

The screenshot displays the SAGA GIS desktop environment. On the left is a 'Manager' panel with a tree view of project layers. The top center shows an 'Overview' window with a 3D terrain model and a 'LandSat Imagery (3D View)'. Below the overview is a 'Command Line' window showing the execution of a 'Morphometry' tool. To the right of the command line is a 'Python' window with code for creating a SAGA API object. Further right is an 'R Script' window with R code for reading and processing SAGA data. At the bottom left is a 'System Architecture' diagram, and at the bottom right is a 'Tool Options' dialog for the 'Morphometry' tool, showing parameters like 'Grid (input)', 'Slope', and 'Aspect'.

SAGA

System for Automated Geoscientific Analyses

Automating your analysis using SAGA GIS

Johan Van de Wauw

Slides by:
Olaf Conrad, Michael Bock, Volker Wichmann

SAGA | System for Automated Geoscientific Analyses

The screenshot displays the SAGA GIS interface with several key components:

- Overview Panel:** Shows a 3D terrain model with elevation contours and a color-coded legend.
- LandSat Imagery (3D View):** Displays a 3D visualization of satellite imagery overlaid on the terrain.
- Command Line:** Shows the execution of SAGA commands, such as `libray path: (C:\saga-2.1.0_bin32\module\dev\morphometry.dll)`.
- Python:** Displays a Python script for processing SAGA data, including grid creation and analysis.
- R Script:** Shows an R script for connecting the map to SAGA grid format and performing statistical analysis.
- System Architecture Diagram:**
 - Graphical User Interface:** Includes the SAGA GUI, Management Analysis, and Visualization.
 - Command Line Interpreter:** Includes Shell, R/SAGA, and Shell scripts.
 - Script Languages:** Includes Python and Java.
 - Tool Libraries:** Includes Library A, B, C, D, and E.
 - Application Programming Interface:** Includes Data Management and Tool Management.
 - Data Structures:** Includes Tables, Vector and Raster Data, and TIN.
 - Helpers:** Includes Memory Handling, File, String, Numeric, and Statistics.
 - Tools:** Includes Library Interface, Parameter Lists, and User Interaction.

- SAGA is a Geographic Information System (GIS) software with strong capabilities for geodata processing and analysis.
- SAGA is programmed in the object oriented C++ language and supports the implementation of new functions with a very efficient Application Programming Interface (API).
- Tools are provided by framework independent Tool Libraries and can be accessed most simply via SAGA's Graphical User Interface (GUI) or various scripting environments..

Key Features

The screenshot displays the SAGA GIS interface with several key components:

- Overview Panel:** Shows a 3D terrain visualization with a color-coded elevation scale and a data table with columns for 'JAHR', 'TREND', and 'RESIDUAL'.
- Options Panel:** Configurable settings for tool execution, including Name, Description, Style, and Display options.
- Command Line:** A terminal window showing the execution of SAGA commands and their output.
- Python:** A code editor window showing Python scripts for data processing and analysis.
- R Script:** A code editor window showing R scripts for statistical analysis and visualization.
- System Architecture Diagram:** A flowchart showing the interaction between the Graphical User Interface, Command Line Interpreter, and Script Languages.
- Tool Libraries:** A diagram showing five distinct tool libraries (Library A through E).
- Application Programming Interface:** A diagram showing the integration of Data Management and Tool Management components.

- Object oriented system design
- Modular structure with framework independent tool development
- API with strong support for geodata handling
- GUI for intuitive data management, analysis and visualization
- Far more than 650 free tools
- Runs on Linux as well as on Windows operating systems
- Portable software runs without installation even from USB sticks
- Free and Open Source Software
- 10 years of continuous development

Drivers of Development

- SAGA's development is mainly driven by the research interests of its inventors and developers
 - Physical Geography, Hamburg University
 - SciLands GmbH, Göttingen
 - Laserdata GmbH, Innsbruck
- SAGA's publication as FOSS lead to several external inputs enriching the spectrum of developments
 - **V. Olaya:** SEXTANTE (Sistema Extremeno de Analisis Territorial), SAGA manual, module & system development
 - **T. Schorr:** GEOSTEP project, Linux, Unicode and 64bit compatibility, SAGA-Python interface
 - **V. Wichmann:** PhD Thesis (rockfall modelling), Laserdata GmbH, support & documentation, module & system development
 - **V. Cimmery:** SAGA 2 User Guide, documentation
 - **A. Brenning:** RSAGA plugin, accessing SAGA modules from R environment
 - **J. Van de Wauw:** Linux support and distribution (Debian/Ubuntu), bug fixes, module development
 - **J. Brunke:** Power User, Environment Agency, County Gifhorn
 - and many other contributions, mainly modules and documentation

Drivers of Development | Selected Projects, Physical Geography, Hamburg



- CARBIOCIAL – Carbon sequestration, biodiversity and social structures in Southern Amazonia: models and implementation of carbon-optimized land management strategies.
- The Future Okavango – Scientific support for sustainable land and resource management in the Okavango basin – GIS-based landscape analyses, environmental modelling, and decision support for integrated resource management.
- CHELSA - Climatologies at High Resolution for the Earth's Land Surface Areas. A research cooperation of Institute of Systematic Botany, University Zürich, Biodiversity, Macroecology & Conservation Biogeography Group, University Göttingen and Physical Geography, University Hamburg.
- SALEM - Development of a Model for the Spatial Prediction of Periglacial Deposits – Funded by the German Federal Institute for Geosciences and Natural Resources (BGR)
- SAGA-REKLIM – Climate Change and Forestry: Researches and developments for a SAGA based problem oriented regionalisation of spatially distributed climate data for Baden-Württemberg“.
- Spatial high resolution regionalization of urban climates, integrating statistical-dynamical downscaling and remote sensing techniques. Integrated Climate System Analysis and Prediction (CliSAP).
- Apart from any project: **GIS Training and Education**



System Architecture

- SAGA's architecture is modular.
- Its base is the **Application Programming Interface (API)**, which provides data object models, basic definitions for the programming of scientific modules and numerous helpful classes and functions.
- **Tool libraries** are Dynamic Link Libraries (DLL), or Shared Objects in Linux context, and provide the scientific methods. To access and run the tools you need a front end program.
- A **Graphical User Interface (GUI)** and a **Command Line Interpreter (CLI)** are the two generic SAGA **front ends**. Alternatively SAGA tools can be used with **scripting**.

Front Ends

Graphical User Interface
Interactive Data Management, Analysis and Visualization

Command Line Interpreter
Shell, RSAGA, Sextante

Script Languages
Python, Java

Tool Libraries

Library A

Library B

Library C

Library D

Library E

Application Programming Interface

Data Management

Tool Management

Data Structures
Tables, Vector and Raster Data, TIN

Helpers
Memory Handling, Files, Strings, Numerics, Statistics

Tools
Library Interface, Parameter Lists, User Interaction

Tool Programming

```

Chillshade::CHillshade(void)
{
    Parameters.Add_Grid(
        NULL, "ELEVATION", "Elevation", PARAMETER_INPUT);

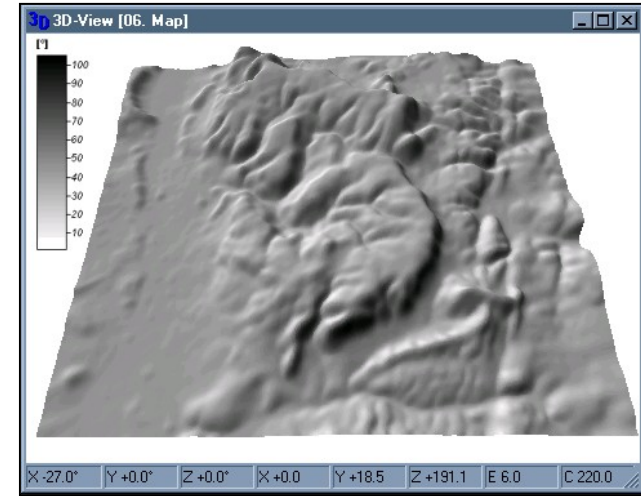
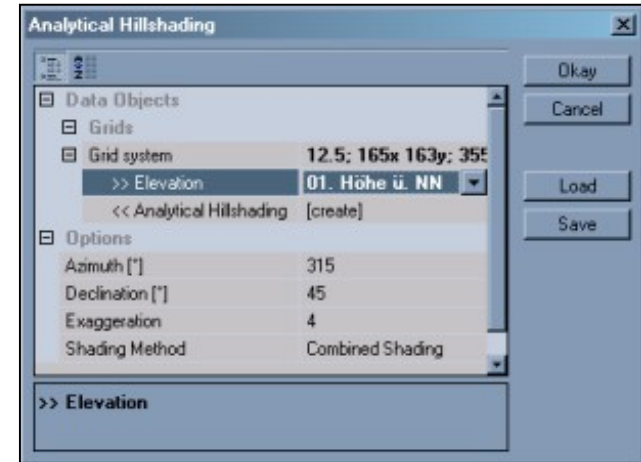
    Parameters.Add_Grid(
        NULL, "SHADE"      , "Shade"      , PARAMETER_OUTPUT);
    ...
}

```

```

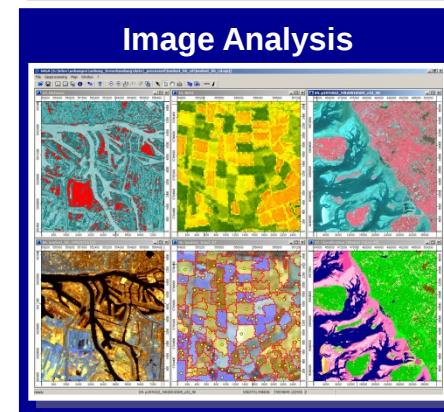
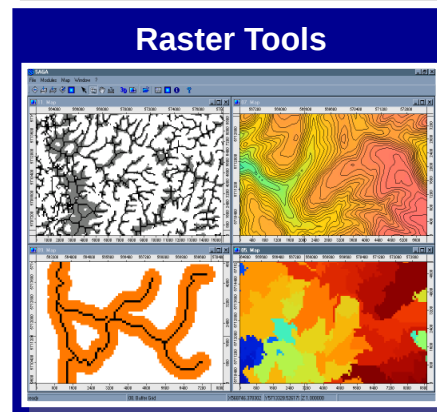
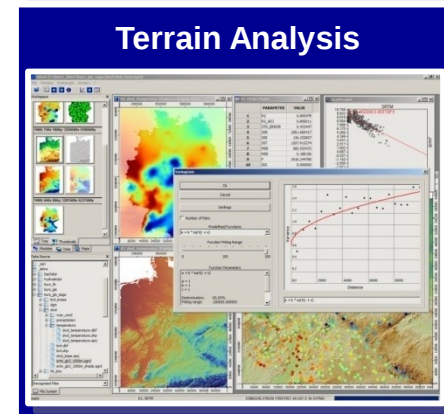
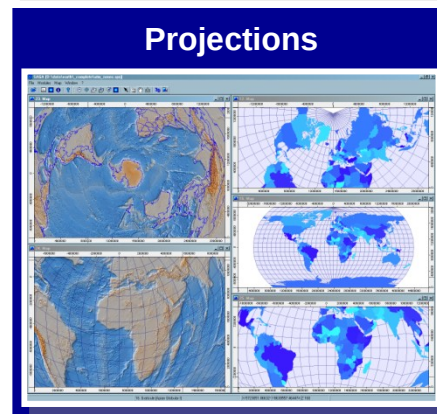
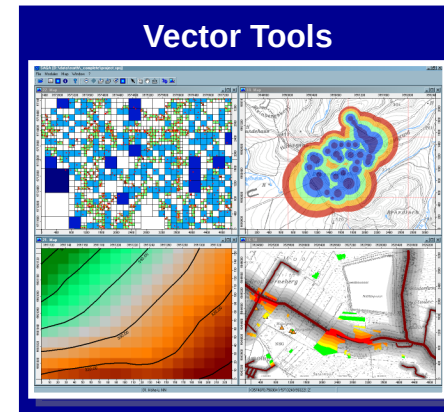
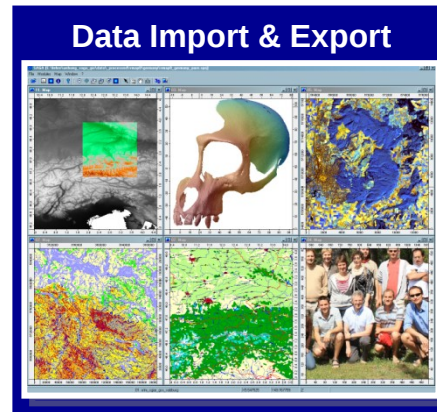
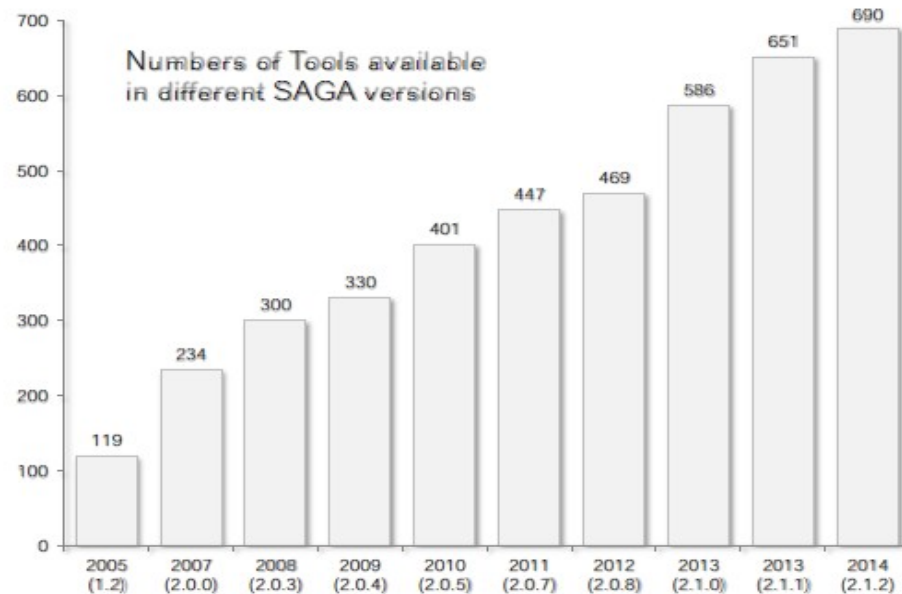
bool Chillshade::On_Execute(void)
{
    CSG_Grid *pDEM      = Parameters("ELEVATION")->asGrid();
    CSG_Grid *pShade    = Parameters("SHADE")    ->asGrid();
    ...
    for(y=0; y<Get_NY(); y++)
    {
        for(x=0; x<Get_NX(); x++)
        {
            if( pDEM->Get_Gradient(x, y, s, a) == false )
            {
                pShade->Set_NoData(x, y);
            }
            else
            {
                d      = acos(sin(s) * sin(Dec) + cos(s) * cos(Dec) * cos(a - Azi));
                pShade->Set_Value(x, y, d);
            }
        }
    }
}

```



The SAGA Toolset

- SAGA offers a comprehensive and growing set of free tools.
 - Data Import & Export
 - Cartographic Projections
 - Numerous Raster & Vector Data Tools
 - Image Processing
 - Terrain Analysis
 - Spatial & Geostatistics
 - and many more...



Front Ends | Graphical User Interface

Manager

Map View

Histogram

Print Layout

The screenshot displays the SAGA GIS interface with several key components highlighted by arrows:

- Manager:** The left sidebar showing a tree view of data layers and shapes.
- Map View:** The central area displaying a topographic map with elevation contours.
- Histogram:** A chart showing the distribution of values for the selected layer, with a color-coded legend.
- Print Layout:** A window showing a map with a legend and a scale bar, intended for printing.
- Properties:** A panel on the right showing the settings for the selected layer, including name, description, and display options.
- Notifications:** A message window at the bottom showing execution logs and error messages.
- Attributes:** A table window showing the attribute data for the selected layer.
- Scatterplot:** A plot showing the relationship between two variables, with a regression line and statistical data.

Properties

Notifications

Attributes

Scatterplot

Front Ends | Graphical User Interface

- Three Manager Controls
 - Modules, Data, Maps
- Properties depend on the object type selected in the manager control.
 - A settings and a description tab are common to all items.
 - In case of a tool, the settings show the tool's execution parameters. The description gives further information about the tool.
 - In case of a data set, the settings allow to change data set name, memory handling, symbology and other data type specific options. Besides a description a legend and a data set history is added.

The screenshot illustrates the SAGA GUI's Manager and Object Properties panels. The Manager panel is divided into three sections: Tools, Data, and Maps. The Object Properties panel is divided into three tabs: Settings, Description, and Attributes. The Settings tab shows parameters for a tool, a data set, or a map. The Description tab provides detailed information about the tool or data set. The Attributes tab shows a table of data attributes. The Legend tab shows a color-coded legend for a map.

Manager

Tools

- Contributors - Grids
- Geostatistics - Grids
- Geostatistics - Kriging
- Grid - Analysis
- Grid - Calculus
- Grid - Discretisation
- Grid - Filter
- Grid - Gridding
- Grid - Spline Interpolation
- B-Spline Approximation
- Thin Plate Spline Interpolation**
- Thin Plate Spline (Global)
- Thin Plate Spline (Local)
- Thin Plate Spline (TIN)
- Grid - Tools
- Grid - Visualisation
- Import GPS Data
- Import/Export - ESRI E00
- Import/Export - Grids
- Import/Export - Grids using GDAL

Data

- Grids
 - 12.5y.1888.1889.3550150W.568950
 - 01. Höhe ü. NN
 - 03. Hangneigung
 - 04. Exposition
 - 05. Wölbung (vertikal)
 - 5. 937x.633y.3549000W.56888405
 - 01. TK50_Ziegenhagen.dgm
- Shapes
 - Line
 - 01. Isohypsen (10m)
 - Point
 - 01. Bodenfeuchte, 12.12.1994
 - 02. Bodenfeuchte, 16.05.1994
 - Polygon
 - 01. Forstl. Standortkartierung
- T.I.N.
 - 01. Höhe ü. NN
- Tables
 - semivariogram.dbf

Maps

- 01. Map
 - 01. TK50_Ziegenhagen.dgm
 - 02. Schummerung
 - 01. Forstl. Standortkartierung
 - 01. Höhe ü. NN
- 02. Map
 - 02. Bodenfeuchte, 16.05.1994
 - 03. Bodenfeuchte, 31.10.1994
 - 01. Bodenfeuchte, 12.12.1994
 - 02. Schummerung
 - 09. log(Einzugsgebietsgröße)
- 03. Map
 - 02. Gewässernetz
 - 02. Schummerung
 - 08. Bodenfeuchte-Index (Böhner)
- 04. Map
 - 03. Chart (sectors):
 - 03. Upslope Area
 - 01. Isohypsen (10m)
 - 09. log(Einzugsgebietsgröße)

Object Properties

Settings

Data Objects

- Shape
 - Points [not set]
- Options
 - Target Grid: user defined
 - Method: with B-spline refinement
 - Threshold Error: 0.0001
 - Maximum Level: 11
 - Update View:

Apply Restore Execute Load Save

Parameters Description

Description

Module: Thin Plate Spline Interpolation

Copyrights (c) 2006 by Olaf Conrad
Menu: Grid > Gridding > Spline Interpolation

Description

Multilevel B-spline algorithm for spatial interpolation of scattered data as proposed by Lee, Wolberg and Shin (1997). The algorithm makes use of a coarse-to-fine hierarchy of control lattices to generate a sequence of bicubic B-spline functions, whose sum approaches the desired interpolation function. Large performance gains are realized by using B-spline refinement to reduce the sum of these functions into one equivalent B-spline function.

Parameters Description

Settings

Options

General

Name: Forstl. Standortkartierung

Show Legend:

Display

Chart: 107 parameters

Fill Style: Opaque

Outline:

Outline Color: Black

Outline Size: 0

Show Points:

Display: Visibility

Always Show:

Scale Dependent: 0; 1000

Display: Color Classification

Type: Lookup Table

Attribute: BOTYP

Unique Symbol

Color: Red

Apply Restore Load Save

Parameters Description Attributes

Attributes

ID	Name	Value
1		28
2	AREA	153026.060000
3	PERIMETER	4610.418000
4	LINDABODEN_d	28
5	LINDABODEN_C	40
6	BOTYP	14
7	ASUBS	Lu
8	BSUBS	SH
9	CSUBS	SH
10	DSUBS	SH
11	AHOR	Ah
12	BHOR	Bv
13	CHOR	IBv
14	DHOR	ITv
15	ATIEFE	6.000000
16	BTIEFE	27.000000
17	CTIEFE	120.000000

Apply Restore

Description Attributes Legend

Settings

Options

Frame

Show:

Width: 17

Print Layout

Show Legend:

Display Resolution: 2

Frame

Show:

Width: 7

Apply Restore Load Save

Parameters Description Legend

Legend

Isohypsen (10m)

Forstl. Standortkartierung

BOTYP

- Gley
- Auenbraunerde, tiefr.
- Braunerde, lehmig-schluffig, tiefr.
- Braunerde, sandig-schluffig, tiefr.
- Braunerde, sandig-schluffig, mittel-tiefr.
- Braunerde, sandig-schluffig, mittel.
- Braunerde, sandig-lehmig, tiefr.

