Tango8 + Continuous Integration at ALBA

Sergi Rubio Manrique

(a wish of) Tango8 + Continuous Integration at ALBA

Sergi Rubio Manrique

Tango8@ALBA:

We are eager to get rid of notifd due to its memory leaks and CORBA exception issues.

Plan:

- Upgrade of laboratories (ongoing)
- Upgrade of BL09 (started, April-June)
- Upgrade of other 6 beamlines (July-August)

Tango 8 Upgrade

- Platforms: openSuse 11.1/12. on 32/64 bit.
- We had some problems with bug patches of Tango 8.0.5 and PyTango 8.0.2.
- We found that for some reason Tango linked to zmq 64 bits instead of 32.
- Problems hard to detect easily as some devices crashed after some (20+) seconds running.
- Finally solved using the right revision of Tango/PyTango for the builds. Now waiting for Tango/PyTango 8.1

Tango 8 upgrade

- Laboratories Tango Host DB upgraded
- Java archivers still Tango 7
- Vacuum servers and PLC's upgraded (including Modbus and SerialLine device servers)
- Ongoing (solving compilation issues): adlink cards, cameras, yat4tango, motion servers

BL09 upgrade:

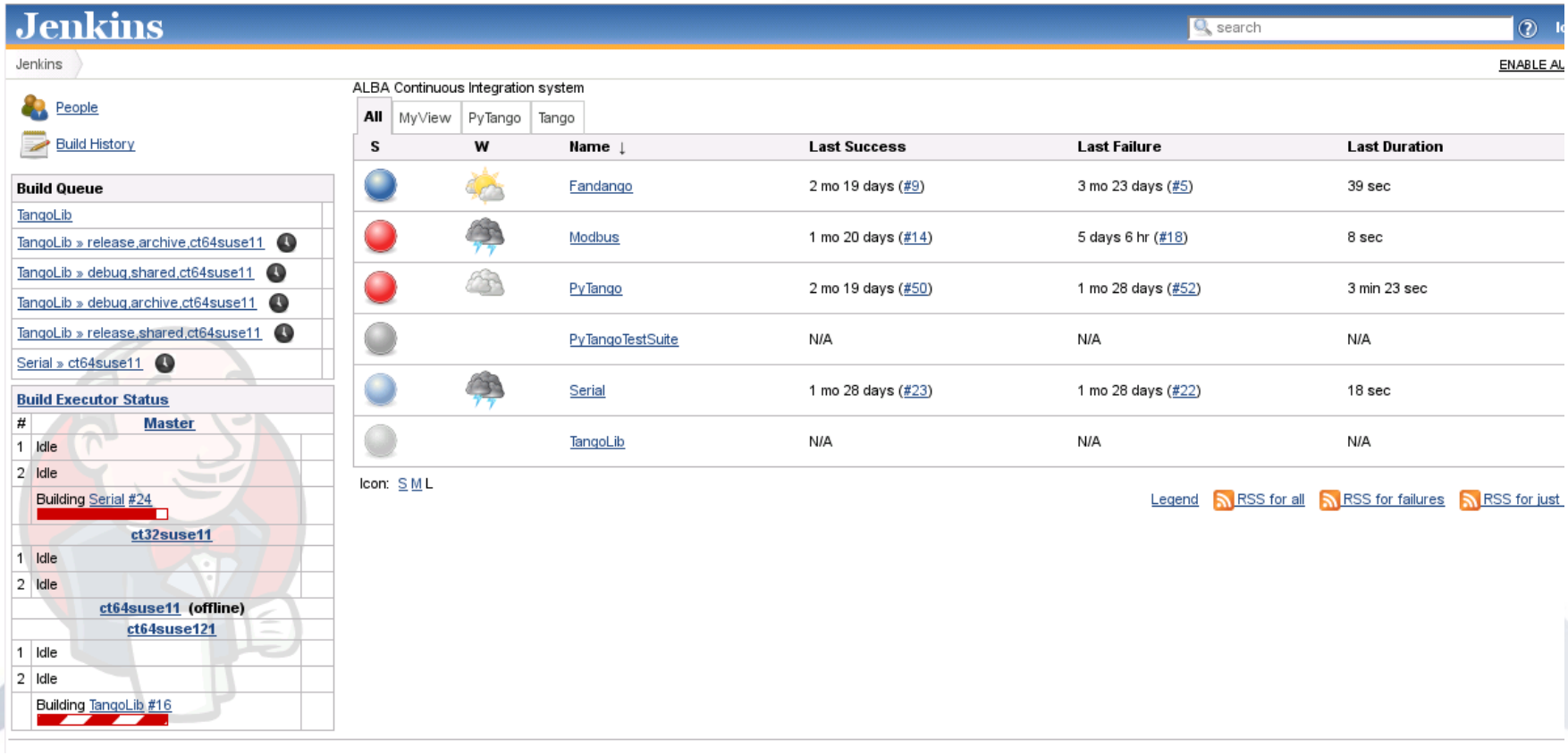
- 3 out of 6 controls machine upgraded to Tango8.
- All Vacuum/EPS/Alarms/Database devices migrated to Tango8 + 1 client machine
- The industrial PC with DAQ cards and Sardana is still Tango7 + 1 client machine
- All client/server combinations with Tango 7/8 and notifd/no-notifd were tried.
- And everything worked! Events worked well between T8 servers and T7 clients and viceversa.
- Sardana not tested yet because we wanted to test it on the optics lab first, actually still dealing with DAQ and Cameras compilation issues.

