Beam-Dust Studies of SuperKEKB Relevant to FCC-ee

Chad Mitchell
Lawrence Berkeley National Laboratory

BLAST
BEAM PLASMA & ACCELERATOR SIMULATION TOOLKIT

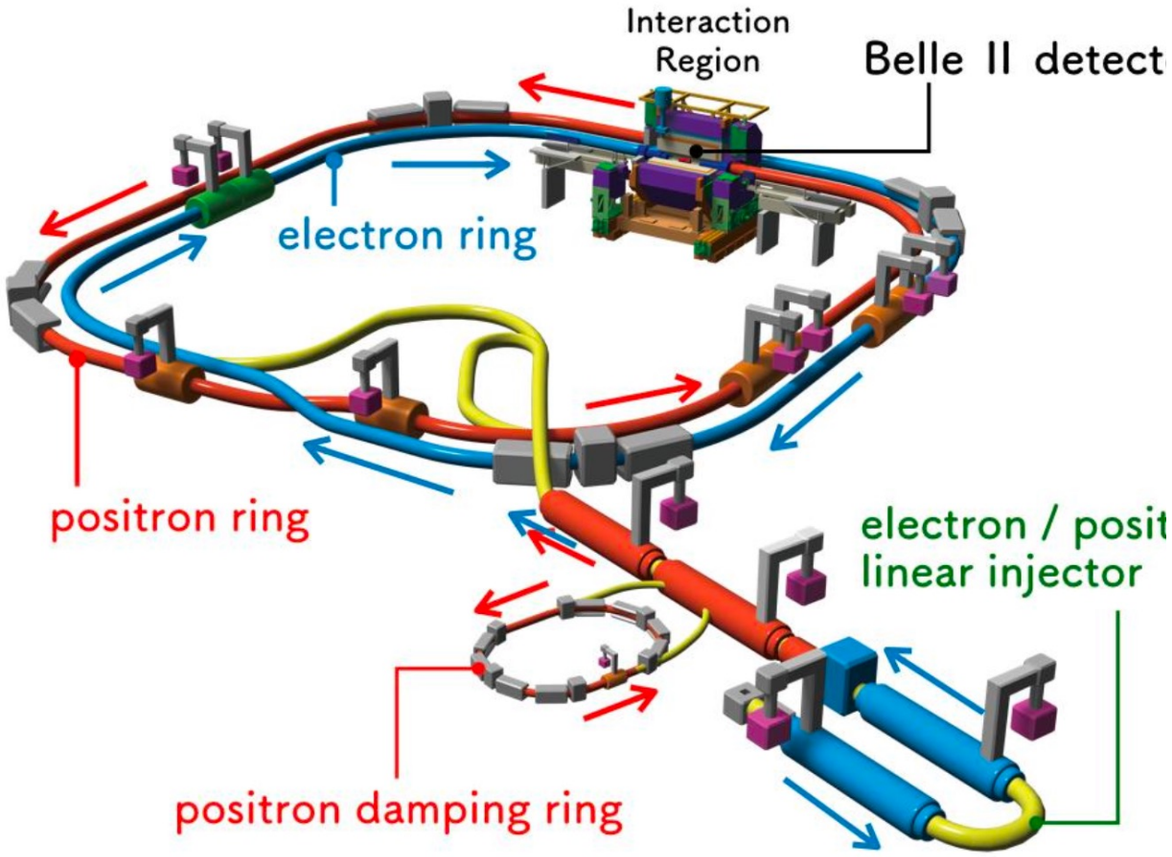
Fourth US FCC meeting
SLAC, Sep. 8, 2026

BERKELEY LAB
ACCELERATOR TECHNOLOGY & APPLIED PHYSICS DIVISION ATAP
U.S. DEPARTMENT OF ENERGY | Office of Science

Overview

- Background: beam-dust models of SBL at SuperKEKB
- Challenges and goals of this study
- Simulation Tools: HIPACE++ and ImpactX
- Multi-turn modeling of beam interaction with dust plasma
  - single-bunch studies
  - multi-bunch studies
- Future Work

SuperKEKb Layout



		e^+ LER	e^- HER
Beam energy	E [GeV]	4.0	7.0
Circumference	C [m]	3016	
Harmonic number	h	5120	
RF frequency	f_{RF} [MHz]	509	
Revolution frequency	f_0 [kHz]	100	
Minimum bunch spacing	[ns]	4	
Beam current	I [A]	1.632	1.259
Number of bunches	n_b	2346	
Charge per 1 mA bunch	[nC]	10	
Horizontal beta function at IP	β_x^* [mm]	60	60
Vertical beta function at IP	β_y^* [mm]	1.0	1.0
Horizontal tune	ν_x	44.525	45.531
Vertical tune	ν_y	46.589	43.599
Vertical beam size at IP	σ_y^* [nm]	~ 300	
Bunch length	σ_z [mm]	~ 6	
Instantaneous Luminosity	L [cm^{-2}s^{-1}]	~~5.1×10^{34}~~	$5.319x10^{34}$

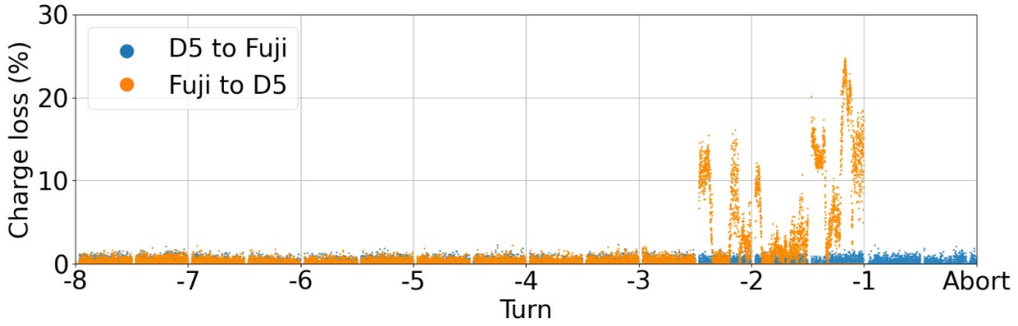
Table 1: Main parameters of the SuperKEKB main ring. Parameters below the beam current correspond to the conditions at the time of the highest recorded luminosity on December 27, 2024 [9].

- B-factory for CP violation studies and beyond Standard Model searches
- World record luminosity 5.319 x10^{34} cm^{-2}s^{-1} (June 2026) – still well below (1/15) design value

Beam-dust models of sudden beam loss

Rapid loss ~1 turn (10μs)

Vertical beam size blowup ~ 10 turns



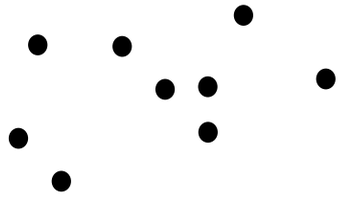
(*SBL event Oct. 28, 2024 in LER)
Most 2024 events in LER

Current understanding and mitigation:

K. Ohmi, H. Fukuma, and S. Terui, PRAB **29**, 021002 (2026)
*R. Nomaru, G. Mitsuka, and L Ruckman, JINST 21 03, T03008 (2026)
S. Terui *et al*, Phys. Rev. Accel. Beams (accepted 28 Aug, 2026)

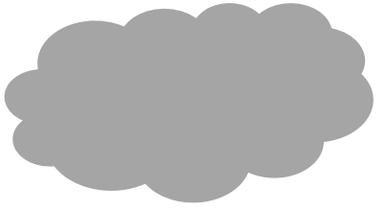
positron bunches

dust (graphite or silicon)



- ionization and heating
- fission
- evaporation/sublimation

dust plasma

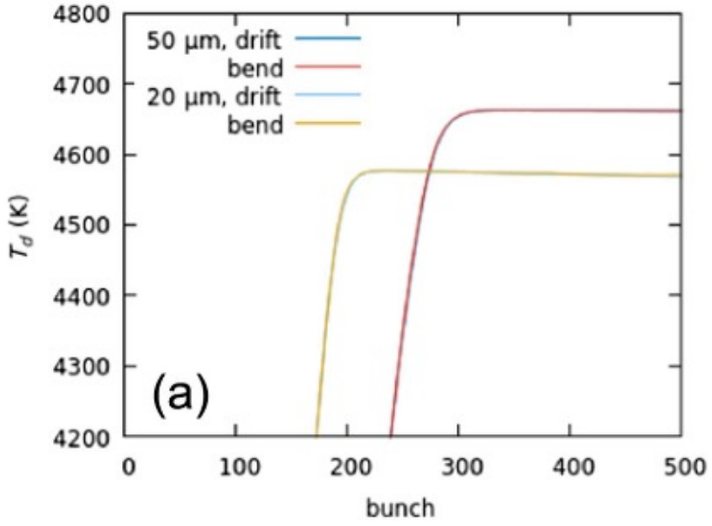


- Cleaning of VacSeal in MO-type vacuum flanges mitigated most SBL events in the LER.
- In 2026, SBLs are seen primarily in the HER when operating at high current.

