Warp

J.-L. Vay

May 7-9, 2018

1st BLAST workshop

BERKELEY LAB ACCELERATOR SIMULATION TOOLKIT

History

- **1989:** code started by **A. Friedman**; electrostatic PIC with 3D Poisson solver in square pipe using sine transforms
  - with inputs and contributions from I. Haber.
  - name Warp chosen to denote speed, later also denotes “warped” Fresnet-Serret coordinates used in bends.
- **1991**: **D. Grote** joins and becomes main developer, contributing more general 3D Poisson solver and beam loading module, followed by countless contributions since.
- **1995**: **S. Lund** start contributing, in particular beam loading module using various distributions, such as waterbag, Gaussian, and thermal.
- **Late 1990’s**: **R. Kishek** begins using Warp to model experiment at UMD, and in undergraduate and graduate classes. He contributes some code, including various diagnostics used at UMD and UMER, and standardized machine description for UMER.
- **2000**: **D. Grote** develops Forthon, transitioning Warp from Basis to Python, and parallelizes Warp using MPI (replacing previous PVM parallelization).
- **2000**: **J.-L. Vay** joins the development team and contributes the EM solver, ES and EM AMR, boosted frame, Lorentz invariant particle pusher, quasistatic solver, etc.
- **2013**: Warp becomes Open Source (BSD license)
- **2015**: **R. Lehe** joins the development team and contributes the EM Circ solver, OpenPMD and hdf5 IO, mpi4py; also develops spectral EM circ stand-alone module that became FBPIC (original coupling to Warp on standby).
- **2016**: **RadiaSoft**'s open source Docker and Vagrant containers make WARP available in the cloud

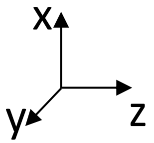
  -> available via http://beta.sirepo.com/warp

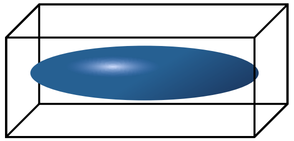
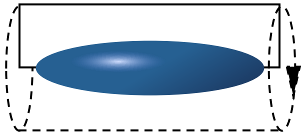
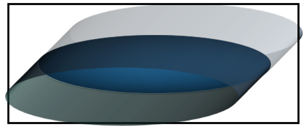
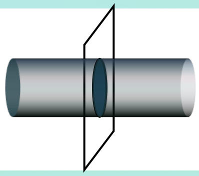
History

- Other contributors include:
  - **Ron Cohen**: drift-Lorentz mover, other fixes.
  - **Bill Sharp**: Circe module.
  - **Michiel deHoon**: Hermes module.
  - **Sven Chilton**: generalized KV envelope solver (with S. Lund).
  - **Arun Persaud**: making Warp PEP8 compliant, various fixes and updates, also adding capability to download parameters and diagnostic output from NDCX-II run database, automatically run Warp, and overlay results with experimental data, thereby facilitating machine optimization.
  - **Henri Vincenti**: fixes to Circ, OpenPMD, and EM solver; development of optimized PICSAR kernel with tiling, OpenMP and advanced vectorization.
  - **Manuel Kirchen**: mpi4py interface, boosted particle diagnostics, development of FBPIC module.
  - **Patrick Lee**: boosted particle diagnostics, other fixes.
  - **Irene Dornmair**: EM initialization for relativistic beams.
  - **Mathieu Lobet**: optimization of PICSAR kernel.
  - **Mathieu Blaclard**: generalized laser antenna.
  - **Maxence Thevenet**: various fixes.
  - **Haithem Kallala**: Warp+PXR improvements.
  - …

Warp: A Particle-In-Cell framework for accelerator modeling

- **Geometry:** 3-D (x,y,z) | axisym. (r,z) + FFT(θ) | 2-D (x,z) | 2-D (x,y)



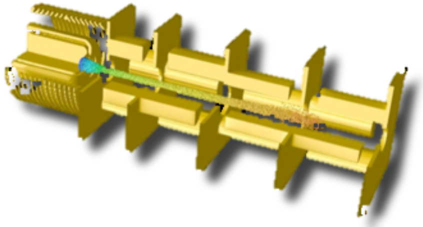





- **Reference frame:**

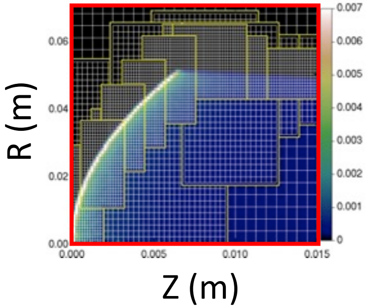
	lab	moving-window	Lorentz boosted
	z	$z-vt$	$\gamma(z-vt)$; $\gamma(t-vz/c^2)$

- **Field solvers**
  - electrostatic/magnetostatic - FFT, multigrid; AMR; implicit; cut-cell boundaries

Versatile conductor generator accommodates complicated structures



Automatic meshing around ion beam source emitter



  - Fully electromagnetic – Yee/nodal mesh, arbitrary order, spectral, PML, MR

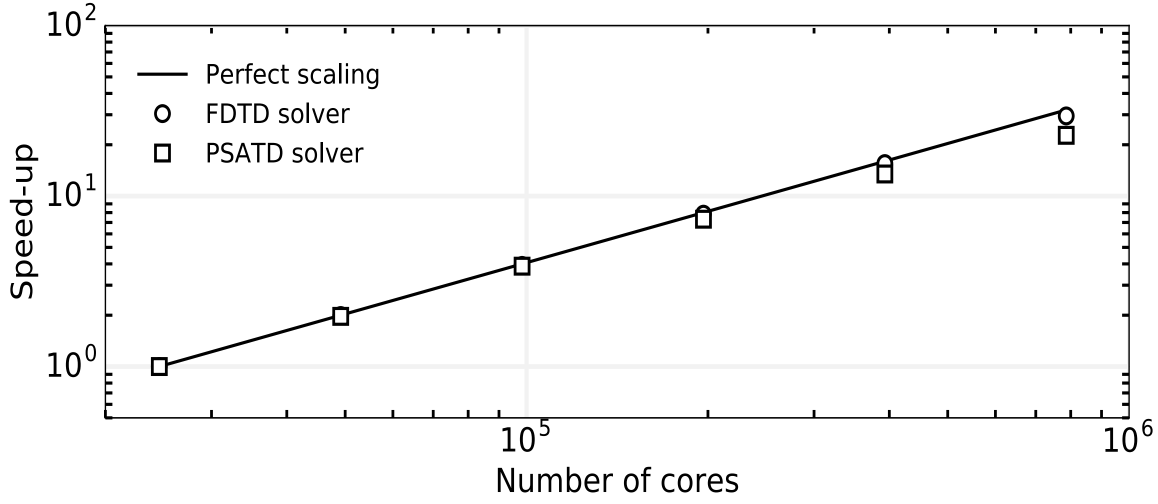
- **Accelerator lattice:** general; non-paraxial; can read MAD files
  - solenoids, dipoles, quads, sextupoles, linear maps, arbitrary fields, acceleration.

- **Particle emission & collisions**
  - Particle emission: space charge limited, thermionic, hybrid, arbitrary.
  - Secondary e- emission (Posinst), ion-impact electron emission (Txphysics) & gas emission.
  - Monte Carlo collisions: ionization, capture, charge exchange.

Warp is parallel, combining modern and efficient programming languages

- **Parallellization:** MPI (1, 2 and 3D domain decomposition)

Warp-PICSAR
strong parallel scaling
for uniform warp plasma
up-to >800,000 cores
(Mira – ANL)



- **Python and FORTRAN*:** "steerable," input decks are programs

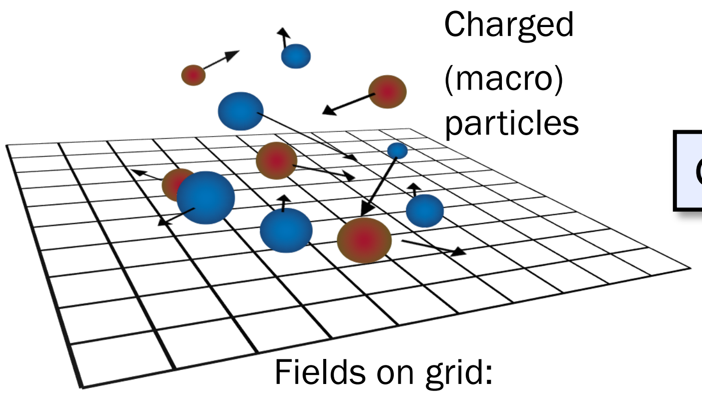
```
From warp import *
…
nx = ny = nz = 32
dt = 0.5*dz/vbeam
…
initialize()
step(zmax/(dt*vbeam))
…
```

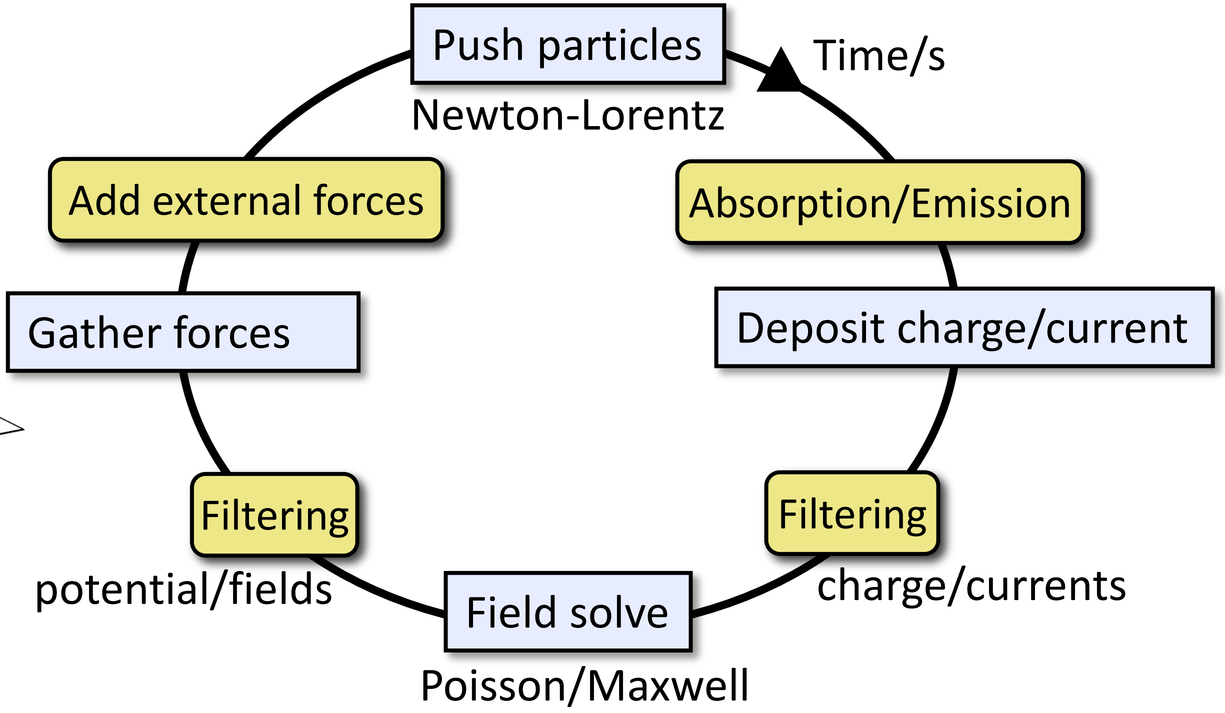
- From warp import * ← Imports Warp modules and routines in memory
- nx = ny = nz = 32 ← Sets # of grid cells
- dt = 0.5*dz/vbeam ← Sets time step
- initialize() ← Initializes internal FORTRAN arrays
- step(zmax/(dt*vbeam)) ← Pushes particles for N time steps with FORTRAN routines

*http://hifweb.lbl.gov/Forthon (wrapper supports FORTRAN90 derived types) – dpgrote@lbl.gov

Warp’s standard PIC loop

Particle-In-Cell (PIC)





+ external forces (accelerator lattice elements),

+ absorption/emission (injection, loss at walls, secondary emission, ionization, etc),

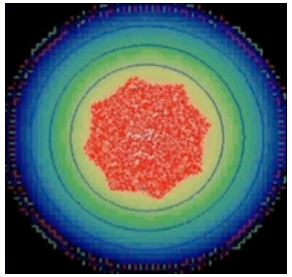
+ filtering (charge,currents and/or potential,fields).

