SMITHSONIAN INSTITUTION ARCHIVES

Archival Preservation of WEB Resources

Digital Quality Assurance Tools:

Technical Evaluation and Recommendations

Dollar Consulting

October 15, 2002

PREFACE

This report presents the results of a study undertaken by Dollar Consulting for the Smithsonian Institution Archives (SIA) as part of a larger effort to test and evaluate the feasibility of preserving Web sites and pages in an accessible, usable and trustworthy form for as far into the future as is necessary. Specifically, this study evaluates three digital quality assurance tools that can verify the integrity of Web sites and pages in the custody of the SIA over time. The intended audience is the Smithsonian Institution Archives and the report reflects the archives' understanding of its mission, requirements, and technology infrastructure. Nonetheless, it is hoped that other archivists, librarians, and preservationists concerned with preserving the integrity of electronic records in general will find this study useful as they develop their own digital preservation programs.

Table of Contents

EXECUTIVE SUMMARY

In July 2002 the Smithsonian Institution, Office of the Inspector General, issued a report entitled "Independent Evaluation of the Smithsonian Institution's Information Security Program" that identified significant information security deficiencies, especially with regard to the safety and security of its electronic assets and knowledge base. These security deficiencies are relevant for the Smithsonian Institution's use of Web based digital technologies to inform the public of various activities and programs, to offer "virtual exhibits," which only exist in electronic form, and to facilitate greater access to its electronic assets and knowledge base. Archival preservation of these electronic assets and knowledge base, which involves transfer of this material to the custody of the Smithsonian Institution Archives (SIA), focuses on the preservation of their integrity. Protecting the integrity of Smithsonian Institution electronic assets and knowledge base that exist in Web resources is a process that begins the moment Web resources come into the custody of the archives and continues for as long as they are retained in the archives.

Historically, archival preservation theory and practice have held that the physical chain of unbroken custody of records by a trusted third party is the most effective means for protecting the integrity of records. This is not an absolute defense against malicious efforts to alter records in the custody of archives through clandestine activities as evidenced by the discovery of forged documents in the custody of the National Archives and Records Administration. Although the integrity of electronic records potentially is far more vulnerable to accidental and malicious alterations that leave no visible trace than are their paper counterparts, powerful digital technology tools are available that can detect minute changes in individual electronic records and aggregations of electronic records. It is an interesting irony of history that these tools are more powerful, accurate, and speedy than any comparable techniques used with paper records.

In this study, these tools are referred to as digital quality assurance tools because protecting the integrity of electronic records over time is in fact a key component of a digital quality assurance program. Specifically, these digital quality assurance tools are algorithms that through a variety of mathematical processes compress an electronic document of any size or

format (i.e., images, text, and sound) to a specified number of bits called digital fingerprints, or integrity check values. Essentially, there are two types of digital fingerprints. One is called a Cyclic Redundancy Code (CRC) and the other is called a hash digest value.

CRC values (hereafter referred to as integrity check values) can be either 16 or 32 bits in length and are called CRC16 or CRC32, respectively. They are widely used to confirm the accuracy of digital material transferred from one system to another. In fact, most users never realize that a CRC integrity check value has been used. Despite its wide use, a CRC32 integrity check value provides weak protection for the integrity of electronic records. A CRC32 is "collision" prone in the sense that mathematically it is possible for two different documents to produce the same CRC integrity check value. Another weakness of a CRC32 is that it is computationally reversible. It would only take 16 computational cycles to generate a different electronic document that matched an existing CRC32 integrity check value, which would make it very difficult to determine which of the electronic records was authentic! Use of a CRC32 integrity check value, therefore, is problematic for protecting the integrity of electronic records in the custody of the SIA.

A hash digest is a unique digital fingerprint of any digital object that results from a series of mathematical manipulations in accordance with an open standard. Typically, this digital fingerprint is either 128 or 160 bits in length. The 128-bit hash digest is produced by an algorithm called MD5 while the 160-bit hash digest is produced by an algorithm called SHA-1 (Standard Hash Algorithm), the National Institute of Standards and Technology and the National Security Agency jointly developed. SHA-1 hash digests are considered unique and non-reversible. It is computationally infeasible for two different documents to have the same SHA-1 hash value or to create a document that matches an existing hash digest.

Another digital integrity protection tool included in the analysis is called time-stamping whereby an independent trusted third party time stamps ('notarizes') when a specific event occurs. In the instance of data integrity protection, time-stamping a hash digest would constitute irrefutable proof when the hash digest occurred. Time-stamping is somewhat costly and its implementation in protecting the integrity of SIA Web source material is problematic.

This study evaluated three digital quality assurance software tools that are commercially available. They are File Check, Veracity Personal, and the Digital Notary. Five evaluation criteria were used: (1) Ease of Use; (2) Scalable; (3) Execution Time; (4) Multiple Platform Compatibility; and (5) Integrity Check Robustness. Each digital quality assurance tool was run against a test bed made up 135 MB of Web page material from the Archives Center of the National Museum of American History. Table 1.1 below summarizes the technical evaluation of these three digital quality assurance tools.

Table 1.1: Strengths and Weaknesses of File Check, Veracity, and Digital Notary

Software	Weakness	Strength
File Check		
Ease of use		X
Scalable		X
Execution time		X
Multiple platform compatibility		X
Integrity value robustness	X	
Veracity		
Ease of use		X
Scalable		X
Execution time		X
Multiple platform compatibility		X
Integrity value robustness		X
Digital Notary		
Ease of use		X
Scalable		X
Execution time	X	
Multiple platform compatibility	X	
Integrity value robustness		X

Based upon this analysis, a resource allocation metric was developed to assist the SIA in estimating the human and technical resources required to generate integrity check values for

10,000 page of SI Web resource material through two media migrations over two decades. This metric is displayed in Table 1.2 below.

Table 1.2: Digital Quality Assurance Resource Allocation Metric

Digital Quality Assurance Resource Allocation Metric			
Feature	**File Check**	**Veracity**	**Digital Notary**
Scan time*	1 min 8 sec per 1,000 items	1 min 31 sec per 1,000 items	9 min. 15 sec per 1,000 items
Verify time*	1 min 4 sec per 1,000 items	1 min 13 sec per 1,000 items	18 – 56 min per 1,000 items
Software	Free	$65 per workstation	Free
Time-stamping	NA	NA	$50 per 1,000 transactions

*Scan and verify are generic terms that respectively describe the process of generating a CRD or hash digest (and time-stamping as appropriate) and generating a second hash digest that is compared with the first hash digest to confirm that no change has occurred.

Table 1.2 highlights two significant considerations when the quality assurance resource allocation metric is applied to 10,000 Web pages. First, there is a substantial difference in the amount of time required to generate and compare CRC values and hash digests. File Check would require approximately 1 hour and 30 minutes to scan and verify 40,000 CRC values and Veracity would require approximately 2 hours to generate and validate 40,000 hash digests. The Digital Notary would take between 18 to 37 hours to generate and compare 40,000 hash digests. This disparity in the execution time of each software package is a function of the robustness of the integrity check value used. File Check uses CRC32, which is "collision prone" and easily replicated. Veracity uses SHA-1, which is "computationally infeasible" to reverse or replicate. The Digital Notary combines two hash digest algorithms – MD5 and SHA-1 – with digital time-stamping to produce integrity check values that are even more "computationally infeasible" by several orders of magnitude to reverse or replicate. The second consideration is cost of the software. File Check is free, Veracity costs $65.00, and although the Digital Notary software is free, it would cost approximately $2,000 to notarize (i.e., time-stamp) 40,000 Web pages. Of course, this is an on-going cost that occurs during each instance of media migration.

Dollar Consulting recommends that the Smithsonian Institution Archives:

Implement a digital quality assurance program to confirm the integrity of electronic records when they are migrated to new media or converted to a technology neutral forma,

Adopt Veracity Personal (10/3/2002) as the digital quality assurance tool to confirm the integrity of electronic records when they are migrated to new media over the next five years, and

As digital quality assurance software tools that can confirm the integrity of electronic records migrated to new technology neutral formats become available, consider adopting one of them.

1 INTRODUCTION

1.1 Purpose

In 2001 the Smithsonian Institution Archives (SIA) commissioned a white paper on "Archival Preservation of SI Web Sites and HTML Pages"[1] that recommended among other things that the SIA adopt a two part preservation strategy to ensure the long-term usability and trustworthiness of SI Web sites and Web Pages. The first part of the strategy involved the migration of HTML pages to XHTML, a vendor and technology neutral format, to support data interchange and interoperability across heterogeneous technology platforms and a to write the migrated XHMTL pages to removable digital storage media.[2] Digital storage media, of course, are vulnerable to a poor storage environment, media obsolescence, and the potential loss of authenticity through accidental or malicious changes to the bit stream that comprises records. Consequently, the second part of the strategy called for:

1. Establishment of a program of periodic media renewal to overcome media obsolescence,
2. Maintenance of a secure facility where the media could be stored, and
3. Use of digital quality assurance tools to verify that no changes in digital records occurred while they were in storage or when the records were transferred to new storage media.

This study focuses on the practical and technical issues and costs associated with the use of digital quality assurance tools (item 3 above) such as Cyclic Redundancy Codes (CRC) and hash digest techniques to confirm the trustworthiness of Web sites and pages that no unauthorized change[3] has occurred in them while they are in the custody of the SIA. Specifically, the study provides the SIA with a metric for assessing the feasibility – both practical and technical - and cost of using COTS (Commercial Off The Shelf) digital quality assurance software to verify the integrity of electronic records for as far into the future as may be required.

[1] This study is available at http://www.si.edu/archivesw/links.html#publications.
[2] This study is available at http://www.si.edu/archives/archives.dollarprt2.html.
[3] The emphasis is on "unauthorized" change. As will be noted later, a change in the underlying bit stream of Web resources will occur when, for example, 4.0 HTML pages are migrated to 1.0 XHTML pages or when JPEG images are migrated to JPEG-2 images.

