High-Availability Monitoring and Big Data: Using Java Clustering and Caching Technologies to Meet Complex Monitoring Scenarios

San Francisco, California
6 – 11 October 2013

M. Bräger, M. Brightwell, E. Koufakis, R. Martini, A. Szwalska,
CERN, Geneva, Switzerland

Monitoring and control applications face ever more demanding requirements: as both data sets and data rates continue to increase, non-functional requirements such as performance, availability and maintainability become more important. C2MON (CERN Control and Monitoring Platform) is a monitoring platform developed at CERN over the past few years. Making use of modern Java caching and clustering technologies, the platform supports multiple deployment architectures, from a simple 3-tier system to highly complex clustered solutions.

C2MON Architecture overview

C2MON implements a three-tier Java architecture using the Java Messaging (JMS) framework ActiveMQ as middleware, which allows an anonymous, fault-resilient and horizontally scalable communication. A major aim of the C2MON platform is to provide a clustered server layer that is able to consume data updates in a load-balanced manner. The modular concept allows writing functional extensions for all three layers and to profit from many ready-to-use components. Together with its flexible deployment C2MON is adaptable in a short time-scale to many different monitoring scenarios.

Client Layer
The C2MON Client API uses JSON messages to enable the communication between server and client layer, and the execution of pre-configured commands.

Server Layer
The C2MON server runs as a standalone Spring application, and comprises of a core part, and a set of optional modules.

Data acquisition Layer
The DAQ layer offers drivers to acquire data from a variety of sources (RPCs, PLDs, Oracle databases or other CERN specific protocols). Each DAQ process runs on a common DAQ core, which manages the communication with the C2MON server tier, and can also apply filters improving the quality of the data.

A modular server concept, adaptable to individual needs

The heart of the system, based on an optionally distributed cache, provides a basic set of functionalities: communication, lifecycle management and configuration of the DAQ, initial load of the in-memory cache, recovery options, and evaluation of alarms and business rules. The possibility of adding as many individual server modules as needed makes the use of the system very flexible.

C2MON: 3 Deployment Scenarios

**Deployment 1**

- C2MON client
- C2MON server
- ActiveMQ Broker

- DAQ process

**Deployment 2**

- C2MON client
- C2MON server
- Terracotta standby
- Terracotta server
- ActiveMQ Broker

- DAQ process

**Deployment 3**

- C2MON client
- C2MON server
- Terracotta standby
- Terracotta server
- ActiveMQ Broker

- DAQ process

Fast and Simple
Good performance, easy setup, and very simple management. Made for scenarios where availability is less critical.

Redundant and Available
The added redundancy on the server level allows for rolling updates, as well as providing instant failover in case of a single server failure.

Maximum Performance, Maximum Availability
Multiple server nodes form a cluster, and are also optimised for certain data points in the cache. In this scenario data distribution is highly optimized, achieving maximum performance and availability to meet even the most stringent requirements.

@CERN: DIAMON
DiAgnostic and MONitoring (DIAMON) uses C2MON to provide the CERN operators with tools to monitor more than 3000 devices, high level applications and servers across CERN.

@CERN: TIM
The Technical Infrastructure Monitoring (TIM) uses C2MON to supervise and control 120,000 monitoring points and to handle more than 60,000 different alarms.

BIG DATA SCENARIO
This scenario is not yet deployed, but it would fit a system that gathers data at a high rate from millions of data points. At the same time it would be possible to instantly deliver the data to a large set of clients, or for complex real time processing across the entire cache.