Database Futures Workshop

Monday, 29 May 2017 - Tuesday, 30 May 2017

CERN

Book of Abstracts

Contents

Implementations & Technologies / 2

A Conditions Data Management System for HEP Experiments

Authors: Paul James Laycock[1]; Dave Dykstra[2]; Andrea Formica[3]; Giacomo Govi[2]; Andreas Pfeiffer[1]; Shaun Roe[1]; Roland Sipos[4]

[1] *CERN*

[2] *Fermi National Accelerator Lab. (US)*

[3] *CEA/IRFU,Centre d'etude de Saclay Gif-sur-Yvette (FR)*

[4] *Eotvos Lorand University (HU)*

Corresponding Authors: andreas.pfeiffer@cern.ch, shaun.roe@cern.ch, paul.james.laycock@cern.ch, andrea.formica@cern.ch, giacomo.govi@cern.ch, dwd@fnal.gov, roland.sipos@cern.ch

Conditions data infrastructure for both ATLAS and CMS have to deal with the management of several Terabytes of data. Distributed computing access to this data requires particular care and attention to manage request-rates of up to several tens of kHz. Thanks to the large overlap in use cases and requirements, ATLAS and CMS have worked towards a common solution for conditions data management with the aim of using this design for data-taking in Run 3. In the meantime other experiments, including NA62, have expressed an interest in this cross-experiment initiative. For experiments with a smaller payload volume and complexity, there is particular interest in simplifying the payload storage.

The conditions data management model is implemented in a small set of relational database tables. A prototype access toolkit consisting of an intermediate web server has been implemented, using standard technologies available in the Java community. Access is provided through a set of REST services for which the API has been described in a generic way using standard Open API specifications, implemented in Swagger. Such a solution allows the automatic generation of client code and server stubs and further allows changes in the backend technology transparently. An important advantage of using a REST API for conditions access is the possibility of caching identical URLs, addressing one of the biggest challenges that large distributed computing solutions impose on conditions data access, avoiding direct DB access by means of standard web proxy solutions.

Requirements for run3&4 / 14

ALICE database requirements

Corresponding Author: costin.grigoras@cern.ch

Going beyond relational / 10

ATLAS ADC analytics

Authors: Mario Lassnig[1]; Ilija Vukotic[2]

[1] *CERN*

[2] *University of Chicago (US)*

Corresponding Authors: ilija.vukotic@cern.ch, mario.lassnig@cern.ch

The ATLAS Analytics effort is focused on creating systems which provide ATLAS Distributed Computing (ADC) with new capabilities for understanding distributed systems and overall operational performance. These capabilities include to correlate information from multiple systems (PanDA, Rucio, FTS, Dashboards, Tier0, PilotFactory, ...), predictive analytics to execute arbitrary data mining

or machine learning algorithms over raw and aggregated data, the ability to host new third party analytics services on a scalable compute platform, to satisfy a variety of use cases for different user roles for ad-hoc analytics, and to provide an open platform with documented collections and tools. ADC Analytics is hosted on two backends: ElasticSearch and HDFS. We use Jupyter and Zeppelin as web-based notebooks for advanced analytics, (dist-)Keras+Tensorflow for machine learning, and Pig+Spark for HDFS batch computation. With this talk we will detail the usage numbers and our projections and expectations for the future.

Going beyond relational / **11**

ATLAS configuration database evolution

Author: Leonidas Georgopoulos[1]

[1] *CERN*

Corresponding Author: leonidas.georgopoulos@cern.ch

We introduce a first working implementation of a distributed object store along with a network cache for distribution of information in the ATLAS Trigger and Data Acquisition (TDAQ) system primarily during system online configuration. The TDAQ system of the ATLAS detector at the Large Hadron Collider at CERN is a large distributed system at a range of a few tens thousand processes and servers providing data-taking functionality. During data taking runs, the different components of this distributed system need to be configured by accessing the configuration database. The latter is a middleware client based federated object oriented distributed database with XML persistent representation. We are planning to replace the latter with google-protobuf based representation and a simple network cache mechanism based on memcached. This greatly simplifies maintenance, operational system design, while augmenting flexibility of data representation, and extensibility to other data formats. Compatibility with existing code using the TDAQ configuration database is ensured by providing a legacy plugin and tools to transition the data. The current status of development is presented along with system performance evaluation in a controlled environment.

