



Vocabularies and Architecture for Implementing Trust in the Semantic Web

Project name:

Semantic Web Advanced Development for Europe (SWAD-Europe)

Project Number:

IST-2001-34732

Workpackage name:

11. Trust

Workpackage description:

<http://www.w3.org/2001/sw/Europe/plan/workpackages/live/esw-wp-11.htm>

Deliverable title:

11.2: Implementation of a Trust System

URI:

<http://www.w3c.rl.ac.uk/SWAD/deliverables/11.2.html>

Authors:

- Alvaro Arenas, CCLRC, UK
- Brian Matthews, CCLRC, UK
- Michael Wilson, CCLRC, UK
- Jan Grant, ILRT, University of Bristol

Abstract:

This document is a description of an ontological description of a trust assertion, and the implementation in N3 of a trust system for an advanced electronic house scenario.

Status:

Completed, 2004-10-31.

Version:

Comments on this document are welcome and should be sent to Brian Matthews

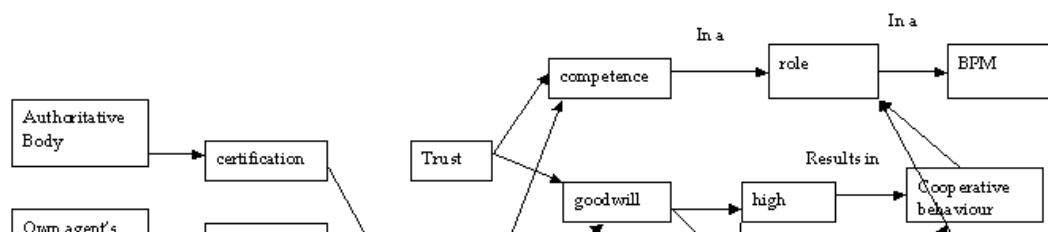
Contents

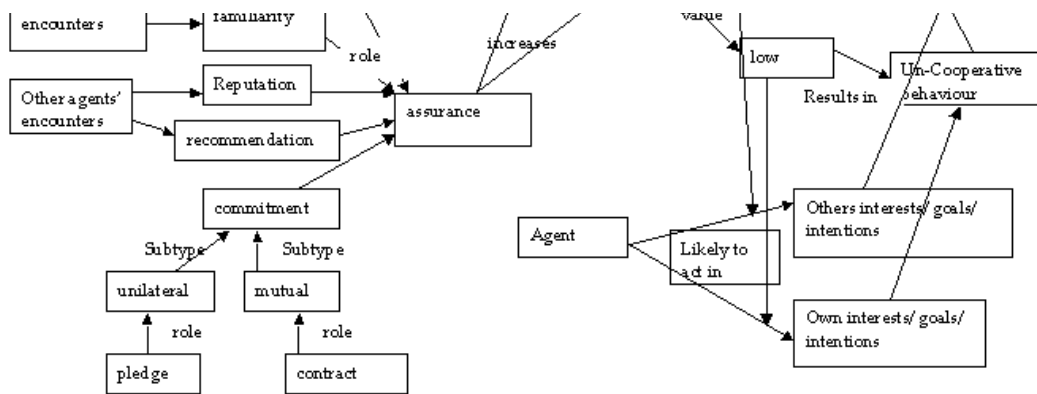
1. Introduction
2. Semantic Web Conceptualisation of Trust
3. An Architecture for a generic Semantic Web trust system
4. Conclusion

1. Introduction

Deliverable 11.1 has outlined a framework in which trust can be considered. The present deliverable takes that framework a step further to show how the concepts relating to trust can be represented in a Semantic Web ontology to interact with other Semantic Web technologies, and be used in an implemented system.

