

TOWARDS OPEN ELECTRONIC CONTRACTING

BY RONALD M. LEE, ERASMUS UNIVERSITY, ROTTERDAM, THE NETHERLANDS*

ABSTRACT

A design and pilot implementation of a system, called InterProcs, supporting electronic contracting is presented. A key concept in the design of this system is the notion of electronic trade scenarios (or procedures), which are generic may be downloaded by the trading parties for a particular transaction. These scenarios may be fixed in structure, with simple parameter substitution, or they may also be designed to be customizable, within a certain range of flexibility, depending on the type of transaction. The system also includes multi-lingual text generation, in support of international trade transactions.

ELECTRONIC CONTRACTING

Electronic commerce divides roughly into two main aspects: marketing and contracting. Electronic marketing focuses on promoting the firm's products and services to customers, and is illustrated by various multi-media sites on the World Wide Web. Related innovations include on-line product catalogs, and electronic shopping malls, and electronic Yellow Pages, supporting search for products and services. Electronic contracting, on the other hand, focuses on negotiation of the terms and conditions of the contract, and the monitoring of contract performance. Current developments in electronic contracting are mainly in the area of electronic data interchange, or EDI¹. Electronic contracting may be viewed from the perspective of its 'openness' to new contracting relationships. Closed electronic contracting is the use EDI (etc) to expedite contracting among parties that already have a trading relationship established. The focus of such applications is to make customer-vendor trading more efficient. The more challenging case is open electronic contracting, which allows the formation of contracts among parties with no prior trading relationships, so called 'arm's length' transactions. Note that closed-open is actually a

* Ronald M. Lee

(RLee@Euridis.fbk.eur.nl)

is currently Director of the Erasmus University Research Institute for Decision Information System (Euridis) in Rotterdam, the Netherlands. Prior to that, he was Associate Professor of Information Systems at the Management Science and Information Systems Department at the University of Texas at Austin. Current research focuses on applications of artificial intelligence to business; special focus is 'logic modeling', the use of formal logic representations for management science applications; current projects involve the use of logic modeling and Petri nets to represent and manage formal business communications, specifically electronic document interchange (EDI), focusing on electronic commerce and governmental bureaucracies.

Theoretical aspects include the role of deontic and illocutionary logic in representing formal business conversations. Practical value is the reduction of bureaucratic and legalistic red tape. This work also includes multi-lingual business communications, supporting structured communications between parties with different native languages.

¹ In EDI, as in conventional business practice, data is organized into units corresponding to business documents, e.g. a purchase order or an invoice. As will be seen in later discussion, these documentary units have properties themselves that are significant for electronic contracting, e.g. whether the purchase order constitutes a contractual offer. Thus, the reading of "electronic document interchange" for EDI may be more apt.

continuum, allowing for intermediate cases, e.g. trading in an area where industry trading practices are well established and accepted. A central issue for open electronic contracting is in representing and negotiating the ways of doing business among the parties. For instance, is the purchase to be pre-paid or post-paid? Will an invoice be sent? Is a receipt required for delivery?

In conventional business practice, these ways of doing business are typically specified in fine print on the back of the document. Occasionally, especially where the parties are engaging in a transaction that crosses trade sector or cultural boundaries, the legal terms and conditions expressed in the fine print of their respective documents disagrees or conflicts in some way, and this conflict goes undetected until later, when some further dispute arises. This is known in the legal domain as the "battle of the forms". Thus, whereas EDI generally focuses on the front side of the document, which contains data about the transaction, our attention will be more to its back side, which contains these terms and conditions.

One approach to capture these ways of doing business in computational form is to define standard scenarios, electronic trade procedures that guide the sequence and interpretation of the sending of electronic documents. An ISO/IEC sub-committee (ISO/IEC JTC1/SC30) is working on the definition of standard, EDI based, trade scenarios (ISO, 1996). This initiative is called "Open-EDI". However, a difficulty of this approach is that these scenarios are fixed in their structure, and do not allow flexible adaptation and customization to meet the needs of a particular situation.

