Provenance-based Trust for Grid Computing — Position Paper —

Luc Moreau[1], Syd Chapman[2], Andreas Schreiber[3], Rolf Hempel[3], Omer Rana[4], Lazslo Varga[5], Ulises Cortes[6], Steven Willmott[6]

[1] School of Electronics and Computer Science, University of Southampton, Southampton, SO17 1BJ, United Kingdom
[2] IBM United Kingdom Ltd, Hursley Park, Winchester, SO21 2JN, United Kingdom
[3] German Aerospace Center (DLR), Linder Hoehe, 51147 Koln, Germany
[4] School of Computer Science, University of Wales, 5 the Parade, CF24 3XF, Cardiff, United Kingdom
[5] MTA SZTAKI, Kende u. 13–17, 1111 Budapest, Hungary
[6] Software Department (LSI-AI), Universitat Politecnica de Catalunya, 20 Jordi Girona, 08034 Barcelona, Spain

l.moreau@ecs.soton.ac.uk, syd_chapman@uk.ibm.com, Andreas.Schreiber@dlr.de, Rolf.Hempel@dlr.de, O.F.Rana@cs.cf.ac.uk, laszlo.varga@sztaki.hu,ia@lsi.upc.es,steve@lsi.upc.es

Abstract. Current evolutions of Internet technology such as Web Services, ebXML, peer-to-peer and Grid computing all point to the development of large-scale open networks of diverse computing systems interacting with one another to perform tasks. Grid systems (and Web Services) are exemplary in this respect and are perhaps some of the first large-scale open computing systems to see widespread use - making them an important testing ground for problems in trust management which are likely to arise. From this perspective, today's grid architectures suffer from limitations, such as lack of a mechanism to trace results and lack of infrastructure to build up trust networks. These are important concerns in open grids, in which "community resources" are owned and managed by multiple stakeholders, and are dynamically organised in virtual organisations. *Provenance* enables users to trace how a particular result has been arrived at by identifying the individual services and the aggregation of services that produced such a particular output. Against this background, we present a research agenda to design, conceive and implement an industrial-strength open provenance architecture for grid systems. We motivate its use with three complex grid applications, namely aerospace engineering, organ transplant management and bioinformatics. Industrial-strength provenance support includes a scalable and secure architecture, an open proposal for standardising the protocols and data structures, a set of tools for configuring and using the provenance architecture, an open source reference implementation, and a deployment and validation in industrial context. The provision of such facilities will enrich grid capabilities by including new functionalities required for solving complex problems such as provenance data to provide complete audit-trails of process execution and third-party analysis and auditing. As a result, we anticipate that a larger uptake of grid technology is likely to occur, since unprecedented possibilities will be offered to users and will give them a competitive edge.

1 Introduction

The Grid [10] is a very large scale computer system which is capable of coordinating resources that are not subject to centralised control, whilst using standard, open, general-purpose protocols and interfaces, and delivering non-trivial qualities of service [9]. Grids are therefore likley to become one of the largest classes of open computing systems and provide us with a well advanced architectural basis for assessing the type of trust management problems which might arise. As part of the endeavour to define the Grid, a service-oriented approach has been adopted, by which computational resources, storage resources, networks, programs, libraries and databases are all represented by services [11]. In this context, a service is a network-enabled entity capable of encapsulating diverse implementations behind a common interface. A service-oriented view is powerful since it allows the composition of services to form more sophisticated services. The service-oriented Open Grid Service Architecture (OGSA) defines a *Grid Service* as a Web Service [3] that follows specific conventions and provides a set of well-defined interfaces [11].

In the "Anatomy of the Grid", Foster, Kesselman and Tuecke describe the problem underlying the Grid concept as coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organisations [12]. While the underpinning mechanisms for creating and managing such virtual organisations still remain to be understood, effort is required to allow users to place *their trust* in the data produced by such compositions. Understanding how a given service is likely to modify data flowing into it, and how this data has been generated, is crucial as illustrated by the following generic question:

> *In an open grid environment, let us consider a set of services that decide to form a virtual organisation with the aim to produce a given result; how can we determine the process that generated the result, especially after the virtual organisation has been disbanded?*

Against this background, *provenance* is an annotation able to explain how a particular result has been derived; such provenance information can be used to better identify the process that was used to reach a particular conclusion.

