

Design and Implementation of a Multimedia Document Automation System

Chin-Hwa Kuo, Timothy K. Shih, Hong-Sheng Chui, and Li-Chun Sung

Multimedia Information NETWORKing (MINE) Laboratory

Department of Computer Science and Information Engineering

Tamkang University, Tamsui, Taiwan China

e-mail: chkuo@mail.tku.edu.tw

Abstract — An electronic document automation system serves the purpose of processing the documents over networks in a paperless manner. Obviously, such a system is significantly beneficial to our working efficiency. In the present paper, we design and implement a multimedia document automation system. We make use of HyperText Markup Language (HTML) to construct the multimedia document formats. We also design a web-based document management system and a workflow control scheme to facilitate query and management processes. We implemented the proposed system on Windows NT platform in the Multimedia Information NETWORKing (MINE) Laboratory at the Tamkang University.

Keywords: World Wide Web (WWW), Hyper Text Markup Language (HTML), multimedia documents, document automation

I. INTRODUCTION

An electronic document automation system serves the purpose of processing the documents over networks in a paperless manner. Obviously, such a system can be significantly beneficial to working efficiency. Designing of such a system involves many complex and challenging technical, social, as well as cultural issues. Many software vendors have developed products for completing this market, for example, Lotus Notes[1], Microsoft Exchange[2], Novell Groupwise[3].

In the present paper, we design and implement a multimedia document automation system. Such a system requires to be able to integrate multiple media within the document format and provide co-sign tools. In other word, it is not only an electronic document automation system with traditional text format, but also with multimedia presentation and co-sign capability. So that information content inside a document is significantly enhanced.

Designing of such a system involves many complex and challenge issues. In our work, we focus on the following tasks: (1) Hyper Text Markup Language (HTML) based multimedia document formats and editing tools; (2) document workflow control; (3) web-based document management system; (4) document access and transmission security [4].

II. MULTIMEDIA DOCUMENT AUTOMATION SYSTEM OVERVIEW

In traditional document processing procedure, after editing of a document, this document is delivered to each corresponding party to co-sign, judge, return, and then file it as long as the required process is accomplished. In the above process, however if the deliver process is performed manually and if the document is delayed at any point, it becomes very time consuming to find out the current status of a particular document. Thus, from the automation viewpoint, a document automation system which makes use of the advances in the computer and network technologies can be of benefits to the processing efficiency.

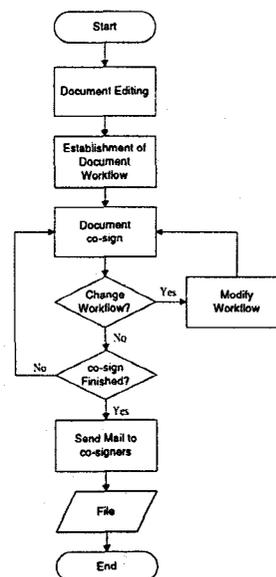


Fig. 1 Document co-sign flow chart

In the present paper, we design and implement a multimedia document automation system. Fig. 1 depicts the document processing procedure. From document editing to file, all processes can be performed by making use of the advances of multimedia PCs and networks. The whole system network infrastructure is shown in Fig. 2. It is a

centralized network topology. Each corresponding user makes use of a multimedia PC to get connection to the multimedia document server. When a document is created by a user, then it is transmitted to the document server. In the mean while, a workflow chart is also generated. Based on the workflow chart, all the corresponding parties will be informed *sequentially* until the whole process is completed. Although the process is straightforward, the design of such a system involves many challenging issues. Our focal points are as follows:

