

Mapping and modeling Earth Science Data

*Segment III: Some notes on other
approaches to data analysis and
Visualization*

Thorsten Becker

University of Southern California, Los Angeles

Universita di Roma TRE, June 2012

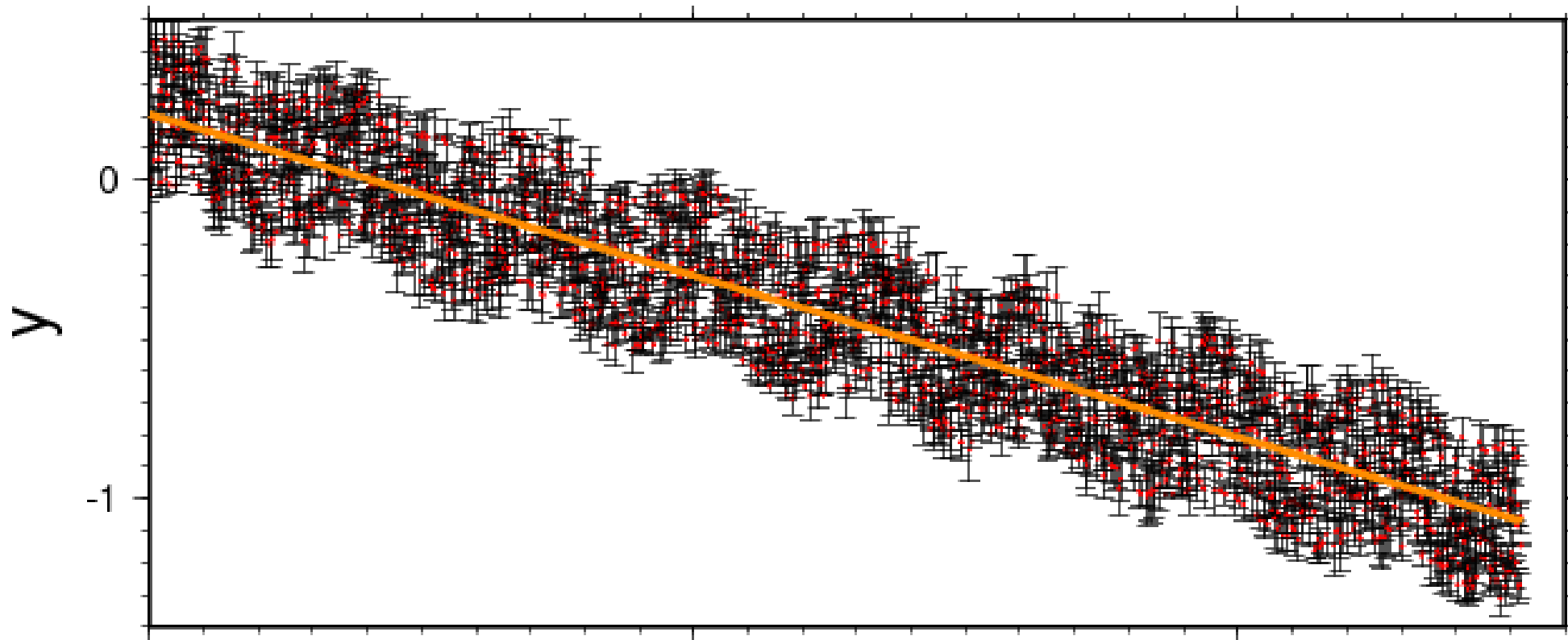
Visualization: Geographic data

- Google Earth (KML converters for GMT), [geomapapp](#)
- matlab/octave, IDL/ENVI
- free GIS systems:
 - GRASS, grads (installed in UGESCE)
- Workflows: [Kepler](#)
- Low level routines (proj etc.)
- For our purposes:
 - [GMT](#): tons of projections, tool of choice for professional quality plots
 - [iGMT](#): obtain initial scripts later

