Analysis of Workload Behavior in Scientific and Historical Long-Term Data Repositories

IAN F. ADAMS, University of California, Santa Cruz
MARK W. STORER, NetApp
ETHAN L. MILLER, University of California, Santa Cruz

The scope of archival systems is expanding beyond cheap tertiary storage: scientific and medical data is increasingly digital, and the public has a growing desire to digitally record their personal histories. Driven by the increase in cost efficiency of hard drives, and the rise of the Internet, content archives have become a means of providing the public with fast, cheap access to long-term data. Unfortunately, designers of purpose-built archival systems are either forced to rely on workload behavior obtained from a narrow, anachronistic view of archives as simply cheap tertiary storage, or extrapolate from marginally related enterprise workload data and traditional library access patterns.

To close this knowledge gap and provide relevant input for the design of effective long-term data storage systems, we studied the workload behavior of several systems within this expanded archival storage space. Our study examined several scientific and historical archives, covering a mixture of purposes, media types, and access models—that is, public versus private. Our findings show that, for more traditional private scientific archival storage, files have become larger, but update rates have remained largely unchanged. However, in the public content archives we observed, we saw behavior that diverges from the traditional "write-once, read-maybe" behavior of tertiary storage. Our study shows that the majority of such data is modified—sometimes unnecessarily—relatively frequently, and that indexing services such as Google and internal data management processes may routinely access large portions of an archive, accounting for most of the accesses. Based on these observations, we identify areas for improving the efficiency and performance of archival storage systems.

Categories and Subject Descriptors: D.4.2 [**Operating Systems**]: Storage Management; H.3.4 [**Information Storage and Retrieval**]: Systems and Software—*Performance evaluation*; D.4.8 [**Operating Systems**]: Performance—*Measurements*

General Terms: Measurement, Performance

Additional Key Words and Phrases: Archival storage, tertiary storage, trace analysis

ACM Reference Format:
Adams, I. F., Storer, M. W., and Miller, E. L. 2012. Analysis of workload behavior in scientific and historical long-term data repositories. ACM Trans. Storage 8, 2, Article 6 (May 2012), 27 pages.
DOI = 10.1145/2180905.2180907 http://doi.acm.org/10.1145/2180905.2180907

This research was supported in part by the National Science Foundation under grants CNS-0917396 (part of the American Recovery and Reinvestment Act of 2009 [Public Law 111-5]) and IIP-0934401, and by the industrial sponsors of the Storage Systems Research Center and the Center for Research in Intelligent Storage.

Authors' addresses: I. F. Adams and E. L. Miller, University of California, Santa Cruz, 1156 High Street, Santa Cruz, CA 95064; email: iadams@soe.ucsc.edu, elm@ucsc.edu; M. W. Storer, NetApp, 495 East Java Drive, Sunnyvale, CA 94089; email: mark.storer@netapp.com.

DOI 10.1145/2180905.2180907 http://doi.acm.org/10.1145/2180905.2180907

1. INTRODUCTION

Archival storage has traditionally been viewed as inexpensive, tertiary storage [Jensen and Reed 1993; Miller and Katz 1993; Smith 1981a, 1981b]; however, this anachronistic definition is too narrow to accurately describe the current gamut of long-term storage use cases. Changing storage media and access methods and increasingly digital workflows have radically affected how long-lived content is created, stored, and published. In the business arena, data preservation is often mandated by law, and data mining has proven to be a boon in shaping business strategy. For individuals, archival storage is being called upon to preserve sentimental and historical artifacts such as photos, movies, personal documents, and medical records [Chronicles 2011; HIPAA 1996]. Finally, in one of the biggest disruptive shifts in recent history, the Internet has radically changed how users interact with data, catalyzing an explosion in the number of publicly-accessible, long-term content repositories [Alaska State 2010; Cornell University Library 2010; New York State 2010; NOAA 2010; ORNL 2010; Washington State 2010].

While the growing variety of archival use cases share a common characteristic of long data lifetimes, understanding of their respective workloads is out of date at best, and non-existent at worst. There have been no detailed studies of archival storage activity in nearly two decades; the most recent studies were of super-computing environments in the early 1990s [Jensen and Reed 1993; Miller and Katz 1993]. Dayal [2008] published a study of high end computing storage at rest and investigated a few archival systems, but the study largely focused on high-performance storage systems and showed no trends over time. As a result, current work in long-term storage [Baker et al. 2005; Storer et al. 2008] relies upon questionable workload assumptions: observations of out-dated archives with different media types and purposes, marginally related studies of shorter-term enterprise workloads, or access patterns found in traditional library data stores. In contrast, a number of general purpose and high performance workload studies have been published in recent years, demonstrating the value of up to date, empirical data [Agrawal et al. 2007; Anderson 2009; Leung et al. 2008; Roselli et al. 2000].

To close this knowledge gap, we present a study of *long-term data repository* characteristics and workload behavior covering between one and three years of activity in several long-term data stores. The archives we chose allow us to compare and contrast a range of workloads found in tertiary storage and public content repositories, comprising prime examples of use cases aimed at preserving data with indefinite life times. One archive stores an approximately 1.3 PB scientific data set from Los Alamos National Laboratory (LANL), spread across disk and tape, with file metadata crawl summaries covering 1 year; this store is most similar to those from prior archival storage studies and typifies the tertiary storage system use case. Another store, representative of the expanded role of long-term storage, is the Washington State Digital Archives [Washington State 2010]; we analyzed 2.5 years of access logs and record metadata from this store. The third archive is a repository of water table reports from the California Department of Water Resources [California DWR 2010], with logs covering 3 years of activity. We want to stress, however, that our study is not an exhaustive examination of the archival storage space. There are many cases now grouped in the archival space. These include backup [Lillibridge et al. 2003; Zhu et al. 2008], it compliance [HIPAA 1996; Sarbanes-Oxley 2002], and personal archiving [Chronicles 2011]. Adding further complexity to the picture is the rise of remote cloud storage and backup services such as Dropbox [2011], and Amazon's S3 [Amazon 2011] that can be used explicitly or in an ad-hoc manner as remote archive. These use cases and services have significantly different intent and thus may exhibit different behavior, warranting their own examination as well.

Our analysis of the LANL metadata summaries revealed that tertiary storage archives have changed in a number of ways. First, the current LANL data set is considerably larger than the NCAR system studied in 1993 [Miller and Katz 1993], over 1 PB in 2010, as compared to NCAR's 25 TB in 1992. Further, this archive is merely one of many within LANL. Second, the ratio of disk to tape has changed from the NCAR system's ratio of 1:262, to the LANL system ratio of 1:3.3 at the end of our study. Interestingly however, despite the increased use of hard disks, overall update behavior is largely similar to previous studies. Third, the typical file size has grown considerably, although many of the files are quite sparse.

Our results with the public water and historical archives mark one of the first critical examinations of the emerging class of publicly accessible long-term data repositories, and reveal that their behavior deviates radically from conventional wisdom. First, the majority of data in both the Washington State and water datasets were updated at least once, and often several times over the course of our traces. This finding directly contradicts the widely held belief that archives are "write-once". Second, we found that large batch processes routinely touch vast amounts of repository data, dominating traffic to publicly accessibly archives. This behavior suggests that a separate batch interface for low-priority accesses could provide significant benefit. Third, we found that even though some items within the data repositories are moderately more popular, the distribution of accesses is extremely long tailed, reducing the effectiveness of LRU caching for reads. Fourth, we found that accesses exhibit strong content locality–that is, semantically similar content—within user sessions, though a variety of content tends to be retrieved across user sessions. This suggests that grouping data based on semantic content could yield performance and efficiency gains.

The rest of the article is arranged as follows. Section 2 further illustrates the current gap between archival systems, and our understanding of long-term workloads. Section 3 describes each of our datasets included in our study, as well as the traces collected over them. Next, in Section 4, we present our observations. Finally we discuss the implications of our observations and their implications upon archival storage system design in Section 5, and conclude in Section 6.

2. RELATED WORK

In this section, we present a brief timeline of workload studies, illustrating that our understanding of long-term data behavior predates the ubiquity of the Internet, and the expanding role of archival storage. As a result, relevant systems utilize behavior assumptions that are out-dated, and overly narrow in scope. Traeger et al. [2008] surveyed many earlier studies and benchmarking methods, highlighting the same deficiencies we point out in regards to outdated studies and methodologies, albeit none that specifically focus on archival storage.

The first generation of studies date back to 1981 [Smith 1981a, 1981b]. In those studies, Smith studied the file system of the Stanford Linear Accelerator Center in the context of optimizing file migration algorithms, and defined the basic patterns of tertiary storage behavior.

The next generation of studies occurred in the early nineties. Workloads investigated included workstation file systems [Gibson and Miller 1998; Gibson et al. 1998; Strange 1992], and mixed disk and tape tertiary storage [Jensen and Reed 1993; Miller and Katz 1993]. Miller and Katz [1993] examined storage use at the National Center for Atmospheric Research (NCAR), which at the time consisted of approximately 100 GB of disk, and 25 TB of tape. In contrast, the LANL corpus we studied is expected to grow to 2 PB and beyond during 2011, and is only one of multiple archives within LANL.

Table I. Corpora Overview

Owner	Name	Size	Records	Access	Media	Data Types
LANL	SCIENTIFIC	1.3 PB	60,000,000	Private	Disk, Tape	Multiple
WA State Archive	HISTORICAL	Unknown	28,000,000	Public	Disk	Multiple
Cal. Dept. of Water Res.	WATER	2.6 GB	57,000	Public	Disk	Single

Overview of the corpora and archives covered by this study. We use *Name* to identify the sketch throughout this paper. Data types are the number of different types of records in the corpus.

Finally, the recent past has seen the latest generation of workload studies [Agrawal et al. 2007; Anderson 2009; Leung et al. 2008; Roselli et al. 2000], albeit none that specifically examines long-term data behavior. Thus, while these examinations demonstrate the value of an up to date understanding of storage system behavior, it is difficult to generalize their findings to long-term data repositories. For example, Agrawal et al. examined long-term metadata trends, but for desktop PC workloads, and Anderson examined high performance workload traces from an animation company. Similar studies done over other relevant datasets have proven to be a boon to the research community [Agrawal et al. 2007; Bairavasundaram et al. 2007, 2008; Pinheiro et al. 2007; Schroeder and Gibson 2007; Traeger et al. 2008].

As a result of the research community's lack of analysis of archival storage system behavior, a number of recent long-term preservation systems are based on assumptions that may not be valid, or only pertain to a narrow view of archival storage. For example, many long-term data systems explicitly assume that contents are immutable, or imply this by using WORM media [Baker et al. 2006; Maniatis et al. 2005; Quinlan and Dorward 2002; Storer et al. 2007; You et al. 2005; Zhang et al. 2007]. Others assume that data is rarely read. For example, Pergamum claims dramatic cost savings largely predicated on the ability to keep drives spun down due to low read rates [Storer et al. 2008]. Though these assumptions might hold true, we have no up-to-date knowledge with which to confirm them. It has been over 15 years since the last tertiary storage study, and to the best of our knowledge, there have been no studies of access behavior in modern public content archives.

