Department of Sustainability
Division Models and technologies for Risks Reduction

Project “Strengthening Early Warning and Climate Services in Eswatini”. Support to develop "Numerical Weather Prediction System"

WP 1. Numerical weather prediction system

Implementation of operational NWP system at Eswatini Meteorological Service.

Deliverable: WP1.D2

Date: 08 June 2023

AUTHORS

Massimo D’Isidoro (*) (massimo.disidoro@enea.it)

Guido Guarnieri (guido,guarnieri@enea.it)

Gaia Righini (gaia.righini@enea.it)

Lina Vitali (lina.vitali@enea.it)

(*) Contact author

Department of Sustainability
Division Models and technologies for Risks Reduction

Contents

1 INTRODUCTION

A Memorandum of Understanding (MoU) on cooperation in the field of climate change vulnerability, risk assessment, adaptation and mitigation, was signed in Bonn on 17 May 2017 between the Italian Ministry for the Environment, Land and Sea and the Swazi Ministry of Tourism and Environment Affairs. Based on the above cited MoU, an Italy-Eswatini Joint Committee (JC) was established and approved the implementation of MoU on 4th July 2017.

Following one of the proposed activities in the MoU, concerning the enforcement of the Eswatini Early Warning System, the Project "Strengthening Early Warning and Climate Services in Eswatini" was funded by the Italian Ministry. In this framework, under a Letter of Agreement signed between ENEA (Italian National Agency for the New Technologies, Energies and Sustainable Economic Development) and UNDP (United Nations Development Program) Regional Representative in Eswatini for a duration of one year from 21st February 2022, ENEA was enrolled to develop a Numerical Weather Prediction (NWP) system at Eswatini Meteorological Service premises.

The aim of the ENEA activity is to enhance the Eswatini Department of Meteorology forecast capabilities by establishing a NWP, using a state-of-the-art mesoscale model at high horizontal resolution, necessary to resolve scales of convective systems which are the prevalent weather disturbance in the country.

The Work Package 1 (WP1) of the Project is devoted to the development and implementation of the NWP system and is basically divided in two phases: 1) development and testing the system on ENEA machines and 2) porting the system on the High Parallel Computing facility funded by the Italian Ministry of Ecological Transition and established in Eswatini at the Meteorological Service premises.

The phase 1 tasks were accomplished in September 2022 and were documented in the project deliverable report WP1.D1.

The present document is the WP1.D2 final deliverable and gives a detailed description of the activities conducted between October 2022 and the end of the contract, focussed on the installation of the NWP system at Eswatini Met Service.

In Section 2 a description of the accomplished technical activities is given, while Section 3 is devoted to describing how the system works and what users need to know to make some customization.

2 SYSTEM DESCRIPTION AND INSTALLATION

During the first phase of the Project a NWP system targeted on Eswatini (Eswatini Numerical Weather Prediction, hereafter ENWP) was developed and tested on ENEA HPC facility based in Portici (Italy), named CRESCO (Computational RESsearch Centre on COmplex systems; Iannone et al., 2019). A description of those activities was given in WP1.D1.

Since late July 2022, the ENEA team started working remotely on the HPC machine located at Eswatini Met Service, to prepare and install the necessary software to run the system. The work was finalised during a mission to Eswatini conducted between 23rd and 27th January 2023.

Here the work performed remotely and in situ to establish the NWP at Eswatini Meteorological Service is described. Initially, the aim of the effort was to install the Operating System (OS) on computing nodes and storage server of the HPC facility and to proper configure all hardware and network connections. This first step was almost completely managed remotely, with the collaboration of local Met Service personnel, in particular Sive Shabalala and Bafana Simelane. The final part of the work was done in Eswatini: the installation of all the libraries and software on the HPC nodes and storage server, needed by the meteorological model WRF (see WP1.D1 for a detailed description) and the installation of the numerical model itself, including pre-processing and post-processing tools, the latter implemented ad hoc to produce meteorological maps and graphs for Met Service forecasters.

