HDB++

L. Pivetta
R. Bourtembourg
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
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 server 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
EventSubscriber Class configuration

http://www.tango-controls.org
The ConfigurationManager TANGO device server simplifies HDB++ archiving system management:

- **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
ConfigurationManager
Class configuration

Class properties [HdbConfigurationManager]

<table>
<thead>
<tr>
<th>Property name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>cvs_location</td>
<td>/home/cvsadm/cvsroot/termi/servers/hdb++/hdb++cm/src/</td>
</tr>
<tr>
<td>DbHost</td>
<td>cassandra2</td>
</tr>
<tr>
<td>DbName</td>
<td>hdbtest</td>
</tr>
<tr>
<td>InheritedFrom</td>
<td>TANGO_BASE_CLASS</td>
</tr>
<tr>
<td>LibConfiguration</td>
<td>keyspace=hdbtest, contact_points=cassandra2, user=hdbwriter, password=MyPass, libname=libhdb++cassandra.so</td>
</tr>
<tr>
<td>ProjectTitle</td>
<td>Hdb++ configuration manager</td>
</tr>
</tbody>
</table>

Environment:
- Jive 7.17 [acudebian7:10000]
Database interface

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

- **libhdb++**: database abstraction layer
- **libhdb++mysql**: implementation, HDB++ schema support, MySQL back-end
- **libhdb++cassandra**: implementation, HDB++ schema support, Cassandra back-end
- **libhdbmysql**: implementation, legacy HDB schema support, MySQL 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?
In one picture

HDB++Extractor GUI (Qt, C++)
C++ extraction library

HDBViewer GUI (Java)
Java extraction library

MySQL replica or Cassandra analytics

MySQL or Cassandra Production

HDBConfigurer GUI

EventSubscriber TANGO DS

OMQ TANGO archive events

TANGO DS

TANGO DS

TANGO DS

Key
- TANGO API
- Archive events
- Archive queries
- Application API
- DS
- TANGO device
- GUI
- GUI application

http://www.tango.controls.org
Historical data extraction

C++ and Java native extraction libraries

The data extraction library shall be able to \textbf{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 \textbf{archiving on change}, care must be taken in case of \textbf{periodic archiving}
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
HDBConfigurator GUI

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

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

<table>
<thead>
<tr>
<th>System</th>
<th>Faulty</th>
<th>Started</th>
<th>Paused</th>
<th>Stopped</th>
<th>Pending</th>
<th>ev/sec</th>
<th>Fail/sec</th>
<th>Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;HDB++&quot; statistics</td>
<td>0</td>
<td>78</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>12.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>Beam Position Monitors</td>
<td>0</td>
<td>293</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>Booster 1</td>
<td>0</td>
<td>234</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>D-RGA</td>
<td>0</td>
<td>62</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>8.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>Experiments Gamma Monitors</td>
<td>0</td>
<td>43</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>6.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>FE Vacuum Pressures</td>
<td>0</td>
<td>375</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>Fast BPM</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>Front Ends</td>
<td>0</td>
<td>175</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>ID22</td>
<td>0</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>Injext</td>
<td>0</td>
<td>38</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>Insertion Devices 1</td>
<td>0</td>
<td>94</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>Insertion Devices 2</td>
<td>0</td>
<td>115</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>Insertion Devices Corrections</td>
<td>0</td>
<td>32</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>LINAC</td>
<td>0</td>
<td>171</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>Power Supplies</td>
<td>0</td>
<td>188</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>RGA 1</td>
<td>0</td>
<td>368</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>RGA 2</td>
<td>0</td>
<td>305</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>RGA 3</td>
<td>0</td>
<td>364</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>RGA 4</td>
<td>0</td>
<td>497</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>RGA 5</td>
<td>0</td>
<td>303</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>RGA 6</td>
<td>0</td>
<td>427</td>
<td>0</td>
<td>61</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>RGA 7</td>
<td>0</td>
<td>244</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>RGA 8</td>
<td>0</td>
<td>363</td>
<td>0</td>
<td>61</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>Radiations</td>
<td>0</td>
<td>199</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>SR 1</td>
<td>3</td>
<td>65</td>
<td>0</td>
<td>17</td>
<td>0</td>
<td>6.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>SR 2</td>
<td>0</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>SR Correctors</td>
<td>0</td>
<td>94</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>SR RF</td>
<td>7</td>
<td>443</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>16.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>SR Steerers</td>
<td>0</td>
<td>201</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>SR Vacuum 1</td>
<td>0</td>
<td>61</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>SR Vacuum Pressures</td>
<td>0</td>
<td>741</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>SR Vacuum Temperatures</td>
<td>0</td>
<td>934</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>SY RF</td>
<td>1</td>
<td>107</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>SYSTEM 1</td>
<td>1</td>
<td>215</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>Run</td>
</tr>
<tr>
<td>TLL/TL2</td>
<td>0</td>
<td>43</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00 ev/Run</td>
</tr>
<tr>
<td>test labs</td>
<td>0</td>
<td>12</td>
<td>0</td>
<td>58</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00 ev/Run</td>
</tr>
<tr>
<td>E.S. Manager</td>
<td>12</td>
<td>8045</td>
<td>0</td>
<td>202</td>
<td>0</td>
<td>85</td>
<td>0.00</td>
<td>Run</td>
</tr>
</tbody>
</table>
Cassandra

