Rensselaer

why not change the world?®

# Parallel Coupler for Multimodel Simulations (PCMS): A new approach for independent model coupling of Tokamak plasma simulations

**Jacob Merson**, Cameron Smith, Mark Shephard

FASTMath Seminar | 8/16/2023

# Parallel Coupler for Multimodel Simulations

**Goal:** make it easier to couple established fusion applications with minimal code modifications without sacrificing scalability, efficiency, or stability on exascale supercomputers.

**Goal:** Abstractions that make it easy for application specialists to focus on the unique parts of coupling. E.g., specific fields that are stored, transformations needed to “standard” format.

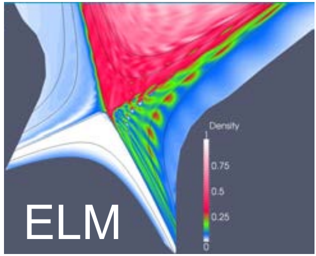
**Goal:** Support fusion system Whole Device Modeling (WDM) and Full Plant Modeling.

# Plan

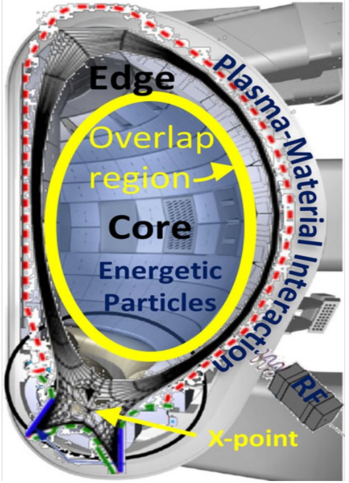
- Motivation for independent application coupling of fusion simulations
- Challenges and requirements
- Coupling approach
- Baseline core-edge plasma coupling
- Dynamic Coupling with Benesh, PCMS, and Adisos2
- Conclusions

# Role of Computing in Tokamak Design

- Answer fundamental questions about plasma physics
- Predict and avoid situations that render reactor non-functional
    - Plasma instabilities
    - Degradation of plasma facing components
    - Significant incursion of impurities

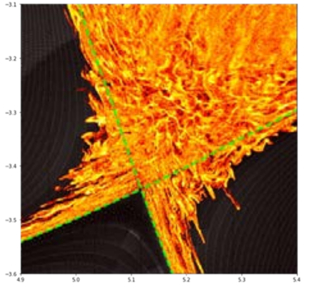



https://www.iter.org/newsline/259/1509



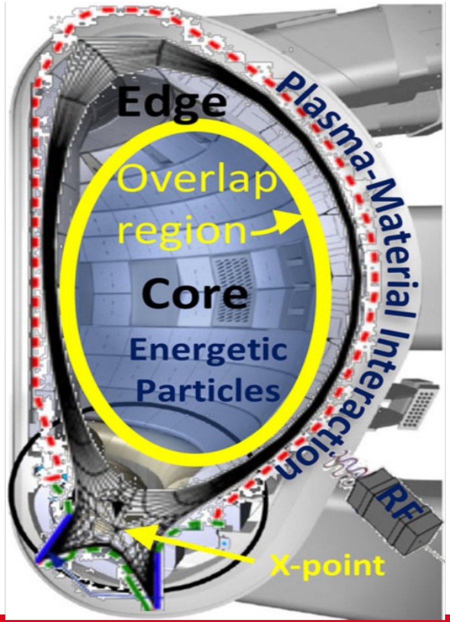

**Range of Computational Methods Necessary**

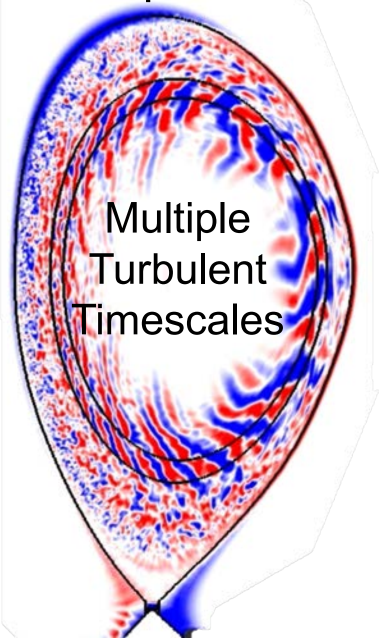
- Plasma facing materials:
    - Molecular dynamics (material evolution)
    - Finite element (heat transfer, structural design)
    - Kinetic Monte Carlo (wall interactions)
- Plasmas (solve Boltzmann and Maxwell's equations):
    - Particle-in-cell: 5D gyrokinetic, etc.
    - Continuum: often magnetohydrodynamics, or 5D gyrokinetic



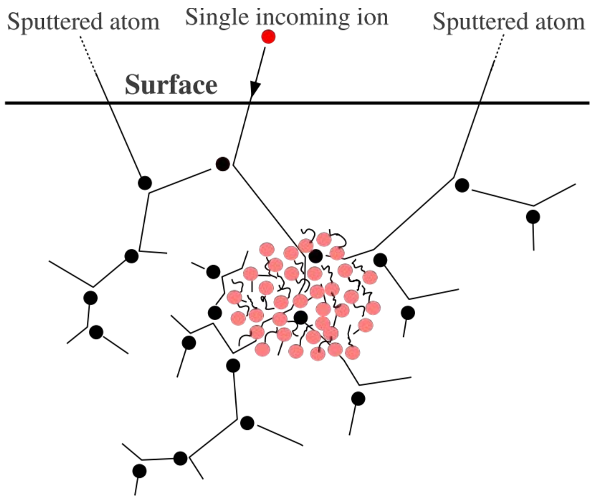
# Exascale Computing is Essential

- Individually modeling each reactor region requires exascale computations.
- Answering fundamental design questions requires coupling these models across spatial-temporal scales, spatial domains, and physical models.