2.1 HPC system

The Hewlett Packard HPE machine is composed of the following hardware, embedded in a rack (see ANNEX I for details about the hardware supply):

- 1 Management Node, HPE DL360 Gen 10;
- 1 Storage Server, HPE Apollo 4200 Gen 10, equipped with a Mellanox Infiniband EDR card, and containing 24x10 TB Hard Disks;
- 4 Computational Nodes, HPE Apollo 2600 Gen10, every node with 48 cores and a Mellanox Infiniband EDR card;
- 1 Mellanox Infiniband EDR Switch;

- 1 Ethernet Switch Aruba 8320

2.2 OS installation

The OS chosen for the HPC machine is Rocky Linux, version 8.7, downloaded from the official site (http://rockylinux.org).

Along with the system administrator (user: *root*; password: *eswmet123!!*), a user dedicated to the operational execution of forecast was created (user: *eswmet*, password: *eswmet123!!*) [(*)]. This choice is necessary to maintain administration tasks separated from operational ones.

In addition to the OS, many other software libraries, preparatory to the numerical model set up, have been installed by *root* user on all system computational nodes. The installation wad executed by means of the *dnf* tool. Libraries are listed hereafter:

- *git*
- *g++*
- *wget*
- *libcurl-devel*
- *vim-enhanced*
- *readline-devel*
- *libx11-devel*
- *libxt-devel*
- *epel-release* (followed by the command: *dnf config-manager –set-enabled powertools*)
- *R*
- *imagemagick*
- *xorg-X11-xserver-xorg*

[(*)] For security reasons, *root* and *eswmet* passwords shown in this document will be changed in agreement with Met Service technicians.

- *xorg-X11-xauth*
- *xorg-X11-apps*

2.3 WRF installation

WRF model version 4.2.2 (Skamarock et al., 2019; *https://github.com/wrf-model/WRF*, *https://www.mmm.ucar.edu/models/wrf*) was installed on the machine. The code, together with the additional software needed to properly operate, was installed using the dedicated user *enwp* in its home space, after downloading the corresponding packages from their web sites[(**)]. The libraries installation order was the following:

- *zlib-1.2.11*
- *szip-2.1.1*
- *libpng-1.2.50*
- *hdf5-hdf5-1_10_8*
- *netcdf-4.6.1*
- *netcdf-fortran-4.4.4*
- *jaspet-1.900.1*
- *wgrib2*

Source codes are stored in the folder *lib_src* in the *enwp* home directory. In the same folder, a README_INSTALL file explains all the steps performed to compile each package.

WRF model and its pre-processing tools (WPS) are stored in the *wrf_code* folder and were compiled using *gfortran*. The *configure* script that comes with the software was used to optimise

[(**)] Note that all the software used to build the system is in the public domain and can be used without any fee or charge (e.g., GNU GENERAL PUBLIC LICENSE or similar).

the compilation. After that, some executables were created, four of which are those needed to perform the weather forecast, that is:

- *ungrib.exe*
- *metgrid.exe*
- *real.exe*
- *wrf.exe*

The executables are stored under *wrf_code/WRF-4.2.2/* (*real.exe* and *wrf.exe*) and *wrf_code/WPS-4.2/* (*ungrib.exe* and *metgrid.exe*). Details on the tasks performed by each executable are given in Section 3.

Finally, *python* language with some necessary modules were installed by means of *anaconda* suite. Python is needed to produce the graphical output of the meteorological quantities predicted by the model. After downloading and installing *anaconda* (www.anaconda.org; https://docs.anaconda.com/anaconda/install/linux/), a "*wrf-python*" environment was created and the additional packages were installed by means of the *conda* command, in the following order:

- *conda create --name wrf-python --channel conda-forge*
- *conda activate wrf-python*
- *conda install --channel conda-forge wrf-python*
- *conda install --channel conda-forge netcdf4*
- *conda install --channel conda-forge pyngl*
- *conda install --channel conda-forge pynio*
- *conda install --channel conda-forge matplotlib*
- *conda install --channel conda-forge cartopy*

