SciDAC Frameworks and Solvers for Multi-physics Beam Dynamics Simulations

James Amundson
Fermilab

with
Panagiotis Spentzouris *Fermilab*,
J. Qiang, R. Ryne *LBNL*,
D. Dechow *Tech-X*

SciDAC2: COMPASS Collaboration

Community Petascale Project for Accelerator Science and Simulation

A Proposal Submitted to the DOE Office of Science

Participating Institutions and Principal Investigators

Argonne National Laboratory:	Michael Borland[*], Lois Curfman McInnes[*‡]
Brookhaven National Laboratory:	Wolfram Fischer[*]
Fermi National Accelerator Laboratory:	James Amundson, Panagiotis Spentzouris[*‡]
Lawrence Berkeley National Laboratory:	William Fawley[*], Esmond Ng[*‡], Ji Qiang[*], Robert Ryne[*‡]
Los Alamos National Laboratory:	Peter Walstrom[*]
Oak Ridge National Laboratory:	Richard Barrett[*]
Stanford Linear Accelerator Center:	Lie-Quan Lee, Cho Ng[*‡]
Stony Brook University:	James Glimm[*]
Tech-X Corporation:	David L. Bruhwiler, John R. Cary[*‡]
Thomas Jefferson National Accelerator Facility:	Rui Li[*], Haipeng Wang
University of California, Davis:	Kwan-Liu Ma[*]
University of California, Los Angeles:	Viktor Decyk, Warren Mori[*‡], Sven Reiche[*]
University of Maryland:	Tom Antonsen[*]
University of Southern California:	Tom Katsouleas[*]

8 laboratories
5 universities
1 company

Mission

- To provide high-fidelity multi-physics simulations to the accelerator physics community
  - Including space-charge, electron-cloud, wakefields, beam-beam, nonlinear optics, etc.
- In order to do this, we will
  - Avoid “re-inventing the wheel”
    - worse, avoid “re-inventing the sledge”
  - Use state-of-the art tools and techniques
    - Especially those provided by the Computer Science/Applied Math community

Parallel computers, large and small

In order to enable leading-edge simulations, we focus on large-scale parallel computations. However, we must also consider the entire spectrum of current parallel computing

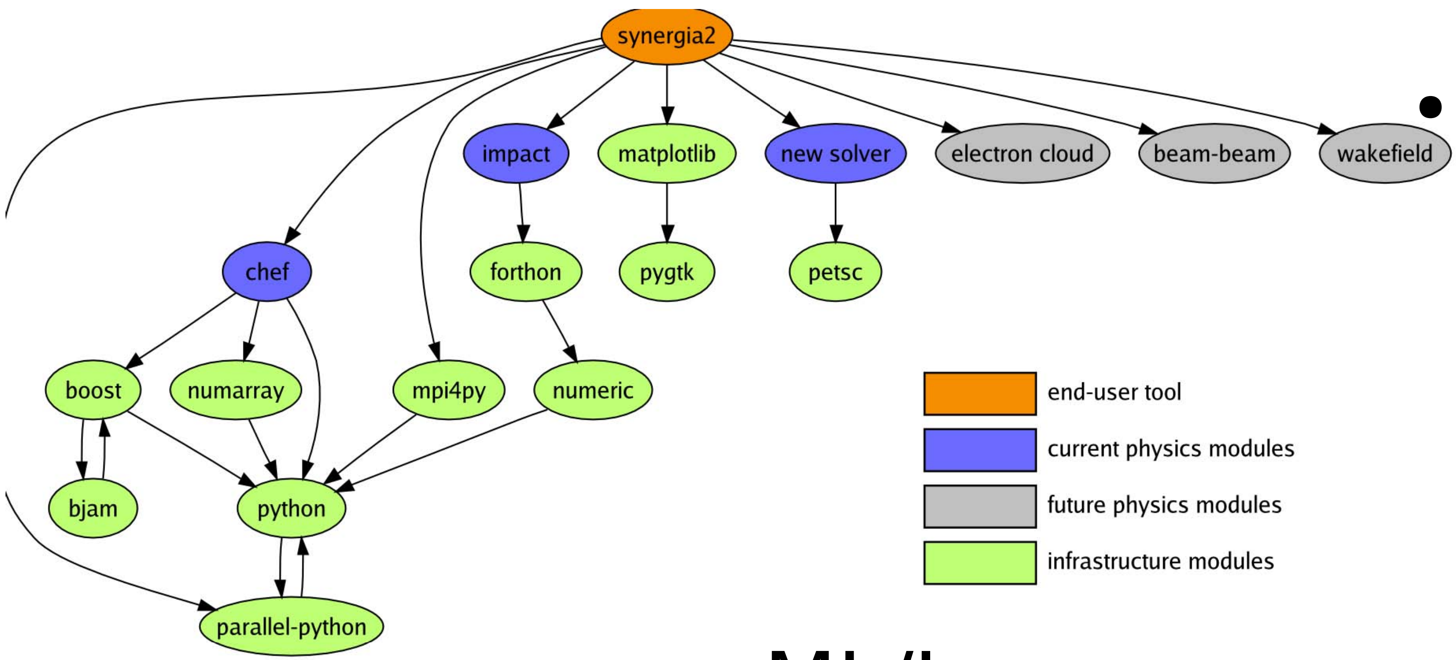
Desktop machines (even laptops) are increasingly including multiple processors/cores