Visualization: General 2-D

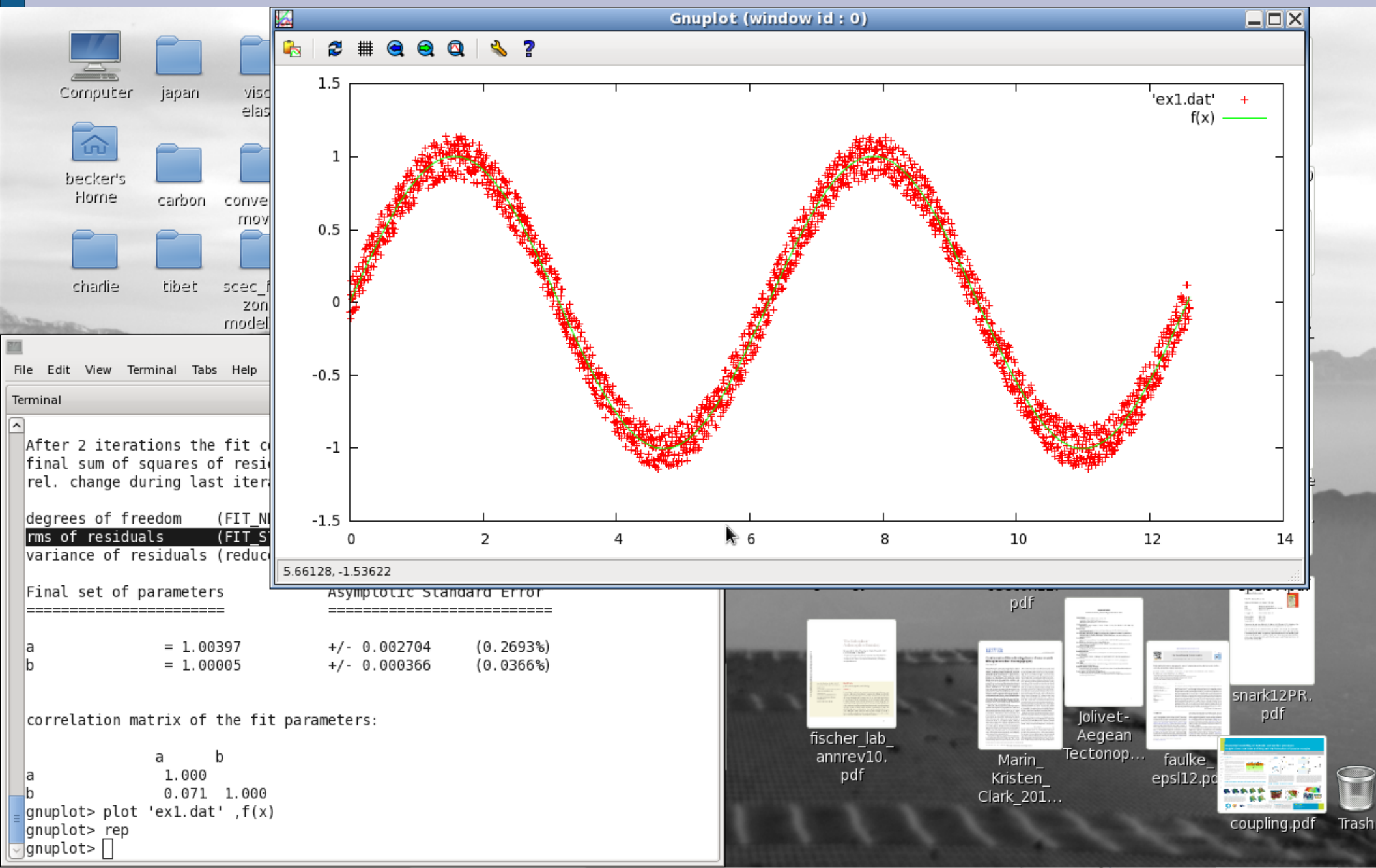
- matlab, IDL, and 1000 others
- I recommend:
 - *GNU PLOT*: nice 2-D PS plots, scriptable
 - plot 'a.dat' using 1 : (\$3/10) title 'data' w lp
 - *GMT*: made for mapping, but can produce outstanding quality EPS figures (with some work)
 - Matplotlib for python
 - those are script driven programs, no real GUI (but there is iGMT and some stuff for gnuplot)
 - both are available as C source code

