THE CERN ACCELERATOR LOGGING SERVICE - 10 YEARS IN OPERATION: A LOOK AT THE PAST, PRESENT, AND FUTURE

C. Roderick, L. Burdzanowski, G. Kruk, CERN, Geneva, Switzerland

Abstract

During the 10 years since it's first operational use, the scope and scale of the CERN Accelerator Logging Service (LS) has evolved significantly: from an LHC specific service expected to store 1TB / year; to a CERN-wide service spanning the complete accelerator complex (including related sub-systems and experiments) currently storing more than 50 TB / year on-line for some 1 million signals. Despite the massive increase over initial expectations the LS remains reliable, and highly usable - this can be attested to by the 5 million daily / average number of data extraction requests, from close to 1000 users. Although a highly successful service, demands on the LS are expected to increase significantly as CERN prepares LHC for running at top energy, which is likely to result in at least doubling current data volumes. Furthermore, focus is now shifting firmly towards a need to perform complex analysis on logged data, which in-turn presents new challenges. This paper reflects on 10 years as an operational service, in terms of how it has managed to scale to meet growing demands, what has worked well, and lessons learned. On-going developments, and future evolution will also be discussed.

INTRODUCTION

The CERN accelerator Logging Service (herein referred to simply as the "LS") is used to store and retrieve billions of data acquisitions per day, from across the complete CERN accelerator complex, related sub-systems, and experiments [1].

The LS is considered a mission critical service, heavily relied upon to support day-to-day operation, with close to 1000 users of the logged data. As such, the availability and performance of this service are paramount.

EVOLUTION OF SCOPE

In 2001 the LHC Logging project was launched, with the scope of data logging for the LHC only, and the need to be ready for operational use during the commissioning of the LHC sub-systems (several years before the planned LHC start-up).

Based on past experience of data logging for LEP (LHC's predecessor), it was estimated that 1TB / year would be logged during LHC operation, which was estimated to last for approximately 20-25 years.

With the evolution in storage systems (growth in capacity / relative to cost), it was decided to store all captured data on-line (i.e. on disk) beyond the LHC lifetime.

The LS was first used operationally in September 2003 (5 years before the start of LHC beam commissioning), to capture data during TT40 extraction tests (extraction of beam from the SPS accelerator into the TT40 transfer line towards the LHC tunnel). This first usage proved extremely useful to understand and tune the SPS-to-LHC beam extraction process, and quickly led to establishing data logging for a significant number of other data from the SPS accelerator.

This trend continued over the following years, including data from LHC sub-system hardware and beam commissioning, subsequent beam-operation, the complete CERN injector complex, and general services such as water distribution and electrical networks. Overall: a huge increase beyond the initial scope of the LHC Logging Project, leading to a CERN-wide Logging Service spanning the complete accelerator complex.

Since the start of the LHC hardware-commissioning phase, the LS is classified as mission-critical, as the data captured is regularly used for decision making after unforeseen events.

The latest LS data processing represents throughput of more than 100 TB / year, for some 1 million signals, and storing more than 50 TB / year on-line. Figure 1 show the evolution of records logged in the LS, with markers indicating reaching 1TB logged, and the start of LHC commissioning and operation.

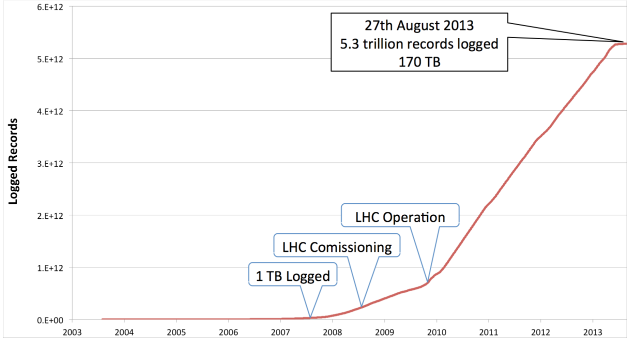

Figure 1: Evolution of logged records.

ARCHITECTURE OVERVIEW

Figure 2 shows a basic overview of the current LS architecture, which is comprised of:

- Two Oracle databases: A so-called Measurement database (MDB) where raw data from Java processes and other Oracle databases is persisted during seven days, and a Logging database (LDB) where a sub-set of MDB data and pre-filtered data from industrial SCADA systems are stored on-line indefinitely.
- Distributed Java processes / APIs are responsible for loading data into the databases.
- A sub-set of MDB data is transferred to the LDB using in-house developed PL/SQL code that uses a

comprehensive set of metadata to dynamically filter the data of long-term interest.