Small to large Linux clusters

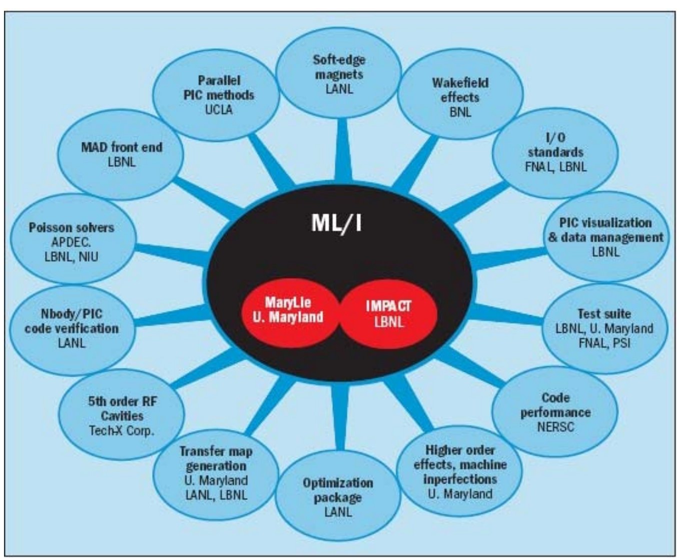
with commodity (gigabit) or high-end (Myrinet, Infiniband) interconnects

Massively parallel supercomputers with specialized networking hardware and topologies

Frameworks from SciDAC1



- Synergia2
  - CHEF
    - MAD input
    - arbitrary-order maps
    - etc.
  - IMPACT
    - 3D space-charge
  - FFTW solver
    - 3D space-charge

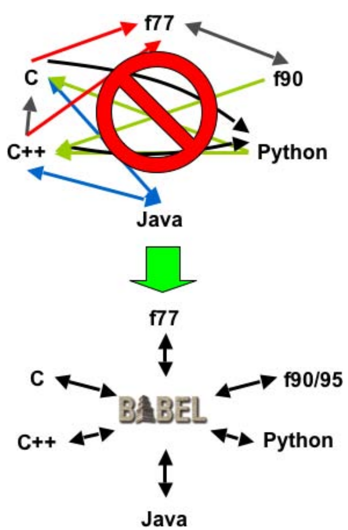


- ML/I
  - MarieLie
    - 5th-order maps
    - etc.
  - IMPACT
    - 3D space-charge

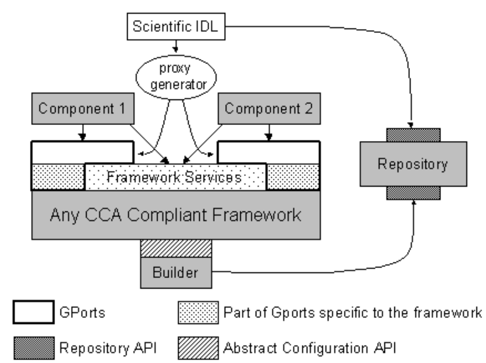
SciDAC2: Component-based architecture

While Synergia2 and ML/I have demonstrated that it is possible to integrate existing beams physics modules, it was (and is) hard work.

The Common Components Architecture (CCA) Forum aims to make the process of creating true component-based scientific software easier

