



Diagnostics and Monitoring the CERN Controls Infrastructure

The DIAMON Project : First Deployment in Operation



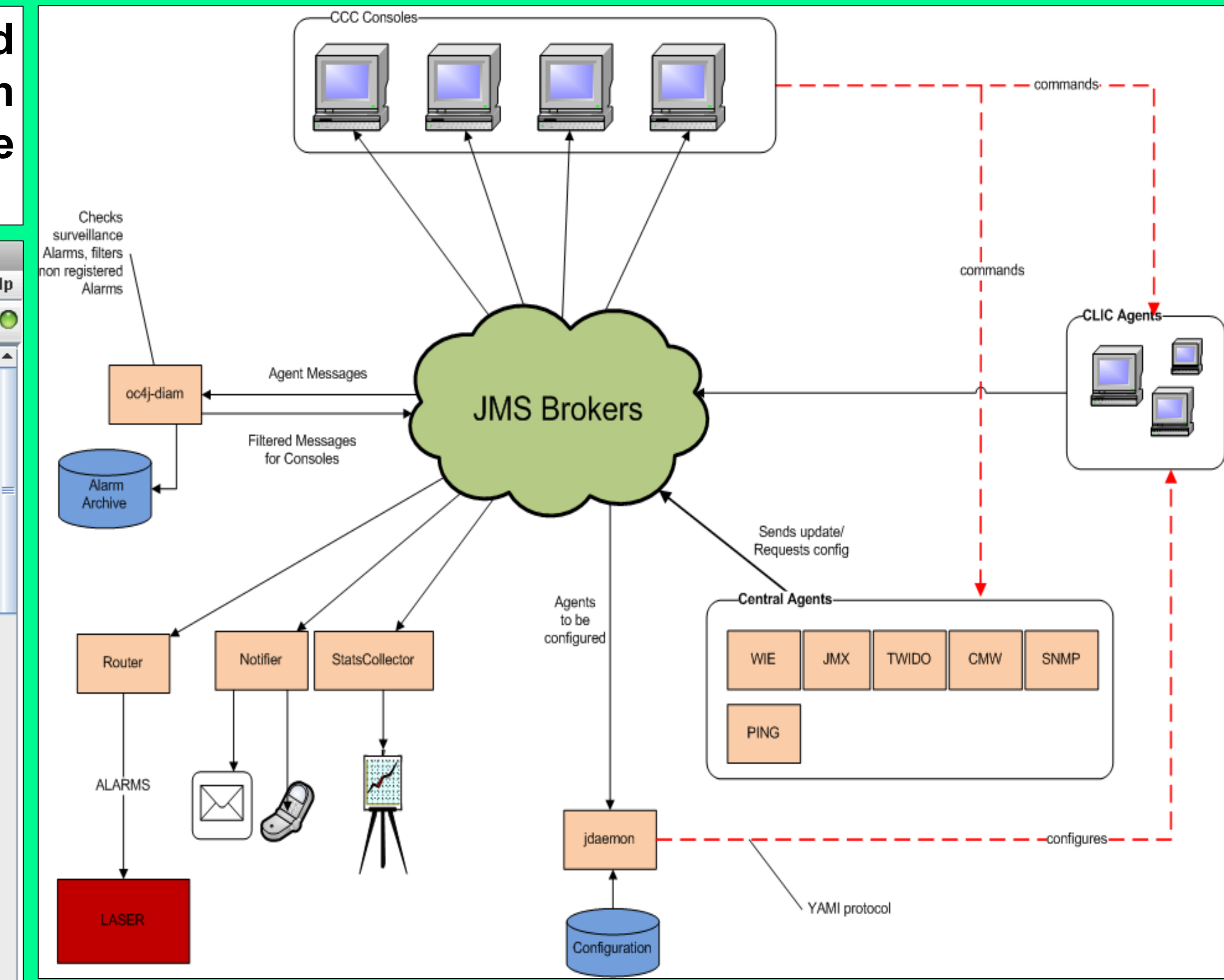
Mark Buttner, Pierre Charrue, Felix Ehm, Joel Lauener, Maciej Sobczak
Beams Department - Controls Group
CERN - European Organization for Nuclear Research - Geneva, Switzerland



Abstract

The CERN accelerator controls infrastructure spans over several machines and several thousands of devices are used to collect and transmit piece of control data. Each of these remote devices might fail and therefore prevent correct operation. A complete diagnostic and monitoring infrastructure has been developed in order to provide Operation crews with complete and easy to use graphical interface presenting the state of the controls system. Simple agents running in each surveyed item periodically report monitoring information to a central server. Graphical JAVA clients in the operation centers subscribe to this monitoring data and display a view of the current state of the machines. Mouse actions from these clients allows for diagnostic commands to be sent to the agent to get additional details or to repair a faulty situation. This presentation will describe the overall architecture of DIAMON, present the different agents running in the controls system and a few views of the graphical clients. The outcome of the first months in operation of the DIAMON tools will also be presented. Finally, the future plans will be exposed.

The purpose of the **DIAMON project** is to propose to the operators and equipment groups **tools to monitor the Accelerator Controls infrastructure with easy to use first line diagnostics and tools to solve problems or help to decide about responsibilities for first line of intervention.**



OVERALL ARCHITECTURE OF DIAMON

- DIAMON is a typical 3-tier application, communicating over JMS (Java Messaging Service).
- The **“data acquisition tier”** (agents) collects the data to be monitored,
 - The **middle-tier** based on SonicMQ message brokers and a business logic implemented using OC4J (the Oracle application server for J2EE) processes the data
 - The **presentation tier** is a Java Swing application receiving the information from the middle-tier over JMS.

OUTCOME OF EXPLOITATION

DIAMON is now used in the CERN Control Center for more than 1 year. Approximately 1'500 computers, 300 PLCs, 300 fan trays and many other equipments are actually monitored by DIAMON. Almost the **whole controls infrastructure of CERN is visible in the application in a unified way.** The tool provides rich functionality: ranging from the display of equipment status to the possibility to reboot frontend computers or restart failing processes.

FUTURE PLANS

- The future plans for DIAMON are based on the limitations observed during exploitation: the application must become **more robust, faster** in some areas and should make a better use of CERN **standard components.**
- To achieve this, we mainly plan to develop a new middle-tier to replace LASER. Key elements to the new development should be:
- Completely **isolate** the middle-tier from the **database** (no direct connection)
 - The data-tier to middle-tier **communication** should use **JAPC** (Java API for Parameter Control) instead of the pure JMS approach. JAPC is the recommended API to collect data from devices in CERN's controls infrastructure.
 - Implement the possibility to **load/replace** definitions in the configuration without restarting the server
 - Optimize the way **configuration data** is structured, in order to reduce the start up time of the DIAMON GUI.

- The **DIAMON GUI** is an easy to use and highly configurable graphical interface. The application window contains three separate panels:
- The tree on the left-hand side provides a **hierarchical view** on the monitored equipment, for instance with 3 levels: the accelerator, the controlling computer, the connected equipment. Users can configure the levels and content of the tree..
 - The top right panel displays the **full list of devices** corresponding to the element selected in the tree. The background color clearly displays the status of each item (green: OK; yellow: warning; red: error).
 - The bottom right panel (the “Detail panel”) displays **all details** for the element selected in the left hand-side panel (if the selected element is a leaf node) or in the top right panel.