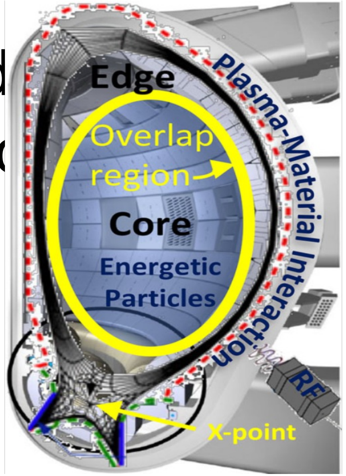

Kim et. al., 2018




Norlund et. al., 2014

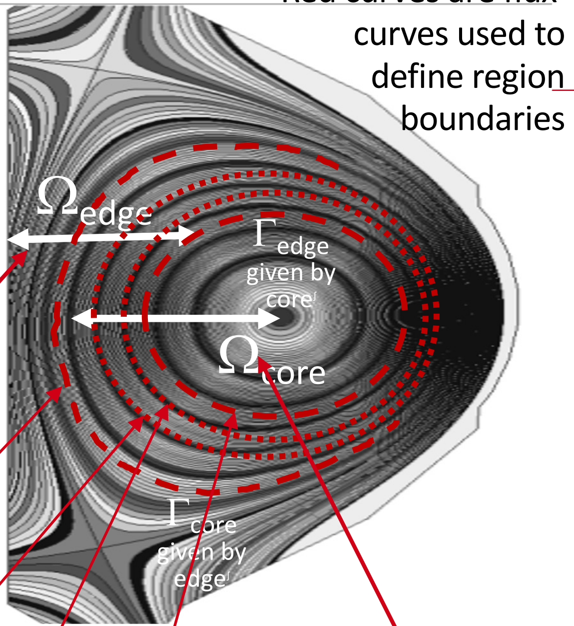
# A First Step: Plasma Coupling

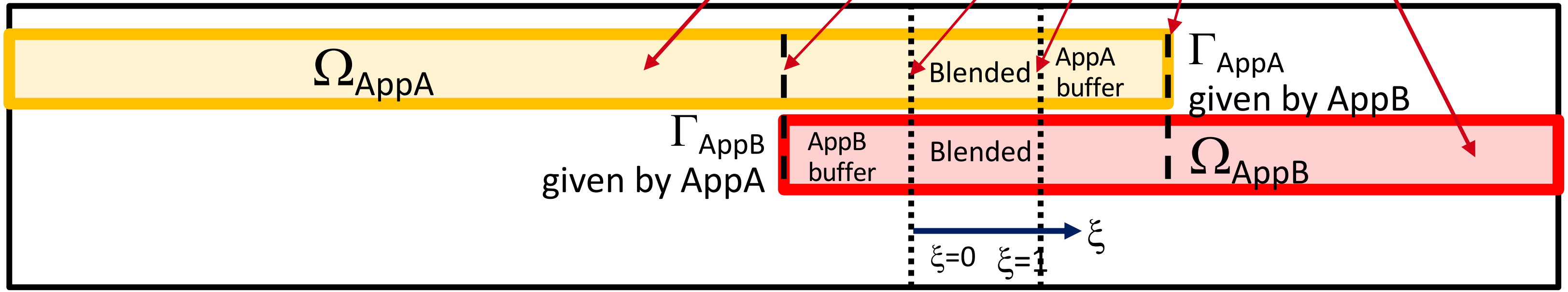
- Want to take advantage of simplified physics models in reactor core and couple to high-fidelity edge codes.
- Existing high-fidelity codes need to be coupled
- This is a concurrent multiphysics problem.
- Eventually support whole device modeling (includ scrape off layer, impurity transport, neutronics, etc



# Concurrent Coupling

- Each application solves its model(s) over a portion of the domain.
- The domains overlap: The overlap can include three subregions
  - The blended region in which the fields are coupled based on a field blending strategy
  - A buffer region for Application A (edge) in which the "right" end boundary conditions are determined by Application B (core) and/or source terms added
  - A buffer region for Application B (core) in which the "left" end boundary conditions are determined by Application A (edge) and/or source terms added



# A complicated problem

- Multitude of field following and radial coordinate systems.
- Field data stored in application dependent combinations of real and Fourier space.
- Both structured and unstructured meshes.
- Field data distributed with varying partitioning schemes and distributed data structures.
- Must run on exascale supercomputers.
- Need to relate fields across scales and through transformations

