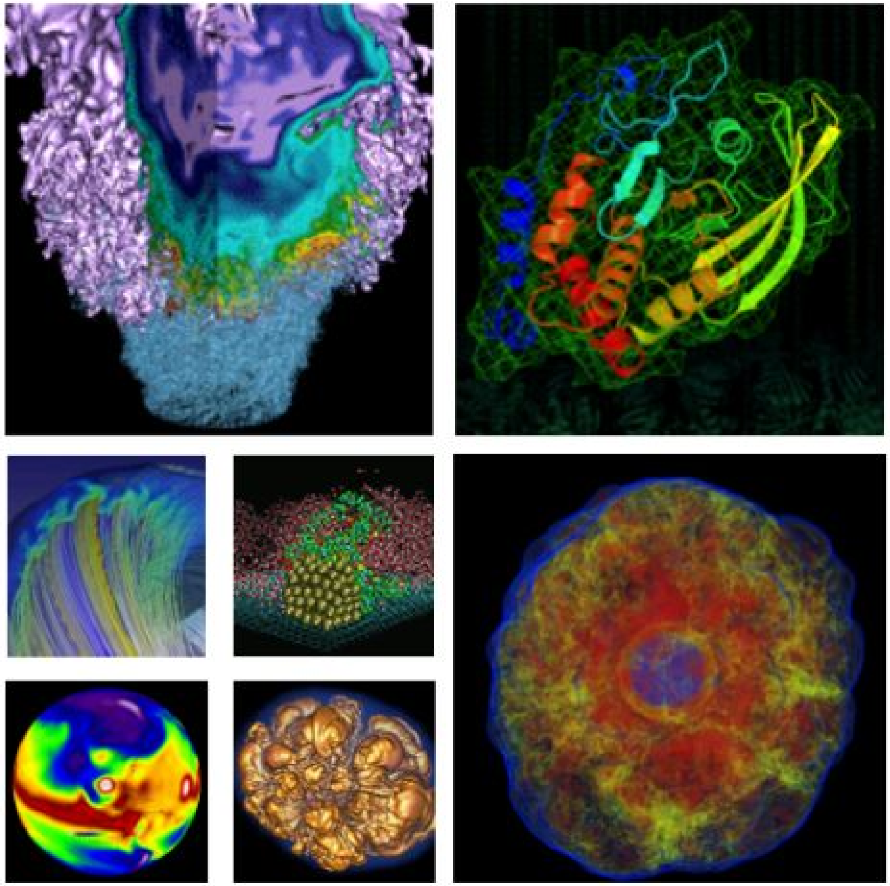
NERSC Users Group Monthly Meeting



NERSC

September 22, 2016

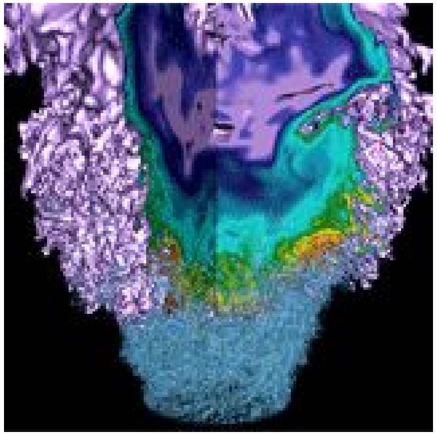
U.S. DEPARTMENT OF ENERGY

BERKELEY LAB

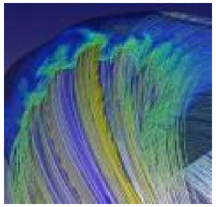
Agenda

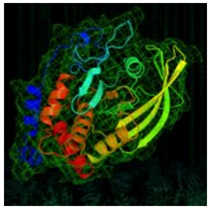
- Getting access to Cori Phase II
- Call for Applications: NESAP for Data program
- Data Management Tutorial
- Charm++ in a Nutshell for NERSC Users

