UPDATE ON THE EBS STORAGE RING BEAM DYNAMICS DIGITAL TWIN

S.M. Liuzzo*, J.-L. Pons†, N. Carmignani, L.R. Carver, L. Hoummi,
N. Leclercq, T. Perron, S. White, ESRF, Grenoble, France

Abstract

The EBS storage ring control system is presently paired with an electron beam dynamics digital twin (the EBS control system simulator, EBSS). The EBSS reproduces many of the beam dynamics related quantities relevant for machine operation. This digital twin is used for the preparation and debug of software to deploy for operation. The EBSS is presently working only for the main storage ring and it is not directly connected to the machine operation but works in parallel and on demand. We present here the steps taken towards an on-line continuous use of the EBSS to monitor the evolution of not directly observable parameters such as beam optics. Keywords: digital twin, storage ring, tango, python

INTRODUCTION

The ESRF storage ring was upgraded in 2019 to provide brighter X-rays [1]. A very tight schedule was defined for dismantling, installing and commissioning of the new storage ring. In order to cope with this strict schedule, the design and update of the whole software infrastructure had to be performed as early as possible. High-level control applications needed on the first day of commissioning, such as the magnets control, orbit correction, tune correction and beam threading algorithms [2] were tested using the EBS control system Simulator [3].

This control system simulator was strongly focused on the EBS magnets control system that needed to be completely redesigned, but included as well most of the diagnostic devices (beam position monitor (BPMs), tunes, emittances, etc..) needed for the development of the tools used for beam operation. The output values of the simulated diagnostic devices are generated from a given lattice optics model (in pyAT [4]) that is updated upon a magnetic strength or RF frequency variations in the simulated control system - as depicted in Fig. 1. This structure results in an effective beam dynamics digital twin (the EBS control system simulator, EBSS): the actions performed in the simulator are strictly identical to those performed on the real machine and the results are extremely similar or identical. The major limitations are the absence of computation of beam Lifetime, losses, injection efficiency and collective effects [5].

From the user's point of view, the simulator is identical to the production control system. For example the devices providing the information on the beam position have identical names for attributes and properties as their physical counterparts. This is the case also for tune, emittance, RF and current devices. This allows to transfer the applications from the simulator environment to the real machine by simply changing the environment variable pointing to the TANGO [6] database.

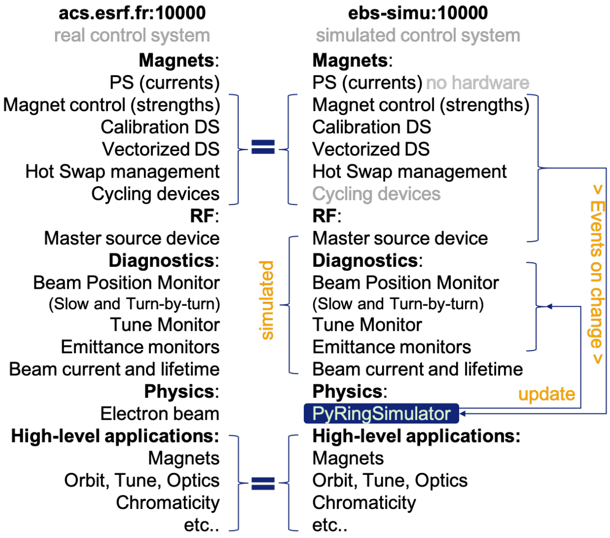


Figure 1: Real control system vs EBS simulator.

EBSS is a clone of almost all single particle beam dynamics features of the EBS storage ring control system. It allows to interact with a simulated beam and to visualize the expected behaviour of most of the relevant electron beam and lattice observables: orbit, tunes, emittances, optics and coupling.

In the past, the ability to run off-line control room applications was already available, for example via toolkits such as the Matlab-Middle-Layer [7], used in many 3rd generation light sources or specific solutions - like the one on which the Virtual XFEL [8] relies.

