Smithsonian Institution Archives

ARCHIVAL PRESERVATION OF WEB RESOURCES: HTML to XHTML Migration Test Technical Considerations, Evaluation, and Recommendations

Produced by Dollar Consulting
July 1, 2002

PREFACE

This report presents the results of a study undertaken by Dollar Consulting for the Smithsonian Institution Archives (SIA) as part of a larger effort to test and evaluate the feasibility of preserving Web sites and HTML pages in an accessible, usable and trustworthy form for as far into the future as is necessary. Specifically, this report presents the results of migrating a sample of 1,844 Smithsonian Institution pages from HTML to XHTML and storing these migrated pages in the TAR format. The target audience is the Smithsonian Institution Archives. This report reflects the Archives' understanding of its mission, requirements, and technology infrastructure. Nonetheless, it is hoped that other archivists, librarians, and preservationists concerned with preserving their Web sites and HTML pages will find this study useful as they develop their own digital preservation programs.

EXECUTIVE SUMMARY

Since 1995, when the Smithsonian Institution created its first Web site, it has increasingly employed Internet technology to inform the public of various activities and facilitated greater access to its wide ranging resources programs by offering "virtual exhibits," which only exist in electronic form. As a result, in 2002 the Smithsonian Institution has more than seventy-five (75) Web sites and thousands of HTML pages. These pages comprise a vital component of the documentary history of the nation's leading cultural research center and museum that enriches the lives of Americans and others throughout the world. The Smithsonian Institution's use of Internet technology to carry out the diffusion of knowledge is likely to expand substantially in the future, particularly in the National Museum of American History, as additional funding and support are made available.

Current and future Smithsonian Institution Web sites and HTML pages are at risk of being lost forever because of technology obsolescence. If unchecked, future generations of Americans will be deprived of the opportunity to view these original Web sites, and understand, and appreciate the vital role of the Smithsonian Institution in the diffusion of knowledge in the late 20th and early 21st centuries.

In 2001 the Smithsonian Institution Archives (SIA) commissioned a high-level requirements assessment for the archival preservation of Smithsonian Institution Web sites and HTML pages. This assessment also developed strategies, guidelines, and best practices to facilitate access to usable and trustworthy Web sites and HTML pages for as long into the future as necessary. One recommendation to help mitigate some of the effects of technological obsolescence was for the SIA to develop a program to transfer a copy of each Web site and associated HTML pages to an electronic archival repository and adopt a migration strategy to repackage these pages in World Wide Web Consortium (W3C) compliant XHTML, a technology neutral format.

Very little is known about the utility and cost-effectiveness of migration software in an on-going large-scale migration project or the resources required to implement such a program. Therefore, in 2002 the Smithsonian Institution Archives commissioned a follow-on study to assess the utility and cost-effectiveness of currently available software migration and validation tools and to develop a metric to estimate the resources necessary to undertake such a project. During the course of the study, TAR (Tape Archive), a technology neutral electronic format to encapsulate migrated and validated XHTML pages, was also explored.

The study employed an HTML test bed from the Archives Center of the National Museum of American History Web site that consisted of 1,844 HTML pages for a three-part analysis. The first part of the analysis focused on the actual migration of HTML pages to XHTML pages. With a completed time of 57 minutes or slightly less than 2 seconds per HTML page, Tidy.exe should be the preferred software package used to migrate HTML pages to XHTML.

The second part of the analysis examined the validation of XHTML pages to ensure they comply with the World Wide Web Consortium (W3C) XHTML standards. Users may access a W3C validation service either by opening the URL or by opening the HTML-Kit, which is an integrated software package. Two pages that were migrated from HTML to XHTML were sent to the W3C validation service using the two approaches described above. The HTML-Kit required 2 minutes and 5 seconds to complete validation of a single XHTML page while the same validation process using the W3C validation service required 2 minutes and 28 seconds.

The use of TAR to encapsulate validated XHTML pages was not directly done because no tape drives were available but estimates were extrapolated from other studies that suggest validation would be very speedy and its

overall cost negligible. For example, the data transfer rate in a TAR encapsulation would be on the order of 1 MB per second so actual processing time would not be great even for a Web Site of 10,000 pages. Correct execution of TAR requires knowledge of the software and understanding of DOS command structure.

The results of these analyses were integrated into a resource allocation metric that the Smithsonian Institution Archives can use to estimate the resources required to migrate, validate, and encapsulate a specific number of HTML pages. Use of this resource allocation metric suggests that there are three implementation options:

1. CombineTidy.exe with direct access to W3C validation service. Use of this option would take approximately 520 hours or 13 weeks to complete the migration, validation, and encapsulation of 10,000 HTML pages.
2. Combine Tidy GUI with direct access to W3C validation service. This combination of software tools would take approximately 1100 hours or 28 weeks to migrate, validate, and encapsulate 10,000 HTML pages.
3. Combine Tidy.exe with HTML-Kit integration of W3C validation service. With this combination of software tools the migration, validation, and encapsulation of 10,000 HTML pages would take approximately 442 hours or 11 weeks.

The recommendation to the Smithsonian Institution Archives is to implement Option 3.

---

1. INTRODUCTION

1.1 Purpose

In 2001 the Smithsonian Institution Archives (SIA) commissioned a white paper on "Archival Preservation of Smithsonian Institution Web Sites and HTML Pages." Among other issues, the white paper addressed the issue of long-term access to usable and trustworthy SI Web Sites and HTML pages and called for the SIA to adopt a policy of converting SI Web sites and HTML pages from HTML 4.0 (or earlier) to XHTML when they are accessioned into the archives. This report is a follow-on to the white paper recommendation. It provides the SIA with a metric for assessing the cost and feasibility of adopting and implementing an archival preservation policy that mandates converting Web Sites and HTML pages to XHTML once the SIA accessions them.

1.2 Scope

The overall scope of this report was set by the terms of reference for the study, which stipulated the following:

1. Review relevant literature and World Wide Web Consortium publications on XHTML,
2. Identify software, including the "Tidy Utility," for converting HTML 4.0 (and earlier) pages to XHTML,
3. Acquire or gain access to the appropriate migration software,
4. Use the SI Web "test bed" to determine the level of technical expertise required,
5. The through-put rate, and the accuracy of migration to XHTML, and
6. Prepare a final report that presents findings and recommendations.

The test bed referred to above consists of 135 MB of HTML pages, GIF and JPEG images, and AVI material from the Archives Center of the National Museum of American History. The HTML pages (1,844 pages in 14 folders) represent about one-third (45.6 MB) of the test bed. The migration of the HTML pages to XHTML has no affect on the GIF and JPEG images or AVI material, which means that as those formats become obsolescent they must be updated to successor formats.

During the course of the study, it became apparent that the Tape Archive (TAR) format, which is a well-established technology neutral encapsulation storage format, should be included in the study, and after consultation with the Project Director the study was expanded to include it. Hence, the report examines both the migration of HTML pages to XHTML and the encapsulation of HTML/XHTML pages in TAR. TAR and XHTML are not mutually exclusive so TAR can be used to supplement XHTML.

One other key scope consideration is that the focus of this migration project is archival preservation, not operational management of Smithsonian Web sites and HTML pages. Some Smithsonian Institution Webmasters may choose to convert their current HTML pages to XHTML but this an operational issue and is beyond the scope of this study.

1.3 Methodology

The methodology employed in producing this report includes three components. The first component is a literature review and analysis of relevant source material relating to migration of HTML pages to XHTML pages and to software tools currently available that support this migration. The second component is the design of evaluation criteria that could be mapped against the requirements for technical expertise required, the throughput rate, and the accuracy of migration. The third component is the migration of 1,844 static HTML pages in the Smithsonian Institution Web test bed taken from the Archives Center of the National Museum of American History. This project focuses on the static HTML pages, which require 45.6 MB of storage. Many of these HTML pages include GIF and JPEG images or links to them along with links to audio data.

Milovan Misic, Head of Document Management and Archives at the World Intellectual Property Organization in Geneva, Switzerland handled the computational aspects of the test bed migrations. Limited resources precluded an actual encapsulation of the SI test bed in TAR. Rather, documentation manuals and several published reports on the use of TAR in different technology settings were the sources for the assessment of TAR.

