NEW DEVELOPMENTS FOR THE HDB++ TANGO ARCHIVING SYSTEM

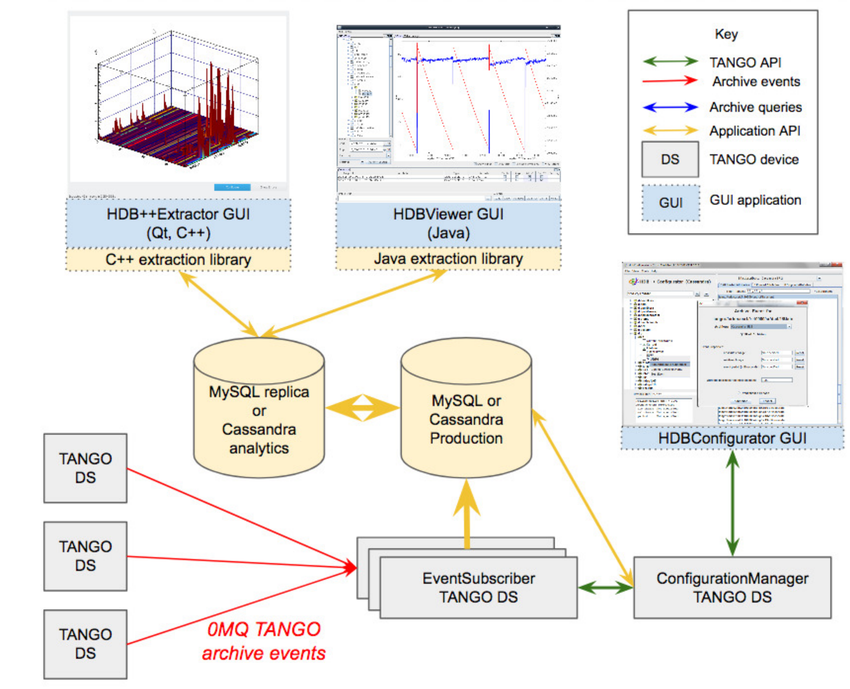
L. Pivetta, G. Scalamera, G. Strangolino, L. Zambon - ELETTRA-Sincrotrone Trieste
R. Bourtembourg, J.L. Pons, P. Verdier - ESRF
S. Rubio-Manrique - ALBA

THUPA166

TANGO HDB++ is a high performance event-driven archiving system which stores data with micro-second resolution timestamps, using archivers written in C++. HDB++ currently supports MySQL and Apache Cassandra back-ends but could be easily extended to support additional back-ends. Since the initial release many improvements and new features have been added to the HDB++. In addition to bug-fixes and optimizations, the support for context-based archiving allows to define an archiving strategy for each attribute, specifying when it has to be archived or not. Temporary archiving is supported by means of a time-to-live parameter, available on a per-attribute basis. The Cassandra back-end is using Cassandra TTL native feature underneath to implement the time-to-live. With dynamic loading of specific libraries switching back-ends can be done on-the-fly and is as simple as changing a property. Partition and maintenance scripts are now available for HDB++ and MySQL. The HDB++ tools, such as extraction libraries and GUIs, followed HDB++ evolution to help the user to take full advantage of the new features.

HDB++

The HDB++ TANGO archiving system exploits the TANGO archive event, that can be triggered on threshold comparison end/or periodic basis. The HDB++ architecture is fully event based; therefore, a part of HDB++ setup consists of conveniently configure TANGO devices to send events as required. The TANGO archiving system consists of two main components, namely the EventSubscriber TANGO device, or archiving engine, and the ConfigurationManager TANGO device, that simplifies archiving configuration and management.



Back-end configuration

Configuration for different back-ends is supported via the LibConfiguration property; a specific set of key=value entries can be specified for each back-end. The libraries implementing the back-end are now dynamically loaded.

