

# GRASS 3D Workshop – 3D data visualization with VTK

Sören Gebbert<sup>1</sup>

<sup>1</sup>Technical University Berlin  
Hydrogeology Group  
Institute of Applied Geosciences

**FOSS4G2006 Workshop**



# Table of Contents

- 1 VTK and ParaView
  - What is VTK
  - How to use ParaView
- 2 Raster map export and visualization
  - How to export with r.out.vtk
  - Raster maps in ParaView
- 3 Vector map export and visualization
  - How to export with v.out.vtk
  - Vectors maps in ParaView
- 4 Volume map export and visualization
  - How to export with r3.out.vtk
  - Volume maps in ParaView



# The Visualization ToolKit (VTK)

- software system for 3D computer graphics, image processing and visualization
- multi-platform and supports Windows, several Unix's and MacOS X
- written in C++ and bindings for Python, Tcl/Tk and Java available
- open source and freely available from <http://www.vtk.org>



# Why do visualization with VTK

- the most sophisticated visualization toolkit available on the market
- provides leading edge data processing and visualization capabilities
- supports all types of raster, vector and volume data implemented in GRASS
- is actively developed and has an advanced software design
- easy to implement visualization applications with VTK (supports rapid prototype development)



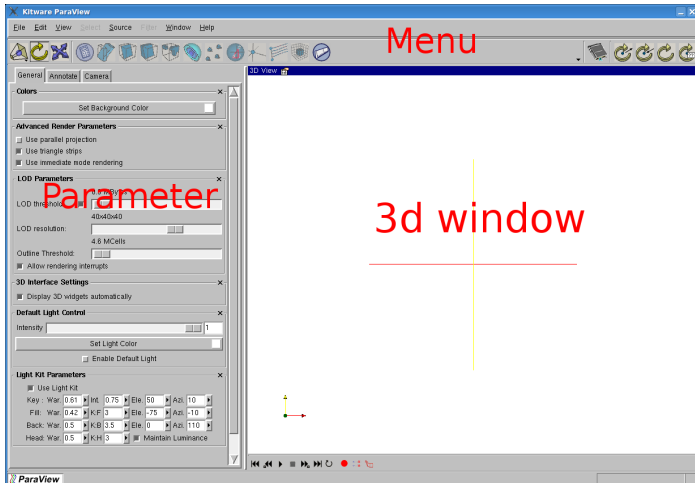
# ParaView

- multi-platform visualization application based on VTK
- designed to visualize large data
- provides many tools of VTK for data processing and visualization
- has a flexible and intuitive user interface
- open source and freely available from  
<http://www.paraview.org>

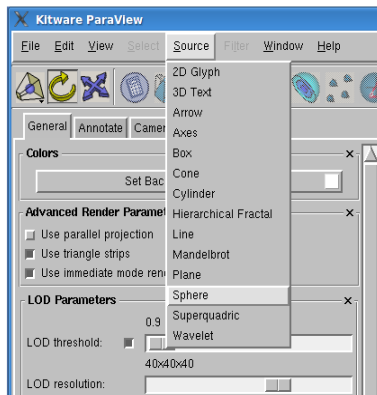
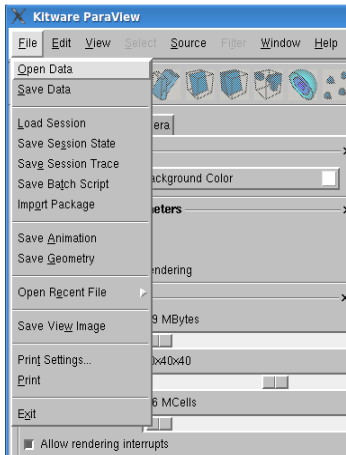
Please start ParaView by typing: `paraview`



