Managing and Transforming Digital Forensics Metadata for Digital Collections

Kam Woods, Alexandra Chassanoff, Christopher A. Lee
University of North Carolina - School of Library and Information Science
216 Lenoir Drive, CB #3360, 100 Manning Hall
Chapel Hill, NC 27599-3360
(919) 962-8366
{kamwoods, achass, callee}@email.unc.edu

ABSTRACT

In this paper we present ongoing work conducted as part of the BitCurator project to develop reusable, extensible strategies for transforming and incorporating metadata produced by digital forensics tools into archival metadata schemas. We focus on the metadata produced by open-source tools that support Digital Forensics XML (DFXML), and we describe how portions of this metadata can be used when recording PREMIS events to describe activities relevant to preservation and access. We examine open issues associated with these transformations and suggest scenarios in which capturing forensic metadata can support digital curation goals by establishing clear documentation of integrity and provenance, tracking events associated with pre-ingest and post-ingest forensic processing, and providing specific evidence of authenticity.

Categories and Subject Descriptors

H.3.7 [**Information Storage and Retrieval**]: Digital Libraries – *collection, dissemination, systems issues*

General Terms

Documentation, Reliability, Security, Legal Aspects.

Keywords

Digital forensics, disk imaging, preservation metadata, DFXML, interoperability, BitCurator.

1. INTRODUCTION

The process of preserving data encoded on digital media by extracting a bit-identical disk image has many advantages [14,15,23], but it also presents unique technical and organizational challenges. Many of these challenges are related to the metadata produced during acquisition, processing, and archival management of born-digital materials. The challenges often arise from working with implementations of metadata schemas that are *document-centric*; that is, schemas which have been designed primarily to accommodate the acquisition, analysis, and transformation of individual files (e.g., Microsoft Word documents or TIFF images). A disk image, in contrast, may contain hundreds, thousands, or even millions of files with many potential internal dependencies [23]. The disk image itself may not always be the final preservation target, but capturing and describing information about the internal structure and any potential dependencies is an important aspect of supporting ongoing preservation activities, as well as meaningful access and use.

Forensic analysis of disk images often produces large quantities of metadata. Much of this forensic metadata is initially reported at a very low level; for example, as patterns identified at various offsets into the raw bitstream. These reports may be transformed using a variety of intermediate procedures in order to generate derived metadata for specific tasks: retention within an Archival Information Package; storage within a database in preservation and access systems; or to support archival lifecycle processes.

In the following sections we describe specific metadata elements that can be extracted or derived using the BitCurator environment, and our evolving approach to mapping these items to archival metadata standards. In this paper, the preservation metadata target we focus on is PREMIS.

2. ACQUIRING FORENSIC METADATA IN BITCURATOR

The BitCurator Project is a collaborative effort led by the School of Information and Library Science at the University of North Carolina at Chapel Hill and the Maryland Institute for Technology in the Humanities at the University of Maryland. BitCurator aims to address two fundamental needs and opportunities for collecting institutions: (1) integrating digital forensics tools and methods into the workflows and collection management environments of libraries, archives and museums; and (2) supporting (potentially mediated) public access to forensically acquired data [16].

We are developing and disseminating a suite of open source tools. These tools are currently being developed and tested in a Linux environment. The majority of the software on which they depend can be compiled for Windows environments (and in most cases are currently distributed as both source code and Windows binaries), or runs in a cross-platform interpreter. We intend the majority of the development for BitCurator to support cross-platform use of the software. We are freely disseminating software developed by BitCurator under an open source (GPL, Version 3) license. All other software packaged within the BitCurator environment is distributed in accordance with the terms of the