Parameters Description Legend

Front Ends | Command Line Interpreter

- The SAGA Command Line Interpreter (CLI) makes it possible to execute SAGA tools from a command line or shell console.
- This is not very user friendly, but allows to write batch/shell scripts for further automation of process flows.

```
C:\Windows\system32\cmd.exe
C:\>saga_cmd ta_morphometry 0

#####  ##  #####  ##
###    ##  ##  ###
###  # ## ##  ### # ##
### #####  ##  # #####
##### # ##  ##### # ##

library path:  c:\saga_2.1.0_win32\modules\ta_morphometry.dll
library name:  Terrain Analysis - Morphometry
module name :  Slope, Aspect, Curvature
author       :  O.Conrad (c) 2001

Usage: saga_cmd -ELEVATION <str> [-SLOPE <str>] [-ASPECT <str>] [-CURU <str>] [-
-HCURU <str>] [-UCURU <str>] [-METHOD <str>]
-ELEVATION:<str>      Elevation
                     Grid (input)
-SLOPE:<str>          Slope
                     Grid (output)
-ASPECT:<str>        Aspect
                     Grid (output)
-CURU:<str>          Curvature
                     Grid (optional output)
-HCURU:<str>         Plan Curvature
                     Grid (optional output)
-UCURU:<str>         Profile Curvature
                     Grid (optional output)
-METHOD:<str>      Method
                     Choice
                     Available Choices:
                     [0] Maximum Slope (Traub et al., 1975)
                     [1] Maximum Slope (Tessier et al., 1996)
```

```
Usage:
saga_cmd [-h, --help]
saga_cmd [-v, --version]
saga_cmd [-b, --batch]
saga_cmd [-d, --docs]
saga_cmd [-f, --flags][=qrsilpx][-c, --cores][=#] <LIBRARY>
saga_cmd [-f, --flags][=qrsilpx][-c, --cores][=#] <SCRIPT>

[-h], [--help]      : help on usage
[-v], [--version]   : print version information
[-b], [--batch]     : create a batch file example
[-d], [--docs]     : create module documentation in current working directory
[-c], [--cores]    : number of physical processors to use for parallel processing
[-f], [--flags]    : various flags for general usage [qrsilpx]
q                  : no progress report
r                  : no messages report
s                  : silent mode (no progress and no messages)
i                  : allow user interaction
l                  : load translation dictionary
p                  : load projections dictionary
x                  : use XML markups for synopses and messages

<LIBRARY>          : name of the library
<MODULE>           : either name or index of the module
<OPTIONS>          : module specific options
<SCRIPT>           : saga cmd script file with one or more module calls
```

```
1  SET FLAGS=-f=q
2  REM SET SAGA=.
3
4  IF EXIST srtm.tif GOTO :SRTM
5  ECHO create a Gaussian landscape
6  saga_cmd %FLAGS% recreations_fractals 5 -GRID=dem.sgrd -MX=400 -NY=400 -H=0.75
7  :SRTM
8  ECHO import and project srtm (geotiff)
9  saga_cmd %FLAGS% srtm2asc srtm.tif srtm.asc
10 saga_cmd %FLAGS% srtm2asc srtm.tif srtm.asc
11
12 ECHO do some
13 saga_cmd %FLAGS% cluster_analysis cluster_analysis.sgrd
14 saga_cmd %FLAGS% cluster_analysis cluster_analysis.sgrd
15 saga_cmd %FLAGS% imagery_classification imagery_classification 1 -GRIDS=%INPUT% -CLUSTER=cluster.sgrd -NORMALISE -NCLUSTER=5
16 saga_cmd %FLAGS% majority_filter majority_filter 1 -INPUT=cluster.sgrd -RADIUS=3
17
18 ECHO run saga
19 SET INPUT=cluster.sgrd
20 saga_cmd %FLAGS% grid_filter 6 -INPUT=cluster.sgrd -RADIUS=3
21
22 # vectorization
23 shapes_grid 6 -GRID=cluster.sgrd -POLYGONS=cluster.shp -CLASS_ALL=1
24
25 # select cluster class 1
26 shapes_tools 3 -SHAPES=cluster.shp -FIELD=ID -EXPRESSION="a = 1"
27
28 # save selectione
29 shapes_tools 6 -INPUT=cluster.shp -OUTPUT=cluster_class1.shp
```

Front Ends | Scripting

- Via the **SWIG** compiler (Simplified Wrapper and Interface Generator) it is possible to expose the SAGA API as well as SAGA tools to various (script) programming languages, e.g. Python, Java, C#, R.
- The **Python** interface allows to run SAGA modules from (web)server processes and the integration with many other software e.g. ArcGIS.
- SAGA modules can be execute directly from R Scripts via the **RSAGA** interface.



```

74 grid_difference.py · D:\develop\saga\saga_2_python\grid_difference.py
File Edit Format Run Options Windows Help

#!/usr/bin/env python
"SAGA-Script Beispiel fuer Python"

import saga_api, sys, os

def grid_difference(fA, fB, fC):
    A = saga_api.SG_Create_Grid()
    if A.Create(fA) == 0:
        print 'Fehler: <' + fA + '> konnte nicht geladen werden!'
        return 0

    B = saga_api.SG_Create_Grid()
    if B.Create(fB) == 0:
        print 'Fehler: <' + fB + '> konnte nicht geladen werden!'
        return 0

    if A.is_Compatible(B) == 0:
        print 'Fehler: Raster <' + fA + '> und <' + fB + '> sind nicht
        return 0

    bPixelWise = 0
    if bPixelWise == 1: # pixelweise auswertung (langsamer als zweite
        C = saga_api.SG_Create_Grid(A.Get_System())

        for y in range(0, C.Get_NY()):
            print '.',
            for x in range(0, C.Get_NX()):
                if A.is_NoData(x, y) or B.is_NoData(x, y):
                    C.Set_NoData(x, y)
                else:
                    C.Set_Value(x, y, A.asDouble(x, y) - B.asDouble(x, y))

        C.Save(fC)
        print

    else: # benutzte CSG_Grid funktion Subtract zur diff
        A.Subtract(B)
        A.Save(fC)

    print 'fertig!'
    return 1
  
```

```

Tinn-R [E:\tmp\tomislav\bugs_in_SAGA.R]
R complex
bugs_in_SAGA.R

# Comparison SAGA versus gstat
# SAGA Version: 2.0.3

library(gstat)
library(maptools)
library(rgdal)
NL_RD <- "+proj=sterea +lat_0=52.1561605555555 +lon_0=5.38763888888889 +k=0.999908 +x_0=

# points:
data(meuse)
coordinates(meuse) <- ~x+y
proj4string(meuse) <- CRS(NL_RD)
meuse$logzinc <- log1p(meuse$zinc)

# grids:
data(meuse.grid)
coordinates(meuse.grid) <- ~x+y
gridded(meuse.grid) <- TRUE
fullgrid(meuse.grid) <- TRUE
proj4string(meuse.grid) <- CRS(NL_RD)
gridcell <- mean(meuse.grid@grid$cellsize)
# convert the map to SAGA grid format:
write.asciigrd(meuse.grid["dist"], "meuse_dist.asc", na.value=-1)
saga_api.to_saga_grid(meuse_dist.asc, "meuse_dist.saga", in.path=TRUE)
  
```

RSAGA

Python scripts

- Via ZOO-WPS
- Via QGIS Processing

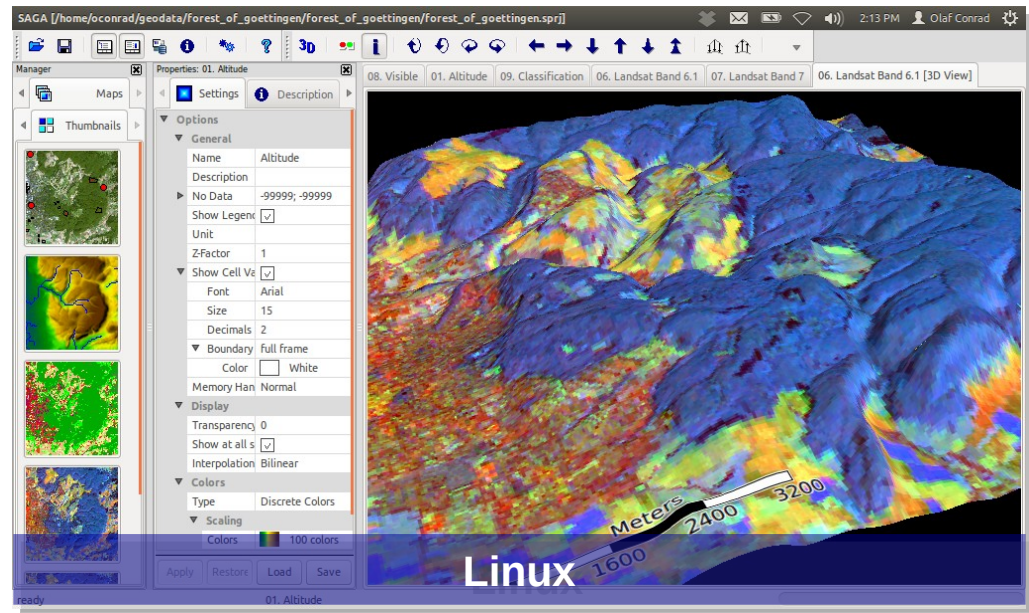
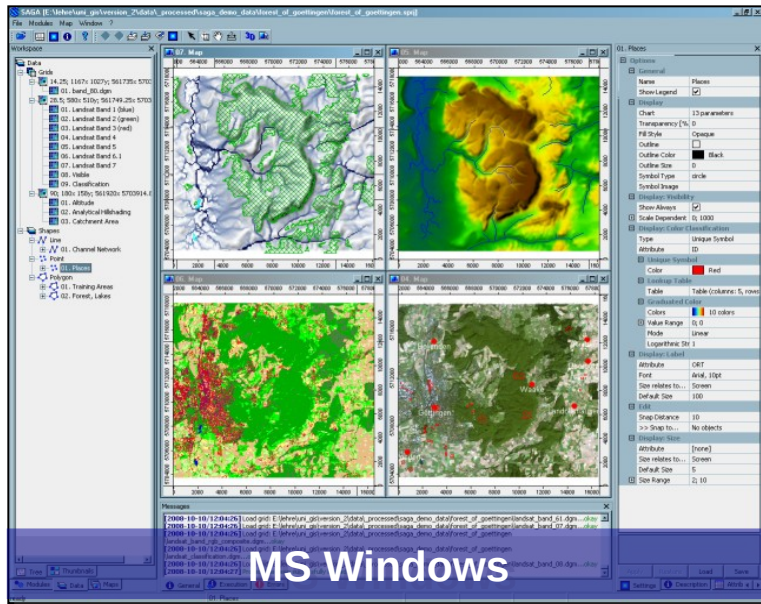
<http://www.zoo-project.org/docs/kernel/sagagis.html>

The screenshot displays the QGIS Processing Modeler interface. The main window is titled "Processing modeler" and shows a workflow diagram for "Watershed from DEM and threshold". The workflow consists of the following steps:

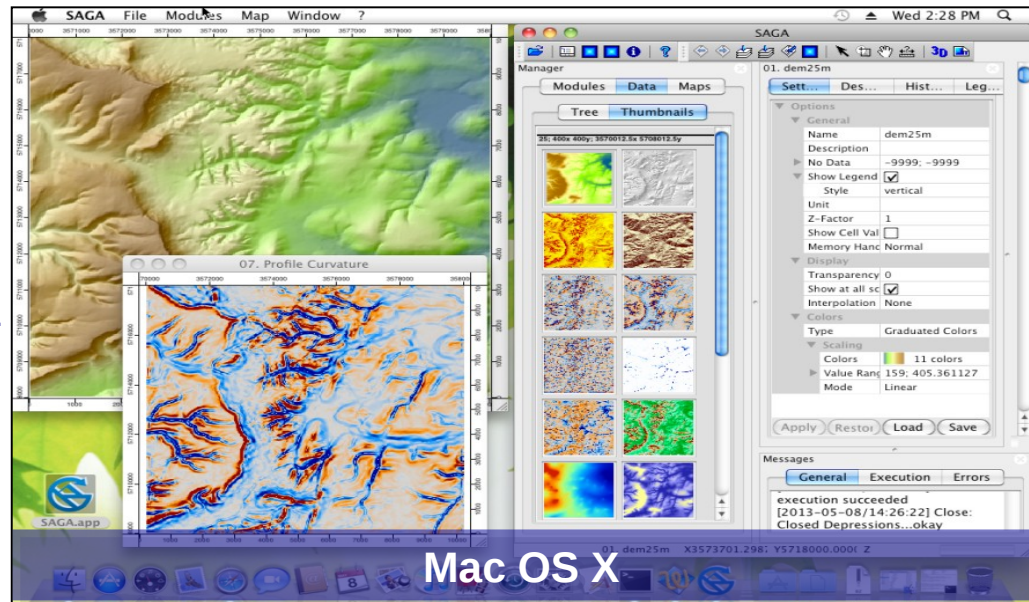
- DEM** (Raster layer) is the primary input.
- Fill Sinks** (Raster) receives input from DEM.
- Catchment area (parallel)** (Raster) receives input from DEM.
- Channel network** (Vector) receives input from both the output of "Fill Sinks" and the output of "Catchment area (parallel)".
- Threshold for channel definition** (Raster) receives input from DEM.
- Watershed basins** (Raster) receives input from the output of "Channel network" and the output of "Threshold for channel definition".

The left sidebar shows the "Parameters" list, including Boolean, Extent, File, Number, Raster Layer, String, Table, Table field, and Vector layer. The right sidebar shows the "Processing Toolbox" with a list of algorithms, including "Watershed from DEM and threshold", "Convergence index", "Polygon centroids", "r.aspect", "v.voronoi", and "Cartographic to geographic coordinates conversion". The bottom of the toolbox is labeled "Simplified interface".

Supported Platforms

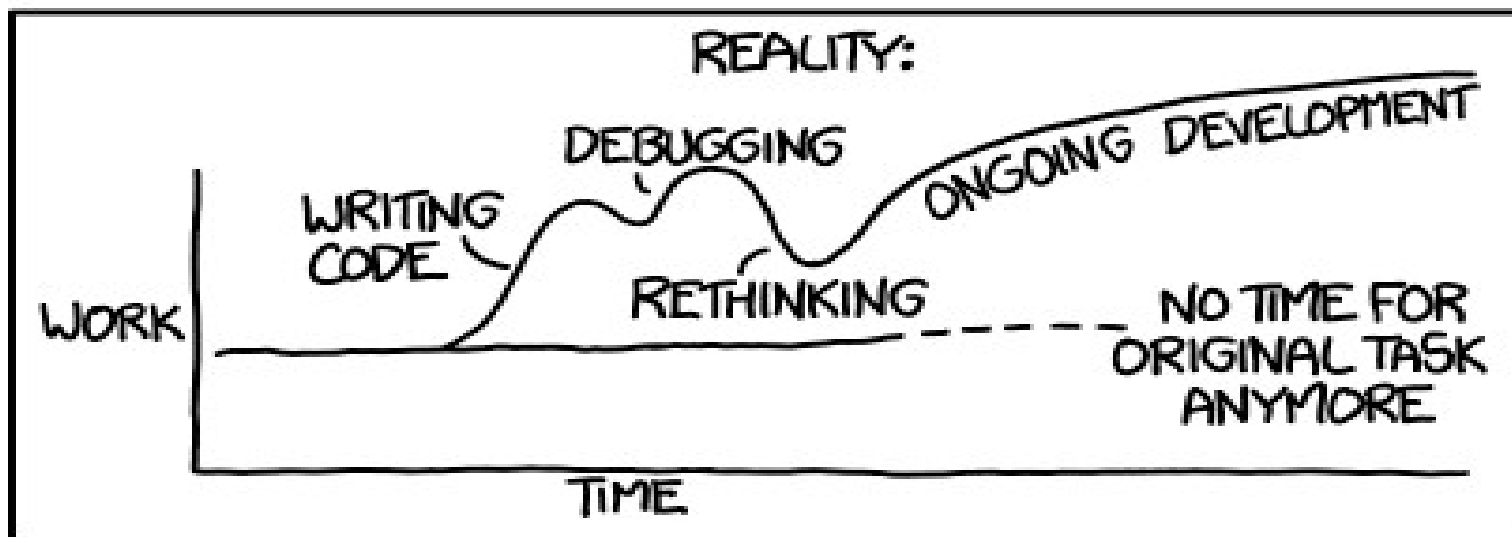
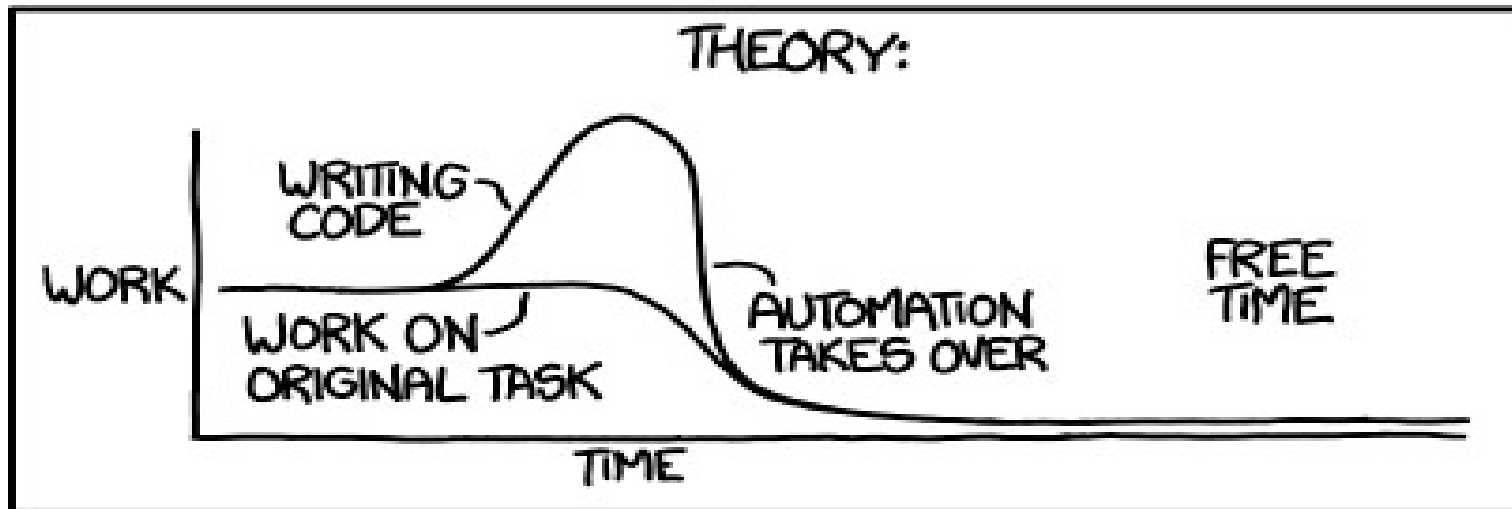


- MS Windows
- Linux
- FreeBSD
 - Maintainer: Rainer Hurling, NW-FVA
- Mac OS X
 - Maintainer: Wanted !



The SAGA Development

"I SPEND A LOT OF TIME ON THIS TASK.
I SHOULD WRITE A PROGRAM AUTOMATING IT!"



Changes and New Features in the GUI

The screenshot displays the SAGA GIS interface with several key components:

- Top Menu:** File, Geoprocessing, Map, Window, ?
- Toolbox:** A list of tools including 'Load Tool Library', 'Find and Run Tool', and 'Climate'. The 'Find and Run Tool' option is highlighted.
- Run Tool Dialog:** A dialog box with a search field containing 'slope'. The search results list several tools, with '05: Direct neighbours - slope and aspect' selected. Other visible results include 'Aspect-Slope Grid', 'DTM Filter (slope-based)', 'Downslope Area [interactive]', 'Downslope Distance Gradient', 'Relative Heights and Slope Positions', 'Slope Length', 'Slope Limited Flow Accumulation', and 'Slope, Aspect, Curvature'.
- Map View:** A topographic map showing elevation contours and a grid. The map is overlaid with a dashed grid. A scale bar at the bottom indicates distances in meters (0 to 9600). The status bar at the bottom shows coordinates: X560759.674717 Y5715226.856... Z.
- Manager Panel:** A panel on the right side of the map showing a tree view of the project. The 'Maps' folder is expanded, showing layers: '01. Altitude', 'Graticule', '01. Channel Network', and '01. Altitude'.
- Bottom Status Bar:** Displays 'ready' and 'Graticule'.

