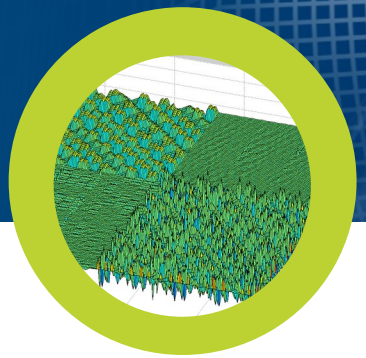
HZB BESSY II Light Source

OPERATIONAL TOOLS AND ALGORITHMS AT BESSY II AND MLS

19.06.2024, T. Birke, W. Sulaiman Khail,
T. Olsson, M. Ries, A. Schälicke, P. Schnizer et al.



CONTENTS

- Where are we?
  Overview about existing stack of operational tools and algorithms applied at BESSY II and MLS
- What do we do presently?
- Where do we want to go?

HZB / Dirk Laubner

BESSY II

1.7 GeV, 300 mA, TopUp, in operation since 1998

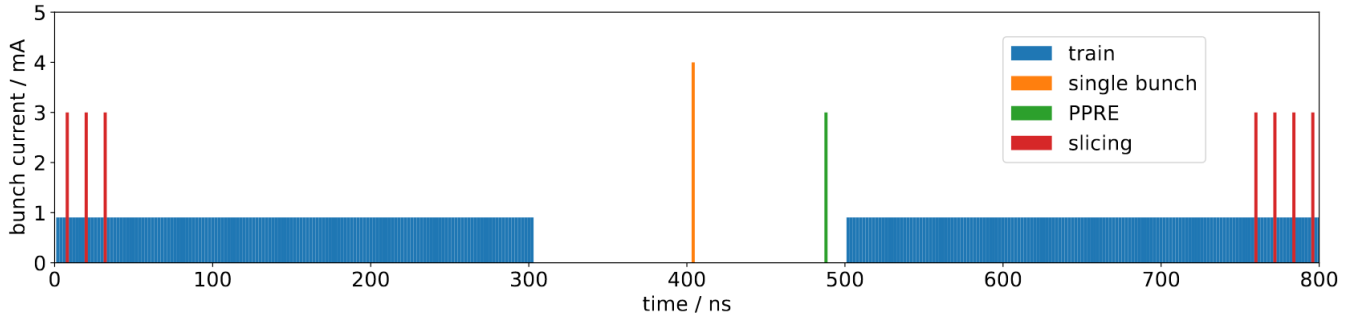
slicing facility, superconducting wavelength shifter (x2), in-vacuum undulator

hybrid filling pattern operation
- flux / brightness
- brilliance
- timing / dynamics

single bunch, few bunch
- timing / dynamics, pulse length ~ 100ps

low-alpha
- coherent THz radiation (timing) 2ps

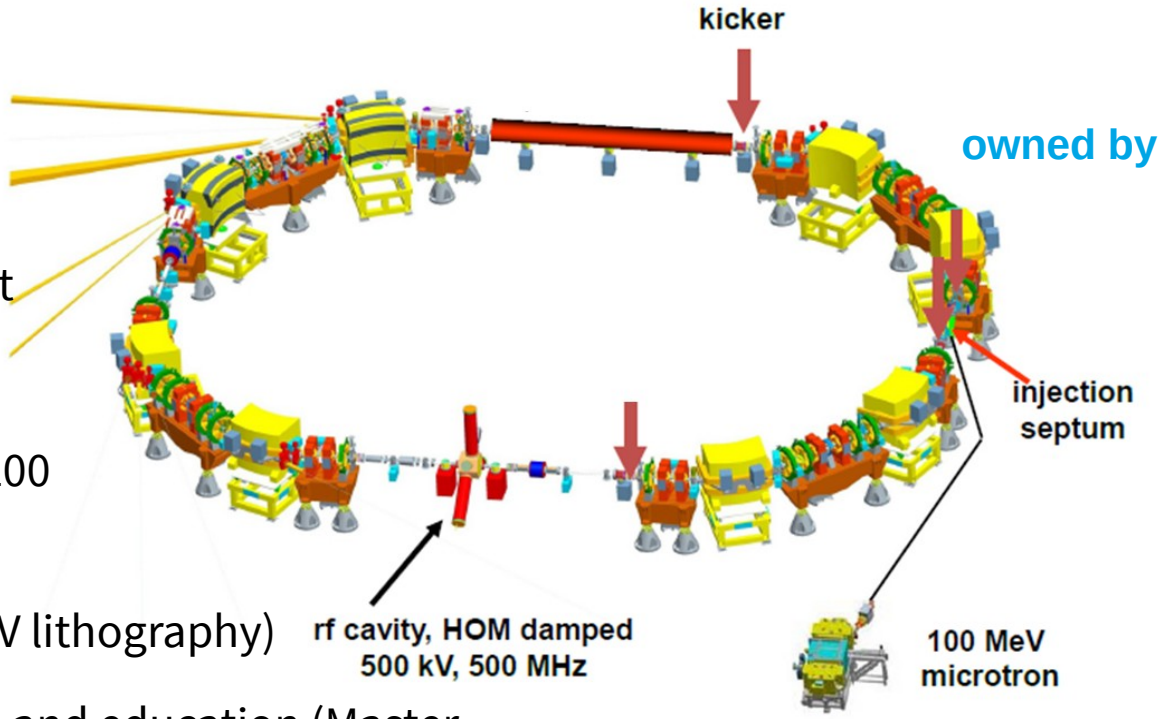
EPICS facility



METROLOGY LIGHT SOURCE

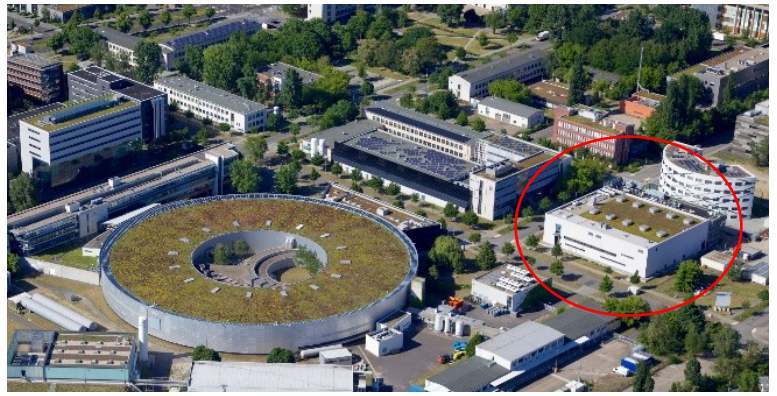
owned by PTB

developed and operated by HZB



- Physikalisch-Technische Bundesanstalt
- user operation since 2008
- E / MeV = 50…630; C / m = 48; I / mA = 200
  design emittance / nradm ~ 100
- EUV reflectometry (next generation EUV lithography)
- important test-bed for accelerator R&D and education (Master, PhD Students)
- first SR optimized for low alpha operation by using a dedicated sextupole and octupole correction scheme.
  → optimized for low-alpha operation (IR / THz generation)
- experimental studies of steady state microbunching (SSMB)
- EPICS facility

