STARFISH WHITE PAPER August 2022

Summary

Starfish Storage is an enterprise class software solution that:

- Helps you understand, manage, protect and optimize your primary and secondary storage, at both executive and detailed levels
- Provides visibility (both current and historic) into file systems, directories and files
- Orchestrates the data-management lifecycle and research data pipelines from file detection to data migration among file systems and object stores
- Provides data actions including move, copy, delete, hash, archive, backup and restore
- Keeps going long after applications, scripts and ad hoc management methods exceed their limits
- Can be used by storage administrators as well as power users

Take the "meh" out of metadata
Put the "chi" in archiving
Take the "rage" out of storage
Put the "tada" in metadata

With Starfish, previously opaque file system metadata is made accessible and actionable. Quickly locate files and groups of files. Get immediate information on file counts, sizes, duplication and ages, disk utilization (`du`), and change rates on billions of files and petabytes of data. Make data-informed decisions and automate actions (move, migrate, backup/restore, archive/restore) from the results of those queries even at very large scale.

Starfish creates and maintains a Starfish Unstructured Data Catalog (or UDC) containing file metadata, file history, per-file job results, and customizable data such as ad hoc tags or metadata extracted from the files. The file metadata is gathered using Starfish Scan Accelerators and high-performance scanning. The Starfish database can be queried and updated using our sophisticated CLI, HTML 5 GUI, and API interfaces.

Starfish includes a job manager that provides automation and data-management orchestration, and includes default operations such as archive and backup but can run arbitrary commands. The job manager is a flexible framework. It allows integration with a variety of custom workflows, and scales jobs easily across any number of agents. Job results can be referenced or acted upon by querying the Starfish database.

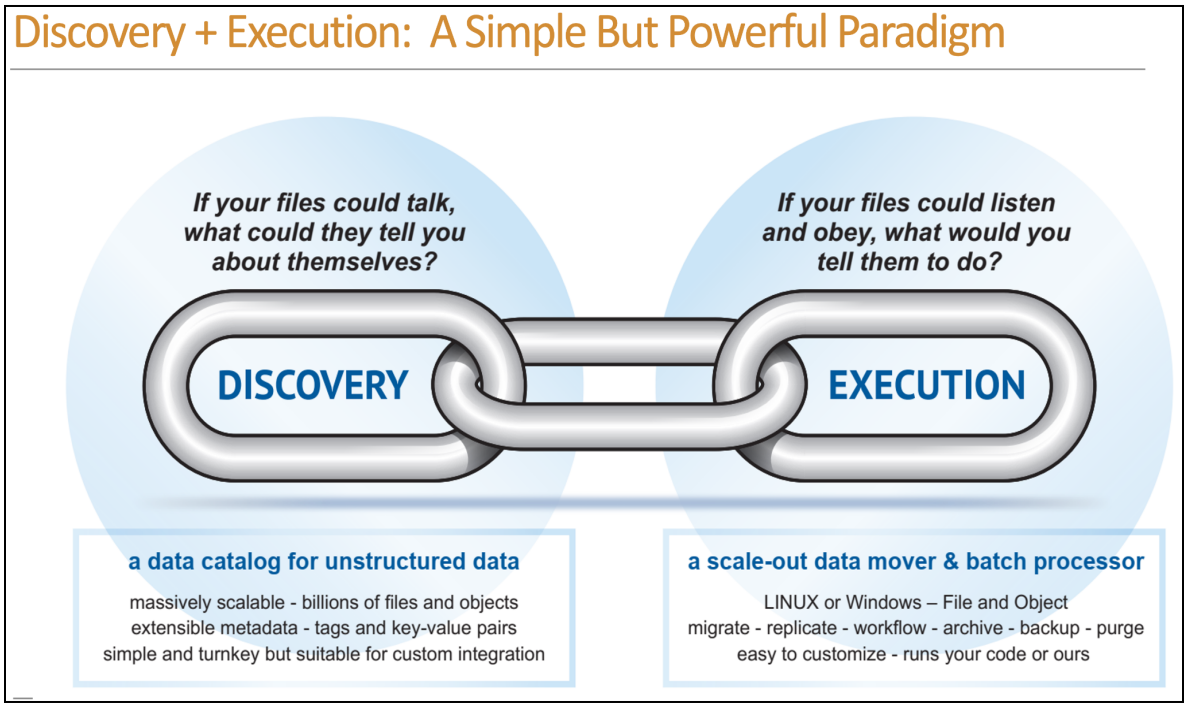
Table of contents:

STARFISH

Overview

Big Data has made a big splash recently, causing alarming warnings of the upcoming Big Data deluge. Countless talks have been given, articles written, and research presented on the three (Four? Five?) V's of Big Data (Volume, Variety, Velocity, Veracity, Value) and the challenges they present. As the amount of data has increased, challenges that used to be limited to the very top-most tier of organizations like the Library of Congress, NIH or Google are now faced by smaller companies, individual departments, and even individual research projects. Due to the proliferation of research instruments generating enormous amounts of data, the explosion of new image and video content, research data being shared globally, and the reduction in the price of storage, more organizations than ever are faced with managing petabytes of data, in billions of files, on-site and remote.

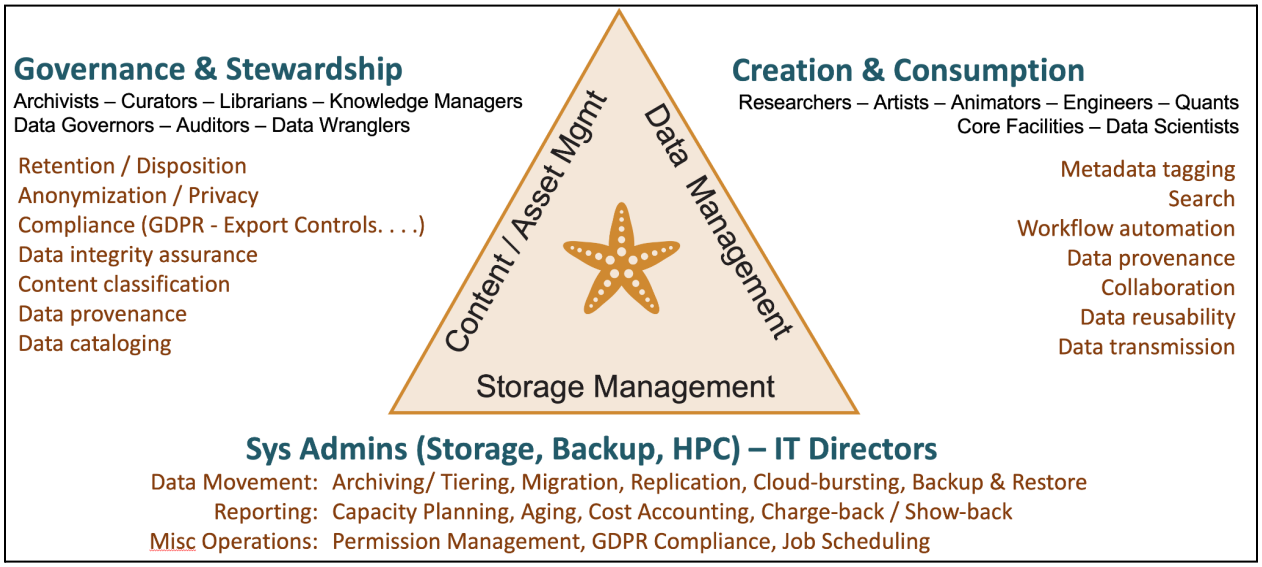
Managing this ensemble of data is time consuming, risky, expensive and usually tedious. What is simple to do with one million files gets complex as it grows through billions. Some sites try to ignore the problem, letting files accumulate or buying more storage, which is inefficient and gets very expensive very quickly. Others try to build their own tools for monitoring and managing storage. However, such DIY tools pull resources away from other projects, and in many instances eventually become abandonware. Others attempt to use open-source or other commercial tools to solve the problem, but these solutions can't function efficiently in large-scale environments. Many are inflexible or designed for a specific role rather than to meet the needs of all stakeholders. Also, these tools are often appropriate to only one type of file system or storage device when, in practice, many sites have a diversity of platforms , including the cloud.



