

Association for Information Systems AIS Electronic Library (AISeL)

PACIS 2001 Proceedings

Pacific Asia Conference on Information Systems
(PACIS)

December 2001

Component-based Approach to the Development of Internet EDI Software for Electronic Document Management

Jian Ma

City University of Hong Kong

Quan Zhang

City University of Hong Kong

Follow this and additional works at: <http://aisel.aisnet.org/pacis2001>

Recommended Citation

Ma, Jian and Zhang, Quan, "Component-based Approach to the Development of Internet EDI Software for Electronic Document Management" (2001). *PACIS 2001 Proceedings*. 50.

<http://aisel.aisnet.org/pacis2001/50>

This material is brought to you by the Pacific Asia Conference on Information Systems (PACIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in PACIS 2001 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Component-based Approach to the Development of Internet EDI Software for Electronic Document Management

Jian Ma, Jiazhi Liang and Quan Zhang
Department of Information Systems, City University of Hong Kong
Kowloon Tong, Hong Kong

Abstract

Internet EDI plays an important role in electronic business applications. One of its major functions is the translation and extraction of electronic documents. This paper presents a component-based approach to the development and application of the Internet EDI software for exchange and management of electronic documents. The software has been used in the Hong Kong SAR Government's Innovation and Technology Commission and the National Natural Science Foundation of China for electronic submission of project proposals, progress/final reports and on-line evaluation of project proposals. The proposed approach and the software greatly simplify the processes for R&D project management and create a paper less work environment.

1. Introduction

Electronic data interchange (EDI) is an electronic mean for computers to exchange and process business documents. In the past twenty years, it has been widely used by organizations to process their business information efficiently and accurately. An EDI system usually consists of four major components, i.e. document standard, networking equipment, EDI software, and network transmission service [1, 2]. Due to the rapid development of the Internet technologies, the network transmission services have changed greatly. Current research indicate that using Internet instead of traditional Value Added Networks (VANs) greatly reduces the network service cost and hence creates opportunities for small organizations to adopt EDI solutions [3, 4]. Internet EDI is then a solution for business partners to exchange and manage electronic documents over the Internet. It is a supplementary approach for large organizations, which have developed traditional EDI systems, to trade with business partners through the Internet. It has the advantages of low cost service charge, large user groups, alternative services and user interactions. A detailed comparison of traditional EDI and Internet EDI is shown in Table I.

Internet EDI software consists of the translation program, extraction program and communication program, as shown in Figure 1. Translation program is designed to format the data into EDI messages according to the pre-defined electronic document standards. Extraction program extracts and decodes the data from incoming EDI messages based on the document standards. Communication program provides the facility to exchange EDI messages between source organization and destination organizations. TCP/IP is often used as the basic network transmission protocols for the Internet EDI software applications.

Table I: Differences between Traditional EDI and Internet EDI (Modified from [4])

Layer	Traditional EDI	Internet EDI
Message standardization layer	EDI formatted documents (EDIFACT or ANSI X.12)	EDI formatted documents (Proprietary flat files HTML forms)
Enveloping layer	X.400 standard Proprietary enveloping methods	Simple Mail Transport Protocol (SMTP) Multipurpose Internet Mail Extension (MIME) File Transfer Protocol (FTP) Hypertext Transfer Protocol (HTTP)
Transport layer	X.25 packet switching network Proprietary protocols	Transmission Control Protocol (TCP)/Internet Protocol (IP)
Physical layer	Direct connections Dial-up lines Private networks	Internet Dial-up lines

