SWGDE Establishing Confidence in Digital and Multimedia Evidence Forensic Results by Error Mitigation Analysis

Disclaimer:
As a condition to the use of this document and the information contained therein, the SWGDE requests notification by e-mail before or contemporaneous to the introduction of this document, or any portion thereof, as a marked exhibit offered for or moved into evidence in any judicial, administrative, legislative or adjudicatory hearing or other proceeding (including discovery proceedings) in the United States or any Foreign country. Such notification shall include: 1) The formal name of the proceeding, including docket number or similar identifier; 2) the name and location of the body conducting the hearing or proceeding; 3) subsequent to the use of this document in a formal proceeding please notify SWGDE as to its use and outcome; 4) the name, mailing address (if available) and contact information of the party offering or moving the document into evidence. Notifications should be sent to secretary@swgde.org.

It is the reader's responsibility to ensure they have the most current version of this document. It is recommended that previous versions be archived.

Redistribution Policy:
SWGDE grants permission for redistribution and use of all publicly posted documents created by SWGDE, provided that the following conditions are met:

1. Redistribution of documents or parts of documents must retain the SWGDE cover page containing the disclaimer.
2. Neither the name of SWGDE nor the names of contributors may be used to endorse or promote products derived from its documents.
3. Any reference or quote from a SWGDE document must include the version number (or create date) of the document and mention if the document is in a draft status.

Requests for Modification:
SWGDE encourages stakeholder participation in the preparation of documents. Suggestions for modifications are welcome and must be forwarded to the Secretary in writing at secretary@swgde.org. The following information is required as a part of the response:

a) Submitter's name
b) Affiliation (agency/organization)
c) Address
d) Telephone number and email address
e) Document title and version number
f) Change from (note document section number)
g) Change to (provide suggested text where appropriate; comments not including suggested text will not be considered)
h) Basis for change

Intellectual Property:
Unauthorized use of the SWGDE logo or documents without written permission from SWGDE is a violation of our intellectual property rights.

Individuals may not misstate and/or over represent duties and responsibilities of SWGDE work. This includes claiming oneself as a contributing member without actively participating in SWGDE meetings; claiming oneself as an officer of SWGDE without serving as such; claiming sole authorship of a document; use the SWGDE logo on any material and/or curriculum vitae.

Any mention of specific products within SWGDE documents is for informational purposes only; it does not imply a recommendation or endorsement by SWGDE.

Abstract

Digital and multimedia evidence forensic practitioners are confident in the ability of their methods and tools to produce reliable conclusions; however, they often struggle to establish their confidence on a scientific basis. Some forensic disciplines use an error rate to describe the chance of false positives, false negatives, or otherwise inaccurate results when determining whether two samples actually come from the same source. But in digital and multimedia evidence forensics, there are fundamental differences in the nature of many processes that can make trying to use statistical error rates inappropriate or misleading.

The key point to keep in mind is the difference between random errors and systematic errors. Random errors are characterized by error rates because they are based in natural processes and the inability to perfectly measure them. Systematic errors, in contrast, are caused by many different factors. In computer software, for example, an imperfect implementation can produce an incorrect result when a particular condition, usually unknown, is met. Digital forensics – being based on computer science – is far more prone to systematic than random errors.

Digital and multimedia forensics includes multiple tasks which, in turn, use multiple types of automated tools. For each digital and multimedia evidence forensic tool, there is an underlying algorithm (how the task should be done) and an implementation of the algorithm (how the task is done in software by a tool). There can be different errors and error rates with both the algorithm and the implementation. For example, hash algorithms used to determine if two files are identical have an inherent false positive rate, but the rate is so small as to be essentially zero.

Once an algorithm is implemented in software, in addition to the inherent error rate of the algorithm, the implementation can introduce systematic errors that are not statistical in nature. Software errors manifest when some condition is present either in the data or in the execution environment. It is often misleading to try to characterize software errors in a statistical manner since such errors are not the result of variations in measurement or sampling. For example, the hashing software could be poorly written and may produce the same hash every time an input file starts with the symbol "$."

The primary types of errors found in digital and multimedia evidence forensic tool implementations are:

- Incompleteness: All the relevant information has not been acquired or found by the tool. For example, an acquisition might be incomplete or a search does not identify all existing relevant artifacts.

- Inaccuracy: The tool does not report accurate information. Specifically, the tool should not report artifacts that do not exist, should not group together unrelated items, and should not alter data in a way that changes the meaning. Assessment of accuracy in digital and multimedia evidence forensic tool implementations can be categorized as follows:

    - Existence: Do all artifacts reported as present actually exist? For example, a faulty tool might add data that was not present in the original.

- Alteration: Does a forensic tool alter data in a way that changes its meaning, such as updating an existing date-time stamp (e.g., associated with a file or e-mail message) to the current date?
- Association: For every set of items identified by a given tool, is each item truly a part of that set? A faulty tool might incorrectly associate information pertaining to one item with a different, unrelated item. For instance, a tool might parse a web browser history file and incorrectly report that a web search on "how to murder your wife" was executed 75 times when in fact it was only executed once while "history of Rome" (the next item in the history file) was executed 75 times, erroneously associating the count for the second search with the first search.
- Corruption: Does the forensic tool detect and compensate for missing and corrupted data? Missing or corrupt data can arise from many sources, such as bad sectors encountered during acquisition or incomplete deleted file recovery or file carving. For example, a missing piece of data from an incomplete carving of the above web history file could also produce the same incorrect association.

- Misinterpretation: The results have been incorrectly understood. Misunderstandings of what certain information means can result from a lack of understanding of the underlying data or from ambiguities in the way digital and multimedia evidence forensic tools present information.

The basic strategy to develop confidence in the digital and multimedia evidence forensic results is to identify likely sources of error and mitigate them. This is done by applying tool testing and quality control measures as described in this document including:

- Tool Testing:
  - Determine applicable scenarios that have been considered in tool testing
  - Assess known tool anomalies and how they apply to the current case
  - Find untested scenarios that introduce uncertainty in tool results
- Sound Quality Control Procedures:
  - Tool performance verification
  - Personnel training, certification and regular proficiency testing
  - Written procedures in accordance with applicable organizational quality assurance procedures
  - Examinations should be documented utilizing applicable organizational quality procedures
  - Document deviations/exceptions from standard operating procedures
  - Laboratory accreditation

- Technical/Peer review
- Technical and management oversight
- Multiple tools and methods complement capabilities
- Awareness of past and current limitations
- Reasonableness and consistency of results for the case context

A more formalized approach to handling potential sources of error in digital and multimedia evidence forensic processes is needed to address considerations such as those in *Daubert*.

