

Geoinformationssysteme und der GFZ-Compute Cluster

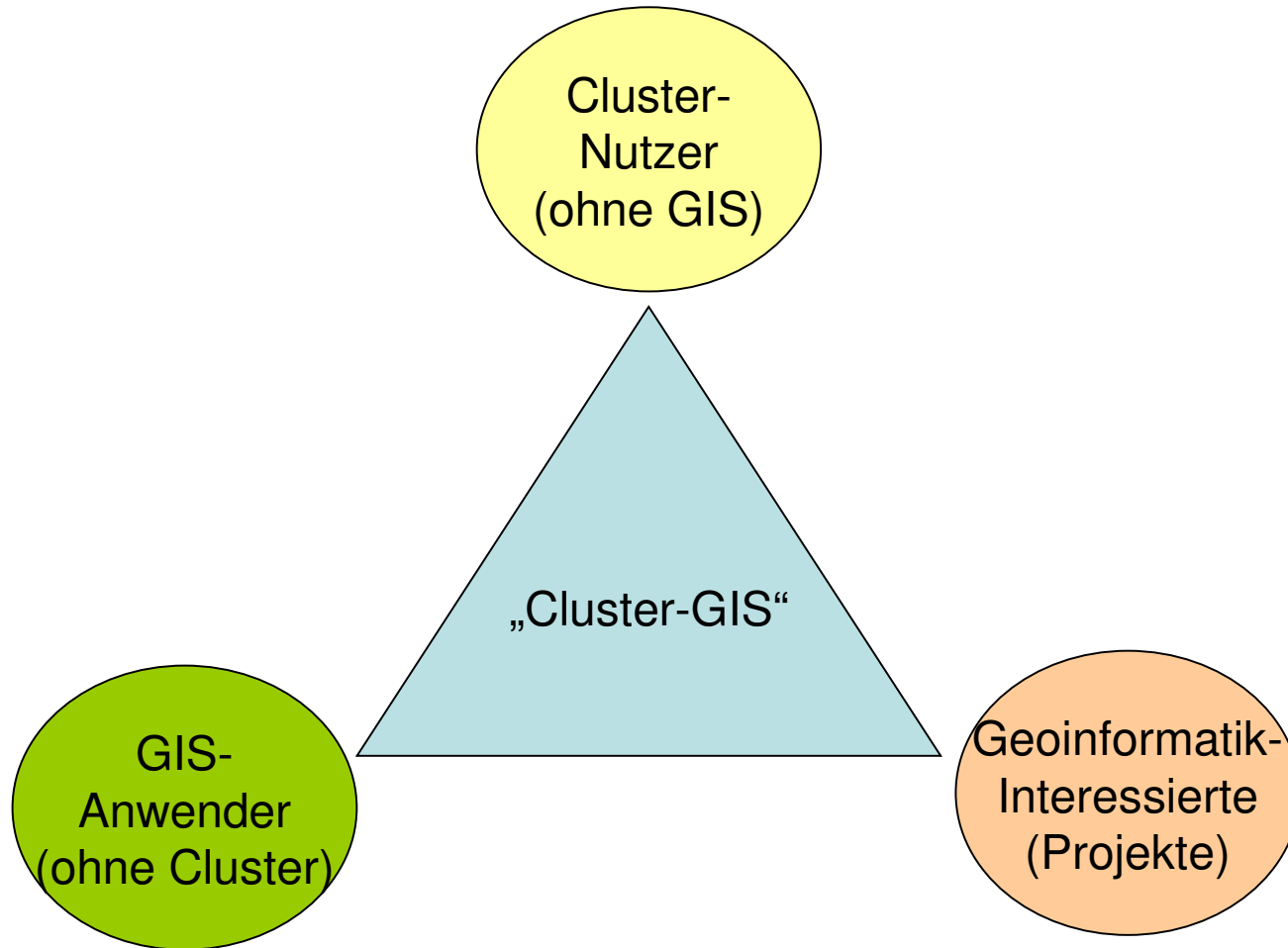
Ein Werkstattbericht: Erfahrungen
mit GRASS GIS