2D plotting with GMT and model fitting

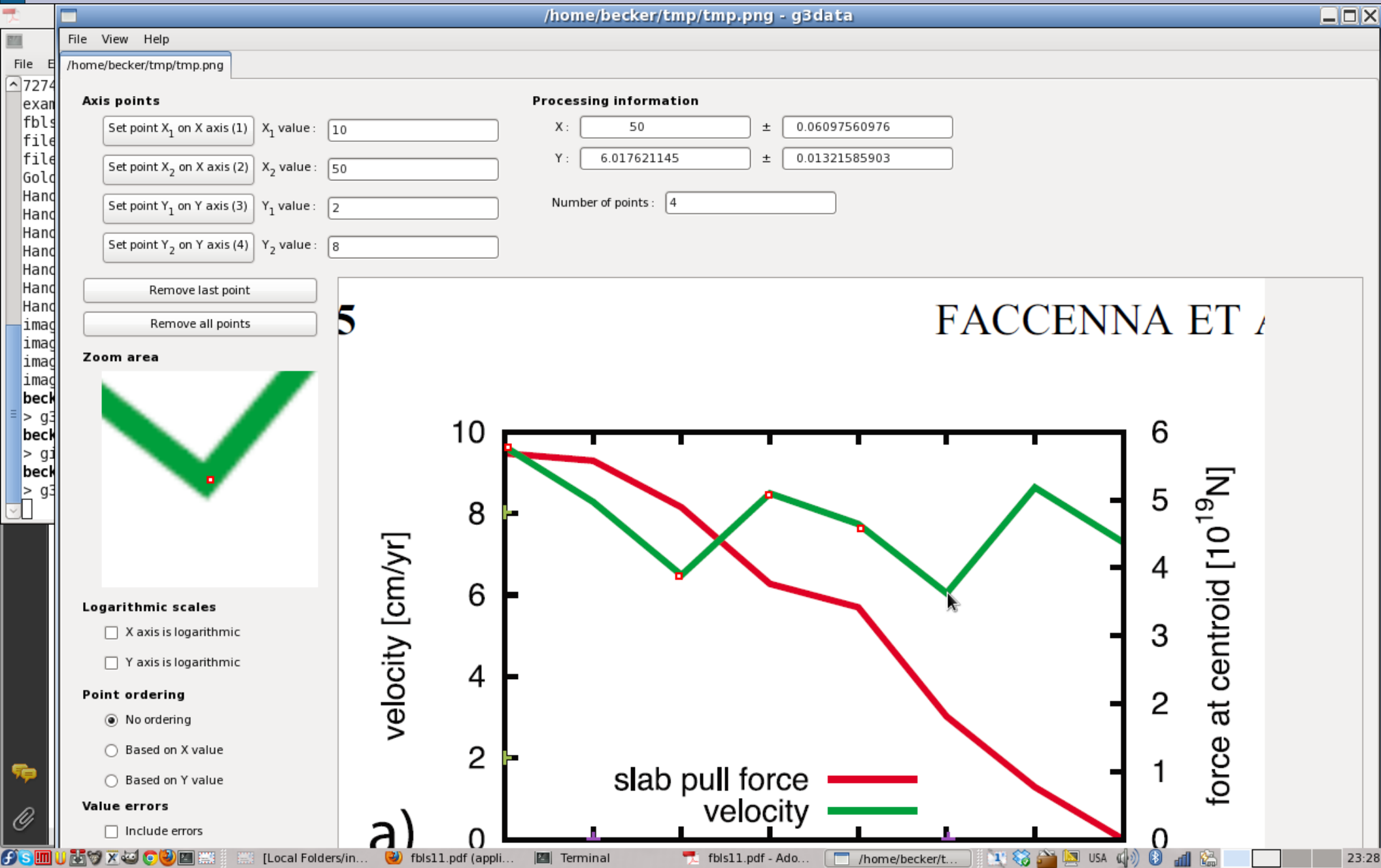


See the exercises on using `psxy`, `trend1d`, `filter1d`

GNU PLOT



Tools: g3data



Visualization: 3-D

- Commercial
 - Matlab
 - IDL (not so good)
- free software with available source code:
 - **OPEN DX**: object oriented, powerful package
 - **paraview**: VTK based parallel tool
 - Octave: like matlab, sort of
- I recommend paraview and open-dx
- There is plenty of stuff around (don't write your own)



Kitware

Search

Tell us what you think

PROJECT

RESOURCES

HELP

OPEN SOURCE

ParaView is an open-source, multi-platform data analysis and visualization application. ParaView users can quickly build visualizations to analyze their data using qualitative and quantitative techniques. The data exploration can be done interactively in 3D or programmatically using ParaView's batch processing capabilities.

ParaView was developed to analyze extremely large datasets using distributed memory computing resources. It can be run on supercomputers to analyze datasets of terascale as well as on laptops for smaller data.