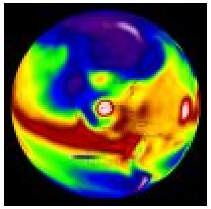
Getting Access to Cori Phase II

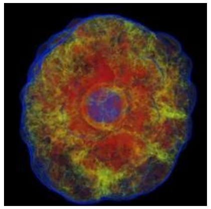


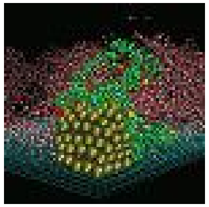
NERSC











Early access to Cori Phase II

- Before Cori enters production sometime in 2017 users will have an opportunity to gain access the Knight’s Landing partition
- We anticipate that this period of “early science” will begin sometime in late 2016 and last for several months
- Charging on Phase II will start mid-year 2017. DOE allocations managers will distribute additional time to projects in early 2017.

Goals of Early Science Period

- Allow users to test and optimize code for KNL
- Provide an opportunity for significant science runs to be completed, unconstrained by limited allocation awards
- Gather real-world user experiences to help guide configuration decisions and set machine charge factor based on realized performance

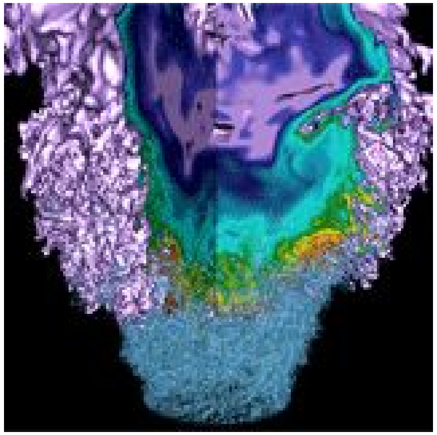
Criteria for Early Access

- **All users will get early access to a debug queue for testing and on-node optimization work**
- **NESAP (tier 1-3) teams that are ready for KNL will get early access to the full system**
- **Other codes that can demonstrate KNL readiness will get large-scale early access**
  - NERSC is developing a questionnaire / worksheet to evaluate application readiness for KNL
  - You can start preparing your codes now, see http://www.nersc.gov/users/computational-systems/cori/application-porting-and-performance/
  -

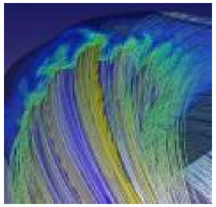
Early access to Cori Phase II

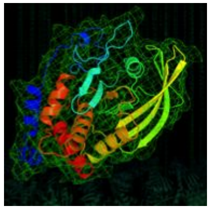
- **More to come about the application process in coming weeks**
- **Don’t wait for the application to start looking at your code’s performance; start now!**
- **We have lots of resources for optimizing codes on existing architectures that will benefit KNL as well**
  - https://www.nersc.gov/users/computational-systems/cori/application-porting-and-performance/

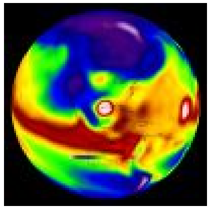
Call for Applications for NESAP Data Program

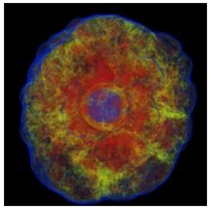


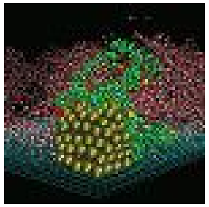
NERSC











U.S. DEPARTMENT OF ENERGY | Office of Science

BERKELEY LAB
Lawrence Berkeley National Laboratory

NESAP for Data Call: Open Oct 1

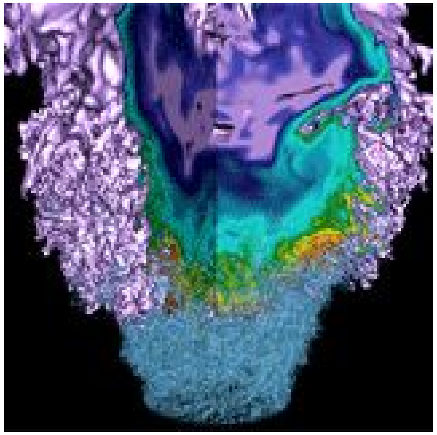
October 1: NERSC will begin accepting applications for participation in the NERSC Exascale Science Applications Program (NESAP) from developers of data-intensive science codes:

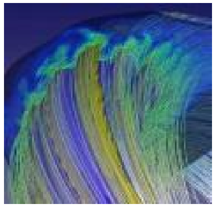
Processing and analysis of massive datasets acquired from experimental and observational sources.

