



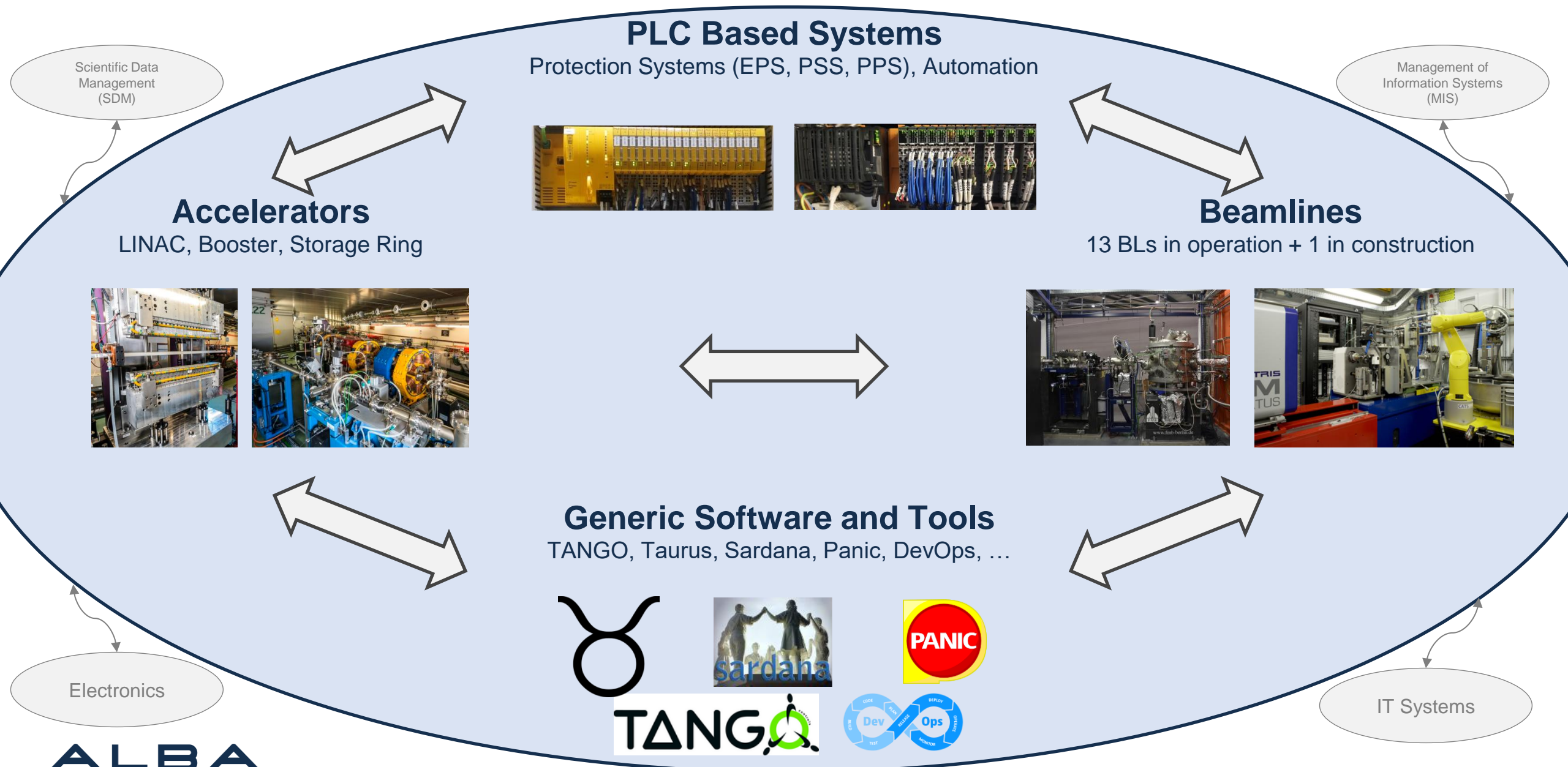
ALBA Controls Status Update

Zbigniew Reszela on behalf of Controls Section

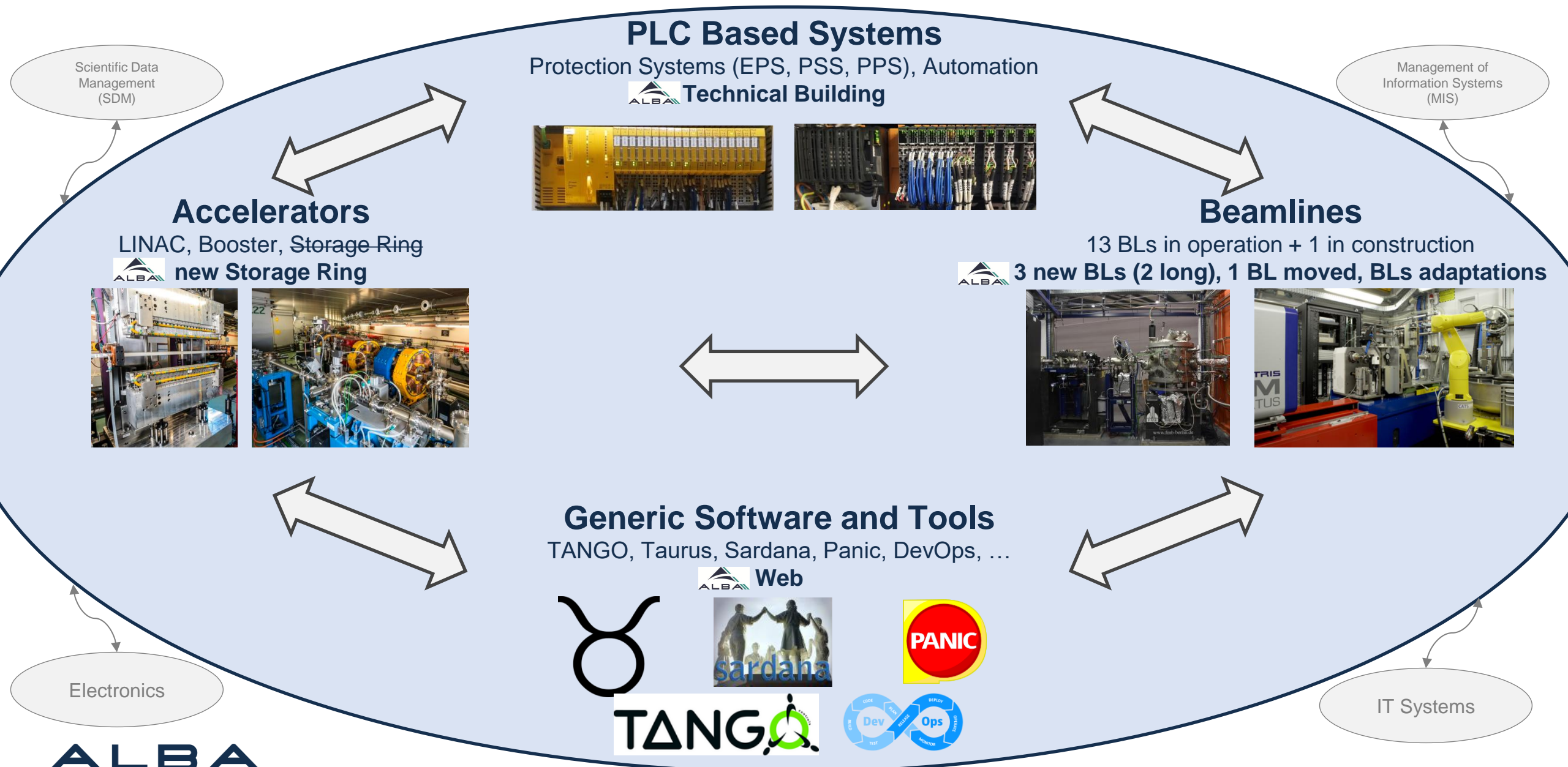
40th Tango Community Meeting – 8/06/2026



ALBA Controls today



ALBA Controls *tomorrow*



PLC Based Systems
 Protection Systems (EPS, PSS, PPS), Automation
 ALBA Technical Building



Accelerators

LINAC, Booster, Storage Ring
 ALBA new Storage Ring



Beamlines

13 BLs in operation + 1 in construction
 ALBA 3 new BLs (2 long), 1 BL moved, BLs adaptations



Generic Software and Tools

TANGO, Taurus, Sardana, Panic, DevOps, ...

