Data Visualization using ParaView on Mistral

Niklas Röber
Deutsches Klimarechenzentrum (DKRZ)

Lectures & Breaks

Monday / Tuesday

- 09.00 – 10.30 Lecture 1
- 10.30 – 11.00 Coffee & Cookies
- 11.00 – 12.30 Lecture 2

Lunch

2

- 13.30 – 15.00 Lecture 3
- 15.00 – 15.30 Coffee & Cookies
- 15.30 – 17.00 Lecture 4

Outline

Day1 (Basics)

- Introduction, getting started
- ECHAM data
- ParaView basics

Lunch

- HAMOCC TP6M data
- ICON Ocean data
- Snapshots and Animations

Day2 (Advanced)

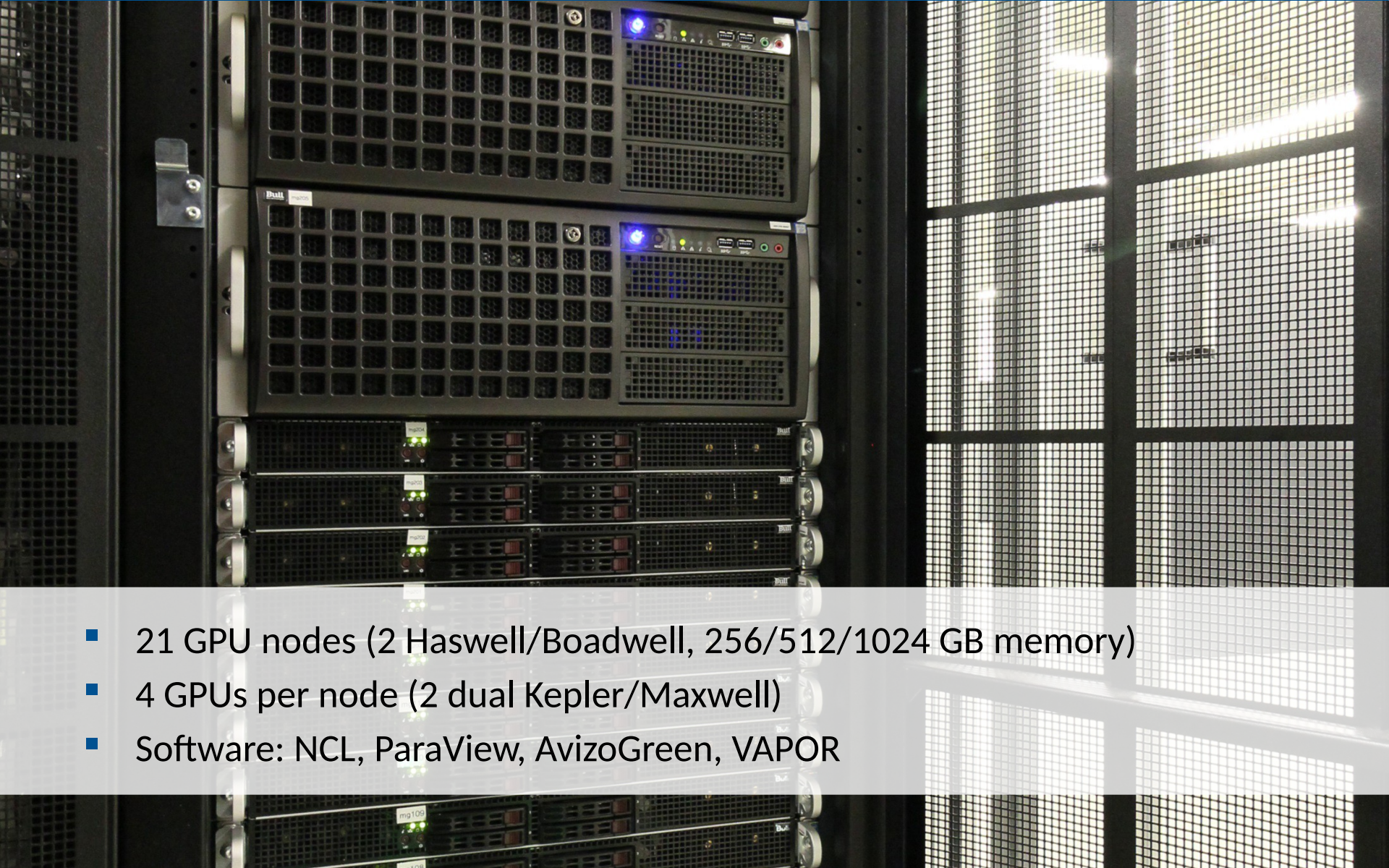
- HD(CP)^2 data (lon/lat)
- Subsets
- Derive Vorticity, Divergence …

