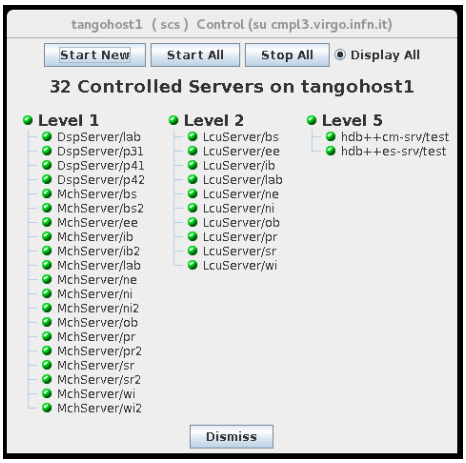
TANGO usage @ Advanced Virgo SCS

Giovanni Cerretani[1] and Luca Rei[2]

[1]Università di Pisa, INFN Sezione di Pisa
[2]INFN Sezione di Genova

July 14, 2015



Introduction

Gravitational waves, predicted on the basis of the General Relativity, are ripples in the curvature of space-time that propagate as a wave. The passage of a gravitational wave induces tiny oscillations in the relative separation between two test masses, that can be measured. Nevertheless these oscillations are extremely small, so that only a very sensitive detector is able to measure them. The Advanced Virgo project is a major upgrade of the $3\,\mathrm{km}$-long interferometric gravitational wave detector Virgo, with the goal of increasing its sensitivity by about one order of magnitude in the whole detection band. It's expected to

Figure 1: 12 Udspt boards in a uTCA chassis. On the right, the NAT-MCH switch. There are $800\,\mathrm{GFLOPS}$ available inside the chassis.

have a maximum strain amplitude sensitivity of $4 \times 10^{-24} \frac{1}{\sqrt{\mathrm{Hz}}}$ at $\sim 300\,\mathrm{Hz}$. In other words this means that it will be able to detect a relative displacement between mirrors of about $10^{-20}\,\mathrm{m}$, by averaging for one second. This sensitivity should allow to detect several tens of events per year.

Among the various ongoing updates, an important improvement is represented by the new electronics used to control the Superattenuators, complex mechanical structures that insulate optical elements from seismic noise by a factor 10^{15} at at $10\,\mathrm{Hz}$. Using the information of several inertial sensors, a digital control system called Suspension Control System (**SCS**) keeps these structures as stable as possible also at lower frequencies.

A new board for the Superattenuator control, called Udspt, has been designed and produced by the INFN Pisa group of Virgo. It incorporates analog-to-digital and digital-to-analog converters, a Field Programmable Gate Array (FPGA) and a Digital Signal Processor (DSP) into a single unit.

The computer power of the eight-core DSP inside the Udspt amounts to $8.4\,\mathrm{GFLOPS}$ per core for double precision floating point: it is enough to make it able to handle every single part of the Superattenuator control. It performs the computation of feedback forces, and it is used, for example, to synthesize sine wave to drive the coils of the inertial sensors, as well as to demodulate their output.

Udspt boards are installed in uTCA crates, and represent the core of the Suspensions Control System. There are 20 crates in Advanced Virgo, that means ~200 Udspt boards.

TANGO has been chosen as framework for the Software Supervisor of the

Suspension Control System (SCS) of Advanced Virgo.

System summary

TANGO Version	8.1.2
OS of the TANGO_HOST	Scientific Linux 6
TANGO Database version	MySQL 5.6.21
Database engine	ndbcluster (instead of MyISAM)
Archiving Tool	Hdb++ 0.90 with the new schema
Number of devices	~300
Server code	C++ and Python
Clients	PyTANGO and PyQt4; TangORB for Android.

TANGO Version

The version of TANGO used for the SCS is the 8.1.2, build from sources using Scientific Linux 6.

Build instructions

TANGO has been compiled on Scientific Linux 6 with the default options. The only problem found in building steps was on configuration script: for some reasons, during (an optional) MySQL test connection a library was not found. We solve this issue passing `LIBS=-lrt` in front of configure script.

TANGO Database

The TANGO MySQL is running in a clustered database with **ndbcluster** as engine, instead of the default MyISAM. This choice has been done because ndbcluster is a high-availability, high-redundancy version of MySQL adapted for the distributed computing environment, and allows us to improve reliability and failure recovery. The cluster is currently maintained by the Virgo computing team.

Patches

The Database device server released whith TANGO 8.1.2 does not support ndbcluster, because it passes `NULL` values to a `NOT NULL` column. While MyISAM and InnoDB use the default value when passing a `NULL` value to a `NOT NULL` column, ndbcluster does not have the same behavior and the queries fail. As a workaround, we've replaced all the occurrences of

```
1 "updated=NULL, accessed=NULL"
```

with

```
"updated=NOW(), accessed=NOW()"
```

in the queries in the Databased sources. This problem should be resolved with TANGO 9.[1]

Archiving tool

We have installed **Hdb++** (release 0.90.0) to save the values of some key attributes. The system is event based, and the data are stored another MySQL database with, of course, **ndbcluster** as engine. The old Hdb is not compatible with our infrastructure, due to some privileges to the database user that cannot be granted to the user.

Access control

The access control is used, to allow the operator to access only to the high-level devices.

TANGO clients

The clients have been written in Python, using **PyTANGO** and **PyQt4**. Some experiments have been done also with Android: a very simple client running in a Nexus 5 can talk with a TANGO Device, using **TangORB for Android**.

TANGO devices and architecture

We have three classes of low-level TANGO devices. Then, there are several other high-level TANGO devices, called *Device supervisor*, that collect the information of the lower level system in order to give an overall status of the system. These *Device supervisors* are mainly coded in Python.

NAT MicroTCA Carrier Hub

There is a MicroTCA Carrier Hub (MCH) produced by NAT Europe for each uTCA chassis. The NAT-MCH switches accept communications through IPMI and telnet. We have integrated telnet and Ipmitool within the TANGO class. Ipmitool[2] is a C-coded open source interface for IPMI.

[1] http://www.tango-controls.org/forums/topic/70/?page=7#post-480
[2] http://sourceforge.net/projects/ipmitool/

Info

- Language: C/C++
- Number of devices: ~20

Udspt

The class interfaces with the DSP installed inside the Udspt boards, and allows to check its status and to perform several operations on them.

Info

- Language: C/C++
- Number of devices: ~200

Galil DMC-4183

The DMC-4183 motion controller is going to be controlled with the GalilDMC[3] TANGO classes, developed at Desy.

Info

- Number of devices: ~60

[3] http://www.esrf.eu/computing/cs/tango/tango_doc/ds_doc/tango-ds/Motion/index.html