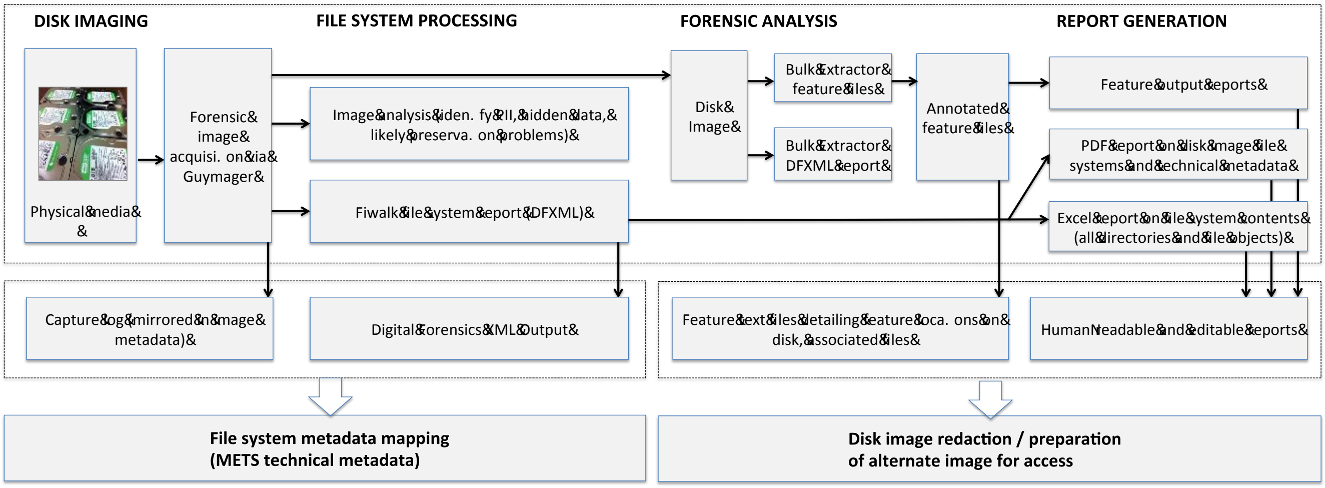
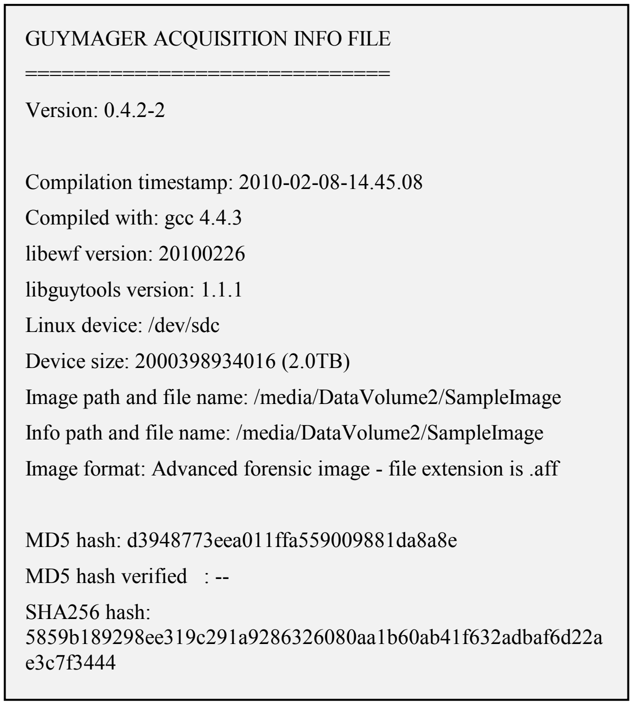

DISK IMAGING
FILE SYSTEM PROCESSING
FORENSIC ANALYSIS
REPORT GENERATION
Physical&media&
&
Forensic&
image&
acquisition&via&
Guymager&
Image&analysis&(identify&PII,&hidden&data,&
likely&preservation&problems)&
Fiwalk&file&system&report&(DFXML)&
Disk&
Image&
Bulk&extractor&
feature&files&
Bulk&extractor&
DFXML&report&
Annotated&
feature&files&
Feature&output&reports&
PDF&report&on&disk&image&file&
systems&and&technical&metadata&
Excel&report&on&file&system&contents&
(all&directories&and&file&objects)&
Capture&log&(mirrored&in&image&
metadata)&
Digital&forensics&XML&output&
Feature&text&files&detailing&feature&locations&on&
disk,&associated&files&
Human&readable&and&editable&reports&
File system metadata mapping
(METS technical metadata)
Disk image redaction / preparation
of alternate image for access

