Scalable Computing: Practice and Experience
Volume 6, Number 3, pp. 95–106. http://www.scpe.org

ISSN 1895-1767
© 2005 SWPS

TOWARD REPUTABLE GRIDS

G. VON LASZEWSKI*, B. K. ALUNKAL[†], AND I. VELJKOVIC[‡]

Key words. Grid, Quality-of-service, Trust, Reputation

Abstract.
The Grid approach provides a vision to access, use, and manage heterogeneous resources in virtual organizations across multiple domains and organizations. This paper foremost analyses some of the issues related to establishing trust and reputation in a Grid. Integrating reputation into quality management provides a way to reevaluate resource selection and service level agreement mechanisms. We introduce a reputation management framework for Grids to work toward facilitating the complex task of improving the quality of resource selection. Based on community experience we adapt trust and reputation of entities through specialized services. Simple contextual quality statements are evaluated in order to effect the reputation for a monitored resource. Additionally, we introduce a novel algorithm for evaluating Grid reputation by combining two known concepts using eigenvectors to compute reputation and integrating global trust.

1. Introduction. The Grid approach [18, 21] provides a *vision* to develop an environment for coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations under quality-of-service constraints [5, 10]. However, optimal use of these distributed services and resources requires not only knowledge about the capabilities of the resources, but also the assurance that the available and requested capabilities can be used successfully. Grid users are faced with questions such as which resources are available remotely, which capabilities these resources have, whether one is authorized to use these resources, whether the information for a resource selection is accurate, and on which resources a task is likely to execute with the most success.

In a typical Grid scenario users identify possible candidate resources through metainformation obtained from directories, databases, or registries. However, the current generation of Grid information services provides only the most elementary information to guide quality-of-service based resource selection. For example, the Globus Toolkit Monitoring and Directory Service (MDS) [19] provides a limited set of information about Grid resources, including static and possibly dynamic attributes and properties. In many cases the information returned by this service is costly to obtain, inaccurate, or outdated and does not integrate a resource selection service. We observe that, similar to Heisenbergs uncertainty principle [13], the more variability (momentum), the information in regards to a resource attribute contains, the less we can predict the accuracy of its value at a time, and vice versa. This principle is of especial importance if we consider the use of multiple resources in a coordinated fashion, multiplying this effect. Furthermore, the sporadic nature of the Grid and its measured values as well as the possibility of integrating ad hoc services [21] in a Grid environment for which no historical data is available, poses a severe limitation on the current generation of prediction services. Additionally, we often lack information provided on the quality of the participating entities, similar to an Internet shopping site, which classifies included items while augmenting them with information not only about functionality, appearance, availability, and price, but also about appreciations and ratings by its shoppers.

In our framework we propose a probabilistic preselection of resources based on likelihood to deliver the requested capability and capacity. Such a service can be integrated into a quality-of-service management framework [7] to enable the reevaluation of the effectiveness of quality-of-service policies and service level agreements.

This motivated us to design a reputation framework for Grids to assist in the selection process for resources while integrating the notions of trust and reputation. Trust is already a critical parameter in the decision-making process of several peer-to-peer (P2P) frameworks. Reputation is computed by using a trust rating provided by users of services through a feedback mechanism. Reputation-based service and product selection have proved to be a great asset for online sites such as eBay [9] and Amazon [3].

Hence, we propose a framework that selects through a hierarchical process, with the help of sophisticated Grid service, sets of resources and services as suitable candidates to fulfill quality-of-service requirements. This includes the selection of trusted resources that best satisfies application requirements according to a predefined

*Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439, U.S.A. gregor@mcs.anl.gov
[†]Department of Computer Science, Illinois Institute of Technology, Chicago, IL 60616, U.S.A.
[‡]Department of Computer Science and Engineering, The Pennsylvania State University, PA 16802, U.S.A.

trust metric. Therefore, we propose that our hierarchical resource selection process be augmented by qualitative and quantitative experiences based on previous transactions with resources so we can integrate this experience in future resource selections.

We envision such a reputation system for Grids, in which resources and services are ranked based on the reputation they obtain. Generating a reputation or establishing trust by entities (resources, services, and individuals) in regards to their availability and capability. We believe that such a reputation service framework is of crucial importance for Grid computing to increase reliability, use, and popularity. Trust and reputation serve as an important metric to avert the use of underprovisioned and malicious resources; they provide the ability to simplify the selection process while focusing first on qualitative concerns.

Consider a Grid environment that agglomerates expensive and specialized resources including high-performance servers, storage databases, advanced scientific instruments, and sophisticated services to visualize macromolecules [22] or nanomaterial [4] structures. Such an environment requires reliable ad hoc Grid services to fulfill the necessary quality-of-service required by secure real-time use. Furthermore, the sporadic and time-limited nature of the services and resources used may result in a lack of historical data, posing severe limitations on existing prediction services.

Community-based adaptive metrics such as trust and reputation serve as building blocks to support our quality-of-service requirements. We emphasize that the self-evaluation of a service must be an integral part of the Grid architecture in order to increase reliability and predictability. Consider the case in which a service claims it will provide a particular level of quality and engages in a service level agreement with another service. Assume that this service fails to deliver the promised agreement. Such a scenario might exist when the metrics available for selection do not coincide with the goals. Choosing a more reliable service can avoid this problem. But how do we know that another service is more reliable?

