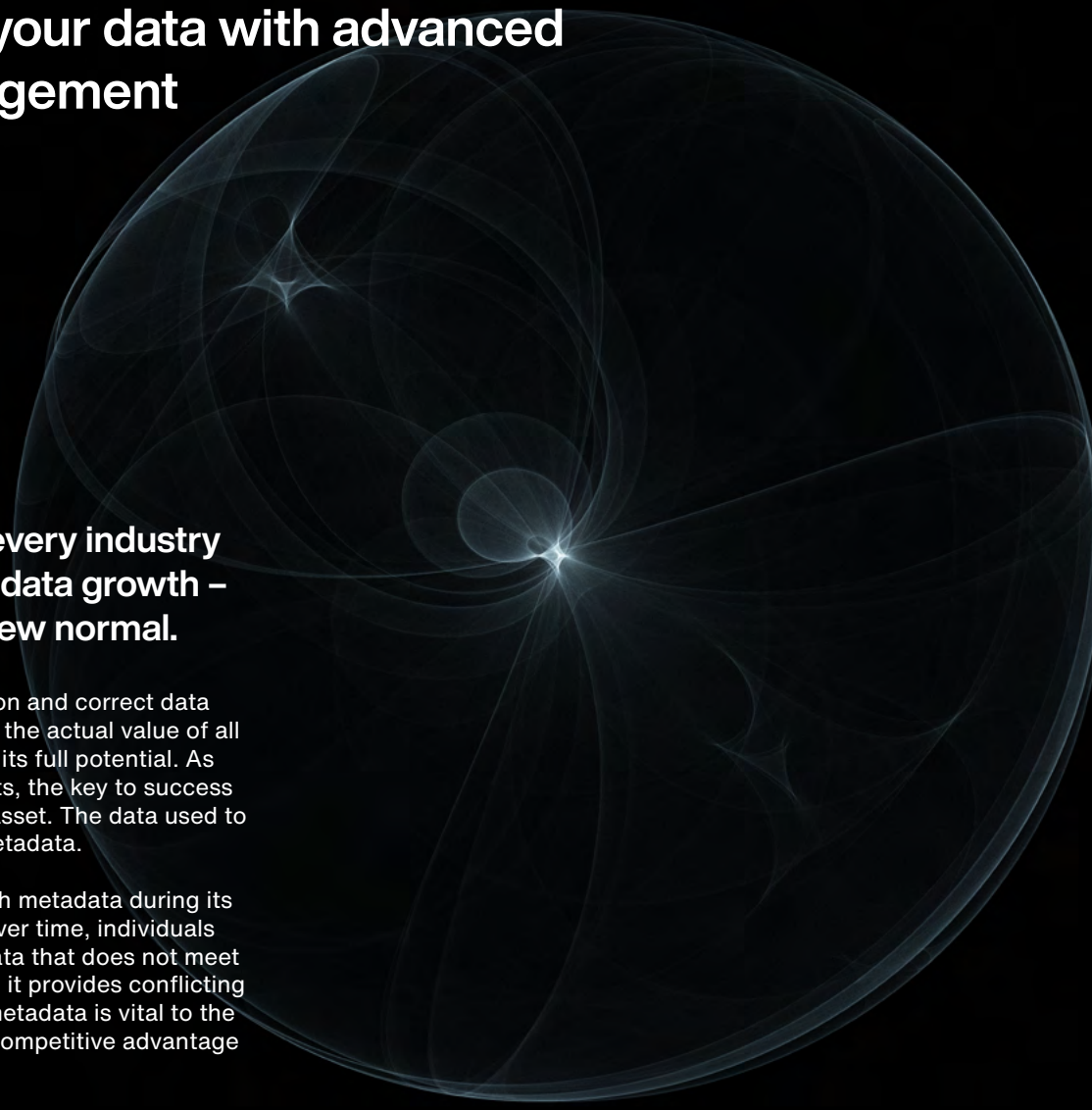
Get more from your data with advanced metadata management



Organizations across every industry are facing exponential data growth – petabyte scale is the new normal.

But without proper data acquisition and correct data management across its life cycle, the actual value of all this data is not always realized to its full potential. As with managing any valuable assets, the key to success is having information about that asset. The data used to manage and use data is called metadata.

