Pyapas: A pure python framework for High-Level Application development

Xiaohan Lu

2025/10/30

Content

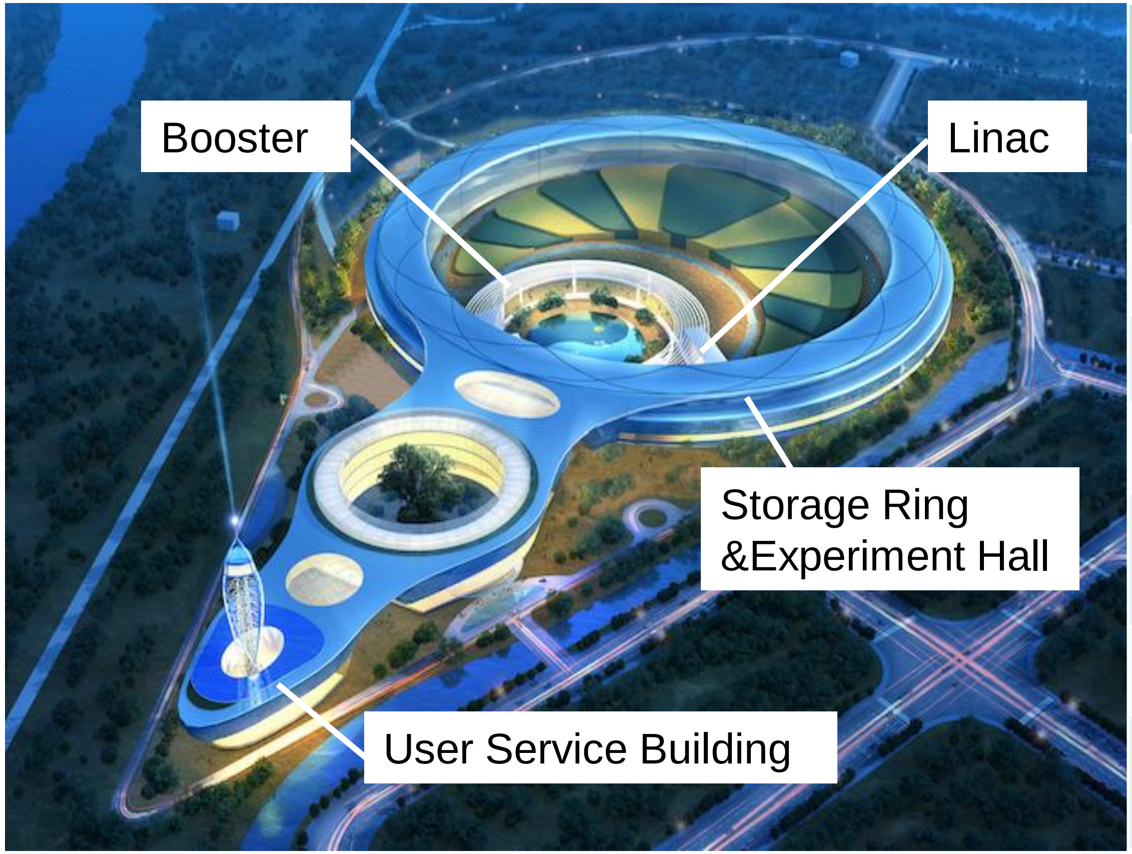
- HLA development requirements
- *Pyapas* for HLA development
- HLAs development for HEPS based on *Pyapas*
- Conclusion

- HLA development requirements
- *Pyapas* for HLA development
- HLA development based on *Pyapas*
- Conclusion

Introduction to HEPS

High Energy Photon Source

One of the world's brightest fourth-generation synchrotron light sources



	APS-U	ESRF-EBS	MAX-IV	Sirius	HEPS
E(GeV)	6	6	3	3	6
C(m)	1104	844.4	528	518	1360.4
Lattice	7BA	7BA	7BA	5BA	7BA
Cell	40	32	20	20	48
Emittance (pm*rad)	42	150	330	250	34
Brightness	$> 10^{22}$	$> 10^{22}$	$\sim 10^{21}$	$\sim 10^{21}$	$> 10^{22}$
Construction period	2018-2024	2015-2020	2010-2016	2015-2018	2019~2025

Yesterday (10.29) HEPS has successfully passed the process qualification and acceptance

Introduction to HEPS

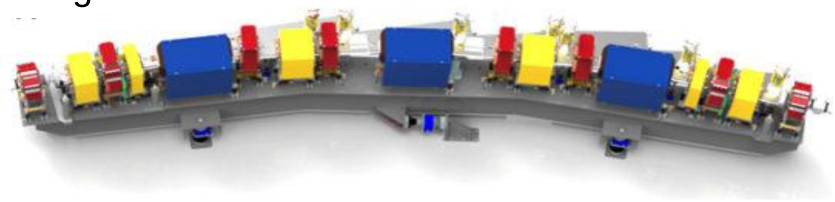
→**The emittance is reduced by 1or 2 orders of magnitude, approaching the X-ray diffraction limit**

→**The number of magnets increases by an order of magnitude**

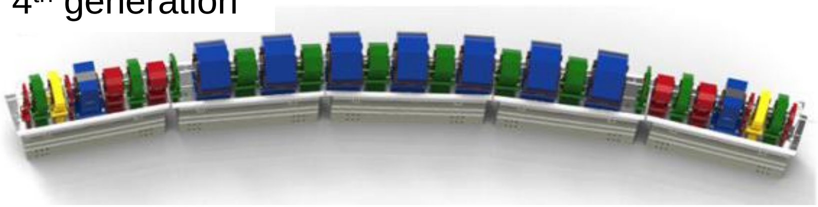
HEPS is currently the largest accelerator facility in China.

- High control complexity
- Large scale of control variables
- High precision control required
- Large number of HLAs required

