Monitoring Service

Monitoring is the key service needed to gain insights into an infrastructure. It needs to be continuous and on-demand to quickly detect, correlate, and analyze data for a fast reaction to anomalous behavior. The challenge of this type of monitoring is how to quickly identify and correlate problems before they affect end-users and ultimately the productivity of the organization.

Service Description

The ARGO Monitoring Service (https://argo.egi.eu/egi/documentation) provides a flexible and scalable framework for monitoring status, availability and reliability of a wide range of services provided by infrastructures with medium to high complexity. ARGO generates reports using customer defined profiles (e.g. for SLA management, operations, etc.). During the report generation, ARGO takes into account custom factors such as the importance of a specific service endpoint and scheduled or unscheduled downtimes. Foundations of ARGO Monitoring Service are:

- Sources of truth - registries containing information about what should be monitored and how the monitoring should be performed.
- Configuration Management Database is a registry which contains information about the topology of the infrastructure - entities such as sites, service endpoints, entity organization (groups, hierarchies) and contact information of users responsible for operations.
- Registry of metrics for monitoring different services .

Management teams can monitor the availability and reliability of the services from a high level view down to individual system metrics and monitor the conformance of multiple SLAs. The dashboard design enables easy access and visualization of data for end-users. APIs are also supported so as to allow third parties to gather monitoring data from the system .

Features

The key features of ARGO Monitoring Service are:

- Multiple reports availability and reliability,
- Multiple Tenants
- High availability of the different components of the system
- Loosely coupled: support API's in the full stack so that components are independent in their development cycles

- Support for Topology Configurations, Metrics and profiles to add flexibility and ease of customisation.
- Dashboard design
- Real Time Alerts
- Customer Defined thresholds

High Level Architecture

The ARGO Monitoring service collects status, performance (metrics) results from one or more monitoring engine(es) and delivers daily and/or monthly availability (A) and reliability (R) results of distributed services. Both status results and A/R metrics are presented through a Web UI, with the ability for a user to drill-down from the availability of a site to individual test results that contributed to the computed figure
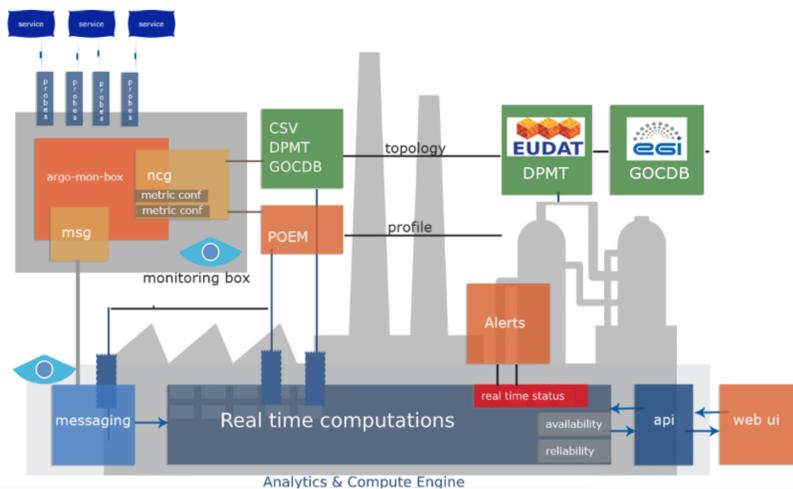
Monitoring Engine: This service executes the service checks against the infrastructure and delivers the metric data (probe check results) to the Messaging Service.

POEM: This service is used in order to define checks (probes) and associate them to service types. Each grouping of checks and service types forms a POEM profile.

ARGO Analytics & Compute Engine: ARGO Analytics & Compute Engine includes computational job definitions for ingesting data, calculating status and availability/reliability and a management service to automatically configure, deploy and execute those jobs on an Apache Flink Cluster and forward the results to the appropriate destinations (HDFS, Argo Web API, Notifications).

ARGO WEB API: Rest-like HTTP API service that provides access to status and availability/reliability results. Supports token based authentication and authorization with established roles. Results are provided in JSON Format.

ARGO Notifications: If there is a problem with a service an alert notification should be sent. Based on the real-time layer, alerting is introduced to the ARGO Monitoring Service. Real-time status events are the basis of alerts. Events are generated in the Analytics engine during computations, based on a set of rules. The alerts are customizable and contain detailed information about the various levels of groups (service endpoint, group of sites, site).

WEB UI - Lavoisier: The Web UI is based on a data aggregation framework called Lavoisier. Lavoisier is the component used to store, consolidate and "feed" data into the web application. The global information from the primary and heterogeneous data sources retrieved by means of the use of the different plug-ins. The collected information is structured and organized within configuration files in Lavoisier and, finally, made available to the web application without the need for any further computations.

Procedure to integrate a service with the ARGO monitoring

Follow the steps:

1. Open a GGUS ticket on the ARGO/SAM EGI Support Unit with:
    a. Small description of the integration - use of the service
    b. A name for the new project - infrastructure / project / service to monitor
2. The Monitoring team will create a new project into the development infrastructure for testing.
3. If the request refers to a new service type / probe then the probe should follow the guidelines mentioned in the interoperability section.
   https://wiki.eosc-hub.eu/display/EOSC/ARGO+Guidelines+for+monitoring+probes

Related Information

- POEM Documentation
- UI Documentation
- Web API Documentation
- Guidelines for monitoring probes