The error mitigation analysis involves recognizing potential sources of error, taking steps to mitigate any errors, and employing a quality assurance approach of continuous human oversight and improvement. Rather than focusing only on error rates, this approach considers all the measures that can be taken to ensure that digital and multimedia evidence forensics processes produce reliable results. When error rates can be calculated, they should be included in the overall error mitigation analysis.

Table of Contents

1. Purpose

The purpose of this document is to provide a process for recognizing and describing both errors and limitations associated with tools, techniques, and methods used to support digital and multimedia evidence forensics. This is accomplished by explaining how the concepts of errors and error rates should be addressed in digital and multimedia evidence forensics. It is important for practitioners and stakeholders to understand that digital and multimedia evidence forensic techniques and tools have known limitations, but those limitations have differences from errors and error rates in other forensic disciplines. This document proposes that confidence in digital and multimedia evidence forensic results is best achieved by using an error mitigation analysis approach that focuses on recognizing potential sources of error and then applying techniques used to mitigate them, including trained and competent personnel using tested and validated methods and practices. Sources of error not directly related to tool usage are beyond the scope of this document.

2. Background

Currently, digital and multimedia evidence forensics needs a more disciplined and structured approach to recognizing and compensating for potential sources of error in evidence processing. Digital and multimedia evidence forensics is a complex field that is heavily reliant on algorithms that are embedded in automated tools and used to process evidence. Weaknesses or errors in these algorithms, tools, and processes can potentially lead to incorrect findings. Indeed, errors have occurred in a variety of contexts demonstrating the need for more scientific rigor in digital and multimedia evidence forensics. There is concern in government bodies such as the U.S. National Academy of Sciences (NAS) and U.K. Home Office that digital forensics lacks scientific rigor. We therefore propose a disciplined approach to mitigating potential errors in evidence processing to reduce the risk of inaccuracies, oversights, or misinterpretations in digital and multimedia evidence forensics. This approach provides a scientific basis for confidence in digital and multimedia evidence forensic results.

What is an Error?

In science, the word *error* does not carry the usual connotations of the term *mistake* or *blunder*. Error in a scientific measurement means the inevitable uncertainty that attends all measurements. As such, errors are not mistakes; you cannot eliminate them by being very careful. The best you can hope to do is to ensure that errors are as small as reasonably possible and to have a reliable estimate of how large they are [**1**].

Error rates are used across the sciences to characterize the likelihood that a given result is correct. The goal is to explain to the reader (or receiver of the result) the confidence the provider of the result has that it is correct. Many forensic disciplines use error rates as a part of how they communicate their results. Similarly, digital and multimedia evidence forensics needs to communicate how and why there is confidence in the results. Because of intrinsic differences between the biological and chemical sciences and computer science, it is necessary to go beyond error rates. One difference between chemistry and computer science is that digital technology is constantly changing and individuals put their computers to unique uses, making it infeasible to develop a representative sample to use for error rate calculations.

Furthermore, a digital and multimedia evidence forensic method could work well in one environment, but fail completely in a different environment.

This document provides a disciplined and structured approach for addressing and explaining potential errors and error rates associated with the use of digital and multimedia evidence forensic tools/processes without regard to environment. This approach to establishing confidence in digital and multimedia evidence forensic results addresses *Daubert* considerations.

Note: terms used in this document are defined either in the SWGDE Digital & Multimedia Evidence Glossary or in standard references for statistics or computer science [**2**].

3. Error Mitigation Analysis in Digital Forensics

Mitigating errors in a digital forensics process begins by answering the following questions:

1. Are the techniques (e.g., hashing algorithms or string searching) used to process the evidence valid science?
2. Are the implementations of the techniques (e.g., software or hardware tools) correct and appropriate for the environment where they are used?
3. Are the results of the tools interpreted correctly?

Considering each of these questions is critical to understanding errors in digital and multimedia evidence forensics. The next three sections explain the types of error associated with each question. In the first section, *Techniques*, the basic concept of error rates is addressed along with a discussion of how error rates depend on a stable population. The second section, *Implementation of Techniques in Tools* addresses systematic errors and how tool testing is used to find these errors. The third section, *Tool Usage and Interpreting Results*, summarizes how practitioners use the results of digital and multimedia evidence forensic tools. This overall approach to handling errors in digital and multimedia evidence forensics helps address *Daubert* considerations.

Systematic and Random Errors

Error rates for many procedures can be treated statistically, however not all types of experimental uncertainty can be assessed by statistical analysis based on repeated measurements. For this reason, uncertainties are classified into two groups: the random uncertainties, which can be treated statistically, and the systematic uncertainties, which cannot [**1**]. The uncertainty of the results from software tools used in digital and multimedia evidence forensics is similar to the problems of measurement in that there can be both a random component (often from the underlying algorithm) and a systematic component (usually coming from the implementation).

3.1 Techniques

In computer science, the techniques that are the basis for digital processing includes copying bits and the use of algorithms to search and manipulate data (e.g., recover files). These techniques can sometimes be characterized with an error rate.

3.1.1 Error Rates

An error rate has an explicit purpose – to show how strong the technique is and what its limitations are. There are many factors that can influence an error rate including uncertainties associated with physical measurements, algorithm weaknesses, statistical probabilities, and human error.

Error rates are one of the factors described in *Daubert* to ascertain the quality of the science in expert testimony [**3**]. The underlying computer techniques are comparable to the type of science that is described in *Daubert*. Are the underlying techniques sound science or junk science? Are they used appropriately? In computer science, the types of techniques used are different from DNA analysis or trace chemical analysis. In those sciences, the technique or method is often

used to establish an association between samples. These techniques require a measurement of the properties of the samples. Both the measurements of the samples and the associations have random errors and are well described by error rates.

Differences between digital and multimedia evidence and other forensic disciplines change how digital and multimedia evidence forensics uses error rates. There are error rates associated with some digital and multimedia evidence forensic techniques. For example, there are false positive rates for cryptographic hashing; however, the rate is so small as to be essentially zero. Similarly, many algorithms such as copying bits also have an error rate that is essentially zero. See Appendix B, ***Section 2. Hashing*** and ***Section 3. Hard Drive Imaging***, for a discussion of error rates associated with hashing and copying.

3.1.2 Error Rates and Populations

There are other major differences between digital and multimedia evidence forensics and natural sciences-based forensic disciplines. In biology and chemistry-based disciplines, the natural components of a sample remain fairly static (e.g., blood, hair, cocaine). Basic biology and chemistry do not change (although new drugs are developed and new means of processing are created). In contrast, information technology changes constantly. New types of drives (e.g., solid-state drives) and applications (e.g., Facebook) can be radically different from previous ones. There are a virtually unlimited number of combinations of hardware, firmware, and software.

