AIMS: An Agent-Based Information Management System in JBI-like Environments

Srikanth Kallurkar, Bin Yu, Jay Askren,
Lewis Fishgold, Ganesh Vaidyanathan
Donald Steiner
Quantum Leap Innovations, Inc.
Newark, DE 19711, USA
{*svk, byu, jda, lhf, gv, dds*}*@quantumleap.us*

Robert Flo
Air Force Research Lab/IFSB
525 Brooks Road
Rome, NY 13441
Robert.Flo@rl.af.mil

ABSTRACT

One of the challenges for many mission-critical applications is how to determine the quality of information from distributed and heterogeneous data sources. In this paper we consider Joint Battlespace Infosphere (JBI) or JBI-like collaborative information sharing environments, where the quality of published information often depends on the quality of the contributing information and of the sources that publish them. However, as the environment becomes larger and more diverse, it is becoming increasingly difficult for human operators to assess the quality of information from various data sources. To address this challenge, we develop AIMS, an Agent-based Information Management System, to manage the quality of information in JBI-like environments, e.g., information trustworthiness. In our approach, each operator is associated with a software agent, called client agent. The client agent enables its operator to interact with the JBI repository via the services of query, publish, and subscribe. Moreover, the client agent collaborates with other agents to assess and learn the trustworthiness of information and the reliabilities of corresponding data sources from its pedigree and the feedback from operators.

KEYWORDS: Joint Battlespace Infosphere(JBI), information sharing, software agents, trust, pedigree.

1. INTRODUCTION

Collaborative information sharing environments such as JBI (Joint Battlespace Infosphere) integrate widely dispersed human decision makers with various data sources into a highly dynamic information sharing system [1]. The JBI collects data from a wide variety of sources, aggregates this information, and distributes the information to users at all force levels [9]. One challenge in JBI-like collaborative information sharing environments is that information provided by different clients, including operators and data sources, will be of varying quality, while many mission-critical applications such as battlefield operations and intelligence analysis often require trustworthy and high-quality information [11]. Current implementations of JBI developed by the Air Force Research Lab and other agencies provide a rich collection of functionalities for human operators, such as publish, query, subscribe, but none of them support any mechanisms that can assess the changing quality of information from various data sources in dynamic environments [8, 15, 16].

Assessment of information quality can be partially captured by its trustworthiness and pedigree/provenance, e.g., source and history [3]. For example, information from reliable sources usually has higher quality than the one provided by unknown or questionable sources. However, in a dynamic network centric environment with diverse communities of interests, it becomes increasingly difficult for human operators to record and analyze the history and trust of derived information and the reliabilities of various clients, including both data sources and operators [12].

In this paper we describe AIMS, an agent-based information management system, to manage the quality of information in JBI, e.g., trustworthiness. The goal of this work is to develop an experimental JBI system that can effectively manage the trustworthiness of information in a highly dynamic information sharing environment, where the exchanged information may be changed and merged by operators after it is published. We demonstrate the use of agent technology in AIMS within the domain of the intelligence analysis community, where the AIMS system automatically extracts news articles from a number of RSS (Really Simple Syndication) news feeds specified by its user and stores them in a repository.

Specifically, we describe how pedigree (or provenance) can be used to support and enhance the assessment of the trustworthiness of information. In our approach, there are multiple client agents (each of which is associated with an oper-

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **MAY 2007**	2. REPORT TYPE	3. DATES COVERED **00-00-2007 to 00-00-2007**

4. TITLE AND SUBTITLE **AIMS: An Agent-Based Information Management System in JBI-like Environments**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Air Force Research Lab/IFSB,525 Brooks Road,Rome,NY,13441**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S)
	11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

Proceedings of International Symposium on Collaborative Technologies and Systems (CTS 2007), 21-25 May, Orlando, FL

14. ABSTRACT

see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**8**	

ator) and a provenance agent. The client agent collaborates with the provenance agent to track the pedigree of the information and learn the reliability of each client, e.g., an analyst, a news source or a sensor. Moreover, the client agent reasons and learns about the trustworthiness of the derived information based on the known reliability of each information provider in the pedigree graph and their trust on each object.

The rest of this paper is organized as follows. In Section 2 we present an overview of JBI-like information sharing environments. Section 3 describes the design and implementation of AIMS in the domain of intelligence analysis. We describe our approach to managing the trustworthiness of information in Section 4. Section 5 discusses some related work in trust management and Section 6 concludes the paper with suggested directions for future research and development.

2. JBI-LIKE ENVIRONMENTS

This section provides some background on JBI-like information sharing environments, such as information objects, metadata, clients, and information pedigree.

2.1 Information Objects and Clients

Joint Battlefield Infosphere (JBI) [1] is a common framework for tactical information sharing and dissemination among commanders in different levels of echelons and operators in the battlefield command and control center. The wealth of information may include (1) synthetic aperture radar (SAR) imagery, electro-optical/infrared (EO/IR) imagery, and moving target indication (MTI) target status data; (2) meteorological and oceanographic data.

