

# Visualization and Analysis with Visit

## DOE ACTS Collection Workshop 2012

**Cyrus Harrison**

*cyrush@llnl.gov*

*Lawrence Livermore National Laboratory*

**Harinarayan Krishnan**

*hkrishnan@lbl.gov*

*Lawrence Berkeley National Laboratory*

August 17, 2012



LLNL-PRES-576392

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC



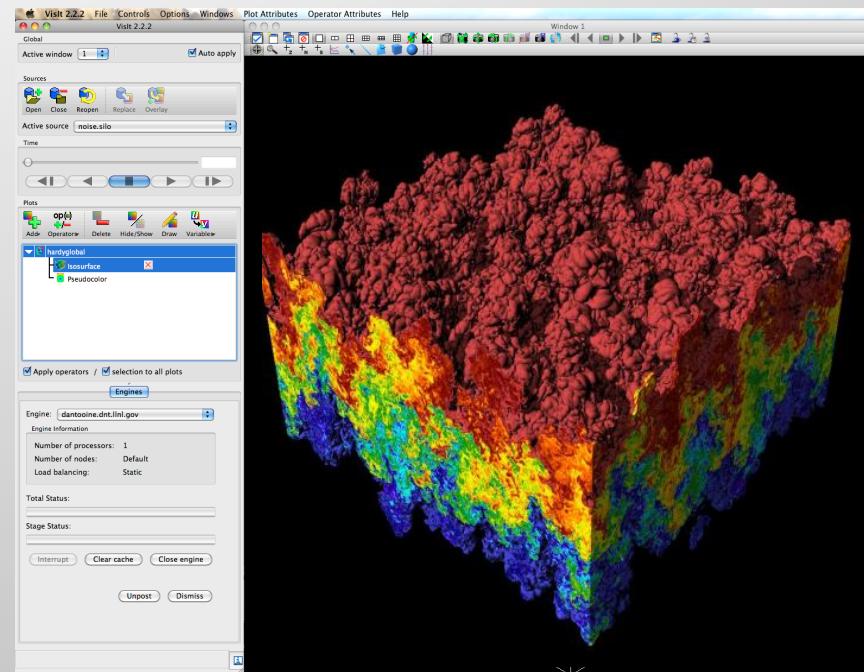
# Schedule

- 1:30 – 1:45: **VisIt Project Introduction**
- 1:45 – 2:20: **VisIt Basics**
- 2:20 – 2:30: **VisIt Installation Help**
- 2:30 – 2:40: ***Break***
- 2:40 – 2:45: **VisIt Client / Server Demo**
- 2:45 – 3:20: **Scripting VisIt**
- 3:20 – 3:30: **Wrap-up & Tips**

# VisIt Project Introduction

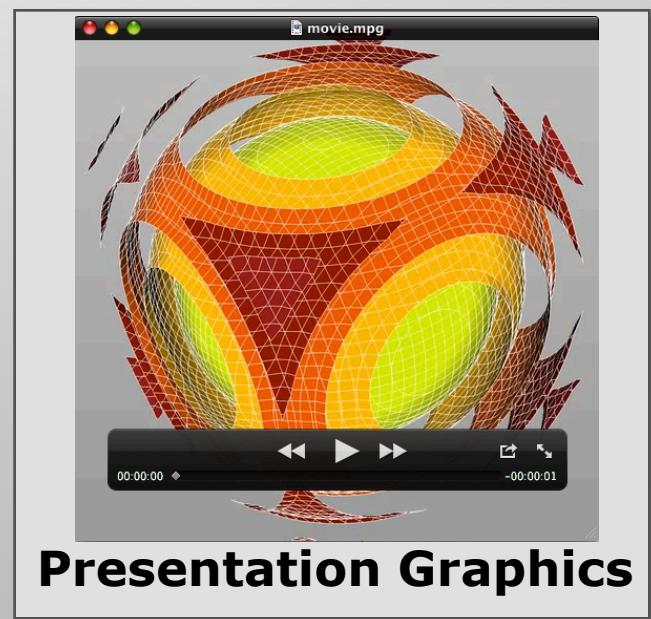
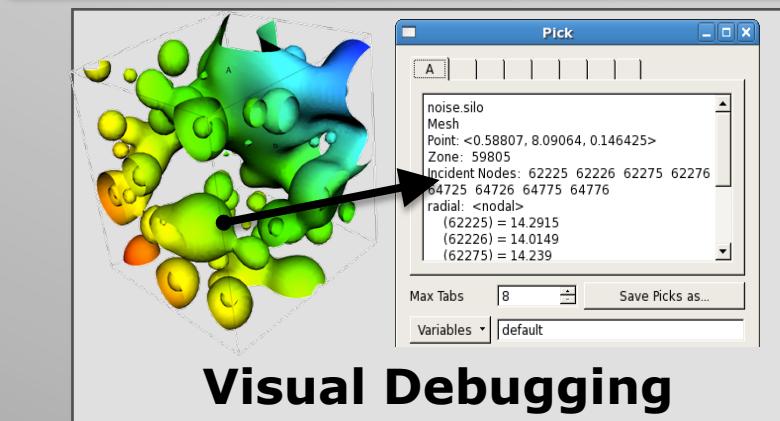
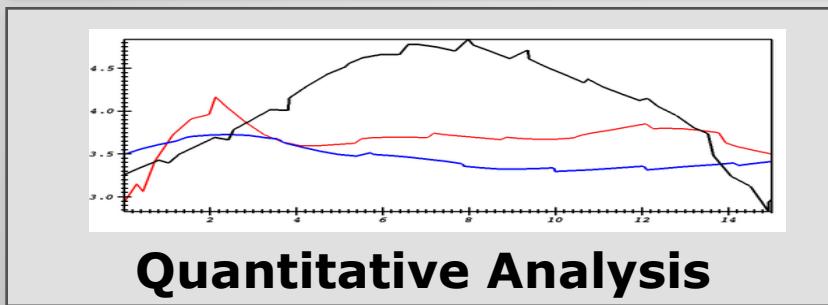
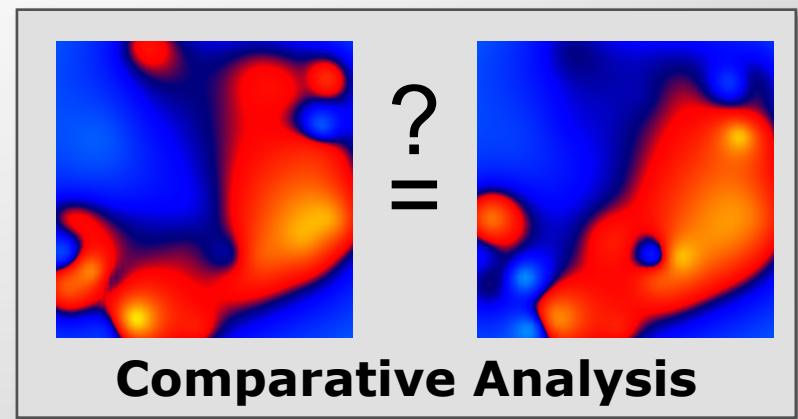
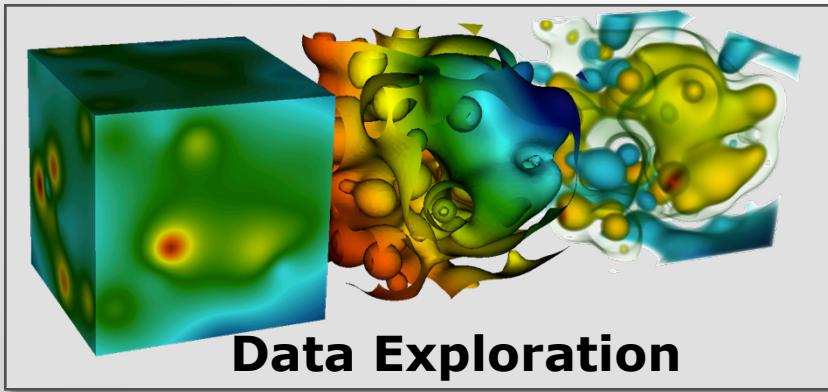
# VisIt is an open source, turnkey application for data analysis and visualization of mesh-based data.

- Production end-user tool supporting scientific and engineering applications.
- Provides an infrastructure for parallel post-processing that scales from desktops to massive HPC clusters.
- Source released under a BSD style license.

