



Global Monitoring for Environment and Security (GMES) and Africa

eStation 2.0

Administration manual

Version 2.1.2

January 2019

Date: 22/01/2019

Ref: MESA/DOC/eStation/AdminManual



Abstract / Résumé

This document provides instructions for the management of the eStation 2.0 system on MESA computers.

	Name	Position
Prepared by	Marco Clerici	JRC-EC responsible for MESA
Contributions/Reviews by	Antoine Royer	JRC-EC thematic Expert for MESA
	Jurriaan Van't Klooster	IT-GIS Specialist
	Vijay Charan Venkatachalam	IT-GIS Specialist

	Details	Date
Version history	Draft 0.1	December 2014
	Draft 1.0	May 2015
	Draft 1.1	July 2015
Version 1.0	Version delivered to SuCr#1 for MESA stations installation	November 2015
Version 1.1	Changes after 1.0: archives creation from requests	August 2016
Version 2.1.2	Changes for version 2.1.2: manual version change	January 2019

Contents

1. INTRODUCTION	1
1.1 SCOPE OF THE DOCUMENT.....	1
1.2 DOCUMENT ORGANIZATION.....	1
1.3 APPLICABLE AND REFERENCE DOCUMENTS	2
2. ACCESS TO THE ESTATION 2.0	3
2.1 LOGIN TO PC2 AND PC3	3
2.2 REMOTE ACCESS TO THE E-STATION	5
2.2.1 <i>Connecting a Windows PC to the MESA LAN (optional)</i>	5
2.2.2 <i>Access the eStation dataset via ftp</i>	6
2.2.3 <i>Access the eStation GUI from a browser</i>	7
2.2.4 <i>Connection by remote shell (ssh)</i>	8
3. ESTATION MAINTENANCE	10
3.1 MONITORING THE STATUS OF THE SYSTEM	10
3.1.1 <i>Monitoring the status of the disk</i>	10
3.1.2 <i>Monitoring the status of the System</i>	12
3.1.3 <i>Checking the Log files</i>	14
3.2 IMPORTING DATASETS.....	15
3.2.1 <i>Historical archives import</i>	16
3.2.2 <i>Archives from User Request</i>	17
3.3 PERFORMING VERSIONS UPGRADES	22
3.4 RECOVERY MODE (FAILOVER/FAILBACK).....	24
3.5 MANAGING THE POSTGRESQL DATABASE	24
3.5.1 <i>Check the status of the postgresql server</i>	24
3.5.2 <i>Visualize the database tables (pgAdminIII)</i>	25
3.5.3 <i>Restore a previous copy of the database</i>	29
3.6 DIAGNOSTIC TOOLS AND USER SUPPORT	31
3.6.1 <i>System Report</i>	31
3.6.2 <i>Install Report</i>	32
3.6.3 <i>Tuleap for User Support</i>	32
4. SYSTEM DESCRIPTION	33
4.1 DIRECTORY STRUCTURE.....	33
4.2 CONFIGURATION FILES OF E-STATION 2.0.....	35
4.3 FILE AND DIRECTORY NAMING CONVENTIONS.....	40
4.3.1 <i>PCs synchronization</i>	45

4.3.2	Description of the eStation .rpm	49
4.3.3	Database Description	50
5.	TROUBLESHOOTING	54
5.1	THE E-STATION 2.0 WEB INTERFACE IS NOT DISPLAYED	54

List of Tables and Figures

Table 1:	Applicable documents.....	2
Table 2:	Reference documents	2
Figure 1:	Login Users.....	3
Figure 2:	Login Password	4
Figure 3:	Change language at login.....	4
Figure 4:	Network configuration in Windows 7	5
Figure 5:	Modify the host file in Windows 7.....	6
Figure 6:	Filezilla to access eStation data	7
Figure 7:	Confirm host key message	7
Figure 8:	eStation accessed in Filezilla	7
Figure 9:	eStation GUI from a remote PC.	8
Figure 10:	Example of PuTTY connection screen.....	9
Figure 11:	PuTTY first connection.....	9
Figure 12:	Disk Analyser Tool.....	10
Figure 13:	Disk Usage Analyser View	11
Figure 14:	System Monitor utility	12
Figure 15:	Monitor Tab - System Description	12
Figure 16:	Monitoring System Resources	13
Figure 17:	System Processes	13
Figure 18:	Logfiles of eStation 2.0.....	14
Figure 19:	Archives disk connected.	16
Figure 20:	Update location of the Historical Archives.	16
Figure 21:	Change dir for Ingestion.....	17
Figure 22:	Start the ingestion.....	17
Figure 23:	Historical Archives ingestion finished	17
Figure 24:	Structure of request file.....	19
Figure 25:	Check postgresql status	24
Figure 26:	Stop postgresql	24
Figure 27:	Start postgresql.....	25

Figure 28: Launch pgAdmin III.....	25
Figure 29: new connection in pgAdminIII	26
Figure 30: connection parameters for estationdb.....	26
Figure 31: pgAdminIII - save password	27
Figure 32: estationdb overview in pgAdminIII.....	27
Figure 33: estationdb tables	27
Figure 34: server instrumentation message	28
Figure 35: list of dumps of estation DB.....	29
Figure 36: dbrestore.sh script.....	30
Figure 37: dbrestore script execution.....	30
Figure 38: dbrestore script result	30
Figure 39: Creation of System Report.....	31
Figure 40: Save System Report	31
Figure 41: Location of the System Report.....	32
Figure 42: Create Install Report	32
Table 4: System Settings	35
Table 5: User settings.....	37
Figure 43: data directory structure - 1.....	40
Figure 44: data directory structure – 2	41
Figure 45: data directory structure - 3.....	41
Figure 46: Example of eStation 2.0 file naming	42
Table 6: List of eStation 2.0 images metadata.....	42
Figure 47: bucardo from the command line	46
Figure 48: all bucardo objects.....	47
Figure 49: status of bucardo sync	47
Figure 50: 'products' schema	50
Figure 51: spirits table.....	52
Figure 52: 'analysis' schema.....	53

ACRONYMS and DEFINITIONS

AMESD	African Monitoring of Environment for Sustainable Development
ACMAD	African Centre of Meteorological Applications for Development
AGRHYMET	Centre Régional de Formation et d'Application en Agrométéorologie et Hydrologie Opérationnelle
AU	African Union
BDMS	Botswana Department of Meteorological Services
CICOS	Commission Internationale du Bassin Congo-Oubagui-Sangha
CWG	The MESA Continentalisation Working Group
EO	Earth Observation
EUMETSAT	European Organisation for the Exploitation of Meteorological Satellites
EUMETCast	EUMETSAT's primary dissemination mechanism for the near real-time delivery of satellite data and products
FTP	File Transfer Protocol
GIS	Geographical Information System
IOC	Indian Ocean Commission
JRC	Joint Research Centre of the European Commission
LAN	Local Area Network
MESA	Monitoring for Environment and Security in Africa
MOI	Mauritius Oceanography Institute
REC	Regional Economic Communities
RIC	Regional Implementation Centre
TA	Technical Assistance
TAT	Technical Assistance Team
THEMA	Regional and Continental Thematic Actions

1. INTRODUCTION

1.1 SCOPE OF THE DOCUMENT

This document describes how to maintain the eStation 2.0 application, installed on the MESA computers, or on a different hardware. It is meant mainly for the IT Specialist taking care of the basic operations of configuring the system, performing diagnostic actions and recovering from faulty conditions are described. It also contains some additional insights in the eStation 2 (Chapter 4) for Advanced Users willing to further customize the processing performed on the system.

1.2 DOCUMENT ORGANIZATION

The present document is structured into the following chapters:

- **Chapter 2: Access to the Station**
It describes how to connect to the system, both from the local PC and from a remote computer that can access the eStation through a LAN.
- **Chapter 3: System Maintenance**
It describes the main operations to be performed for the system maintenance, including monitoring the disk space, perform diagnostic and recovery operations.
- **Chapter 4: System Description**
It describes the organization of the System in terms of directory structure, postgresql database structure. It is often used as a reference from other chapters of the document.
- **Chapter 5: Trouble Shooting**
It describes the most common errors/problems that can be found on the system and how to deal with them.
- **Annex: MESA Station Administration and Maintenance Manual**
This document is generated by the Supply Contractor and represents a Reference Document (RD-3) for the current one. Therefore it is attached to the current document in order to facilitate reading of the User (many references are done to RD-3).

1.3 APPLICABLE AND REFERENCE DOCUMENTS

Id	Title	Date	Reference
AD-1			
AD-2			
AD-3			
AD-4			

Table 1: Applicable documents

Id	Title	Date	Reference
RD-1	eStation 2.0 User Manual	October 2015	MESA/DOC/eStation/UserManual
RD-2	MESA Station SW Installation Manual	November 2015	TASF-MESA-MIM-10 (EN) TASF-MESA-MIM-13 (FR)
RD-3	MESA Station Administration and Maintenance Manual	November 2015	TASF-MESA-MAM-11

Table 2: Reference documents

2. ACCESS TO THE EStATION 2.0

This section describes how to log in the MESA Station, which are the defined Users, and how to access the dataset from an external PC

2.1 LOGIN TO PC2 AND PC3

The following Users are defined on the AMESD/MESA computers for eStation 2:

Name	Password	Role	Comments
adminuser	mesadmin	Manages the installation, the upgrades and ensure maintenance of the system	He/she is expected to be an IT System Administrator. He/she has administrative rights on the machine ¹
analyst	mesa2015	Thematic Expert using the system to perform environment monitoring.	He/she is expected to be an Environmental Analyst.

Table 3: eStation2 Users

When the computer is switched on, a login interface is presented (see Figure 1); the User can log as 'adminuser' (for maintenance and configuration) or as 'Thematic User' (for normal usage of the eStation).

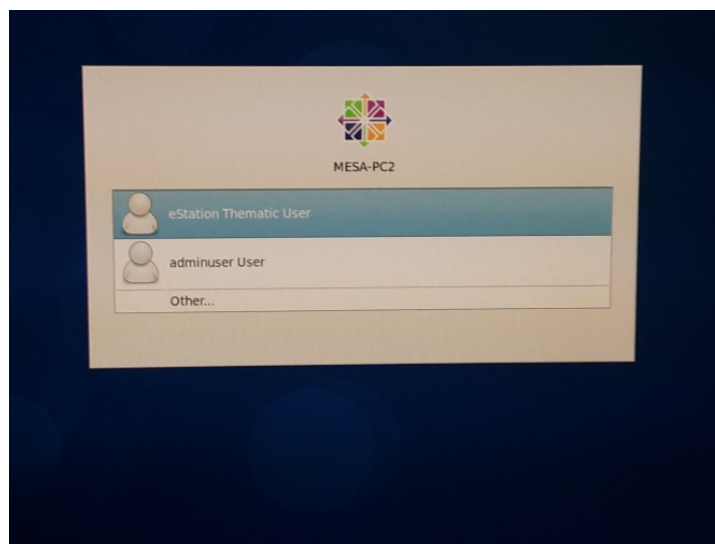


Figure 1: Login Users

¹ In the Linux terminology, the **adminuser** has 'root' rights and can execute 'sudo' commands.

The user is requested for the 'password' (see **Table 3**) to be entered, as represented in Figure 2.

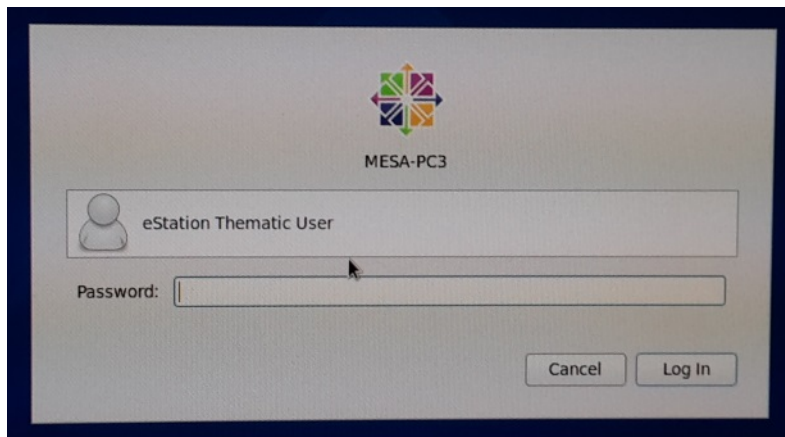


Figure 2: Login Password

Please note that according to the language of the installation, the keyboard is also assigned as 'QWERTY' or 'AZERTY'. It might happen that the password is not accepted because the keyboard you are using does not correspond to the Language settings. In this case, change the 'Language' by clicking on the menu at the bottom line of the monitor (see Figure 3).

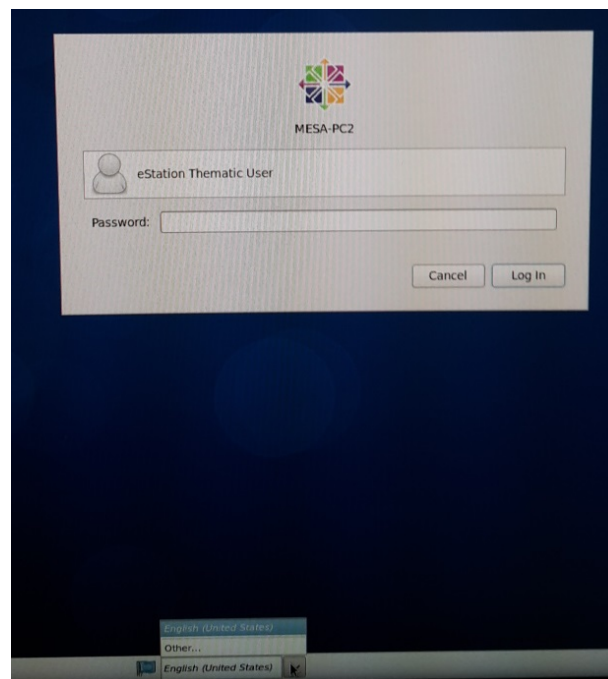


Figure 3: Change language at login

2.2 REMOTE ACCESS TO THE E STATION

The eStation 2.0 is meant to be connected to a LAN in the beneficiary Institution, in order to allow the access to its datasets and its web-client interface from other computers. The configuration of the IP addresses in order to connect to a LAN is done in the MESA Station SW Installation Manual (RD-2) in paragraph 8.1.

In alternative, a desktop or laptop can be added to the MESA LAN, by connecting it to the provided switch. In paragraph 2.2.1 we describe how this operation can be done on Windows 7 machines. This operation varies according to the OS, and might require the assistance of your System Administrator.

Once the external PC and the MESA computers (PC1, PC2 and PC3) are on the same network, it is possible to access the eStation datasets (see 2.2.2) and the eStation GUI (see 2.2.3) from the external computer.

2.2.1 Connecting a Windows PC to the MESA LAN (optional)

If a Windows computer is connected to the switch of the MESA Station, its configuration has to be modified in order to 'manually' assign an IP (v4) in the same 'Subnet' as the other three computers. In Figure 4 an example of configuration (for Windows 7) is shown, and an IP equal to 192.168.0.8 is assigned to the machine.

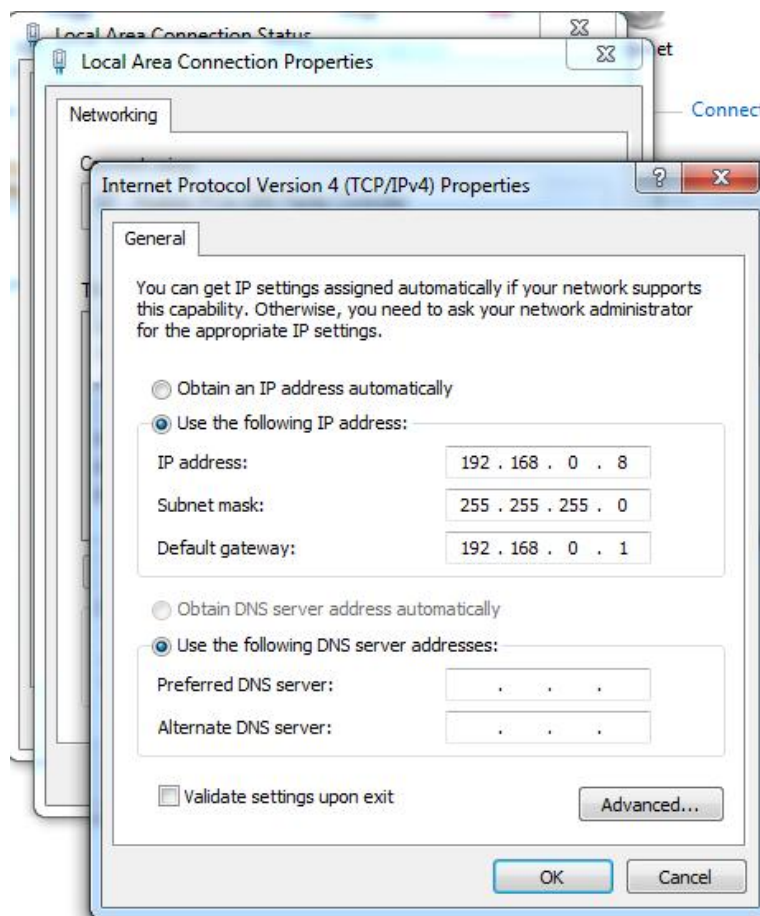


Figure 4: Network configuration in Windows 7

The above configuration allows having the additional computer on the same network as MESA PCs, which can therefore be accessed through their IP address. In order to access by using the hostname (e.g. MESA-PC2), the 'hosts' file in directory 'Windows\System32\drivers\etc\' has to be edited to define the correspondence between IPs and hostnames. After this operation, the PC has normally to be rebooted.

