International Training

Application of Satellite Remote Sensing to Support Water Resources Management in Latin America and the Caribbean

Foz de Iguazú, Brazil, 13 - 20 July 2016

Tutorial
SPIRITS version 1.3.0 – December 2015

Roel Van Hoolst, Carolien Toté, Herman Eerens, Dominique Haesen

Table of Contents

Introduction

While image processing software packages in general focus on the processing and analysis of single or multi-temporal images, the concept of SPIRITS ('Software for the Processing and Interpretation of Remotely sensed Image Time Series') is to provide automated and advanced time series processing for large series of images with a temporal resolution of one day, one dekad (10-days), one month or one year. SPIRITS was developed by VITO for JRC-MARS.

The objective of this tutorial is to introduce the participants of the training workshop to some seasonal analysis protocols for the Earth Observation based monitoring of cropland. We give an **example over Ethiopia** which can be adapted later to any site.

In this tutorial specific actions dealing with the software are separated from the accompanying text:

- ✓ Actions in an exercise are preceded by a ✓.
- **?** Throughout most exercises, questions will appear. These questions provide opportunity for reflection and self-assessment on the concepts just presented or operations just performed.
- **!** An exclamation mark is used for remarks.

<*Button*> are menus, buttons or drop-down boxes to be presses or selected.

'Directory' is the notation for directories or specific files. (e.g. '*D:\TUTORIAL\DATA*')

"Text" is the notation for text to be entered.

Before starting the course, users should copy the training data set (the entire 'TUTORIAL' directory) on their hard disk.

Apart from this Tutorial, the SPIRITS Manual will serve as a reference for all the operations. The Manual can be opened after the installation of SPIRITS from the <*About*> menu, or by clicking <*Help*> in any of the SPIRITS tools. The SPIRITS Manual is also available in the '*C:\SPIRITS*\' directory.

We welcome your feedback, comments and suggestions for improving the SPIRITS Tutorial (contact: carolien.tote@vito.be, roel.vanhoolst@vito.be or herman.eerens@vito.be).

NB: More tutorials are available on the SPIRITS website: http://spirits.jrc.ec.europa.eu/.

Part 1 The SPIRITS Environment

This first part of the tutorial covers the following issues:

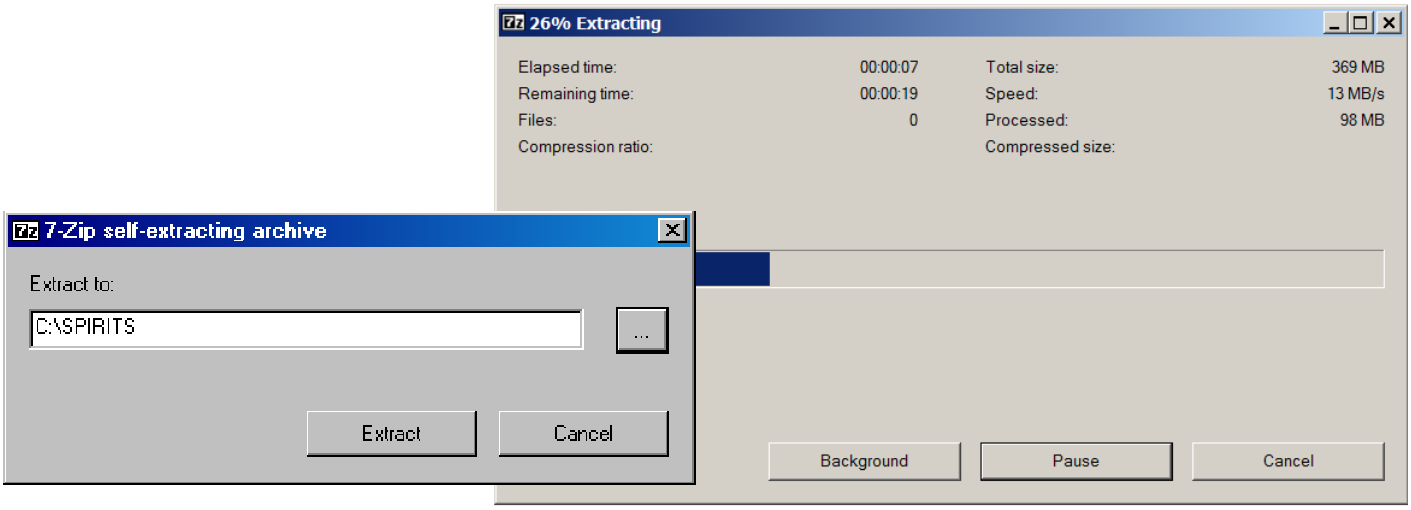
- Installation of SPIRITS

- Getting started
- Tutorial test data over Ethiopia
- Declaring this Ethiopia data set as a SPIRITS “project”

Installation of SPIRITS

The full software, including the Manual, is contained in the self-extracting program ***SpiritsExtract.exe***. This file can be copied to any location, and when it is run the “7-Zip Self-Extractor” is opened and installs the software in the specified folder.

✓ Install the SPIRITS software on your computer: double click on *SpiritsExtract.exe*.
✓ For this exercise: Install it in ‘***C:\SPIRITS***’.
✓ Click on <*Extract*>. After the files are unzipped, close the 7-Zip Self-Extractor.

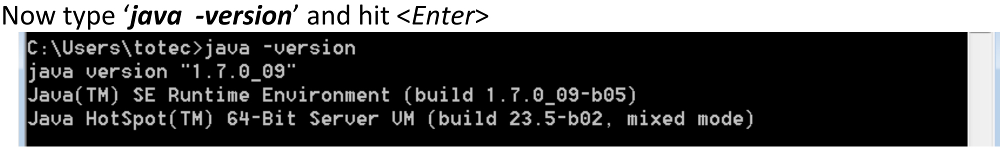


WARNING: In order for SPIRITS to work properly, JAVA version 1.6 (or higher) needs to be installed on your computer.

Perform the following steps only in case of SPIRITS is not running properly:

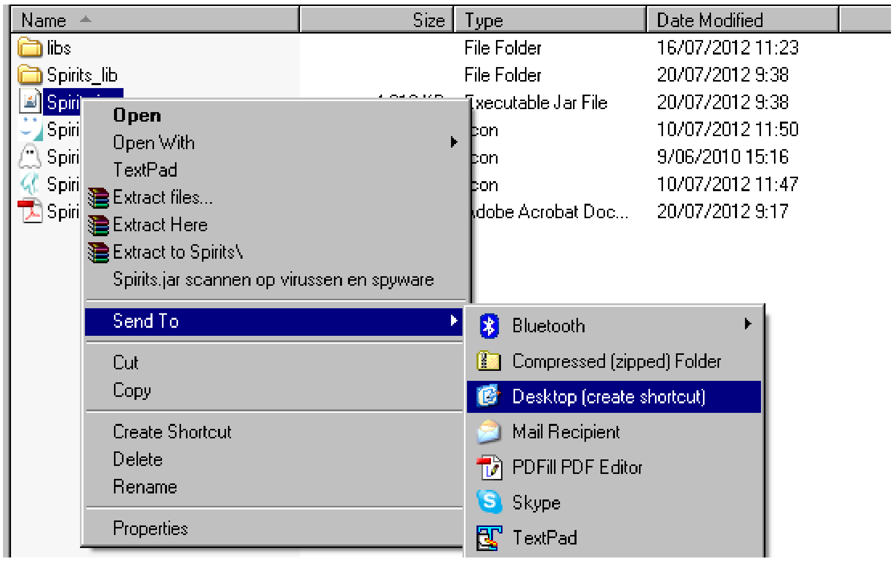
✓ Open a DOS command line (click ‘Start’, ‘Run’ and run ‘cmd’).

✓ Now type ‘***java -version***’ and hit <*Enter*>

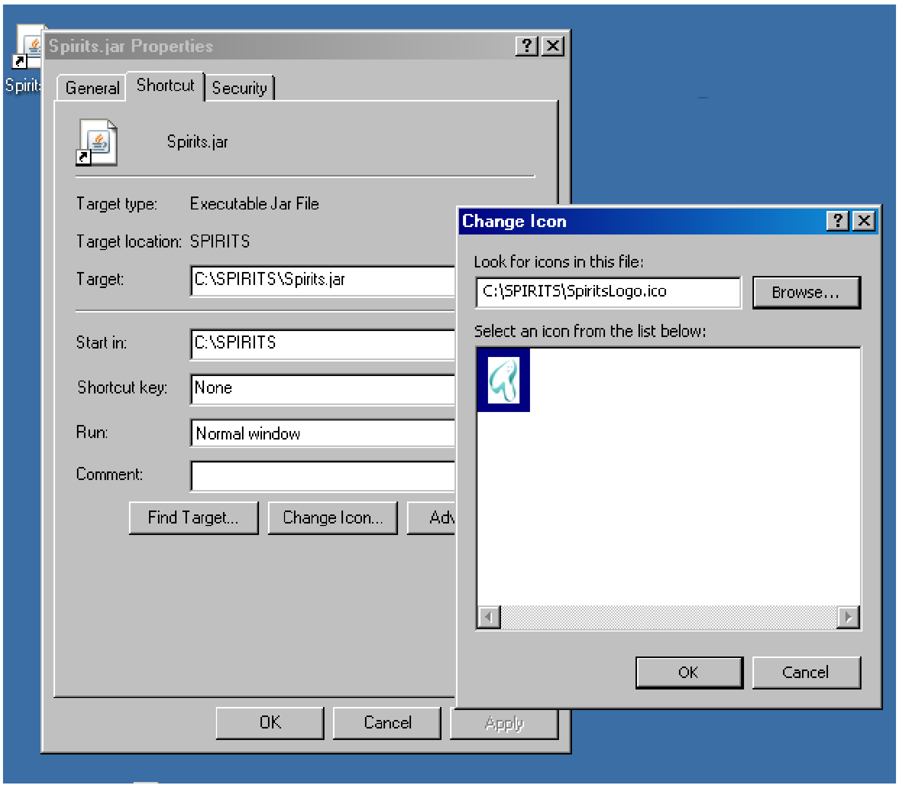


✓ If there is no java version installed (or the version is too old), download an installer file from http://www.java.com/en/download/installed.jsp.

✓ Browse to the SPIRITS-folder (C:\SPIRITS\) and create a shortcut on the desktop for the ***Spirits.jar*** file (the core program).



✓ Go to the desktop, and open the properties of the Spirits-shortcut. Click on <*Change Icon*> and select one of the *.ico files in the installation folder. Click <*Open*> and twice <*OK*>.



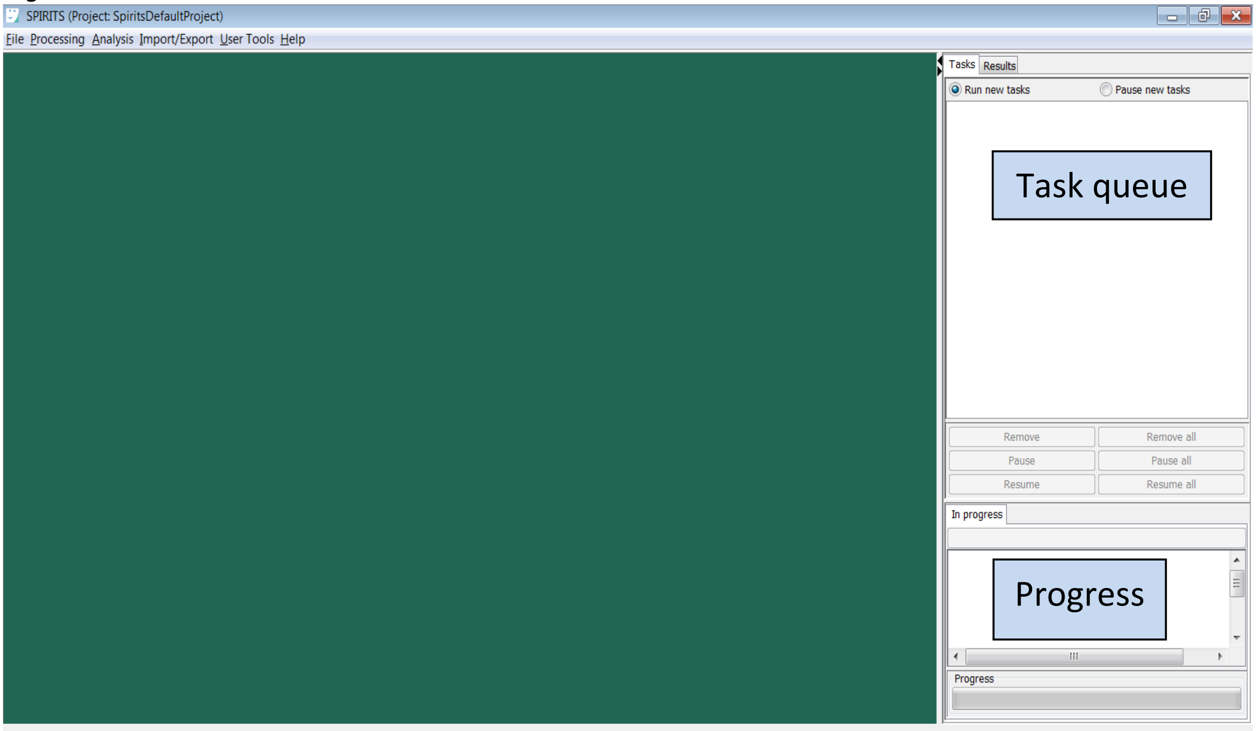
For more information on the extracted files and the Spirits directory structure, see the Spirits Manual.

Getting started

- ✓ To start SPIRITS, double-click on the SPIRITS icon on your desktop, or double-click on *Spirits.jar* in the installation folder.

- ✓ Now explore the SPIRITS main window.

The SPIRITS graphical user interface consists of a *Title bar*, a *Menu bar*, a *Main Pane*, a *Task Pane* and a *Progress Pane*.



The Title bar on top shows the current “**project**” (for now this is the ‘***SpiritsDefaultProject***’).

The Menu bar contains the SPIRITS procedures. Some of them will be explained in the following exercises.

In the panels on the right SPIRITS will display the running tasks, their progress and results. This way, it will be easy to follow up running routines. By clicking on the tiny black arrows in the top left corner of the tasks pane, it can be minimized or maximized.

- ✓ Open the <*Help*> menu and click on <*About Spirits*>. Notice the version number and release date of SPIRITS.
- ✓ In the <*Help*> menu you also find the Manual of SPIRITS.

The TUTORIAL Data over Ethiopia

Data contents

For this exercise, two image data sets were prepared, both covering Ethiopia. Their main characteristics are summarized in the table below.

CAT	ITEM	SPOT-VGT	ECMWF
SPECTRAL	Variable	**NDVI [-]**	**Daily mean rainfall [mm/day]**
	Data type	1=unsigned byte	2=signed short integer
	Significant range V_{lo} - V_{hi}	0 – 250	0 – 32767
	Scale	NDVI [] = -0.08 +0.004 * V	RFE [mm/day] = 0.01 * V
	Flags	251=missing, 252=cloud, 253=snow, 254=sea, 255=back	V < 0: Missing values
SPATIAL	Map system	Geographic Lon/lat	Geographic Lon/lat
	Geodetical datum	WGS84	WGS84
	Nr. of Colums	1681	61
	Nr. of Records	1289	47
	Resolution	1°/112 (≈ 1 km)	1°/4 (≈ 25 km)
	Range in X = LON	33.0 → 48°	33.0 → 48°
	Range in Y = LAT	3.402 → 14.902°	3.5 → 15°
TEMPO	Periodicity	10 days (dekadal)	10 days (dekadal)
	Years covered	1999 – 2013 (15 years)	1999 – 2013 (15 years)
	Nr. of dekads / IMGs	540	540
	IMG names	vt***YYTT***i.img/hdr	wt***YYTT***y.img/hdr

Remarks: The LON/LAT ranges indicate the co-ordinates of the centre of the image corners.
All this information can also be found in the HDR-files.
The flagging system (251-255) of the NDVI images is labelled as "**UNIflags**".

The SPIRITS header file for the last NDVI image in the VGT series (vt1336i.hdr):

```
ENVI
description = {SPOT-VGT, type=S10, date=20131221 (UNI-flags)}
samples = 1681
lines = 1289
bands = 1
header offset = 0
file type = ENVI Standard
data type = 1
interleave = bsq
map info = {Geographic Lat/Lon, 1, 1, 32.9955357, 14.90625, 0.00892857143, 0.00892857143}
values = {NDVI-toc, -, 0, 250, 0, 248, -0.08, 0.004}
flags = {251=missing, 252=cloud, 253=snow, 254=sea, 255=back, 255=back}
date = 20131221
days = 10
sensor type = SPOT-VEGETATION
```

The SPIRITS header file for the last Rainfall image in the ECMWF series (wt1336y.hdr):

```
ENVI
description = {ECMWF and JRC-MARSOP: Mean daily precipitation}
samples = 61
lines = 47
bands = 1
header offset = 0
file type = ENVI Standard
data type = 2
interleave = bsq
byte order = 0
map info = {Geographic Lat/Lon, 1, 1, 32.875, 15.125, 0.25, 0.25}
values = {daily precipitation, mm, 0, 32767, 0, 1057, 0, 0.01}
flags = {-32000=back, -32001=missing, -32768=compo}
date = 20131221
days = 10
sensor type = Meteo
```

Folder organisation

Below it will be assumed that the tutorial data are stored in ‘***D:\TUTORIAL\ETH***’. Under this “Project folder”, we already created the following sub-folders:

<table>
<tr><th>DIR1</th><th>DIR2</th><th>DIR3</th><th>CONTENTS</th></tr>
<tr><td rowspan="3">IMG
QLK
RUM</td><td rowspan="3">NDVI
ECM</td><td>ACT</td><td>Actual data</td></tr>
<tr><td>LTS</td><td>Long-Term Statistics (or “Historical Year”)</td></tr>
<tr><td>DIF</td><td>Anomalies = Difference of actual state w.r.t. LTS</td></tr>
<tr><td rowspan="3">REF</td><td>VEC</td><td></td><td>GAULg.shp/dbf with g=0/1/2: FAO administrative regions</td></tr>
<tr><td>REG</td><td></td><td>Raster versions of these GAUL maps</td></tr>
<tr><td>MSK</td><td></td><td>Land Use information from FAO’s “GLC_Share” V1.0</td></tr>
<tr><td>SPI</td><td></td><td></td><td>All SPIRITS-specific files: tasks (*.tnt), scenario’s (*.sns), etc.</td></tr>
</table>

Two examples of sub-folders:

- ‘***.\IMG\RFE\ACT***’ contains the actual images (per dekad) with rainfall from ECMWF.
- ‘***.\QLK\NDVI\DIF***’ will contain the QuickLooks (*.png) of the NDVI-anomalies.

File names

For the treatment of time series, SPIRITS requests that all file names follow the generic pattern:

[P]date[S].ext

The prefix ***P*** and suffix ***S*** may be empty, and the prefix P may also include a (complete or partial) data path, but the ***date*** must be specified according to one of the twelve formats listed in the table below. The extension (***.ext***) can be “img/hdr” for the images, “png” for the QuickLooks, or “rum” for the RUM-files.

<table>
<tr><th>N</th><th>DATE FORMAT</th><th>MINIMAL PERIOD</th><th>EXPLANATION of SYMBOLS</th></tr>
<tr><td>1</td><td>YYYYMMDD</td><td rowspan="4">Day</td><td rowspan="12">YYYY = Year [1950 … 2049]
YY = Year [50=1950 … 49=2049]
MM = Month in year [01=Jan. … 12=Dec.]
m = Month in year [A=Jan. … L=Dec.]
TT = Dekad in year [01 … 36]
DD = Day in month [01 … 31]</td></tr>
<tr><td>2</td><td>YYMMDD</td></tr>
<tr><td>3</td><td>YYYYmDD</td></tr>
<tr><td>4</td><td>YYmDD</td></tr>
<tr><td>5</td><td>YYYYTT</td><td rowspan="2">Dekad</td></tr>
<tr><td>6</td><td>YYTT</td></tr>
<tr><td>7</td><td>YYYYMM</td><td rowspan="4">Month</td></tr>
<tr><td>8</td><td>YYMM</td></tr>
<tr><td>9</td><td>YYYYm</td></tr>
<tr><td>10</td><td>YYm</td></tr>
<tr><td>11</td><td>YYYY</td><td rowspan="2">Year</td></tr>
<tr><td>12</td><td>YY</td></tr>
</table>

For this exercise, which only works with dekadal data, we use the following convention:

- Prefix P="**vt**" for the NDVI data. The "v" stands for SPOT-VGT, "t" for ten-daily. For the ECMWF-data, P="**wt**" ("w"=weather, "t"=ten-daily).
- The date format is always "**YYTT**" (N=6 in the table), with "YY"=year and "TT"=dekad_in_year.
- Suffix S is used to indicate the concerned variable: S="**i**" for the original NDVI, "**k**" for the smoothed NDVI, "**y**" for rainfall. For the anomalies a second character (mostly a number) is added to indicate the type of the used difference operator.

