Association for Information Systems
AIS Electronic Library (AISeL)

PACIS 1997 Proceedings

Pacific Asia Conference on Information Systems (PACIS)

December 1997

Supporting Electronic Commerce with Interoperable Trade Documents

Hung Wing
The University of Queensland

Robert Colomb
The University of Queensland

Supporting Electronic Commerce with Interoperable Trade Documents

Hung Wing and Robert M. Colomb

CRC for Distributed Systems Technology*
Department of Computer Science
The University of Queensland
Brisbane, Qld 4072, Australia
E-mail: wing@cs.uq.edu.au

Abstract

The practical contribution of this research is to provide a model for the purpose of specifying and representing trade information, common in electronic commerce. The conventional EDI standards such as X12 or UN/EDIFACT are too time consuming, due to the elaborate negotiation required to set up the '1st trade'. Furthermore, these standards are computationally too restricted to support trade procedures which may require regular modifications. In contrast, the Open/EDI proposals allow trade procedures to be effectively standardised to promote the 'openness' of trading. However, there are several closely related problems which have not been properly addressed. Trade problems relating to object coordination , revision, and interoperability aspects are to be investigated in this research. In particular, an electronic documentary dossier, a prototyping mechanism designed to support the above trade problems, is considered useful in facilitating electronic commerce.

1 Introduction

This paper presents a model to prototype and support the coordination and managing of trade documents common in electronic commerce.

In general, message communication problems exist at three levels (Shannon and Weaver, 1949): 1) the technical level which relates to the accuracy of sending and receiving the messages, 2) the semantic level which relates to the accuracy in conveying the message to its intended meaning, and 3) the effectiveness level associated with how effectively the received messages have achieved the desired intention (co-operation of information parcels).

We believe that the technical problem has been sufficiently dealt with due to the maturity of the various network transport mechanisms such as ONC RPC, DCE RPC, or TCP/IP etc.

In this paper, we address the semantics and the effectiveness problems relating to the organisation and managing of trade documents.

There are several attempts to solve the semantic problem. For example the conventional standard Electronic Data Interchange (or closed EDI) messages are designed to resolve some of the semantic heterogeneity problems associated with trade information. However, there are many limitations in using these standards. In particular, there are several standards in existence (eg. ANSI ASC X12, UN/EDIFACT, ODETTE etc.) therefore different versions of standards could easily escalate the cost as business grows. Other problems such as the non-existence of a guide to use the conditional field in a message and the overhead cost of unused data elements in the message's segments can be tremendous. Furthermore, there is no way to update a new standard without changing application codes, and there is no apparent mechanism to support data sharing, and the management of trades.

One of the main weaknesses of these standards is the initially high cost to establish the so called '1st trade'. The 1st trade problem refers to the elaborate negotiation required to set up the initial trade procedures ('ways of doing business') which must be agreed upon by the associated partners.

By standardising trade procedures, many benefits in regard to what can be expected from a trade scenario can be realized. Therefore, in examining the standardised trade procedures, a potential trader may know in advance the

*The work reported in this paper has been funded in part by the Cooperative Research Centres Program through the Department of the Prime Minister and Cabinet of the Commonwealth Government of Australia.

important properties of the information parcels embedded in the trade. Thus, the various business operations can be analysed and planned prior to engaging in the trade agreements. One of the key benefits in standardising trade procedures is to promote the 'openness' of trades.

In working towards an open trade environment, the standardisation of the trade procedures has been introduced in the Open/EDI proposal (ISO/IEC/JTC1/WG3, 1994).

In this paper, we introduce the key information structure - the Electronic Documentary Dossier - a compound framework containing computer process-able information such as trade procedures, different types of trades, and the EDI message protocols etc. This information structure supports the interoperability of information parcels in an open trade environment. In a sense, this electronic dossier is essentially a documentary folder that is capable of storing and managing the various types of trades and the associated commodities (eg. purchase order, letter of credit, invoice, etc.) which can be both formally represented and inter-operated across heterogeneous platforms.

As previously mentioned, the closed and open EDI proposals can be used to promote the standardisation of trade messages and procedures, respectively. However, what is lacking in these proposals are solutions to the many closely related problems which have not yet been previously addressed. In studying problems associated with :

- How the standardised trade messages and procedures can best be coordinated and managed;
- What the various information properties which can be embedded in these trade messages and procedures are;
- How these entities can be shared and distributed;
- What the practical benefits in examining and analysing these properties are;
- And most importantly, how these trade entities can be considered as 'EDI process-able' (eg. systematically interpret and process the trade information) and 'system inter-operable' (eg. populate and migrate trade information from one platform to another);

they can help us define a construct to effectively promote electronic commerce.

This paper is organised as follows: Section 2 discusses other related works. Section 3 gives an example of the electronic documentary dossier and the proposed model. Section 4 describes the implementation aspects associated with the flow of information parcels, trade semantics and interoperability of trade information. Section 5 describes the step-by-step usages and benefits of the dossier construct. Section 6 concludes this paper.

2 Motivation and Related Work

An example of a trade is an education brokerage on the Internet. In this environment, a course customer, provider, and education broker may not know each other's 'ways of doing business' (eg. trade procedures which can always be systematically processed) prior to commencing a trade such as taking or providing a course. Each participant therefore views each other's internal processes as 'black-boxes'. An integrated trade procedure can be formulated, however, after a series of trading steps and negotiations. This is an example of a 'bottom-up' approach to workflow since the relevant business process can only be formed on the fly, step-by-step. This is in contrast with the top-down approach in which an overall workflow must be known in advance and available for partitioning, monitoring and execution purposes. Associated with these advanced applications are the trade documents which represent the various structured information records necessary to instantiate a trade.

For instance, within the electronic commerce area, a *Home-Buying* dossier may contain the loan application, buyer particulars, credit information, and escrow instruction, etc. With respect to the electronic education area, a *Course-Taking* folder may be used to store the student's education needs, background, test information, and course templates, etc. In advanced trade applications, these information dossiers need to be captured, represented, collaborated, and distributed among the various trading interchanges. Based on a person X's organisational role and computing capability, X's workspace (eg. workstation, PC, or mobile computer) may store and process several such information dossiers. The idea reflected here is that each of the dossiers contains sufficient knowledge (eg. workflow's meta data) about its part of the business process in order to correctly coordinate, execute, monitor, and manage the various trade messages and documents.

In the context of the above example of the education brokerage on the Internet, we observe the following characteristics: 1) *There exist many different types of trade documents* (eg. digital cash, test templates, course credits); 2) *Trade documents are compose-able* (eg. credit information, buyer particular, and property information, etc., when combined, form a loan application); 3) *Trade documents are cross-platformed* (eg. a test template may be originally designed and formulated in a Unix-based machine in which a student who may use another totally different platform (a PC, workstation, or mobile computer for instance) can still operate on this test template); 4) *Trade documents must be EDI-process-able*, if they are to be systematically interpreted and processed.

The different documents' characteristics require different management techniques and treatments. In order to support the management and distribution of these documents across heterogeneous platforms and environments, we have to integrate and extend the many different technologies and approaches.

To support advanced applications in electronic education or commerce, the following technologies are essential:

- The Distributed Object Computing (DOC) framework allows computational objects to be widely distributed across different platforms and native environments (Sheth and Larson, 1990; Brodie, 1993). A system such as CORBA(OMG, 1995; Vinoski, 1993) supports this type of distribution. It provides a mechanism in which objects can be distributed among different network operating systems, programming languages, database management systems, and object-oriented user interfaces.
- To support groupware application, Microsoft's OLE (Brockschmidt, 1995) or Apple's Open/Doc (Williams, 1994) allow objects to be linked and embedded as a collection of loosely-coupled entities known as compound objects.

