APARAVI®

Aparavi Active Archive℠

Architecture White Paper

March 2019

APARAVI®

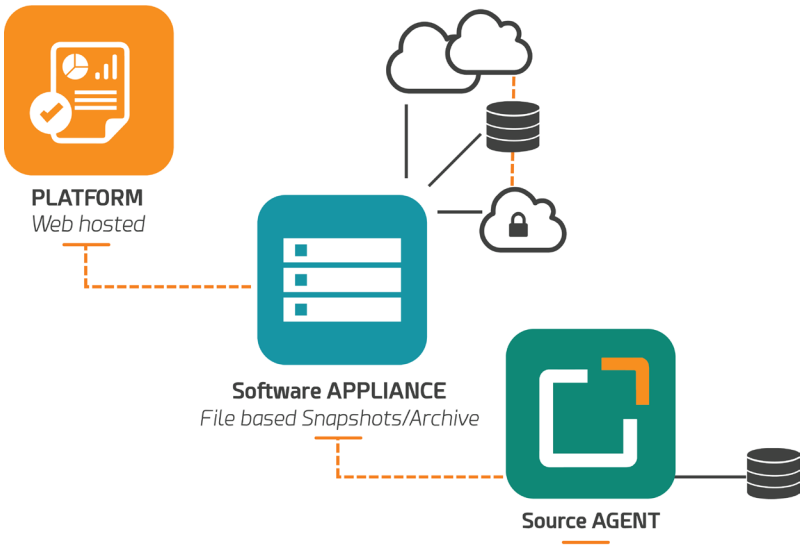
Table Of Contents

Aparavi® Active Archive™ delivers intelligent multi-cloud data management to organization with large volumes of unstructured data. Aparavi indexes, classifies, retains, and archives either on-premises or in any cloud. Our policy engine automates and simplifies data retention while our patented cloud-active-data-pruning reduces long-term storage by automatically removing data as specified. Data classification and tagging, along with full-content search provides fine-grained control over data stored, and enables regulatory and internal compliance. True multi-cloud management enables organizations to take advantage of changeing cloud economics, and an open data format removes vendor lock-in forever. Our SaaS model along with flexible, usage-based pricing means easier and predictable costs and for a lower total-cost-of-ownership.

This paper explores how Aparavi Active Archive works and what typical data management scenarios look like.

I. Aparavi Architecture and Storage Model Options

Aparavi Active Archive consists of a three-tier architecture and storage model. The 3-tier architecture consists of a web application, software appliances, and agents. These three tiers work in concert with the storage model consisting of snapshots (optional) and archives.



Aparavi Architecture

Agents

Agents can be considered as the data sources. Agents are typically file servers that contain a wide variety of unstructured data such as pdfs, spreadsheets, images, and more. These servers contain data that needs to be archived for long-term data retention. For maximum data security and efficiency, the agent encrypts, compresses, and de-dupes all data before it leaves the agent. The encrypted data is then passed to the software appliance or to the archive destination of your choice.

Software Appliance

Software appliances are usually on-premises machines that temporarily store data sent from the agents. These data sets are referred to as snapshots. An appliance can store any number of snapshots, but usually only a few copies are kept for quick retrieval of recently lost or damaged data.

Software appliances are also responsible for sending data to a long-term archive, such as a centralized data center (in case of a private cloud) or to a public cloud like Amazon S3 or Microsoft Azure.

Although appliances are typically setup utilizing on premises machines, appliances can also reside in the cloud. The advantage of placing software appliances on-premises is that the data transfer rate from agents to the appliance over a LAN is typically faster than an internet connection to the cloud.