Beam-dust models of sudden beam loss

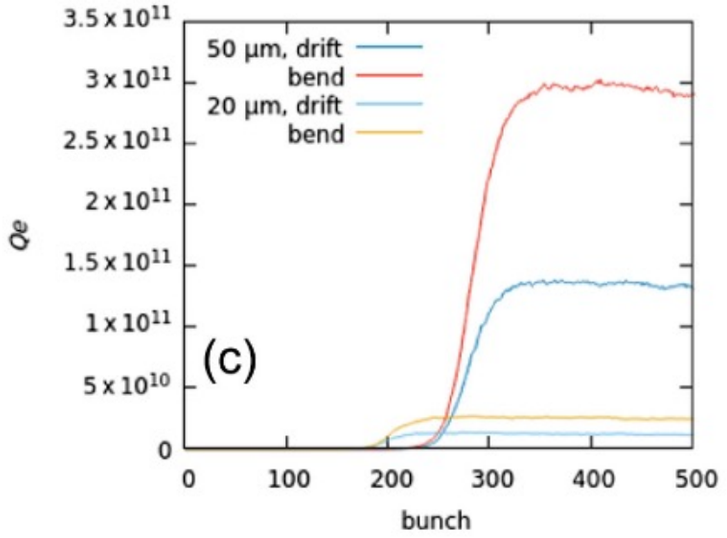
K. Ohmi *et al* (2026):

Fig. 4a



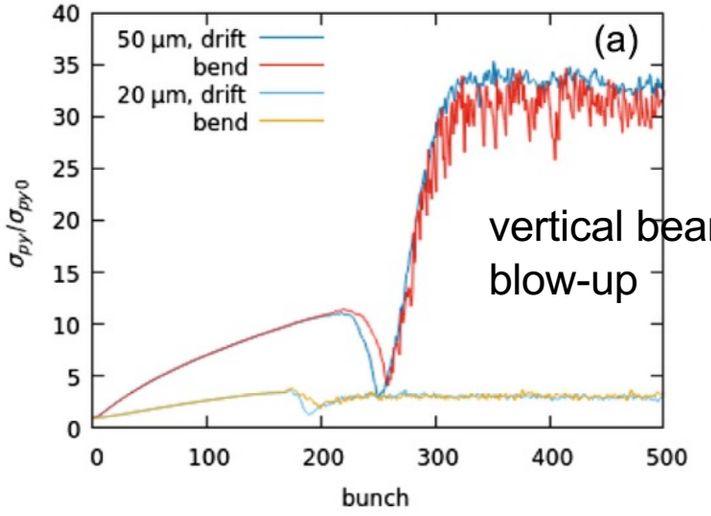
Temperature of dust/plasma

Fig. 5c



Electron charge in dust plasma

Fig. 6a



Bunch vertical momentum spread induced by plasma passage

- modeling of bunch interaction with graphite dust particle of size 20 or 50 μm (in either a drift or a bend)
- dust sublimation point is reached after 130 (20 um) - 190 (50 um) bunch passages in the train
- dust plasma reaches ~steady state *100 bunch passages after sublimation*

Challenges and goals of this study

Challenges:

Complexity of physical processes, including dust ionization/sublimation and plasma processes.

Modeling the interaction of an entire bunch train (2,346 bunches) with dust x ~10 turns is computationally infeasible. [K. Ohmi] An approximate scheme is used, using 100 bunch passages per turn (after sublimation).

The motion of electrons and ions during the interval between bunches (bunch gap of 4 ns) is challenging to resolve (plasma frequencies as large as ~284 GHz). [K.Ohmi] “Interactions within the dust plasma itself, including plasma oscillations, will be addressed in future work.”

The plasma is treated using a 2D electrostatic particle-in-cell model. [K. Ohmi] “... Since the electrons and ions condensing around the beam are the primary focus, the two-dimensional approximation is considered appropriate.”

To analyze beam loss accurately, it is necessary to track the particle distribution in phase space through the accelerator using a beam tracking code and examine losses at the collimator (code coupling).

Challenges and goals of this study

Previous work:

Small-scale study of single bunch+plasma interaction at typical SuperKEKB parameters (positron LER)

Goals of this work:

- demonstrate an integrated multi-bunch, multi-turn workflow on GPU
- coupling of plasma + lattice/beam transport codes
- contribute to validation/benchmarking of previous simulation efforts
- (longer term) investigate the size of neglected effects (e.g. 3D effects, plasma oscillations, nonlinear lattice transport)

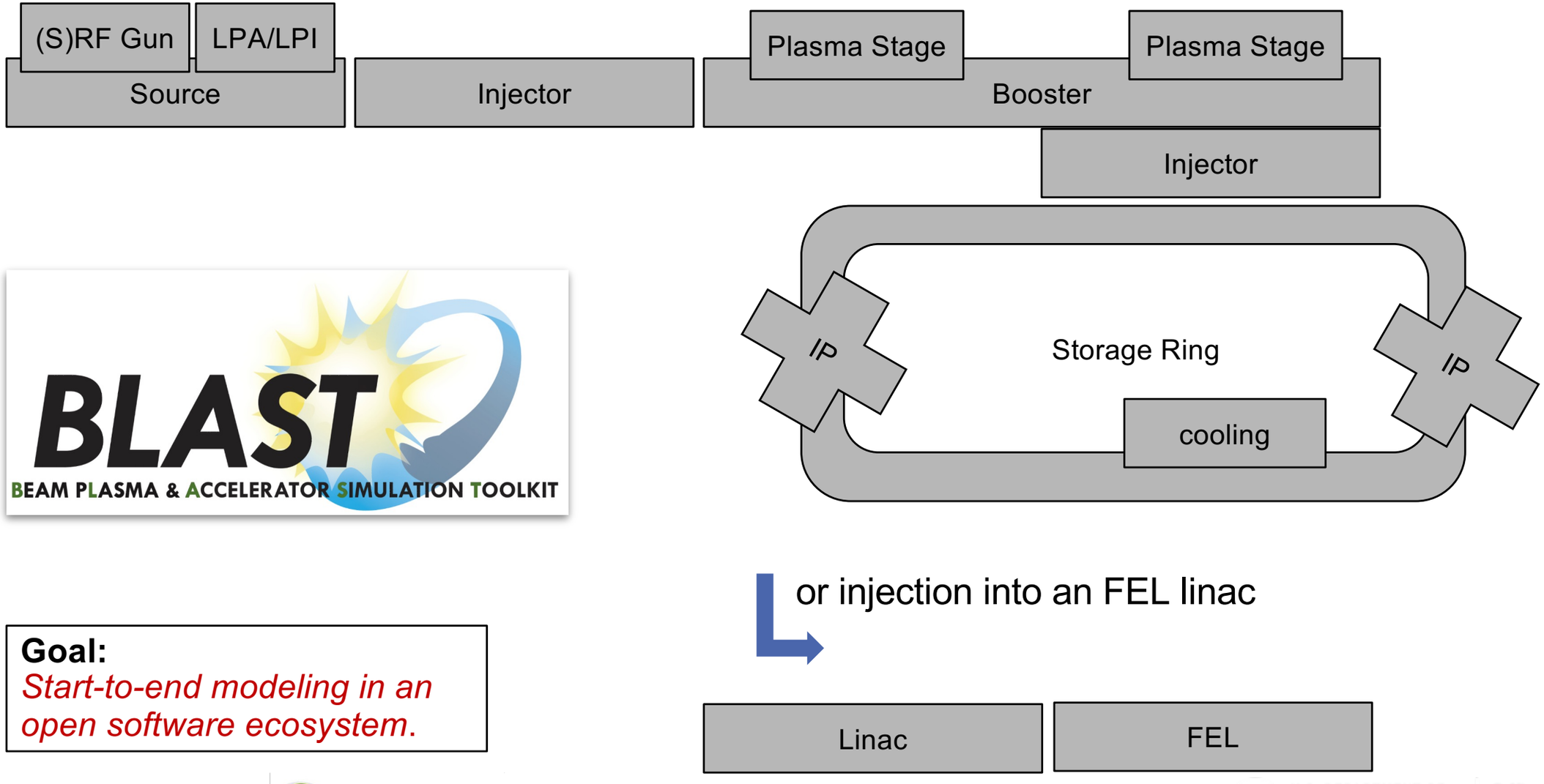
Notes:

- We track through the sublimated dust plasma, including ionization processes
- Initial heating and interaction with the dust particle are not treated

- Simulation Tools

Beam Plasma and Accelerator Simulation Toolkit: integrated beam physics across accelerator subsystems

Imagine a future, ***hybrid*** **particle accelerator**, e.g., with conventional and plasma elements.



BLAST
BEAM PLASMA & ACCELERATOR SIMULATION TOOLKIT

Goal:
Start-to-end modeling in an open software ecosystem.

