

A J2EE SOLUTION FOR TECHNICAL INFRASTRUCTURE MONITORING AT CERN

J. Stowisek, P. Sollander, R. Martini, CERN, Geneva, Switzerland

Abstract

The Technical Infrastructure Monitoring (TIM) system has been designed and is being implemented as the new control system for CERN's technical services. This control system is being built on industry standard components, industrial PLCs and enterprise server hardware, using the Java 2 Enterprise Edition (J2EE). The TIM project will use J2EE technology, notably Enterprise Java Beans and the Java Message Service, to develop a highly available, reliable, scalable and flexible state-of-the-art control system.

TECHNICAL INFRASTRUCTURE MONITORING AT CERN

The CERN Technical Control Room (TCR) has the task to monitor the laboratory's entire technical infrastructure 24 hours a day, 365 days a year. The supervised installations include the electrical distribution network; heating, cooling, ventilation and air conditioning equipment; high vacuum and cryogenics systems, safety systems, etc.

To fulfil their mission, TCR operators rely on a set of application programs that are part of the control system that continuously provides them with accurate monitoring data. The current TCR control system, which was designed to meet the requirements of the LEP era, has become obsolete. Hence, the TIM project was launched to (a) identify the monitoring needs of technical data for the Large Hadron Collider (LHC) era, (b) to propose a solution for a coherent, reliable and maintainable control system and (c) to implement this solution before the start-up of the accelerator in 2007.

FUNCTIONALITY

The following supervision applications and tools are used to monitor and control CERN's technical services:

- Alarm consoles alerting the operators about equipment failures, system malfunctions and potential safety hazards.
- High-level synoptic diagrams showing the overall state of an installation, one example of which is the *Gestion Technique de Panne Majeures* (GTPM) application for monitoring the state of a particle accelerator and its auxiliary systems [1]
- Synoptic displays showing detailed information about an installation or subsystem and enabling control parameters to be sent to the supervised equipment through an animated graphical representation.
- Data analysis and trending tools for displaying and graphically analysing historical monitoring data.

- System supervision and remote control tools for monitoring the state of the control system itself.

All these different application programs need to be fed with accurate and reliable monitoring data at all times. Therefore, data acquisition, data processing (including alarm treatment and the computation of composite states), data logging, data distribution as well as data configuration are the core functions of the TIM system.

Data Acquisition

The TIM system will have to treat an estimated 100,000 different *data tags*, comprising digitised analogue measurements, like temperatures and voltages, and equipment states and commands. These data tags are acquired from various kinds of data sources, including about 60 Siemens and Schneider PLCs, 40 Supervisory Control and Data Acquisition (SCADA) systems, the CERN Safety Alarm Monitoring (CSAM) system and the CERN Data Interchange Protocol (DIP). The ever-changing environment in which the TIM system will have to evolve requires that the interfaces to these sources be standardised, so as to facilitate the integration of new sources, when the need arises.

Data Processing

All signals acquired by the TIM system need to be processed before they are stored and distributed to clients. A first level of processing consists of range checks and dead band filtering. By ignoring negligible value changes and rejecting faulty values on the lowest level, the overall load on the system can be reduced considerably. In a second step, a tag value change may trigger the evaluation of a rule for computing a composite state. Composite states are higher-level data tags representing the state of an entire subsystem. They are useful for clients like the GTPM application, which are not interested in detailed equipment data but only display the overall state (e.g. OK, WARNING, ERROR) of a system. Moreover, certain tag changes, e.g. the state of a fire detector, will activate or terminate an alarm, which must be passed on to the LHC Alarm Service (LASER) to be displayed on TCR alarm consoles. As TIM clients can connect/disconnect at any time, data tag values must be persistent, so that connecting clients can always obtain the latest value for any requested tag.

Data Distribution

TIM monitoring data will be distributed to various kinds of client applications, including animated synoptic displays and trending tools, via a publish/subscribe mechanism. Furthermore, selected tags will be made available to external users via the CERN Data Interchange Protocol (DIP).

Data Logging

All incoming data will be written to a short-term log, where it will be kept for a minimum period of 2 weeks. Outgoing data, i.e. commands sent to controlled equipment, will also be logged but kept indefinitely. The short-term log will be accessible to end-users and system administrators through a Web interface as well as graphical data analysis and trending tools. A configurable subset of the logged data will be transferred to the LHC/TCR Logging System for long time archival and off-line analysis.