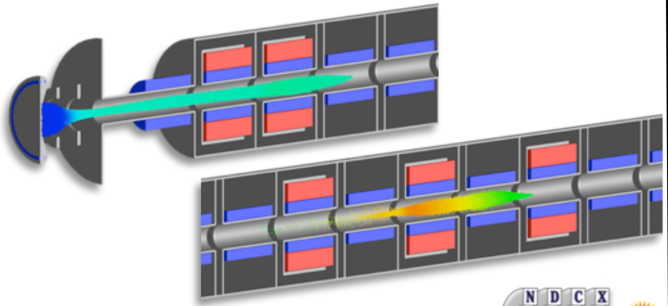
Arbitrary number and type of particle species.

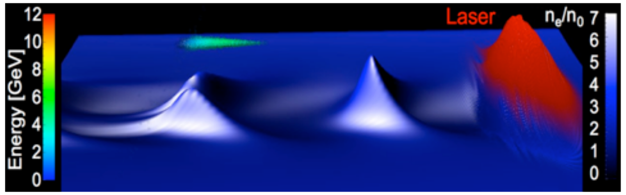
- Predefined types: periodic table, electron, positron, proton, anti-proton, muon, anti-muon, neutron, photon.
- user can define additional types.

Warp's versatile programmability enables great adaptability

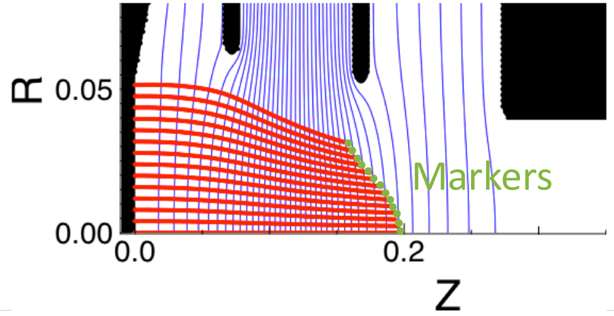
Standard PIC		
Laboratory frame	Moving window	Lorentz Boosted frame
Alpha anti-H trap	Example: Beam generation and transport	Example: Laser plasma acceleration

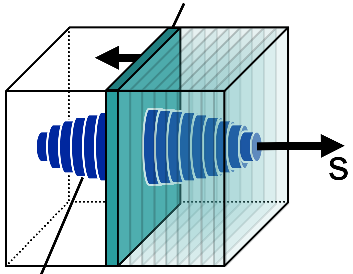


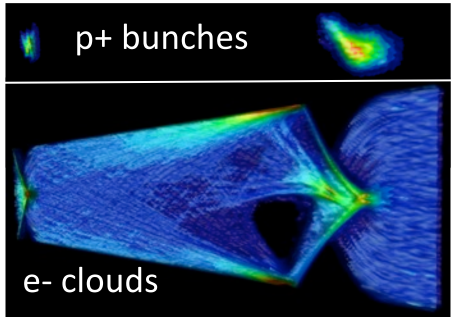




Non-standard PIC	
Steady flow	Quasi-static
Example: Injector design	Example: electron cloud studies