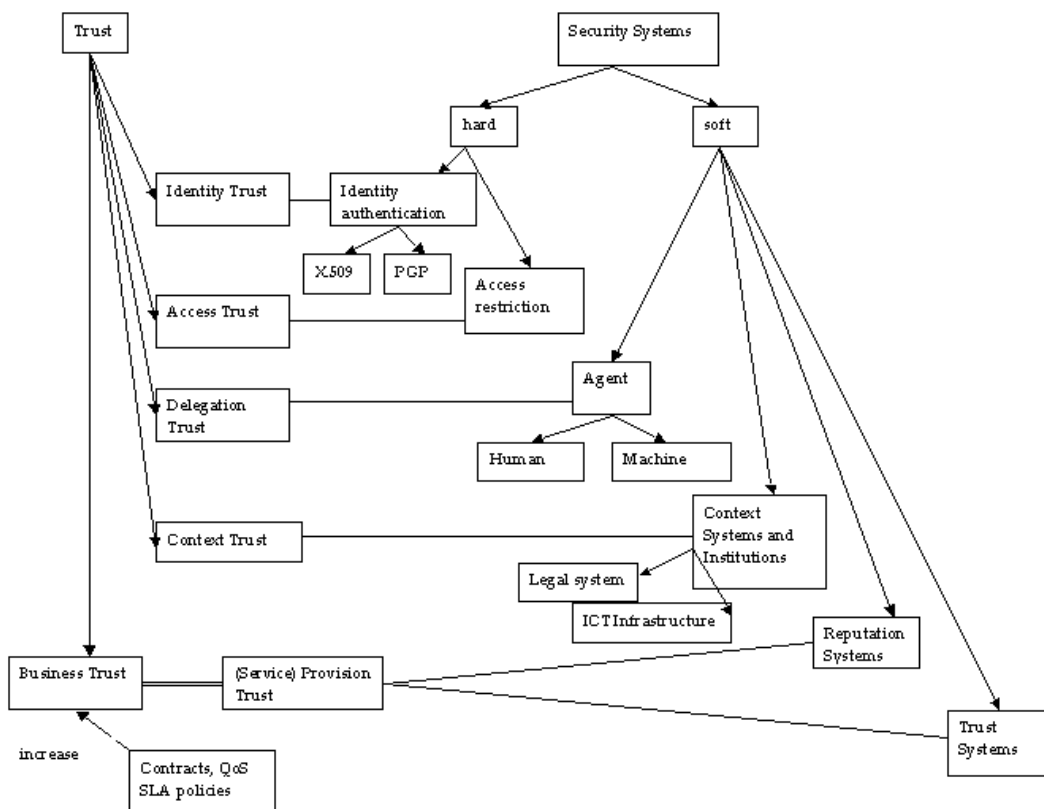
Figure 1 illustrates the core concepts relating to trust as used in social and economic sciences. The important point this makes is that trust is not competence at something, which can be separately determined, but a measure of goodwill. The goodwill of an agent to another is a measure the probability that their actions will be motivated towards the goals of all parties involved in the relationship, and not purely be motivated by their own goals. In the accountancy literature there is considerable work which attempts to accept a financial valuation of this notion of good will between a company and both its suppliers and its customers. This effort results in the frequent updating of regulations on the permissibility of goodwill as an intangible asset in auditable accounts, however, it does hold out hope that the concepts of trust in social science can be grounded both in financial prices and in security operations in computing. The breadth of the use of the concept of trust between finance, social science and computing both demonstrates the power of formalising conceptualisations of trust, and the conflicts that arise from incompatible interpretations used for different purposes.





1. The core trust concepts as used in social and economic sciences

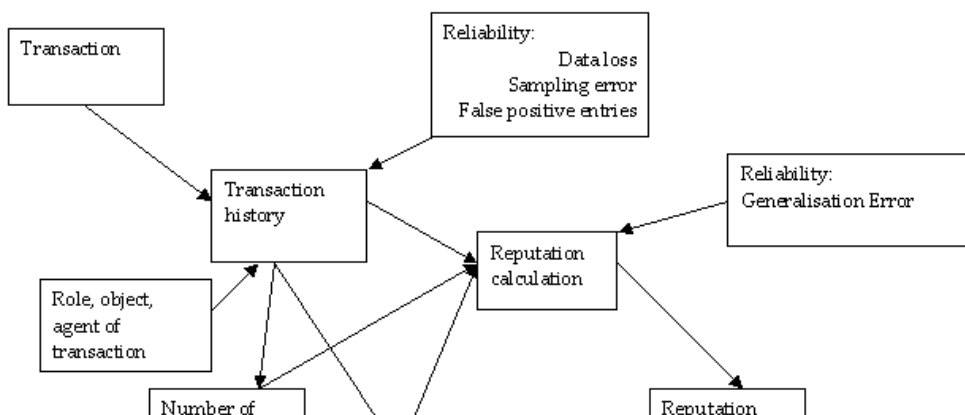
This view of trust is strongly contrasted with that used in the computer science literature which is illustrated in figure 1, which relates trust concepts to those of security in systems under consideration.



2. The core trust concepts as used in the computer science literature

This shows that the case of trust under consideration in the previous deliverable and that used in social science, is classified in computer science as Provision Trust, Service Provision Trust, or Business Trust. This concept of trust has been little addressed in the computing literature because of the vagueness of the conceptualisation of the trust relation itself, and the scope to which it applies.

In computing, the issue of competence has been addressed separately by reputation systems. These record a history of the transactions of an agent, and calculate an assurance from this that can be passed as a certificate. The reputation calculation must include components to address the possible reliability errors in both the data collected, and the size of the data set in presenting assurances, as illustrated in Figure 3. However, these issues can be addressed from establish statistical theory. Work has already been undertaken showing how Semantic Web, or Semantic Grid, technologies can be used to represent reputation systems alone [Majithia, Ali, Rana, Walker] so this will not be taken further, as an issue in itself, here.



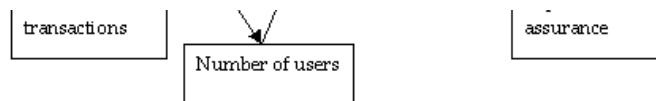


Figure 3. Factors used to calculate reputation

The next section of the paper will refine this concept of trust in the computing literature using the Sultan system as an example. In describing this Grandison and Sloman go further than the previously established computer science view of trust to define trust as a composition of concepts as illustrated in figure 4 below.