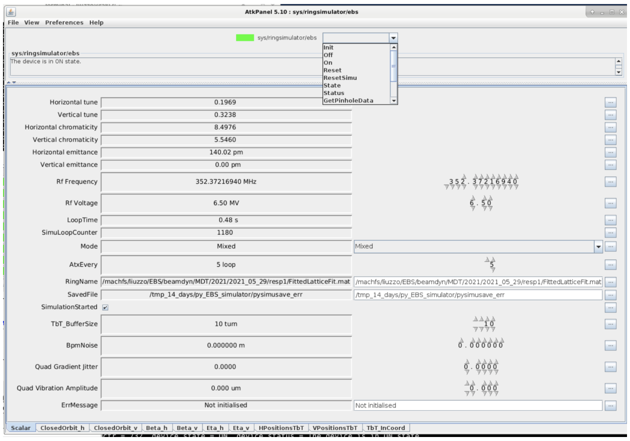
STRUCTURE OF THE SIMULATOR

The core of an EBSS instance [9] is composed of more than ~4000 Tango devices compared to ~25000 in the physical accelerator complex (including the 3 accelerators - linac, booster + storage ring). Each EBSS instance runs on its own Tango control system - the associated Tango database being isolated in a dedicated physical machine. This allows multiple EBSS to run simultaneously and independently on the same host. Presently three EBSS instances are deployed at the ESRF - each one on a 16-CPUs/64-Gb host. Initially the different simulators were run on individual Docker containers, but this was more complicated for the users compared to simply dedicating a virtual machine to each EBSS instance.

---

* simone.liuzzo@esrf.fr
† pons@esrf.fr

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-THPDP010



sys/ringsimulator/ebs
The device is in ON state.

Horizontal tune	0.1969	
Vertical tune	0.3238	
Horizontal chromaticity	8.4976	
Vertical chromaticity	5.5460	
Horizontal emittance	140.02 pm	
Vertical emittance	0.00 pm	
Rf Frequency	352.37216940 MHz	352.37216940
Rf Voltage	6.50 MV	6.50
LoopTime	0.48 s	
SimuLoopCounter	1180	
Mode	Mixed	Mixed
AtxEvery	5 loop	5
RingName	/machfs/liuzzo/EBS/beamdyn/MDT/2021/2021_05_29/resp1/FittedLatticeFit.mat	/machfs/liuzzo/EBS/beamdyn/MDT/2021/2021_05_29/resp1/FittedLatticeFit.mat
SavedFile	/tmp_14_days/py_EBS_simulator/pysimusave_err	/tmp_14_days/py_EBS_simulator/pysimusave_err
SimulationStarted	☑	
TbT_BufferSize	10 turn	10
BpmNoise	0.000000 m	0.000000
Quad Gradient Jitter	0.0000	0.0000
Quad Vibration Amplitude	0.000 um	0.000
ErrMessage	Not initialised	Not initialised

Figure 2: PyRingSimulator ATK panel view.

Tango Device Servers

Each EBSS Tango device exposes the exact same interface as its production counterpart. This ensures a total transparency to its clients and guarantees a smooth transition from simulation to production. At the lowest level of the control system, most of the devices are written in C++ and support a compilation flag providing a way to specify the target platform - i.e., simulation or production. For instance, the BPMs read the beam position from the dedicated hardware when running in production mode, whereas they obtain it from the ring simulator while running in the context of an EBSS instance. The behaviour of some devices has also been modified in order to broadcast or specifically notify (upon change) the EBSS environment - e.g., a strength modification on a magnet will trigger a modification of the EBS storage ring magnetic model and a consequent update of the relevant beam parameters. These asynchronous notifications rely only on the Tango events mechanism. A total of ~100 Device Classes and ~500 Device Servers are configured to run the EBSS.

Physics Core: PyRingSimulator The PyRingSimulator device server (see Fig. 2) runs high energy electron beam dynamics simulations [10].

The PyRingSimulator DS runs a dedicated process at startup (`ebs_simu.py`) that continuously computes the optics using python Accelerator Toolbox (pyAT) [4]. The loop runs continuously and not only upon changes to allow the introduction of stochastic errors. For each EBSS instance, one CPU core is fully dedicated to this process. This structure allows to better exploit the CPU resources and provides a clear separation among the issues and workload related to control and those related purely to beam dynamics. Moreover, this separation allows for future extensions of the beam dynamic simulations (including lifetime and injection efficiency) to exploit more CPU resources, GPUs or the ESRF computing cluster, keeping the pyTANGO DS unchanged.

The simulated diagnostic DS continuously reads/monitors the updated parameters in PyRingSimulator instead of measuring real beam properties. From the user's point of view, the devices providing orbit, tune, chromaticity, etc, behave as the physical ones available in the control room. The PyRingSimulator device is the only one not existing in the real control system: it actually replaces the electron beam. More details about the PyRingSimulator DS may be found in [3].

