Application of the *B*erkeley *L*ab *A*ccelerator *S*imulation *T*oolkit to the Modeling of Plasma Accelerator Experiments

BLAST
BERKELEY LAB ACCELERATOR SIMULATION TOOLKIT

CERN – June 1, 2018

Jean-Luc Vay

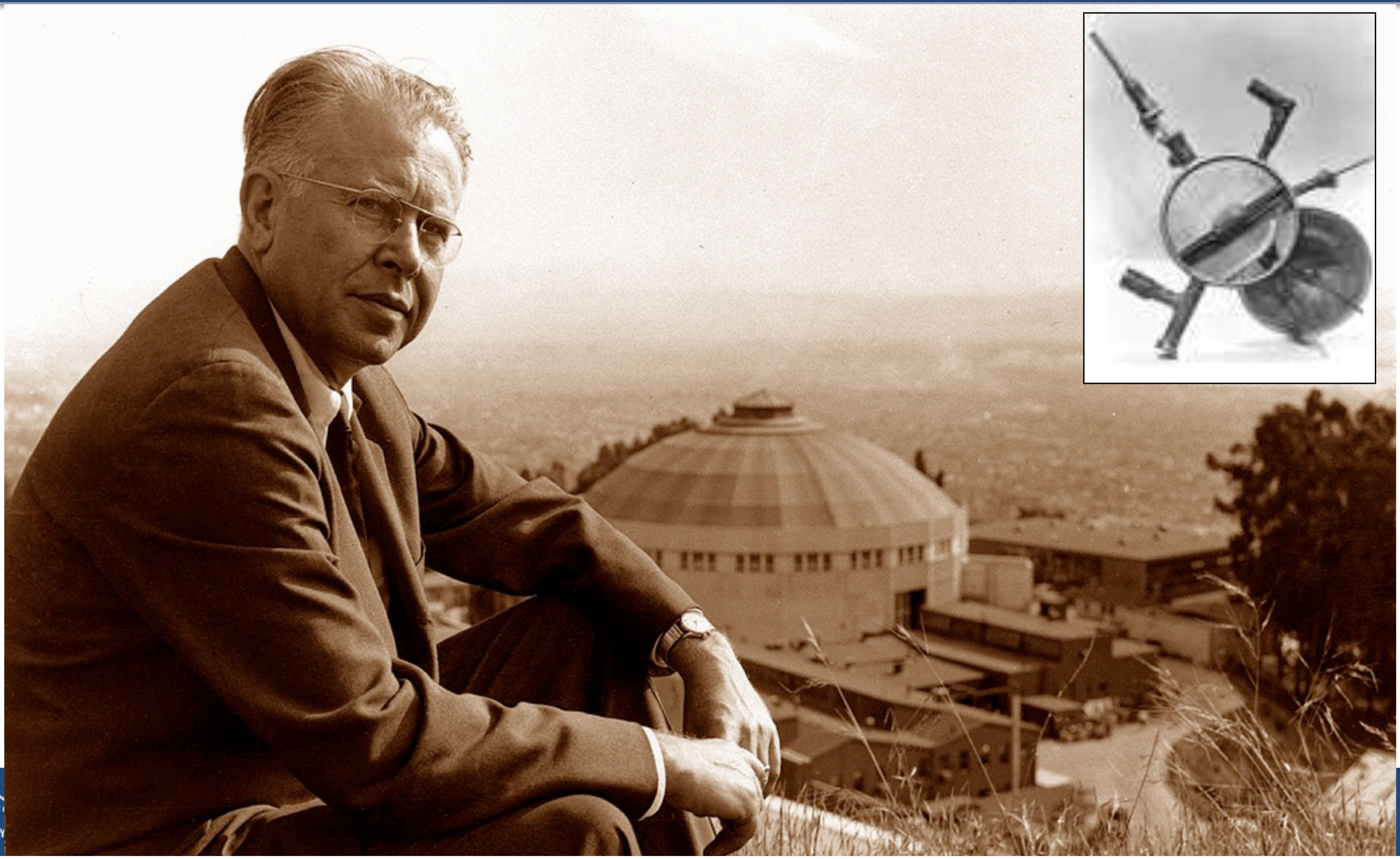
Lawrence Berkeley National Laboratory

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

Outline

- Intro & overview of the BLAST toolkit
- Advanced algorithms for plasma accelerator modeling
- BLAST codes for plasma accelerator modeling: Warp, FBPIC, WarpX.
- Examples
- Summary & outlook

Founded in 1931 on Berkeley Campus
Moved to Current Site in 1940



Lawrence's Successful Legacy of Team Science

Laboratory Director
M.S. Witherell

M.T. BRANDT *Deputy Director for Operations*	**H.D. SIMON** *Deputy Director for Research*

J.A. Blair, *Laboratory Counsel*
M.K. Dick, *Chief of Staff*
J.D. Elliott, *Sustainability Officer*
A. Flores, *Department Head, Internal Audit*
J. German, *Chief Communications Officer*
D. Medley, *Head of Government and Community Relations*
E. Quaite-Randall, *Chief Technology Transfer Officer*
T. Triplett, *Interim Director, Institutional Assurance and Integrity*
K.P. Williams, *Chief Financial Officer*

SCIENCE

ALS-U
D. ROBIN, *ALS-Upgrade Project Director*

BIOSCIENCES **M.E. MAXON** *Associate Laboratory Director*	**BIOLOGICAL SYSTEMS & ENGINEERING** B.A. Simmons *Division Director*	**DOE JOINT GENOME INSTITUTE** N.J. Mouncey *Facility Director*	**ENVIRO. GENOMICS & SYSTEMS BIOLOGY** N.L. Glass *Division Director*	**MOLECULAR BIOPHYSICS & INTEGRATED BIOIMAGING** P.D. Adams *Division Director*
COMPUTING SCIENCES **K.A. YELICK** *Associate Laboratory Director*	**COMPUTATIONAL RESEARCH** D.L. Brown *Division Director*	**NAT. ENERGY RESEARCH SCIENTIFIC COMPUTING CENTER (NERSC)** S.S. Dosanjh *Division Director*	**SCIENTIFIC NETWORKING (ESNET)** I.S. Monga *Division Director*	
EARTH & ENVIRONMENTAL SCIENCES **S.S. HUBBARD** *Associate Laboratory Director*	**CLIMATE & ECOSYSTEM SCIENCES** W.D. Collins *Division Director*	**ENERGY GEOSCIENCES** J.T. Birkholzer *Division Director*		
ENERGY SCIENCES **J.B. NEATON** *Associate Laboratory Director*	**ADVANCED LIGHT SOURCE** S.D. Kevan *Interim Division Director*	**CHEMICAL SCIENCES** K. Wilson *Interim Division Director*	**MATERIALS SCIENCES** M. Asta *Division Director*	**MOLECULAR FOUNDRY** J.B. Neaton *Division Director*
ENERGY TECHNOLOGIES **R. RAMESH** *Associate Laboratory Director*	**BUILDING TECHNOLOGY & URBAN SYSTEMS** M.A. Piette *Division Director*	**ENERGY ANALYSIS & ENVIRONMENTAL IMPACTS** T. Kirchstetter *Division Director*	**ENERGY STORAGE & DISTRIBUTED RESOURCES** R. Prasher *Division Director*	**CYCLOTRON ROAD** I. Gur *Division Director*
PHYSICAL SCIENCES **T.J. SYMONS** *Associate Laboratory Director*	**ACCELERATOR TECH & APPLIED PHYSICS.** W. Leemans *Division Director*	**ENGINEERING** H. von der Lippe *Division Director*	**NUCLEAR SCIENCE** B.V. Jacak *Division Director*	**PHYSICS** N.A. Roe *Division Director*

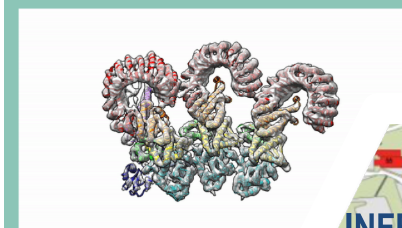
JGI JOINT GENOME INSTITUTE
A DOE OFFICE OF SCIENCE USER FACILITY

jbei Joint BioEnergy Institute

ABPDU

Agile BioFoundry

SCIENCE AT THE BIOCAMPUS



BREAKTHROUGH SCIENCE AT THE EXASCALE

NERSC

ESnet
ENERGY SCIENCES NETWORK

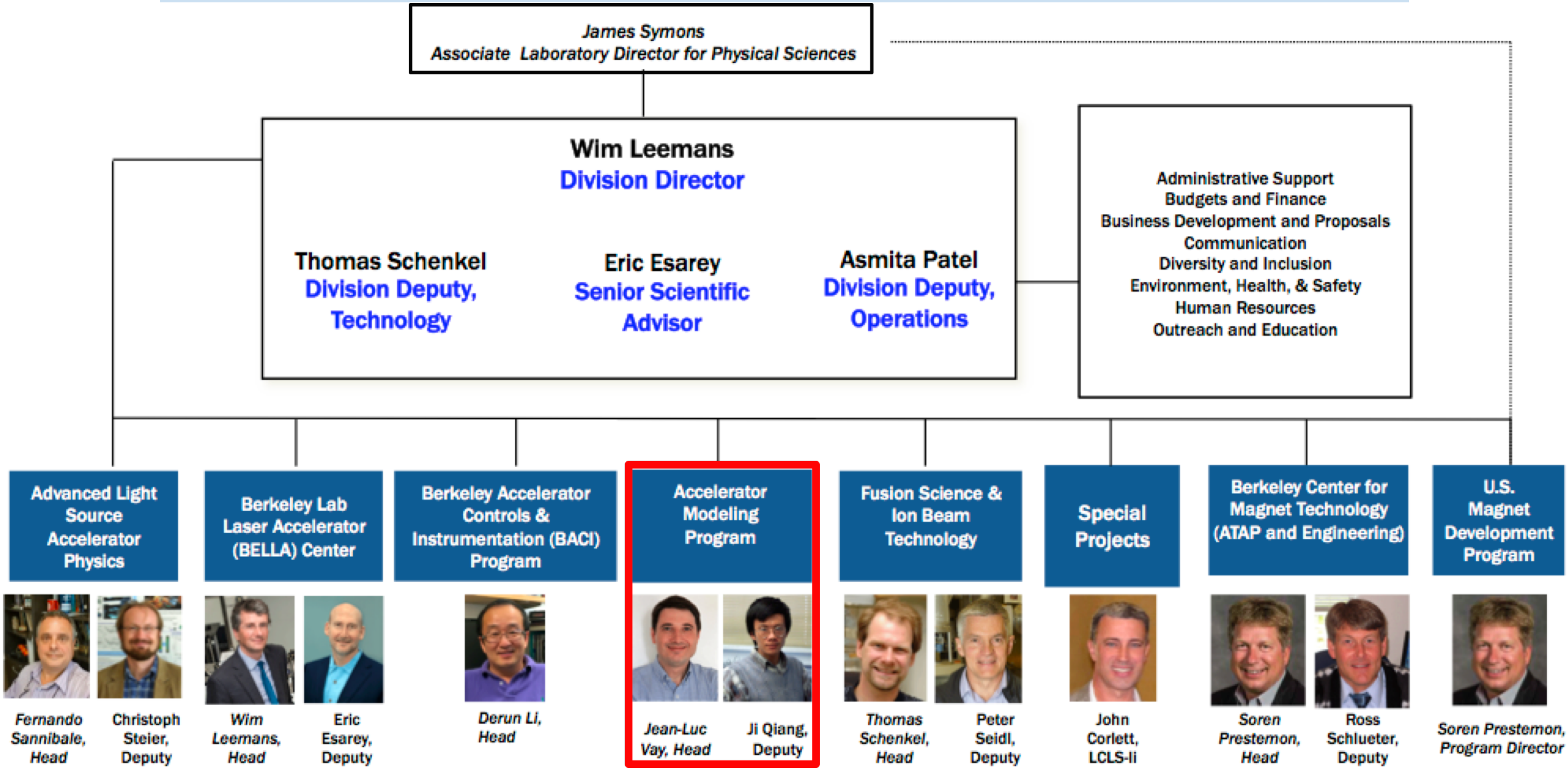
ALS
ADVANCED LIGHT SOURCE

MOLECULAR FOUNDRY

SCIENCE WITH THE UPGRADED ADVANCED LIGHT SOURCE

DISCOVERY SCIENCE IN FUNDAMENTAL PHYSICS

Accelerator Technology and Applied Physics Division



LBNL ATAP Accelerator Modeling Program

Scientific Staff

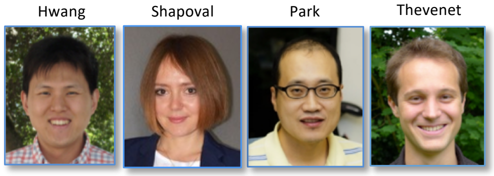
Jean-Luc Vay (lead) | Ji Qiang (deputy) | Chad Mitchell | Greg Penn | Rob Ryne | Rémi Lehe

Admin. Support

Lucky Cortez

Postdocs

Kilean Hwang | Olga Shapoval | Jaehong Park | Maxence Thevenet



Students

Biaobin Li | Zhicong Liu | Chuwen Luo | Yue Tao | Pierre Pelletier

Affiliates

Guillaume Blaclard | William Fawley | Alex Friedman | Miguel Furman | Brendan Godfrey

LBNL/SLAC | LLNL | LBNL | U. Maryland

David Grote | Axel Huebl | Manuel Kirchen | Steven Lund | Henri Vincenti

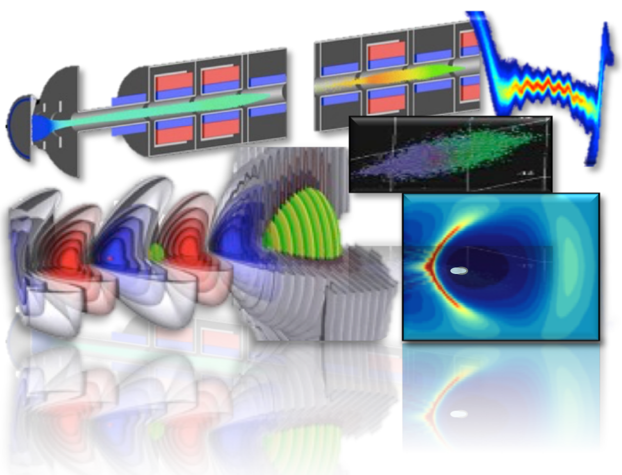
LLNL | U. Dresden | U. Hamburg | Michigan State U. | CEA Saclay

AMP is home of "BLAST" simulation toolset
for conventional & advanced concepts accelerators

blast.lbl.gov

BLAST

BERKELEY LAB ACCELERATOR SIMULATION TOOLKIT



State-of-the-art simulation tools*:

- ***Multi-physics frameworks***: IMPACT, Warp, WarpX.
- ***Specialized codes***: BeamBeam3D, FBPIC, LW3D, POSINST.
- ***Library***: PICSAR.

Wide set of physics & components:

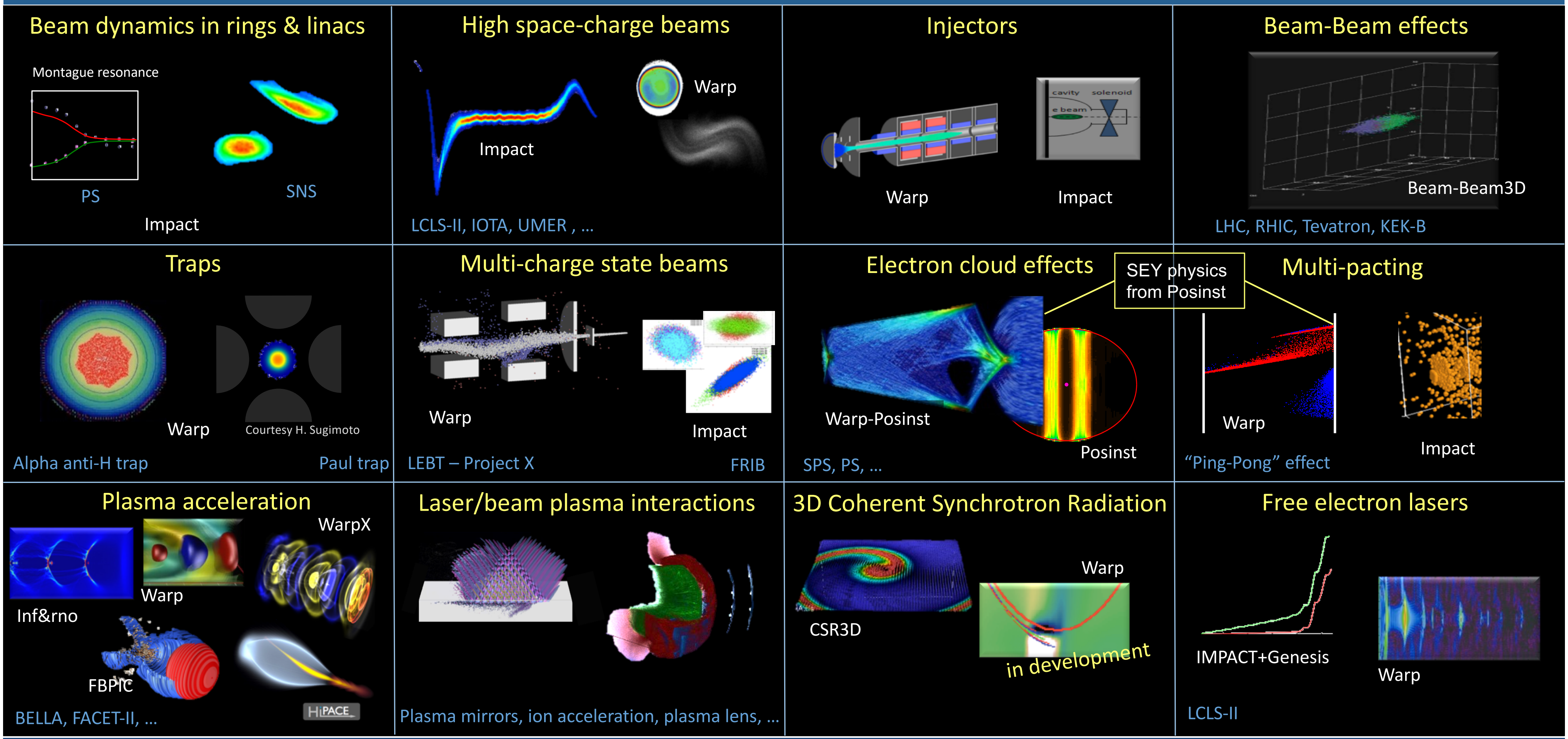
- beams, plasmas, lasers, structures, beam-beam, e^- clouds, …
- linacs, rings, injectors, traps, …

At the forefront of computing:

- ***novel algorithms:*** boosted frame, particle pushers, etc.
- SciDAC, INCITE, NESAP, DOE Exascale.