1.4 Report Organization

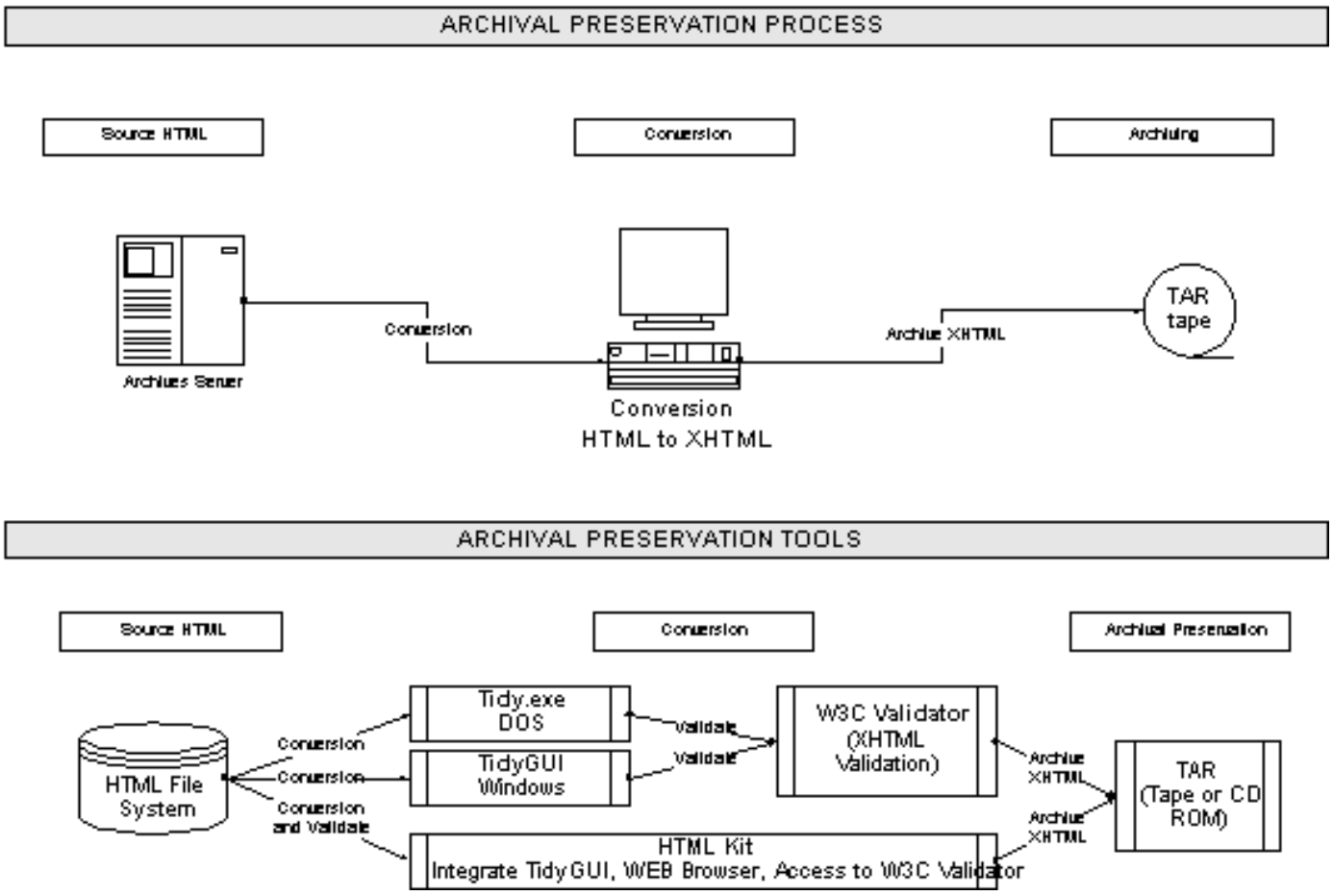
This report consists of five chapters and one appendix. It begins with an introduction to the study and delineates briefly the purpose, scope, and methodology of the study. Chapter 2 provides technical details on two HTML to XHTML migration tools, HTML Tidy Utility and HTML-Kit, and TAR encapsulation of HTML/XHTML pages. Chapter 3 addresses special hardware and software requirements. Chapter 4 addresses eight specific migration issues involved in using Tidy Utility and HTML-Kit and reviews the use of the World Wide Web Consortium (W3C) Data Validation Service to confirm accuracy of converted XHTML pages. The final chapter presents findings and recommendations. There is one appendix, Appendix A [not included in this Web document], which allows readers to actually compare the HTML code in the source documents with the XHTML code in the migrated documents as well as browser presentations.

2. TECHNICAL CONSIDERATIONS

2.1 Introduction to migration tools

This chapter focuses upon two software migration tools. Tidy Utility and HTML-Kit can "clean up" HTML pages and convert these "cleaned up" HTML pages to XHTML, which is a technology neutral file format. The following diagram provides an overview of these two tools and their functions in the migration process:

Figure 2.1 Software HTML to XHTML Migration Tools



2.1.1 HTML Tidy Utility

There are two different HTML Tidy Utility tools that can be used to "tidy up" HTML pages by fixing a host of problems, including:

- Misplacement of elements
- Uppercase versus lowercase elements and attributes
- Quotes around attribute values
- Adding correct XHTML declarations when prompted.

As noted above, Tidy Utility software is available in two different modes. The first is Tidy.exe, which is command line (DOS) software initially developed by David Raggett for the World Wide Web Consortium. Tidy.exe supports 43 different options or parameters that allow users to customize clean up and migration. Selecting these options is cumbersome for people unfamiliar with DOS so in general, Tidy.exe is not user friendly. One of the Tidy.exe options is to display a message log of warnings that identifies each instance where Tidy.exe corrected or cleaned up HTML code to comply with XHTML requirements. This message log allows users to review each instance of corrected HTML code and accept or reject the correction, which is analogous to the "find and replace" functionality of MS-Word. This is a time consuming process that is likely to be useful only for the authors of HTML pages who want to post "valid," interoperable HTML pages on a Web site.

One very useful feature of Tidy.exe is that it can clean up and convert single HTML pages or multiple pages. The latter requires that all of the "related" HTML pages be cleaned up and converted to a separate directory. Although batch processing of HTML pages containing both text and images could result in text being overwritten on an image or some other form of misalignment, there were no instances of text being overwritten on an image as a result of batch migration of the test bed. In addition, Tidy.exe migration of HTML pages to XHTML may not consistently produce 100 per cent valid and well-formed XHTML pages in every instance, so some form of visual inspection may be prudent. Interestingly, the DOS tool in Windows 98 runs in a Windows environment where drag and drop functionalities are supported.

The second mode of the Tidy Utility is Tidy GUI, which is an adaptation of David Raggett's HTML Tidy.exe. Tidy GUI has familiar Window features that make it relatively user friendly. Tidy GUI supports all of the Tidy.exe options, which can be selected by clicking on pull-down menus. Although Tidy GUI is a significant improvement over Tidy.exe, it processes only one HTML page at a time, which can become quite tedious when thousands of HTML pages are to be converted to XHTML. Like Tidy.exe, Tidy GUI migration of HTML pages to XHTML may not consistently produce 100 per cent valid and well-formed XHTML pages, so W3C provides an on-line validation service to identify and correct errors. The W3C Validation Service is not integrated into Tidy GUI.

2.1.2 HTML-Kit

HTML-Kit, which includes a full-featured text editor, was designed to assist authors of HTML XML script to create, edit, format, validate, preview, and publish Web pages. HTML-Kit is a native 32-bit Windows program that currently runs on Windows 95, 98, XP, and ME, NT, 2000 or any other platform that emulates 32-bit Windows functionality. HTML-Kit executes the following migration and validation functions within the same software:

- Opens an original HTML page,
- Starts Tidy GUI, selects options, and executes "clean up,"
- Converts the cleaned up page to XHTML,
- Saves the newly created XHTML page,
- Validates the newly created XHTML page, and
- Obtains on-line certification that a converted XHTML page is compliant with the W3C standard.

HTML-Kit supports all of the Tidy Utility functions menus, and as a windows application, it allows the opening of multiple pages or documents at the same time but the migration process deals with one document or page at a

