

John A. Blythe. Digital Dixie: Processing Born Digital Materials in the Southern Historical Collection. A Master's paper for the M.S. in Information Science degree. 75 pages. July 2009. Advisor: Katherine M. Wisser.

Archives are increasingly accessioning digital materials as part of collections of personal papers. Proper preservation of these digital items requires archivists to add several new steps to their processing workflow. This paper discusses the steps developed to remove digital files from the media on which they are housed in the Southern Historical Collection of the University of North Carolina at Chapel Hill's Wilson Library. The paper is divided into three major sections. The first section examines literature related to digital archaeology, computer forensics and digital preservation. The second section describes the Southern Historical Collection, its technological environment and the process of developing a workflow. The paper concludes with a discussion of lessons learned from the project, unresolved issues and potential solutions.

Headings:

Data recovery

Digital preservation

Electronic archives

Electronic records – Conservation and restoration

Personal papers

Preservation of library materials -- Automation

DIGITAL DIXIE: PROCESSING BORN DIGITAL MATERIALS IN THE
SOUTHERN HISTORICAL COLLECTION

by
John A. Blythe

A Master's paper submitted to the faculty
of the School of Information and Library Science
of the University of North Carolina at Chapel Hill
in partial fulfillment of the requirements
for the degree of Master of Science in
Information Science.

Chapel Hill, North Carolina

July 2009

Approved by

Katherine M. Wisser

INTRODUCTION

For the past two years the British newspaper the *Guardian* has run in its Saturday edition a short feature titled “Writers’ Rooms” (<http://www.guardian.co.uk/books/series/writersrooms>). Each week there is a photo of a British author’s workspace along with a few paragraphs in which the writer describes the significance of items in the photo. The writing spaces include comfy chairs and couches, family photos, maps, crammed bookcases, seashells, toys, piles of paper, and, with few exceptions, a computer. This 20th century invention has become as important to the writer as the pen and paper, whose origins are much older. Novels, poems, plays and correspondence are birthed (and sometimes killed) within the confines of desktop PCs or laptops. Drafts may never make it onto paper. Instead, they sit on hard drives or removable media as MS Word or WordPerfect files fighting for space among Quicken financial reports, Outlook emails and iTunes downloads. The computer is the contemporary writer’s notebook but it’s also her file cabinet, checkbook, calendar and music collection. And the same holds true for the composer, the politician, the banker and the photographer. Each has found the computer a tool for work and a toy for leisure. As such, the computers, their files and the associated media are a rich source of information for those seeking to delve into the lives of the famous, the not-so-famous and the infamous — just as valuable as the notebooks, letters and newspaper clippings that scholars have long used to better understand our cultural history.

These digital artifacts of our personal lives go by many names — personal papers (Henry, 1998, p. 315), personal digital archives (*Paradigm workbook*, 2007, p. 1), personal digital collections (Beagrie, 2005), digital manuscripts (John, 2008, ¶4), e-manuscripts (John, 2008, ¶1) and, sometimes, even electronic records¹. Using these terms and others, archivists, librarians and other information professionals have noted the importance and urgency of collecting these materials and preserving them for future generations (Beagrie, 2005; Cunningham, 1999; Hyry & Onuf, 1997; Lukesh, 1999; Marshall et al., 2006; *Preserving Our Digital Heritage*, 2002; Task Force on Archiving of Digital Information, 1996). As one assemblage of concerned information professionals has suggested, “If we are effectively to preserve for future generations the portion of this rapidly expanding corpus of information in digital form that represents our cultural record, we need to understand the costs of doing so and we need to commit ourselves technically, legally, economically and organizationally to the full dimensions of the task” (Task Force on Archiving of Digital Information, 1996, pp. 3-4).

Admittedly the task described above is Herculean and one for which there is no single solution. Instead answers have, and will, come from diverse fields, including archives, library and computer science and information technology. Certain approaches may work in one setting, but not another — a result of different decisions about technology, staffing and material collected. This paper addresses the solution to a specific problem in a single environment. It discusses the workflow developed to remove digital

¹ Although electronic records generally refers to those records related to organizations and business, neither the SAA glossary, nor the glossaries of InterPARES 1 and 2 rule out the term’s use to describe the digital versions of items we generally call manuscripts or personal papers. In fact Henry (1998) argues that the term *electronic record* need not exclude personal papers.

files from the media on which they are housed in the Southern Historical Collection of the University of North Carolina at Chapel Hill's Wilson Library. The paper is divided into three major sections. The first section examines literature related to digital archaeology, computer forensics and digital preservation. The second section describes the Southern Historical Collection, its technological environment and the process of developing a workflow. The paper concludes with a discussion of lessons learned from the project, unresolved issues and potential solutions.

LITERATURE REVIEW

Data capture

The first steps in the processing and preservation of digital manuscripts have been labeled *digital capture* (John, 2008), *data capture* (Carrier, 2005; Farmer & Venema, 2005; *Good Practice Guide*), *data recovery* (Ross & Gow, 1999) or *digital archaeology* (Arms, 2000, Chapter 13, ¶35; Paradigm project, 2007, p. 242, Ross & Gow, 1999). The terms are mostly synonymous. The term *data capture* is often used in the field of computer forensics and is closely tied to the legal process of seizing evidence — computer data, in this case. Because the results of their work will likely be used in legal proceedings, experts in computer forensics must use techniques that allow them to remove data from computers and other digital media without making any changes to it (Carrier, 2005; Farmer & Venema, 2005; *Good Practice Guide*). In the parlance of archival science, they must be able to guarantee the data's *authenticity*, which the Society of American Archivists defines as "the quality of being genuine, not a counterfeit, and

free from tampering" (Pearce-Moses, 2005). *Digital capture* is a variation of *data capture* and refers to the movement of data to "modern, fresh and secure media" (John, 2008).

Digital archaeology is a term whose usage is growing among archivists and information professionals (Arms, 2000, Chapter 13, ¶35; Paradigm project, 2007, p. 242, Ross & Gow, 1999). It refers to the retrieval of "data from obsolete software or hardware environments, and obsolete or damaged media, such as punch cards, 8" floppy disks and the wealth of other removable media which have been used since the earliest days of computing" (Paradigm project, 2007, p. 242). Similarly *data recovery* refers to the process of restoring data from damaged media (Forensics Wiki, *data recovery*). In essence *digital archaeology* and *data recovery* are part of the *data capture* process, describing the extra efforts that may be required to seize data from certain media. In this paper *data capture*, *digital capture*, *data recovery* and *digital archaeology* are used interchangeably to refer to the transfer of content from its original medium to a more, stable preservation environment.

Those seeking to recover data may face a number of obstacles (Ross and Gow, 1999, Executive summary, pp. iv-v). The issues they may confront include:

1. Media degradation,
2. Loss of functionality of access devices,
3. Changes in hardware or operating systems that make manipulation of data difficult,
4. Changes in software applications or in video display technology that make presentation of data difficult,

5. Poor documentation of the file creation process (examples are failure to note the key used to encrypt files or to document the compression algorithm applied to data before it was written to a medium).

The problems listed above are ones that may delay processing of born-digital items. But, as Ross & Gow detail in their report, frequently there are hardware or software methods by which to overcome these hurdles. The specific mechanisms they suggest are beyond the purview of this paper. However the underlying principle of their report — that some digital items may require special attention before their data can be transferred — is one that cannot be overemphasized.

Archivists at the University of Texas' Harry Ransom Center have broken the digital archaeology process into six steps (Stollar & Kiehne, 2006, p. 2). Their breakdown provides helpful guidance on the tasks that must be accomplished by a workflow for processing born-digital items. The steps are as follows:

1. Receive and identify physical media.
2. Create a cataloging system for the physical media.
3. Copy files from the physical media and record metadata.
4. Perform initial file processing (virus checking and file recovery).
5. Create an item-level listing of all recovered files.
6. Create working copies of all files and protect the originals.

British archivist Jeremy John has likewise developed a workflow for processing born-digital items. John suggests that digital curators "adopt and modify existing technologies for new purposes rather than necessarily designing from scratch" (John, 2008, ¶12). Consequently John's workflow is modeled after the methods of computer

forensics experts, who: (a) acquire the evidence without altering or damaging the original, (b) establish and demonstrate that the examined evidence is the same as that which was originally obtained, and (c) analyze the evidence in an accountable and repeatable fashion (John, 2008, ¶14). Specifically, John writes, the "capture workflow" should include: (a) an audit trail; (b) write-protection; (c) forensic imaging, with hash values created for disk and files; (d) examination and consideration by curators (and originators), with filtering and searching; (e) export and replication of files; (f) file conversion for inter-operability and; (g) indexing and metadata extraction and compilation (John, 2008, ¶ 25 and ¶ 31).

There is much in common between the seven steps of John's digital capture workflow and the six steps of the Ransom Center team's digital archaeology process. Both emphasize documentation and the importance of automatically extracting and retaining metadata from the digital items. Both also stress the need for virus checking. But while John urges the creation of a forensic image of the digital item (also known as creating a bitstream copy), the Ransom Center archivists do not include such a practice in their workflow. These topics will be explored in further detail below.

Creating bitstream copies

In computer forensics one of the first steps in any investigation is the copying of a hard drive or other digital media bit by bit (APCO; Carrier, 2005; Farmer & Venema, 2005; John, 2008). As Carrier writes, "the rule of thumb is to acquire data at the lowest level that we think there will be evidence" (2005, Chapter 3, ¶ 5) and that lowest level is the sequence of bits known as a bitstream. It is the computer operating system and software that converts a bitstream to meaningful information. By applying software-

based and hardware-based forensic analysis to the *bitstream copy* rather than the original hard drive, investigators are able to avoid irreversible damage to the source medium. In so doing, they adhere to one of the central principles of computer-based criminal investigations: "No action taken by law enforcement agencies or their agents should change data held on a computer or storage media which may subsequently be relied upon in court" (APCO, p.4).

While those processing personal digital collections may not have criminal prosecution as their ultimate aim, they are concerned about maintaining the authenticity of the items with which they are working. As the Paradigm researchers write, "The preservation strategy of a digital archive should include provisions for ensuring that unaltered bitstreams are preserved intact over time so that the authenticity of digital objects is not compromised" (Paradigm project, 2007, p. 223). Put simply, the creation of a bitstream copy allows an archive to guarantee the public that it has an exact copy of the digital item that it accessioned.

Of course the creation of a bitstream copy also ensures that an archive has one copy of a digital item that "has not been subject to data loss or corruption induced by preservation actions" (Paradigm project, 2007, p. 223). No doubt as archives expand their collections of e-manuscripts, they will accession materials created with applications that have become obsolete or are rare. To provide access to such digital items, processors will need to convert them to different formats. The conversion process may subject the digital items to data loss or corruption. But, by creating and maintaining a bitstream copy, the archive can always revert to the digital item in its original state (Paradigm project, 2007, p. 223).

Finally, the creation of a bitstream copy may allow archivists to expand their collection of an individual's digital papers beyond simply her computer hard drive. As individuals create and store increasing amounts of their personal digital papers on the Internet (i.e. Google Mail, Google Docs, Facebook, web calendars, etc), archivists may want to accession items from those environments. But mapping an individual's "Internet footprint" (Garfinkel, 2009, p. 5) can be difficult. By using forensic analysis techniques, including the creation of a bitstream copy, archivists can begin to discover the personal papers that exist "in the cloud" (Garfinkel, 2009, p.1). Garfinkel (p. 5) suggests that an individual's hard drive stores traces of his Internet activity in several places, including:

1. Web browsers, which maintain bookmarks and caches of web pages.
2. Email messages, which may include links, notifications, and password reset instructions.
3. Address books, which may contain URLs and user names and passwords.
4. Calendars, which occasionally include URLs
5. A computer's logfiles.
6. Word processing documents.

Garfinkel writes that the clues listed above can sometimes be discovered by scanning the bitstream copy of a hard drive or other digital media with a forensic feature extractor. Such a tool can produce a report of all email addresses and URLs found on a hard drive. But Garfinkel acknowledges that the technique doesn't always work. "The originator may have explicitly attempted to hide [account names, aliases and pseudonyms], or may have accessed them exclusively from another machine, or [the account names, aliases and

pseudonyms] may have been used so long ago that references to the accounts have been overwritten" (p.5).

There are numerous methods for creating a bitstream copy of a hard drive or other digital storage medium. Most computer forensics software packages include bitstream copying (also referred to as "forensic imaging") among their tools, but such software is expensive and may be cost-prohibitive for many archives.² However, there are two methods frequently used by computer forensics experts that are available as part of most Unix operating systems. Forensic examiners have long relied on the "dd" command in UNIX, which can create a bit for bit copy of all information on a disk. This includes data both inside and outside the file structure (Farmer & Venema, 2005, chap. 4, ¶ 9; John, 2008, ¶ 29). The UNIX command "dcfldd" is another method by which to create a bitstream copy. But this command includes the added feature of creating a hash value (sometimes known as a checksum) for the bitstream copy (Carrier, 2005, A case study using dd ¶ 24). The importance of checksums will be discussed later in this paper.

Copying Individual Files

While the creation of a bitstream image is certainly one important means of capturing a digital item and its files, it is not the only method processors should employ.

² Some of the more well-known forensic software packages include:

Encase Forensic (http://www.guidancesoftware.com/products/ef_index.asp)

Forensic Toolkit (<http://www.accessdata.com/forensictoolkit.html>)

CD/DVD Inspector (http://www.infinadyne.com/cddvd_inspector.html)

Helix 3 (<http://www.e-fense.com/register-overview.php>)

The Sleuth Kit (<http://www.sleuthkit.org/sleuthkit/>)

SleuthKit is shareware and available at no or low cost. Encase Forensic ranges from \$2,850 to \$3,600. Forensic Toolkit costs \$2,995

(http://www.scmagazineus.com/Forensic_Toolkit_v20/Review/2380/). CD/DVD Inspector lists its price as \$549 and Helix 3 advertises for \$14.95 a month.

