Los Alamos
NATIONAL LABORATORY

EST. 1943

NNSA
National Nuclear Security Administration
Operated by Triad National Security, LLC for the U.S. Department of Energy's NNSA

LA-UR-19-24811

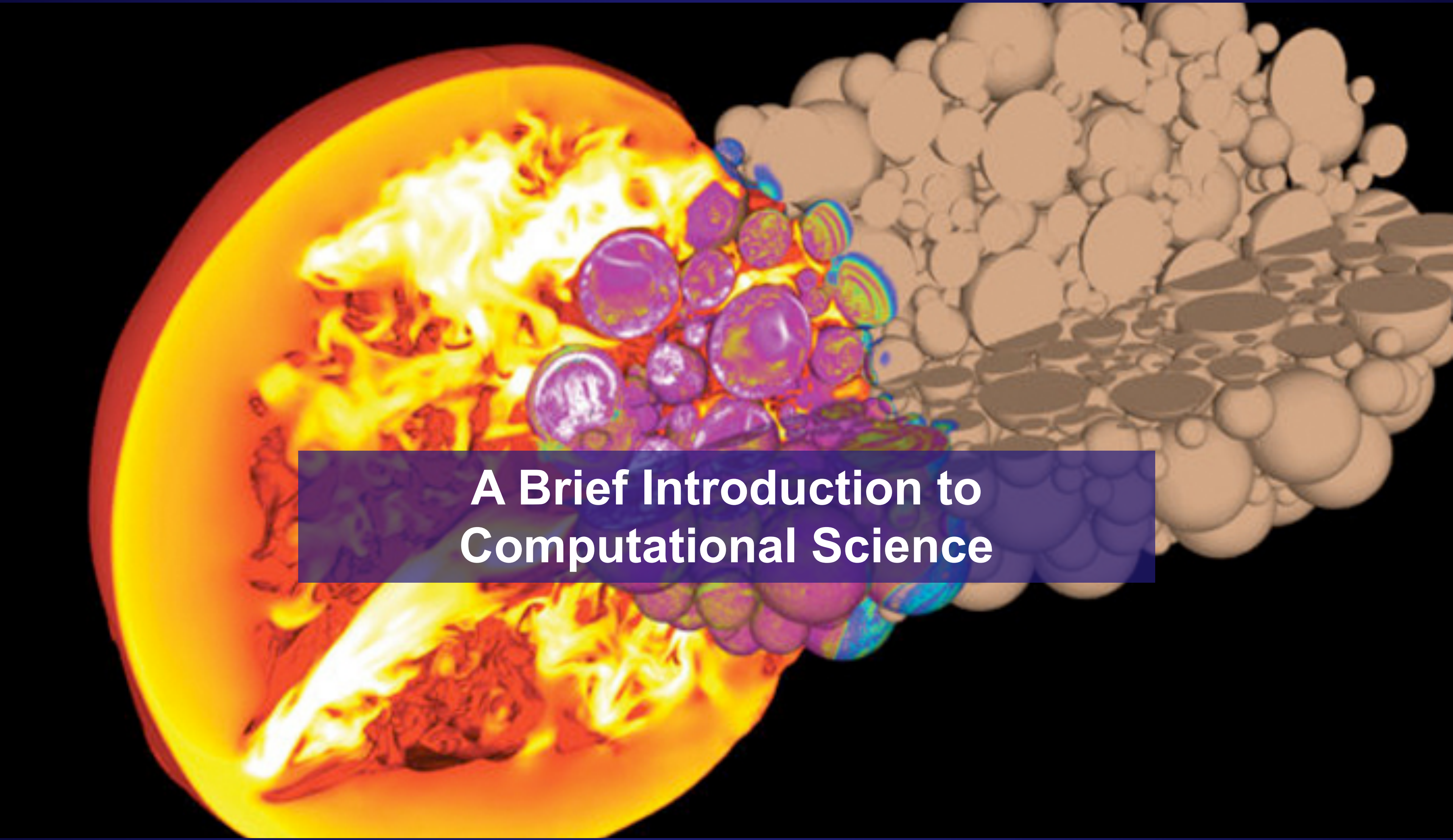
# Understanding Storage System Challenges for Parallel Scientific Simulations

**Brad Settlemyer**

Los Alamos National Laboratory

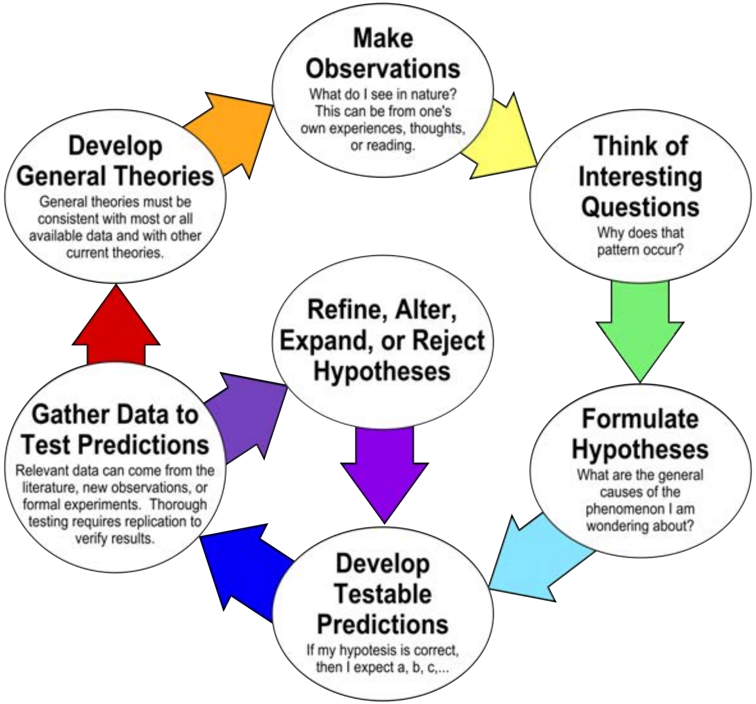
# Outline

- **Intro to Computational Science**
- **VPIC Overview**
  - PIC Introduction
  - VPIC Scientific Workflow
  - VPIC I/O Workloads
- **Real VPIC I/O Challenges**



