ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA019

MONARC: SUPERVISING THE ARCHIVING INFRASTRUCTURE OF CERN CONTROL SYSTEMS

J-C. Tournier*, E. Blanco Vinuela† , CERN, Geneva, Switzerland

Abstract

The CERN industrial control systems, using WinCC OA as SCADA (Supervisory Control and Data Acquisition), share a common history data archiving system relying on an Oracle infrastructure. It consists of 2 clusters of two nodes for a total of more than 250 schemas. Due to the large number of schemas and of the shared nature of the infrastructure, three basic needs arose: (1) monitor, i.e. get the inventory of all DB nodes and schemas along with their configurations such as the type of partitioning and their retention period; (2) control, i.e. parametrise each schema individually; and (3) supervise, i.e. have an overview of the health of the infrastructure and be notified of misbehaving schemas or database node.

In this publication, we are presenting a way to monitor, control and supervise the data archiving system based on a classical SCADA system. The paper is organized in three parts: the first part presents the main functionalities of the application, while the second part digs into its architecture and implementation. The third part presents a set of use cases demonstrating the benefit of using the application.

INTRODUCTION

The CERN team in charge of the Industrial Controls systems, in collaboration with equipment groups, has developed and maintains around 250+ controls applications whose domain range from cold and warm magnets protection, to cryogenics, cooling and ventilation or electrical network supervision systems. These applications, based on WinCC OA [1] and the CERN standard frameworks JCOP [2] and UNICOS [3], archive their values in a centralized Oracle infrastructure. This infrastructure is split in two main parts with one cluster of two nodes dedicated to the applications belonging to the LHC magnets protection domain (referred as QPSR in 1) and another set of two nodes dedicated to the other applications (referred as SCADAR in Figure 1). To improve the availability in case of failure of the SCADAR database, the infrastructure is duplicated to a different datacenter. Each WinCC OA application has a dedicated database schema hosted by one of the two clusters holding the archive values of the application but also some metadata related to how the schema should be managed. Figure 1 shows an overview of the Oracle database infrastructure for the CERN industrial control applications based on WinCC OA.

* tournier@cern.ch
† enrique.blanco@cern.ch

CHALLENGES

While having two Oracle database clusters dedicated to all 250+ WinCC OA industrial control applications allows to have a robust and scalable archiving solution, it poses a number of challenges in terms of inventory, configuration, monitor and technical expertise:

- **Configuration** Once created for an application, each schema needs to be configured. The two main configuration parameters are the retention period, i.e. for how long the archived data should be kept, and the partitioning policy, i.e. how the tables storing the different archived values are partitioned to maximize query performance. The configuration of a schema may evolve over time depending on the user requirements, but also on the application characteristics, i.e. the number of signals to be archived and their frequency.
- **Monitor** All schemas are sharing a common resource, i.e. one of the database cluster, and the behaviour of one schema can penalize the performance experienced by the users of other schemas. It is therefore important to supervise each schema in order to be alerted when some queries are taking too much time or when a schema archives too much data compared to what was initially expected.
- **Inventory** To each WinCC OA application is associated a database schema with its own configuration such as the schema version, the data retention period or the archived data partitioning policy. Having an exhaustive list of schemas with their configuration is not necessarily trivial as applications have been developed and deployed over the years and may have been retired by the users without informing all interested parties.
- **Technical Expertise** While the Oracle database clusters themselves are managed by the CERN central database team, the schemas are under the responsibility of the team in charge of the industrial control systems whose main technical expertise is in building control systems but not necessarily in database technology.