3. DATASETS

In an effort to achieve consistency in our discussion, we begin by establishing a set of concise definitions. An individual element in a set of data is a *record*. A record may be a file, bitstream, or even a literal SQL record. We refer to a collection of records as a *corpus*, and a copy of that corpus as an *instance*. The hardware and software used to store an instance of the corpus is the *archive*; the long-data lifetimes and relatively short refresh cycle of modern hardware suggest that a corpus will reside on several archives over its lifetime. A *system* is a holistic view of the archives, corpus and potentially even users. Finally, we refer to the aggregate body of knowledge about a system as a *sketch*. A sketch includes trace logs, profiles, record metadata, as well as communication with system architects and administrators. As we found out during our study, communication with the architects and administrators was a necessity in understanding the subtleties of the various data we obtained.

Tables I and II provide an overview of the sources we used to conduct this study, illustrating that the scope of this study is focused on tertiary storage and public content archives. Our first source, from Los Alamos National Labs (LANL), allows us to update our understanding of traditional tertiary storage systems. Our second source, illustrative of the shift the Internet and lowering storage costs have brought, is a public repository of digitized historical documents, the Washington State Digital

Table II. Trace Overview

Corpus	Type	Length	Entry count
SCIENTIFIC	Daily FSStats histograms	13 months	4716
HISTORICAL	User access logs	33 months	5.8 million
HISTORICAL	Record metadata	33 months	28.3 million
WATER	Record update and metadata logs	51 months	900 thousand
WATER	User access logs	33 months	100 thousand

Overview of reports and logs in our sketches, including the duration and number of distinct entries in each log.

Table III. Trace Overview

Histogram type	Description
Reported size	File length returned by `stat`
Allocated space	Number of bytes actually allocated
mtime	File modification times
mtime (KB)	File modification times, grouped by file size
Overhead	Difference between reported size and allocated space

FSstats histogram reports collected over the SCIENTIFIC corpus. One set of histograms covers the entire archive, and the other set is run once for each individual top-level directory, corresponding roughly to specific projects.

Archives [Washington State 2010]. The third source we examine is a publicly accessible repository of water table reports—such as ground water levels and salinity—from the California Department of Water Resources [California DWR 2010]. This source is particularly interesting as it illustrates yet another new direction in the long-term data space; small, per-department specialized content repositories. Understanding the use of these smaller corpora is important for several reasons. First, many corpora may be stored on a single physical archive where the aggregate behavior of many small corpora may be more important than individual corpus behavior. Second, the physical size of a corpus may be completely independent from its relative importance to its users. Third, it demonstrates a new use-case trend in long-term digital storage.

3.1. Los Alamos National Laboratory

The corpus from LANL contains files used in their supercomputing environment. We refer to this as the SCIENTIFIC corpus, and it most closely resembles the structure and intent of the classical view of long-term storage as tertiary storage. The corpus contains approximately 60 million files, totaling 1.3 PB spread across disk and tape. When a user is allocated compute time, he or she is allocated a top-level directory in the archive.

We have 13 months of two daily histogram reports collected over this corpus from a daily crawl of the system's inode metadata by FSstats [Dayal 2008]. One daily report covers the entire file system. The second covers each top-level directory corresponding roughly to summaries of individual projects. Table III describes the histograms we used. Note that atime (access time) tracking was explicitly disabled in the file system, so we could not effectively analyze retrieval patterns.

3.2. Washington State Digital Archives

One public corpus we examined is a collection of digitized, historical artifacts—such as census information, military records, photographs, and land records—stored in an SQL database at the Washington State Digital Archives. We refer to this as the

Table IV. HISTORICAL Corpus Record Metadata

Field Name	Example	Null
Record ID	123555	No
Date 1	10-10-1910	Yes
Date 2		Yes
Type	Marriage Record	No
Ingest date	11-12-2008 12:25:06	No
Modify date	9-4-2009 12:52:00	Yes
Num. of objects	0	No

HISTORICAL corpus record metadata. A yes in the Null column indicates the value may be null. Number of objects is the number of digital objects associated with a record, possibly zero. The two date fields are used to hold record specific dates, such as birth and death times.

HISTORICAL corpus. At the time of capture, it contained approximately 90 million records, 28 million of which are accessible via their public web interface; the rest must be accessed on-site. Records occasionally move between public and private status based on content or explicit request. In this study, we focus on the publicly available records, since this is the only portion of the corpus covered in the provided user access logs.

We obtained two logs for this corpus, spanning September 27, 2007 to June 17, 2010. The first log details per-record metadata, described in Table IV. The second is a user access log that records accesses to individual records. Each record is linked to zero or more digital objects, such as photographs and documents, but each digital object is only associated with one record. The trace does not note whether the digital objects linked to that record were retrieved. Further, while the access log provides information allowing us to group accesses from the same session, we cannot link different sessions to specific individuals or hosts.

It should be noted that our logs only reflect user retrieval of records within the corpus database and do not reflect access to any other content, for example, HTML pages. Additionally our logs do not track the activity from data migration or integrity checking processes. As we discuss further in Sections 4 and 5, these administrative processes actually make up the dominant fraction of accesses.

3.3. California Department of Water Resources

The final corpus in our study, the WATER corpus, is a relatively small set of water table reports consisting of 57,000 records. We have two traces for the WATER corpus. The first is a set of update logs from approximately weekly and quarterly batch scripts. Each update log notes the records written to, the date, and record metadata, summarized in Table V. The second is a set of access traces consisting of a per-user access log, where each entry notes the IP address that retrieved the record, as well as the site, period of record, and record retrieved. As with the HISTORICAL corpus, the logs here do not reflect accesses to general web content, only downloads of the reports themselves. Similarly, if there are any internal indexing or integrity processes running, they are not reflected in our logs.

In the update trace, we identify a unique record using a tuple of site name, period of record, and file name. Complicating this, however, was an intermittent change in file naming conventions that made it difficult to map old names to new, introducing the danger of over-counting files and mapping updates to incorrect file names. To address this, we only count updates to files that map to names in existence on the last day of

Table V. WATER Corpus Record Metadata

Field name	Example
Site	A00268
Site type	Surface Water
Parameter	Flow
Period of record	1997
File name	GW_DEPTH_POINT_DATA
File size	13050
File type	Plot

WATER corpus record metadata, and representative values. Unique records are identified using a (Site, Period of Record, File Name) tuple.

the update log. Though this discards approximately 50% of the 1.7 million updates, it ensures we have both a correct file count, and an accurate lower-bound on file update activity; more updates may have been required to keep the relevant files up to date, but no fewer.

4. ANALYSIS

Our analysis is motivated by a hypothesis covering four primary areas. First, as media capacities and costs have changed, the tertiary storage use case has seen increased use of hard drives. Second, with the broadening variety of archival use cases, “write-once” does not accurately describe modification behavior in all long-term data stores. Third, it is similarly not accurate to characterize all long-term storage as “read-maybe”. Fourth, when looking at system wide record retrieval patterns, locality and “hot-spots” are limited.

We begin by comparing the SCIENTIFIC sketch of the tertiary storage archive at LANL to the archive Miller and Katz [1993] describe at NCAR, since theirs was the most recent study of a large—for the time—tertiary storage system. Following that, we examine update behavior in modern content preservation systems. Finally, we analyze retrievals to examine system wide and user-session access locality and complete our investigation into the validity of the “write-once, read-maybe” assumption [Damoulakis 2007; Storer et al. 2008; Wildani et al. 2009].

4.1. Tertiary Storage Evolution

Compound Annual Growth Rates. Compared to the previous study (NCAR), total corpus size exhibited a compound annual growth rate (CAGR) of 25.8%. The CAGR for tape was only 23.9% compared to 60.2% for disk.

As summarized in Table VI, the SCIENTIFIC corpus from LANL contained about 1.3 PB at the end the report period, and is hosted on 1000 TB of tape, and 285 TB of hard drives. Note however that the LANL corpus is continuing to grow, and is estimated to grow to 2 PB and beyond during 2011. While the LANL administrators have designed the tape library to expand to partially accommodate this growth, overall system growth will still be dominated by disk. Compared to the SCIENTIFIC corpus of the NCAR study, the total corpus exhibited a CAGR of 25.8%. However, most of the growth in capacity occurred in hard drive storage. Compared to the earlier study, our data shows a hard drive CAGR of 60.2%. Note, we restricted our hard drive comparison to a holistic high level view, as the NCAR archive did not use commodity hard drives, relying instead on proprietary storage modules. Interestingly, that hardware

Table VI. Simulation Configuration

	Disk (TB)	Tape (TB)	Total (TB)
1993	0.1	26.2	26.3
2010	300	1000	1300
Ratio	1:3000	1:38.2	1:49.4
CAGR	60.2%	23.9%	25.8%

Tertiary storage comparison between the NCAR system in the 1993 Miller study [Miller and Katz 1993], and the current LANL system, showing the ratio between the 1993 value and the 2010 value, and compound annual growth rate (CAGR).

was fairly old when it was studied in 1993; the IBM 3380 systems [IBM 2010] in the NCAR archive were introduced in 1980, with the final revisions released in 1987.

Similarly, the NCAR archive used IBM 3480 tapes, with a capacity of 200 MB per tape. This format was introduced in 1984, making it 9 years old at the time of the study; by 1992 IBM was producing the fourth generation 3490E IDRC tapes, with a capacity improvement of 12 times that of the 3480. By comparison, the current LANL archive uses the relatively recent LTO-4 format tapes, with 1 TB of capacity, 5000 times the storage of the IBM 3480 tapes. Even with the potentially exaggerated gap in tape capacity, we still see a CAGR of 23.9%, which lags slightly behind the total storage CAGR of 25.8%. The impact of this shift towards more disk-centric tertiary storage on file usage and migration patterns is of keen interest. For example, does a relatively larger and cheaper disk cache lead to files having the ability to reside longer in the cache before migration? If so, does this shift usage patterns at all? Or are files simply idle longer before moving to disk? However, the LANL sketch lacks access time and user behavior information so we must relegate its investigation to future work.

Interestingly, NCAR's use of relatively obsolete equipment does not appear to have been unique. The San Diego Supercomputing Center maintained and used 3 generations of tapes and drives as of 2007 [Moore et al. 2007]. This begs the question of *why* these institutions held onto their tertiary systems so long. Our intuition tells us that the expense and difficulty of upgrading to a new system with large-scale tape based approaches may impact hardware refresh cycles, though admittedly any large scale system may be challenging to upgrade and migrate, and there may be other factors at play as well.

File Sizes. Files between 1 and 2 GB consumed 40% of reported storage, but many large files were sparse.

We next examined file sizes within the corpus by studying the trace at the record level. Figure 1 shows a CDF of file sizes calculated from the last day's histogram in our dataset. Nearly 50% of the data written in the NCAR study consisted of files between 10 and 100 MB; in contrast, we found that 40% of the total reported usage in the LANL corpus consisted of files between 1 and 2 GB.