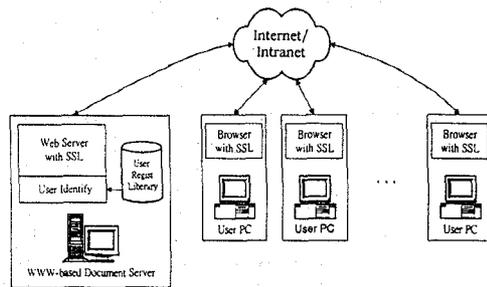


Fig.2 Network topology

A. HTML-based multimedia document formats and editing

There are multimedia document formats such as HTML, MHEG, ODA, and HyTime [5]. Due to the popularity of the World Wide Web (WWW), designing of multimedia document formats based on HTML has many advantages. For instances, an authority user can view the document from any web browsers without using proprietary interface. Furthermore, the web page designing skill becomes common, it does not require an expert to perform such a task. Moreover, the HTML can easily incorporate multimedia in the document format. Obviously, with these advantages, the software cost and user training cost can be reduced.

B. Document workflow control scheme

A workflow is established as a document created. The workflow is used to manage the required document processing time. If a document stays longer than the predetermined time at any particular point, some actions may require. In a manual approach, it may be very time consuming to find out the delay point. In electronic approach, by using information obtained from workflow control chart, the above problem can easily be resolved. For example, a warning signal is generated to inform the operator of document management system. To ensure the workflow can be processed in a reliable and efficient way, we design a notification mechanism inside the management system as well as the client ends. If the network or the document process has any abnormal condition, the document operation manager is informed. Then a suitable action, for example an off-line action, can be taken to eliminate the problem.

C. Web-based multimedia document management system

A document management system serves the purposes of facilitating the query and statistical report processes. Our document management system integrates database and web technologies. It maintains the original documents and the corresponding comments. One can query each document based on keyword searching or document sequence number to find out the status of any particular document. The management system also provides statistical reports and figures to evaluate the document processing efficiency of each corresponding party. Thus, the performance of all involved parties can be evaluated.

D. Document access and transmission security

Although the document is resident in the server most of the time, its copy need transmit to the corresponding party through the network. As shown in fig. 2, before sending to network, each document is encapsulated by using the SSL technique [6] [7]. The access security is handled by the operating system.

III. DESIGNING OF MULTIMEDIA DOCUMENT AUTOMATION SYSTEM

A. HTML-based multimedia document formats and editing

The advantages of using HTML-based documents are mentioned before. So that media such as text, images, animation, video, and sound can be integrated in a document to enhance the information content. The original document could be created by editing an HTML file or by using the designed input form. In the following, we make use of an example to illustrate the editing process. It consists of the following steps:

1) Fill in basic information:

In this step, we design a web page, as shown in fig. 3, for an initiator to fill in document basic information. It may contain data such as document serial number, date, subject etc. Note that this example is provided for illustrating purpose. The contents in fig. 3 may be different, but, function is the same.

2) Fill in subject and contents:

A web page is designed to facilitate the document editing process. We key in the subject and contents, attach graphics, images, and sound, as illustrated in fig. 4.

3) Create workflow:

Fig. 5 depicts a workflow web page. This page is a *dynamic* web page. The co-sign order as shown is updated when a new party is added or removed. Once a workflow is ready, user presses the execute bottom and the system enters to the co-sign process.

4) Display document:

After editing process, we generate a document as shown in fig. 6. The document subject, contents, co-sign parties, and document statuses are shown in one web page. From this page, we proceed the co-sign process.