In JBI-like environments, information is stored and made available in the form of information objects. The JBI platform maintains a repository of such information objects. These objects are statements about the real world, such as a SAR sensor report, a recorded UAV video, or a description of available fuel stores in a specific area. A JBI information object contains metadata about the information, *information metadata*, as well as the information itself, *information payload* (see Figure 1). The metadata defines a common set of attributes of a JBI object, such as version number, publication time, publisher, pedigree (parents), subject keywords, language, and etc.

Figure 2 describes the metadata of an information object in the domain of intelligence analysis, where an object is a news article from a given news source such as Reuters, BBC, VOA, and NPR. From the metadata we can find the information object is a news article from BBC and it is created based on two other articles: $Reuters\_124325235$ and $NPR\_2352161$. The metadata also includes the date of being created, the source of the article, the URL of RSS news feed and a list of keywords for the article.

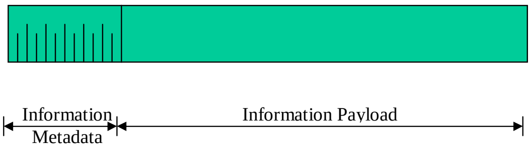
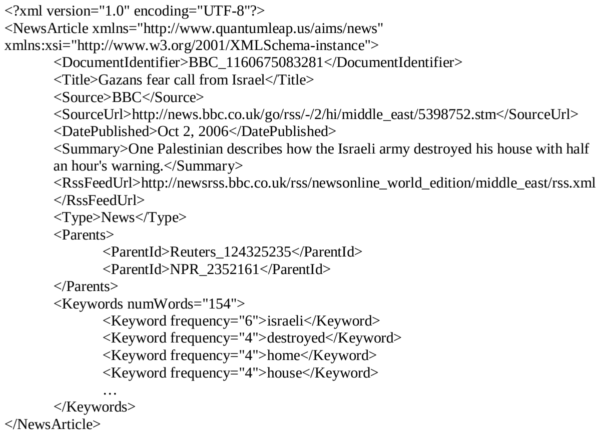


Figure 1: The information metadata and payload in a JBI information object.



```
<?xml version="1.0" encoding="UTF-8"?>
<NewsArticle xmlns="http://www.quantumleap.us/aims/news"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <DocumentIdentifier>BBC_1160675083281</DocumentIdentifier>
        <Title>Gazans fear call from Israel</Title>
        <Source>BBC</Source>
        <SourceUrl>http://news.bbc.co.uk/go/rss/-/2/hi/middle_east/5398752.stm</SourceUrl>
        <DatePublished>Oct 2, 2006</DatePublished>
        <Summary>One Palestinian describes how the Israeli army destroyed his house with half
        an hour's warning.</Summary>
        <RssFeedUrl>http://newsrss.bbc.co.uk/rss/newsonline_world_edition/middle_east/rss.xml
        </RssFeedUrl>
        <Type>News</Type>
        <Parents>
                <ParentId>Reuters_124325235</ParentId>
                <ParentId>NPR_2352161</ParentId>
        </Parents>
        <Keywords numWords="154">
                <Keyword frequency="6">israeli</Keyword>
                <Keyword frequency="4">destroyed</Keyword>
                <Keyword frequency="4">home</Keyword>
                <Keyword frequency="4">house</Keyword>
                …
        </Keywords>
</NewsArticle>
```

Figure 2: Metadata of a JBI information object.

A JBI client is any software application (or its operator) that makes use of JBI platform services to publish, subscribe, or otherwise interact with JBI information objects. There are two kinds of clients in a JBI-like environment: *operators* and *data sources*. The data sources such as sensors and news sources can publish information to the system, but they cannot manipulate information objects once they are published. Instead, an operator can query, manipulate and merge the exchanged information. The operator can also publish the merged information object and become its owner (the publisher of the object). For example, in the battlefield, a commander may publish a “spot report” based on available intelligence reports, sensor data and other background information such as terrain and weather. Similarly, in the domain of intelligence analysis, an analyst often needs to assess the reports based on his/her understanding of the past and current relevant documents. The analyst will also write reports that reflect his/her critical thinking and prediction about the likely course of action of a specified event. The derived information published in the system can be shared with other analysts with similar interests.

2.2 Information Pedigree

In order to allow users to find detailed information, a derived information object usually contains references to the objects from which it is derived, e.g., its parents. In JBI this “parents” information is recorded in the metadata of an information object. Given the parents information in the metadata of information objects, we can easily build up the pedigree graph for the derived information object. Figure 3 shows an example of pedigree graph for a derived information object D. As shown in Figure 3, if the user merges three pieces of information A, B, and C to create a fourth piece of information D, the parents of D are A, B, and C. The grandparents of D are the union of the parents of A,B, and C which in Figure 3 would include X, Y, and Z.