New Tools | Data Base Integration

- Database access via Open Data Base Connection (ODBC) interface.
 - SQL – Structured Query Language
 - Problem: binary data types (e.g. BLOBs)
- PostgreSQL + PostGIS
 - Direct linking

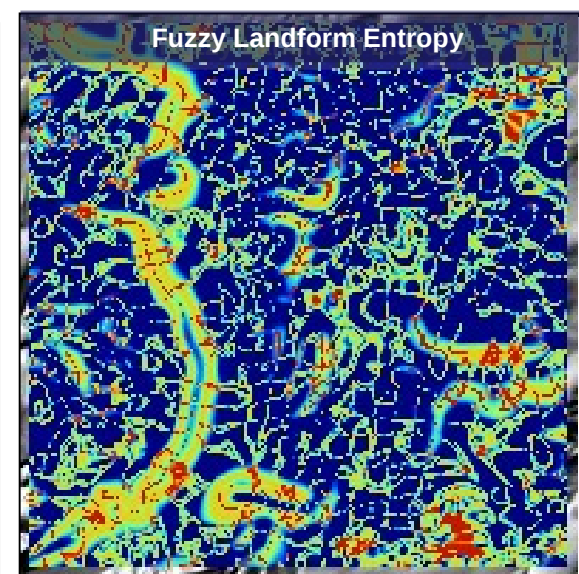
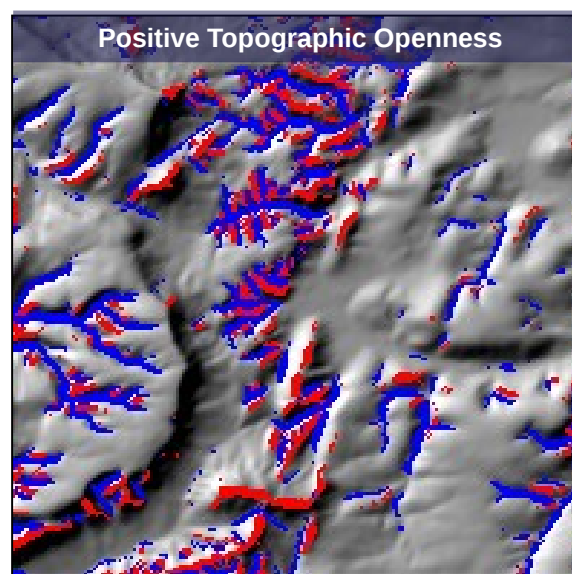
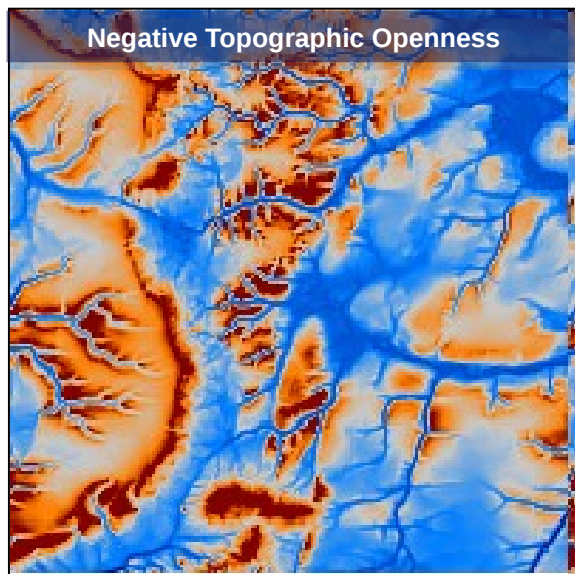
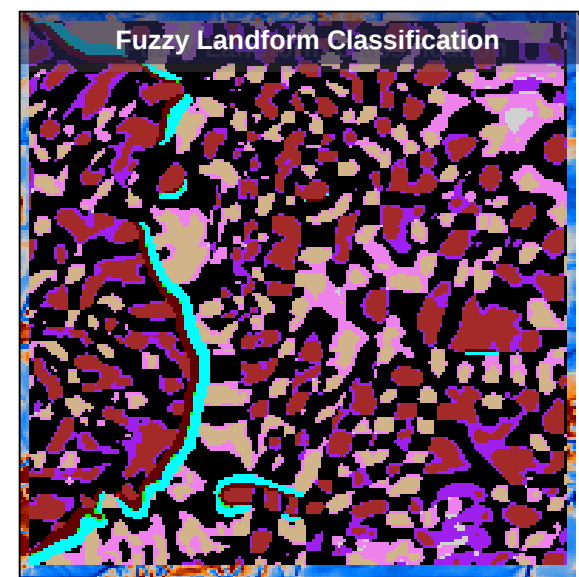
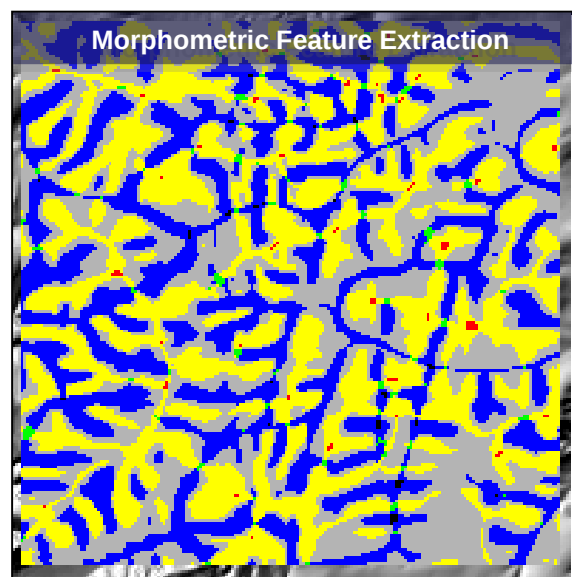
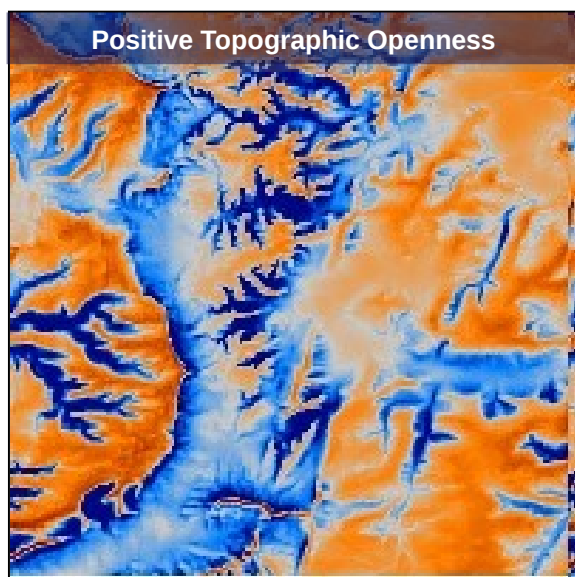
The screenshot illustrates the process of database integration in SAGA GIS. It shows several overlapping windows:

- Import/Export - ODBC/OTL**: A menu with options like 'ODBC Commit/Rollback Transaction', 'ODBC Connect', 'ODBC Disconnect', 'ODBC Execute SQL', 'Points Export', 'Points Import', 'PostGIS Shapes Export', 'PostGIS Shapes Import', 'Table Deletion', 'Table Export', 'Table Field Description', 'Table Import', and 'Table from Query'.
- Data Source**: A dialog box showing a list of data sources. Under 'GHCN <Postgres>', several tables are listed: 'geography_columns', 'geometry_columns', 'ghcn_countries', 'ghcn_p', 'ghcn_t_stations', and 'spatial_ref_sys'. A context menu is open over 'ghcn_t_stations' with options 'Refresh', 'Close', and 'Query'.
- Table from Query**: A dialog box for creating a table from a SQL query. The 'Where' clause is set to 'a.wmo_id=b.wmo_id AND a.wmo_mod=b.wmo_mod AND a'. A blue arrow points from this clause to the map below.
- PostgreSQL Sources**: A dialog box showing a list of PostgreSQL sources. Under 'lisws [localhost:5432]', several tables are listed: 'tdtaecho_12_29', 'tdtaecho_12_30', 'tdtaecho_13_31', 'tdtaecho_13_32', 'tdtarasterlayer', 'tdtatile', 'geography_columns', 'geometry_columns', and 'spatial_ref_sys'.
- World Map**: A world map displaying the results of the SQL query as colored points. The map has a coordinate system with longitude from 0 to 340 and latitude from 0 to 140.

The SQL query shown in the 'Table from Query' dialog is:

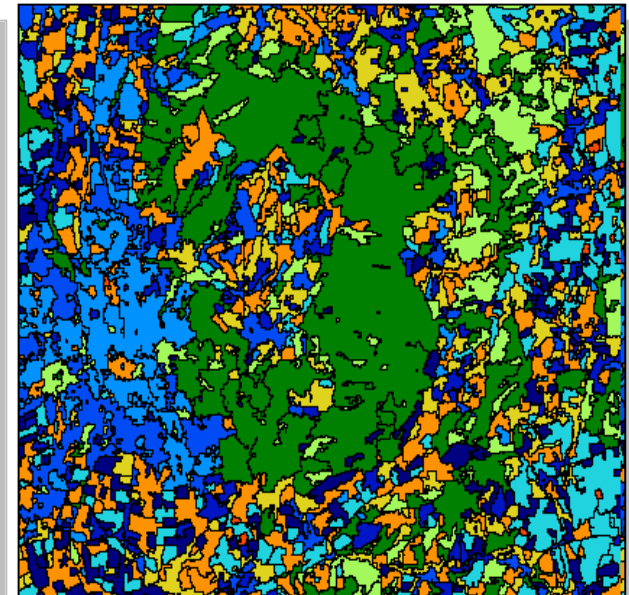
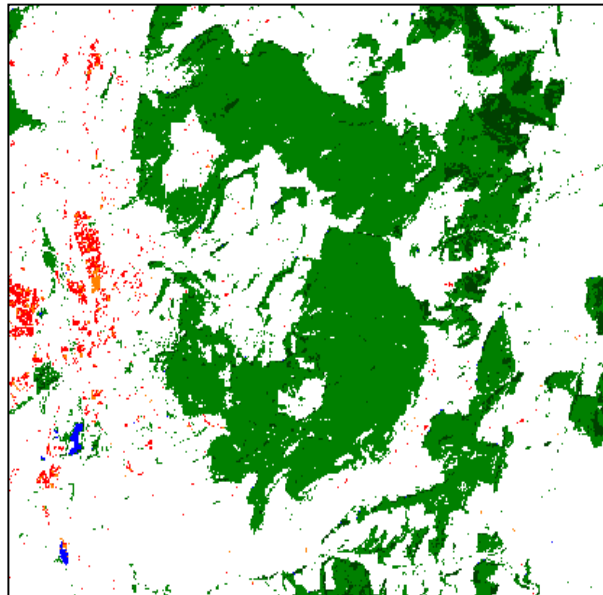
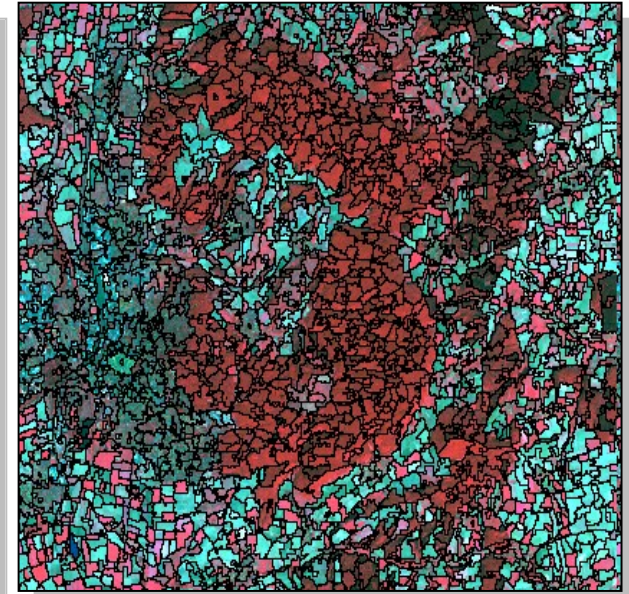
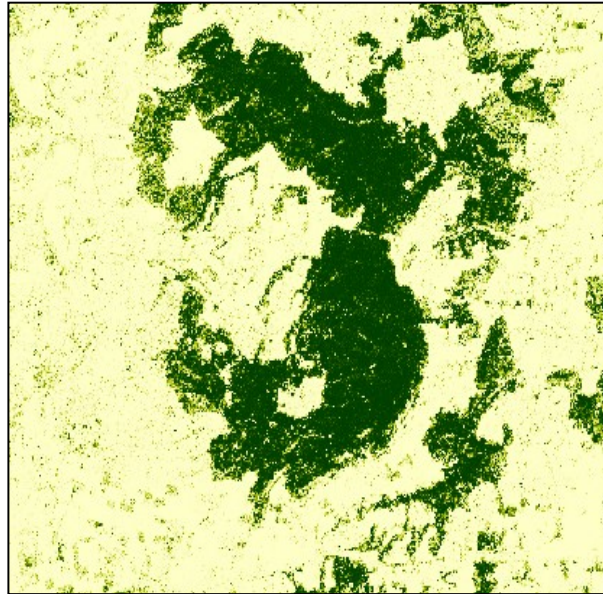
```
a.wmo_id=b.wmo_id AND a.wmo_mod=b.wmo_mod AND b.year=1950
```

New Tools | Terrain Analysis & Classification

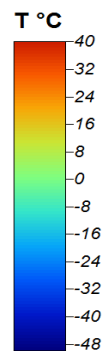
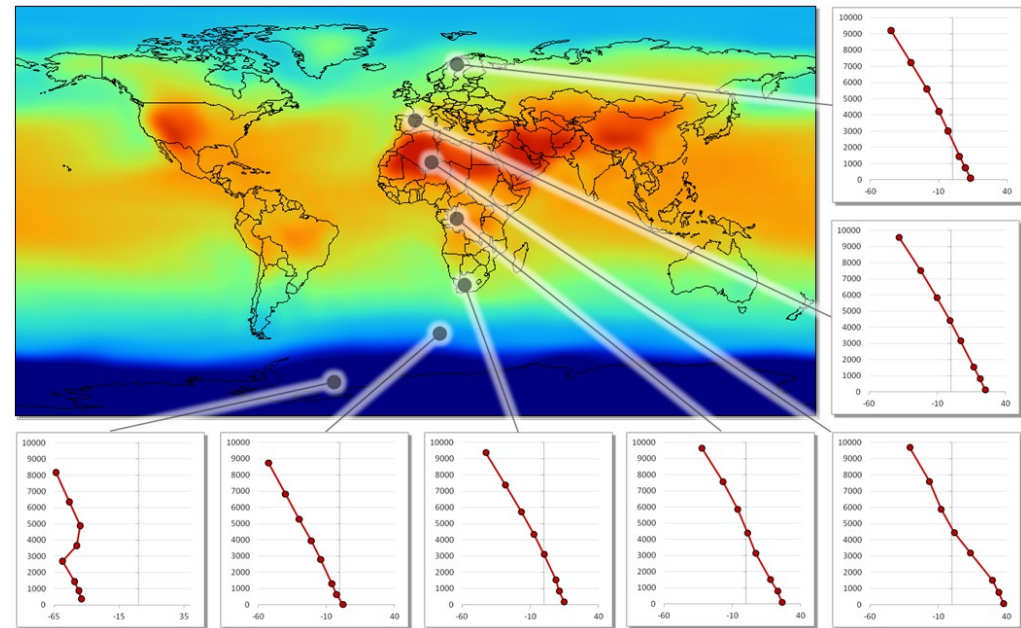
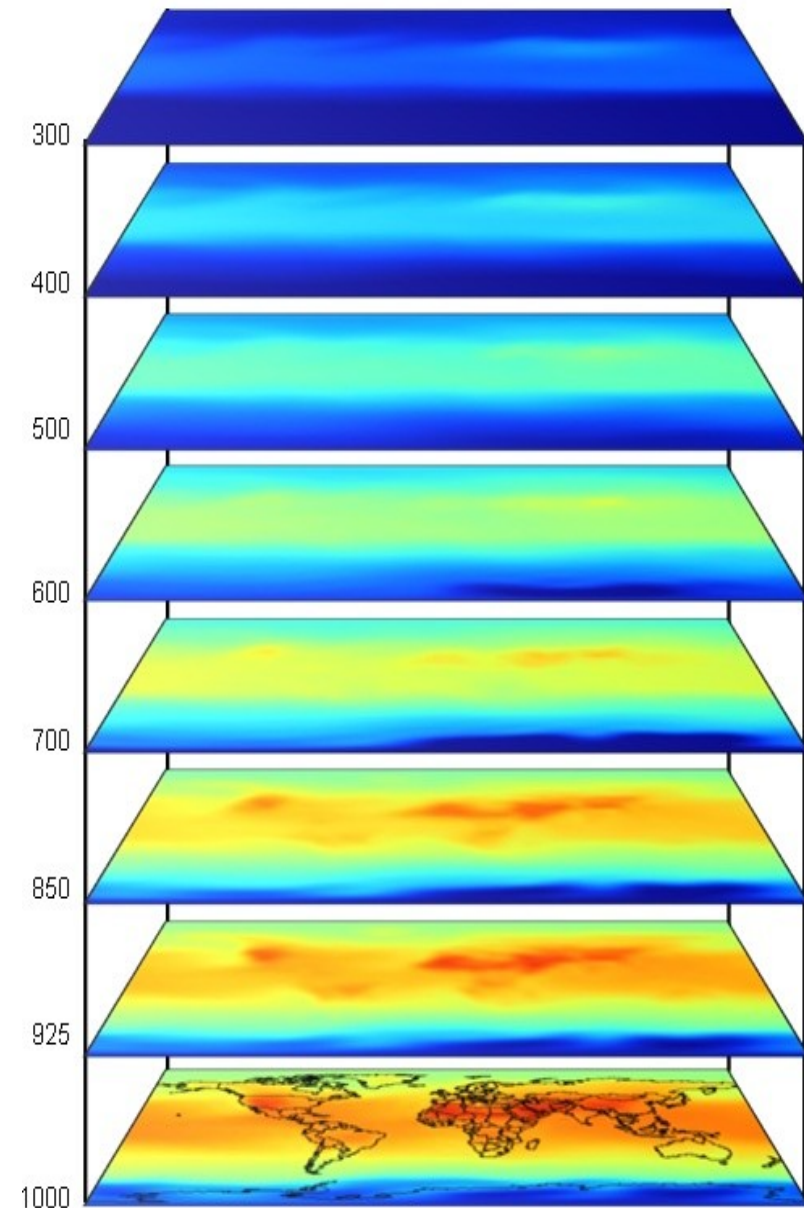


New Tools | Remote Sensing & Image Analysis

- Landsat Tools
 - Reflectance from metadata
 - Cloud Cover Assessment
- Orthorectification
 - From flight parameters
- Classification Tools
 - Support Vector Machine (SVM)
 - Maximum Entropy
 - Random Forest
 - OBIA