Some examples, all under our project folder '***D:\TUTORIAL\ETH***':

- '**.\IMG\NDVI\ACT\vt0004i.img/hdr**': Original NDVI IMG of dekad 4 of year 2000.
- '**.\QLK\ECM\DIF\wt0004y1.png**': QuickLook of the rainfall anomaly (difference type 1) for the same dekad.

Remarks:

- The "dekadal" system works as follows: the first two dekads of each month always count 10 days (01-10, 11-20), while the third one has a variable number of days (21→end_of_month).
- As to the main folders (IMG, QLK, RUM), the ETH project folder is mostly empty. The only folders with data are the ones with the original actual images:
  - '**.\IMG\NDVI\ACT\vt*YYTT*i.img/hdr**': VGT-NDVI
  - '**.\IMG\ECM\ACT\wt*YYTT*y.img/hdr**': Rainfall from ECMWF

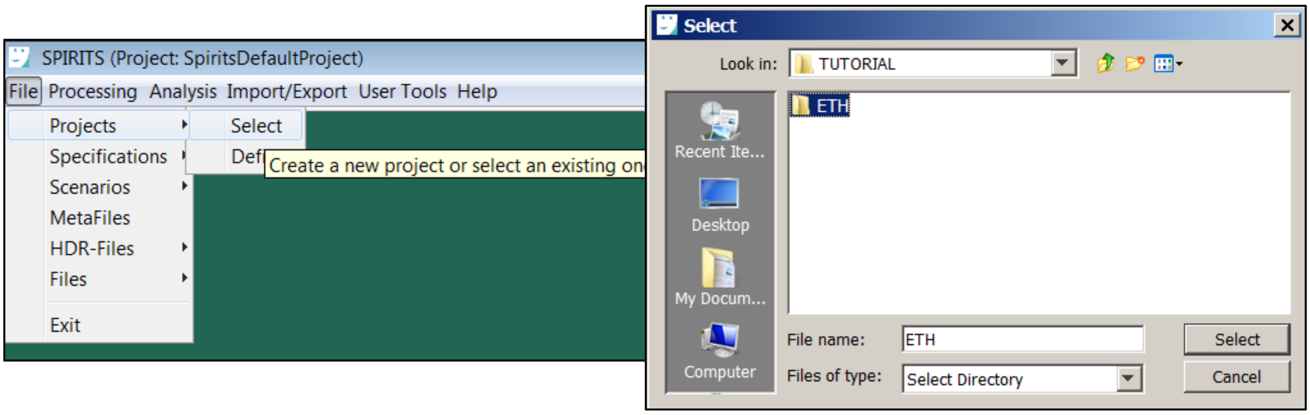
  The objective of the below exercises is to fill the other folders with appropriate derivatives.

Declaring Ethiopia as a SPIRITS Project

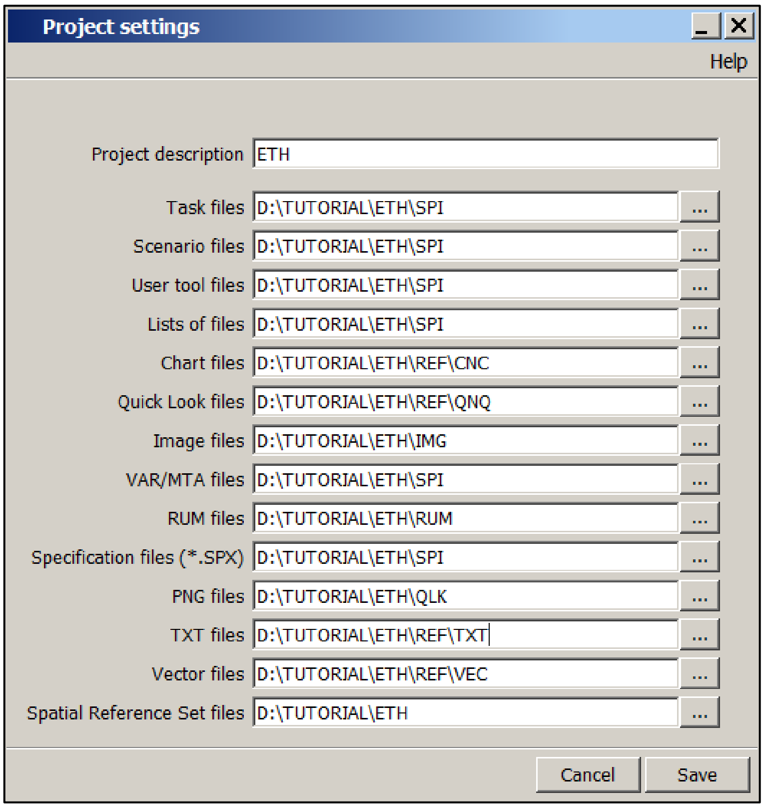
SPIRITS always works on a specific "**project**". A project corresponds with a specific disk folder and all the data within this folder. When the software is run for the first time (or in any case where no real project is known), SPIRITS automatically creates an "empty" project in the installation folder, called '***SpiritsDefaultProject***'. Thus in our case: '***c:\SPIRITS\SpiritsDefaultProject***' (see before on page 6). But it's not a good idea (and most inpractical) to continue working with this default folder, or to put real data in it.

In this exercise we will define a new SPIRITS project for the Ethiopia data.

- ✓ Go to <*File*> <*Projects*> <*Select*> and specify the project directory: '***D:\TUTORIAL\ETH***'. Using the ... button, you can browse through your directory structure. Press <*Select*> when finished.



- ✓ Now go to <*File*> <*Projects*> <*Define*> and adapt the references to the different sub-folders in agreement with the folder organisation of our Ethiopia data set (see the figure below).



- ✓ ***Save the project via the <Save> button!***

What are the consequences?

- Under the selected “project folder” (***D:\TUTORIAL\ETH***), SPIRITS has automatically created a new sub-folder, called ‘***SpiritsProjectData***’, with a.o. the following:
    - File “**SPIRITS.pnp**” contains all the settings of the project (also the ones just defined here).
    - An empty skeleton is created for the database which might be used/filled in later stages.
- It took some time to fill in the above settings via <*File*> <*Projects*> <*Define*>. But now SPIRITS more or less knows the general folder stucture of the project. Later menus will very often ask for the paths of the input and output files, plus a wide range of auxiliary files. Their selection is now much easier.
- You can create as many projects as needed, but at any instance only one can be active.

Part 2 Some Practical Tips

1. Write down your processing schema

Before pressing any button, make clear to yourself which end-results you want to achieve. Write down a processing scheme: the different steps you need to take to achieve this end-result. Most often, you will need to go through a chain of processes, where each of those actions will generate a number of intermediate files. The output files of one action typically are the inputs for the next action. You can work out a processing scheme on paper to show the various steps: Step 1: Import the data, Step 2: Generate Maps to check the imported data, Step 3: Calculate the long term average,... etc.

2. Device a transparent directory structure and file naming convention

The one of this Ethiopia exercise is just a simple example. Other projects might be more complex, for instance because they involve data at different frequencies (daily S1, dekadal S10, monthly S30). But in any case, very long path/file names should be avoided.

3. First test the procedure on a single file

Before applying a certain procedure (or SPIRITS “tool”) to a time series, try it out on a single date. After checking the content and the name of the generated file, you can run a scenario for a series of input files.

4. Double-check all the parameters before hitting the <Execute> button

One press on an <Execute> button can have serious consequences if the parameters are not correct. One can overwrite important files (without warning!), or start a wrong procedure which takes long to finish.

5. Regularly check the available disk space

Some processes can generate huge amounts of results, especially for large ROIs (Regions of Interest) and long time series. Regularly delete intermediate or unnecessary files.

6. Check for errors

After each step, it is a good practice to check for errors in the <Tasks> and <Results> pane. All tasks which contained errors will be marked with a red bullet, tasks which executed correctly have a black bullet. By clicking on the error-tasks, one can examine the log (including error description) in the <Progress> pane.

7. Check the results with your preferred GIS tool.

Even when SPIRITS finsishes a process without error messages, the actual contents of the generated files can be erroneous. Therefore, after each operation, check the contents of the results by using the <Map> functionality of SPIRITS or by opening the file with your preferred GIS or image processing software.

8. Check the HDR-files.

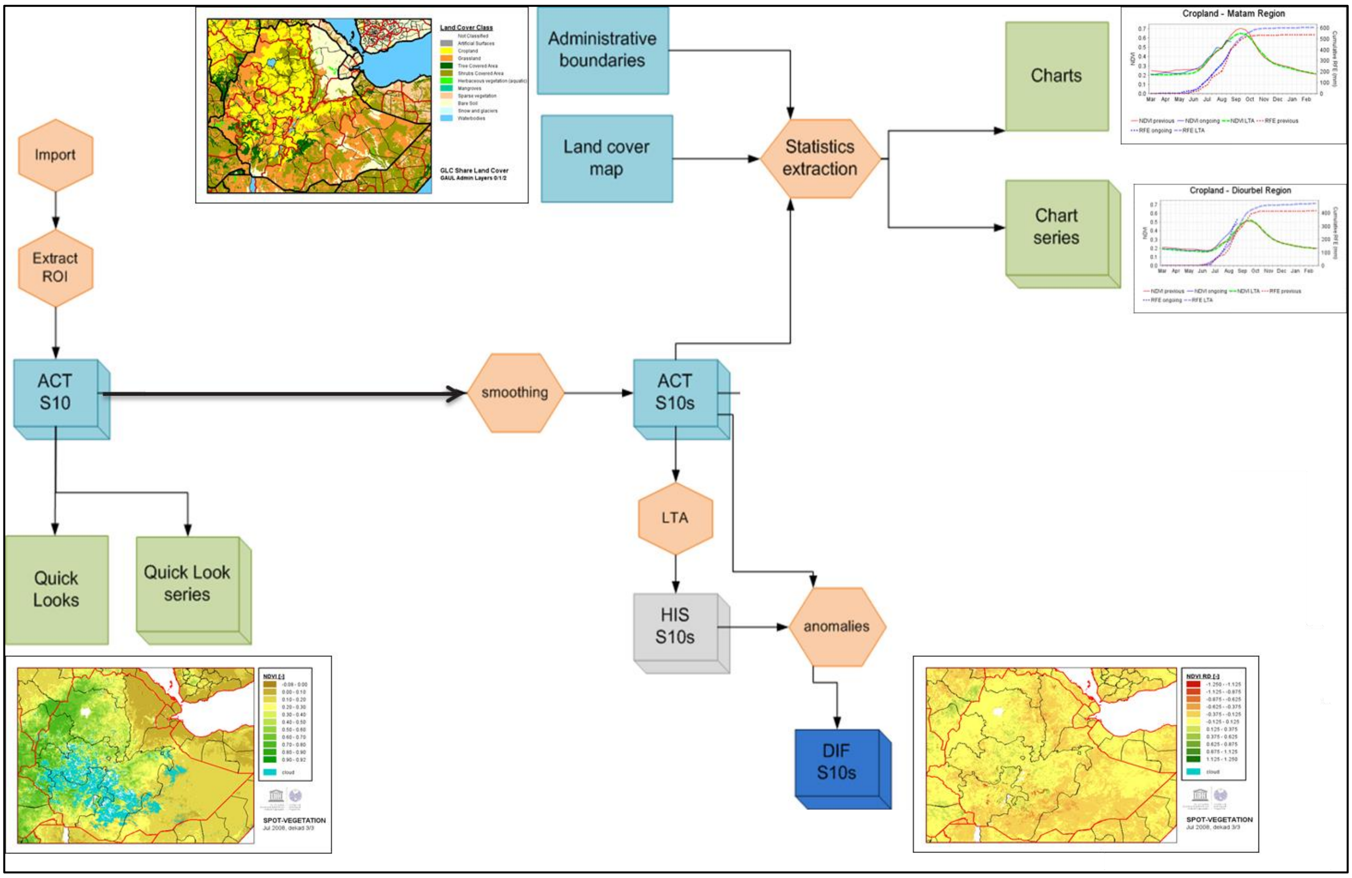
When procedures don’t work or generate wrong results, it is often more instructive to check the contents of the header files, associated to each image.

9. Think about the WHAT and the WHY of your actions.

When running the exercises in this tutorial, do not follow the instructions blindly but try to realize why the steps are performed in the way they are presented. And invent alternative ways for this and other situations.

Typical SPIRITS workflow for Vegetation Monitoring

Here for NDVI from SPOT-VGT over Somalia (1999-2013)



Part 3 Map generation

A QuickLook (QLK) or Map is a graphical file (*.png) which provides a general overview of the image from which it is derived, with the following differences:

- Whereas the original IMG may cover many MB or even GB, the QLK has a moderate size of less than 1 MB, so it can be easily included in documents, presentations, bulletins, websites, ...
- While the IMG presents the "tough" data in all details (to be inspected and analysed for instance with ENVI), the QLK is really a "map" foreseen with all accessories such as: colour scheme, legend, vectors in overlay, logos, etc.

The creation of such QLKs is handled in two steps:

- First, the general layout of the QLK of a specific image must be defined, including colours, legend, vectors, etc. This layout can be saved as a "template" file, with the extension *.QnQ. This is explained in the first exercise (3-1).
- Afterwards this QnQ template can be used to generate QLKs for all similar images in a time series. This is the subject of the second exercise (3-2).

Exercise 3-1 Creation of a Map Template (*.QnQ)

In this exercise, we will make a QLK-map of the NDVI image:

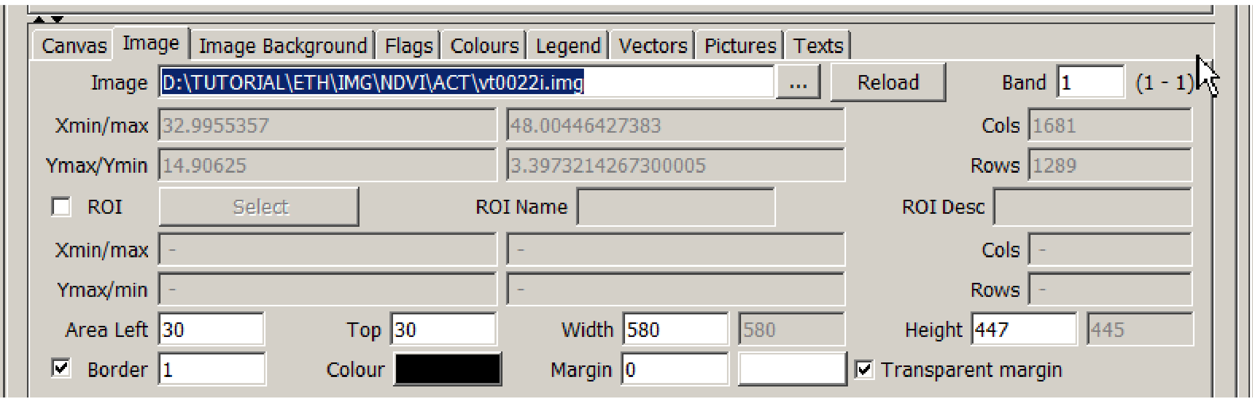
D:\TUTORIAL\ETH\IMG\NDVI\ACT\vt0022i

Or more in general: we will try to make the "best-of-our-needs" QnQ-template to map all the images of this type, regardless their date/dekad.

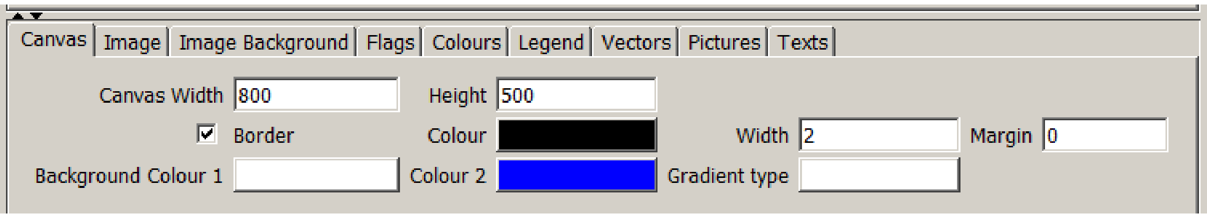
- ✓ Go to ***<Analysis> <Maps> <Create template>***.
- ✓ In the ***<Image>*** tab, click on ... and load the above NDVI image. Initially it is displayed in grey.
- ✓ At the top of this window you can either visualize the Map, or the HDR file of this loaded image. At the bottom, all aspects of the map can be changed.
- ✓ Go to the <*View HDR*> tab on the top. At some points, it can be very useful to check the metadata of the visualized image. But then return to the <*Quick Look*> tab.

In the ***<Image>*** tab, change its size and position.

- ✓ Change the image position and size so it is placed in the upper left corner (e.g. *Left* = "*30*", *Top* = "*30*", *Width* = "*580*" pixels). The value of the *Height* field (447) will automatically be adapted to maintain the same ratio between Height and Width.
- ✓ Change the *Border Width* to "*1*" and *Border Margin* to "*0*".

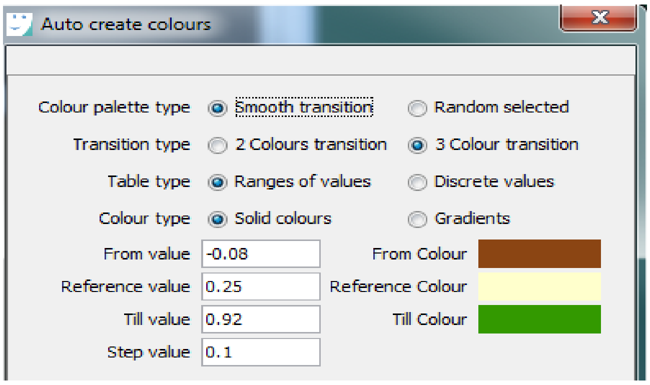


- ✓ Go to the **<Canvas>** tab. Make the canvas larger (e.g. *Canvas Width* = “800” pixels, *Canvas Height* = “500” pixels). This creates space for other elements (legend, logo,...).

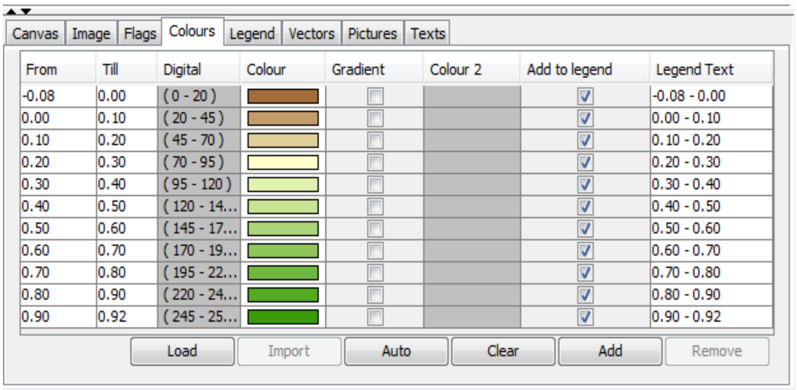


Now change the colour scaling, background colours and legend of the image.

- ✓ Go to the **<Colours>** tab and click <*Auto*>. Notice that the values relate to the physical values in the image. Select <3 Colour transition>. Define the F*rom value* (“*-0.08*”), Reference value (“0.25”), *Till value* (“*0.92*”) and *Step value* (“*0.1*”).