Concretely, if we try to transfer 10 Gbytes of data between remote resources through a network, we might be tempted to select a network path with the highest observed peak throughput. However, if the network gets interrupted and the transfer would fail, the measurement and metric must take this into account. We cannot rely on a service that selects the route for transfer based only on a simple bandwidth measurement. Rather, we require a service that evaluates the promised agreement and is available for future reference. Hence, we are not only concerned with the quality-of-service, but also with the quality-of-information [20] to establish such a service.

We need to address in an effective quality-of-service framework the following issues:

1. Identify the metrics that are defining the service,
2. Implement a quality-of-service policy,
3. Provide measurements that can help selecting resources under metric service level agreements,
4. Decide for a service agreement,
5. Preselect a number of resources that will likely fulfill the agreement,
6. Execute the service,
7. Evaluate the policies by measuring a successful response,
8. Adapt the strategy if it was not successful, to select new resources (i.e, return to Step 5).

In this paper we will focus on Step 8 of this framework. Other aspects are addressed in [2].

Our paper is structured as follows. In Sections 2, 4, and 5, we define the terms trust and reputation and provide an overview of the existing reputation systems for the Grids and their limitations. In Section 3, we present the general requirements of Grid reputation framework and service. In Section 6, 7, and 8, we propose a new algorithm for managing reputation in Grid-based systems and discuss its underlying architecture. After we provide an overview of other related work we summarize future work and conclude our work.

2. Trust and Reputation. In this section we define the basic terminology used throughout the rest of the paper.

2.1. Definition: Entity. For simplicity, we refer to a resource, agent, service, organization, or user as an *entity*. This definition allows us to specify the term “trust in the most general way while applying it to the Grid approach.

2.2. Definition: Entity Trust. As pointed out by many researchers, trust is an ambiguous concept that defies exact definition. Based on economic models [11], however, we can define trust as a commodity for reducing risk in unknown situations. Hence, trust has an important role in enabling interactions in an unfamiliar environment while weighing the risks associated with actions performed in that environment. The protection of trust through economic incentives is an important factor to allow trust to become a stable commodity. For our proposed framework, trust is the underlying principle that we determined through local or global interactions among entities and their decisions based on it.

2.3. Definition: Virtual Trust. So far we have not discussed the flow relationships between trustors and trustees. If a trust value in a community is assigned to an entity (the trustor) its trust value can be reused by a new trustee who joins the community and adheres in principle to the same values as the community members. In this case we use the term *community trust*, or *virtual trust*.

2.4. Definition: Entity Reputation. Reputation refers to the value we attribute to a specific entity in the Grid, based on the trust exhibited by it in the past. It reflects the perception that one has of another's intentions and norms. Entity reputation provides a way of assigning quality or value to an entity. Reputation is usually associated with a time factor; it is often gained over time, based on qualities attributed to it by evaluations of other entities. In many reputation models, reputation decreases quickly based on adverse behavior.

2.5. Definition: Entity Reputation Service. An entity reputation service is defined as a secure information service responsible for maintaining a dynamic and adaptive trust and reputation metric within a community. Entities in the Grid continuously interact with the reputation service to create a community rating mechanism that cooperatively assists their future decisions based on the overall community experiences.

3. Trust Models. To define a trust model, we need to establish trust requirements, assign trust ratings, and define trust mediation frameworks and algorithms. Because of the diversity of the Grid and its communities, we cannot define a single trust model suitable for every case. Instead, we need to revisit the requirements and the circumstances in which such a trust model brings added value to the Grid infrastructure. Some of the most common ingredients used to design trust models for Grids are neighborhoods, communities, virtual organizations, contracts, branding, and ownership.

3.1. Neighborhoods and Communities as Trust Models. One of the most common trust models is based on the definition of neighborhoods and communities. Here a group of entities form a relationship network that can be used to query about the trust the members have for another entity to be accessed or used. Neighborhoods are typically small peer-to-peer groups where each member typically knows the others. In contrast to this model, communities contain many more members, and it may no longer possible that for member of the community to know the others. In both groups, however, trust and reputation are established through standards and common views governed by the communities and neighborhoods. Ratings are Adapters through interpersonal communication or through publication on a community-wide scale. A good example of a neighborhood trust model is the close interaction among computational scientists to interpret the outcome of a particular scientific experiment. A good example of a community trust model is the collection and publication of opinions about a particular topic. In some cases trusted neighborhoods are established to provide the community with trust ratings. An example is an editorial board for the publication of articles in a scientific journal. The scientific community pays more attention to an article reviewed by its peers than to an article published on a unmoderated Web page.

3.2. Geography and Political Boundaries as Trust Model. A simple way to establish neighborhoods and communities is to consider geographical distance or political boundaries. Being a citizen of a foreign country will be in most cases require special clearance to participate in entities controlled by a government or university as is often the case for supercomputing centers. Geographical constraints may be needed in order to restrict adaptive trust algorithms to a number of entities in close vicinity. This is often the case for certificate authorities that have branches operating in geographical distributed location to verify the physical existence of a person. Hierarchical Grids such as the TeraGrid or the Physics Data Grid function in such fashion. Although considered a virtual organization, membership into this organization sponsored by the community is determined by local trust authorities.