ALBA Web



Scientific Data Management (SDM)

Management of Information Systems (MIS)

Electronics

IT Systems

Accelerators Projects



New DLLRF

New Digital Low-Level RF (DLLRF) system:

- DLLRF controls the amplitude, phase, and frequency of RF signals.
- New DLLRF is developed by the SAFRAN company, and based on FPGA in MTCA format. FPGA program is developed by ALBA RF team.
- The Tango DS is running with **Tango 10.1** on custom Linux for the ARM platform (build w/ buildroot).

Due to the systemd identified issue regarding DNS resolution in the host we suffered from CORBA timeouts when writing memorized attributes into the Tango DB and on server startup. Our workaround meanwhile the issues is solved by systemd: avoid using DNSSEC and DNSOverTLS.

- The DS and Taurus GUIs are highly dynamic, they are generated based on YAML configuration files with memory addresses to access and GUI configuration.



Control of the ALBA DLLRF in MTCA format

```
Primary: [
  {name: Kp, access: read_write, dataType: short,
    max_value: 128, min_value: 0,
    read_address: 0, formula: f4,
    gui_tab: control, write_address: 0, unit: ''},
  {name: Ki, access: read_write, dataType: long,
    max_value: 32767, min_value: 0,
    read_address: 1, formula: f1,
    gui_tab: control, write_address: 2, unit: ''},
  ...
]
```

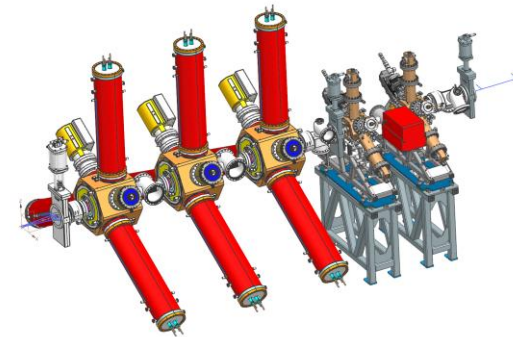
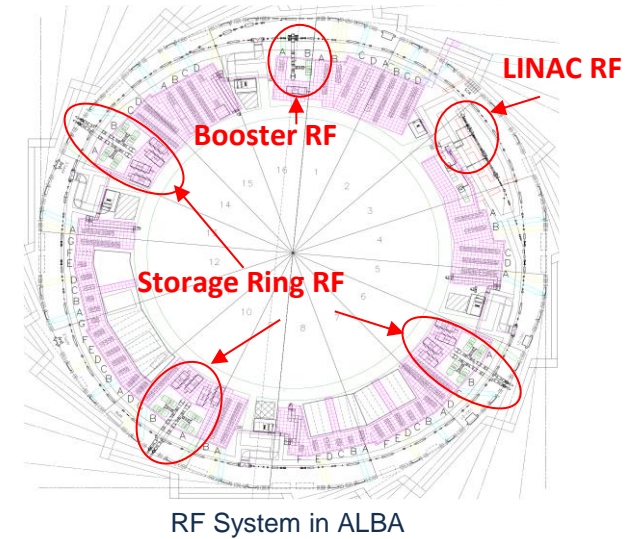
DS and GUI YAML configuration file



Taurus GUI for DLLRF

3rd Harmonic Cavity System (3HC):

- 3HC system (1.5 GHz) is being installed at ALBA and will be used for bunch lengthening to increase the lifetime in ALBA II.
- The 3rd HC will be installed this year.
- Currently we are integrating 3rd HC into the Equipment Protection System (EPS), deploying the DLLRF, and soon will integrate the SSPA into Tango via OPC UA communication.



2 x RF Plants in ALBA II (SR):
3 x FC + 2 x 3HC

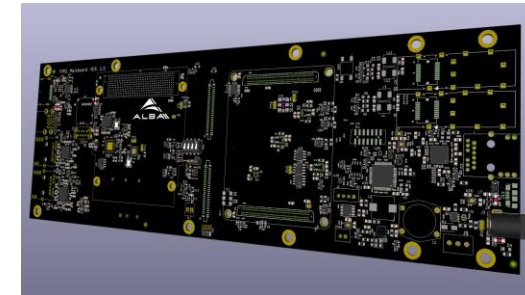
New FIM

New Fast Interlock Module (FIM2):

- FIM manages the alarm conditions of an RF plant to stop it and inform other systems in a short time (~200 ns).
- FIM2 is being developed by ALBA Electronics group, and based on **KRIA K26 SoM** with Zynq UltraScale+ SoC (FPGA + ARM).
- We have tried the custom Linux distribution from Xilinx (build w/ Yocto).
- We will evaluate embedding Tango DS using **meta-tango** layer from **Voltumna Linux** (Elettra).



KRIA K26 SoM

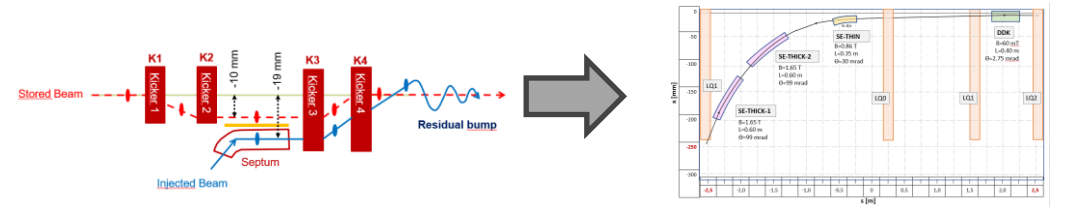


FIM2 main board (w/o KRIA K26)

New Kicker

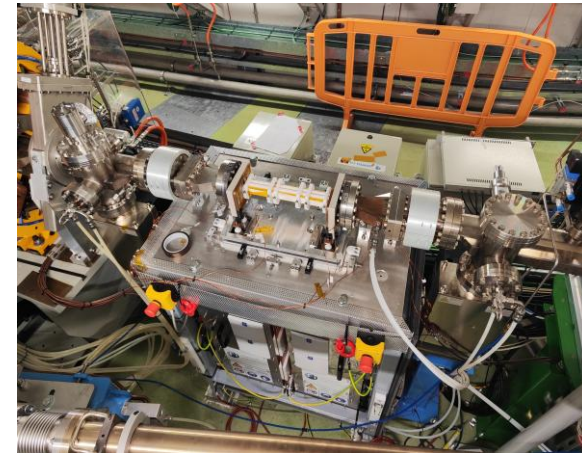
Dual Dipol Kicker (DDK):

- A DDK is a type of Non-Linear-Kicker (NLK) magnet designed by the ALBA Beam Dynamics group.
- The DDK plays the equivalent role than the 4-kicker bump scheme, but it provides better performance.
- The Pulsed Power Supply (PPS) and its Magnet Control Unit (MCU) were developed by the RI company.
- The MCU is based on a Siemens PLC and expose two remote communication interfaces: socket with custom protocol, and the OPC UA server.
- We have adapted the pulsed magnets Tango DS (socket) and developed another one, as a pilot project, for the OPC UA.



Injection scheme in ALBA

Planned injection scheme in ALBA II



DDK and its two PPS installed in ALBA SR.