All the challenges presented in this section can be tackled by a set of custom SQL queries which need to be applied to 250+ schemas taking into account the two different clusters as well as each node of the cluster. There is therefore the need for a high level tool to manage, configure and monitor the database schemas without having to deal with complex SQL queries.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA019

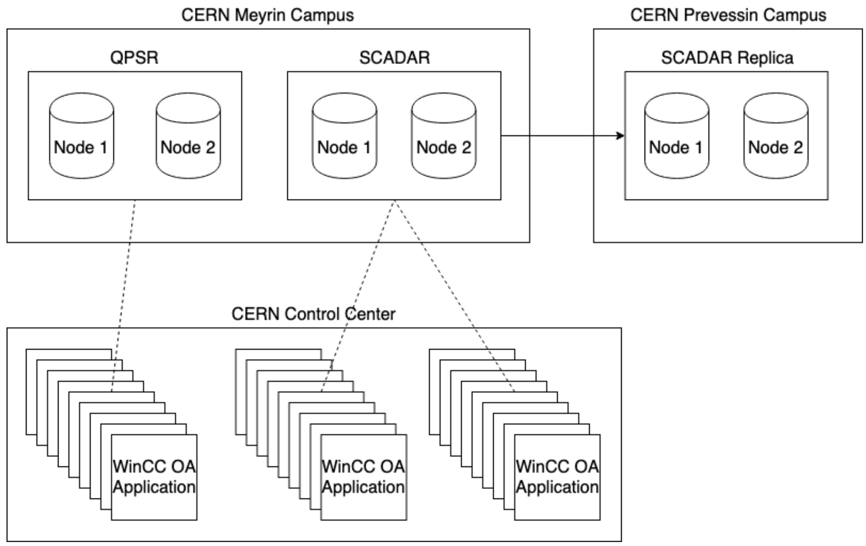

Figure 1: Archiving infrastructure used by CERN industrial control application based on WinCC OA.

Contribution

This paper presents MONARC, a WinCC OA application dedicated to the management, configuration and monitoring of all 250+ Oracle schemas used by the WinCC OA industrial control applications. The motivations for developing the tool using WinCC OA and the CERN standard frameworks are threefold: (1) in the context of industrial control applications, the Oracle database infrastructure can be considered as any other primary process with the usual requirements for managing, configuring and supervising the process, i.e. that the classical supervision tools such as trends, alarm and event screen are readily available; (2) the targeted users of the application are already familiar with WinCC OA and the learning curve is short; and (3) rely on the extensive WinCC OA expertise inside the group for the maintenance and extension of the application and at the same time benefiting of the integration into the CERN environment already developed for the standard WinCC OA applications.

The paper is organized in three parts: the first part presents the main functionalities of the application, while the second part digs into its architecture and implementation. The third part presents a set of use cases demonstrating the benefit of using the application.

MAIN FUNCTIONALITIES

Inventory

The first functionality of the application is to give an exhaustive list of all schemas used by WinCC OA applications. The list of schemas should contain all relevant metadata such as:

- the schema version: different versions of the WinCC OA software use a different schema version. Moreover, a set of PL/SQL packages are installed for each schema, those packages are themselves versioned.
- the archive groups: each WinCC OA application can define different archive groups representing data with different archiving requirements: e.g. alarms may need to be kept for a longer period compared to events.
- the retention policy settings: each schema configures its archive groups to keep their data for a specific period of time ranging from a couple of months to several years.
- the partition policy settings: each schema configures its archive groups to be partitioned either by hour, day, week or month. Partitioning policy has a direct impact of the queries and the overall database performance.

Configuration

From a configuration point of view, the main functionality is to allow users to easily set the retention and partition policy settings of the different archive groups and schemas. The configuration aspect addresses two main issues: hiding the complexities of the different SQL queries from the users and allowing to apply many configuration changes at once by simply selecting the different archive groups and/or schemas.

Archiving Contracts

In order to manage the increase of archived data and the number of schemas, there is a need for the so-called *archiving contract*. An *archiving contract* is a contract between the WinCC OA application and the archiving infrastructure. The contract is characterized by the number of archived signals, the amount of data archived per day and the overall amount of data stored in a schema. Such contracts are specified by the WinCC OA application responsible and any violation is detected by the application. The benefits of setting up such contracts is to make the users aware of the amount of data archived and preserve the overall performance of the database.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA019

Query Performance Monitoring

One key aspect of the archiving infrastructure is its performance. Two main aspects of performance are monitored: data retrieval which usually corresponds to a trend or search in an event screen and data insertion which corresponds to the data archiving per se. While the first aspect allows to detect long running queries experienced by users and take corrective actions to improve the situation (e.g. by adjusting the partition policy), the second aspect allows to detect issues related to the archiving infrastructure itself (e.g. disk failure).