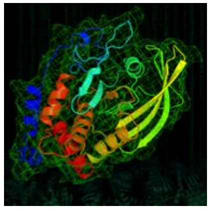
Goal: Enable data-intensive applications to fully utilize KNL on Cori.

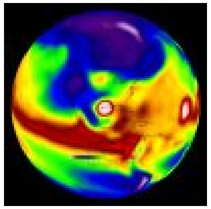
NESAP partners application teams with resources at NERSC, Cray, and Intel, and will last through final acceptance of the Cori system.

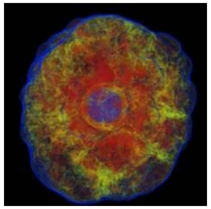
Data Management at NERSC

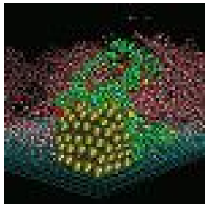












Quincey Koziol

NERSC Users Group

September 22, 2016

koziol@lbl.gov

Outline

- Best Practices and Guidelines
- I/O Libraries
- Databases
- Related topics

Outline

- **Best Practices and Guidelines**
- I/O Libraries
- Databases
- Related topics

Why Manage Your Data?

- “Data management is the development, execution and supervision of plans, policies, programs and practices that control, protect, deliver and enhance the value of data and information assets.”*

*DAMA-DMBOK Guide (Data Management Body of Knowledge) Introduction & Project Status

Data @ NERSC

NERSC offers a variety of services to support data-centric workloads. We provide tools in the areas of:

- **Data Analytics (statistics, machine learning, imaging)**
- **Data Management (storage, representation)**
- **Data Transfer**
- **Workflows**
- **Science Gateways**
- **Visualization**

http://www.nersc.gov/users/data-analytics/

http://bit.ly/NUG16-09-DM

Data @ NERSC

NERSC offers a variety of services to support data-centric workloads. We provide tools in the areas of:

- ~~Data Analytics (statistics, machine learning, imaging)~~
- **Data Management (storage, representation)**
- Data Transfer
- Workflows
- Science Gateways
- Visualization

http://www.nersc.gov/users/data-analytics/

http://bit.ly/NUG16-09-DM

General Recommendations

- NERSC recommends the use of modern, scientific I/O libraries (HDF5, NetCDF, ROOT) to represent and store scientific data.
- We provide database technologies (MongoDB, SciDB, MySQL, PostGreSQL) for our users as a complementary mechanism for storing and accessing data.
- Low-level, POSIX I/O from applications to NERSC file systems, if necessary. Details here:

  http://www.nersc.gov/users/storage-and-file-systems/

Notes on NERSC File I/O

- Use the local scratch file system on Edison and Cori for best I/O rates.
- For some types of I/O you can further optimize I/O rates using a technique called file striping.
- Keep in mind that data in the local scratch directories are purged, so you should always backup important files to HPSS* or project space.
- You can share data with your collaborators using project directories. These are directories that are shared by all members of a NERSC repository.