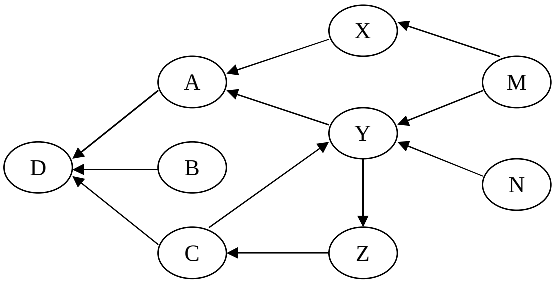


Figure 3: An example of the pedigree graph for information object D.

However, for information derived from many sources, a brute force method that annotates and retrieves entire pedigree information with a tracking log may be combinatorially expensive in storage requirements and in processing time. We have developed a preliminary framework for scalable pedigree information representation and exploitation, in which we provide a suitable representation of pedigree data, *information chromosome* or *VChrome*, that can be configured to provide cost-apportioned provenance information given an operator’s requirement.

Specifically, the operators can explicitly configure the scope of provenance in the pedigree graph within certain storage and time bounds. For example, in Figure 3, an operator might only want to access the upstream nodes of 2 hops away from object D, e.g., A, B, C, X, Y, and Z. The motivation here is that the size of the annotations may grow exponentially with the scope of a pedigree graph, while the utility of these annotations may diminish quickly along the path from the derived information object.

3. Design and Implementation of AIMS

In this section we first describe the architecture of AIMS and then we introduce its main functionalities such as query, publish, and subscribe.

3.1 Architecture of AIMS

The AIMS system is a Java-based lightweight multiagent system developed for managing the trustworthiness of information in JBI-like information sharing environments.

In AIMS the operator obtains information from the information sharing environment such as JBI and derives new information through a client agent. The client agent provides the primary interface to facilitate the interactions between a client and the JBI repository. For example, when a client publishes an information object, the client agent will pass the derived information and a list of contributing information to the Provenance agent, an agent for retrieving pedigree information from the metadata in the information object. The provenance agent establishes either complete or approximate pedigree graph to calculate various qualities. The mechanics of establishing provenance are based on the system defaults. The service invocation parameters govern the scope of provenance to be established by some subsets of direct and indirect contributing information objects in the pedigree graph.

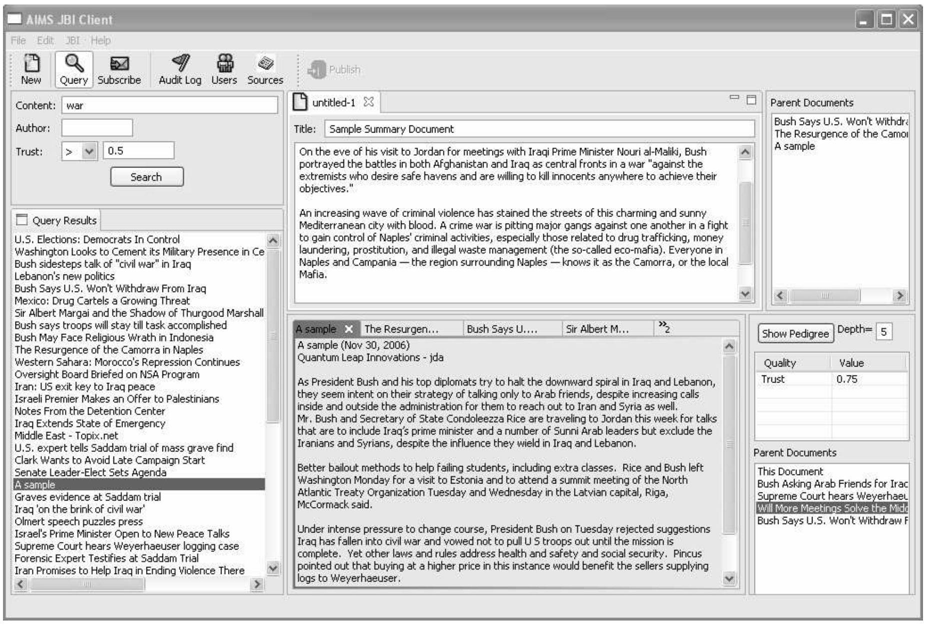
3.2 Operations in AIMS

The AIMS architecture is developed for JBI-like publish-subscribe information sharing environments. Currently, the AIMS system was evaluated by a specific group of researchers and developers at Quantum Leap Innovations. The system automatically extracts news articles from a number of RSS (Really Simple Syndication) news feeds specified by a user. Examples of RSS we used in AIMS include Yahoo Iraq: http://rss.news.yahoo.com/rss/iraq and Yahoo Middle East: http://rss.news.yahoo.com/rss/mideast. The raw XML-format news articles are parsed and stored into an information repository, which is implemented using the MySQL database.