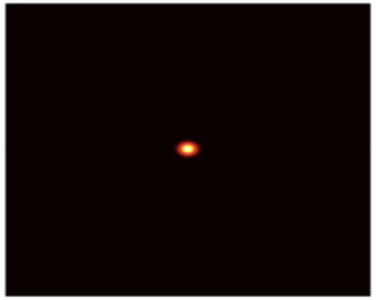
3rd generation



4th generation

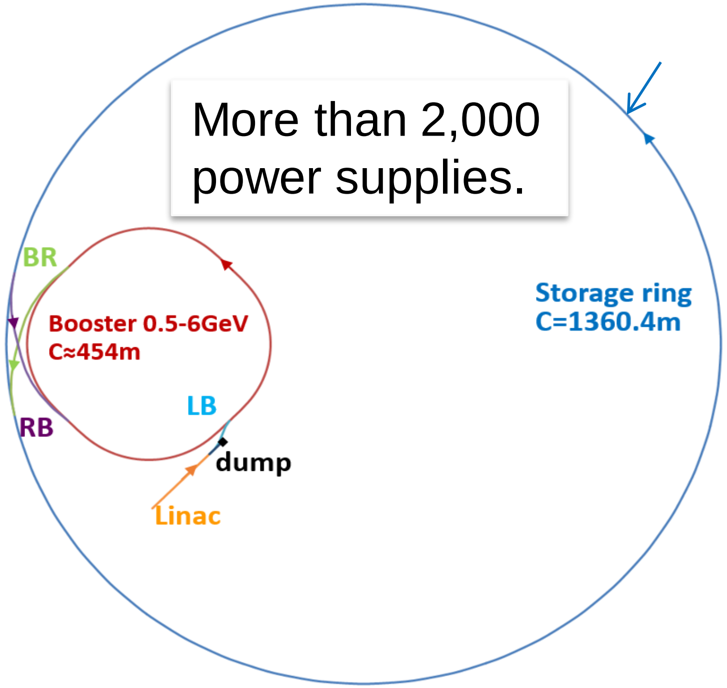


Layout



1 mm

Beam Profile

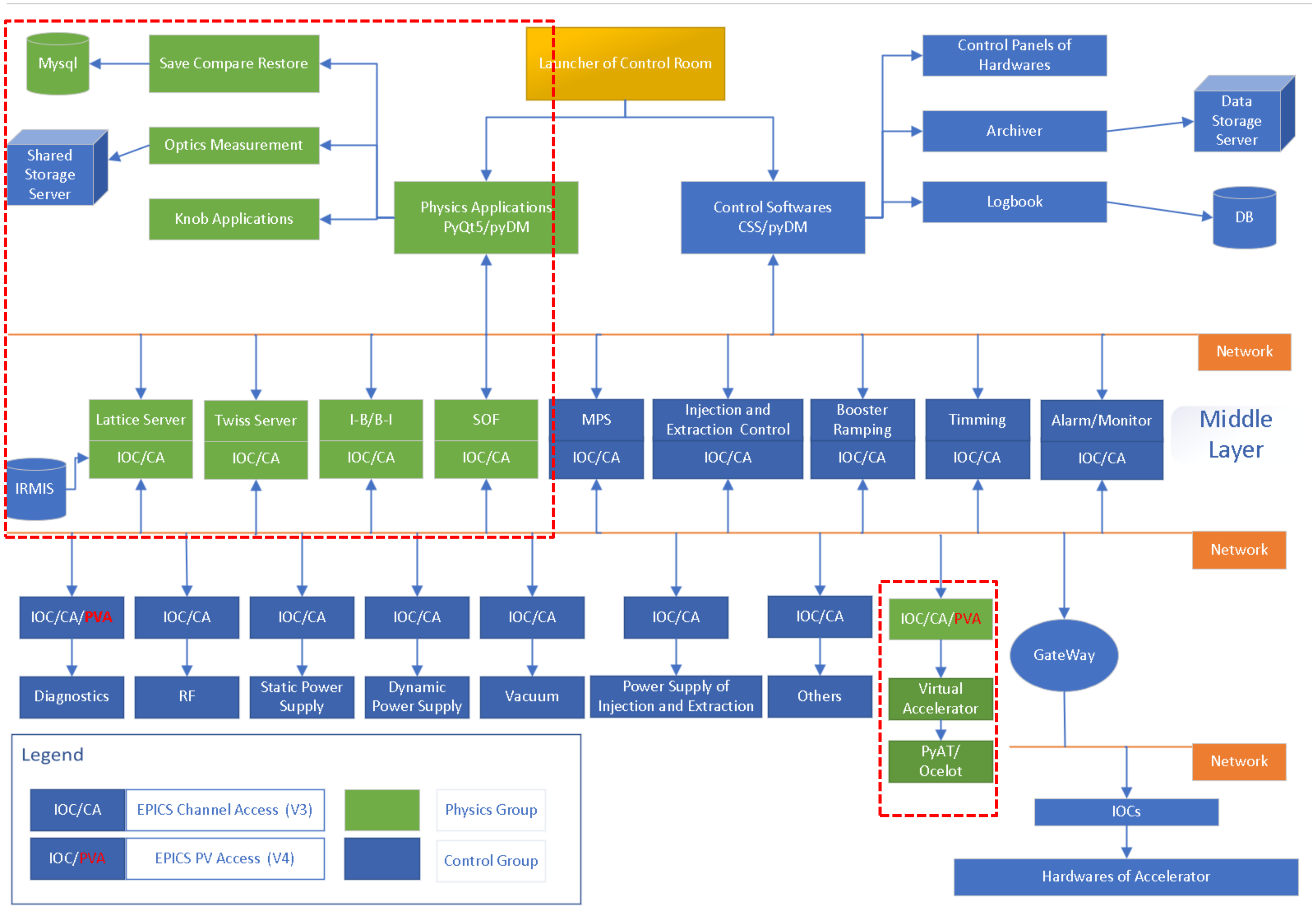


Overall control structure of HEPS

High level physical application focus more on physical process: optimization, simulation, feedback et.al.

High Level Application
(HLA)

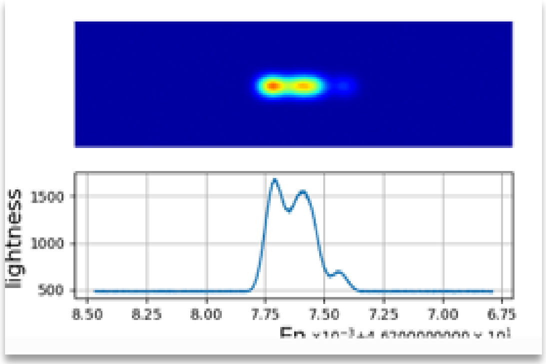
- High control complexity
- Large scale of control variables
- High precision control required
- Large number of HLAs required



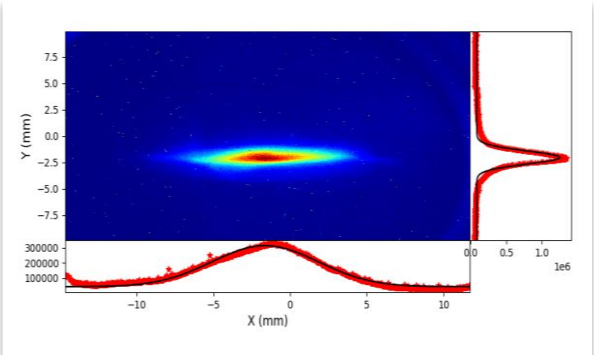
High Level Application

- **High-Level Application** (HLA) is typically an integrated system that **monitors, analyzes, and controls** the accelerator's operational status and beam parameters — serving as the accelerator's **"central nervous system"**!

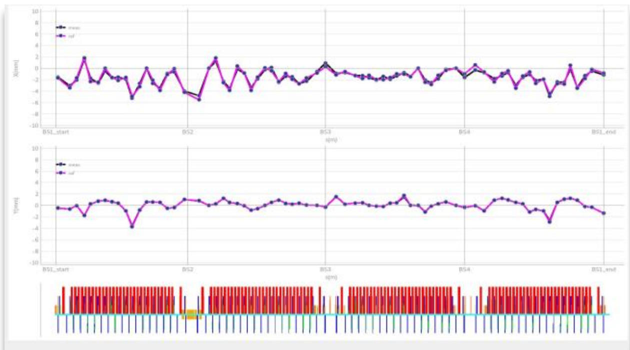
Energy / Energy Spread Analysis



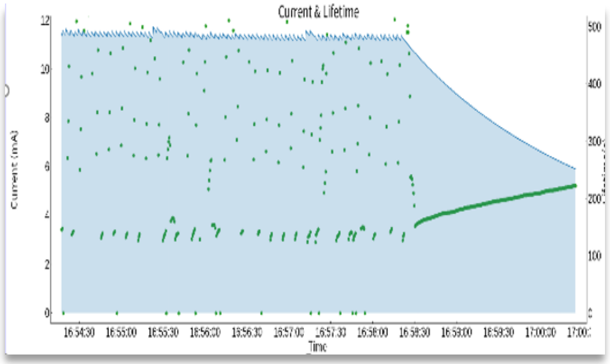
Beam Emittance Measurement



Closed Orbit Correction



Lifetime calculation



Idea → Algorithm Abstraction → Software implementation → Hardware and Beam Parameters → Physics Model Calculation → Human–Machine Interface → High Availability and Long-Term Operation → Data Management and Archiving

- Hardware and Beam Parameters: Real-time control and device communication
- Physics Model Calculation: Integration of physical models and online computation
- Human–Machine Interface: Development of visual, interactive interfaces
- High Availability and Long-Term Operation: server deployment and reliable runtime
- Data Management and Archiving: Establishment of connected data storage systems

HLA Framework:

Providing a Systematic Solution

- HLA development requirements
- *Pyapas* for HLA development
- HLA development based on *Pyapas*
- Conclusion

New HLA framework: *Pyapas*

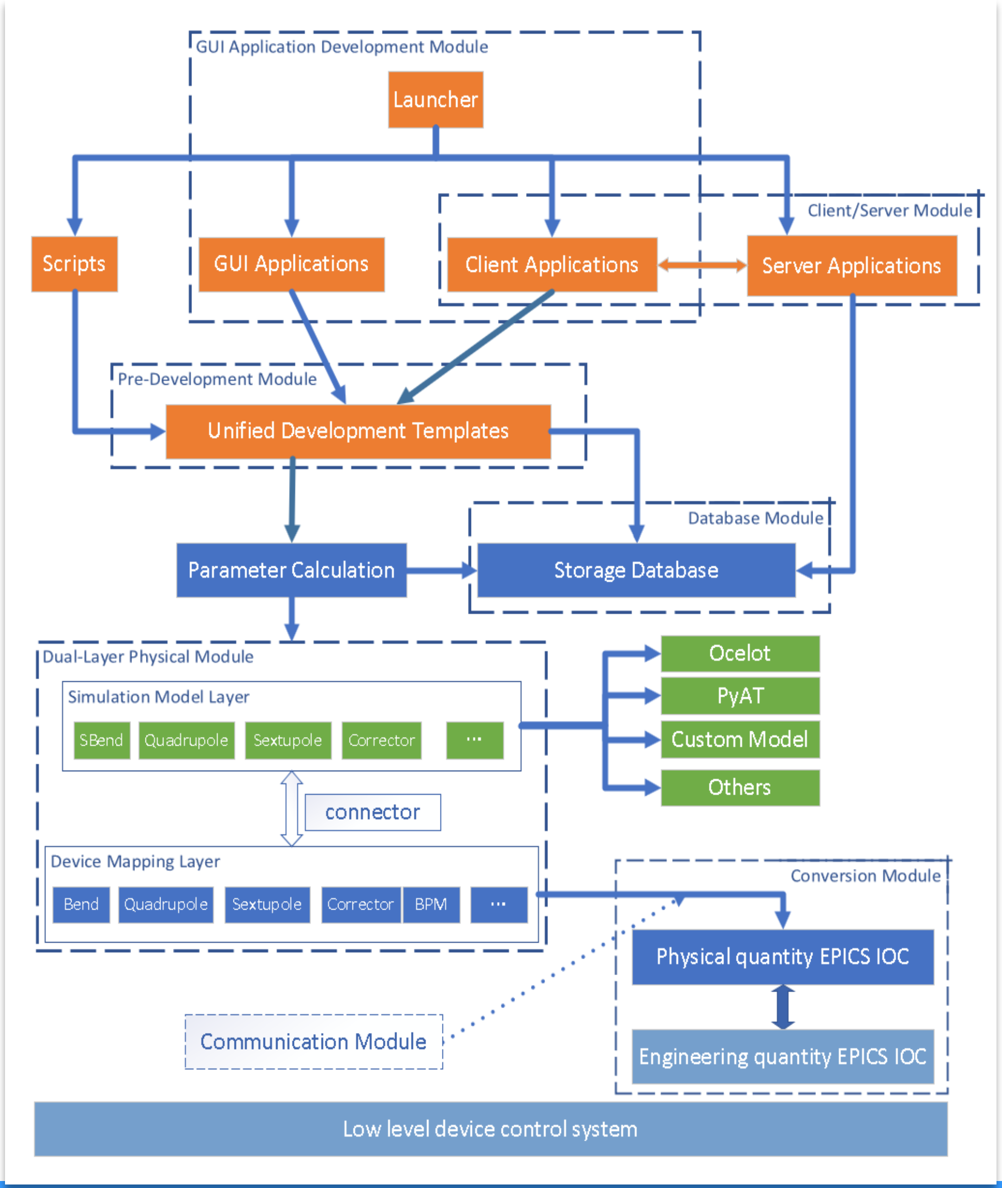
- A Pure Python Framework for High Level Application Development.