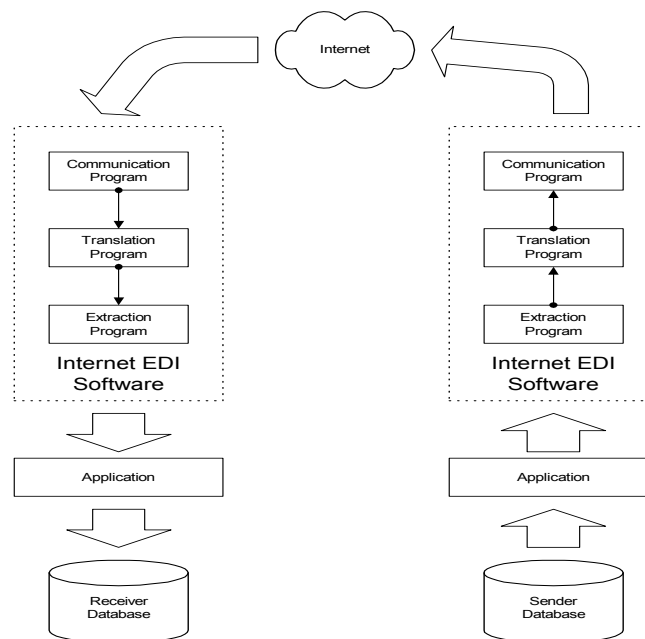


Figure 1: Architecture of Internet EDI Software

Several approaches [2, 3, 5] have been proposed to implement Internet EDI software for exchange of electronic documents over the Internet. These approaches can be classified depending on 1) whether an EDI software is used; 2) whether third party Internet service is used; and 3) whether the use of EDI software is compulsory for both the sender and receiver. The decision to choose proper approaches for Internet EDI implementation should take into consideration the degree of system integration (e.g. communications between business applications) and scale of the Internet adoption. The CML model [3] presents the development and application of Internet EDI software using the pre-installed client software

plus web browser. The client software displays the data on the web browser to capture the user data entry, translate the data into EDI messages and then transmit the data across Internet to hub server. Fu et al [5] develop the similar Internet EDI software using Web browser plus Java applet as the client translation program. The Internet EDI software uses hub server to capture and store data. Thus, the client is free of the pre-installed software support under the mobile code execution mechanism (e. g., Java Virtual Machine), where Java applet can be downloaded and executed in any Java-enabled browser to perform the translation function.

Design of the Internet EDI software starts with different layers of the Internet protocols/services. For example, an electronic form can be implemented in HTML to display information in a web browser. Data processing is supported by both web browser and web server communication based on Hypertext Transport Protocol (HTTP). Table II presents several available Internet applications and their corresponding protocols/services on which the applications are built.

Table II: Internet Applications and Their Correspondent Protocols (Modified from [6])

Application	Protocol	Purpose
E-mail	Simple Mail Transport Protocol (SMTP) Post Office Protocol version 3 (POP3) Multipurpose Internet Mail Extensions (MIME)	Allows the transmission of text messages and binary attachments across the Internet.
File Transfer	File Transfer Protocol (FTP)	Enables files to be uploaded and downloaded across the Internet.
Chat	Internet Relay Chat Protocol (IRC)	Provides a way for users to talk to one another in real time over the Internet. The real-time chat groups are called channels.
UseNet Newsgroups	Network News Transfer Protocol (NNTP)	Discussion forums where users can asynchronously post messages and read messages posted by others.
World Wide Web	Hyper Text Transport Protocol (HTTP)	Offers access to hypertext documents, executable programs, and other Internet resources.

The decision to choose Internet protocols/services for implementation of Internet EDI software leads to different complexities for software development. The web-based implementation is always based on a three-tier architecture [7, 8], where the browser is the user interface responsible for the HTML interpretation and web content presentation. Java applets or other embedded software components can also be displayed and executed via browser in the second tier. The Web server resides on the final tier and is responsible for data processing. In a three-tier architecture, browser is a presumed client resource providing programming space for coding and scripting. Regardless of client machine settings, Java applets are designed with the platform-independent mobile-code system. Thus, it is compatible with different client settings (i.e. software and hardware). At front-end, Web browser potentially has the ability

to capture user data submitted from HTML form, and it is used as extraction program to capture user data. Communication software is handled by transmission mechanism built for World Wide Web.

Component-based software technologies [7, 9, 10] have been used to develop the Internet EDI software. They provide standards and facilities to support software design by selecting and integrating pre-built software components with high quality. These technologies have led to several industry standards such as Microsoft ActiveX/OCX, COM/DCOM, OMG CORBA 2 and Sun's Java Beans. Different from the traditional software engineering methodologies, the component-based technologies emphasize the application of component selection, integration, and production at different stages of software life cycle (e.g. modeling, design and implementation). Thus the outcome from a software project is a set of components that represent certain function modules. The components have the advantages of code reuse and ease of distribution. When Internet EDI software is constructed by assembling software components, it is necessary to guarantee that each component referenced by the software is successfully installed in the distribution site.