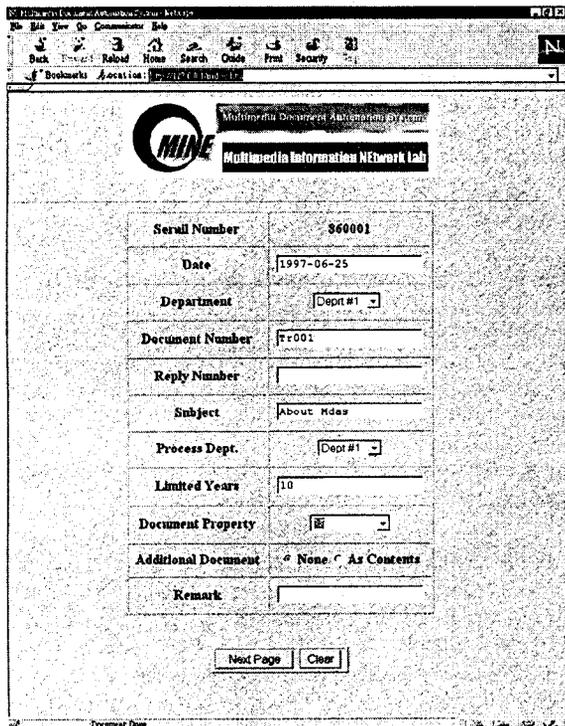


Fig.3 Document basic information web page

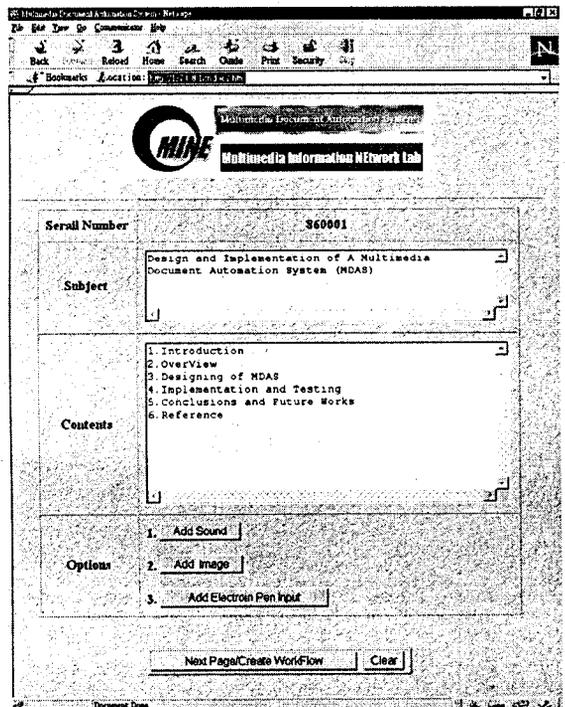


Fig. 4 Document subject and contents

To co-sign a document, we provide users three different ways, i.e., *key-in*, *electronic pen*, and *voice*. The

current version of HTML (HTML 3.0) does not provide input mechanism to deal with the last two items. It requires special treatment. We apply Active X technique to design a mechanism which transmit a prepared file from the user end to the server end [8]. And the electronic pen signature is handled as graphic file and treated in a similar manner.