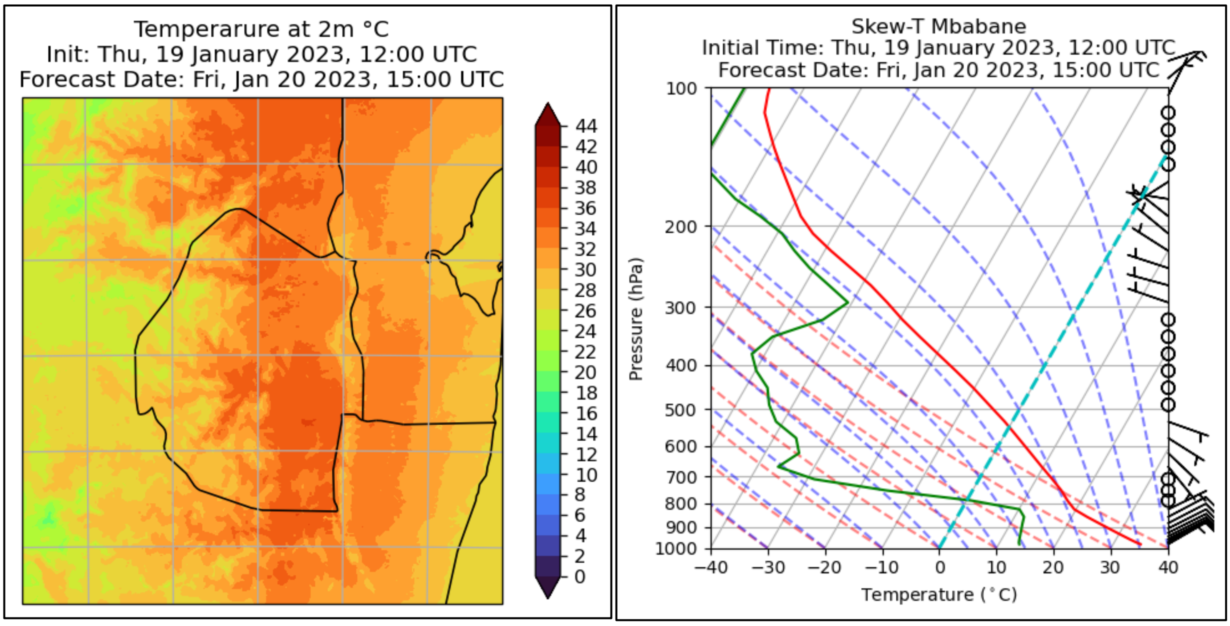
2.4 First weather forecast test

After installing all the components as described above, a test was made performing 48-h weather forecast on the HPC facility, initialising the model on 19th January 2023. The computation was made requiring only 48 cores (1 computational node) and took about 6 hours to perform the 48-h forecast.

Once the system will go full speed (192 cores, 4 nodes) the computational time is estimated to be around 1 hour and 30 minutes. Section 4 explains the roadmap needed to accomplish the full operational configuration of the system, expected by the end of May 2023.

2.4.1 Graphical output

There are three different types of graphical output that are produced by the postprocessing software: maps (at surface and pressure levels), skew-T diagrams and meteograms (temperature, precipitation, wind speed). The plots are produced by scripts written the possibility of output customisation is given. In Figure 1 some plots of the forecast test performed are shown.



