DATABASE ARCHIVING SYSTEM FOR SUPERVISION SYSTEMS AT CERN: A SUCCESSFUL UPGRADE STORY

P. Golonka, M. Gonzalez-Berges, J.Hofer, A. Voitier
CERN, Geneva, Switzerland

Abstract

Almost 200 controls applications, in domains like LHC magnet protection, cryogenics and vacuum systems, cooling-and-ventilation or electrical network supervision, have been developed and are currently maintained by the CERN Industrial Controls Group in close collaboration with several equipment groups. The supervision layer of these systems is based on the same technologies as 400 other systems running in the LHC Experiments (e.g. WinCC Open Architecture, Oracle). During the last two-year LHC Long Shutdown 1, the 200 systems have been successfully migrated from a file-based archiver to a centralized infrastructure based on Oracle databases. This migration has homogenized the archiving chain for all CERN systems, and at the same time has presented a number of additional challenges. The paper presents the design, the necessary optimizations and the migration process that allowed us to meet unprecedented data-archiving rates (unachievable for the previously used system), and liaise with the existing long-term storage system (LHC LoggingDB) to assure data-continuity.

INTRODUCTION

Storage and retrieval of historical process values and alarms is one of the invisible yet essential tasks of SCADA systems, and is always delegated to a dedicated *historian* or *archiver* tool. WinCC OA [1] offers three technologies for archiving: the file-based archiver ("valarch"), the simple database logging facility ("DBLogger", available as of version 3.13) and the high-performance, scalable Oracle database archiver ("RDB Archiver").

Even though a number of large control systems (Detector Control Systems for the LHC experiments) made use of the RDB archiver since the beginning, the control systems for infrastructure and LHC services initially chose to use the file archiver. The divergence in the choice of archiving technology stems from the requirements. The detector control system had to have the historical values available as the so called "conditions" in an Oracle database to make the data analysis possible, and it was this requirement that drove the collaboration between CERN and ETM to develop and optimize the database archiver. On the other hand, the priority for the LHC and infrastructure systems was the use of a stable, proven technology to maximize the reliable operation of the systems.

With an increasing number of control systems to deliver and maintain, and the need to scale-up applications to meet the challenges of Run II of the LHC, it became clear that use of the file-based archiver reached its limit, and a migration to the more performant and centralized archiving architecture became a necessity. In addition, the centralized management of archived data offers several advantages such as better tools for data analysis, simplified management, easier handling of the SCADA server nodes, etc.

In the paper we present the process of migrating the 200 applications of the SCADA Application Service from file-based archiver to Oracle database archiver: the initial requirements, thorough testing and optimisation phase and proof of scalability, redesign of the long-term data storage architecture, actual migration effort. We will then summarize the current state of this enterprise.

EVOLUTION OF ARCHITECTURE

To enable the analysis of data from the control systems for the LHC, data needs to be transferred to a long-term storage Oracle database called the LHC LoggingDB designed to store large amount of data throughout the life time of the LHC [2].

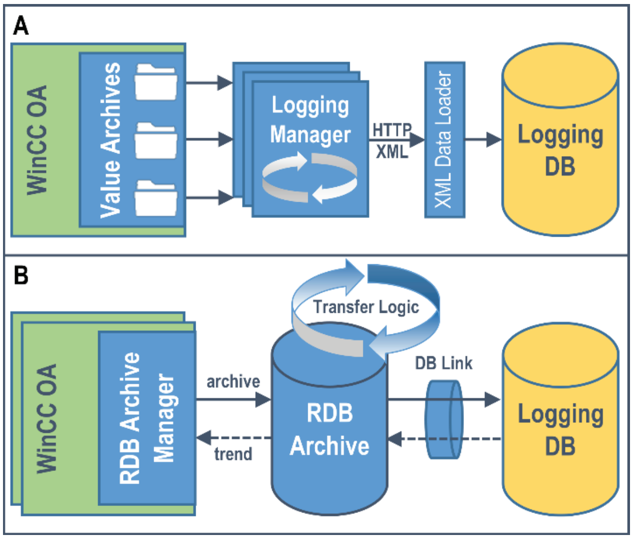

Figure 1: Archiving architecture: (A) with old logging manager; (B) with Oracle archiver and DB-link data transfer.