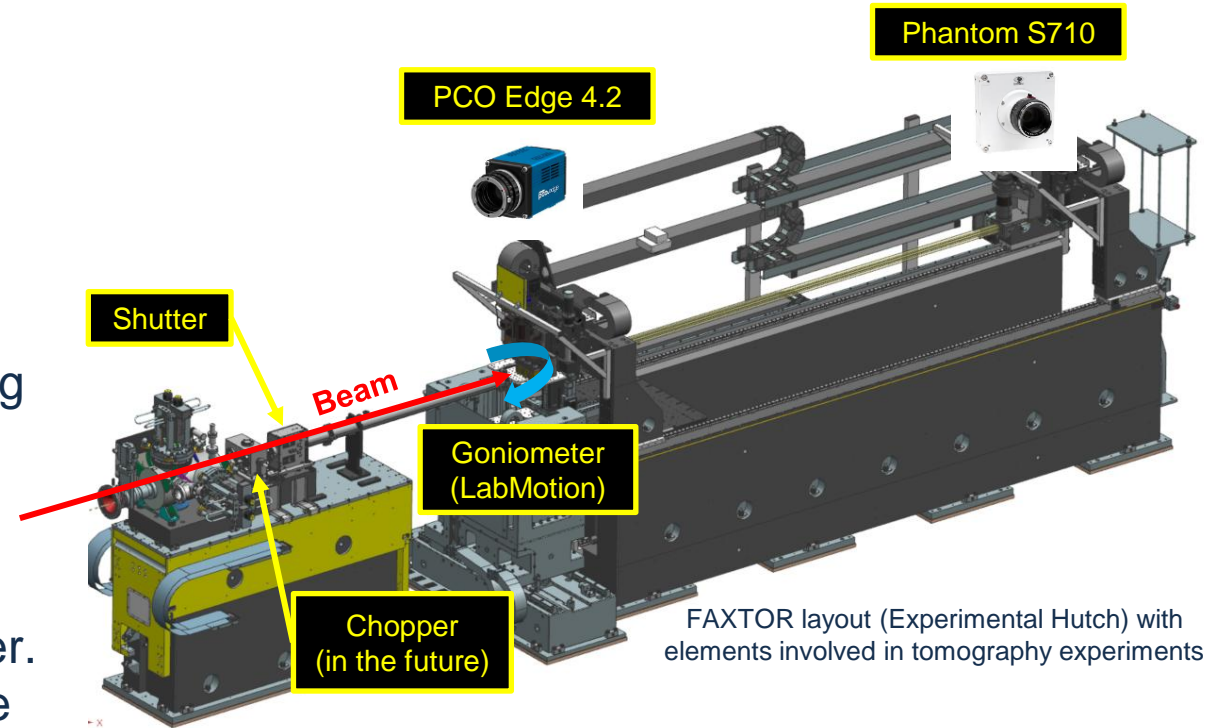
Magnet Control Unit for the DDKPPS

Beamlines Projects

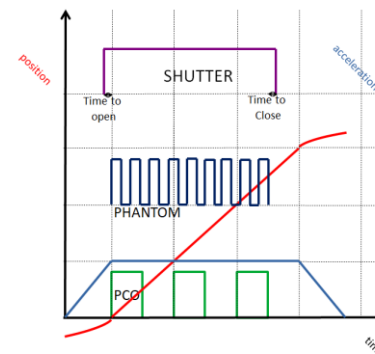


FAXTOR - fast tomography beamline:

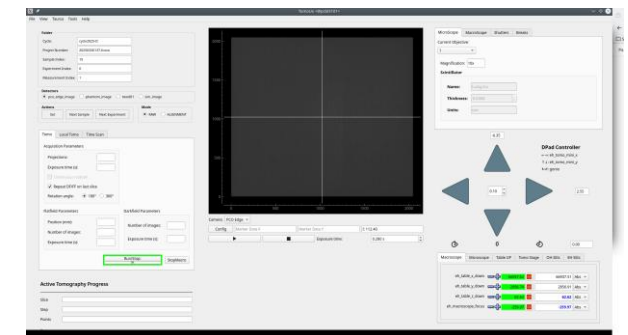
- Tomography experiments are implemented as continuous scans with Sardana and involve:
 - . Multiple detectors (4 in total, 2 used simultaneously)
 - . Shutter (allows 10 ms opening)
 - . Goniometer (max. rotation speed: 4860 deg/s - 15 tomos/s)
 - . Synchronization implemented using NI6602 (counting card)
- Phantom – high speed camera: (max. acq. rate: 7000 fps in 8bit mode (60 Gbit/s))
 - . Control is implemented with the REST server.
 - . Data transfer is implemented with the HDF5 file writer.
 - . Files written into the IBM Storage Scale clustered file system (GPFS) – data transfer rate up to 30-40 Gbit/s.
- The other detectors are integrated in Tango using LImA (ESRF): PCO Edge, PCO DiMax, C-BLUE.
- Beamline GUI implemented using Taurus and LaVue (DESY).



FAXTOR layout (Experimental Hutch) with elements involved in tomography experiments



Example of synchronization description used at FAXTOR

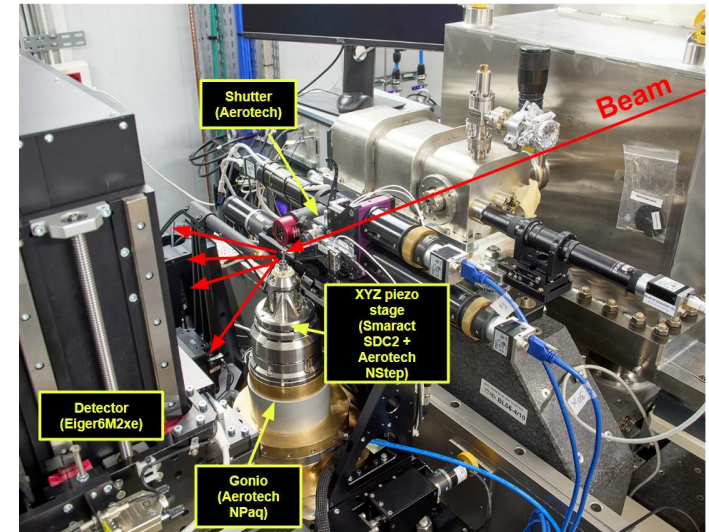


TomoUX (based on Taurus and LaVue)

XAIRA & XALOC

XAIRA – microfocus MX beamline in He atmosphere:

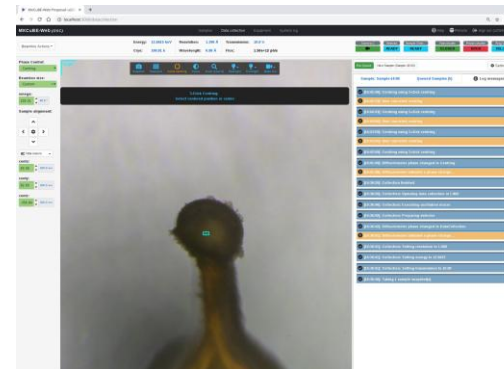
- MX experiments are implemented as continuous scans (oscillation: simple and helical and X-ray centering: raster scan) with Sardana and involve:
 - . Eiger2xe 6M detector
 - . Shutter
 - . Goniometer / XYZ piezo stage (Aerotech + Smaract)
 - . Synchronization implemented using Aerotech motion controller
- MXCuBE Web (ESRF *et al.*) – data acquisition: Mounts sample → Centers it → Uses online processing → Collects a full dataset
- iCAT DRAC (ESRF *et al.*) – work of the SDM and MIS sections: Gives access to raw and processed data.