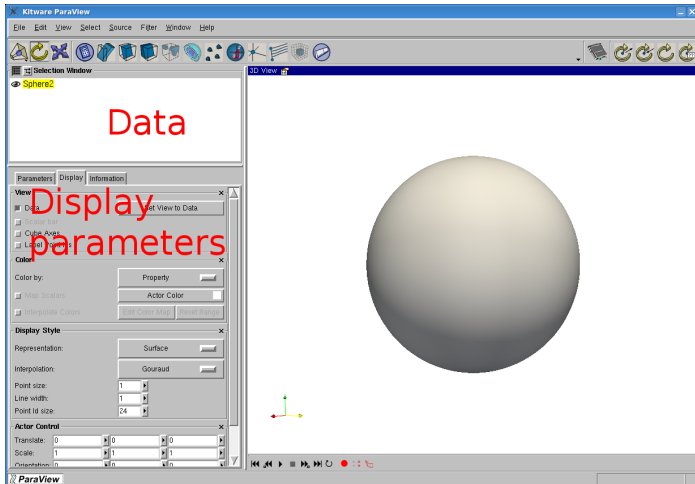
# ParaView



# Data import and creation



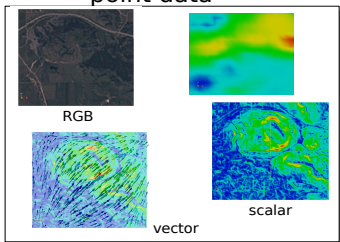
# Display



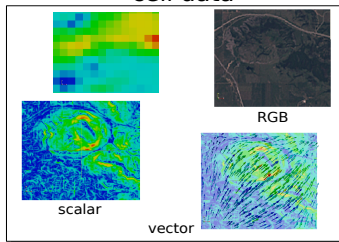
# Features of r.out.vtk

## r.out.vtk

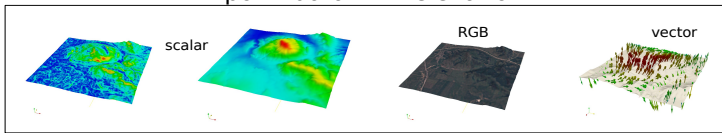
point data



cell data



point data with elevation

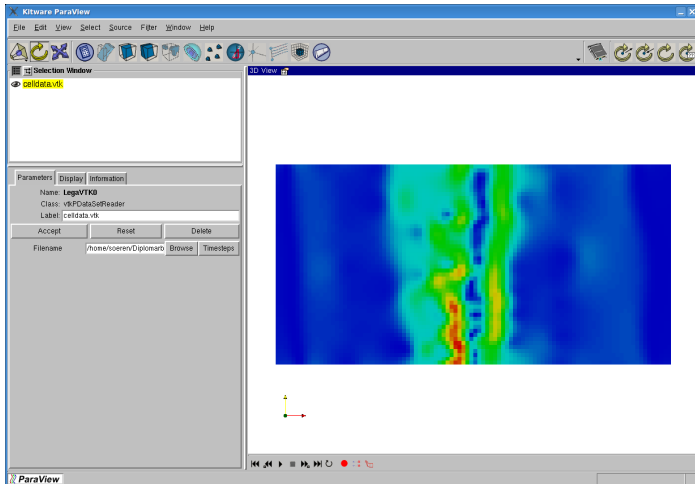


# How to use r.out.vtk

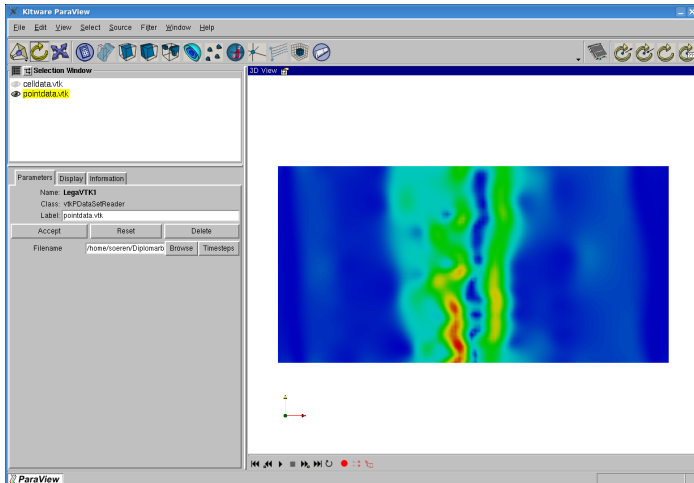
- **Exporting cell data:** `r.out.vtk in=slope out=celldata.vtk`
- **Exporting point data:** `r.out.vtk -p in=slope out=pointdata.vtk`
- **Exporting several data with elevation:** `r.out.vtk in=slope,aspect,elevation elevation=elevation out=elev.vtk`



