Revised version will appear in:

Proceedings of the Montreal Conference on e-Technologies
May 16-18, 2006

INSS - A New Approach in Designing Web-based Negotiation Support Systems ♦

Shikui Wu, Gregory Kersten and Morad Benyoucef

InterNeg Research Centre and John Molson School of Business,
Concordia University, Montreal,

School of Management, University of Ottawa, Ottawa, Canada

Abstract

Negotiation is vital in establishing business relationships between organizations and individuals. It is a complex, costly, time intensive and iterative process which involves heavy exchange and processing of information. E-negotiation uses information and communication technologies to automate exchange and processing. With the rapid development of online business interactions, the need for such systems has increased. Consequently, research initiatives are being focused towards designing and implementing these systems and deploying them on the Internet. INSS is a web-based e-negotiation system that allows business partners to negotiate over open and dynamically modifiable problems using different strategies and tactics. The system is based on an existing negotiation platform called Inspire although new design concepts were introduced and several features were added. This paper discusses the design and architecture of INSS and comments on a survey that was conducted to assess its usability. The novelty of our approach resides in the construction of a negotiation protocol allowing negotiators to formulate their own negotiation case and to specify the process as well as the permissible activities of the participants, and the introduction of the concepts of problem specification and issue modification in the design of the system

♦ This work has been partially supported by the Natural Sciences and Engineering Research Council, Canada and the Social Sciences and Humanities Research Council, Canada.

1. Introduction

Negotiation is defined as a decision making process by which two or more parties exchange offers in order to resolve initial differences in preferences [1-3].

The success of this decision making process is vital in establishing business-to-business (B2B) relationships and in facilitating consumer-to-consumer (C2C) commercial interactions. Indeed, “80% of commerce is performed through negotiated trade” [4], and nowadays, most negotiated trade takes place electronically, facilitated by e-commerce and e-business.

1.1 E-negotiations

Electronic negotiation (e-negotiation) takes place when the negotiating function is performed through electronic means. We talk of fully automated e-negotiation when all parties involved are software agents, semi-automated e-negotiation when a human negotiates with a software agent, and manual e-negotiation when all parties are human [5].The interest in e-negotiation is motivated by its potential to provide business partners with more efficient processes, enabling them to arrive at more satisfying agreements in less time.

The negotiation process is manually intensive thus costly to the participating parties. Using e-negotiation systems to automate it usually reduces the costs associated with it [4]. This is why the interest in designing e-negotiation systems has mainly focused on achieving higher efficiency and lower transaction costs [6].

1.2 E-negotiation systems

We distinguish three categories of e-negotiation systems [7]:

1. Negotiation support systems assist users with communication and decision-making activities;

2. Negotiation software agents replace users in their communication and decision-making activities; and

3. E-negotiation media provide a platform that implements a negotiation protocol.

This paper deals with the first category, i.e., support systems which are used to conduct e-negotiations (ENSs).

Numerous academic and commercial ESSs provide support to the negotiating parties by addressing specific negotiation problems with various techniques. Some facilitate communication and document generation; some provide analytical support; while others involve software agents or human experts as advisors. Literature reports on significant achievements in negotiation support; but, when it comes to the negotiation problem formulation, users still conduct negotiations using a fixed structure with a predefined set of issues. In real life, however, it is common for negotiators to bring new issues to the table during the negotiation process.

Some existing systems allow for changing issues (e.g., adding a new issue, dropping an existing issue, etc.). However, the intervention of a human (e.g., the system administrator, a mediator, etc.) is usually required. Thus, there is need for a system capable of supporting the negotiators directly in their

dynamic formulation of the negotiation problem.

In situations where human mediators are not employed, an ENS equipped with some intelligence (e.g., embedded in analytical models), may act as a mediator. In these situations the system formulates and suggests offers and agreements. It may also, if needed, suggest other alternatives obtained by reducing the same amount of satisfaction (e.g., utility value) for each party [8].

Support systems which do not allow for dialogue and exchange of arguments may be seen as too mechanistic. They do not allow the negotiators to participate in the discussion regarding their concessions (e.g., decrease of the utility value) and are unable to search for innovative solutions. This, in turn, may decrease the effectiveness of the negotiation and limit the use of the system. Consequently, other approaches need to be considered to cope with complex negotiations.