Provenance is therefore important to enable a user to trace how a particular result has been arrived at, and the sequence of steps that are involved. Specifically, we consider the specific notion of execution provenance, which identifies what data is passed between services, what services are available, and how results are eventually generated for particular sets of input values. Using execution provenance, a user can trace the "process" that led to the aggregation of services producing a particular output.

It is our belief that provenance support should be part of a grid infrastructure, so that users can put their trust into such a new paradigm. The purpose of this position paper is to present a *research agenda for trust-based provenance*. This paper is organised as follows. We review background work on provenance in Section 2. We then present the desired characteritics of a provenance architecture, which would allow users to trust a grid computing environment (Section 3). We then discuss three grid applications, which would benefit from provenance and trust in Section 4. We then discuss why provenance offers a good approach for establishing trust in open environments (Section 5) before concluding the paper with Section 6.

2 Background on Provenance

The vision of the Grid as an open environment in which collaborations are dynamically negotiated and organised with the goal to produce some specific results has inevitably resulted in the concern that users need to be able to trust results produced by such computations. This motivated two workshops on provenance [19, 20]. The idea of providing provenance is relatively new and unexplored. So far, work on provenance has mainly identified uses, properties and requirements of provenance in multiple application domains.

In modern information systems, data can be collated from a variety of distributed and diverse resources and processed to form new data. We can view the sequence of taking a dataset, processing it and producing a new dataset as a dataset transformation. In order to provide provenance, all datasets and their transformations must be recorded. Saltz [23] suggests that we can achieve a sound lineage record by recording enough information to ensure that any dataset transformation is reproducible. Goble also presents some notable uses of provenance: reliability and quality; justification and audit; re-usability and reproducibility; change and evolution [15]. The storage and maintenance of provenance records is an important consideration. Frew and Bose [13] propose the following requirements for provenance collection. *(i)* A standard lineage representation is required so data lineage can be communicated reliably between systems (currently there is no standard lineage format). *(ii)* Automated lineage recording is essential since humans are unlikely to record all the necessary information manually. *(iii)* Unobtrusive information collecting is desirable so that current working practices are not disrupted.

In the context of databases, the provenance of a specific piece of data identifies parts of the database that contributed to it. Buneman *et al.* [4, 5] distinguish between "why" provenance (the set of tuples that contribute to the result) and "where" provenance (the location(s) in the source database from which the result was extracted). They formulate a precise definition of provenance using a general data model that applies to both relational databases and hierarchical data structures such as XML.

Some authors are starting to investigate architectural support for provenance. The Chimera Virtual Data System [8] comprises a virtual data catalogue, for representing data derivation procedures and derived data, with a virtual data language interpreter that translates user requests into data definition and query operations on the database. The explicit representation of data derivation provides a documentation of data provenance, which can be used to audit and trace the lineage of derived data produced by computation.

Szomszor and Moreau [24] propose a provenance recording capability for service-oriented architectures such as the Grid and Web Services. They offer a Web Service to record provenance information and to view and retrieve provenance. In particular, they provide a provenance-based result-validation mechanism by which provenance is used to determine whether previous computed results are still up to date.

None of these approaches regards the problem of provenance generation as a collaborative activity between multiple parties. The implication is that, in such systems, provenance data cannot be trusted or audited by third parties since provenance is only provided by one component (typically a workflow enactor), without any verification that execution took place in the way reported by that very component. It is this specific aspect that will be studied in the recently UK-funded e-Science project PASOA (www.pasoa.org) led by Southampton and Cardiff, also authors of this paper. In PASOA, the focus is on the theoretical underpinning and algorithmic foundations of provenance generation and reasoning.