A key deliverable of this project is a model expert system for producing trade scenarios customized to a particular situation, yet making use of stored knowledge and experience on their design and legal controls. Thus, the focus of this project is a generalization of the Open-EDI notion of trade scenarios to an artificially intelli-

gent framework of how to construct them. In order to do this, we need not only understand the sequencing of document flows, but, more deeply, understand why these documents are sent, and what they do, that is, their legal effects (Lee, 1988; Dewitz and Lee, 1989; Dewitz, 1992). For this reason, we speak not only of trade scenarios, but more fundamentally of contracting, i.e. the exchange of commitments. Thus, we propose to develop a formal representation and knowledge base of for electronic contracting, that:

- ◆ represent trade scenarios, specifying the temporal sequence of the inter-organizational document exchanges for a given transaction;
- ◆ are computable, allowing for fully automated, computer-to-computer trade transactions;
- ◆ are customizable to fit specific contracting situations, while yet retaining the legal and control qualities of the generic version
- ◆ friendly end-user interface, whereby the terms, rules and procedures of the trade scenario can be easily read and understood by the contracting parties (as well as judges, arbitrators)
- ◆ represent the intended legal effects of the documentary communications
- ◆ have reusable components, allowing for the growth and refinement of generic knowledge bases for electronic trade scenarios.

The business problem addressed by this project is the following: Various kinds of business trading, especially those involving international trading, involve complex documentary requirements ("red tape"), as needed for legal evidence, and as required by various regulatory agencies (e.g. customs, tax). This red tape imposes high costs on transactions, and, even more importantly, creates uncertainties as to whether the transaction is legally secure. These are significant obstacles to trade, especially to small and medium sized enterprises, who do not support an in-house legal staff. The emerging technologies of electronic commerce are both an aggravation and a potential solution to this problem. They

REFERENCES

- Bons, R. "Designing Trustworthy Trade Procedures for Open Electronic Commerce", PhD Dissertation, Euridis and Faculty of Business, Erasmus University, September, 1997.
- Chen, Kuo-Tay. *Schematic Evaluation of Internal Accounting Control Systems*, University of Texas at Austin, PhD Dissertation, 1992.
- Dewitz, S.D. and Lee, R.M. "Legal Procedures as Formal Conversations: Contracting on a Performative Network." *Proceedings of the Tenth International Conference on Information Systems*, 1989.
- Dewitz, S.D. *Contracting on a Performative Network: Using Information Technology as a Legal Intermediary*, PhD Dissertation, University of Texas at Austin, 1992.
- ISO, *The Open-EDI Reference Model*, IS 14662, ISO/IEC JTC1/SC30, 1996.
- Lee, R.M. "A Logic Model for Electronic Contracting." *Decision Support Systems*, Volume 4, 1988, pp. 27-44.
- Lee, R.M.: "Dynamic Modeling of Documentary Procedures: A CASE for EDI", *Proceedings of Third International Working Conference on Dynamic Modeling of Information Systems*, Noordwijkerhout, NL, June 1992.

are an aggravation in that new legal uncertainties are introduced when trades are negotiated and performed (evidenced) electronically. However, electronic commerce also provides potential solutions, not only to these new difficulties, but to older difficulties as well.

One usually thinks of electronic commerce, especially EDI, as impacting mainly on the clerical aspects of trading, e.g. immediate sending of messages, avoidance of tran-

scription errors. Also, more recently, the World Wide Web has offered a new class of benefits, mainly in expediting search for product information. Here, however, we focus on a different kind of benefit from electronic commerce, namely in reducing red tape uncertainties and complexities for contracting.

We believe that the success of this endeavor will help to make electronic contracting much more open than conventional contracting has ever been, in that SME's can venture into new forms of trading, with distant and unknown trading partners, with confidence that the transaction is legally secure, and that adequate contractual controls are in place (Bons, 1997). Further, since these documentary requirements are generated in a computable form, they can be automatically executed by the parties' computers.