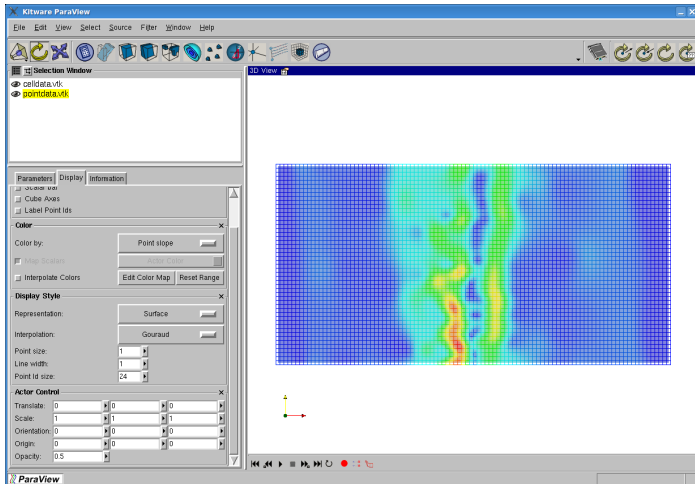
# Cell data



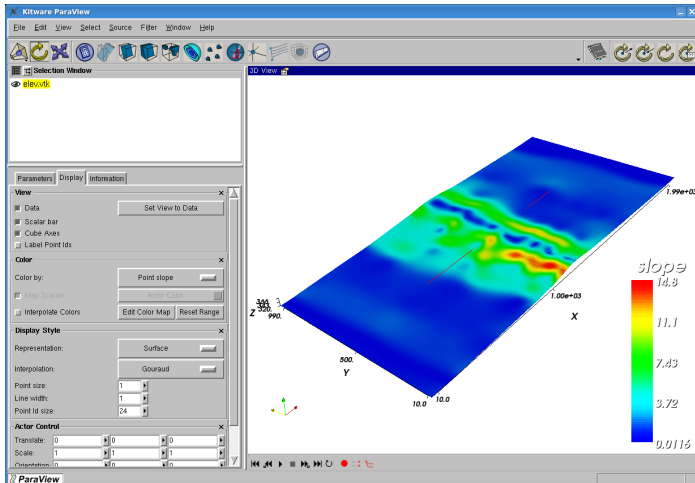
# Point data



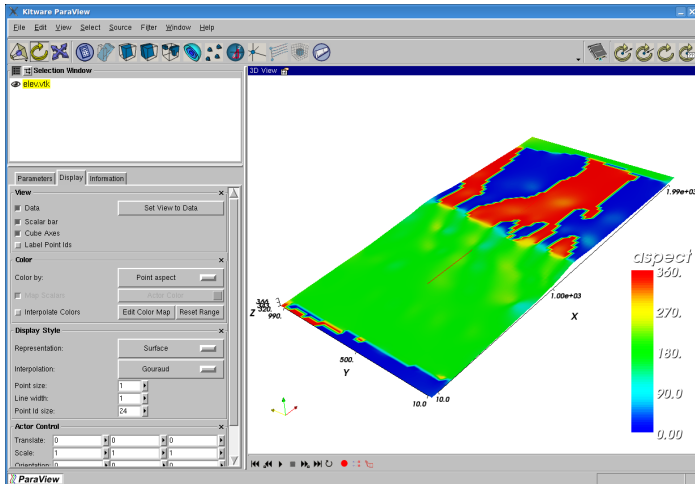
# Point and Cell data together



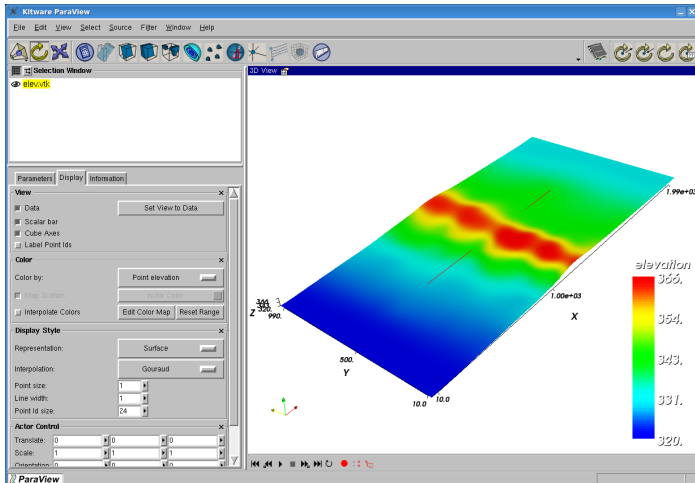
# Elevation and data



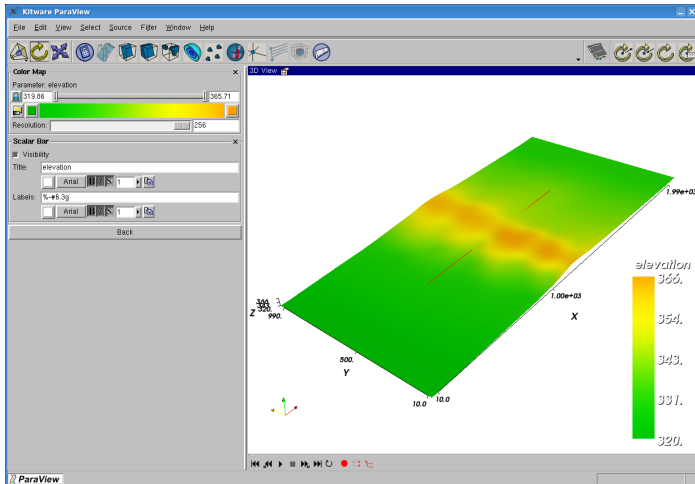
# Elevation and data



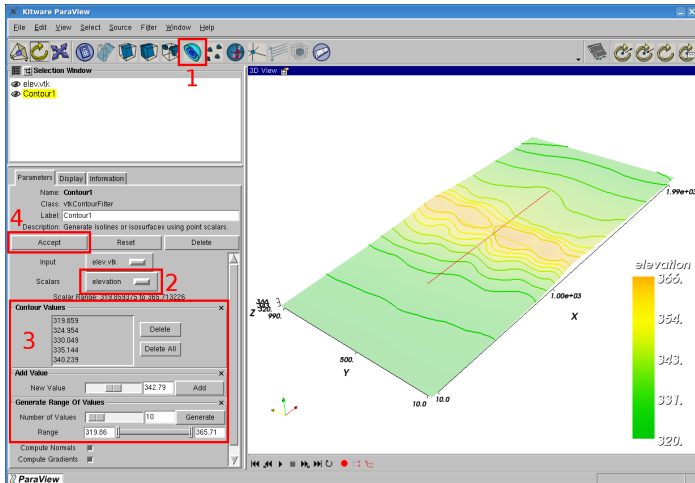
# Elevation and data



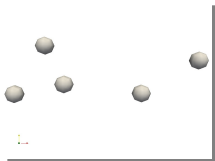
# Elevation color table



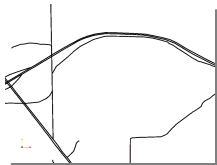
# Contouring



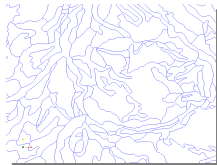
# Features of v.out.vtk



Points