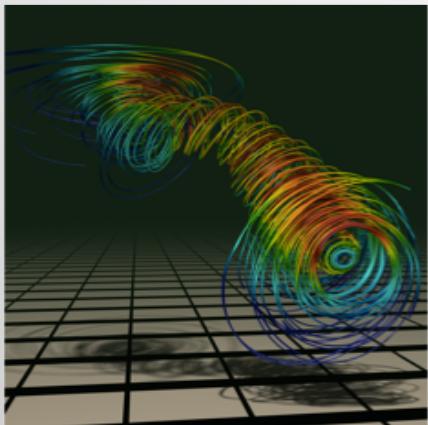


**Density Isovolum of a  
3K<sup>3</sup> (27 Bz) dataset**

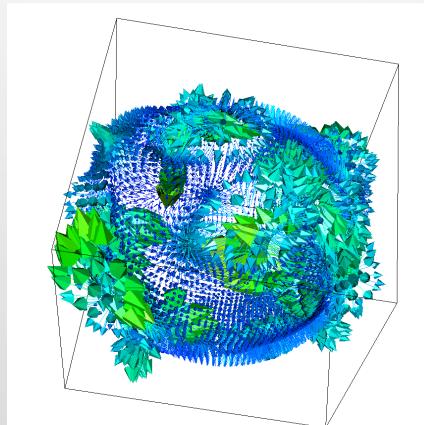
# VisIt supports a wide range of use cases.



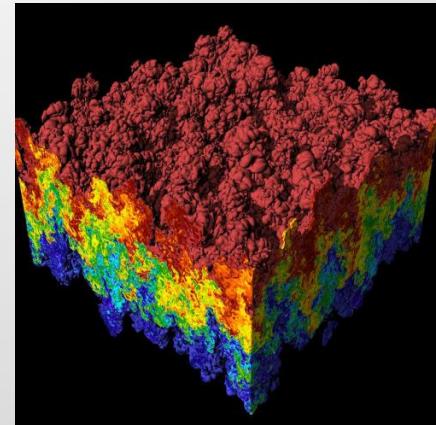
# Examples of VisIt's visualization capabilities.



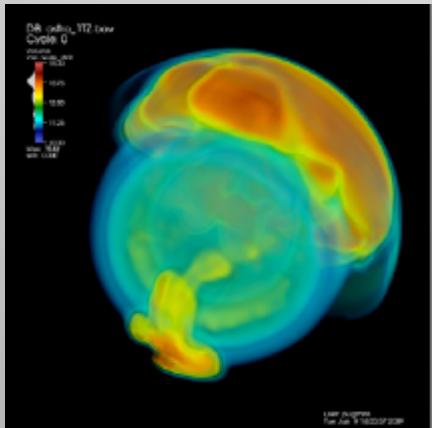
Streamlines



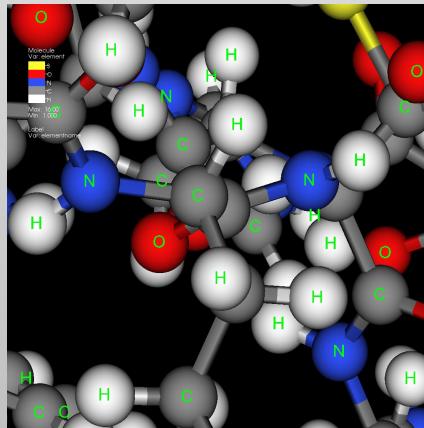
Vector / Tensor Glyphs



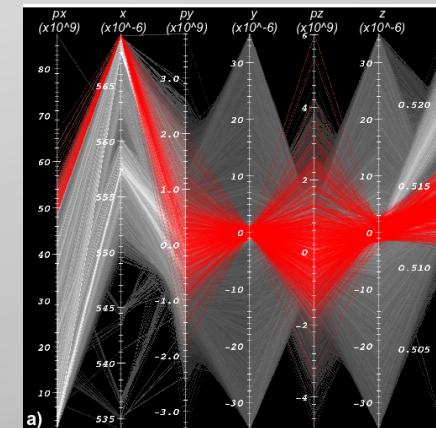
Pseudocolor Rendering



Volume Rendering



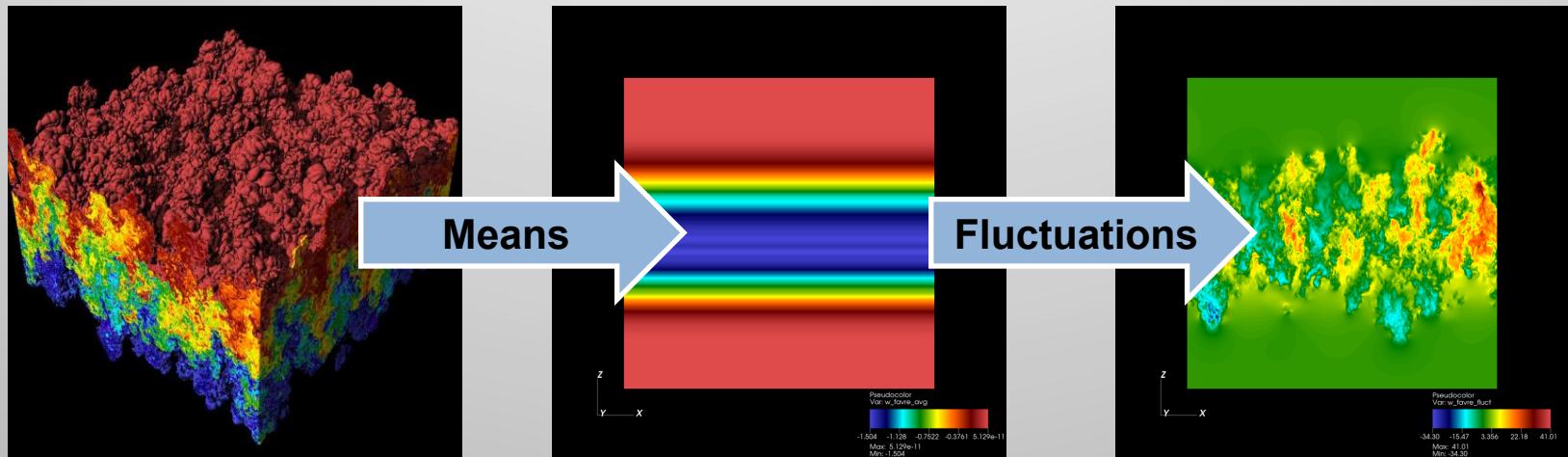
Molecular Visualization



Parallel Coordinates

# Analysis Example: Turbulence Operators

- **Goal:** Provide a one set of turbulence tools that can be used across multiple codes.
- **Application:** Validate RANS model parameters from high fidelity DNS simulations.

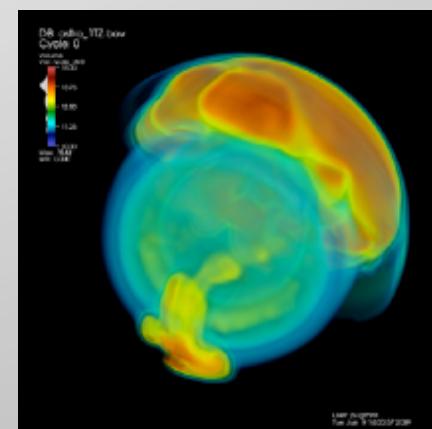
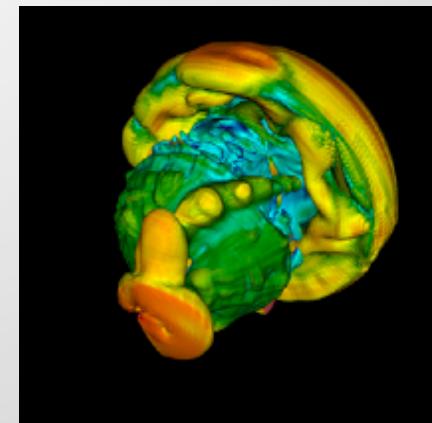


Joint work with Oleg Schilling, LLNL

We are developing scripted building blocks for flow analysis,  
including field means and fluctuations.