The technology approach in this project is based on artificial intelligence (AI) techniques. Its key deliverable will be to produce an expert system to generate electronic trade scenarios customized to fit the contracting situation at hand. Like other expert systems, this will involve an inference engine and a knowledge base. However, unlike other expert systems, which usually only provide advice, this project also involves a transaction system, which is able to execute the trade scenarios automatically (Lee and Dewitz, 1992; Lee and Bons, 1996).

The central point of this project is a generalization of the notion of trade scenarios to an artificially intelligent framework of how to construct them. In order to do this, we need not only understand the sequencing of document flows, but, more deeply, understand why these documents are sent, and what they do, that is, their legal effects. For this reason, we speak not only of trade scenarios, but more fundamentally of contracting, i.e. the exchange of commitments. Thus, we propose to develop a formal representation of (electronic) trade scenarios, that includes the following aspects:

- ◆ procedural representation – they are a computable representation of a trade scenario, specifying the temporal sequence of the inter-organizational document exchanges for a particular type of transaction.
- ◆ computer inter-operability – they are computable, allowing for fully automated, computer-to-computer trade transactions
- ◆ familiar, end-user interface – they are also capable of presentation in a notation that can be easily read and understood by the contracting parties
- ◆ legal (deontic) effects – in addition to document flows, they also express the intended legal effect of the documentary communications
- ◆ customizability – trading parties need to be able to customize generic trade scenarios to fit their specific situation, while yet retaining the legal and control qualities of the generic version
- ◆ reusability – the composition of trade scenarios should make use of re-usable constituent parts

A key objective for the design of trade scenarios is the inclusion of appropriate documentary controls, e.g. protecting against fraud, accident or mis-interpretation, and providing appropriate evidence of the contract status, should the contract come into dispute and go to court. As will be discussed later in more detail, these controls may be either detective, recognizing when something has gone wrong, or preventative, in avoiding the error in the first place. Usually preventative controls are preferable, but they may be too costly or cumbersome relative to the likelihood of the problem. Such cost/benefit analyses are also part of our modeling objectives. Two additional desiderata, which remain open challenges, are:

- ◆ revisability – while a given trade scenario is being 'executed' – i.e. performance of the contract is in process – it should be capable of revision, subject to constraints imposed by previous commitments in the contract (monotonic vs non-monotonic changes)

- ◆ evolvability – the knowledge base of trade scenarios should be able to evolve, based on learning and experience from past modeling. This evolution should also accommodate new control problems (scams) as they arise. This we call "evolutionary knowledge engineering" (Evoke).

Lee, R.M. "INTERPROCS: A Java-based Prototyping Environment for Distributed Electronic Trade Procedures", *Proceedings of the Hawaii International Conference on System Sciences*, January, 1998.

Lee, R.M. "Automated Generation of Electronic Procedures: Procedure Constraint Grammars", *Euridis Working Paper*, (WP 98-02-03), February, 1998b.

Lee, R.M., and Bons, R.W.H., *Soft-Coded Trade Procedures for Open-EDI*, *International Journal of Electronic Commerce*, pp. 27-50, Volume 1 Number 1, 1996.

Lee, R.M. and Dewitz, S.D. "Facilitating International Contracting: AI Extensions to EDI", *International Information Systems*, January, 1992.

Perlman, G. *Natural Artificial Languages: Low Level Processes* (ONR Report 8202). San Diego: University of California, Center for Human Information Processing, December 1982.

Peterson, J.L. *Petri Net Theory and the Modeling of Systems*, Prentice-Hall, 1981.

Wrigley, C.D., *EDI transaction protocols in international trade*, *Proceedings Conference on interorganizational systems in the global environment*, Bled, Slovenia, September, 1992.