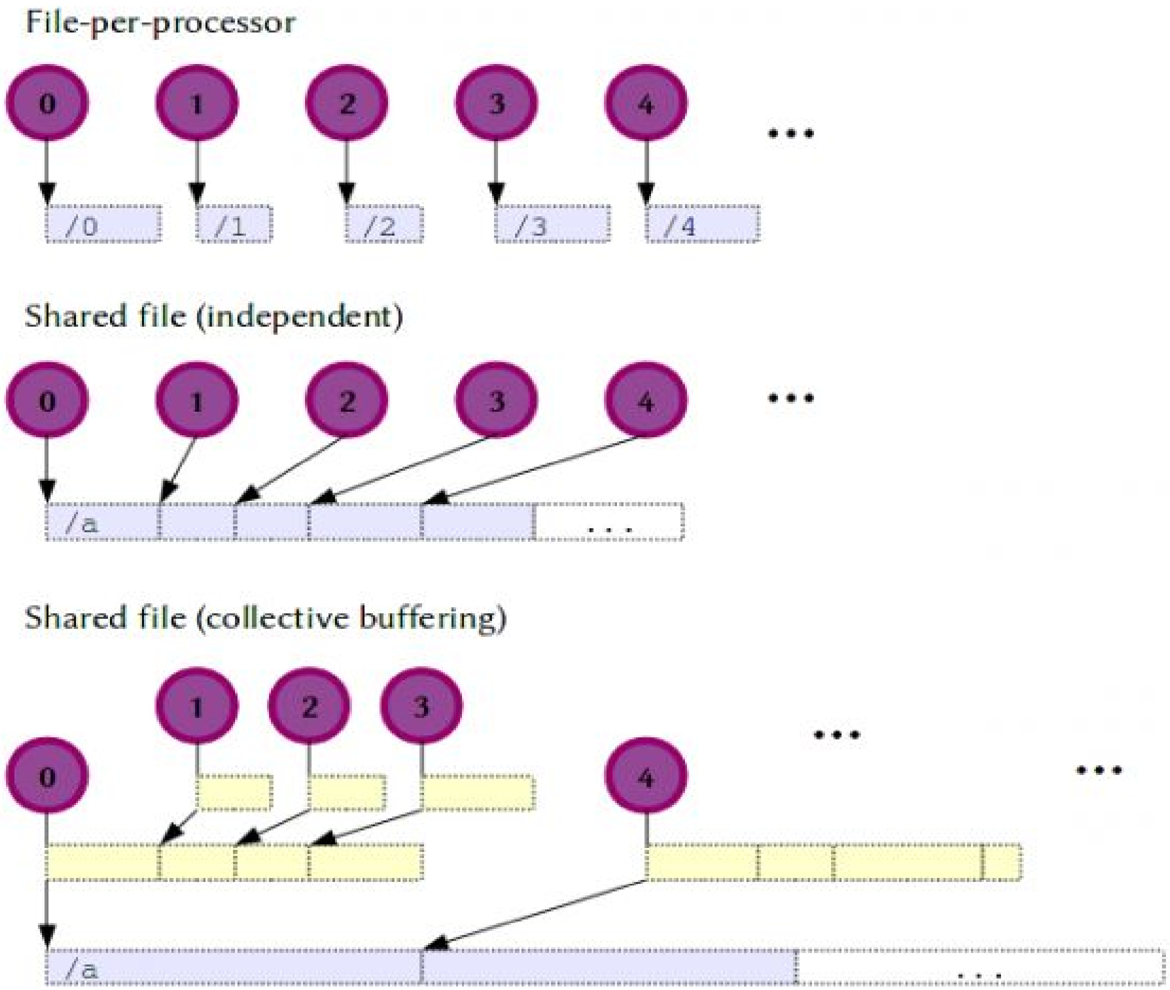
*HPSS: http://www.nersc.gov/users/storage-and-file-systems/hpss/getting-started/

