Report on the WARP code/framework/module

J.-L. Vay[1], D. P. Grote[2], R. H. Cohen[2], A. Friedman[2], I. Haber[3], R. Kishek[3]

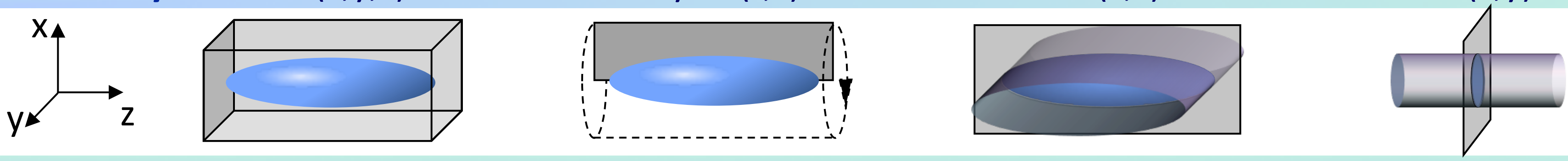
[1]Lawrence Berkeley National Laboratory, CA, USA
[2]Lawrence Livermore National Laboratory, CA, USA
[3]University of Maryland, MD, USA

Space Charge 2013
CERN, Switzerland – April 16-19, 2013

U.S. DEPARTMENT OF ENERGY Office of Science
BERKELEY LAB
SciDAC-III Compass
NERSC

Warp*: PIC modeling of beams, accelerators, plasmas

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

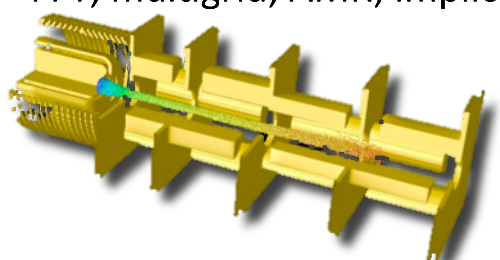


- **Reference frame:** **lab** **moving-window** **Lorentz boosted**

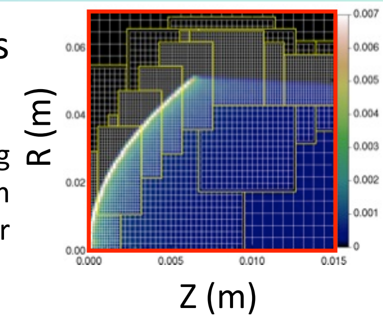
  z z-vt $\gamma(z-vt)$; $\gamma(t-vz/c^2)$

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

Versatile conductor generator accommodates complicated structures



Automatic meshing around ion beam source emitter

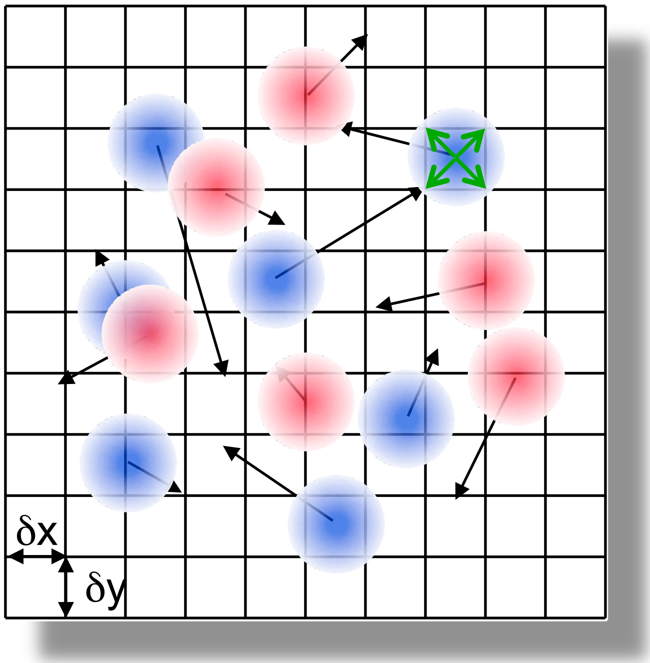


  - Fully electromagnetic – FDTD, PML, MR, new prototype spectral solver

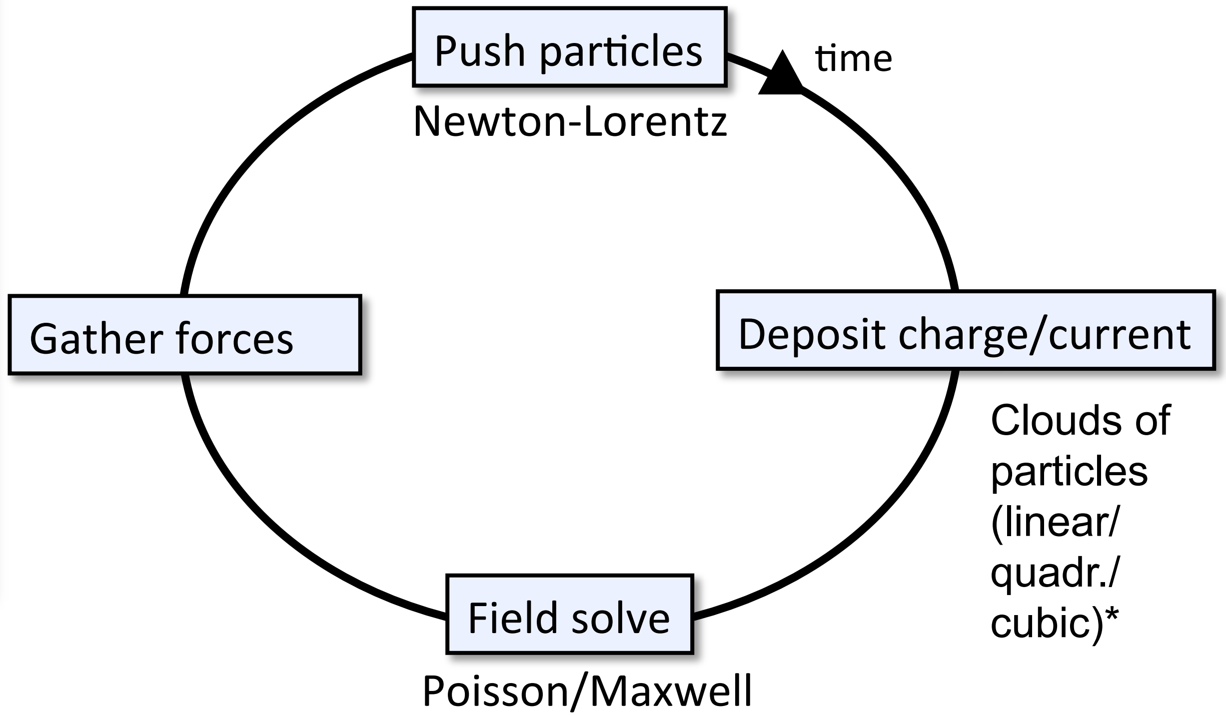
- **Accelerator lattice:** general; non-paraxial; can read MAD files
  - solenoids, dipoles, quads, sextupoles, linear maps, arbitrary fields, acceleration.
- **Particle emission & collisions**
  - particle emission: space charge limited, thermionic, hybrid, arbitrary,
  - secondary e- emission (Posinst), ion-impact electron emission (Txphysics) & gas emission,
  - Monte Carlo collisions: ionization, capture, charge exchange.

*warp.lbl.gov (open source, BSD license)

Warp Particle-In-Cell workflow

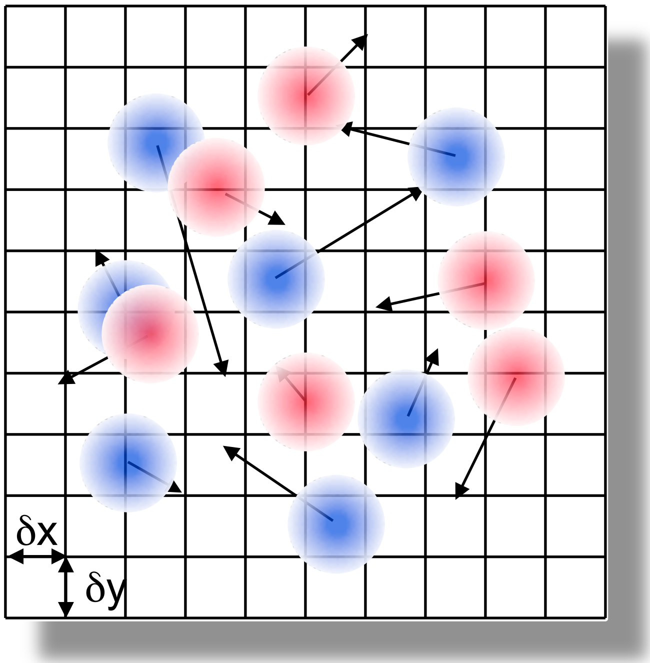


Plasma=collection of interacting charged particles

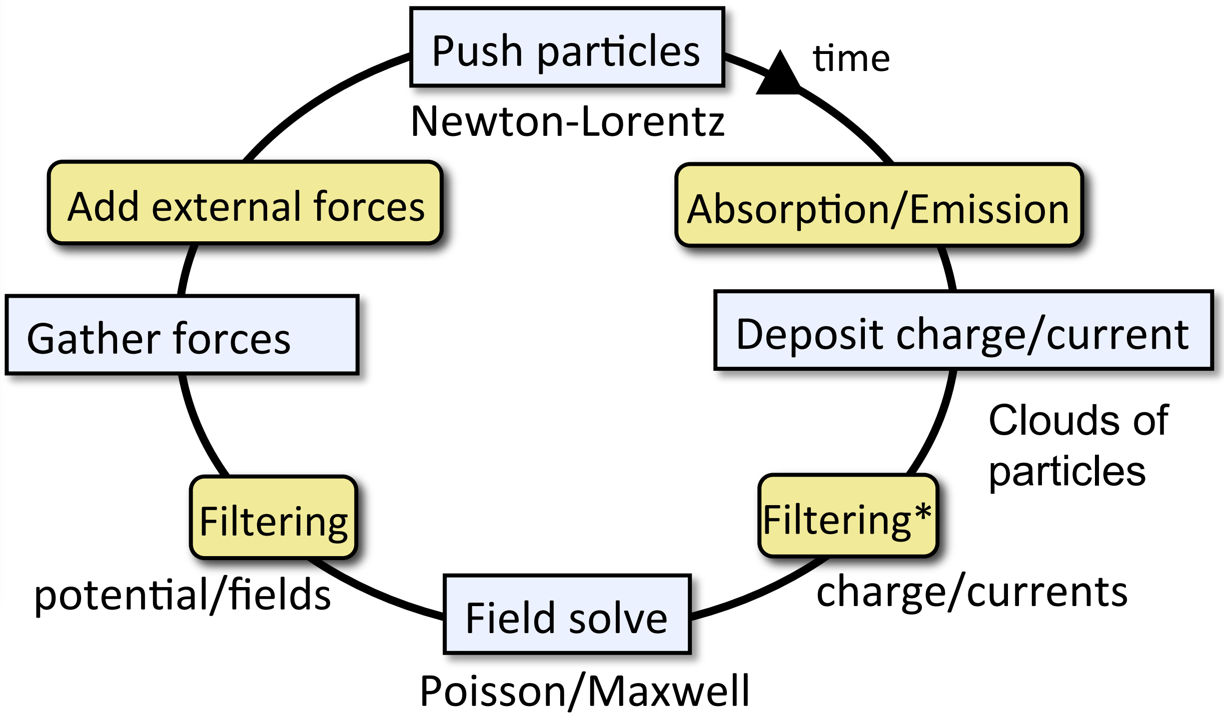


*H. Abe, N. Sakairi, R. Itatani, H. Okuda, High-order spline interpolations in the particle simulation, *Journal of Computational Physics* **63**, 247 (1986)

Warp Particle-In-Cell workflow



Plasma=collection of interacting charged particles



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

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

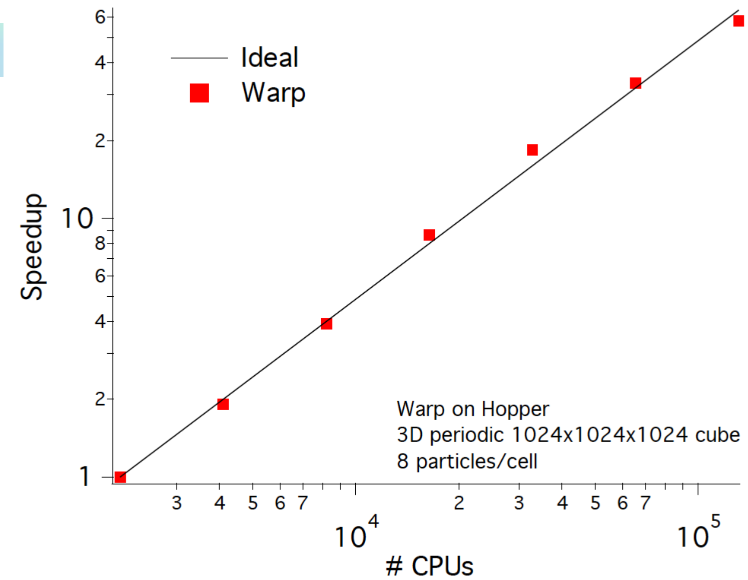
+ external forces (accelerator lattice elements),