INTERPROCS

The overall objective of this project has been to develop a general framework for electronic trade scenarios, and to demonstrate its feasibility through a realistic pilot system with real transactions based on various model scenarios. This pilot system is called InterProcs. This divides into two separate systems, one to aid in the modeling and knowledge base development, called InterProcs Designer, and the other, which executes transactions, utilizing this knowledge base, called InterProcs Executor.

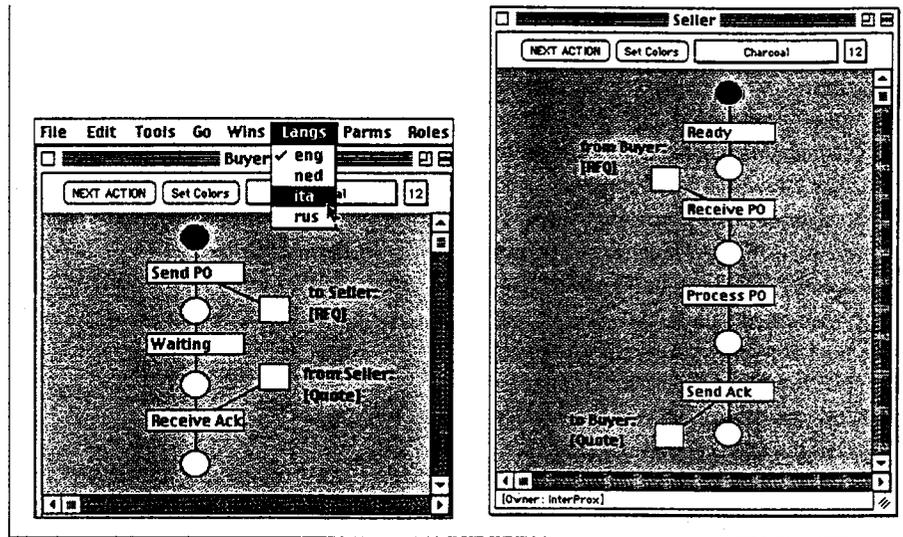
FORMAL REPRESENTATION OF TRADE SCENARIOS

The contracting process is divided (roughly) into three main phases: shopping, negotiation and performance. (While we recognize that this over-simplifies somewhat, it is sufficient for present purposes.) Interpreting this in an electronic commerce context, the shopping phase involves navigating among the product offerings of various vendors (e.g. electronic advertisements on the Web). As a result of this phase, the customer identifies a prospective vendor and specifies the product characteristics to be purchased. During the next phase, negotiation, the parties will arbitrate two things: the doing tasks of the contract (payment terms, delivery terms); and the control tasks – including documentary requirements. The output of this negotiation will be the trade scenario that the parties will follow for the execution of the contract. The negotiation process itself also follows a scenario. For many types of trading, this scenario remains more or less the same. The documentary controls for such a negotiation scenario will be such standardized (EDI) documents such as purchase order, PO acknowledgment, etc. However, in certain contract domains, additional controls are required. One example is secured loans, such as home mortgages or documentary credits. In these cases, substantial additional control documentation is needed such as credit worthiness of borrower, inspection of the asset, market value of the asset, etc.

Figure 1
Role DPN's (Buyer and Seller)
for Simple InterProcs Model

A basic issue for this project is how electronic trade scenarios should be represented (a.) from the modeler's perspective, and (b.) from a computation (inferential) perspective. In the course of our prior research, we have examined a wide number of such representations, including state-transition diagrams, marked graphs, event nets, event grammars, the event calculus, process algebras, temporal and dynamic logics. Eventually, we found Petri Nets (see e.g. Peterson, 1981) to be the most appropriate representation for capturing the temporal/dynamic aspects of electronic trade scenarios, offering both a graphical representation (for modelers) and a formal basis for the verification of various properties (computational). In addition, Petri nets have become popular in a wide variety of problem domains where sequence, contingency and concurrency of activities need to be modeled. This wide acceptance facilitates the training and understandability for electronic trade scenarios. However, Petri nets by themselves offer only a temporal framework for knowledge representation. For that reason, we have found it necessary to add various extensions to the Petri net representation, making it more appropriate for the modeling of trade scenarios, what we call Documentary Petri Nets (DPN's). The actions represented in a DPN can include the sending or receiving of a document, goods or funds, or the expiration of a deadline.