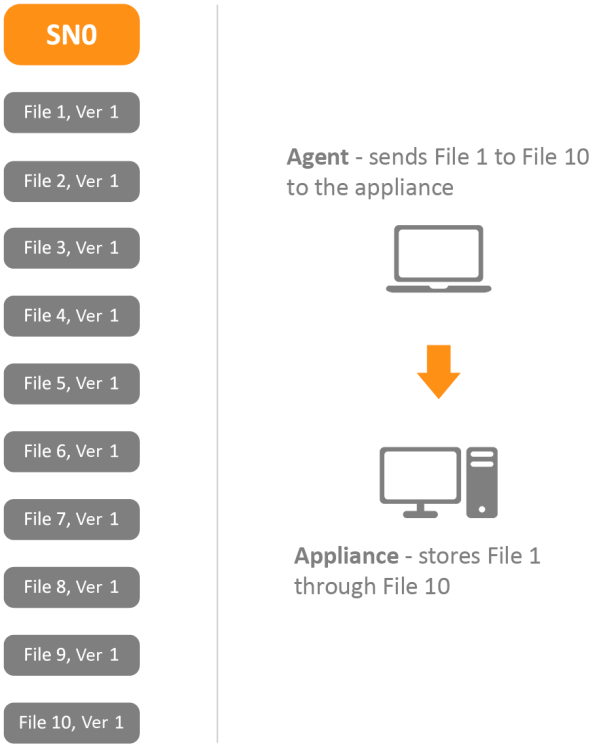
Software appliances are also used to fulfil full-text and metadata search requests and hold index information to expedite the search process.

Web Application

The web application is the overall monitoring and management system. It is where you go to monitor archive jobs, set up policies, manage users, and much more. The web application does not contain any source data and is not involved in any data transfers. It is used as a powerful centralized user interface for you to manage and monitor the overall Active Archive process.

Storage Model

To understand how Aparavi Active Archive works, it is necessary to define snapshots, archives, and policies. Snapshots and archives are both data sets with a defined retention policy. Policies define the archive and snapshot strategy (e.g., which files to archive, how long to retain). Each policy is predefined with default best-practice settings which can be used without modification in most instances.



Policies

Definition

Policies specify the type and location of data that should be archived, the archive frequency, the data retention strategy, the long-term storage services, and more.

Purpose

The purpose of policies is to establish an organization-wide standard for data retention.

Snapshots

Definition

Snapshots are an interim unit of storage. Creating a snapshot is an optional starting point for the archives. If you opt not to utilize snapshots, the agent data is sent directly to the archive destination in what is called “direct to cloud”.

The initial snapshot (SN0) is a complete copy of all data you choose to archive. Subsequent snapshots only contain data

that has changed since the last snapshot implementing an incremental forever archive strategy.

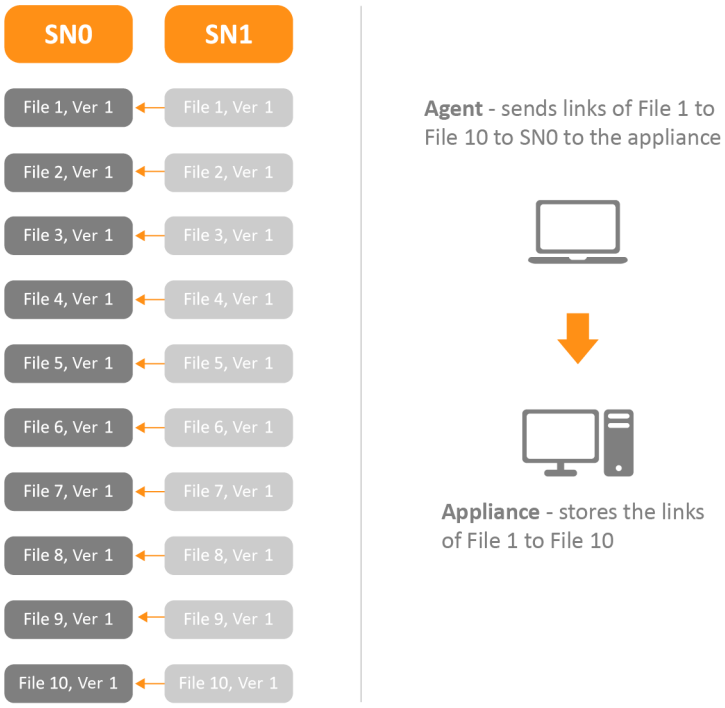
Purpose

The purpose of snapshots is to store a complete data set from a given point in time for short term retrieval purposes and act as a staging area for data transfer to the archive. If you opt for utilizing snapshots, the agent data is typically copied over a LAN to the software appliance. If you opt for the direct-to-cloud scenario, an appliance is still required but will only hold metadata for searching and retrieval purposes.