*C. Birdsall, A. Langdon, Plasma physics via computer simulation, Adam-Hilger, 1991.
J.-L. Vay et al, *Journal of Computational Physics* **230**, 5908 (2011)

Warp is parallel, combining modern and efficient programming languages

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

Parallel strong scaling of Warp 3D PIC-EM solver on Franklin supercomputer (NERSC)



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

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

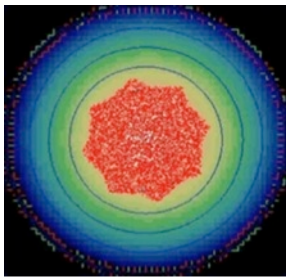
Warp's versatile programmability enables great adaptability

Standard PIC

Laboratory frame

Example:

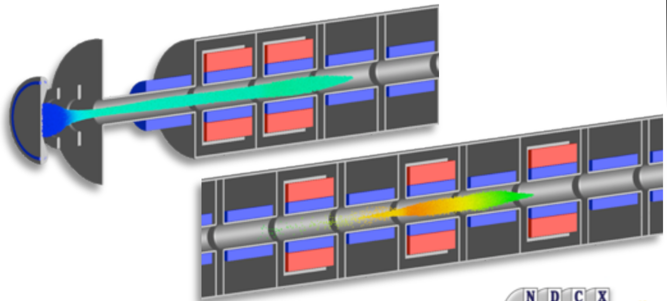
Alpha anti-H trap



Moving window

Example:

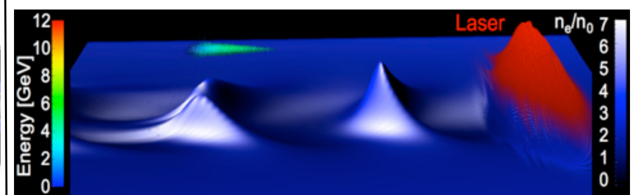
Beam generation and transport



Lorentz Boosted frame

Example:

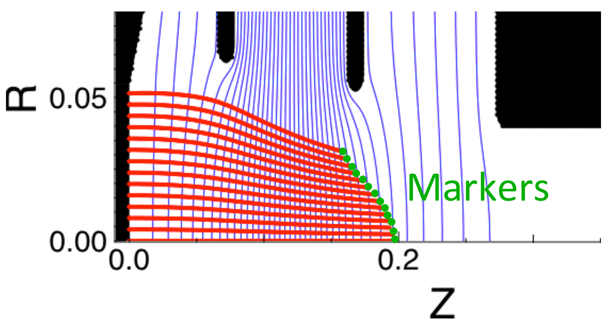
Laser plasma acceleration



Non-standard PIC

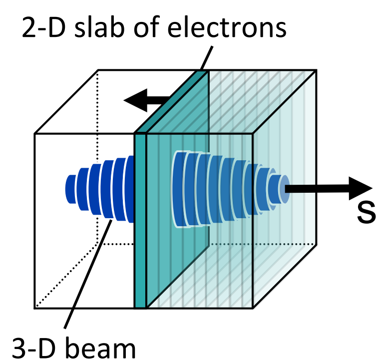
Steady flow

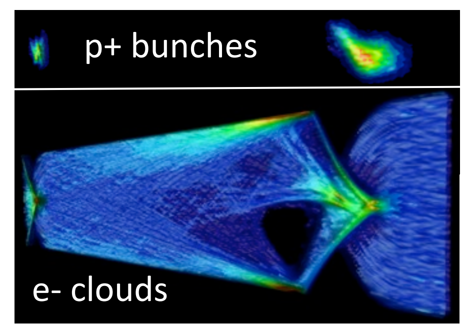
Example: Injector design



Quasi-static

Example: electron cloud studies

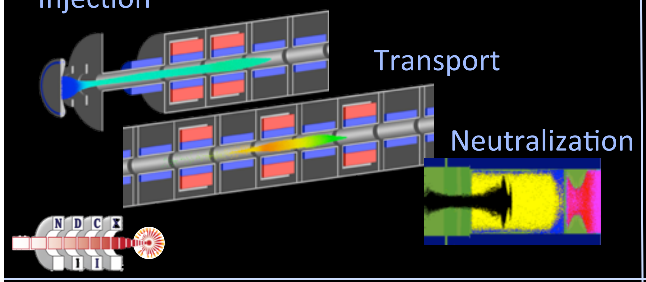




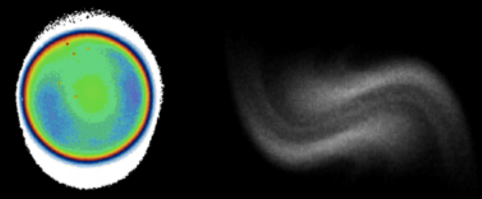
SPS - CERN

Sample applications

Space charge dominated beams

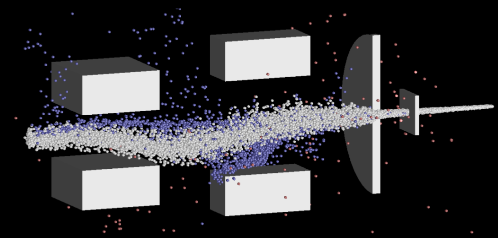


Beam dynamics in rings



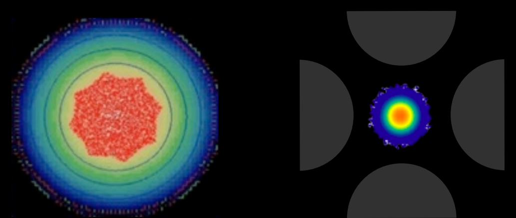
UMER

Multi-charge state beams



LEBT – Project X

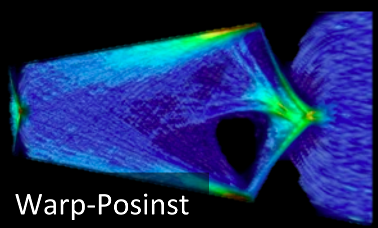
Traps



Alpha anti-H trap

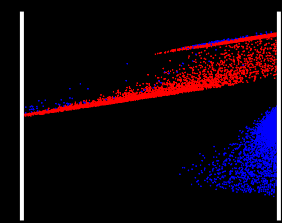
Paul trap

Electron cloud effects



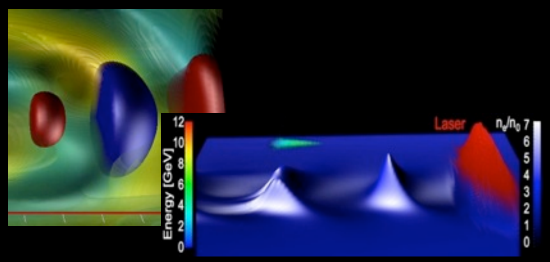
SPS

Multi-pacting



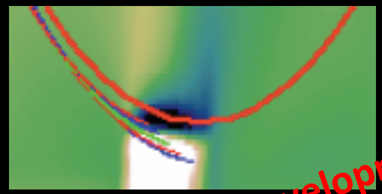
"Ping-Pong" effect

Laser plasma acceleration



BELLA

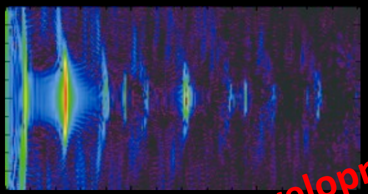
3D Coherent Synchrotron Radiation



In development

NGLS

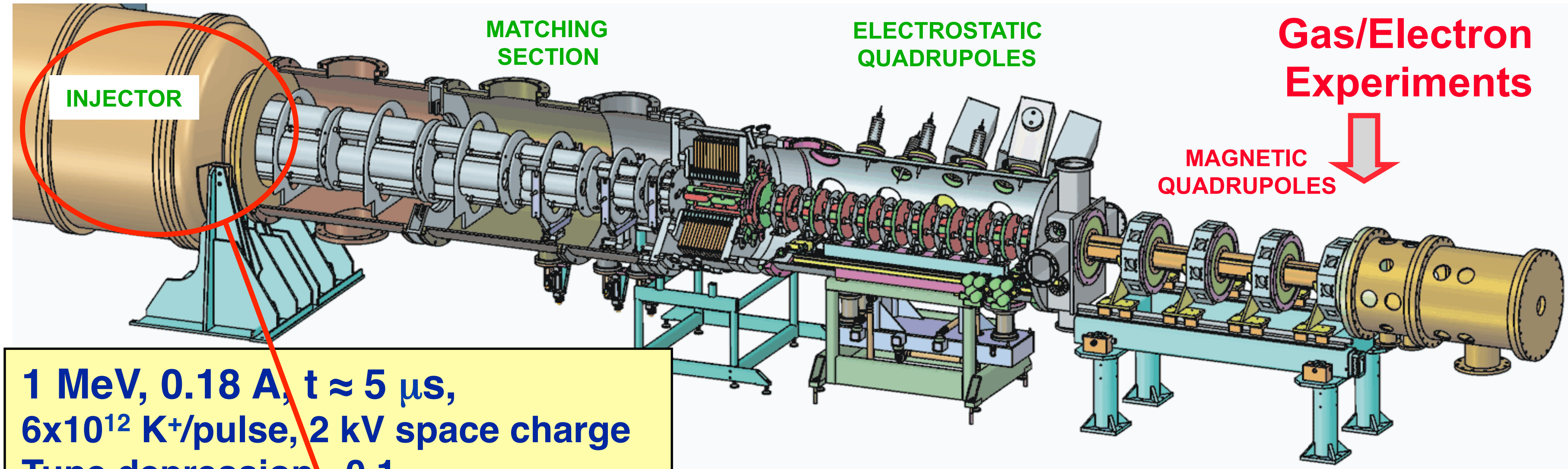
Free Electron Lasers



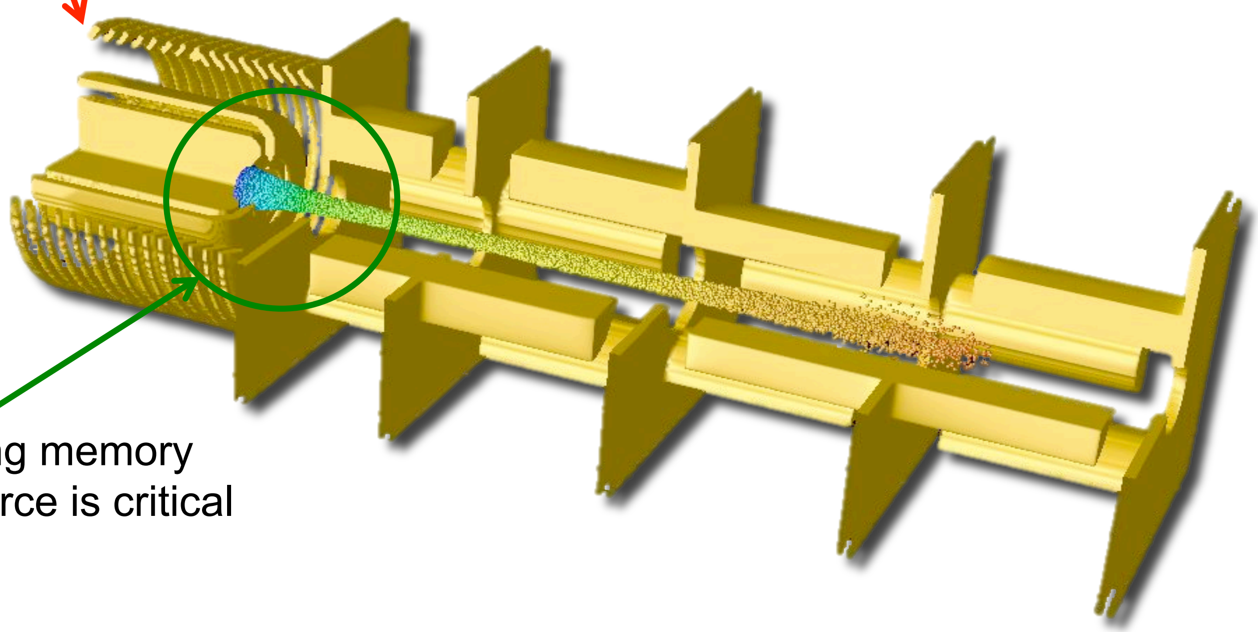
In development

NGLS

Benchmarked against High Current Experiment (HCX)



