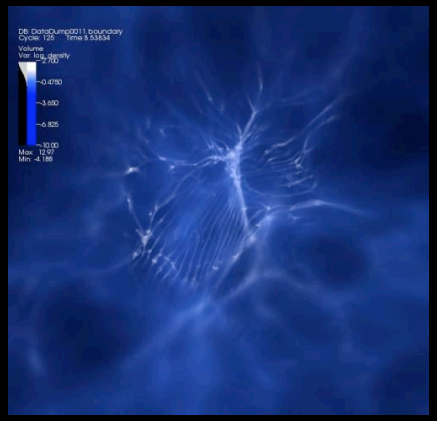
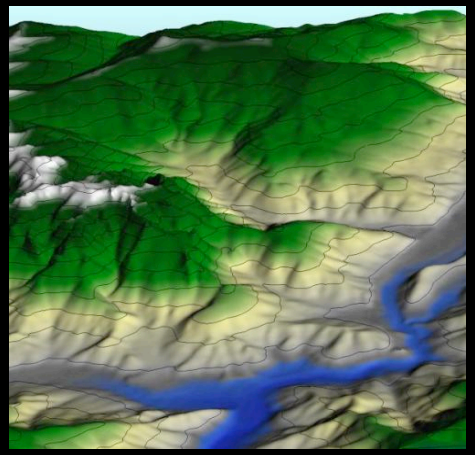
Visualization and Analysis with VisIt

Hank Childs, Cyrus Harrison, and Hari Krishnan

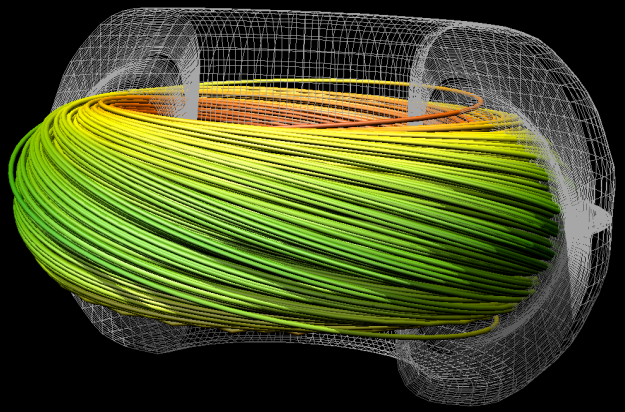
12th DOE ACTS Workshop

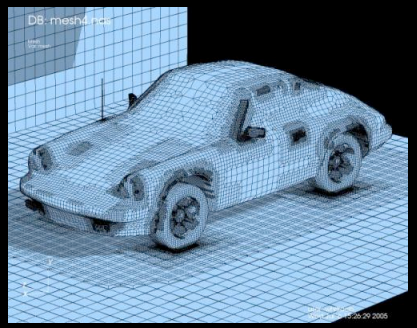
9:30-10:30 & 3:30-4:30, Aug 19th





- Basic usage
- Data analysis
- Derived quantities
- Scripting
- Moviemaking
- Comparisons
- + more!



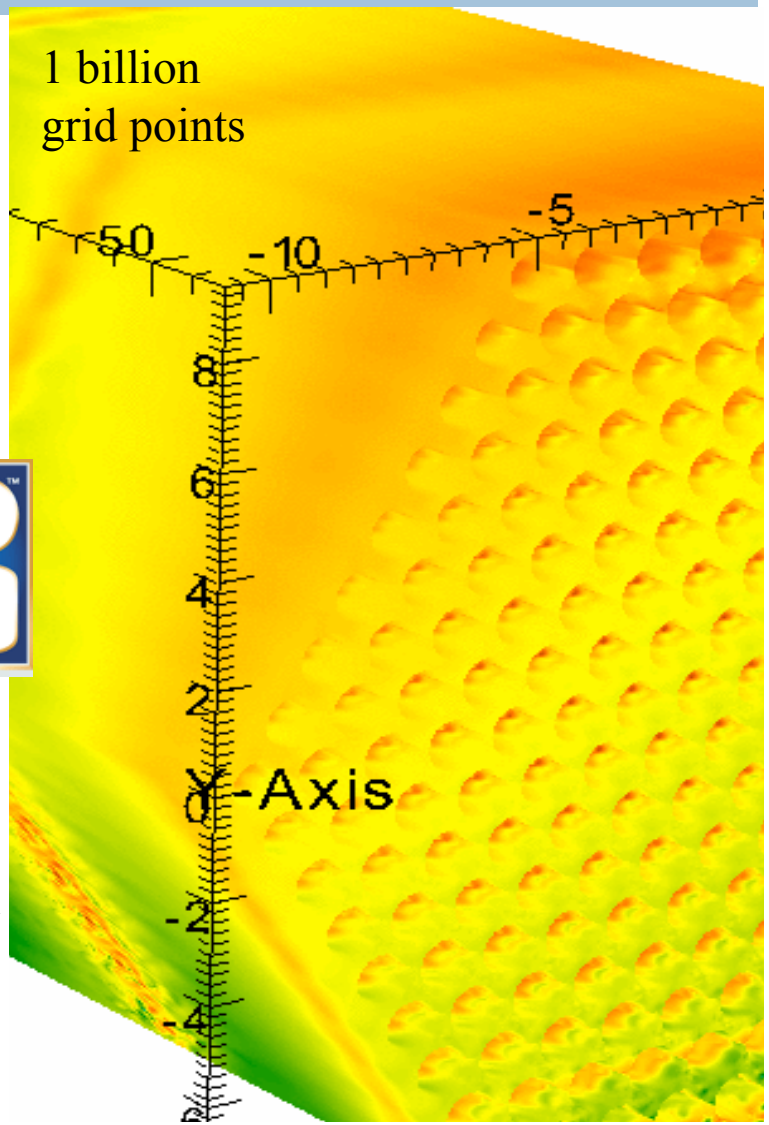


Tutorial Schedule

- 9:30-9:40: Vislt project overview (powerpoint)
- 9:40-10:20: Vislt basics: plots, operators, & more…
- 10:20-10:30: Mass installs
- 3:30-3:55: Scripting
- 3:55-4:15: Moviemaking
- 4:15-4:20: Group client-server example
- 4:20-4:30: How to Succeed With Vislt After This Tutorial (powerpoint)

VisIt is an open source, richly featured, turn-key application for large data.

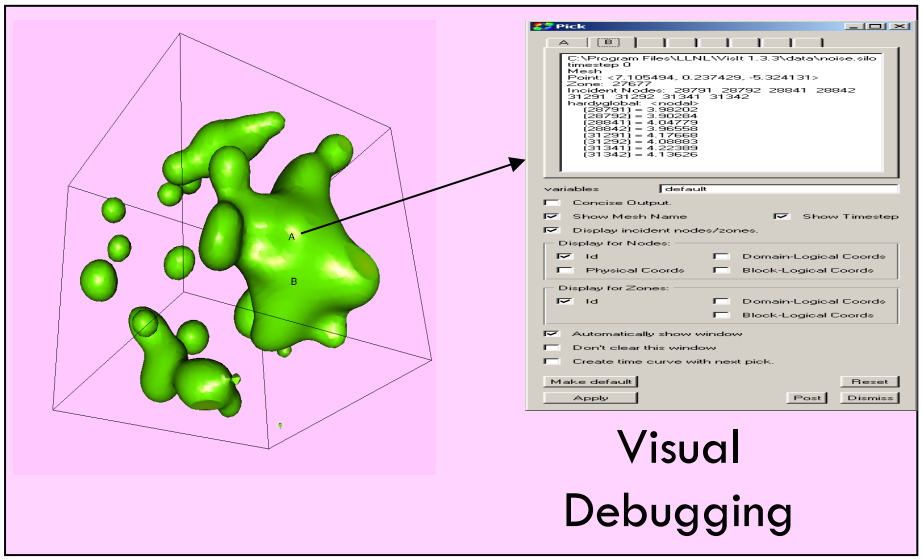
- Used by:
  - Visualization experts
  - Simulation code developers
  - Simulation code consumers
- Popular
  - R&D 100 award in 2005
  - Used on many of the Top500
  - >>>100K downloads



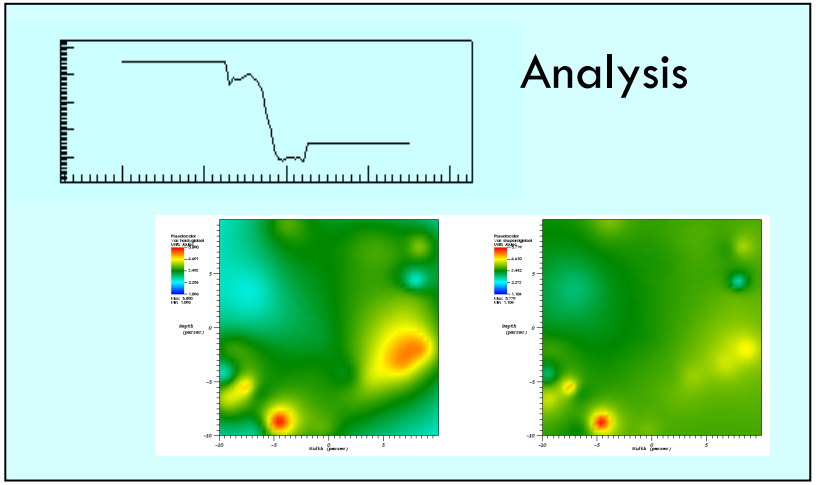
217 pin reactor cooling simulation
Run on ¼ of Argonne BG/P
Image credit: Paul Fischer, ANL

Terribly Named!!!