Example

The following section shows a simple snapshot example. For this scenario, thousands of files exist on a file server; however, only ten files meet the archive policy criteria for long-term data retention.

Archive policies are defined within the web application and propagated down to the software appliances and agents to perform the actual work. Once an agent receives the policy details, the agent starts processing the policy to create the first snapshot, called SN0. As mentioned above, this initial snapshot is a copy of all data that complies with the defined policy - in this simple scenario, just the ten files (but of course, could be thousands or millions of files in a real-world scenario).

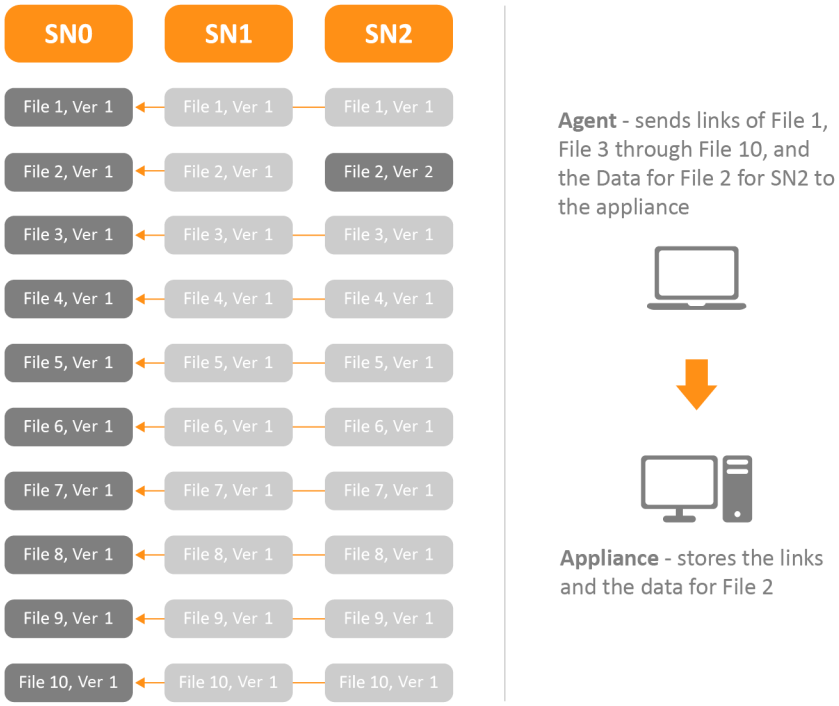


At the next scheduled time (typically once per day) the next snapshot will be created - in this case, snapshot 1 (SN1). SN1 will only contain files that were changed since SN0 was created (i.e., incremental changes).

For this scenario let's assume none of the ten files were changed. So, the agent will only send SN1 metadata to the software appliance. The SN1 metadata will contain links for each file that point back to SN0. No additional file content is required since none of the files were changed.

For the following snapshot, SN2, let's assume File 2 has changed. The agent doesn't need to resend the complete set of data for Files 1 through 10, since the appliance already has those files in the exact same state. The only data the agent needs to send to the appliance is data for File 2. For all other files, the agent once again just sends metadata containing links to SN0.

The snapshot process will continue in the exact same manner sending only file changes or new files to the software appliance until you have reached your predefined snapshot retention period. Once that retention period is reached, the pruning process begins (see pruning section below).



Archives

Definition

Archives are for long-term storage and typically utilize a cloud object storage service, such as Amazon Web Services or Microsoft Azure. An archive is a complete copy of the last snapshot including both the data and metadata. If utilizing a direct-to-cloud approach, the archive will hold the same information, but will not use snapshots as an interim storage set.

Purpose

The purpose of archives is to provide long-term storage to conform to your organization's data retention strategy.

Typically, archive policies are set up to run weekly, but they can be defined more frequently by adjusting the policy. A common schedule would be:

- Snapshots once a day
- Archives once a week on Friday

Alternatively, you can use a direct-to-cloud strategy. In this case, there are no snapshots, so archives could potentially run more frequently.