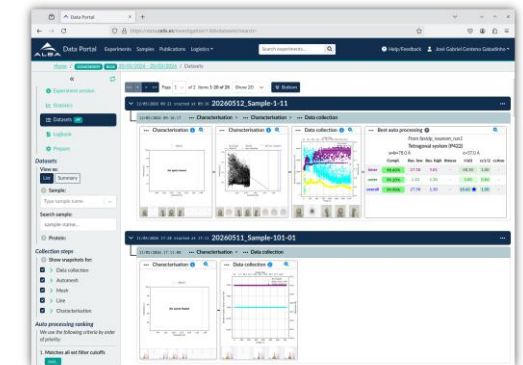
XAIRA End Station

XALOC (another MX beamline) refactoring:

- New diffractometer – HiDiffX – from IRELEC will be installed this summer. Currently working on:
 - . The Tango DS for the diffractometer control
 - . LImA plugin for its OAV camera
- Soon after we will migrate the MXCuBE Qt to MXCuBE Web and iCAT DRAC.



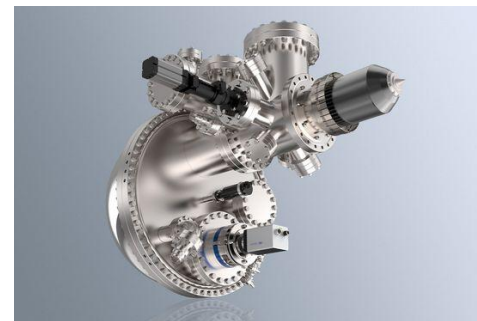
MXCuBE Web at XAIRA



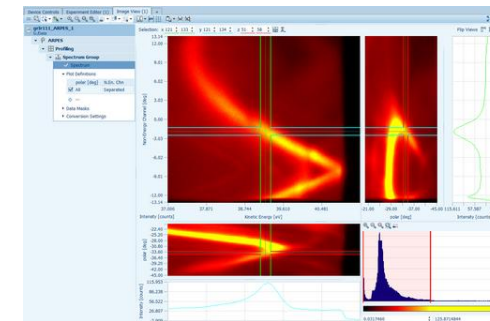
iCAT DRAC at XAIRA

3Sbar: surface diffraction and surface spectroscopy at high pressure beamline:

- In construction, beam in the OH in Nov'26.
- **Prodigy (SPECS):**
 - . Rich software application to control and perform XPS acquisitions. Allows remote connections.
 - . Integrated with an **async Tango DS (MAXIV)**, and extended with a **Sardana** controller.
- Detectors:
 - . **Pilatus4 2M (Si)** (Up to 4kHz and 80Gbit/s). – Dedicated server and fiber optics link. Still evaluating definitive software solution to integrate it in the controls system. Most probably **LimA** with hardware saving mode.
 - . **Lambda 750 k** (Up to 100 Hz) - Integrated with Tango via **LimA** (ESRF).
- 2x **PandABox** connected via SFP will allow for a fast acquisition of analog signals, synchronized with motion and the rest of hardware-triggered devices.
- Diagnostics cameras: extensive setup with **11 Basler Ace** cameras located along the optics and end station.
 - . Our plan is to use **LimA** and the already tested **LaVue** application for live view.



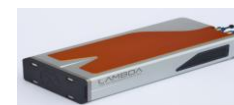
Hemispherical electron energy analyzer developed by SPECS Group



Prodigy software by SPECS Group



Pilatus4 2M (Si)



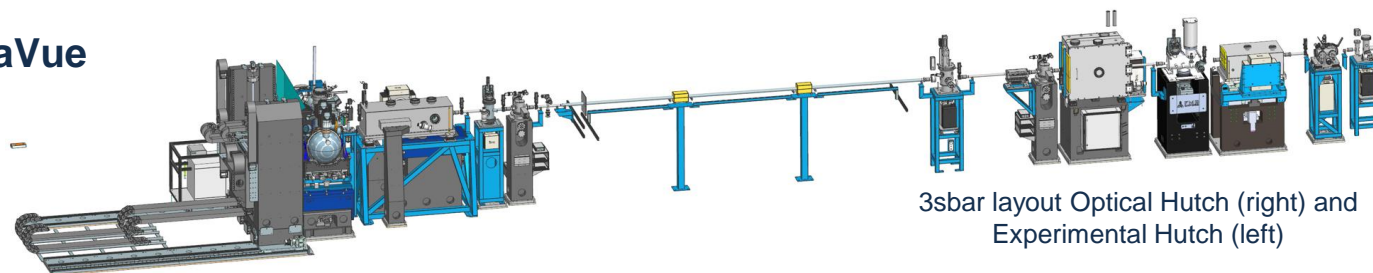
Lambda 750 k



2x (via SFP) PandABox



11x Basler Ace cameras



3sbar layout Optical Hatcher (right) and Experimental Hatcher (left)

Generic Projects



Control System Base Stack Dependency

- ALBA Control System was released in 2012 with **OpenSUSE 11**, **Python 2.6**, and **Tango 7**.
 - . No major OS upgrade until 2018, but:
 - . workstations → OpenSUSE 12 and 13
 - . some systems → Tango 8.
 - . The main difficulties were the **DAQ cards Linux drivers** and **Tango version upgrades**.
- Started gradual migration to **Debian 9** and **Tango 9** in 2018.
- Our Service Area diskless image was migrated to **Debian 10** in 2022.
- We are currently evaluating the migration to **Debian 12**:
 - . Adlink 2005 and NI6602, were successfully migrated to Debian 12.
 - . Currently new systems are being built with Debian 12.
 - . Our diskless image is ready for migration to Debian 12.
- Few remaining big projects are being migrated to **Python 3**: EPS Expert GUI, VACCA.



	OpenSUSE	Debian 9	Debian 10	Debian 12
Service Area	6	5	23	2
Control Room	1	0	13	0
Beamlines	14	56	55	19
TOTAL	21	61	91	21

Number of Linux Controls machines and their OS (w/o diskless cPCI which use Debian 10)

Conda Packaging and Salt Stack

- Evaluated new software distribution approaches to improve dependency management and maintainability.
- **Conda introduced in 2020**, initially using multiple channels (Anaconda, esrf-bcu, conda-forge) and locally built ALBA packages.
- Adoption grew rapidly, especially in Beamlines, leading to many **unmanaged environments**.
- In **2024**, a project was launched to define Conda policies, automate package creation through **CI pipelines**, and enable reproducible deployments with **Salt Stack**.
- Packaging workflow is based on **MAX IV** approach, using **Micromamba/Miniforge3**, **rattler-build**, conda-forge, esrf-bcu, and internal ALBA channels.
- Environment deployment is automated with Salt services, with centralized configuration and version management through Pillars.
- A stable controls software stack is maintained across services, ensuring consistent versions of Python, Tango, Taurus, Sardana, and Fandango.



Salt Service Monitor Updated: 2026-06-04T10:30:04Z • Hosts: 153

Groups: BL20, BL22, BL24, BL25, BL29, BL31

Hosts: cbl2901, cbl29wch02, cbl29cc02, cbl29e01, cbl29w01, lbl2901, pch2906, pch2907, tbl2902

Service filter: Filter services by name... Only not up to date

Export PNG CSV

Legend: Stable (green), Older (red), Fixed (blue), Newer (orange), Unknown (yellow)

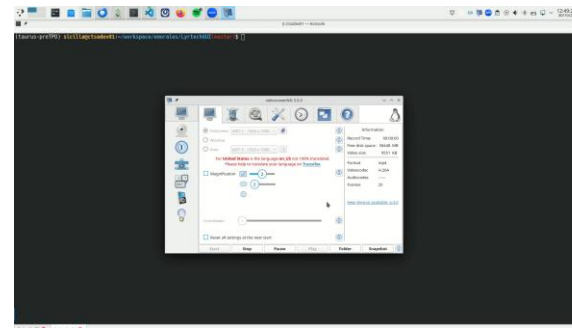
Service	ctb2901	ctb29wch02	ctb29cc02	ctb29e01	ctb29w01	lbl2901	pch2906	pch2907	tbl2902
albaem	v20251108								
brookscrypump	v20251215								
common_gui							v20260217	v20260217	
conda	v20251002		v20251002	v20251002	v20251002		v20251002	v20251002	
cryocon	v20251108								
eps		20250414			20250414		20250414	20250414	
fandango		20261216			20261216	20261216	20261216	20261216	
htsmagnetmonitor	v20251215								
icepapcms							v20250619		
icepapcms-db									Configured
kepcobop	v20251215								
limaccds_basler			v20260216						
limaccds_tucsendthiana				v20251107					
limaccds_xspress3				v20251111					
mysql			Debian pkg						Debian pkg
sardana							20250411	20250411	
sardana-conda					v20260409		v20260409	v20260409	
sardana-conda -- boresas					1.0.10		1.0.10	1.0.10	
sardana-conda -- pymca					5.9				
sardana-conda -- sardana-adlink					2.0.1				

Salt Monitor – a web tool to inspect the status of Salt services deployment in production machines.