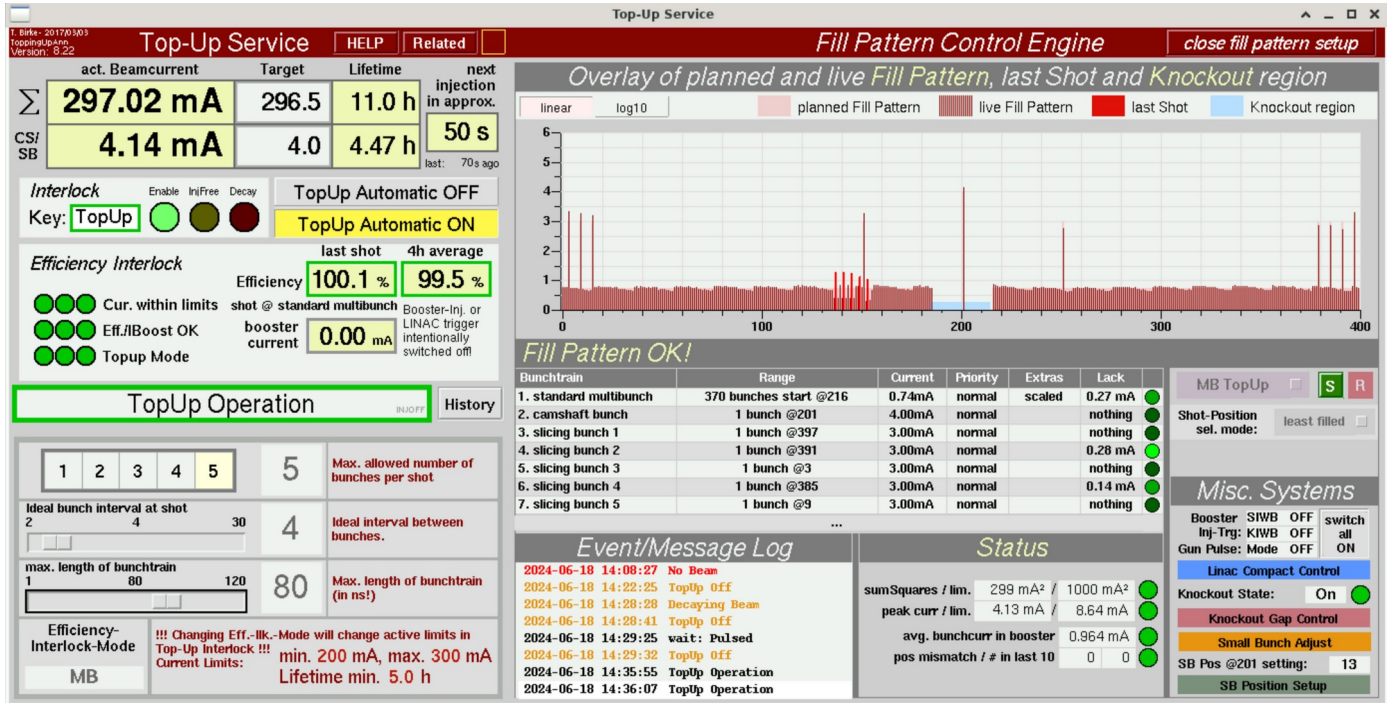
ASML ZEISS



HZB BESSY II Light Source

TOPUP MASTER – BESSY II

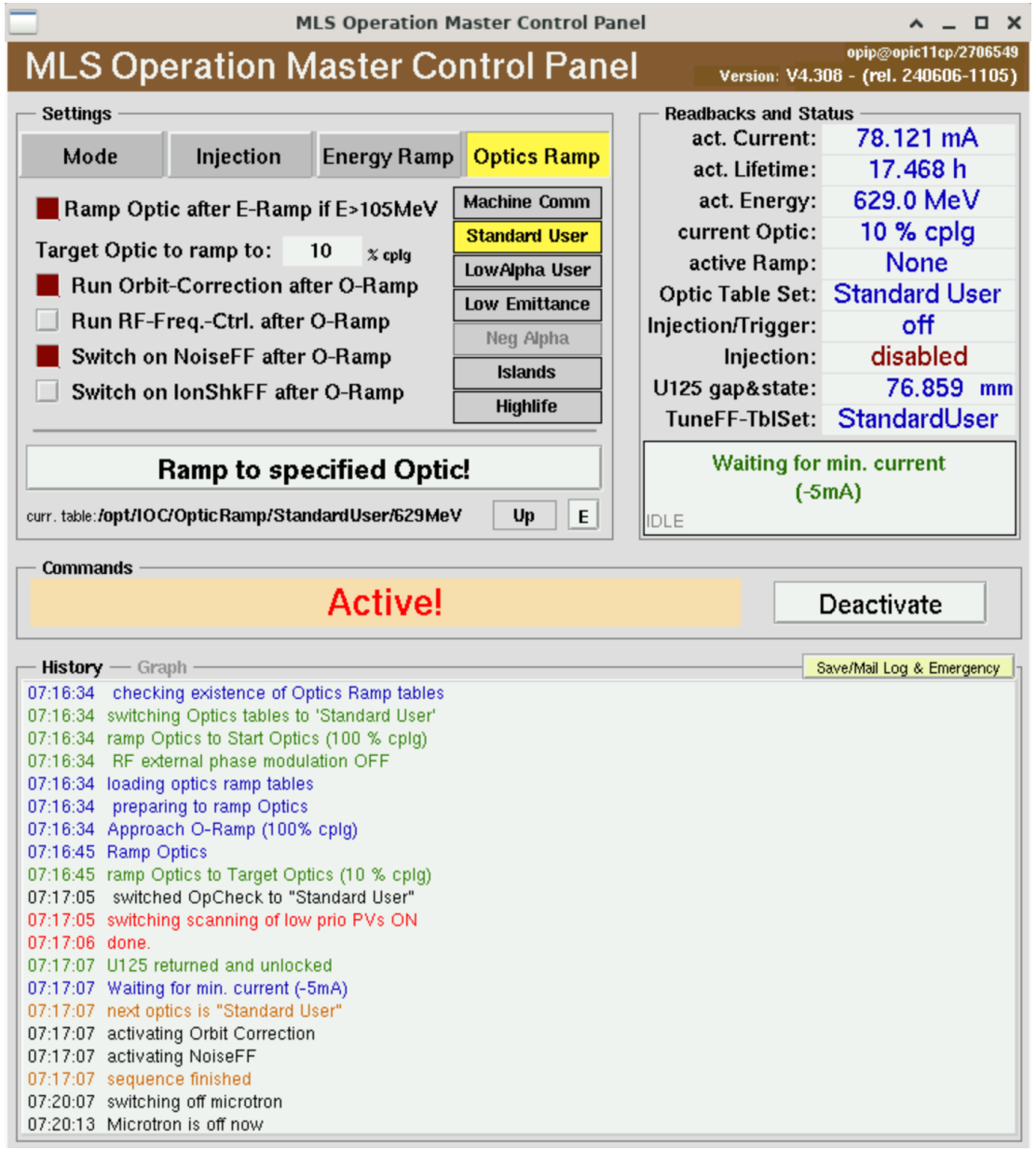
- BESSY II, self-written, maintained, EPICS State Notation
- automatized TopUp operation including helper functions
- lines of code ~600 state machine, ~600 next shot config, ~400 fill pattern and TopUp simulation for testing
- Running on an IOC, interface is EPICS (including panels, alarms, archiving…)
- cleanup planned, rewrite for next-shot config



OPERATION MASTER MLS

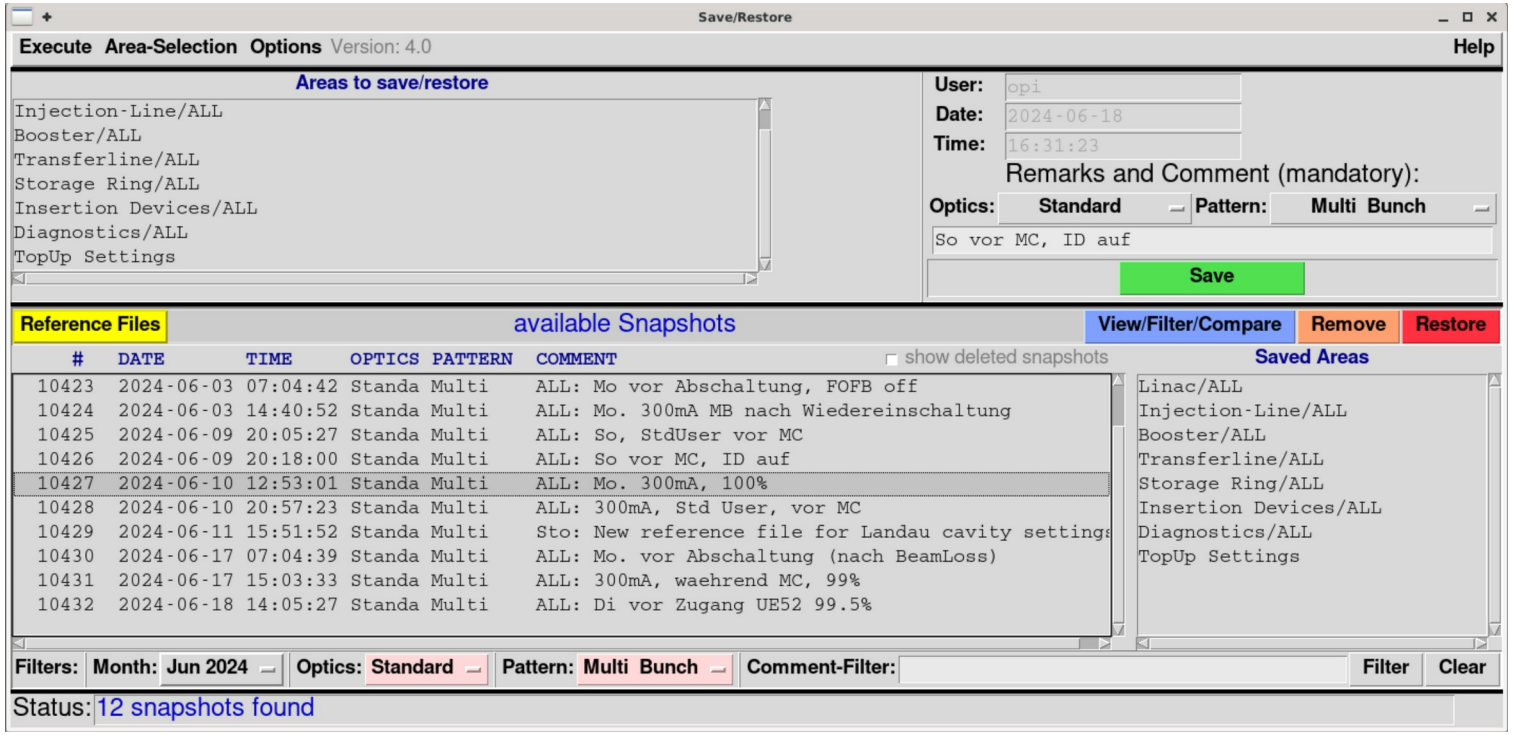
- MLS, self-written, maintained, Tcl
- fully automatized operation including all transitions
- lines of code ~250 generic state machine, ~500 MLS Operation Master, ~2000 initialization & helper functions
- running as a service, interface is EPICS-DB (including panels, alarms, archiving…)
- rewrite with cleanup planned

Fully automatized operation is key to efficient and reliable operation of the Metrology Light Source



SAVE & RESTORE

- BESSY II & MLS, Tcl/Tk
- configuration – hierarchy of files listing PVs to save/restore (in engineering units)
- 10000 machine snapshots so far
- lines of code ~3700
- running as application on the user terminal, no python or command line interface
- probably switch to Phoebus S&R in the future