The rapid and significant changes in information technology lead to another significant difference. Error rates, as with other areas of statistics, require a "population." One of the key features of a statistical population is that it is stable, that is, the essential elements of the composition remain constant. This allows predictions to be made. Since IT changes quickly and unpredictably, it is often infeasible to statistically describe a population in a usable way because, while the description may reflect an average over the entire population, it may not be useful for individual situations. See sidebar for an example of this.

Deleted File Recovery Example

File fragmentation is significant to the performance of deleted file recovery algorithms. In general, the more fragmented the files, the harder it is to recover the original files. For conventional (magnetic) hard drives, the amount of fragmentation was governed by the size of the hard drive (which change rapidly as bigger drives are brought to market) and usage patterns (which change rapidly such as storing large amount of multimedia files or using new applications). The resulting complexity itself meant that it was very difficult to determine what performance could be expected for a given drive type or user. This then changed completely when solid state drives (SSDs) were introduced and became popular. They no longer optimize performance by keeping files contiguous, rather moving files to prolong storage cell life. Additionally, the drive may "clean" deleted material. These kinds of paradigm shifts in IT are common and sometimes have unknown effects on forensic tools.

In examining these two differences – 1) the virtually infinite number of combinations and 2) the rapid pace of change – it can be seen that error rates for digital and multimedia evidence forensics are different from other forensic disciplines. It is apparent that the error rate for many

techniques being close to zero would imply that the topic of errors is of no concern to the digital and multimedia evidence forensics profession; this is clearly not the case. Similarly, it is not useful to say that potential sources of error cannot be addressed because of the lack of a meaningful population.

In order to understand error meaningfully, it is necessary to look at digital and multimedia evidence forensic tools. The tools implement a variety of computer science techniques and are "where the rubber hits the road" in digital and multimedia evidence forensics. Errors in tools and their use can have a much more significant negative impact on a digital and multimedia evidence forensic process. The next section discusses these types of errors.

3.2 Implementation of Techniques in Tools

The kinds of errors that occur in tools are systematic errors, not the random errors generally associated with measurements. See earlier sidebar for an explanation of random and systematic errors. Digital and multimedia evidence forensic tools (e.g., software, hardware, and firmware) are implementations of techniques. Tools are known to contain bugs of varying impact. Bugs are triggered by specific conditions and result in an incorrect output. For example, a tool can have a bug that causes it to underreport the size of a hard drive leading to a partial acquisition.

Because software bugs are logic flaws, the tool will produce the same result if given the same inputs. (In some rare cases, not all inputs are known or reproducible, in which case the program output can vary from run to run.) The output is not random, even though it is wrong. These are the systematic errors. Appendix B has digital and multimedia evidence forensics-based examples showing the difference between the error rate of a technique and systematic errors of tool.

In order to address systematic errors in tools, one must draw on computer science and software engineering. Software engineering provides methods for testing software to ascertain if it does what it is supposed to do. *Software testing and validation is the primary method for mitigating the risk of errors in tools.* Software testing can never prove that a tool is always functioning correctly; however, good testing can lead to confidence that the tool is unlikely to fail within the situations for which it has been tested.

There is another situation – primarily within forensic imaging of hard drives – that can cause tools to give different, but acceptable, results when processing the same drive. While imaging a hard drive, tools might not be able to read bad sectors on a drive. Tools could skip varying amounts of readable sectors that surround the bad sector for performance reasons. The resulting forensic images of a given drive made by different tools can be different and will have different hash values. Neither the tools' differing strategies for imaging a hard drive with bad sectors, nor the resulting images that differ are errors. They are, instead, the result of basic limitations with reading failing hardware.

When searching for something, such as a keyword or type of file, it is possible that the tool will find things that are not relevant (false positive) or fail to find things that are (false negative). These are not errors in the colloquial sense of a mistake, but are a method to describe the limitations of the tool. Digital and multimedia evidence forensic tools are designed to report only

information that actually exists on the original drive, and not to report anything that does not exist. One of the goals of tool testing is to verify that this holds true.

3.3 Tool Usage and Interpreting Results

Even when a technique is properly implemented in a tool, the tool can be used improperly, leading to errors. Furthermore, misinterpretation of what certain information means can result from a lack of understanding of the underlying data or from ambiguities in the way digital and multimedia evidence forensic tools present information.

Another significant consideration related to the interpretation of results is assessing the quality of data that was reconstructed from deleted material or recovered in an unusual manner. Such data might be incomplete, mix data from multiple original sources, or have other problems. Technical/peer review and use of a second method are often needed to address the limitations of reconstruction and recovery.

The errors associated with the improper tool usage, misinterpretation of results, and human factors errors are beyond the scope of this document. They can best be addressed by sound management practices including training, proficiency testing, peer review, and best practices. Additional information is available in the *SWGDE-SWGIT Guidelines and Recommendations for Training* and the *SWGDE Model Quality Assurance Manual for Digital Evidence Laboratories*, Sections 5.2 and 5.9 [**4**] [**5**].

4. Error Mitigation Analysis

The field of digital and multimedia evidence forensics requires an approach to error analysis that goes beyond error rates, and addresses the broader scope of errors that are relevant to digital and multimedia evidence forensics. Digital and multimedia evidence forensics is best served by a framework that guides practitioners to state the sources of potential errors and how they were mitigated in a disciplined manner. This document presents an error mitigation analysis process that addresses each discrete digital and multimedia evidence forensic task/process to accomplish this. The analysis must be flexible enough to address the wide range of evidence types and sources. Mitigation techniques will not be able to address every potential situation and the resulting error mitigation analysis should clearly state this.

An error mitigation analysis must address the potential sources of error for each major process and document the mitigation strategies that were employed. A list of common mitigation strategies is described below. Appendix A provides three examples for how to accomplish this.

5. Error Mitigation Techniques

This section summarizes key error mitigation techniques. Appendix A includes three approaches for applying these as part of an Error Mitigation Analysis. Many of these activities are discussed in ISO/IEC 17025, *General requirements for the competence of testing and calibration laboratories* [**6**]. Effective implementation of these activities will reduce the risk of errors.

5.1 Tool Testing

Evaluation of a tool is usually conducted by testing it against known data to provide confidence that a given tool is working as expected. Testing has been demonstrated in computer science to be an effective method for revealing errors in tools. Testing provides confidence in multiple situations by eliminating known sources of systematic error.

