Towards Believing Answers from Cyber-Infrastructure-based Applications

Gilbert Ornelas, Paulo Pinheiro da Silva
Computer Science Department, the University of Texas at El Paso
{gtornelas, paulo}@utep.edu

Abstract

Within a cyber-infrastructure (CI), there is increasing collaboration between services as well as increasing use of multiple information sources with different level of quality by these services. This suggests that service results might come from multiple sources including other services. Thus, it is becoming important for CI-based services to provide some mechanism for computing belief recommendations if users are going to accept and use service results. In this paper, we discuss the combined use of provenance information, i.e., information describing how and where the results were derived, and trust relation networks between CI agents to compute belief recommendations. Agents may use these trust-based belief recommendations to build opinions about the trustworthiness of service results and confidently accept, further question or reject them.

Keywords: cyber-infrastructure, trust, provenance, web services, belief recommendations

1. Introduction

Cyber-infrastructures (CIs) have become an important part of the way scientists are conducting research. CIs enable scientists to build complex applications by seamlessly creating, using and reusing distributed resources such as services and tools. Along with increased collaboration between services, there is increased use of multiple information sources by CI-based applications and services. This suggests that information might be coming from sources that are potentially unknown to scientists and that may intentionally or unintentionally provide information that has discrepancies or is inaccurate. Thus, it is becoming more important for CI-based applications to have some kind of tracking mechanism indicating where results computed by a service were derived from in terms of data and methods used. Such provenance information can be used to compute and thus provide recommendations about the trustworthiness of final service results, and about the trustworthiness of any intermediary results generated during the processes of manipulating data.

Computing the trustworthiness of results requires a web of trust relations between agents and information sources as well as among agents themselves. Along with provenance information, this web of trust can be used to support multiple ways of computing belief recommendations based on agent-specific ways of representing trust including the notions of mistrust, distrust, inconsistencies and ignorance like in [2, 4, and 5]. Such strategy to support trust management within a CI, as described in this paper, is important because it enables agents to decide how much to believe in service results. This gives agents more control over which results to confidently accept or which results to further question or reject.

This paper is organized as follows. Section 2 presents related work. Section 3 provides a real application motivational scenario based on CI resources to illustrate the need of trust within CIs. Finally, section 4 presents the use of the strategy to implement a trust management solution for the scenario described in section 3.

2. Related Work

Previous research efforts have dealt with encoding provenance information and including trust values along with answers returned by a service.

The Inference Web (IW) framework in [3] proposes a framework for the Semantic Web aimed at explaining answers from the Web. Here, provenance information that consists of inference rules, proofs, conclusions, and other metadata about information sources is also encoded in PML as a justification to a user's answer. The IW framework itself however, does not provide a way for encoding trust values along with an answer.

The IWTrust framework, which is used as the base of the prototype implemented in this paper, is an extension to the IW framework. In this extension, in addition to the provenance information provided with each answer, a trust value is also attached to that provenance information [2]. Although its TrustNet component enables trust to be expressed between users, services, and information sources, the framework is not geared specifically towards Web services.

3. GeoTrust: A CI-Based Scenario for Trust

The Geosciences Network (GEON) is a National Science Foundation-funded cyber-infrastructure designed to advance the field of geoinformatics through furthering research and education in the geosciences. Through its service-oriented architecture, GEON provides a single access point to geological data repositories and software tools distributed across the country that can be combined to manipulate these data and obtain meaningful results.

Geoscientists are often interested in putting together complex workflows, which describe a process of ordered steps taken to complete a task within GEON. Geoscientists are ultimately interested in getting results from their scientific tasks. However, like other scientists, geoscientists must be sure that they understand and believe the final results of CI-based services. For instance, geoscientists must know where results came from, how reliable they are, and whether or not to accept, further question, or reject them.

GeoTrust is a testbed scenario based on GEON that is being developed at UTEP to study trust relations between geoscientists and geological data repositories and to research different belief recommendation models that represent these. At a top level, the scenario consists of a geoscientist accessing the GEON portal to obtain a contour map given a set of longitude and latitude points defining the area for which the contour is to be constructed. The scenario is illustrated in Figure 1.

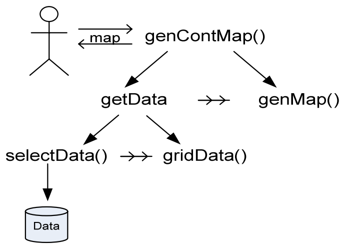

Figure 1. GeoTrust Scenario

In the scenario, the main application to be called is *genContMap()*, taking as parameters two longitude and latitude coordinates – area within which mapping data would be collected. In GEON and other CIs, an application often makes use of several other services and accesses to remote data repositories to complete its task. In the case of *genContMap()*, it invokes the *getData()* and *genMap()* services sequentially. *getData()* is responsible for retrieving the data. *genMap()* generates the actual contour map by using the output of *getData()*. Like *genContMap()*, *getData()* is also a complex task specified as a sequential composition of the *selectData()* and *gridData()* services – the first service is used to select remote data and the second service is used to convert the selected data into a format called grided format.