News

[More News >](#)

04.10.2012 Kitware to Develop Simulation Workflow for the Department of Ener...

03.29.2012 Scalable Data Management, Analysis and Visualization (SDAV) Insti...

02.22.2012 ParaView 3.14 Released

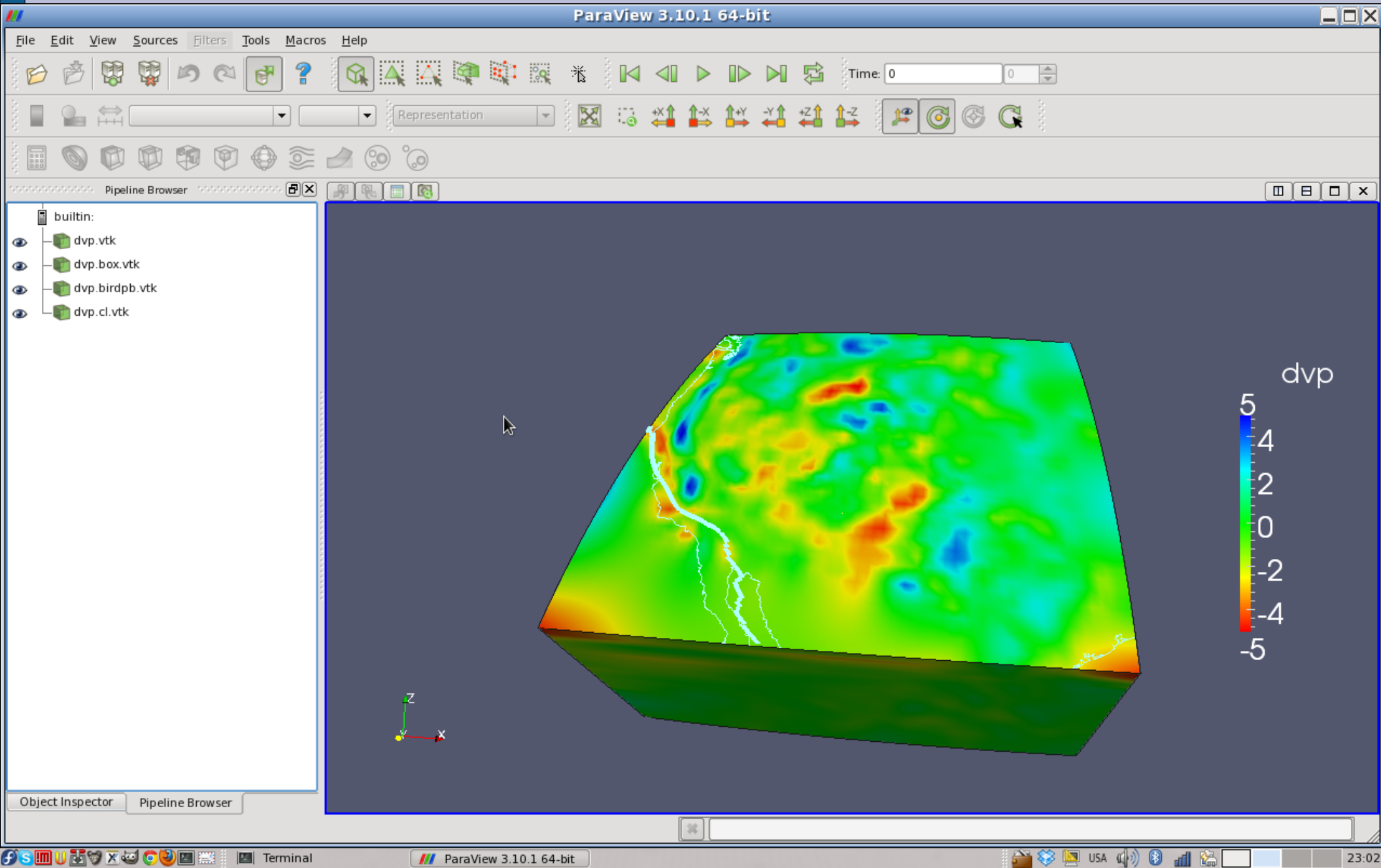
02.08.2012 ParaView Used in Atmospheric Climate Modeling

11.09.2011 ParaView 3.12.0 is Now Available!

Kitware receives HPCwire's Editors' & Readers' Choice awards for ParaView. Thank you to the global ParaView community for contributing to its success!