  Python-based Accelerator Physics Application Set (*Pyapas*)

  - **Modular Design**
  - **Model-Based**
  - **Based on Physical Quantities**

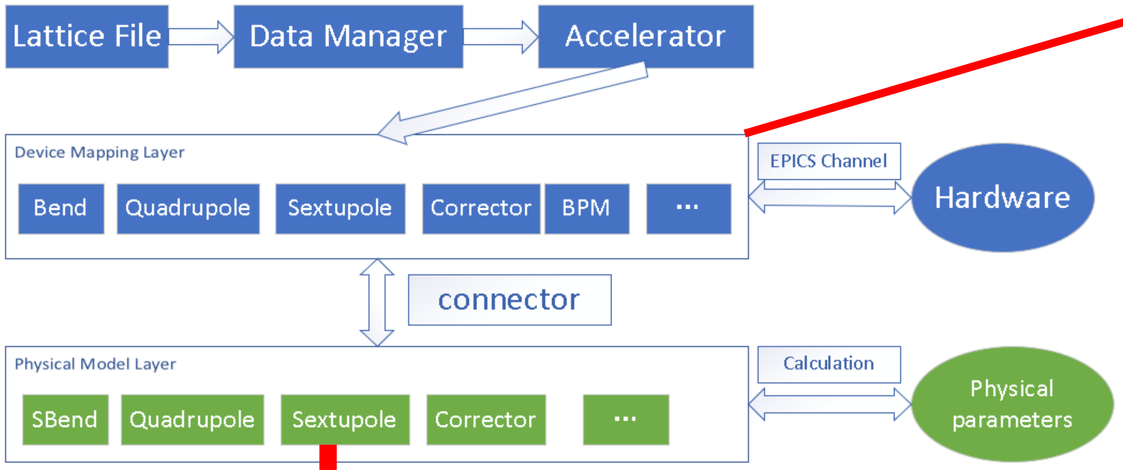
① Dual-layer physical module
② Friendly user interface module
③ Hardware communication module
④ Database connection module
⑤ Client-Server development module
⑥ Various phyiscal algorithm toolkits

Pyapas framework



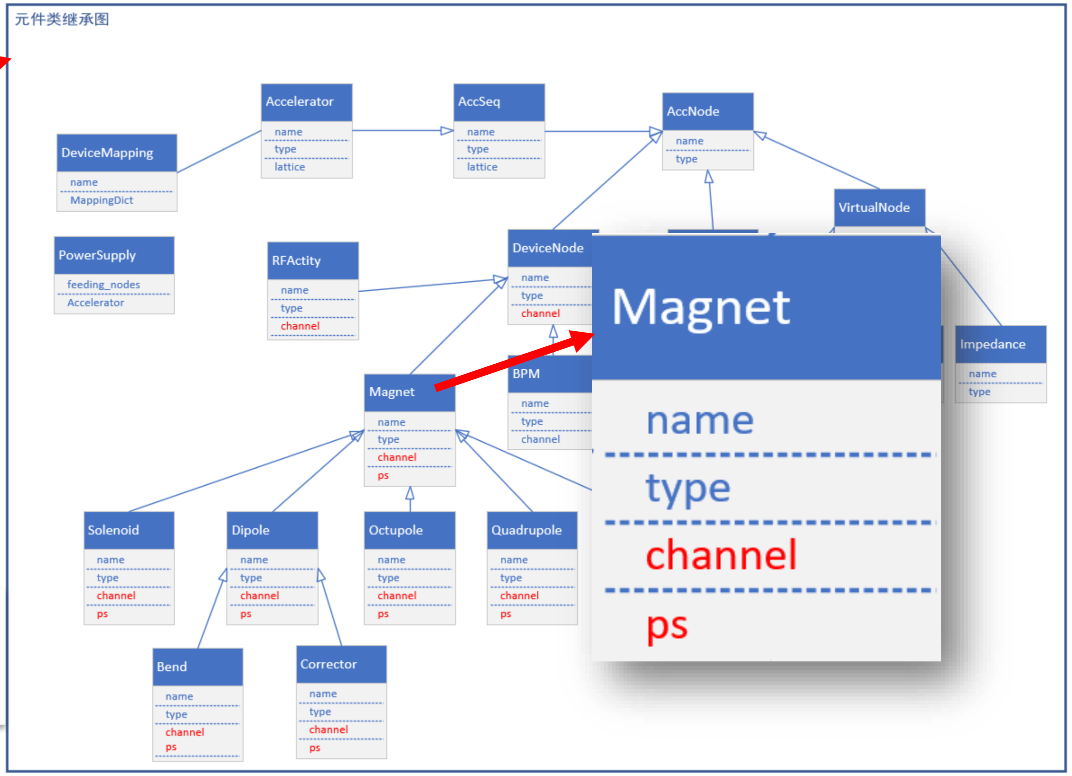
Xiaohan Lu, et al, “A new modular framework for high-level application development at HEPS”, Journal of Synchrotron Radiation, 31: 385-393, 2024.

Low-coupling dual-layer physical module



- Ocelot
- PyAT
- Custom model
- RF-Track?
- APES?

The "Connector" is responsible for invoking simulation algorithms



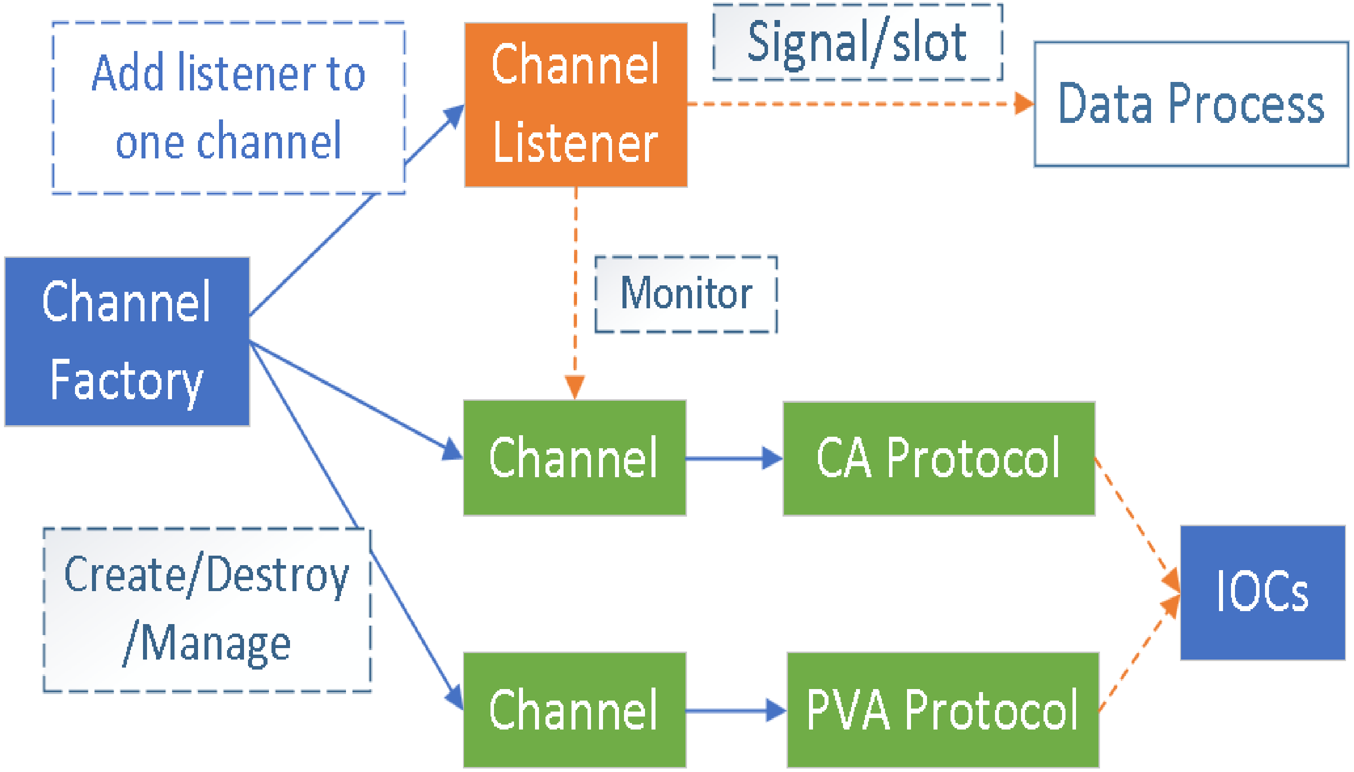
The component class maps to the actual machine components

Not dependent on specific physical models, easy to extend

Physical model ⟷ Real machine

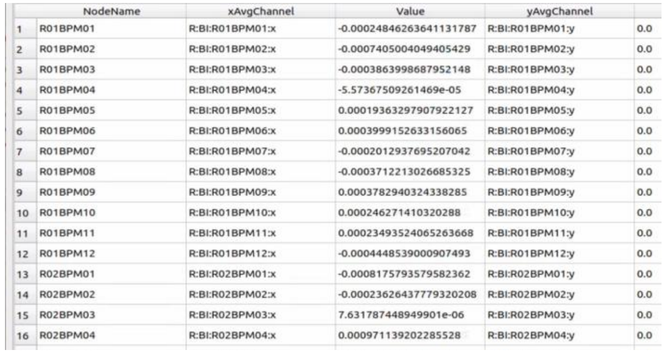
Real-time Data Interaction System

- Design the communication structure (active and passive channel data acquisition)



- ✓ Combining PyQt's signal-slot mechanism to achieve efficient real-time interaction
- ✓ The 'Singleton Factory Pattern' centrally manages interaction channels in the program, avoiding redundant creation and improving interaction efficiency."
- ✓ The channel factory has namespace functionality

The Channel class contains the EPICS communication interface responsible for specific communication with the IOC



