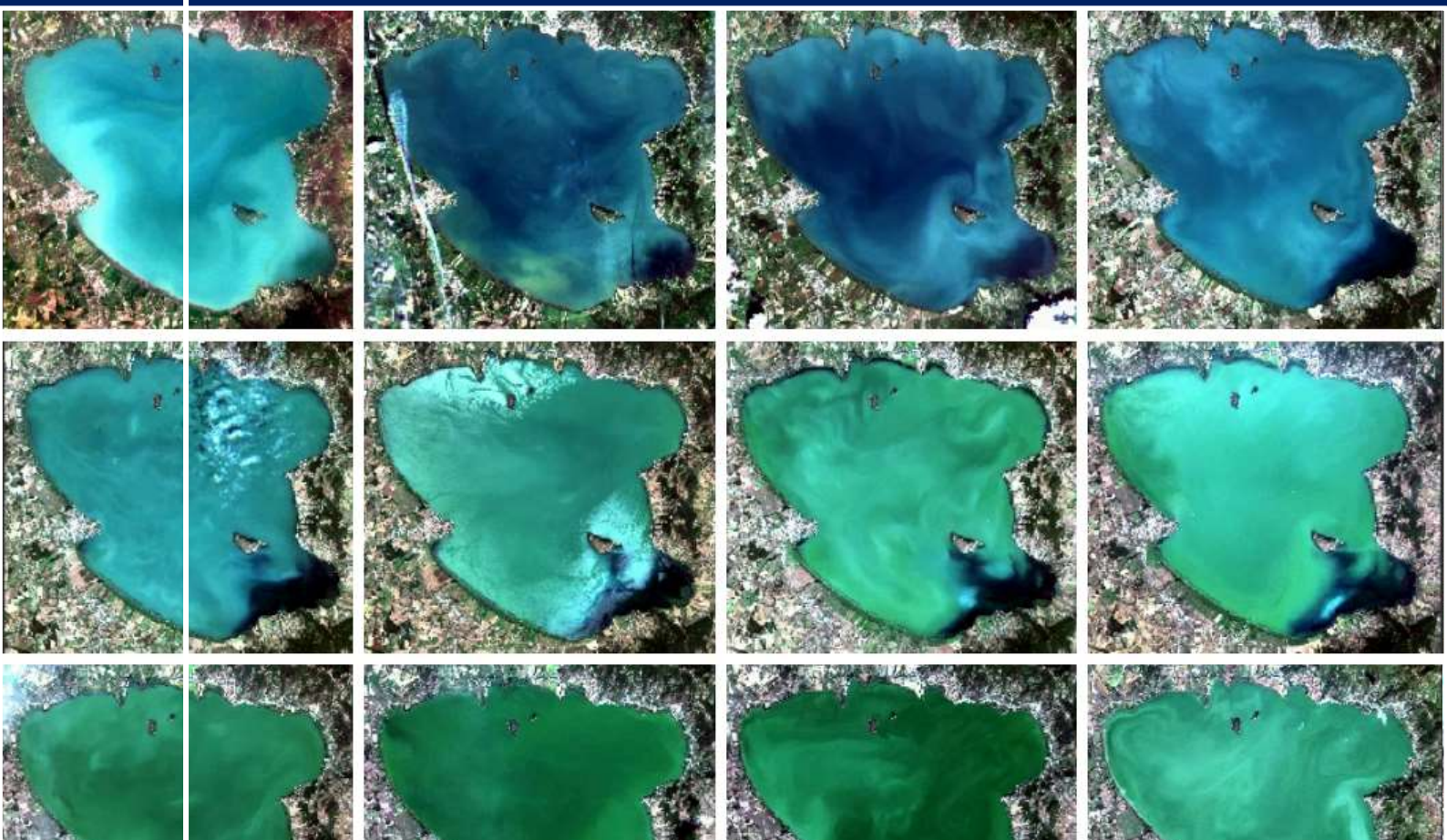




# RUS

Copernicus



## TRAINING KIT – HYDR01

FRESHWATER QUALITY MONITORING FROM SPACE  
LAGO TRASIMENO, ITALY



Research and User Support for Sentinel Core Products

The RUS Service is funded by the European Commission, managed by the European Space Agency and operated by CSSI and its partners.

Authors would be glad to receive your feedback or suggestions and to know how this material was used. Please, contact us on [training@rus-copernicus.eu](mailto:training@rus-copernicus.eu)

Cover images produced by RUS Copernicus

The following training material has been prepared by Serco Italia S.p.A. within the RUS Copernicus project.

Date of publication: February 2020

Version: 1.1

Suggested citation:

Serco Italia SPA (2020). *Freshwater Quality Monitoring from Space - Lago Trasimeno, Italy (version 1.1)*. Retrieved from RUS Lectures at <https://rus-copernicus.eu/portal/the-rus-library/learn-by-yourself/>



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/).

---

#### DISCLAIMER

While every effort has been made to ensure the accuracy of the information contained in this publication, RUS Copernicus does not warrant its accuracy or will, regardless of its or their negligence, assume liability for any foreseeable or unforeseeable use made of this publication. Consequently, such use is at the recipient's own risk on the basis that any use by the recipient constitutes agreement to the terms of this disclaimer. The information contained in this publication does not purport to constitute professional advice.

## Table of Contents

1	Introduction.....	3
2	Training.....	3
2.1	Data used.....	3
2.2	Software in RUS environment.....	3
3	Register to RUS Copernicus.....	4
4	Request a RUS Copernicus Virtual Machine.....	5
5	Step by step.....	8
5.1	Data download – ESA SciHUB.....	8
5.2	Data exploration.....	12
5.3	STEP 1 – Pre-processing.....	13
5.4	STEP 2 – Atmospheric correction & bio-optical model.....	18
5.5	STEP 3 – Empirical algorithms.....	22
6	Extra steps.....	30
6.1	Downloading the outputs from VM.....	30
7	Sources & References.....	30

# 1 Introduction

The Research and User Support for Sentinel core products (RUS) service provides a free and open scalable platform in a powerful computing environment, hosting a suite of open source toolboxes pre-installed on virtual machines, to handle and process data derived from the Copernicus Sentinel satellites constellation.



Lago Trasimeno Credits: Photo by

Freshwater is a fundamental resource for human civilisation, the availability or unavailability of it is directly linked to water security, food security, economic productivity and conflicts. Moreover, freshwater ecosystems play a major role in driving global biodiversity and ecological stability, however they are also exceptionally vulnerable to human mismanagement and climate change. With the progress of climatic change and growing world population, the pressure on our freshwater resources is growing and with it

the need for careful monitoring and management, to preserve them for current and future generations. Remote sensing enables us to monitor a number of water-quality indicators that reflect the state of the entire freshwater ecosystems. These indicators observable from space help us to estimate among others the concentrations of Chlorophyll-a, Total suspended matter (TSM), Coloured dissolved organic matter (CDOM), as well as the density of cyanobacteria and water turbidity.

**For theoretical introduction please watch: <https://youtu.be/Fx01Jtif884> on RUS Copernicus Training channel.**

## 2 Training

Approximate duration of this training session is two hours.

### 2.1 Data used

- 12 low-cloud Sentinel-2A Level 1A tile (Tile ID: T32TQN) acquired from 16 April to 15 November 2018 [downloadable @ <https://scihub.copernicus.eu/>]

S2A\_MSIL1C\_20180416T100031\_N0206\_R122\_T32TQN\_20180416T120852.zip  
S2B\_MSIL1C\_20180511T100029\_N0206\_R122\_T32TQN\_20180511T134849.zip  
S2B\_MSIL1C\_20180531T100029\_N0206\_R122\_T32TQN\_20180531T120906.zip  
S2B\_MSIL1C\_20180630T100029\_N0206\_R122\_T32TQN\_20180630T134105.zip  
S2B\_MSIL1C\_20180710T100029\_N0206\_R122\_T32TQN\_20180710T135302.zip  
S2A\_MSIL1C\_20180718T101031\_N0206\_R022\_T32TQN\_20180718T171829.zip  
S2B\_MSIL1C\_20180809T100019\_N0206\_R122\_T32TQN\_20180809T140850.zip  
S2B\_MSIL1C\_20180822T101019\_N0206\_R022\_T32TQN\_20180822T142412.zip  
S2B\_MSIL1C\_20180829T100019\_N0206\_R122\_T32TQN\_20180829T190727.zip  
S2A\_MSIL1C\_20180906T101021\_N0206\_R022\_T32TQN\_20180906T122259.zip  
S2B\_MSIL1C\_20180928T100019\_N0206\_R122\_T32TQN\_20180928T170858.zip  
S2A\_MSIL1C\_20181115T101251\_N0207\_R022\_T32TQN\_20181115T103245.zip

### 2.2 Software in RUS environment

Internet browser, SNAP + Sentinel-2 Toolbox, QGIS, (Extra steps: Sen2Cor, Google Earth)



### 3 Register to RUS Copernicus

To repeat the exercise using a RUS Copernicus Virtual Machine (VM), you will first have to register as a RUS user. For that, go to the RUS Copernicus website ([www.rus-copernicus.eu](http://www.rus-copernicus.eu)) and click on **Login/Register** in the upper right corner.



Select the option **Create my Copernicus SSO account** and then fill in ALL the fields on the **Copernicus Users' Single Sign-On Registration**. Click **Register**.



Within a few minutes you will receive an e-mail with activation link. Follow the instructions in the e-mail to activate your account.

You can now return to <https://rus-copernicus.eu/>, click on **Login/Register**, choose **Login** and enter your chosen credentials.

### Login / Register

The registration system to access the RUS service platform has moved toward the COPERNICUS Single sign On authentication server.

- New Users who have not yet registered to the RUS portal shall first create a COPERNICUS SSO account.

Note that your Copernicus SSO account will be activated only after the reception of the first email sent by the Copernicus service. We advise you to consult [this document](#) and [this video](#) to facilitate your registration procedure.

**REGISTER COPERNICUS SSO account**

Users who already have a COPERNICUS SSO account can login here:

**Login**

Close

### Credentials

CDS-SSO ID

Password

Max Idle Time

Max Session Time

**Login** **Reset**

[Forgot your password?](#)

Upon your first login you will need to enter some details. You must fill all the fields.

### Do you want to subscribe for a new RUS account?

Your ESA SSO subscription data:

First Name

Last Name

Email

Organization

Country

Additional subscription information

Please complete the following information:

Where did you learn about the RUS service?

Select one or more options:

workshop event  
colloquium  
conference  
social media  
other

Workshop event

Workshop number

Webinar

Title

**Subscribe** **Cancel**

## 4 Request a RUS Copernicus Virtual Machine

Once you are registered as a RUS user, you can request a RUS Virtual Machine to repeat this exercise or work on your own projects using Copernicus data. For that, log in and click on **Your RUS Service** → **Your Dashboard**.

### RUS Research and User Support

The RUS Service \* The RUS Offer \* The RUS Library \* The RUS Community \* **Your RUS service**

**Your RUS service**

This section gathers pages related to your RUS services:

- Your profile:** displays your personal information linked to your ESA SSO and RUS accounts.
- Your dashboard:** allows you to access your private dashboard.
- Your training:** allows you to register to a training session you have been invited to participate in.

**News from RUS**

One year on!

