

Metrics Correlation and Analysis Service (MCAS)

Project Detailed Requirements Document

Authors:

MCAS Team

Table of Contents

Change Log.....	3
1 System definition.....	3
2 Requirements.....	3
2.1 Data Requirements.....	4
2.2 Data integration.....	4
2.3 Content Hosting.....	4
2.4 Content Aggregation.....	4
2.5 The “Drill down”.....	5
2.6 Data Warehouse.....	6
2.7 Operational Requirements.....	6
2.7.1 Security.....	6
2.7.2 Response time.....	6
2.7.3 Maintainability.....	7
2.7.4 Scalability.....	7
3 Sample Portal Display.....	8

Change Log

Version	Date	Change Description	Prepared By
V1.0	07/21/09	First Version	Andrew Baranovski
V1.1	07/22/09	Comment from Tanya	Andrew Baranovski
V1.2	07/29/09	Comment from GG	Andrew Baranovski

1 System definition

The mission of the MCAS (Metric Correlation and Analysis Service) project is to deliver a software solution to help with adaptation, management, and display of the data describing the state of applications on the grid. These data are used for monitoring and troubleshooting and are available from either application directly or supplied by the existing data collection and display middle-ware software.

A core business idea of this project is to factor out presentation and business analysis logic from available monitoring solutions into a standalone model, supporting common standards in data manipulation and presentation.

For information on project use cases refer to http://cd-docdb.fnal.gov/cgi-bin/RetrieveFile?docid=3267&version=3&filename=MCAS_requirements_lite.pdf

2 Requirements

- a) The product must offer a toolkit and a framework to design and deploy dynamic reports based on data produced or permanently stored by other applications. These reports should address use cases of visual data summarization, content aggregation and data drill down.

The product will contain the following applications and tools:

- b) **web frames(widgets)** – widgets will be part of the report composition. Widgets should be designed with the idea that their composition can reflect a report desired by a user. The key requirement is that widgets must be re-usable in different reporting application contents. The re usability of the widget implies existence of abstract API specific to that widget or data format that widget should automatically accept for rendering.
- c) **web hosting environment** – an application to host, co-display and configure a collection of web frames (widgets)
- d) **data integration framework** – a user friendly software tool-kit to define rules to aggregate, summarize and select data from disparate data sources (including other monitoring systems). These transformations will be adapting data to format and semantics acceptable by the web frame elements.

- e) **data warehouse** – a light weight service to temporarily store data used to build dynamic reports.

2.1 Data Requirements

- a) Adopted data format should require a minimum amount of user coding for the purpose of storage, retrieval, processing, and adaptation of the new data. For example: an XML data table format satisfies this requirement..
- b) The system should allow users to define the “data source” and describe structure of data participating in format transformation and display activities.

2.2 Data integration

- a) Support data sampling of configured data source.
- b) Support a business rules engine to generate data presentation format compliant with web frame input. Initially, business rules engine may only work with data stored at the warehouse.
- c) Provide intuitive and easy to use tools to setup data transformation work-flows that produce data format compliant with web frame input
- d) Notify portal administrators of errors occurring during the execution of rules by the engine.

2.3 Content Hosting

- a) The content of the dynamic report must be available via the Web.
- b) A web page containing the report must be protected in compliance with the access rights of a particular user(an entity with proven credentials).

2.4 Content Aggregation

- a) The content of the dynamic report web page must be composed of one or more re-usable frames. Each frame must be independently initialized with data describing some unique aspect of a particular system.
- b) A user must be able to add or remove frames to/from the dynamic report window.
- c) A user must be able to orchestrate the rendering of a collection of frames using a common set of parameters. In addition, the page itself (it's entire content) must be able to respond to parameters set in the URL of that page.
- d) A user must be able to edit a set of parameters common to a portal page in a separate frame or special portal admin window.
- e) A user must be able to open and edit preferences of each web frame.

Proposed list of common frame types:

f) Generic Table view frame

The purpose of the table view display is to summarize value-pair set array data using features of the spreadsheet. In its basic form, Table view should be parametrized by

1. maximum number of rows to display.
2. sorting column – a column with predefined name containing value that has been set according to priority function. The priority function must be defined by the user as part of query to the warehouse.

g) Generic graph rendering/image display frame

1. A user must be able to setup a Graph display frame. The graph display will reference list of image locations. Image data will be displayed in the table-like form. Number of columns in that table should be configurable.
2. Image data inside the table grid may be resized to fit the dashboard display. The original(normal size) image should be display after its corresponding thumbnail has been clicked by the user.
3. Be able to embed links in the plots.(see drill downs) For example, when clicking on a ganglia plot, we want to bring up the whole ganglia page (with the other links).
4. Be able to easily rearrange the layout of the plots. Ideally, we'd like to say where the plot goes in the grid. Most important (supported) feature is add / delete plots to a page.

h) Generic time series display frame

Support a display to render the value of a function applied over a sequence of documents. The resulting time series plot should display how the content of the document(or data element) is changing over time. This frame serves need of application users who wish to define their own views on metrics mined from the warehouse service. The warehouse service may be internal to the project, or it can be external deployment suitably adapted to conform to project's data format.

i) Health frame: display a sequence of health indicators.