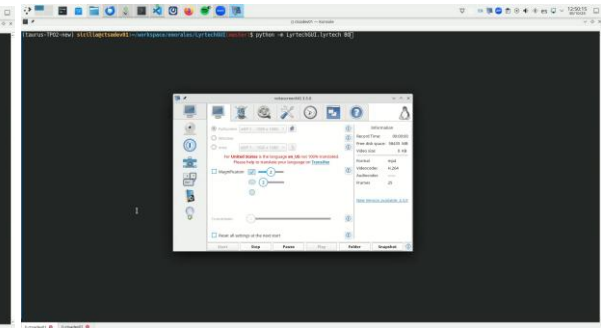
Taurus & SVG Synoptic

- Performance and stability issues affecting GUIs connecting to hundreds of attributes had driven the **Taurus Performance Optimization (TPO)** project. Results – Taurus applications are now more **stable** and up to **3x faster** at startup.
- To fully benefit from the improvements, Taurus requires Tango 10.1, then we started testing the Accelerators GUIs in Control Room with Conda.
- We developed several **SVG Synoptic (MAXIV)** GUIs, for gas handling systems and vacuum control.
- More about Taurus, not only at ALBA, in Oriol's talk tomorrow.

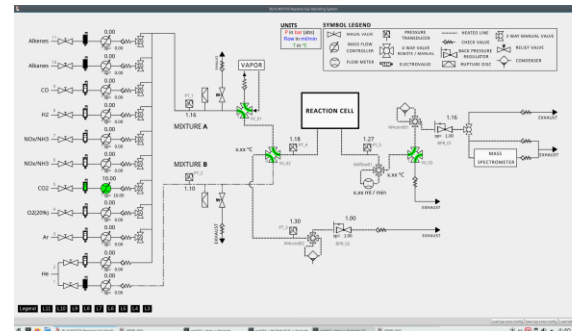
Before optimization: 20 s



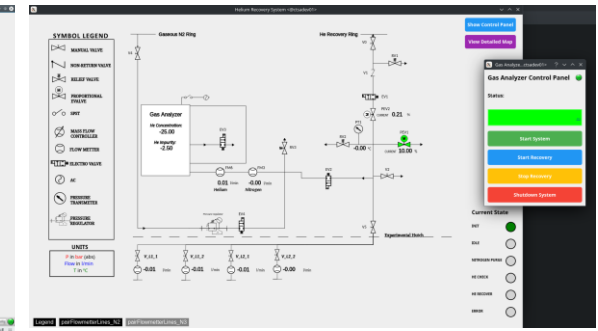
After optimization: 10 s



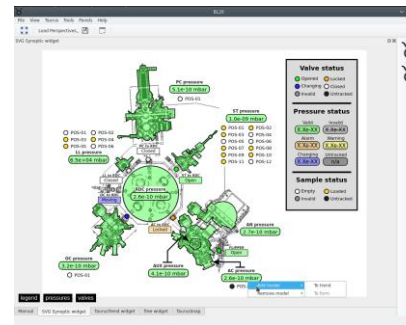
Example: Booster DLLRF (Lyrtech) connecting to approx. 500 attributes.



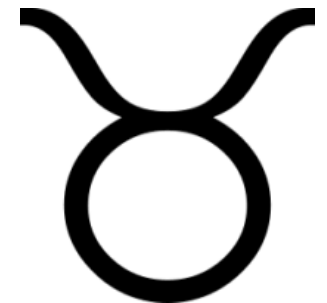
SVG Synoptic for BL16 gas handling system



SVG Synoptic for BL04 He recovery line

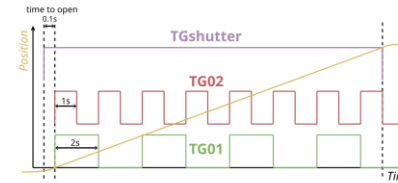


SVG Synoptic for BL20 end station vacuum



Sardana

- FAXTOR and XAIRA requirements had driven our recent Sardana developments – enhancing the **configuration of synchronization elements in continuous scans**.
- We are finishing migration (13/14 BLs done) of the Sardana deployment at ALBA from Debian to **conda**. Next step is use Sardana with Tango 10.1.
- This year our objectives are:
 - . Implement data publishing using **blissdata** (ESRF).
 - . Use in production at all beamlines the Sardana **configuration tool**.
 - . Introduce **synchronized motion** (trajectories) in the core following MAXIV advances from last years.
 - . Offer Sardana training to ALBA BL scientists.
- More about Sardana, not only at ALBA, in Oriol’s talk tomorrow.



Synchronization description

Synchronizer	Delay (s)	Active (s)	Total (s)	Nr Points
TG01	0.1	2	4	8
TG02	0.1	1	2	4
TGshutter	0	16.1	16.1	1

Measurement Group configuration

Channel	Synchronizer	Synchronization
twod01	TG01	gate
ct01	TG02	trigger
shutter	TGshutter	trigger

Multiple synchronization description

CONDA

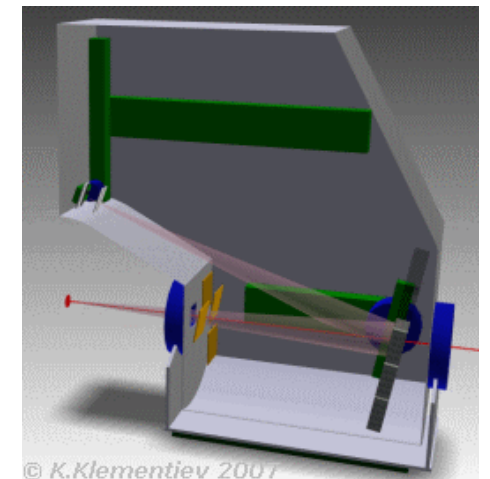


TANGO 10.1

```

1 tango_host: pt222.cells.es:10000
2
3 pools:
4   demo1:
5     instruments:
6       /slit:
7         class: NXcollimator
8       /mirror:
9         class: NXmirror
10      /monitor:
11        class: NXmonitor
12
13    controllers:
14      motctrl01:
15        type: Motor
16        python_module: DummyMotorController.py
17        python_class: DummyMotorController
18      elements:
19        mot01:
20          axis: 1
21          instrument: /slit
22        mot02:
23          axis: 2
24          instrument: /slit
25        mot03:
26          axis: 3
27          instrument: /mirror
28        mot04:
29          axis: 4
30          instrument: /mirror
    
```