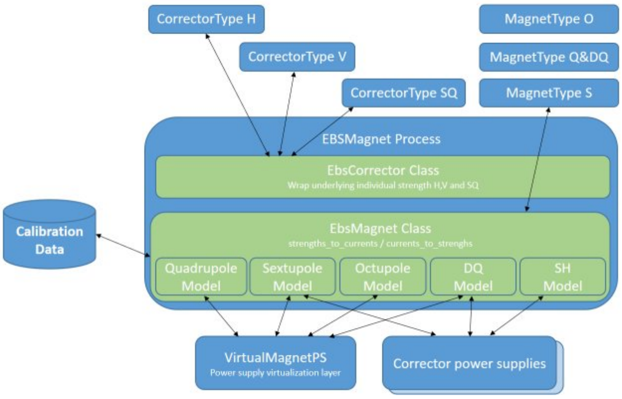


Figure 3: EbsMagnet class architecture.

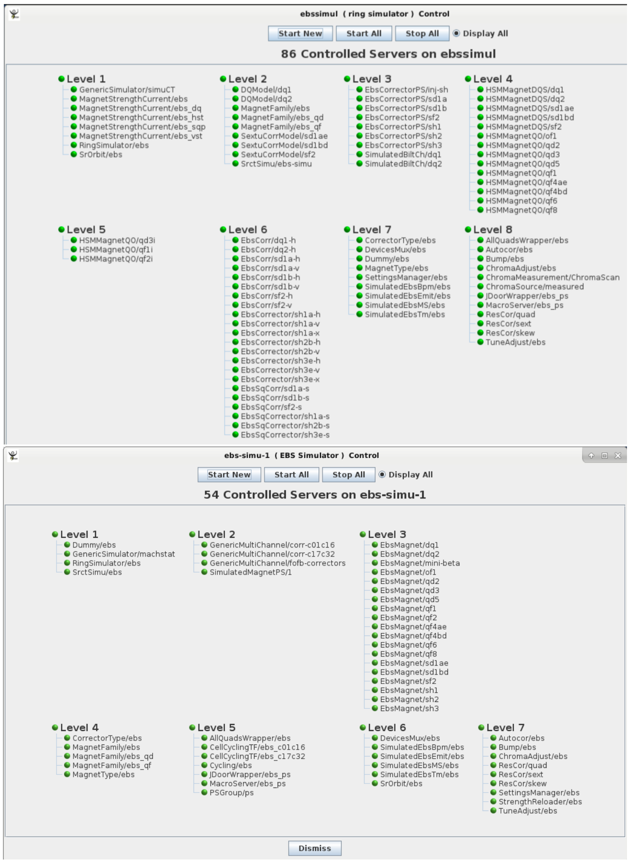
Simulated Lifetime/Current Decay The lifetime/current device is a standalone device implementing an exponential decay starting from a given current, with a given lifetime. Simulations of beam (Touschek) lifetime are at the moment impractically long (hours on a single core, minutes on a computing cluster), and are not crucial for the purpose of the EBSS. A proposal to run such simulations based on artificial intelligence and deep learning is under evaluation and could be the solution to this missing component of the EBSS.

RECENT DEVELOPMENTS PROFITING OF THE EBSS

New Magnets Control Architecture

The EBSS proved to be extremely useful not only before commissioning but also during the operation of the EBS storage ring, in particular to prepare the deployment of new control system features. After a first version (operational for about four years) the magnets control system has been re-written from scratch. For example with the first version of the control system: 1) it was not possible to vary the DesignStrengths attribute of some magnets without killing the beam, 2) the strength set points for arrays of strengths were not always correctly set especially for combined function magnets such as sextupoles, 3) transient CORBA errors [11] were frequent and it was necessary to wait a sensible amount of time (seconds) among two consecutive inputs, 4) some correctors set points were actually ignored, 5) it was not possible to integrate easily additional magnets (for example those needed for mini-beta optics, see later).

The main goal of this refurbishment was first to simplify and ease the maintainability of the EBS magnet software. The new structure has been designed to centralize all magnet calibration data and calculations in a single object in



ebssimul (ring simulator) Control

86 Controlled Servers on ebssimul

Level 1
- GenericSimulator/simuCT
- MagnetStrengthCurrent/ebs
- MagnetStrengthCurrent/ebs_dq
- MagnetStrengthCurrent/ebs_hst
- MagnetStrengthCurrent/ebs_sqp
- MagnetStrengthCurrent/ebs_vst
- RingSimulator/ebs
- SrOrbit/ebs

Level 2
- DQModel/dq1
- DQModel/dq2
- MagnetFamily/ebs
- MagnetFamily/ebs_qd
- MagnetFamily/ebs_qf
- SextuCorrModel/sd1ae
- SextuCorrModel/sd1bd
- SextuCorrModel/sf2
- SrctSimu/ebs-simu