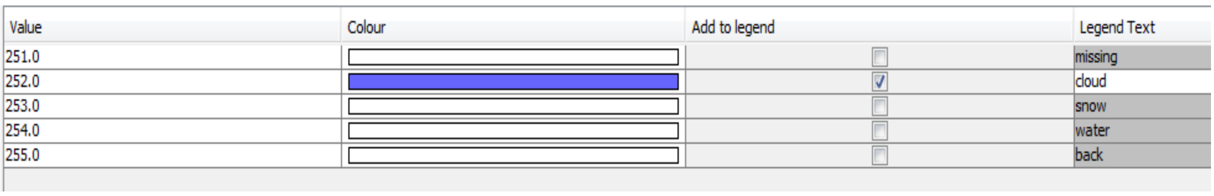


- ✓ Define a minimum, reference and maximum colour for the colour transition. Since you are displaying a vegetation index, scale the values for the *From Colour, Reference colour* and *Till Colour* respectively between brown, light yellow and dark green.
- ✓ Click <*Apply*> and <*Close*>. The image is now coloured and the “legend” looks like this:



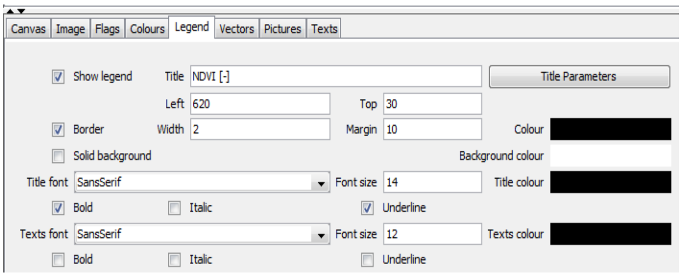
From	Till	Digital	Colour	Gradient	Colour 2	Add to legend	Legend Text
-0.08	0.00	(0 - 20)		☐		☑	-0.08 - 0.00
0.00	0.10	(20 - 45)		☐		☑	0.00 - 0.10
0.10	0.20	(45 - 70)		☐		☑	0.10 - 0.20
0.20	0.30	(70 - 95)		☐		☑	0.20 - 0.30
0.30	0.40	(95 - 120)		☐		☑	0.30 - 0.40
0.40	0.50	(120 - 14...		☐		☑	0.40 - 0.50
0.50	0.60	(145 - 17...		☐		☑	0.50 - 0.60
0.60	0.70	(170 - 19...		☐		☑	0.60 - 0.70
0.70	0.80	(195 - 22...		☐		☑	0.70 - 0.80
0.80	0.90	(220 - 24...		☐		☑	0.80 - 0.90
0.90	0.92	(245 - 25...		☐		☑	0.90 - 0.92

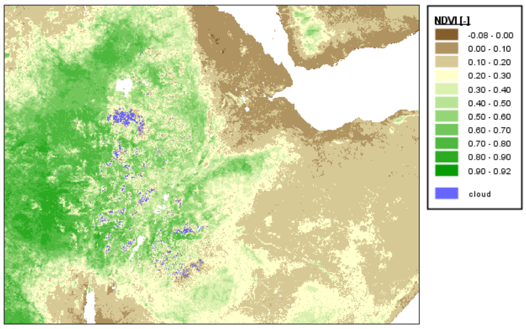
- ✓ Go to the **<Flags>** tab and click <*Import*>. Note that Flags are missing data values, and that <*Import*> retrieves this information from the image header (see <*View HDR*> tab, ‘Flags’ field).
- ✓ Disable ‘Add to Legend’ for all entries except ‘cloud’ (code 252) pixels.
- ✓ Change the colours (e.g. blue-gray for ‘*clouds*’ and white for the others).



Value	Colour	Add to legend	Legend Text
251.0		☐	missing
252.0		☑	cloud
253.0		☐	snow
254.0		☐	water
255.0		☐	back

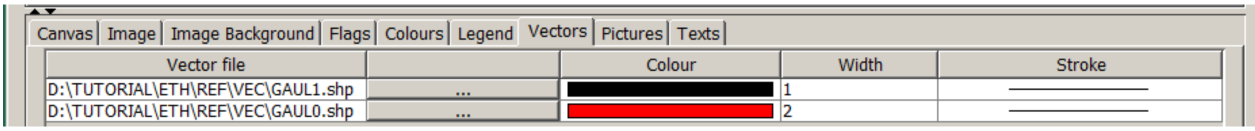
- ✓ Go to the ***<Legend>*** tab. Check the 'Show legend' tick box, and change the legend title, position, border and font size.



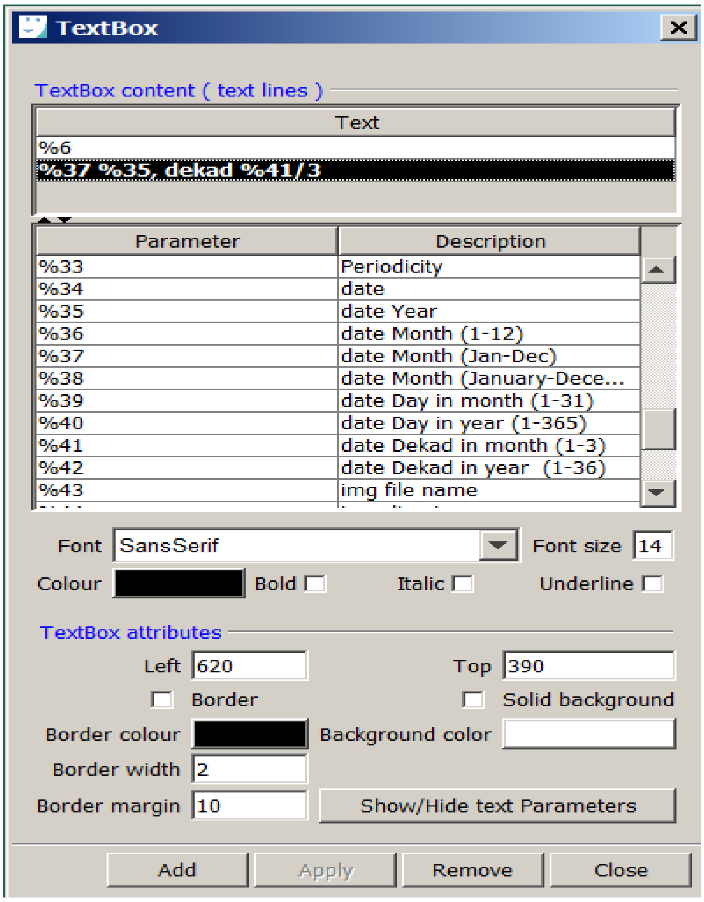


Now we will add a vector layer, a logo and a title.

- ✓ Go to the ***<Vectors>*** tab and click <*Add*> and .
- ✓ Add the Shapefiles[1] with the Ethiopian administrative boundaries (GAUL1 and GAUL0 level boundaries) in the '*REF\VEC*\' directory. Adjust Colour and Width if necessary.



- ✓ Go to the ***<Pictures>*** tab and click <*Add*> and .
- ✓ Add a logo to the map. For example, select the Unesco (in the "\REF\LOG" directory). Change the size (e.g. width "*108*" pixels and height "*65*" pixels) and position (e.g. "*620*" left, "*330*" top).



[1] The vector file should have the same projection system as the image. SPIRITS is not a GIS and cannot perform 'on-the-fly' reprojections.

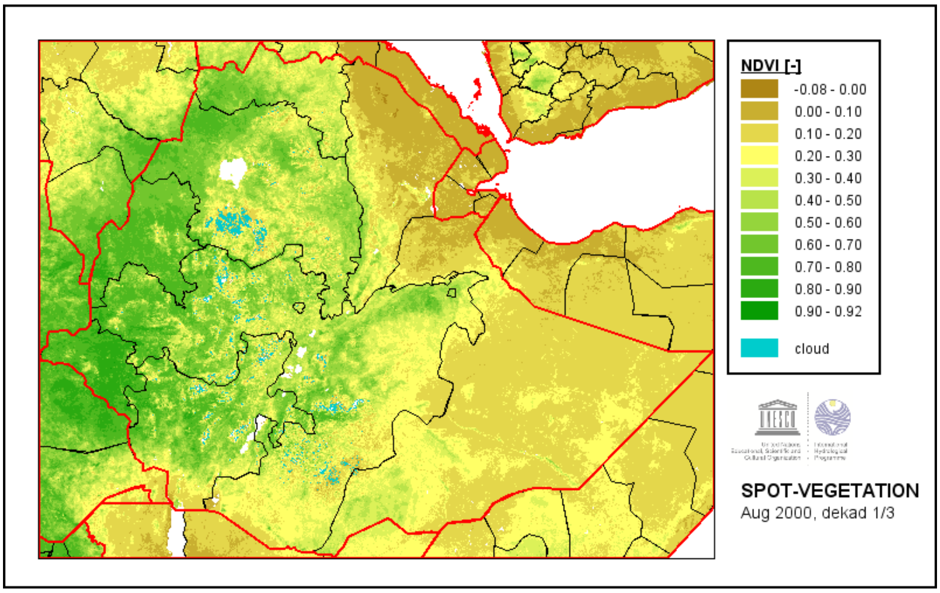
- ✓ Go to the ***<Texts>*** tab and click <*Add*> to add a new textbox.
- ✓ In the TextBox window (see figure above), again click <*Add*> to add a first line. In this line, we want to display the sensor. To retrieve this information from the image header: click on <***Show/Hide text Parameters***>.
- ✓ Click on the empty text line and add content (e.g. "*%6*") to the text box, change the font size to "*16*", click <*Apply*> and check what happens.
- ✓ Click <*Add*> to add a second line in the text box showing the image date. Also this information can be read from the hdr-file. Enter '*%37 %35, dekad %41/3*', click <*Apply*> and again see what happened.
- ✓ Remove the border of the textbox. Note that you can change the text size and display for each line separately. Put "*14*" as the font size for the second line. Click <*Apply*> and <*Close*>.
- ✓ Change the positioning of the Map title (e.g. "*620*" left and "*390*" top).

NB: To edit an existing text field in the TextBox window: press the F2 key!

The advantage of using these "text parameters" is that they are automatically updated when loading another image. To verify this: In the <*Image*> tab, load another NDVI image of a different dekad. As you will see, the date in the textbox is adapted accordingly.

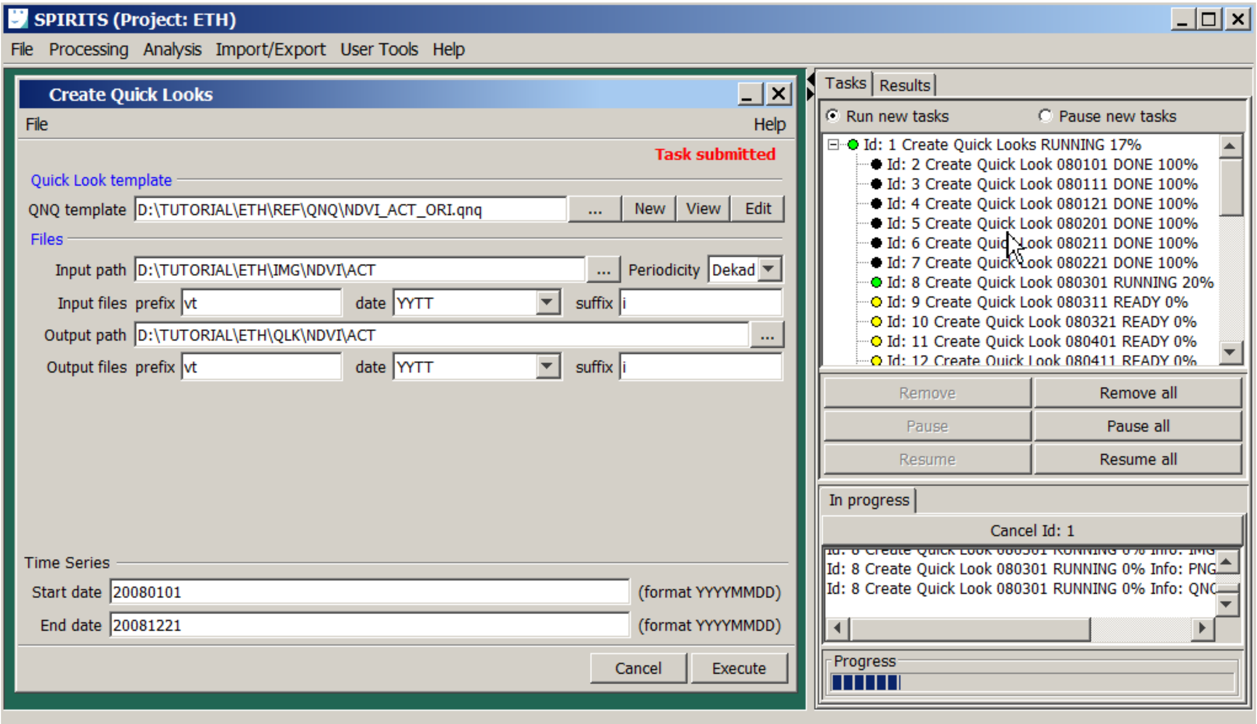
- ✓ At this stage you can optionally export the Map to a PNG file (try <*File*>, <*Export PNG*>), which can be used in reports, presentations, on websites, etc.
- ✓ Certainly save the Map "Template" by clicking <*File*> <*Save As*>. Save it "***NDVI_ACT_ORI.qnq***" in the 'REF\QNQ' directory. We will need it in the next exercise.

Close the Quick Look generator screen. Below the final result.



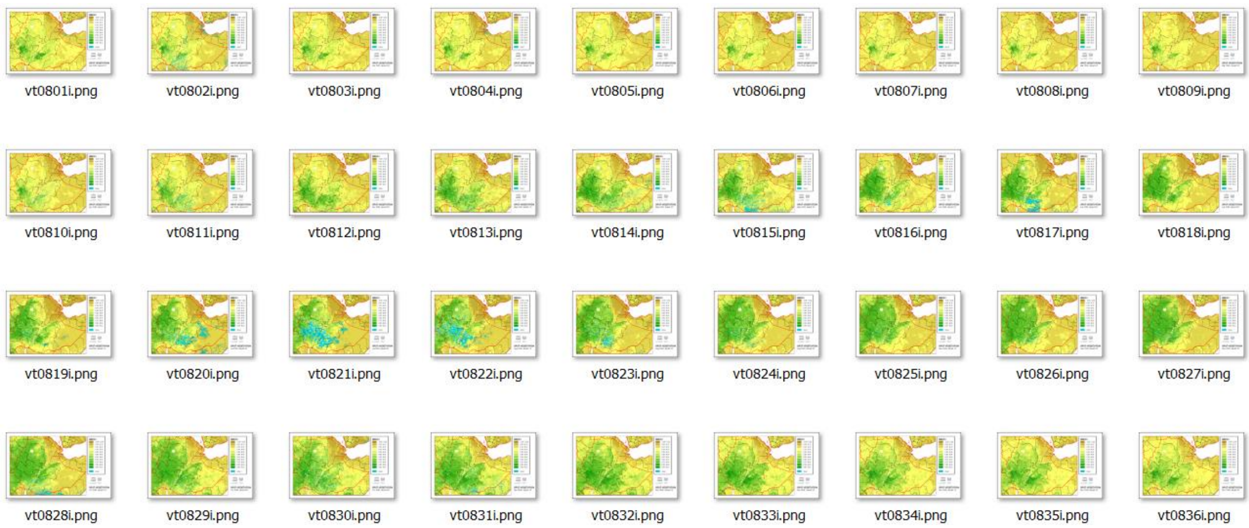
Exercise 3-2 Generation of the Maps of a Time Series

- ✓ Go to *<Analysis><Maps><Map series> <Time series>*.
- ✓ Use the button to load the Map template created in Exercise 3-1 and stored as '***REF\QNQ\NDVI_ACT_ORI.qnq***'.
- ✓ In the second field, select '.*IMG\NDVI\ACT*\\' as Input path. This is the folder where all the actual 10-daily NDVI images are stored. Meanwhile, inspect the names of the images in this directory (here: vt*YYTT*i.img).
- ✓ Select <*Dekad*> as periodicity in the drop-down menu, and enter the file name structure for the Input Filenames: "*vt*" as the prefix, "*YYTT*" as the date format and "i" as the suffix.
- ✓ For the output files, i.e. the QuickLooks files (*.png) to generate, we will use the same file names: "*vt*" as prefix, "YY*TT*" as date format and "i" as suffix.
- ✓ Select '*QLK**NDVI**ACT*' as the Output Path.
- ✓ Enter a start/end date (e.g. from 2008101 until 20081221).
- ✓ Press <*Execute*> and watch the "Task Pane" on the right.
- ✓ Expand the Task in the Task Pane by clicking on ⊞ and watch how data are processed.



Remark 1: A Task will show up in the "***Tasks Pane***" ("*Create Quick Looks RUNNING ...%*") which includes an indication of the progress. You will be able to follow the progress of the process in the <*Tasks*> and <*In progress*> tab windows. Tasks marked in yellow were not yet processed, tasks marked in green are in progress, and black marked tasks were executed without any problem. If a task is marked in red, there was an error message. Once a process was finished, it will automatically move to the 'Results' tab, where you can check the status of the processed tasks and open the task log by double clicking on any of the tasks, for example in order to check error messages (tasks marked in red).

- ✓ Use Windows Explorer to inspect the generated png-files in directory '*\QLK\NDVI\ACT*'. Open any QuickLook with any Graphical Viewer.
- ✓ Note that the dates in the Map textboxes have been adapted automatically to the dates of the NDVI images.
- ✓ Most graphical viewers (e.g. *IrfanView*, *Microsoft Picture Manager* or *Windows Picture Viewer*) allow to scan through the series of Maps by keeping the <*right arrow*> (→) pressed down. Inspect the seasonality of the vegetation in Ethiopia...



Series of QuickLooks of (non-smoothed) VGT-NDVI in Ethiopia for the 36 dekads in 2008.

Remark 2: In the menu "Create Quick Looks", use the "File" tab to save this "task" as a "***TnT***"-file. Here for instance as: "***D:\TUTORIAL\ETH\SPI\NDVI_ACT_ORI_QLK.tnt***".

Later on, you can come back to this same menu (***<Analysis><Maps><Map series> <Time series>***), and re-open this task file via <File><Open>. Then it suffices to adapt the start/end dates to generate QuickLooks for another series.

Remark 3: This is a general and most practical feature of SPIRITS. The "File" tab in each individual menu can be used to save a "task" (*.tnt) and to recall it later.

Part 4 Basic SPIRITS routines

Exercise 4-1 Smoothing

Dekadal (S10) composites, such as our NDVI images from SPOT-Vegetation (VGT) over Ethiopia, often still contain a lot of perturbations. Below-normal vegetation indicator values may appear in regions where insufficient registrations are available for the maximum value compositing (MVC) process. Missing values occur for example in winter at higher latitudes. The most important source of noise however are clouds, because clouds often persist longer than 10 days. In temporal profiles, clouds can be recognized as irregular dips (local minima in the temporal profile). These perturbations are sometimes so prevalent that they influence the analysis of the original composites. The simplest solution is to use a longer compositing period, and for instance create monthly instead of 10-daily MVC, but this sacrifices temporal resolution. Therefore, several procedures were developed for 'smoothing' the 10-daily image series, for instance BISE[2], SWETS[3], etc. In this exercise we use a modified Swets approach, as described by Klisch et al[4].

The objective of this exercise is to smooth the time series of VGT-NDVI, in order to reduce the effect of clouds and atmospheric noise on the dekadal images.

✓ Open the user interface via ***<Processing> <Temporal> <Smoothing> <Smooth>***.