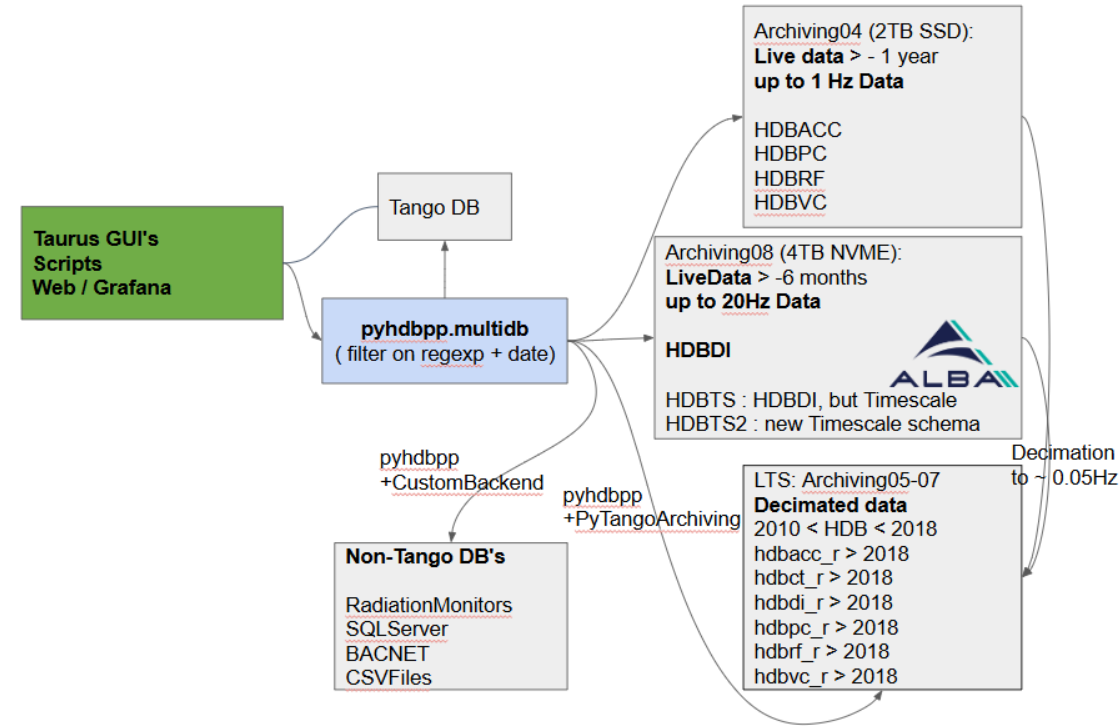
Example of the sardana configuration file (YAML).



Example of synchronized motion in CLEAR spectrometer (by K. Klementiev)

Tango Archiving

- We use **HDB++ with MariaDB** backend since 2018. All data are stored at change event rates (0.1 - 20Hz) (change events, archive events, and periodic archiver).
- [pyhdbpp](#) library provides transparent access to data for both GUI and CLI extraction, including the multidb extraction.
- **Accelerators archiving architecture:** multiple databases per subsystem, live data and decimated data. All on physical machines.
- **Beamlines archiving architecture:** one database per beamline on a VM.
- This year our objective is to define the **Archiving Architecture for ALBA II**: we are evaluating TimescaleDB and Grafana.
- More in Sergi's talk on Thursday (Archiving Workshop).



Tango archiving database architecture for Accelerators

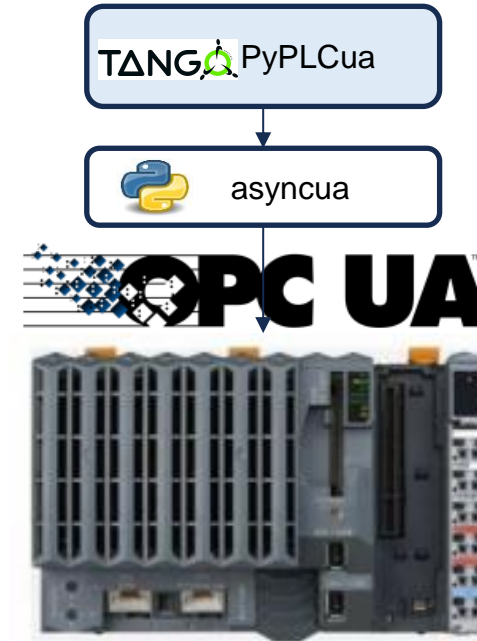
BEAMLINES	MACHINE	ctb04arch02	ctb04arch02	ctb06arch01	ctb09arch02	ctb11arch02	ctb13arch02	ctb16arch01
Total GIB	194.62	194.39	192.38	179.81	194.39	194.39	194.39	192.38
Used GIB	79.65 (+0.07)	92.85 (+0.12)	105.61 (+0.16)	71.36 (+0.09)	105.15 (+0.20)	86.38 (+0.50)	109.74 (+0.14)	109.74 (+0.14)
Health	98.13%	72.13%	59.89%	55.23%	79.41%	34.79%	82.41%	82.41%
% Used	40.89 (+0.04)	47.76 (+0.06)	54.90 (+0.09)	39.69 (+0.06)	54.09 (+0.10)	44.44 (+0.26)	57.04 (+0.07)	57.04 (+0.07)
ok	59	136	185	122	81	105	170	170
stall	78	172	296	126	243	134	364	364
on	146	427	571	269	180	687	620	620
off	6	0	243	180	2	0	28	28
lost	3	11	29	17	59	5	12	12
novals	5	26	3	148	0	0	0	3
Archivers:	0 / 15 devices offline	0 / 11 devices offline	0 / 11 devices offline	0 / 11 devices offline	0 / 11 devices offline	1 / 12 devices offline	0 / 8 devices offline	0 / 8 devices offline
All devices are online								
Top Tables								
Total GIB	202.22	194.39	194.39	194.39	263.29	194.39	194.39	194.39
Used GIB	139.48 (+0.14)	172.58 (+0.21)	112.06 (+0.15)	120.40 (+0.30)	99.85 (+0.17)	94.15 (+0.07)	94.15 (+0.07)	94.15 (+0.07)
Health	65.28%	69.64%	89.12%	64.17%	36.10%	49.95%	49.95%	49.95%
% Used	68.97 (+0.06)	88.70 (+0.11)	57.65 (+0.08)	61.94 (+0.16)	37.93 (+0.07)	48.44 (+0.04)	48.44 (+0.04)	48.44 (+0.04)
ok	265	188	190	86	271	10	10	162
stall	81	234	367	188	172	0	0	0
on	512	582	624	420	459	420	420	420
off	18	24	1	7	2	0	0	0
lost	11	15	30	19	3	2	2	101
novals	56	99	29	65	0	0	0	0
Archivers:	0 / 7 devices offline	0 / 12 devices offline	0 / 12 devices offline	1 / 7 devices offline	0 / 12 devices offline	0 / 12 devices offline	0 / 12 devices offline	0 / 12 devices offline
All devices are online								
Top Tables								

WIP: web dashboard for Tango Archiving health check (tab with Beamlines databases)

