TANGO Controls

http://www.tango-controls.org/

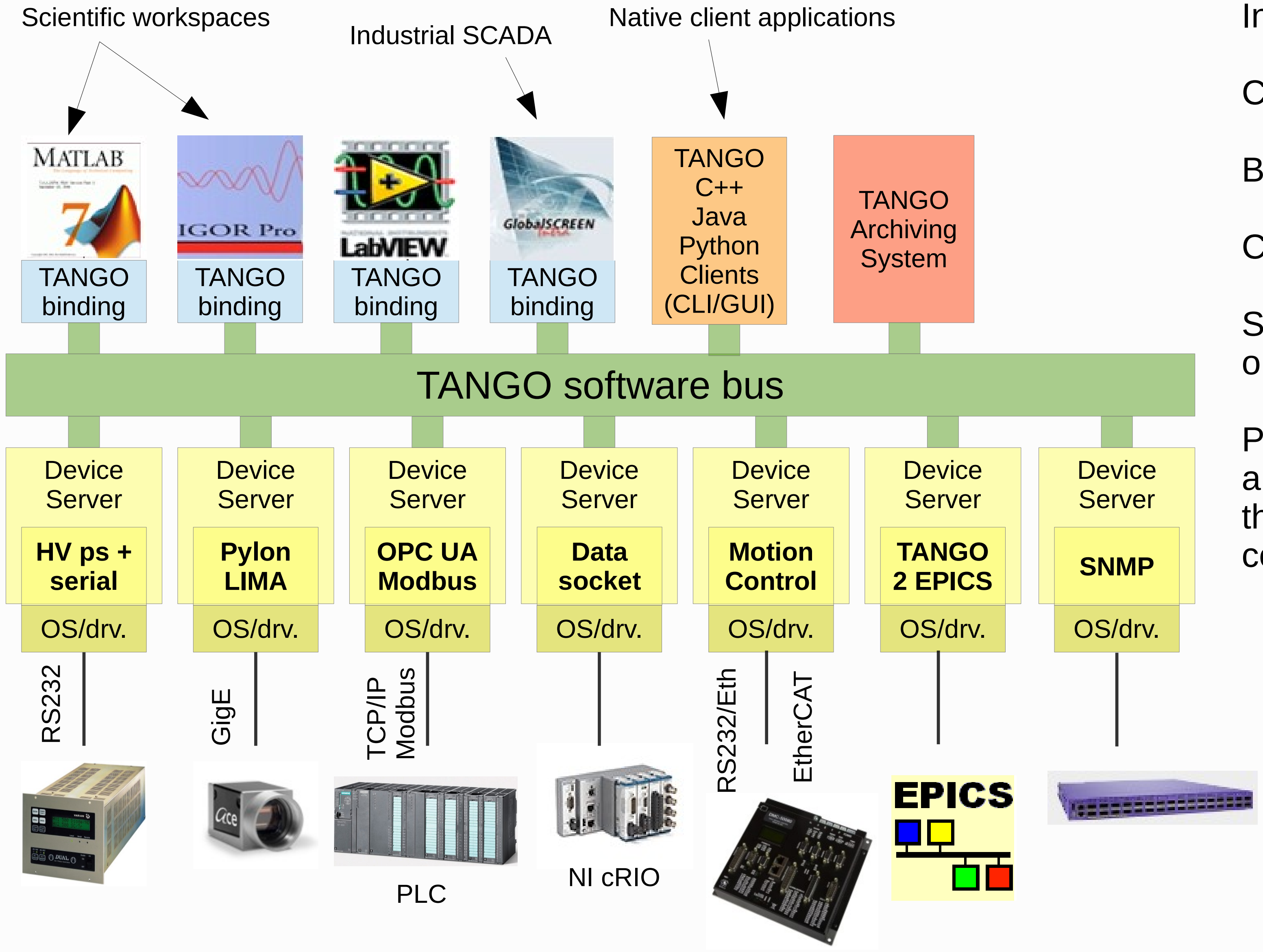
Outline

What is TANGO?
Language/OS/Compilers
CORBA and ZeroMQ
TANGO Database
TANGO device and device server
Communication models
Multicast
Polling
Events
Alarms
Logging system
Historical DataBase

Jive/Starter/Astor
ATK/Jdraw/Synoptic
Qtango/Mango
Taurus
E-giga/Canone
TANGO Bindings

Connecting things **together**

What is TANGO?



Scientific workspaces

Industrial SCADA

Native client applications

TANGO binding

TANGO binding

TANGO binding

TANGO binding

TANGO C++ Java Python Clients (CLI/GUI)

TANGO Archiving System

TANGO software bus

Device Server **HV ps + serial** OS/drv.

Device Server **Pylon LIMA** OS/drv.

Device Server **OPC UA Modbus** OS/drv.

Device Server **Data socket** OS/drv.

Device Server **Motion Control** OS/drv.

Device Server **TANGO 2 EPICS** OS/drv.

Device Server **SNMP** OS/drv.

RS232

GigE

TCP/IP Modbus

RS232/Eth

EtherCAT

PLC

NI cRIO

In short:

Control system framework

Based on CORBA and ZMQ

Centralized config. database

Software bus for distributed objects

Provides unified interface to all equipments hiding **how** they are connected/managed

Language/OS/Compilers

TANGO release 8.1.2 (C++98, C++11)

Pre-release TANGO 9

Languages
- Server side: C++, Java, Python
- Client side: C++, Java, Python, Matlab, LabView, IgorPro, Panorama

OS – Linux (PREEMPT_RT, Xenomai hard real-time)
- Architecture: x86, PPC, ARM
- Compiler: gcc 3.3 – gcc 4.8

OS – Windows XP/Vista/7
- Architecture: x86
- Compiler: VC9, VC10, VC11

OS – MacOSX
- Architecture: x86
- Compiler: gcc 4.6 – gcc 4.8

CORBA and ZeroMQ

CORBA – http://www.omg.org

- Common Object Request Broker Architecture specification
- Defines the ORB and the services available for all objects
- Uses an Interface Definition Language (IDL) and defines bindings between IDL and programming languages
- An Interoperable Object Reference (IOR) identifies each object
- TANGO adopts omniORB for C++ and JacORB for Java
  http://www.omniorb.sourceforge.net
  http://www.jacorg.org

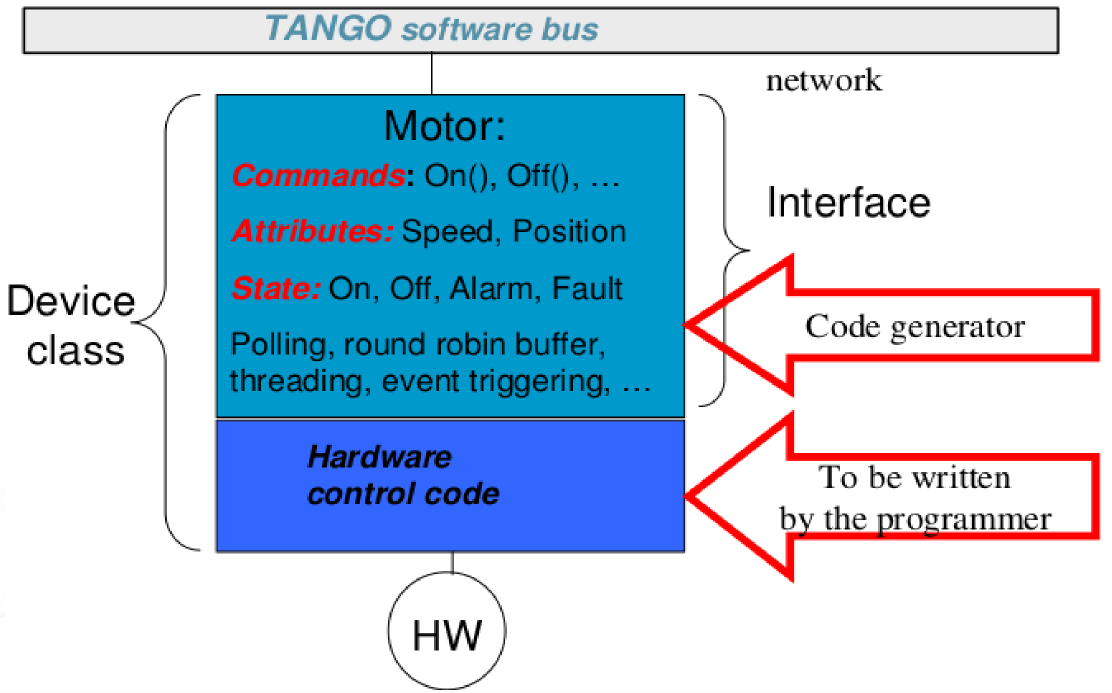
ZeroMQ, ZMQ, 0MQ – http://zeromq.org

- An embeddable networking library that acts like a concurrency framework
- Sockets that carry whole messages across various transports like in-process, inter-process, TCP and multicast
- Used for event-based communication in TANGO ≥ 8

TANGO Device

Everything which needs to be controlled is modeled as a Device
Each Device is identified by the Fully Qualified Domain Name (FQDN)
tango://host:port/domain/family/member
Each Device belongs to a TANGO class and exposes the **same interface**:
- **Command**(s): act on devices (e.g. power on)
- **Attribute**(s): set/get physical values (e.g. set/get motor position)
  - Attribute properties: per-attribute configuration parameters
  - State: TANGO Device finite state machine value
- **Properties:** configuration parameters