1.2 Scope

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

1. Review relevant literature on digital quality assurance tools,
2. Identify and evaluate Commercial Off the Shelf (COTS) software that implements digital quality assurance tools,
3. Acquire or gain access to the appropriate digital quality assurance software,
4. Use a SI Web "test bed" to determine the level of technical expertise use that the software requires and to estimate the technical and human resources required to support digital quality assurance tools over time, and
5. Prepare a final report that presents findings and recommendations.

The test bed referred to above consists of 135 MB of HTML pages, GIF and JPEG images, and moving images from the Archives Center of the National Museum of American History.

One other key scope consideration is that the focus of this study is archival preservation, not operational management of Smithsonian Institute Web sites and HTML pages. Some SI Webmasters may choose to utilize one of the digital quality assurance techniques discussed in this report to protect Web pages from hacker defacement, but this is an active Web site management issue and therefore beyond the scope of this study.

1.3 Methodology

The methodology employed in producing this report includes three components. The first component is a literature review and analysis of relevant source material relating to digital quality assurance techniques and software tools currently available. A survey of COTS digital quality assurance tools identified more than ten but this number was reduced to three because either the software package was primitive or the functionality was embedded in a larger and very expensive system that could not be used in the SIA.[4]

[4] An instance of primitive software is an implementation of SAHA-1 by the Netherlands Forensic Science Laboratory called SHA4labs.

The second component is the design of evaluation criteria that could be mapped against the requirements for technical expertise required, the computer execution time involved in using software, and the reliability of the outputs. The evaluation criteria used in assessing the digital quality assurance tools were: (1) ease of installation and use, (2) scalability, (3) computer processing time, (4) multiple platform compatibility, and (5) robustness of the integrity check value.

The third component is the actual testing of three selected digital quality assurance software tools using the SI Web test bed made available by the Archives Center of the National Museum of American History. The test bed was transferred to new storage media five times and an integrity check value was computed before and after each transfer. After each sequence of media transfer the before and after integrity check values were compared to determine if any bits had been lost. An exact match of the two integrity check values meant that no bits had been lost. The test bed also was segmented into data types HTML text, JPEG and GIF images, and audiovisual material and each data type went through the same sequence.

Milovan Misic, Head of Document Management and Archives at the World Intellectual Property Organization in Geneva, Switzerland handled the computational processing aspects of testing the software tools using the Web Site of the Archives Center of the National Museum of American History.

1.4 Report Organisation

This report consists of four chapters. It begins with an introduction to the study and delineates briefly the purpose, scope, and methodology of the study. Chapter 2 elucidates archival considerations in protecting the integrity of records from four different perspectives: (1) paper records, (2 cyber terrorism (3) digital integrity protection techniques, and (4) a trusted digital repository. The discussion of digital techniques includes bit/byte comparisons, cyclic redundancy codes, and hash digests. Chapter 3 evaluates each of three digital quality assurance software tools – File Check, Digital Notary, and Veracity. The final chapter presents findings and recommendations.

2 ARCHIVAL PRESERVATION OF THE INTEGRITY OF WEB SOURCE MATERIAL

2.1 The Archives Historical Preservation Mandate[1]

Albertino Barisoni (1587 – 1667), author of *De Archivis Commentarius*, one of the first two books written about archival science, declared that the purpose of archives (quoting Justinian) is "to have custody of them [records] so that they may remain uncorrupted and may be quickly found by those requiring them."[2] Archivists have expanded this notion of "uncorrupted records" into records that have not been altered or changed over time and whose provenance is protected through reliance on an unbroken chain of archival custody. This in turn has given rise to a presumption of authenticity of records in an archives.

Many scholars and researchers who use primary source material in their research that is in the custody of an archives generally take for granted the authenticity of archives. This is "presumptive authenticity" because most of what is considered primary source material - records for archivists - is the by-product of routine transactions required to complete an action (legal, financial, or business) and these by-products typically form a corpus of related material preserved in an archives. The presence of a document in a corpus of similar forms of material that is known or believed to have been produced in accordance with standard procedures and maintained in an archives carries a presumption of authenticity. Therefore, both the context and the content of documents bear witness to authenticity, unless there are grounds for suspicion. In this regard, diplomatics, which incorporates tools for analysing documents, developed analytical procedures for ascertaining the authenticity of documents that helped pave the way for historians and other scholars to develop internal and external rules of corroborative evidence for this purpose. Typically, these "documentary forensic tools" focus upon physical attributes of records that include watermarks, ink, type font, and the like.[3]

---

[1]This discussion draws upon material that the author developed in 1992 – 1993, especially Charles Dollar, "Archivists and Records Managers in the Information Age," *Archivaia* 36 (Autumn 1993): pp. 43 – 44 and expanded in Charles Dollar, *Authentic Electronic Records; long-term Access Strategies* (Cohasset: Chicago, 1999).
[2]Quoted in Lester k. Born, "The de Archivis Commentarius of Albernito Barisoni," *Archivalische Zeitschrift*, No. 50 – 51 (1955): 21
[3] Charles Cullen, "Authentication of Digital Objects: Lessons from a Historian's Research," *Authenticity in a Digital Environment* (Council on Library and Information Resources: Washington, DCL 2000): 1 – 7.

One can argue that an unbroken chain of archives custody of records over time is a sufficient basis for their authenticity. Unfortunately, there are exceptions that call this into question. One exception involves U.S. Navy records in the custody of the National Archives and Records Administration that relates to the USS Constellation, a wooden-hulled naval vessel permanently moored" in Baltimore Inner Harbor that purportedly was a "sister ship" to the USS Constitution, "Old Ironsides," that was built in 1797 and anchored in Boston. Records found in a record series in the National Archives and the Franklin D. Roosevelt Presidential Library, supported the claim that the Constellation had in fact been built in 1797. Based upon the chain of custody principle, these records were presumed authentic because they were created and maintained by the U.S. Navy and then transferred to the National Archives.

Dana Wegner confirmed that much of the written documentation to support a construction date of 1797 and subsequent service in the U.S. Navy that paralleled that of the USS Constitution was false.[4] It appears that some forged documents were surreptitiously inserted into files at the National Archives and the Franklin D. Roosevelt Library, where researchers subsequently "found" them. Interestingly, Wegner was able to identify the "forged" documents because they could be subjected to forensic tests such as ink, typewriter font, spelling errors, and markings made by the National Archives. In this particular instance, it appears that the National Archives did not a adhere to a fundamental archives responsibility "to ensure the integrity of the documents even after they are legally transferred to a repository."[5]

2.2 Digital Record Vulnerability, Hackers, and Cyber Terrorists

Unlike paper records, most digital records and aggregations of digital records can be easily changed with no visible evidence of the change.[6] The digital records that comprise SI Web resources have no inherent physical attributes such as ink, type font, watermarks, and the like. In fact, these digital records exist as logical records that become physical artefacts only when

[4] Dana M. Wegner, *Fouled Anchors: The Constellation Question* Answered (Behtesda, Md.: David Taylor Research Center, 1991). Peter Hirtle's excellent essay, "Archival Authenticity in a Digital Age,"in Authenticity in a Digital Environment (Council on Library and Information Resources: Washington, DCL 2000): 8 –23 called this to my attention.
[5] Peter Hirtle, "Archival Authenticity in a Digital Age," p. 13.
[6] A PDF document can be password protected so that only the person who has the password can change the document. Similarly, digital documents can be encrypted so that they are invulnerable to change and alteration.

software interprets the 1s and 0s and renders them on a monitor and printer in a humanly understandable form. This separation of the logical structure of digital records from their physical structure means that content can be changed with no visible evidence of that change and the presentation or rendering can also be changed with no visible evidence of the change. This vulnerability of digital records means that when the SIA takes custody of Web resources, that is, digital records, it faces at least three major challenges in ensuring their authenticity.

First, it is reasonable to assume that some individuals will try to "reinvent reality" by attempting to gain access to SI digital archives and alter specific digital records or insert counterfeit digital records to support an ideological goal or simply to cause havoc. Furthermore, the skill of hackers to break into secure systems and the emergence of cyber terrorism should sound a warning to the SI about protecting the integrity of digital records.[7] In this regard, the "Independent Evaluation of the Smithsonian Institution's Information Security Program" study confirmed that SI electronic assets and knowledge bases are at substantial risk and called for immediate action to address serious security deficiencies. This is particularly troublesome in terms of assuring future users of digital source in the custody of the SIA that they have not been altered or deleted by hackers or cyber terrorists. There are digital quality assurance tools that can detect changes in digital objects, which could be useful for ensuring the authenticity of a single electronic record, a record series or some other comparable aggregation.

The second challenge that the SIA faces is how to deal with digital records that will inevitably undergo some change as researchers and other users obtain copies of digital records and then cut and paste them into a variety of contexts. Over time these changes are likely to be perpetuated so that there may be different versions of the same digital record. What are the grounds for the SIA to assure future users that in fact the digital record(s) in its custody are authentic as opposed to others outside the custody of the archives? Again, digital quality assurance tools are available to generate digital fingerprints of records or aggregations of records that can attest to the authenticity of the digital records in the custody of the archives.

[7] There is considerable literature on hackers. One fairly typical article, 'Hackers to Corporate America: You're Lazy," notes that defacement of Web sites is a special problem. One of the proposed ways to protect Web content is the use of digital hashing of HTML documents and pages. For cyber terrorism, see publications by Dorothy Denning that are available at http://www.cs.georgetown.edu/~denning.

The third challenge the SIA faces is how to ensure the integrity of digital records as preservation activities are undertaken to extend the usability of the records in the face of technology obsolescence. These preservation activities will involve media renewal and migration to new formats and technology platforms. It is likely that digital records will have to be transferred to new storage media every ten to fifteen years, and during each media renewal process there is the potential for some bits to be lost or otherwise changed. Although the actual loss of such bits during media renewal or transfer appears to be rare,[8] there is a potential risk for undetected errors to occur over time that could have a significant impact on the authenticity of the records. Digital quality assurance tools can generate digital fingerprints of digital records before and after media renewal activity that can confirm that not a single bit was lost or changed during the process and thereby provide unequivocal evidence of authenticity.