Requirements for run3&4 / **5**

ATLAS requirements for conditions DB for Run 3 and beyond

Author: Andrea Formica[1]

[1] *CEA/IRFU,Centre d'etude de Saclay Gif-sur-Yvette (FR)*

Corresponding Author: andrea.formica@cern.ch

Conditions data are in general non-event data varying with time. A particular subset is critical for physics data processing (detector status and configuration, run information, detector calibration and alignment,$\cdots$). Part of these data is used instead for monitoring of the detector.
Atlas has been using the COOL framework as a generic abstraction layer to deal with conditions data during Run1 and Run2. Online systems are using also direct CORAL connections for accessing dedicated relational tables.
The main relational backend for the storage of conditions data and related metadata is Oracle. In this talk we summarise the present usage of Oracle in conditions data infrastructure and present the future ideas for the evolution of the infrastructure for the management of conditions data, with simple estimates for the data volumes expected in future runs. We will also present an overview of the software requirements for Run3 in terms of database access.

Requirements for run3&4 / **19**

CMS Database Overview

Corresponding Author: giacomo.govi@cern.ch

"The CMS experiment relies on Relational Databases to store essential data for the most important production operations. Several subsystems critical for data taking, data processing and daily operation have been designed and optimised for a Relational storage. Their variety in terms of architecture and complexity, and the specific needs of the experiment organisation has required the deployment of several database services. These services implies a reliable support in terms of maintenance and expertise. In this presentation, we will provide and overview of the systems using Oracle as a storage backend.
In specific use cases, the storage with technologies different from RDBMS has been evaluated."

Requirements for run3&4 / **37**

CMS Database Overview

Author: Giacomo Govi[1]

[1] *Fermi National Accelerator Lab. (US)*

Corresponding Author: giacomo.govi@cern.ch

"The CMS experiment relies on Relational Databases to store essential data for the most important production operations. Several subsystems critical for data taking, data processing and daily operation have been designed and optimised for a Relational storage. Their variety in terms of architecture and complexity, and the specific needs of the experiment organisation has required the deployment of several database services. These services implies a reliable support in terms of maintenance and expertise. In this presentation, we will provide and overview of the systems using Oracle as a storage backend.
In specific use cases, the storage with technologies different from RDBMS has been evaluated."

Going beyond relational / **38**

ClickHouse - realtime analytical DBMS

In my talk, I will present brief history and motivation behind ClickHouse
- distributed analytical DBMS, designed for maximum query execution speed.
I will tell about architectural choices and usage scenarios.
Then I will show typical and some unusual applications of ClickHouse in companies around the world; Also I plan to highlight available tools, integrations and ClickHouse community.

Implementations & Technologies / **26**

Database on Demand status and future

Corresponding Author: ignacio.coterillo.coz@cern.ch

Going beyond relational / **33**

ElasticSearch

Corresponding Author: ulrich.schwickerath@cern.ch

Requirements for run3&4 / **6**

Evolution of ATLAS ADC relational databases

Author: Gancho Dimitrov[1]

[1] *CERN*

Corresponding Author: gancho.dimitrov@cern.ch

The ATLAS Distributed Computing (ADC) project delivers production tools and services for ATLAS offline activities such as data placement and data processing on the Grid. The system has been capable of sustaining with high efficiency the needed computing activities during the first and in the ongoing second run of LHC data taking.
Databases are a vital part of the whole ADC system. The Oracle Relational Database Management System (RDBMS) has been addressing a majority of the ADC database requirements for many years. Much expertise was gained which could be used as a good foundation for next generations PanDA (Production ANd Distributed Analysis) and DDM (Distributed Data Management) systems.
By extrapolating of the current data volume in the roadmap to Run4, one could expect Grid operations on exabytes of data (factor ten from now). The corresponding catalog entries in the database and the rate of DB data change and read operations will greatly depend on the used object granularity: containers, datasets, files or events plus having event-level analysis workflow instead of file-level. It is expected that such evolution would require larger database capabilities and capacity in the medium-term future.

Requirements for run3&4 / **7**

Evolution of the ATLAS Metadata architecture and databases

Authors: Davide Costanzo[1]; Borut Paul Kersevan[2]

[1] *University of Sheffield (GB)*

[2] *Jozef Stefan Institute (SI)*

Corresponding Authors: borut.kersevan@cern.ch, davide.costanzo@cern.ch

In the future of ATLAS the event should be the atomic information for metadata. Events could be either from data or from Monte Carlo, and different "representation" of the event would be available to reflect the processing stage under consideration. Eg Raw, Analysis Object Data (AOD), or derived AOD. The metadata should carry the provenance information of the event, as well as the logical location of the event itself.
Collections are built from events with the same characteristics, e.g. from the same luminosity block or generated with the same event generator configuration. Collections will carry all the metadata with physics content and production system configurations. The metadata at the collection level will be dynamical, and stored into a "whiteboard". The information of the whiteboard will change

over time as collections are extended, physics information is improved/added, or the collection is declared obsolete. A versioning mechanism will be needed to manage this evolution.
Finally each physics analysis would have metadata corresponding to the collections in use. The analysers access the metadata information of the collections in use to find the physics details and may further annotate the whiteboard with the analysis specifics. The analysis level whiteboard will also contain information such as the software used to carry out the analysis, information on physics ntuples/files produced from the centrally managed storage, personnel information (list of authors, editorial board, etc)