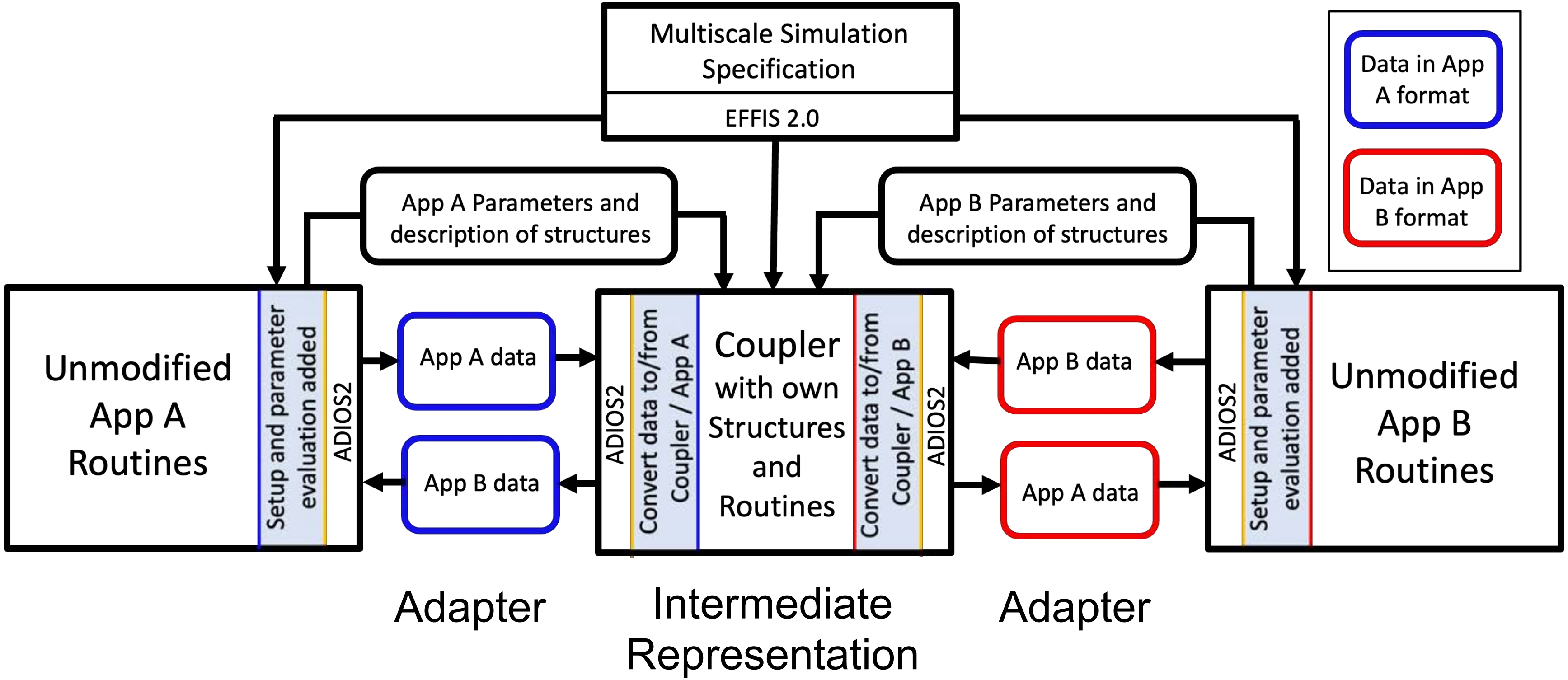
# Possible Paths to Support Coupling

- Ad-hoc coupling (currently under development for some application pairs e.g., GENE and XGC)
  - Not practical for testing all desired pairs of existing core/edge simulation codes.
  - Does not scale beyond core/edge coupling.
- Use existing coupling and mesh transfer libraries (e.g., preCICE, DTK, MOOSE, etc.)
  - Current frameworks cannot reasonably be extended to support all requirements.
  - Many frameworks require all codes to conform to their structure.

# Generalized Coupler Requirements

- Do not modify existing data structures or algorithms.
- Make effective use of exascale computing systems.
- Efficiently handle data and coordinate transformations.
- Perform accurate and efficient field transfer operations on structured and unstructured meshes.
- Handle parallel coordination and communication of distributed field data.

# Approach



Multiscale Simulation Specification
EFFIS 2.0
App A Parameters and description of structures
App B Parameters and description of structures
Data in App A format
Data in App B format
Unmodified App A Routines
Setup and parameter evaluation added
ADIOS2
App A data
App B data
ADIOS2
Convert data to/from Coupler / App A
Coupler with own Structures and Routines
Convert data to/from Coupler / App B
ADIOS2
App B data
App A data
ADIOS2
Setup and parameter evaluation added
Unmodified App B Routines
Adapter
Intermediate Representation
Adapter

# "Rendezvous" Algorithm to Control Coupler Domain Partition

Challenge

- The Core, Edge, and Coupler applications need to be able to control their domain partitioning to meet their specific needs.
- How can we efficiently transfer information to and from the Core/Edge while maintaining flexible control over the Coupler's partition?