**Visit scales well on current HPC platforms.**

Machine	Architecture	Problem Size	# of Cores
<i>Graph</i>	X86_64	<b>20,001<sup>3</sup> (8 Tz)</b>	12K
Dawn	BG/P	15,871 <sup>3</sup> (4 Tz)	64K
Franklin	Cray XT4	12,596 <sup>3</sup> (2 Tz)	32K
JaguarPF	Cray XT5	12,596 <sup>3</sup> (2 Tz)	32K
Juno	X86_64	10,000 <sup>3</sup> (1Tz)	16K
Franklin	Cray XT4	10,000 <sup>3</sup> (1Tz)	16K
Ranger	Sun	10,000 <sup>3</sup> (1Tz)	16K
Purple	IBM P5	8,000 <sup>3</sup> (0.5 Tz)	8K

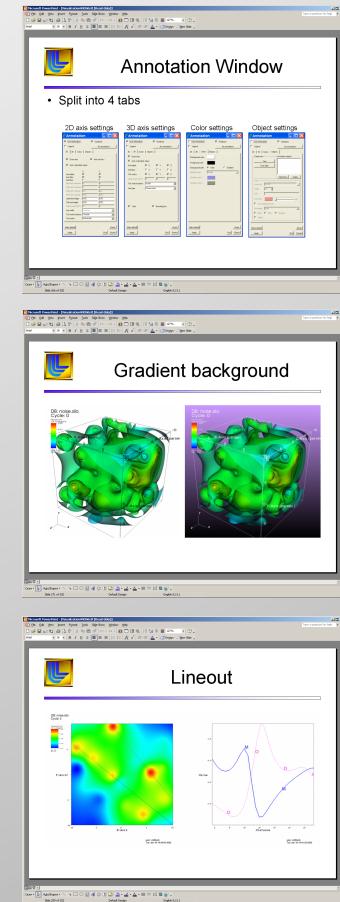


# Scaling Studies of Isosurface Extraction and Volume Rendering (2009)

VisIt is also used daily by domain scientists.

# The VisIt team focuses on making a robust, usable product for end users.

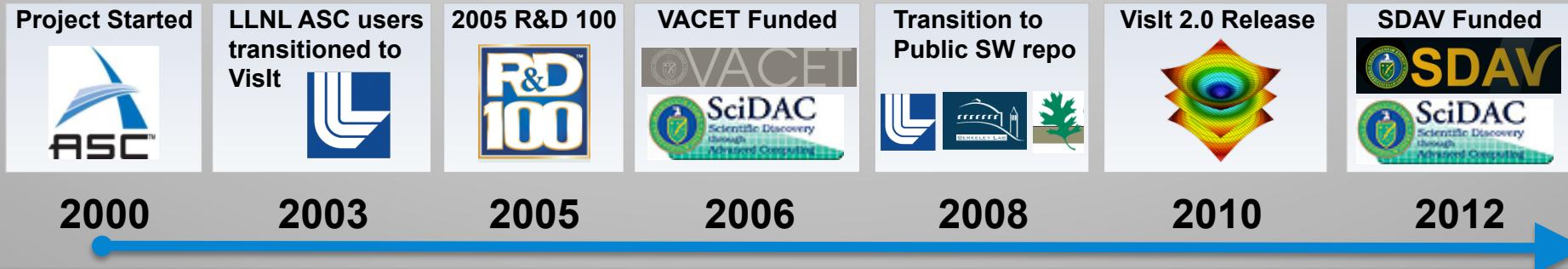
- Regular releases (~ 6 / year)
  - Executables for all major platforms
  - End-to-end build process script ``build\_visit''
- Customer Support and Training
  - visitusers.org, wiki for users and developers
  - Email lists: visit-users, visit-developers
  - Beginner and advanced tutorials
  - VisIt class with detailed exercises
- Documentation
  - “Getting data into VisIt” manual
  - Python interface manual
  - Users reference manual



*Slides from the VisIt class*

# VisIt is a vibrant project with many participants.

- The VisIt project started in 2000 to support LLNL's large scale ASC physics codes.
- The project grew beyond LLNL and ASC with research and development from DOE SciDAC and other efforts.
- VisIt is now supported by multiple organizations:
  - LLNL, LBNL, ORNL, UC Davis, Univ of Utah, ...
- Over 75 person years effort, 1.5+ million lines of code.



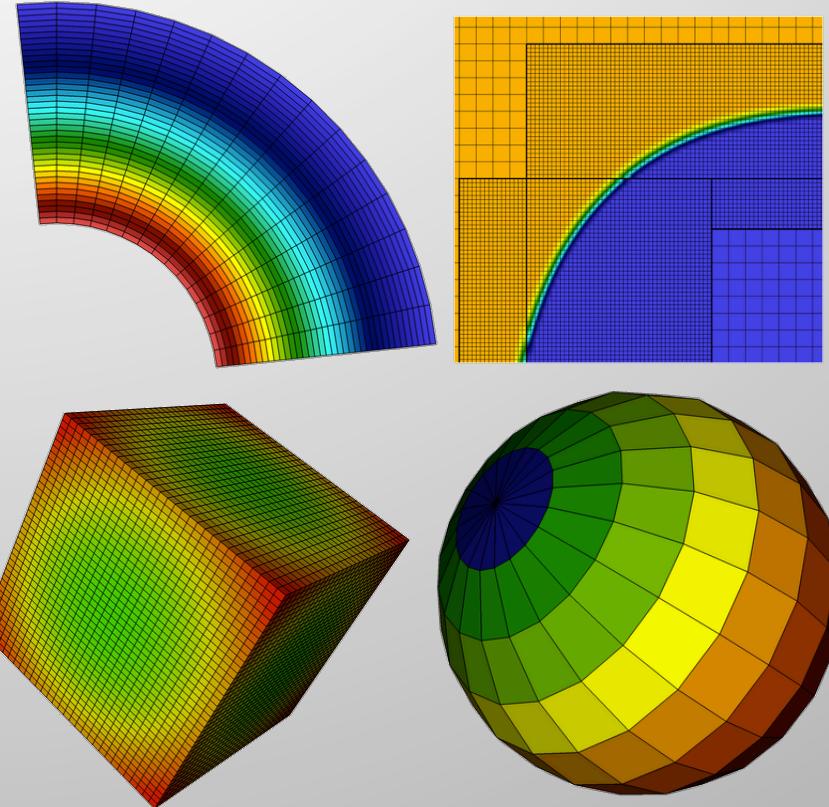
# VisIt provides a flexible data model, suitable for many application domains.

- **Mesh Types:**

- Point, Curve, 2D/3D  
Rectilinear, Curvilinear,  
Unstructured
- Domain Decomposed, AMR
- Time Varying

- **Fields:**

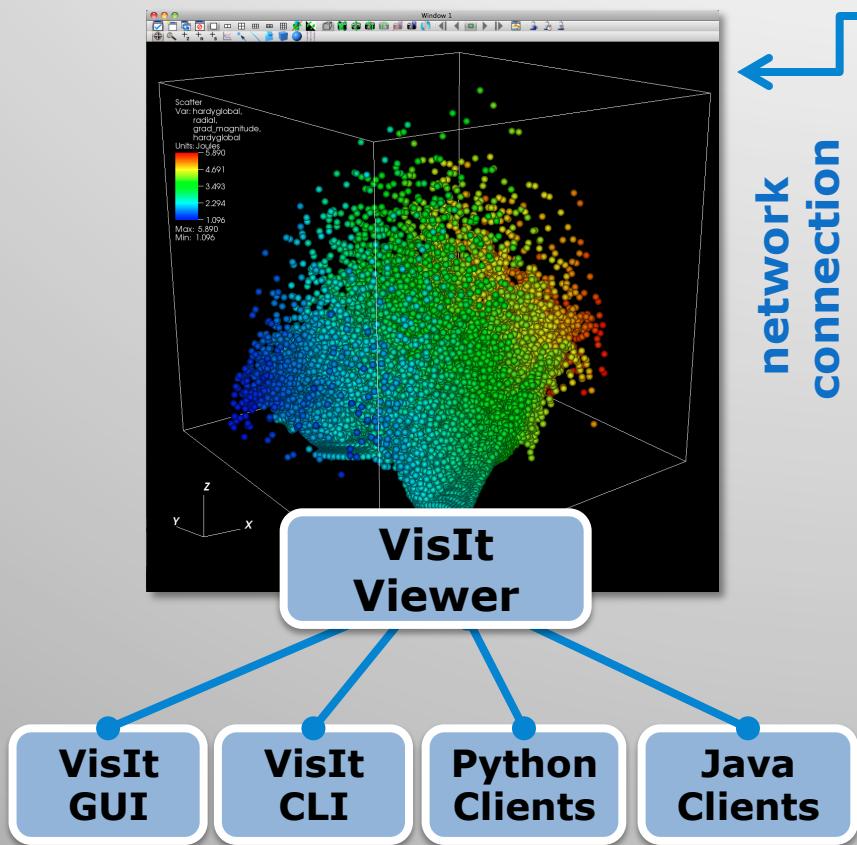
- Scalar, Vector, Tensor,  
Material volume fractions,  
Species