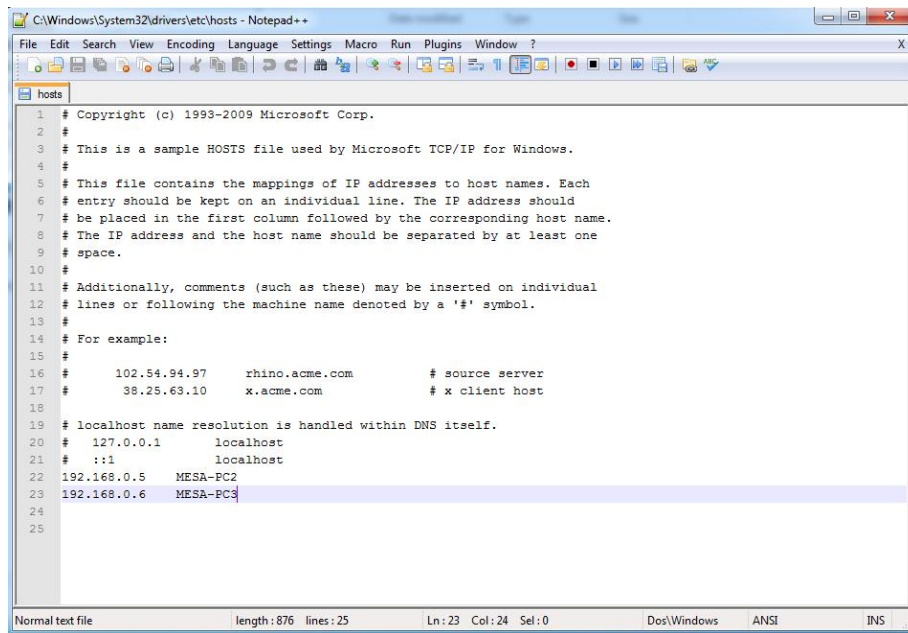


Figure 5: Modify the host file in Windows 7

2.2.2 Access the eStation dataset via ftp.

Once the eStation PCs are accessible on a LAN, you can use an ftp client to browse the eStation directories; in the following we use – as an example – the Filezilla tool (see Figure 6).

For the connection, you have to enter the following parameters:

Host: either the machine IP or the hostname²

Username: 'adminuser' or 'analyst'

Password: 'mesadmin' or 'mesa2015'

Port: 22³

Note that in the example of Figure 6 the IP 192.168.0.5 is indicated, and that has to be changed according to the real IP of the machine (normally, 192.168.0.152 for PC2 and 192.168.0.153 for PC3).

² The hostname will work, ONLY if configured on the local machine – see 2.2.1.

³ For sftp connections, you can either indicate the host preceded by the stfp:// string, or port=22

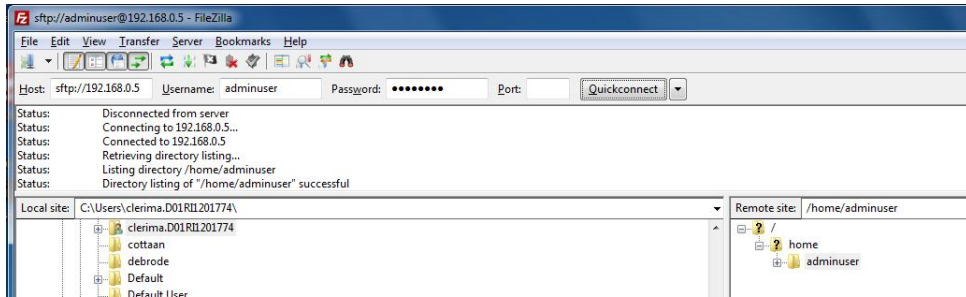


Figure 6: Filezilla to access eStation data

At the first connection to the machine, the system will raise the message of ‘Unknown host key’ (see Figure 7): click on OK.

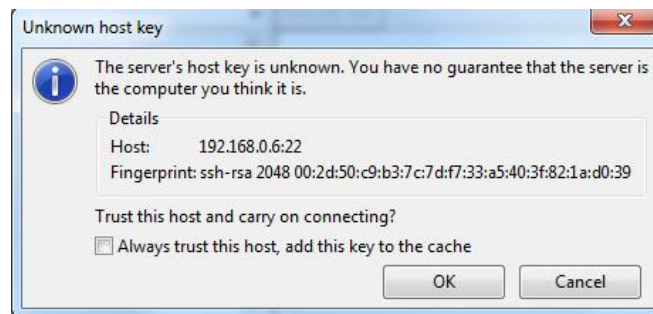


Figure 7: Confirm host key message

At this point, you will see in Filezilla the right panel, the eStation-PC2 filesystem, and the home directory of the User you have indicated for the connection. You can navigate to a different directory (e.g. /data/processing/) and copy locally the desired files.

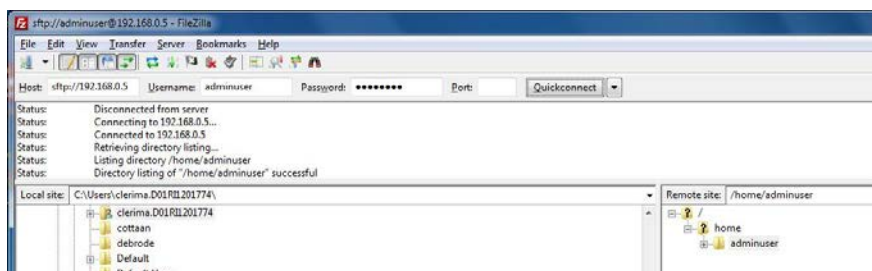


Figure 8: eStation accessed in Filezilla

2.2.3 Access the eStation GUI from a browser.

In order to see the eStation graphical interface from the Windows PC, it is enough to open a browser (Internet Explorer in our example) and enter the machine IP or hostname².

The result of the connection is shown in Figure 9: you can navigate through the different tabs (Acquisition, Processing and so on) and perform the same actions as when you are operating directly on the machine. An option is also to open in the browser two different tabs, for having access on PC2 and PC3 at the same time.

This remote operation option is indeed powerful and risky at the same time, as you can remotely stop the services (Acquisition and Processing) and de-activate the server data processing: we therefore

recommend to be especially careful, especially when remotely accessing the computer performing the processing (PC2, in Nominal Mode).

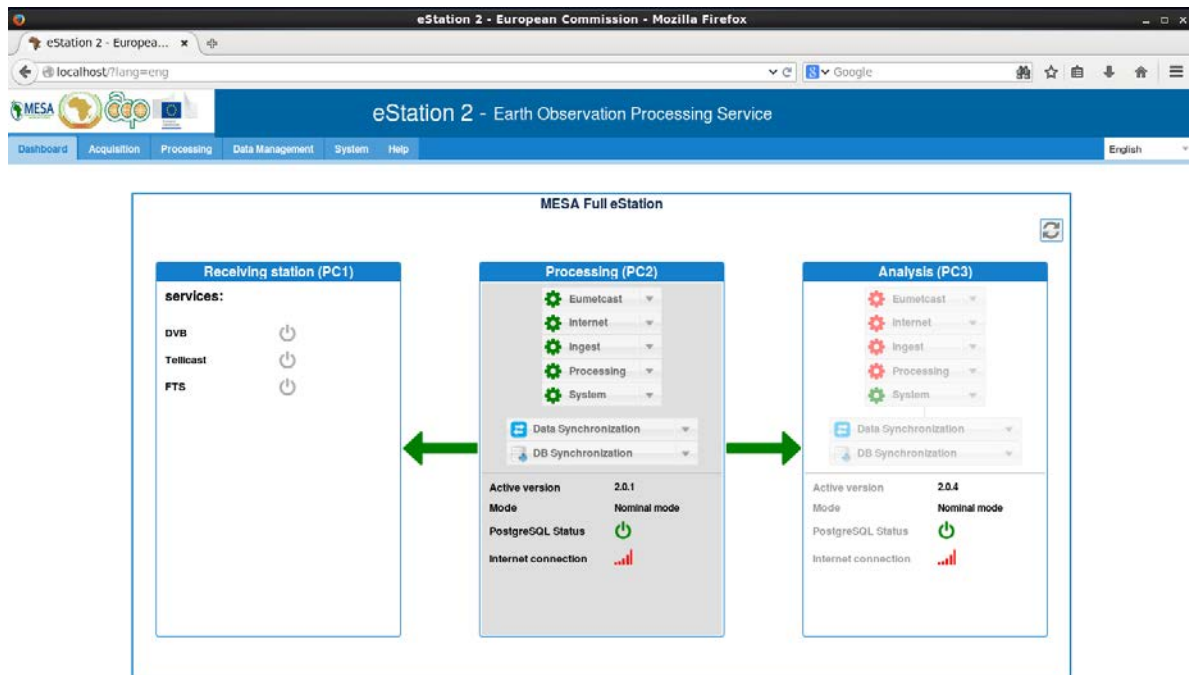


Figure 9: eStation GUI from a remote PC.

2.2.4 Connection by remote shell (ssh)

Both PCs are equipped with the **ssh** protocol (part of CentOS). Therefore, any host connected to the same network as the eStation will be able to remotely access it, to open a 'terminal' session and to type commands, as if the user was physically connected to the console.

Note that a software client, such as PuTTY (directly and freely downloadable from the internet) is required. Also, if there is a firewall between your host and the eStation the ssh connection will probably not work. In such case ask assistance to your system administrator.

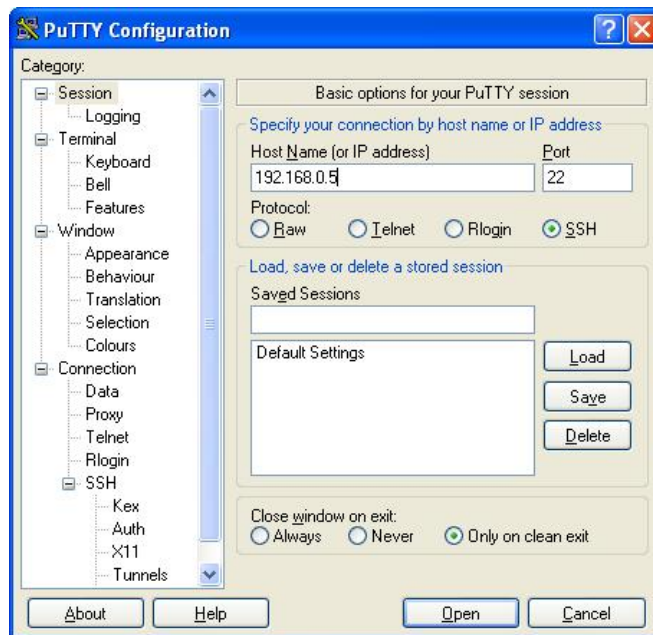


Figure 10: Example of PuTTY connection screen

In PuTTY – see *Figure 10* - you have to enter the same parameters as for the ftp connection, i.e. the IP of the machine (which should correspond to the real one – do not consider the 192.168.0.5 that is only for example) and the port 22. Again, a message notifying the ‘new host key’ might be displayed, and has to be acknowledged (type ‘YES’ – see *Figure 11*).



Figure 11: PuTTY first connection

Afterwards, the Username and password are requested, before opening the ‘terminal’ session on the machine.

3. eSTATION MAINTENANCE

This section describes the main operations to perform for the system maintenance, including monitoring the disk space, perform diagnostic and recovery operations.

3.1 MONITORING THE STATUS OF THE SYSTEM

3.1.1 Monitoring the status of the disk

The MESA eStation 2.0 is installed on DELL Optiflex 9020 machine equipped with two 1 TB disks, in RAID1 redundant configuration.

The User can check this configuration in several ways, according to the status of functionality of the system.

Check the disk status by using CentOS tools

CentOS makes available a 'Disk Usage Analyser' tool from the Applications -> System Tools menu (see Figure 12).

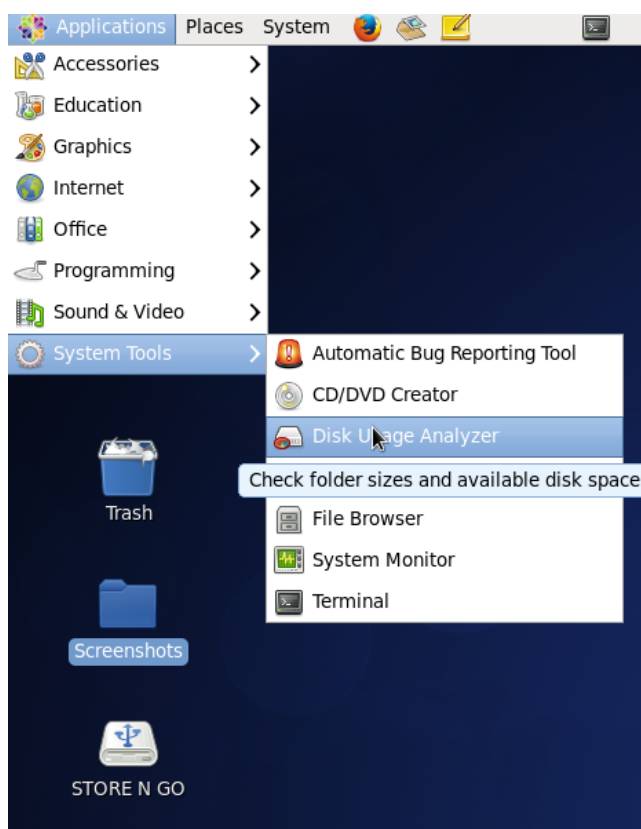


Figure 12: Disk Analyser Tool

Once started, it visualizes the total filesystem capacity and the Usage (see Figure 13) and allows additional operations like the scanning a specific directory, to see the total size it occupies in the filesystem.

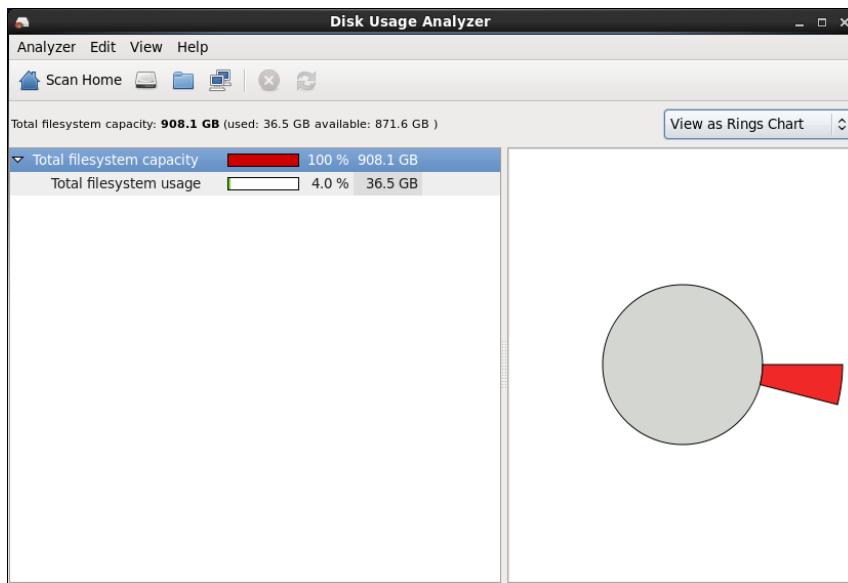


Figure 13: Disk Usage Analyser View

Check the status of RAID1 disk

The MESA computers host two HDs of 1 Tb in RAID1 configuration. The checking and administration of the RAID 1 is described in MESA Station Administration and Maintenance Manual (RD-3) in paragraph 7.3.