Approach

- For coupling applications A and B this means letting them have their own partitions.
- The Rendezvous algorithm uses a third partition to coordinate data transfers between the applications.
- "Rendezvous" algorithm "enables scalable algorithms which are most useful when processors neither know which other processors to send data to, nor which other processors will be sending data to them" [1]
- Used by LAMMPS [1], in PUMI for loading multi-billion element meshes [2,3], and in DTK [4].

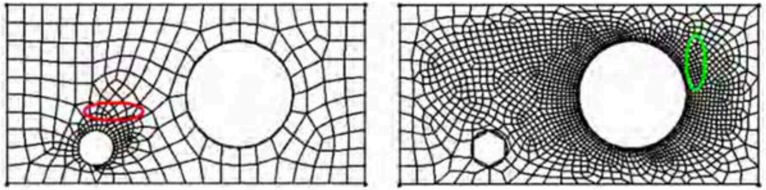


**Fig. 4.** Thermal (left) and stress (right) grids. The colored ovals are clumps of grid cells the same processor owns in the two grids.

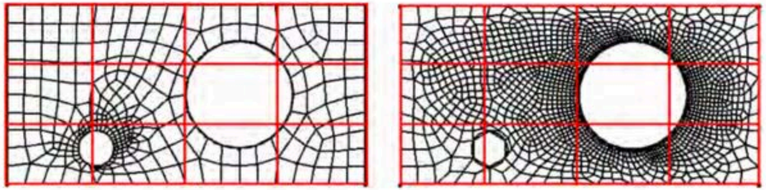


**Fig. 5.** Rendezvous decomposition overlaid on thermal and stress grids.

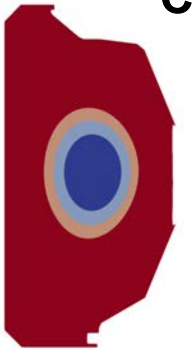

[1] "Rendezvous algorithms for large-scale modeling and simulation", Plimpton, et.al., JPDC, 2021.
[2] PUMI Github Issue
[3] "Optimizing CAD and Mesh Generation Workflow for SeisSol", Rettenberger, et.al., SC14, 2014.
[4] "The Data Transfer Kit: A geometric rendezvous-based tool for multiphysics data transfer", Slattery et.al., M&C 2013.