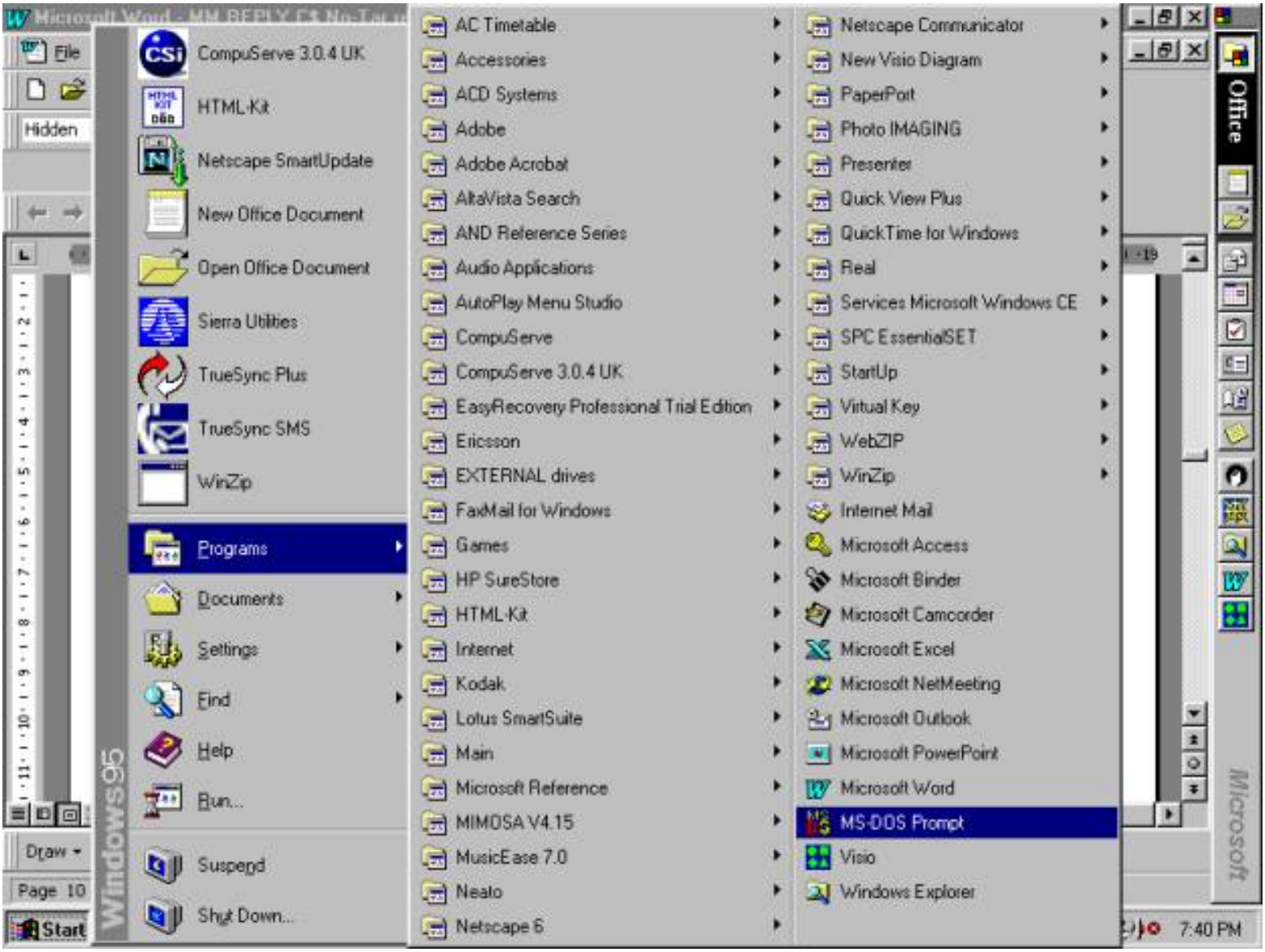
time. More importantly, it integrates the validation service into the migration so that it can be activated through the graphical user interface. Once validation is completed, the automatically corrected code is displayed in a window for side-by-side comparison with the original converted XHTML page or file. HTML-Kit supports a plugin interface functionality using third-party plugins such as JavaScript, XSLT, SMIL, MathML, WML, WMLScript, Perl, PHP and others. No programming experience is required to install plugins. All of these features combined make the HTML-Kit the most comprehensive, user-friendly, and up-to-date GUI tool to support HTML Tidy.

2.2 Tidy Utility - Migration of Web Pages from HTML to XHTML

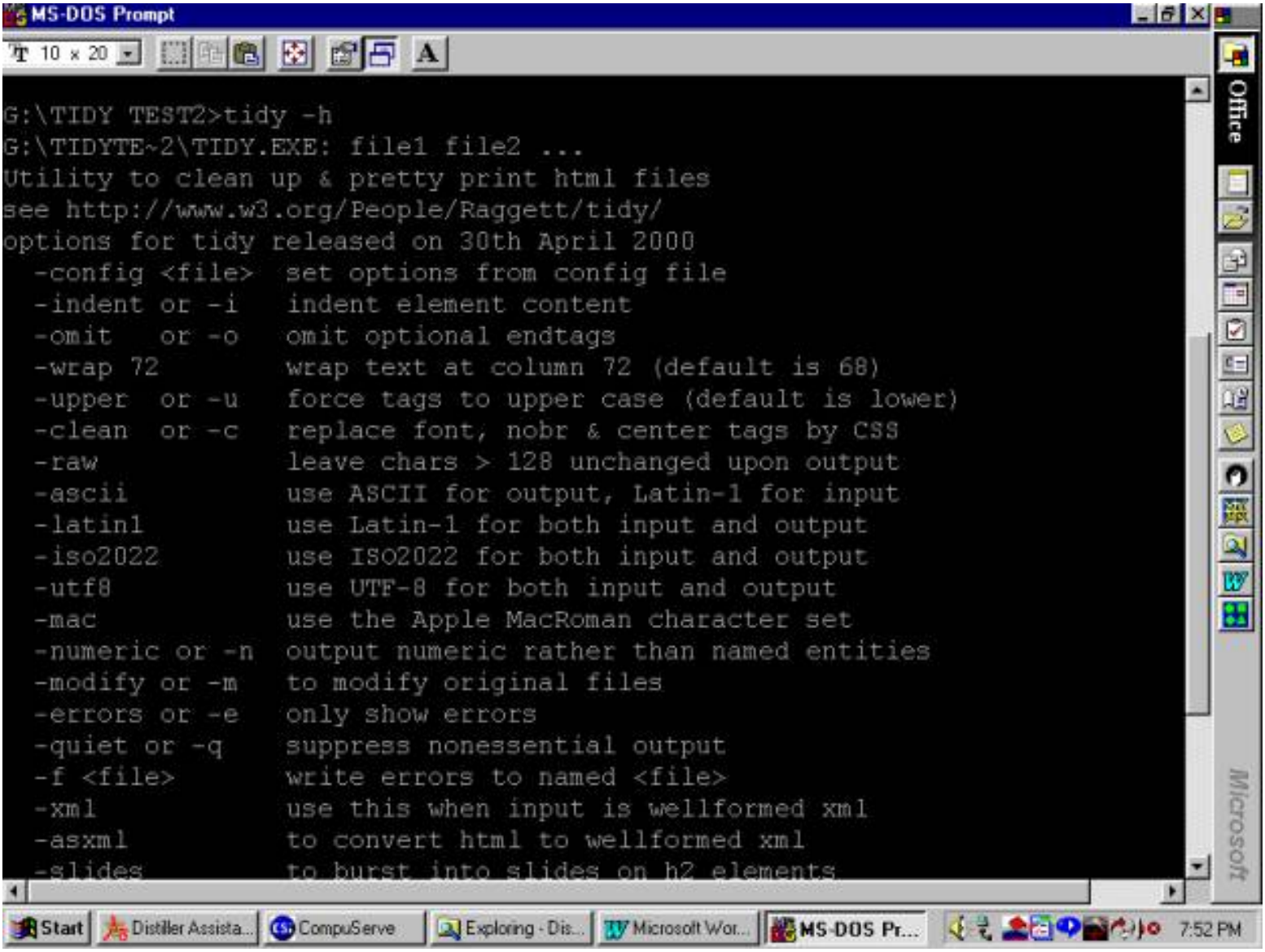
2.2.1 Setting Up and Running TIDY Utility

To use Tidy.exe to convert HTML pages to XHTML, the software must be installed in the same directory as the HTML files ready for migration. Tidy GUI on the other hand works from within its own directory and can execute all functions on any HTML file regardless of where it is stored.

Using Tidy.exe requires opening a DOS screen, which can be the DOS Prompt under Windows or exit from Windows and start-up of DOS operating system. It is better to work in the Windows environment because Tidy. exe can be started through the DOS Prompt under Windows. The DOS Prompt is available as shown on the following screen shot.