ARCHITECTURE & IMPLEMENTATION

Device Model

As with the design of any supervision system, the first step is to identify the devices to be monitored and controlled. In the context of MONARC, five main device types are identified as illustrated in Figure 2. The device type *Schema* represent an Oracle schema as its name implies. It holds all the properties related to the schema itself, such as the schema version, the PL/SQL packages versions, the retention and partition policies, the number of archived signals, etc. A device of type *Schema* is associated to one and only one device of type *DbNode*, which itself is strongly associated to one and only *DbCluster*. The device type *RdbContract* is strongly associated to a device of type *Schema* and represents the archiving contract as presented in the previous section. Note that a schema can have several *RdbContract* devices (e.g. a contract on the number of alarms archived per day and one for the number of events). Finally, the device type *PeriodicQuery* is associated to a *DbNode*. A *PeriodicQuery* device is a device representing a SQL query, and its results, executed at regular interval on a specific node to detect for example long standing queries or blocking sessions. A *PeriodicQuery* device also holds information related to the action to be taken depending on the results of the query (e.g. trigger an alarm, send a notification, etc.).

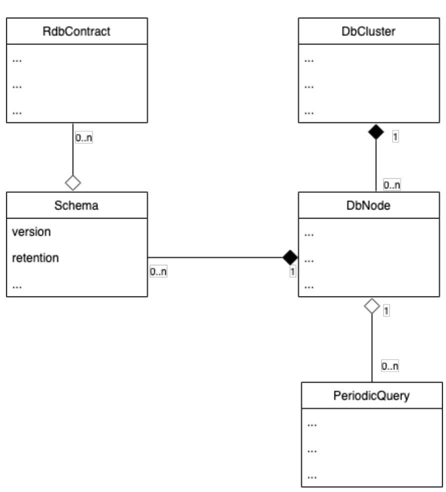

Figure 2: MONARC device model.

Data Acquisition

In a traditional SCADA system, the data acquisition processes are responsible for getting the data from the field devices into the supervision through industrial protocols such as Modbus, S7 or OPC UA. In the context of the MONARC application, since the field devices are Oracle databases, the data acquisition processes are simply processes executing SQL queries on the difference database nodes. There exists one data acquisition process per database cluster to handle the devices of type DbNodes, DbCluster and Schema. There exists another set of data acquisition processes per database node dedicated to the handling of PeriodQuery devices. The reason for having two set of device acquisition processes is that the first set is bound to a database cluster and handle the devices which are specific to a cluster (e.g. a device schema belong to a cluster but can be accessible by any node of the cluster), while the second set is specific to a database node (e.g. a periodic query needs to be always executed on the same node). Finally a third set of data acquisition processes are dedicated to the device RdbContract as their data is not retrieved from the Oracle database but from the ELK (Elasticsearch, Logstash, KIbana) stack which computes the different statistics of each schema daily [4].

Instantiation

The device model and data acquisition described in the two previous sections have been implemented and instantiated in a WinCC OA application to monitor the database infrastructure represented in Figure 1. The device model is implemented using the concept of WinCC OA datapoint type, DPT, with one DPT for each class identified in Figure 2. The attribute of the classes are implemented as datapoint elements, DPE, except for the name of each node which is directly implemented in the name of the datapoint instance itself. From an instantiation point of view, 250 schema devices are created along with 150 RDB contract devices and 10+ periodic query devices.

The data acquisition is implemented as WinCC OA CTRL manager which are background processes executing C-like code. Two CTRL managers are dedicated to the monitoring of the SCADAR and QPSR database cluster to retrieve the number of schemas and their characteristics such as the number of signal archived, the partition policies, etc. Once configured to be connected to one of the database cluster, the CTRL manager auto-discover all the WinCC OA schemas and automatically starts the monitoring of the schema. Similarly when a schema is deleted from the cluster, e.g. when an application is decommissioned, the CTRL manager marks the corresponding schemas as inactive. On top on these two CTRL managers, four other CTRL managers are dedicated to the execution of the periodic queries on each node of the clusters, i.e. two for the QPSR cluster and two for the SCADAR cluster. The implementation of the these six CTRL managers is relatively straightforward and mainly contains a set of threads which periodically executes the appropriate SQL queries. The periodic query CTRL manager