There are two kinds of operators in AIMS system: users and superusers. Users of such an environment have three main operations - publish, subscribe, and query. Besides the three functionalities a normal user has, a superuser also has the functions of managing the quality library and user accounts. For example, a superuser might add a new method to calculate the trustworthiness of a derived information object besides $average$, max and min, depending on the applications of AIMS. Also, all users can query their log files about their past operations.

3.2.1 Register

Sharing a broad range of information over a diverse environment like JBI is non-trivial and requires careful consideration of security issues such as validation and authentication of information sources. For example, both the subscriber



Content: war
Author:
Trust: > 0.5

U.S. Elections: Democrats In Control
Washington Looks to Cement its Military Presence in Ce
Bush sidesteps talk of "civil war" in Iraq
Lebanon's new politics
Bush Says U.S. Won't Withdraw From Iraq
Mexico: Drug Cartels a Growing Threat
Sir Albert Margai and the Shadow of Thurgood Marshall
Bush says troops will stay till task accomplished
Bush May Face Religious Wrath in Indonesia
The Resurgence of the Camorra in Naples
Western Sahara: Morocco's Repression Continues
Oversight Board Briefed on NSA Program
Iran: US exit key to Iraq peace
Israeli Premier Makes an Offer to Palestinians
Notes From the Detention Center
Iraq Extends State of Emergency
Middle East - Topix.net
U.S. expert tells Saddam trial of mass grave find
Clark Wants to Avoid Late Campaign Start
Senate Leader-Elect Sets Agenda
A sample
Graves evidence at Saddam trial
Iraq 'on the brink of civil war'
Olmert speech puzzles press
Israel's Prime Minister Open to New Peace Talks
Supreme Court hears Weyerhaeuser logging case
Forensic Expert Testifies at Saddam Trial
Iran Promises to Help Iraq in Ending Violence There

Title: Sample Summary Document

On the eve of his visit to Jordan for meetings with Iraqi Prime Minister Nouri al-Maliki, Bush portrayed the battles in both Afghanistan and Iraq as central fronts in a war "against the extremists who desire safe havens and are willing to kill innocents anywhere to achieve their objectives."

An increasing wave of criminal violence has stained the streets of this charming and sunny Mediterranean city with blood. A crime war is pitting major gangs against one another in a fight to gain control of Naples' criminal activities, especially those related to drug trafficking, money laundering, prostitution, and illegal waste management (the so-called eco-mafia). Everyone in Naples and Campania — the region surrounding Naples — knows it as the Camorra, or the local Mafia.

Parent Documents
Bush Says U.S. Won't Withdra
The Resurgence of the Camor
A sample

A sample | The Resurgen... | Bush Says U.... | Sir Albert M... | »2

A sample (Nov 30, 2006)
Quantum Leap Innovations - jda

As President Bush and his top diplomats try to halt the downward spiral in Iraq and Lebanon, they seem intent on their strategy of talking only to Arab friends, despite increasing calls inside and outside the administration for them to reach out to Iran and Syria as well.
Mr. Bush and Secretary of State Condoleezza Rice are traveling to Jordan this week for talks that are to include Iraq's prime minister and a number of Sunni Arab leaders but exclude the Iranians and Syrians, despite the influence they wield in Iraq and Lebanon.

Better bailout methods to help failing students, including extra classes. Rice and Bush left Washington Monday for a visit to Estonia and to attend a summit meeting of the North Atlantic Treaty Organization Tuesday and Wednesday in the Latvian capital, Riga, McCormack said.

Under intense pressure to change course, President Bush on Tuesday rejected suggestions Iraq has fallen into civil war and vowed not to pull U S troops out until the mission is complete. Yet other laws and rules address health and safety and social security. Pincus pointed out that buying at a higher price in this instance would benefit the sellers supplying logs to Weyerhaeuser.

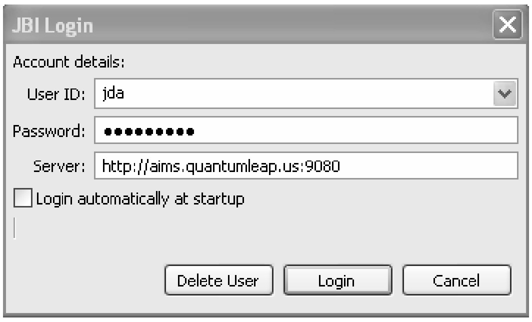
Depth= 5

Quality	Value
Trust	0.75

Parent Documents
This Document
Bush Asking Arab Friends for Irac
Supreme Court hears Weyerhaeu
Will More Meetings Solve the Midc
Bush Says U.S. Won't Withdraw F

Figure 4: JBI query, publish and subscribe interface.

and the publisher have to be authenticated by the JBI platform before any action. At the same time, the subscriber wants to make sure that the subscribed information object is coming from a legitimate publisher.