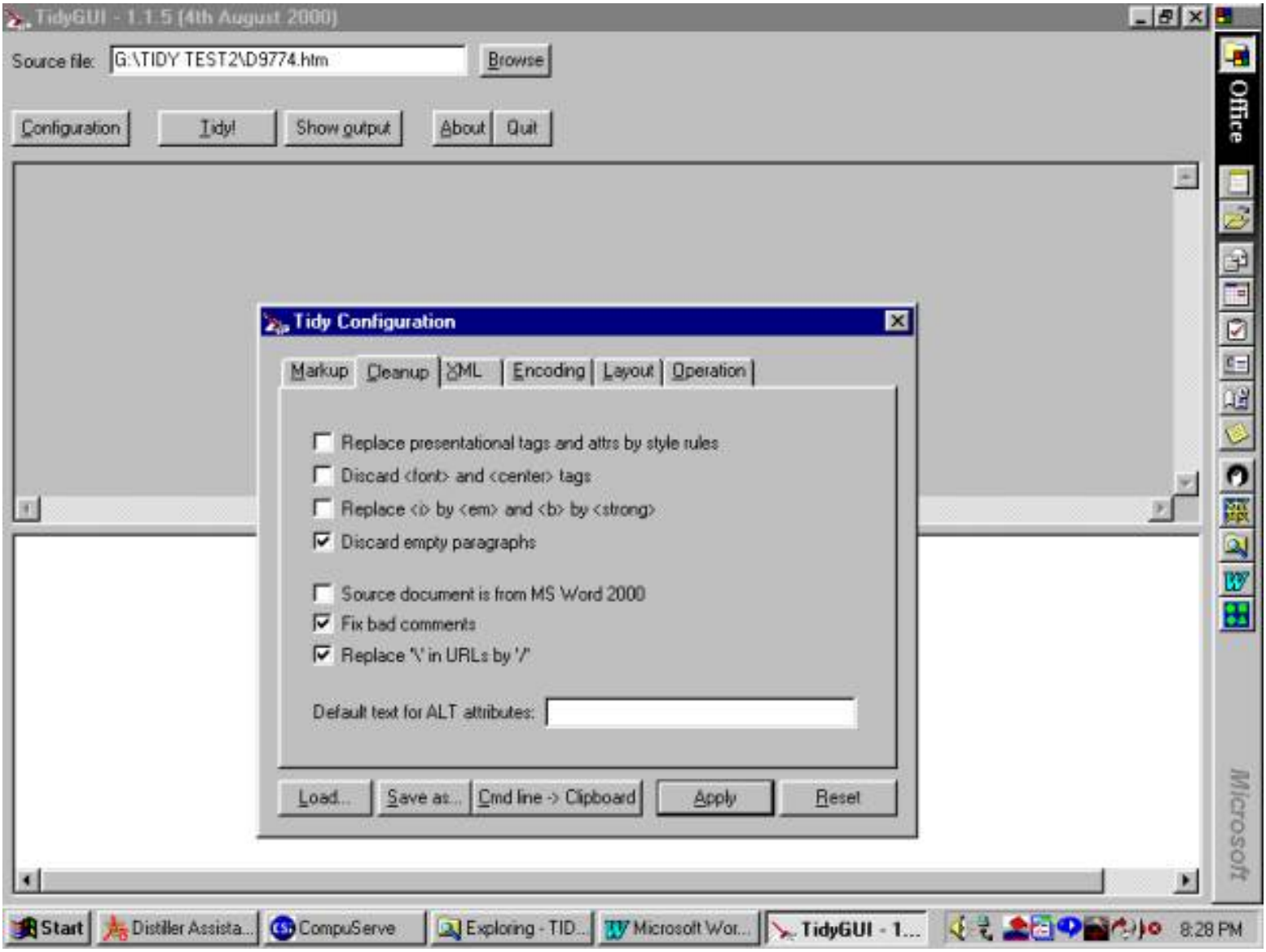


Once within the DOS environment all DOS commands are available. Begin by switching to the directory where HTML pages to be converted are located, and then initiate Tidy.exe commands. The following captured screens provide guidelines for setup of the Tidy.exe



A typical Tidy user command is: C:\migration\tidy *.htm -f errs.txt This command will list all (-e) errors on the screen and save them in the text file. The migration can be initiated with the following line: Tidy -as xml -clean filename.html > filename.xhtml

Using Tidy GUI is exactly the same as starting any other Windows application (program). After double clicking on the Tidy GUI icon, the software will appear on the screen. It is ready to execute the selected action immediately upon the selection of the HTML page by entering its file name or by using the browse on the Tidy GUI screen. The Tidy GUI configuration page identifies options that may be selected. The following screen represents Tidy GUI functions:



Running Tidy without any flags lowercases all elements and attributes and ensures that all attribute values have quotes, and checks that the HTML is well formed with no stray elements. The asxml flag adds all the XHTML features including the namespace declaration, the XHTML DOCTYPE declaration, and the XML declaration. It also makes sure that all the "empty" elements are formatted correctly for XHTML. Finally, the clean flag converts elements and center attributes to their CSS counterparts.

2.2.2 Validation with Tidy Utility

Migration of HTML pages to XHTML, whether performed manually or by a computer, is not precise. The TIDY Utility has not been adequately tested to see if it consistently produces 100 percent valid and well-formed XHTML pages in every instance. Consequently, it is important to use a computer validation tool along with visual inspection of selected pages to confirm that the migration was carried out accurately.

The best Web-based solution for XHTML validation is the W3C Validation Service (http://validator.w3.org), which allows users to submit XHTML documents to the service. The Validation Service identifies any aspects of XHTML that may have been missed in migration. Anomalies can be quickly corrected. After these anomalies are corrected, the Validator Service issues a validation certificate for each converted XHTML page. When

running the XHTML validation tool, a XHTML DOCTYPE declaration must be specified (menu selection) or the page will be validated as HTML.

The W3C Validation Service operates equally well with Tidy.exe or Tidy GUI produced XHTML pages. An upload function contains a drop down menu that requires identification of the file or page (browse feature) to be converted and if it is HTML or XHTML. Up to ten XHTML pages can be highlighted and selected for validation. The actual validation is done on the W3C validation server so that the amount of time required to validate a single XHTML will depend upon several factors that include the data transmission rate and the number of XHTML pages in the queue for validation.

2.3 HTML-Kit Migration of HTML Pages to XHTML

2.3.1 Setting Up and Running HTML-Kit

Setting up HTML-Kit consists of downloading the software from http://www.chami.com/html-kit and running the setup program. A Windows screen with familiar options and tools will open. From the file menu select "open" and either key in the name and path of the file or document or use the browser to locate it and open it. Double click on "Tools" and then select "Preferences" and select the options to be used during migration, including warning messages, split screens, and browser preview. These split screens may be used to display the original HTML page and the newly converted XHTML. HTML-Kit also supports a "browser" preview of the newly converted XHTML page in all current browsers that allows an immediate visual check.

2.3.2 Validation with HTML-Kit

One of the strongest features of HTML-Kit is its integration of the W3C Validation Service as a plugin. Unlike Tidy.exe and Tidy GUI, the Validation Service is integrated into HTML-Kit so that pointing and clicking at a specific XHTML page in a directory automatically initiates an upload to the Validation Service. As noted earlier, the Validation Service identifies any errors that may have been missed in the migration. These errors are corrected, after which the Validation Service issues a validation certificate for each converted XHTML page.

To open the W3C Validation Service through HTML-Kit, go to the tool bar and double click on the "Actions" icon and then click "On-line." Under the "On-line" options select W3C, which establishes an on-line connection with the W3C Validation Service. The service automatically detects the character encoding and the document type (XHTML in this instance). At the conclusion of the validation, a screen indicates how many warnings (corrections made) or errors were found. If errors are found, a message list at the bottom of the screen identifies the line numbers and kinds of errors. Double clicking on each error message displays the text and XHTML code as editable text and the error can be corrected. Using the "Save As" function, the validated XHTML page or document can be saved to a hard disk and later transferred to off-line storage.

2.4 Encapsulation of HTML/XHTML Pages TAR (Tape archive)

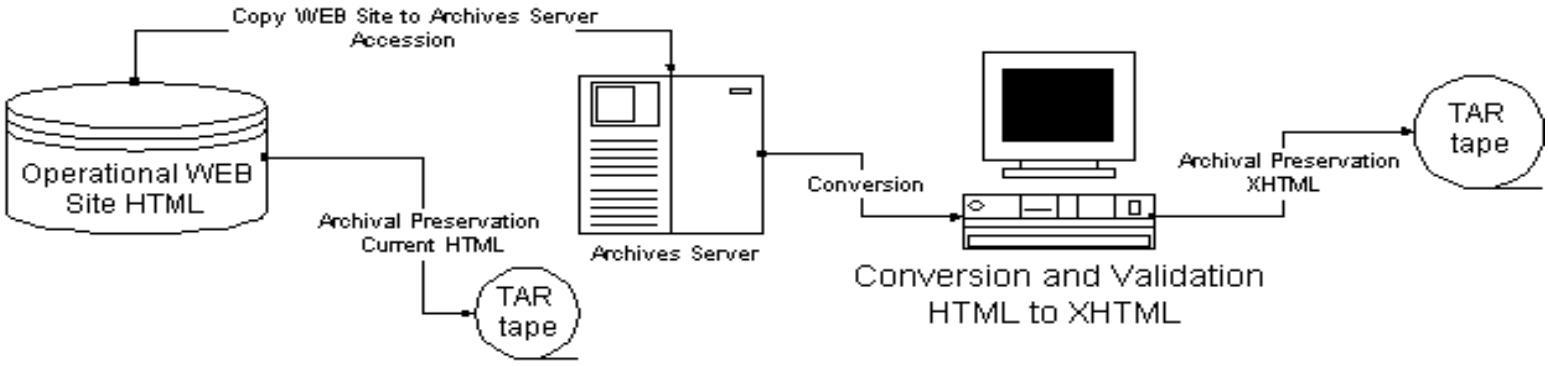
The TAR utility was initially designed for UNIX platforms but now runs on a variety of platforms. TAR can be used for running the tape in the tape-archive device (tape writer / reader) or migration from tape to CD-R or CD-RW. TAR is used to write digital material to a storage medium when the target platform is not known. TAR is intended to be a platform independent storage format so that the HTML pages in the test bed can be written to

CD using the TAR format and later migrated to XHTML or whatever format is available at the time.