Suppose data is not managed with metadata during its lifecycle, then or what happens over time, individuals will create silos of inconsistent data that does not meet the organization's needs because it provides conflicting or confusing information. Thus, metadata is vital to the administration, productivity and competitive advantage of an organization.

This technical brief discusses metadata, what it is, how best to manage it and how its efficient use can reduce costs and increase revenue through the optimal management of data through its lifecycle.

www.arcitecta.com | talk@arcitecta.com

What is metadata?

Simply put, metadata is data about other data. Typically, metadata is defined as either system-generated or user-generated. A great example of this is a modern digital photograph from a smartphone. System metadata might include fields, commonly called tags, for items such as the type of camera used, F-stop, shutter speed, the time, longitude and latitude when the photographer took the photo, and so forth. User-defined data typically requires a human's input or intelligent algorithms that can learn from humans and automate some of these steps. For example, user-defined metadata about a photo might include tags that identify the people in the image or whether the image has been saved as a favorite.

Why enterprises need to manage their data via metadata

Data must address specific organizational needs to achieve strategic goals and generate real value. To consider data an asset, it must have metadata describing various facets of the data attached because this is what unlocks the usefulness of data and improves its usability throughout its life cycle. As with any organizational asset, the more valuable it is, the more critical it is to manage. Therefore, key data sets should have rich metadata attached to reflect the importance of finding and managing it in the future.

Harvesting system-generated metadata from files

Most files have a published MIME (Multi-Purpose Internet Mail Extensions) type that various programs use to read and access the file. It is possible to use the MIME to parse the file for metadata. The operating system (OS) also keeps some of this metadata, often called inode information depending on the OS. These system attributes include items such as the files last modified time, the size of the file, etc. Metadata derived from the MIME type or the operating system is system-generated metadata. Rudimentary systems often use this information to manage files.

Attaching user-generated or custom system data to files

In addition to system-generated metadata, custom or user-defined metadata can also be added to files to improve data management. For example, at the enterprise level, a system administrator might be interested in what lab or machine operator scanned an image into the system. In the defense arena, a user might want to know how many tanks are in an image. A user could also create a custom lexicon to make searches easier so that if someone is looking through a list of dog types, "Labrador" = "LABRADOR" = "labrador".

There are many ways organizations can add user-defined metadata to their data. Typically, it would happen through a portal where information about a user's data is collected when the file is created or goes through a specific process within the system. Many enterprises have custom LIMS (Laboratory Information Management Systems) or algorithms or processes that add this type of metadata. Other times it is extracted from an associated file in a known format, such as YAML. In any case, the data's related metadata is parsed and kept.

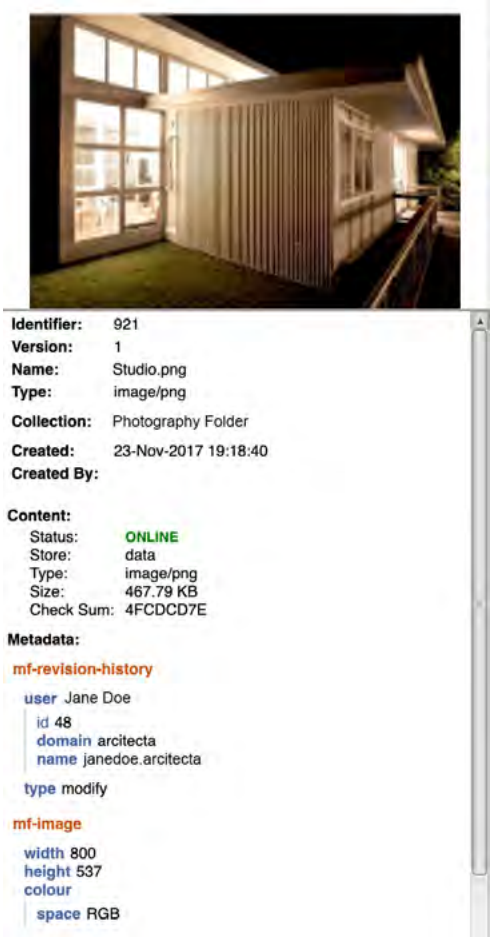


Figure 1: System generated metadata collected from a picture of a shed.

