

# Identifying Barriers To File Rendering In Bit-level Preservation Repositories: A Preliminary Approach

Kyle R. Rimkus

University Library

University of Illinois at Urbana-Champaign

[rirkus\[at\]illinois\[dot\]edu](mailto:rirkus[at]illinois[dot]edu)

Scott D. Witmer

School of Information Sciences

University of Illinois at Urbana-Champaign

[sdwitme2\[at\]illinois\[dot\]edu](mailto:sdwitme2[at]illinois[dot]edu)

## ABSTRACT

This paper seeks to advance digital preservation theory and practice by presenting an evidence-based model for identifying barriers to digital content rendering within a bit-level preservation repository. It details the results of an experiment at the University of Illinois at Urbana-Champaign library, where the authors procured a random sample of files from their institution's digital preservation repository and tested their ability to open said files using software specified in local policies. This sampling regime furnished a preliminary portrait of local file rendering challenges, and thus preservation risk, grounded not in nominal preferences for one format's characteristics over another, but in empirical evidence of what types of files present genuine barriers to staff and patron access. This research produced meaningful diagnostic data to inform file format policymaking for the repository.

## Keywords

digital preservation; file format policy; random sampling

## 1. INTRODUCTION

File formats are important to digital preservation—but are they understood? Repository managers often require or recommend specific formats over others, believing that favored file varieties will give their digital content a better chance at long-term viability than the riskier alternatives. This practice comes with acknowledged limitations. As DeVorse and McKinney explain, "...files contain multifarious properties. These are based on the world of possibilities that the format standard describes, but can also include non-standard properties. The range of possibilities and relationships between them is such that it is quite meaningless to purely measure a file's adherence to a format standard" [4]. In other words, one ought to take endorsements of file formats in name only with a grain of salt, in lieu of better methods for representing the technical conditions necessary for the accurate rendering of digital content. This problem is explored in the Literature Review below, and is at the heart of the experiment presented in this paper.

## 2. LITERATURE REVIEW

As a young field, digital preservation is short on empirical evidence of file format risk, and most literature on the subject has been speculative in nature. In their 1996 report *Preserving Digital Information*, Waters and Garret suggested that repository managers faced with curating massive collections might adopt the practice of normalizing sets of heterogeneous file types to a smaller number of trusted formats [17]. Subsequently, repository managers and digital preservation researchers sought consensus on this approach, striving in particular to learn what qualities distinguish a trustworthy file format from an untrustworthy one.

Numerous studies, e.g., work conducted at the National Library of the Netherlands [12], Stanford University [1], and the Online Computer Library Center [15], strove to identify risk factors inherent to file formats. These research efforts, while

complemented by the dissemination of public file format recommendations by institutional repository managers [11], have not however led to consensus on what qualities make a file format unassailably good. For example, many practitioners favor open over proprietary file formats because the way they encode content is transparent and publicly documented. On the other hand, the broad adoption of a proprietary file format by an active user community tends to ensure ongoing software support, and therefore long-term accessibility, for the format in question. Thus, it isn't always clear whether a particular external factor will without doubt positively or negatively affect a file format's long-term viability.

Becker et al point out that the "passive preservation" of bit-streams, even in so-called trusted file formats, is most effective when complemented by permanent access to legacy software environments [2]. This point of view has been elaborated by David Rosenthal, who challenges the utility of file format risk assessment, emphasizing that genuinely endangered formats are often so obscure or proprietary that no known rendering software exists for them in contemporary operating systems. In such cases, Rosenthal advocates for bit-level preservation of endangered files along with their fully emulated rendering environments [13].

Recent research has encouraged a situational approach to managing file format risk in repositories. In her 2014 paper "Occam's Razor and File Format Endangerment Factors," Heather Ryan denigrates the term file format *obsolescence* in favor of *endangerment* "to describe the possibility that information stored in a particular file format will not be interpretable or renderable using standard methods within a certain timeframe" [14]. This line of thinking is shared by a British Library study of that same year which posits that academic fretting over whether file format obsolescence exists or not is irrelevant in practice: "Working on the assumption that data in the vast majority of file formats will be readable with some degree of effort does not take into account two crucial issues. Firstly, what is the degree of effort to enable rendering, and what does it mean for an organization...?" [8]. Or, as DeVorse and McKinney point out, risk assessment policies tend to stress the evaluation of potential external threats to digital files rather than the properties of the formats themselves: "At risk is not an inherent state of files and formats, it is an institution's view of its content determined by the policies, guidelines, and drivers it has at any one point in time" [4].

In a 2013 publication, an author of the present study found that the digital preservation file format policies of Association of Research Library member institutions were "very much rooted in relatively small-scale data management practices—stewarding files through digitization workflows, for example, or curating a university's research publications," but that, "As libraries and archives begin to set their sights on collections of heterogeneous files such as born-digital electronic records and research data, this is expected to spur on further evolution not only in the file formats that appear in digital preservation

policies, but in the way file format policies are articulated and implemented” [11].

There is however a dearth of studies investigating the capacity of organizations to identify and assess file format risk as it exists within their repositories. Holden conducted a 2012 sampling and analysis of files on archived web pages conducted at France’s Institut national de l’audiovisuel [5]. Similarly, Cochran published a report on file rendering challenges faced by the National Library of New Zealand [3]. In a similar vein, and influenced by concepts of organizational file format endangerment elaborated above, this paper seeks an evidence-based approach to assessing challenges to file rendering in bit-level preservation repositories.

