TANGO

HDB++

L. Pivetta
on behalf of the HDB++ collaboration

Reynald Bourtembourg
Johan Forsberg
Thomas Juerges
Damien Lacoste
Sergi Rubio-Manrique
Jan David Mol
Graziano Scalamera

http://www.tango.controls.org

Requirements (2012)

- **Written in C++**
- **Event-driven**: exploit the TANGO publish/subscribe mechanism
- Architecture based on:
  - One or more archivers (EventSubscriber TANGO ds)
  - Configuration management (ConfigurationManager TANGO ds)
  - Libraries for data insertion and extraction (C++ and Java)
  - Data extraction: TANGO ds / clients
- **Fast**
  - One database for slow and fast archiving (up to 1000 samples/s, possibly more)
- **Flexible**
  - Easy to manage and maintain even without GUI front-ends
- **Self contained**
  - Single source for all configuration parameters (TANGO database)
- **Modular**
  - Abstraction+implementation libraries to support different database engines and schema
    - Support for existing HDB schema on MySQL
    - Support for **hdb++ new schema** with improved features (µs timestamp)
    - Support for **noSQL** back-end (Apache Cassandra)
    - Easily extensible to additional database/schema
- **Scalable**: same as TANGO, deploy as many DS as needed
- **GUI**: for HDB++ configuration and data extraction as well

Original requirements
Updates in the following slides

http://www.tango.controls.org

Archive event

- TANGO provides specific event for archiving purposes

- The **archive** event can be sent:
  - on value change → specify absolute or relative threshold
  - periodically → specify period

- Choosing the right thresholds is mandatory:
  - if the threshold is too large no events are sent → no archiving
  - if the threshold is too small too many events are sent → "noisy" archiving

- The right threshold is **strictly related to the variable/signal** to be archived (type, bandwidth, sampling rate...)

http://www.tango.controls.org

EventSubscriber

The EventSubscriber TANGO device is the core of the HDB++ archiving system

- Event based; TANGO provides **archive events** on change and periodic basis
- Configuration stored in the TANGO database (device)
- One thread in charge of event(s) subscription and callback execution: fills a FIFO acting as producer
- One thread in charge of pushing data into the database; reads the FIFO as consumer

- Device methods allow to perform the following per-instance operations:
    - **add/remove** an Attribute to/from archiving
    - **start/stop** the archiving for all Attributes
    - start/stop the archiving for one Attribute
    - read the status of an Attribute
    - read the number/list of Attributes currently archived (started)
    - read the number/list of Attributes currently not archived (stopped)
    - read the number/list of Attributes in charge
    - read the configuration parameters of each Attribute
    - read the number/list of working Attributes
    - read the number/list of faulty Attributes with diagnostics
    - read the number/list of Attributes pending in the FIFO

    + Manage context
    + Manage time-to-live

- The EventSubscriber exposes some **additional figures**:
    - **for each instance**, total number of records per time
    - for each instance, total number of failures per time
    - **for each attribute**, number of records per time
    - for each attribute, number of failures per time
    - for each attribute, time stamp of last record
    - for each attribute, min and max processing and storing times

EventSubscriber device configuration

07-10-2023

http://www.tango

EventSubscriber Class configuration

http://www.tango.controls.org

Archiving strategies

Support several operating modes, or strategies, in the archiving engine on a per-attribute basis.

→ Modify AttributeList device property to support a name=value approach for the strategy

*tango://srv-tango-srf.fcs.elettra.trieste.it:20000/eos/climate/18b20_eos.01/state; *
strategy=RUN|SHUTDOWN
*tango://srv-tango-srf.fcs.elettra.trieste.it:20000/eos/climate/18b20_eos.01/temperature; *
strategy=RUN|SHUTDOWN

Labels for the strategy defined in a free property, and/or in the class property and/or in the device property, with increasing priority (ContextList)
The defaults values, as well as the DefaultStrategy, are pre-defined but can be modified by the user. The default values for ContextList are:

ALWAYS	0
RUN	1
SHUTDOWN	2
SERVICE	3

Get|SetAttributeStrategy methods to modify per-attribute strategy.