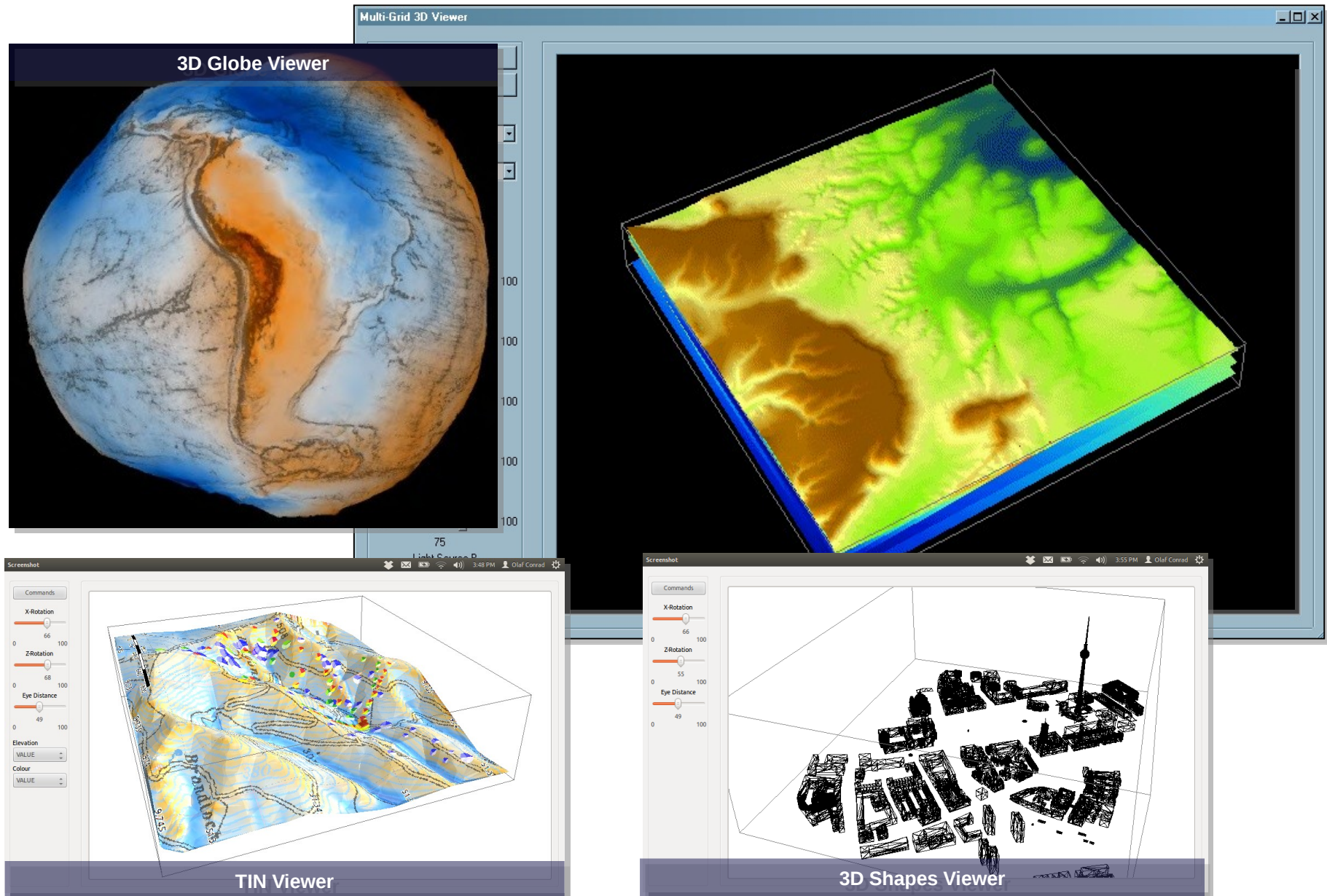


New Tools | Climate Data & Regionalization



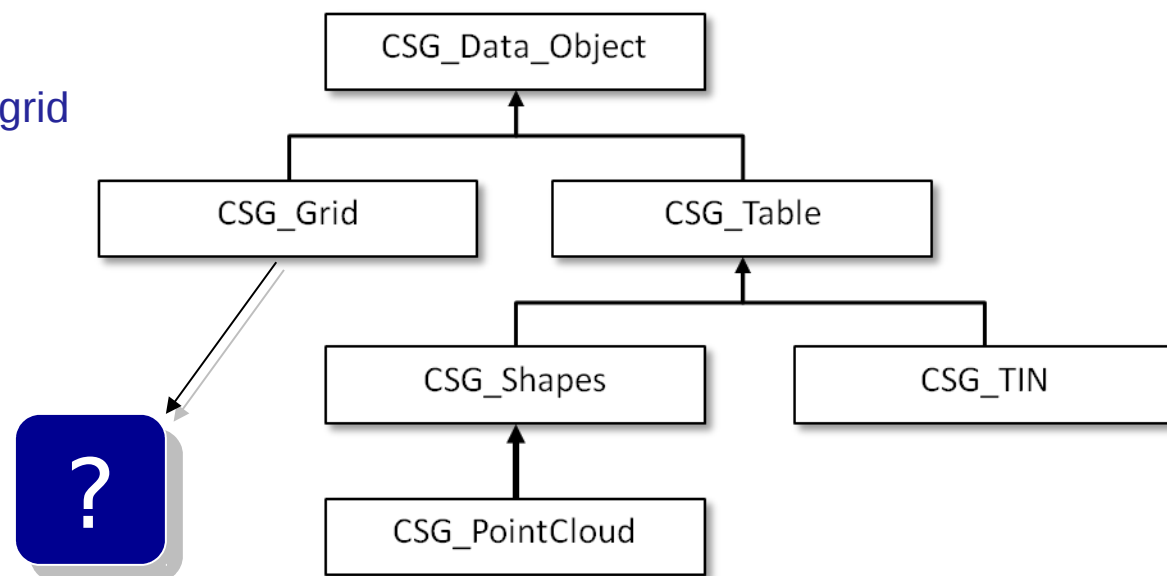
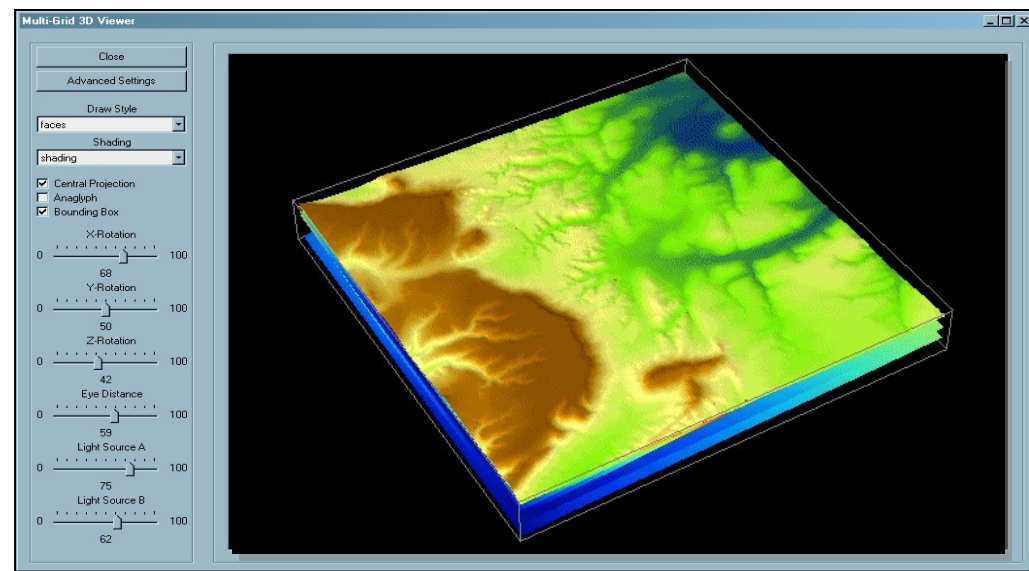
- Improved NetCDF support
- Downscaling tools
 - General and Regional Climate Models (GCM / RCM)

New Tools | 3D Viewer



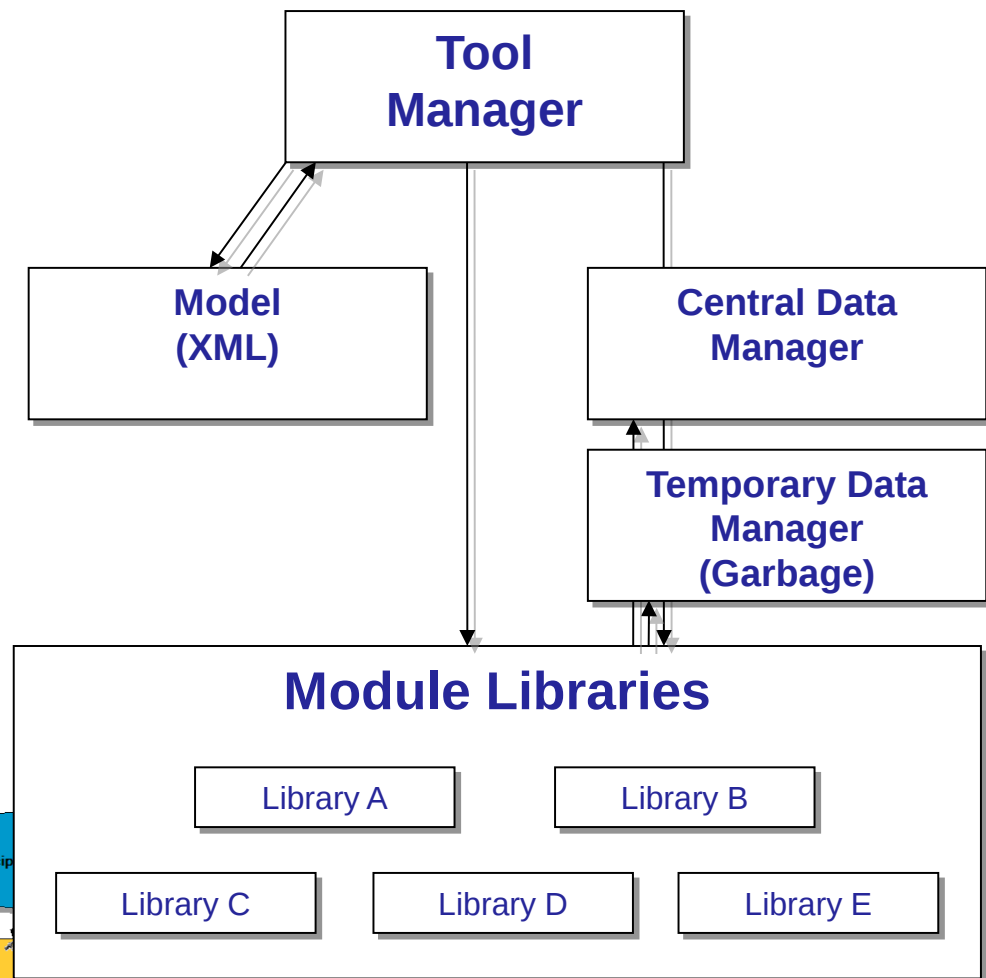
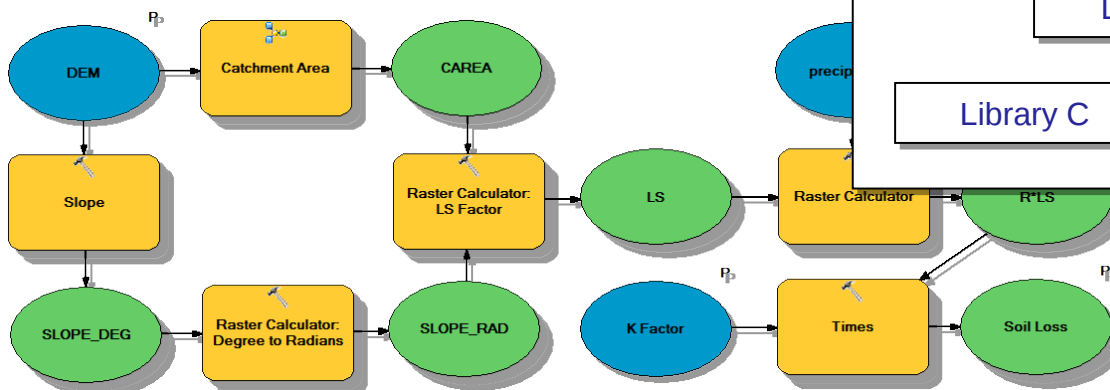
Outlook | Towards Multidimensional Raster

- There is a need for ,Multi-Raster'
 - Multi-/Hyperspectral Data
 - Remote Sensing
 - Volume Representation
 - Geology, Soils, Atmosphere
 - Time Series
- Points for discussion
 - Class inheritance
 - 3 or more dimensions
 - Visualisation tools
 - How to use with standard grid tools



Outlook | Towards a SAGA Model Builder

- XML based definition
 - Model parameters
 - Which tools to use and how to combine them
- Interpretation by the Module Manager
 - Performs plausibility checks
 - Module execution
 - Temporary data to a garbage collector
- Model building
 - Typing XML code
 - From data set history
 - Visual model designer



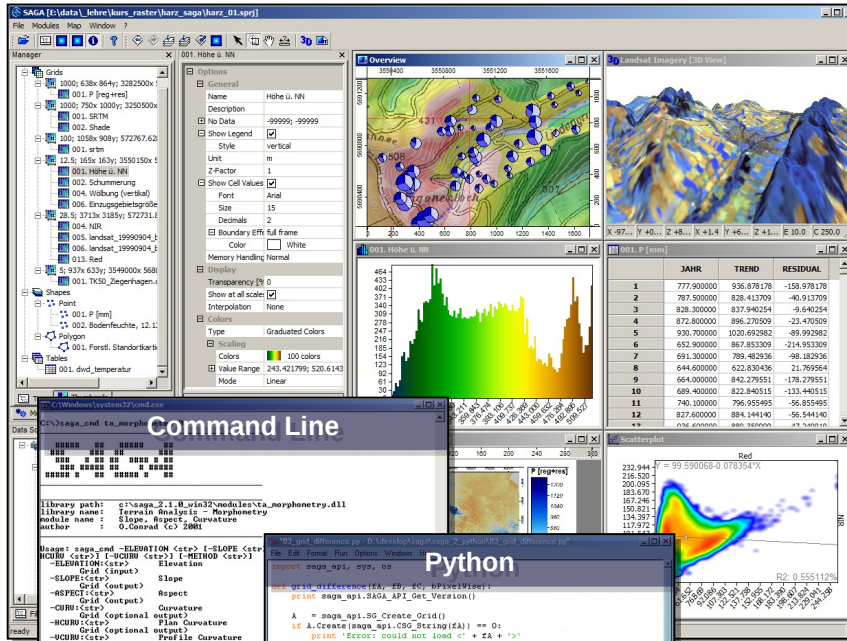
SAGA | System for Automated Geoscientific Analyses

Ways of Automation

with

SAGA

System for Automated Geoscientific Analyses



```

C:\> saga_cmd -t sagecmd -i 0 -m 32 -n 1920 -p 2 -d 1024 -f 1 -a 1 -s 1 -v 1 -e 1 -g 1 -h 1 -b 1 -c 1 -o 1 -r 1 -w 1 -l 1 -k 1 -x 1 -y 1 -z 1 -aa 1 -ab 1 -ac 1 -ad 1 -ae 1 -af 1 -ag 1 -ah 1 -ai 1 -aj 1 -ak 1 -al 1 -am 1 -an 1 -ao 1 -ap 1 -aq 1 -ar 1 -as 1 -at 1 -au 1 -av 1 -aw 1 -ax 1 -ay 1 -az 1 -ba 1 -bb 1 -bc 1 -bd 1 -be 1 -bf 1 -bg 1 -bh 1 -bi 1 -bj 1 -bk 1 -bl 1 -bm 1 -bn 1 -bo 1 -bp 1 -bq 1 -br 1 -bs 1 -bt 1 -bu 1 -bv 1 -bw 1 -bx 1 -by 1 -bz 1 -ca 1 -cb 1 -cc 1 -cd 1 -ce 1 -cf 1 -cg 1 -ch 1 -ci 1 -cj 1 -ck 1 -cl 1 -cm 1 -cn 1 -co 1 -cp 1 -cq 1 -cr 1 -cs 1 -ct 1 -cu 1 -cv 1 -cw 1 -cx 1 -cy 1 -cz 1 -da 1 -db 1 -dc 1 -dd 1 -de 1 -df 1 -dg 1 -dh 1 -di 1 -dj 1 -dk 1 -dl 1 -dm 1 -dn 1 -do 1 -dp 1 -dq 1 -dr 1 -ds 1 -dt 1 -du 1 -dv 1 -dw 1 -dx 1 -dy 1 -dz 1 -ea 1 -eb 1 -ec 1 -ed 1 -ee 1 -ef 1 -eg 1 -eh 1 -ei 1 -ej 1 -ek 1 -el 1 -em 1 -en 1 -eo 1 -ep 1 -eq 1 -er 1 -es 1 -et 1 -eu 1 -ev 1 -ew 1 -ex 1 -ey 1 -ez 1 -fa 1 -fb 1 -fc 1 -fd 1 -fe 1 -ff 1 -fg 1 -fh 1 -fi 1 -fj 1 -fk 1 -fl 1 -fm 1 -fn 1 -fo 1 -fp 1 -fq 1 -fr 1 -fs 1 -ft 1 -fu 1 -fv 1 -fw 1 -fx 1 -fy 1 -fz 1 -ga 1 -gb 1 -gc 1 -gd 1 -ge 1 -gf 1 -gg 1 -gh 1 -gi 1 -gj 1 -gk 1 -gl 1 -gm 1 -gn 1 -go 1 -gp 1 -gq 1 -gr 1 -gs 1 -gt 1 -gu 1 -gv 1 -gw 1 -gx 1 -gy 1 -gz 1 -ha 1 -hb 1 -hc 1 -hd 1 -he 1 -hf 1 -hg 1 -hh 1 -hi 1 -hj 1 -hk 1 -hl 1 -hm 1 -hn 1 -ho 1 -hp 1 -hq 1 -hr 1 -hs 1 -ht 1 -hu 1 -hv 1 -hw 1 -hx 1 -hy 1 -hz 1 -ia 1 -ib 1 -ic 1 -id 1 -ie 1 -if 1 -ig 1 -ih 1 -ii 1 -ij 1 -ik 1 -il 1 -im 1 -in 1 -io 1 -ip 1 -iq 1 -ir 1 -is 1 -it 1 -iu 1 -iv 1 -iw 1 -ix 1 -iy 1 -iz 1 -ja 1 -jb 1 -jc 1 -jd 1 -je 1 -jf 1 -jg 1 -jh 1 -ji 1 -jj 1 -jk 1 -jl 1 -jm 1 -jn 1 -jo 1 -jp 1 -jq 1 -jr 1 -js 1 -jt 1 -ju 1 -jv 1 -jw 1 -jx 1 -jy 1 -jz 1 -ka 1 -kb 1 -kc 1 -kd 1 -ke 1 -kf 1 -kg 1 -kh 1 -ki 1 -kj 1 -kk 1 -kl 1 -km 1 -kn 1 -ko 1 -kp 1 -kq 1 -kr 1 -ks 1 -kt 1 -ku 1 -kv 1 -kw 1 -kx 1 -ky 1 -kz 1 -la 1 -lb 1 -lc 1 -ld 1 -le 1 -lf 1 -lg 1 -lh 1 -li 1 -lj 1 -lk 1 -ll 1 -lm 1 -ln 1 -lo 1 -lp 1 -lq 1 -lr 1 -ls 1 -lt 1 -lu 1 -lv 1 -lw 1 -lx 1 -ly 1 -lz 1 -ma 1 -mb 1 -mc 1 -md 1 -me 1 -mf 1 -mg 1 -mh 1 -mi 1 -mj 1 -mk 1 -ml 1 -mm 1 -mn 1 -mo 1 -mp 1 -mq 1 -mr 1 -ms 1 -mt 1 -mu 1 -mv 1 -mw 1 -mx 1 -my 1 -mz 1 -na 1 -nb 1 -nc 1 -nd 1 -ne 1 -nf 1 -ng 1 -nh 1 -ni 1 -nj 1 -nk 1 -nl 1 -nm 1 -nn 1 -no 1 -np 1 -nq 1 -nr 1 -ns 1 -nt 1 -nu 1 -nv 1 -nw 1 -nx 1 -ny 1 -nz 1 -oa 1 -ob 1 -oc 1 -od 1 -oe 1 -of 1 -og 1 -oh 1 -oi 1 -oj 1 -ok 1 -ol 1 -om 1 -on 1 -oo 1 -op 1 -oq 1 -or 1 -os 1 -ot 1 -ou 1 -ov 1 -ow 1 -ox 1 -oy 1 -oz 1 -pa 1 -pb 1 -pc 1 -pd 1 -pe 1 -pf 1 -pg 1 -ph 1 -pi 1 -pj 1 -pk 1 -pl 1 -pm 1 -pn 1 -po 1 -pp 1 -pq 1 -pr 1 -ps 1 -pt 1 -pu 1 -pv 1 -pw 1 -px 1 -py 1 -pz 1 -ra 1 -rb 1 -rc 1 -rd 1 -re 1 -rf 1 -rg 1 -rh 1 -ri 1 -rj 1 -rk 1 -rl 1 -rm 1 -rn 1 -ro 1 -rp 1 -rq 1 -rr 1 -rs 1 -rt 1 -ru 1 -rv 1 -rw 1 -rx 1 -ry 1 -rz 1 -sa 1 -sb 1 -sc 1 -sd 1 -se 1 -sf 1 -sg 1 -sh 1 -si 1 -sj 1 -sk 1 -sl 1 -sm 1 -sn 1 -so 1 -sp 1 -sq 1 -sr 1 -ss 1 -st 1 -su 1 -sv 1 -sw 1 -sx 1 -sy 1 -sz 1 -ta 1 -tb 1 -tc 1 -td 1 -te 1 -tf 1 -tg 1 -th 1 -ti 1 -tj 1 -tk 1 -tl 1 -tm 1 -tn 1 -to 1 -tp 1 -tq 1 -tr 1 -ts 1 -tt 1 -tu 1 -tv 1 -tw 1 -tx 1 -ty 1 -tz 1 -ua 1 -ub 1 -uc 1 -ud 1 -ue 1 -uf 1 -ug 1 -uh 1 -ui 1 -uj 1 -uk 1 -ul 1 -um 1 -un 1 -uo 1 -up 1 -uq 1 -ur 1 -us 1 -ut 1 -uu 1 -uv 1 -uw 1 -ux 1 -uy 1 -uz 1 -va 1 -vb 1 -vc 1 -vd 1 -ve 1 -vf 1 -vg 1 -vh 1 -vi 1 -vj 1 -vk 1 -vl 1 -vm 1 -vn 1 -vo 1 -vp 1 -vq 1 -vr 1 -vs 1 -vt 1 -vu 1 -vv 1 -vw 1 -vx 1 -vy 1 -vz 1 -wa 1 -wb 1 -wc 1 -wd 1 -we 1 -wf 1 -wg 1 -wh 1 -wi 1 -wj 1 -wk 1 -wl 1 -wm 1 -wn 1 -wo 1 -wp 1 -wq 1 -wr 1 -ws 1 -wt 1 -wu 1 -wv 1 -ww 1 -wx 1 -wy 1 -wz 1 -xa 1 -xb 1 -xc 1 -xd 1 -xe 1 -xf 1 -xg 1 -xh 1 -xi 1 -xj 1 -xk 1 -xl 1 -xm 1 -xn 1 -xo 1 -xp 1 -xq 1 -xr 1 -xs 1 -xt 1 -xu 1 -xv 1 -xw 1 -xx 1 -xy 1 -xz 1 -ya 1 -yb 1 -yc 1 -yd 1 -ye 1 -yf 1 -yg 1 -yh 1 -yi 1 -yj 1 -yk 1 -yl 1 -ym 1 -yn 1 -yo 1 -yp 1 -yq 1 -yr 1 -ys 1 -yt 1 -yu 1 -yv 1 -yw 1 -yx 1 -yy 1 -yz 1 -za 1 -zb 1 -zc 1 -zd 1 -ze 1 -zf 1 -zg 1 -zh 1 -zi 1 -zj 1 -zk 1 -zl 1 -zm 1 -zn 1 -zo 1 -zp 1 -zq 1 -zr 1 -zs 1 -zt 1 -zu 1 -zv 1 -zw 1 -zx 1 -zy 1 -zz