Considering the metadata lifecycle

For both system and user-generated metadata, it is crucial to consider where it is stored. For example, system-generated metadata may embed photo information in a .jpg file. This process works well enough with a smaller number of files because there is no danger of the metadata and the contents of the files getting separated. However, at the enterprise level, the number of files can easily reach millions if not billions of files, so a more robust and scalable architecture is required.

At these scales, metadata should be managed externally to the file system. Having user and system-generated metadata versioned separately from the content in the file allows the owner to better extract value from legacy content as future uses and new analyzers are developed. For example, a photo might have a metadata tag for specific people in the image. Years later, the geographic location of the image becomes of interest, so a new metadata tag is attached to the file. In doing so, a file's metadata can evolve whilst consuming the smallest possible footprint, which is only possible if the metadata can be versioned independently of the underlying content data.

Although unattainable with any relational or standalone document database, this process is intuitive and well documented. Anyone who's visited a library in the past hundred years will have used a very successful analog equivalent – the Dewey Decimal Classification system. The Dewey index cards commonly contain metadata about a book. The index cards are held in a separate store, and the referenced content, the book itself, can be at one or more of several sites. The index card helps users search for and retrieve the content. The Dewey Decimal system has also been successfully digitized, making a similar metadata system for an organization's data holdings seem like a precedent choice. Figure 2 shows a a hybrid model using Mediaflux's XODB, an object database where everything exists as related objects and where the object "state" is persisted as compact XML.

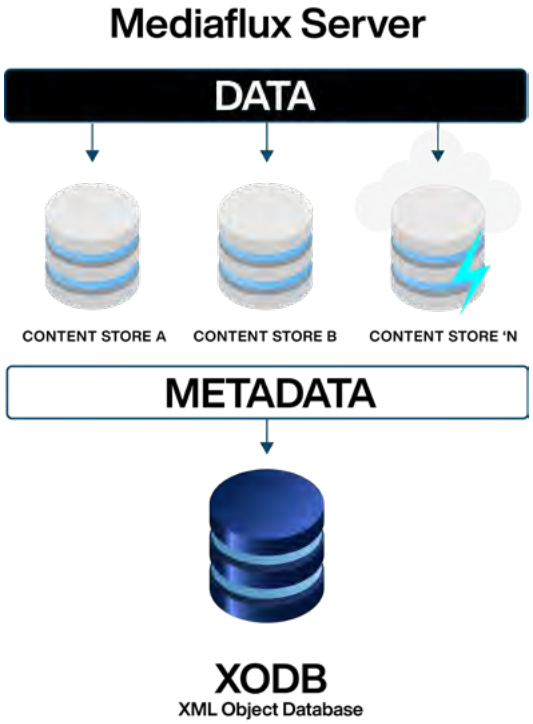


Figure 2: XODB bridges across any data type, on any hard- ware or storage system

The power of metadata for data management

In the same way the Dewy Decimal System revolutionized library cataloging, applying metadata management capabilities using a separate metadata database can significantly impact how organizations deal with sprawling data architecture. Here are some of the most powerful ways an enterprise can ramp up data efforts whilst optimizing data costs by holding metadata separate from the data itself.

Eliminate data silos

When metadata resides outside of the file itself, it exists as its own persistent, unified data repository making it easy to search across all data, regardless of where it exists. Additionally, metadata can drive all the storage, security and governance protocols for the underlying content data, which can even be distributed across the world in separate storage repositories at multiple sites.

Eliminating data silos enables rapid, ad-hoc search across the entirety of data. For example, from a local PC, any authorized researcher can search all data recently created and currently residing on the fast first tier of storage; the less active data on slower, less expensive storage; the long-term archive in another location, potentially the cloud; and even data residing in deep archives (such as on tape) in a data warehouse.

Know what data you have

The ease and speed at which users can find files of interest correlate to the continuity of operations. For example, suppose an employee leaves the organization or is out on a leave of absence. In that case, an organization can remain productive because it can locate data using metadata, allowing reuse throughout the lifecycle of the data asset.