Peter Löwe

Motivation

- Alltagserfahrung:
 - Es gibt wiederkehrende GIS-Aufgaben, die den PC-Einzelplatzbetrieb übersteigen.
- Motivation dieses Vortrags:
 - Bestehende Infrastruktur nachhaltig nutzen
 - Erfahrungen/Wissen weitergeben
 - Ausprobierte Fettnäpfchen und Betonwände benennen
 - Einsatzmöglichkeiten vorstellen
- Entwicklungspotentiale:
 - Modellintegration
 - Visualisierung
 - Web Processing Services (WPS)

Zielgruppen



Der Compute-Server („Cluster“)

„A group of computers (hosts) running LSF that work together as a single unit, combining computing power and sharing workload and resources. A cluster provides a single-system image for disparate computing resources“

(LSF Cluster-Definition)

- Administration: Marina Köhler
 - Tel +49 (0)331 288 1682
 - E-Mail koe@gfz-potsdam.de
- Infos: <http://www.gfz-potsdam.de/portal/-?%24part=CmsPart%26%24event=display%26docId=1333664%26cP=DRZ.content.detail> (siehe auch Wiki)

Cluster Hardware

- One dual CPU master node with 4 GB main memory, 2 x 80 GB hard disk
- 32 dual CPU compute (slave) nodes with 2GB main memory , 80 GB hard disk
- One fileserver as failover for the master
- Data disks: ca. 2 TB mounted via NFS to all nodes
- all nodes are connected with a Gigabit Ethernet Network

Cluster Software

- Operating System: SUSE Linux Enterprise Server 9
- Compiler : gcc, Intel, Portland (all 32 und 64 Bit) for C, C++, Fortran95
- Tools: [Etnus Totalview Debugger](#), Intel Trace Analyzer
- Math.Libraries
- Parallel Environment: MPICH, LAM MPI
- Queuing System: Torque (OpenPBS)
- Scheduler: MAUI

„Industrial Strength GIS“

- Speicherintensiv:
 - Viele große Datensätze (Input/Output)
- Lange Berechnungen
- Prozessketten
- Parallele Prozesse
- Kein Ersatz für lokale GIS auf dem Arbeitsplatzrechner.

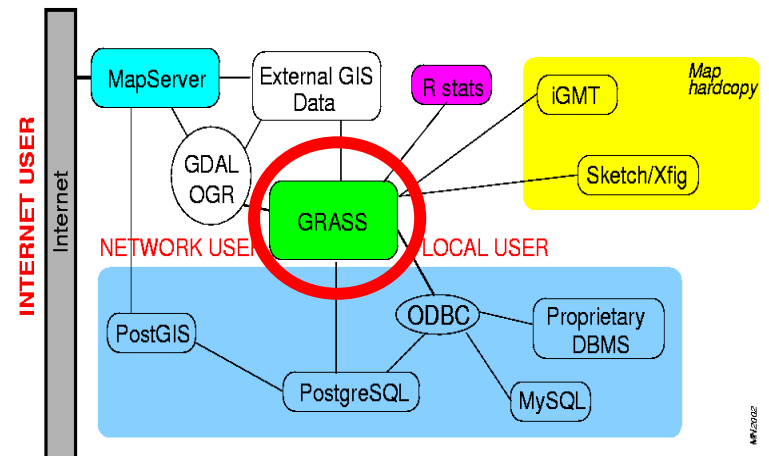
Open Source GIS

- Open Source = „Quelloffen“
 - Programmquellcodes können
 - eingesehen,
 - erweitert und
 - weitergegeben werden.
 - Freie Lizenzen:
 - Keine Kosten,
 - keine „Lizenzschlüssel“ und andere Ärgernisse,
 - Beliebig viele Instanzen, überall