Level 3
- EbsCorrectorPS/inj-sh
- EbsCorrectorPS/sd1a
- EbsCorrectorPS/sd1b
- EbsCorrectorPS/sf2
- EbsCorrectorPS/sh1
- EbsCorrectorPS/sh2
- EbsCorrectorPS/sh3
- SimulatedBiltCh/dq1
- SimulatedBiltCh/dq2

Level 4
- HSMMagnetDQS/dq1
- HSMMagnetDQS/dq2
- HSMMagnetDQS/sd1ae
- HSMMagnetDQS/sd1bd
- HSMMagnetDQS/sf2
- HSMMagnetQ0/of1
- HSMMagnetQ0/qd2
- HSMMagnetQ0/qd3
- HSMMagnetQ0/qd5
- HSMMagnetQ0/qf1
- HSMMagnetQ0/qf4ae
- HSMMagnetQ0/qf4bd
- HSMMagnetQ0/qf6
- HSMMagnetQ0/qf8

Level 5
- HSMMagnetQ0/qd3i
- HSMMagnetQ0/qf1i
- HSMMagnetQ0/qf2i

Level 6
- EbsCorr/dq1-h
- EbsCorr/dq2-h
- EbsCorr/sd1a-h
- EbsCorr/sd1a-v
- EbsCorr/sd1b-h
- EbsCorr/sd1b-v
- EbsCorr/sf2-h
- EbsCorr/sf2-v
- EbsCorrector/sh1a-h
- EbsCorrector/sh1a-v
- EbsCorrector/sh1a-x
- EbsCorrector/sh2b-h
- EbsCorrector/sh2b-v
- EbsCorrector/sh3e-h
- EbsCorrector/sh3e-v
- EbsCorrector/sh3e-x
- EbsSqCorr/sd1a-s
- EbsSqCorr/sd1b-s
- EbsSqCorr/sf2-s
- EbsSqCorrector/sh1a-s
- EbsSqCorrector/sh2b-s
- EbsSqCorrector/sh3e-s

Level 7
- CorrectorType/ebs
- DevicesMux/ebs
- Dummy/ebs
- MagnetType/ebs
- SettingsManager/ebs
- SimulatedEbsBpm/ebs
- SimulatedEbsEmit/ebs
- SimulatedEbsMS/ebs
- SimulatedEbsTm/ebs

Level 8
- AllQuadsWrapper/ebs
- Autocor/ebs
- Bump/ebs
- ChromaAdjust/ebs
- ChromaMeasurement/ChromaScan
- ChromaSource/measured
- JDoorWrapper/ebs_ps
- MacroServer/ebs_ps
- ResCor/quad
- ResCor/sext
- ResCor/skew
- TuneAdjust/ebs

ebs-simu-1 (EBS Simulator) Control

54 Controlled Servers on ebs-simu-1

Level 1
- Dummy/ebs
- GenericSimulator/machstat
- RingSimulator/ebs
- SrctSimu/ebs

Level 2
- GenericMultiChannel/corr-c01c16
- GenericMultiChannel/corr-c17c32
- GenericMultiChannel/fofb-correctors
- SimulatedMagnetPS/1

Level 3
- EbsMagnet/dq1
- EbsMagnet/dq2
- EbsMagnet/mini-beta
- EbsMagnet/of1
- EbsMagnet/qd2
- EbsMagnet/qd3
- EbsMagnet/qd5
- EbsMagnet/qf1
- EbsMagnet/qf2
- EbsMagnet/qf4ae
- EbsMagnet/qf4bd
- EbsMagnet/qf6
- EbsMagnet/qf8
- EbsMagnet/sd1ae
- EbsMagnet/sd1bd
- EbsMagnet/sf2
- EbsMagnet/sh1
- EbsMagnet/sh2
- EbsMagnet/sh3

Level 4
- CorrectorType/ebs
- MagnetFamily/ebs
- MagnetFamily/ebs_qd
- MagnetFamily/ebs_qf
- MagnetType/ebs

Level 5
- AllQuadsWrapper/ebs
- CellCyclingTF/ebs_c01c16
- CellCyclingTF/ebs_c17c32
- Cycling/ebs
- JDoorWrapper/ebs_ps
- MacroServer/ebs_ps
- PSGroup/ps