Legend

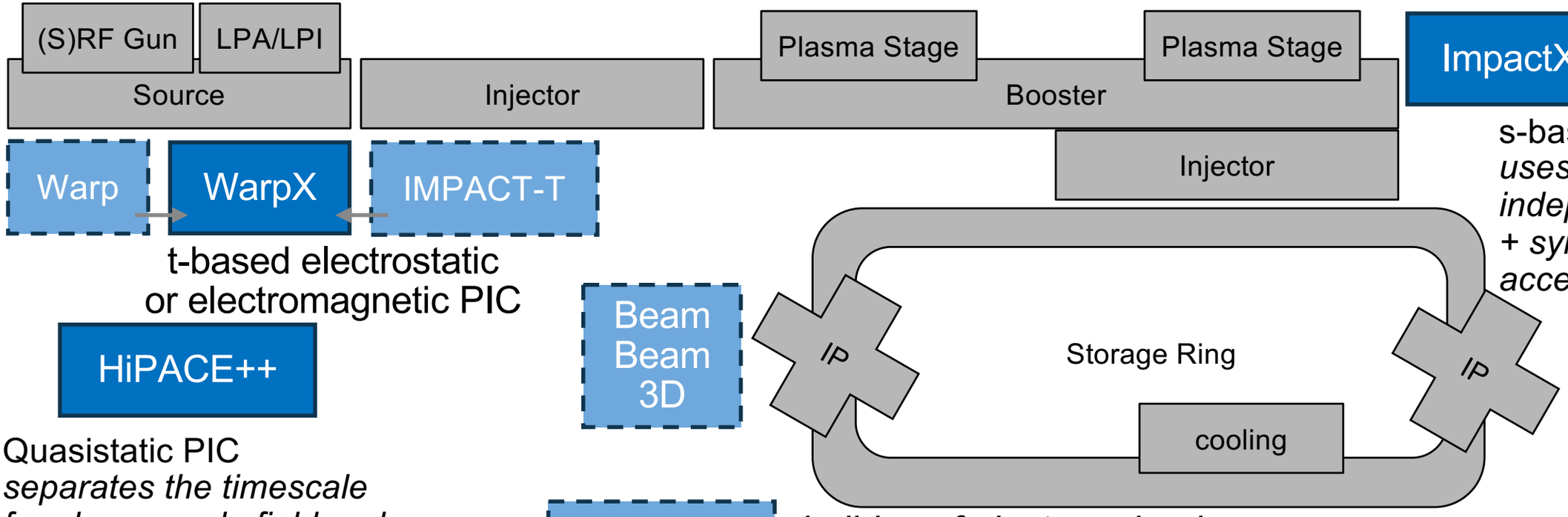
BLAST CPU-only

BLAST CPU+GPU

SciDAC
Scientific Discovery through Advanced Computing

Beam Plasma and Accelerator Simulation Toolkit: integrated beam physics across accelerator subsystems

Imagine a future, ***hybrid* particle accelerator**, e.g., with conventional and plasma elements.



Goal:
Start-to-end modeling in an open software ecosystem.

Legend

BLAST CPU-only

BLAST CPU+GPU

SciDAC
Scientific Discovery through Advanced Computing

Software stack for the BLAST ecosystem

User Interfaces

Python Bindings & Particle-In-Cell Modeling Interface (PICMI)

Apps

WarpX | HiPACE++ | ImpactX | ...

Libraries

pyAMReX

ABLASTR
common PIC components

PICSAR
QED modules

Computer Science

AMReX
Containers, Communication, Portability, Utilities

openPMD I/O, streaming, in situ: ADIOS/HDF5

Math
LinAlg., FFT

Vendor

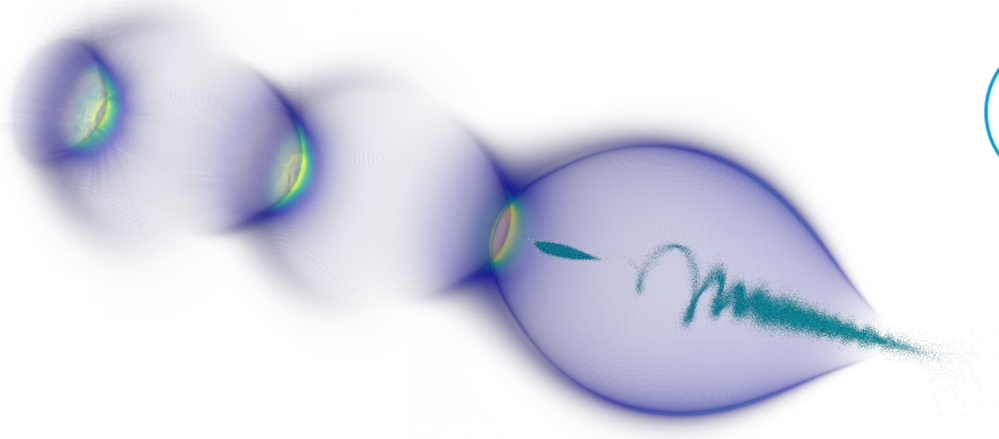
MPI multi-node comm.

CUDA, OpenMP, DPC++, HIP

Desktop to HPC

HIPACE++: quasi-static PIC on GPU for plasma acceleration

- Supports multiple beam and plasma species
- Built on top of AMReX framework (LBNL)
  Data structures and communications
  Performance-portability layer: supports **NVIDIA** & **AMD** GPUs
- HPC programming standards
  - Documented, **open-source**
  - Automated testing (CI)
- openPMD I/O



$$\nabla_\perp^2 \psi = -\frac{1}{\epsilon_0}\left(\rho - \frac{1}{c} j_z\right)$$

$$E_x - c\, B_y = -\partial_x \psi$$

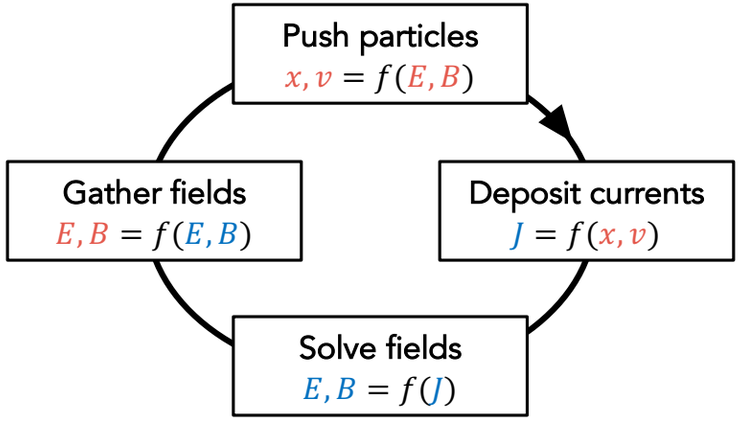
$$E_y + c\, B_x = -\partial_y \psi$$

$$\nabla_\perp^2 E_z = c\mu_0 \left(\partial_x j_x + \partial_y j_y\right)$$

$$\nabla_\perp^2 B_x = \mu_0 \left(-\partial_y j_z + \partial_\zeta j_y\right)$$

$$\nabla_\perp^2 B_y = \mu_0 \left(\partial_x j_z - \partial_\zeta j_x\right)$$

$$\nabla_\perp^2 B_z = \mu_0 \left(\partial_y j_x - \partial_x j_y\right)$$



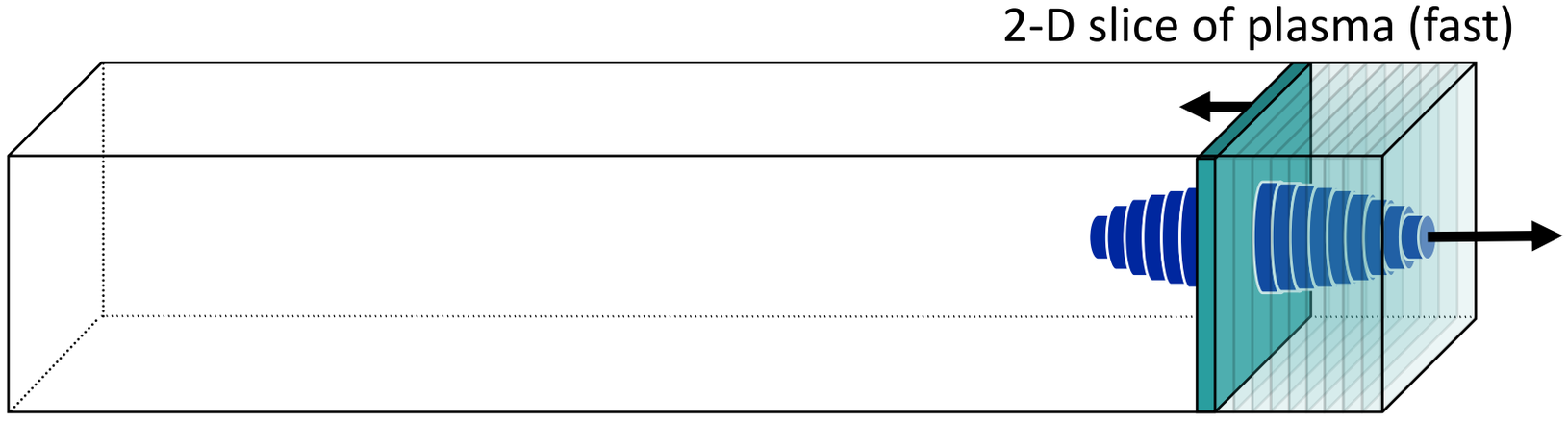
HiPACE++

3D quasi-static PIC code
Developed by DESY & LBNL
PI: Maxence Thévenet

https://github.com/Hi-PACE/hipace
Doc: https://hipace.readthedocs.io
Diederichs et al., CPC 278, 108421 (2022)