1.3 InterNeg Support System

The objective of this research is to design, implement, deploy and assess an e-negotiation system called INSS (InterNeg Support System) which allows negotiators to use different negotiation strategies and tactics, and to negotiate over open and dynamically modifiable problems. The proposed system is based on the Inspire system [9] (see a description in the next section), although it is significantly enhanced and expanded with new features that aim at supporting more practical negotiations, including:

1. The construction of a negotiation protocol which allows participants to formulate their own negotiation case online and to specify the process and the permissible activities of the participants;

2. The introduction of a problem specification component and a modification component;

3. The specification of different types of issues and options which can be introduced by the users and interpolated (or extrapolated) by the system; and

4. The establishment of a negotiation case library.

The INSS negotiation protocol and all its components are implemented and deployed within Inspire, a multi-protocol e-negotiation software platform [10] designed based on the Fusebox methodology which is known to provide a high level of modularity and flexibility [11]. The usability of the INSS system is assessed through a survey involving a group of participants characterized by different contexts, various levels of problem complexity and user characteristics.

The paper is organized as follows. In Section 2 we provide a brief description of the Inspire e-negotiation platform as well as other related e-negotiation systems. In Section 3 we introduce the methodology used in the design and development of INSS. In Section 4 we detail the design of the system and present and discuss its architecture. Implementation details are provided in Section 5. The complete process of assessing the system's usability is covered in Section 6. Section 7 concludes the paper and discusses future work.

2. Inspire and Other Related Systems

2.1 Three ENS types

From the perspective of the role of participants and their behaviour in negotiations, ESSs can be categorized in three broad types [12]: (1) passive systems, (2) active facilitative-mediation systems, and (3) proactive intervention- mediation systems.

As part of the InterNeg research program (http://InterNeg.org), several systems which correspond to these classes have been implemented within the SSHRC project on e-negotiation, that is, respectively, SimpleNS, Inspire, and Aspire.

The SimpleNS system provides a virtual negotiation table allowing users to exchange text-based offers and messages. It facilitates communication among negotiators but it does not provide any type of decision support such as analysis, visualization, or simulation [13].

2.2 Inspire ENS

Inspire (InterNeg Support Program for International Research Experiments) [9] is the first academic ENS for bilateral multi-attribute negotiations to be developed and deployed on the web. The system supports the exchange of structured and unstructured information between two distant negotiating parties. It also provides support to the parties in evaluating offers and counteroffers, and in viewing the negotiation history. Inspire negotiations range from simple C2C negotiations over items such as personal computers to very complex and lengthy B2B multi-issue negotiations over matters such as supply contracts.

Inspire has also been used in negotiating collective agreements between a union and management [12]. The negotiation process in Inspire goes through three distinct phases: pre-negotiation, negotiation, and post-settlement [14].

During the pre-negotiation phase, the NSS helps negotiators understand the case at hand as well as the issues involved and their possible values. Based on their own preferences, negotiators use the NSS to rate the issues and options (possible values for the issues) which leads to the construction of a utility function for each negotiator.

In the negotiation phase, negotiators use the NSS to build offers, to evaluate counteroffers by means of ratings based on the utility function, and to exchange messages. Offers and counteroffers are structured, but messages are in free text. Negotiators are able to view a graphical representation of their negotiation history. Moreover, they are offered the opportunity to revise their preference by updating the rates of the issues and options.

In the post-settlement phase, the ESS acts as a mediator and checks for the Pareto-optimality of the agreement (if an agreement is reached). In case the system finds the agreement not optimal, it suggests different alternatives to the negotiators who may then decide to resume their negotiation.

2.3 Aspire and other ENSs

Aspire is an e-negotiation system that combines decision support with software agents [14]. It is a proactive intervention-mediation system which extends Inspire with a software agent capable of retrieving information from the negotiation knowledge base and providing advice to the negotiator.

Other academic and commercial systems provide support by addressing specific negotiation problems with various techniques. For example, WebNS [15], which is a Java-based NSS, mainly facilitates communication during the negotiation process and allows the parties to discuss each issue separately. SmartSettle (www.smartsettle.com), [8, 16] provides extensive decision-analytic support and allows for the negotiation on multiple issues simultaneously. Negoisst [17] uses a document-oriented approach with semi-structured messages to form the final contract. There are also systems that allow human experts to be involved in assessing offers (e.g., www.cybertsettle.com, www.electronicourthouse.com).

3. Methodolgy

3.1 Fusebox and FLiP

The first element of the methodology is Fusebox. It is a framework for building web-based applications. It addresses development problems such as unmanageability, complexity, redundancy of effort, time-consuming code maintenance, and slow development. Fusebox views web development in terms of components (known as fuses) that are assembled to create a webpage (known as a fuseaction).

The fuses are linked together using the Fusebox engine, metaphorically similar to the electrical Fusebox found in homes. The Fusebox engine also parses elements of web pages and passes them to the ColdFusion server, which sends them to the HTTP server. The Fusebox methodology known as the Fusebox Lifecycle Process (FLiP) introduces an elegant way of managing the software development process with the aim of reducing the software failure rate. Generally, the FLiP steps are as follows [11]:

1. Wireframing: a wireframe is an initial model of the business process for which the application is being built. This step aims to identify the key activities in the system.