Data Configuration

The data configuration for TIM will be stored and managed in an off-line reference database. User-friendly configuration management tools and strict consistency checks on off-line configuration data will reduce the risk of applying faulty configurations to the operational system. The on-line system will then be configured from the reference database using automated loading procedures. The ultimate goal is to be able to reconfigure the entire system without having to restart any of its parts.

CONSTRAINTS

The TCR operates 24 hours a day, 365 days a year, without interruption, a requirement that imposes very strict availability and reliability constraints on the TIM system. As scheduled downtime for maintenance is essential and occasional system failures are inevitable, the reliability of TIM monitoring data (displayed on TCR applications, written to the logging system or passed on to external systems) is paramount. Reliability is attained through the notion of *data quality*. In addition to its value and a timestamp, each data tag will have a quality attribute indicating whether the tag's current value can be "trusted". Whenever a data source becomes unavailable or a partial system failure is detected, all affected data tags will be marked as invalid and all clients will be notified of the change.

Scalability is crucial for the TIM system as it is likely that other control systems, like the currently SCADA-based cooling & ventilation control system or the access control system for PS and SPS secondary zones will be replaced by the TIM solution. As the number of tags and the frequency of value changes increase, the system will have to be scaled to be able to cope with this higher load without decreasing the overall system performance and data throughput.

OPTING FOR A J2EE SOLUTION

In an extensive feasibility study, three candidate solutions were evaluated:

- upgrading the current SmartSockets-based TCR control system and porting it to a new hardware platform,
- customising a commercial SCADA product and

- developing a custom-built solution based on standard J2EE components.

Initially, all three options seemed suitable for building a coherent control system that would fulfil the functional requirements described above. However, a simple upgrade of the current control system proved to be unworkable and was abandoned very early in the decision-making process. Firstly, known limitations of the TDS application would have required a complete reengineering of the system to be able to cope with the increased monitoring needs of the LHC era. Secondly, TIM would have been the only controls project at CERN using the SmartSockets middleware, which would have made it difficult to profit from synergies with other controls groups.

The SCADA option was investigated next and soon discarded after the implementation of a functional prototype. Notwithstanding that SCADA systems are widely used at CERN and in other industrial controls environments, the TIM feasibility study showed that commercial SCADA products are more adapted to homogeneous applications where the whole system is built from scratch and the entire data acquisition chain is centrally managed than to the heterogeneous and loosely coupled TCR controls environment. The TIM system has to interface with several external systems, such as the LHC Alarm Service (LASER), the LHC/TCR Logging System and the CERN Safety Alarm Monitoring system (CSAM). Hence, the difficulties encountered when trying to build these interfaces with PVSS II, the CERN recommended SCADA system, convinced the team that adapting a black-box commercial system was much more difficult and costly than integrating standard components and existing tools into a custom-built system fulfilling the TIM requirements.

With simplicity, portability, scalability, and easy legacy integration, the J2EE platform has become an industry standard for developing component-based multi-tier applications. The design goals for TIM were to keep the system as simple as possible and to base it on standard technologies and components. J2EE is a specification for which many different implementations are available on the market. Java was expressly designed for use in distributed environments and has become a de facto industry standard. For all these reasons, J2EE seemed to be the platform of choice for TIM. A prototype implementation of some core use cases on an open source implementation of the J2EE platform convinced the team that J2EE was – technically and economically – the best solution for TIM [2]. Finally, a series of performance tests gave support to this choice.

CHOOSING A PLATFORM

The operational platform for the TIM system was chosen in collaboration with the CERN J2EE Working Group [3]. Apart from TIM, several other controls projects at CERN had opted for a J2EE solution and thus needed a stable Enterprise Java Beans (EJB) platform for

operation. Hence, reaching agreement on a common hardware and software platform was a vital step towards building up a shared support infrastructure and consequently reducing the cost of system administration and maintenance.

Choosing a J2EE implementation

Many implementations of the J2EE specification – either open source or commercial – are available on the market and the experience gained from using the JBoss EJB container for a prototype implementation was very positive. However, due to CERN having a general license and support agreement with Oracle, Oracle OC4J was chosen for the operational system. Using an application server and a database from the same vendor is sensible with respect to potential integration problems.