### 3. BACKGROUND

In 2012, the University of Illinois at Urbana-Champaign (hereafter Illinois) Library established the Medusa digital preservation repository<sup>1</sup> for the long-term retention and accessibility of its digital collections. These consist primarily of digitized and “born digital” books, manuscripts, photographs, audiovisual materials, scholarly publications, and research data from the library’s special collections, general collections, and institutional repositories. All master files created by the library’s digitization units, for example, are by default deposited into Medusa.

Developed and managed locally by the Illinois library’s repository group<sup>2</sup>, Medusa features a web-accessible management interface, which provides collection managers with tools for initiating preservation actions. It provides forms for editing collection-level descriptive, administrative, and rights metadata; allows for the download of files or batches of files; tracks preservation events, file provenance, and file statistics; and provides on-demand verification of file fixity (md5 checksum values) and the extraction of technical metadata using the File Information Tool Set<sup>3</sup> (FITS) for files or groups of files. The library manages Medusa file storage in partnership with the National Center for Supercomputing Applications, also located on the Illinois campus. Medusa’s storage infrastructure consists of two copies of every file replicated daily across two distinct campus nodes, both on spinning disk, and a third copy of every file backed up and stored out of state on magnetic tape.

As of March 23, 2016, the Medusa repository houses 8,209,807 files requiring just over 60 terabytes of storage space (180 if one takes into account all three copies). These files are predominately in image formats, but also feature a significant number of text, audio, and video files, also in a variety of formats.

The variegated nature of digital content housed in Medusa stems from the many departmental libraries, special collections units, scholarly communication initiatives, and grant-funded digitization projects the repository serves. Its collections derive however from five key areas of focus. The first three of these, which began in earnest in 2007, are: 1) the largescale digitization of books, newspapers, and documents, both in-house and in partnership with external vendors; 2) the digitization of special collections manuscript content conducted on-site or with vendors; and 3) the deposit of scholarly publications and other materials related to teaching and learning into the Illinois Digital Environment for Access to Learning and

Scholarship (IDEALS)<sup>4</sup> institutional repository. The other two areas of focus, which began gathering momentum in 2012, are: 4) the acquisition of born digital electronic records in the University Archives, and 5) the digitization of audio and moving image content from the special collections undertaken on site or by vendors (see Table 1).

**Table 1. Approximate distribution of content source in Medusa repository by size**

Source	Size (TB)
Digitized books, newspapers, documents	39
Digitized manuscripts, photographs, maps	10
Digitized audio and video	8
Born digital electronic records	2
Institutional repository (self-deposit)	1
<b>TOTAL</b>	<b>60</b>

Medusa does not at present enforce file format validation or normalization on ingest. While Medusa managers acknowledge these as best practices, they have sought, in their initial phase of provisioning a preservation repository, to focus on collection-level control of their holdings, stable storage, and bit-level services such as fixity monitoring and file format identification. Prior to the existence of the Medusa digital preservation service, collection curators at Illinois had stored archival master files on a variety of storage media, many of them precarious. These included optical disks, portable hard drives, and file servers without consistent backup. Having taken custody of more than 8,000,000 files in Medusa’s first four years of existence, its managers are now interested in answering the following question: What are the most prevalent barriers to file access for curators and patrons who try to open files in Medusa’s collections?

## 4. METHODOLOGY

### 4.1 Medusa Feature Development

According to specifications provided by the authors, developer Howard Ding introduced three new features in the Medusa web application to enable data collection and analysis:

1. Testing Profiles
2. Random Sampler
3. File Tester

#### 4.1.1 Testing Profiles

The authors created a Testing Profile<sup>5</sup> to specify rendering conditions for each file format tested. Every Testing Profile listed a particular set of known extensions and MIME type values for a given file format. In addition, it specified the software, software version, operating system environment, and operating system version the authors would use for testing.

In identifying operating system and software values, the authors gave preference to tools deployed on site for library staff and users. Illinois Library Information Technology presently supports the Windows operating system for the majority of its employees, and web logs show that most library patrons also use Windows to access library resources. During the testing period, the operating system version of choice—for library staff and many patrons, and thus for this experiment—was Windows 7. The research goal being to assess file format challenges within the local access environment, this ensured results of practical relevance to collection curators and the communities they serve.

<sup>1</sup> <https://medusa.library.illinois.edu/>

<sup>2</sup> Source code for the Medusa collection registry application and its integrated microservices is available on Github (<https://github.com/medusa-project>).

<sup>3</sup> <http://projects.iq.harvard.edu/fits/home>

<sup>4</sup> <https://www.ideals.illinois.edu/>

<sup>5</sup> Go to [https://medusa.library.illinois.edu/file\\_format\\_profiles](https://medusa.library.illinois.edu/file_format_profiles) for a full list of current profiles.