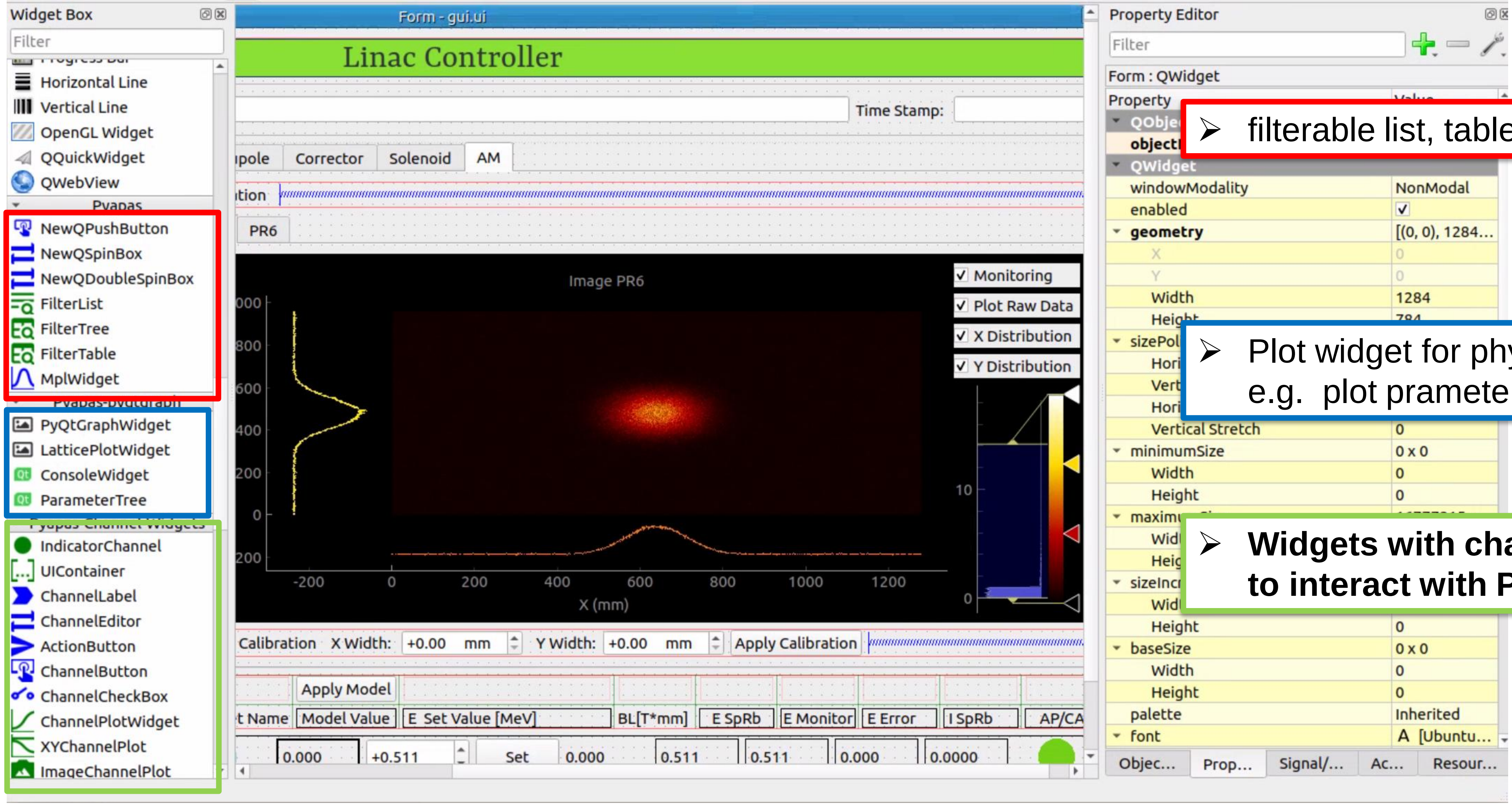
	NodeName	xAvgChannel	Value	yAvgChannel	
1	R01BPM01	R:BI:R01BPM01:x	-0.00024846263641131787	R:BI:R01BPM01:y	0.0
2	R01BPM02	R:BI:R01BPM02:x	-0.0007405004049405429	R:BI:R01BPM02:y	0.0
3	R01BPM03	R:BI:R01BPM03:x	-0.0003863998687952148	R:BI:R01BPM03:y	0.0
4	R01BPM04	R:BI:R01BPM04:x	-5.57367509261469e-05	R:BI:R01BPM04:y	0.0
5	R01BPM05	R:BI:R01BPM05:x	0.00019363297907922127	R:BI:R01BPM05:y	0.0
6	R01BPM06	R:BI:R01BPM06:x	0.0003999152633156065	R:BI:R01BPM06:y	0.0
7	R01BPM07	R:BI:R01BPM07:x	-0.0002012937695207042	R:BI:R01BPM07:y	0.0
8	R01BPM08	R:BI:R01BPM08:x	-0.0003712213026685325	R:BI:R01BPM08:y	0.0
9	R01BPM09	R:BI:R01BPM09:x	0.0003782940324338285	R:BI:R01BPM09:y	0.0
10	R01BPM10	R:BI:R01BPM10:x	0.000246271410320288	R:BI:R01BPM10:y	0.0
11	R01BPM11	R:BI:R01BPM11:x	0.00023493524065263668	R:BI:R01BPM11:y	0.0
12	R01BPM12	R:BI:R01BPM12:x	-0.0004448539000907493	R:BI:R01BPM12:y	0.0
13	R02BPM01	R:BI:R02BPM01:x	-0.0008175793579582362	R:BI:R02BPM01:y	0.0
14	R02BPM02	R:BI:R02BPM02:x	-0.00023626437779320208	R:BI:R02BPM02:y	0.0
15	R02BPM03	R:BI:R02BPM03:x	7.631787448949901e-06	R:BI:R02BPM03:y	0.0
16	R02BPM04	R:BI:R02BPM04:x	0.000971139202285528	R:BI:R02BPM04:y	0.0

The same application can smoothly run while monitoring 10,000 PVs

Low-Code development

Extend QtDesigner ⇒ Customize widgets for HLA development

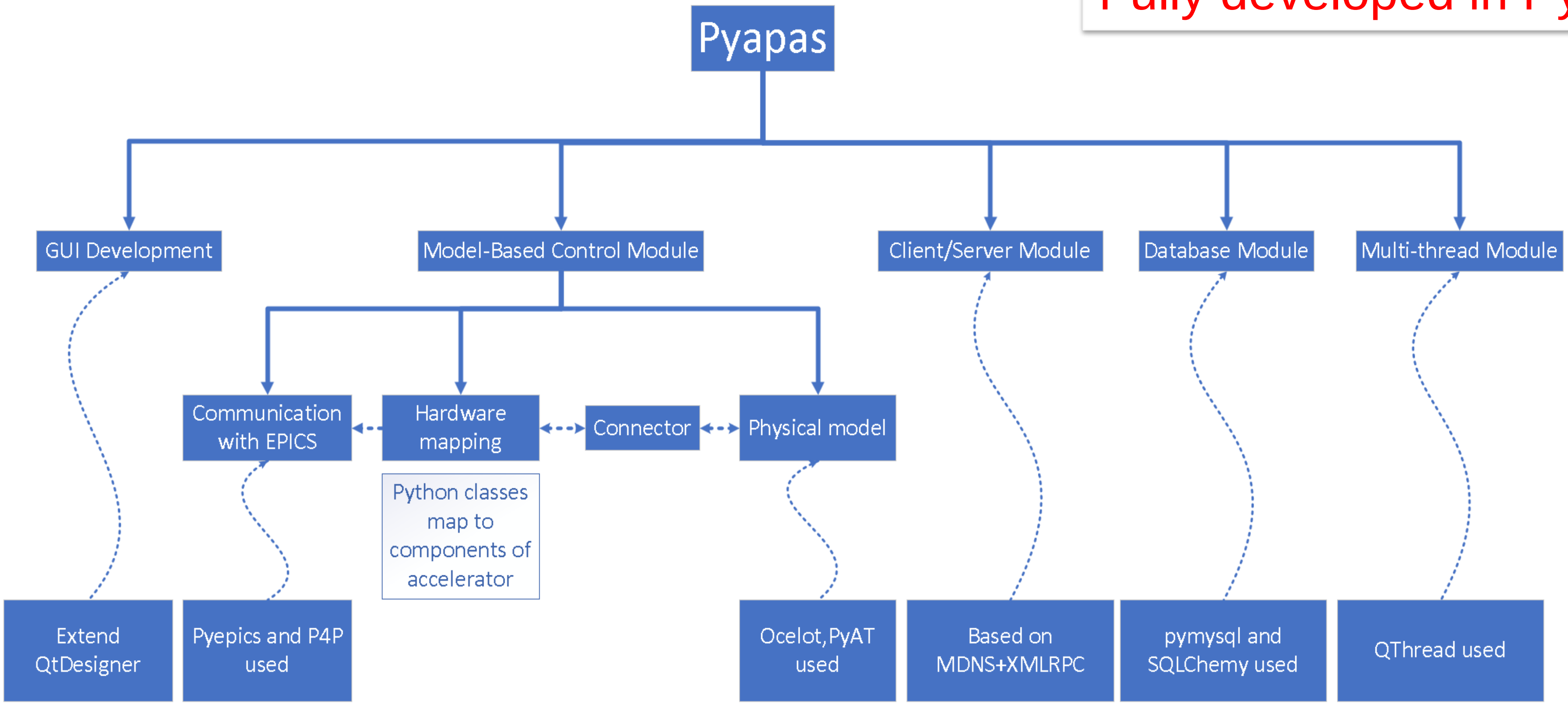
“Drag and Drop”



- ➢ filterable list, table, tree
- ➢ Plot widget for physical image, e.g. plot prameters with lattice
- ➢ **Widgets with channel property to interact with PV directly.**

The main modules of Pyapas are essentially complete

Fully developed in Python



- ✓ **User-friendly**
- ✓ **Efficient**
- ✓ **Stable**

The modules included in Pyapas basically meet all the requirements of HLA development for HEPS