Lunch

- Open questions
- Working with your data

HLRE3 – Mistral

- 3000 Bullx DLC 720 (>100.000 cores, 240 TB memory, 54 PB disk)
- 3.6 PetaFLOPS, Infiniband FDR (Top500: 33)

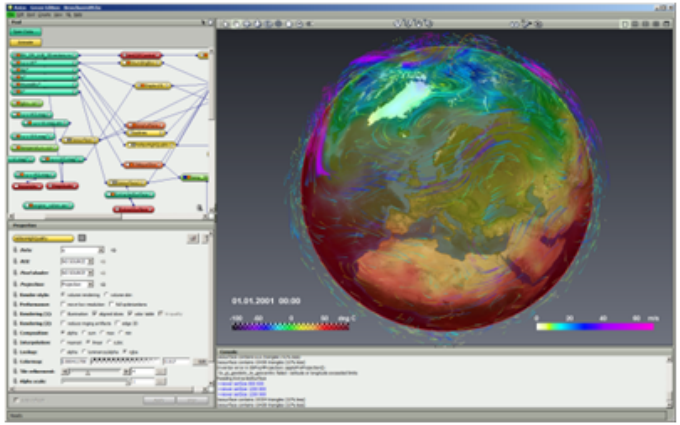


- 21 GPU nodes (2 Haswell/Boadwell, 256/512/1024 GB memory)
- 4 GPUs per node (2 dual Kepler/Maxwell)
- Software: NCL, ParaView, AvizoGreen, VAPOR

Hardware

- Linux nodes with XAS software tree:
  - avizo/9.0.1
  - paraview/5.0.1
  - simvis/3.4.4
  - vapor/2.5
  - idl/8.5
  - ncl/6.2.1
  - grads/2.0.2
  - ferret/6.9.3
  - gmt/5.1.2

Remote 3D Rendering



Local Workstation:
- Keyboard, Mouse
- Network Client (Receiver)
- 2D Graphics

Visualization Cluster:
- 3D Computergraphics (OpenGL)
- Network 3D-Server (Sender)
- Data- & Visualizationsoftware

News

- New supercomputer „Mistral" at DKRZ delivers particularly detailed regional climate simulations for Germany — Oct 05, 2015
- Preview: DKRZ at SC'15 — Sep 30, 2015
- Kick-off for ESiWACE and ESCAPE — Sep 29, 2015
- Allocations 2016 - request resources — Sep 18, 2015

More news...

Visualization on Mistral

Our new supercomputer Mistral has 12 GPU nodes, which can be used for 3D visualization, data analysis, and pre/post processing of data. This website explains how to reserve and access a GPU node, and how to run the 3D visualization software for the analysis of your data.

Access and reservation of a GPU node

Our supercomputer Mistral includes 12 GPU nodes with four GPUs in each node. Unlike the nodes in our visualization cluster Halo, these nodes are equipped with the same hard and software.

While we might later switch back to our web-based reservation system, currently the reservation has to be done by hand using a console on one of Mistrals login nodes. Furthermore, the vncserver has to be started by hand, as well as cleaned up after your work is done. To access Mistral, simply ssh into the machine.

```
somewhere:~> ssh user_name@mistral.dkrz.de
```

On Mistral, we have to allocate a GPU node. This is done using the SLURM command "salloc", in which you have to provide your account group ("-A <your project id>"), the number of nodes ("-N 1"), the maximum number of parallel tasks ("-n 24"), as well as the node type ("-p gpu"). Currently, the maximum time allowed is 4 hours and is set automatically. More information on SLURM and salloc can be found in our SLURM documentation.

