A Web-Based Paradigm for File Migration

Frank L. Walker and George R. Thoma
Lister Hill National Center for Biomedical Communications
National Library of Medicine
Bethesda, Maryland

Abstract

The migration of files in obsolete formats to those expected to survive into the future is a key task in proposed techniques for the preservation of electronic documents. To this end, MyMorph has been developed as a web-based file migration service that allows the bulk conversion of electronic documents to PDF in a manner that minimizes certain aspects of the migration cost. It uses client software running on the user's computer to send files via SOAP to a computer system at the National Library of Medicine called DocMorph, which converts more than fifty different file formats to PDF and returns the results to the user. The MyMorph software has been beta tested since June 2002 by more than 2,000 users who have used it to convert thousands of files to PDF. Nearly all users found the software easy to learn to use, and most report that the conversion is fast. From the user's viewpoint the software minimizes the cost of migration in two ways. First, since MyMorph is freely available, anybody with a Windows-based computer and access to the Internet can use it. Second, the interface permits file migration to proceed in batch mode, requiring minimal user interaction regardless of the number of files converted. This paper describes the architecture of this web-based paradigm for file migration, and summarizes the results of the beta test.

Introduction

Libraries and other institutions are increasingly acquiring collections in electronic form in place of, or in addition to, paper-based material. These institutions are now facing a problem that has confronted the computer industry for years: viz., how to preserve electronic archives for the far future. Not only do electronic media holding these archives decay, but also file formats face obsolescence. Two techniques for the preservation of electronic material have been discussed in recent years: emulation and migration. Emulation has been proposed as a way to use future computer technology to function as if it were the obsolete technology of a previous generation. This technique requires the archiving of both the original electronic files as well as the software for using them. Then, via emulation of the defunct computer hardware and operating system, the archived software can be run to display the archived files, thereby maintaining the "look and feel" of the original user interface. While early experiments with emulation have demonstrated some success, the jury is still out on this approach.[1] With current user interfaces (keyboard and mouse) likely to change or disappear someday, just as hard-copy teletypewriters, keypunch machines and paper tape readers from just a quarter century ago have disappeared, and with operating systems becoming very complex, it is likely to be challenging to emulate obsolete hardware and operating systems far into the future. The second approach to maintaining electronic archives, file migration, has been used for decades by the computer industry for electronic preservation, primarily of computer databases. In the sense used here, file migration is the process of transforming decreasingly used or unsupported file formats to widely used modern formats expected to survive for the foreseeable future.[2]

Investigators have cited certain drawbacks to electronic document preservation via file migration. Because archaic file formats are converted to modern formats, the software that uses the latter is likely to have a different user interface from the software that used the original file formats. This means that some user interface features may be different, or no longer exist. An example would be conversion of a word processing file to the Portable Document Format™ (PDF): while a word processing file is fully editable, after conversion to PDF the information is displayable and printable, but usually not editable. As a result, some user functionality may be sacrificed, though the original information is faithfully retained, allowing the user to display, print, or play the new file on future hardware and operating systems, without loss of the information conveyed. A key task of file migration is to judiciously select the new file format to guarantee conveyance of information without losing or altering any part of the original document. File migration has also been criticized on grounds of cost [Ref: Rothenberg, 1999]. Cost is dependent on a number of factors. One is the frequency of migration. Ideally, new file formats should last a long time, and if not, would require more frequent migration. Cost is also dependent on the software used for the migration, the size of the electronic collection to be migrated, the time to migrate each file, and the amount of human labor required to do the migration.

MyMorph

MyMorph is a technique for bulk file migration that is intended to minimize certain aspects of the migration cost. It is a web-based approach that allows the conversion of a potentially large collection of electronic documents to PDF. While the current system is designed to primarily handle electronic document files, the architecture serves as a useful model for the migration of other types of files, such as audio and video. MyMorph was developed as a file migration service at the Lister Hill National Center for Biomedical Communications, an R&D division of the National Library of Medicine (NLM).[3,4] It consists of client and server software that employ Simple Object Access Protocol (SOAP), a technology utilizing extensible markup language (XML) sent over the Internet via the Hypertext Transfer Protocol (HTTP). The MyMorph service relies on client software running on the user's computer to send files via SOAP to a system at NLM called DocMorph, which can convert more than fifty different file formats to PDF, and returns the results to the user. The MyMorph software has been beta tested since June 2002 by more than 2,000 users to convert thousands of files to PDF. Beta testers found the software easy to learn to use, and the conversion to be fast. The software is designed to minimize the cost of migration for users in two ways. First, since MyMorph is freely available, anybody with a Windows-based computer and access to the Internet can use it. This enables many organizations to avoid costly internal software development or procurement of migration software. Second, the user interface permits file migration to proceed in batch mode with minimal user interaction regardless of whether a single file or hundreds at a time are converted.

