Benchmarking Kudu and Oracle in typical WinCC OA historical data retrieval use cases

JULY 2018

AUTHOR(S):
Rishi Shah

SUPERVISOR(S):
Rafal Kulaga
Fernando Varela Rodriguez

CERN Openlab Summer Student Report 2018

ABSTRACT

WinCC Open Architecture is a toolkit for creating Supervisory Control and Data Acquisition (SCADA) applications, which is widely used at CERN. Hundreds of controls applications, both in the accelerator complex and the experiments are based on it, including domains such as cryogenics, machine protection equipment, vacuum systems, electrical network supervision, detector control and cooling and ventilation.

Storage and retrieval of historical process values and alarms is an essential feature of a SCADA system. WinCC OA allows this data to be archived in Oracle databases, which is the solution used in almost all production systems at CERN. Its robustness and scalability has been proven even in very challenging applications, such as Quench Protection System (QPS), in which number of archived changes per second exceeds 200,000.

To increase flexibility of archiving and further improve its performance by using new database technologies, CERN and ETM/Siemens started an R&D project to develop a Next Generation Archiver for WinCC OA. The main advantage of its architecture is that it allows simultaneous use of multiple data storage backends, which can be added by implementing a high-level interface based on ZeroMQ and Google Protocol Buffers. As this interface is not bound to WinCC OA, connecting external applications to the archives also becomes possible.

InfluxDB and Oracle are the two first databases supported by the Next Generation Archiver. Additionally, support for Apache Kudu is also considered due to its good balance between real-time and batch processing performance and simple integration with data analytics tools such as Apache Spark and SQL query engines such as Apache Impala.

The goal of this project is to prepare the tools that will help to compare the performance of Kudu and Impala with the current Oracle schema for various data retrieval scenarios. In order to streamline the benchmarks and make them more reliable and repeatable, two tools are developed: DataPump and QueryBenchmark. DataPump allows to transmit data from existing Oracle archives to Kudu, thus making sure that the tests are executed on the same, representative data sets. It also allows to measure the highest achievable write rate to Kudu. Readout performance of Oracle and Kudu is measured by QueryBenchmark, which executes sets of queries specified in the configuration file and writes results to report files, which can later be processed to generate performance statistics and plots.

TABLE OF CONTENTS

1. INTRODUCTION

The current WinCC OA archiving system only has the support of the Oracle database. It is a stable and proven solution used at CERN for many years. To enable the use of new storage technologies to archive historical data through pluggable backends, CERN together with ETM/Siemens decided to start the Next Generation Archiver (NGA) R&D project in the scope of Openlab. Currently, NGA has the support of Oracle and InfluxDB backend.

Apache Kudu strikes a good balance between real-time and batch data processing world. Kudu can easily integrate with Apache Impala and Apache Spark, to provide basic SQL commands to interact with SCADA systems and integration with modern data analytics platforms. Before providing support for Kudu in the NGA, its performance in typical data retrieval scenarios should be evaluated using real data archived by the current Oracle archiver.

The following diagram describes the workflow to transfer data from Oracle to Kudu and to run sets of test queries on those two storage solutions.