Quasistatic approximation uses separation of scales

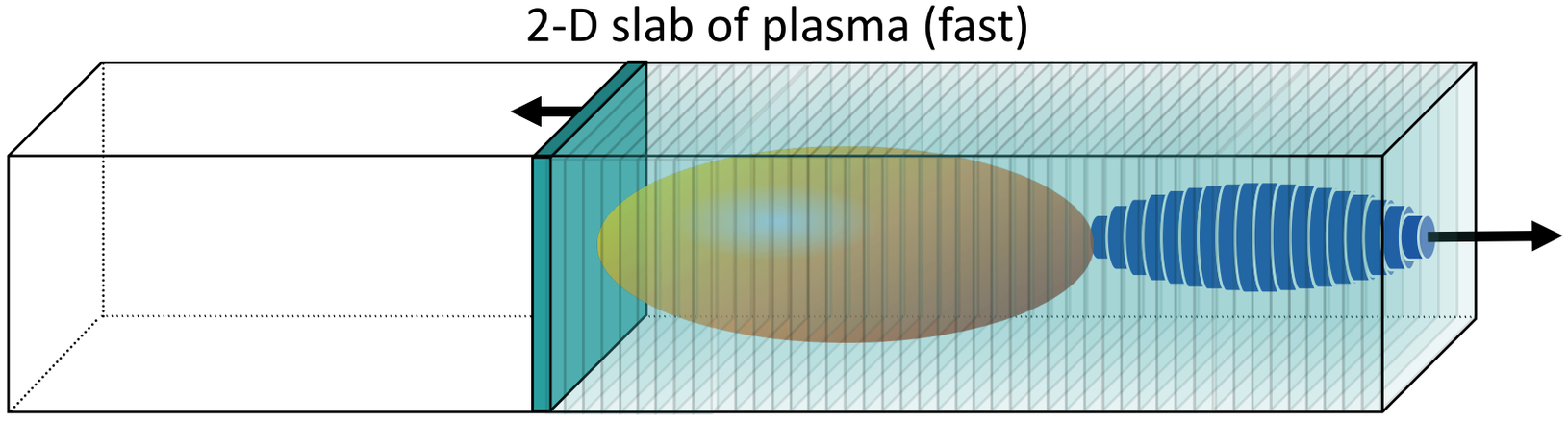
1. **Fast time scale -** 2-D slice of plasma is **stepped "backward"** (with **small steps**) through the beam field and its self-field (solving iteratively 2-D Poisson-like equations at each step).



- Quasi-static approximation (QSA)
  Sprangle, P. et al., *PRL* 64.17 (1990): 2011

Quasistatic approximation uses separation of scales

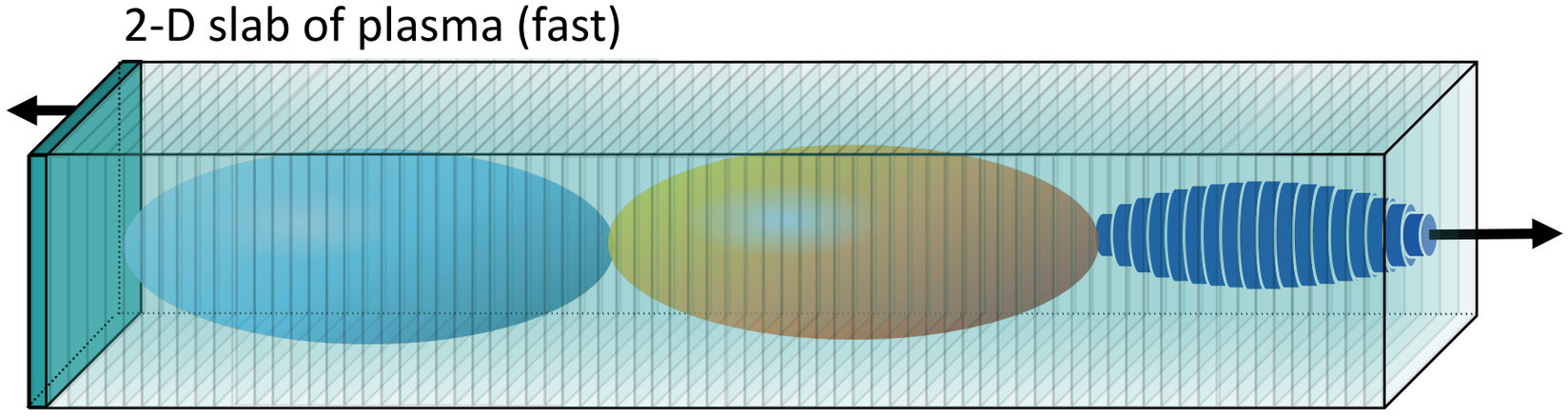
1. **Fast time scale -** 2-D slice of plasma is **stepped "backward"** (with **small steps**) through the beam field and its self-field (solving iteratively 2-D Poisson-like equations at each step).



- Quasi-static approximation (QSA)
  Sprangle, P. et al., *PRL* 64.17 (1990): 2011

Quasistatic approximation uses separation of scales

1. **Fast time scale -** 2-D slice of plasma is **stepped "backward"** (with **small steps**) through the beam field and its self-field (solving iteratively 2-D Poisson-like equations at each step).

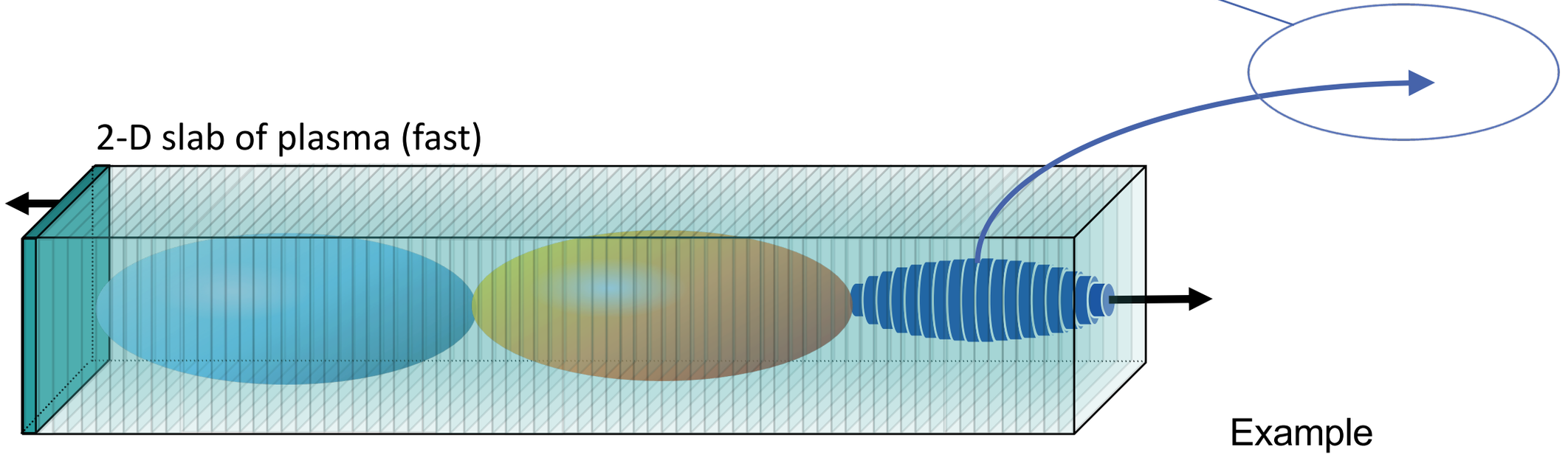


- Quasi-static approximation (QSA)
  Sprangle, P. et al., *PRL* 64.17 (1990): 2011

Quasistatic approximation uses separation of scales

repeat

1. **Fast time scale -** 2-D slice of plasma is **stepped "backward"** (with **small steps**) through the beam field and its self-field (solving iteratively 2-D Poisson-like equations at each step).
2. **Slow time scale -** 3-D beam is pushed far down the plasma with **large time steps**.



3-D beam + driver + wake
(slow: frozen during interaction)

Example

Costs: 1 hour on a laptop (NVIDIA RTX2070) in FP32, $\mathbf{1024}\times\mathbf{1024}\times\mathbf{1024}$ grid points, 10^7 beam particles, **300 time steps**

- Quasi-static approximation (QSA)
  Sprangle, P. et al., *PRL* 64.17 (1990): 2011

ImpactX: GPU-, AMR- & AI/ML-Accelerated Beam Dynamics

Physical Model based on IMPACT-Z

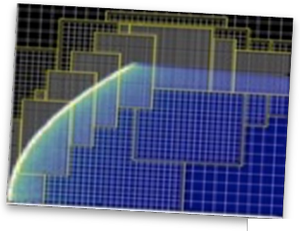
- s-based tracking with symplectic maps
- detailed RF cavity models and standard magnetic elements (w/soft-edge models)
- exact nonlinear maps for sbends, drifts
- translation/rotation errors and apertures
- radiation (CSR/ISR) and spin tracking

Space Charge Model

- space charge included using a split-operator approach
- 3D, 2D, and 2.5D models
- Multi-Level, Multi-Grid Poisson solver

Triple Acceleration Approach

- GPU support
- Mesh Refinement (AMReX)
- AI/ML & Data Driven Models



User-Friendly

- single-source C++, full Python control
- fully tested
- fully documented

Same Script
CPU/GPU & MPI

Multi-Node parallelization

- MPI: domain decomposition
- dynamic load balancing (in dev.)

