SOLEIL UI experience, … and future plans

P.MADELA, G.ABEILLE, R.GIRARDOT, YM.ABIVEN

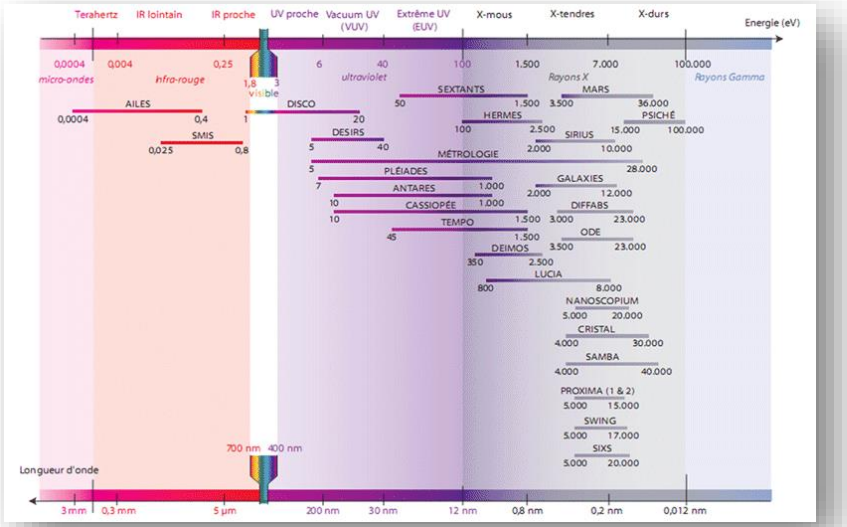
Outline

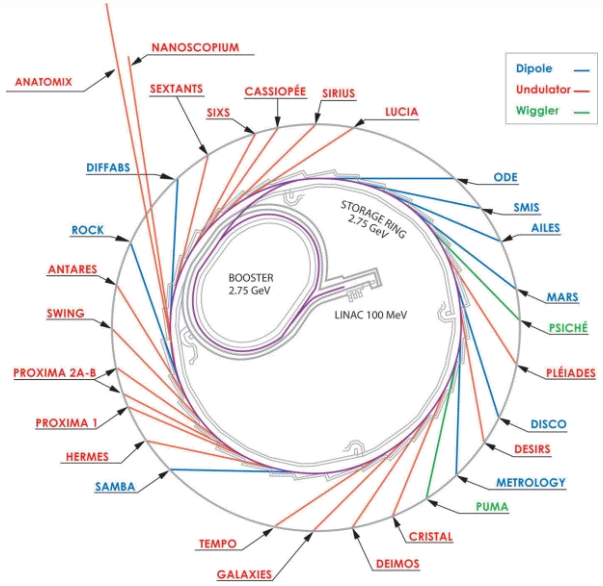
- Towards SOLEIL II
- UI status around control system
- UI organization
- Future plans

Towards SOLEIL II

SOLEIL in a nutshell

- Storage ring 354m, 2.75GeV
- 29 beamlines
- From far IR to hard X-rays
- Open to external users in 2008
- ~ 450 staff members





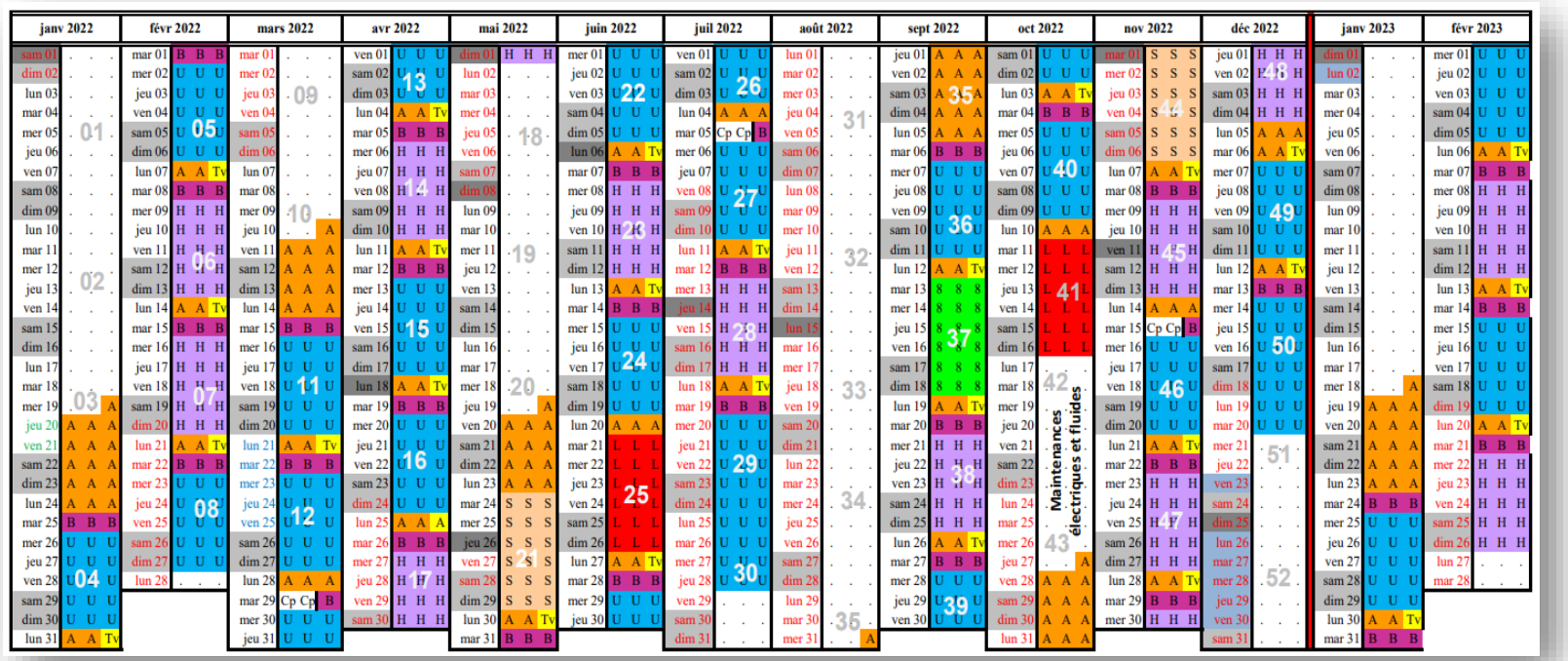
in 2022
~ 2746
single users
(~150 remote access)