The evolution of the architecture for archiving is presented in Figure 1. System and process data retrieved by WinCC OA systems used to be fed to the LoggingDB using a dedicated program (Logging Manager) and complex infrastructure (XML Data Loader). This transfer was repeated every couple of minutes, and hence the data was available in the LoggingDB with a lag of

ISBN 978-3-95450-148-9

approximately 10 minutes. The reliability of the whole process was however affected by the complexity of the system and the significant number of points of failure. A number (sometimes exceeding 10) of WinCC OA archive manager programs needed to be configured in every control system, and configured correctly. Proper tuning of file archiver's parameters was also challenging for systems where the archive rate was varying during operation. Managing disk space and properly configuring data retention policies for disk-stored archive data was not an easy task. Many of these issues would vanish should one use the Oracle RDB archiver instead. Moreover, centralizing all archiving infrastructure in a single database service would simplify many management tasks.

With Long Shutdown 1 the LHC Quench Protection System had to be modernized in order to safely operate LHC magnets at the energy of 13 TeV to be achieved during Run II. The amount of data acquisition channels and data rates have been multiplied leading to the requirement to handle a sustained rate of 150 000 value changes per second, acquired and transferred to the LoggingDB. With the existing architecture, this was not possible to achieve. In addition, the XML Data Loader service of LoggingDB was going to be abandoned.

The new architecture for archiving was therefore proposed, based on the WinCC OA Oracle archiver and a central Oracle database (RDB Archive) providing service to all controls applications. Data transfer from the RDB Archive database to the LoggingDB would then be handled directly between the two databases, over a data link, using the already existing and recommended mechanism used by other clients of LoggingDB.

From a functionality point of view, the elements necessary to build the complete data path already existed and could be adapted using standard database development techniques and tools (views, database links, stored procedures, job scheduling) as well as standard features of the WinCC OA Oracle RDB archiver. However, initial tests uncovered that with existing tools and data organization, the archiving database would not be able to handle the throughput required. It became necessary to optimize the components of the system, starting at WinCC OA, through database schema and storage technology, up to the data-transfer database procedures.

Having data transferred from the RDB archive to the long-term LoggingDB storage resulted in a general policy for data retention for the former one: the data would only be kept for a limited period of time (typically three months). All the data that needs to be conserved for long term storage has to be transferred to the LoggingDB. This led to the formulation of another requirement: as the RDB and LoggingDB are already logically connected to assure data push, it should be possible to transparently retrieve historical data physically present in the LoggingDB from the RDB database; as a consequence, the operator could see the trends of historical values not limited to the three month data retention period.

RDB ARCHIVER OPTIMIZATIONS AND DATABASE REDESIGN.

To probe the performance of the archiver system a large scale test set-up was arranged for: a 2-node RAC Oracle cluster and around 50 servers capable of running up to 200 WinCC OA projects. A test database was provided by CERN's database service, with a highly detailed monitoring tool (Oracle Enterprise Manager) and the support of experts and database administrators interpreting the measured database metrics, providing guidance for tests and tuning database parameters. Similarly, the LoggingDB team provided a replica of their infrastructure and necessary expertise. This allowed us to proceed with integration tests and tuning of the performance of the transfer mechanisms.

The initial tests with the standard settings demonstrated that the requested performance would not be achieved. Even though the performance of data-insert was achievable [3], the readout of data was performing badly due to a bottleneck on the I/O subsystem (i.e. too slow access and low throughput to the disk-based large storage) of the database. Certain additional options like database block check-summing were identified as having high processing costs as well and were disabled.

The key ingredient for performance optimization was the optimization of the readout path, for the most critical task of data transfer to the LoggingDB. This operation requires that a vast majority of data stored into the RDB database is read-out within next 15 minutes to be pushed to the LoggingDB. Data needs to be accessed for all registered signals, one-by-one, for the specified period of time. This access pattern is common with another typical use case for SCADA systems, i.e. access to historical data for a specific signal, in a specific time period to make a trend plot of historical data, or to retrieve data for analysis.