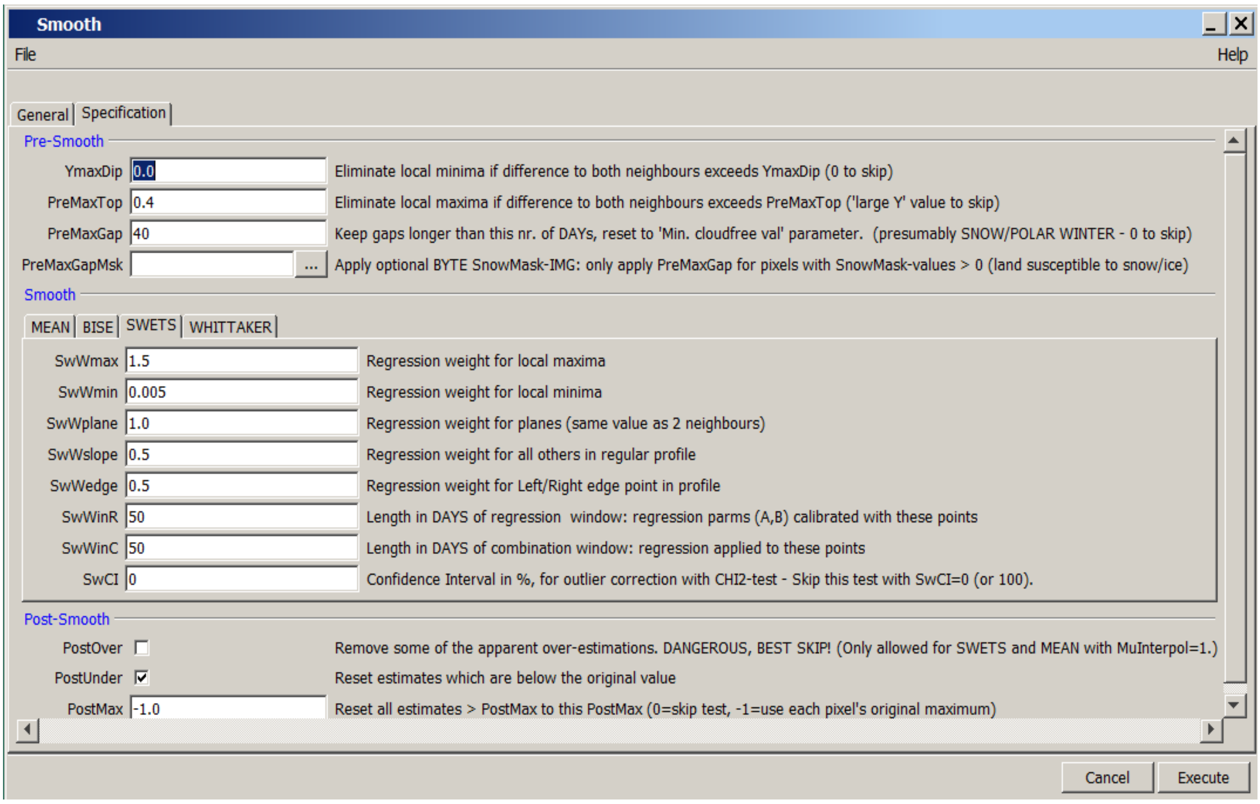
! Smoothing is a complex operation which requires many parameters, but in this exercise we will use the default settings. For the explanation of all settings, see the SPIRITS Manual (click <*Help*> in the upper right corner of the 'Smooth' menu). The **Specification tab** allows adjusting the many parameters. Four smoothing methods are available, but in this exercise we will use the SWETS procedure with the default settings which were tuned for NDVI.

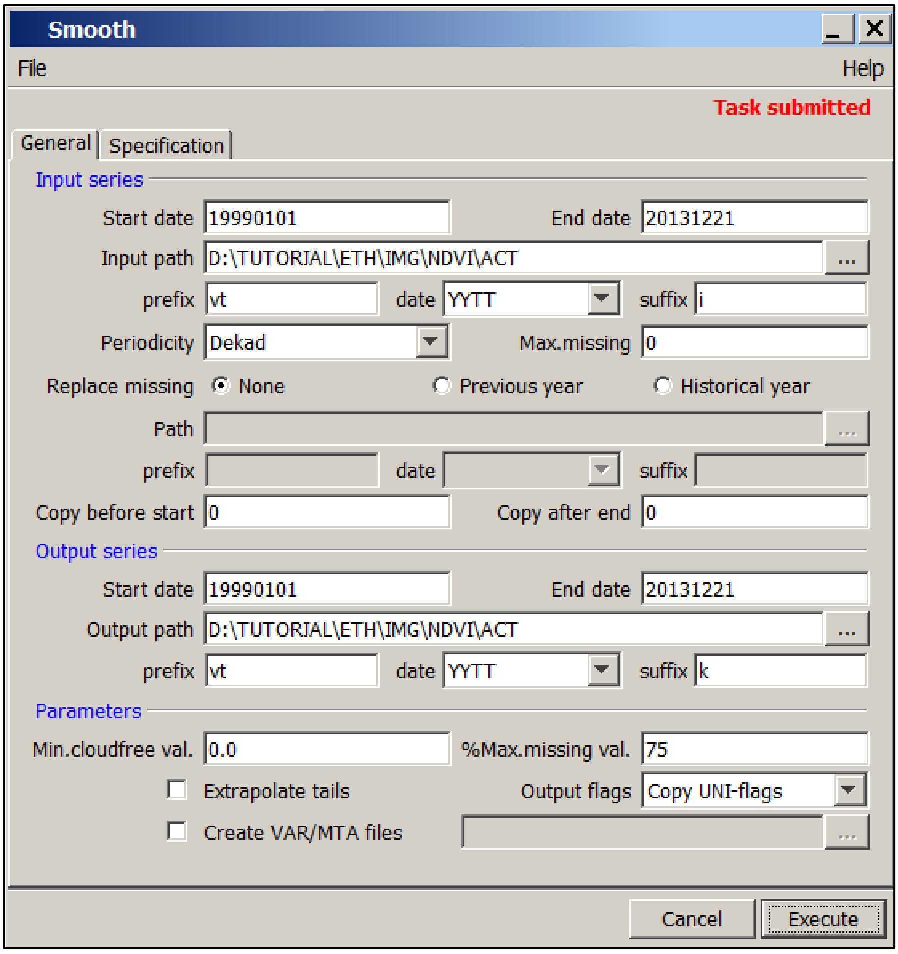
✓ So first open the "Specification" tab and select "SWETS" as smoothing method. All the other parameters can be left unchanged.

✓ In the "General" tab, define the input parameters (for instance):
- The in-period runs from "*19990101*" until "*20131221*" (15 years) and the input path is '*\IMG\NDVI\ACT*'.
- The file name *prefix* is "*vt*", *date* format is "*YYTT*", and *suffix* is "*i*".
- Max. Missing (Centre): This is the maximum number of consecutively missing actual images allowed in the centre of the time series. It is possible to replace missing images with interpolated values. But if the input series is complete (as is the case for Ethiopia), this can be set to "*0*".

---

[2] Viovy, N., Arino, O., Belward, A., 1992. The best index slope extraction (BISE) : a method for reducing noise in NDVI time-series. International Journal of Remote Sensing, 13(8), 1585-1590.
[3] Swets, D., Reed, B., Rowland, J., Marko, S., 1999. A weighted least-squares approach to temporal NDVI smoothing, ASPRS Annual Conference, Portland, Oregon, pp. 526-536.
[4] Klisch, A., Royer, A., Lazar, C., Baruth, B., Genovese, G., 2006. Extraction of phenological parameters from temporally smoothed vegetation indices. ISPRS Archives XXXVI-8/W48 Workshop proceedings: Remote sensing support to crop yield forecast and area estimates.


Smooth
File
Help
General
Specification
Pre-Smooth
YmaxDip 0.0
Eliminate local minima if difference to both neighbours exceeds YmaxDip (0 to skip)
PreMaxTop 0.4
Eliminate local maxima if difference to both neighbours exceeds PreMaxTop ('large Y' value to skip)
PreMaxGap 40
Keep gaps longer than this nr. of DAYs, reset to 'Min. cloudfree val' parameter. (presumably SNOW/POLAR WINTER - 0 to skip)
PreMaxGapMsk
Apply optional BYTE SnowMask-IMG: only apply PreMaxGap for pixels with SnowMask-values > 0 (land susceptible to snow/ice)
Smooth
MEAN
BISE
SWETS
WHITTAKER
SwWmax 1.5
Regression weight for local maxima
SwWmin 0.005
Regression weight for local minima
SwWplane 1.0
Regression weight for planes (same value as 2 neighbours)
SwWslope 0.5
Regression weight for all others in regular profile
SwWedge 0.5
Regression weight for Left/Right edge point in profile
SwWinR 50
Length in DAYS of regression window: regression parms (A,B) calibrated with these points
SwWinC 50
Length in DAYS of combination window: regression applied to these points
SwCI 0
Confidence Interval in %, for outlier correction with CHI2-test - Skip this test with SwCI=0 (or 100).
Post-Smooth
PostOver
Remove some of the apparent over-estimations. DANGEROUS, BEST SKIP! (Only allowed for SWETS and MEAN with MuInterpol=1.)
PostUnder
Reset estimates which are below the original value
PostMax -1.0
Reset all estimates > PostMax to this PostMax (0=skip test, -1=use each pixel's original maximum)
Cancel
Execute

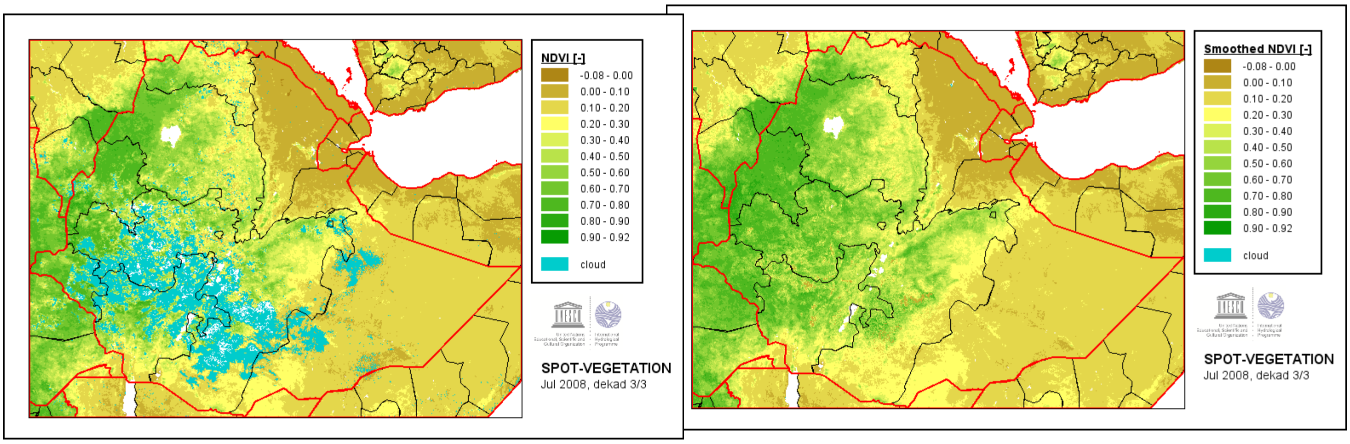

Smooth
File
Help
Task submitted
General
Specification
Input series
Start date 19990101
End date 20131221
Input path D:\TUTORIAL\ETH\IMG\NDVI\ACT
prefix vt
date YYTT
suffix i
Periodicity Dekad
Max.missing 0
Replace missing
None
Previous year
Historical year
Path
prefix
date
suffix
Copy before start 0
Copy after end 0
Output series
Start date 19990101
End date 20131221
Output path D:\TUTORIAL\ETH\IMG\NDVI\ACT
prefix vt
date YYTT
suffix k
Parameters
Min.cloudfree val. 0.0
%Max.missing val. 75
Extrapolate tails
Output flags Copy UNI-flags
Create VAR/MTA files
Cancel
Execute

- Replace missing IMGs at edges: ideally there should be images before the beginning and after the end of the input series to allow smoothing of the first and last images. In the exercise you choose the simplest option ('none'). It is possible to extend the series at the edges with images of the previous year or with the long term averages ('historical year').
- Profile tails: In order to improve extrapolation at the start/end of the in image time series, the front and tail images can be copied. But in this exercise that will not be done.
- We set the output period from "*19990101*" until "*20131221*" (the same 15 years as used for the input series).
- The smoothed images will be stored in the same path as the input series: '\IMG\NDVI\ACT'.
- As to the file names, the *prefix* remains "*vt*", and the *date* format "YYTT", **but the *suffix* is set to "*k*"**. Thus this suffix is the only feature which distinguishes the original NDVI images ("i") from the smoothed ones ("k").
- The minimum NDVI for land pixels without clouds is "*0.00*". Observations below this value are considered as missing values.
- The maximum percentage of missing values (per pixel) is "*75%*". Pixels with more than this amount of missing values will be flagged in the output images.
- Output flags: "Copy UNI-flags". In the alternative there are only "two flags" (water and missing.

✓ Save the task as '***NDVI_ACT_SMO.tnt***'.

✓ Now click on <*Execute*>. The smoothing process can take some minutes to complete.

✓ Of course: QuickLook maps can also be made from this smoothed time series.



QuickLooks of the NDVI in dekad 21 of 2008. Left: original, right: smoothed.

Exercise 4-2 Long-Term Statistics LTS (for NDVI)

The objective of this exercise is to create a series of images with the long Term Statistics (LTS) or “Historical Year” for each period (in our case: dekad) in the year. The LTS have merits on their own, but they are also needed for the later computation of “anomalies”, which compare the actual data (of a given dekad) with theire LTS analogues.

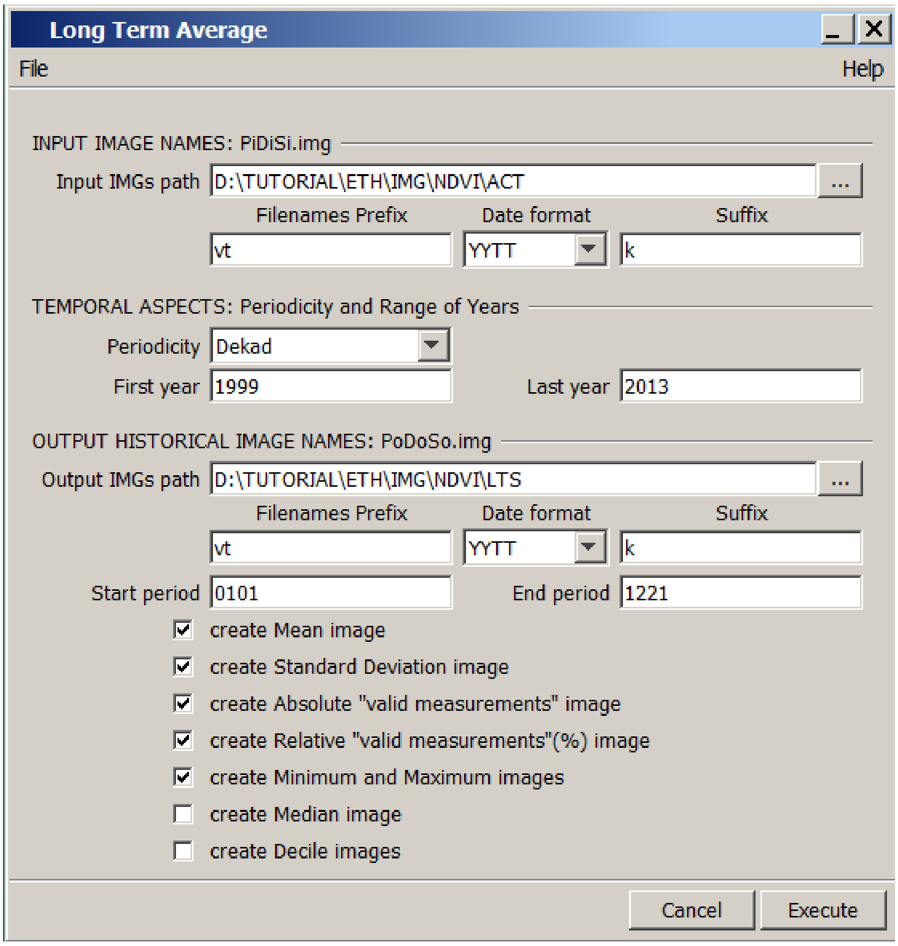
Per time step (here dekad: for instance dekad 3 of June), the procedure inspects all the images available in the multi-annual series, and computes a number of new scenes with LTS. In the first place this is the LTA (Long-Term Average). Of second importance are the LTS Minimum, Maximum, standard deviation, and the number of “good” (non-flagged) observations. Optionally, one can also compute the 11 “deciles” (P0=Minimum, ..., P50=Median, ..., P100=Maximum).

As to the file names, SPIRITS labels all these different LTS images via the (pre-remote sensing) years:

- 1950-1960: deciles (1950=Min, 1955=Median, 1960=Maximum)
- 1961: Absolute number of good observations
- 1962: LTA
- 1963: Standard deviation
- 1964: Relative number of good observations (in % of the number of available years).

Calculation of the LTS for the smoothed NDVI

✓ Go to the tool ***<Processing> <Temporal> <Long Term Statistics>***.



- ✓ Define the input image path, filename prefix, date format and suffix as '.*IMG\NDVI\ACT*', '*vt*', '*YYTT*' and 'k'. The periodicity is '*dekad*'.
- ✓ The LTS will be derived from the full series. Hence the first and last years are set to 1999 and 2013.
- ✓ The output images will be stored in the '*LTS*' subfolder. Thus "Output IMGs path"= '.*IMG\NDVI\LTS*'.
- ✓ The structure of the output file names can be identical to the input: prefix "vt", date format "YYTT", suffix "k".
- ✓ As we want to compute the LTS for all the 36 dekads in the year, "start period" must be set to 0101 (January, dekad 1), and "end period" to 1221 (December, dekad 3).
- ✓ As mentioned, per dekad 15 different output images can be generated. We only select the most important (default) ones: Mean, Standard Deviation, Minimum, Maximum and Nr. of good observations.
- ✓ If you want to save the task file, click <*File*> <*Save*> and name it "***NDVI_LTS_SMO.tnt***".
- ✓ Click <*Execute*>

Exercise 4-3 Vegetation status anomalies

In this exercise we will create vegetation anomaly maps based on the comparison of actual NDVI with the LTS.

ACT	Per	1	2	...	*t*	...	N_t
Year = 1		X(1,1)	X(1,2)	...	X(1,t)	...	X(1, N_t)
...		...	...	...	...	...	...
y		X(y,1)	X(y,2)	...	**X(*y*,*t*)**	...	X(y,N_t)
...		...	...	...	...	...	...
N_y		X(N_y,1)	X(N_y,2)	...	X(N_y,t)	...	X(N_y, N_t)

LTS	Per	1	2	...	*t*	...	N_t
Mean		μ(1)	μ(2)	...	μ(t)	...	μ(N_t)
Minimum		Min(1)	Min(2)	...	Min(t)	...	Min(N_t)
Maximum		Max(1)	Max(2)	...	Max(t)	...	Max(N_t)
St. Dev.		σ(1)	σ(2)	...	σ(t)	...	σ(N_t)
Good obs.		N(1)	N(2)	...	N(t)	...	N(N_t)

ANO	Per	1	2	...	*t*	...	N_t
Year = 1		A(1,1)	A(1,2)	...	A(1,t)	...	A(1, N_t)
...		...	...	...	...	...	...
y		A(y,1)	A(y,2)	...	**A(*y*,*t*)**	...	A(y,N_t)
...		...	...	...	...	...	...
N_y		A(N_y,1)	A(N_y,2)	...	A(N_y,t)	...	A(N_y, N_t)

Starting point is a multi-annual series of actual (ACT) images X (N_y years x N_t periods per year, X=any variable such as NDVI, rainfall, ...). Program LTA derives a number of new scenes with for each period t the Long-Term Statistics (LTS) such as the mean μ, standard deviation σ and number of good (non-flagged) observations. Program DIFFER compares the ACT data with their LTS-analogues and computes new images *with anomaly (ANO) indicators A, for the concerned variable X and again for all the N_y years and N_t periods. To this goal, various "difference operators" are available.*

DIFFERENCE OPERATOR [1-12]	NB: y=year, p=period in year (day/dekad/month)
1. Abs. Difference to previous period	ADpp(y,p) = X(y,p) – X(y, p-1)
2. Rel. Difference to previous period	RDpp(y,p) = [X(y,p) – X(y, p-1)] / X(y, p-1)
3. Abs. Difference to previous year	ADpy(y,p) = X(y,p) – X(y-1, p)
4. Rel. Difference to previous year	RDpy(y,p) = [X(y,p) – X(y-1, p)] / X(y-1, p)
5. Abs. Difference to LTS-Median	ADhm(y,p) = X(y,p) – MEDIAN(p)
6. Rel. Difference to LTS-Median	RDhm(y,p) = [X(y,p) – MEDIAN(p)] / MEDIAN(p)
7. Abs. Difference to LTS-Mean (LTA)	ADha(y,p) = X(y,p) – LTA(p)
8. Rel. Difference to LTS-Mean (LTA)	**RDha(y,p) = [X(y,p) – LTA(p)]/LTA(p)**
9. Standardized difference	SDh(y,p) = [X(y,p) – LTA(p)] / StDEV(p)
10. Relative range	RRh(y,p) = [X(y,p) – MIN(p)] / [MAX(p) – MIN(p)]
11. Historical probability	HPh(y,p) = Prob. of X(y,p) in historical distribution
12. Classified historical probability	CPh(y,p) = Idem but 10 classes: 0-10%...90-100%

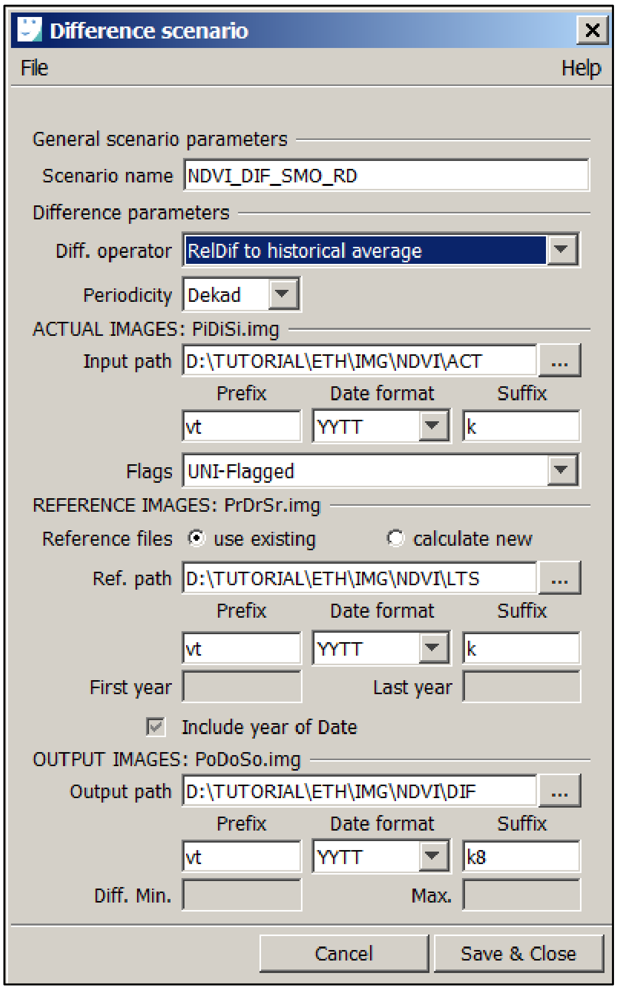
! In the calculation of anomalies, the current dekad is compared to a certain "reference situation". This can be the same dekad in the 'historical year' or in the previous year, or simply the previous dekad.