They should also copy the files in their original formats. A bitstream by itself is nothing more than a long string of ones and zeros. For that string to represent information, it must be parsed by a computer's operating system and software (Paradigm project, 2007, p. 228; White, 2008, part 3;). Confronted with a long string of bits, the operating system must try to determine the startpoint and endpoint of each file. This process takes time and may tax the computer's operating system. There is also no guarantee that the operating system has guessed correctly. By storing a digital item as a file (or series of files), a processor has, in essence, taken some of the guesswork out of file identification. Additionally, the creation of an exact copy of the file ensures that a user can study the file exactly as it was when accessioned by the archive (John, 2008, "Consolidation of the capture workflow," ¶ 4). Of course, the key determinant of whether a user can examine the file in its original form is the existence of the version of the application that created it. In some cases it may be necessary to migrate the file to a more recent version of the application. But issues related to migration are beyond the focus of this paper.

There is no one prescribed method for copying a digital item. The process can be as simple as using the copy and paste function found in many computer operating systems. Researchers at the Harry Ransom Center relied on such a method (Stollar & Kiehne, p. 3). By contrast, processors at Duke University use a tool developed by their school's electronic records archivist specifically to migrate files. The Data Accessioner automatically copies the directory structure and files on a digital medium. The tool also performs error checking by creating checksums before and after a file is migrated.³

³ See <http://library.duke.edu/uarchives/about/tools/data-accessioner.html>

Finally, various forensic software applications also copy files, in many cases providing additional functions that allow a user to analyze a file's structure and origins.⁴

Depending on the methods that processors use to copy a file from its source medium to a preservation environment, the file's MAC time may be changed. MAC time is automatically created metadata logging the date and time when a file was last modified (M), accessed (A), and changed or created (C). *Modified* (also known as *mtime*) refers to date and time when the content of the file most recently changed. *Accessed* (*atime*) identifies when a file was most recently opened by a person or software. File systems use *ctime* differently. On Unix systems, *ctime* identifies when a file's metadata was last changed (i.e. a change in permissions, a change of owner or even a change in other MAC time metadata). Windows file systems treat *ctime* as *creation time*. The metadata refers to the time a file is created. In an archival setting MAC times could provide processors with a rough timeline of when files were created. But archivists should be wary of placing too much value in metadata gathered from MAC times. Windows systems generate a new *ctime* every time a file is copied, leaving processors unsure as to whether *date created* actually refers to the date the content of a file was created or the date of the copy (Kiehne, Spoliansky, & Stollar, 2005a, p. 6; Stollar & Kiehne, 2006, p. 3).

Virus Checks

As any savvy computer user knows, their systems could potentially fall victim to viruses and other forms of malware. The number of malicious code signatures created by one anti-virus software developer increased fourteen fold between 2005 and 2008, a clear indication of the rapid growth of viruses and malware (Symantec Corporation, 2009, p.

⁴ Please see websites listed in footnote 2.

56). Common sense suggests that some digital media accessioned by archives will arrive infected with viruses.⁵ This eventuality further reinforces the need to follow the established computer forensics practice of creating bitstream copies and scanning them for viruses before performing additional processes on them. Following these procedures reduces the risk of infecting the storage environments tasked with the long-term preservation of digital papers. Prior to virus checking, researchers at the Ransom Center suggest, processors should confirm that the software they are using is able to check for viruses contemporary at the time of the digital items creation (Kiehne, Spoliansky, Stollar, 2005a, "Perform initial file processing," ¶ 1). Most anti-virus software dictionaries are believed to be cumulative, but processors should verify this to be the case with their software.

Write Blocking

By creating a bitstream copy of a digital item and relying on the copy to perform preservation processes, archivists greatly reduce the possibility of damage to the source medium. But, of course, the creation of a bitstream copy is a process itself and during its creation the source medium is at risk for inadvertent alteration or damage. Computer forensics experts have addressed this concern through the use of write blockers. These devices, which are available as both hardware and software, work by allowing read commands to pass from a processor's computer to the digital medium, but by blocking write commands (Forensics wiki, http://www.forensicswiki.org/wiki/Write_Blockers).

John suggests the extension of write-blockers to the archival setting when dealing with

⁵ Christopher A. Lee at the University of North Carolina reports that the dataset he used for his research on the significant properties of e-mail attachments arrived infected with viruses. Five of 41 e-mail accounts from the 1997-2001 administration of North Carolina governor James B. Hunt Jr. included the Anna Kournikova virus (*Preserving attachments from an e-mail collection: the good, the bad, the ugly and the thought provoking*, 2008, slide 14).

digital media (2008, ¶ 27). Of course, by its nature some digital media is already write-protected. It is not possible to rewrite a CD-R or DVD-R. Neither John nor others have made a case for write-protection of such media.

Fixity Checks

Long-term preservation of digital media requires a means of periodically checking the integrity of files — guaranteeing that the content of a file has not changed from when it was originally created. Those involved in digital preservation are in agreement as to the importance of *fixity checks* (Kenney & Rieger, 2000, p. 143; Paradigm project, 2007, p. 28; Task force on archiving of digital information, p. 12; Lynch, 1996, p. 739), but there is little literature that explicitly states the best type of fixity check to use and the frequency for such checks. Fixity checks generally work in the same manner. Using any number of mathematical equations, a value is assigned to a file or bitstream. When there is a need to check that the file or bitstream is still the same, the mathematical equation is re-computed. The re-computed value should match the original value. Frequently fixity checks are referred to as checksums, but, as Novak (2006) suggests, checksums are but one form of fixity check. And, the fixity check often used for digital preservation, the MD5 hash algorithm (Novak, 2006, ¶ 8; Paradigm project, 2007, pp. 28-29), is in fact a message digest and not a checksum (Novak, 2006, ¶ 4).

Fixity checks are most important before and after a file is moved or copied. These processes can result in advertent damage to file. In relation to the accessioning of born-digital items, a fixity check should be carried out prior to the creation of a bitstream copy (APCO, n.d.; Farmer & Venema, 2005) and prior to the copying of individual files (Paradigm project, 2008, p. 152). A fixity check should be performed again when the

bitstream copy and individual files have been transferred to a preservation environment (Paradigm project, 2008, p. 152).

Fixity checks are also useful in determining whether an accession of born-digital items includes duplicate files. Theoretically each unique file should produce a unique checksum value. Consequently two files with the same values contain duplicate information (Lynch, 1996, p. 739). By comparing the checksums of each file on a digital medium (a task best accomplished with automated or scripted processes since MD5 strings are 32 characters of numbers and letters), a processor can easily determine whether two files are the same (Stollar & Kiehne, 2006, p. 3). Before disposing of seemingly identical files, processors may want to use other means to confirm their suspicions. Researchers at the Ransom Center checked the "date modified" and "date created" metadata of files, as well as the files' format and size before getting rid of duplicate files that they had accessioned (Kiehne, Spoliansky, Stollar, 2005a, "The Appraisal Process," ¶ 7).

Finally fixity checks can help an archive weed out digital items that are software or operating system files. Whether accessioning a hard drive or floppy disks, processors may come across such files (Peters, 2006, p. 26), and, because of copyright regulations or limited storage space, they may want to dispose of them. Processors can identify files they should not keep by automatically comparing the checksum values of accessioned files with known hash or checksum values for software and operating system files. Such values are stored in software libraries (John, 2008, Consolidation of Capture Workflow section, ¶ 3).

Metadata

According to the Paradigm workbook, "Archivists must create, manage and use preservation metadata in order to administer and maintain access to authentic digital archives, their context and provenance over the long-term" (2007, p. 73). But what is *preservation metadata* for digital items? Put simply, preservation metadata is "information that supports and documents the preservation process" (Preservation Metadata Implementation Strategies, Background, What is preservation metadata?). More specifically, the PREMIS working group writes, preservation metadata for digital items includes information about:

1. Provenance (Who has/had custody of the digital object?)
2. Authenticity (Is the digital object what it purports to be?)
3. Preservation activity (What has been done to preserve the digital object?)
4. Technical environments (What is needed to render and use the digital object?)
5. Rights management (What intellectual property rights must be observed?)

The result of the PREMIS working group's deliberations was a vast XML metadata schema, which provides a structure for documenting information related to the preservation of digital files. But the schema is complex and the working group did not recommend a minimal set of metadata that should be maintained for all files.

Researchers at the Ransom Center offer some direction for the types of preservation metadata that processors should keep during the digital capture process. They maintained spreadsheets while processing the digital files of hypertext author Michael Joyce (Stollar & Kiehne, 2006, p. 3; Kiehne, Spoliansky, and Stollar, 2005b;

Schmidt, 2007). The fields they included in their spreadsheets were: (a) file size, (b) kind of file (folder, text document, document, picture, application, font file, control panel, etc.), (c) creator, (d) creation date (e) format name (the application that created the file, if known), (f) comments by Michael Joyce (contextual information that Joyce provided to processors about the files), (g) file path, and (h) checksums (Kiehne, Spoliansky. and Stollar, 2005b; Schmidt, 2007). Using the broad categories suggested by the PREMIS definition described above, preservation metadata gathered and recorded during data capture of the Joyce files addressed questions of authenticity and the technical environment needed to read the files.

Software, including some from the field of computer forensics (John, 2008), can make the task of documenting metadata about digital files easier. The Ransom Center researchers relied on one shareware tool to extract much of the technical metadata and a second tool to generate checksums. Both software tools exported their results to a delimited format. But, according to the Ransom Center researchers, integrating the two delimited text files into a single file proved difficult because of differences in the way the two applications worked (Stollar & Kiehne, 2006, p. 3).

THE PROJECT

The Southern Historical Collection

This project set out to develop a capture workflow for digital items in the Southern Historical Collection (SHC). The SHC is one of five special collections housed at the University of North Carolina's Wilson Library. Officially established in 1930, the SHC holds documentation of Southern history and culture dating back to the late 18th century. The archive contains more than 4,600 individual collections, including 26 with

digital items. Digital materials are found in the collections of such luminaries as Pulitzer Prize-winning author Taylor Branch, writers Jill McCorkle and Elizabeth Spencer, bandleader Kay Kyser and composer Roger Hannay. Digital media include 102 compact discs, 81 3½” floppy disks, 114 5¼” floppy disks and 7 DVDs. Content on the digital media includes photographs, manuscripts, e-mails, databases, audio and video. These material types are represented by such file formats as jpeg, tiff, .doc, dBase IV and dBase V.

In recent years SHC staff have grown concerned about the stability of digital items in the collection and they've sought to transfer the content of the media to more stable storage environments. Because of limited finances they have wanted to develop a workflow that would make use of existing staff and technology. The SHC includes one full-time processor and from five to ten undergraduate and graduate student processors. Prior to the commencement of this project, processors mostly followed the same procedures for digital items as they did for paper materials. They assigned each digital item a number, noted its existence in the finding aid and then shelved it with like-items in the stacks. But unlike the steps for processing paper materials, archivists could not fully determine the condition of the item. They could assess the outward appearance of the medium, but they could not be certain of the content on the media. They were also not able to gauge whether the files it contained were corrupt or otherwise unusable.

Processing of paper and digital materials is performed in the SHC's technical services department. The space is equipped with four IBM/Lenovo workstations loaded with the Windows XP operating system and a suite of software, including web browsers, and applications for word processing, text editing and anti-virus protection. All

computers have CD or DVD drives. Some are also equipped with a 3½” floppy disk drive. The workstations are shared by student processors and are networked to several library-maintained servers. The University Library systems department also operates a digital archive (dark archive), which is designed as a protected storage environment for files that are not regularly accessed. Unfortunately during much of this project’s tenure, the digital archive was undergoing an upgrade and files could not be saved there. Instead, files were saved to one of the library’s servers, a process that was not considered as trustworthy for long-term preservation of data. Servers and workstations are maintained by the library’s systems staff, who are based in another library building about 200 yards from Wilson Library. Systems staff are available via telephone and appointment, but they do not regularly visit Wilson Library.

Design Considerations

The development of a workflow for processing born-digital items began in September 2008. From the outset several factors affected the design of the project. As stated in the previous section, the workflow needed to rely on existing staffing arrangements and technology. Because the workflow was being designed for use primarily by student processors, the procedures needed to be ones that individuals with varying levels of technical skills could execute. Similarly any new technology required for the project needed to be of a type that could be installed and maintained with minimal support from the library’s systems staff. While SHC administrators did not rule out the purchase of additional software, they strongly preferred the use of free applications or those already installed on workstations. Additionally, those applications needed to work

on the SHC's Windows-based workstations since there was no budget for the purchase of new hardware.

In addition to addressing the limits on staffing and new technology, the method developed for processing born-digital items needed to work for a host of different media types. Although the SHC's digital holdings currently include only CD-ROMs, DVDs and floppy disks, future donors will likely wish to transfer digital content on compact flash cards, thumb drives and hard drives. While the means of connecting each of these media types to SHC workstations may vary slightly, the steps for moving the data off the media and preparing it for ingest needed to be the same.

The workflow also needed to include methods to create bitstream copies, check for viruses, copy individual files, gather metadata and create file integrity checks. SHC staff hoped the software tools used to perform each of these tasks could be integrated, which, in turn, could increase automation.

Creating the Bitstream Copies

SHC staff considered several software applications for creation of bitstream copies. Their efforts were only moderately successful because the tool selected, RecordNow, is capable only of creating bitstream copies of CDs and cannot copy floppy disks. Additionally the tool does not copy all formats of CDs and is proprietary software.

SHC staff spent several weeks considering use of CDRDAO (<http://cdrdao.sourceforge.net>). The application is a Unix-based, freeware application that records data as one large block onto a CD-R, a method known as disk-at-once (DAO). A CD recorded with the disk-at-once method does not include track separations and, consequently, contains a single bitstream of data. While CDRDAO is primarily used for

burning CDs, it can also be used for extracting data from one CD and copying it to another disk.