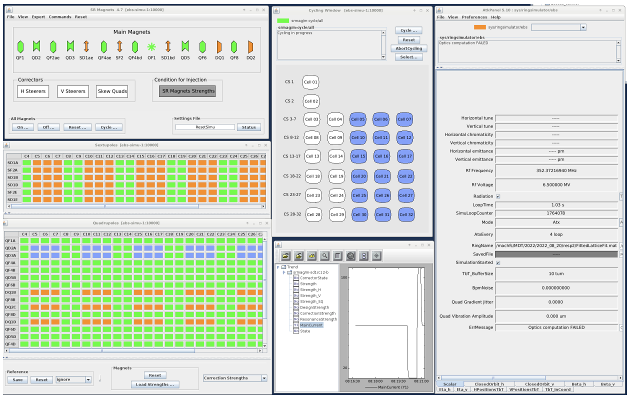
Level 6
- DevicesMux/ebs
- SimulatedEbsBpm/ebs
- SimulatedEbsEmit/ebs
- SimulatedEbsMS/ebs
- SimulatedEbsTm/ebs
- SrOrbit/ebs

Level 7
- Autocor/ebs
- Bump/ebs
- ChromaAdjust/ebs
- ResCor/quad
- ResCor/sext
- ResCor/skew
- SettingsManager/ebs
- StrengthReloader/ebs
- TuneAdjust/ebs

Figure 4: Old(top) and new(bottom) magnets control DS architecture.

order to avoid all possible calculation conflicts. To do so, the sextupole model C++ library [12, 13] based on Eigen linear algebra third party library [14] has been extended to integrate all EBS magnets models within the EbsMagnet Tango class [15] as shown in Fig. 3. This will also allow to speed up the EBS orbit correction by doing horizontal and vertical planes at once which was not possible with the previous control architecture (as shown in Fig. 4). One other main requirement was to have exactly the same code running on both EBSS and EBS in order to ease software maintenance. At the EBSS level, the coil currents computed by the EBSMagnet class are sent to the SimulatedMagnetPS Tango class which recomputes strengths from coil currents and send them back to the PyRingSimulator (pyAT). EBSS brought a significant help in replacing the magnet control step by step during machine dedicated time without introducing new bugs and keep a normal EBS operation. The new magnet control also allows to use Badger [16] for lifetime optimisation during normal operation without orbit correction disturbance. In fact multiple application such as badger and the orbit correction feedback can act on combined function magnets simultaneously without creating control setpoint conflicts.

Hot Swap Manager EBS magnets are powered by individual power supplies. In order to improve the reliability of the system, in the event of a PS failure, the power supplies can be automatically swapped without disturbing the stored beam (Hot Swap Manager [17]). A virtualization layer has been implemented to make fully transparent to the control an automatic swap of a power supply (Hot Swap). The VirtualMagnetPS Tango class [18] is able to manage to integrate standalone power supplies or swapable power supplies and keep the same interface for the magnet control system. For instance, in the EBS injection zone, 2 specific quadrupoles are powered by a standalone power supply and are seen as all other magnets. At the EBSS level, the VirtualMagnetPS class is simply replaced by the SimulatedMagnetPS class.



srmag/m-cycle/all: Cycling in progress

sys/ringsimulator/ebs: Optics computation FAILED

Horizontal tune: -----
Vertical tune: -----
Horizontal chromaticity: -----
Vertical chromaticity: -----
Horizontal emittance: ----- pm
Vertical emittance: ----- pm
Rf Frequency: 352.37216940 MHz
Rf Voltage: 6.500000 MV
Radiation: ☑
LoopTime: 1.03 s
SimuLoopCounter: 1764078
Mode: Atx
AtxEvery: 4 loop
RingName: /machfs/MDT/2022/2022_08_20/resp2/FittedLatticeFit.mat
SavedFile: -----
SimulationStarted: ☑
TbT_BufferSize: 10 turn
BpmNoise: 0.000000000
Quad Gradient Jitter: 0.0000
Quad Vibration Amplitude: 0.000 um
ErrMessage: Optics computation FAILED

Figure 5: Cycling in progress in the EBSS. The blue color indicates moving state. The EBSS DS (right) is in alarm (orange) as the strengths during cycling do not correspond to stable optics. The main magnets control window (top left) and the sextupole and quadrupole control subpanels (bottom left) indicate the specific magnets that are changing their set points. The strength of a selected magnet is monitored (bottom center) over time to show the PS current set point cycle of a selected magnet.

SimulatedMagnetPS Set Point Settlement Time Usually magnets take a finite time to reach their requested set points. This effect is reproduced artificially also in the simulator which makes EBSS more realistic for testing applications and especially the cycling procedure.