Example

For this scenario, let's assume the following: you opt for the traditional strategy using daily snapshots and weekly archives configured to use a cloud object storage service. The following is how the process will play out:

Day 1
As soon as the agent receives the policy details from the web application, SN0 will be sent to the software appliance containing all data that meets the archive policy.

Later in the day when the scheduled daily snapshot time arrives, a second snapshot (SN1) will be sent to the appliance containing all data changed since the initial snapshot earlier in the day.

Day 2
Snapshot 2 (SN2) will be sent to the appliance containing all changes since SN1

Day 3
Snapshot 3 (SN3) will be sent to the appliance containing all changes since SN2

Day 4
Snapshot 4 (SN4) will be sent to the appliance containing all changes since SN3

Day 5
Snapshot 5 (SN5) will be sent to the appliance containing all changes since SN4

At the end of the week, the first archive (AR0) will be created and sent to the cloud object storage service for long-term storage. The archive will contain the latest data stored in the snapshots created throughout the week. The software appliance will navigate through snapshot 5 back to 0 to produce the latest source data for the archive.

The following week, the process will start all over. The only difference will be that on the following Friday the next archive (Archive 1) will only contain changes since the prior archive (Archive 0).

II. Only the Changes, All the Time

Definition

Aparavi's unique Active Archive solution ensures that only changes are propagated to the snapshots and archives.

Purpose

Aparavi's delta-differencing functionality ensures that significantly less storage space is required as compared to traditional storage models.

The first time a file is archived it is saved as a full copy of the original file. The next archive will contain only changes since the prior archive.

Files < 1MB

Upon changes to a file that is smaller than 1 MB, the file is copied in its entirety for both snapshots and archives. For small files like these, the benefit of delta-differencing has minimal effect and skipped for optimal performance.

Files > 1MB

Upon changes to a file that is larger than 1 MB, delta-differencing processing is employed. Only de-duped byte-level changes to the file will be saved in the next snapshot or archive.

Process Flow

Traditional Strategy Using Snapshots

As an example, let's say that on Monday an existing 20 MB PowerPoint presentation is selected for archiving. The first time the PowerPoint file is copied to the software appliance as SN0, the entire 20 MB file will be sent because there are no earlier copies.

On Tuesday, you insert one slide in the PowerPoint presentation. Only that new slide will be copied to the appliance in SN1 because Aparavi's delta-differencing recognizes that only one slide has changed since the original snapshot. In addition to the file changes, Aparavi will update the archive metadata to include a pointer indicating that the rest of the file exists in SN0.

On Wednesday, you insert a second slide. Wednesday's snapshot will only include this new slide with a pointer to SN1 for the earlier changes which in turn points to SN0 for the original data.

On Friday, no changes are made to the PowerPoint. At the end of the day, the first archive (AR0) will be sent to the configured archive service. The initial archive will contain a single complete presentation as of Friday comprised of the original file from SN0 but with the changes stored in SN1 and SN2.

The following Tuesday, you insert a third slide. Only the third slide will be sent in the daily snapshot.

No other changes are made for the rest of the week. Then on Friday, the next archive (AR1) will contain only the third slide with a pointer back to AR0 for the remainder of the file.

Direct-to-Cloud Strategy

This scenario uses the same example as above, but with a direct-to-cloud archive strategy where archives are processed daily.

On Monday, an existing 20 MB PowerPoint presentation is selected for archiving. The first time the PowerPoint file is copied to the archive storage as AR0, the entire 20 MB file will be sent because there are no earlier copies. In addition, the file metadata will be stored on the software appliance for search and retrieval processing.

On Tuesday, you insert one slide in the PowerPoint presentation. Only that new slide will be copied to the archive in AR1 because Aparavi's delta-differencing recognizes that only that one slide has changed since the original archive. In addition to the file changes, Aparavi will update the archive metadata to include a pointer indicating that the rest of the file exists in AR0.