Interestingly, however, when comparing the reported file sizes to the amount of storage space actually allocated to files, we see that 60% of allocated space is consumed by files sizes between 2 and 8 MB. Thus, while the bulk of storage is consumed by files that are considerably larger than the previous study, those files tend to be sparse; over a petabyte is accounted for when looking at file sizes, but only around 100 TB is actually allocated. This behavior may be partially attributable to scientific supercomputing's use of shared checkpoint files [Bent et al. 2009].

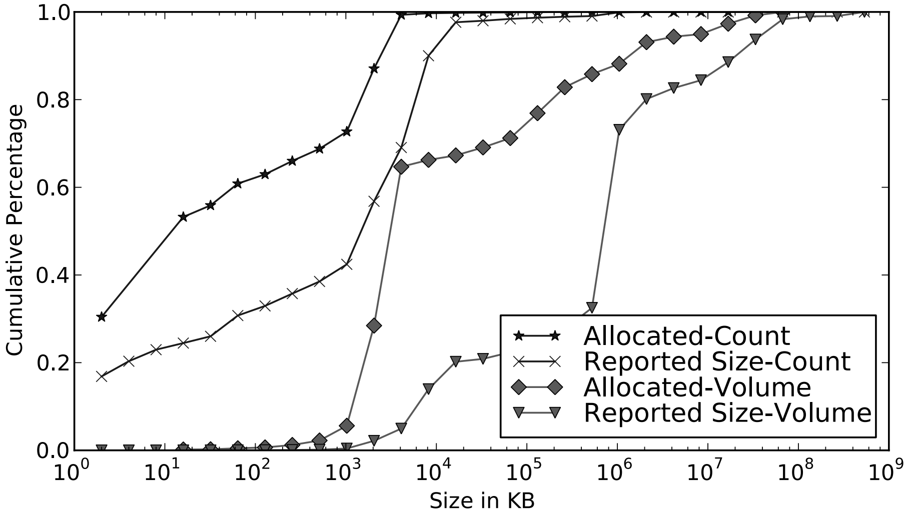


Fig. 1. CDF of reported file sizes and allocated space at the end of the SCIENTIFIC corpus trace. Volume refers to the aggregate amount of storage consumed (in KB), count to the number of files. Sparse files have larger reported file size than allocated space, causing the difference between the curves.

It is important however to note the impact workload has on the specific distributions we are seeing. Dayal surveyed a variety of high end computing (HEC) storage systems using FSstats and found wide variation in file size distributions [Dayal 2008]. However, the general trend towards larger files, and the gap between reported file size and allocated space exist in the systems he surveyed as well.

4.2. Data Modifications

Tertiary Storage Updates. Despite a greater fraction of the system being disk based, traditional tertiary corpora continue to be fairly static; 60% of the LANL content we observed was not modified in nearly a year.

We begin by examining the private SCIENTIFIC dataset, the corpus most similar to the previous study of tertiary storage [Miller and Katz 1993]. Figure 2 is a heat-map showing the fraction of individual records (files) in the corpus that fall into various age ranges over time. The y-axis corresponds to histogram bucket ranges, and the x-axis the day of the trace. The heat-scale on the right maps shade to the total fraction of archive contents. Thus, a corpus that exhibited a high degree of content modification across many files would be warmest along the base of the y-axis; many records would have a recent modification time.

When records are ingested into the archive, they tend to be ingested in batches, and they maintain their existing modification time, explaining why the temperature warms in areas other than the histogram bucket for 0–2 days. At the start of the trace, the archive ingested a batch of files with recent modification times. In Figure 2, this appears as a warm area near trace day 0, for the histogram bucket with files 2–4 days old. As the trace proceeds, those records remain static, and age steadily. This is seen on the heat map by the high temperature region of the histogram moving from the 2–4 day bucket to the 64–128 day bucket as the trace proceeds from day 0 to day 100. Other ingests follow the same behavior, as seen near days 60, 100, 150 and finally 310.

Despite the growing use of hard drives, our results show that aggregate modification behavior in traditional tertiary storage is still much the same as it was at

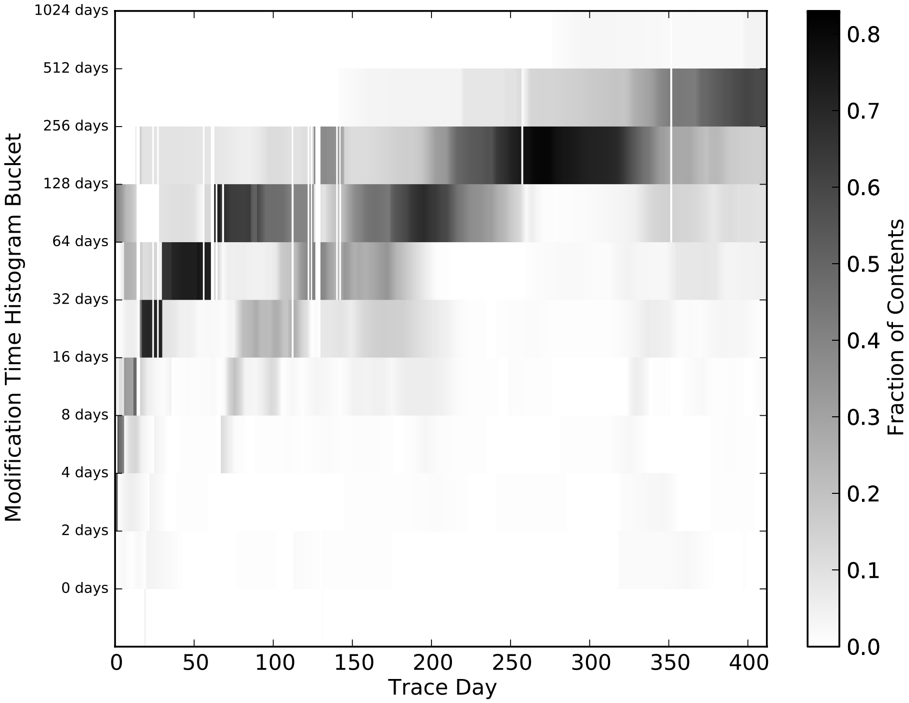


Fig. 2. Heatmap of the SCIENTIFIC corpus's daily record modification histograms over 400 days. The color indicates the fraction of the archives contents, x-axis the day in the trace, and y-axis the modification time histogram bucket. For example, on day 275 of the trace, 80% of archive contents received their most recent modification 128–256 days ago. Note y-axis is log scaled (to match the histogram report), and we truncate it after 1024 days, as most contents are below that age.

NCAR over 15 years ago. That study showed that 65% of files referenced in the 24 month trace were only written to a single time, and over 20% were read but never written to. Similarly, at the end of our dataset's duration, despite only having a 13-month trace, we see that approximately 60% of corpus records had modification dates more than 256 days in the past. Similar proportions of modification times were shown in Dayal's study as well, though he did not track trends over time as we did [Dayal 2008].

Content Storage Mutability. Long-term content corpora are highly dynamic: 50% of records in the WATER corpus received 5 or more updates, often stemming from automatic data management processes. Similarly, 75% of the HISTORICAL corpus saw at least one update during the trace.

To compare tertiary storage update times with those of long-term content repositories, we examine data updates within the publicly accessible WATER corpus. One complication to note is that we can only deduce record creation in the WATER sketch by noting a record's first appearance in an update log. In our analysis, we associate each record with a list of updates generated from the update logs. As discussed in Section 3, we filtered the updates such that only updates mapped to files present on the last day were included in our analysis. Though this introduces the danger of under-counting updates, it ensures that our results remain conservative and removes potentially misleading update counts caused by record renaming.

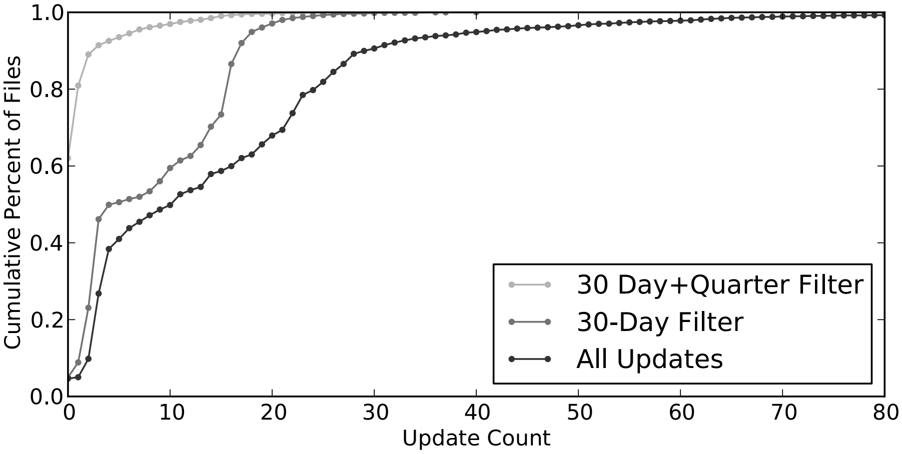


Fig. 3. CDF of records, showing the number of updates per record over the duration of the WATER sketch's update trace. The 30 day filter removes updates that occur more frequently than every 30 days, while the quarter filter removes quarterly overwrites of the data. These filters were implemented to approximate a lower bound for necessary updates as policies frequently needlessly overwrote records.

Examining the logs in the WATER sketch reveals a surprisingly high number of updates caused by corpus management: automatic policy rules frequently overwrote generated reports, whether or not they had actually received updated data. Two scripts in particular generated a large volume of data updates. The first ran approximately weekly, and modified any report that had data updated within 30 days. The second ran on an irregular, but roughly quarterly schedule, and overwrote all reports in the corpus regardless of the last update they received.

To identify the source of updates, we break our analysis into three sets. The first contains all the updates seen by the corpus. To isolate the results of the weekly script, the second set only considers updates that occur to a file after 30 days have passed. We call this the *30-day filter*. The last set takes the results of the 30-day filter, and removes all mass updates that touch over 10,000 records. We call this the *quarter filter*. Using this approach, we can identify a lower bound on the number of necessary updates; more may have been required to keep the relevant reports up to date, but no fewer.

The results, shown in Figure 3, demonstrate behavior that deviates dramatically from the "write-once" assumption of traditional tertiary storage. When no filters are applied, we see that only 40% of the records receive 5 or fewer updates, and those that receive 20 or fewer updates still only account for around 65% of all records. Applying the 30-day filter, we see that a significant fraction of the corpus still receives 5 or more updates. We observe a shift, however, when we filter out the quarterly updates, as 60% of the records receive zero updates. This is still far from the "write-once" scenario; 10% of the records receive 3 or more updates, and 20% receive one or more. Many of these are complete "Period of Record" reports that are running summaries of all prior data for a site.

The HISTORICAL sketch contained individual record update times, though it had no log the individual updates themselves. This meant that while we could identify *if* a record had been updated, we could not tell if it had received multiple updates. Despite this, we still found that over 75% of records received at least a single modification to either their metadata or associated object. Like the WATER corpus, this is in stark contrast to the oft-quoted "write-once" assumption.