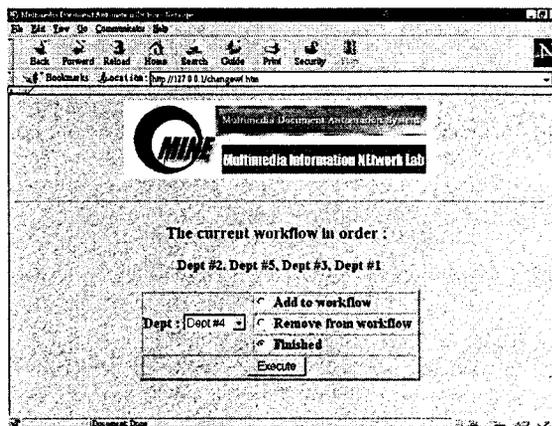


Fig. 5 Workflow creation web page

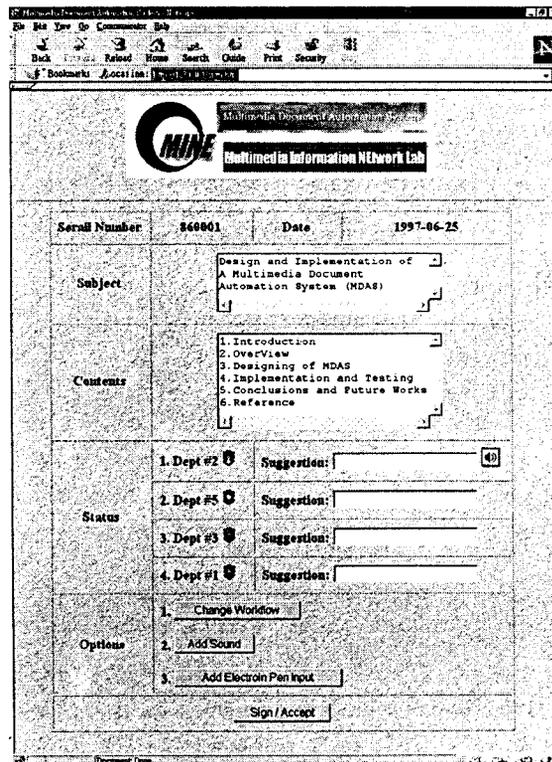


Fig. 6 Document display and co-sign

B. Document processing management

To manage a document processing efficiency, we require two mechanisms: (1) workflow control mechanism [9] [10]; and (2) inform mechanism.

1) Workflow control mechanism

As illustrated before, once a document is created, the workflow is also generated. The document is delivered to each corresponding party based on this workflow sequentially. However, this workflow may be inappropriate and need to be changed. As shown in fig. 7, we illustrate different workflow cases. Fig. 7(a) is a normal case, the document goes through the path as original design. Fig. 7(b) is a reverse case, the document may return to its previous party. Fig. 7(c) depicts a general case, a new path may create during the document processing. To handle these cases, as shown in fig. 5, we design a dynamic web page which dynamically reflects its workflow to the corresponding users.



Fig. 7(a) Workflow - normal case



Fig. 7(b) Workflow - reverse case

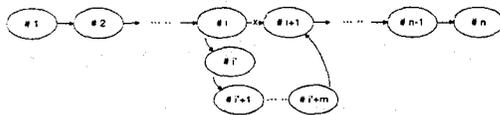


Fig. 7(c) Workflow - general case

Fig. 7 Workflow cases

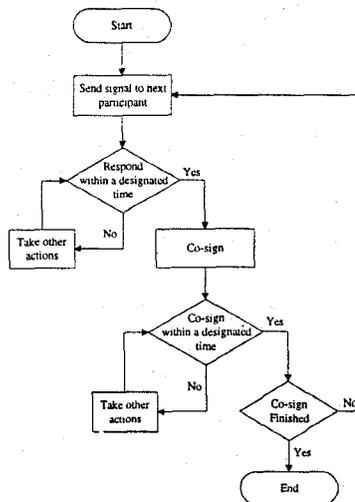


Fig. 8 Co-sign workflow

2) Inform mechanism

The inform mechanism play a crucial role in the management of workflow. Due to the urgency of documents is different, different urgency levels are assigned. There are two basic levels, i.e., regular case and urgent case. To deal

with different cases, different inform methods are required. In the regular case, the inform mechanism sends an e-mail to the user's end for illustrating a document is on queue. In the mean time, an item appended in the corresponding user's document web page so that users will be informed. This is a passive notification way. Users know a document on queue by checking their e-mail, or by checking their corresponding document web pages. In the urgent case, we design a mechanism which uses the ping operation to find out the participant's MPC current status. As shown in fig. 8, there are two loops in this figure. In the first loop, the participant machine is on-line. Then, a signal is sent to inform the participant. If the participant responds within a designated time, the system transits from inform state into co-sign state. Then the process proceeds as shown. This is an active notification way. If the participant may neglect the urgent document and result in long processing duration. Other inform operations may require.