# Example Two-Application Partition Scheme (D3D)

## Edge Application

16 Part RIB Partition

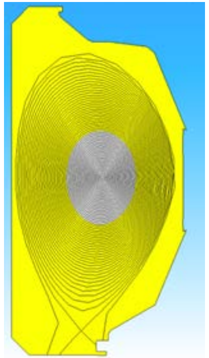
## Coupler



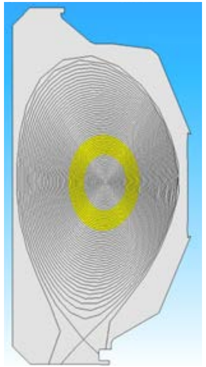
Feature Based Partition

## Core Application

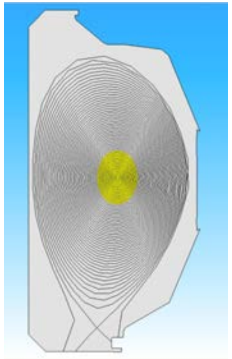
16 Part RIB Partition



Edge Domain



Overlap Domain

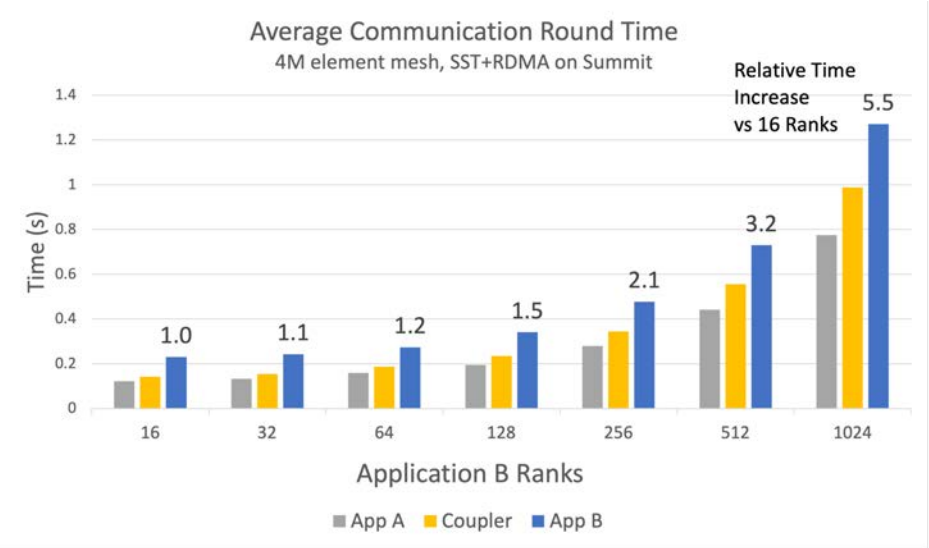


Core Domain

# Coupler Data Transfer Testing with Omega_h: Summit

Two client overlap test on 4 million element D3D mesh

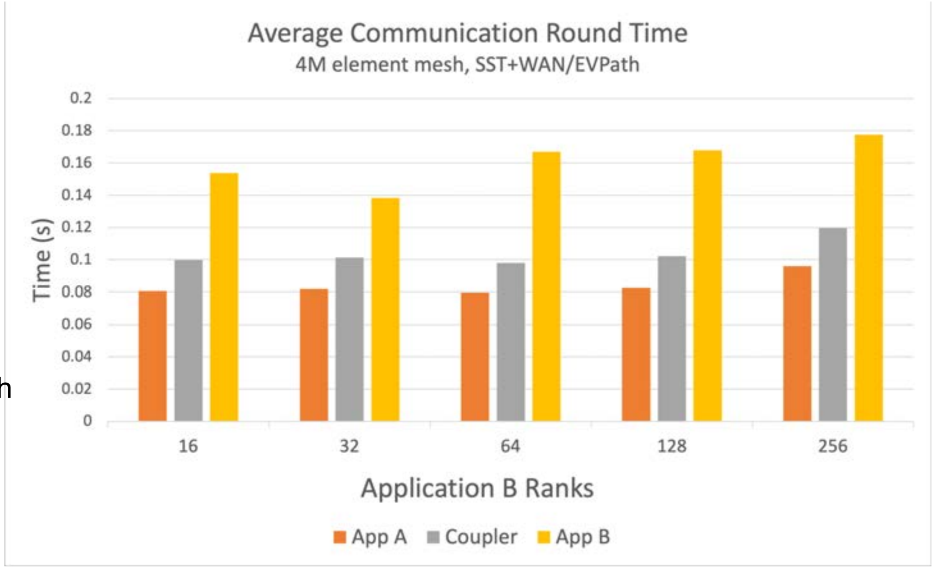
- Using high level coupler API
- Running on up to 32 Summit Nodes.
- Using ADIOS2 SST engine with **RDMA**.
- Application A: 16 processes, RIB partition
- Coupler: 16 processes, feature-based partition
- Application B: 16-1024 processes (32/node), RIB partition
- Communication Round in Overlap:
  - Applications perform a ‘forward’ send to the Coupler
  - Coupler performs ‘reverse’ send to both Applications
  - Received data is checked for correctness and attached to the mesh
- Meshes with 4M elements
- Time spent in high level coupler API send and receive, 10 communication rounds



# Coupler Data Transfer Testing with Omega_h: Crusher

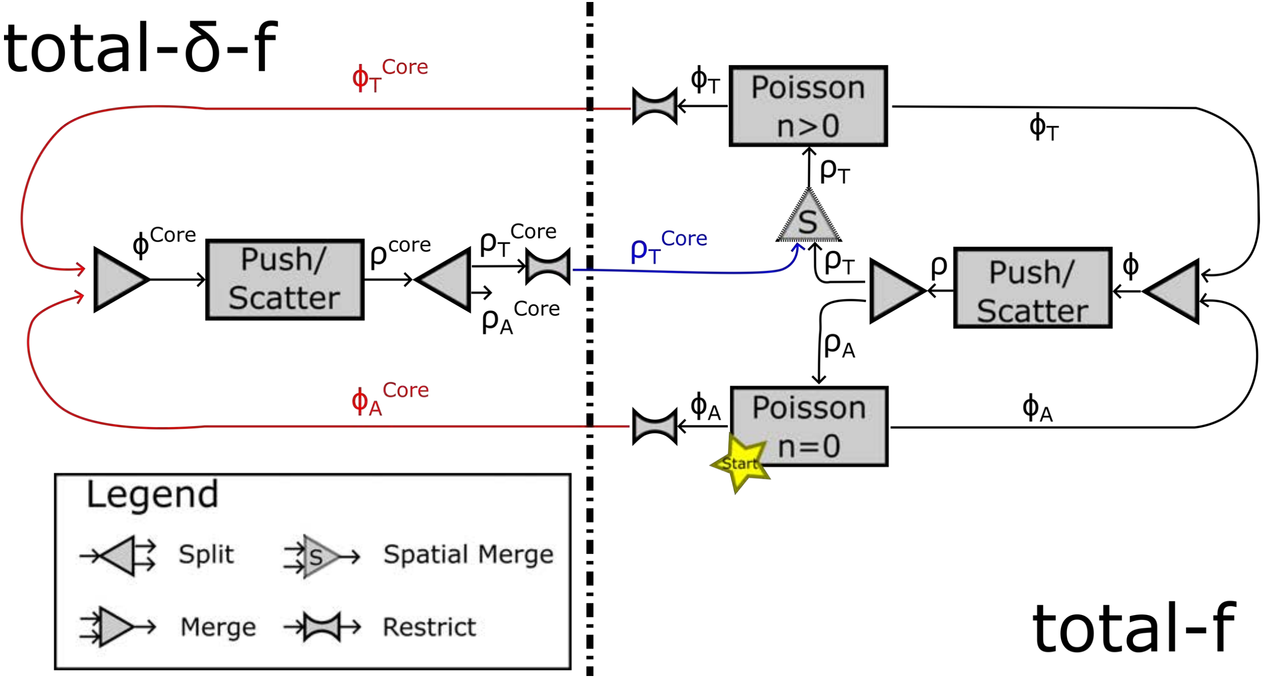
Two client overlap test on 4 million element D3D mesh