After login to Mistral, you can allocate a GPU node and automatically ssh into the node reserved.

```
mistral:~> salloc -N 1 -n 24 -p gpu -A <project> -- /bin/bash -c 'ssh -X $SLURM_JOB_NODELIST'
salloc: Granted job allocation 284896

user_name@mg100's password:
```

Now you can start a VNC server to connect to the virtual desktop.

```
mg100:~> /opt/TurboVNC/bin/vncserver -geometry 1920x1200
Desktop 'TurboVNC: mg100:1 (user_name)' started on display mg100:1
```

At the first time, you have to supply a password that you later need to access your VNC session remotely. If everything is set, start a vncviewer in the console on your home computer, or using the gui of an application.

```
somewhere:~> vncviewer mg100.dkrz.de:1
```

After you are done with your work, please stop the VNC server and clean up the node.

```
mg100:~> /opt/TurboVNC/bin/vncserver -kill :1
```

Running visualization applications

Now a window opens, showing you the virtual X11 session that is running on your GPU node. On Mistral, all our visualization software is made available using modules. The command

News

- New supercomputer „Mistral" at DKRZ delivers particularly detailed regional climate simulations for Germany
  Oct 05, 2015
- Preview: DKRZ at SC'15
  Sep 30, 2015
- Kick-off for ESiWACE and ESCAPE
  Sep 29, 2015
- Allocations 2016 - request resources

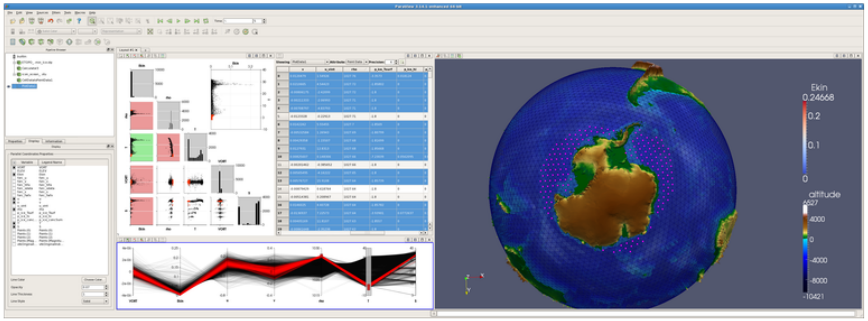
ParaView

Paraview is an open source visualization package that reads a variety of different data formats and lattices and implements the most common visualization techniques. More specifically, Paraview also reads netCDF files and supports different grids, so that it can be used to visualize climate and earth science data sets.

Paraview 4.1 is installed on all visualization nodes of Halo and can be started from the command line via 'paraview'. Older versions of Paraview can be started by appending the version number, such as 'paraview3.98'.

Paraview has come a long way and is used and developed by a very large community from a variety of different sciences. It is installed on DKRZ's Halo nodes since the end of 2012, and we have now prepared a little tutorial that will teach you how to use Paraview for the visualization of your own climate research data.

More general information on Paraview, along with some tutorial data can also be found online on the Paraview website.