Jenkins

- We started to use Jenkins due to our collaboration through Tomasz in the CI&Test at the ESRF.
- Our plan is to have a Jenkins project for each controls software package used at ALBA.
- Testing to be integrated in Jenkins as part of the packaging project

Jenkins Status

Continuous Integration scripts prepared for 10+ classes, but still learning the proper way of managing Jenkins and test policies
(Stable vs Last, for current project and dependencies; client testing?)



CI on Shared / Orphan projects

We want to include in Jenkins all the source packages we use; and it includes a lot of projects from external sources (e.g. you!).

To our projects we will add to svn a ./ci/ALBA folder with our build.sh, Makefiles and .opt files.

What to do with other institutes projects? We do not want to write on anyone's folder so we keep a /jenkins/ci/ $PROJECT folder with our customized build.sh

There's any oficial TANGO guideline for .opt files?

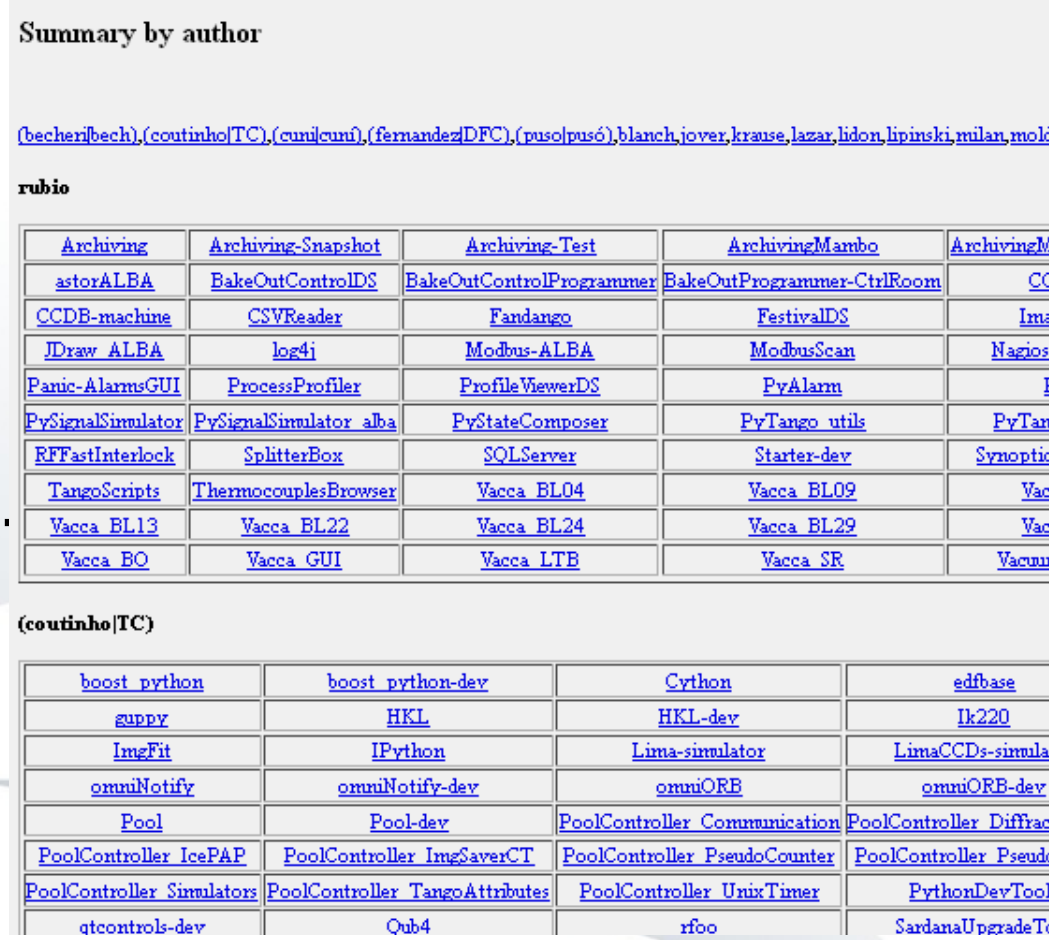
(using everybody the same name for _ROOT/_PATH/_FOLDER/_whatever)

