WS-Negotiation: An Overview of Research Issues

Patrick C. K. Hung
Commonwealth Scientific and Industrial Research Organization (CSIRO)
ICT Research Centre, GPO Box 664, Canberra, ACT 2601, Australia
Phone: +612 6216 7031 Fax: +612 6216 7111
Email: Patrick.Hung@csiro.au

Haifei Li
Department of Computer Science
Nyack College, 1 South Blvd.,
Nyack, NY, 10960, USA
Phone: 845 358 1710 Ext. 2923 Fax: 845 353 5812
Email: lih@nyack.edu

Jun-Jang Jeng
IBM Thomas J. Watson Research Center
1101 Kitchawan Road, Route 134, Yorktown Heights, NY, 10598, USA
Phone: +1 914 945 1740 Fax: +1 914 945 4527
Email: jjjeng@us.ibm.com

Abstract

A Web service is defined as an autonomous unit of application logic that provides either some business functionality or information to other applications through an Internet connection. Web services are based on a set of XML standards such as Simple Object Access Protocol (SOAP), Universal Description, Discovery and Integration (UDDI) and Web Services Description Language (WSDL). In particular, Web services discovery is the process of finding most appropriate Web services providers needed by a Web services requestor. One of the important issues in the discovery process is for Web services providers and Web services requestors to negotiate and find a solution that is acceptable to both sides. Thus, a more sophisticated business model with negotiation feature is required for this challenging research area. As there are increasing demands for negotiation technologies in the context of Web services, this paper proposes an independent declarative XML language called WS-Negotiation for Web services providers and requestors. In general, WS-Negotiation contains three parts: Negotiation Message, which describes the format for messages exchanged among negotiation parties, Negotiation Protocol, which describes the mechanism and rules that negotiation parties should follow, and Negotiation Decision Making, which is an internal and private decision process based on a cost-benefit model or other strategies. This paper also presents a Service Level Agreement (SLA) template model with different domain specific vocabularies for supporting different types of business negotiations in WS-Negotiation.

Keywords: WS-Negotiation, Web Services, Negotiation Message, Negotiation Protocol, Negotiation Decision Making, Service Level Agreement Template.

1. Introduction

In the past few years, many companies have been forced to reorganize their businesses by using heterogeneous technologies in order to remain competitive in a business world. Current trends in Information and Communication Technology (ICT) may accelerate the widespread use of Web services in business [30]. Web services have become more and more popular in the research community as well as industry. Some studies [29] even show that the Web services market is expected to grow to $28 billion US dollars in sales in the coming three years. In this paper, a Web service is defined as an autonomous unit of

application logic that provides either some business functionality or information to other applications through an Internet connection. It is well known that Web services are based on a set of XML standards [11, 12] such as Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL), and Universal Description, Discovery and Integration (UDDI).

Web services have cost on the one hand, and they also produce benefit on the other hand. Like any for-profit organizations, Web services providers use resources to provide services to requestors in return for benefits. On the other side, Web services requestors pay for services from providers in return for benefits as well. In such an environment, it is believed that some requestors may tend to find the geographically closest Web services, the cheapest, the best quality, or any combination of the above. In many cases, both parties should have their own cost-benefit models for making such a business decision. One can imagine that Web services providers should register their Web services descriptions at a registry (e.g., UDDI) for public to access. In such an environment, there may have a mediator (i.e., a service locator) that helps to find appropriate Web services for requestors. This process is called matchmaking [13].

Negotiation is a decision process in which two or more parties make individual decisions and interact with each other for mutual gain [1]. Proposals are sent to other parties, and a new proposal may be generated after receiving a counter-offer. The process continues till an agreement or a deadlock is reached, or even one or more parties quit. In general, negotiation activities include the tasks of problem definition, generation of alternatives, evaluation of alternatives, preference modeling and consensus building [10]. In fact, negotiation plays a crucial part in many businesses. In a Web services discovery process, there is always more than one Web service claiming that they have the same or very similar capabilities to accomplish a requestor's requirements. One of the important issues in the discovery process is for Web services providers and requestors to negotiate and find an integrative solution that is optimal to both sides. In many cases, Web services providers may have to negotiate with requestors about cost of service (CoS) and quality of service (QoS) before any service invocation. The CoS is mainly related to the price that the requestor has to pay for the provider, and the QoS is usually related to the technical issues such as response time, availability, throughput and security [25].

As many other business activities become automated as electronic transactions, negotiation between human agents (e.g., Web services requestor and provider) can be a bottleneck. One of the major problems with negotiation between human agents that it is relatively slow, which is further complicated by issues of culture, ego and prejudice [1]. Thus, a more sophisticated business model with negotiation feature is required for this challenging research field. There are three important tasks to automate this negotiation process between Web services providers and requestors: (1) formalize the negotiation process, (2) develop an XML negotiation language for defining negotiation message, negotiation protocol and negotiation decision making, and (3) incorporate negotiation support technologies into the Web services architecture [20].

This paper proposes an independent declarative XML language called WS-Negotiation for Web services providers and requestors. Next, this paper also presents a Service Level Agreement (SLA) template model with different domain specific vocabularies for supporting different types of business negotiations in WS-Negotiation. The remainder of this paper is organized as follows: Section 2 discusses related work in the literature review. Next, Section 3 presents the research issues of the proposed WS-Negotiation. Then, Section 4 discusses an implementation model and a use case for studying WS-Negotiation. Lastly, Section 5 concludes the paper and points out future work.