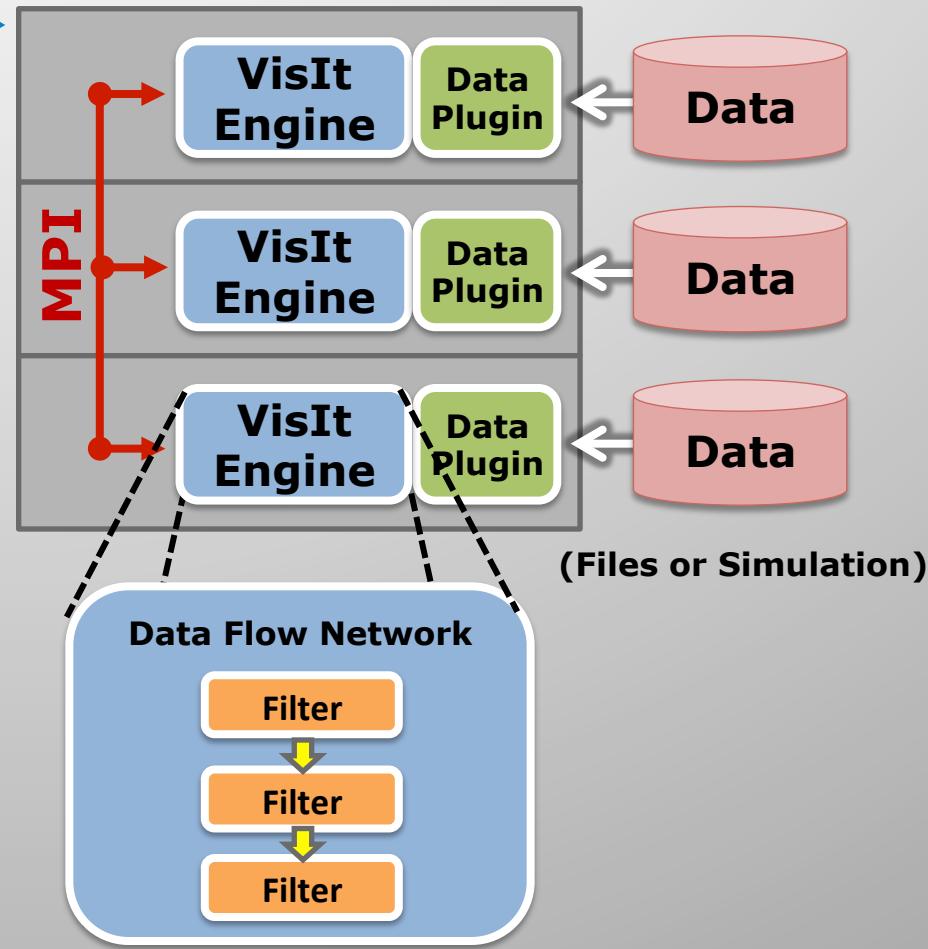
VisIt currently supports over 110 file formats.

# VisIt employs a parallelized client-server architecture.

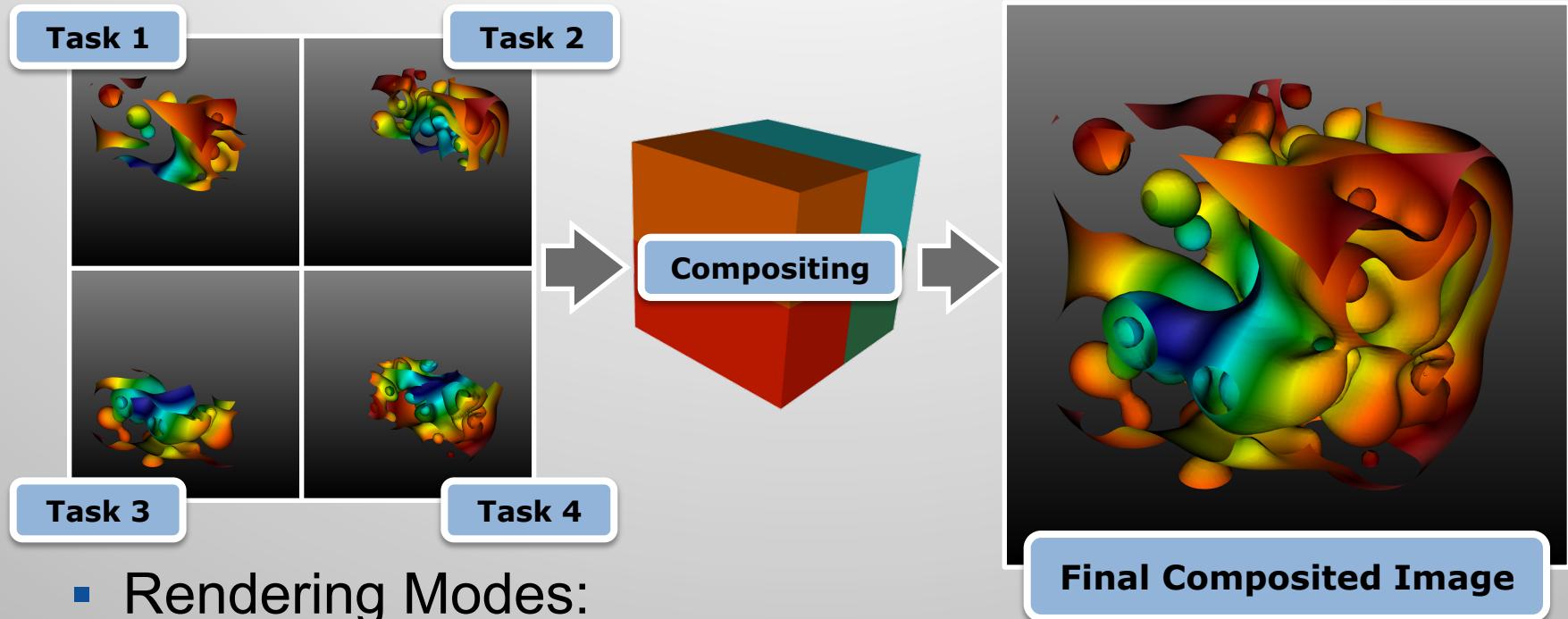
## Local Components



## Parallel Cluster



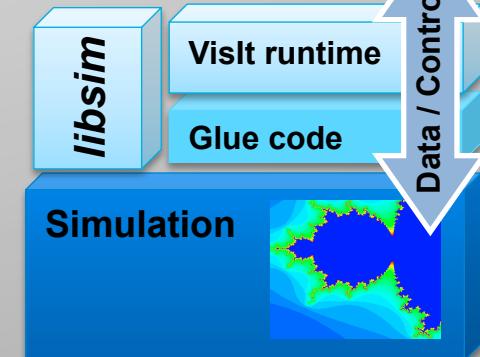
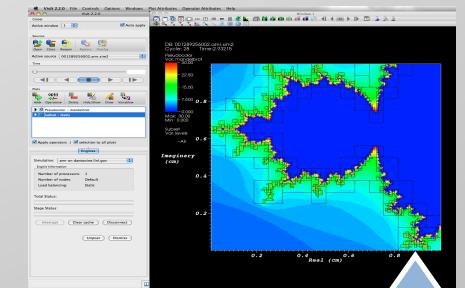
# VisIt automatically switches to a scalable rendering mode for large data sets.



- Rendering Modes:
  - Local (hardware)
  - Remote (software or hardware)
- Beyond surfaces:
  - VisIt also provides scalable volume rendering.

# VisIt's infrastructure provides a flexible platform for custom workflows.

- C++ Plugin Architecture
  - Custom File formats, Plots, Operators
  - Interface for custom GUIs in Python, C++ and Java
- Python Interfaces
  - Python scripting and batch processing
  - Data analysis via Python Expressions and Queries.
- “libsim” library enables coupling of simulation codes to VisIt for in situ visualization.



# VisIt is used as a platform to deploy visualization research.

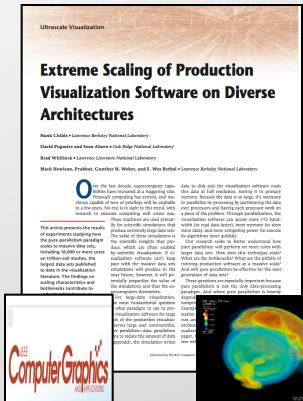
## ■ Research Collaborations:



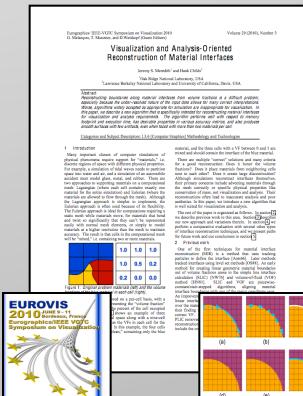
## UT/TACC Subcontract

## ■ Research Focus:

- Next Generation Architectures
- Parallel Algorithms



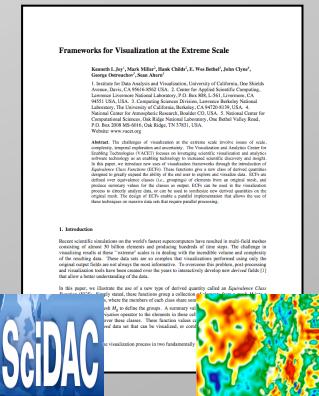
**Scaling research:**  
Scaling to 10Ks of cores  
and trillions of cells.