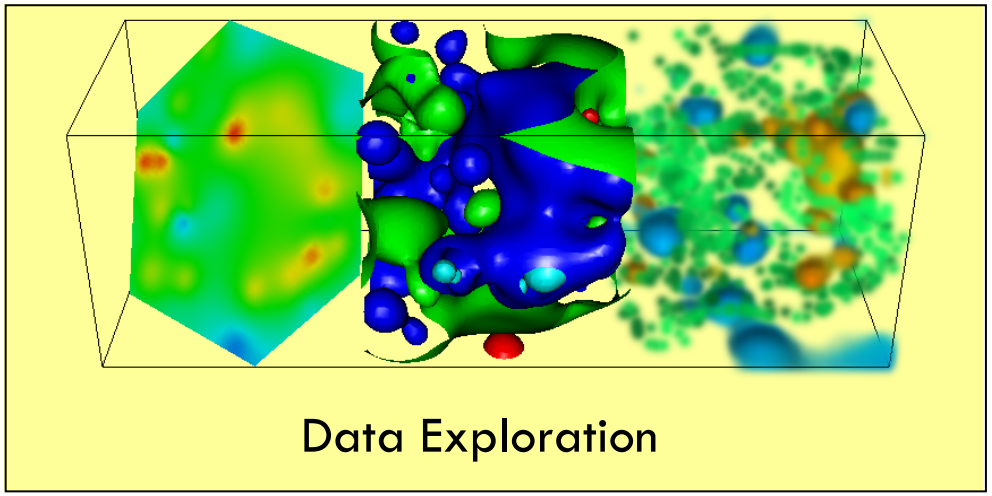
… intended for much more than just visualization



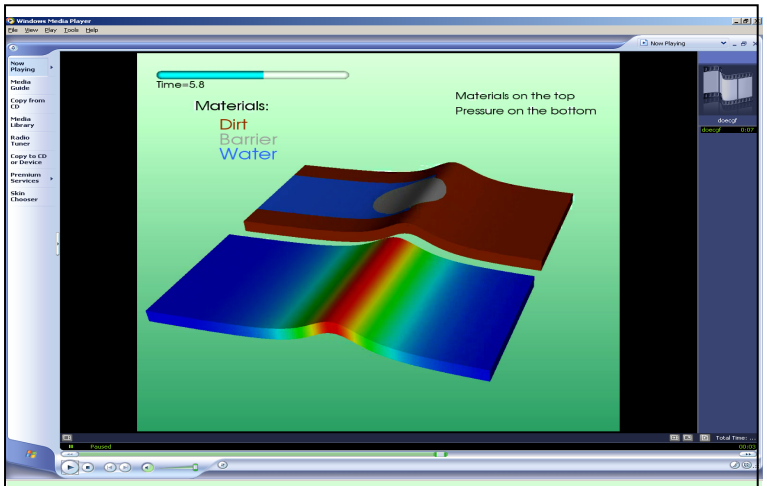
Visual Debugging



Analysis



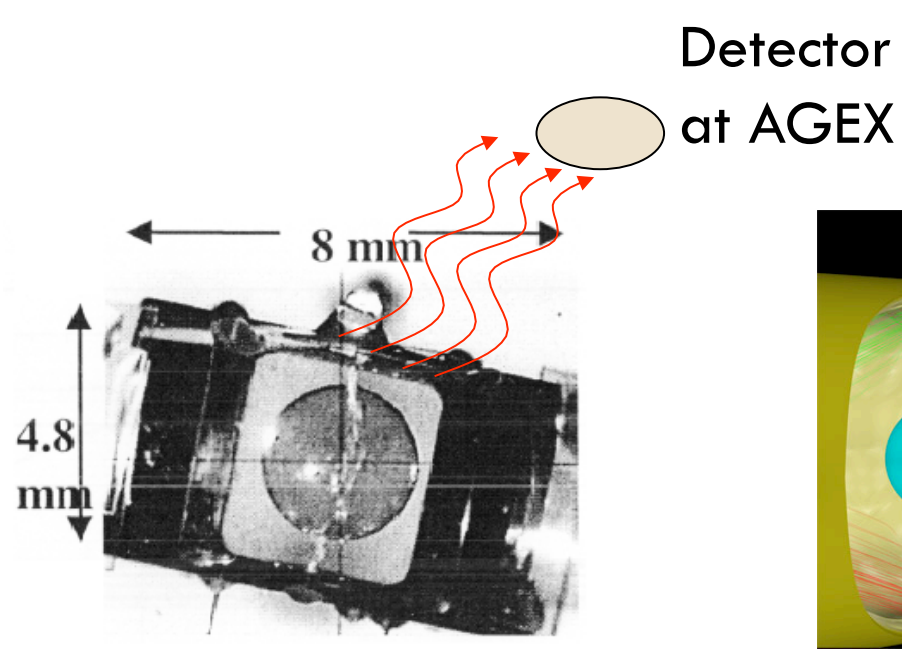
Data Exploration

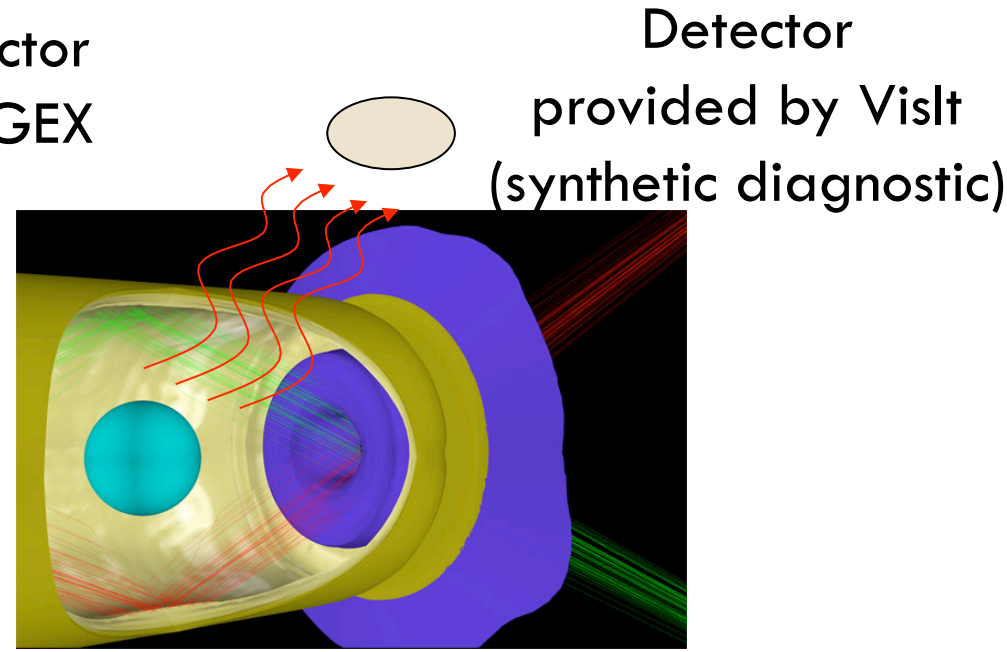


Presentations

What sort of analysis is appropriate for VisIt?

- General techniques (e.g. integration, volumes, surface areas, etc.)
- Specialized analysis (e.g. hohlraum flux at AGEX)

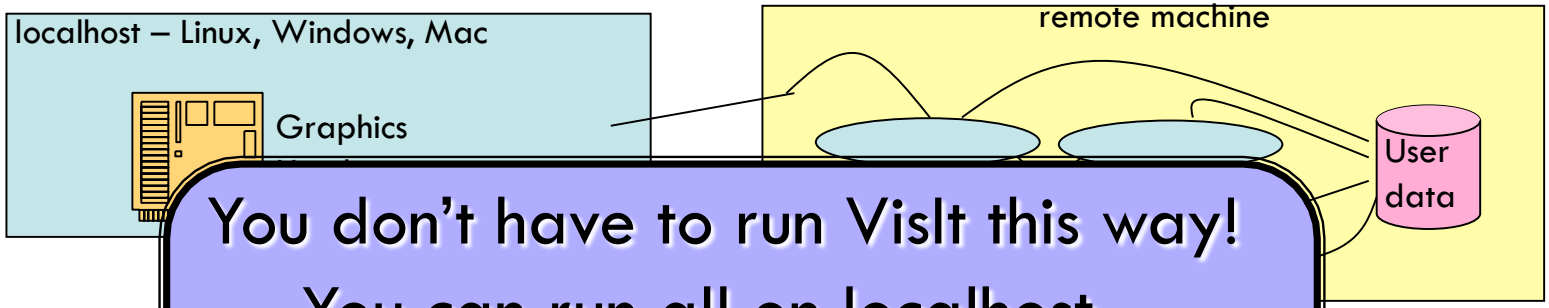




VisIt has a rich feature set.

- Meshes: rectilinear, curvilinear, unstructured, point, AMR
- Data: scalar, vector, tensor, material, species
- Dimension: 1D, 2D, 3D, time varying
- Rendering (~15): pseudocolor, volume rendering, hedgehogs, glyphs, mesh lines, etc…
- Data manipulation (~40): slicing, contouring, clipping, thresholding, restrict to box, reflect, project, revolve, …
- File formats (~110)
- Derived quantities: >100 interoperable building blocks
  - +,-,*,/, gradient, mesh quality, if-then-else, and, or, not
- Many general features: position lights, make movie, etc
- Queries (~50): ways to pull out quantitative information, debugging, comparative analysis

VisIt employs a parallelized client-server architecture.



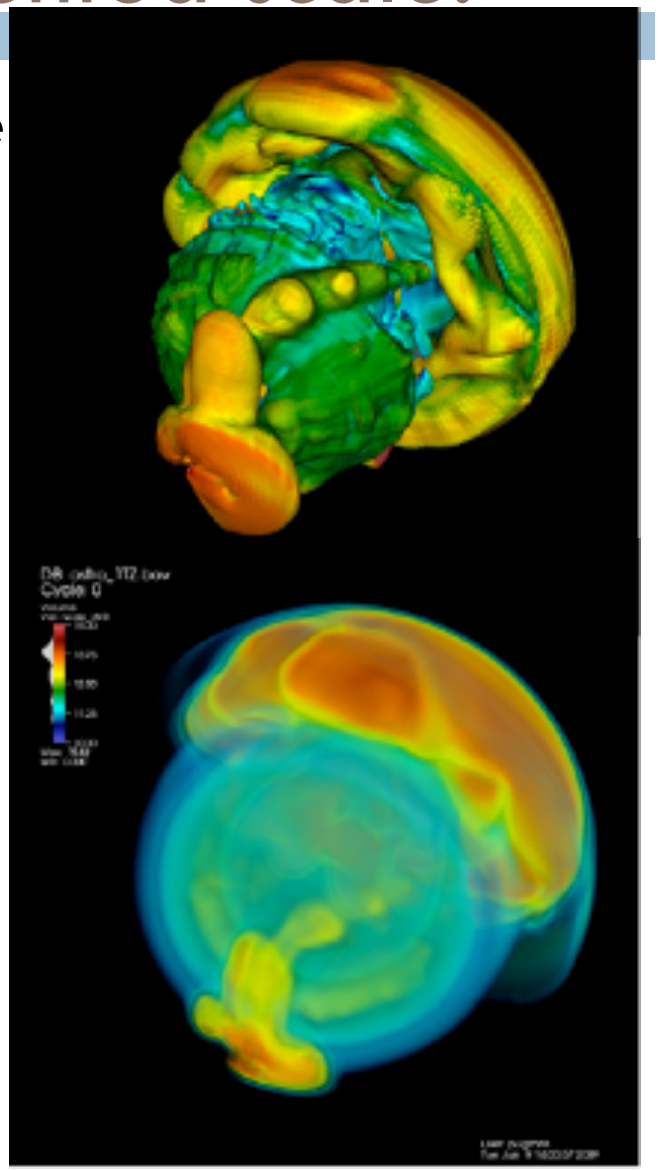
You don't have to run VisIt this way!
You can run all on localhost (like this tutorial!)
You can tunnel through ssh and run all on the remote machine

- Client-ser
observati
  - Good
visualization
  - Leverages available resources
  - Scales well
  - No need to move data
- Heavy use of VTK
- Multiple UIs: GUI (Qt), CLI (Python), more…

Vislt recently demonstrated good performance at unprecedented scale.

