

THE TANGO CONTROLS COLLABORATION IN 2015

Andrew Götz, Jean-Michel Chaize, Emmanuel Taurel, Pascal Verdier,
Jean-Luc Pons, Tiago Coutinho, Faranguiss Poncet, Reynald Bourtembourg, ESRF, Grenoble, France
Gwenaëlle Abeille, SOLEIL, Gif-sur-Yvette, France, Sandor Brockhauser,
Lajos Jenő Fulop, ELI-ALPS, Szeged, Hungary, Alejandro Vázquez-Otero, ELI-BEAMS, Prague, Czech Republic
Mihail Octavian Cernaianu, IFIN-HH, ELI-NP, Bucharest - Magurele, Romania
Cristina Knapic, Ricardo Smareglia, INAF, Trieste, Italy
Igor Khokhriakov, Helmholtz-Zentrum Geesthacht, Geesthacht, Germany

Abstract

This paper presents the latest news from the TANGO collaboration. TANGO is being used in new domains. The three ELI pillars - ELI-Beamlines, ELI-ALPS and ELI-NP in Czech Republic, Hungary and Romania respectively have selected TANGO for many of their control systems. TANGO will be extended with new features required by the laser community. These features will include nanosecond time-stamping. The latest major release of TANGO V9 includes the following features - data pipes, enumerated types, dynamic commands and forwarded attributes. A prototype REST API (mtango) has been developed. The collaboration has been extended to include the new members and to provide a sustainable source of resources through a collaboration contract. A new website (<http://www.tango-controls.org/>) has been designed which has improved the communication within the community. Finally the updated roadmap is summarised.

THE COLLABORATION

New Sites

The TANGO Controls collaboration has continued to grow. New sites since the last ICALEPCS in 2013 are ELI-BEAMS, ELI-ALPS, ELI-NP, VIRGO EGO and SKA to mention a few. In 2015 two new TANGO-based synchrotrons have been commissioned – Solaris (Poland) and MAX IV (Sweden).

Student Trainings

The number of students attending classes at high schools, universities and training courses has increased. TANGO courses are now available as part of a Bachelor's degree in Grenoble (France), and as part of a Master's degree offered by the University of Szeged (Hungary). TANGO is used for training courses at the EPFL in Lausanne (Switzerland). More students learning TANGO means it is possible to find trainees for TANGO related projects. A group of French students submitted a TANGO project running on a Zynq platform to the Open Hardware 2015 Xilinx student competition. The IAEA has sponsored 3 training courses on TANGO as part of a further education programme in Northern Africa.

TANGO V9

The TANGO approach to development is to continue improving the code base by making regular major releases. The latest major release is version 9. It has been

in beta test at the ESRF since October 2014. The official source code release 9.1.0 was made in October 2015.

Why is it a major release? Firstly because of the addition of a new communication channel – pipes. Secondly because there is a new device interface (Device_5Impl). Thirdly because of new implementations in Java for device servers and in Python with a more Pythonic interface. As always the new release is backward compatible with all previous versions. The following sections summarises the new features which have been added to TANGO V9.

Enum Attributes

A common need in control systems is to allow only a restricted set of values for certain attributes e.g. SINGLE, MULTI, HYBRID filling patterns. This is usually solved in programming languages using enumerated types. This needs special treatment when traversing the network. Version 9 has implemented enumerated attributes to make programmers and users lives easier.

Forwarded Attributes

Attributes are the well defined data exchanged between device servers and clients. In a hierarchical distributed control system system it is common to build hierarchies of device servers. When doing so it is very useful to be able to make some of the attributes of sub-devices available as attributes of other higher level devices. In the past this could only be done by manually re-programming the passing of the attribute of the sub-device in the higher level device. Version 9 has imported the notion of forwarded attributes which require no programming. They only need to be declared in the database. Less code means more reliability.

Pipes

TANGO supports a wide set of data types. For exchanging data these are mostly made up of scalars or arrays of scalars of the same type. This is sufficient in most cases. However there are some cases which have a need to exchange data of varying types coming from the same source. A typical example of this is when returning scanned data from an experiment scanning server. In this case it is useful to mix data types on the same communication channel. A new data communication channel, called pipes, has been introduced to address this problem. Version 9 introduces Tango Pipes for sending and receiving blobs of data of mixed types. Blobs of blobs are supported too.