2. Design and implementation of the Internet EDI software

As shown in Figure 1, the Internet EDI software consists of the extraction program, translation program and communication program. Extraction program runs on the server site to extract data from electronic documents in different standards. Various open infrastructures have been applied to the construction of the extraction programs [11, 12, 13, 14]. For HTML documents, web server is used to extract the data from the form POST/GET data packet. For other electronic document standards (e.g., Microsoft Office, PDF and XML), the extraction program is developed based on different industrial component standards. For example, Microsoft Office supports the Distributed Component Object Model (DCOM) and PDF reader supports the Document Object Model (DOM). XML is supported by the parser component implemented in different component standards such as the Common Object Request Broker (CORBA) standard.

Translation program is supported by various client applications and platforms. For HTML document standard, the translation program is the web browser. For Microsoft Office document and Acrobat PDF file, the translation program is the third party application software launched as the standard-alone installation package (Microsoft Office) or free plug-in (PDF reader). For XML, the translation program is implemented in mobile code (Java applet) system that runs on the web browser with enhanced features, e.g., Java Virtual Machine (JVM). Current trend of research and development work is to develop the new versions of web browser to support the XML document standards.

HTML, MS Word, Acrobat PDF and XML are the most commonly used electronic document formats in the practical electronic business applications. Thus, this paper presents the system components and functions for implementation of the Internet EDI software to support the exchange of these electronic document formats.

HTML format

Table III presents the layered architecture of the system components and its implementation using the HTML document standard.

Table III: System Components for HTML Format

Internet EDI Components	Function Description	Correspondent Component
Document Standard	Pre-defined data structure for electronic messages	Standard HTML Form POST/GET Data Packet

Translation Software	Software tools for translating business data into EDI Message in defined structure	Web Browser
Extraction Software	Software tool for extracting business data from EDI Message in defined structure	Web Server Service
Communication Software	Provide the transmission support between two Communication sites	Hypertext Transport Mechanism

A three-tier software architecture is used to implement the Internet EDI software to support HTML document standard. The first tier of the software handles the form message submitted from the client side. It includes a browser as the user interface for HTML interpretation and web content presentation. The second tier is the web service for dynamically generating the HTML forms and directing other form templates. The third tier is the data-centric service, which includes gateway functions for data transaction processing. In this three-tier architecture, web browser is a presumed client application to download and submit the application forms.

Microsoft Word format

Table IV presents the layered architecture of the system components and its implementation using Microsoft Office document standard.

Table IV: System Components for Word Format

Component of EDI Software	Function Description	Correspondent Component
Document Standard	Pre-defined data structure for electronic messages	Microsoft Word supported document standard
Translation Software	Software tools for translating business data into EDI message in defined structure	Microsoft Word Application
Extraction Software	Software tool for extracting business data from EDI message in defined structure	Microsoft Word components with server transaction container service (Microsoft Transaction Server, for example)
Communication Software	Provide the transmission support between two Communication sites	Web browser Hypertext Transport mechanism

A new type library is used to implement the extraction program. It integrates the database manipulation components with Word components to provide both database operation and document extraction function. ActiveX standard is used as the basic protocol to communicate with Word component. Microsoft Transaction server (MTS) is also used to support the enhanced components with distributed database transaction support to integrate both the extraction and database operation processes.

Adobe Acrobat PDF format

Acrobat PDF file format is one of the most widely used document standards for exchange of business documents in electronic business applications [9]. It is portable and is independent from computer hardware and operating systems. The PDF file format with form functions is selected as the document standard in our proposed design.

Table V presents the layered architecture of the system components and its implementation using Adobe PDF document standard.

