Integration of the EventIndex with other ATLAS systems

CHEP2015 – Computing in High Energy and Nuclear Physics 2015, 13-17 April 2015, Okinawa, Japan

D. Barberis[1], S.E. Cárdenas Zárate[2], E.J. Gallas[3], F. Prokoshin[2*]
on behalf of the ATLAS Collaboration
[1] Università di Genova and INFN, Genova, Italy, [2] Universidad Técnica Federico Santa Maria, Valparaíso, Chile,
[3] Department of Physics, Oxford University, Oxford, United Kingdom
*Corresponding author

The ATLAS EventIndex System[1,2] stores information about every processed event in ATLAS. Its storage infrastructure is based on Hadoop[3].

This presentation describes auxiliary systems deployed to facilitate efficient data collection, optimize storage, and enhance information in user interfaces.

Run Metadata

The information transferred per event from event processing jobs is encoded to minimize network transmitted data volume. At the destination site, this information is decoded and organized so that storage in Hadoop is optimized for user access. Information needed for decoding (such as the correspondence of trigger bits to trigger names) and for grouping sets of events corresponding to those found in user datasets is obtained from run-wise metadata collected in the COMA database[4].

Run Metadata from COMA

Trigger Menus

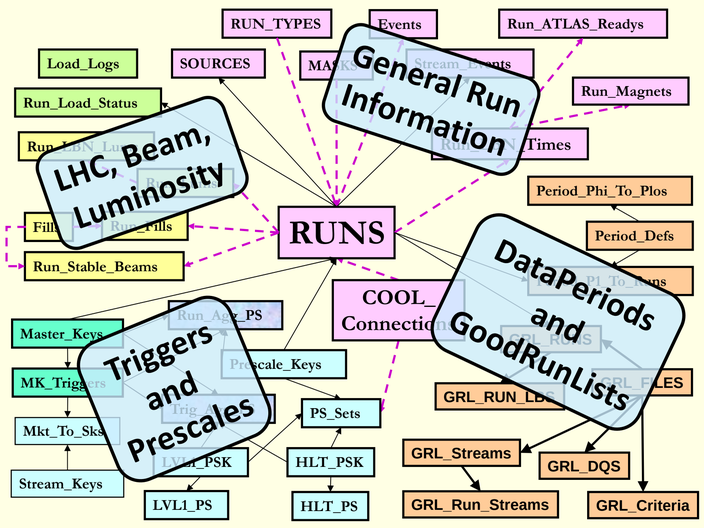
- ATLAS data taking evolves with time in terms of the deployed triggers which select the events recorded for offline processing. There is a fixed set of triggers (the trigger menu) used for each Run (typically a few hours of data taking)
  - Each Trigger Menu contains hundreds of multi-level trigger chains, each identified by a name and assigned a unique bit.
  - The EventIndex system gets the trigger name to trigger bit mapping from the COMA system

Data Periods

- A Data Period is a set of ATLAS Runs grouped for a purpose. Periods are defined by ATLAS Data Preparation Experts and stored in the COMA system.
  - Periods are used in many stages of ATLAS data processing, assessment, and selection.
  - User event samples for analysis are generally based on Data Periods.
  - EventIndex can get Data Period information from COMA to enhance user queries.

Other COMA information

- Other metadata from COMA may be incorporated into the EventIndex to facilitate user searches, such as Good Run Lists, which select partial runs during Data Periods which satisfy Data Quality criteria.

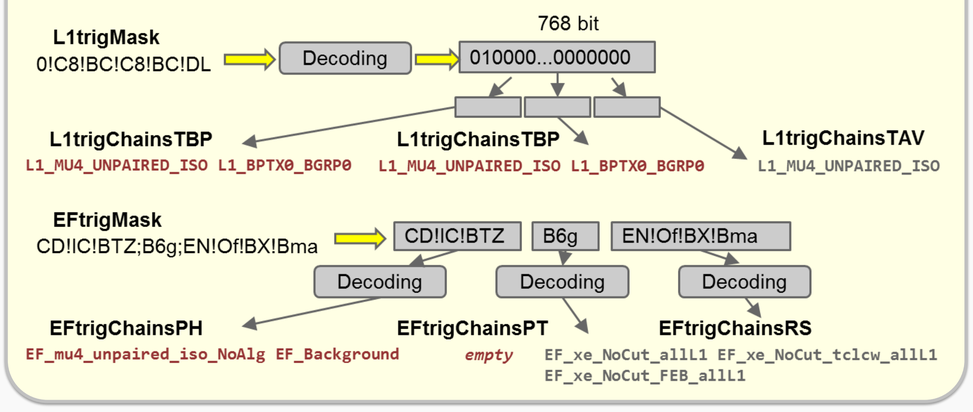


Other COMA information

- Other metadata from COMA may be incorporated into the EventIndex to facilitate user searches, such as Good Run Lists, which select partial runs during Data Periods which satisfy Data Quality criteria.