2. Prototyping: the creation of the HTML representation of the final application.

3. Architecture and fuse coding: based on the wireframing and prototyping, the system's requirements can be organized into circuits, fuseactions and fuses. Then, the fuses are written in ColdFusion Markup Language (CFML).

4. Unit testing: consists of test runs on single fuses.

5. Integration and deployment: this includes integrating fuses into fuseactions, assembling fuseactions under circuits, testing the application, and deploying it on the server.

3.2 Model-View-Controller

The second element of the methodology is the MVC (Model-View-Controller) design pattern [18]. The MVC design pattern deals with the overall architecture of an application by classifying its objects into three types: model objects, view objects, and controller objects. The controller receives a request from the user and determines how to process it. Model objects are then called to handle the individual processing. Finally, the view object displays the output to the user.

MVC is integrated into the Fusebox framework by way of several file types that compose the web application including the following types:

1. Model: actions (act_*.cfm) and database queries (qry_*.cfm) for business model and rules
2. View: displays (dsp_*.cfm) and layouts (lay_*.cfm) for user interface; and
3. Controller: Fusebox core files and circuit files (circuit.xml.cfm)

These three types of files can be easily included in the structured architecture of a web application, and if necessary, within a hierarchical MVC pattern [19]. The files for the model or view are fuses which are the lowest level components. Fuseactions link a set of fuses to implement a specific function of the application.

3.3 Negotiator-ENS interaction

Once the system receives a request from the user, the controller targets the corresponding fuseaction(s) to invoke the fuses to redirect a webpage, setup variables or access a database, and finally return the results to compose the web pages shown to the user.

Figure 1 illustrates the relationship among negotiation activities, components, page composers and pages in the Inspire platform.

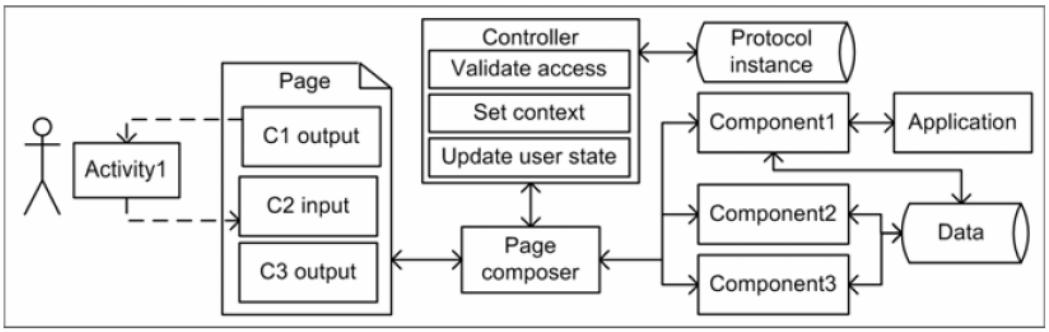

Figure 1. Activities, components, page composers, and pages in Inspire.

This conceptualization of the negotiation framework reflects the MVC design pattern, where a page composer corresponds to a view and a component reflects a model. The controller governs the sequencing of page composers. Because the conventional MVC pattern does not contain provisions for executing negotiation protocols, a "negotiation" controller is designed to execute the instances of negotiation protocols.

4. Analysis and Design

4.1 Negotiation process model

The use of software in negotiations requires that a process model and a protocol be constructed [20]. The Inspire platform enables participants to map negotiation activities to system components and construct their own protocols by creating a sequence of layouts invoking components and rules. It also provides a database-driven approach for the design of web-based systems that allow for customizable protocols [21].

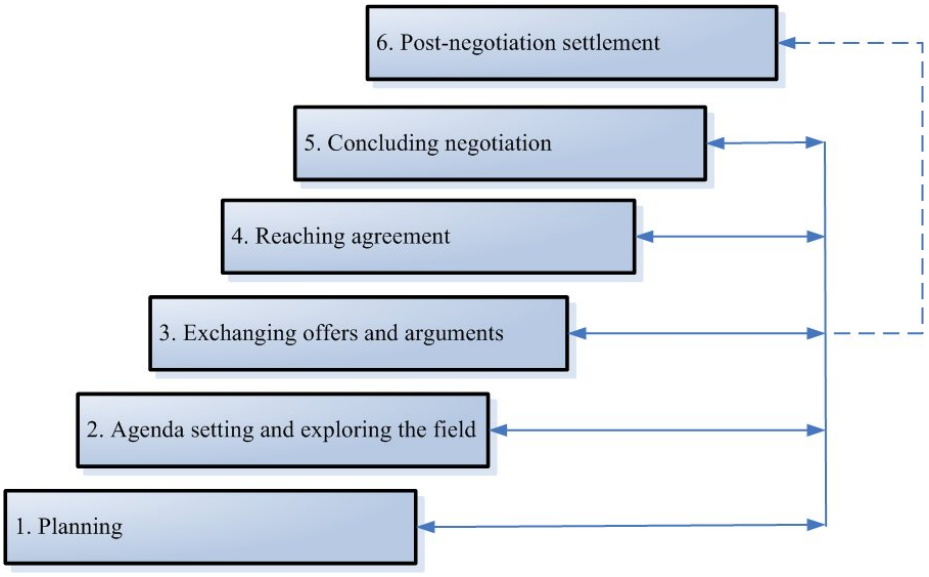

Figure 2: Six-phase negotiation process model.