In conversations across a broad range of industries - scientific research, media and entertainment, oil and gas, government and education - the same problem is revealed: there is an abundance of data, and the data users want to sort or manipulate this data by some meta category or concept, such as a project, grant, or contract. Once finished with the data, they wish to move the data off to an archive, without the difficulty of managing the 50,000 individual objects. Ideally, researchers and IT users want to manage all data in aggregate, with lightweight and highly efficient tools that should allow for the usage of file, block and cloud storage, both locally and remotely.

Roles that Need Starfish

Most storage management and content management systems are geared toward one type of user, the IT Operator, the Content Creator, or the Publisher and Preservationist. The people managing the storage have different drivers than the Content Creators (scientists, engineers, and artists), who also have different incentives and concerns than those involved with Publication and Preservation (librarians, archivists, curators). While some tools meet the needs for one type of user, they can leave others out in the cold, with no way to access or control the tools. That leaves an organization using multiple tools or empowering one type of user without enabling others to do their jobs efficiently. Starfish was designed to be useful for all three of those roles, with many features already available.



Today Starfish allows access by superusers, as well as project leads (called "zone administrators"). Zoneadmins can include primary investigators, department heads, and so on. Those zoneadmins are defined by the superusers, have access to assigned directory trees, and have superuser-defined tags for actions and classifications. Actions based on zoneadmins assigning tags are defined and executed by the superusers. Zones allow content creators to analyze, report on, understand, and manage their data sets. In the future, end users will have similar features with respect to files they own and have access to.

If there were a way to annotate data at creation and have that metadata follow the work regardless of where it is stored or moved, the challenges associated with data management for all of these three roles would be greatly reduced.

The Problem – Current Technologies are Incomplete

Current technology and purchasing processes have resulted in large organizations having multiple storage products, introducing a wide range of management tools, file systems, storage devices, types of storage, and storage locations. Whether this is due to procurement requirements, vendor viability or changing needs, many organizations have found that a single hardware vendor's integrated solution is highly unlikely to meet all requirements.

There are a number of current storage systems that support transparent tiering to allow for migration of data to less expensive storage or tape (Isilon pools, GPFS AFM, etc.). There are also a number of ways to support archiving management (SGI DMF or HPSS). What most of these solutions have in common is some type of

lock-in to their technology and/or proprietary formats. Few tools work with multiple storage systems and allow multiple types of data movement.

Similarly, there are systems and available tools for simple data management (SRB, iRods, Datadobi). These tend to lack scalability and functionality, requiring extensive administrative effort, scripting, tuning and tweaking as well as expensive servers to maintain, dramatically reducing their effectiveness. To add to the overhead of managing large-scale datasets, virtualization solutions (such as MarFs from Los Alamos, which uses GPFS to front-end a wide range of object stores) are increasingly being used. These tools will typically provide solutions for specific use cases such as data migration, data federation, data stewardship, or large-scale data storage, but don't fulfill the general requirements of large-scale metadata management and actions based on metadata queries.

As the complexity of environments grew, the paucity of useful tools became obvious.

The Solution – Metadata Management with Starfish

For a solution to succeed, it must meet the requirements of the IT Operators, Content Creators, and Publication and Preservationists. Metadata is the key to managing data at a higher conceptual level. As an example – the U.S. National Center for Biotechnology Information's "bioproject" includes experiments, samples and analyses. Over time, it may be optimal to archive the raw data and keep the analyses. At that point, it would be optimal to have high-speed "find" of all the data according to an attribute, and migrate all the raw data to archive storage as a single batch operation. Tracking the old and new locations via metadata would allow contiguous operation; the view of the file system via its metadata would stay the same while the raw data is moved out of the way.

Some Use Cases for Starfish

- Locate all files by owner, date, size, or custom tagged metadata
- Automatically archive data based on usage patterns
- Generate Chargeback/Showback reports
- Automatically update old versions of files (for example to new codecs)
- Manage data pipeline from cradle to grave
- Identify PII-, PHI-, and GPDR-containing files
- Extract metadata programmatically from files and store in Starfish database
- Automatically add custom tags based on policies (Anything produced by MachineX, by UserY, gets TagZ)
- Manage file provenance and ensure data consistency
- Automate file lifecycles according to grant specifications
- Identify largest storage consumers, recent growth
- Quickly see the monthly cost of storing each volume, directory, etc.
- Move files into the cloud or object storage (Amazon, Oracle, etc.)
- Rapidly migrate data using Starfish parallelized data mover
- Ensure file integrity with fixity checks
- Find all files impacted by a virus (CryptoLocker) or ransomware
- Quickly generate reports about files and data storage on large/multiple file systems
- Instantly see directory rollups (DU) with size, modified, file count
- Find and manage orphaned files (files with no owners)
- Implement low-cost DR (backup to AWS Glacier)
- Extract the liquid produced when threatened and use it to treat inflammatory illnesses. (A feature of the echinoderm not the software.)

Such an approach can be useful in many ways. If it is updated periodically (based on site need) with all file changes, a history of all files becomes available. For example, if data is copied into an archive, with a simple query based upon the creation time, a complete, up to date metadata catalog could show the system before, during or after that copy.

What if the data was moved to a new storage system? How does this affect users? Currently, this is usually handled via folklore – that is, a note is posted that file system `PRJ24` has been moved to `Isl-south:/2015`. Unfortunately, all files and paths of that file system mentioned in workflows are now broken and need to be fixed. But, by tracking the history of all data in an environment, such folklore is no longer needed. Data replaces Lore. The metadata now reveals where a set of data was and where it has moved to, allowing tools such as electronic lab notebooks to still locate and work with the files they are linked to, regardless of whether or not they are stored in new locations.

Driving activities based on metadata liberates the IT Operators to focus on data at the appropriate level of abstraction, rather than worrying about files, file systems, tracking spreadsheets, myriad tools, and ad hoc file movers like rsync.

It becomes simple to ask questions such as how many files or how much space is taken by:

- duplicates
- files owned by a given group
- files unaccessed in the year
- files market with an ad hoc tag
- files owned by a given user
- files matching specific criteria.

Also easy to find are:

- details about whether files have changed (fixity)
- space allocation over time
- and so much more!
- how many files have been added/removed/changed
- what files have caused free space to decrease