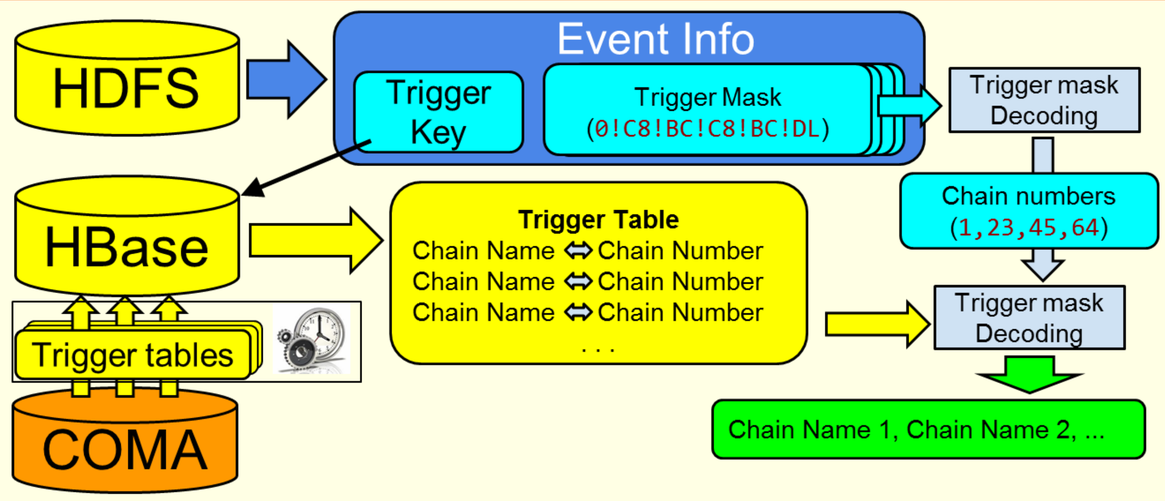
Trigger information in data

- The trigger decision is stored in data files as bitmasks. If a bit of the mask is set, this means that one of the sets of trigger conditions (Trigger chains) has been passed in this event. A position of this bit in mask (chain counter) indicates which chains have passed.
- Depending on the trigger configuration, the same chain counter may correspond to a different chain name. The relation is uniquely defined by the trigger table corresponding to the SMK
- For the Level 1 trigger mask has three 256–bit words (Before Prescale, after Prescale, After Veto). For HLT (L2 and EF) there are 3 variable length parts – Physics, PassThrough and Resurrected.



Decoding trigger information using trigger tables replicated from COMA

- Trigger tables, with information on the relation between chain names and chain counters for a specific interval are replicated from COMA to HBase to maximize performance.
- Trigger masks from the event record are decoded, chain counters are converted to chain names.
- The list of trigger chains, obtained after decoding are then stored in updated Event Records. These are available for search, skim and count operations. This format is convenient for analysis and can be easily indexed by Hadoop tools.
- Trigger tables can appear in COMA some time after Event Data. In that case, trigger chains information can be added later. This can also be done in case some issues will be found in the event or conditions data.
- The software developed for the import of trigger related information project will be used for other tasks of the EventIndex project: data period information support, data integrity verification, etc…