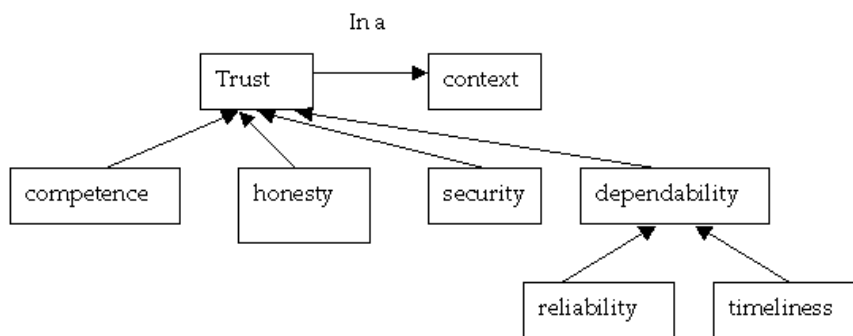


Figure 4. Trust as a composition of concepts in Sultan

Unfortunately for our current purposes, they choose to address the issue of representing a trust assertion in detail, rather than presenting a detailed account of the how these sub-components of trust can be combined in different algorithms to produce a single trust value. The structure of the trust assertion that they propose is presented below.

```

PolicyName: trust(Tristor, Trustee, ActionSet, Level)? ConstraintSet

Where:

PolicyName – the name of the policy
Tristor – the name of the trust service providing the construct
Trustee – the name of the entity being trusted
ActionSet – the action to be undertaken by the trustee as the context of the construct
Level – -100 to +100 range integer
ConstraintSet – Boolean evaluable constraints that have to be true for the construct to apply

```

This assertion, or policy as they call it, states that a Tristor trust a Trustee to perform the actions listed in the ActionSet to a Level, given that the constraints in the ConstraintSet hold true. The constraints can include statements about the environment in which the relationship holds including recommendations from third parties which are represented in a recommendation policy as shown below.

```

PolicyName: recommend(Recommendor, Recommendee, ActionSet, Level)?
ConstraintSet

```

The recommendation policy uses the same form as the trust policy, to hold as true if a Recommendor, recommends a Recommendee to perform the actions in the ActionSet at a stated level when further constraints stated in the ConstraintSet hold true. This use of recommendation policies as constraints on the trust policy very neatly avoids problems of users having to accept transitive trust relations whose role is hidden within the calculation of a trust value.

Both the trust and recommendation policies use an ActionSet to define the set of operations over which the policy applies. This aspect provides an opportunity for the Semantic Web technologies to bring added value to the existing approach. In trust systems of this class the actions in this set have no generalisable semantics. That is to say, if a bank is trusted to transfer money to another bank by a stated method, that is the limit of the trust. There is no mechanism in the semantic definition to generalise this trust in a bank to other services such as providing accounts to hold money whose existence the transfer is dependent upon, or to other facilities such as mortgage or insurance provision. This is not a criticism of approaches such as SULTAN, since it is not an issue that they intend to address. However, a richer representation of the pre- and post-conditions for the trusted action, and an ontological definition of the entities involved will support generalisations, either directly implicit in the trust assertion, entailed by it, or that can be drawn from it with additional knowledge. This ability to make generalisation inferences on the basis of explicit semantic knowledge is the benefit that the Semantic Web can bring to the business trust process.

So far we have established a rich conceptualisation of trust, mapping between that used in social science and computing, and described a detail statement of a trust assertion in a non-Semantic Web application. The issues of calculating the trust metric as a single value, and the transitivity of the trust relation have been highlighted. The issue of trust scope and generalisation has been identified and shows prospects of benefits from the Semantic Web approach to modelling. In the next section we will show how the assertion, and the complexities of the metric can be addressed using Semantic Web technologies.

2. Semantic Web Conceptualisation of Trust

A working definition of trust can be given as follows:

Trust of a party A to a party B for a service X is the measurable belief of A in that B behaves dependably for a specified period within a specified context (in relation to service X).

Note that:

- Dependability is deliberately understood broadly to include security, safety, reliability, timeliness, maintainability
- Trust is relative to a specific service. Different trust relationships appear in different business contexts.
- The measurement may be absolute (e.g. probability) or relative (e.g. dense order).
- This period may be in the past (history), the duration of the service (from now and until end of service), future (a scheduled or forecasted critical time slot), or always.

Thus any usable definition of trust on the semantic web needs to represent all of these factors. Special account needs to be taken of the *trust metric* used.

The assertion of trust between members of a human community have been addressed based on the FOAF schema [Goldsack Hendler and Parsia]. This assertion uses a single trust metric in a an integer range from 0-9. As outlined in the previous deliverable, in this discussion we wish to focus more specifically on trust relations in e-business, and more specifically on the trust relations between Web Services and their users.

Several research teams have addressed web services within the framework of the Semantic Web developing ontologies to define them [OWL-S, WSMO]. These define properties of web services, including a free field to define quality, but do not specifically address trust. However, we will return to these later to refer to concepts which they do define.

One of the most influential approaches to trust is that of [Josang] who has proposed mechanisms based upon Subjective Logic to calculate both reputation and recommendation providing a recommendation value including three measures that are meaningful within that framework. A Semantic Web ontology has been developed within SWAD-e to define this specific assertion (see appendix B, and SWAD Deliverable 11.1).

However, there has been no development in the Semantic Web to represent a more generic trust assertion following the SULTAN model.