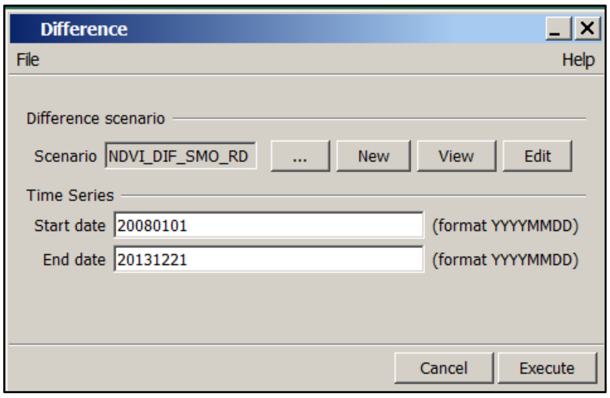
! A wide range of "Difference operators" is available (see table), but they mostly yield similar results. In this exercise we will use Operator 8: the Relative Difference w.r.t. the LTA:

$$\mathbf{RD_{y,p} = (X_{y,p} - LTA_p) / LTA_p}$$

with *y*=year and *p*=dekad in year.

✓ Open **<*Processing*> <*Temporal*> <*Anomalies*> <*Time series*>** and first create a <*New*> scenario.





✓ Name the scenario "*NDVI_DIF_SMO_RD*".

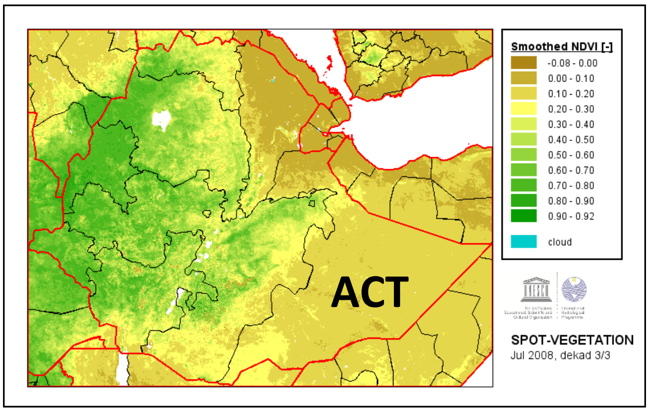
- ✓ Select difference operator ‘*RelDif to historical average*’ and periodicity ‘Dekad’.
- ✓ The ACTUAL images are the smoothed NDVI, with prefix ‘vt’, date format ‘YYTT’ and suffix ‘k’.
- ✓ The flags of the NDVI images follow the “UNIflags” system (see p 7). Hence in the entry Flags choose “*UNI-Flagged*”.

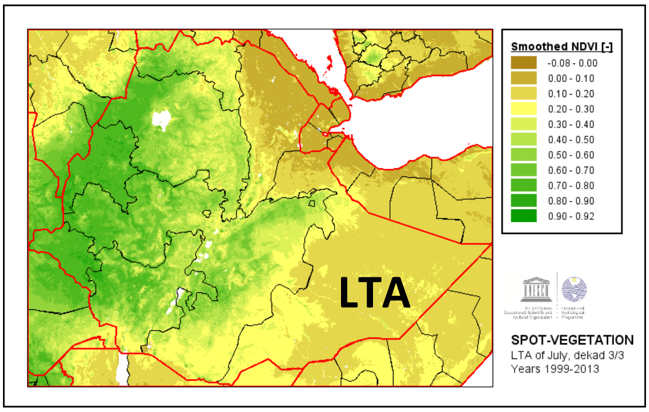
! You can check the input image headers by opening one of them in a text editor, or by using the <*View HDR*> operation in the File menu. In this case the input flag values are: 251=missing, 252=cloud, 253=snow, 254=sea, 255=back, 254=back. By checking this option, also the generated anomaly images will follow the UNIflags system.

- ✓ For ‘REFERENCE IMGs’, use the LTS which was computed in the previous exercise. Specify the path (‘\IMG\NDVI\LTS’), prefix (vt), date format (YYTT) and suffix (k).

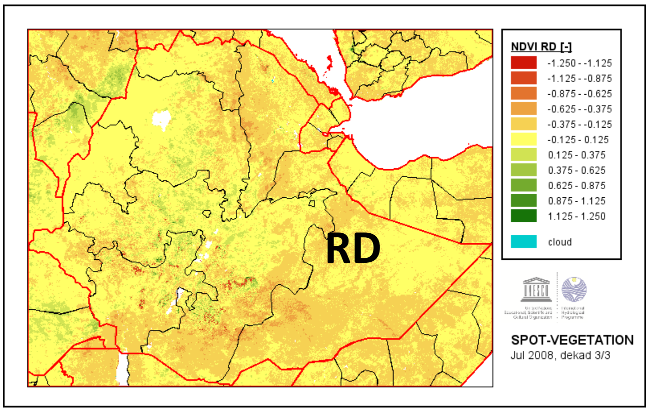
! When the LTS has not been computed in advance, or if you want to use LTS different from the existing one, it can be done by checking the “***calculate new***” option. In this case the computing time will increase significantly.

- ✓ The anomaly output images will be stored in ‘\IMG\NDVI\DIF’, again with prefix “vt” and date format “YYTT”. For the suffix, we use “k8”, because the selected RD is the 8^{th} operator in the list.
- ✓ Click <*Save & Close*> and save the difference scenario as “***.\SPI\NDVI_DIF_SMO_RD.sns***”.
- ✓ In the original “Difference” menu, specify the period for which the difference scenario must be applied. In the above figure, we selected all dekads for the years 2008 to 2013.
- ✓ Click <*Execute*> to start the task.





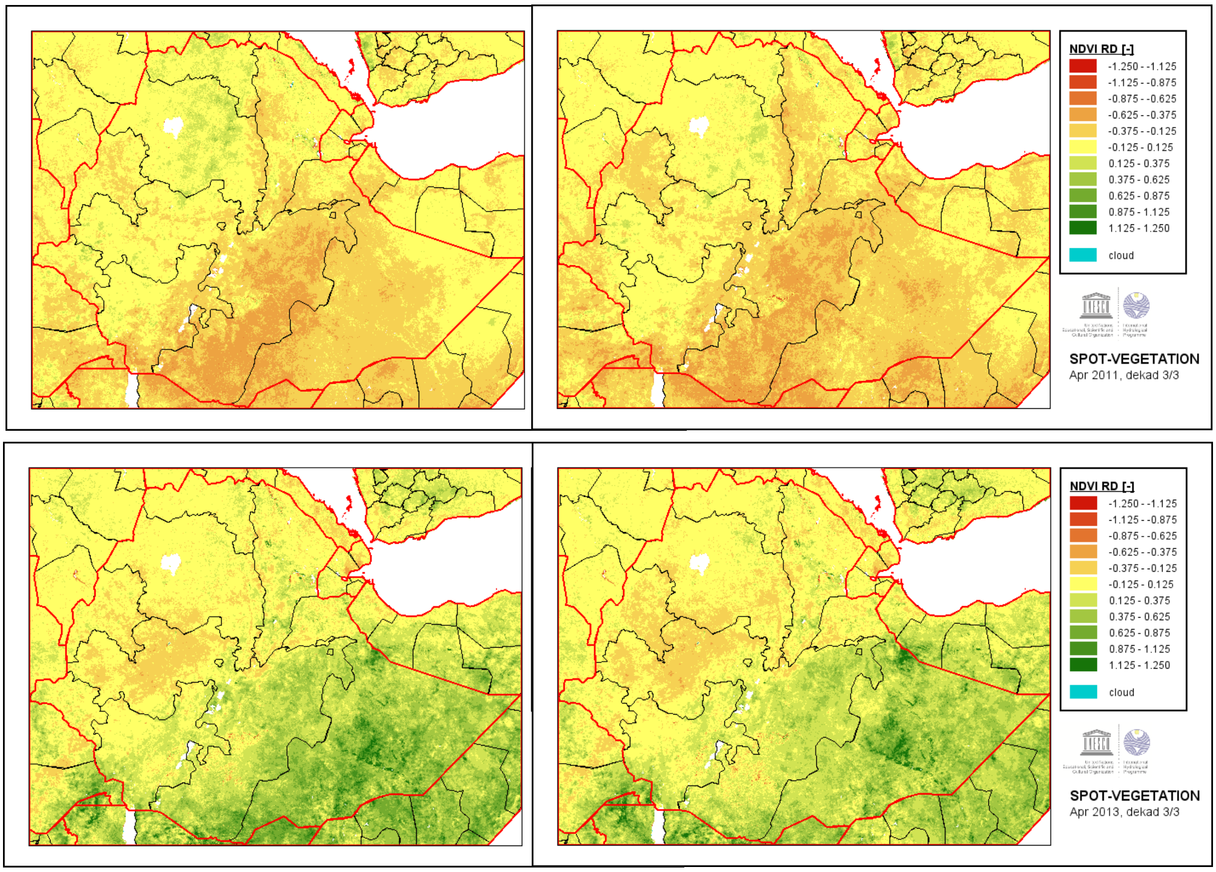
Dekad 3 of July 2008: ACT, LTA and RD.



DEKAD 2 of APRIL   DEKAD 3 of APRIL

2011

2013



Relative difference of smoothed NDVI against the LTA (over 1999-2013)
for the last two dekads of April in years 2011 (drought) and 2013 (normal).

Other optional exercises:

- Try out some other difference operators (see table above), for instance:
  - 9 = Standardize difference (z-score)
  - 10 = Relative range. When applied on NDVI, the resulting anomaly is called VCI (Vegetation Condition Index).
  - 11 = Historical probability. When applied on NDVI, the result is known as VPI (Vegetation Productivity Index).
- The same procedures (LTS + Anomalies) can also be run on the rainfall data. However, in this case it is more common practice to compute the SPI. See *<Processing> <Temporal> <Standardized Precipitation Index>*.
- In all cases, QuickLook templates and maps can be prepared for the generated time series (LTA and anomalies.

Part 5 Extraction of Statistics

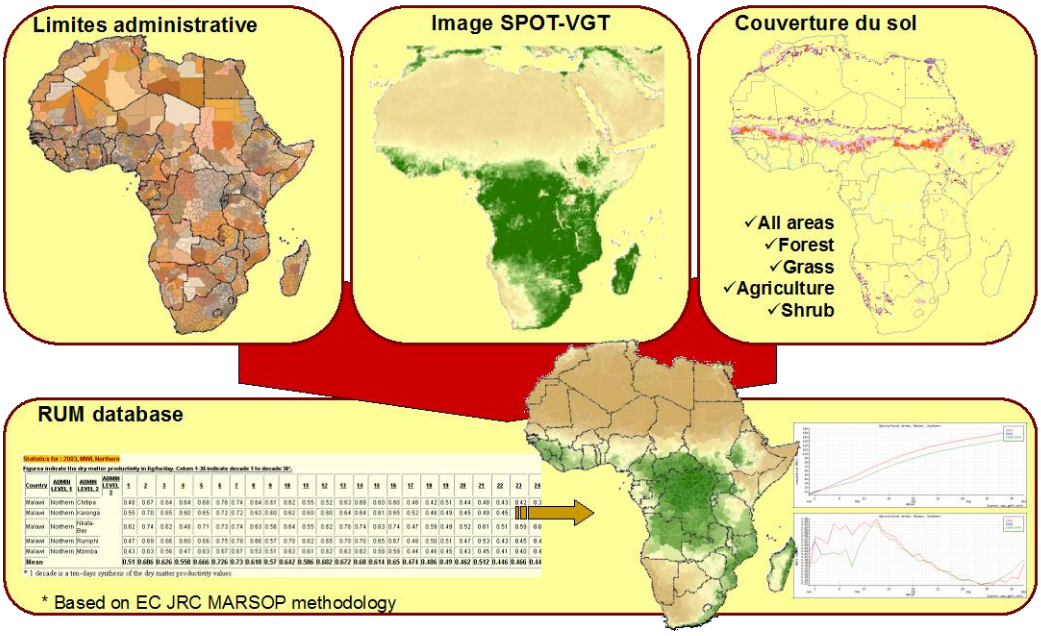
The principles of this procedure are outlined in the figure below. GLIMPSE program IMG2RUM extracts an ASCII-formatted RUM-file with "Regional Unmixed Means" (RUM) from any RS-image. Essential inputs are the names of the concerned RS-image and of the associated "regions" image. The latter can be any higher-level zonation, which groups the pixels in spatial units (administrative, agro-ecological, meteorological grid cells,...). In this most simple case the retrieved RUM-files only contain the mean of all pixel values for each of these **regions *r* (*r*=1→*N*r)**.

But the procedure becomes more interesting when the data have to be "unmixed" for different land use (LU) **classes *k* (*k*=1→N_k)**. The concerned LU information has to be provided in one of the following ways:

- **Hard**: In the form of a single classification image, where each pixel is assigned univocally to a single class.
- **Soft**: Via a set of "Area Fraction Images" (AFIs), one for each class *k*, and expressing per pixel the **area fraction f_k** (0-100%) covered by the concerned class *k*.

Whenever such LU information is provided (hard or soft), the retrieved files contain "real" RUM-values, i.e. the mean of the RS-values per region and unmixed per class. Moreover, in the "soft" mode where the land use information is given via AFIs, two modalities arise:

- The final RUM-values (per region and class) are "weighted means" based on the area fractions f_k: pixels with high f_k contribute more to the overall RUM-mean.
- Class-specific **"rejection thresholds" T_k** can be set in advance to withdraw all pixels whose area fraction f_k falls below them.



If we label the regions as *r*, the classes as *k* (with k=0 standing for "any class") and the target variable (NDVI, rainfall,..., for a given date) as *X*, then the computation of the RUM-values (alias $\mu_{r,k}$) goes as follows – depending on the provided LU information:

- None: $\mu_{r,0} = \Sigma X/N_r$. The summation runs over the number of pixels N_r in the concerned region.
- Hard: $\mu_{r,k} = \Sigma X/N_{r,k}$, with summation over the $N_{r,k}$ pixels in region *r* and belonging to class *k*.
- Soft: $\mu_{r,k} = \Sigma f_k.X/\Sigma f_k$. This RUM value covers all N_r pixels in the region, but each one weighted as to its area fraction f_k for the concerned class. Moreover, if a rejection threshold T_k is specified all pixels with $f_k<T_k$ are withdrawn from the calculation.

Finally, it must be remarked that all pixels with missing or “flagged” input X-values are also excluded from the RUM-computations. Of course, this number may fluctuate over time.

The process can be repeated for all the images in a time series and the results can afterwards be incorporated in an overall database. This “**spatial aggregation**” via IMG2RUM provides **many advantages**:

- It realizes an important data reduction because the “region” replaces the “pixel” as elementary spatial unit.
- The differences between various sensors and spatial resolutions are neutralized. Thanks to the regional averaging process, in the form of RUM-values $\mu_{r,k}$, all data become comparable.
- Most agro-statistical data (crop areas, yields, productions,...) are available per administrative unit. The RUM-conversion adapts the RS-information (originally per pixel) to the same level.
- Whereas the images can only be inspected via specialised image processing software (ENVI, SPIRITS,...), the RUM-files can be analysed with easier and widespread GIS and database tools.

In practice, the extraction of image statistics in SPIRITS involves the following steps and exercises:

1. Preparation of the ancillary images with “Regions” and “Land Use Classes”.

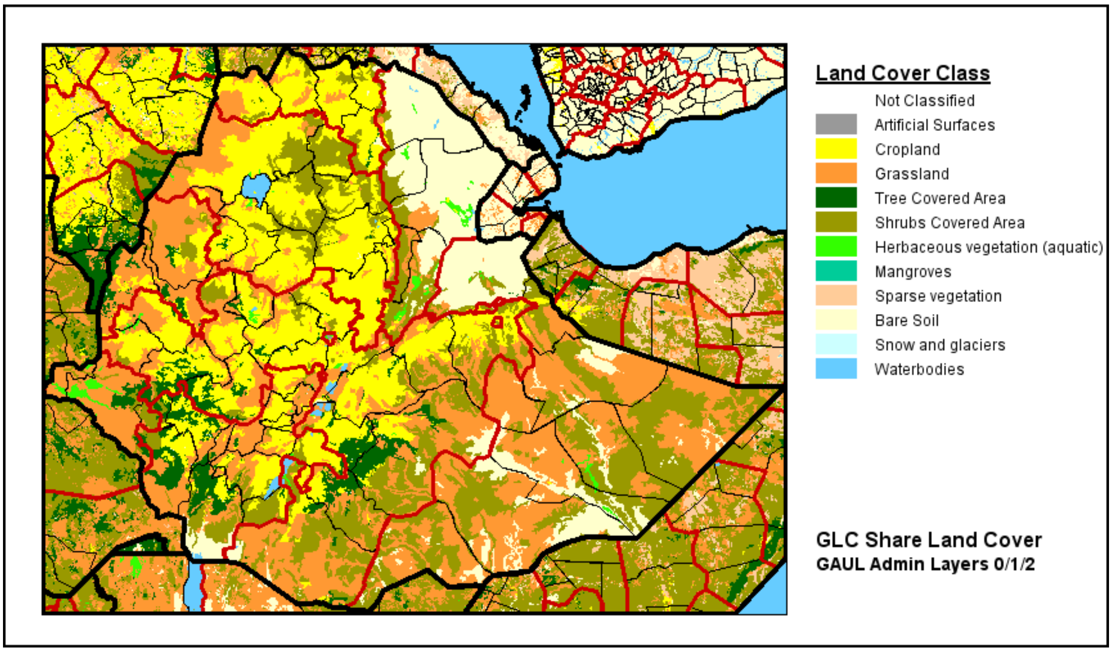
2. Initialisation of the RUM-database: declaration of sensors, variables, regions and classes.

3. Extraction of the RUM-values from a time series of input images.

4. Visualisation of the RUM-statistics.

Exercise 5-1 Preparation of the ancillary images

In this tutorial we will treat the most simple case where the LU-information comes from a hard classification: FAO’s GLC_SHARE land cover map (glcn.org). It can be found under: ***d:\TUTORIAL\ETH\REF\MSK\GLC_SHv10.img***. The same folder also contains the corresponding QuickLook (GLC_SHv10.png), which is shown below.



Also the contents of file **GLC_SHv10.hdr** are listed below. In a spatial sense (samples, lines, map info) it is identical to the NDVI images (see page 7). But here there are three new keywords:

- "Classes=12" indicates that there are 12 LU-classes, with "class-IDs" consecutively ranging from 0 to 11.
- "Class names = {...}" lists their names, separated by commas.
- "Class lookup = {...}" gives for each class the RGB colour values.