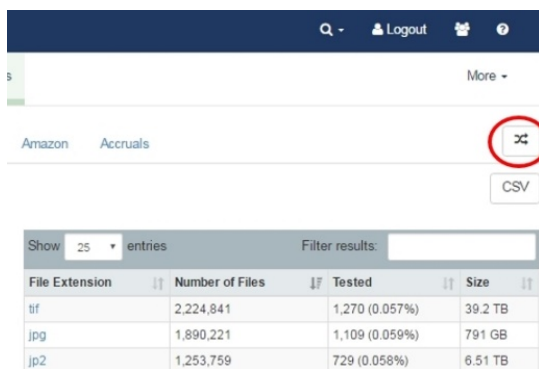
As an example, the profile for the format “TIFF” reads:

**TESTING PROFILE: TIFF**  
Software: Adobe Photoshop  
Software Version: CC2015  
OS Environment: Windows  
OS Version: 7  
MIME types: image/tiff  
File Extensions: tif, tiff

The authors emphasize that their approach to defining “file formats” in relation to these Testing Profiles constitutes a shorthand, and that the format standards under analysis can frequently take many forms. However, the use of such shorthand was deemed suitable to the purpose of this study.

#### 4.1.2 Random Sampler

The Random Sampler provided the authors, at the click of a button, a file selected randomly from the repository for testing.



The screenshot shows a dashboard with a table of file statistics. The table has columns for File Extension, Number of Files, Tested, and Size. The 'Random Sampler' button is circled in red.

File Extension	Number of Files	Tested	Size
tif	2,224,841	1,270 (0.057%)	39.2 TB
jpg	1,890,221	1,109 (0.059%)	791 GB
jp2	1,253,759	729 (0.058%)	6.51 TB

Figure 1. Medusa dashboard file statistics view (Random Sampler button circled in red)

#### 4.1.3 File Tester

The File Tester provides an interface for logging the success or failure of attempts to open files according to Testing Profiles. Specifically, it logs the operator, the date of the test, the Testing Profile in use, whether the test passed or failed, notes pertinent to the examination, and, in the case of failure, the reason why.

### 4.2 Testing Steps

The authors followed the steps below to gather data for this study:

1. Navigate to Medusa “dashboard” and press Random Sampler button (Figure 1)
2. Run technical metadata extraction tool File Information Tool Set (FITS)<sup>6</sup> on randomly selected file
3. Download and open file according to its corresponding Testing Profile
4. Fill out Analysis form with results of test (Pass/Fail, with reason for failure logged)

The authors assigned the status “Pass” to files that opened in the software program specified by their format profile without

<sup>6</sup> During testing, Medusa ran FITS version 0.8.3. FITS itself runs several metadata extractors such as Jhove (<http://jhove.sourceforge.net>) and DROID (<http://www.dcc.ac.uk/resources/external/droid>). FITS fields that accompany the full test data set are too numerous to list, but include PRONOM value, MIME type, file format name, file size, and last-modified-date.

apparent rendering problems. If problems were apparent, they assigned the status “Fail,” and appended a reason for the failure to the test record.

A sample test result reads:

**FILE TEST: 00000004.jp2**  
**UUID:** 714621f0-5cb8-0132-3334-0050569601ca-f  
Tester Email: email@illinois.edu  
Date: 2015-12-08  
Testing Profile: JPEG2000  
Status: Fail  
Notes: Renders in Kakadu, but not in Photoshop.  
Test Failure Reasons: Software's file format module cannot parse the file

#### 4.2.1 Constraints on Pass/Fail Criteria

Given the “multifarious” properties of computer files, a binary pass/fail distinction when evaluating files is no simple proposition. For this reason, the authors placed constraints on evaluations for several types of files:

- Files that clearly required ancillary files to execute, such as HTML documents that depend on image files or CSS stylesheets to render as intended, were evaluated on whether they opened as plain text.
- Programming or scripting files authored in plain text were tested as text files; they were not tested to see if the code they contained executed properly.
- Certain files deemed “unreadable” out of context of the associated files in their directory were considered to pass if they opened; for example, single-frame AVI files isolated from sequence.
- Package files, such as ZIP, passed if the package opened. The package contents were not tested.

### 4.3 Testing Timeline

The authors conducted testing over a five-month period from October 12, 2015 to March 23, 2016. The second author had a 13 hour per week appointment to the project, and conducted 97% of all initial tests. Prior to finalizing results, the primary author verified all files identified with status “fail” with the exception of those in the JPEG 2000 format (explanation to follow). During testing, ingest into the Medusa repository continued uninterrupted. The final population size reflects the number of files in Medusa on the final day of testing.

## 5. RESULTS

### 5.1 Overview

The authors tested 5,043 randomly sampled files<sup>7</sup> from a population of 8,209,807 (the population constituted the totality of files then housed in the Medusa repository). Statistically, this ensures to within a 2% margin of error and a 99% confidence level that the results are representative of repository-wide file format risk. Results, however, are not valid to within the margin of error for subpopulations of specific file formats. For example, the repository houses approximately 1.9 million files in the JPEG format (about 23% of all files), and indeed, approximately 1,141 files (about 22% of the sample set) were tested against the JPEG testing profile, ensuring a 4% margin of error for JPEG results at the desired 99% confidence level. On the other hand, the repository houses about 13,500 audio files with the format WAV (0.16% of all files), and tested 9 (0.18% of sample), meaning that the results are only valid to within a

<sup>7</sup> This paper presents snippet tabular views of project data; a comprehensive data-set is available at <http://hdl.handle.net/2142/89994>.