System Design

MyMorph was produced as part of an ongoing R&D program in document imaging that has spanned many aspects of electronic document conversion and preservation, Internet document transmission and document usage. MyMorph runs as a web service on the DocMorph website, which in turn is an R&D system created to investigate the issues of delivering, processing and using electronic library information.

Located at http://docmorph.nlm.nih.gov/docmorph, DocMorph was launched in May 1999. It enables remote users of web browsers to upload files for processing in five different ways. First, it can convert files in any of more than fifty format types to the PDF format. The file types accommodated include black and white images, grayscale and color images, and word processing files. Second, it can convert any of these file types to TIFF images. While PDF has the advantage over TIFF in portability, TIFF images are easier to edit, since current PDF readers do not allow editing. To help with TIFF editing, DocMorph includes a third function for splitting a multipage TIFF file into single TIFF images. DocMorph also has a fourth function for extracting text from any of the file types it processes: in 1999 it became the first publicly available website to offer image-to-text conversion via optical character recognition. Finally, as a tool for researching and improving accessibility to library information, DocMorph has a function for converting files to synthesized speech.

As shown in Figure 1, DocMorph handles user requests from both web browsers and MyMorph client software. While its multi-computer architecture permits expansion up to ten processors, the current configuration of six processors is more than adequate for handling current workloads.

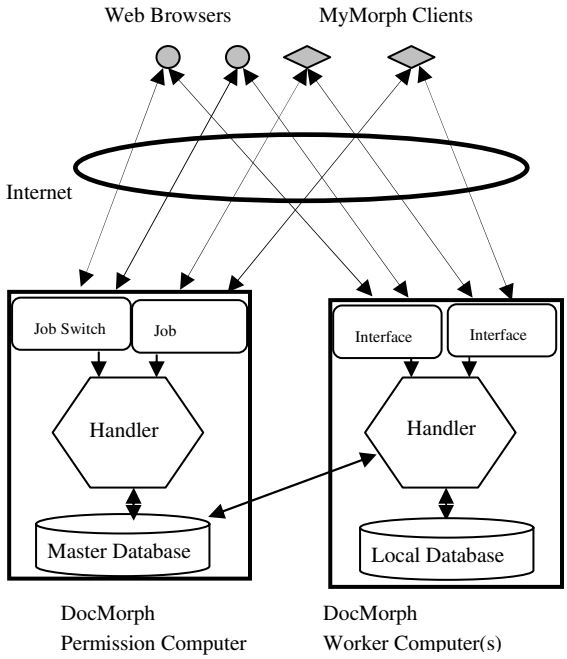

Figure 1. DocMorph/MyMorph System Architecture

One processor, termed the Permission computer, contains a master database based on Microsoft Access that keeps track of all transactions throughout the system, and retains this information for the long term. The remaining computers, of which there may be a maximum of nine, are called the Worker computers, each of which has a local database that tracks the transactions on that specific machine. Web browsers and MyMorph clients are directed to the Permission computer to request a particular job, such as file conversion to PDF. In response to such a request, the Permission computer examines the entire system to first determine which of the Worker computers can handle that type of job. Of those computers, it selects the one that has the least amount of current and pending work. It then routes the request to that computer, at which the user's file is processed. This job switching mechanism ensures that jobs are distributed evenly among the Worker computers, to avoid overloading any one.