Copernicus Info Session - Reykjavik - 19 September 2018

SPIE Remote Sensing 2018 - Berlin (Germany) - 11-12 September 2018

SIWI World Water Week 2018 - Stockholm - 26-31 August 2018

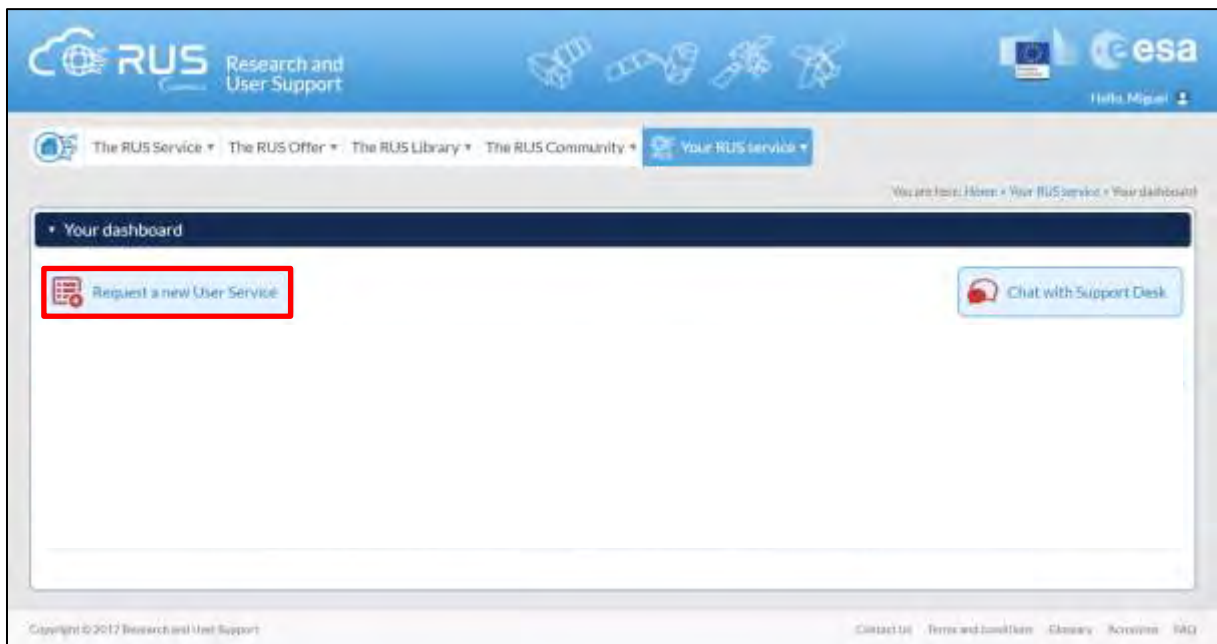
MedRIP Kick-off Meeting - Charla - 13 & 14 July 2018

RUS Webinar - Special edition "AskRUS - Sentinel-1" - 12 July 2018

RUS Training Session - Valencia - 22 July 2018

ICARSS 2018 - Valencia - 22-27 July 2018

Click on **Request a new User Service** to request your RUS Virtual Machine. Complete the form so that the appropriate cloud environment can be assigned according to your needs.



If you want to repeat this tutorial (or any previous one) select the one(s) of your interest in the appropriate field.

The image shows the 'User Support Request' form, Step 1/3: Your experience. The form asks for user experience details and tutorial exercise preferences. A red rectangle highlights the section for selecting tutorial exercises. The form includes the following fields:

- How many years of experience in Remote Sensing do you have? (Dropdown menu)
- Have you already downloaded Copernicus data via the Copernicus Open access hubs? (Radio buttons: Yes, No)
- Have you already handled/processed Copernicus data? (Radio buttons: Yes, No)
- Do you wish to practice a tutorial exercise shown in a RUS webinar? If yes, please select your choice (hold down CTRL key for multiple selections). This section is highlighted with a red rectangle and contains a list of exercises: HAZARD1 - Flood Mapping in Malawi, HAZARD2 - Burned Area Mapping in Portugal, HYDRO1 - Water Bodies Mapping over Northern Poland, LAND01 - Crop Mapping in Seville, LAND04 - Land Monitoring in Cyprus, OCEAN1 - Ship Detection in Gulf of Trieste.
- If you wish to request another tutorial exercise that doesn't appear in the above list, please type here its name or code. Note that you can request multiple tutorial exercises.

At the bottom of the form, there are 'Cancel' and 'Next' buttons.

Complete the remaining steps, check the terms and conditions of the RUS Service and submit your request once you are finished.

**User Support Request**

Summary information on your request:

This is a collection of information selected across the USR forms. You can go back and edit this information if necessary.

General information on your request:

Years of experience in Remote Sensing: 5-10 years  
 Downloaded Copernicus data?: ☒  
 Handled/processed Copernicus data?: ☒  
 Webinar code: HAZA02\_LAND004

About your RUS project:

Thematic area: Cryosphere (ice and snow)  
 Operations to perform on RUS: Algorithm development  
 Preference for downloading process: Self-downloading  
 Foreseen activities and support needs: Develop a land cover classification  
 Project name: RUS\_Project1

Earth Observation Data Information:

Type of Earth Observation Data: Sentinel-1  
 S1-Product type: S1-Product 1  
 S1-Sensor mode: GRD  
 S1-Polarisation: -  
 S1-Orbit direction: -  
 Sentinel-2: ☒  
 Sentinel-3: ☒  
 Other: ☒  
 I don't know: ☒  
 Region of interest:  
 Min Latitude: 39.5300  
 Max Latitude: 40.5677  
 Min Longitude: -4.6736  
 Max Longitude: -3.7206  
 Reference polygon:

Data acquisition date(s):  
 None  
 Additional data specifications:

☒ I have read and agree to the Terms and conditions of RUS Service.

[Back and edit](#) [Submit the request](#)

Further to the acceptance of your request by the RUS Helpdesk, you will receive a notification email with all the details about your Virtual Machine. To access it, go to **Your RUS Service** → **Your Dashboard** and click on **Access my Virtual Machine**.

**RUS Research and User Support**

Logo: RUS Research and User Support

Navigation: The RUS Service, The RUS Offer, The RUS Library, The RUS Community, **Your RUS service**

Header: Hello, Miguel

Sub-header: You are here: Home > Your RUS Service > Your dashboard

**Your dashboard**

Buttons: Request a new User Service, Chat with Support Desk

Project Name	ID	Date of submission	Status	Actions			Virtual Environment	
RUS_training1	230	2017-08-31	Open	Follow my project	Get support	Close my service	<b>Access my Virtual Machine(s)</b>	Access my CPU monitoring dashboard
				Leave my dashboard	Get a webinar kit	Rate my service	Erase my Virtual Machine(s)	Report a technical incident

Footer: Copyright © 2017 Research and User Support | Contact Us | Terms and Conditions | Glossary | Accessible | FAQ



Fill in the login credentials that have been provided to you by the RUS Helpdesk via email to access your RUS Copernicus Virtual Machine.



This is the remote desktop of your Virtual Machine.



## 5 Step by step

### 5.1 Data download – ESA SciHUB

In this step we will download a Sentinel-2 scene from the Copernicus Open Access Hub using the online interface (**Applications** → **Network** → **Web Browser** or click the link below).

Go to <https://scihub.copernicus.eu/>



Go to “**Open HUB**”, if you do not have an account please register by going to “**Sign-up**” in the LOGIN menu in the upper right corner.



After you have filled in the registration form, you will receive an activation link by e-mail. Once your account is activated or if you already have an account, “**LOGIN**”.



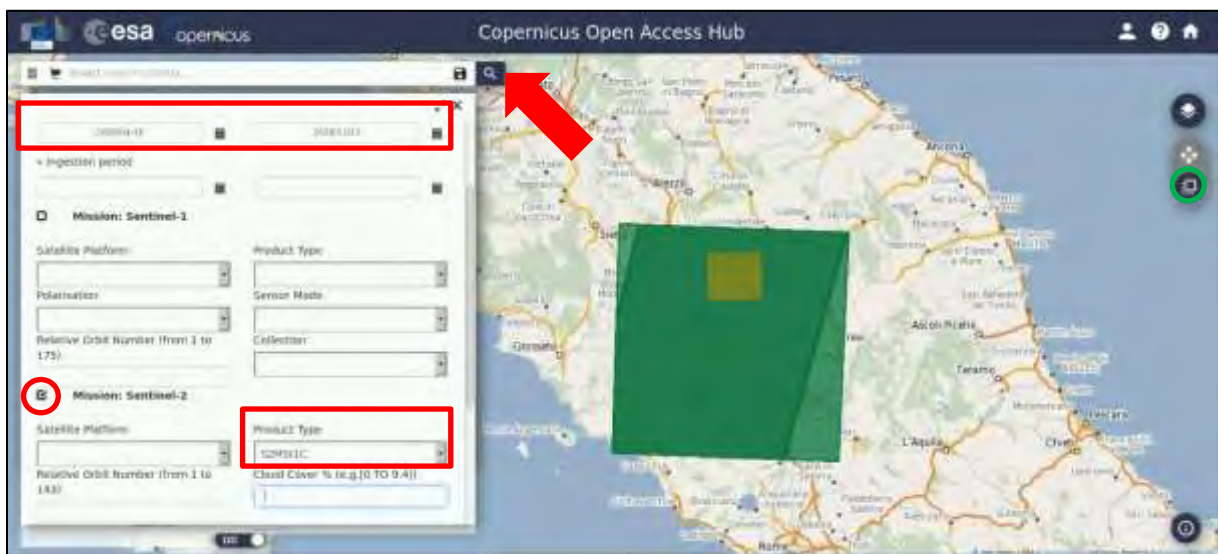
Switch the rectangle-drawing mode to pan mode by clicking on the “**Pan**” icon in the lower left corner of the map (**Green arrow**) and navigate over Portugal (**approximate area – blue rectangle**).




Switch to drawing mode and draw a search rectangle over the lake approximately as indicated below (yellow). Open the search menu (red arrow) and specify the following parameters:

**Sensing period:** From 2018/04/16 to 2018/11/15  
**Check Mission:** Sentinel-2  
**Product Type:** S2MSIL1C

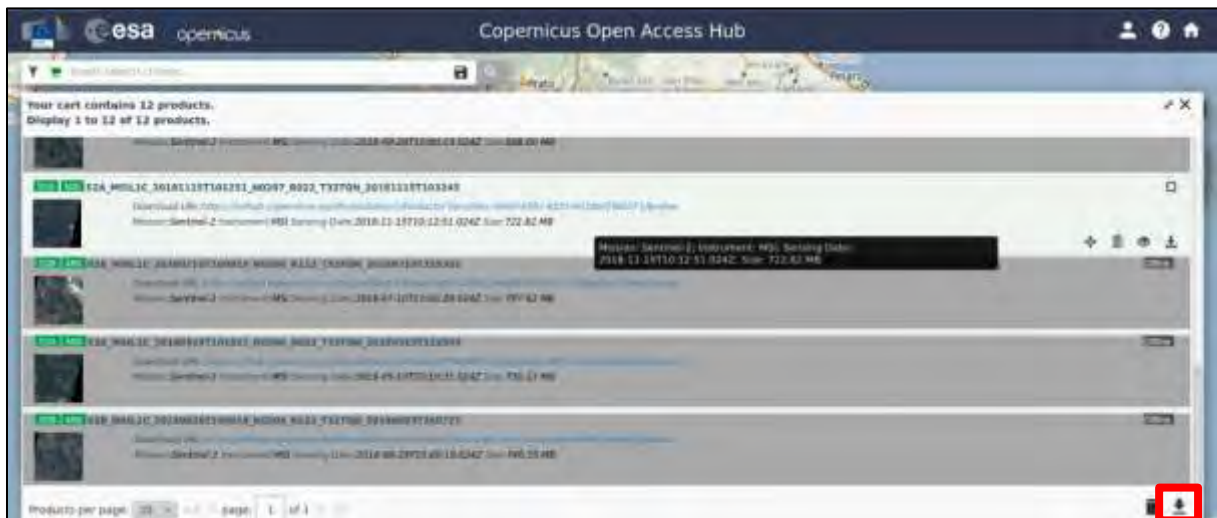
Press “Search” (red arrow below).



In our case, the search returns four results depending on the exact search area defined. Using the  icon, import only the following products to **Cart**:

S2A\_MSIL1C\_20180416T100031\_N0206\_R122\_T32TQN\_20180416T120852.zip  
 S2B\_MSIL1C\_20180511T100029\_N0206\_R122\_T32TQN\_20180511T134849.zip  
 S2B\_MSIL1C\_20180531T100029\_N0206\_R122\_T32TQN\_20180531T120906.zip  
 S2B\_MSIL1C\_20180630T100029\_N0206\_R122\_T32TQN\_20180630T134105.zip  
 S2B\_MSIL1C\_20180710T100029\_N0206\_R122\_T32TQN\_20180710T135302.zip  
 S2A\_MSIL1C\_20180718T101031\_N0206\_R022\_T32TQN\_20180718T171829.zip  
 S2B\_MSIL1C\_20180809T100019\_N0206\_R122\_T32TQN\_20180809T140850.zip  
 S2B\_MSIL1C\_20180822T101019\_N0206\_R022\_T32TQN\_20180822T142412.zip  
 S2B\_MSIL1C\_20180829T100019\_N0206\_R122\_T32TQN\_20180829T190727.zip  
 S2A\_MSIL1C\_20180906T101021\_N0206\_R022\_T32TQN\_20180906T122259.zip  
 S2B\_MSIL1C\_20180928T100019\_N0206\_R122\_T32TQN\_20180928T170858.zip  
 S2A\_MSIL1C\_20181115T101251\_N0207\_R022\_T32TQN\_20181115T103245.zip

Then click on the Profile icon in the upper left corner (marked with **green circle** above) and go to **Cart**. You should now have six products in your cart. Click on the arrow to **Download the Cart**.