3.3. Contracts as Trust Model. A contract is a binding agreement between two or more persons or parties. Contracts are currently under much discussion as part of service level agreements in QoS-based frameworks such as Web services and Grid services. Here a contract between entities is formed and agreements are cast to fulfill a particular service. This concept is based on the trust that the agreement will be fulfilled. If an unrepeatable entity is present, however, the model will not function, and adaptations need to be made to enforce the agreement (e.g., through litigation or punishment). One of the earliest such models used in Grid computing was experimented with by the Java CoG project in 1997 in a high-throughput structural biology project. Resources were put together in a pool and if a resource failed to report or the average time taken by other resources to respond was above a threshold, that resource was marked as unfavorable and was chosen only if no other resources were available. In other words, the resource obtained a certain reputation based on its contractual fulfillment.

3.4. Ownership as Trust Model. Highperformance computing has traditionally focused on ownership models. Such models are an extension of the community model in which, however, the ownership of an explicit entity forms a community. In the 80s and 90s these models were driven by supercomputer centers that offered their users exclusive use of supercomputers through batch queuing systems. Today, in Grid, the ownership model is the most common one. We believe that in future, however, we will see a shift toward virtual ownerships (as already promoted by the concept of virtual organizations). Not only will we see virtual organizations but we will also see soon virtual memberships to these organizations.

To apply the concept of ownership to community Grids [21], one must revisit the role of virtual organizations, institutions, and members creating them. Since shared resources in a virtual organization are contributed by various institutions, an elaborate reputation service is needed, that deals with the fact that resources can be part of multiple domains and VOs. The different cases are depicted in Figure 3.1. We use the following nomenclature: ${}^{n}E_i$ defines an entity with the label i that is shared by n organizations. In case we do know a percentage of share, we augment it appropriately ${}^{p_1 \cdots p_n}E_i$ where p_k defines the percentage of ownership of organization k. Considering this nomenclature, we can define use of entities based on the reputation entities obtain. We note that entities within organizations can evaluate each other. To make the system work, however, we need to define a value-based system across the organizations or maintain reputation for different communities and virtual organizations.

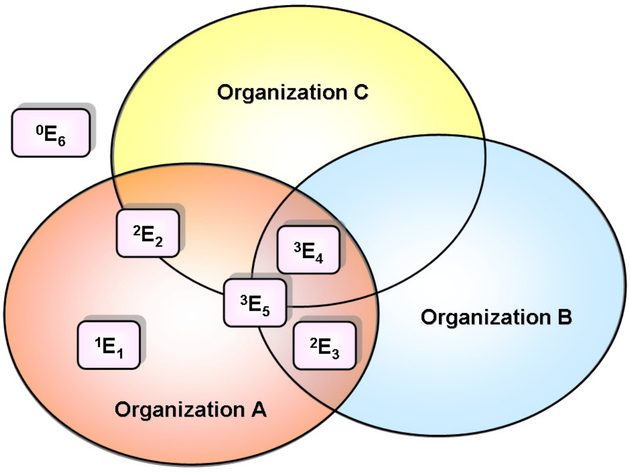

Fig. 3.1. *Institutions contribute in various ways their resources and services to possibly various virtual organizations.*

3.5. Use as Trust Model. One of the simplest trust models is based on the number of uses. The concept is the following: the more the entity is used, the higher the trust in this resource. Common sense suggests that when so many perceive this entity as desirable, it must be so. Use statistics have long been popular in the computer industry, although these often give a first impression of which entities should be considered, one must make sure that the concept of popularity is independent of other attributes such as security or even content. One need only consider popular but insecure operating systems on Web pages with dubious content appreciated by a large number of Internet users that have achieved more popularity than true content driven pages.

3.6. Branding as Trust Model. One other important concept in industry that is related to reputation is *branding*. Here the reputation of continuously high recommended entities that belong to a particular class or organization may create the desire by other customers to use the same well known brand. Branding is usually in business a good concept as outliers of poor accidental events effecting the reputation negatively are damped.

In computer science the concept of branding is also often used in regards to organizations and products derived from these organizations.

3.7. Time as Trust Model. Time is an essential variable as part of each trust model. Trust and reputation models have sometimes a wide variety of potential conflicting time assumptions. We have branding that clearly augments an entity with a reputation that is less time sensitive than establishing short term contracts between entities that only deal with one time interactions. A similar concept to branding is seniority with time in which the assumed entity becomes a seniority value that is based on experience gained through interaction with the community. Statements such as I have done it this way for years, it must therefore be working for the upcoming years are common.

3.8. Economy as Trust Model. In order to establish a better reevaluation methodology, trust models can be augmented through economic models. For example, contracts can be signed under exchange of real or virtual money, use can be rewarded through a coupon system, and auctions or markets can be put in place to bid for the most trusted and capable resources. This approach naturally can succeed only if a common, controllable commodity such as (virtual) money is used. Business and economic research in these areas is plentiful; indeed the term *virtualization* in business models long before the Grid community used these terms [17].

3.9. Reputation as Trust Model. As indicated earlier, reputation can be used as a major enhancement to each of the models introduced. Since reputation defines a metric, we should be able to use this metric to select entities for closer consideration as part of a neighborhood, community, or virtual organization and help support models employing economic goals, usage, and to establish contracts. This is of especial importance because the time it takes to query all available entities for the best possible fit may be too large. Hence we need to group a class of properties of interest to a particular community and preselect from the many thousands or millions those that give the highest likelihood of success.

4. Application of Reputation Related Trust Models. Trust models and use of reputation frameworks has been considered in a wide variety of systems. The most visible frameworks have been used to enhance business and information services available for a large community through the Internet.