On-Node Parallelization

- GPU: CUDA, HIP and SYCL
- CPU: OpenMP

Scalable, Parallel I/O

- openPMD
- in situ analysis

github.com/**BLAST-ImpactX/impactx**

BERKELEY LAB LDRD SciDAC Scientific Discovery through Advanced Computing

- Multi-pass modeling of beam interaction with dust plasma

Beam and lattice parameters (modeled in HIPACE++)

Positron bunch

beam energy: 4 GeV
positrons per bunch: 2×10^{10}
geometric rms emittances: $\varepsilon_x = 5.56$ nm, $\varepsilon_y = 29.9$ pm
RMS bunch length: 6.5 mm
RMS relative momentum spread: ~0.08%

Parameters targeting sudden beam loss event March 2024 (SuperKEKb - LER)

Setup and parameters suggested by F. Zimmerman

Lattice focusing

Twiss functions: $\beta_x = \beta_y = 10$ m, $\alpha_x = \alpha_y = 0$
Tunes (2023): 44.540 (x) / 46.601 (y) / -0.0235 (synchrotron)

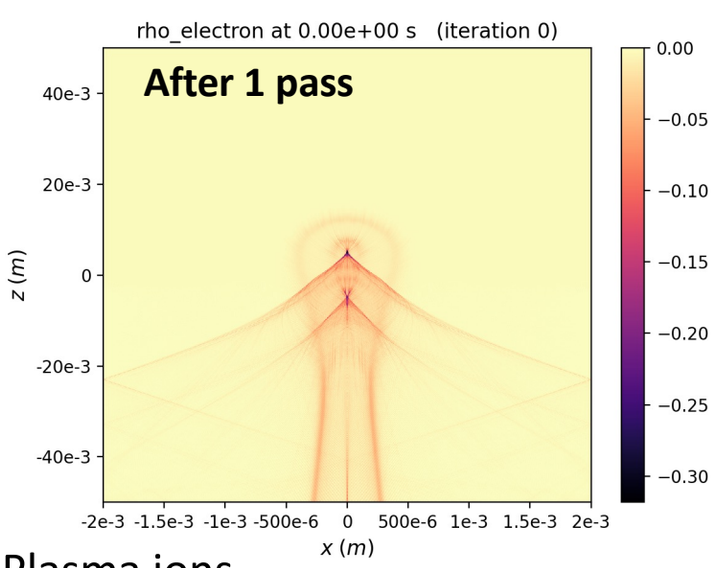
Dust plasma / e-cloud

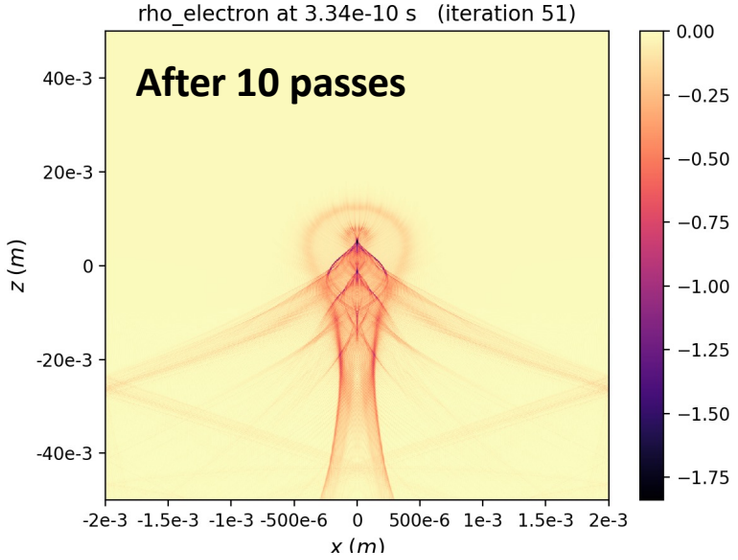
Dust plasma over 10 cm length – for several values of density in the range 10^{11} – 10^{17} / m^3
Dynamical effects on a single pass become visible at plasma e- densities ~ 10^{16} / m^3

Model multiple passes of a bunch interacting with the same plasma (via one-turn map)

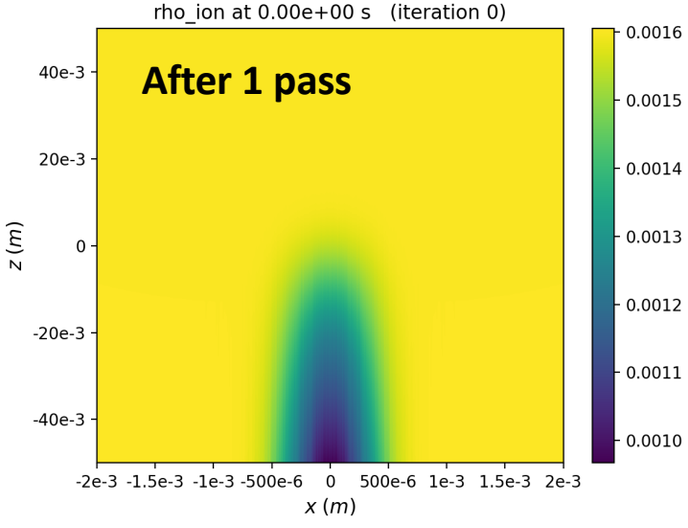
Plasma evolution for a 0.1 m plasma of density $10^{17}/m^3$

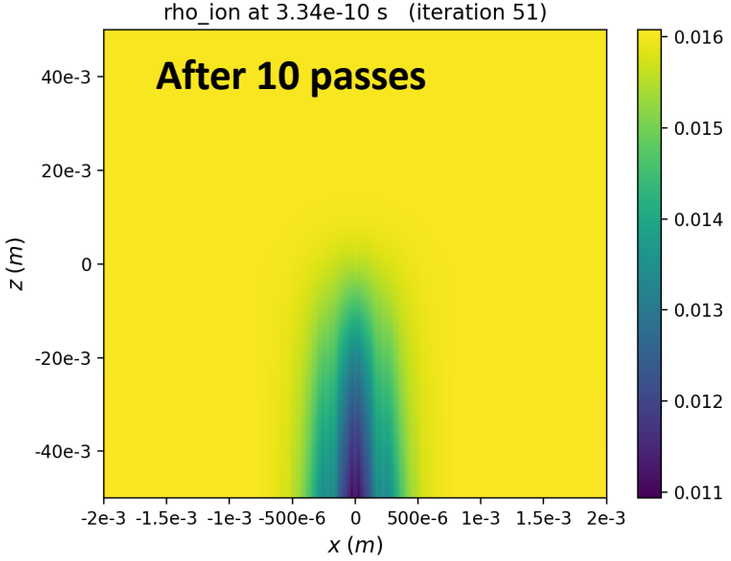
Plasma electrons





Plasma ions



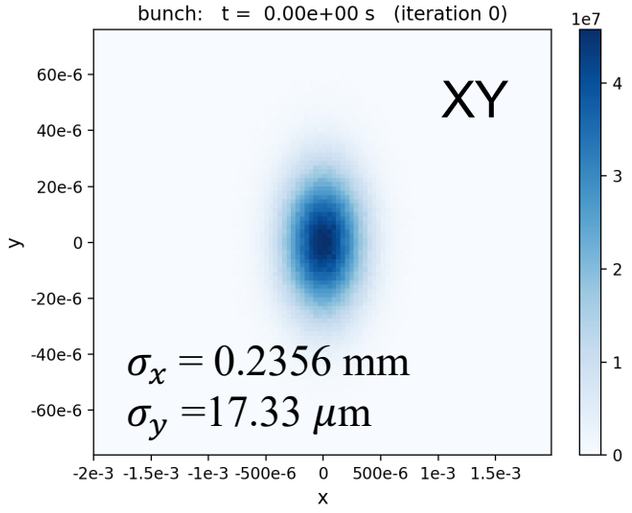


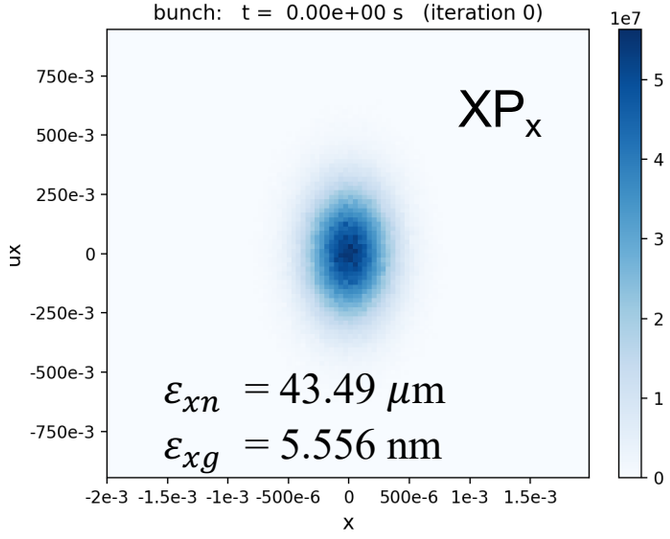
10 passes of the positron bunch

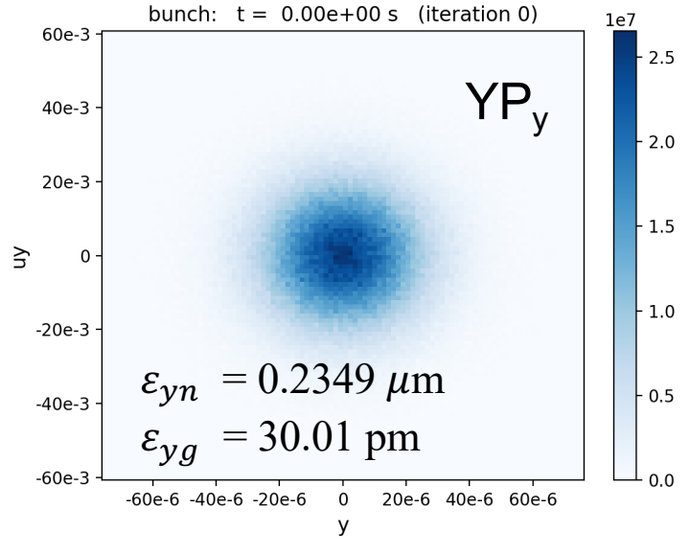
Visualization using openPMD-viewer (Jupyter notebook)