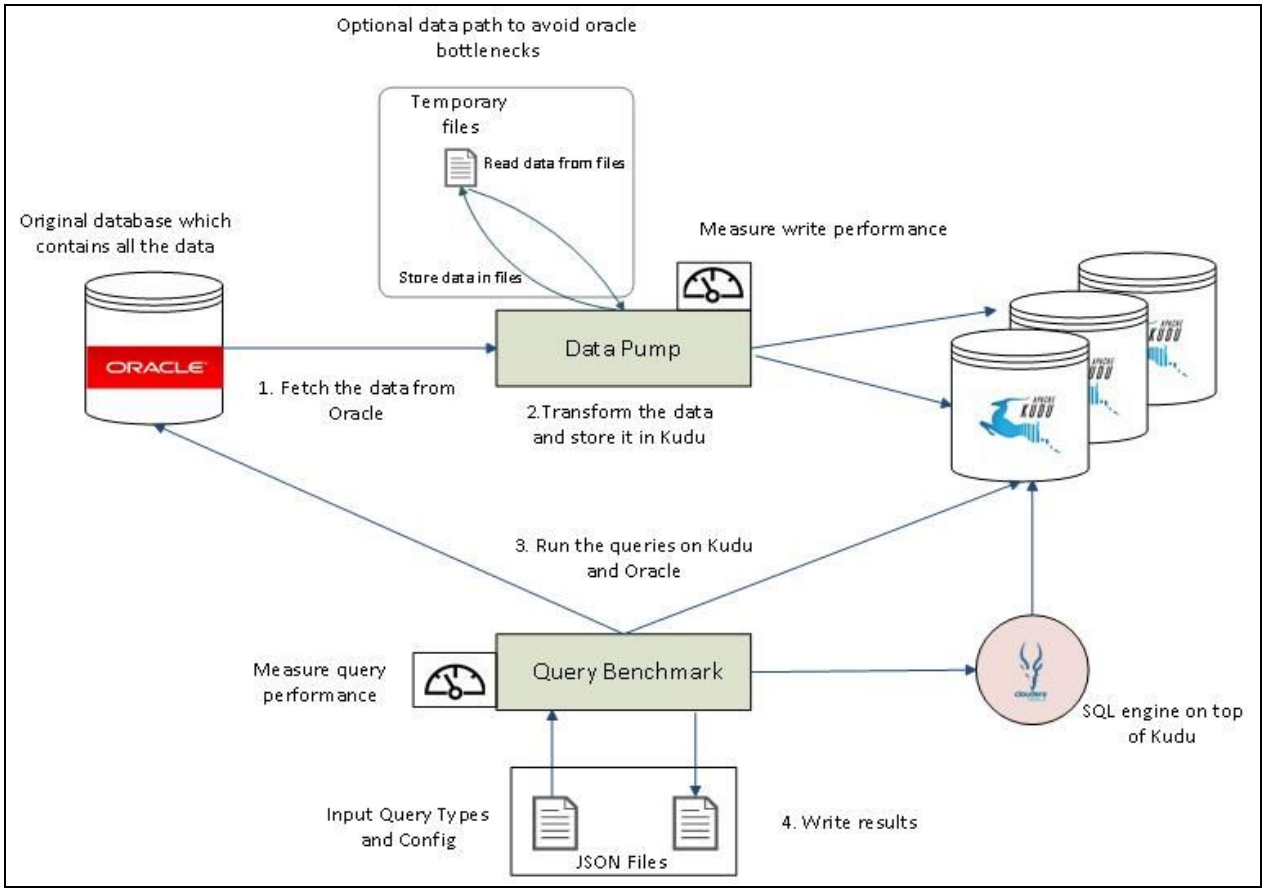

Figure 1: Workflow of the project

As shown in the diagram, the architecture is composed of databases and two tools developed in the scope of this project: DataPump and QueryBenchmark. DataPump enables data to be transferred from Oracle databases to Kudu, while measuring write performance. QueryBenchmark is designed to execute multiple queries on different databases to measure and compare the query performance.

There are two different workflows in this project. The first is to transmit the data from Oracle to Kudu using DataPump, and the second one is to run and benchmark the queries by QueryBenchmark. While transferring the data from Oracle to Kudu, DataPump has an option to use files as an intermediate step, in order to remove bottlenecks when measuring write performance .

Following are the key data retrieval use cases for the WinCC OA based archiving system:

- Trending - retrieve value changes of a single or a few datapoints (signals) over a time period, with or without last value before query start time and first value after query end time.
- Event Screen - retrieve value changes over a specified period of time, potentially for a very high number of datapoints (up to hundreds of thousands in certain applications).
- Event Replay - retrieve two types of data: state of all datapoints at the replay start time and all value changes from the replayed time period.

2. DATAPUMP TOOL

In order for performance tests to yield reliable results, they need to be performed on representative data sets, mimicking those produced by real applications. Tests data needs to contain enough datapoints with different frequencies of changes and various types (boolean, integer, floating point, string and arrays). Therefore, it was decided to use the data archived by real production WinCC OA systems as the base for performed benchmarks.