Going beyond relational / **36**

Kafka

Corresponding Author: lionel.cons@cern.ch

Implementations & Technologies / **4**

NXCALS Data Extraction and Analysis with Apache Spark

Author: Nikolay Tsvetkov[1]

Co-authors: Jakub Wozniak [1]; Marcin Sobieszek [1]

[1] *CERN*

Corresponding Authors: n.tsvetkov@cern.ch, marcin.sobieszek@cern.ch, jakub.wozniak@cern.ch

The CERN Accelerator Logging Service (CALS) was designed in 2001, and has been in production for 14 years. It is a mission-critical service for the operation of the LHC (Large Hadron Collider).

CALS uses an Oracle database for storage of technical accelerator data and persists approx 0.75 petabytes of data coming from more than 1.5 million pre-defined signals. These signals represent data related to CERN's core infrastructure such as electricity, cooling and ventilation, industrial data such as cryogenics, vacuum and control devices, and beam-related data such as beam positions, currents, losses, etc.

Over time, the scope of the service and the data mining requirements have evolved significantly resulting in the current infrastructure slowly reaching hard scalability limits. In order to address this, a next generation Hadoop based Logging System (NXCALS) is currently being developed. This new system provides a Data Analysis Platform based on Apache Spark.

This presentation will briefly introduce the background of the Logging Service, give a general overview of the new NXCALS system, and then go into more details regarding the use of Apache Spark.

Implementations & Technologies / **25**

Next generation Archiver

Corresponding Author: piotr.golonka@cern.ch

Implementations & Technologies / **32**

Next generation for Post Mortem event storage and analysis

Authors: Maciej Piotr Pocwierz[1]; Serhiy Boychenko[2]; Tiago Martins Ribeiro[3]; Kamil Henryk Krol[3]; Marc-Antoine Galilee[3]; Janet Que Chi Do[4]; Jean-Christophe Garnier[3]; Anita Stanisz[5]; Maciej Krzysztof Osinski[3]; Markus Zerlauth[3]; Zinour Charifoulline[3]

[1] *Warsaw University of Technology (PL)*

[2] *Universidade de Coimbra (PT)*

[3] *CERN*

[4] *University of Applied Sciences (DE)*

[5] *AGH University of Science and Technology (PL)*

Corresponding Authors: marc-antoine.galilee@cern.ch, janet.que.chi.do@cern.ch, serhiy.boychenko@cern.ch, zinour.charifoulline@cern.ch, kamil.krol@cern.ch, anita.stanisz@cern.ch, maciej.osinski@cern.ch, maciej.piotr.pocwierz@cern.ch, tiago.martins.ribeiro@cern.ch, jean-christophe.garnier@cern.ch, markus.zerlauth@cern.ch

The Post Mortem was designed almost a decade ago to enable the collection and the analysis of high-resolution, transient data recordings of relevant events, such as beam dumps in the LHC accelerator. Since then, the storage has been constantly evolving both to accommodate larger datasets and to satisfy new requirements and use-cases for the LHC but also first machines in the injector complex. The operational experience allowed to identify some of the drawbacks of the initially designed solution which, in order to be solved in an efficient way, will require substantial changes of the currently deployed infrastructure.

This contribution summarizes the recent work and R&D towards the definition of the next generation Post Mortem storage architecture, in line with modern data storage and processing systems which provide solutions to the major limitations of the current deployment and enable an easier integration of future use cases. The proposed architecture provides in addition a better integration with the next generation CALS storage, serving the users with the most accurate data in a more transparent way while replying to the determinism in terms of response time imposed by certain LHC use-cases.

Requirements for run3&4 / **1**

Outlook for Accelerator Databases

Author: Chris Roderick[1]

[1] *CERN*

Corresponding Author: chris.roderick@cern.ch

The aim of this presentation is to give an overview of the foreseen database needs related to accelerator operation in the coming years.