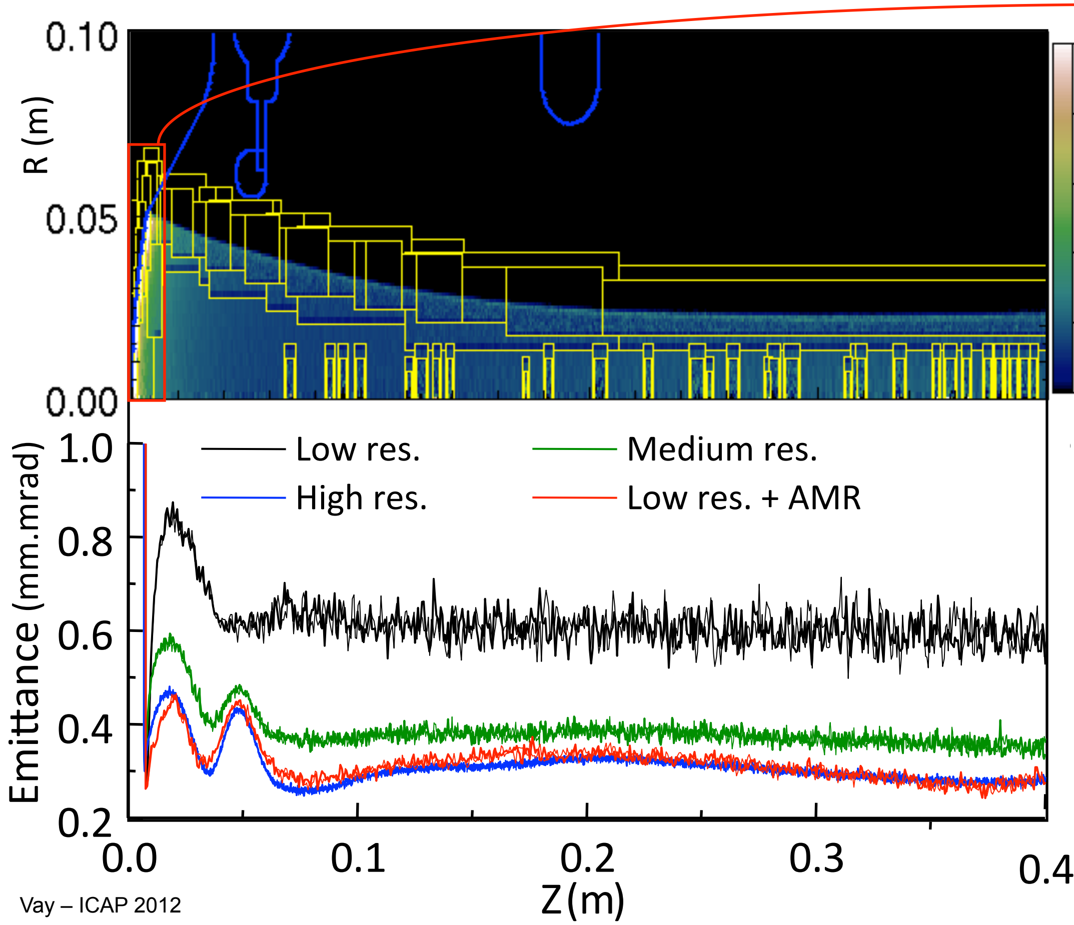
1 MeV, 0.18 A, t ≈ 5 μs,
6x10^{12} K^+/pulse, 2 kV space charge
Tune depression ~0.1



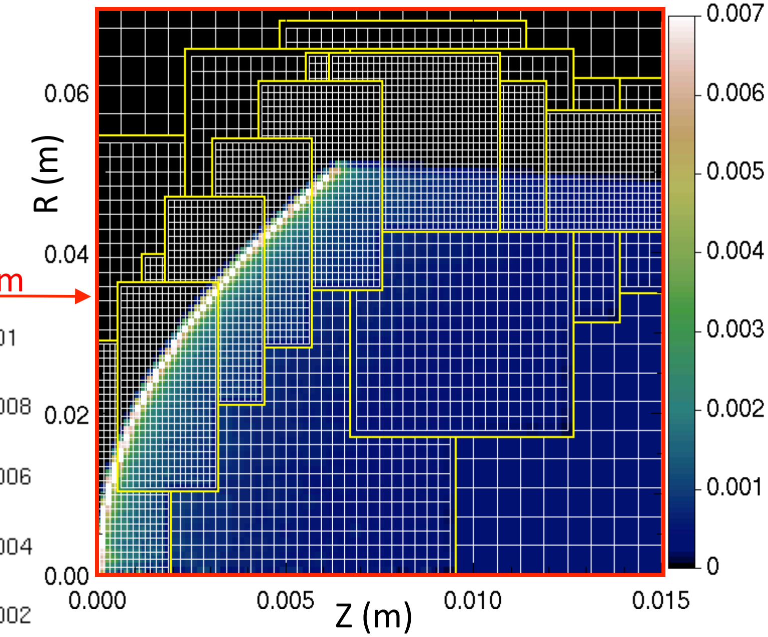
Beams have long memory
Modeling of source is critical

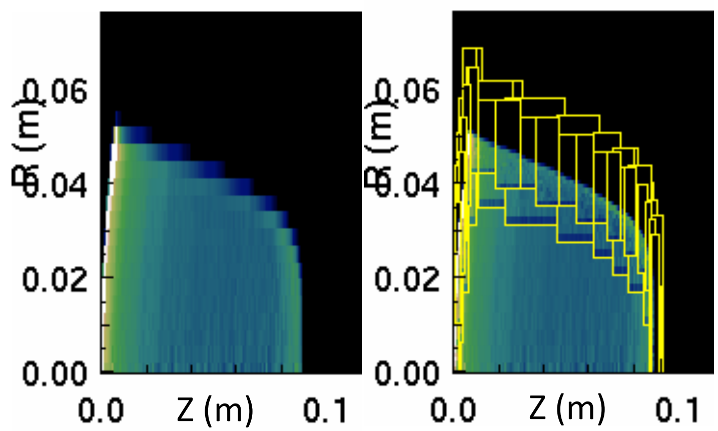
Simulation of ion source using Adaptive Mesh Refinement (AMR)
-- speedup from AMR x10

Run	Grid size	Nb 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



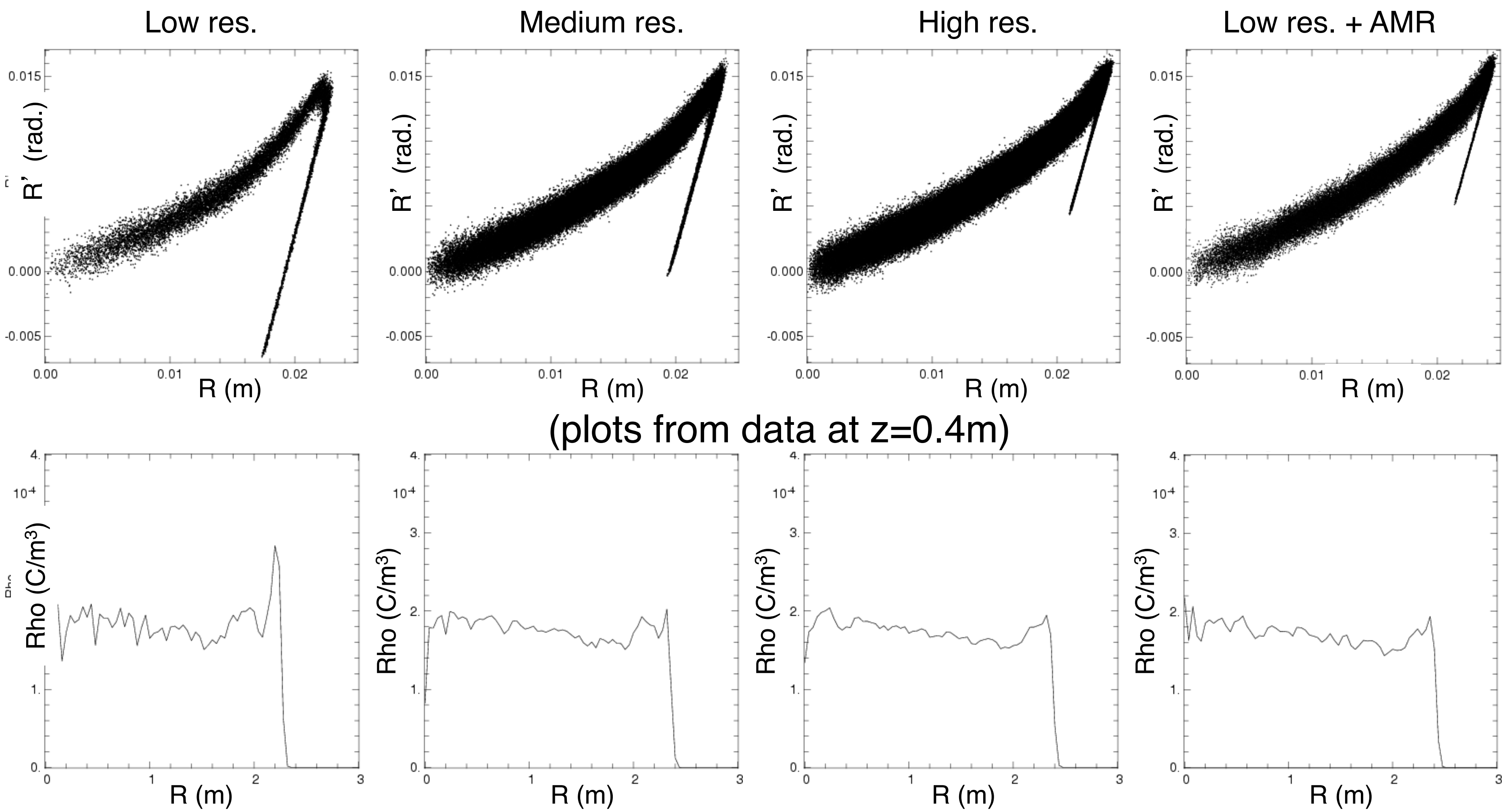
zoom





Automatic mesh refinement follows gradients: emitting area, beam edge and front.

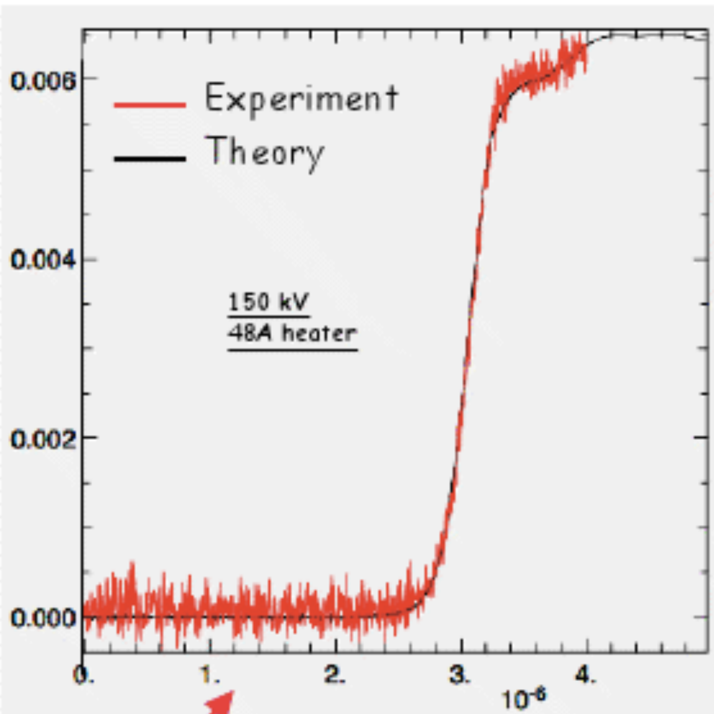
Details of the distribution evolve with resolution.