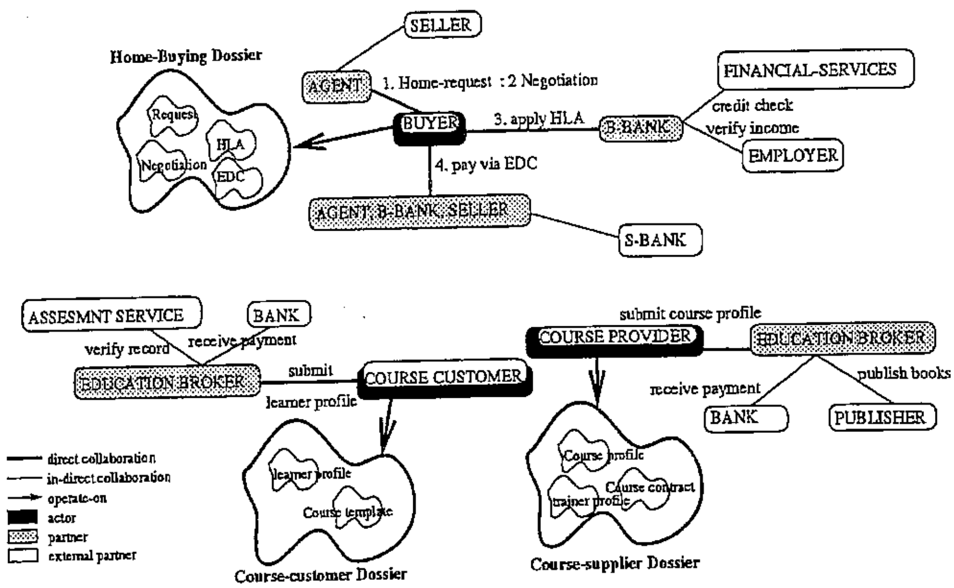


Figure 1. An example of the dossier collaboration

- To deal with the semantic heterogeneity problems (or systematically process and exchange trade messages) (Elmagarmid et al., 1993; Sheth and Larson, 1990) we need EDI's third party standard messages (eg. UN/EDIFACT or X12 (Kimberley, 1991)), closed trade procedures (Bons et al., 1994; WfMC, 1995; Dietz, 1994), and/or open trade procedures (ISO/IEC/JTC1/WG3, 1994).
- To support the specification, coordination, execution, and management of the trade activities and data, we need the technologies provided by the Workflow Management Systems (WFMSs) (Reinwald and Mohan, 1996; IBM, 1995; WfMC, 1995; Georgakopoulos et al., 1995).
- Finally, technologies provided in the Transactional Workflow systems (TWFs) (Sheth and Rusinkiewicz, 1993; Sheth and Rusinkiewicz, 1995; Alonso et al., 1994; Alonso et al., 1996) help to incorporate the well-defined failure semantics and sophisticated recovery features of the advanced transactional models (ATMs) (Elmagarmid, 1992; Zhang et al., 1994) into the workflow context.

The successful emergence of these new technologies indicates that the various inter-organisations' compound trade document and procedures will become the primary paradigm for capturing trade commodities common in electronic commerce. This is because these technologies, when combined, will become the overall framework for managing non-record oriented trade information.

Problems arise when these technologies treat work entities (eg. trade documents) as a collection of encapsulated - stand alone - computational objects. The objects' collaborated behaviours and infrastructures are often vaguely represented and somewhat ignored. The difficult task of specifying and coordinating the various work entities is left to the application, thus leaving the specification and coordination unguided to the individual trade designer. This leads to more, larger applications and added maintenance cost.

Building and supporting a simple trade application may not be cost effective due to the tremendous programming and support requirements associated with the different object models. To overcome the complexity problem, we allow trade documents to be specified, executed, monitored, and managed under one unified object model instead

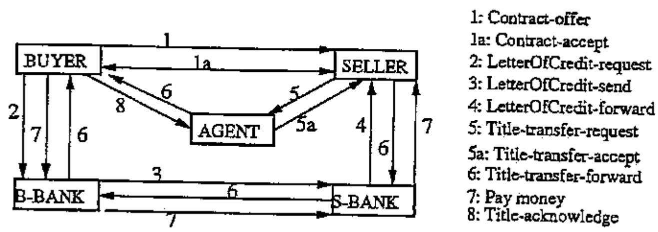


Figure 2. Electronic Documentary Credit (EDC) Dossier

of having to deal with the many different sets of localised programming abstractions and supporting environments. The architecture described in (Wing and Colomb, 1997) provides a possible implementation.

We next give an example of the dossier and present the dossier model.

3 Electronic Documentary Dossier

3.1 An example

To facilitate discussion on the organisation and representation of the trade information, let us assume that person X may act as a home-buyer, a course-customer, or a course-provider in the various trading tasks.

In the conventional way (eg. no EDI), these tasks require documents (eg. home-loan application, Letters-of-credit, course template etc.) to be manually created, exchanged, processed, and stored in an appropriate manner. For example, to apply for a home loan, X probably needs to see the bank manager, get an application form, fill out the necessary information, submit it, wait for the outcome, etc. This information can then be utilised for the purpose of recording and/or establishing other trades (eg. home buying information can later be used for filing income tax etc). The various related documents and personal reminders (eg. attached yellow pads with notes) can be organised and placed in the appropriate information drawers and folders.

This information processing and recording system is indeed simple, until it is extended to be electronically processed. All of the trade procedures and information pertaining to the trades are 'soft-coded' and computerised. Our aim is to formally represent these documents in an appropriate electronic construct.

The work described in (Wing and Colomb, 1997) specifies the underlying design and principles of the electronic documentary dossier. This paper is concerned with the characteristics of the dossier in regarding to the implementation of the above design. As an example, Figure 1 shows the dossier collaboration of person X's workspace. As suggested by the figure, X involves in the trade as the following role players (actors): home-buyer, course-customer, and course-provider.

The information represented in these dossiers can be very complex and interrelated. For example, a documentary credit may comprise a finite set of trade procedures designed for paying money and receiving goods in a secure manner (O'Hare, 1990). Figure 2 illustrates the information flow pertaining to the Electronic Documentary Credit (EDC) which is an internal part of the Home-buying dossier shown in Figure 1. The flow of the EDC shown in Figure 2 suggests that the buyer may receive the title-transfer from either B-Bank or Agent.

What we want to point out here is that in order to effectively facilitate trades in an electronic manner, the complex and useful collaborated information must be properly captured and modelled. The next section describes the proposed model.

3.2 The Model

We observe that a user's workspace can be captured and represented by its dossiers. A dossier is described as (D,A,R,E) where D represents a finite set of dossiers (eg. Home-buying, Grad-school-application, Paper-submission etc.) belonging to the workspace. The internal structures of a dossier can be described as: D comprises a finite set of acts A; an act A comprises a finite set of roles R; and each role R can be formally described by a finite set of episodes E.

Each episode is a 4-tuple(EV,IS,AC,FS) to model the Complete Semantic Unit (CSU) of a role R. In a sense, the CSUs specify the vital EDI messages which are necessary to carry out the collaborated activities required by a particular role. Each CSU is either executed completely or not at all. This feature is similar to the ACID properties popular in transaction processing (Gray and Reuter, 1993). In our work, it is designed to support the commit, abort and rollback notions associated with trades.