- HLA development requirements
- *Pyapas* for HLA development
- HLA development based on *Pyapas*
- Conclusion

CPM of HLA Development for HEPS

2022 | 2023

Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun

Linac

Transfer Line

Booster

Storage Ring

Linac

- Linac Virtual Accelerator✓
- Transimission Efficiency✓
- Emittance Measurement✓
- Orbit Correction✓
- Phase Scan✓
- Energy Spread Measurement✓
- Beam-based Alignment✓
- Linac Control✓

Transfer Line

- LB Virtual Accelerator✓
- Emittance Measurement✓
- Orbit Correction✓
- Beam-based Alignment✓
- Local Orbit Correction ✓

Booster

- Lattice Preparation✓
- Booster Virtual Accelerator✓
- First Turn Measurement✓
- Dispersion Measurement ✓
- Chromaticity Measurement✓
- Tune adjustor✓
- Global Orbit Correction✓
- Local Orbit Correction✓
- Beam Lifetime✓

Storage Ring

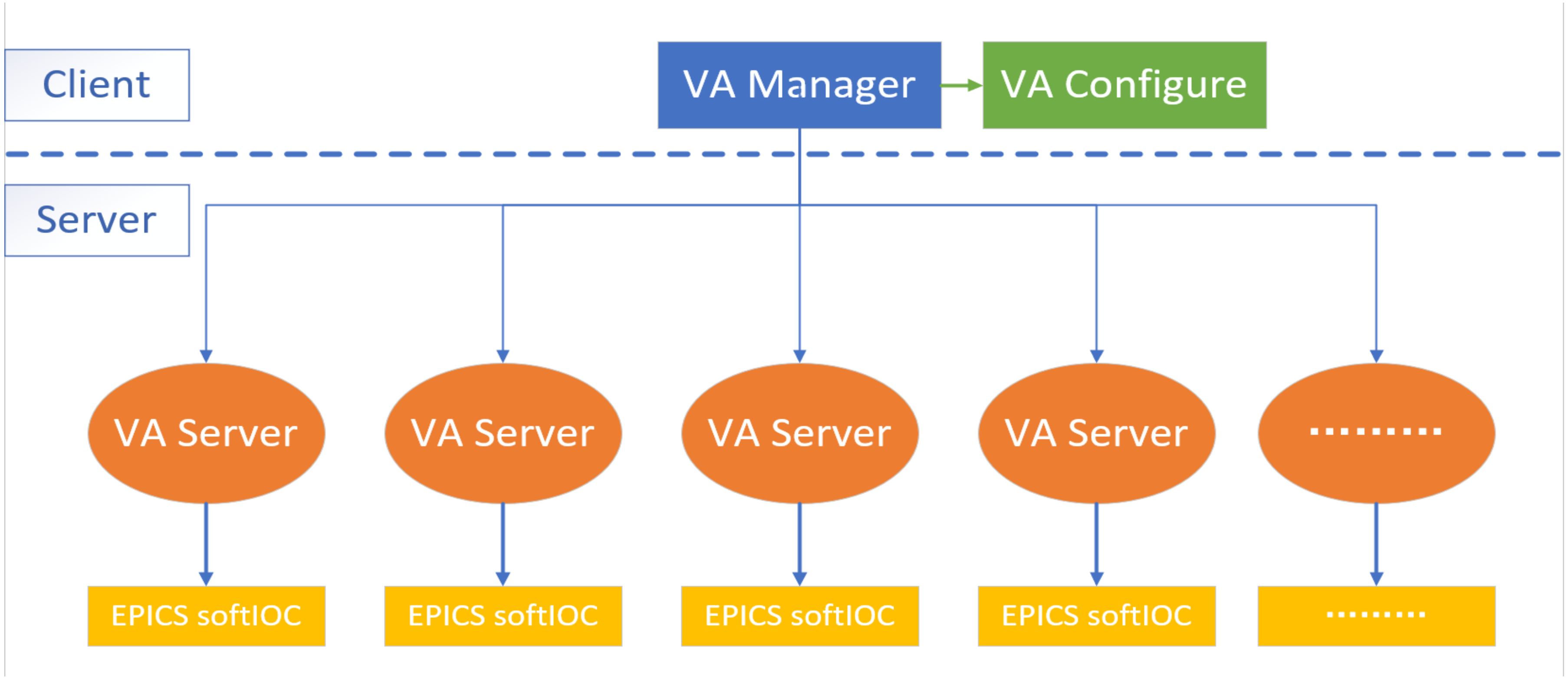
- Lattice Preparation✓
- Booster Virtual Accelerator✓
- First Turn Measurement✓
- Dispersion Measurement ✓
- Chromaticity Measurement✓
- Global Orbit Correction✓
- Local Orbit Correction✓
- Beam Lifetime ✓
- Orbit Display ✓
- Slow Orbit Feedback ✓
- K-Value adjustor ✓
- Fast BBA ✓

All the HLAs are developed based on *Pyapas*

Design of A Multi-User Virtual Accelerator

- ➢ Create virtual hardware to simulate real machines, providing corresponding PV channels and beam information

C/S Architecture

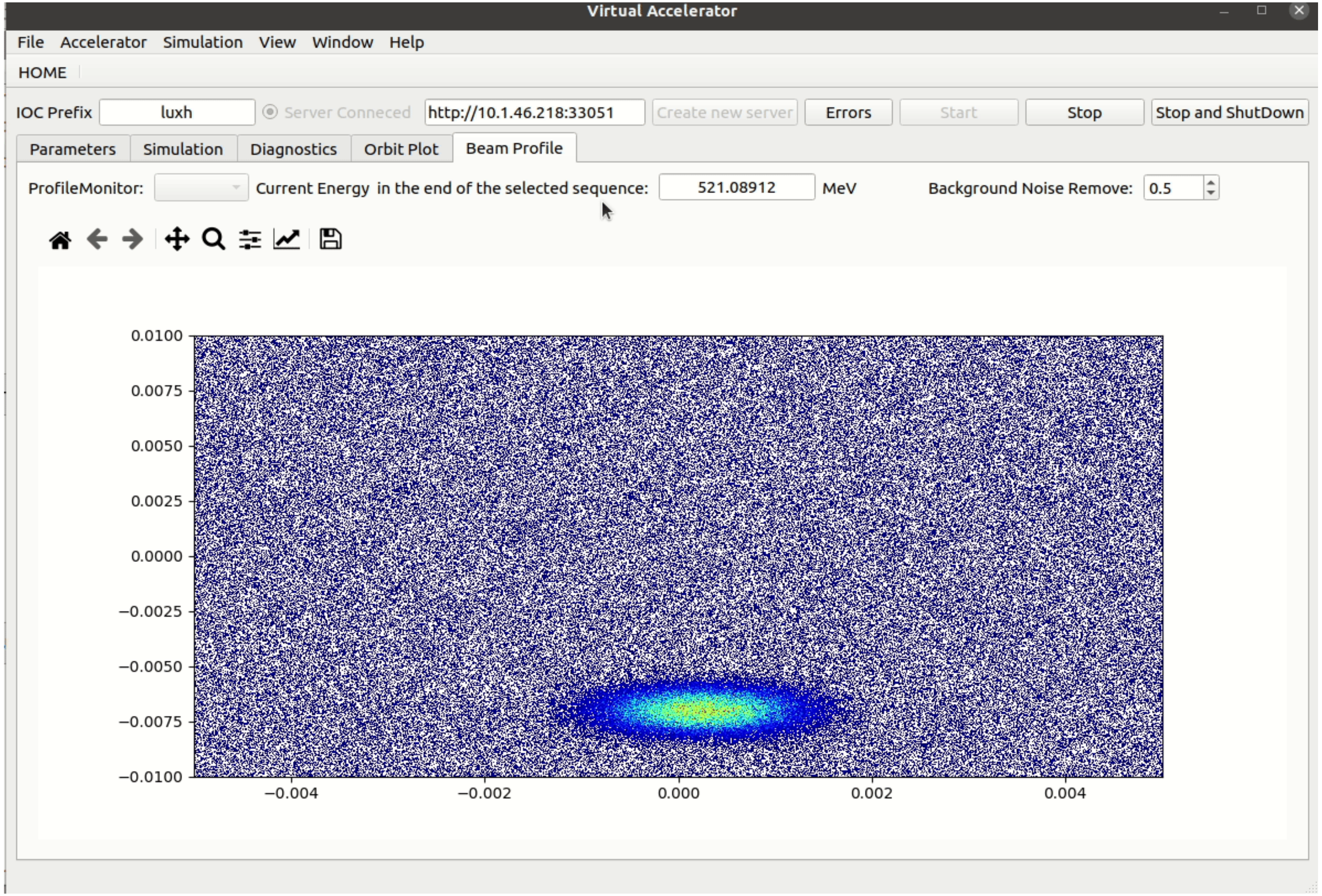


- ✓ Multiple tasks running concurrently
- ✓ Multiple users operating simultaneously
- ✓ Unified management with one-click start and stop

Within the same network environment or on the same server, any number of virtual accelerators can be launched without conflicting with each other

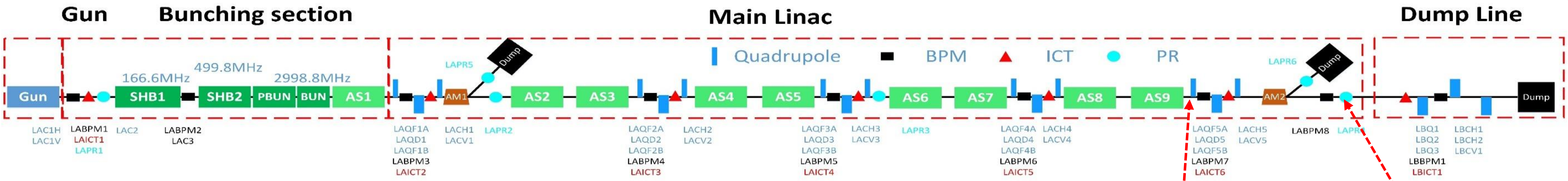
Virtual Accelerator of HEPS Linac

- Single Particle Tracking
- Multi-Particle Tracking
- Vritual PR
- Full feature IOC
- Errors included