We gives a generic method for modelling Trust on the Semantic Web. The basic construct is a trust statement given as in Figure 5

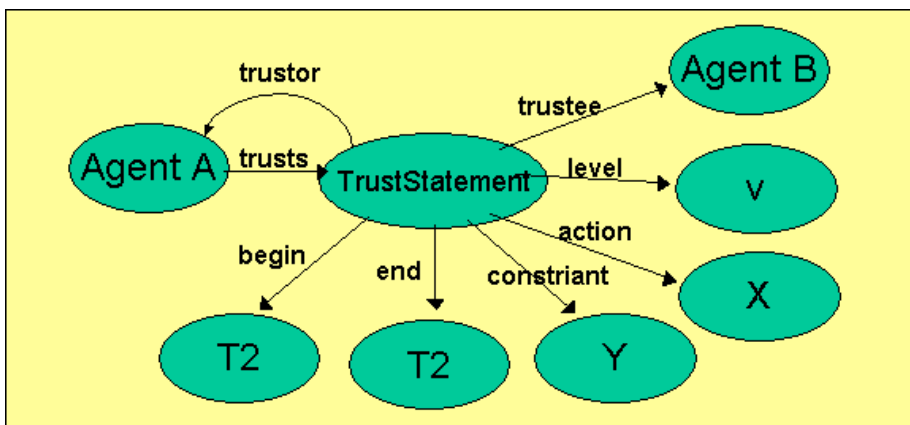


Figure 5: An example trust statement graph in RDF

Within the ontology presented are included various terms which are not defined in detail. These include Actor, Action and Metric. The Action and the Service can be defined as they are in the existing Semantic Web ontologies for web services such as OWL-s and SWMO. Thus the basic trust statement such as

A has trust in B to do X in context Y in time period (T1, T2) to value 0.8

can be given as the set of RDF statements as follows.

```

<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
  <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
  <!ENTITY rdf_ 'http://protege.stanford.edu/rdf/'>
  <!ENTITY xsd 'http://www.w3.org/2001/XMLSchema'>
  <!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema#'>
  <!ENTITY trust 'http://www.w3c.rl.ac.uk/SWAD/TrustStatement#'>
]>
<rdf:RDF xmlns:rdf="&rdf;"
  xmlns:rdf_="&rdf_;"
  xmlns:xsd="&xsd;"
  xmlns:trust="&trust;"
  xmlns:rdfs="&rdfs;">

<trust:TrustStatement rdf:ID="s111" >
  <trust:trustor>
    <trust:Agent rdf:about="A" />
  </trust:trustor>

```

```

<trust:trustee>
  <trust:Agent rdf:about="B" />
</trust:trustee>
<trust:action>
  <trust:Action rdf:about="X" />
</trust:action>
<trust:begin>T1</trust:begin>
<trust:end>T2</trust:end>
<trust:constraint>
  <trust:Constraint rdf:about="Y" />
</trust:constraint>
<trust:level>
  <trust:TrustLevel>
    <trust:metric
      rdf:resource="http://example.org/trustMetric" />
    <trust:level>0.8</trust:level>
  </trust:TrustLevel>
</trust:level>
</trust:TrustStatement>
</rdf:RDF>

```

Here, an Agent A (human or machine) can trust another Agent B for some Action X, within some time bound T1 to T2 within constraints (or context) Y, to a trust level defined with some trust metric defined by some URI <http://example.org/trustMetric> with a value 0.8.

The full RDF schema for trust statements is given in Appendix A.

Further extensions of this RDF format is to allow the RDF model to give generic rules based on classes of agents. In Figure 5 we give a a rule stating that Agent A trust any Agent to a trust level V. Such rules can form the basis of trust policies.

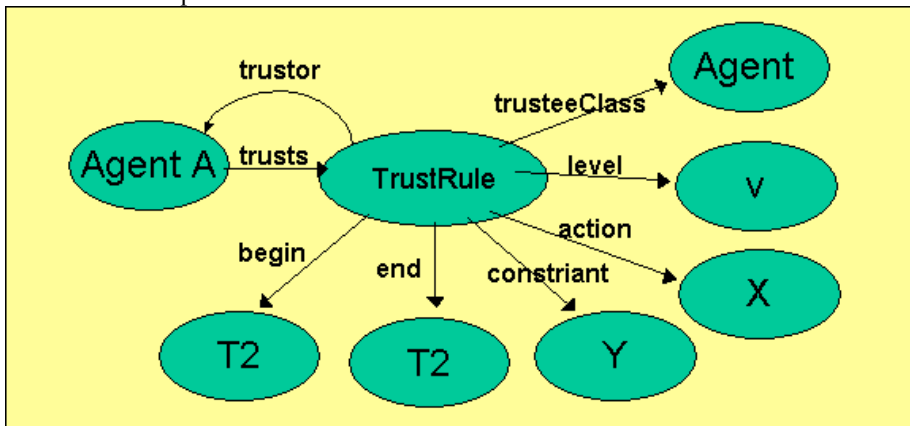


Figure 5: An example trust rule graph in RDF

Trust Metrics - Within this model, a crucial role is played by the Trust Metric. There are a variety of possible metrics for trust that could be included in an ontology with different values that would be used in the trust assertion as outlined in figure 6.