# A Brief Introduction to Computational Science

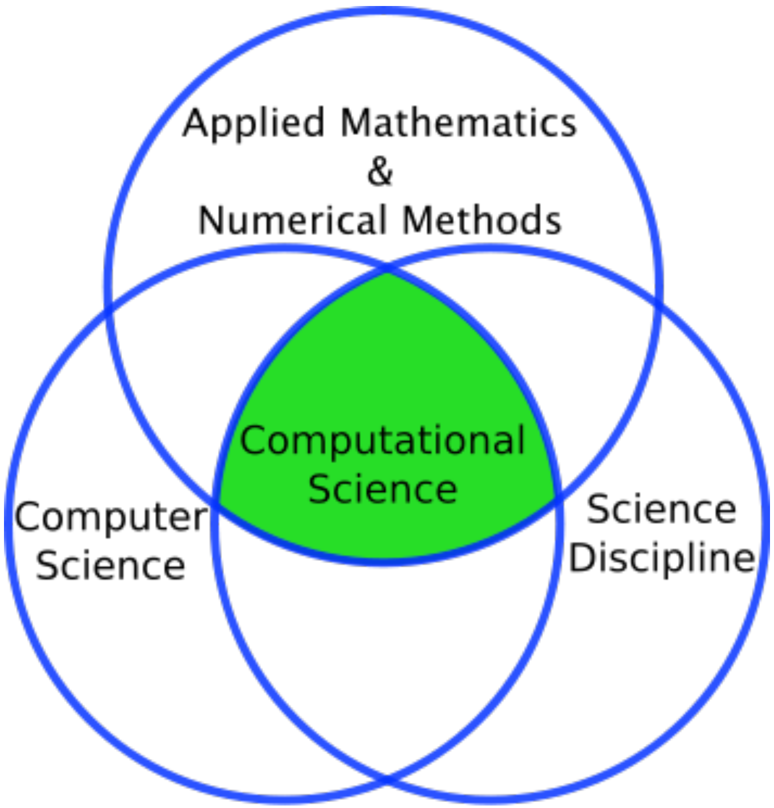
# The Traditional Scientific Method




- **A method for understanding the physical world**
- **Begins with observation**
- **Some parts of the physical world are not well suited to observation**
  - Galaxy formations/collisions
  - Climate models
  - Asteroid collisions
  - Fluid dynamics

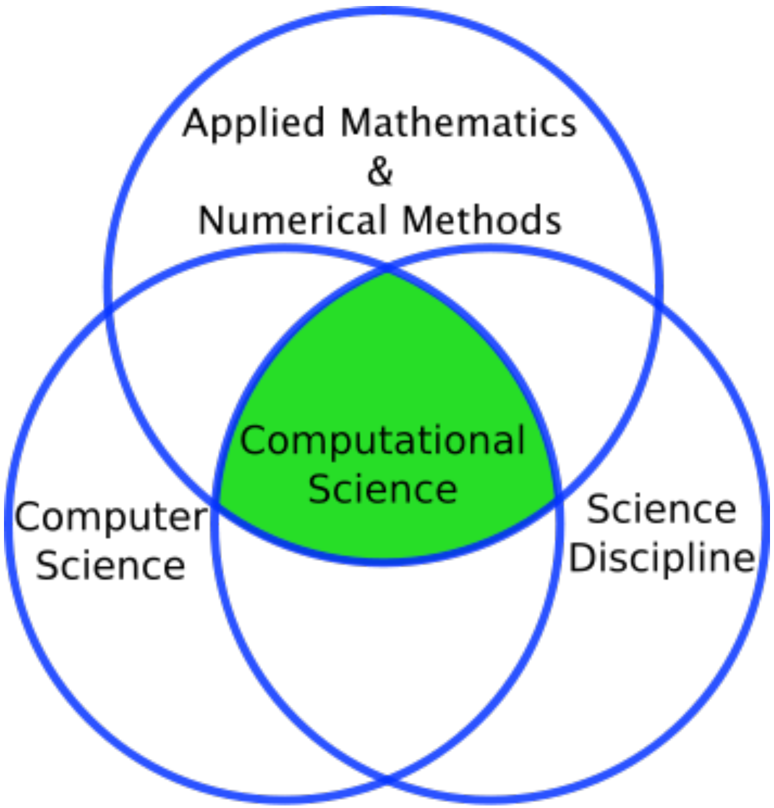
# Incorporating Simulation into The Scientific Method

- **Computer-based simulation enables new scientific inquiry**
  - Long time-scales
  - Complex interactions
  - Dangerous interactions
- **Computational Challenges**