"context" r/w memorized attribute, set by higher layer logic, tells the archiver the required Context transition

http://www.tango.controls.org

To support temporary storage of historical data one new column has been added to the att_conf table:

```
CREATE TABLE IF NOT EXISTS att_conf (
        att_conf_id INT UNSIGNED NOT NULL AUTO_INCREMENT PRIMARY KEY,
        att_name VARCHAR(255) UNIQUE NOT NULL,
        att_conf_data_type_id INT UNSIGNED NOT NULL,
        ttl INT UNSIGNED NULL DEFAULT 0,
        facility VARCHAR(255) NOT NULL DEFAULT '',
        domain VARCHAR(255) NOT NULL DEFAULT '',
        family VARCHAR(255) NOT NULL DEFAULT '',
        member VARCHAR(255) NOT NULL DEFAULT '',
        name VARCHAR(255) NOT NULL DEFAULT '',
        INDEX(att_conf_data_type_id)
) ENGINE=MyISAM COMMENT='Attribute Configuration Table';
```

The ttl column defines the time-to-live in hours on a per-attribute basis.
Configuring the ttl can be done from the Configuration Manager with the appropriate
Method, and modified with Get|SetAttributeTTL methods; ttl is stored in the AttributeList
configuration as name=value:

*tango://srv-tango-srf.fcs.elettra.trieste.it:20000/eos/climate/18b20_eos.01/state; \
strategy=RUN|SHUTDOWN; ttl=8760*
*tango://srv-tango-srf.fcs.elettra.trieste.it:20000/eos/climate/18b20_eos.01/temperature; \
strategy=RUN|SHUTDOWN; ttl=8760*

Deleting expired data delegated to the SQL backend.

http://www.tango.controls.org

ConfigurationManager

The ConfigurationManager TANGO device simplifies HDB++ archiving system management:

moved to EventSubscriber

- **handle** the request of archiving a new Attribute
  - ~~create an entry in the database if not already present~~
  - setup the Attribute archive event configuration
  - assign the Attribute to one of the archivers
- **move** an Attribute from one archiver to another
- keep trace of which Attribute is assigned to which archiver
- **start/stop** the archiving
- **remove** an Attribute from archiving
- + manage EventSubscriber context
- + manage Attribute time-to-live

The Configuration manager exposes some **global statistics**:
- total number of Archivers
- total number of working/faulty attributes
- total number of events per second
- overall minimum and maximum processing and storing time

http://www.tango.controls.org

ConfigurationManager
Class configuration

Database interface

A C++ API decouples the archiving engine (EventSubscriber) from the database back-end

Deprecated

- **libhdbpp** : database abstraction layer
- **libhdbpp-mysql** : implementation, HDB++ schema support, MySQL back-end
- ~~**libhdbpp-cassandra** : implementation, HDB++ schema support, Cassandra back-end~~
- ~~**libhdb-mysql-legacy** : implementation, legacy HDB schema support, MySQL back-end~~
- **libhdbpp-postgresql** : implementation, HDB++ schema support, PostgreSQL back-end
- ~~**libhdbpp-elkl** : implementation, HDB++ schema support, ELK back-end~~
- **libhdbpp-timescale** : implementation, HDB++ schema support, TimescaleDB back-end
- **libhdbpp-sqlite** : implementation, HDB++ schema support, SQLite back-end

The libraries allow reusing the EventSubscriber, the ConfigurationManager and the GUIs without any change

HDB++ is easily extendable to support additional back-ends(*) just writing the specific implementation library

(*) not limited to database engines... HDF5 format on file?

http://www.tango.controls.org

HDB++ in one picture

Hdb++ overview:
- Events based
- Scalable through the use of multiple EventSubscribers
- Support several databases, and the use of multiple clusters to separate concerns
- Get your data through various viewers or via libraries