There are no digital quality assurance tools that can automatically confirm that no changes occur in Smithsonian Institution digital records that are migrated to new file formats or to new technology platforms because the bit stream underlying the records undergoes some degree of change so pre-migration digital fingerprints and post-migration digital fingerprints are different. One viable alternative is to create a preservation history trail log that captures information when each instance of file format migration and technology platform migration occurred, who conducted it, how it was done, and what the results were. It is likely that the tools used for migration to new file formats and to new technology platforms will be widely available and their characteristics well-documented. Take, for example, migration of digital records from Office 97 to Office 2000. Each software package is well-documented so specific changes are predictable and it should be relatively easy to establish specific representation changes A second alternative would be to implement a digital time-stamping functionality along the lines discussed later in section 2.3 below.

[8] One of the few published statistics on this topic involves The University Licensing Program (TULIPP, a project to test a system for the networked delivery to, and use of journals, at the user's desktop. Elsevier Science and nine leading universities in the United States participated in the project. One part of the project involved scanning hard copy pages of journals and stored them in a TIFF format. The project report stated that a single incorrected bit in a TIFF image makes it useless and reported that "On average we have encountered one incorrect bit, resulting in a fully incorrect image, per maybe 20,000 correct images. See Elsevier Science, TULIP Final Report (New York, 1996), p.10, Chapter II. This report is available at http://222.elsevier.nl/homepage/about/resproj/trchp2.htm. This page was retrieved on October 26, 2001.

2.3 Protecting the Authenticity of Digital Records: Bit/Byte Counts, Cyclic Redundancy Codes, Hash Digests, Time-stamping, and Digital Signatures

Integrity check techniques to confirm the accuracy of data transferred to new storage media or transmitted over the Internet or a private network have been in use for several decades. They have evolved from a simple bit/byte count to Cyclic Redundancy Codes (CRC) to robust hash digests, time-stamping, and digital signatures. The evolution of digital integrity check techniques from simple bit/byte counts to more powerful ones tracks closely with the increasing computational power available to users. The underlying premise of this section's focus on protecting the authenticity of digital objects is that it is too important to be left to simple and ineffectual integrity checks.

Bit/Byte Counts. In the late 1970s the Machine-Readable Archives Division of the National Archives and Records Service was the first archives in the world to use bit/byte counts to confirm that no data was lost or corrupted during file tape copying or reformatting.[9] Essentially, a bit/byte count tabulates the number or incoming bits and compares the number of bits transferred to a new tape. If there is an exact number of bits recorded on the source tape, with a similar count of bits recorded on the output tape, it is presumed that no error had occurred. In the early 1990s, the Center for Electronic Records at the National Archives of the United States contracted for the development of an automated preservation system that incorporated a bit/byte count into the functionalities of the Archives Preservation System. This is a crude measure of data integrity because the bits in a binary stream that represent a document (thereby altering its content) can be modified without changing the overall bit count.

Peter Graham's 1993 article, "Intellectual Preservation in the Electronic Environment,"[10] was the first public acknowledgement within the library community of the potential problem of ensuring the authenticity of electronic library material. Graham argued that the fundamental problem

[9] For more information on this practice see Charles Dollar, "An Insider/Outsider Perspective on the Electronic Records Program of the National Archives of the United States," forthcoming in Bruce Ambacher, ed., *Thirty Years of Electronic Records* (Scarecrow Press).

[10] Peter Graham, "Intellectual Preservation in the Electronic Environment," in Arnold Hirshon, Ed. *After the Revolution, Will you be the First to Go?* (Chicago, 1993): pp. 24 – 33.

libraries would face with electronic records was not their physical preservation but rather with preserving their intellectual content from accidental or malicious changes. Graham suggested that cyclic redundancy codes and hash digests could serve as master digital fingerprints or digital surrogates of any material in electronic form and establish the basis for protecting the intellectual content or integrity of electronic library material. In 1993 Charles Dollar extended Graham's recommendations to the archives community in an article entitled "Archivists and Records Managers in the Information Age." [11] He later expanded the concept of using digital cyclic redundancy codes and hash digests to verify that no change had occurred during media renewal or storage.[12] More recently, Bill Underwood proposed the use of hash digests as part of an electronic records preservation program to confirm the integrity of electronic records.[13]

Cyclic Redundancy Code (CRC).[14] Although CRCs are used largely to ensure error-free telecommunication transmission, they also can be used to confirm the integrity of digital objects.**[15]** Calculation of a CRC generally occurs in the following manner. An originating device or unit computes a CRC for a message using a standard algorithm and appends it to the message or digital object. A receiving device or unit computes a CRC of the underlying bit stream of the transmitted digital object using the same standard algorithm and compares it with the appended CRC. If the two CRCs match, then both messages are identical. In effect a CRC is a digital fingerprint, or surrogate, of a digital object of arbitrary length.

The actual computation of a CRC involves use of an algorithm that divides a fixed number of bits (usually 16 or 32 bits) from a digital object by a known divisor of the same number of bits (called a polynomial). The result of this division is a CRC that becomes the divisor for the next segment of bits. This process is repeated and the resulting quotient of the last bit segment becomes the CRC for all of the bit segments that comprise the message. Computation of a CRC

---

[11] *Archivaria* 36 (Autum 1993): 37 – 52.
[12] Charles Dollar, *Authentic Electronic Records: Long-Tern Access Strategies* (Chicago, 1999).
[13] Bill Underwood, "Using JARS to Preserve Electronic Records," *Proceedings from an International Symposium February 2001* [InterPARES], pp. 112 – 119. It is available electronically at http://www.interpares.org/documents/interpares_symposium_2001.pdf.
[14] For a useful technical explanation of CRC32 see "Selection of Hashing Algorithms," (June 30, 2000), which was written by Tim Boland and Gary Fisher of the National Institute of Standards and Technology. It is available electronically at http://www.nsrl.nist.gov/documents/hash-selection.doc.
[15] Bill Burr of the National Institute of Standards and Technology takes exception to this claim. He argues that CRC32 is useful for "detecting errors caused by random noise. But CRC32 wouldn't do at all, for example, to protect code against intentional modification." Bill Burr to Charles Dollar, June 9, 2002 (EM to the author).

is fast because lookup tables containing every possible combination of bits in a bit segment are pre-generated. The length of the message is of no consequence other than requiring more computation time.

CRC algorithms with established polynomial divisors are available as either 16 or 32 bits. A CRC32 is considered more robust because the divisor and bit segment contains 32 bits. One weakness of CRC32 is that it is "collision prone," which means that mathematically it is possible for two different documents to produce the same CRC. Another weakness of CRC32 is that it is computationally reversible in that it would take only 16 computational cycles to generate a digital object that matched an existing CRC32. This digital object could be altered and a new CRC generated and appended. Computation of a second CRC32 would yield an identical CRC and there would be no evidence in the digital object or the CRC that a change had occurred. A number of organisations use CRC32 to confirm the integrity of digital objects[16] but its collision prone tendency along with the fact that it is computationally feasible to reverse a CRC32 to its originating digital object makes its use problematic for protecting the integrity of digital objects against intentional modification.[17]

HASH Digest Algorithms.[18] A hash digest is a unique digital fingerprint of any digital object of arbitrary length and format that is obtained by dividing a digital object into multiple segments of 512 bits or 16 32-bit words.[19] Each segment goes through a series of computations that employ four or five 32-bit words (128 bits or 160 bits) that are defined by a standard. During the first round, the 32 bit words are run against the first 512 bit segment and through a series of computations are transformed so that the resulting 32 bit words are a compressed version of the 512 bit segment. These 32 bit words are run against the next 512-bit segment. This process is repeated until there are no segments left. The last set of computed 32 bit words is the hash digest for the entire digital object.

[16] David Holdsworth of the University of Leeds notes that CRC32 is relied upon to ensure that "electronic equipment has not gone wrong . . . .In the case of long-term storage, there also is the issue of whether you feed the need to confirm the integrity in between migration process. We do not, but we have triplicate copies – 2 on-site and one remote." David Holdsworth to Charles Dollar, November 19, 2001 (EM to the author).
[17] Bill Burr is the source for this statement. Bill Burr to Charles Dollar, June 9, 2002 (EM to the author).
[18] For an informed discussion of hash digests see "Selection of Hashing Algorithms".

Hash digest algorithms focus on the underlying bit stream of the 1s and 0s of a digital object so it makes no difference what its intellectual content is. In other words, it is irrelevant whether the digital object is a book, music, an image, or a combination of data types. Hash digests are considered non-reversible, or one-way, because it is computationally infeasible to derive the text or content of a full document from its hash value.

There are two standard hash digest algorithms in the public domain that are in general use today. One is the Message Digest Level 5 (MD5), which was developed by Ron Rivest, and it is a 128-bit hash. The other is the Standard Hash Algorithm (SHA-1), which the National Institute of Standards and Technology and the National Security Agency jointly developed, and it is a 160-bit hash. Both hash digests are considered "one-way" because it is computationally infeasible to find two digital objects that hash to the same value, or given only a hash digest, to generate a digital object that hashes to that value. "Computationally infeasible" means that the amount of time and computer resources required to "break" a hash digest with today's technology makes it unlikely that it could be done within a reasonable period of time. For example, MD5 is a 128 bit algorithm, which means that it would take on the order of 2**64 operations to produce two messages with the same hash digest and 2**128 operations to produce a digital object from any given hash digest. If we assume that each operation would take 1 second, then producing two digital objects with the same hash would take more than 10^{10} years, which is the estimated age of the universe.[20]

MD5 and SHA-1 respectively produce hash digests of 128 bits and 160 bits. They are so sensitive to changes in a digital object that changing only 1 bit in a digital object will result in a substantially different hash value. For example, running the SHA-1 hash algorithm against the scanned image of a $100 bill in which several pixels had been changed, results in a substantial change in the hash digest value. It is this sensitivity to minute changes in the underlying bit stream of digital objects that makes hash digests a powerful digital quality assurance tool that can help give future users of archived Web source material confidence that it has been preserved against corruption and alteration.

---

[19] If the length of a digital object is not an exact multiple of 512 bits then "padding bits" are appended to the end of the digital object

[20] This time estimate is extrapolated from Jalal Feghhi, Jalil Feghhi, and Peter Williams, *Digital Certificates: Applied Internet Security* (Addison-Wesley: New York, 1999): 50.