```

```

# sage_cmd [-ELEVATION Cstr] [-SLOPE Cstr]
# NCMUR [-Cstr] [-OCUR Cstr] [-METHOD Cstr]
# ELEVATION Cstr:
#   Cstr: (optional)
#   Slope
# -ASPECT Cstr:
#   Cstr: (optional)
#   Aspect
# -CURV Cstr:
#   Cstr: (optional)
#   Curvature
# -NCURV Cstr:
#   Cstr: (optional)
#   Plan Curvature
# -QCURV Cstr:
#   Cstr: (optional)
#   Profile Curvature
# -METHOD Cstr:
#   Cstr: (optional)
#   Method
# Available Choices:
# (1) Radian Slope (Fras et al. 1975)
# (2) Radian Slope (Fras et al. 1975)
# (3) Least Squares Fitted Plane (Fras)
# (4) Fit 1 Degree Polynom (Cheerberg 8)
# (5) Fit 2 Degree Polynom (Cheerberg 8)
# (6) Fit 3 Degree Polynom (Ravallik 19)
# Default: 5

```

Python

```

# Create a SAGA Grid
A = saga_api.SG_Create_Grid()
if A.Create(saga_api.SG_String(EA)) == 0:
    print "Error: could not load < EA + >"
    return 0

# Create a SAGA Grid
B = saga_api.SG_Create_Grid()
if B.Create(saga_api.SG_String(EB)) == 0:
    print "Error: could not load < EB + >"
    return 0

if A.is_Compatible(B) == 0:
    print "Error: grids < A + >
    return 0

if B.PixelSize == 1: # pixelwise
    C = saga_api.SG_Create_Grid()
    for x in range(0, C.GetSizeX()):
        for y in range(0, C.GetSizeY()):
            C.Set_Value(x, y, C.Get_Value(x, y) * C.Get_Value(x, y))
    C.Save(saga_api.SG_String(

```

R Script

```

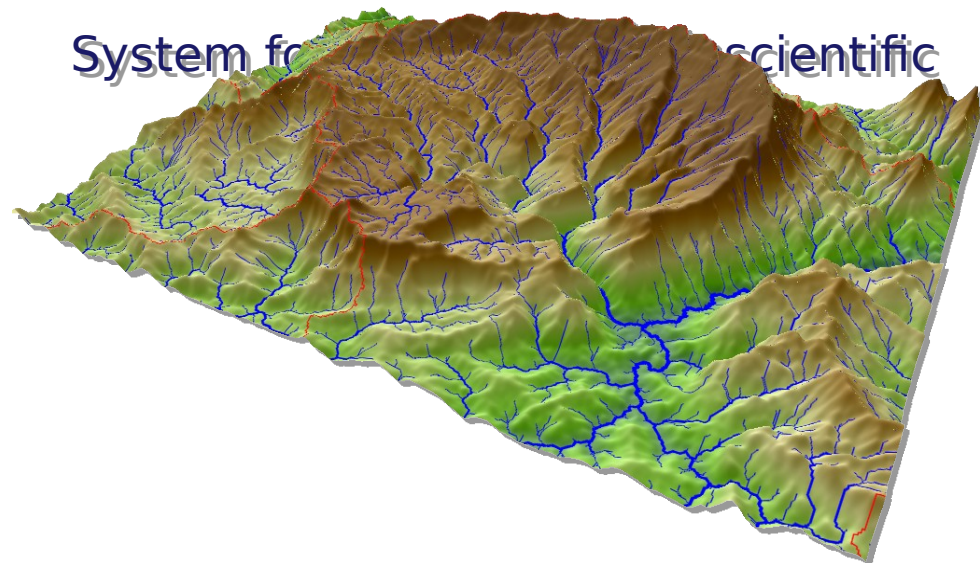
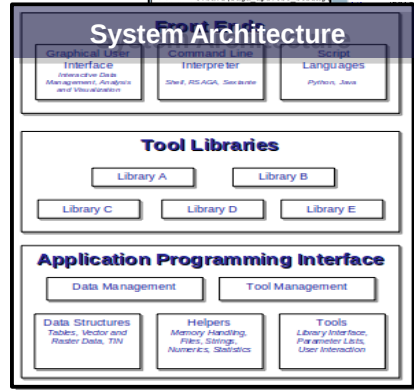
# Convert the map to SAGA grid format:
write.asc(saga_grid("meuse_dist.asc"), na.value=-1)
saga.asc.to.sgrid(in.grid="meuse_dist.asc", out.sgrid="meuse_dist.sgr")
writeOGR(meuse[["meuse", "logistic"], "meuse.shp", "meuse", "ESRI Shapefile"])

saga.lib("geostatistica_kriging", module=5, param=list(OR:
  list(in.sgrid="OK_zinc.sgrid", out.grid="OK_zinc.asc", out:
  readGDAL("OK_zinc.asc")$band1)
  var1.pred, ok.zinc$OKA, pol=19))

saga.lib("geostatistica_kriging", module=8, param=list(OR:
  list(in.sgrid="BK_zinc.sgrid", out.grid="BK_zinc.asc", out:
  readGDAL("BK_zinc.asc")$band1)
  var1.pred, bk.zinc$BKAA, pol=19))

option
error(lib="pj_proj4", 2, param=list(SOURCE_PROJ=NLL_RD, TA
error(lib="io_grid_image", 0, param=list(GRID="meuse_dist

```



Michael Bock, Olaf Conrad, Volker Wichmann

SAGA | System for Automated Geoscientific Analyses

Ways of Automation

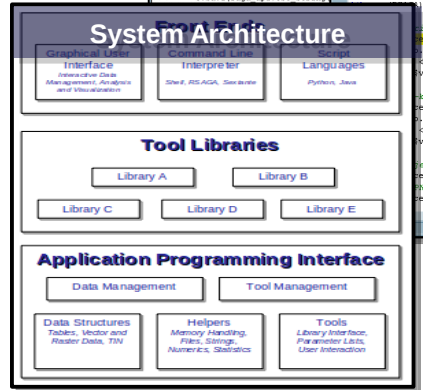
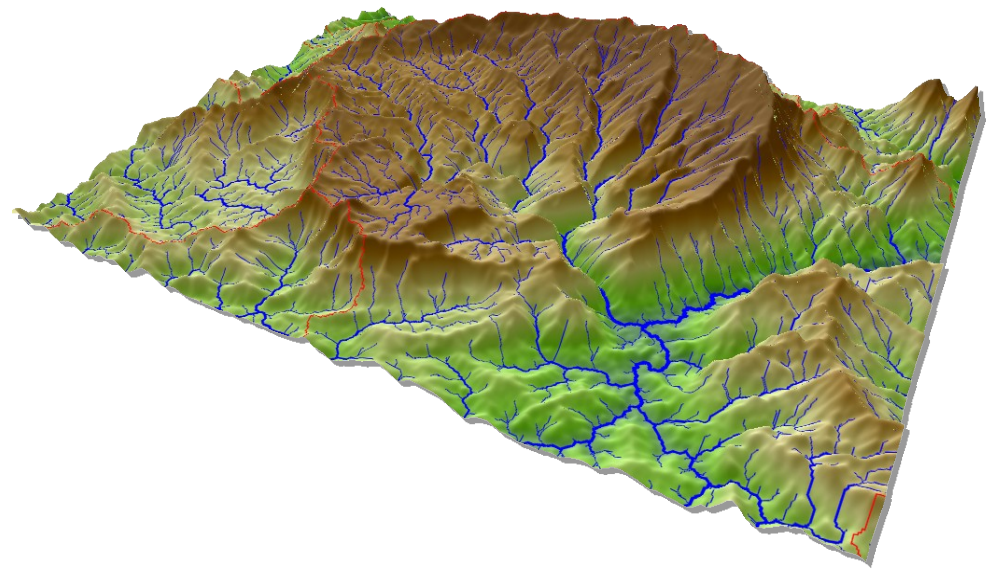
Tool Chains

The screenshot displays the SAGA GIS desktop environment. On the left is a 'Layers' panel with a tree view of data layers. The main workspace is divided into several panes: 'Overview' showing a 2D map with elevation contours and points; 'Landscape Imagery (3D View)' showing a 3D perspective of the terrain; 'Elevation (m)' showing a histogram of elevation values; 'Grid Profile' showing a line profile across the terrain; and 'Statistics' showing a table of statistical data for a selected grid.

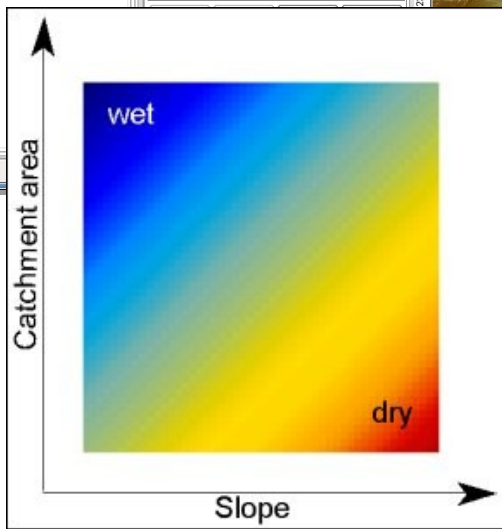
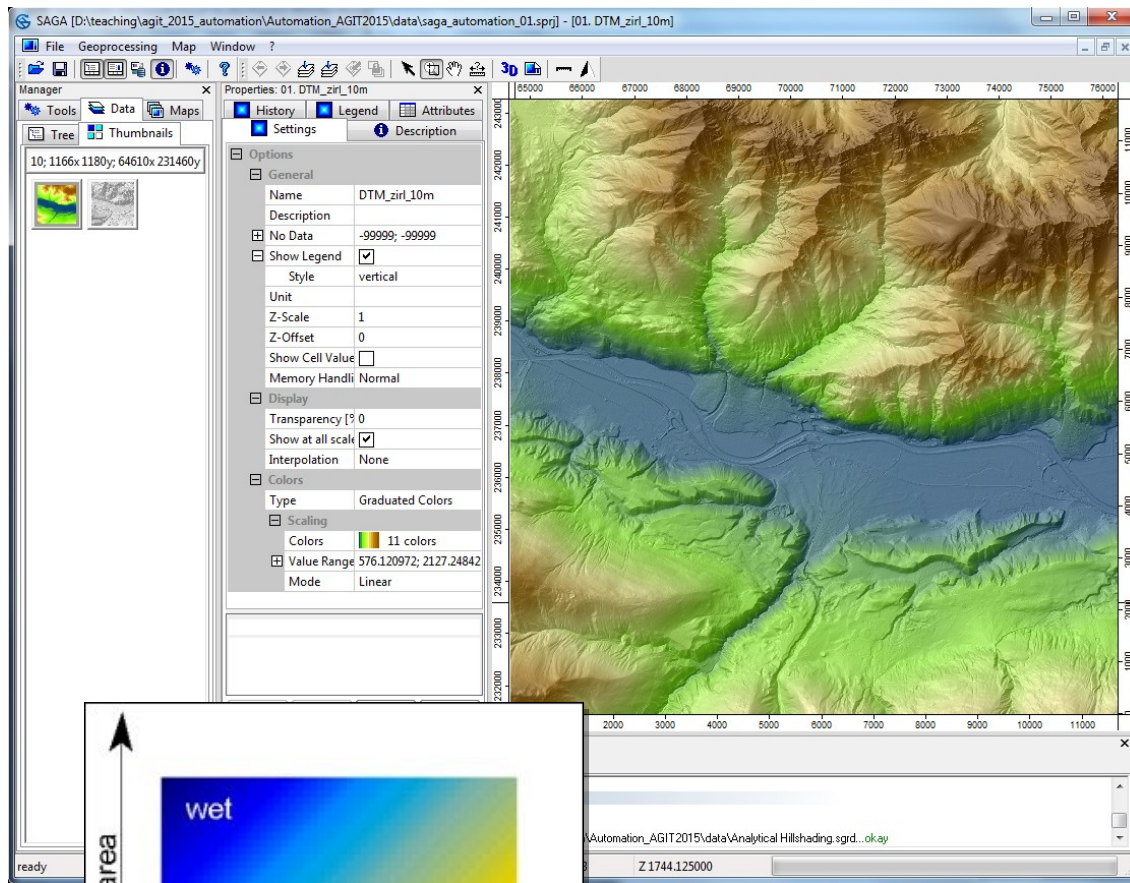
JAH	TREND	RESIDUAL	
1	777.900000	936.878178	-158.978178
2	787.500000	826.413769	-40.913769
3	818.300000	837.846254	-6.646254
4	872.800000	886.270589	-23.470589
5	930.700000	1020.692982	-89.992982
6	1052.900000	967.853309	-114.953309
7	691.300000	789.482936	-98.182936
8	644.600000	622.830436	21.769564
9	664.000000	842.279551	-178.279551
10	689.400000	822.848515	-133.448515
11	740.100000	796.955485	-56.855485
12	827.600000	884.141140	-56.541140

Overlaid on the interface are three windows illustrating automation:

- Command Line:** Shows a terminal window with the command `saga_cmd -k morphology` and its output, including a list of available tools like `ELEVATION`, `SLOPE`, and `ASPECT`.
- Python:** Shows a Python script using the `saga_api` module to create a grid and perform calculations.
- R Script:** Shows an R script using the `sagaR` package to interact with SAGA GIS, including commands like `readGDAL` and `writeGDAL`.

