storfirstEAS

seven10 STORAGE SOFTWARE

MANAGE DATA SMARTER™

StorFirst Enterprise Archival Storage
Whitepaper: Active Archiving Through Tiered Storage

Introduction to Enterprise Active Archives

Since the advent of low cost/high performance communication infrastructure, many of today's enterprises are required to have immediate access to old information. An example of this expectation of "infinite institutional memory" includes medicine, where new digital imaging modalities create gigabytes of clinical data per patient. All data is expected to be available to clinicians with minimal delay, regardless of the clinician's location. In the financial sector, billions of transactions and email are retained, based on policies closely tied to litigation and regulatory requirements. We have now begun to see the entertainment industry digitizing its assets to provide on-demand access to increasingly smaller and specialized audiences.

This new class of data, fixed content, is accessed infrequently and commonly referred to as archive data. Archive data presents some unique challenges to today's common infrastructure. The requirement that data be instantly available precludes the application of traditional backup and "shelf management" approaches to old information. In fact, most IT professionals realize that archived tapes represent a "Write Once Read *Never*" approach to data management which no longer satisfies users' requirements for data availability. Capacity and infrequency of access also challenge traditional SAN fabrics which have been designed to support online transactional processes. Plus, most current file systems lack support for multi-petabyte archives and embedded retention policies required by active archives.

While efforts have been made to address hardware problems in long-term storage by implementing low cost alternatives, little has been offered by traditional file systems to meet the functional requirements of the problem. Users are quickly discovering that traditional on-disk file systems such as NTFS have been developed for general purpose computing and lack the scalability, data security, and performance to cope with the infinite influx of fixed content. At Seven10, we applied our collective experience with file systems and archive data to develop StorFirst EAS™ (Enterprise Archival Storage). This paper describes the challenges we perceived during the design process, how our software design met those challenges, and the ease with which the software can be implemented.

Characteristics of Active Archives

Archive data and its source applications have the following characteristics:

Immutability – Reference data typically does not change, and is not usually modified or edited by users.

Example: Radiologists cannot alter a patient's x-ray images.

Example: Traders cannot alter transactional data.

Example: Corporate email and documents are discoverable assets and cannot be modified.

Policy driven long-term retention – Data needs to be retained for extended periods, perhaps even indefinitely.

Example: Under HIPAA guidelines, healthcare records are maintained for the life of the patient and beyond.

Example: Unenforced corporate document retention policies make all documents discoverable.

Infinite Scale/Infrequent access – While active archives are often enormous in both the number of files and total data under management, users tend to only need access to a small portion of the data at any given time.

Example: Medical images infrequently need to be loaded for viewing.

Example: Email and documents for litigation support are rarely accessed.

Example: Historical transactional data for analyst or fund manager access is only accessed periodically.

High throughput and continuous streams – Streaming access to bulk data is required and read/write access is mostly sequential, rather than random.

Example: Multimedia recordings of a doctor's notes, such as audio or video files, need to be available for streaming.

Example: Surveillance video and remote sensing data are all stream intensive.

Event-driven – An event initiates the need for data to be accessed.

Example: For a patient's annual health exam, all records can be archived for storage.

Example: A criminal investigation requires selective recall of surveillance images.

Business vital – Not mission-critical in terms of needing instantaneous access, but data must be accessible in seconds or minutes.

Example: Historical patient records do not have to be immediately accessible in the event of a disaster, unlike mission-critical transactional data, which may include patients currently being treated when disaster strikes.

Traditional file systems have focused on high performance with large numbers of small transactions. They also rely on access control and permission mechanisms to prevent data modification and have no intrinsic support for immutable file types.

StorFirst EAS Design Philosophy

The driving philosophy behind StorFirst EAS is simplicity. We actively avoided features which add complexity and are only peripherally associated with the problem. The following core principles have been strictly followed, ensuring the highest level of scale, security, and flexibility possible:

No Client Software Required – While active archives can be managed by relatively few servers, it may be accessed by hundreds or thousands (perhaps millions) of users in a given application area. Installation of file system extension software on client servers is an unacceptable and unnecessary hurdle for end users.

No Integration/Validation/Certification Required– Early attempts solving the active archive management issue have required the use of proprietary platforms to store and retrieve data from these "all-in-one" storage appliances. StorFirst EAS eliminates vendor lock-in by unifying all types of storage subsystems under a single management layer. Ideally, access to enterprise archives should be transparent across multiple tiers of storage while supporting platform independence.

Ease of Scale – We have preserved the underlying file system and have tried to virtualize the assets under management to achieve scalability. While storage virtualization has many important applications, it adds unnecessary complexity to merely achieve some sense of data unification, ignoring the importance of scale.