Hash functions are well suited for ensuring the integrity of digital objects because there is a strong element of predictability incorporated into MD5 and SHA-1. To put it differently, running a hash algorithm against a specific digital object that remains unchanged will produce the same hash digest or value, regardless how many times the algorithm is run. As noted earlier, if there is only a change of 1 bit in the digital object, the hash value will be substantially different. It should be noted that hash algorithms do not provide confidentiality nor do they require use of a private encryption key.

Time-stamping. MD5 and SHA-1 are powerful indicators of the trustworthiness of digital objects but they do can not resolve the more fundamental issue of when the hashing (or associated activities) occurred, which clearly has an impact on non-repudiation of digitally signed transactions. Time-stamping, therefore, consists of techniques that establish when a specific digital object was hashed. The general procedures underlying time-stamping include the following:

1. Create a hash digest of one or more digital objects,
2. Transmit the hash digest (or multiple hash values) to a trusted third-party digital Time-stamping Authority (TSA),[21]
3. The TSA generates and appends a time and date to the hash value, signs it with its digital signature (discussed below), and returns it to the customer using standard Internet protocols such as a Web page or Email, and
4. The customer can compare the time-stamped hash digest with the hash digest originally computed and establish that they match, thereby providing certification by a third party of when hashing occurs.

Digital time-stamping of a hash digest provides irrefutable evidence of when it occurred and thereby indirectly guarantees the trustworthiness of a document when a digitally time stamped hash digest is subsequently compared with a second hash digest of the document. If the two hash digests match then the document has not been altered.

[21] Trusted TSAs currently available include Surety Technology, Digital Authentication Systems, Digital Stamp Company, and Entropia Internet Notary Service.

On-going Verification of the Integrity of Migrated Digital Objects. As noted earlier, when digital objects are migrated to new file formats and to new technology platforms, the bit stream underlying the records undergoes some degree of change. Pre-migration digital fingerprints and post-migration digital fingerprints, or integrity check values, will be different and therefore unable to provide unequivocal evidence of no change. However, Stuart Haber (co-found of Surety Technologies) has suggested a scenario whereby digital time-stamping could be used to verify the integrity of digital documents over successive migrations to new formats or technology platforms. The scenario would approximate the following steps.[22]

Suppose a hash digest of a HTML digital document (HTML1/HD1) is generated in 2000 and then time-stamped. Three years later the HTML1/HD1 document is migrated to XHTML and is named XHTML-1 and a hash value is generated that is named XHTML-1/HD1. These two hash values – HTML1/HD1 and XHTML-1/ HD1 – are merged and then time-stamped. The time-stamped certificate of HTML1/ HD1 and XHTM-1L/HD1 memorializes this action in 2003. Repeating this process over time creates a "digital memorialization chain" that inextricably links each pre-migration and post-migration hash value. Of course, this does not provide an absolute guarantee that no loss of meaning occurred during migration. One way to mitigate this problem is to conduct a visual inspection of the pre and post migration digital documents to verify that no loss of meaning occurred during migration and to include an attestation to this effect with each migrated digital document that is hashed and then time-stamped. A second way to mitigate this problem, which presumes that the migration tool(s)/procedure(s) is (are) deterministic (i.e., it always produces the same output with the same input and procedures) and executable at the time of verification, is to replicate a specific migration and then generate a new hash digest of each migrated digital document. Each new hash digest could be mapped against its hash value in the "digital memorialization chain" to verify the integrity of the migrated digital documents.

These integrity check verifications are likely to be labor intensive and cumbersome if they are manually performed, and in practice, are unworkable if thousands of integrity check verifications are undertaken. Automation of these procedures could substantially reduce the requirement for

[22] This description draws heavily upon a draft document entitled 'Time-stamping for long-term preservation of digital documents' that Haber generously shared with the author. Any errors or misinterpretations of Haber's view are

human involvement and make them more workable. However, all of the digital quality assurance tools currently available ultimately rely on trust that people are truthful when they say what they do and do what they say with regard to protecting the integrity of electronic records.

Digital Signatures. Hash digests are powerful indicators of the integrity of digital objects but they do not guarantee authentication, which is where public key cryptographic digital signatures come into play. In public key cryptography (known generally as Public Key Infrastructure [PKI]), a trusted third party issues a set of unique, mathematically related keys called a key pair. One key is a private or secret key that only the signer or owner knows and has access to, and the second key is a public key that corresponds to a private key. The two keys are mutually dependent in that only one public key corresponds to only one private key. The private key is used to sign electronic documents and the public key, which is available to anyone, is used to verify the authenticity of signature. This presumes that the owner of the private key protects it from disclosure and security compromise.

The public and private key combination works in the following manner. Suppose one person wants to send an electronic document over the Internet to another person. The sender first creates a hash digest of the electronic document and then encrypts it with the private key. The clear or full text of the message is sent along with the encrypted hash digest to the intended recipient who uses the public key that matches only one private key to decrypt the message. The successful decryption guarantees that only one person – the holder of the private key – sent the message. A hash digest algorithm is run against the clear text document and the resulting hash digest compared to the decrypted hash digest. A match of the two hashes digests means that no one has tampered with the content of the document.[23]

Suppose this same individual wants to make sure that only the intended recipient can read the document. The clear text message is run against the hash algorithm and the resulting hash digest is encrypted with the recipient's public key. The fact that only one key – the recipient's

solely the responsibility of the author.

[23] Digital signature algorithms, especially for long keys (1024 bits or higher) are slow and inefficient and typically they are not used to encrypt full text messages except in special circumstances or where this does not degrade system performance.

private key – can decrypt the hash digest and guarantees that only the intended recipient can decrypt the encrypted hash digest.

PKI digital signature technology offers a powerful tool to ensure that the private key signature is authentic, is not reusable, and cannot be repudiated. Furthermore, once an electronic document is signed, it cannot be altered without detection through the comparison of hash values. Nevertheless, PKI digital signature technology poses issues for long-term access to trustworthy and usable electronic records. PKI makes users responsible for managing and protecting private keys and many knowledgeable observers believe that most end users fail to understand the importance of protecting the security of a private key. Under current policy and procedures most PKI digital signature certificates expire every two years. Re-validation of a digital signature requires access to either the public or private key and over time this becomes problematic because of software and hardware obsolescence. It is largely for this reason that the National Archives of the United States requires the decryption of any encrypted records of permanent value before they are transferred to its custody.[24]

2.4 A Trusted Digital Repository

The hash digest and time-stamping technologies reviewed above are crucial to providing future users a high level of confidence in the integrity of electronic records. However, protecting the integrity of electronic records should not occur solely within a technology vacuum but rather within a technology environment that is informed by a clear and coherent digital archives policy and accepted procedure, and good practice. It is for this reason that the Research Library Group's (RLG) recent publication (May 2002) of a report entitled, *Trusted Digital Repositories: Attributes and Responsibilities* is noteworthy. The report defines a trusted digital repository as "one whose mission is to provide reliable, long-term access to managed digital resources to its designated community, now and in the future."

[24] 36 CFR 1234 .1288 does not specifically prohibit encrypted electronic records as such but rather refers to compressed electronic records as not being eligible for transfer to NARA. For an informed review of key issues associated with the long-term retention of encrypted electronic records, see Wayne Schoeder and Tom Perrine, "Security Models for NARA Electronic Records Management," pp. 14 – 16, SCSC TR-2001-4 (January 2001). Reagan Moore et al note that records embedded in formats using a compression method can be accessible in the future only if the decompression method is executable. See Bertam Ludascher, Richard Marciano, and Reagan Moore, "Towards Self-Validating Knowledge-Based Archives," *San Diego Supercomputer Center Technical Report 2001-1,* (January 2001) p.3. Both technical reports are available at the National Archives Electronic Archive Records URL.

Among other things, the report lays out a broad framework to support the development of a process for the certification of digital repositories and to determine the technical strategies that best provide for continuing access. The general framework and technical strategies that the RLG publication articulates are consistent with the focus of this SIA report on digital quality assurance tools. By implementing the recommendations of this study the SIA can lay the foundation for ensuring that it can meet the requirements for a trusted digital repository as RLG fleshes out the broad framework and technical strategies for trusted digital repositories.

3 TECHNICAL COMPARISON AND ASSESSMENT OF DIGITAL QUALITY ASSURANCE TOOLS

3.1 Methodology Review

This section discusses in greater detail the overall methodology used in this study. It includes:

Establishing the parameters of the study through a literature review on software availability,

Defining the evaluation criteria to be used in evaluating specific digital quality assurance software tools,

Reviewing the test environment of digital quality assurance software tools, and

Conducting the actual test of selected digital quality assurance software tools.

3.1.1 Literature Review on Software Availability

A wide ranging Google search of Internet and other documentary sources identified a number of articles and other sources that were reviewed. In addition, several specialists were consulted for their insights.[1] One especially useful informative source consulted was the Secure Hash Standard (SHS) Validation List maintained by the Computer Security Research Center of the National Institute of Standards and Technology.[2] This is a list of some 115 organizations whose product is certified as compliant with SHA-1. One such product is called SHA4LABS that the Netherlands Forensic Laboratory uses to hash digital criminal evidence to prove that data has not been manipulated. SHA4LABS processes single discrete files rather than multiple files in a production mode and runs in DOS or command line mode, which makes it quite cumbersome, and it produces no reports. This limited functionality precluded its inclusion in this review of digital quality assurance tools.

[1] They included Professor Margaret Hedstrom (University of Michigan), Don Sawyer (Goddard Space Flight Center), and Frank Lambert (Iwitness).
[2] http://csrc.nist.gov/cryptval/shs/shaval.htm.

Other digital quality assurance tools that use SHA-1 and are compliant with the SHS include Entrust, Baltimore Technologies, and other PKI related software. However, their use of SHA-1 is tightly linked to other system functionalities that are not germane to the SIA so they are not included in the study.