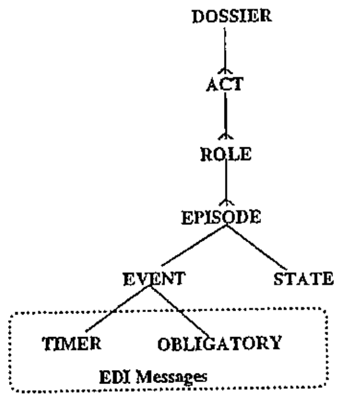


Figure 3. Dossier Model

A CSU comprises event (EV), initial state (IS), final state (FS), and action (AC). There are at least two types of events (others are being investigated): timer-event (eg. if camera-ready version of the paper is not receive within 10 days then send a reminder notice) and obligatory-event (eg. if the B-bank receives a confirmation about 'title transferred' then it is obliged to release the payment to the S-bank). The structure of these concepts is illustrated in Figure 3.

From our observation, the workhorses of a dossier are the EDI messages that are going back and forth between the various dossiers. These EDI messages are essentially the information parcels that are being sent and received at each interchange. For example, the system clock may issue an EDI message relating the timer-events; essential to the senders and receivers of information parcels are the EDI messages which are part of the obligatory-events.

To give an example of a prototype version of a dossier, shown in the appendix A are the associated entities such as acts, roles, and episodes.

3.3 The Participants

In a complex collaboration, different role players may have different purposes and capabilities associated with a dossier. For example, Figure 1 illustrates that person X may act as a home-buyer, a course-customer, and also a course-provider. He/she is the **'actor'** who wishes to carry out the various roles associated with the following trades: buying-home, taking-a-course, teaching-a-course, respectively. The **'partners'** associated with person X are the following interchanges: Agent, Buyer-Bank, Seller, and Education-Broker. Other role players are the **'associates'** (eg. financial services, seller's bank, title company etc.) who are indirectly related to the trade procedures and therefore they are considered external to person X's trading concerns. In general, the information related to these associates' ways of doing business are unknown to person X.

In order for the dossier construct to be considered useful in facilitating the global trades, it is essential that: first, the trade objects can be inter-operated across heterogeneous platforms; and second, the trade objects can be systematically understood by the different interchanges.

To achieve this goal, the next section describes the implementation aspects relating to the trade semantics and interoperability of trade information.

4 Dossier Implementation

This section examines the essential properties of a dossier with respect to its dynamic behaviours, information parcels, and role players. In examining these properties, we hope to provide an implementation model to prototype the electronic drawers and folders which can be used to store and manage the interrelated trade information.

4.1 Documentary Petri-Net (DPN)

The above model suggests that the roles capture the dynamic aspects of a trade agreement. The DPN (Bons et al., 1994), composed of petri-nets, is used to model the finite set of complete semantic units of events. These CSUs specify the vital EDI messages which are necessary to carry out the collaborated activities. As mentioned before, each CSU is either executed completely or not at all. It is designed to support the commit, abort and rollback notions associated with trades.

Since Petri Nets combine both State Transition Network and Marked Graph properties into its representation, its graphical representation can be very complex at times. There are other non-graphical methods for representing

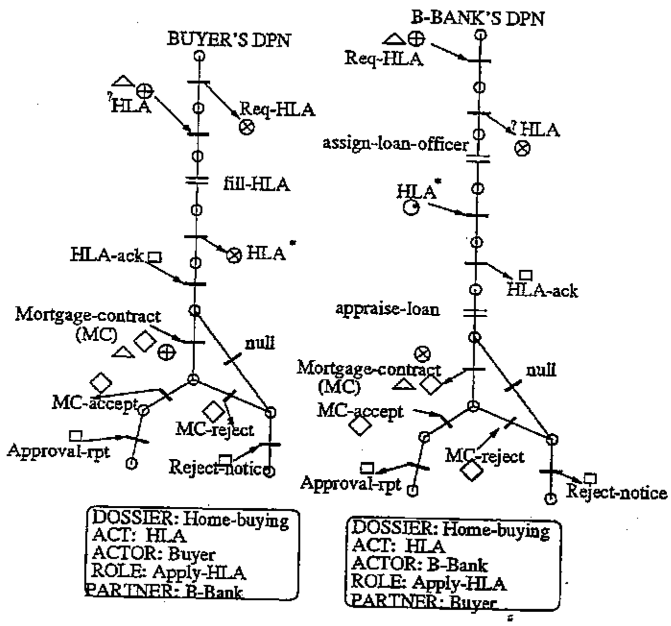


Figure 4. Graphical representation of the individual roles

process workflow such as Augmented Transition Network (ATN) and Definite Clause Grammar (DCG) which were often used as a parsing technique in context free language. In order to facilitate discussion on the dynamic behaviour of the trade procedures, we next examine how a generic role can be represented.

Graphical representation of a role As previously mentioned, a role captures the dynamic aspects of any particular part of a trade. For instance, to apply for a home loan (apply-HLA): 1) the buyer sends a req-HLA to the bank, 2) the bank then sends a sequence of query forms (denoted as ?X) 3) the buyer fills out these query forms (depicted as X*), sends them back to the bank and the process continues. The dynamic aspects of the information parcels can be captured and represented by the Documentary Petri-Net (DPN).

To give an example of the documentary petri-net, Figure 4 combines the basic properties of the petri-net and the extended notations to illustrate the graphical representation of the above apply-HLA role instance. In this figure, the incoming arrow denotes the enabling of an obligatory event, for example, by receiving an information parcel (eg. B-bank receives the Req-HLA). The outgoing arrow denotes the action(s) required for that obligatory event (eg. in receiving Req-HLA, the B-bank is obliged to send a sequence of query forms (?HLA) to the Buyer). When a query form has been successfully filled, it becomes a filled form, HLA*. An internal process is denoted by the '=' label (eg. =fill-HLA, =assign-loan-officer).

Note that the semantics of the internal process are only accessible and only have meaning to the internal actors of that DPN. For example, as illustrated in Figure 4, the buyer may know how to fill out a HLA form (depicted as =fill-HLA) but only the B-bank knows how to assign an officer to a loan request (depicted as =assign-loan-officer).

In regard to Figure 4, the following notations are used to represent the different types of information parcels which described in (Kimbrough et al., 1984): Performative (△IP), Informative (□IP), Negotiative (◇IP), Import (⊕IP), Export (⊗IP), Surrogate (⊙IP), and Forward (⊘). Detailed descriptions of these types are to be discussed in section 4.3.

Integrating roles

So far, we have illustrated that by using the documentary petri-nets we can capture and represent the dynamic aspects of the information parcels belonged to a particular trader. Based on the formal property of DPN, an integrated DPN can be formulated by combining the other related DPNs which represent the individual roles of a particular actor. As an example, shown in Figure 5 is the integrated role of Figure 4. It illustrates the combined relationship between the buyer and the bank. Figure 5 shows that the composite roles shown in Figure 4 can be integrated into one representation. This can be done by simply merging the place holders which contain the same information parcels.

Allowing roles to be composed and decomposed has many potential benefits. First of all, it enables trade agreements to be designed with either the top down or bottom up approach. The top down approach favours a close-knit group of traders (eg. stock brokers) where the overall information and central policy can be carefully designed and governed.In contrast, the bottom up approach is designed for a situation in which the individual trader is allowed to participate in a trade, with other partners, without being required to engage in an elaborate negotiation.

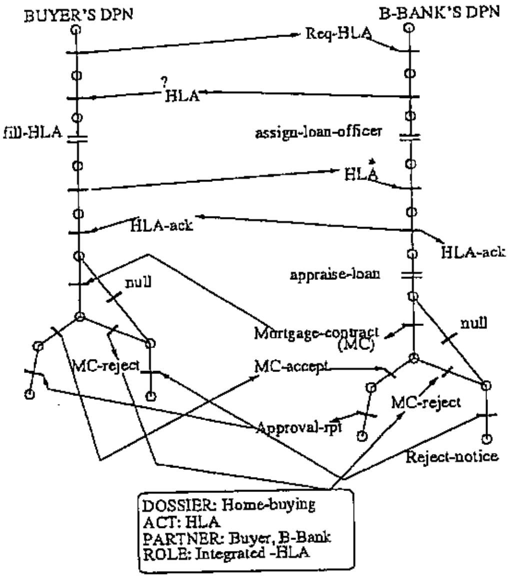


Figure 5. An integrated role

For example, the bottom up approach can be applied to the CSCW type of applications where the participants are allowed to participate in an electronic forum at any given time. A bottom up approach to distributed workflow has been discussed in (Wing et al., 1997).