Cycling The visualization and test of cycling was not possible in the first version of the EBSS. With the renewal of the EBS magnet control it is now possible to run magnet cycling also in the EBSS. Figure 5 shows an example of cycling in progress. No hysteresis effects are implemented in EBSS, but it is possible to visualize the correct steps in the cycling of each magnet including the de-Gaussing loops individually defined for each magnet family. Power consumption is also considered in the cycling sequence to avoid overload of the common power sources. This has greatly contributed to facilitating the development of a ”per magnet family” cycling procedure which allowed to optimise the EBS cycling by determining de-Gauss mini-cycles on various type of magnets.

Mini-Beta Optics

An other example of use of the EBSS for operation is the set up of *mini-beta* optics [19]. These optics add to the storage ring four additional magnets that are in a first phase not to be visible to the users, but available as DS. These four magnets have been added to the control system and the pyRingSimulator has been also updated consequently. This update had to be reversible, so, the concept of *monitored devices* was included in the pyRingSimulator DS. A list of monitored devices is provided, if those do not exist they are simply ignored. Only if they exist their impact will be considered by the PyRingSimulator DS. This configuration allowed to keep the mini-beta magnets in the list of monitored devices even if the lattice model used by the EBSS was actually missing them. The EBSS allowed to test the implementation of mini-beta optics in terms of beam threading, orbit, tune and linear optics corrections. Local errors were set in the mini-beta magnets and linear optics could restore the usual 1 % beta-beating. Also thanks to the preparatory tests in the EBSS the initial commissioning of mini-beta optics run as expected.

TOWARDS A DIGITAL TWIN

The EBSS infrastructure is a first step towards an online Digital Twin [20] of the Storage Ring. In fact the only device that distinguishes the simulator from the real control system is the pyRingSimulator DS. A complete digital twin continuously monitors the real machine settings and updates the expected output accordingly. The EBSS permits to use a measured lattice to trigger the simulator computations and to set the strengths so that EBSS matches the real machine configuration. An instance of the PyRingSimulator device is then run in the real control system sourcing the currently loaded optics as a model (a trivial action requiring a simple modification of a Tango property) and use the real magnets settings to update the optics model. This allows to have constantly (at the speed of the simulator loop, ~0.5 s) an estimate of the expected orbit, tunes, dispersion and optics of the SR. The absolute comparison being far from easy to obtain with the existing lattice model at the ESRF, relative variations monitoring is possible without major issues. For this purpose a *StrengthsOffset* attribute is added to the pyRingSimulator DS to determine the start conditions for the relative optics variations computations. There is no need to worry about conversion of PS currents to strength as this is done within the new magnets control architecture at the control system level. This also limits to a single device server (the pyRingSimulator DS) the servers needed for the realization of the online EBSS instance. The real strengths set point are sent to this device via a dedicated refresher device server. The EBSS events system is not available in the real control system, thus this additional layer is added with no impact on overall performances.

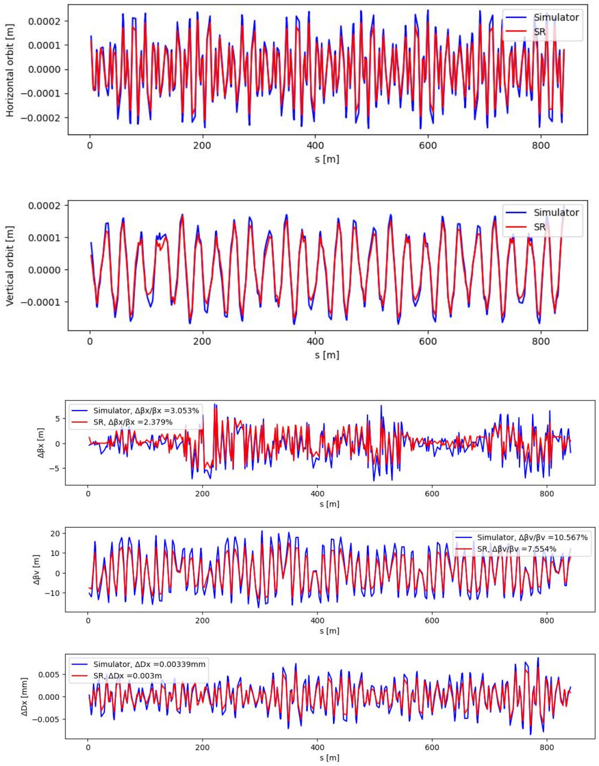


Figure 6: Measured variation of orbit (top), beta functions (center) and horizontal dispersion (bottom) in the SR after a change of quadrupole correction strengths and the same provided by the online-pyRingSimulator instance.