Device level
Class level
Free/Global


TANGO software bus
network
Motor:
Commands: On(), Off(), …
Attributes: Speed, Position
State: On, Off, Alarm, Fault
Polling, round robin buffer, threading, event triggering, …
Interface
Code generator
Device class
Hardware control code
To be written by the programmer
HW

Connecting things **together**

TANGO Device Server

The DS is the process where the TANGO class(es) run

Device number and names for a TANGO class are defined within the database, **not in the code**

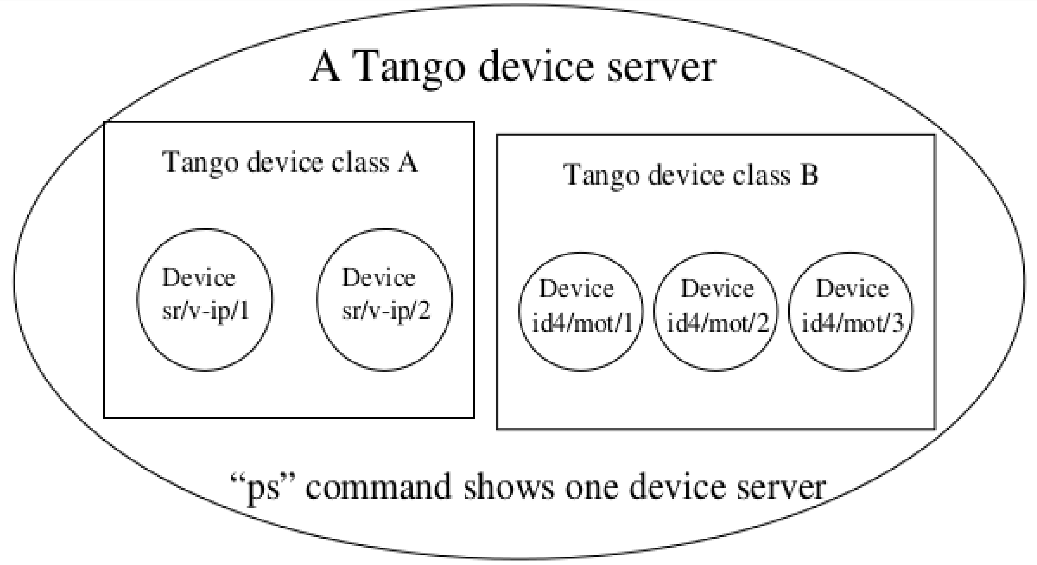
Which TANGO class(es) are part of a DS process is defined in the database **but also in the code**

The DS **can** host several TANGO classes, each class **can** be instantiated several times

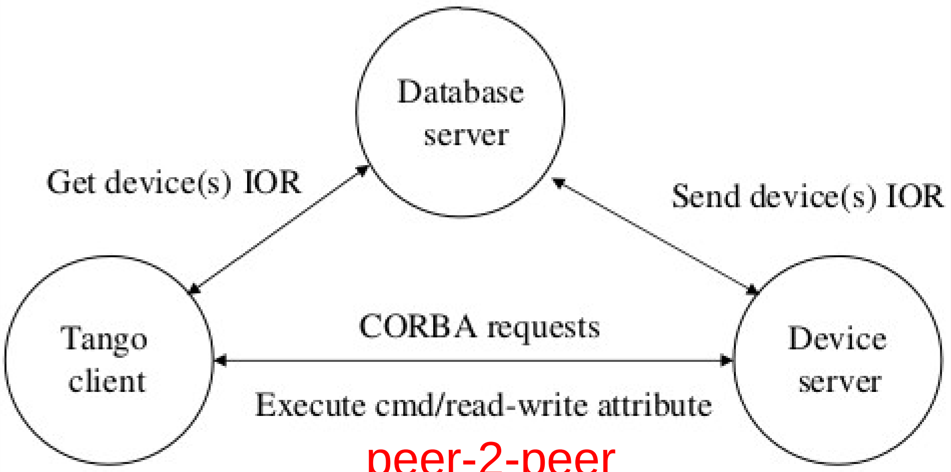
...but be careful with code or DLLs not thread safe

DS configuration is stored into the TANGO database (MySQL)

Advice: design for speed; never, ever do any assumption about the nature and the number of clients → always minimize response time



Startup sequence



peer-2-peer

Communication models

Two communication models available

Client/server: the communication between clients and servers can be synchronous and/or asynchronous

- The **client inquires** the server; the reply can be synchronous (blocking) or asynchronous (non blocking)

Publish/subscribe: the communication is event-driven

The device **server informs** the client that something has happened

Additionally, as a special case, **multicast** is also available through ZMQ, that uses the OpenPGM implementation of PGM protocol (RFC 3208 – reliable multicasting Protocol). Has to be configured, defining the global property CtrlSystem->MulticastEvent containing the following fields:

multicast address,	*226.20.21.22*
port number,	*2222*
[rate in Mbit/s]	*20*
[ivl in s]	*10*
event name	*device/with/multicast/state.change*

Polling

The Polling mechanism allows the Tango device to **decouple** the real device from the client(s) request(s)

Each Tango device server may have **one or more polling thread**(s) (tuning)

Polling allows to continuously monitor the “health” of the equipment

Attributes and/or Commands can be polled

The polling result is stored in a **buffer with configurable depth**, just limited by available memory

A client is able to read data from:

- The real device
- The last record in the polling buffer
- The polling buffer with fall-back to the real device

The complete buffer history is also available to the client → large buffers mean “automatic” shared memory mechanism available

Advice: the frequency of real hardware access has to be tuned on the equipment
(e.g. accessing that old reliable 9600 baud serial line...)

Events

Implement the publish/subscribe pattern; based on ZeroMQ since Tango 8
(no more notification service)

Available on **attributes**

The client registers her interest **once** in an event (value)

The server informs the client every time an event has occurred

Default based on device server polling: needs configuration but does not require changes in the device server code

Additionally the event generation can be managed by the developer: **events pushed by code**

Client callback executed when an event is received

Six types of events available:

- **Change**: absolute change, relative change
- **Periodic**: period
- **Archive**: absolute change, relative change, period
- **Attribute configuration**: no parameters
- **Data ready**: managed by the developer
- **User**: managed by the developer

Heartbeat to check that the device server is alive (10s)

Alarms

Device alarms

- Warning and alarm **thresholds available** as **per-attribute** configuration
- TANGO changes the State of the Device and the Quality factor of the attribute depending on attribute value and thresholds

TANGO alarms

Specialized TANGO device servers, useful to handle complex alarm rules based on multiple values/multiple logics

- C++ alarm device server: event based
- Python alarm device server: polling/event (with Taurus)

Parser for arbitrary alarm formula support

*kg01/mod/linkstabilizer_kg01.01/State == ON && kg01/mod/linkstabilizer_kg01.01/Drift1_Threshold && *
abs(kg01/mod/linkstabilizer_kg01.01/Drift1_rate) > kg01/mod/linkstabilizer_kg01.01/Drift1_Threshold

Support for alarm groups and alarm levels (LOG, WARNING, FAULT)
Support for external command execution (TANGO DS)

Scalability: any number of TANGO alarm servers can be deployed, based on requirements, architectural constraints, performance required...

TANGO groups

TANGO groups provide the user with a **single control point for a collection of devices**.
For instance, the TANGO Group API supplies a *command_inout()* method to execute the same command on all the elements of a group.
Tango Group is also a **hierarchical object**: in other words, it is possible to build a group of both groups and individual devices.
Simple and effective way to create logical views of the control system.

Example: Beam Loss Monitors

```
blm2-srv
|→ 01
|        |→bc01/radiation_protection/blm_bpm_bc01.05
|        |→bc01/radiation_protection/blm_b_bc01.01_l
|        |...
|→ 02
|        |→ bc02/radiation_protection/blm_b_bc02.01_l
|        |...
|...
```

193 total device number

```
blm = Group('radiation_protection')
blm.add('*/radiation_protection/*')
if blm->ping() == True:
        print “all devices alive”
else
        print “at least one device dead”
```

TANGO ACL

Two kind of users (identified by system login name):

- users defined in the ACL
- users not defined in the ACL → rights fall below “All users”

Two kind of rights, at host **and** device level:

- Read (+ optional **per-class** allowed commands)
- Write

taurel

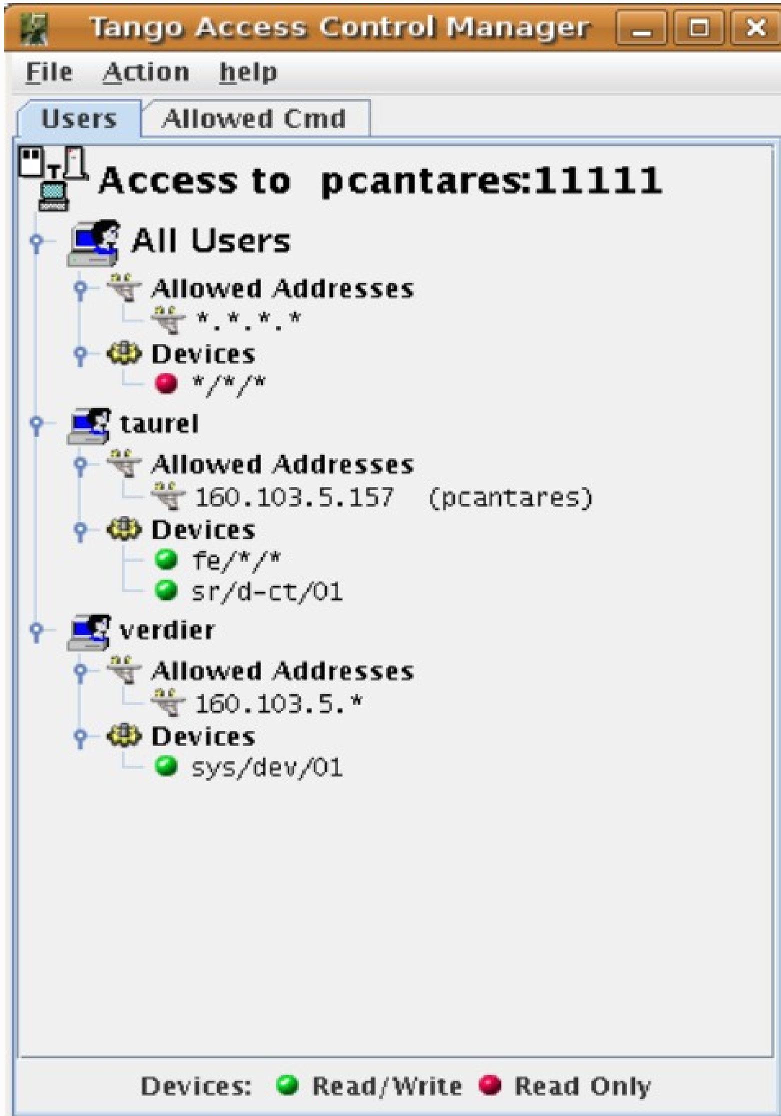
- write to sr/d-ct/01 and fe/*/* only from pcantares
- read all other devices only from pcantares

verdier

- write to sys/dev/01 from any host on 160.103.5.0/24 subnet
- read all other devices from the same subnet

all users

- read-only access from any host



```
Access to  pcantares:11111
  All Users
    Allowed Addresses
      *.*.*.*
    Devices
      🔴 */*/*
  taurel
    Allowed Addresses
      160.103.5.157   (pcantares)
    Devices
      🟢 fe/*/*
      🟢 sr/d-ct/01
  verdier
    Allowed Addresses
      160.103.5.*
    Devices
      🟢 sys/dev/01

Devices:  🟢 Read/Write  🔴 Read Only
```

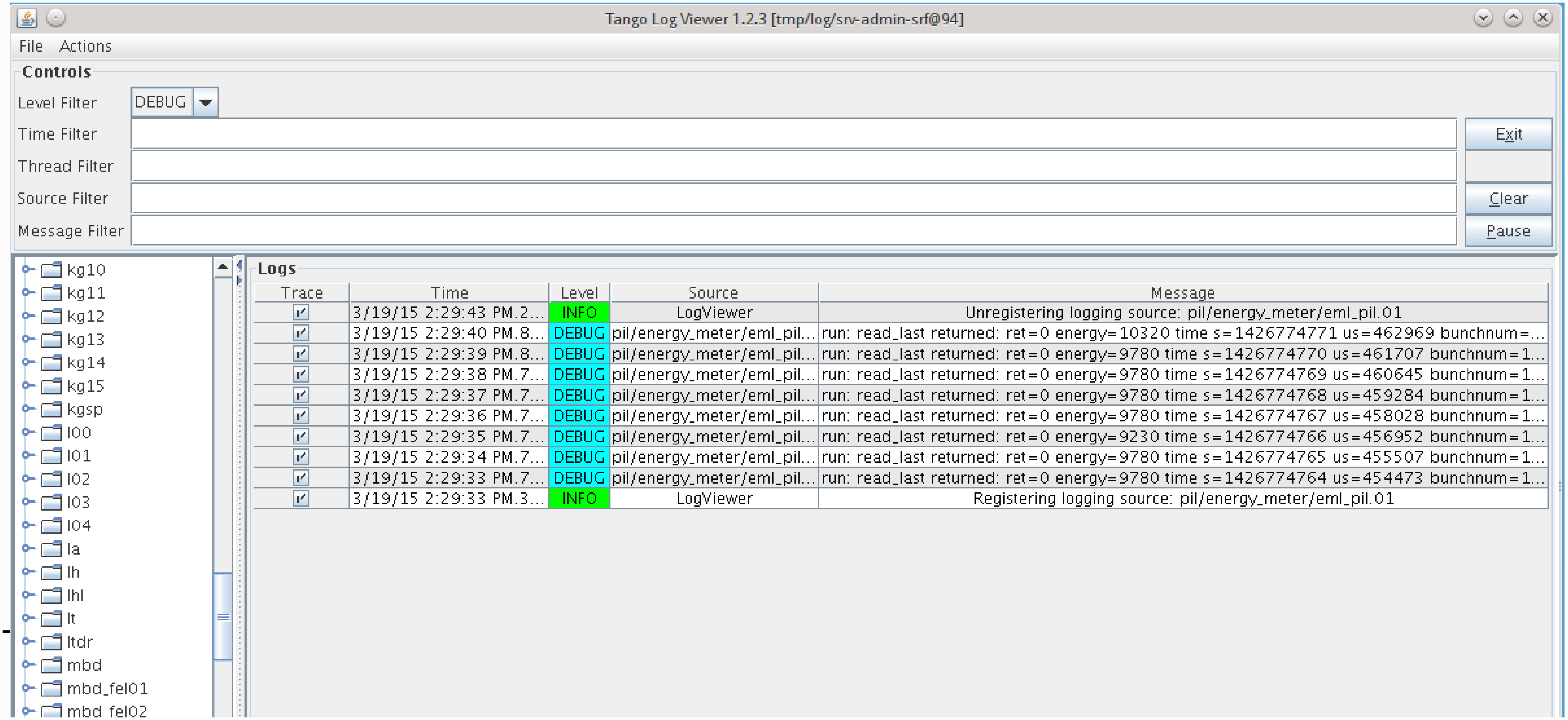
Logging system

The TANGO logging system allows a device server to send messages to:

- The console
- A file
- An application called LogViewer (GUI)
- A file on a remote host via specialized TANGO device server exposing the appropriate API

Six logging levels: DEBUG<INFO<WARN<ERROR<FATAL<OFF

LogViewer: Java graphical application to display, filter and sort logging messages



Level Filter: DEBUG

Time Filter:

Thread Filter:

Source Filter:

Message Filter:

Trace	Time	Level	Source	Message
☑	3/19/15 2:29:43 PM.2...	INFO	LogViewer	Unregistering logging source: pil/energy_meter/eml_pil.01
☑	3/19/15 2:29:40 PM.8...	DEBUG	pil/energy_meter/eml_pil...	run: read_last returned: ret=0 energy=10320 time s=1426774771 us=462969 bunchnum=...
☑	3/19/15 2:29:39 PM.8...	DEBUG	pil/energy_meter/eml_pil...	run: read_last returned: ret=0 energy=9780 time s=1426774770 us=461707 bunchnum=1...
☑	3/19/15 2:29:38 PM.7...	DEBUG	pil/energy_meter/eml_pil...	run: read_last returned: ret=0 energy=9780 time s=1426774769 us=460645 bunchnum=1...
☑	3/19/15 2:29:37 PM.7...	DEBUG	pil/energy_meter/eml_pil...	run: read_last returned: ret=0 energy=9780 time s=1426774768 us=459284 bunchnum=1...
☑	3/19/15 2:29:36 PM.7...	DEBUG	pil/energy_meter/eml_pil...	run: read_last returned: ret=0 energy=9780 time s=1426774767 us=458028 bunchnum=1...
☑	3/19/15 2:29:35 PM.7...	DEBUG	pil/energy_meter/eml_pil...	run: read_last returned: ret=0 energy=9230 time s=1426774766 us=456952 bunchnum=1...
☑	3/19/15 2:29:34 PM.7...	DEBUG	pil/energy_meter/eml_pil...	run: read_last returned: ret=0 energy=9780 time s=1426774765 us=455507 bunchnum=1...
☑	3/19/15 2:29:33 PM.7...	DEBUG	pil/energy_meter/eml_pil...	run: read_last returned: ret=0 energy=9780 time s=1426774764 us=454473 bunchnum=1...
☑	3/19/15 2:29:33 PM.3...	INFO	LogViewer	Registering logging source: pil/energy_meter/eml_pil.01

Historical Database

HDB (Java) - Set of three databases

- HDB: permanent, up to 0.1 Hz archiving rate
- TDB: temporary, up to 1 Hz archiving rate
- Snap: context save/restore
- Support for Oracle and MySQL RDBMS
- 4(+3)+3 Device servers
- **Polling** based
- GUI: Mambo, Bensikin

