NeSSy: Enabling Mass E-Negotiations of Complex Contracts

Willy Picard
Department of Information Technology
The Poznan University of Economics
Mansfelda 4, 60-854 Poznan, Poland
picard@kti.ae.poznan.pl

Abstract

In the context of economy globalization, the need for globally distributed negotiations involving a high number of negotiators communicating through the Internet becomes an important business issue. In such negotiations, the amount of information describing the negotiation process is too high to be easily understood by humans. In this paper, a prototype negotiation support system adapted to mass e-negotiations is presented. The presented prototype, named NeSSy*, is based on a negotiation model consisting of a multi-facet analysis mechanism which provides synthesized views of the negotiation process, allowing to extract knowledge concerning various aspects of the negotiation process.*

1. Introduction

Negotiation is a fundamental act in business. Every business transaction is based on a contract that has been previously negotiated. In the context of economy globalization, companies doing business with other companies all around the world need to negotiate at a global scale. Such negotiations are needed not only for multinational enterprises spread in many countries, but also for small and medium size enterprises, which are working more and more in an international environment.

Classical ways of conducting negotiations are not well adapted to negotiations at a global scale. People involved in a negotiation process are used to personally meet to exchange information and to confront their interests and goals. Personal meetings are, however, costly in terms of time and money, as well as difficult to organize, in particular if negotiators work in different countries.

With the wide acceptance of Internet, negotiators may be arbitrarily geographically distributed. Internet allows a potentially unlimited number of negotiators from the whole world to remotely negotiate on a given contract. The problem arises to organize and manage remote negotiations conducted by a high number of negotiators (of a range of a few dozens or more), denoted here *mass e-negotiations*.

An attempt to achieve this goal is to delegate the responsibility for the negotiation from a human negotiator to a computer. In such case we talk about automated negotiations. The negotiation is said to be fully automated if negotiations are conducted by software agents without human intervention in the negotiation process. Research topics involved in automated negotiation are the following [7]:

- negotiation protocols defining types of participants, valid actions, negotiation states, and events that cause negotiation states to change [1];
- establishment of ontologies defined as agreements among the negotiators about how the negotiation objects are defined and what is the meaning of these definitions. XML Schemas [11][6] and UML [4][10] have been proposed as candidates to the design of ontologies;
- decision-making models that are used by software agents to achieve their goals.

In the case of multi-attribute contracts with both aggregable attributes (e.g. price, quantity, etc.) and non-aggregable attributes (e.g., legal clauses, appendices, quality clauses, etc.), automated, humanless negotiations are not a viable solution. Software agents cannot operate on non-aggregable attributes, because of the lack of semantics concerning these attributes. We conclude that negotiations on complex multi-attribute contracts have to be conducted by humans. However, humans without any support are unable to deal with negotiations involving a high number of negotiators. The amount of data generated during such a negotiation process is too high to be understood by humans. Therefore, negotiation support systems that may facilitate mass e-negotiation processes conducted via the net are required.

A negotiation support system is particularly important in a case of mass e-negotiations because it is almost impossible to remember all the propositions made by a high number of negotiators. In mass e-negotiations, a

negotiation process is possible only if negotiators are provided with synthetic views of the negotiation process. A fundamental element of every negotiation strategy is the planning process ([8], pp. 40-51) based mainly on various analyses of the current status of negotiations. In mass e-negotiations, negotiators cannot conduct these analyses manually, because the amount of data to be analyzed is too high. Moreover, a mass e-negotiation support system has to provide negotiators with a possibility of various analyses of contract versions authored by different negotiators to well understand different aspects of a conducted negotiation process. For instance, a negotiator may want to analyze the involvement of different negotiators in the negotiation process, or analyze the correlation between a contract clause defining a delivery date and other contract clauses.

In this paper, we propose a negotiation support system adapted to mass e-negotiations. The proposed solution consists of an implementation of a multiversion contract model and a multi-facet hierarchical analysis mechanism. The multi-facet hierarchical analysis mechanism [9] provides synthetic views of the negotiation process, extracting knowledge related to various aspects of the negotiation process.

The analysis mechanism should be generic enough to allow dealing with contracts of different types. On the contrary to the ontology approach, which is a semantic one, a syntactic analysis based on versioning techniques [3] is proposed in this paper. The proposed syntactic analysis is based on relationships between negotiators' proposals. These relationships may be captured by a proper contract model. In this paper, a negotiation process is modeled as a multiversion contract. One may notice that a contract is usually modified many times until the final agreement. Various versions of the contract reflect various negotiators' propositions. Information concerning relationships between negotiators' proposals may be retrieved from the multiversion contract model and be the object of various analyses.