*Most codes open source, available at blast.lbl.gov or upon request.

Sample applications of *BLAST* codes



Outline

- Intro & overview of the BLAST toolkit
- **Advanced algorithms**
- BLAST codes for modeling of plasma accelerators: Warp, FBPIC, WarpX.
- Examples
- Summary & outlook

Pushing the state-of-the-art in beam/accelerator modeling ➔ nurturing the development of *new algorithms*

Algorithm/method	Reference	Originated	Adopted by
Integrated Maps for rf cavity dynamics	*Ryne, LANL Report* ***1995***	ML/IMPACT	D. Abell nonlinear model
Stochastic Leap-Frog for Brownian motion	*Qiang & Habib, PRE* ***2000***	IMPACT	
Spectral-finite difference multigrid solver	*Qiang & Ryne, CPC* ***2001****/****2006***	IMPACT	
Improved Perfectly Matched Layers	*Vay, JCP 2000/JCP* ***2002***	Warp	Osiris
AMR-PIC electrostatic	*Vay et al, LPB* ***2002****/PoP* ***2004***	Warp	pyPIC, WarpX
PIC w/ shift-Green function method	*Qiang et al, PRSTAB* ***2002****/CPC* ***2004***	BBeam3D	
Secondary emission of electrons algorithm	*Furman & Pivi, PRST-AB* ***2003***	Posinst	TxPhysics, Warp, spacecraft charging codes
AMR-PIC electromagnetic	*Vay et al, CPC* ***2004***	Emi2D	Warp, WarpX
3D Poisson solver with large aspect ratio	*Qiang & Gluckstern, CPC* ***2004***	IMPACT	
PIC w/ integrated Green function	*Qiang et al, PRSTAB* ***2006***	ML/IMPACT	BB3D, Pyheadtail, Opal
Hybrid Lorentz particle pusher	*Cohen et al, NIMA* ***2007***	Warp	PICSAR (UCB)
Lorentz boosted frame	*Vay, PRL* ***2007*** 	Warp	INF&RNO, JPIC, Osiris, V-sim, FBPIC, WarpX
Explicit Lorentz invariant particle pusher	*Vay, PoP* ***2008***	Warp	Tristan, QED, PIConGPU, Osiris, Photon-Plasma, etc.
New convolution integral w/ smooth kernel	*Qiang, CPC* ***2010***		

is ongoing

Algorithm/method (cont.)	Reference	Originated	Adopted by
Mixed Particle-Field decomposition method	*Qiang & Li, CPC* ***2010***	BBeam3D	
PIC with tunable electromagnetic solver	*Vay et al, JCP* ***2011***	Warp	V-sim, Osiris
Efficient digital filter for PIC	*Vay et al, JCP* ***2011***	Warp	V-sim, Osiris
Laser launcher from moving antenna	*Vay et al, PoP* ***2011***	Warp	V-sim, Osiris, FBPIC, WarpX
High-precision laser envelope model	*Benedetti et al,* ***2011***	Inf&rno	
Domain decomposition for EM spectral solver	*Vay et al, JCP* ***2013***	Warp	PICSAR, WarpX
Mitigation of num. Cherenkov instability	*Godfrey&Vay, JPC/CPC* ***2014-15***	Warp	Osiris, WarpX
Adaptive unified differential evolution algo.	*Qiang & Mitchell, OO dig.* ***2015***		
Spectral solver with azimuthal decomposition	*Lehe et al, CPC* ***2016***	FBPIC	
Galilean PSATD w/wo azimuthal decomp.	*Lehe, Kirchen, PRE/PoP* ***2016***	FBPIC/Warp	WarpX
Arbitrary order stencil variations analysis	*Vincenti et al, CPC* ***2016***		
Novel algorithm for vectorization	*Vincenti et al, CPC* ***2017***	PICSAR	Warp, WarpX, SMILEI
Novel Poisson solvers in pipe with open BC	*Qiang, CPC* ***2017***	IMPACT	
Fully symplectic tracking model	*Qiang, PRAB* ***2017***	IMPACT	

Investment into new algorithms

is driven

by physics problems.

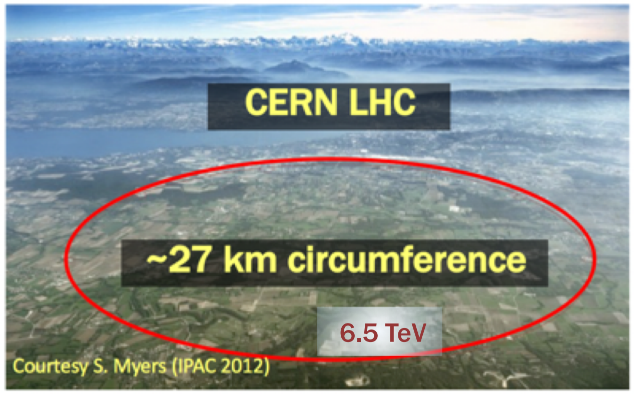
E.g. plasma-based colliders.

Plasma-based acceleration has the potential to make accelerators small (again), and cut cost dramatically

E. O. Lawrence

First cyclotron ~0.1 MeV

1930



W. Leemans

Plasma accelerator 1 GeV

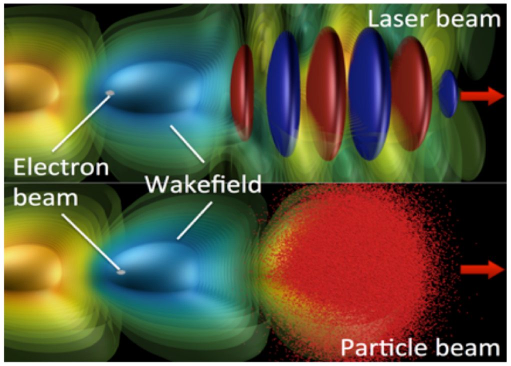
2009

Tens of plasma accelerator stages needed for a 1 TeV e^-e^+ collider.

BUT: simulations in 2-D can take days for 1 stage (even at insufficient resolution for collider beam quality).

➔ Full 3-D modeling of tens of stages needs advanced algorithms + Petascale to Exascale computing.

Plasma accelerators are challenging to model



Short driver/wake propagates through long plasma

➔ Many time steps.

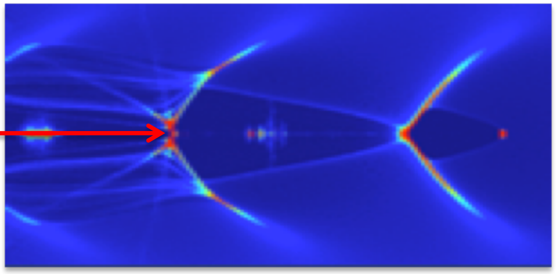
For a 10 GeV LPA scale stage:

~**1μm** wavelength laser propagates into ~**1m** plasma

➔ millions of time steps needed

Non-linear regime:

very small features

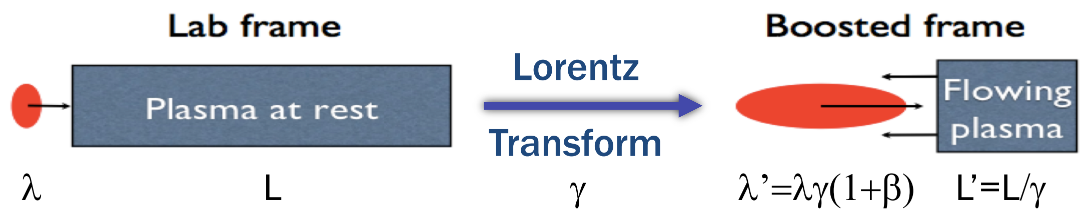


➔ small grid cells

Advanced algorithms are essential to reach goal

Lower # time steps:

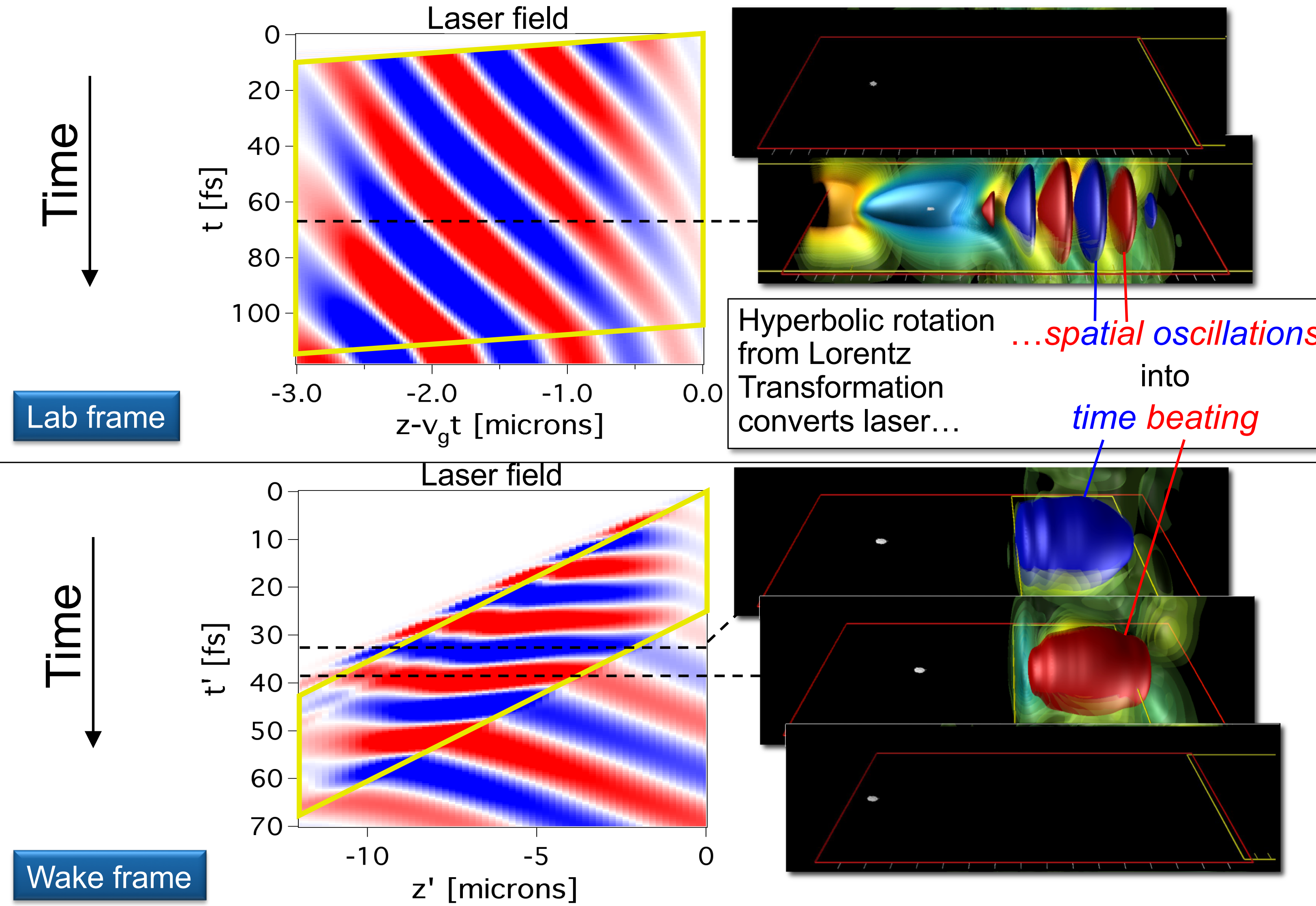
- optimal Lorentz boosted frame (alternate to quasistatic)



time steps $\propto L/\lambda$

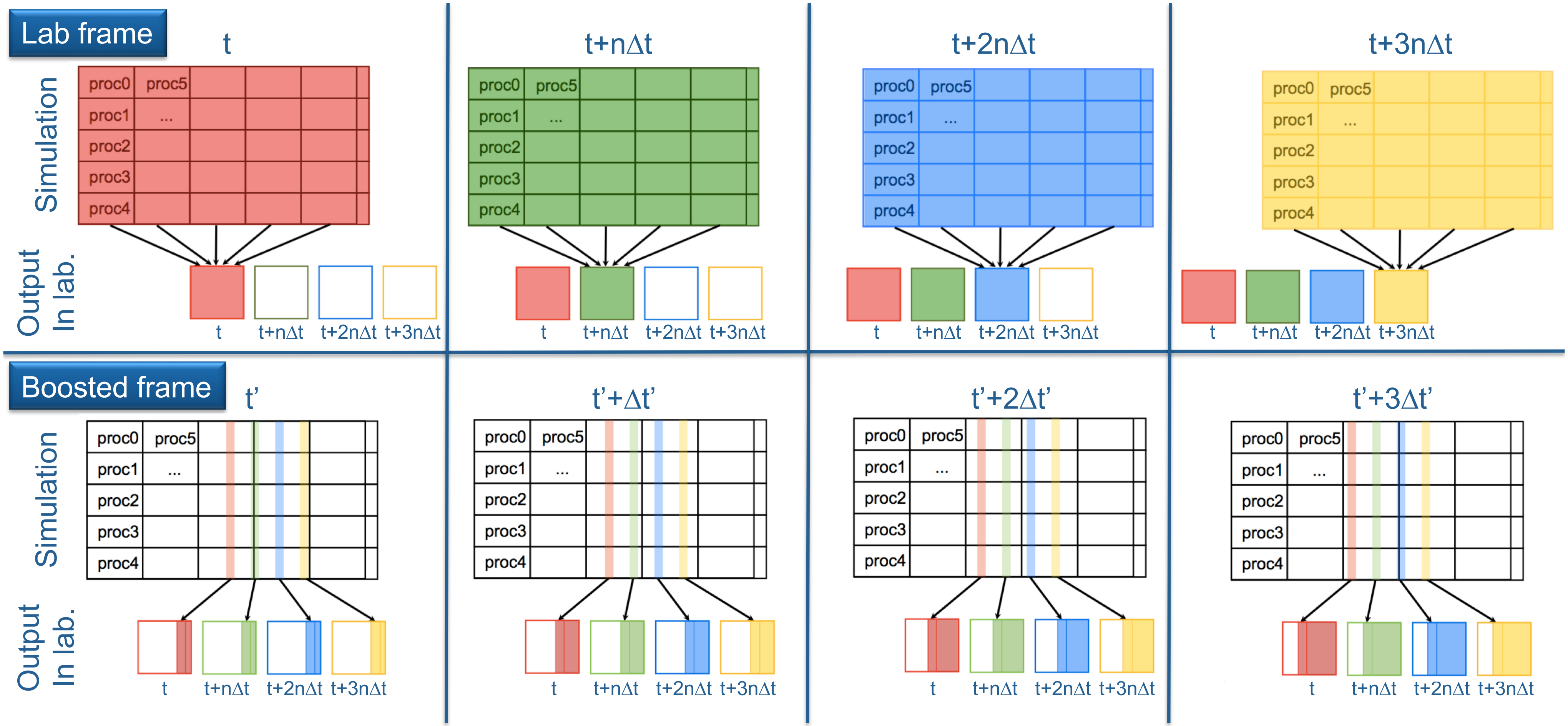
time steps $\propto L'/\lambda'$

➔ Speedup = $(L'/\lambda')/(L/\lambda) = \gamma^2(1+\beta)$



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

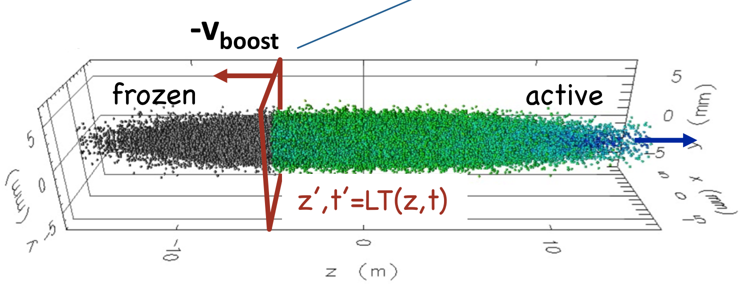
Specialized output needed to reconstruct data in lab frame

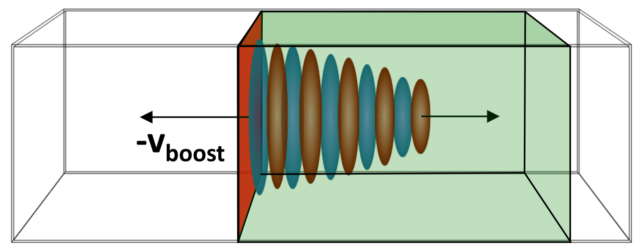


Care needed to ensure frame-independent initial conditions

Initial conditions known in lab frame:

1. **Lorentz transform in boosted frame**
2. **perform injection of particle & laser beams through a moving plane**





J.-L. Vay, W.M. Fawley, C.G.R. Geddes, E. Cormier-Michel, D.P. Grote, *Proc. PAC 2009*, paper TU1PBI04
J.-L. Vay, C. Geddes, E. Cormier-Michel, D. Grote, *Phys. Plasmas* **18**, 123103 (2011)

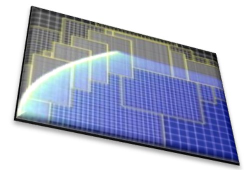
Advanced algorithms are essential to reach goal

Lower # time steps:

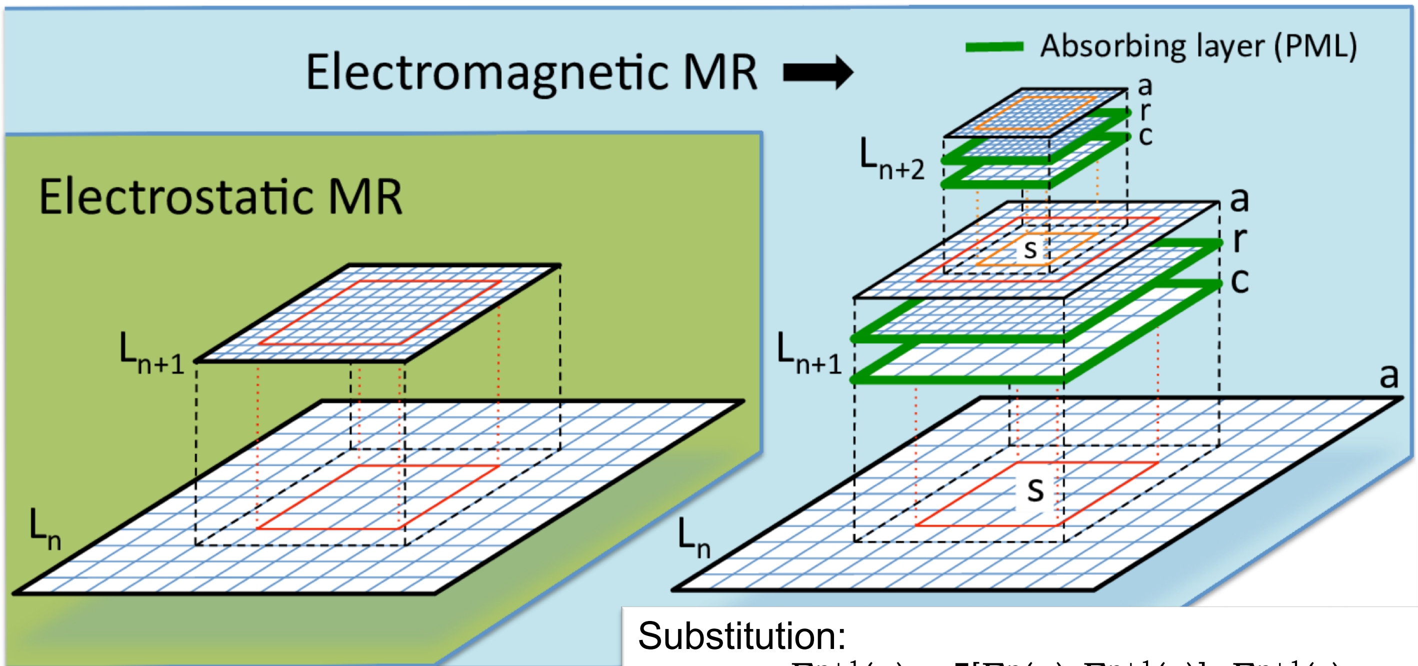
- optimal Lorentz boosted frame

Higher accuracy:

- Adaptive Mesh Refinement



Mesh refinement requires special care



Substitution:

$$F^{n+1}(a) = \mathbf{I}[F^n(a) - F^{n+1}(c)] + F^{n+1}(r)$$

- Need to avoid spurious:
  1. self-forces
  2. wave reflections
  3. `ghost stuck' particles

➔

1. buffer regions
2. PMLs around patches
3. Extended Maxwell with divergence cleaning

$$\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E}$$

$$\frac{\partial \mathbf{E}}{\partial t} = \nabla \times \mathbf{B} - \mathbf{J} + \nabla F$$

$$\frac{\partial F}{\partial t} = \nabla \cdot \mathbf{E} - \rho$$

1. J.-L. Vay, P Colella, P Mccorquodale, B Van Straalen, A Friedman, and D. P. Grote. *Laser & Particle Beams* **20**, 569–575, (2002).
2. J.-L. Vay, J.-C. Adam, A. Héron, *Computer Physics Comm.* **164**, 171-177 (2004).
3. J.-L. Vay, D. P. Grote, R. H. Cohen, & A. Friedman, *Computational Science & Discovery* **5**, 014019 (2012).

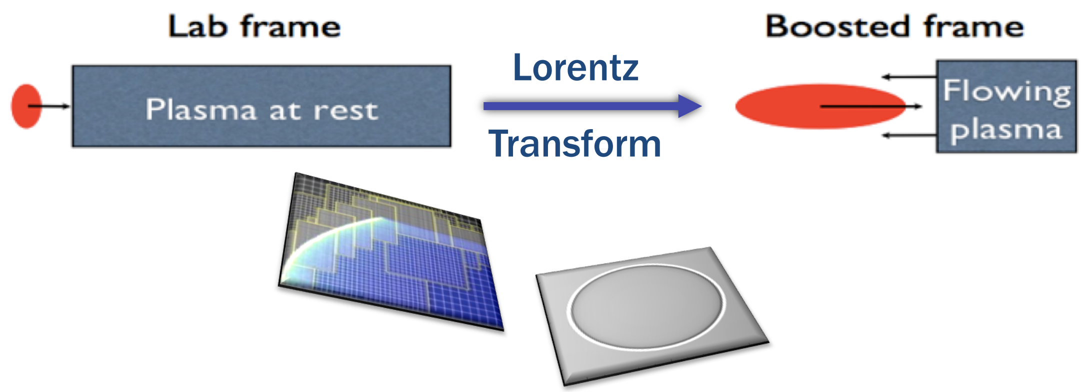
Advanced algorithms are essential to reach goal

Lower # time steps:

- optimal Lorentz boosted frame

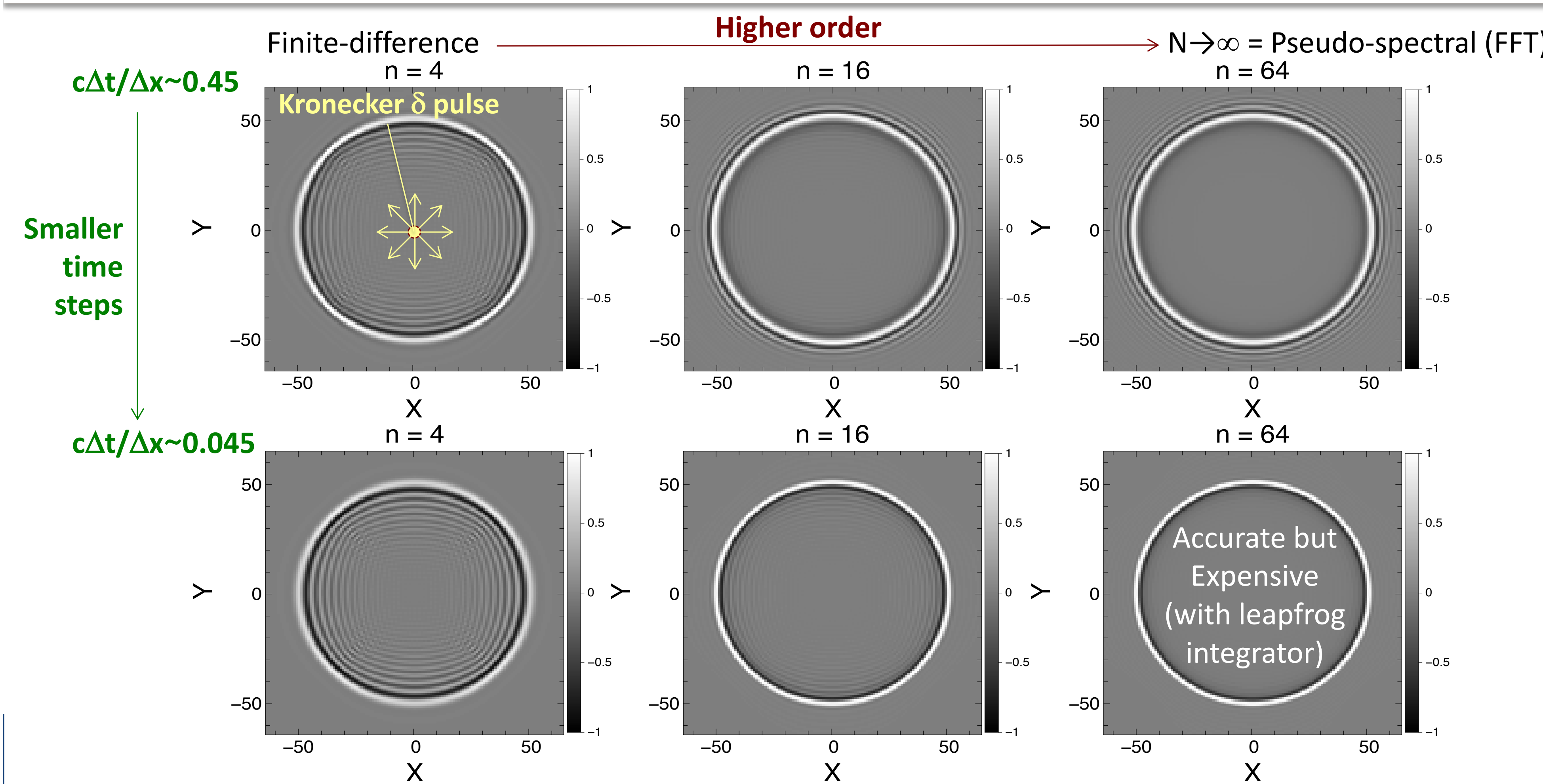
Higher accuracy:

- Adaptive Mesh Refinement
- Ultrahigh-orders spectral Maxwell solvers



Arbitrary-order Maxwell solver offers flexibility in accuracy

(on centered or staggered grids)

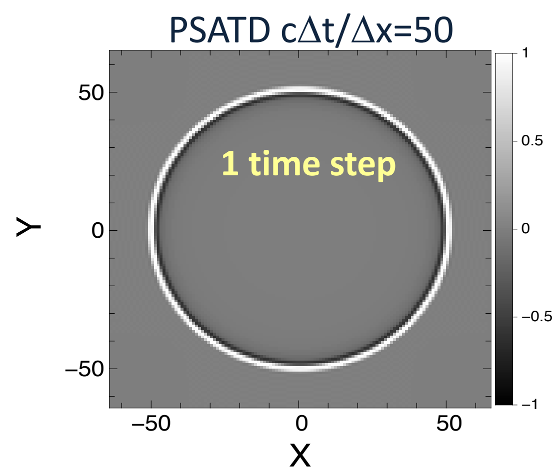


Analytical integration in Fourier space offers infinite order

Pseudo-Spectral Analytical Time-Domain[1] (PSATD)

$$B_z^{n+1} = \mathcal{F}^{-1}\left(C\,\mathcal{F}\left(B_z^n\right)\right) + \mathcal{F}^{-1}\left(iSk_y\ \mathcal{F}\left(E_x\right)\right) - \mathcal{F}^{-1}\left(iSk_x\ \mathcal{F}\left(E_y\right)\right)$$

with $$C = \cos\left(kc\Delta t\right); \quad S = \sin\left(kc\Delta t\right); \quad k = \sqrt{k_x^2 + k_y^2}$$



Easy to implement arbitrary-order *n* with PSATD ($k=k^{\infty} \rightarrow k^{n}$).

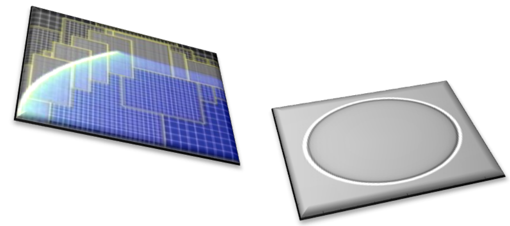
Both arbitrary order FDTD and PSATD to be implemented in WarpX.

[1] I. Haber,R. Lee,H. Klein & J. Boris,*Proc. Sixth Conf. on Num. Sim. Plasma,* Berkeley, CA, 46-48 (1973)

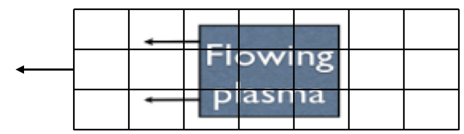
Advanced algorithms are essential to reach goal

Lower # time steps:
- optimal Lorentz boosted frame

Higher accuracy:
- Adaptive Mesh Refinement
- Ultrahigh-orders spectral Maxwell solvers



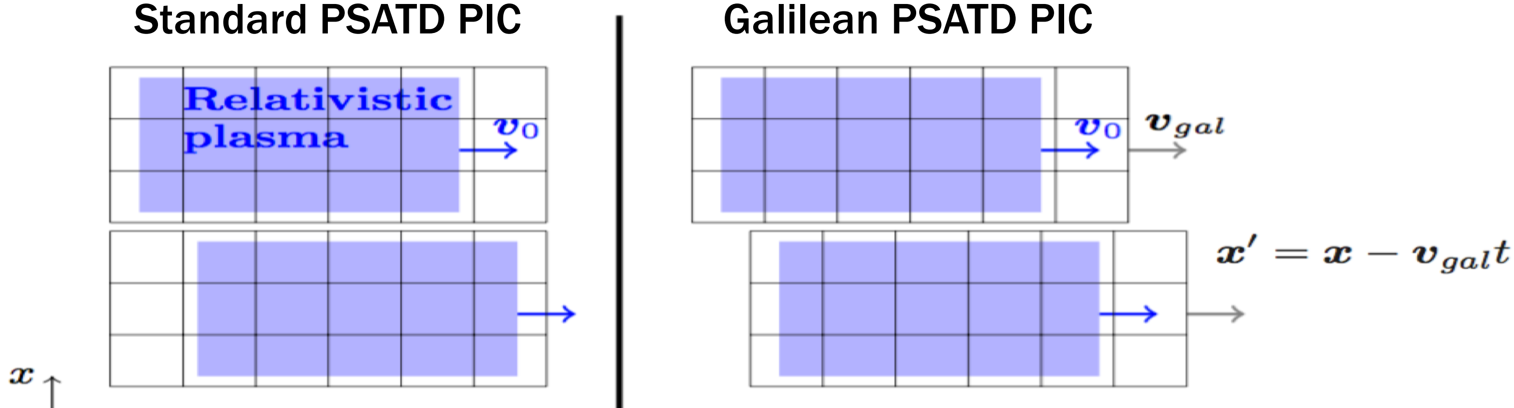
Higher stability
- Galilean T. to suppress Numerical Cherenkov Instability



PSATD also enables integration in Galilean frame

Use Galilean coordinates that follow the relativistic plasma.

Standard PSATD PIC | **Galilean PSATD PIC**

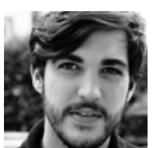


Time

$$\frac{\partial \boldsymbol{B}}{\partial t} = -\nabla \times \boldsymbol{E}$$
$$\frac{1}{c^2}\frac{\partial \boldsymbol{E}}{\partial t} = \nabla \times \boldsymbol{B} - \mu_0 \boldsymbol{j}$$

$$\left(\frac{\partial}{\partial t} - \boldsymbol{v}_{gal}\cdot\boldsymbol{\nabla}'\right)\boldsymbol{B} = -\boldsymbol{\nabla}' \times \boldsymbol{E}$$
$$\frac{1}{c^2}\left(\frac{\partial}{\partial t} - \boldsymbol{v}_{gal}\cdot\boldsymbol{\nabla}'\right)\boldsymbol{E} = \boldsymbol{\nabla}' \times \boldsymbol{B} - \mu_0 \boldsymbol{j}$$

+ integrate analytically, assuming $\boldsymbol{j}(\boldsymbol{x}, t)$ $\boldsymbol{j}(\boldsymbol{x}', t)$ is constant over one timestep.



Original idea by Manuel Kirchen (PhD student at U. Hamburg)

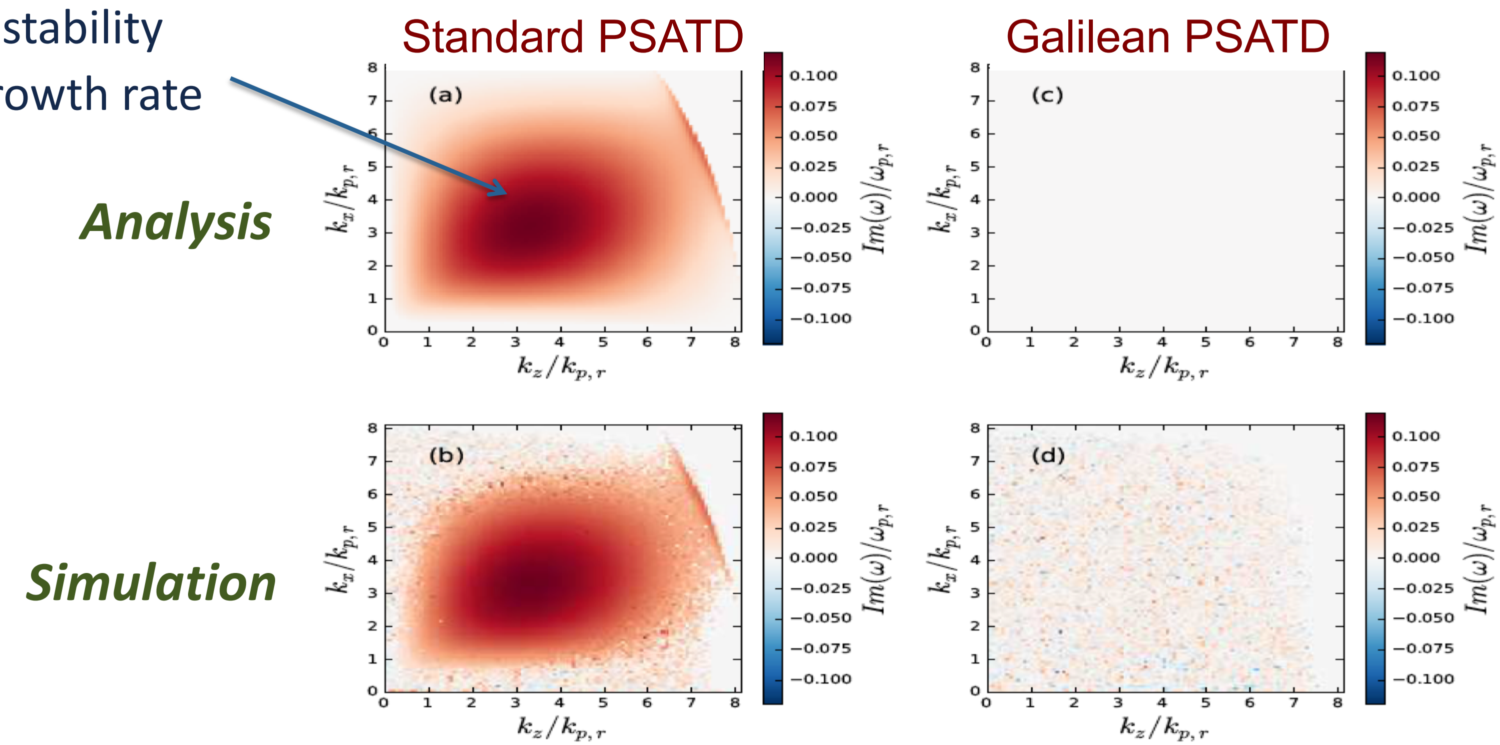
Concept and applications: *Kirchen et al., Phys. Plasmas 23, 100704 (2016)*

