Supporting Formation and Operation of Virtual Organisations in a Grid Environment

Jianhua Shao, W Alex Gray, Nick J Fiddian
Vikas Deora, Gareth Shercliff, Patrick J Stockreisser
School of Computer Science, Cardiff University

Timothy J Norman, Alun Preece, Peter M D Gray, Stuart Chalmers, Nir Oren
Department of Computing Science, University of Aberdeen

Nicholas R Jennings, Michael Luck, Viet D Dang
Thuc D Nguyen, Jigar Patel, W T Luke Teacy
School of Electronics and Computer Science, University of Southampton

Simon Thompson
BT Exact Technologies, Adastral Park

Abstract

The ability to create reliable and scalable virtual organisations (VOs) on demand in a dynamic, open and competitive environment is one of the challenges that underlie the Grid concept and research. In this paper, we describe the agent-based mechanisms that we are currently developing within the CONOISE and CONOISE-G projects for supporting VO formation and operation.

1 Introduction

In a Grid environment, there can exist a large number of service providers who are autonomous and offer some computational or other problem-solving capabilities. It is widely considered the case that some service providers may wish to team up, at some point in time, to form an alliance or a virtual organisation (VO), in order to respond to or exploit a particular market opportunity. The ability to create reliable and scalable VOs on demand in a dynamic, open and competitive environment is therefore desirable, and is one of the challenges that underlie the Grid concept and research [11, 10].

In this paper, we describe the agent-based techniques that we are currently developing within the CONOISE[1] and CONOISE-G projects for supporting effective VO formation and operation. The CONOISE project focuses primarily on VO formation and we have developed a model which is based on three key technologies: the decision-making mechanism of an individual service provider, an auction mechanism for the allocation of contracts, and a service discovery mechanism incorporating quality of service (QoS) assessment [14]. The CONOISE-G project, on the other hand, seeks to support robust and resilient VO operation by investigating how the policing within a VO may be supported, the members' trust and reputation may be established, and the quality of their service provision may be monitored.

In the following sections, we will first introduce a specific VO formation and operation scenario. We will then illustrate, using the scenario as an example, the challenges associated with the task of supporting VO formation and operation. Finally, we will describe how the techniques that we are currently developing help to achieve effective VO formation and operation within a Grid environment.

2 A Motivating Scenario

To help illustrate VO formation and operation, and to highlight the challenging issues involved in creating and operating a VO, we introduce a specific scenario in this section.

[1]CONOISE = Constraint-Oriented Negotiation in an Open Information Services Environment. Information about both CONOISE and CONOISE-G is available at *http://www.conoise.org*.

Table 1: Potential Service Providers

Service Provider	Entertainment (mins/day)	News (updates/day)	Text (msgs/month)	Game Clips (mins/day)	Ticketing (alerts/day)
SP1	30	20			5
SP2		10	50		
SP3			100	30	5
SP4	30	10		60	
SP5			50	45	10

Suppose that someone visits London, say in 2012 for the Olympic Games. Whilst in London she wishes to make use of her PDA/phone to take full advantage of various multimedia services on offer, for example, to keep in touch with the latest world and local news, to watch clips from the games, to locate ticket opportunities for the events, to use a text messaging service and to take advantage of ad-hoc entertainment opportunities (such as streaming video). Typically, there can exist many service providers who offer the services that she requires, so our visitor will need to find the potential providers (from the many), select an optimal package (from the available offers), and then keep tracking the ever-changing market for possible better deals. These are non-trivial tasks.

The idea of creating a VO on demand is to help service requesters in such situations, so that they only need to specify their service requirements, and VOs will be created automatically to provide them with the required service(s). However, forming and operating a VO is not a simple task and a number of important issues must be taken into account. To illustrate these issues, let us consider the following specific case. Suppose that we have five providers (`SP1`–`SP5`), as shown in Table 1, who offer multimedia services. These services form three groups: *video content* (Entertainment and Game Clips services), *HTML content* (News and Ticketing services) and *text messaging* (Text service); and they can be requested individually or taken as a package, with the exception that the two services offered by `SP2` must be taken together.