The primary limitation of testing is that no amount of testing can prove that the tool is functioning correctly in all instances of its use. Even if all tests produce the expected results, a new test scenario could reveal unexpected results. In practice, the more testing of diverse test scenarios, the more confidence you have that the software works correctly.

Tool Testing

Tool testing focuses on how the tool performs in situations that it was designed to handle. If a tool is used in other situations, such as if anti-forensics tools have been used, additional testing or verification will be needed. The Computer Forensics Tool Testing program at NIST provides testing material including specifications, procedures, and test sets [**7**].

Another limitation of testing is that each version of a tool could have flaws that are unique to that version operating in a particular environment. As new operating systems, hardware, software, and protocols evolve and new applications emerge, tools are updated to address these new developments in IT. Tool testing is further challenged by the large number of variables related to the tool and environment in which it is used.

These issues relate directly to the discussion of populations in ***Section 3.1.2 Error Rates and Populations,*** and deciding how much testing is enough is an active area of research in computer science. The amount of testing often depends on the application of the software. For example, safety control systems for nuclear power stations are tested more rigorously than other non-life critical systems. Tools and functions that address the integrity of the evidence need to be tested more rigorously than functions that can be verified by alternative methods, including manual inspection.

5.2 Performance Verification

Performance verification refers to checking a specific tool in the environment in which it is used to ensure it can perform its given function. This is not a repetition of the in-depth tool testing already performed, but rather a quick check that the hardware has not failed, that a piece of software can interact with the environment in which it is run, or that new copies of tools that have been received are working. This may consist of running a subset of the tests from in-depth tool testing. See also *SWGDE Standards and Controls Position Paper* [**8**].

5.3 Training

Training in forensic processes in general and in the specific tool used mitigates the risk that the tool is used incorrectly. Per *SWGDE-SWGIT Guidelines and Recommendations for Training*, forensic practitioners should be trained on the tools they are using [**4**]. Formal training can include classes. Informal training can include review of tool documentation and on the job training. See also *SWGDE Proficiency Test Guidelines* [**9**].

5.4 Written procedures

Having written procedures mitigates risk by documenting the correct procedures so forensic practitioners can more easily follow them. Procedures can be updated to keep current with industry best practices, and to state the limitations of specific tools and in what situations they are unsuitable for use.

5.5 Documentation

Documentation mitigates errors by allowing for review of work performed and for supporting reproducibility. A forensic practitioner's work must be reviewable in a meaningful way, including repetition of the process to assess the reliability of the results. Following written procedures and documenting significant outcomes should cover the majority of a practitioner's work. It is also important to retain and review audit/error logs of digital and multimedia evidence forensic tools to assess whether they functioned properly or encountered problems. Thorough documentation is especially critical for situations not fully covered by standard operating procedures. When such exceptions occur, detailing the situation and how it was handled is essential for error mitigation analysis.

5.6 Oversight

Technical and management oversight of digital and multimedia evidence forensic processes mitigates errors by ensuring that practitioners are trained in the tools they are using, that tools are tested, that documentation is produced and that procedures are followed.

5.7 Technical/Peer Review

Technical/Peer review mitigates error by having another qualified forensic practitioner look for errors or anomalies in digital and multimedia evidence forensic results. This is especially important if there are novel techniques used or outcomes or findings are outside of expected results.

5.8 Use of Second Method

The use of a second method by the forensic practitioner mitigates errors by verifying results. Common second methods include:

- After acquiring a forensic image of a hard drive with a tested hard drive imager and write blocker, forensic practitioner uses cryptographic hashes to verify that evidence is unchanged
- Manual review of reconstructed files, such as from deleted file recovery or file carving
- Manual review of files identified by a hash as being part of a contraband collection
- Use of multiple tools such as virus scanners, which while providing similar functionality, work differently

Possibility of Multiple Tests

Since most digital and multimedia evidence forensic processes are non-destructive, it is possible to repeat most forensic processes as many times as necessary without "using up" the evidence. The forensic practitioner can use multiple techniques or repeat specific processes (including peer review) on copies of the evidence because the copies can be verified to be identical to the original.

5.9 Awareness of Past and Current Problems

Digital and multimedia evidence forensics is a rapidly moving field. Forensic practitioners can mitigate errors by staying current with problems discovered in their laboratory and elsewhere. There are several sources including vendor blogs, conferences, listservs, forums, professional publications, and peer reviewed journals. Before relying on a particular source, forensic practitioners should carefully consider the reliability of the information and, when feasible, verify the problem for themselves.

5.10 Error Rates

The use of error rates can mitigate errors by showing the limits of a technique. Many digital and multimedia evidence forensics techniques, such as copying and cryptographic hashing, have very small error rates.

Other techniques, such as file recovery, have error rates that are dependent on multiple conditions present on the media, which are often unique to that piece of media. Therefore, it is not advisable to state an error rate for such techniques as it not likely to be relevant. There are cases where an error rate can be determined but techniques require a method to establish a baseline and might only be able to be applied in specific circumstances [**10**].[1] Error mitigation for these situations must employ other techniques, such as use of a tested tool (that reveals the tools limitations) or use of a second method.

[1] An example of an error rate for a specific situation can be found in "An Automated Solution to the Multiuser Carved Data Ascription Problem" by Simson Garfinkel et. al.

5.11 Context/Consistency of Data Analysis

Context/Consistency Analysis mitigates error by checking that recovered or identified material makes sense. Does the data make sense in context? Is it in the expected format? For example, the tool purports to recover a JPEG file that further examination reveals is actually a PDF file.

5.12 Other

This is not an all-inclusive list of error mitigation strategies. Forensic practitioners should document and explain other strategies they employed.

6. Summary

Many processes in digital and multimedia evidence forensics have fundamental differences from those in other forensic disciples that make them unsuitable for error rate evaluations. As a result, relying solely on error rates is insufficient and potentially misleading as a method to address the quality of the science when applying *Daubert*-type factors to digital and multimedia evidence forensics. In general, assessing the reliability of scientific testimony goes beyond error rates to include whether results are the product of sound scientific method, whether empirical testing was performed, and whether standards and controls concerning the process have been established and maintained. Therefore, when applying *Daubert*-type factors to digital forensics, it is necessary to go beyond merely stating an error rate – it is necessary to perform a comprehensive error mitigation analysis that addresses potential sources of error and how they have been mitigated. Mitigation techniques will not be able to address every potential situation and the resulting error mitigation analysis should clearly state this.