- Weak scaling study: ~62.5M cells/core

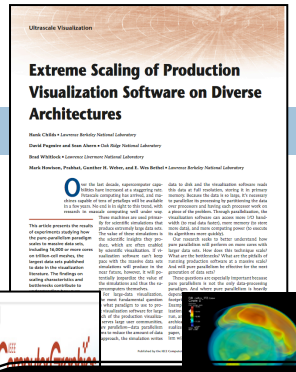
Machine	Model	Problem Size	#cores
Franklin	Cray XT4	1T, 2T	16K, 32K
Dawn	BG/P	4T	64K
JaguarPF	Cray XT5	2T	32K
Juno	X86_64	1T	16K
Purple	IBM P5	0.5T	8K
Ranger	Sun	1T	16K

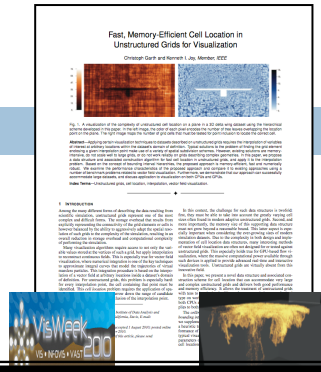


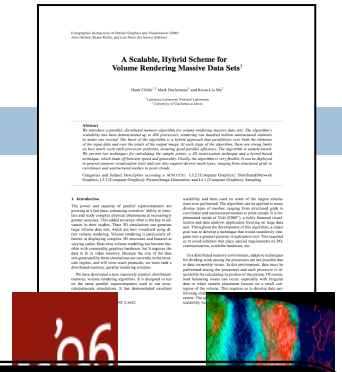
Two trillion cell data set, rendered in Vislt by David Pugmire on ORNL Jaguar machine

It has taken a lot of research to make VisIt work

Systems research:
Adaptively applying algorithms in a production env.



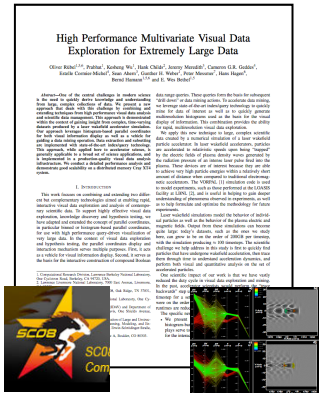




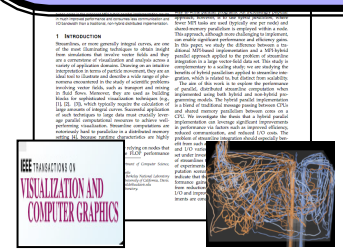
Algorithms research:
How to efficiently calculate particle paths in parallel.

Don't be scared! ...

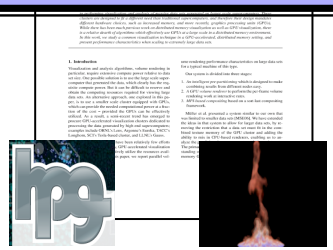
VisIt developers mostly wear their software engineering hats – not their research hats.



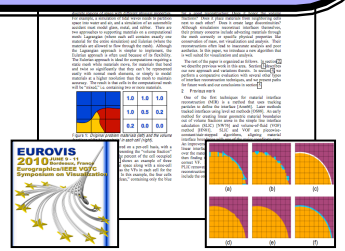
Systems research:
Using smart DB technology to accelerate processing



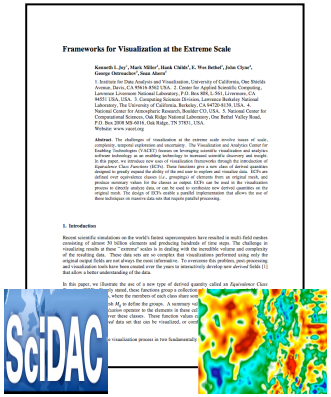
Architectural research:
Hybrid parallelism + particle advection



Architectural research:
Parallel GPU volume rendering



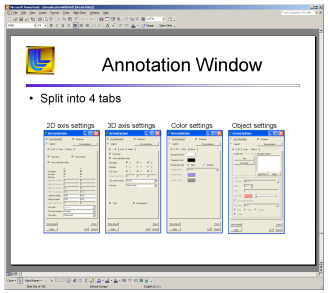
Algorithms research:
Reconstructing material interfaces for visualization

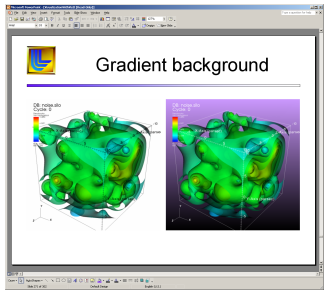


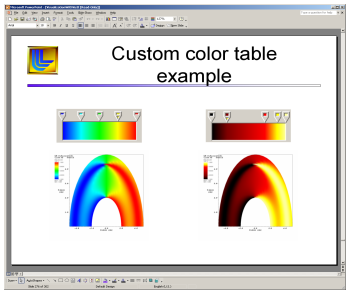
Methods research:
How to incorporate statistics into visualization.

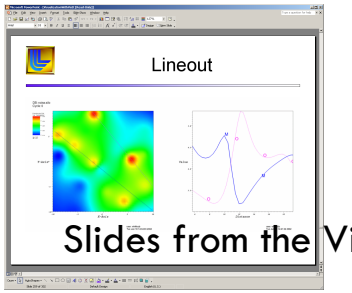
The VisIt team focuses on making a robust, usable product for end users.

- Manuals
  - 300 page user manual
  - 200 page command line interface manual
  - "Getting your data into VisIt" manual
- Wiki for users (and developers)
- Revision control, nightly regression testing, etc
- Executables for all major platforms
- Day long class, complete with exercises





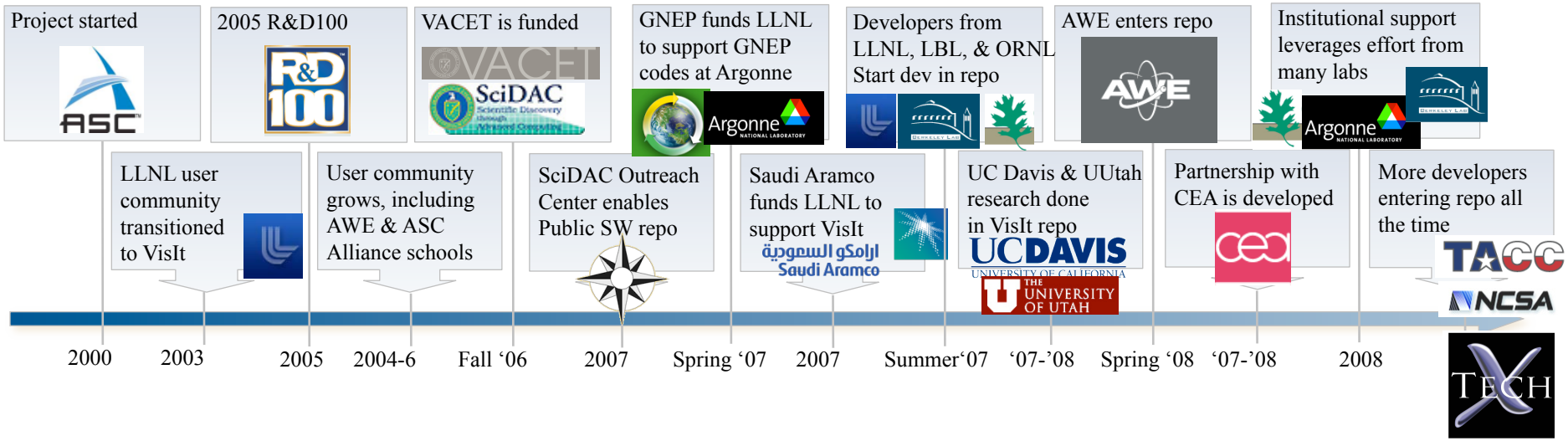




Slides from the VisIt class

VisIt is a vibrant project with many participants.

- Over 75 person-years of effort
- Over 1.5 million lines of code
- Partnership between: Department of Energy's Office of Science, National Nuclear Security Agency, and Office of Nuclear Energy, the National Science Foundation XD centers (Longhorn XD and RDAV), and more….



VisIt: What's the Big Deal?

- Everything works at scale
- Robust, usable tool
- Features that span the "power of visualization":
  - Data exploration
  - Confirmation
  - Communication
- Features for different kinds of users:
  - Vis experts
  - Code developers
  - Code consumers
- Healthy future: vibrant developer and user communities

Before we begin…

- Reminder: we will discuss file format issues, installation issues, and how to get help at the end of the tutorial
- Important: ask questions any time!

<demonstration>

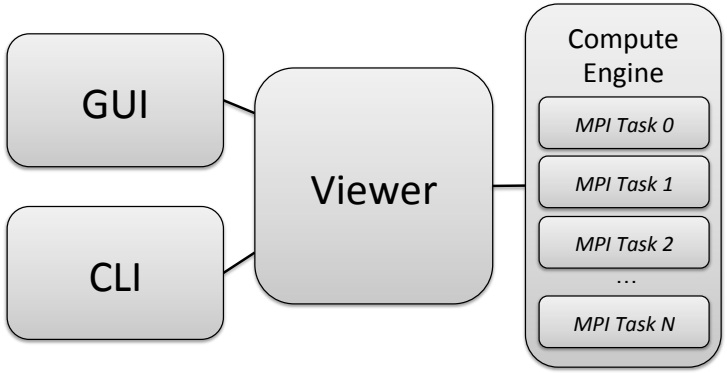
- Vislt basics
- Queries and expressions
- Scripting
- Moviemaking

VisIt Python Scripting

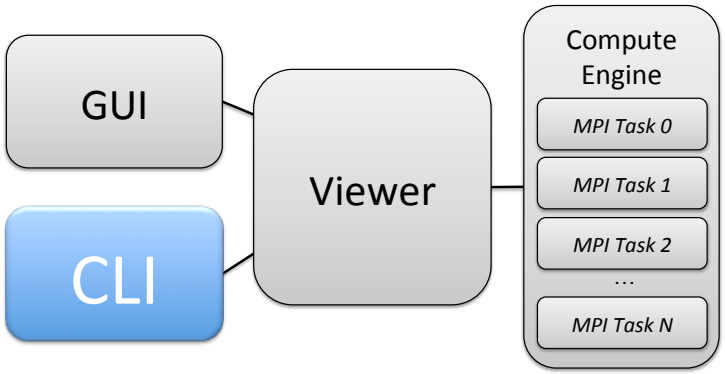
August 19, 2011

VisIt + Python

How does VisIt use Python?