Our approach has been to provide the user with a familiar environment while making the details of the underlying implementation transparent whenever possible. For example, StorFirst EAS provides a standard CIFS interface and relegates device management to a simple explorer like data administration UI.

StorFirst EAS Implementation

StorFirst EAS is an installable file system designed for Microsoft Windows. Several features distinguish it from other file systems on Windows servers and provide unique facilities for persistent data:

Dynamic Device and LUN Aggregation – StorFirst EAS treats hardware, storage devices, and virtual LUNs as the atomic unit of aggregation. Any number of storage targets can be combined into a single virtual file system (VFS). Storage subsystems or devices are dynamic and can be added, deleted or moved offline using the administrator UI. This device management allows removable and fixed media to be mixed within the same VFS. This feature allows StorFirst EAS to mix FC disk (i.e. EMC CLARiiON), EMC Centera, VTL, tape, and future storage subsystems (such as Atmos cloud storage) into a single logical volume.

N-Tier Virtualization of Heterogeneous On-Disk Formats – StorFirst EAS optimizes the on-disk formats for the media in question. For example, EMC's unique C-Clip format is used for Centera while a proprietary WORM format on disk is used for FC Disk, MAID, or DAS. This is provided so that all volumes can be self-describing and be moved between StorFirst EAS installations. Most other efforts at virtualization have introduced assembly specific limitations which prevent devices from being moved to other server environments. For example, this fragility greatly limits the usefulness of NTFS dynamic disks. Support for N-Tier data replication allows StorFirst EAS to retain data on access-appropriate technology. The integrated retention engine (see below) uses policies to preserve data on the lowest cost media possible.

High Performance External Metadata Services – StorFirst EAS maintains all necessary metadata to make the storage environment self-describing. The same metadata is also externalized by the StorFirst EAS metadata service. This has significant implications for performance. Traditional file systems require that metadata on the volume be scanned for queries and searches, imposing significant performance challenges in very large or power-managed storage systems.

Searching such systems with a traditional file system can result in poor performance (imagine a full meta-data search on a 5 PB volume) or pathological behavior (disk thrashing). The highly indexed metadata service provides very response query operation with less than 0.1% storage overhead.

True WORM – StorFirst EAS was designed as a true WORM file system. Users can configure the software to ensure that files cannot be altered or removed by any end users. The retention policy manager is responsible for purging files. The software methods it uses are not exposed in any user level code or API. File systems which rely on permissions to simulate indelibility can easily be circumvented.

Massive Single Server Scalability – Engineered as an installable file system, StorFirst EAS currently imposes a limit on a single server of about 18 quintillion files (or 2^64). By supporting a 64-bit addressable namespace per server, as compared to Microsoft's 32-bit addressability, StorFirst EAS was designed for scale. The practical limit at the time of this writing using high density disk storage is about 40 PB/server, providing 200 times the scalability of NTFS.

High Availability – StorFirst EAS offers two methods of high availability support. The feature driven model is to leverage StorFirst EAS' asynchronous multi-target writes which provides built-in replication by dynamically writing to all available tiers of storage. This represents a new approach to lifecycle management and provides immediate access to data should any one of the tiered storage subsystems fail. The baseline model supports Microsoft clustering as well as mirroring of the StorFirst EAS directory. This capability allows a replacement server to quickly remount some or all of an EAS assembly rebuilding the meta-data storage directly from the on-disk file system(s).

Integrated Retention Policy – Seven10's StorFirst software supports simple user programmable retention based on file longevity. Data cannot be purged from a storage tier except through the deliberate action of the policy manager.

Application Driven Meta Data Extensions – StorFirst EAS supports an extensible meta-data schema. This schema is exposed to the application layer in the form of extended file system attributes. Advanced versions of the retention manager use these attributes (managed by the meta-data service) to provide fine-grained retention policy down to the file level.

MS Active Directory Support – StorFirst EAS uses AD and an authorization manager to provide user and role-based security through published Microsoft APIs.

Conclusion

StorFirst EAS provides the essential elements to support long-term compliant storage without blurring the line between OS services and application level responsibilities. Simple to implement and hardware agnostic, the Seven10 file system eliminates complexity and many of the limitations of appliance-based fixed content management systems. These attributes bring all the virtues of an infinitely scalable, tiered archiving concept, including lower storage cost, policy driven data retention, lower administrative burden, and data immutability and security.

About Seven10

StorFirst products deliver enterprise archive and disaster recovery software that helps organizations store data intelligently and securely. With proven success in healthcare, finance, and compliance-driven markets, and an install base that includes many of today's Fortune 500 companies, Seven10 is fast becoming one of the most trusted names in storage.