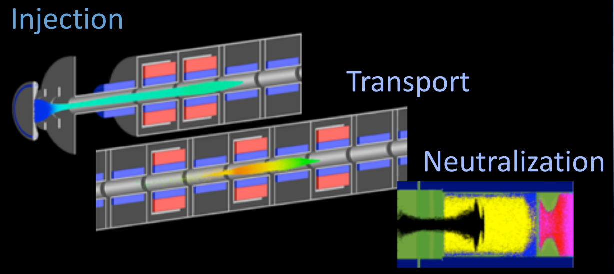




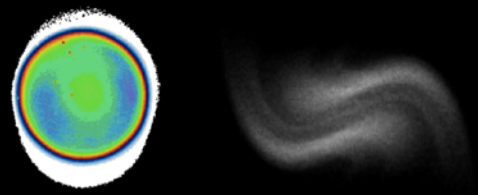
SPS - CERN

Sample applications

Injectors/linacs

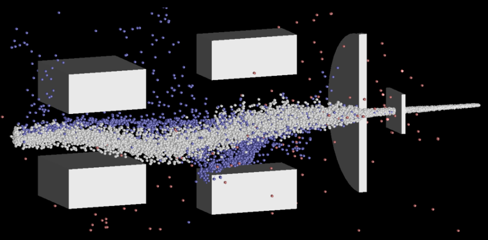


Beam dynamics in rings



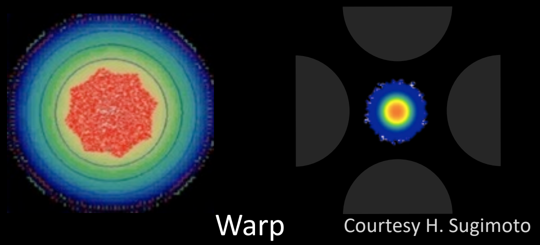
UMER

Multi-charge state beams



LEBT – Project X

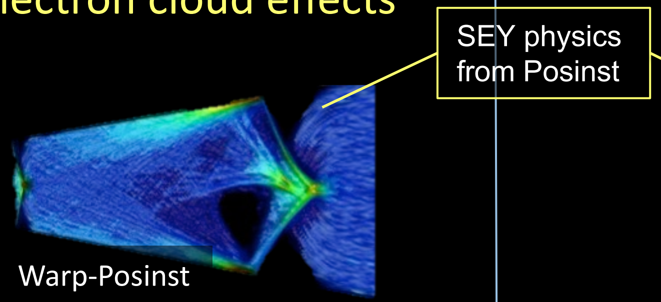
Traps



Alpha anti-H trap

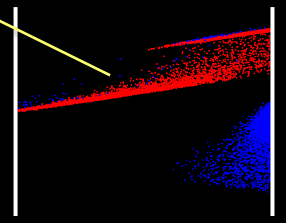
Paul trap

Electron cloud effects



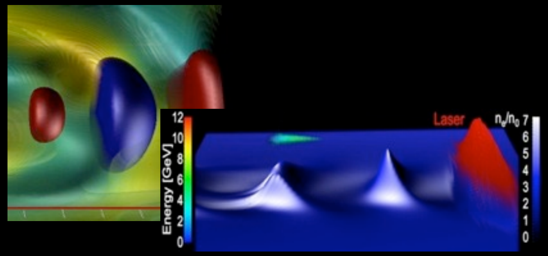
SPS

Multi-pacting



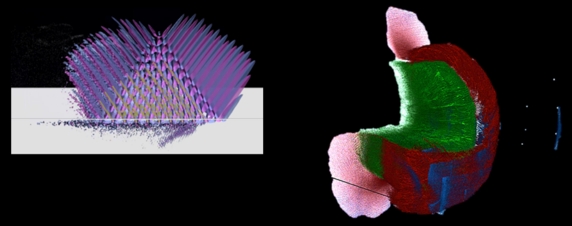
"Ping-Pong" effect

Plasma acceleration



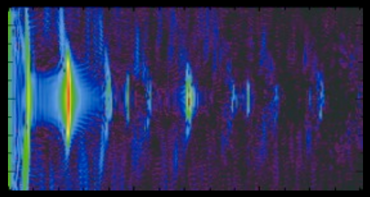
BELLA, …

Laser/beam plasma interactions



Plasma mirrors, ion acceleration, plasma lens, …

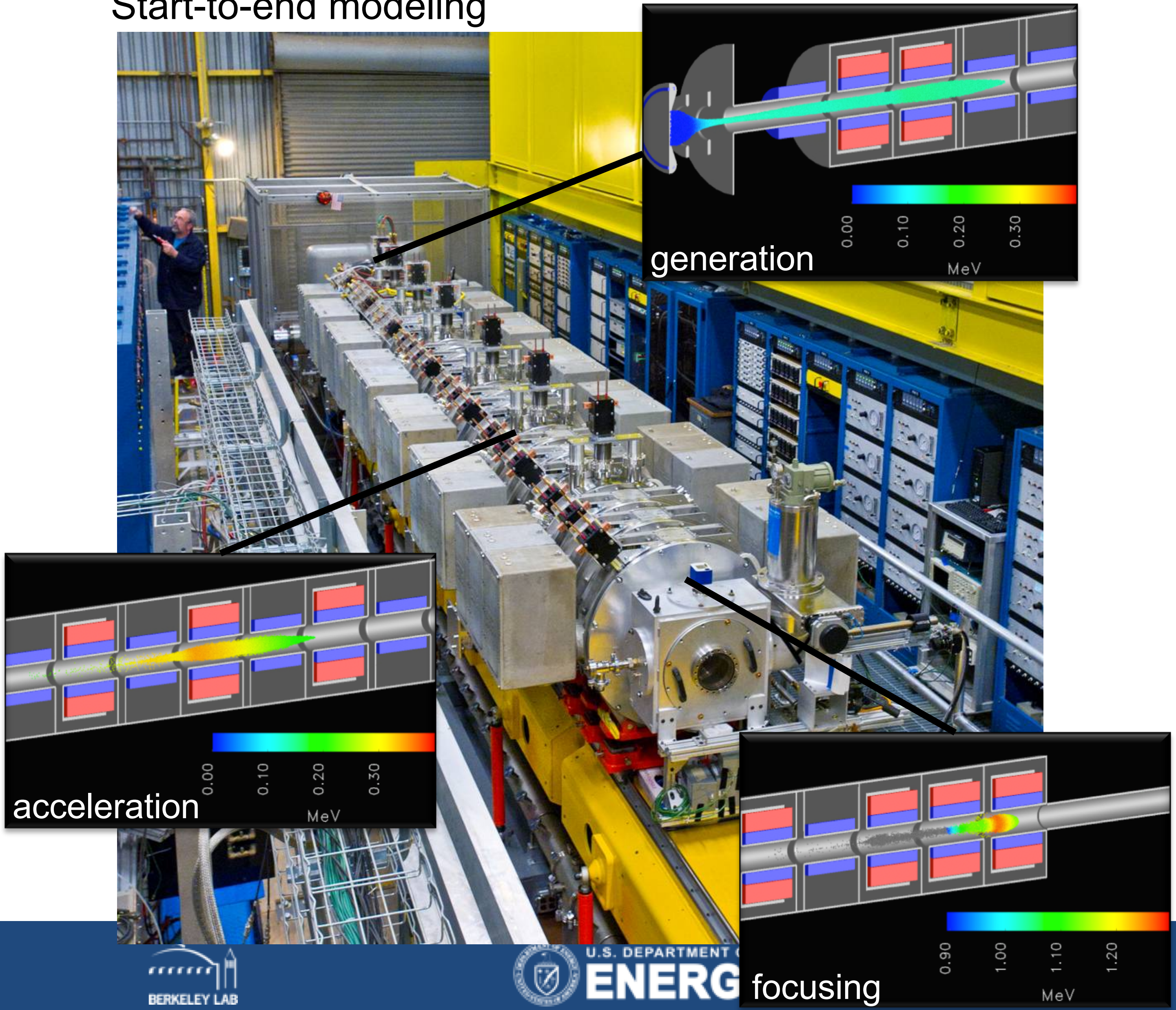
Free Electron Lasers

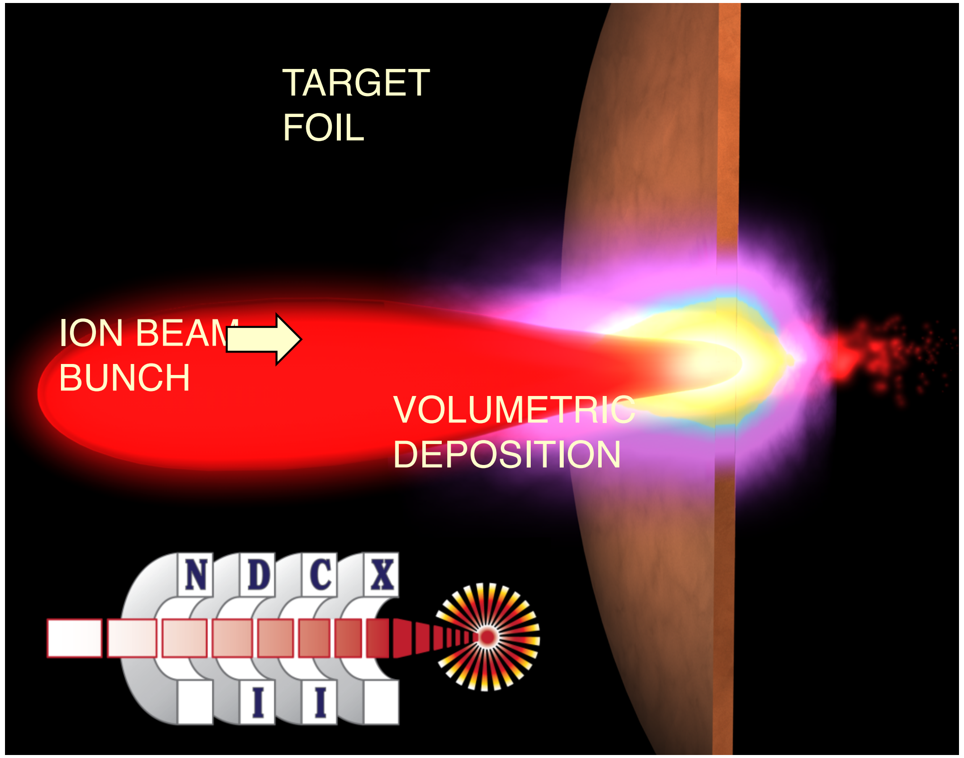


Start-to-end modeling of
Neutralized Drift Compression Experiment (NDCX)

Neutralized Drift Compression Experiment-II (NDCX-II) at LBNL

Start-to-end modeling



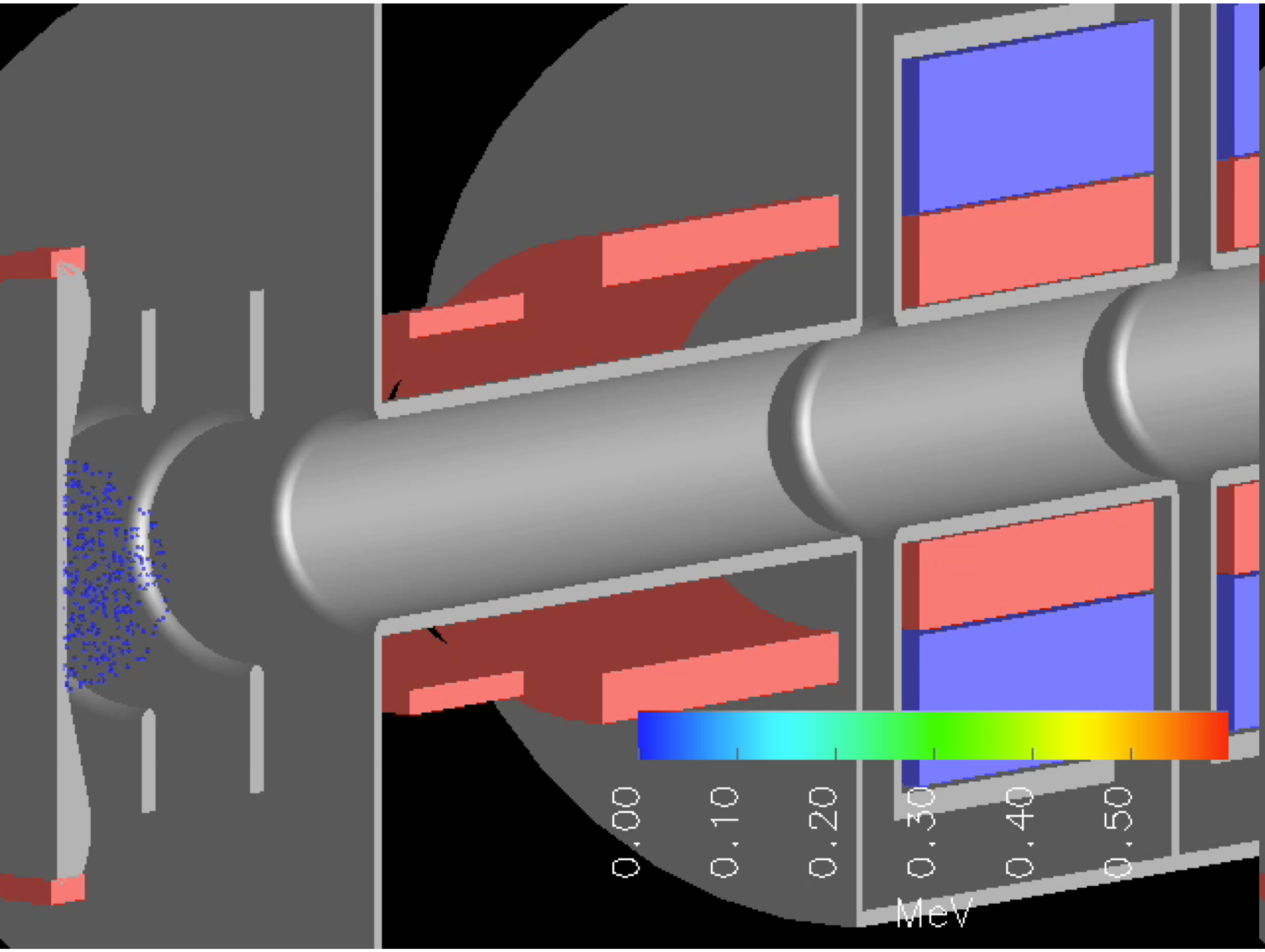


A user facility for studies of:

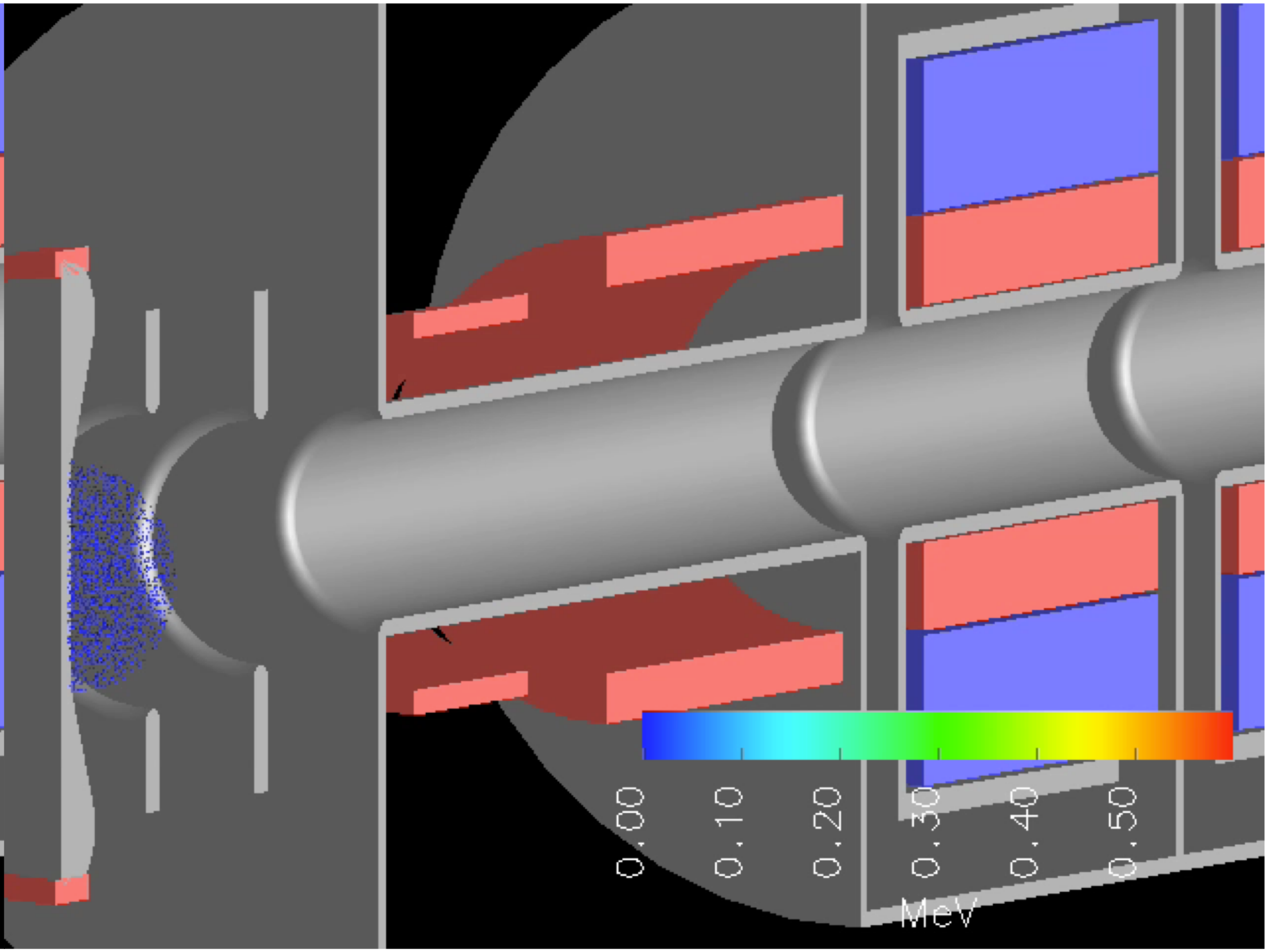
- warm dense matter physics
- heavy-ion-driven target physics
- space-charge-dominated beams