In the GeoTrust scenario, several trust-related questions arise that a geoscientist may ask once she obtains the contour map as a result. Which information sources, i.e., data repositories, have been used to generate the map? Who owns the information sources? How much should I trust these sources and believe their contents? How much can I trust the services and their developers? How much can I believe the contour map? These are all questions whose answers would increase the confidence of a geoscientist in a final result and would allow her to develop an opinion about the results.

A geoscientist may have a hard time believing a map that was manipulated by unknowledgeable people, if the developers of the main application have a low degree of trust on intermediate services, or if the geoscientist herself does not trust the services. GeoTrust keeps a database of trust relations among geoscientists and between geoscientists and information sources. Tables in the database define different trust relationships between pairs of geoscientists, services, and sources. As section 4 explains, these trust relations will be used to derive result trust values and group them with provenance information upon returning final results.

4. A Strategy for Implementing Trust Management Solutions in GEON

The implementation of a trust management layer for GEON services that combines provenance information and a web of trust relations would provide agents with a sense of assurance and facilitate the exchange of derived information among them.

4.1. Provenance Information and PML

Being able to express provenance information along with results produced by these services is the first step towards developing a trust-based belief recommendation infrastructure. Thus, a format is needed that can encode and provide structure to such information. The Proof Markup Language (PML) provides a proof Interlingua designed to represent justification of information manipulation steps [1]. PML encodes provenance information such as source information, reasoning and retrieval information as well as proof-generation information. Further, PML is represented in XML format which, given that services communicate through XML-based documents, would facilitate its integration with service results.

Using PML would solve the problem of expressing provenance information along with service results within CIs. However, expressing this provenance information does not provide a way of expressing a degree of trust along with a returned result. Even if such a result includes PML documentation to show exactly how it was derived, it does not provide any information about the trustworthiness of such result. Thus, extending PML to include trust representations seems to be the normal next step that would provide the base of our solution.

4.2. Trust Relations

A trust relation from agent A to agent B is unidirectional and represents a degree of trust that A has on B. Trust can be propagated in several ways and no single approach has shown to be appropriate for most cases. Thus, flexible ways of representing trust are needed to represent trust relations between agents. A PML document must convey the type of trust relationship between agents collaborating to form an answer including the use of agents as authors/owners of documents. Revealing this information through a PML document would allow an agent not only to follow exactly the origin and manipulation steps of data, but also to identify trust relationships that existed between services and information sources along the way. A basic PML trust relation has the following attributes:

- *hasTrusteeParty*: it is a registered information source. If the source is an agent then the relationship is representing a typical trust relation between agents. If the source is a document then the degree of trust represented in the relation needs to be interpreted as a degree of belief than the trusting party has on the document.
- *hasTrustingParty*: it is another registered information source. It must be an agent. If the agent is an individual then the relationship is representing a personalized trust relation possible involving more than one level of trust propagation. If the agent is an organization or a named group of individuals then the relationship is representing an aggregated trust relation.
- *hasContext*: identifies the subject field where the trust relation is valid. Indeed, A may trust B for buying a house but A may not trust B for recommending a movie.

Optional attributes for trust relations are not described here. Few additional required attributes are described below.

4.3. Flexible Use of Multiple Models of Trust

Since belief-recommendations would be derived solely from trust recommendations, one of the properties that the solution being suggested in this paper has is its flexibility to work with multiple already existing or proposed models of trust. The trust recommendation is computed according to the way specified in the trust model. PML along with trust relations provide the infrastructure for applying the trust model of choice. The specification of the trust model and values for trust relations are provided by three additional required attributed for trust relations:

- *hasTrustModel*: A pointer to the model describing the correct interpretation of trust values as well as the valid set of operations used to propagate and aggregate trust.
- *hasTrustValue1* (and *hasTrustValue2*): a bi-lattice representing most aspects of trust has proved to be a minimum required space for representing trust concerns in a trust-provenance-preserving way [4]. Thus, these trust values should be enough for interpreting different trust models.

A provenance-preserving trust model is one that preserves information related to how trust values were derived along the computation of a result. Such model is important because users have access to trust-related information that is usually lost in a non-preserving trust model and that can be used as a trust recommendation for a service(s). In CIs like GEON these recommendations can then be used as a degree of belief about a derived service result. This section suggests a possible solution to the trust management problem within CIs through the prototype implementation of TrustNet as well as through a proposed extension of the PML Interlingua to include trust information.