As a second case for how metadata supports productivity, consider a hypothetical researcher investigating a particular form of cancer. Before developing a series of experiments, the researcher searches all the available metadata – not just data they have created, but all data across the entire organization related to this particular type of cancer cell. This research is on the cutting edge, so it's unlikely they find the exact experimental data they were planning to develop. However, perhaps the researcher finds a significant amount of related data, which they can then review to refine the design of the new experiment. Or perhaps after reviewing it, the researcher finds that while the data isn't exactly what they were looking for, it does provide additional information on the cells they were planning to target. Say it refutes the new hypothesis, the researcher may change course to pursue a different and more promising line of research. This ability to search across all existing data in an organization not only reduces time to market by enabling researchers to contextualize better and refine their research, but it can also potentially reduce time spent pursuing less optimal paths and can help to avoid accidental duplication of research.

Flexible security and governance

Applying security or governance protocols also becomes more flexible when metadata is stored separately from data. For example, suppose specific data are subject to compliance, such as Health Insurance Portability and Accountability Act (HIPAA) protocols. In that case, some researchers may not have access to the files themselves. Still, they could potentially see that those files exist and what kind of data is available via metadata without needing to access the files themselves or view any associated metadata restricted by protocols. Or, in the case of 'black operations', only a few users would be able to see any metadata associated with 'black files'. It would be as though these files did not exist for users who lack the appropriate credentials.

Keeping ahead of evolving data requirements

Managing data is not a static event. Data structures and metadata evolve as infrastructures grow, new use cases arise, and new technology becomes available. The perfect infrastructures and metadata today might not be ideal in two years, ten years or 20 years from now. Metadata management should be designed with this reality in mind. Ensuring a file's metadata can evolve independent of the underlying content data, and the metadata is fully portable and compatible with open standard environments, an organization can bridge across any data type on any hardware or storage system for the life of its data.

Optimize storage costs

Metadata also significantly enhances the effectiveness of data tiering and archiving strategies, compared to data-tiering based strictly upon basic filesystem attributes.

Continuing the library analogy above, imagine a sizeable inner-city library where space is premium but still aims to provide users with the best quality of service. It might sensibly keep the most frequently used books in the most accessible place. They might best store esoteric journals accessed less regularly off-site in a warehouse where they are still accessible, but it takes some time and effort to get the desired volume. The library would consider this a wise choice because it is cheaper to store those journals in the warehouse than in the downtown library.

Similarly crucial in the digital domain, the underlying storage tiers are often characterized based on their 'quality of service', measured with attributes such as access speeds, cost, reliability, and so on. When tiering data based on user-generated metadata, the file's content is easily migrated to the lowest tier of active media or even comfortably pushed to deep archive or tape, or even data destruction. Regardless of the location of the file contents, the file would still appear in metadata searches and be useful to the organization, even at the lower cost.

Use and Reuse Best Practice - FAIR Data Principles

Metadata is at the core of the 'FAIR Guiding Principles for scientific data management and stewardship' published in 2016 in Scientific Data. The authors intended to provide best-practice guidelines to improve data's Findability, Accessibility, Interoperability, and Reuse to gain the maximum potential from data assets.

Findable

The first step in (re)using data is for a user to be able to find what they are looking for. Because humans increasingly rely on computational support to deal with data due to the increase in volume, complexity, and creation speed of data, metadata and data should be easy to find for both humans and computers.

Accessible

Once the user finds the required data, they need to know how to access it, possibly including authentication and authorization. As discussed earlier, when metadata is stored separately from the content itself, organizations can use metadata to apply and update security or governance protocols as requirements or user roles change.

Interoperable

Metadata provides structured information that defines and describes data. It plays an essential role in ensuring users and systems understand the meaning of exchanged information when integrating with other data or when it needs to interoperate with applications or workflows for analysis, storage, and processing.

Reusable

The greater the ability to share and reuse an organization's data, the more valuable the data becomes to an organization. Rich descriptive metadata adds to the value and longevity of data by optimizing the reuse of data, which is the ultimate goal of FAIR.

