation management is the online auction system eBay [2]. In eBay, buyers and sellers can rate each other after each transaction and the overall reputation of a participant is the sum of the ratings over a certain time period. This permits to build a user profile that might be used for future interactions with that user. Similarly, our approach exploits such an information that is however formally specified and used into SLAs with the service providers in order to compute their reputation value.

In [3] the authors describe a reputation system named EigenTrust that allows peers in a P2P enviroment to decrease the number of downloads of inauthentic files based on a unique global trust value assigned to each peer in the system. The value of trust is computed by taking into account the history of uploads. In essence, the EigenTrust system peers can choose from whom they can download files, thus effectively identifying and isolating malicious peers.

In [8] the authors propose an approach to P2P security where servents keep track and share with others information regarding the reputation of their peers. The reputation sharing mechanism is based on a distributed polling algorithm through which resource requestors can evaluate the reliablity of providers before initiating the download of files in a P2P system.

In [7] the authors describe a probability-based adaptive reputation mechanism by which newcomers will be trusted based on system's trust-probability that is dynamically changed according to the actions undertaken by the newcomers. In particular, the system presented in [7] relies on a group-based P2P environment in which users are organized into groups and one user belongs to only one group The groups are created based on the physical topology, the interest locality and so on. When a user joins the network a new reputation value is computed that is based on an initial probability; this probability is adjusted at run time according to the behaviors of all the new users.

All these works share a number of similarities with our approach. In all cases, a feedback-based mechanism is enabled in order to assess the reputation of participants of the system. In particular, in most cases a local reputation value is used as we use the local history function to compute the reputation of the provider locally to a specific group of consumers. In addition, all the works above are used in P2P environments for computing the reputation of peers for file sharing purposes. Typically, in these systems there is no contract used to regulate relationships among peers; the system is "flat" and the reputation is not controlled and differentiated on the basis of the type and requirements specified for the service provision. Only in [7] the authors design their reputation mechanism in a group-based P2P system. However, the groups are not formed based on contracts established among provider and consumers of services as we propose.

Owing to this latter observation, we have investigated a number of researchers that fall into the contract-based systems category in order to highlight the main differences from the reputation management standpoint. Once again, we report in this paper those that allowed us to derive important design principles for the development of our framework.

Specifically, in [10] the authors describe a distributed mechanism for managing the load in a federated system. The mechanism is based on a pairwise contracts negotiated offline between participants. Contracts set tightly bounded prices fro migrating each unit of load between two participants and specify the set of tasks that each is willing to execute on behalf of the other.

In [11] the authors describe an SLA-driven clustering mechanism for dynamic resource management. Specifically, they propose a middleware architecture that is capable of dynamically change the amount of clustered resources assigned to hosted applications on-demand, so as to meet application QoS requirements specified within SLAs.

In [12] the authors introduce a number of definition; namely, trust, reputation, trustworthiness in service-oriented environments. In addition they provide the readers with a definition of the trust and reputation relationships that can be created in such environments. Based on these definitions, they introduce a graphical notation for representing those relationships.

In [13] the authors present a contract framework in which accounting policies can be specified. The participants' trustworthiness is measured against these policies, thus ensuring that resources are allocated rationally. In their system the contract represents an action that is to be performed, and simultaneously provides a high-level description of the action. A subjective trust model is one of the foundation of their contract architecture: the belief, disbelief and uncertainty functions are used to represents the apparent trustworthiness of a participant; these functions are subjectively determined by each participant, based on their experiences.

From the assessment of the above works we can notice that most of them do not consider reputation as a requirement to be included in contracts. Only [13] seems to be the closest work to ours. However, to the best of our knowledge, the authors do not foresee a system in which contracts can be related one another by taking as input reputation policies specified in other contracts.

6 Concluding Remarks

In this paper we have described a framework that allows service consumers of a distributed system to assess the reputation of service providers in an SLA-based service provision relationship. In particular we have presented an example of usage of our framework in which groups of service consumers are able to dynamically adapt an initial reputation value specified into SLAs according to the observations on the behavior of the provider. We have carried out a preliminary evaluation of our framework that aims at showing how the reputation judgement of groups of consumers is dynamically adjusted in case the provider is reliable, malicious, or behaves sometimes in a reliable way and sometimes maliciously.

Future works include the design of a complete implementation of our framework that is able to deal with asynchronous settings and failures in the distributed system.

References

1. The CoMiFin project (2009), http://www.comifin.eu
2. Ebay, http://www.ebay.com
3. Kamvar, P.S., Schlosser, M.T., Garcia-Molina, H.: The EigenTrust Algorithm for Reputation Management in P2P Networks. In: ACM Proc. of WWW 2003 Budapest, Hungary, May 20-24 (2003)
4. Gupta, M., Judge, P., Ammar, M.: GossipTrust for Fast Reputation Aggregaion in Peer-to-Peer Networks. IEEE Transactions on Knowledgement and adata Engineering (February 2008)
5. Zhou, R., Hwang, K., Min, C.: A Reputation System for Peer-to-Peer Networks, In: Proc. of NOSSDAV 2003, Monterey, Califoria, USA, June 1-3 (2003)
6. Zhu, Y., Shen, H.: TrustCode: P2P Reputation-Based Trust Management Using Network Coding. In: Sadayappan, P., Parashar, M., Badrinath, R., Prasanna, V.K. (eds.) HiPC 2008. LNCS, vol. 5374, pp. 378–389. Springer, Heidelberg (2008)
7. Sun, L., Jiao, L., Wang, Y., Cheng, S., Wang, W.: An adaptive group-based reputation system in peer-to-peer networks. In: Deng, X., Ye, Y. (eds.) WINE 2005. LNCS, vol. 3828, pp. 651–659. Springer, Heidelberg (2005)
8. Cornelli, F., Damiani, E., De Capitani di Vimercati, S., Paraboschi, S., Samarati, P.: Choosing Reputable Servents in a P2P Network. In: ACM Proc. of WWW 2002, Honolulu, Hawaii, USA, May 7-11 (2002)
9. Procaccia, A.D., Bachrach, Y., Rosenschein, J.S.: Gossip-Based Aggregation of Trust in Decentralized Reputation Systems. In: Autonomous Agents and Multi-Agent Systems, January 22 (2009)
10. Balazinska, M., Balakrishnan, H., StoneBraker, M.: Contract-Based Load Management in Federated Distributed Systems. In: 1st Symposium on Networked Systems Design and Implementation, San Francisco, CA, USA (March 2004)
11. Lodi, G., Panzieri, F., Rossi, D., Turrini, E.: SLA-driven Clustering of QoS-aware Application Servers. IEEE Transactions on Software Engineering 33(3), 186–197 (2007)
12. Chang, E., Dillon, T.S., Hussain Khadeer, F.F.: Trust and Reputation Relationships in Service-Oriented Environments. In: Proc. of the Third International Conference on Information technology and Applications, ICITA 2005 (2005)
13. Shand, B., Bacon, J.: Policies in Accountable Contracts. In: Proc. of the Third International Workshop on Policies for Distributed Systems and Networks, POLICY 2002 (2002)