Time to live

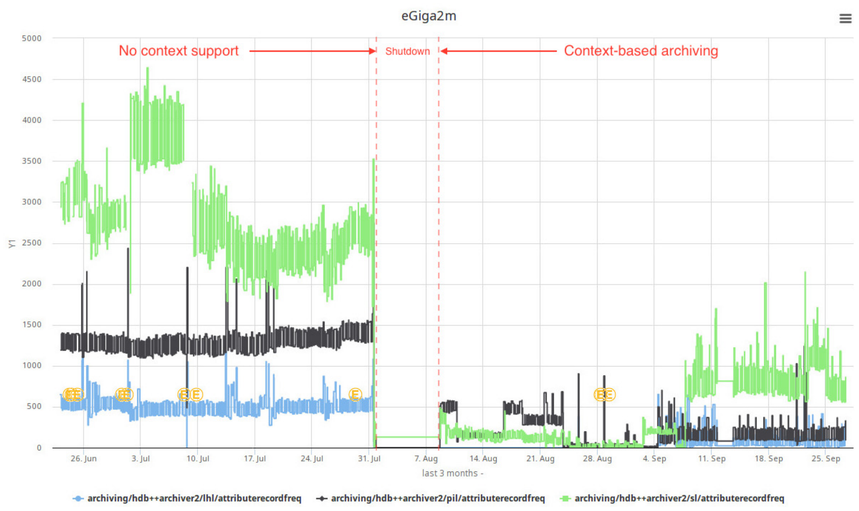
Support for temporary archiving has been added to the HDB++. A time-to-live parameter, expressed in hours, can be defined on per-attribute basis, and is saved as part of the attribute archiving configuration in the AttributeList device property.

tango://srv-tango-srf.fcs.elettra.trieste.it:20000/eos/climate/state; strategy=RUN|SHUTDOWN; ttl=8760
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/pil/laser/evops.01/state; strategy=RUN; ttl=24

Context-based archiving

The support for context-based archiving allows to define an archiving strategy for each attribute. A strategy is the list of contexts for which the attribute has to be archived. When an EventSubscriber is set to a context, only attributes that have this specific context in their strategy are archived, and all the remaining attributes are automatically stopped. The strategy is saved in the AttributeList device property:

tango://srv-tango-srf.fcs.elettra.trieste.it:20000/eos/climate/18b20.01/state; strategy=RUN|SHUTDOWN; ttl=8760
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/pil/laser/evops.01/state; strategy=RUN; ttl=24



Archiving rates decrease with context-based archiving for FERMI laser systems. Graph plotted with eGiga2m web based tool.

Disk usage

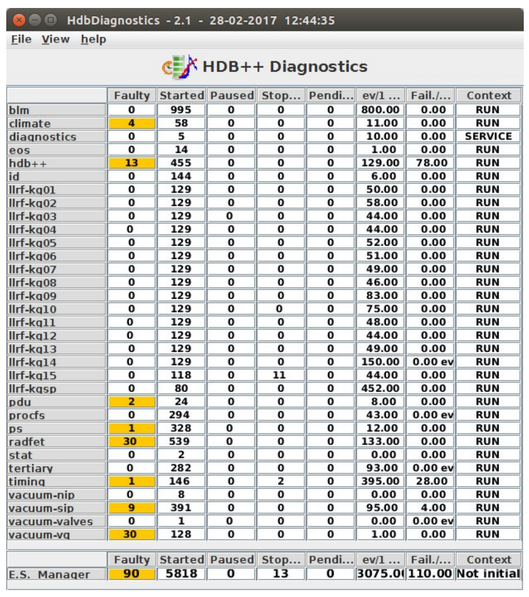
To reduce disk usage, when required, HDB++ can be configured with a reduced MySQL schema. Some of the columns of the tables used to support TANGO data types have been made optional: two of the three timestamp columns, e.g. the TANGO event timestamp and the insertion timestamp, the attribute quality and the error description carried by the event
The back-end abstraction library can be configured to ignore the optional columns (lightschema=true) or can autodetect the existing columns in the tables.

HDB++ at ELETTRA/FERMI

EventSubscriber instances	10
ConfigurationManager instances	1
Nr. of attributes archived	~2200
Average events/minute rate	~1200

EventSubscriber instances	39
ConfigurationManager instances	3
Nr. of attributes archived	~8100
Average events/minute rate	~7300
Archive disk size	~400GB

HDB++ diagnostics GUI, FERMI.