More than
12 000
users since 2008

Operation

- Beam delivery **24 hours**, **7 days** a week
- **5019 h** (x 29!) ~ 145000 hours of beamtime in 2022
- Beam availability **98,95%** in 2022
- Mean time between failures (MTBF) **139 hours** in 2022

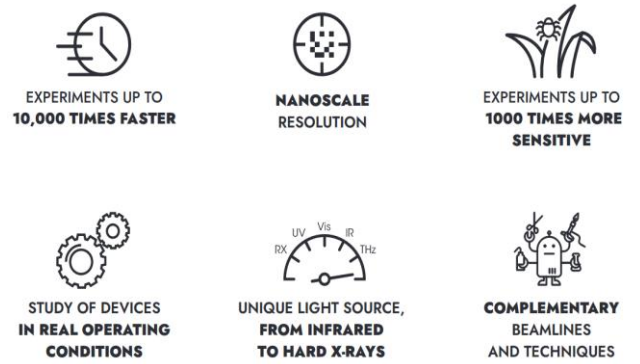
U	Uniforme Top-Up - 500 mA
H	Hybride Top-Up - 450 mA
8	8 paquets Top-Up - 100 mA
S	1 paquet Top-Up - 16 mA
L	Low-Alpha Top-Up
B	Beamlines
Cp	Contrôles RP périodiques, 3 mardis de 7h à 23h
Tv	Tests RP de validation, faisceau Lignes redonné à 10h
A	Temps Accélérateurs
.	Arrêt Machine



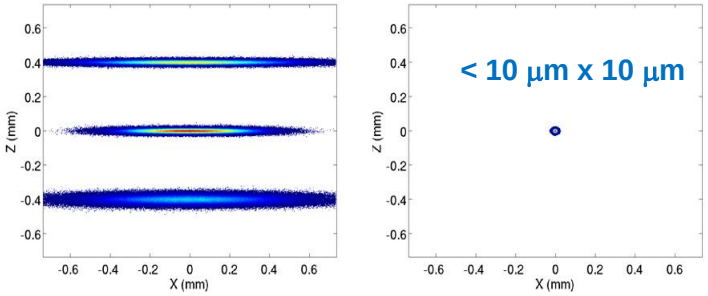
SOLEIL Upgrade

- Better performances for Accelerator and photon sources
  - Reaching an emittance **< 100 pm.rad.**
  - Keeping the same electron beam energy : **2.75 GeV**
  - Preserving a maximum current of **500 mA** in the multibunch mode.

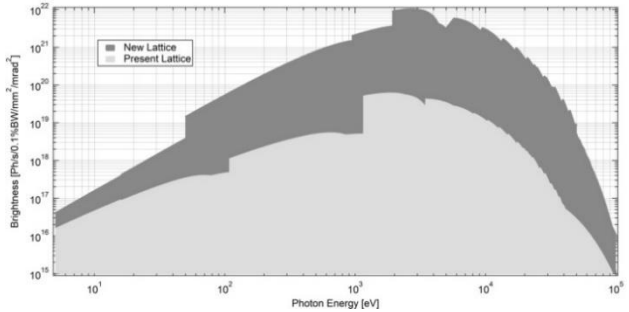
- New access mode with **more efficient use** of the SOLEIL Beamline



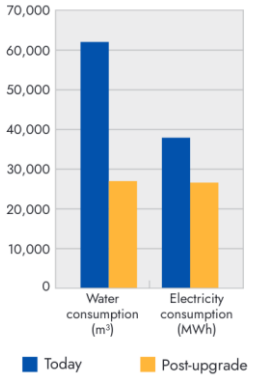
Beam SIZES



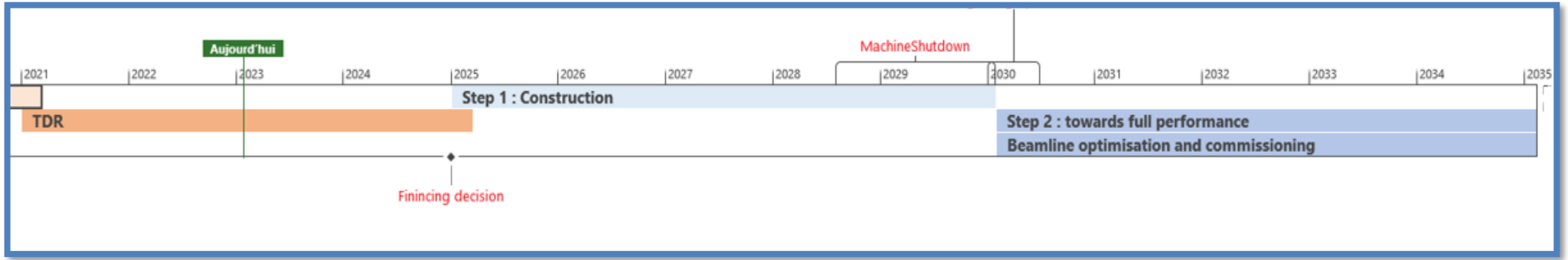
Brightness



- Green infrastructure
  - Reduction in the facility environmental footprint
  - Lower power and water consumption
  - Reduce operational cost



- Upgrade Timeline (1 year delay from initial planning)



Operational organization

Acquisition and Control System Engineering (ISAC) team,

in charge of

- Accelerators and beamlines control systems
- ~12000 Electronics devices (motion, cPCI, PLC, Robotics, …
- ~24000 Tango devices and ~12000 device servers
- Involved in daily operation with 3 on-call duty 24/7:
  - Software
  - DaQ and motion control
  - PLC, Robotic