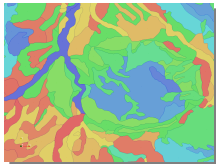
Lines



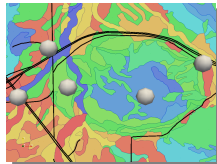
Boundaries

## v.out.vtk

Areas



Mixed



Faces

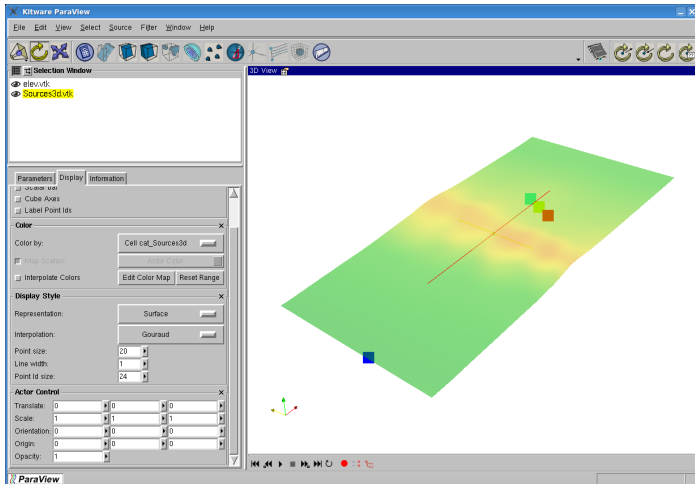


# Export data with v.out.vtk

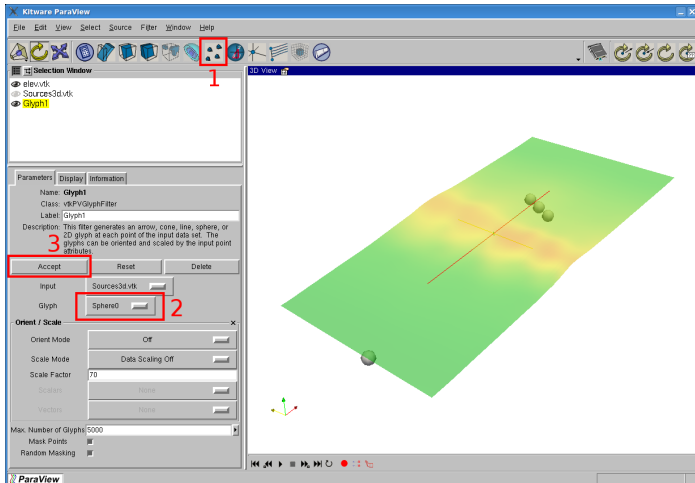
- **Exporting vector points:** `v.out.vtk input=Sources3d output=Sources3d.vtk`
- **Exporting vector lines:** `v.out.vtk input=roads3d output=roads3d.vtk type=line`
- **Exporting polygonal data**
  - **trees:** `v.out.vtk input=trees3d output=trees3d.vtk type=line,face`
  - **buildings:** `v.out.vtk input=industry3d output=industry3d.vtk type=face`



