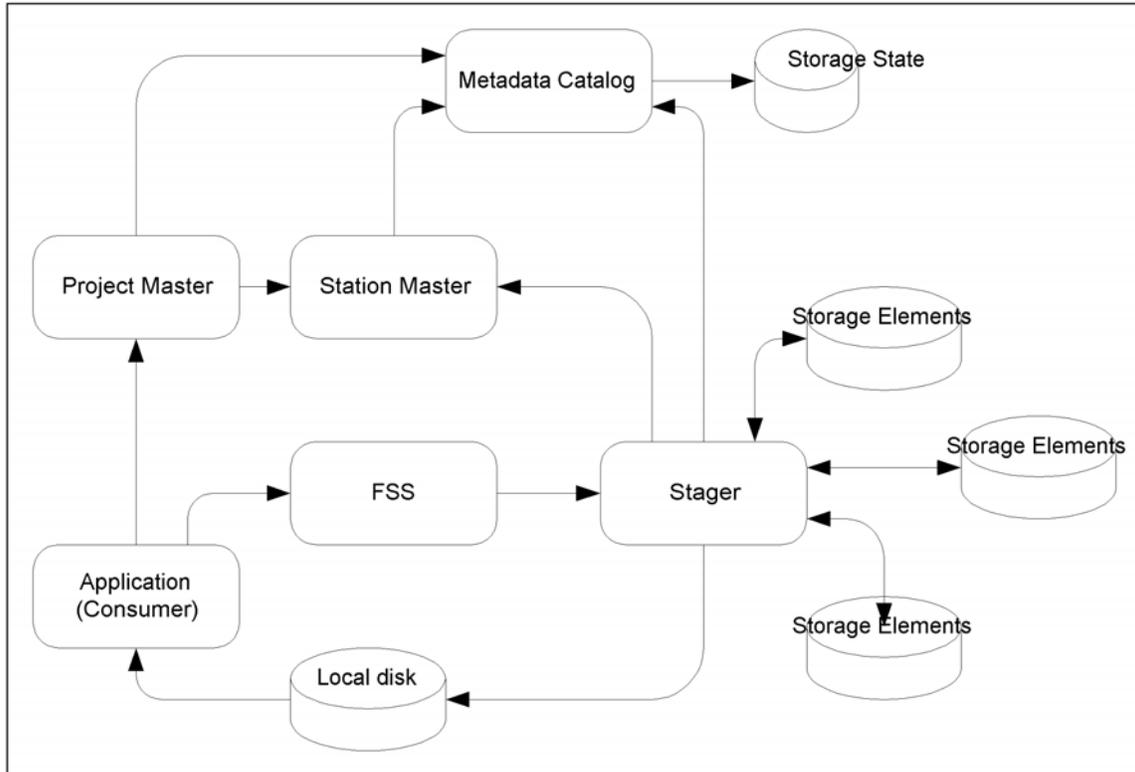


SAM Notes

Andrei Baranovsky, Igor Mandrichenko

High Level Design Overview



Station Master

Manages operation of SAM instance. It optimizes data movement, storage and delivery to the application within application and storage element constraints and with the goal of most efficient data processing.

- Stores state information about all components into metadata catalog
- Station Master uses Stager as an interface to storage elements
- Station Master can communicate with Station Master of another SAM Station for inter-station data transfers

SAM Cache is a component of Station Master, which is responsible for data movement within the station. It uses Metadata Catalog and Stager.

Station Master is a process, one per SAM Station. It is a CORBA server.

Metadata Catalog

Metadata Catalog has broader functionality than its name implies. It is a persistent data storage of state information for all SAM Station components. Metadata Catalog provides access to the data through high-level interface called Metadata Catalog Server.

Data item information consists of:

- Metadata
- Physical Location(s) – related to storage state

Metadata Catalog maintains snapshot of reasonably accurate representation of data storage state. This representation is used by the Station Master to optimize data access. In cases when the state is not correct, the errors are expected to be non-fatal and results of the optimization close enough to the optimal.

CDF and D0 Metadata Catalog use Oracle DBMS as the data storage. Currently D0 Catalog is ~10-100M records.