Most of the DocMorph software is a combination of in-house designed C++ code, and two software packages, ImageMagick and Ghostscript, available on the Internet for a variety of computer platforms. DocMorph usually takes less than 20 seconds to process a job. However, this figure does not include the time taken by the user of a web browser to select a file from hard disk, upload it to DocMorph, choose a folder on the hard disk to save the returning results, and download the results to the computer. These operations may take far longer than for DocMorph to process the job. Depending on the user's Internet connection, the time taken for uploading and downloading files can vary considerably. MyMorph was designed to overcome this problem, and allow conversions of large numbers of files to PDF with a minimal amount of user interaction. At present, MyMorph is designed to handle only one of the five functions provided by DocMorph: conversion of files to PDF, since this has proven to be DocMorph's most-used function. Both the web browsers and MyMorph clients communicate with DocMorph via Hypertext Transfer Protocol (HTTP). However, MyMorph uses an additional layer on top of that: SOAP. The latter is the building block of web services that allow application-to-application communication and control over the Internet. SOAP facilitates the creation of custom client applications such as MyMorph that can improve upon some things currently done with web browsers.

An alternative approach to a web-based file migration system would be a standalone design in which the conversion software is installed and run on each computer doing the migration. This would decrease the time for conversion since data would not need to be sent over the Internet. It would also ensure security for cases in which the file data is sensitive and needs to be protected. While relatively few MyMorph beta testers have actually inquired about information security, this is a feature that may easily be included, should there be high demand for it. The approach would entail incorporating Secure Sockets Layer or encryption that would ensure security for all data sent to or received from the DocMorph system. The main advantage of MyMorph's web-based architecture over a standalone design is that when conversion algorithms are modified at the server, they become immediately available at each client, without requiring a redistribution of upgraded client software. If algorithms change in standalone software (or when bugs are fixed), it is often difficult to propagate new software to a large community of users, especially if some of these users are anonymous.