Figure 2: HDB++ overview

http://www.tango.controls.org

Historical data extraction
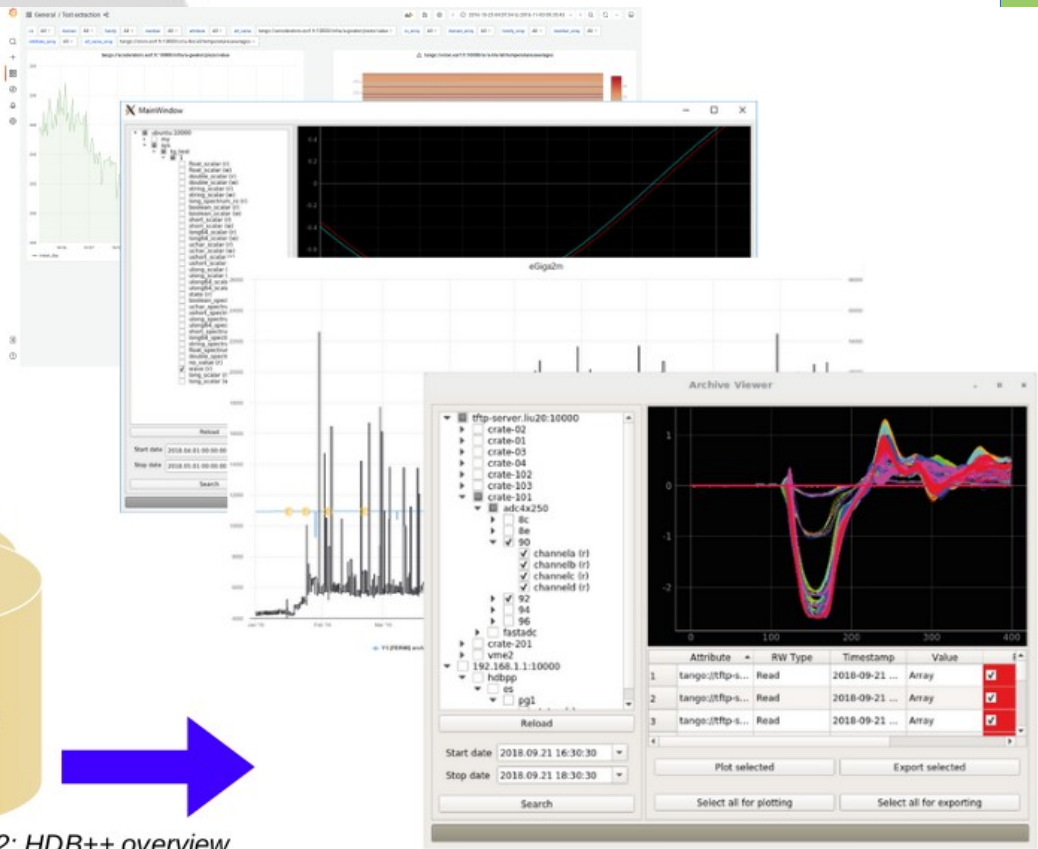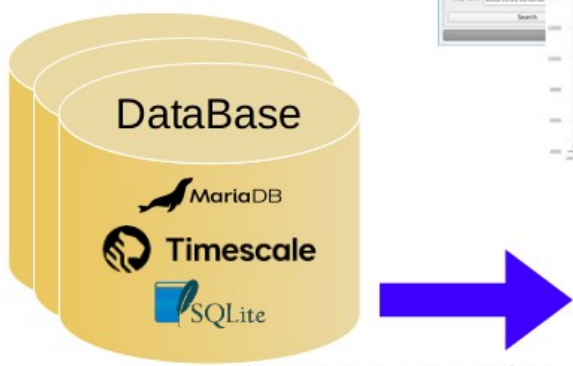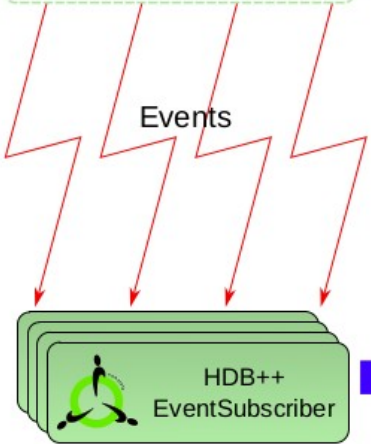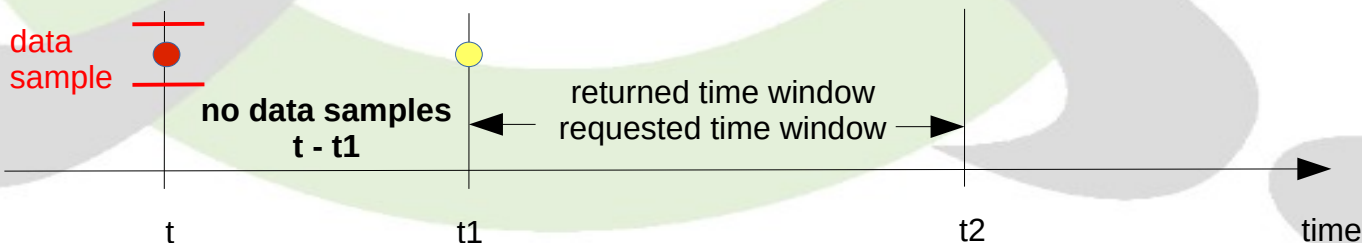
C++ and Java native extraction libraries