Package database

Reports generated crossing Blissinstaller and Tango databases.

Summaries sorted by host, author, version, repository, …

Useful to detect “orphan” projects and/or focus testing on an specific subsystem.



Summary by author

(becheri|bech),(coutinho|TC),(cuni|cuní),(fernandez|DFC),(puso|pusó),blanch,jover,krause,lazar,lidon,lipinski,milan,mold

rubio

Archiving	Archiving-Snapshot	Archiving-Test	ArchivingMambo	ArchivingM
astorALBA	BakeOutControlDS	BakeOutControlProgrammer	BakeOutProgrammer-CtrlRoom	C
CCDB-machine	CSVReader	Fandango	FestivalDS	Ima
JDraw_ALBA	log4j	Modbus-ALBA	ModbusScan	Nagios
Panic-AlarmsGUI	ProcessProfiler	ProfileViewerDS	PyAlarm	
PySignalSimulator	PySignalSimulator_alba	PyStateComposer	PyTango_utils	PyTan
RFFastInterlock	SplitterBox	SQLServer	Starter-dev	Synoptic
TangoScripts	ThermocouplesBrowser	Vacca_BL04	Vacca_BL09	Vac
Vacca_BL13	Vacca_BL22	Vacca_BL24	Vacca_BL29	Vac
Vacca_BO	Vacca_GUI	Vacca_LTB	Vacca_SR	Vacuu

(coutinho|TC)

boost_python	boost_python-dev	Cython	edfbase
guppy	HKL	HKL-dev	Ik220
ImgFit	IPython	Lima-simulator	LimaCCDs-simula
omniNotify	omniNotify-dev	omniORB	omniORB-dev
Pool	Pool-dev	PoolController_Communication	PoolController_Diffrac
PoolController_IcePAP	PoolController_ImgSaverCT	PoolController_PseudoCounter	PoolController_Pseud
PoolController_Simulators	PoolController_TangoAttributes	PoolController_UnixTimer	PythonDevTool
qtcontrols-dev	Qub4	rfoo	SardanaUpgradeT

Testing Device Servers

Jenkins will tell us if a commit broke something based on:

- basic tango features tests already in Tango TestSuite?
- test for specific changes between revisions?
- a dynamic TestSuite to crosscheck between releases?

Basic testing script for Tango device servers:

- Testing both dependencies and dependent projects/libraries.
- create instance, launch (sometimes there are HW dependencies!), get expected state/status.
- CopyCatDS (fandango.PySignalSimulator) to copy/emulate communications (Serial/Modbus/Socket); dynamic attrs/commands + pickle history

Encourage Dogfooding

- The only way of having effective testing is using it daily.
- So Jenkins+Testing will be useful only if it is fully integrated in our packaging process.

Questions?

Thank you for your attention