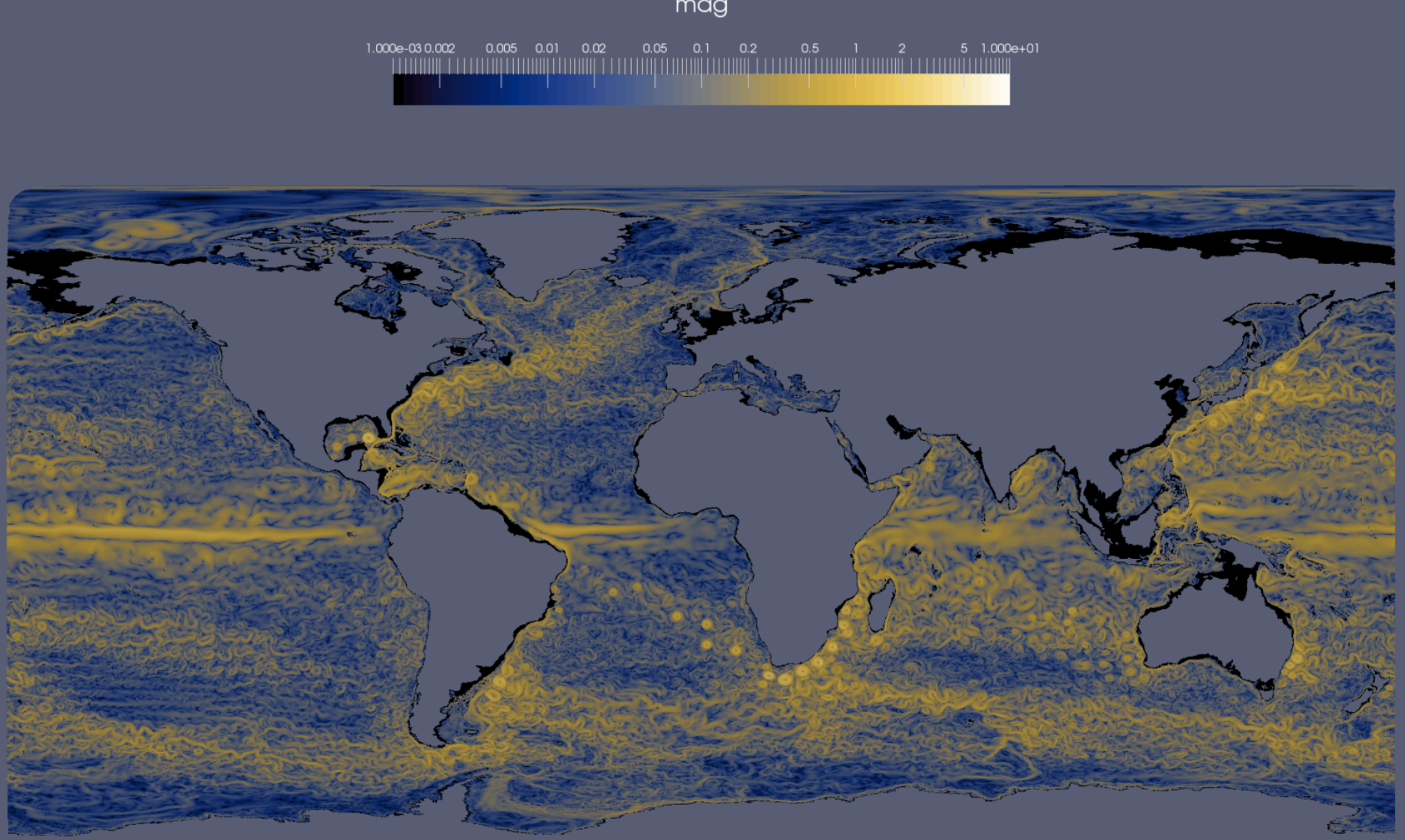
The above example shows a complex visualization of an ICON ocean data set using Paraview. The viewport on the right displays the data, the selection made, as well as the Earth's topography. The three viewports on the left hand side are used to specify the selection, based on a scatterplot matrix and parallel coordinates. These techniques are especially well suited for an in-depth data analysis and exploration.