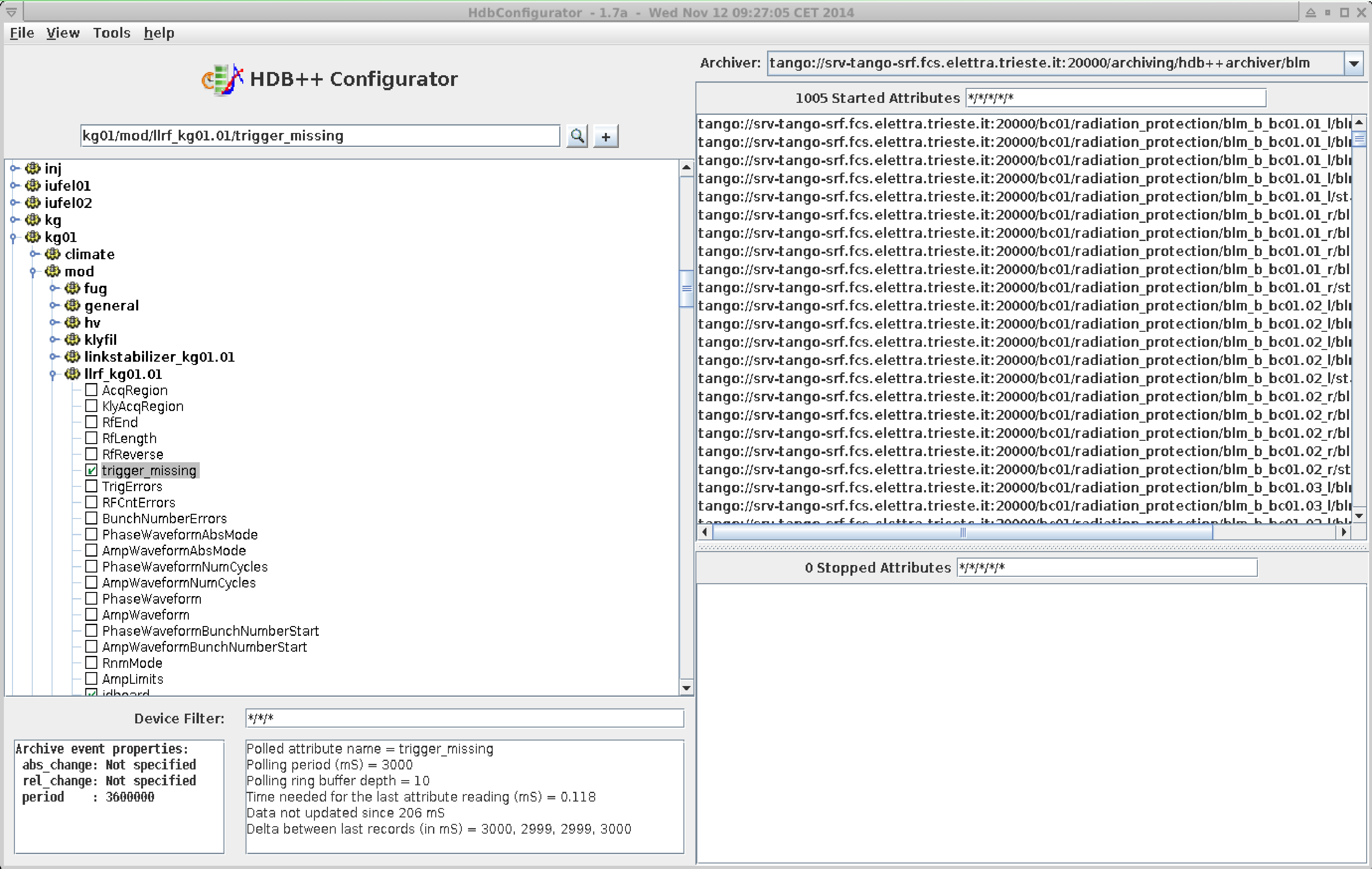
HDB++ (C++)

- One database for slow and fast archiving (up to 1 Khz)
- Support for existing HDB schema on MySQL
- Support for **hdb++ new schema** with improved features (µs timestamp)
- Support for **noSQL** backend (Apache Cassandra)
- 2 Device servers (EventSubscriber, ConfigurationManager)
- **Event** based
- Fast data extraction library
- GUI: HdbConfigurator, qhdbextractor (plotting)
- **Scalability**: same as TANGO, deploy as many DS as you need

TimeMachine

- System restoring tool based on context, HDB++ archived data and extraction library

Historical Database



HdbConfigurator - 1.7a - Wed Nov 12 09:27:05 CET 2014

HDB++ Configurator

kg01/mod/llrf_kg01.01/trigger_missing

- inj
- iufel01
- iufel02
- kg
- kg01
  - climate
  - mod
    - fug
    - general
    - hv
    - klyfil
    - linkstabilizer_kg01.01
    - llrf_kg01.01
      - □ AcqRegion
      - □ KlyAcqRegion
      - □ RfEnd
      - □ RfLength
      - □ RfReverse
      - ☑ trigger_missing
      - □ TrigErrors
      - □ RFCntErrors
      - □ BunchNumberErrors
      - □ PhaseWaveformAbsMode
      - □ AmpWaveformAbsMode
      - □ PhaseWaveformNumCycles
      - □ AmpWaveformNumCycles
      - □ PhaseWaveform
      - □ AmpWaveform
      - □ PhaseWaveformBunchNumberStart
      - □ AmpWaveformBunchNumberStart
      - □ RnmMode
      - □ AmpLimits
      - ☑ idboard

Device Filter: */*/*

```
Archive event properties:
 abs_change: Not specified
 rel_change: Not specified
 period     : 3600000
```

Polled attribute name = trigger_missing
Polling period (mS) = 3000
Polling ring buffer depth = 10
Time needed for the last attribute reading (mS) = 0.118
Data not updated since 206 mS
Delta between last records (in mS) = 3000, 2999, 2999, 3000

Archiver: tango://srv-tango-srf.fcs.elettra.trieste.it:20000/archiving/hdb++archiver/blm

1005 Started Attributes */*/*/*/*

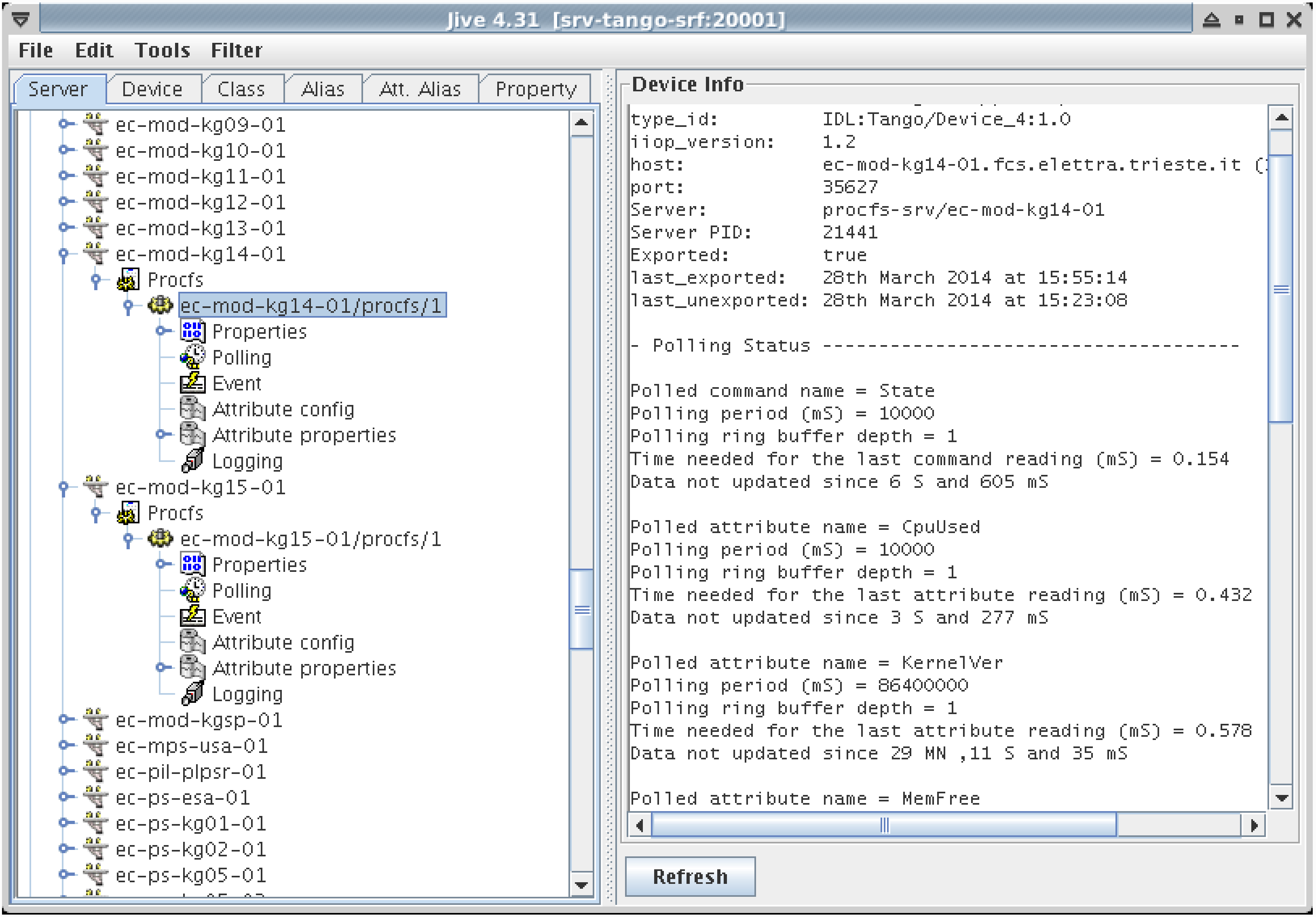
```
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.01_l/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.01_l/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.01_l/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.01_l/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.01_l/st
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.01_r/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.01_r/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.01_r/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.01_r/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.01_r/st
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.02_l/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.02_l/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.02_l/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.02_l/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.02_l/st
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.02_r/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.02_r/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.02_r/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.02_r/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.02_r/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.02_r/st
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.03_l/bl
tango://srv-tango-srf.fcs.elettra.trieste.it:20000/bc01/radiation_protection/blm_b_bc01.03_l/bl
```