Beam evolution in a 0.1 m plasma of density $10^{17}/m^3$

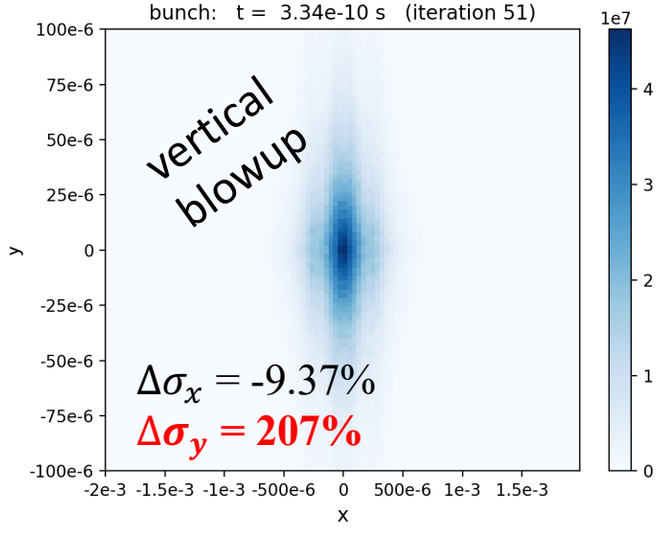
Initial bunch

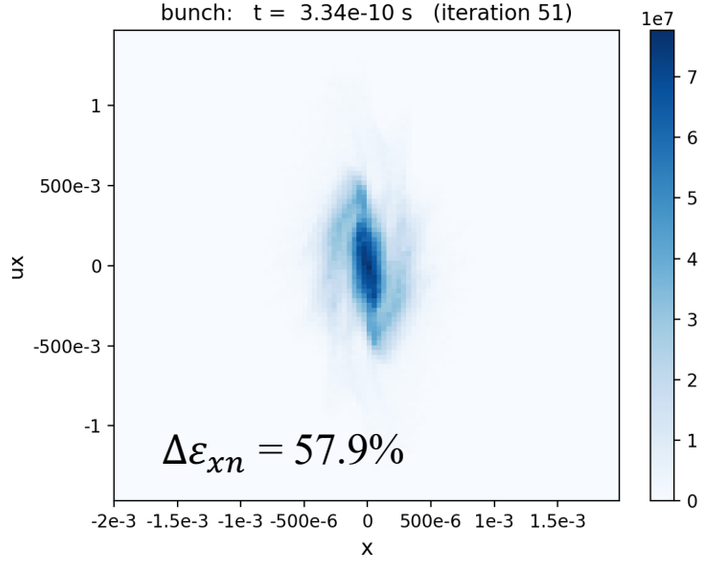


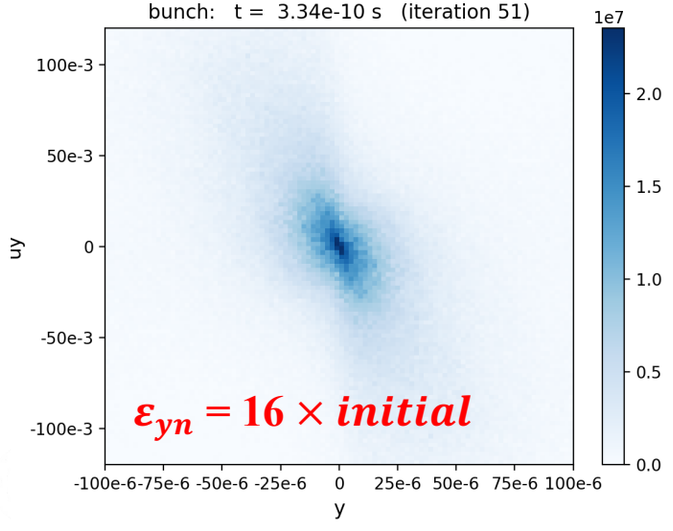




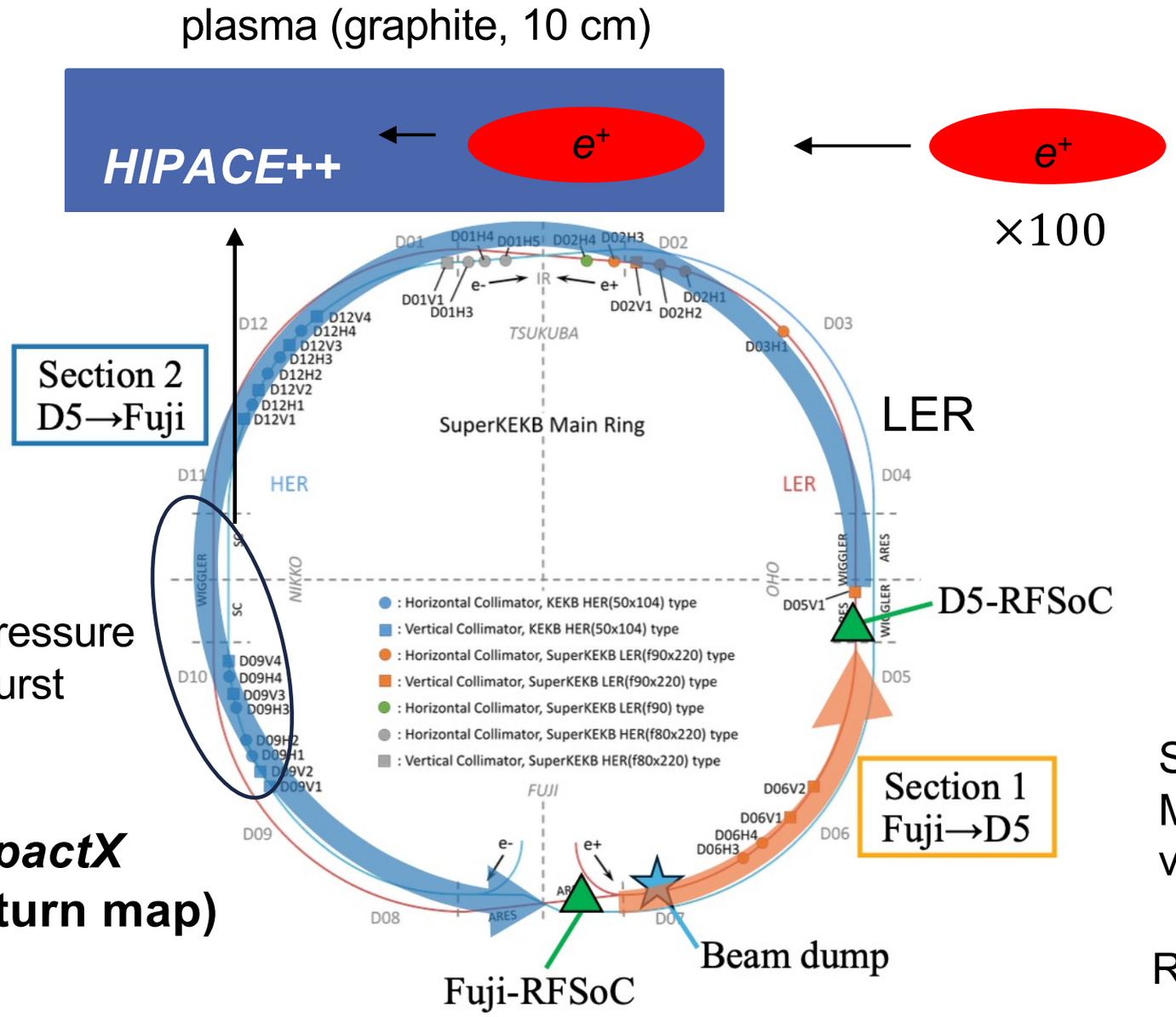
After 10 passes







Workflow for integrated multi-bunch multi-turn modeling



ImpactX
(1-turn map)

Plasma transit time: 0.33 ns

Circulation time: 10.06 μs

Bunch spacing: 1-4 ns

Number of bunches: 2,346
(Simulated train: 100)

Bunch population: 2-6.25 x 10^{10}
(Simulated particles/bunch: 10^6)