Furthermore, many other interoperability benefits can be derived from the information available in the enormously complex network of many integrated DPNs - assuming that the traders are willing to co-operate in order to share the trading benefits.

For example, the many different Internal Revenue Service's related documents, when combined, form a 'Very Complex Document' (VCD). This VCD may contain many nested trade items of individual and corporate tax information; and for deduction and verification purposes, it may associate with other nested documents of home loan documents and fore-closure articles etc. With this kind of interoperability, we can see that without a good modelling technique and the 'know-how' management tools to sufficiently understand the underlying object behaviours and characteristics, the complexity involved with the managing of these objects can be very difficult. What we try to point out here is that the underlying implementation of this research can lead to other interesting research concerned with the management and implementation of VCDs.

Representing a deadlock situation

There are several reasons for using DPN to describe a trade dossier. First, DPN can be used to model the dynamic aspects of a trade procedure. Second, the formal properties of petri-net allow the application to reason about the modelled aspects. For example, by using the formal properties of petri-nets, the 'liveness' of trade procedures can be analysed. As an example, Figure 6 illustrates a deadlock situation such that if the buyers base their trades on the 'prepaid' premise and if the sellers base their trades on the 'post-paid' assumption then an ill-formed trade has been introduced to the integrated role. We need to extend the DPN tool so that this kind of situation can be easily identified and solved by the trade designers. With this purpose in mind, the graphical representation shown in Figure 6 illustrates both the ill-formed (top) and the normal form (bottom).

So far, we have used the documentary petri-nets to capture and represent the dynamic aspects of an electronic documentary dossier. In analysing these petri-nets, we can derive a collection of network activities which are essential to carrying out the trades.

4.2 Network activities

The network activities of a particular interchange can be derived by examining the roles' specifications which are defined by the DPNs. More specifically, the collections of events and actions specified in the DPNs' episodes formulate the network activities. As an example, shown in appendix B are the network activities that are derived from the events and actions of the episodes shown in appendix A.

By examining the network activities we can further analyse and characterise the information parcels associated with the trades. In the following, we describe the different characteristics of the trade information.

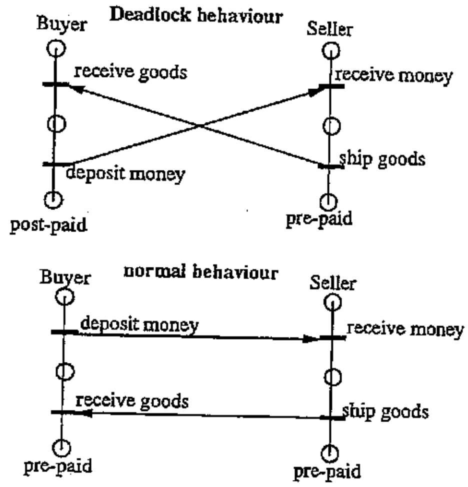


Figure 6. Deadlock and normal behaviour

4.3 Information Parcels

In general, a copy of a book is still considered a book. However, a copy of a check is no longer valid as a check. In the paper word (non-EDI), a copy of a check can be recognised due to the stamps that are marked on the check or due to other authenticating marks. What we try to point out here is that it would be very useful for a user to understand the various characteristics inherent in the underlying information. For example, it would be a violation of the trade agreements if the users attempted to make a copy of an 'electronic check'. In the following, we attempt to characterise and represent the various information types.

Performative, informative and negotiative information: The various works in speech act theory (Searle, 1969; Kimbrough et al., 1984) have basically established that an information parcel can be classified into one of the following three categories: performative, informative and negotiative. Performative information causes other activities to be performed (eg. to authorise a loan approval). Informative parcels are used for the purpose of informing others about a specific piece of information (eg. a loan-approval-report or loan-reject-notice). Negotiative information parcels are used to establish an obligatory situation during an exchange (eg. if the seller receives a LC he/she is obliged to send goods; or if the buyer receives the sale title he/she is obliged to pay money). We next examine the other information types relating to the collaboration.

Import, export and surrogate information: There are other information types that are also important to characterise. In a collaborated environment, the information may flow from one interchange to another. The success of a collaboration greatly depends on a mechanism that can guarantee the 'authenticity' of the traded information.

Information parcel can be one of the following types: import, export and surrogate. For example, imported into the buyer's dossier are the query forms (depicted as ?HLA in Figure 4) which can be used by the applicant to provide the necessary information pertaining to the application. The original conceptual schemas of these queries are owned and managed by the bank's DPN. From the B-bank's point of view, these same query forms are considered as export queries.

The surrogation mechanism (Fang et al., 1991) is used to refer to the 'remote' objects by using 'local' database system tools without modification. In general, a surrogate information parcel could be used to prevent an un-authorised use and modification. For example, when the ?HLA query has been successfully filled out, it becomes an 'export-able' piece of information (HLA*). This information parcel is considered a surrogate within the B-Bank's DPN.

Forward information: Similarly, the 'Forward information' is used as a mechanism to forward an information parcel without being able to access its content.

For example, the recommender may forward a reference letter to a student so he/she may, later, issue this letter to the suitable registrars without the student being able to see the recommended information. For the purpose of this paper, the CASE tool should be able to send and receive 'forward information' based on the header of the information parcel. The content of the information parcel can be encrypted with a private key mechanism, and only the selected registrars may have the necessary key to open it.

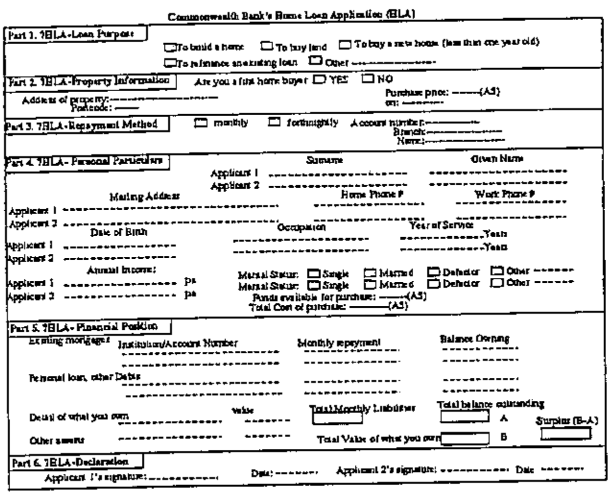
4.4 Interoperability in trades

In order to promote the global trade environment, one of the essential requirements is to ensure that trade commodities can be distributed among heterogeneous platforms.

OMG's Object Management Architecture (OMA) can be used to solve this kind of problem. Essentially, " by providing granular levels of interoperability between objects in heterogeneous distributed environments, it will turn everything into nails and it will give everyone a hammer" (Orfali et al., 1994). In our application, the 'nail' is the information parcel (eg. mortgage contract, loan approval report etc.) which can be defined in the CORBA Interface Definition Language (IDL). The 'hammer' is the OMG's Object Services which support the basic functions such as finding which trade objects are on the network, which methods they provide, etc. The CORBA IDL is the language used to describe the interfaces that client objects call and object implementations provide. It has the same lexical rules as C++, with the addition of new keywords which can be used to support distribution concepts.