Digital and multimedia evidence forensics is best served by a framework that guides practitioners to state the sources of potential errors and how they were mitigated in a disciplined manner. This document provides a disciplined and structured approach to recognizing and compensating for potential sources of error in evidence processing. This error mitigation analysis process involves recognizing sources of potential error, taking steps to mitigate any errors, and employing a quality assurance approach of continuous human oversight and improvement. This more comprehensive process for addressing error is more constructive to establishing the scientific rigor and quality of digital and multimedia evidence forensic results than merely seeking out an error rate.

In the face of ever changing technology, digital forensic practitioners can provide reliable results by continuing to apply and develop best practices that provide guidance for how to perform forensic processes across disparate technology landscapes. Best practices can include implementing an array of error mitigation strategies such as those listed above, the foundation of which includes competent personnel implementing tested and validated tools and procedures, and employing a quality assurance approach of continuous human oversight and improvement.

7. References

[1] John R. Taylor, *An Introduction to Error Analysis: The Study of Uncertainties in Physical Measurements*, 2nd ed. Sausalito, CA: University Science Books, 1997.

[2] Scientific Working Group on Digital Evidence, "SWGDE Digital & Multimedia Evidence Glossary," 2016. [Online]. https://www.swgde.org/documents

[3] Daubert v. Merrell Dow Pharmaceuticals (92-102), 509 U.S. 579 (1993). [Online]. http://www.law.cornell.edu/supct/html/92-102.ZS.html

[4] Scientific Working Group on Digital Evidence and Scientific Working Group on Imaging Technology, "SWGDE/SWGIT Guidelines & Recommendations for Training in Digital & Multimedia Evidence," 2010. [Online]. https://www.swgde.org/documents

[5] Scientific Working Group on Digital Evidence, "SWGDE Model QAM for Digital Evidence Laboratories," 2012. [Online]. https://www.swgde.org/documents

[6] *General requirements for the competence of testing and measurement laboratories*, ISO/IEC 17025:2005.

[7] National Institute of Standards and Technology (NIST). (2018, February) Computer Forensics Tool Testing Program (CFTT). [Online]. https://www.nist.gov/itl/ssd/software-quality-group/computer-forensics-tool-testing-program-cftt

[8] Scientific Working Group on Digital Evidence, "SWGDE Standards and Controls Position Paper," 2008. [Online]. https://www.swgde.org/documents

[9] Scientific Working Group on Digital Evidence, "SWGDE Proficiency Test Guidelines," 2015. [Online]. https://www.swgde.org/documents

[10] Simson L. Garfinkel, et. al., "An Automated Solution for the Multiuser Carved Data Ascription Problem," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 4, December 2010. [Online]. http://simson.net/clips/academic/2010.TFIS.Ascription.pdf

[11] The PC Guide. [Online]. http://pcguide.com/ref/hdd/geom/errorRead-c.html

[12] B. Carrier. (2003, September) Open Source Digital Forensics Tools: The Legal Argument. [Online]. http://www.digital-evidence.org/papers/opensrc_legal.pdf

[13] E. Casey, *Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet*, 3rd ed. Waltham, MA: Academic Press (Elsevier Inc.), 2011.

[14] E. Casey, "Error, Uncertainty, and Loss in Digital Evidence," *International Journal of Digital Evidence*, vol. 1, no. 2, Summer 2002.

[15] J. Elerath, "Hard-Disk Drives: The Good, the Bad, and the Ugly," *Communications of the ACM*, vol. 52, no. 6, June 2009. [Online]. http://cacm.acm.org/magazines/2009/6/28493-hard-disk-drives-the-good-the-bad-and-the-ugly/fulltext.

[16] E. E Kenneally, "Confluence of Digital Evidence and the Law: On the Forensic Soundness of Live-Remote Digital Evidence Collection," *UCLA Journal of Law and Technology*, 2005 UCLA JL & Tech 5.

[17] E. E. Kenneally, "Gatekeeping Out Of The Box: Open Source Software As A Mechanism To Assess Reliability For Digital Evidence," *Virginia Journal of Law and Technology*, 5 VA J.L. & Tech 13, Fall 2001.

[18] G. C. Kessler, "Judges' Awareness, Understanding, and Application of Digital Evidence," *Journal of Digital Forensics, Security and Law*, vol. 6, no. 1, 2011.

[19] S. L. Savage, *The Flaw of Averages: Why We Underestimate Risk in the Face of Uncertainty*. Hoboken, NJ: John Wiley & Sons, Inc., 2009.

Appendix A – Example Error Mitigation Analysis Reports

1. Purpose

The purpose of Appendix A is to provide several examples for what an error mitigation report might look like. The purpose is to provide sample language and sample structures for the reports.

There are 3 examples that are quite different from one another. The first is quite comprehensive and shows the full breadth of applying the error mitigation strategies.

The second example addresses a more specific situation and has a more focused error mitigation report.

The third is focused on addressing the use of a new technique within a forensic process.

It is expected that the reader will select from the examples to create a template that works well within their laboratory and is appropriate for the type of forensic process performed. The goal is to document and communicate the steps taken to reduce errors and expose areas where there is still a significant source of error. For example, the use of a non-tested tool should be obvious from an error mitigation report and would require additional explanation for why untested tools were used.

Appendix A – Example Error Mitigation Analysis Reports

Example #1

The case involves intellectual property theft and includes web-based email and cell phone analysis.

Report:

Confidence in the results from the cell phone analysis, including conspirator's contacts from the address book and text messages with conspirators that included references to new product development is based on:

- Use of a tested tool: The tool, MobileImager version XYZ, was tested by NIST and by the lab; however, NIST tested an earlier version and neither NIST nor the laboratory tested the model of phone in question, but both the NIST and the laboratory tests included other models from the same manufacturer. Testing showed that the tool could retrieve contact information and text messages. Anomalies found during testing were not relevant to this examination.
- Context Analysis: The tool returned well-formatted data.
- It is possible that not all contact information was recovered.
- Text message recovery is limited to what was still stored on the phone.
- Lab-based procedures, including training, documentation, and oversight, were followed.

Confidence in the results of the web-based email analysis, including identification of emails that contained company intellectual property being directed outside the company, is based on:

- Internet Tool ABC and Other Internet Tool DEF were used to acquire the email have been tested within the lab.
- Context analysis showed that the returned data was well formatted consistent with web-based email.
- Or: Context analysis showed that attachments were not returned. Only header information and the email message itself were returned but they were well formatted.
- It is possible that not all emails were discovered.
- Lab-based procedures, including training, documentation, and oversight, were followed.

Example #2

During the course of a forensic examination, a new technique is developed to address a particular aspect of the examination. The technique could be developed in-house or brought in from outside. This example addresses error mitigation strategies appropriate to this situation.