VisIt + Python

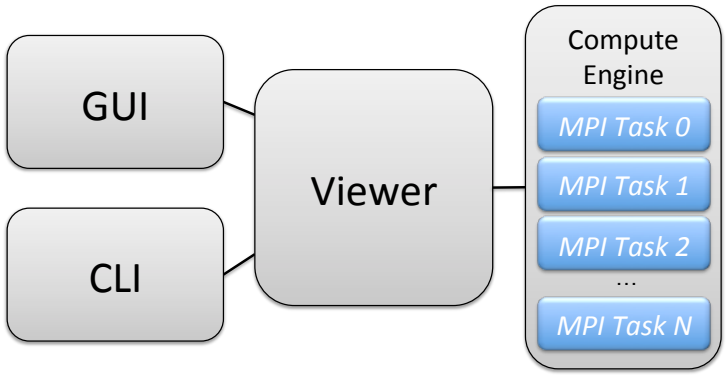


Python Client Interface

A python interface for *driving* and *scripting* VisIt, the heart of:

- VisIt's Command Line Interface (CLI)
- The VisIt Python Module

VisIt + Python

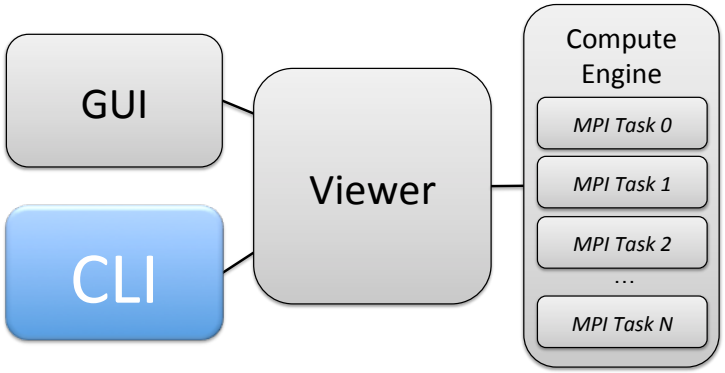


Python Filters

A python framework for creating new data processing filters in the compute engine.

- Enables custom *Expressions* & *Queries* written in python.

VisIt + Python



For this scripting tutorial we will focus on functionality provided by VisIt's Python Client Interface.

Getting Started

- Launch VisIt
- Open ‘noise.silo’
- From the *Controls* Menu, select the *Commands* Window
- Type the following:

```
AddPlot("Pseudocolor","hardyglobal")
DrawPlots()
```

- Click ‘Execute’

Getting Started

- Launch VisIt
- Open ‘noise.silo’
- From the *Controls* Menu, select the *Commands* Window
- Type the following:

```
AddPlot("Pseudocolor","hardyglobal")
DrawPlots()
```

- Click ‘Execute’

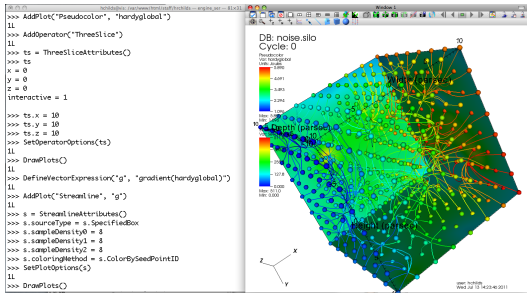
Tutorial Scripts

http://visitusers.org/index.php?title=
VisIt_Tutorial_Python_Simple_Scripts

Tips for Learning the Client Interface

Python Interface Manual

VisIt Python Interface Manual

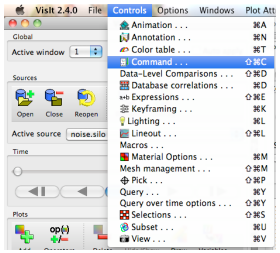


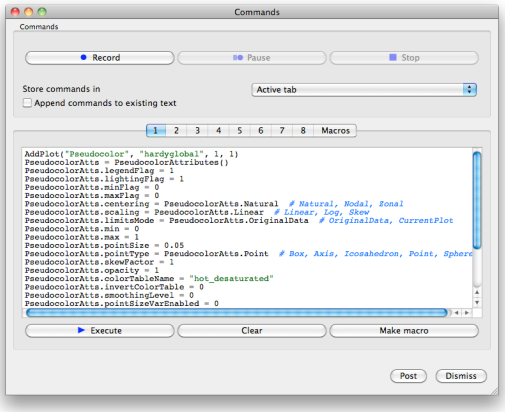
Version 2.3.0

http://www.visitusers.org/visit/2.3.0/VisItPythonManual.pdf

Tips for Learning the Client Interface

VisIt Commands Window

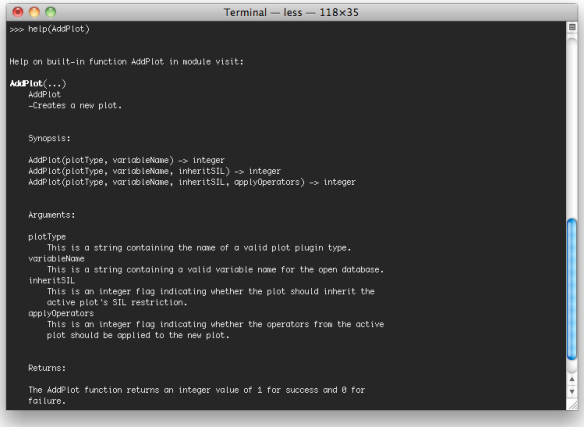




Record your GUI actions to capture equivalent python.

Tips for Learning the Client Interface

Interactive Python ‘help’



Use ‘help(<obj>)’ to display module documentation for ‘<obj>’.

“How to make VisIt work after you get home”

- How to get VisIt running on your machine
  - Downloading and installing VisIt
  - Building VisIt from scratch
- How to get VisIt to read your data
- How to get help when you run into trouble
- I like the power of VisIt, but I hate the interface
- How to run client-server

“How to make Vislt work after you get home”

- How to get Vislt running on your machine
  - Downloading and installing Vislt
  - Building Vislt from scratch
- How to get Vislt to read your data
- How to get help when you run into trouble
- I like the power of Vislt, but I hate the interface
- How to run client-server

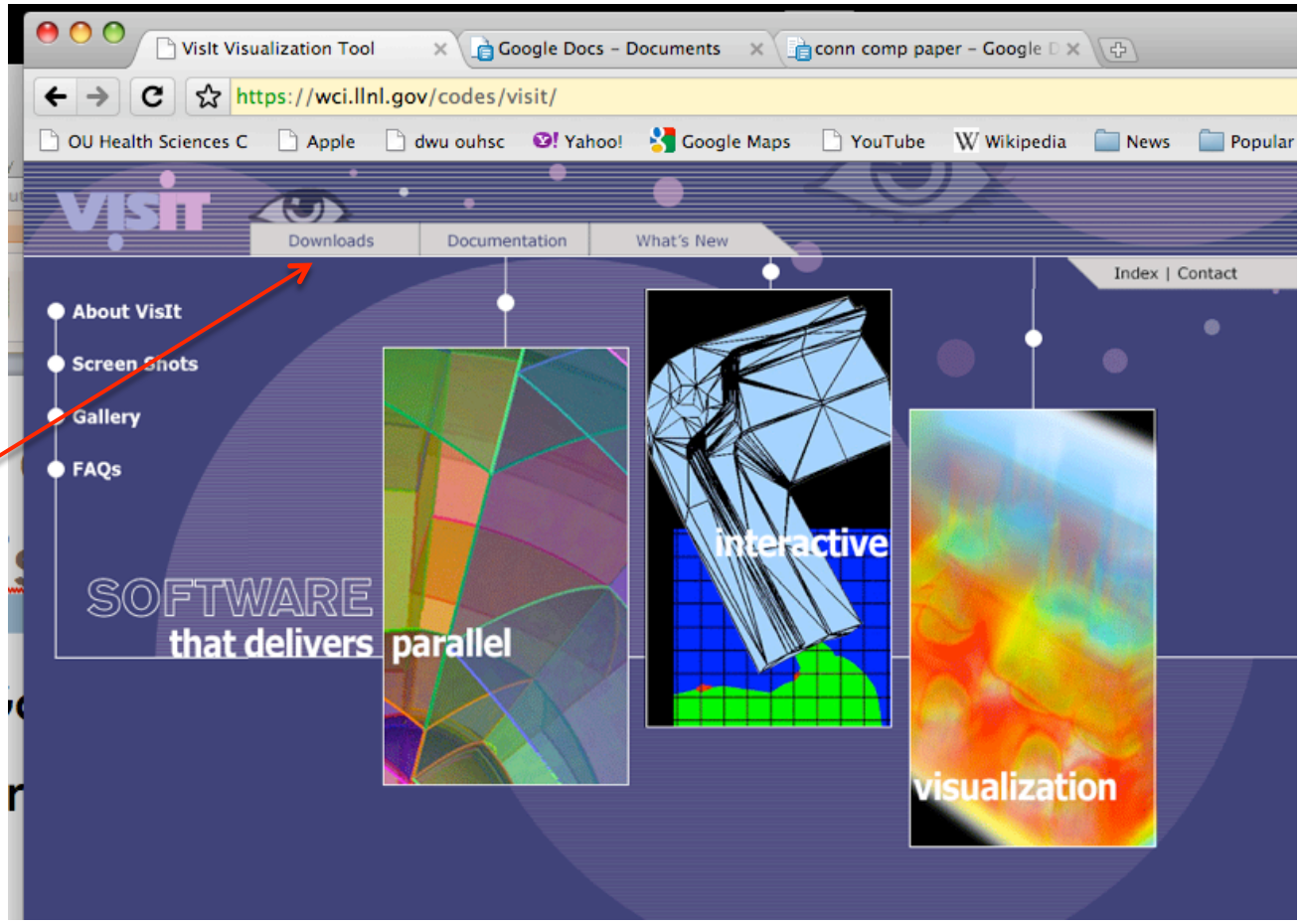
Can I use a pre-built VisIt binary or do I need to build it myself?

- Pre-built binaries work on most modern machines.
- … but pre-built binaries are serial only.
  - Why the VisIt team can't offer parallel binaries:
    Your MPI libraries, networking libraries are unlikely to match ours
- … and it is difficult to use your own custom plugins with the pre-builts.