Derivation of the algorithm: *Lehe et al., Phys. Rev. E 94, 053305 (2016)*

Galilean PSATD is stable for uniform relativistic flow

Uniform plasma streaming in 2D periodic box

Instability growth rate



Lehe et al., Phys. Rev. E 94, 053305 (2016)

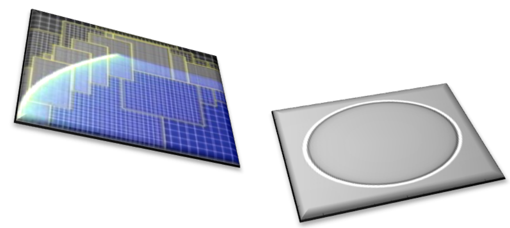
Advanced algorithms are essential to reach goal

Lower # time steps:

- optimal Lorentz boosted frame

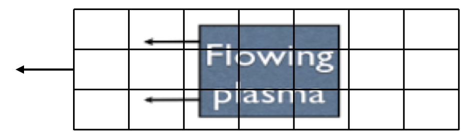
Higher accuracy:

- Adaptive Mesh Refinement
- Ultrahigh-orders spectral Maxwell solvers



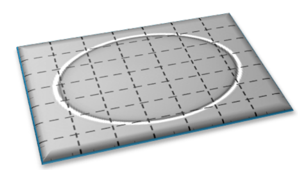
Higher stability

- Galilean T. to suppress Numerical Cherenkov Instability

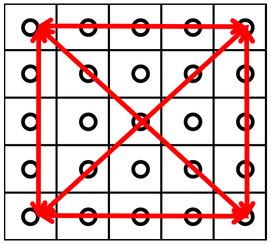
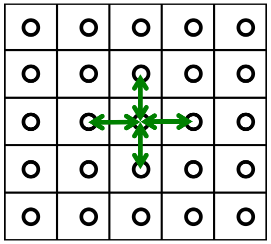


Higher scalability

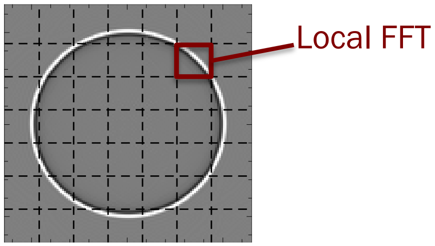
- FFT Maxwell solvers on local domains + domain decomposition



Spectral solvers involve global operations ➔ harder to scale to large # of cores

Spectral		Finite Difference (FDTD)
global "costly" communications		local "cheap" communications
	vs	
Harder to scale		Easier to scale

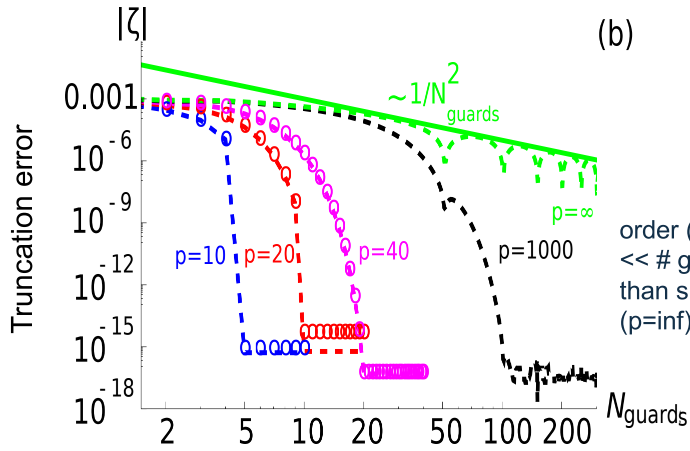
Finite speed of light ➔ local FFTs ➔ spectral accuracy+FDTD scaling!



J.-L. Vay, I. Haber, B. Godfrey, *J. Comput. Phys.* **243**, 260 (2013)
H. Vincenti, J.-L. Vay, *Comput. Phys. Comm.* **200**, 147 (2016)

Finite-order stencil offers scalable ultra-high order solver

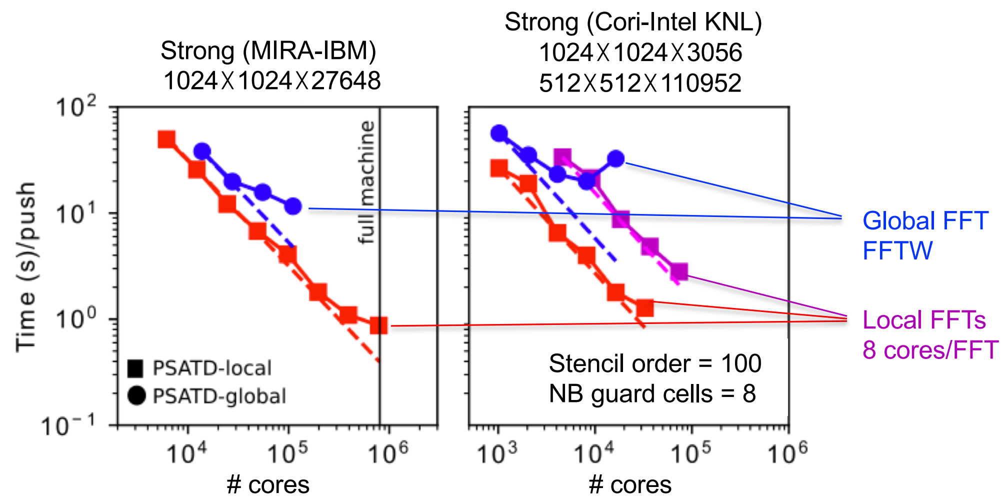
Truncation error analysis ➔ ultra-high order possible with much improved stability



order (p) requires << # guard cells than spectral (p=inf).

H. Vincenti et al., *Comput. Phys. Comm.* 200, 147 (2016).

Enabled demonstration of novel spectral solver with local FFTs scaling to ~1M cores



H. Vincenti, J.-L. Vay, Comput. Phys. Comm., in press

Applied successfully to modeling of LPAs at DESY[1] and plasma mirrors at CEA Saclay[2,3] in cases where standard second-order FDTD solvers fail.

[1] S. Jalas, I. Dornmair, R. Lehe, H. Vincenti, J.-L. Vay, M. Kirchen, A. R. Maier, *Phys. Plasmas* **24**, 033115 (2017).
[2] G. Blaclard, H. Vincenti, R. Lehe, J. L. Vay, Phys. Rev. E **96**, 033305 (2017)
[3] A. Leblanc, S. Monchoce, H. Vincenti,S. Kahaly, J-L. Vay, F.Quere, *Phys. Rev. Lett.* **119**, 155001 (2017)

Outline

- Intro & overview of the BLAST toolkit
- Advanced algorithms
- **BLAST codes for modeling of plasma accelerators: Warp, FBPIC, WarpX.**
- Examples
- Summary & outlook

BLAST codes for modeling plasma acceleration

	Warp	FBPIC	WarpX
Geometries			
Processors			
Maxwell solvers			
Adaptive mesh refinement			
Particle pushers			
Ionization			
Run modes			
Conventional accelerators			
Internal surfaces/volumes			
Particle emission			
Usage			

[] = in/future development

RZ = axisymmetric

RZ^{++} = RZ + Fourier azimuthal decomposition $\cos(m\theta)+i\,\sin(m\theta)$

Warp

open source: http://warp.lbl.gov

History

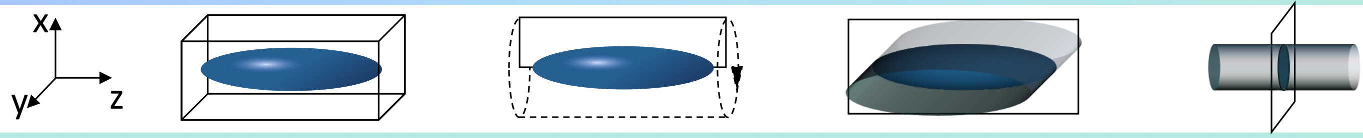
- **1989:** code started by **A. Friedman**; electrostatic PIC with 3D Poisson solver in square pipe using sine transforms
  - with inputs and contributions from I. Haber.
  - name Warp chosen to denote speed, later also denotes "warped" Fresnet-Serret coordinates used in bends.
- **1991**: **D. Grote** 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 XYZ | axisym. RZ, $RZ^{m\theta}$ | 2-D XZ | 2-D XY

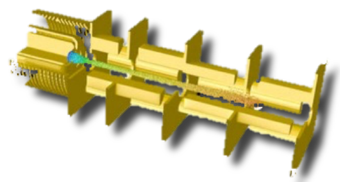


- **Reference frame:** **lab** | **moving-window** | **Lorentz boosted**

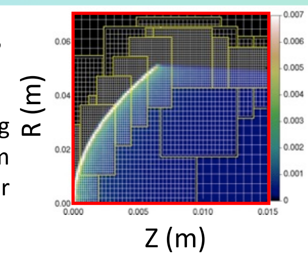
  z | z-vt | $\gamma(z-vt)$; $\gamma(t-vz/c^2)$

- **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

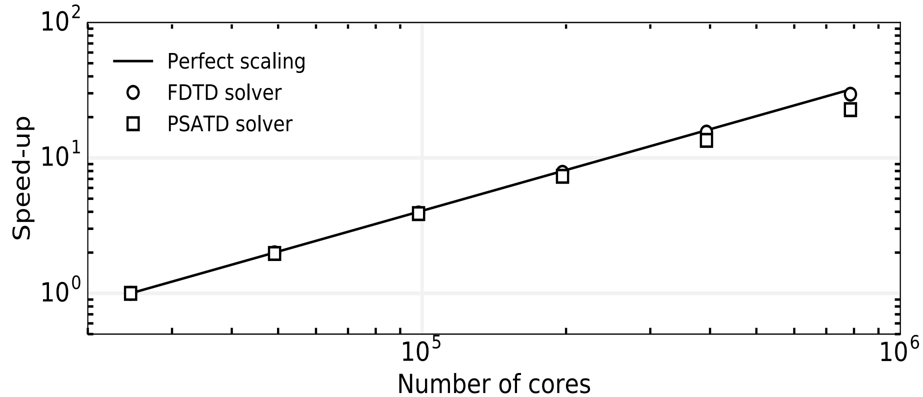
- **Accelerator lattice:** general; non-paraxial; can read MAD files
  - solenoids, dipoles, quads, sextupoles, linear maps, arbitrary fields, acceleration.

- **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

- **Parallelization:** 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

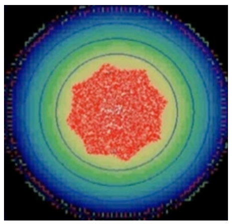
*http://hifweb.lbl.gov/Forthon (wrapper supports FORTRAN90 derived types) – dpgrote@lbl.gov

Warp's versatile programmability enables great adaptability

Standard PIC

Laboratory frame

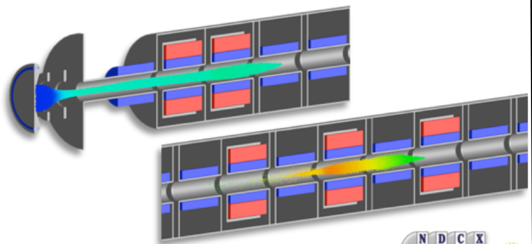
Alpha anti-H trap



Moving window

Example:

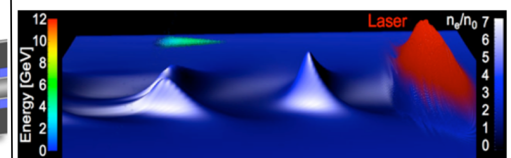
Beam generation and transport



Lorentz Boosted frame

Example:

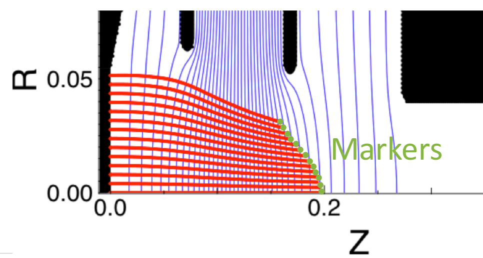
Laser plasma acceleration



Non-standard PIC

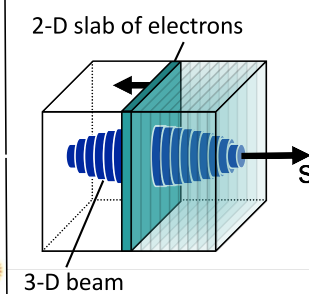
Steady flow

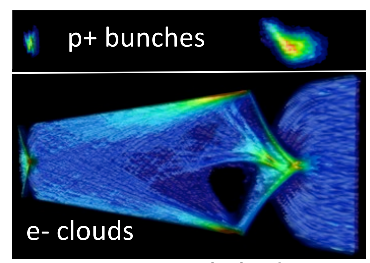
Example: Injector design



Quasi-static

Example: electron cloud studies





SPS - CERN

FBPIC

open source: https://github.com/fbpic

developed since 2015 by

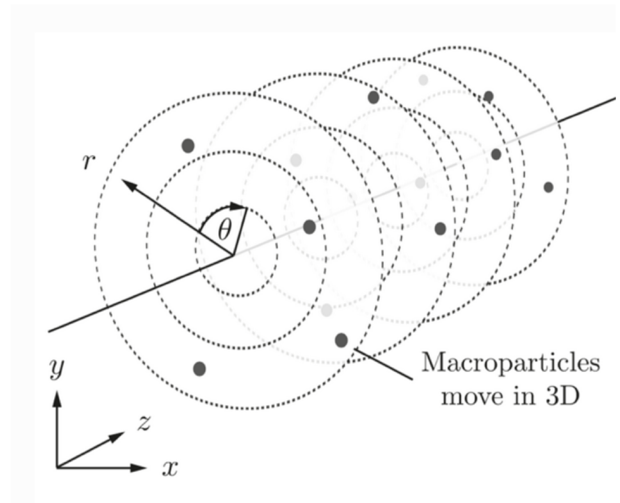
Rémi Lehe – Berkeley Lab, USA
Manuel Kirchen & Soeren Jalas – CFEL, Hamburg, Germany

Spectral EM-PIC in Cyl. RZ$^{m\theta}$ geometry for fast simulations

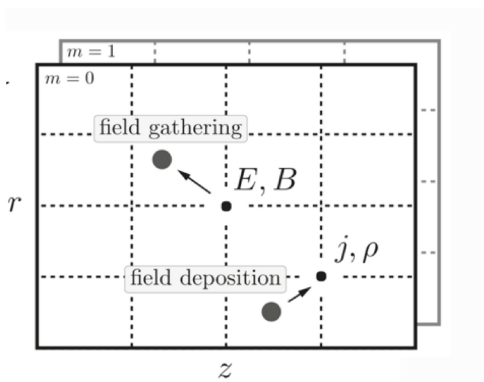
- **The fields are decomposed into azimuthal modes**

$$F(r,z,\theta) = Re\left[\sum_{m=0}^{N_m-1} \hat{F}_m(r,z)e^{im\theta}\right]$$

m=0: purely cylindrical mode
m=1: dipole mode
m=2: quadrupole mode



- **Each azimuthal mode is represented by complex 2D r-z grids**



Derivatives in Maxwell equations evaluated in Fourier space

- **FFT transforms in Z**
- **Hankel transforms in R**

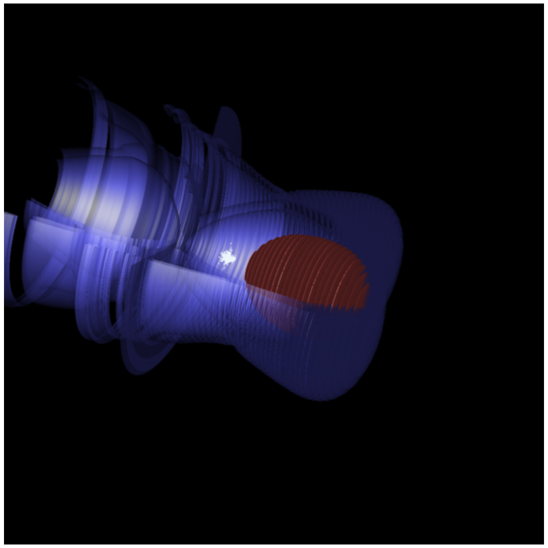
Cylindrical ($RZ^{m\theta}$) geometry: faster simulations

Key advantage

For many physical problems, **only a few modes** are needed.

Using only a few modes (2D grids) requires **vastly less computational time & memory** than a full 3D Cartesian grid.

- **Beam-driven plasma acceleration with <u>cylindrical</u> driver beam**
  m=0
- **Beam hosing <u>in the weakly-perturbed regime</u>**
  m=0 (unperturbed beam)
  and m=1 (first-order perturbation)
- **Laser-driven plasma acceleration with <u>cylindrical</u> laser**
  m=0 (for the wakefield)
  and m=1 (for the laser)

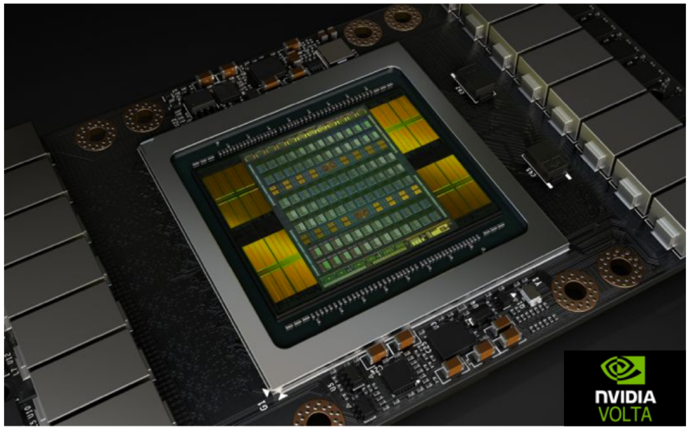


GPU acceleration

In $RZ^{m\theta}$ geometry, many simulations can often fit on a **single GPU**

Whole PIC loop was ported to GPU
(to avoid CPU-GPU transfer)

Acceleration of **~3x-10x** compared
to the multi-threaded CPU version
(for large simulations)



WarpX

open source release: fall 2018.