- Accelerator Control System
- Beaml ine Control System
- Tango Device Server
- OMQ
- HDB++
- Event Subscriber
- HDB++ Configuration Manager
- Production Datacentre
- Analytics Datacentre (statistics, decimation)
- Cassandra Node
- Cassandra/Spark Node
- Diagnostics GUI
- HDB++ Configurator GUI
- HDB++ Viewer GUI
Cassandra

Cassandra OpsCenter

CassandraMonitor TANGO device

The Cassandra Monitoring device server is a TANGO device server (see http://www.tango-controls.org), able to monitor what's happening on a given Apache Cassandra node.

Zeppelin

TANGO Workshop
Important: learn it before putting it in production:

https://academy.datastax.com/courses

http://cassandra.apache.org/doc/latest/

http://www.tango-controls.org/community/project-docs/hdbplusplus/hdbplusplus-doc/hdbplusplus-presentations/
https://github.com/search?q=org%3Atango-controls+hdbpp

Two branches

- “master” – legacy hdb++ stuff, Makefile based build
- “build-and-package” - cmake based build, WIP

- Move to https://github.com/tango-controls-hdbpp organization in progress
Compiling from “master” branch:

```bash
git clone --recursive http://github.com/tango-controls/libhdbpp-mysql.git
cd libhdbpp-mysql
export TANGO_DIR=/usr/local/tango-9.2.5a
export OMNIORB_DIR=/usr/local/omniorb-4.2.1
export ZMQ_DIR=/usr/local/zeromq-4.0.7
make
```

```bash
git clone --recursive http://github.com/tango-controls/hdbpp-es.git
cd libhdbpp-mysql
export TANGO_DIR=/usr/local/tango-9.2.5a
export OMNIORB_DIR=/usr/local/omniorb-4.2.1
export ZMQ_DIR=/usr/local/zeromq-4.0.7
make
```

```bash
git clone --recursive http://github.com/tango-controls/hdbpp-cm.git
cd libhdbpp-mysql
export TANGO_DIR=/usr/local/tango-9.2.5a
export OMNIORB_DIR=/usr/local/omniorb-4.2.1
export ZMQ_DIR=/usr/local/zeromq-4.0.7
make
```

Note: Makefile uses “mysql_config” script to retrieve MySQL installation paths. Adapt Makefile to use “mariadb_config” if you’re using MariaDB
Debian packages

Pre-release debian package available on bintray:
  • https://bintray.com/tango-controls/debian

```bash
> echo "deb https://dl.bintray.com/tango-controls/debian stretch main" |
> sudo tee -a /etc/apt/sources.list

> sudo apt-get update

> sudo apt-get install hdb++mysql

> sudo apt-get install cassandra-cpp-driver
> sudo apt-get install hdb++cassandra
```

hdb++xxx are meta-packages that install all the dependencies
To be included in the official debian repository