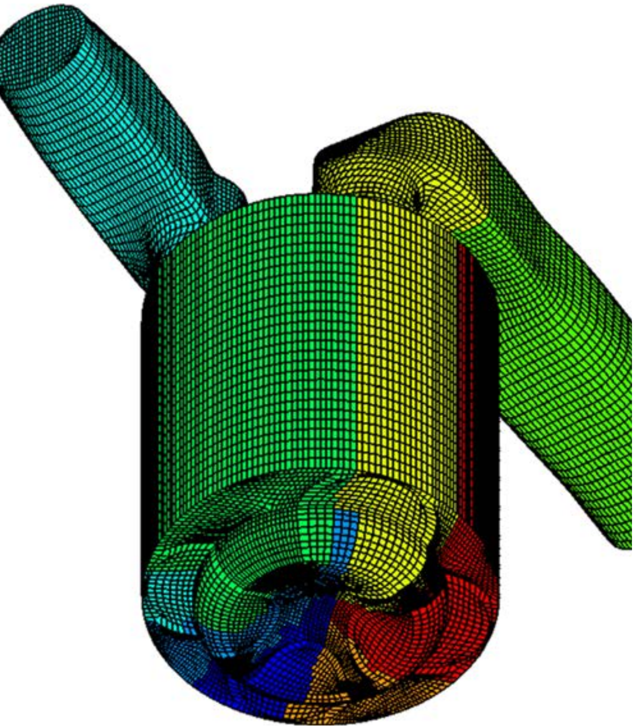


# Incorporating Simulation into The Scientific Method

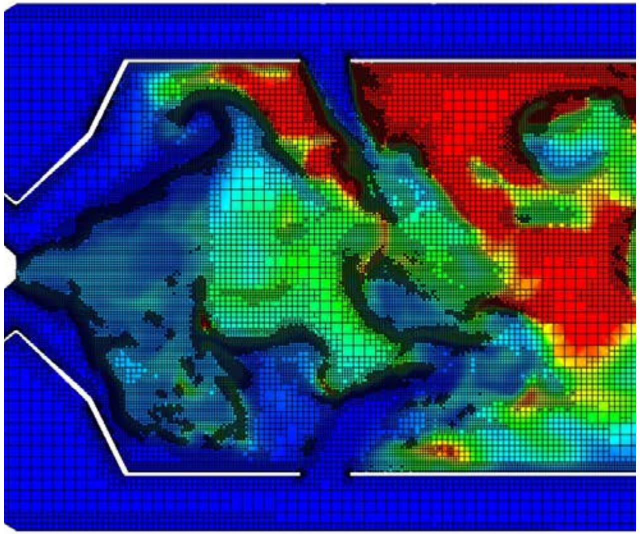
- **Computer-based simulation enables new scientific inquiry**
  - Long time-scales
  - Complex interactions
  - Dangerous interactions
- **Computational Challenges**
  - **Tightly-coupled simulations imply bulk-synchronous I/O**
  - **A single job may require months of compute time**



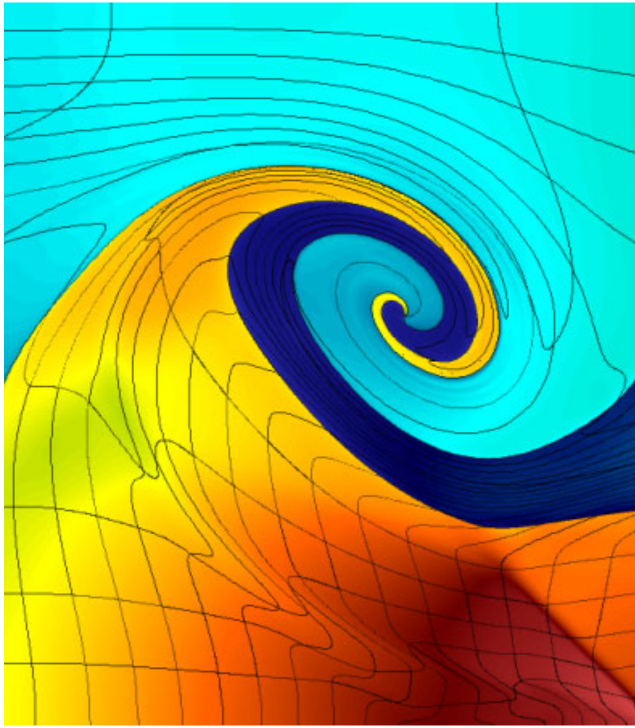
# 1. Create Mesh (Computational Science Workflow)