3.1.2 Monitoring the status of the System

The CentOS installation offers a 'System Monitor' Utility that can be accessed from the main Menu (Applications -> System Tools -> System Monitor) as displayed in Figure 14.

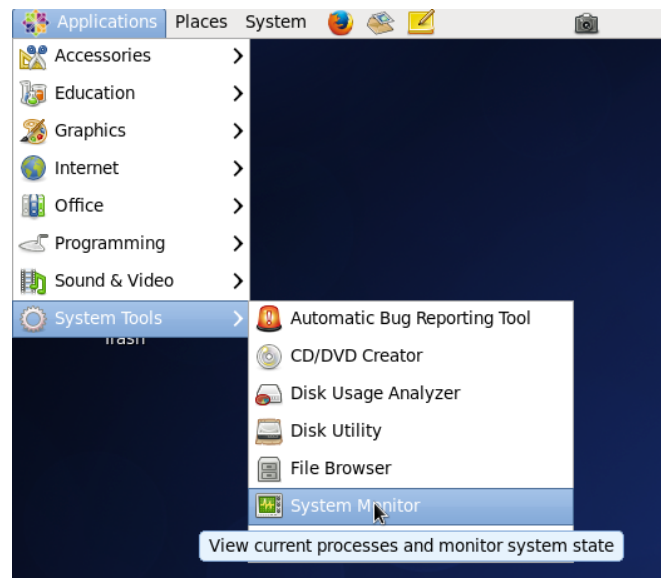


Figure 14: System Monitor utility

The application is composed by 4 tabs, starting from the 'System' one, which displays the name of the computer ('hostname', which is MESA-PC3 in case of Figure 15).

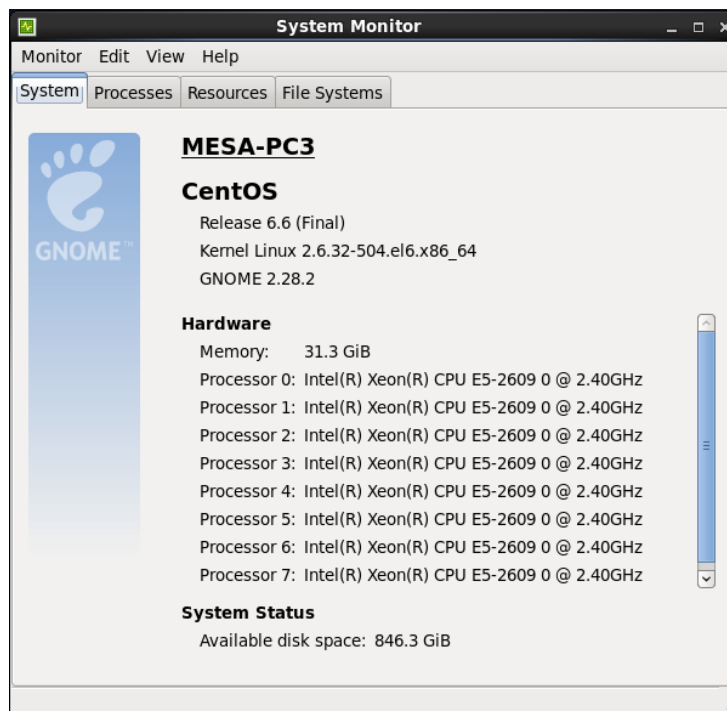


Figure 15: Monitor Tab - System Description

The most relevant tab is 'Resources' (see Figure 16), where the status of usage of CPUs, including a short history of usage, is displayed, together with the history of memory use and 'Swap' operations

and the Network History, which reports on the amount of data exchanged on the LAN, both in reception and uploading.

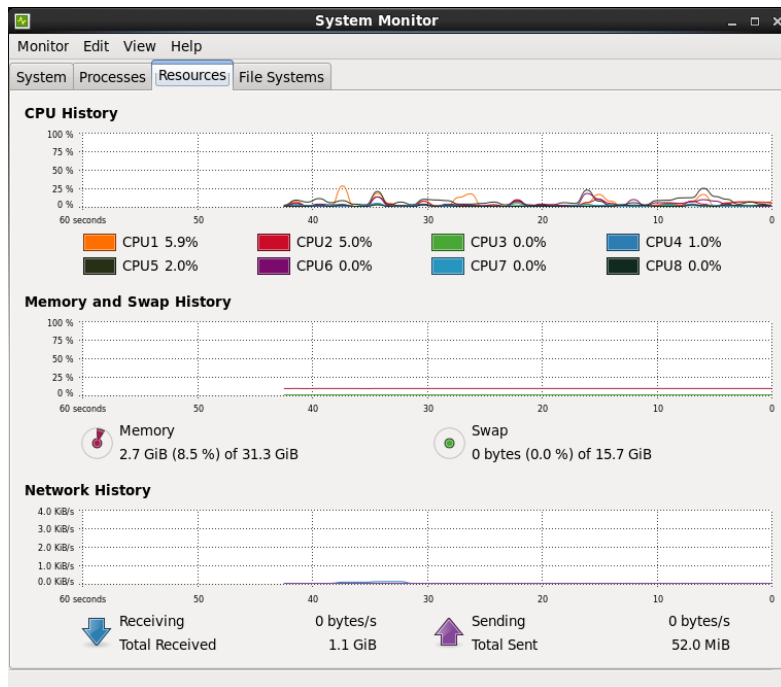


Figure 16: Monitoring System Resources

Another relevant tab is the 'Processes' one, where the list of active processes, their status, their usage of computer resources (%CPU and Memory) is displayed, alike in the Windows 'Task Manager'.

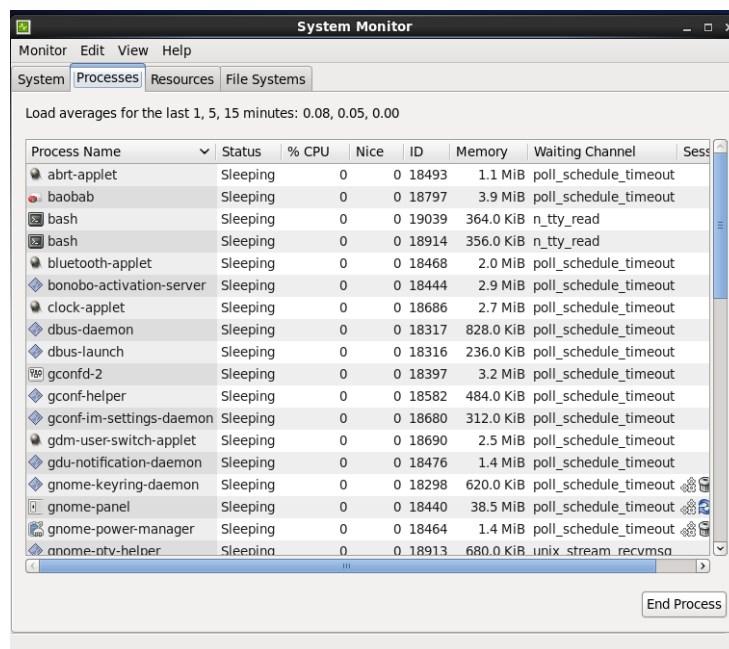


Figure 17: System Processes

3.1.3 Checking the Log files

All eStation2 logfile are created in the /eStation2/log directory, and they are not touched by the upgrade to a newest version of the system. We have implemented for all python modules a logfile named according to the position of the .py file, and its name; e.g. the *apps.acquisition.ingestion.log* is created by the module *ingestion.py*, located under the eStation 2.0 base directory, in *./apps/acquisition* directory (i.e. the file */var/www/eStation2/apps/acquisition/ingestion.py*).

The log mechanism generates and keeps 4 versioned files for each of python module: the latest messages are in a file named with the suffix *.log* (e.g. *apps.acquisition.ingestion.log*), while the older ones in files having a suffix from *.1* to *.3* (*.3* is the oldest file).

Note that the log files generated by the Services, or related to a specific acquisition or processing trigger, can be accessed in an easier and more direct way through the GUI (see User Manual Chapter 3); the access to the logfile directory is therefore suggested when the GUI is not working, or when the User would like to see the full list of logfile, including the older ones (*'log.1'* to *'log.3'*)

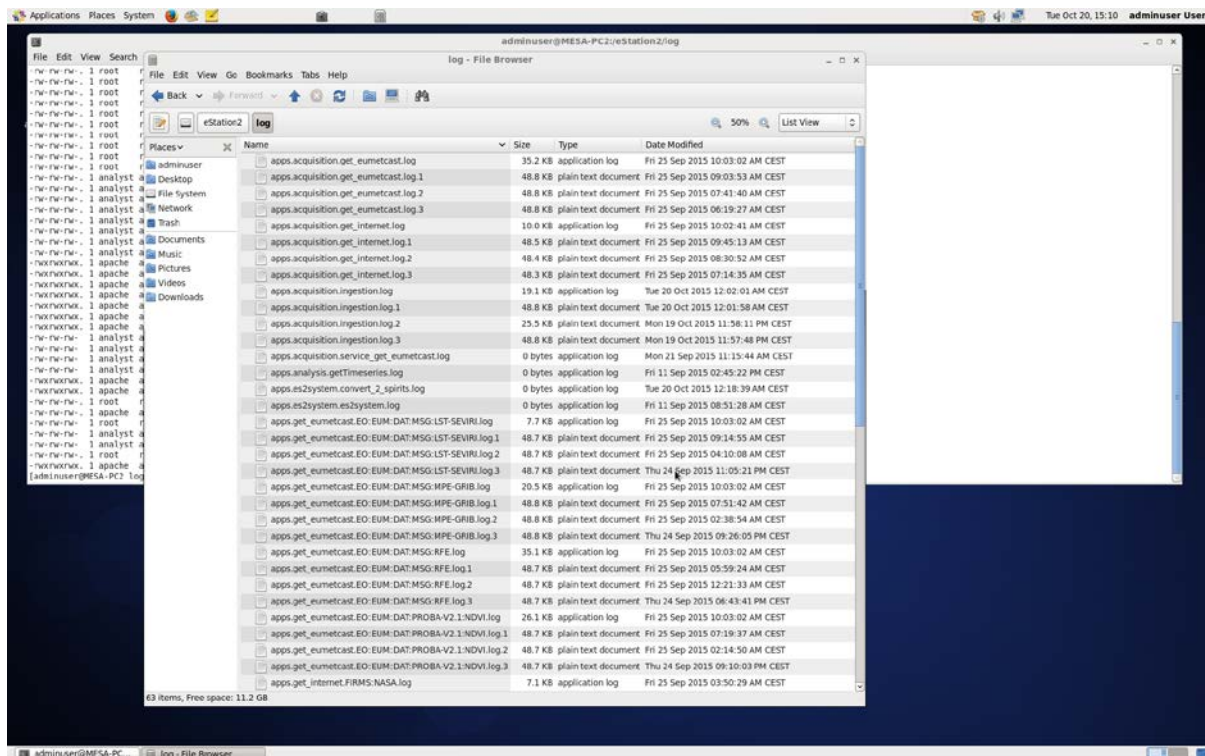


Figure 18: Logfiles of eStation 2.0

3.2 IMPORTING DATASETS

On the eStation 2.0, a part from the 'Get' services, two mechanisms exist in order to populate the local filesystem with EO datasets, namely:

1. Importing 'Historical Archives', which is mainly meant to be done after installation.
2. Importing 'Request Archives', which will copy to the local machine the missing data after generation of a 'User Request'.

Here below a table that summarizes the properties, and the differences, between the two mechanisms.

Type of Archive	Goal	Data Type	How to Install	Notes
Historical	Populate the eStation after installation	The archives are in form of directories containing GTIFF files, with file naming like MESA_JRC_...tif	By defining the archive path under 'System' tab and running the python script (see 2.4.1. below)	<p>1. Re-projection is done from the original mapset to User-defined mapset (typically from Continent to Regional extent).</p> <p>2. Selection of relevant data for the User is done, by considering the products/subproducts activated.</p>
Request	Complete the missing data	The archives are self-extracting .bsx files containing the eStation 2.0 products in GTIFF format.	By double-clicking on the .bsx file (see below).	<p>1. NO Re-projection is done: archives are created on the mapset requested by the User.</p> <p>2. NO selection is done; all data are copied to local filesystem.</p>

3.2.1 Historical archives import

As described in the User Manual in paragraph 2.4.9, after installation of the eStation the local disk has to be populated with raster file generate by JRC. These archives are made available on an external medium (usually an USB external disk) that as to be connected to the PC2.

Therefore, as first operation you have to be logged in as 'adminuser' and connect the external disk; at this point, normally the disk is automatically recognized, mounted, and will appear on the 'Desktop' of the User (see Figure 19). In our example, the disk is labelled 'MESA-2' and you see this name on the desktop. In particular, we have copied the archives in the 'Archives2.0' directory.

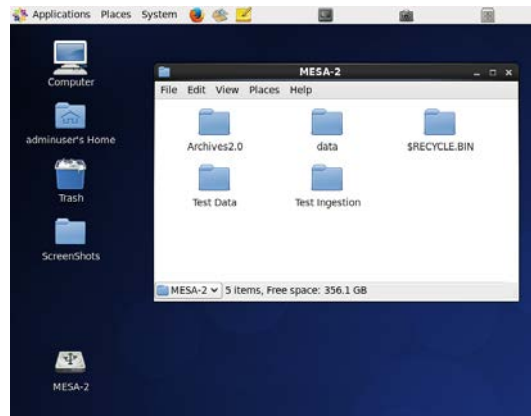


Figure 19: Archives disk connected.

As a consequence, the files are visible in the filesystem under:

/media/MESA-2/Archives2.0

You have to enter this location in the 'System' tab, as 'Archive directory' and push the 'SAVE' button to have the eStation record this variable (see Figure 20).

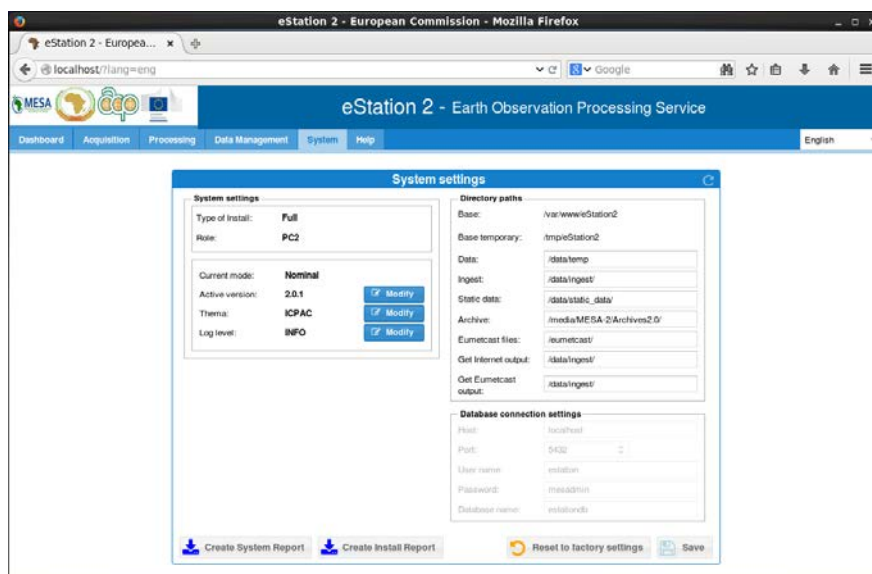


Figure 20: Update location of the Historical Archives.

Now you have to launch the python routine to execute the ingestion; open a terminal and change your current directory to: `/var/www/eStation/apps/tools` (see Figure 21)

```
adminuser@MESA-PC2:~
File Edit View Search Terminal Help
[adminuser@MESA-PC2 ~]$ cd /var/www/eStation2/apps/tools/
```

Figure 21: Change dir for Ingestion

Enter the command: `python ingest_historical_archives.py` (see Figure 22)

```
adminuser@MESA-PC2:/var/www/eStation2/apps/tools
File Edit View Search Terminal Help
[adminuser@MESA-PC2 ~]$ cd /var/www/eStation2/apps/tools/
[adminuser@MESA-PC2 tools]$ python ingest_historical_archives.py
```

Figure 22: Start the ingestion

The 'ingest' routine starts scanning the defined directory, and does a local copy, optionally with reprojection to the local ROI, of all subproducts that are relevant for the Thema defined. It is therefore mandatory to execute this operation after having properly set the THEMA (see paragraph 3.9 of User Manual).

