Pre-/Postprocessing and Remote Visualization

May 19, 2015
Dr. Herwig Zilken
Our Offering

- Special hard-/software infrastructure for
  - preprocessing, e.g. data conversion for simulation input data
  - postprocessing, e.g. data analysis with mathematical software of simulation output
  - remote data visualization

- Now: Visualization Cluster JUVIS
- Later: Partition of JURECA (Visualization Nodes)
JUVIS: Technical Data

- One login node (zam1164.zam.kfa-juelich.de, alias juvis.zam.kfa-juelich.de)
- 16 nodes for data processing and rendering:
  - 2 quad core 3.00 GHz Intel Xeon, 16 GB main memory
  - 8 nodes with Nvidia Quadro FX 4800 GPU
  - juvisn01.zam.kfa-juelich.de ... juvisn16.zam.kfa-juelich.de
- 10 Gbit/s Myrinet internal network for MPI
- One fileserver with 7.5 TB raid system
- Connected to GPFS of JUST (mounted on /gpfs, e.g. /gpfs/homea, ...)

- See http://www.fz-juelich.de/ias/jsc/EN/Expertise/Support/Visualization/ScientificVisualization/JUVIS/_node.html
JURECA Visualization-Nodes: Technical Data

- 12 Visualization-Nodes total: 10 soon, 2 later
- 2x Intel Xeon CPUs (Haswell), 2.5 GHz
- 12 cores/CPU, 24 cores/node, 2 way SMT/Hyper Threading, 48 logical cores
- 2 GPUs Nvidia Tesla K40 per node, 12 GB RAM on each card
- soon: 10 nodes with 512 GB RAM
- later: 2 nodes with 1024 GB RAM (when RAM modules are available)

- connection to vis-nodes via login-nodes and ssh tunnels (for security reasons)
Installed Software on JUVIS (so far)

- ParaView
- Visit
- Interactive Data Language IDL
- PyMol Molecular Viewer
- Visual Molecular Dynamics VMD
- Vapor
- Octave (similar to Matlab)
- Libs for NetCDF (cdo, nco), HDF5
- GNU and Intel compiler
Remote Rendering on JUVIS with VNC/VirtualGL

- Remote rendering with VNC (virtual network client) together with VirtualGL
- VNC/VirtualGL is a good solution for many common OpenGL applications, e.g. IDL, PyMol
Parallel ParaView

- ParaView has three main components:
  - Client
  - Data Server
  - Render Server

![Diagram of ParaView components]
Parallel ParaView (continued)

- ParaView can be started in non parallel (standalone) mode:
  - All three components in one single process
    - command: paraview

- ParaView can be started in two parallel modes:
  1. Local client and parallel server (data server and render server in one process) → used on JUVIS!
    - command on local client: paraview
    - command on remote cluster: `mpiexec -x -n <num_processes> pvserver`
Parallel Paraview (continued)

2. Local client and parallel data and render server
   (data server and render server may run on different machines)
   - command on local client: `paraview`
   - command on data processing server:
     `mpiexec –n <num_dataserver> pvdataserver`  
     `–m=machines.pvx`
   - command on render server:
     `mpiexec –n <numrenderserver> pvrenderserver`  
     `–m=machines.pvx`

   (num_dataserver >= num Renderserver)
Remote Rendering Modes

• Parallel ParaView server (data and render server) on JUVIS, ParaView client on local workstation
• **Server side rendering** and **client side rendering** possible
How to get an Account on JUVIS

Internet:

Enter www.fzj.de/jsc

and click on

“JSC online, application forms” and on “Computing time allocation”

there, select the correct link for user accounts
How to get an Account on JUVIS

Later, when you apply for an account, select JUVIS as system