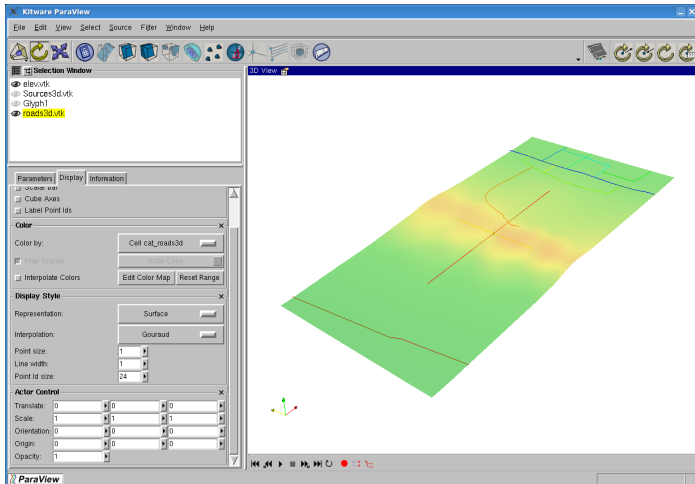
# Vector points



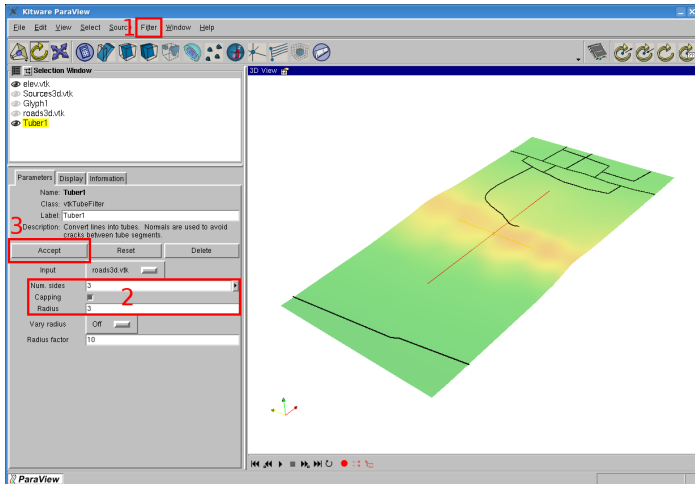
# Using the *Glyph* filter



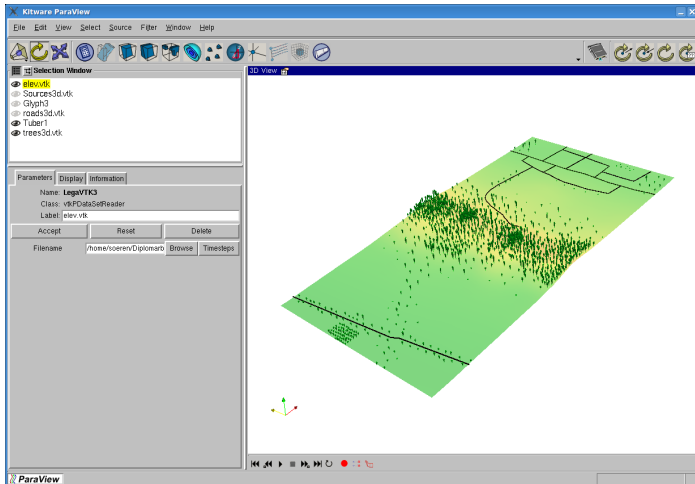
# Vector lines



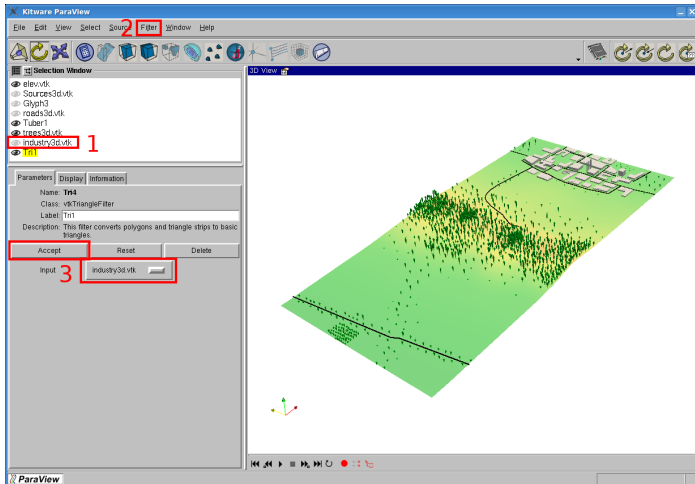
# Using the *Tube* filter



# Trees



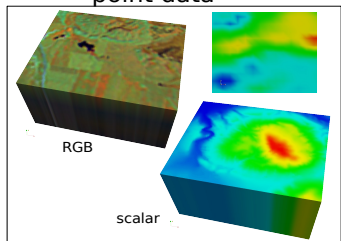
# Buildings and *Triangulate* filter



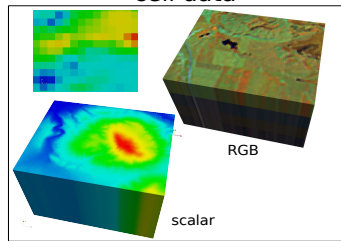
# Features of r3.out.vtk

## r3.out.vtk

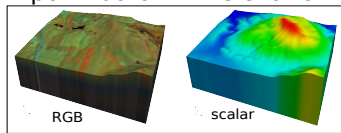
point data



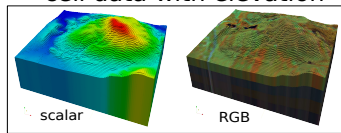
cell data



point data with elevation



cell data with elevation