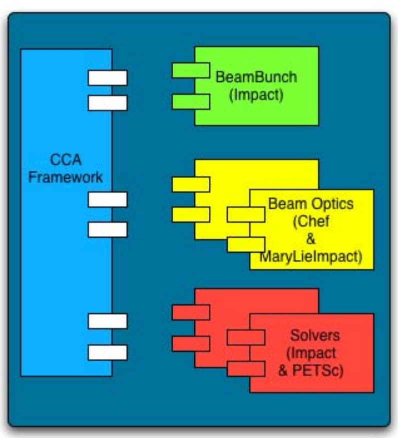


Babel: tool for language-independent module creation



CCA provides component infrastructure

The goal is true components providing that can be simply plugged together

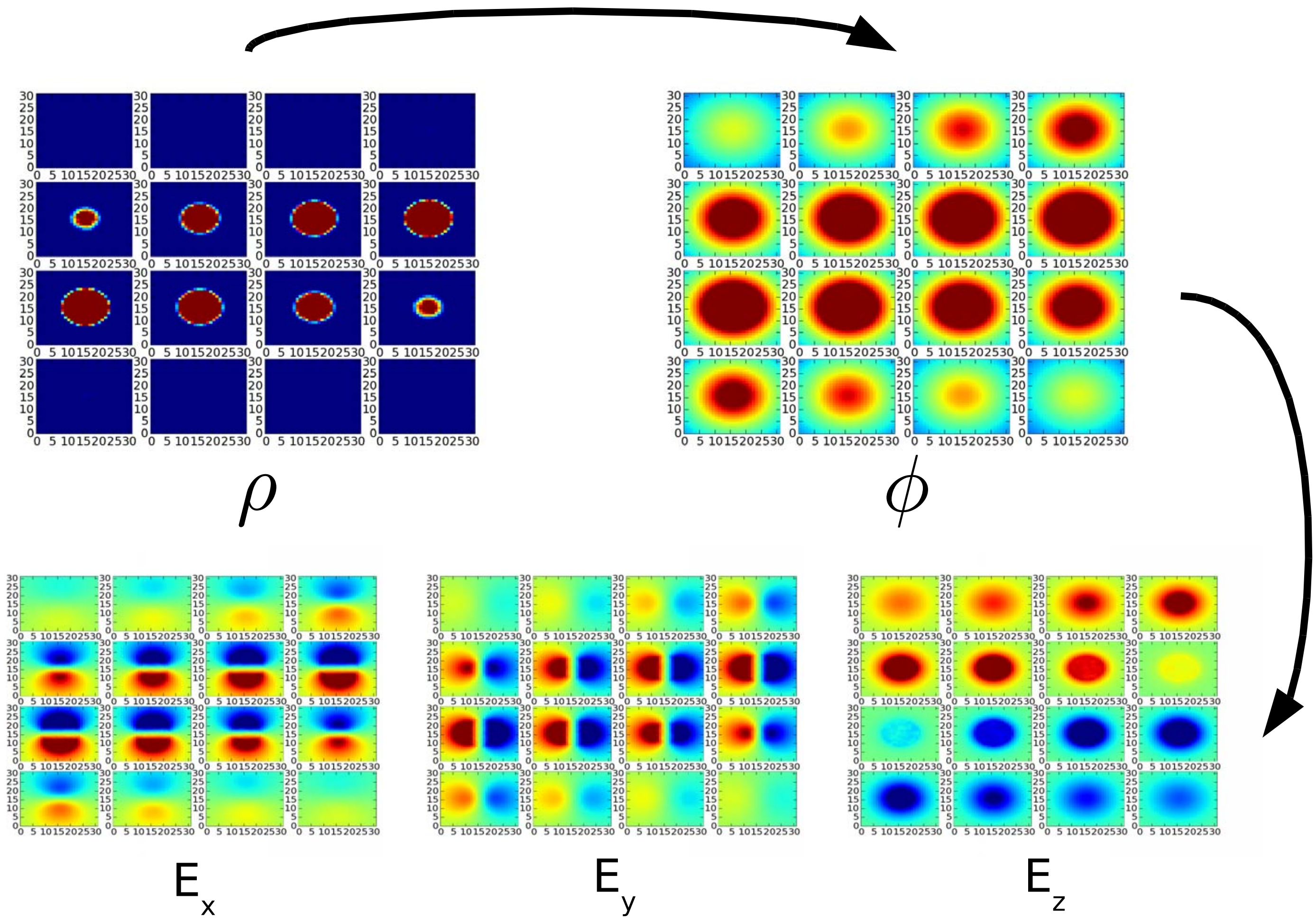


See: Doug Dechow, Poster THPAS019:
A Beam Dynamics Application Based on the Common Component Architecture

Poisson solvers

Prototypical element of collective effect simulation:
Poisson solver

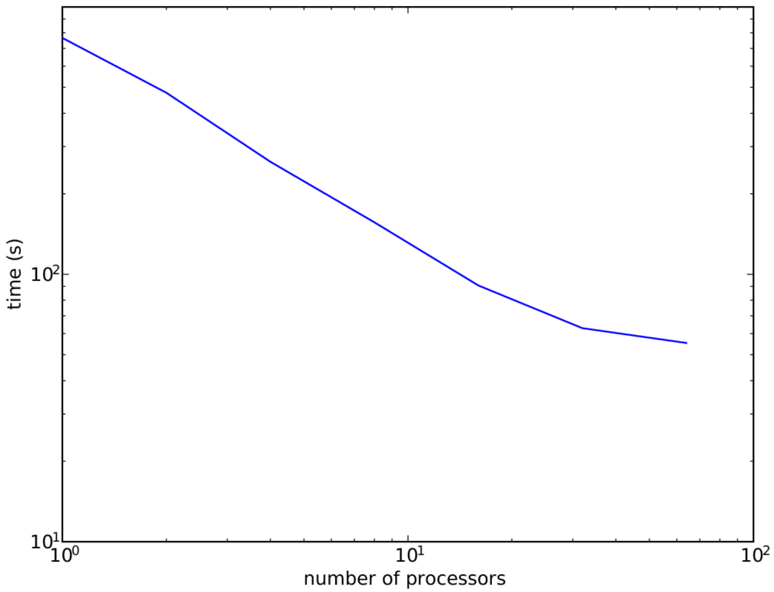
Use space charge as an example application