SBL event Oct. 28, 2024:
Most charge lost on
vertical collimators in Section 1

R. Nomaru *et al* (2026), Fig. 3

Workflow for integrated multi-bunch multi-turn modeling

Simulation Workflow **(NERSC Perlmutter, 8 x A100 GPUs)**

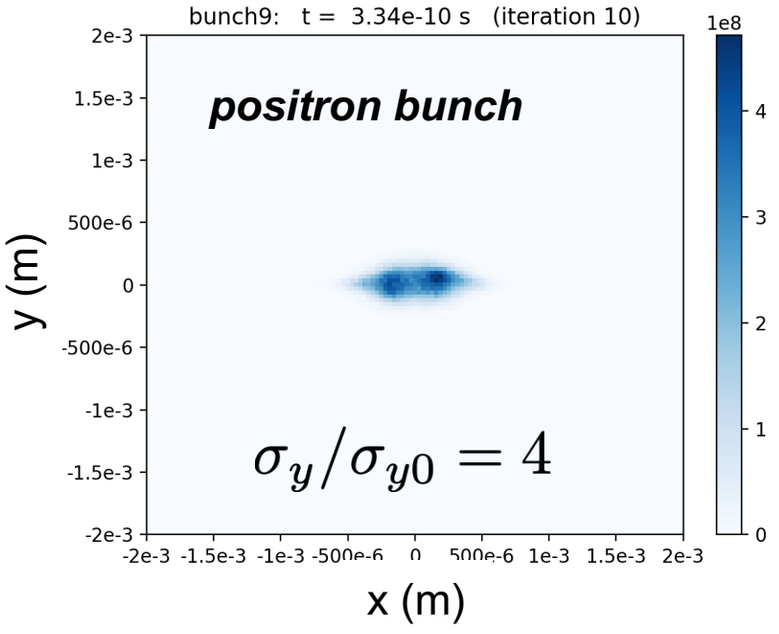
- use HIPACE++ to model interaction of a train of 100 identical bunches with the (10 cm) plasma channel
- store the final state of the *last bunch* and the *plasma* as openPMD (hdf5) data [to approximate steady-state plasma and bunch properties associated with the train passage]
- read the bunch particle data into ImpactX and propagate for 1 turn (currently, using the 1-turn map within ImpactX)
- use the updated bunch data to populate a new train of 100 bunches together with the stored plasma in HIPACE++
- push the updated bunch train through the updated plasma using HIPACE++
- repeat the previous steps x 10 times to model 10 turns

Observations (ongoing / WIP)

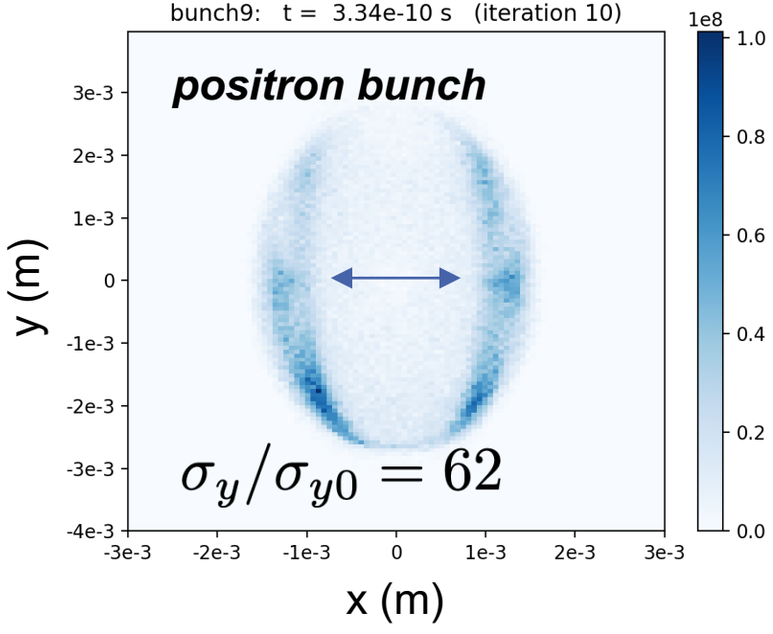
- Low-resolution exploratory runs reveal compute time is dominated by: 1) number of longitudinal grid cells required to resolve the train (>10^5), 2) openPMD data management for the bunch train I/O, 3) queue wait times (chained dependency jobs). Note 2) + 3) could be addressed with improved code integration.

- Demonstration using a short train of 10 bunches (~10^4 longitudinal cells):

After 3 turns (10th bunch)

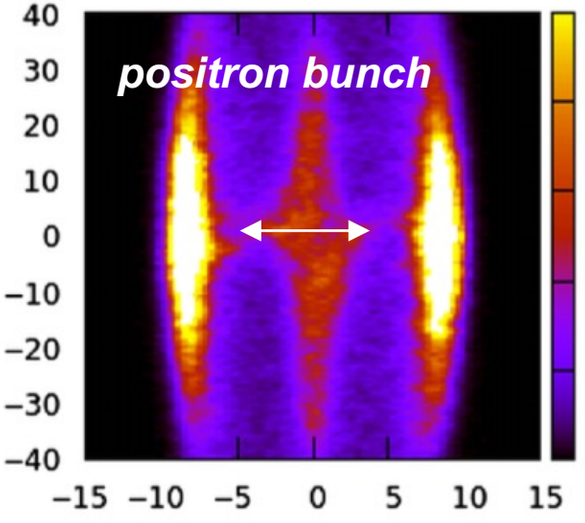


After 4 turns (10th bunch)



$n_e = n_i = 2\times10^{18}$ / m^3

Qualitative comparison only



K. Ohmi *et al* (2026), Fig. 13A

Summary and Future Work

- Developed an integrated simulation workflow modeling beam blow-up triggered by a long bunch train interacting with graphite dust plasma over 10 turns in the SuperKEKB LER.
- Exploited coupling of HIPACE++ with ImpactX running on NERSC Perlmutter GPU; compute time is strongly dominated by plasma interaction.

Next steps:

- Use full element-by-element tracking in ImpactX model to support lattice transport from proposed dust location (pressure bump) to vertical collimator, and back to dust location.
- Collaborate toward benchmarking among codes and with experimental data for SuperKEKB.
- Investigate the size of neglected effects (e.g. 3D effects, plasma oscillations, nonlinear lattice transport, etc.)

Backup material

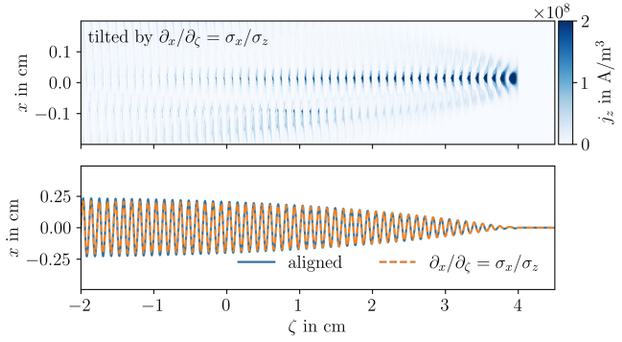
Examples: Applications of HIPACE++

DESY

*Our JUWELS allocation: 26 000 node hours
"large JUWELS allocation": 156 000 node hours

Tilted proton bunch in AWAKE

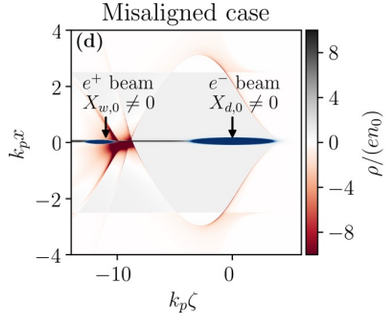
Long beam, many beam particles, long simulation box, long plasma



Costs: 1 node hour*
on the JUWELS Booster (16 nodes), in FP64, **512×512×2048** grid points,
120×10^6 beam particles, **400 time steps**

e^+ acceleration in a plasma column

High-resolution, sharp plasma spikes, many time steps, many beam particles

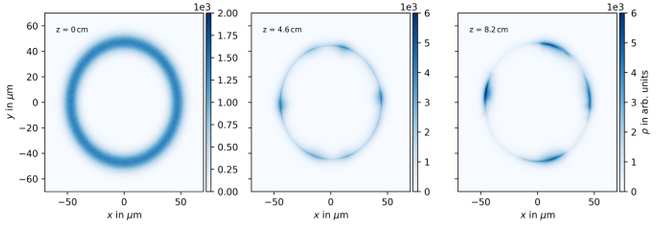


Costs: 18 node hours*
on the JUWELS Booster (32 nodes), in FP64, **4096×4096×2048** grid points, 225×10^6 beam particles, **750 time steps**

e^+ acceleration with hollow-core driver

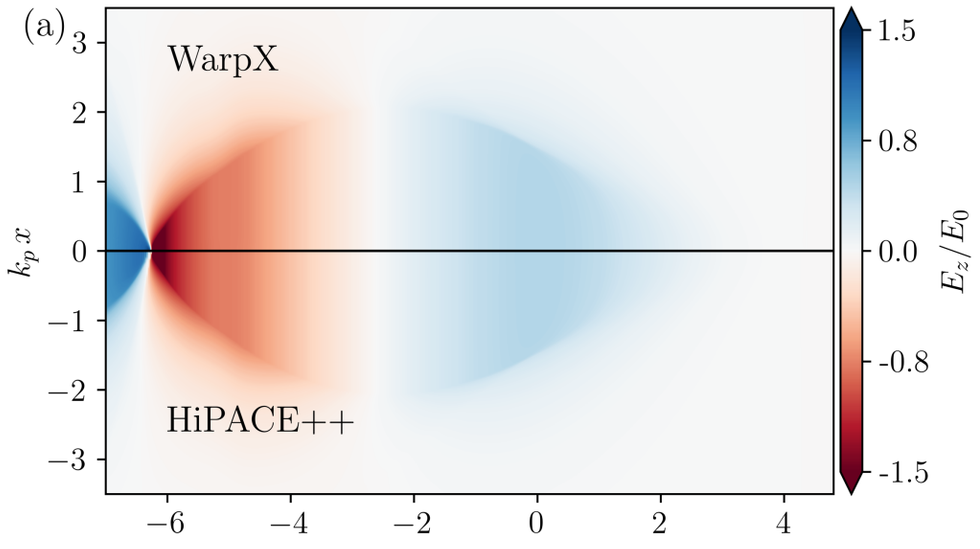
(Jain et al., PRL 2015)