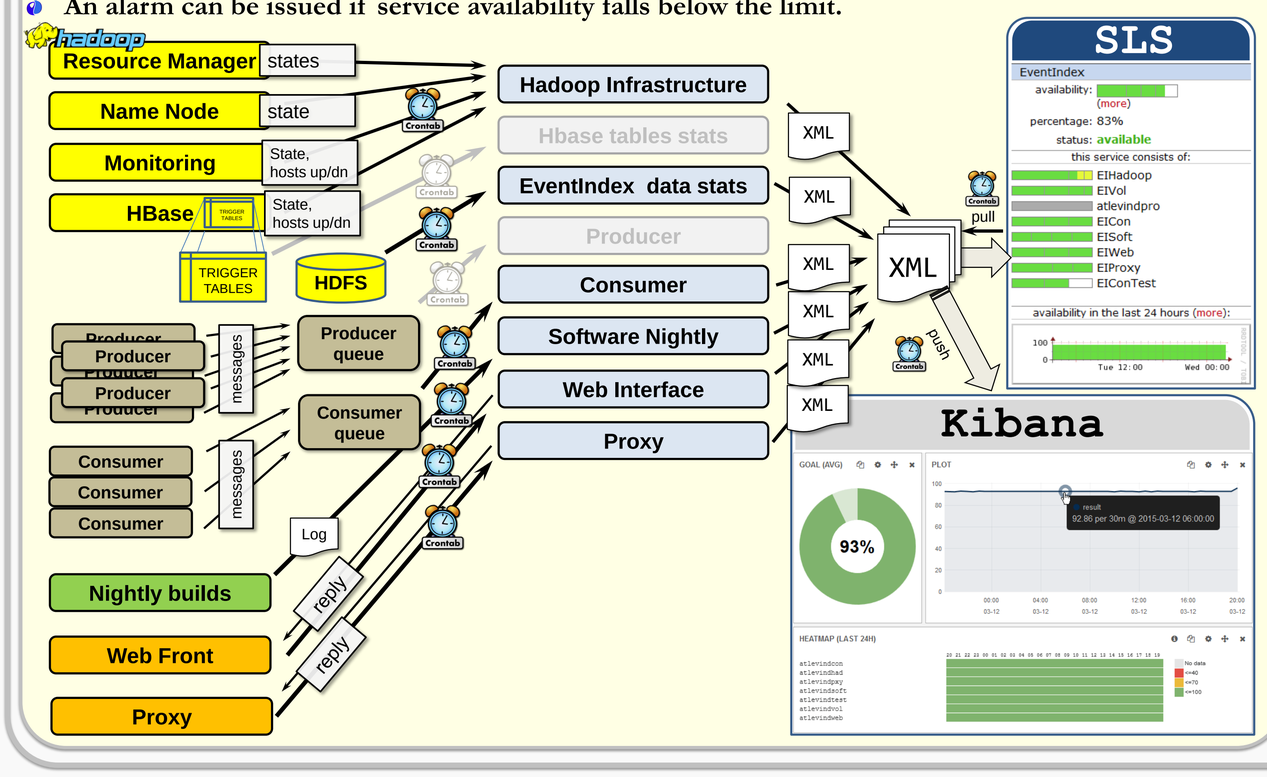


System Monitoring

EventIndex operation depends on a number of components that must be monitored constantly to assess processing rates, recognize bottlenecks and identify component failures. We use CERN Monitoring infrastructure to compose dynamic interfaces which display important information on EI component status.

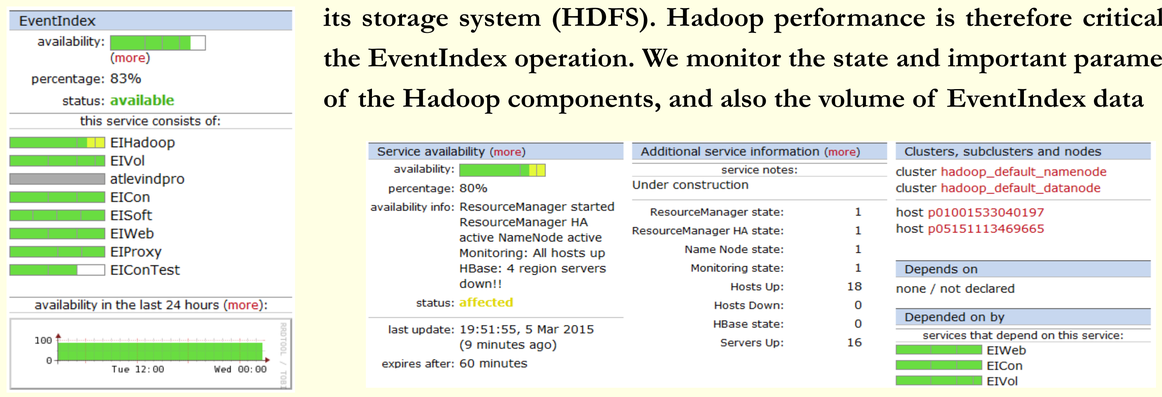
EventIndex Monitoring

- Monitoring tasks are launched by scheduler (acron) on one of the EventIndex computers, and gather component information by making CURL requests, analyzing logs, collecting messages… The way information is collected is very different for each of the components.
- Collected data are then organized into XML files and then made available for both old and new service monitoring systems
- "Old": SLS XML files are shared via a tomcat web server and periodically collected by the service.
- "New": Kibana XML files are pushed to the central service via CURL
- An alarm can be issued if service availability falls below the limit.



Monitoring of EventIndex components

- EventIndex data is managed by the CERN Hadoop cluster and resides on its storage system (HDFS). Hadoop performance is therefore critical for the EventIndex operation. We monitor the state and important parameters of the Hadoop components, and also the volume of EventIndex data



- We monitor Producer and Consumer messages sent to the queue. By analyzing information on the number of messages and data volume sent and received, we can ensure that there is no loss of information due to the overloaded consumer.
- The result of EventIndex software nightly builds is monitored for errors
- We check Web services and proxy used for access from outside CERN

References

[1] D.Barberis et al., **"The ATLAS EventIndex: an event catalogue for experiments collecting large amounts of data"**, 20th International Conference on Computing in High Energy and Nuclear Physics (CHEP2013), Amsterdam, The Netherlands, 14-18 Oct 2013. Published in J.Phys.Conf.Ser.513:042002,2014.

[2] D.Barberis et al., **"The future of event-level information repositories, indexing, and selection in ATLAS** ", CHEP2013, Amsterdam, The Netherlands, 14-18 Oct 2013. Published in J.Phys.Conf.Ser.513:042009,2014.

[3] **Hadoop**: see http://hadoop.apache.org, http://hbase.apache.org and http://hive.apache.org

[4] E J Gallas et al., **"Conditions and configuration metadata for the ATLAS experiment"**, CHEP2012, New York, United States, 21-25 May 2013. Published in J.Phys.Conf.Ser.396:052033,2012.