Parallel FFT-based Poisson solvers

- IMPACT
  - F90
  - State-of-the-art for a long time
    - established, benchmarked
- New FFTW-based solver
  - C++
  - Uses FFTW3.2 (currently in alpha release
    - First FFTW3 with MPI parallelism
    - “Fastest Fourier Transform in the West”
    - Actively maintained
      - widely used, tested
  - Can also use FFTW2
    - previous stable *parallel* FFTW

FFTW-based solver



Performance for single revolution of FNAL Booster:

32x32x256 grid
~1M particles
96 space-charge kicks

48 dual 2.4 GHz Linux boxes
Myrinet interconnects

- Fully integrated in Synergia2
- ~25% faster than IMPACT solver on 1 processor
- Decent scalability
  - single processor performance is the enemy of scalability(!!)
- Can use arbitrary grid shapes – not limited to powers of two!

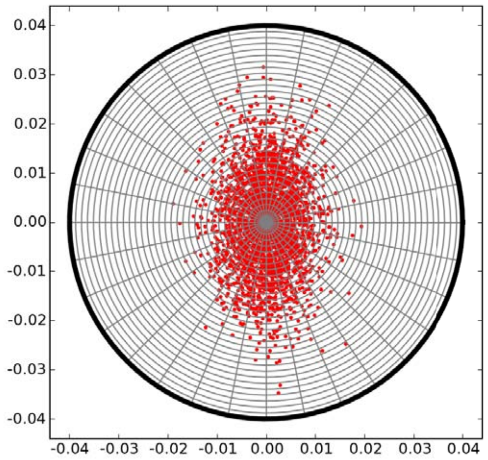
Finite Difference Poisson Solver

- FFT-based solvers are fast, but have limitations
  - uniform grids
  - simple boundary conditions
- Finite difference solvers reduce to linear algebra
  - Vast literature on parallel linear algebra
    - and libraries!
    - Multigrid method is *theoretically* optimal
  - Much greater flexibility in grids, BCs

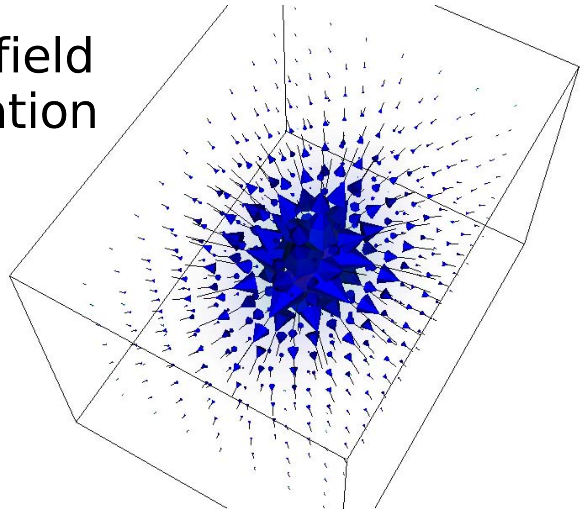
PETSc-based multigrid solver

- **P**ETSc, the **P**ortable, **E**xtensible **T**oolkit for **S**cientific **c**omputation
  - state-of-the-art for parallel linear algebra (and more)
  - uniform interface to a wide variety of solvers
  - collaborators
- Our first PETSc-based solver implements a dual-density grid
  - efficient compromise between inefficient uniform grids and Adaptive Mesh Refinement (AMR) which would add substantial overhead