The routine reports the images it is processing (see Figure 23). The duration of the process varies according to the number of files to be treated, and goes from few minutes to an hour, or more.

```
adminuser@MESA-PC2:/var/www/eStation2/apps/tools
File Edit View Search Terminal Help
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20150511_CHIRP-Africa-5km_2.0.tif
2015-12-17 13:47:09 - estacion2.apps.acquisition.ingestion - INFO - Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20130921_CHIRP-Africa-5km_2.0.tif
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20130921_CHIRP-Africa-5km_2.0.tif
2015-12-17 13:47:09 - estacion2.apps.acquisition.ingestion - INFO - Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20110321_CHIRP-Africa-5km_2.0.tif
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20110321_CHIRP-Africa-5km_2.0.tif
2015-12-17 13:47:10 - estacion2.apps.acquisition.ingestion - INFO - Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20140711_CHIRP-Africa-5km_2.0.tif
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20140711_CHIRP-Africa-5km_2.0.tif
2015-12-17 13:47:10 - estacion2.apps.acquisition.ingestion - INFO - Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20120121_CHIRP-Africa-5km_2.0.tif
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20120121_CHIRP-Africa-5km_2.0.tif
2015-12-17 13:47:10 - estacion2.apps.acquisition.ingestion - INFO - Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20121201_CHIRP-Africa-5km_2.0.tif
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20121201_CHIRP-Africa-5km_2.0.tif
2015-12-17 13:47:10 - estacion2.apps.acquisition.ingestion - INFO - Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20150521_CHIRP-Africa-5km_2.0.tif
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20150521_CHIRP-Africa-5km_2.0.tif
2015-12-17 13:47:10 - estacion2.apps.acquisition.ingestion - INFO - Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20130701_CHIRP-Africa-5km_2.0.tif
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20130701_CHIRP-Africa-5km_2.0.tif
2015-12-17 13:47:10 - estacion2.apps.acquisition.ingestion - INFO - Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20121111_CHIRP-Africa-5km_2.0.tif
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20121111_CHIRP-Africa-5km_2.0.tif
2015-12-17 13:47:11 - estacion2.apps.acquisition.ingestion - INFO - Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20130801_CHIRP-Africa-5km_2.0.tif
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20130801_CHIRP-Africa-5km_2.0.tif
2015-12-17 13:47:11 - estacion2.apps.acquisition.ingestion - INFO - Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20140501_CHIRP-Africa-5km_2.0.tif
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20140501_CHIRP-Africa-5km_2.0.tif
2015-12-17 13:47:11 - estacion2.apps.acquisition.ingestion - INFO - Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20140911_CHIRP-Africa-5km_2.0.tif
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20140911_CHIRP-Africa-5km_2.0.tif
2015-12-17 13:47:11 - estacion2.apps.acquisition.ingestion - INFO - Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20120601_CHIRP-Africa-5km_2.0.tif
INFO:estacion2.apps.acquisition.ingestion:Entering routine ingest file archive for file /media/MESA-2/Archives2.0/chirps-dekad/10dperc/MESA_JRC_chirps-dekad_10dperc_20120601_CHIRP-Africa-5km_2.0.tif
```

Figure 23: Historical Archives ingestion finished

3.2.2 Archives from User Request

The procedure for completing the local datasets with 'User archives' is composed by three steps:

1. Creating a 'User Request' and send it to RICs/JRC
2. Creating a 'User Archive' (done by RICs/JRC)
3. Installing the 'User Archive'

[Creating a 'User Request'](#)

The User can generate requests for completing the existing datasets, as explained in the User Guide at paragraph 3.7. These requests are saved in a text file (.req), containing the description of the dataset, and namely the files that are missing: this file is used to generate an 'ad-hoc' archive.

The requests can be generated at three different levels, and the .req file is named accordingly:

a) Product/version: all existing mapsets and subproducts are considered

File name: <product>_<version>_all_enabled_mapset_<YYYY-MM-DD HHMM>

e.g.: vgt-dmp_V1.0_all_enabled_mapsets_2016-02-23 1200.req

b) Product/version/mapset: all subproducts are considered.

File name: <product>_<version>_<mapset>_all_enabled_datasets_<YYYY-MM-DD HHMM>

e.g.: vgt-fapar_V1.4_SPOTV-Africa-1km_all_enabled_datasets_2016-02-23 1200.req

c) Product/version/mapset/subproduct: only the specified subproduct is considered.

File name: <product>_<version>_<mapset>_<subproduct>_<YYYY-MM-DD HHMM>

e.g.: vgt-fapar_V1.4_SPOTV-Africa-1km_dmp_2016-02-23 1200.req


```

{ "product": "vgt-fapar",
  "productmapsets": [
    {
      "mapsetcode": "SPOTV-Africa-1km",
      "mapsetdatasets": [
        {
          "missing": [
            {
              "from_start": true,
              "info": {
                "firstdate": "2015-05-11",
                "intervals": [
                  {
                    "fromdate": "2015-05-11",
                    "intervalpercentage": 93.10344827586206,
                    "intervaltype": "present",
                    "missing": false,
                    "todate": "2016-02-01",
                    "totfiles": 27
                  },
                  other intervals
                ],
                "lastdate": "2016-02-21",
                "missingfiles": 2,
                "totfiles": 29
              },
              "mapset": "SPOTV-Africa-1km",
              "mapset_data": {
                full description of the mapset
              },
              "product": "vgt-fapar",
              "subproduct": "fapar",
              "to_end": true,
              "version": "V1.4"
            }
          ],
          "product_type": "",
          "subproductcode": "fapar"
        }
      ]
    }
  ],
  "version": "V1.4"
}

```

Figure 24: Structure of request file

The .req file is meant to be sent to another installation of eStation 2.0 (namely at JRC or one of the RICs) for the creation of the ad-hoc archives. The structure of the .req file, for the case c. above, is displayed in Figure 24; it contains all information for correctly identifying the dataset (i.e. product/version/mapset/subproduct) or datasets, and the existing/missing files. In the cases a. and b., the structure remains the same, but more than one 'productmapsets' and 'mapsetdatasets' exist.

Creating a 'User Archive'

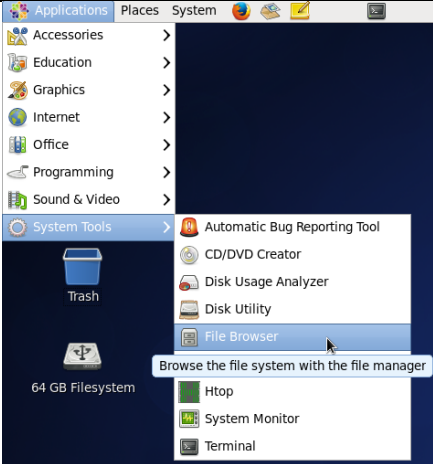
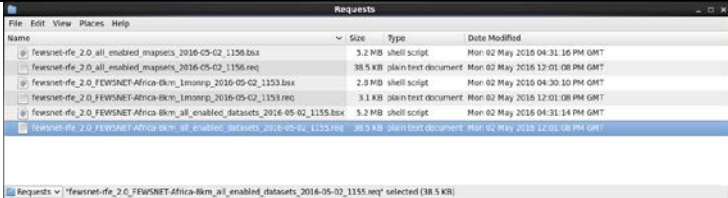
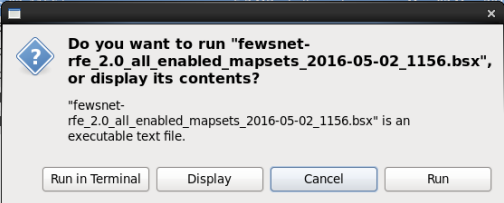
On the basis of the received request, the eStation server generates an archive, which is a self-extracting compressed file, containing all missing images (GTIFF files). The archive is created according to the following rules:

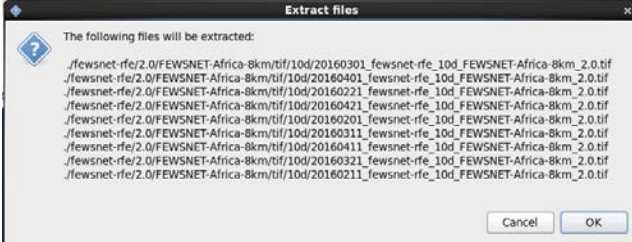
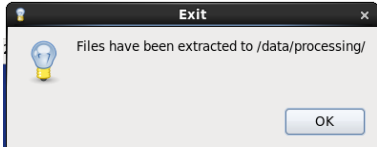
1. The correct product/version/mapset/subproducts are considered only: in case a 'larger' mapset exists (e.g. 'SPOTV-Africa-1km' rather than 'SPOTV-IGAD-1km'), re-projection is done to the User mapset; this is to limit the size of the archive to be transmitted.

2. In case some files are missing on the source machine as well (e.g. the most recent products not yet generated), a warning message is raised, but the generation of the archive continues. Therefore, the 'ad-hoc' archive is produced and delivered also if it does not fully satisfy the original request.

Installing a 'User Archive'

The 'User Archive' (.bsx file) has to be copied to a storage medium, like an USB memory stick or HDD. Then, the following procedure applies:

Log to PC2 as Thematic User	
Connect the storage device to PC2	
Open the file browser	
Locate the 'bsx' file containing the Archive generated from a request. In this example, the file is in directory 'Requests' on an external key	
Double click on the .bsx file name.	
A message as the one on the right figure appears, Click on Run	

<p>A list of files that will be copied to the PC2 appears: Click on OK.</p> <p>NOTE: The copy of the files can take some time, depending on their overall size.</p>	 <p>The following files will be extracted:</p> <pre> ./fewsnet-rfe/2.0/FEWSNET-Africa-8km/tif/10d/20160301_fewsnet-rfe_10d_FEWSNET-Africa-8km_2.0.tif ./fewsnet-rfe/2.0/FEWSNET-Africa-8km/tif/10d/20160401_fewsnet-rfe_10d_FEWSNET-Africa-8km_2.0.tif ./fewsnet-rfe/2.0/FEWSNET-Africa-8km/tif/10d/20160221_fewsnet-rfe_10d_FEWSNET-Africa-8km_2.0.tif ./fewsnet-rfe/2.0/FEWSNET-Africa-8km/tif/10d/20160421_fewsnet-rfe_10d_FEWSNET-Africa-8km_2.0.tif ./fewsnet-rfe/2.0/FEWSNET-Africa-8km/tif/10d/20160201_fewsnet-rfe_10d_FEWSNET-Africa-8km_2.0.tif ./fewsnet-rfe/2.0/FEWSNET-Africa-8km/tif/10d/20160311_fewsnet-rfe_10d_FEWSNET-Africa-8km_2.0.tif ./fewsnet-rfe/2.0/FEWSNET-Africa-8km/tif/10d/20160411_fewsnet-rfe_10d_FEWSNET-Africa-8km_2.0.tif ./fewsnet-rfe/2.0/FEWSNET-Africa-8km/tif/10d/20160321_fewsnet-rfe_10d_FEWSNET-Africa-8km_2.0.tif ./fewsnet-rfe/2.0/FEWSNET-Africa-8km/tif/10d/20160211_fewsnet-rfe_10d_FEWSNET-Africa-8km_2.0.tif </pre> <p>Buttons: Cancel, OK</p>
<p>At the end of the copy, a message as the one on the figure below appears: Click on OK.</p>	 <p>Files have been extracted to /data/processing/</p> <p>Button: OK</p>

3.3 PERFORMING VERSIONS UPGRADES

The three packages composing the eStation application (see 3.3.2) are originally installed on the MESA station in version 2.0.1. During the MESA project, the application will be upgraded to newer version (starting from 2.1.0) and the packages disseminated through EUMETCast.

The MESA station implement an automatic mechanism the upgrade the full installation, which is described in the MESA Station Administration and Maintenance Manual (RD-3) in paragraph 5.3

Once the eStation application is upgraded to a newer version, this is automatically considered for operation.

If the User intends to switch between versions (not recommended), he should access the System tab of the eStation 2.0 GUI, as described in the User Manual (RD-1) in paragraph 3.10.

From version 2.1.2, it is not possible anymore to change the version from the user interface. If you still want to change from version 2.1.2 to a lower version already installed, follow these steps (not recommended):

1. Rename the `httpd.conf` (used for version 2.1.2) to `httpd_2.1.2.conf`

```
sudo mv /usr/local/src/tas/eStation_wsgi_srv/httpd.conf
/usr/local/src/tas/eStation_wsgi_srv/httpd_2.1.2.conf
```

2. Rename the `httpd_before_2.1.2.conf` to `httpd.conf`

```
sudo mv /usr/local/src/tas/eStation_wsgi_srv/httpd_pre_2.1.2.conf
/usr/local/src/tas/eStation_wsgi_srv/httpd.conf
```

3. Change symbolic link `/var/www/eStation2` to the version you want to change to

```
rm /var/www/eStation2
ln -s /var/www/eStation2-<version_to_change_to> eStation2
```

4. Remove proxy lines from `/eStation2/settings/user_settings.ini`

```
gedit /eStation2/settings/user_settings.ini
```

Remove the following lines and save the file:

```
proxy_host =
proxy_port =
proxy_user =
proxy_userpwd =
```

5. Restart apache

```
sudo /etc/init.d/tas_estation_apached restart
```

If you want to change from a lower version than 2.1.2 to a version 2.1.2 already installed, follow these steps (not recommended):

1. Rename the httpd.conf to httpd_pre_2.1.2.conf

```
sudo mv /usr/local/src/tas/eStation_wsgi_srv/httpd.conf  
/usr/local/src/tas/eStation_wsgi_srv/httpd_pre_2.1.2.conf
```

2. Rename the httpd_2.1.2.conf to httpd.conf

```
sudo mv /usr/local/src/tas/eStation_wsgi_srv/httpd_2.1.2.conf  
/usr/local/src/tas/eStation_wsgi_srv/httpd.conf
```

3. Change symbolic link /var/www/eStation2 to version 2.1.2.

```
rm /var/www/eStation2  
ln -s /var/www/eStation2-2.1.2 eStation2
```

4. Add the proxy lines to /eStation2/settings/user_settings.ini

```
gedit /eStation2/settings/user_settings.ini
```

Add the following lines and save the file:

```
proxy_host =  
proxy_port =  
proxy_user =  
proxy_userpwd =
```

5. Restart apache

```
sudo /etc/init.d/tas_estation_apached restart
```

3.4 RECOVERY MODE (FAILOVER/FAILBACK)

The Recovery Mode is managed on the MESA station through a general mechanism for managing the 3 PCs implemented by the Supply Contractor (TPZF/TAS). Please refer to the MESA Station Administration and Maintenance Manual (RD-3) paragraph 6.6 (Failover) and 6.7 (Failback).

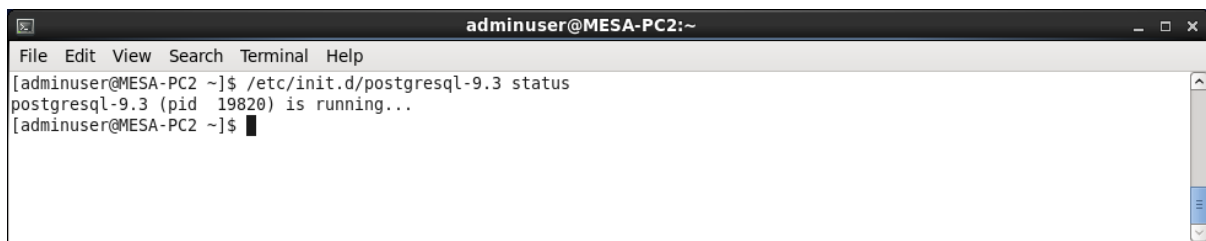
3.5 MANAGING THE POSTGRESQL DATABASE

The current section describes how to check the postgresql server is running, how access the 'estationdb' database, and how to restore a previous copy of the database in case of corruption.

3.5.1 Check the status of the postgresql server

The postgresql 9.3 server is installed on eStation 2.0, and its status can be checked from the terminal by typing the command:

```
/etc/init.d/postgresql-9.3 status
```



```
adminuser@MESA-PC2:~  
File Edit View Search Terminal Help  
[adminuser@MESA-PC2 ~]$ /etc/init.d/postgresql-9.3 status  
postgresql-9.3 (pid 19820) is running...  
[adminuser@MESA-PC2 ~]$
```

Figure 25: Check postgresql status

The expected result is represented in Figure 25: *'postgresql-9.3 (pid NNNNN) is running...'*

To stop the server, a similar command is executed (see Figure 26): note that 'sudo' should precede the command, as root rights are required.

```
sudo /etc/init.d/postgresql-9.3 stop
```



```
adminuser@MESA-PC2:~  
File Edit View Search Terminal Help  
[adminuser@MESA-PC2 ~]$ sudo /etc/init.d/postgresql-9.3 stop  
Stopping postgresql-9.3 service: [ OK ]  
[adminuser@MESA-PC2 ~]$
```

Figure 26: Stop postgresql

Similarly, for starting (see Figure 27) or restarting the server, the following commands have to be typed:

```
sudo /etc/init.d/postgresql-9.3 start
```

```
sudo /etc/init.d/postgresql-9.3 restart
```



Figure 27: Start postgresql

3.5.2 Visualize the database tables (pgAdminIII)

On the eStation 2.0 the pgAdminIII tool is installed for accessing the database, verifying its contents and modifying it (only for Advanced Users). To run the tool, go to the Applications -> Programming -> pgAdminIII entry of the CentOS menu (see Figure 28).