The paper is organized as follows. In Section 2, a multiversion contract model that addresses the problem of capturing relationships between negotiators' proposals is presented. In Section 3, a multi-facet negotiation analysis mechanism allowing extraction of knowledge concerning various aspects of the negotiation process is discussed. Section 4 concludes the paper.

2. Capturing the Relationships between Proposals

In the proposed negotiation support system, a contract consists of a number of versions. Each contract version corresponds to a negotiator's proposal. Various versions of a contract are organized hierarchically to capture the "offer↔counter-offer" relationship between proposals. The tree root is the initial contract version. When a new negotiator joins the negotiation, she/he must derive a version of an existing contract version.

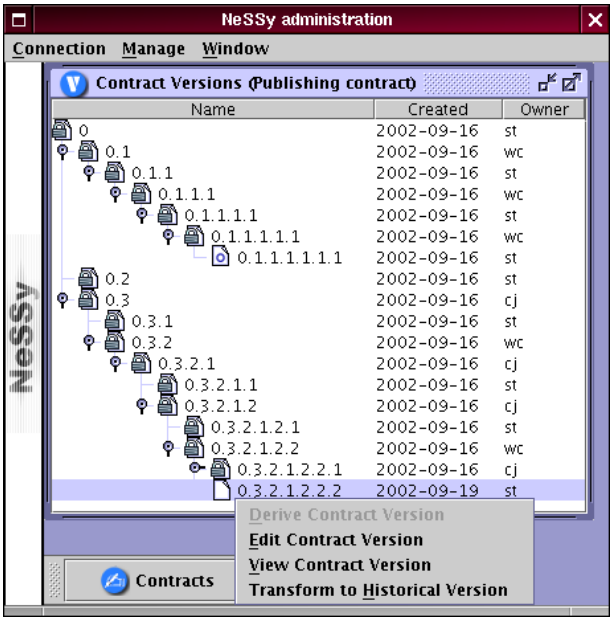

Figure1. An example of a contract version tree

An example of a contract version tree is presented in Figure 1. In this example, three negotiators are involved in the negotiation process. Negotiator st starts the negotiation process with the publication of the root version 0. From the version tree, we can deduce that negotiator wc proposes a counter-offer 0.1 to offer 0. Similarly, negotiator cj proposes a counter-offer 0.3 to offer 0.

A contract consists of *members*. A member may be, for instance, a paragraph, multimedia data, a picture or a digital signature, or a representation of the structure of the contract. In the contract model it is assumed that a multiversion contract consists of multiversion members, while a given contract version consists of given versions of these members. It is assumed that all versions of a contract are composed of the same set of members. Differences between contract versions are reduced to differences between member versions. If a member is missing in a given contract version, the version of this member in this contract version is null.

Information concerning an agreement among two or more negotiators on a given contract member may be captured by the multiversion contract model. When two or more negotiators agree on a given contract member, the same member version occurs in various contract versions. The concept of member instance is proposed to capture member version sharing among contract versions. The value of a member instance is a member version. A

member instance is associated with one or more contract versions. One-to-many relationships between member instances and contract versions are implemented as *association tables*. An association table associates each member instance with at least one contract version. An association table consists of rows, one row per member instance. Each row is a pair (memberInstanceID , set of contract versions associated with the given member instance).

Domain objects are used to model various facets of the negotiation processes. Domain objects may for instance represent the activity of negotiators, the importance of paragraphs, etc. Domain objects are generated by an Analysis Domain Function (ADF). An ADF is a function whose image is an analysis domain. A new programming language, named Analysis Domain Language (ADL), is used to define ADFs. ADL is a dialect of XML — the eXtensible Markup Language [2]. ADL is described in

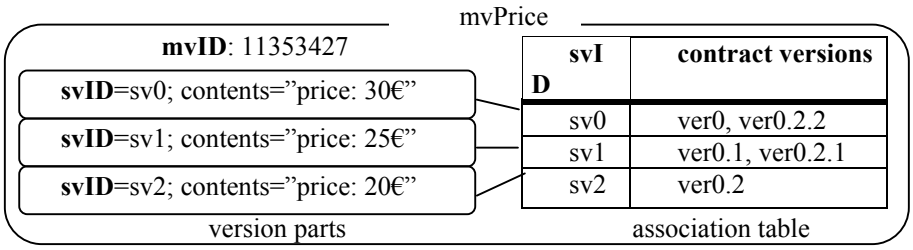

Figure 2. An example of a multiversion member of price

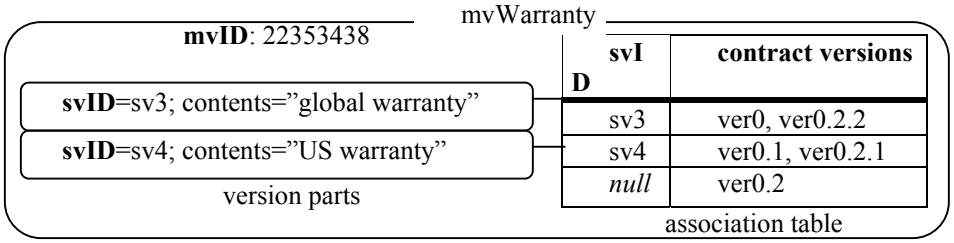

Figure 3. An example of a multiversion member of warranty

Figures 2 and 3 illustrate the representation of two multiversion members of a multiversion contracts. The contract consists of a price and a warranty. The structure of the association table for the price is presented in Figure 2, while the structure of the association table for the warranty is presented in Figure 3.