In order to advance the state of the art in grid computing, it is essential to provide *industrial strength* provenance support when running complex applications. Industrial strength provenance support includes the following key aspects:

1. A scalable architecture capable of sustaining high volumes of data, complex workflows with large number of services, and a high number of requests for navigating/reasoning over provenance;
2. A secure architecture relying on industrial standards for security to promote inter-operability, and ensuring that provenance information is securely managed;
3. Standardisation of protocols and data structures to promote inter-operability of components provided by multiple manufactures or institutions;
4. A set of tools for configuring and using the provenance architecture;
5. Deployment and validation in industrial context.

By achieving these goals, a provenance infrastructure will become an essential building block of a Next Generation Grid [1], which will help users to trust the results delivered by the grid paradigm.

3 A Provenance Architecture

Service composition and orchestration have been identified as key objectives for the Grid (and Web Services) communities [14]. In particular, workflow engines allow users to identify, choose and compose services based on their own particular interests. Workflow based computations can be seen as a simplified (and tractable) form of virtual organisation, scripted explicitly using workflow languages such as BPEL4WS [7], WSFL [17], or XLANG [25].

A preliminary architecture capable of generating provenance and reasoning over it is sketched in Figure 1. First, provenance gathering is a collaborative process that involves multiple entities, including the workflow enactment engine, the enactment engine's client, the service directory, and the invoked services. Provenance data will be submitted to one or more "provenance repositories" acting as storage for provenance data. Upon user's requests, some analysis, navigation and reasoning over provenance data can be undertaken. We foresee here that storage could be achieved by a provenance service, and that a library, optionally hosted in the provenance service, would perform the analysis, navigation or reasoning.

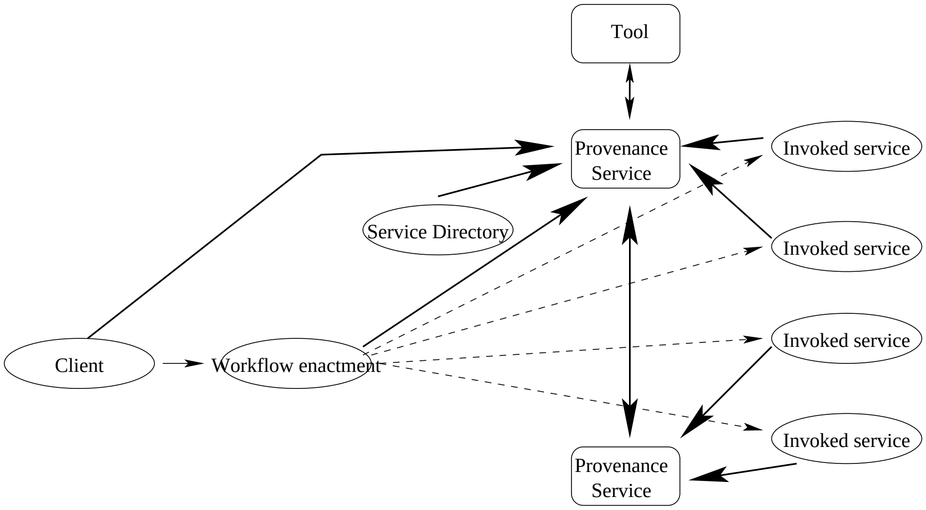

Fig. 1. Provenance Architecture

Coordination is needed between the different entities involved in workflow enactment. All entitities have to agree on the repositories in which provenance data is to be stored. The current enactment should also be identified in a unique manner, and this identification should be shared by all entities involved in it, so that provenance pertaining to a given execution is not stored in the repository agreed for another execution. As provenance data may become very quickly huge, the level of details to be recorded may be agreed before a session is started.