0 Stopped Attributes */*/*/*/*

TANGO
Connecting things **together**

Administration: Jive

TANGO database browser and device configuration/administration/testing tool

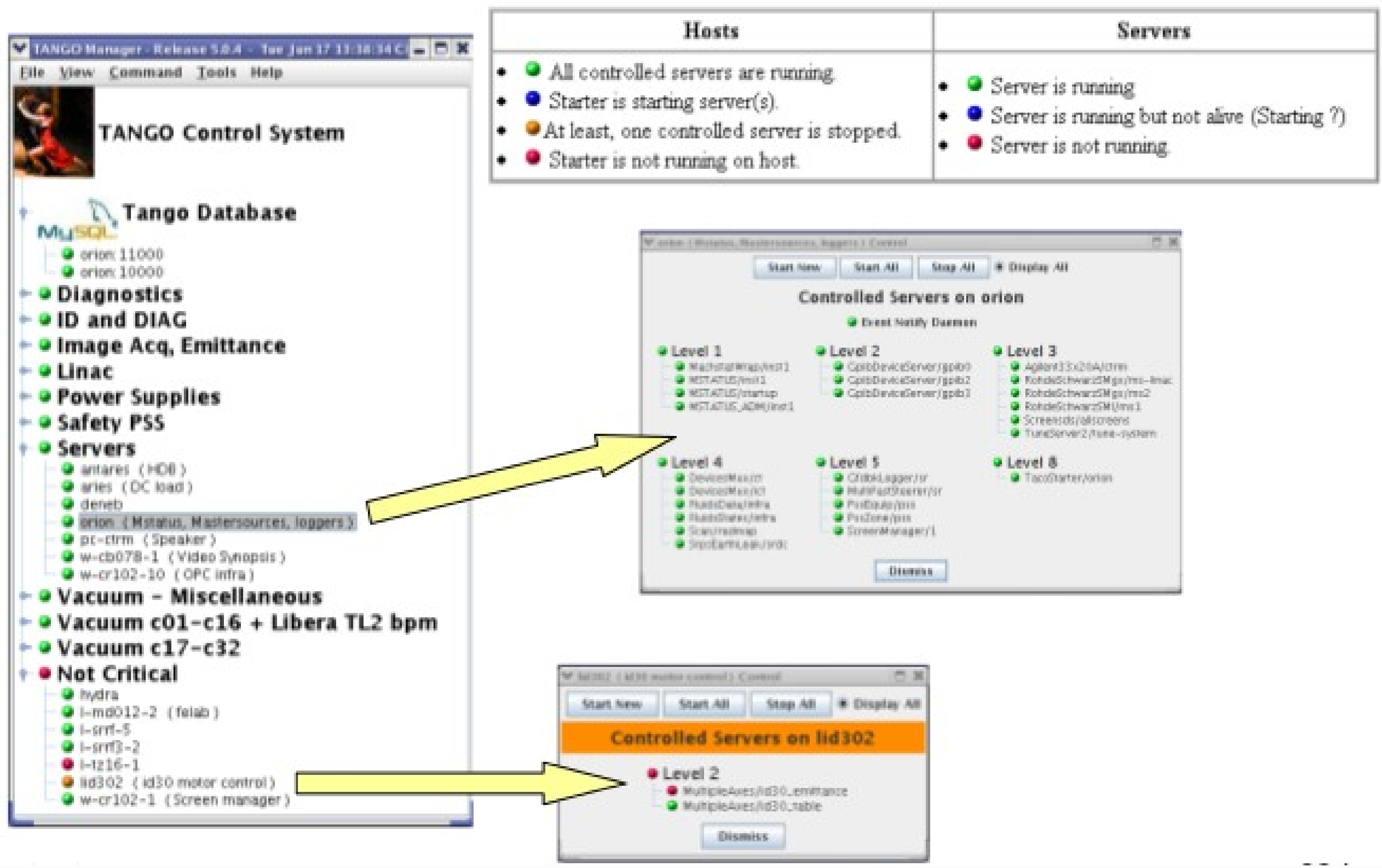


```
ec-mod-kg09-01
ec-mod-kg10-01
ec-mod-kg11-01
ec-mod-kg12-01
ec-mod-kg13-01
ec-mod-kg14-01
  Procfs
    ec-mod-kg14-01/procfs/1
      Properties
      Polling
      Event
      Attribute config
      Attribute properties
      Logging
ec-mod-kg15-01
  Procfs
    ec-mod-kg15-01/procfs/1
      Properties
      Polling
      Event
      Attribute config
      Attribute properties
      Logging
ec-mod-kgsp-01
ec-mps-usa-01
ec-pil-plpsr-01
ec-ps-esa-01
ec-ps-kg01-01
ec-ps-kg02-01
ec-ps-kg05-01
```

```
type_id:          IDL:Tango/Device_4:1.0
iiop_version:     1.2
host:             ec-mod-kg14-01.fcs.elettra.trieste.it (
port:             35627
Server:           procfs-srv/ec-mod-kg14-01
Server PID:       21441
Exported:         true
last_exported:    28th March 2014 at 15:55:14
last_unexported:  28th March 2014 at 15:23:08

- Polling Status -------------------------------------

Polled command name = State
Polling period (mS) = 10000
Polling ring buffer depth = 1
Time needed for the last command reading (mS) = 0.154
Data not updated since 6 S and 605 mS

Polled attribute name = CpuUsed
Polling period (mS) = 10000
Polling ring buffer depth = 1
Time needed for the last attribute reading (mS) = 0.432
Data not updated since 3 S and 277 mS

Polled attribute name = KernelVer
Polling period (mS) = 86400000
Polling ring buffer depth = 1
Time needed for the last attribute reading (mS) = 0.578
Data not updated since 29 MN ,11 S and 35 mS

Polled attribute name = MemFree
```

Administration: Starter/Astor

Starter: TANGO DS to manage device servers on hosts
Astor: control system manager GUI



Hosts	Servers
🟢 All controlled servers are running. 🔵 Starter is starting server(s). 🟠 At least, one controlled server is stopped. 🔴 Starter is not running on host.	🟢 Server is running 🔵 Server is running but not alive (Starting ?) 🔴 Server is not running

TANGO Control System

Tango Database
- 🟢 orion: 11000
- 🟢 orion: 10000

- 🟢 Diagnostics
- 🟢 ID and DIAG
- 🟢 Image Acq, Emittance
- 🟢 Linac
- 🟢 Power Supplies
- 🟢 Safety PSS
- 🟢 Servers
  - 🟢 antares (HDB)
  - 🟢 aries (DC load)
  - 🟢 deneb
  - 🟢 orion (Mstatus, Mastersources, loggers)
  - 🟢 pc-ctrm (Speaker)
  - 🟢 w-cb078-1 (Video Synopsis)
  - 🟢 w-cr102-10 (OPC infra)
- 🟢 Vacuum - Miscellaneous
- 🟢 Vacuum c01-c16 + Libera TL2 bpm
- 🟢 Vacuum c17-c32
- 🔴 Not Critical
  - 🟢 hydra
  - 🟢 l-md012-2 (felab)
  - 🟢 l-srrf-5
  - 🟢 l-srrf3-2
  - 🔴 l-tz16-1
  - 🟠 lid302 (id30 motor control)
  - 🟢 w-cr102-1 (Screen manager)