Polling Thread Improvement

In previous version of TANGO the polling thread which is the main mechanism for generating events, accessed device attributes one by one. This could lead to longer than necessary readout times. Version 9 has modified the polling thread to read multiple attributes in one call. This is more efficient, leads to better performance and fewer errors.

Dynamic Commands

Previous versions of TANGO allowed on dynamic attributes. Dynamic means they are created at runtime as opposed to statically at compile time. Prior to Version 9 dynamic commands were only possible in Java. Version 9 has added the possibility of defining dynamic commands in C++ and Python device classes too.

New Java Device Implementation

For a long time the TANGO Java server implementation was frozen. In 2010 it was un-frozen and completely rewritten using the latest features of Java. Support was added for annotations at compile or runtime. This allows any POJO to be turned into a TANGO device. The following annotations are supported :

```
@Device: class
@Attribute: field
@Command: method
@State: field
@Init: method
@DeviceProperty
```

The latest version of the TANGO Java server and client api support all the new features of TANGO V9 (cf. above).

High Level PyTango API

The Python implementation of Tango is based very closely on the C++ API. This approach is not very natural for Python programmers. Consequently a Tango Enhancement Proposal (TEP1 [1]) was proposed with the following rationale - "It would be nice if non tango experts could create tango device servers without having to code some obscure tango related code. It would also be nice if the tango programming interface would be more pythonic. The final goal is to make writing tango device servers as simple as possible". TEP1 has been implemented as the High Level API (HLAPI) [2] in PyTango 8.1.2. The HLAPI has been implemented as a template in Pogo to generate pythonic Python Tango device servers. The HLAPI is based on PyTango binding. Consequently programmer's have access to all the low level features of PyTango if needed. Here is a simple example on how to write a *Clock* device server using the high level API:

```
import time
from PyTango.server import run
from PyTango.server import Device, DeviceMeta
from PyTango.server import attribute, command
```

```
class Clock(Device):
    __metaclass__ = DeviceMeta
    time = attribute()

    def read_time(self):
        return time.time()

    @command(din_type=str, dout_type=str)
    def strftime(self, format):
        return time.strftime(format)

if __name__ == "__main__":
    run((Clock,))
```

mtango REST API

Growing demand in web and mobile applications that would communicate with TANGO devices was the reason of a request for a new API. A prototype has been designed and implemented following the REST semantics. The API simplifies the access to devices from the browser or mobile clients. Web/mobile clients can be written in JavaScript, Java, Python or any other language that supports http. Security is implemented via basic http authentication and integrated with the TANGO Access Control (TAC). Refer to the documentation [3] for more information. See Fig. 1 for a schematic of the architecture.

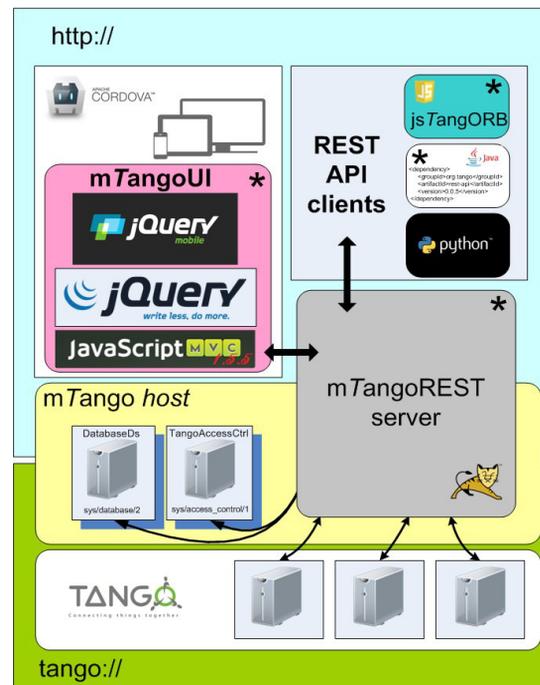


Figure 1 : mtango REST API architecture

This prototype is a starting point for a new TANGO REST API specification (see roadmap section below).

New Java Client Library

A new Java library – ezTangORB [easyTangORB] - has been added. This library facades the standard Java Tango library. ezTangORB provides convenient abstractions to

most commonly used features: reading/writing attributes, commands execution and handling events subscription.

NEW WEBSITE

A new web site <http://tango-controls.org> has been developed. The website contents was designed with the help of a communications specialist while the layout was done by a company specialised in web design. New communication tools like discussion forums, news and event pages are now extensively used. By allowing live exchange between beginners and specialists, these new tools foster knowledge sharing and help newcomers to learn TANGO. The next step is to boost code sharing via the website by setting up a market place (see Roadmap section).