Introduction to Scientific I/O

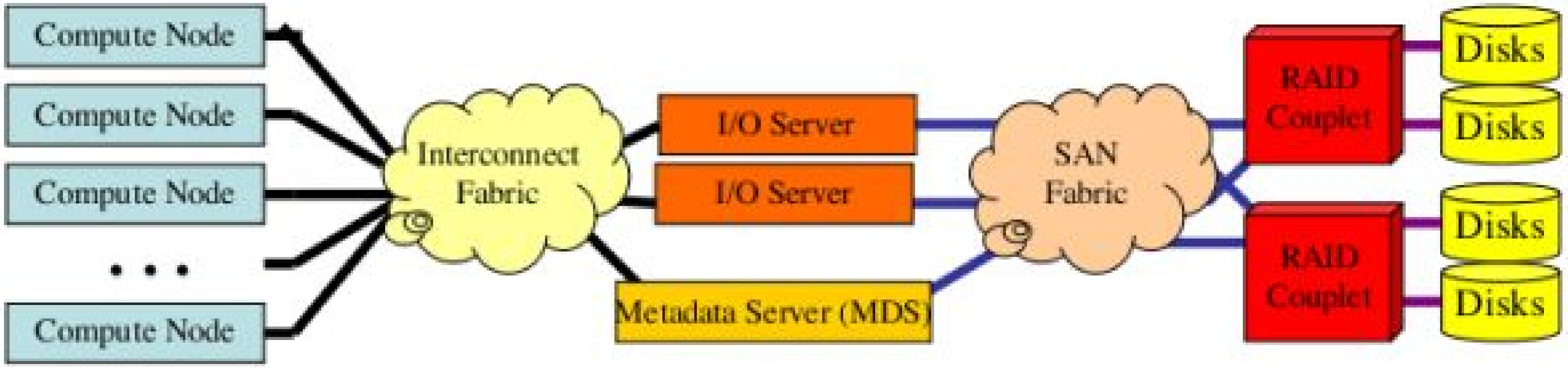
I/O is commonly used by scientific applications to achieve goals like:

- Storing numerical output from simulations for later analysis or workflow stages
- Implementing 'out-of-core' techniques for algorithms that process more data than can fit in system memory and must page in data from disk
- Checkpointing application state to files, in case of application or system failure.

Types of Application I/O to Parallel File Systems



Lustre



- Scalable, POSIX-compliant parallel file system designed for large, distributed-memory systems
- Uses a server-client model with separate servers for file metadata and file content

MPI Collective I/O

- ***Collective I/O*** **refers to a set of optimizations available in many implementations of MPI-IO that improve the performance of large-scale IO to shared files.**
- **To enable these optimizations, you must use the *collective* calls in the MPI-IO library that end in _*all***
  - For instance: MPI_File_write_at_all().
- **And, all MPI tasks in the given MPI communicator must participate in the collective call, even if they are not performing any IO operations.**
- **The MPI-IO library has a heuristic to determine whether to enable *collective buffering*, the primary optimization used in collective mode.**

Outline

- Best Practices and Guidelines
- **I/O Libraries**
- Databases
- Related topics

Why I/O Middleware?

- **The complexity of I/O systems poses significant challenges in investigating the root cause of performance loss.**
- **Use of I/O middleware for writing parallel applications has been shown to greatly enhance developer's productivity.**
  - Such an approach hides many of the complexities associated with performing parallel I/O, rather than relying purely on programming language aids and parallel library support, such as MPI.

I/O Middleware @ NERSC

- **HDF5**
  - A data model and set of libraries & tools for storing and managing large scientific datasets.
- **netCDF**
  - A set of libraries and machine-independent data formats for creation, access, and sharing of array-oriented scientific data.
- **ROOT**
  - A self-describing, column-based binary file format that allows serialization of a large collection of C++ objects and efficient subsequent analysis.
- **Others**
  - http://www.nersc.gov/users/data-analytics/data-management/i-o-libraries/i-o-library-list/

HDF5

- **The Hierarchical Data Format v5 (HDF5) library is a portable I/O library used for storing scientific data.**
- **The HDF5 technology suite includes:**
  - A versatile data model that can represent very complex data objects and a wide variety of metadata.
  - A completely portable file format with no limit on the number or size of data objects in the collection.
  - A software library that runs on a range of computational platforms, from laptops to massively parallel systems, and implements a high-level API with C, C++, Fortran 90, and Java interfaces.
  - A rich set of integrated performance features that allow for access time and storage space optimizations.
  - Tools and applications for managing, manipulating, viewing, and analyzing the data in the collection.
- **HDF5's 'object database' data model enables users to focus on high-level concepts of relationships between data objects rather than descending into the details of the specific layout of every byte in the data file.**

netCDF

- **netCDF (Network Common Data Form) is a set of software libraries and machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data.**
- **netCDF is:**
  - Typically used in the climate field
  - More constrained than HDF5
  - At a higher level of abstraction
- **More netCDF information here:**

  http://www.unidata.ucar.edu/software/netcdf/docs/netcdf/

ROOT

- **A set of object oriented frameworks with the functionality needed to handle and analyze large amounts of data in a very efficient way.**
- **ROOT is written in C++ and uses an indexed tree format as it base data unit, with substructures called branches and leaves.**
- **Originally designed for particle physics, its usage has extended to other data-intensive fields like astrophysics and neuroscience.**
  - ROOT is mainly used for data analysis at NERSC.
- **ROOT Docs: https://root.cern.ch/drupal/**