2. Literature Review

Computer applications were first employed for negotiation support in the 1960s [2]. In the 1980s, computer-based negotiation support systems (NSS) emerged, and they were typically used for training and research in a laboratory environment but rarely used in practice. In general, NSS has the following basic features: (1) a formalization to describe negotiation activities in terms of choices and outcomes, (2) a way to generally characterize the associated outcome probabilities, and (3) a methodology for processing the model to evaluate the expected value of choice alternatives. NSS normally assist negotiators to assess situations, generate and evaluate options, and implement decisions. NSS attempt to match the negotiation needs and technological capabilities in order to assist the user in overcoming cognitive biases, manage the complexity of the negotiation environment, guide the user towards competitive or collaborative action, and reduce the risk of emotional negotiation. There are several initiatives such as evaluating the effectiveness of NSS. For example, there is a study [4] of applying NSS to small manufacturing enterprises (SMEs) in Hong Kong in order to shorten the new product development time. Current trends in e-commerce may accelerate the widespread use of NSS in the business world [3]. NSS creates or aids strategies for negotiation, as illustrated by the following examples:

- NEGOTIATOR [5] seeks to guide negotiators to move their individual goals and judgments to enhance the chance of achieving a common solution. It supports

problem adaptation through information sharing, concession making, and problem restructuring or re-framing. However, NEGOTIATOR helps the negotiators to make decision only without any support to other entities involved in negotiation activities.

- INSPIRE (InterNeg Support Program for Intercultural Research) [6] is a Web-based prototype NSS for inter-cultural as well as intra-cultural negotiations. INSPIRE can conduct negotiation anonymously, evaluate the goodness of an offer, and review the history of a negotiation. INSPIRE supports the tasks of preference assessment, analysis of alternative offers, offer exchange, counter-offer evaluation, and assessing compromise efficiency using the Pareto-optimality approach [24]. Although INSPIRE supports the communication among negotiators by exchanging messages, it does not deal with the interactions among different entities in negotiation activities.
- NegoPlan [8] is an expert system to structure the strategic issues. It uses a rule-based formalism to represent negotiation activities, to develop a problem representation, to provide information, and to maintain a structure that allows the consistency and validity of the model to be verified. A strategy can be developed by the knowledge representation from goals. NegoPlan focuses on the strategies for negotiation activities only, providing no support for other aspects in negotiation activities.

Researchers of the agent technology conducted another stream of work to support negotiation activities. In general, an agent has to be proactive, to be capable of personalization, and to have a certain level of autonomy. Further, most of these works have been implemented in the Internet and their experiences are valuable for developing Web services negotiation technologies. Here are several examples:

- MIT Media Lab's Kasbah [7] is an online, multi-agent consumer-to-consumer transaction system. Users create autonomous agents to buy and sell goods on their behalf, and also specify parameters to guide and constrain an agent's overall behavior. Buying and selling agents meet and negotiate in the Kasbah Marketplace directly.
- Tete-a-Tete [9] provides an integrative negotiation approach to retail sales. Shopping and sales agents negotiate across multiple terms of a transaction, including warranties, delivery times, service contracts, return policies, loan options, gift services, and other merchant value-added services.

In the recent Web services research area, there are increasing demands and discussions about negotiation technologies for different Web services applications. For example, WS-Policy [19] provides a grammar for expressing Web services policies. WS-Policy is used to specify policy information on a broad range of service requirements, preferences, and capabilities. The WS-Policy is represented by a policy expression that is an XML Infoset representation of one or more policy statements. The WS-Policy includes a set of general messaging-related assertions defined in WS-PolicyAssertions [19] and a set of security policy assertions related to supporting the WS-Security specification defined in WS-SecurityPolicy [19]. But nevertheless the current WS-Policy specification also mentions that WS-Policy by itself does not provide a negotiation solution for Web services, negotiation supports between Web services requestor and provider on an agreement about security requirements and services can be foreseen for WS-Policy in the future.

Next, W3C mentions in the Web Services Architecture Usage Scenarios [20] document that the current SOAP 1.2 specification does not provide any assertion to specify appropriate QoS mechanisms. And W3C also requires that the specification of the QoS extension have to include negotiations in the future. Further, the W3C Platform for Privacy Preferences Project (P3P) [21] provides a language called P3P Preference Exchange Language 1.0 (APPEL1.0) [22] that is used to express the user's preferences for making automated or semi-automated decisions regarding the acceptability of machine-readable privacy policies from P3P enabled Web sites. Though the APPEL 1.0 specification mentions that the current APPEL needs not be capable of expressing negotiation strategies, it is believable that negotiation technologies will be helpful to enhance privacy practices for P3P.