The classical solution for this access pattern is the Index Organized Tables (IOT) data organization, whereby data is organized in the storage such that entries for the same primary key (in this case, the signal) are stored adjacently, making their lookup and retrieval faster. In the previous rounds of RDB archiver optimization efforts [3] the use of IOT was disfavoured for requiring more computing resources during the data-insertion process. We decided to re-visit the performance of IOT with modern hardware and software. To our surprise, the IOTs seemed to be only marginally (<5% of CPU load) more computing-intensive for workloads where 200 systems pushed their data at the accumulated data-change rate of 200 000 values/second. The estimated improvements in the data-readout performance was at least an order of magnitude. Moreover, due to the fact that the indexing data was stored in-line with data, storage space consumption dropped by more than 50% in certain cases [4].

This modification was introduced directly on the database schema, without impacting the source code, and with full compatibility. The optimization could therefore be easily applied to new systems; for all existing systems the re-structuring of large existing data sets would be prohibitively time- and resource-consuming, hence the optimization could be performed for newly gathered data, while keeping the old data in unmodified structure.

However, the database still suffered from the I/O throughput problems related to the so-called REDU log size (auxiliary structure which allows for the restoration of the database in case of failure). The only viable method of reducing its size while still keeping the high data rate was to reduce the size of a row inserted to the database. Then we compared the average row size in the RDB Archive, 59 bytes when using IOTs (or 120 with no IOTs and with additional indices) to 19 bytes in the LoggingDB. It became clear, that a lot of reduction could potentially be achieved.

We reviewed the meta-data stored with every data row of archive data, and identified those containing non-vital information. An extension was then proposed to the already existing concept of WinCC OA *archive groups* to have the archived meta-data parameterizable. The implementation required modification of the database schema (layout of tables, stored procedures), and the WinCC OA Oracle RDB archiver (C++ source code, configuration panels). A prototype was developed at CERN, in collaboration with ETM, and put to test in a test set-up. Fast-changing data generating the bulk of the throughput was directed to the specially crafted custom archive group, where meta-data was truncated to the minimum, whereas data for other signals were archived into the default archive group, with all the details intact. The reduction in data throughput (and also disk space consumption) approached a factor of three, and allowed to keep the performance at the required level.

With a growing amount of data accumulated in tables over time, the performance of data queries may degrade. Even with the queries that access data according to index organization, scanning the huge index tree penalizes the overall performance. We reached for one more optimization level by applying time-based data partitioning. At the same time we also optimized the native data segmentation of RDB archiver, based on table sets and a view accumulating them, by applying pushed predicates (an Oracle hinting technique) on the view. As a result, again, the overall perceived performance of interactive data access as well as data retransmission improved significantly.

With the above optimizations in place we conducted the ultimate performance test to simulate the "worst case scenario" in a set-up mimicking the LHC Quench Protection System, with conditions presented in Table 1. WinCC OA systems were generating value changes with varying rates of changes per second; the data was stored into RDB Archive test database and then transferred to the integration database acting as the LoggingDB. After 24 hours of stable work at this nominal throughput we simulated the scenario of RDB database failure by shutting it down completely. As expected, WinCC OA projects reacted by storing pending data in a buffer formed on their local disks. Then, the database service was re-established and we observed the buffered data being pushed to the RDB test database and then retransmitted to the LoggingDB test database at maximum achievable rate. At the same time, new data was still arriving at the steady rate.

Table 1: Test Setup for QPS Database-Recovery Scenario

WinCC OA projects	50
Archived signals	150 000
Accumulated data input rate (values/s)	200 000
Database outage duration	8 hours
Recovery time	2 hours
Peak recovery rate (insert + transfer)	1 mln rows/s

Even in a degraded mode, with only a single database node in service, it was possible to recover the buffered data. The transmission rate to the database was largely exceeding the rate at which data was generated, resulting in complete recovery in an acceptable time.

The results of these tests convinced us to deploy a production database service for our controls applications. Two dedicated (dual-node) databases were arranged for, maintained by CERN central database service. One database, called QPSR, is dedicated for the QPS applications due to the very high throughput and data reliability requirements. The other, called SCADAR, is used to archive data from all other applications.

Managing data partitions for around 200 database schemas, and executing partition-maintenance tasks (allocating new ones, dropping unnecessary ones) would not be possible without an automated tool. Partition management code contributed by the LoggingDB team was adopted, and then enhanced to allow for more dynamic control on partition allocation. Partition management allowed us also to easily configure and apply data-retention policies for each application: majority of them will have up to 90 daily partitions kept in a moving-window policy, and older ones would automatically be dropped. For some applications specific policies are applied according to requirements (e.g. hourly partitions and 20 days data retention period for the QPS). In any case, data that needs to be kept for longer is preserved in the LoggingDB.