17th Int. Conf. on Acc. and Large Exp. Physics Control Systems ICALEPCS2019, New York, NY, USA JACoW Publishing
ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA019

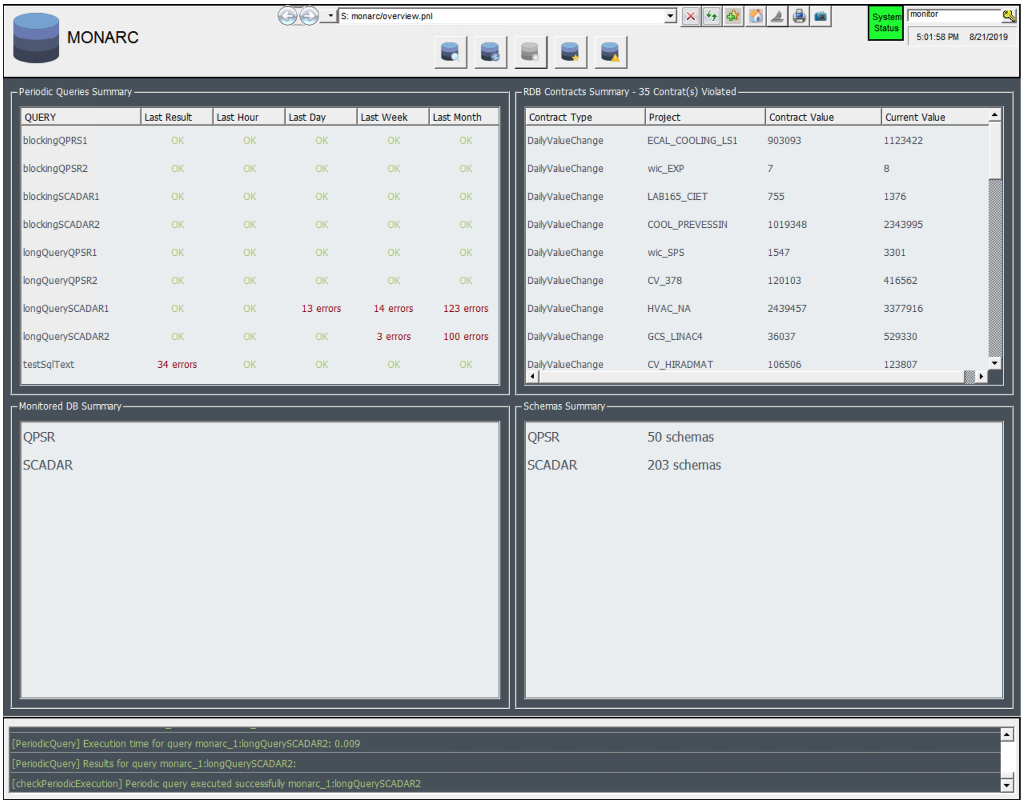

Figure 3: Screenshot of the MONARC application as used in production.

is completely configurable from the application HMI and allows the user to define which query should be executed, on which cluster node, its frequency, but also the action to be taken if the query result match a certain criteria. In total, 13 periodic queries are configured to detect long standing and blocking queries. Finally, an additional CTRL manager is dedicated to the evaluation of the RDB contract. It is executed once a day and the CTRL manager extracts the results from the ELK stack performing the SCADA statistics once a day through a REST API. A contract violation leads to an alarm in the application along with a notification to the administrator. The implementation of the manager is done such that a contract is considered violated if the number of archived data per day exceed a given threshold for 5 consecutive days.

Figure 3 presents a screenshot of the home page of the MONARC application which gives a quick overview of the number of database clusters monitored (bottom left) and the overall number of schemas monitored in each cluster (bottom right). It also summarizes all the RdbContracts which are currently violated (top right) as well as all periodic queries with abnormal results (top left).