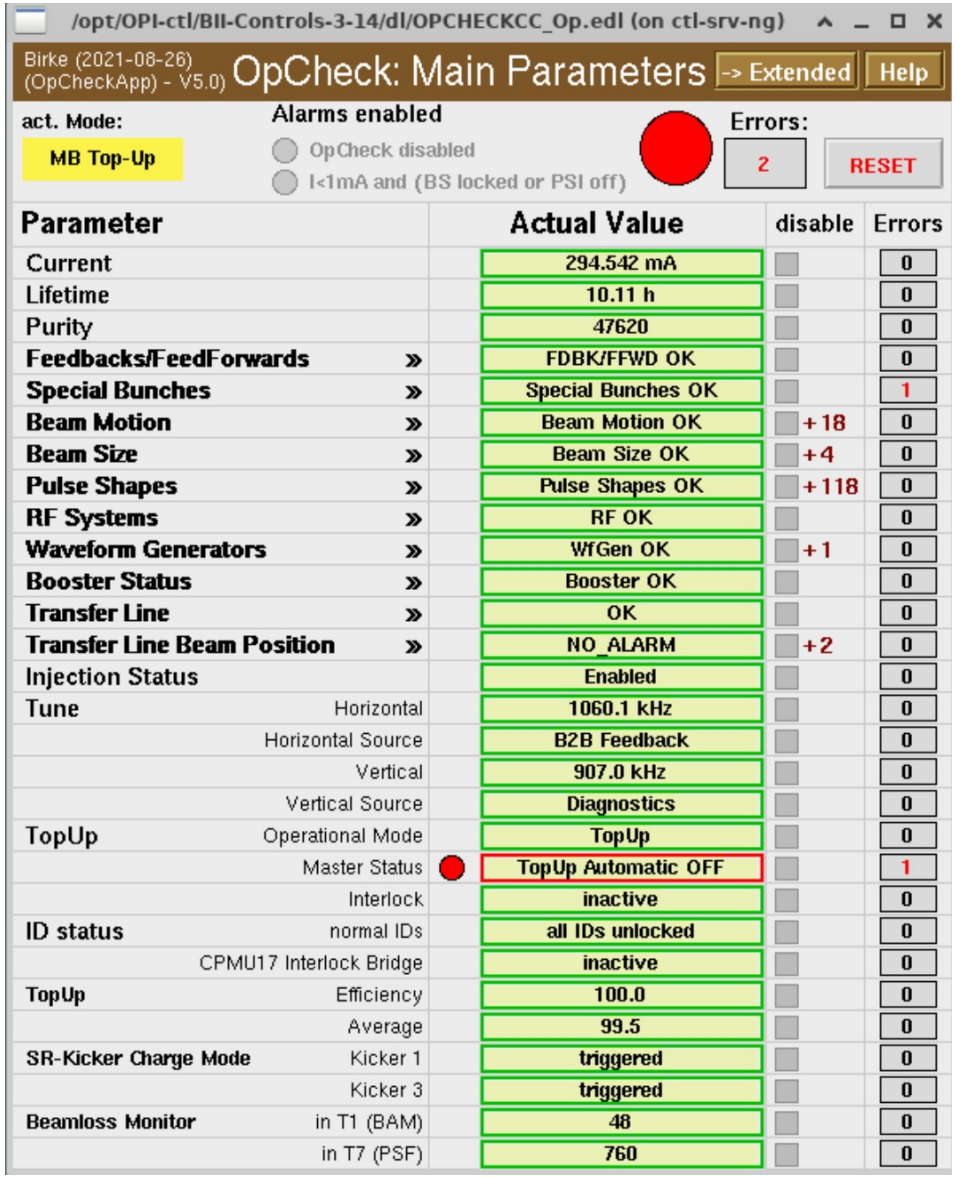


OPERATIONAL PARAMETERS CHECK

- BESSY II & MLS, self-written, maintained, EPICS
- automatized monitoring of beam parameters and machine configuration (in contrast to device health ALH)
- lines of code ~3000 automatic generation of EPICS-DB & config data for BESSY II and MLS
- Running as a service, interface is EPICS (including panels, alarms, archiving…)
- 16500 records for BESSY II and 5000 records for MLS

Very important for high availability & reliability

“eyes” of virtual supervisor / operator



ENERGY & OPTICS RAMP

- used at MLS, self-written, maintained, EPICS State Notation
- linear transition of complex ramps with human maintained fixed points
- lines of code ~300 generic state machine, ~500 python
- running as a service, interface is EPICS-DB (including panels, alarms, archiving…)
- rarely operated by humans, but daily but Operation Master

Mandatory ingredient for maintaining the complex state space of MLS

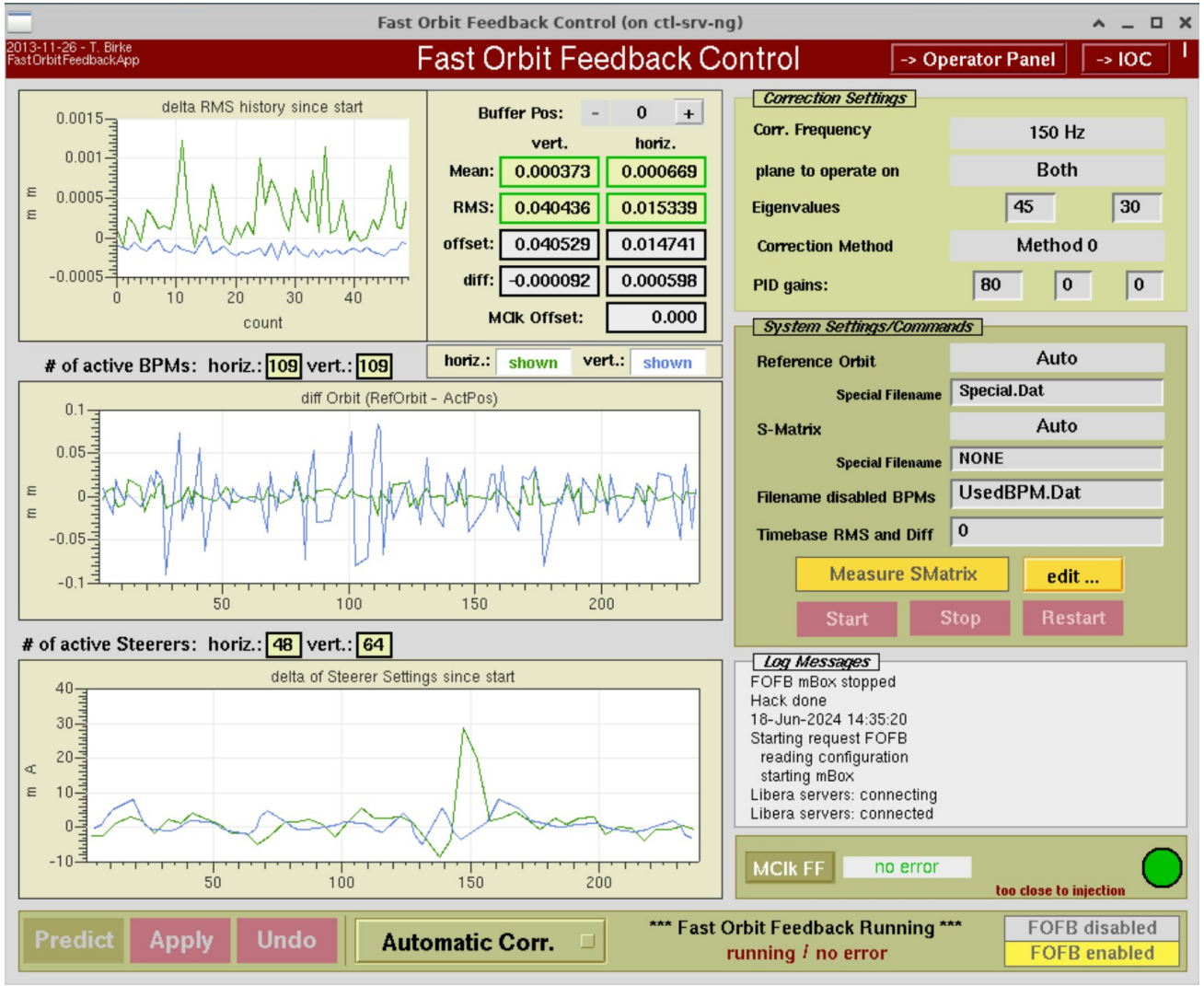
SLOW ORBIT FEEDBACK

- BESSY II & MLS, self-written, legacy, C++
- slow (1/6 s) orbit correction, visualization & bumps
- lines of code ~20000 orbit, ~37000 mapper (GUI - motif)
- local running application with GUI
- legacy code 32bit

We need urgently replacement(s) for correction and visualization
Community solution? Outsourcing?