Werkzeug- Kombinationen

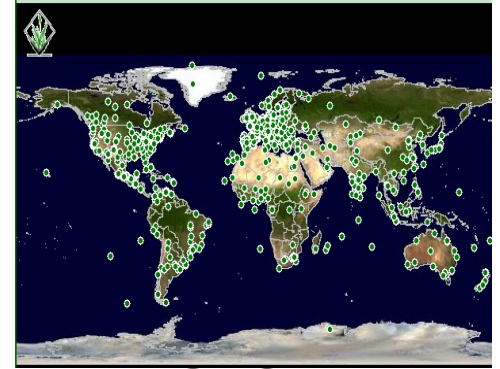
Quelloffene Werkzeuge können leicht miteinander verschränkt werden:

- **GDAL**: Rasterformat Ex/Import
- **OGR**: Vektorformate Ex-/Import
- **R**: Statistik/Modellierung
- **Quantum GIS**: hübsche GUI
- **RDBMS**: MySQL, PostGIS, etc.
- **Paraview** (3d/4dVisualisierung)





GRASS GIS



- Ältestes und größtes Free Open Source GIS Projekt.
- Große Community, diverse Mailinglisten
- Raster + Vektoren + Voxel/Volumen
- Ca. 350 Programmmodule
- Quellcode in C/C++, OSGEO zertifiziert
- Scriptbar in bash/Perl/Phython,etc.

„Eine große Menge von unabhängigen Programmmodulen, die auf eine einheitliche Datenhaltung zugreifen“

Reality Check: Software-Verfügbarkeit

- R (Statistik/Modellierung) ist verfügbar
- GRASS war schon verfügbar
- Optionen zukünftiger Installationen:
 - Vorkompilierte Binärpakete nicht anwendbar:
 - Bauen aus dem Quellcode ist notwendig
 - Neue zentrale Installation wünschenswert
 - Alternative: Lokale Installation auf User-Ebene nötig (DIY).
 - GUI funktioniert (bisher) nicht: Kein „ssh -x“

Beispiel: Softwarekonfiguration

Bearbeiten

Configure-Skript

Das Configure-Skript kann mit folgenden Parametern genutzt werden:

```
./configure
--enable-64bit
--with-libs=/usr/lib64
--without-tcltk
--without-opengl
--with-cxx
--with-postgres
--with-mysql
--with-sqlite
--with-odbc
--with-blas
--with-lapack
--with-freetype
--with-postgres-includes=/usr/include/postgresql
--with-mysql-includes=/usr/include/mysql
--with-freetype-includes=/usr/include/freetype
```

Danach noch make / install ausführen → voila !

Aus noch unbekanntem Gründen kann die TclTK-GUI nicht gebaut werden (ist beim Betrieb auf dem Cluster aber auch nicht nötig).

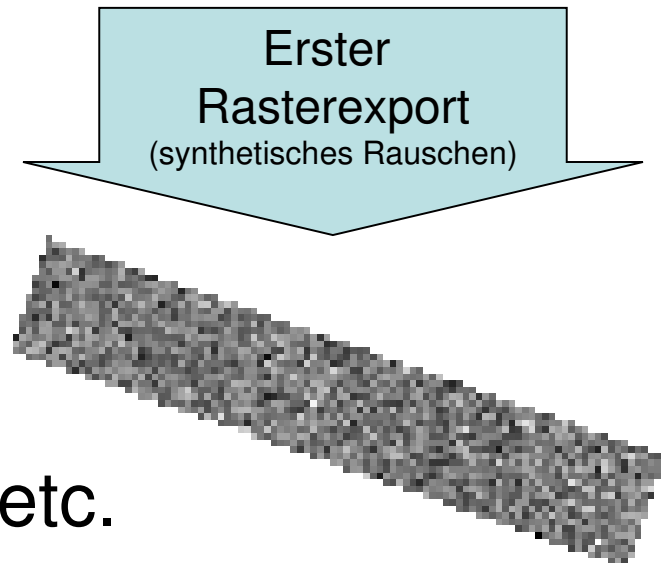
Sollte theoretisch funktionieren, scheitert an der Realität:

```
./configure --enable-64bit --with-libs=/usr/lib64 --with-tcltk-includes=/usr/include
--with-tcltk-libs=/usr/lib64
```