4.4. Computing Belief Recommendations

To enable trust-based belief recommendations of CI-based services, propagation of provenance information must be supported. That is, provenance information and trust relations expressed by intermediary services must also be integrated into the PML information that will be returned with a final service result. Further, operations that define propagation and aggregation of trust must be defined and integrated into the trust model. The specification the IWTrust framework already provides a foundation for propagating and aggregating trust relations and for using these relations to derive belief relations. Different than IWTrust, the work proposed here is flexible on how trust can be represented, propagated and aggregated. Also, the approach is new in the sense that methods are CI-enabled services and question-answering is achieved by the execution of scientific workflows.

The IWTrust framework enables users to access sources used in answer computation along with trust values for those sources [2]. IWTrust consists of several components such as a proof fragments and queries component, a component named IWBase and a component named TrustNet. The proof fragments and queries component uses PML to trace information manipulation. IWBase is a distributed repository of proof and metadata information used to build the PML justifications. Recalling the attributes inside trust relations, "registered" sources are sources registered within IWBase. Conceptually, TrustNet is a graph, where nodes represent the users, services and sources while edges represent trust values between them. However, it is not specified or implemented in the context of Web services.

In general, an agent may or may not know how much to trust a service, thus does not know how much to believe a result provided by it. In figure 2, the geoscientist is requesting map (request Q) from service *genContMap()*. The geoscientist may not know that *genContMap()* is calling service *getData()*. If the geoscientist can figure out that *getData()* has been used, he may not know how much to trust service *getData()*. However, that geoscientist has a degree of trust in another user who in turn has a degree of trust in service *getData()*. This relationship can now be used by the geoscientist requesting the map to establish a degree of trust in *getData()*. Eventually the user requesting the map Q can obtain a trust recommendation along with the result from *genContMap()* that is based on the provenance information and the trust degrees from both *getData()* and *genMap()*. This recommendation can be used by the geoscientist to establish an opinion about how much he may believe the map generated by *genContMap()*.

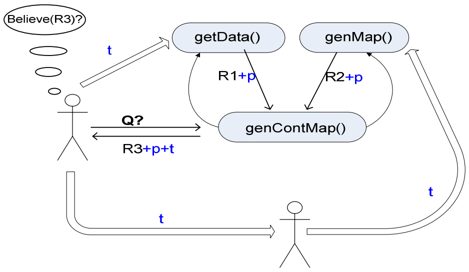

Figure 2. General trust-based scenario

Alternatively, it is necessary for a service to be able to compute a belief recommendation given the trust recommendation from another service. As opposed to an agent (i.e. a geoscientist) that can decide how much to believe a result based on a trust value associated with it, a service needs to consult a function that maps such trust value to a belief recommendation that also takes into account factors such as the domain within which trust is taking place.

5. Conclusion

If the use of services within CIs as well as the collaboration among them is to increase, there needs to be a mechanism to provide trust management among them, the information sources they use, and the agents requesting the service. This paper presented a scenario and a strategy to implement trust management solutions within a CI by enabling trust computation components to use multiple models for representing, propagating and aggregating trust, extending the PML interlingua to include trust relations along with provenance information, and prototyping the TrustNet component of the IWTrust framework to support the extended PML trust relations. The example of the GEON cyber-infrastructure provided a real-life application where trust is needed in order to provide certainty and assurance about the results produced by a service.

6. Acknowledgements

This work is funded in part by the National Science Foundation (NSF) through grant NSF EAR-0225670.

7. References

[1] P. Pinheiro da Silva, D.L. McGuiness, and R. Fikes, "A Proof Markup Language for Semantic Web Services", Knowledge Systems Laboratory, Stanford University, Tech. Rep. KSL-04-01, 2004.

[2] I. Zaihrayeu, P. Pinheiro da Silva, and D.L. McGuiness, "IWTrust: Improving User Trust in Answers from the Web", Informatica e Telecommunicazione, University of Trento, Italy, Tech. Rep. DIT-04-086, 2005.

[3] D.L. McGuinness, and P. Pinheiro da Silva, "Explaining Answers from the Semantic Web: the Inference Web approach", *Journal of Web Semantics*, Vol.1 No.4, October 2004, pp. 397-413.

[4] P. Victor, M. De Cock, C. Cornelis, and P. Pinheiro da Silva, "Towards a Provenance-Preserving Trust Model in Agent Networks", WWW2006, Edinburgh, UK, May 22-26

[5] A. Josang, "Multiplication and Comultiplication of Belief", *International Journal of Approximate Reasoning*, Vol. 38 No. 1, March 2004, pp. 19-55