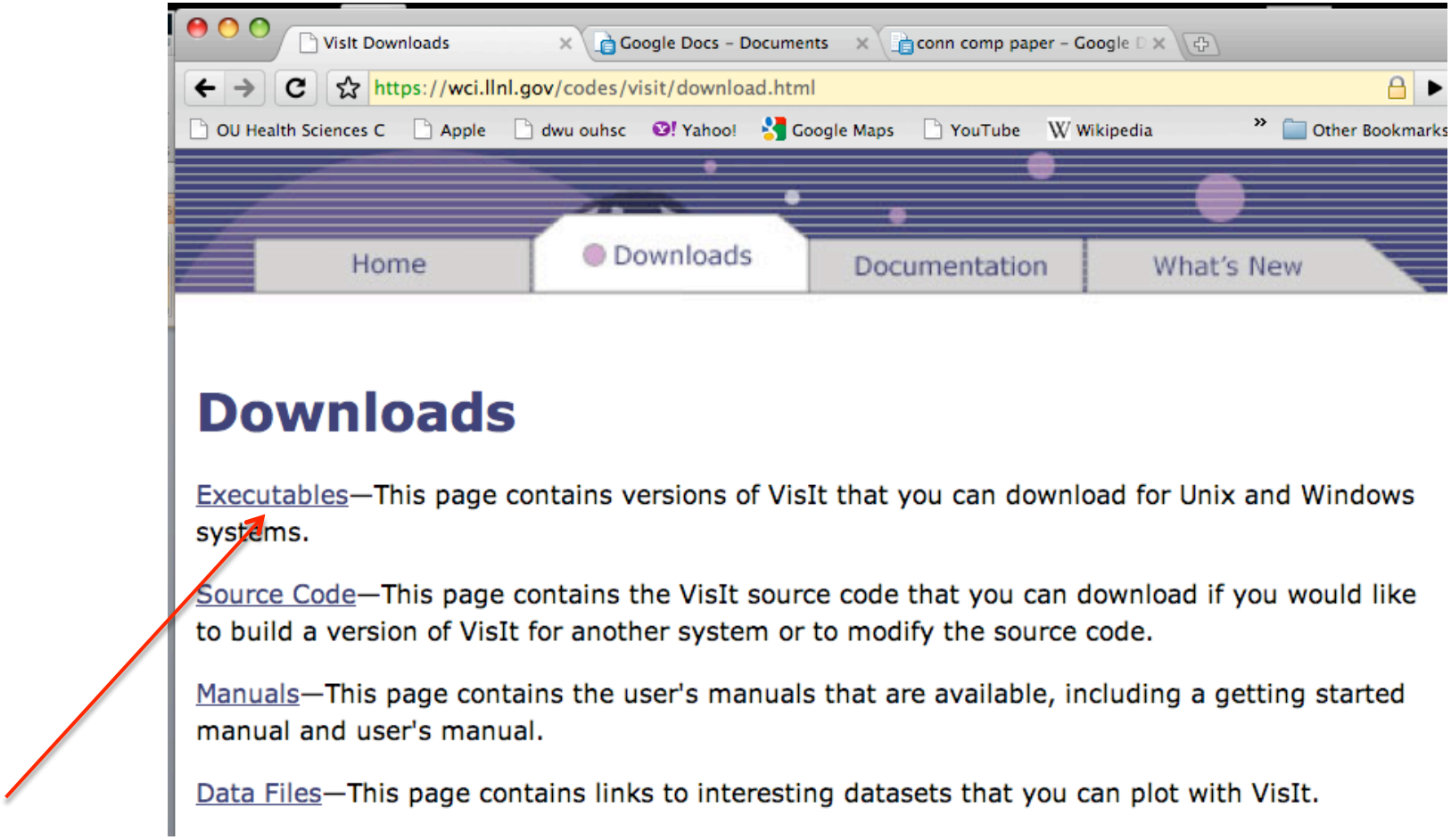
- Recommendation: try to use the pre-builts first and build VisIt yourself if they don't work.
- Also: all VisIt clients run serial-only. If you want to install VisIt on your desktop to connect to a remote parallel machine, serial is OK.

How do I use pre-built VisIt binaries?

- A: Go to http://www.llnl.gov/visit



How do I use pre-built VisIt binaries?



How do I use pre-built VisIt binaries?

VisIt Executables

This page contains links to download VisIt executables for Unix, Windows, and Mac OS X systems. The page contains several versions of VisIt, organized from the most recent to the oldest. The unix and Mac OS X executables require downloading an install script along with the file containing the executable. The Windows executables are packaged in a self contained installer. Instructions for installing VisIt can be found in the install notes. Md5 and sha1 checksums, as well as file sizes are provided for checking that the files were properly downloaded if corruption of the files is suspected during the download process.

VisIt 2.1.0

- VisIt release notes
- VisIt install script
- VisIt install notes
- VisIt md5 checksums
- VisIt sha1 checksums
- VisIt file sizes

Important

platform	executable
Linux - x86 32 bit Redhat Enterprise Linux 3, hoth.llnl.gov 2.4.21-27.0.2c.ELsmp, gcc 3.2.3 Will work on most Linux x86 systems.	download
Linux - x86_64 64 bit Ubuntu 8.04, pion.ornl.gov 2.6.24-19, gcc 4.2.4	download
Linux - x86_64 64 bit Redhat Enterprise Linux 4, photon.ornl.gov 2.6.9-89.0.20.ELsmp, gcc 3.4.6 Will work on most Linux x86_64 systems.	download

How do I use pre-built VisIt binaries?

Linux - x86_64 64 bit Ubuntu 8.04, pion.ornl.gov 2.6.24-19, gcc 4.2.4	download
Linux - x86_64 64 bit Redhat Enterprise Linux 4, photon.ornl.gov 2.6.9-89.0.20.ELsmp, gcc 3.4.6 Will work on most Linux x86_64 systems.	download
Linux - x86_64 64 bit Redhat Enterprise Linux 5, yana.llnl.gov 2.6.18-76chaos, gcc 4.1.2 Will work on most Linux x86_64 systems.	download
Linux - x86_64 64 bit Scientific Linux SL release 5.4, euclid.nersc.gov 2.6.18-164.9.1.el5-bsdv3, gcc 4.1.2	download
Windows (Xp / Vista / 7) 32 bit MSVC8, Visual Studio 2005	download
Mac OS X - Intel Darwin 10.5, Darwin Kernel Version 9.7.0, gcc 4.0.1, OpenMPI *(Includes parallel VisIt compatible with MacOS X 10.5's default MPI)*	download
Mac OS X - Intel 64 bit Darwin 10.6.3, Darwin Kernel Version 10.3.0, gcc 4.2.1, OpenMPI *(Includes parallel VisIt compatible with MacOS X 10.6's default MPI)*	download
Mac OS X - Intel Darwin 10.4	download
AIX - 32 bit AIX 5.3, up.llnl.gov 00C5D6DD4C00, xlc	download
AIX - 64 bit AIX 5.3, up.llnl.gov 00C5D6DD4C00, xlc	download
Java client library (jar file, compiled classes, source code, examples)	download

How do I use the pre-built VisIt binaries?

- Unix:
  - Download binary
  - Download install script
  - Run install script
  - --or—
  - Download binary
  - Untar

Good for host profiles, maintaining multiple versions, multiple OSs

Quick & easy

- Mac:
  - Download and open disk image.
  - Follow instructions in the README file: run included install script
- Windows:
  - Download installer program & run
- Full install notes:
  - https://wci.llnl.gov/codes/visit/2.2.1/INSTALL_NOTES

Important step: choosing host profiles

- Many supercomputing sites have set up “host profiles”.
  - These files contain all the information about how to connect to their supercomputers and how to launch parallel jobs there.
- You select which profiles to install when you install VisIt.
- Profiles that come with VisIt:
  - NERSC, LLNL Open, LLNL Closed, ORNL, Argonne, TACC, LBNL desktop network, Princeton, UMich CAC
- Other sites maintain profiles outside of VisIt repository.
  - If you know folks running VisIt in parallel at a site not listed above, ask them for their profiles.

“How to make Vislt work after you get home”

- How to get Vislt running on your machine
  - Downloading and installing Vislt
  - Building Vislt from scratch
- How to get Vislt to read your data
- How to get help when you run into trouble
- I like the power of Vislt, but I hate the interface
- How to run client-server

Building VisIt from scratch

- Building VisIt from scratch on your own is *very* difficult.
- … but the "build_visit" script is fairly reliable.

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 third party libraries
- Patches them to accommodate OS quirks
- Builds the third party libraries.
- Creates “config-site” file, which communicates information about where 3rd party libraries live to VisIt’s build system.
- Downloads VisIt source code
- Builds VisIt

“build_visit” details

- There are two ways to use build_visit:
  - Curses-style GUI
  - Command line options through --console
    - Developers all use --console and it shows!!
- Tip:
  - Don’t build every third party library unless you really need to.
    - Set up a “--thirdparty-path”.

“build_visit” details

- Q1: How long does build_visit take? A: hours
- Q2: I have my own Qt/VTK/Python, can I use those?
  - Hank highly recommends against
- Q3: What happens after build_visit finishes?
  - A1: you can run directly in the build location
  - A2: you can make a package and do an install like you would with the pre-built binaries

“build_visit” details

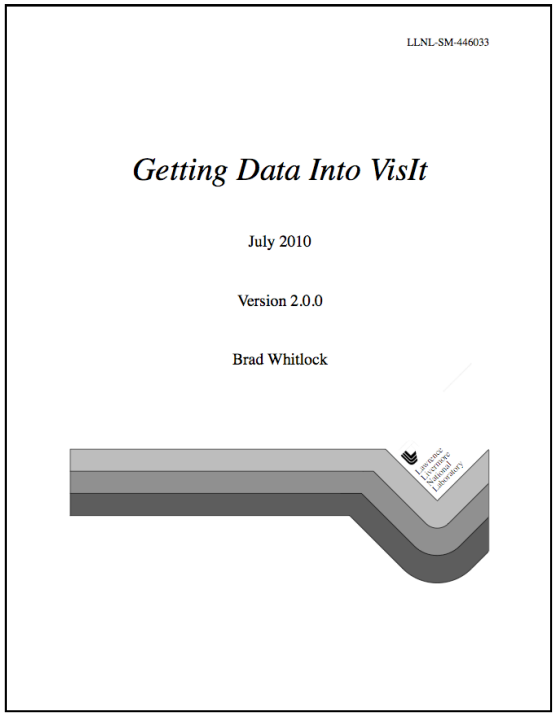
- Most common build_visit failures:
  - gcc is not installed
  - X11 development package is not installed
  - OpenGL development package is not installed
    - We should probably improve detection of this case, but we’re leery about false positives.
- Most common VisIt runtime failure: really antique OpenGL drivers.
  - Hank runs SUSE 9.1 (from 2005) at home.
- Build process for Windows is very different. Rarely a need to build on Windows, aside from VisIt development.

“How to make VisIt work after you get home”

- How to get VisIt running on your machine
  - Downloading and installing VisIt
  - Building VisIt from scratch
- How to get VisIt to read your data
- How to get help when you run into trouble
- I like the power of VisIt, but I hate the interface
- How to run client-server

How to get your data into VisIt

- There is an extensive (and up-to-date!) manual on this topic: "Getting Your Data Into VisIt"
- Three ways:
  - Use a known format
  - Write a file format reader
  - In situ processing
    - Latter two covered in afternoon course