Each party (role) to the contract has a separate DPN graph, indicating its actions in the overall transaction². While DPN's are used in this project as the representation for trade scenarios, we allow for the possibility that better representations may later be found. While the prototyping environment (explained below) is built using this representation in its graphical interface, the underlying computational representation is actually predicate logic (not seen by the modeler), which enables us to support a wide variety of alternative modeling representations.



INTERPROCS DESIGNER

GRAPHICAL INTERFACE FOR
MANUAL DRAFTING

The development of an electronic trade scenario is done using InterProcs Designer³. To design a documentary petri net, the user interacts using a graphical interface. This graph is then compiled into an internal (Prolog) representation⁴. This supports the capability of simulating or actually executing these trade procedures.

The simulation mode allows the designer to verify the DPN model, and as well demonstrate it to users and clients. This can be done locally, or over the Internet.

The architecture of these electronic trade scenarios assumes that they will be distributed among multiple, distant parties. Thus, the trade procedure is a collection of separate sub-procedures, one for each

party to the transaction. The coordination among these role procedures is done exclusively by the (EDI) documents they exchange.

A tiny example⁵ of an InterProx model is displayed in Figure 1. Here a documentary petri net is shown for the buyer role, as well as a sample electronic document. Figure 2 shows a sample electronic document in English and Dutch.

PROCEDURE CONSTRAINT GRAMMARS

A limitation of the manual design approach (indeed, more generally of other Open-EDI approaches) is that the scenarios produced are fixed; that is, they cannot be adapted or adjusted to meet additional needs of a given situation. In this part of the project, we address this problem with an expert system approach, by which scenario components are broken down into reusable component parts, which can be flexibly reassembled to meet the needs of a wide variety of situations.

The computational formalism employed is called a procedure constraint grammar (PCG)⁶. As its name suggests, an objective of the PCG representation is to describe procedures by their temporal ordering constraints, rather than the absolute sequence of steps. This allows for more flexible re-combination of procedural components (doing and control tasks).

² The separation of trade procedures into distinct role graphs was suggested by Wrigley (1992).

³ Developed by Lee (1992; 1998).

⁴ A Prolog interpreter has been implemented in the Java version of InterProcs.

⁵ This and other more complex models can be seen in operation at the Euridis Web site: <http://www.euridis.fbk.eur.nl/Euridis/>

⁶ For further detail on procedure constraint grammars see Lee (1998b).

Using a procedure constraint grammar, the user interacts with the system, specifying constraints and objectives of the contracting situation. Based on these specifications, the system composes a trade procedure, which is presented in graphical form, and which can then be compiled and simulated. Here, the term 'grammar' is used in the linguistic sense of generative grammars, i.e. a set of rules for generating syntactically correct or well-formed sentences in a language. The objective of PCG rules is to generate procedures that are not only well-formed syntactically, but also from a control standpoint. In this aspect, a procedure constraint grammar operates like an expert system shell that may be used to develop knowledge bases about contracting and associated legal and documentary requirements.

Unlike language grammars, however, which are typically represented as an integrated hierarchy of rules, PCG's are organized as constraints on a target procedure. It is the job of the PCG constraint solver to identify a (minimal) solution procedure (according to some preference ordering of the user – e.g. minimal duration vs minimal risk).

We classify these tasks of the procedure into two broad categories: doing tasks and control tasks. Doing tasks are those organized to achieve the respective goals of the parties, e.g. building a building or transporting of goods. Control tasks are additional documentary actions meant to provide evidence (including an audit trail) of the parties' performance. More specifically, control tasks are means of ensuring that each of the parties conforms to their obligations in the contract. Thus, controls may be either detective – in recognizing that a contractual violation (breach) has occurred, or preventative, in verifying pre-conditions for activities (e.g. checking that someone has an import license).