Figure 1: Overview of BitCurator disk image processing, metadata extraction, and redaction.



```
GUYMAGER ACQUISITION INFO FILE
===============================
Version: 0.4.2-2


Compilation timestamp: 2010-02-08-14.45.08
Compiled with: gcc 4.4.3
libewf version: 20100226
libguytools version: 1.1.1
Linux device: /dev/sdc
Device size: 2000398934016 (2.0TB)
Image path and file name: /media/DataVolume2/SampleImage
Info path and file name: /media/DataVolume2/SampleImage
Image format: Advanced forensic image - file extension is .aff


MD5 hash: d3948773eea011ffa559009881da8a8e
MD5 hash verified   : --
SHA256 hash:
5859b189298ee319c291a9286326080aa1b60ab41f632adbaf6d22a
e3c7f3444
```

Figure 2: Sample log metadata produced by Guymager during disk image acquisition.

capabilities of the physical device and would only be used in specialized circumstances. Essential details of the image capture (in the original DFXML) may be wrapped as technical metadata

capture), technical details of the processing environments, and MD5 and SHA256 checksums.

2.2 File system metadata

Information about the file system(s) contained within a disk image can be extracted using the *fiwalk* tool integrated into the current version of The Sleuth Kit, which is itself incorporated into the BitCurator environment. Output of *fiwalk* incorporates Dublin Core tags to identify the *creator* of the DFXML file (*fiwalk*, along with technical details on the environment in which it was run) and the *source* (the disk image file that was scanned). Note that these tags should not be treated the same as archival descriptive metadata. They are more accurately incorporated as *technical metadata* corresponding to an intermediate event (analysis of the file system(s) contained within the disk image).

A partial example of the DFXML output produced by *fiwalk* is shown in Figure 3. This section was extracted from the head of the DFXML file. Note the inclusion of technical details corresponding to the capture environment, a start timestamp in ISO 8601 format, and the name of the resulting image.

```
<?xml version='1.0' encoding='UTF-8'?>
<dfxml version='1.0'>
  <metadata

xmlns='http://www.forensicswiki.org/wiki/Category:Digital_Fo
rensics_XML'
  xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
  xmlns:dc='http://purl.org/dc/elements/1.1/'>
    <dc:type>Disk Image</dc:type>
  </metadata>
  <creator version='1.0'>
    <program>fiwalk</program>
    <version>4.0.2</version>
    <build_environment>
      <compiler>GCC 4.7</compiler>
      <library name="afflib" version="3.7.1"/>
      <library name="libewf" version="20130303"/>
    </build_environment>
    <execution_environment>
      <command_line>fiwalk -f -X
/media/kamwoods/DataVolume2/SampleImage.xml
SampleImage.aff</command_line>
      <start_time>2013-03-29T16:46:13Z</start_time>
    </execution_environment>
  </creator>
  <source>
    <image_filename>SampleImage.aff</image_filename>
  </source>
```

Figure 1: Sample DFXML output produced by *fiwalk*.

The DFXML file produced by *fiwalk* includes entries for each volume, partition, and associated file system (for those partitions that contain file systems). For each file system, a set of *fileobjects* corresponding to all of the files and directories identified is reported. If individual files within a disk image are themselves preservation targets, one can map metadata from the associated *fileobject* entry within a technical metadata section to one's schema of choice. If the disk image itself is a preservation target, the reporting tools developed for BitCurator can aggregate the data into an overview of the image's content. These reports can be generated several ways; as human-readable and editable documents; as part of a METS technical metadata section describing the file system(s) and formats of files identified; and as graphical visualizations.

The approaches described here – preserving individual files extracted from a disk image, and preserving the image itself – are not mutually exclusive, and collecting institutions may wish to employ both methods.

2.3 Forensic analysis metadata

The BitCurator environment incorporates Simson Garfinkel's *bulk extractor* software to identify and report on "features" (specific sequences of characters or bytes within the bitstream) contained within a disk image or live file system. Instances of these features are recorded by scanning modules designed to identify specific patterns in the raw bitstream of the disk image, including those that may correspond to potentially private, personally identifying, and sensitive information [10].