The proposed system (i.e., INSS) follows the 6-phase negotiation process model used in Inspire (see Figure 2) but involves more activities. For instance, in the second phase, negotiators can add new issues and options. Note that it is not necessary for one negotiation to include all the six phases. This means that the negotiation could, for example, be terminated during Phase 3 (exchanging offers and arguments).

The six phases may overlap, i.e., negotiators may go back to revise their preferences while constructing and evaluating offers. Particularly, INSS allows negotiators to propose new issues and add options during the negotiation process, which could occur at any point as long as an agreement has not been reached.

4.2 Negotiation protocol

A top-down approach is used to construct the negotiation protocol (see Figure 3). Each negotiation can be decomposed into a set of sequences, and each sequence contains several states. In each state, negotiators can undertake one or more activities.

A participant may, for instance, start a negotiation by first reading the case information, and then deciding to propose a new issue. When the participant is in the issue proposing state, he/she can also review the current issues, add more options, etc. The negotiation protocol as described above is represented in INSS as a set of tables. The initial sequences and states for proposing new issues is shown in Table I.

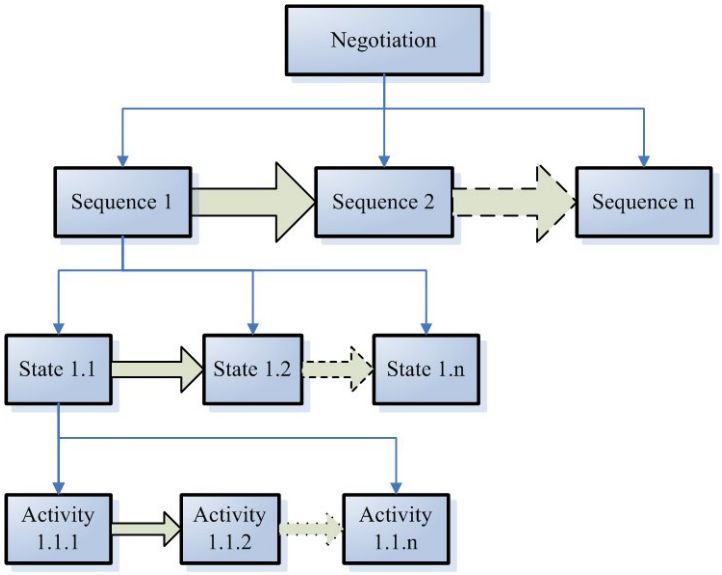

Figure 3: Top-down approach to construct a negotiation protocol.

In Table I, a negotiator proposes a new issue then the counterpart receives the proposal and decides whether to accept, reject, or modify it. The second row in the table represents a high level sequence called a block which corresponds to one of the main activities in the negotiation process model of INSS. Each of the other rows shows a sequence and its initial state, mandatory state, initial optional states and exit points.

Table I. Initial Sequences/states for Proposing New Issues

Sequence	Initial State	Mandatory State	Initial Optional States	Exit Points
Block 7				
Propose New Issue	Propose New Issue	Propose New Issue	Read Public Case Read Private Case Write Message (Message) Review Issues	Add Option
Receive Proposal	Receive Proposal	Read Issue Proposal	Read Public Case Read Private Case Write Message (Message) Review Issues	Accept Reject Modify
Accept Proposal	Accept Proposal	Accept Proposal	Accept Proposal (Issue Approval)	
Reject Proposal	Reject Proposal	Reject Proposal	Reject Proposal (Issue Reject)	
Modify Proposal	Modify Proposal	Modify Proposal	Read Public Case Read Private Case Write Message (Message)	

4.3 Intervening information and forwarding

The intervening information generated in the sequence is also shown in parentheses. Some sequences have no exit points in the initial protocol table because the negotiators are forwarded to other sequences according to the intervening rules. The intervening rules are based on the type of intervening information. Tables II and III illustrate the intervening rules for issue proposal.