Thus, there are 11 "real" classes (ID: 1-11). Class 0 must be present but represents the "not classified" pixels. To remind: cropland, grassland and water respectively have ID=2, 3 and 11.

```
ENVI
description = {Dominant land cover layer from the GLCshare dataset}
samples = 1681
lines = 1289
bands = 1
header offset = 0
file type = ENVI classification
data type = 1
interleave = bsq
map info = {Geographic Lat/Lon, 1, 1, 32.9955357, 14.90625, 0.00892857143, 0.00892857143}
values = {classes, -, 1, 11, 1, 11, 0, 1}
flags = {0=not classified}
comment = {Extract from w:\dmpval\sigma\ref\msk\glc_shv10_dom.img: Band 1/1, ROI (UL=23857/6732, BR=25537/8020)}
program = {IMGcvt.exe (V906)}
classes = 12
class names = { Not Classified, Artificial Surfaces, Cropland, Grassland, Tree Covered Area, Shrubs Covered Area,
                Herbaceous vegetation (aquatic),  Mangroves,  Sparse vegetation, Bare Soil,  Snow and glaciers,  Waterbodies }
class lookup = {
  255,255,255,
  153,153,153,
  255,255,  0,
  255,153, 51,
    0,102,  0,
  153,153,  0,
   51,255,  0,
    0,204,153,
  255,204,153,
  255,255,204,
  204,255,255,
  102,204,255
}
```

For the **administrative regions**, we started from the **GAUL*n* vector files** shape files, with n=0 (countries), n=1 (provinces), n=2 (districts). The limits of all these regions are also shown in the above figure (in overlay to the GLC_SHv10 map). Important is to also have a look at the contents of the **GAUL*n*.dbf** files, the "**attribute tables**" which come together with each SHP-file. This can for instance be done with EXCEL. The figure below shows the first 23 lines of the file GAUL1.dbf with the definitions of the provinces. Each spatial unit (here: province) has a unique region_ID ("ADM1_CODE"), a name ("ADM1_NAME") and the table also indicates to which higher level entity the unit belongs: in this case the country via "ADM0_NAME". It appears that Ethiopia covers 11 different GAUL1 units or provinces. Ceteris paribus, the same type of information is also provided by the other DBF-files. But of course the number of Ethiopian units increases from GAUL0 (1) over GAUL1 (11) to GAUL2 (75).

For the RUM-extraction we will use raster versions of these three SHP-files. These were generated with SPIRITS via <Import/Export><Vectors><Rasterize SHP-file> and have the following characteristics:

- In a spatial sense they are identical to the NDVI images (same nr. of columns and lines, same map info).
- Each pixel is labelled with the "region_ID" of the unit to which it belongs (see field ADM*n*_CODE).
- Because these IDs vary between 1 and huge numbers (>32768), the data type is LONG INTEGER.

These images can be found in ***d:\TUTORIAL\ETH\REF\REG\GAULn.img***.

	A	B	C	D	E	F	G	H
1	ADM1_CODE	ADM1_NAME	STR1_YEAR	EXP1_YEAR	STATUS	DISF	ADM0_CODE	ADM0_NAME
2	1093	Ali Sabieh	1000	3000	Member State	NO	70	Djibouti
3	1094	Dikhil	1000	3000	Member State	NO	70	Djibouti
4	1095	Djibouti	1000	3000	Member State	NO	70	Djibouti
5	1096	Obock	1000	3000	Member State	NO	70	Djibouti
6	1097	Tadjourah	1000	3000	Member State	NO	70	Djibouti
7	1207	Debub	1000	3000	Member State	NO	77	Eritrea
8	1208	Debubawi Keih Bahri	1000	3000	Member State	NO	77	Eritrea
9	1209	Gash Barka	1000	3000	Member State	NO	77	Eritrea
10	1211	Semenawi Keih Bahri	1000	3000	Member State	NO	77	Eritrea
11	**1227**	**Addis Ababa**	**1994**	**3000**	**Member State**	**NO**	**79**	**Ethiopia**
12	**1228**	**Afar**	**1994**	**3000**	**Member State**	**NO**	**79**	**Ethiopia**
13	**1229**	**Amhara**	**1994**	**3000**	**Member State**	**NO**	**79**	**Ethiopia**
14	**1230**	**Beneshangul Gumu**	**1994**	**3000**	**Member State**	**NO**	**79**	**Ethiopia**
15	**1231**	**Dire Dawa**	**1998**	**3000**	**Member State**	**NO**	**79**	**Ethiopia**
16	**1232**	**Gambela**	**1994**	**3000**	**Member State**	**NO**	**79**	**Ethiopia**
17	**1233**	**Hareri**	**1994**	**3000**	**Member State**	**NO**	**79**	**Ethiopia**
18	**47678**	**Oromia**	**1994**	**3000**	**Member State**	**NO**	**79**	**Ethiopia**
19	**1236**	**SNNPR**	**1994**	**3000**	**Member State**	**NO**	**79**	**Ethiopia**
20	**47679**	**Somali**	**1998**	**3000**	**Member State**	**NO**	**79**	**Ethiopia**
21	**1237**	**Tigray**	**1994**	**3000**	**Member State**	**NO**	**79**	**Ethiopia**
22	61033	Administrative unit not available	1000	3000	Sovereignty unsettled	YES	61013	Ilemi triangle
23	51327	Eastern	1000	3000	Member State	NO	133	Kenya

First lines in the attribute table GAUL1.DBF.

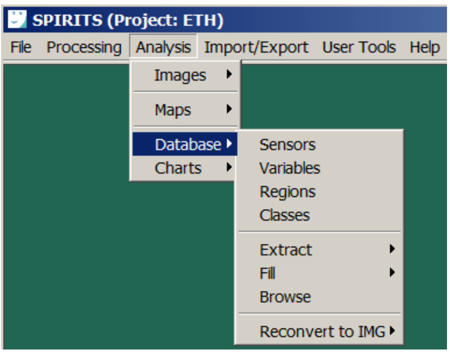
Exercise 5-2 Initialisation of the RUM Database

For each project (e.g. ETH) SPIRITS maintains a single database. But this database can comprise the RUM-data of different resources:

- Different sensors (or sources) such as SPOT-VEGETATION, ECMWF, MODIS,... regardless the spatial framing of the underlying images.
- Different variables such as NDVI, NDVI anomalies, rainfall, rainfall anomalies, etc...
- Different "region sets". In our case we have three: GAUL0, GAUL1 and GAUL2. Each set has its own list of "region-IDs".
- Different "classes sets". In this exercise we only have a single one: GLC_Share with its 11 classes. But the RUM-values for another classification (e.g. a more detailed crop map) could be added as well to the system.

However, the database does not store names (or "strings" – that would be space-consuming) but rather numerical IDs. That was already evidenced for the classes ("class_IDs") and regions ("region_IDs"). But the same holds for the sensors and the variables. So we will also have "sensor_IDs" and "variable_IDs".

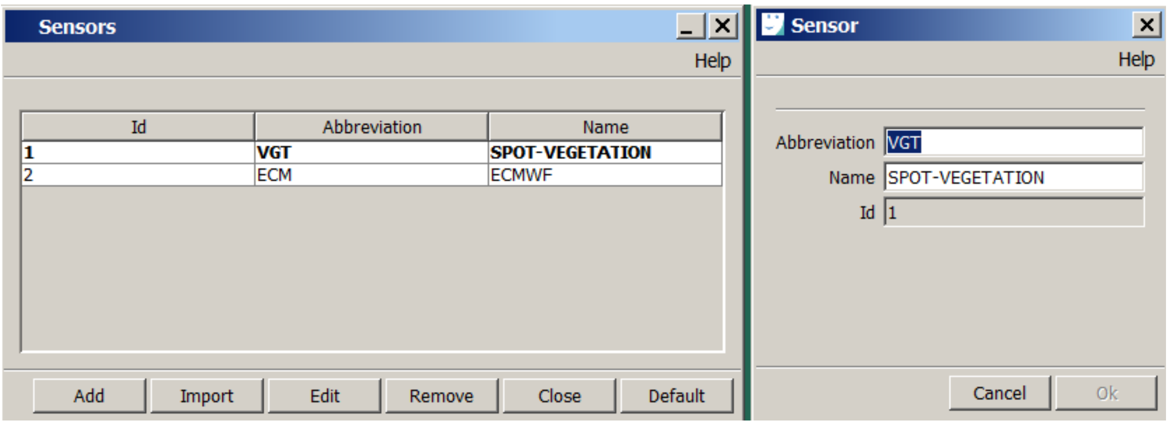
But in view of the later analyses and visualizations – and also for internal checks – SPIRITS needs to know the semantic meaning of each ID. Before the real RUM extraction can start, we first need to provide "translation tables" for the four mentioned elements. This is the subject of this exercise.



The above figure shows the contents of the “DATABASE” menu. Below we will treat the first 4 entries.

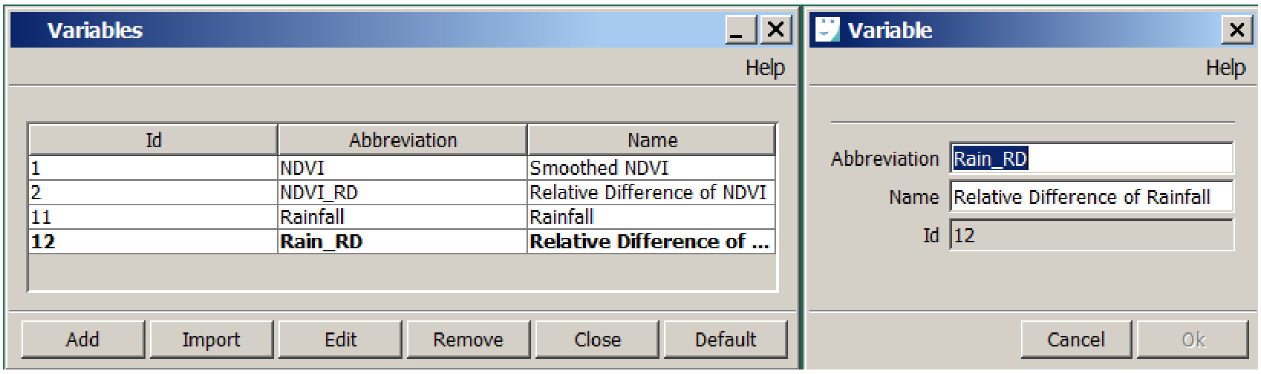
Declaration of the Sensors

- ✓ Go to ***<Analysis> <Database> <Sensors>*** and add some sensors as indicated below. NB: It’s here that we assign an ID to each sensor.
- ✓ “Close” saves the data to the database.



Declaration of the Variables

- ✓ Go to ***<Analysis> <Database>* <Variables>** and add some sensors as indicated below. NB: It’s here that we assign an ID to each variable.
- ✓ “Close” saves the information to the database.



Declaration of the “Region Sets”

- ✓ Go to ***<Analysis> <Database>* <Regions> and first** click *<Add>* in the top panel. A small menu pops up in which you can define the settings of the first “**Regions Set**”. In this case: Abbreviation=GAUL1, Name=GAUL1, ID=1. When the “OK” button is clicked, the new region sets (GAUL1) appears in the general menu.
- ✓ Via the lower panel we now have to define the region_IDs and names of all the units in this GAUL1 set. This lower panel becomes “active” when you click on the GAUL1-line, just added in the top panel. Then click on the “Import” button to open the “**Import Regions**” menu.

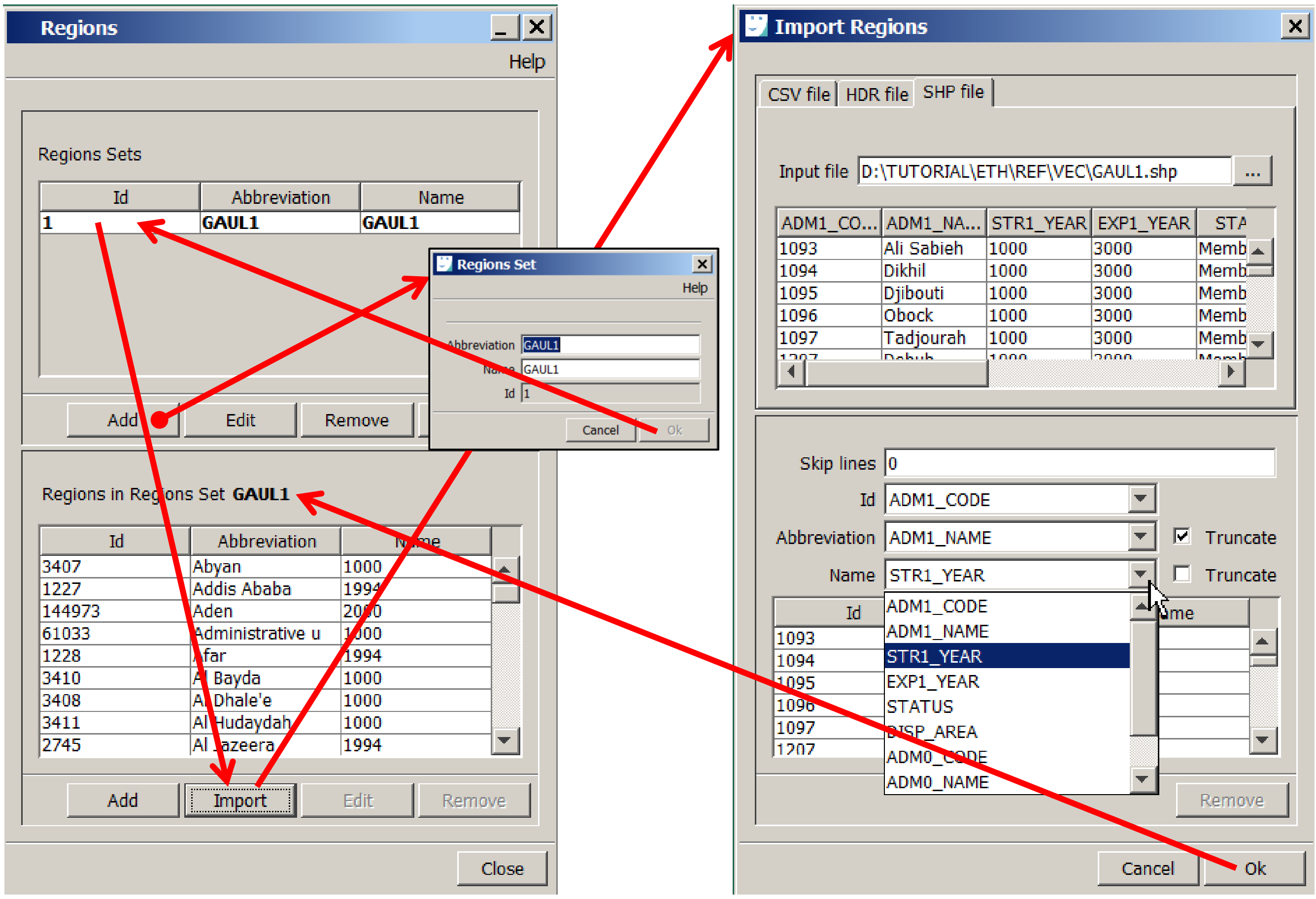
- ✓ In general, the requested information can be acquired from a CSV text-file, a HDR-file or a SHP-file, but in 99.99% of the cases the last option will be used. So we **select the “SHP” tab** and specify the concerned vector file (.\REF\VEC\GAUL1.shp).

! *This terminology might be misleading. Actually the region_IDs and their names are not extracted from the SHP-file but from the associated *.DBF attribute table!*

- ✓ The top of this “Import Regions” menu now displays the full contents of this GAUL1.dbf file. In the lower panel, the requested information can be specified, starting from the initial suggestions made by SPIRITS – in this case:
    - “Skip line=0” is correct.
    - “ID=ADM1_CODE” is correct. This DBF-field indeed contains the region_IDs.
    - “Abbreviation=ADM1_NAME” is also correct. The DBF-file does not contain abbreviations for the regions, so using the full names is the only alternative. But by enabling “Truncate” the abbreviated strings will at most contain the 16 leftmost characters.
    - “Name=STR1_YEAR” is completely wrong. It should be reset to “**Name=ADM1_NAME**”. And in this case “Truncation” should obviously be disabled.

  The applied adaptations become visible in the lower table of the “Import Regions” menu. When finished and the “OK” button is pushed, this menu disappears and the final list of region_IDs and their names appears in the bottom table of the original “Regions” menu.

- ✓ Pressing the “Close” button saves all the information of this Regions Set to the database.

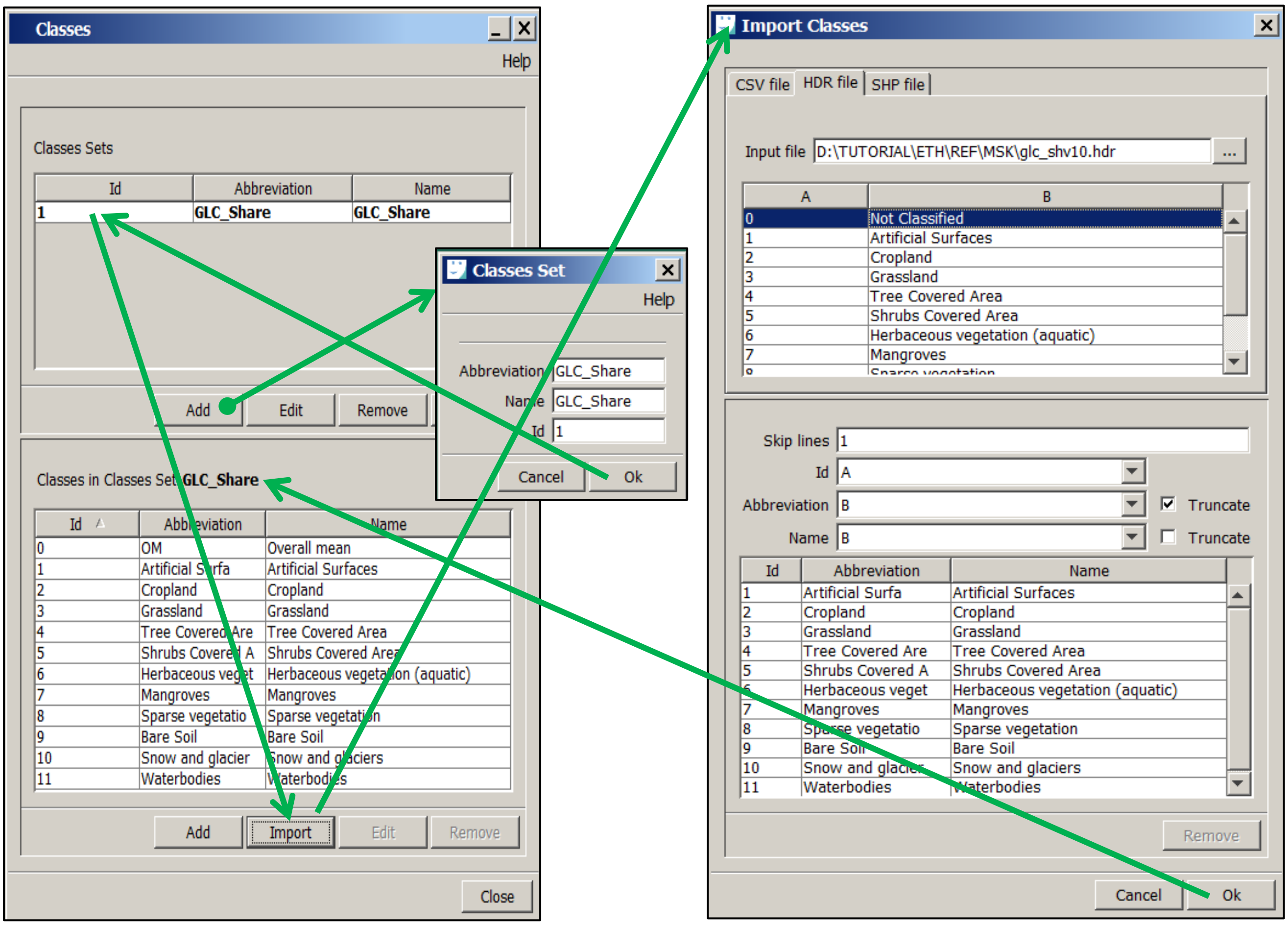


NB: Optionally you can repeat the entire procedures for two other “Region Sets”: GAUL2 and GAUL0.

Declaration of the "Classes Sets"

This procedure follows the same logics as the one for the declaration of the regions. But in this case we only have one single "classes set": the hard classification of GLC-Share.

- ✓ Go to ***<Analysis> <Database>* <Classes> and first** click *<Add>* in the top panel. A small menu pops up in which you can define the settings of the first "**Classes Set**". In this case: Abbreviation=GLC_Share, Name=GLC_Share, ID=1. When the "OK" button is clicked, the new classes set (GLC_Share) appears in the general menu.