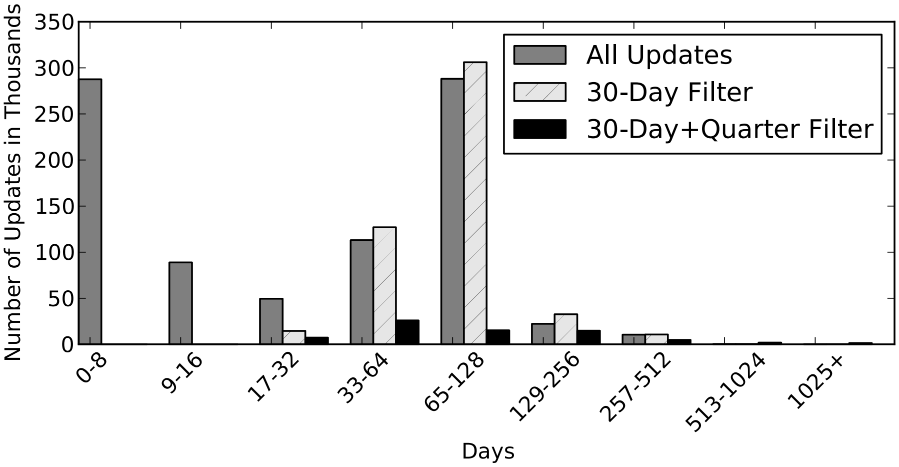


Fig. 4. Histogram of inter-update arrival times for all records in the WATER corpus.

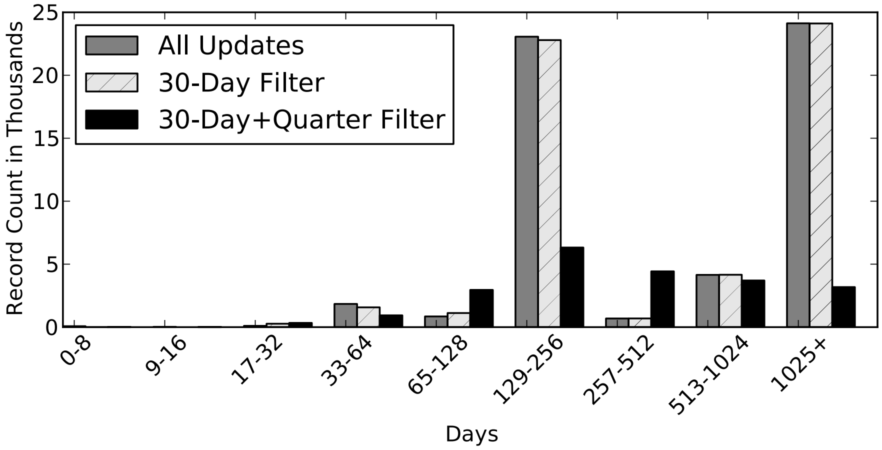


Fig. 5. Histogram of time between a record's creation, and its last update within the WATER corpus. Note records that receive no updates are not counted.

Content Storage Activity. The records in both the WATER and HISTORICAL corpora were both highly active long after their ingest times. In the WATER corpus, 50% of records received updates more than 256 days after their creation. In the HISTORICAL corpus 85% of modification times were more than 256 days past the record's creation date.

When we examine the inter-arrival time of updates, the time between any two consecutive updates to a record, illustrated in Figure 4, we see surprisingly large numbers of records with long inter-update periods. 35% of the approximately 900,000 observed updates occurred after a period of over 64 days. When we apply our 30-day and quarterly filters, we still see 70% and 50%, respectively, of updates occur with an inter-arrival time of more than 64 days, though the total volume of updates drops significantly.

To further investigate update behavior within the WATER corpus, we examine the range of time over which records were receiving updates. Figure 5 shows a histogram

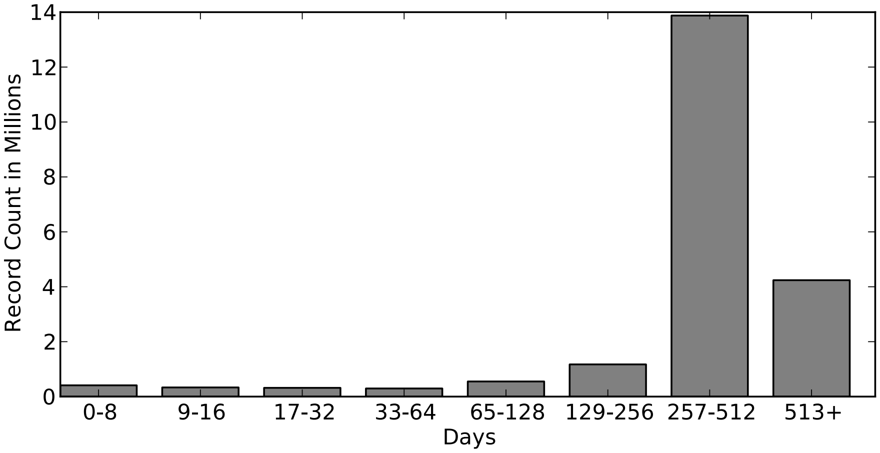


Fig. 6. Histogram for the HISTORICAL sketch, showing the range of time between a record's ingest date and its last modification time. Note that records not updated are not counted.

of the time between a record's creation and the last update it received. There are, however, two important points to note with this histogram. First, it does not include records that were never updated after creation as they would not contribute to the update count. Second, the record's ingest time relative to the start and end of the trace period impacts the update range we observe; for example, a record ingested 2 days before the end of trace would have at most a 2 day range. Nonetheless, this is still a valid method of demonstrating that records continue to be modified long after their ingest time. Using this approach, we see that over 50% of records that receive updates do so over a range of over 256 days. When we apply the 30 day and quarter filters we see the proportions remain roughly the same; approximately 50% of records that received updates were modified over 256 days after record creation. However, the *number* of records receiving updates drops due to the filtering.

In our HISTORICAL sketch we do not have the same level of access granularity we did in the WATER sketch; rather, we can only see a record's last modification time. This time reflects the most recent time that any of a record's fields or associated digital objects are modified. This level of detail is still sufficient to show that of the approximately 28 million publicly accessible records, over 21 million had a nonnull modification date, meaning that approximately 75% of the corpus content was updated at least once. This is significantly more than that shown in both Agrawal's desktop trace study [Agrawal et al. 2007], where over 80% of files remained unwritten each year for over 5 years; and the modern tertiary storage behavior illustrated in Figure 2, where approximately 80% of the corpus remained unmodified.

Update time ranges were also similar between the WATER and HISTORICAL sketches. When we look at the time between a HISTORICAL record's ingest and its last recorded modification time, shown in Figure 6, we see that 85% of the modification times show a difference of 256 or more days from the record's creation.

The surprising amount of update activity we see across both the WATER and HISTORICAL corpora is made possible—and easy—by the use of cheap random access media. The use of tape or optical media in the face of so many modifications would be problematic, as they require significant extra hardware to maintain high access rates. Additionally, the long access times of such media are a barrier to frequent modification of data. The relative ease of updating modern media may have subtle, but important, implications however. For example, repeated mass updates, like that seen in WATER corpus, can make identifying the source and nature of an update difficult. While relatively innocuous in the WATER corpus, repeated, potentially unnecessary, modifications of data can have profound implications in other situations, for example, legal rulings.

4.3. Accesses

In our logs for the WATER and HISTORICAL corpora, record modifications appear as session-less, system-generated operations. In contrast, record accesses are associated with a distinct session. Thus, our access analysis looks at both aggregate and session-oriented access behavior. As mentioned previously, the LANL archive disabled access time updates, and histogram reports were generated from metadata mirrors. Thus, we are unable to analyze user accesses in the SCIENTIFIC corpus.

Large-Scale Retrievals. Accesses are dominated by a few, often machine generated large-scale retrievals, such as a Google crawl or integrity checking process.

In the HISTORICAL sketch, we observe approximately 5.88 million distinct accesses between September 27, 2007 and June 16, 2010. The accesses are across 1.05 million user sessions, accessing 2.28 million distinct records. From discussions with the repository administrators, we also know that *all* records are integrity checked monthly. Though only 8% of the 28 million publicly available records were accessed by users over 3 years, 100% of the records were read via the integrity checking process each month. If we consider integrity checking to be equivalent to record retrieval, then less than 1% of reads come from end-users. Even assuming a less aggressive integrity checking schedule of once per year, only 10% of read traffic would come from users. This finding has significant implications on archive design. Effective, low-latency end-user retrievals are critical to the perception of a useful system, but only make up a small fraction of the actual workload. On the other hand, administrative processes, which make up the bulk of accesses and are critical to the integrity of the system, are typically less latency-sensitive. Thus, as we discuss further in this section as well as in Section 5, a separate batch interface for bulk accesses could provide significant benefit to future systems.

In the WATER sketch, we see roughly 98,000 distinct retrievals between August 28, 2007, and July 1, 2010. By artificially grouping accesses originating from the same IP address that arrive within 10 minutes of one another, we identify approximately 8500 user sessions. We choose 10 minutes as the threshold based on our observation that the number of sessions created by our grouping method taper off after approximately 10 minutes. We exclude approximately 1200 retrievals that had a null value for their files, accounting for approximately 1% of all retrieval requests.

We find that approximately 70,500 of the 98,000 total accesses in the WATER sketch originated from Google, and 27,000 from other users. Since there were 57,000 records in the last quarterly update, and non-Google users made 27,000 requests, we observe that no more than 50% of the archive's contents could have been retrieved by non-Google users. On the other hand, Google likely requested nearly all of the reports given the methodical nature of their crawls, though we cannot conclusively state this given the file renaming issues we noted earlier.

All told, most retrievals to both the WATER and HISTORICAL corpora came from automated sources. Interestingly, this has rough similarities to Vogels' 1999 study of file system usage in Windows NT . He found that the majority of file system accesses came from processes that took no user input [Vogels 1999]. He does not discuss the nature or purpose of the source processes, nor were the systems under study archival in nature. They were a mixture of personal, scientific, and administrative computers.

LRU Caching. LRU caching is moderately effective at absorbing per-session record re-retrievals and flash traffic. As a whole it is ineffective at absorbing day-to-day traffic due to limited record popularity.

One peculiar behavior we notice in both the WATER and HISTORICAL sketches is significant numbers of user-sessions re-retrieving the same record in the same session, often within a few seconds. Communication with system administrators and architects yielded no explanation for this odd behavior, though in the case of the HISTORICAL sketch we suspect the design web interface and retrieval system may be partially responsible, with activities such as a user clicking “back” in their browser causing a re-retrieval. We note that these re-retrievals accounted for 3% of the retrievals in the WATER corpora retrieval log, and nearly 35% (2.04 million) of the record retrievals for the HISTORICAL archive. These re-retrievals have a noticeable impact on our results and implications for archival system design.

From the daily access counts in the HISTORICAL sketch shown in Figure 7(a), we observe that the number of accesses on any given day is relatively stable, and exhibits a slow growth trend. We do, however, observe a number of moderate spikes, and one large spike around day 900.