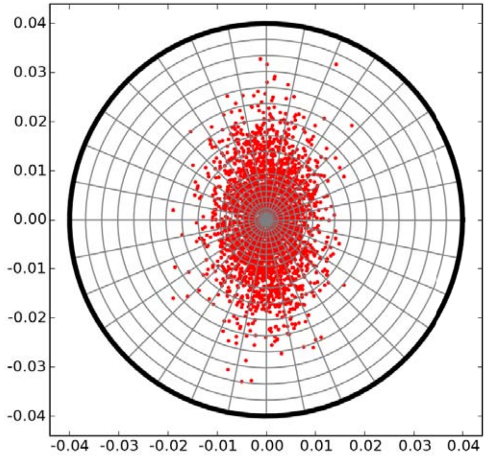
PETSc-based multigrid solver



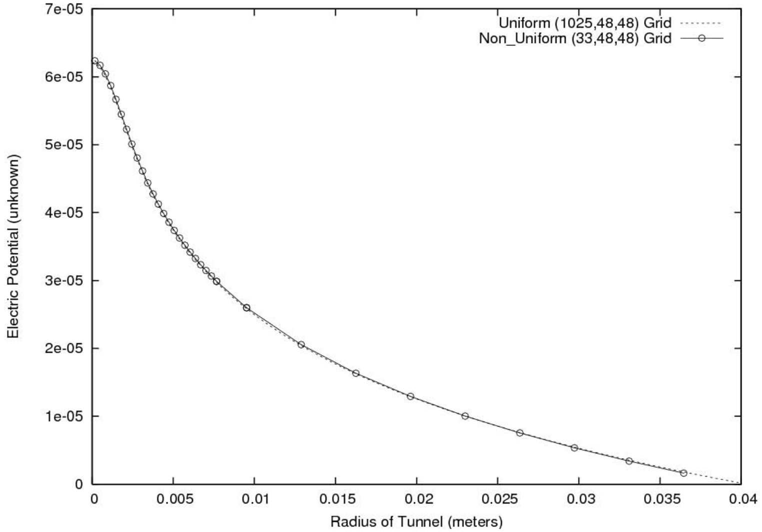
uniform grid wastes many cells on empty space