In this case, files had been deleted using a known wiping program. Normally, not only are the files not recoverable, but the wiping program removes any trace of the deleted files, file names, and of the tool's activity. The laboratory develops a technique to recover the deleted file names based on a journaling capability of the file system. In this example, it is important to determine what files the suspect possessed and then deleted. The resulting tool is called Zombie Resurrection.

- **Step 1:** Zombie Resurrection was used on a copy of the evidence and was able to find 50 file names for files that were not present on the drive.
- **Step 2:** Since it appears that Zombie Resurrection might be useful for finding deleted file names, Zombie Resurrection was tested.

A controlled test data set was created with known content. The controlled test data set used the same operating system as the evidence.

The known wiping tool was used on the controlled test set to delete 100 files.

Zombie Resurrection was used on the controlled test set. The result was that Zombie Resurrection produced a list of 75 file names that had been on the system, but the list did not include 25 file names. There were no file names included on the list that had not been on the system.

Zombie Resurrection was deemed to be effective for finding deleted file names but cannot be used to claim that the list provided is complete.

- **Step 3:** Documentation was written for Zombie Resurrection for both the use of the tool and for the testing performed.
- **Step 4:** Zombie Resurrection and its documentation were given to a colleague to test on a similar system. The colleague got consistent results as the initial test. Because Zombie Resurrection uses a straightforward technique, the colleague was able to understand how it works and was able to conclude that it was unlikely for there to be errors in the implementation using the tool for this situation.

Error Mitigation Report: The novel tool, Zombie Resurrection, was developed and tested in-house, documentation written and peer reviewed in-house by a competent forensic practitioner familiar with digital forensic tools and techniques. It is best practice to have tested tools that produce repeatable and reproducible results and to have peer review for new techniques.

Other error mitigation strategies will be needed if the tool is applied more broadly. Additional testing will increase confidence in the reliability of the results and its applicability to other environments.

Appendix A – Example Error Mitigation Analysis Reports

Example #3

In this case, digital and multimedia evidence forensics was used to find information about a criminal plot. One drive was imaged and deleted files were recovered.

This example uses a table to be filled in by the forensic practitioner to document the relevant error mitigation strategies that were employed. A brief discussion of the fields in the table is provided along with a table that has been filled in.

Fields:

Mitigation strategies that apply throughout should be noted up front. Only when there are exceptions should these overall strategies be discussed for each process. For example, if the operator were trained on 6 of the 7 tools used, that would only need to be noted when the 7th tool is discussed.

- **Techniques:** Describe the underlying computer science techniques or algorithms employed.
- **Tool:** List the tools used including all relevant versioning information
- **Techniques Mitigation Strategy:** Techniques could have relevant error rates. NIST will be providing analysis of error rates for common forensics techniques. Check www.cftt.nist.gov. Other sources of error rate information are valid to cite. If an unusual technique is employed, refer to relevant documentation and literature.
  - Since testing is a primary mitigation strategy, list what relevant test reports are available. Be sure that any referenced test reports are reviewed for problems or limitations encountered during tool testing that are related to the current forensic examination. If the specific version has not been tested, be sure to be clear about this. The other mitigation strategies that were used should also be listed. It will be helpful to take the generic strategies and state how they were applied in this examination. It will probably be helpful to state that the tool was or was not used according to its documentation and is appropriate for the given situation.
- **Findings:** List facts that show that the examination produced relevant findings and summarize any key issues related to error mitigation.

Appendix A – Example Error Mitigation Analysis Reports

Table 1. Documenting Error Mitigation Strategies Using a Table (Example #3)

Techniques	Technique Mitigation Strategy	Tools	Tool Mitigation Strategy	Findings
Write Blocking	The ability to block commands is well established in literature. See NIST report on write blocking	Writeblocker ABC, version 1.2.3	▪ Drive type is XYZ, which Writeblocker ABC supports. ▪ Tool has been tested by NIST and this version (including firmware) by our lab. The lab testing included the relevant operating environment. ▪ Hashing was used as a secondary verification.	Confidence is based on use of tested tools, secondary verification, and adherence to lab-based mitigation strategies.
Drive Imaging	The ability to copy content from drives is well established in literature. See X and Y. See NIST report on hard drive imaging.	Driveimager DEF, version 5.6	▪ Drive type is XYZ, which Driveimager DEF supports. Drive had HPA, which Driveimager DEF can acquire. ▪ Tool has been tested by NIST and this version by our lab. ▪ Hashes were verified. Operator has not been trained on Driveimager DEF, but is familiar with several other hard drive imaging programs.	Confidence is based on use of a tested tool and verification of hashes.
Deleted File Recovery (DFR)	The ability to recover files using metadata based tools is established. See NIST report on DFR testing.	Deleted File Recovery Tool GHI, version 7	▪ Drive contained NTFS file systems, which Deleted File Recovery Tool GHI can recover. ▪ Tool tested by NIST (provide reference) and found to be able to recover files if there is little fragmentation. ▪ There is a possibility that the tool will join file fragments from different files to recreate a recovered file.	Confidence is based on use of a tested tool and manual inspection of the files that contained relevant search terms to eliminate incorrectly recovered files and adherence to lab-based mitigation strategies.

Appendix B – Example Error Analysis for Selected Techniques

1. Purpose

The purpose of Appendix B is to show the relationship between the error rate of a technique and the systematic errors of an implementation. Several examples are presented.

An error rate is stated for an algorithm and an analysis of possible implementation errors with strategies for mitigation of the implementation errors.

The topics covered are:

- **Section 2. Hashing**
- **Section 3. Hard Drive Imaging**
- **Section 4. Hardware Write Blocker**
- **Section 5. File Recovery**

2. Hashing

Use of hashing in a forensic context is usually used to determine if a file has changed (e.g., image of a hard drive) or if a given file is exactly the same as some known file.

2.1. Hashing Algorithm Error Rates

Two types of errors that are possible are:

- Two files are the same but produce different hashes (false negative).
- Two files are different but produce the same hash value (false positive).

The design of the algorithm is such that it always produces the same result for the same input, so the false negative rate for the algorithm is zero.

Hash algorithms have a false positive error inherent in the algorithm design. The size (number of digits) of the hash value determines the false positive error rate. For example, consider a (not very useful) hash algorithm that computes a two-decimal digit hash value. If 101 unique files are hashed then there must be at least two files with the same hash value. In practice, hash algorithms are designed to have a vanishingly small false positive rate near zero. The MD5 algorithm computes a 128-bit hash value, i.e., 1 chance in 2^{128} of a given file having the same hash as another file chosen at random. The SHA1 algorithm is 160 bits with an even lower false positive rate.