Metadata Catalog – CORBA Server, in front of Oracle Server.

Stager

Stager is an abstract interface to specific Storage Element. Stager represents various specific storage elements in more or less common way. It provides basic data access and namespace management functionality:

- Put
- List
- Delete

Stager is a CORBA process which works as a server for the application and as a client to the Station. Multiplicity of stagers is determined by the station configuration. Minimal multiplicity is one stager per storage element. In particular, there is one stager per farm (cluster) node.

Consumer

Consumer is the application process that processes the data. Consumer consumes and/or produces data in units of single file.

Consumer communicates with Project Master, Station Master (to start project). It may request Project Master to start.

Consumer-to-SAM interface consists of:

- Set of shell level commands
- CORBA interface specifications in form of IDL files

- C++ API based on CORBA interface

FSS – File Storage Server

FSS is data movement utility not associated with any project or application. It can move large amounts of data between Storage Elements in background mode. The functionality is similar to RFT. It updates Metadata Catalog.

Project Master

Essentially Project maintains list of data files to be processed. Project Master coordinates delivery of data associated with specific project. It distributes input data files among individual consumers. Project Master communicates with Station Master to initiate data movement and stores its state and state of application consumers in the Metadata Catalog.

CORBA Server. There is one process per project. Created by Station Master when the project is created. Currently Project Master does not start automatically. They can be started on demand by the application. There is a possibility of a race condition if there are too many concurrent requests to start the same Project Master. There is no reliable way to detect and recover from failure and re-start of a Project Master, mainly because while it is down, some state information may be lost.

Typical project lasts not more than couple days for many practical reasons.

Storage Element

Storage Element is used for temporary and permanent data storage. Station Master (SAM Cache) manages Storage Elements attached to the SAM Station. SAM uses Stager as a common interface to storage elements it works with. Stager interface seems to be general enough to accommodate wide variety of storage elements, but on the other hand not all storage-specific functions can be accesses through Stager.

Local disk on the node where the consumer runs is a special kind of storage element. SAM delivers data to local disk for the consumer to read. As with other storage elements, SAM keeps track of data stored in local disk storage.

In order to keep metadata catalog up-to-date, all data transfers must be managed by the SAM Station. SAM assumes that it has complete control and accurate knowledge of the contents of storage elements attached to the SAM Station.

Metadata Catalog recognizes 3 states of a data item:

- Known, not stored
- Known, stored, not cached (slow access)
- Known, stored, cached (fast access)

For dCache, the assumption is that N last accessed files are still cached, others are stored, where N is configurable constant.

Analysis Notes

Database consistency

Because the mechanism of database transactions is not used, it is possible (although this may not be happening with any noticeable frequency in reality) for the database state to become inconsistent with the state of the data processing. Detection of or recovery from errors of this kind may not always be easy or efficient. Often it is easier to just reinitialize the project. Information supplied with immediate error message is not always sufficient for error detection and diagnostic.

CORBA

CORBA is sophisticated remote object access protocol. It is binary-based (as opposed to text-based protocols like XMLRPC and SOAP) that requires complex and not easily portable software library. It is not widely popular or accepted. Its relative unpopularity causes obvious problems:

- It is difficult to maintain portability of CORBA software across OS flavors and versions.
- There is no widely accepted security model that would provide necessary level of client authentication or data integrity verification.

Because CORBA protocol is binary, CORBA communication is difficult to debug. There are no built-in methods of detection of incompatibility between client and server versions, no interface publishing methods.

Scalability

Main source of non-scalability is communication between stagers and the Station Master. Number of stagers grows with the cluster size while Station Master remains single. Typically, for known installations maximal number of stagers per Station Master is ~300. For larger farms, they use multiple SAM stations per cluster.

Robustness

Most noticeable potential reliability issue is probably the fact that Project Master process must be running as long as the project is in progress. This is especially difficult to achieve in batch environment, especially on Grid, where it is not always possible to estimate when the project will complete or even when consumer batch processes will start. This means that Project Master processes must run on robust and well supported computers in stable environment, and it complicates OS and hardware maintenance procedures that require reboot of the computer or interruption of other services.