In the architecture for procedure constraint grammars, we want to provide a clear separation of specification for these two kinds of tasks, doing and control. The reasons are twofold. First, the sources for these two types of activities in a contract are different: doing tasks represent the goals of the contracting parties, whereas control tasks are often imposed for legal considerations, etc. (Indeed, the interests may conflict to some extent – excessive of control tasks may add overhead that actually impedes the completion of the doing tasks.) Second, the commitments or obligations of the contract (mainly) govern the doing tasks, but not (so much) the control tasks. Thus, in mid-execution of a contract, a control task might be revised or achieved in some other manner without violating the contract terms.

INTERPROCS EXECUTOR

In addition to the knowledge engineering tools provided by InterProcs, a transaction system has been developed, called InterProcs Executor. This is a pilot version for a commercial product/service, supporting open electronic contracting (Open-EDI) for multiple concurrent transactions among distant parties. Implemented in Java, it operates over the Internet, and may be executed as an

applet, via a Web site, or as a stand-alone application. It has the following three modes of operation:

- ◆ Viewer mode – this is single-user simulation of all the roles together; it is normally executed as an applet from a Web site.
- ◆ Gaming mode – this is multi-user simulation, also executed as an applet from a Web site; each user executes a separate role of the transaction from a different computer which may be at different locations, and EDI documents are sent among the roles utilizing the Web site as a central post-office.
- ◆ Network mode – this is also a multi-user simulation, where each user executes a stand-alone version of the Executor, after downloading his/her role model for the transaction. EDI documents are sent via normal Internet email (POP). This mode is a realistic pilot of actual commercial operation, since transaction results may be stored, and other interfaces to local applications may be introduced.

The InterProcs Executor includes an electronic document facility (like other EDI servers), but most importantly, it executes transactions distributed across the various parties automatically based on electronic trade scenarios.

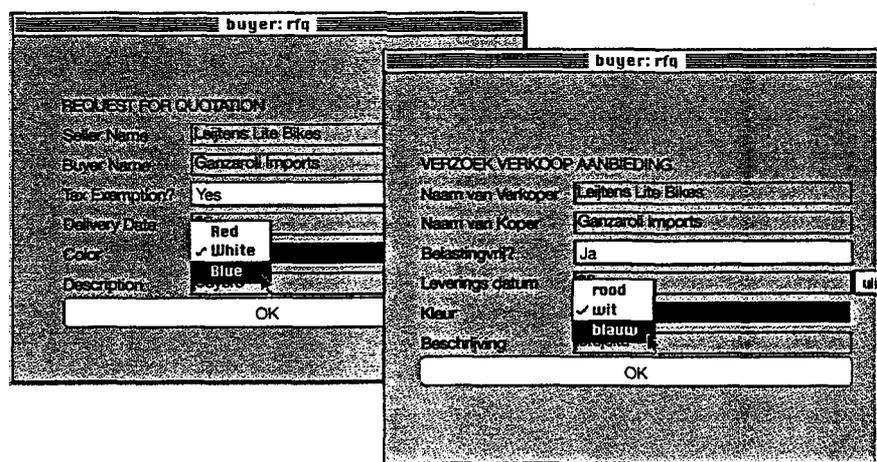


Figure 2
Sample Electronic Document:
English and Dutch

MAKING ELECTRONIC COMMERCE EASIER TO USE WITH NOVEL USER INTERFACES

BY JOSÉ L. ABAD PEIRO AND PATRICK STEIGER, IBM ZURICH RESEARCH LABORATORY, SWITZERLAND*

INTRODUCTION

THE PROBLEM

Certain security properties of electronic commerce (e-commerce) services are too complex to be fully understood by non-professional users. For example, group signatures or anonymity (Chaum 1981) with

* José L. Abad Peiro

(jla@zurich.ibm.com) has been a member of the network security research group at the IBM Zurich Research Laboratory since 1995. Prior to joining IBM, he was a network manager engineer in the communications division of the European Space Agency in Darmstadt, Germany. He is currently working towards a Ph.D. degree in applied sciences from the Catholic University of Louvain, Belgium.