Involve in TDR around 50 WBS for

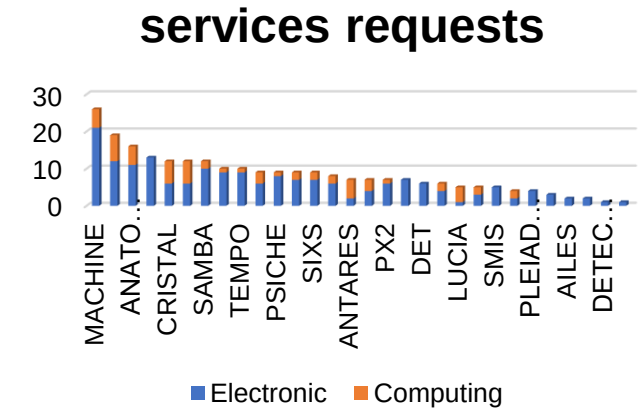
- Accelerators program
- Infrastructure Program
- BeamLine and Laboratory (BL2) program
- IT and Data management (transverse to support the others program)

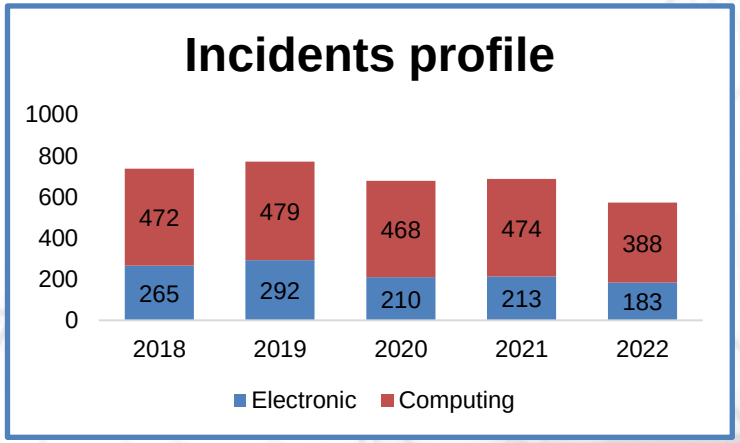
Works with other computing teams for SOLEIL IT systems integration

- ISI: Infrastructure & networks
- ISG: Management Information System
- GRADES: Data analysis

Software in operation

- **459** c++ Devices
- **32** Java Devices
- **44** GUI / API
- **300** Embeded codes for motion, DAQ, Robotic, PLC





UI status around control system

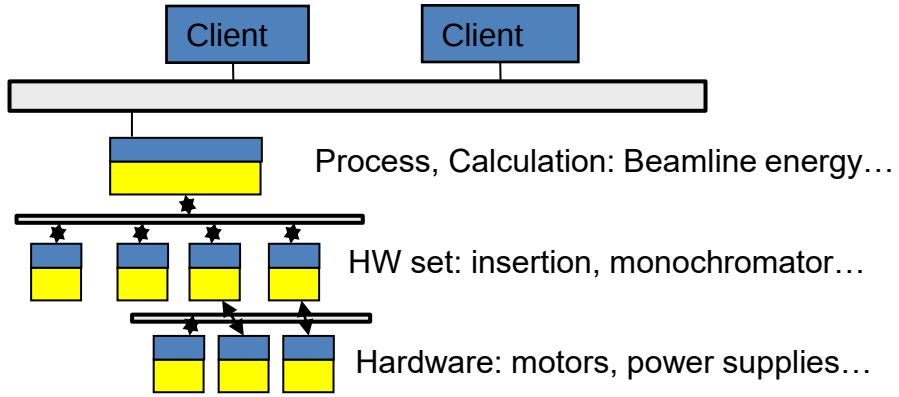
What is your control system made of?

Tango has become a SOLEIL standard de-facto

- Tango is used on all control systems (Accelerators, Beamlines and Labs)

Strategy to integrate everything in Tango devices *(ancestor of SOA/µService architecture?)*:

- Hardware (motors, vacuum, power supplies, …)
- Hardware sets (insertions devices, monochromators,…)
- External systems (Building Management in OPC, LINAC in LabVIEW….)
- Calculations, orchestration, workflows (beamline energy, beamline acquisitions processes, experimental data management, archiving…)



Outcomes:

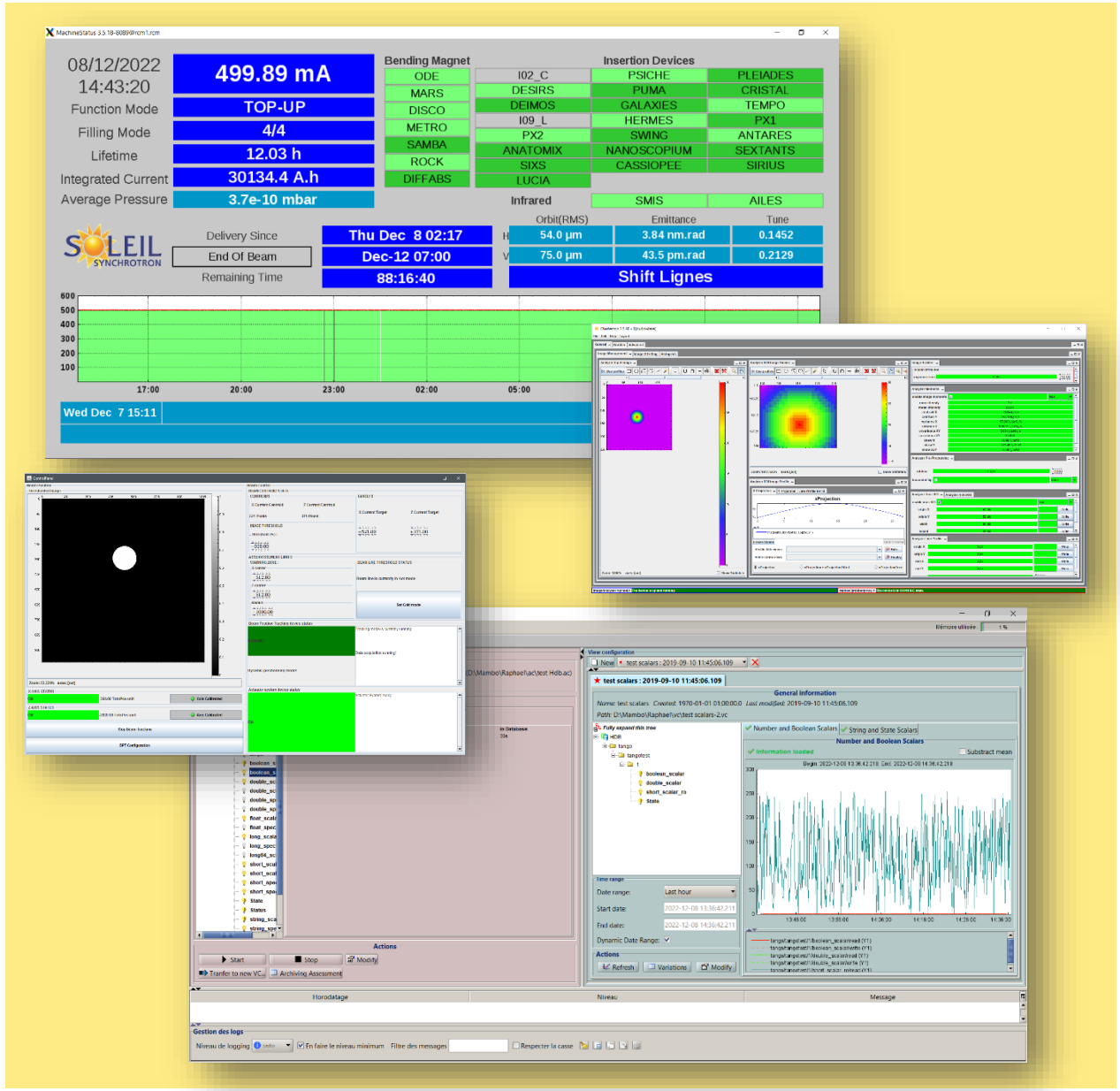
- A seamless integration in all Tango clients (GUIs), in archiving…
- Built-in data correlation
- Autonomy to our users with Tango client API (Python, Matlab, Labview, Igor Pro)