Enables design and tolerance studies

No misalignment

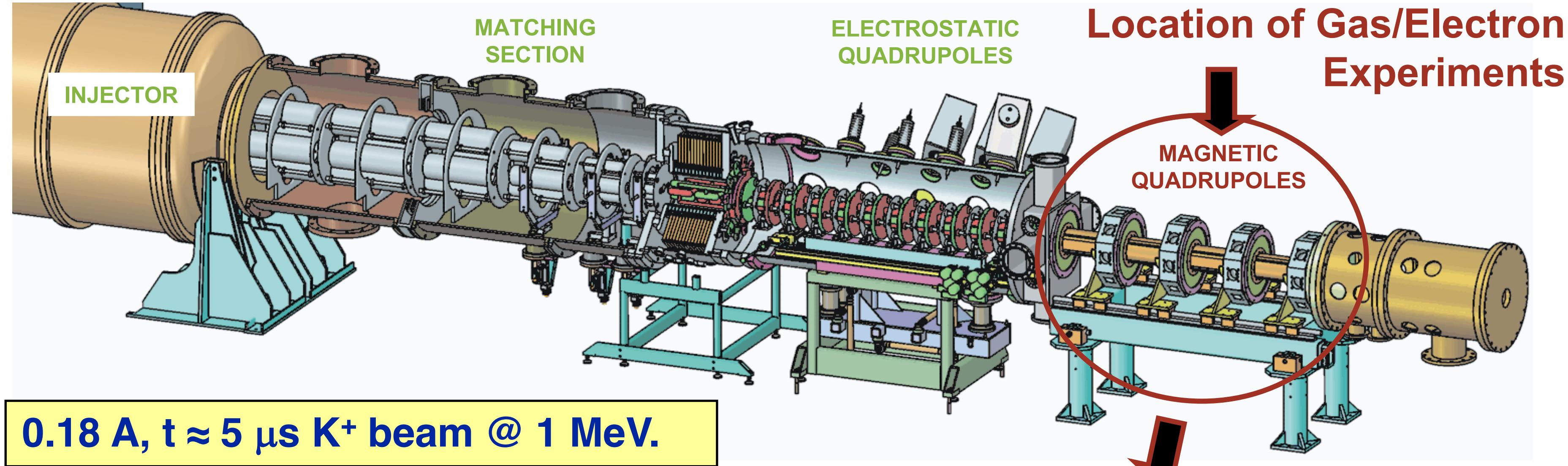


Random misalignments of lattice elements ~2mm

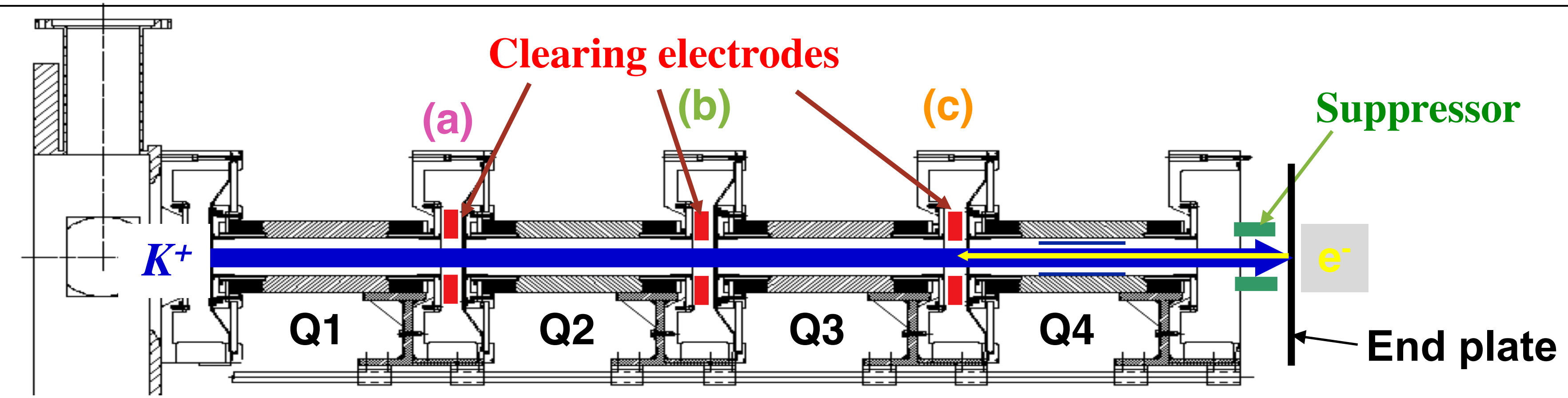


Start-to-end modeling of High-Current Experiment (HCX)

High-current experiment (HCX)

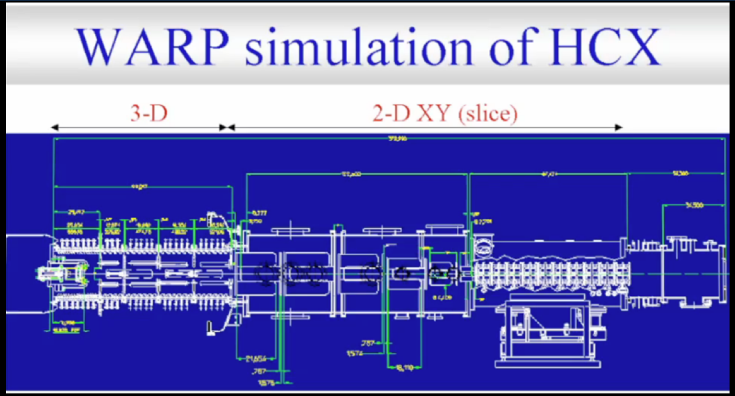


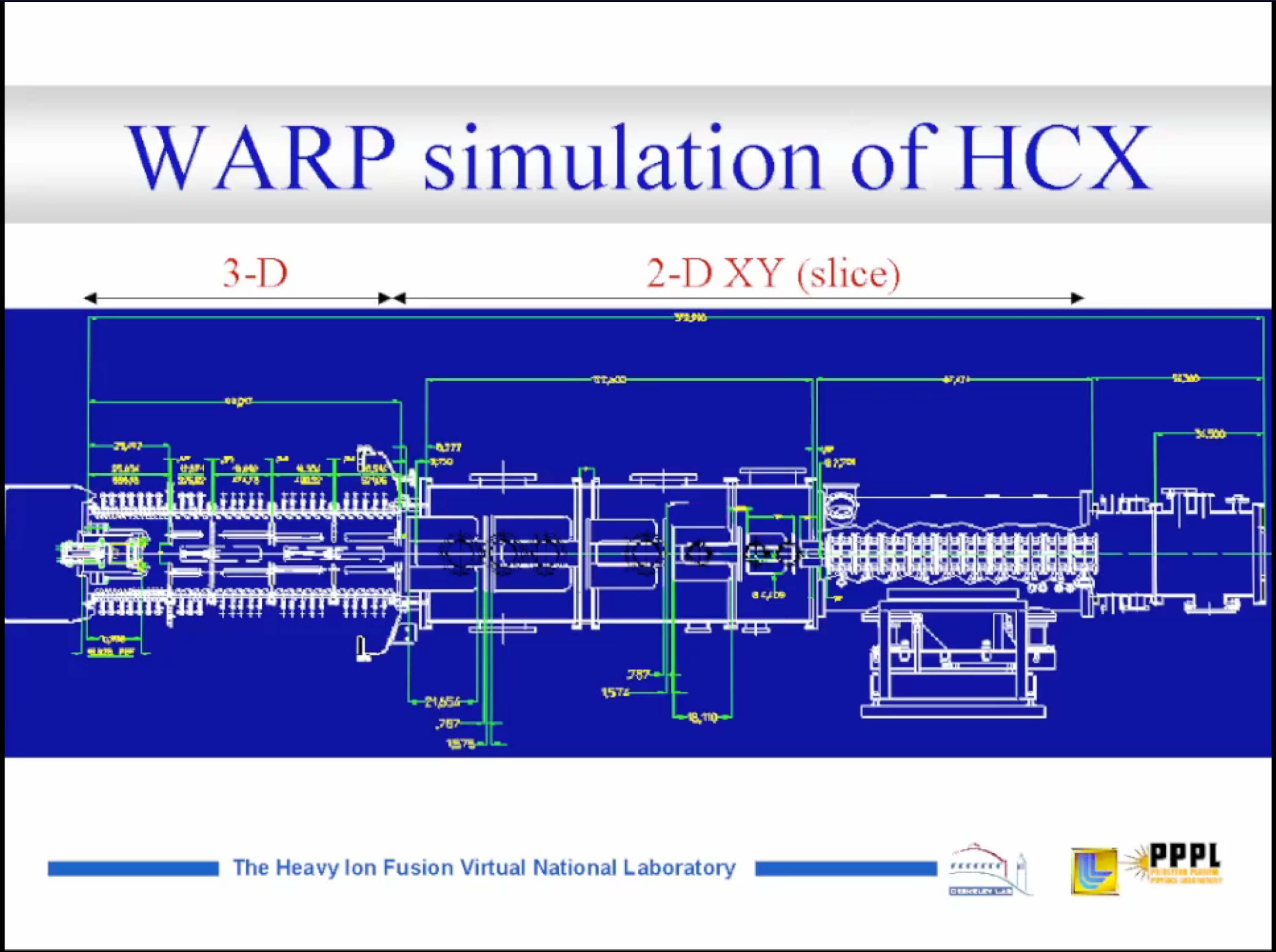
0.18 A, t ≈ 5 μs K⁺ beam @ 1 MeV.



Short experiment => need to deliberately amplify electron effects:
let beam hit end-plate to generate copious electrons which propagate upstream.

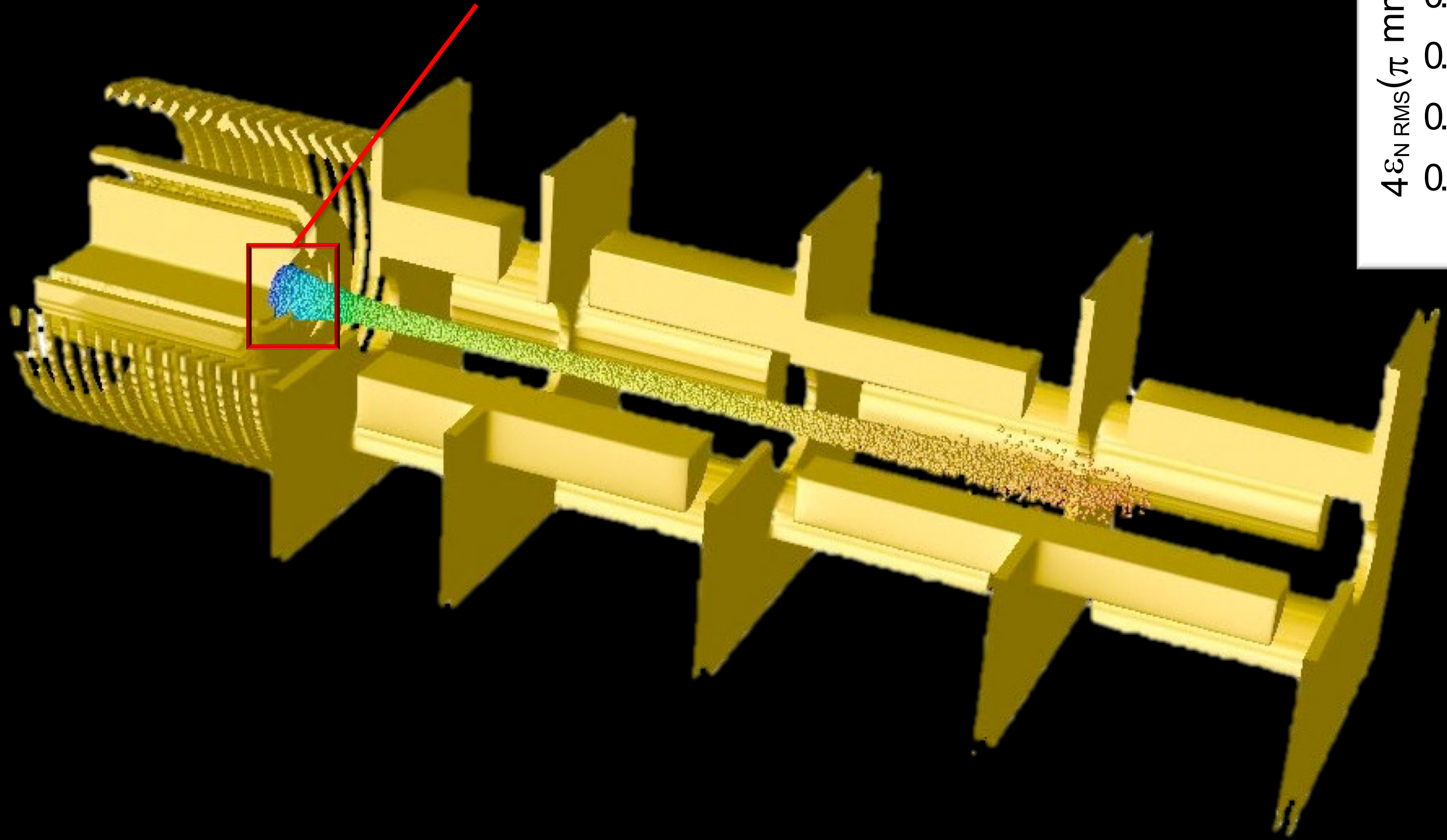
Modeling of injector in 3-D and accelerator in 2-D slice



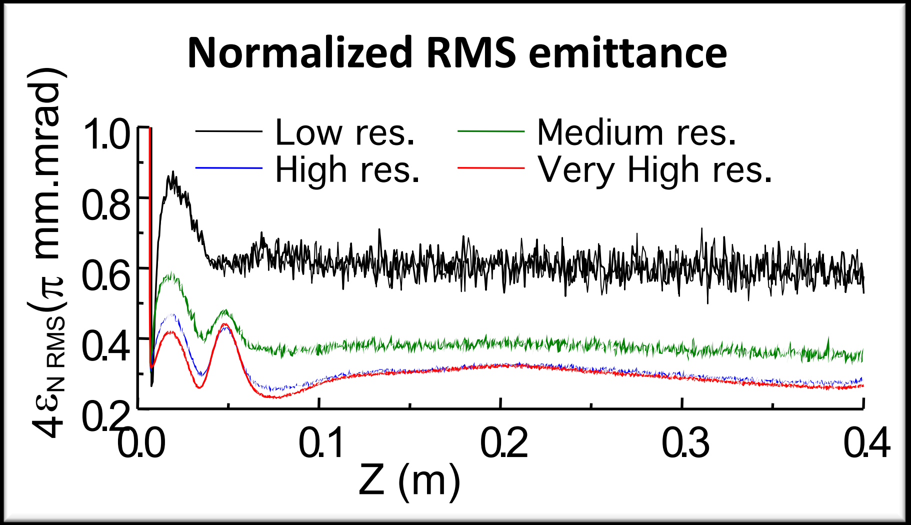


Modeling of High-Current Experiment (HCX)

Very high resolution was needed to model the source of high-intensity beam with sharp edges.