The SHC began considering CDRDAO after reading about the National Library of Australia's Prometheus project and consulting with Prometheus staff. Prometheus (<http://www.nla.gov.au/pub/gateways/issues/96/story02.html>) is a semi-automated system developed to perform the same basic task as that which the SHC sought to carry out — the transfer of data from various physical media to a preservation storage environment. Prometheus staff reported their preference for CDRDAO because of its ability to make disk copies of many types of file systems (e-mail from Nicholas Del Pozo, November 9, 2008). It's highly likely that personal digital archives, like those held by the SHC, will include CDs containing a mix of file systems, including HFS and ISO 9660.

Unfortunately, CDRDAO requires the Unix operating system or the Win32 API to run. As mentioned earlier, SHC workstations are not loaded with the Unix operating system and SHC staff lacked the technical skill to install a Unix operating system on top of the Windows platform or set up CDRDAO within the Windows 32 API.

Instead, SHC staff decided to use RecordNow, CD/DVD-burning software that was pre-installed on the SHC workstations. RecordNow's options include the creation of a disc image. Operation of the software is relatively simple, but its use did present some problems. Documentation for the version of RecordNow found on the SHC workstations was not readily available in the SHC nor could it be found easily on the web.

Consequently, there was no way to be certain that the disc image created was a full capture of the entire bitstream of a CD. Secondly, RecordNow was not able to create disc images of CDs created in such file systems as HFS (commonly found on Macs) or ISO

9660 Rock Ridge (a Unix-based file system). To make disc images of CDs created with the aforementioned file formats, the SHC relied on Disk Utility, an application found within Mac OS X for disk-related functions. Because SHC technical services lacks a Mac computer, the SHC relied on a staff member's personal MacBook Pro laptop to create disc images of those CDs that could not be imaged using RecordNow. Such a method is hardly a long-term solution.

The combination of RecordNow and Disk Utility proved sufficient for creating bitstream copies of most CDs, but, as discussed previously, neither application can create copies of floppy disks. However, the Unix applications `dd` and `dcfldd` are designed for such a purpose. Thanks to a cooperative arrangement with the online library *ibiblio*, this SHC project had available for its use an offsite, Unix-based server. Staff experimented with use of `dcfldd`, an enhanced version of `dd` that includes creation of MD5 hashes, but difficulties with mounting SHC workstation drives to the Unix server kept the project from incorporating `dcfldd` into the capture workflow for floppy disks. Consequently, no bitstream copies were made of floppy disks in the SHC collection.

Virus Checks

As with methods for creating bitstream copies, the SHC considered several software packages for virus checking. Again, staff looked first to Prometheus for a possible solution. That project uses the anti-virus application ClamAV (<http://www.clamav.net>) to check source media for viruses (Elford et al., 2008, p. 10). Unfortunately ClamAV is designed to work in a Unix environment. So, after attempts to mount SHC workstation drives to the *ibiblio* Unix server failed, SHC staff considered other methods for virus checking.

Ultimately the SHC chose to use the anti-virus software installed on workstations in technical services. The workstations are equipped with Symantec Anti-Virus 2006, version 10. The application's virus definitions are automatically updated daily. In documentation of the processing of the Michael Joyce digital papers, researchers at the Ransom Center stressed the need for verifying that anti-virus software includes definitions for viruses that were contemporary at the time of digital papers' creation (Kiehne, Spoliansky, Stollar, 2005a, "Perform initial file processing," ¶ 1). SHC staff attempted to check this fact in technical documentation of the Symantec anti-virus software in use, they were unable to find an answer. However, several individuals with experience in data capture assured staff that properly maintained anti-virus software should contain definitions for old viruses.

Once the SHC decided to use Symantec Anti-Virus 2006, they needed to determine the point at which to include virus checking into the data capture workflow. While literature on the data capture process underscores the need for virus checks, it does not suggest when such a step should occur. Should the source media be checked for viruses and then a bitstream copy created? Or should a bitstream copy be created and then *the copy* checked for viruses? As noted elsewhere in this paper, standard procedure in computer forensics is to create a bitstream copy of the source media and then perform such tests, as virus checks, on the copy. Such a process reduces the chances of accidental damage to the source medium. However, performing a virus check on the bitstream copy would have required first that the copy be mounted on the workstation — a sometimes cumbersome process and one that could possibly place infected data onto the workstation or in the preservation storage environment. Consequently, the recommended computer

forensics procedure was modified. A bitstream copy was created and then a virus check performed on the source medium (as opposed to the bitstream copy). This method ensured that a complete backup of the source medium existed in case the original was damaged during the virus check.

If software had revealed viruses on the source medium, the affected files would have been noted in documentation of the data capture process. During the course of this project, no viruses were detected. Concerns about the possible spread of viruses onto library computers and servers could also be allayed by never opening the files copied onto the library servers. Because arrangement and description was not a part of this process, there was no need to examine the contents of individual files. Of course, in the future files may need to be opened. At that time, the virus checking process may need to be re-evaluated.

Copying Individual Files

In the early stages of the SHC project, staff learned of the Data Accessioner tool (<http://library.duke.edu/uarchives/about/tools/data-accessioner.html>). Developed by Duke University's electronic records archivist, the Java-based application migrates data off disks and into a preservation storage environment. Data Accessioner also integrates tools for collecting metadata about the migrated files and creates a checksum of each. Duke's electronic records archivist freely shared the application with the SHC and the tool was installed on the four, shared workstations in technical services.

The SHC digital capture workflow calls for use of the Data Accessioner after a bitstream copy of the source medium has been created and the medium checked for viruses. A user follows several steps to use the application (each step is laid out in clear

detail in Appendix A). She selects the file identification and validation tools she wishes to use, choosing DROID (<http://droid.sourceforge.net/wiki/index.php/Introduction>), JHOVE (<http://hul.harvard.edu/jhove/>) or both. And she chooses the format in which she would like metadata to appear. Her options include no metadata, a default manager, which lists the full output of JHOVE and DROID, or the Duke Premis structure, which includes only the file identification and validation portions of the JHOVE and DROID processes. JHOVE and DROID complement each other. Both identify file formats, but DROID is able to recognize more formats. JHOVE can determine whether a file meets all the specifications of its format, i.e. whether it is well-formed and valid. Because each tool offers a feature not found in the other, SHC staff chose to use both the JHOVE and DROID features of the Data Accessioner. Staff chose Duke Premis as the output structure for metadata.

The Data Accessioner provides an archival processor with the option of migrating all files on a digital medium or deselecting certain files that he does not want to migrate. When files and folders are migrated to the preservation environment, they are arranged in the same directory structure as exists on the source medium.

The tool provides a great service to the SHC with its ability to create MD5 hashes. The Data Accessioner creates a MD5 of each individual file on the source medium. After the file has been migrated to the preservation environment, the tool creates an MD5 of the migrated file and then compares the second hash value with the hash value of the original file. If there is a difference between the two values, the Data Accessioner creates an error message for the processor. The Data Accessioner's ability to create hashes is not only advantageous for error checking, but also useful for preservation. By automatically

generating an MD5 of each file copy, the tool provides the SHC with a means of checking file integrity over time. The MD5 created by the Data Accessioner serves as the checksum to which all future checksums can be compared.

Unfortunately there is a downside to the Data Accessioner's error-checking capability. If the tool detects an error with a file during the copy process, it immediately stops copying. Consequently, those files on the disk that fall below the error-ridden file in the directory are not copied. The only way to migrate such files is to begin the copying process again, de-selecting in the Data Accessioner the file that has errors.

A second problem encountered during use of the Data Accessioner is that there is little documentation for the tool. When errors occurred the Data Accessioner produced an error message, but the meaning of such messages was not always clear. SHC staff had to contact Duke's electronic records archivist to understand the significance of such messages as "Insufficient system resources exist to complete requested service" and "Error occurred while migrating Henry7thcontinuous.mp3. Unable to process source file Henry7thcontinuois.mp3 for MD5."

Checksums

As noted previously, the Data Accessioner produced checksums for individual files. But the SHC still needed a tool to produce checksums of the bitstream copies. Staff chose the MD5 Hash Generator (<http://drnaylor.co.uk/software/md5/>) for this purpose. It is freeware and fairly easy to use. The tool can generate a file containing the MD5 hash value in the directory right next to the file for which the hash was created. Unfortunately the MD5 Hash Generator can only generate a checksum of the bitstream copy. Because the computer operating system does not view the source media as a single file, MD5

Hash Generator is unable to create a checksum of the media itself. Without a checksum of the source there is no way to be certain that the bitstream copy is, in fact, a true copy.

Write blocking

Although write blocking is often used in data capture performed by computer forensics experts, no software or hardware for such a purpose currently exists in SHC technical services. SHC staff were not greatly concerned during data capture from CDs because the media processed during this project were fixed. They had already been burned and finalized. Consequently their data could not be overwritten. Such was not the case for the floppy disks included in this project. As noted elsewhere, the SHC lacked a method for creating bitstream copies of floppy disks — a fact that left staff without a backup copy to which to resort if the original media was accidentally altered. Staff took some comfort in knowing that the Data Accessioner does not open individual file as it migrates them to a preservation environment, reducing the chances of accidental alteration of source files. Nevertheless, processors worked with heightened awareness during data migration from floppy disks.

Whether from CDs or floppy disks, files were made read-only once they had been migrated. Although the Data Accessioner's creator says a future version may include a setting for permissions control, the current version of the software does not automatically make copied files read-only. SHC staff write-protected captured data by working within the Windows file system to change the permission to "read-only" on the top-level directory for each digital item migrated. Once "read-only" was selected for the top-level directory, the Windows operating system provided the option of applying the permission to all sub-level folders and files.

Metadata

The SHC project relied on two tools to document metadata, the Data Accessioner and an Excel spreadsheet. As described elsewhere, the Data Accessioner automatically records basic metadata about the data capture process and stores it in an XML document. The tool records the date of the capture, the name of the processor who performed the capture, the collection from which the digital media originated and the “last modified” date for each file. The software also performs file format identification, assesses a file’s well-formedness and creates checksums. Finally, the Data Accessioner documents the directory structure of the digital media, providing processors with a means of noting the relationship between individual files found on the source media. The XML document created by the Data Accessioner is stored in a directory along with the migrated files (see Appendix B for a sample XML-document created by the Data Accessioner).

As suggested by PREMIS, there is other metadata that is important for the longterm preservation and use of digital data (Preservation Metadata Implementation Strategies, Background). The SHC project did not seek to collect metadata about provenance or use restrictions since that information is found in the finding aids to the digital items’ respective collections. However, staff did deem it important to record details about the data capture process for each digital item since that information was not fully recorded by the Data Accessioner and was not included in finding aids. The metadata was manually recorded in a spreadsheet. SHC staff looked to other data capture projects for models of the types of metadata tracked in spreadsheets. The only examples found were spreadsheets used by researchers at the Ransom Center during the processing

of the digital papers of hypertext author Michael Joyce, but these documents didn't track the metadata deemed most important for the SHC project.

The spreadsheet created for the SHC project includes fields for the digital item identification number, a number assigned by processors in technical services; the name of the collection from which the digital item originates; the type of medium; the accession date; the name of the XML file created by the Data Accessioner and the checksum of the bitstream copy of the source medium. The spreadsheet also includes a field in which a processor records whether a virus check was run and whether the file copies have been made "read-only." Finally, there is a notes field in which the processor can document problems that may have occurred during data capture or information about the file system of the source medium. Originally the spreadsheet also included a field in which a processor could record the number of files migrated from each source medium. But staff stopped using this field after concluding that this information was sufficiently captured by the Data Accessioner.

The Workflow

Determining the appropriate tools to use for data capture was but one part of the development of a workflow for the Southern Historical Collection. Staff also had to decide the order in which the various tools would be used. Because the applications weren't developed to work in concert with each other, there was no prescribed sequence for their use. As previously discussed, proper investigative techniques in computer forensics suggest that a digital item be copied prior to any examination of its content. And the copy, rather than the original, is the medium that should be examined. Additionally, as a part of the copying process, checksums should be created of the source

medium and its copy. A comparison of the checksum values of each will determine if the copy is an exact match of the original. Because the SHC project had no way to create a checksum of an entire compact disk or floppy disk, this step had to be left out. Only a checksum of the bitstream copy was created.

Staff also had to determine the point during the workflow at which they should run a virus check. They had to decide whether the virus check should occur before a bitstream copy was created or after. By running a virus check prior to creating a bitstream copy, SHC staff could reduce the risk of a virus infecting the SHC workstations or library servers. But, at the same time, they increased the chances of damaging the source medium before a copy of it existed. Lacking clear direction on which process should precede the other, staff settled on creating a bitstream copy and then running a virus check on the source medium. Because the data capture process did not include opening files, the risk of spreading a virus seemed low.

In the end, the SHC settled on the workflow described in a general sense below (See Appendix A for full details):

1. Create a bitstream copy (for CDs only).
2. Generate an MD5 hash value for the bitstream copy.
3. Run a virus check on the source medium.
4. Migrate individual files using the Data Accessioner.
5. Make all copied files and bitstreams read-only.
6. Record metadata in spreadsheet.

CONCLUSIONS

When this project concluded in May 2009, SHC staff had migrated 51 CDs, 20 3½” floppy disks, 3 5 ½” floppy disks and 4 DVDs to the library’s digital archive. Migrated media ranged across 17 of the archive’s 26 collections with digital items. While media remain for future migration, the nine-month project saw the Southern Historical Collection make its first foray into digital archaeology. Those initial steps allowed SHC staff to become more familiar with the processes required for migration and preservation of digital items. But many questions remained unanswered and there are still many decisions to be made.

During the past year a data capture workflow was developed and tested by staff members. This project produced two documents for use in the Southern Historical Collection, a step-by-step guide for data capture (Appendix A) and a more general document outlining the issues related to data capture (Appendix C). This second document was circulated to special collections supervisors and served as the basis for several discussions about future plans for data capture among all special collections.

The basic actions required to migrate and preserve digital items are clear. Staff should:

1. Create a bitstream copy of the item,
2. Check the digital item for viruses,
3. Run integrity checks on the item and its individual files before and after migration,
4. Copy the item’s directory structure and its individual files,
5. Write-block all migrated material,

6. Create and maintain metadata that will allow future archivists and users to understand the migrated material and help them preserve it.

