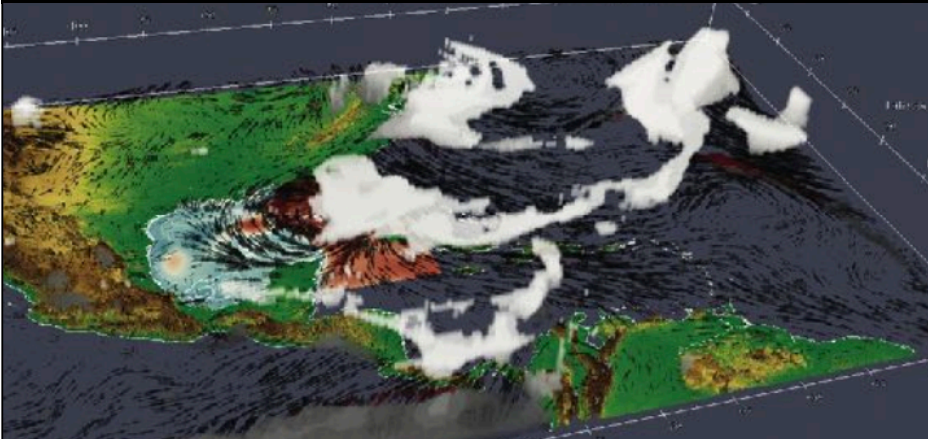


Master Project: Visualization Toolkit Evaluation



University of
Zurich^{UZH}



with C++. Knowledge about scientific visualization is not mandatory, and you will learn it during the project.

Work Load

- 20% theory
- 50% implementation
- 30% testing

Student Project Type

This topic can be done as a Master Project with a group of 2-3 persons. Detailed task requirements and the final goal will be specified depending on the number of students.

Supervision

Prof. Dr. Renato Pajarola
Haiyan Yang (assistant)
Alireza Amiraghdam (assistant)

Contact

Write an E-Mail to haiyan@ifi.uzh.ch

Reference

- VTK: <https://www.vtk.org/> and <https://en.wikipedia.org/wiki/VTK>
- ParaView: <https://www.paraview.org/>
- Visit: <https://wci.llnl.gov/simulation/computer-codes/visit/>

Context

The Visualization Toolkit (VTK) is an open source software that is widely used in data visualization, especially for manipulating and displaying scientific data. It's also the basis of many advanced visualization applications such as ParaView, VisIt, MayaVi, etc.

Current State

The VMML group has a spacious lab in IFI where a computer graphics cluster and a display wall are always ready for testing and research. Currently we have demonstrated the usefulness of the visualization toolkits on single computers and would like to evaluate the possibility of extending these VTK based applications into display walls and to explore advanced interactive visualization on computer graphics cluster and display wall.

Assignment

In this project, three main tasks are focused:

- Evaluate several different visualization toolkits (e.g. VisIt and ParaView) for their usability and deployability on a computer graphics cluster and display wall.
- Evaluate and demonstrate the feasibility of integrating custom graphics rendering or data visualization into the visualization toolkit's output, that is, combine toolkit display with separately generated image data.
- Test and deploy the 3D images rendered by one of the VTKs onto a stereo display simultaneously.

Requirements

The implementation and testing will mainly be Python, but you should also be familiar