Table V: System Components for Acrobat PDF Format

Component of EDI Software	Function Description	Correspondent Component
Document Standard	Pre-defined data structure for electronic messages	Standard HTML Post/Get form data packet
Translation Software	Software tools for translating business data into	Acrobat Reader

	EDI Message in defined structure	
Extraction Software	Software tool for extracting business data from EDI Message in defined structure	Web Server service
Communication Software	Provide the transmission support between two communication sites	Acrobat Reader Hypertext Transport mechanism

At the front-end, a PDF file can be presented as a form to capture user inputs. Web browser with enhanced Acrobat plug-in is capable to download and read PDF form file. The reader software has been implemented to support the same Document Object Model (DOM) as the standard browser. When reader application interacts with the back-end server, it performs as a web browser to interact with the web server. Therefore, standard HTML form POST/GET data packet can be generated from Acrobat reader to submit the filled form data to back-end web server. In our design, Web server service has been installed and configured to receive Acrobat Reader data. Besides, off-line form filling is also supported. Additional security control such as digital signature is also embedded in PDF form design.

XML format

XML is standardized by World Wide Web Consortium (W3C) group and is targeted to provide a method for putting structured data in a text file [1]. With tokens and frameworks of syntax defined, XML is capable to represent entity data with complex hierarchical element structures. Along with the global standardization of XML, Document Type Definition (DTD) embeds transactions and related rules that can be transferred and processed interactively. Compared to the traditional EDI solution, XML provides the facility to support the EDI message translation by using standard object definition for XML document elements. XML is now well supported by software community. More and more software components are generated to support the standardized XML text element translation and XML presentation.

Table VI presents the layered architecture of the system components and its implementation using XML as the document standard.

Table VI: System Components for XML Format

Component of EDI Software	Function Description	Correspondent Component
Document Standard	Pre-defined data structure for electronic messages	XML
Translation Software	Software tools for translating business data into EDI message in defined structure	Enhanced browser function
Extraction Software	Software tool for extracting business data from EDI message in defined structure	Standard XML browser
Communication Software	Provide the transmission support between two Communication sites	Hypertext Transport mechanisms

The XML file is maintained using Java technology. Certain Java applets are designed to leverage the user interaction so as to support XML file generation. Normally, users can use web browser with enhanced Java VM support to execute the Java applet. When applet starts, users can fill in the data and send the submit instruction to applet. The applet then wraps the user data into XML formatted text and sends the data to the back-end server.

The server implementation includes two components of the software package for XML extraction and database access. In the standard package for XML extraction, components help to rebuild the document object model by translating the serialized XML text. The document object is accessible via document

object model. The database access component provides the basic support for database communication, including connection creation, transaction coordination and query execution.

3. Application of the Internet EDI software

The developed Internet EDI software has been used as extended functions for exchange of electronic documents in the Internet-based Technology Information System (ITIS: <http://itf.gov.hk>) and Internet-based Science Information System (ISIS: <http://isis.nsf.gov.cn>). The extended ITIS has been used for the electronic submission of project proposals in the Hong Kong SAR Government's Innovation and Technology Commission. Where the project coordinators/applicants can submit their electronic proposals in any of the two approaches as shown in Figure 2: 1) upload a completed MS WORD document in the predefined format; and 2) fill in the partial project form and then upload the completed electronic document (in the formats of MS Word, RTF, PDF, etc). The extended ISIS has been used for on-line evaluation of project proposals in the National Natural Science Foundation of China. Where the external reviewers can choose to submit their review reports in 1) web forms; 2) MS Word document in the predefined format. The available function is shown in Figure 3.