In particular, the OASIS ebXML Collaboration Protocol Profile and Agreement Technical Committee [23] has been formed an auto-negotiation sub-team since August 2001. Their primitive goal is to automate the negotiation process between two negotiation parties for bargaining different technical issues in the context of Collaboration-Protocol Profile (CPP). As a result, an agreement between two negotiation parties is expressed in the format of Collaboration-Protocol Agreement (CPA). In addition, they are also planning to migrate their work to include negotiation of higher-level issues such as business parameters and legal matters. However, their work mainly focuses on the CPP and CPA templates. WS-Negotiation is an independent XML language that can be applied to different types of agreement templates. Negotiation on business information is a much more complex subject, and WS-Negotiation is targeting on the business parameters and legal matters that will be discussed in the coming sections. Lastly, Tumer et al. [31] propose a negotiation model to support a privacy framework for Web services. In this privacy framework, the semantic of Web services such as the input and output parameters is described by DAML-S. On the other hand, the users can specify their privacy preferences in different permission levels based on DAML-S. Thus, the proposed negotiation model is used to

compare the user's data privacy and service data's request in order to reach an agreement. However, Tumer et al. do not discuss how the proposed negotiation model is used to generate an agreement in details. In addition, the model is based on a domain-specific problem. WS-Negotiation is a domain independent language for generating agreements between Web services. And WS-Negotiation also supports different Web services semantics by applying different domain specific vocabularies.

3. WS-Negotiation: Research Issues

Traditionally, there are two types of negotiations. Distributive negotiations (also known as zero-sum and competitive negotiations) are classified as win-lose negotiations. For example, one party reaches its goals and the other party must fail to realize it. On the other side, integrative negotiations (also known as collaborative and cooperative negotiations) are classified as win-win negotiations. The term integrative negotiation refers to the processes by which both parties locate and adopt the option that provides greater joint utility to the parties taken collectively. The major reason to adopt integrative negotiation is that integrative negotiation always reduces the likelihood that negotiations will fail, by making it possible to locate options that satisfy parties' ultimate expectations [15]. This paper focuses on a negotiation model for the Web services provider and requestor [14].

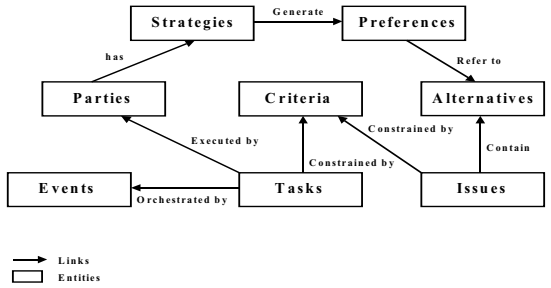

Figure 1. Relationships of Negotiation Entities

In the literature, there are three phases of an automated negotiation process [6]: (1) pre-negotiation for understanding the negotiation problem, opponent, issues, alternatives, and preference elicitation, (2) conduct of negotiation in terms of offers construction and counter-offer based on different strategies, and (3) post-settlement computation of possible offers that dominates the most recent compromise. Referring to Figure 1, negotiation includes a set of tasks that are orchestrated by a set of events. Figure 2 shows a coordination plan of negotiation process based on the literature [14, 17]. When the negotiation process is initiated from the task "Start," there is an event "Negotiation Session Started" generated to trigger the dependent task "Define Negotiation Environment" to be executed. All the entities (of Figure 1) of the negotiation process should be defined in the task "Define Negotiation Environment." As a result, the event "Negotiation Plan(s)" is generated once the negotiation plan(s) is/are finalized. The negotiation process may have to support more than one negotiation plan. Then, the task "Initial Contact" attempts to contact the opponent and finalizing the negotiation plan(s) with the opponent in order to build a consensus. Once the consensus is built, an event "Hand Shake" will trigger the task "Offer(s) and Counter-Offer(s)" to be executed. Both parties start to submit offer(s) and counter-offer(s), and each side would evaluate the proposal(s) and may submit the revised proposal(s) to other side. In the task "Evaluation," the agents should have been compromised a set of strategies that would be used to evaluate each proposal from the opponent. If the interactions between the tasks "Offer(s) and Counter-Offer(s)" and "Evaluation" are terminated, the outcomes should be either "an agreement is made" or "any side quits the negotiation" (i.e., event "Agreement or Quit"). The task "Outcomes" finalizes the deal and generates the event "Negotiation Session Ended" to task "End." Thus, the whole negotiation process is completed successfully. This negotiation process is also called bargaining [27]. There are several variations of bargaining: bilateral bargaining, multilateral bargaining, single-issue bargaining and multi-issue bargaining. Back to Figure 1, negotiation also involves a set of issues and every issue contains a set of alternatives. Further, the set of issues may be constrained by a set of criteria. In order to reach a settlement, all parties must take complementary actions on each issue. Each party has a set of strategies such as Multi-Attribute Decision-Making [16] or Cost-Benefit Decision Model [33] for generating its set of preferences. Each party has a set of preferences with respect to what alternatives are taken on each issue and how important these matters are. For example, a user may have some preferences on the types of car at a car rental; company's Web service for making a reservation such as small-size, compact, full-size and etc. In details, there are four types of negotiation issues [10]:

- Primary issues are those matters that must be negotiated.
- Auxiliary issues could be negotiated but are not important or relevant enough to introduce initially.
- Fixed issues are so important to the party that under no circumstances is any compromise possible. In many cases, a primary issue may also be a fixed issue.
- Inconsequential issues are unimportant so that the party is willing to agree to whatever the other side has proposed.