Das Beispiel ist im WDC Terra Wiki verfügbar

Zugriff auf das Cluster-GIS

- Steuerung über die Kommandozeile:
 - SSH login (Windows: Putty)
 - Interaktive GIS-Nutzung
 - Bash/Perl/Python-Scripting
 - Cronjobs
- Export von Endprodukten:
 - Raster: GeoTIFF, PNG etc.
 - Vektoren: Shapefiles, GML, etc.
 - Daten: Datenbanken, XML



Beispiel-Session:

```
ploewe@tiger:~> grass63
```

```
tiger.gfz-potsdam.de - PuTTY
GRASS 6.3.0

DATABASE: A directory (folder) on disk to contain all GRASS maps and data.

LOCATION: This is the name of a geographic location. It is defined by a
co-ordinate system and a rectangular boundary.

MAPSET: Each GRASS session runs under a particular MAPSET. This consists of
a rectangular REGION and a set of maps. Every LOCATION contains at
least a MAPSET called PERMANENT, which is readable by all sessions.

The REGION defaults to the entire area of the chosen LOCATION.
You may change it later with the command: g.region
-----
LOCATION:  mdv_data_____ (enter list for a list of locations)
MAPSET:   2001_01_01_____ (or mapsets within a location)

DATABASE: /home/dc/ploewe/geodata/locations_____

AFTER
```

```
GRASS GIS

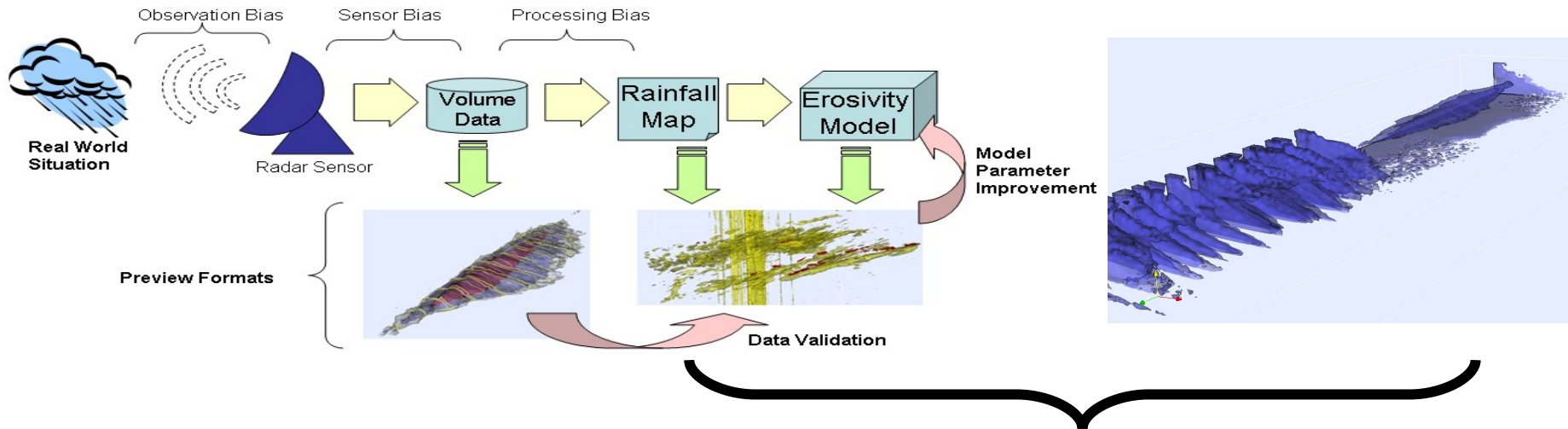
Welcome to GRASS 6.3.0 (2008)
GRASS homepage:                http://grass.osgeo.org/
This version running thru:     Bash Shell (/bin/bash)
Help is available with the command: g.manual -i
See the licence terms with:    g.version -c
Start the GUI with:            g.gui tcltk
When ready to quit enter:      exit

GRASS 6.3.0 (mdv_data):~ >
```

```
g.proj: error while loading shared libraries: libproj.so.0: cannot open shared object file: No such file or directory
```