WARP simulations model STS-500 experiments using 10-cm-diameter K^+ alumino-silicate source

Rise time

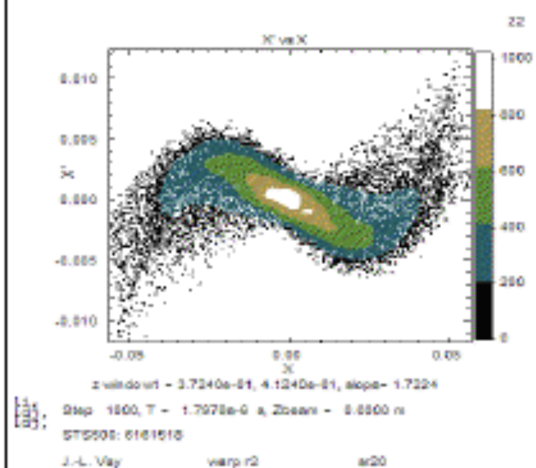
Current at Faraday cup



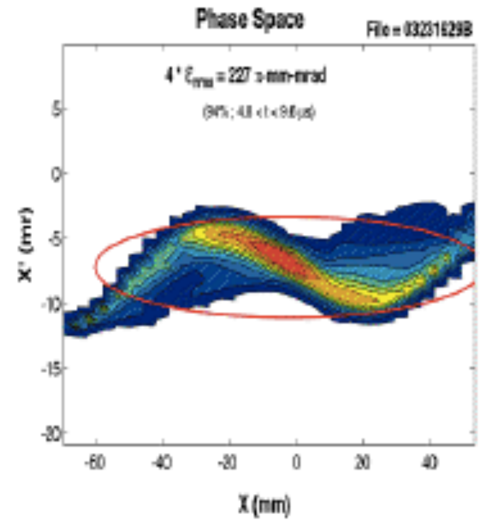
Result depends critically on Mesh Refinement

Phase Space at End of Diode

Warp simulations

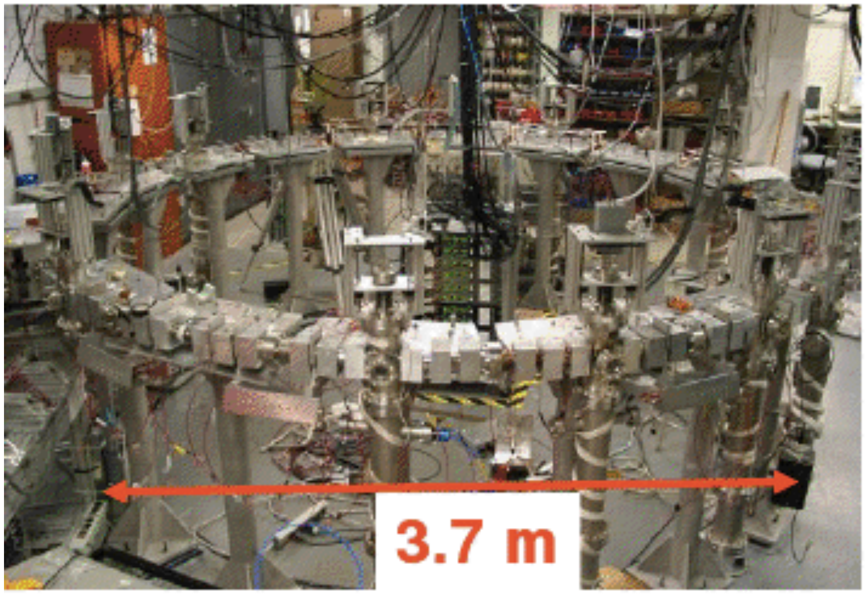


Experimental results



(simulations by I. Haber, J-L. Vay, D. P. Grote)

UMER: Scaled Model for Space-Charge Dynamics



low energy
10 keV

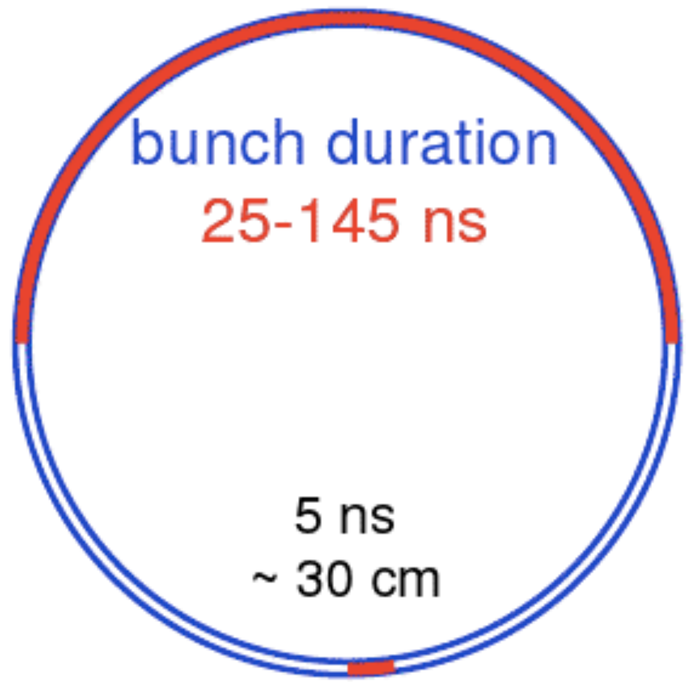
high charge
0.1-10 nC/bunch

low-emittance
0.3-3 μm

Small, flexible, yet compact

Excellent training ground for students

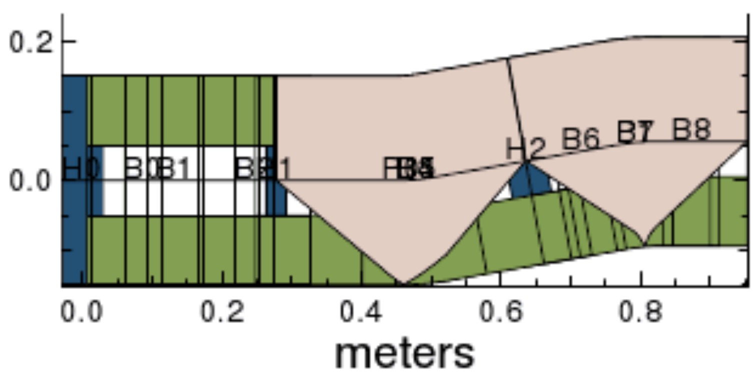
Available and Accessible –
no "Users" other than beam physicists

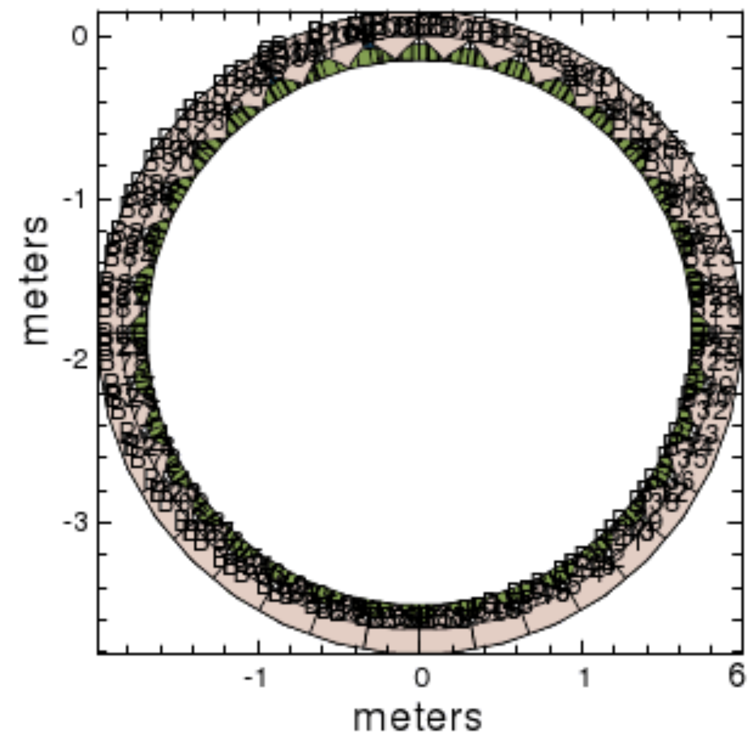


Comprehensive modeling (UMERGeometry WARP module)

WARP repository that includes:

- descriptions and choice of models for every single lattice element
    - quadrupoles, dipoles, steering dipoles, injection pulsed magnets
    - Earth field data and Helmholtz coil information
    - induction cells
- models of certain diagnostics and procedures for data processing parallel to those used in experiment
- Can read/write file formats for UMER settings from experiment

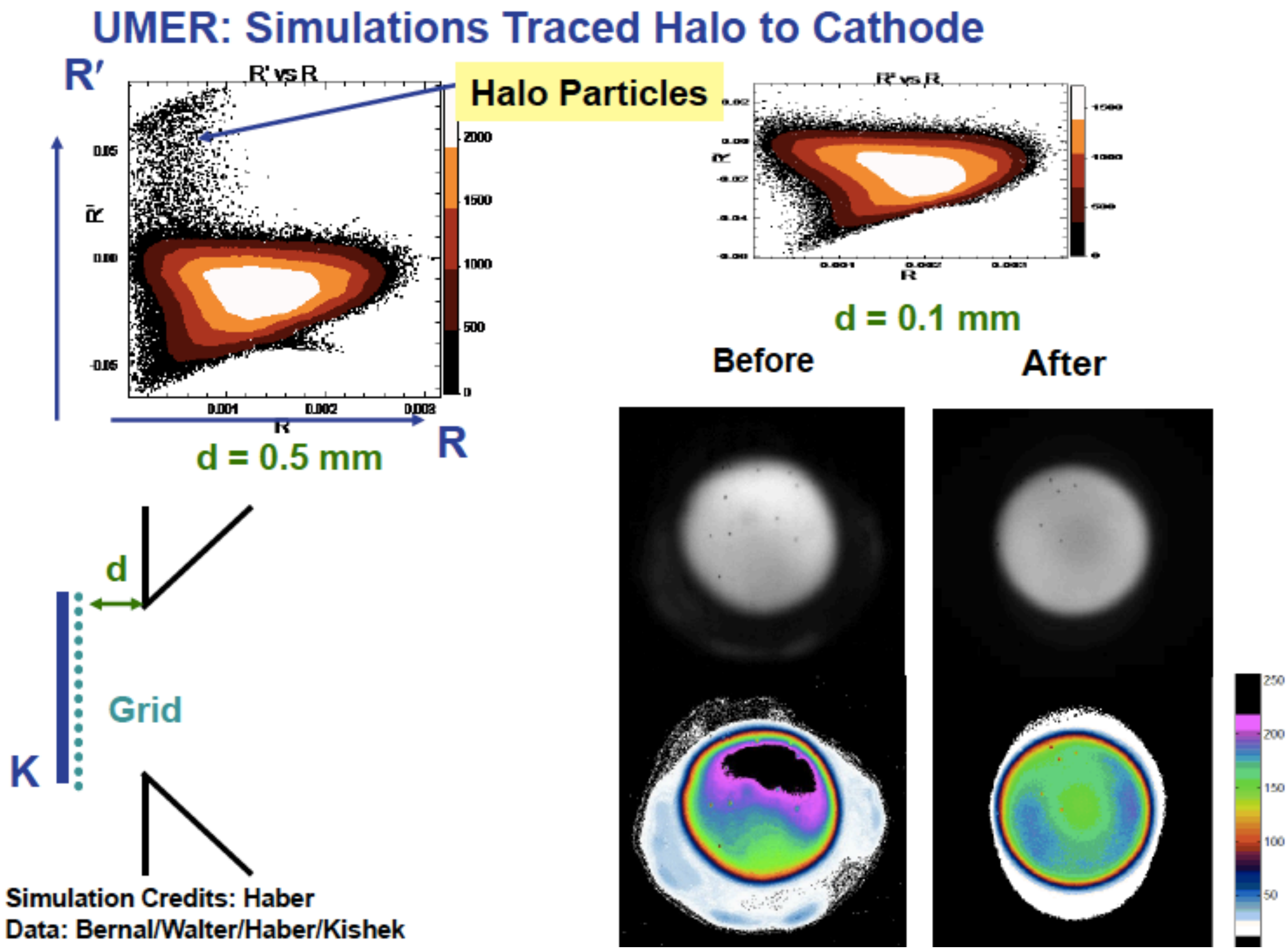




WARP success at UMD

- Transverse
  - Five-Beamlet Evolution (Haber, et al, Phys. Rev. A15, 44, 5194 (Oct. 15, 1991).
  - Identification of Source-generated halo I. Haber, et. al.NIM-A **606**, 64-68 (2009).
- Longitudinal (r-z geometry)
  - Bunch-end interpenetration (Koeth, PAC)
  - Soliton Formation (Charles Thesis)
  - Multi-Stream Instability
- Three-Dimensional
  - Prediction of virtual cathode oscillations in UMER gun I. Haber, et. al. NIM-A **577**, 157-160 (2007).