In order for provenance data to provide information that is trustworthy, we expect such a protocol to support some "classical" properties of distributed algorithms. For instance, using *mutual authentication*, an invoked service can ensure that it submits data to a specific provenance server, and vice-versa, a provenance server can ensure that it receives data from a given service. With *non-repudiation*, we can retain evidence of the fact that a service has committed to executing a particular invocation and has produced a given result. We anticipate that cryptographic techniques will be useful to ensure such properties. Such techniques are usually regarded as rather expensive, and we may not like the process of provenance generation to hinder the progress of workflow execution. In some cases, it may be useful to generate provenance data in a manner that is *asynchronous* to workflow execution. Additionally, it may not be realistic for all parties to submit provenance data to a single store, as it would become a bottleneck. Instead, multiple provenance stores may be desirable to store provenance data, on a temporary or long-term basis. Finally, if all parties submit provenance information, some redundancy may occur: indeed, in normal circumstances, A's account of an invocation of B should match B's perception of it; hence, some parties may be required to submit full provenance data, while others may only submit summaries. Therefore, asynchronous processing, multiple store configuration, and submission details will have to be agreed by all parties in a set-up phase.

Having stored the provenance information pertaining to a given workflow execution, we need to provide facilities to access, navigate, analyse and reason over such data. Some of this functionality can be generic such as navigation of raw data or filtering by activity type. More complex functionality, but still generic, includes a *provenance-based result-validation capability* [24], which can decide if the results logged in a provenance trace are still up to date. Such a capability can be beneficial to users who have run a workflow and need to decide if they have to re-enact the workflow because some of its services now produce different results. Finally, domain specific provenance reasoning may also be required, as illustrated by some of the applications we discuss in Section 4.

In order to design, conceive and implement an industrial-strength open provenance architecture for Grid computing, the following steps must be undertaken:

1. To specify the contents of provenance in relation to workflow enactment.
2. To design and implement a scalable and secure distributed co-operation protocol to generate provenance data in workflow enactment.
3. To conceive and implement tools to navigate, harvest and reason over provenance data, also in a scalable and secure manner.
4. To design and engineer a scalable and secure software architecture to support provenance generation and reasoning.
5. To contribute to the standardisation effort in this area within the Grid and Web Services communities by writing a standardisation proposal on provenance.

Specifically, this would require the implementation of the architecture and tools, including: *(i)* Implementation of a scalable and secure distributed provenance service; *(ii)* Implementation of a navigation/analysis/reasoning tool; *(iii)* Client-side and server-side libraries handling provenance generation during services invocations; *(iv)* Implementation of a tool for configuring and managing the system.

4 Three Grid applications for Provenance

In this section, we introduce three grid-based applications, with essential needs for provenance, but also with very different requirements. We also discuss the kind of trust that can be derived from provenance in these contexts.

4.1 Application 1: Aerospace Engineering

The SISTEC group at DLR (German Aerospace Center) are involved in developing workflow based approaches for combining components that undertake scientific simulation, data pre- and post-processing and visualisation. Each of these involves complex software packages, some of which require specialised hardware resources to execute. Some of these packages are developed in-house, but others are obtained from a number of different vendors and consortium partners. The workflow must support both static, pre-defined interactions between such components, and in some cases real-time interactions to support "computational steering". The TENT system at DLR is an example of such a system, which utilises distributed object technologies to couple such software components.

Provenance is crucially required in this context, as the need to maintain a historical record of outputs from each sub-system is an important requirement for many customers that utilise the end result of simulations. Associating provenance information with the workflow engine itself is also useful, as information about aircraft structures developed as a consequence of this work needs to be maintained over long time periods. For instance, aircrafts' provenance data need to be kept for up to 99 years when sold to some countries. Currently, however little direct support is available for this, and today's methodology for producing this information is ad-hoc.

4.2 Application 2: Organ Transplant Management

E-Health is a major application area both for grid technology and provenance solutions. Medical information systems, databases and in particular decision support systems [6] rely on a wide range of data sources, human input and access to patient data. In many cases, domains are highly regulated, must retain careful audit data and rely heavily not only on information in the system but knowledge added by doctors, surgeons and other individuals using the systems. Organ and tissue transplant processes are examples of such medical information systems and are are characterised by the following constrains:

- European, national, regional and site specific rules govern how decisions are made; furthermore, the application of these rules must be ensured, be auditable and both rules and application may change over time;
- Patient recovery is highly dependent not only on the organ allocation choice but subsequent extraction and insertion methods as well as the care/recovery regime (while much is understood about certain types of transplants, many

elements of post transplant care and the relationship of organ/tissue acceptance rate to the match made as well as the care applied require much more detailed study);
- Patient records, organ/tissue bank databases and other information are distributed across a number of sites (many large cities in Europe have multiple donor and transplant sites).