The data extraction library shall be able to **deal with event based archiving, i.e. data value change with respect to specified thresholds;** the possible lack of data in the requested time window shall be properly managed:

- returning some no-data-available error: in this case the reply contains no data

- enlarging the time window to include some archived data; no fake samples introduced
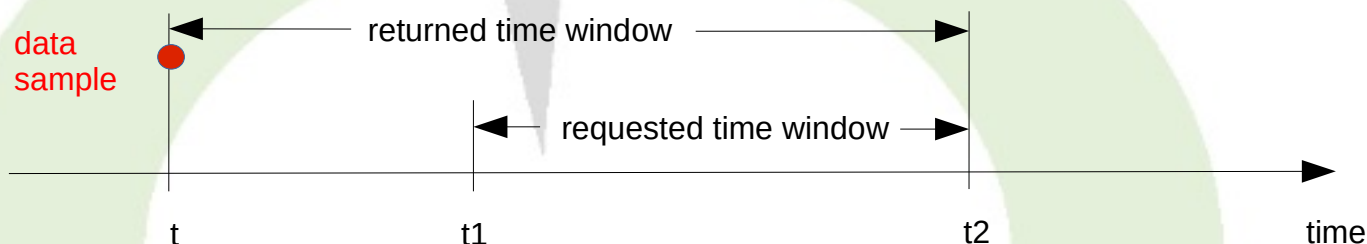
- returning the value of the last archived data anyhow; the requested time interval is kept and the last available data sample returned; the data value is guaranteed when **archiving on change,** care must be taken in case of **periodic archiving**

http://www.tango.controls.org

HDBConfigurator GUI

HdbConfigurator: a graphical user interface for the ConfigurationManager device

- Jive-like device tree
- Selected Attribute archive event parameters bottom left

- Started, stopped, paused attribute lists
- Pop-up to select archiver and parameters

07-10-2023

HdbExtractor++ GUI

- Qt based GUI using the MathGL framework for plotting
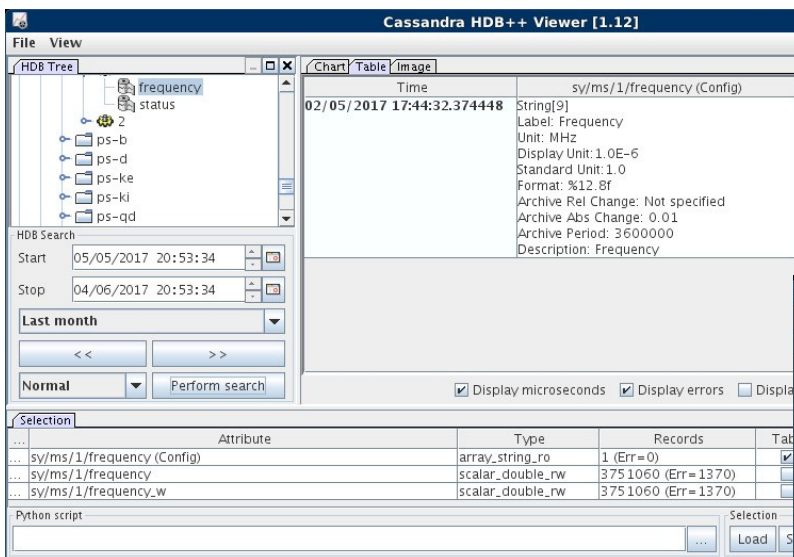- Exploits the C++ extraction library
- Supports multiline and surface plots