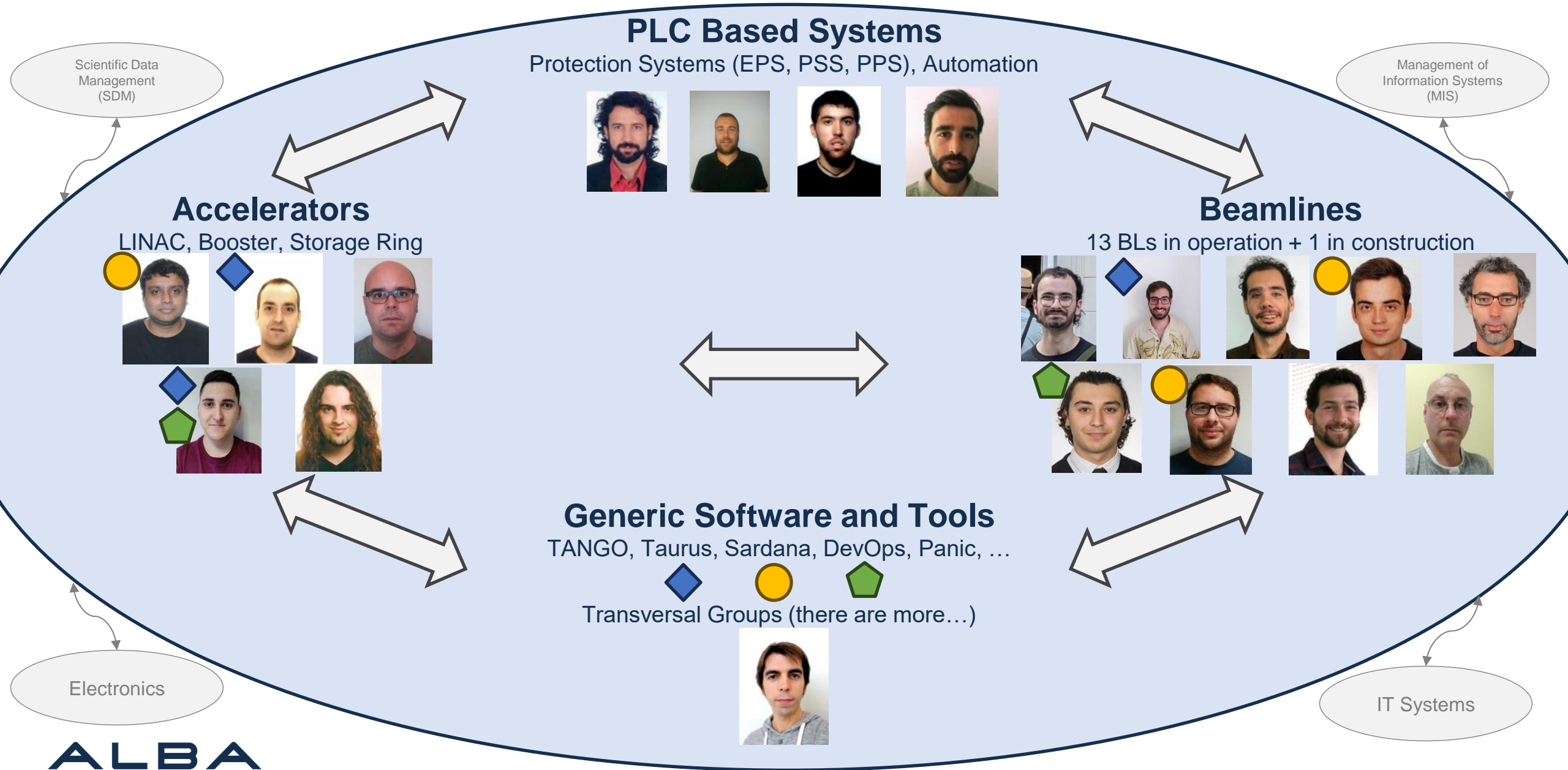
Software for PLC systems

PyPLCua – Tango DS for OPC UA:

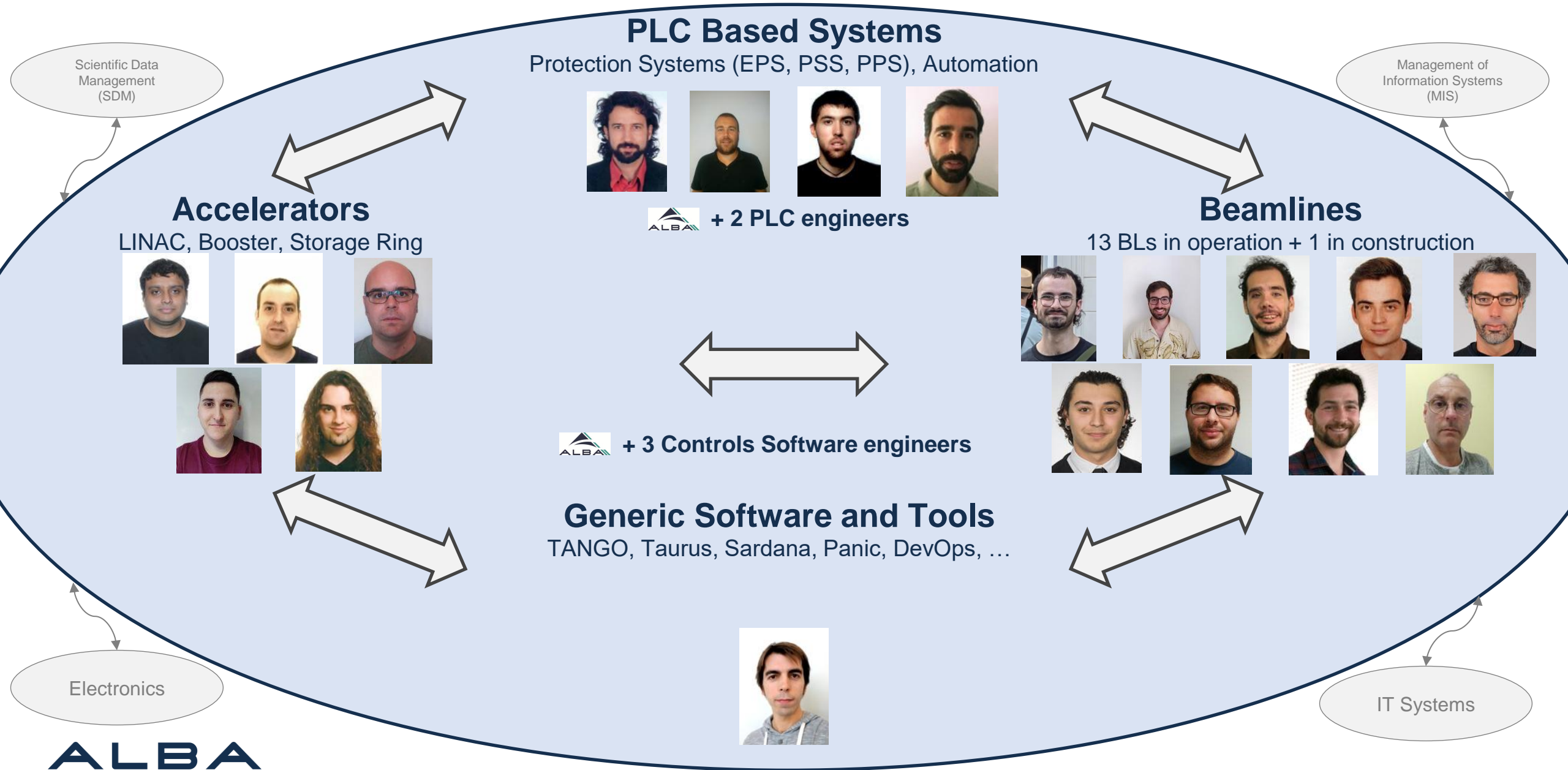
- Currently we use **PyPLC** based on Modbus TCP. PLC code and configuration is autogenerated from the cable database. The **configuration CSV file** is used to generate Tango DS dynamic attributes.
- As study of technologies for ALBA II we have developed the PyPLCua DS which uses:
 - . self-descriptive model (no need for configuration files)
 - . Client-Server Events (no need for polling)
- We have benchmarked both implementations and OPC UA shows better performance than Modbus.
- More about PyPLCua in Emilio's talk tomorrow.



ALBA Controls Team today



ALBA Controls Team *tomorrow*



Scientific Data Management (SDM)

Management of Information Systems (MIS)

Accelerators

PLC Based Systems

Protection Systems (EPS, PSS, PPS), Automation

LINAC, Booster, Storage Ring

ALBA + 2 PLC engineers

Beamlines

13 BLs in operation + 1 in construction

ALBA + 3 Controls Software engineers

Generic Software and Tools

TANGO, Taurus, Sardana, Panic, DevOps, ...