Comete Framework

For its common GUIs, SOLEIL use Java SWING language and has developed the Comete framework.

Library of widgets and data sources, with intermediates that manage the connection in between.

This framework enables to quickly develop new GUI.



SCADA Coox

No-code tool to create GUIs with drag & drop.

Developed in Java 8 by ORDINAL company

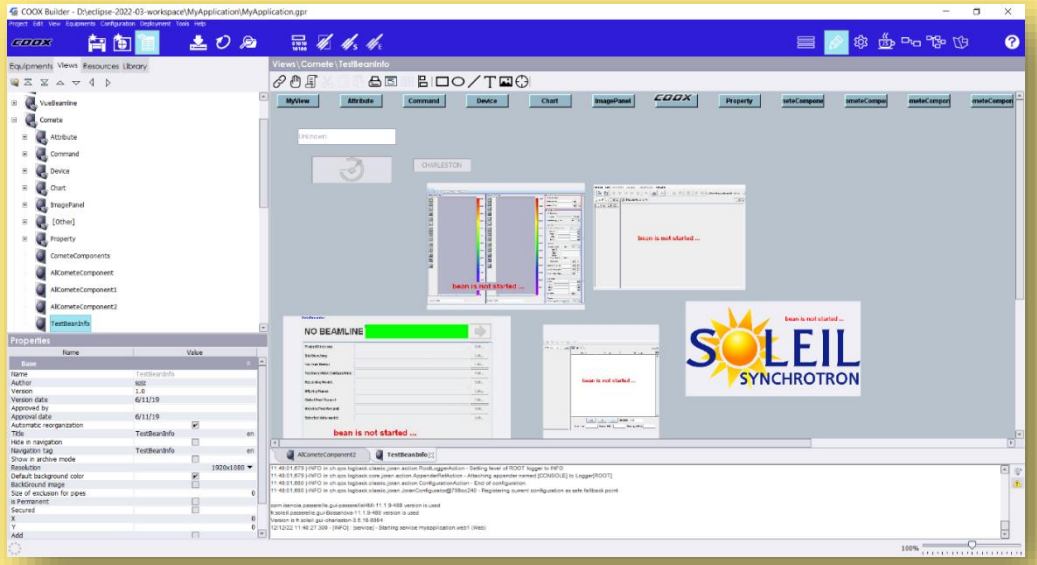
Based on Comete wigdets

Used by almost all beamlines which are autonomous in creating their views

Builder

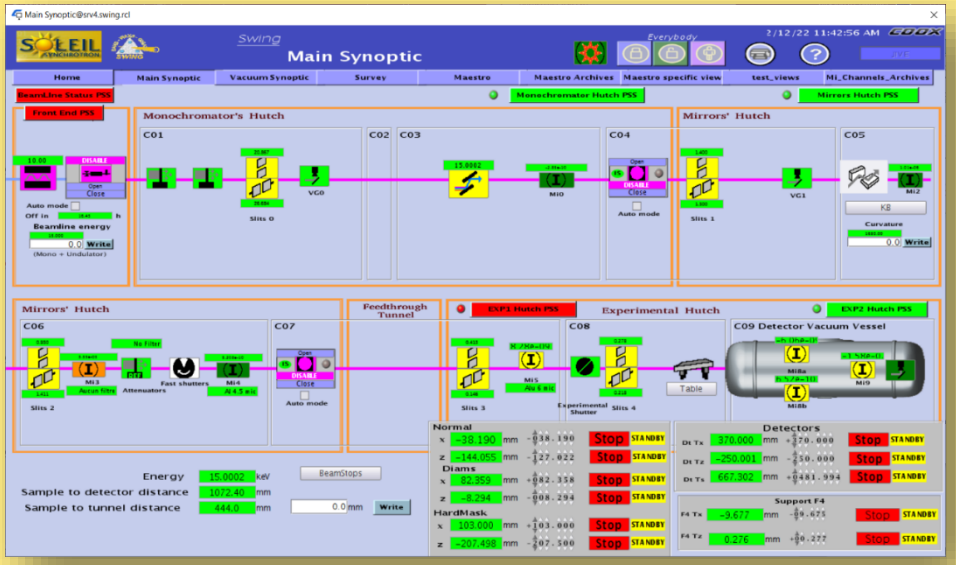
Creation of views by drag'n'drop

(uses SOLEIL's components with Comete)