Patrick Steiger (steiger@acm.org) has been member of the e-business solutions group at the IBM Zurich Research Laboratory since 1995. He is currently a Ph.D. student at the University of Zurich, Switzerland. His research is focused on increasing consumer sovereignty on the Internet by providing tools to make their needs explicit. He is working on a prototype for personal risk management.

As discussed earlier, these scenarios are of two types: fixed form (manually developed) and parameter driven (expert system) For the fixed form scenarios, users need only select the scenario, and the transaction procedure is determined. For parameter driven scenarios, a user-dialog is needed to determine the situational characteristics. As with other aspects of contract negotiation, this will proceed from one party to another in an offer, counter-offer, acceptance sequence. Once this dialog is complete, the inference engine will assemble the trade scenario.

CONCLUDING REMARKS

A design and pilot implementation of a system, called InterProcs, supporting electronic contracting has been presented. A key concept in the design of this system is the notion of electronic trade scenarios (or procedures), which are generic may be downloaded by the trading parties for a particular transaction. These scenarios may be fixed in structure, with simple parameter substitution, or they may also be designed to be customizable, within a certain range of flexibility, depending on the type of transaction.

This system, at present, has mainly been used for creating pilot models to demonstrate the usefulness and feasibility of electronic trade scenarios. These pilot applications models for an electronic negotiable bill of lading, the port community of Rotterdam, and international trade transaction models (in collaboration with the International Trade Procedures Working Group of the United Nations).

Though electronic trade scenarios have not yet appeared in actual commercial practice, we have observed that as electronic commerce turns more towards the needs for business-to-business transactions, interest in these concepts is growing quickly. We look forward to further commercial experimentation.

fair-exchange properties for online purchases are not easy to use by inexperienced users, who may not recognize the equivalent meaning in the real world, or may find that the parameters required in the protocols are too complex.

We claim that by adopting new user-interface technologies we can provide the users of electronic-commerce services with powerful and easier-to-use tools. New technologies applied to user interfaces, e.g., virtual worlds and network-based games, have been targeted to increase sales in the entertainment industry. Many suppliers on the Internet may have an interest in adapting these novel interfaces to capture a bigger share of the market, especially because many of their potential customers have grown up using video games more than wooden blocks.

In this paper we address the border between user-interface technologies and protocols used to implement secure e-commerce services, see Fig. 1. As both user interfaces and e-commerce services are increasing in complexity and functionality, we believe that a deeper reasoning for their optimal integration is required.

Our goal is to bring the marketplace closer to users by means of new interface technology in such a way that users need not be concerned about technical issues related to their communication media. For example, when a merchant is selling a product worth \$100,000, he should not have to be concerned about the bandwidth on his link or the key length of the encryption channel. The underlying e-commerce services process the transaction respecting security and legal requirements, whereas the user-interface supports the deal on a pure business level. In Fig. 2, the integration of these two elements is marked with a dashed line. This model is

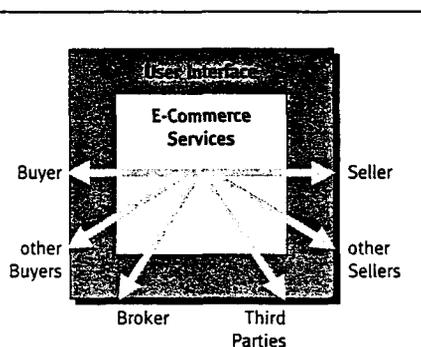


Figure 1 Model of Electronic Commerce.