4.1. Review Trust Model. One popular use to establish reputation is to design information portals, similar to C|net [8], which maintains ratings for products based on the ratings of an editor. Integrating feedback from the community provides an additional value in order to reevaluate the judgment of the editor against input from a larger community. Although, the community feedback is not integrated into the editors rating it is still available for review. Hence, the consumer must review both pieces of information to obtain an accurate picture. Detailed textual reviews are also provided to provide the consumer with a semantic explanation on the reason for the given grade by another consumer. The advantages of integrating a community are that the bias of an editor may be minimized. The disadvantage is that invalid responses not corresponding to the editors standard could result in an incorrect evaluation.

4.2. Buyers and Sellers Reputation Trust Model. The online auction system eBay [9] is an important example of successful reputation management. In eBay's reputation system, buyers and sellers can rate each other after each transaction. The feedback system is based on a simple point system, that assigns a positive point for a positive feedback, No points for neutral feedback, and a negative point for a negative feedback. The reputation is the summation of all feedbacks for a buyer or seller over the last six month. Additionally, the feedback is classified in a detailed view to be groups in time periods of the past 7 days, the past month, and the past six month. E-bay points out the a high feedback rating not necessarily means a good reputation. It "is a good sign, but a consumer "should always check a member's feedback profile for any negative remarks. It's best not to judge users only on their feedback ratings.

4.3. Information Ranking. The search engine Google [6, 15] provides a reputation and trust model based on a method called PageRank that uses the links between pages as input. Here a link from other pages to the page in question is interpreted as a positive sign and indicates that the page has some importance. The model is based on the concept that the more links can be found the more important the page is. Additionally, it weighs the pages based on the importance of the voting page.

5. Basis of GridEigenTrust. Before discussing our Grid reputation management framework and the GridEigenTrust algorithm, we provide a short overview of current research efforts that form the basis of our work. The GridEigenTrust algorithm is inherently based on the peer-to-peer (P2P) EigenTrust algorithm [16] and the use of reputation to define evolving and managed trust in Grids through the introduction of global trust [1]. The GridEigenTrust algorithm combines these algorithms making it conducive for a large Grid environment by increasing its scalability.

5.1. EigenTrust Algorithm for P2P Networks. A reputation management algorithm for P2P networks, called EigenTrust, is introduced in [16]. We summarize the main principle but use within this section the term entity instead of peer in order to provide a uniform nomenclature. Every entity E_i rates other entities based on the quality of service they provide. Therefore, every entity E_j with whom E_i had business will be rated with a grade g_{ij} $(i \overset{g_{ij}}{\rightarrow} j)$ and is normalized as described in [16]. Hence, for each entity E_j, the normalized local trust value c_{ij} is defined as follows:

$$c_{ij} = \frac{\max(g_{ij}, 0)}{\sum_j \max(g_{ij}, 0)} \tag{5.1}$$

The normalized local trust values throughout the P2P domain needs to be aggregated. This procedure can be done by means of a transitive trust mechanism: entity E_i asks his friends for their opinions about other entities:

$$t_{ij} = \sum_k c_{ik} c_{kj} \tag{5.2}$$

where t_{ij} represents the trust that entity E_i puts in entity E_j based on the opinion of his k friends. The coefficients are assembled into a matrix, $C = [c_{ik}]$. Hence, equation (5.2) can be written in matrix notation as shown in equation (5.3):

$$\vec{t_i} = C^T \vec{c_i} \tag{5.3}$$

The process of obtaining the trust values of friends is repeated to obtain the transitive closure of the matrix. After n iterations, where n is the rank of the matrix, the transitive trust is obtained. For large n, $\vec{t_i}$ converges rapidly as shown in [12], to the same value $\vec{t}$. Hence, $\vec{t}$ shows how much trust the system as a whole has for every entity E_i.

5.2. Managing Reputation in Grid Networks. In [1, 14] several aspects of trust values are considered as part of a *global* reputation model. In this model it is assumed that the trust values decay with time. It is also assumed that the trust model should stimulate organizations to sanction entities who are not behaving consistently in the Grid environment and who break trust relations. Finally, it is assumed that trust relationships are based on a weighted combination of a *direct* relationship between domains and the *global* reputation of the domains. The model is also based on contexts that, in Grids, can be numerous, varying from executing a specific job, to storing information, downloading data, and using the network. To reflect more accurately the terminology of the Grid, we replace the term domain with organization. We believe that the domain is not an appropriate division for trust within Grids.

Our goal is to define a formula for the trust relationship function Γ, based on the parameters time, context, and the organizations involved.

- Let O_i and O_j denote two organizations.
- Let $\Gamma(O_i, O_j, t, c)$ denote a trust relationship based on a specific context c at a given time t of O_i toward O_j.

Next we define Γ with the help of the following functions:

- Let $\Theta(O_i, O_j, t, c)$ denote a direct relationship for the context c at time t of O_i towards O_j, which is the relationship between neighboring organizations that have direct relationships between entities in both.
- Let $\Omega(O_j, t, c)$ denote the global reputation of O_j for the context c at time t.
- Let $DTT(O_i, O_j, c)$ denote a direct trust table entry of O_i for O_j for context c. The table records the trust value from the last transaction between O_i and O_j.

- Let $\Upsilon(t - t_{ij}, c)$ denote the decay function for specific context c, where t is current time and t_{ij} is the time of the last update of DTT or the time of the last transaction between O_i and O_j.