Current organ transplant systems are very far from grid ready (most interactions are still done by phone between different sites). But in the long-term, we expect such systems to:

- Link up all the tissue banks, organ recipient lists, emergency centres, etc. held at different hospitals and link the decision support systems which guide the allocation process;
- Connect allocation mechanisms across regional and state boundaries to ensure that all EU, national and regional regulations are rigorously enforced in the process;
- Maximise the efficiency in matching and recovery rate of patients.

This application will benefit from grid technologies because there are a large number of patient record sites, tissue banks and other databases in the region and in general data cannot be sent and cached (due to confidentiality and size). Also computation is very complex: surgeons should match over about 50 dimensions, for e.g. a cornea, but tend to just use 4 because the reasoning becomes too complex and effects are not understood.

In this application, the major provenance problems are:

- Tracking back previous decisions in any one centre to identify "whether the best match was made" (verifying/proving this and generating an explanation), who was involved in the decision, what was the context.
- Aggregating partial results from searches in different centres and applying the rules that apply between centres. Maintaining the validity of partial results.

These two uses of provenance require some specific understanding of the application domain.

4.3 Application 3: Bioinformatics Grid

The project myGrid aims at providing a personalised environment for bioinformaticians to perform *in silico* experiments [18]. In order to demonstrae Grid-based bioinformatics, myGrid focuses an application that helps scientists study Graves' Disease, which is a hormonal disorder caused by over-stimulation of the thyrotrophin receptor by thyroid-stimulating autoantibodies secreted by lymphocytes of the immune system. In order to study the Graves' Disease, a scientist would follow an *in-silico* experimental process typical of bioinformatics, according to which he or she: *(i)* attempts to discover information about candidate genes; *(ii)* makes an educated guess of the gene involved in the disease; and *(iii)* designs an experiment to be realised in the laboratory in order to validate the guess. This *in-silico* experiment operates over the Grid, in which resources are geographically distributed and managed by multiple institutions, and the necessary tools, services, and worflows are discovered and invoked dynamically. It is a *data-intensive* Grid, where the complexity is in the data itself, the number of repositories and tools that need to be involed in the computations, and the heterogeneity of the data, operations and tools.

Concretely, myGrid provides a form of "lab book" environment, in which the e-Scientist can construct and discover *in silico* experiments; the lab book also keeps the scientist informed about the provenance of the data visible in their experimental space.

In myGrid, provenance is stored in a user's personal repository and provenance generation is tightly integrated with the enactment engine [16]. In that context, the focus is not on the architecture and protocols required for supporting provenance in service-oriented architectures, but on personalising provenance information when presented to the scientist. Personalisation of provenance data may require an understanding of the application domain, so that a suitable level of abstraction of provenance is presented to the scientist.

4.4 Provenance Based Trust

Automatic recording of provenance is a facility that does not exist in current grids, in which ad-hoc, semi-automated and human efforts are used today to attempt to achieve such functionality. We anticipate that automatic provenance tracking in grid systems will give a competitive edge to early adopters of such technology; it will also enable further use of complex service compositions since it allows users to trust the results produced.

Specifically, for the applications we discussed in this section, new levels of trust will be achievable, namely *confidence in the quality of results*, *certification*, *validation of correctness*, *auditing*, *studying and analysing complex processes*, which we now discuss:

- In aerospace engineering and related fields where safety-critical systems are developed, simulations in every stage of the product development cycle are becoming more and more complex. For example, complex multidisciplinary coupled simulations or optimizations are being performed in distributed computing environments.

  In order to get some level of *confidence* in the quality of the simulation results, it is important to collect the complete history of the computation. The collected provenance information is also an important part of the information that airplane manufacturers have to provide for the *certification* process of new airplanes.
- Healthcare domains are highly regulated and audit trails are essential for almost all aspects of any healthcare process: information about cases must be carefully managed, decision processes and decision making must be carefully

tracked. Often only partial views of data will be available to any particular actor in the scenario and both data and decision making is distributed across multiple sites. Furthermore, regulations change from medical specialty to medical specialty, from region to region and from country to country.