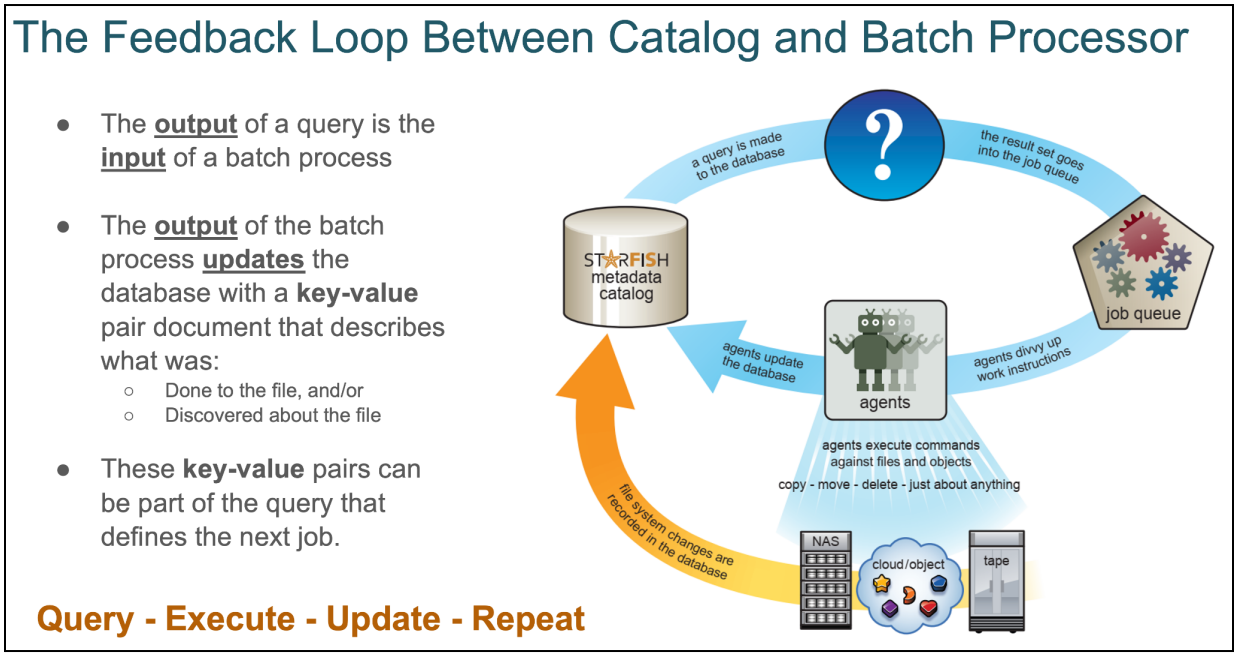
Once the complete metadata of all file systems and object stores, including history, is in a fast, up-to-date database, the list of potentially useful questions is endless and the answers become key to data and site management. The ability to take actions based on the metadata is key for all three roles: consider project owners archiving a specific file type, storage managers migrating unused files to lower cost storage or the cloud, and archivists tracking files via ownership tags.

STARFISH

This backdrop lays out the vision and mission for Starfish - to add intelligence to the management of large data sets, keeping the viewpoints of the IT Operators, Content Creators, and Publication and Preservationists in mind. As in the classic illustration of blind men describing an elephant, each of these groups see a different world of data. Starfish enables all three roles to be more effective and more efficient, accomplishing activities difficult or impossible before, and allowing for a view of the entire elephant. Starfish meets all of their requirements by harnessing metadata from the file system and objects, metadata from the file contents, and provenance information and metadata from processing pipelines. It assists in the management of data and works in heterogeneous environments, both on premise and cloud storage.

Product Vision

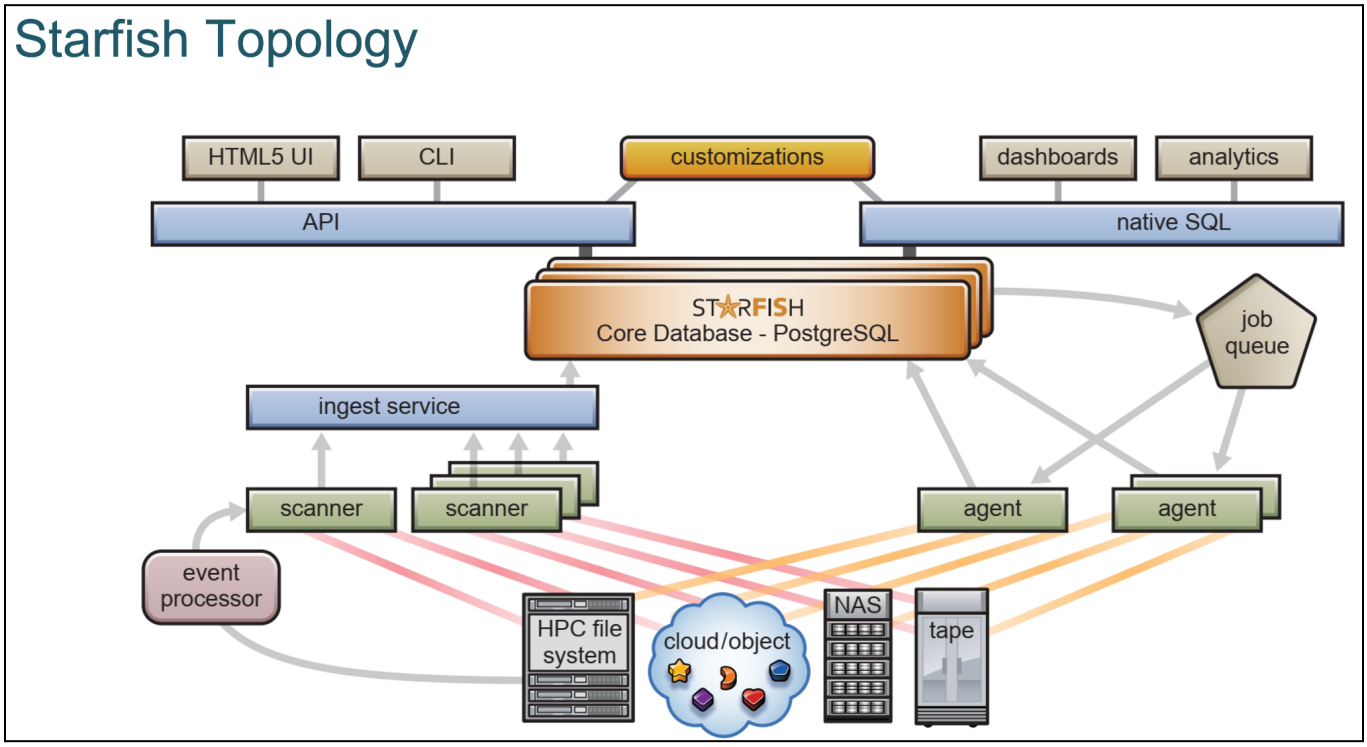
The idea of Starfish arose from the inability of POSIX file systems to carry additional information and easily manage large numbers of objects. In POSIX, a file is labeled by its path; move the file and you have a new path. It is the same object, but for applications, there is no record of this change or ability to track the file. Files themselves contain inherent information - metadata - that if easily accessible can permit more intelligent operations to be performed on the unstructured data.



As a broad generalization, IT Operations cares about the bits and bytes, data location, backup/archive, compliance, purchasing, and implementation of new hardware systems. Moving petabytes of data around becomes a chore and 99% correct is not good enough. This leads to a great deal of time spent checking and double checking during data movement operations. Associating data usage to specific projects is a daunting task, and managing and maintaining the total allocations to a project is often so difficult that it is only done on an as-demanded basis. Starfish provides IT Operations with an easy way to search and address any set of data, and to gain visibility into its attributes and location. Then Starfish goes one step further - allowing IT Operations to quickly take action on any queried set of data (*a predicate*) by applying a command against that data set. Alternatively, IT Operators can also create policies to perform repeated or scheduled actions on any data which fits a given set of criteria.

Content Creators want to focus their time managing the experiments, projects and workflow of their data. If they can be freed up from managing thousands, millions or even billions of files to allow them to manage a single project, with different states of data (for example "ingest", "qa", "scratch", "derived", and "publication"), it allows them much more time for research, engineering, video production, and other productive activities. If researchers can press an "archive" button at the end of a grant, it eliminates the time spent by Content Creators and IT Operations who previously had to initiate the archiving and tagging themselves; all appropriate files move to an archive with the correct data retention flag associated. Auditing burdens are lessened because the process is automated. Starfish can assist Content Creators in automating workflows, movement, and actions taken on files. Starfish can also be used to automatically apply metadata tags such as project names, grant specifications, or retention requirements on files that have certain attributes.