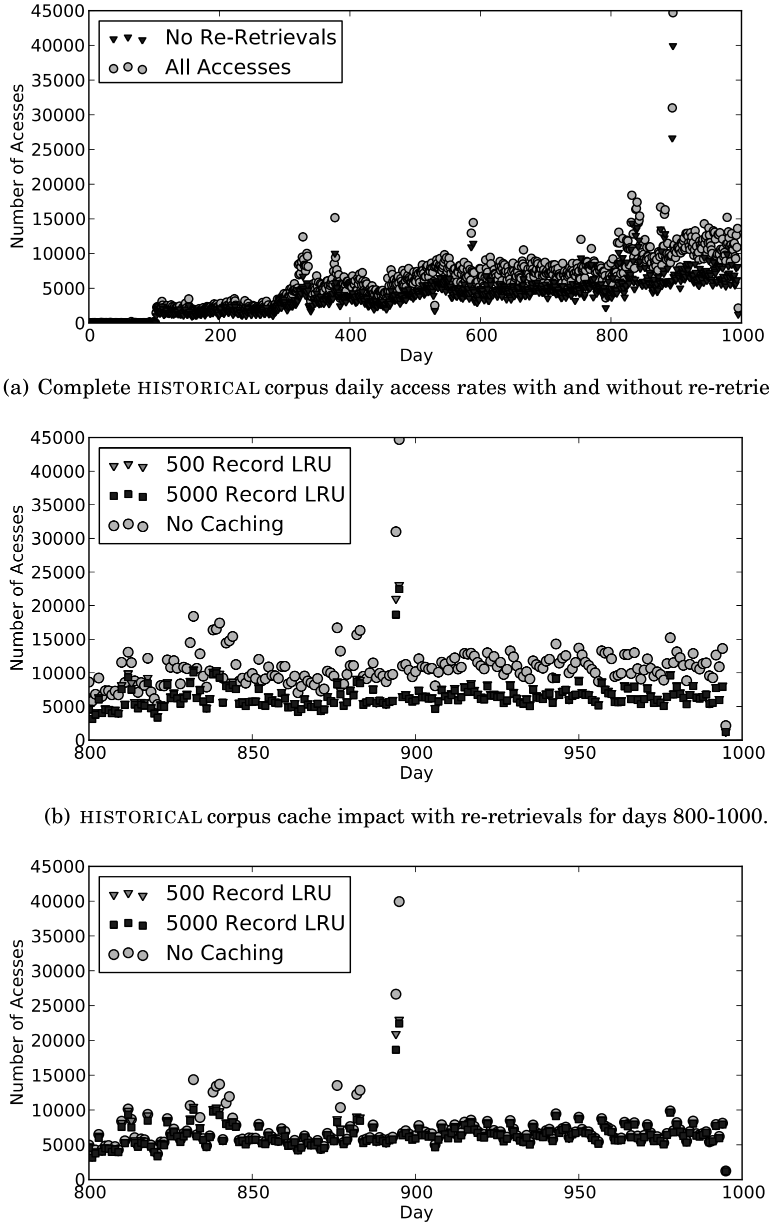
A microanalysis of the large spike finds that it is comprised almost entirely of sessions that only retrieved a single record, and that the records retrieved were predominantly (over 90%) photographs. Further, the upper quartile of distinct records retrieved in the spike received 5 or more accesses, as opposed to the usual 1 or 2 on most prior days we examined. Consultation with the system and corpus administrators yielded no clear explanation for behavior seen in the spike. Further, we confirmed that external indexers, such as Google, only have access to around 6000 records, ruling it out as a possible explanation.

To explore the potential effectiveness of caching on daily traffic and spike mitigation on the HISTORICAL corpus, we ran our daily access count analysis with two different sizes of a simple LRU caching filter: 0.01% and 0.1% of the total number of retrievals, corresponding to 500 and 5000 records. When we include re-retrievals during the same session in the count, even a small cache is shown to be moderately effective at absorbing accesses, with overall hit ratios of 37% and 38% for a 500 and 5000 record cache, respectively. When we remove the re-retrievals the cache effectiveness plummets, exhibiting an overall hit ratio of less than 7% for even the 5000 record LRU cache.

Interestingly, as Figure 7(c) shows, the cache is effective at reducing the magnitude of several of the access spikes. Even the small cache absorbed nearly 50% of the traffic during the day 900 spike. Thus, while their overall impact is low, read caches in long-term content stores may be useful for handling flash traffic and record re-retrievals.

Next, we examine daily access counts and cache effectiveness for the WATER corpus, illustrated in Figure 8. One of the first things we note is an extended access spike, approximately between days 700 and 750. Using a reverse IP look up we confirmed this was Google slowly crawling the repository contents. In total, Google accounted for over 70% of *all* record retrievals. For our subsequent analysis of the WATER corpus accesses, we filtered the large, external index crawl from the dataset. Note that while other user sessions did occasionally exhibit bot-like behavior (fast inter-retrieval times, and mass retrievals) we could not conclusively identify them as such, and left them in the trace.

In the WATER sketch, as with the HISTORICAL sketch, we see a moderate number of re-retrievals within user sessions, and examine the impact of caching with and without these re-retrievals, shown in Figure 9. As with our previous observations, with re-retrievals included, we see low to moderate cache effectiveness with hit rates of 12% for a cache size of 10 records, and 17% for 100 records. When we eliminate session



(a) Complete HISTORICAL corpus daily access rates with and without re-retrievals

(b) HISTORICAL corpus cache impact with re-retrievals for days 800-1000.

(c) HISTORICAL corpus cache impact without re-retrievals for days 800-1000.

Fig. 7. Daily access counts to the HISTORICAL corpus with and without re-retrievals and the associated LRU cache impacts. If a retrieval was absorbed by a cache hit it was not counted.

level re-retrievals the hit ratios drop to 2% and 8% respectively. In the WATER sketch, however, caching remains largely ineffective even on days with significantly increased traffic, as Figure 9(b) illustrates.

We believe there are two primary, and related, reasons for such poor cache performance. First, records show long inter-retrieval times on a system wide basis, as

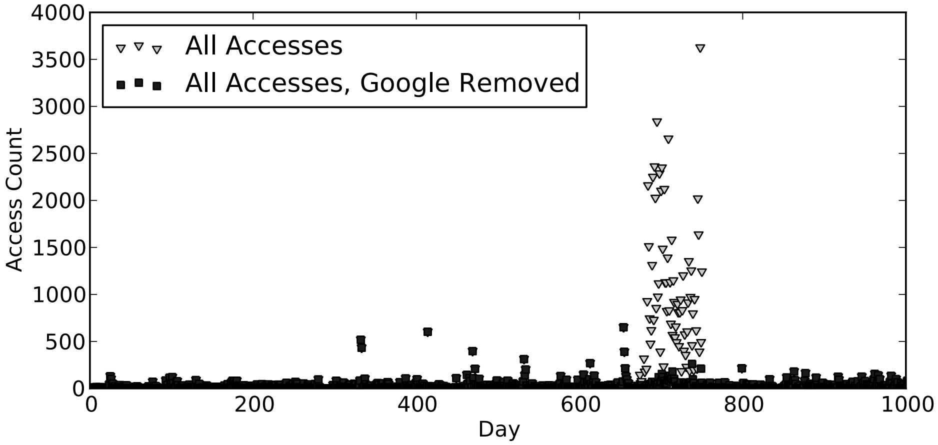


Fig. 8. Daily access counts to the WATER corpus with and without the retrievals by Google.

illustrated in Figure 10. Over 50% of retrievals, to records that were retrieved multiple times during the trace , occurred after an interval of more than 10 days. Second, as we discuss further later on in this section, there is little in the way of “hot” or popular content where an LRU cache would be more effective.

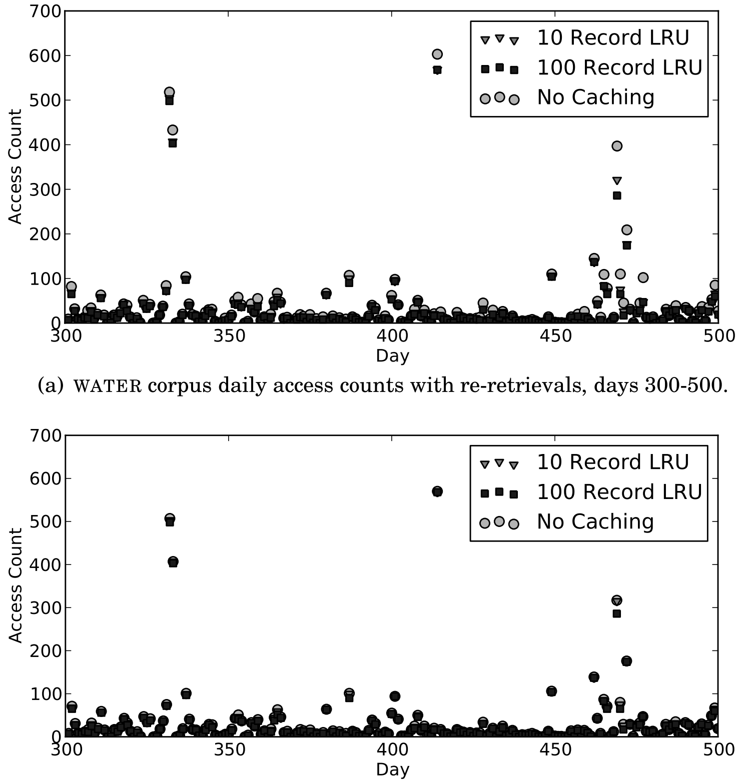
Per Session Behavior. 50% of users’ sessions only retrieve a single record, though this accounts for fewer than 10% of the total retrievals.

In Figure 11, we illustrate the number of retrievals per session with and without re-retrievals. Interestingly, for both the HISTORICAL and WATER sketches, we see that over 50% of sessions only retrieve a single record. Further we observe that the distribution quickly flattens out, with approximately 90% of sessions retrieving 15 or fewer records. Since many sessions are coming from humans interacting via a web interface, the time between user retrievals is relatively long, often seconds to minutes.

While 50% of sessions—with re-retrievals—only retrieve a single record, those sessions in the HISTORICAL trace account for fewer than 10% of the total retrievals, and fewer than 5% for the WATER sketch, as shown in Figure 12. The vast majority of data was accessed from larger sessions. In the HISTORICAL corpus, 40% of all accesses come from sessions of more than 20 retrievals, and nearly 80% in the WATER sketch are made during similarly large sessions. In the WATER sketch, this skew is due to a Google index crawl of the corpus that occurred over several large sessions, each retrieving hundreds to thousands of records. The prevalence of these large mass retrievals, much like the wholesale integrity checking, suggests the utility of a batch interface, as we discuss further in Section 5.

User-Based Retrieval Locality. Users’ sessions tend to show strong content correlation, retrieving a limited number of content types, for example, census records. Intersession (system-wide) content popularity is extremely long tailed.