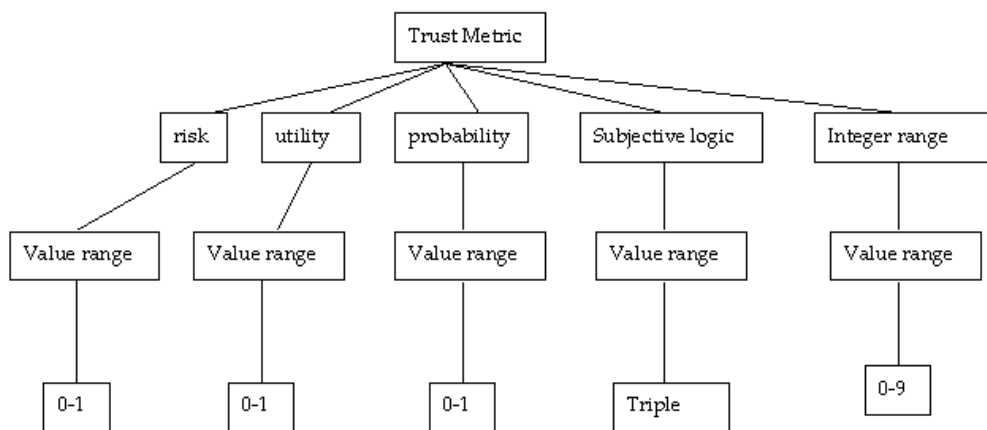


Figure 6: Options for Trust Metrics for the Trust Assertion

We give an RDF Schema for representing common classes of trust metric, such as risk level (could then be enumerated as high, medium and low), the subjective logic triple valuation, (see Deliverable 1.1 for more details), a simple probability (typically in the range 0-1), or one of a range of values (such as 0-9 in Goldsack and Hendler).

```

<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
  <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
  <!ENTITY xsd 'http://www.w3.org/2001/XMLSchema'>
  <!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema#'>

```

```

    <!ENTITY trust 'http://www.w3c.rl.ac.uk/SWAD/TrustStatement# '>
] >
<rdf:RDF xmlns:rdf="&rdf;"
        xmlns:xsd="&xsd;"
        xmlns:trust="&trust;"
        xmlns:rdfs="&rdfs;">

<rdfs:Class rdf:about="Risk">
  <rdfs:label="Risk">
  <rdfs:subClassOf rdf:resource="trust:TrustMetric"/>
</rdfs:Class>

<rdfs:Class rdf:about="Utility">
  <rdfs:label="Utility">
  <rdfs:subClassOf rdf:resource="trust:TrustMetric"/>
</rdfs:Class>

<rdfs:Class rdf:about="Probability">
  <rdfs:label="Probability">
  <rdfs:subClassOf rdf:resource="trust:TrustMetric"/>
</rdfs:Class>

<rdfs:Class rdf:about="IntegerRange">
  <rdfs:label="IntegerRange">
  <rdfs:subClassOf rdf:resource="trust:TrustMetric"/>
</rdfs:Class>

<rdfs:Class rdf:about="SubjectiveLogic">
  <rdfs:label="SubjectiveLogic">
  <rdfs:subClassOf rdf:resource="trust:TrustMetric"/>
</rdfs:Class>

</rdf:RDF>

```

The definition of the Action element of the trust assertion in terms used in OWL-S, which can also be used in domain specific ontologies allows for the implementation of pre- and post- condition based reasoning about actions that will support generalisation inferences.

The ontology proposed here allows trust assertions of the class used in SULTAN to be used within a Semantic Web context that will support inferences

Recommendations - Another key feature of the trust model is the use of recommendations. This is covered in [Deliverable 11.1](#), and so is not repeated here.

Classes of Agent - A trust model such as [\[Dimitrakos\]](#) defines several classes of agents. Trust is (unintentionally) transitively transferred along certain mediating parties.

Guarantors

Guarantors offers a formal promise or assurance, that all obligations of the parties she guarantees for will be fulfilled in the context of a transaction and will be of a specified quality and durability.

Intermediates

Intermediates intervenes between other parties in a business transaction and mediates so that they establish a business relationship with or without their knowledge. Intermediates themselves can come in four varieties:

- **Transparent:** an intermediate who identifies the parties she is mediating between to each other.
- **Translucent:** an intermediate who identifies the existence of the parties she is mediating between to each other but not their identity.
- **Opaque:** an intermediate who hides the existence of the parties she is mediating between from each other.
- **Proxy:** an intermediate who is authorised to act as a substitute of another entity.

Advisers

Adviser offers recommendations about the credibility of another party.

Thus we can present an Ontology of types of agents.

```

<?xml version='1.0' encoding='UTF-8'?>

<!DOCTYPE rdf:RDF [
  <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns# '>
  <!ENTITY xsd 'http://www.w3.org/2001/XMLSchema '>
  <!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema# '>
  <!ENTITY trust 'http://www.w3c.rl.ac.uk/SWAD/TrustStatement# '>
] >
<rdf:RDF xmlns:rdf="&rdf;"
        xmlns:xsd="&xsd;"
        xmlns:trust="&trust;"
        xmlns:rdfs="&rdfs;">

<rdfs:Class rdf:about="Guarantor">
  <rdfs:label="Guarantor">

```

```

    <rdfs:subClassOf rdf:resource="trust:Agent"/>
</rdfs:Class>

<rdfs:Class rdf:about="Intermediate">
  <rdfs:label="Intermediate">
  <rdfs:subClassOf rdf:resource="trust:Agent"/>
</rdfs:Class>

<rdfs:Class rdf:about="Advisor">
  <rdfs:label="Probability">
  <rdfs:subClassOf rdf:resource="trust:Agent"/>
</rdfs:Class>

<rdfs:Class rdf:about="TransparentIntermediate">
  <rdfs:label="IntegerRange">
  <rdfs:subClassOf rdf:resource="#Intermediate"/>
</rdfs:Class>

<rdfs:Class rdf:about="TranslucentIntermediate">
  <rdfs:label="SubjectiveLogic">
  <rdfs:subClassOf rdf:resource="#Intermediate"/>
</rdfs:Class>

<rdfs:Class rdf:about="OpaqueIntermediate">
  <rdfs:label="SubjectiveLogic">
  <rdfs:subClassOf rdf:resource="#Intermediate"/>
</rdfs:Class>

<rdfs:Class rdf:about="ProxyIntermediate">
  <rdfs:label="SubjectiveLogic">
  <rdfs:subClassOf rdf:resource="#Intermediate"/>
</rdfs:Class>

</rdf:RDF>

```

Thus generic rules of behaviour, such as those specified in [Dimitrakos] can be specified.