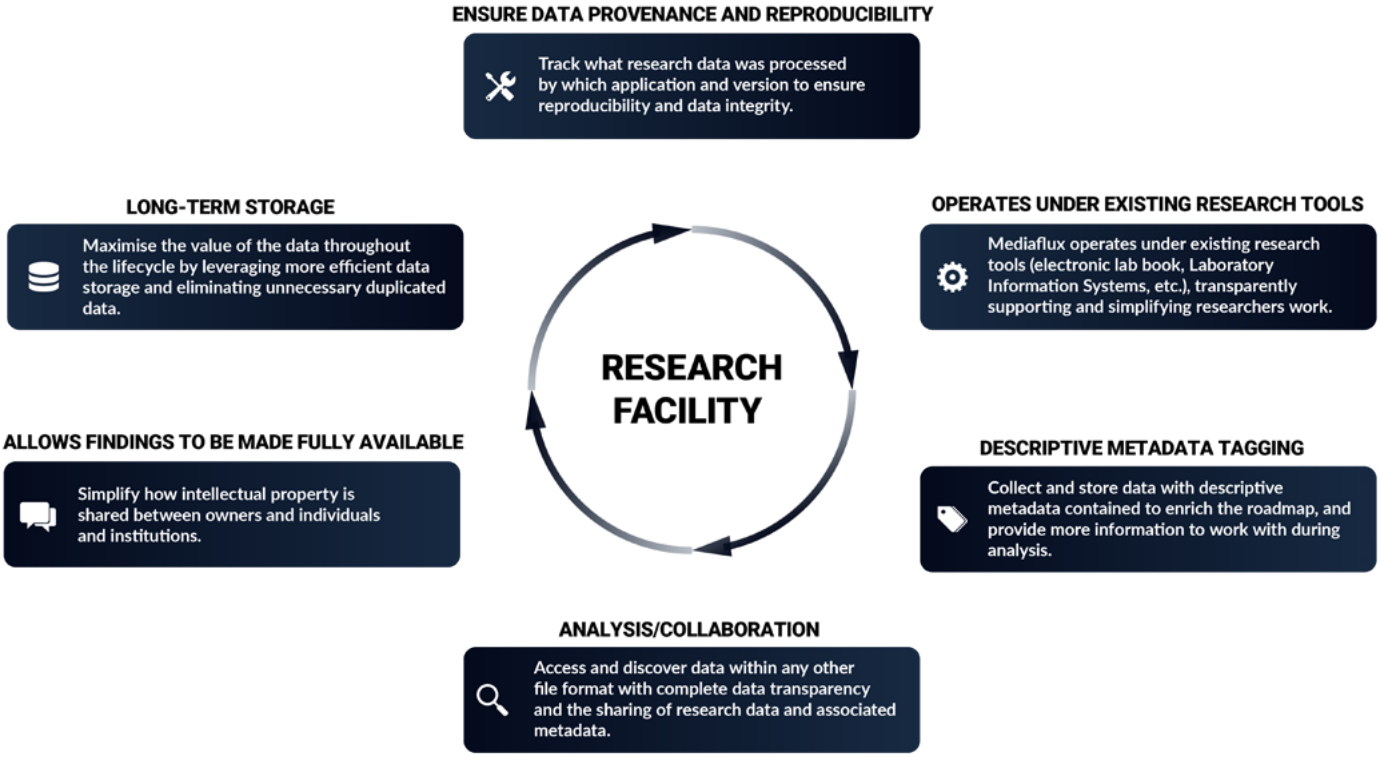
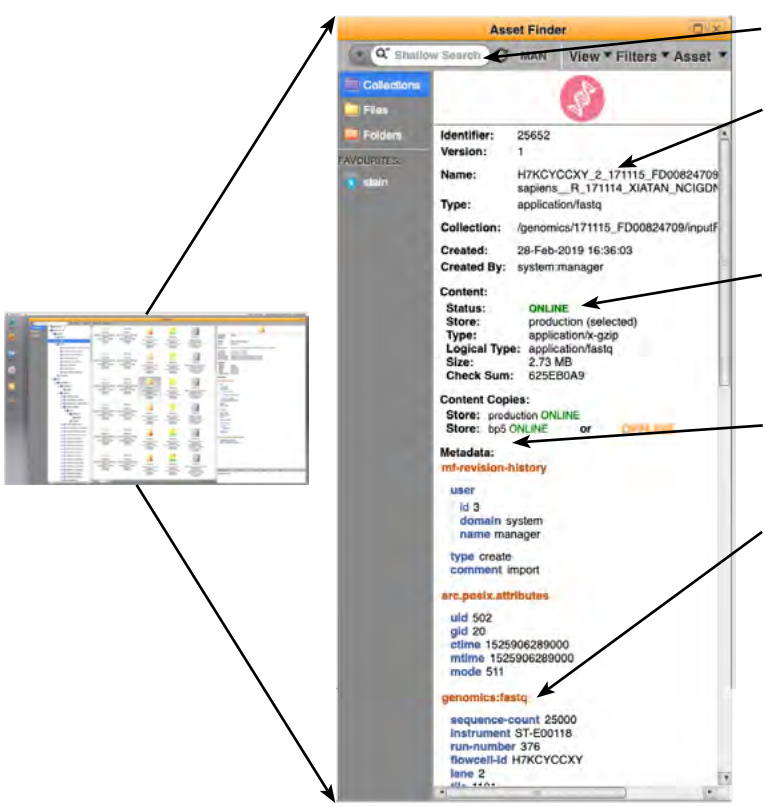


