Toward Integrating Social Trust into Web Service Compositions

Sharon M. Paradesi and Prashant Doshi
LSDIS Lab, Department of Computer Science
University of Georgia
Athens, GA 30602
paradesi, pdoshi}@cs.uga.edu

Abstract
Web 2.0 helped bring about the development of social communities, Web services and a host of applications. Web service compositions traditionally utilize the functional and quality-of-service parameters of candidate services to decide which services to include in the composition. However, users of services often form an opinion of a service. User-driven composition techniques will form compositions that likely behave as stated in practice, and which are better received by the users. We develop a preliminary framework for integrating social trust in Web service compositions and experimentally validate the utility of our trust framework.

Introduction
Web 2.0 helped bring about the development of social communities, a host of applications and, more importantly, e-commerce and other web services. Web service compositions traditionally utilize functional and quality-of-service (QoS) parameters of candidate services to decide which services to include in the composition (Zhao and Doshi 2007). This is a straightforward way of selecting services, which allows the composition to theoretically meet its goal and satisfy non-functional requirements. However, users of services often form an opinion, somewhat subjective, of a service (analogous to other Web 2.0 applications). This opinion may be based on prior interactions with the service, and may include judgments such as whether the perceived behavior of the service conforms to its stated behavior and intangibles such as the overall experience of the user with the service.

We present a simple method by which trust values are progressively updated over time based on interactions with multiple users. We show how trust may be aggregated into a single value for different types of compositions having sequential, concurrent, and other types of flows. Finally, we empirically validate our approach.

Formulation of Trust in Web Services
Trust of a requester, B, on a Web service, w, is the confidence of B on w’s competency, reliability and honesty, which will make B depend on w to perform the actions on which B’s welfare depends, even though negative consequences are possible.

We may objectively measure honesty of a service as the difference between the advertised or agreed upon values of QoS parameters appearing in the service level agreements and the actual observed values of the QoS parameters. As cost is not observed, we limit our focus to the parameters, response time \( \bar{R} \) and availability (A). The response times are normalized, as shown below, to make them comparable with availability. Let \( A_a \) and \( R_a \) be the advertised values while \( A_m \) and \( R_m \) be the observed values of the QoS parameters. Then, we define an objective measure of honesty as:

\[
h = 1 - \frac{|A_m - A_a| + |R_m - R_a|}{2} \text{ where } R = \frac{R_m - R_a}{R_{\text{max}} - R_{\text{min}}} \text{ if } R_{\text{max}} - R_{\text{min}} \neq 0, \text{ otherwise 1; and } R_{\text{max}}, R_{\text{min}} \text{ are the maximum and minimum values of response times respectively.}
\]

Reliability, \( r \), is measured as one minus the fraction of times the service fails or does not behave as per its function. Finally, we formalize competency, \( c \), as a binary valued concept indicating whether the service is able to satisfy the goals. If it does, we assign \( c = 1 \), otherwise 0. Thus, the objective component of trust is averaged as:

\[
t_{cw} = \frac{h + r + c}{3}
\]

We now define our model of trust, \( t_{cw} \), as a convex combination of the subjective feedback and objective measures.

Trusted Compositions of Web Services
We describe how trust values of services may be utilized while forming compositions, and subsequently updated. Our initial trust framework is shown below in Fig. 1.

In order to compute the composite trust, we consider four types of basic flows of services that are often encountered in compositions. We point out that the methods for computing the composite trust are somewhat analogous to the computations in (Cardoso et al. 2004).

- Sequential, Concurrent flows and Loop: As each of the services is executed, we construct the composite trust, \( t_{cw} \) using the honesty, reliability and competency of the individual component services. The composite honesty is the average of the honesty of all the services:

\[
h_{cw} = \frac{\sum_{i=1}^{n} h_{wi}}{n}
\]

The composite reliability aggregates the reliability of all individual services:

\[
t_{cw} = \prod_{i=1}^{n} t_{wi}
\]

Finally, the composition is competent if all component services are competent, otherwise it is not.

Copyright © 2009, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.
combination of the objective and subjective trusts. Subsequently, the aggregate trust of the composition is a value for the service, thereby remembering the history.

We seek to update the trust values of the component services within a composition after it has executed. We utilize the new mean value to update the trust. This approach is Bayesian, and integrates the new trust with the previous trust value for the service, thereby remembering the history.

We point out that trust values associated with a service are not private to a user and are updated by all users interacting with compositions containing the service. This has the advantage that a service deemed untrustworthy by some user (\( t_w \) is below her threshold) could have its trust value improve over time because of interactions with other users, until it meets the threshold of the user. This is what makes trust in our framework social - the ability for a community of users to influence the trust of a web service.

**Experimental Validation**

The objective of our experiments is to validate the utility of our trust framework. We empirically demonstrate that a consideration of trust in the selection of compositions results in compositions that progressively exhibit less deviations from their advertised or agreed upon QoS values.

In conclusion, with the growing number of mashups and websites delivering Web services (contributing to the emergence of Web 3.0), we believe that this work will be relevant in pulling together the threads needed to deliver trustworthy services to users.

**References**