Table II. Addition of Optional States for Issue Proposal

Information type	Sequence	Optional state to add
Issue Proposal	Read Private Case Propose New Issue Receive Issue Proposal Modify Issue Proposal Rate Issues Rate Options Rate Packages Exchange Offer	Read Issue Proposal

In Table II, once a new issue is proposed, if the negotiator is in one of these sequences, the 'Read Issue Proposal' state will be added as an optional state so that the negotiator can review it.

Once a negotiator receives an issue proposal, then—providing that she is in one of these sequences—she is forwarded to the 'Receive Issue Proposal' sequence. All issue proposals are listed in this sequence. This situation is illustrated in Table III.

Table III. Forwarding Table for Issue Proposal

Information Type	Origination (From sequence)	Destination (To sequence)	Destination (To State)
Issue Proposal	Read Private Case Rate Issues Rate Options Rate Packages Exchange Offer	Receive Issue Proposal	List Issue Proposal

4.4 Architecture

INSS is designed as a four-tier web application including: a client, a web server, a web application server, and a database server (see Fig. 4). The system runs on a ColdFusion application server. The negotiation controller works with the Fusebox engine to invoke page composers according to the negotiation protocol.

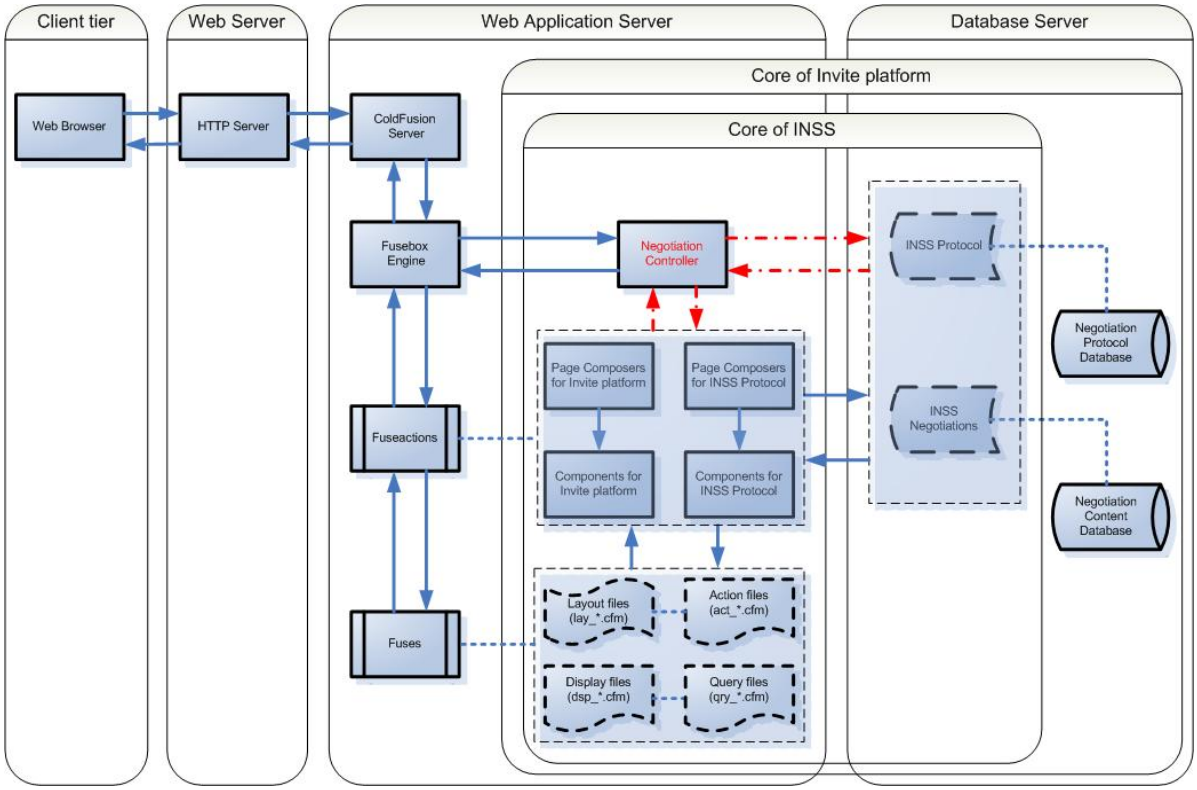

Fig. 4: Architecture of INSS.

The core of INSS includes: the page composers, the components for the negotiation protocol, and the databases.

There are two databases on the database server: the negotiation protocol database (NPDB) and the negotiation content database (NCDB). The negotiation protocol tables are deployed in the NPDB and the data about negotiations is stored in the NCDB.

5. Implementation

5.1 Database schema

A MySQL database was used to manage the tables in INSS. The NPDB schema remains the same but its content varies based on the actual negotiation protocol. To implement a negotiation protocol, the protocol tables are transferred to the NPDB. In the current implementation, the transfer is performed manually, but we envisage automating it in the future.