**Algorithms research:**  
Reconstructing material  
interfaces for visualization



**Algorithms research:**  
How to efficiently calculate  
particle paths in parallel.



**Methods research:**  
How to incorporate  
statistics into visualization.

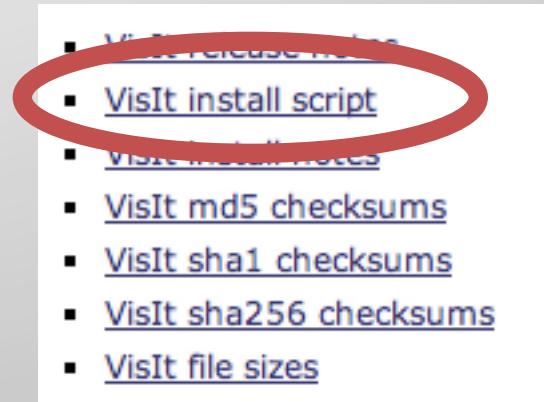
# VisIt Basics

- [http://visitusers.org/index.php?title=Short\\_Tutorial](http://visitusers.org/index.php?title=Short_Tutorial)
  - VisIt Basics
  - Data Analysis

# Visit Installation Help

# Using Release Binaries

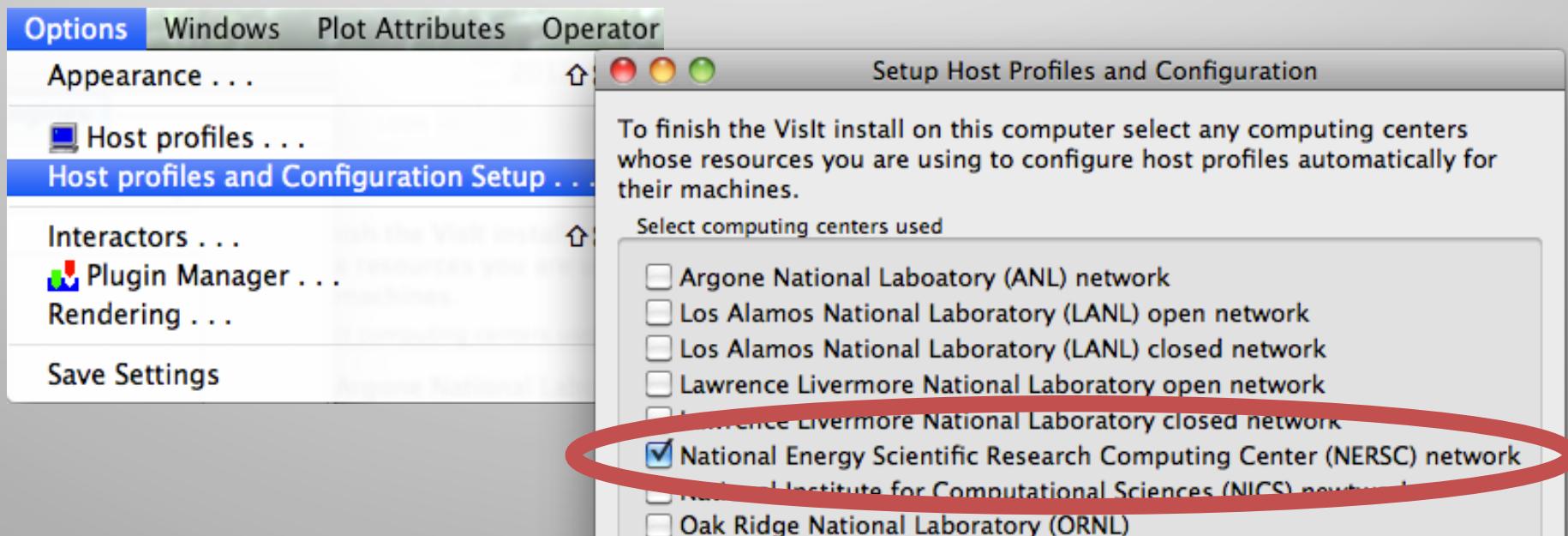
- <https://wci.llnl.gov/codes/visit/executables.html>
- Windows:
  - Run installer & select NERSC profiles.
- Linux:
  - Install using “visit-install”.



```
>./visit-install2_5_2 -c nersc \
  2.5.2 linux-rhel5 <dest_path>
```

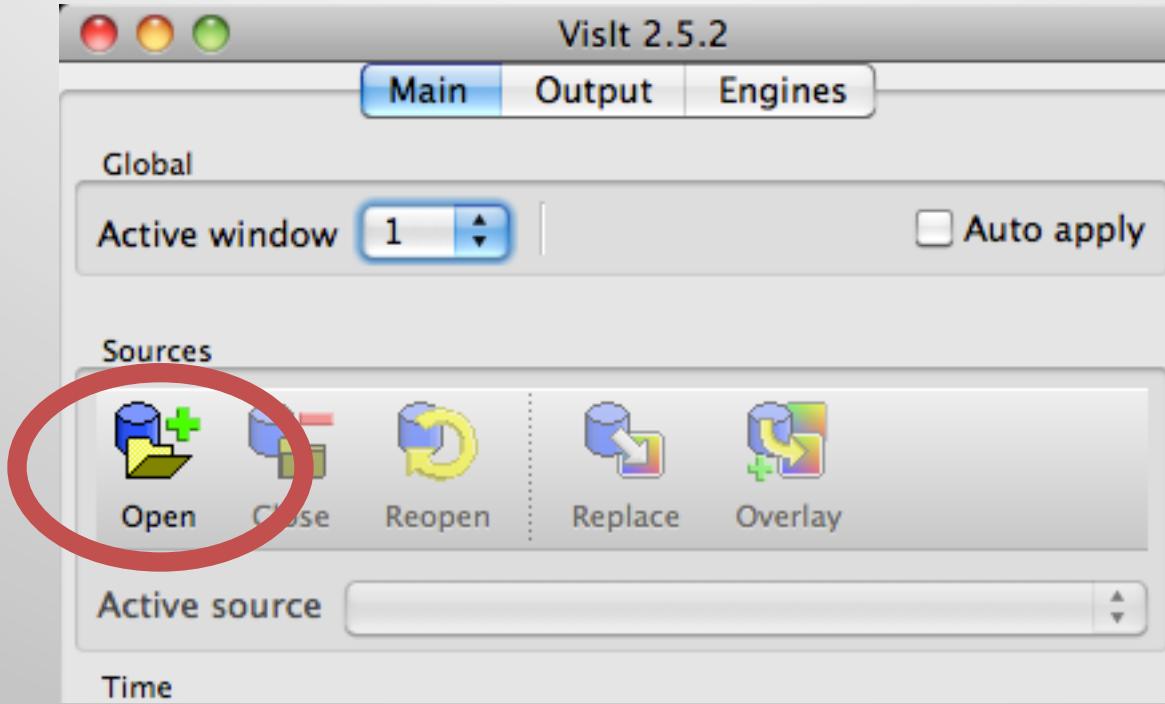
# Using Release Binaries

- <https://wci.llnl.gov/codes/visit/executables.html>
- OSX:
  - Open DMG & drag VisIt to desktop.
  - Select NERSC profiles:

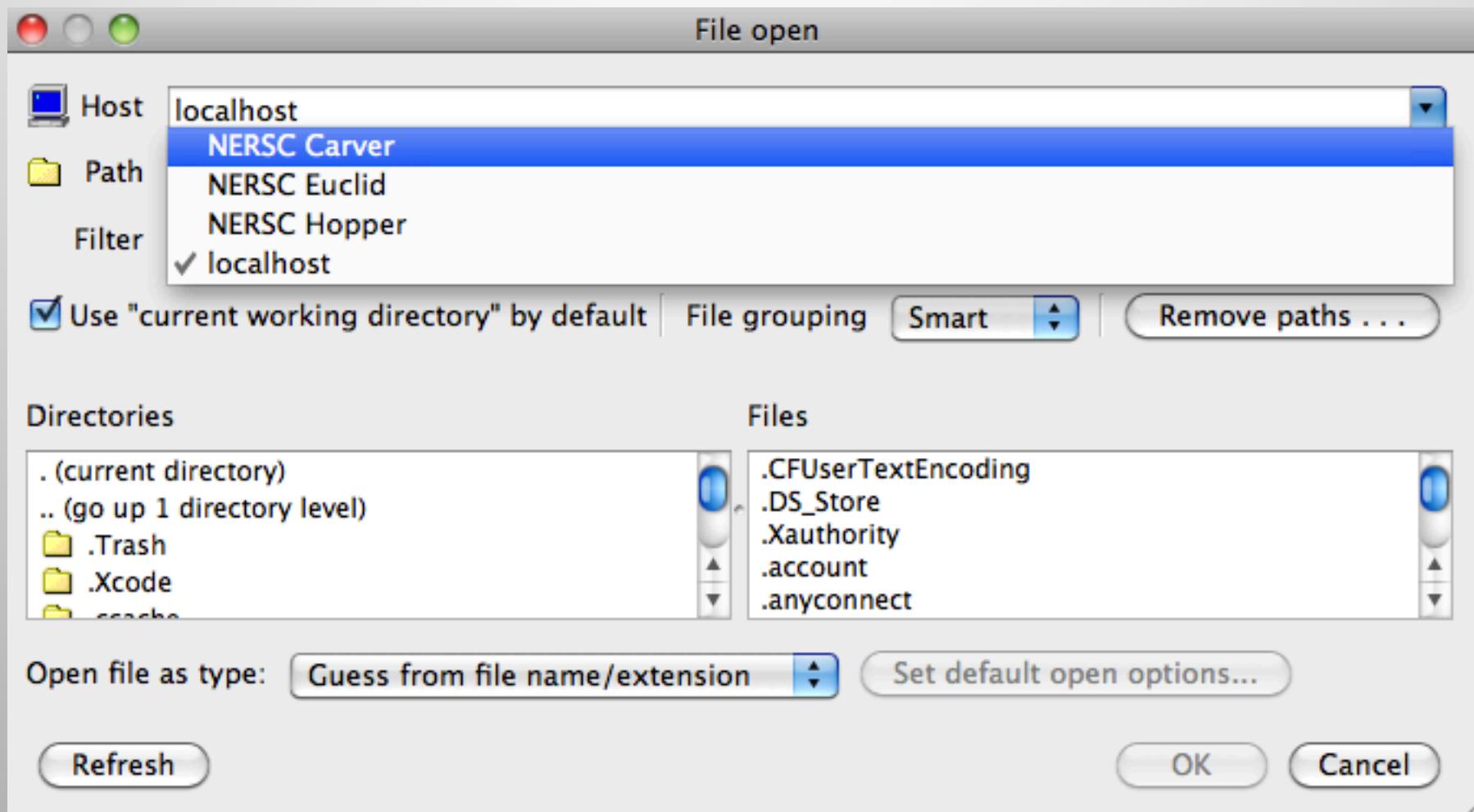


# Visit Client / Server Demo

# Connecting to NERSC's Carver Cluster



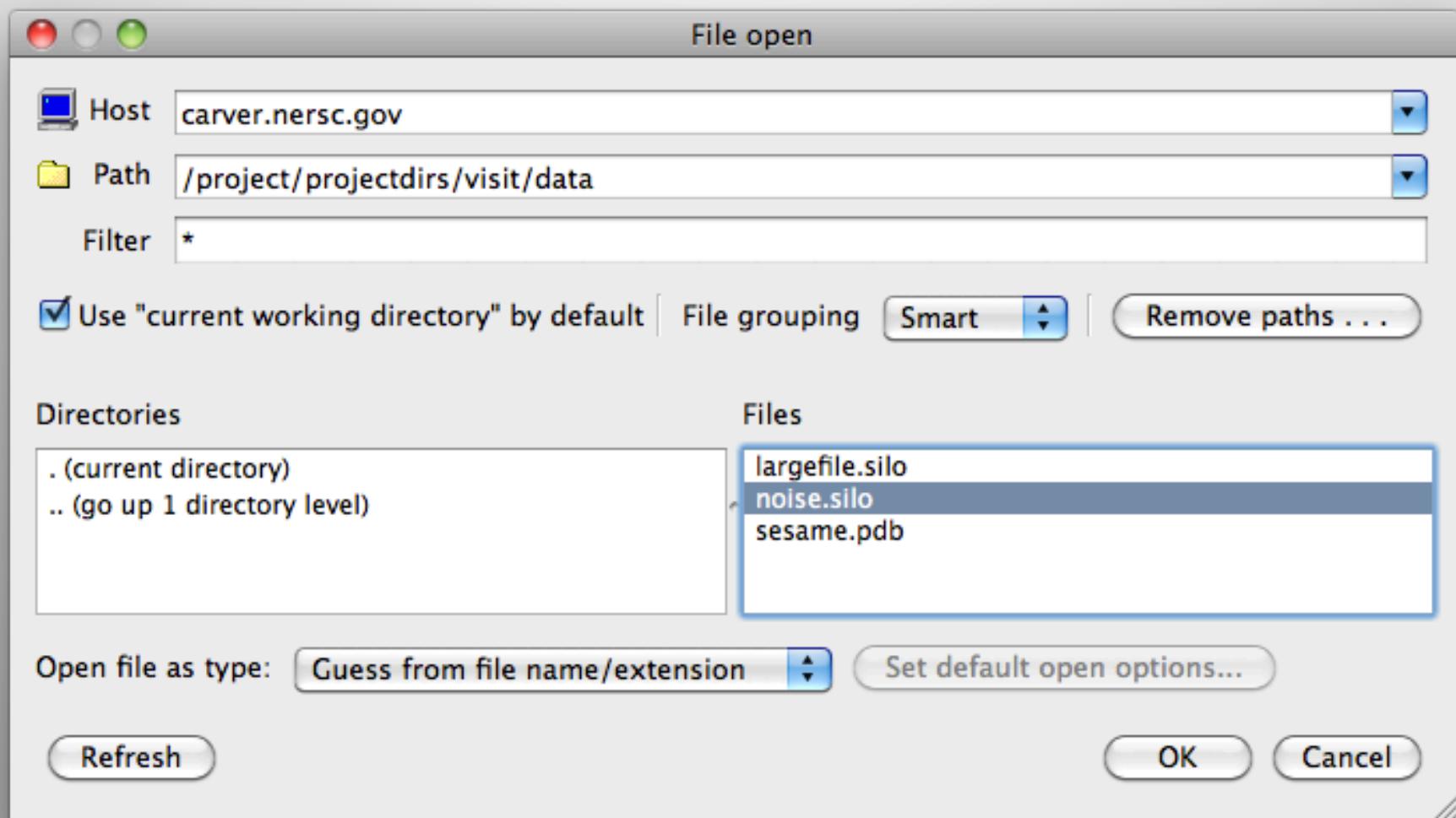
# Connecting to NERSC's Carver Cluster



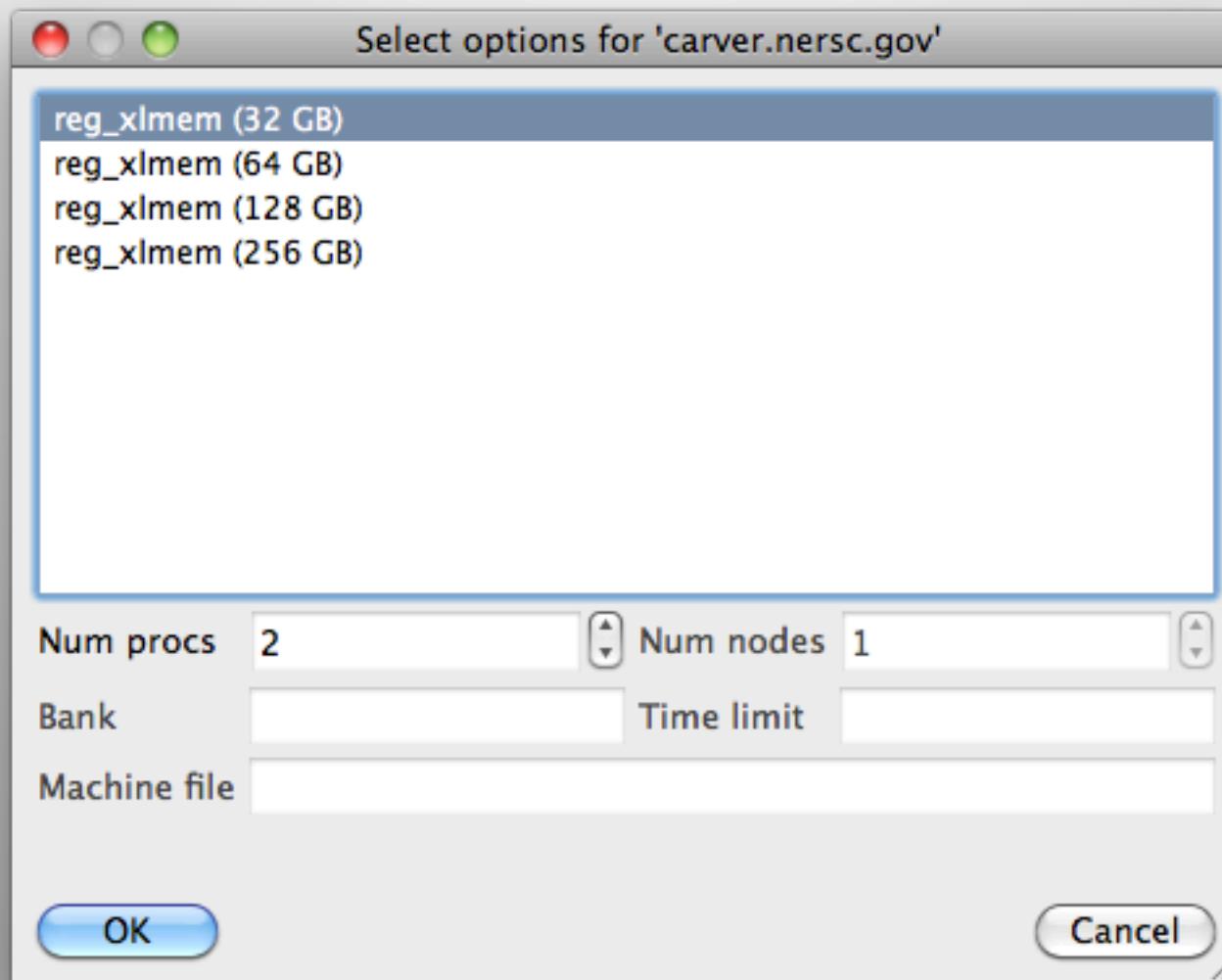