Account details:
User ID: jda
Password: ●●●●●●●●●●
Server: http://aims.quantumleap.us:9080
□ Login automatically at startup

Figure 5: JBI logon interface.

The current version of AIMS provides a simple user authentication service using passwords. When a user first comes up to the AIMS system, he needs to register and set up his password. The first step for an AIMS user is to log on his name and input the password (see Figure 5). If the password is correct, AIMS will initialize the environment(e.g., connect to server socket $http : //aims.quantumleap.us :$ 9080 and the corresponding databases on the server) and go to the main window of AIMS. Also, a user in AIMS system may unregister himself from JBI platform by clicking on the "delete user" button in the logon window.

3.2.2 Query, Publish and Subscribe

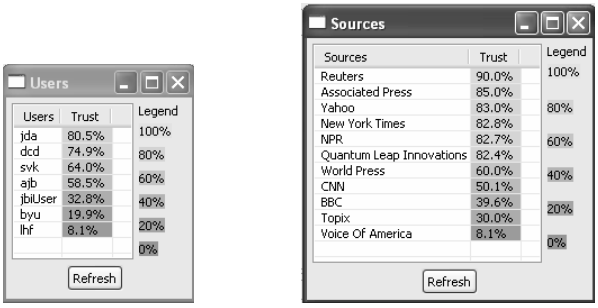
Figure 4 shows the main window of AIMS, which is implemented using Standard Widget Toolkit (SWT) - a graphical widget toolkit for the Java platform originally developed by IBM and maintained now by the Eclipse Foundation.

An authenticated user can type any text (e.g., keywords) to query the content and authors of information objects using SOAP/JDBC protocols. The AIMS system will pass the query as an XPath expression to the JBI repository and returns a list of news articles. Note that actually the Xpath expression only searches over the metadata of information objects. The user can click on the title of an article to view its content, parent documents and quality information. The user can simultaneously open multiple article in the same window and each tab (in the bottom part of the window) corresponds to an opened article.

The AIMS system also allows the user to summarize the articles in the repository for a given topic, e.g., "Iraq", and derive some new articles. Figure 4 displays the edit window for a derived article "Sample Summary Document", where the user can either type, copy and paste part of an article. The user can view the title, author, date and content of the derived article as well as its quality and pedigree. When the user decides to publish an article, the AIMS system will pop up a window to show the estimate trust for the derived article. The user may then choose to change the value of trust. The changed value of trust is used to learn the trustworthiness of parental documents and reliabilities of clients. Section 4 gives more details of our learning algorithm in AIMS.

Moreover, the AIMS system allows the user to subscribe to articles published in the repository. This is reminiscent of the force templates used by commanders of different echelons in the battlefield, but here we only support some simple features of subscribe in JBI, where a user can only specify the content, author and a threshold of trust for subscribed news articles.

3.2.3 User Management



Users	Trust
jda	80.5%
dcd	74.9%
svk	64.0%
ajb	58.5%
jbiUser	32.8%
byu	19.9%
lhf	8.1%

Sources	Trust
Reuters	90.0%
Associated Press	85.0%
Yahoo	83.0%
New York Times	82.8%
NPR	82.7%
Quantum Leap Innovations	82.4%
World Press	60.0%
CNN	50.1%
BBC	39.6%
Topix	30.0%
Voice Of America	8.1%

Figure 6: JBI monitoring interface for superusers.

Besides the basic features of JBI, such as query, publish, and subscribe, AIMS also provides a tool of user management for superusers. Figure 6 shows a list of AIMS users and their values of trust for both operators (users) and data sources. The tool is beneficial to a community of intelligence analysts, as the superuser can easily identify and strength and weakness of each analyst in a given task.

4. LEARNING IN AIMS

One contribution of AIMS is that we provide a learning mechanism that can assess the changing trustworthiness of information from various data sources in dynamic environments. In AIMS, a user can change the value of trust for a derived document. The value is captured as feedback and integrated into the AIMS system to better assess the trustworthiness of documents in the future. In this section we describe the details of our learning algorithm in AIMS. Specifically, we extend the existing JBI model in two ways,

- **Trustworthiness of an object** A scalar t is used to model the degree of trust on an information object, where $0 \leq t \leq 1$.

- **Reliability of a client** The reliability of each client s_i is captured as a scalar r_i in $[0, 1]$, which is dynamically adjusted based on the feedback from operators for derived information objects. The value of r_i is close to zero if the entity or source s_i is unreliable.

The trustworthiness of each information object is stored as an additional entry in the metadata of the object. We also introduce a database in JBI to store the reliability of all registered clients, including both operators and data sources.
In order to facilitate the interactions between operators and the system, each operator is associated with a software agent, called client agent. Moreover, there is a provenance agent in the system. The provenance agent will assess the metadata of each information object and reliabilities of each client. The provenance agent will provide the following two services to all client agents,

- Reason about the trustworthiness of a derived information object.