electric field visualization

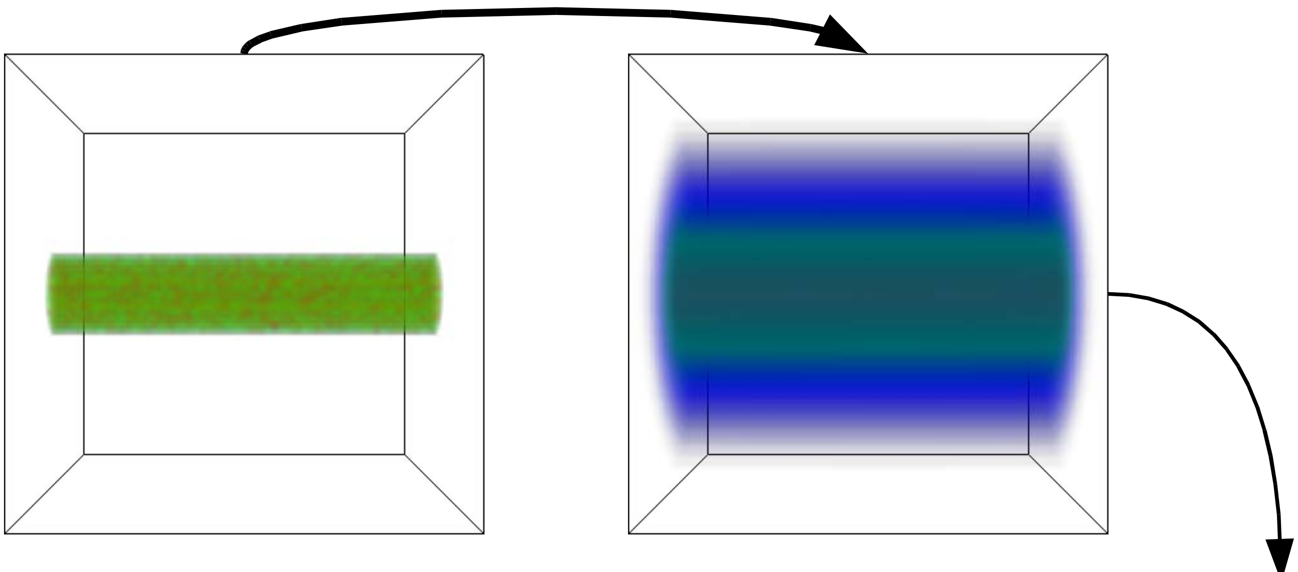


dual-level grid focuses cells where field is changing most rapidly

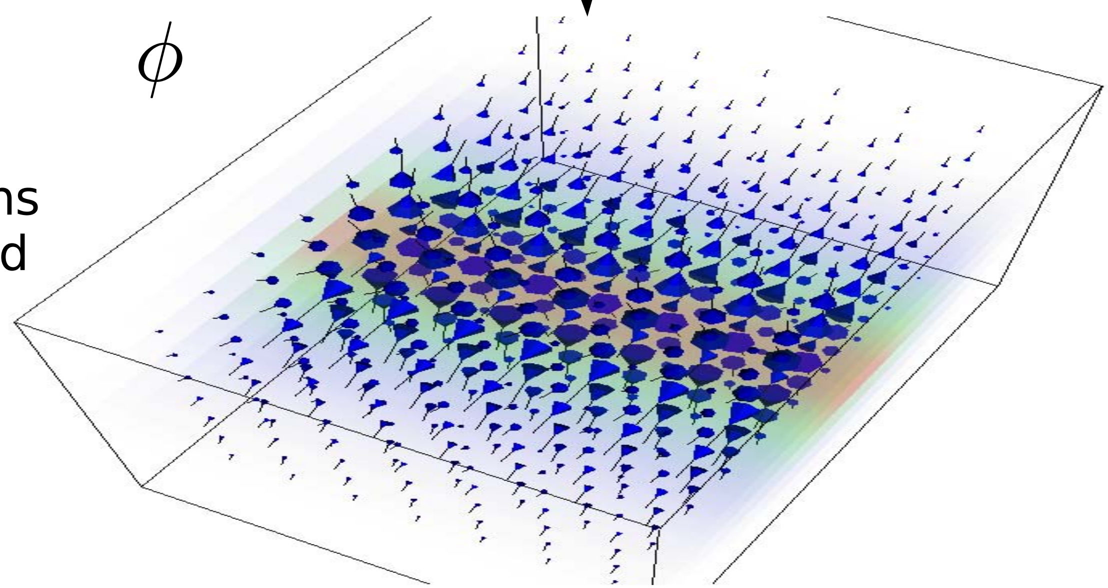


similar accuracy with far less computational effort

PETSc-based multigrid solver



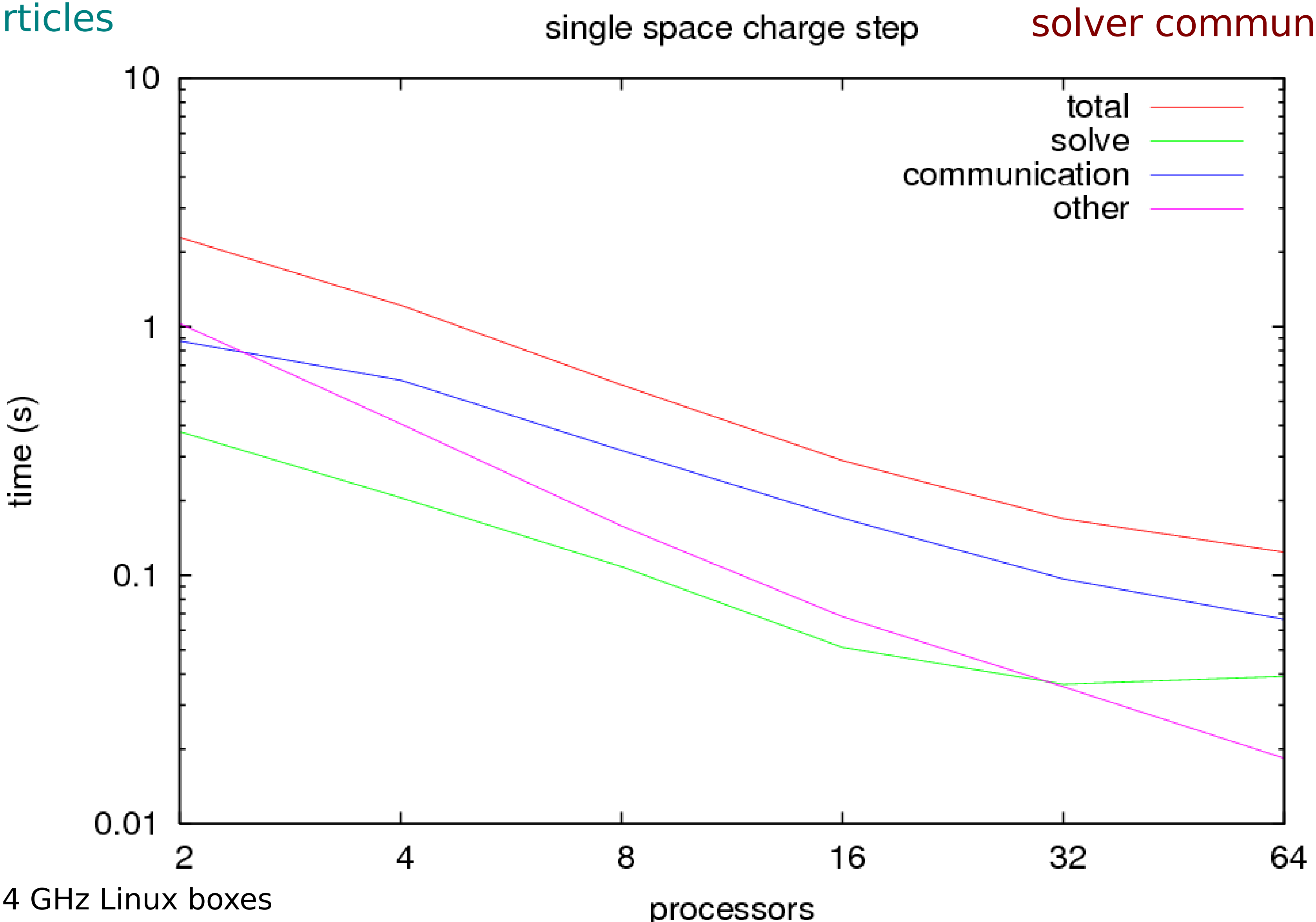
Note how statistical fluctuations in charge density are smoothed away by Poisson operator