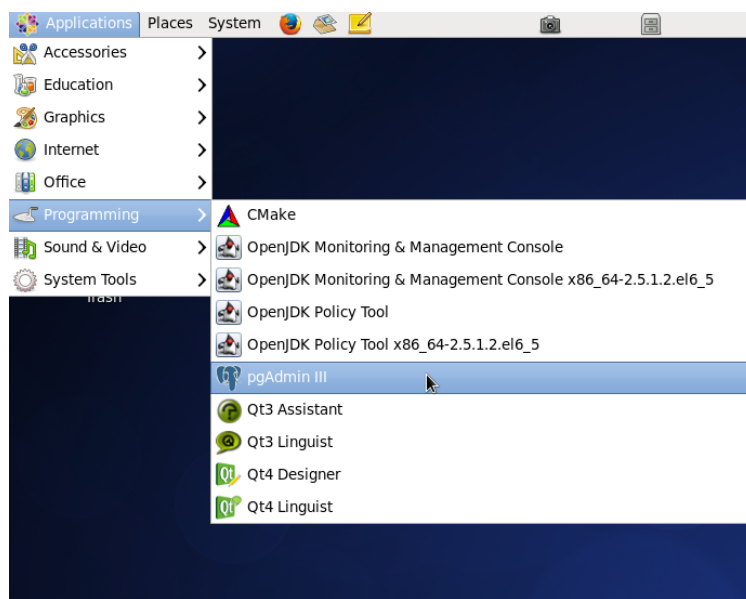
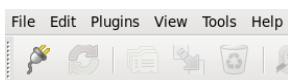


Figure 28: Launch pgAdmin III

At the first run, the connection to the estationdb has to be established. Click on the 'plug' icon displayed just under the 'File' menu entry (top-left corner):



An interface will open (see Figure 29) for entering the parameters for the connection.

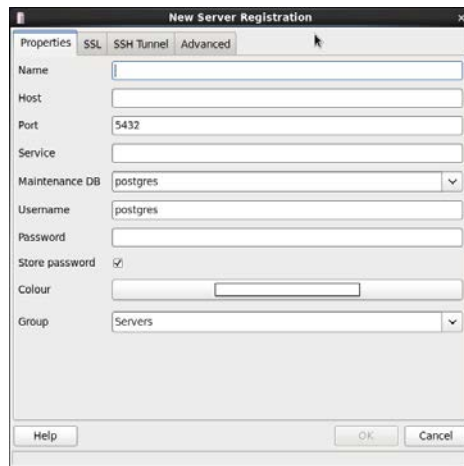


Figure 29: new connection in pgAdminIII

The variables to be used are shown in Figure 30, and repeated here below:

Name: estationdb
 Host: localhost
 Maintenance DB: estationdb
 Username: adminuser (recommended – also estation accepted)
 Password: mesadmin

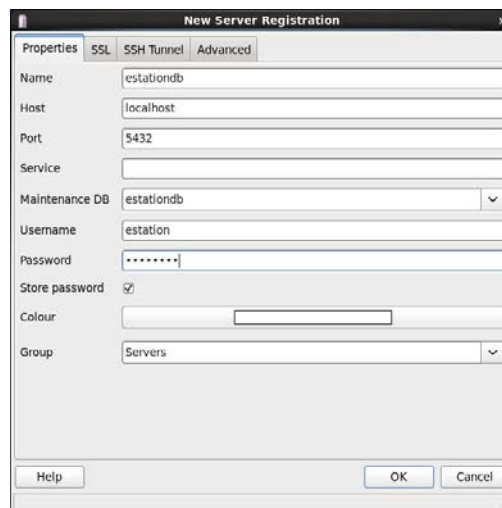


Figure 30: connection parameters for estationdb

When all variables are entered, you can click on the 'OK' button, and a message proposing to store the typed password is displayed (see Figure 31). Click on 'OK'.

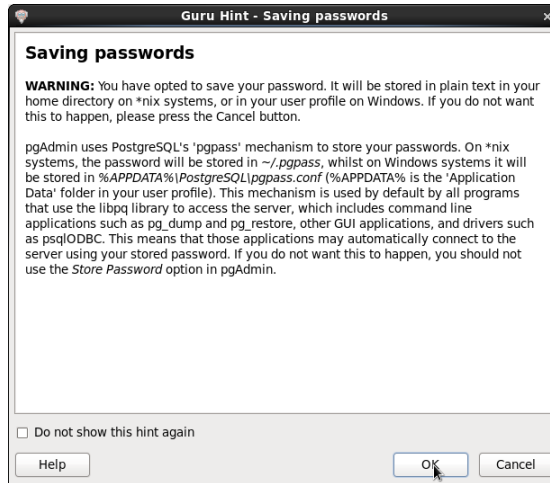


Figure 31: pgAdminIII - save password

At this point, the connection with the local estationdb is established, and you will be presented the interface displayed in Figure 32.

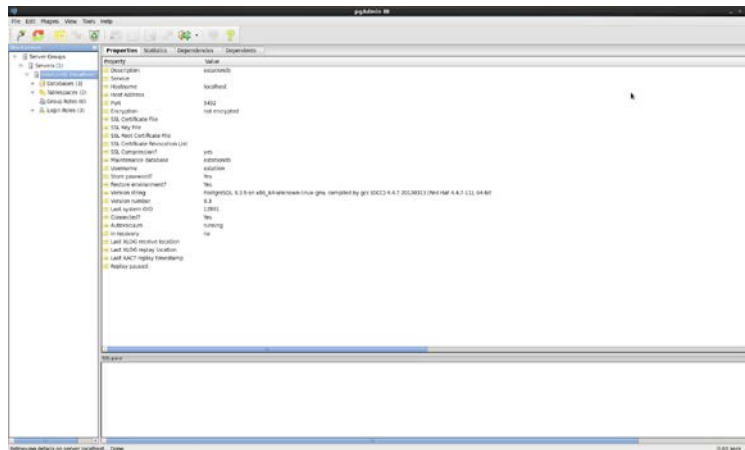


Figure 32: estationdb overview in pgAdminIII

You can expand the list of servers by clicking on the '+' symbol to its left, and see the 'estationdb' database. Then you can access its schemas and tables by navigating the tree in the left control panel (see Figure 33).

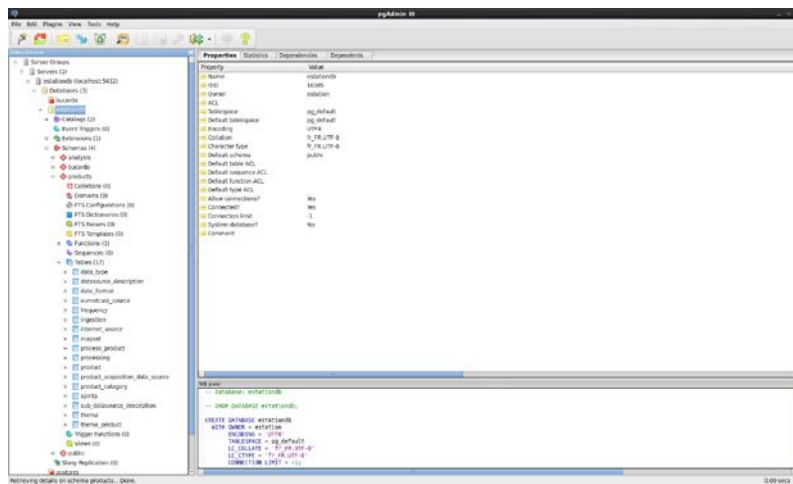


Figure 33: estationdb tables

Note that at the first launch of pgAdminIII the notification of missing server instrumentation (see Figure 34) might be displayed. Just click the 'OK' button.

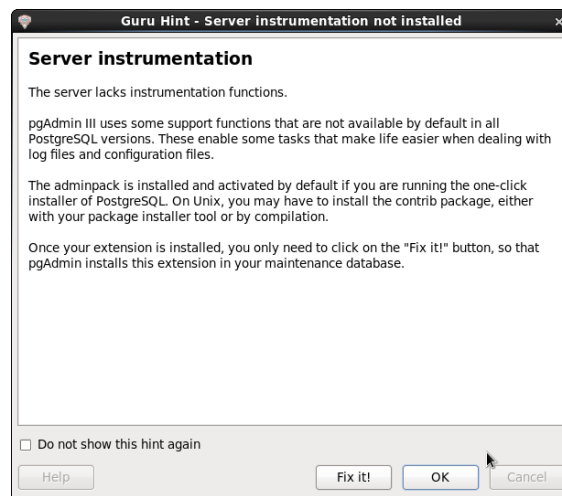


Figure 34: server instrumentation message

3.5.3 Restore a previous copy of the database

The estationdb is dumped every day on the local disk, so that it is possible to restore it in case it is corrupted, e.g., for issues on the HD. This issue was very frequent on AMESD stations, mainly caused by disk failures.

How database is dumped

The system service executes a task that saves – every day at midnight – a local copy of both ‘analysis’ and ‘products’ schema of the database.

The dump is executed in the directory /eStation2/db_dump/

A ‘rotation’ mechanism is implemented in order to keep only some of the saved dumps, and namely:

- a) for previous months: the file of first day of the month
- b) for the current month: the file of first day of the month and the latest two days.

This mechanism limits the disk occupancy and facilitates the management of the dumps in the directory, ensuring that the most recent files are made available, but also some older files. An example of the list of dumps is shown in Figure 35. The files are named with the following convention:

estationdb_<schema>_<date_of_the_dump>-<time_of_the_dump>.sql

```
[adminuser@MESA-PC2 dbInstall]$ ll /eStation2/db_dump
total 23420
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2014-10-20-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2014-11-20-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2014-12-20-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-01-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-02-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-03-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-04-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-05-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-06-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-07-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-08-31-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-09-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-10-01-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-10-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-10-22-17:54:38.sql
-rwxr-xr-x 1 analyst estation 326973 Oct 26 22:07 estationdb_analysis_2015-10-26-22:05:06.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2014-10-20-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2014-11-20-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2014-12-20-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2015-01-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2015-02-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2015-03-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2015-04-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2015-05-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2015-06-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2015-07-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2015-08-31-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2015-09-21-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2015-10-01-17:54:38.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2015-10-26-22:00:56.sql
-rwxr-xr-x 1 analyst estation 1248714 Oct 26 22:08 estationdb_products_2015-10-26-22:05:06.sql
```

Figure 35: list of dumps of estation DB

When to restore the database

Considering the database synchronization that exists between PC2 and PC3, the need to re-install the database from a previous dump is *a rather rare action to be performed*. In case of a major fault of one of the computer, and re-installation of the full application, the Recovery mode procedure described in paragraph 2.6 takes care of replicating the database from the other PC, and no database restore from previous dump is necessary.

Therefore, this action has to be applied only in one of the following cases:

- a) The estationdb has been corrupted on both computers.
- b) The estationdb has been corrupted on one computer that is in Recovery mode, i.e. the other computer is not available.
- c) There is any reason for which some important information previously generated in the database has been deleted on both computers.

How to restore the database

A specific script exists for dropping the contents of a schema (if the schema exists), recreating the schema and populating it with the selected dump. As displayed in Figure 36, the script is named 'db_restore.sh' and located in directory /var/www/eStation2/database/dbInstall.

```
[adminuser@MESA-PC2 dbInstall]$ cd /var/www/eStation2/database/dbInstall/
[adminuser@MESA-PC2 dbInstall]$ ll
total 1644
-rwxr-xr-x 1 adminuser estation 904 Oct 23 21:16 dbdrop.sh
-rwxr-xr-x 1 adminuser estation 1513 Oct 23 21:16 dbinstall.sh
-rwxr-xr-x 1 adminuser estation 1627 Oct 23 21:16 dbrestore.sh
-rwxr-xr-x 1 adminuser estation 1420282 Oct 23 21:16 products_dump_data_only.sql
-rwxr-xr-x 1 adminuser estation 120508 Oct 23 21:16 products_dump_structure_only.sql
-rwxr-xr-x 1 adminuser estation 125573 Oct 23 21:16 products_full_dump.backup
[adminuser@MESA-PC2 dbInstall]$
```

Figure 36: dbrestore.sh script

Therefore the User should open a terminal, and change directory by typing:

```
cd /var/www/eStation2/database/dbInstall
```

Then, once the User has identified the date-time of the dump to restore (2015-10-06-22:05 in our example) enter the command:

```
sudo ./dbrestore.sh <date-time of dump>
```

as shown in Figure 37.

```
[adminuser@MESA-PC2 dbInstall]$ sudo ./dbrestore.sh 2015-10-26-22:05
```

Figure 37: dbrestore script execution

The script is executed, for both 'analysis' and 'products' schema, and the series of insert statements reported on the terminal, as shown in Figure 38.

```
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 23091 1
INSERT 23092 1
INSERT 23093 1
INSERT 23094 1
INSERT 23095 1
[adminuser@MESA-PC2 dbInstall]$
```

Figure 38: dbrestore script result

3.6 DIAGNOSTIC TOOLS AND USER SUPPORT

3.6.1 System Report

The eStation 2.0 implements a mechanism for creating an archive containing some core information on the status of the application, and namely:

- All the eStation 2 log file
- The list of products acquired by Get EumetCast and Get Internet
- The local settings
- The status of services on the system
- The postgresql database configuration
- The apache configuration
- The rsync configuration

This report can be created as indicated in the User Guide (RD-1) in paragraph 3.9.1. In particular, when you click on 'Create System Report' an interface is presented as in Figure 39.

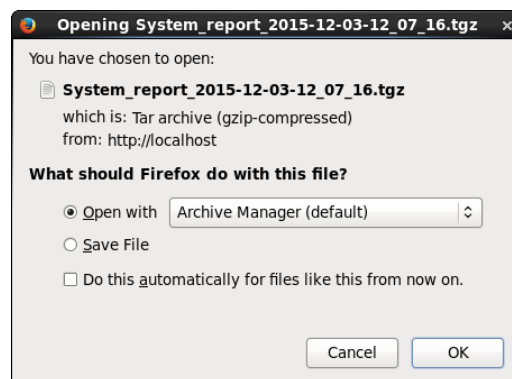


Figure 39: Creation of System Report

You can either decide to open the archive that is going to be created (only for tests) or to save it, which is the normal procedure. Click therefore on the 'Save File' option, as displayed in the next figure.

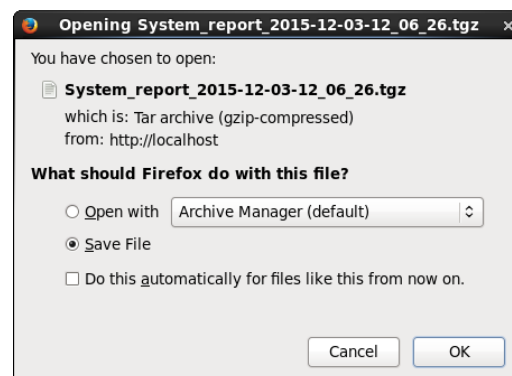


Figure 40: Save System Report

As a consequence, the archive file is saved on the local machine under the Download directory in the user home (see Figure 41).

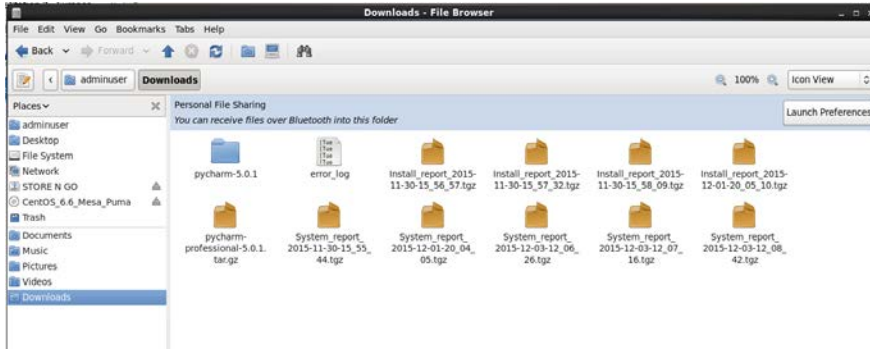


Figure 41: Location of the System Report

The file can then be send to the User Support.