- Retrieve, track and update the reliabilities of clients.

The client agent collaborates with the provenance agent to annotate the calculated trustworthiness of a derived information object in its metadata when the client s_i publishes the object into the repository.

4.1 Trustworthiness of Derived Objects

Obviously, information from reliable clients usually has higher quality than the one provided by unreliable sources. In this paper we use a scalar r_i to represent the reliability of a client s_i. Given each client's belief in the information object and the client's reliability, the agent's belief in the derived object can be computed in many different ways. For simplicity, here we only consider the effects of its direct parents in the pedigree graph on its trustworthiness.
The initial trustworthiness of an information object is assigned to 0.5. Suppose d is derived from a list of information objects $\{d_1, d_2, \dots, d_L\}$ by client s and the provenance agent will estimate the trustworthiness of object d. The trustworthiness of object d can be determined as the average of trust worthiness of corresponding objects $d_1, d_2, \dots, d_L$, e.g., $\frac{\sum_{i=1}^{L} t_i}{L}$, where t_i is the trustworthiness of document d_i.

4.2 Learning the Reliability of Clients

The value of trust can be adjusted by an operator if he/she finds the estimated trustworthiness of the object by the agent is not correct. [1]For example, the client can simply change the degree of trust based on his/her experiences. Formally we can define positive and negative feedback for an information object as follows.

Definition 1 Suppose t is the trustworthiness for a derived object d, and t' is the adjusted value of trust by the client, t'

[1] Note that in JBI an operator can change the value of trust for an object only when he/she publishes it.

is considered to be a positive feedback if and only if $t \leq t'$; otherwise, t' is a negative one.

Given the estimated trust t and the feedback t' for a derived object d, the trustworthiness of object d will be adjusted as t'.

The next question is how to utilize the feedback to update the reliability r_i for a client s_i, where d_i is one of the parents of d and d_i is published by client s_i. The basic idea here is to estimate the distance between t and t'. A client will receive more penalty for its reliability if there is a larger distance.

Now let's study the mechanism of updating the reliability of other clients corresponding to the information objects in the pedigree graph.

For an information object d that is derived from a list of objects $\{d_1, d_2, \ldots, d_L\}$ by client s, the owner of d is denoted as $owner(d) = s$ and the direct parents of d are denoted as $parents(d) = \{d_1, d_2, \ldots, d_L\}$. From the metadata of each parent $d_i \in parents(d)$, we can easily build up the pedigree graph for information object d. Note that a client can be the publisher of multiple information objects in a pedigree graph.

Definition 2 A pedigree graph for a derived information object d can be defined as a directed graph $G = \{d, \Lambda, H\}$, where Λ is the set of derivative information objects and H is the maximal hop number from an information object to d.

Note that the value of the reliability of a client r_i is bounded by 0 and 1. Therefore, when we update the reliability for each client, we cannot increase or decrease it infinitely. Next we give one definition to capture the intuition. Its value increases gradually to one given the positive feedback, but decreases linearly for negative feedback. The idea here is that the reliability should be hard to build up, but easy to tear down. The following definition gives a rule to update the reliability of the publisher for an information object.

Definition 3 Given the distance ρ between t and t' for an object d and $\rho = |t - t'|$, for any of its parents s_i, its reliability can be calculated as $r_i' = \Pi(r_i, \rho) = r_i + \rho(1 - r_i)$ if the feedback is positive; otherwise, $r_i' = \Pi(r_i, \rho) = r_i(1 - \rho)$.

Besides the reliability of the publisher of object d, the client agent needs to update the reliability of other clients whose objects are used to derive new information objects. Algorithm 1 summarizes feedback propagation algorithm, where we use a breadth-first search to propagate credits/penalties. One key issue here is to deal with *cycles* in the pedigree graph, e.g., the cycle $C \rightarrow Y \rightarrow Z \rightarrow C$ in Figure 3. In our algorithm, the agent stores the visited information objects into a list $updateParents$ during the propagation process. For a given information object in the pedigree graph, we only update its client once regarding to this object. Note that, *a client can be updated multiple times for a given pedigree graph.* Moreover, the client agent of s also maintains a list of information objects at the current hop *currentLevel* and a list of information objects to be visited at the next hop *nextLevel*.