We assume that the providers of these services may demand different prices for the same service, depending on the number of units requested. For example, `SP1` may offer 20 news updates per day at £30 per month, and 10 updates per day at £25 per month. We also assume that the quality of services may not be stable. For example, `SP4` may claim that it can offer game clips with a frame rate at no less than 24 frames per second, but in the actual service provision it may in fact drop below that level. Finally, we do not assume that all our service providers are entirely trustworthy, thus what they claim may not be what a service requester will get. For example, `SP5` may often advertise itself in possession of some sought-after tickets for the popular events, but time will perhaps tell that many such claims are not genuine and orders for tickets through `SP5` may not always be honoured.

With these characteristics in mind, let us now suppose that our London visitor wishes to purchase the following service package:

Service Required	Units Required
Entertainment	50 mins per month
News	10 updates per day
Text messages	100 per month
Game Clips	60 mins per day
Ticketing	10 alerts per day

Obviously, many solutions to this request are possible, using the providers available in Table 1. For example, to offer 50 mins of entertainment as requested, both `SP1` and `SP4` must be used, but different compositions of the two services are possible. While they will all offer a combined 50 mins of entertainment, each can have a different price, quality and degree of trust. Thus, to find an optimal solution for a given service request, some rational decisions will have to be made.

Generally speaking, during VO formation, there may be multiple service providers available who may offer broadly similar services. The services themselves are described by multiple attributes, for example, price, quality, reputation and delivery time. We therefore need to consider how the relevant services may be discovered for a given service request, and how an optimal package may be selected for the requester, based on various considerations such as price, quality and provider reputation.

During VO operation, on the other hand, the services available may change over time: new services may become available, or providers may alter the way in which existing services are offered. The quality of services and their providers' reputation may also change over time. Therefore, there is a need to monitor the performance of the members of a VO in terms of their trustworthiness, quality of service and conformance to contract, and to re-structure the VO when necessary so that the integrity and usefulness of the VO are maintained: a poorly performing service may be replaced, a contract breaking service may be dropped, and a new user requirement may be accommodated.

So, creating a VO automatically, then managing it effectively in a dynamic environment is a significant research challenge. Any solution to this problem must ensure that the formation of a VO is rational and is optimised w.r.t. some given criteria, and its operation is robust and resilient, even when disruptive and potentially malicious entities exist. In the next section, we will discuss these issues more fully, and will describe how the techniques we are currently developing in the CONOISE and CONOISE-G projects help to address these issues.

3 Supporting VO Formation and Operation

In the CONOISE and CONOISE-G projects, we are developing an agent-based model for VO formation and operation. In our model, the following computation is performed in the forming and operating of a VO.

Assuming that service providers have already advertised their services to a Yellow Pages agent, the whole VO formation process starts with the Requirements agent which analyses the requester's service requirements, locates the relevant providers through the Yellow Pages agent, and then invites the identified providers to bid for the requested services. The quality and trustworthiness of the received bids are then assessed by the QoS agent and the trust component, respectively, and the outcome of this assessment is combined with the price structure to determine which combination of the services/providers will form an optimal VO to serve the requester.

Once a VO is formed, the Requirements agent takes on the role of VO Manager (VOM), responsible for ensuring that each member of the VO provides its service according to its contractual requirements. The VOM, through the QoS monitoring component and the Policing agent, will closely monitor the quality of the services offered by the members of the VO and the implementation of their contractual obligations. If any fall in service quality is identified or predicted, or some breach of contract is observed, the VOM will start a VO re-formation process, and relevant information will be fed into the trust component to ensure that the provider concerned is penalised to an appropriate level by updating its record of trust.

The following sections will discuss some of the main technical issues involved in our VO formation and operation model in some detail.

3.1 Service Discovery

In an open, dynamic environment such as the Grid, autonomous service providers may join and leave at any time. It is necessary therefore that the participants' behaviour is informed by exploiting diverse forms of information, such as advertisements and ontologies, and their capabilities are discovered dynamically at the time when a service request is made. Various service description languages and matchmaking mechanisms have been proposed to enable this, e.g. [21, 1], but so far little support has been given to incorporating QoS assessment in service discovery.