Electronics

IT Systems



Summary & Acknowledgement



Multiple **Accelerators** (new DLLRF, 3rd HC, FIM2, DDK), **Beamlines** (new BLs: FAXTOR, XAIRA, 3Sbar; XALOC), **Generic** (DevOps, Sardana, Taurus, Archiving, PLC Software) were presented.



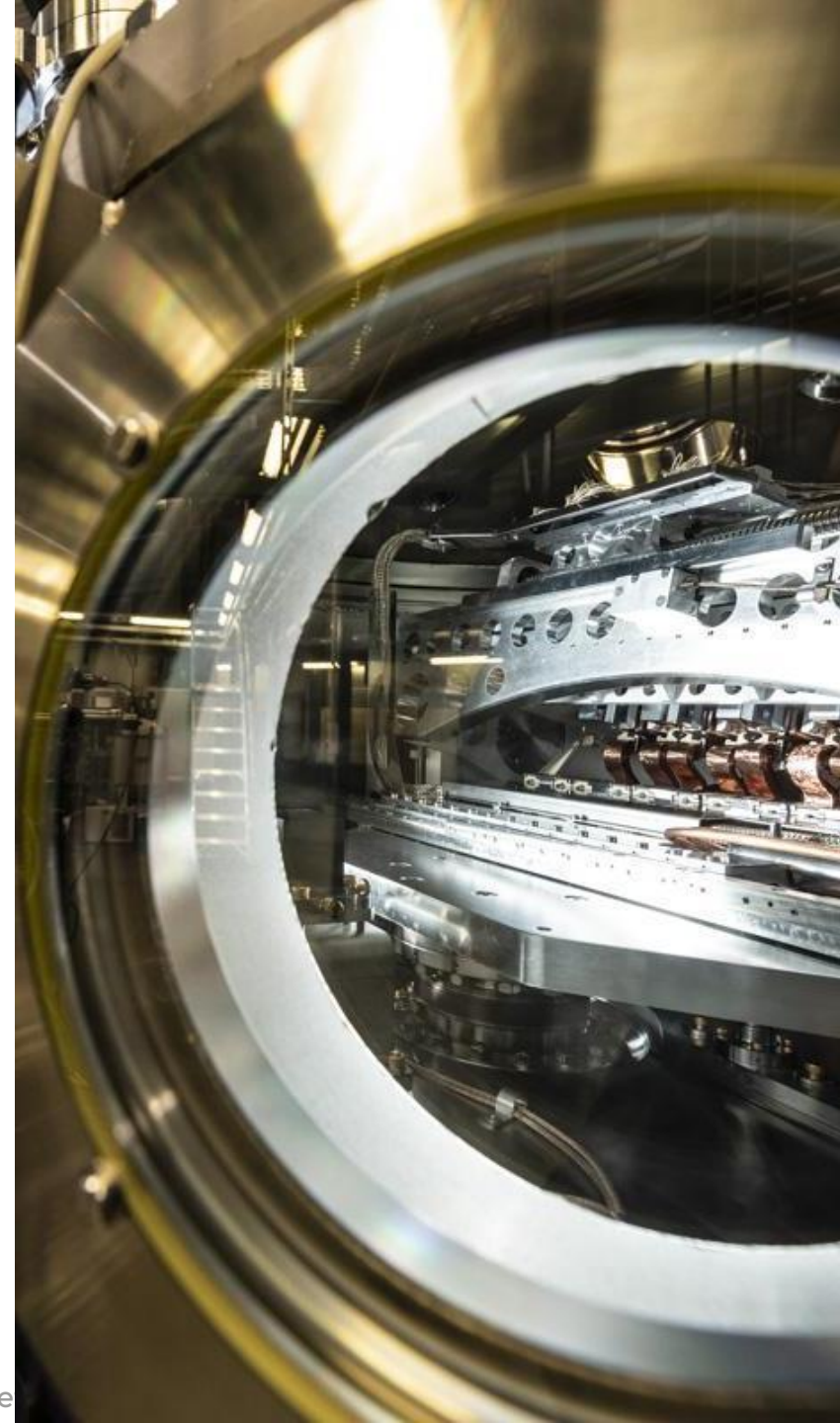
TANGO Community for developing and sharing the TANGO and its ecosystem projects.



Controls Section and **Computing Division** for your hard work.



ALBA Customer Units for the excellent collaboration.





www.cells.es

Cerdanyola del Vallès (Barcelona) Spain

Tel: (+34) 93 592 43 00

ALBA II Core Data Services Project

- ALBA Controls is involved in all ALBA II Controls Programs: Experiments, Accelerators, Computing & Data, Infrastructure* and Implementation. * We will start to be involved soon in the Infrastructure program.
- The Core Data Services Project, within the Computing & Data program, is focused on defining the ALBA II Control System and Electronics architecture and developing generic solutions and tools to serve other programs. It is composed, so far, from 6 Work Packages and 19 projects:

Low Level Architecture	IOC Architecture	PLC Architecture	Virtualization and controls software packaging strategy	Detectors Integration Strategy
Instrumentation, Motion, and Synchronization	Beamlines Instrumentation and DAQ	Experiment Synchronization Platform	Motion Control Platform	Fast/Real Time Control Loops
Tango, Archiving, and Alarms	Tango Enhancements	Tango Archiving Architecture	Alarms Enhancements	
Experiment Orchestration	Sardana Enhancements			
User Interfaces	Taurus Enhancements	User Interfaces	Vacca (Vacuum Control) enhancements	
Sustainability and Quality	Control System configuration and management	Controls software quality	Controls software configuration	CS observability

Core Data Services Project (gray – work packages, blue – projects in progress, yellow – projects in definition).

Kanban Work Organization (Software Teams)

Too many hats



KISS: Keep It Super Simple

- We use two dimensional classification of activities: category & size.
- Engineers act as either Service Support or Development on a given day.

Local resource management



Priorities are set by those who benefit from the work

- Projects are prioritized by Management (ALBA Activity Plan).
- ~30 backlogs are directly prioritized by Customer Units.

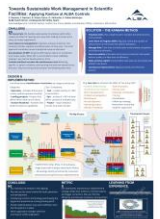
Accumulation of Work in Progress



Stop starting, start finishing

- The board was initially cleaned up to remove outdated tasks.
- We use distributed and asynchronous pull board replenishments.

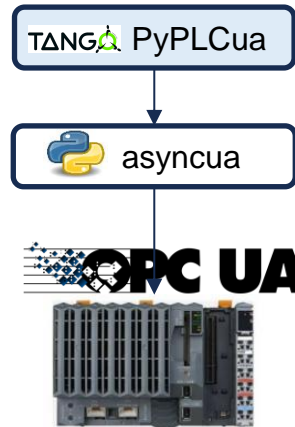
More details can be found in “Towards sustainable work management in scientific facilities: applying Kanban at ALBA Controls” - [TUMR007](#). I can share the poster with you with better graphics.



Software for PLC systems

PyPLCua – Tango DS for OPC UA:

- Currently we use **PyPLC** based on Modbus TCP. PLC code and configuration is autogenerated from the cable database. The **configuration CSV file** is used to generate Tango DS dynamic attributes.
- As study of technologies for ALBA II we have developed the PyPLCua DS which uses:
 - . self-descriptive model (no need for configuration files)
 - . Client-Server Events (no need for polling)
- We have benchmarked both implementations and OPC UA shows better performance than Modbus.
- More about PyPLCua in Emilio’s talk tomorrow.



We are deploying **EPS User GUI (Taurus)**, a user oriented tool which provides an easy to interpret visual representation of the EPS logic.

Example: *Beamline not ready to open FE, what to do?*

EPS Expert GUI

vs.

EPS User GUI

Panel with the PLC logic (EPS User GUI)

→

Panel with signals corresponding to a given group (EPS User GUI)