Solvers aren't the end of the story

IMPACT solver
32x32x256 grid
1M particles

"communication"
is above and beyond
solver communication



48 dual 2.4 GHz Linux boxes
Myrinet interconnects

Parallel decomposition schemes

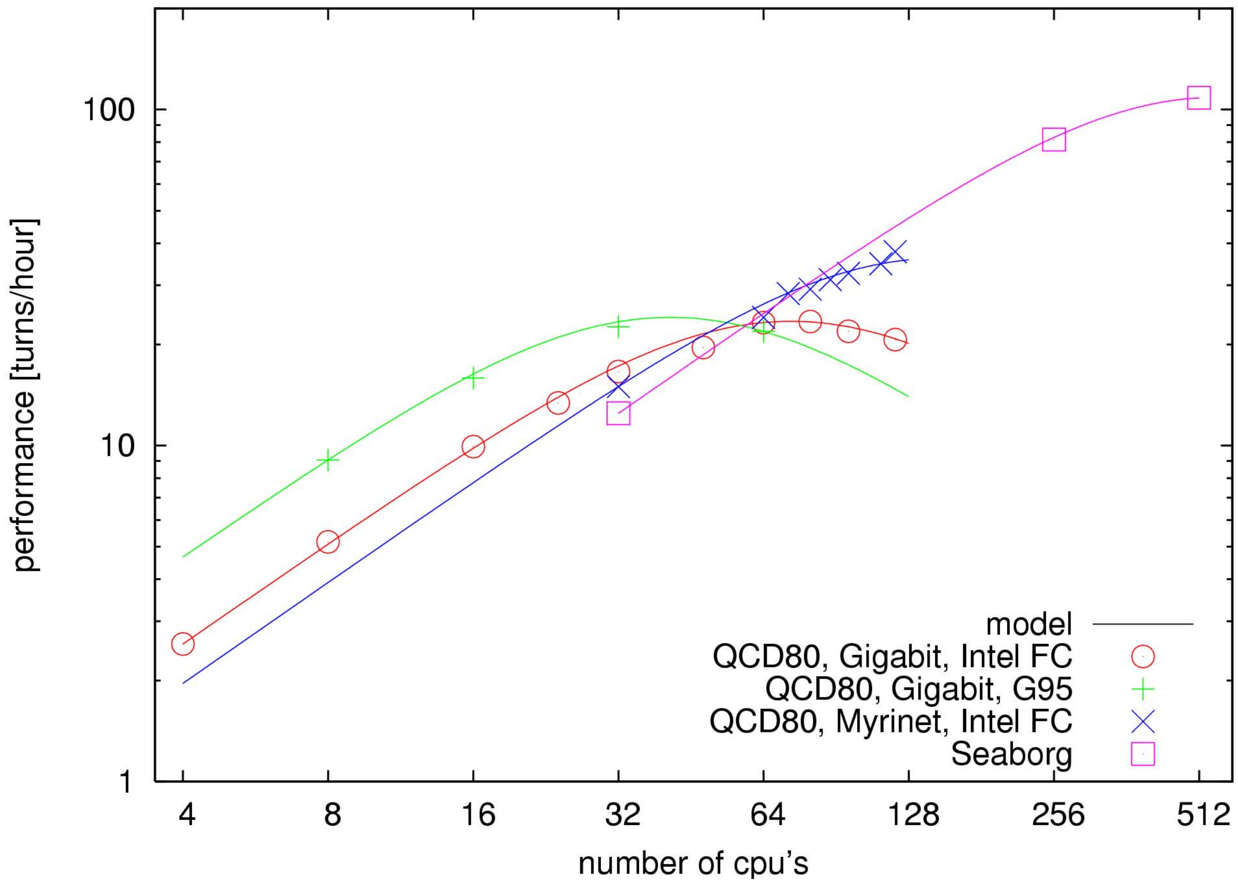
- Particle/field decomposition
  - spatial decomposition of fields
  - spatial decomposition of particles
    - ***performance depends on physics***
  - used in IMPACT
- Field decomposition
  - spatial decomposition of fields
  - particles uniformly, randomly distributed
    - ***performance independent of physics***
  - currently used with new FFTW-based solver