(pre-release available to collaborators)

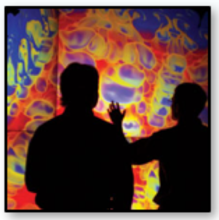
U.S. DOE Exascale Computing Project (ECP)

- As part of the National Strategic Computing initiative, ECP was established to accelerate delivery of a **capable exascale** computing system that integrates hardware and software capability to deliver approximately 50 times more performance than today's 20-petaflops machines on mission critical applications.
  - DOE is a lead agency within NSCI, along with DoD and NSF
  - Deployment agencies: NASA, FBI, NIH, DHS, NOAA
- ECP's work encompasses
  - applications,
  - system software,
  - hardware technologies and architectures, and
  - workforce development to meet scientific and national security mission needs.

Capable Exascale System Applications Will Deliver Broad Coverage of 6 Strategic Pillars

National security

Stockpile stewardship



Energy security

Turbine wind plant efficiency

Design and commercialization of SMRs

Nuclear fission and fusion reactor materials design

Subsurface use for carbon capture, petro extraction, waste disposal

High-efficiency, low-emission combustion engine and gas turbine design

Carbon capture and sequestration scaleup

Biofuel catalyst design

Economic security

Additive manufacturing of qualifiable metal parts

Urban planning

Reliable and efficient planning of the power grid

Seismic hazard risk assessment

Scientific discovery

Cosmological probe of the standard model of particle physics

Validate fundamental laws of nature

Plasma wakefield accelerator design

Light source-enabled analysis of protein and molecular structure and design

Find, predict, and control materials and properties

Predict and control stable ITER operational performance

Demystify origin of chemical elements

Earth system

Accurate regional impact assessments in Earth system models

Stress-resistant crop analysis and catalytic conversion of biomass-derived alcohols

Metagenomics for analysis of biogeochemical cycles, climate change, environmental remediation

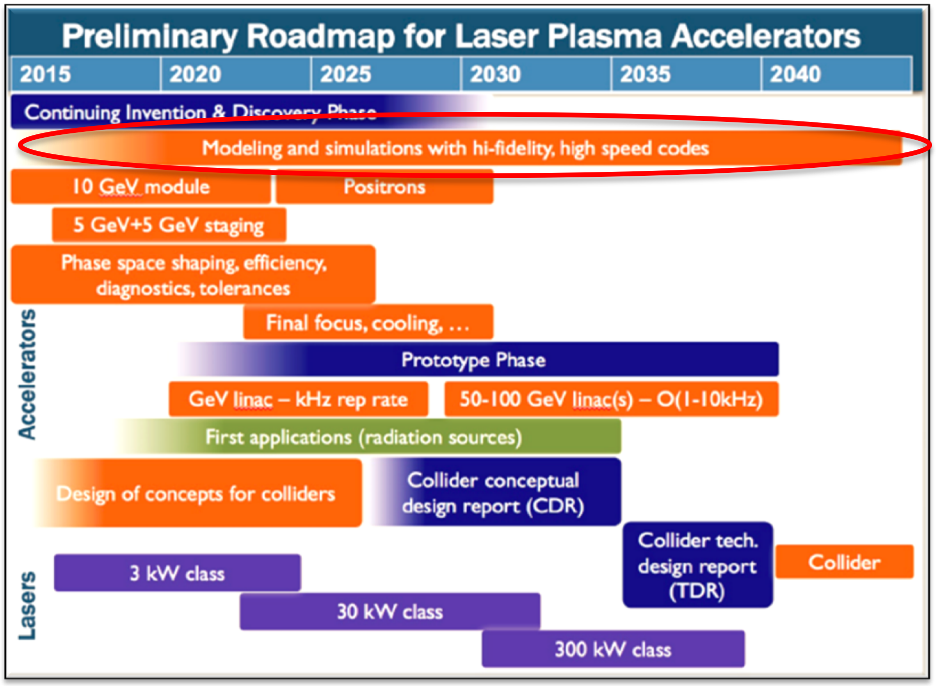
Health care

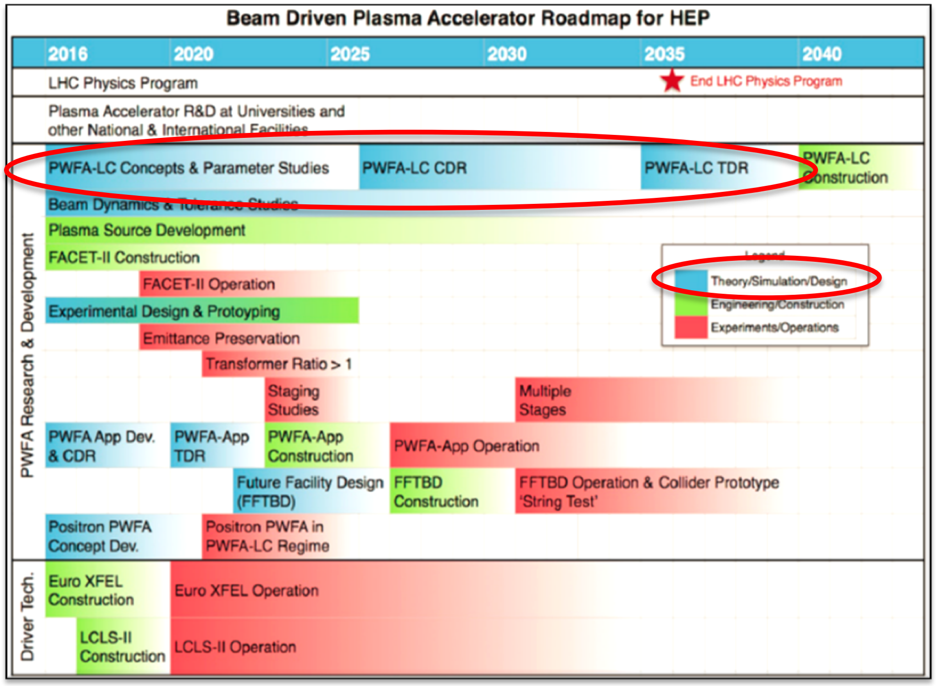
Accelerate and translate cancer research

EXASCALE COMPUTING PROJECT

BERKELEY LAB

To deliver simulation tools for plasma-based collider design by 2035-2040 – in sync with recent community roadmaps.



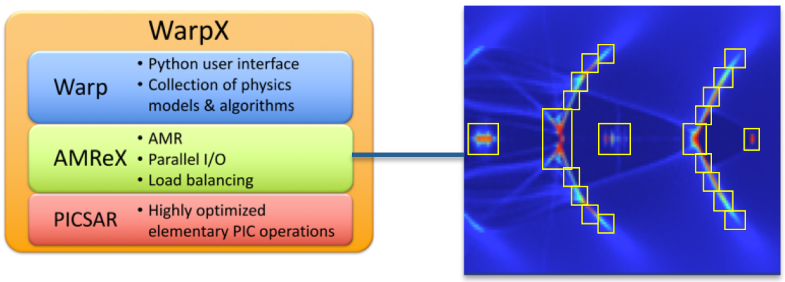


ECP Project WarpX: "Exascale Modeling of Advanced Particle Accelerators"

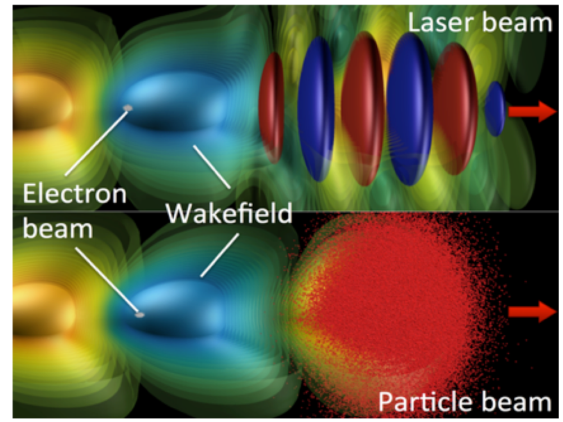
Goal (4 years): Convergence study in 3-D of 10 consecutive multi-GeV stages in linear and bubble regime, for laser- & beam-driven plasma accelerators.

How: ➔ Combination of most advanced algorithms

➔ Coupling of Warp+AMReX+PICSAR



➔ Port to emerging architectures (Intel KNL, GPU, ...)



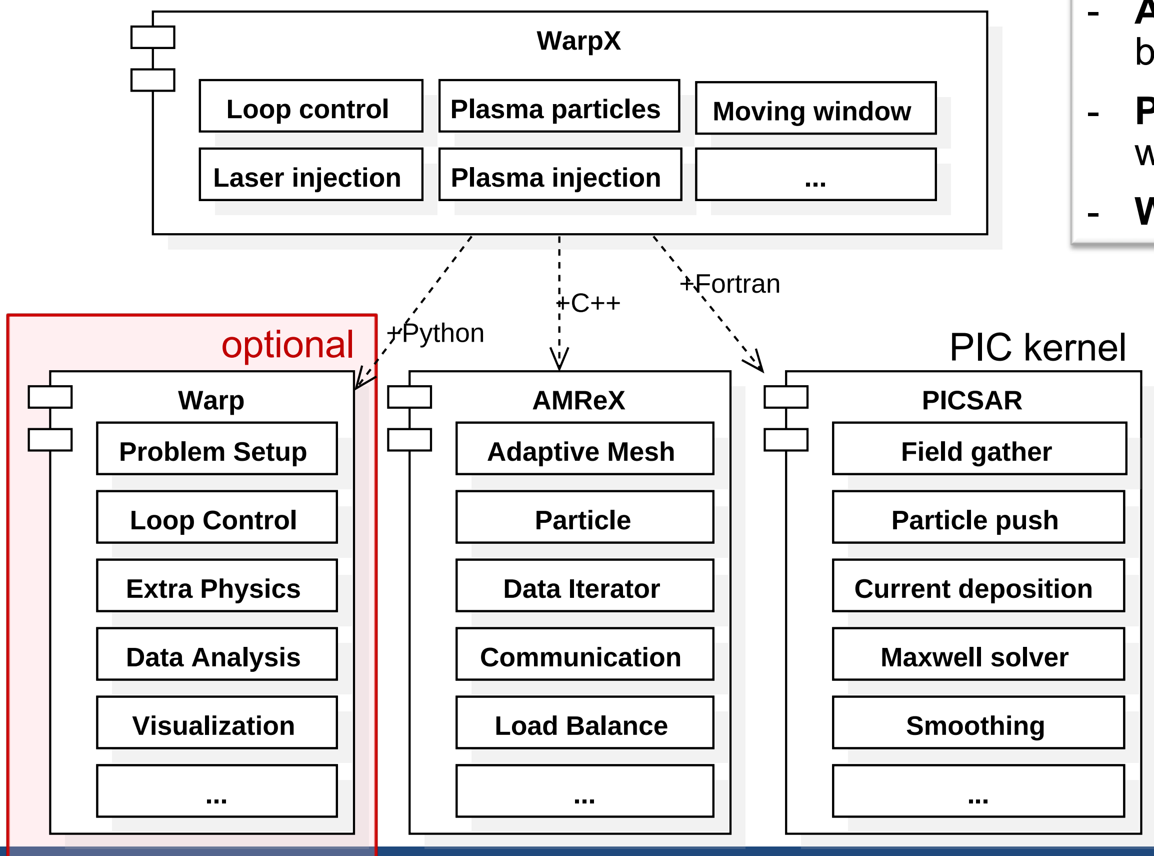
EXASCALE COMPUTING PROJECT

NNSA U.S. DEPARTMENT OF ENERGY Office of Science

Team: LBNL (accelerators + computing science divisions) + SLAC + LLNL

Ultimate goal: enable modeling of 100 stages by 2025 for 1 TeV collider design!

WarpX code structure

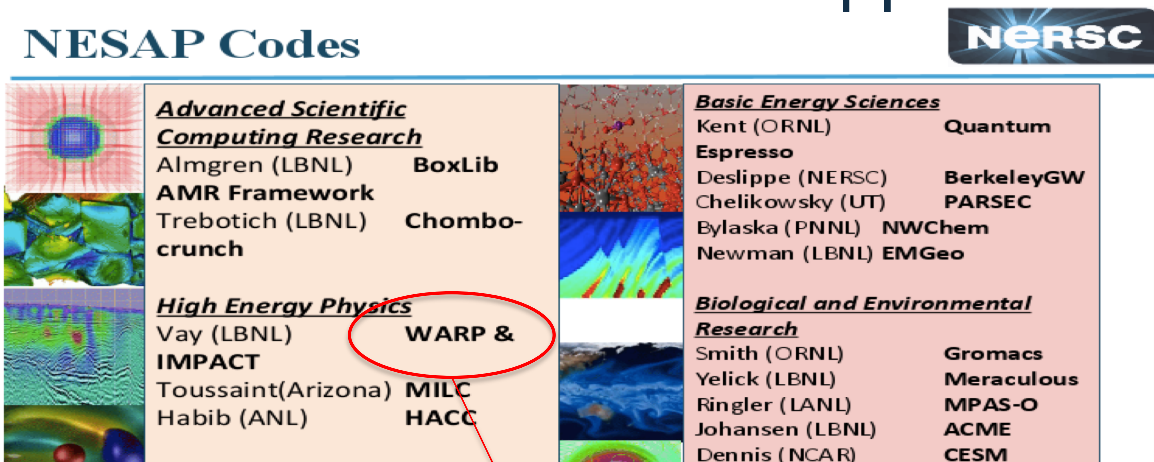


- **WarpX source**: particles/fields data structures, loop control, problem setup, I/O, etc.
- **AMReX**: adaptive grid, parallelism & load balancing handling.
- **PICSAR**: PIC kernel library developed jointly with CEA Saclay.
- **Warp**: preexisting advanced PIC algorithms.

WarpX can be run in 3 modes:

1. Compiled languages only: for supercomputers with no/limited support of Python.
2. Compiled languages + Python: main in Python
   a. Standalone.
   b. With Warp.

PICSAR created as part of NERSC Exascale Applications Program (NESAP)



Mathieu Lobet

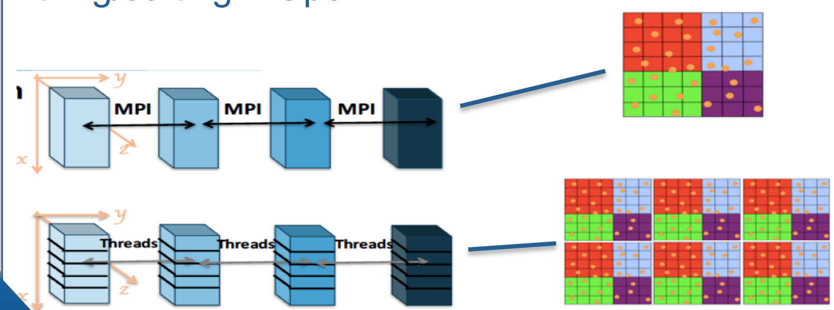
Ex-NESAP postdoc
(now at CEA, Saclay, France)

Henri Vincenti

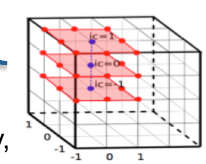
Marie Curie postdoc fellowship
(now at CEA, Saclay, France)

Warp EM-PIC kernel extracted
➔ Particle-In-Cell Scalable Architecture Resources (PICSAR) library + miniapp

Optimized with new vectorization algo.* + tiling/sorting + OpenMP + MPI




```
|--> DO I = 1, N
|       R(I) = B(I) + A(I)
|--> ENDDO
```



*H. Vincenti, R. Lehe, R. Sasanka and J.-L. Vay, *Comp. Phys. Comm.* , 210, 145-154 (2017).

PICSAR is now open source: https://picsar.net

Used with Warp, WarpX & SMILEI.

Plan for Exascale project WarpX for existing ECP project & beyond

From initial code coupling to ensemble of 100 GeV-scale stages

- **2030** — Collider designs
- **2023** — Modeling of ensemble of 100 consecutive GeV-scale plasma accelerator stages
- **2022** — Modeling of 100 consecutive GeV-scale plasma accelerator stages
- **2021** — Modeling of 30 consecutive GeV-scale plasma accelerator stages

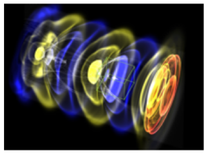
End current ECP project

- **2020** — Modeling of 10 consecutive GeV-scale plasma accelerator stages — Mi. FYQ4
- **2019** — Modeling of 3 consecutive GeV-scale plasma accelerator stages — Mi. FY19Q4
- **2018** — Modeling a single GeV-scale plasma accelerator stage — Mi. FY18Q4
- **2017** — Modeling of single plasma accelerator stage with static mesh refinement — Mi. FY17Q4
- **2016** — Initial coupling of Warp+Boxlib+PICSAR — Mi. FY17Q1

2nd ECP ExaFlop system

A21 – 1 ExaFlops

Code to be released open source to community.