LLNL-SM-446033

Getting Data Into VisIt

July 2010

Version 2.0.0

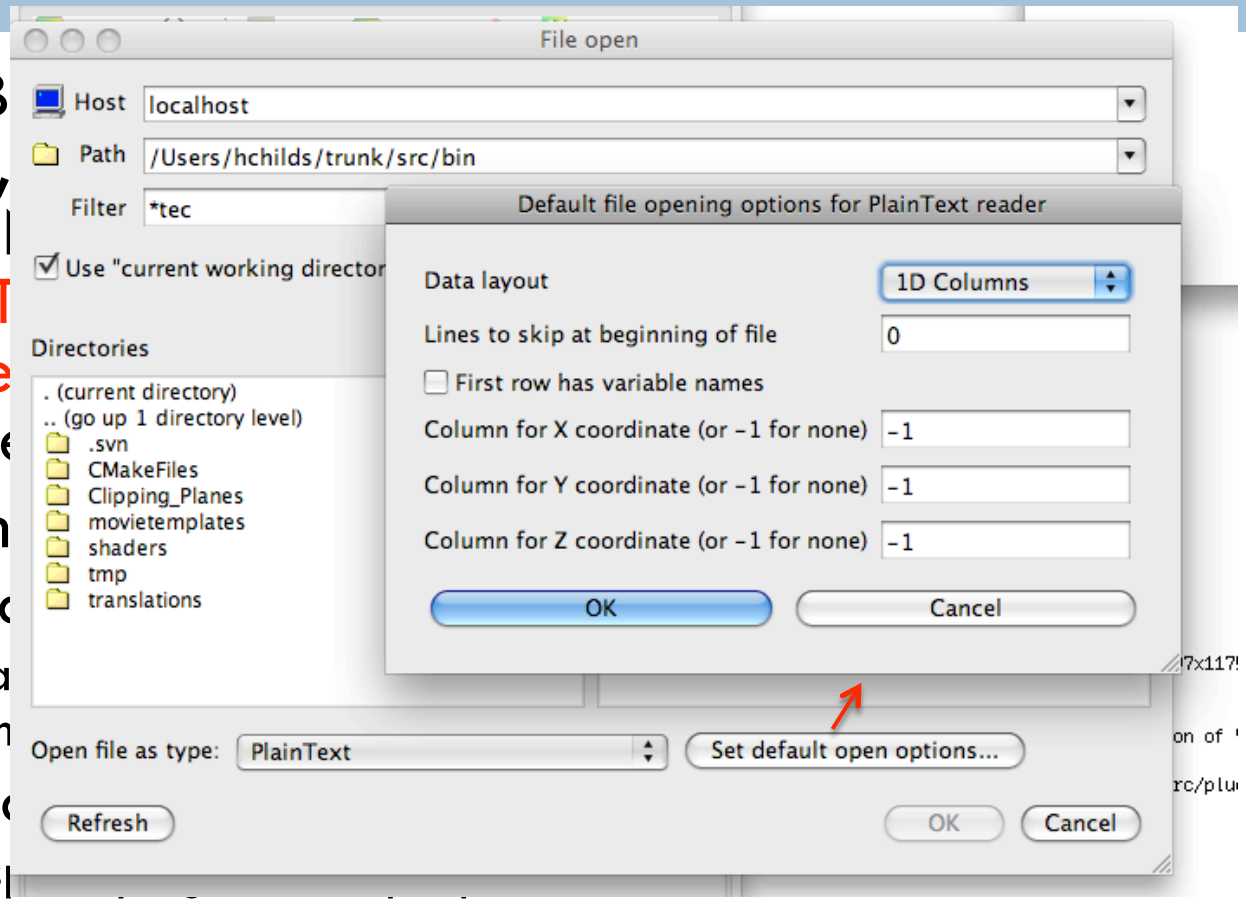
Brad Whitlock

File formats that VisIt supports

- 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 more
  - 113 total readers
  - http://www.visitusers.org/index.php?title=Detailed_list_of_file_formats_VisIt_supports
- Some readers are more robust than others.
  - For some formats, support is limited to flavors of a file a VisIt developer has encountered previously (e.g. Tecplot).

File formats that VisIt supports

- ADIOS, BOV, B
  EnSight, ENZO,
  Images (TIFF, PI
  NASTRAN, NET
  PlainText, Pixie
  and many more
- Two readers th
  - BOV: raw binc
    - →you have a
      describes dim
  - PlainText: reac
    - Controls for s



File formats that VisIt supports

- 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 more
- Common array writing libraries:
  - NETCDF: VisIt reader understands many (but not all) conventions
  - HDF5
    - Pixie is most general HDF5 reader
    - Many other HDF5 readers

File formats that VisIt supports

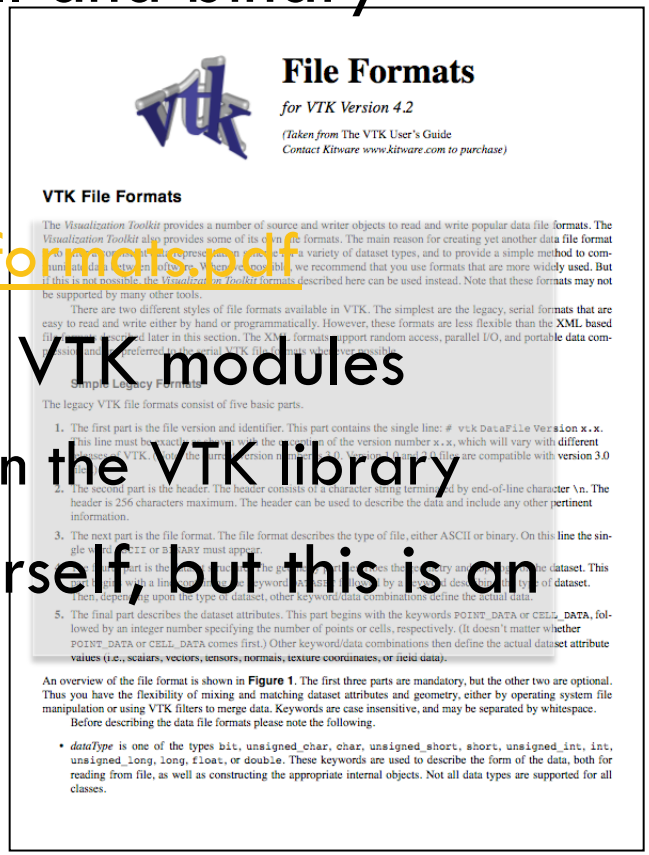
- ADIOS, BOV, Boxlib, CCM, CGNS, Chombo, CLAW, EnSight, ENZO, Exodus, FLASH, Fluent, GDAL, Gadget, Images (TIFF, PNG, etc), ITAPS/MOAB, LAMMPS, NASTRAN, NETCDF, Nek5000, OpenFOAM, PLOT3D, PlainText, Pixie, Shapefile, Silo, Tecplot, VTK, Xdmf, Vs, and many more
- Xdmf: specify an XML file that describes semantics of arrays in HDF5 file
- VizSchema (Vs): add attributes to your HDF5 file that describes semantics of the arrays.

File formats that VisIt supports

- 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 more
- VTK: not built for performance, but it is great for getting into VisIt quickly
- Silo: higher barriers to entry, but performs well and fairly mature

VTK File Format

- The VTK file format has both ASCII and binary variants.
  - Great documentation at http://www.vtk.org/VTK/img/file-formats.pdf
- Easiest way to write VTK files: use VTK modules
  - … but this creates a dependence on the VTK library
- You can also try to write them yourself, but this is an error prone process.
- Third option: visit_writer



File Formats

for VTK Version 4.2

(Taken from The VTK User's Guide
Contact Kitware www.kitware.com to purchase)

VTK File Formats

The *Visualization Toolkit* provides a number of source and writer objects to read and write popular data file formats. The *Visualization Toolkit* also provides some of its own file formats. The main reason for creating yet another data file format is to [illegible] represent [illegible] a variety of dataset types, and to provide a simple method to communicate data between others. Whenever possible, we recommend that you use formats that are more widely used. But if this is not possible, the *Visualization Toolkit* formats described here can be used instead. Note that these formats may not be supported by many other tools.

There are two different styles of file formats available in VTK. The simplest are the legacy, serial formats that are easy to read and write either by hand or programmatically. However, these formats are less flexible than the XML based file formats discussed later in this section. The XML formats support random access, parallel I/O, and portable data compression and are preferred to the serial VTK file formats whenever possible.

Simple Legacy Formats

The legacy VTK file formats consist of five basic parts.

1. The first part is the file version and identifier. This part contains the single line: `# vtk DataFile Version x.x`. This line must be exactly [illegible] with the exception of the version number `x.x`, which will vary with different versions of VTK. [illegible] version [illegible] files are compatible with version 3.0
2. The second part is the header. The header consists of a character string terminated by end-of-line character \n. The header is 256 characters maximum. The header can be used to describe the data and include any other pertinent information.
3. The next part is the file format. The file format describes the type of file, either ASCII or binary. On this line the single [illegible] `ASCII` or `BINARY` must appear.
4. [illegible] the dataset. This [illegible] of dataset. Then, depending upon the type of dataset, other keyword/data combinations define the actual data.
5. The final part describes the dataset attributes. This part begins with the keywords `POINT_DATA` or `CELL_DATA`, followed by an integer number specifying the number of points or cells, respectively. (It doesn't matter whether `POINT_DATA` or `CELL_DATA` comes first.) Other keyword/data combinations then define the actual dataset attribute values (i.e., scalars, vectors, tensors, normals, texture coordinates, or field data).

An overview of the file format is shown in **Figure 1**. The first three parts are mandatory, but the other two are optional. Thus you have the flexibility of mixing and matching dataset attributes and geometry, either by operating system file manipulation or using VTK filters to merge data. Keywords are case insensitive, and may be separated by whitespace.

Before describing the data file formats please note the following.

- *dataType* is one of the types `bit`, `unsigned_char`, `char`, `unsigned_short`, `short`, `unsigned_int`, `int`, `unsigned_long`, `long`, `float`, or `double`. These keywords are used to describe the form of the data, both for reading from file, as well as constructing the appropriate internal objects. Not all data types are supported for all classes.

VisltWriter writes VTK files

- It is a "library" (actually a single C file) that writes VTK-compliant files.
  - The typical path is to link visit_writer into your code and write VTK files
- There is also Python binding for visit_writer.
  - The typical path is to write a Python program that converts from your format to VTK
- Both options are short term: they allow you to play with Vislt on your data. If you like Vislt, then you typically formulate a long term file format strategy.
- More information on visit_writer:
  - http://visitusers.org/index.php?title=VisltWriter

Python VisItWriter in action

```
import visit_writer
import math
import sys

nX = 20
nY = 20
conn = []
for i in range(nX-1):
   for j in range(nY-1):
      pt1 = j*(nX) + i;
      pt2 = j*(nX) + i+1;
      pt3 = (j+1)*(nX) + i+1;
      pt4 = (j+1)*(nX) + i;
      conn.append([ "quad", pt1, pt2, pt3, pt4 ])

pts = []
rad = []
for i in range(nX):
   for j in range(nY):
      pts.extend([ float(i), float(j), 0 ])
      rad.append( math.sqrt(i*i + j*j) )

var_datum = [ "radius", 1, 1, rad ]
vars = [ var_datum ]
visit_writer.WriteUnstructuredMesh("ugrid.vtk", 0, pts, conn, vars)

sys.exit()
```

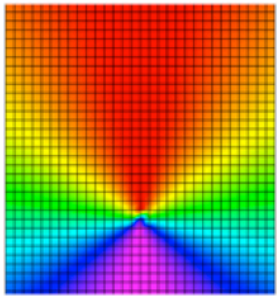
Silo file format

- Silo is a mature, self-describing file format that deals with multi-block data.
- It has drivers on top of HDF5, NetCDF, and “PDB”.
- Fairly rich data model
- More information:
  - https://wci.llnl.gov/codes/silo/

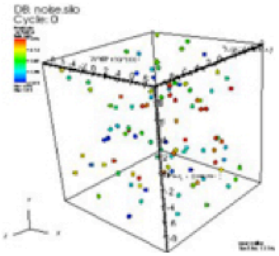
Silo features

Welcome to Silo

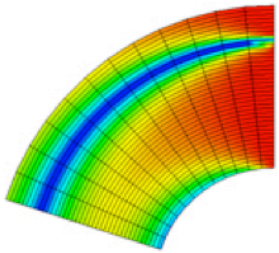
A mesh and field I/O library and scientific database