In our model, we extend the current approaches to service discovery by creating service and quality ontologies, allowing service providers and consumers to advertise their capabilities and request services using the two ontologies, and dynamically matchmaking between advertised and requested services based on functional as well as QoS requirements [8]. More specifically, we extend DAML-S [1] to include QoS specifications and our matchmaking follows a two-stage process. First, a service request containing functional and/or QoS requirements is sent to the Yellow Pages agent to search for providers who claim to be able to offer the required service. Then, the QoS agent is asked to assess how well each identified provider can actually provide the service at the quality level they claim, using a novel expectation based quality calculation model [7]. The outcome of this assessment is then used to weed out the providers whose QoS levels are too low and to provide another basis for negotiation.

3.2 Decision Making in VO Formation

In a VO environment, a resource (or company providing the resource) is represented by a service provider agent. This agent must check, when asked to contribute a bid for the provision of service, how much of its resources are currently being used (by prior commitments as a result of already successful bids), and decide if it can make an offer to provide the new service. It may also look at the collective resources available if it is in an existing VO, and make a new bid on the VO's behalf. Alternatively, it may decide that the provision of this new resource is more beneficial to it than its prior commitments and decide to break some or all of these to allow the successful creation of a bid for the new service provision. It also has the option of forming a new VO to cater for the provision of the resources. These options are explained in Figure 1 below.

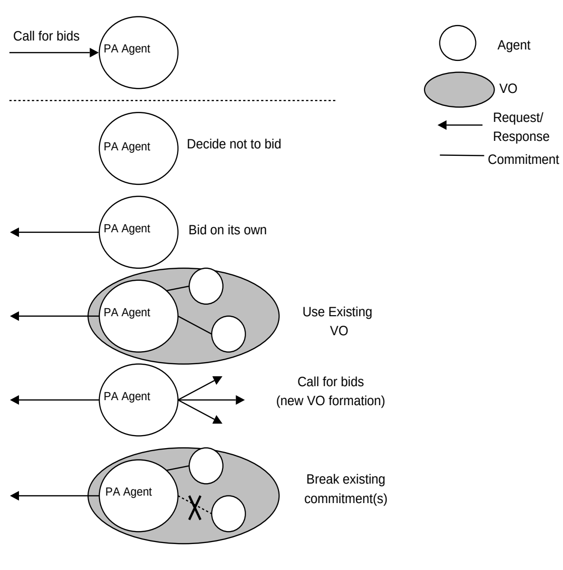

Figure 1 Decision Making Options

As can be seen, this decision making process can be quite complex. We use a Cumulative Scheduling algorithm based on Constraint Satisfaction Programming (CSP) techniques [3] to aid the agent in this process.

The scheduling algorithm models the agents available resources using two metrics: the length of time the agent can provide the resource for and the amount of resource available over that time. It then models the existing resource provision by constraining these metrics. The remaining free resources represent the units that can be used by the agent to construct its bid. To model the agent's ability to discard existing resources in favour of new resources, we use constraint reification [4]. This attaches a true/false reified value to each existing commitment and attempts to find a subset of true reified values which will allow the scheduling algorithm to satisfy the time and resource metrics plus the new provision of resource. The existing commitments not in this satisfied subset will then represent the set which needs to be broken in order for the new commitment to be successful.

When a service provider decides that it is beneficial to bid for the provision of a certain task, it will forward the bid to the Requirements agent who initiated the call for bids. Since mutiple bids for the same request are possible, the bids received from the service providers must be *cleared*. That is, we must decide which ones to accept (or which partners to select) in the formation of the VO. Given the open nature of the environment and the lack of a pre-ordained structure, we believe this selection process is best achieved using some form of marketplace structure (auction). Two sets of clearing algorithms have been developed: one with polynomial complexity and has been shown to produce a solution that is within a finite bound of the optimal, while the other is not polynomial but is guaranteed to produce the optimal allocation [5].