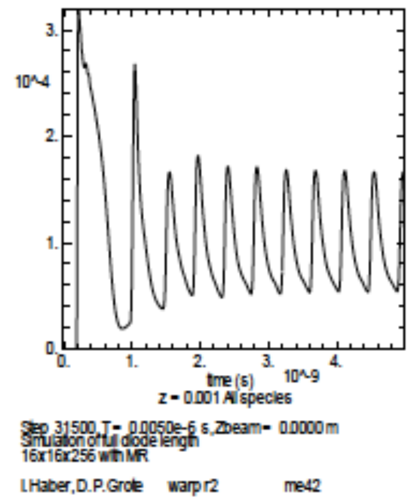
Identification of source halo*



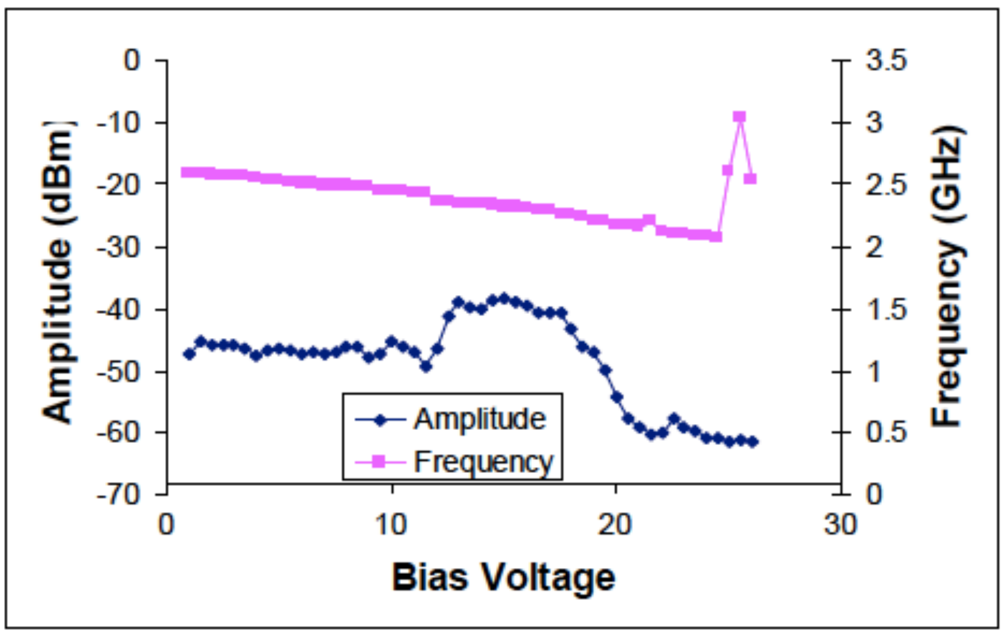
*I. Haber, et al. NIM-A **606**, 64 (2009)

Prediction of virtual cathode oscillations in UMER gun

Virtual cathode oscillations predicted by simulation were measured by spectrum analyzer near predicted frequency.



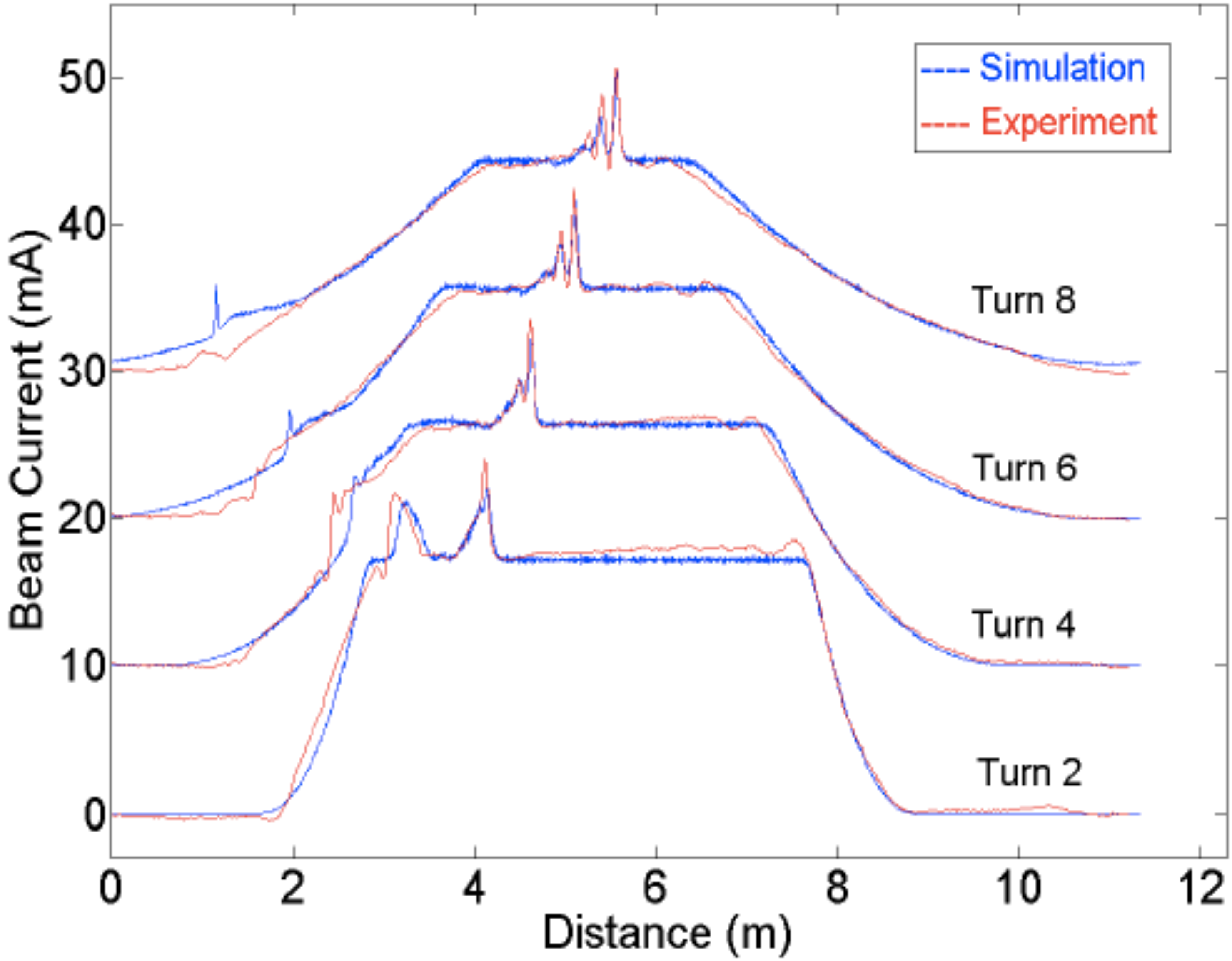
Simulated current waveform



Peak of measured spectrum

*I. Haber, et al. NIM-A **577**, 157-160 (2007)

Soliton wave trains in electron beams

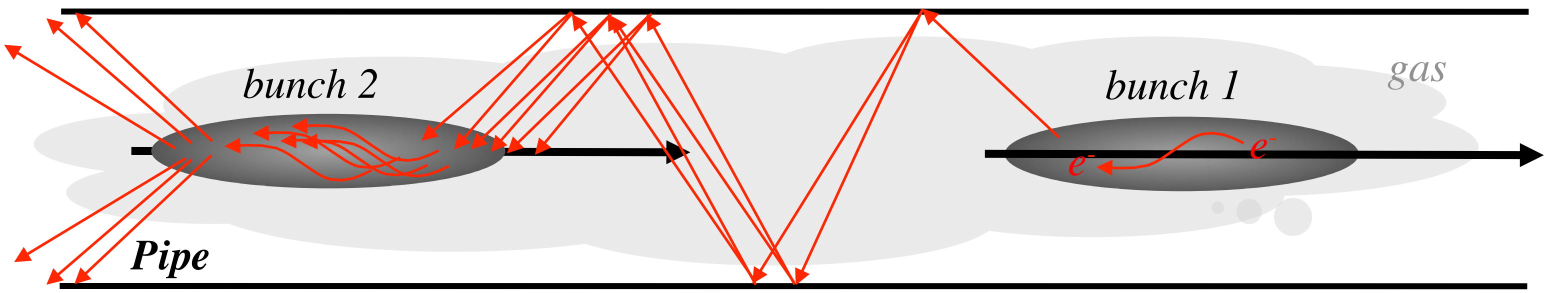


*Y. Mo, et al. PRL 110, 084802 (2013)

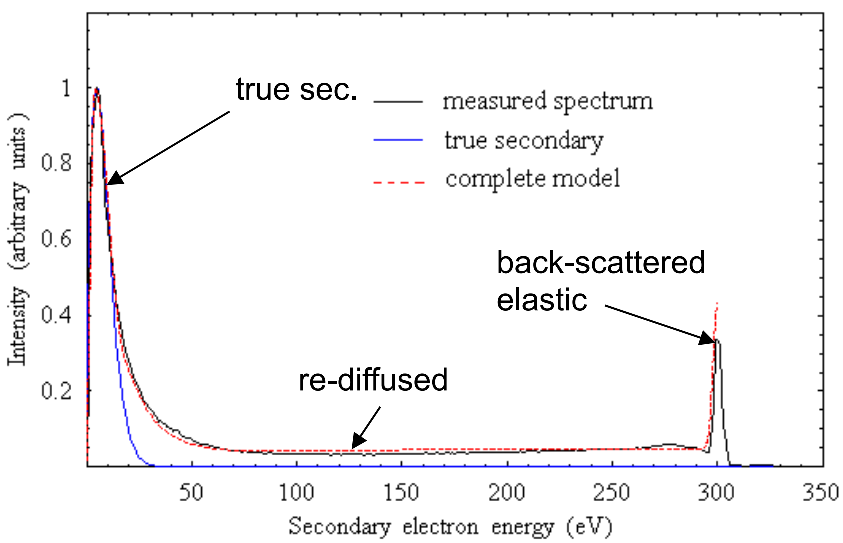
Warp and Posinst integrated in a modular "combo" package

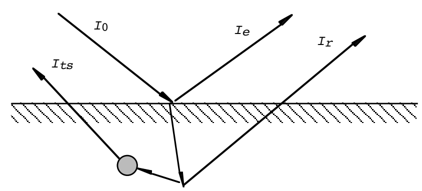
Enabling fully self-consistent modeling of e$^-$ cloud effects: *build-up* & *beam dynamics*:

- Beyond standard practice of simulating e- cloud buildup (ECLOUD, Vorpal, etc) and then its effect on beams (Headtail, SYNERGIA, etc)



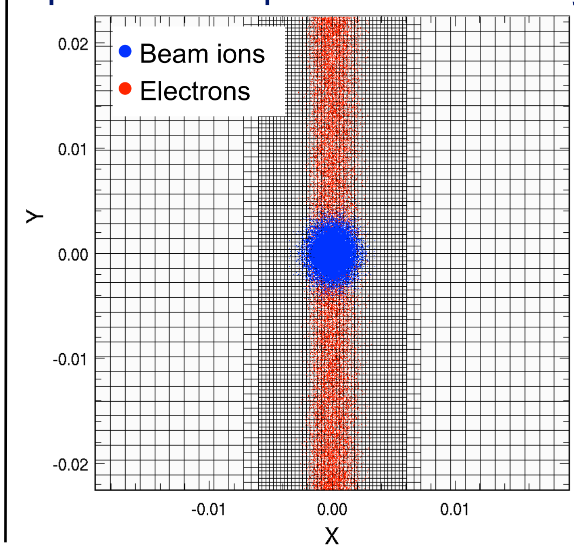
Posinst provides advanced secondary electrons model.





Monte-Carlo generation of electrons with energy and angular dependence.