Next, we look at content popularity, independent of sessions, to see if we can identify a subset of records or content types that account for a disproportionate fraction of accesses. Figure 13 shows that all of the distributions exhibit a long tail, with the exception of the types-based popularity for the HISTORICAL corpus. For example, the sites—water well location—in the WATER corpus exhibit the second strongest popularity affinity with 20% of sites accounting for 60% of accesses, but the next 20% of



(a) WATER corpus daily access counts with re-retrievals, days 300-500.

(b) WATER corpus daily access counts without re-retrievals, days 300-500.

Fig. 9. Daily access counts for the WATER corpus with and without re-retrievals, and associated LRU cache impacts of size 10 and 100 records. If a retrieval was absorbed by a cache hit it was not counted. Google accesses have been filtered out. Note cache impact is nearly eliminated when re-retrievals are removed.

sites only account for another 20% of accesses. At file level granularity, this trend becomes even more pronounced. A particularly interesting observation is that while the HISTORICAL corpus has more than two orders of magnitude more records than the WATER corpus, their per-record access CDFs are nearly identical. We note, however, that the file naming issues within the WATER corpus may mask some amount of file popularity. The content popularity distribution corroborates our early findings showing LRU read-caching to be largely ineffective; while certain categories of data are more popular, individual records do not appear to be particularly more popular.

We next examine access locality within sessions to see if individual user sessions tended to access a single or few types of content. In our analysis, we first remove re-retrievals from all sessions; we then exclude sessions in which only a single record is retrieved. Additionally we eliminate records from the HISTORICAL sketch that are found to be missing metadata (less than 0.5% of retrievals), as well as those with the category listed as “Restricted Type”; these records were originally public and subsequently moved into the private archive, so we cannot determine their category. The restricted type retrievals account for 12% of accesses after removing re-retrievals and excluding singleton sessions.

Figure 14 shows that, across both the HISTORICAL and WATER traces, individual user sessions tend to retrieve strongly related content. We see that nearly 50% of

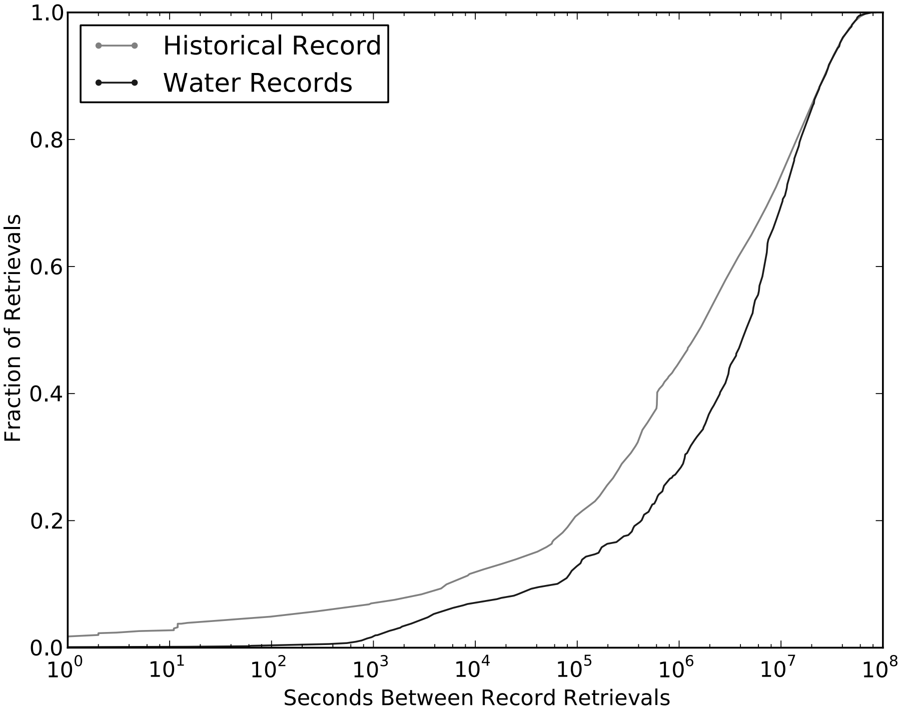


Fig. 10. CDF of inter-retrieval times for records in both the HISTORICAL and WATER corpora. Re-retrievals within a single user session were filtered, as were records that were only retrieved a single time.

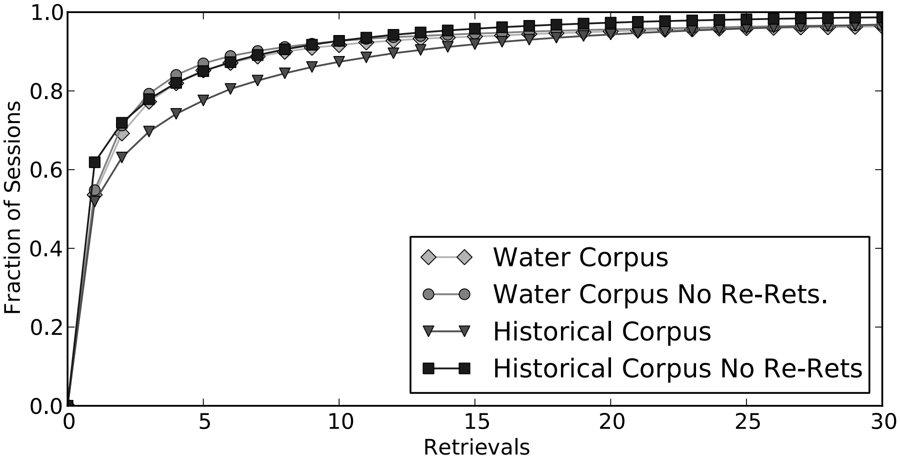


Fig. 11. CDF of accesses per session showing the number of records retrieved per session in the WATER and HISTORICAL corpora, with and without per session re-retrievals. We truncate at 30 accesses, the few large sessions would distort the plot.

sessions in the HISTORICAL trace retrieve three or fewer record types, and similarly 50% of sessions in the WATER trace retrieve data pertaining to only a single site. When we include the year, 25% of sessions retrieve records within a single site-year combination, but still exhibit strong per-session content locality.

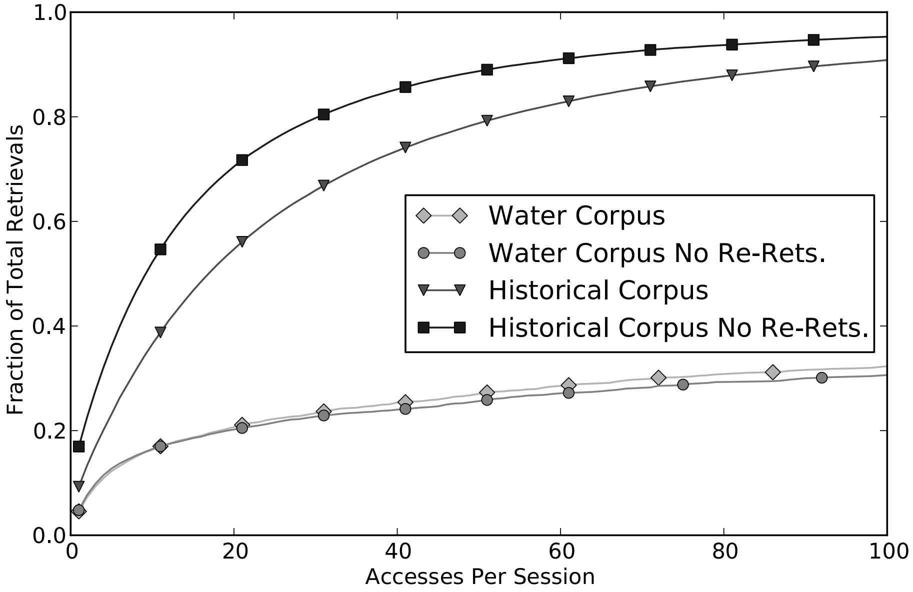


Fig. 12. CDF showing what fraction of total retrievals were contributed per session size with and without re-retrievals for the WATER and HISTORICAL corpora. Note we truncate the x-axis at 100 to maintain readability as retrievals to the WATER corpus dominated by a few large sessions; sessions with over 300 retrievals account for 60% of the total retrievals.

Across sessions, however, a wide variety of content and individual records are retrieved. This is evident in the poor cache performance, as shown earlier in Figures 7 and 9. The strong individual session locality does suggest that grouping data based on content along with prefetching may be effective [Wildani and Miller 2010], provided the content type has a sufficiently small number of records. For example, this would likely be more effective for the WATER corpus where any given site-year combination rarely has more than 15 or so reports at a few tens of kilobytes apiece, than for the HISTORICAL corpus where there may be many millions of records connected by an associated category. In contrast the lack of individually popular records we noted earlier impacts systems that aim to conserve energy by duplicating or migrating commonly used data, as they require relatively fine grained, predictable system-wide accesses in order to be effective [Colarelli and Grunwald 2002; Pinheiro and Bianchini 2004]. This is because while we can make strong statements about individual session behavior, aggregate system wide activity is largely unpredictable in regards to the popularity of records.

5. LESSONS LEARNED

Our investigation into the behavior of tertiary storage and long-term public content corpora revealed a number of high-level lessons. In this section, we interpret our observations, and discuss their implication for long-term storage system design and future research directions.

5.1. Read-Write Behavior

While the SCIENTIFIC corpus exhibited the classical tertiary storage behavior of fairly static content, the WATER and HISTORICAL corpora showed a surprisingly high degree of change; data modifications were frequent, widespread, and over a much longer duration than we expected. However, even in the SCIENTIFIC corpus we observed a

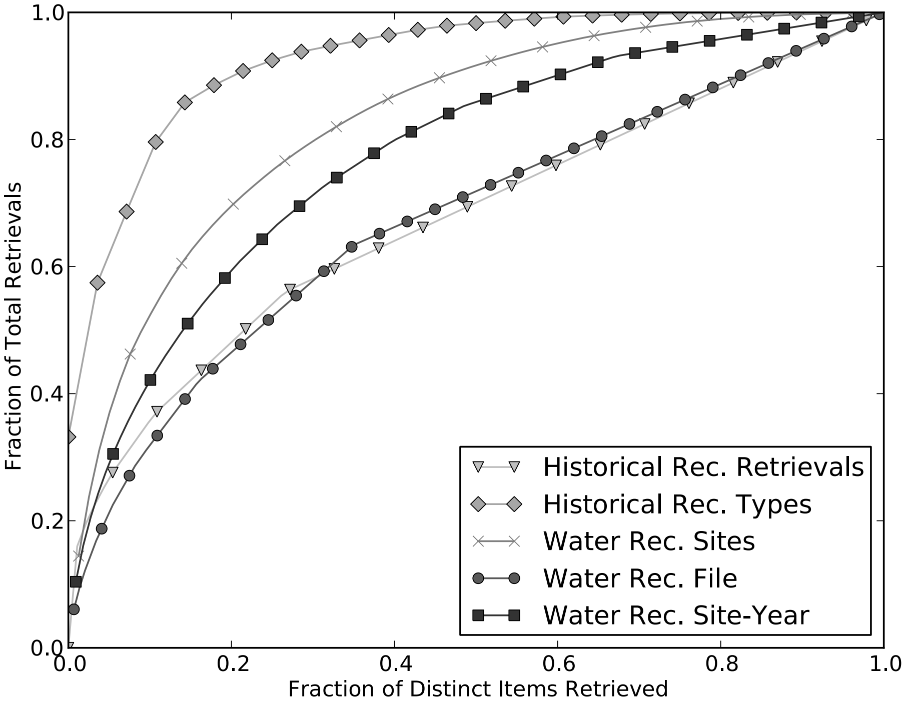


Fig. 13. CDF of record popularity by individual record, and distinct content type. Note that x-axis values are ordered by popularity, that is, the most popular items are plotted first. With the exception of the HISTORICAL corpus record types all CDFs are subsampled.