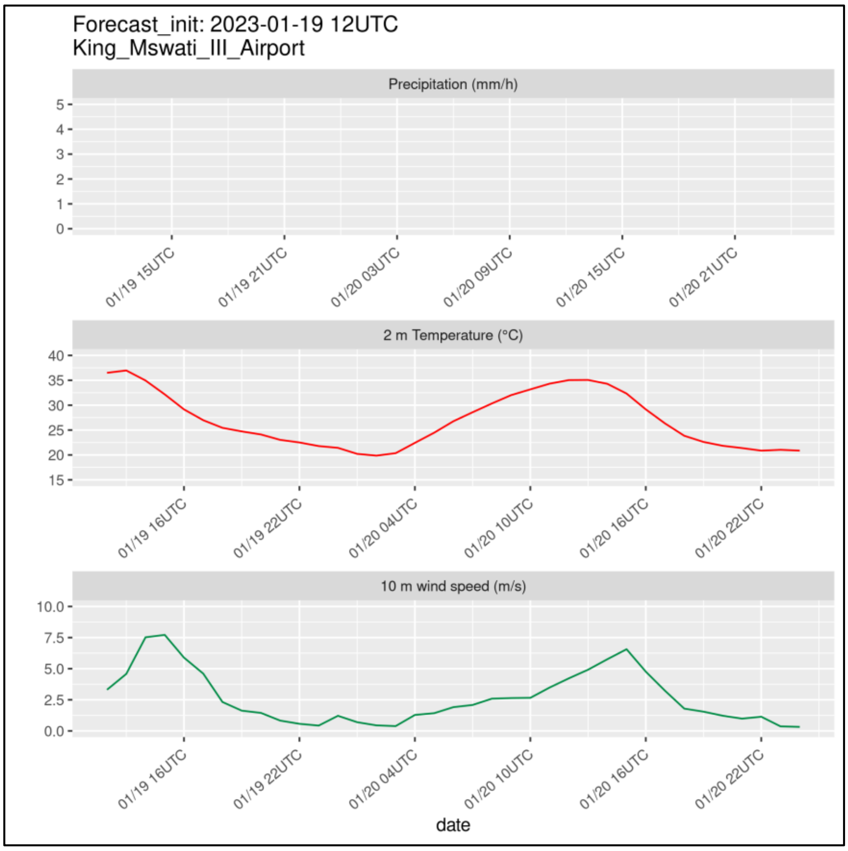


Figure 1. Sample weather forecast plots initialized on January 19th, 2023: 2 m temperature forecast and skew-T diagram predicted on 20th January, 12UTC at Mbabane (top left and right, respectively); meteogram of hourly precipitation, 2 m temperature and wind speed (from top to bottom) at King Mswati III International Airport (bottom panel).

3 USER GUIDE

As written in Section 2.2, two users were created to operate on the HPC machine OS: the administrator (*root*) and the one devoted to the automated execution of the weather forecasts (*eswmet*). The former is expected to operate on the whole system (OS management), while the latter must be used to manage the NWP codes and if some graphical output customisation is needed. In this case, a text editor (e.g., vim, nano, gedit) should be used to modify configuration files (see Section 3.2.3). A SSH client (e.g., MobaXterm) is needed to access the machine node01, typing the following command:

ssh eswmet@192.168.128.16

3.1 Folders tree

Under the *enwp* user home directory, the folder */home/enwp/forecast_system/* contains all the software (wrf and dedicated libraries) and scripts related to the various tasks to be performed by the NWP system. A README file explains the content of the different folders, which are organised as follow:

- **ESWATINI_NWP**:

  This folder is the core of the NWP system. It contains all the scripts needed for managing every step of the forecast chain. In any case, it is possible to recover the folder contents on *github* (*https://github.com/massimodisidoro/ESWATINI_NWP.git*).

  In turn, this folder contains the following sub directories:

  1. *bin*: WRF executables are stored here;
  2. *namelists*: configuration files for WRF;
  3. *postprocessing*: contains R and Python scripts and configuration files to produce the graphical output;
  4. *scripts*: shell scripts managing every forecast step, described more in detail in the next sub-section. This folder also contains a configuration file named

settings, which defines the environment and the parameters needed to perform the forecast. The file looks as follow:

```
#!/usr/bin/bash
#python path (from anaconda installation)
export python=/home/enwp/forecast_system/libs/anaconda3/wrf-python/bin/python
export omp_wps_threads=2
export nprocs_wrf=48  #192
export omp_wrf_threads=2
#once establishd the $root_dir, all other folders are automatically defined
# we establish the forecast system is under /home/enwp/forecast_system/ :
export dir_root="/home/enwp/forecast_system/"
export dir_master="$dir_root/ESWATINI_NWP"
export dir_wrf_code="$dir_root/wrf_code/WRF-4.2.2"
export dir_script="$dir_master/scripts"
export dir_namelist="$dir_master/namelists"
export dir_exe="$dir_master/bin"
export dir_post="$dir_master/postprocessing"

export dir_input="$dir_root/INPUT"
export dir_geogrid_files="$dir_input/geogrid_files"
export dir_input_meteo="$dir_input/gfs"
export dir_input_sst="$dir_input/sst"
export dir_wrf_static_data="$dir_root/wrf_static_data"
export dir_wrf_lookup_tables="$dir_wrf_code/run"

export dir_archive="$dir_root/OUT_ARCHIVE"

export gfs_res=0p50 #0p50 0p25  #gfs resolution fields in decimal defgrees
```

#forecast length in hours

export forecast_length=48 # forecast length in hours (at least 24h!)

export wrf_timestep=30 #model timestep in seconds

export adaptive_timestep=true # if true wrf_timestep is overrided (more stable)

export met_em_freq=3 #hours between met_em files

 # settings for graphical outputs

 # Note that maps are plotted starting at init_time +3 (model spin-up time)

export deltastep_maps=3 # setp in hours for the maps to be produced

 # please, edit the tslist file in the directory:

 # /home/enwp/forecast_system/ESWATINI_NWP/postprocessing

 # to choose the geographcal points where meteograms and

 # profiled must be automatically produced in output

- **INPUT**:

Here is stored the data in input to wrf model, that is:

1. *geogrid_files:* this folder contains two files, *geo_em.d02.nc* and *geo_em.d02.nc*, that were produced once and for all and contain static data interpolated to the model grids such as soil parameters, LAI (Leaf Area Index), Landuse, etc.
2. *gfs* and *sst:* here are stored the 2D and 3D meteorological fields downloaded by the system from the GFS global model to initialise WRF forecasts;

- **libs**:

this folder contains the locally (user *eswmet*) installed software;

- **libs_src**:

  here are stored the source codes of the locally installed software, including a README file with the description of all the configurations used to compile each package;

- **OUT_ARCHIVE**:

  In this folder the model outputs are stored (plots and WRF netcdf files). It is currently on the single node #1 disk space, but it will be set as a part of the server storage when the system will be fully operational (see Section 4);

- **SCRATCH_00UTC**:

  folder where WRF output and temporary files are produced during model run for the forecast initialised with 00UTC GFS fields;

- **SCRATCH_12UTC**:

  folder where WRF output and related temporary files are produced during model run for the forecast initialised with 12UTC GFS fields;

- **wrf_code**:

  here the source code of WRF, version 4.2.2 and its pre-processor WPD, version 4.2, are stored;

- **wrf_lookup_tables**:

  lookup tables coming with WRF code, needed runtime by the model

- **wrf_static_data**:

  contains all static data (e.g. orography, landuse, LAI, etc.) needed by WRF to build the computational grid.

3.2 Operational Chain description

3.2.1 Sequential tasks

The NWP system is composed of different steps performed sequentially by means of shell (bash) scripts located in the folder *ESWATINI_NWP/scripts/crontab_script*. Those scripts themselves call the correspondent scripts located in *ESWATINI_NWP/scripts/.* The sequence is illustrated below:

1. *01_submit_download_gfs.sh:* downloads the Global Forecasting System (GFS) meteorological fields (in grib2 format) from the NOMADS server (https://nomads.ncep.noaa.gov/), hosted at the National Centre for Environmental Prediction (NCEP) in the United States, and checks its availability and integrity, resubmitting the download in case of unexpected failures. The script prepares the environment and calls *01-download_gfs.sh* located in *ESWATINI_NWP/scripts/.*

2. *02_submit_ungrib.sh:* manages the grib2 GFS fields (executable: ***ungrib.exe***), properly formatting the data to be used by the WRF pre-processor WPS (WRF Preprocessing System). The operations are accomplished launching the script *02-ungrib.sh* located in ESWATINI*_NWP/scripts/.*

3. *03_submit_metgrid.sh:* manages the first step of the WPS tool (executable: ***metgrid.exe***), by interpolating the GFS fields horizontally over the two WRF computational grids. It calls the script *03-metgrid.sh* located in *ESWATINI_NWP/scripts/.*

4. *04_submit_real.sh:* manages the final WPS step (executable: ***real.exe***), consisting in the vertical interpolation of the fields on the sigma hybrid WRF levels. At the end of this stage,

initial and boundary conditions for the model are ready. The operation is performed launching the script *04-real.sh* located in *ESWATINI_NWP/scripts/.*

5. *05_submit_wrf.sh:* Finally, this script manages the submission of the forecast step on HPC nodes (executable: ***wrf.exe***) by means of a parallel job, launching the correspondent script *05-wrf.sh* located in *ESWATINI_NWP/scripts/.*

6. *06-postpc.sh:* Production of the graphical output by means of different scripts for skewT diagrams (***plot_skewt.py***), maps (***plot_figures.py***) and meteograms (***meteograms.R***). Such operation is performed launching the script *06-postpc.sh* located in *ESWATINI_NWP/scripts/.*

After the end of the forecast procedure, all the logs are stored in the OUT_ARCHIVE/logs folder and cleaning of temporary files is accomplished by the scripts *clear_archive.sh* and *clear_scratch.sh*, respectively.