MyMorph SOAP Communication

The MyMorph client has two channels of SOAP communication with DocMorph: one with the DocMorph Permission computer, and the other with one of the Worker computers. The MyMorph client initiates the communication by using SOAP communication protocols built into a Web Services Description Language (WSDL) file. This file describes the service provided by the server in XML, and is built into the MyMorph client, unlike other types of web services that send the file from server to client. The advantage of building the WSDL file into the client is greater speed: critical time is not wasted in transferring this file across the Internet, and file conversion proceeds as fast as the network data rates dictate. A WSDL file defines three main functions available at DocMorph's Permission computer: Register, GetVersion and GetPermission. The first time MyMorph runs on the user's computer, it displays a dialog box that allows the user to register to use the software. MyMorph employs the Register function only once to send the user information to DocMorph. In response, DocMorph returns a 32-character UserID that is saved on the user's computer in an XML-based initialization file. MyMorph uses the GetVersion function upon startup to check for the latest version number of the MyMorph client software. If there is a newer version available, the MyMorph software allows the user to download and install it. This function also returns the web address at which the latest MyMorph version is located. Figure 2 shows an example of the response for GetVersion sent from DocMorph to MyMorph.

```
<xml version="1.0" encoding="UTF-8 " standalone="no "?>
<SOAP-ENV:Envelope SOAP-ENV:encodingStyle=http://schemas.xmlsoap.org/soap/encoding/"
xmlns:SOAP-ENV=http://schemas.xmlsoap.org/soap/envelope/>
  <SOAP-ENV:Body>
    <SOAPSDK1:GetVersionResponse xmlns:SOAPSDK1=http://tempuri.org/message/>
        <Result>1.0></Result>
        <WebAddress>http://docmorph.nlm.nih.gov/mymorph/setup.exe>
        </WebAddress>
    </SOAPSDK1:GetVersionResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Figure 2. Example of MyMorph SOAP Communication

The third function available at the Permission computer is GetPermission. MyMorph sends the GetPermission request to the Permission computer for each file to be converted. Included with each request is the 32-character UserID assigned to each user upon registration. The status information that DocMorph returns for this function contains two items. The first is an indication as to whether the MyMorph client may immediately send a file for processing. If DocMorph is busy and cannot handle MyMorph's request, this status data indicates a time delay before the client may repeat a request for permission. The second item is the path to the Worker computer to which the client is routed.

Once the MyMorph transmission is routed to an appropriate DocMorph Worker computer, it has access to one function available there: MakePDF. For each MakePDF request sent to the Worker computer, MyMorph sends its UserID along with the file to be converted. The output of MakePDF is the resulting PDF file. To maximize throughput via SOAP, MyMorph sends and receives files as attachments via Direct Internet Message Encapsulation (DIME). By allowing file attachments to be sent in binary form, rather than XML, DIME permits fast transmission of data since

XML-formatted data is usually larger in size than pure binary data. We did a study to determine the benefits of transmitting files as DIME attachments versus encoding them in base-64 and including them within the HTML/XML stream. Table 1 shows the measured times for sending five different size files roundtrip (from client to server and return). The files varied in size from 2 kilobytes to 15 megabytes. The tests were conducted at two speeds: 10 megabits/sec and 26.4 kilobits/sec. It is likely that real-world transmission times would fall within these two extremes, as would the file sizes. The table shows that while there was no improvement in transmission speed for small files at 10 megabits/sec, there was improvement for all other files at that speed, and there was improvement for all file sizes at 26.4 kilobits/sec transmission speed. Generally these test results show that sending files as DIME attachments results in approximately fifty percent higher throughput than encoding the files in base-64 and including them in the HTML/XML stream.

Table 1. Roundtrip File Transmission Times for DIME Attachments versus Base-64 Encoding

File Size	10 Mbs Base-64	10 Mbs DIME	% Decrease	26.4 Kbs Base-64	26.4 Kbs DIME	% Decrease
1 KB	.04 sec	.08 sec	----	6.55	1.75	73.2
11 KB	.08	.09	----	15.95	8.10	49.1
126 KB	.74	.35	52.6	96.65	45.42	53.0
1MB	7.65	2.94	48.7	1414.7	791.1	44.0
15 MB	287.82	40.82	86.0	12889	6614	48.6

MyMorph User Interface

Figure 3 shows the MyMorph user interface.

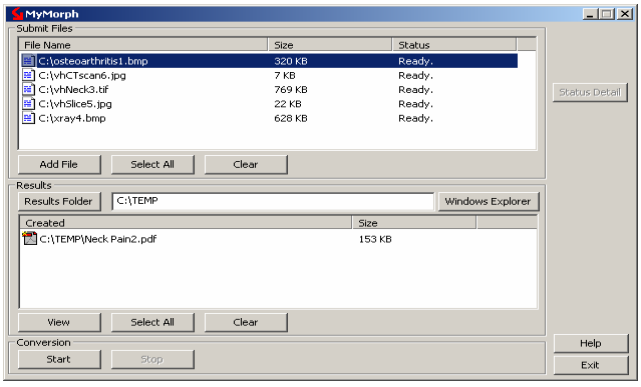

Figure 3.MyMorph User Interface

It is a single dialog box that allows the user to select many files simultaneously for conversion to PDF. Once the user clicks the Start button, the client software operates in batch mode, sending all the files listed in the upper window for conversion to PDF, and listing the returned results in the lower window. The user interface allows the user to select a thousand files or more for conversion to PDF.

The entire user interaction can take as little as ten seconds, after which MyMorph takes over, and frees up the user for other activities. Table 2 shows the increase in efficiency (or decrease in time spent by the user), at two different communication speeds, of the MyMorph user interface over a browser-based interface for converting a test set of five 970-kilobyte TIFF files to PDF.

Table 2. Operator Time Required for Converting Five 970 KB Files

Network Speed	100 Megabits/sec	26.4 Kilobits/sec
Browser Interface	3 min 2 sec	1 hour 51 min 46 sec