Viewer

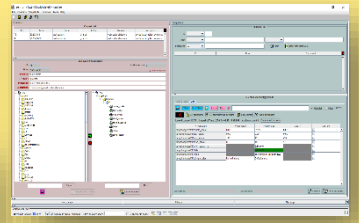
Runs the views previously created with Builder

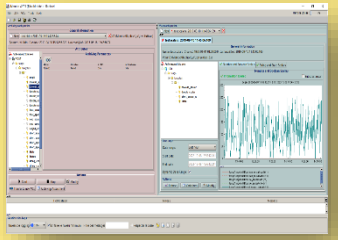


Common GUIs

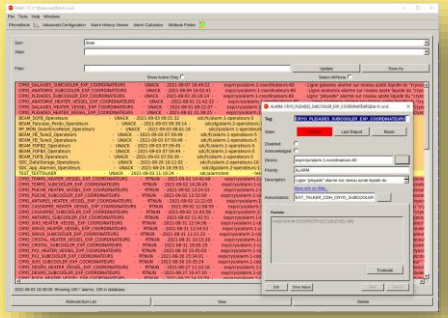
Tango Archiving

Configure & plot
HDB & TDB & Snap



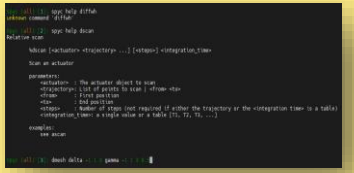


Alarms - PANIC

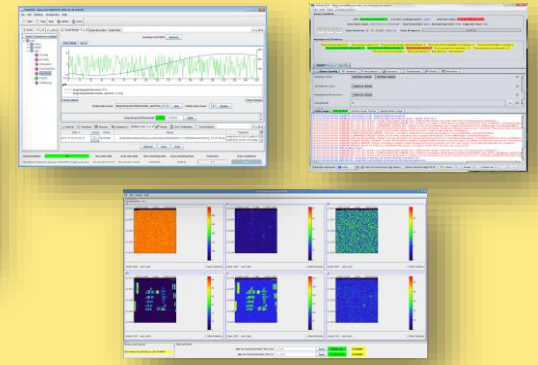


Acquisition

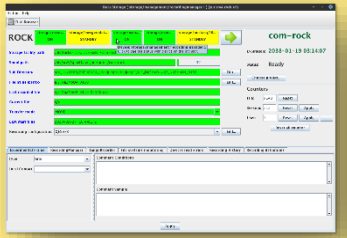
Spyc: SOLEIL Python CLI



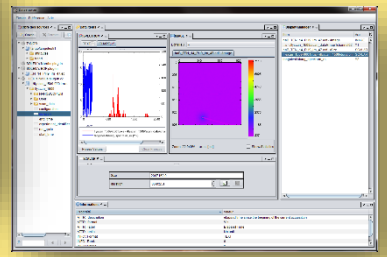
Fronts for scan devices
(FlyScan & ScanServer)



Experimental data recording

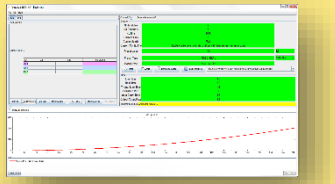


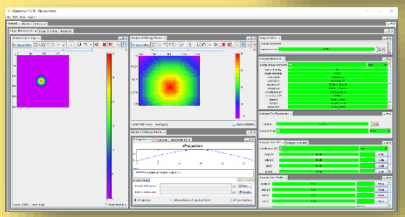
Configure & plot



Detectors

Multi Channel Analyzer

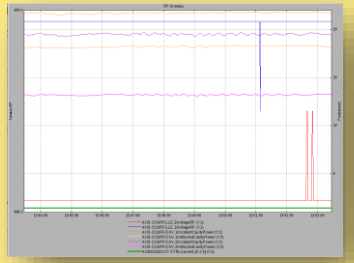


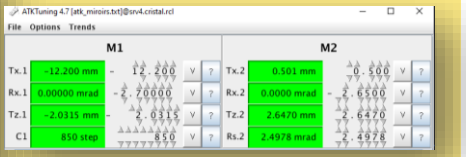


Beam image calculations

Tango tools

Large usage of ATKPanel, ATKTuning, Jive, Astor, ATKTrend

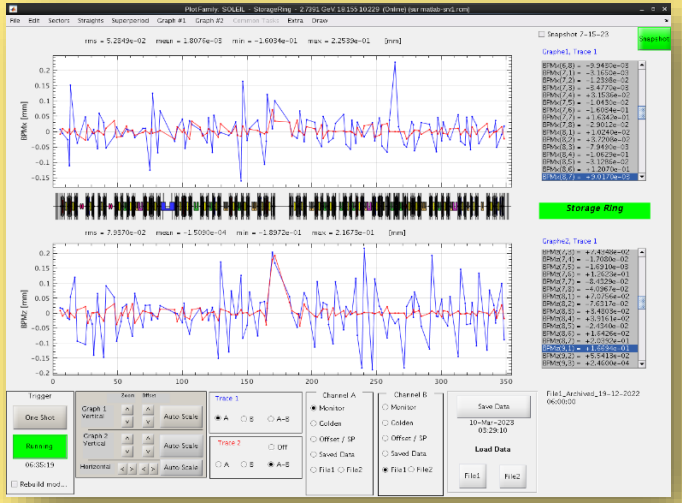




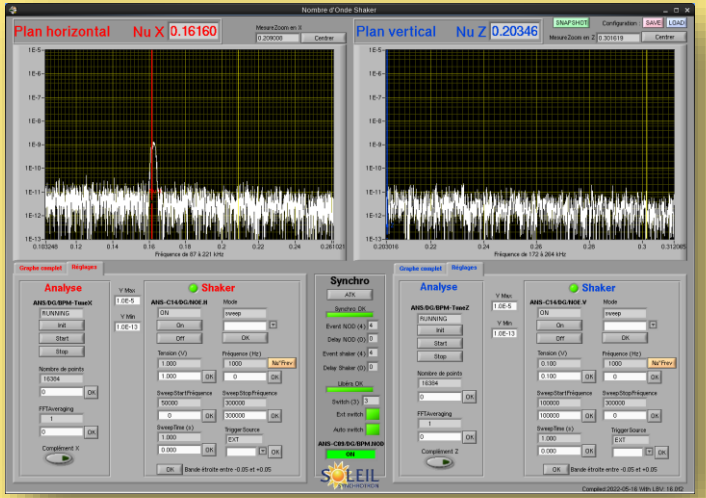
All the Java GUI based on Comete are also beans that can be integrated into COOX screens