ECP
EXASCALE COMPUTING PROJECT

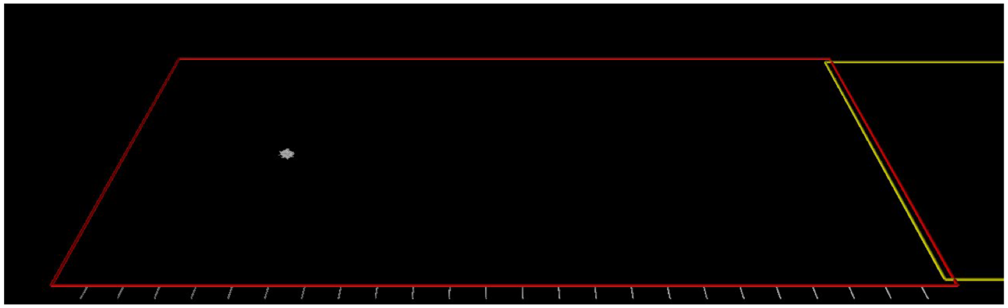
Outline

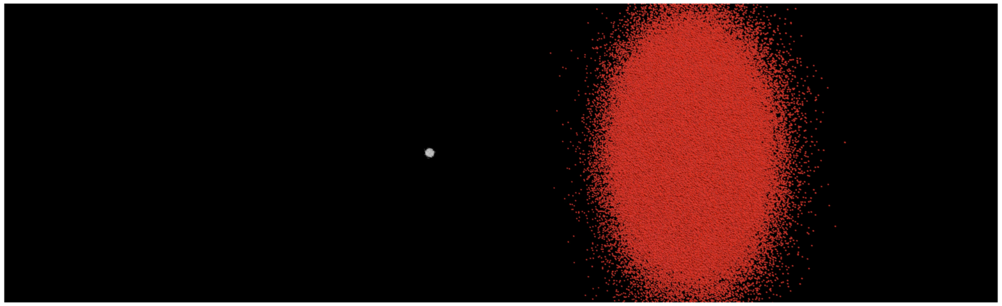
- Intro & overview of the BLAST toolkit
- Advanced algorithms
- BLAST codes for modeling of plasma accelerators: Warp, FBPIC, WarpX.
- **Examples**
- Summary & outlook

Warp example simulations of plasma accelerators

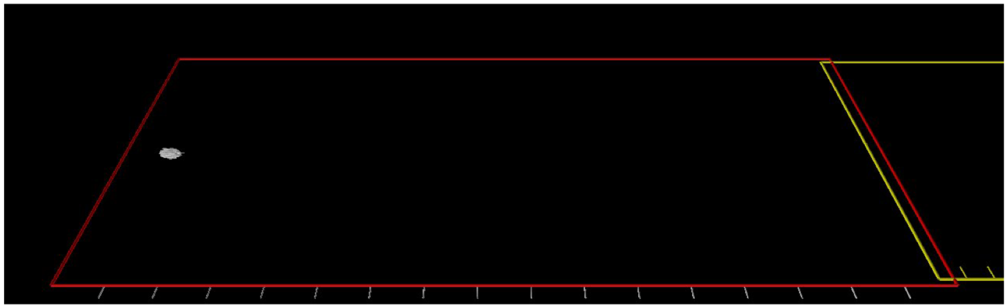
Lab frame

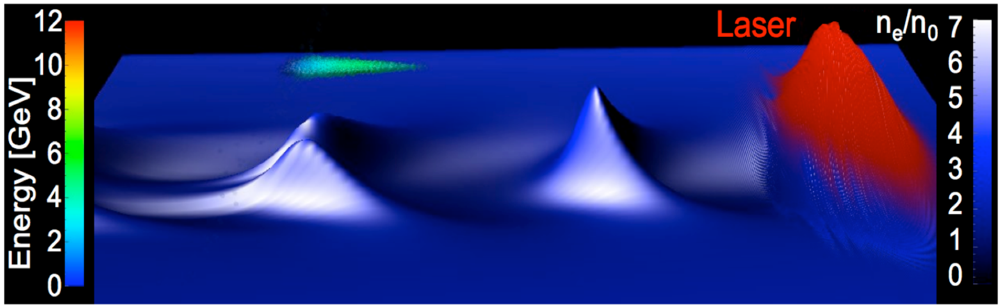
Laser driven | **Particle beam driven**





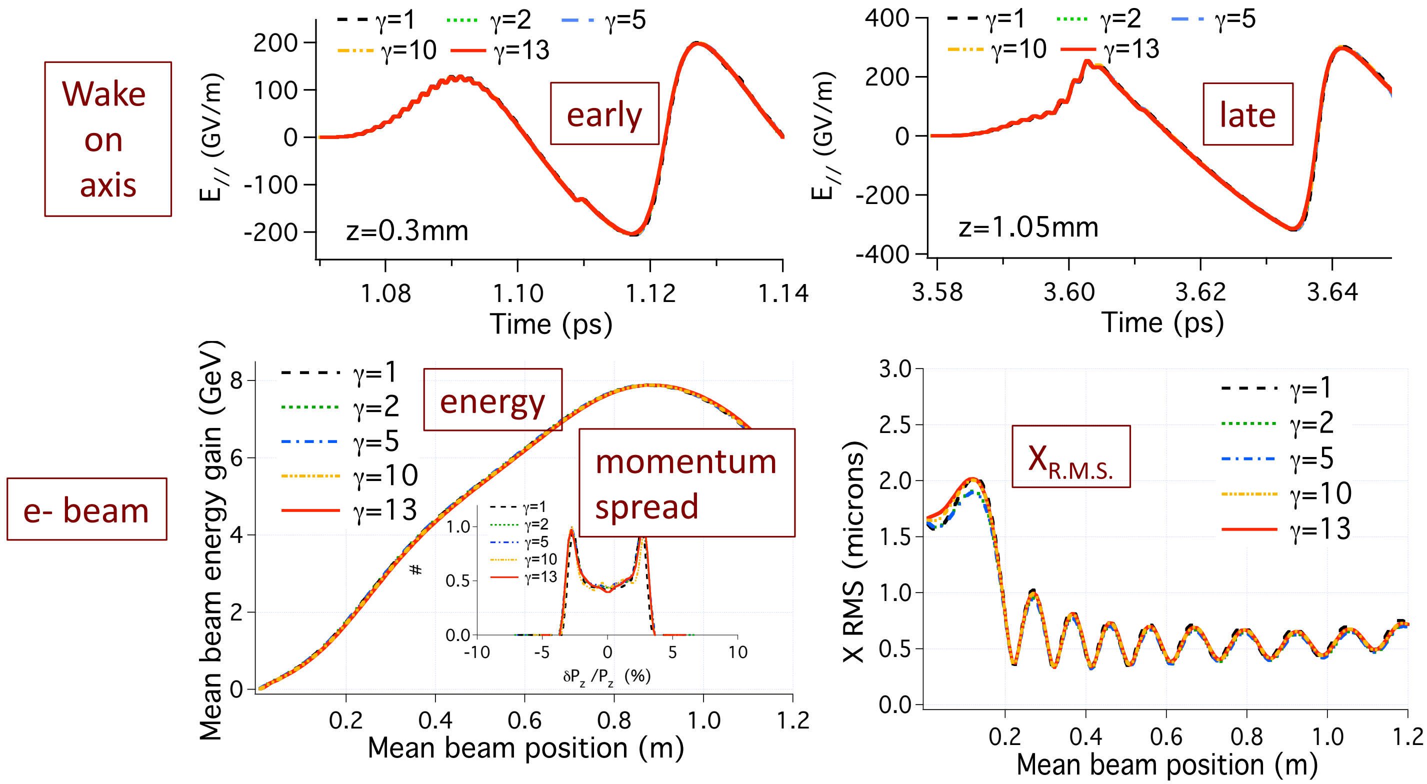
Boosted frame





Laser/beam plane injection enabled convergence for all γ boost

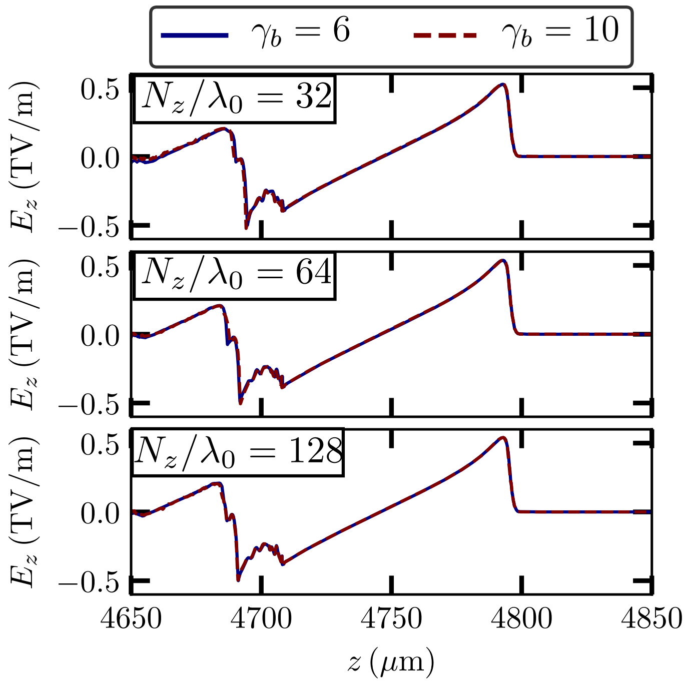
Warp-3D



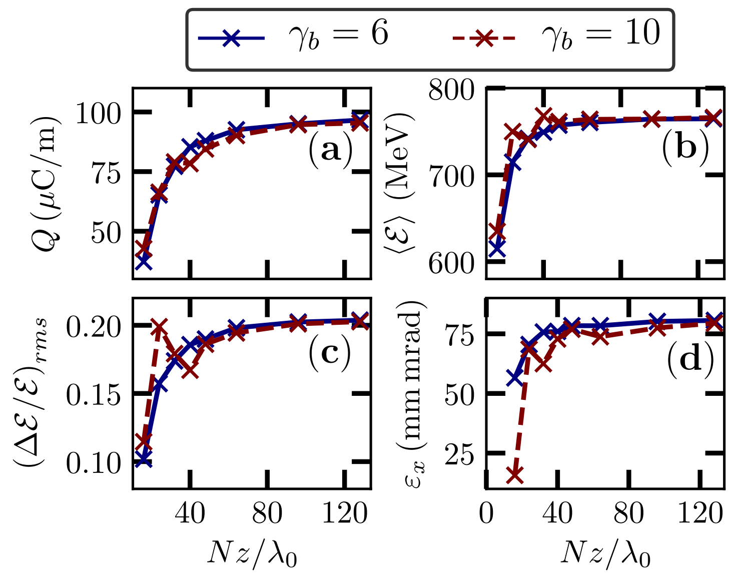
Parameters – a_0=1, n_0=10^{19}cm^{-3} (~100 MeV) scaled to 10^{17}cm^{-3} (~10 GeV).

Convergence demonstrated recently with self-injection

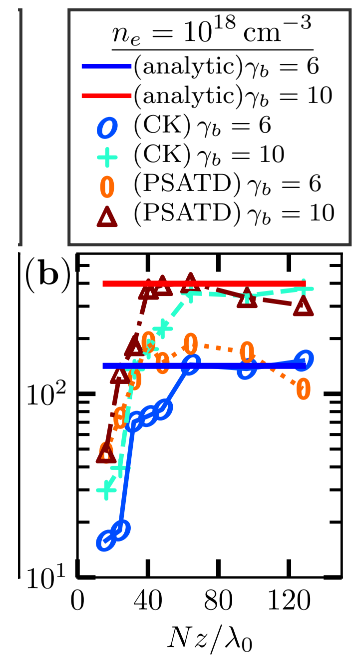
Longitudinal electric field



Self-injected beam moments



Speedup



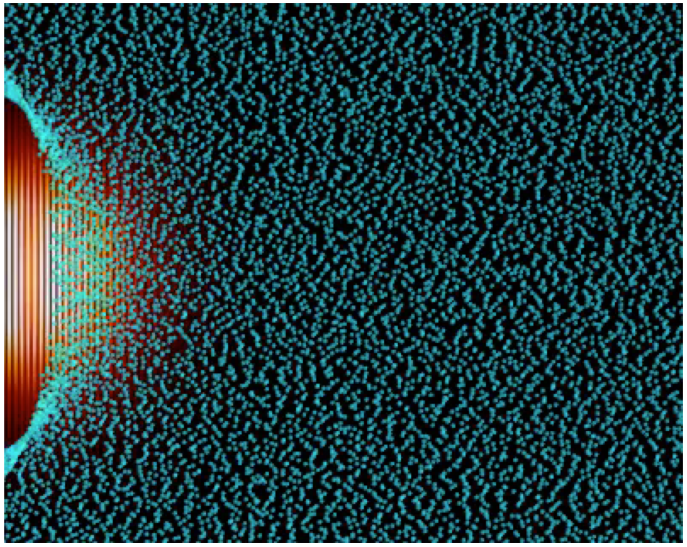
Parameters – a_0=8, n_0=10^{18}cm^{-3}

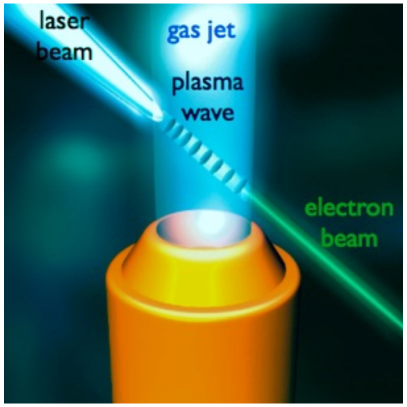
P. Lee, J.-L. Vay, arxiv 1803.01890 (2018)

Predicting injection in laser-plasma accelerators with FBPIC

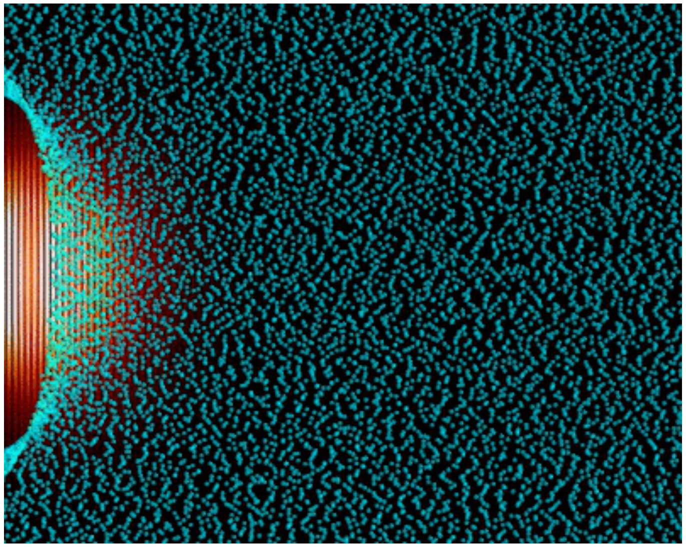
Depending on the parameters (gas density, laser intensity, …) **injection** may or may not happen.

No injection



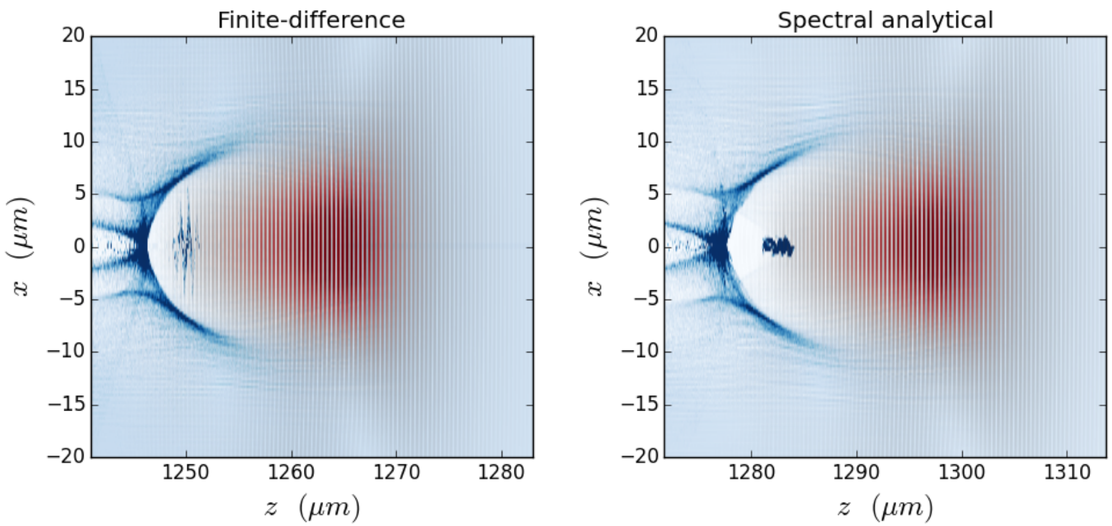


Injection

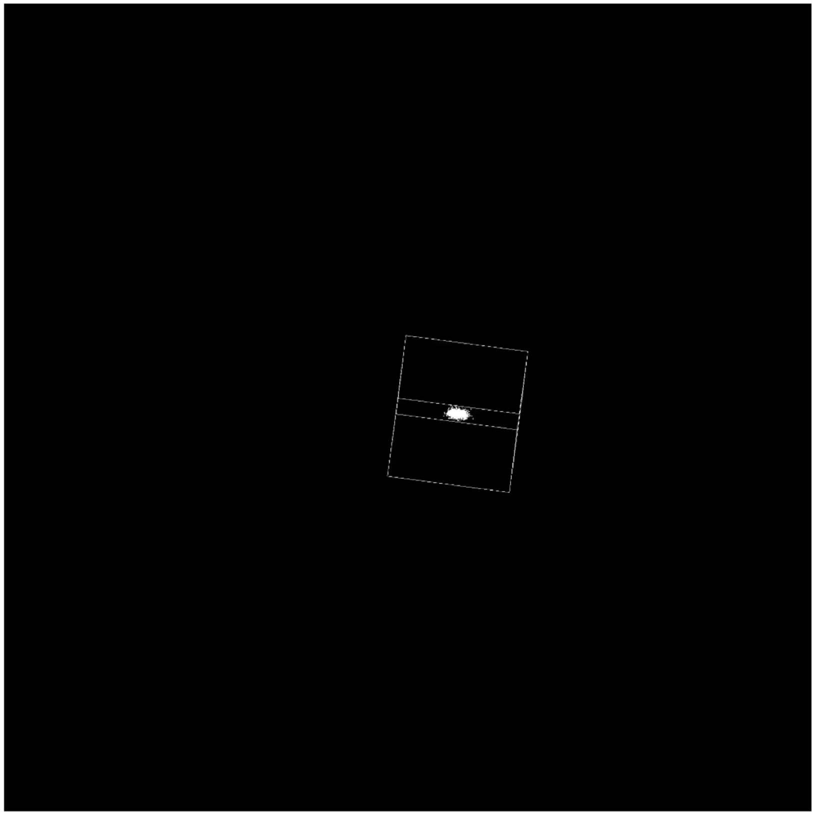


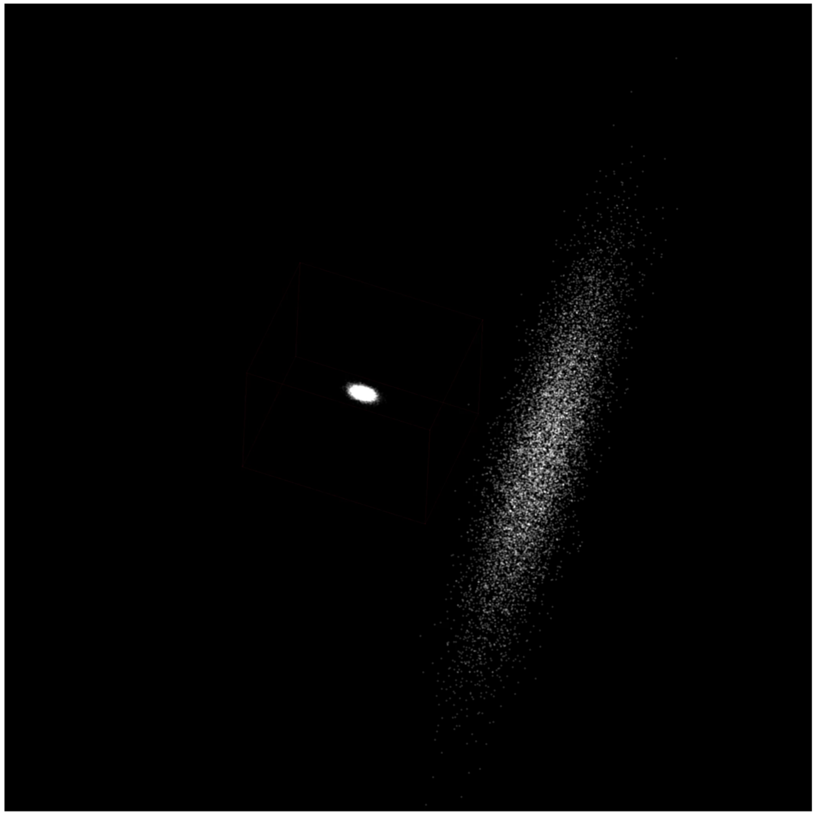
Illusttration with FBPIC of E+VxB cancelation with spectral EM PIC

- In finite-difference codes, E and B are usually staggered in time and space.
- This makes it difficult to accurately capture the **physical cancelation** between E and v x B
- This does not happen in **centered** spectral codes.