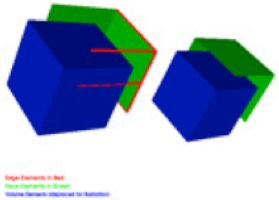
Structured Rectilinear Mesh



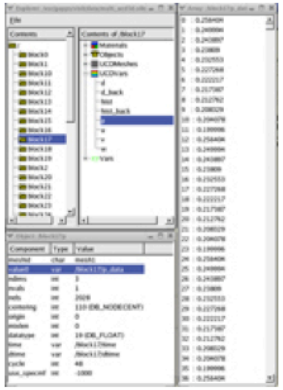
Gridless Point Mesh



Structured (Curvilinear) Mesh

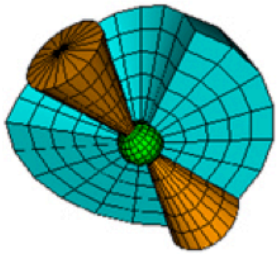


Arbitrary Subsets

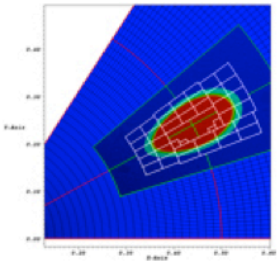


Silex browser for Silo files

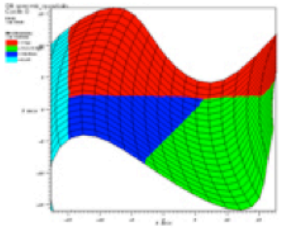
Constructive Solid Geometry (CSG) Mesh



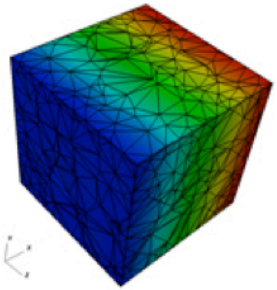
Unstructured Zoo (UCD) Mesh



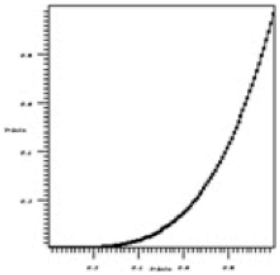
Adaptive Mesh Refinement (AMR) Mesh



Mixing Materials



Arbitrary Polyhedral Mesh



XY Curve

“How to make Vislt work after you get home”

- How to get Vislt running on your machine
  - Downloading and installing Vislt
  - Building Vislt from scratch
- How to get Vislt to read your data
- How to get help when you run into trouble
- I like the power of Vislt, but I hate the interface
- How to run client-server

How to get help when you run into trouble

- Six options:
  - FAQ
    - http://visit.llnl.gov/FAQ.html
  - Documentation
    - https://wci.llnl.gov/codes/visit/doc.html
    - http://www.visitusers.org
  - Vislt-users mailing list
  - Vislt-users archives
  - Vislt users forum
  - Vislt-help-XYZ mailing list

FAQ: http://visit.llnl.gov/FAQ.html

Home | Downloads | Documentation | What's New

Index | Contact

Frequently Asked Questions

1. Contact information
2. Supported platforms
3. Optimal hardware/software
4. Debugging problems starting VisIt or opening files
5. Stereo rendering
6. VisIt won't run on Linux
7. Slow performance on Linux
8. Slow performance Using SSH
9. No output in visualization window
10. Accessing data on remote machine
11. Running VisIt in parallel
12. Supported data file formats
13. Getting your data into VisIt
14. Making a movie of your data
15. Setting your user name to connect to a remote machine
16. Cannot connect to a remote computer
17. Building VisIt on a Windows computer
18. Installing VisIt on a MacOS X computer
19. Hanging at 12% on Windows computers
20. Getting the Plugin Developer's Guide
21. Writing a plugin for VisIt
22. When new versions of VisIt are released
23. What is new in the latest version of VisIt
24. Compilers that can be used to build VisIt
25. VisIt's licensing agreement
26. Slow performance with ATI cards on Linux
27. Custom plugins with a downloaded VisIt binary
28. Getting HDF5 data into VisIt
29. Getting NETCDF data into VisIt
30. When I run VisIt on my Linux machine, I get a black screen
31. I get the message 'Publisher cannot be verified' when installing VisIt on Windows
32. Which libraries should I enable in `build_visit`?

Manuals & other documentation

- Getting started manual
- Users manual (old, but still useful)
- Python interface (to be updated in two weeks)
- Getting Data Into VisIt
- VisIt Class Slides
- VisIt Class Exercises
- This Tutorial

Visitusers.org

- Users section has lots of practical tips:
  - "I solved this problem with this technique"
  - "Here's my script to do this functionality"
- In practical terms, this is a staging area for formal documentation in the future.

Misc [edit]

- Using VisIt in an mxterm
- Using derived data functions (DDFs)
- Using the command line interface
- How volume rendering works in VisIt
- Using cross-mesh field evaluations ... how to do differences, access other time slices, etc
- Keyframing example
- Exporting databases
- Directions for specific machines
- Using the VisIt Python API with a standard Python interpreter
- Pages that contain instructions specific to certain user groups and needs
- Issues related to running VisIt on Windows under cygwin
- VisIt's Camera model
- Using VisIt's mpeg2encode
- Molecular data features
- Extracting alpha
- (Very) High resolution rendering
- Elevating shapefiles
- Raytracing your visualizations with POV-Ray and a tutorial POV-Ray exporting example

[visit-users] Building Parallel Visit: Issue w/ Qt

Vedran Coralic vcoralic at caltech.edu
Wed Nov 3 00:21:37 EDT 2010

- Previous message: [visit-users] Building Parallel Visit: Issue w/ Qt
- Next message: [visit-users] makemili
- **Messages sorted by:** [date] [thread] [subject] [author]

```
Thank you very much Jeremy! That seemed to do the trick. I have now finished
successfully building VisIt.

2010/11/2 Meredith, Jeremy S. <jsmeredith at ornl.gov>

>  Here's what I did to work around this problem:
>
> - when the Qt build fails, cd into the Qt directory and type "make install"
>
> - this appears to immediately start putting the libraries in the
> installation location even though the build "failed"
>
> - as soon as it's put the libQCLucene stuff into the installation location,
> kill the build
>
> - now type "make", and it will finish building successfully
>
> - and when it's done, type "make install" and it will finish installing
>
```

May 2010:	[Thread] [Subject] [Author] [Date
April 2010:	[Thread] [Subject] [Author] [Date

Archives by thread

ject] [author] [date]

DT 2010
ST 2010

llel Visit: Issue w/ Qt *Vedran Coralic*
g Parallel Visit: Issue w/ Qt *Meredith, Jeremy S.*
uilding Parallel Visit: Issue w/ Qt *Vedran Coralic*
Daniel, James L ERDC-GSL-MS
li *Seipel, William F NWO*
akemili *Seipel, William F NWO*
se/RCP] *Leguay Romain*
Eclipse/RCP] *Hank Childs*
of recorded macros? *Cyrus Harrison*
r hardware acceleration problems *Patrick Shinpaugh*
on a CentOS server *Katie Boyle*
- [visit-users] Running Visit on a CentOS server *Cyrus Harrison*
- [visit-users] Running Visit on a CentOS server *J.S. van Bethlehem*

- [visit-users] Controlling Annotation objects through cli *Shriram Jagannathan*
  - [visit-users] Controlling Annotation objects through cli *Cyrus Harrison*
    - [visit-users] Controlling Annotation objects through cli *Shriram Jagannathan*
      - [visit-users] Controlling Annotation objects through cli *J.S. van Bethlehem*

question from Australia to be answered by a European all while I'm asleep

- List: visit-users@ornl.gov
- More information: https://email.ornl.gov/mailman/listinfo/visit-users
- Archive: https://email.ornl.gov/pipermail/visit-users/

Board Topics

new CGNS OversetHoles « Pages 1 2 »	tpg2114	16	160	11/11/10 at 22:39:11 By: cean	
new 3d vector on 2d mesh?	tsch	2	34	11/10/10 at 09:26:36 By: Jeremy Meredith	
new Image messed up when save	Pinpin	13	150	11/09/10 at 12:14:29 By: BradWhitlock	
pseudocolor plot legend attributes in python	Jennifer	2	17	11/07/10 at 22:11:27 By: Jennifer	
new graph along 2D					
threshold variable					
Python compatibilit					
python interface ...					
new Mesa: 'make' f failed					
new applyOperator					
how to get cycle on annotation? « Pages 1 2 »					
new Averaging 2D s databases					
smooth operator					
Appearance of lines					
new Add and read p					
new No image was					

Members viewing this topic (1): **Hank Childs**.

pseudocolor plot legend attributes in python (Read 18 times)

Jennifer
YaBB Newbies
Offline

Posts: 4
Fort Collins, CO

pseudocolor plot legend attributes in python
11/07/10 at 19:06:30

Quote Modify Delete

Hello. I want to set the attributes for a pseudocolor plot legend in a python script such as the location of the legend (turn off Let VisIt manage legend position), the X-scale & Y-scale, the number of Tic Marks, and the label appearance (number format, font height). Is it possible to set these properties in a python script? If so, how can I do this?

I tried to use the Command Control to record these changes, but the output states:
"# Logging for AddAnnotationObject is not implemented yet.
Logging for SetAnnotationObjectOptions is not implemented yet."

Thanks,
Jennifer

Back to top

PM

IP Logged

Hank Childs
YaBB Moderator
Online

I use VisIt and I develop VisIt

Posts: 135
Davis, CA

Re: pseudocolor plot legend attributes in python
Reply #1 - 11/07/10 at 19:47:03

Quote Modify Split Delete

Hello Jennifer,

Each plot has an index and the plot's legend is referred to through that same index.

```
>>> GetAnnotationObjectNames()
('Plot0003',)
>>> a = GetAnnotationObject("Plot0003")
>>> a
active = 1
managePosition = 1
position = (0.05, 0.9)
xScale = 1
yScale = 1
```

Visit-help-xyz

- Some customer groups pay for VisIt funding and get direct support.
  - These customers can post directly to visit-help-xyz without being a subscriber
  - The messages are received by all VisIt developers and supported collectively
- Lists:
  - Visit-help-asc, visit-help-scidac, visit-help-gnep, visit-help-ascem