Implementations & Technologies / **8**

Relational database evolution in ATLAS

Authors: Elizabeth Gallas[1]; Gancho Dimitrov[2]

[1] *University of Oxford (GB)*

[2] *CERN*

Corresponding Authors: gancho.dimitrov@cern.ch, elizabeth.gallas@physics.ox.ac.uk

Relational databases are critical backend storage for many systems in ATLAS (both online, offline, and on the grid), storing essential data for the processing of past and current data as well as support daily operations. These systems have been refined over time into robust applications optimized and provisioned for established use cases. Relational storage is well suited for many of these systems and so it is logical to assume that their support must be continued. In this talk, we summarize the current usage of relational databases in ATLAS and project into the future the needed data volumes to support their future operations. In addition, relational technologies have evolved over time, offering ever-expanding functionality as well as supporting increasing data volumes. These factors, along with the suitability of relational storage for many applications, will result in an increasing need for new applications utilizing this form of storage.

Implementations & Technologies / 3

Status and Plans of the CMS Big Data Project

Authors: Jim Pivarski[1]; Kacper Surdy[2]; Luca Canali[2]; Maria Girone[2]; Matteo Cremonesi[3]; Oliver Gutsche[3]; Vaggelis Motesnitsalis[2]; Viktor Khristenko[4]

[1] *Princeton University*

[2] *CERN*

[3] *Fermi National Accelerator Lab. (US)*

[4] *The University of Iowa (US)*

Corresponding Authors: luca.canali@cern.ch, oliver.gutsche@cern.ch, kacper.surdy@cern.ch, maria.girone@cern.ch, pivarski@fnal.gov, matteo.cremonesi@cern.ch, victor-khristenko@uiowa.edu, vaggelis.motesnitsalis@cern.ch

Experimental Particle Physics has been at the forefront of analyzing the world's largest datasets for decades. The HEP community was among the first to develop suitable software and computing tools for this task. In recent times, new toolkits and systems for distributed data processing, collectively called "Big Data"technologies have emerged from industry and open source projects to support the analysis of Petabyte and Exabyte datasets in industry. While the principles of data analysis in HEP have not changed (filtering and transforming experiment-specific data formats), these new technologies use different approaches and tools, promising a fresh look at analysis of very large datasets that could potentially reduce the time-to-physics with increased interactivity. Moreover these new tools are typically actively developed by large communities, often profiting of industry resources, and under open source licensing. These factors result in a boost for adoption and maturity of the tools and for the communities supporting them, at the same time helping in reducing the cost of ownership for the end-users. In this talk, we are presenting studies of using Apache Spark for end user data analysis. This could inform the discussion of future database and analytics needs of the community.
CMS is working together with CERN openlab and Intel on the CMS Big Data Reduction Facility. The goal is to reduce 1 PB of official CMS data to 1 TB of ntuple output for analysis. We are presenting the progress of this 2-year project with first results of scaling up Spark-based HEP analysis. We are also presenting studies on using Apache Spark for a CMS Dark Matter physics search, investigating Spark' s feasibility, usability and performance compared to the traditional ROOT-based analysis.

35

Summary & Discussion

Corresponding Author: eric.grancher@cern.ch

Going beyond relational / **9**

The ATLAS EventIndex: possible evolution towards an Event Whiteboard

Author: Dario Barberis[1]

[1] *Università e INFN Genova (IT)*

Corresponding Author: dario.barberis@cern.ch

The ATLAS EventIndex was designed during LS2 to satisfy a small but important number of use cases, primarily event picking. Its contents and storage architecture were tailored to the primary use cases, favouring performance and robustness over the possibility to expand its scope. The EventIndex is in operation since the start of Run2 and shows satisfactory performance for event picking and also for additional use cases like production consistency checks and trigger and derivation dataset overlap counts. Discussions are starting about the possibility to increment its contents and extend its scope for Run3. The general idea is to have an "Event Whiteboard"where the initial metadata can be supplemented with information produced during event processing tasks, like references to the algorithms that were run and their results, especially concerning event selections. With this tool one could construct "virtual datasets", consisting only of the lists of events that satisfy some group of selection criteria, avoid creating many copies of the same events on storage, and run jobs to extract the main parameters for histogramming and fitting only at the end of the analysis tasks. Of course a thorough study of the available technologies has to be launched first, to make sure that this system can sustain the expected i/o rates.

Requirements for run3&4 / **18**

The LHCb experiment offline databases and LHCb requirement for RUN3

Corresponding Author: luca.tomassetti@cern.ch

Going beyond relational / **30**

Time Series databases

Corresponding Author: antonio.romero.marin@cern.ch

Welcome / **34**

Welcome & Introduction

Corresponding Authors: eva.dafonte.perez@cern.ch, eric.grancher@cern.ch