A Google search on "checksum", "hash digest," and "time-stamping" identified several other sources. One of them was TripWire, which is a very powerful and modular set of software that monitors network and system performance against intrusions. TripWire is generally considered to be a "top of the line" product and there is an academic version that is available at no cost. However, the execution of the academic application is in DOS and is similar to that of SHA4LABS so it is not included in the project. The Google search also identified three other digital quality assurance tools – File Check, Digital Notary, and Veracity – that are included in the analysis.

3.1.2 Digital Quality Assurance Tools Evaluation Criteria

Five evaluation criteria were developed for this study that incorporated some features of the evaluation criteria used in an earlier study on the migration of HTML pages to XHTML. The five evaluation criteria are:

1. Ease of use
2. Scalable
3. Execution time
4. Multiple platform compatibility
5. Integrity check robustness

Ease of use (1) refers to the level of technical expertise and manual procedure features required to install and run a digital quality assurance tool. For example, a DOS based application is much more cumbersome to use than is one that is Windows based. **Scalable (2)** means that the digital quality assurance tool can process equally well a single document, a folder, a directory, or hundreds of documents or folders in a production environment. The only significant difference so far as scalability is concerned is the increase in the amount of time required to generate and compare integrity check values for multiple documents, folders, and

directories. **Execution time (3)** includes the various steps and activities that must be invoked during creation and comparison of integrity check values. **Multiple platform compatibility (4)** refers to the software being capable of running on a variety of operating systems and environments. **Integrity check robustness (5)** relates to the specific integrity check algorithm's resistance to be "cracked." For example, a CRC32 integrity check algorithm generates a weak integrity check while SHA-1 generates a strong one.

3.1.3 Test Environment.

The test environment consisted of three broad components. The first was a test bed comprised of HTML pages, JPEG/GIF images, moving images and sound files from the Archives Center of the National Museum of American History. They were organized into three categories: (1) The entire Web site; (2) HTML pages; and (3) JPEG and GIF images. The second component of the test environment consisted of three different technology platforms on which the integrity check values were run: a Pentium MMX 266 MHz, a Pentium II 333 MHz, and a Pentium 4 1.7 GHz. The third component included two different storage media: a 40 GB Hard Disk Drive and a 16X CD-Reader and Burner.

3.1.4 The Test Process

The testing involved using each digital quality assurance software tool to compute an integrity check before and after transfer of the test bed to a new storage medium. The sequence was as follows:

1. From HD to CR-WR
2. From CD-WR to HD
3. From HD to another HD
4. From HD to CD-WR
5. From CD-WR to DC

After each sequence of media transfer the before and after integrity check values were compared to determine if any bits had been lost. An exact match of the two integrity check values meant that no bits had been lost. As noted above, the test bed was also segmented into

data types – HTML text, JPEG and GIF images, and sound and moving images and each data type went through the same sequence.

3.2 File Check - Technical Review and Findings

3.2.1 Background

File Check is a user-friendly tool developed by Earl F. Glynn[3] for verifying that a copy of an original document, folder, directory or an aggregation of documents, folders, or directories written to any media has exactly the same content as the original. File Check, which operates in a Windows and NT environment with the familiar point, click, and drag tools, has two key functions. The first function is to scan an "original" document, folder, or directory and compute a CRC32 integrity check value that is coded and rendered as a hexadecimal representation. The second function is to verify at some later date the accuracy of a copy of the copied document, folder, or directory by generating a CRC32 second integrity check value and comparing it with the first CRC32 integrity check value of the original document, folder, or directory. An exact match of the two CRC32 integrity check values indicates that no loss of bits occurred during the transfer.

3.2.2 Ease of Use

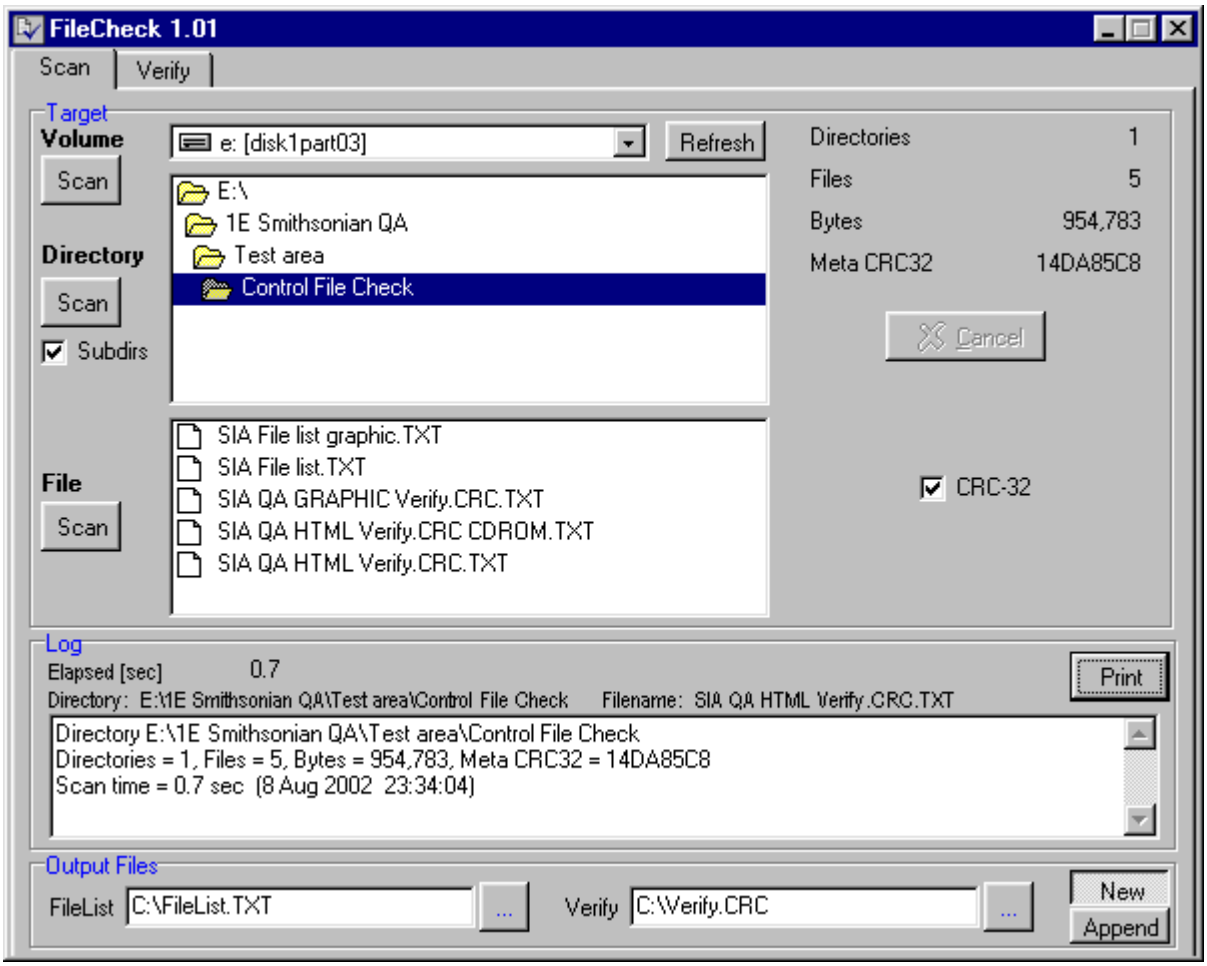
Installation of File Check is very easy. After downloading from the EFG Website, a ZIP executable file is unpacked and automatically installed into the same directory with the ZIP file where it can be moved or copied into any other subdirectory. It is not necessary to modify any setting or to modify configuration of the software. File Check runs in a user-friendly Windows environment (i.e., drag and drop features) and is easy to use. It does not support a help function.

"Scan" and "Verify" are the two main functions of File Check. The scan operation is initiated by selecting the document(s), folder(s), or directory (ies) to be scanned. The familiar Windows browse function can be used to locate them. After selecting the digital objects to be scanned,

[3] http://www.efg2.com

clicking on the scan button initiates the computation of a CRC32 for each document that is written to a file list. As each document is scanned, a 32-bit integrity check value is generated. Upon completion of the scanning of all the documents, the individual 32-bit integrity check values are collapsed into a single value called Meta CRC32, which is displayed in the upper right quadrant of the screen shot in Figure 3.1 as well as in the message area. Other information included in the message area is the name of the input document (file), the date, the number of bytes in the document, and the scan time.

Figure 3.1 FILE CHECK SCAN AND VERIFY



3.2.3 Scalable

File Check is scalable from a single document or file at a time, regardless of file format (HTML, JPG, AVI), through a subdirectory or a directory up to an entire single drive or even group of networked drives. File Check supports all current storage media, including CD-R, DVD-R, and magnetic tape. It is also possible to verify results between two different drives simultaneously. This is useful when data are transferred from one medium to another.

3.2.4 Execution Time for File Check

File Check is very fast. The scan time when 1,844 test bed pages were read from the hard disk of a Pentium MMX 266 MHz computer was 125 seconds. The verify time for the same dataset was 119 seconds. Of course, both the scan time and verify time are reduced substantially when a Pentium II 333 MHz and Pentium 4 1.7 GHz are used. The scan and verify times for other file types and hardware configurations are displayed in Table 2.

Table 3.1 File Check Scan and Verify Times

File Check	Pentium MMX 266 MHz	Pentium II 333 MHz	Pentium 4 1.7 GHz
	Scanning / Verify time	Scanning / Verify time	Scanning / Verify time
Hard Disc All files	Scan 2 min 5 sec Verify 1 min 59 sec	Scan 1 min 12 sec Verify 1 min 11 sec	Scan 42 sec Verify 51 sec
Hard Disc HTML files	Scan 49 sec Verify 44 sec	Scan 34 sec Verify26 sec	Scan 24 sec Verify21 sec
Hard Disc Graphic files	Scan 1 min 19 sec Verify 1 min 18 sec	Scan 48 sec Verify 46 sec	Scan 28 sec Verify26 sec
CD-RW* All files	Scan 4 min 26 sec Verify 3 min 35 sec	Scan 2 min 53 sec Verify 2 min 20 sec	Scan 1 min 42 sec Verify 1 min 26 sec
CD-RW HTML files	Scan 1 min 39 sec Verify 1 min 37 sec	Scan 59 sec Verify 58 sec	Scan 39 sec Verify 34 sec
CD-RW Graphic files	Scan 2 min 33 sec Verify 2 min 12 sec	Scan 1 min 31 sec Verify 1 min 19 sec	Scan 55 sec Verify 48 sec
*The increase in execution time reflects the relatively slow data transfer rate for CD-RW media.			