SonicMQ was agreed on as a Java Message Service (JMS) provider. In addition to being fully compliant to the JMS specification, SonicMQ offers some extra features, such as continuous availability and load balancing through clustered brokers, which proved to be extremely useful for TIM. Furthermore, SonicMQ includes a C++ API for integrating legacy data sources with JMS.

The integration of J2EE components from different vendors may yield incompatibilities, mostly due to grey areas in the J2EE specification. One such conflict, which prevented message-driven beans on OC4J from automatically reconnecting after a SonicMQ JMS provider failure, was solved in collaboration with SonicSoftware for the entire CERN user community.

Choosing an operational hardware platform

The TIM EJB containers and JMS brokers will be hosted on HP ProLiant DL380 servers. According to Hewlett-Packard, the ProLiant DL380 has been performance optimised for running Web and application servers. The TIM ProLiant machines have redundant power supplies, CPUs, cooling, network adapters etc. which makes them highly reliable and easily maintainable hardware platforms. The machines hosting the EJB containers and JMS brokers as well as most TIM drivers will be running RedHat Enterprise Linux 3.0. Only one data acquisition machine will be running Windows.

The back end of the TIM system is an Oracle 9i database cluster running on two Sun Fire V240 machines with Sun StoreEdge disk arrays. This Oracle database will be running on Sun Solaris 9. Two HP rp2405 servers with a shared RAID array serve as TIM file servers.

SYSTEM ARCHITECTURE

The TIM system is a classic example of a multi-tier architecture, as illustrated by Figure 1. Approximately 100 data acquisition processes, also referred to as drivers, form the *data acquisition tier*. An EJB application, running inside a redundant OC4J installation with a clustered JNDI service and a distributed Java Object Cache forms the core of the TIM *business tier*. The

business tier is completed by databases for persistence and logging as well as a PL/SQL application for periodically exporting batches of TIM data to the long-term logging system. Finally, a set of client applications consuming data tags and enabling the user to send commands to controlled equipment form the *client tier*.

The three tiers of the TIM system are strictly separated. Drivers, server-side components and clients only communicate with each other via JMS messages. A strict control of the interfaces between the tiers makes the system more robust and secure.

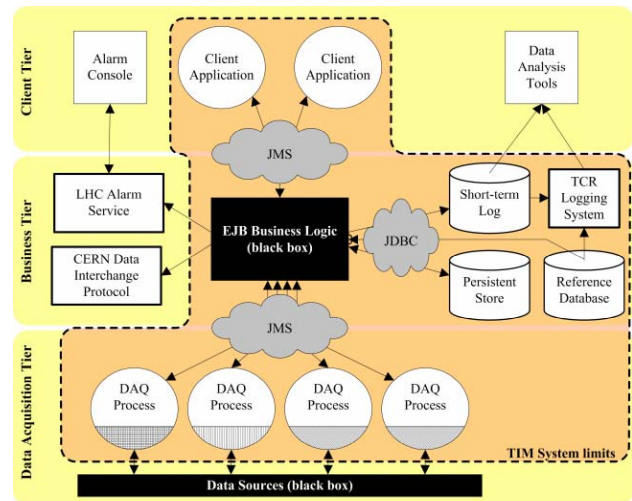


Figure 1: Overview of the TIM system's architecture

An in-depth discussion of the system's implementation would go beyond the scope of this paper. However, the reader is invited to contact the authors, should he wish to obtain more detailed (technical) information.

STATUS

A first TIM application will be put into operation in July, 2004. It will replace the currently PVSS-based GTPM system. This first release of the system will not implement all planned functionality. Furthermore, it will use a bridge to the current TCR control system for data acquisition. All missing core functionality, e.g. command execution and system supervision, as well as further DAQ modules must be developed and tested before the end of the year in order for the TIM system to be fully operational.

REFERENCES

- [1] M. Batz, "Engineering Process Monitoring for Control Room Operation", CERN Yellow Report 2001-2002, February 2001, p. 65
- [2] P. Sollander, R. Martini and J. Stowisek, "Migrating the TIM System to J2EE", July 2003.
- [3] <http://cern.ch/j2ee-wg>