Generate 'real-time' data for testing HLAs

Example of a Linac Emittance Measurement Application



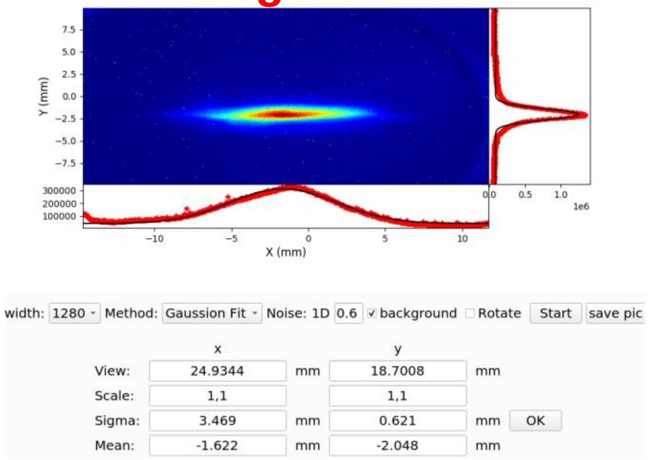
Measurement point

Profile Monitor

$$\begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{pmatrix} = \begin{pmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ \vdots & \vdots & \vdots \\ R_{n1} & R_{n2} & R_{n3} \end{pmatrix} \begin{pmatrix} \beta\varepsilon \\ \alpha\varepsilon \\ \gamma\varepsilon \end{pmatrix}_0$$

$$R_{n1} = M[1,1] * M[1,1]|_{step=n}$$
$$R_{n2} = -2 * M[1,1] * M[1,2]|_{step=n}$$
$$R_{n3} = M[1,2] * M[1,2]M[1,2] * M[1,2]|_{step=n}$$

Fitting online

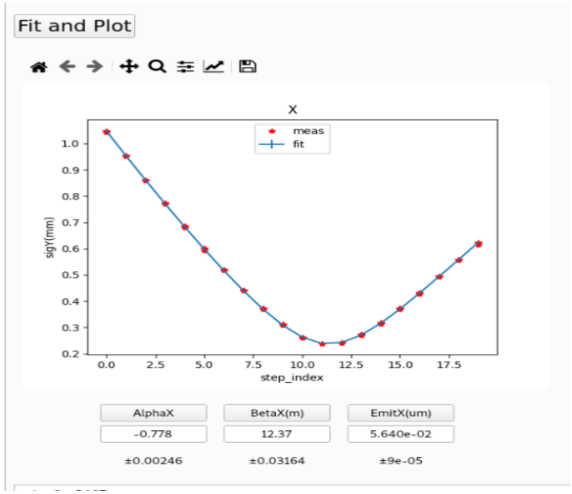


Real-time online calculation

```
calculator = TwissCalculator()
init_parameter = TwissParam()
accelerator_manager = AcceleratorManager(lattice_path)
accelerator = Accelerator_manager.get_accelerator()
sequence = Accelerator.get_sequence('Linac')
load_init_params(sequence, init_parameter)
sequence.sync()  # get the live value from real machine

# call physical model
physics_cell = write_cell(sequence, 'ocelot')
results = calculator.calc(physics_cell, init_parameter)
```

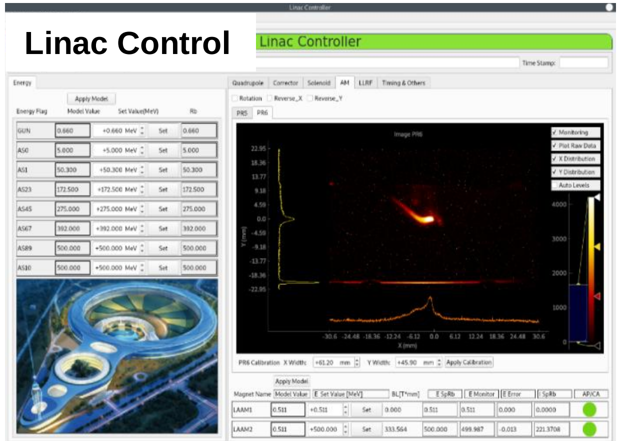
Measurement Result

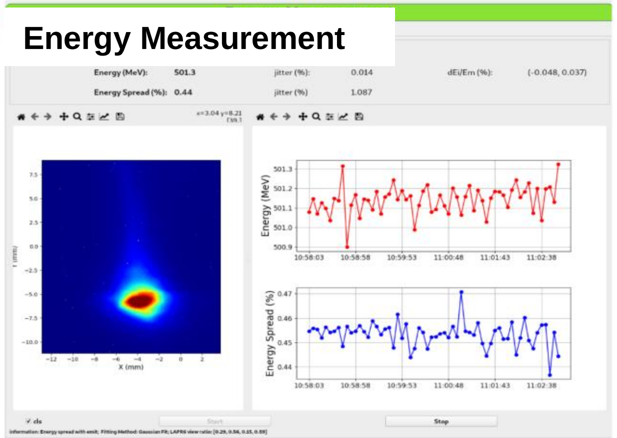


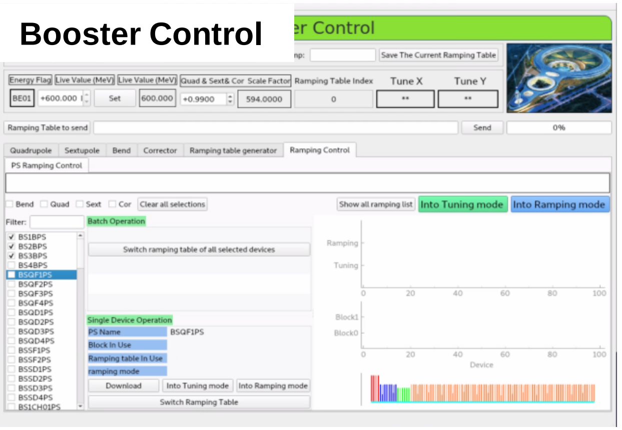
•Using pyapas, implement online measurement applications with **very little code**, increasing development efficiency

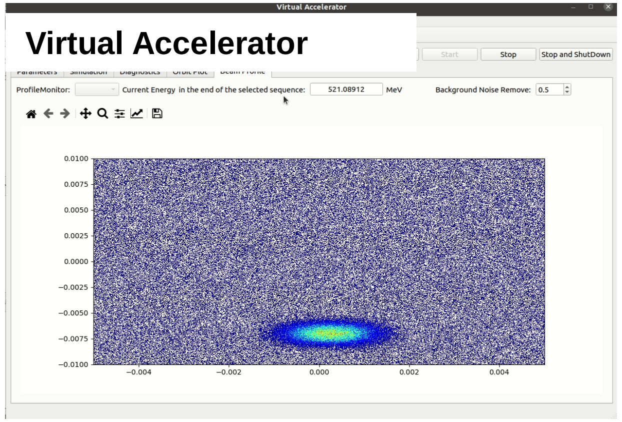
Pyapas-based HLAs for HEPS

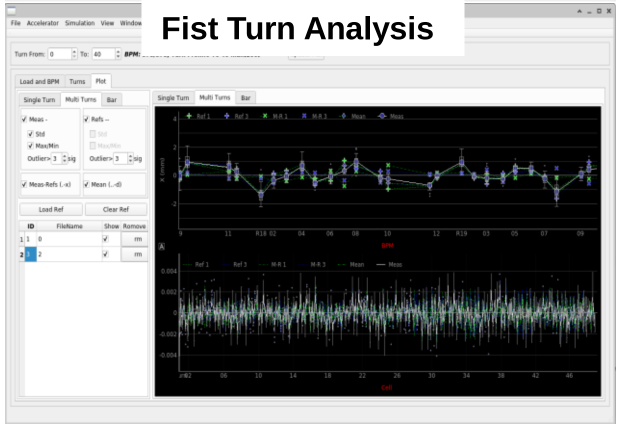
- All the HLAs are developed based on Pyapas
- A Virtual Accelerator (VA) was developed based on Pyapas, and all the HLAs will be tested on the VA before deployment