WarpX - first simulations of plasma accelerators with mesh refinement





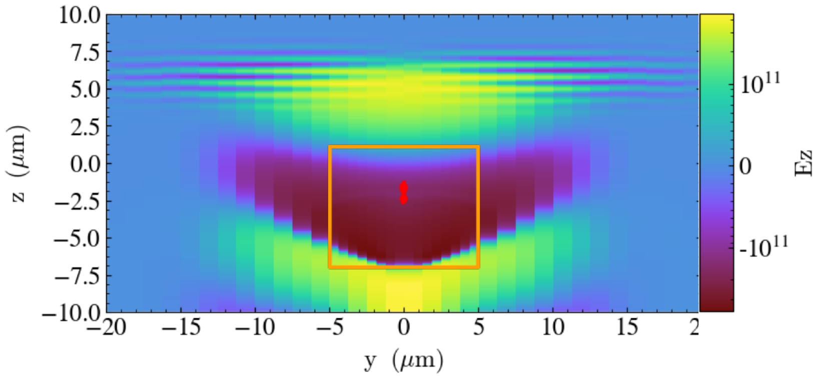
Movies by Maxence Thevenet

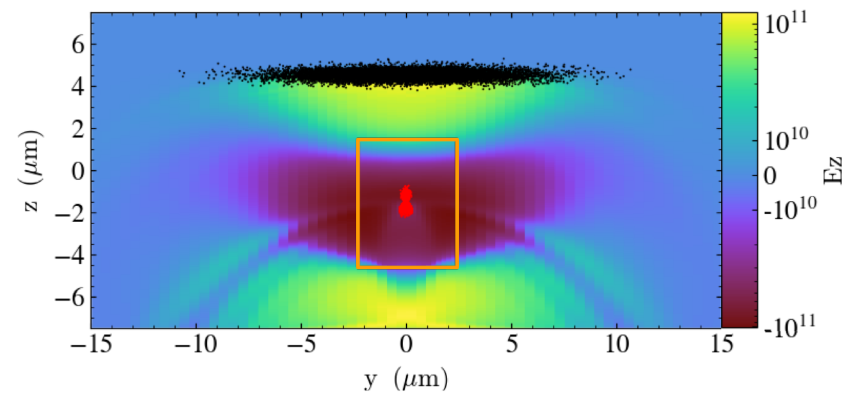
NERSC

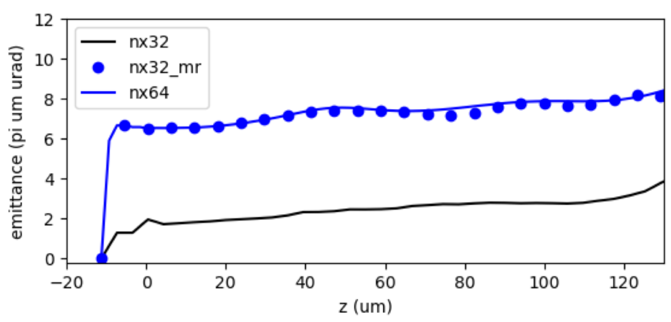
BERKELEY LAB | U.S. DEPARTMENT OF ENERGY | Office of Science | ACCELERATOR TECHNOLOGY & APPLIED PHYSICS DIVISION ATAP

Mesh refinement focuses resolution only where needed

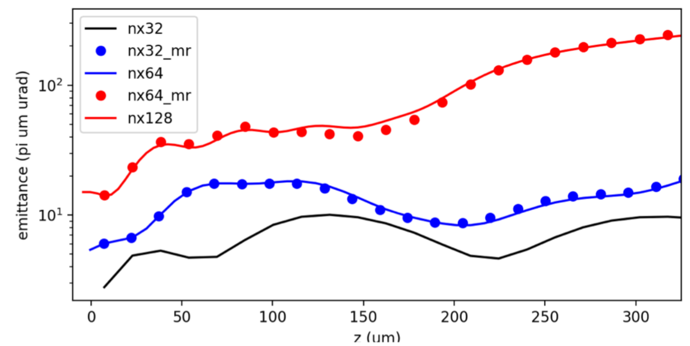
Laser driven | **3-D** | **Particle beam driven**







Simulations with small MR patch recover results using finer grid over the entire box.



J.-L. Vay, et al, *Nucl. Inst. Meth. A* in press (2018)

Outline

- Intro & overview of the BLAST toolkit
- Advanced algorithms
- BLAST codes for modeling of plasma accelerators: Warp, FBPIC, WarpX.
- Examples
- **Summary & outlook**

BLAST codes for modeling plasma acceleration

	Warp	FBPIC	WarpX
Geometries	3D, 2D-XZ/XY, $RZ^{m\theta}$	$RZ^{m\theta}$	3D, 2D-XZ, RZ, [$RZ^{m\theta}$]
Processors	CPU	CPU, GPU	CPU, [GPU]
Maxwell solvers	Finite-difference, spectral (arbitrary order: 2→∞)	Spectral Fourier-Bessel (arbitrary order: 2→∞)	Finite-difference, spectral (arbitrary order: 2→∞)
Adaptive mesh refinement	Some support	N/A	[Full support]
Particle pushers	Boris, Vay, Drift-Lorentz, [Higuera-Cary]	Vay	Boris, Vay
Ionization	ADK, impact ionization, capture, charge exchange	ADK	[ADK, impact]
Run modes	Full PIC, quasistatic, e-gun	full PIC	Full PIC
Conventional accelerators	linacs, rings, injectors	N/A	[inherit from Warp]
Internal surfaces/volumes	conductors, dielectrics	N/A	N/A
Particle emission	space charge limited, thermionic, arbitrary	N/A	N/A
Usage	Most general for multiphysics simulations	Fastest for quasi-cylindrical geometries	Large scale simulations or ensembles

[] = in/future development

RZ = axisymmetric

RZ^{++} = RZ + Fourier azimuthal decomposition $\cos(m\theta)+i\sin(m\theta)$

The three codes support laboratory, moving window and Boosted frames of reference, numerical Cherenkov suppression, I/O boosted-lab frames.

Analyzing/visualizing the output of the code

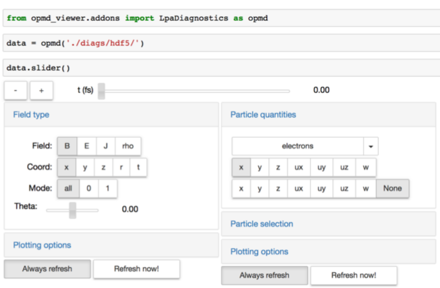
- **Codes output use the openPMD format**
  Standardized layout for HDF5 files, adopted/in adoption by several codes: ACE3P, BeamBeam3D, FBPIC, Impact, Osiris, PIConGPU, QuickPIC, Smilei, Synergia, Warp.

openpmd.org

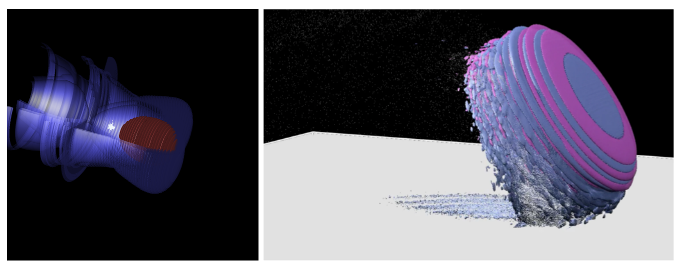
Initiated & led by Axel Huebl (HZDR, Germany)

- **Can leverage open-source visualization tools**

Python/Jupyter openPMD-viewer:
github.com/openPMD/openPMD-viewer



Plugins for Paraview, VisIt, Yt:
github.com/openPMD/openPMD-visit-plugin

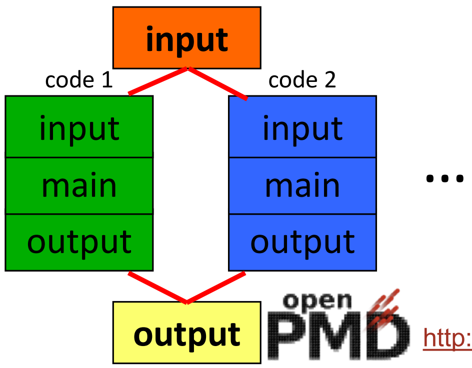


Beta PySide GUI:
ecp-warpx.github.io/visualization/pyside.html

Common input/output standards to ease usage of multiple codes

- Up to now, each code has own input script & output format

  ➔ user needs 1 input script/code & different data reader or software



http://www.openpmd.org

Bridge codes to enable:

- **unified** input/output interface
- **separation** of description/resolution/analysis

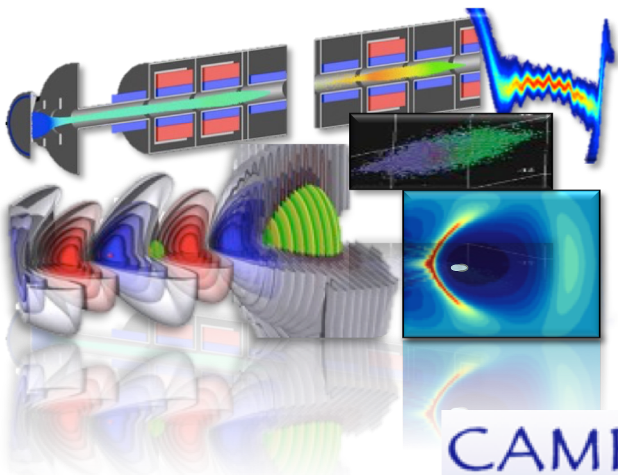
- In the process of defining standard for common input

  ➔ translate to individual code "language"

  - **PICMI**: Particle-In-Cell Modeling Interface
  - **AMI**: Accelerator Modeling Interface (aim to compatibility with MAD8/MAD-X)

Goal is to develop BLAST as integrated ecosystem
with interoperability with other codes in the community

blast.lbl.gov
BLAST
BERKELEY LAB ACCELERATOR SIMULATION TOOLKIT



CAMPA

Where BLAST simulation tools:

- ***Multi-physics frameworks***: IMPACT, Warp, WarpX.
- ***Specialized codes***: BeamBeam3D, FBPIC, LW3D, POSINST.
- ***Libraries***: PICSAR.

Share common Python interface & I/O formats

- ***Enabling integrated multi-physics simulations***
- ***Mixed conventional/plasma accelerator elements***
- ***Unified interfacing, data analysis & viz. tools***

Consortium for Advanced Modeling of Particle Acc.

- ***Collaboration between LBNL, SLAC, FNAL, UCLA is setting foundations for interoperability of larger pool of codes***
- ***Also collaborating with CEA Saclay, U. Dresden & Cornell U.***
- ***Other collaborators welcome***

Thank you

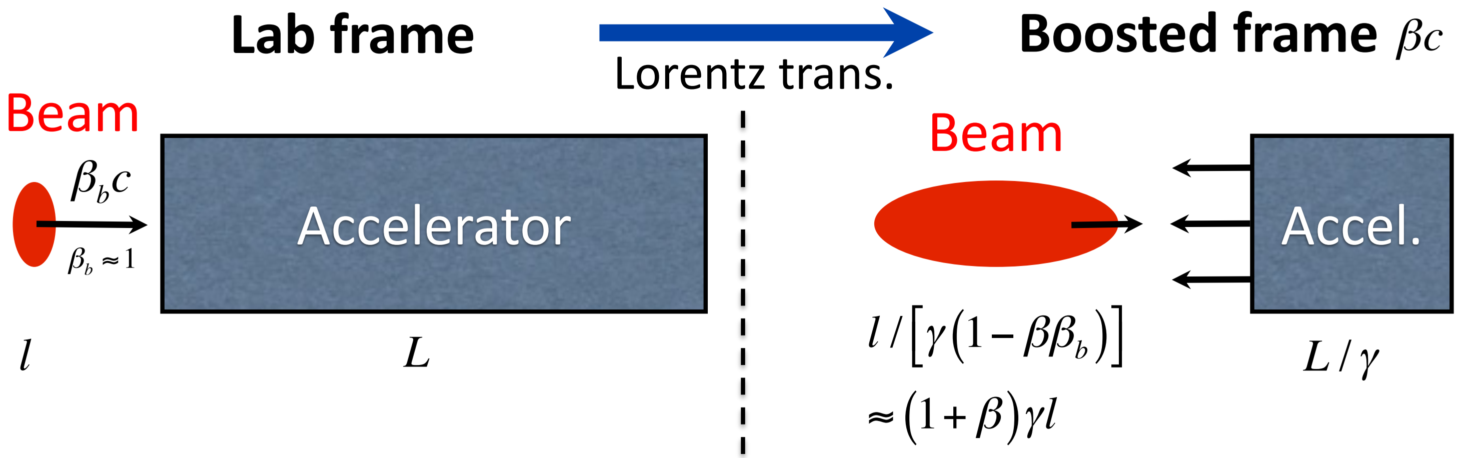
Extras

Standard would prescribe 4 steps & syntax for setting runs

```
Run parameters – can be in separate file
""""
Laser_waist = 5.e-6 # The waist of the laser (in meters)

""""
Physics part – can be in separate file
""""
import PICMI
laser = PICMI.Gaussian_laser(waist = Laser_waist,  …

""""
Numerics part – can be in separate file
""""
nx = 64
…

""""
PICMI input script (for codes with script-based front-end
""""
sim.step(max_step)
```

Separation in files enables sharing of sections, similarly to sharing mad files for lattices for example.

Optimal Lorentz boosted frame approach



Many time steps needed to follow:
short & **stiff** high-energy beam/laser
into
long accelerator filled with
fast reacting electron clouds/plasma.

Much less time steps needed to follow:
longer & **more mobile** low-energy beam/laser
into
shorter accelerator filled with
stiffer electron clouds.

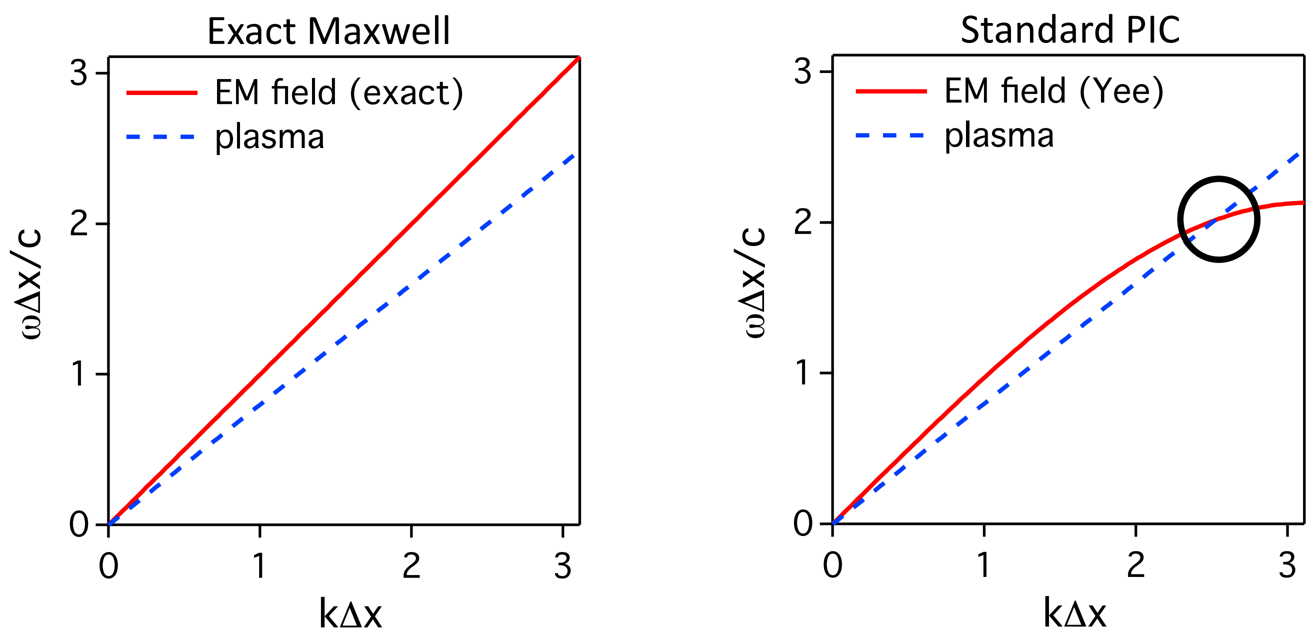
Number of time steps divided by $(1+\beta)\gamma^2$

With high γ, orders of magnitude speedups are possible.