***Fixed Mesh***
*(Valves, cylinders)*

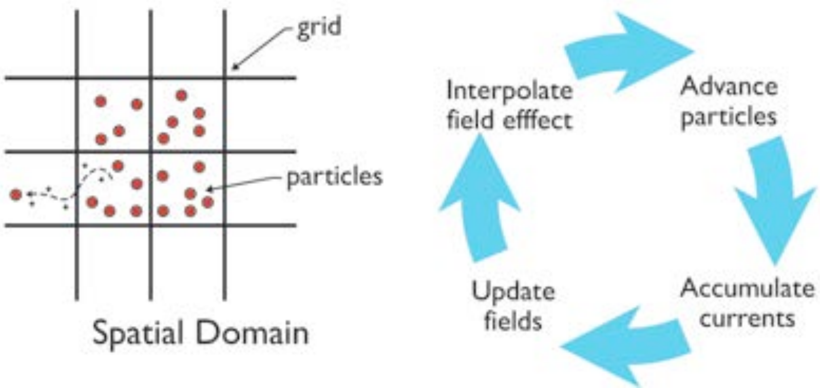


***Adaptive Mesh***
*(Turbulent combustion)*



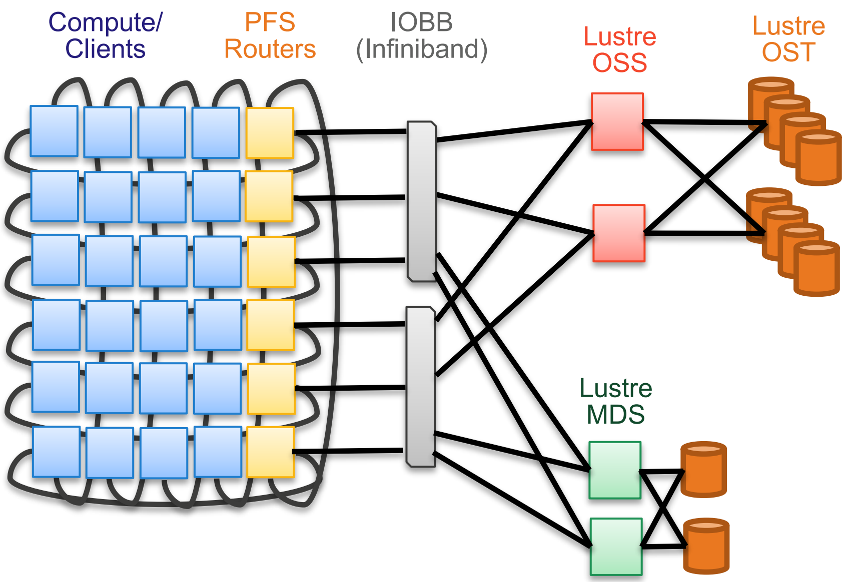
***Mesh deformation***
*(Shock propagating in fluid)*

# 2. Calculate Physics (Computational Science Workflow)



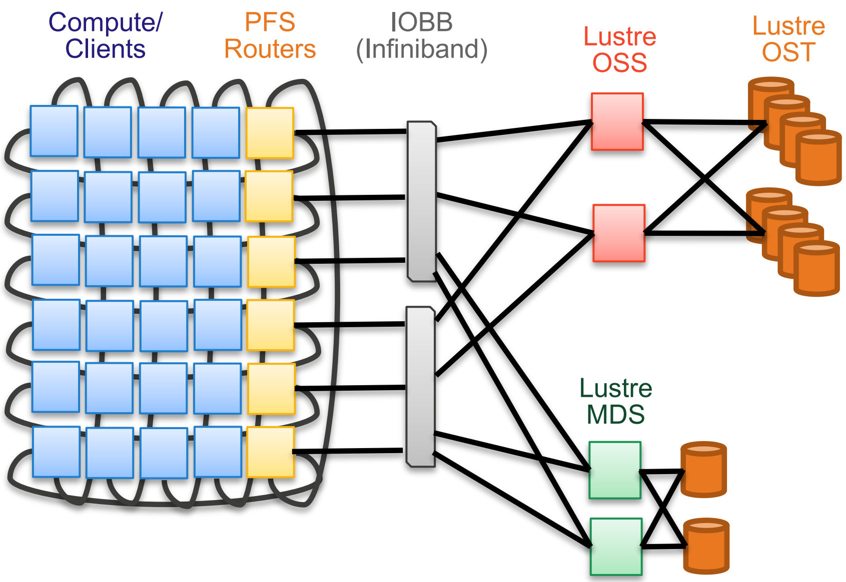

- **Often takes weeks or months**
- **Figure shows particle-in-cell (PIC) method**
  - Many other methods
  - Finite Element Methods
  - Finite Difference Methods
  - Monte Carlo Methods
- **The actual scientific question being answered typically favors one method or another**

# 3. Generate Data (Computational Science Workflow)



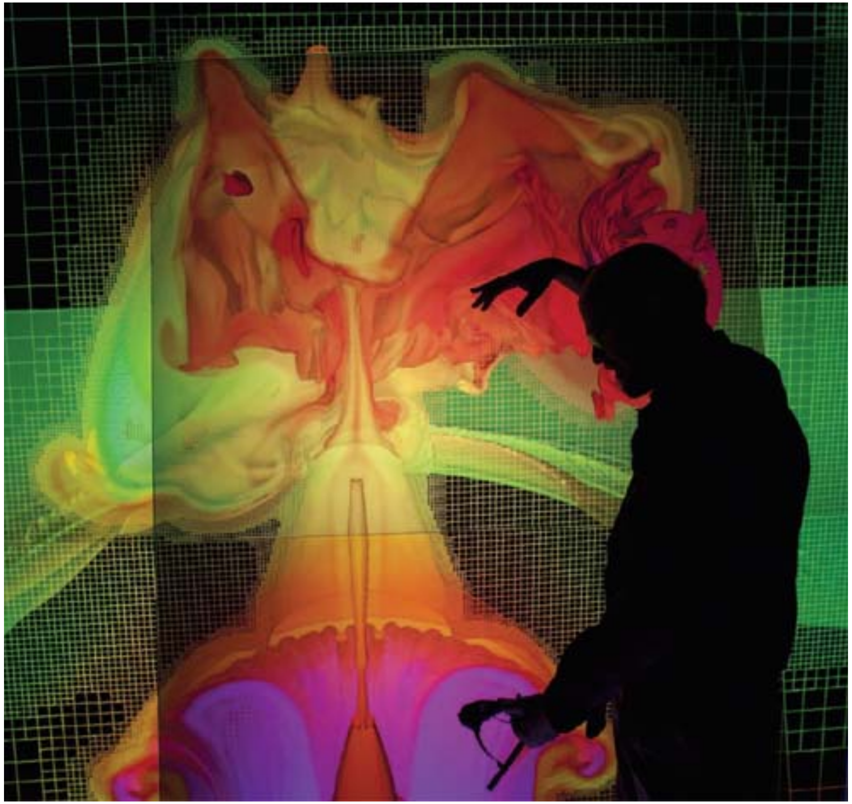

- **Simulation *pauses* when all processes reach some interesting point in the simulation**
  - Save state to protect against a failure (checkpoint/restart)
  - Save state for later analysis
  - Machine failures and scientific insight occur at different frequencies ☹
- **Once I/O is complete, simulation resumes**