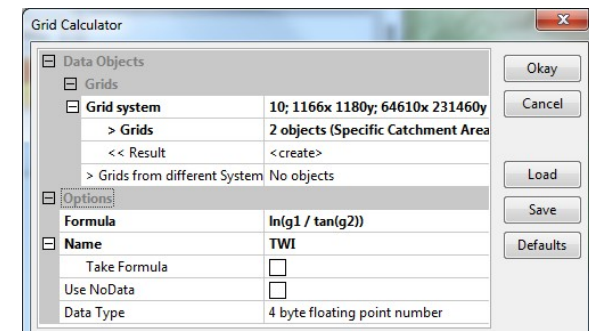


TWI | Topographic Wetness Index

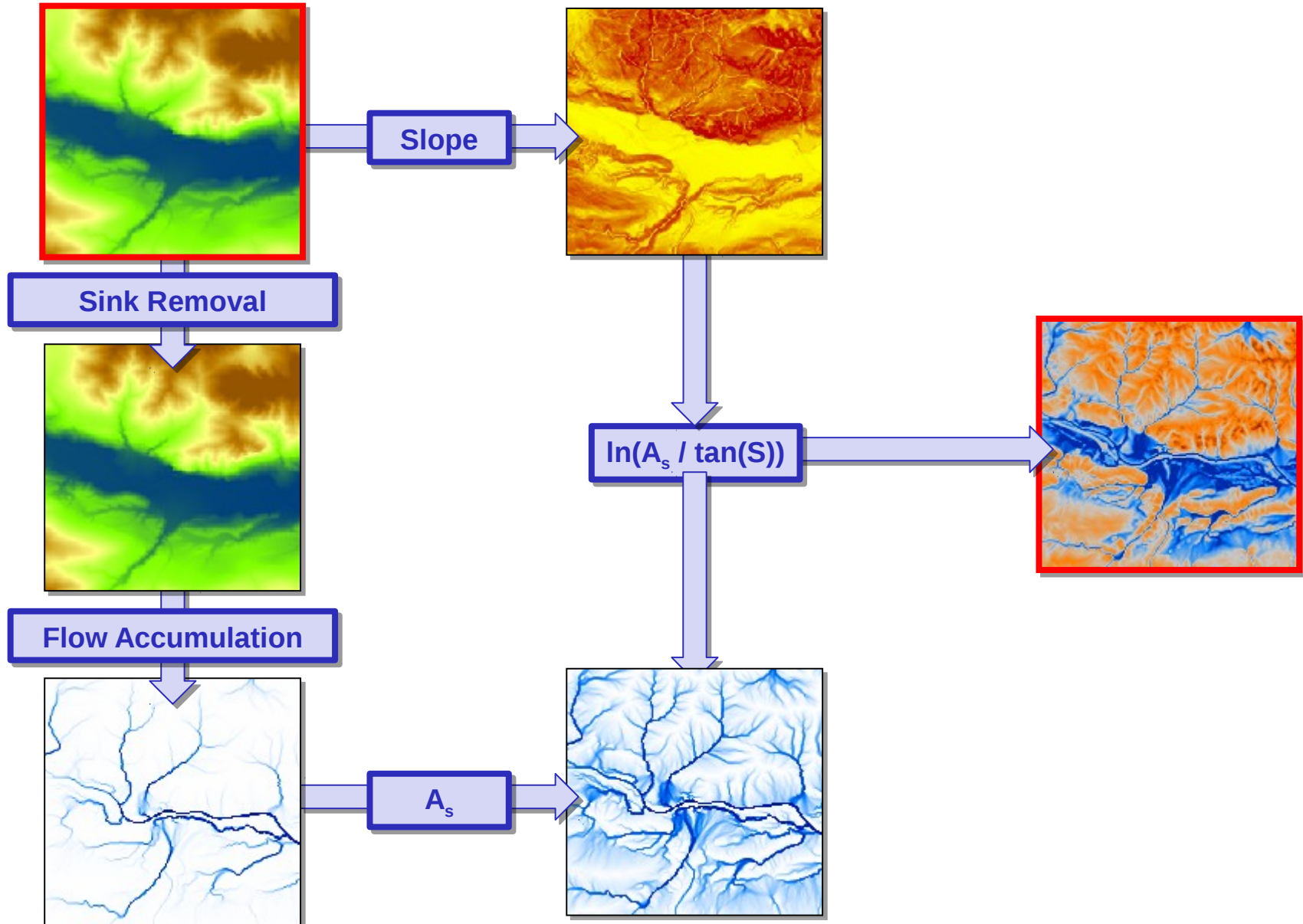


$$TWI = \ln \left(\frac{A}{\tan(\beta)} \right)$$

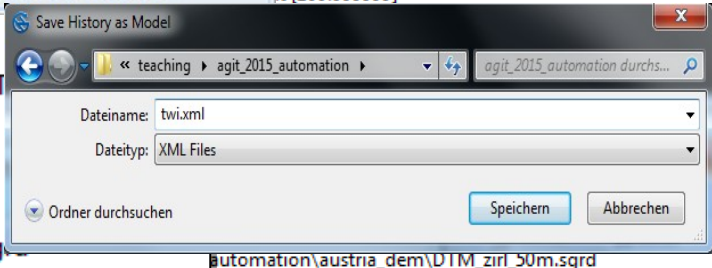
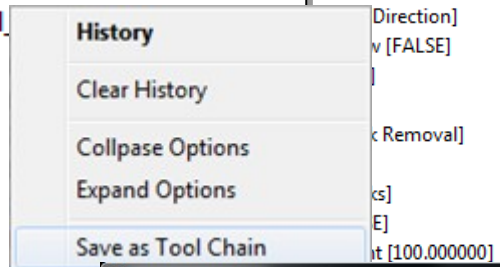
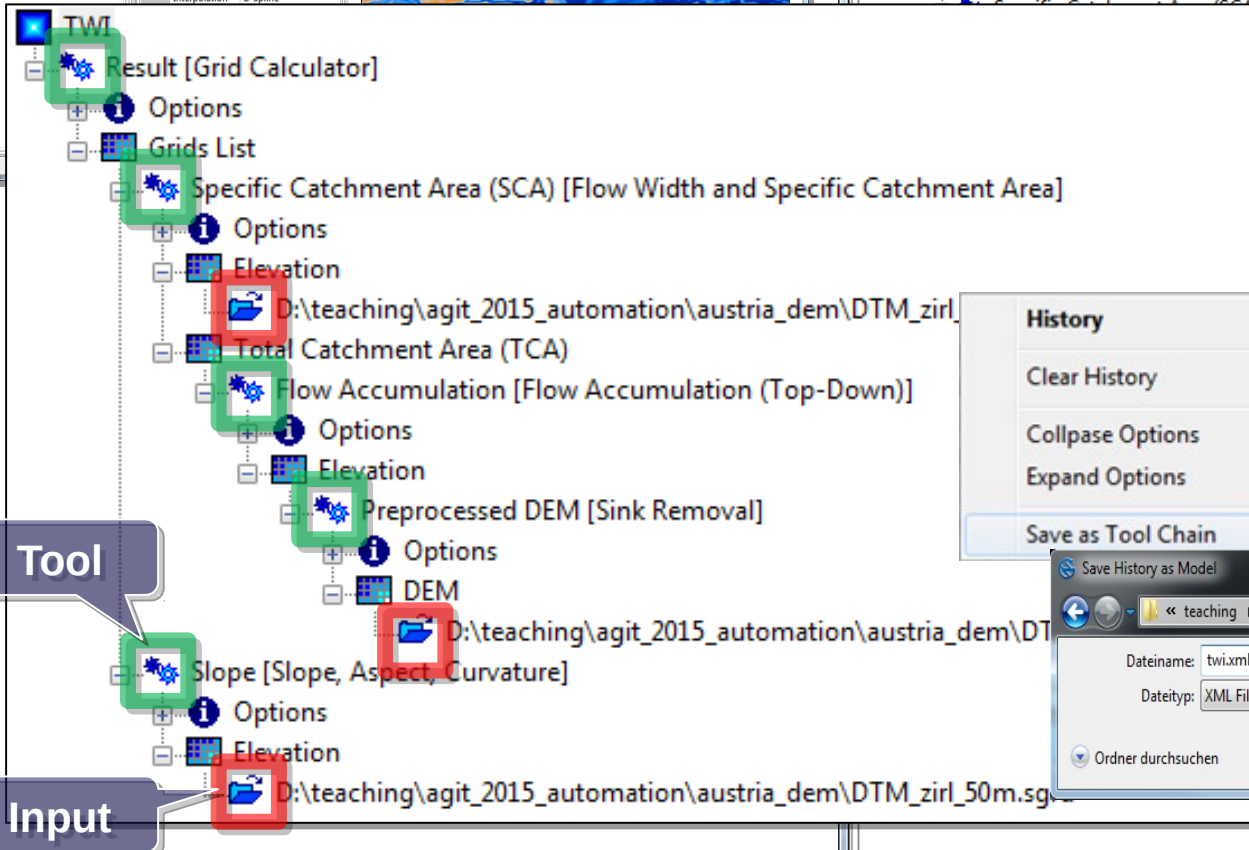
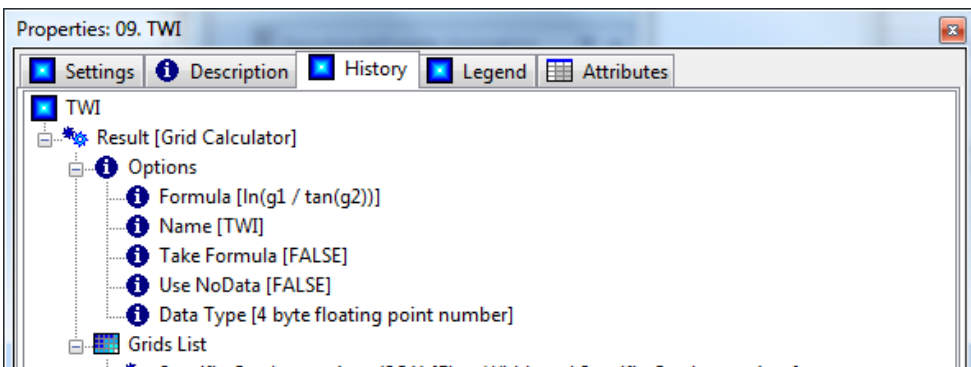
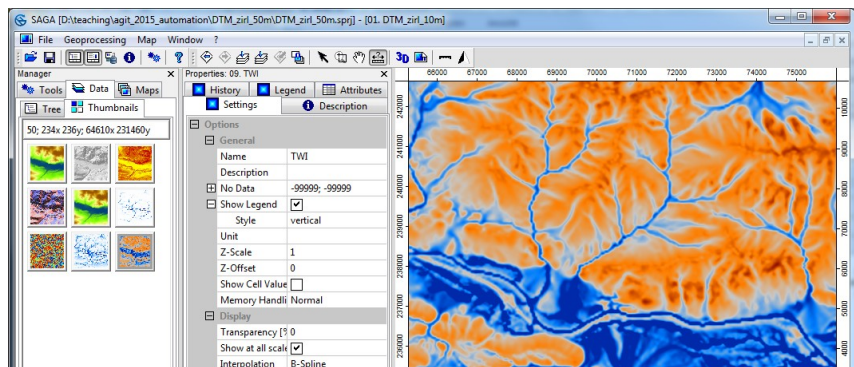
- $TWI = \log(A_s / \tan(S))$
 - A_s = Specific Catchment Area
 - S = Slope Angle
- Input Data
 - DEM
- Tools
 1. Slope, Aspect, Curvature
 2. Sink Removal
 3. Flow Accumulation
 4. Flow Width and Specific Catchment Area
 5. Grid Calculator



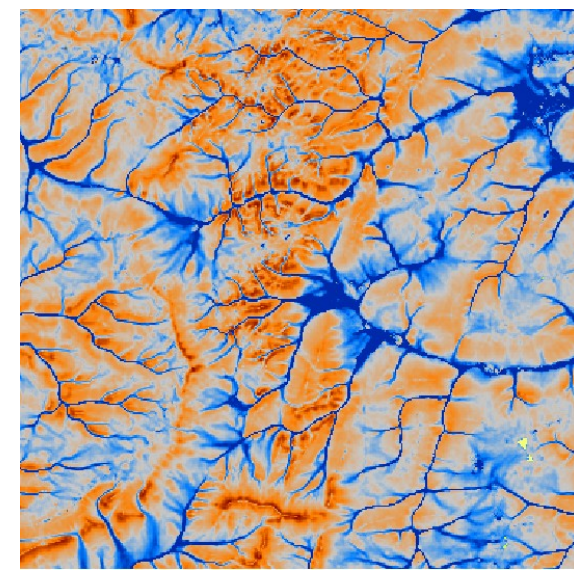
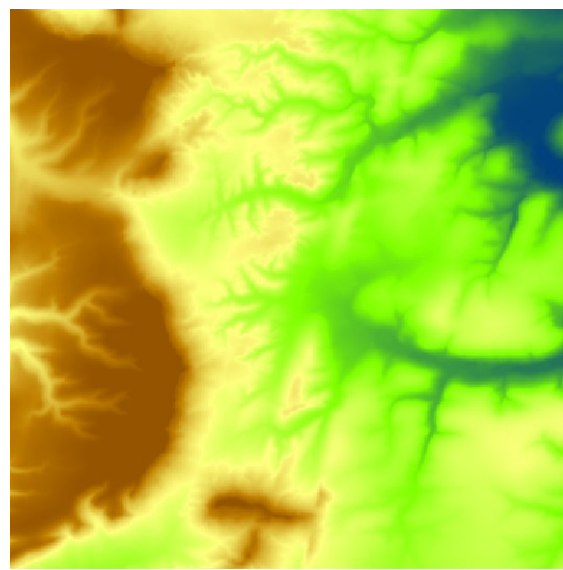
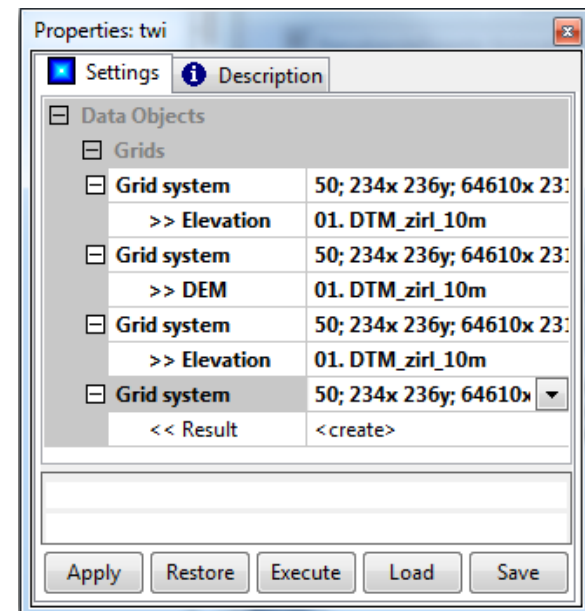
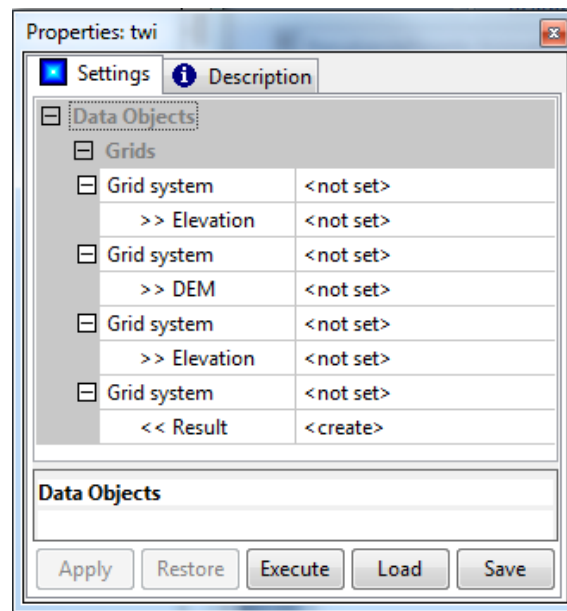
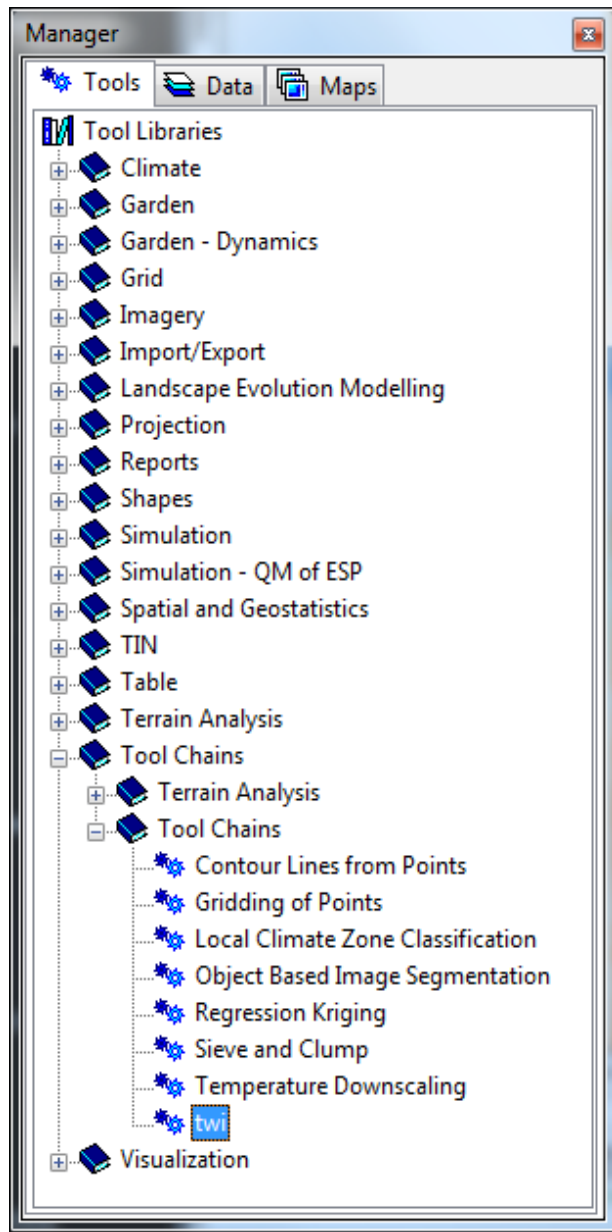
TWI | Workflow



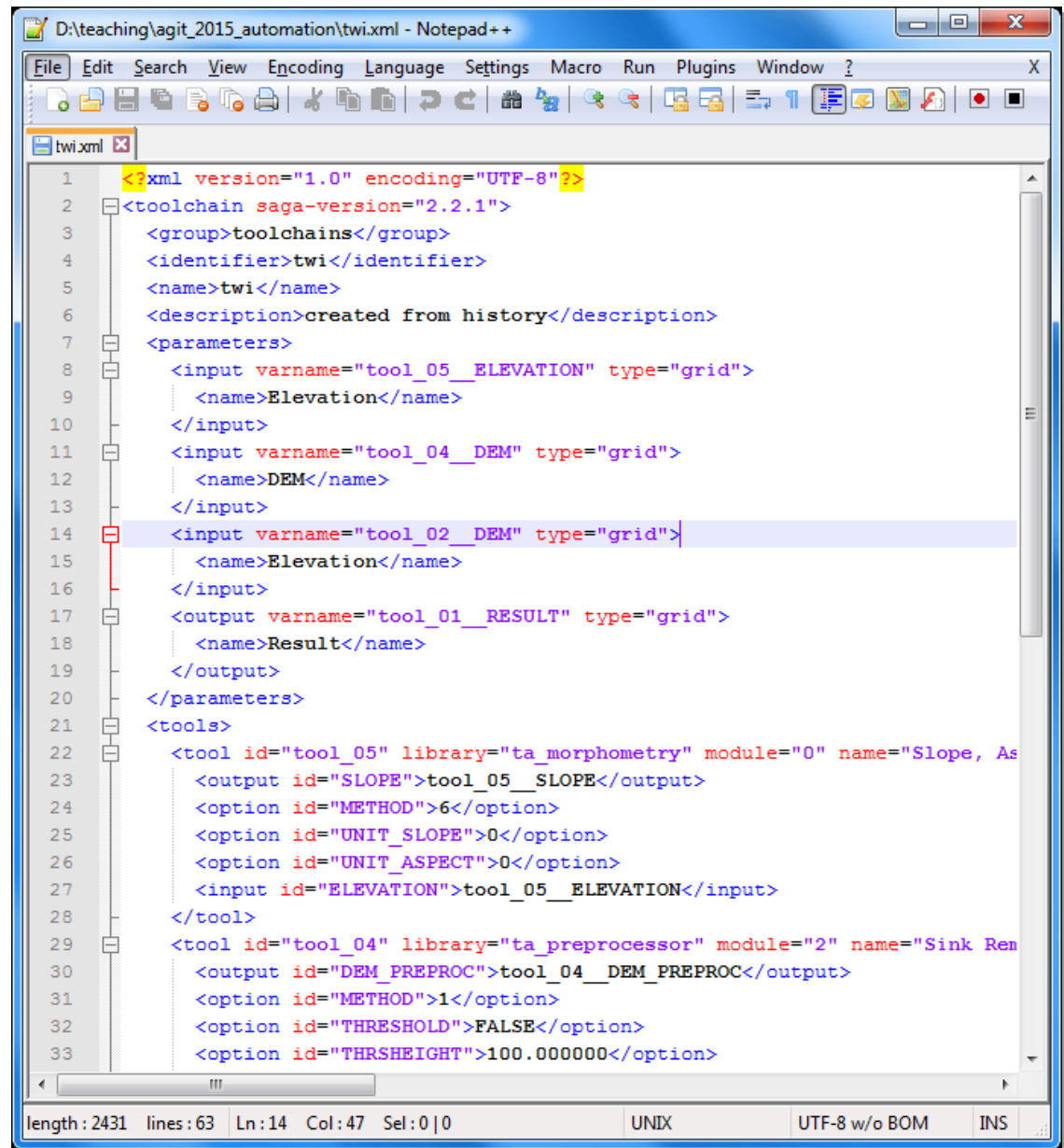
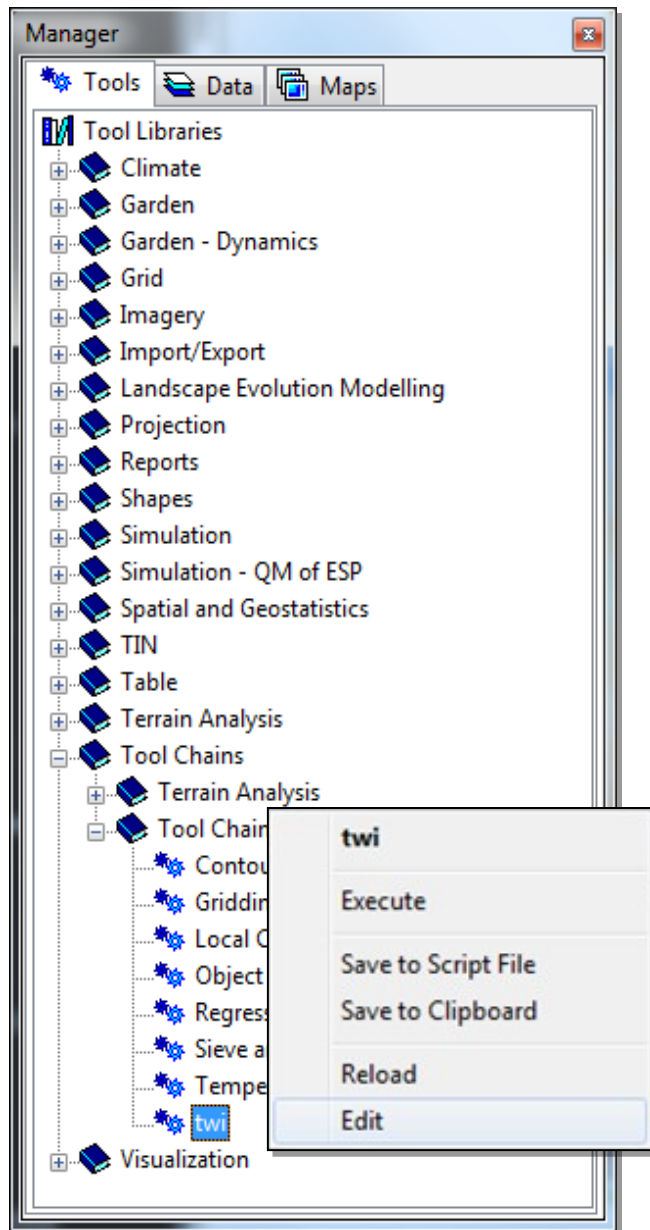
TWI | Creation History



TWI | Apply Tool Chain to Another Data Set



TWI | Edit Tool Chain



Tool Chains | Main Structure

<toolchain>

- Attributes

- saga-version

- Content

- <group>
- <identifier>
- <name>
- <author>
- <description>
- <menu>
- <parameters>
- <tools>

```
<?xml version="1.0" encoding="UTF-8"?>
<toolchain saga-version="2.2.1">
  <group>toolchains</group>
  <identifier>twi</identifier>
  <name>twi</name>
  <description>created from history</description>
  <parameters>
    <option varname="GRID_SYSTEM" type="grid_system">
      <input varname="DEM" type="grid" parent="GRID_SYSTEM">
        <output varname="TWI" type="grid" parent="GRID_SYSTEM">
          </output>
        </input>
      </option>
    </parameters>
  <tools>
    <tool id="tool_05" library="ta_morphometry" module="0" name=">
      <tool id="tool_04" library="ta_preprocessor" module="2" name=">
        <tool id="tool_03" library="ta_hydrology" module="0" name=">
          <tool id="tool_02" library="ta_hydrology" module="19" name=">
            <tool id="tool_01" library="grid_calculus" module="1" name=">
              </tool>
            </tool>
          </tool>
        </tool>
      </tool>
    </tools>
</toolchain>
```

Tool Chains | General Key Words

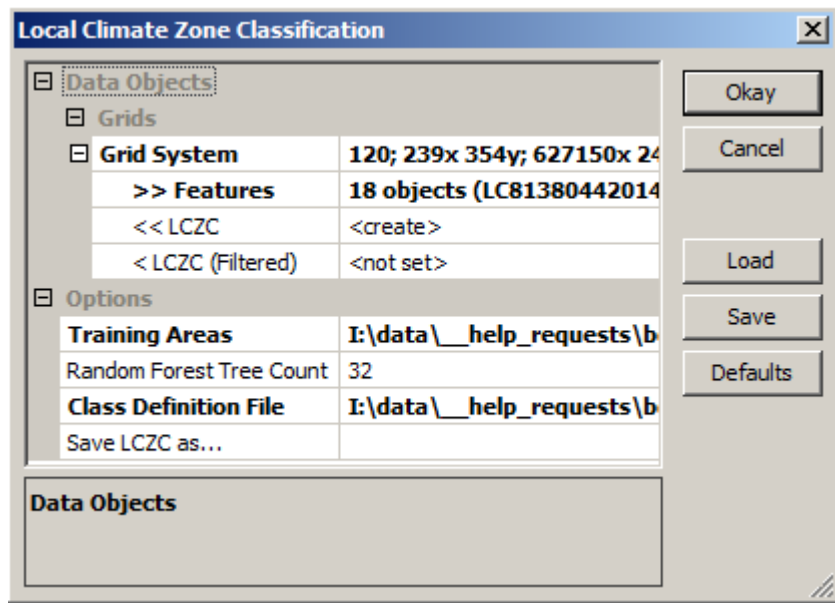
- `<group>`
 - Content: A category to which the tool belongs to
- `<identifier>` [obligatory]
 - Content: unique(!) id, used to identify and run the tool from other scripts or tool chains
- `<name>`
 - Content: human readable name of the tool
- `<author>`
 - Content: the authors, copyrights, ...
- `<description>`
 - Content: more explicit description.
- `<menu>`
 - Attributes:
 - `absolute=true/false` (default=false)
 - Content: menu path of the tool (`saga_gui`)

Tool Chains | The Parameters Interface

<parameters>

Content

- <input>
- <output>
- <option>



```

<parameters>
  <option varname="GRID_SYSTEM" type="grid_system">
    <name>Grid System</name>
  </option>
  <option varname="FILE_TRAINING" type="file">
    <name>Training Areas</name>
    <filter>RML/KMZ Files|*.kml;*.kmz|All Files|*.*</filter>
  </option>
  <option varname="RF_TREE_COUNT" type="integer">
    <name>Random Forest Tree Count</name>
    <value min="1">32</value>
    <description>How many trees to create?</description>
  </option>
  <option varname="FILTER_RADIUS" type="integer">
    <condition type="exists">LCZC_FILTERED</condition>
    <name>Majority Filter Radius</name>
    <value min="1">2</value>
  </option>
  <option varname="FILE_CLASS_DEF" type="file">
    <name>Class Definition File</name>
    <filter>Table Files|*.txt;*.dbf;*.csv|All Files|*.*</filter>
  </option>
  <option varname="FILE_LCZC" type="file" save="true">
    <name>Save LCZC as...</name>
    <value/>
    <filter>KMZ Files|*.kmz|All Files|*.*</filter>
  </option>
  <option varname="FILE_LCZC_FILTERED" type="file" save="true">
    <condition type="exists">LCZC_FILTERED</condition>
    <name>Save LCZC (Filtered) as...</name>
    <value/>
    <filter>KMZ Files|*.kmz|All Files|*.*</filter>
  </option>
  <input varname="FEATURES" type="grid_list" parent="GRID_SYSTEM">
    <name>Features</name>
  </input>
  <output varname="LCZC" type="grid" parent="GRID_SYSTEM">
    <name>LCZC</name>
  </output>
  <output varname="LCZC_FILTERED" type="grid" optional="true" parent="LCZC">
    <name>LCZC (Filtered)</name>
  </output>
</parameters>