The grey products are currently placed in an Offline archive: to learn how to request them please follow the steps outlined here: <https://scihub.copernicus.eu/userguide/LongTermArchive>



TIP: Alternatively, you can try to retrieve the products from other repositories such as: PEPS (<https://peps.cnes.fr/rocket/#/home>) or ONDA DIAS Catalogue (<https://catalogue.onda-dias.eu/catalogue/>) or others. Registering for a free account is usually necessary, but archived data retrieval will be faster than with Open Access Hub.

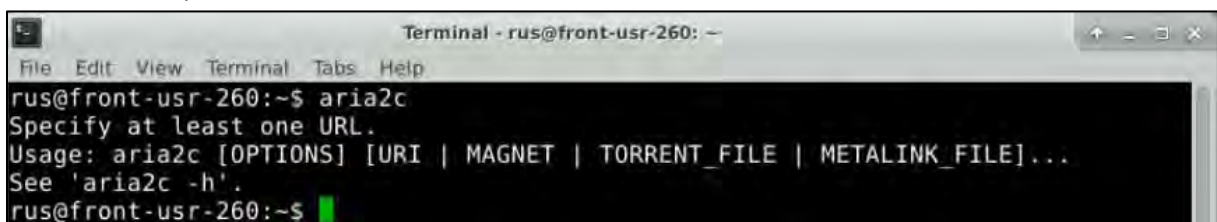
A **products.meta4** file will be downloaded to your **/home/rus** folder. To download our data, we will use **aria2** tool. To use the tool, we first need to place the **products.meta4** file to the folder where we wish our data to be downloaded to:


**/shared/Training/HYDR02\_FreshWaterQuality\_Trasimeno/Original**


First let's test our aria2 installation. To do this we open the Command Line  (in the bottom of your desktop window) and type:

```
aria2c
```

The correct response should be as follows:



 **NOTE 1:** If (**and only if**) the response is “-bash aria2c: command not found”. Then we have to install the tool, to do this in command line type: `sudo apt-get install aria2`  
When requested type: `Y`  
Then you can test your installation again.

If the response is “-bash aria2c: command not found” see  **NOTE 1**. If you have received the correct response, then we can run the tool by typing following commands in the command line (replace **<username>** and **<password>** with your login credentials for Copernicus Open Access Hub):





```
cd /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Original

aria2c      --http-user='<username>'      --http-passwd='<password>'      --check-
certificate=false --max-concurrent-downloads=2 -M products.meta4
```

The first line changes our directory to the target directory. The second line runs the download tool (Type the red text all in single line). All twelve products will be downloaded to the **Original** folder two products in parallel automatically.

(Note that the constraint of maximum 2 parallel downloads at a time is imposed by the Copernicus Access Hub, if you increase the number, your download will fail). This might take some time.

## 5.2 Data exploration

Launch SNAP (icon on desktop ). When the SNAP window opens click **Open product**  and navigate to: **/shared/Training/HYDR02\_FreshWaterQuality\_Trasimeno/Original/**

Open the first S2 product (from 16 April 2018). We can first investigate the structure of the Sentinel-2 Level 1C products. Click on the dot next to the product name to expand the structure. The L1C products contain (among others):

- 13 TOA (top-of-atmosphere) reflectance bands
- Quality flags

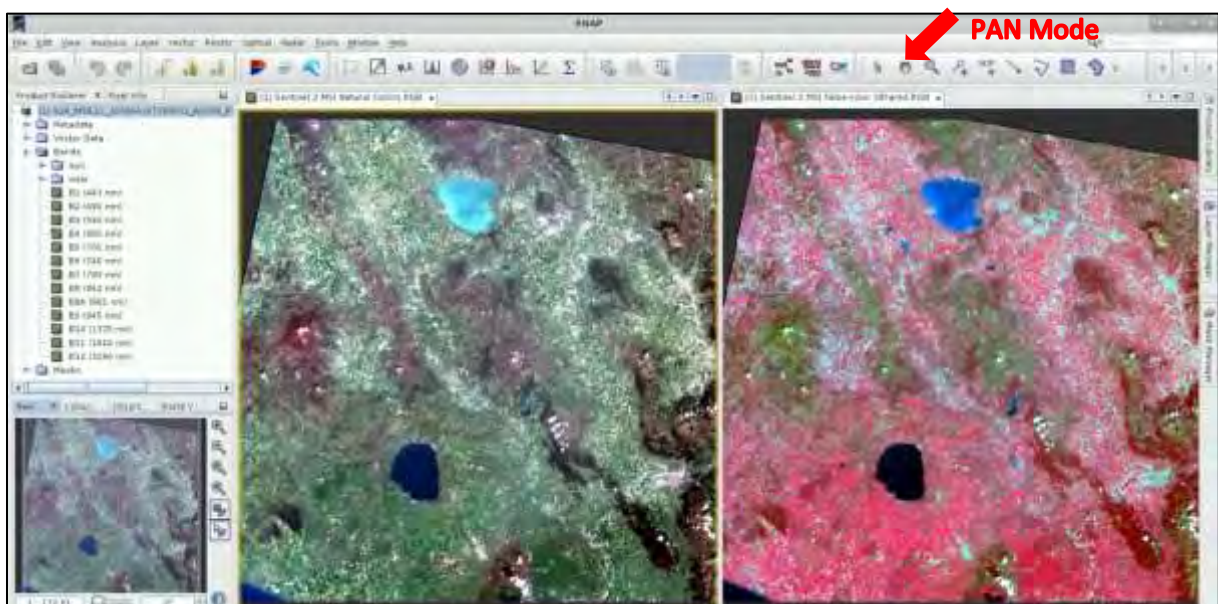
Now, let's visualize the product. We will visualize it in true (natural) colours and as a false color composite which is better for distinguishing water surfaces. **Right-click** the product and click **Open RGB image window**, a new window will open. From the drop-down menu select:



**Profile : Sentinel 2 MSI Natural Colors**  
Click **OK**.

When the image opens, repeat the same but choose the false colour Profile.

**Profile : Sentinel 2 MSI False-color Infrared**  
Finally, go to **Window → Tile Horizontally**.





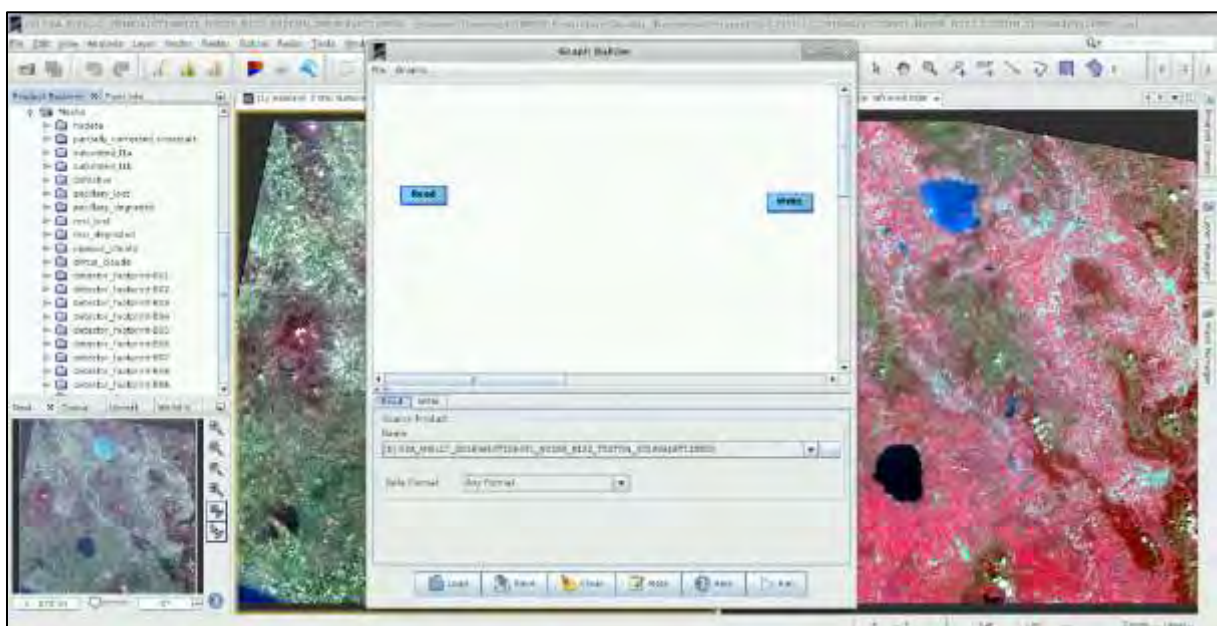
- Level-1C – Top-Of-Atmosphere reflectances, systematically generated since the start of the mission.
- Level-2A – Bottom-Of-Atmosphere reflectances (atmospherically and radiometrically corrected), systematically produced for products acquired over Europe since the spring of 2017, the coverage has increased through 2018 to reach global coverage in the beginning of 2019. Can be produced on user side by applying the Sen2Cor algorithm.

### 5.3 STEP 1 – Pre-processing


Processing the data one by one would not be very time effective. However, we can use either the **Batch Processing** tool available in SNAP or the GPT tool (command line) to process all images automatically.

To use either method, we first need to define the process we want to apply and all its steps. We can do this using the **GraphBuilder** tool. Another advantage of the **GraphBuilder** is that only the final product will be physically saved, and we save valuable disk space.

So, let's build our graph. First, in the **Product Explorer** select the first loaded product (so it is highlighted), then go to **Tools → GraphBuilder**.

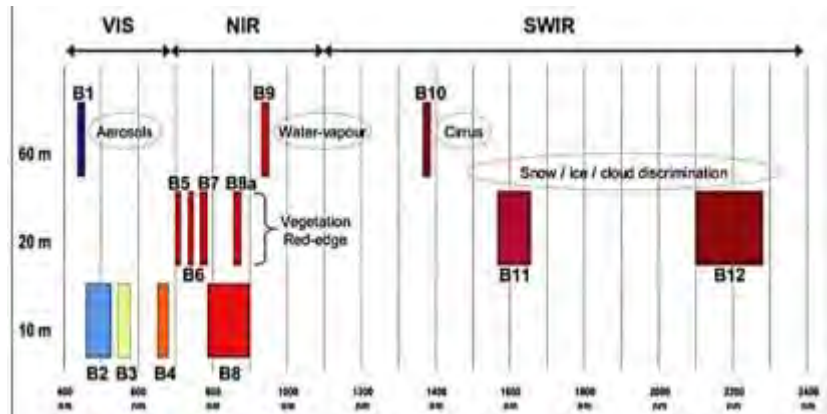


At the moment, the graph only has two operators: **Read** (to read the input) and **Write** (to write the output).

The 13 bands in Sentinel-2 products do not all have the same resolution (therefore size) as mentioned in  NOTE 3. Many operators do not support products with bands of different sizes so first we need to resample the bands to equal resolution.

To add the appropriate operator, right-click the white space between existing operators and go to **Add → Raster → Geometric → Resample**.

NOTE 3: The input product contains 13 spectral bands in three different spatial resolutions (The surface area measured on the ground and represented by an individual pixel). When we open the RGB view all our input bands have 20 m resolution, however, the view is displayed in the full 10 m resolution.

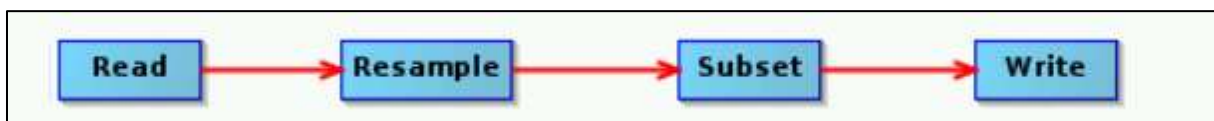


Credits: ESA 2015

A new operator rectangle appears in our graph and new tab appears below. Now connect the new **Resample** operator with the **Read** operator by clicking to the right side of the **Read** operator and dragging the red arrow towards the **Resample** operator.



Next step will be to subset the images to the area of interest, we do this by right-clicking the white space somewhere right of the resample operator and going to **Add → Raster → Geometric → Subset**. Connect the **Subset** operator with the **Resample** operator.



Finally, connect the **Subset** operator to the **Write** operator.