- A powerful distributed Java API is the sole means of extracting data from the databases, which includes a command line interface. Applications wishing to use the API must be pre-registered. At the time of writing there are 124 applications registered to a heterogeneous client community, which collectively account for an average of 5 million extraction requests per day. Direct SQL access is not permitted.
- A generic Java GUI called TIMBER is also provided as a means to visualize and extract logged data. The tool is heavily used, with more than 800 active users.

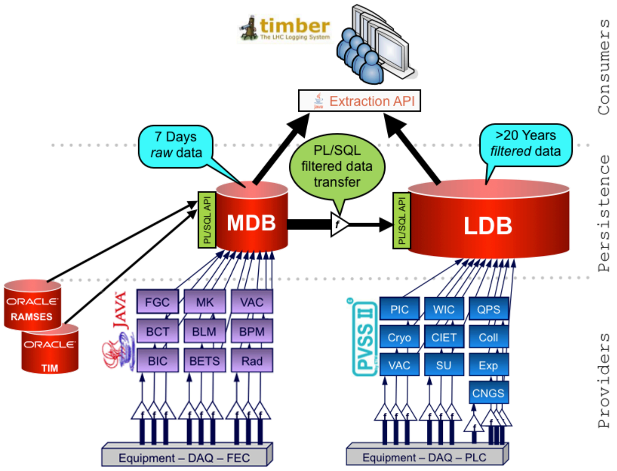

Figure 2: Logging service architecture overview.

The Java APIs for both logging and extracting data are significantly optimized for performance, and more importantly – service stability. For data extraction clients, the fact that database access is actually made in a distributed manner via a remote application server is hidden within the Java client API.

High-availability (HA) is an essential requirement for the mission-critical LS, as such the MDB and LDB databases are run as 2-node Oracle RAC (Real Application Cluster) clusters, using mid-range Linux based machines. For performance reasons, only one node of the cluster is actually used to run the service at any given moment [2].

ACHIEVING SCALABILITY

Combing the following facts, there should be no doubt that scalability has been successfully achieved so far:

- With a throughput of 100 TB / year, the LS is dealing with data rates two orders of magnitude above the initial expectations.
- The minimal, mid-range hardware used, has only been replaced twice in 10 years upon reaching end-of-life, and is currently running with low resource consumption (~10%).

Achieving this scalability can be attributed to a number of points:

A Good Database Design

The database schema used is scalable by design [1] – not being affected by the need to log additional signals from new equipment, and exhibiting constant data extraction times independently of the number of logged records.

Extensive Instrumentation

All elements of the LS, from client libraries to database procedures have been heavily instrumented. This enables an understanding of how the LS is being used and performing, in terms of who, is doing what, from where, how it is being done, and how long things take [3].

Optimal Use of Software Technology

The database model implementation, integrated business logic (written in PL/SQL), and the surrounding Java infrastructure interacting with the database has been engineered to use the most appropriate features from Oracle, to maximize performance [1], [2]. Knowing how the LS systems are being used (or misused) from the aforementioned instrumentation is vital in order to know which features and techniques to use to improve system performance.

Data Quality Control & Filtering

The MDB (introduced in 2005) has advanced data filtering capabilities when transferring data to the LDB for long-term storage. Thanks to a significant human effort [4] to ensure appropriate per-signal filtering configurations, the MDB manages to reduce the data transferred to the LDB by 95% (2 TB / week), only persisting value-changes of long-term interest.

RECENT / ON-GOING DEVELOPMENTS

Flexible Data Lifetimes

The initial idea in the LHC Logging Project to keep all logged data on-line beyond the LHC lifetime was aimed at simplifying the management of logged data over time, and was based on the assumption that 1 TB of data would be acquired per year. Some 12 years later, with data rates already at 50 TB / year stored, and expected to reach 300 TB / year in 2015 (after the LHC long shutdown), it is no longer cost-effective to simply store all data indefinitely. The actual required data lifetimes vary greatly (from days to tens of years) according to the nature of the data.