Overall, Figure 2 is the negotiation process template for

considering and designing the role of WS-Negotiation in the context of Web services negotiation scenario. In the coming paragraphs, different concepts are discussed to tackle each of these tasks. For example, the service level agreement template is used to define the negotiation issues (i.e., the task "Define Negotiation Environment") and also generate an agreement for Web services (i.e., the task "Outcomes"). The negotiation message and protocol are used to tackle the tasks "Initial Contact" and "Offer and Counter-Offer(s)." Lastly, the negotiation decision-making is used to tackle the task "Evaluation" at both sides.

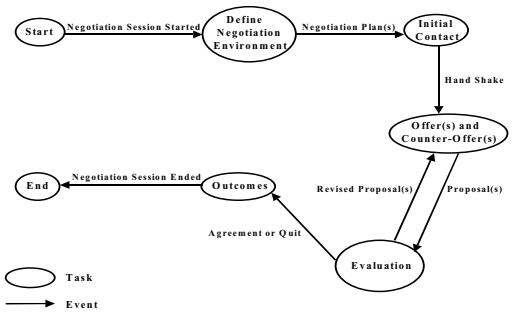

Figure 2. Coordination Plan of Negotiation Process

Figure 3. The Structure of WS-Negotiation

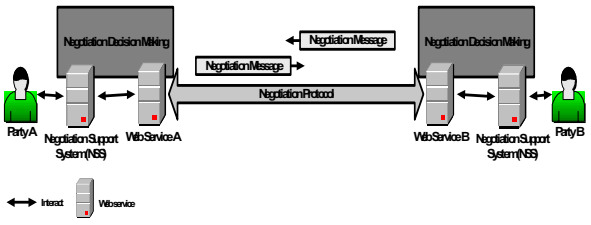

Figure 4. Overview of WS-Negotiation Framework

This section briefly overviews the research issues of the proposed WS-Negotiation for supporting negotiation activities between Web services providers and requestors. WS-Negotiation is a bilateral and multi-issue bargaining Web services language. Referring to Figure 3, the structure of WS-Negotiation contains three main parts:

Negotiation Message: It is used to describe the format for messages exchanged among negotiation parties (i.e., Web services requestor and provider) as shown in Figure 4. In the future, the WS-Negotiation specification will define the schema and semantics of each message between the Web services requestor and provider. To illustrate the concept, Figure 5 shows a negotiation message example for a car rental negotiation process. This is only an illustration of what the negotiation message may look like in the future.

```
<negotiationMessage id="1854" ref="None"
type="Offer">
  <sender>user.example.com</sender>
  <receiver>car-rental.example.com</receiver>
  <content>
    <issue>types-of-car
      <alternative preference = "1">
        compact
      </alternative>
      <alternative preference = "2">
        full size
      </alternative>
    </issue>
    …
  </content>
  <expiry>07/01/2003</expiry>
</negotiationMessage>
```

Figure 5. An Illustrated Negotiation Message Example

Referring to [23], here is a set of message types suggested:

- Offer
- Counter-Offer
- Rejected
- Accepted
- Expired
- SinglePartySigned
- Signed
- Unsigned

Negotiation Protocol: It is used to describe the mechanism and rules that negotiation parties (i.e., Web services requestor and provider) should follow. Referring to Figure 4, the negotiation protocol consists of exchanges of negotiation messages that contain the details of offers, counter-offers and etc. The Web services architecture requires the exchange of negotiation messages between both Web services requestor and provider with a third-party negotiation service called negotiation support system (NSS) discussed above. The negotiation service can also be a Web service, and this negotiation service involves

negotiation function at each party including possible human intervention [23]. Offers and counter-offers usually use a set of negotiation primitives to orchestrate the negotiation protocols. In the future, WS-Negotiation will define a set of negotiation primitives for the Web services requestor and provider to effectively communicate and inter-operate with each other. Referring to [27], here is a set of possible negotiation primitives:

- Call For Proposal (CFP): a primitive to solicit proposals from related parties.
- Propose: a primitive to propose an offer or counter-offer to the other party.
- Accept: a primitive to accept the offer / counter-offer received from the other party.
- Terminate: a primitive to unilaterally terminate the negotiation process.
- Reject: a primitive to deny the offer / counter-offer received from the other party. However, the party is willing to consider the revised offer / counter-offer.
- Acknowledge: a primitive to acknowledge the receipt of offer / counter-offer.
- Modify: a primitive to modify the sent offer / counter-offer before receiving the reply from the other party.
- Withdraw: a primitive to withdraw the previously sent offer / counter-offer and is committed to send a new offer / counter-offer to the other party.