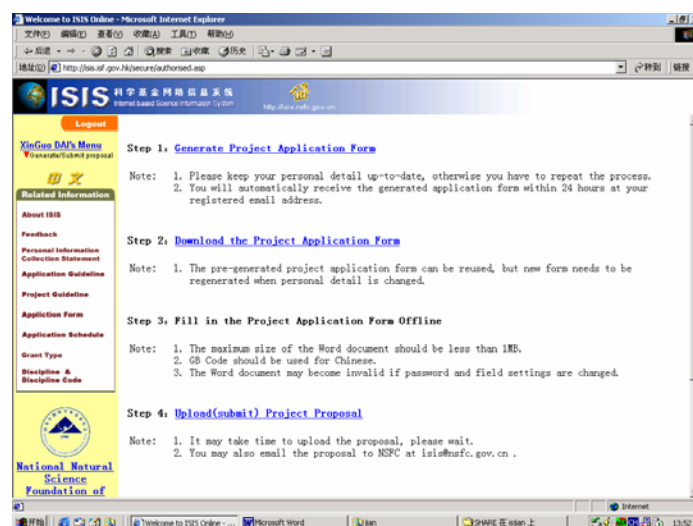


Figure 2: Interface of Approaches for Electronic Submission of Project Proposals

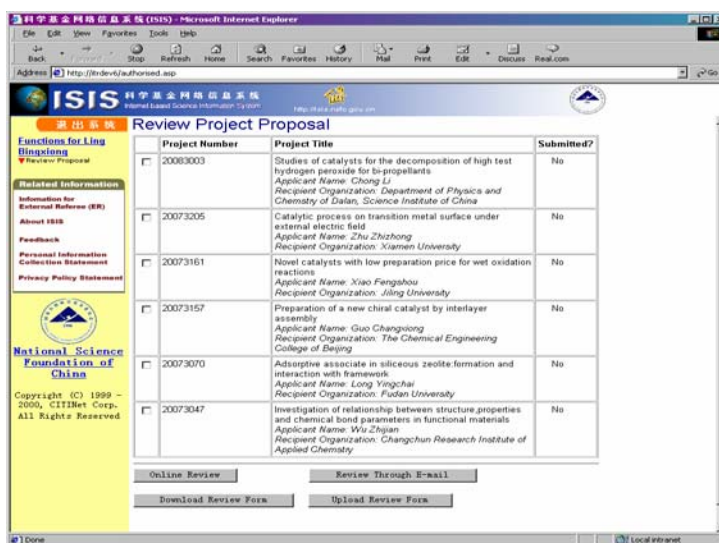


Figure 3: Reviewer's Interface for On-line Evaluation of Proposals

The users of the Internet EDI software can have several choices to submit their electronic documents over the Internet, as is shown in Figure 4. At backend, a special server is used to store the electronic documents received. A program then runs at backend server in batch mode. It translates the electronic documents into the database on daily bases. The interface of the extraction and translation programs is also shown in Figure 5.

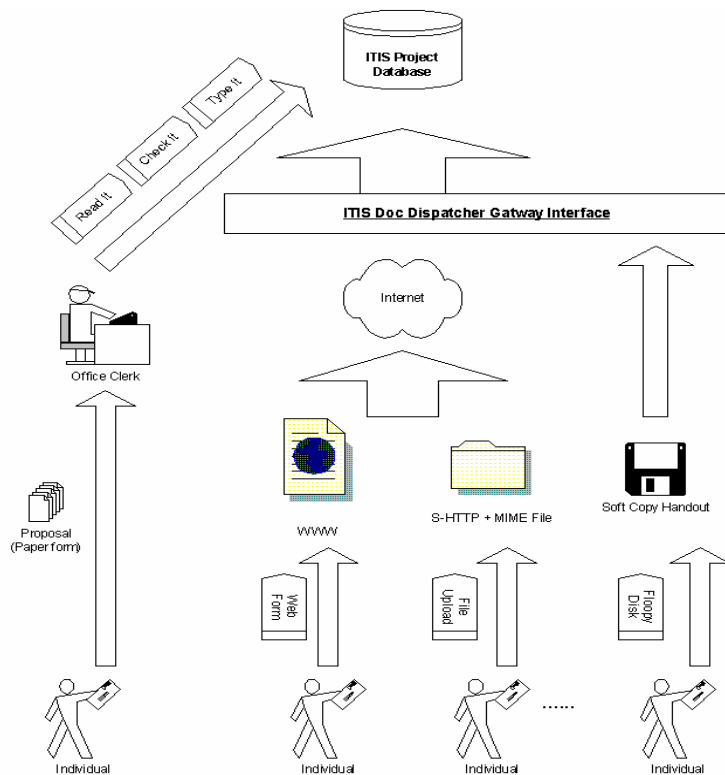


Figure 4: Architecture and approaches for submitting electronic documents

The screenshot shows the Data Abstractor software interface. The main window displays project details for '103'. The 'Project Title' section includes English and Chinese descriptions. The 'Classification of the Project' section shows the nature as 'Process/Product/Technology Development or Improvement - Product Development/Improvement' and the sector as 'Information Technology'. The 'Project Summary' section provides a detailed description of the project's goals and impact. The 'Manual-add Information' section includes fields for 'Assign Project Number', 'Assign Programme', 'ITSP', and 'Assign to Committee'. The interface also shows a list of project entries on the left and a status bar at the bottom.