For the Publication/Preservationists, including librarians and curators, Starfish provides a wide variety of solutions. It can provide full identification of the types and contents of files via its suite of interfaces. Tags can be added (for example to track authorship or importance), then subsequently queried, with actions performed against the results. Libraries for example use Starfish to receive metadata from other sources to drive archiving of digital assets to various repositories.



What does Starfish do?

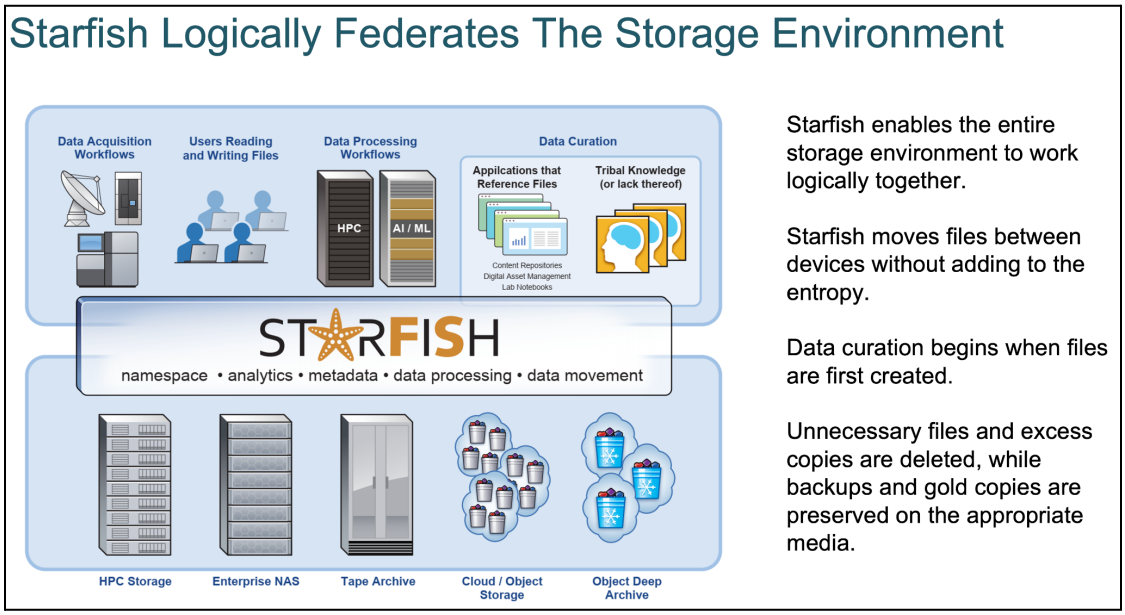
- Starfish is an application designed to manage institution-scale file collections.
- Starfish associates metadata with files, directories and directory trees on conventional POSIX file systems and NFS-mounted devices, CIFS mounts, cloud storage services, and object stores, and stores the derived data in full.
- Starfish extracts raw and derived properties from the contents of identified volumes into the UDC.
- Starfish can execute actions based upon query predicates, such as archiving based upon tags.
- Starfish keeps version histories of all scanned files and objects, allowing data provenance and change tracking.
- Starfish can store checksums of all items and use those checksums for fixity checking, duplicate detection and content addressing (for example store a checksum in the site lab notebook to find the item no matter its location).
- Starfish can even be used to automate research data pipelines, track down and remove ROT (Redundant, obsolete, or trivial) data, and a myriad of other per-site data management needs.

Metadata is leveraged for search, classification, reporting, graphing, and driving a jobs engine that enforces storage management policies. Storage management policies may include backup, archiving, disposition, data integrity checking, and deduplication. Policies are programmable, not limited to any built-in functionality.

Starfish metadata collection and management sits outside of the data path, such that it is transparent to users and does not introduce latency or points of integration or failure.

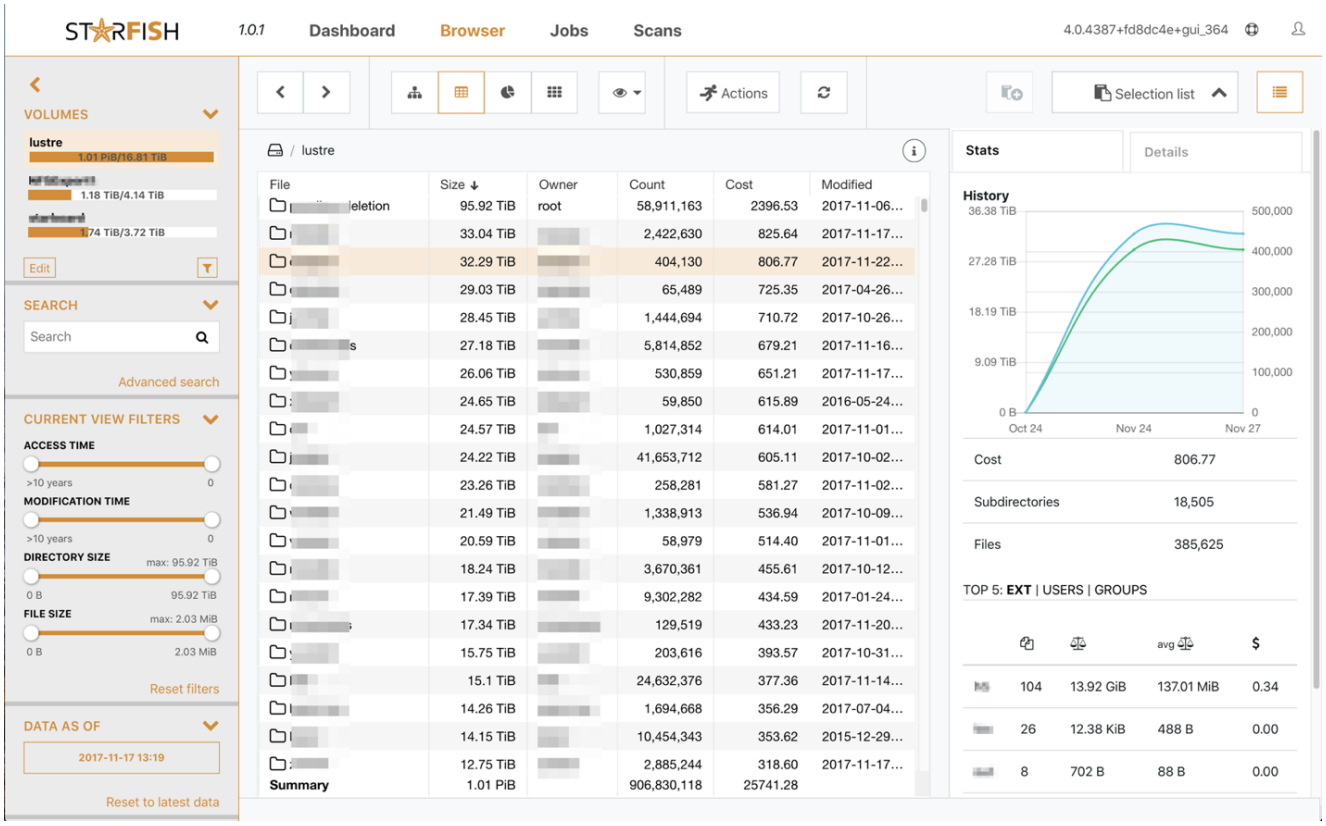
Starfish is enterprise-class storage management software. It is designed to handle billions of files, provide version histories, and provide a unified namespace across a diversity of storage devices. Starfish enables both common workflows (such as backups and archiving) as well as customizable ones using policies that are refined by extensible metadata.

Starfish creates an ecosystem that incorporates both the file system metadata (names, create times, permissions) and enables storage of file metadata (author, DICOM header, PDF version, etc.). It creates a catalog that spans file systems and storage subsystems, unifying them into a common infrastructure. Starfish maintains this information via out-of-band scanning or event monitoring from the file systems to keep this catalog current.