On Wednesday, you insert a second slide. Wednesday's archive will only include this new slide with a pointer to AR1 for the earlier changes which in turn points to AR0 for the original data.

The following Tuesday, you insert a third slide. So, on Tuesday, only the third slide will be sent in the daily archive.

Benefits of Methodology

The benefits of storing only the changes (all the time) are reduced storage capacity requirements (on the software appliance and the archive storage) and the ability to quickly and easily retrieve data from any point in time.

III. Data Pruning and Retention

Data pruning and retention are at the heart of Aparavi's Active Archive solution. Pruning ensures that only the minimum amount of data needed to recreate a given data point is kept in storage, greatly reducing storage requirements compared to most other methods in use today. Pruning is the process whereby data that is no longer needed is removed from snapshots and archives keeping your total cost of data to a minimum.

Snapshot Retention and Pruning Example

Let's assume a simple use case where 10 files fall into the archive policy, the snapshot retention period is five days, and daily snapshots will be created.

Data changes day by day as follows:

Day 1: 10 files identified for archiving

SN0 is created and contains all 10 files. The files are all marked as version 1 since it is the first version of the file that is being archived.

Assuming no changes to any of the 10 files during day 1, SN1 will be created as the daily snapshot with links that point from SN1 back to SN0 for all 10 files.

Day 2: file 2 was detected as changed

SN2 is created. It will contain the changes to file 2 and marked as version 2 with links back to SN0 for the original file. All other files will have direct links back to SN0.

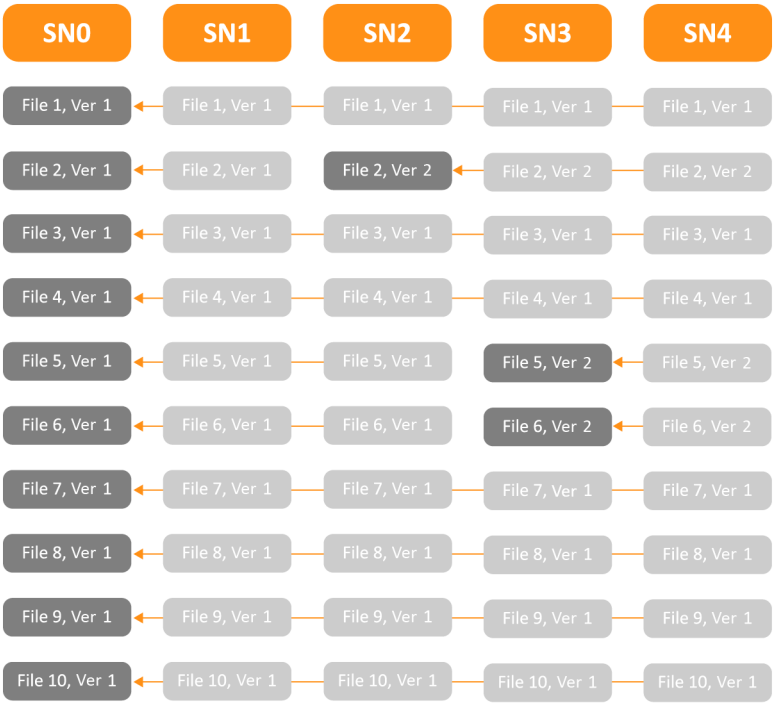
Day 3: files 5 and 6 detected as changed

SN3 is created. It will contain the changes to files 5 and 6 as version 2 with links back to SN0 for the original files.

Day 4: no changes detected for any of the 10 files

SN4 is created. It will reference:

- SN2 for File 2 V2
- SN3 for version 2 of Files 5 and 6
- SN0 for all other unchanged files