Negotiation primitive has more rich semantics than negotiation messages. Specifically, each negotiation primitive has pre-condition and post-condition. For some primitives, there are corresponding messages. However, for some primitives, it also enforces the rules and constraints that should be obeyed by both parties. For example, in order for the "Modify" primitive to be valid, it must be sent before receiving the reply from the other party. It is usually used in the case for human intervention. The primitives of negotiation protocol can be implemented as Web methods on both sides. Referring to the car rental negotiation process, Figure 6 shows an illustrative simplified WSDL document for the negotiation primitive "Propose." Again this is only an illustration of what the primitive "Propose" at the Web services provider side may look like in the future.

```
<?xml version='1.0' encoding='UTF-8'?>
<definitions
 xmlns:tns=
  'http://car-rental.example.com/provider.wsdl'
 xmlns='http://schemas.xmlsoap.org/wsdl/'
 xmlns:soap=
  'http://schemas.xmlsoap.org/wsdl/soap/'
 targetNamespace=
  'http://car-rental.example.com/provider.wsdl'
 name='Car Rental Company'>

 <import
  namespace=
   'http://negotiation.example.com/msg/schemas'
  location=
   'http://negotiation.example.com/2003/ns.xsd'/>

 <message name='ProposeInput'>
  <part element='NegotiationMessage'
   name='body'/>
 </message>
 <message name='ProposeOutput'>
  <part element='Status' name='body'/>
 </message>

 <portType name='ProposePortType'>
  <operation name='Propose'>
   <input message='tns:ProposeInput'/>
   <output message='tns:ProposeOutput'/>
  </operation>
 </portType>

 <binding type='tns:ProposePortType'
  name='ProposeSoapBinding'>
  <soap:binding
   transport=
    'http://schemas.xmlsoap.org/soap/http'
   style='document'/>
  <operation name='Propose'>
   <soap:operation
    soapAction=
     'http://car-rental.example.com/propose'/>
   <input>
    <soap:body use='literal'/>
   </input>
   <output>
    <soap:body use='literal'/>
   </output>
  </operation>
 </binding>

 <service name='CarRentalNegotiationService'>
  <documentation>
   Car Rental Negotiation Service
  </documentation>
  <port binding='tns:ProposeSoapBinding'
   name='ProposePort'>
   <soap:address
    location=
     'http://car-rental.example.com/propose'/>
  </port>
 </service>
</definitions>
```

Figure 6. An Illustrative Simplified WSDL Document for the Negotiation Primitive "Propose"

Negotiation Decision Making: It is an internal and private decision process based on a cost-benefit model [33] or other strategies. Referring to Figure 4, the negotiation decision-making process is the private process at the Web services requestor and provider side. Each negotiation party uses its own negotiation strategies, in conjunction with an agreement template (to be discussed in coming paragraphs), to arrive at an offer or counter-offer in the negotiation protocol. With the assumption of negotiation technologies to handle different negotiation strategies, WS-negotiation only defines the name of negotiation

strategy with criteria (if any) as shown in Figure 7.

```
<negotiationDecisionMaking id="1001086"
ref="none">
  <strategy>CostBenefitAnalysis</strategy>
  <criteria>None</criteria>
</negotiationDecisionMaking>
```

Figure 7. An Illustrative Negotiation Decision Making Example

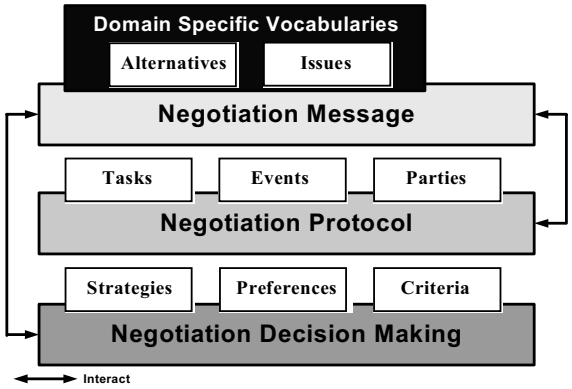

Figure 8. Mapping between WS-Negotiation and Negotiation Entities

After all, Figure 8 depicts the mapping between the negotiation entities (discussed in Figure 1) and WS-Negotiation (discussed in Figure 3). The negotiation message part in WS-Negotiation handles the information flow of the issues with relevant alternatives between Web services requestor and provider. This paper introduces the concept of domain specific vocabularies for different types of business negotiations into WS-Negotiation framework. For example, one can imagine that there exists a privacy specific vocabulary like P3P [26]. Next, it is obvious that those negotiation primitives are used to coordinate and execute the negotiation tasks and to handle the negotiation events between two negotiation parties. Then the negotiation decision-making part uses the negotiation strategies with criteria (if any) to determine the preferences for each issue during the negotiation process. In a picture, the negotiation protocol orchestrates the negotiation message and the negotiation decision-making is driven by the received negotiation messages.

Although the negotiation issues vary from one business domain to another, one can imagine that the number and nature of the issues are alike or fixed in a specific negotiation domain. Thus, the negotiation messages exchanged between two negotiation parties include a set of predetermined issues and a negotiation template is composed of those issues. Further, the negotiation template is the initial layout of an agreement between two negotiation parties that contains a set of service level indicators such as response time and availability with a target level to achieve. A service-level agreement (SLA) is a formal contract between a Web services requestor and provider guaranteeing quantifiable issues at defined levels only through mutual concessions [25]. The negotiation issues are described as SLA parameters, and the SLA parameters are based on the domain specific vocabularies. W3C also emphasizes the important role of SLA in the selection of QoS [20].