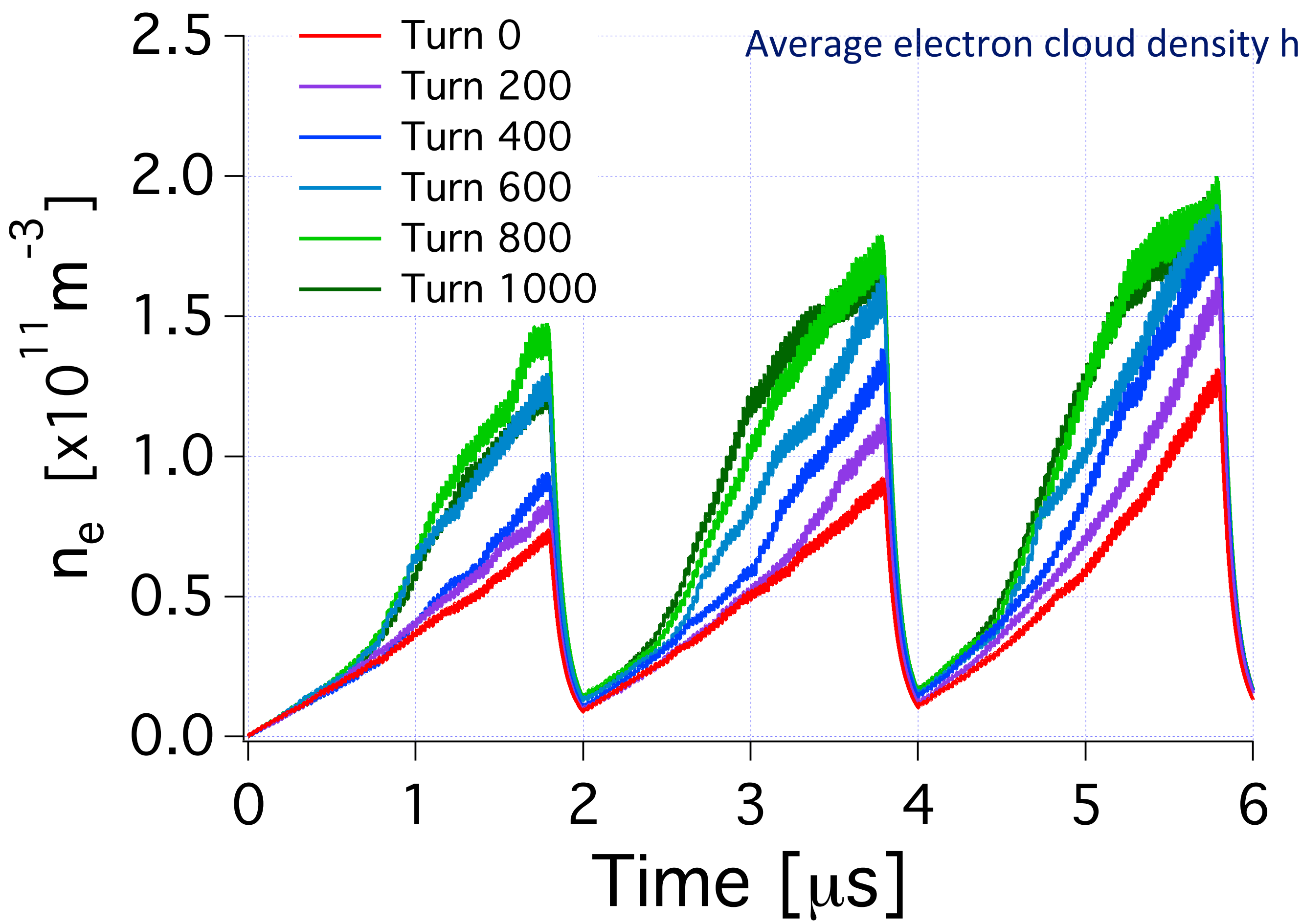
Warp's mesh refinement & parallelism provide efficiency.



Simulation parameters for SPS at injection

- Bunch
  - energy $W=26.$ GeV
  - population $N_p=1.1\times10^{11}$
  - RMS length $\sigma_z=0.23$ m (Gaussian profile)
  - momentum spread $\delta p/p=2\times10^{-3}$
  - transverse normalized emittance $\varepsilon_x= \varepsilon_y=2.8$ mm.mrad
  - longitudinal normalized emittance $\varepsilon_z=0.3$ eV.s
- $\perp$: continuous focusing
  - beta functions $\beta_{x,y}= 33.85, 71.87$
  - betatron tunes $\nu_{x,y}= 26.13, 26.185$
  - chrom. $Q_{x,y}=0.,0.$
- // : continuous focusing
  - momentum compaction factor $\alpha=1.92\times10^{-3}$
  - cavity voltage $V = 2$ MV
  - cavity harmonic number $h = 4620.$
- assumed 100% dipole
- Bunch-to-bunch feedback system in horizontal plane (gain=0.1)
- 10 interaction stations/turns

Warp-Posinst enabled first direct simulation of a train of 3x72 bunches
-- using 9,600 CPUs on Franklin supercomputer (NERSC, U.S.A.)

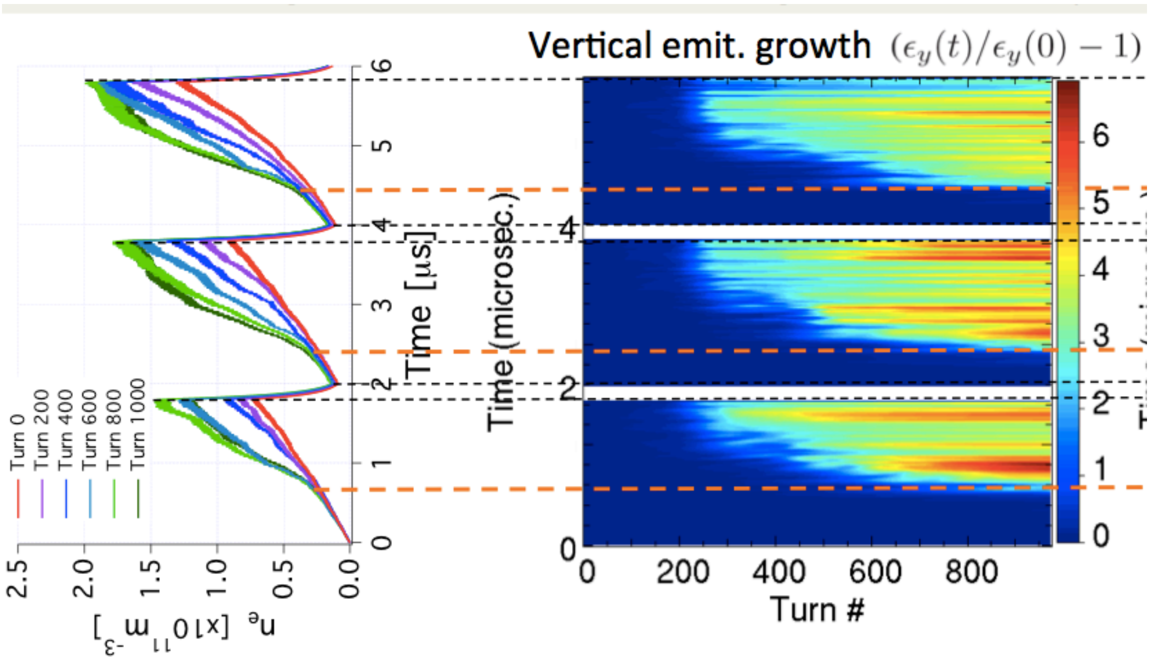


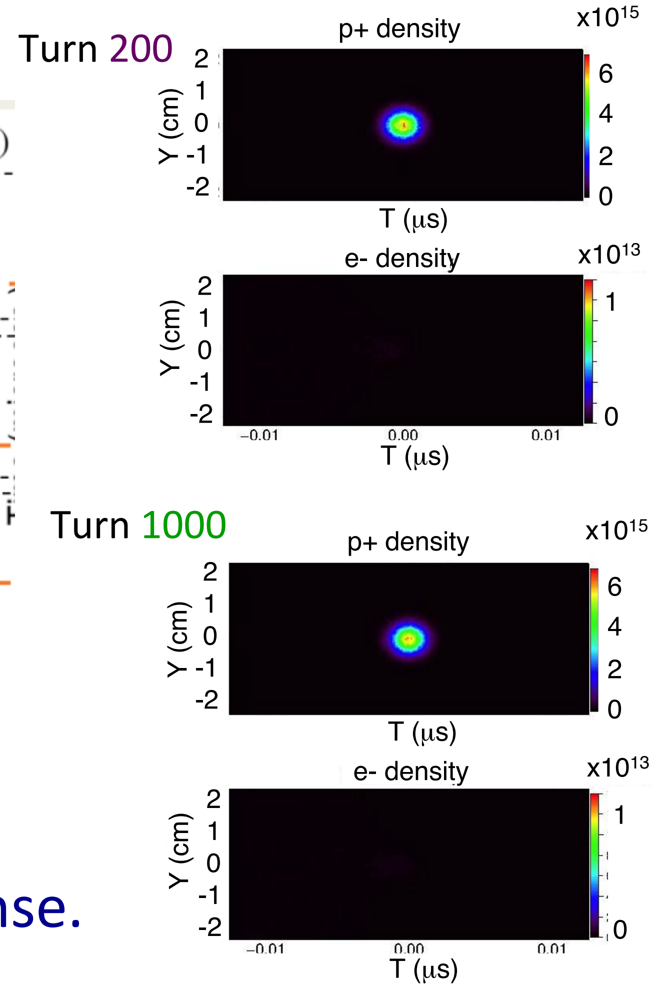
Substantial density rise in tails of batches between turns 0 and 800.

Substantial density rise in tails of batches between turns 0 and 800.

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

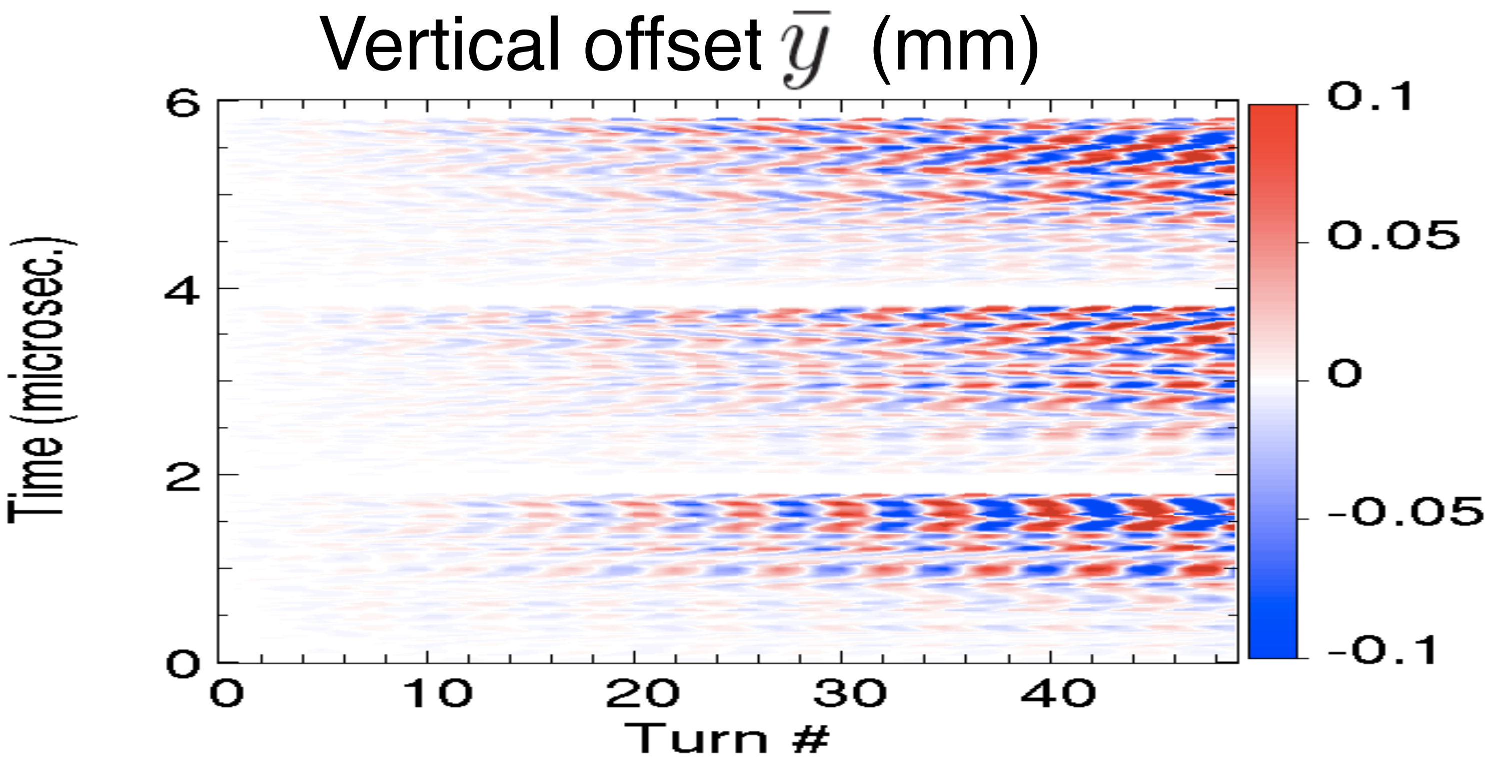
E-cloud density raise coincides with growth of vertical emittance





⇒ Positive coupling between the e-cloud buildup and the bunches dynamical response.