3.3 Establishing Trust and Reputation

Whenever interactions take place between different agents, the issues of trust and reputation become important. Questions of deception and fraud in communication and interaction, of assurance and reputation, and of risk and confidence, are particularly significant, especially where interactions take place with new partners. In particular, during the formation of a VO, we often have a choice of service providers to whom we may delegate tasks. Trust serves as an indicator of which of these possible partners are likely to carry out the task as specified, and its usefulness extends into the other stages of the VO lifecycle. It reduces the uncertainty in the decision making process for an agent.

Many sociologists have carried out research on the ideas behind trust within human society [13, 12]. For our research, we define trust to be: *"a particular level of the subjective probability with which an agent assesses that another agent will perform a particular action, both before she can monitor such action and in a context in which it affects her own action"* (adapted from [12]).

This probabilistic view of trust allows us to determine the *subjective probability* by considering the outcomes of previous encounters (known as direct interaction-based trust). However, in an open community it is likely that an agent will interact with many unknown entities with which it may not share an interaction history. In the absence of this shared history, the CONOISE-G trust system uses *reputation* information to establish the level of trust to place in another. Reputation can be defined as *a commonly held set of opinions about an entity* [19], and it is the aggregation of these common opinions that forms a level of trust.

We have designed mechanisms that allow agents in the CONOISE-G system to calculate a level of trust based both on direct experience and on opinions provided by others when the agents share no direct interaction experiences.

The trust and reputation system consists of two distinct parts. The first part is a trust component, which is internal to all agents that require a trust metric in their decision-making process. Its function is to provide its owner agent with a level of trust ($T \in [0, 1]$) for a given service and service provider. The component is insulated from the external environment by the agent that embodies it. As the agent interacts with others in the community the outcomes of these interactions are stored in this component, and are used to determine a trust value when required. Outcomes can either be successful or unsuccessful, where a successful interaction is defined as one for which the service provider has delivered the service specified by the contract. In addition to calculating T, the trust component calculates a level of *confidence* C to be placed in the value of T. Confidence represents the accuracy of T and is obtained by examining how much evidence was used to calculate T. It is used by the trust component to reason about whether an agent has adequate evidence or whether it needs to obtain further (reputation) information from other agents. When the confidence in its own calculation does not exceed a particular threshold, the trust component requests such reputation information from others. In our model, we do not assume that reputation is necessarily accurate, and we allow for the possibility that an agent may intentionally supply misleading information; the trust component will assess the likelihood that a reputation provider will supply accurate information, based on the accuracy of information it supplied in the past.

The second part of the trust system is a *reputation brokering* agent, and several of these agents in the system may serve as a distributed store of reputation information. A reputation broker agent provides an aggregated store of trust information relating to specific service provider agents and each of their services. However, before any agent can query the broker, the broker must obtain the trust information that will form the query result. We propose to achieve this using a *subscribe and publish* mechanism, by which the broker subscribes to agents in the community which then publish their internal information (the store of outcomes based on their individual direct experiences) to the broker. Agents in the community can obtain reputation information from these brokers by sending query messages, to which the brokers can reply with the relevant information or a failure message in the case where they do not have such information.

The main contribution of this work is a model of trust and reputation that allows agents to make trust-based decisions to take into account the uncertainty in opponent behaviour and, as a result, helps to assure good interactions in open systems like the Grid. The Grid exacerbates these problems due to its distribution and scale, and due to the large volume of agents and the complexity of interactions between them. Additionally, the self-interest of the many and varied organisations in the Grid can cause problems of strategic lying and collusion amongst agents, which have been identified as still unsolved issues [17]. Addressing this is integral to the wide-scale acceptance of the Grid and agent-based virtual organisations.

At present, the implemented system includes the trust component and the reputation broker agents, and it uses a trust ontology that forms the basis of the trust-related communication between agents in order to exchange trust information. Further work will aim to undertake experimentation in order to fully evaluate our approach and extend the basic trust model to address the Grid-specific problems identified above.

3.4 Policing within a VO

Although the modelling and establishment of trust and reputation are essential, this is not sufficient for the prevention and detection of fraudulent transactions. The goal of policing is to determine whether a party is in breach of contract. The grid-based nature of the project means that the scalability of a policing system is a critical issue, and a complaint-initiated investigation system is thus envisaged. The policing agent will gather evidence from the various agents involved in the contested transaction, and identify those agents it believes are guilty of an infraction to the trust agent.