```

Tool Chains | Input & Output Data

<input>, <output>

- **Attributes**
 - varname: unique(!) variable id
 - type: data set type
 - parent: e.g. a single grid system for grids
- **Content**
 - <condition>
 - conditionally en-/disable the parameter
 - <name>
 - human readable name
 - <description>
 - more explicit parameter description

Type	
grid	
table	
shapes	
tin	
points	
grid_list	
table_list	
shapes_list	
tin_list	
points_list	

Tool Chains | Options

<option>

- **Attributes**
 - varname: unique(!) variable id
 - type: variable type
 - parent: e.g. a table for table field selection
- **Content**
 - <condition>
 - conditionally en-/disable the parameter
 - <name>
 - human readable name
 - <description>
 - more explicit parameter description
 - <...>
 - dependent on the option type, various type specific attributes and content keys might apply, e.g.:
 - <value> default value for numeric and text variables
 - <filter> format filter for file selection

Type	
node	
boolean	
integer	
double	
degree	
range	
choice	
text	
long_text	
file	
table_field	
table_fields	
grid_system	

Tool Chains | The Tools Section

<tools>

- Content

- <tool>
- <condition>

```

<tools>
  <tool library="ta_morphometry" module="0" name="Slope, Aspect, Curvature">
  <tool library="ta_preprocessor" module="2" name="Sink Removal">
  <tool library="ta_hydrology" module="0" name="Flow Accumulation (Top-Down)">
  <tool library="ta_hydrology" module="19" name="Flow Width and Specific Cat">
  <tool library="grid_calculus" module="1" name="Grid Calculator">
</tools>

```

<tool>

- Attributes

- library: tool library of the tool
- module: tool identifier (unique!)
- name: unused, good for better reading

- Content

- <input>/<output>/<option>
 - Attributes
 - id: tool's parameter identifier
 - varname=true/false (options only, default=false)
 - Content
 - varname or value (options only)

```

<tool library="ta_morphometry" module="0" name="Slo
  <output id="SLOPE">tool_05_SLOPE</output>
  <input id="ELEVATION">DEM</input>
  <option id="METHOD">6</option>
  <option id="UNIT_SLOPE">0</option>
  <option id="UNIT_ASPECT">0</option>
</tool>

```

TWI | One Single Input instead of Three

The image shows two XML configuration files side-by-side. The left file represents the original state with three separate input parameters for 'Elevation', 'DEM', and 'Elevation'. The right file shows the updated state where these three inputs are consolidated into a single input parameter named 'DEM'. Blue arrows indicate the mapping from the old parameters to the new one. A callout box on the right states: "One single input grid named 'DEM'".

Left XML (Original):

```
<parameters>
  <input varname="tool_05_ELEVATION" type="grid">
    <name>Elevation</name>
  </input>
  <input varname="tool_04_DEM" type="grid">
    <name>DEM</name>
  </input>
  <input varname="tool_02_DEM" type="grid">
    <name>Elevation</name>
  </input>
  <output varname="tool_01_RESULT" type="grid">
    <name>Result</name>
  </output>
</parameters>
<tools>
  <tool id="tool_05" library="ta_morphometry" module="0"
    <output id="SLOPE">tool_05_SLOPE</output>
    <option id="METHOD">6</option>
    <option id="UNIT_SLOPE">0</option>
    <option id="UNIT_ASPECT">0</option>
    <input id="ELEVATION">tool_05_ELEVATION</input>
  </tool>
  <tool id="tool_04" library="ta_preprocessor" module="2"
    <output id="DEM_PREPROC">tool_04_DEM_PREPROC</output>
    <option id="METHOD">1</option>
    <option id="THRESHOLD">FALSE</option>
    <option id="THRSHEIGHT">100.000000</option>
  </tool>
  <tool id="tool_02" library="ta_preprocessor" module="19"
    <output id="DEM_PREPROC">tool_02_DEM_PREPROC</output>
    <option id="METHOD">1</option>
    <option id="THRESHOLD">FALSE</option>
    <option id="THRSHEIGHT">100.000000</option>
  </tool>
</tools>
```

Right XML (Updated):

```
<parameters>
  <input varname="DEM" type="grid">
    <name>Elevation</name>
  </input>
  <output varname="tool_01_RESULT" type="grid">
    <name>Result</name>
  </output>
</parameters>
<tools>
  <tool id="tool_05" library="ta_morphometry" module="0"
    <output id="SLOPE">tool_05_SLOPE</output>
    <option id="METHOD">6</option>
    <option id="UNIT_SLOPE">0</option>
    <option id="UNIT_ASPECT">0</option>
    <input id="ELEVATION">DEM</input>
  </tool>
  <tool id="tool_04" library="ta_preprocessor" module="2"
    <output id="DEM_PREPROC">tool_04_DEM_PREPROC</output>
    <option id="METHOD">1</option>
    <option id="THRESHOLD">FALSE</option>
    <option id="THRSHEIGHT">100.000000</option>
  </tool>
  <tool id="tool_02" library="ta_preprocessor" module="19"
    <output id="DEM_PREPROC">tool_02_DEM_PREPROC</output>
    <option id="METHOD">1</option>
    <option id="THRESHOLD">FALSE</option>
    <option id="THRSHEIGHT">100.000000</option>
  </tool>
</tools>
```

Properties: twi (Left):

Grid system	<not set>
>> Elevation	<not set>
>> DEM	<not set>
>> Elevation	<not set>
<< Result	<create>

Properties: twi (Right):

Grid system	<not set>
>> Elevation	<not set>
<< Result	<create>

TWI | Using One Single Grid System for Input and Output

```

<parameters>
  <input varname="DEM" type="grid">
    <name>Elevation</name>
  </input>
  <output varname="tool_01_RESULT" type="grid">
    <name>Result</name>
  </output>
</parameters>
<tools>
  <tool id="tool_05" library="ta_morphometry" module="0">
    <output id="SLOPE">tool_05_SLOPE</output>
    <option id="METHOD">6</option>
    <option id="UNIT_SLOPE">0</option>
    <option id="UNIT_ASPECT">0</option>
    <input id="ELEVATION">DEM</input>
  </tool>
  <tool id="tool_04" library="ta_preprocessor" module="2">
    <output id="DEM_PREPROC">tool_04_DEM_PREPROC</output>
    <option id="METHOD">1</option>
    <option id="THRESHOLD">FALSE</option>
    <option id="THRSHEIGHT">100.000000</option>
    <input id="DEM">DEM</input>
  </tool>
  <tool id="tool_03" library="ta_hydrology" module="0" n
  <tool id="tool_02" library="ta_hydrology" module="19">
    <output id="SCA">tool_02_SCA</output>
    <option id="METHOD">2</option>
    <input id="DEM">DEM</input>
    <input id="TCA">tool_03_CAREA</input>
  </tool>

```

One single grid system as parent for all grids

```

<?xml version="1.0" encoding="UTF-8">
<toolchain saga-version="2.2.1">
  <group>toolchains</group>
  <identifier>twi</identifier>
  <name>twi</name>
  <description>created from history</description>
  <parameters>
    <option varname="GRID_SYSTEM" type="grid_system">
      <name>Grid System</name>
    </option>
    <input varname="DEM" type="grid" parent="GRID_SYSTEM">
      <name>Elevation</name>
    </input>
    <output varname="TWI" type="grid" parent="GRID_SYSTEM">
      <name>Result</name>
    </output>
  </parameters>
  <tools>
    <tool id="tool_05" library="ta_morphometry" module="0" na
    <tool id="t
    <tool id="t
    <tool id="t
    <tool id="t
    <output i
    <option i
    <option i
    <input i
    <input i
  </tool>
  </tools>
</toolchain>

```

Properties: twi

Settings | Description

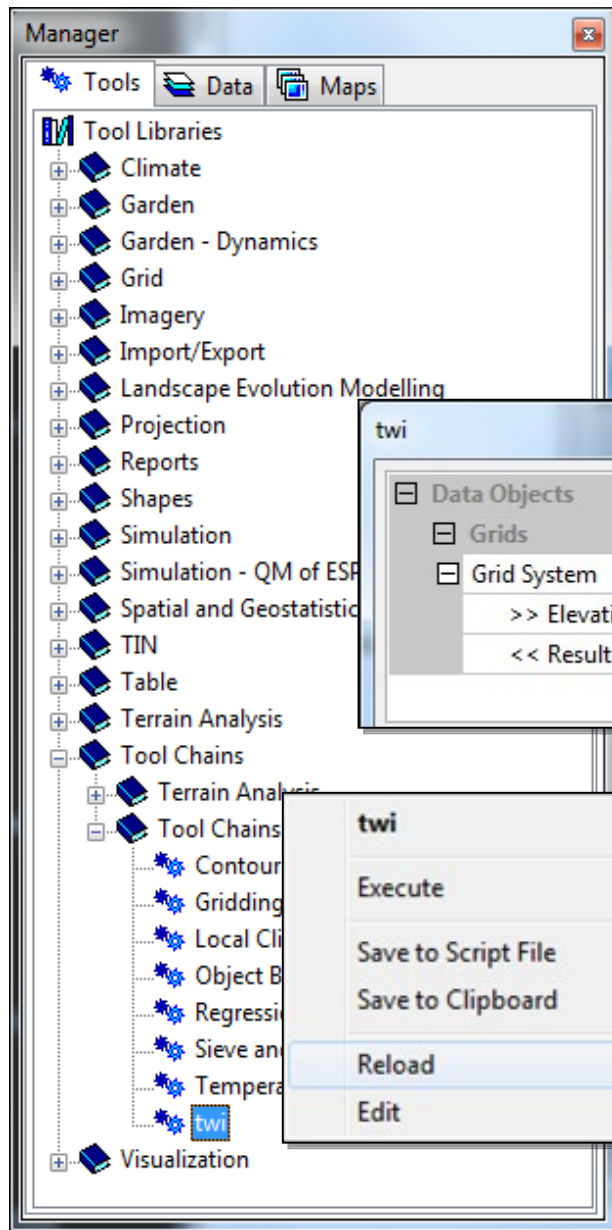
Data Objects

- Grids
 - Grid System

>> Elevation	<not set>
<< Result	<create>

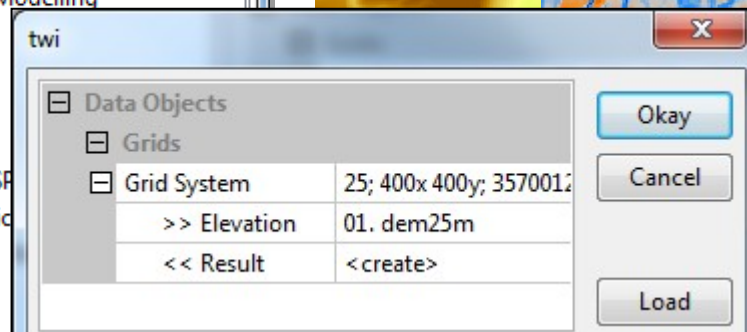
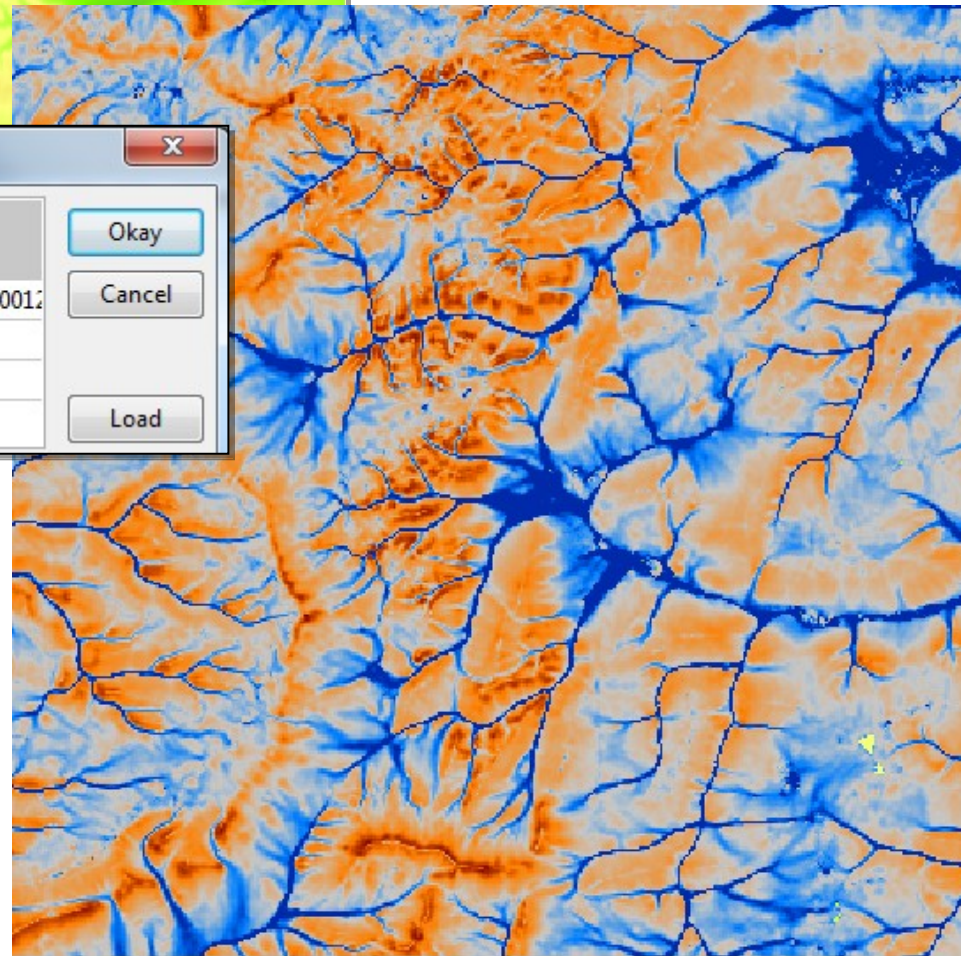
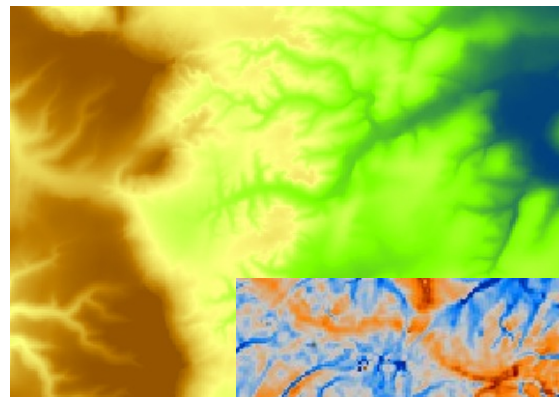
Apply Restore Execute Load Save

TWI | Reload Changed Tool Chain



The Manager window displays the following tool libraries and chains:

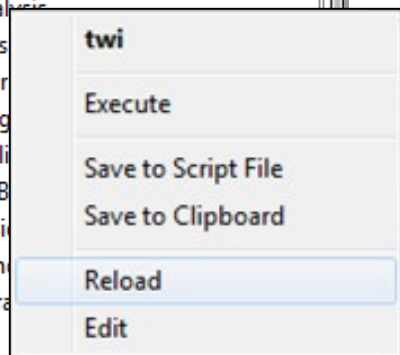
- Tools
- Data
- Maps
- Tool Libraries
 - Climate
 - Garden
 - Garden - Dynamics
 - Grid
 - Imagery
 - Import/Export
 - Landscape Evolution Modelling
 - Projection
 - Reports
 - Shapes
 - Simulation
 - Simulation - QM of ESP
 - Spatial and Geostatistic
 - TIN
 - Table
 - Terrain Analysis
 - Tool Chains
 - Terrain Analysis
 - Tool Chains
 - Contour
 - Gridding
 - Local Cl
 - Object B
 - Regressi
 - Sieve an
 - Tempera
 - twi
- Visualization



The 'twi' dialog box displays the following configuration:

Data Objects	
Grids	
Grid System	25; 400x 400y; 3570012
>> Elevation	01. dem25m
<< Result	<create>

Buttons: Okay, Cancel, Load



Context menu for 'twi':

- twi
- Execute
- Save to Script File
- Save to Clipboard
- Reload
- Edit

OBIA | Object Based Image Analysis

The screenshot displays the SAGA GIS interface with several components:

- Top Left:** A map window titled "05. Landsat Band 5" showing a satellite image with a coordinate grid (X: 564000, 568000, 572000; Y: 5716000, 5712000, 5708000).
- Top Right:** A code editor window containing a sequence of tool commands for OBIA:

```
<tools>
  <tool library="imagery segmentation" module="2" name="Seed Generation">
  <tool library="imagery segmentation" module="3" name="Simple Region Growing">
  <tool library="grid filter" module="6" name="Majority Filter">
  <tool library="shapes_grid" module="6" name="Vectorising Grid Classes">
  <tool library="shapes_grid" module="2" name="Grid Statistics for Polygons">
  <tool library="table calculus" module="14" name="Cluster Analysis">
  <tool library="shapes polygons" module="5" name="Polygon Dissolve">
  <tool library="shapes polygons" module="10" name="Polygon Parts to Separate Polygons">
</tools>
```
- Center:** A dialog box titled "Object Based Image Segmentation" with the following settings:

Data Objects	
Grids	
Grid System	28.5; 580x 510y; 561749.25x 5...
Features	7 objects (Landsat Band 1 (t...
Shapes	
Objects	<create>
Options	
Band Width	2
Generalization	1
Number of Clusters	12
Normalize	<input type="checkbox"/>
- Bottom Left:** A map window titled "04. Landsat Band 4" showing a satellite image with a coordinate grid (X: 564000, 568000; Y: 5716000, 5712000, 5708000).
- Bottom Center:** A map window showing a segmented grid with a coordinate grid (X: 0, 4000, 8000, 12000; Y: 5712000, 5708000).
- Bottom Right:** A map window showing the final segmented polygons with a coordinate grid (X: 0, 4000, 8000, 12000).

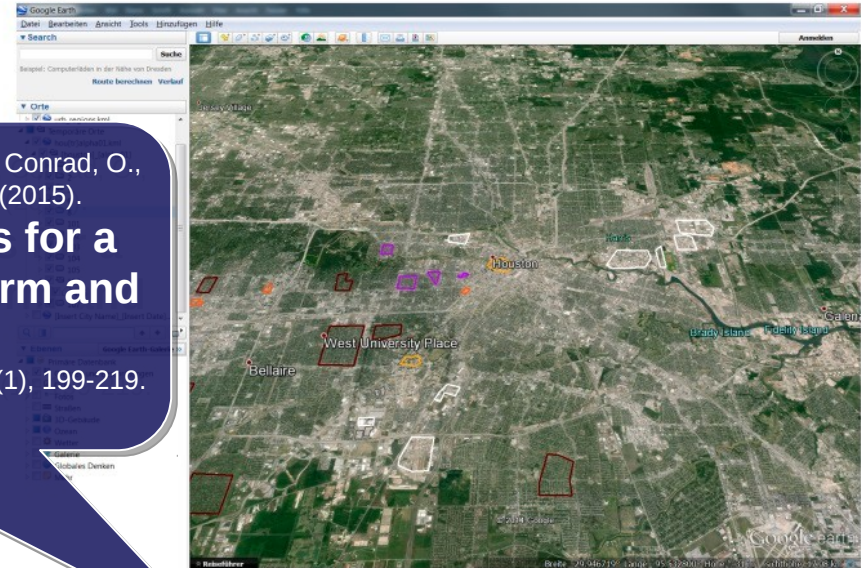
LCZC | Local Climate Zone Classification

a)

USGS Earth Explorer search results showing multiple data sets for Local Climate Zones (LCZ) in Houston, Texas. The results list acquisition dates and paths for various LCZ categories.

Bechtel, B., Alexander, P. J., Böhner, J., Ching, J., Conrad, O., Feddema, J., Gerald, M., See, L., Stewart, I. (2015).
Mapping local climate zones for a worldwide database of the form and function of cities.
 ISPRS International Journal of Geo-Information, 4(1), 199-219.
 doi:10.3390/ijgi4010199.