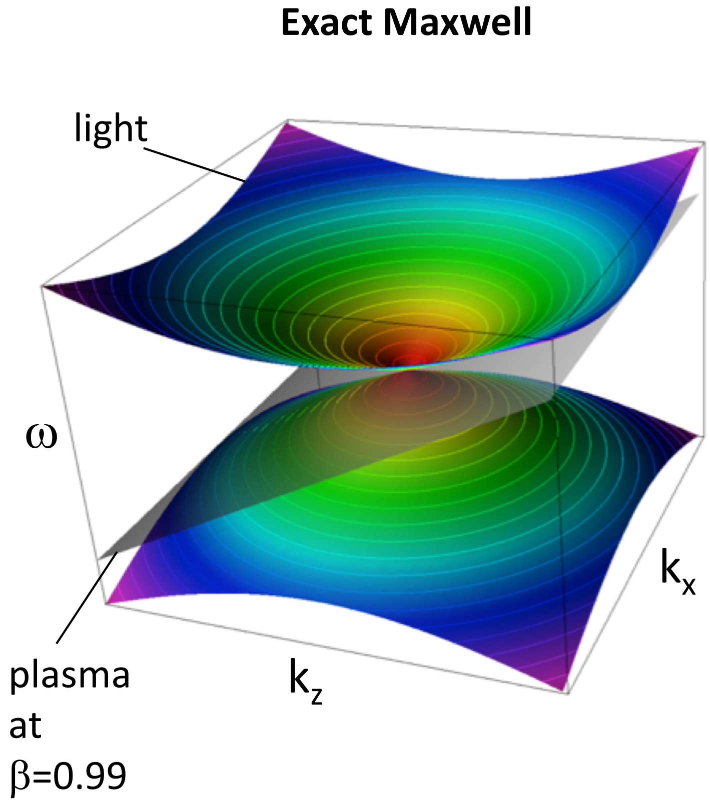
Relativistic plasmas PIC subject to "numerical Cherenkov"

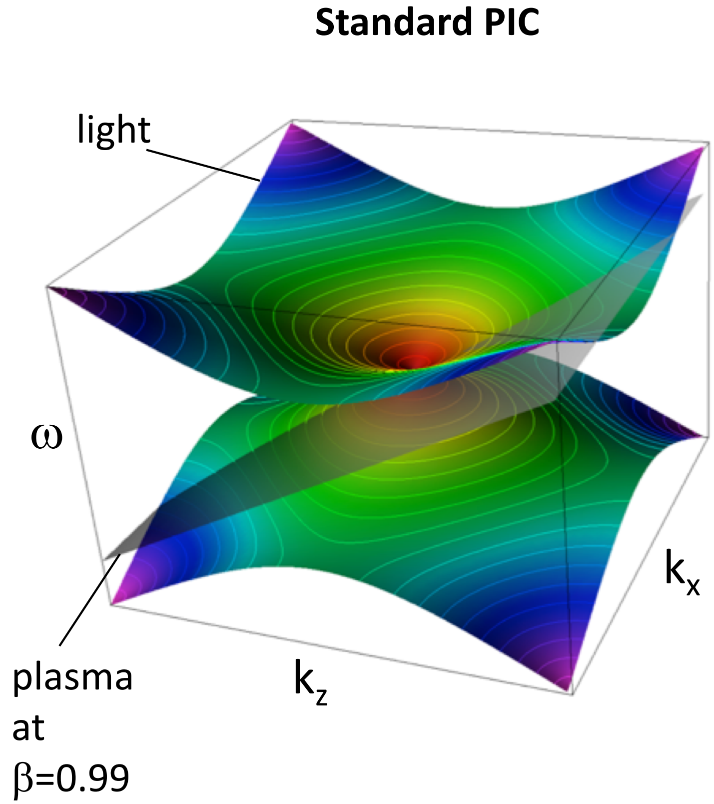
B. B. Godfrey, "Numerical Cherenkov instabilities in electromagnetic particle codes", *J. Comput. Phys.* **15** (1974)

Numerical dispersion leads to crossing of EM field and plasma modes -> instability.

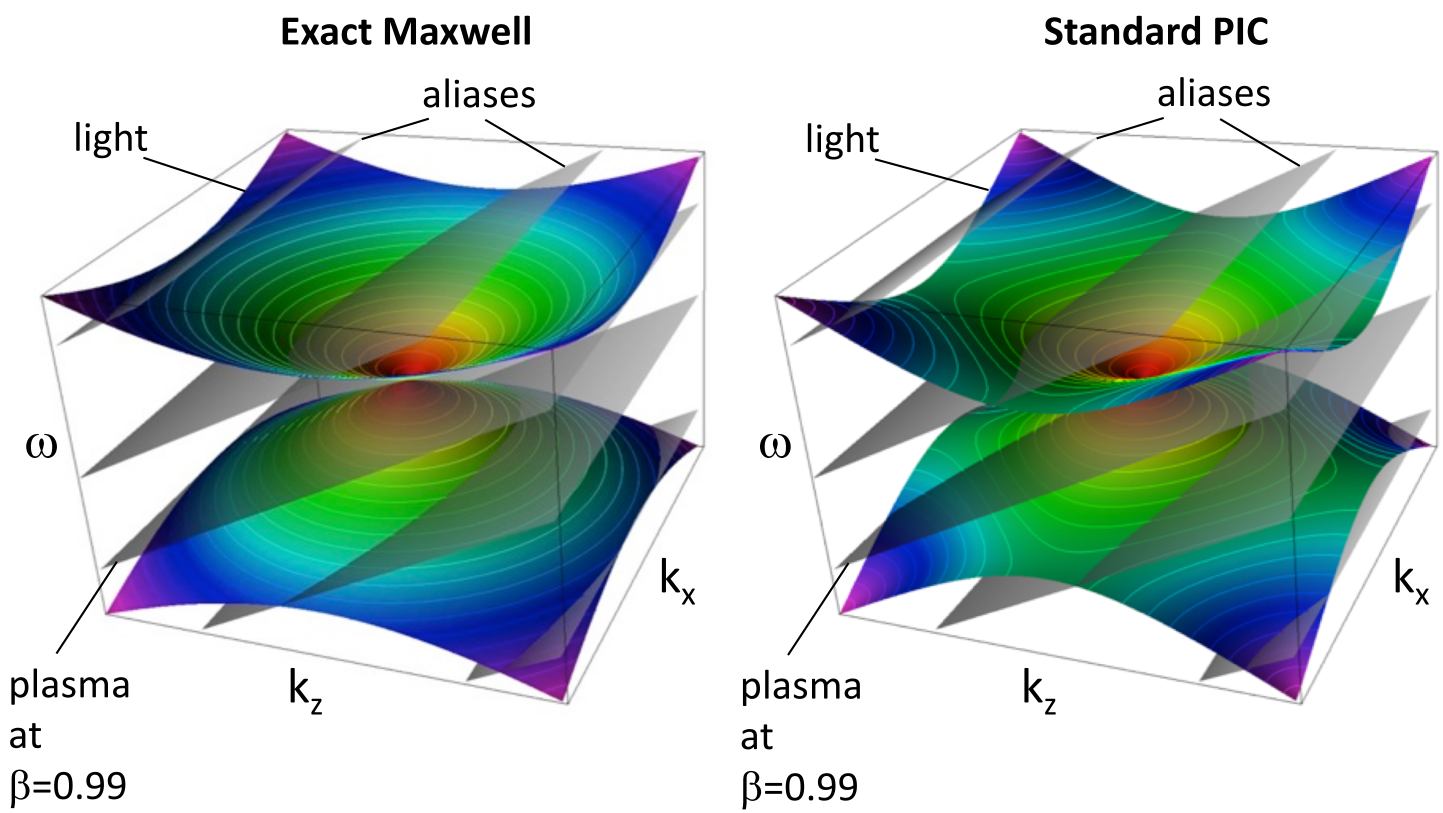


Space/time discretization aliases ➔ more crossings in 2/3-D



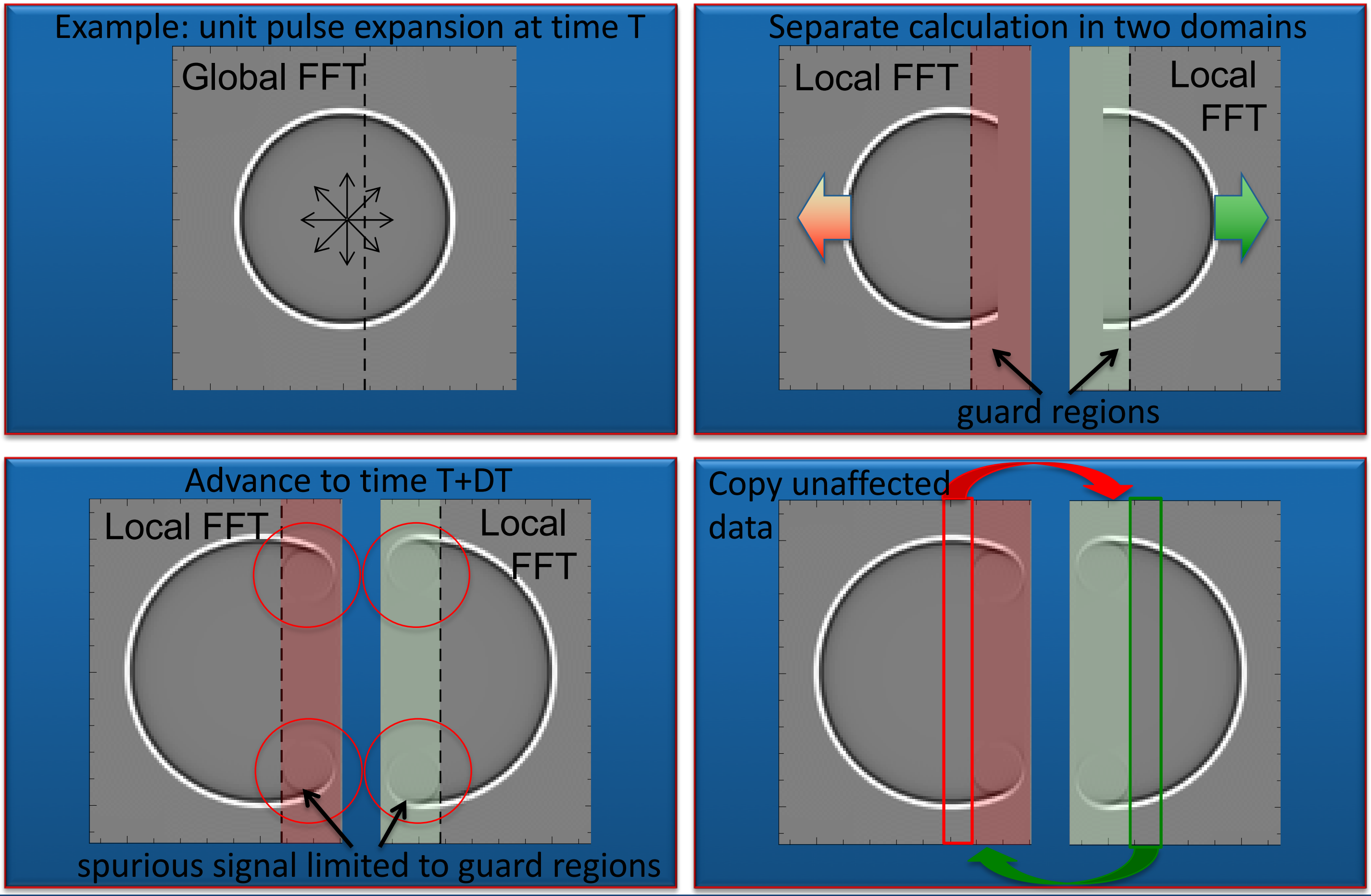


Space/time discretization aliases ➔ more crossings in 2/3-D



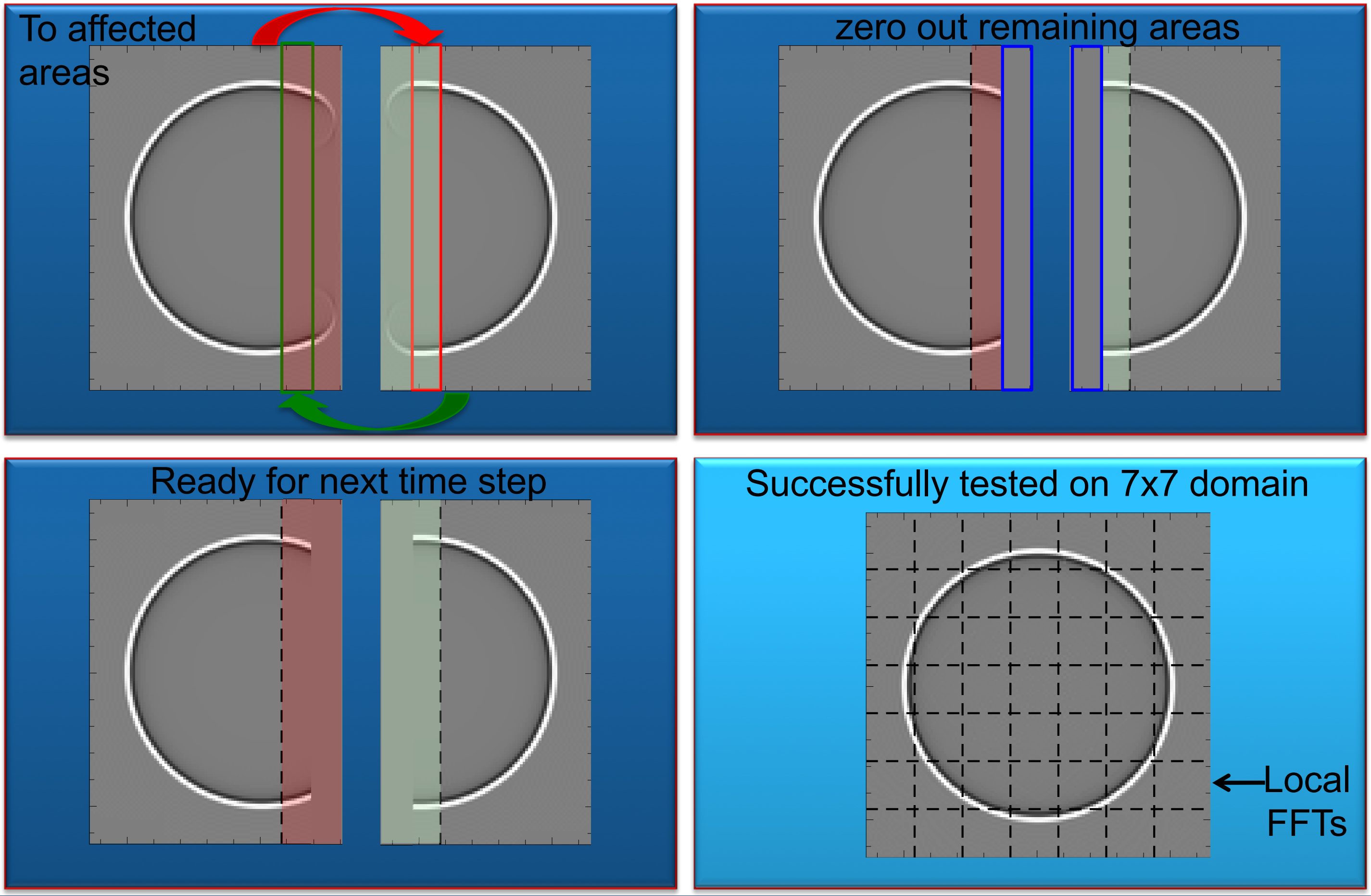
Need to consider at least first aliases m_x={-3…+3} to study stability.

Local FFT example on single pulse – part 1



*J.-L. Vay, I. Haber, B. Godfrey, *J. Comput. Phys.* **243**, 260-268 (2013)

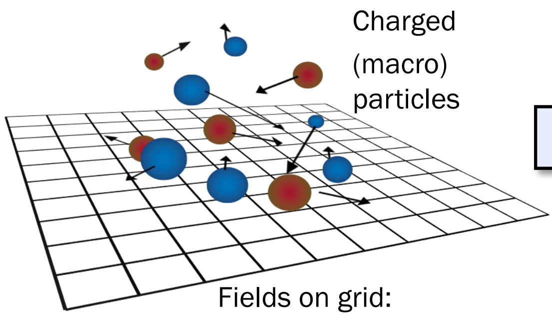
Local FFT example on single pulse – part 2

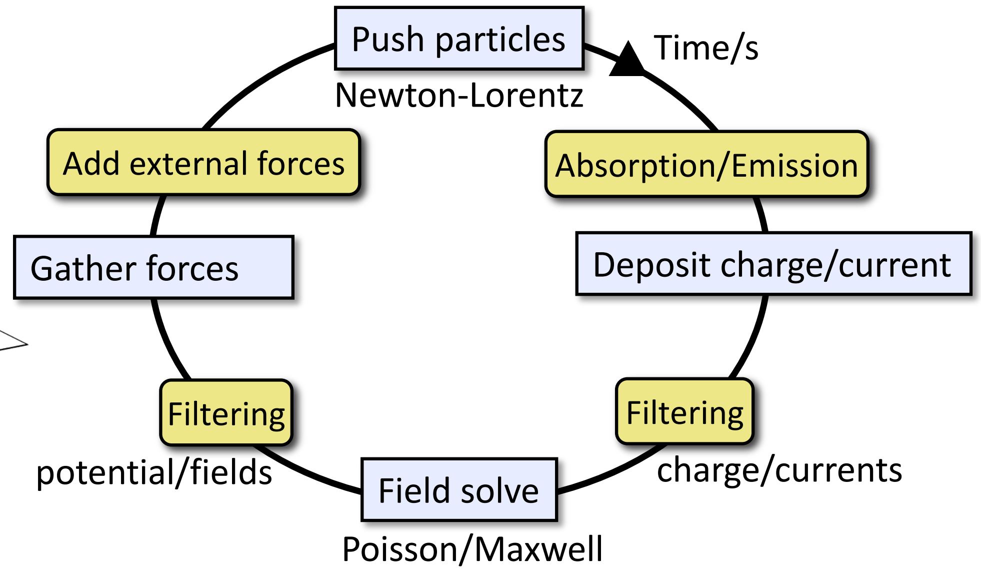


Truncation errors may lead to instabilities that needed to be studied.

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.