Post-processing of the feature files produced by *bulk extractor* generates a series of text files linking each individual feature at an absolute byte offset into the disk image with a specific file where the feature appears (or indicating that the feature appears in an area currently unallocated by a file system).

Institutions can use the output of *bulk extractor* (and associated processing scripts) to make decisions about processing and potentially redacting disk image content. If documentation of due diligence in identifying sensitive information on a disk is a high priority, or if a repository wants to provide cross-drive descriptors and access points (e.g. all email addresses that appear on the disks within or across collections), the repository can choose to retain some or all of the *bulk extractor* feature reports. This may often not be warranted, as the files are often large (hundreds or thousands of lines) and include sequences of escaped characters in non-ASCII encodings including UTF-8. If the disk image itself is the preservation target, one could opt to generate features reports as needed in the future. In either case, the *events* associated with the production of *bulk extractor* reports can be used to record the process by which curatorial decisions (and subsequent actions) about a disk image are made. In cases of redaction (for example, when private information needs to be removed from a publicly-accessible version of the materials), such documentation can be used to provide a precise account of the nature and number of redacted and their locations.

3. METADATA MAPPING

In order to support preservation and access activities for disk images within archival workflows, we are developing mappings from metadata elements encoded in DFXML to a range of metadata schemas including PREMIS, METS, and EAD. In the following sections, we discuss how this work is supported by specific digital forensics tools incorporated into BitCurator, along with our evolving model for recording digital forensics preservation actions.

3.1 Creating PREMIS metadata for disk images

When working with tools to create and process forensic disk images, the user may wish to treat the disk images solely as intermediate products in identifying, extracting, and repackaging individual file items (which then become uniquely defined archival objects). In other situations, the user may wish to treat the disk image itself as the primary object to be preserved. For the purposes of this work, we focus on the situations in which the disk image itself is the main preservation target. In practice, the two cases are not mutually exclusive. It is often desirable to retain both the full disk image and extracted copies of files that were stored on the disk.

Forensic tools support the capture and analysis of two types of metadata relevant to the born-digital lifecycle, specifically with respect to ongoing access and preservation. These activities are primary candidates for the creation of PREMIS metadata events.

First, metadata produced by forensic tools includes information about the physical source from which the disk image is extracted, providing important context about the creation environment. This may include manufacturer information, a serial number, and other hardware specifications. This information may be of interest to future users for historical purposes, and may also assist in supporting future access.

Second, this metadata may describe forensic actions performed prior to submission of a disk image to a repository (including analysis and triage tasks), or produced by the use of forensic tools on disk images contained within archival packages.

We have developed a set of PREMIS objects and events associated with extracting and processing disk images from physical media. Each preservation event is linked to a specific software tool that can be executed by a user of BitCurator. The objects, events, and encodings are described in the following sections.

3.1.1 PREMIS object encoding

PREMIS objects capture significant technical properties about digital objects. A disk image extracted from a physical medium can be treated as the instantiation of the preservation object (P-Object1). It is assigned an *objectIdentifier* with a universally unique identifier (UUID) value generated locally The disk image is described at the file level in accordance with the PREMIS data model, which states that files can be read, written, and copied, and have names and formats [17]. At the time of writing, records of forensic disk formats are absent from format registries such as PRONOM, but the most common formats – including the Expert Witness Format and the Advanced Forensic Format – are well documented online and have mature, robust software access libraries.

The PREMIS container *objectCharacteristics* can be used to capture significant technical properties about digital objects. In our mapping, we use the semantic unit *fixity* to record cryptographic hashes including MD5 and SHA1. These hash values are typically verified prior to ingest to ensure the integrity of the disk image, and to avoid inadvertent alteration and detect bitrot during ongoing preservation actions. PREMIS requires that an object's file format be identified either through the use of *formatDesignation* (containing *formatName* and *formatVersion*) or *formatRegistry* (containing *formatRegistryName* and *formatRegistryKey*). A file format registry can be used to validate formats. Selecting either *formatDesignation* or *formatRegistry* is a local implementation decision based on existing resources and workflow. However, the value of *formatDesignation* can be mapped directly using the metadata produced by Guymager, the open source forensic imaging tool incorporated into BitCurator.