3.6.2 Install Report

The eStation 2.0 implements a mechanism for creating an archive containing core information on all CentOS packages installed on the PC. Namely:

- The list of the CentOS packages, but the eStation ones.
- The list of eStation installed packages
- The list of the installed python module.

This report can be created as indicated in the User Guide (RD-1) in paragraph 3.9.1. In particular, when you click on ‘Create Install Report’ a form is presented as in Figure 42.

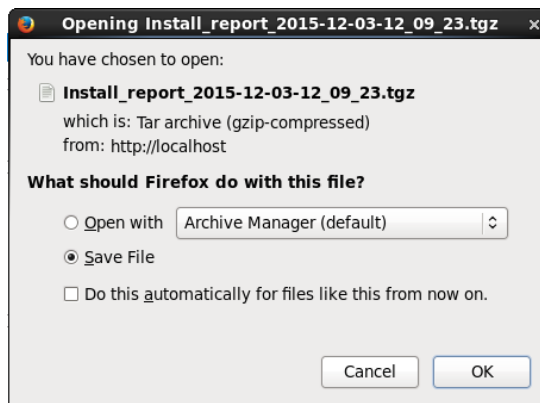


Figure 42: Create Install Report

The same options as in the creation of the System Report applies (see above paragraph). Please select the ‘Save File’ option and the report will be generated in the ‘Download’ directory.

3.6.3 Tuleap for User Support

The user Support mechanism in place under the SuCr#1 is described in the MESA Station Administration and Maintenance Manual (RD-3) in section 2.

4. SYSTEM DESCRIPTION

This section describes the eStation 2.0 application filesystem and postgresql database, and the naming conventions adopted for both the directories hosting the GTIFF eStation files, and the files themselves. Furthermore, the mechanisms implemented for synchronizing both data and database between PC2 and PC3 are presented.

4.1 DIRECTORY STRUCTURE

The eStation 2.0 is integrated into CentOS 6.6 and make use of the Linux predefined directories and their management: e.g. the /tmp directory is used for writing temporary data that are automatically cleaned at each reboot of the machine.

In addition to the standard Linux directories, some dedicated ones are created specifically for the eStation 2.0 application. These directories and their usage are described in the following paragraphs.

Base directory of the eStation 2.0 software

The eStation software (mainly python and JavaScript files) is installed in a directory tree that is created and populated at the installation of the PC. This base directory is defined by default as:

```
ESTATION2_BASE_DIR_SYSTEM = /var/www/eStation2/
```

This directory is actually a symbolic link to a version-dependending directory, which is created at the initial installation, and at the following upgrades, with the convention below:

```
/var/www/eStation2-V.V.V-Rel/
```

The initial system will be therefore under:

```
/var/www/eStation2-2.0.1-1/
```

Release independent data and settings

Some data and User's settings will be stored under a directory which is independent from the software release, so that they are not impacted by the software upgrade. This directory is identified by the variable <ESTATION2_LOCAL_DIR> and defined as below:

```
ESTATION2_LOCAL_DIR=/eStation2/
```

Under this base directory, the following subdirectories exist:

- /eStation2/settings: contains user settings and configuration of the system.
- /eStation2/get lists: list of files acquired by the 'get' services (Get EUMETCast and Get internet)
- /eStation2/log: all log files of the eStation 2.0 application
- /eStation2/requests: 'request' file created for completing the local datasets
- /eStation2/docs: documentation of the eStation 2.0 (from eStation-Docs package)
- /eStation2/layers: static layers for the application (from eStation-Layers package)

Temporary/working directories

Another category of data is represented by the temporary files that are generated by the runtime application in the background, and that need to be cleaned at each reboot. An example is represented by the working directories used for intermediate processing steps. The base directory for these files is:

```
ESTATION2_TEMP_DIR=/tmp/eStation2/
```

And subdirectories are:

```
/tmp/eStation2/services:      pid files of the running services
/tmp/eStation2/processing:    lock files of the individual processing chains
/tmp/eStation2/ingested_files: working directory for the ingestion service, where we keep track of
                             the files successfully processed, and to be deleted at the end of the ingestion loop.
```

Data Directories

The EO datasets retrieved and processed on the eStation2 are stored in predefined directories, under the base path:

```
ESTATION2_BASE_DIR_DATA=/data/
```

which subdirectories are:

```
/data/ingest/:              native data ready for ingestion
/data/ingest.wrong/:        native that raised an error in the ingestion process4
/data/processing/:         ingested and derived products
/data/Spirits/:            this is an 'optional' directory where some selected datasets are stored in
                           'SPIRITS' format.
```

Please note that, unlike the three previous categories that should not be changed by the User in normal circumstances, these 'Data' directories can be freely changed to process/visualize data that are stored on a different file system, e.g. an external disk connected to the machine.

⁴ These files are moved in this directory, for that the ingestion loop won't try to reprocess them at each loop iteration.

4.2 CONFIGURATION FILES OF EStATION 2.0

The configuration of eStation 2.0 application is done through two set of variables, the ‘System settings’ and ‘Factory and User Settings’.

System Settings

These are the main settings of the eStation, which describes the type of installation, the role of the computer, the activated version and so on.

These settings are stored in the following file:

/eStation2/settings/system_settings.ini

Please note that this file must not be manually modified: rather, it is updated using the eStation 2.0 GUI. The System Settings are listed in:

Variable	Role	Possible values
type_installation	Define the type of Installation	‘Full’
role	Define the role of the PC	PC2 or PC3
mode	Define the current mode of the PC	Nominal, Recovery or Maintenance
active_version	Define the currently activated eStation 2.0 version	2.0.1 (or next releases, e.g. 2.1.0)
data_sync	Define if the data synchronization toward the other PC is activated	true or false
db_sync	Define if the DB synchronization toward the other PC is activated	true or false
thema	Define the Thema the eStation is associated to (i.e. name of the RIC)	ACMAD AGRHYMET BDMS CICOS ICPAC MOI UoG
ingest_archive_eum	Define if the historical archives disseminated through EUMETCast have to be processed (normally always TRUE)	true or false
log_general_level	Severity Level of logging mechanism	DEBUG INFO WARNING ERROR FATAL
get_eumetcast_output_dir	Target Directory of Get Eumetcast	/data/ingest/
get_internet_output_dir	Target Directory of Get Internet	/data/ingest/

Table 4: System Settings

Factory and User Settings

These settings define the working environment of the eStation (directories and file naming), some working parameters (e.g. the sleep time of the Services) and the parameters for accessing the database. All these parameters are defined in the file:

`/var/www/eStation2/config/factory_settings.ini`

The full list of factory settings is reported as Annex at the end of paragraph. Only a part of these variables (see Table 5) are meant to be modified by the Users, and are therefore duplicated in the file:

`/eStation2/settings/user_settings.ini`

As for the system settings, the User should not modify directly the configuration file, rather operate on the System user interface.

Variable	Role	Possible values
base_dir	Define the base directory of installation	/var/www/
base_tmp_dir	Define the base directory for temp directories	/tmp/eStation2
data_dir	Define the base directory for data	/data/
ingest_dir	Define the directory hosting data to be ingested	/data/ingest/
processing_dir	Define the directory hosting data ingested or derived	/data/processing/
eumetcast_files_dir	Define input directory of get EumetCast	/eumetcast/
static_data_dir	Define the directory hosting the static layers.	/eStation2/layers/
archive_dir	Define the directory hosting the historical datasets (to be modified by the User before import)	/media/MESA-2/Archives/
host	eStation DB host	localhost
port	eStation DB port	5432
dbuser	eStation DB owner	estation
dbpass	Password of eStation DB owner	mesadmin
dbname	Name of eStation DB	estationdb
default_language	Default language	EN, FR
log_general_level	Severity Level of logging mechanism	DEBUG INFO WARNING ERROR FATAL

get_eumetcast_output_dir	Target Directory of Get Eumetcast	/data/ingest/
get_internet_output_dir	Target Directory of Get Internet	/data/ingest/

Table 5: User settings

The system reads first the factory settings, and then overwrites the parameters that are also defined, as no empty fields, in the user settings.

```

[FACTORY_SETTINGS]
www_root_dir = /var/www
proj4_lib_dir = /usr/share/proj
postgresql_executable = /etc/init.d/postgresql-9.3
webserver_root =

base_dir = %(www_root_dir)s/eStation2
base_tmp_dir = /tmp/eStation2
base_local_dir = /eStation2
data_dir = /data
ingest_dir = /data/ingest/
ingest_error_dir = /data/ingest.wrong/
get_internet_output_dir=%(ingest_dir)s
get_eumetcast_output_dir=%(ingest_dir)s
static_data_dir = %(data_dir)s/static_data/
archive_dir = %(data_dir)s/archives/
eumetcast_files_dir = /eumetcast/
host = localhost
port = 5432
dbuser = estation
dbpass = mesadmin
dbname = estationdb
apps_dir = %(base_dir)s/apps
config_dir = %(base_dir)s/config/
settings_dir = %(base_local_dir)s/settings/
processing_dir = %(data_dir)s/processing/
log_dir = %(base_local_dir)s/log/
log_general_level=info
template_mapfile = %(apps_dir)s/analysis/MAP_main.map
es2_sw_version = 2.0.0
processed_list_base_dir = %(base_local_dir)s/get_lists
processed_list_eum_dir = %(processed_list_base_dir)s/get_eumetcast
processed_list_int_dir = %(processed_list_base_dir)s/get_internet
get_eumetcast_processed_list_prefix = %(processed_list_eum_dir)s/get_eum_processed_list_
get_internet_processed_list_prefix = %(processed_list_int_dir)s/get_internet_processed_list_
poll_frequency = 5
pid_file_dir = %(base_tmp_dir)s/services
get_internet_pid_filename = %(pid_file_dir)s/get-internet.pid
get_eumetcast_pid_filename = %(pid_file_dir)s/get-eumetcast.pid
ingestion_pid_filename = %(pid_file_dir)s/ingest.pid
processing_pid_filename = %(pid_file_dir)s/processing.pid
system_pid_filename = %(pid_file_dir)s/system.pid
processing_tasks_dir = %(base_tmp_dir)s/processing/
gdal_dir = /usr/bin
gdal_merge = %(gdal_dir)s/gdal_merge.py
gdal_polygonize = %(gdal_dir)s/gdal_polygonize.py
gdal_translate = %(gdal_dir)s/gdal_translate
acq_service_dir=%(base_dir)s/apps/acquisition
proc_service_dir=%(base_dir)s/apps/processing
status_system_dir=%(base_dir)s/apps/es2system
status_system_pickle=%(status_system_dir)s/system_status.pkl

```

```
schema_products = products
schema_analysis = analysis
schema_data = data
default_language = eng
system_delay_data_sync_min=10
system_sleep_time_sec=10
system_time_db_dump_hhmm=00:00
prefix_eumetcast_files=MESA_JRC_
db_dump_dir=%(base_local_dir)s/db_dump
get_eumetcast_sleep_time_sec=60
get_internet_sleep_time_sec=60
processing_sleep_time_sec=10
estation2_layers_dir = %(www_root_dir)s/eStation2_Layers
requests_dir = %(base_local_dir)s/requests/
docs_dir = %(base_local_dir)s/docs/
db_dump_exec = /usr/pgsql-9.3/bin/pg_dump
spirits_output_dir = /data/Spirits/
ftp_eumetcast_url= ftp://h05-ftp.jrc.it/eumetcast/
ftp_eumetcast_userpwd= narmauser:narma11
system_time_spirits_conv= 00:10
```

4.3 FILE AND DIRECTORY NAMING CONVENTIONS

Both eStation 2.0 dataset files (i.e. the ingested and derived products) and the directory trees where they are stored obey to strict naming rules, so that the data location is depending on very few key elements. These elements are namely the product, version, subproduct and mapset and their meaning is described in Section 2 of the User Manual, and briefly repeated hereafter:

Product: the product identification code (indicated as <product>)

Version: the version of the dataset, corresponding to the algorithm (indicated as <version>)

Subproduct: the specific dataset that belong to a product (indicated as <subproduct>)

Mapset: the geo-reference and boundary box of the file (indicated as <mapset>)

Directory naming

All files are stored under a base directory, which is by default **/data/processing/**; under this root, the files are organized per product and version, as below:

<product>/<version>/ (e.g. tamsat-rfe/2.0/)

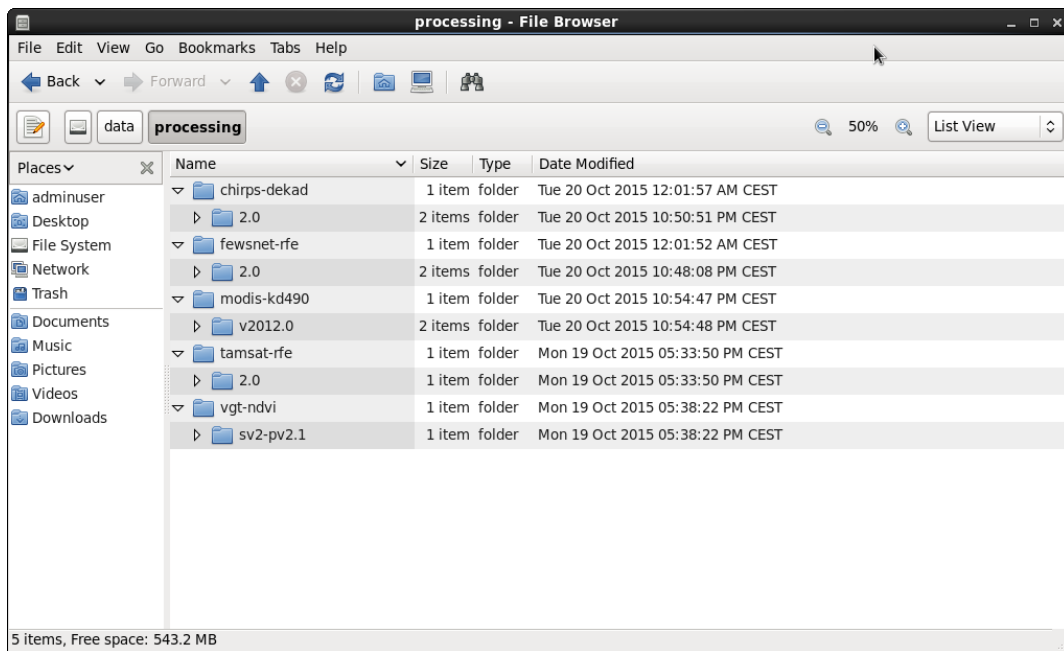


Figure 43: data directory structure - 1

At this level, a folder called 'archive' exists for hosting the files in original format; besides that, a folder for the mapsets (one or more) of the ingested and processed files exists (see Figure 44).

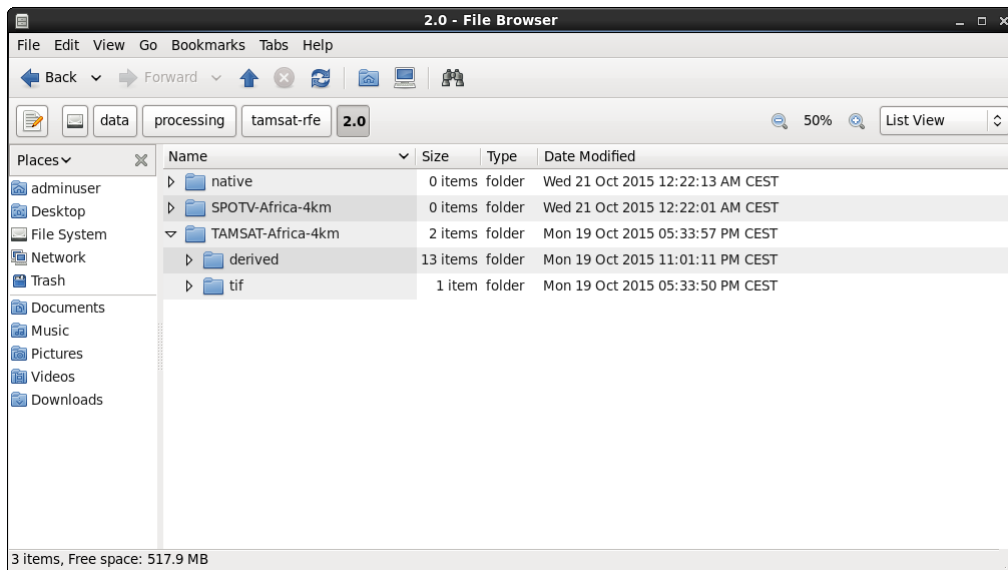


Figure 44: data directory structure – 2

Under each 'mapset' folder, the ingested files are located in the '/tif' subdirectory, while the processed ones in the 'derived' subdirectory. The 'tif/' and 'derived/' folders contains eventually a subdirectory named after the subproduct code (see Figure 45).