Absent from this list are specific methods for carrying out each step. The field of digital curation is evolving and, consequently, the tools used for data capture, migration and preservation will change. The choice of tools used by staffers in such archives as the Southern Historical Collection will also vary based on the media, file types and condition of the digital materials accessioned. The Southern Historical Collection must be open to this evolution.

The Data Accessioner currently provides the SHC with a useful means of carrying out several of the steps described above — the ability to automatically run integrity checks, copy files and capture basic metadata. But the tool does not create bitstream copies, check for viruses or write-block migrated files. Additionally the Data Accessioner's ability to reliably determine file types is quite limited. Automatic file and format identification is important for long-term preservation because digital curators will need such information to determine which files hold the greatest need for format migration. Finally, the Data Accessioner has great difficulty migrating audio and video files if the CDs and DVDs on which they reside are formatted as audio or video CDs rather than data CDs. The Data Accessioner's creator maintains that his tool was not designed to migrate such files and that providing such a feature would require additional programming, if even possible. The Data Accessioner's limitation kept SHC staff from attempting to migrate music CDs featuring the works of a composer whose papers are held by the archive. The restriction may also account for difficulties experienced by staff when using the tools to migrate audio and video files from data CDs.

As stated elsewhere, the Data Accessioner proved attractive to the SHC for several reasons. For one, the tool integrated several steps in the data capture process. Secondly, the Data Accessioner did not need additional programming to function in the SHC and, consequently, did not require the support of library systems staff. Finally, the application was free. The first two points demand further discussion. With the exception of the Data Accessioner, the tools used for data capture in the SHC did not integrate multiple steps of the workflow or function automatically. Staff had to perform each virus check separately, create bitstream copies and their accompanying checksums individually and write-block folders manually. The relatively small volume of digital material with which staff was dealing reduced the effect of such inefficiencies. But as the Southern Historical Collection accessions more digital items, the lack of integration and automation could increase both the time required for data capture as well as the possibility of human error.

In the waning days of this project discussions began with library systems staff on tools and methods for integration of data capture steps. The impetus for such talks was the University Library's development of a digital repository for the University of North Carolina at Chapel Hill. The system is expected to serve the entire campus and preservation of multiple types of digital files is one of its primary functions. While the digital files for the SHC project were migrated to the library's digital archive, this method was considered a short-term approach. Plans call for SHC staff to migrate digital material to the digital repository once the system is deployed. Consequently the repository's developers, library systems staff, were interested in the SHC's data capture needs. Although discussions are continuing, programmers have suggested they might be able to

develop scripts to mimic — and integrate — the functions of such tools as the Data Accessioner, RecordNow and Symantec Anti-Virus.

Discussions with the repository's developers continued a long-running debate among SHC staffers on the utility of creating and preserving bitstream copies. These individuals were concerned about the tools and time required to create such copies. As noted elsewhere, SHC staff had to use two different applications to create bitstream copies of the CDs included in this project. And staff lacked a tool for bit-by-bit copying of floppy disks. Those questioning the utility of the bitstream copy step also were concerned about the storage space required to preserve such files. Because a bitstream image is a bit-for-bit copy of the digital media, such files can be large. For example, a bitstream copy of a 4.7 GB DVD will produce a 4.7 GB file, even if the DVD contained no more than 1GB of data. Over the course of this project, those SHC staffers with reservations grew more comfortable with including a bitstream copying step. They appeared to be won over by the argument that creation and preservation — at least for the short term — of bitstream copies is one of the primary methods for guaranteeing that migrated files are authentic copies of those found on the source media. Once they entered the conversation, repository developers appeared to further reduce concerns by suggesting that they could easily integrate a tool for creation of bitstream copies and that the copies' file size would likely prove little concern in the early days of the proposed 40 TB digital repository. Admittedly storage space may become an issue at a later date and SHC staff and digital repository developers agreed to address the issue again when, and if, that time occurs. Those who consider the authenticity of digital files a primary issue

will likely argue that bitstream copies should be maintained forever. But as digital curation evolves, other methods for guaranteeing authenticity may develop.

Of course, the creation of a bitstream copy is not possible if no device is available to read the digital medium. The project described in this paper dealt with a select few media types — CDs, DVDs and 3½” floppy disks. The SHC has yet to resolve such questions as how to read other digital media and whether to limit the media types it collects.

As discussed previously, this project relied on the SHC’s existing computers. Those workstations include drives for reading CDs, DVDs and 3 ½” floppy disks. The SHC does not have a 5 ¼” drive. Because more than half of the archive’s current digital holdings are 5¼” floppy disks and it’s possible that future accessions could include such disks, staff wanted to experiment with data capture from this type of media. During one of the last days of the project, a staff member took about 30 5¼” floppy disks from the SHC to the Odum Institute for Research in Social Science. The Odum Institute maintains a large archive of digital social science data and, for that reason, has a variety of computers and disk drives. Its collection includes a 5¼” drive. The SHC staff member planned to copy the data from the 5¼” floppy disks to CDs and then, back at the SHC, use the Data Accessioner for other parts of the data capture process. Unfortunately, only two of the 30 floppy disks were readable. The SHC staffer and the Odum Institute employee with whom he worked were unable to determine whether the disks had become corrupt or whether there was a problem with the 5¼” drive. After consultation with SHC supervisors, staff decided to leave the disks unmigrated. The brief experiment with migrating 5 ¼” disks was not thorough enough for staff to draw any reliable conclusions

about data capture from such a disk type. Nevertheless, the experiment revived earlier discussions within the SHC about whether the archive should limit the types of digital media that it collects.

The issue first surfaced in fall 2008 as SHC staff was reading about the National Library of Australia's Prometheus project. That project developed an automated workflow for data capture. In the course of their work, National Library staff created *mini-jukeboxes* (Elford et al., 2008, pp. 10-11). These devices resemble small portable radio/tape/CD players, commonly known as boomboxes. However, instead of audio equipment, they contain drives for reading a variety of media. For instance, a mini-jukebox might contain a 3 1/2" floppy drive, a 5 1/4" floppy drive, a compact flash card reader and a CD-drive. Another might contain two CD-drives, a flash card reader and a 3 1/2" drive. The National Library found such devices useful because they were portable and could be attached to almost any workstation in the National Library. Additionally the construction of the jukeboxes allowed for easy configuration of drives most suitable for the processing project.

After reading about the mini-jukeboxes, SHC staff considered building one or more mini-jukeboxes for use in the archive. After several discussions staff decided against creating a mini-jukebox for the SHC data capture project, primarily because they would need the assistance of library system staff to do so. However, staff did not rule out the possibility of building a mini-jukebox for use at a later date. In fact, the topic has resurfaced during recent conversations with library systems staff and digital repository developers. A systems supervisor, who was aware of the Prometheus project, suggested that his staff might be able to build a mini-jukebox. A decision has yet to be made.

While some SHC staff advocate construction and use of a mini-jukebox, others are concerned about the implications of such a move. Those who would like a mini-jukebox believe that such a device would allow curators greater flexibility in the types of digital material they collect and would provide processors with a more direct means of capturing data from disparate media. That is, archivists would have at hand the various drives they need for reading the media. Those less supportive of a mini-jukebox worry that reliance on such a device will begin a never-ending cycle of purchasing new hardware to read the ever-changing digital media that may be accessioned by the SHC. These individuals argue that curators can, and should, decline to accession certain media types from donors. They readily admit that the archival value of certain digital items may merit their collection even if the archive does not have the appropriate devices for reading them. But, they suggest, in those cases the SHC should contract with an outside vendor for data capture.

Both of the views described above have merit and, ultimately, the decision as to which to pursue will be based on several factors — ones that have been discussed elsewhere in this paper. The level of technical support from library system staff will play a part in the decision. Staff will be able to carry out data capture from certain types of media with an out-of-the box computer. But, as this project has proven, processors will be working with non-integrated tools and their workflow will be inefficient. Additionally, they will find it difficult to capture data from certain media. The problems described with migrating data from 5 ¼” floppy disks are but one example. How will staff capture data from a donor’s hard drive, be it in the field or at the archive? Similarly, how will the SHC capture and preserve a donor’s email? Or websites? There are software applications and

hardware to assist with many of these tasks. But, as noted previously, a number of the software applications are Unix-based and, consequently, would require modification of the standard Windows-based library workstations for use. Likewise the hardware on the standard workstation would require changes, either with the addition of a mini-jukebox or change of the workstation's internal drives.

Cost, no doubt, will also be a factor in determining the SHC's approach to data capture, although it need not be a major determinant. Many of the software tools necessary for data capture are shareware or freeware. And the components required to build a mini-jukebox would cost less than \$500⁶.

Incorporation of the data capture workflow described in this paper into the ingest process for the university's digital repository could reduce the SHC's need for special software and the accompanying technical support. But even if such steps as bitstream imaging, virus checking, file copying, integrity checking and metadata capture were carried out by the digital repository, the SHC would still need devices capable of reading the digital media it collects.

Indeed, the Task Force on Archiving Digital Information was correct, and prescient, in 1996, when the group wrote that preservation of "the rapidly expanding corpus of information in digital form that represents our cultural record" will require institutions to understand and commit "technically, legally, economically and

⁶ With "Appendix A: Mini-Jukebox Configuration Report" of the "Prometheus Installation Guide" (National Library of Australia, 2008, pp. 13-17) as a model, SHC staff priced the components to create a mini-jukebox for the Southern Historical Collection in January 2009. The proposed SHC mini-jukebox consisted of a four-bay storage tower filled with two SATA DVD drives, one 3 1/2" floppy disk drive combined with a compact flash card reader and one 3 1/2" floppy disk drive. Plans called for the mini-jukebox to connect to an SHC workstation via a multi-lane connection, which required a multi-lane card for the workstation and a multilane cable to connect the mini-jukebox and workstation. The projected cost for the mini-jukebox and accompanying equipment was \$385. Although staff might choose to configure a mini-jukebox with different drives today, the cost likely still would not exceed \$500.

organizationally to the full dimensions of the task” (Task Force on Archiving of Digital Information, 1996, pp. 3-4). For the Southern Historical Collection that time has come.

REFERENCES

- ACPO. (n.d.) *Good Practice Guide for Computer-Based Electronic Evidence*. Retrieved February 10, 2009 from http://www.7safe.com/electronic_evidence/ACPO_guidelines_computer_evidence_v4_web.pdf.
- Arms, W. (2000). *Digital Libraries*. Cambridge, Mass: MIT Press [Electronic version]. Retrieved 11/22/08 from <http://www.cs.cornell.edu/wya/diglib/MS1999/index.html>
- Beagrie, N. (2005). Plenty of room at the bottom? Personal digital libraries and collections. *D-Lib Magazine*, 11,6, June 2005. Retrieved January 12, 2008, from <http://www.dlib.org/dlib/june05/beagrie/06beagrie.html>.
- Carrier, Brian (2005). *File System Forensic Analysis*. Boston, Mass.: Addison Wesley Professional [Electronic version]. Retrieved October 23, 2008 from <http://proquest.safaribooksonline.com.libproxy.lib.unc.edu/0321268172>.
- Cunningham, A. (1999). Waiting for the ghost train: Strategies for managing electronic personal records before it is too late. *Archival Issues*, 24, 1, 55-64. Retrieved September 12, 2008 from WilsonWeb database.
- Elford, D.; Del Pozo, N.; Mihajlovic, S.; Pearson, D.; Clifton, G.; & Webb, C. (2008). Media matters: Developing processes for preserving digital objects on physical carriers at the National Library of Australia. Presented at World Library and Information Congress: 74th annual IFLA General Conference and Council, Quebec City, Canada, August 10-14, 2008. Retrieved October 2, 2009 from <http://www.ifla.org/IV/ifla74/papers/084-Webb-en.pdf>.
- Farmer, D. & Venema, W. (2005). *Forensic Discovery*. Upper Saddle River, N.J. : Addison-Wesley [Electronic Version]. Retrieved February 1, 2009 from <http://www.porcupine.org/forensics/forensic-discovery/>.
- Forensics wiki, <http://www.forensicswiki.org>.

- Garfinkel, S. & Cox, D. (2009, February 10). *Finding and Archiving the Internet Footprint*. Paper presented at the First Digital Lives Research Conference: Personal Digital Archives for the 21st Century, London, England, 9-11 February 2009. Retrieved March 16, 2009 from <http://www.simson.net/clips/academic/2009.BL.InternetFootprint.pdf>
- Henry, L. (2000). Schellenberg in Cyberspace. In R.C. Jimerson (Ed), *American Archival Studies: Readings in Theory and Practice* (pp. 569-588). Chicago, Il: Society of American Archivists.
- Hyry, T. , & Onuf, R. (1997). The personality of electronic records: the impact of new information technology on personal papers. *Archival Issues*, 22, 1, pp. 37-44.
- InterPARES 1 Project, “The InterPARES glossary,” [electronic version] in The Long-term preservation of authentic electronic records: Findings of the InterPares project. Retrieved March 20, 2009 from www.interpares.org/book/interpares_book_q_gloss.pdf
- InterPARES 2 Project, “The InterPARES 2 Project Glossary,” [electronic version] in *International Research on Permanent Authentic Records in Electronic Systems (InterPARES) 2: Experiential, Interactive and Dynamic Records*, Luciana Duranti and Randy Preston, eds. (Padova, Italy: Associazione Nazionale Archivistica Italiana, 2008). http://www.interpares.org/display_file.cfm?doc=ip2_book_glossary.pdf
- John, J. (2008, September 29). *Adapting technologies for digitally archiving personal lives: digital forensics, ancestral computing, and evolutionary perspectives and tools*. Paper presented at iPres 2008: The Fifth International Conference on Preservation of Digital Objects, London, England. Retrieved November 18, 2008 from http://www.bl.uk/ipres2008/presentations_day1/09_John.pdf.
- Kenney, A., & Rieger, O. (Eds.). (2000). *Moving theory into practice: digital imaging for libraries and archives*. Mountain View, CA : Research Libraries Group.
- Kiehne, T., Spoliansky, V., & Stollar, C. (2005a). *From floppies to repository: a transition of bits*. Unpublished manuscript. Retrieved October 10, 2008 from <https://pacer.ischool.utexas.edu/handle/2081/941>.
- Kiehne, T., Spoliansky, V., & Stollar, C. (2005b). Index of files with Joyce comments. [Spreadsheet]. Retrieved November 16, 2008 from <https://pacer.ischool.utexas.edu/handle/2081/584>.