bc01/power_supply/psch_bc01.01/Current
bc01/power_supply/psch_bc01.02/Current
bc01/power_supply/psch_bc01.03/Current
bc01/power_supply/psch_bc01.04/Current

Extracted 4364 rows in 0.284538s
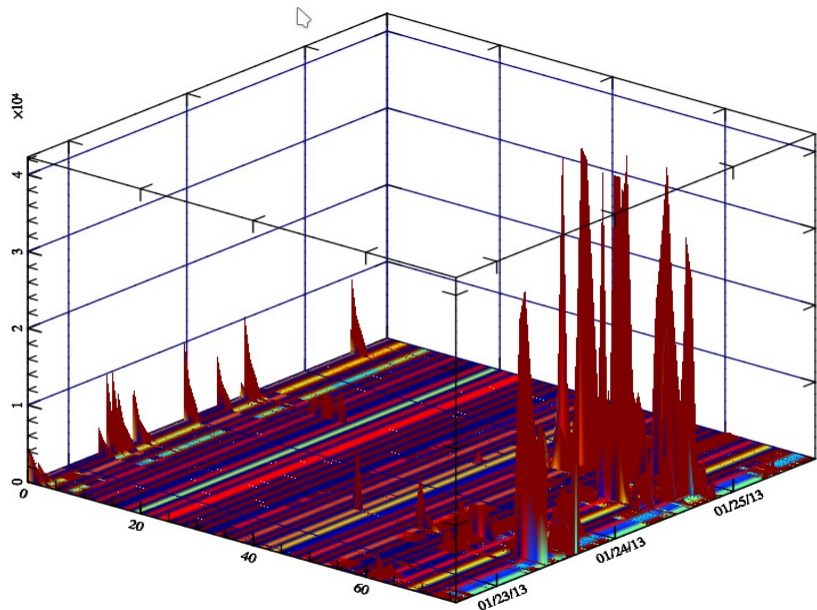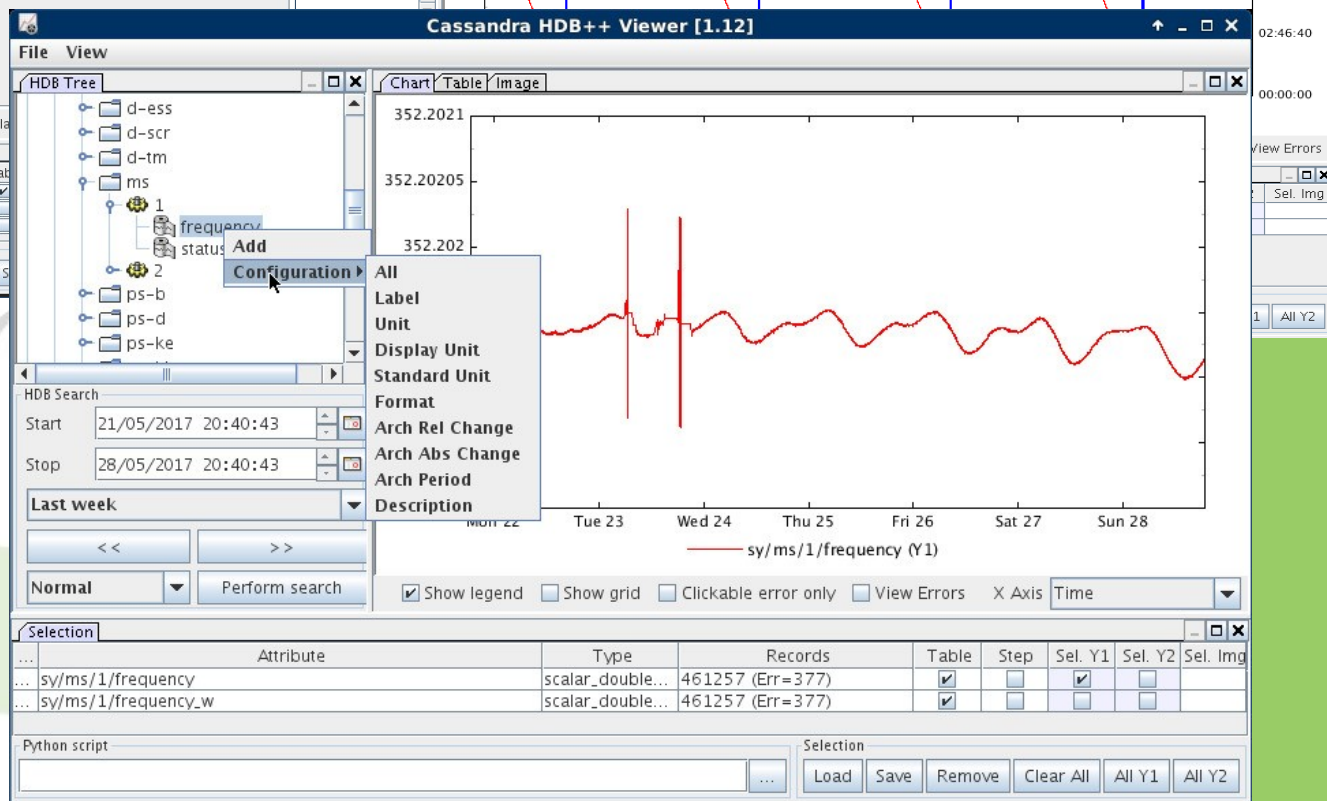
HDBViewer GUI