NEW MEMBERS

The TANGO community has continued to grow into new domains – lasers, astronomy, gravitational wave and radio astronomy etc. How TANGO will be used in a selection of the new larger projects are described briefly below.

ELI-BEAMS

ELI-Beamlines [4] will be a high-energy, repetition-rate laser pillar of the ELI (Extreme Light Infrastructure) project. It will be an international facility for both academic and applied research, planned to provide user capability since the beginning of 2018. The main objective of the ELI-Beamlines project is the delivery of ultra-short high-energy pulses (between 10 and 150 fs) for the generation and application of high-brightness X-ray sources and accelerated particles.

The Central Control System (CCS) design is based on TANGO, whilst the Laser Controls (LC) will use a combination of EPICS and LabView. The CCS will talk to the LC using the already existing Tango2Epics Gateway. Matlab and Labview will be extensively used in the experimental chambers for controlling local equipment, though all of them will be integrated into the CCS through adequate Tango device servers.

ELI-ALPS

The primary mission of the ELI-ALPS [5] research facility (Hungary) is to make a wide range of ultrashort atto-second light sources accessible to the international scientific community user groups. Laser driven secondary sources emitting coherent extreme-ultraviolet (XUV) and X-ray radiation confined in atto-second pulses is a major research initiative of the infrastructure.

ELI-ALPS will use TANGO Controls in several places. The control systems and the gateways of the beam transport and secondary source systems will be based on TANGO, as well as the gateways of the turn-key laser source systems delivered by external suppliers. The integration of these systems will be also based on TANGO as well as the central control system.

ELI-NP

The ELI-NP facility [6] will have a dedicated major control system for the HPLS (High Power Laser System), developed in TANGO and a second dedicated control system for the Gamma Beam (GBS) developed in EPICS. The facility requires a modern implementation of a distributed monitoring and control system for the experiments that are proposed in the eight available experimental areas E1-E8.

In the HPLS based experiments, multiple TANGO control systems will be developed and deployed, one for each experiment, for redundancy, safety and better maintenance. In the case of the GBS based experiments, one EPICS framework will handle all the experiments. Additionally, the Laser Beam Transport System which routes the laser beams into the experimental areas will be controlled using TANGO in order to provide a unified interface with the HPLS and the experiments.

In the first stage of the development, the TANGO device servers will only implement access to the HW parameters of the devices and not the logic (e.g. only the means to send commands and receive the status from a motor and not the logic to perform an automatic alignment). Due to a long period of testing, development and fine tuning will be needed to implement the logic. This will be made in the HMI layer and not inside the device servers. As developments advance and stable working solutions emerge, the maths and logic from the supervisory HMI and client HMI will be progressively shifted from LabVIEW and Matlab to the device servers themselves.

Various TANGO implementations have already started at ELI-NP, for interfacing equipment that will be used in the experiments (motor controllers, CCD cameras, etc) and performance tests are being carried out.

INAF

A new use of TANGO was recently implemented by INAF- Trieste Observatory [7] of Trieste to handle the Distributed Large Binocular Telescope Data Archive and the INAF Radio Archive. Using the features offered by TANGO, a configurable infrastructure to ingest and replicate astronomical data over geographically distributed sites was developed. Since the INAF institute joined the TANGO Controls collaboration, the TANGO control system has triggered interest in several astronomical fields, including instrumental control in more complex infrastructures of Radio Astronomy. INAF has foreseen to hold a number of internal TANGO trainings in the near future.

SKA

The Square Kilometer Array (SKA) [8] is currently the most challenging astronomical project world wide. It will be composed of several thousand antennas arrays and dishes distributed over two sites (South Africa and Australia). It presents several technological very challenging issues. The SKA community has shown strong interest in the TANGO control system for the implementation of several parts of the project. A large part

of the consortium in charge of controls met in Trieste March 2015 to discuss a common control system infrastructure. TANGO is currently considered the most complete, well suited and modern framework capable of fulfilling all the requirements. Two trainings were organized in order to learn and understand the paradigms and features of TANGO. Presentations, meetings and a webinar on TANGO are planned within the SKA community.

ARCHIVE DATABASE

A large project called HDB++ has been started to improve the archive database in TANGO. The new database supports SQL and NoSQL type databases. It offers orders of magnitude more in time resolution. The HDB++ project is described in detail in two other papers in these proceedings ([9] and [10]).