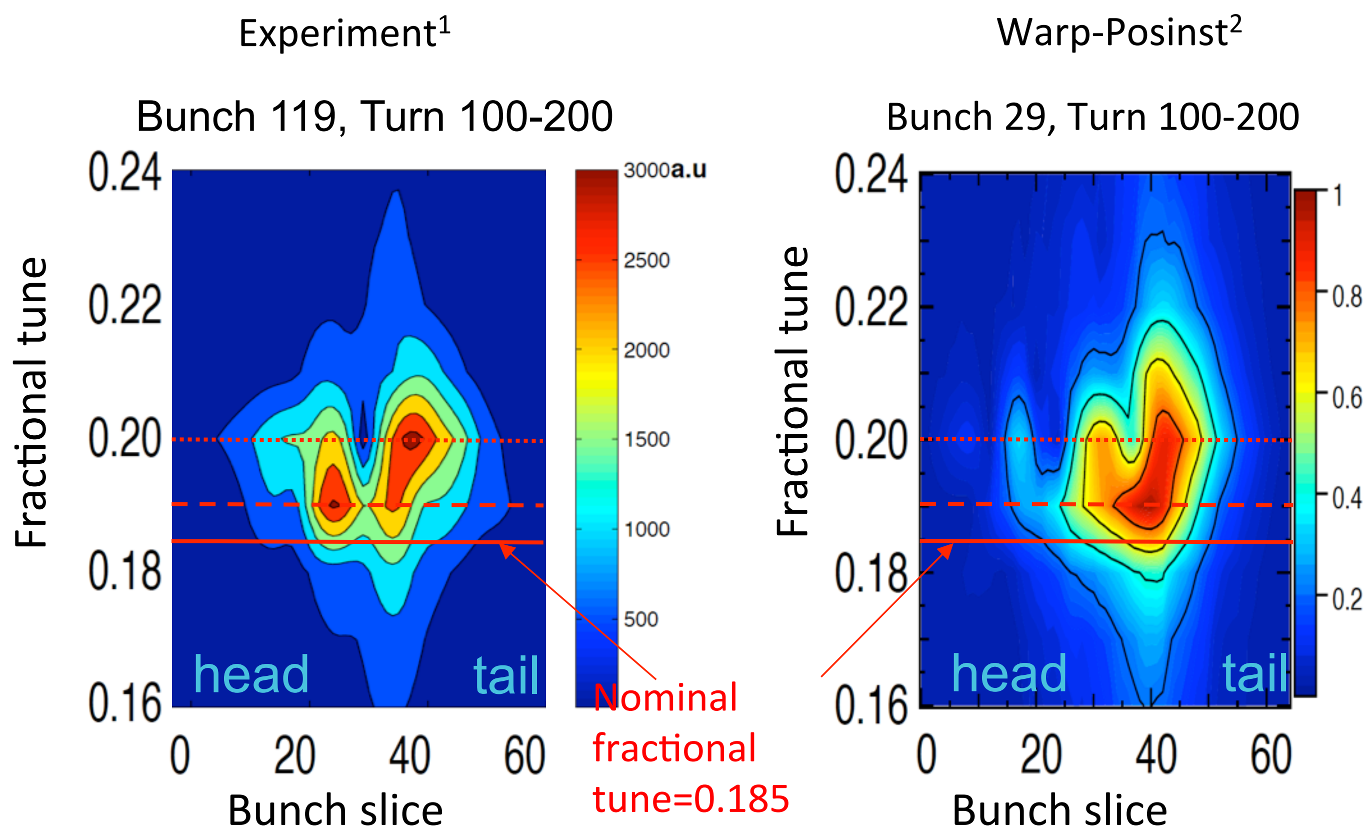
Pattern of stripes in the history of vertical bunch offsets



⇒ phase of the oscillations is not purely random

E-cloud provides coupling between bunches.

Comparison with experimental measurements
-- collaboration with SLAC/CERN

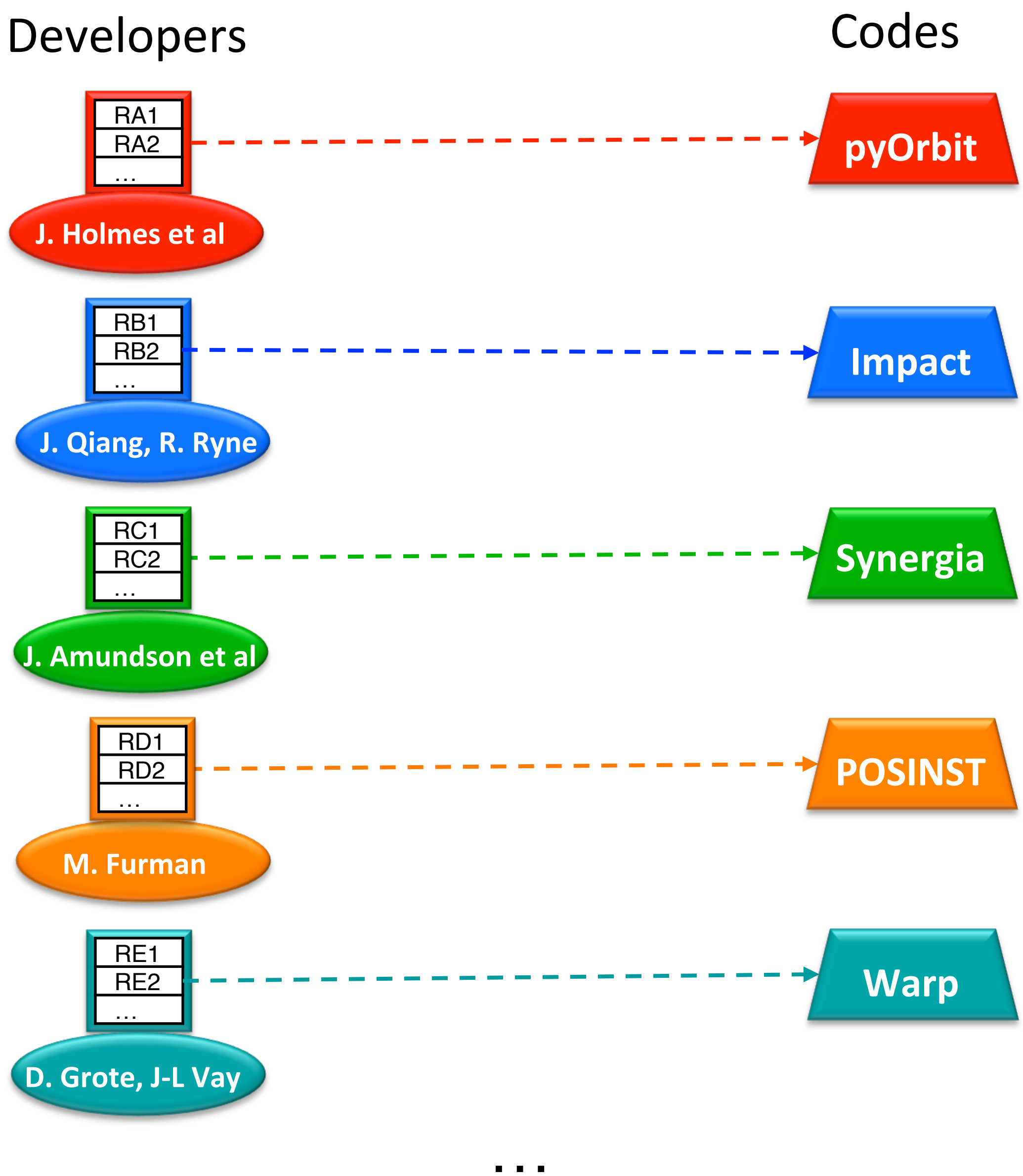


[1]J. Fox, et al, *IPAC10 Proc.*, p. 2806 (2011) [2]J.-L. Vay, et al, *Ecloud10 Proc.*, (2010)

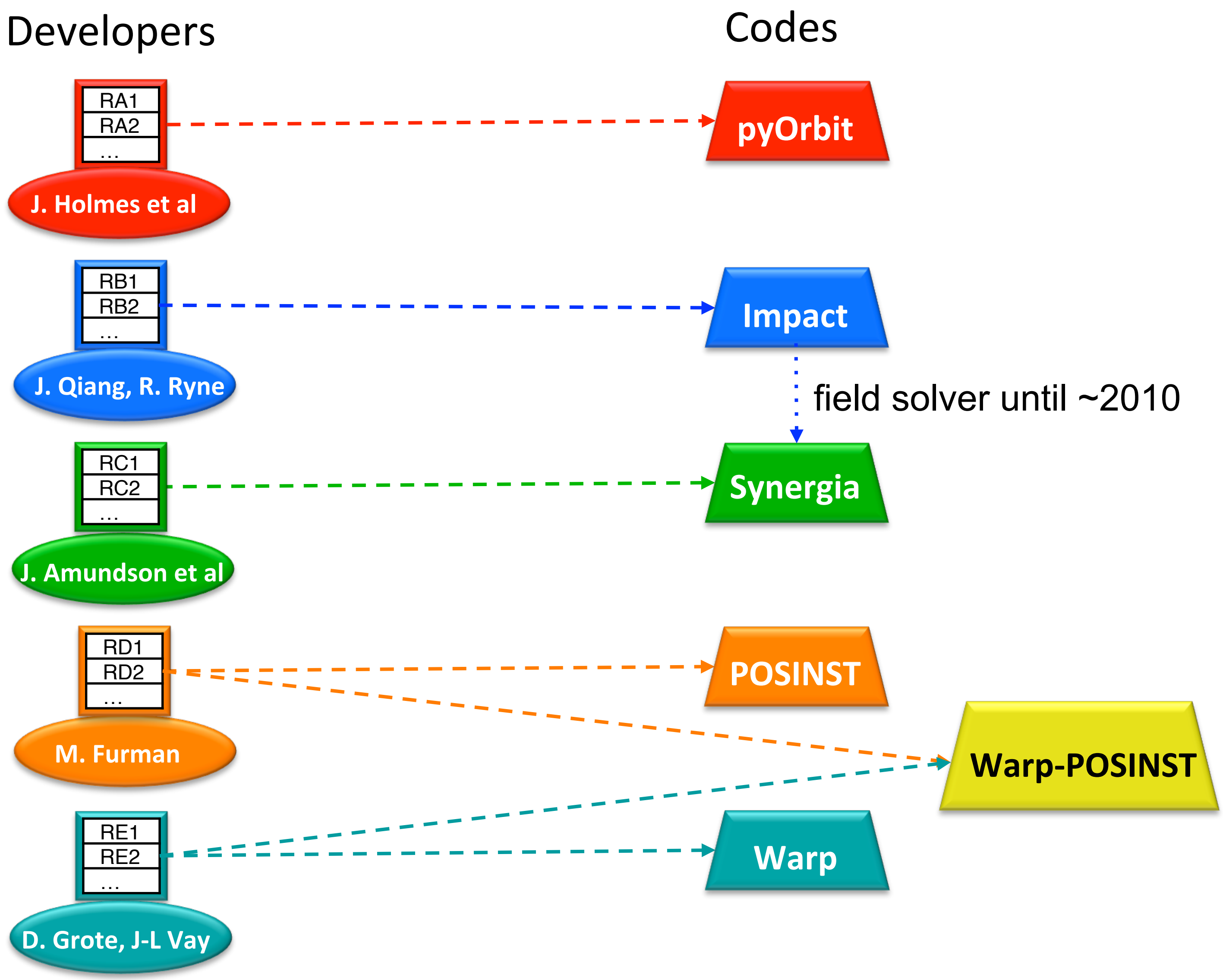
Good qualitative agreement: separation between core and tail with similar tune shift.