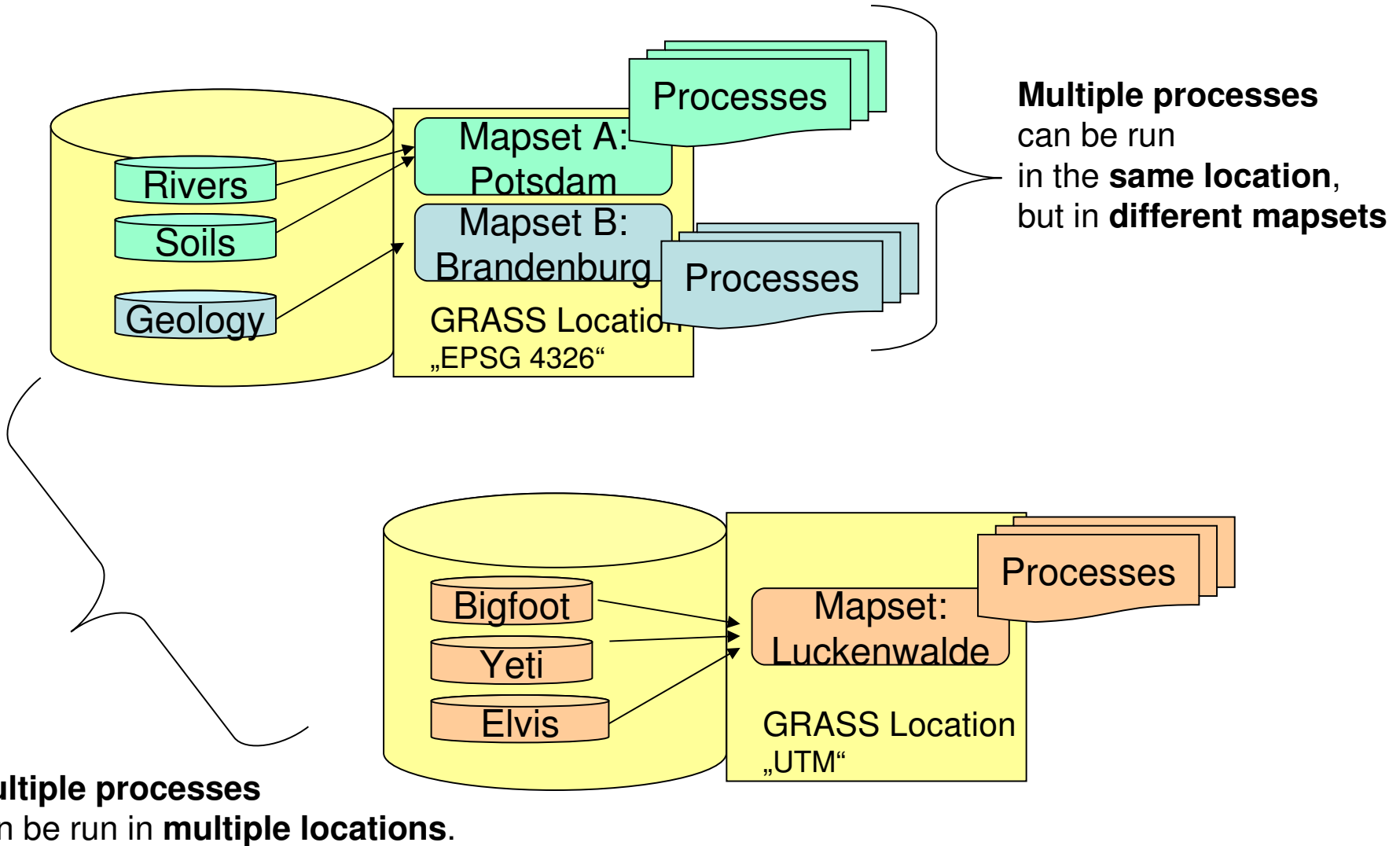
Anwendungsbeispiel

- Aufwändige Berechnungen für 3d/4d
Datenvisualisierung für Datenqualitätsaussagen:
 - Viele 3-d Datensätze werden in 2d Subräume kollabiert.
 - Die Mengen der 2d Datensätze werden zu einem 3d-Datensatz vereinigt