FAST ORBIT FEEDBACK

- BESSY II, self-written, Matlab (7.7/2008)
- “fast” (150 Hz) orbit correction, visualization
- lines of code ~1300 orbit, ~2000 C-code auf IOCs
- local running application with GUI
- legacy system and software
  (reflective memory)
  planned to be rewritten soon as new BPM
  electronics are deployed

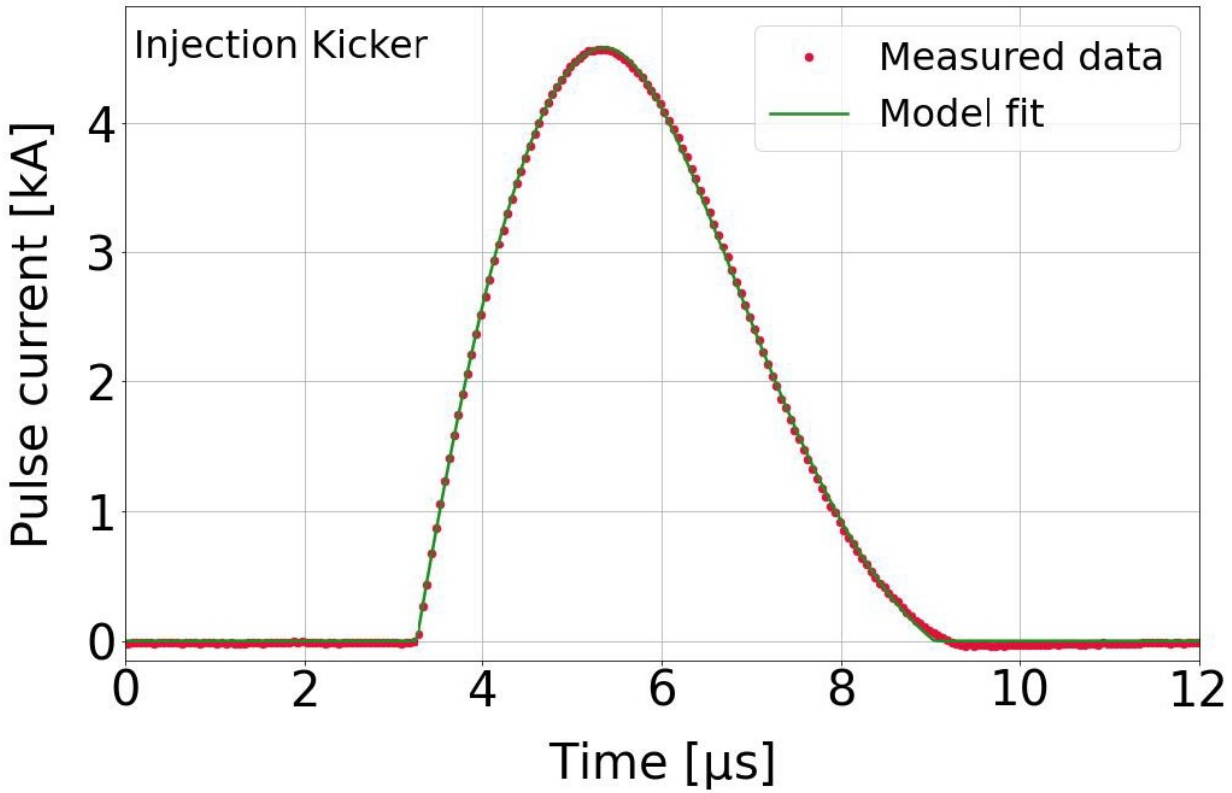


ADVANCED PULSED ELEMENT ANALYSIS

- BESSY II, self-written, maintained, python
- online model-based analysis of pulse currents for half-wave and full wave pulses
- lines of code ~ 400
- running as a service, interface is EPICS (including panels, alarms, archiving…)

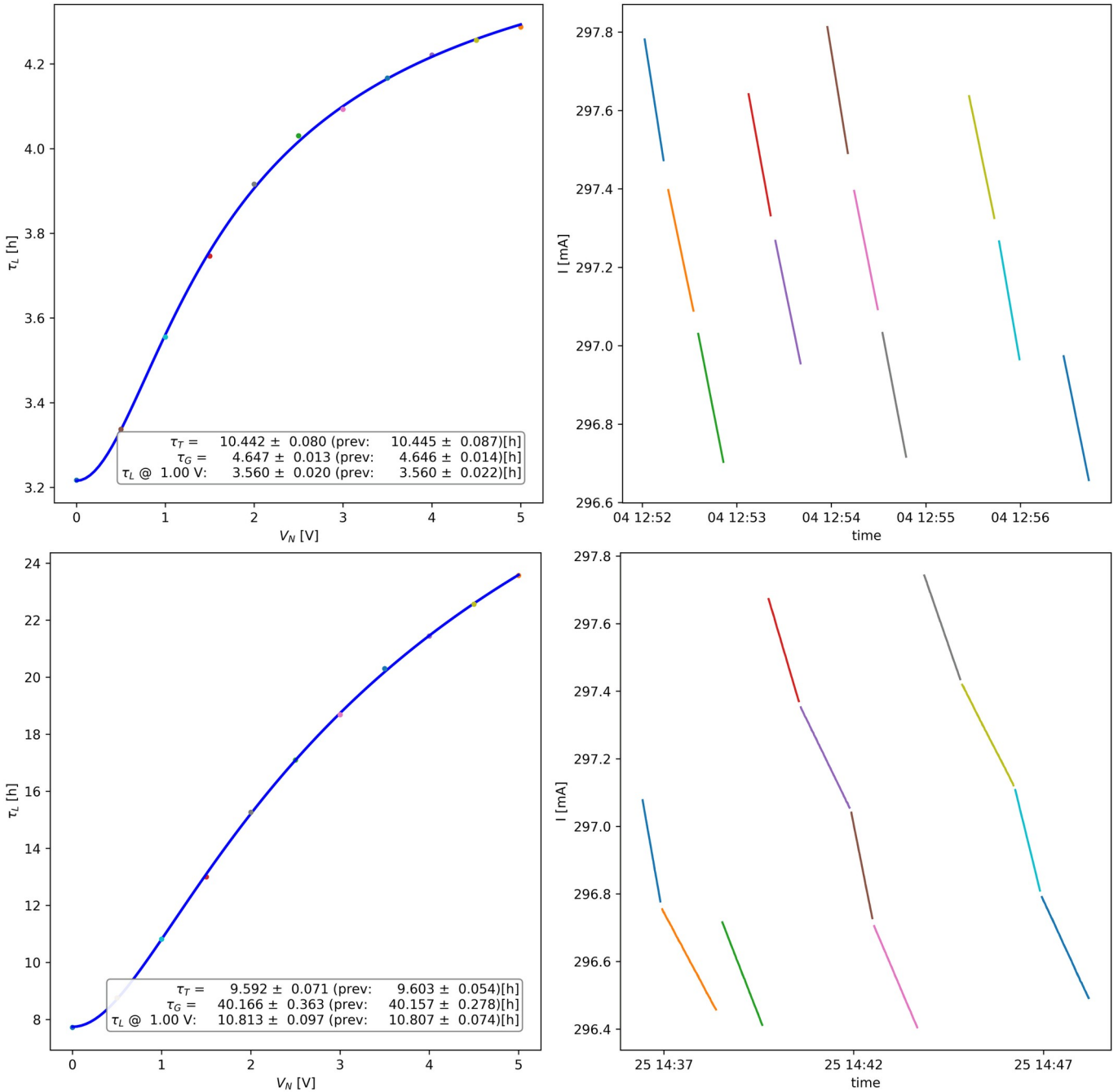
Permanent monitoring of amplitudes of septum currents in the range is extremely helpful for TopUp with 99% injection efficiency

$$I_{\text{model}}(t) = \begin{cases} 0 & , t < t_0 \ \& \ t > t_2 \\ I_{\text{rising}}(t) & , t_0 \leqslant t < t_1 \\ I_{\text{falling}}(t) & , t_1 \leqslant t \leqslant t_2 \end{cases}$$