Starfish is built with a modular architecture that has three main sets of components:

- Starfish Unstructured Data Catalog (UDC) includes a PostgreSQL database that holds all metadata for all of the files and objects in your enterprise or stored within a selected group of devices. It maintains a history of those items over time, is fed by the Scanning Engine and is used by the Jobs Engine. It can be explored via an API, CLI and a GUI that includes summary charts as well as a file/directory browser.
- The Scan Engine – Acquires file and object metadata and updates the catalog. It includes highly sophisticated tools for synchronizing the metadata contents of your storage devices with the UDC, via POSIX file system scanning or via the Starfish Scan Accelerators (currently available for Lustre, Qumulo, PowerScale (Isilon), GPFS, and HPSS). These Accelerators use native file systems tools to detect differences in near real-time, allowing Starfish a nearly-up-to-date view of the file system.
- The Jobs Engine – Performs actions against your files. Jobs are initiated by a scheduler and consist of a query to the Catalog Server metadata database and an associated process. The results of the database query go into a central queue. Starfish agents pull work out of the queue and run the process. Agents can be run in parallel (scale out, across multiple systems) to handle large volumes of work. Agents can be metadata managers or data movers. Agents then return results to the catalog. Jobs can be defined as policies and continuously run against any data that meets a set of query criteria.



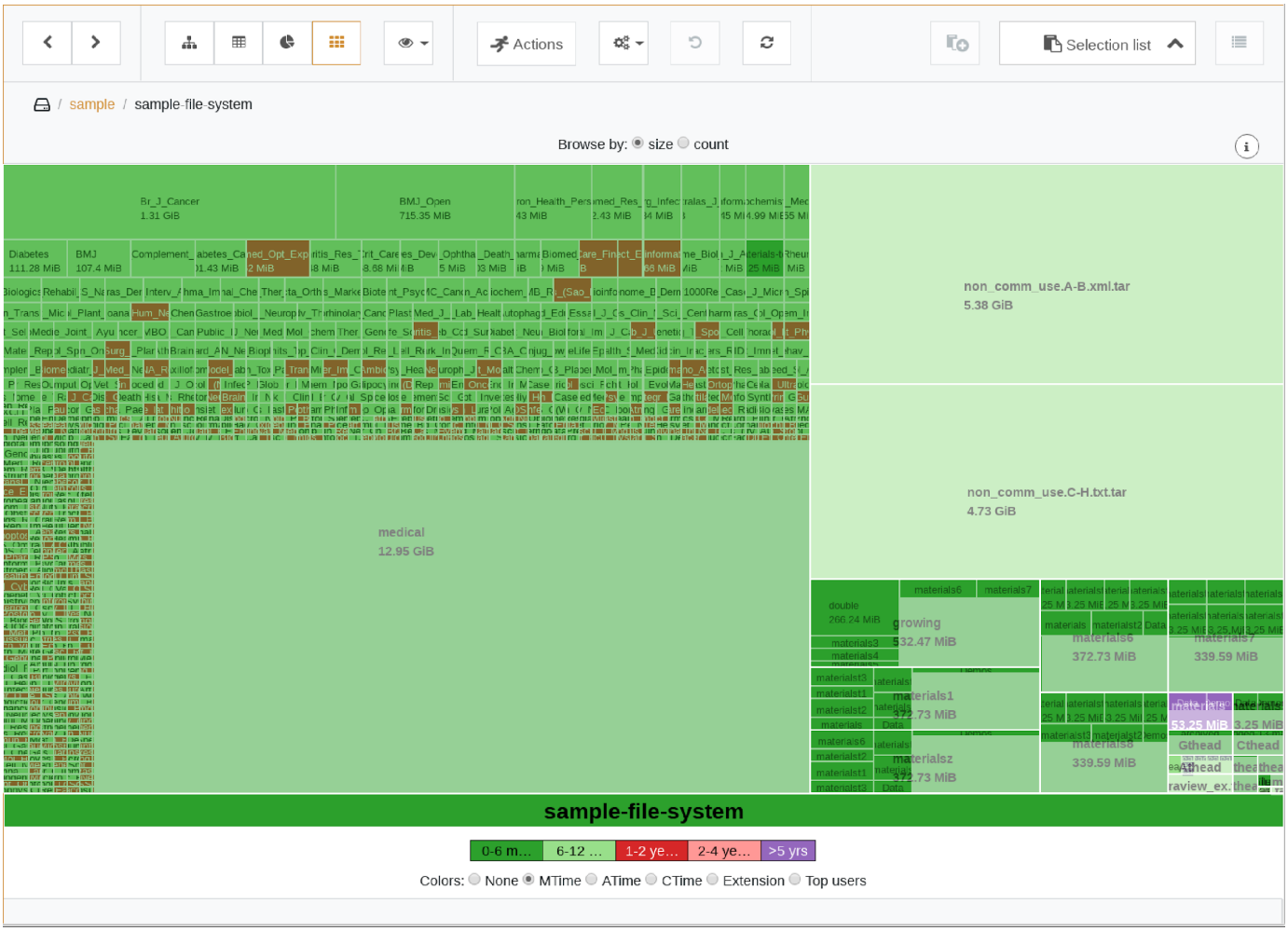
The Starfish Unstructured Data Catalog (UDC)

Once you have a catalog, there are many opportunities to derive value. The first example is reporting. Once the catalog is in place, querying for "all files belonging to user X" becomes straightforward, even though there may be millions or billions of files across dozens of stores (file system and object). Aggregate analyses become possible. For example, reporting on all files belonging to each user, binned by age of access, or similarly by group.

Since metadata can be extracted from files, reports listing the cost per department of storage are enabled for showback or chargeback. Often, all that is missing for better data management is better communication of costs and utilization to the appropriate individuals.

The catalog also permits users to create unique subsets of data. It's a database - so database queries can reduce the datasets to useful components. For example, copying data to an archive can be accomplished by requesting a copy of all files not yet migrated to the archive, or that request can be filtered to only migrate all files not migrated with an age greater than two years and an extension of ".MOV". Creation of rules and filters enable flexibility in processing data. Starfish allows action rules to act upon data sets.

Starfish supports cloud and object stores, allowing archive/recovery, backup/restore, and migration to S3 services (including on-site S3 like Amplidata, GPFS, NooBaa, Symphony), and to Openstack/Swift object stores (including Oracle Cloud).



The Scan Engine

The Starfish Scan Engine ingests file and object metadata into the catalog. Starfish scales out to support larger workloads by parallelizing the scan effort across all the cores permitted to break the work effort down into smaller tasks. In a large environment with multiple file systems and multiple Starfish agents, the work is distributed across the agents, with information fed back to the UDC.

Starfish Scan Accelerators take advantage of unique file system features to accelerate the flow of data into the UDC, enabling your Starfish catalog to be current with the state of your file system within minutes of detecting these changes and reducing the I/O overhead for both Starfish and the storage system under management. Scan accelerators are currently available for Lustre, Qumulo, PowerScale (Isilon), GPFS, and HPSS, with more planned.

The Jobs Engine

Actions on files and objects are managed by the Starfish Jobs Engine. The Jobs Engine essentially provides a small compute-cluster scheduler that analyzes queued requests and intelligently distributes that work to servers capable of meeting the need, dynamically balancing for load, data access, and priority. This jobs engine accepts both predefined and custom tasks. Predefined tasks such as moving data, archiving, copying to cloud, or hashing of data for validation, come standard. Custom tasks are created by the Starfish administrator or via professional services, can be written in any language, and extend the functionality of Starfish. As an example, one site was interested in extracting several fields from MRI files to drive their workflow. Using open source tools and a simple filter, these fields were extracted, added to the UDC and were then available for further queries and dataset operations.