- Lee, C. (2008, April 7). *Preserving attachments from an e-mail collection: the good, the bad, the ugly and the thought-provoking*. Slides presented at the conference What to preserve? Significant properties of digital objects, British Library, London, England. Retrieved March 15, 2009 from <http://www.dpconline.org/docs/events/080407/sigpropsLee.pdf>.
- Library of Congress, National Digital Information Infrastructure and Preservation Program. (2002). *Preserving our digital heritage: plan for the National Digital Information Infrastructure and Preservation Program: a collaborative initiative of the Library of Congress*. Washington, D.C. Retrieved November 7, 2008, from <http://purl.access.gpo.gov/GPO/LPS27275>.
- Lukesh, S. (1999). E-mail and potential loss to future archives and scholarship of the dog that didn't bark. *First Monday*, 4,9, 1999, September 6. Retrieved November 26, 2008 from <http://firstmonday.org/htbin/cgiwrap/bin/ojs/index.php/fm/article/view/692/602>.
- Lynch, C. (1996). The Integrity of Digital Information: Mechanics and Definitional Issues. *Journal of the American Society for Information Science*, 45, 10, pp. 737-744. Retrieved March 20, 2009 from Wiley Interscience database.
- Marshall, C., Bly, S., Brun-Cottan, F. (2006). The long term fate of our digital belongings: toward a service model for personal archives, *Proceedings of IS&T 2006*, (Ottawa, Canada, May 23-26, 2006), Society for Imaging Science and Technology, Springfield, VA, 2006, pp. 25-30. Retrieved January 12, 2008 from <http://arxiv.org/abs/0704.3653>.
- National Library of Australia. (2008, November 25). *Prometheus Installation Guide*. Retrieved December 6, 2008 from [http://sourceforge.net/projects/prometheus-digi/files/Prometheus Rel v1.1b/Prometheus_Component_Installation_Guide_v1.Pdf](http://sourceforge.net/projects/prometheus-digi/files/Prometheus%20Rel%20v1.1b/Prometheus_Component_Installation_Guide_v1.Pdf).
- Novak, A. (2006). Fixity checks: Checksums, message digests and digital signatures. Retrieved March 10, 2009 from http://www.library.yale.edu/iac/DPC/AN_DPC_FixityChecksFinal11.pdf
- Paquet, L. (2000) Appraisal, Acquisition and Control of Personal Electronic Records: From Myth to Reality. *Archives and Manuscripts* 28, no. 2, pp. 71-91
- Paradigm project. (2007). Workbook on Digital Private Papers. Retrieved October 15, 2008 from <http://www.paradigm.ac.uk/workbook>.
- Pearce-Moses, R. (2005). A Glossary of Archival and Records Terminology. Retrieved March 15, 2009 from <http://www.archivists.org/glossary/>

- Peters, C. (2006). When not all papers are paper: A case study in digital archivy. *Provenance*, 24, pp. 23 – 35. Retrieved October 9, 2008 from <https://pacer.ischool.utexas.edu/handle/2081/2226>.
- PREMIS Working Group. *Preservation Metadata for Digital Materials*. PREMIS: PREservation Metadata Implementation Strategiess website. Retrieved October 25, 2008 from <http://www.oclc.org/research/projects/pmwg/default.htm>.
- Ross, S. & Gow, A. (1999, February). *Digital archeology: Rescuing neglected and damaged data resources*. Retrieved 3/22/09 from <http://www.ukoln.ac.uk/services/elib/papers/supporting/pdf/p2.pdf>
- Schmidt, L. (2007). *Preservation of a born digital literary genre: Archiving legacy Macintosh hypertext files in DSpace — Appendix A (Works by other authors series)*. Unpublished manuscript. Retrieved November 16, 2008 from <https://pacer.ischool.utexas.edu/handle/2081/9008>.
- Stollar, C. & Kiehne, T. (2006). Guarding the guards: Archiving the electronic records of hypertext author Michael Joyce. *New Skills for the Digital Era*. Colloquium at the National Archives, May 31-June 2, 2006. Retrieved October 25, 2008 from http://rpm.lib.az.us/NewSkills/CaseStudies/4_Stollar_Kiehne.pdf.
- Symantec Corporation. (2009, April). *Symantec Global Internet Security Threat Report: Trends for 2008*. Retrieved May 18, 2009, from http://www4.symantec.com/Vrt/wl?tu_id=gCGG123913789453640802
- Task Force on Archiving of Digital Information. (1996). *Preserving digital information: report of the Task Force on Archiving of Digital Information*. Washington, D.C.: Commission on Preservation and Access. Retrieved November 7, 2008, from <http://www.clir.org/pubs/abstract/pub63.html>.
- White, R. (2008). *How Computers Work* (9th ed.). Indianapolis, IN: Que Publishing.

Appendix A

STEPS FOR INGEST OF DIGITAL MEDIA IN THE SOUTHERN HISTORICAL COLLECTION

PREPARING TO INGEST

(1) After retrieving digital object (CD, DVD, floppy disk) from stacks, open the document **Dig_Med_Transfer.xls**. The file can be found at **G:\mss\Digpres_files\Dig_Med_Transfer.xls**. Enter the item number of the SHC item number of item that you are ingesting (i.e. DCD_5111_1).

(2) *This is an optional step. If the media and/or its label are distinct (i.e. you can't read the label, the medium shows signs of age, etc), then photograph it.* Photograph each item and its case (if it has one). Make sure the photo captures all labels and writing on the item as well as on the case and jewel box. Give the photo image the same file name as the item, adding the suffix *photo* to the end of the file name. Save the photo to the top level of the directory in which digital object and files will be stored. For instance, an image of the digital object DCD_5111_1 would receive the file name *DCD_5111_1_photo.jpg* and be saved in a folder labeled *DCD_5111_1*.

Write down in your notes all information written on the medium or its label. You'll need to use this information later.

(3) Insert the item into the appropriate drive on your workstation. Do not open any files on the medium. By opening a file you risk making changes to the file content or its metadata. You might also expose your workstation to viruses.

CREATE A BITSTREAM COPY (FOR CD'S ONLY)

- (4) Make a disk image (bitstream copy) of the digital object. Follow these steps:
- Open the application *Record Now* on your workstation.
 - Select the *Copy* tab
 - Click on the digital object for which you want to make a disk image
 - Choose *Save Image to Hard Drive*
 - Name the image file, using the name of the source digital object followed by the additional extension *bitstream*. For example a bitstream copy of DCD_5111_1 should be named *DCD_5111_1_iso_bitstream*.
 - Save the disk image to the Digital Manuscripts folder in the mss_proj directory of the Digital Archive. You will later move the disk image into a folder created by the Data Accessioner.

Note: If *RecordNow* is unable to create a disk image of the digital object, temporarily set the object aside and note the difficulty in the notes section of the spreadsheet. *Record Now* is unable to copy certain types of disk formats.

One short-term solution to this problem is use of *Disk Utility* on Mac operating systems. Within *Disk Utility* select “New Image.” Then select “read-only” for “image format” and “none” for “compression.” Use the same naming conventions described above. *Disk Utility* will produce a file ending with the extension .dmg. Rename the file, changing the .dmg extension to .iso. You have now created a disk image that will be recognized across operating systems.

CREATING A CHECKSUM FOR THE BITSTREAM COPY (FOR CD’S ONLY)

(5) Open the program *MD5 Hash Generator* on your workstation. If, for some reason, the workstation does not have *MD5 Hash Generator*, the application can be downloaded for free from <http://drnaylor.co.uk/software/md5/>.



MD5 hash generator has drag-and-drop functionality. Simply drag the bitstream copy into the “For the file” section of the hash generator window. The tool will generate an MD5 hash value (a checksum) for the bitstream copy.

Once you have generated a checksum, click “Generate MD5 file next fo file” on the Hash Generator. Also click on “Copy to Clipboard.”

Paste (Control V) this hash value into the Checksum column of the Dig_Media_Transfer spreadsheet.

VIRUS CHECK

(6) Go to the “Program” menu and select “Symantec AntiVirus” from the Symantec Client Security folder. Depending on the digital medium that you are ingesting, select either “Scan a Floppy Disk” or “Custom Scan” and select the drive onto which you’ve loaded the CD.

Select “Scan Options”

Under file types, choose “All types”

Click on the “Advanced” button. Under “When scanning compressed files” select “Scan files inside compressed file” and choose for the software to scan 3 levels deep.

Click “OK” within the “Scanning Advanced Options” window.

You will then return to the “Scan Options” window.

Click on the “Actions” button.

Click on the “Actions” tab.

Select “Macro virus” and then, within the “First Action” window, select “Leave alone (log only)”

Select “Non-macro virus” and then, within the “First Action” window, select “Leave alone (log only)”

Select “Security Risks” and then, within the “First Action” window, select “Leave alone (log only)”

Click “OK” within the “Actions” window.

COPY FILES FROM DIGITAL MEDIUM

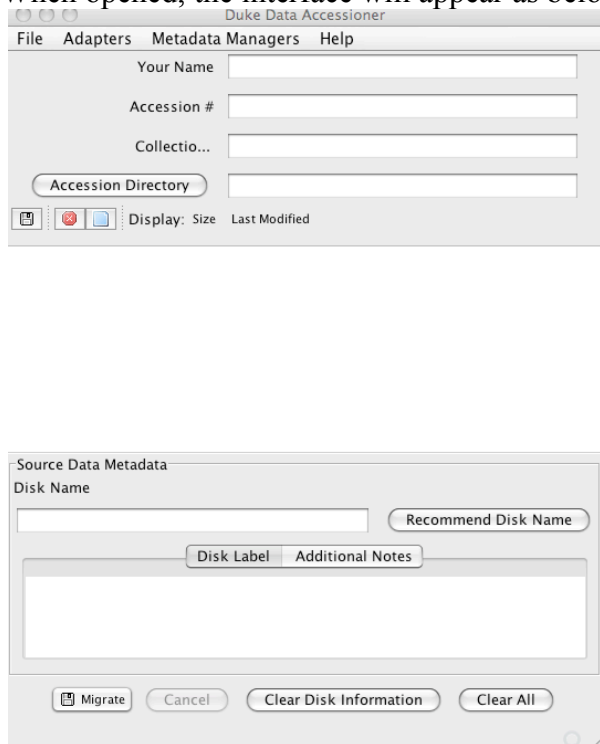
(7) Keep the source digital object in the workstation drive and open *Data Accessioner*.

The program can be found at **C:\Program**

Files\Electronic_Records_Toolkit\DataAccessioner_0_3.

On a Windows computer, double click the file called “start.bat.” This will open a terminal window. You may ignore the terminal window, but do not close it.

When opened, the interface will appear as below.



Under the “Adapters” menu, select “JHove” and “Droid Adapter.” Enabling these adapters will allow JHove and Droid to identify file type and validate files during migration.

Under the “Metadata Managers” menu, select “Duke Premis.” This will structure data in Duke’s Premis form when the XML document is created during migration.

Enter the following data in the open fields at the top of the interface:

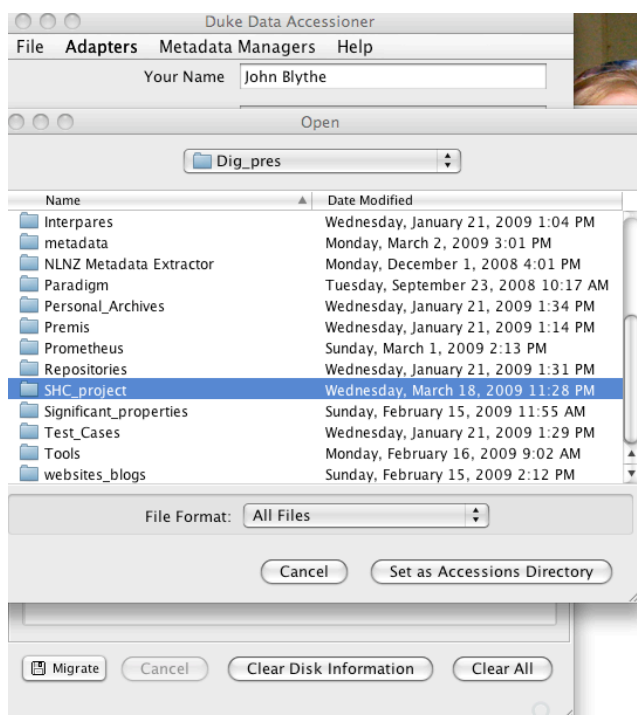
Your name: Name of the processor

Accession number: SHC item number (i.e. *DCD_5111_1*)

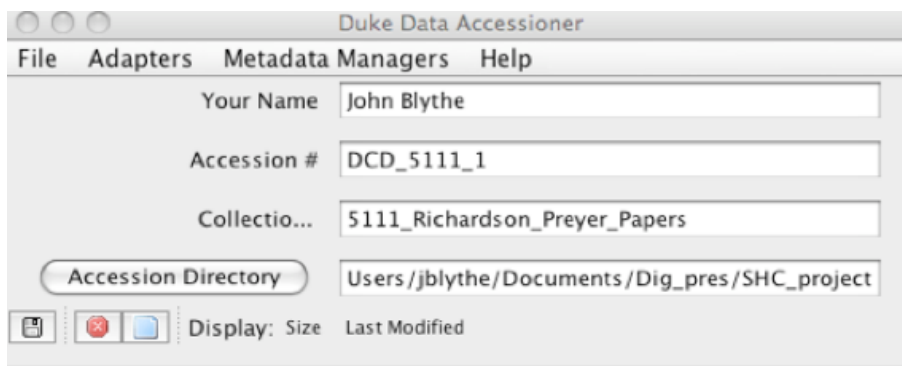
Collection: Collection number followed by name of collection (i.e.