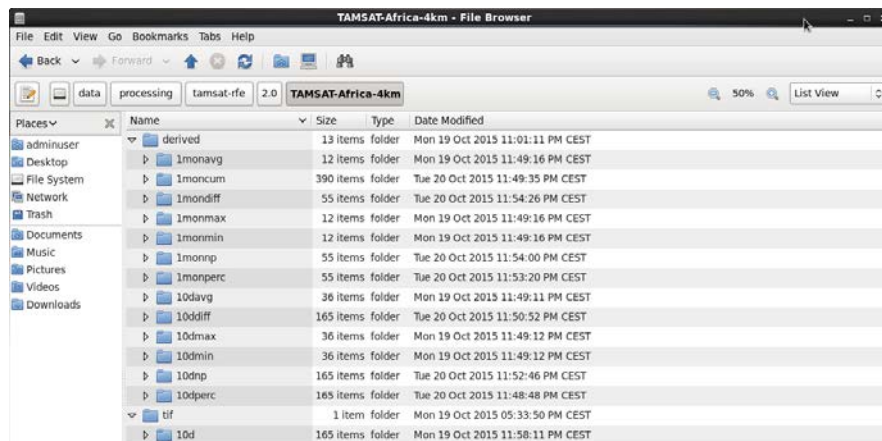


Figure 45: data directory structure - 3

Some examples of directories for ingested and derived files are listed here below:

e.g.: `/data/processing/vgt-ndvi/spot-v1/WGS84_Africa_1km/tif/ndv`

For derived products:

`mapset/derived/subproductcode/`

e.g.: `/data/processing/vgt-ndvi/spot-v1/WGS84_Africa_1km/derived/10dmax`

File Naming

The filename of the geotiff files contains all fields described in the previous section, namely: <product>, <version>, <mapset> and <subproduct>; in addition, the date of the product is located at the beginning of the filename, for ease the file sorting within the directory (see Figure 46). The full filename results therefore from the composition of these elements, in the following order:

<date>_<productcode>_<subproductcode>_<mapset>_<version>.tif
 e.g.: 20150101_vgt-ndvi_ndv_WGS84-Africa-1km_spot-v1.tif

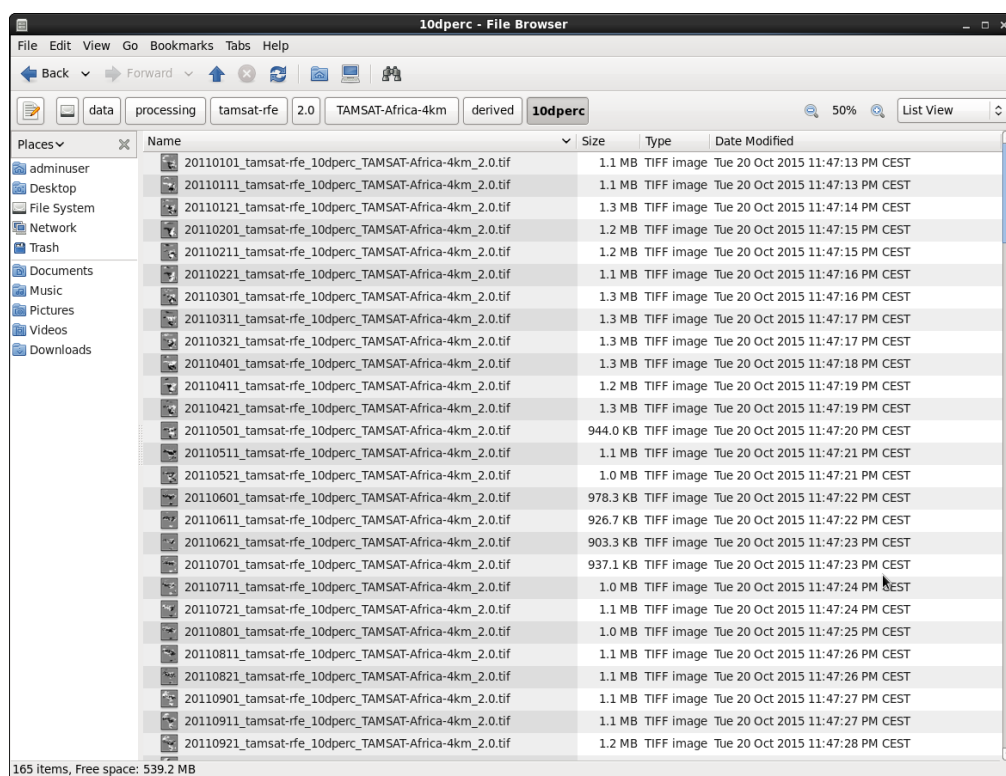


Figure 46: Example of eStation 2.0 file naming

Note that the ‘separator’ between the various field is the ‘underscore’ char (‘_’), which therefore cannot be used in any of the product, subproduct, mapset and version definition.

File metadata

All standard eStation 2.0 files contain a well-defined set of GTIFF tags that contain the metadata associated to each image. These metadata describes all information the User need to interpret the image and to identify how is has been generated.

Table 6: List of eStation 2.0 images metadata.

Tag Name	Definition	Comment
eStation2_category	Product Category	It is used in the GUI to group the products together (e.g. ‘Vegetation’, ‘Rainfall’, ‘Fire’)

<i>eStation2_conversion</i>	<i>Equation to convert from DN to physical value</i>	<i>It is a fixed field: Phys = DN * scaling_factor + scaling_offset</i>
<i>eStation2_date</i>	<i>Date of the subproduct</i>	<i>It refers to the acquisition date of the image (e.g. 20150101)</i>
<i>eStation2_date_format</i>	<i>Format of the date</i>	<i>It can be YYYYMMDD (date) or YYYYMMDDHHMM (datetime)</i>
<i>eStation2_defined_by</i>	<i>Who has defined the subproduct</i>	<i>It can be 'JRC' or the User</i>
<i>eStation2_descr_name</i>	<i>A descriptive name of the subproduct</i>	<i>It is slightly more descriptive than the product code, and is used in the GUI</i>
<i>eStation2_description</i>	<i>Longer description</i>	<i>Long description used in the GUI</i>
<i>eStation2_es2_version</i>	<i>Version of the eStation 2.0</i>	<i>It starts with 2.0.0 and will evolve with the updates (2.1.0 and so on)</i>
<i>eStation2_frequency</i>	<i>Frequency of the subproduct.</i>	<i>How often the subproduct is generated (e.g. 'every dekad')</i>
<i>eStation2_input_files</i>	<i>The list of files used for deriving the subproduct.</i>	<i>This list contains the files in 'native' format for the ingested subproduct, or a list of eStation 2.0 files for the derived ones.</i>
<i>eStation2_mac_address</i>	<i>The unique HW identifier of the computer where the subproduct has been generated.</i>	<i>It allows to recognize if the file has been computed on the local machine, or imported from another one. It is relevant only for traceability/diagnostic.</i>
<i>eStation2_mapset</i>	<i>The subproduct mapset.</i>	<i>Mapset, as defined in the User Manual, in Chapter 2.4.5.</i>
<i>eStation2_nodata</i>	<i>The numeric value used to code 'nodata'</i>	<i>See also User Manual 2.4.8</i>
<i>eStation2_product</i>	<i>Name of the 'product'</i>	<i>See User Manual 2.4.1 for definition of 'products'</i>
<i>eStation2_product_version</i>	<i>Version of the 'product'</i>	<i>See User Manual 2.4.2 for definition of 'version'</i>
<i>eStation2_provider</i>	<i>Data provider</i>	<i>Who has initially generated the EO product (e.g. Copernicus)</i>
<i>eStation2_scaling_factor</i>	<i>Scaling factor to be applied for conversion to physical value.</i>	<i>See also User Manual 2.4.8 and the conversion equation above.</i>
<i>eStation2_scaling_offset</i>	<i>Scaling offset to be applied for conversion to physical value (it is normally 0.0).</i>	<i>See also User Manual 2.4.8 and the conversion equation above.</i>
<i>eStation2_subdir</i>	<i>Subdirectory of the image</i>	<i>It is where the image is stored under the /data/processing root dir.</i>
<i>eStation2_subProduct</i>	<i>Name of the 'subproduct'</i>	<i>See User Manual 2.4.3 for definition of subproduct</i>

As an example, the set of metadata for the 'vgt-ndvi' product, ingested ndv subproduct of date 201501 eStation2_category=vegetation

eStation2_comp_time=2015-03-03 15:12:42
eStation2_conversion=Phys = DN * scaling_factor + scaling_offset
eStation2_date=20150101
eStation2_date_format=YYYYMMDD
eStation2_defined_by=JRC
eStation2_descr_name=Normalized Difference Vegetation Index

eStation2_description=The NDVI can be used to measure and monitor plant growth, vegetation cover, and biomass production. It is a dimensionless variable. The values vary between -1 and 1. Increasing positive NDVI values indicate increasing amounts of green vegetation. Moderate values represent shrub and grassland (0.2 to 0.3), while high values indicate temperate and tropical rainforests (0.6 to 0.8). NDVI values near zero and decreasing negative values indicate non-vegetated features such as barren surfaces (rock and soil) and water, snow, ice, and clouds.

eStation2_es2_version=2.0.0

eStation2_frequency=e1dekad

eStation2_input_files=/data/ingest/g2_BIOPAR_NDVI_201501010000_AFRI_PROBAV_V2_1.ZIP;

eStation2_mac_address=6c:ae:8b:52:77:d2

eStation2_mapset=SPOTV-Africa-1km

eStation2_nodata=-32768

eStation2_product=vgt-ndvi

eStation2_product_version=proba-v2.1

eStation2_provider=VITO/Copernicus

eStation2_scaling_factor=0.001

eStation2_scaling_offset=0.0

eStation2_subdir=vgt-ndvi/proba-v2.1/SPOTV-Africa-1km/tif/ndv/

eStation2_subProduct=ndv01 is represented below:

4.3.1 PCs synchronization

Data synchronization

The data synchronization between PC2 and PC3 is part of a more general mechanism of data sync implemented by the Supply #1 Contractor (TPZF/TAS) as described in the MESA Station Administration Manual (RD-3) in paragraph 6.5.

In particular, the directory hosting the EO datasets ingested and derived on the eStation (normally /data/processing/) is synchronized from PC2 to PC3 every 30 minutes.

DB synchronization

The eStation database contains two schema dedicated to the User operation: the 'products' and the 'analysis' schema. The 'products' schema (see paragraph 3.3.3) is composed by 17 tables hosting all information related to the products definition, the datasource definitions, the data processing services (get, ingest and processing). An additional table, called 'spirits', is dedicated to the conversion of the datasets to 'SPIRITS' format (optional).

The 'analysis' schema (see paragraph 3.3.3) is composed by 7 tables that defines the legends, their association with the products, the languages (including the translation table 'i18n') and the timeseries draw properties.

What it is

The rationale for having two separate schemas is that, in Nominal mode, the User modifies the 'products' schema only on PC2, and 'analysis' schema only on PC3. Therefore, in the standard condition, there is only need to 'push' the changes of 'products' from PC2 -> PC3 and the changes of 'analysis' from PC3 -> PC2.

The database synchronization is performed by the 'Bucardo' tool, which is an asynchronous master-master replication system for PostgreSQL (see <https://bucardo.org/wiki/Bucardo>). Bucardo is installed together with the eStation 2.0 application at the MESA install, and configured in the 'Post Installation' phase (add reference).

How it is configured

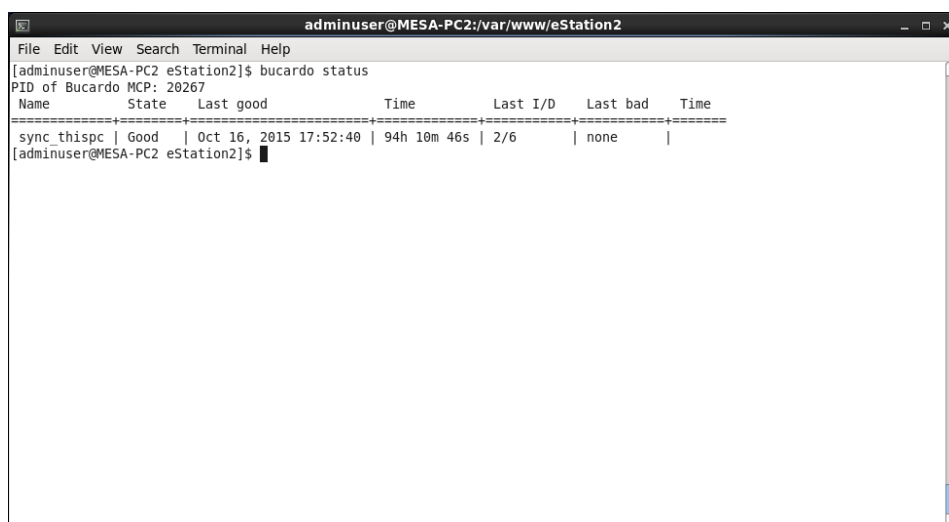
Bucardo is a service running in the background, and is completely independent from the eStation2 services. Its status can be checked by entering in a terminal the command:

bucardo status

The expected result is shown in Figure 47: note that the indication of the PID of bucardo MCP is the actual proof that bucardo is running, and not only configured. The full list of the bucardo objects created is displayed by entering the command:

bucardo list all

which reports the current setup of the machine (see Figure 48).



```
adminuser@MESA-PC2:/var/www/eStation2
File Edit View Search Terminal Help
[adminuser@MESA-PC2 eStation2]$ bucardo status
PID of Bucardo MCP: 20267
-----
Name          State    Last good          Time          Last I/D    Last bad    Time
-----
sync thispc  | Good   | Oct 16, 2015 17:52:40 | 94h 10m 46s | 2/6        | none      |
[adminuser@MESA-PC2 eStation2]$
```

Figure 47: bucardo from the command line

The key feature of bucardo we exploit on the eStation is the 'sync' object. Namely, a 'sync' exists, with the same name ('sync_thispc') but different definition on PC2 and PC3.

```

adminuser@MESA-PC2:/var/www/eStation2
File Edit View Search Terminal Help

[adminuser@MESA-PC2 eStation2]# bucardo list all
-- dbgroups
#dbgroup: group_thispc Members: mesa_otherpc:target mesa_thispc:source
-- databases
Database: mesa_otherpc Status: stalled Conn: psql ip: 5432 @ bucardo: d: eStationdb: h: MESA-PC2
Database: mesa_thispc Status: active Conn: psql ip: 5432 @ bucardo: d: eStationdb: h: MESA-PC2
-- relgroups
Relgroup: sync_thispc DB: mesa_thispc Members: products:data type, products:datasource:description, products:date:format, products:datetime:source, products:frequency, products:ingestion, products:internet:source,
products:mapset, products:process:product, products:processing, products:product, products:product_acquisition:data:source, products:product_category, products:sub_datasource:description, products:theme, products:theme:product
-- syncs
Sync: sync_thispc
-- syncs
Sync: sync_thispc Relgroup: sync_thispc [stalled]
DB group: group_thispc mesa_otherpc:target mesa_thispc:source
-- tables
1. Table: analysis:libn DB: mesa_thispc PK: label (varchar)
2. Table: analysis:languages DB: mesa_thispc PK: language (varchar)
3. Table: analysis:layers DB: mesa_thispc PK: layerid (bigint)
7. Table: analysis:legend DB: mesa_thispc PK: legend_id (integer)
8. Table: analysis:legend_step DB: mesa_thispc PK: legend_id[from:step]to:step (integer[float])float8
9. Table: analysis:product:legend DB: mesa_thispc PK: productcode[subproductcode]version:legend_id (varchar[varchar][varchar]bigint)
10. Table: analysis:timeSeries:drainproperties DB: mesa_thispc PK: productcode[subproductcode]version (varchar[varchar][varchar])
11. Table: products:date:format DB: mesa_thispc PK: date:format (varchar)
12. Table: products:datetime:source DB: mesa_thispc PK: datetime:source_id (varchar)
13. Table: products:frequency DB: mesa_thispc PK: frequency_id (varchar)
14. Table: products:internet:source DB: mesa_thispc PK: productcode[subproductcode]version:internet_id (varchar[varchar][varchar]varchar)
15. Table: products:mapset DB: mesa_thispc PK: mapsetcode (varchar)
16. Table: products:process:product DB: mesa_thispc PK: process_id[productcode][subproductcode]version:mapsetcode (integer[varchar][varchar][varchar]varchar)
17. Table: products:processing DB: mesa_thispc PK: process_id (integer)
18. Table: products:product DB: mesa_thispc PK: productcode[subproductcode]version (varchar[varchar][varchar])
19. Table: products:product_acquisition:data:source DB: mesa_thispc PK: productcode[subproductcode]version:data:source_id (varchar[varchar][varchar]varchar)
20. Table: products:product_category DB: mesa_thispc PK: category_id (varchar)
21. Table: products:sub_datasource:description DB: mesa_thispc PK: productcode[subproductcode]version:datasource_descr_id (varchar[varchar][varchar]varchar)
22. Table: products:theme DB: mesa_thispc PK: theme_id (varchar)
23. Table: products:theme:product DB: mesa_thispc PK: theme_id[productcode]version:mapsetcode (varchar[varchar][varchar]varchar)
-- sequences
There are no sequences.
[adminuser@MESA-PC2 eStation2]#

```

Figure 48: all bucardo objects

On PC2 the 'sync' copies 'products' table from the local database to the one of the other PC, while on PC3 the 'analysis' table is synchronized.