# 3. Generate Data (Computational Science Workflow)




- **Simulation *pauses* when all processes reach some interesting point in the simulation**
  - Save state to protect against a failure (checkpoint/restart)
  - Save state for later analysis
  - Machine failures and scientific insight occur at different frequencies ☹
- **Once I/O is complete, simulation resumes**

# 4. Analyze Data (Computational Science Workflow)

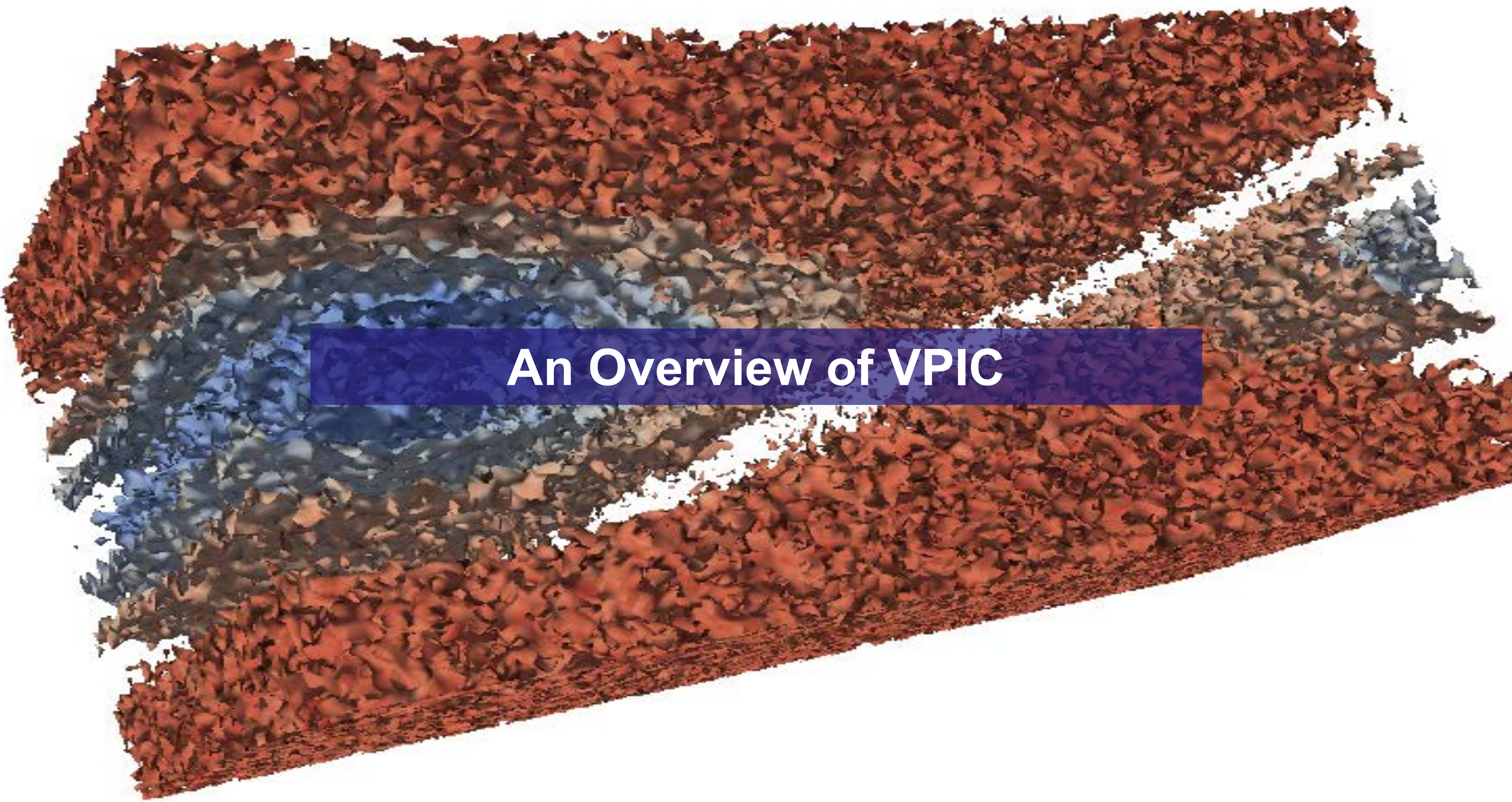


- **Scientists analyze/visualize simulation output**
  - Test and validate hypotheses
  - Source of new phenomena observations!
- **Automatic and in-situ analysis emerging as relevant to some scientific fields**

# What makes HPC computing unique and difficult?

- **Simulation Scale**
  - Frequently billions or trillions of mesh cells (1.5PB simulations on Trinity)
  - Simulations run for weeks or months
    - Longest simulation on Trinity: 7 months
    - Longest I've heard of: 18 months
- **Universe tends toward disorder (entropy increases)**
  - As simulation progresses, high % of memory is frequently modified
  - Tight-coupling, frequent communication due to boundary condition exchanges and load balancing over time
- **Large storage system requirements**
  - Checkpoint/restart bursts to support long running jobs
  - Capacity to store large quantities of restart dumps and analysis data