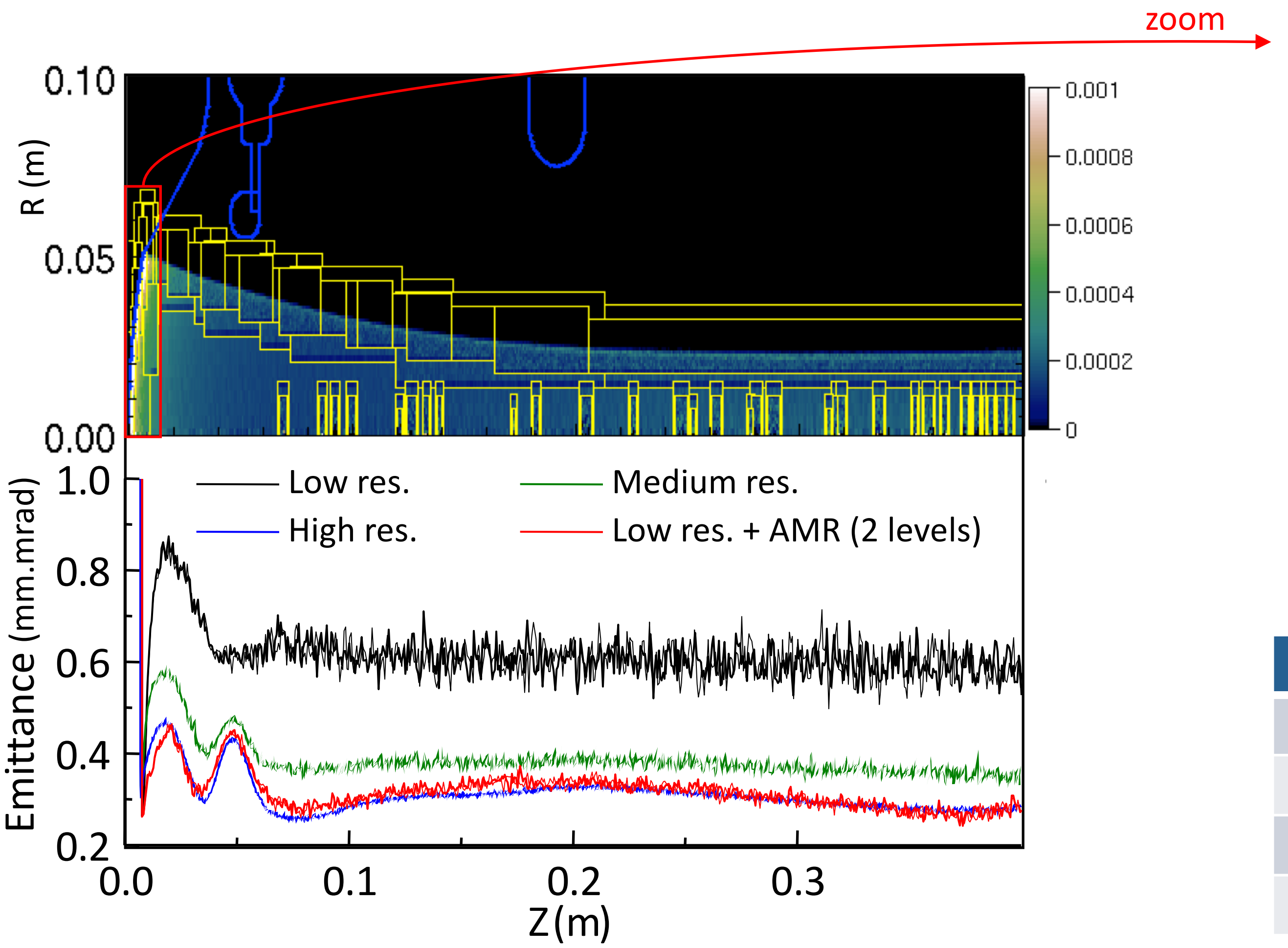


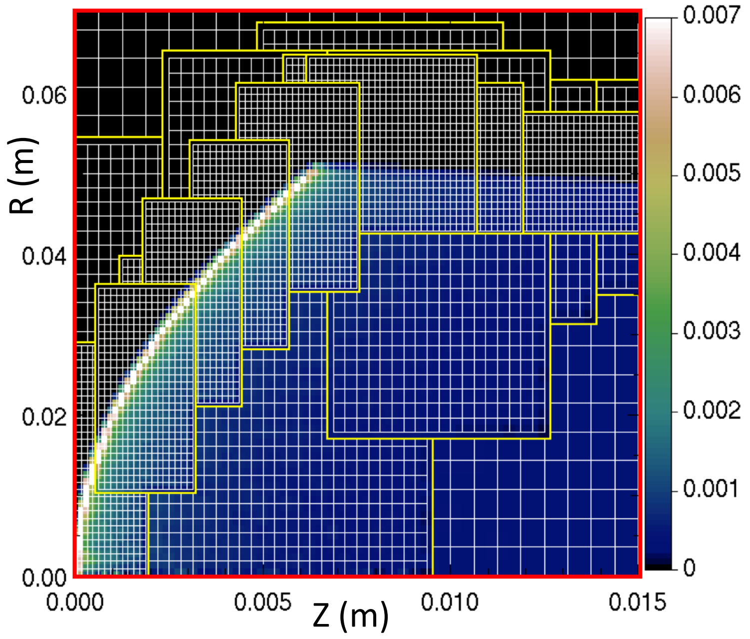
Convergence study for emitter in RZ geometry



Run	Grid size	# particles
Low res.	56 x 640	~1M
Medium res.	112 x 1280	~4M
High res.	224 x 2560	~16M
Very High res.	448 x 5120	~64M

Adaptive Mesh Refinement can lower the computational cost

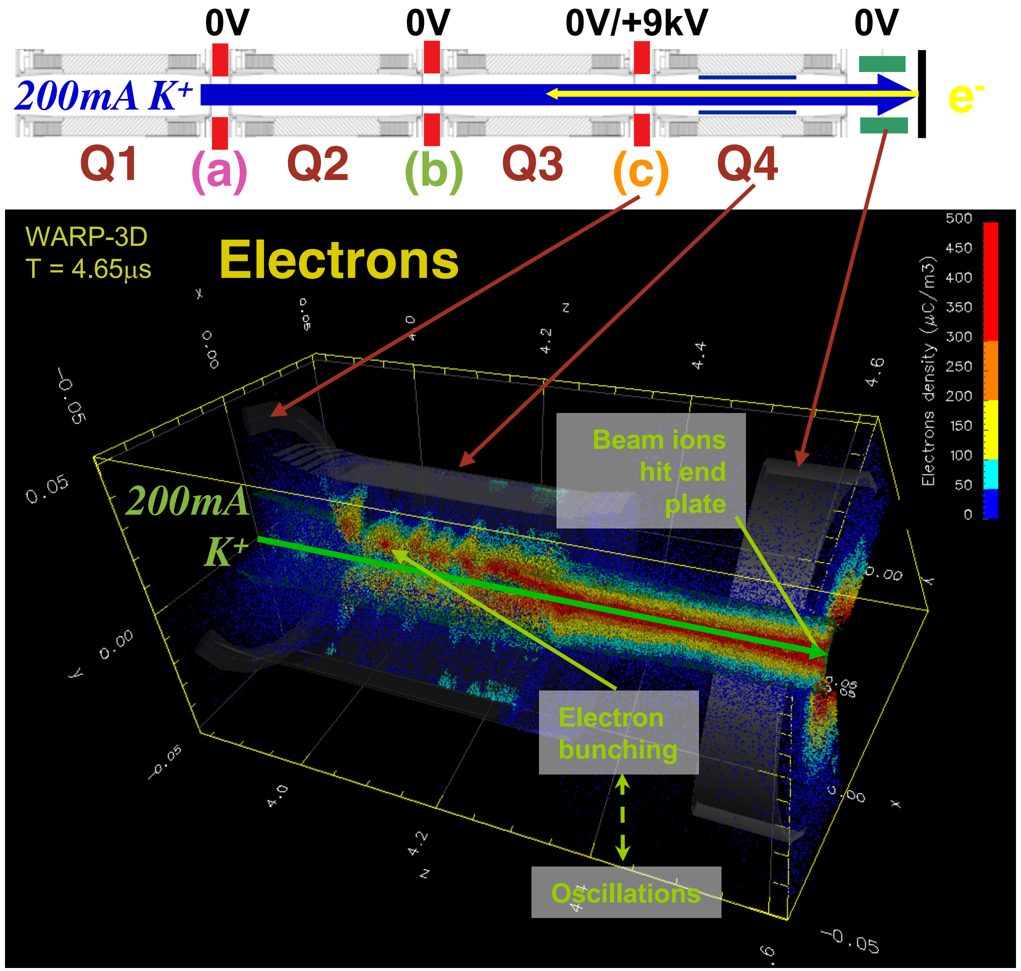


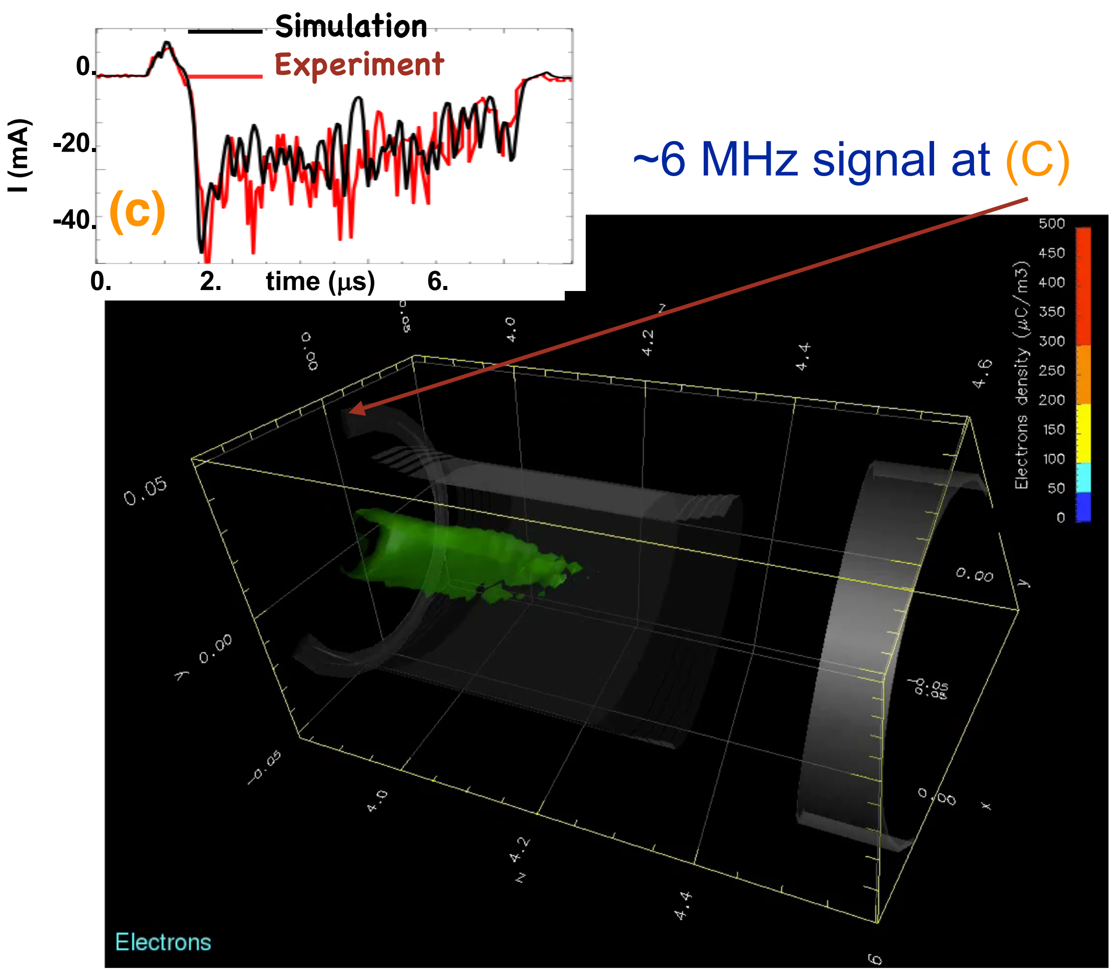


Run	Grid size	# particles
Low res.	56 x 640	~1M
Medium res.	112 x 1280	~4M
High res.	224 x 2560	~16M
Low res. + AMR	56 x 640	~1M