These indicators are meant to visualize health, viability, performance or other relative to a threshold characteristic. The indicators may color codes to distinguish between the states/characteristics. However, the design of the indicator should also be color blind friendly. Health indicators should be click-able if detailed status about that indicator is available in HTML. Indicators may be sorted in ascending order using each indicator's weight.

j) Support scatter plotting of data table column values - generic table plotting display.

2.5 The “Drill down”.

- a) Each frame element must export list of click-able tags, which provide enough context to initialize a template URL.
- b) Each frame must support a template-like URL string which will be used in conjunctions with values of tags clicked by the user. This template URL should be composed such that its instance is the URL pointing to more detailed information about clicked tag. The purpose of using template URLs is to decouple drill down design from implementation of the frame. That is, the design of the frame should allow insertion of that frame into potentially many independent drill down hierarchies.
- c) URL of the drill down page (see above) may be managed by one of the portal windows or it can be hosted by other applications.

2.6 Data Warehouse.

- a) Using a common data format, establish support for acceptance and transient storage of data which can be used to generate content of individual web frames.
- b) Provide mechanism for querying the warehouse.
- c) Support limits on size , quantity and time of validity for the data which can be stored by the warehouse.

2.7 Operational Requirements

2.7.1 Security

- a) A web page must offer sign-in and recognize sequence of requests as belonging to a single user. We do not require support for GSI authentication at this time. (kerberos tickets are supported via kerberos to X509 conversion)
- b) All services must be deployed in a “secure” web container environment. The web container environment should prohibit remote administration interfaces which can potentially enable upload or modification of project's codes. This requirement relates to both data transformation and frame management systems.
- c) The data sensitive to domain should not be available outside of that domain. Note to implementer: this can be accomplished by setting up HTTP traversal rules specific to address of the portal and/or password protecting the portal web page.

2.7.2 Response time

- a) We expect the system to render a “reasonably” complicated portal web page within 1 or 2

seconds. Consequently , no data source should require more then 2 seconds to respond with valid result or provide standard error message indicating the timeout or other error condition.

- b) The MCAS deployment will be operating as service shared between multiple users/experiments. The MCAS deployment must also able to manage several independent contexts at a time. The system should be comfortable supporting several portal pages that may be owned by different user groups.
- c) The MCAS service is not required to be a critical system.

2.7.3 Maintainability

Routine manual maintenance should be limited to:

- a) selective backups of data (if requested by data users)
- b) software and hardware upgrades

System should automatically

- c) Prune data that does not fit the service agreement with data warehouse
- d) Rotate log files
- e) Produce and send alarms in response to internal to MCAS system error conditions.

In addition, the system should provide:

- f) Software documentation as per -"FermiGrid software acceptance policies", <http://cd-docdb.fnal.gov/cgi-bin/ShowDocument?docid=2684>
- g) Process of testing and deployment of the software should comply with : <http://cd-docdb.fnal.gov/cgi-bin/ShowDocument?docid=2684>

2.7.4 Scalability

- a) All objects used in communicating between components of the infrastructure must carry sufficient information to be independently processable. That is, the MCAS system should not rely on shared context to execute or render *independent* user requests.

3 Sample Portal Display

The screenshot shows a web browser window with the URL `http://ft3x6:8080/portal`. The page contains a dashboard with several components:

- Generic Graphs/Image View frame:** A grid of small charts and graphs, each representing different system metrics.
- Zoomed image:** A larger, detailed view of a specific graph titled "Sum Restricted DCAP sessions" showing a bar chart with a y-axis labeled "Count" and an x-axis with time markers.
- Color coded bar graphs:** A set of horizontal bar graphs on the left side, each representing the health of a different system component. The bars are color-coded (e.g., green, yellow, red).
- Generic Table View frame:** A table titled "CellInfoTableView" with columns for "cell", "domain", "prod ping", "time", "version", and "offline".
- Configurable, 3 Column display:** A large area at the bottom of the dashboard, which is a 3-column grid of various charts and graphs.

Annotations with blue callouts provide further details:

- "Generic Graphs/Image View frame" points to the grid of small charts.
- "Each graph/image can be fetched from external data sources" points to the same grid.
- "Click on the Image to Zoom it" points to the zoomed-in graph.
- "Zoomed image" points to the detailed view of the "Sum Restricted DCAP sessions" graph.
- "Color coded bar graphs to display the health of different components in the system" points to the horizontal bar graphs on the left.
- "Generic Table View frame" points to the "CellInfoTableView" table.
- "Configurable, 3 Column display makes it easy to visually correlate information from different sources" points to the large 3-column grid at the bottom.

4 References

1. http://cd-docdb.fnal.gov/cgi-bin/RetrieveFile?docid=3267&version=3&filename=MCAS_requirements_lite.pdf
2. <http://www.fnal.gov/docs/products/mcas/>
3. <https://sourceforge.net/apps/trac/mcas/>