Any job that outputs JSON has that result added to the UDC. For example, jobs that extract metadata place their results as JSON back into the catalog, where they are available as structured metadata for future queries and jobs. Starfish also supports a more freeform tag structure, which permits inherited and non-inherited tags to be applied to files and directories. As an example, a directory could be tagged by grant and owner. Subsequent jobs may execute an archive or move task including those tag values as parameters.

In Starfish, a job is a task that is run based on a schedule and against a collection of data. The collection is defined as a set of files or directories with records retrieved from the catalog. The collection is specified by several methods: matching tag, file type, age, modification time, size, extension, and regex on file names or paths. If this is not sufficient, an extended query can be defined which utilizes the full power of the database behind the catalog to specify the collections.

Once a job has been initiated, it collects its dataset and pushes that onto a jobs queue. The queue manager identifies eligible worker processes across the job cluster and distributes tasks across the cluster. Jobs respond with status and results. These results are, in turn, updated back into the catalog and tagged back to the original objects.

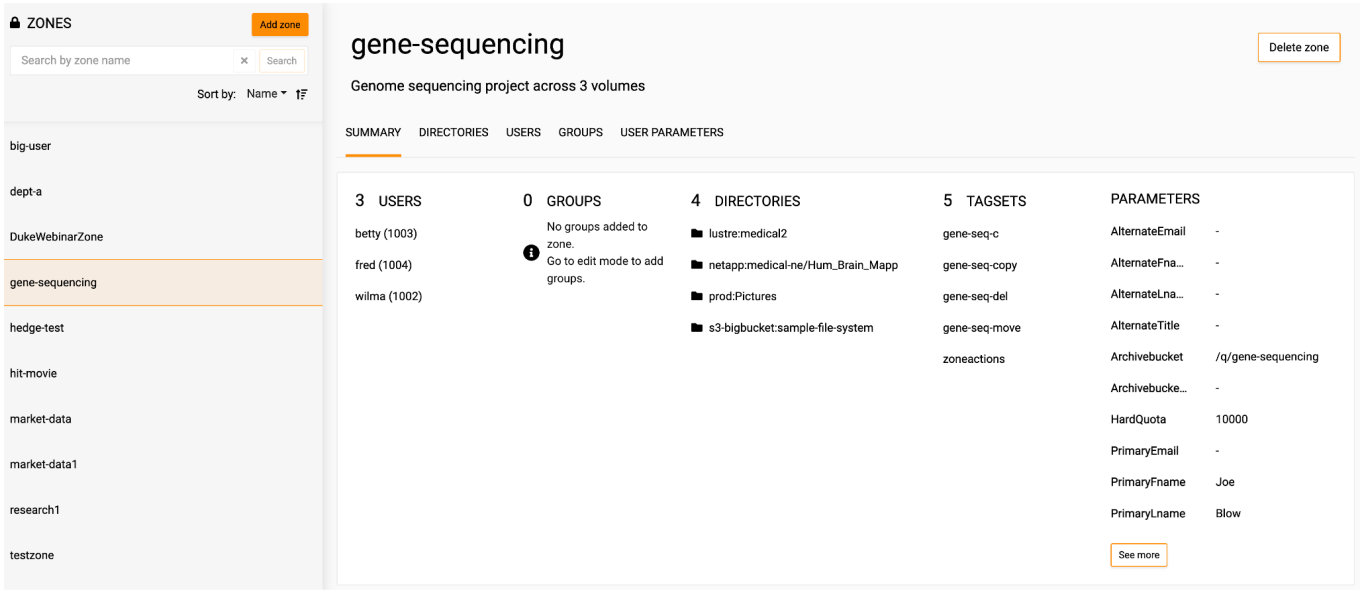
```
[root@osboxes query]# ./query-all.sh
This job runs a number of queries, some of which may take more than 10 minutes to complete.
Number of setuid root files: 0
Number of directories with more than 100000 files: 37
Number of directories with 0 files (may have subdirectories): 49
Number and size of files > 100G: 88 files, 837228339 bytes
Number of directories named *delete*: 0
Number and size of mp3 files: 0 files, 0 bytes
Number and size of jpg files: 360 files, 16476162 bytes
Number of 0 byte files: 334
Number and size of files last modified before 2011: 380 files, 70644256 bytes
Number and size of files last accessed more than 3 years ago: 0 files, 0 bytes
Number of files that have permissions matching /o+w: 873
EOM
[root@osboxes sample-file-system]# sfdu -hs /vol1/experiment1
245.9GB /vol1/experiment1
```

Zones

The “zones” feature is designed to allow non-superusers to access some Starfish features. We call those users “ZoneAdmins”, and they are users between a standard user and a superuser. Examples include primary investigators, group leads, department heads, departmental IT staff, data wranglers and so on. Many sites use dozens of zones and at least one has more than 100 zones.

A zone is defined by its name, directories added to the zone, and users and / or groups added to the zone (making them ZoneAdmins). Each zone can have different ZoneAdmins. A zone can also have a unique set of tags, created and managed by the superusers, available to the zone’s ZoneAdmins for them to apply to files, directories or directory trees. Optionally, one or more (or all) zoneadmins in a zone can be given the right to recover files from backup or archive - allowing self service and the reassurance that they can bring back their data at any time from other tiers of storage. Zoneadmins have no other important actions available to them - they can only apply tags. Those tags can be turned into actions by the site’s superusers, who can create jobs to take actions based on those tags.

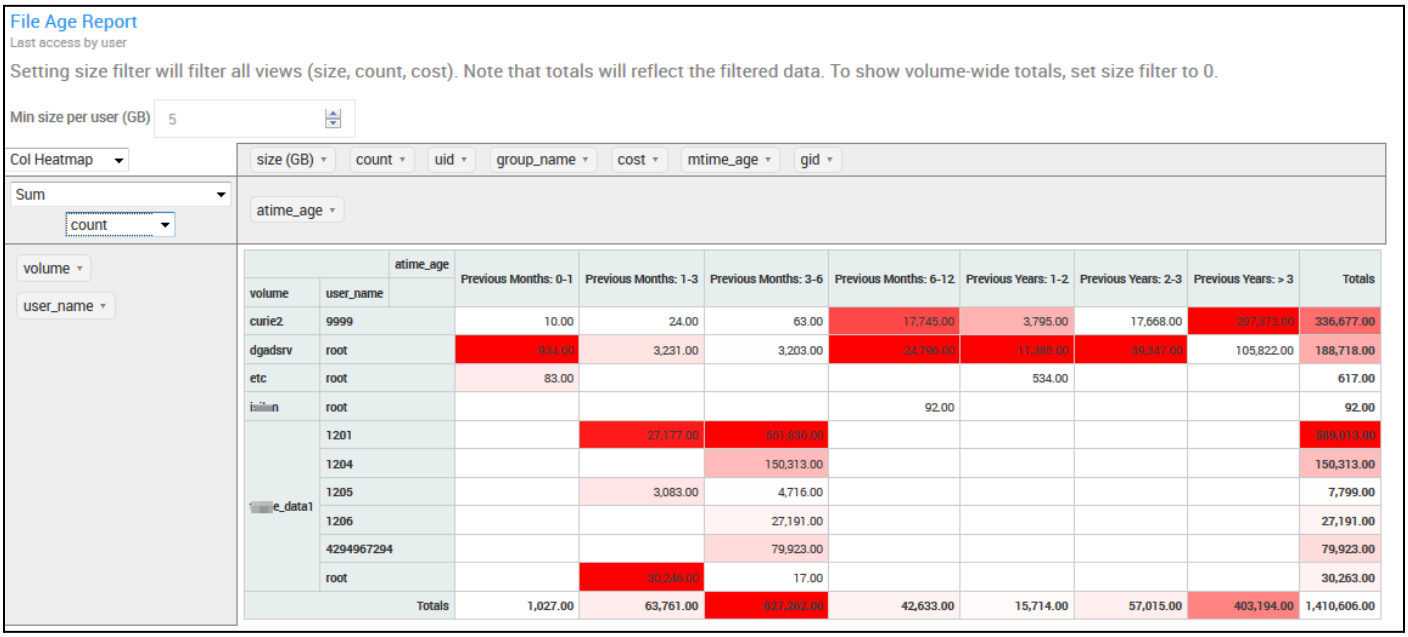
In summary, a zone contains a subset (or the full set) of the site’s directory trees, and when ZoneAdmins login to the GUI they only see their zones and the zones’ directory trees(s). ZoneAdmins can apply and remove tags that have been created for their zone by the superusers. Zones can also be created, managed, and used at the CLI level, providing the same feature set described here. ZoneAdmins can use the full power of the GUI to search, analyze, and visualize information about that zone but some commands aren’t available to them (such as most actions, and most reporting features that would show them information from beyond their zone).