Accelerators GUIs

Beam positions - Matlab

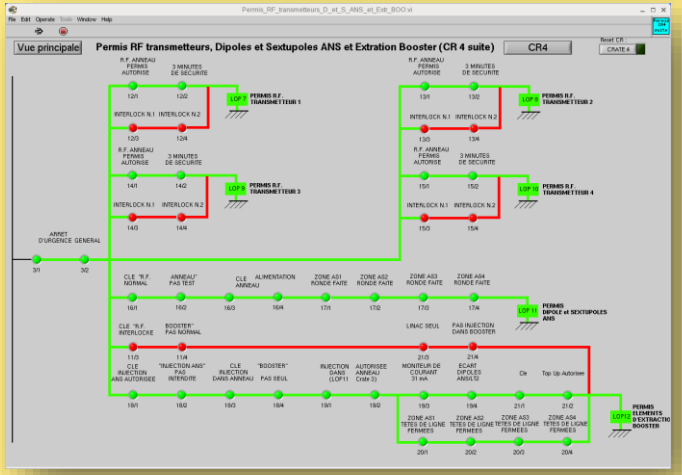


Beam Nu/WaveNumber measure - Labview

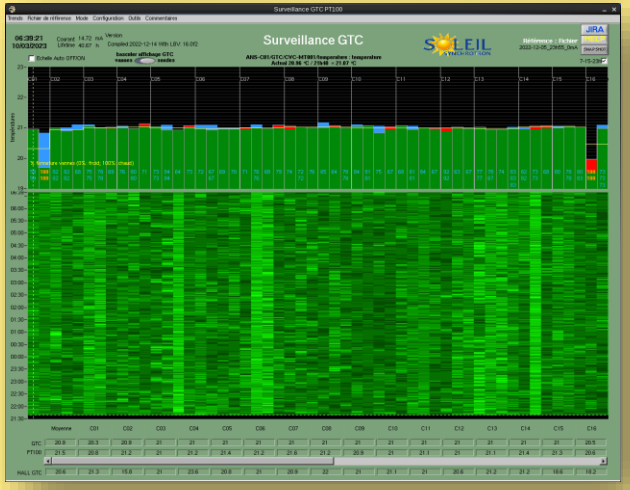


Postmortem analysis - Labview

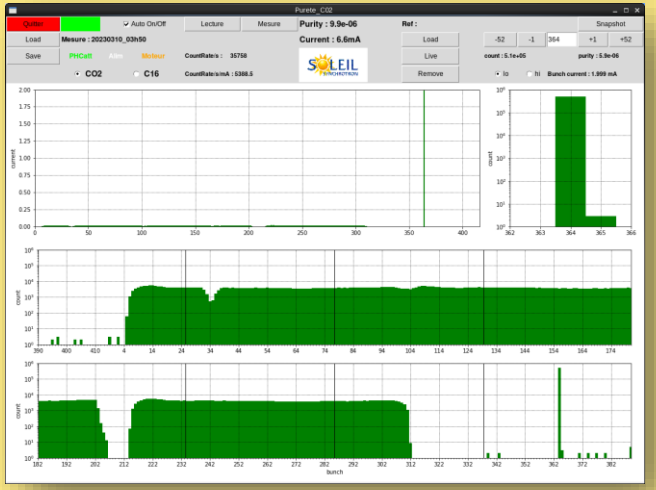
PSS permits (Personal Safety System)- Labview