2.2. Errors Implementing Hash Algorithms

The implementation of a typical hash algorithm has several sections, including a section to input the data to hash and a section to compute the hash value. Some possible errors and implications include:

- Computer code to do the hash calculation could be incorrect. This type of error is readily apparent by software testing with a few files with known hashes. Most likely all the hashes will be incorrect. Such a tool is defective and a different tool should be used. An error rate for this implementation would be 100%.

- The input section could change the data before passing the data to the program section that calculates the hash value. An example is that under certain conditions extra characters might be added by the operating system to the end of each line of text for text files. Such a tool incorrectly computes hashes for text files, but correctly computes hashes for other file types. This can be detected by software testing using a variety of file types including text files. Such a tool should not be used. An error rate for this tool could be calculated as a proportion of the text files relative to the total number of files. However, such a calculation would not be useful for any other case.

3. Hard Drive Imaging

Hard drive imaging is the acquisition of the digital contents of a secondary storage device.

3.1. Hard Drive Imaging Algorithm Error Rates

The basic algorithm for imaging a hard drive is:

1. Determine the size of the target device.
2. Read all readable data and save.

The algorithm for reading data and saving it incorporates error correcting codes, which prevent reading data incorrectly. It is called a miscorrection when the error correcting codes do not produce the correct data. Per *The PC Guide*: "A typical value for this occurrence is less than 1 bit in 10^{21} [11]. That means a miscorrection occurs every trillion gigabits read from the disk--on average you could read the entire contents of a 40 GB drive over a million times before it happened!" In other words, the algorithm has an error rate that is zero for all practical purposes. (See *Read Error Severities and Error Management Logic* on http://pcguide.com for a further explanation of reading hard drives [**11**].)

3.2. Errors Implementing the Hard Drive Imaging Algorithm

Implementation of hard drive imaging tool is vulnerable to many systematic errors. Some examples:

- The size of the hard drive is determined incorrectly by the operating system or storage device reporting a smaller than actual size to the tool. The tool then stops the acquisition before all data has been read. This error is usually a consequence of a change in storage device technology. Tool testing can be used to detect this problem by using test drives that are the most recent available in addition to a mix of older drives.
- The size of the hard drive is determined incorrectly if the tool ignores hidden areas. This is often an intentional tool design decision and not really an error. Tool testing can detect this behavior by including test drives that contain hidden areas. This behavior can be mitigated by checking for a hidden area before imaging; if hidden sectors are present, another tool or technique can be used to reconfigure the drive to unhide the hidden areas.

- Some imaging tools offer a feature to restore a previously acquired drive image to another drive. Some operating systems under report the size of hard drives to the tool. In such a situation, the tool will stop the restore before the entire image has been restored. Tool testing will detect this error by testing with a restore drive exactly the same size drive that was imaged. This can be mitigated by always using a restore drive larger by the underreported amount than the original.

4. Hardware Write Blocker

A hardware write blocker is a device used to connect a storage device to a computer that allows access to data storage device without altering the content of the device.

4.1. Write Block Algorithm

The basic write block algorithm is:

1. Intercept each command sent from the host to the storage device.
2. Examine the command function.
3. If the command could change content of the storage device, do not pass the command on to the storage device.
4. For other commands, pass the command on to the storage device.

The algorithm prevents any commands that can alter the content of the storage device being passed to the device. The error rate of the algorithm is zero; that is a perfect implementation would have no errors.

4.2. Errors implementing Write Blocking

Some errors that can occur are:

- Not all possible write commands are blocked. Such a device might appear to protect a device as long as the host computer uses one of the blocked commands and then silently fail if the host computer uses one of the other commands that are not blocked. Tool testing can detect such errors by transmitting all known commands from the host to the storage device through the write blocker. The commands not blocked will always write to the storage device. This allows identification of a potentially unsafe write blocker and selection of a safe write blocker.

5. File Recovery

Recovery of deleted files presents a tool user with a collection of recovered files, possibly with file sizes, names, and other recovered metadata. Some of the many possible recovery results are the following:

1. A deleted file is recovered completely along with the file name and other metadata. This is the ideal case.
2. A deleted file is recovered completely, but the file name and other metadata is not recovered. One situation when this happens is when some tools recover files from a Linux ext2 file system.
3. A deleted file is partially recovered sequentially from the first data block.
4. A deleted file is partially recovered sequentially not including the first data block.
5. A deleted file is recovered with some data blocks skipped. This scenario can lead to misinterpretation of results.
6. A deleted file is recovered with some data blocks assembled out of order. This scenario can lead to misinterpretation of results.
7. A recovered file contains data that was not present anywhere on the original drive. This would be a serious flaw in a tool; the tool has invented data.
8. A recovered file contains data that was not ever present in a file, active, or deleted. This would be another flaw in a tool; the tool has included data that may not have been created or used by the drive owner.
9. A recovered file contains data from multiple deleted files. This scenario can lead to misinterpretation of results.

These results occur as a result of the interaction of the data available, the recovery algorithm, and the algorithm implementation. Before an error rate can be discussed, the error to be measured must be defined. There are many possible errors that can be defined and usually more than one way to define an error in the context of deleted file recovery. Many of the results listed above are really the best that can be done under the limitations imposed on tools by the data available. For this discussion, all the results other than the first result are treated as errors in the sense that the result is not a complete, accurate reconstruction of the original deleted file.

Some examples of possible errors that can be defined:

1. **Multiple Source Error**: Recovered file is constructed from multiple sources.
2. **Size Error**: Recovered file is the wrong size. (The definition of the *right size* is not relevant for this example.)
3. **Gap Error**: There are one or more missing blocks between two recovered blocks.

Recovery is usually accomplished either by metadata based file recovery or by file carving. The algorithms used for each method are very different.

5.1. Metadata Based File Recovery

Metadata based deleted file recovery exploits storage device characteristics, operating system behaviors, and file system behaviors that do not overwrite file data and could leave enough metadata to locate at least some of the file data.

The actual deleted file recovery algorithm implemented by a given tool is often proprietary and not available for examination or analysis. However, the general approaches are well known and can be considered in light of known operating system behavior and limitations. A typical algorithm looks for metadata describing deleted files and then uses the metadata to locate the deleted data. As an example, consider the FAT file system.