# An Overview of VPIC



# Quick Particle-In-Cell (PIC) Overview

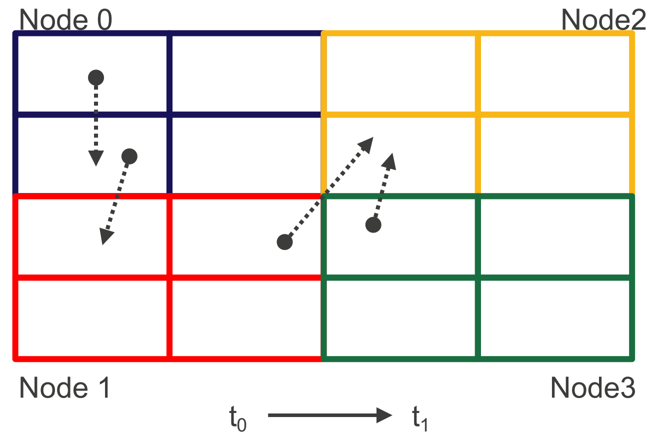
**Particles model material**

- Millions of particles per process
- Trillions of particles per simulation

**Fixed Mesh**

- Method extends to 3D well
- Each process maintains a contiguous chunk of the mesh
- Updates fields and materials

- **Solves the Maxwell-Boltzmann kinetic equations**
  - Applications in astrophysics, fusion, plasma interactions

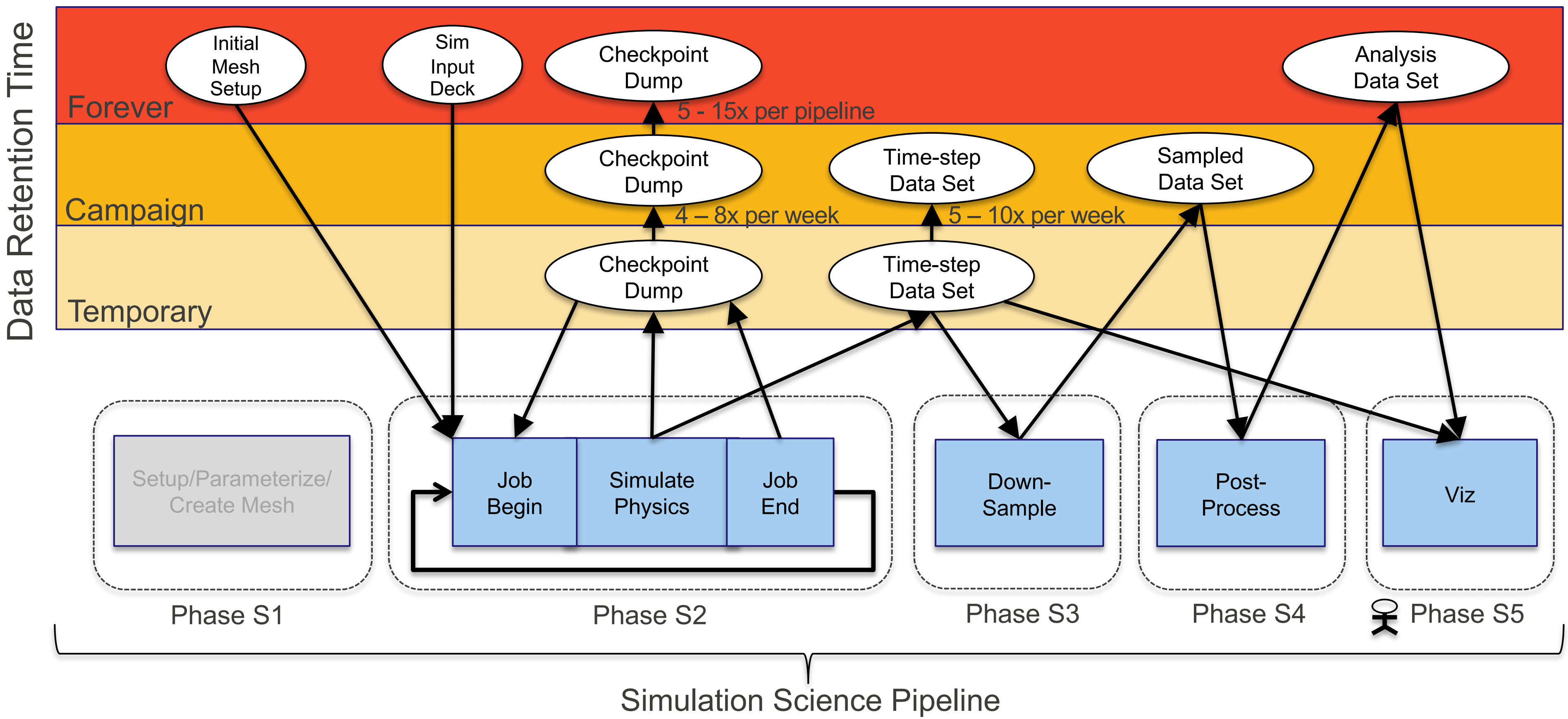



**PIC Introduction:** https://www.youtube.com/watch?v=CmhSWPpa_6w

# Why do I/O researchers use VPIC?

- **Excellent scaling**
  - Demonstrated across 4096 Trinity nodes (32k processes)
- **Flexible code**
  - Popular CS languages (engine is 16k sloc C/C++)
  - Supports MPI, OpenMP, and Pthreads
  - Can be field dominant or particle dominant
  - Can be compute/comm/memory intensive