We use the semantic unit *creatingApplication* to capture information about the environment in which the disk image is created. The output from Guymager is processed to extract specific technical details about the creation process, including tool version and time of image creation.

Events and relationships in the digital object lifecycle are also recorded using PREMIS. In order to acknowledge that P-Object1 was created by a specific event, one can use *linkingEventIdentifier* to record the link between the preservation event and the created object. Depending on local repository policies, the location of the original physical media can also be described using *storage/Contentlocation*.

If a repository decides to redact sensitive information from P-Object1, they can create a second PREMIS Object (P-Object2). The redaction tool iredact.py creates a new redacted version of a disk image (P-Object2). The redaction tool output records technical details that are used to describe P-Object2, including fixity information and file format types. We have also mapped tool output to *creatingApplication* to capture details about the image creation environment. It will often be advisable to retain P-Object1 for preservation purposes and make P-Object2 available for public access.

P-Object1 and P-Object2 differ in their relationships. P-Object1 is associated with a specific event (disk image capture) but no other PREMIS objects. P-Object2 can be related to P-Object1 using *relationshipType*, with the input value "derived" and *relationshipSubtype* "derived from." The *relationship/ relatedObjectIdentification/ relatedObjectIdentifierType* type can be used to record the UUID of P-Object1 and the *relationship/relatedEventIdentification* to record the UUID of the redaction event.

3.1.2 PREMIS Preservation Event Encoding

Using the information described in the previous section, we are modeling a set of PREMIS events that capture preservation activities performed on disk images. In this section, we describe five of these preservation events: imaging, file system analysis, feature analysis, report generation, and redaction. We describe each event in turn, along with encoding recommendations for integrating the output of the associated BitCurator tool into an existing repository implementation. Technical details that persist across events (such as unique identifiers assigned by local repositories) are described only in Event 1.

Event 1: Imaging

In the *Imaging* event, the disk image is extracted from the original media source. The event records metadata produced by a capture tool such as Guymager in one of the available forensic formats. One can identify the event by assigning it a unique identifier produced by the local repository. One can then describe the eventType as input value "capture" and use the timestamp produced by Guymager to map to *eventDateTime.* The *eventDetail* is used to record specific features of Guymager, including tool version, compilation timestamp, and associated library dependencies. The *eventOutcome* consists of two possible values: "Image created and verified" or "Image creation failed." The format of the newly-created disk image is mapped to *eventOutcomeDetail* (either .e01 or .aff).

Event 2: File System Analysis

The *File System Analysis event* describes the extraction of the file system(s) from the raw or forensically-packed image. The *file system analysis* event incorporates output from the *fiwalk* tool. We describe the *eventType* as "file system analysis." The BitCurator environment parses the XML file produced by *fiwalk* to capture specific details of the event. As an example, *eventDateTime* records the time of file system analysis and e*ventDetail* stores specific information about fiwalk. The results of the event, either "file system(s) analyzed" or "failed to identify file system(s)" are mapped to *eventOutcome.* Note that for disk images containing more than one partition, a partial analysis outcome is possible.

Semantic unit	Semantic component	Example value(s)	Derived from
eventidentifier	**eventidentifierType**	**UUID**	**N/A**
eventidentifier	**eventidentifierValue**	**8jb50321-6d7b-4291-89ag-a8b0fhc1f276**	**N/A**
eventType	**none**		
eventDateTime	**none**	**2013-03-29T16:46:13Z**	**Report.xml -> Start Time**
eventDetail	**none**	**version="bulk extractor 1.3.1"**	**from Report.xml -> Program, Version, SVN_Version, Compiler**
eventOutcomeInformation	**eventOutcome**	**report generated; report not generated**	
eventOutcomeInformation	**eventOutcomeDetail**	**Log output of reporting tool**	
linkingAgentIdentifier	**linkingAgentIdentifierType**	**preservation system**	
linkingAgentIdentifier	**linkingAgentIdentifierValue**	**[name of preservation system]**	
linkingAgentIdentifier	**linkingAgentIdentifierRole**		**Institution-specific**
linkingObjectIdentifier	**linkingObjectIdentifierType**	**UUID**	
linkingObjectIdentifier	**linkingObjectIdentifierValue**	**4bc90445-8d7b-8032-23cb-b7a2cah2e358**	

Table 1: Sample encoding of a *Feature Analysis* event using *bulk extractor*.