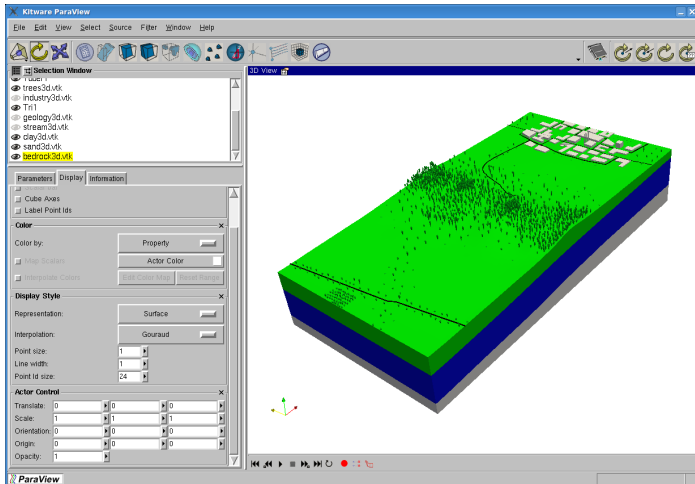


## Export scalar data

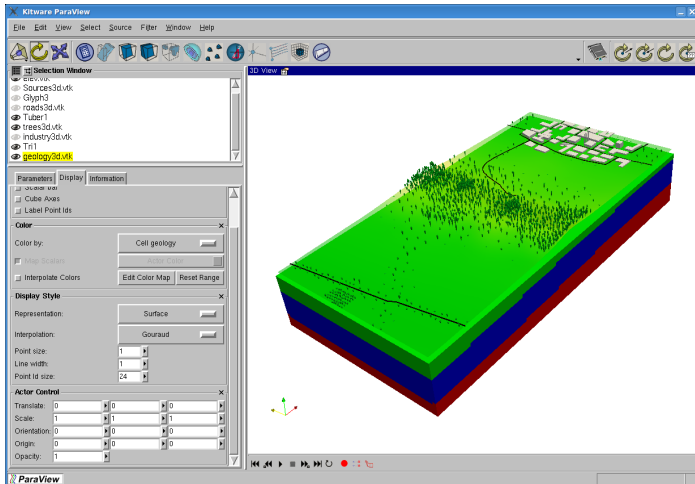
- **Exporting cell data:** `r3.out.vtk in=geology out=geology3d.vtk`
- **Exporting point data:** `r3.out.vtk -p in=Boundaries,ResultStream out=gw3d.vtk`
- **Exporting elevation data**
  - **reduce the z-resolution** `g.region tbres=150`
  - `r3.out.vtk -sp top=elevation bottom=border_sand_clay out=clay3d.vtk`
  - `r3.out.vtk -sp top=border_sand_clay bottom=border_bedrock_sand out=sand3d.vtk`
  - `r3.out.vtk -sp top=border_bedrock_sand bottom=bottom out=bedrock3d.vtk`