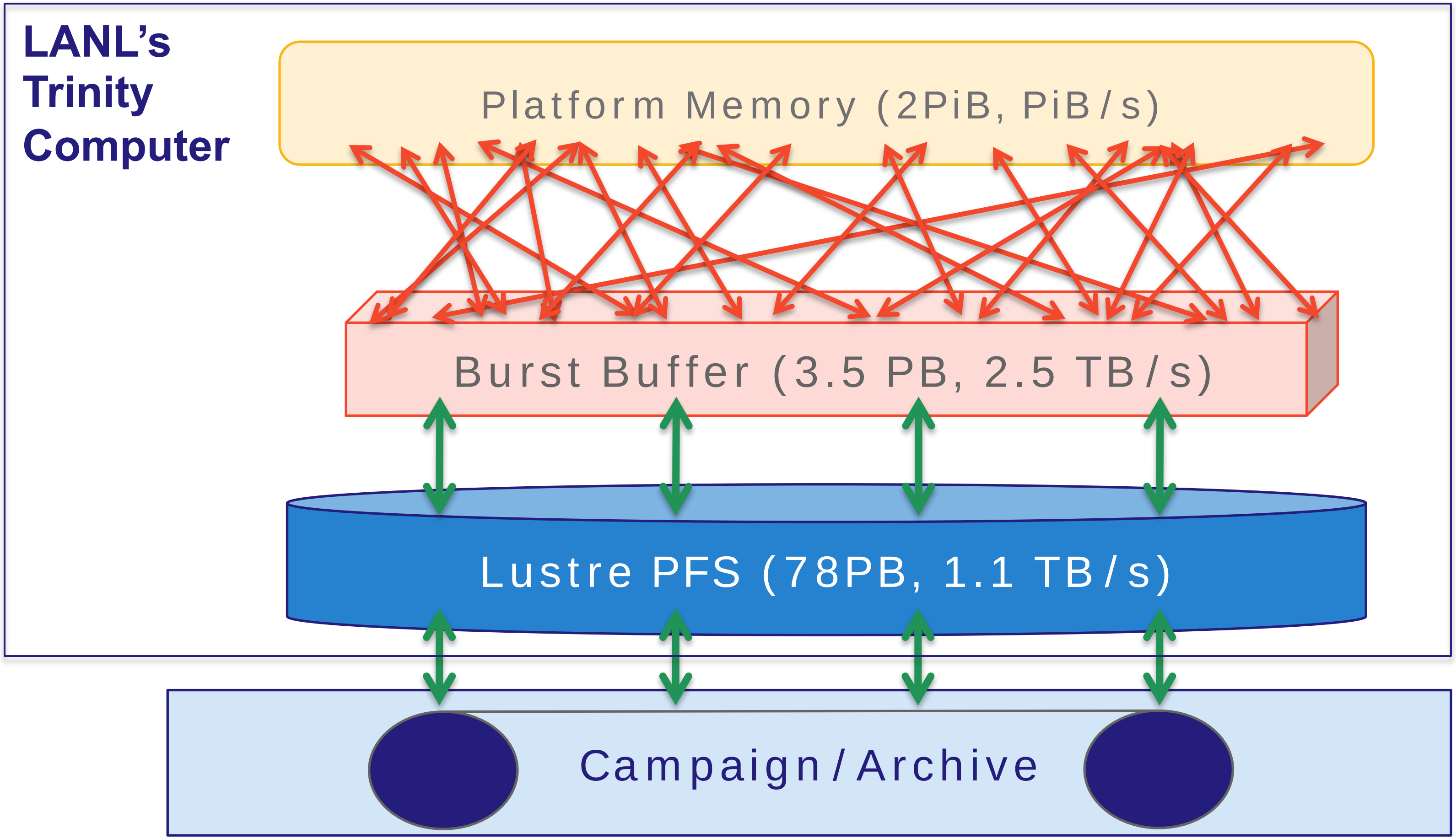
# VPIC's Simulation Science Workflow



# VPIC Checkpoint/Restart

- **Essential for simulations running for long duration over thousands of nodes**
  - Basic paradigms: N-N, N-M, N-1
  - Typically the largest consumer of bandwidth/capacity
- **In general must store both the particles and the fields**
  - Why?! Performance!
  - Approximately 80% of system memory
  - VPIC uses N-N file organization for checkpoint/restart