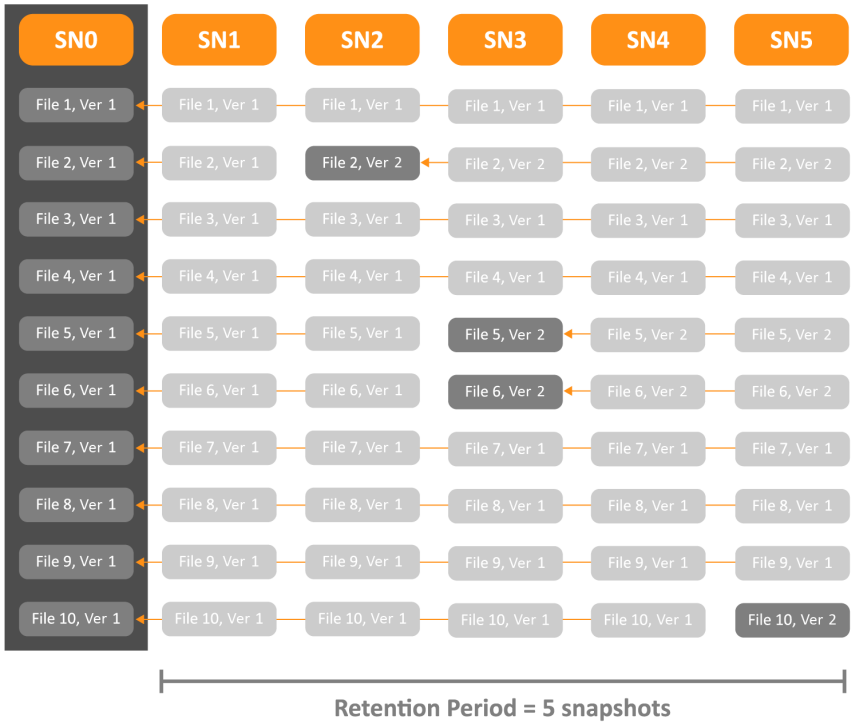
Day 5: file 10 is detected as changed

SN 5 is created. It will contain File 10 V2.

In addition, something new happens since we have hit the retention period of 5 days: pruning can begin.

Pruning means we remove all data out of old snapshots that are no longer needed for file-level retrieval.

Since the policy dictates 5 snapshots, SN0 is no longer presented as a retrievable snapshot. However, even though it is not listed as one of the five retrievable snapshots, all the data contained in SN0 must be kept since all files have a reference to that snapshot.



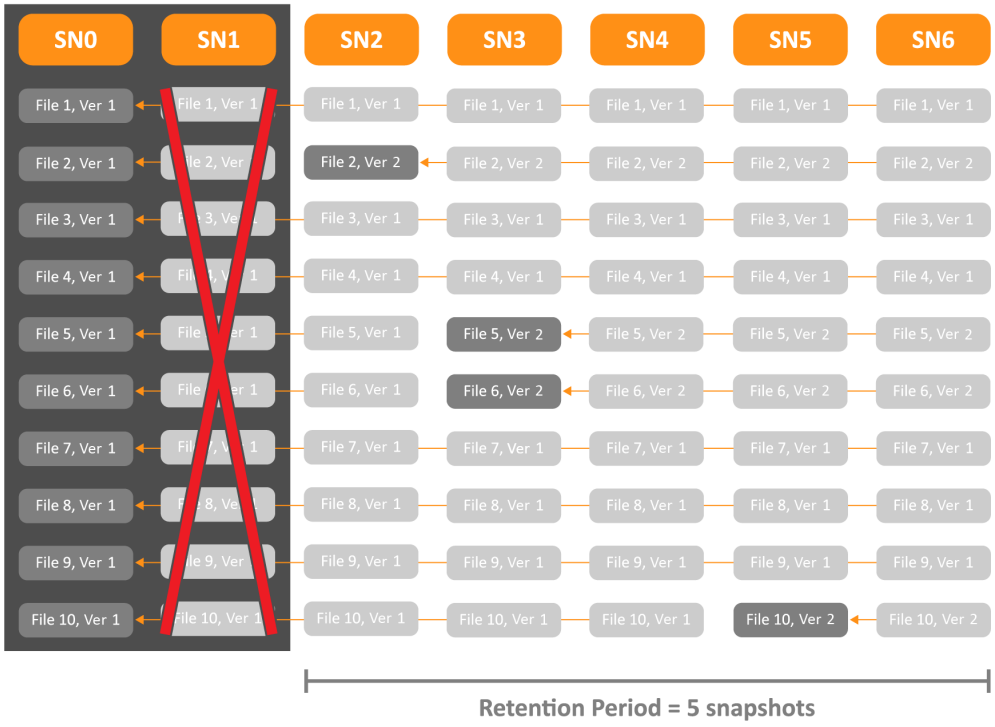
Day 6: no files detected as changed

SN6 is created.