Controlled Servers on orion

🟢 Event Notify Daemon

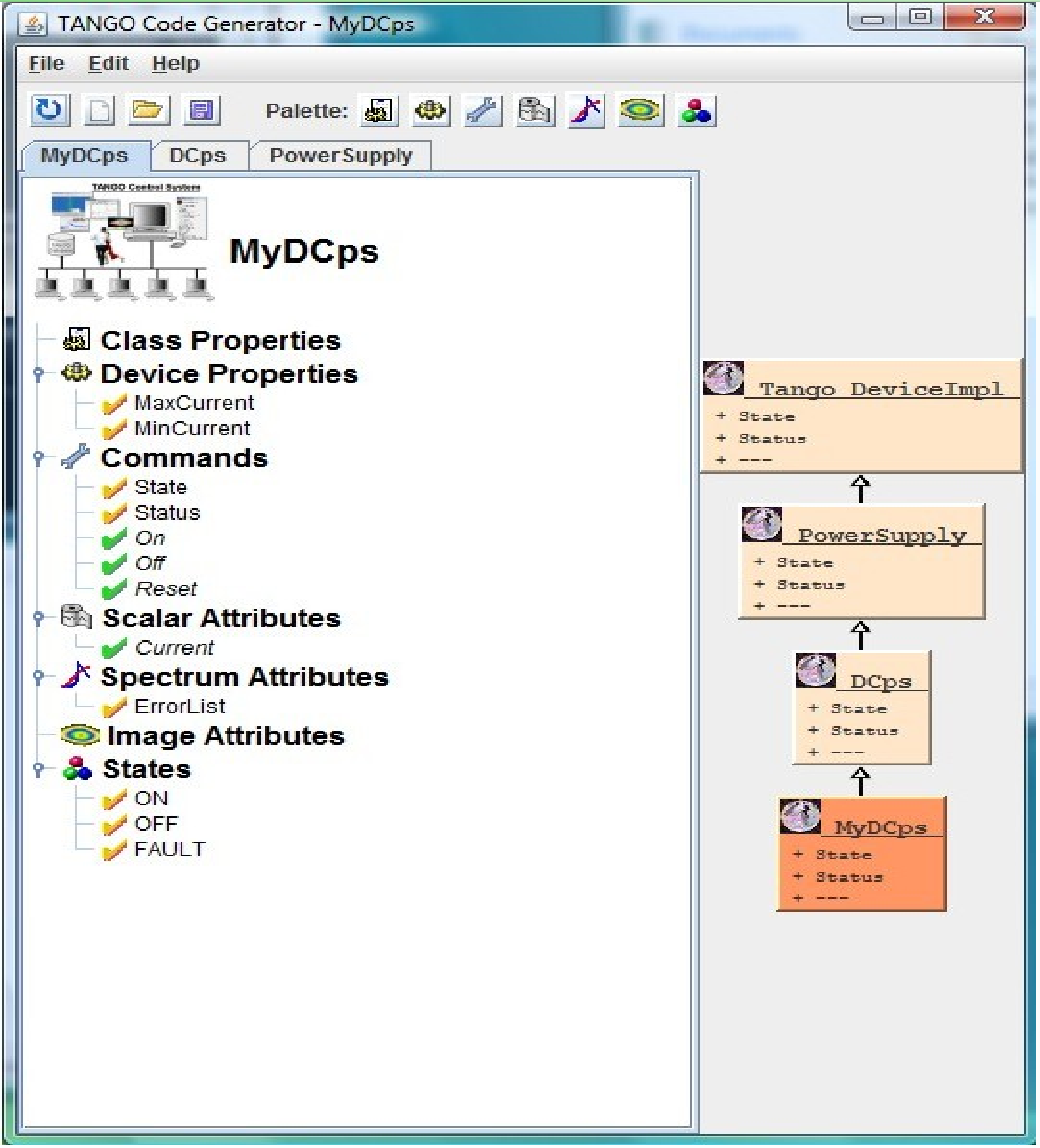
- 🟢 Level 1
  - 🟢 MachstatWrap/inst1
  - 🟢 MSTATUS/inst1
  - 🟢 MSTATUS/startup
  - 🟢 MSTATUS_ADM/inst1
- 🟢 Level 2
  - 🟢 GpibDeviceServer/gpib0
  - 🟢 GpibDeviceServer/gpib2
  - 🟢 GpibDeviceServer/gpib3
- 🟢 Level 3
  - 🟢 Agilent33x20A/ctrm
  - 🟢 RohdeSchwarzSMgx/ms-linac
  - 🟢 RohdeSchwarzSMgx/ms2
  - 🟢 RohdeSchwarzSMl/ms1
  - 🟢 Screends/allscreens
  - 🟢 TuneServer2/tune-system
- 🟢 Level 4
  - 🟢 DeviceMux/rt
  - 🟢 DeviceMux/ct
  - 🟢 FluidsData/infra
  - 🟢 FluidsStates/infra
  - 🟢 Scan/radmap
  - 🟢 SrpcEarthLeak/srdc
- 🟢 Level 5
  - 🟢 GlobalLogger/sr
  - 🟢 MultiFastSteerer/sr
  - 🟢 PssEquip/pss
  - 🟢 PssZone/pss
  - 🟢 ScreenManager/1
- 🟢 Level 8
  - 🟢 TacoStarter/orion

Controlled Servers on lid302

- 🔴 Level 2
  - 🔴 MultipleAxes/id30_emittance
  - 🟢 MultipleAxes/id30_table

Connecting things together

Development: Pogo



MyDCps

- Class Properties
- Device Properties
  - MaxCurrent
  - MinCurrent
- Commands
  - State
  - Status
  - On
  - Off
  - Reset
- Scalar Attributes
  - Current
- Spectrum Attributes
  - ErrorList
- Image Attributes
- States
  - ON
  - OFF
  - FAULT

Pogo is a TANGO class generator

Generates C++, Java and Python Source code and html documentation

The class skeleton is saved in a .xmi file

Well defined areas for programmer's code

GUI: ATK/Jdraw/Synoptic

Application ToolKit: provides a framework to speed up the development of TANGO applications

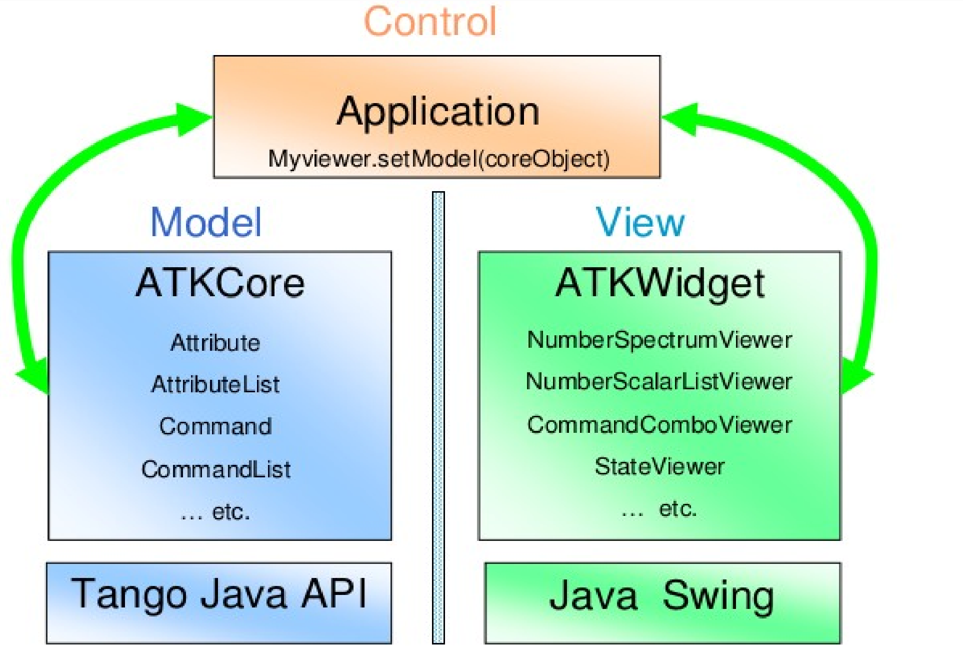
Core of any TANGO Java client

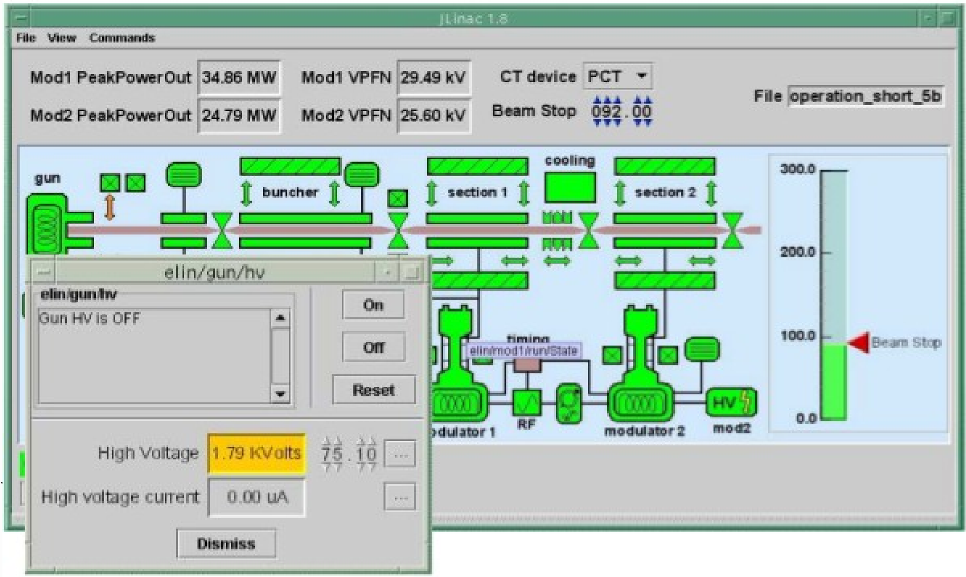
ATKpanel: generic GUI (data introspection)

Use Jdraw to draw the specialized synoptic

Design your own specific ATK application
Using your favorite Java IDE

Final result...





Mod1 PeakPowerOut 34.86 MW
Mod2 PeakPowerOut 24.79 MW
Mod1 VPFN 29.49 kV
Mod2 VPFN 25.60 kV
CT device PCT
Beam Stop 092.00
File operation_short_5b
elin/mod1/run/State

elin/gun/hv
Gun HV is OFF
High Voltage 1.79 KVolts 75.10
High voltage current 0.00 uA

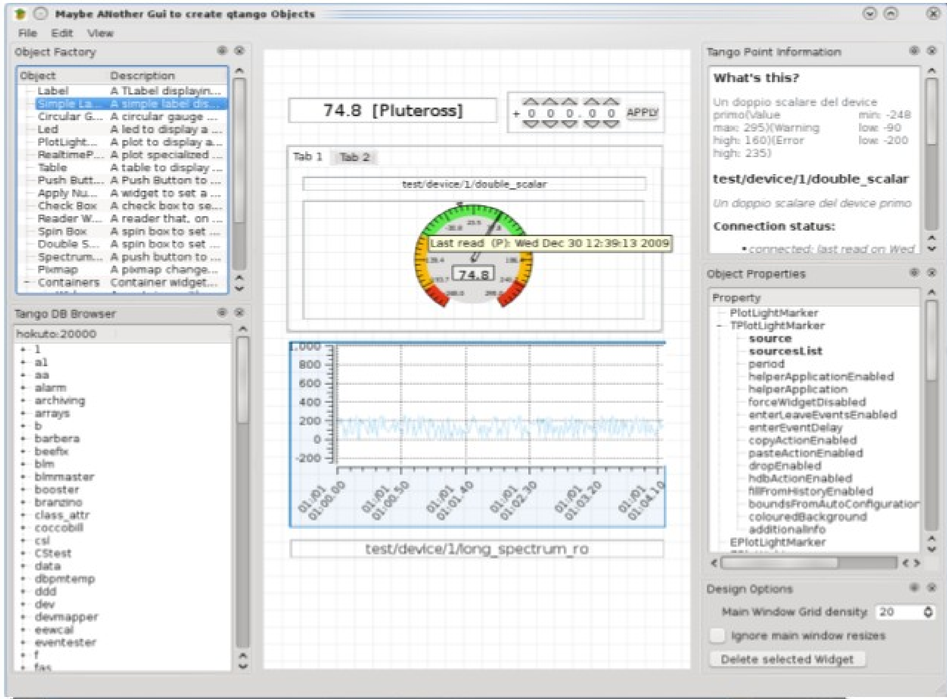
GUI: Qtango/Mango

Qtango

- A multi-threaded framework to develop TANGO applications
- Based on Qt
- API to manage/talk to TANGO devices
- Widgets to draw the GUI
- For programmers