The Ensemble

Together, these components provide the fastest, most complete set of tools for monitoring and managing institutional-scale data sets.

Starfish resource consumption scales from minimal, for a small installation, to moderate, for facilities with large data needs. All tasks and cataloging can be scaled out across multiple servers and agents to distribute and parallelize the workload. Due to the efficiency of the database, scanning and jobs engines, the resource overhead and hardware needs are minimized. For example, one Starfish customer is managing over 20PB of data with 9 billion files with a single server running a single Starfish instance. Current Starfish-managed sites scale to greater than 85PB and 17 billion files. The rate of scanning of files for tracking in the metadata database has reached 2.8 billion objects per day at one site.



volume	user_name	atime_age	Previous Months: 0-1	Previous Months: 1-3	Previous Months: 3-6	Previous Months: 6-12	Previous Years: 1-2	Previous Years: 2-3	Previous Years: > 3	Totals
curie2	9999		10.00	24.00	63.00	17,745.00	3,795.00	17,668.00	297,372.00	336,677.00
dgadsrv	root		934.00	3,231.00	3,203.00	24,796.00	11,385.00	39,347.00	105,822.00	188,718.00
etc	root		83.00				534.00			617.00
[illegible]n	root					92.00				92.00
[illegible]e_data1	1201			27,177.00	561,836.00					589,013.00
	1204				150,313.00					150,313.00
	1205			3,083.00	4,716.00					7,799.00
	1206				27,191.00					27,191.00
	4294967294				79,923.00					79,923.00
	root			30,246.00	17.00					30,263.00
Totals			1,027.00	63,761.00	827,262.00	42,633.00	15,714.00	57,015.00	403,194.00	1,410,606.00

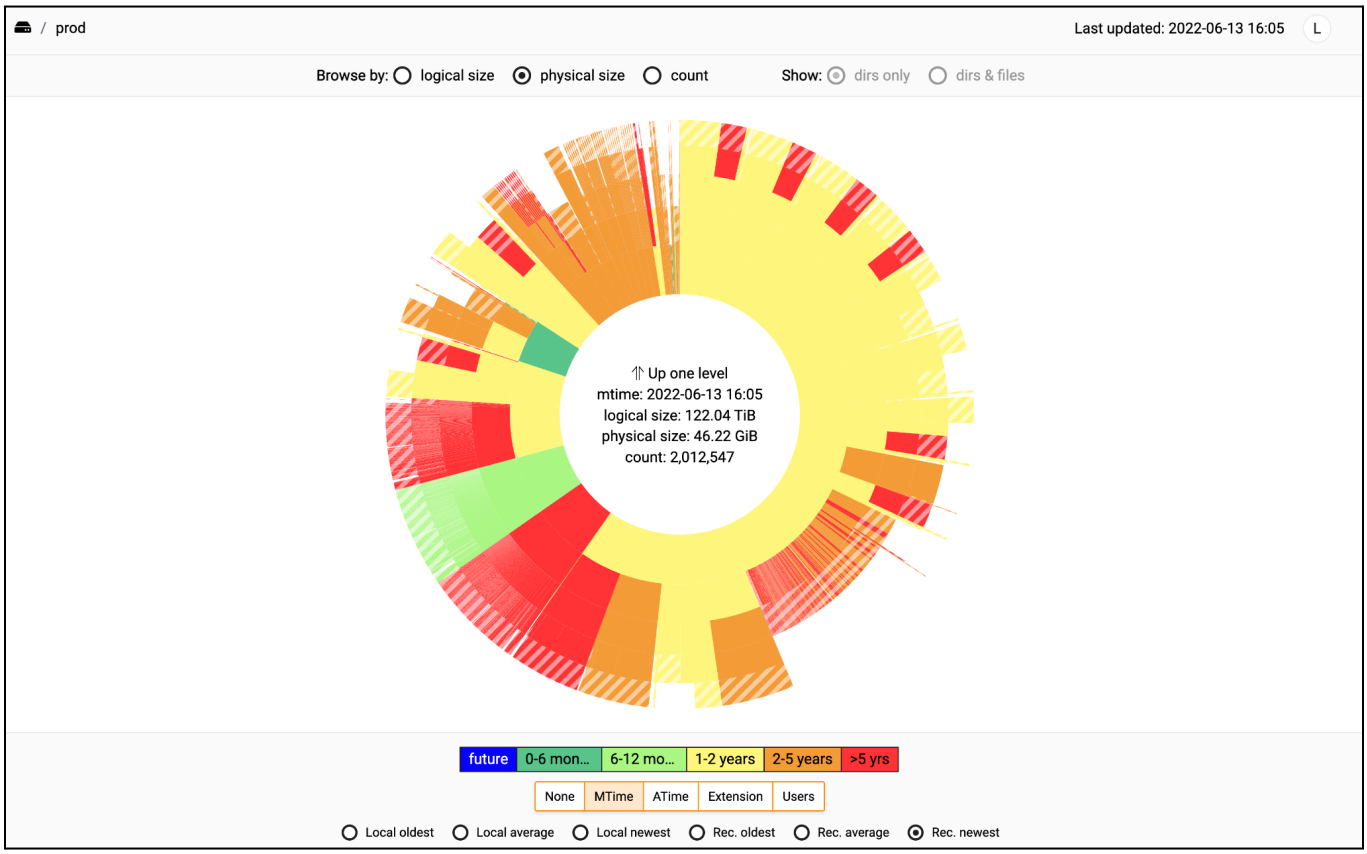
There are many reasons to appreciate Starfish for its fast, informative reporting features. For many organizations, there has been an inability to perform actions against their data without these reporting capabilities. Starfish actions are based on sophisticated metadata queries. Imagine using Starfish to tag a subset of billions of files as low priority when they are created. This "low priority subset" then goes unaccessed for three months which triggers Starfish to move the files to archive, a lower-cost tier, or the cloud. Built-in actions such as archive/restore can be augmented by arbitrary actions implemented via programs or scripts and run via the job scheduler.

Starfish includes a CLI, APIs, and a GUI. All three components provide access to the catalog and the ability to create and manage jobs. The GUI is especially useful for visualizing the state and history of volumes, files, and summary information.

By adding intelligence to the data, Starfish enhances and enables data management at petabyte scale

What Can STARFISH Do for You?

- Starfish installs in as little as 15 minutes, and can provide you with useful, actionable information about all of your files within hours.
- Starfish works with heterogeneous storage, including on premises file stores (Linux and Windows), object stores, and cloud stores.
- Starfish captures file metadata, tracks file history over time, and allows you to add programmatic or ad hoc metadata.
- You can then run pre-built or custom queries against the metadata, via CLI, GUI, or API, to explore the data or create jobs to take actions based on the metadata, such as migration, archive/recovery, heterogeneous replication, research data pipeline automation, and backup/restore, or any action that is scriptable.
- Starfish provides executive and detailed reports via its integrated redash dashboards.
- Starfish can integrate with third-party tools such as ElasticSearch, BA Insight, and RSpace, with extensibility and simple APIs to add additional tools based on customer demand.
- With all of this file metadata in the Starfish database, you can add even more insightful queries quickly and easily.