Speedup ~x10

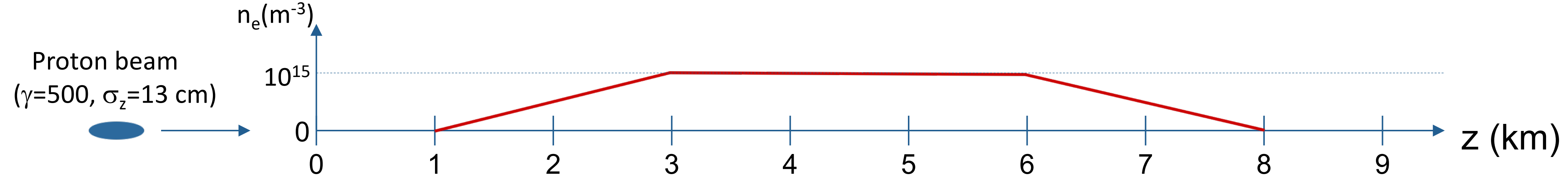
Simulations predicted electron bunching oscillations on HCX when the ion beam was deliberately directed onto the end wall

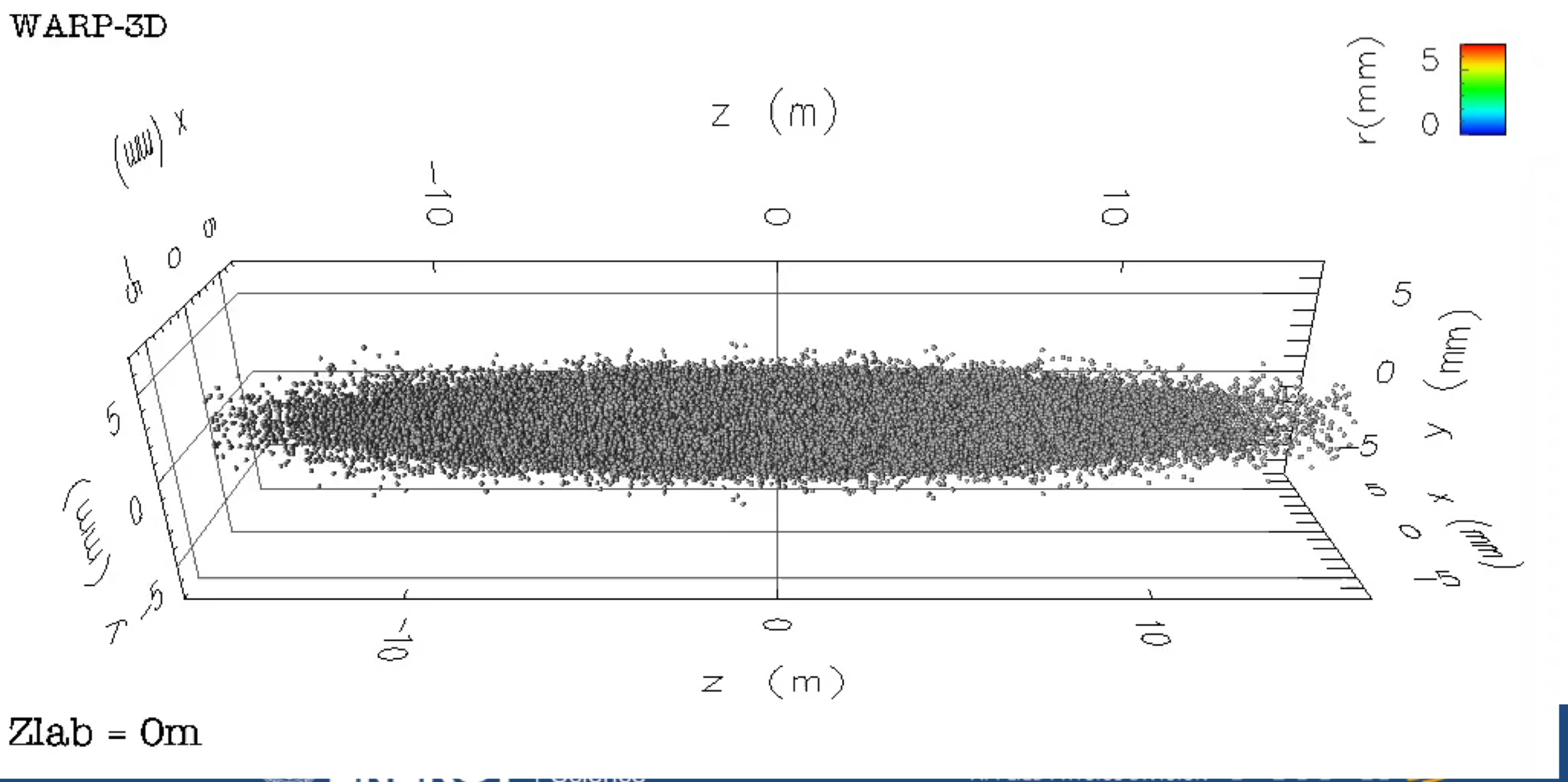




Modeling of electron-cloud-driven two-stream instability

Relativistic proton beam crossing electron cloud



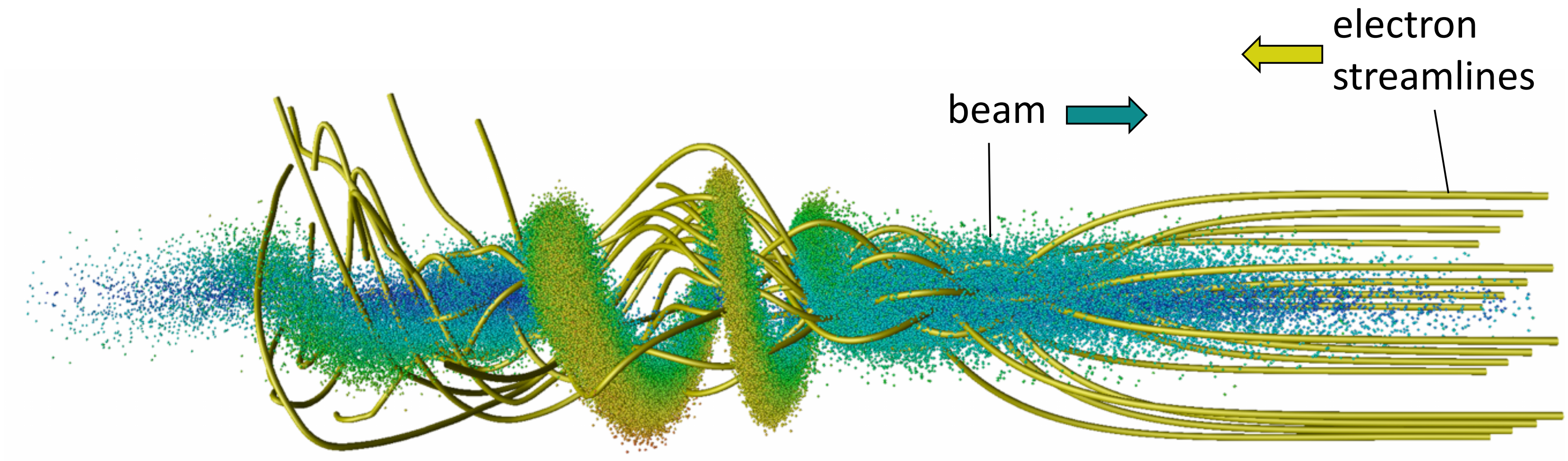


Modeling of two-stream instability is expensive

Need to follow:

- short (σ_z=13 cm) and stiff (γ=500) proton beam for kilometer
- mobile background electrons reacting in fraction of beam length

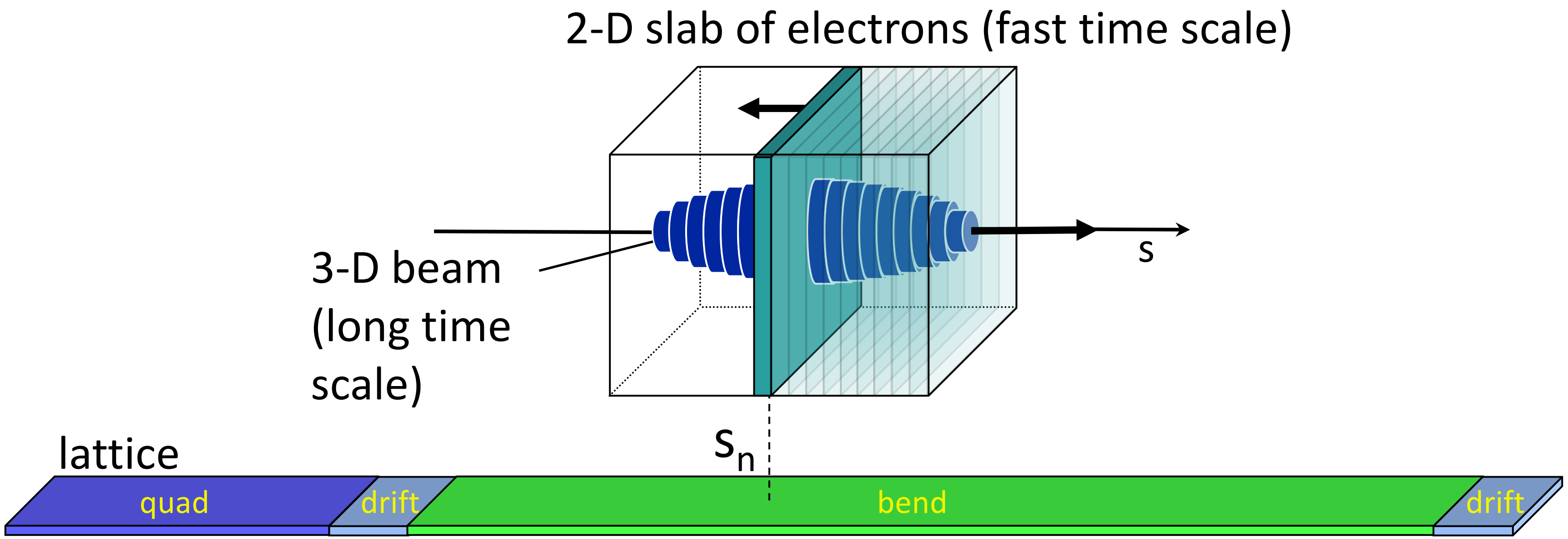
➔ many small time steps



Two solutions:

- separate treatment of slow (beam) and fast (electrons) components ➔ quasistatic approximation
- solve in a Lorentz boosted frame which matches beam & electrons time scales