In [1, 14], $\Gamma(O_i, O_j, t, c)$ is computed as the weighted sum of direct relationship between domain and global reputation of the domain:

$$\Gamma(O_i, O_j, t, c) = \alpha \cdot \Theta(O_i, O_j, t, c) + \beta \cdot \Omega(O_j, t, c) \tag{5.4}$$

where $\alpha, \beta \geq 0$, $\alpha + \beta = 1$.

The direct relationship is affected by the time elapsed between interdomain contacts, hence

$$\Theta(O_i, O_j, t, c) = DTT(O_i, O_j, c) \cdot \Upsilon(t - t_{ij}, c) \tag{5.5}$$

The global trust for domain O_j is computed as

$$\Omega(O_j, t, c) = \frac{\sum_{k=1}^{n} R(O_k, O_j) \cdot RTT(O_k, O_j, c) \cdot \Upsilon(t - t_{kj}, c)}{\sum_{k=1}^{n} (O_k)} \tag{5.6}$$

where $R(O_k, O_j)$ is the recommender's trust level, and RTT is usually equal to DTT. Since reputation is based primarily on what organizations say about another domain, the recommender's trust factor $R(O_k, O_j)$ is introduced to prevent cheating through collusion among a group of domains. Hence, $R(O_k, O_j)$ is a value between 0 and 1 and will have a higher value if O_k and O_j are unknown or have no prior relationship among each other and a lower value if O_k and O_j are allies through, for example, a virtual organization relationship.

6. GridEigenTrust Framework. In this section we introduce more details about our proposed GridEigenTrust framework. We begin by pointing out some of the limitations of the two other approaches discussed in Section 5. Then, we show how one can build a more advanced framework by combining the two approaches, while avoiding their limitations while applied to the Grid.

The eigenvalue approach discussed in 5.1 is explicitly designed for P2P networks. It has not been applied to the underlying architecture of Grids that introduce virtual organizations, providing an obvious classification of resources, users, and their reputation that is needed to establish scalability. The approach discussed in [1] has several limitations. First, as already pointed, the use of the term domain is not appropriate for Grids. Hence we have modified the original formulation as shown in Section 5.2. Second, in case of a large number of organizations, it will be costly to compute the global trust (Equation 5.6) because we will have to consider all relationships to increase accuracy. To improve scalability, one can compute the global trust among a set of neighbors; however, such a computation would represent only trust between neighbors but not a global trust value. Third, the authors suggest in their study limiting the number of contexts on. Specifically, the authors reduced the number of contexts in the study to only three: printing, storage, and computing. In Grid environments, however, we deal with many more contexts. An example is the evaluation of trust and reputation for network characteristics, an essential part of any Grid infrastructure. Fourth, the function Υ, which depends on the duration of the interaction between two organizations, must be chosen carefully. We believe that for contexts such as file transfer, a time decay function may have to be chosen far larger than the longest file transfer to be considered, otherwise the decay function may invalidate the reputation even before the transaction is completed. Hence, it will be necessary to introduce classes of similar context, for example, for file transfers with different numbers of bytes. Another limitation is that in the case of networks the actual speed between resources could vary, making it even more complex to obtain the proper trust values.

We design a new algorithm, called Grid EigenTrust, that overcomes some of the limitations of these two approaches. We apply the EigenTrust algorithm explained in Section 5.1 to address the problems of scalability and multiple contexts; at the same time we introduce a global trust value based on the ability of institutions to maintain a trusted Grid environment and provide the high-performance community with reputation services.

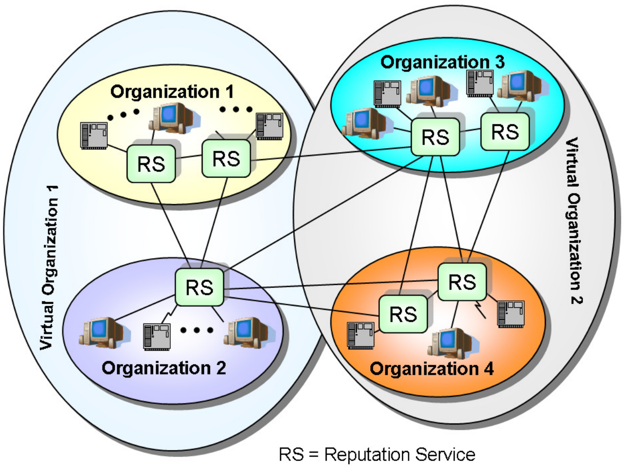

Fig. 6.1. *Example of a distribution of reputation management framework based on reputation services in a Grid.*

7. GridEigenTrust Algorithm. We address the complexity issue by introducing a set of reputation services arranged in hierarchical graphs. To illustrate this point, we consider the scenario shown in Figure 6.1.

In this scenario, two VOs are depicted containing two organizations each. Each organization has a set of entities. Hence, we have introduced an implicit hierarchy based on entities, institutions, and virtual organizations. We assign a reputation to the entities in the lowest level. Based on the reputation of the entities, the reputation of the organization gets updated. Finally we compute the reputation of a virtual organization by using the reputation values of all the organizations that belong to the virtual organization. Our reputation service can be reused and integrated in each level of the hierarchy.