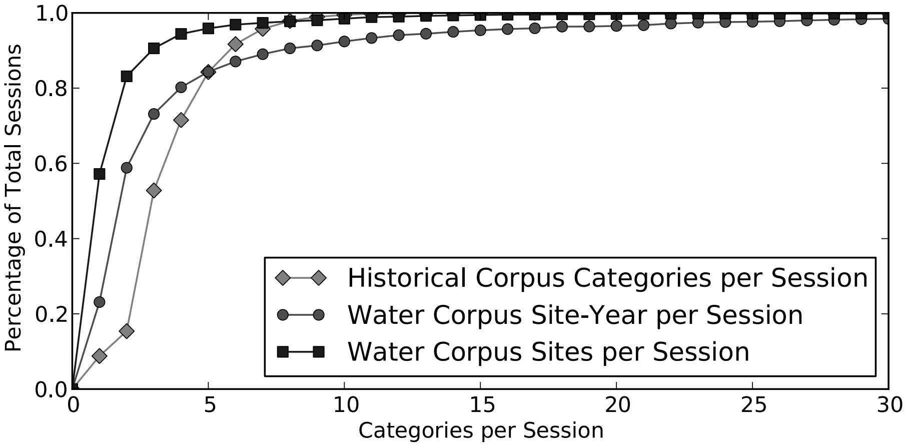


Fig. 14. CDF showing the number of different content categories retrieved per session. In the HISTORICAL corpus, records have an associated category. In the WATER corpus, we examine how many distinct sites as well as site year combinations are accessed per user session. We truncate at 30 types to avoid a distorted plot.

storage medium shift from tape to disk, suggesting that file immutability should be an enforceable policy independent of the media type. This flexibility is even handy for explicitly immutable data, such as compliance stores, as such datasets often have a specific expiration date, after which the owners would like the data to be immediately deleted.

With respect to reads, contrary to assumptions established by tertiary storage, we found that both the WATER and HISTORICAL corpora were quite active. While user requested reads were relatively rare, data management tasks, indexing requests, and the inevitable migration of long-term data, make the "read-maybe" pattern patently false; all content is eventually read, and it is often read en masse. With our additional observation of largely unpredictable user reads, this could severely impact the effectiveness of system designs that rely on low read-rates. For example, systems that rely on spun down disks for power savings may overestimate the cost savings they can deliver [Pinheiro and Bianchini 2004; Storer et al. 2008], unless accesses can be tightly controlled and scheduled.

Further, our results suggest a potential danger in optimizing for the wrong operations. In the WATER trace, the vast majority of total accesses were from a few large-scale requests, such as Google crawls, with the remainder originating from user accesses that often only retrieve a single record. We argue, however, that these small numbers of user requests are latency sensitive, and critical to the users' perception of effective, long-term storage.

Within these critical user sessions, we found that there were a few favored content types, and the same data was often requested multiple times within a single session. Across sessions, however, it was much more difficult to identify popular content. Thus, aside from assisting with the re-retrieval problem, strategies that rely on migrating popular data may be ineffective at best, and harmful at worst [Pinheiro and Bianchini 2004; Zhu et al. 2005]; in a disk spin-down scenario, such movement could incur additional energy penalties for little or no benefit. Depending on the scale, this may also make the use of tape-based architectures, and those based upon immutable media types significantly less efficient.

It is important, however, not to downplay the importance of bulk accesses, since they dominate an archive's workload. Recall that we observed integrity checking accounting for 99% of read accesses in our analysis of the HISTORICAL sketch. The disparity in large and small access properties suggests that current archive interfaces are insufficient. Since our data suggests that these large-scale accesses are often latency-insensitive administrative processes, we propose an asynchronous batch interface for large requests as a complement to the traditional single record interface. The benefit to the system is that such a request would provide full a priori knowledge of the records in the requests, allowing the archive to optimize its resource scheduling most effectively.

Such an interface would allow a client to specify the set of records desired, a schedule of when it needs the requests fulfilled by, and a means to alert the client when the request is complete. For writing, this could be useful for data management functions: we observed that public content archives provide anonymous read access, but writes came from the system itself. In the case of external indexing services, such an interface could help shift the large-scale requests from appearing parasitic at an energy cost and workload spike standpoint, to a more symbiotic relationship in which the indexing service receives the data, and the archive can efficiently provide the means for users to find it. To prevent pathological use of the traditional, single-record access interface, archives could utilize strategies such as throttling or retrieval caps.

5.2. Understanding Corpus Behavior

While the results presented in this article have helped to expand our understanding of long-term corpus behavior, a number of trends motivate the need to understand the aggregate behavior of multiple corpora hosted by a single archive. First, the growth of cloud storage services marks a shift towards centralized data centers. Second, increasingly digital workflows have spurred the proliferation of small and mid-sized corpora. Combined, this leads to potential problems in optimization, as superficially similar corpora may be naively hosted on the same physical hardware in a cloud or other multi-corpus environment, despite the fact that they may benefit from different configurations. For example, while both the HISTORICAL and WATER corpora showed strong content locality within user sessions, their record granularity is vastly different; a single record type in the WATER corpus may only contain 20 or 30 records, while in the HISTORICAL corpus one record type may have millions of records. An optimal retrieval technique for one may be pathological for the other. This suggests two future work directions. First, we need further workload studies to continue identifying similar and divergent workload behavior in archives. Second, as cloud storage is becoming more pervasive, we need to do a critical examination from both a user and provider side into how to effectively handle archival workloads.

Even within the scope of a single corpus there is more work to be done with our current sketches. First, we would like to investigate both temporal and content locality in record ingests and updates. Second, we plan to more closely examine short-term behavior within the traces, to see if short-term behavior in accesses and updates is similar to that of the long-term behavior we observed. Finally, as an aid in the development of effective long-term repository systems, we hope to publish a series of workloads based on these traces. This would help in the evaluation of archival designs, and in the reproducibility of published results [Agrawal et al. 2009].

Finally, as we have noted, there is a wide gamut of systems that now fall within the category of long-term data repositories, ranging from personal storage arrays to massive centralized systems such as the Internet Archive [Jaffe and Kirkpatrick 2009]. To that end we hope to locate and examine additional traces across all areas of archival storage in order to identify and quantify their common and divergent characteristics. For example, even within the tertiary storage area, more granularity and study is needed; the sketch we obtained from LANL had disabled recording of access times, impeding our ability to analyze read behavior.

5.3. Tracing Difficulties

Acquiring high-quality trace data for this study proved to be a vexing challenge. The worst examples we encountered were logs with no field descriptions or supporting documentation, making them effectively unusable. However, with immeasurable assistance from the archive owners, we were able to obtain and refine several relevant and useful traces. To this end, we see a strong need to continue the development of tools that enable organizations to easily collect and efficiently store descriptive and relevant long-term access data [Anderson et al. 2009]. This is particularly true for archival storage, since, in contrast to storage systems such as those used in enterprise and the desktop, traces must be gathered over years. If data are not gathered properly, reacquiring a trace can take years, perhaps preventing the trace from driving an analysis to guide the design of the archival storage system's successor. In addition, there is a need for consistent tracing standards to ensure ease of use and readability far into the future.

Good tracing tools would also be a boon to archive operation, as our observations revealed counter-productive behavior of which the system administrators and architects were unaware. For example, in the HISTORICAL sketch, our analysis revealed frequent record re-retrievals within the same session; the administrators did not know of this behavior, and were unable to explain it. Additionally, the WATER corpus exhibited many needless overwrites coming from data management processes. These irregularities highlight the need for good analysis tools to help administrators identify pathological behavior within the system.

To this end, we see a need for data-centric (corpus-centric) tools for long-term tracing for several reasons. First, seemingly trivial actions can make analysis of longer-term trends extremely difficult. For example, in the WATER trace, files were identified by path names, but file renames did not capture the information needed to link old path names to new path names. Second, as systems become increasingly distributed, there may be multiple instances of the same corpus in a single system, motivating the need for tracing tools that can provide a holistic view across archives [Thereska et al. 2006]. Third, given the intended long lifetime of many corpora, data will live on many systems over its life. In order to understand how data behavior evolves, a long-term trace must extend beyond the lifetime of any single system.

Even with useful tools and traces, the importance of good communication with system architects and administrators cannot be overstated. They can provide information that is not captured by trace reports, significantly altering conclusions. For example, in the HISTORICAL sketch, communication with the administrators was instrumental in understanding the scale of user request traffic and system-generated integrity checking traffic, and in the WATER sketch the administrators explained the nature of their periodic batch processes.

6. CONCLUSIONS

As ever-growing quantities of our society's data are stored in long-term digital archives, it is increasingly important to understand how these archival storage systems are used, and how they behave. To address this question, we presented a detailed analysis of behavior in three archival storage systems, including both scientific and public data. Our study provides the first examination of a large tertiary storage system in over 15 years, and the first ever analysis of the behavior of public content archives. Based on our findings, we have made concrete suggestions for both archival storage system implementers and administrators.

By analyzing the LANL SCIENTIFIC sketch, we were able to see how tertiary storage archives have evolved in the last seventeen years. Our analysis reveals that, compared to the NCAR system studied in 1993, the LANL SCIENTIFIC corpus exhibits a CAGR (compound annual growth rate) of 25.8%. Further, hard drives play an increasingly important role in the archive; the NCAR system had a disk to tape ratio of 1:262, in sharp contrast to the LANL archive's ratio of 1:3.3. Despite this shift, the update patterns between then and now are largely unchanged. Additional access time information is needed to fully understand how the random access performance of hard drives is being utilized and how larger disk caches have impacted file migration patterns.

The public WATER and HISTORICAL sketches demonstrated how long-term storage now covers a wide range of behavior. We found that the contents were both accessed and modified frequently; 75% of the historical corpus saw at least one update over the trace period, and 50% of the water corpus saw 5 or more writes. Access traffic was dominated by a few large-scale requests such as data management scripts and Google crawls. This behavior, along with the latency sensitivity of small requests suggest that two different interfaces are called for: one for the small, but critically important, user requests; and another for the large-scale, but latency-insensitive bulk requests. The smaller requests demonstrated strong per-session access locality, but little inter-session popularity. This has a two-fold implication. First, physically grouping semantically related content may yield efficiency and performance gains. Second, the limited popularity and the unpredictable (random) access patterns of content across the system may curtail the effectiveness of spin-down and popular data concentration techniques on power-efficiency and performance. In total, the results of our workload study and the guidelines developed from the results will help archival system designers in the construction and maintenance of archives that can efficiently and effectively preserve society's digital legacy for future generations.