Figure 5: Interface for extraction and translation at backend server

Simple Mail Transfer Protocol (SMTP), Secure Multi Purpose Internet Mail Extension (S/MIME) and Secure HyperText Transfer Protocol (S/HTTP) have been used as the three major protocols for the enveloping layer of the communication program. Security infrastructure has been designed for the three protocols. Implemented on the Secure Socket Layer (SSL), data transfer via the selected protocols is under the control of public key infrastructure (PKI). In practical applications, the software requires users to provide their certified key when doing the transfer. The owners of the system (e.g. NSFC is the owner of ISIS) will release the keys during the user registration and authentication processes.

The application results show that the Internet EDI software can greatly reduce the workload and simplify the working processes for submission of electronic proposals and on-line evaluation of proposals in government funding agencies. It also minimizes the human errors.

3. Summary

This paper reviews methods and techniques for design and implementation of the Internet EDI software. It also presents a component-based approach to develop the Internet EDI software for exchange of electronic documents in different formats. The software consists of the extraction program, translation program and translation program. It has been used in the Internet-based Technology Information System (ITIS) for electronic submission of project proposals in the Hong Kong SAR Government's Innovation and Technology Commission, and in the Internet-based Science Information System (ISIS) for on-line evaluation of project proposals in the National Natural Science Foundation of China. The proposed software is easy to use, compatible with many existing document standards, and integrated with Secure Protocol (S-HTTP) and client application layer protection.

Reference:

- [1] B. K. Blacker, "The development of an EDI system", *Computing & Control Engineering Journal*, Vol. 2 (5), pp. 231–237, 1991.
- [2] H. C. Mak and R. B. Johnson, "Tools for implementing EDI over the Internet", *Journal of Electronic Commerce*, Vol. 11 (1), pp. 44–53, 1998.
- [3] M. S. Threlkel and C. B. Kavan, "From traditional EDI to Internet-based EDI: managerial considerations", *Journal of Information Technology*, Vol. 14, pp. 347–360, 1999.
- [4] R. B. Johnston, *Trading Systems and Electronic Commerce*, Eruditions Publishing, 1998.
- [5] S. Fu, J. Y. Chung, W. Dietrich, V. Gottemukkala, M. Cohen and S. Chen, "A practical approach to Web-based Internet EDI", *Proceedings of the 19th IEEE International Conference on Distributed Computing Systems*, pp. 53–58, 1999.

- [6] E. Turban, J. Lee, D. King and H. M. Chung, *Electronic Commerce, A Managerial Perspective*, Prentice Hall, 1999.
- [7] S. M. Lewandowski, "Frameworks for component-based client/server computing", *ACM Computer Survey*, Vol. 30 (1), pp. 3–27, 1998.
- [8] G. R. Voth, C. Kindel, J. Fujioka, "Distributed application development for three-tier architectures: Microsoft on Windows DNA", *IEEE Internet Computing*, Vol. 2, pp. 41–45, 1998.
- [9] D. F. D'souza and A. C. Wills, *Objects, Components, and Frameworks with UML, The Catalysis Approach*, Addison-Wesley, 1999.
- [10] C. Szyperski, *Component Software, Beyond Object-Oriented Programming*, Addison-Wesley.
- [11] G. L. Rein, D. L. McCue and J. A. Slein, "A case for document management functions on the Web", *Communications of the ACM*, Vol. 40 (9), pp. 81–89, 1997.
- [12] M. Harmison, "Creating electronic documents that interact with diagnostic software for on-site service", *IEEE Transactions on Professional Communication*, Vol. 40 (2), pp. 92–101, 1997.
- [13] G. Heba, "Digital architectures: a rhetoric of electronic document structures", *IEEE Transactions on Professional Communication*, Vol. 40 (4), pp. 275–283, 1997
- [14] C. Madigan, M. K. Silber and S. Wilson, "Lessons learned prototyping an SGML-based computerized document management system", *IEEE Transactions on Professional Communication*, Vol. 40 (2), pp. 139–143, 1997.