43% margin of error for the repository’s WAV files. While a future phase of research will focus on intensive testing within data strata such as file formats of interest, the authors acknowledge the limitations inherent to a purely random sample in this paper’s results.

As shown in Table 2 below, approximately 11% of files tested received a Fail status. While alarming at first glance, files failed to open for a variety of reasons, which are expanded on below.

**Table 2. Results of testing by pass or fail**

Status	Number	% of sample
Pass	4,479	89%
Fail	564	11%
<b>TOTAL SAMPLE</b>	<b>5,043</b>	<b>(100%)</b>

## 5.2 Triaging Results by File Format Profile

There isn’t a simple, programmatic way to triage test results by file format. One could sort by MIME type, PRONOM identity, or file format name, but these all represent different things. In the sample, FITS results show 47 MIME types, 67 PRONOM file formats (FITS reported no PRONOM value for 382 files, or about 8% of the sample set), and 77 file formats. However, the authors tested files against 93 Testing Profiles (see above), each one generally named after a file extension, and present these as the most consistent value for sorting data.

**Table 3. Pass/Fail status for ten most frequently occurring file formats in sample**

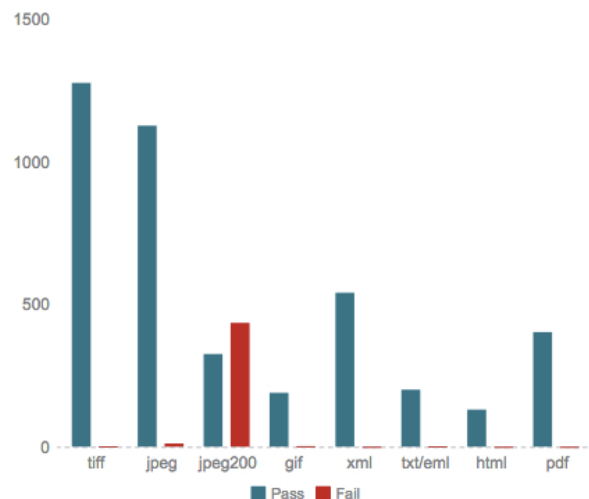
Testing Profile	Pass	Fail	Total Tested
TIFF	1276	1	1277
JPEG	1124	13	1137
JPEG2000	325	434	759
XML	540	2	542
PDF	402	0	402
GIF	192	3	192
HTML	130	0	130
TXT	114	0	114
EMLX	81	0	81
DOC	37	2	39

## 6. ANALYSIS

### 6.1 Files with Status Pass

Among files that passed muster, TIFF, PDF, and TXT performed especially well. 1276 out of 1277 TIFFs tested passed, as did all 402 PDFs and all 114 TXT files.

**Common Formats Tested**



**Figure 2. Pass/Fail for frequently occurring file formats in sample (visual representation based on Table 3)**

### 6.2 JPEG 2000 Files with Status Fail

The majority of failed tests (434 of 564, or 77% of all tests with status Fail) occurred for files in the JPEG 2000 format, the third-most common file format in the repository behind TIFF and JPEG. To understand what this failure rate represents, some background on JPEG 2000 at Illinois is necessary. In 2007, the library adopted JPEG 2000 as its file format of choice for high-resolution preservation master image files produced in monographic digitization efforts, primarily to benefit from storage gains that JPEG 2000 lossless compression promised over the uncompressed TIFF alternative. The potential for JPEG 2000 to become a trusted format for access and preservation image files had at that point garnered considerable traction in the library field [7], and Illinois’ then-preservation managers felt confident enough to prefer JPEG 2000 to TIFF.

Acting on this policy, Illinois contracted with an off-site vendor to both deliver page image files of digitized items in the JPEG 2000 format, and to create a set of scripts to support the output of JPEG 2000 files in locally managed digitization workflows. As a result, Illinois took custody of hundreds of thousands of page images produced externally and in-house from 2007-2014, all using a related set of scripts to generate JPEG 2000 files.

While these image files are viewable in certain software applications, they are considered corrupt by others. FITS data on 100% of failed JPEG 2000 files confirms them as well-formed and valid to the format standard, a status bolstered by informal spot checks of several files using the JPLYZER<sup>8</sup> tool. In addition, the problematic JPEG 2000 files are able to render in certain open-source image manipulation software applications like ImageMagick<sup>9</sup> and Kakadu<sup>10</sup>. However, many consumer-grade software applications cannot open them, with Photoshop in particular throwing the error: “Could not complete

<sup>8</sup> JPLYZER (<http://jplyzer.openpreservation.org/>) is a “validator and feature extractor for JP2 images” produced by the EU FP7 project SCAPE (SCalable Preservation Environments).

<sup>9</sup> <http://www.imagemagick.org/script/index.php>

<sup>10</sup> <http://kakadusoftware.com/>

your request because the file format module cannot parse the file.”