ACKNOWLEDGMENTS

Thanks to Gary Grider of LANL, Greg Smith of the California Department of Water Resources, and June Timmons and Brian Smith of the Washington State Digital Archives for all of their help and support.

REFERENCES

AGRAWAL, N., BOLOSKY, W. J., DOUCEUR, J. R., AND LORCH, J. R. 2007. A five-year study of file-system metadata. In *Proceedings of the 5th USENIX Conference on File and Storage Technologies (FAST)*. 31–45.

AGRAWAL, N., ARPACI-DUSSEAU, A. C., AND ARPACI-DUSSEAU, R. H. 2009. Generating realistic *impressions* for file-system benchmarking. In *Proceedings of the 7th USENIX Conference on File and Storage Technologies (FAST)*. 125–138.

ALASKA STATE. 2010. Alaska's digital archives. `vilda.alaska.edu`.

AMAZON. 2011. Amazon's simple storage service. `http://aws.amazon.com/s3/`.

ANDERSON, E. 2009. Capture, conversion, and analysis of an intense NFS workload. In *Proceedings of the 7th USENIX Conference on File and Storage Technologies*.

ANDERSON, E., ARLITT, M., CHARLES B. MORREY, I., AND VEITCH, A. 2009. DataSeries: An efficient, flexible data format for structured serial data. *ACM SIGOPS Operat. Syst. Rev. 43,* 1, 70–75.

BAIRAVASUNDARAM, L. N., GOODSON, G. R., PASUPATHY, S., AND SCHINDLER, J. 2007. An analysis of latent sector errors in disk drives. In *Proceedings of the SIGMETRICS Conference on Measurement and Modeling of Computer Systems*.

BAIRAVASUNDARAM, L. N., GOODSON, G. R., SCHROEDER, B., ARPACI-DUSSEAU, A. C., AND ARPACI-DUSSEAU, R. H. 2008. An analysis of data corruption in the storage stack. In *Proceedings of the 6th USENIX Conference on File and Storage Technologies (FAST)*. 223–238.

BAKER, M., KEETON, K., AND MARTIN, S. 2005. Why traditional systems don't help us save stuff forever. In *Proceedings of 1st IEEE Workshop on Hot Topics in System Dependendability*.

BAKER, M., SHAH, M., ROSENTHAL, D. S. H., ROUSSOPOULOS, M., MANIATIS, P., GIULI, T., AND BUNGALE, P. 2006. A fresh look at the reliability of long-term digital storage. In *Proceedings of EuroSys'06*. 221–234.

BENT, J., GIBSON, G., GRIDER, G., MCCLELLAND, B., NOWOCZYNSKI, P., NUNEZ, J., POLTE, M., AND WINGATE, M. 2009. PLFS: A checkpoint filesystem for parallel applications. In *Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*.

CALIFORNIA DWR. 2010. California Department of Water Resources water reports. `http://www.water.ca.gov/waterdatalibrary/docs/Hydstra/index.cfm`.

CHRONICLES. 2011. Chronicles of life: Save your memories forever. `http://www.chronicleoflife.com/`.

COLARELLI, D. AND GRUNWALD, D. 2002. Massive arrays of idle disks for storage archives. In *Proceedings of the ACM/IEEE Conference on Supercomputing (SC'02)*.

CORNELL UNIVERSITY LIBRARY. 2010. Cornell University Library arXiv. `http://arxiv.org/`.

DAMOULAKIS, J. 2007. Opinion: Tape backup is WORN (write once, read never). `http://www.computerworld.com/s/article/9026619/Opinion_Tape_backup_is_WORN_write_once_read_never_`.

DAYAL, S. 2008. Characterizing HEC Storage Systems at Rest. Tech. rep. CMU-PDL-08-109, Carnegie Mellon University.

DROPBOX. 2011. Dropbox. `http://www.dropbox.com/`.

GIBSON, T., MILLER, E. L., AND LONG, D. D. E. 1998. Long-term file activity and inter-reference patterns. In *Proceedings of the 24th International Conference for the Resource Management and Performance and Performance Evaluation of Enterprise Computing Systems (CMG'98)*. CMG, Anaheim, CA, 976–987.

GIBSON, T. J. AND MILLER, E. L. 1998. Long-term file activity patterns in a UNIX workstation environment. In *Proceedings of the 6th Goddard Conference on Mass Storage Systems and Technologies/15th IEEE Symposium on Mass Storage Systems*. 355–372.

HIPAA. 1996. Health Information Portability and Accountability Act.

IBM. 2010. IBM 3380 direct access storage device. http://www-03.ibm.com/ibm/history/exhibits/storage/storage_3380e.html.

JAFFE, E. AND KIRKPATRICK, S. 2009. Architecture of the Internet archive. In *Proceedings of the Israeli Experimental Systems Conference (SYSTOR'09)*.

JENSEN, D. W. AND REED, D. A. 1993. File archive activity in a supercomputing environment. In *Proceedings of the 7th International Conference on Supercomputing (SuperComputing'93)*. 387–396.

LEUNG, A. W., PASUPATHY, S., GOODSON, G., AND MILLER, E. L. 2008. Measurement and analysis of large-scale network file system workloads. In *Proceedings of the USENIX Annual Technical Conference*.

LILLIBRIDGE, M., ELNIKETY, S., BIRRELL, A., BURROWS, M., AND ISARD, M. 2003. A cooperative Internet backup scheme. In *Proceedings of the USENIX Annual Technical Conference*. 29–42.

MANIATIS, P., ROUSSOPOULOS, M., GIULI, T. J., ROSENTHAL, D. S. H., AND BAKER, M. 2005. The LOCKSS peer-to-peer digital preservation system. *ACM Trans. Comput. Syst. 23,* 1, 2–50.

MILLER, E. AND KATZ, R. 1993. An analysis of file migration in a Unix supercomputing environment. In *Proceedings of the Winter USENIX Technical Conference*. 421–433.

MOORE, R. L., D'AOUST, J., MCDONALD, R. H., AND MINOR, D. 2007. Disk and tape storage cost models. In *Archiving 2007*.

NEW YORK STATE. 2010. New York State digital archives. http://www.archives.nysed.gov/aindex.shtml.

NOAA. 2010. National Climatic Data Center. http://www.ncdc.noaa.gov/oa/ncdc.html.

ORNL. 2010. Distributed Active Archive Center. http://daac.ornl.gov/.

PINHEIRO, E. AND BIANCHINI, R. 2004. Energy conservation techniques for disk array-based servers. In *Proceedings of the 18th International Conference on Supercomputing*.

PINHEIRO, E., WEBER, W.-D., AND BARROSO, L. A. 2007. Failure trends in a large disk drive population. In *Proceedings of the 5th USENIX Conference on File and Storage Technologies (FAST)*.

QUINLAN, S. AND DORWARD, S. 2002. Venti: A new approach to archival storage. In *Proceedings of the Conference on File and Storage Technologies (FAST)*. USENIX, 89–101.

ROSELLI, D., LORCH, J., AND ANDERSON, T. 2000. A comparison of file system workloads. In *Proceedings of the USENIX Annual Technical Conference*. USENIX Association, 41–54.

SARBANES-OXLEY. 2002. Sarbanes-Oxley act 2002. www.soxlaw.com.

SCHROEDER, B. AND GIBSON, G. A. 2007. Disk failures in the real world: What does an MTTF of 1,000,000 hours mean to you? In *Proceedings of the 5th USENIX Conference on File and Storage Technologies (FAST)*. 1–16.

SMITH, A. J. 1981a. Analysis of long term file reference patterns for application to file migration algorithms. *IEEE Trans. Softw. Engin. 7,* 4, 403–417.

SMITH, A. J. 1981b. Long term file migration: Development and evaluation of algorithms. *Comm. ACM 24,* 8, 521–532.

STORER, M. W., GREENAN, K. M., MILLER, E. L., AND VORUGANTI, K. 2007. POTSHARDS: Secure long-term storage without encryption. In *Proceedings of the USENIX Annual Technical Conference*. 143–156.

STORER, M. W., GREENAN, K. M., MILLER, E. L., AND VORUGANTI, K. 2008. Pergamum: Replacing tape with energy efficient, reliable, disk-based archival storage. In *Proceedings of the 6th USENIX Conference on File and Storage Technologies (FAST)*.

STRANGE, S. 1992. Analysis of long-term UNIX file access patterns for application to automatic file migration strategies. Tech. rep. UCB/CSD 92/700, University of California, Berkeley.

THERESKA, E., SALMON, B., STRUNK, J., WACHS, M., ABD-EL-MALEK, M., LOPEZ, J., AND GRANGER, G. R. 2006. Stardust: Tracking activity in a distributed storage system. In *Proceedings of the SIGMETRICS Conference on Measurement and Modeling of Computer Systems*.

TRAEGER, A., ZADOK, E., JOUKOV, N., AND WRIGHT, C. P. 2008. A nine year study of file system and storage benchmarking. *ACM Trans. Storage 4,* 2.

VOGELS, W. 1999. File system usage in Windows NT 4.0. In *Proceedings of the 17th ACM Symposium on Operating Systems Principles (SOSP'99)*. 93–109.

WASHINGTON STATE. 2010. Washington State digital archives. http://www.digitalarchives.wa.gov/.

WILDANI, A. AND MILLER, E. L. 2010. Semantic data placement for power management in archival storage. In *Proceedings of the 5th International Workshop on Petascale Data Storage (PDSW10)* (held in conjunction with *SC2010*).

WILDANI, A., SCHWARZ, T., MILLER, E. L., AND LONG, D. D. E. 2009. Protecting against rare event failures in archival systems. In *Proceedings of the 17th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*.

YOU, L. L., POLLACK, K. T., AND LONG, D. D. E. 2005. Deep store: An archival storage system architecture. In *Proceedings of the 21st International Conference on Data Engineering (ICDE'05)*.

ZHANG, Z., LIAN, Q., LIN, S., CHEN, W., CHEN, Y., AND JIN, C. 2007. BitVault: A highly reliable distributed data retention platform. *ACM SIGOPS Operat. Syst. Rev. 41,* 2, 27–36.

ZHU, B., LI, K., AND PATTERSON, H. 2008. Avoiding the disk bottleneck in the Data Domain deduplication file system. In *Proceedings of the 6th USENIX Conference on File and Storage Technologies (FAST)*.

ZHU, Q., CHEN, Z., TAN, L., ZHOU, Y., KEETON, K., AND WILKES, J. 2005. Hibernator: Helping disk arrays sleep through the winter. In *Proceedings of the 20th ACM Symposium on Operating Systems Principles (SOSP'05)*. ACM.

Received May 2011; revised August 2011; accepted October 2011