Developments are underway to provide a flexible means to configure scheduled removal of data on a per signal basis. Although this may sound simple, due to the scalable database schema design and the optimizations in place for storage and backups [2], actually recovering space for re-use presents a significant challenge.

One Logging Solution for All Data

Until now, the LS has been comprised of the database centric solution described above, and an SDDS (Self Describing Data Sets) files based solution, with the latter

ISBN 978-3-95450-139-7

being used mainly for complex data structures such as image captures or multi-megabyte beam profiles [4].

At the time of writing, work is almost complete to incorporate the logging of complex data structures into the database centric solution, thus suppressing the need for SDDS files. The main aims of this work are to:

- Provide LS users with a single interface to extract all types of data, with identical functionality regardless of data type.
- Reduce the costs of providing and maintaining two distinct infrastructures for data logging.

Improving the Java Logging Processes

The Java Logging processes that subscribe to accelerator data published over the Controls Middleware [5], and write received values to the MDB are currently being re-written in order to become simpler, more robust, deterministic, and easily diagnosable.

Since these processes are effectively located in a long logical chain between accelerator equipment and the MDB, it is vital that potential data loss be detected as early as possible, with the cause understood immediately (e.g. device data not logged because device disconnected from the middleware, or device produced invalid values).

The main objective is to move away from the current situation whereby problems may go undetected for long periods of time, or insufficient information is available to determine the cause.

The key means to achieve the objectives are constant capture and monitoring of health and activity metrics, exposed via JMX, and monitored using the DiaMon (Diagnostics and Monitoring) infrastructure [6]. Investigations are also on going into the use of the Esper CEP (Complex Event Processing) engine [7] to facilitate the capture and exploitation of logging process metrics.

Data Analysis

The LS data is successfully used for analysis of systems and particle beam behavior by hundreds of users.

However, once analysis had been completed, there was no central means to store and share the analysis results with others. During 2013, the LS infrastructure was further developed to allow the publishing and logging of analysis results – just like raw data acquisition logging.

Despite the excellent performance of the LS, it can still take a long time to extract huge amounts of data in order to perform the required analysis.

To address this point, and try to simplify data analysis in general, an initiative has been launched with the key objective of being able to perform analysis as close to the relevant data source as possible. The current idea is to have a means to describe the analysis to be performed using a DSL (Domain Specific Language), with the option to schedule the analysis based on time or beam related events, and describe how to persist the results for subsequent access by anyone. This would mean for example that instead of performing the analysis on a client machine, the analysis could be described and executed inside the LS databases using PL/SQL or Oracle Enterprise R. Thanks to the DSL, the actual implementation details would be hidden from the end user.

SUMMARY

After 10 years in operation, with close to zero downtime, and playing a vital role in the commissioning and operation of CERNs particle accelerators, sub-systems and experiments – the Logging Service can be considered a great success.

Good instrumentation has proven to be a vital ingredient for this success – understanding system usage has helped shape the LS to meet requirements.

Providing a successful service inevitably leads to ever increasing demands being placed upon it. Efforts are on going to bring additional value to the end-users of the LS, and to ensure continued scalability for the future.

REFERENCES

[1] C. Roderick and R. Billen, “Capturing, Storing and Using Time-Series Data for the World’s Largest Scientific Instrument”, November 2006, CERN-AB-Note-2006-046 (CO).

[2] C. Roderick et al., “The LHC Logging Service: Handling Terabytes of On-line Data”, ICALEPCS’09, Kobe, Japan, October 2009, WEP005.

[3] C. Roderick et al., “Instrumentation of the CERN Accelerator Logging Service: Ensuring Performance, Scalability, Maintenance and Diagnostics”, ICALEPCS’11, Grenoble, France, October 2011, THCHAUST06.

[4] C. Roderick et al., “The CERN Accelerator Measurement Database: On The Road To Federation”, ICALEPCS’11, Grenoble, France, October 2011, MOPKN009.

[5] K. Kostro et al., “The Controls Middleware (CMW) at CERN - Status and Usage”, ICALEPCS’03, Gyeongju, Korea, October 2003, WE201, p. 318 (2003).

[6] M. Buttner et al., “Diagnostic and Monitoring CERN Accelerator Controls Infrastructure - The DIAMON Project - First Deployment in Operation”, ICALEPCS’09, Kobe, Japan, October 2009, RPPA35.

[7] http://www.espertech.com