5.1.1. FAT

When a file is deleted from a FAT file system, some metadata is immediately overwritten. The file entry is marked with a hex value of 0xE5. This overwrites the first character of one copy of the file name (However, there could be two copies of a file name: a DOS 8.3 name and a long file name. The first character of the DOS 8.3 file name is overwritten, but the long file name remains intact.). The metadata that locates the first block of data and the file size is preserved, but the metadata to locate the remainder of file blocks is cleared to zero. This establishes limits that any algorithm recovering files from a FAT file system:

- The first block, the file name and the file size can be recovered immediately after a file is deleted.
- The actual location of the remainder of the file is unknown. However, it is possible to make a guess about the location of the remainder of the file because the operating system tries to avoid file fragmentation by allocating file blocks contiguously. Consider four layouts of deleted files at the time of data acquisition:
    1. The file data blocks are contiguously allocated.
    2. A file is fragmented such that the fragments are sequential and separated only by blocks from allocated files.
    3. A file is fragmented such that the fragments are sequential and separated by blocks from either allocated files or other deleted files.
    4. Once other file system activity occurs, overwriting of both metadata and file data might occur.

5.1.2. Some Simple Recovery Algorithms

Here are three possible simplified algorithms for locating the remainder of file blocks when recovering files from a FAT file system:

- Include enough unallocated blocks following the first file block until the recovered file is the same size as in the deleted file metadata entry.
- Include enough blocks, regardless of allocation state, following the first file block until the recovered file is same size as in the deleted file metadata entry.
- Stop recovering after the first block.

The following table describes algorithm behavior in terms of the multiple source error defined above on each of the four data layouts.

Table 2. Algorithm Behavior by Data Layout

Algorithm	Layout			
	Contiguous	**Frag/Active**	**Frag/Deleted**	**Overwritten**
A	No error	No error	Multi source	Unknown*
B	No error	Multi source	Multi source	Unknown*
C	No error	No error	No error	No error

* If the original source were completely overwritten, from a single source, then the recovered file would be from a single source. If the original source were partially overwritten, then the recovered file would be from multiple sources.

An error rate for each algorithm can be defined, but calculating the error rate is not really practical. For algorithm A, none of the files recovered from layouts 1 or 2 have the multiple source error and all files from layout 3 have the multiple source error. (Ignoring layout 4), an error rate for a particular drive can be calculated by counting the number of occurrences of each layout. An estimate of the error rate could be estimated if a large corpus of drives were examined where the layouts were accurately known. However, there is not a practical way to know what the actual layouts are. The same considerations apply to algorithm B. As for algorithm C, the multiple source error never occurs. However, algorithm C has the limitation that only the first block is recovered.

Tool testing can give a general indication for what the deleted file recovery algorithm does for specific conditions and file systems.

5.2. File Carving

File carving algorithms depend on the following characteristics of certain file types to determine the beginning and end of a file for carving:

- File types have a unique structure including a beginning marker (or signature) and an ending marker.
- File systems try to allocate file space contiguously.
- Files are allocated in cluster size units (multiples of 512).

A typical file carving algorithm includes the following steps:

1. Scan through unallocated space for paired file beginning marker and ending marker.
2. Check for reasonableness.
3. Collect the clusters between the two markers into a recovered file.

For some file types, e.g., pictures and videos, a visual examination can identify most incomplete or incorrectly recovered files. The picture does not display, the content is not recognizable or some similar result. For other file types, care must be used to examine the recovered file if data could be missing or come from multiple sources.

For example, suppose a file is recovered that tracks web sites visited and the number of times a site has been visited. The format of the file is as follows:

1. Web site URL
2. ‘;’
3. Unspecified other data
4. ‘;’
5. Visit count
6. ‘;’

The original file has the following content:

Cluster Number	Content
0	Beginning marker
1	www.alpha.com;aaaaaaaaaaaa;5; www.beta.net;bbbbb;7; . . . www. How-to-chloroform.com;hhh
2	Hhhhh;1; www.irs.gov;xxxx;20; . . . www.trees.edu;ttttttttttt;60; www.biology.edu;bbbbb;30; www.how-to-chlorophyll.com;ccccc
3	Cccc;74; www-movies.com;mmmm;8; . . . Ending marker

If this file is carved and cluster 2 is omitted, an incorrect inference about the interests of the user might be made.

5.3. Summary

It is difficult to have a meaningful error rate for deleted file recovery tools. Tool testing can reveal the quirks of tool behavior and guide the tool user in areas where additional detailed examination can mitigate misinterpretation.

SWGDE Establishing Confidence in Digital and Multimedia Evidence Forensic Results by Error Mitigation Analysis

History

Revision	Issue Date	Section	History
1.0 DRAFT	2012-09-12	All	Initial draft created and voted by SWGDE for release as a Draft for Public Comment.
1.0 DRAFT	2012-10-05	--	Formatted and released as a Draft for Public Comment.
1.1 DRAFT	2013-02-10	All	Revised on basis of public comments.
1.2 DRAFT	2013-03-21	All	Revised based on enhanced legal perspective.
1.2 DRAFT	2013-06-15	--	Formatted/edited document for re- release as a Draft for Public Comment.
1.3	2013-09-14	All	Updated document based on public comments and SWGDE voted to publish as an Approved document. Formatted and released as an Approved Document (version 1.3).
1.4	2014-06-11	Disclaimer, Section 7	Updated disclaimer. Updated reference style and formatting. No changes made to content.
1.5	2015-02-05	All	Minor grammatical/stylistic changes made throughout; voted to re-release as an Approved document. Formatted/edited document for release as an Approved document (version 1.5).
1.6	2017-01-12	3.1.2 and sidebar	Made corrections to the statistics in section 3.1.2 and the "Deleted File Recovery Example" sidebar. Voted to re-release as an Approved document
1.6	2017-02-21	--	Formatted/edited document for release as Approved document (version 1.6).
1.7 DRAFT	2017-07-19	Abstract; 1	Removed matching example in Abstract and made minor edits to Abstract and section 1. Formatted and posted as a Draft for Public Comment.
1.7	2017-08-24	All	Made minor grammatical corrections. SWGDE voted to publish an Approved document.
1.7	2017-09-25	--	Formatted and published as Approved (version 1.7).

Revision	Issue Date	Section	History
2.0 DRAFT	2018-06-14	All; Title	Minor editorial changes throughout. Changed "Digital Evidence" to "Digital ***and Multimedia*** Evidence" throughout document and in the title; this is a minor edit that expands the document's audience & scope to include multimedia evidence. SWGDE voted to release as a Draft for Public Comment.
2.0 DRAFT	2018-07-09	--	Formatted and released as a Draft for Public Comment.
2.0	2018-09-20	--	No changes were made following the Public Comment period. SWGDE voted to publish as an Approved document.
2.0	2018-11-20	--	Formatted and published as Approved version 2.0.