- Java based GUI for plotting
- Exploits the Java extraction library
- Table and multiline plots

Archviewer

- Webapp for plotting HDB++ from TimescaleDB
- back-end in Python >= 3.8
- front-end javascript+preact+d3

http://www.tango.controls.org

eGiga2m
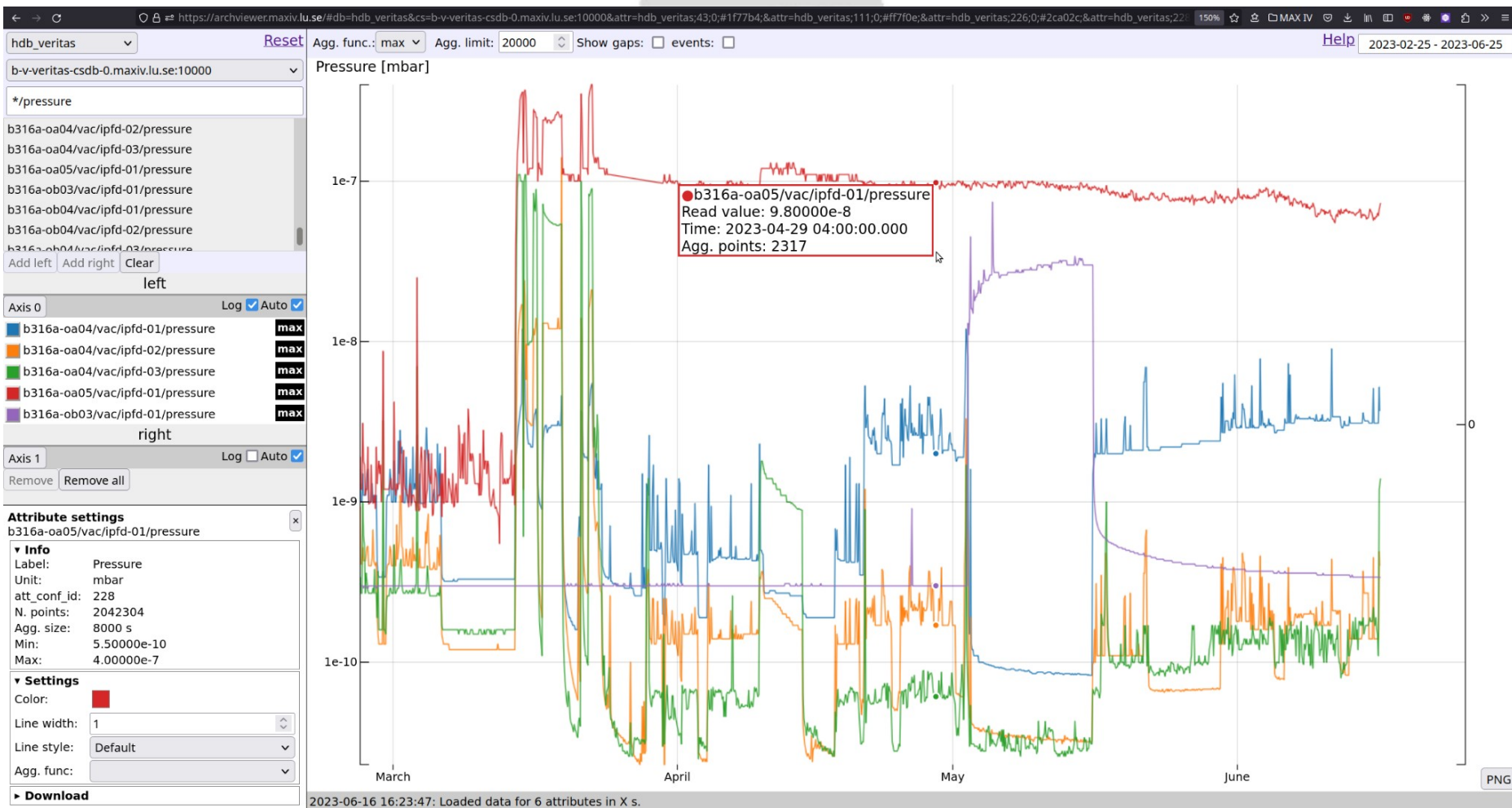
- Web interface
- MySQL and TimescaleDB backend

1) Time period selection
2) Time series selection. Strikethrough means time series are not current. A tooltip shows the last date the archiviation was active, if available.
3) Chart tooltip shows time series numerical value or error explanation
4) Legend. Click to hide/show the corresponding curve. A tooltip shows the number of samples and extraction time
5) Adjustable separation bar

http://www.tango.controls.org

HDB Diagnostics GUI

HdbDiagnostics - 2.2 - 23-05-2017 13:18:05

File View help

HDB++ Diagnostics (Cassandra)

	Faulty	Started	Paused	Stopped	Pending	ev/sec	Fail./sec	Context
"HDB++" statistics	0	78	0	0	0	12.00	0.00	Run
Beam Position Monitors	0	293	0	1	0	0.00	0.00	Run
Booster 1	0	234	0	0	0	2.00	0.00	Run
D-RGA	0	62	0	0	0	8.00	0.00	Run
Experiments Gamma Monitors	0	43	0	0	0	6.00	0.00	Run