Experts in digital preservation have expressed concern that the nature of the JPEG 2000 standard would lead to this sort of problem. In 2011, van der Knijff wrote, “the current JP2 format specification leaves room for multiple interpretations when it comes to the support of ICC profiles, and the handling of grid resolution information. This has lead [sic] to a situation where different software vendors are implementing these features in different ways” [16]. While Illinois has not determined with certainty what variable differentiates its problematic JPEG 2000 files from those that open in Photoshop and other common software applications, it now knows that its repository houses hundreds of thousands of files that are unwieldy to many staff and patrons. The open source tools that can open these files without error are utilized primarily by specialists in file manipulation. They are not regularly employed by the library’s back-end users in its digitization lab or special collections units, nor by the scholars or graphic designers who frequently request image files from collection curators. When these users encounter such files, they most often find they cannot use them.

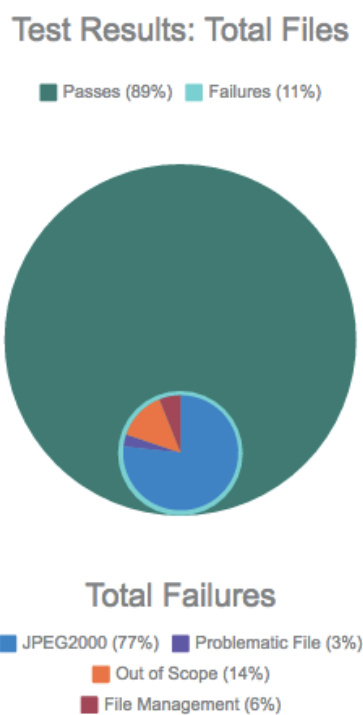


Figure 3. Percentage of Pass/Fail Test Status with Breakdown by Failure Type

### 6.3 Non-JPEG 2000 Failures

Files deemed to have failed to open according to their assigned profile did so for a variety of reasons, not all of which indicate file-format-based risk. In fact, by classing reasons for failure into groups *Out of Scope* (indicating they are not within the parameters of the testing regime), *Problematic File* (indicating the bit-stream itself is not readily openable), and *File Management* (indicating issues related to practices of naming and organizing files prior to their acquisition), the analysis below shows that only a small portion of non-JPEG 2000

failures are symptomatic of file format endangerment as it is generally understood.

#### 6.3.1 Overview of Non-JPEG 2000 Failures for Reason Out of Scope

78 of the 130 non-JPEG 2000 files flagged as failures represent varieties of bit-streams that, while unfit to be opened and evaluated as discrete entities, are nonetheless currently retained by the repository as essential to their collections. 48 of them fell into the category of *System file not within scope of current testing*. Formats with this result included APMaster, AUX, BAK, BIN, COM, DAT, DB, DLL, DS\_STORE, EMLXPART, FRF, FRM, FRX, ICM, LOCK, MYD, PFB, PLIST, SCR, SYS, and V. These are predominately system files, executable files, and auxiliary files such as those created by software during data compilation, and belong overwhelmingly to born digital electronic records acquired by the University Archives. Most system and auxiliary files in these formats are not meant to be opened by a human computer user. (Executable files, on the other hand, frequently represent items of interest to patrons, and shall provide the focus of a future phase of research).

12 files fell into the category *Auxiliary file created and used by a software program, not meant to be opened as individual file*. Most of the files with this result were in the FRDAT format produced by AbbyFineReader software. FRDAT is a proprietary file format used by AbbyFineReader in digital imaging and optical character recognition workflows at Illinois. The files have been retained with a significant number of digitized book packages, although their long-term utility merits question.

11 files were temporary files with underscores, tildas, or dollar signs in their names that are not meant to be opened. Many repositories delete such files on ingest, but Medusa administrators have at present not adopted this practice for deposits. Specifically, 9 files fell in the category *Not meant to be opened--Mac system file with underscore in name*, 1 file fell in the category *Not meant to be opened - temporary file with ~\$ in name*, and 1 file fell in the category *Not meant to be opened--software system file with @ symbol in name*.

Similarly, 5 bitstreams fell into the category *Not a file - artifact of disk formatting*. These bitstreams registered with Medusa as files, although with names like FAT1 and FAT2 and sizes of 1KB, they are clearly artifacts of formatting on storage devices accessioned in collections of born digital electronic records.

Finally, 2 files failed testing with the reason, *Software available on market, but testers have not yet acquired it*. One was in the SAV format containing binary statistical data for the SPSS<sup>11</sup> platform. The other was a TBK file, a proprietary electronic learning platform file for software called ToolBook<sup>12</sup>. While the software to open these files exists for purchase on the market, in neither case did the testers procure it in time for publication.

#### 6.3.2 Overview of Non-JPEG 2000 Failures for Reasons Related to File Management Practices

16 files fell in the category *No file extension*. Most of these were plain text files, frequently notes or works in progress, from collections of born digital personal records. Along similar lines, 2 files were appended with ad hoc file extensions and were given the failure reason *Not a file extension*. On closer inspection, these also turned out to be personal notes in collections of electronic records, where the depositor made up a file extension as a mnemonic device (e.g., authoring a text

<sup>11</sup> <http://www.ibm.com/analytics/us/en/technology/spss/>

<sup>12</sup> <http://www.sumtotalsystems.com/enterprise/learning-management-system/>

document about a colleague and giving it an extension with that person's initials). While these files do not indicate file format endangerment, they do pose certain challenges to curation.