Prozessierung auf dem
ComputeServer/Cluster

GRASS GIS: Multiple Jobs



Bestehende Parallelisierungsansätze für GRASS GIS

Aus dem GRASS GIS Wiki:

- GDPE using OpenMP
- OpenMosix
- PBS
- SGE – SUN Grid Engine

(http://grass.osgeo.org/wiki/Parallel_GRASS_jobs)

Interner Wissensspeicher: <https://dokuwiki.gfz-potsdam.de/gis>

The screenshot shows a DokuWiki interface. At the top, there is a navigation bar with the text "[[wdc-terra:gdi:start]]" and "DOKUWIKI". Below this are buttons for "Diese Seite bearbeiten", "Ältere Versionen", "Letzte Änderungen", and "Suche". A search input field is also present. Below the navigation bar, the page title "Geodaten-Infrastruktur (GDI)" is displayed. The main content area contains a paragraph about geospatial data and a table of contents. A second table of contents is visible on the right side of the page, corresponding to the article "GRASS Prozessierung auf dem Linux-Cluster des RZ".

Geodaten-Infrastruktur (GDI)

Fas alle Daten, die am GFZ erhoben oder errechnet werden, haben einen Raumbezug ¹⁾. Um den Umgang mit diesem Raumbegriff in der Erfassung als auch in der Datenverarbeitung im Geodaten-Infrastrukturzentrum zu erleichtern, werden Einzelne Vorarbeiten zur Infrastruktur geographischer Daten dauerhaft genutzt. GIS und Bibliotheken für GIS in Betrieb ist umgestellt werden und veröffentlicht ³⁾.

GRASS Prozessierung auf dem Linux-Cluster des RZ

Es folgt die Beschreibung der bisher gemachten Erfahrungen zum Thema GRASS GIS auf dem Cluster des GFZ.

Bisher wurde GRASS jeweils auf der Portalmaschine ausgeführt. Spezielles Cluster-skripting sollte möglich sein, wurde aber noch nicht ausprobiert. Im GRASS GIS Wiki finden sich Beschreibungen zum Betrieb des GIS auf Clustern, jedoch (leider) nicht für unsere Clusterarchitektur. Positiv ausgedrückt wird damit Neuland betreten.

Inhaltsverzeichnis

- Geodaten-Infrastruktur (GDI)
 - Web Map Service (WMS)
 - Web Feature Service (WFS)
 - Web Coverage Service (WCS)
 - Gazetteer (WFS-G)

Inhaltsverzeichnis

- GRASS Prozessierung auf dem Linux-Cluster des RZ
 - Compilieren der Software
 - Configure-Skript
 - In the beginning was the command line
 - Neue Location anlegen (ohne die GUI zu benutzen)
 - Zusatzmodule einbinden
 - Neues Mapset in bestehender Location bzw zwischen zwei mapsets schnell wechseln
 - Was fehlt
 - Autoprozessierung

GIS-Wiki:

<https://dokuwiki.gfz-potsdam.de/gis>

Externe Hilfe

- GRASS Verein Deutschland / OSGEO
 - Monatlicher Stammtisch im „Gleis 6“, Babelsberg
- Facebook Gruppe „GRASS makes me want to kill kittens“
- OSGEO Mailing Listen
- Schulungen rund um GRASS werden von verschiedenen Unternehmen angeboten

Ausblick

- Integration und Anbindung von anderen am GFZ verfügbaren Produkten:
 - GoCAD Integration
- Grid
- 3d Lab: Anbindung über bestehende Software bzw. Paraview

Vielen Dank für ihre
Aufmerksamkeit !

Kontakt:

Peter Löwe ploewe@gfz-potsdam.de

Jens Klump jklump@gfz-potsdam.de

<https://dokuwiki.gfz-potsdam.de/gis>