CASE STUDIES

This section presents three case studies illustrating the benefits of using the application:

- **Schema Configuration Tuning:** the first aspect focuses on the user experience, i.e. the time it takes for a user of a SCADA application to retrieve data from the archiving infrastructure to either plot a trend or perform a post mortem analysis through an event screen. Thanks to the periodic query, monitoring any query lasting for more than 3 minutes, we have been able to better understand the reasons why some queries may take too long. These long standing queries were the results of a badly configured schema, i.e. the partitioning policy was not adapted to the amount of data archived, or the queries execution plan was not optimal due to bad statistics. Another reason is more fundamental to the Oracle database itself and is linked to the (very) large amount of data requested by the user: e.g. trending a pressure sensor for the last five years is a time consuming operation for the database. In these cases, which are usually not linked to the day to day operation of a SCADA system, the users were advised to use other more appropriate tools to perform these long term analysis. When the application was first put in production, there were an average of 20 queries per day extending more than 3 minutes. After one year and the different corrective action performed, the number of long standing queries per week is rarely exceeding 5.

- **Misconfigured archived signal:** since the application has been put in production more than a year ago, the users are requested to provide an estimate of the amount of data archived per day when a new application is cre-

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA019

ated. This parameter forms the basis of the archiving contract. For all other existing schemas, a pre-defined contract has been established such that the parameter value corresponds to the maximum value for 5 consecutive days over the last 24 months. The archiving contract violation notifications allowed to identify misconfigured signals in several cases. It also allowed to make the user conscious of the resources used and to not treat the archiving infrastructure as free resource.

- **Mass-configuration:** all monitored control systems are grouped by domain such as Cooling and Ventilation, Cryogenics, Machine Protection, Gas Systems, etc. Each domain has tens of SCADA applications which relates to tens of Oracle Schema. All the Oracle schemas for a domain are usually configured the same way, i.e. with the same retention policy and the same partitioning policy. A typical use case is the need to extend, or shrink, the retention period policy settings of application schemas belonging to a domain from e.g. three years to one year. Through the HMI of the MONARC application, this operation can be performed on all selected in a couple of clicks and verify that the changes are properly applied right away. It can also monitor that the new retention period policies are properly enforced by checking the overall size of the schemas are decreasing over the following days.

CONCLUSION

This paper presented the MONARC application. The application aims at providing a high level tool to monitor, configure and supervise the archiving infrastructure of more than 250 WinCC OA applications managed by the CERN section managing the industrial control systems. The paper details the architecture and the implementation and motivates why it has been developed based on the standard software stack to develop CERN industrial control supervision application, i.e. WinCC OA, JCOP and UNICOS.

While MONARC is in production for little more than a year, it already allowed us to have a close control of the performances of the archiving infrastructure and to take some corrective measures when needed (e.g. correcting the archiving settings of an application, enable statistics computation to optimize query execution plan). Several improvements and developments are foreseen. A first aspect is to generalize the application to make it applicable to other WinCC OA archiving infrastructures based on Oracle so that it can be used e.g. by the LHC experiments. Another important aspect of future development is to extend the application in order to support non Oracle backends as this will be the scenario in the near future in our control systems.

REFERENCES

[1] Siemens ETM, SIMATIC WinCC Open Architecture `https://www.winccoa.com/`

[2] O. Holme,M. Gonzalez-Berges,P. Golonka,S. Schmeling, “The JCOP Framework”, in *Proc. ICALEPCS’05*, Geneva, Switzerland, Oct. 2005, paper WE2.1-6O.

[3] E. Blanco, F. B. Bernard, P. Gayet, and H. Milcent, “UNICOS: An Open Framework”, in *Proc. ICALEPCS’09*, Kobe, Japan, Oct. 2009, paper THD003, pp. 910–912.

[4] J. A. G. Hamilton, M. Gonzalez-Berges, B. Schofield, and J-C. Tournier, “SCADA Statistics Monitoring Using the Elastic Stack (Elasticsearch, Logstash, Kibana)”, in *Proc. ICALEPCS’17*, Barcelona, Spain, Oct. 2017, pp. 451–455. doi:10.18429/JACoW-ICALEPCS2017-TUPHA034