Outline

- Best Practices and Guidelines
- I/O Libraries
- **Databases**
- Related topics

Databases @ NERSC

- NERSC supports the provisioning of databases to hold large scientific datasets, as part of the science gateways effort.
- Data-centric science often benefits from database solutions to store scientific data or metadata about data stored in more traditional file formats like HDF5, netCDF or ROOT.
- Our database offereings are targeted toward large data sets and high performance. Currently we support:
  - MySQL
  - PostgreSQL
  - MongoDB
  - SciDB
- If you would like to request a database at NERSC please fill out this form and you'll be contacted by NERSC staff: http://www.nersc.gov/users/data-analytics/data-management/databases/science-database-request-form/

PostgreSQL

- PostgreSQL is an object-relational database. It is known for having powerful and advanced features and extensions as well as supporting SQL standards.
- NERSC provides a set of database nodes for users that wish to use PostgreSQL with their scientific applications.
- PostgreSQL documentation here:

  http://www.postgresql.org/docs/

MySQL

- **MySQL is a very popular and powerful open-source relational database.**
- **It has many features:**
  - Pluggable Storage Engine Architecture, with multiple storage engines:
    - InnoDB
    - MyISAM
    - NDB (MySQL Cluster)
    - Memory
    - Merge
    - Archive
    - CSV
    - and more
  - Replication to improve application performance and scalability
  - Partitioning to improve performance and management of large database applications
  - Stored Procedures to improve developer productivity
  - Views to ensure sensitive information is not compromised
  - …
- **MySQL user documentation:**

  http://dev.mysql.com/doc/

SciDB

- SciDB is a parallel database for array-structured data, good for TBs of time series, spectra, imaging, etc.
- A full ACID database management system that stores data in multidimensional arrays with strongly-typed attributes (aka fields) within each cell.
- SciDB User Documentation:

  https://paradigm4.atlassian.net/wiki/display/ESD/SciDB+Documentation
- To request access to NERSC SciDB instances, please email consult@nersc.gov

MongoDB

- A cross-platform document-oriented database.
- Classified as a *NoSQL* database, MongoDB eschews the traditional table-based relational database structure in favor of JSON-like documents with dynamic schemas, making the integration of data in certain types of applications easier and faster.
- MongoDB user documentation:

  https://docs.mongodb.com/v2.6/

Outline

- Best Practices and Guidelines
- I/O Libraries
- Databases
- **Related topics**

Data Transfer

- **The best ways to get your data into and out of NERSC.**
- **Several methods supported:**
  - Globus – A service for fast reliable managed data transfers.
    http://www.nersc.gov/users/storage-and-file-systems/transferring-data/globus-online/
  - GridFTP - A high-performance, secure, reliable data transfer protocol optimized for high-bandwidth wide-area networks.
    http://www.nersc.gov/users/software/grid/data-transfer/
  - Data Transfer Nodes - Optimized for moving data into and out of NERSC
    http://www.nersc.gov/users/storage-and-file-systems/data-transfer-nodes/

Workflow Tools

- **Supporting data-centric science often involves the movement of data across file systems, multi-stage analytics and visualization.**
- **Workflow technologies can improve the productivity and efficiency of data-centric science by orchestrating and automating these steps.**

Science Gateways

- **A science gateway is a web-based interface to access HPC computers and storage systems.**
- **Gateways allow science teams to access data, perform shared computations, and generally interact with NERSC resources over the web:**
  - To improve ease of use in HPC so that more scientists can benefit from NERSC resources
  - To create collaborative workspaces around data and computing for science teams that use NERSC
  - To make your data accessible and useful to the broader scientific community.
- **NERSC Science Gateways info:**