Analyzing simultaneously association tables for price and warranty, one may notice that price and warranty are dependent clauses, each new warranty corresponding to a new price and vice-versa. This result is obtained only from syntactic information, i.e. the relationships between negotiators' proposals. Therefore, association tables may be used as the basis for analysis of the negotiation process.

3. Multi-facet Analysis

The goal of multi-facet analysis is to provide a synthetic view of an aspect of the negotiation process. To analyze the multi-thread history and the current status of a negotiation process, both the abstract objects to be analyzed and the analysis criteria must be defined. Therefore, the multi-facet analysis consists of two parts: definition and retrieval of data concerning a given facet of a negotiation process, and classification of the retrieved data.

details in [9]. The ADL language allows to define new ADF processing domain objects which model new aspects of the negotiation process. The choice of a facet of a negotiation process corresponds to the choice of an ADF. The result of the execution of an ADF is an analysis domain, i.e. a set of domain objects. An ADF defines a facet of the negotiation process to be analyzed, generating domain objects modeling the given facet. The NeSSy prototype provides an ADL compiler.

In NeSSy, results of analyses are hierarchical classifications. A classification groups domain objects according to their proximity. The concept of proximity can be considered as the similarity between items. The more two items are similar, the closest their are. Therefore, an analysis criterion is a metric on a given analysis domain. Given an analysis domain and an analysis criterion operating on this analysis domain, it is possible to generate a hierarchical classification. The chosen hierarchical classification algorithm is the single-link hierarchical classification algorithm [5]. In NeSSy, analysis criteria can be also defined in ADL.

As domain objects may model complex views of a negotiation process, and interests of a given negotiator may be different from interests of other negotiators, many

analyses may be performed on the same domain objects. Having an analysis domain modeling association tables of all multiversion members, a negotiator may be interested in influence of a given multiversion member on others, while another negotiator may be interested in the degree of controversy measured as a number of versions of each multiversion member. For this reason, the concept of *parametric analysis* is proposed. An analysis is parametric if various criteria may be used to perform various analyses of a given analysis domain.

The "Classification Chooser" window, presented in Figure 4, allows negotiators to choose an analysis from among all available ones. Negotiators can choose the object of the analysis, e.g. number of paragraph version or proposed payment clauses. Having a given object of the analysis, negotiators can potentially choose various analysis criteria, e.g. relative importance of payment amount or payment delay. Various cases are presented in Figure 4. The two windows on the left side present two analyses of two different facets with a common criterion. The window on the right side presents a negotiation facet that can be analyzed according to various criteria.

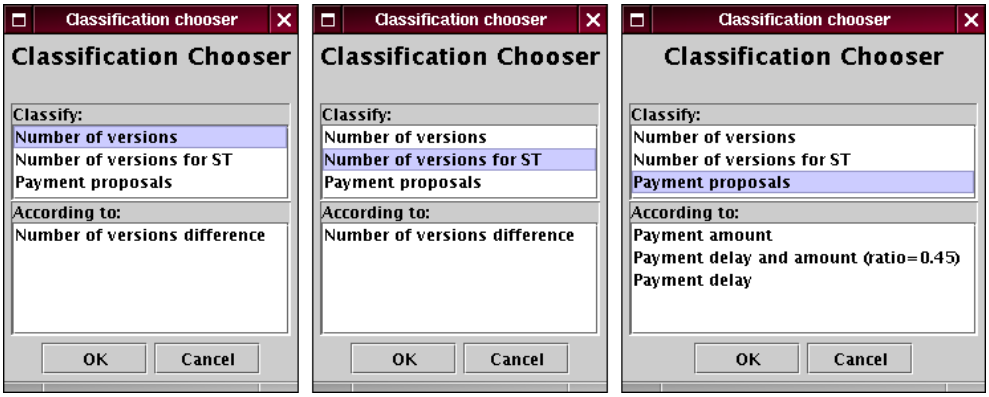

Figure 4. "Classification chooser" windows for various negotiation facets.