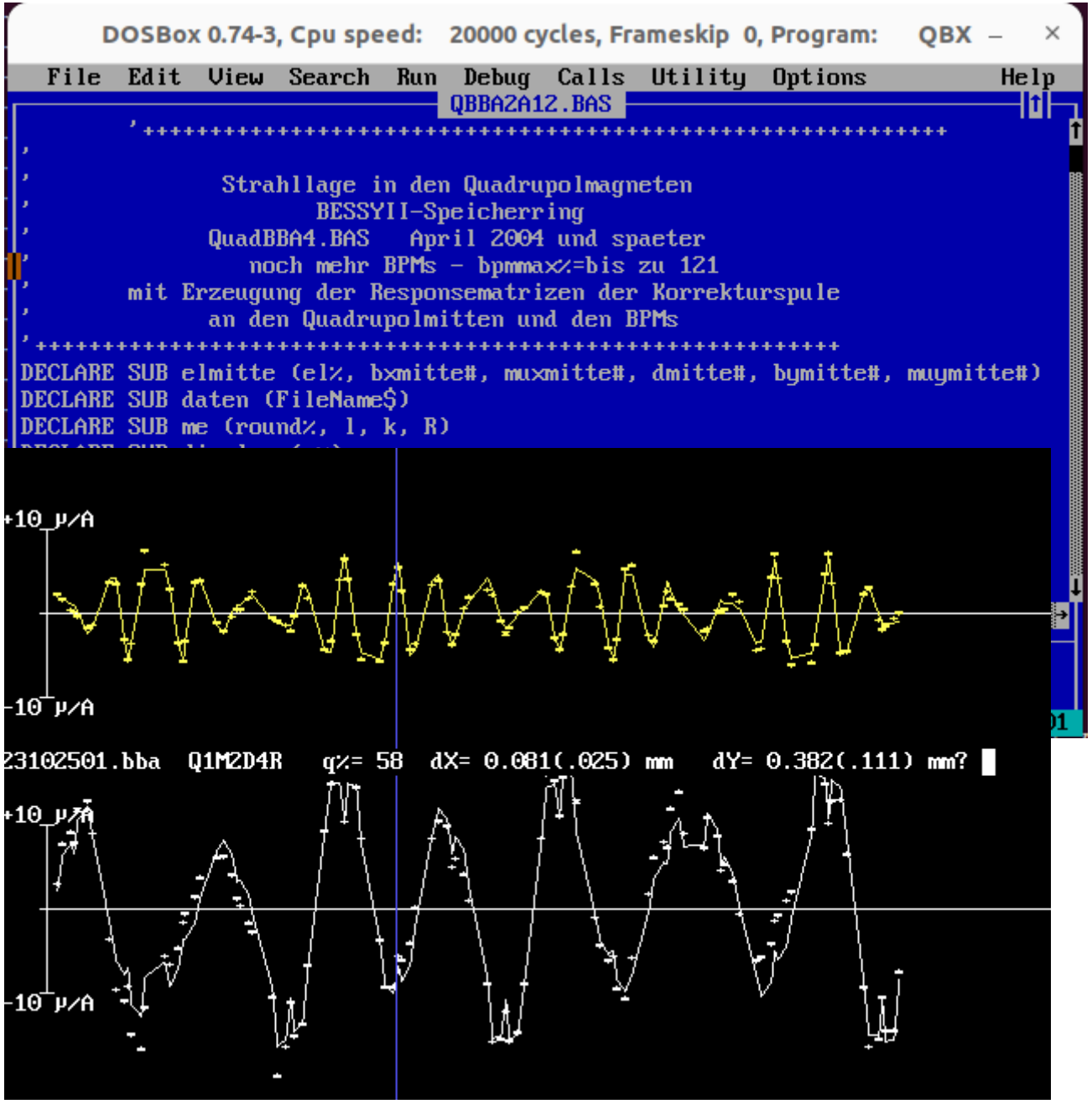
LIFETIME MEASUREMENT

- BESSY II, self-written, maintained, python
- weekly monitoring ratio of Touschek lifetime and gas lifetime
- operator friendly → fully automatized measurement, analysis and documentation (automated ELOG posts)
- local application including GUI



BBA-OLD

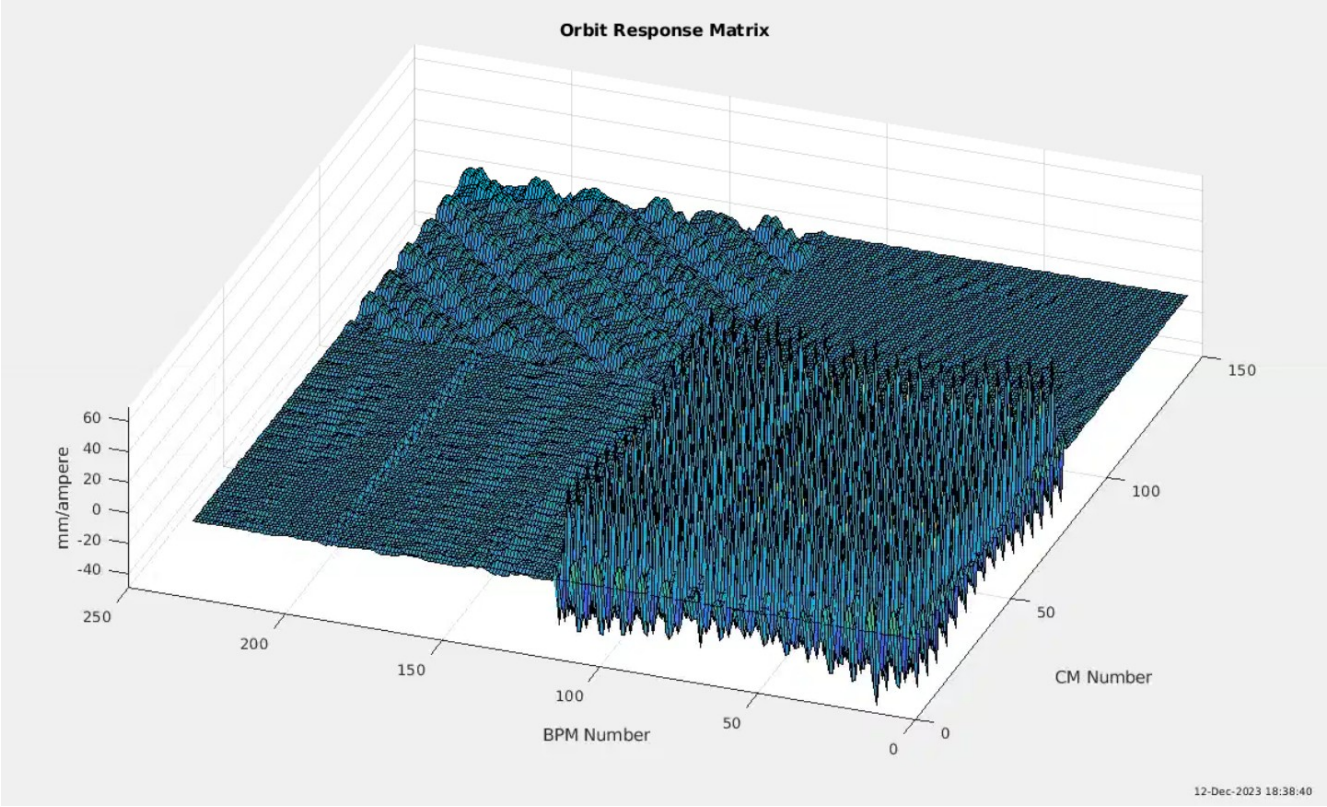
- BESSY II, self-written, unmaintained, QBasic (dosbox)
- model based beam-based alignment procedure (much faster) including golden orbit generation
- ~700 lines of code
- competitive at the time, but clearly out of date



In the process of being replaced by a modern rewrite and complemented by MML BBA (quadcenter)

MML

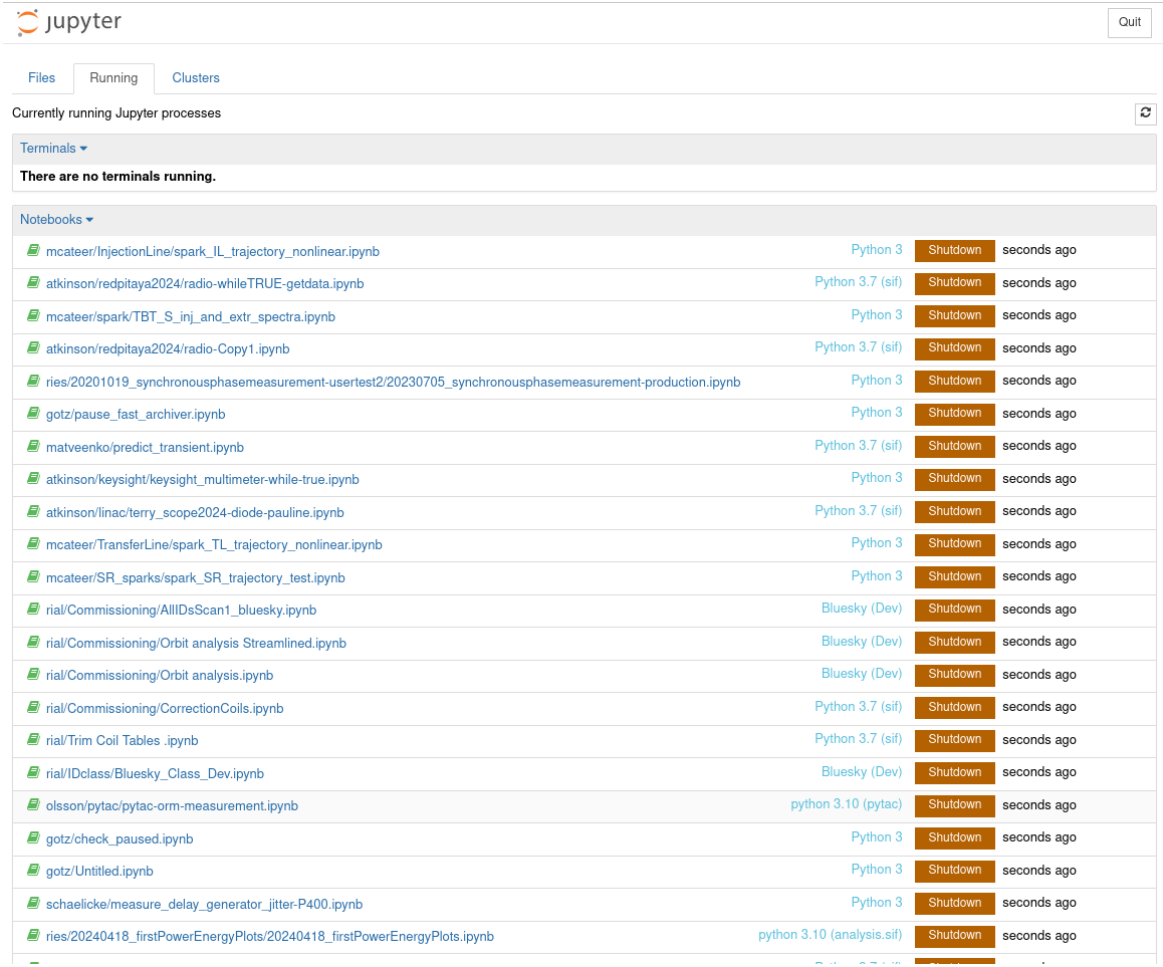
- established at BESSY II and MLS
- around for a decade, but not intensely used, maintained or developed, not part of the production eco system
- since half a year: complete model review, pyAT model input for MML
- extending MML setup and capabilities on both machines to connect to the community for preparing our contribution towards a new accelerator middle layer process
- orbit correction, generate bumps, quadcenter, ORM, LOCO