2 files were *Saved with incorrect extension*, both for unknown reasons. One was a JPEG with extension 000, and the other was a Microsoft Word file with extension 2\_98, both of which files opened without a problem when appended with the correct extension. Both file formats were identified correctly by FITS.

More problematic are the 14 files that failed for the reason, *Despite file extension, file is in a folder designating it for another system purpose*. File formats with this result included GIF and JPEG—ostensibly image formats, although the files in question do not render as such, because they were created by a content management system for other purposes. Namely, numerous files from collections of born digital records acquired by the University Archives from former users of the FrontPage website authoring and management software contain files nested in a folder named "\_vti\_cnf". These software-generated folders contain files with the same names and extensions as JPEG and GIF files one level up in the directory hierarchy, but they are not in fact image files—rather, they were generated by FrontPage to keep track of versioning information of those files. Similarly, a JPEG file nested in folders called ".AppleDouble" indicate it to be a version tracking file used by an early Unix-like iteration of the Macintosh operating system. This "JPEG" does not render as an image file.

### 6.3.3 Overview of Failures for Reason Problematic File

18 non-JPEG 2000 files failed for reasons related to problematic file formatting.

13 failed for the reason, *Software considers file invalid*. 2 were JPEGs from the same collection of born digital electronic records, both with a last-modified-date in the year 2000. In attempting to open them, Photoshop provided the error: "Could not complete your request because a SOFn, DQT, or DHT JPEG marker is missing before a JPEG SOS marker." These files were generated by a little-known (though apparently still available) software called CompuPic(R)<sup>13</sup>. The other 11 files in this category have the WMZ extension, and appear to be compressed images from a slide presentation (the Windows operating system thinks they are Windows Media Player skin files, but some web research<sup>14</sup> shows that Microsoft Office software has used the WMZ extension for other purposes in the past; at present, testers have had no success opening WMZ files in the Medusa repository). The WMZ files in question were created in 2001, and also belong to a collection of born digital electronic records.

3 files failed for the reason, *File does not render in software*. Two are document files, one in the Microsoft Word DOC format, and the other in RTF. Embedded technical metadata in both files suggests they were created, at an indeterminate date, by an instance of Corel WordPerfect. Both files originate from a collection of born digital electronic records. The third file in this category is a GIF from a collection of born digital electronic records that appears to have been corrupt at the time of deposit, as it is in a folder of GIF files, and the others open without fail.

<sup>13</sup> A trial version is still available for download at <http://www.photodex.com/compupic>, but the software was created in 2003 and does not successfully install in the Windows 7 environment.

<sup>14</sup> <http://stackoverflow.com/questions/3523083/decompress-wmz-file>

**Table 4. Number of Test Failures by Reason and Type of Reason for all non-JPEG 2000 Failures**

Reasons	Reason Type	Total
System file not within scope of current testing	out of scope	48
Auxiliary file created and used by a software program, not meant to be opened as individual file	out of scope	12
Not meant to be opened—Mac system file with underscore in name	out of scope	9
Not a file—artifact of disk formatting	out of scope	5
Software available on market, but testers have not yet acquired it	out of scope	2
Not meant to be opened—software system file with @ symbol in name	out of scope	1
Not meant to be opened - temporary file with ~\$ in name	out of scope	1
<b>TOTAL OUT OF SCOPE</b>		<b>78</b>
No file extension	file management	16
Despite file extension, file is in a folder designating it for another system purpose	file management	14
Not a file extension	file management	2
Saved with incorrect extension	file management	2
<b>TOTAL FILE MANAGEMENT</b>		<b>34</b>
Software considers file invalid	problematic file	13
File does not render in software	problematic file	3
Software unavailable	problematic file	1
Software attempts to convert file to new version of format and fails.	problematic file	1
<b>TOTAL PROBLEMATIC FILE</b>		<b>18</b>
<b>TOTAL ALL CATEGORIES</b>		<b>130</b>

1 file failed for the reason, *Software unavailable*. This was in the format 411, a proprietary thumbnail image format for early Sony digital cameras, and originated from a collection of born digital electronic records.

1 file failed for the reason, *Software attempts to convert file to new version of format and fails*. This is a Corel WordPerfect WPD file that cannot be opened in the latest version of WordPerfect. It originated from a collection of born digital electronic records.

## 7. DISCUSSION

Success and failure rates reflected in this study's results do not necessarily bespeak the preservation viability of specific file formats over others. Frequently they reflect the practices of the community of users who produced them, or the circumstances under which they were created. For example, problematic files in the sample were often either produced using software that

never established a broad user base, or were output by one company's software but in a competitor's proprietary format (e.g. unreliable RTF and DOC files created by WordPerfect). In the case of perennially reliable file formats like TIFF, PDF, and TXT, however, a strong support system has emerged around them, with consistent software support across multiple operating systems.

## 7.1 JPEG 2000 Policy

In contrast to its TIFF holdings, the repository houses a number of JPEG2000 files (approximately 700,000, to extrapolate from the failure rate into the entire subpopulation of files with extension JP2) whose image bit-streams are intact, but whose file structure makes them inaccessible in common image management software. These files do not pose an immediate preservation risk, as it is well within the institution's ability to reformat them without loss [10]; rather, they pose a genuine access hurdle for many users.