Particle/Field Decomposition

- Communication pattern consists of many small point-to-point communications
- Benchmarking verifies that the performance is limited by latency

fit to $t = aN + b/N$

find $a \propto$ latency

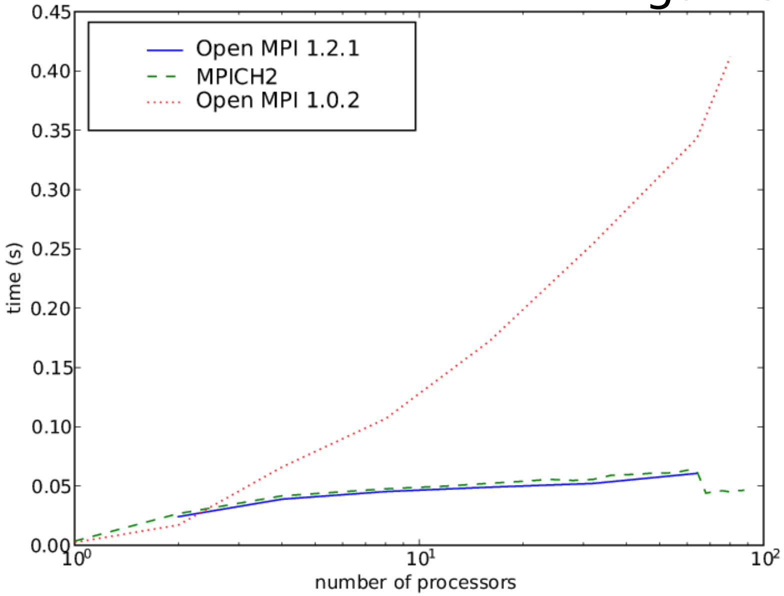


Particle decomposition

- No particle movement necessary
  - Charge density requires gather
  - Electric fields must be scattered
- Collective communications on few large messages
  - less sensitive to latency
    - better on commodity networks
  - takes advantage of optimized MPI collectives
    - particularly on supercomputing platforms such as BlueGene

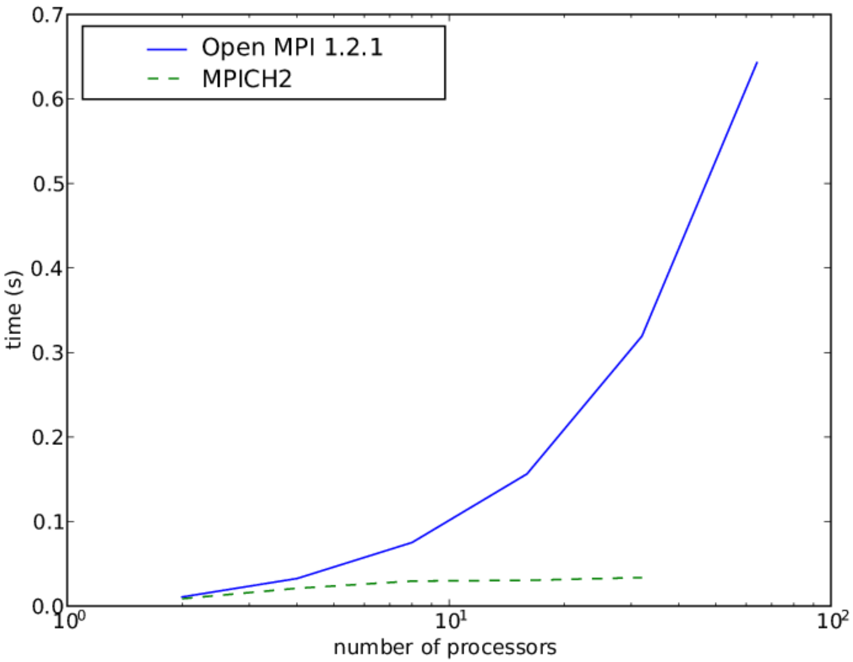
MPI collectives subtleties

gather



- Open MPI 1.0
  - naive collectives
- Open MPI 1.2
  - improved collectives
- MPICH2
  - less convenient, etc.

scatter



- Open MPI 1.2
  - not optimized on all collectives

48 dual 2.4 GHz Linux boxes
Myrinet interconnects

Conclusions

- COMPASS collaboration will build efficient, parallel platforms to perform next-generation multi-physics simulations
  - Build upon successes of SciDAC1 project
  - Component technology
  - Emphasize CS/Applied Math input (CCA, PETSc, FFTW)
    - state of the art
- Next-generation solver development underway
  - FFTW-, PETSc-based solvers implemented