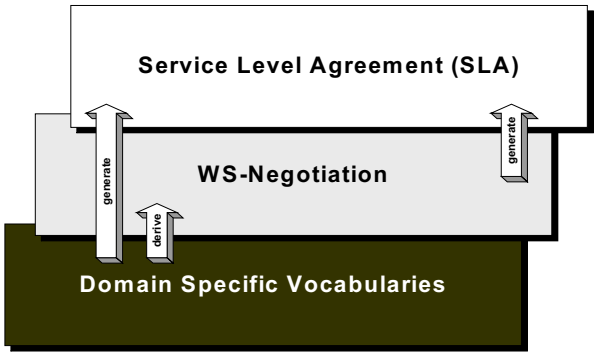

Figure 9. The SLA Template Model

Up to this moment, W3C does not have any specification for modeling Web services agreements. Figure 9 proposes a generic framework of SLA as an agreement template to facilitate WS-Negotiation. A SLA template can be published in a registry such as UDDI [28]. A SLA template can be seemed as a partially completed SLA document with issues for filling in unspecified information. After a sequence of negotiation messages exchange via WS-Negotiation, a SLA document is created. A SLA document comprises the parties, their roles and the action interfaces they expose to the other parties of the agreement. A SLA document is used to specify the service level parameters that describe the negotiation parties' guarantees and obligations. Referring to Figure 9, the domain specific vocabularies can directly generate the SLA documents or the vocabularies can derive a negotiation process in the context of WS-Negotiation. Only through mutual concessions can the negotiation process generate a SLA document [25]. According to some existing SLA specifications, one potential solution that can be applied in this model is Web Service Level Agreement (WSLA). Further, WSLA even provides an extensible mechanism to include domain specific vocabularies [28].

4. Discussion

This section discusses one possible technical approach to implement WS-Negotiation and a use case for studying WS-Negotiation in the future. This is by no mean the only approach. Figure 10 shows a picture that negotiation

messages are embedded into SOAP messages, either in the SOAP header or body. To illustrate, Figure 11 shows a sample WS-Negotiation embedded SOAP message. OASIS also suggests using SOAP messages to implement a negotiation process [23]. As usual, SOAP messages can be binding to HTTP POST or GET. The negotiation primitives are implemented as Web methods at each Web service, and the negotiation decision-making mechanisms are located at each end.

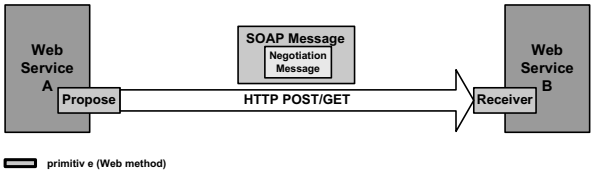

Figure 10. An Implementation Model for WS-Negotiation

Figure 12 presents a use case called healthcare administrative data integration for studying the impact and practice of WS-Negotiation in the future [18]. The major initiative of this application aims to provide a platform for improving healthcare services through the provision of an integrated view of heterogeneous information and information processing resources that are distributed across healthcare enterprises in a loosely coupled environment. In particular, health informatics researchers have changed the emphasis in healthcare from treatment to prevention. Though the full integration of health administrative information is not an easy task, it is vital to facilitate public health planning and studies of diseases (e.g., health data and epidemiological statistics) in many countries for the public good. To improve healthcare services and remain competitive, health informatics researchers must be able to access comprehensive, accurate, and timely integrated health information for forecasting demands on health services, allocating resources, and monitoring performance. Figure 12 shows the architecture that consists of a set of Web services (e.g., data and data analysis services) located at *M* locations, a workflow management system (WFMS), and system interfaces. Typically, the WFMS accepts user requests with login information from the system interfaces, generate an authenticated security-token such as WS-Security [19] for the user, connects the Web services from different locations with the user's security-token, integrates a set of datasets from the Web services with the data custodian or service provider's read-access approval into an integrated view, and then presents it to the user. The interactions between the WFMS and Web services are all read-only. Thus, the WFMS integrates the datasets from various isolated health and social databases through Web services into integrated views and perform certain data analysis processes. In a result, the WFMS sends the results in certain format such as some graphical representations to the users such as policy-makers, practitioners, and researchers.

```
<?xml version='1.0' encoding='UTF-8'?>
<env:Envelope
 xmlns='http://provider.example.com/2003/ns'
 xmlns:env=
  'http://www.w3.org/2003/05/soap-envelope'>
 <env:Body>
  <negotiationMessage id="1854" ref="None"
  type="Offer">
   <sender>user.example.comURI</sender>
   <receiver>car-rental.example.com</receiver>
   <content>
    <issue>types-of-car
     <alternative preference = "1">
      compact
     </alternative>
    <alternative preference = "2">
     full size
    </alternative>
    </issue>
    …
   </content>
   <expiry>07/01/2003</expiry>
  </negotiationMessage>

 </env:Body>
</env:Envelope>
```

