GRASS GIS in Grid Environment

22nd April 2009
ISGC2009
- **UNOSAT**
  - Grid projects
- **GRASS GIS**
  - Integration in Grid environment
- **Current Status**
- **Future Plans**
What is UNOSAT?

- UNOSAT is the UN Institute for Training and Research (UNITAR) Operational Satellite Applications Programme, implemented in co-operation with the European Organization for Nuclear Research (CERN).

- Supports early warning, crisis response, human rights, sustainable recovery, vulnerability reduction and local capacity building

What we do?

- Humanitarian: Rapid mapping in support to disaster management, relief & human rights

- Training and awareness raising: Customized, from user-perspective
Global Monitoring for Security and Stability

March 2004 - February 2008

- Satellite imagery storage, metadata processing
- Web interface
- Mobile phone user interface

CERN
IT Department
March 2004 - Present

PC based users, Mobile phones with GPS

Web-browser based user-interface, mobile phone application

Compression, processing and storage
Geographic Resources Analysis Support System
GRASS GIS

http://grass.osgeo.org/

Platforms
- Linux
- Mac
- Microsoft Windows

Features
- Data management
- Image processing
- Graphics production
- Spatial modeling
- Results comparable to those obtained with commercial software

Migration to OSGeo infrastructure
Full fledge OSGeo project
Version: **GRASS 5.4.1** released 26 July **2007**

Latest stable release: GRASS 6.4.0RC4 released 12 Apr 2009

**Additional Libraries needed for most operations**
- GDAL - Geospatial Data Abstraction Library
- PROJ.4 - Cartographic Projections Library

GRASS modules are designed under the Unix philosophy and hence can be combined using shell scripting to create more complex or specialized modules by a user without knowledge of C programming.
GRASS GIS

GRASS Database

/home/user/grassdata

Location

/prov_trentino
/trento
/user1
/europa
/wake
/world

Mapset

/PERMANENT

Geometry and attribute data

/cats
/cell
/cellhd
/cell_misc
/colr
/fcell
/hist

RASTER maps

/grid3
/soil3d

Volume maps

/streets
/parks
/lakes
/poi
/streets.dbf
/parks.dbf
/lakes.dbf
/poi.dbf

VECTOR maps
Integration in Grid environment

- Image data
- GRASS GIS (tar.gz)

Storage Resource Manager

CASTOR
CERN Advanced Storage manager

Storage Element

Job Description Language & Shell script files
Integration in Grid environment

- Image data
- GRASS GIS (tar.gz)
- GRASS GIS 5.4.1 Binaries
- GDAL - Geospatial Data Abstraction Library
- PROJ.4 - Cartographic Projections Library
Integration in Grid environment

Job Description Language & Shell script files

- Very basic jdl file for the initial testing
- The operations to perform on the data files are defined in the wrapper script which is uploaded at submission time
  - To do a different analysis of an image we only need to change the wrapper script

```
Executable = "filename.sh";
StdOutput = "filename.out";
StdError = "filename.err";
InputSandbox = {"./filename.sh"};
OutputSandbox = {"filename.out","filename.err"};
```

- Copies the source date and GRASS GIS tar ball to the worker node.
- Sets the GRASS GIS environment variables
- Executes the GRASS commands
- Copies the results back to the storage element
#!/bin/bash

# variables
FILE=image1.tif
OUTPUT_PREFIX=unsupclass_6_

# 'static' variables
GRASS_FILE=grassgis54.tar.gz
LFN_GRASS=/grid/unosat/agagodas/grass/$GRASS_FILE
LFN_DATA_SOURCE=/grid/unosat/agagodas/data/source/
LFN_DATA_RESULTS=/grid/unosat/agagodas/data/results/
LFN_DATA_SOURCE_FILE=$LFN_DATA_SOURCE$FILE
GRASS_ROOT=grass_root
mkdir -p $GRASS_ROOT

cd $GRASS_ROOT

lcg-cp -v lfn:$LFN_GRASS $GRASS_FILE

tar xvzf $GRASS_FILE

lcg-cp -v ifn:$LFN_DATA_SOURCE_FILE $FILE

# variables to customize:
GISDBASE=$PWD
GISBASE=$PWD
MAPSET=PERMANENT

#generate temporary LOCATION:
TEMPDIR=$$$.

mkdir -p $GISDBASE/$TEMPDIR/temp

(...)

Copy the files from SE to the WN

Set the GRASS GIS Environment variables
grassdata=$(ls $GISDBASE/$location_name/$MAPSET/cell/)

i.group group=group subgroup=subgroup input=$ALL
i.cluster group=group subgroup=subgroup sigfile=sigfile classes=6
i.maxlik group=group subgroup=subgroup sigfile=sigfile

class=unsupclass
r.out.tiff input=unsupclass output=unsuplcass_$(date +%d%m%y_%H%M%S)

for outputfile in `ls $OUTPUT_PREFIX*`; do echo "### Saving output file $outputfile:"; lcg-cr -v file:$outputfile -d srm://srm://public.cern.ch/castor/cern.ch$LFN_DATA_RESULTS$outputfile -l $LFN_DATA_RESULTS$outputfile; done
Current Status

- Data as been manually uploaded to the Storage Element

![Diagram showing data flow from User to Storage Element]

- A generic self contained GRASS GIS installation has been created
- Two different types of jobs have been sucessfully submitted and executed.

![Diagram showing job flow from User to Storage Element]

- The results have been manually retrieve from the Storage Element
- The validity of the results have been verified

![Diagram showing result flow from Storage Element to User]
Future Plans

- Upload all available imagery to the storage element while registering it in a metadata database.

- Select and test a list of predefined image processing operations.
- Create a web portal where registered users can request that existing images are processed using GRASS GIS. This will mask the use of the Grid infrastructure from the users and enable the use of the system from different platforms.

- Perform common operations on all available imagery (extract contour lines, unsupervised classification, vegetation index) and vector data.
www.unitar.org/unosat
unosat@unitar.org

Thank you!