The pruning process determines that nothing in SN1 is needed to support recovery of files contained in SN2 through SN6, thus SN1 is completely pruned.

However, all the orginal 10 files in SN0 are still required, so none of those files can be pruned nor can SN0 be pruned in entirety.

As was the case on day 5, SN0 does not appear as a recoverable snapshot even though it contains most of the data.



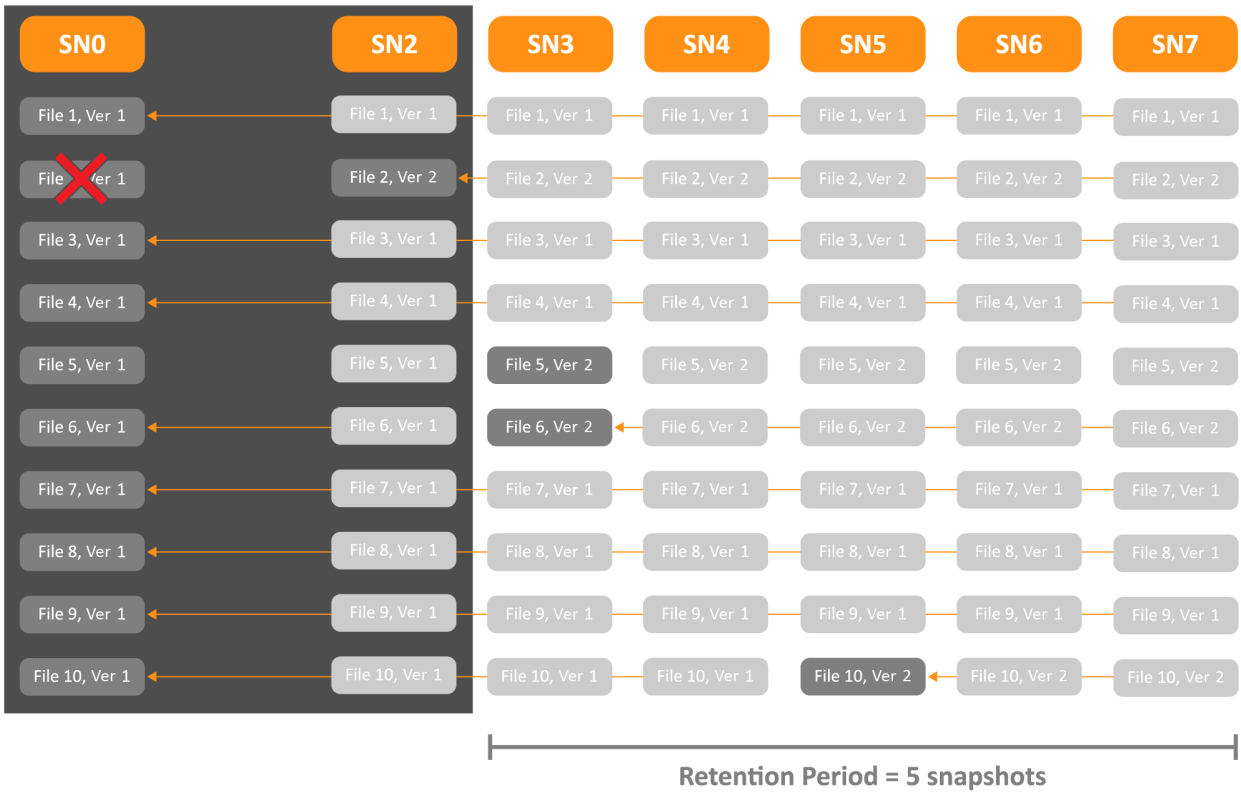
Day 7: no files changed

SN 7 is created.

When no changes are detected only links to prior snapshots are sent from the agent to the software appliance.