As has happened already in the past, the collaboration between CERN and ETM allowed us to work with the source code of WinCC OA, and have the changes reviewed, accepted and included in the official WinCC OA distribution. Migration tools for existing projects were developed and made available to CERN users.

MIGRATION AND CURRENT STATUS

Migration of existing applications to the new archiving architecture infrastructure was a challenging task not only from a technical standpoint, but also from a coordination point of view. Many of the tasks such as the reconfiguration of WinCC OA projects were automated using the central deployment tool [5]. Provisioning of database accounts (one per application) and maintenance of account credentials was streamlined yet still required manual coordination. A WinCC OA configuration and diagnostic tool for the LoggingDB transfer was developed from scratch and delivered on time to migrate the applications. Each application in turn needed to be migrated with human-supervision. A lot of attention was required when reconfiguring and initiating the LoggingDB transfer for every application to minimize possible data loss. Even though the migration was performed during the Long Shutdown 1 of the LHC (June-December 2014), many applications needed to be already available for pre-commissioning of the LHC (cryogenics, QPS), or were in production (Cooling & Ventilation). This required that the migration, requiring application downtime, is performed as quickly as possible (no longer than 4 hours). During the migration and first months of operation we encountered numerous performance and stability problems, mainly due to improper configuration of data-retention and time-based partitioning. These have since been resolved.

The migration was successfully completed in January 2015 for all production applications. Assistance was also given to the LHC experiments to upgrade their archiving systems and activate the new optimizations.

As of September 2015 there are around 200 WinCC OA projects which make use of the RDB archiver for historical data and data transfer to the LoggingDB. The statistics for the two production databases are summarized in Table 2. We consider that the old method of data transfer to the LoggingDB may therefore be decommissioned now.

Table 2: Statistics for Production Archive Databases

	SCADAR	QPSR
Projects (DB schemas)	140	48
Registered signals	2 000 000	135 000
Rows recorded/transferred	20 mln/h	400 mln/h
Storage space used	11 TB	11 TB
I/O throughput	40 MB/s	70 MB/s

Efforts are currently under way to generate and present daily reports with database statistics, so that the experts might detect unexpected changes in the data rates for their applications and also tune data-reduction (smoothing) algorithms. With the large number of applications to cover and huge data payload this is a non-trivial task.

CONCLUSION

The migration of archiving system for controls application in CERN SCADA Application Service was completed with success and on time for Run II of the LHC. Single, consistent archiving technology is deployed in production CERN-wide, and the need to maintain custom programs for the transfer to the LoggingDB was eliminated. After the initial adaptation and tuning period, the reliability and performance of the service is acceptable, and new functionality was delivered to users.

ACKNOWLEDGMENT

The success of the migration presented in this paper is a fruit of collaboration of experts from three CERN departments: Engineering, Information Technologies and Beams. We would like to express our particular gratitude to Oracle database administrators and LoggingDB experts: Zbigniew Baranowski, Luca Canali, Emil Pilecki, and Chris Roderick. We also acknowledge the development and testing effort of Kacper Szkudlarek, who worked as an ETM/Siemens openlab fellow on improvements in the WinCC OA Oracle archiver.

REFERENCES

[1] SIMATIC WinCC Open Architecture (previously PVSS) SCADA software from ETM (Siemens subsidiary), http://www.etm.at

[2] C. Roderick *et al*, "The LHC Logging Service: Handling Terabytes of On-line Data", ICALEPCS 2009, Kobe, Japan, CERN-ATS-2009-099

[3] M. Gonzalez-Berges, "The High Performance Database Archiver for the LHC Experiments", ICALEPS 2007, Knoxville, USA, (2007)

[4] P. Golonka *et al*, "Performance and Scalability of Oracle RDB Archiver in WinCC Open Architecture 3.11; Test Report", CERN EDMS note 1271192, November 2013.

[5] F. Varela *et al*, "High-level Functions for Modern Control Systems: A Practical Example", ICALEPCS 2013, San Francisco, USA, (2013)