Due to frustration with managing files in the JPEG 2000 file format as reflected in this research, the Illinois library has shifted its practices around the stewardship of preservation master files back to TIFF. The library, however, has not abandoned the JPEG 2000 format entirely—rather, it is limiting the scope of its use. Despite its drawbacks, JPEG 2000 has distinguished itself as particularly advantageous for online image presentation systems, thanks to the speed and efficiency with which web applications retrieve and render high-resolution JPEG 2000 images. In digital libraries, JPEG 2000 has found its home in the back-end of many image presentation systems, particularly those that serve millions of pages of library content online (both Chronicling America<sup>15</sup> and the HathiTrust Digital Library<sup>16</sup> rely on JPEG 2000 for serving page images). Likewise, the Illinois library is using JPEG 2000 as a back-end presentation format in its own locally managed digital image collections<sup>17</sup>, while retaining preservation master files for digital images in the TIFF format.

## 7.2 Born Digital Electronic Records

Electronic records make up only a small slice of Medusa's collections (about 2 TB out of 60), but their files are disproportionately represented in failed tests. The 52 non-JPEG 2000 files that failed testing for reasons of questionable *File Management* practices (34) and for the reason *Problematic File* (18) constitute 1% of the sample set, and originate overwhelmingly from collections of born digital electronic records. This suggests that the curation of born digital collections represents a hot spot, so to speak, warranting the attention of local preservation managers.

Collections of born digital electronic records acquired by the University Archives and collections of digitized collections from departmental libraries, however, often have different curatorial needs. In the sample, the authors discovered the 411 format used by an early Sony digital camera called the

Mavica<sup>18</sup>. Because proprietary rendering software for 411 files is presently unavailable without going to great lengths, the tested 411 file (created in 2002) was given a "Fail" status as unopenable. Some would say that such a file ought to be discarded on ingest and not retained at all—after all, if usable thumbnails are needed, they can be generated from the full-size image files stored in the same folder. However, the model name "Mavica" does not show up in any of the technical metadata for the full-size JPEG from which this thumbnail was derived, and the only way to know that this camera was used at all is *because* the associated thumbnail file with extension 411 was retained in the repository. From this perspective, the 411 file possesses potential research value. It provides evidence of the camera the person who took the photo used. It also demonstrates how an early digital camera platform generated thumbnail images. A technically useless file, it nevertheless provides historical context to the creation of other files in the collection, ensuring an unbroken "archival bond"<sup>19</sup> between bit-streams.

This suggests a need for different retention policies for different types of content within the repository. While curators of digitized monographs may look approvingly on disposing of "noise"—wiping the slate clean of artefacts of former image display software, system-generated files, and the like—an archivist may prefer a more conservative file retention policy for collections of born digital records, since these files may well provide insight into the creation and use of other files, or even help a researcher judge the authenticity of files as records.

## 7.3 Limitations of Methodology

The random sampling method, as employed by this study, poses certain limitations on the relevance of results to specific subpopulations of data, and implies the need for future work. The Medusa repository's collections originated from a variety of sources and workflows, some of which have produced more files than others. This means that image formats from book digitization efforts occurred much more frequently in the sample than audio formats from the library's nascent media preservation program, and that files from vendor-digitized general collections appeared with greater frequency than those from born digital special collections. By analyzing a random sample of files across a repository of highly disparate subpopulations of data, results provide an initial assessment of risk that is only statistically meaningful from a bird's eye view.

More importantly, the authors find the testing methodology described in this paper to be useful only as a blunt instrument for assessing barriers to content access. While other institutions may find a similar exercise useful, it is the authors' hope that their experiment will serve as a preliminary step toward elaborating a more sophisticated and effective means of assessment.

## 8. NEXT STEPS

Based on this study, the authors recommend that Medusa's digital preservation managers 1) isolate problematic JPEG 2000 files, particularly those that demonstrate high use, and remediate to TIFF format, and 2) devise an improved methodology for a follow-up study focused exclusively on collections of born digital electronic records, with an eye toward appraisal policy development and enhanced repository services for them.

## 9. CONCLUSION

The testing and analysis process detailed in this paper has forced Illinois preservation managers to identify and confront

<sup>15</sup> <http://chroniclingamerica.loc.gov/>

<sup>16</sup> HathiTrust (<http://hathitrust.org/>) relies on JPEG 2000 as a preservation format as well, but notably enforces formatting requirements on ingest, ensuring strict technical uniformity of all image files [10].

<sup>17</sup> Currently in beta at <https://digital.library.illinois.edu/>, the digital library utilizes the IIIF (<http://iiif.io/>) image interoperability framework, which allows for on-the-fly conversion and delivery of access derivatives in a variety of image formats to patrons, largely obviating the problem of keeping a single "master" file on hand in a readily accessible format to deliver to patrons in need.

<sup>18</sup> [http://fileformats.archiveteam.org/wiki/Sony\\_Mavica\\_411](http://fileformats.archiveteam.org/wiki/Sony_Mavica_411)

<sup>19</sup> <http://www2.archivists.org/glossary/terms/a/archival-bond#.V4P-4vkrJmM>

genuine problems curators and patrons face when attempting to open and use files stewarded in the Medusa repository. In the absence of similar studies, it is difficult to know whether Illinois' specific challenges are generalizable to those experienced by other institutions. Nevertheless, the testing method and findings presented here ought to prove useful to other researchers and managers interested in taking an evidence-based approach to assessing barriers to file rendering in digital preservation repositories.