The number of reputation services needed for a virtual organization or institution may vary based on its implicit size, determined by the entities and the hierarchy they define. Each reputation service is responsible for a subset of entities within the hierarchy. The reputation services compute the reputation in a collaborative, but distributed, fashion. Under the assumption that the interchange of reputation data is secure and can not compromised, and the time interval that a datum is valid is longer than the Smallest update, it may be possible to distribute previous reputation values from entities in the network in order to reduce the network overhead for lookups through a simple caching mechanism. In order to calculate and maintain the reputation, each reputation service uses the GridEigenTrust algorithm described in the next section. To guarantee accuracy, the reputation services must exchange messages with each other in a secure way and the semantics of the reputation service must be secured through a service signature that can be used to clearly identify wether the service has been tampered with.

7.1. Calculating Trust. To describe our GridEigenTrust algorithm, we use the notation used in Section 5.2. To simplify our discussion, we assume each entity is in only one organization (compare Section 3.1).

We establish a trust value for each entity based on various contexts it supports within an organization. We use the term *organization trust* to refer to a trust value for each organization. *Organization trust* differs from other context trust in that it agglomerates several context trust values to a single one. It reflects a general opinion of the reliability of an organization to provide accurate information on what resources this organization supplies. As a result, a reliability trust between organizations can be calculated quickly to obtain the global trust. Although this strategy sounds initially counterintuitive, it is often used in an economic model based on the trust model through branding.

By combining organization trust and the trust level of an entity within an organization (for a specific context c at time t), we derive a reliable trust value for the given entity. We apply the eigenvector mathematical model to compute the global reputation of an organization. Currently, we compute the reputation of a virtual organization as weighted sum of the reputations of all organizations that belong to the virtual organization.

7.1.1. Calculating the Trust of Entities. To describe how an organization maintains trust parameters of its entities, we modify the notation from Section 5.2. Since we are calculating trust values *locally*, (i.e. within an organization), we omit the first parameter in the function specification Θ, which denotes the entity from which the trust value was obtained.

All entities that use resources or collaborate with users within another organization grade the quality and reliability of the requested entity. The overall grade of the entity is established as the weighted sum of the previous grade (which decays with time) and the new grade. It is also important to consider how much we trust the organization from which the remote entity (i.e., entity that gives the grade) originates its requests.

If $\Theta_p(E_i, t_i, c)$ is the previous cumulative grade established at time t_i for entity E_i within context c, then $g_{ij}(t, c)$ is a new grade given by entity from organization O_j, and $T(O_j)$, then reliability trust level of organization O_j, is the overall new cumulative grade. Then, $\Theta(E_i, t, c)$ can be calculated as

$$\Theta(E_i, t, c) = \frac{\alpha(c) \cdot \Theta_p(E_i, t_i, c) \cdot \Upsilon(t - t_i, c) + \beta(c) \cdot T(O_j) \cdot g_{ij}(t, c)}{\alpha(c) + \beta(c)} \tag{7.1}$$

where $\alpha(c), \beta(c) \geq 0$.

Equation 7.1 is similar to Equation 5.5 from Section 5.2. However, the parameters $\alpha(c)$ and $\beta(c)$ reflect the context importance of the latest grade the entity received.

If an organization just joined the Grid, the initial trust values will be set to a low initial value because the trust must be earned first. However, if the entity for which we assign the trust is sufficiently similar to others in the already existing Grid, an initial value can be obtained from these integrated entities. We chose the lowest trust value and add as penalty a linear correction function.

Let $\Theta_0(E_i, t_0, c)$ denote the initial trust value for an entity E_i within our organization for a context c. Let $\Theta(E_i, t_i, c)$ denote the cumulative reputation value gathered from other entities (defined by equation (7.1)). Then the initial trust of the entity is the weighted sum of these two values:

$$\Gamma(E_i, t, c) = \frac{\gamma(c) \cdot \Theta_0(E_i, t_0, c) + \delta(c) \cdot \Theta(E_i, t_i, c)}{\gamma(c) + \delta(c)} \tag{7.2}$$

where $\gamma(c), \delta(c) \geq 0$.

7.1.2. Calculating the Reliability Trust between Organizations. The reliability trust of organization O_i toward organization O_j reflects *the opinion of organization O_i about the quality and trustworthiness of information organization O_j supplies*. Therefore, besides maintaining individual contexts, we introduce global context (compare Section 5.2). We use a similar notation as in Section 5.2, but we omit the parameter c. If we have a priori knowledge about the initial trust information, we assign this value at initialization time of our algorithm.

Let the initial value of trust be represented as $C(O_j)$. *Reliability* trust should be obtained through the weighted sum of direct experience and global trust value of organization O_j.

Direct experience can be calculated in the same way as in equation 7.1. It is a normalized weighted sum between $C(O_j)$, the cumulative grade from the previous period $\Theta_p(O_i, O_j, t_{ij})$ and the new grade $G(t)$.

Users within organization O_i grade the reputation of a certain entity E_j within organization O_j with grade $\Phi(E_j)$. Also, organization O_j advertises the quality-of-service of this entity with grade $\Delta(E_j)$. Then, organization O_i will grade reliability of information given by organization O_j with grade $G(t)$. For determining grade $G(t)$ we have three cases:

- If $\Phi \in [\Delta - \epsilon, \Delta - \zeta]$, the new grade $G(t)$ is 1.
- If $\Phi > \Delta - \zeta$, the new grade $G(t)$ is bigger than 1.
- If $\Phi < \Delta - \epsilon$, the new grade $G(t)$ is less than 1, depending on how much the Φ differs from Δ

Direct experience that organization O_i has with O_j at some time t, $\Theta(O_i, O_j, t)$ can be calculated in the same way as in equation 7.1. It is a normalized weighted sum between $C(O_j)$, cumulative grade from the previous period $\Theta_p(O_i, O_j, t_{ij})$ and the new grade $G(t)$.

$$\Theta(O_i, O_j, t) = \frac{\alpha \cdot C(O_j) + \beta \cdot \Theta_p(O_i, O_j, t_{ij}) \cdot \Upsilon(t - t_{ij}) + \gamma \cdot G(t)}{\alpha + \beta + \gamma} \tag{7.3}$$

where $\alpha, \beta, \gamma \geq 0$.