We are examining a number of open questions related to policing within the context of CONOISE-G. In the short term, we are looking at extending a logic for contract representation [9] such that the generated contracts can be reasoned about by agents. These extensions are required due to the prevalence of VOs in our environment which means that standard two party contracts are limited in their representative power.

VOs and the delegation of responsibility also complicate the blame assignment procedure. By incorporating work by authors such as Norman [15] and Pacheco [16], together with our contract representation language, we intend to be able to apportion blame in an intuitively fair manner.

When dealing with both deceitful agents and a partially observable environment, determining when failure occurs, and reasoning about who to blame for a failure becomes very difficult. Daskalopulu et al.[6] use subjective logic in an attempt to reason about such issues, but a richer form of reasoning is required when agents are allowed to undermine each others actions. We intend to utilize argumentation theory [18] to allow agents to present evidence and counter-evidence in response to a questioning policing agent in an attempt to prove their guilt or innocence. The policing agent will then be able to weigh up the evidence so as to perform blame apportionment.

3.5 Monitoring QoS Levels

During the operation of a VO, it is important that the quality of service provision is monitored. The QoS data collected from this monitoring process is vital in supporting the creation of a resilient VO in the CONOISE environment. First, it serves as "evidence" in a range of critical assessment. For example, the QoS data is used by the trust component to establish the level of trust that one may place in a service and service provider; by the Policing agent to deal with complaints; and by the QoS agent to assess QoS for services during future VO formations. Second, the QoS data helps monitor and predict any QoS degradation within a VO. Any detection or prediction of such degradation will result in a possible replacement of a VO member, or trigger a re-formation of the VO, thereby ensuring that the VO will maintain an agreed level of QoS provision and will limit possible damages to its reputation.

We are currently examining the issues relating to the monitoring of QoS for service provision, and are developing a QoS monitoring component called QoS Consultant (QoS-C) within our model. The QoS-C is an infrastructure service which can either monitor service provision on behalf of a VOM (to monitor the quality of services offered by its members), or sample the quality of a service itself. The latter is particularly important when addressing the "cold start" problem, i.e. when we have a new service provider joining the market and we do not have sufficient knowledge of the quality of its service provision.

The QoS-C component consists of three processes. The first process entails the recording and gathering of QoS data, which is a continuous activity that will populate a QoS database. This will be performed through the use of network sensors. For simplicity, we assume that the QoS at any point on the link from the provider to the consumer is equal. The second process involves the monitoring of the QoS level, using the data that has been collected in the QoS database. This is to calculate the current level of service provision and compare it to the minimum level of QoS required in the service level agreement. Any service whose QoS has dropped below the level required will then be reported to the VOM concerned. We are currently attempting to develop some robust, scalable monitoring techniques to handle the potentially huge set of QoS data. The final process to be performed by the QoS-C component is that of alerting the VOM of any anticipated drop in QoS, the "prediction". It is on the successful prediction of QoS degradation that we will focus our efforts in the near future.

In the current Grid research, there exists an extensive range of studies on resource and QoS prediction and forecasting techniques, ranging from predictive resource reservation to forecasting computational performance [20], the majority of them are associated with attempting to maximise or guarantee the quality of the Grid computing infrastructure services themselves, and little has been done to help predict QoS levels for user services. From our service-centric viewpoint of quality we see potential in the application of prediction and forecasting techniques at the service layer. We intend to develop time series based analysis techniques [2] for analysing the QoS data and predicting the QoS trend at the service layer in a Grid environment.

4 Conclusions

In this paper we outlined our approach to supporting robust and resilient VO formation and operation. The main contribution made by this research lies in the development of a range of advanced agent-based techniques, and their integration in a coherent VO management system. We believe that the models, theories and techniques we are developing within the CONOISE and CONOISE-G projects are of significant importance to the Grid research, and our model can be deployed not only to support e-Science applications, but also in realistic electronic commerce scenarios. We have already constructed a prototype for demonstrating the formation of a VO based on the techniques outlined in this paper. Currently, we are working on the techniques for supporting the effective operation of a VO, and are integrating them into the prototype.