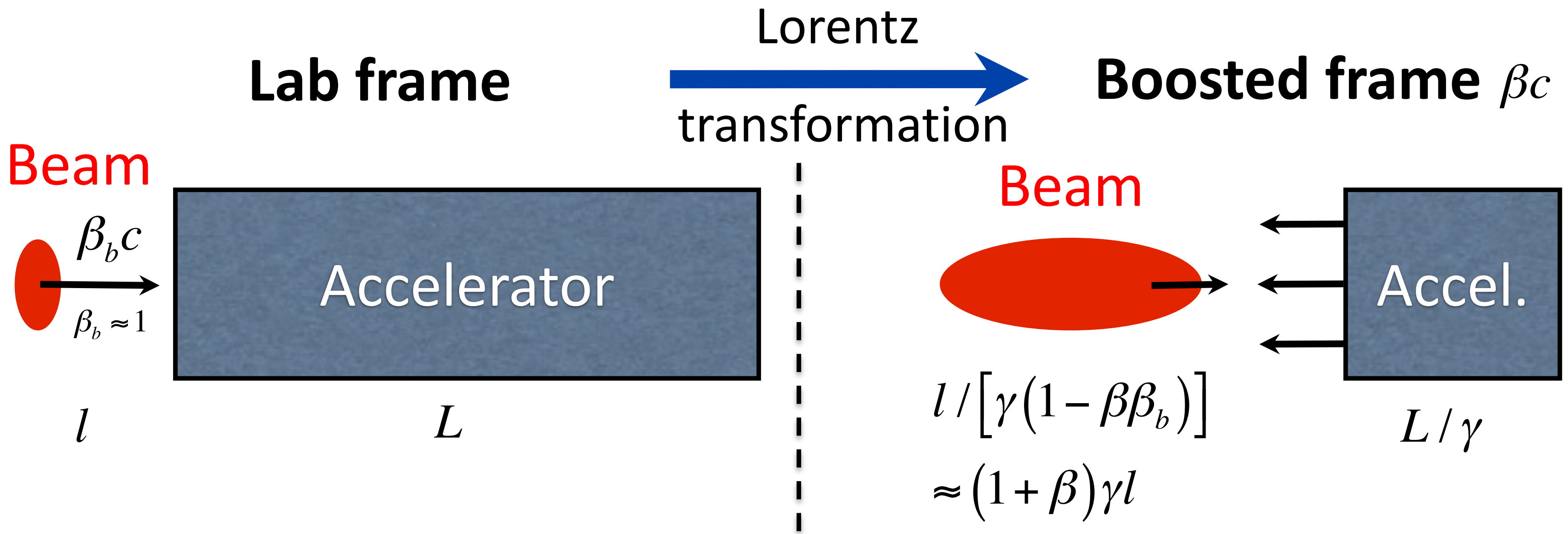
Quasistatic approximation approach



repeat

1. 2-D slab of electrons is stepped backward (with small time steps) through the beam field and its self-field (solving 2-D Poisson at each step),
2. 2-D electron fields are stacked in a 3-D array and added to beam self-field,
3. 3-D field is used to kick the 3-D beam,
4. 3-D beam is pushed to next station with large time steps,
5. Solve Poisson for 3-D beam self-field.

Optimal Lorentz boosted frame approach



Many time steps needed to follow short stiff high-energy beam into long accelerator filled with fast reacting electron clouds.

Much less time steps needed to follow long low-energy beam into shorter accelerator filled with stiffer electron clouds.

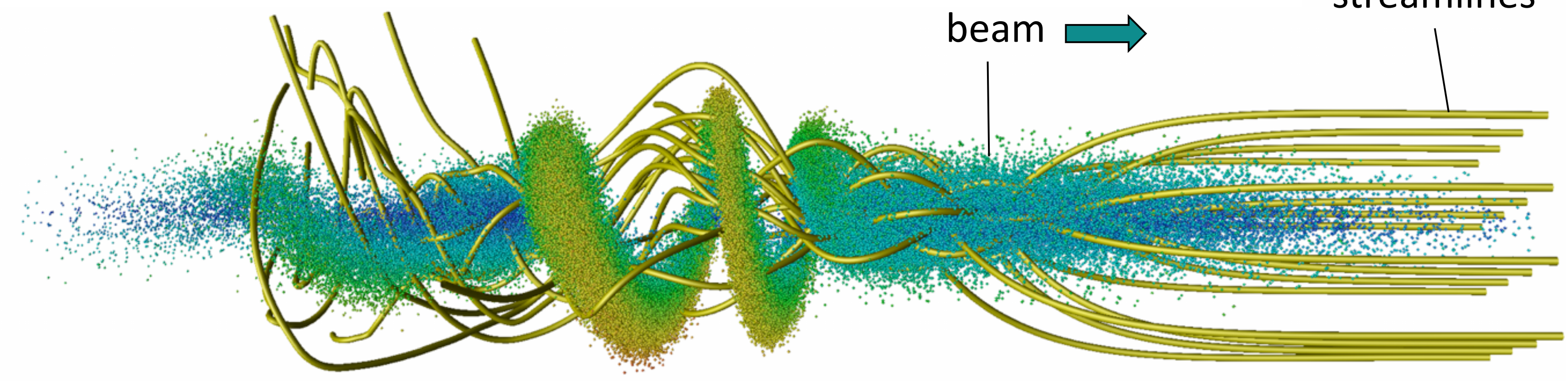
Number of time steps divided by $(1+\beta)\gamma^2$

With high γ, orders of magnitude speedups are possible.

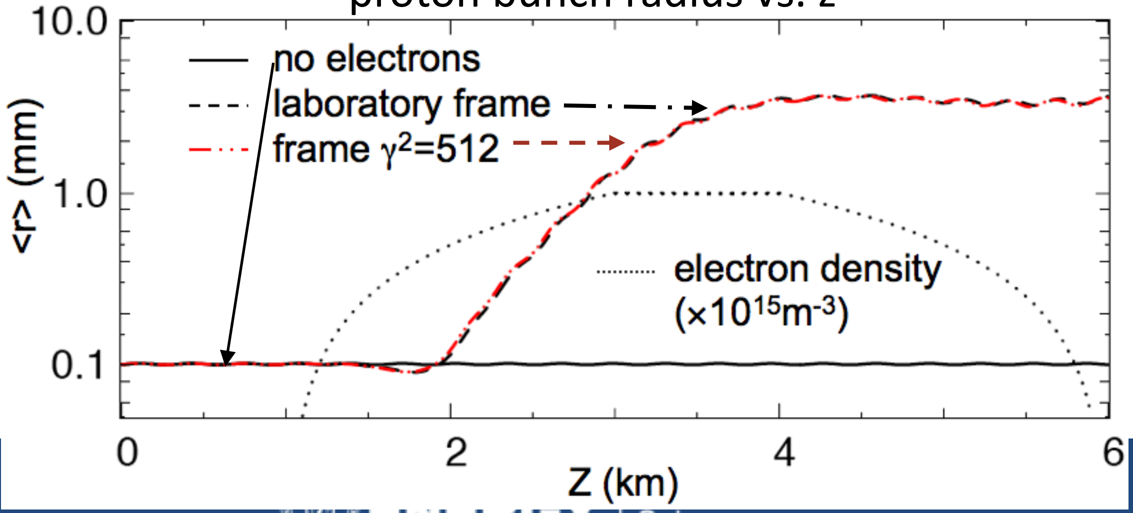
Application to modeling of two-stream instability

Calculation of e-cloud induced instability of a proton bunch

- Proton beam: γ=500, σ_z=13 cm
- L=6 km, continuous focusing



proton bunch radius vs. z

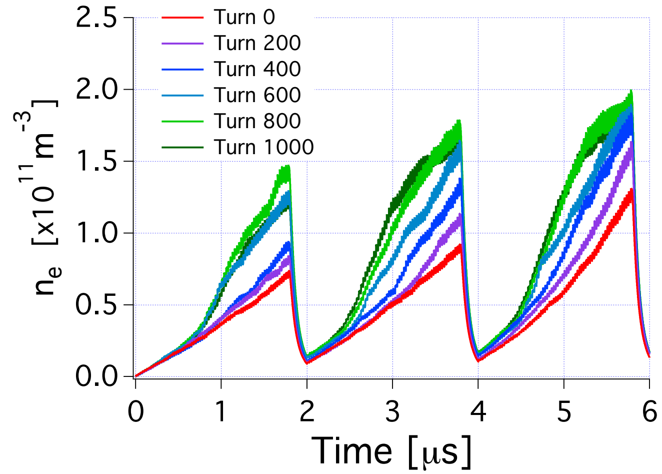


CPU time (on 8 cores in 2006):

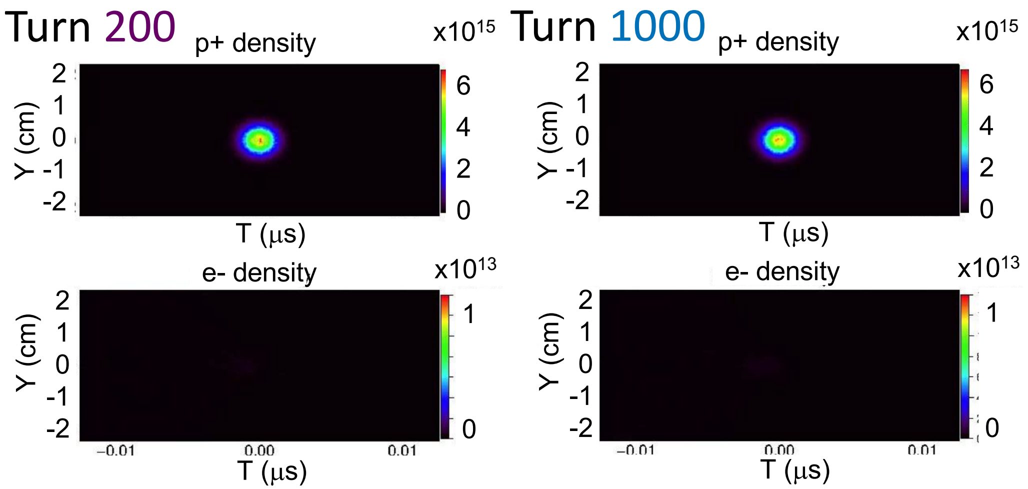
- **lab frame: >2 weeks**
- **frame with γ^2=512: <30 min**

Speedup x1000

Warp+Posinst used to study e-cloud buildup and effect on 3 batches of 72 bunches in CERN SPS (>10k cores, 1 MPI group/bunch)

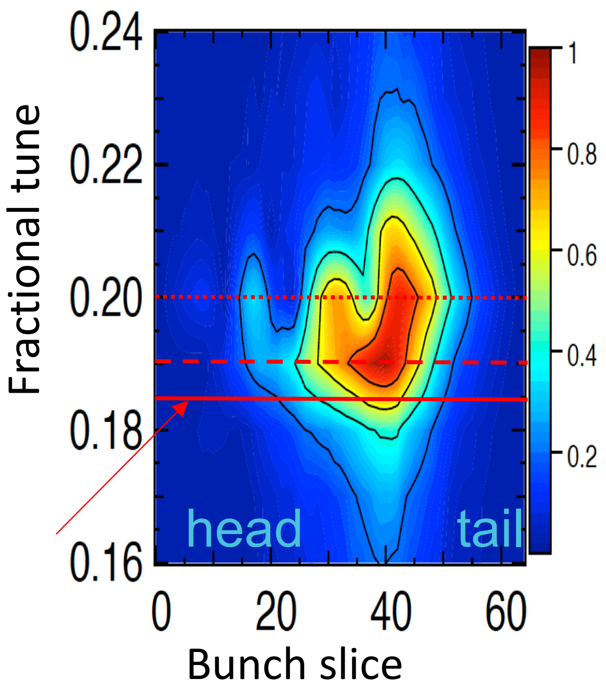


Bunch-by-bunch snapshosts



Warp-Posinst[2]