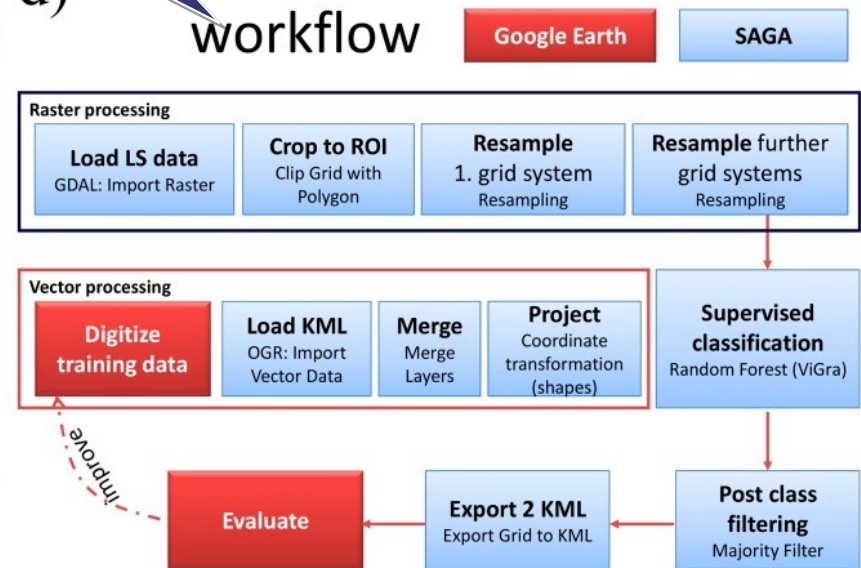
b)



c)

SAGA GIS software interface showing a city map with various processing tools and data layers. The interface includes a menu bar, toolbars, and a main workspace displaying a grayscale city map with colored polygons and a grid overlay.

d)



LCZC | Local Climate Zone Classification

a)

USGS EarthExplorer search results showing a list of satellite imagery data sets. The search criteria summary indicates 4 results. The results list includes acquisition dates, paths, and coordinates for various satellite passes.

Bechtel, B., Alexander, P. J., Böhner, J., Ching, J., Conrad, O., Feddema, J., Gerald, M., See, L., Stewart, I. (2015).
Mapping local climate zones for a worldwide database of the form and function of cities.
 ISPRS International Journal of Geo-Information, 4(1), 199-219.
 doi:10.3390/ijgi4010199.

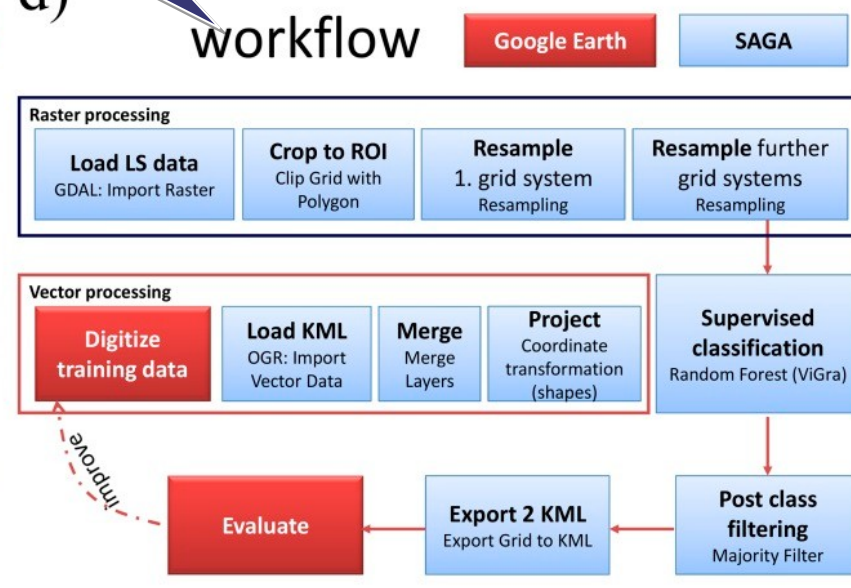
b)

Google Earth interface showing a satellite view of a city with several colored polygons (red, purple, yellow) overlaid on different urban areas, representing local climate zones.

c)

SAGA GIS software interface showing a city map with various processing tools and data layers. The interface includes a menu bar, toolbars, and a main workspace displaying a grayscale city map with colored polygons.

d)



Tool Chains | Conditional Tool Execution

- `<condition>`
 - type
 - value
 - variable

Look-up table will be loaded and applied, if the user selected a file

Majority filter will be applied, if the user chose to create the resulting grid

```

<tool library="io gdal" module="3" name="OGR: Import Vector Data">
<tool library="shapes tools" module="2" name="Merge Layers">
<tool library="grid tools" module="32" name="Select Grid from List">
<tool library="pj proj4" module="2" name="Coordinate Transformation (Shapes)">
<tool library="imagery vigra" module="9" name="Random Forest Classification (ViGrA)">
<condition type="not_equal" value="" variable="FILE_CLASS_DEF">
  <tool library="io table" module="1" name="Import Text Table">
  <tool library="grid visualisation" module="10" name="Select Look-up Table for Grid
</condition>
<condition type="not_equal" value="" variable="FILE_LCZC">
  <tool library="io grid image" module="2" name="Export Grid to KML">
</condition>
<condition type="exists" variable="LCZC_FILTERED">
  <tool library="grid filter" module="6" name="Majority Filter">
  <tool library="grid visualisation" module="10" name="Select Look-up Table for Grid
  <condition type="not_equal" value="" variable="FILE_LCZC_FILTERED">
    <tool library="io grid image" module="2" name="Export Grid to KML">
  </condition>
</condition>

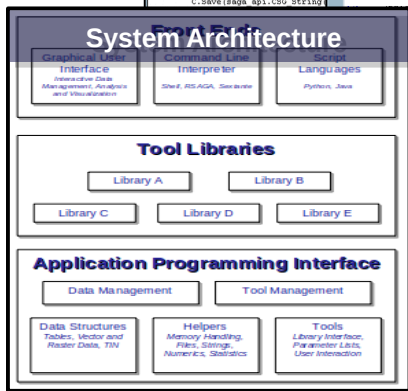
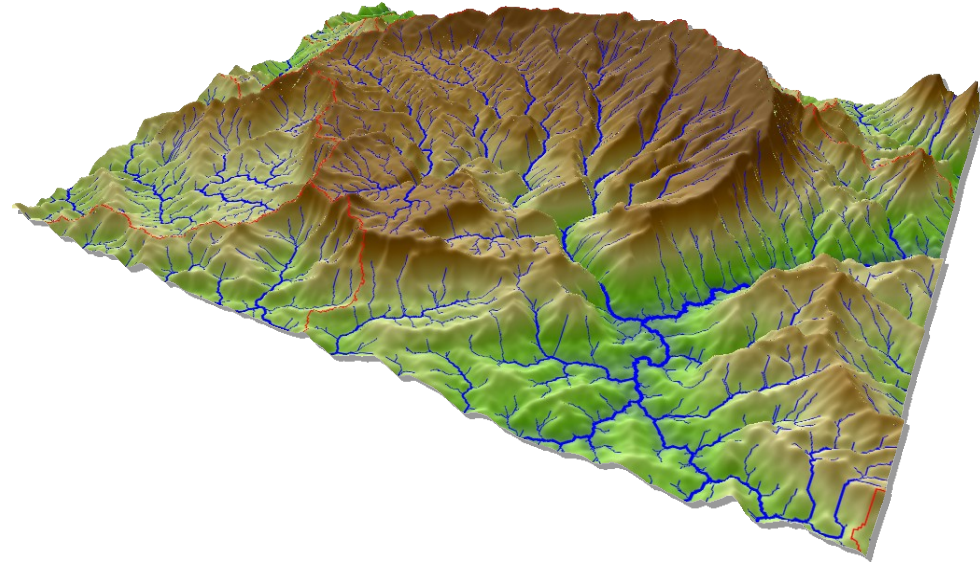
```

SAGA | System for Automated Geoscientific Analyses

The screenshot displays the SAGA GIS desktop environment. On the left, a 'Manager' panel lists various data layers and tools. The central area features a 3D terrain visualization with a blue stream network. To the right, a 'Regression' table shows statistical data for different models. Below the main interface, three windows are open: a 'Command Line' window showing system commands, a 'Python' window with a script for grid processing, and an 'R Script' window with a script for data conversion and analysis.

JAHRE	TREND	RESIDUAL	
1	777.900000	936.878178	-158.978178
2	787.500000	828.413769	-40.913769
3	818.300000	837.846254	-6.646254
4	872.800000	886.270509	-23.470509
5	930.700000	1020.692982	-89.992982
6	152.900000	867.853309	-114.953309
7	691.300000	789.482936	-98.182936
8	644.600000	622.830436	21.769564
9	664.000000	842.279551	-178.279551
10	689.400000	822.849515	-133.849515
11	740.100000	796.955485	-56.855485
12	827.600000	884.141140	-56.541140

Thank you for your attention

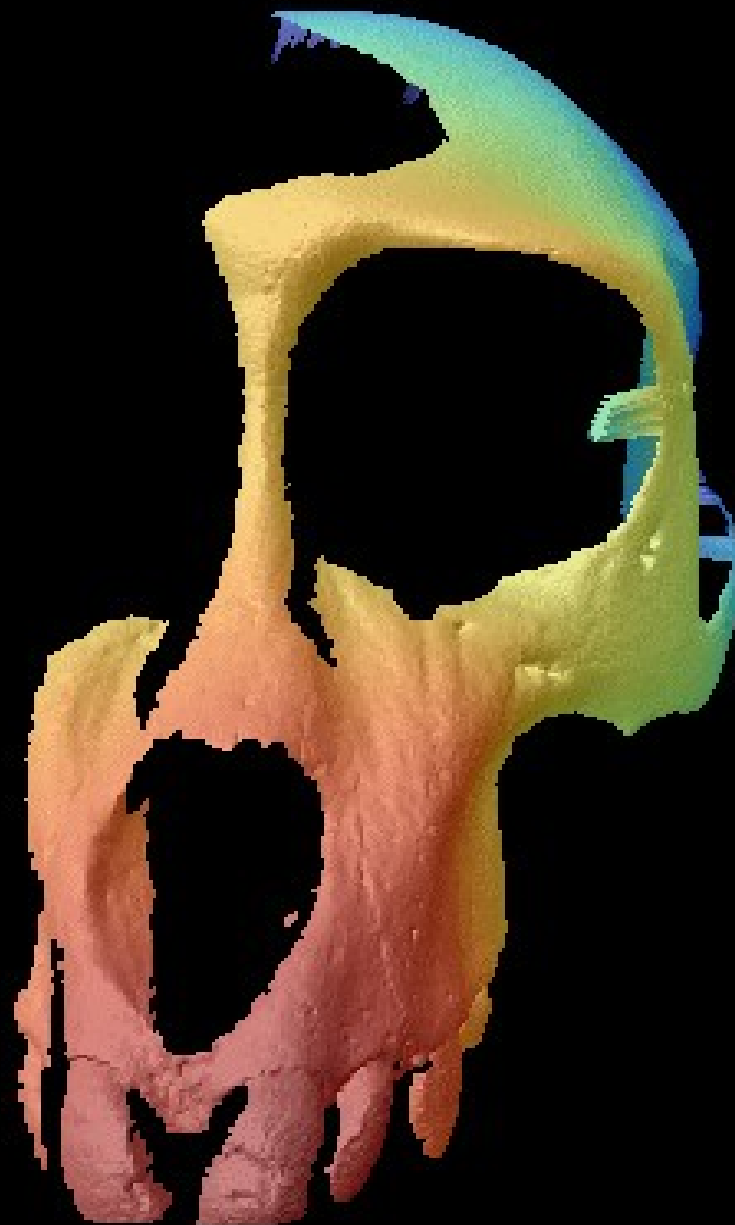
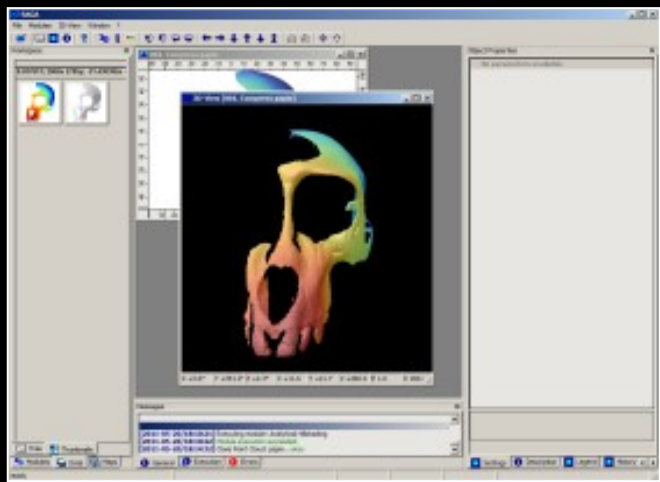


```

R Script
# convert the map to SAGA grid format:
write.asc(grid=meuse.grid[,"meuse_dist"], na.value=-1)
rsaga.esri.to.sgrd(in.grid="meuse_dist.asc", out.sgrd="meuse_dist.sgr")
writeOGR(meuse[["meuse"], "meuse.shp"], "meuse", "ESRI Shapefile")

rsaga.esri.to.sgrd(in.grid="meuse_dist.asc", out.sgrd="meuse_dist.sgr")
writeOGR(meuse[["meuse"], "meuse.shp"], "meuse", "ESRI Shapefile")

option
rsource(lib="pj_proj4", 2, param=list(SOURCE_PROJ=NLL_RL, TARGET_PROJ=UTM))
rsource(lib="io_grid_image", 0, param=list(GRID="meuse_dist"))
    
```



Many thanks
for your attention

www.saga-gis.org

SAGA | Resources

Explore the world of SAGA GIS

<http://www.saga-gis.org>

Basic information

Comprehensive list of references

<http://sourceforge.net/projects/saga-gis>

SourceForge > host for OSS projects

Download software, documents, data

SAGA Wiki

Bug, Feature Tracker

User Forum

User Guide and Manual

Discussion Forums: **User Forum**

Topic	Topic Starter	Replies
area calculation	is-baillv	4
RSAGA on Hardy 64	montanabay	3
geometrical properties of polygons	hydrociost	1
Sampling a table dataseting data from tables.	derri1	2
compiling on Hardy 64	montanabay	3
Ubuntu Hardy 64bit	montanabay	0
Staightening a linestring of points	lthaelein	0
shade grid with...	huberthilbert	2
TIN tools	nideux	1
surf4saga	huberthilbert	2
saga 2.0.3 and fedora 9	lnoagio	0
gridding	jozenunv	2

User Forum

SAGA | Other Information Sources

Marine Data Literacy

marinedataliteracy.org

The **Marine Data Literacy Project** is an attempt to bring together detailed, profusely illustrated instructions for many specific marine data management and analysis procedures, including basic GIS, ocean station data, satellite imagery, and operational data streams. The exercises are grouped according to an informal typology, but users are advised to simply browse through and see what's available. In general, the entire collection is constructed as a sequence of activities to build a "national marine data resource" for a selected area. Since 2010 this location is the area offshore the Ivory Coast. The exercises are currently used by the UNESCO/IOC/IOCE marine data training program, the Japan Foundation/POGO young scientists training program at Bermuda/BIOS, and the Ghent University-Erasmus Mundus masters degree program. SAGA is extensively used in all "marine GIS" lessons and in lessons dealing with grids, rasters and images. Contributing authors are always welcome, and an HTML exercise template is provided for their use. Intensive use of illustrations, and an absolute adherence to the step-by-step approach for all exercises are the only requisites.

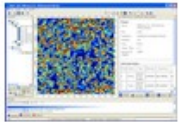
<p>Saga</p>	<p>This is the general-purpose, "workhorse" program we recommend to all data management students, even if they also use other commercial or public domain GIS solutions. One shortcoming is the minimal documentation.</p>	<p>General information:</p> <ul style="list-style-type: none"> • Saga Homepage • Saga Forum on Sourceforge <p>Saga installation files.</p> <ul style="list-style-type: none"> • Saga Files on Sourceforge • The ZIP version (not the unexplained EXE setup version) should be copied to a convenient location and unzipped to C:\ • Run by clicking on saga_gui.exe 	<ul style="list-style-type: none"> • Windows 32 or 64. Create a shortcut to the executable saga_gui.exe to run the program • Saga's Tutorials Collection • Australia-Indonesia Training in Saga for Resource Management with Imagery • Rohan Fisher's Saga Tutorials (in English and Indonesian) • 1.3 Running 32-Bit Saga on a Mac with WINE - Provided by a student • Saga Wiki on Sourceforge for Linux information • "Mac users might like to hear that efforts are going on to make SAGA work on MacOS more smoothly. You find a thread regarding the MacOS port in the SAGA User Forum at http://sourceforge.net/p/saga-gis/discussion/790705/thread/b11de126/ Have a look at http://www.wxwidgets.org for background information about the wxWidgets project." • DOMINOC925 - An amazingly good set of illustrated tutorials for Saga and other geospatial software; possibly hundreds of exercises, but not indexed -- use search function to find Saga examples
--------------------	--	---	---

SAGA | More Sources of Information

dominoc925.blogspot.com

dominoc925
 About Geospatial Applications, Intergraph GeoMedia, FME, Visual Studio, gvSIG, Google Maps, SAGA GIS, Android, QGIS

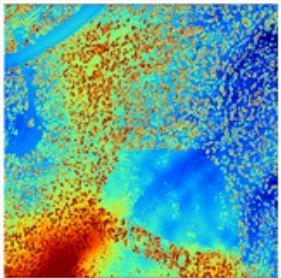
Monday, February 20, 2012
 Simple method to count trees using Saga GIS



It is possible to make a rough estimation of the number of trees in an area from LIDAR derived digital surface (DSM) and digital terrain models (DTM). One method is to use some of the grid analysis modules algorithm in SAGA GIS, such as Gaussian Filter, and Watershed Segmentation. Then simply count the number of segmented table records with height greater than a value.

- The example here counts the trees using the following general steps:
1. Load the DSM and DTM datasets
 2. Calculate the canopy heights
 3. Smooth the canopy heights
 4. Segment the canopy heights
 5. Count the number of segments with canopy heights above a certain value

- Load the source datasets
1. Start SAGA GIS.
 2. Load and display the digital surface model (DSM) grid file, e.g. C:\data\dsdm.asc.



3. Load and display the digital terrain model (DTM) grid file, e.g. C:\data\dtm.asc.

rohanfisher.wordpress.com/ open-source-geo-spatial

rohanfisher
 ICT4D - Appropriate tech for decentralisation



Open Source Geo-spatial
 Capacity building using Open Source Geo-spatial Software

Saga GIS
 SAGA GIS is raster focused spatial analysis software with modules that allow for sophisticated work with satellite imagery and geomorphometric modeling using digital elevation data. I have produced a range of training material whilst (1) delivering capacity building in West Timor and South East Sulawesi for this project [Satellite image display and analysis with a focus on Nusa Tenggara Timur](#) and (2) as part of my work producing burnt area data for [IAF](#).

[Download latest version SAGA GIS here.](#)
 Training Screen Shot Videos:
 Terrain Analysis with SAGA GIS
 Some terrain analysis (morphometric) functions with SAGA GIS using SRTM data for [east Sulawesi](#).

Fire mapping
 Image classif

- MY LINKS
[Monitoring impacts and risks of Mangrove mining in West Timor](#)
[SMS for Health Information in Eastern Indonesia](#)
- FIRE
[Darwin Center for Bushfire Research](#)
[North Australia Fire Information](#)
- CATEGORIES
[Bios](#)
[Fire](#)
[ICT4D](#)
[Indonesia](#)
[Mapping](#)
[Unclassified](#)

dst-iget.in

Search...

- HOME
- ABOUT US
- TUTORIALS
- RESOURCES
- CONTACT US
- FORUM

- UNDERSTANDING GIS >
- REMOTE SENSING >
- DATABASE
- SPATIAL ANALYSIS
- TRENDS IN GIS >
- CUSTOMISATION

- Introduction to SAGA
- Visual Interpretation
- Georeferencing
- Mosaicking Subsetting satellite image
- Introduction to Filters
- Unsupervised Classification
- Supervised Classification
- Terrain Analysis
- Change Detection

Lonar Crater, Maharashtra

Understanding GIS Remote Sensing Database Spatial Analysis Trends in GIS Customisation

www.cdu.edu.au/itl/AII-RS/

Satellite image display and analysis with a focus on Nusa Tenggara Timur.
 Penampilan dan analisa citra satelit dengan focus terhadap Nusa Tenggara Timur

- Workshop
- Tutorial
- Links - contacts

The use of satellite data for mapping and monitoring is an important tool assisting effective and timely natural resource management. Furthermore the application of local knowledge in the interpretation of satellite data is often key to understanding the mapped landscape, observed changes and for deriving useful management outcomes. Currently, most satellite based assessments of natural resources in Eastern Indonesia are conducted by non-locals. However, with evolving technologies and changing research methods, new opportunities are arising for the wider use of satellite technology. This tutorial has been created as part of ongoing collaborative engagement between Charles Darwin University (Darwin, Australia) and Nusa Cendana University (Kupang, Indonesia) and has been funded by the Australia Indonesia Institute.



Pemetaan dan monitoring dengan data citra satelit adalah alat-alat yang penting untuk pengelolaan sumber daya alam yang efektif dan tepat waktu. Selanjutnya pemanfaatan kebijakan lokal dalam penafsiran pemetaan data satelit, seringkali menjadi kunci untuk mendalami pengertian tentang darat, perubahan-perubahan yang dilihat dan mendapat kegiatan pengelolaan yang tepat.

Tutorial ini adalah sebagian dari kolaborasi lebih luas, antara Charles Darwin University (Darwin, Australia) dan Universitas Nusa Cendana (Kupang, Indonesia) yang memperkenalkan ketrampilan dasar dalam pemetaan dan monitoring dengan data citra satelit. Dana dari Australia Institute.