References

[1] A. Ankolekar, M. Burstein, J. Hobbs, O. Lassila, D. Martin, S. McIllraith, S. Narayanan, M. Paolucci, T. Payne, K. Sycara, and H. Zeng. Daml-s: Web service description for the semantic web. In *Proceedings of International Semantic Web Conference (ISWC)*, pages 348–363, 2002.

[2] G. Box, G. Jenkins, and G. Reinsel. *Time Series Analysis: forecasting and control.* Prentice-Hall, 1994.

[3] Y. Caseau and F. Laburthe. Cumulative scheduling with task intervals. In *Logic Programing Proceedings of the 1996 Joint International Conference and Syposium on Logic Programming*, pages 363–377, 1996.

[4] S. Chalmers, A. Preece, T. J. Norman, and P.M.D. Gray. Commitment management through constraint reification. In *to appear in Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi- Agent Systems*, 2004.

[5] V. D. Dang and N. R. Jennings. Polynomial algorithms for clearing multi-unit single item and multi-unit combinatorial reverse auctions. In *Proceedings of the 15th European Conf. on AI (ECAI-2002)*, pages 23–27, 2002.

[6] A. Daskalopulu, T. Dimitrakos, and T. Maibaum. Evidence-based electronic contract performance monitoring. *Group decision and negotiation*, 11(6):469–485, 2002.

[7] V. Deora, J. Shao, W. A Gray, and N. J Fiddian. A quality of service management framework based on user expectations. In *Proceedings of the First International Conference on Service Oriented Computing (ICSOC)*, pages 104–114, 2003.

[8] V. Deora, J. Shao, G. Shercliff, P. J. Stockreisser, W. A Gray, and N. J Fiddian. Incorporating qos specification in service discovery. Technical report, CONOISE Project, 2004.

[9] V. Dignum, J. J. Meyer, F. Dignum, and H. Weigand. Formal specification of interaction in agent societies. In *Proceedings of the second Goddard workshop on formal approaches to agent based systems (FAABS)*, 2002.

[10] I. Foster, N. R. Jennings, and C. Kesselman. Brain meets brawn: Why Grid and agents need each other. In *to appear in Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi-Agent Systems*, 2004.

[11] I. Foster, C. Kesselman, and S. Tuecke. The anatomy of the Grid: Enabling scalable virtual organizations. *International Journal of Supercomputer Applications*, 15(3):200–222, 2001.

[12] D. Gambetta. Can we trust trust? In *Trust: Making and Breaking Cooperative Relations*, chapter 13, pages 213–237. Basil Blackwell, 1988.

[13] B. Misztal. Trust in modern societies: The search for the bases of social order, 1996.

[14] T. J. Norman, A. Preece, S. Chalmers, N. R. Jennings, M. Luck, V. D. Dang, T. D. Nguyen, V. Deora, J. Shao, W. A. Gray, and N. J. Fiddian. CONOISE: Agent based formation of virtual organisations. In *Proceedings of the 23rd SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence*, pages 353–366, 2003.

[15] T. J. Norman and C. Reed. Group delegation and responsibility. In *Proceedings of AAMAS 2002*, pages 491–498, 2002.

[16] O. Pachheco and J. Carmo. A role based model for the normative specification of organized collective agency and agents interaction. *Autonomous Agents and Multi-Agent Systems*, 6:145–184, 2003.

[17] S. Ramchurn, D. Huynh, and N. R. Jennings. Trust in multi-agent systems. In *The Knowledge Engineering Review*, 2004.

[18] C. Reed and T. J. Norman, editors. *Argumentation Machines, New frontiers in argumentation and computation*. Kluwer, 2003.

[19] J. Sabater and C. Sierra. Regret: A reputation model for gregarious societies. In *Fourth Workshop on Deception Fraud and Trust in Agent Societies*, pages 61–70, 2001.

[20] W. Smith, I. Foster, and V. Taylor. Scheduling with advanced reservations. In *Proceedings of the 14th Parallel and Distributed Processing Symposium*, 2000.

[21] WSDL. Web services description language. http://www.w3.org/TR/wsdl.