For example, shown in appendix C are the CORBA objects associated with network activities shown in appendix B. These CORBA objects are used to support the an application form shown in Figure 7. What we want to illustrate is that, regardless of what system or platform the buyer and the bank use, the loan application can still be operated across different environments, because the loan application has been implemented as a CORBA object in which the object's services can be provided to the client with the support of the OMA architecture.

Another essential requirement is that the trade commodities must be semantically understood by the different trading interchanges. The next section describes how this can be done.



Commonwealth Bank's Home Loan Application (HLA)

Part 1. 7HLA-Loan Purpose
☐ To build a home ☐ To buy land ☐ To buy a new home (less than one year old)
☐ To refinance an existing loan ☐ Other ----------------

Part 2. 7HLA-Property Information
Are you a first home buyer ☐ YES ☐ NO
Address of property: ----------------
Postcode: -----
Purchase price: ------(A$)
on: -------

Part 3. 7HLA-Repayment Method
☐ monthly ☐ forthnightly
Account number: ----------
Branch: ----------
Name: ----------

Part 4. 7HLA- Personal Particulars
Surname / Given Name
Applicant 1 ---------------- ----------------
Applicant 2 ---------------- ----------------
Mailing Address / Home Phone # / Work Phone #
Applicant 1 ---------------- ---------- ----------
Applicant 2 ---------------- ---------- ----------
Date of Birth / Occupation / Year of Service
Applicant 1 ---------- ---------- ----------Years
Applicant 2 ---------- ---------- ----------Years
Annual Income:
Applicant 1 ---------- pa
Applicant 2 ---------- pa
Marital Status: ☐ Single ☐ Married ☐ Defacto ☐ Other ------
Marital Status: ☐ Single ☐ Married ☐ Defacto ☐ Other ------
Funds available for purchase: ------(A$)
Total Cost of purchase: ------(A$)

Part 5. 7HLA- Financial Position
Existing mortgages / Institution/Account Number / Monthly repayment / Balance Owning
Personal loan, other Debts
Detail of what you own / Value / Total Monthly Liabilities / Total balance outstanding A / Surplus (B-A)
Other assets / Total Value of what you own B

Part 6. 7HLA-Declaration
Applicant 1's signature: ---------- Date: ------ Applicant 2's signature: ---------- Date ------

Figure 7. An example of a home loan application

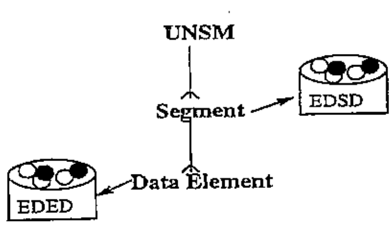


Figure 8. UN/EDIFACT Standard Message Structure

4.5 Trade semantics

Previously we have shown that in using CORBA IDL to assemble trade information, we ensure that trade commodities can be inter-operated across different platforms. Equally important, we must also ensure that trade objects can be systematically processed by the associated interchanges. In order to achieve this goal, we package the information parcels into a series of low level EDI message protocols.

Essentially, what the network must transmit between the various interchanges are the standard EDI messages, groups of segments and data elements. To show the structure of these EDI messages, Figure 8 illustrates that an UN/EDIFACT's Standard Message (UNSM) can be potentially composed from the different groups of segments stored in the standard EDI Segment Directories (EDSDs). Embedded in these segments are the various data elements which can be precisely described by the standard EDI Data Element Directories (EDEDs).

Shown in appendix D are the various standard repositories which can be used to support the different acts and roles. Associated with each act is a collection of standard messages in which their structures and definitions can be derived from the associated standard directories. Appendix E illustrates a sample UN/EDIFACT message (CREADV) which can be used to construct the letter of credit, an information parcel part of the pay-EDC role. The next lower message protocol is the message segment.

A standard EDI message can be further decomposed into a series of group segments which are used to specify the details of the message. Shown in appendix F are the two segment samples NAD (name and address) and EMP (employment details). These segments can be used to specify the personal particular part of the home loan application. The next lower protocol, after the segment, is the data element which describes the basic data type such as alphabetic, numeric, and alphanumeric etc. Appendix G shows the format notations of these data elements.

Provide a construct to promote shared segments. At this stage we would like to point out that because each act and its associated roles are being supported by a collection of EDI standard repositories contained messages, segments and data elements, it would be useful to have a surrogation mechanism similar to the one described in (Wing and Colomb, 1994; Fang et al., 1991) to support data sharing and integrity constraint.

In the context of this research, surrogation refers to the remote objects by using local database system tools without modification. With this kind of surrogation mechanism, it is possible to share trade data such as application objects, messages and segments etc. For example, containing in person X's workspace, the Name and Address (NAD) segment can be shared within a single dossier. Thus, a surrogate object allows trade messages and segments be dynamically linked into the trade applications.

So far, what we have described is the dossier model, an electronic documentary folder, a construct to facilitate electronic commerce. What we have described shows how this model can be implemented as a collection of inter-operable objects and low level EDI message protocols. The next section describes the different types of trades and the potential benefits in using these electronic documentary dossiers.

5 Usages and benefits

In this section, we want to show how the different types of trades can be supported by the dossier construct. Also pointed out are the associated benefits.

In the context of this research, we categorise trades into three different types. 1) The '1st trade' refers to the exchange of information in which such a trade has never before been carried out. For example, person X may wishes to buy his/her first home via electronic means. 2) The 'established trade' refers to the exchange of information in which that similar trade has been carried out before. An example of such a trade is the ordering of a book from one of the discount book brokers. In this scenario, the user's access and ways of doing business have been previously examined and carried out. 3) The 'established trade with revision' refers to a scenario in which the trade has been previously carried out but due to the new business requirements, the trade must be altered. For example, a book buyer may wish to place a time constraint in ordering a book.

The following sub-section describes, step-by-step, how we can overcome the 1st trade problem.

5.1 Overcoming the 1st trade problem

We will assume that, person X wants to subscribe to the EDI server as a home buyer, a course customer, and also a course provider in order to participate, for the first time, in the act of buying a home, taking a course, and teaching a course. Assuming that the dossier construct has been successfully introduced and implemented in the global trade environment, person X should have no problem in 'systematically' carrying out the trades with much efficiency. What we mean systematically is that in a long lived transaction such as buying a home, it may involve many other sub-transactions associated with manual activities (eg. fill out an application form) and 'thinking gap' (eg. decision to when is the best time to submit the loan application). It is our attempt to minimise these manual activities and to remove (eg. by providing baby-stepping transactions) the thinking gaps out of the transaction loop as much as possible. In doing so, we can better predict and enhance the dynamic aspects of the flow of information parcels.

• **Step 1, Making contact:** The person X contacts the EDI server (eg. via email or other EDI protocols) to request trade procedures under particular trading scopes. For example, the trade procedure for buy-home can be found under the (business(real-estate(buying))) catalog of hierarchies.

Down loading from the EDI server to X are the following dossiers:

```
Dossier: HOME
ACT: buy-home
```

```
ACTOR: buyer
ROLE: make-request, involve-negotiation,
      apply-HLA ,pay-EDC
PARTNER: agent, b-bank, seller

Dossier: STUDY
ACT: study-course
ACTOR: course-customer
ROLE: select-school, log-application, request-referrals
PARTNER: education-broker, admission, recommender

Dossier: TEACH
ACT: teach-course
ACTOR: course-provider
ROLE: submit-course-template, provide-course-profile
PARTNER: Student, education-broker
```

The various partners can be involved in the above trades in one of the following two ways: 1) The actor designates the partner to be involved with the trade. This can be done via the white page of the trader service. Note that an interchange, represented as a node in the global network, can subscribe to a collection of roles depending on the interchange's business requirements. In our example, person X may subscribe to the EDI server as a buyer, a course customer and a course provider in order to participate in the following acts: buy-home, study-course, and teach-course, respectively. 2) The potential partners can voluntarily make themselves be available for the trades. This can be done via the yellow page of the trader service.