3.2.5 Multiple platform compatibility

File Check has a high cross platform compatibility. It can be loaded and executed in a Windows 95/98/XP, NT, and Unix environments.

3.2.6 Integrity Check Robustness

File Check implements the CRC32 integrity check value in scanning documents and later verifying that no change in them has occurred. As noted in Chapter 2, a CRC32 integrity check value is "collision prone" and it is computationally feasible to reverse a CRC32 check value to its originating digital object. Although a CRC32 integrity check value is useful for identifying accidental changes in digital objects that may occur during transmission, its use is problematic for protecting the integrity of digital objects against intentional modification.

3.2.7 File Check - Technical Summary and Evaluation

Table 3.2 summarizes the technical assessment and findings regarding File Check as a potential digital quality assurance tool to protect the integrity of SIA Web resource materials.

TABLE 3.2 FILE CHECK TECHNICAL EVALUATION SUMMARY

Feature	Pentium MMX 266 MHz	Pentium II 333 MHz	Pentium 4 1.7 GHz	Other
Ease of Use				
Installation	Easy	Easy	Easy	
Operation	Easy	Easy	Easy	
Validation	Easy	Easy	Easy	
Scalable				Yes
Execution Time (Scan)				
Test Bed	125 sec	72 sec	42 sec	
HTML Pages	49 sec	34 sec	24 sec	
Images	79 sec	48 sec	28 sec	
Moving Images/Sound	79 sec	48 sec	28 sec	
Execution Time (Verify)				
Test Bed	119 sec	71 sec	51 sec	

Feature	Pentium MMX 266 MHz	Pentium II 333 MHz	Pentium 4 1.7 GHz	Other
HTML Pages	44 sec	26 sec	26 sec	
Images	78 sec	48 sec	26 sec	
Moving Images/Sound	78 sec	46 sec	26 sec	
Multiple Platform Compatibility				
DOS				No
Windows				Yes
NT				Yes
Unix				
Integrity Check Value				
CRC32				Yes
MD5				No
SHA-1				No

3.3 Veracity

3.3.1 Background

Veracity is a digital quality assurance tool that uses cylic redundancy codes or hash digest algorithms to detect unauthorized changes in a computer file system or digital documents stored on fixed and removable storage media.[4] Veracity supports a network monitoring system as well as Veracity Personal on a standalone computer. The SIA's archival preservation of Web resources does not involve a live, operational system so this review focuses on Veracity Personal, which utilizes a "Local Agent" functionality.

There are two primary local agent functions in Veracity Personal – Snapshot and History. In the Snapshot mode, after selection of a directory, folder, or sub-folder, clicking on the create snapshot button generates the computation of a file integrity value (CRC32, MD5, or SHA-1) for each document in the designated directory, folder, or sub-folder. The Snapshot pull-down menu includes a "Check Integrity" command that when clicked, generates a second hash digest of the selected director, folder, or sub-folder and compares it with the original hash digest. If there is a discrepancy of a single bit between the two items, then a screen message displays the name

[4] Veracity is an Australian based company. More information about Veracity can be obtained at http://www.veracity.com.

of the document and the date and time that a change occurred. If there is no discrepancy, then "Identical" message is displayed on the screen. Each time a Snapshot is created, refreshed, or checks, this information is logged in a journal file attached to the Snapshot. Clicking on the "View log" in the "History pull down menu" displays this log. The hash digest of each document is not directly viewable but can be retrieved as text file from the Snapshot file (.vs.) in the Veracity sub-directory.

It should be noted that Veracity Personal does not support a centralized repository where all the directories, folders, and sub-folders are stored. For archival preservation purposes, a separate archival preservation directory with folders, organized perhaps by date, should be created and written to removable storage media. The preservation directory could be updated periodically as new Web site material comes into the custody of the archives.

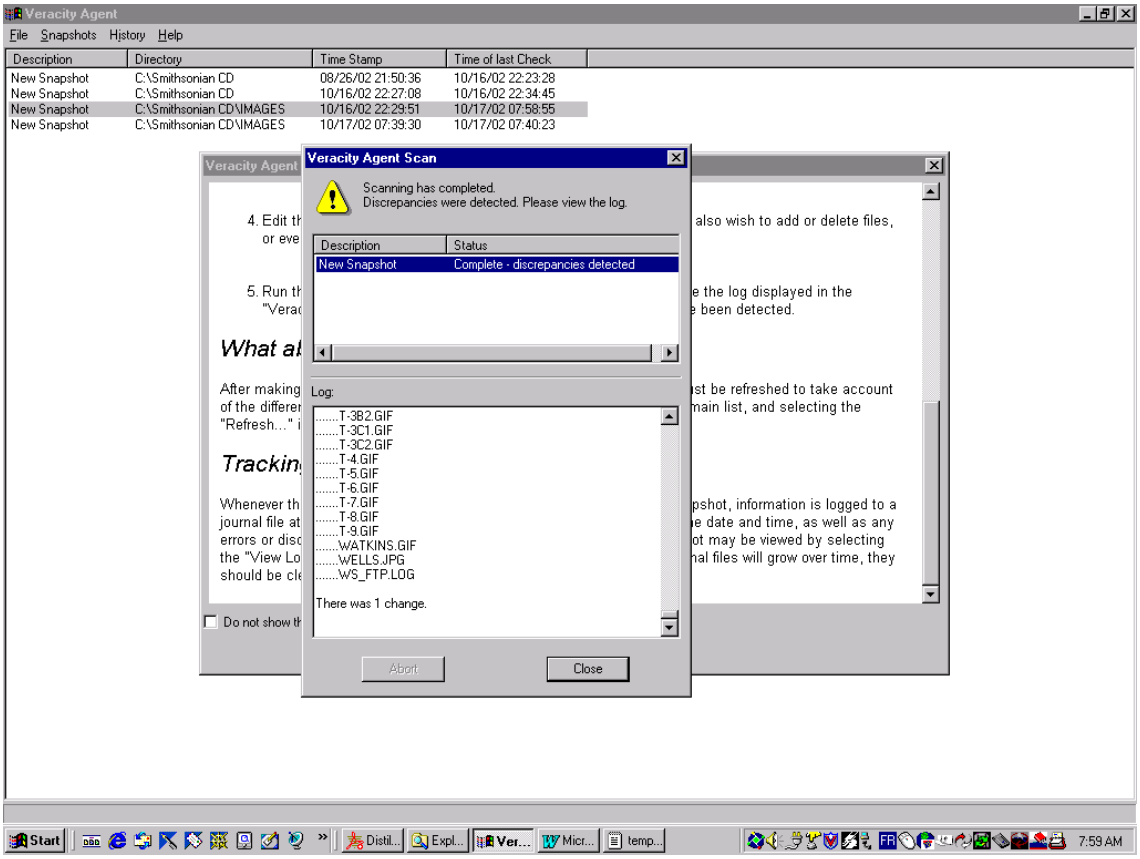
Veracity Personal has another useful function called "File Transfer Verification," which monitors the transfer of files to a different computer system. There are two options: the same operating system and a different operating system. In the instance of the same operating system (i.e., Windows 98), Veracity creates a snapshot of the directory tree. After the files are transferred, a second snapshot of the files in the target directory is computed. A Veracity check identifies any changes that occur during the transfer.

3.3.2 Ease of Use

Veracity Personal is easy to install and use after downloading it from the Veracity Web site. An executable file is unpacked and automatically installs all executable and support files in a predefined directory and sub directory structures. When Veracity Personal is opened, a Veracity Guide is displayed that offers a brief explanation of the Snapshot and History functions. Clicking on the Snapshot pull-down menu displays Create, Check Integrity, Refresh, and Delete options. After clicking on the create option, the appropriate directory, folder, or sub-folder can be selected and with one click, the hash digest is executed. Upon completion of this function, all of the documents in the sub-folder are displayed along with the message "Snapshot successfully taken." Once a snapshot has been successfully taken, the Check Integrity option can be initiated. The Check Integrity option generates a second hash digest and then compares it with one the hash digest for this sub-folder, folder, or director previously computer. If the two hash

digests are identical, the list of documents is displayed along with the message "Identical: There were no differences." If the two hash digests are not identical, the name of the specified document in which the differences were found is listed along with a message that specifies how many changes were made. The Local Agent screen also displays the time and date the snapshot was successfully taken along with the time that the last Check Integrity was executed. All of the information such as "Snapshot successfully taken" that is displayed on the screen is stored in one or more Veracity sub-directories that can be accessed through Windows Explorer and displayed as text. It can also be printed or copied to another directory.

Figure 3.2 VERACITY SNAPSHOT CHECK INTEGRITY



3.3.3 Scalable

Veracity Personal can work with a single server, end user's PC, physical or logical drive, directory or part of the directory structure. In all cases Veracity Personal can provide integrity

check values for designated sub-folders, folders or directories that are written on a hard disk, removable magnetic media, or CD-R media.

3.3.4 Execution Time for Veracity

The execution time for Veracity is very fast, especially when the MD5 algorithm is used. Typically, the execution time for SHA-1 takes about three times longer.