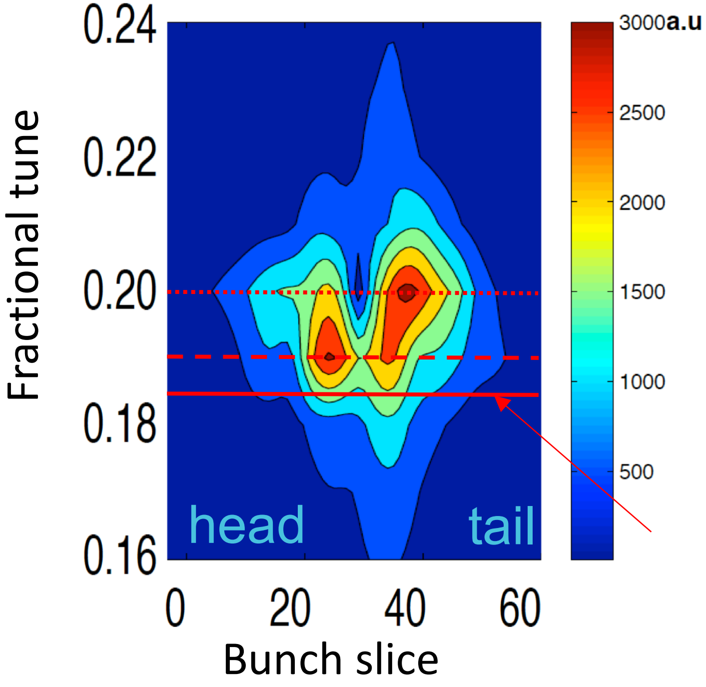
Bunch 29, Turn 100-200



[2]J.-L. Vay, et al, *Ecloud10 Proc.*, (2010)

Experiment[1]

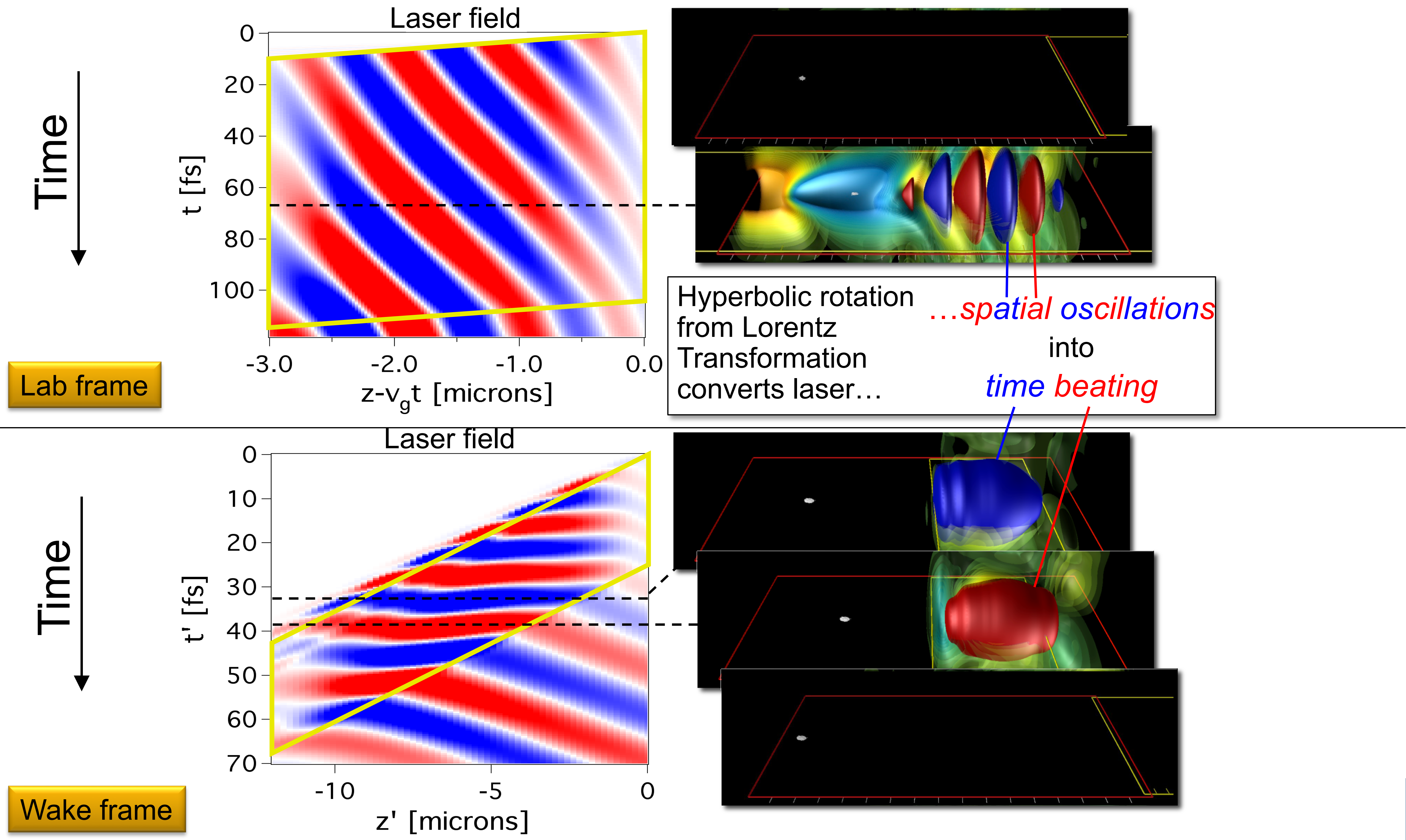
Bunch 119, Turn 100-200



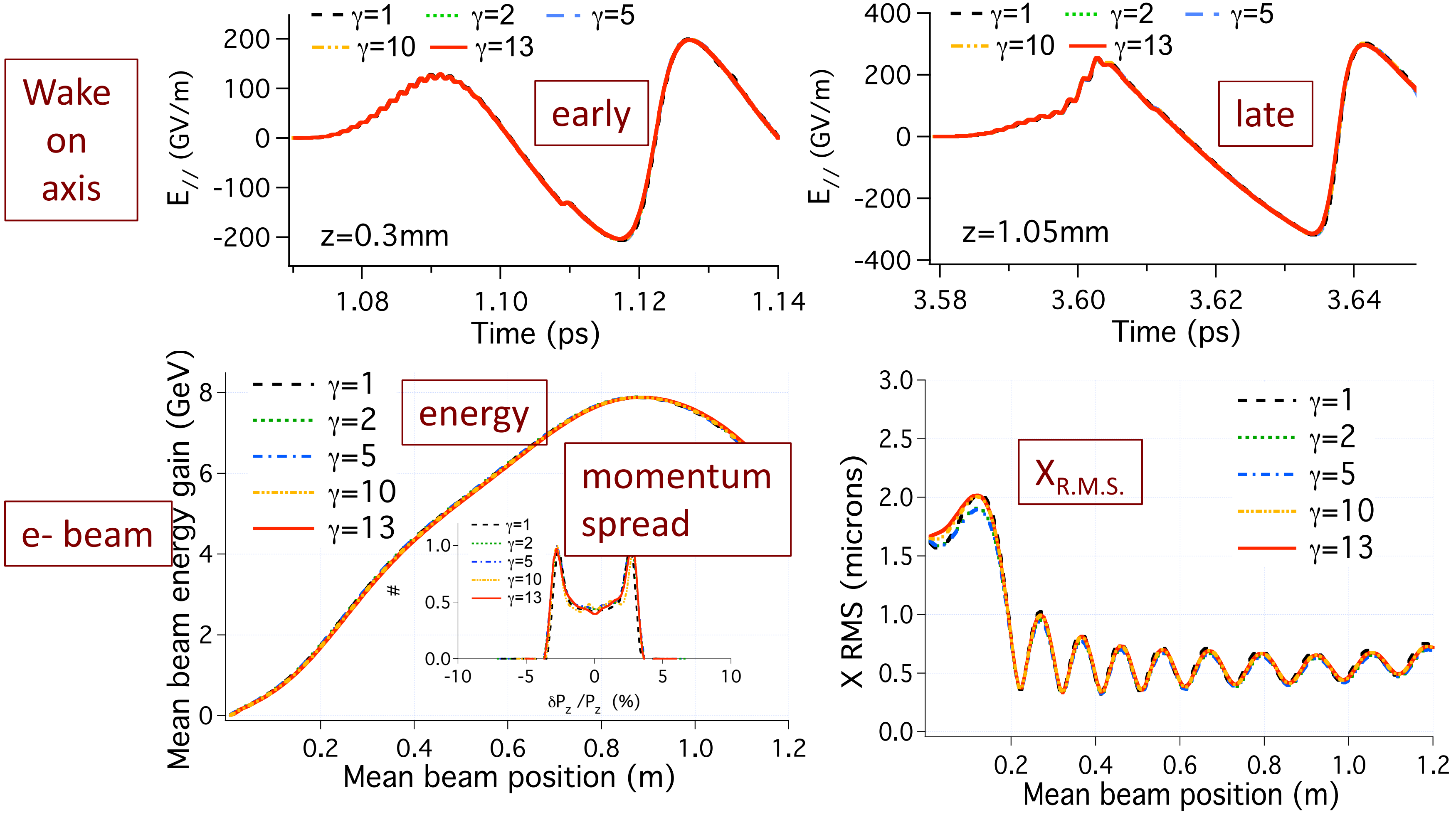
[1]J. Fox, et al, *IPAC10 Proc.*, p. 2806 (2011)

J.-L. Vay, et al, *IPAC12 Proc.*, (2012) TUEPPB006

Modeling of plasma-based accelerators

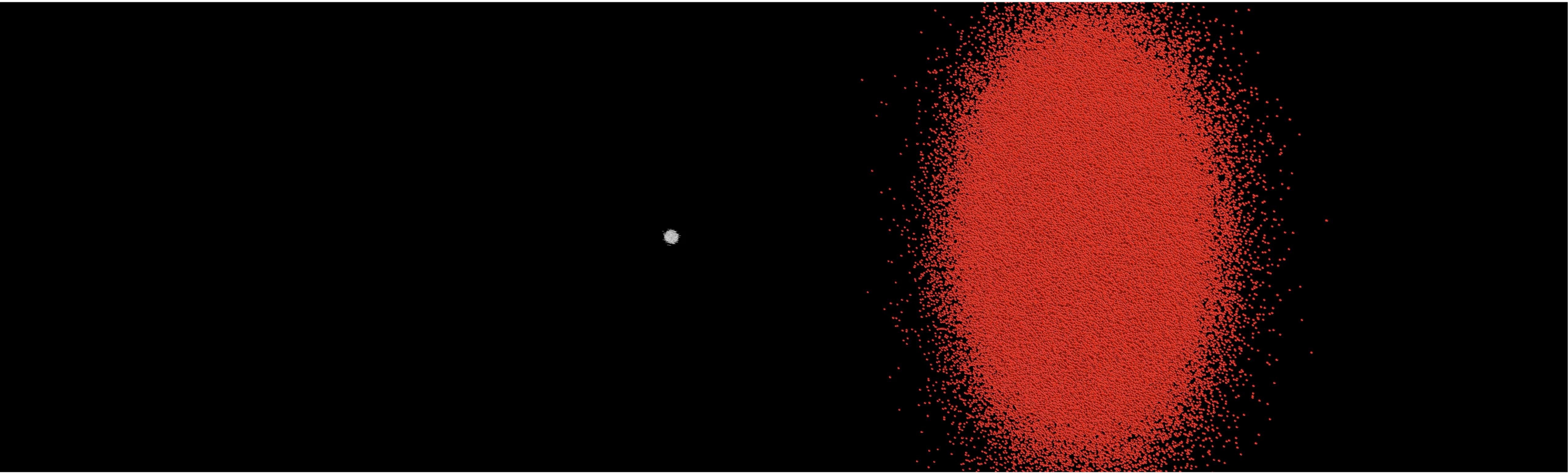

Time
Laser field
t [fs]
0
20
40
60
80
100
-3.0
-2.0
-1.0
0.0
z-v_g t [microns]
Lab frame
Hyperbolic rotation from Lorentz Transformation converts laser…
…spatial oscillations
into
time beating
Time
Laser field
t' [fs]
0
10
20
30
40
50
60
70
-10
-5
0
z' [microns]
Wake frame

Verified with high precision that converge to same solution for all γ



Warp-3D – a_0=1, n_0=10^{19}cm^{-3} (~100 MeV) scaled to 10^{17}cm^{-3} (~10 GeV).

Can also model PWFA



More examples to be presented by others and during hands-on sessions…