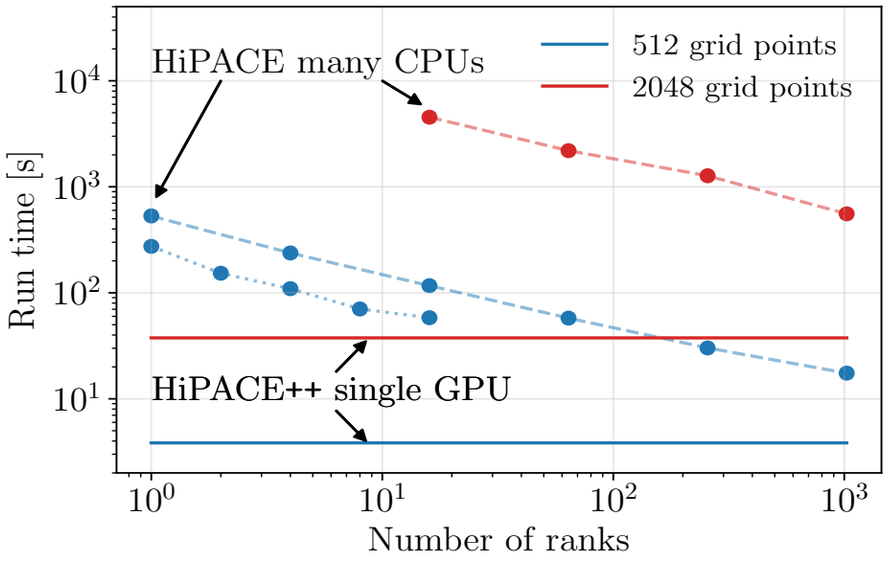
Standard 3D simulation



Costs: 1 hour on a laptop (NVIDIA RTX2070)
in FP32, **1024×1024×1024** grid points, 10^7 beam particles, **300 time steps**

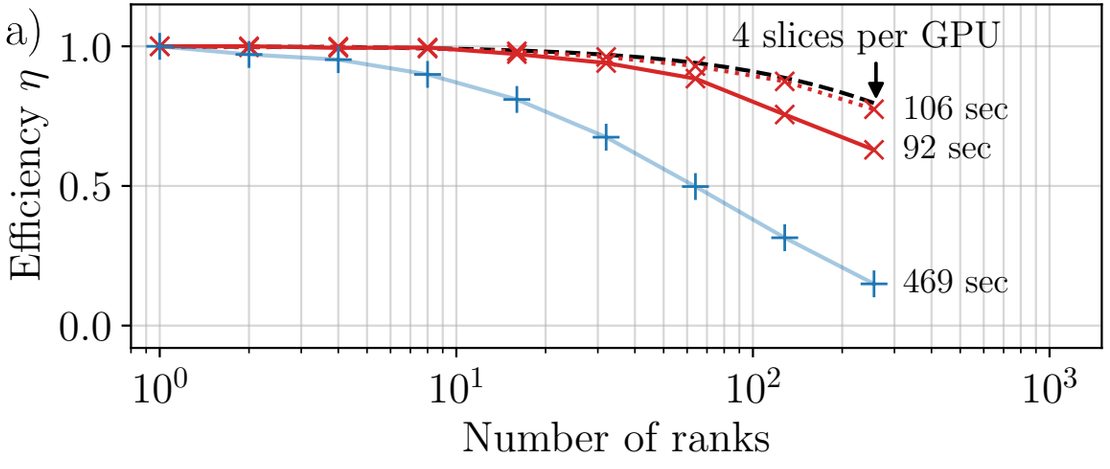
HIPACE++ accuracy and performance benchmarks





- 1 GPU MUCH faster than many CPU cores

- Successfully benchmarked
- Large speedup w.r.t. CPU
- High-resolution simulations within minutes
- Production simulations possible on a laptop (`float`)



Summit

Juwels

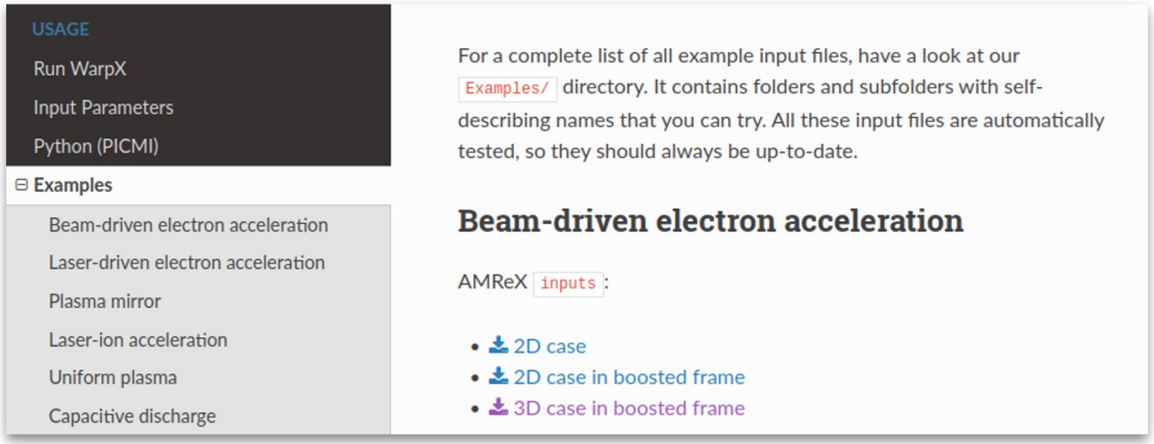
- Excellent scaling to hundreds of GPUs

Slide from M. Thevenet (DESY)

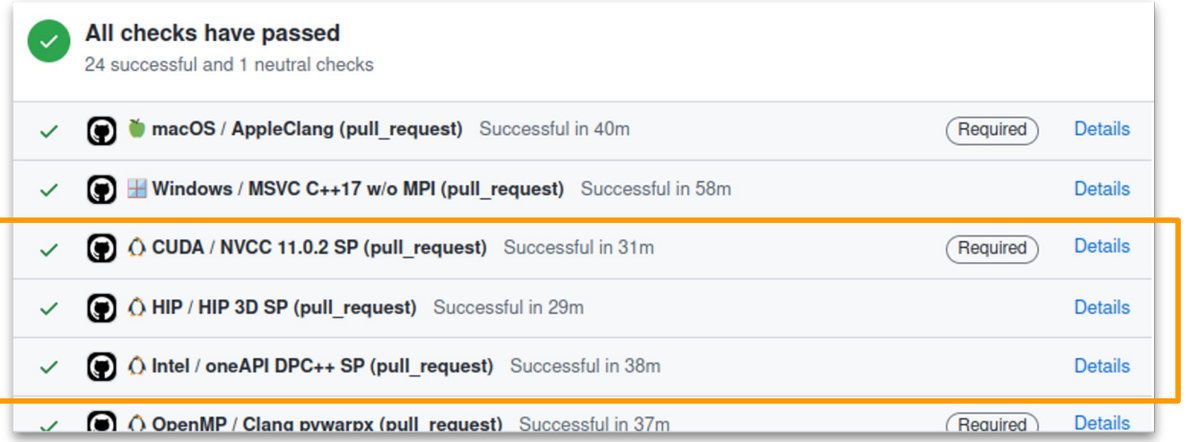
Documentation & Testing is Part of Our Development

BLAST
BEAM PLASMA & ACCELERATOR SIMULATION TOOLKIT

Online Documentation:
warpx|hipace|impactx.readthedocs.io



Open-Source Development & Benchmarks:
github.com/BLAST-ImpactX



264 physics benchmarks *run on every code change* of WarpX
262 physics benchmarks for ImpactX (Sept. 2026)

Rapid and easy installation on any platform:

conda install
-c conda-forge warpx

spack install warpx
spack install py-warpx

cmake -S . -B build
cmake --build build --target install

python3 -m pip install .

brew tap ecp-warpx/warpx
brew install warpx

module load warpx
module load py-warpx