3.2.2 Automated submission

The NWP system is set to operate twice a day, initialised with 00UTC and 12UTC GFS forecasts. The task is done in a completely automated way, considering that GFS fields for initialization are made available with approximately 4 hours delay respect to their reference time. This mean that 00UTC global fields are ready around 4UTC (06:00 Eswatini time) and 12UTC at 16UTC (18:00 Eswatini time). The automation of the various steps is managed by the *crontab* Linux tool, which configuration file can be edited from terminal with the command *"crontab -e*" (*crontab -l* to only show its content) and appears as follow:

 # 00UTC init forecast

*00 00 * * * clear_scratch.sh*

*30 00 * * * submit_clear_archive.sh*

*00 06 * * * 01_submit_download_gfs.sh*

*15 06 * * * 02_submit_ungrib.sh*

*20 06 * * * 03_submit_metgrid.sh*

*25 06 * * * 04_submit_real.sh*

*28 06 * * * 05_submit_wrf.sh*

*30 08 * * * 06_submit_postpc.sh*

 # 12UTC init forecast

*00 12 * * * clear_scratch.sh*

*30 12 * * * submit_clear_archive.sh*

*00 18 * * * 01_submit_download_gfs.sh*

*15 18 * * * 02_submit_ungrib.sh*

*20 18 * * * 03_submit_metgrid.sh*

*25 18 * * * 04_submit_real.sh*

*28 18 * * * 05_submit_wrf.sh*

*30 20 * * * 06_submit_postpc.sh*

In the crontab file, the first two column represent the local minutes and hours, respectively, at which the script (last column) is launched, every day (* * *). In the scheduled procedure, the first two steps are devoted to cleaning the working directory from various temporary files and to clean the OUT_ARCHIVE folder from files older than a user given period (default value is 365 days: to change the value, just modify the *clear_older_days* variable within the configuration file *settings*). Fore an extensive explanation on crontab usage, please refer to *https://crontab.guru/examples.html.*

3.2.3 Customisations

The graphical output has a certain level of customisation. Concerning meteograms, the user can add several points of interest to be plotted in output using a text editor to modify the configuration file *tslist* located in the *ESWATINI_NWP/postprocessing* folder. The file appears as follow:

```
File  Edit  View  Search  Terminal  Help
#-------------------------------------------------#
# 24 characters for name | pfx |  LAT   |   LON   |
#-------------------------------------------------#
King_Mswati_III_Airport   Airpt -26.357   31.717
Mbabane                   Mbab  -26.315   31.133
```

By default, two locations are present (King Mswati III International Airport and Mbabane). To add (or delete) geographical locations a README_TSLIST file, in the same folder, explains how to modify *tslist*. The very important thing to keep in mind is to maintain the column alignment for each field. As stated in the README_TSLIST:

#Add as many records as many meteograms are desired in output

Each record must provide:

#name: max 24 character length, **no blank spaces within the name**

#pfs: a prefix, to be used in output files (max 5 characters)

#LAT: latitude (total 7 characters)

#LON: longitude (total 8 characters)

NOTE: !!! be careful not to get outside the

start and the end of each field

Keep the same alignment as the default records

Mababane and King Mswati III Airport !!!!

Once the *tslist* file is modified, the following weather forecast will produce meteogram plots for each location specified in *tslist*. If this doesn't happen, something is wrong in the *tslist* formatting, and the editing must be done again, carefully following the formatting indications.

The same *tslist* file drives also the skew-T plots generation, that is for every location defined in *tslist* a correspondent skew-T plot will be produced.