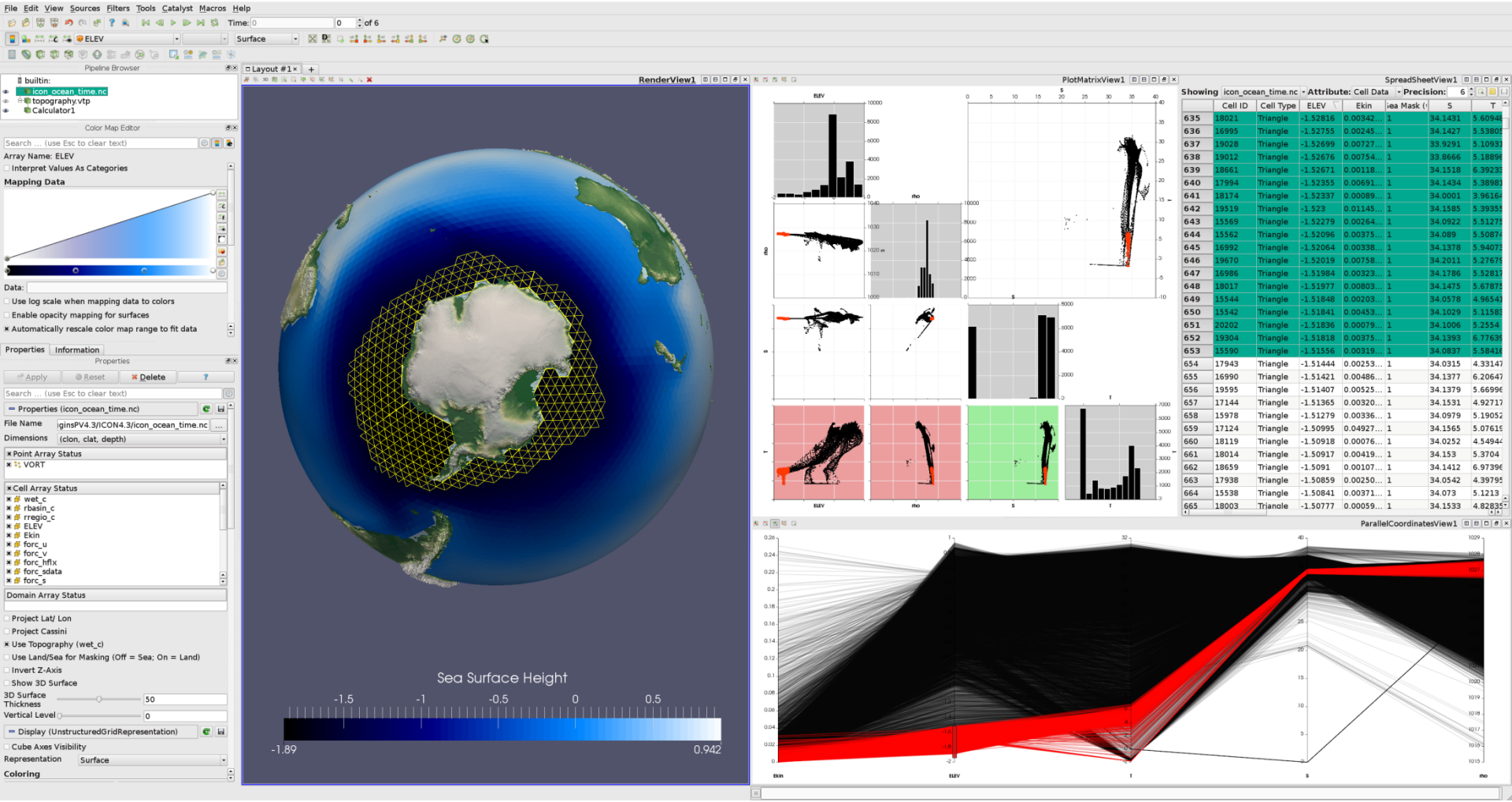
Paraview Tutorial

The final tutorial document will comprise 8 chapters and will be released at the end of the summer in 2014. Alongside, we will provide courses to teach Paraview in a hands-on setting. The first course will already start in December 2013.

Here is a glimpse of the content from the tutorial:

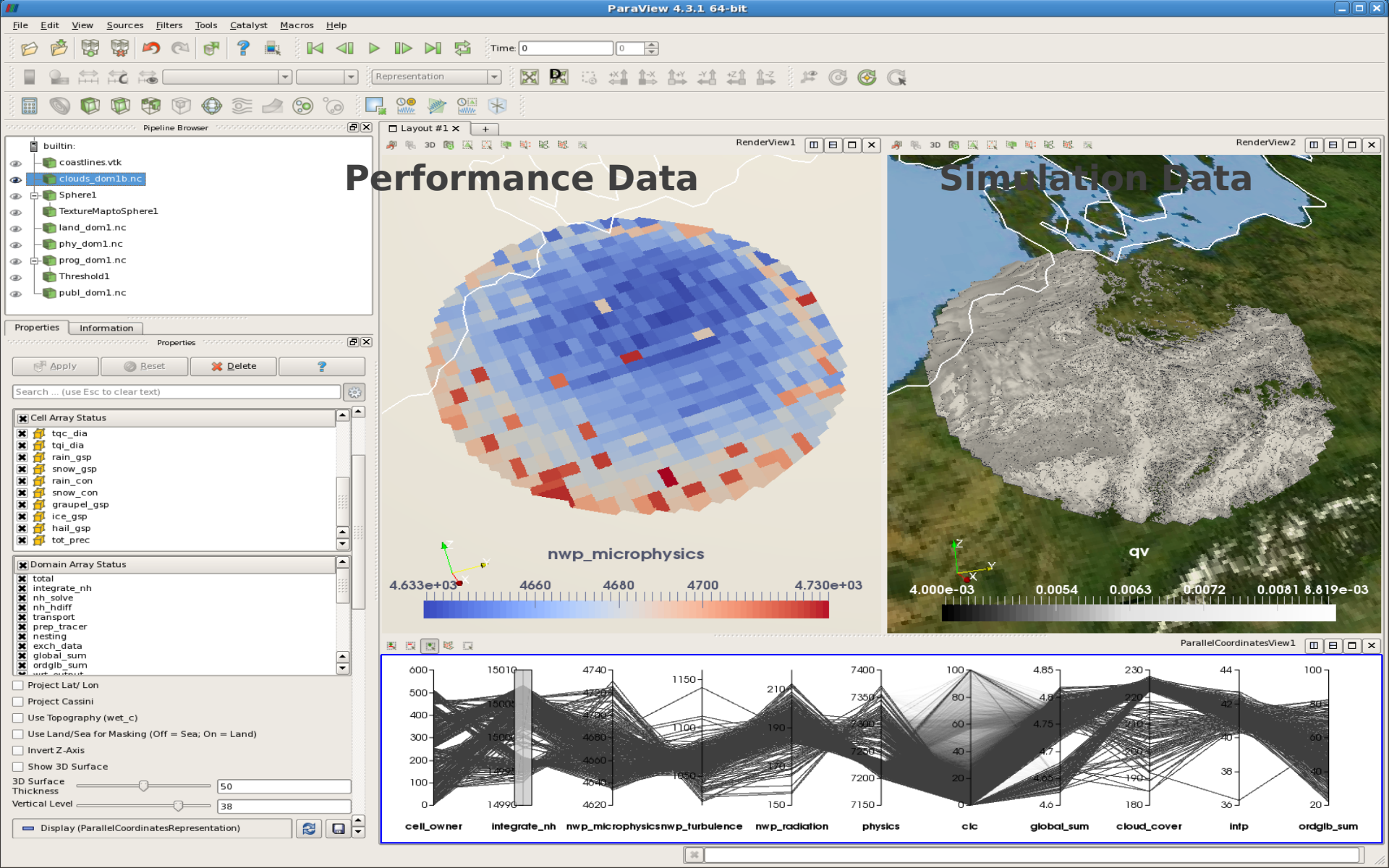
- Chapter 1 "Introduction and Overview" --- The first Chapter starts with an overview of Paraview and briefly explains the underlying visualization toolkit pipeline. The second part of this chapter concentrates on an introduction of the