- Using high level coupler API
- Running on up to 6 Crusher Nodes.
- Using ADIOS2 SST engine with **WAN/EVPath**.
- Application A: 16 processes, RIB partition
- Coupler: 16 processes, feature-based partition
- Application B: 16-256 processes (64/node), RIB partition
- Communication Round in Overlap:
  - Applications perform a 'forward' send to the Coupler
  - Coupler performs 'reverse' send to both Applications
  - Received data is checked for correctness and attached to the mesh
- Meshes with 4M elements – less than a MB being sent/received per App B rank at 256 ranks
- Time spent in high level coupler API send and receive, 10 communication rounds



# Core Edge Plasma Coupling

- **Model**: Cyclone ITG
- **Mesh**: 590k elements in overlap
- **Equations:** electrostatic Vlasov with adiabatic electrons
- **Solution procedure:** gyrokinetic particle-in-cell
- **Location:** Perlmutter using ADIOS2 SST with RDMA

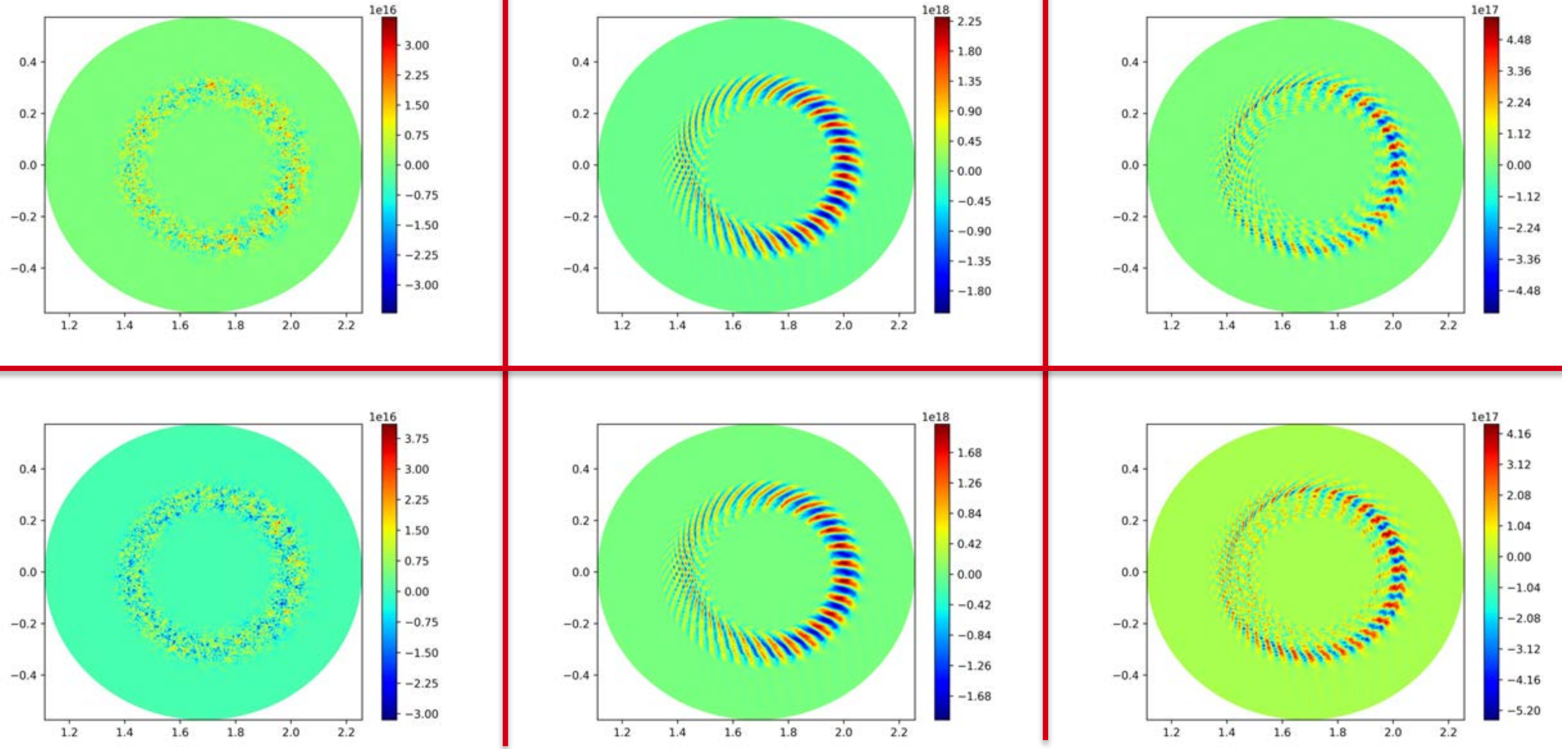