Algorithm 1 Algorithm propagateFeedback(d, ρ, H)

```
Initially, we have hop number $h = 0$, $updatedParents = \{\}$, $currentLevel = \{\}$, and $nextLevel = \{\}$.
Update the reliability of s, where $s = owner(d)$
$updatedParents = \{d\}$
$currentLevel = \{d\}$
while $h < H$ do
  while $currentLevel \neq \{\}$ do
    For each $d_i \in currentLevel$
    $s_i = owner(d_i)$
    $nextLevel = nextLevel \cup parents(d_i)$
    $currentLevel = currentLevel - d_i$
    if $s_i \notin updatedParents$ then
      Update the reliability for $s_i$ by $\Pi(r_i, \rho/2^h)$.
      $updatedParents = updatedParents \cup \{d_i)\}$
    end if
  end while
  $h = h + 1$
  For each $d_i \in nextLevel$
  $currentLevel = currentLevel \cup \{d_i\}$
  $nextLevel = nextLevel - \{d_i\}$
end while
```

For a derived object in its pedigree graph, different objects may have different contributions depending on their distance to the derived object d. Here we assume the distance from object d to object d_i is h hops, where $h = 0$ for object d. Moreover, we can specify the scope of propagation in the algorithm, e.g., $H = 2$ hops. The value of H can be easily configured in the AIMS system (see Figure 4). In each level of propagation, a client in the pedigree graph will get a credit/penalty of $\rho/2^h$ from the derived information object.

Figure 7 shows a simple example of learning in AIMS, where document D is derived from documents A, B, and C. Assume the value of estimated trust for document D is 0.5 and the feedback from the user 4 is 0.8. The AIMS system will update the trustworthiness of document D as 0.8 and the reliabilities of clients 1, 2, 3, 4 as 0.575, 0.575, 0.575, 0.65, respectively, assuming the initial values of reliability for all of them are 0.5. Figure 8 shows the changes of reliabilities for both operators (users) and data sources in AIMS.

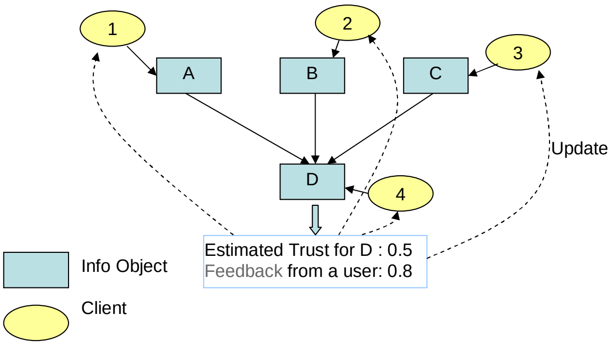


Figure 7: An example of learning in AIMS, where the circles denote clients and rectangles denote the information objects .

5. RELATED WORK

Recent research in trust is motivated by large-scale open and distributed applications such as resource sharing and electronic commerce.

One common way of assessing trust are through using reputation mechanisms. Researchers have developed various distributed reputation mechanisms for ratings representation, collection, and aggregation in peer-to-peer systems and multiagent systems [4, 10, 18, 19, 20]. More recently, [17] discusses trust representation and aggregation via a probability certainty distribution. However, most of them cannot be directly used for tracking and calculating varying quality of information published by different clients in JBI-like environments.

In the Semantic Web, [14] gives an approach to trust management based on path algebra. He assumes there are total M users and N statements and defines a class of functions for merging the trust matrix from different users using path algebra. In our system both M and N can be dynamically changing. Moreover, learning is not considered in [14]. [13]'s approach is similar to [14], but the difference is that EigenTrust computes a global trust value, instead of a personalized view of trust.

Some researchers study a similar problem in web-based information systems, but none of them consider the use of pedigree information for trust assessment. For example, Gil and Ratnaker develop a system to derive consensus trust on sources in a community of users as they use or dismiss sources for information analysis tasks [6]. Also, [2] examines the cognitive side of trust in information sources. [2] proposes a model for making trust decisions about sources, differentiating internal and external attributes affecting trust in a source. The authors note that the composition of the inputs to a trust decision affect the decision, and thus the decision itself can not be characterized by a final probability.

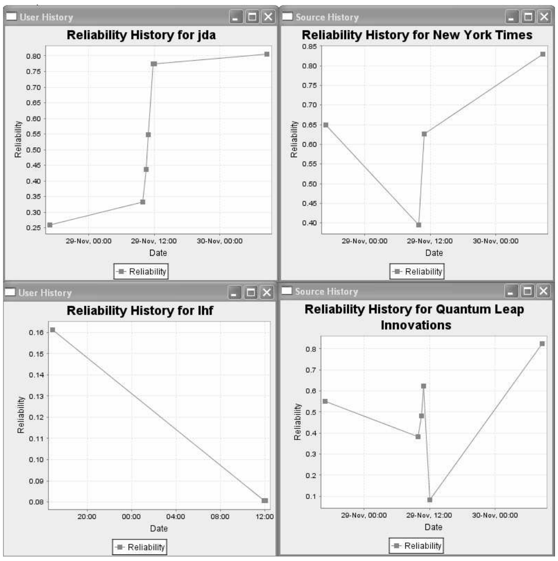


Figure 8: Screenshots of the changes of reliabilities for both operators (users) and data sources.