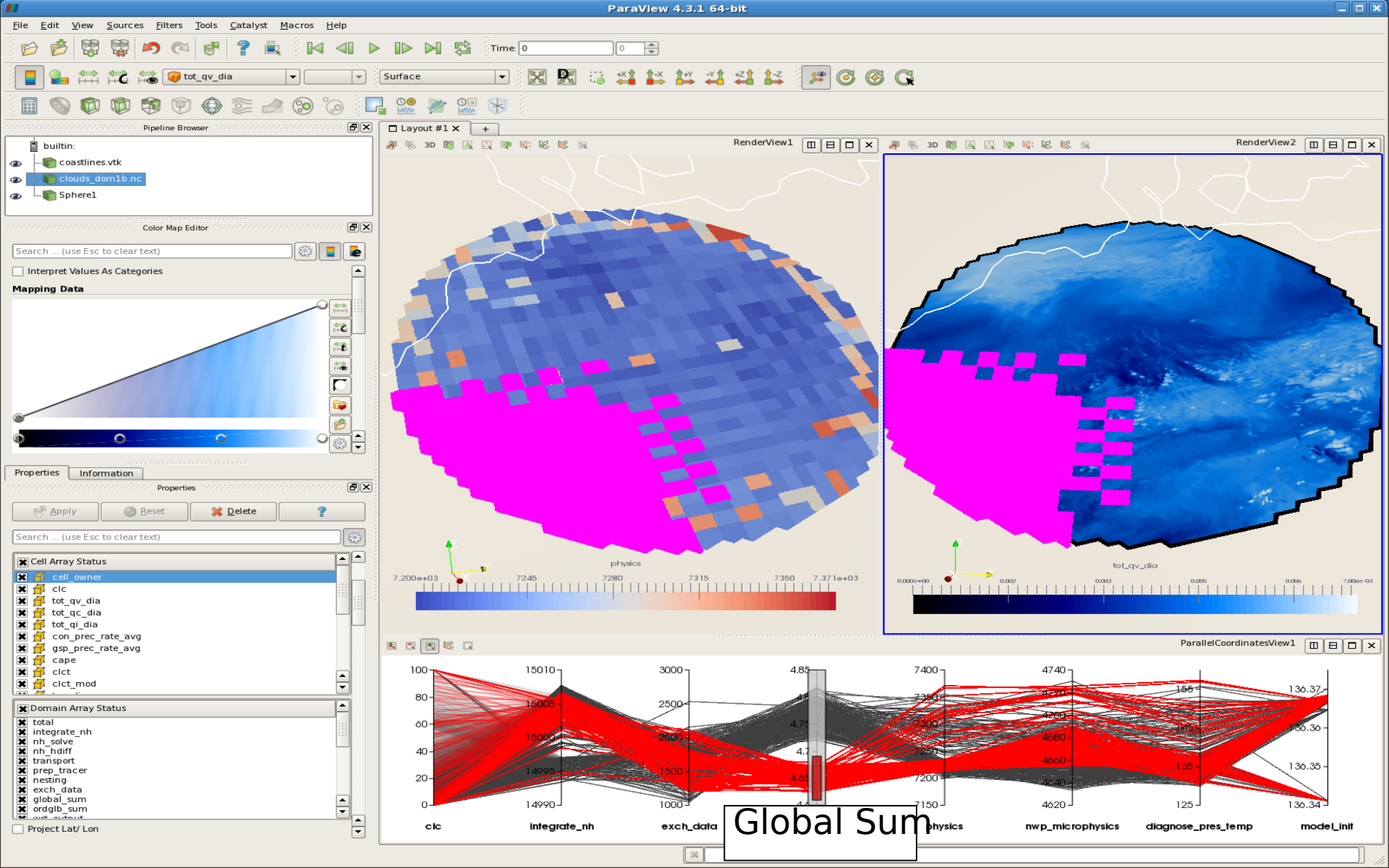

mag
1.000e-03 0.002 0.005 0.01 0.02 0.05 0.1 0.2 0.5 1 2 5 1.000e+01


File Edit View Sources Filters Tools Catalyst Macros Help
Pipeline Browser
builtin:
icon_ocean_time.nc
topography.vtp
Calculator1
Color Map Editor
Array Name: ELEV
Interpret Values As Categories
Mapping Data
Use log scale when mapping data to colors
Enable opacity mapping for surfaces
Automatically rescale color map range to fit data
Properties
Information
Point Array Status
VORT
Cell Array Status
wet_c
rbasin_c
rregio_c
ELEV
Ekin
forc_u
forc_v
forc_hflx
forc_sdata
forc_s
Domain Array Status
Project Lat/ Lon
Project Cassini
Use Topography (wet_c)
Use Land/Sea for Masking (Off = Sea; On = Land)
Invert Z-Axis
Show 3D Surface
Cube Axes Visibility
Representation
Surface
Coloring
RenderView1
Sea Surface Height
-1.5
-1
-0.5
0
0.5
-1.89
0.942
PlotMatrixView1
ELEV
rho
S
T
SpreadSheetView1
Showing icon_ocean_time.nc
Attribute: Cell Data
Precision:
Cell ID
Cell Type
ELEV
Ekin
S
T
ParallelCoordinatesView1
Ekin
ELEV
T
S
rho

Visualization of performance data

- ICON data with cell/point variables **and** performance data
- Select interesting areas and display results using parallel coordinates view
- Integrated into vtkCDIReader
- Requires domain decomposition and performance data


ParaView 4.3.1 64-bit
File Edit View Sources Filters Tools Catalyst Macros Help
Time: 0
Representation
Pipeline Browser
builtin:
coastlines.vtk
clouds_dom1b.nc
Sphere1
TextureMaptoSphere1
land_dom1.nc
phy_dom1.nc
prog_dom1.nc
Threshold1
publ_dom1.nc
Properties
Information
Apply
Reset
Delete
Search ... (use Esc to clear text)
Cell Array Status
tqc_dia
tqi_dia
rain_gsp
snow_gsp
rain_con
snow_con
graupel_gsp
ice_gsp
hail_gsp
tot_prec
Domain Array Status
total
integrate_nh
nh_solve