Utilities monitoring (AC & Fluids) - Labview

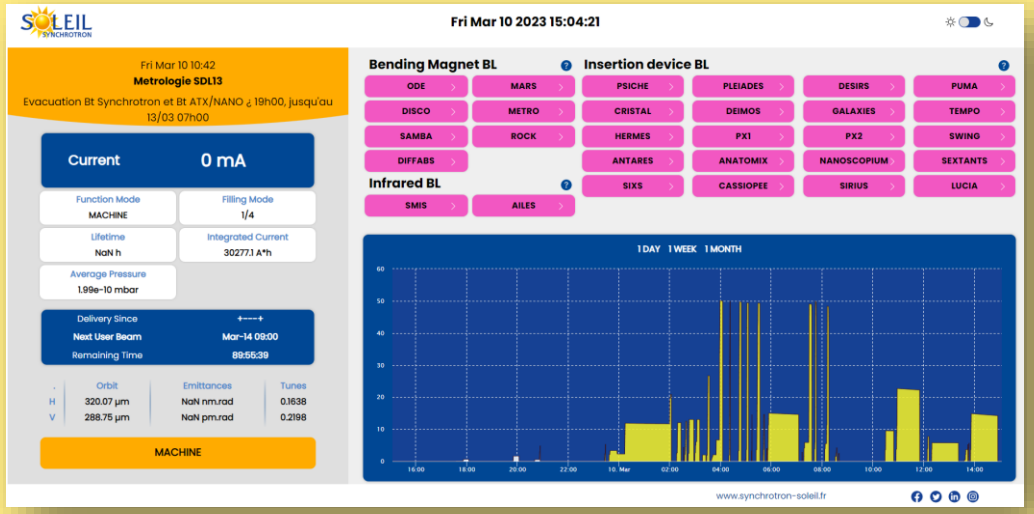


Beam purity measure - Python



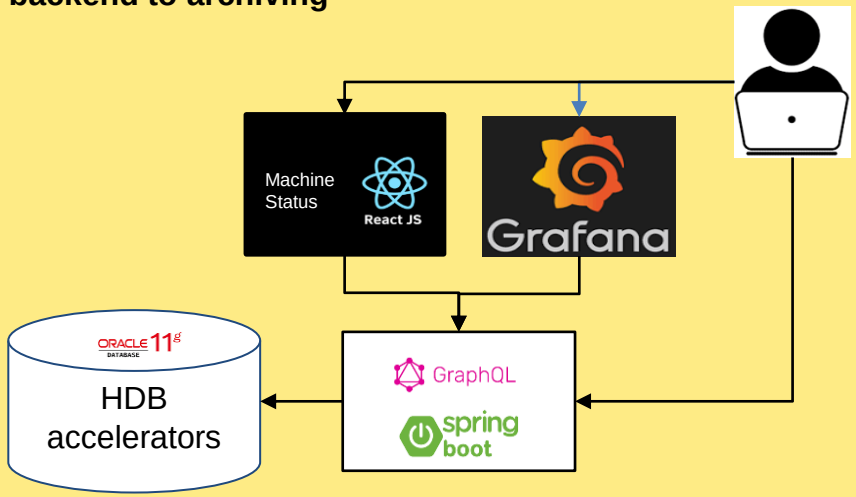
Apps for web evaluation

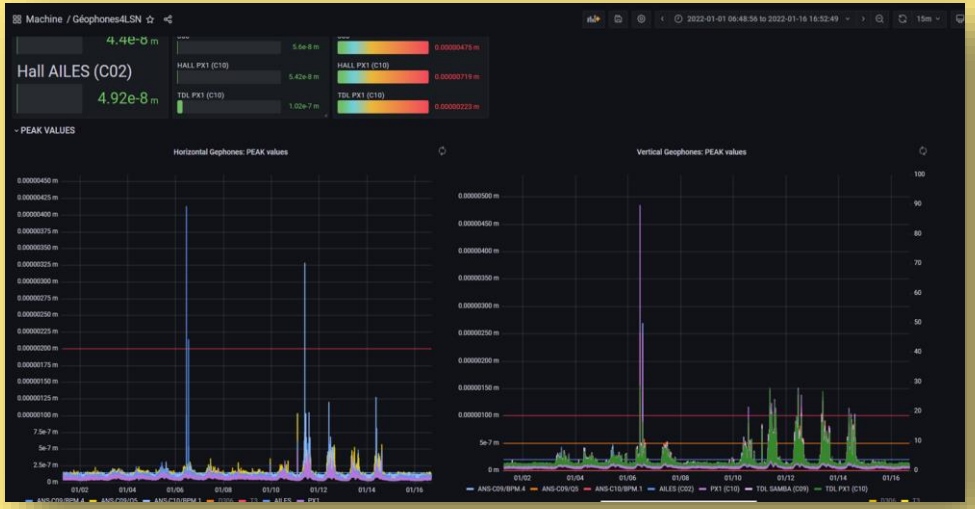
Machine status



Grafana

GraphQL backend to archiving





UI Organization

Accelerators practices

Accelerators as owner of

Labview apps
Python apps
Matlab apps

Accelerators as user of

Tango GUI (Jive, ATK…)
PANIC
Databrowser
Mambo / Bensikin
MachineStatus
Grafana (under evaluation)

152 LabVIEW apps
56 Python apps
114 Matlab apps
60 Grafana dashboards

Accelerators staff develop their own scripts/devices/GUI

- LabView, maintained by operators, acc. groups
- Matlab, maintained by physicists
- Python (TkInter, scripts, Tango devices), maintained by operators, physicists and acc. groups

GUI apps that embed the process are critical

- Top-up Injection process (Labview)
- Some feedbacks (Slow orbit feedback, …) (Matlab) …

Lots of accelerators' GUI are only for experts, and not for operations

Beamlines practices

Beamlines as owner of

Python apps
Igor Pro apps
Labview apps

Beamlines as user of

Tango GUI (Jive, ATK…)
PANIC
COOX
Salsa / FlyScan GUIs
Datastorage/Databrowser
Mambo/Bensikin
MachineStatus

Beamline staff develops their own scripts/devices/GUI

- Python (TkInter, scripts, Tango devices)
- On a few beamlines and laboratories
  - Igor Pro,
  - LabVIEW
  - Matlab

Beamline staff creates and maintains views of COOX synoptics

Computing teams practices

Computing teams as owner of

Tango generic GUI (Jive…)
Comete framework
GUI based on Comete
PANIC
COOX
Spyc
Tango Bindings
Grafana (under evaluation)

Support on deployment of products (Matlab,Labview,Python…)

Tango

Nexus

Computing teams provide common GUI solutions (Comete, COOX, Tango binding)

- Development, delivery, deployment, training and support

Computing teams provide a very limited support for Matlab, Labview, Python and Igor Pro

- Provides help for deployment of these products, their licenses and their tango bindings

Resources for GUI (not fulltime)

- ISAC :
  - 1 expert for Java Swing GUI development
  - 1 expert for back-end in Java
  - 1 expert for front in React.JS
- ISI:
  - 1 sys. admin for products and licenses installation, OS upgrades

Who is responsible for the development and the maintenance of the GUIs?

Accelerators

Accelerators as owner of

Labview apps
Python apps
Matlab apps

Accelerators as user of

Tango GUI (Jive, ATK…)
PANIC
Databrowser
Mambo / Bensikin
MachineStatus
Grafana (under evaluation)

Computing teams

Computing teams as owner of

Tango generic GUI (Jive…)
Comete framework
GUI based on Comete
PANIC
COOX
Spyc
Tango Bindings
Grafana (under evaluation)

Support on deployment of products (Matlab,Labview,Python…)

Tango

Nexus

Beamlines

Beamlines as owner of

Python apps
Igor Pro apps
Labview apps

Beamlines as user of

Tango GUI (Jive, ATK…)
PANIC
COOX
Salsa / FlyScan GUIs
Datastorage/Databrowser
Mambo/Bensikin
MachineStatus

Feedbacks

Feedbacks on user autonomy

- Positive feedbacks from users of COOX and apps w/ Tango bindings
- First feedbacks w/ Grafana are also positive

Sharing concerns between computing teams and business teams difficult to balance :

- Computing teams: focus on stability and reliability
- Business teams: focus on flexibly and changes

- Long term maintainability to be build :
  - *Migration from Python 2 to Python 3 is in progress with a support of ISAC on Beamlines and Accelerators*

- At some point, IT expertise is mandatory to operate GUI applications as they interact with a lot of IT components (Tango devices, the OS, network….)