JUPYTER NOTEBOOK / LABS

- aka “commission tools”, managed to get all physicists under one umbrella
- daily accelerator physicists working place (debugging, data analysis and visualization…)
- lots of code snippets
- bad for many reasons (in particular sustainability)
- very low threshold – ideal for getting students on board
- ~5000 notebooks

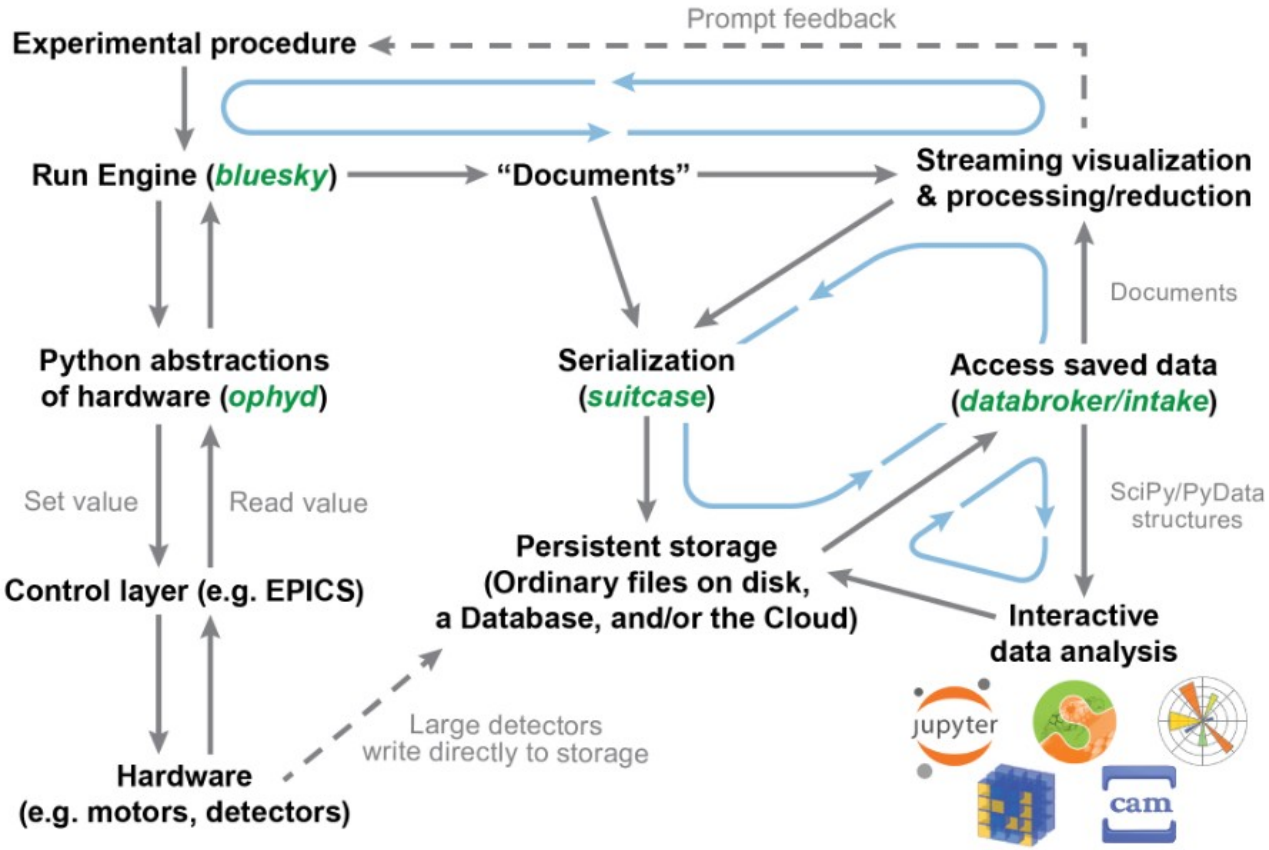
Seems to be a necessary evil for the time being



OPHYD / BLUESKY

- motivation: frameworks towards better – in particular more structured and sustainable physicists code
- ophyd → key / bottleneck are sustainable complex devices
- Bluesky → structured measurements / data acquisition
- needs some time to get used to database use (used to archive data or manually generated files)
- higher level integration enables much more complex possibilities than bare-metal

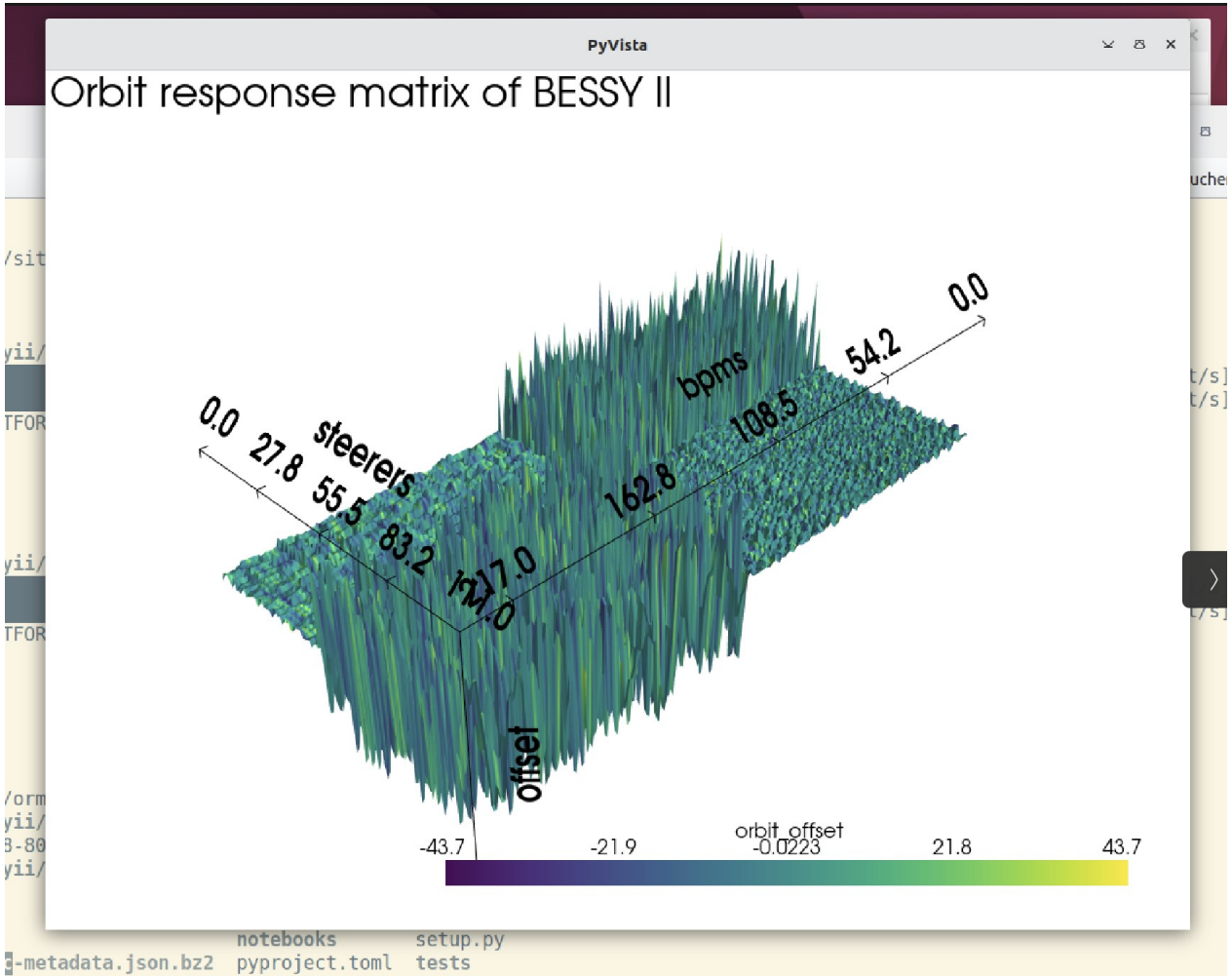
Work in progress, trying to deploy to "user" for some time now… quite a threshold to breach



ORM AND NEW BBA

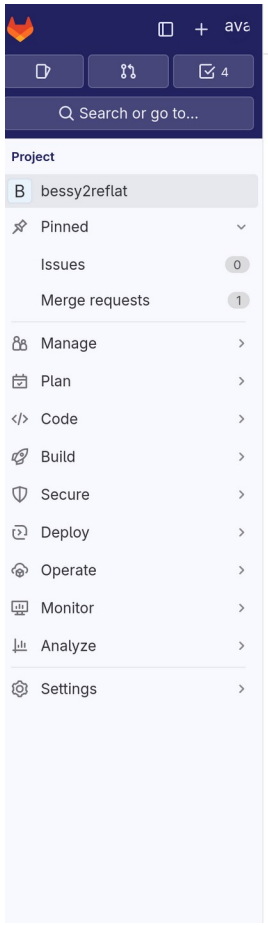
- BESSY II, self-written, development, python
- complete rewrite of the model old model-based BBA
- ophyd + bluesky + database
- runs on digital twin and machine
- in physics review
- Target: production method for BBA at BESSY II & MLS

Could be contributed as an application to a new accelerator middle layer as a standard tool