# Core Edge Coupling: Turbulent Ion Density

T=100 | T=600 | T=1000

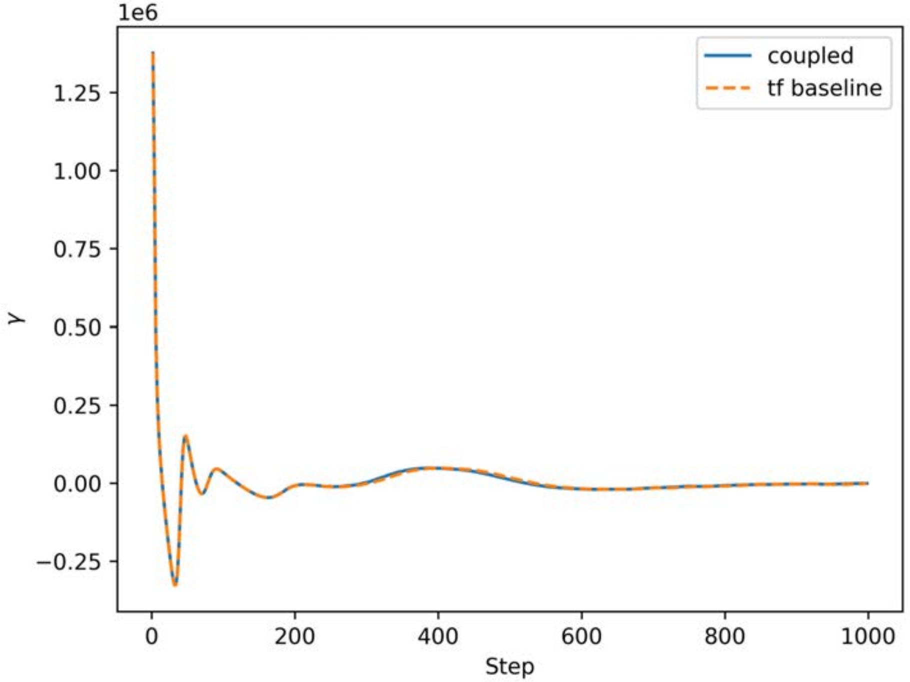
Baseline total-$\delta$f

Coupled



# Core Edge Coupling: Turbulence Growth Rate

- $\gamma$ provides quantitative measure of error and simulation saturation
- $\gamma = \dfrac{d \log |\Phi|_{l2}}{dt}$



# Dynamic Coupling with Benesh

A programming model for developing in-situ workflows

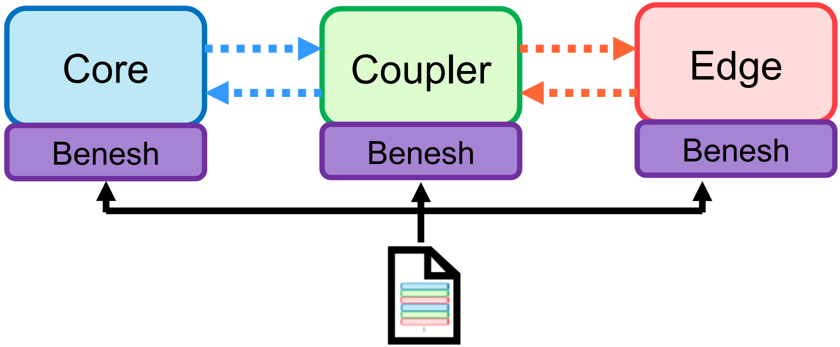
- Take existing codes and make them work together
- Abstractions aimed at supporting multiphysics use cases
- Hide complexities of data flow synchronization points and dependencies from developer

Key Components

- Programming-language hooks for a preparing an existing code for use in a Benesh workflow
- Workflow description language for specifying flexible interactions of workflow components
- Middleware for instantiating Benesh workflows

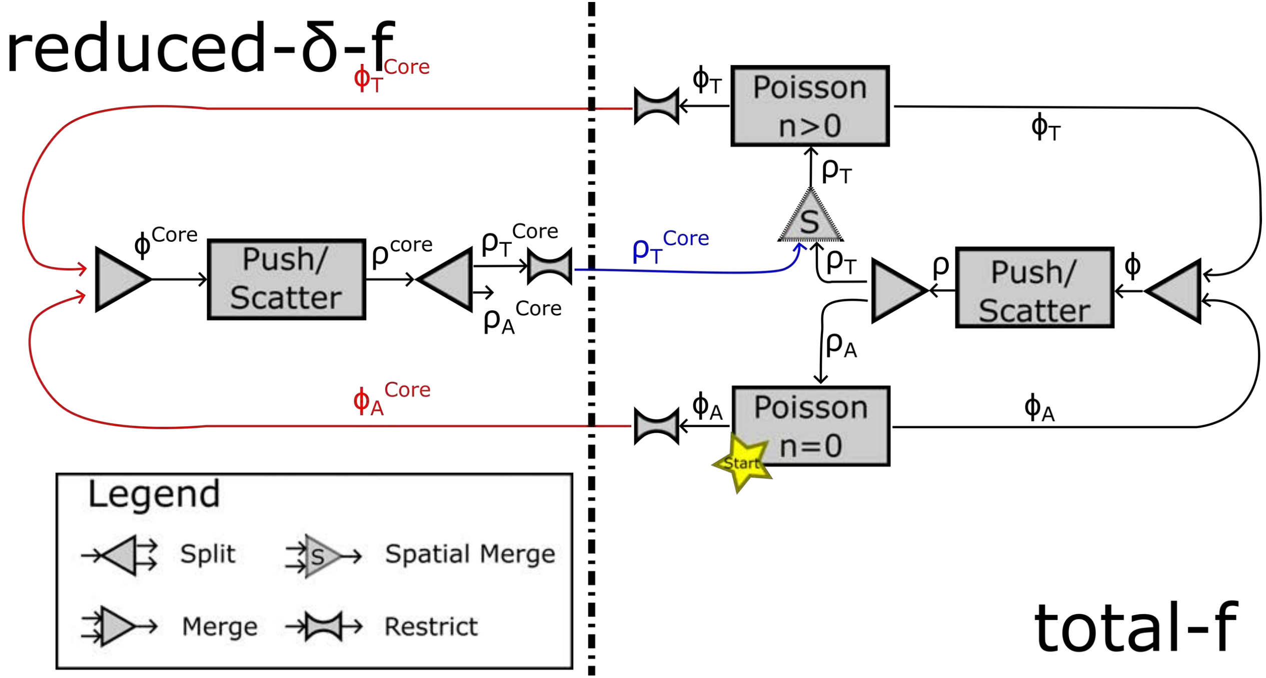
Declare common data domain across XGC, GEM, and coupler