3. An Architecture for a generic Semantic Web trust system

We give a description of a generic architecture for implementing trust within the context of virtual organisations. Figure 7 gives an overview of this architecture.

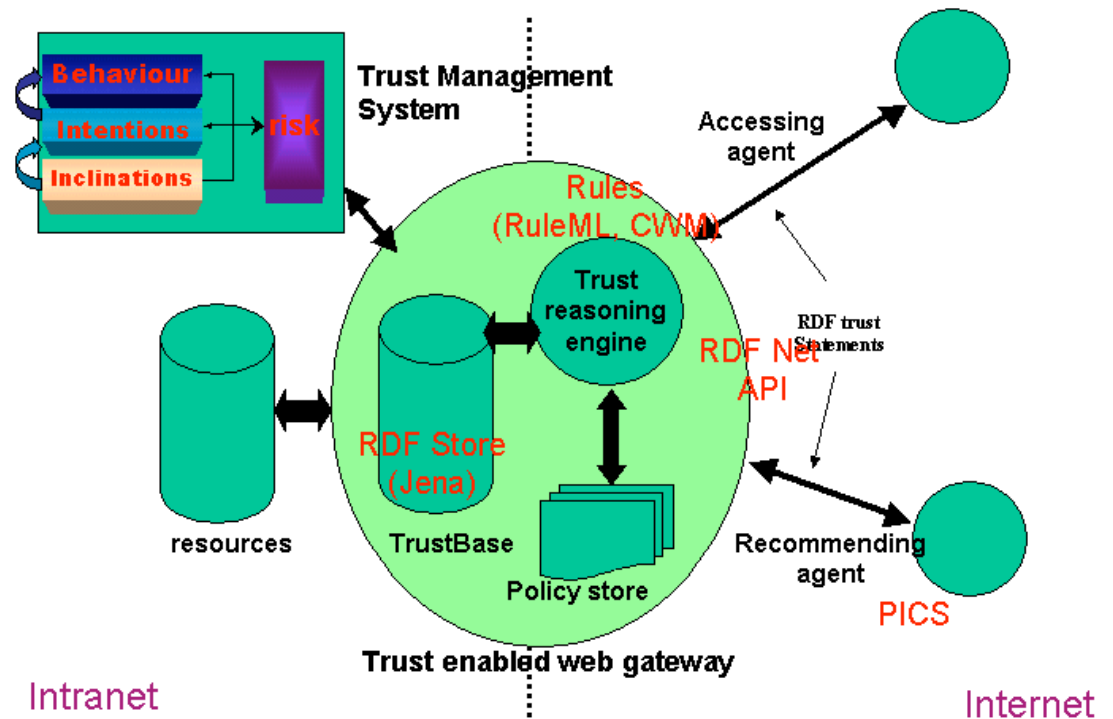


Figure 7: An generic architecture for implementing trust.

We go through the components of this architecture as follows.

Trust Enabled Gateway

Provides a server which access the internet, and uses trust based reasoning to evaluate the requests which it receives. Within the gateway, there are a *trust reasoning engine*, a *trust-*

base, and *trust-policy store*.

Trust base

Provides a local set of trust statements, in the form of a set of a RDF trust-statements. Typically, this could be based on an existing triple store such as Jena.

Trust policy store

Provides a local set of trust rules set as policies for the resources on the gateway. These will be used to evaluate the requests coming from agents for access to resources in the light of the trust valuations given in the trust base.

Trust engine

A reasoning tool to evaluate the policy in the light of the trust valuation. Also, monitors the behaviour of the accessing agent, and any recommendations from elsewhere, and reevaluates the trust valuation in the trust-base.

Rules in the trust policy and rules for modifying the trust valuations in the light of experience and recommendations, are determined by the resource holder's trust management system which assesses the risks involved on accessing resources and sets the rules accordingly.

4. Conclusion

We have presented a set of RDF vocabularies for representing trust properties, and an architecture for delivering trust based reasoning on the semantic web.

References

- Theo Dimitrakos "A Service-Oriented Trust Management Framework Trust", Reputation and Security: Theories and Practice LCNS/LNAI special volume, Springer Verlag, (2003)
- Jennifer Golbeck, James Hendler, and Bijan Parsia. Trust Networks on the Semantic Web. In Twelfth International World Wide Web Conference (WWW2003), Budapest, Hungary, May 20-24 2003.
- Tyrone Grandison and Morris Sloman. SULTAN - A Language for Trust Specification and Analysis. In Proceedings of the 8th Annual Workshop HP OpenView University Association (HP-OVUA), Berlin, Germany, June 24-27 2001.
- A. Jøsang. A subjective metric of authentication. In D.Gollmann, editor, Proceedings of the 5th European Symposium on Research in Computer Security (ESORICS'98), Springer-Verlag, 1998
- S. Majithia, A. Shaikh Ali, O. F. Rana, and D. W Walker, "Reputation-based Semantic Grid Service Discovery," in Proceedings of the Workshop on Emerging Technologies for Next generation GRID (ETNGRID-2004), June 2004.
- DAML Services Homepage <http://www.daml.org/services/owl-s/>
- Web Service Modelling Ontology Homepage <http://www.wsmo.org/>