Global reliability trust of organization O_j, $\Omega(O_j,t)$ can now be calculated with the EigenTrust algorithm explained in Section 5.1. If we replace c_{ij} with $\Theta(O_i,O_j,t)$ in Section 5.1, we obtain a matrix $C=[\Theta(O_i,O_j,t)]$, and initial vector $\vec{T_0}=t_0(i)$, $t_0(i)=C(O_i)$. Now we have all the ingredients to apply a power iteration for computing the principal eigenvector of C^T, which represents global reliability trust values for organizations in Grids.

We can summarize the basic steps of the algorithm as follows:

Entity E_i within organization O_1 wants to use entity E_j within organization O_2 in the context c at time t.

- Consider the reliability trust of O_2 computed using the EigenTrust algorithm, $\Omega(O_2,t)$.
- Ask E_i about $\Gamma(E_j,t,c)$, the trust value of organization E_j within organization O_2.
- In calculating the overall trust value for entity E_j, in formula (5.4) replace $\Omega(E_j,t,c)$ with $\Omega(O_2,t)\cdot\Gamma(E_j,t,c)$
- Compute the overall trust for the entity $\Gamma(E_i,E_j,t,c)$ with formulas (5.4) and (5.5).

After computing the trust values, we can compare them to suggest the resource with the highest reputation. Modifications, such as the introduction of a statistical selection algorithm based on random variables, are possible.

This combined approach has several advantages. First, the algorithm converges rapidly and introduces less overhead than computing global trust values for individual entities within every context. One of the reasons is that the number of values for computation is not too large because we are computing global trust values of organizations through hierarchies, not an overall pool of individual entities. Second, organizations will make an effort to report accurate trust information about their entities because wrong information will be penalized, lowering the global trust of the organization.

8. Reputation Service Architecture. The architecture of an individual reputation service is shown in Figure 8.1. It consists of a collection manager, computation manager, storage and collection manager, and reporter. The collection manager is responsible for evaluating the quality statement describing the requested reputation, and collecting relevant data from the entities such as resources and users. It gives the collected data to the computation manager. The computation manager computes the reputation values of entities based on the context specified and gives the result to the storage manager, which stores the values to maintain a global and historical view. The reporter contacts the storage manager to report the reputation values whenever queried by some entity in the Grid.

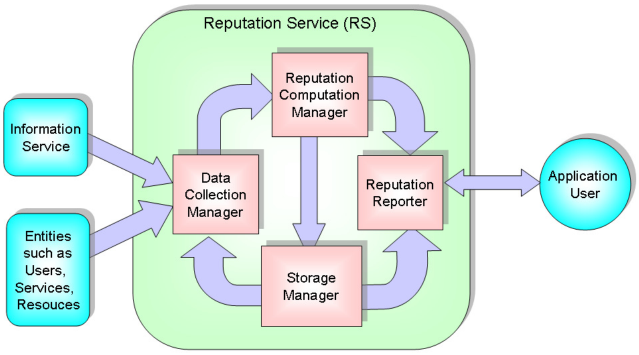

Fig. 8.1. *Architecture of a reputation service.*

Hence, when an application submits a request for a service cast in a qualitative statement to the reputation service, the reputation service evaluates the statement and computes the reputation for all the entities providing the required service using the heuristics explained in Section 7.1. It contacts other reputation services if required and returns the information regarding the services and their reputation back to the requester. The requester can decide to select the service by looking at the reputation values. This procedure can be easily modified for enabling and enhancing automating resource selection decisions in the Grid.

9. Conclusion and Future Work. In this paper, we have described a framework for calculating reputation in Grid-based system. The paper was mostly focused on issues that have to be addressed while working toward Grid services that integrate reputation concepts into their functionality. We have identified several of these issues. Second we have experimented with an architecture and algorithm to gain experience with this new area of research for the Grid community. We have identified a framework and algorithm, that is a combination of other research efforts. The underlying algorithm is based on introducing a global trust value that is updated with an eigenvalue based trust calculation algorithm. At present we are enhancing and evaluating our framework by introducing a variety of reputation measurements that are controlled through adaptive parameters.

Acknowledgment. This work was supported by the Mathematical, Information, and Computational Science Division subprogram of the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract W-31-109-Eng-38. DARPA, DOE, and NSF support Globus Alliance research and development. The Java CoG Kit Project is supported by DOE SciDAC and NSF Alliance. We thank Dr. Paul E. Plassman and Kaizar Amin for detailed and insightful comments on the paper. The algorithm described in this paper was first presented by Gregor von Laszewski in the Workshop on Adaptive Grid Middleware, in New Orleans, LA, September 28, 2003. A draft publication of part of the material that was presented is authored by B. K. Alunkal I. Veljkovic, G. von Laszewski, and K. Amin. This draft is available from `http://www.mcs.anl.gov/~gregor`

REFERENCES

[1] *Evolving and Managing Trust in Grid Computing Systems*, Hotel Fort Garry, Winnipeg, Manitoba, Canada, May 12-15 2002, IEEE Computer Society Press. Available from World Wide Web: `http://www.cs.mcgill.ca/~anrl/PUBS/ccece2002_farag.pdf`.