In order to monitor the execution of the sync, the user can type in a terminal:

bucardo status sync_thispc

which returns the latest successful execution of the sync, the current start (Good), the number of tables involved in the sync and a number of ancillary information on the 'sync' definition, including the 'Autokick'⁵ attribute, which is set by default.

```

adminuser@MESA-PC2:/var/www/eStation2
File Edit View Search Terminal Help

[adminuser@MESA-PC2 eStation2]$ bucardo status sync_thispc
=====
Last good           : Oct 16, 2015 17:52:40 (time to run: 1s)
Rows deleted/inserted : 2 / 6
Sync name           : sync_thispc
Current state        : Good
Source relgroup/database : sync_thispc / mesa_thispc
Tables in sync      : 16
Status               : Stalled
Check time           : None
Overdue time         : 00:00:00
Expired time         : 00:00:00
Stayalive/Kidsalive  : Yes / Yes
Rebuild index        : No
Autokick             : Yes
Onetimecopy          : No
Post-copy analyze     : Yes
Last error:          :
=====
[adminuser@MESA-PC2 eStation2]$

```

Figure 49: status of bucardo sync

⁵ This attribute causes the sync to be executed automatically once a field of one of the involved tables is modified in the 'source' database.

How it works

Bucardo is an asynchronous replication system for postgresql, allowing multi-master and multi-slave operations; it is free and open source software released under the BSD licence⁶.

On the eStation it is installed directly from the DVD on PC2 and PC3, and configured at the first run of the machines, when both PC2 and PC3 are up and running and they 'see' each other.

A dedicated 'bucardo' database is created on both machines, where the configuration is stored; in addition, a 'bucardo' schema is added to the 'estationdb' database, where all changes to each table are recorded, in order to trigger the synchronization actions from one OC to the other.

⁶ See <https://bucardo.org/wiki/Bucardo>

4.3.2 Description of the eStation .rpm

The eStation application is composed by 3 .rpm packages that are installed during the initial installation and upgraded through EUMETCast dissemination. Beside the main 'eStation-Apps' package, two minor packages are distributed, containing the static layers for visualization and the multilanguage documentation.

Package name	Purpose	Pre/post installation actions	Target Directories
eStation-Apps	eStation 2.0 code (python and.js)	<ul style="list-style-type: none">• Configure the machine (users, db, directories)• Symbolic links	/var/www/eStation2-2.X.X
eStation-Layers	eStation 2.0 static layers (e.g. Administrative regions .shp)	None	/eStation2/layers
eStation-Docs	Documentation (.pdf)	Symbolic links	/eStation2/docs

4.3.3 Database Description

As anticipated in paragraph 2.7.3, the estation database is composed by two main schemas: 'products' and 'analysis'.

Products schema

The 'products' schema contains 17 tables whose definition and relationships are represented in Figure 50⁷.

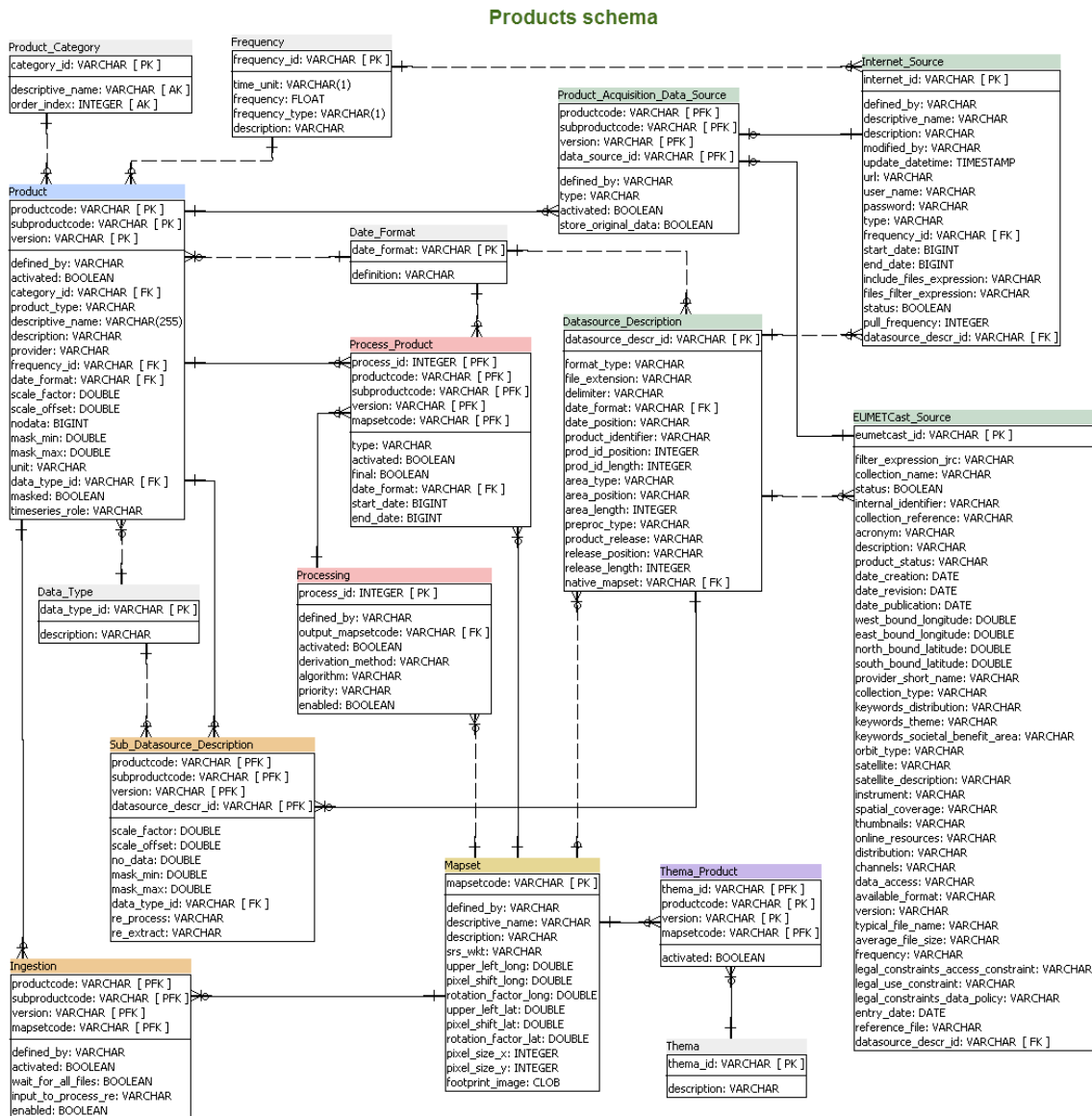


Figure 50: 'products' schema

The tables are arbitrarily grouped in categories in the following description, only for facilitating the explanation.

⁷ With the exception of the 'spirits' table, which is represented separately in Figure 51.

Definition's Tables

These tables describe objects that are used by the eStation 2.0 data processing server, namely:

- data_type : the type of data contained in raster files (Byte, Int16, and so on)
- date_format: the various formats of the date and date-time fields of the EO products (YYYYMMDD for daily products identified by a field like 20150101)
- frequency: describes the frequency or periodicity of the EO products (e.g. 'every 1 month' or '2 per day').
- product_category: the thematic category of the EO products: 'rainfall', 'fire', vegetation indicators and so on.

Products, sources and mapset tables

The following tables define the 'core' objects of the eStation, i.e. the EO products, the sources they are retrieved from, and the 'mapset' object.

- product: defines all the EO products and subproducts processed by the eStation. For each product – that is always defined together with its version – a 'native' subproduct exists, one or more 'ingested' product and, optionally, 'derived' products.

- datasource_description: describes a source of EO dataset, either from the EUMETCast dissemination or an ftp/http server (every record is associated to a record in the 'eumetcast_source' or 'internet_source'). Main role of the table is defined the filenaming of the 'native' (i.e. incoming) files.
- eumetcast_source: defines the datasets disseminated by EUMETCast.
- internet_source: defines EO products made available on the internet, on ftp or http sites. It describes the type, url, credentials and directory structure of the site.
- sub_data_source_description: defines the contents of the 'native' products, i.e. of the incoming files in original format (HDF, netcdf, MSG). In particular, describes the layers to be extracted from the files in terms of nodata, values scaling, range, type of data, and so on.
- mapset: defines a geo-referenced region and its projection
- thema: the list of the MESA Thema, which correspond to the list of the Regional or Continental Implementation Centres.
- thema_product: establish the associations between the products and Themas, i.e. which products are relevant for one Thema.

Service definition Tables

The following tables are used to define and control the get, ingest and processing services of the eStation.

- product_acquisition_data_source: establishes associations between the sources and the products, i.e. which are the sources (at least one) a products is retrieved from. It controls the get Services.

ingestion: defines which 'ingested' subproducts have to be extracted for each mapset. It controls the ingestion Service.

processing: defines a number of processing chains, i.e. which algorithms have to be applied for which chain. It is complemented by the processing_products table. It controls the processing Service.

processing_products: defined which subproducts are used, as input and output, for each processing chain declared in 'processing' table.

Additional Tables

spirits: this table is presented separately, as it is an optional one, which configures the conversion of datasets from the eStation 2.0 format to the SPIRITS one.

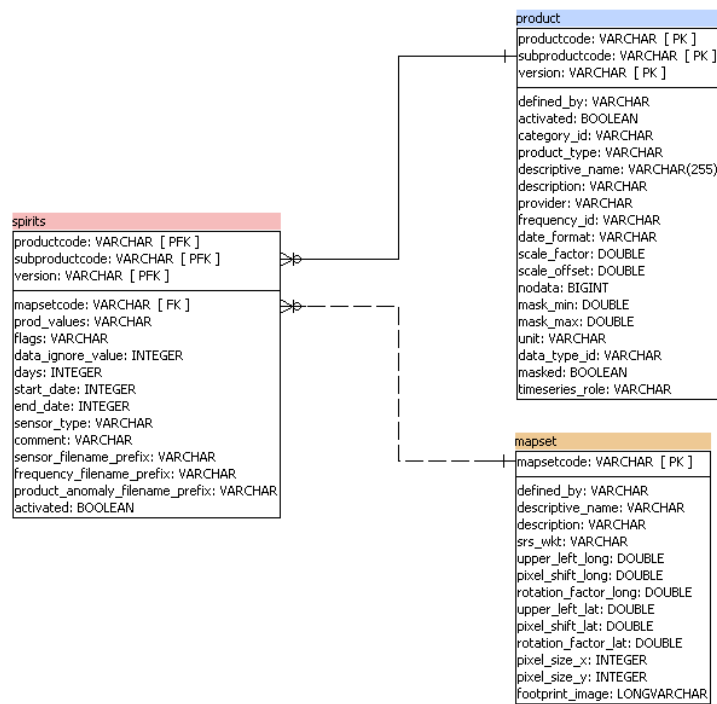


Figure 51: spirits table

Analysis schema

The 'analysis' schema contains 7 tables, and is associated to the product table of 'products' schema. Its role is mainly to host all parameters used in the visualization (Analysis tab).

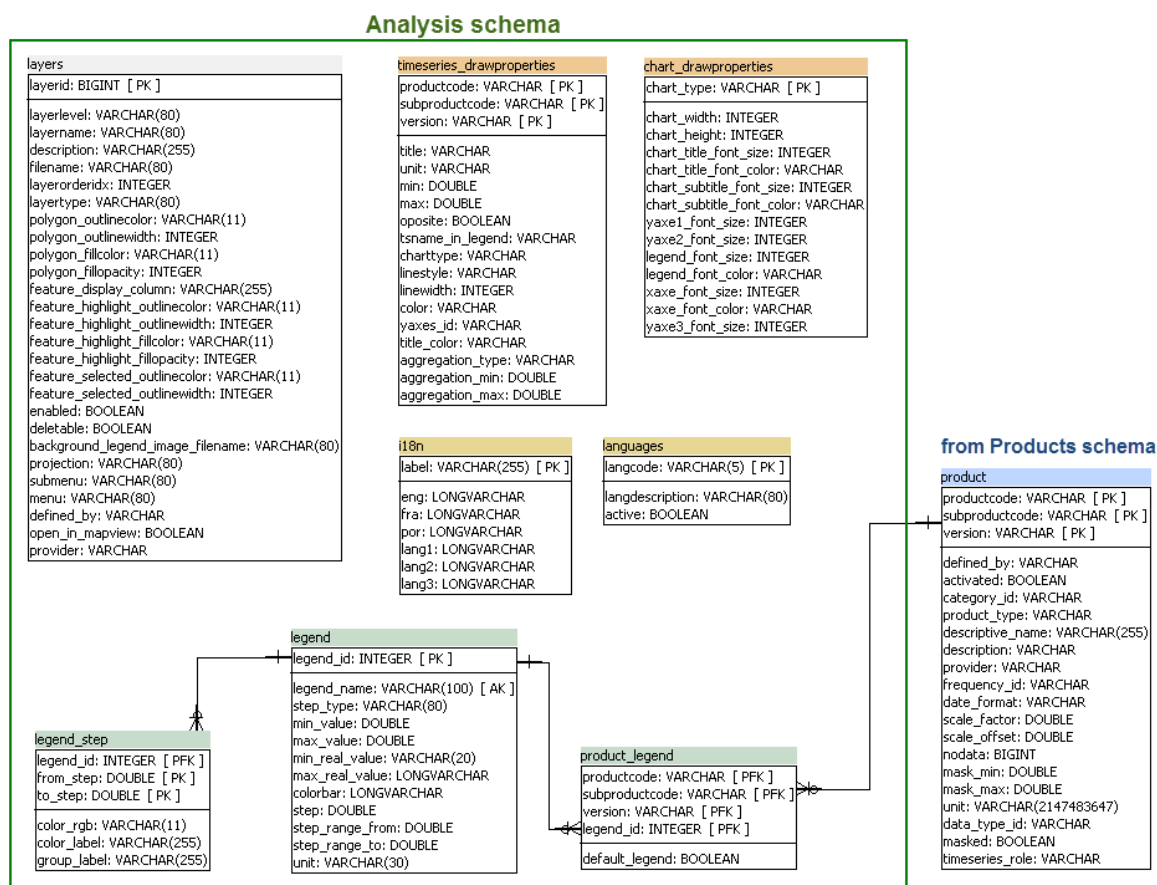


Figure 52: 'analysis' schema

i18n:	Contains all language definitions, i.e. how to translate all labels and texts of the GUI in various languages.
languages:	Lists the languages supported by the system and defined in i18n
legend:	Defines 'legends' or 'color palettes', i.e. which values are assigned to given ranges of the products. It is used for rendering the EO products in Analysis.
legend_step:	Define all ranges of a legend, i.e. associate each interval/range to a color.
product_legend:	Associates the products to the legends.
layers:	vector layer properties of all the supplied vector layers and user defined vector layers
timeseries_drawproperties:	The draw properties of the available time series datasets and the yAxe definitions
chart_drawproperties:	Contains the properties of the different chart types, default, matrix, etc.

5. TROUBLESHOOTING

This section describes the most common issues/errors in the system, and how to deal with them.

5.1 THE eSTATION 2.0 WEB INTERFACE IS NOT DISPLAYED

If the eStation GUI is not displayed in the Firefox browser (e.g. an 'Internal Error' message is represented), the administrator has to:

- Check if apache is running: control the status of services as described in the RD-3 document in paragraph 6.1.
If the apache service is not running, start as indicated in RD-3 document in paragraph 6.2.

- Check the apache logfile: open a terminal and type the command:

```
tail /usr/local/src/tas/eStation_wsgi_srv/error.log
```