  The development of generic provenance services would enable a far greater number of these processes to be automated (by providing the necessary auditing guarantees) and would provide a much higher degree of *validation of correctness* of electronic systems. Building provenance-like services into bespoke systems is likely to be both very costly and error prone, whereas re-use and customisation would both improve overall quality and encourage correct modelling of *auditing* systems in new healthcare applications.

- Pharmaceutical companies are required by governmental agencies (such as the Food and Drug Agency) to provide supporting documentation of drug discovery as part of the drug application process. Provenance records provide a log of the in-silico experimental process, which an important component of this documentation. It allows such agencies to study, audit and analyse the processes involved the drugs design and establish its safety. Supporting documentation must be submitted in pre-defined formats, based on which domain-specific provenance analysis tools can generate the required information.

5 Discussion

Provenance identifies the process that was used for producing some data. It crucially helps users decide whether to *place their trust in the results* obtained by composing dynamically discovered services in open grids. To become trustworthy, data provenance must be unforgeable, must unambiguously authenticate the services that were involved in a process, and where appropriate should retain non-repudiable evidence of their involvment.

Given the provenance of data, we anticipate that *trust metrics* of such data can be derived from the trust a user places in the services involved in the data production, from the workflow use, and from the provenance service that recorded provenance data. For instance, in bioinformatics, users prefer to use some specific databases because they trust the provider more, or because the provider has better reputation. Hence, if such services were involved in a computation, the final outcome is likely to be more trustworthy to them. Reputation is defined as "the opinion or view of one about something". Several models of reputation have been proposed [26, 22]. They rely on a social network that structure the reputation knowledge that components have of their neighbours. According to [21], the meaning of reputation in this context is an aggregation of opinions of some or all agents about one of them over a particular issue. Given the reputation (whether computational or informal) of services involved in a computation, given the workflow that was executed, given the reputation of the provenance service, and given the provenance of the data resulting from the computation, some form of metrics can be devised to reflect the trust that the user will be able to place of the result. We expect a compositional mechanism to be used to compute such metrics.

If suitable protocols are in place to provide unforgeable evidence of execution, users will have to trust provenance services to execute such protocols properly, to archive provenance traces without altering them, and to execute queries over them correctly. We anticipate provenance services to be hosted by reputable providers, who have the incentive, financial or other, to behave according to the public specification of provenance protocol. The protocol should also be open to ensure that the process adopted to record provenance is also well understood by all parties.

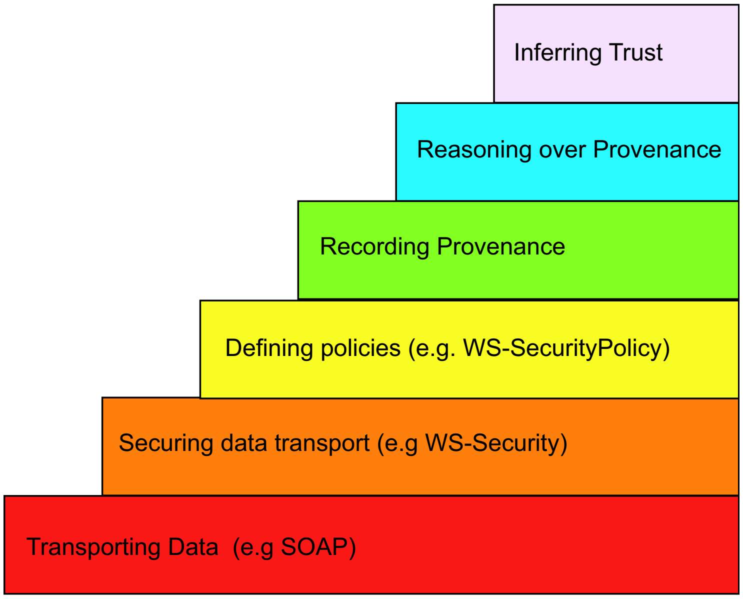
Inferring Trust
Reasoning over Provenance
Recording Provenance
Defining policies (e.g. WS-SecurityPolicy)
Securing data transport (e.g WS-Security)
Transporting Data (e.g SOAP)