THE MACHINE MODEL

- 505 drift sections
- 35 dipole magnets
- 145 quadrupole magnets
- 112 sextupole magnets
- 112 (121) BPMs
- python interface → pyAT
  calculating tune & closed orbit ~ 80 ms without any optimization on a laptop
- using this model also in MML now
- in-depth model review ongoing
- targeting a model to describe the real machine



Accelerator Tools / Reference Lattices / **bessy2reflat**

```
        at.Quadrupole('Q4M1D1R',0.5,k=k4d1),\
        at.Drift('DU_S4M1D1R',0.153),\
        at.Sextupole('S4M1D1R',0.16,h=h4d1,Corrector='H',KickAngle=[0,0]),\
        at.Drift('DU_BPMZ4D1R',0.07),\
        at.Monitor('BPMZ4D1R',Group='BPM'),\
        at.Drift('DU_KIK1D1R',0.2295),\
        at.Drift('KIK1D1R',0.595),\
        at.Drift('DU_BPMZ41D1R',0.1265),\
        at.Monitor('BPMZ41D1R',Group='BPM'),\
        at.Drift('DU_BPMZ42D1R',0.20939),\
        at.Monitor('BPMZ42D1R',Group='BPM'),\
        #at.Drift('DU_KIK2D1R',0.13289-0.00639),\
        at.Drift('DU_KIK2D1R',0.12011),
        at.Drift('KIK2D1R',0.595),\
        at.Drift('DU_FOMZ1D1R',0.3465),\
        at.Monitor('FOMZ1D1R'),\
        at.Drift('DU_MRING_END',0.514),\
        at.Marker('MRING_END'),\
        ]

    # Ring definition
    ring=[D1_secondhalf, T1, D2, T2, D3, T3 , D4, T4, D5, T5, D6, T6, D7, T7, D8, T8, D1_firsthalf]

    # Expand ring until list completely flattened
    while any(isinstance(x, list) for x in ring):
        ring = list(chain.from_iterable(i if isinstance(i, list) else [i] for i in ring))

    # Build the lattice
    ring = at.Lattice(ring,name='BESSY II',energy=energy)

    # Turn cavity and radiation on
    ring.enable_6d()

    # Set cavpts to point to main cavities
    ring.cavpts = ring.get_uint32_index('CAV*')

    # Set only main cavity phases
    ring.set_cavity_phase(cavpts=ring.cavpts)

    return ring

if __name__ == '__main__':
    ring = bessy2Lattice()
```

DIGITAL TWIN

- core motivation: enable accelerator physics software development, testing and training outside the control room without machine time
- creates PVs with prefix and manages
- architecture with different engines → pyAT & thor-scsi
- twin for everyone → shall be running on office laptops
- container available

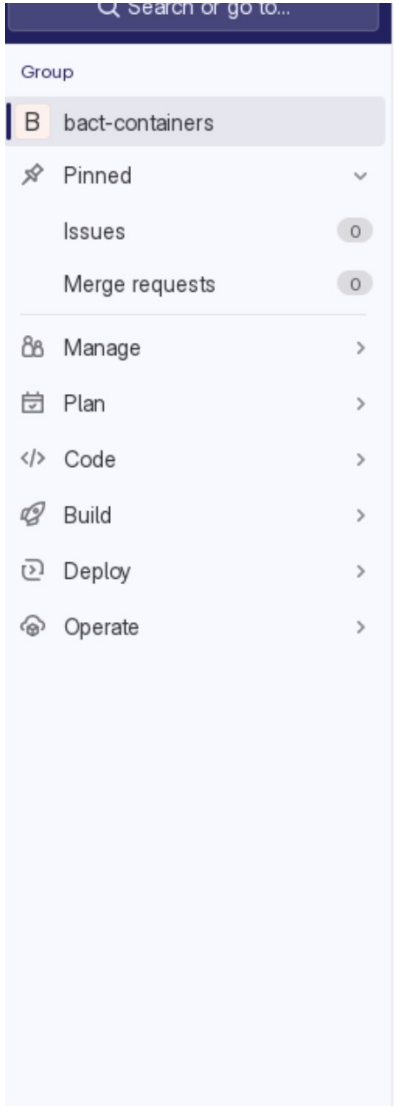
First friendly users profit,
e.g. bluesky @ BESSY II digital twin

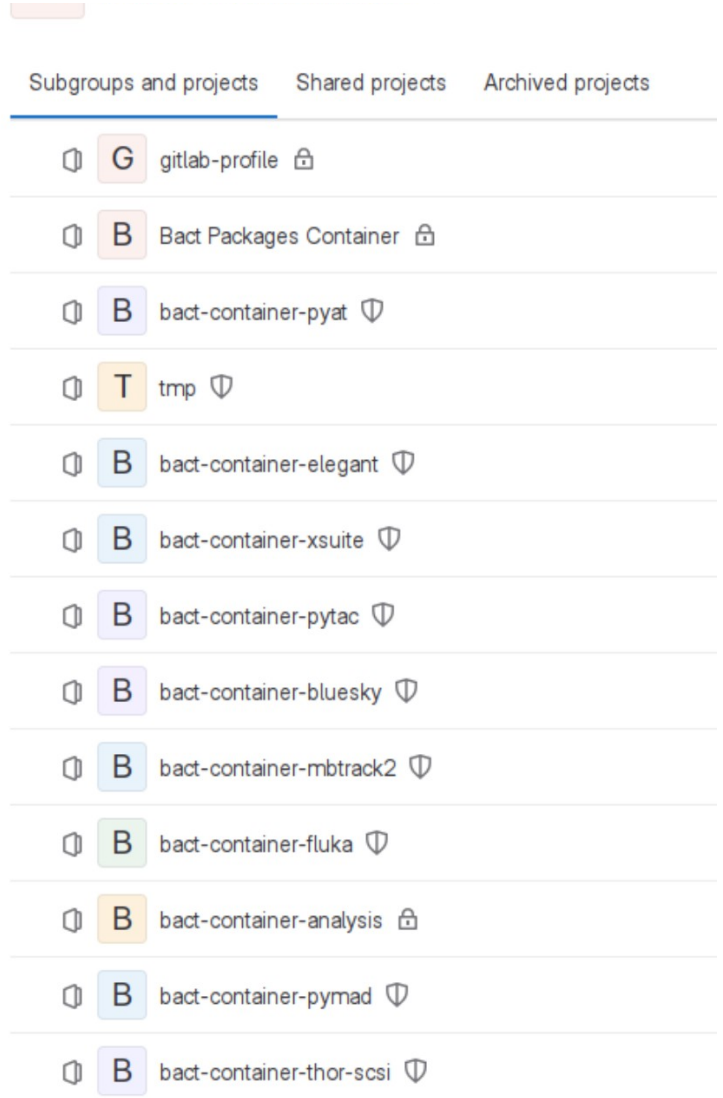
```
Starting iocInit
################################################################################
## EPICS R7.0.8.1-DEV
## Rev. R7.0.8-20-gc77f32b19cc81d309740
## Rev. Date Git: 2024-05-03 04:04:41 +0900
################################################################################
Forward-link uses Channel Access without pointing to PROC field
    Teresia:DT:dt:updates.FLNK => Teresia:DT:dt:im:update:fanout
recGblRecordError: scanAdd: I/O Intr not valid Error (65535,65535) PV: Teresia:D
_main_: dbProcess of 'Teresia:DT:PMUXZR:off'
_main_: dbProcess of 'Teresia:DT:QSPAZR:im:set'
_main_: dbProcess of 'Teresia:DT:par:mode'
_main_: dbProcess of 'Teresia:DT:mux:im:set'
_main_: dbProcess of 'Teresia:DT:mux:out'
_main_: dbProcess of 'Teresia:DT:QSPAZR:rdbk'
_main_: dbProcess of 'Teresia:DT:beam:orbit:found'
_main_: dbProcess of 'Teresia:DT:beam:publish'
iocRun: All initialization complete
epics> pyat orbit calculation starting (find_orbit)
pyat twiss calculation starting (get_optics)
label Teresia:DT:beam:orbit:found = True type(<class 'bool'>)
cbLow: dbProcess of Active 'Teresia:DT:beam:orbit:found' with RPRO=0
label Teresia:DT:beam:names
Executing Python code: mux.off(0)
ERROR: triggering calculation needs to be implemented
Executing Python code: publish(what='lattice')
Need to implement publishing lattice?
Executing Python code: pydev.iointr('Teresia:DT:beam:orbit:found')
epics>
```

CONTAINERS

- We try to establish a container based eco system for central maintenance and usage at
  - laptop
  - control room
  - HPC
  - Collaborating with partners
- recipes maintained in gitlab
- automatic building and testing via CI/CD
- containers can be access through a registry or download
- very helpful with students