Finally, concerning 2 dimensional fields, map plots are created by means of the python code by *plot_figures.py*. It comes with a configuration file, named *var.yaml*, containing the description of which variables are plotted in output. The file header contains an explanation of the content:

```
# This is a cofiguration file used by plot_figures.py
# to produce map plots.
# NOTE: the file var.yaml is the one considered by the postprocessing code
# to produce the plots; the file var.yaml_all_vars contains the
# configuration to plot a very complete set of fields: the user can
# modify vars.yaml considering the whole var.yaml_all_vars or a subset
#
# The configuration is based on different keys.  For every field we have:
#
# 'wrf_name' : if any, is the name of the wrf variable
#            (https://wrf-python.readthedocs.io/en/latest/diagnostics.html
#             for reference)
#
# 'type'     : can be 'prec' (precipitation field)
#                   'surf' (surface field)
#                   'lev'  (upper level field)
#
# 'pressure level':   0 (for surface fields)
#                   integer number (press level in hPa: e.g. 850)
#
# 'varname'      :  it must be one of the 'varname' values in def_vars.py
#                   (e.g. mslp). It defines contour levels to be used.
#
# 'contour'      :  False (no contour lines will be plotted)
#                   True  (contour lines will be plotted)
#
```

```
# 'overlap fields' : Can be used if one wants to plot two fields overlapped
#                   Note: thie feature is not well tested, so that
#                   it is advisable to set overlap_fields to False
#                   as well as wrf_name2 to 'none'
# 'windvectors'   : True (if when wind field has to be plotted)
#                  False (for other fields)
#
# 'wrf_name2'     : 'none' (not well tested. See 'overlap fields')
#
# 'varname_additional' : 'none' (related to 'overlap fields', not  tested)
#
# 'title'        : A string containing the title that appears on the map
#
# 'colormap'      : 'prec'  (dedicated colorbar for precipitation fields)
#                  'turbo' (predefined colormap in the matplotlib library)
```

Such a yaml file may be tricky to manage. In this respect, the file *var.yaml_all_vars*, containing all possible quantities that can be generated, may help the user to put additional fields in output, simply adding a given block to the *var.yaml* file. For example, if one wants to add temperature at 300 hPa map in output, they have to select the correspondent block from *var.yaml_all_vars*, as shown below

```
File  Edit  View  Search  Terminal  Help
    'colormap'          : 'turbo'
'T300':
    'wrf_name'          : 'tc'
    'type'              : 'lev'
    'pressure_level'    : 300
    'varname'           : 't300'
    'contour'           : False
    'overlap_fields'    : False
    'windvectors'       : False
    'wrf_name2'         : 'none'
    'varname_additional'     : 'none'
    'title'             : 'Temperature at 300 hPa (°C)'
    'colormap'          : 'turbo'
'T250':
    'wrf_name'          : 'tc'
    'type'              : 'lev'
    'pressure_level'    : 250
    'varname'           : 't250'
    'contour'           : False
    'overlap_fields'    : False
    'windvectors'       : False
    'wrf_name2'         : 'none'
    'varname_additional'     : 'none'
    'title'             : 'Temperature at 250 hPa (°C)'
    'colormap'          : 'turbo'
```

and copy it at the bottom of the *var.yaml* file. If all is done correctly, the following forecast will produce the additional map in output. Note that the user can also set the levels of the fields to be plotted. Those are defined in the file *def_vars.py*: its header contains an explanation of how to edit the file itself if different levels are needed for a given quantity instead of the current defined ones.

4 FINAL SYSTEM SETUP

During the ENEA team working mission in January 2023 the NWP system was installed on the HPC facility as described in the previous Sections, but the system was not put in operational mode because of the lack of a UPS unit that can assure the power stability. This crucial aspect suggested us to freeze the system status as it was at the end of January, power off the machine and wait for the UPS startup in the computing room. Without such a device, the possibility of power failure leading to potential hardware break is too high. Moreover, investigating the logs of the storage server during the mission, some power instabilities occurred in the previous period were recorded, which were likely the cause of two out of 24 10TB disks failure.

During the first week of May 2023, Eswatini Meteorological Services staff communicated that the UPS device was installed in the room, therefore the ENEA team started remote working sessions to perform the final settings on the system, with the aim to put it in daily operational mode at full speed using all the 4 computational nodes and the 192 available cores, including the storage server space. The following actions were performed:

1. Installation of OS in the storage server.
2. Installation of Mellanox Infiniband package to manage and setting up the low latency network.
3. Formatting of the disk array in a proper way to obtain a unique file system (disk space) that can be accessed from all the computing nodes.
4. Setting up an IP over IB connection to export the filesystem via NFS to all the nodes, taking advantage of the low latency network high speed.
5. Make the necessary modifications to the NWP procedures in order to launch the parallel jobs on the 4 nodes.
6. Test the system in automated mode activating the crontab tool to perform two daily 48 hour forecasts initialized at 00UTC and 12 UTC, respectively.

At the beginning of June 2023 all the above points were finalised and the NWP system was delivered to Eswatini Meteorological Services in operational mode.