5111_Richardson_Preyer_Papers)

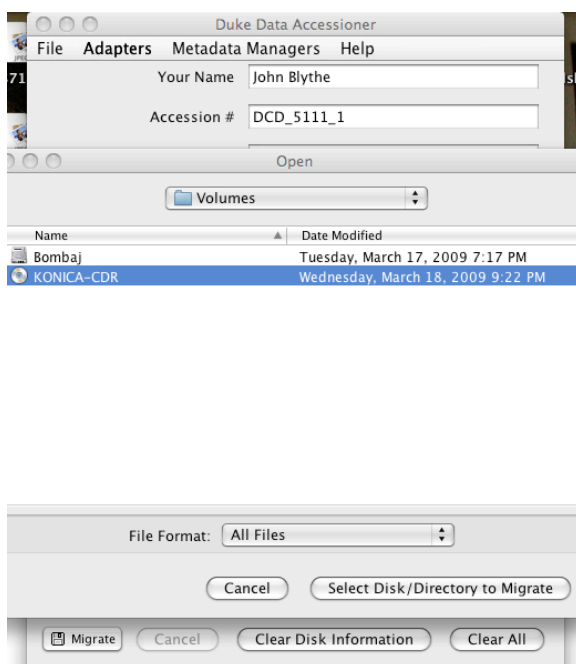
Then click the “Accession Directory” button and select the directory (folder) to which you plan to migrate the digital object. Currently digital items should be migrated to **G:\mss\Digpres_files**. Once you have selected the directory, click on “Set as Accession Directory.”



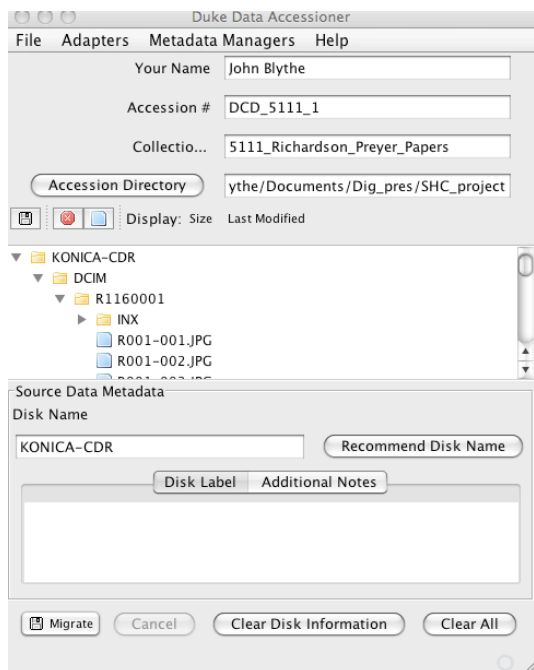
The file path for the folder that will contain the migrated object should then appear in the box to the left of “Accession Directory.”





Next, click on the  button (it resembles a floppy disk). This button will present you with possible digital objects for migration.



Once you have chosen the digital object, click on “Select Disk/Directory to Migrate.” In the large window of the Data Accessioner interface, you will then see a directory tree of the digital object that you have chosen to migrate.

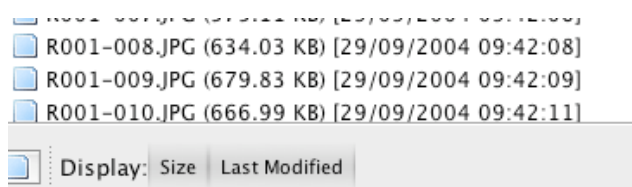


You can expand or collapse the tree by clicking once on the arrows or double-clicking on the folders.

Note the   buttons below the display window. The red Stop-sign-like button allows processors to deselect files that they choose not to migrate. The blue disk-like button allows processors to reverse their de-selection decision and instead include a file in migration.

The instances when a processor should not migrate all files are few and far between. The most likely scenario for de-selection is when the donor and archivist have agreed that certain files are not appropriate for the archive – either because of privacy issues or because the content of the files ranges beyond the scope of the collection.

Clicking on the words “Size” and “Last Modified” next to “Display” will add each file’s last modified date and size into the display window.



The “Disk Name” text box may already be populated or blank. When the text box is already filled in, then the Data Accessioner has automatically entered the digital object’s display name. Leave the display name as it is

If the text box is blank, then enter again the item number for the digital object.

The Southern Historical Collection does not use the “Recommend Disk Name” button.

Processors should note any text on the digital object or its label in the “Disk Label” text box. You can access the “Disk Label” text box by clicking on the “Disk Label” tab. The “Additional Notes” text box can be used to record information about restrictions, context and applications necessary to view the files.

The “Clear Disk Information” button will clear the directory window and the disk name, disk label, and additional notes boxes.

The “Clear All” will reset your name, accession number, collection title, accession directory, directory window, disk name, disk label, and additional notes.

Once you have entered information into the appropriate fields, click the “Migrate” button. Data Accessioner will then begin migrating files to the accession directory that you have designated.

A progress bar will appear at the bottom of the interface as the data is migrated. While the progress bar confirms that data is migrating, it is not an accurate predictor of the time remaining.

When the migration is complete, the message “Migration Successful” will display at the bottom of the interface.

Note: Additional documentation on use of the Data Accessioner is available in the program’s folder.

THE XML DOCUMENT

In addition to migrating files, Data Accessioner creates an XML document with MD5 hash values for each file migrated and the output of JHOVE and DROID evaluations of each file.

```

<?xml version= 1.0 encoding= UTF-8 ?>
<collection name="">
  <accession number="DCD_5111_1">
    <folder name="KONICA-CDR">
      <note>KONICA-CDR transferred by John Blythe on Wed Feb 04 23:02:07 EST 2009</note>
      <title qualifier="collection">5111_Richardson_Preyer_Papers</title>
      <folder name="DCIM">
        <folder name="R1160001">
          <folder name="INX">
            <file name="I001-01.JPG" last_modified="2004-09-29 09:42:44.0" size="417710"
              MD5="da785be0be3131a7f78d849acd2a1d38">
              <p1:object xmlns:p1="http://www.loc.gov/standards/premis/v1/">
                <p1:originalName>I001-01.JPG</p1:originalName>
                <p1:significantProperties>
                  <p1:dateLastModified source="DataAccessioner">2004-09-29
                    09:42:44.0</p1:dateLastModified>
                </p1:significantProperties>
                <p1:fixity source="DataAccessioner">
                  <p1:messageDigestAlgorithm>MD5</p1:messageDigestAlgorithm>
                  <p1:messageDigest>da785be0be3131a7f78d849acd2a1d38</p1:messageDiges
              <p1:messageDigestOriginator>DataAccessioner</p1:messageDigestOriginator>
            </p1:fixity>

```

This portion of the XML document includes the file name, the date the file was last modified and the MD5 hash value of the source file as well as the hash value of the file copy. The hash values should match. If the hash values did not match then Data Accessioner would indicate such during the migration process.

```

<p1:messageDigestOriginator>DataAccessioner</p1:messageDigestOriginator>
  </p1:fixity>
  <p1:format>
    <p1:formatDesignation source="JHOVE">
      <p1:formatName>JPEG</p1:formatName>
      <p1:formatVersion>1.01</p1:formatVersion>
      <p1:formatRecognitionStatus>Well-Formed and
        valid</p1:formatRecognitionStatus>
      <p1:formatType>image/jpeg</p1:formatType>
    </p1:formatDesignation>
    <p1:formatDesignation source="DROID">
      <p1:notes>DROID Version: V3.0. Signature File Version:
        13</p1:notes>
      <p1:formatName>JPEG File Interchange Format</p1:formatName>
      <p1:formatVersion>1.01</p1:formatVersion>
      <p1:formatRecognitionStatus>Positive (Specific
        Format)</p1:formatRecognitionStatus>
      <p1:formatType>image/jpeg</p1:formatType>
    </p1:formatDesignation>
  </p1:format>
</p1:object>
</file>
</folder>

```

This second excerpt shows the output of the JHOVE and DROID evaluations of a file.

The metadata shown in the two images above is provided for each migrated file. A single XML document contains metadata on every file that is included in one migration session. You will find the XML document in the top level folder of the migrated digital object.

PROTECTING THE FILES

Make all migrated files read-only. To do so, go to the folder of the item that you have just migrated. The folder will be within the directory **G:\mss\Digpres_files**. The folder will have the same name as the item that you have just transferred. For instance, the folder for DCD_5111_1 will be *DCD_5111_1*.

Right click on the folder.

Select "Properties"

Under the "General" tab look for the section marked "Attributes." Select "Read-only". If "Read-only" is already checked, uncheck it and then check it again. This will ensure that you are presented with the next step.

When you click apply you will be asked to choose between two options.

Select "Apply changes to this folder, subfolders and files"

Click "OK."

RETURN TO THE SPREADSHEET (DIG_MEDIA_TRANSFER.XLS)

Fill in the appropriate data in each field of the spreadsheet.

APPENDIX B

Sample Metadata Record Created by Data Accessioner

```

<?xml version="1.0" encoding="UTF-8" ?>

<collection name="">

<accession number="DCD_5203_9">

<folder name="050312_2041">

<note>050312_2041 transfered by John Blythe on Wed May 13 14:44:42 EDT 2009</note>

<description qualifier="label">J. Daniel Mahar James W. Davis Deeds and Probate</description>

<title qualifier="collection">5203_Adkins_Davis_Fulton_Family_Papers</title>

<file name="JWDavis.001a.JPG" last_modified="2004-11-15 22:31:40.0" size="4177812"
MD5="667ba6aef502e6f111a4cf9d6446bdd4">

<p1:object xmlns:p1="http://www.loc.gov/standards/premis/v1/">

<p1:originalName>JWDavis.001a.JPG</p1:originalName>

<p1:significantProperties>

<p1:dateLastModified source="DataAccessioner">2004-11-15 22:31:40.0</p1:dateLastModified>

</p1:significantProperties>

<p1:fixity source="DataAccessioner">

<p1:messageDigestAlgorithm>MD5</p1:messageDigestAlgorithm>

<p1:messageDigest>667ba6aef502e6f111a4cf9d6446bdd4</p1:messageDigest>

<p1:messageDigestOriginator>DataAccessioner</p1:messageDigestOriginator>

</p1:fixity>

<p1:format>

<p1:formatDesignation source="DROID">

<p1:notes>DROID Version: V3.0. Signature File Version: 13</p1:notes>

```

```

<p1:formatName>JPEG File Interchange Format</p1:formatName>

<p1:formatVersion>1.01</p1:formatVersion>

<p1:formatRecognitionStatus>Positive (Specific Format)</p1:formatRecognitionStatus>

<p1:formatType>image/jpeg</p1:formatType>

  </p1:formatDesignation>
: <p1:formatDesignation source="JHOVE">

  <p1:formatName>JPEG</p1:formatName>

  <p1:formatVersion>1.01</p1:formatVersion>

  <p1:formatRecognitionStatus>Well-Formed and valid</p1:formatRecognitionStatus>

  <p1:formatType>image/jpeg</p1:formatType>

    </p1:formatDesignation>

    </p1:format>

    </p1:object>

  </file>

<file name="JWDavis.001aa.JPG" last_modified="2004-11-15 22:35:20.0" size="4277703"
  MD5="2f25b4f9b5d05830e33ef2a425def1a3">

: <p1:object xmlns:p1="http://www.loc.gov/standards/premis/v1/">

  <p1:originalName>JWDavis.001aa.JPG</p1:originalName>

: <p1:significantProperties>

  <p1:dateLastModified source="DataAccessioner">2004-11-15 22:35:20.0</p1:dateLastModified>

  </p1:significantProperties>

: <p1:fixity source="DataAccessioner">

  <p1:messageDigestAlgorithm>MD5</p1:messageDigestAlgorithm>

  <p1:messageDigest>2f25b4f9b5d05830e33ef2a425def1a3</p1:messageDigest>

  <p1:messageDigestOriginator>DataAccessioner</p1:messageDigestOriginator>

  </p1:fixity>

```

```
<p1:format>
<p1:formatDesignation source="DROID">
  <p1:notes>DROID Version: V3.0. Signature File Version: 13</p1:notes>
  <p1:formatName>JPEG File Interchange Format</p1:formatName>
  <p1:formatVersion>1.01</p1:formatVersion>
  <p1:formatRecognitionStatus>Positive (Specific Format)</p1:formatRecognitionStatus>
  <p1:formatType>image/jpeg</p1:formatType>
  <p1:formatDesignation>
<p1:formatDesignation source="JHOVE">
  <p1:formatName>JPEG</p1:formatName>
  <p1:formatVersion>1.01</p1:formatVersion>
  <p1:formatRecognitionStatus>Well-Formed and valid</p1:formatRecognitionStatus>
  <p1:formatType>image/jpeg</p1:formatType>
  <p1:formatDesignation>
</p1:format>
</p1:object>
</file>
</folder>
</accession>
</collection>
```


APPENDIX C

PROCESSING BORN-DIGITAL ITEMS IN THE SOUTHERN HISTORICAL COLLECTION

John Blythe

April 2009

In September 2008 Wilson Library's Technical Services Department began developing a workflow for processing born-digital items collected by the Southern Historical Collection for the past decade. The SHC's born-digital holdings consist of:

- 102 CDs
- 81 3½" floppy disks
- 114 5¼" floppy disks
- 7 DVDs

The digital media hold a variety of file formats, including .jpeg, .tiff, .doc, dBase IV, dBase V and Microsoft Entourage.

The document that follows discusses the general issues related to capture of digital items and the current methods employed by SHC staff. As Wilson Library moves further into collection of born-digital items, the methods and approaches will likely change.