Mango

- An on-line designer to easily create graphical interfaces based on Qtango
- Quick development of simple GUI
- Useful for the device server programmer, the control room operator, the tests, the end-user



Object	Description
Label	A TLabel displayin...
Simple La...	A simple label dis...
Circular G...	A circular gauge ...
Led	A led to display a ...
PlotLight...	A plot to display a...
RealtimeP...	A plot specialized ...
Table	A table to display ...
Push Butt...	A Push Button to ...
Apply Nu...	A widget to set a ...
Check Box	A check box to se...
Reader W...	A reader that, on ...
Spin Box	A spin box to set ...
Double S...	A spin box to set ...
Spectrum...	A push button to ...
Pixmap	A pixmap change...
Containers	Container widget...

hokuto:20000

- 1
- a1
- aa
- alarm
- archiving
- arrays
- b
- barbera
- beefix
- blm
- blmmaster
- booster
- branzino
- class_attr
- coccobill
- csl
- CStest
- data
- dbpmtemp
- ddd
- dev
- devmapper
- eewcal
- eventester
- f
- fas

74.8 [Pluteross]

+ 0 0 0 . 0 0

test/device/1/double_scalar

74.8

Last read (P): Wed Dec 30 12:39:13 2009

test/device/1/long_spectrum_ro

What's this?

Un doppio scalare del device primo(Value min: -248 max: 295)(Warning low: -90 high: 160)(Error low: -200 high: 235)

test/device/1/double_scalar

Un doppio scalare del device primo

Connection status:

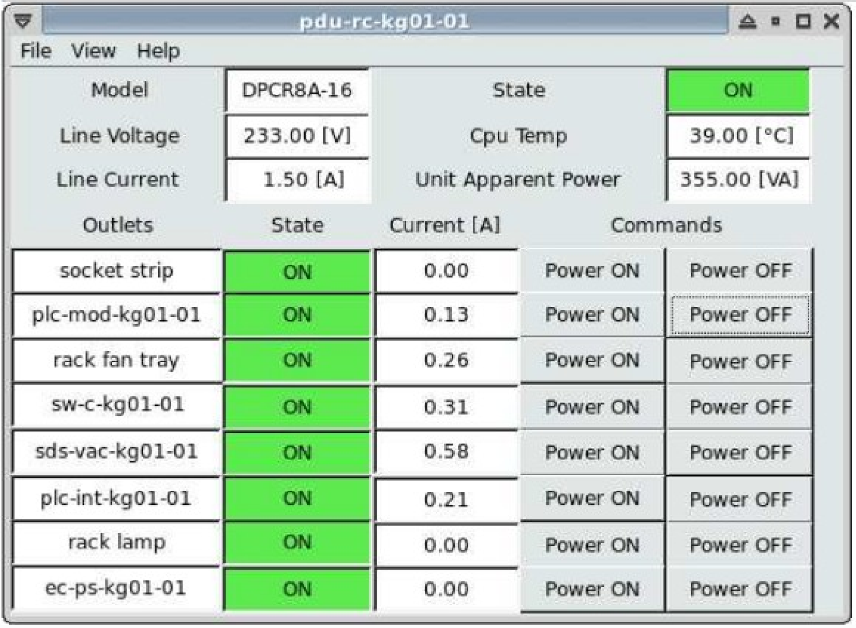
- *connected: last read on Wed*

Property

- PlotLightMarker
- TPlotLightMarker
  - **source**
  - **sourcesList**
  - period
  - helperApplicationEnabled
  - helperApplication
  - forceWidgetDisabled
  - enterLeaveEventsEnabled
  - enterEventDelay
  - copyActionEnabled
  - pasteActionEnabled
  - dropEnabled
  - hdbActionEnabled
  - fillFromHistoryEnabled
  - boundsFromAutoConfiguration
  - colouredBackground
  - additionalInfo
- EPlotLightMarker

Main Window Grid density: 20

□ Ignore main window resizes



Model	DPCR8A-16	State	ON
Line Voltage	233.00 [V]	Cpu Temp	39.00 [°C]
Line Current	1.50 [A]	Unit Apparent Power	355.00 [VA]

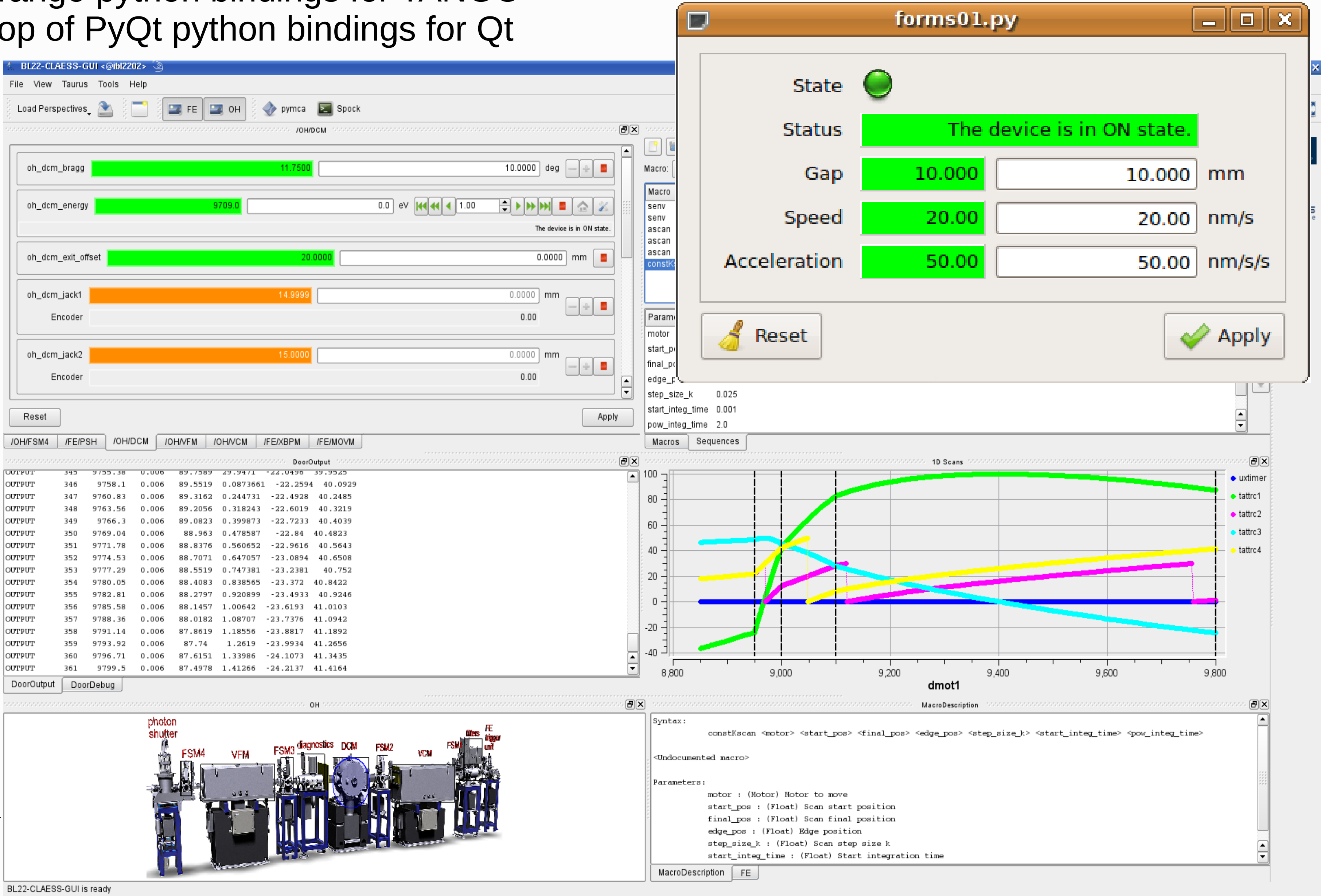
Outlets	State	Current [A]	Commands	
socket strip	ON	0.00	Power ON	Power OFF
plc-mod-kg01-01	ON	0.13	Power ON	Power OFF
rack fan tray	ON	0.26	Power ON	Power OFF
sw-c-kg01-01	ON	0.31	Power ON	Power OFF
sds-vac-kg01-01	ON	0.58	Power ON	Power OFF
plc-int-kg01-01	ON	0.21	Power ON	Power OFF
rack lamp	ON	0.00	Power ON	Power OFF
ec-ps-kg01-01	ON	0.00	Power ON	Power OFF

GUI: TAURUS

A library for connecting client-side apps (CLI/GUI) to TANGO device servers

Based on PyTango python bindings for TANGO

GUI built on top of PyQt python bindings for Qt



oh_dcm_bragg	11.7500	10.0000	deg	
oh_dcm_energy	9709.0	0.0	eV	1.00
oh_dcm_exit_offset	20.0000	0.0000	mm	
oh_dcm_jack1	14.9999	0.0000	mm	
Encoder		0.00		
oh_dcm_jack2	15.0000	0.0000	mm	
Encoder		0.00		