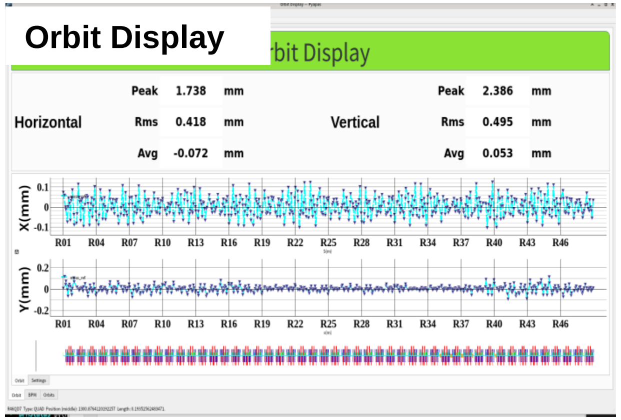


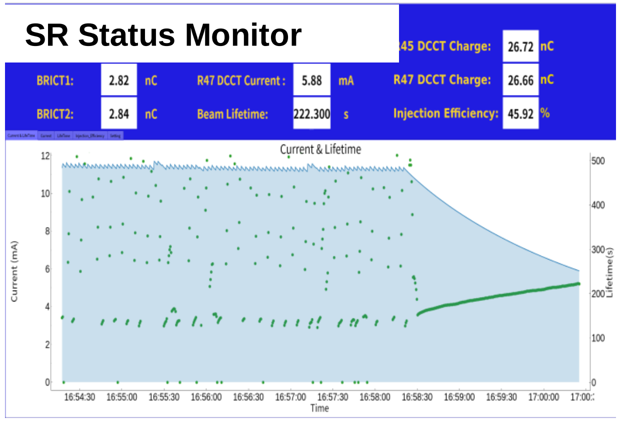


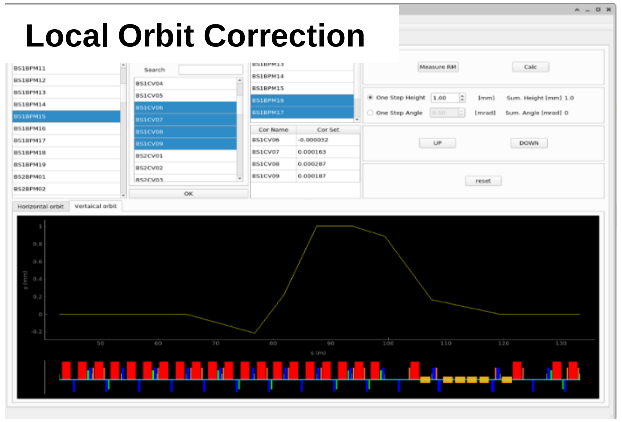






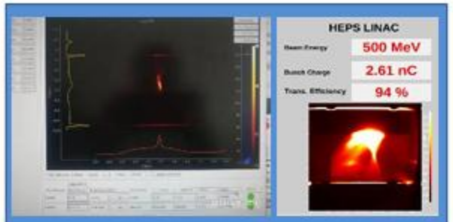




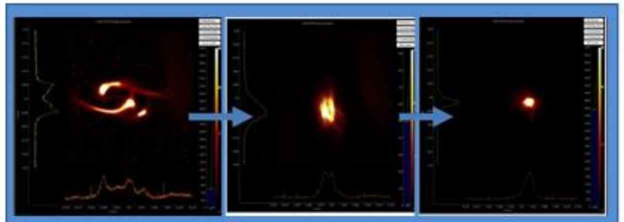


Y.L. Zhao et al., Journal of Instrumentation, 2023; Y.M. Peng et al., Journal of Instrumentation, 2023;

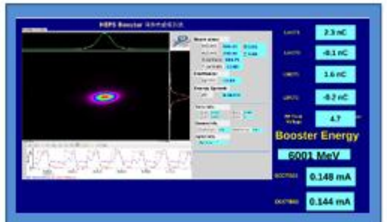
Beam Commissioning Milestones of HEPS



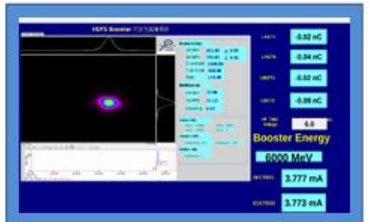
500 MeV, ~2.5 nC
1st electron beam



500 MeV, max bunch charge > 7 nC
LINAC passed SAT



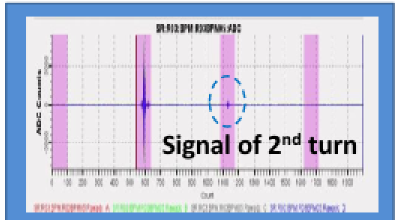
Ramped to 6 GeV
1st electron beam in Booster



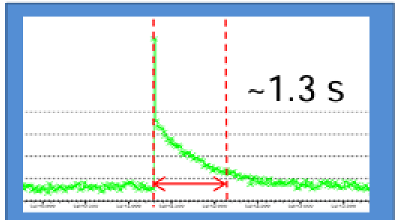
>5 nC of single bunch charge
Booster passed SAT

3.9 LINAC CX started
3.14
6.5
7.25 Booster CX started
8.9
11.17

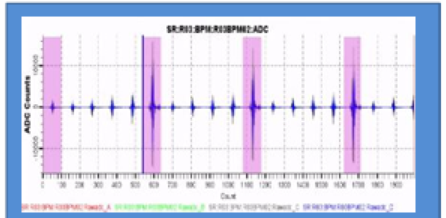
2023



First turn

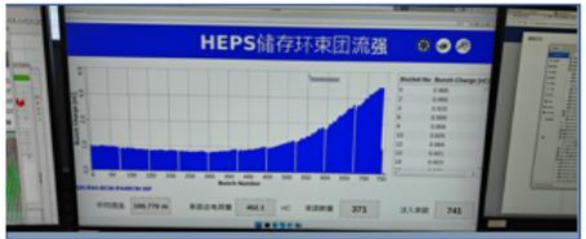


One bunch stored



Current: ~0.4 mA, Life time: ~1 min
Multi-bunch stored

Current: ~12 mA, Life time: ~2 min
12 mA stored

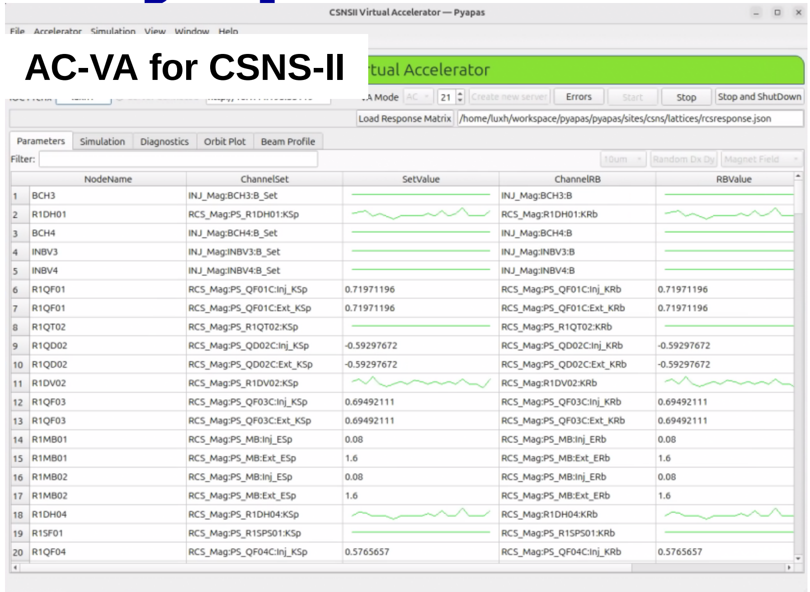


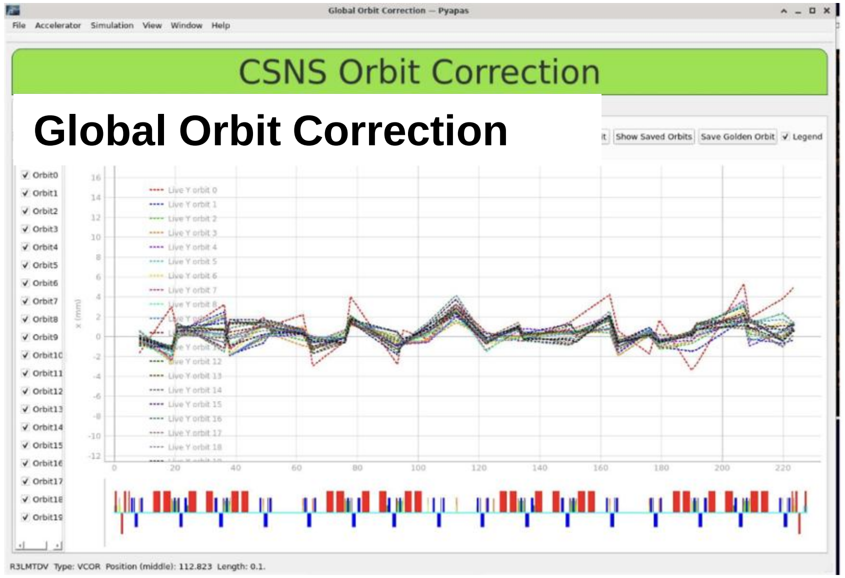
Current: ~100 mA, Emittance: ~100 pm
100 mA on Sept.12 2025

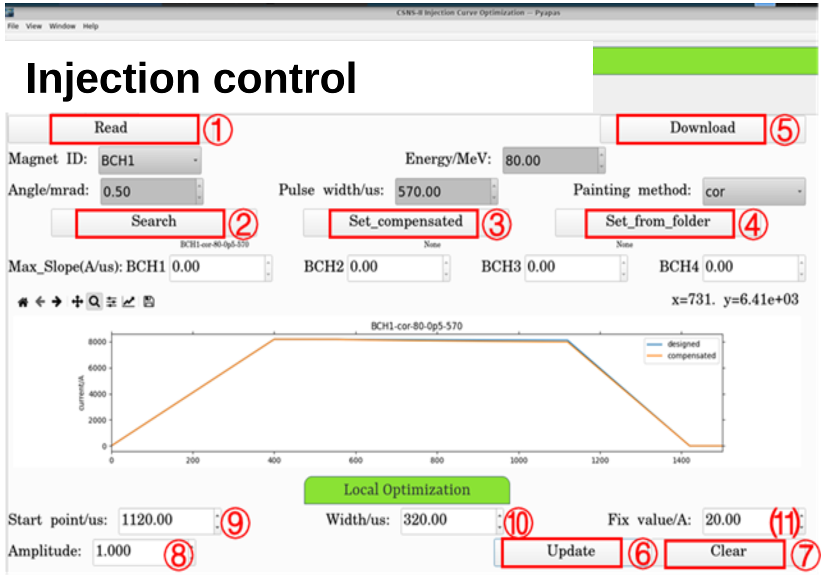
Comm. Started
7.23
7.29 10 turns
8.4 1000 turns
8.6
8.8
8.15 7 mA, 2 min
8.18