DataPump tool developed as a part of this project allows to read datapoint value changes from Oracle schemas supported by the Oracle Archiver, which is used in almost all production WinCC OA projects at CERN. Datapoint metadata is read and transmitted in two ways, to account for different schema designs:

- Entire metadata history can be read and written to Kudu in one transfer - used for schemas with separate datapoint metadata tables.
- Datapoint metadata is also transferred with each event - useful for Kudu schemas with denormalized tables.

Input arguments for the DataPump tool include transfer type (metadata, events or both), configuration of data source (Oracle or binary-serialized files) and data sink (Kudu or binary-serialized files). For event transfers, time bounds also need to be specified.

As the process of copying data from larger schemas to Kudu can take several days, DataPump creates progress logs that can be used to resume transfers if they are interrupted or if write errors occur for some event batches.

3. QUERYBENCHMARK TOOL

The primary purpose of the QueryBenchmark tool is to measure the performance of different potential Kudu schemas and compare them with performance of the currently used Oracle schema.

QueryBenchmark tool is written in C++. It has facilities to accept different types of queries through the JSON file and to execute each test on multiple Oracle and Kudu databases. Adjusting query test configuration parameters enables testing of all typical data retrieval scenarios.

Overall test procedure:
- Read and parse test parameters from the test description file.
- Construct and execute the queries on all the targets, measure their duration and record it in the test output file.

Key input parameters for the QueryBenchmark tool include:
- For queried datapoints:
  - Predefined data points:
    - Provide a list of data points through the JSON config.
  - Random datapoints:
    - Provide the total number of data points to query. QueryBenchmark will pick the random data points for the particular query to limit the impact of caching.
- For query time range:
  - Start time:
    - Define the start time through the JSON config. If it is not provided, then it will be chosen randomly.
  - End time:
    - Either define the specific end time through the JSON config or specify the time duration. QueryBenchmark can also calculate end time using specified duration.
- Should additional values be retrieved?
  - These additional values are the last value from before query start time and first value after query end time.
- Other parameters:
  - Connection parameters for Oracle and Kudu/Impala.

QueryBenchmark tool starts with two input parameters: path of the test description file and output file. Before making database connections, QueryBenchmark validates the

input config and throws the error if it is not in the correct format. At the end of the test it writes all the test results in a JSON document

4. FUTURE WORK

Generic architecture of the DataPump and QueryBenchmark tools enable the addition of new storage solutions by implementing a simple interface. Support for InfluxDB is planned to be added to the tools in the future in order to compare its performance with Kudu and Oracle.

i. DataPump

Current version of DataPump is single-threaded: consecutive batches of events are read from Oracle and written to Kudu sequentially. It is planned to add an option to parallelize this process by using multiple threads. In such configuration, different time portions are simultaneously read by multiple threads using many database connections and sent to Kudu through multiple sessions. Apart from simplifying the use of the tool (no need to run multiple instances of the tool to transfer large amounts of data in reasonable time), this would help to determine the highest achievable insertion rate for a single process, which is very important for WinCC OA systems with high write rates, such as the Quench Protection System (QPS).

Even though the primary purpose of the DataPump is to enable benchmarks, it could also be used as a data transfer mechanism in the process of migration from existing WinCC OA Oracle archives to new storage technologies. Should the need for such a tool arise, this potential use case will be further explored.

Archives of existing WinCC OA applications, such as CERN's Electrical Network SCADA (PSEN) are good candidates as sources of data for performance tests. Archive of PSEN contains values for around 400,000 datapoints, stored over a timespan of almost four years. Approximately 10,000,000 value changes are archived each day. As the application is typically engineered at least twice per month, datapoint metadata history contains many changes, further increasing the complexity of the queries. So far, only small subset of that data has been copied to Kudu. This process will have to be finished before the performance tests using QueryBenchmark are started.

ii. QueryBenchmark

QueryBenchmark tool is still under development and certain queries are not yet implemented for Kudu + Impala. Once it is finished and a full test schema is copied to Kudu, it will be used to measure the performance of the current Kudu schema by running several sets of queries, for trending, event screen and event replay use cases.

5. SUMMARY

Two tools developed during this project, DataPump and QueryBenchmark, are essential for performance evaluation of Kudu and optimization of the schema for WinCC OA archiving use case. With some additional development, they could also be used to transfer data from existing Oracle schemas to other types of storage, such as InfluxDB, during migration to the NextGen Archiver.