COLLABORATION CONTRACT

To date the TANGO collaboration is governed by a Memorandum of Understanding. Institutes who have signed the memorandum form the Executive Committee (EC). Major decisions about TANGO are taken by the EC. In order to ensure the sustainability of TANGO the MOU will be replaced by the TANGO Controls Collaboration in 2016. The collaboration will be governed by a so-called collaboration contract. Institutes who sign the contract will commit to financing the development of TANGO and participate in the decision making process. Two types of members will be recognised – core members and contributing members. All members contribute financially to maintaining TANGO. Core members contribute in addition code to the core. The new organisation does not change the free open source nature of TANGO i.e. TANGO remains freely available and open source for everyone. The new organisation will ensure the sustainability of TANGO by financing resources to work on tasks of common interest like the roadmap.

ROADMAP

TANGO is a mature and reliable toolkit to build distributed control systems for installations which need to run for the next 10 to 20 years. For this reason TANGO needs a roadmap which will ensure its future evolution and stay a good choice for the coming years. At the TANGO meeting in May held at Solaris in Krakow (Poland) the community gave their input on the essential features of the current roadmap. Input was provided via email and discussed during an interactive session. The results presented in [11] are summarised here.

(1) **Improve Documentation** – re-factor and consolidate documentation, write a cook book of recipes and concepts, (2) **Move to Git** – move source code repository from svn to git, (3) **Remove CORBA completely** – introduce and abstract transport layer and replace the underlying synchronous synchronous CORBA protocol with ZMQ, (4) **Grow the community**, (5) **REST API** – implement a REST API for TANGO, (6) **Web browser application** – implement a device browser for the web, (7) **Secure encryption** – implement a secure

encrypted protocol for public networks, (8) **Database performance** – improve to remove bottlenecks for memorised attributes, (9) **Device class Marketplace** – implement a marketplace to improve code sharing, (10) **Long Term Support** – provide upgrades for older versions, (11) **Tango Virtual Machine** – upgrade virtual machine to latest versions, (12) **Auto-generate Unit tests** – Pogo to generate unit tests automatically, (13) **SysML support** – add support for using SysML to specify device servers, (14) **Replace Boost.Python** – replace boost.python in PyTango with a lighter weight solution.

A number of the features on the roadmap have already been implemented or are in the process of being implemented e.g. 5, 8, 11. Others (2, 9, and 12) will be addressed in the short term. The remainder require more resources and will be implemented as part of the next major release. The Collaboration Contract will be extremely helpful in finding resources to implement the roadmap.

CONCLUSION

TANGO Controls is a growing collaboration for building distributed control systems with a clear roadmap for the future. The new members of the collaboration will provide a big boost to the community in terms of new applications and developments. The new collaboration contract will help ensure the sustainability of TANGO for the next years.

ACKNOWLEDGEMENT

The TANGO Controls community for their ideas, input and innovative use of TANGO.

REFERENCES

- [1] Tango Enhancement Proposal 1 - TEP1: http://www.esrf.eu/computing/cs/tango/tango_doc/ke_rnel_doc/pytango/latest/tep/tep-0001.html#pytango-tep1
- [2] PyTango High Level API – HLAPI: http://www.esrf.eu/computing/cs/tango/tango_doc/ke_rnel_doc/pytango/latest/server_api/server.html
- [3] mtango REST API documentation: <https://bitbucket.org/hzgwpm/mtango/wiki/browse/>
- [4] ELI-BEAMS: <http://www.eli-beams.eu/>
- [5] ELI-ALPS: <http://www.eli-alps.hu/>
- [6] ELI-NP: <http://www.eli-np.ro/>
- [7] INAF – Trieste: <http://www.oats.inaf.it/>
- [8] The Square Kilometer Array – SKA: <https://www.skatelescope.org/>
- [9] L.Pivetta et al., “HDB++: A New Archiving System for TANGO”, WED3004, ICALEPCS 2015, these proceedings.
- [10] R.Bourtembourg et al., “How Cassandra Improves Performances and Availability of HDB++ Tango Archiving System”, WEM310, ICALEPCS 2015, these proceedings.
- [11] Tango Roadmap presentation at Tango meeting - http://ftp.esrf.eu/pub/cs/tango/meetings/2015_may_solaris/Andy_Gotz_Roadmap_presentation.pdf