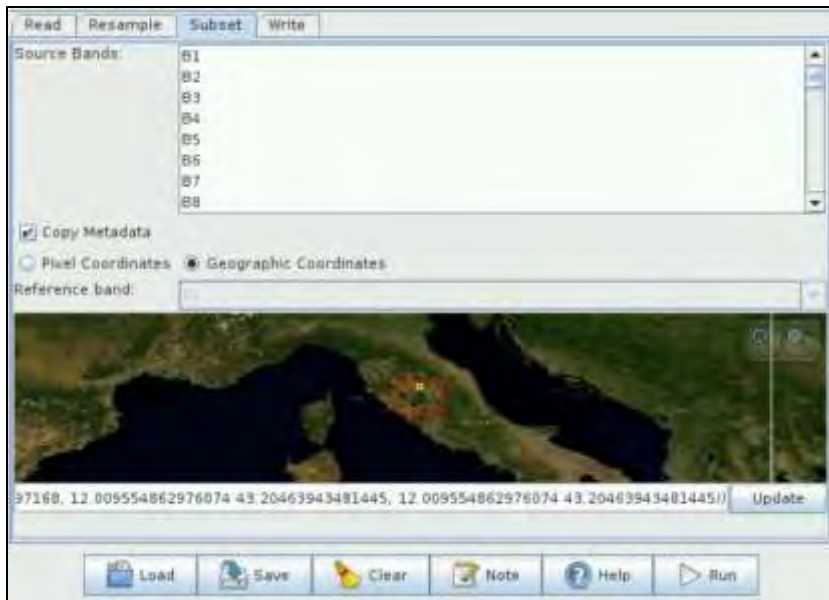
Now, let's set the parameters. In **Resample** tab, we set:

Under **Define size of resampled product**: Select the "By reference band from source product" and choose the **B2** band (we will resample all the bands to 10m resolution).

In the **Subset** tab, select **Geographic Coordinates**, in the text-window paste the subset polygon coordinates in Well-Known-Text format (WKT) from *Expressions\_WQ.txt* file in:

*/shared/Training/HYDR02\_FreshWaterQuality\_Trasimeno/*


```
POLYGON ((12.009554862976 43.20463943481, 12.198662757873 43.20463943481,
12.198662757873 43.0686149597, 12.009554862976 43.0686149597, 12.009554862976
43.20463943481, 12.009554862976 43.20463943481))
```




We can leave the **Read** and **Write** tabs unchanged. Now, save the graph as *STEP1\_Graph\_Prep.xml* to: */shared/Training/HYDR02\_FreshWaterQuality\_Trasimeno/Processing/* by clicking **Save** at the bottom of the window and then close the **GraphBuilder** window.

In this tutorial, we will use the GPT and shell script to automatize the processing of our 12 products. To do this, we must first edit the saved graph file.

To edit the file, we leave SNAP and in the file explorer, navigate to the *.../Processing* folder and right-click on the saved graph file. Select **Open with -> Open with "Mousepad"**. When the file is open, go to **View -> Line Numbers** for easier navigation.

Now, we need to replace the input and output file path with a pattern that will allow the **shell script** (See  NOTE 4) to recognize and replace it with appropriate value for each file automatically.

 **NOTE 4:** A shell script is a computer program designed to be run by the Unix shell, a command-line interpreter. The various dialects of shell scripts are considered to be scripting languages. The one used in this exercise is called Bash script ([https://en.wikipedia.org/wiki/Bash\\_\(Unix\\_shell\)](https://en.wikipedia.org/wiki/Bash_(Unix_shell))). Typical operations performed by shell scripts include file manipulation, program execution, and printing text. If you wish to run this exercise on Windows, you can replicate the steps in any programming language you are familiar with, such as Python or R.

In line 7 (in the node with id='Read') replace the path to input file with (between <file> and </file>): *\$Input* ("**\$**" signifies that what follows is a parameter).



```

STEP1_Graph_preprocess_temp.xml - Mousepad
1 <graph id="Graph">
2 <version>1.0</version>
3 <node id="Read">
4 <operator>Read</operator>
5 <sources/>
6 <parameters class="com.bc.ceres.binding.dom.XppDomElement">
7 <file>/home/rus/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Original/
  S2A_MSIL1C_20180416T100031_N0200_R122_T32T0N_20180416T120052.zip</file>
8 </parameters>
9 </node>

STEP1_Graph_preprocess_temp.xml - Mousepad
1 <graph id="Graph">
2 <version>1.0</version>
3 <node id="Read">
4 <operator>Read</operator>
5 <sources/>
6 <parameters class="com.bc.ceres.binding.dom.XppDomElement">
7 <file>$Input</file>
8 </parameters>
9 </node>

```

In line 51 (in the node with id='Write') replace the path to output file with (between <file> and </file>): `$Output`

```

45 <node id="Write">
46 <operator>Write</operator>
47 <sources>
48 <sourceProduct refId="Subset"/>
49 </sources>
50 <parameters class="com.bc.ceres.binding.dom.XppDomElement">
51 <file>/home/rus/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/
  Subset_S2A_MSIL1C_20180416T100031_N0200_R122_T32T0N_20180416T120052_resampled.dim</file>
52 <formatName>BEAM-DINAP</formatName>
53 </parameters>
54 </node>

45 <node id="Write">
46 <operator>Write</operator>
47 <sources>
48 <sourceProduct refId="Subset"/>
49 </sources>
50 <parameters class="com.bc.ceres.binding.dom.XppDomElement">
51 <file>$Output</file>
52 <formatName>BEAM-DINAP</formatName>
53 </parameters>
54 </node>

```

Now, go to the `.../Processing/Code/` folder and open the `STEP1_script_gpt.sh`

```

STEP1_script_gpt.sh - Mousepad
1 #!/bin/bash
2
3 output_folder=/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Pre-processed/
4 nameStart=Subset_S2_MSIL2A_
5 nameEnd=.dim
6
7 for i in $(ls -d -l /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Original/*.zip)
8 do
9     product_basename=${basename $i}
10    ac_date=$(echo $product_basename | cut -d '-' -f 3)
11    output_pathname=$output_folder$nameStart$ac_date$nameEnd
12    gpt /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/STEP1_Graph_Prep.xml -Pinput=$i -#Output=$output_pathname
13 done
14

```

On line **1**, we specify that the script should be run with Bash scripting language. (See NOTE 4)

```
1 #!/bin/bash
```

On line **3**, **4** and **5**, we specify our desired output folder path, and the pattern beginning and ending of the output name respectively.

```
3 output_folder=/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Pre-processed/
4 nameStart=Subset_S2_MSIL2A_
5 nameEnd=.dim
```

Line **7** signifies the start of the loop. The script will loop over all files in the specified folder ending with `".zip"`. In each iteration, the path to one input file is denoted by `"i"`.

`"do"` on line **8** and `"done"` on line **13** enclose the steps that should be performed on each input product.

```
7 for i in $(ls -d -l /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Original/*.zip)
8 do
```

On line **9**, we extract the name of the input product; on line **10**, we extract the date of the acquisition from the name.

```
9     product_basename=$(basename $i)
10    ac_date=$(echo $product_basename | cut -d '_' -f 3)
```

On line **11**, we create the final path and name of our output, combining our specified output folder path, the beginning of the output name, the acquisition date and specified name end including the format suffix.

```
11    Output_pathname=$output_folder$nameStart$ac_date$nameEnd
```

On line **12**, we call GPT and specify the path to our saved and edited graph file. Then with `-P{variable name}` we specify the parameters to be replaced in the graph file (marked with `$` in the graph file). We have set the variables when we edited the .xml graph file – **Input**, **Output**. On line **13**, the loop is closed.

```
12    gpt /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/STEP1_graph Prep.xml -Pinput=$i -Poutput=$Output_pathname
13 done
```

If your input and output folders are different you need to edit the file accordingly and save it.

To run the script, we navigate to the folder where it is saved (`/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Code/`) and right-click on the white space and go to **Open Terminal Here**. In the terminal type:

```
sh STEP1_script_gpt.sh
```

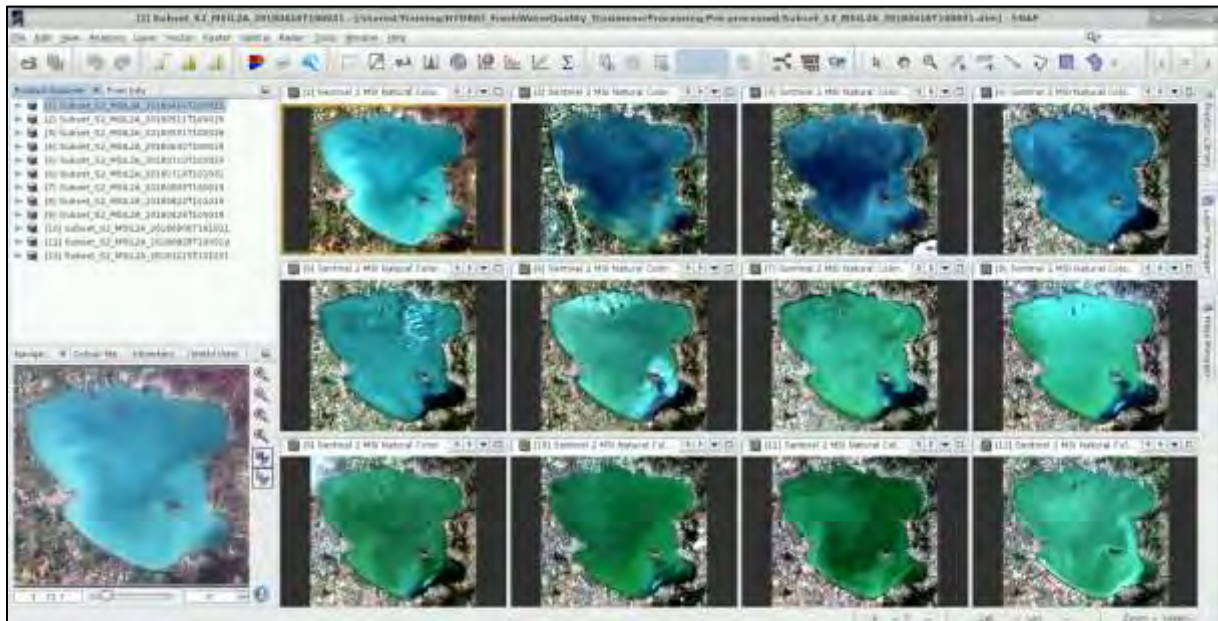
where `STEP1_script_gpt.sh` is the name of the Bash script file. Click **ENTER** to run the script.

```
Terminal: /usr/bin/terminal: /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Code
rus@front: /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Code$ sh STEP1_script_gpt.sh
INFO: org.esa.snap.python.gpt.PyOperatorSpi: Python operator 'S2RutOp' registered (Python module: 's2_rut', class: 'S2RutOp', root: '/home/rus/.snap/system/modules/org.esa.snap-snap-rut.jar')
INFO: org.esa.snap.python.gpt.PyOperatorSpi: Python operator 'py_sambuca_snap_op' registered (Python module: 'sambuca_snap_op', class: 'sambuca_snap_op', root: '/home/rus/.snap/system/modules/org.esa-sen2coral-sen2coral-inversion.jar')
INFO: org.esa.snap.core.gpt.operators.tooladapter.ToolAdapterIO: Initializing external tool adapters
SEVERE: org.esa.s2tbx.dataio.gdal.activator.GDALDistributionInstaller: The environment variable LD_LIBRARY_PATH is not set. It must contain the current folder '.'.
INFO: org.esa.snap.core.util.EngineVersionCheckActivator: Please check regularly for new updates for the best SNAP experience.
Executing processing graph
INFO: org.esa.s2tbx.dataio.s2.ortho.S2OrthoProductReaderPlugin: Building product reader - EPSG:32632
WARNING: org.esa.s2tbx.dataio.metadata.GenericXmlMetadata: Metadata: the path to element [metadata_level] does not exist
WARNING: org.esa.s2tbx.dataio.metadata.GenericXmlMetadata: Metadata: the path to element [granuleidentifier] does not exist
WARNING: org.esa.s2tbx.dataio.metadata.GenericXmlMetadata: Metadata: the path to element [bandid] does not exist
INFO: org.hsqldb.persist.Logger: dataFileCache open start
22%45%67%90% done
First product processed
INFO: org.esa.snap.python.gpt.PyOperatorSpi: Python operator 'S2RutOp' registered (Python module: 's2_rut', class: 'S2RutOp', root: '/home/rus/.snap/system/modules/org.esa.snap-snap-rut.jar')
INFO: org.esa.snap.python.gpt.PyOperatorSpi: Python operator 'py_sambuca_snap_op' registered (Python module: 'sambuca_snap_op', class: 'sambuca_snap_op', root: '/home/rus/.snap/system/modules/org.esa-sen2coral-sen2coral-inversion.jar')
...
INFO: org.hsqldb.persist.Logger: dataFileCache open start
22%45%67%90% done.
All products completed
rus@front: /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Code$
```

The processing time will depend on your VM setup. (approx. 3 mins with 30GB ram). You can check the `/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Pre-processed` for the output products. Close the terminal window.