Paraview



Paraview

- I like it because
 - It's free
 - It's powerful
 - It's quantitative
 - It's scriptable (hey!)
 - It's parallel
- Can read a range of formats, including VTK
- `grd2vtk, grd2vtk_cart`
- `gmtpoly2vtk.awk,`
`gmtcpt2paraview.awk`

ParaviewGeo - Mozilla Firefox

File Edit View History Bookmarks Tools Help

file:///home/...homepage.html x ParaviewGeo x +

google.com https://sites.google.com/a/objectivity.ca/paraviewgeo/ paraview geo

Search this site



ParaViewGeo

Open Source Visualization for Geoscience

[Documentation](#) > [User Manual](#) >

ParaViewGeo

ParaViewGeo is a free, BSD-licensed, open source visualization package for the exploration and mining industry.

ParaViewGeo is designed to run on distributed and shared memory parallel systems and single processor systems and can visualize datasets of varying sizes from small to very large. The application can integrate a large set of data and make compelling 3D presentations to give more context to the data as a whole, or to individual sets of data in particular.

Our developers are continuously adding enhancements to the application, supporting additional file formats and increasing the functions available; a free download of the latest version is always available on our [download](#) page.

Please read the [licensing information page](#) for information on ParaViewGeo licensing.

Open Source - The Free Software Alternative

ParaViewGeo is being distributed as free software. There is no intent to charge any money to sell, rent or license the software to the geological/mining community.

ParaViewGeo is a direct off shoot of ParaView, a parallel scientific visualization software package, that was developed by Kitware and was funded by a number of major US research organizations. You can find more information about ParaView and Kitware at <http://www.kitware.com>.

Kitware is a principled promoter of open source software and as such has made the complete source code for

ParaViewGeo

- ▼ **Documentation**
 - User Manual
 - ▶ ParaViewGeo
 - Plugins
 - Documentation
 - Compile
 - ParaViewGeo (Linux)
 - ParaViewGeo Plugins Pack
 - Installation
- ▼ **Downloads**
 - ParaviewGeo
 - Executables
 - ParaViewGeo Plugins Pack
 - Source Code
- Contact Us**
- La versión en español
- Licensing Information**
- Sitemap**

Done



Terminal

ParaviewGeo - Mozilla...

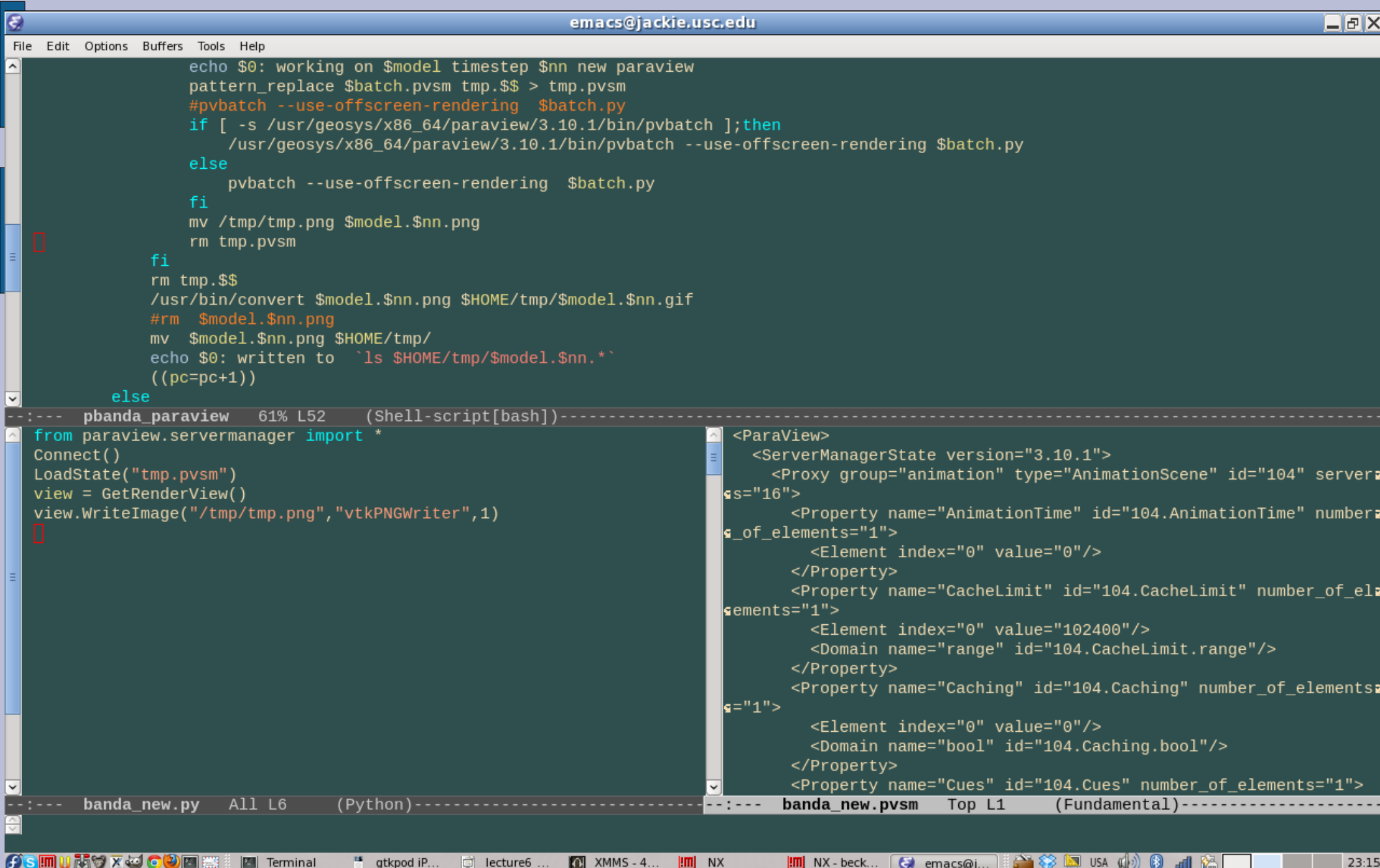
Local Folders/lists/gm...