Early phase… usage is ramping up now

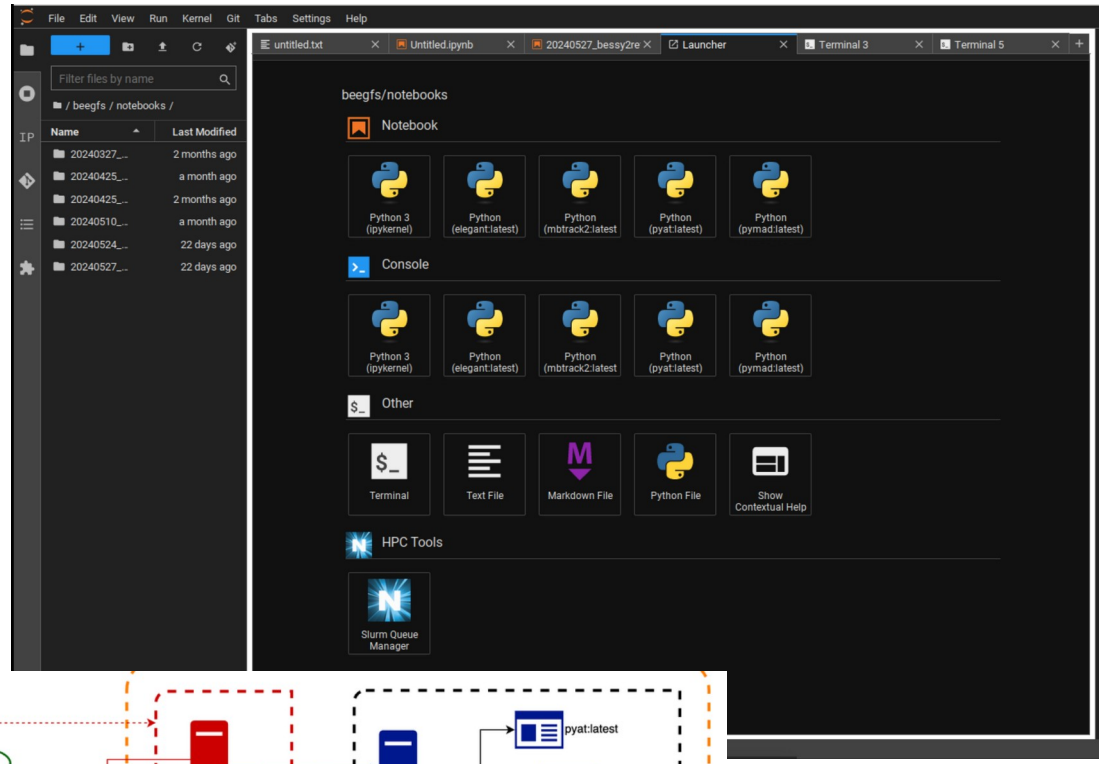


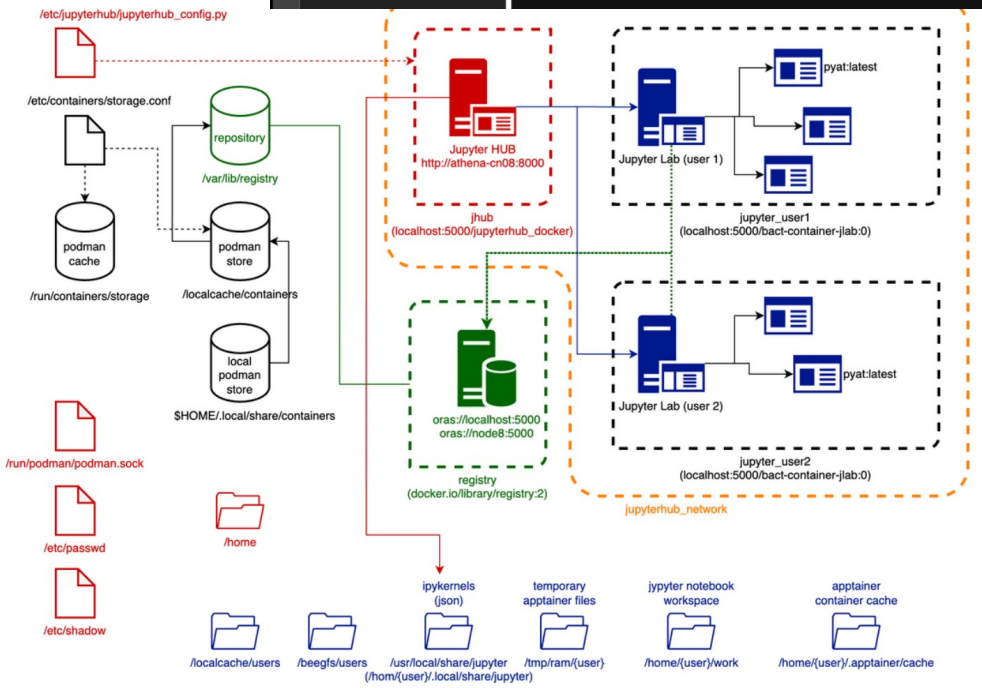


HPC

- 8 nodes
  total: 1536 cores, 3072 threads, 8TB RAM
- 1 x interactive + 7 x Slurm
- file system BeeGFS
- devoted to storage ring physics
- quite some backlash through cyber attack
  (fully self setup and maintained)

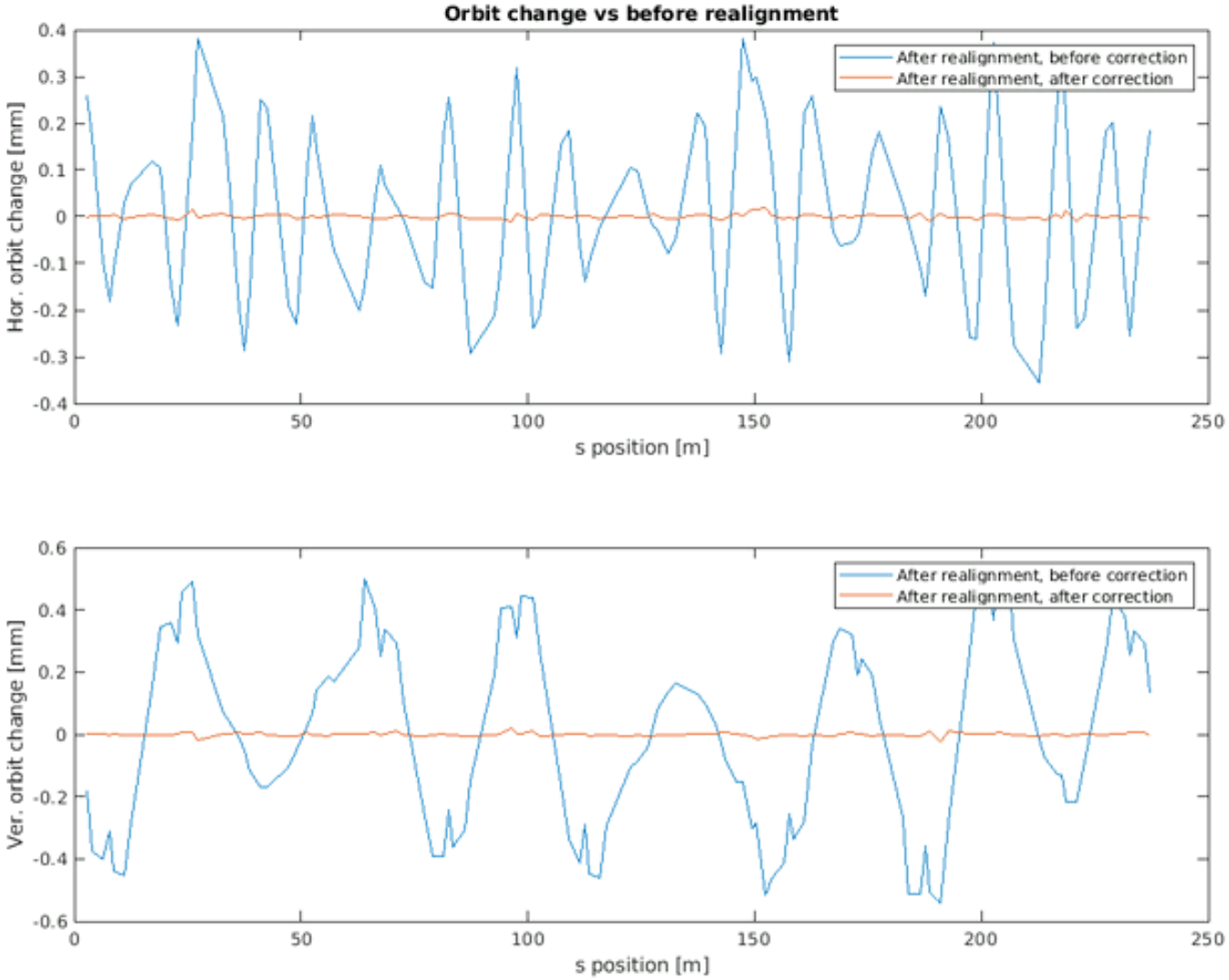
Friendly users are already getting used to lots of resources





BESSY III

- preparing at BESSY II, e.g.
- moving dipole by ~300µrad roll and 600µrad pitch cross checking with beam and pyAT model
- sufficient model for the flawed machine
- benchmark real simulated commissioning at BESSY II (not accelerator physics robustness analysis)



FUTURE

- end of the year we will have access to TbT data → new set of data & tools required
- use of more sustainable code, i.e. standardized and maintained by a larger community of multiple machine
  → participate, contribute and deploy to new middle layer effort
- “physics ready data”, data models, database
- fire up the use of the digital twin
- connect / align the tool stack with storage rings community
- trying to attract the interest of machine learning people

We do not want do drive island projects / solutions.
We want to share what we develop.

CONCLUSION

- operational tools are the backbone of high performance and reliability user operation and mandatory for more handling complex machines and operational states
- team moved to python, moving to pyAT, keep other codes supported as crosscheck
- difficult transition from “physicists demo code” to a real operations tool
- BESSY II and MLS are running fine, but…
  - deploy state of the art
  - sustainable codes are an issue
  - a lot of potential in the transition from bare-metal to a higher level
- BESSY III … is a long way to go, but we try to find the path an prepare at the existing facilities

We can and want to contribute!
We have excellent machine access (2!)
and a young team of coding physicists + software engineer

THE EXPECTATION / DREAM / ILLUSION