However, something different happens during the pruning process. File 2 version 1 and version 2 are both now in non-retrievable snapshots. There is no reason to keep both versions of the file. First, SN2 is updated with a merged copy of File 2 version 1 and 2. Then, file 2 version 1 is pruned from SN0.

As you can see from this simple example, Aparavi only keeps data required to adhere to your data retention policy. Aparavi will prune data as it 'falls' off the retention cycle reducing your storage requirements to a minimum.



Archive Retention and Pruning

The same pruning process discussed for snapshots is utilized for archives. However, an archive retention policy is typically set as several years with a default of 7 years. Archives will continually be pruned, and files removed as soon as they meet the retention period.

Benefits of Methodology

Aparavi's storage model allows you to seamlessly retrieve files from any combination of on-premises or cloud storage. Pruning data that is no longer referenced keeps storage space to a minimum reducing your overall cost of data.

This method is expected to reduce total storage needed over time by 75% as compared to traditional solutions and completely ends storage vendor lock.

IV. Content Indexing and Classification

As files are selected and prepared for archiving, the content of each file is scanned and indexed. The indexes are passed to the software appliance to expedite search functionality. In addition, during the file scanning process, files are optionally tagged based on classification rules defined in the policy.

Content Indexing

All supported file types (including modern Office documents, PDF's, text files, and zip files) are scanned during the archive process (direct-to-cloud scenario) or the snapshot process (traditional approach). All unique words are stored in index files on the software appliances. The content indexing allows for full-content search without accessing or downloading the archive files directly.

Classification

Prior to archiving data, you can optionally setup classification rules. Classification rules contain search strings, patterns, or metadata such as file name, file location, and file type. As documents are indexed, the classification rules are processed to determine if one ore more rules apply. If a classification rule applies, the document is tagged with the classification name. Examples of classification rules include "Confidential", "Sales Related", "US Social Security Number". Several example classification rules are delivered as default polices to be used as examples and speed implementation time. When searching for documents in the archives, you can also use classification names to expedite the search or filter the results.

V. Full-text and Metadata Search

As a powerful Active Archive solution, Aparavi allows you to search for files within archives or snapshots to respond any number of business requirements, such as legal discovery, lost documents, test data management, and much more. You can perform a search not only by metadata (e.g., file name, date-last-changed), but also on text within documents.

Metadata Search

File metadata is often very useful information to help narrow down your search. File metadata includes the following:

- Classification tags
- Date file was created, modified, and last-accessed
- File size
- File extensions (e.g., PDF, DOCX …)
- File name and location

Classification tags can be used as a valuable search criterion. If you previously setup classification rules to organize your organization's documents (e.g. sales, confidential), you can now utilize those classification names during search to provide a narrowed-down list of documents to streamline your search process.

Full-text Search

In addition, to the metadata search, Aparavi gives you the ability to search the content within each file.

Full-text search provides the following options:

- All these words
- Phrase
- Any of these words
- None of these words
- Pattern

As example, you could search for files that include the terms "Financial" and "2019" with the intent to find all financial documents created in the year 2019.

Patterns allow you to search for ID's or document numbers such as account numbers or phone numbers. As example, a US phone number pattern would be ???.???.????. The "?" represents a single-character wildcard while the "." Represents a delimiter such as a dash (-) or period (.).

Once files have been identified, you can request the documents to be downloaded out of the archives or snapshots. As mentioned above in the indexing section, the search process does not access the archive data directly, but rather utilizes indexes stored on the software appliances. Only when you actively request documents for retrieval, will Aparavi access the archives. The Aparavi solution reduces overall data transfer and their resulting fees while providing an effective search process.

VI. Conclusion

Aparavi Active Archive helps organizations master out of control unstructured data growth. With a three-tier architecture consisting of a web application, software appliances, and agents, we simplify the archive process. The software appliance executes the policies and acts as an optional intermediary storage repository for file-based snapshots. The policy-driven approach is extremely flexible, allowing you to select the number of snapshots to retain and number of years to retain your archives. In addition to meeting corporate governance requirements, Aparavi active archive also delivers true storage independence with on-premises and multi-cloud mobility.