Warp also applied to study of feedback control system (R. Secondo in collaboration with SLAC)

So far, codes developed mostly separately



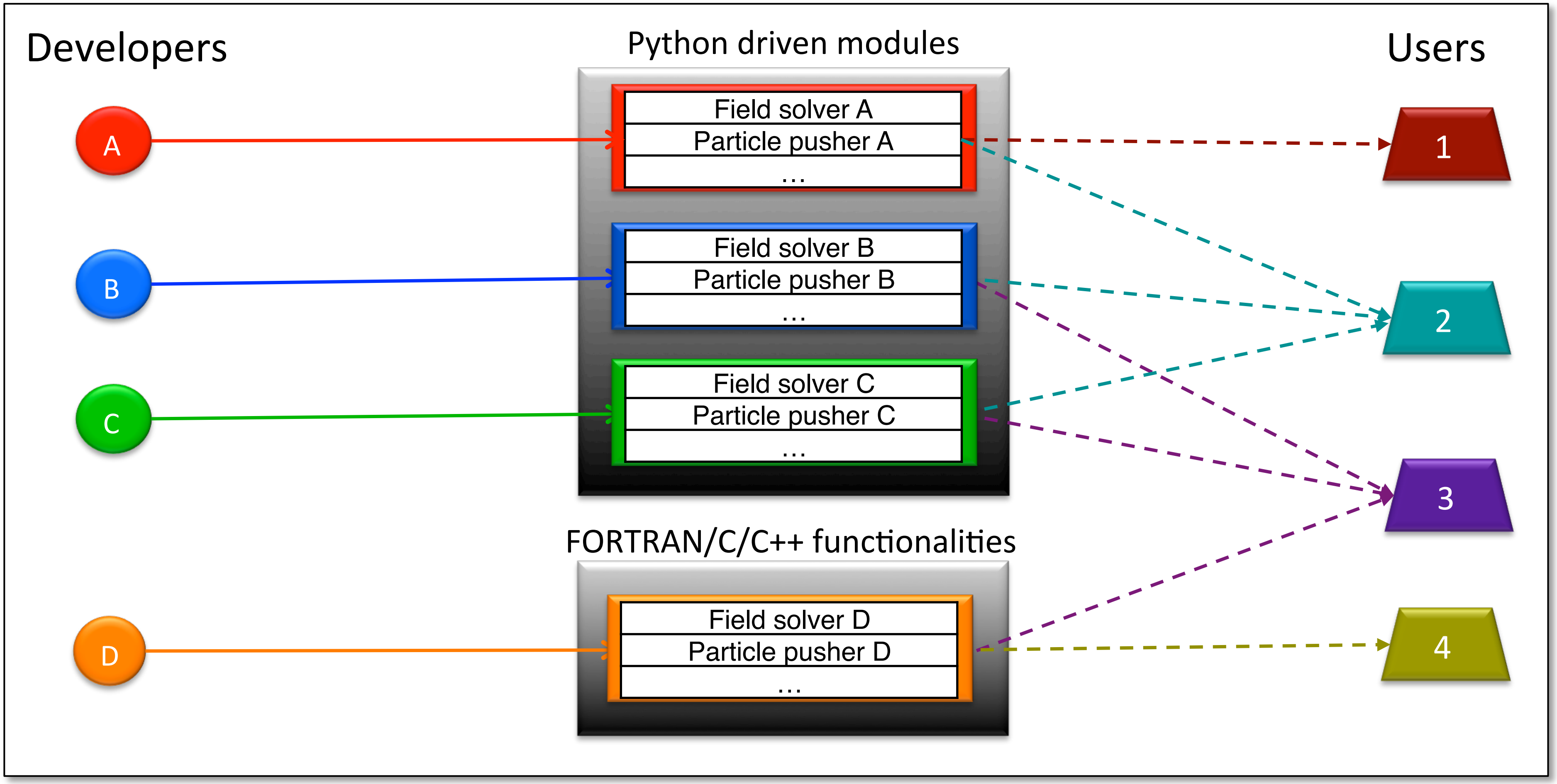
Although…


Developers
Codes
RA1
RA2
…
J. Holmes et al
pyOrbit
RB1
RB2
…
J. Qiang, R. Ryne
Impact
field solver until ~2010
RC1
RC2
…
J. Amundson et al
Synergia
RD1
RD2
…
M. Furman
POSINST
Warp-POSINST
RE1
RE2
…
D. Grote, J-L Vay
Warp

Further integration to increase capabilities, reduce duplication

With modular programming, we can share modules like Lego pieces.

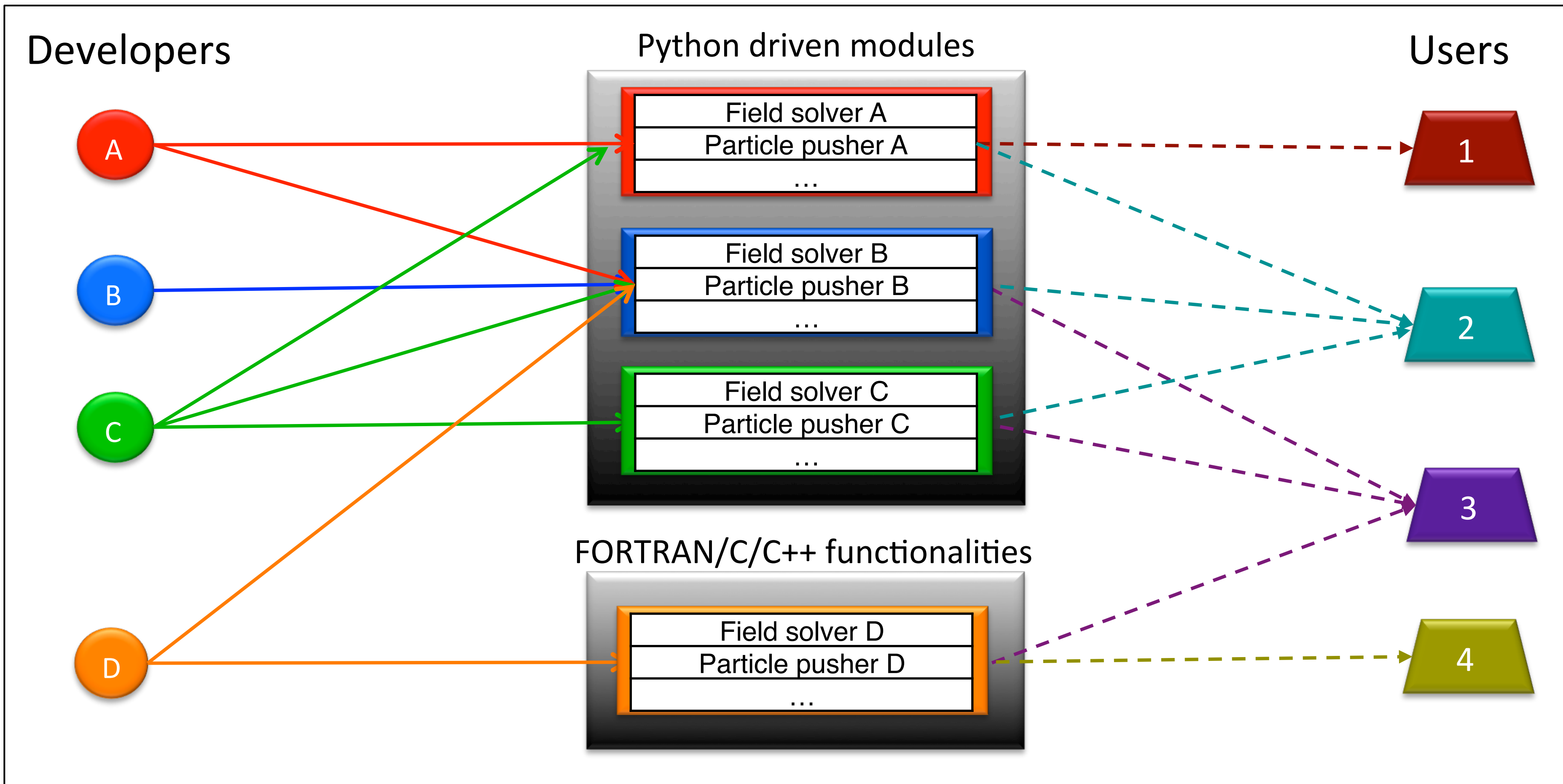
Having Python based codes makes it especially easy!



Developers can also collaborate on modules

Leveraging competencies will lead to better more capable tools:

- offers environment for improvement via collaboration/competition/”natural selection”.



Summary

- **Warp is a parallel Particle-In-Cell simulation code/framework/module for the modeling of accelerators, beams, plasmas and laser-plasma interaction.**

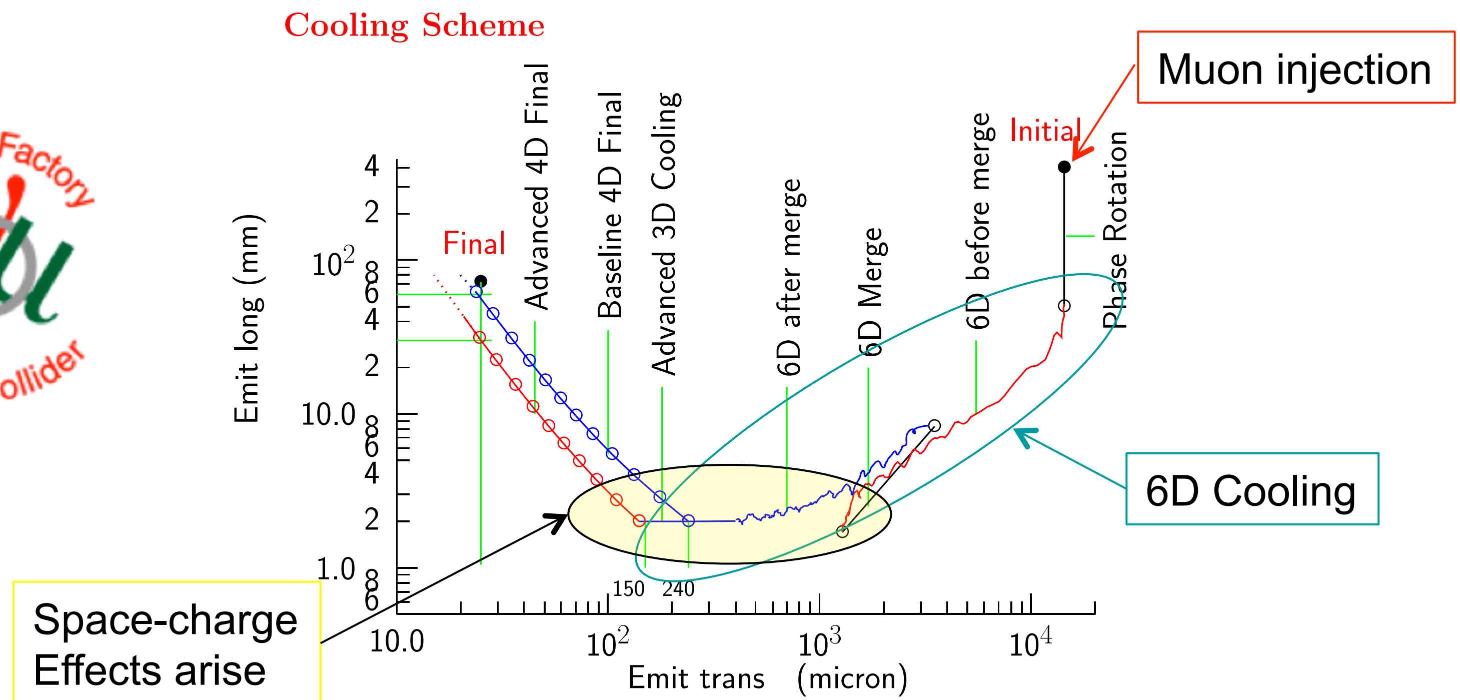
- **Has been successfully applied to the modeling of space charge dominated beams**
  - with extensive benchmarking against experiments

- **Increased collaborations would leverage various efforts:**
  - enabling more capable & better performing tools,
  - greatly facilitated by growing number of codes with Python front-end,
  - developers would develop functionalities rather than codes, then assemble end tools by assembling the "Lego" pieces,
    - note that this does not exclude coexistence of current tools.

- **For questions regarding Warp, email to** DPGrote@lbl.gov, JLVay@lbl.gov **or** AFriedman@lbl.gov

Warp applied to muon cooling in US Muon Accelerator Program

- **After capture, muon bunch requires extensive cooling**
- **During cooling, the beam shortens, and space-charge effects rise**
- **Warp is used to study and minimize space-charge effects**



Warp & ICOOL codes were combined via Python

- **Warp provides most of the particle tracking plus self fields**
  - Advances particles
  - Applies solenoid and RF fields
  - Calculates and applies self fields (ES or EM)
  - Diagnostics (some ICOOL diagnostics replicated in Python)
- **ICOOL provides absorption**
  - Warp calls setup routines for absorption (specifying absorbing material)
  - For particles in the absorber, Warp calls DEDX routine from ICOOL, which modifies the momentum
- **No changes to ICOOL (except skipping main routine)**
  - ICOOL wrapped using Forthon – have direct access to data in commons and the needed routines
- **Particle handling in Python – passing appropriate particles to ICOOL**