2.4.1 TAR Encapsulation Process

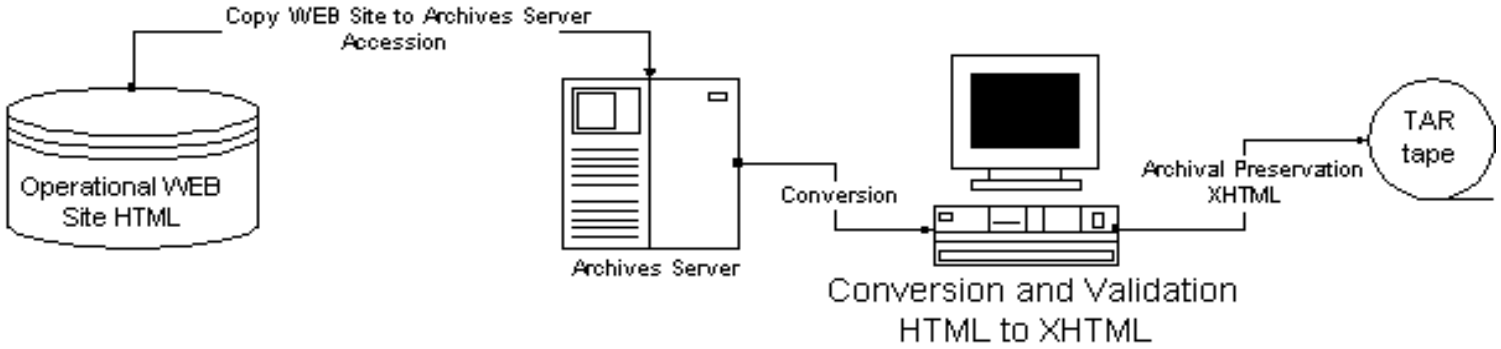
A TAR encapsulation can be done that retains either the before migration of HTML pages to XHTML, or the XHTML pages afterwards. The "before encapsulation" begins with copying the entire Web site on to a tape or CD-ROM, thus preserving the original code, layout and data from incidental technical changes, deletion or loss during the encapsulation. Tape or CD-ROM is used as the "source" for the encapsulation, which includes transfer to magnetic media during the encapsulation. The last step is the second "duplicate" on to a tape or CD-ROM that preserves the entire Web site in XHTML format after the migration.

Figure 2.2 Backup Tar Encapsulation Before Migration



Migration of the Web site HTML pages to XHTML and then encapsulation in TAR is the second option. This option assumes migration within the existing environment. However, proper technical support is necessary in order to prevent incidental loss of data. Periodic duplicate copies are recommended with the safe storage of all media until the migration is completed and verified.

Figure 2.3 Post Migration Tar Encapsulation



2.4.2 TAR Encapsulation Issues

This section presents a detailed review of key issues likely to be involved in the use of TAR to encapsulate

HTML/XHTML pages.

2.4.2.1 Encapsulation of the Test bed in TAR

The basic processes are summarized in the previous diagrams. Essentially, the TAR is a useful tool for storing a large volume of digital material on magnetic tape or optical media. TAR is not user friendly and if the storage medium is magnetic tape, there are certain steps that must be followed that to ensure full control of the process.

The following description is only an illustration of the procedure. It is included here to indicate the level of technical knowledge necessary for a valid encapsulation.
First, log onto TAR utility like this:

cdf17> TAR -cvf /dev/nst0 filename

-c stands for create
v stands for verbose so that a printout of the files will occur as they are done transferring
f stands for form
This process creates a directory in which the TAR files can be copied. Filename is the name of the file/directory designated for archiving.
*TAR indicates the use of tape archiving.

NOTE: Only the filename should be in the command line. Typing in the entire file path can cause complications later.

Restoring Files:
The tape should be in the drive. At the prompt, the following command should be entered:
cdf17> tar -xvf /dev/nst0

NOTE: -xvf will extract a file and printout the name of the file as it is extracted. The TAR command will create the directory on the tape into whatever directory is defined by the user.

2.4.2.2 TAR Encapsulation Execution Time

The time required to complete a TAR encapsulation of the HTML, or the converted XHTML test beds, depends upon the data transfer rate of the storage device and the computer platform used. For example, in most instances the data transfer rate for CD-R is significantly lower than for most magnetic storage media. The inability of gaining access to one or more tape systems to encapsulate the entire test bed meant that published reports had to be relied on. For example, the Department of Chemistry at the University of Alberta in Canada, which utilizes TAR heavily in its management of nuclear magnetic resonance data, reported its assessment of DDS-2 tape (4mm Digital Audio Tape) and CD-R. Table 2.1 presents selected features extracted from that report. The two key considerations in the assessment were the differences in media storage capacity and data transfer rates. One DDS-2 tape can store 4 GB of data while seven (7) CDs would be required to store the same 4 GB of data. The data transfer rate of DDS-2 tape is 2.5 times greater than that of CDs. The Department of Chemistry at the University of Alberta believes that magnetic storage media is "clearly the way to go" when the volume of data is very large. DDS-2 and CD-R are established storage media with known data transfer rates so with all other

things being equal it is possible to extrapolate from the University of Alberta data to other storage media, such as DLT.

Table 2.1 TAR Execution Time Comparisons

Feature	DDS-2	CD-W
Capacity	4 GB	635 MB
Speed of Transfer	1 MB/sec	400 KB/sec
Access	Sequential	Random
Device dependent	Yes	Any CD Drive
Long-term Stability	Medium	High
Cost (per MB)	40 cts	24 cts

3. SPECIAL HARDWARE/SOFTWARE REQUIREMENTS

Tape drives or CD Writers attached to the main Internet server do require reading / writing / proofing software that is usually supplied by manufacturer. While CD - Rs are readable from almost any PC, it should be noted that CD WR (re-writeable) format cannot be accessed from older types CD-drives.

3.1 Migration of HTML Pages in TAR to XHTML

There is no direct migration from TAR to XHTML. Instead, the TAR file should be extracted in its HTML coded form and then converted to XHTML using Tidy Utility.

3.2 Knowledge of TAR Format and Procedures

TAR requires a substantial level of technical knowledge for the operation as well as a thorough knowledge of special commands. TAR does not run in a GUI environment. The following list identifies TAR commands that must be fully understood.

Summary of TAR Options:
Tape drive command (only one can be selected):
c create a new archive on tape, at the same time deletes existing contents
r append a new archive on tape, preserving existing contents
x extract contents of the archive on tape
t read table of contents of archive on tape
Modifiers (multiple choices can be made):
v verbose mode; provides basic information on the process.
R place files or file hierarchies (subdirectories, etc) relative to the current directory. Without this extension, tar reads files/directories from tape and puts them right at their locations when the tape was recorded.
o (lower case o) give ownership of the files to the user executing tar. This option MUST be used if tape was

created by third party. Otherwise TAR will read the tape and will try to create files in default directories that are owned by the creator of the tape. The shell will respond with 'could not create filename' error messages, and no files will be copied.

f this option must be followed by a tar filename. Tar will use this filename as its input when writing to tape, and as its output when reading from tape. If the filename is a minus sign, then it will use stdin/stdout.

3.3 Scalability of TAR

TAR is inherently scalable because the only limit on the volume of Web sites and HTML pages is the density of the storage medium. For example, TAR works equally well with CD-R with a storage capacity of 650 MB, with DDS-2 tape with a storage capacity of 4 GB, and DLT 7000 tape with a storage capacity has a storage capacity of 40 to 60 GB. New files can also be added to the existing tape. This scalability feature of TAR presumes that all required instructions and operations are carried out correctly.

3.4 Writing TAR Files to Tape

When writing TAR files to tape, the 'Write' option must be set. Any file or directory icon can be dragged and dropped into the tape drive window. As this is done, a list of files is created. Any of these files can be removed from the list by clicking the Remove from List button. Note that placing files into the window does not copy them to the tape. Only after Apply is clicked, the tape loaded, and Accept is pressed, will the files be copied to tape. The tape can be ejected by highlighting the tape drive icon and choosing Eject from the Selected tool chest.

3.5 Confirming the Content of A TAR Encapsulation

The content of a TAR encapsulation can be confirmed by clicking" Apply" when the List option is set. After the tape(s) is loaded, the contents can be listed by clicking "Accept."

3.6 Proprietary Issues associated with the Use of TAR

TAR was designed for use within a UNIX environment but the code is open source and can be run in many environments, which makes TAR a technology neutral file format. In this sense, therefore, TAR output is platform independent and it could be used to write HTML/XHTML pages to a storage medium when the target file format or platform is not known. For example, XHTML pages encapsulated in the TAR format and written on DLT or DVD media could be placed in the archives for long-term preservation and at some point in the future extracted back to the current browser/viewer technology platform. There are some anomalies in using TAR and operating the tape loaded into the tape drive. The following discussion explains the handling and commands necessary to prevent erase or loss of data. Some technical familiarity would be the minimum requirement for operating the hardware.

If a tape has been used to store some data, taken out of the tape drive, then reinserted for copying additional data, TAR WILL NOT automatically wind the tape to the end of the data previously written. The tape must be spaced forward beforehand. This may be done by listing the contents of the tape, thus leaving it at the end.

To do this use the command: tar tvf /dev/dat1 (or dlt1)

If a tape has not been removed from the tape drive after one lot of archive, it might or might not be at the end of the saved data. To be certain, you should take the tape out and reinsert it, and forward space it as described above.