Table 1: Variation of Tunes, Emittances and Chromaticity Set in the ESRF-EBS Storage Ring After a Change of Magnets Settings and the Same Variations as Reported by the Online-pyRingSimulator

	measured SR	**simulated EBSS-online**
$(\Delta Q_h, \Delta Q_v)$	(+0.02,-0.02)	(0.03, -0.03)
$(\Delta\xi_h, \Delta\xi_v)$	(-4.0,+2)	(-4.5,+2.5)
$(\Delta\epsilon_h, \Delta\epsilon_v)$	(+0.0,+11.0)	(0.0,7.0)

Figure 6 and Table 1 show the comparison of measured variations of orbit, tune, chromaticity, emittances, beta functions[1] and dispersion with their simulated counterparts reported in real time by the online pyRingSimulator DS instance. The changes observed in the online-pyRingSimulator DS (digital twin, DT) are relative to a given initial configuration of the storage ring and give continuously information on the actual state of the optics. Large deviations would indicate large (potentially unexpected) changes of the magnets setpoints. Deviation from the nominal beta functions, dispersion, orbit, tunes, chromaticity and emittances may then be monitored by operators and beam physics experts. The discrepancies among the measured and simulated variations in Fig. 6 and Table 1 are accountable in part to the measurement errors and in part to the lattice model. The agreement is in general more than acceptable, although there is clearly

---

[1] fitted from a response matrix measurement

margin for improvement. For example future actions will try to provide a more detailed and complete linear and non linear optics model.

Future Improvements of the Online EBSS Digital Twin

Several updates to the EBSS and to its online instance could be envisaged:

- Presently the simulator tracks only magnets and RF variations. A further improvement could be to include Insertion Device (ID) gaps models in the optics model, and link them to the real ID gaps values available in the control system. Such update would require minor changes to the EBSS device server, such as the list of monitored devices and the lattice optics model. However, presently the pyAT EBS optics model does not include ID gaps.
- An other possible use of the EBSS could be to estimate optics properties based on archived magnets strengths. This option could be implemented with a modified refresher DS, looking at a given time in HDB rather than at the present magnets strengths configurations.
- The online EBSS could be updated based on the PS currents rather than with the strengths available in the control system.
- Connecting the digital twin to the beamline control system simulator is another area to be explored. This would allow beamlines to test developments with a simulated copy of the accelerator i.e. the EBSS.
- The realization of a digital twin also for the ESRF energy ramping synchrotron booster is being considered. For this case the introduction of an energy set point to the pyRingSimulator to describe the different energies along the booster ramp would be needed along with the definition of simulated diagnostics and simulated ramped magnets.

PORTING EBSS

There are a number of new projects started or planned to replace the current 3rd generation storage rings with 4th generation ones. For those projects which are using Tango for the accelerator controls or are interested in trying it out with a simulator, the EBSS offers a powerful solution.

To port the EBSS to another storage ring the following would need to be carried out:

- Define the devices according to the naming scheme of the new storage ring
- Implement simulation devices for each of the device classes. One option is to start off with a generic device simulator e.g. using the tango-simlib [21]
- Populate the Tango database with the simulated devices
- Define the optics of the new storage ring for pyAT

This was done in a first test for the Elettra-II storage ring [22] and shows that some progress is needed to make the code simpler to use for other storage rings, in particular concerning the pyRingSimulator DS.

With the updates introduced by the online-pyRingSimulator the amount of work required to port the EBSS to a new storage ring has been reduced compared to the work done for Elettra-II.

CONCLUSIONS

The EBSS proved to be an extremely powerful tool for the preparation of the ESRF-EBS storage ring commissioning. It allowed to eradicate most of the coarse bugs and to include several additional features without any delay on the initial schedule. The EBSS was used with large profit also after the EBS storage ring commissioning and is presently used for the continuous development and upgrade of all the softwares for which a direct impact on the electron beam is expected.

Thanks to the online-EBSS, the expected (simulated) beam properties from the storage ring, such as optics and response to magnetic fields variations may be continuously monitored in control room. This gives access to observables that are not directly available from diagnostics during user operations such as beta functions and dispersion. This recent development is also necessary to implement artificial intelligence and machine learning algorithm that will constantly compare model vs measurement enabling: 1) early detection of faults, 2) slow drifts of parameters potentially linked to hardware failures, 3) continuous lattice optics model update, and many other features.

The extension to other accelerators is presently not trivial, due to the intrinsic peculiarity of each control system. Nevertheless the strategy is set and at least the PyRingSimulator DS would require few modifications (the specific name of the devices) to be used for other storage rings, or high energy electron transfer lines. Specific developments will be needed instead to extend the use of the simulator to linacs, energy ramped accelerators and hadron accelerators.

