Data-Centric Web Infrastructure for CERN Radiation and Environmental Protection Monitoring

Adrien Ledeul, Catalina Cristina Chiriac, Gonzalo de la Cruz, Gustavo Segura, Jan Sznajd, CERN, Geneva, Switzerland

Data-Centric Architecture for SCADA data exploitation

Objective
Provide comprehensive yet accessible means to extract and exploit SCADA generated data.

Data-Centric Mindset
The data-centric mindset considers the data to be the permanent assets, and the applications the temporary ones. The key principles, as expressed in the Data-Centric Manifesto, can be summarized as follows:
- Data is the key asset.
- Data is self-describing.
- Data is stored in non-proprietary formats.
- Data access control and security is the responsibility of the data layer itself.

This approach suits well CERN’s Radiation and Environment Monitoring Unified Supervision (REMUS), as data must be kept indefinitely.

REMUS Web Architecture

Architecture Summary
- REMUS Web is deployed on the CERN Platform-as-a-Service (PaaS) infrastructure, based on RedHat OpenShift.
- The front-end runs on the user’s web browser and communicates with the back-end via:
  - WebSockets for real-time communication.
  - HTTP requests for standard communication.
- The back-end is based on Spring Boot and follows a classic three-layer architecture, with Presentation, Business, and Data layers.
- The Data Layer is based on:
  - Spring Data for accessing REMUS Web entities and the SCADA metadata.
  - EPLSUS for accessing both historical and near-real-time data from the SCADA’s instrumentation.

REMUS Data Pipeline

The Lambda Architecture
REMUS Data pipeline follows the Lambda Architecture:
- Batch processing:
  - Data is filtered from instrument’s internal memories.
  - Batch data is fed with JSDL Loader.
  - Batch transfer to next CERN Accelerator Logging Service (XKALS).
- Stream processing:
  - Data stream through Kafka, via WkCC DA.
  - InfluxDB is used for temporary data retention.
- API:
  - Radiation Unified Data Integration Service (RUUDS), based on Akka and Alpakka, unifies the access to the various data sources.

REMUS Data Pipeline: Applications consume data aggregated by the data pipeline API. Data is filtered from various data sources with different semantics and time resolutions. Jodel and XKALS are used for historical data, Akka and InfluxDB for near-real-time data.

REMUS Web Functionalities

Near-real-time Trends: REMUS Web allows to access and plot both historical and near-real-time data from a specific data source and select various parameters on the data to visualize.

Near-real-time Alarms: REMUS Web allows to access and plot both historical and near-real-time data from a specific data source and select various parameters on the data to visualize.

Metadata Statistics: REMUS Web allows access to the complete inventory of SCADA entities, users can filter the data using multiple criteria and perform data aggregations for various collections.

Notification Configuration: Users can configure personalized email and SMS notifications based on events produced by the SCADA.

Dashboard: REMUS Web includes a powerful tool for creating and viewing dashboards. Users can monitor a wide range of data sources using different types of widgets. In addition, REMUS Web automatically generates a dashboard for every instrument connected to REMUS.

Domain-specific reports: REMUS Web includes tools to generate these reports.
Data-Centric Architecture for SCADA data exploitation

**Objective**

Provide comprehensive yet accessible means to extract and exploit SCADA generated data.

**Data-Centric Mindset**

The data-centric mindset considers the data to be the permanent assets, and the applications the temporary ones. The key principles, as expressed in the Data-Centric Manifesto, can be summarized as follows:

- Data is the key asset.
- Data is self-describing.
- Data is stored in non-proprietary formats.
- Data access control and security is the responsibility of the data layer itself.

This approach suits well CERN’s Radiation and Environment Monitoring Unified Supervision (REMUS), as data must be kept indefinitely.

Data-Centric principles applied to REMUS: The center of the diagram represents all the data associated with radiation and environmental protection monitoring at CERN. The external blocks represent applications consuming and producing data through APIs.
REMUS Data Pipeline:

Applications (web & standalone)

Data access API - ERUDIS

Historical: Reduced

Historical: Non-reduced

Near-real-time buffer

Near-real-time

query

query

query

subscribe

Oracle

NXCALS

InfluxDB

kafka

WCCOA-Kafka Driver

Data files

Drivers

WinCC OA

Pooling

Streaming

REMUS Data Pipeline: Applications consume data aggregated by the data access API. Data is fetched from various data sources with different latencies and time resolutions. Oracle and NXCALS are used for historical data; Kafka and InfluxDB for near-real-time data.

The Lambda Architecture

REMUS Data pipeline follows the Lambda Architecture:

**Batch processing**
- Pooling from instrument’s internal memories.
- Batch data file injection with SQL*Loader.
- Batch transfer to Next CERN Accelerator Logging Service (NXCALS).

**Stream processing**
- Stream through Kafka, via WinCC OA.
- *InfluxDB* is used for temporary data retention.

**API**
- *Radiation Unified Data Integration Service* (ERUDIS), based on Akka and Alpakka, unifies the access to the various data sources.
REMUS Web Architecture

Architecture Summary

• REMUS Web is deployed on the CERN Platform-as-a-Service (PaaS) infrastructure, based on RedHat OpenShift.

• The front-end runs on the user’s web browser and communicates with the back-end via:
  • WebSockets for real-time communication.
  • HTTP requests for standard communication.

• The back-end is based on Spring Boot and follows a classic three-layer architecture, with Presentation, Business and Data layers.
  • The Data Layer is based on:
    • Spring Data for accessing REMUS Web entities and the SCADA metadata.
    • ERUDIS for accessing both historical and near-real-time data from the SCADA’s instrumentation.
**REMUS Web Functionalities**

**Near-real-time Trends**: REMUS Web allows to access and plot both historical and near-real-time SCADA data. Users can fetch data from multiple data sources and define various parameters on the data to select.

**Metadata Statistics**: REMUS Web allows access to the complete inventory of REMUS entities. Users can filter the data using multiple criteria and perform data aggregations for statistics extraction.

**Near-real-time Alarm Screen**: REMUS alarm screen is accessible directly from the web and displays the generated alarms in near-real-time.

**Notification Configuration**: Users can configure personalized e-mail and SMS notifications based on events produced by the SCADA.

**Dashboards**: REMUS Web includes a powerful tool for creating and visualizing dashboards. Users can compose their own dashboards by combining different types of widgets. In addition, REMUS Web automatically generates a dashboard for every instrument connected to REMUS.

**Domain-specific reports**: REMUS users require domain-specific reports for radiation and environmental protection. REMUS Web includes tools to generate these reports.