Event 3: Feature Analysis

The *Feature Analysis* event describes forensic analysis of the raw bitstream, which identifies features of interest to BitCurator users. This event incorporates those reports output by *bulk extractor*. Similar to Events 1 and 2, the repository assigns an event UUID. The *eventType* input value is "feature analysis." *EventDateTime* and *eventDetail* can be mapped from the <Execution Environment> section of the XML report produced by *bulk extractor*. An example of the relevant PREMIS encodings for the *Feature Analysis* event is provided in Table 1.

Event 4: Report Generation

The *Report Generation* event describes the collation and aggregation of intermediate forensic metadata into actionable, human-readable reports as PDF files or editable .xlsx files that may be used to inform additional preservation actions. The *eventOutcome* specifies the success or failure of report generation, which can include a string or integer representation of "none."

Event 5: Redaction

The *Redaction* event describes the process of eliminating potentially private and sensitive data from a disk image or copy thereof. In the BitCurator environment, the *iredact.py* Python script distributed with Simson Garfinkel's DFXML tools can be used to overwrite specific patterns within a disk image according to a rule set provided by the user. The *EventDateTime* and *eventDetail* are mapped from tool output. The *eventOutcome* specifies either "redaction completed" or "redaction not completed." In this event, *eventOutcomeDetail* records the full name of the newly created disk image, including its file format. One can also create an explicit relationship between the redaction event and the resulting preservation object (P-Object2) by using the UUID of P-Object2 in the linkingObjectIdentifierValue.

3.2 Encapsulating descriptive, administrative, and technical metadata for preservation

METS records metadata on acquisition, management, preservation, and access activities. Local METS profiles and tools used to process such metadata can vary significantly in coverage and functionality.

To support interoperability among institutions and existing collections, we are developing metadata export routines that encapsulate Digital Forensics XML produced by software such as *fiwalk*. These include the *fileobject* described in Section 2.2, automatically generated descriptive metadata, and general technical metadata about file systems encountered within disk images.

4. DISCUSSION AND FUTURE WORK

The metadata production and transformation methods described here are intended to supplement and enhance workflows organized around existing archival processing systems. As part of our ongoing work, we are continuously reviewing and responding to feedback from existing users of BitCurator, enhancing the capabilities of the environment, and streamlining tool implementations.

Proposed changes to version 3.0 of the PREMIS data dictionary will enhance lifecycle support for born-digital materials [4]. One proposal involves transforming the semantic unit *Environment* into its own entity (alongside *Object, Event, Agent, and Rights)*, enabling PREMIS to record and capture important metadata about the computing environment. Such enhancements aim to further enable rendering and deployment of preserved digital objects over the long term [5]. These proposed changes complement BitCurator's objectives by providing the necessary structure to preserve critical metadata describing original software environments.

5. CONCLUSION

We have detailed work conducted as part of the BitCurator project to develop strategies for transforming and incorporating metadata produced by digital forensics tools into preservation and archival metadata schemas. We have shown how metadata produced by open-source digital forensics tools can be encoded into PREMIS events and objects, and then packaged along with other archival metadata in XML format for long-term access and preservation in a repository setting.

6. ACKNOWLEDGMENTS

This research is being performed as part of the BitCurator project, funded by the Andrew W. Mellon Foundation.

7. REFERENCES

[1] AIMS Working Group. "AIMS Born-Digital Collections: An Inter-Institutional Model for Stewardship." 2012.

[2] Cohen, M., Garfinkel, S. L., and Schatz, B., Extending the advanced forensic format to accommodate multiple data sources, logical evidence, arbitrary information and forensic workflow, Proceedings of DFRWS 2009, Montreal, Canada, 2009.