The several aspects mentioned above will be addressed in the near future and are presently part of a proposal for a future EU collaboration project.

REFERENCES

[1] P. Raimondi *et al.*, “The extremely brilliant source storage ring of the european synchrotron radiation facility”, *Commun. Phys.*, vol. 6, no. 1, p. 82, 2023. `doi:10.1038/s42005-023-01195-z`

[2] S. Liuzzo *et al.*, “Preparation of the ebs beam commissioning”, *J. Phys.: Conf. Ser.*, vol. 1350, p. 012 022, 2019. `doi:10.1088/1742-6596/1350/1/012022`

[3] S. M. Liuzzo *et al.*, “The ESRF-EBS Simulator: A Commissioning Booster”, in *Proc. ICALEPCS’21*, Shanghai, China, 2022, pp. 132–137. `doi:10.18429/JACoW-ICALEPCS2021-MOPV012`

[4] S. White *et al.*, “Status and recent developments of python accelerator toolbox”, in *Proc. IPAC’23*, Venice, Italy, 2023, pp. 3185–3188. `doi:10.18429/JACoW-IPAC2023-WEPL031`

[5] L. R. Carver *et al.*, "Beam based characterization of the European Synchrotron Radiation Facility Extremely Brilliant Source short range wakefield model", *Phys. Rev. Accel. Beams*, vol. 26, no. 4, p. 044 402, 2023. `doi:10.1103/PhysRevAccelBeams.26.044402`

[6] A. Gotz *et al.*, "State of the Tango Controls Kernel Development in 2019", in *Proc. ICALEPCS'19*, New York, NY, USA, 2020, p. 1235. `doi:10.18429/JACoW-ICALEPCS2019-WEPHA058`

[7] G. Portmann, J. Corbett, and A. Terebilo, "An accelerator control middle layer using matlab", in *Proc. PAC 2005*, 2005, pp. 4009–4011. `doi:10.1109/PAC.2005.1591699`

[8] R. Kammering *et al.*, "The Virtual European XFEL Accelerator", in *Proc. ICALEPCS'15*, Melbourne, Australia, Oct. 2015, pp. 578–580. `doi:10.18429/JACoW-ICALEPCS2015-TUD3O04`

[9] Ebs simulator tango device servers, `https://gitlab.esrf.fr/accelerators/Simulators/EbsSimulator`

[10] Python ring simulator optics computation, `https://gitlab.esrf.fr/BeamDynamics/pythontools/optics_for_pyringsimulator`

[11] Corba transient error description, `https://courses.cs.washington.edu/courses/cse341/98au/java/jdk1.2beta4/docs/api/org/omg/CORBA/TRANSIENT.html`

[12] G. L. Bec, "A Fast Non-Linear Model for the EBS Combined Sextupole-Corrector Magnets", in *Proc. IPAC'21*, Campinas, Brazil, May 2021, pp. 2381–2383. `doi:10.18429/JACoW-IPAC2021-TUPAB369`

[13] Ebs magnet device servers, `https://gitlab.esrf.fr/accelerators/Magnets/MagnetModel`

[14] Eigen library, `https://gitlab.com/libeigen/eigen`

[15] Ebs magnet device server, `https://gitlab.esrf.fr/accelerators/Magnets/EbsMagnet`

[16] S. M. Liuzzo *et al.*, "Optimisation of the Touschek Lifetime in Synchrotron Light Sources Using Badger", presented at ICALEPCS'23, Cape Town, South Africa, October 2023, paper MO3AO01, this conference.

[17] Reliability of ebs power supplies and hotswap redundancy system, `https://indico.cern.ch/event/1222337/timetable/#38-reliability-of-ebs-power-su`

[18] Virtualization layer of magnet power supplies, `https://gitlab.esrf.fr/accelerators/Magnets/VirtualMagnetPS`

[19] S. White *et al.*, "Mini-beta optics for the european synchrotron radiation facility", in *Proc. IPAC'23*, Venice, Italy, 2023, pp. 3177–3180. `doi:10.18429/JACoW-IPAC2023-WEPL029`

[20] A. Fuller, Z. Fan, C. Day, and C. Barlow, "Digital twin: Enabling technologies, challenges and open research", *IEEE Access*, vol. 8, pp. 108 952–108 971, 2020. `doi:10.1109/access.2020.2998358`

[21] Tango-simlib: Easily generate tango device simulators, `https://github.com/ska-sa/tango-simlib`

[22] C. Scafuri, S. Krecic, J.-L. Pons, S. Liuzzo, *et al.*, "A digital twin for elettra 2.0 commissioning", to be published.