The device is in ON state.

motor	
start_p	
final_p	
edge_p	
step_size_k	0.025
start_integ_time	0.001
pow_integ_time	2.0

```
OUTPUT     345    9755.38   0.006   89.7589  29.9471   -22.0496   39.9525
OUTPUT     346     9758.1   0.006   89.5519  0.0873661  -22.2594  40.0929
OUTPUT     347    9760.83   0.006   89.3162  0.244731  -22.4928   40.2485
OUTPUT     348    9763.56   0.006   89.2056  0.318243  -22.6019   40.3219
OUTPUT     349     9766.3   0.006   89.0823  0.399873  -22.7233   40.4039
OUTPUT     350    9769.04   0.006    88.963  0.478587    -22.84  40.4823
OUTPUT     351    9771.78   0.006   88.8376  0.560652  -22.9616   40.5643
OUTPUT     352    9774.53   0.006   88.7071  0.647057  -23.0894   40.6508
OUTPUT     353    9777.29   0.006   88.5519  0.747381  -23.2381    40.752
OUTPUT     354    9780.05   0.006   88.4083  0.838565   -23.372  40.8422
OUTPUT     355    9782.81   0.006   88.2797  0.920899  -23.4933   40.9246
OUTPUT     356    9785.58   0.006   88.1457  1.00642  -23.6193  41.0103
OUTPUT     357    9788.36   0.006   88.0182  1.08707  -23.7376  41.0942
OUTPUT     358    9791.14   0.006   87.8619  1.18556  -23.8817  41.1892
OUTPUT     359    9793.92   0.006     87.74   1.2619  -23.9934  41.2656
OUTPUT     360    9796.71   0.006   87.6151  1.33986  -24.1073  41.3435
OUTPUT     361     9799.5   0.006   87.4978  1.41266  -24.2137  41.4164
```

```
Syntax:
          constKscan <motor> <start_pos> <final_pos> <edge_pos> <step_size_k> <start_integ_time> <pow_integ_time>

<Undocumented macro>

Parameters:
          motor : (Motor) Motor to move
          start_pos : (Float) Scan start position
          final_pos : (Float) Scan final position
          edge_pos : (Float) Edge position
          step_size_k : (Float) Scan step size k
          start_integ_time : (Float) Start integration time
```

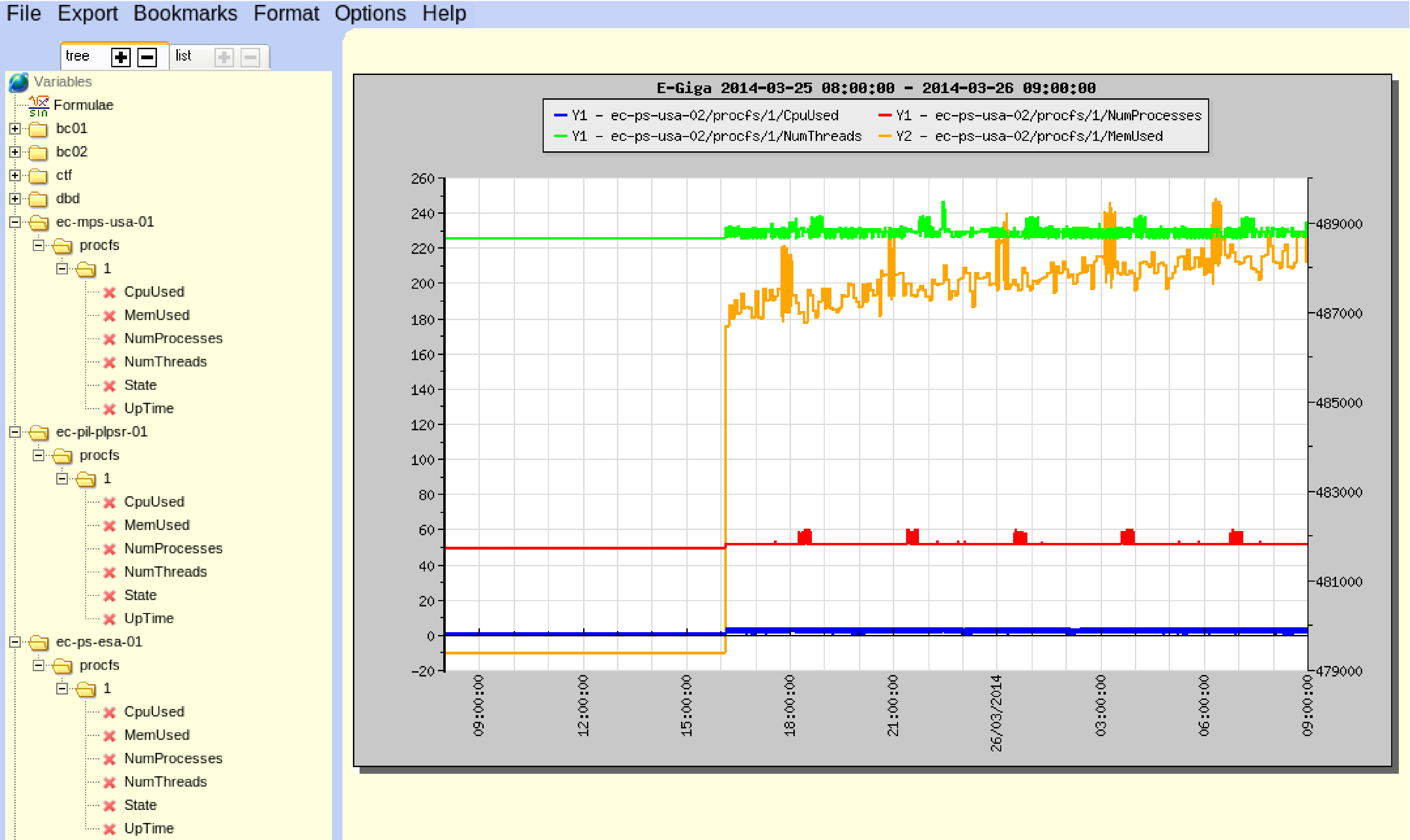
BL22-CLAESS-GUI is ready

State	●		
Status	The device is in ON state.		
Gap	10.000	10.000	mm
Speed	20.00	20.00	nm/s
Acceleration	50.00	50.00	nm/s/s

TANGO
Connecting things **together**

E-giga/Canone

E-Giga: a WEB interface to historical archive data
Canone: a tool to develop WEB interfaces to Tango devices



TANGO bindings

Access TANGO control systems from different high level “programming” environments.

TANGO provides bindings for the following “languages”:

- C language (partial support)
- Matlab (>= R2009b)
  Windows and Linux, 32 and 64 bit
- Octave (>= 3.6.2)
  Windows and Linux, 32 and 64 bit
- LabVIEW 2010 → 2012
  Windows, Linux, MacOSX, 32 and 64 bit
- LabVIEW 2013 (2.0.0 RC2)
  TANGO 8.1.2 with patches; Windows and Linux, 64 bit
- Igor Pro (>= 6.0)
  Windows, Linux, MacOSX, 32 and 64 bit
- Panorama
  Tango 7.2.1, Windows, 32 and 64 bit

Connecting things **together**

Device Hierarchy
TANGO Domains

A Tango control system ~~can~~ must be hierarchically (logically) organized
Devices associated with hardware equipments usually live at lower level
Higher level devices aim to:

- abstract functionalities from mechanisms
- group similar devices
- implement services based on many low level devices (e.g. alarms)

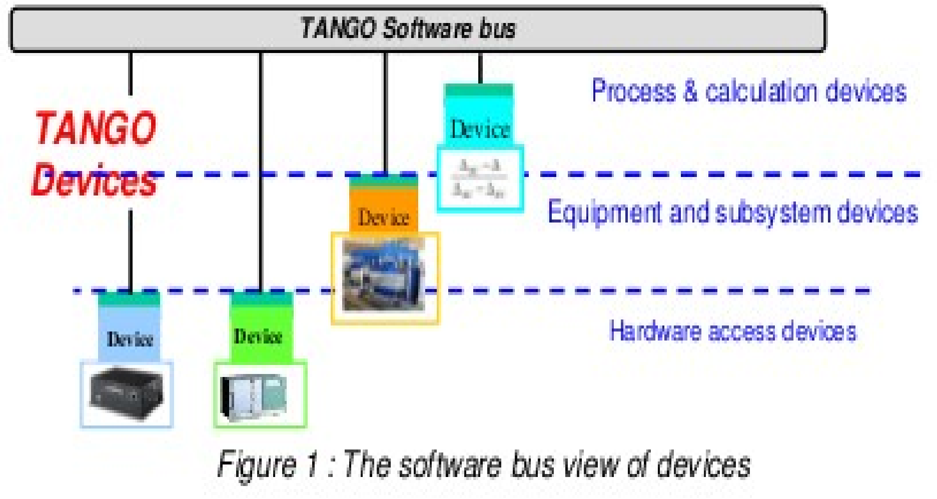


Figure 1 : The software bus view of devices

Each domain is identified by the TANGO_HOST/port couple, e.g. by the TANGO Database
An arbitrary number of devices may belong to a domain, just limited by available memory, processing power, network bandwidth (Operating Database limit ~ 5*10^5 devices)
...but...
Multiple domains **can** be configured in a control system

- Complex systems ~~can~~ **must** be splitted into different Domains
- Each Domain ~~can~~ **must** be hierarchically organized

Multiple domains + Device hierarchy + Peer-2-Peer architecture = unlimited scalability