## 10. ACKNOWLEDGMENTS

The authors would like to thank the University of Illinois at Urbana-Champaign Campus Research Board for generously supporting this project. In addition, they'd like to express their gratitude to Howard Ding for providing invaluable technical expertise and code development critical to its success.

## 11. REFERENCES

- [1] Anderson, R., Frost, H., Hoebelheinrich, N., & Johnson, K. (2005). The AIHT at Stanford University: Automated Preservation Assessment of Heterogeneous Digital Collections. *D-Lib Magazine*, 11(12), 10. <http://doi.org/10.1045/december2005-johnson>
- [2] Becker, C., Kulovits, H., Guttenbrunner, M., Strodl, S., Rauber, A., & Hofman, H. (2009). Systematic planning for digital preservation: evaluating potential strategies and building preservation plans. *International Journal on Digital Libraries*, 10(4), 133–157. <http://doi.org/10.1007/s00799-009-0057-1>
- [3] Cochrane, E. (2012). *Rendering Matters - Report on the results of research into digital object rendering*. Archives New Zealand. Retrieved from <http://archives.govt.nz/rendering-matters-report-results-research-digital-object-rendering>
- [4] De Vorsey, K., & McKinney, P. (2010). Digital Preservation in Capable Hands: Taking Control of Risk Assessment at the National Library of New Zealand. *Information Standards Quarterly*, 22 (2), 41–44.
- [5] Holden, M. (2012). Preserving the Web Archive for Future Generations. In *The Memory of the World in the Digital age: Digitization and Preservation* (pp. 783–795). Vancouver: United Nations Educational, Scientific, and Cultural Organization. Retrieved from [http://ciscra.org/docs/UNESCO\\_MOW2012\\_Proceedings\\_FINAL\\_ENG\\_Compressed.pdf](http://ciscra.org/docs/UNESCO_MOW2012_Proceedings_FINAL_ENG_Compressed.pdf)
- [6] Illinois Digital Environment for Access to Learning and Scholarship. (n.d.). FormatRecommendations. Retrieved July 30, 2013, from <https://services.ideals.illinois.edu/wiki/bin/view/IDEALS/FormatRecommendations>
- [7] Kulovits, H., Rauber, A., Kugler, A., Brantl, M., Beinert, T., & Schoger, A. (2009). From TIFF to JPEG 2000?: Preservation Planning at the Bavarian State Library Using a Collection of Digitized 16th Century Printings. *D-Lib Magazine*, 15(11/12). <http://doi.org/10.1045/november2009-kulovits>
- [8] Pennock, M., Wheatley, P., & May, P. (2014). Sustainability Assessments at the British Library: Formats, Frameworks, and Findings. In *iPres2014: Proceedings of the 11th International Conference on Preservation of Digital Objects* (pp. 142–148).
- [9] Rieger, O. Y. (2008). *Preservation in the Age of Large-Scale Digitization: A White Paper*. Washington, D.C.: Council on Library and Information Resources. Retrieved from <http://www.bib.uh.edu/fileadmin/fdocs/pub141.pdf>
- [10] Rimkus, K., & Hess, K. (2014). HathiTrust Ingest of Locally Managed Content: A Case Study from the University of Illinois at Urbana-Champaign. *The Code4Lib Journal*, (25). Retrieved from <http://journal.code4lib.org/articles/9703>
- [11] Rimkus, K., Padilla, T., Popp, T., & Martin, G. (2014). Digital Preservation File Format Policies of ARL Member Libraries: An Analysis. *D-Lib Magazine*, 20 (3/4). <http://doi.org/10.1045/march2014-rimkus>
- [12] Rog, J., & Van Wijk, C. (2008). Evaluating file formats for longterm preservation. *Koninklijke Bibliotheek*, 2, 12–14.
- [13] Rosenthal, D.S.H. (2010). Format obsolescence: assessing the threat and the defenses. *Library Hi Tech*, 28(2), 195–210. <http://doi.org/10.1108/07378831011047613>
- [14] Ryan, H. (2014). Occam's Razor and File Format Endangerment Factors. In *iPres2014: Proceedings of the 11th International Conference on Preservation of Digital Objects* (pp. 179–188).
- [15] Stanescu, A. (2005). Assessing the durability of formats in a digital preservation environment: The INFORM methodology. *OCLC Systems & Services: International Digital Library Perspectives*, 21(1), 61–81. <http://doi.org/10.1108/10650750510578163>
- [16] van der Knijff, J. (2011). JPEG 2000 for Long-term Preservation: JP2 as a Preservation Format. *D-Lib Magazine*, 17(5/6). <http://doi.org/10.1045/may2011-vanderknijff>
- [17] Waters, D., & Garrett, J. (1996). *Preserving Digital Information: Report of the Task Force on Archiving of Digital Information*. The Commission on Preservation and Access and The Research Libraries Group. Retrieved from <http://www.clir.org/pubs/reports/pub63/watersgarrett.pdf>