At this stage, each partner should have his/her own dossiers based on the role he/she must play. For example, the b-bank is associated with the buyer via the apply-HLA and pay-EDC role and the agent is associated with the buyer via the make-request and involve-negotiation role. The dynamic aspects of these dossiers may be made known to the partners but not to their associates. For example, X may know what sort of information can be exchanged between the bank and X. In contrast, the information exchanged between the bank and financial services (external to X) may not be known to X.

We should make a note that, due to the private nature in business, the overall integrated trade procedure (DPNs) in which the actor may have access to the associates' ways of doing business may not be available; however, they are technically possible. We envisage that an integrated DPN model is possible for relatively smaller and tightly controlled group of traders where each trader may have similar interests and benefits resulting from the co-operation.

• **Step 2, EDI mapping:** This step describes the framework in which it can support the mapping of the various EDI trade messages. For example, the home loan information provided by the buyer can be electronically transmitted and systematically processed by the bank's computer system, thus shortening the processing time.

In step 1, down loaded with the DPNs (from the EDI server) are the associated standard repositories which can be used to support the acts, roles of a dossier. For example, associated with apply-HLA role there are the EDSDs (concerning with message segments) and EDEDs (concerning with data elements). In this way, each act (and subsequently each role) has a collection of standard protocols which can be used to define the trade objects.

By dividing the standard repositories into smaller sets of definitions associated with each act and role, consequently we provide a scoping mechanism to support trade heterogeneity. Thus, we now no longer depend on the overly complex and centrally large UN/EDIFACT standard directories to promote trades. What we do here is to subscribe and down load the only definitions which are necessary to support the related acts and roles. This is where the proposed model gains efficiency in using the standard definitions.

Under this construct, each partner has the flexibility to design the underlying objects as long as the objects are being composed by the definitions given by the associated standard repositories. For example, the Commonwealth bank of Australia and First City bank of New York may have different types of queries and forms associated with their home loan applications. However, the applications' queries and forms can be correctly interpreted and the associated information can be shared due to the inter-related standard repositories.

At this stage, each actor should have the associated dossiers which can be prototyped similar to the one shown in appendix A.

• **Step 3, Analysing network activities:** The existence of DPNs results from step 2 above allowing the network activities such as send(info-parcel, partner), receive(info-parcel, partner), forward(info-parcel, partner), request(info-parcel, partner) etc. to be formulated and examined.

For example, some integrity constraints can be enforced due to the nature of the activities and their associated information parcels. For instance, a carrier (eg. third party VANs), who is forwarding a letter-of-credit (LC) to the bank, should not be able to access the content of the LC. This is possible since the LC's body could be encrypted with a private key mechanism. The LC's header may contain the destination address which is made available to all of the involved partners. However, only the designated bank may have the necessary key to open its content.

As an example, shown in appendix B are the network activities associated with the apply-HLA role.

• **Step 4, Making it inter-operable:** We have indicated that the standard objects associated with the above activities must be both EDI capable and system inter-operable.

In dealing with the heterogeneous environments, the objects must be capable of inter-operating across different platforms. It means that the object should be packaged in CORBA IDL so that the objects' services and attributes could be exported to client applications, without worry about the underlying supporting platforms. This step can be done during the design of the trade procedures as discussed in section 4.1. In this section, it is essential that the trade objects, created in the bottom up or top down approach, be defined and assembled in CORBA IDL. This is how we can turn trade objects into nails, and the CORBA's object services provide traders with the necessary hammers!

As an example, shown in appendix C are the CORBA objects associated with the apply-HLA role's network activities.

• **Step 5, Overcoming the semantic heterogeneity problem:** In dealing with trade semantics, the related objects must be capable of being electronically processed (EDI capable) at the ends of other interchanges. This can be achieved by using the EDI messages and standard trade protocols. Note that, it is possible to have the message segments be shared within a single dossier of different environments.

As mentioned in section 4.5, by using the surrogation mechanism similar to the one described in (Wing and Colomb, 1994; Fang et al., 1991) we can facilitate the sharing of trade messages and segments in a single dossier belonged to the different interchanges.

Now we have illustrated that by providing sufficient resources such as EDI server, standard repositories and dossier construct we can overcome the so-called 1st trade problem. This contribution enhances the 'openness' of the global trade environment, thus enabling the Small-to-Medium size Enterprises (SMEs) and individual net users to participate in electronic commerce without the elaborate negotiation required to set up the initial trades.

As an example, shown in appendix D are the standard repositories associated with the following roles and acts: apply-home-loan, apply-grad-school, and make-publication. Also shown in appendix E, F, and G are the samples of the segmented standard UN/EDIFACT message, segment, and data element, respectively.

5.2 Carrying out an established trade

Once a trade has been carefully designed and tested through the 1st trade encounter, it then becomes an established trade.

In general, established trade is designed for transactions that are to be carried out on a regular basis. For example, every sold item in a retail store can be automatically recorded and re-shelved by the trade agreements established between the retail outlets and the suppliers. Another established trade is the EFTPOS banking scenario in which an account's fund can be regularly withdrawn and/or deposited at an ATM.

The difference between the established trade and the 1st trade is this:

• In the established trade, the DPN number (analogous to OID in OODBMS) and its associated partners, EDI message and segment directories are already established and - continuously to be managed - by the EDI server. In contrast with the 1st trade, the DPN number and its associated partners, EDI message and segment repositories are to be formulated during the trade instantiation. To simplify and ease the sharing of computing resources, these associated entities are to be discarded from the EDI server environment after the trade is completed. However, definitions of these standard messages, segments and objects are to be kept in local storages for future interpretations of the dossiers.

• For better security and faster services, established trade may decide to use third party VANs to enhance their trading capabilities. For example, Ford automotive manufacturer may carry out trades with its suppliers and sub-contractors via the private FORDNET protocols.

5.3 Participating in a trade with revision

In order to make trades more flexible and adaptable to the rapid changes of business, we need a mechanism to allow trades to be easily revised and to continue to operate with new updates. For example, an ATM user wishes to have the overdraft protection incorporated into all of his/her future withdrawals. Another example relating to the above home loan application is that the bank may want to impose a constraint in which the mortgage contract is invalid after 7 days if the buyer still has not authorised and returned the contract offer.

These new business agreements could be easily added to the appropriate DPNs, and repositories of all the associated traders. Each local DPN and repository may be associated with a revision number which references to the different updated circumstances and constraints.

6 Conclusion

To facilitate effective communication in electronic commerce we need a protocol to overcome the interoperability and semantics heterogeneity problems associated with a trade. Consequently, we need an unambiguous formal language to represent and reason about the trade objects.

In this paper, we have investigated the essential requirements and characteristics related to the electronic documentary dossier. We reason that the semantic and structural information associated with trade instances can be very complex at times. For example, the formal property of Petri Nets can be used to capture and represent the different traders' roles or to identify the trade procedures that may lead to deadlock situations. After carefully examining and analysing the various properties associated with trade procedures, network activities and information parcels, we provide a model to formally represent an electronic folder. This folder is designed for the purpose of coordinating and managing trade information.

In general, translating an unambiguous formal language such as conceptual graph (CG) (Sowa, 1984) to a trading language, or procedure, is possible but not in the reverse order due to the ambiguity inherent in the different trading languages. However, the ambiguity of the trading languages can be clarified by using the standard message protocols (eg. UN/EDIFACT) or procedures (eg. DPN). Thus by packaging the context of a trade into a collection of information dossiers representing the different standard protocols and procedures we can establish an effective communication within a trading system.