To extract the contents of a tape use the command: tar xvf /dev/dat1 (or dlt1)

4. EVALUATION OF TIDY UTILITY AND HTML-KIT

The Smithsonian Institution Archives plans to initiate a program of archival preservation of HTML pages from more than seventy-five (75) Smithsonian Institution Web sites. A major part of this archival preservation program is the migration of HTML pages to W3C compliant XHTML, which is a technology neutral format. There is very little known about the utility and cost-effectiveness of migration software or validation services in an on-going, large-scale migration project, or the resources required to implement such a program. The purpose of this evaluation is to assess the utility and cost- effectiveness of currently available software migration and validation tools that can be used to develop a metric to estimate the resources necessary to undertake such a project.

4.1 Evaluation of Tidy Utility

This section is an evaluation of Tidy.exe and Tidy GUI software based upon eight criteria.

4.1.1 Ease of Use

As noted earlier, Tidy.exe is a command line utility that may be unfamiliar to many computer users and its limited Windows functionality (drag and drop features) might make it cumbersome to use. Tidy Utility GUI is a Windows GUI adaptation of Dave Raggett's HTML Tidy, a free utility application from the World Wide Web Consortium that helps clean up web pages. Tidy is a powerful, efficient, multi-platform application that can be tuned with multiple options from the desktop. The major value of Tidy GUI is that it supports a Windows environment and all the options (some 43) for markup, cleanup, XHTML output, encoding, and different layout and user customized features. All features are user-defined based upon menu selections and therefore do not require programming skills. TIDY GUI processes one HTML page at a time.

4.1.2 Scalability

Tidy.exe can convert a single HTML page or multiple pages through a batch processing mode. The batch processing mode is important when there are hundreds or thousands of HTML pages to convert to XHTML. In contrast, the migration logic of Tidy GUI focuses on a single HTML page so it can only convert a single HTML page at a time. TIDY GUI cannot automatically process hyperlinked pages.

4.1.3 Data Anomalies

What anomalies, if any, do Tidy.exe or Tidy GUI introduce in the data? In other words, does the Tidy software support a "one to one" migration of HTML pages, or are there certain characters or features that either are dropped or distorted? Font changes and declarations that are not consistent with the original HTML Code are the most visible changes. Neither the content nor the layout of drawings and text in the test database were changed

during the migration. However, the rendering of content on a screen varied slightly between browsers. In several instances, Netscape 4.7 could not open certain graphical pages.

4.1.4 Architecture

Neither Tidy.exe nor Tidy GUI breaks internal and external links in any way. Links are not affected either by the cleanup or migration process or the overall integrity of the HTML test database. Nonetheless, it should be noted that migration and validation require considerable human intervention so there is the potential for errors to occur. A rigorous quality assurance program can address this problem.

4.1.5 Web Page Text/Script Presentation

As a general rule, the use of Tidy.exe or Tidy GUI does not substantially affect the presentation of Web page text/script presentation. In some instances, a slight difference in spacing around graphics might occur due to different font in the original text.

4.1.6 Web Page Image Presentation

Neither Tidy.exe nor Tidy GUI affects the use of images in HTML pages because the "clean up logic" focuses only on the internal structure of HTML code and does not interfere with the coding of images. Consequently, the use of Tidy.exe or Tidy GUI has no affect on GIF or JPEG images.

4.1.7 Web Browser Presentation

Apart from a slight difference in spacing or wrapped text around an image, there is very little observable difference in the way Netscape or Internet Explorer presents material for viewing.

4.1.8 Computer Execution Time

The Smithsonian Institution has thousands of HTML pages that at some point must be migrated to XHTML pages. What level of resource allocation is required for such an effort using TIDY Utility? The elements that would be involved in such an effort include the data transfer rate of input/output device and the technology platform used.

Tidy.exe was run against the entire test bed and Tidy GUI was run against selected HTML pages from the HTML test bed on several different platforms. Using a Pentium 233, Tidy.exe required almost 57 minutes to correct and migrate all of the 1,844 HTML pages. It generated a "warnings" log of more 300 pages that contained more than 59,000 "warnings." Table 4.1 displays batch migration times using Tidy.exe in DOS mode under different Windows versions with different processors. Faster platforms, such as the Pentium 4 1.7 GHz, that run under Windows XP reduced Tidy's execution time dramatically to only 8 minutes with identical results. In contrast, the average Tidy GUI execution time on the slowest platform, a Pentium 233, was about four seconds per MB. This suggests that Tidy GUI could clean up and migrate all of the 1,844 HTML pages in the test bed in about 10 minutes. However, the execution time of Tidy GUI represents only a small percentage of the total time required to complete migration. It takes about thirty (30) seconds to work through the Tidy GUI menus and select the appropriate HTML page to be migrated (migration options may be saved so that they will be

equally applied to all HTML pages throughout the migration). Using Tidy GUI to convert all of the 1,844 HTML pages in the test bed would require more than fifteen (15) hours of work.

Table 4.1 Tidy Batch Migration Execution Time

Processor Pentium	Operating System	Number of HTML Pages	Number of HTML "Errors"	Migration Time
Pentium 233	Windows 95	1,844	59,489	57 minutes
Pentium 4 1.7 GHz	Windows XP	1,844	59,489	8 minutes

4.2 Evaluation of HTML Kit

This section is an evaluation of HTML-Kit's migration of HTML pages to XHTML and their subsequent validation based upon eight criteria. So far as migration of HTML pages to XHTML is concerned, HTML-Kit implements Tidy GUI. Consequently, most of the assessments in this section track closely with those of Tidy GUI.

4.2.1 Ease of Use

As noted earlier, HTML-Kit runs in a Windows environment so it has all of the migration features that Tidy GUI supports, which makes it very user-friendly. No programming skill is required to download and install HTML-KIT or to implement "plugins." The major difference in terms of ease of use between HTML-Kit and Tidy GUI is that the former seamlessly integrates the validation service into its functionalities.

4.2.2 Scalability

HTML-Kit can support multiple documents (windows) but because it actually implements Tidy GUI it can only migrate a single HTML page at a time. Like Tidy GUI, HTML-Kit cannot automatically migrate internal hyperlinked pages. They must be processed as "stand alone" HTML pages that have hyperlinks.

4.2.3 Data Anomalies

What anomalies, if any, does HTML-Kit introduce in the data? The HTML migration to XHTML component of HTML-Kit is in fact Tidy GUI, so font changes and declarations that are not consistent with the original HTML Code are the most visible changes. Neither the content nor the layout of drawings and text in the test database were changed during the migration. However, the rendering of content on a screen varied slightly between browsers. In several instances Nettscape 4.7 could not open certain graphical pages.

4.2.4 Architecture

Like Tidy.exe and Tidy GUI, HTML-Kit had no affect on the "architecture" of the test bed. HTML-Kit retained all internal and external links during the clean up process so the overall integrity of the HTML test database was not affected.

4.2.5 Web Page Text/Script Presentation

Because HTML-Kit simply implements the existing Tidy GUI functionalities and options, there may be some slight differences in spacing around graphics that might occur due to different fonts in the original text. However, this is more cosmetic than anything else.

4.2.6 Web Page Image Presentation

Using HTML-Kit to convert HTML pages that contain GIF and JPEG images has no affect on the images, just as is the case with Tidy GUI.

4.2.7 Web Browser Presentation

Aside from an occasional, slight difference in spacing or wrapped text around images, there are no visible different differences in the way Netscape or Internet Explorer interpret XHTML objects.

4.2.8 Migration Execution Time

As noted earlier, the HTML to XHTML migration functionality of Tidy GUI is implemented within HTML-Kit so there was no difference in computer execution times when individual pages from the test bed were migrated. It took only 200 seconds for the computer processor to finish the migration of all 1,844 HTML pages into XHTML. However, as the migration can only process one HTML page at a time, additional time is required for an operator to check warnings of repairs made, resolve errors, and load a new HTML page to be repaired and migrated.

There are two important differences with the HTML-Kit implementation of the Tidy GUI HTML to XHTML migration. HTML-Kit displays split screens so that double clicking on a "warning" in one screen automatically displays the HTML code in the second screen that has been corrected. Clearly, this could lead to a significant reduction in the time required to verify corrections. The migration options (flags) initially established are automatically repeated for successive migration. Nonetheless, overall it would probably take an operator about 20 seconds to review warnings, resolve errors, and retrieve the next HTML page to be converted to XHTML. Using the HTML-Kit to convert the 1,844 HTML pages in the test bed to XHTML would still probably require at least ten hours of tedious work.

4.3 W3C Validation Service

As noted earlier, the Tidy Utility may not consistently produce XHTML pages that are fully compliant with the W3C standard. To ensure that XHTML page are compliant, W3C established a Web-based Validation Service that allows users to submit XHTML pages to the service. The W3C Validation Service can be accessed directly through its URL or through the HTML-Kit. The following assessment reviews each method.