Figure 3: Mediaflux ensures research data meets FAIR guidelines and supports publicly funded Open Science research data availability requirements.

The case against managing data via metadata

The biggest objection to data management via meta- data is the perceived labor of creating the metadata in the first place. Data managers argue that end-users will refuse to curate their files because it takes too much time and effort. This objection would be valid if each user had to individually apply the description, security, governance, and project codes to each file, which would undoubtedly be a demanding task.

However, data management platforms can automate the majority of this work. For example, using metadata scraping, a workflow could be as simple as creating a project directory and ensuring each file and subfolder inherits the project's metadata tags. Then, the user would only need to add metadata specific to that file. In the case of machine-generated data, the machine itself might provide much of that metadata in combination with folder-inherited metadata fragmwwents. Thus, the automated workflow could transform what used to be the most obscure, opaque data in the organization into the most transparently documented.



File Browser GUI Asset Finder searches metadata fields of interest to filter and create compound queries

File system level information (metadata) such as: Name, Location, Creation Date, etc.

Content information, checksum, size and data accessibility
ONLINE – content is available on nearline storage
OFFLINE – content is on deep storage (on BlackPearl Tape)
ONLINE+OFFLINE – content is available on both
MISSING – content is missing

Revision history, audit trails, provenance and reproducibility

Descriptive metadata

- Extracted during the ingest process
- Custom-built scraper that extracts specific metadata based on the data type. These plug-in content analyzers extract image type and resolution, scanner instrument information, project, study, population, PI, etc.
- User generated metadata such as: annotations, labels, tags, comments, and workflow-specific actions

Figure 4: Example of data with attached metadata viewed through the Mediaflux GUI

Advanced metadata driven data management with Mediaflux

Mediaflux has been developed and refined over multiple decades of deployments. Its metadata capabilities reflect Arcitecta's deep experience with diverse forms of data in a wide range of environments, from simple to incredibly demanding. Embedded at the heart of Mediaflux is XODB, the powerful binary-XML object database engine that holds metadata in a database separate from underlying content data.

XODB removes context switching and network overhead between the application to function as a single executable process. As a result, XODB offers immense power and flexibility to metadata and data management unattainable with any relational or standalone document database. XODB's footprint is also much smaller, so organizations can keep it on faster storage and more easily share metadata across an enterprise. In addition, XODB can be rapidly evaluated to increase performance and easily reinflated into the original XML to ensure portability. To learn more about XODB, download the brochure here.

Key XODB features and benefits

- Clusters all data for an object in a compressed binary XML format.
- Allows incremental, on-demand, schema evolution.
- Controls data accessibility for improved authentication and authorization, compliance and classification.
- Supports federation and replication.
- Supports partial query evaluation.
- Supports complex XPath expressions.
- Supports custom metadata through an extensible framework of services.
- Provides high levels of concurrency.
- Supports billions of assets per instance.
- May be used as a single node, or multiple instances clustered for extreme scalability.
- Written entirely in Java and does not require the installation of additional database software.
- Requires no administration except periodic backup.