Now, let's have a look at them. Go back to SNAP and go to **File → Session → Open session..** Navigate to `/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/` and open `Pre-processed.snap`

Once all products are loaded, right-click the first and go to **Open RGB image window**. Select the **Sentinel 2 MSI Natural Colors** profile and click OK. You can repeat this step for all products and the go to **Window → Tile Evenly**.



Even by visual assessment we can determine that increase in biological production occurs throughout the summer and fall and decreases again towards colder months. However, to be able to quantify the concentration of phytoplankton and other substances we need to process the data further.

#### 5.4 STEP 2 – Atmospheric correction & bio-optical model

As mentioned previously, atmospheric correction is crucial, in order to be able to separate different optically active constituents in water (chl-a, TSM, CDOM, phosphorus etc.). Number of different atmospheric correction algorithms exist for Case 2 waters. Here we will use the Case 2 Regional Coast Colour Processor.

While available in SNAP graphical interface and in the command line interface (GPT), it is not available in the Graph builder tool. We cannot therefore create a graph in the GUI as we have done in the previous step, however we can use the GUI to create a parameter file which we can then use in combination with GPT and bash script (or python or another scripting language) to run it in batch.

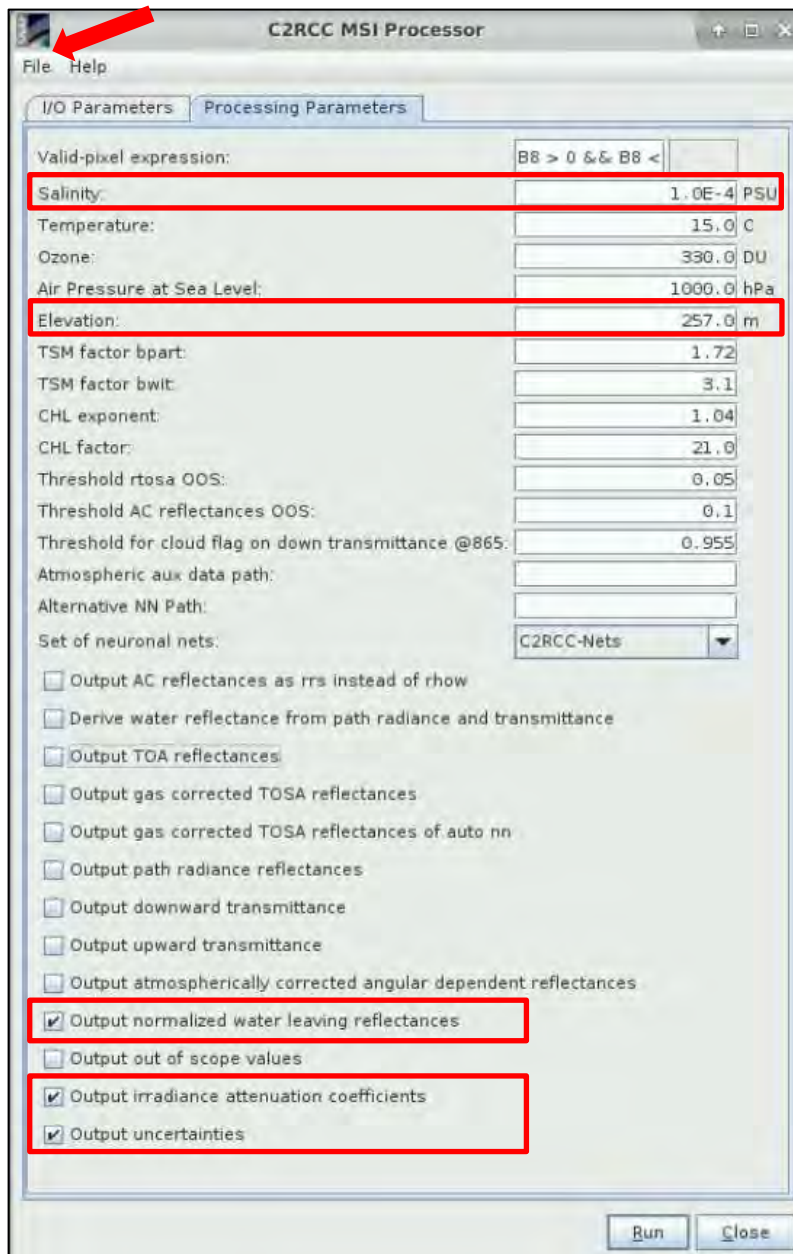
The C2RCC processor allows many different parameters to be set, to accommodate more accurate atmospheric correction and extraction of the optically active constituents.

Let's go to SNAP and set our parameters. Go to **Optical → Thematic Water Processing → C2RCC Processors → S2-MSI**. In the **I/O Parameter** tab, we do not need to change anything.

In the **Processing Parameters** tab, we can set following:



**Salinity:** 0.0001 PDU  
**Elevation:** 257 m



The atmospheric parameters – Temperature (Water), Ozone and Air Pressure at Sea Level – will be set in our bash script, because here we can not specify different value for each image.

We will leave all other parameters as default, including the thresholds and factors used for TSM and Chl-a extraction.

Finally, we can select which data we wish to include in the output. Here we select:

- Output normalized water leaving reflectances
- Output irradiance attenuation coefficients
- Output uncertainties

Then click **File** → **Save Parameters** and save them as **c2rcc\_param.xml** (XML format as the SNAP graphs) to: **/shared/Training/HYDR02\_FreshWaterQuality\_Trasimeno/Processing**

Go to the file location (in file explorer) and open it. It will automatically open in the internet browser window.

Now, we will use another Bash processing script to run the C2RCC algorithm on all our images and assign the respective water temperature, ozone and pressure values.

These parameters will naturally change over time. The values we will use, were manually acquired from ECMWF ERA-Interim global atmospheric reanalysis dataset (<https://www.ecmwf.int/en/forecasts/datasets/reanalysis-datasets/era-interim>) for each image acquisition date. A CSV (*Met\_Aux.csv*) file containing the values is saved in the *AuxData* folder. Below you can see the screenshot of the data, you do not need to open them.



	A	B	C	D	E	F
1	Date	Ozone[DU]	MeanSeaLevelPressure[hPa]	Temperature[C]		
2	20180416	375.9	1010.342	10		
3	20180511	333.5	1015.939	18.2		
4	20180531	326.83	1016.981	20		
5	20180630	345.41	1011.56	25		
6	20180710	335.42	1013.863	26		
7	20180718	323.14	1014.703	26		
8	20180809	314.5	1016.832	27		
9	20180822	321.41	1015.507	27		
10	20180829	310.06	1016.216	27		
11	20180906	299.32	1013.29	22.8		
12	20180928	293.53	1017.787	20		
13	20181115	289.1	1024.182	10		

Now we can go to the Bash script we will use to run the processor on all the images automatically. It is similar to the one used previously for pre-processing. It is named **STEP2\_script\_C2RCC.sh** and you can find it in: **/shared/Training/HYDR02\_FreshWaterQuality\_Trasimeno/Processing/Codes/**

```
*STEP2_script_C2RCC.sh - Mousepad
File Edit Search View Document Help
1 #!/bin/bash
2
3 output_folder=/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/C2RCC/
4 metData=/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/AuxData/Met_aux.csv
5 paramFile=/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/c2rcc_param.xml
6 oldEnd=.dim
7 newEnd=_C2RCC.dim
8
9 for i in $(ls -d -l /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Pre-processed/*.dim)
10 do
11     name=$(basename $i)
12     idate=$(basename $i | cut -d '_' -f 4 | cut -d 'T' -f 1)
13     ozone=$(grep -F $idate $metData | cut -d ',' -f 2)
14     press=$(grep -F $idate $metData | cut -d ',' -f 3)
15     temp=$(grep -F $idate $metData | cut -d ',' -f 4)
16     gpt c2rcc.msi -SsourceProduct=$i -p $paramFile -Pozone=$ozone -Ppress=$press -Ptemperature=$temp
17     -t $output_folder${name%$oldEnd}$newEnd
18 done
```

On lines **3 to 7** we specify input parameters, such as the path to out output directory, path to the meteorological ancillary data (table above), the path to the C2RCC parameter file saved from SNAP GUI and the input file end and output file end, respectively.

```
3 output_folder=/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/C2RCC/
4 metData=/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/AuxData/Met_aux.csv
5 paramFile=/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/c2rcc_param.xml
6 oldEnd=.dim
7 newEnd=_C2RCC.dim
```

Line **9** signifies the start of the loop. The script will iterate over all files in the specified folder ending with **“.dim”**. In each iteration the path to one input file is denoted by **“i”**. **“do”** on line **10** and **“done”** on line **17** enclose the steps that should be performed on each input product.

```
9 for i in $(ls -d -l /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Pre-processed/*.dim)
10 do
```

On line **11**, we extract the input file name, on line **12** we extract the image acquisition date. On **13 to 15** we extract from the **Met\_aux.csv** file the ozone, air pressure at sea level and water temperature for the appropriate date.

```
11 name=$(basename $i)
12 idate=$(basename $i | cut -d '_' -f 4 | cut -d 'T' -f 1)
13 ozone=$(grep -F $idate $metData | cut -d ',' -f 2)
14 press=$(grep -F $idate $metData | cut -d ',' -f 3)
15 temp=$(grep -F $idate $metData | cut -d ',' -f 4)
```

Finally, on line **16** we call GPT (SNAP command line utility) and the c2rcc.msi tool. Then, we load the parameter file we have created in the SNAP GUI. However, in this file the ozone pressure and temperature parameters were not set in the parameter file, instead we overwrite the values in the parameter file with the values extracted from the **Met\_aux.csv** file. “done” at line **17** closes the loop.

```
16 gpt c2rcc.msi -SsourceProduct=$i -p $paramFile -Pozone=$ozone -Ppress=$press
-Ptemperature=$temp -t $output_folder${name%$oldEnd}$newEnd
17 done
```

To run the script, we navigate to the folder where it is saved (**/shared/Training/HYDR02\_FreshWaterQuality\_Trasimeno/Processing/Code/**) and right-click on the white space and go to **Open Terminal Here**. In the terminal type:

```
sh STEP2_script_C2RCC.sh
```

where **STEP2\_script\_C2RCC.sh** is the name of the Bash script file. Click **ENTER** to run the script.

```
Terminal - rus@front: /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Code
rus@front: /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Code$ sh STEP2_script_C2RCC.sh
INFO: org.esa.snap.python.gpf.PyOperatorSpi: Python operator 'S2RutOp' registered
(Python module: 's2_rut', class: 'S2RutOp', root: '/home/rus/.snap/system/modules/
org-esa-snap-snap-rut.jar')
INFO: org.esa.snap.python.gpf.PyOperatorSpi: Python operator 'py_sambuca_snap_op'
registered (Python module: 'sambuca_snap_op', class: 'sambuca_snap_op', root: '/ho
me/rus/.snap/system/modules/org-esa-sen2coral-sen2coral-inversion.jar')
INFO: org.esa.snap.core.gpf.operators.tooladapter.ToolAdapterIO: Initializing exte
rnal tool adapters
SEVERE: org.esa.s2tbx.dataio.gdal.activator.GDALDistributionInstaller: The environ
ment variable LD_LIBRARY_PATH is not set. It must contain the current folder '.'.
INFO: org.esa.snap.core.util.EngineVersionCheckActivator: Please check regularly f
or new updates for the best SNAP experience.
INFO: org.hsqldb.persist.Logger: dataFileCache open start
INFO: org.esa.s3tbx.c2rcc.ancillary.AtmosphericAuxdataBuilder: Atmospheric auxdata
product can't be used. At least one is not specified. Using constant values for o
zone (375.9) and surface pressure (1010.342).
INFO: org.esa.snap.core.gpf.common.WriteOp: Start writing product c2rcc.msi$170AC2
93801 to /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/C2RCC/Subs
et S2 MSIL2A_20180416T100031_C2RCC.dim
Writing...
....10%....20%....30%....40%....50%....60%....70%....80%....90%....100% done.
INFO: org.esa.snap.core.gpf.common.WriteOp: End writing product Subset S2 MSIL2A 2
```