Appendix A: A strawman RDF Schema for Trust Statements

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE rdf:RDF [
  <!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
  <!ENTITY a 'http://protege.stanford.edu/system#'>
  <!ENTITY rdf_ 'http://protege.stanford.edu/rdf/'>
  <!ENTITY xsd_ 'http://www.w3.org/2001/XMLSchema'>
  <!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema#'>
]>
<rdf:RDF xmlns:rdf="&rdf;"
  xmlns:rdf_="&rdf_;"
  xmlns:xsd="&xsd;"
  xmlns:a="&a;"
  xmlns:rdfs="&rdfs;">

  <rdfs:Class rdf:about="Action">
    <rdfs:label="Action">
    <rdfs:subClassOf rdf:resource="&rdfs;Resource"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="Agent">
    <rdfs:label="Agent">
    <rdfs:subClassOf rdf:resource="&rdfs;Resource"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="Constraiant">
    <rdfs:label="Constraiant">
    <rdfs:subClassOf rdf:resource="&rdfs;Resource"/>
  </rdfs:Class>
```



```

<rdfs:Class rdf:about="TrustLevel">
  <rdfs:label="TrustLevel">
  <rdfs:subClassOf rdf:resource="#rdfs;Resource"/>
</rdfs:Class>

<rdfs:Class rdf:about="TrustMetric">
  <rdfs:label="TrustMetric">
  <rdfs:subClassOf rdf:resource="#rdfs;Resource"/>
</rdfs:Class>

<rdfs:Class rdf:about="TrustStatement">
  <rdfs:label="TrustStatement">
  <rdfs:subClassOf rdf:resource="#rdfs;Resource"/>
</rdfs:Class>

<rdfs:Class rdf:about="TrustValue">
  <rdfs:label="TrustValue">
  <rdfs:subClassOf rdf:resource="#rdfs;Resource"/>
</rdfs:Class>

<rdf:Property rdf:about="action">
  <rdfs:label="action">
  <rdfs:range rdf:resource="#Action"/>
  <rdfs:domain rdf:resource="#TrustStatement"/>
</rdf:Property>

<rdf:Property rdf:about="begin">
  <rdfs:label="constraiant">
  <rdfs:domain rdf:resource="#TrustStatement"/>
  <rdfs:range rdf:resource="#xsd;dateTime"/>
</rdf:Property>

<rdf:Property rdf:about="end">
  <rdfs:label="constraiant">
  <rdfs:domain rdf:resource="#TrustStatement"/>
  <rdfs:range rdf:resource="#xsd;dateTime"/>
</rdf:Property>

<rdf:Property rdf:about="constraiant">
  <rdfs:label="constraiant">
  <rdfs:range rdf:resource="#Constraiant"/>
  <rdfs:domain rdf:resource="TrustStatement"/>
</rdf:Property>

<rdf:Property rdf:about="trustee">
  <rdfs:label="trustee">
  <rdfs:range rdf:resource="#Agent"/>
  <rdfs:domain rdf:resource="#TrustStatement"/>
</rdf:Property>

<rdf:Property rdf:about="trustor">
  <rdfs:label="trustor">
  <rdfs:range rdf:resource="#Agent"/>
  <rdfs:domain rdf:resource="#TrustStatement"/>
</rdf:Property>

<rdf:Property rdf:about="level">
  <rdfs:label="level">
  <rdfs:domain rdf:resource="#TrustStatement"/>
  <rdfs:range rdf:resource="#TrustLevel"/>
</rdf:Property>

<rdf:Property rdf:about="metric">
  <rdfs:label="metric">
  <rdfs:domain rdf:resource="#TrustLevel"/>
  <rdfs:range rdf:resource="#TrustMetric"/>
</rdf:Property>

<rdf:Property rdf:about="value">
  <rdfs:label="value">
  <rdfs:domain rdf:resource="#TrustLevel"/>
  <rdfs:range rdf:resource="#TrustValue"/>
</rdf:Property>

</rdf:RDF>

```

Appendix B: A strawman RDF Schema for Recommendations

```
<?xml version="1.0"?>
```

```

<!--
Copyright World Wide Web Consortium, (Massachusetts Institute of
Technology, Institut National de Recherche en Informatique et en
Automatique, Keio University).

All Rights Reserved.

Please see the full Copyright clause at
<http://www.w3.org/Consortium/Legal/copyright-software.html>

Description: Strawman RDF Schema for publishing recommendations.