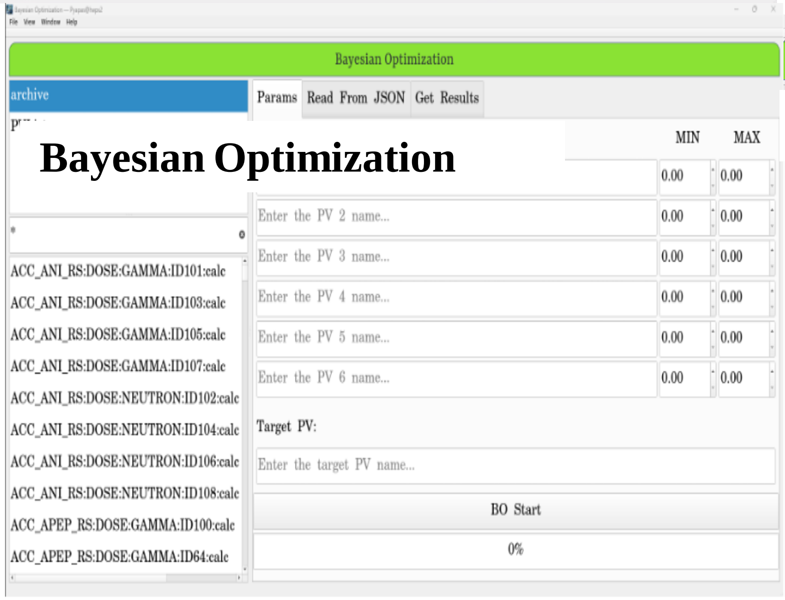
First light in Oct. 2024

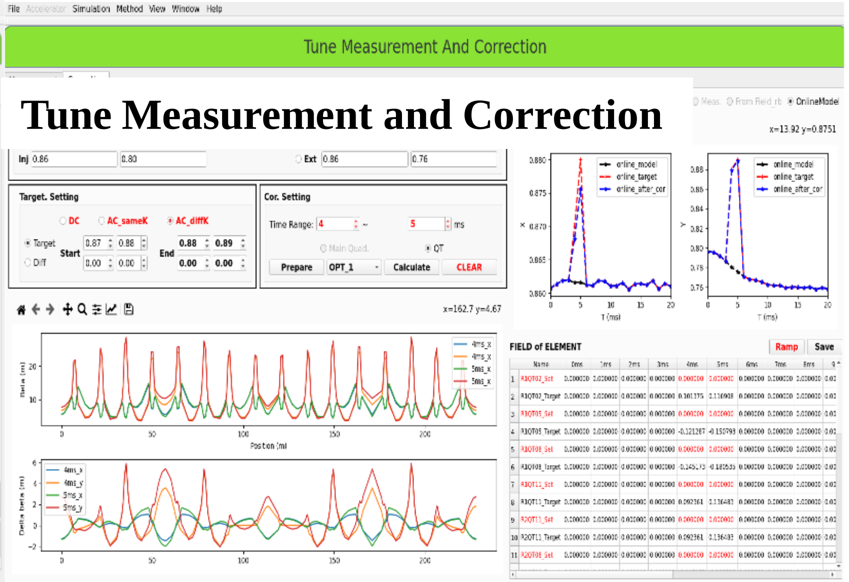
Pyapas-based HLAs for CSNS-II

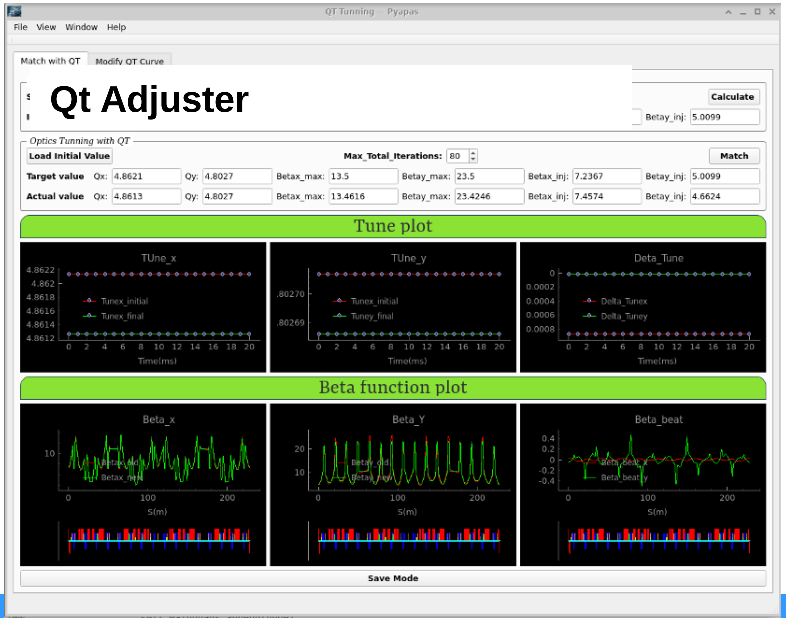






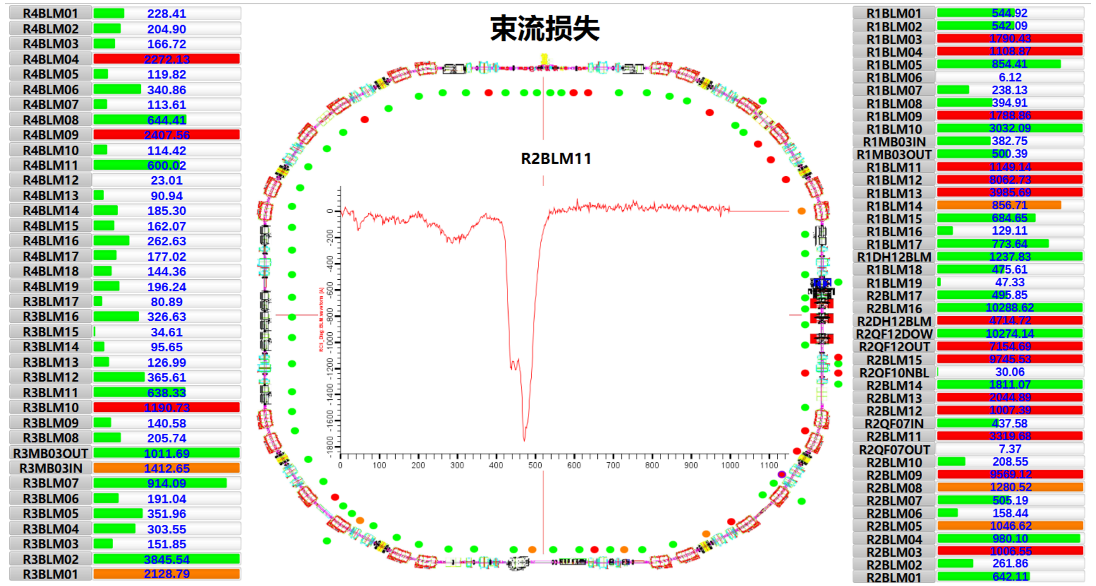


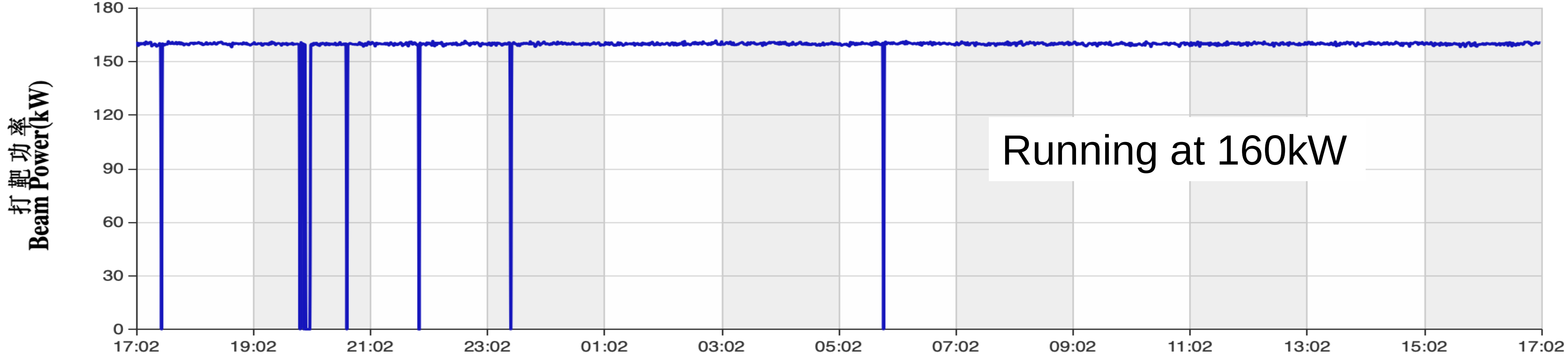




Beam Commissioning of CSNS for 160kW

LEBT CT01	36.25	mA	RTBT CT02	2.511	E13
LEBT CT02	5.10	mA	RTBT CT03	2.513	E13
MEBT CT01	7.60	mA	MEBT Trans	100.1	%
MEBT CT02	7.61	mA	DTL Trans	99.6	%
LRBT CT01	7.54	mA	LRBT Trans	101.1	%
LRBT CT02	7.54	mA	EXT Trans	100.6	%
LRBT CT03	7.62	mA	RCS Trans	97.0	%
DCCT-INJ	2.562	E13	RTBT Trans	100.6	%
DCCT-EXT	2.485	E13	Linac Energy	80.371	MeV
RTBT CT01	2.499	E13	Beam Power	161.32	kW





- HLA development requirements
- *Pyapas* for HLA development
- HLA development based on *Pyapas*
- Conclusion

Conclusion

- A brand-new framework, *Pyapas*, was developed based on Python to enhance the efficiency of HLA development.
- A multi-user virtual accelerator was developed based on *Pyapas* for testing the HLA.
- All the HLAs of HEPS were developed based on *Pyapas*
- The HLAs of CSNS-II were developed based on *Pyapas*