FE Vacuum Pressures	0	375	0	0	0	0.00	0.00	Run
Fast BPM	0	4	0	0	0	0.00	0.00	Run
Front Ends	0	175	0	2	0	0.00	0.00	Run
ID22	0	7	0	0	0	0.00	0.00	Run
Injext	0	38	0	0	0	1.00	0.00	Run
Insertion Devices 1	0	94	0	0	0	0.00	0.00	Run
Insertion Devices 2	0	115	0	0	0	0.00	0.00	Run
Insertion Devices Corrections	0	32	0	0	0	0.00	0.00	Run
LINAC	0	121	0	0	0	0.00	0.00	Run
Power Supplies	0	188	0	1	0	2.00	0.00	Run
RGA 1	0	368	0	0	0	2.00	0.00	Run
RGA 2	0	305	0	0	0	0.00	0.00	Run
RGA 3	0	364	0	0	0	2.00	0.00	Run
RGA 4	0	497	0	0	0	0.00	0.00	Run
RGA 5	0	303	0	0	0	3.00	0.00	Run
RGA 6	0	427	0	61	0	0.00	0.00	Run
RGA 7	0	244	0	0	0	1.00	0.00	Run
RGA 8	0	363	0	61	0	0.00	0.00	Run
Radiations	0	199	0	0	0	0.00	0.00	Run
SR 1	3	65	0	17	0	6.00	0.00	Run
SR 2	0	200	0	0	0	1.00	0.00	Run
SR Correctors	0	94	0	0	0	0.00	0.00	Run
SR RF	7	443	0	0	0	16.00	0.00	Run
SR Steerers	0	201	0	0	0	0.00	0.00	Run
SR Vacuum 1	0	61	0	0	0	0.00	0.00	Run
SR Vacuum Pressures	0	741	0	0	0	0.00	0.00 ev/	Run
SR Vacuum Temperatures	0	934	0	1	0	1.00	0.00 ev/	Run
SY RF	1	107	0	0	0	2.00	0.00	Run
SYSTEM 1	1	215	0	0	0	0.00	0.00	Run
TL1/TL2	0	43	0	0	0	2.00	0.00 ev/	Run
test labs	0	12	0	58	0	0.00	0.00 ev/	Run