# Connecting to NERSC's Carver Cluster



# Connecting to NERSC's Carver Cluster



# Connecting to NERSC's Carver Cluster

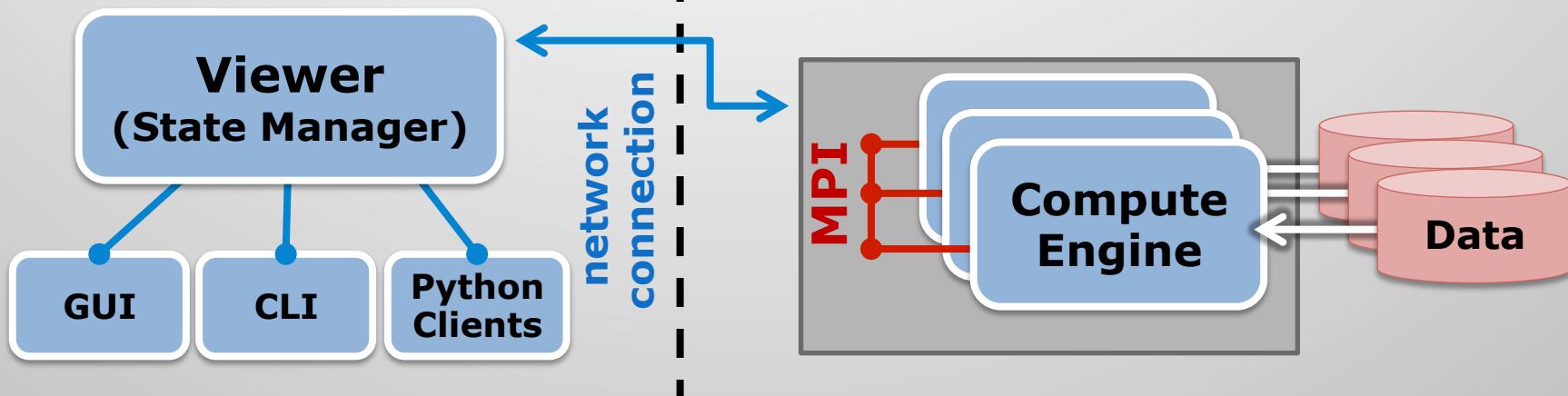


# Scripting VisIt

- [http://visitusers.org/index.php?title=Short\\_Tutorial](http://visitusers.org/index.php?title=Short_Tutorial)
  - Scripting