lecture6_visualization...



16:41

Paraview scripting



The screenshot shows an Emacs editor window with the title bar "emacs@jackie.usc.edu". The main window displays a shell script for Paraview batch processing. The script uses `echo`, `pattern_replace`, `pvbatch`, `if`, `else`, `fi`, `mv`, and `rm` to process a series of Paraview models. A status bar at the bottom indicates the current buffer is "pbanda_paraview" at line 52, column 61, in a bash shell script.

```
echo $0: working on $model timestep $nn new paraview
pattern_replace $batch.pvsm tmp.$$ > tmp.pvsm
#pvbatch --use-offscreen-rendering $batch.py
if [ -s /usr/geosys/x86_64/paraview/3.10.1/bin/pvbatch ];then
    /usr/geosys/x86_64/paraview/3.10.1/bin/pvbatch --use-offscreen-rendering $batch.py
else
    pvbatch --use-offscreen-rendering $batch.py
fi
mv /tmp/tmp.png $model.$nn.png
rm tmp.pvsm

fi
rm tmp.$$
/usr/bin/convert $model.$nn.png $HOME/tmp/$model.$nn.gif
#rm $model.$nn.png
mv $model.$nn.png $HOME/tmp/
echo $0: written to `ls $HOME/tmp/$model.$nn.*`
((pc=pc+1))
else
```

The bottom panel shows a Python script for Paraview server manager. The script imports `paraview.servermanager` and uses `Connect()`, `LoadState()`, `GetRenderView()`, and `WriteImage()` to render a Paraview state. A status bar at the bottom indicates the current buffer is "banda_new.py" at line 6, column 1, in a Python script.

```
from paraview.servermanager import *
Connect()
LoadState("tmp.pvsm")
view = GetRenderView()
view.WriteImage("/tmp/tmp.png", "vtkPNGWriter", 1)
```

The right panel shows the XML output of the Paraview server manager. The XML structure includes `<ParaView>`, `<ServerManagerState version="3.10.1">`, `<Proxy group="animation" type="AnimationScene" id="104" server>`, and various `<Property>` and `<Element>` tags for animation and caching settings.

```
<ParaView>
  <ServerManagerState version="3.10.1">
    <Proxy group="animation" type="AnimationScene" id="104" server>
      <Property name="AnimationTime" id="104.AnimationTime" number_of_elements="1">
        <Element index="0" value="0"/>
      </Property>
      <Property name="CacheLimit" id="104.CacheLimit" number_of_elements="1">
        <Element index="0" value="102400"/>
        <Domain name="range" id="104.CacheLimit.range"/>
      </Property>
      <Property name="Caching" id="104.Caching" number_of_elements="1">
        <Element index="0" value="0"/>
        <Domain name="bool" id="104.Caching.bool"/>
      </Property>
      <Property name="Cues" id="104.Cues" number_of_elements="1">
```

The bottom status bar shows the current buffer is "banda_new.pvsm" at line 1, column 1, in a Fundamental script.