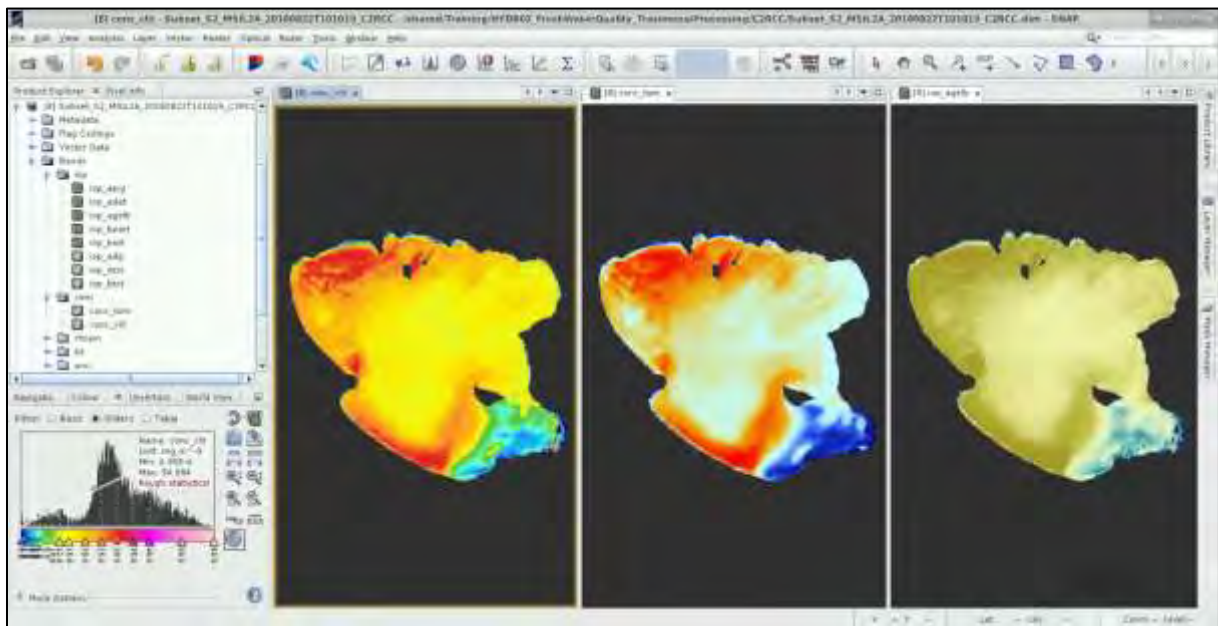
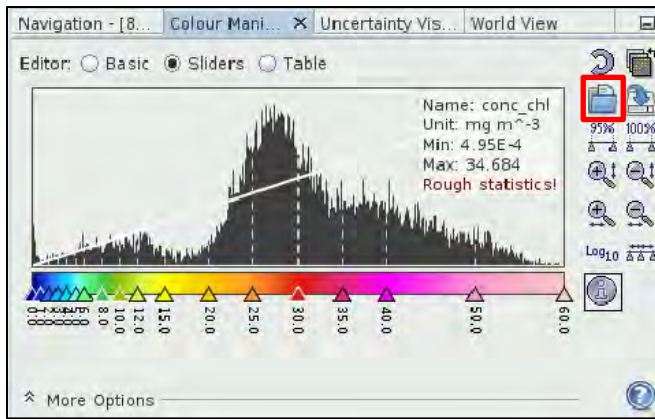
Finally, we can go back to SNAP and go to **File → Session → Open Session**, then click **Yes** in the dialog that opens. Then open:

**/shared/Training/HYDR02\_FreshWaterQuality\_Trasimeno/C2RCC\_processed.snap**

Now, let’s investigate the results. In **Product Explorer** open the structure of one product and go to **Bands**. Here we choose the product #8 from 22 August 2018 and opened the **conc → chl\_conc & tsm\_conc** and the **iop → iop\_agelp** (gelbstoff == CDOM attenuation coefficient).

For each opened view you can go to the **Colour Manipulation** tab and click on **Import colour palette** from text file . In **/shared/Training/HYDR02\_FreshWaterQuality\_Trasimeno/AuxData** you can find the appropriate colour map for *Chlorophyll*, *TSM* and *aCDOM*. To keep the pre-set values, click **No** in the **Import Colour Palette** dialog. To visualize the images next to one another you can go to **Window → Tile Horizontally**.





Apart from the chlorophyll-a, TSM and aCDOM extracted using bio-optical model, in the **rhown** folder we can also find the atmospherically corrected normalized water leaving radiances, which we can use for our empirical models.

### 5.5 STEP 3 – Empirical algorithms

Empirical approaches require in-situ measurements of the variable of interest which serve as a basis for establishing an empirical relationship - linear or non-linear regression - with the water leaving radiance measured in one or multiple sensor bands.

While there are many different band ratios and equations reported in literature (see Ansper and Alikas (2019) and Dörnhöfer and Oppelt (2016) for exhaustive summary), we will use the following two:

- Two-Band NIR–Red Ratio Model - It uses red band located in the range of maximum chl-a absorption between 660 and 690 nm and Near-infrared band between 700 and 720 nm.

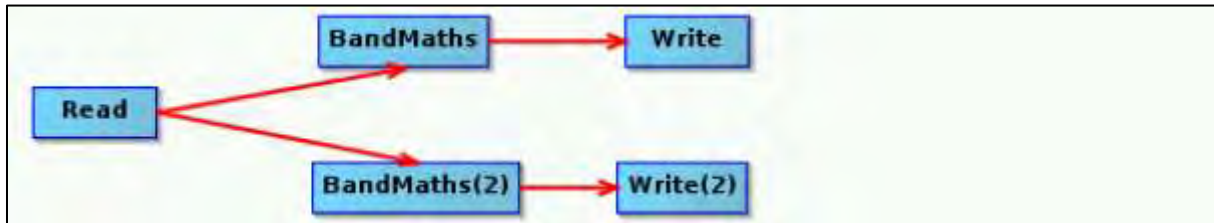
$$Chl\ a \sim \frac{R(\lambda_{705})}{R(\lambda_{665})} \rightarrow \frac{B5}{B4}$$

- Maximum Chlorophyll Index (MCI) – It is so-called line-height algorithm, developed for detecting surface blooms and near-surface vegetation in inland, coastal and ocean waters, it has been used with chl-a over 10 mg/m<sup>3</sup>.

$$MCI = R_{705} - R_{665} - 0.53 * (R_{740} - R_{665}) \rightarrow B5 - B4 - 0.53 * (B6 - B4)$$

Where index 0.53 represents the ratio of wavelengths of the used bands. Instead of band 6, band 7 is sometimes used.

To calculate the band-ratios we will create another short graph. Go to **Tools → Graph Builder** and add two Band Maths operators (**Add → Raster → BandMaths**). Then add another Write operator (**Add → Input/Output → Write**). Then connect the operators as shown below.



Now, in the BandMaths operator tab below, set the appropriate band math expressions as follows.

Target Band: R705\_R665

Target Band Type: float32

Band Unit:

No-Data Value: NaN

Expression: if Valid\_PE then (rhown\_B5/ rhown\_B4) else NaN

Edit Expression...

Load Save Clear Note Help Run

Arithmetic Expression Editor

Expression: if Valid\_PE then (rhown\_B5/ rhown\_B4) else NaN

Ok, no errors.

OK Cancel Help

In BandMath (2) operator tab, set:

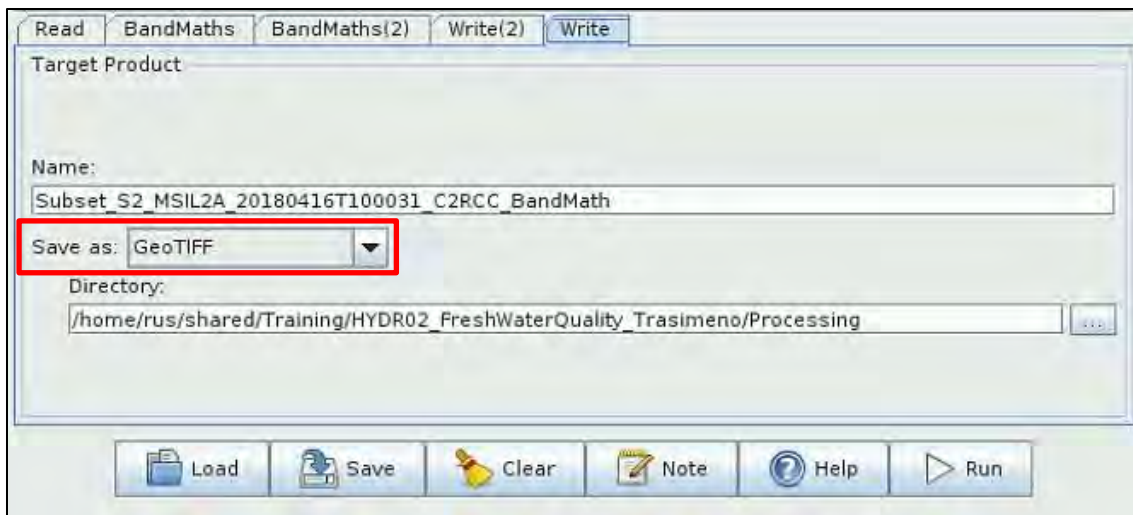
Target Band: MCI

No-Data value: NaN

Expression: `if Valid_PE then (rhown_B5 - rhown_B4 - 0.53*(rhown_B6 - rhown_B4)) else NaN`

Then go to the **Write** tabs, in both write tabs change the format – **Save as: GeoTIFF**.

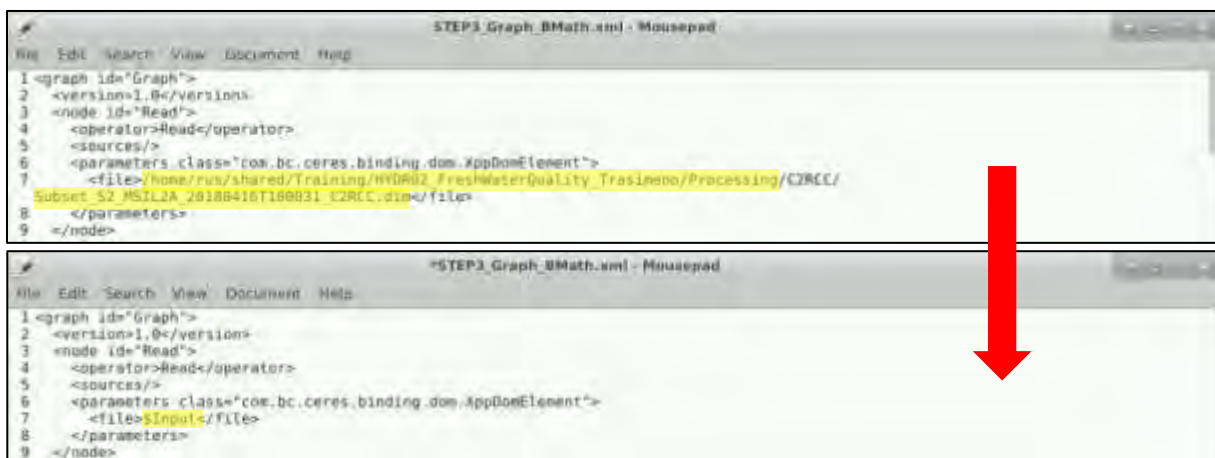




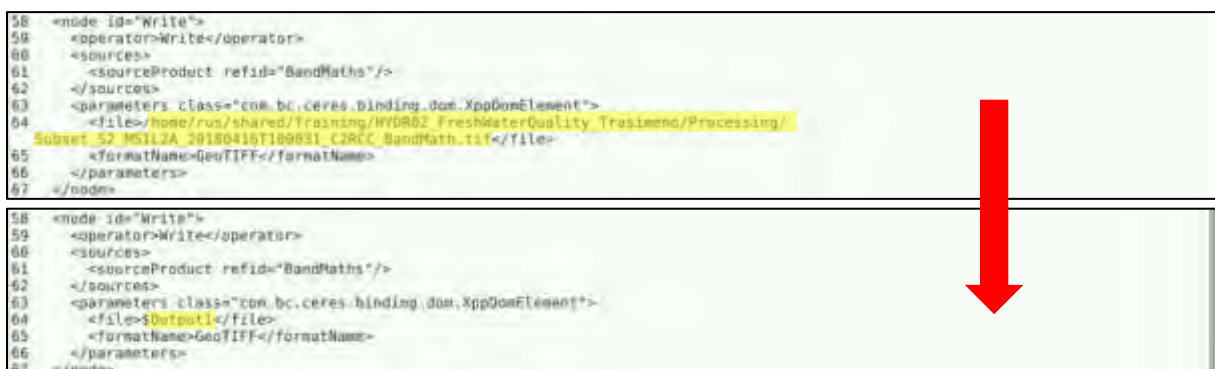
Then save the graph as **STEP3\_Graph\_BMath.xml**, to the **.../Processing** folder. Now we will edit the graph as we have done for the pre-processing step. The only difference this time is that we have two outputs, instead of one.