- Common definition of high-level domain geometry
- Core and edge definition with implied boundary
- Done in runtime configuration (can be partially generated by EFFIS)



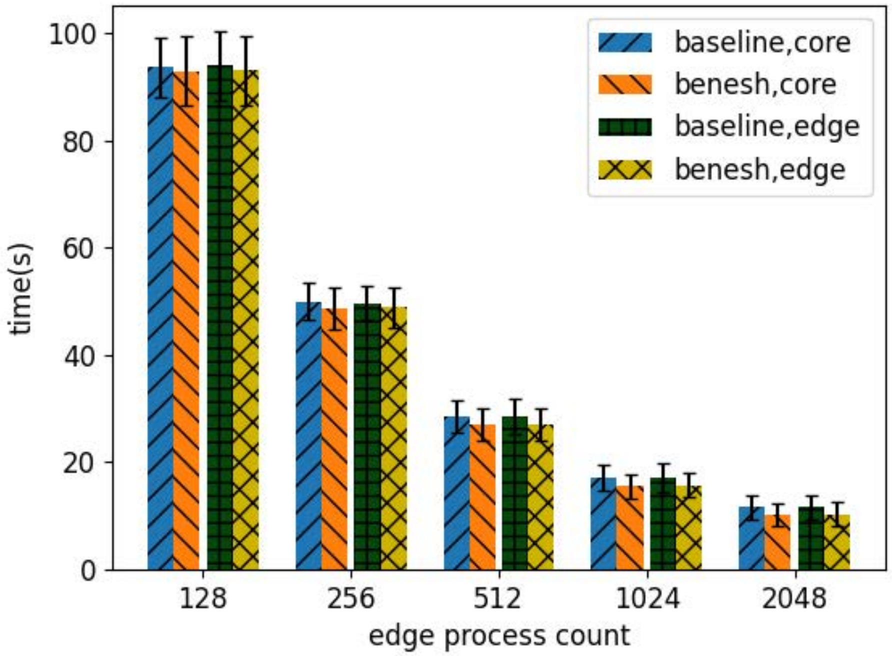
# Core Edge Plasma Coupling (version 2)

- **Model**: Cyclone ITG
- **Mesh**: 590k elements in overlap
- **Equations:** electrostatic Vlasov with adiabatic electrons
- **Solution procedure:** gyrokinetic particle-in-cell
- **Location:** Frontier using ADIOS2 SST with RDMA



# PCMS and Benesh on Frontier

- Ranks
  - Core:Edge:Coupler, 8:32:1
- Using ADIOS2 with RDMA
- Timing matches between baseline PCMS and PCMS+Benesh
- No GPU usage (yet)



# FES Parterships

PCMS will be used to provide code coupling tools for four FES partnerships including 3 with direct RAPIDS collaborations

- HifiStell: High-Fidelity Simulations for Stellarators
- Center for Advanced Simulation of RF - Plasma - Material Interactions
- Computational evaluation and design of actuators for Core- Edge Integration
- High-fidelity Digital Models for Fusion Pilot Plant Design

# Conclusions

- Coupling is hard. It requires interaction between application specialists, applied mathematicians, and computer scientists.
- PCMS provides a scalable framework that handles many complexities of coupling field-based data at scale.
- PCMS requires no modification to internal data structures.
- PCMS enabled quickly testing two different core/edge coupling schemes.

## Future Work

- Frontier GPU aware support (flang segfault)
- Conservative field transfer methods
- Improved support for asynchronous control and dynamic coupling
- Support for one-way coupling workflows
- Support for alternative time-stepping schemes.
- Collaborate with RAPIDS on a common data schema

# Funders and Collaborators

ECP EXASCALE COMPUTING PROJECT

PPPL PRINCETON PLASMA PHYSICS LABORATORY

SCI www.sci.utah.edu

SciDAC Scientific Discovery through Advanced Computing

OAK RIDGE National Laboratory

Simmetrix

SCOREC SCIENTIFIC COMPUTATION RESEARCH CENTER

# Questions



Rensselaer

Rensselaer

why not change the world?®