6. CONCLUSION

In this paper we present AIMS, a prototype implementation of a JBI publish-subscribe information sharing system. The prototype system provides a basis for the development and evaluation of mechanisms of managing the quality of information in dynamic information sharing environments. The examples show that AIMS can capture, retrieve, and reason about the trustworthiness of information and reliabilities of clients in an effective and coherent way.

In this paper we only consider the upstream learning from the users' feedback, where feedback is propagated from a derived object to its parents and grandparents. In practice, the change of source information may also affect the trustworthiness of derived information. In the future work we plan to develop similar techniques for downstream learning, where the agents can automatically propagate the changes of source information and update all derived information objects.

Moreover, we plan to study the synergies between symbolic arguments and probabilistic reasoning [5, 7]. The motivation is that two users might get different conclusions even they choose a similar set of documents for analysis. The basic idea is that we can trace the pedigree graph and compare the strength of their conclusions using some probabilistic argumentation techniques. The method will enable the newbie with less experience to share the expertise from experienced analysts in the collaborative information sharing systems.

ACKNOWLEDGEMENTS

This research has been sponsored in part by AFRL Grant DFARS 252.227-7018. We gratefully acknowledge anonymous reviewers for their valuable comments.

REFERENCES

[1] Joint battlespace infosphere (JBI). http://www.rl.af.mil/programs/jbi/.

[2] Castelfranchi, C., R. Falcone, and G. Pezzulo. Trust in information sources as a source for trust: a fuzzy approach. In *Proceedings of Second International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 89–96, 2003.

[3] Ceruti, M., S. Das, A. Ashenfelter, G. Raven, R. Brooks, M. Sudit, G. Chen, and E. Wright. Pedigree information for enhanced situation and threat assessment. In *Proceedings of the Ninth International Conference on Information Fusion*, 2006.

[4] Cornelli, F., E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati. Choosing reputable servents in a P2P network. In *Proceedings of the Eleventh International World Wide Web Conference*, pages 376–386, 2002.

[5] Das, S. Symbolic argumentation for decision making under uncertainty. In *Proceedings of the Eighth International Conference on Information Fusion*, 2005.

[6] Gil, Y. and V. Ratnakar. Trusting information sources one citizen at a time. In *Proceedings of the First International Semantic Web Conference*, pages 162–176, 2002.

[7] Haenni, R., J. Kohlas, and N. Lehmann. Probabilistic argumentation systems. In J. Kohlas and S. Moral, editors, *Handbook of Defeasible Reasoning and Uncertainty Management Systems*. Kluwer, 2000.

[8] Hall, D. and S. McMullen. *Mathematical Techniques in Multisensor Data Fusion*. Artech House Publishers, second edition, 2004.

[9] Holzhauer, D., V. Combs, M. Linderman, R. Duncomb, J. Quigley, M. Dyson, R. Paragi, and D. Young. Building an experimental joint battlespace infosphere (YJBI-CB). In *Proceedings of the sixth International Command and Control Research and Technology Symposium (ICCRTS)*, 2006.

[10] Huynh, T., N. Jennings, and N. Shadbolt. An integrated trust and reputation model for open multi-agent systems. *Autonomous Agents and Multiagent Systems*, 13(2):119–154, 2006.

[11] Johnson, M. and K. Chang. Quality of information for data fusion in net centric publish and subscribe architectures. In *Proceedings of the Eighth International Conference on Information Fusion*, 2005.

[12] Kallurkar, S., A. Johnson, D. Steiner, and R. Flo. AIMS: Agent-based information management system. In *Proceedings of AAAI Fall Symposia on Semantic Web for Collaborative Knowledge Acquisition*, 2006.

[13] Kamvar, S., M. Schlosser, and H. Garcia-Molina. The EigenTrust algorithm for reputation management in P2P networks. In *Proceedings of the Twelfth International World Wide Web Conference*, pages 640–651, 2003.

[14] Richardson, M., R. Agrawal, and P. Domingos. Trust management for the semantic web. In *Proceedings of the Second International Semantic Web Conference*, pages 351–368, 2003.

[15] Rogova G. and V. Nimier. Reliability in information fusion: Literature survey. In *Proceedings of the Seventh International Conference on Information Fusion*, pages 1158–1165, 2004.

[16] Steinberg, A., C. Bowman, and F. White. Revisions to the JDL data fusion model. In *Proceedings of the SPIE Sensor Fusion: Architecture, Algorithms, and Applications*, pages 430–441, 1999.

[17] Wang, Y. and M. Singh. Trust representation and aggregation in a distributed agent system. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI)*, 2006.

[18] Xiong, L. and L. Liu. PeerTrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Transactions on Knowledge and Data Engineering*, 16(7):843–857, 2004.

[19] Yu, B. and M. Singh. An evidential model of distributed reputation management. In *Proceedings of First International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 294–301, 2002.

[20] Yu, B. and M. Singh. Detecting deception in reputation management. In *Proceedings of Second International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 73–80, 2003.