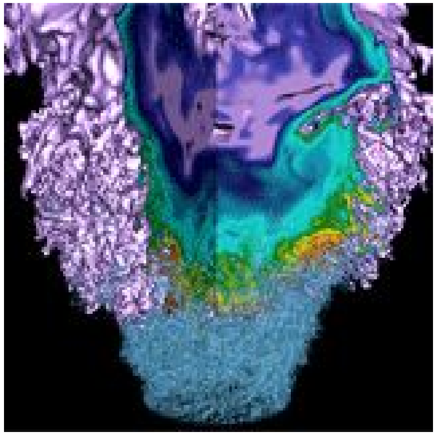
  http://www.nersc.gov/users/data-analytics/science-gateways/

Data Visualization

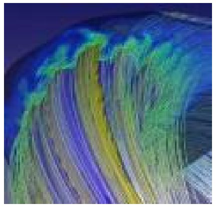
- **Scientific Visualization is the process of creating visual imagery from raw scientific data.**
- **NERSC Supported packages:**
  - ParaView
    - An open-source, multi-platform data analysis and visualization application. Data exploration can be done interactively in 3D or using batch processing.
      http://www.nersc.gov/users/data-analytics/data-visualization/paraview-2/
  - VisIt
    - VisIt is a point-and-click 3D scientific visualization application that supports most common visualization techniques (e.g., isocontouring and volume rendering) on structured and unstructured grids.
      http://www.nersc.gov/users/data-analytics/data-visualization/visit-2/
  - NCAR Graphics
    - NCAR Graphics is a collection of graphics libraries that support the display of scientific data. The low-level utilities (LLUs) are the traditional C and Fortran interfaces for contouring, mapping, drawing field flows, drawing surfaces, drawing histograms, drawing X/Y plots, labeling, and more.
      http://www.nersc.gov/users/data-analytics/data-visualization/ncar/

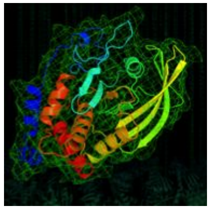
Questions, Comments, Feedback?

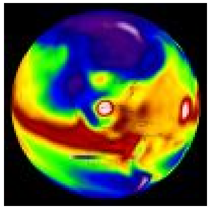
Charm++ in a Nutshell for NERSC Users

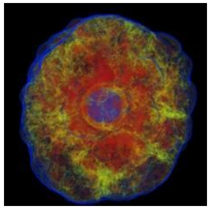


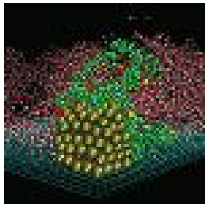
NERSC











U.S. DEPARTMENT OF ENERGY | Office of Science

BERKELEY LAB
Lawrence Berkeley National Laboratory

Motivations - Variability

- **Applications**
    - Irregular problems
    - Non-uniform decomposition
    - Adaptive refinement
    - Local iterative methods
    - Multi-physics
    - Multi-module
- **Systems**
    - Noise
    - Network Contention
    - CPU speeds

Productivity

- **Common functionality - good, shared implementations**
    - Easy Computation/Communication Overlap
    - Object Location
    - Load Balancing
    - Checkpoint/restart (on different processor counts!)
- **Addresses next-order concerns**
    - Node and Network locality
    - Temperature/Power/Energy
    - LB Frequency & Strategy
    - Dynamic Critical Paths

Philosophy

- **Overdecomposition:**
  Many units of parallelism per processor

- **Asynchrony:**
  Units designed to advance based on
  their own data dependencies

- **Migratability:**
  Units can be moved to run on any processor

Model: ‘Chare’ Objects

- C++ objects
- Organized into indexed collections
- Each collection may have its own indexing scheme
  - 1D .. n-D
  - Sparse
  - Bitvector or string as an index
- Chares communicate via asynchronous invocations of designated remote “`entry`” methods
  - `A[i].foo(…);  //` `A` is the name of a collection,
    `//` `i` is the index of the particular chare.
- RTS deals with processor location and reassignment