Future plans

Future strategy

Transversal TDR program has a project dedicated to UI just started

- Strategy definition is under construction in the IT & Data management program
- Accelerators, beamlines and computing teams working together on the project

UI requirements

- Shared development between business & computing teams
- Well-known technologies are important -> Possibility to sub-contract
- Existing framework with a strong community to be able to share experience /dev
- Control layer between Tango device servers & GUI:
  - Add security/authentication: to allow only approved clients
  - Add requests tracking: to track clients requests and abnormal usage

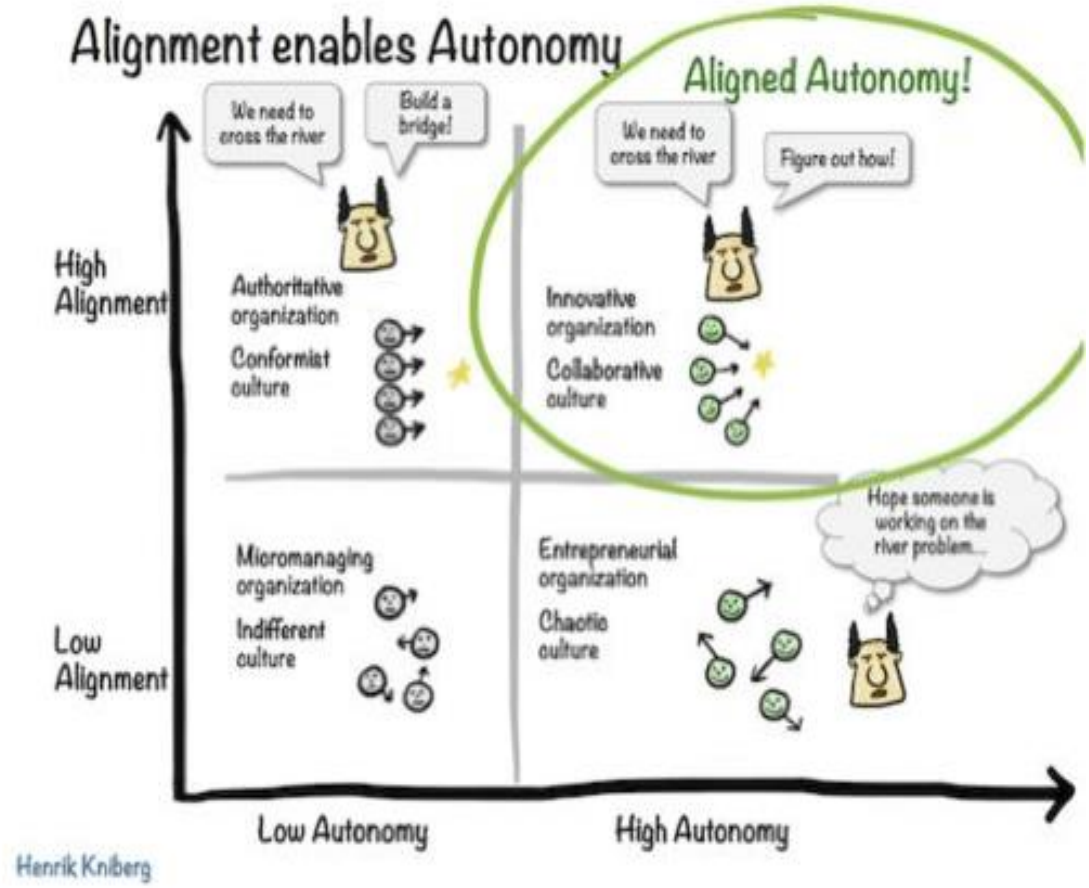
UI short/mid term perspectives:

- Maintain existing Java Swing
- Organize transition to new GUI technologies
- Deploy Grafana for monitoring on the control systems
- Provide Python ecosystem

Accelerators & Beamlines requests:

- Python/PyQt and other Python packages
- Web solutions

Computing teams & business teams work together to share the strategy



Expectations from the workshop

Understanding organization in the other institutes

- Who do what? How?
- Experience of migrating to new GUI technologies: skills, complexity
- What about long-term maintenance of the GUI? Which policy did you choose?
- UX Expertise?

Understanding operational architecture up to the desktop

- Cloud technology approach
- What about security for these upcoming solutions?
- What about library packaging?
- How to transition smoothly?

Feedbacks about Taurus

- Users' feedbacks from other institutes
- Which skills required to provide and deliver Taurus?
- Technical feedbacks: complexity, maintainability

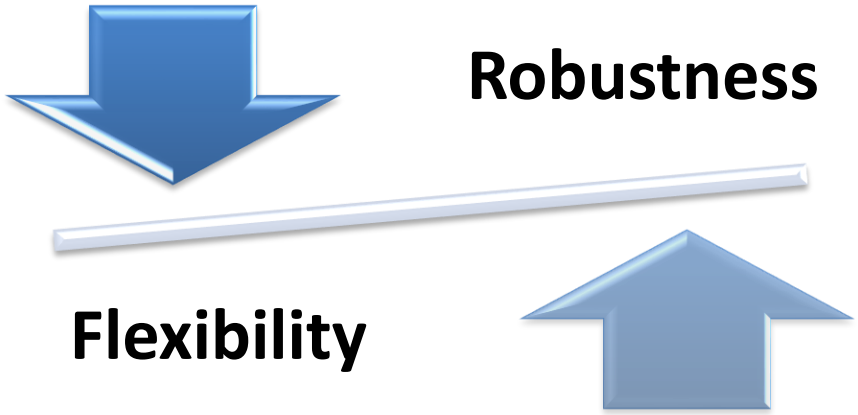
Conclusion

GUI strategy under construction in the context of SOLEIL II TDR

Current challenge is to manage the transition

Balance between desktop and web to be defined

Looking for partners to share UI frameworks and experiences



Architecture and technology Transformation

complexity/reliability/maintenability

Collaboration
Strong community

SOLEIL SYNCHROTRON
UPGRADE