Fig. 2. Provenance-based Trust Wedding Cake

Consequently, our architectural vision can be summarised by the different layers of Figure 2, a refinement of Tim Berners-Lee's Wedding Cake for the Semantic Web [2]. Given the process-oriented nature of provenance, our architectural vision emphasizes processes rather than data or knowledge formats. Our architectures identifies different layers, from communications to trust inference, and it makes explicit the processes of recording and reasoning provenance in a context of policy-based secure communications.

The appeal of a provenance-based trust mechanism is that it is founded on a computationally sound mechanism to record execution of workflows and virtual organisations in grid environements. Its implications are far reaching and well beyond computer science.

1. The legal value of provenance data should be assessed. Even more importantly, as we are still at the stage of requirement capture, elliciting legal requirements that would apply to provenance data can drive the technical design of such infrastructure.
2. Alternatively, the social implication of such system should not be neglected. While Section 4 presented the benefits of provenance in concrete and important applications, there is also a downside to it. A provenance system acts as a generalised form of audit — in other words, a "big brother" watching all the time over our shoulders. Issues of privacy or confidentiality should also be considered. This social dimension introduces further requirements that will have to be met by a provenance architecture.
3. In any discipline where "processes" matter, a provenance system will provide an auditable log of the operations that took place. We can anticipate this can be beneficial to protect intellectual property, to patent processes, or symmetrically to maintain archives for future reference.

6 Conclusion

In this position paper, we have presented a research manifesto for provenance-based trust in grid environments. Our vision is based on the principle that all entities involved in dynamic computations over open grids should contribute to the recording of provenance, so that the origin of results computed by such computations can be established. This would provide us with an unequalled foundation for assessing the trustworthyness of these results.

7 Acknowledgement

This research is funded in part by EPSRC myGrid project (reference GR/R67743/01) and PASOA GR/S67623/01.

References

1. H. Bal, C. de Laat, S. Haridi, K. Jeffery, J. Labarta, D. Laforenza, P. Maccalum, J. Masso, L. Matyska, T. Priol, A. Reinefeld, A. Reuter, M. Riguidel, D. Snelling, and M. van Steen. Next Generation Grid(s). European Grid Research 2005-2001. Expert Group Report, June 2003.
2. Tim Berners-Lee. Semantic web. Presentation at XML 2000, `http://www.w3c.org/2000/Talks/1206-xml2k-tbl/slide10-0.html`.
3. David Booth, Hugo Haas, Francis McCabe, Eric Newcomer, Michael Champion, Chris Ferris, and David Orchard. Web services architecture, August 2003. W3C Working Draft.
4. Peter Buneman, Sanjeev Khanna, and Wang-Chiew Tan. Why and Where: A Characterization of Data Provenance. In *International Conference on Database Theory (ICDT)*, 2001.