- ✓ Via the lower panel we now have to define the IDs and names of all the classes in this GLC_share set. This lower panel becomes "active" when you click on the "GLC_Share" line, just added in the top panel. Then click on the "Import" button to open the "**Import Classes**" menu.

- ✓ In general, the requested information can be acquired from a CSV text-file, a HDR-file or a SHP/DBF-file, but in this case we use the second option because the HDR of our classification IMG does contain the class names (see page 30). So we **select the "HDR" tab** and specify the concerned image file (.\REF\MSK\GLC_SHv10).

- ✓ The top of the "Import Classes" menu now displays the contents of this GLC_SHv10.HDR file, spread over two columns, labeled as A (Class_IDs) and B (Class names). In the lower panel, the requested information can be specified, starting from the initial suggestions made by SPIRITS – in this case:
  - "Skip line=0" is wrong and should be reset to "Skip line=1". Explanation follows...
  - The other settings are correct: "ID=A", "Abbreviation=B" (but again enable "Truncate"), "Name=B".

The applied adaptations become visible in the lower table of the “Import Classes” menu. When finished and the “OK” button is pushed, this menu disappears and the final list of Class_IDs and their names appears in the bottom table of the original “Classes” menu.

- ✓ Pressing the “Close” button saves all the information of this Classes Set to the database.

Remark: Why “Skip line=1”?

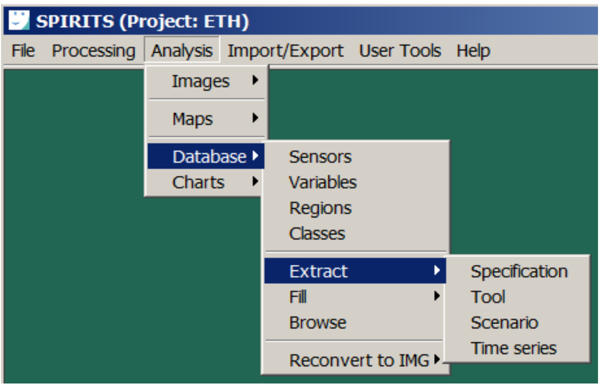
The lower panel of the general “Classes” menu lists the classes in any of the declared sets. But this list always and unavoidably contains a “zero-class” (ID=0, Abbreviation=OM, Name=Overall Mean). In the RUM-files, this class will always be present and represent the overall mean of all the pixels in each region – thus without “unmixing” per LU-class. But this conflicts with the ENVI approach where any classification always has to contain a 0-class representing the group of irrelevant pixels which were “not classified” or which fall beyond the area of interest. Hence we have to eliminate this ENVI zero-class by setting “Skip line=1”.

Some general remark about all these declarations

- The SPIRITS database (one per project!) is maintained in the sub-folder ‘***SpiritsProjectData***’ which is automatically created when a directory is declared as a “project”. Thus in our case in ‘d:\TUTORIAL\ETH\ *SpiritsProjectData*’**.**
- The IDs of the regions and classes are derived from the underlying data (SHP/DBF-files and LU-classifications). The IDs of the sensors and variables can be chosen freely, but once defined they should never be changed. For instance “SPOT-VEGETATION” should remain associated to sensor_ID=1. On the other hand, it is always possible to add more sensors and variable in later stages. For instance: sensor_ID=3 for “TERRA-MODIS”, and variable_ID=78 for “DMP” (Dry Matter Productivity).
- All these declarations are specific per “project”. For instance, if we would add a new project “BRZ” with Brazilian data in “d:\TUTORIAL\BRZ”, its database settings remain completely independent from the ones of the current project “ETH”.

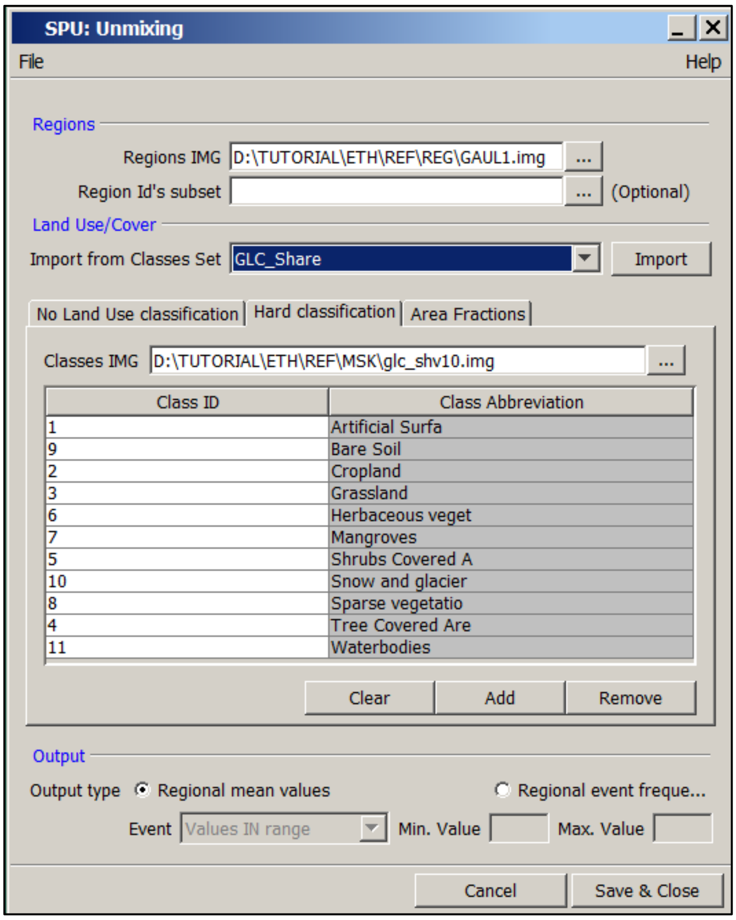
Exercise 5-3 Extraction & Ingestion of the RUM-values

At this moment, the database is ready to receive RUM-values for any (valid) combination of sensors, variables, regions and classes. In this exercise we will extract the RUM-values of the actual NDVI images (variable_ID=1) from SPOT_VGT (sensor_ID=1) for all the GAUL1 Region_IDs (Regions set=1) and unmixed for all the Class_IDs in the GLC_Share map (Classes set=1). At the same moment we will immediately ingest the contents of the RUM-files (generated per dekad) into the database. There are many ways to realize this, but for clarity we will work in three steps.



Definition of a Specifications file (.spu) for GLIMPSE program RUM2IMG*

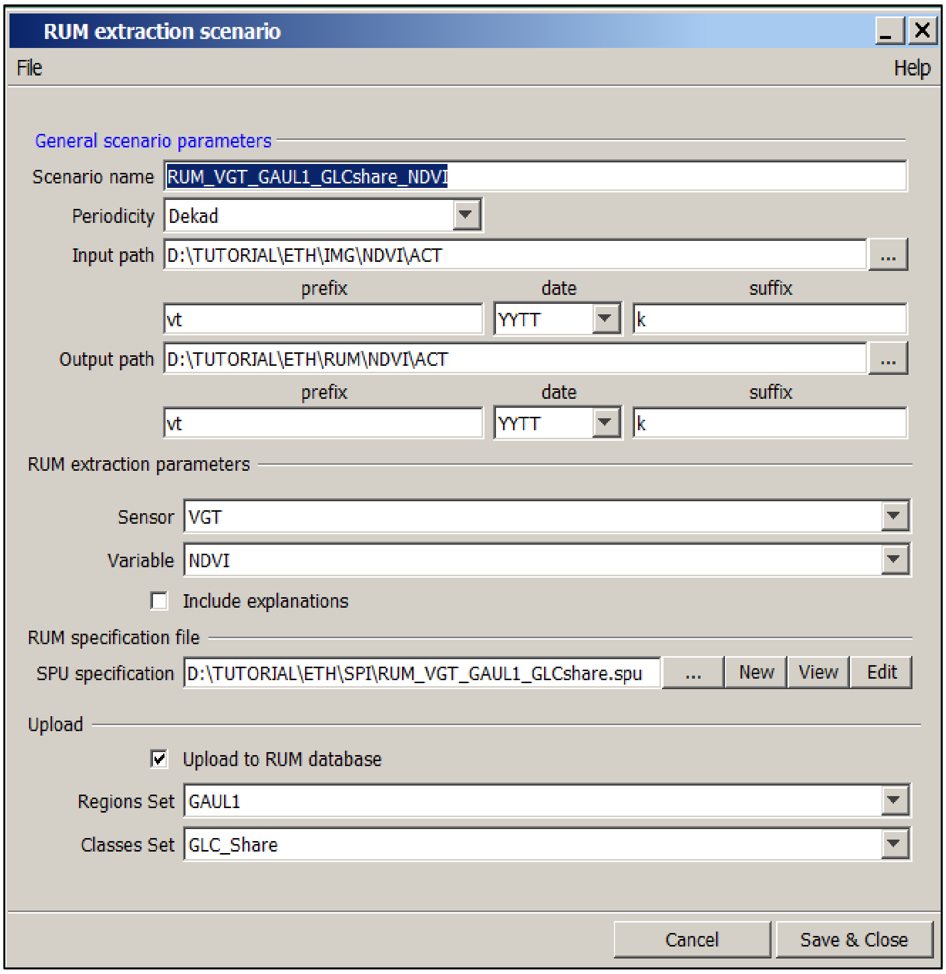
- ✓ Go to **<*Analysis*> <*Database*> <Extract><Specification>.**
- ✓ Set “Regions IMG” to “.\REF\REG\GAUL1.img”.
- ✓ Leave the optional entry “Region IDs subset” blank. If one only wants RUM-values for a subset of regions, these can be listed in a text file (one line per region, ID in first colum) which can be specified here. For instance we could have done that to only retain the Ethiopian provinces. Without this, our RUM-files will also contain information on the GAUL1 regions of the neighbouring countries. See the SPIRITS manual for more information.
- ✓ In the entry “Import from Classes Set”: select our “GLC_Share” and then press the “Import” button on the right. The declared class information (class_IDs and abbreviations) appears in the table below.
- ✓ Select the “Hard classification” tab and specify the “Classes IMG”: ‘.\REF\MSK\GLC_SHv10’.
- ✓ At the bottom of the menu, keep the default “Output type=Regional mean values”. As an alternative, program RUM2IMG can also compute the regional frequency of an “event”, for instance for all pixels with NDVI in the range 0.50 to 0.75. More on this in the SPIRITS manual...
- ✓ When finished, click “Save & Close” and store the settings in file ‘***RUM_VGT_GAUL1_GLCshare.spu***’ (by default under project folder d:\TUTORIAL\ETH\SPI).



Remark: This SPU-file only contains the definitions of the regions and land use classes. In the next steps, it thus can be used to extract RUM-values for any sensor and/or variable.

SCENARIO to extract and ingest the RUM-values of the NDVI time series

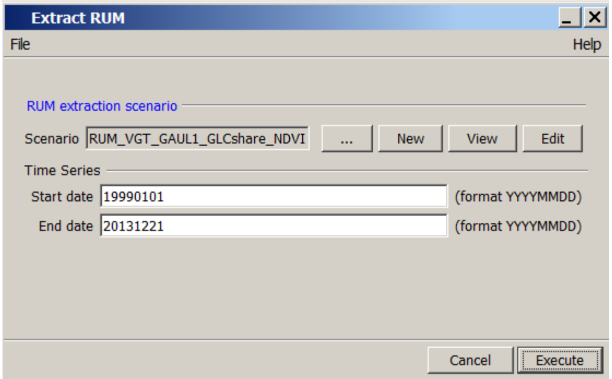
Using the above SPU-file, we now set up a "scenario" which (in the third step) will allow to extract the (smoothed) NDVI from SPOT-VGT – for any time series.



- ✓ Go to ***<Analysis> <Database> <Extract> <Scenario>*** and complete all entries as shown above. These should now appear as evident and straightforward, though with some exceptions discussed below.
- ✓ The scenario must have a name. Call it "***RUM_VGT_GAUL1_GLCshare_NDVI***".
- ✓ For the "RUM specification file", we of course select the one just created.
- ✓ The entry "Include explanations" should be left disabled. When enabled, the generated RUM-file(s) will contain a wide range of textual explanations before the real data. That might be interesting in the testing phase, when applied to a single dekad. But for the application on a time series it would be counterproductive and space-consuming.
- ✓ The entry "**Upload to RUM database" should be enabled**! In this way, the contents of the generated RUM-files are immediately ingested into the database. In the other case, SPIRITS only creates the dekadal RUM-files. Afterwards they can still be ingested via ***<Analysis> <Database> <Fill>***. But it's better to perform both actions together.
- ✓ Click "Save & Close" and store the scenario in '***RUM_VGT_GAUL1_GLCshare_NDVI.SnS***' (by default under project folder d:\TUTORIAL\ETH\SPI).

TASK to extract and ingest the RUM-values of the NDVI time series

Because all preparations have been done before, this third and last step is easy to start in SPIRITS, but on the other hand it takes some time (ca. 20 minutes for the full time series of 15 years or 540 dekads).



- ✓ Go to ***<Analysis> <Database> <Extract> <Time series>*** and complete all entries as shown above.
- ✓ Select the scenario defined in the previous step.
- ✓ Set the start and end dates, so to cover the entire pariod (15 years).
- ✓ Optionally: save this "task" via <File><Save> as: "***d:\TUTORIAL\ETH\SPI\RUM_VGT_GAUL1_GLCshare_NDVI.TnT***'.

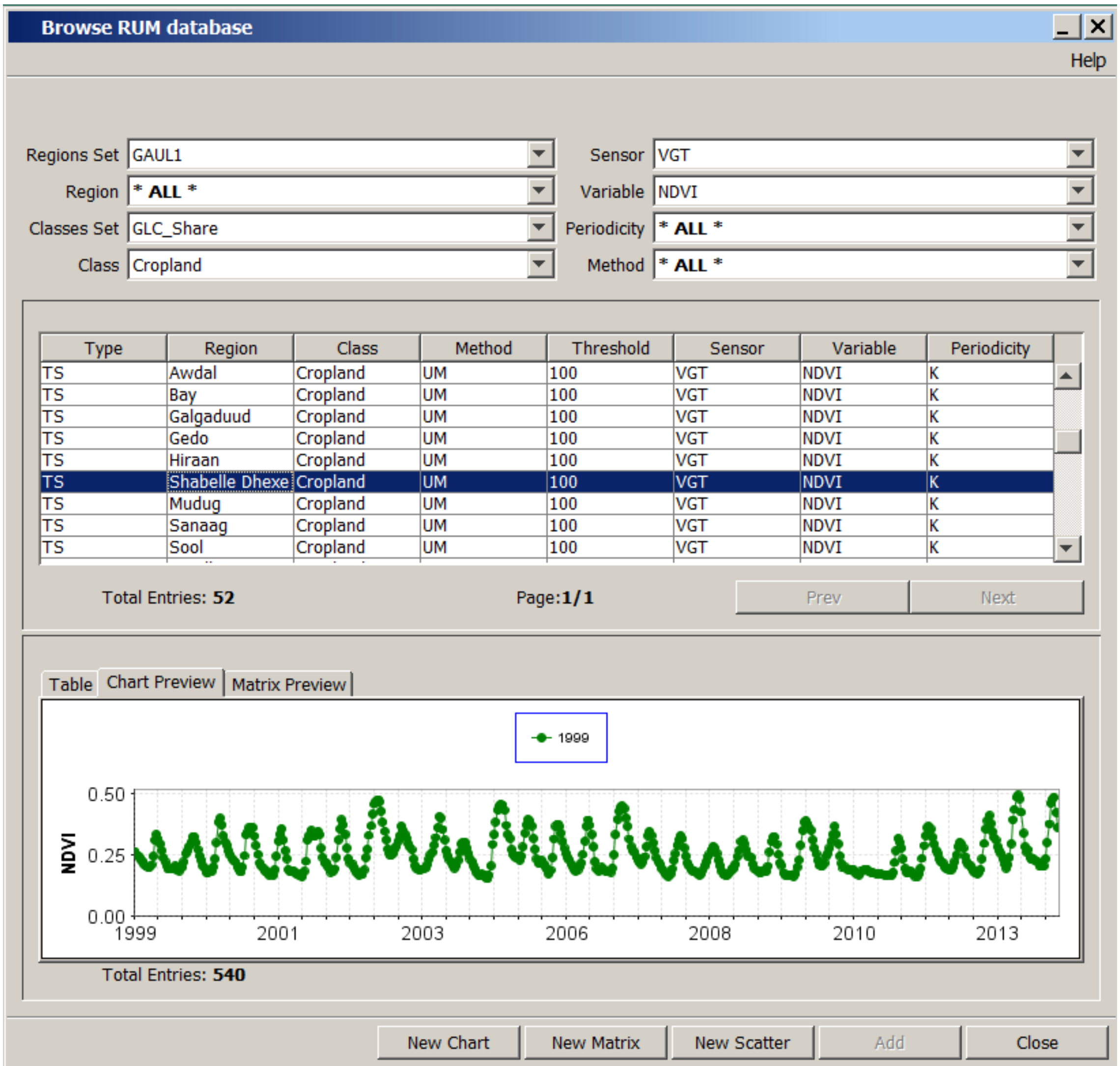
Exercise 5-4 Visualization of statistics

Browsing the RUM database & Selection of a "dataset"

In this context, a "**dataset**" is defined as a time series of RUM-values holding for a specific combination of the following elements (or "labels"):

- A sensor (e.g. "VGT").
- A variable (e.g. "NDVI").
- A region from one of the declared "Regions sets" (e.g. "Tigre" in GAUL1).
- A class from one of the declared "Classes sets" (e.g. "cropland" from GLC-Share).
- An "unmixing method", with three options:
  - OM: Overall regional means without unmixing (class_ID=0).
  - UM: Class-specific unweighted means, derived from a hard classification.
  - WM: Class-specific weighted means, derived from "Area Fraction Images".
- An "Area fraction threshold": only relevant for the WM unmixing method.
- A periodicity: day, dekad, month, year.
- A series "type", with different options:
  - TS: Normal series
  - LTA, MIN, MAX,...: For series related to the Long-Term Statistics. During the ingestion of the RUM-files in the database, these are recognized as such via their specific year (1950-1964).

✓ Goto ***<Analysis> <Database> <Browse>*** to open the "Database browser".



The top panel allows to select one or more "datasets" using the filters for the different labels (regions, classes, sensors, etc.). All the series which meet the set conditions are shown in the central table. By clicking on a specific dataset, its full series is shown in the bottom panel in either of three forms: table, chart or matrix (see the three tabs on top of this lower panel). The above figure shows the chart representation.

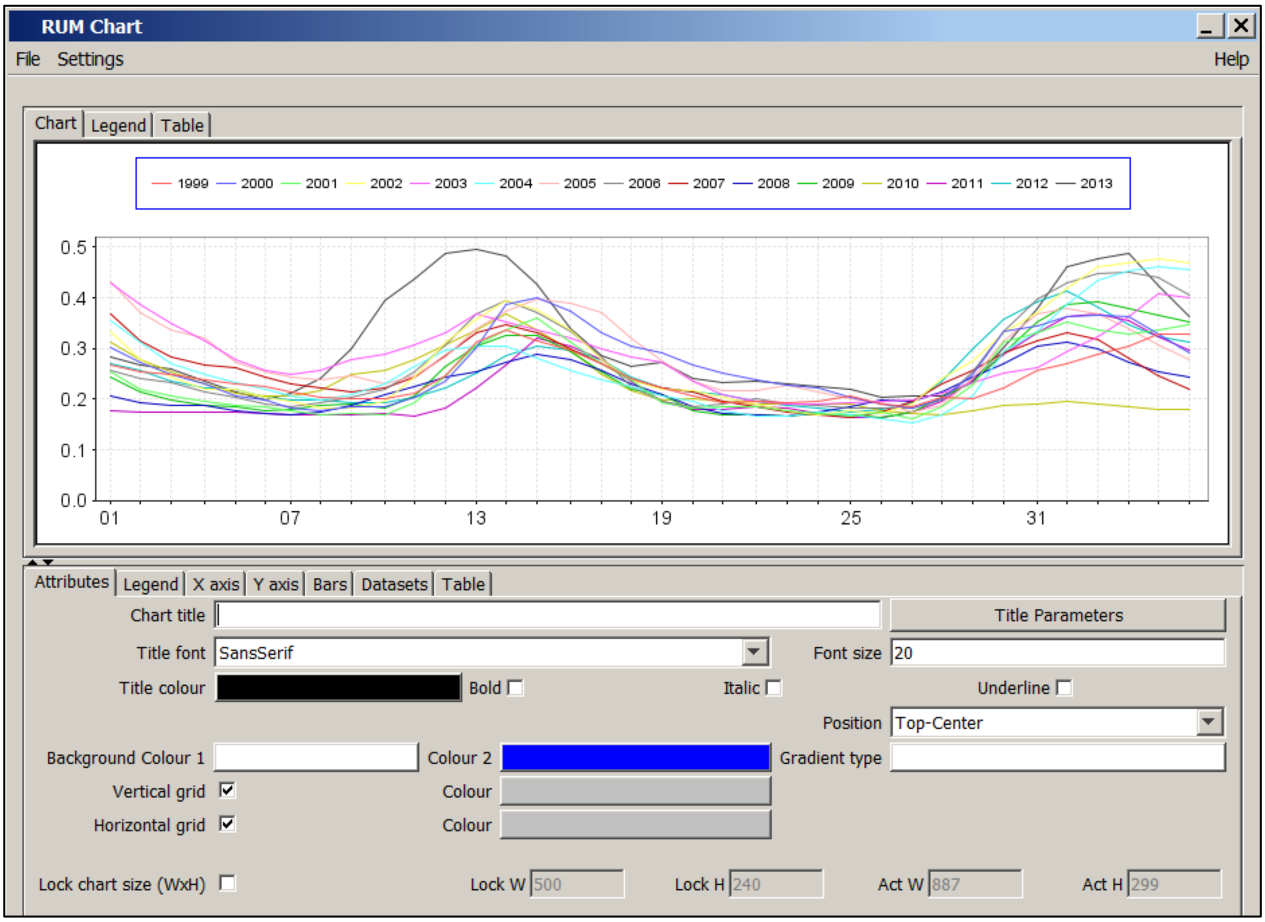
✓ On top of the browser: define the filters. In the central table: select a dataset and observe the effect of switching between the three representation modes in the lower panel.