In **File Explorer**, navigate to the **.../Processing** folder, right-click on **STEP3\_Graph\_BMath.xml** and click ok **Open in Mousepad**.

In line 7 (in the node with id='Read') replace the path to input file with (between <file> and </file>): **\$Input** ("**\$**" signifies that what follows is a parameter).



Now we need to find the **Write** node (in the node with id="Write") – It will likely not be on the same line as shown below as the position might be random. Here, replace the path to output file with (between <file> and </file>): **\$Output1**



Then find the **Write(2)** node (in the node with id="Write(2)") – It will likely not be on the same line as shown below as the position might be random. Here, replace the path to output file with (between <file> and </file>): `$Output2` then save the changes and close the Mousepad.

Now, go to the `.../Processing/Code/` folder and open the `STEP3_script_gpt.sh`

```

STEP3_script_BMath.sh - Mousepad
File Edit Search View Document Help
1 #!/bin/bash
2
3 output_folder1=/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/R705_R665/
4 output_folder2=/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/MCI/
5 oldEnd= C2RCC.dim
6 newEnd1= R705_R665.tif
7 newEnd2= MCI.tif
8
9 for i in $(ls -d -l /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/C2RCC/*.dim)
10 do
11     name=$(basename $i)
12     output1=$output_folder1${name%$oldEnd}$newEnd1
13     output2=$output_folder2${name%$oldEnd}$newEnd2
14     gpt /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/STEP3_Graph_BMath.xml -PInput=$i
        -POutput1=$output1 -POutput2=$output2
15 done
16 |

```

To run the script, we navigate to the folder where it is saved (`/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Code/`) and right-click on the white space and go to **Open Terminal Here**. In the terminal type:

```
sh STEP3_script_BMath.sh
```

where `STEP3_script_BMath.sh` is the name of the Bash script file. Click **ENTER** to run the script.

```

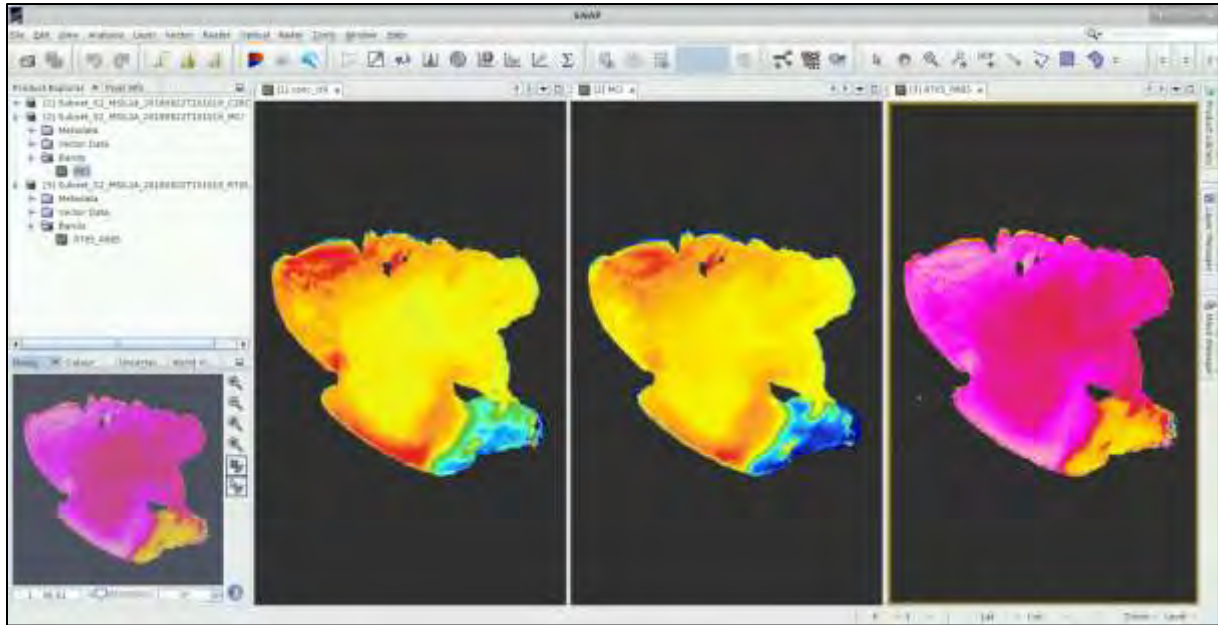
Terminal - rus@front: /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Code
File Edit View Terminal Tabs Help
rus@front:/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/Code$ sh
STEP3_script_BMath.sh
INFO: org.esa.snap.python.gpf.PyOperatorSpi: Python operator 'S2RutOp' registered
(Python module: 's2_rut', class: 'S2RutOp', root: '/home/rus/.snap/system/modules/
org-esa-snap-snap-rut.jar')
INFO: org.esa.snap.python.gpf.PyOperatorSpi: Python operator 'py_sambuca_snap_op'
registered (Python module: 'sambuca_snap_op', class: 'sambuca_snap_op', root: '/ho
me/rus/.snap/system/modules/org-esa-sen2coral-sen2coral-inversion.jar')
INFO: org.esa.snap.core.gpf.operators.tooladapter.ToolAdapterIO: Initializing exte
rnal tool adapters
SEVERE: org.esa.s2tbx.dataio.gdal.activator.GDALDistributionInstaller: The environ
ment variable LD_LIBRARY_PATH is not set. It must contain the current folder '.'.
INFO: org.esa.snap.core.util.EngineVersionCheckActivator: Please check regularly f
or new updates for the best SNAP experience.
Executing processing graph
INFO: org.hsqldb.persist.Logger: dataFileCache open start
11%22%33%45%56%67%78%90% done.
INFO: org.esa.snap.python.gpf.PyOperatorSpi: Python operator 'S2RutOp' registered
(Python module: 's2_rut', class: 'S2RutOp', root: '/home/rus/.snap/system/modules/

```

When the script finishes, we have now two additional folders in our `.../Processing` folder. Go back to SNAP then go to **File → Session → Open session**. Click **Yes** and open:

`/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processed.snap`

Open the `conc_chl` band in the `C2RCC` product as well as the `MCI` and `R705_665` bands. We can apply the same colour pallet (in the case of `C2RCC` select **No**, for other two bands select **Yes** to stretch the colour pallet between `mon` and `max`). Keep in mind that the `MCI` and `R705_R665` are not yet concentration measurements but only indicators.



Now in a final step we can use the in-situ data to define the empirical relationship between the indicators and the true values. The in-situ data below have been acquired from *Agenzia Regionale per la Protezione Ambientale dell'Umbria* (Arpa Umbria – Regional Environment Protection Agency of Umbria region in Italy).

In the **.../AuxData** folder you can find polygons (approx. 50x50 m) surrounding the sampling point. These can be used for example in QGIS to extract the mean value from the images closest to the in-situ data sampling date as shown below (e.g. *Zonal statistics* tool).

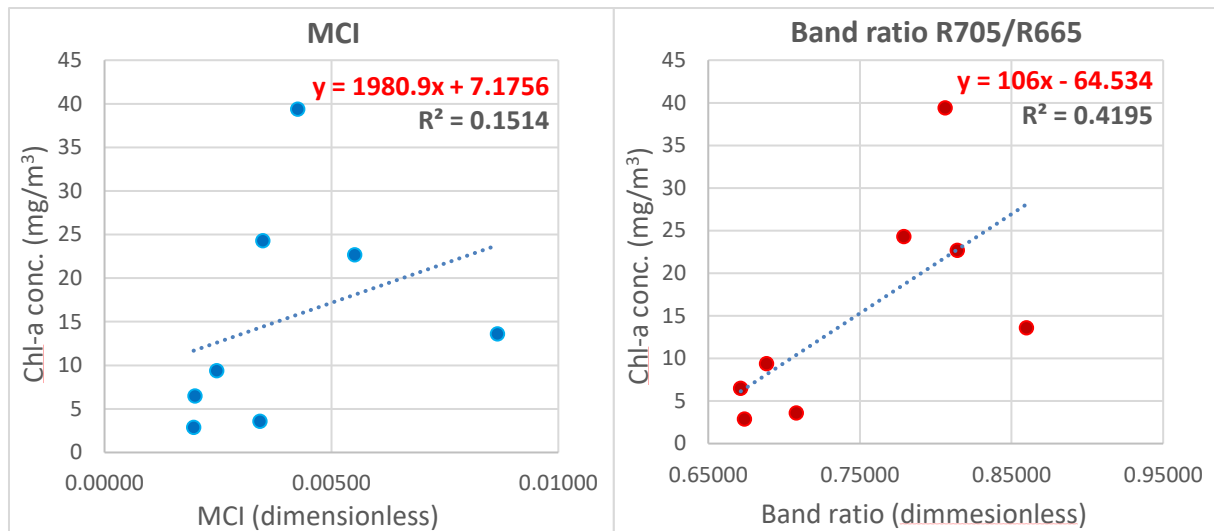
Data	Chl-a (mg/m3)	Station code	Image difference in days	MCI	R705/R665
16-May-18	2.9	TRS30	-5	0.00196	0.67388
16-May-18	9.4	TRS35	-5	0.00247	0.68835
09-Jul-18	6.5	TRS30	1	0.00199	0.67142
09-Jul-18	3.6	TRS35	1	0.00342	0.70807
04-Sep-18	39.4	TRS30	2	0.00425	0.80622
04-Sep-18	24.3	TRS35	2	0.00349	0.77898
21-Nov-18	22.7	TRS30	-6	0.00551	0.81416

Then by simple regression, we calculate the empirical relationship between the indicator value and the in-situ measurement.

NOTE: the  $R^2$  value is the coefficient of determination – how well can our indicator predict actual concentration – here we have very low  $R^2$  for both indicators, which means that our Empirical models will not perform very well. This is due to the fact that we have used very few in-situ data samples and moreover they were not taken on the same day as our images. Chlorophyll value changes dynamically in time and longer period between in-situ measurement and image date will result in decorrelation.

However, in order to demonstrate the method, we can now use our empirical models to retrieve the chlorophyll concentrations and compare them with the one derived by C2RCC algorithm.





You can do this easily by editing the graph used to calculate the MCI and R705\_R665. We can change the expressions and the band names in the appropriate nodes:

```

10 <node id="BandMaths">
11   <operator>BandMaths</operator>
12   <sources>
13     <sourceProduct refid="Read"/>
14   </sources>
15   <parameters class="com.bc.ceres.binding.dom.XppDomElement">
16     <targetBands>
17       <targetBand>
18         <name>Chl_conc_R</name>
19         <type>float32</type>
20         <expression>if Valid_PE then (106*(rrown_B5/ rrown_B4) - 64.534) else NaN</expression>
21         <description/>
22         <unit/>
23         <noDataValue>NaN</noDataValue>
24       </targetBand>
25     </targetBands>
26   </parameters>
27 </node>

```

```

29 <node id="BandMaths(2)">
30   <operator>BandMaths</operator>
31   <sources>
32     <sourceProduct refid="Read"/>
33   </sources>
34   <parameters class="com.bc.ceres.binding.dom.XppDomElement">
35     <targetBands>
36       <targetBand>
37         <name>Chl_conc_MCI</name>
38         <type>float32</type>
39         <expression>if Valid_PE then (1980.9*(rrown_B5- rrown_B4-0.53*(rrown_B6- rrown_B4)) + 7.1756) else NaN</expression>
40         <description/>
41         <unit/>
42         <noDataValue>NaN</noDataValue>
43       </targetBand>
44     </targetBands>
45   </parameters>
46 </node>