[3] Dappert, A., PREMIS Tutorial: Understanding & Implementing the PREMIS Data Dictionary for Preservation Metadata. Presented at the PREMIS Tutorial, Rome, Italy, 2009. Retrieved 11 June, 2013 from http://www.loc.gov/standards/premic/premis-Rome-pt1.ppt

[4] Dappert, A., Proposed Data Model Changes for PREMIS 3.0, PREMIS Implementation Fair, October 2012. Retrieved 11 June, 2013 From http://www.loc.gov/standards/premis/pifpresentations-2012/PREMIS_Data_Model_Changes_final.pdf

[5] Dappert, A., Peyrard, S., Delve, J., and Chou, C., Describing Digital Object Environments in PREMIS, In *Proceedings of the Ninth International Conference on Digital Preservation (iPRES)*, Toronto, Canada, October 1-5, 2012, pp. 69-76

[6] Encase image file format, http://www.forensicswiki.org/wiki/Encase_image_file_format, Retrieved June 21, 2013.

[7] Garfinkel, S. L., AFF: A New Format for Storing Hard Drive Images, *Communications of the ACM* 49, no. 2, 2006), pg 85-87.

[8] Garfinkel, S. L., Digital Forensics Research: The Next 10 Years, Proceedings of DFRWS 2010, Portland, OR, August 2010

[9] Garfinkel, S. L., Digital Forensics XML and the DFXML Toolset, Digital Investigation 8, 2012, pg. 161-174

[10] Garfinkel, S. L., Digital media triage with bulk data analysis and *bulk_extractor*, Computers and Security, Volume 32, Feb 2013, pp. 56-72

[11] Garfinkel, S. L., "Providing Cryptographic Security and Evidentiary Chain-of-Custody with the Advanced Forensic Format, Library, and Tools." (International Journal of Digital Crime and Forensics 1, no. 1, 2009), pg. 1-28.

[12] Garfinkel, S. L., Lessons Learned Writing Digital Forensics Tools and Managing a 30TB Digital Evidence Corpus, Digital Investigation 9, 2012, pg. S80-S89.

[13] Gengenbach, M. J., "The Way We Do it Here" Mapping Digital Forensics Workflows in Collecting Institutions. Masters Paper for the M.S. in L.S degree. August, 2012.

[14] John, J. L., "Digital Forensics and Preservation", Digital Preservation Coalition, 2012.

[15] Kirschenbaum, M. G., Ovenden, R. and Redwine, G., "Digital Forensics and Born-Digital Content in Cultural Heritage Collections." (Council on Library and Information Resources, Washington, DC, 2010).

[16] Lee, C. A., Kirschenbaum, M. G., Chassanoff, A., Olsen, P., and Woods, K., BitCurator: Tools and Techniques for Digital Forensics in Collecting Institutions, D-Lib Magazine 18, No. 5/6, May/June 2012.

[17] PREMIS Editorial Committee, Data Dictionary for Preservation Metadata: PREMIS Version 2.2. July 2013, p.7.

[18] PREMIS Editorial Committee, Guidelines for using PREMIS with METS for exchange. Library of Congress.

[19] PREMIS Editorial Committee, Use of the Data Dictionary: PREMIS examples. Library of Congress. 2005.

[20] Archivematica, Metadata elements. *Archivematica:* https://www.archivematica.org/wiki/Metadata_elements 2005. Retrieved 13 June, 2013.

[21] Yale University Library Preservation Metadata Task Force, Yale Library, 2006. Retrieved 12 June, 2013 from http://www.library.yale.edu/cataloging/metadata/pmtf/tree.html

[22] Woods, K. and Lee, C. A., Acquisition and Processing of Disk Images to Further Archival Goals, Proceedings of Archiving 2012, Springfield, VA, Society for Imaging Science and Technology, pg. 147-152.

[23] Woods, K., Lee, C. A., and Garfinkel, S. L., Extending Digital Repository Architectures to Support Disk Image Preservation and Access, JCDL '11: Proceeding of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries, New York, NY, 2011, ACM Press, pg 57-66.