Table 3.3 Execution Time for Veracity Personal

Veracity	Pentium MMX 266 MHz	Pentium II 333 MHz	Pentium 4 1.7 GHz
Hard Disc All files	Snapshot 2 min 47 sec Check 2 min 18 sec	Snapshot 2 min 19 sec Check 1 min 47 sec	Snapshot 1 min 8 sec Check 1 min 3 sec
Hard Disc HTML files	Snapshot 11 sec Check 12 sec	Snapshot 7 sec Check 4 sec	Snapshot 5 sec Check 4 sec
Hard Disc Graphic files	Snapshot 2 min 11 sec Check 36 sec	Snapshot 1 min 8 sec Check 14 sec	Snapshot 46 sec Check 10 sec
CD-RW All files	Snapshot 6 min 21 sec Check 5 min 14 sec	Snapshot 4 min 2 sec Check 3 min 18 sec	Snapshot 2 min 1 sec Check 1 min 39 sec
CD-RW HTML Files	Snapshot 44 sec Check 34 sec	Snapshot 26 sec Check 20 sec	Snapshot 23 sec Check 9 sec
CD RW Graphic files	Snapshot 4 min 12 sec Check 3 min 13 sec	Snapshot 3 min 11 sec Check 2 min 2 sec	Snapshot 2 min 04 sec Check 1 min 14 sec

3.3.5 Multiple platform compatibility

Veracity is available on a wide variety of platforms. These platforms include:

Windows 95/98/ME
Windows NT 4/2000
Windows XP
Mac OS X
Solaris (SPARC)
Solaris (386)
Compaq Himalaya OSS
Tru64 Unix

Open VMS (Alpha)
Open VMS (VAX)
IMM AIX 4.1.4.3
HP-UX 11.0
MS-DOS
Linus (386)
BSD/OS
NetBSD (386)
FreeBSD
IRIX

3.3.6 Integrity Check Robustness

Veracity allows users to select a specific hash digest algorithm or CRC. The hash digest algorithms supported are SHA-1, MD2, MD4, MD5, HAVAL (four variants) and Snefru (four variants). Veracity also supports the use of the CRC16 and CRC32 integrity check values. Given the inherent weakness of CRC16 and CRD32, their use in Veracity to protect the integrity of digital objects is problematic for archival storage. MD5, which is reported to be over three times faster than SHA-1, is the default hash digest algorithm in Veracity for Windows.[5] The National Institute of Standards and Technology advises against the use of MD5 for any sensitive digital information because the hash algorithm has been "broken. Most experts believe that with today's technology it is computationally infeasible for two different documents to have the same SHA-1 hash digest fingerprint.

3.3.7 Veracity - Technical Summary and Evaluation

Table 3.4 summarizes the technical assessment and findings regarding Veracity as a potential digital quality assurance tool to protect the integrity of SIA Web resource materials.

[5] The current version of Veracity Personal does not support changing the default hash algorithm. Instead, the change has to be made in the DOS version of Veracity. Veracity provides instructions on how to make the change.

3.4 VERACITY PERSONAL TECHNICAL EVALUATION SUMMARY

Feature	Pentium MMX 266 MHz	Pentium II 333 MHz	Pentium 4 1.7 GHz	Other
Ease of Use				
Installation	Easy	Easy	Easy	
Operation	Easy	Easy	Easy	
Validation	Easy	Easy	Easy	
Scalable				Yes
Execution Time Snapshot*				
Test Bed	2 min 47 sec	2 min 19 sec	1 min 08 sec	
HTML Pages	11 sec	7 sec	5 sec	
Images	2 min 11 sec	1 min 08 sec	46 sec	
Moving Images/Sound	14 sec	9 sec	7 sec	
Execution Time Check*				
Test Bed	2 min 18 sec	1 min 47 sec	1 min 03 sec	
HTML Pages	12 sec	04 sec	04 sec	
Images	36 sec	14 sec	10 sec	
Multiple Platform Compatibility				
DOS				Yes
Windows				Yes
NT				Yes
Unix				Yes
Integrity Check Value				
CRC32				Yes
MD5				Yes
SHA-1				Yes

*These execution times do not take into account the use of CD-RW, which increases the execution time.

3.5 Digital Notary Service

3.5.1 Background

Surety's Digital Notary® Service supports a secure, and reliable notary service that can verify a document has not been changed through the use of hash digest and digital time-stamping technology. This verification process involves two Digital Notary functions: (1) Notarization that creates a hash digest of each document and time stamps each hash digest and (2) validation of the integrity of each document that was "hashed" and digitally time stamped.

Notarization. This function is initiated by selecting the document or documents to be notarized and then clicking on the Surety button, which automatically creates a hash digest of the selected document(s) and transmits the hash digest(s) to the Digital Notary Service using the computer's network connection, where each one is assigned a time and date at the moment a Notary Record is created, which typically happens at one second intervals.[6] All of the hash digests received during a one second interval are combined into a super hash value. This sequence is repeated each second so that at any given point in time a super hash value represents an aggregation of all root hashes computed since 1992 (when the service first began time-stamping hash digests). The system then generates a digital notarization certificate that contains both the original hash digest submitted with a time and date stamp and the intermediate hash value and path followed when the hashes were combined into a super hash. Each week Surety Technologies publishes the super hash value for that week in the Commercial Notices section of the national edition of the *New York Times*. This weekly super hash value represents all of the hash values submitted that week as well as all other previous hash values submitted. In effect, Surety Technologies offers an unbroken chain of digital custody of hash digest values.

Validation.[7] The second function is validation of the integrity of a single document or an aggregation of documents that already have been notarized. After selecting the appropriate file(s) and notary record(s) in the local Repository, the Validate button is clicked and the tool does all of the work. It hashes the document and compares that hash value to the hash value for that specific document stored in the Notary Record. If the two hash values match, it sends the hash value and the time stamp to the Surety Validation server where it is combined with the super hash that was computed immediately before the record was originally notarized. This super hash is then compared to the super hash that was generated at the time of notarization and is stored in the Universal Registry. If there has been any change in the hash value and notarization value of a single document, the two super hashes will not match. If they match, a green check mark indicating validation of the document is displayed. If they do not match, then a red x mark indicating non-validation is displayed for that particular document.

[6] Typically, there are numerous hash digests transmitted to the notarization service each second
[7] This validation service is available at no cost.

A second way that the Digital Notary validates a single document, or aggregation of documents, is through an off-line validation service Surety makes a copy of its Universal Registry available on a CD along with a set of command line tools that can validate a file without the Surety servers. It involves the same selection and clicking functions as those used in the on-line validation service. Surety claims that this service is the only long-lived cryptographic service and this set of tools allows anyone to validate a file as long as they have the NY Times for the week in which the file was first fingerprinted and the Universal Registry including that period.

Surety Technologies distributes, without cost, the software for generating hash digests, transmitting them to the Surety server for time-stamp notarisation, and subsequent validation. Surety Technologies does charge a fee for each time-stamping transaction. This per transaction cost fee can vary between 5 cents to 50 cents, depending upon the volume of transactions. In some instances, Surety Technologies may negotiate a site license with an unlimited number of transactions annually.

3.5.2 Ease of use

The Digital Notary is a user-friendly digital quality assurance tool that can be installed directly from a CD or downloaded from Surety Technologies.[8] It runs in a Windows environment so both installation and operation of the software are straightforward and require no specific expertise or skill. In addition, Digital Notary provides a robust help functionality that addresses many common problems that users encounter. The validation service involves additional steps, depending upon whether the on-line or off-line service is used.

Figure 3.3 DIGITAL NOTARY VALIDATION

[8] Surety does charge a modest fee for the time-stamping of hash digests. and subsequent validation of the integrity of hashed documents.

3.5.3 Scalable

The Digital Notary provides full scalability that takes into account user requirements. It stores all results in a central repository that supports projects of various scales. The central repository is especially useful for large scale projects although it is equally useful for one scale ones. This is particularly evident in the way Digital Notary creates a notary file for each document (rather than generating reports behind the scene), which is much more convenient for user retrieval. Smaller projects will likely take longer using Digital Notary due to overhead from the additional steps that are taken, but such gains would be marginal. Large-scale projects benefit greatly from Digital Notary's repository and simple yet effective report generating mechanism.

3.5.4 Execution time for Digital Notary

The execution time for the Digital Notary is considerably longer than for the File Check and Veracity. There are two reasons for this. First, two hash algorithms are used (MD5 and SHA-1). Second, the amount of time required for time-stamping, which is done through an Internet connection, can vary significantly depending upon how many subscribers are accessing the Internet.

Table 3.5 Digital Notary Execution Time

Digital Notary	Pentium MMX 266 MHz	Pentium II 333 MHz	Pentium 4 1.7 GHz
	Fingerprint, Notarize, Validate	**Fingerprint, Notarize, Validate**	**Fingerprint, Notarize, Validate**
Hard Disc All files	FP & Not. 20 min 36 sec Validate 39 – 104 minutes*	FP & Not. 8 min 50 sec Validate 39 – 104 minutes*	FP & Not. 3 min 31 sec Validate 39 – 104 minutes*
Hard Disc HTML files	FP & Not.11 min 50 sec Validate 5 – 11 minutes*	FP & Not. 6 min 37 sec Validate 5– 1 minutes*	FP & Not. 2 min 34 sec Validate 5– 2 minutes*
Hard Disc Graphic files	FP & Not. 15 min 55 sec Validate 24 – 64 minutes*	FP & Not. 5 min 23 sec Validate 24 – 64 minutes*	FG & Not. 2 min 32 sec Validate 24– 64 minutes*
CD-RW All files	FP & Not. 22 min 24 sec Validate 39 – 104 minutes*	FP & Not. 12 min 18 sec Validate 39 – 104 minutes*	FP & Not. 8 min 51 sec Validate 39 – 104 minutes*
CD-RW HTML files	FP & Not. 13 min 42 sec Validate 5– 11 minutes*	FP & Not. 8 min 5 sec Validate 5 – 11 minutes*	FP & Not. 5min 54 sec Validate 5– 11 minutes*
CD-RW Graphic files	FP & Not. 17 min 8 sec Validate 24 – 64 minutes-	FP & Not. 8 min 39 sec Validate 26 – 64 minutes	FP & Not. 5 min 28 sec Validate 24 – 64 minutes

*On-line validation takes between 39 to 104 minutes, depending upon the transmission speed of data to the Surety server. The actual processing time on the Surety server is about 1.25 seconds per transaction.

3.5.5 Multiple Platform Compatibility

Digital Notary runs exclusively in a Windows environment, which includes Windows NT. In this regard, Digital Notary has limited multiple platform compatibility.