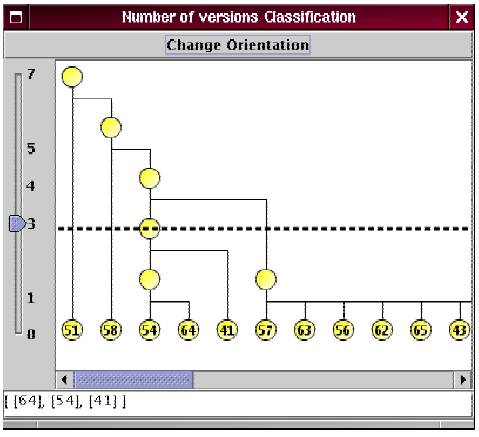

Figure 5. The "Analysis Results" window

The result of an analysis is presented in Figure 5. Three parts may be distinguished in the "Analysis Results" window. A tree in the center represents the hierarchical classification. The horizontal dashed line represents a chosen threshold. The threshold provides the granularity of the obtained partition. The higher the threshold is, the lower the number of classes in the obtained partition is. A slider on the left side allows negotiator to set a threshold at the given value. When a negotiator clicks on a class represented as a circle in the tree the value of the class is displayed on the text area at the bottom of the window. If a negotiator clicks on an atomic class, the domain objects contained in this class are displayed. Otherwise, identifiers of domain objects contained in the class are displayed, as presented in Figure 5.

4. Conclusions

The negotiation support system for mass e-negotiations presented in this paper provides tools to help conducting mass distributed negotiations via Internet, allowing a high number of geographically dispersed negotiators to work on real-life contracts.

An important feature of the multi-facet analysis approach to e-negotiation is its extensibility. Extensibility is an inherent requirement for the classification mechanism. New facets can easily be analyzed because of the use of ADL to extract and classify data. The multiversion contract model is also extensible because the structure of contracts is not fixed in the model. Therefore, advanced contract structures (e.g. tree structured contracts) may be built using the concepts of multiversion members proposed in the multiversion contract model.

The multi-facet analysis approach to e-negotiation opens new directions of research. An interesting example is application of the proposed approach to mobile computing, allowing mobile negotiators, which are potentially off-line, to analyze the negotiation process.

The proposed multiversion contract model captures various important facets of the negotiation process (such as contract member sharing) in small size structures – association tables. These structures can be send efficiently over limited-bandwidth network and can be stored in memory-limited devices like mobile phones or PDAs. Negotiators could therefore analyze some aspects of the negotiation process without having to download the whole multiversion contract. Another example is the use of software agents. Using the analysis mechanism, advanced behavior models can be build. Psychological and social models for negotiating agents may base on data retrieved from the analysis of various facets of the negotiation process. An agent may for example have a "collaborative" behavior, i.e. may look for negotiators having similar proposals to build a group of negotiators in order to increase its weight in the negotiation process.

5. References

[1] Andreoli, J.M. and Castellani S. Towards a Flexible Middleware Negotiation Facility for Distributed Components. In *Proc. of 12th International Workshop on Database and Expert Systems Applications*, Munich, 2001, pp732-736.

[2] Bray, T., Paoli, J., Sperberg-McQueen, C. M., and Maller, E. *Extensible Markup Language (XML) 1.0 (Second Edition), W3C Recommendation*, 6 Oct. 2000. http://www.w3.org/TR/2000/REC-xml-20001006.

[3] Cellary, W., Jomier G. Consistency of Versions in Object-Oriented Databases. In *Proc. of the 16th VLDB Conference*, Brisbane, Australia, August 1990.

[4] Cranefield, S. and Purvis M. UML as an Ontology Modeling Language. In *Proc. of the IJCAI-99 Workshop on Intelligent Information Integration*, Stockholm, 1999.

[5] Duda, R. and Hart P. *Pattern Classification and Scene Analysis*, New-York, Wiley, 1973.

[6] Fallside, D. C. XML Schema Part 0: Primer. W3C Recommendation, May 2 2001, http://www.w3.org/TR/xmlschema-0/

[7] Jennings, N. R., Faratin P., Lomuscio A. R., Parsons S., Sierra C., and Wooldridge M., Automated Negotiation: Prospects, Methods and Challenges. International Journal of Group Decision and Negotiation, 10(2), 2001.

[8] Lewicki, R. J., Saunders D. M., and Minton J. W. *Essentials of Negotiation*, McGraw-Hill, 2001.

[9] Picard, W. *Multi-facet Analysis of e-Negotiations*. PhD thesis, École Nationale Supérieure des Télécommunications, Paris, France, December 2002.

[10] Rumbaugh, J., Jacobson I., and Booch G. *The Unified Modeling Language Reference Manual*, Addison-Wesley, 1998.

[11] Ströbel, M. Communication Design for Electronic Negotiations on the Basis of XML Schema. In *Proc. of the 10th World Wide Web Conference*, Hong Kong, ACM Press New York, 2001, pp.9-20.