✓ In this example we selected "Cropland" in the GAUL1 region "Shabelle Dhexe".

✓ Then click on the "**New chart**" button to open the "RUM Chart" user interface.

The “RUM Chart” User Interface: 1. General

This UI initially shows a default graph of the selected “dataset”. We first explain the general concepts.



Below the graph, the UI shows seven different “tabs”:

- ✓ The **“Attributes”** tab allows to modify the background colour of the graph and to add a legend. Suppose we want a title which indicates the names of the region and class. Clicking on ‘**Title Parameters**’ opens a panel with all possible parameters and their meaning. In the field “Chart title” enter “**%19 - %26**”. Also the attributes of the title can be adapted (font, position, etc.).
- ✓ The **“Legend”** tab allows to define the legend. Here too the legend title can comprise any combination of fixed text and parameters (%n). The available ones can be found by clicking on “**Legend parameters**”. By default, the “”legend pattern” is “%1”, which is OK because %1 stands for the year (YYYY). Change the position of the legend to “Right-Center”. If it covers more than one columns, enlarge the graphical window.
- ✓ In the “**X axis**” tab, set the title to “Dekad” and the interval of the “tick labels” to 3.
- ✓ In the “**Y axis**” tab, set the title to “NDVI”.

On top of the graph, there are three more “tabs”:

- ✓ “**Chart**” is the default and shows the graph. Clicking on it with the right mouse button allows to save the graph as a PNG-file.
- ✓ “**Legend**” only shows the legend in more detail, which is handy when working with the legend tab. Here too, the legend can be saved as a seperate PNG using the right mouse button. If many similar graphs of the same nature have to be made (for different regions and/or classes) it’s often more

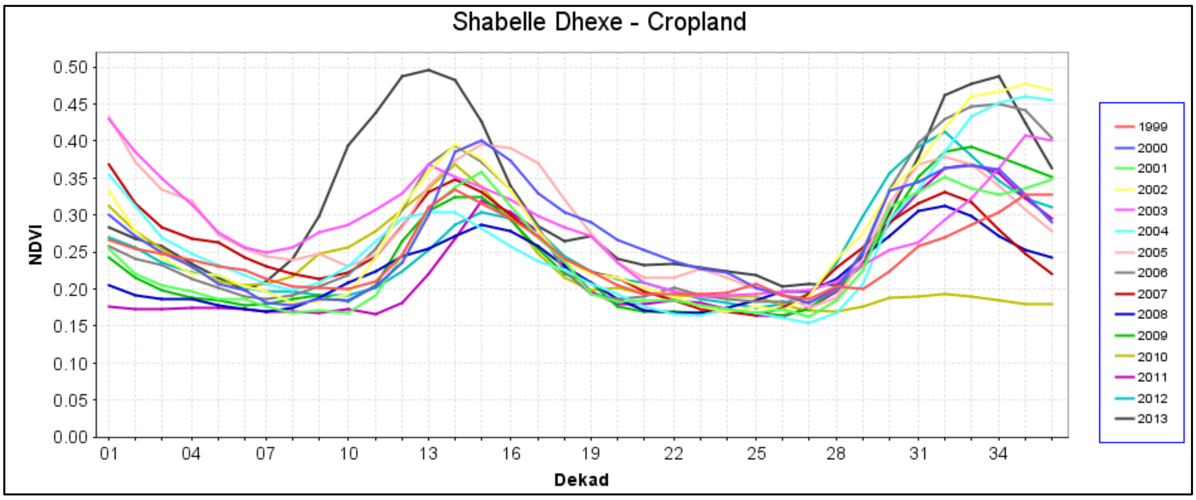
practical to remove the legend from the graphs and to add it to the final document (or website) as a separate figure.

- ✓ "**Table**" shows the full contents of the selected dataset in tabular form. Clicking on it with the right mouse button allows to save it to a text file, which can easily be imported in EXCEL. Notice that this tab is linked with the "Table" tab below the graph. The latter allows to adjust the contents of the table, for instance to eliminate some redundant fields (e.g. in our case sensor and variable are always "VGT" and "NDVI") or to reverse the rows and the colums.

Finally, on top of the UI, there are two last tabs:

- ✓ "**File**" allows to save the current "**chart template**" as a "***.CnC-file**". Remind: according to our project definitions, all such CnC-files are saved in folder 'd:\TUTORIAL\ETH\REF\CNC'. Such CnC-files can later be reopened for application on similar data. The same File menu also allows to save the current graph to a PNG-file.
- ✓ "**Settings**" has two options. "Set defaults" stores the current settings as defaults for all later graphs. "Reset defaults" returns back to the original settings as defined by SPIRITS.

After the above modifications, the graph now looks like this. Save the chart template as '**d:\TUTORIAL\ETH\REF\CNC\T1.cnc**'



The "RUM Chart" User Interface: 2. Operations on datasets (single Y-axis)

The RUM chart can show more than a single "dataset". This can be achieved via the **"datasets" tab** below the graph. At the bottom of this panel there are five buttons:

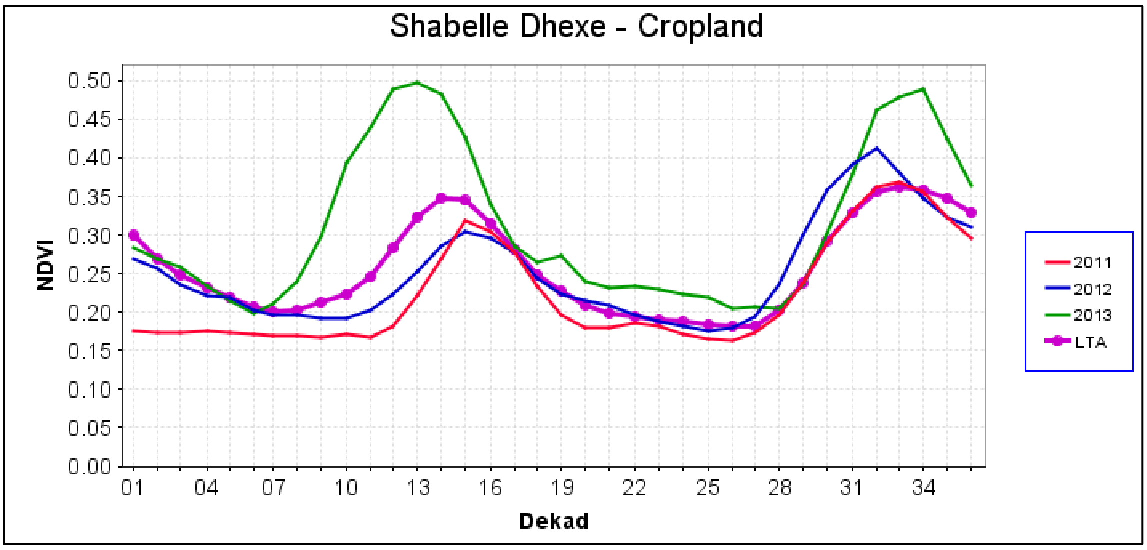
- "**Replace**" again opens the "Database browser" where another series can be selected. For instance if we chose grassland in the Somali region, the data in the current graph are replaced by the ones of this configuration. The new chart title is adapted and becomes "Somali – Grassland".
- "**Copy**" creates a new dataset, a copy of the current one. Further more on this.
- "**Add**" also opens the browser and the selected dataset is added to the chart.
- "**Remove**" withdraws the currently selected dataset.
- "**Clear**" removes all the datasets and the chart becomes empty.

The dataset tab shows the characteristics of each available series via three sub-tabs: “Properties”, “Parameters” and “Series”.

The new dataset obtained via the “Copy” button can be used to perform statistical operations on the original series.

- ✓ Click ‘Copy’ to add the current dataset (Shabelle – Cropland) a second time to the RUM Chart. In the panel, a new entry pops up: “Dataset(2)”.
- ✓ Click on it to activate it, and in the “Parameters” tab change the “Operation” from “Normal” to “Average”. This will compute the LTA (per dekad) over the entire series (15 years), in the same style as done before for the images (per pixel) but now on the RUM-values.
- ✓ In the “Series” tab, adapt the colour, shape and stroke such that this LTA line gets a different aspect than the normal, annual curves. Also: in the right column, set “Description=LTA”.
- ✓ But the graph is too crowded, so we will reduce the number of annual profiles. Activate dataset(1) and in “Series” make all years invisible, except the last three. For these three enter the year (2011, 2012, 2013) in the “Description” boxes at the right. Now the graph becomes much clearer.
- ✓ But the legend is wrong for the LTA line. So in the “Legend” tab, change the “Legend pattern” to “%0”. Now the legend will display the texts (2011, 2011, 2013, LTA) which we provided in the series descripions of both datasets.
- ✓ Save this chart template in ‘**d:\TUTORIAL\ETH\REF\CNC\T2.cnc**’.

The obtained figure now looks like this:



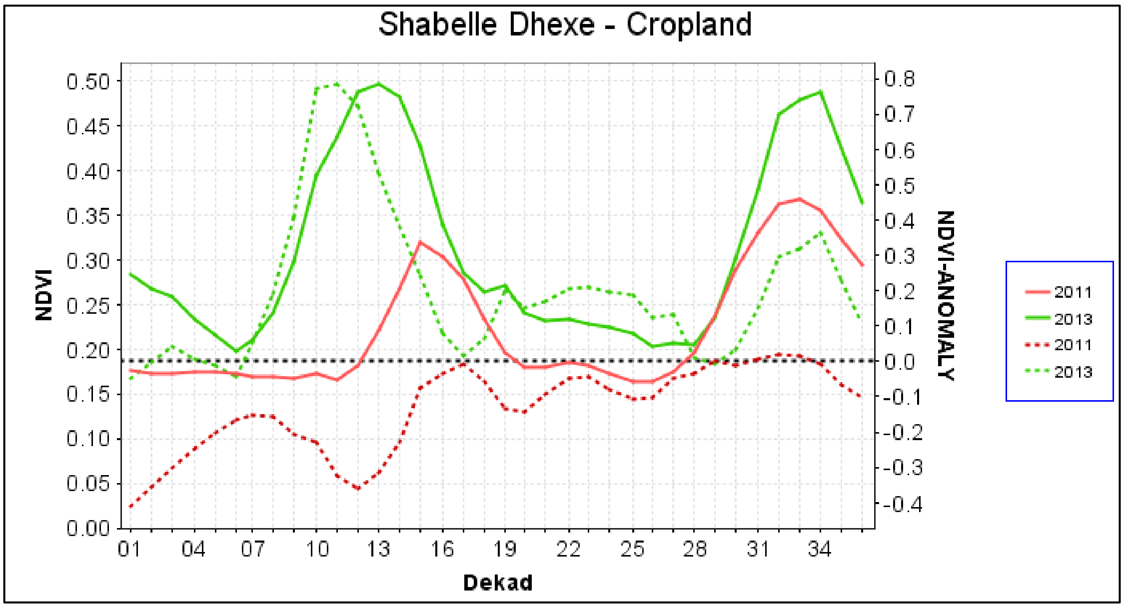
The “Parameters” tab of each dataset has performant functionalities:

- “**Smooth**” allows to apply a running mean filter (of specified length) on the original series.
- “**Operation**” by default is “Normal”, but this option allows to compute a wider range of operations on the RUM series, such as Long-Term Statistics (Historical Mean, Min, Max, SD, etc.) and difference operators (anomalies).
- “**Mode**” by default is “Normal” but it can also be switched to “Cumulative”.

The "RUM Chart" User Interface: 3. Operations on datasets (two Y-axes)

The above procedure thus first made a copy of the original dataset(1). And in this dataset(2), the operation was switched to "Average" (=LTA). The same approach could for instance be repeated to add more datasets with the historical minimum and maximum. These are relatively simple cases because the NDVI Y-axis remains valid for all these profiles.

The figure below has two different Y-axes: NDVI on the left, NDVI-anomaly on the right. It can be created as explained below.



- ✓ Clear the RUM chart and return to the original graph: first reload the dataset of cropland in "Shabelle Dhexe". Then via the File menu, reactivate the template T1.CnC.
- ✓ In this dataset(1): make all years invisible except 2011 and 2013. Colour them in Red (2011) and Green (2013) and add the descriptions "2011-ACT" and "2013-ACT".
- ✓ Copy this series to dataset(2), make this one active and select "operation=Rel.Diff.avg". The new data (anomalies) are added to the graph and a second Y-axis appears on the left.
- ✓ In the tab "Y axis" one now can provide settings for both axes. Select the right one, set its "Axis position" to "Right" and give it the title "NDVI-ANOMALY".
- ✓ In the "Series" tab of dataset(2), give the two years the same colours as for the ACT data but a different stroke (.....). In the desciption fields enter "2011-RD" and "2013-RD".
- ✓ In the "Legend" tab, set "Legend pattern=%0", so the legend displays the manually added descriptions of the series.
- ✓ Finally, in the entry of the second Y-axis, select "Markers" and press "Add". Set "From=0.0", colour=Black and stroke =".....". This will add the black dotted reference line at RD=0.0.

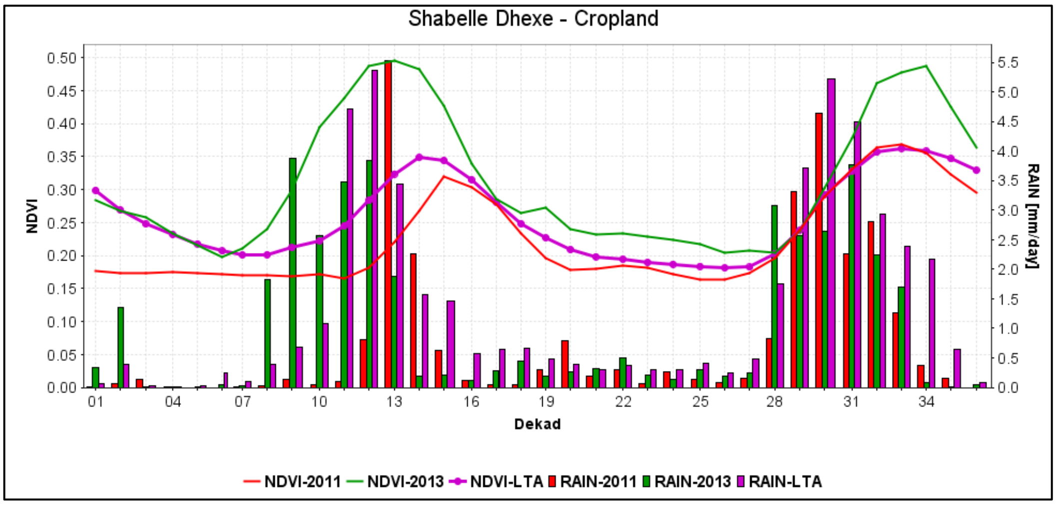
The "RUM Chart" User Interface: 3. Two variables & Bars

In brief words, the figure below was created as follows:

- In the meantime we also extracted the RUM-files for the rainfall series from ECMWF, and the values were ingested in the database. So now we have two independent variables: NDVI and Rain.
- We recalled the situation saved in '**d:\TUTORIAL\ETH\REF\CNC\T2.cnc**', but removed year 2012.

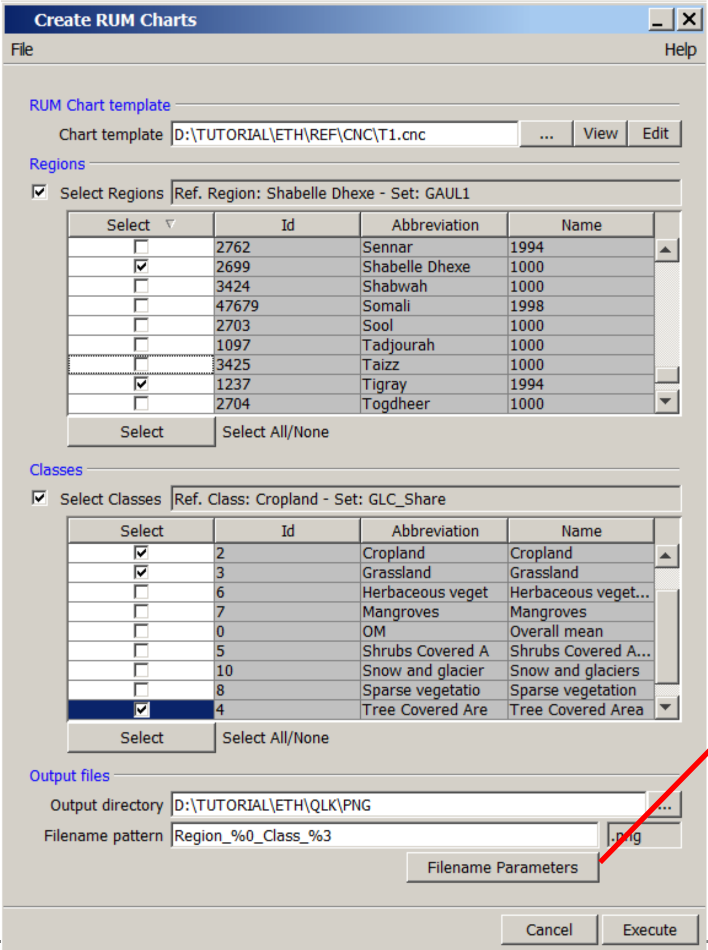
- Then the rainfall series for cropland in “Shabelle Dhexe” was added to the chart. This dataset(3) was copied to dataset(4) in which the LTA of rainfall was stored. The new Y-axis (titled “RAIN [mm/day]”) was moved to the right.
- In the parameters tab of both rainfall datasets, the option “Show as bars” was enabled.
- In the “Bars” tab under the figure, “Equal bar width” must be set to “Series”, and “Invisible series” to “No bar”.

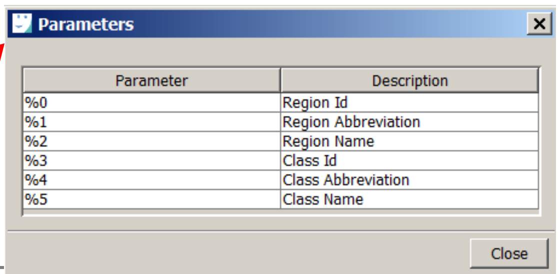
See the SPIRITS Manual for more information.



Generation of a Series of Charts

✓ Goto **<*Analysis*> <*Charts*> <Chart series>** to open the “Create RUM charts” interface.





- ✓ On top: recall one of the chart templates created before, for instance “T1.CnC”.
- ✓ Then make a selection of regions and classes. In the above figure we chose two regions and three classes.
- ✓ We made a new output directory to store the PNG-files to generate: ‘***d:\TUTORIAL\ETH\QLK\PNG***’.
- ✓ The names of the PNG-files are specified via the “Filename pattern”. Clicking on “Filename parameters” displays the list of available %n parameters.
- ✓ Pressing “Execute” automatically generates all the requested PNG-files, in agreement with the settings defined in “T1.CnC”.