```

Then in the Bash script we can rename the output folders and output file names.

```

STEP4_script_BMath_chl.sh - Mousepad
File Edit Search View Document Help
1 #!/bin/bash
2
3 output_folder1=/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/R705_R665_Ch1/
4 output_folder2=/shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/MCI_Ch1/
5 oldEnd= C2RCC.dim
6 newEnd1= R705_R665_Ch1.tif
7 newEnd2= MCI_Ch1.tif
8
9 for i in $(ls -d -l /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/C2RCC/*.dim)
10 do
11   name=$(basename $i)
12   output1=$output_folder1${name%$oldEnd}$newEnd1
13   output2=$output_folder2${name%$oldEnd}$newEnd2
14   gpt /shared/Training/HYDR02_FreshWaterQuality_Trasimeno/Processing/STEP4_Graph_BMath_Ch1.xml -
15   PInput=$i -POutput1=$output1 -POutput2=$output2
16 done

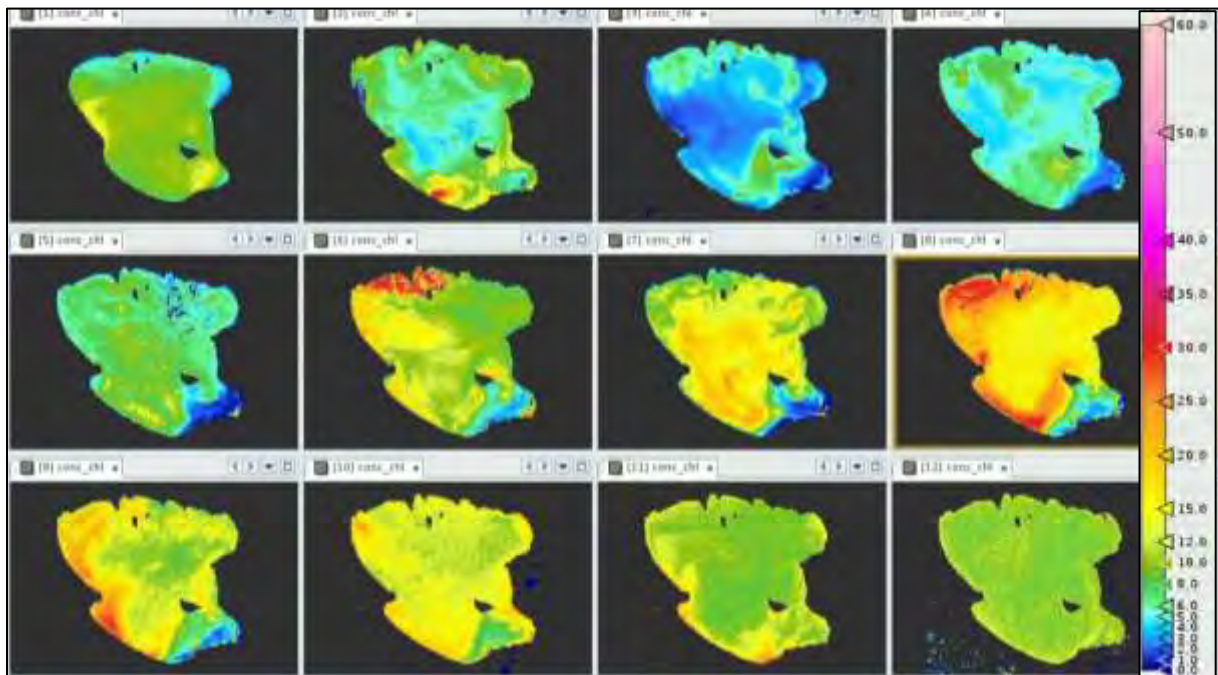
```



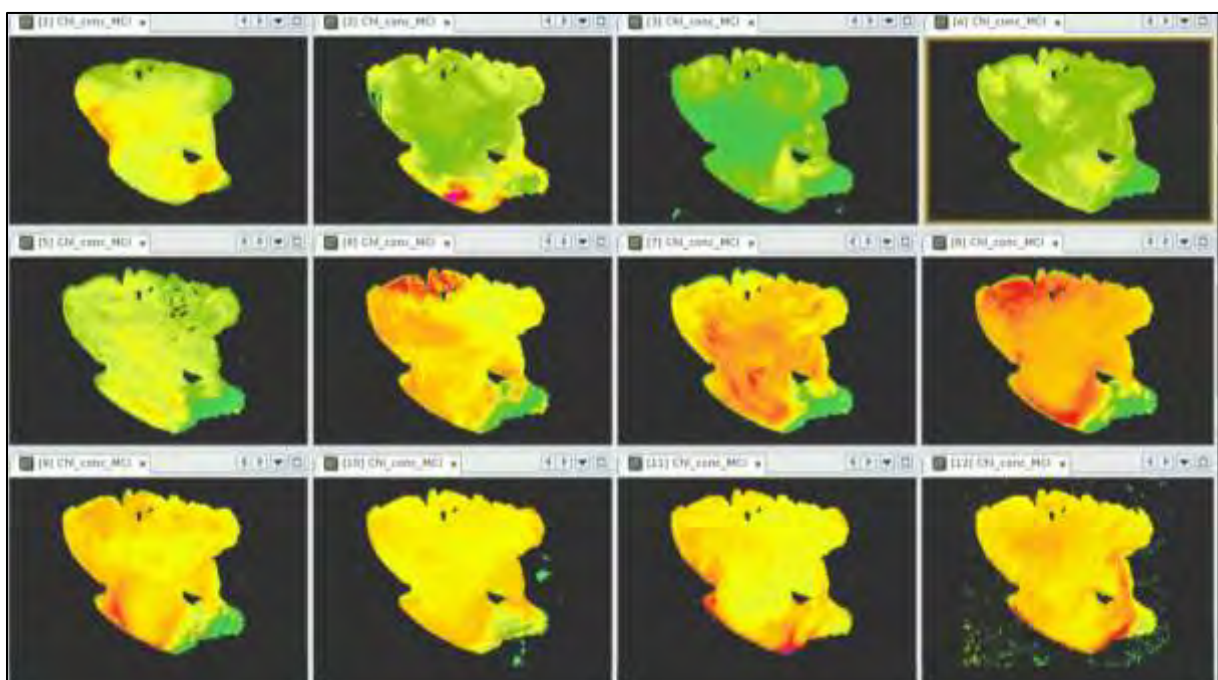
We can save the edited graph and the edited Bash script file under a different name (keep in mind that you need to change the path to the graph in the bash script file as well). And run the processing as previously.

When the script finishes, you should now have two additional folders in our **.../Processing** folder with suffix “\_Chl”. In these folders we can find our Chlorophyll-a concentration calculated using empirical model based on Maximum Chlorophyll Index (**.../MCI\_ChI**) and NIR-RED Ratio (**.../R705\_R655\_ChI**).

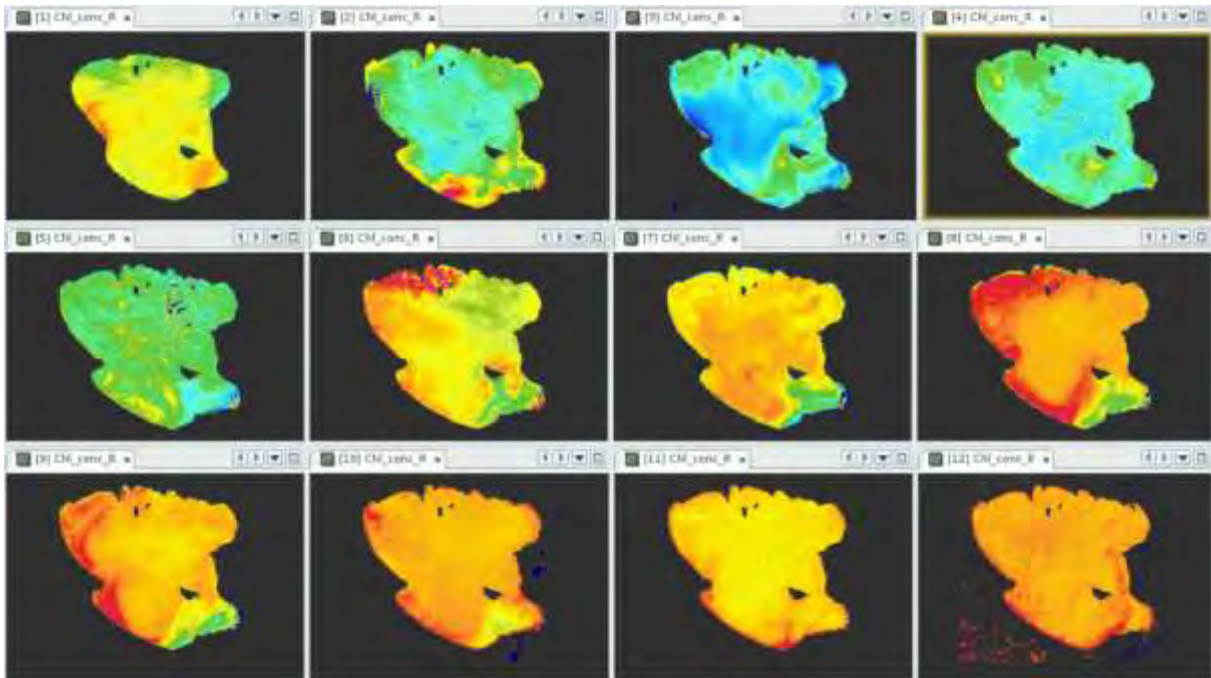
We can now compare our results with the chlorophyll concentration derived using the bio-optical model employed in C2RCC.



First let's have a look at Chlorophyll-a concentration derived from MCI using empirical model calibrated with our in-situ data.



We have used the same colour pallet as for the C2RCC derived concentrations (page 22) and visualized all images side by side (**Window → Tile Evenly**). We can see that compared to the C2RCC; our empirical MCI model does not capture the low values at all. While in higher values the distribution seems quite similar, it tends to be biased towards higher values. Now, let's have look at the Two-Band NIR-Red Ratio model. It seems to capture lower values better than the MCI based model but compared to C2RCC it seems also biased towards higher values.



The main limitation of our models is that we do not have sufficient number of in-situ measurements to properly calibrate the models. Moreover, our in-situ data have been acquired as much as 6 days apart from our images. Since chlorophyll concentration changes dynamically over time, this difference introduces significant uncertainty to our model, making it even more unreliable.

In this exercise we have aimed at demonstrating the method of deriving chlorophyll concentration from remotely sensed data using bio-optical algorithm implemented in C2RCC and two empirical models. However, **for reliable results, larger number of in-situ data for training and validation are necessary!**

**THANK YOU FOR FOLLOWING THE EXERCISE!**

## 6 Extra steps

### 6.1 Downloading the outputs from VM

On your keyboard, press **Ctrl+Alt+Shift**. A pop-up window will appear on the left side of the screen. Click on the bar below **Devices**, the folder structure of your VM will appear. Navigate to your Processing folder and **double click any file you want to download**.



## 7 Sources & References

Bresciani, Mariano, Monica Pinardi, Gary Free, Giulia Luciani, Semhar Ghebrehiwot, Marnix Laanen, Steef Peters, Valentina Della Bella, Rosalba Padula, and Claudia Giardino. 2020. "The Use of Multisource Optical Sensors to Study Phytoplankton Spatio-Temporal Variation in a Shallow Turbid Lake." *Water* 12(1):284.

Dörnhöfer, Katja, and Natascha Oppelt. 2016. "Remote Sensing for Lake Research and Monitoring – Recent Advances." *Ecological Indicators* 64:105–22.

Giardino, C., V. E. Brando, P. Gege, N. Pinnel, E. Hochberg, E. Knaeps, I. Reusen, R. Doerffer, M. Bresciani, F. Braga, S. Foerster, N. Champollion, and A. Dekker. 2019. "Imaging Spectrometry of Inland and Coastal Waters: State of the Art, Achievements and Perspectives." *Surveys in Geophysics* 40(3):401–29.

Koponen, Sampsa. 2006. *Remote Sensing of Water Quality for Finnish Lakes and Coastal Areas*. Helsinki University of Technology.

Toming, Kaire, Tiit Kutser, Alo Laas, Margot Sepp, Birgot Paavel, and Tiina Nõges. 2016. "First Experiences in Mapping Lake Water Quality Parameters with Sentinel-2 MSI Imagery." *Remote Sensing* 8(8):640.

Warren, M. A., S. G. H. Simis, V. Martinez-Vicente, K. Poser, M. Bresciani, K. Alikas, E. Spyrakos, C. Giardino, and A. Ansper. 2019. "Assessment of Atmospheric Correction Algorithms for the Sentinel-2A MultiSpectral Imager over Coastal and Inland Waters." *Remote Sensing of Environment* 225:267–89.

## FOLLOW US!!!



[@RUS-Copernicus](https://twitter.com/RUS-Copernicus)



[RUS-Copernicus](https://www.linkedin.com/company/RUS-Copernicus)



[RUS-Copernicus](https://www.facebook.com/RUS-Copernicus)



[RUS Copernicus Training](https://www.youtube.com/channel/UCv3v3v3v3v3v3v3v3v3v3v3)



[RUS-Copernicus website](https://www.rus-copernicus.org/)



[RUS-Copernicus Training website](https://www.rus-copernicus.org/training/)