Furthermore, for every petri-net representation we can found an equivalence conceptual graph (CG) representation (Sowa, 1993). Thus, by using DPN and CG to model and reason about a trade we have purposely provided our model with reasonable abstraction, generality, readability and logical foundation.

References

Alonso, G., Agrawal, D., Abbadi, A., Kamath, K., Günthör, R., and Mohan, C. (1994). Failure Handling in Large Scale Workflow Management Systems. IBM Research Report RJ9913, IBM Almaden Research Center. available in http://www.almaden.ibm.com /cs/exotica.

Alonso, G., Agrawal, D., Abbadi, A., Kamath, K., Günthör, R., and Mohan, C. (1996). Advanced transaction models in workflow contexts. In *Proc. IEEE 12th International Conference on Data Engineering*.

Bons, R., Lee, R., Wagenaar, R., and Wrigley, C. (1994). Modelling Inter-organizational Trade Procedures Using Documentary Petri Nets. In *The Hawaii International Conference on System Sciences*. about DPN.

Brockschmidt, K. (1995). *Inside OLE, 2nd Edition*. Microsoft Press, Redmond, WA.

Brodie, M. L. (1993). The promise of distributed computing and the challenges of legacy information systems. In Hsiao, D., Neuhold, E. J., and Sacks-Davis, R., editors, *Proc. IFIP TC2/WG2.6 Conference on Semantics of Interoperable Database Systems*, Amsterdam.

Dietz, J. L. (1994). Business modelling for business redesign. In *Proceedings of 27th Hawaii International Conference on System Sciences (HICSS)*.

Elmagarmid, A., editor (1992). *Database Transaction Models For Advanced Applications*. Morgan Kaufmann.

Elmagarmid, A., Chen, J., and Bukhres, O. (1993). Remote System Interfaces: An approach to overcoming the heterogeneity barrier and retaining local autonomy in the integration of heterogeneous systems. *International Journal of Intelligent and Cooperative Information Systems*, 2(1):1–22.

Fang, D., Hammer, J., McLeod, D., and Si, A. (1991). Remote-exchange: An approach to controlled sharing among autonomous, heterogeneous database systems. In *Proceedings of the IEEE Spring Compcon*. IEEE.

Georgakopoulos, D., Hornick, M., and Sheth, A. (1995). An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure. *Distributed and Parallel Databases*, 3(2):119–153.

Gray, J. and Reuter, A. (1993). Transaction processing: Concepts and techniques.

IBM (1995). *FlowMark - Modeling Workflow, Version 2.1*. IBM Document No. SH19-8241-00.

ISO/IEC/JTC1/WG3 (1994). The Open-EDI Reference Model, Working Draft Document N255.

Kimberley, P. (1991). *Electronic Data Interchange*. McGraw-Hill, Inc., New York, New York.

Kimbrough, S., Lee, R., and Ness, D. (1984). Performative, Informative and Emotive Systems: The First Piece of the PIE. In *Proceedings of the Fifth International Conference on Information Systems*, pages 141–148.

O'Hare, W. L. (1990). An Introduction to Documentary Credits. Technical report, The Chartered Institute of Bankers, UK.

OMG (1995). *The Common Object Request Broker: Architecture and Specification, Revision 2.0.* Object Management Group and X Open, Document Number 91.12.1 Revision 1.1.

Orfali, R., Harkey, D., and Edwards, J. (1994). *Essential Client/Server Survival Guide.* John Wiley & Sons.

Reinwald, B. and Mohan, C. (1996). Structured Workflow Management with Lotus Notes Release 4. In *Proc. 4th IEEE Computer Society International Conference (CompCon), digest of papers.* available in http://www.almaden.ibm.com/cs/exotica.

Searle, J. (1969). Speech Acts: An Essay in the Philosophy of Language.

Shannon, C. E. and Weaver, W. (1949). *The mathematical Theory of Communication.* The University of Illinois Press, Urbana, Illinois.

Sheth, A. and Rusinkiewicz, M. (1993). On transactional workflows. *IEEE Data Engineering Bulletin*, 16(2).

Sheth, A. and Rusinkiewicz, M. (1995). Specification and execution of transactional workflows. In Kim, W., editor, *Modern Database Systems: The Object Model, Interoperability, and Beyond.* AMC Press.

Sheth, A. P. and Larson, J. A. (1990). Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases. *ACM Computing Surveys*, 22(3).

Sowa, J. F. (1984). *Conceptual Structures: Information Processing in Mind and Machine.* Addison-Wesley Publishing Co.

Sowa, J. F. (1993). Relating diagrams to logic. In Mineau, G., Moulin, B., and Sowa, J., editors, *Conceptual Graphs for Knowledge Representation, First International Conference on Conceptual Structures ICCS'93.*

Vinoski, S. (1993). Distributed Object Computing with CORBA. *C++ Report Magazine.*

WfMC (1995). *Workflow Reference Model Specification.* Workflow Management Coalition. http://www/aia.ed.ac.uk.wfmc.

Williams, G. (1994). OpenDoc Is Cross-Platform. http://www.cilabs.org /xplaform.html, Appeared in the November 1994 Edition of Apple Directions.

Wing, H. and Colomb, R. M. (1994). Towards Cooperative User Interfaces: An object-oriented application integration approach. In *Proceedings of IEEE GLOBCOM'94, San Francisco.*

Wing, H. and Colomb, R. M. (1997). An Architecture to Support Distributed Trade Documents. In *The Third International Symposium on Autonomous Decentralized Systems (ISADS97), Berlin, Germany.* IEEE Computer Society Press. To appear.

Wing, H., Liu, C., and Colomb, R. M. (1997). A Bottom-up Approach to Distributed Workflow. In *Pacific Asia Conference on Information Systems (PACIS97), Brisbane, Australia.* To appear.

Zhang, A., Nodine, M., Bhargava, B., and Bukhres, O. (1994). Ensuring Relaxed Atomicity for Flexible Transactions in Multidatabase Systems. In *Proc. 1994 SIGMOD International Conference on Management of Data.*

A Prototyping of a dossier

```
Type X Dossier
  Abstract
   Composite: Home-buying, Grad-school,
              paper-submission
   Act:       Buying-home, Applying-grad-school,
              submitting-paper
   Actor:     Buyer, Student, Author
   Partner:   Seller, Agent, B-Bank, Registrar, Organizer
   Associate: Financial-service, Employer, S-bank,
              Registrar, Recommender, Referee, Publisher
   Role:      Home-request, Negotiation, apply-HLA,
              pay-EDC, apply-GSA(X-Uni), apply-GSA(Y-Uni),
              submit(Proj-A), submit(Proj-B),
              submit(Proj-C)
```

```
\\-----------------------------------------------------
Act: Home-buying
Actor: buyer
Role: apply-HLA
Partner: b-bank
\\-----------------------------------------------------
episode 1 ::=
   EV: null
   IS: apply-HLA(begin)
   AC: send(req-HLA, b-bank)
   FS: req-HLA pending, waiting

etc...

episode 9 ::=
   EV: receive(rejection-notice, b-bank)
   IS: rejection-notice pending
   AC: null
   FS: apply-HLA(end)

\\-----------------------------------------------------
Act: Home-buying
Actor: buyer
Role: pay-EDC
Partner: seller, b-bank, agent
\\-----------------------------------------------------
episode 1 ::=
    EV: null
    IS: pay-EDC(begin)
    AC: send (sale-offer-accept, seller)
    FS: counter-offer-accept, waiting

episode 2 ::=
    EV: receive(counter-offer-accept, seller)
    IS: counter-offer-accept, waiting
    AC: send(LC-request, b-Bank)
    FS: LC request pending, waiting

episode 3 ::=
    EV: receive(LC, b-Bank)
    IS: LC request pending
    AC: null
    FS: obtain title pending, waiting

episode 4 ::=
    EV: receive(title, b-Bank) .
        Receive(Title, agent)
    IS: obtain-title pending
    AC: pay(money, b-Bank) ;
        send(pay-receipt, b-Bank)
    FS: title obtained; pay-EDC(end)

etc.
End Type
```