[2] R. Al-Ali, K. Amin, G. von Laszewski, O. Rana, and D. Walker, *An OGSA-based Quality of Service Framework*, in GCC2003, Shanghai, 2003. Available from World Wide Web: `http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--qos.pdf`.

[3] *amazon*. Web page. Available from World Wide Web: `http://www.amazon.com`.

[4] K. Amin, M. Hategan, G. von Laszewski, A. Rossi, S. Hampton, and N. J. Zaluzec, *GridAnt: A Client-Controllable Grid Workflow System*, in 37th Hawai'i International Conference on System Science, Island of Hawaii, Big Island, 5-8 Jan. 2004. Available from World Wide Web: `http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--gridant-hics.pdf`.

[5] F. Berman, G. C. Fox, and T. Hey, eds., *Grid Computing: Making The Global Infrastructure a Reality*, Wiley, 2003.

[6] S. Brin and L. Page, *The Anatomy of a Large-Scale Hypertextual Web Search Engine*, Computer Networks and ISDN Systems, 30 (1998), pp. 107–117. Available from World Wide Web: `http://www-db.stanford.edu/~backrub/google.html`.

[7] *Internetworking technology handbook, quality of service*. Web Page, visited Dec. 2004. Available from World Wide Web: `http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/qos.htm`.

[8] *C net*. Web Page. Available from World Wide Web: `http://www.cnet.com`.

[9] *ebay*. Web page. Available from World Wide Web: `http://www.ebay.com`.

[10] I. Foster and C. Kesselman, eds., *The Grid 2: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann Publishers, Dec. 2003.

[11] D. Gambetta, ed., *Trust*, Blackwell, 1990, ch. Chapter 4: Trust as a Commodity, pp. 49–72.

[12] T. H. Haveliwala and S. D. Kamvar, *The Second Eigenvalue of the Google Matrix*. Web page. Available from World Wide Web: `http://www.stanford.edu/~sdkamvar/papers/secondeigenvalue.pdf`.

[13] W. Heisenberg, *Uber quantentheoretishe umdeutung kinematisher und mechanisher beziehungen*, Zeitschrift fr Physik, 33 (1925), pp. 879–893.

[14] The International Association for Computers and Communications, *Integrating Trust into Grid Resource Management Systems*, Vancouver, B.C., Canada, Aug. 18-21 2002, IEEE Computer Society Press. Available from World Wide Web: `http://www.cs.umanitoba.ca/~anrl/PUBS/icpp2002_farag.pdf`.

[15] S. D. Kamvar, T. H. Haveliwala, and G. H. Golub, *Adaptive Methods for the Computation of Page Rank*. Web page. Available from World Wide Web: `http://www.stanford.edu/~sdkamvar/papers/adaptive.pdf`.

[16] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, *The eigentrust algorithm for reputation management in p2p networks*, in Twelfth International World Wide Web Conference, 2003, Budapest, Hungary, May 20-24 2003, ACM Press. Available from World Wide Web: `citeseer.nj.nec.com/article/kamvar03eigentrust.html`.

[17] A. Mowshowitz, *Virtual organization: a vision of management in the information age*, The Information Society, 10 (1994), pp. 267–288.

[18] G. von Laszewski and K. Amin, *Middleware for Commnications*, in Grid Middleware, Wiley, 2004, pp. 109–130. Available from World Wide Web: `http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--grid-middleware.pdf`.

[19] G. von Laszewski, S. Fitzgerald, I. Foster, C. Kesselman, W. Smith, and S. Tuecke, *A Directory Service for Configuring High-Performance Distributed Computations*, in Proceedings of the 6th IEEE Symposium on High-Performance Distributed Computing, 5-8 Aug. 1997, pp. 365–375. Available from World Wide Web: `http://www.mcs.anl.gov/~gregor/papers/fitzgerald--hpdc97.pdf`.

[20] G. von Laszewski, J. Gawor, C. J. Peña, and I. Foster, *InfoGram: A Peer-to-Peer Information and Job Submission Service*, in Proceedings of the 11th Symposium on High Performance Distributed Computing, Edinbrough, U.K.,

24-26 July 2002, pp. 333–342. Available from World Wide Web: `http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--infogram.ps`.

[21] G. von Laszewski and P. Wagstrom, *Gestalt of the Grid*, in Tools and Environments for Parallel and Distributed Computing, Series on Parallel and Distributed Computing, Wiley, 2004, pp. 149–187. Available from World Wide Web: `http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--gestalt.pdf`.

[22] G. von Laszewski, M. Westbrook, I. Foster, E. Westbrook, and C. Barnes, *Using Computational Grid Capabilities to Enhance the Ability of an X-Ray Source for Structural Biology*, Cluster Computing, 3 (2000), pp. 187–199. Available from World Wide Web: `http://www.mcs.anl.gov/~gregor/papers/vonLaszewski--dtrek.pdf`.

Edited by: Wilson Rivera, Jaime Seguel.
Received: July 8, 2003.
Accepted: September 1, 2003.