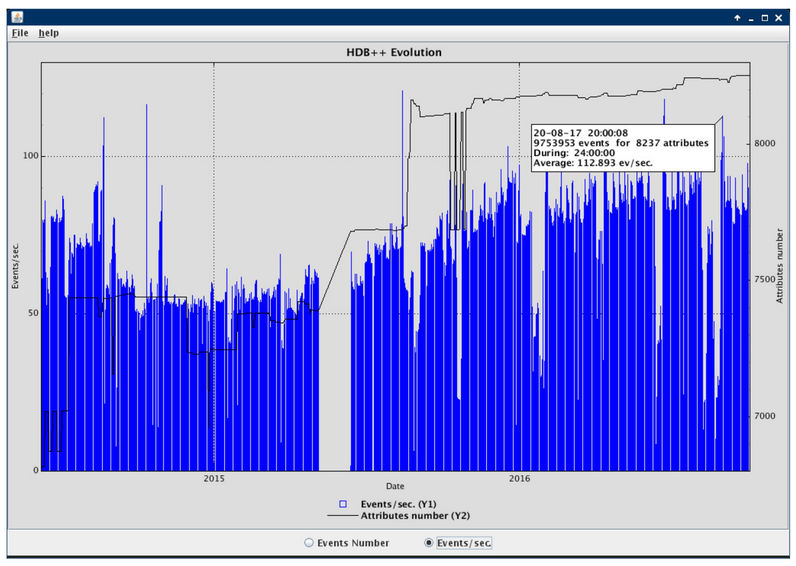
	Faulty	Started	Paused	Stop...	Pendi...	ev/1 ...	Fail./...	Context
blm	0	995	0	0	0	800.00	0.00	RUN
climate	4	58	0	0	0	11.00	0.00	RUN
diagnostics	0	5	0	0	0	10.00	0.00	SERVICE
eos	0	14	0	0	0	1.00	0.00	RUN
hdb++	13	455	0	0	0	129.00	78.00	RUN
id	0	144	0	0	0	6.00	0.00	RUN
llrf-kq01	0	129	0	0	0	50.00	0.00	RUN
llrf-kq02	0	129	0	0	0	58.00	0.00	RUN
llrf-kq03	0	129	0	0	0	44.00	0.00	RUN
llrf-kq04	0	129	0	0	0	44.00	0.00	RUN
llrf-kq05	0	129	0	0	0	52.00	0.00	RUN
llrf-kq06	0	129	0	0	0	51.00	0.00	RUN
llrf-kq07	0	129	0	0	0	49.00	0.00	RUN
llrf-kq08	0	129	0	0	0	46.00	0.00	RUN
llrf-kq09	0	129	0	0	0	83.00	0.00	RUN
llrf-kq10	0	129	0	0	0	75.00	0.00	RUN
llrf-kq11	0	129	0	0	0	48.00	0.00	RUN
llrf-kq12	0	129	0	0	0	44.00	0.00	RUN
llrf-kq13	0	129	0	0	0	49.00	0.00	RUN
llrf-kq14	0	129	0	0	0	150.00	0.00 ev	RUN
llrf-kq15	0	118	0	11	0	44.00	0.00	RUN
llrf-kqsp	0	80	0	0	0	452.00	0.00	RUN
pdu	2	24	0	0	0	8.00	0.00	RUN
procfs	0	294	0	0	0	43.00	0.00 ev	RUN
ps	1	328	0	0	0	12.00	0.00	RUN
radfet	30	539	0	0	0	133.00	0.00	RUN
stat	0	2	0	0	0	0.00	0.00	RUN
tertiary	0	282	0	0	0	93.00	0.00 ev	RUN
timing	1	146	0	2	0	395.00	28.00	RUN
vacuum-nip	0	8	0	0	0	0.00	0.00	RUN
vacuum-sip	9	391	0	0	0	95.00	4.00	RUN
vacuum-valves	0	1	0	0	0	0.00	0.00 ev	RUN
vacuum-vg	30	128	0	0	0	1.00	0.00	RUN

	Faulty	Started	Paused	Stop...	Pendi...	ev/1 ...	Fail./...	Context
E.S. Manager	90	5818	0	13	0	3075.0(	110.00	Not initial

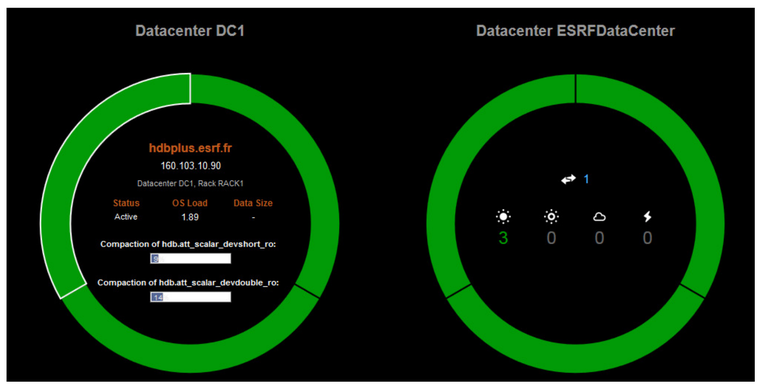
HDB++ at ESRF

– EventSubscriber instances	36
– ConfigurationManager instances	1
– Nr. of attributes archived	~8200
– Archive disk size	~850GB

History trend: number of archived attributes and number of events/second.