	Faulty	Started	Paused	Stopped	Pending	ev/sec	Fail./sec	Context
E.S. Manager	12	8045	0	202	0	85	0	run

http://www.tango.controls.org

libhdbpp-python

Python3 package for data extraction

> **pip3 install pyhdbpp**

- Common API for MariaDB, MySQL and TimeScaleDB
- AbstractReader object provides generic extraction interface
- Dedicated DB back-end implementation is loaded at runtime
- Connection setup is stored in .yaml or Tango properties
- Taurus Widget available! (pyqtgraph)

```
import pyhdbpp

rd = pyhdbpp.get_default_reader()

rd.get_attribute_values('sr/vc/01/pressure',

                        '2023-06-03','2023-06-04')


Out: ((1685782812.169, 1.4e-08, 0),

      (1685782842.011, 1.5e-08, 0), …
```
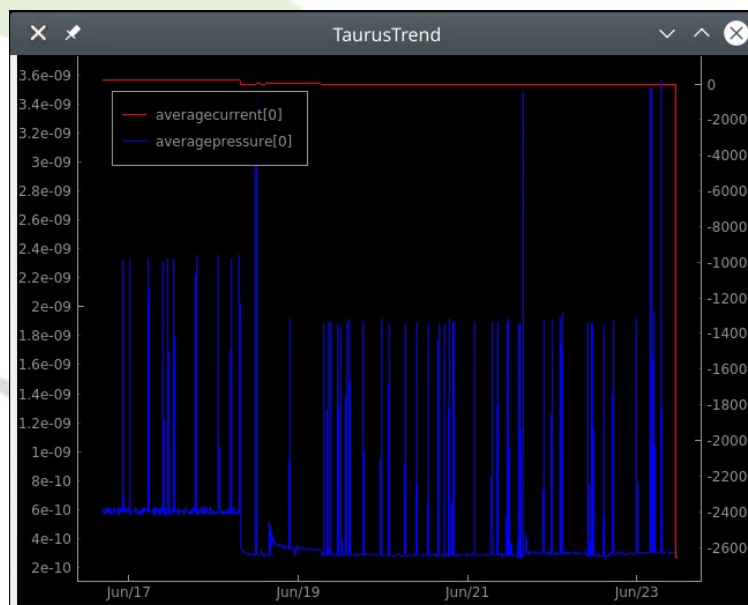

- Elastic? SQLite? … just inherit AbstractReader and contribute with your own implementation!
- ArchivingBrowser and SnapGUI also being migrated to python3 (ongoing)
- Configuration API still pending (PyTangoArchiving)

http://www.tango.controls.org

Gitlab

https://gitlab.com/tango-controls/hdbpp

Thank-you

...and now…

TANGO Workshop - ICALEPCS 2023

http://www.tango.controls.org