The NCDB schema is designed based on the INSS negotiation process model (see Fig. 5) and its content is filled up during the negotiation process.

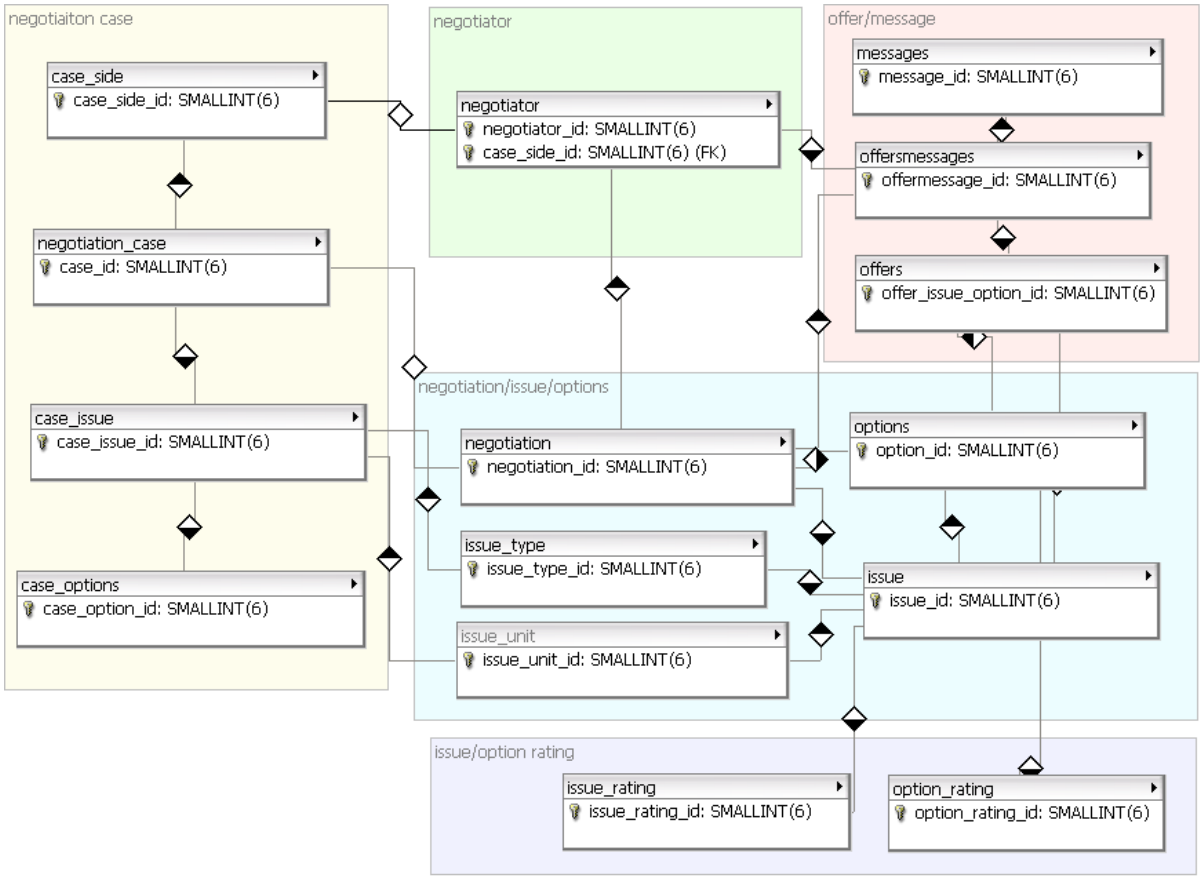

Fig. 5: NCDB Database Schema of INSS.

The completed databases are uploaded to the database server. They are specified on the ColdFusion application server server using data source names (DSNs).

5.2 Fuseactions

Depending on the states/activities in the negotiation protocol, new page composers and components are identified and developed one by one following the FLiP steps (refer to Section III). First, each page composer is decomposed into one or more components (fuseactions) by wireframing. Then the page composers as well as the components are designed. The components are used to display information or provide the user interface with specific layouts and templates.

Each fuseaction is decomposed into a set of fuses. The fuses are then coded. Once a fuse is developed, unit testing is conducted. As it is the case with component-based systems, each complete fuseaction is tested after all fuses of the fuseaction are implemented.

After the fuseactions pass the test, they are integrated into the system. There are two integration considerations.

1. At the page-composer/components level, the fuses and fuseactions are integrated to perform the activities of the INSS protocol.

2. At the system level, all page composers are integrated with the negotiation controller for execution of the protocol.

The organization of the Fusebox environment makes unit testing and integration a structured process, leaving only minor glitches to be resolved using overall testing.

System deployment refers to the assertion of the proper execution of the application in the production environment. For INSS, there are two main deployment tasks: the deployment of the protocol used to build INSS within the Inspire platform, and the deployment of INSS on the server.