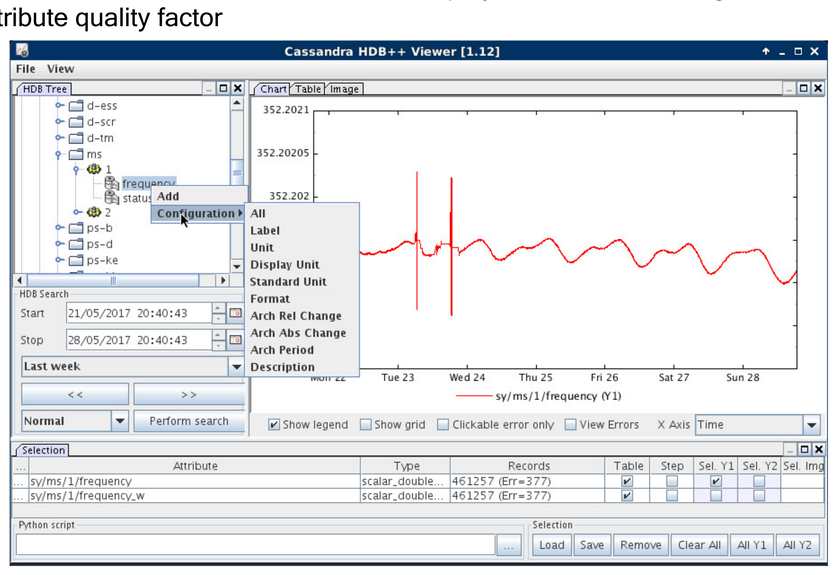


Cassandra at ESRF: 3 nodes Production Datacenter, 3 nodes Analytics Datacenter

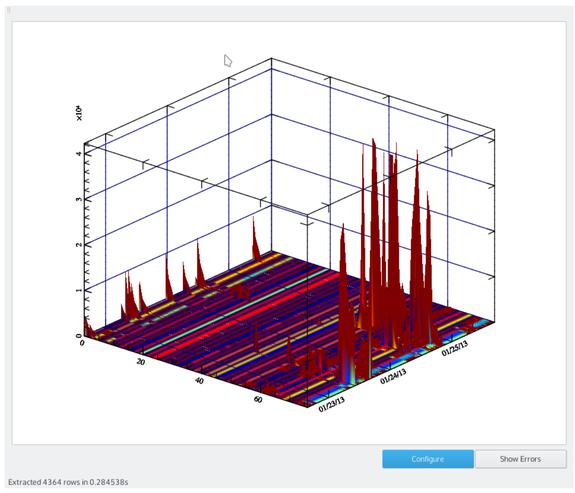


Graphical user interfaces

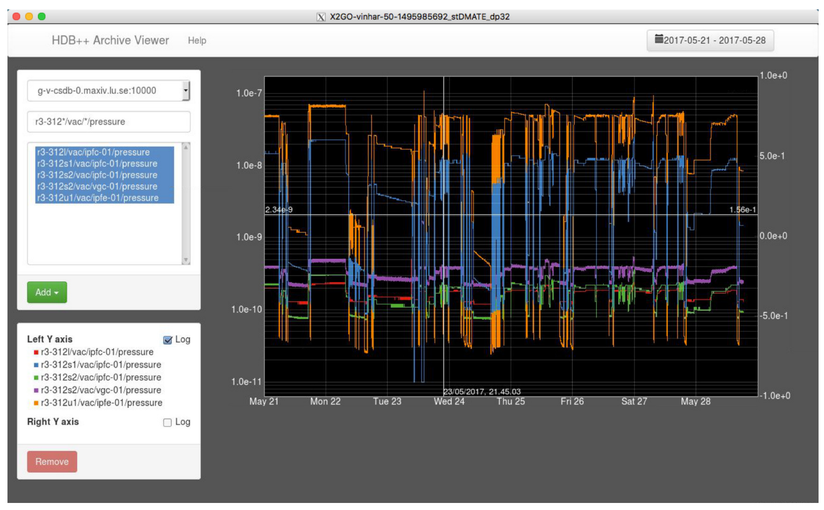
The Configurator GUI supports specifying per-attribute archiving strategies and time-to-live. The HDB++ Viewer can now display the attribute configuration history and attribute quality factor



Surface plot made with HdbExtractor++ viewer (Qt + MathGL framework)



Web-based viewer for HDB++ under development at MaxIV, based on node.js, Babel, React and Redux and managed by Webpack. The back-end uses aiohttp and Boker/datashader.



https://github.com/tango-controls
https://github.com/tango-controls-hdbpp
http://tango-controls.readthedocs.io/en/latest/tools-and-extensions/archiving/HDB++.html