B Network activities associated with apply-HLA

```
1. send(req-HLA*, b-bank)
2. receive( ?HLA(loan-purpose,property-info,
      repayment-method, personal-particulars,
      financial-position), b-bank)
3. send(HLA*, b-bank)
4. receive(HLA-ack, b-bank)
5. receive(mortgage-contract, b-bank)
6. send(MC-accept, b-bank)
7. send(MC-reject, b-bank)
8. receive(approval-rpt, b-bank)
9. receive(rejection-notice, b-bank)
.
```

C CORBA objects associated with apply-HLA

1. Request standard object (request-stdobj)

```
Type REQUEST Atom
  Abstract
    Supertype: Standard Object
    Behaviour: [Passive|Active|Proactive]
    Interfaces:
      Standard-form ?Request-what (X: request)
      NAD ?Name-and-address-of-requester (X: request)
      Date ?Date-of-request (X: request)
      process submit (X: request)
      process receive (X: request)
    (Behaviour definitions)
End Type
```

2. Query standard object (query-stdobj)

```
Type HLA-QUERY Atom
  Abstract
    Supertype: Standard Object
    Behaviour: [Passive|Active|Proactive]
    Interfaces:
      NAD: Name-and-address (X: HLA-query)
      EMP: Employment-detail (X: HLA-query)
      money: Purchase-price (X: HLA-query)
      NAD: Address-of-property (X: HLA-query)
      money: Annual-income (X: HLA-query)
      money: Monthly-liabilities (X: HLA-query)
      an..9: Credit-account-number (X: HLA-query)
      money: Outstanding-balance (X: HLA-query)
      etc.
```

```
      process request (X: HLA-query)
      process respond (X: HLA-query)
      process receive (X: HLA-query)
    [Behaviour definitions]
 End Type

 Type GSA-QUERY Atom
  Abstract
    Supertype: Standard Object
    Behaviour: [Passive|Active|Proactive]
    Interfaces:
      NAD: Name-and-address (X: GSA-query)
      EMP: Employment-detail (X: GSA-query)
      String: Major-field (X: GSA-query)
      String: Minor-field (X: GSA-query)
      string: Department (X: GSA-query)
      float: Grade-point-average (X: GSA-query)
      String: Obtained-degree (X: GSA-query)
      NAD: Referee-Name-and-address (X: GSA-query)
      etc.

      process request (X: GSA-query)
      process respond (X: GSA-query)
      process receive (X: GSA-query)
    [Behaviour definitions]
 End Type

 Type CPA-QUERY Atom
  Abstract
    Supertype: Standard Object
    Behaviour: [Passive|Active|Proactive]
    Interfaces:
      NAD: Name-and-address (X: CPA-query)
      NAD: Conference-address (X: CPA-query)
      an..4: Paper-number (X: CPA-query)
      NAD: referee-mailing-address (X: CPA-query)
      money: Registration-fee (X: CPA-query)
      Date: camera-ready-paper-due-by (X: CPA-query)
      etc.

      process request (X: CPA-query)
      process respond (X: CPA-query)
      process receive (X: CPA-query)
    [Behaviour definitions]
 End Type
```

3. Acknowledgement standard object (ack-stdobj)

```
Type ACK-NOTE Atom
  Abstract
    Supertype: Standard Object
    Behaviour: [Passive|Active|Proactive]
    Interfaces:
      Form: Received-application (X: ack-note)
      Date: Date-received (X: ack-note)
      NAD: Received-by (X: ack-note)

      process send (X: ack-note)
      process receive (X: ack-note)
    [Behaviour definitions]
End Type
```

4. Contract standard object (contract-stdobj)

```
Type CONTRACT Atom
  Abstract
    Supertype: Standard Object
    Behaviour: [Passive|Active|Proactive]
    Interfaces:
      money: loan-amount (X: contract)
      percent: annual-interest (X: contract)
      money: payment-amount (X: contract)
      boolean: monthly-payment (X: contract),
             fortnightly-payment (X: contract)
      Date: payment-start-on (X: contract)
      Date: last-payment-on (X: contract)

      process send (X: contract)
      process receive (X: contract)

    [Behaviour definitions]
End Type
```

etc.

D Standard repositories associated with ACTs and ROLEs

```
apply-HLA ::= name and address (NAD),
        employment detail (EMP),
        purchase price (PUR),
        address of property (NAD),
        annual income (INC),
        monthly liabilities (LIA),
        credit account number (ACT),
        outstanding balance (OBL),
        etc.

apply-GSA ::= name and address,
        employment detail (EMP),
        major field (MAF),
        minor field (MIF),
        department (DEP),
        grade point average (GPA),
        obtained degree (DEG),
        referee mailing address (NAD)
```

```
        etc.

make-publication ::= name and address,
        organizer address (NAD),
        paper reference number (RFN),
        referee mailing address (NAD),
        registration number (RGN),
        registration fee (RGF),
        camera-ready version due by (CDB),
        etc.
```

E A sample UN/EDIFACT message - CREADV (credit advice)

```
-------------------------------------------------------
TAG NAME                                  S  REPT

UNH Message header                        M   1
BGM Beginning of message                  M   1
DTM Date/time/period                      M   7
RFF Reference                             C   3
BUS Business function                     C   1
FTX Free text                             C   1

 Segment Group 1  M   4
MOA Monetary amount                       M   1
CUX Currencies                            C   1
DTM Date/time/period                      C   2
RFF Reference                             C   1

 Segment Group 2 M   4
FII Financial institution information M       1
CTA Contact information                   C   1
COM Communication contact                 C   5

 etc.

 Segment Group 10 C   1
AUT Authentication result                 M   1
DTM Date/time/period                      C   1

UNT Message trailer                       M   1
```

F Samples of UN/EDIFACT segments

```
NAD (Name and Address)
Function: To specifying the name/address and their related
function, either by C082 only and/or unstructured by C058
or structured by C080 thru 3207.

2025 PARTY QUALIFIER                        M an..3
C082 PARTY IDENTIFICATION DETAILS           C
C058 NAME AND ADDRESS                       C
C080 PARTY NAME                             C
C059 STREET                                 C
3164 CITY NAME                              C an..35
3229 COUNTRY SUB-ENTITY IDENTIFICATION      C an..9
3251 POSTCODE IDENTIFICATION                C an..9
3207 COUNTRY, CODED                         C an..3
-------------------------------------------------------
EMP (Employment details)
Function: To specify employment details.

9003 EMPLOYMENT QUALIFIER                   M an..3
C948 EMPLOYMENT CATEGORY                    C
C951 OCCUPATION                             C
C950 QUALIFICATION CLASSIFICATION           C
3494 JOB TITLE                              C an..35
```

G UN/EDIFACT Data Element Format Notation

```
Datatype: a (alphabetic), n (numeric), an (alphanumeric),
          id (alphabetic numeric) or (alphanumeric
          identifier).

a3       3 alphabetic characters, fixed length
n6       numeric characters (numbers), fixed length
an5      5 alphanumeric characters, fixed length
a..6     up to 6 alphabetic characters
an..35   up to 35 alphanumeric characters
n..9     up to 9 numeric characters (number)
```