Figure 11. A WS-Negotiation Message embedded SOAP Message Example

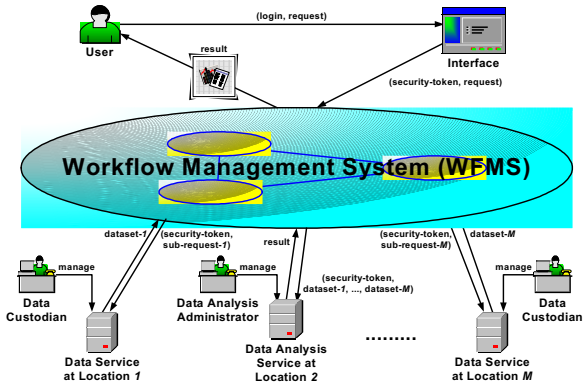

Figure 12. A Use Case for Testing WS-Negotiation

In this specific application, the user has to negotiate with each data custodian of getting the read-access approval separately. The negotiation process between the user and each data custodian is mainly related to the privacy and ethical issues. For illustration, the Bureau of Statistics (BS) may provide a data service to deliver different datasets for conducting research studies. Due to

the privacy act, BS has to enforce different privacy policies [26] to protect the data's subject. The user has to negotiate with BS's data custodian about the privacy issues. In many cases, the price may vary based on the purposes of what the requestor is going to do with the dataset, who are the recipients of the outcomes generated from the dataset, and how long is the retention period that the requestor can keep the dataset. There may have a price differentiation for those purposes that are related to research activities conducted at colleges and commercial activities conducted in industry. Further, both parties may also have to negotiate about the disputes and remedies issues for the situation when the agreement is violated between two parties.

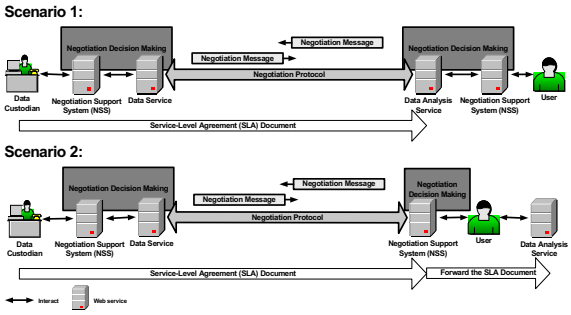

Figure 13. Two Different Negotiation Coordination Scenarios

Referring to Figure 13, there are two possible scenarios. In the scenario 1, the user interacts with NSS and then the NSS directly communicate with the data analysis service. Then the negotiation process is executed between the data service and data analysis service with certain level of human intervention. Once the negotiation process is completed successfully, the SLA document is generated and legislated between the data custodian and the data analysis service. In the scenario 2, the user interacts with NSS and then the NSS directly communicate with the data service. The user has to negotiate with the data custodian about the issue of forwarding authorization to the data analysis service for accessing the dataset. Once the negotiation process is completed successfully, the SLA document is generated and legislated between the data custodian and the user. Then the user has to forward the WSLA document to authorize the data analysis service to access the dataset. Beside our current work on the WS-Negotiation framework in the context of this use case, we are also studying the performance and practical issues between these two negotiation coordination scenarios. Based on these factors, we can design an appropriate negotiation coordination plan.

5. Conclusions and Future Work

Currently we are working on the basic model for WS-Negotiation. On the other hand, we are seeking other related researchers from different backgrounds to participate into this research activity. To explore future research cooperation opportunities, we are now trying to contact researchers in the area of NSS, SLA and semantic Web such as DAML-S. Currently we are investigating the feasibility and applicability of WS-Negotiation into the context of Agreement-based Grid Service Management (OGSI-Agreement) proposed by Global Grid Forum [35].

In this paper, we have discussed and proposed an independent declarative XML language called WS-Negotiation for Web services providers and requestors. In general, WS-Negotiation contains three parts: Negotiation Message, which describes the format for messages exchanged among negotiation parties, Negotiation Protocol, which describes the mechanism and rules that negotiation parties should follow, and Negotiation Decision Making, which is an internal and private decision process based on a cost-benefit model. This paper also presents a plug-in template model for different types of business negotiations into WS-Negotiation. This work can be expanded in several directions. In this paper, we have only discussed the scenario of one-to-one (two parties) negotiation; we are currently investigating the one-to-many (simultaneously dealing with more than two parties at one time) negotiation. We are also investigating the feasibility and applicability of WS-Negotiation with other XML workflow languages such as Business Process Execution Language for Web Services (BPEL4WS) [34].

6. References

[1] L. Thompson. "The Mind and Heart of the Negotiator," Prentice-Hall Inc., 1998.

[2] M. M. Delaney, A. Foroughi and W. C. Perkins. "An Empirical Study of the Efficacy of a Computerized Negotiation Support System (NSS)," Decision Support Systems, vol. 20, no. 3, pages 185-197, 1997.

[3] N. R. Adam and Y. Yesha. "Electronic Commerce: Current Research Issues and Applications," Springer-Verlag Berlin Heidelberg, 1996.

[4] W. W. C. Chung and J. J. F. Pak. "Implementing Negotiation Support System: Theory and Practice," Proceedings of the 30th Annual Hawaii International Conference on System Sciences, vol. 2, pages 505-513, 1997.

[5] T. X. Bui and M. F. Shakun. "Negotiation Processes, Evolutionary Systems Design, and Negotiator," Group Decision and Negotiation, vol. 5, pages 339-353, 1996.

[6] G. E. Kersten and S. J. Noronha. "WWW-based Negotiation Support: Design, Implementation, and Use," Decision Support Systems, vol. 25, no. 2, pages 135-154, 1999.