5. Peter Buneman, Sanjeev Khanna, and Wang-Chiew Tan. Computing provenance and annotations for views, October 2002. Published at [19].
6. U. Cortés, A. López-Navidad, J. Vázquez-Salceda, A. Vázquez, D. Busquets, M. Nicolás, S. Lopes, F. Vázquez, and F. Caballero. Carrel: An agent mediated institution for the exchange of human tissues among hospitals for transplantation. Technical Report LSI-00-33-R, Software Department. UPC, http:/www-lsi.upc.es/-dept/techreps/ps/R00-33.ps.gz, 2000.
7. Francisco Curbera, Yaron Goland, Johannes Klein, Frank Leymann, Dieter Roller, Satish Thatte, and Sanjiva Weerawarana. Business process execution language for web services (bpel4ws). http://www.ibm.com/developerworks/library/ws-bpel/, 2002.
8. I. Foster, J. Voeckler, M. Wilde, and Y. Zhao. Chimera: A virtual data system for representing, querying and automating data derivation. In *Proceedings of the 14th Conference on Scientific and Statistical Database Management*, Edinburgh, Scotland, July 2002.
9. Ian Foster. What is the grid? a three point checklist. `http://www-fp.mcs.anl.gov/~foster/`, July 2002.
10. Ian Foster and Carl Kesselman, editors. *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufman Publishers, 1998.
11. Ian Foster, Carl Kesselman, Jeffrey M. Nick, and Steven Tuecke. The Physiology of the Grid — An Open Grid Services Architecture for Distributed Systems Integration. Technical report, Argonne National Laboratory, 2002.
12. Ian Foster, Carl Kesselman, and Steve Tuecke. The Anatomy of the Grid. Enabling Scalable Virtual Organizations. *International Journal of Supercomputer Applications*, 2001.
13. James Frew and Rajendra Bose. Lineage issues for scientific data and information, October 2002. Published at [19].
14. Grid computing environments working group at the global grid forum. http://www.computingportals.org/, November 2002.
15. Carole Goble. Position statement: Musings on provenance, workflow and (semantic web) annotations for bioinformatics, October 2002. Published at [19].
16. Mark Greenwood, Carole Goble, Robert Stevens, Jun Zhao, Matthew Addis, Darren Marvin, Luc Moreau, and Tom Oinn. Provenance of e-science experiments - experience from bioinformatics. In *Proceedings of the UK OST e-Science second All Hands Meeting 2003 (AHM'03)*, pages 223–226, Nottingham, UK, September 2003.
17. Frank Leyman. Web Services Flow Language (WSFL). Technical report, IBM, May 2001.
18. Luc Moreau, Simon Miles, Carole Goble, Mark Greenwood, Vijay Dialani, Matthew Addis, Nedim Alpdemir, Rich Cawley, David De Roure, Justin Ferris, Rob Gaizauskas, Kevin Glover, Chris Greenhalgh, Peter Li, Xiaojian Liu, Phillip Lord, Michael Luck, Darren Marvin, Tom Oinn, Norman Paton, Stephen Pettifer, Milena V Radenkovic, Angus Roberts, Alan Robinson, Tom Rodden, Martin Senger, Nick Sharman, Robert Stevens, Brian Warboys, Anil Wipat, and Chris Wroe. On the Use of Agents in a BioInformatics Grid. In Sangsan Lee, Satoshi Sekguchi, Satoshi Matsuoka, and Mitsuhisa Sato, editors, *Proceedings of the Third IEEE/ACM CCGRID'2003 Workshop on Agent Based Cluster and Grid Computing*, pages 653–661, Tokyo, Japan, May 2003.
19. Data provenance/derivation workshop. `http://people.cs.uchicago.edu/~yongzh/position_papers.html`, October 2002.

20. Data provenance and annotation workshop. http://www.nesc.ac.uk/esi/events/304/, December 2003.
21. Sarvapali D. Ramchurn, Nicholas R. Jennings, Carles Sierra, and LLuis Godo. A computational trust model for multi-agent interactions based on confidence and reputation. In *Proc. 6th Int. Workshop of Deception, Fraud and Trust in Agent Societies*, Melbourne, Australia, 2003.
22. J. Sabater and C. Sierra. Regret: A reputation model for gregarious societies. In *First international joint conference on Autonomous Agents and Multi-Agent systems (AAMAS'02)*, pages 475–482, 2002.
23. Joel Saltz. Data provenance, October 2002. Published at [19].
24. Martin Szomszor and Luc Moreau. Recording and reasoning over data provenance in web and grid services. In *International Conference on Ontologies, Databases and Applications of SEmantics (ODBASE'03)*, volume 2888 of *Lecture Notes in Computer Science*, pages 603–620, Catania, Sicily, Italy, November 2003.
25. Satish Thatte. XLANG: Web Services for Business Process Design Author. Technical report, Microsoft, 2001.
26. Bin Yu and Munindar P. Singh. Detecting deception in reputation management. In *Second International Joint Conference on Autonomous Agents and Multi-Agent systems (AAMAS'03)*, 2003.