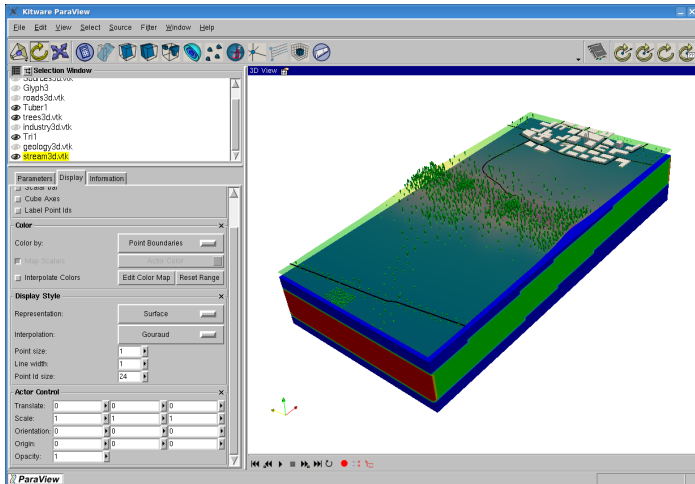
# Elevation data



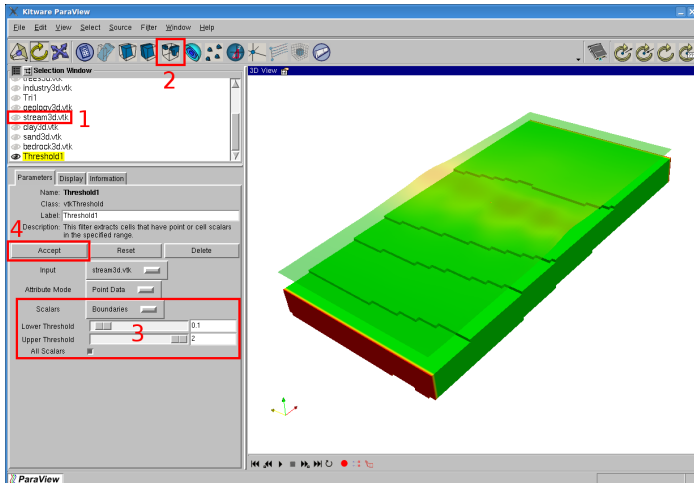
# Cell data



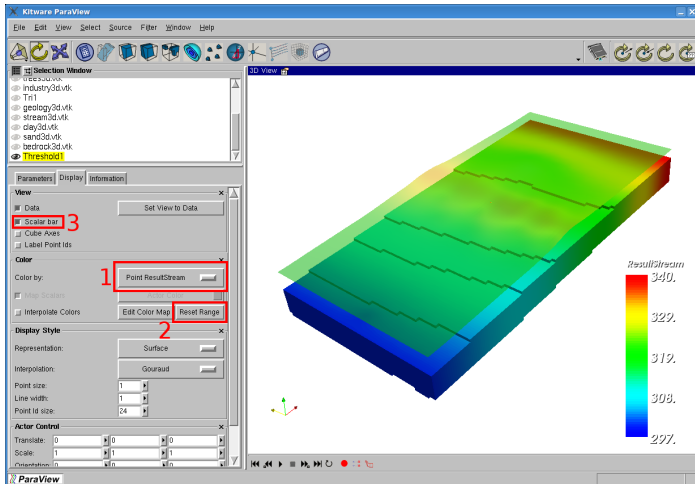
# Point data



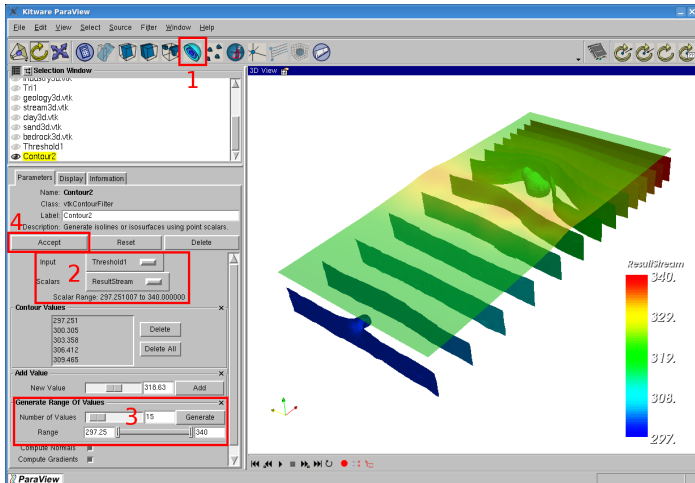
# Data extraction



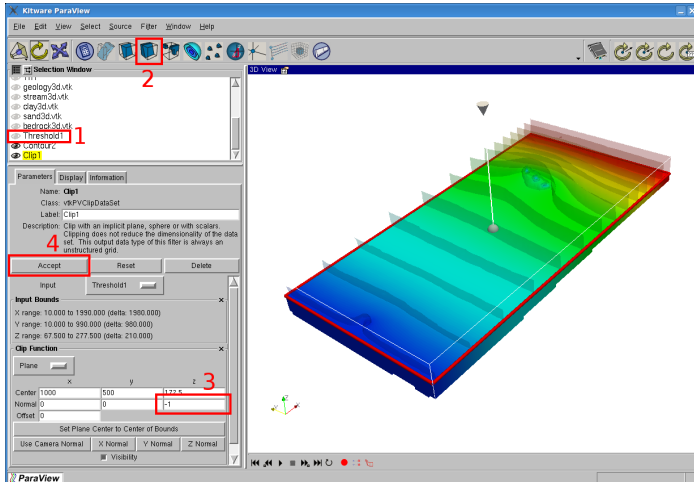
# Data extraction



# Isosurfaces



# Clipping

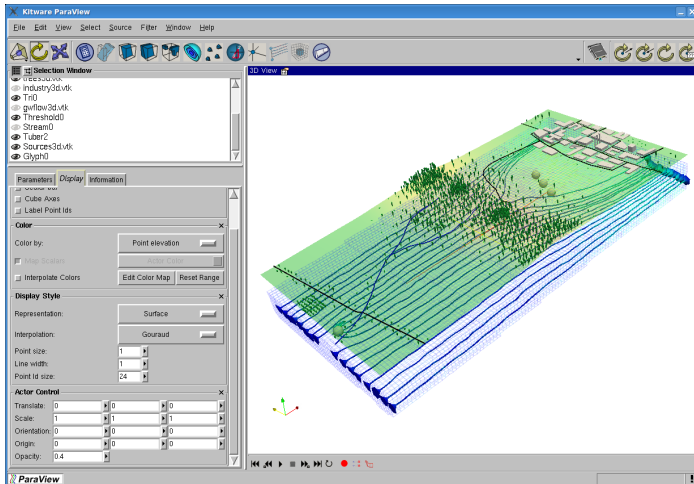


## Export 3d vector data

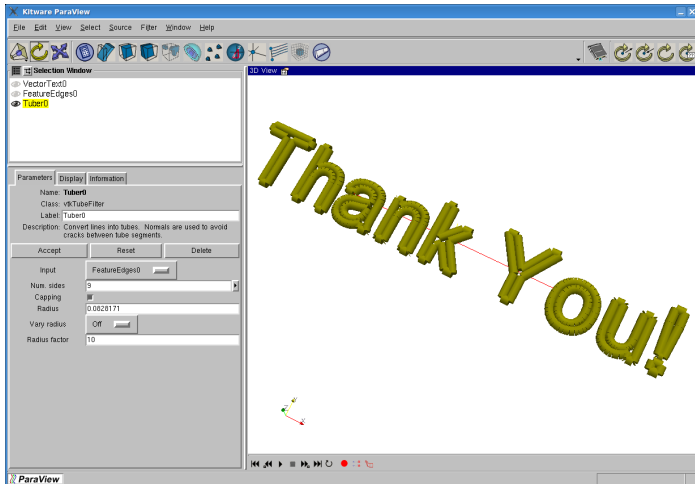
- Set the default region: `g.region -dp3`
- Exporting vector data: `r3.out.vtk -p`  
`in=Boundaries`  
`vectormaps=ResultStreamVector_x,`  
`ResultStreamVector_y, ResultStreamVector_x`  
`out=gwflow3d.vtk`



# All together with stream lines



# The End



# The End

Sören Gebbert

soerengebbert@gmx.de

[www-pool.math.tu-berlin.de/~soeren/grass/modules](http://www-pool.math.tu-berlin.de/~soeren/grass/modules)

Technical University Berlin  
Hydrogeology Group  
Institute of Applied Geoscience