Author: Jan Grant

$Id: dl1.2_trust_vocabularies.html,v 1.1 2004/10/29 17:52:52 lmiller Exp $
-->
<!DOCTYPE rdf:RDF [
  <!ENTITY rdfsns 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
  <!ENTITY rdfsns 'http://www.w3.org/2000/01/rdf-schema#'>
  <!ENTITY recns 'http://www.w3.org/2001/sw/Europe/reports/trust/recSchema#'>
]>
<rdf:RDF xmlns:rdf="&rdfs;"
  xmlns:rdfs="&rdfsns;"
  xmlns:rec="&recns;">

  <!-- classes -->

  <rdfs:Class rdf:about="&recns;Recommendation">
    <rdfs:label xml:lang="en">Recommendation</rdfs:label>
    <rdfs:comment xml:lang="en">
      A published recommendation of subject expertise
      in a given field. The Recommendation represents an assertion that the
      recommender has a given opinion of the proposition: "the recomende is an
      expert in the subject identified by the field of this Recommendation".
    </rdfs:comment>
    <rdfs:isDefinedBy rdf:resource="&recns;" />
  </rdfs:Class>

  <rdfs:Class rdf:about="&recns;Opinion">
    <rdfs:label xml:lang="en">Opinion</rdfs:label>
    <rdfs:comment xml:lang="en">An abstract class that represents the opinion
      of a recommender on a particular proposition.</rdfs:comment>
    <rdfs:isDefinedBy rdf:resource="&recns;" />
  </rdfs:Class>

  <rdfs:Class rdf:about="&recns;SLOpinion">
    <rdfs:label xml:lang="en">Subjective-Logic Opinion</rdfs:label>
    <rdfs:comment xml:lang="en">An opinion expressed in terms of Subjective
      Logic.</rdfs:comment>
    <rdfs:isDefinedBy rdf:resource="&recns;" />
    <rdfs:subClassOf rdf:resource="&recns;Opinion" />
  </rdfs:Class>

  <!-- properties -->

  <rdf:Property rdf:about="&recns;recommender">
    <rdfs:label xml:lang="en">Recommender</rdfs:label>
    <rdfs:comment xml:lang="en">The source of the published opinion.
    </rdfs:comment>
    <rdfs:isDefinedBy rdf:resource="&recns;" />
    <rdfs:domain rdf:resource="&recns;Recommendation" />
  </rdf:Property>

  <rdf:Property rdf:about="&recns;recommende">
    <rdfs:label xml:lang="en">Recommende</rdfs:label>
    <rdfs:comment xml:lang="en">The target that the published opinion concerns.
    </rdfs:comment>
    <rdfs:isDefinedBy rdf:resource="&recns;" />
    <rdfs:domain rdf:resource="&recns;Recommendation" />
  </rdf:Property>

  <rdf:Property rdf:about="&recns;field">
    <rdfs:label xml:lang="en">Field of Recommendation</rdfs:label>
    <rdfs:comment xml:lang="en">The subject that contextualises this Recommendation.
    </rdfs:comment>
    <rdfs:isDefinedBy rdf:resource="&recns;" />
    <rdfs:domain rdf:resource="&recns;Recommendation" />
  </rdf:Property>

  <rdf:Property rdf:about="&recns;validFrom">
    <rdfs:label xml:lang="en">Valid From</rdfs:label>
    <rdfs:comment xml:lang="en">The Recommendation is not valid prior to this date.
    </rdfs:comment>
    <rdfs:isDefinedBy rdf:resource="&recns;" />
    <rdfs:domain rdf:resource="&recns;Recommendation" />
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#dateTime" />
  </rdf:Property>

```

```

</rdf:Property>

<rdf:Property rdf:about="&recns;validUntil">
  <rdfs:label xml:lang="en">Valid Until</rdfs:label>
  <rdfs:comment xml:lang="en">The Recommendation is not valid after to this date.
</rdfs:comment>
  <rdfs:isDefinedBy rdf:resource="&recns;" />
  <rdfs:domain rdf:resource="&recns;Recommendation" />
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#dateTime" />
</rdf:Property>

<rdf:Property rdf:about="&recns;opinion">
  <rdfs:label xml:lang="en">Opinion</rdfs:label>
  <rdfs:comment xml:lang="en">The representation of the opinion associated
    with this Recommendation.</rdfs:comment>
  <rdfs:isDefinedBy rdf:resource="&recns;" />
  <rdfs:domain rdf:resource="&recns;Recommendation" />
  <rdfs:domain rdf:resource="&recns;Opinion" />
</rdf:Property>

<rdf:Property rdf:about="&recns;belief">
  <rdfs:label xml:lang="en">Belief</rdfs:label>
  <rdfs:comment xml:lang="en">A probabilistic measure of belief in a
    proposition;  $0 \leq \text{belief} \leq 1$ .</rdfs:comment>
  <rdfs:isDefinedBy rdf:resource="&recns;" />
  <rdfs:domain rdf:resource="&recns;SLOpinion" />
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#decimal" />
</rdf:Property>

<rdf:Property rdf:about="&recns;disbelief">
  <rdfs:label xml:lang="en">Disbelief</rdfs:label>
  <rdfs:comment xml:lang="en">A probabilistic measure of belief in a
    proposition;  $0 \leq \text{disbelief} \leq 1$ .</rdfs:comment>
  <rdfs:isDefinedBy rdf:resource="&recns;" />
  <rdfs:domain rdf:resource="&recns;SLOpinion" />
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#decimal" />
</rdf:Property>

<rdf:Property rdf:about="&recns;uncertainty">
  <rdfs:label xml:lang="en">Uncertainty</rdfs:label>
  <rdfs:comment xml:lang="en">A probabilistic measure of uncertainty in a
    proposition;  $0 \leq \text{uncertainty} \leq 1$ .</rdfs:comment>
  <rdfs:isDefinedBy rdf:resource="&recns;" />
  <rdfs:domain rdf:resource="&recns;SLOpinion" />
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#decimal" />
</rdf:Property>

</rdf:RDF>

```