6. Assessment of INSS Usability

6.1 Experiment: Itex-Cypress negotiations

People from various backgrounds and age groups were invited to negotiate through the system to verify its functionality in a hands-on approach [22]. The goal was to assess the design and implementation of INSS and to examine its main features to see whether or not it succeeds in supporting negotiators. The "Itex-Cypress" negotiation (a classic case used in Inspire) was selected as a testing case. In this particular negotiation, the participants represent two companies: "Itex Manufacturing", a producer of bicycle parts, and "Cypress Cycles", a bicycle builder. Based on [23], a three-part questionnaire (pre-test, scenario 1 post-test, and scenario 2 post-test) was designed.

Volunteers were divided into pairs. One volunteer in a pair represented Itex Manufacturing and the other represented Cypress Cycles. Participants started by completing the pre-test part of the questionnaire to determine their level of expertise in negotiation and Internet usage.

A demonstration of INSS was provided to show all the functionalities of the system, during which user comments and questions were recorded. The demonstration served to teach the volunteers how to use the system properly as well as the negotiation process. They were then directed into different rooms to start negotiations.

The first negotiation scenario involved adding new issues and options to the negotiation, and participants began by reading the public information concerning the negotiation. After reading their own private information, negotiators could review the pre-defined issues and options.

The users were asked to negotiate at least three issues and at least three options per issue. Thus, one or more issues needed to be proposed and added during the negotiation. Once an issue was added, negotiators indicated their preferences on the issues and options. Following this, they started exchanging offers and messages until the negotiation was terminated (either because an agreement was reached or because one party walked away). Finally, the volunteers were asked to respond to the second part of the questionnaire. This measured how well the INSS supports web-based negotiations (specifically for adding issues and options during the negotiation process).

6.2 Experiment: Hosting negotiations

The second scenario involved hosting negotiations and instances. This time, the volunteers were asked to negotiate on two issues with at least three options per issue. The scenario started when one user hosted a new negotiation. This user could either use the given case or create a new case for the negotiation.

If a new case was created, a minimum of two issues and at least three options for each issue needed to be added. Then, the user was required to setup one instance with another user for this negotiation. For each instance, the two users followed the same negotiation process as in the first scenario.

Once negotiations in the second scenario were completed, volunteers filled out the final part of the questionnaire to determine how well the INSS performed from their perspective (specifically for negotiating on user-defined problems).

The responses provided by the users were collected and processed. The average participants access the Internet several times a week and are familiar with buying/selling something over the web. Most of the respondents had some experience at negotiating, whereas only few of these had used e-negotiation systems before.

6.3 Discussion

In the first scenario, the general feedback showed that all of the respondents were in favour of employing INSS to conduct or prepare negotiations. The users claimed that they were satisfied with the usability of the system and rated their experience between good and great. They appreciated the fact that INSS is easy to use and issues/options can be added during negotiation. In particular, they thought it was helpful to propose new issues and add options when they felt it was hard to make a compromise on the existing negotiation issues and options.

The users commended the function by which the rating value for a new option to a quantitative issue could be calculated automatically. As an improvement, they suggested that more instructions were needed to explain the terminologies on some pages. In addition, they noted that the more issues added to the negotiation the longer the negotiation would take.

In terms of hosting negotiations and instances, most volunteers responded that the system provided a place for users to setup their very own negotiations easily and flexibly. Almost one-third of them created new negotiation cases, and more than 15% reached an agreement. This indicates that the number of initial issues and their options in the negotiation case would influence the effort and outcome of the negotiation. In addition, 93% of the participants agreed that conducting more negotiation instances on the same case is helpful to improve their negotiation outcome. They rated their satisfaction level with this feature between good and very good. Most individuals said that they would use INSS to practice or conduct negotiations in the future.

Moreover, from the negotiation history graphs and the NegoDance graphs which were generated by the system for these negotiations, we see that users negotiated using different strategies and tactics. Some negotiators started the negotiation with less issues and options and then proposed new issues and options during the process as they met a deadlock, while others started with more issues and did not change the negotiation structure too much. This shows that negotiation is influenced by the negotiators' experience, personality and culture as well.

7. Conclusion and Future Work

A review of the literature shows that as e-commerce and e-business grow and e-markets mature, negotiation is set to play a more important role. Moreover, in real-life negotiations support is needed for negotiators to deal with complex issues such as negotiation case management, changes in the negotiation structure, different types of negotiations, etc.

The paper presented a proof of concept protocol for realistic negotiation over open and dynamically modifiable problems; the construction of a negotiation support system within the existing Inspire platform; the development of page composers and components for creating and managing negotiation cases, hosting negotiations and their instances, and adding new issues/options; and a usability study of the INSS system.