# HPC Checkpoint Workload

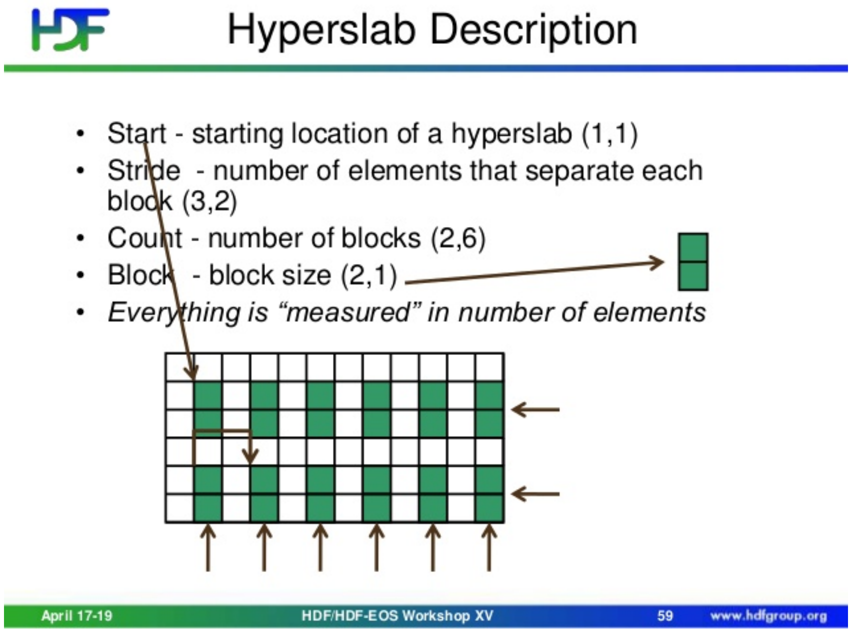


# VPIC Time Step Data Sets

- **Types of data**
  - Particles (32 – 48 bytes each)
  - Fields (typically <1k, but could be much more)
  - Cell Materials (often 0 bytes)
- **2 primary methods for data reduction**
  - Sampling (mean, spatial average, etc.)
  - Decimation
- **Scientist typically determines the processing methods needed**
  - Frequently not well optimized
  - Bound on bandwidth and performed on front ends

# VPIC Visualization

- **Format the data into a parallel visualization format**
  - Paraview, Ensight, VisIt, etc
- **Visualization workflows are typically bound on read performance**
  - Interactivity defeats pre-fetching algorithms
  - Viewing doesn't always occur along the contiguous dimension

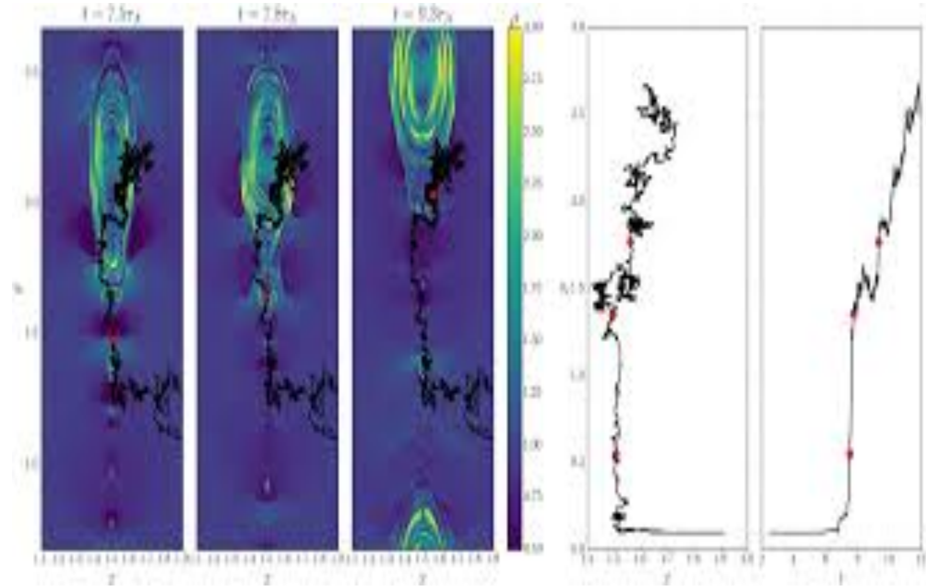


# Real VPIC I/O Challenges

# Tracking the Trajectory of High Energy particles

**Assumptions:**

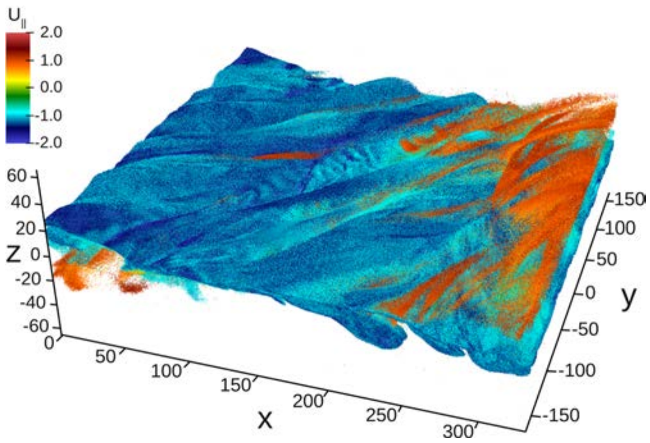
- Simulation has trillions of particles
- Highest energy particles only known at simulation end
- Insufficient memory to track the history of each particle



**Goal:**

- Determine if the trajectory of the high-energy particles follows Fermi acceleration between magnetic islands
- Highly selective queries

# Spatial distribution of particles within energy band




## Assumptions

- **Simulation has trillions of particles**
- **Energy distribution changing over time**

## Goals

- **Filter particles by energy band to examine the spatial location of energy bands**
- **Scan intensive workload**

Image and problem from "***Parallel I/O, Analysis, and Visualization of a Trillion Particle Simulation***," Byna, et al.

# The tip of the iceberg …

- Where are the largest clusters of similarly charged particles (i.e. magnetic islands)?
- Which particles have most recently moved between magnetic islands?
- Which particles are moving as groups and how are they moving?
- Is it possible to develop a taxonomy of formations that occur during a magnetic reconnection?
- And more …

# Conclusions

- **VPIC is an excellent resource for I/O researchers**
  - Open source
  - Popular programming languages (subsets)
  - Doesn't require exotic compilers
  - Highly scalable
  - Important scientific problems
- **VPIC scientists have real I/O problems**
  - A VPIC researcher has consumed all of the Trinity storage systems inodes
  - Extremely small writes are an unsolved problem
  - Data analysis performance severely limits current insight

# Thanks!