DIGITAL ARCHEOLOGY

Some have used the term *digital archeology* to refer to the early stages of the processing and preservation of digital items. The U.K's Paradigm project defines *digital archeology* as the retrieval of "data from obsolete software or hardware environments, and obsolete or damaged media, such as punch cards, 8" floppy disks and the wealth of other removable media which have been used since the earliest days of computing (*Paradigm Workbook*, p. 242)." In addition to *obsolete* and *damaged* media referred to in the Paradigm definition, we might also add *unstable* media. The Southern Historical Collection (SHC) seeks to move files from CDs, DVDs and other media because there are other media that offer the promise of longer-term stability. So, while neither the

Paradigm definition nor the metaphor of archeology are the perfect description of the steps involved in processing digital objects, we will use the term *digital archeology* because of its increasing currency.

Archivists at the University of Texas' Harry Ransom Center have broken down the digital archeology process into six steps (Stollar and Kiehne, p. 2). Their breakdown provides a helpful overview of the tasks that we seek to accomplish in the Southern Historical Collection.

- Receive and identify physical media
- Create a cataloging system for the physical media
- Copy files from physical media and record metadata
- Perform initial file processing (virus checking and file recovery)
- Create an item-level listing of all recovered files
- Create working copies of all files and protect the original copies.

Although this paper is not organized in terms of those six steps, they underpin all of the discussion that follows.

GETTING DATA OFF OF THE SOURCE MEDIUM

Born-digital content will arrive on a variety of media. These media include CDs (of all types), DVDs (of all types), floppy disks (both 3 ½" and 5 ¼"), Jazz disks, Zip disks, flash drives, laptops, desktop computers and even free-standing hard drives (drives not connected to an operating system). In the future there may be a need to pull digital files off such devices as cell phones and PDA's.

The first step in the digital archeology process is the setup of a workstation that can connect to/access any of the source media listed above. The workstation will need a variety of drives or need to connect to a device with numerous different drives.

Once a workstation has been set up, digital curators can begin the process of *capturing* digital objects (that is, moving digital objects from their source media to media

that provide for long-term, safe storage and access). Processors must make identical copies of digital objects without damaging the original in any way. They need to document all their steps and they should be capable of proving that the original digital object and its copy are exactly the same.

Currently, the Southern Historical Collection is equipped with IBM/Lenovo workstations that allow processors to capture files off of CDs and 3.5" floppy disks. However, floppy drives are no longer standard on workstations. As the SHC swaps out old computers for new ones, those in charge will need to ensure that at least one workstation is equipped with a floppy drive.

At present the Southern Historical Collection's digital holdings do not include hard drives, but other archives have begun collecting such. Donors may expect us to take hard drives and ideally we should be prepared to do so. If for no other reason, accessioning hard drives will reduce the amount of time required to copy files to a temporary digital storage medium (be it DVD, CD or a portable hard drive).

CREATING BITSTREAM COPIES

To accomplish the tasks described above, processors should **make a bitstream copy of the digital item** (CD, floppy disk, hard drive, etc). This means copy the digital item bit for bit. The reasons for this step are twofold. One, the creation of a bitstream copy ensures that an exact copy of the digital item is available. In essence the bitstream copy becomes a surrogate for the original digital item and reduces the need to recover files from the original media should corruption of copied files occur. The second reason to create a bitstream copy is to ensure that digital curators have access to "hidden" data. Such data may help processors determine the application that created the files and discover login or password information necessary to access data. This information may exist outside the regular directory structure or may be contained in files that a processor does not perceive as important on her first examination of the content of the digital medium.

There is some debate as to whether creating and retaining bitstream copies is necessary. Currently Duke's electronic records archivist does not create bitstream copies. He argues that the time involved in doing so and the disk space required to store a bitstream copy don't merit the extra effort. But he also says that he may change his view.

Even if a processor does create a bitstream copy, there is no clear guidance on the length of time that the copy should be retained. Some suggest that the copy can be deleted once the individual files on a source medium have been migrated. But others point out that preserving the bitstream copy for a longer period (perhaps two or three years) provides a processor with the ability to carry out additional processes on the content of the entire disk if there is a future need for such.

The easiest way to create a bitstream copy is to make a disk image of the digital medium. There are numerous ways by which to do so. Many CD burning applications allow users to create disk images, often providing the option to "copy this disk." On Mac systems, "Disk Utility" can be used for such a task. On Unix systems there are two commands that enable the creation of bitstream copies. The "dd" command has long been used to image hard drives and other digital media. The command "dcfldd" is one used in computer forensics and includes the additional capability of creating a hash value (i.e. a checksum) of the disk image .

The Southern Historical Collection currently does not have a viable method for creating bitstream copies. Bitstream copies (i.e. disk images) of CD's are created with RecordNow, a CD-burning tool that comes pre-installed on the collection's Windows workstations..While this tool has been readily available at the Southern Historical Collection, it is not a long-term solution. RecordNow creates bitstream copies of only CD's. Record Now is also proprietary software and, as such, there is little documentation about its operation. It's not immediately clear whether the software is actually making a bitstream copy that includes hidden data. A second issue with RecordNow is its inability to make bitstream copies of CDs recorded in

all disk image formats. On several occasions the software was unable to read or copy CDs. The problems appeared to result from CD's created with file systems undecipherable to Windows computers. In some cases the CDs were created with Mac OS file systems . In others they were created in the ISO 9660 Rockridge standard (for more details on the ISO 9660 Rock Ridge standard, visit http://en.wikipedia.org/wiki/ISO_9660).

These CD's were eventually copied using Disk Utility, an application available on Mac computers. The Mac employed for this job belonged to an SHC student staffer. If technical services builds a digital capture work area, it may want to consider equipping such a space with a Windows-based computer and a Mac-based workstation.

The Unix operations “dd” and “dcfldd” promise possible solutions to some of the problems discussed above. Unfortunately the workstations in the SHC lack Unix operating systems.

MIGRATING FILES TO THE PRESERVATION ENVIRONMENT

In addition to making a bitstream copy of the entire digital object, processors should separately copy each individual file on the medium. This step differs from creation of a bitstream copy because the copied bits are structured as files and are more easily read by a computer's operating system. These files should be copied in the format and version in which they exist on the source medium (CD, DVD, hard drive, etc.). Although the files may undergo normalization or migration later, it is important that the archive hold a copy of them as they existed on the source medium.

Depending on the methods that processors use to copy a file from its source medium to a preservation environment, the file's MAC time may be changed. MAC time is automatically created metadata logging the date and time when a file was last modified (M), accessed (A), and changed or created (C). *Modified* (also known as *mtime*) refers to when the content of the file most recently changed. *Accessed* (*atime*) identifies when a

file was most recently opened by a person or software. File systems use *ctime* differently. On Unix systems, *ctime* identifies when a file's metadata was last changed (i.e. a change in permissions, a change of owner or even a change in other MAC time metadata). Windows file systems treat *ctime* as *creation time*. The metadata refers to the time a file is created.. For archives MAC times might provide processors with a rough timeline of when files were created. But archivists should be wary of placing too much value in metadata gathered from MAC times. Windows systems generate a new *ctime* every time a file is copied, leaving processors unsure as to whether *date created* actually refers to the date the content of a file was created or the date of the copy (Kiehne, Spoliansky, Stollar, p. 6; Stollar and Kiehne, p. 3)

In transferring digital items to a preservation environment, processors should seek to preserve (or at least document) the directory structure as it exists on the source medium. This adheres to basic archival principles. In representing the directory structure, the processor is ensuring authenticity and maintaining original order

The Southern Historical Collection copies files from digital media using the Data Accessioner, a Java-based tool developed by Duke University's electronic records archivist (<http://library.duke.edu/uarchives/about/tools/data-accessioner.html>). Running on an archivist's computer, the Data Accessioner copies the directory structure of the digital medium, copies the individual files from the medium and then (using JHOVE and DROID) seeks to identify the file format and determine whether the file is valid and well-formed. In carrying out this process, the tool creates checksums and stores those values as well as the results of the format checks in an XML document. The Data Accessioner also includes a field into which a processor can record writing and other information that appears on the label of a digital item or its casing.

When this project began there was limited space in the digital archive. As a result, file copies created with the Data Accessioner were stored on the G-drive (G:\mss\Digpres_files\). This was designed as a temporary measure. All copies of

files and the source medium should be migrated directly to the digital archive and stored there. The digital archive should be set as the “Accession Directory” when using Data Accessioner

WRITE BLOCKING

The guarantee of authenticity is central to the mission of an archive. With that in mind processors must avoid accidentally altering the original digital items (found on the source medium) or digital copies. This caution applies to both the digital content and its associated metadata. Consequently, write blockers are recommended for accessioning data from digital media that has not been fixed (hard drives, floppy disks, CD-RW, USB thumb drives and compact flash cards) (John, *p. 3*).

In addition to protecting the original digital item from being overwritten, it is also important to protect the digital item’s copy from inadvertent alterations. Change of an object’s properties to “read-only” are the best way to create such protection. Admittedly manually changing the properties of hundreds, if not thousands, of files is a daunting prospect. This step is best accomplished with some type of automation.

The migration of digital media in the Southern Historical Collection has not included use of a write-blocker. Fortunately, the CD’s that were migrated had already been finalized and so there was no risk of accidentally overwriting them. The SHC has also migrated files from 3.5” floppy disks. These media did face the risk of accidental alteration.

To protect the file copies, the SHC has manually changed the file permissions to read-only for each directory, sub –directory and file migrated. Data Accessioner appears to make the top-level folder of a migrated digital item “read-only,” but that permission does not appear to extend down to sub-folders and files. In the Windows operating system, it is possible to set “read-only” permissions for the top level folder and, at the same time, apply the same permission to all lower level folders and directories. However, when using the Data Accessioner, this requires

un-checking the “read-only” box automatically selected by the software tool and then re-checking “read-only,” which, in turn, prompts the Windows operating system to ask whether a user would like to apply the same permission to sub-folders and files.

METADATA

Important preservation-related metadata to capture for all digital items in the Southern Historical Collection include:

- The item’s unique identifier (if an item does not have a unique identifier then the processor should assign it one).
- The source of the digital item (provenance, where it came from, the collection to which it belongs)
- The file format of the digital item (is it a jpeg, tiff, doc, txt, etc?)
- The date the digital item was transferred from the source medium to the medium on which it is being preserved and stored (at least for the short term)
- The process used to migrate and copy the digital item to the storage medium
- A MD5 checksum for the digital item (more on this later).
- Any information about the normalization or migration of a file to another file format – a format that is not in danger of obsolescence.
- All information written on a digital item’s label or casing.

There may be other metadata that is important for preservation for certain types of digital items. For instance, when accessioning a computer or its hard drive, the computer brand, its model number, serial number and operating system are important. This document will not delineate the required preservation metadata for each type of digital item.

The Southern Historical Collection records digital capture-related metadata using two methods. As described elsewhere in this document, the Data

Accessioner creates a PREMIS-structured XML document listing the directory structure and files on a digital item. Using JHOVE and DROID, the tool also seeks to identify each file's format and lists the possibilities in the XML document. Finally, the XML document includes the MD5 hash value for each file.

A spreadsheet is used to record other capture-related information in the Southern Historical Collection. The spreadsheet includes the following fields:

- *Item Number*
- *Item type*
- *Date Accessioned*
- *Virus Check – yes or no*
- *Write-Protected – yes or no*
- *Checksum for bitstream copy*

The SHC has experimented with photographing each digital item. This is a adopted by the British Library and provides processors with a means of recording writing or printing that appears on the digital item or its casing. Although the Data Accessioner includes a field into which a processor can enter such information a photograph allows the processor to capture the look of the digital item when it was accessioned and also provides a means of recording hard-to-decipher handwriting.

VIRUS CHECKS

An important step in the processing of born-digital objects is the virus check. The Southern Historical Collection will be collecting files from donors with a range of computer file management skills. We can't be sure that they regularly ran virus checks on their computers. Consequently some files may arrive at the SHC with viruses. To avoid infecting library workstations and storage systems as well as the computers of future users, processors should check digital items for viruses. Virus checking software should be able to identify viruses contemporary at the time of a file's creation. As with many of

the steps in preserving born-digital items, virus checking is best carried out as a batch process and software chosen for the task should be capable of doing such.

At the Southern Historical Collection, digital items have been checked for viruses only sporadically. Fortunately, simply copying files from their source medium to a preservation storage environment does not require the opening of such files. Consequently the risk of infecting workstations or storage systems in the Southern Historical Collection has been low. When virus checks have been performed, they've been done with the Symantec anti-virus software installed on the processors' workstations. Thus far the software has found no digital items with viruses. However, there has not been sufficient study of the literature to determine whether the Symantec software is an appropriate long-term solution. One concern is that the software allows a user to remove the virus. But doing so runs the risk of altering the file. Currently the software defaults to repairing the file, so a user must change its settings to "log virus only" to ensure that no inadvertent file alteration occurs.

CHECKSUMS AND HASH VALUES

Long-term preservation of digital items requires a mechanism by which those responsible for care of the items can ensure their integrity. Checksums, sometimes also called a hash value, provide that vehicle. Put simply, a checksum is a mathematical equation (an algorithm) that generates a value based on a computation of the bits within a file. Because no two files are the same (unless they are exact copies of each other), the bit strings for each are not identical. Consequently it is possible to create an equation using these bit strings that will likely yield a unique result for each file. The MD5 hash function is one such equation and it is commonly used in digital preservation. The function generates a 32-character hexadecimal digit (a hash value or checksum). By storing the hexadecimal digits, a digital curator can periodically check to make sure the files under her care have not been corrupted, i.e. a file's MD5 value (the hexadecimal digit) should be the same each time the function is performed.

Checksums are also useful in determining whether duplicate copies of files exist within newly accessioned digital objects. For instance, two files may have different names, but if they both have the same checksum then they are identical. Similarly, a checksum will help determine whether two files with similar names in different directories are, in fact, the same file.