The limitations of this work are mostly due to time and budget constraints as well as to the current design of the Inspire platform. In order to determine the usefulness of the system and its impact on the negotiation outcomes as well as its possible extensions, further testing on a larger scale is necessary. This may be done through experiments using other e-negotiation systems and comparing them to INSS and other e-negotiation systems (with different protocols) based on the Inspire platform.

In the meantime, more features may be added to INSS to support real-life negotiations. For example, partial offers which may not contain all negotiation issues as one package can be exchanged. In addition, some contract templates could be introduced to the case library, both for creating negotiation cases and eliciting negotiation issues. As more functions are built into Inspire, other types of negotiations can be supported in INSS, such as multi-bilateral negotiations and group negotiations.

References

[1] R. J. Lewicki, B. Barry, D. M. Saunders, and J. W. Minton, *Essentials of Negotiation*. USA: McGraw-Hill, 2004.

[2] H. Raiffa, *The Art and Science of Negotiation*. Cambridge: Harvard University Press, 1998.

[3] L. L. Thompson, *The Mind and Heart on the Negotiator*. New Jersey: Prentice Hall, 2000.

[4] "Negotiated trade: the next frontier for B2B e-commerce," Hurwitz Report, Technical Report 2000.

[5] C. Beam, A. Segev, and J. G. Shanthikumar, "Electronic negotiation through internet-based auctions," Haas School of Business, UC Berkeley, Tech. Rep. 96-WP1019, 1996.

[6] T. W. Malone, J. Yates, and R. I. Benjamin, "Electronic markets and electronic hierarchies," *Communications of the ACM*, vol. 30, pp. 483-494, 1987.

[7] M. Bichler, G. E. Kersten, and S. E. Strecker, "Towards a structured design of electronic negotiations," *Group Decision and Negotiation*, pp. 311-315, 2003.

[8] E. M. Thiessen, D. P. Loucks, and J. R. Stedinger, "Computer-assisted negotiations of water resources conflict," *Group Decision and Negotiation*, vol. 7, pp. 109-129, 1998.

[9] G. E. Kersten and S. J. Noronha, "WWW-based negotiation support: design, implementation, and use," *Decision Support Systems*, vol. 25, pp. 135-154, 1999.

[10] G. E. Kersten, S. Strecker, and K. P. Law, "Protocols for electronic negotiation systems: Theoretical foundations," in *E-commerce and Web technologies*, vol. Lecture Notes in Computer Science, K. Bauknecht, M. Bichler, and B. Proll, Eds. New York: Springer, 2004, pp. 106-115.

[11] J. Peters and N. Papovich, *Fusebox: Developing ColdFusion Applications*: Indianapolis: New Riders, 2002.

[12] G. E. Kersten, "E-negotiation systems: Interaction of people and technologies to resolve conflicts," presented at UNESCAP Third Annual Forum on Online Dispute Resolution, Melbourne, Australia, 2004.

[13] G. E. Kersten, K. P. Law, and S. E. Strecker, "A software platform for multiprotocol e-negotiations," InterNeg Research Papers, 2004.

[14] G. E. Kersten and G. Lo, "Aspire: Integration of negotiation support system and software agents for e-business negotiation," *International Journal of Internet and Enterprise Management*, vol. 3, 2003.

[15] Y. Yuan, J. B. Rose, and N. Archer, "A Web-based negotiation support system," *Electronic Markets*, vol. 8, 1998.

[16] E. M. Thiessen and D. P. Loucks, "ICANS: Interactive computer assisted negotiation support," in *Computer Assisted Negotiation and Mediation: Prospects and Limits,*, R. Shell, Ed. Cambridge, MA: Harvard, Harvard Law School, 1994.

[17] M. Schoop and C. Quix, "DOC.COM: A framework for effective negotiation support in electronic

marketplaces," *Computer Networks*, vol. 37(2), pp. 153-170, 2001.
[18] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of reusable Object-Oriented Software*. New York: Addison-Wesley, 1995.
[19] J. Cai, R. Kapila, and G. Pal, " HMVC: The layered pattern for developing strong client tiers," *Java World*, 2000.
[20] J. Kim and A. Segev, "A framework for dynamic eBusiness negotiation processes," presented at IEEE Conference on E-Commerce, 2003.
[21] G. E. Kersten and H. Lai, "Satisfiability and completeness of protocols for electronic negotiations," *European Journal of Operational Research*, (to appear).
[22] J. Whitten and L. Bentley, *Systems Analysis and Design Methods*, 4 ed. Boston: McGraw-Hill, 1998.
[23] E. Chen, G. E. Kersten, and R. Vahidov, "An e-marketplace for agent-supported commerce negotiations," *International Journal of Electronic Business*, vol. 3, pp. 28-49, 2005.