MyMorph	10 sec	10 sec

Results of the Beta Test

The MyMorph software went into beta testing in June 2002, and has since been downloaded and successfully used by more than 2,000 people. Users appear to fall in three categories. There are one-time users, for example students, interested in converting term papers to PDF. Then there is a small number who use MyMorph to convert large collections of files to PDF. The majority of users appear to be document delivery librarians using it daily as part of their document delivery operation. Feedback from the users was obtained both by email and by a SOAP-based feedback tool built into MyMorph.[5] While users tended to communicate bug reports via email, they used the feedback tool to provide opinions on the software. As a result of bug reports received from users, MyMorph has undergone nine revisions, and is currently in its tenth beta release. From the information provided by more than 300 users through the built-in feedback tool, we found that they tend to be employed in research libraries, and in health and academic environments. A large majority, 97%, reported that the software was easy to learn to use, 98% felt the software was useful, and 95% said they would recommend the software to colleagues. Most users (86%) also felt that MyMorph was easier to use than web-based DocMorph for conversion of files to PDF, and 96% found the software to be reliable. As for speed, 92% of users felt the software conversion was fast, although it must be noted that most users (84%) had direct, high-speed connections to the Internet. Interestingly, only one user commented that a local PC-based software conversion package would be preferable to MyMorph, because it was consuming too many Internet resources.

The Next Steps

Although MyMorph currently focuses on PDF as a target file format, we have begun investigating the work undertaken by the PDF-Archive Committee on its creation of a specification for the PDF/A file format, which is intended to become a standard of the International Standards Organization.[6,7] This work is very promising, since the PDF/A specification, a subset of PDF, might be a suitable format for long-term preservation of electronic documents. Its long-term survival as a standard format has cost implications in the infrequent migrations required. The PDF/A file would permit lossless conversion of non-PDF files to and from the PDF/A file format. It also contains a complete description of text characters and their semantic properties, fonts, images and document structure, all of which would facilitate migration of PDF/A to future file formats. Furthermore, this file format contains XML-based metadata that can be used to describe items (such as file history or keywords for searching the file contents) that could be useful in either using the file or migrating it to the next-generation file format, both with implications for preservation.

Summary

MyMorph is a web-based method for converting files to the PDF format via the Internet with minimal human interaction. Favorable user feedback shows that the software is easy to use, reliable and fast. This type of design can be used for migration of files to PDF for electronic document preservation. It minimizes the cost of file migration by saving human interaction time, and by being freely available over the Internet. The system architecture permits algorithmic improvements made at the server to be immediately available to all client software. A study is underway to determine how MyMorph can be modified to migrate files to the PDF/A specification, which shows potential for long-term electronic document preservation.

References

1. Jeff Rothenberg, An Experiment in Using Emulation to Preserve Digital Publications, RAND-Europe, Published by The Koninklijke Bibliotheek Den Haag (April 2000).
2. Jeff Rothenberg, Avoiding Technological Quicksand: Finding a Viable Technical Foundation for Digital Preservation, Commission on Library and Information Resources, (1999).
3. Frank L. Walker and George R. Thoma, Web-based Document Image Processing. Proceedings of IS&T/SPIE Conference on Internet Imaging, San Jose, California, 268-277 (January 2000).
4. Frank L. Walker and George R. Thoma, A SOAP-Enabled System for an Online Library Service. Proceedings of InfoToday 2002. Medford N.J.: Information Today, 320-329 (2002).
5. Frank L. Walker and George R. Thoma, A SOAP-Based Tool for User Feedback and Analysis, Proceedings of InfoToday 2003. Medford N.J.: Information Today, 313-322 (2003).
6. PDF Archive Committee, NWI Ballot for Document management – Long-term electronic preservation – Use of PDF (PDF/A)." Available on the Internet at http://www.aiim.org/standards.asp .
7. William G. LeFurgy, PDF/A: Developing a File Format for Long-Term Preservation. RLG DigiNews, Volume 7, Number 6. Available on the Internet at http://www.rlg.org/preserv/diginews/diginews7-6.html#feature1. (December 15, 2003)

Biographies

Frank L. Walker received his B.S. and M.S. degrees in electrical engineering from the University of Maryland. Since he joined the National Library of Medicine in 1979, he has designed, developed, performed research and published a number of papers on computer systems utilizing electronic imaging, primarily for the purpose of electronic document storage, retrieval, transmission and use. His current interest is in developing software and systems for improving the delivery and use of biomedical library information.

George R. Thoma is a Branch Chief at an R&D division of the U.S. National Library of Medicine. He directs R&D programs in document image analysis, biomedical image processing, animated virtual books, and related areas. He earned a B.S. from Swarthmore College, and the M.S. and Ph.D. from the University of Pennsylvania, all in electrical engineering. Dr. Thoma is a Fellow of the SPIE, the International Society for Optical Engineering.