C. Document file management system

To handle the storage of documents and workflow chart, we prefer to have a database. The document files are stored in the file server in HTML format. We design a data structure, as shown in fig. 9, to record a document's workflow. In this example, we have five co-sign parties. The co-sign order and names are given corresponding in column one and two. Column three denotes the co-sign status of each party. Column four and column five indicate a document processing time of the corresponding party. By using this table, we can generate statistical reports and extract some useful information to evaluate the work efficiency of a department.

co-sign Order	co-sign Name	co-sign Status	Time In	Time Out	Limit Time
1	Dept#1	Finished	97/6/5 10:01	97/6/5 11:35	4hr
2	Dept#2	Processing	97/6/5 11:35	Null	4hr
3	Dept#3	Created	Null	Null	4hr
4	Dept#4	Created	Null	Null	4hr
5	Dept#5	Created	Null	Null	4hr

Fig. 9 Workflow table

IV. IMPLEMENTATION AND TESTING

We implement the system in the Multimedia Information Network Laboratory (MINE) at Tamkang University. In hardware portion, we choose a machine with two Pentium-Pro CPUs as server and some multimedia PCs as client ends. In software portion, we select Windows NT 4.0 as platform [11], Microsoft Internet Information Server (IIS) [12] as web server, and Microsoft SQL as database server [13]. Fig. 10 depicts the system software architecture. Currently, we are testing the system functionality with the cooperation of Tamkang University document management department.

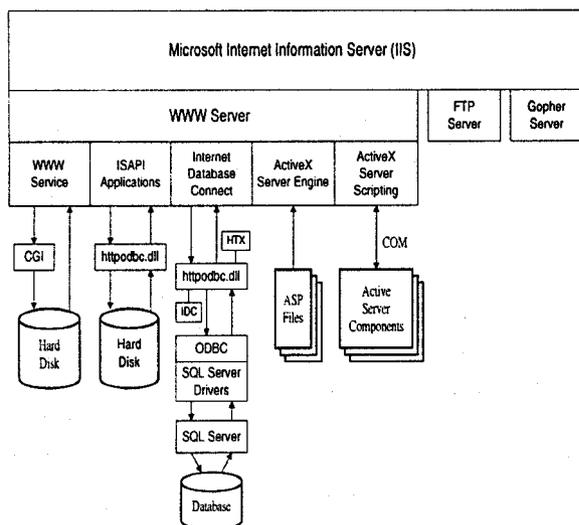


Fig. 10 System software architecture

V. CONCLUSIONS

We design and implement a web-based multimedia document automation system. Due to the popularity of the world wide web, the HTML is selected to construct the document formats. Thus, the corresponding parties can view the document from any usual web browsers. In our work, three input tools, i.e., key-in, voice input, electronic pen, are provided to co-sign a document. The integration of these tools matches the traditional as well as modern requirements. To accurate control the document process speed, we design a inform mechanism to monitor the document processing status. A file storage management system and statistical reports are also included in the design to facilitate performance evaluation. Finally, the present system also supports access and transmission security. Currently, we are testing the system in MINE Laboratory at Tamkang University.

VI. REFERENCES

- [1] Lotus Notes: <http://www.lotus.com/notes/>
- [2] Microsoft exchange: <http://www.microsoft.com/exchange/nwseven.htm>.
- [3] Novell Groupwise: <http://www.novell.com/groupwise/prods/gwise/gwwht.html>.
- [4] Charlie Kaufman, Radia Perlman, Mike Speciner, *Network Security*, Prentice Hall, 1995.
- [5] Ralf Steinmetz and Klara Nahrstedt, *Multimedia: computing, communications & applications*, Prentice Hall, 1995.
- [6] <http://home.netscape.com/assist/security/ssl/index.html>
- [7] <http://www.ietf.org>
- [8] Adam Denning, *ActiveX controls inside out*, Microsoft Press, 1997.
- [9] Meichun Hsu and Mike Howard, "Workflow and legacy system," *Byte*, 19(7), 1994, pp.103-108.
- [10] T.A. May, "Know your workflow tools," *Byte*, 19(7) 1994, pp.103-108.
- [11] Kenneth L.Spencer, *NT Serve: Management and Control*, Prentice Hall, 1996.
- [12] <http://www.microsoft.com/iis/default.asp>
- [13] Mark Spenik, Qrryn Sledge, *Microsoft SQL Server DBA Survival Guide*, SAMS, 1996.