3.5.6 Integrity Check Robustness

The Digital Notary has the most powerful and robust integrity check algorithm of the three software packages evaluated. The Digital Notary first computes a MD5 hash digest (128 bits) and then computes a SAH-1 hash digest (160 bits). These two hash digests are concatenated

into a single integrity check value of 288 bits, which practically speaking means that with today's technologies it would be easier to find a specific atom in the universe than to "break" the concatenated hash value.

3.5.7 Digital Notary - Technical Summary and Evaluation

The execution time of Digital Notary is considerably longer than the time for other software tools largely because it includes digital-time-stamping. In addition, Digital Notary offers benefits such as a graphical user interface and a unique design of a binary tree - hash values can be obtained for a single file (record), a subdirectory, or directory, a super hash or an entire Web site. In this sense the scalability of the Digital Notary is virtually unlimited. Although the Digital Notary is not cross platform, it does run in a Windows environment, which is the dominant computing platform today.

Table 3.6 summarizes the technical assessment and findings regarding Digital Notary as a potential digital quality assurance tool to protect the integrity of SIA Web resource materials.

Table 3.6: DIGITAL NOTARY TECHNICAL EVALUATION SUMMARY

Feature	Pentium II	Pentium III	Pentium IV	Other
Ease of Use				
Installation	Easy	Easy	Easy	
Operation	Easy	Easy	Easy	
Validation	Easy	Easy	Easy	
Scalable				Yes
Execution Time (Finger Print and Notarize				
Test Bed	20 min 53 sec	8 min 50 sec	3 min 31 sec	
HTML Pages	11 min 50 sec	6 min 37 sec	2 min 34 sec	
Images	15 min 55 sec	8 min 23 sec	3 min 20 sec	
Moving Images/Sound				
Execution Time (Validate)				
Test Bed	39 – 104 minutes*	39 – 104 minutes*	39 – 104 minutes*	

Feature	Pentium II	Pentium III	Pentium IV	Other
HTML Pages	4 - 11 minutes*	4– 11 minutes*	4 - 11 minutes*	
Images	21 – 64 minutes*	21 – 64 minutes*	21 – 64 minutes*	
Multiple Platform Compatibility				
DOS				No
Windows				Yes
NT				Yes
Unix				No
Integrity Check Value				
CRC32				No
MD5				Yes
SHA-1				Yes

3.6 Comparison of File Check, Veracity, and Digital Notary Digital Quality Assurance Tools

Table 3.7 below summarizes the weaknesses and strengths of each of the three evaluated software packages based on the five evaluation criteria. Note that two of the software packages has at least one substantial weakness. File Check uses a CRC32 to establish the integrity of electronic records but a CRC is "collision prone" and it is computationally possible to replicate the original document from a CRC32 integrity check value. The execution of the Digital Notary takes longer than either of the other two software packages but this is because a time-stamp is also generated, which with the concatenated MD5 and SHA-1 hash digests, offers an integrity check value that is computationally infeasible to replicate.

Table 3.7: Strengths and Weaknesses of File Check, Veracity, And Digital Notary

Software	Weakness	Strength

Software	Weakness	Strength
File Check		
Ease of use		**X**
Scalable		**X**
Execution time		**X**
Multiple platform compatibility		**X**
Integrity value robustness	**X**	
Veracity Personal		
Ease of use		**X**
Scalable		**X**
Execution time		**X**
Multiple platform compatibility		**X**
Integrity value robustness		**X**
Digital Notary		
Ease of use		**X**
Scalable		**X**
Execution time	**X**	
Multiple platform compatibility	**X**	
Integrity value robustness		**X**

4 FINDINGS AND RECOMMENDATIONS

4.1 Quality Assurance Process

The use of digital quality assurance tools to verify the integrity of electronic records is a vital part of a recommended archival preservation process and should occur within a quality assurance process environment, which should occur as the SI implements the recommendations in the "Independent Evaluation of the Smithsonian Institution's Information Security Program." However, these tools are effective only as long as the underlying bit stream of electronic records undergoes no change, which in fact happens when electronic records are migrated from one technology neutral file format to its technology neutral replacement. [1] This proposed quality assurance environment for the SIA entails two instances of media migration over the course of two decades. Each instance of media migration involves generation of hash digests before and after media migration that are compared to confirm the integrity of the records. Thus, there would be two instances of hash digest generation and two instances of validation.

4.2 Resource Allocation Metric

The purpose of this digital quality assurance resource allocation metric is to facilitate estimating the costs of using digital quality assurance tools in the archival preservation of a specified number of Web pages. This metric encompasses all Web pages (i.e., HTML, GIF, JPEG, etc) and assumes the following:

Static Web pages,
One PC workstation that is at least a Pentium MMX 233 MHz is available,
Staff familiarity with Windows operations,
Two instances of media migration, and
Access to a magnetic tape drive and CD-burner,

The resource allocation metric includes scan execution time, validation execution time, software cost, and any additional cost associated with a special feature (i.e., time-stamping). As noted in

the above assumptions, two instances of media renewal are projected over the next twenty years. Each instance requires computation of a hash digest for each document before and after media renewal. With 10,000 Web pages this means that a total of 40,000 hash digests will be computed and compared.

Table 4.1: Digital Quality Assurance Resource Allocation Metric

Digital Quality Assurance Resource Allocation Metric			
Feature	**File Check**	**Veracity**	**Digital Notary**
Scan time*	1 min 8 sec per 1,000 items	1 min 31 sec per1, 000 items	9 min. 15 sec per 1,000 items
Verify time*	1 min 4 sec per 1,000 items	1 min 13 sec per 1,000 items	18 – 56 min per 1,000 items
Software	Free	$65 per workstation	Free
Time-stamping	NA	NA	$50 per 1,000 transactions

*Scan and verify are generic terms that respectively describe the process of generating a CRC value or hash digest (and time-stamping as appropriate) and generating a second CRC or hash digest that is compared with the previous CRC or hash digest to confirm that no change has occurred.

Table 4.1 highlights two significant considerations that come into play when the quality assurance resource allocation metric is applied to 10,000 Web pages. First, there is a substantial difference in the amount of time required to generate and compare CRC values or hash digests. File Check would require approximately 1 hour and 6 minutes to generatge and validate 40,000 CRC values and Veracity would require approximately 2 hours and 50 minutes to generate and validate 40,000 hash digests. The Digital Notary would take between 12 to 24 hours to generate and compare 40,000 hash digests. This disparity in the execution time of each software package is a function of the robustness of the integrity check value used. File

[1] Digital time-stamping software such as the Digital Notary could be modified to support the concatenation of before and after conversion hash digests into a superhash value.

Check uses CRC32, which is "collision prone" and easily replicated. Veracity uses SHA-1, which is "computationally infeasible" to reverse or replicate. The Digital Notary combines two hash digest algorithms – MD5 and SHA-1 – with digital time-stamping to produce integrity check values that are even more "computationally infeasible" by several orders of magnitude to reverse or replicate. The second consideration is cost of the software. File Check is free, Veracity costs $65.00, and although the Digital Notary software is free, it would cost approximately $2,000 to notarize (i.e., time-stamp) 40,000 Web pages. Of course, this is an on-going cost that occurs during each instance of media migration.

4.3 Implementation Options

This section reviews three implementation options for confirming the integrity of Web source material that is transferred to the SIA and migrated from old storage media to new storage media. Each option incorporates the resource allocation metric discussed above.

Implementation Option 1

This implementation option calls for the use of File Check to confirm the integrity of 10,000 Smithsonian Web pages when they are migrated from old media to new media.

Advantages

Easy installation,
Fast execution time of 47 minutes
User friendly Windows environment,
Runs on multiple platforms,
Scalable,
Screen displays are easily interpreted, and
Software is free

Disadvantages

CRC32 is a relatively weak integrity check value

Implementation Option 2

This implementation option calls for the use of Veracity Personal to confirm the integrity of 10,000 Smithsonian Web pages when they are migrated from old media to new media.

Advantages

Easy installation,
Fast execution time of approximately 55 minutes
User friendly GUI environment,
Runs on multiple platforms
Scalable,
MD5 and SHA-1 hash algorithms are available,
History file, including hash digests for each document, can be retrieved, and
One-time software cost of $65 per seat.

Disadvantages

Individual documents within a folder or sub-folder cannot be selected for snapshot execution,
Verify reports are very brief, and
Overall display of scan and verify results are not easily interpreted by individuals unfamiliar with Veracity functionalities.

Implementation Option 3

This implementation option calls for the use of the Digital Notary to confirm the integrity of 10,000 Smithsonian Web pages when they are migrated from old media to new media.

Advantages

Easy installation
User friendly GUI environment
Well-designed scalability with a unique binary tree hashing system,
Combined MD5 and SHA-1 hash digests with time-stamping,
Quality assurance tools are the most robust of the three software tools,
Software is free,
Issuance of a single certificate for each record provides "pin point" monitoring, and

Validation of one or more electronic records can be done on-line or off-line,

Disadvantages

Slow execution time of between 9 and 22 hours to notarize and validate 10,000 Web pages,

Managing the certificate of each individual electronic record can be challenging, especially when the volume of certificates is large, and

Digital Notary charges a fixed fee for each time-stamping instance.

Despite its advantages, File Check cannot be recommended because of its use of CRC32, a very weak digital quality assurance tool. The choice, therefore, is between Veracity and The Digital Notary. The combination of MD5 and SHA-1 hash digests, with the time-stamping functionality of the Digital Notary, makes a very powerful digital quality assurance tool. If this functionality were enhanced to support the validation of electronic records when they are migrated to new technology neutral formats, then its benefit to the SIA would be substantial. Without this enhanced functionality, the benefit of using the Digital Notary to confirm the integrity of media migrations is problematic.

4.4 Recommendation

Implement a digital quality assurance program to confirm the integrity of electronic records when they are migrated to new media,

Adopt Veracity for Windows (10/3/2002) as the digital quality assurance tool to confirm the integrity of electronic records as they are migrated to new media over the next five years, and

As digital quality assurance software tools that can confirm the integration of electronic records migrated to new technology neutral formats become available, consider adopting one of them.