Two XHTML pages were submitted to the Validation Service through its URL. One HTML page was 95 KB and the other was 1.15 MB. Validation of the second HTML page (1.1.b MB) required 3 minutes and 14 seconds, while validation of the smaller HTML page (96 KB) required 1 minute and 3 seconds. Assuming that an average of these two execution times - 2 minutes and 8 seconds - will recapture worst case scenarios,

execution time alone for the entire test bed would be about 65 hours and 42 minutes. Added to that should be about 10 seconds per upload request to fill out the form or clear a used form, which adds an additional 6 hours and 8 minutes. The total time required to validate the entire test bed of 45.6 MB is about 71 hours and 50 minutes. The average time to validate one XHTML page (estimated 800 KB) in the test bed using the standard W3 Validation, is 2 minutes and 28 seconds.

These same two XHTML pages were submitted to the Validation Service through HTML-Kit. The actual execution time to complete validation is the same as that of Tidy.exe and Tidy GUI using a 56 K modem. However in HTML-Kit there is virtually no setup time because it automatically invokes the validation service when a button is clicked, which should take no more than one (1) second. The average time to validate one XHTML page (estimated 800 KB) in the test bed using HTML-Kit is 2 minutes and 5 seconds.

5. FINDINGS AND RECOMMENDATIONS

5.1 Summary of Findings

1. HTML to XHTML Migration Software

A review of currently available application tools for converting HTML pages to XHTML identified Tidy Utility. exe, Tidy Utility GUI, and HTML-Kit that support the cleaning up of HTML code inconsistencies and migration of the "cleaned up" pages to W3C compliant XHTML. Each software package, which is available at no cost, was run against the HTML test bed that consisted of HTML pages, GIF and JPEG images, and AVI material that totalled 135 MB. About one-third (45.6 MB) of the test bed actually represented HTML pages (1,844). Eight evaluation criteria were used to assess each software tool. Table 5.1 summarizes the results of this assessment.

Table 5.1 Comparison of Tidy.exe, Tidy GUI, and HTML-Kit Migration of Test Bed

Criteria	Tidy.exe	Tidy GUI	HTML-Kit
Ease of Use	No	Yes	Yes
Scalability	Yes	No	No
Data Anomalies	No	No	No
Affect Architecture	No	No	No
Affect Web Text Presentation	No	No	No
Affect Web Page Presentation	No	No	No
Affect Web Browser Presentation	No	No	No
Computer Execution Time (1,844 pages)	57 min	200 sec	200 sec
Operator Time (1,844 pages)	0	15 hours	10 hours
Total Estimated Time (1,844 pages)	57 min	15 hours +	10 hours +
Average Time to Migrate 1 HTML Page	2 sec	2 min 51 sec	1 min 54 sec

There are several observations about Table 5.1 that merit consideration. First, none of the three software tools caused data anomalies and there were no significant adverse effects on architecture, Web text and image presentation, or Web browser presentation. Second, Tidy.exe is not very user-friendly while Tidy GUI and HTML-Kit are user friendly. Third, only Tidy.exe is scalable in the sense of having the capacity to batch process multiple HTML pages. Fourth, the computer execution time and operator intervention time required to complete the migration clearly differentiates the three software packages. With a total completion time of 57 minutes, Tidy.exe is far superior to Tidy GUI and HTML-Kit with a total completion time of more than 15 hours and 10 hours respectively. There is a trade-off between scalability which affects total completion time and ease of use. On the other hand, the user friendliness of Tidy GUI and HTML-Kit is off set by the additional operator time required to complete migration.

2. XHTML Validation

XHTML validation is a critical component of an overall strategy for the migration of HTML pages to XHTML because it ensures that the XHTML pages comply with W3C standards. Over time, this could be a critical factor in minimizing problems in upgrading to new vendor neutral technology standards that will inevitably emerge. The failure to execute validation each time HTML pages are converted to XHTML could lead to a situation where a massive and costly migration project would be required to migrate HTML pages to new vendor neutral technology standard(s).

Users can access the W3C Validation Service by going directly to the URL (http://validator.w3c.org) and keying in the name and path of the XHTML page or invoking the browser to locate the XHTML page to be validated. An alternative to the W3C Validation Service is to use the integrated software functionality of HTML-Kit that supports a one-click validation action, which typically takes less than one (1) second. Once the upload of the XHTML page is completed, the actual computer execution time should be the same because the validation is being done on a W3C server. Two converted XHTML pages from the test bed were submitted to the W3C Validation Service. The major difference is that in HTML-Kit there is virtually no "preparation" time because HTML-Kit automatically invokes the validation service when an operator clicks a button, which should take no more than 1 second. The average time to validate one converted HTML page (estimated 800 KB) in the test bed using HTML-Kit is 2 minutes and 5 seconds. Thus, executing the W3C Validation Service through HTML-Kit to validate the entire test bed would take approximately 65 hours and 54 minutes. Table 5.2 displays a comparison of the estimated time requirements for the two techniques.

Table 5.2 Comparison of Validation Services Time Estimates

Feature	W3C Validation URL	W3C Validation HTML-Kit
Preparation time (1,844 XHTML pages)	6 hours 8 minutes	1,844 sec
Execution time (1,844 XHTML pages)	65 hours 42 minutes	65 hours 42 minutes
Total Time (1,844 XHTML pages)	70 hours 47 minutes	65 hours 45 minutes
Average time per XHTML page	2 minutes 28 seconds	2 minutes 5 seconds

3. Off-Line and Duplicate Copy Storage of Validated XHTML Pages

A focus on off-line and duplicate storage of W3C validated XHTML pages was originally not within the scope of this project. However, as the project evolved, it became apparent that some attention should be given to this issue. There are several reasons for this. First, it is a matter of prudent management to retain a duplicate copy of HTML pages before they are converted to XHTML in the event there is a catastrophic failure during migration or validation. This also applies to migrated and validated XHTML pages. Second, although TAR was originally developed to support magnetic media in a user environment, it currently is media independent. Third, in some instances the number of bytes in a given directory or sub-directory of HTML pages is likely to exceed the storage capacity of a CD-R (635 MB) and it is easier to write to one magnetic tape than to write to multiple CDs. Fourth, there is no other vendor technology neutral product (e.g., XML based encapsulation) on the market that has a comparable user base. This translates into a market place persistence for TAR, especially in institutions of higher education. Of course, over time this market place presence may diminish but for many years to come it will be possible to extract XHTML pages from TAR encapsulated records.

TAR is DOS based so it is not especially user-friendly and proper use of TAR requires knowledge and understanding of its functionalities. TAR runs in a limited Windows environment and this could evolve into a more substantial GUI functionality that would make TAR more usable. In the meantime, users of TAR for security and copy purposes should plan to either acquire expertise or to contract for this expertise.

4. Resource Allocation Metric

The Smithsonian Institution has about seventy-five (75) Web sites that contain an unknown number of HTML pages. An ideal strategy for the archival preservation of these Web sites and associated HTML pages is to identify those Web sites and associated HTML pages whose preservation is essential and to arrange for their transfer to the archives. Once in the archives, a systematic program of migration to XHTML could begin. What resources would be required for such a migration? Without detailed information about these essential Web sites and associated HTML pages, it is not possible to project overall costs. However, the findings of this study can be integrated into a resource allocation metric that could be used to estimate the staff resources and the amount of time required to migrate a specified number of HTML pages.

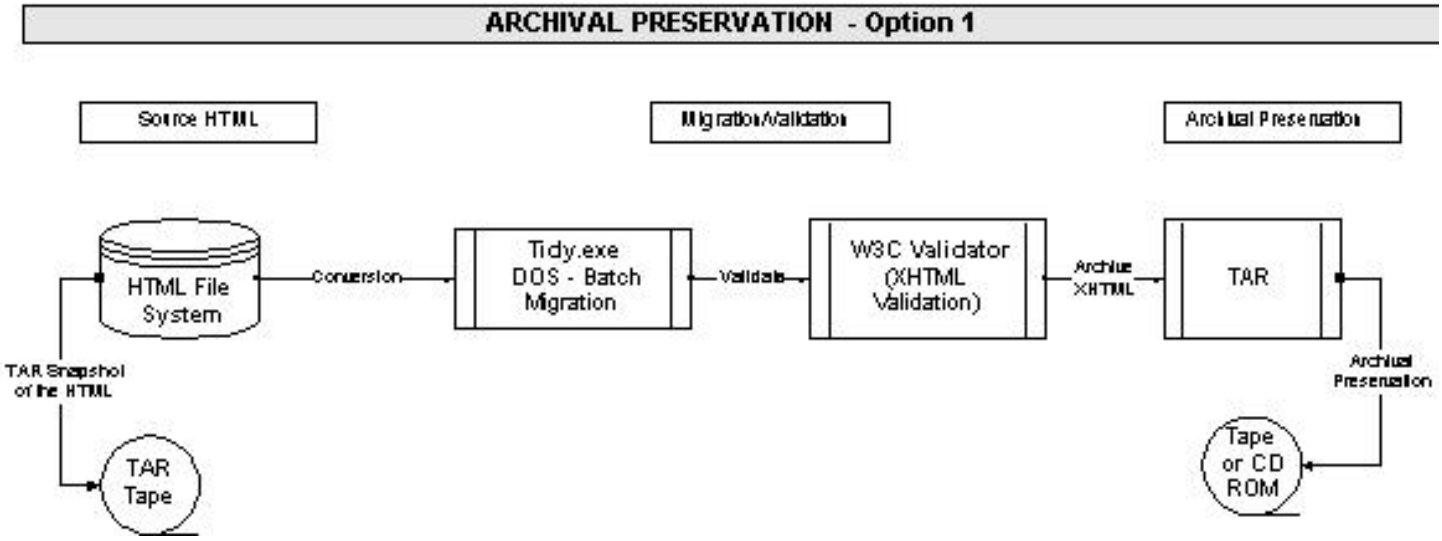
This metric focuses upon HTML pages and assumes the following:

- Static HTML pages,
- The average HTML page contains approximately 800 KB of data,
- Migration software and validation services are free,
- One PC work station that is at least a Pentium 233 is available,
- Expertise is available to set up TAR encapsulation, and
- Access to a magnetic tape drive and CD-burner

Ideally, the metric would consist of three elements that take into account the set-up, execution, and review time for migration to XHTML, validation of XHTML, and encapsulation of XHTML pages in TAR. Unfortunately, at this point it is not possible to identify in sufficient detail the level of resources required for TAR encapsulations. However, this study does suggest that it would take one person between two and five minutes and 18 seconds to convert one HTML page to XHTML and to validate it as compliant with the W3C XHTML standard. To identify the staff resources required, multiply the estimated number of HTML pages to be converted by two and five minutes and 18 seconds, respectively. For example, if 10,000 HTML pages were involved, the metric suggests that it would take between 346 hours (eight weeks) and 883 hours (almost 22 weeks). Of course, it is unrealistic

to expect anyone to work at this sustained rate even for one day, so a "fudge" factor of 25 per cent should be added to the total amount of staff time. This results in an estimated completion time that ranges between 432 hours and 1100 hours.

The diagrams in Implementation Options 1 - 3 delineate three high-level implementation scenarios for the migration, validation, and preservation of Smithsonian Web resources. Each of the scenarios involves the use of TAR encapsulation. A target migration goal of 10,000 HTML pages is used in each option to facilitate comparability.



This implementation option calls for the combination of Tide.exe (DOS batch mode) to migrate HTML pages to XHTML and direct access to the W3C Validation Service to ensure that the XHTML pages comply with the W3C standard.

Advantages

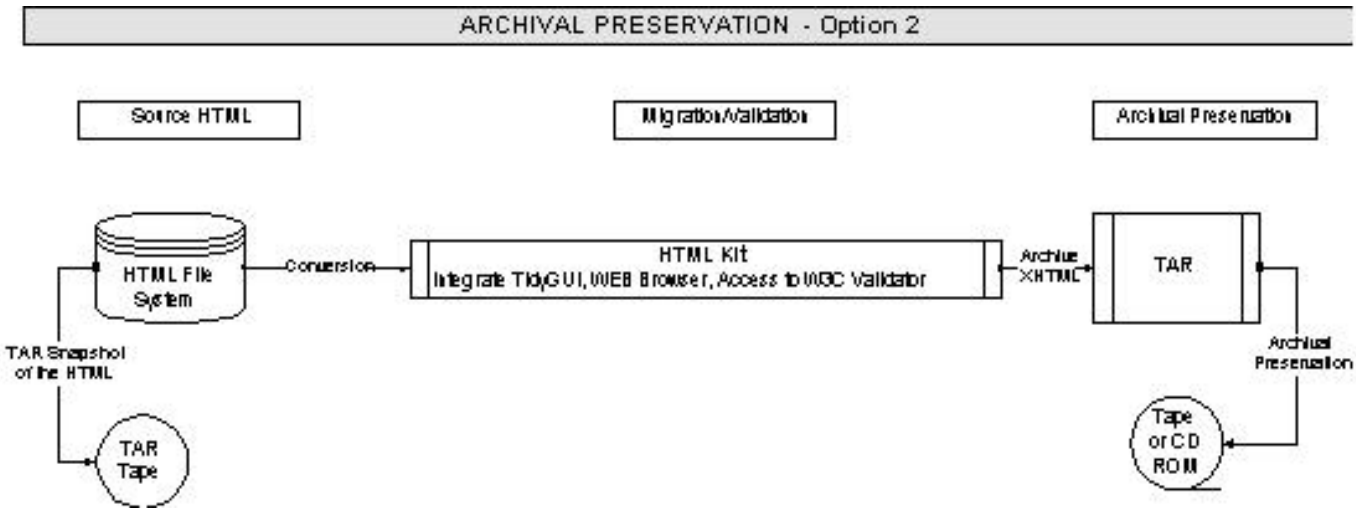
- A TAR snapshot will preserve the current WEB site
- A batch migration can be done with little human intervention during the process
- A batch migration of 10,000 HTML pages will take approximately five hours

Disadvantages

- Knowledge of DOS commands is required
- If there are technical problems during migration, the entire Web site must be reloaded and the process restarted
- Data anomalies, including errors, can only be corrected after completion of the migration process
- Validation of 10,000 converted XHTML pages will take 520 hours or 13 weeks

Implementation Option 2

This implementation option involves the combination of Tidy GUI to migrate HTML pages to XHTML and direct access to the W3C Validation Service to ensure that the XHTML pages comply with the W3C standard.



Advantages

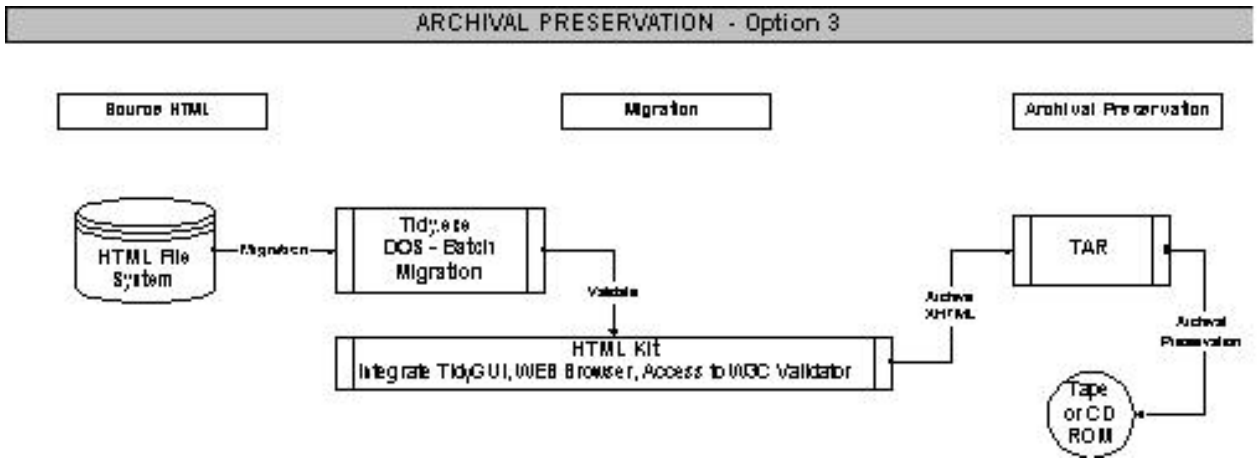
- A TAR snapshot will preserve the current Web site.
- A user friendly Windows (non-DOS) interface.
- All Web pages are converted individually with full visual control throughout the migration so corrections can be made "on the fly."
- The total execution time for migration of 10,000 HTML pages would be approximately 53 minutes.

Disadvantages

- Migration is done one HTML page at a time so that operator intervention is required to complete migration, which is estimated to be approximately 593 hours or 15 weeks.
- Validation of 10,000 converted XHTML pages will take approximately 520 hours or 13 weeks.

Implementation Option 3

This implementation option calls for the combination of Tide.exe (DOS batch mode) to convert HTML pages to XHTML and the use of the HTML-Kit to access the W3C Validation Service to ensure that the XHTML pages comply with the W3C standard.



Advantages

- A TAR snapshot will preserve the current Web site.
- A batch migration can be done with little human intervention during the process.
- A batch migration of 10,000 HTML pages will take approximately five hours.
- Windows interface supports a rapid visual check of 10,000 converted XHTML files and integrated access to the on-line W3C Validation Service.
- Validation of 10,000 XHTML pages will take approximately 437 hours.

Disadvantages

- The migration requires certain technical familiarity with DOS commands.
- Visual checking through the HTML Kit WEB browser can be time consuming.

5.2 Recommendations

1. Make a TAR copy of HTML pages before running a migration to XHTML,
2. Use the Tidy Utility DOS.exe to run batch processing to clean up and migrate HTML pages to XHTML pages,
3. Conduct a visual inspection of selected pages to confirm the accuracy of presentation,
4. Use the integrated validation service functionality in HTML-Kit to obtain certification that the converted XHTML pages comply with the W3C XHTML standard,
5. Ensure that validation certificates are retained as part of the metadata for the XHTML pages,
6. Select a magnetic storage medium such as DLT to store converted XHTML pages, and
7. Encapsulate validated XHTML pages with TAR and write to magnetic storage media.

Contact us at osiaref@osia.si.edu

Smithsonian Institution

Revised: November 15, 2002