# VisIt provides Python interfaces for state control and data manipulation.

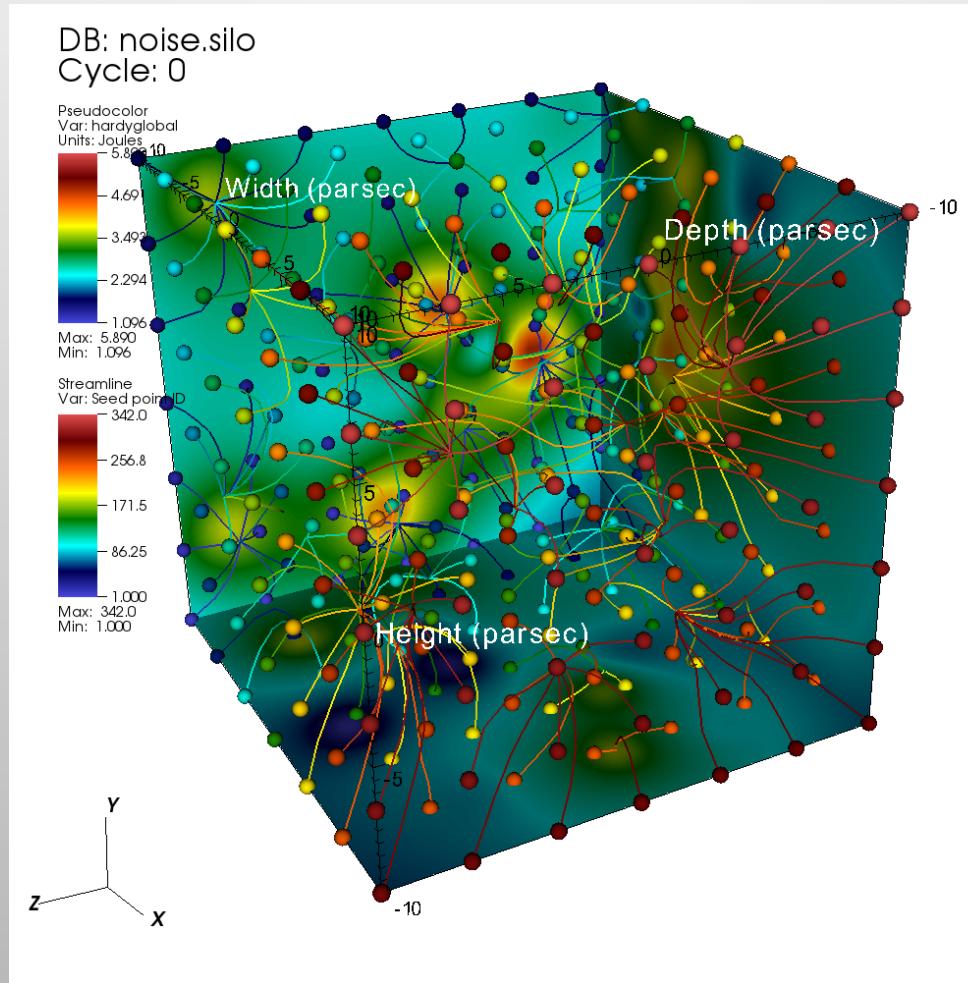
## Local Components | Parallel Cluster



**Python Client Interface  
(State Control)**

**Python Filter Runtime  
(Direct Mesh Manipulation)**

# Python Client Interface Example Script: Using VisIt's Building Blocks



# There are several ways to access VisIt's Python Client Interface.

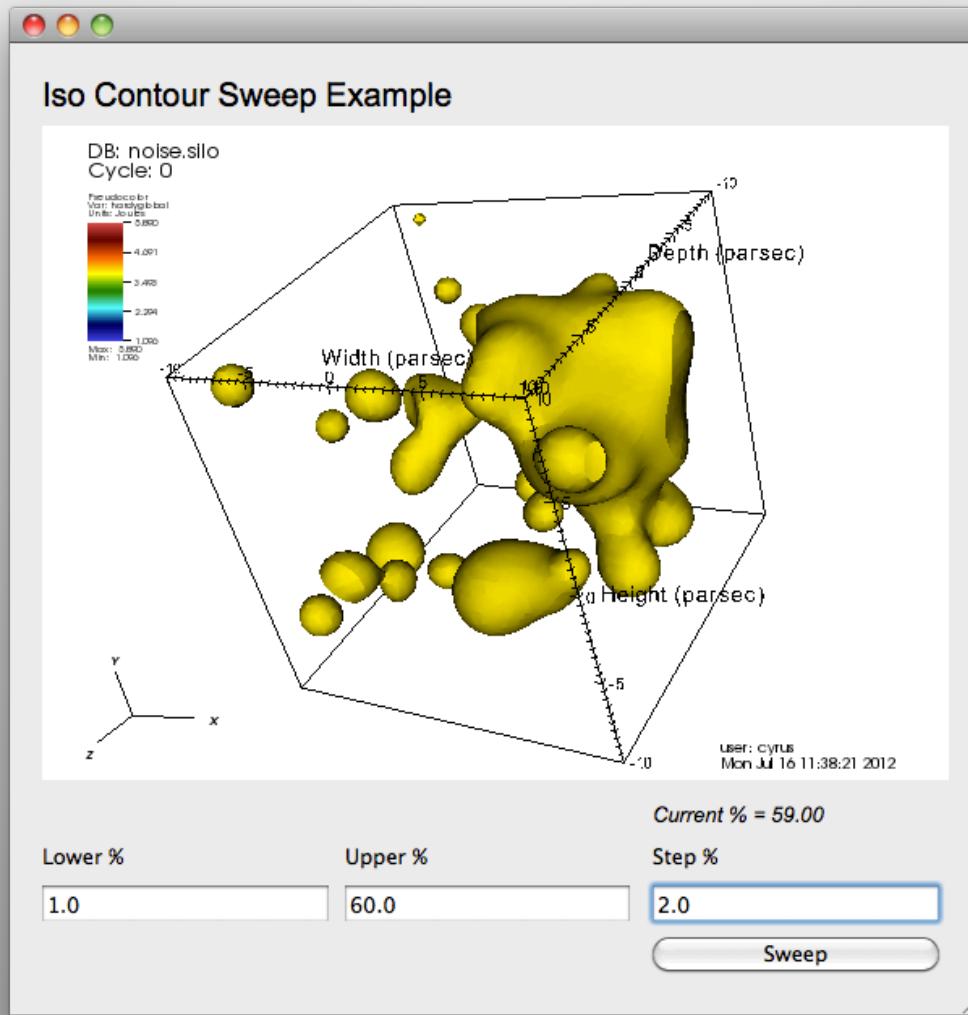
- Launch VisIt's CLI binary:
  - `>visit -cli`
- Launch for windowless batch processing:
  - `>visit -nowin -cli -s <script_file.py>`
- Control VisIt from a Python interpreter:
  - `'import visit'`
  - [http://visitusers.org/index.php?title=Python\\_Module\\_Support](http://visitusers.org/index.php?title=Python_Module_Support)
- Record GUI actions in to Python snippets:
  - Macro Recording provides a quick path to learn VisIt's Python API.

# We extended VisIt's core C++ infrastructure to support Python custom UIs.

- We chose PySide, a LGPL Python Qt wrapper, as the primary UI framework.
- Current support includes the ability to:
  - *Embed VisIt's render windows.*
  - *Reuse VisIt's existing GUI widgets.*
  - *Design UIs via Qt Designer.*



# Custom Python UI Example Script: Isosurface Sweep



# Wrap-up & Tips

***“How to make VisIt work after you get home.”***

***“How to make VisIt work after you get home.”***

- How to build VisIt from scratch.
- How to get VisIt to read your data.
- How to get help when you run into trouble.

# How to build VisIt from scratch.

- Manually building from scratch is difficult,
  - ... but the ``build\_visit'' script is fairly reliable.
  - <https://wci.llnl.gov/codes/visit/source.htm>

## Automatically build VisIt with the *build\_visit* script!

[Download build](#) [visit script here.](#)

VisIt can now be built automatically using the [build\\_visit](#) script on many Linux, MacOS X, and AIX platforms (*more to come*). The [build\\_visit](#) script takes care of downloading relevant VisIt and 3rd party source code, configuring, and building it all using your C++ compiler. We encourage users to build VisIt using the [build\\_visit](#) script when our binary distributions have trouble running on some systems. We also recommend using the [build\\_visit](#) script on your system if you plan to:

- Modify the VisIt source code.
  - Run a parallel compute engine. Building a parallel version of VisIt on your system allows you to configure VisIt so it uses your MPI library, avoiding incompatibilities.
  - Create your own VisIt plugins. Building VisIt on your system ensures that it is built with the same C++ compiler that you will use to develop your plugin, minimizing the chance for runtime library incompatibilities.



*(build\_visit screen shot)*

# What ``build\_visit'' does:

- Downloads source code for required third party libraries.
- Apply patches to accommodate OS quirks.
- Builds the third party libraries.
- Creates a `hostname.cmake' file, which tells VisIt's build system how to use these libraries.
- Downloads the VisIt source code.
- Builds VisIt.

# Running ``build\_visit'':

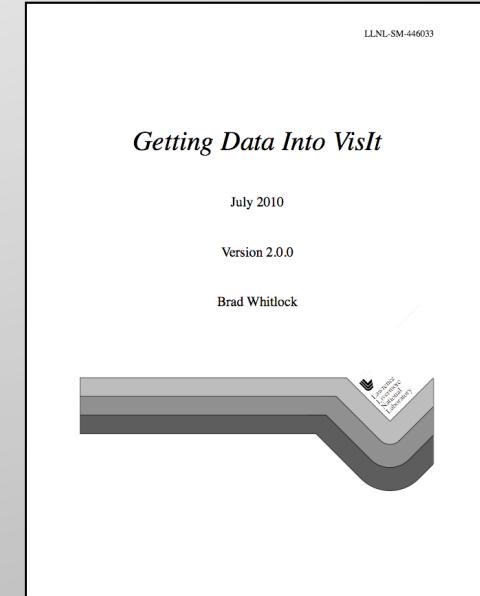
- There are two ways to use ``build\_visit'':
  - Pass command line options via --console
    - VisIt developers all use --console and it shows!
  - Curses-style GUI
- Tips:
  - Don't build every third party library unless you really need to.
  - Setup a `--thirdparty-path'

# How to get VisIt to read your data.

- There is an extensive manual on this topic: “Getting Data Into VisIt”

<https://wci.llnl.gov/codes/visit/manuals.html>

- Three ways:
  - Use a known format
  - Write a file format reader
  - In situ processing



# How to get VisIt to read your data.

- 110+ supported file formats:
  - ADIOS, BOV, Boxlib, CCM, CGNS, Chombo, CLAW, EnSight, ENZO, Exodus, FLASH, Fluent, GDAL, Gadget, Images (TIFF, PNG, etc), ITAPS/MOAB, LAMMPS, NASTRAN, NETCDF, Nek5000, OpenFOAM, PLOT3D, PlainText, Pixie, Shapefile, Silo, Tecplot, VTK, Xdmf, Vs, and many, many, more...

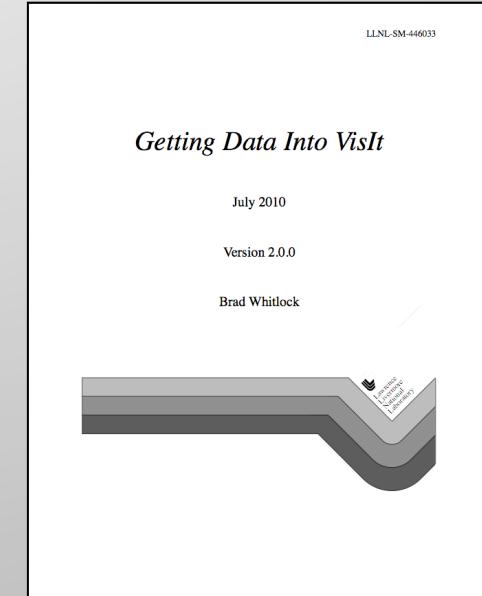
[http://www.visitusers.org/index.php?  
title=Detailed list of file formats VisIt supports](http://www.visitusers.org/index.php?title=Detailed%20list%20of%20file%20formats%20VisIt%20supports)

# How to get VisIt to read your data.

- There is an extensive manual on this topic: “Getting Data Into VisIt”

<https://wci.llnl.gov/codes/visit/manuals.html>

- Three ways:
  - Use a known format
  - *Write a file format reader*
  - *In situ processing*



These topics are covered in advanced VisIt tutorials.

# How to get help when you run into trouble.

- FAQ
  - <http://visit.llnl.gov/FAQ.html>
- VisIt Users Mailing List
  - Address: [visit-users@elist.ornl.gov](mailto:visit-users@elist.ornl.gov)
  - Info: <https://elist.ornl.gov/mailman/listinfo/visit-users>
  - Archive: <https://elist.ornl.gov/pipermail/visit-users/>
- VisIt Users Wiki
  - <http://www.visitusers.org>
- VisIt Users Forum
  - <http://visitusers.org/forum/YaBB.pl>
- Priority support for specific user groups:
  - VisIt-help-XYZ Mailing Lists
- Reference Manuals
  - <https://wci.llnl.gov/codes/visit/manuals.html>