Big Charm++ Applications

Application	Domain	Predecessor	Scale
NAMD	Classical MD	PVM	500k+
ChaNGa	N-body gravity & SPH	MPI	500k+
EpiSimdemics	Agent-based epidemiology	MPI	500k+
OpenAtom	Electronic Structure	MPI	500k+
ROSS	PDES	MPI	500k+
SpECTRE	Relativistic MHD		500k+
FreeON/SpAMM	Quantum Chemistry	OpenMP	50k
Enzo-P/Cello	Astrophysics/Cosmology	MPI	32k
SDG	Elastodynamic fracture		10k

Other Cool Charm++ Applications

Application	Domain	Predecessor	Scale
ADHydro	Systems Hydrology		1000
Disney ClothSim	Cloth dynamics with rigid bodies	TBB	768
Particle Tracking	Velocimetry reconstruction		512
JetAlloc	Stochastic mixed-integer programs		480

Interoperability with MPI

- MPI code calls Charm++, Charm++ code calls MPI
- Time-division or space-division
- EpiSimdemics uses MPI-IO
- Chombo AMR GR code ‘CHARM’ uses Charm++ sorting
- See paper for more details http://ppl.cs.illinois.edu/papers/15-02

Features & Ecosystem

- Automatic offline & online fault tolerance
  - Checkpoint in one line, transparent restart, any number of processors
- Plethora of LB strategies
  - Separate from application logic
  - Easy to plug in your own
- Scalable tools
  - CharmDebug parallel debugger
  - LiveViz online visualization client
  - Projections performance analysis tool

Resources

- http://charmplusplus.org (with tutorials, manual, examples)
- Charm++ Course: https://wiki.illinois.edu/wiki/display/cs598lvk/Lectures
- Charm++ ATPESC Tutorial: https://extremecomputingtraining.anl.gov/sessions/presentation-charm-motivations-and-basic-ideas/

Building (w/) Charm++ on Edison

Clone:

```
git clone http://charm.cs.illinois.edu/gerrit/charm.git
```

Development:

```
./build charm++ gni-crayxc -j8 -g
```

Production:

```
./build charm++ gni-crayxc --with-production -j8
```

Compile:

```
~/path/to/charm/bin/charmc -c file.cpp -o file.o
```

Questions, Comments?

NERSC

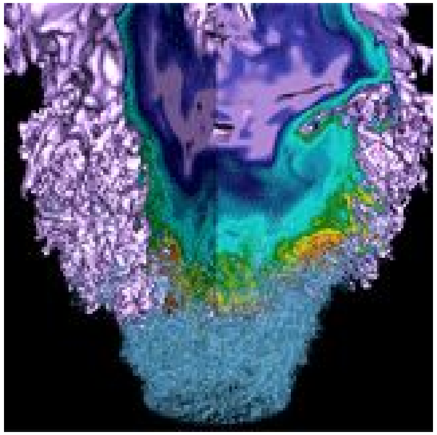
U.S. DEPARTMENT OF ENERGY

Office of Science

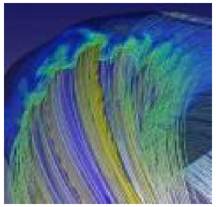
BERKELEY LAB

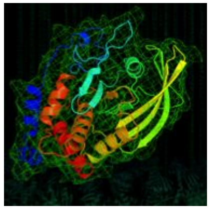
Lawrence Berkeley National Laboratory

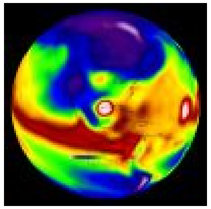
Cori Phase I & II Integration Progress

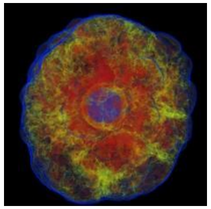


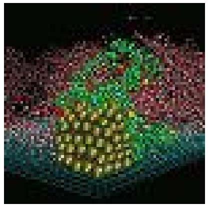
NERSC











U.S. DEPARTMENT OF ENERGY | Office of Science

BERKELEY LAB
Lawrence Berkeley National Laboratory