[7] A. Chavez and P. Maes. "Kasbah: An Agent

Marketplace for Buying and Selling Goods," Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology, pages 75-90, 1996.
[8] S. Matwin, S. Szpakowicz and Z. Koperczak. "Negoplan: An Expert System Shell for Negotiation Support," IEEE Expert, vol. 4, no. 4, pages 50-62, 1996.
[9] MIT Media Lab. "Tete-a-Tete," 2000. Online: ecommerce.media.mit.edu
[10] D. B. Meister. "A Prescriptive Analysis Negotiator Support System," Ph.D. Thesis, Department of Management Sciences, University of Waterloo, 1993.
[11] Universal Description, Discovery and Integration (UDDI). Online: www.uddi.org
[12] World Wide Web Consortium (W3C). Online: www.w3c.org
[13] Z. Zhang and C. Zhang. "An Improvement to Matchmaking Algorithms for Middle Agents," Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems, pages 1340-1347, 2002.
[14] L. Thompson. "Negotiation Behavior and Outcomes: Empirical Evidence and Theoretical Issues," Psychological Bulletin, vol. 108, no. 3, pages 515-532, 1990.
[15] D. G. Pruitt and S. A. Lewis. "The Psychology of Integrative Bargaining, Negotiations: Social-Psychological Perspectives," Sage Publications, Beverly Hills, London, pages 161-192, 1977.
[16] C. L. Hwang and K. Yoon. "Multiple Attribute Decision Making: Methods and Applications," Springer-Verlag Berlin Heidelberg New York, 1981.
[17] K. Karlapalem, H. P. Yeung and P. C. K. Hung. "CapBasED-AMS: A Framework for Capability Based and Event Driven Activity Management System," Proceedings of the Third International Conference on Cooperative Information Systems (CoopIS'95), pages 205-219, June 1995.
[18] P. C. K. Hung and D. K. W. Chiu, "Workflow-based Information Integration in a Web Services Environment," Proceedings of the First International Conference on Web Services (ICWS'03), Monte Carlo Resort, Las Vegas, Nevada, USA, June 23-26, 2003.
[19] IBM Corporation. "Security in a Web Services World: A Proposed Architecture and Roadmap," White Paper, Version 1.0, 2002. Online: www-106.ibm.com/developerworks/library/ws-secroad/
[20] World Wide Web Consortium (W3C). "Web Services Architecture Usage Scenarios," W3C Working Draft, 30 July 2002. Online: www.w3.org/TR/2002/WD-ws-arch-scenarios-20020730/
[21] World Wide Web Consortium (W3C). "The Platform for Privacy Preferences 1.0 (P3P1.0) Specification," W3C Recommendation, 16 April 2002. Online: www.w3.org/TR/P3P/
[22] World Wide Web Consortium (W3C). "A P3P Preference Exchange Language 1.0 (APPEL1.0)," W3C Working Draft, 15 April 2002. Online: www.w3.org/TR/P3P-preferences/
[23] OASIS ebXML Collaboration Protocol Profile and Agreement Technical Committee. "Automated Negotiation of Collaboration-Protocol Agreements Specification," Version 0.04, 2002. Online: www.oasis-open.org/committees/ebxml-cppa/negotiation
[24] H. Li. "A New Category of Business Negotiation Primitives for Bilateral Negotiation Agents and Associated Algorithm to Find Pareto Optimal Solutions," 4th International Workshop on Advanced Issues of E-Commerce and Web-based Information Systems (WECWIS), 2002.
[25] A. Sahai, A. Durante and V. Machiraju. "Towards Automated SLA Management for Web Service," HP Technical Report, 2002.
[26] World Wide Web Consortium (W3C). "The Platform for Privacy Preferences 1.0 (P3P1.0) Specification," W3C Recommendation, 16 April 2002. Online: www.w3.org/TR/P3P/
[27] S. Y. W. Su, C. Huang, J. Hammer, Y. Huang, H. Li, L. Wang, Y. Liu, C. Pluempitiwiriyawej, M. Lee and H. Lam. "An Internet-based Negotiation Server for e-commerce," the VLDB Journal, issue 10, pages 72-90, 2001.
[28] IBM Corporation. "Web Service Level Agreement (WSLA) Language Specification," Version 1.0, 2003.
[29] P. Holland. "Building Web Services From Existing Application," eAI Journal, pages 45-47, September 2002.
[30] L. Aversano, G. De Canfora, A. Lucia and P. Gallucci. "Integrating Document and Workflow Management Tools using XML and Web Technologies: A Case Study," Proceedings of Sixth European Conference on Software Maintenance and Reengineering, pages 24-33, 2002.
[31] A. Tumer, A. Dogac and H. Toroslu. "Semantic based Privacy Framework for Web Services," Proceedings of the WWW'03 workshop on E-Services and the Semantic Web (ESSW 03), Budapest, Hungary, May 2003.
[32] DAML-S. Online: www.daml.org/services/
[33] S. Y. W. Su, J. Dujmovic, D. S. Batory, S. B. Navathe and R. Elnicki. "A Cost-Benefit Decision Model: Analysis, Comparison and Selection of Data Management Systems," ACM Transactions on Database Systems, vol. 12, no. 3, pages 472-520, Sept 1987.
[34] IBM Corporation, "Business Process Execution Language for Web Services (BPEL4WS)," Version 1.0, 2002.
[35] Global Grid Forum, "Agreement-based Grid Service Management (OGSI-Agreement) model," 12 June 2003. Online: http://www.globus.org/research/papers/OGSI_Agreement_2003_06_12.pdf