As archives begin accessioning entire hard drives, they likely will find application and operating system software among the files they have taken in (Peters, p. 26; Kiehne, Spoliansky, Stollar, *p. 13*). In general these files hold little archival value and should not be preserved. Checksums can help processors determine the files that are related to applications or operating systems. A comparison of the checksum values of accessioned files with known hash values for operating system and application files (values found in a *hash library*) will help processors rule out the files that need not be preserved. This comparison process can be done with some basic scripting (John, p. 3).

Checksums should be generated each time a digital item is copied or moved. Consequently, a checksum should be created:

- Prior to the copy of a digital item (i.e. a checksum should be created for each file on a CD and also for the CD itself). These checksums represent the original (source) files.
- After a copy has been made of the digital item. These checksums represent the copies of the original (source) file. The checksum of the file copy should then be compared with the checksum of the original file. The two values should match. If they do not match then the two files are not exact copies of each other.
- When a digital item is moved from one storage medium to another (i.e. if a file is copied to a library server and then later moved to the digital archive, then a checksum should be generated after the file has been moved to the digital archive to ensure that it was not corrupted when moved.

A checksum should be created for each digital item (i.e. a checksum for a disk image of a CD as well as a checksum for each file on the CD).

There should be a means by which to compare the checksum of the original (source) file with the checksum of the file copy. Because checksums are lengthy, it's best for this comparison to be performed by a computer.

Finally, integrity checks should be performed on digital files on a regular basis. An integrity check ensures that files in long-term storage (preservation files) remain unaltered through time and that if a change does occur, those responsible for their preservation and those using them are aware of the change. An integrity check is simply the creation of a second checksum of a file for which a checksum already exists. The second checksum is compared with the first (or original) checksum. If the two match, then the file has not changed. If the two do not match, then the file has changed (whether by accident or deliberately). The literature provides no clear direction on the frequency with which integrity checks should be performed.. Further discussion of integrity checks extends beyond the focus of this document since such tests should be performed within the preservation environment (i.e. a digital repository).

The Southern Historical Collection has relied on two methods to create checksums for digital items. Duke's Data Accessioner creates MD5 checksums in the process of copying original (source) files to a storage medium. The tool creates a checksum of the original file, copies the file and then creates a checksum of the file copy. The Data Accessioner then compares the two checksums. Both the checksums of the original file and the file copy are recorded in the XML document created by the Data Accessioner.

To create checksums for bitstream copies, the SHC uses the freeware tool MD5 Hash Generator (available at <http://drnaylor.co.uk/software/md5/>) The tool was first used in the SHC in conjunction with a project to digitize interviews from the Southern Oral History Program. MD5 Hash Generator allows for drag-and-drop operation and is also capable of batch processing. Unfortunately the MD5 Hash Generator does not provide a means by which to export the checksums to a

spreadsheet or some other document type. This means that each checksum must be manually copied (preferably by “cut and paste”) to the spreadsheet.

CONCLUSION

The workflow for the capture of data from digital items that is described in this document is merely a first step in the Southern Historical Collection’s move to collect and preserve born-digital materials. Many issues and questions developed from this 7-month process. They will need further discussion before the SHC can feel confident in venturing further into collecting in the digital realm. Some of the issues are as follows:

DATA ACCESSIONER

The Data Accessioner has proven useful in creating copies of source files, automatically collecting format metadata and generating checksums. The tool does not currently check for viruses nor does it write block the source medium. Although virus checking may be integrated into a future version and the tool’s creator says that write-blocking could be added, there are still issues to consider in relying on it exclusively for migration. The software is scarcely a year old and it is the project of one person. Although Duke’s electronic records archivist hopes that a community of developers will coalesce around Data Accessioner there is no guarantee that such will happen. If the electronic records archivist does not continue to develop his tool or leaves his job, then the SHC may be relying on an antiquated application.

WORKSTATION LIMITATIONS

The SHC workstations are currently only able to read CD’s, 3.5” floppy disks and USB thumb drives. Although the SHC collection includes 5 ¼” floppy disks, technical services does not have the capacity to read this media format. In the future, as significant digital content is created and stored on other media types, the SHC will need the capacity to read them and transfer digital items from them.

Additionally, the SHC is limited in the tools that it can use for processing born-digital content because its workstations use a Windows operating system. There are several Unix-based applications for creating bitstream copies, virus-checking and

extracting metadata. Several applications that integrate the various tasks involved in migrating digital items are also Unix-based. The SHC should also consider the purchase of a Mac computer to allow for flexibility in the digital capture process.

COLLECTING HARD DRIVES

The SHC has yet to accession a hard drive, but it's likely that curators will be presented with the opportunity in the future. As more of donors' papers are created and stored in digital formats, the larger the file size of the items the SHC will wish to accession. Burning donors' files to CD or DVD while in the field and then transferring the media to the SHC will likely prove time-consuming and risks the loss of important data because of an improperly burned CD/ DVD or a damaged CD/ DVD. If the SHC is to continue collecting digital items, it will need to develop methods by which to accession donors' hard drives or provide curators with the equipment to accession data from computers while in the field. In a similar vein the SHC will need to provide curators with a means of previewing digital materials on the donor's hard drive while also protecting them from inadvertent alterations. This same functionality is required for cases where a hard drive contains culturally significant data but the computer operating system of which it is a part is no longer operational.

WRITE BLOCKING , BITSTREAM COPIES AND VIRUS CHECKING

As the Southern Historical Collection increases both the volume of digital materials it collects and the types of digital materials (hard drives, compact flash cards, etc.), certain methods tried on current digital holdings may not scale well.

The SHC has no way to write block. This was not a problem for the born-digital items migrated this year because, with the exception of floppy disks, the media were fixed and could not be inadvertently altered. But hard drives, PDA's and flash memory devices are not fixed and data could be damaged accidentally.

If the SHC intends to create bitstream copies of all digital items accessioned (and not just CD's and DVD's) then it will need to find software or hardware that is capable of copying floppy disks, hard drives, compact flash cards, PDAs and other storage media.

Use of Symantec anti-virus software is also not a good long-term solution. Such software is designed to quarantine or repair files that have been infected or damaged. And it appears to default to repair rather than asking a user whether she would like the file repaired.

READING THE DIGITAL ITEMS

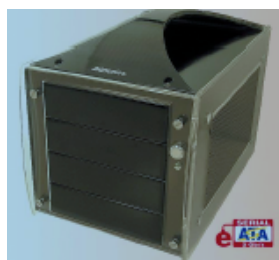
The Southern Historical Collection may find itself accessioning digital items created with software applications that the SHC does not possess. Without the proper software, processors may not be able to read the digital file to determine whether to include it and (since item-level processing is unlikely) those like it in the collection. Proper evaluation of digital items will require hardware or software that allows processors to examine the items even if the application does not create the exact look and feel of the item when it was first created.

POSSIBLE TOOLS

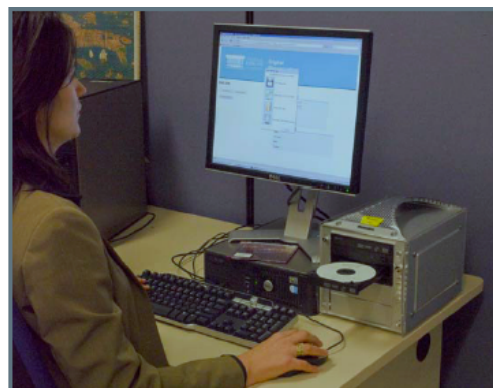
A listing of tool in use at various repositories, but because of the technical support they require, they were not tested here.

INTEGRATED DIGITAL CAPTURE

The National Library of Australia's Prometheus (Digital Preservation Workbench) project has made use of a mini-jukebox. This is a storage tower with four drive enclosures. The enclosures are fitted with a mix of drives – floppies, CD, USB connections, even hard drives (see **Digital_Objects_On_Physical_Carriers.pdf** — available from <http://prometheus-digi.sourceforge.net/download.html> — and **Component Installation Guide.pdf** — available from <http://prometheus-digi.sourceforge.net/faq.html>).



The mini-jukebox



The mini-jukebox in use

The British Library's Digital Lives project uses eMag Floppy Disk Conversion System Model MMC4000 and Stack-a-Drives for 8", 5.25", 3" and 3.5" floppy disks working with a proprietary floppy disk controller. The setup is much like a mini-juke box, i.e. a storage tower with several different types of drives (http://www.bl.uk/ipres2008/presentations_day1/09_John.pdf).

CREATING BITSTREAM COPIES

The open source software "cdrdao" (<http://cdrdao.sourceforge.net>) may be useful for making disk images. The application, which runs from a command line, is primarily intended to record audio and data CD in the "Disk-At-Once" (DAO) mode (that is, it does not create individual tracks on CD). Developers of the National Library of Australia's Prometheus workbench have suggested the use of CDRDAO for the creation of disk images. They point out that CDRDAO is the only application they've found that can make a disk image from CD's created with almost any image format (the various ISO 9660 standards as well as UDF (universal disk format)).

There are numerous computer forensics software packages that include bitstream copying among the tasks they perform. The advantage to such packages is the integration of bitstream copying into a multi-stepped process that includes the extraction of metadata and the generation of checksums. The cost of the integrated applications (which include tools that may not be useful in an archival context) may rule out their use in most archives. Many of the computer forensic applications cost \$2800 or more.

Other software is touted as being able to copy CD's formatted in the ISO 9660 Rock Ridge standards (Magic ISO Maker, Power ISO and ISO Recorder), but these tools require purchase and so were not tested in the Southern Historical Collection

METADATA

For further information on the characteristics of individual formats, consult the website *Sustainability of Digital Formats: Planning for Library of Congress Collections* (<http://www.digitalpreservation.gov/formats/>). For a wide-ranging discussion of preservation metadata (PREMIS, METS and other schemas) please consult Chapter 5 of the *Paradigm Workbook on Personal Digital Archives* (<http://www.paradigm.ac.uk/workbook/index.html>).

There are a variety of Java-based tools that can help with automatic collection of metadata. JHOVE (<http://hul.harvard.edu/jhove/>) allows users to determine the format of a digital file and whether a digital file meets the qualifications of a particular format (i.e. whether the file is *well-formed* and *valid*).

Like JHOVE, DROID (<http://droid.sourceforge.net/wiki/index.php/Introduction>) performs batch identification of file formats. Some have used these two tools in concert to guarantee reliable metadata about files. For instance, Duke's Data Accessioner Tool (whose functions are described in more detail elsewhere in this paper) checks the file format and well-formedness of files using both JHOVE and DROID.

Australia's Prometheus digital workflow employs DROID AND JHOVE as well as the National Library of New Zealand's Metadata Extraction Tool (<http://meta-extractor.sourceforge.net/documentation.htm>). This application automatically extracts preservation-related metadata from files and then creates an XML file containing the captured information. In many cases the preservation metadata captured automatically by the NLNZ Metadata Extraction Tool is technical information. For instance, the tool captures EXIF metadata for images – that is, camera settings and date and time the image

was created. Archivists may frequently find that information on provenance and rights has not been embedded in the digital file and, therefore, is not subject to automatic capture.

In the future, the arrangement and description of digital files may require an examination of the contents within them (and possibly opening them). At the Southern Historical Collection we have randomly viewed the contents of digital files using the Electronic Records Processor, another tool developed by Duke's electronic records archivist. The Electronic Records Processor is Java-based software designed to aid in the arrangement and description of digital files. The tool's features include a browser with which a user can preview a file without actually opening it (and thus preventing the spread of viruses) . But this tool works only on digital items that have been migrated with Data Accessioner.

Tools that may be helpful in the digital archeology process but that are untested in the Southern Historical Collection include:

- **CatFinder** (<http://www.mindspring.com/~shdtree/newsite/id9.html>) – a Mac-based utility for cataloging hard drives, CD ROMS, floppy disks and ZIP disks. The catalog can display file names, sizes, dates, file types and creators. The catalog can be exported as a single file. The tool may be helpful in capturing the directory of a digital object. CatFinder was used by the Harry Ransom Center in the processing of the digital files of hypertext author Michael Joyce.

The five applications listed below are tools used by computer forensic investigators.

- **Encase Forensic** (http://www.guidancesoftware.com/products/ef_index.asp)
- **Forensic Toolkit** (<http://www.accessdata.com/forensictoolkit.html>)
- **CD/DVD Inspector** (http://www.infinadyne.com/cddvd_inspector.html) \$549 + S/H
- **Helix 3** (<http://www.e-fense.com/register-overview.php>) \$14,95/month

- **The Sleuth Kit** (<http://www.sleuthkit.org/sleuthkit/>) - Available from Sourceforge

These tools allow a user to copy files, verify that the copy and the original are the same and repeat processes in a well-documented manner. They can read and examine numerous file systems as well as file types. For instance, Encase Forensic software is capable of viewing files from 400 file formats.

Although all the tools perform the same basic functions, each has tasks at which it excels. CD/DVD Inspector, for instance, is geared specifically toward analysis of optical discs. With the exception of Sleuth-Kit, which is open-source and shareware, these tools operate with proprietary software and are not cheap. While Data Accessioner or the Prometheus workflow may suffice for the short term, it may be necessary at some point to use forensic software to examine files for which the library does not possess the requisite software.

REFERENCES

John, J. (2008, September 29). *Adapting technologies for digitally archiving personal lives: digital forensics, ancestral computing, and evolutionary perspectives and tools*. Paper presented at iPres 2008: The Fifth International Conference on Preservation of Digital Objects, London, England. Retrieved November 18, 2008 from http://www.bl.uk/ipres2008/presentations_day1/09_John.pdf.

Paradigm project. (2007). Workbook on Digital Private Papers. Retrieved October 15, 2008 from <http://www.paradigm.ac.uk/workbook>.

Stollar, C. and Kiehne, T. (2006). Guarding the Guards: Archiving the Electronic Records of Hypertext Author Michael Joyce. *New Skills for the Digital Era*, Case Study 4. Retrieved Feb. 25, 2008 from http://rpm.lib.az.us/NewSkills/CaseStudies/4_Stollar_Kiehne.pdf