nh_hdiff
transport
prep_tracer
nesting
exch_data
global_sum
ordglb_sum
Project Lat/ Lon
Project Cassini
Use Topography (wet_c)
Use Land/Sea for Masking (Off = Sea; On = Land)
Invert Z-Axis
Show 3D Surface
3D Surface Thickness
50
Vertical Level
38
Display (ParallelCoordinatesRepresentation)
Layout #1
RenderView1
Performance Data
nwp_microphysics
4.633e+03
4660
4680
4700
4.730e+03
RenderView2
Simulation Data
qv
4.000e-03
0.0054
0.0063
0.0072
0.0081
8.819e-03
ParallelCoordinatesView1
cell_owner
integrate_nh
nwp_microphysics
nwp_turbulence
nwp_radiation
physics
clc
global_sum
cloud_cover
intp
ordglb_sum


ParaView 4.3.1 64-bit
File Edit View Sources Filters Tools Catalyst Macros Help
Time: 0
tot_qv_dia
Surface
Pipeline Browser
builtin:
coastlines.vtk
clouds_dom1b.nc
Sphere1
Color Map Editor
Search ... (use Esc to clear text)
Interpret Values As Categories
Mapping Data
Properties
Information
Properties
Apply
Reset
Delete
Search ... (use Esc to clear text)
Cell Array Status
cell_owner
clc
tot_qv_dia
tot_qc_dia
tot_qi_dia
con_prec_rate_avg
gsp_prec_rate_avg
cape
clct
clct_mod
Domain Array Status
total
integrate_nh
nh_solve
nh_hdiff
transport
prep_tracer
nesting
exch_data
global_sum
ordglb_sum
Project Lat/ Lon
Layout #1
RenderView1
RenderView2
physics
tot_qv_dia
ParallelCoordinatesView1
clc
integrate_nh
exch_data
nwp_microphysics
diagnose_pres_temp
model_init
Global Sum

Login

https://www.dkrz.de/Nutzerportal-en/doku/vis/visualization-on-mistral

local:~> ssh **user_name**@mistral.dkrz.de

mistral:~> salloc -N 1 -n 24 -p gpu -A **<project>** -- /bin/bash -c 'ssh -X

mg100:~> /opt/TurboVNC/bin/vncserver -geometry **1920x1200**

local :~> vncviewer **mg100.dkrz.de:1**

mg100:~> cd /work/kv0653/ParaView-Workshop

mg100:~> module load paraview

mg100:~> paraview