Starfish is deployed in many verticals and environments, including HPC, business management, media and entertainment, oil & gas, government, education and life science

25 surprising things you can do with Starfish

1. **COMPARE TWO DIRECTORY TREES**
   Compare two directory trees (even big ones) highlighting differences between the two.
2. **CREATE "DEEP ARCHIVE TIERS"**
   Easily take advantage of "deep archive" tiers on cloud object stores. Copy files directly to and automate the retrieval process from the archive.
3. **MIGRATE TO THE CLOUD PAINLESSLY**
   Estimate storage requirements and costs for cloud migrations. Identify and quantify files that can be left behind or stored in lower classes of service.
4. **TAKE THE "MEH" OUT OF METADATA**
   Extract metadata from file headers (office, rich media, scientific. etc.) and turn these key-value pairs into discoverable file properties.
5. **RECLAIM STORAGE SPACE**
   Find uncompressed files and compress them to reclaim storage space.
6. **REMOVE DUPLICATE FILES**
   Detect and remediate duplicate files, even across different file systems and object stores.
7. **RUN SCRIPTS IN PARALLEL**
   Enable programs and scripts that were not designed for scale to run in parallel across billions of files.
8. **REMEDIATE FILES**
   Identify directories with unknown foreign (non-UTF8) characters in file names for remediation.
9. **SEARCH FILE HISTORY**
   Search back in time for files that might have been moved or deleted.
10. **REPLICATE MASSIVE NAS SYSTEMS EASILY**
    Replicate massive NAS systems to third-party devices (even object stores) without depending on built-in snapshot and replication utilities.
11. **MIGRATE FILES AT UNPRECENDENTED SPEEDS**
    Migrate files to new storage devices at unprecedented speed with true-ups, nuanced prioritization, hash verification, ACL support, and WORM support.
12. **POWER YOUR REPORTS**
    Use tagging to create project-specific dashboards and reports across your entire storage environment.
13. **DISCOVER AND MANAGE PII**
    Discover and manage files that contain PII (personally identifiable information), confidentiality markers, export control restrictions, or other sensitive data.
14. **RE-ASSIGN FILES AND DIRECTORIES**
    Find files and directories that are owned by people who have left your organization and re-assign ownership.
15. **MANAGE AND FIX PERMISSIONS**
    Identify and fix anomalies in file system permissions.
16. **AUTOMATE WORKFLOWS**
    Automate data-driven workflows for files coming from data acquisition devices and laboratory instruments.
17. **DISCOVER INACTIVE DATA**
    Discover the top-most directories of file system trees where no subtree has active data in a given time period.
18. **MONITOR FILE SYSTEM CHURN**
    Easily monitor file system churn and reveal who has been consuming all of the free storage space.
19. **DISCOVER AND DELETE STATIC COLLECTIONS**
    Discover files that come from public data sets or static collections and mark them as candidates for local deletion.
20. **GET RID OF "ROT"**
    Give the users simple tools to enable them to delete and archive data (redundant, obsolete, and trivial (ROT) files.
21. **AUTOMATE CLOUD BURSTING**
    Automate cloud bursting for HPC and AI jobs where data is primarily stored on premises.
22. **SAVE MONEY WITH COMPRESSION**
    Transparently compress during archiving and decompress on restore, saving money on Cloud transport and storage costs.
23. **ENCRYPT, AUDIT, AND REMEDIATE**
    Reveal encrypted versus unencrypted files and take remedial actions. Clean up from ransomware or audit for encryption.
24. **ELIMINATE NDMP AND NAS-TO-NAS REPLICATION**
    Enable file-based backup and restore with versions and point-in-time. Eliminate the dependence on NDMP or NAS-to-NAS replication.
25. **PARALLELIZE DATA MOVEMENT OF SMALL FILES**
    Enable massively parallelized data movement of small files to S3 object stores and edge devices (e.g., Amazon Snowball).

What Clients have to Say

"Starfish is one of the best engineered software packages that we've come across." – Very large national health protection agency

"Did 5 or 6 migrations in the past year with Starfish: fantastic, easy, quick, wouldn't want to do it any other way." – US Gov't Dept of Energy Lab

"Found that we have DMP (crash) files all over the place. Would take us forever to clean them all up using a script. Much easier to search in Starfish, tag and delete em. Problem solved. Effort, almost none." – Major US Gov't health agency

"Starfish did something in 3 clicks that took 2 hours where your competitor spent weeks on it with several people and couldn't get it to finish in 14 hours." – A top 10 hedge fund

"It actually does what Jacob said it would." – Multinational biotech company

"Starfish paid for itself in two weeks." – Media & Entertainment company

"I don't know the answer, but if I had Starfish I would." – Kansas federal bank

"If I'd had Starfish I could have right-sized our backup facility." – Genomic research institute

"I think I'm in love. It has everything I want in it." – A leading medical institute

"The cadence of releases you're able to push out is amazing." – Major aerospace company

"This is the future." – Major research university

"Starfish is 100x better than our previous solution." – A leading children's hospital

"We are using Starfish to fish for stars." – Research institute using Starfish to manage astronomy data

"Starfish dominates its space." – Leading animation studio

"Jacob is a genius!" – VP of IT and Security at a leading medical center

"I can run a du on a PB of data in 5 seconds, just brilliant!" – System Architect, Top 5 Biopharma company

"Our growth rate went from 10% per year to flat because of Starfish." – Leading medical school

"I'm syncing entire file systems with starfish, and it is awesome." – US Gov't Dept of Energy Lab

"The support is excellent!" – Ivy league university

"Managing data is one of our users' biggest pain points. And on the operational side, lack of insight into how our file systems are being used really impacts our ability to manage our storage resources effectively. Starfish allows us to offload data management tasks from the file system into a database that has been optimized for this purpose. We are already seeing operational benefits and are looking forward to sharing Starfish capabilities with our users." – Robin Goldstone, I/O Solutions Architect, LLNL

Starfish was built for Research Data Management

Starfish Key Features

- Designed for extreme sites (billions of files, thousands of changes per second) –But sites with millions of files love it too
- Scales to the cloud, scales horizontally for huge migration, syncing, archiving throughput
- Highly efficient – does with one server what others need many to do, Lightweight – a single server can manage billions of files
- Bi-weekly release cycle
- Community forum for discussions
- Powerful / programmable / automate-able
- Search, discover, analyze, report, (ls, find, du killer) and take actions – ours or yours
- Useable by storage administrators, data creators, project leads
- By geeks, for geeks -> CLI, REST API, GUI, GUI teaches REST and CLI commands
- Add metadata manually or automatically, and it's your metadata, not ours
- No lock-in - out of band so safe to install and remove, no proprietary formats, no latency or points of failure added to the data path
- Installs in minutes, easy to test
- Runs on popular Linux distros
- Remote agents can offload scanning and jobs to distributed hosts
- Scan accelerators for Lustre, Qumulo, PowerScale (Isilon), GPFS, and HPSS
- All POSIX-like file systems supported via scanning over a mount point
- Windows file systems supported
- Very fast & smart metadata ingest and jobs actions
- CLI, HTML 5 GUI, and full REST API interfaces
- Integration with enterprise tools such as BA Insight, Elasticsearch, and RSpace
- Zones allow non-superusers to analyze their storage use and participate in its management

Take the *meh* out of your metadata and use it for data-informed decisions and actions.

To learn more about how Starfish Storage can help solve your business and IT challenges, contact us at info@starfishstorage.com or visit http://www.starfishstorage.com/.