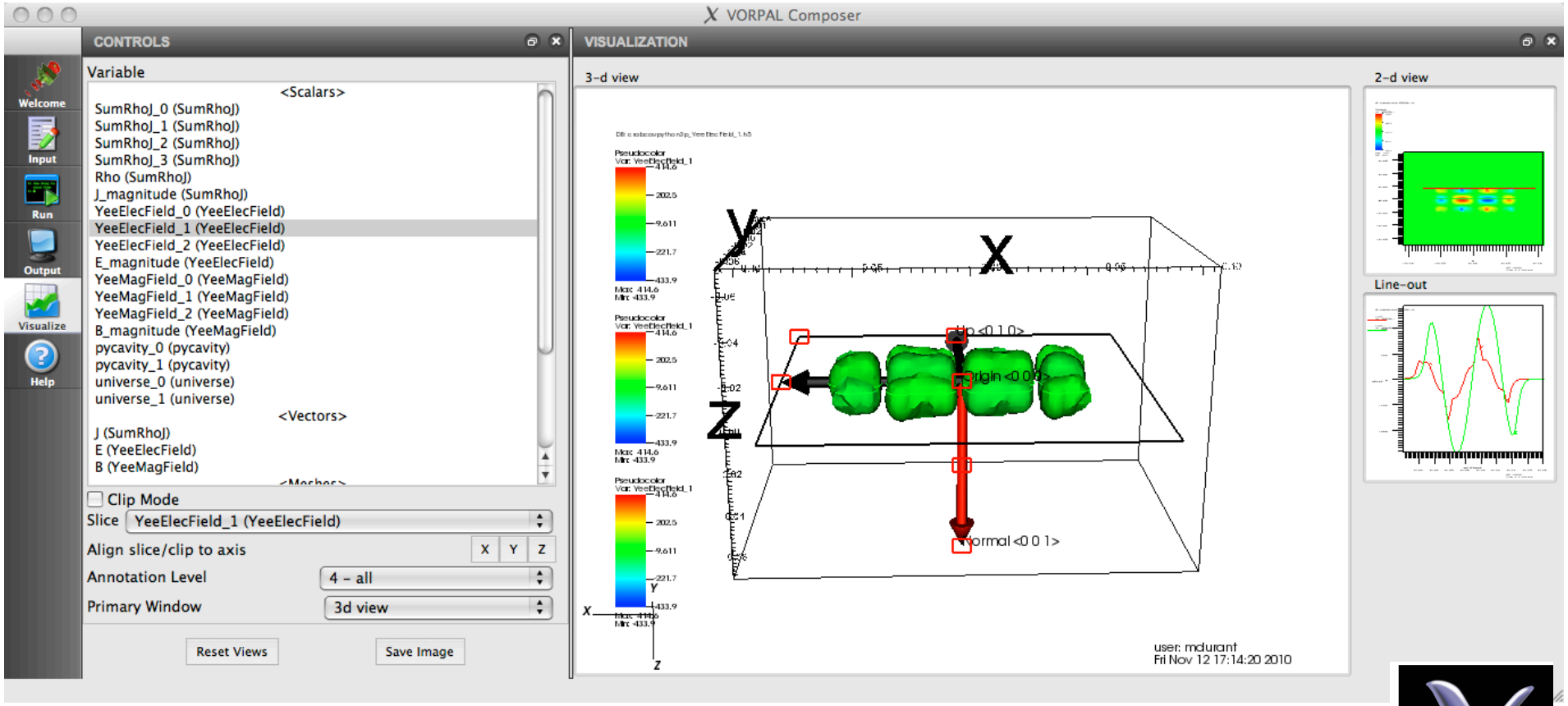
How to get help when you run into trouble

- Six options:
  - FAQ
    - http://visit.llnl.gov/FAQ.html
  - Documentation
    - https://wci.llnl.gov/codes/visit/doc.html
    - http://www.visitusers.org
  - Vislt-users mailing list
  - Vislt-users archives
  - Vislt users forum
  - Vislt-help-XYZ mailing list

“How to make Vislt work after you get home”

- How to get Vislt running on your machine
  - Downloading and installing Vislt
  - Building Vislt from scratch
- How to get Vislt to read your data
- How to get help when you run into trouble
- I like the power of Vislt, but I hate the interface
- How to run client-server

It is possible (although non-trivial) to write a custom user interface to VisIt

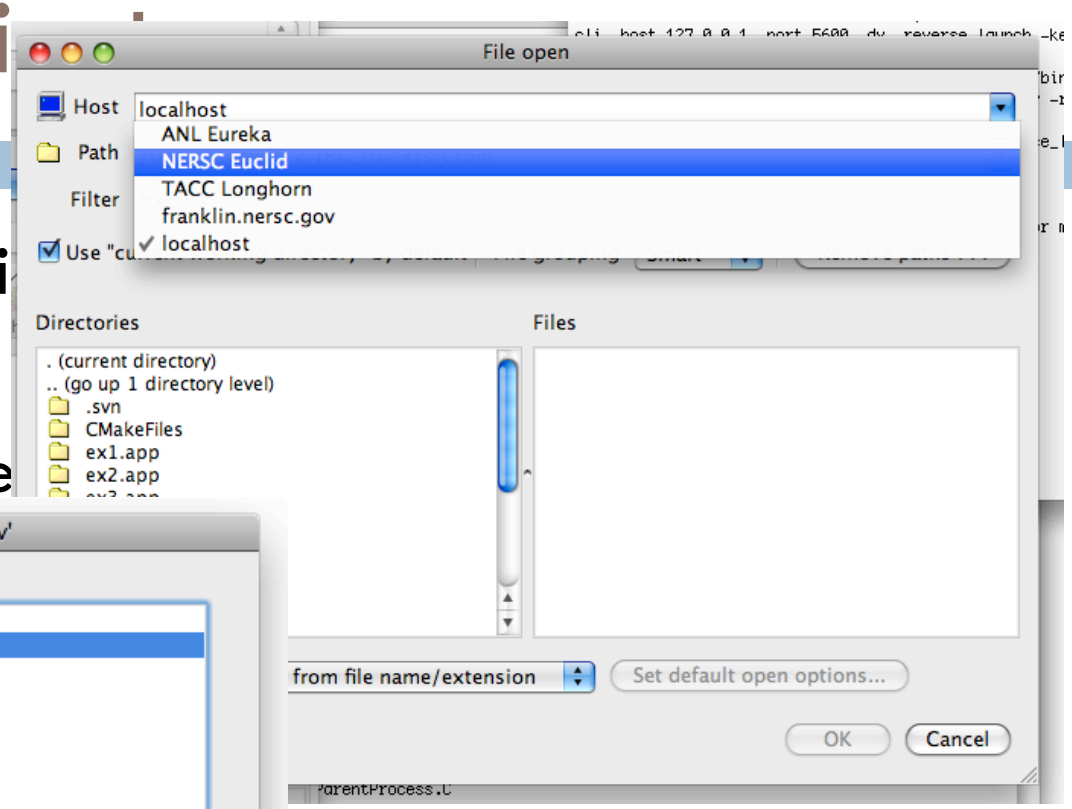


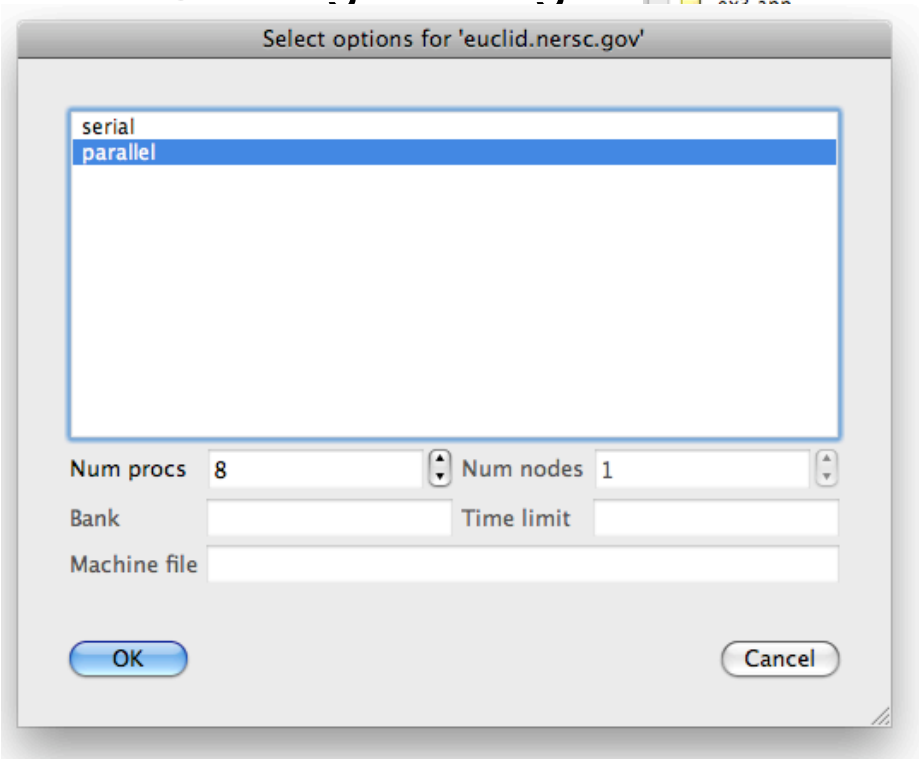
“How to make VisIt work after you get home”

- How to get VisIt running on your machine
  - Downloading and installing VisIt
  - Building VisIt from scratch
- How to get VisIt to read your data
- How to get help when you run into trouble
- I like the power of VisIt, but I hate the interface
- How to run client-server

How to run cli

- There are two criti
  - Connecting to the
  - Getting an engine

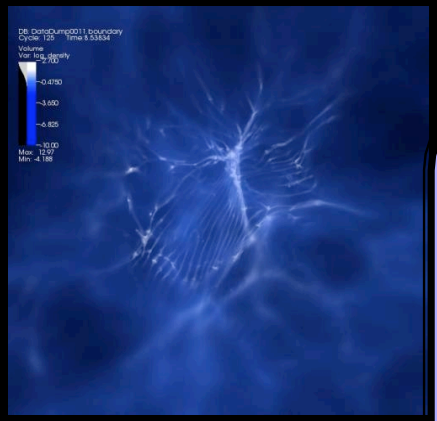


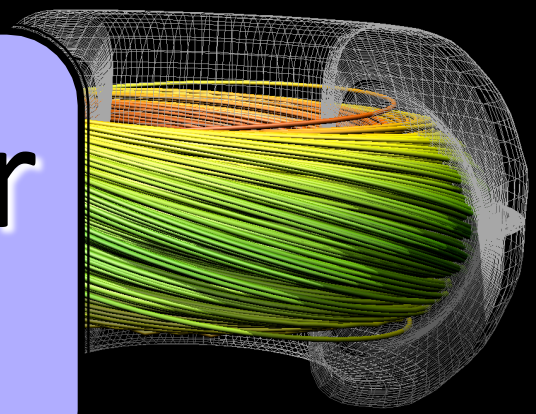


Advanced Analysis and Visualization with VisIt

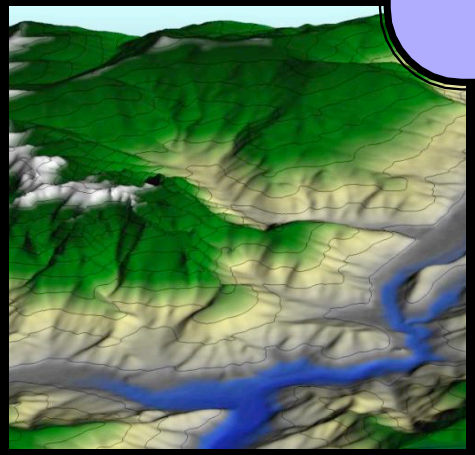
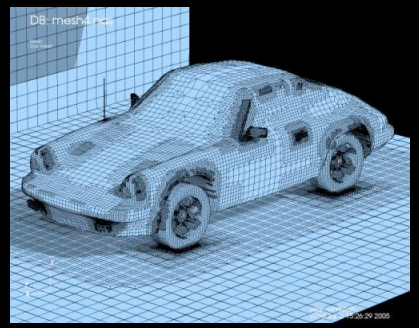
Hank Childs & David Pugmire

1:00 PM, July 15





Thank you for coming!!



- Moviemaking
- Comparisons
- + more!