5 BIBLIOGRAPHY

Iannone, F., Ambrosino, F., Bracco, G., De Rosa, M., Funel, A., Guarnieri, G., Migliori, S., Palombi, F., Ponti, G., Santomauro, G., Procacci, P., 2019. CRESCO ENEA HPC clusters: a working example of a multifabric GPFS Spectrum Scale layout, in: 2019 International Conference on High Performance Computing Simulation (HPCS). Presented at the 2019 International Conference on High Performance Computing Simulation (HPCS), pp. 1051–1052. https://doi.org/10.1109/HPCS48598.2019.9188135

Skamarock, W.C., Klemp, J.B., Dudhia, J., Gill, D.O., Liu, Z., Berner, J., Wang, W., Powers, J.J., Duda, M.G., Barker, D., Huang, X.-Y., 2019. A DESCRIPTION OF THE ADVANCED RESEARCH WRF MODEL VERSION 4.1 (No. NCAR/TN-556+STR).

6 ANNEX I

PACKING LIST

Packing list no.: 28976301
Page no.: 1
Date: 09-03-2021
Our ref.: 140-289763
Your ref.: **UNDP1-0000099644**

Consignee:
UNDP Eswatini (Swaziland)
MBABANE HH H100
Swaziland

Deliver to:

Danoffice IT

Delivery overview

Item	Product no.	Description	Ordered	Delivered	Shippede	Outstanding
	P9K08A	HPE 42U 600x1075mm G2 Kitted Advanced Shock Rack	1			
	P9K08A#001	HPE Factory Express Base Racking Service	1			
	867158-B21	HPE Apollo r2600 Gen10 24 SFF Premium CTO Chassis	1			
	867158-B21#0D1	Factory Integrated	1			
	867158-B21#B19	HPE Apollo r2600 24SFF-Prem CTO Chassis	1			
	867055-B21	HPE ProLiant XL170r Gen10 1U Node Configure-to-order Server	4			
	867055-B21#0D1	Factory Integrated	4			
	867055-B21#B19	HPE ProLiant XL170r Gen10 CTO Svr	4			
	P19270-L21	HPE XL1x0r Gen10 Xeon-G 5220R FIO Kit	4			
	P19270-B21	HPE XL1x0r Gen10 Xeon-G 5220R Kit	4			
	P19270-B21#0D1	Factory Integrated	4			
	P00922-H21	HPE 16GB 2Rx8 PC4-2933Y-R Smart Kit (OE)	48			
	P00922-H21#0D1	Factory Integrated	48			
	P04560-H21	HPE 480GB SATA RI SFF SC PM883 SSD	4			
	P04560-H21#0D1	Factory Integrated	4			
	874296-B21	HPE XL1x0r Gen10 Left Low Profile Riser Kit	4			
	874298-B21#0D1	Factory Integrated	4			
	825110-H21	HPE IB EDR/EN 100Gb 1P 840QSFP28 Adptr	4			
	825110-H21#0D1	Factory Integrated	4			
	874300-B21	HPE XL170r Gen10 P1 LP Riser Kit	4			
	874300-B21#0D1	Factory Integrated	4			
	868227-B21	HPE XL1x0r Eth 10Gb 2p 568FLR-MMT Adptr	4			
	868227-B21#0D1	Factory Integrated	4			
	BD505A	HPE iLO Advanced 1-server LTU with 3 years support	4			
	BD505A#0D1	Factory Integrated	4			
	874305-B21	HPE XL170r Gen10 S100i SATA Cable Kit	4			
	874305-B21#0D1	Factory Integrated	4			
	784308-B21	HP FIO Enable Kit Hamer BIOS Setting	4			
	876935-B21	HPE 1800W-2200W Flex Slot Platinum Hot Plug Power Supply Kit	2			
	876935-B21#0D1	Factory Integrated	2			
	R1C68A#0D1	**Factory Integrated**	2			
	880189-B21	HPE r2x00 Gen10 1800-2200W Redundant PS Enablement Kit	1			
	880189-B21#0D1	Factory Integrated	1			
	874308-B21	HPE r2x00 Gen10 Redundant Fan Module Kit	1			
	874308-B21#0D1	Factory Integrated	1			
	740713-B21	HP t2500 Strap Shipping Bracket	1			
	740713-B21#0D1	Factory Integrated	1			
	822731-B21	HPE 2U Shelf-Mount Adjustable Rail Kit	1			
	822731-B21#0D1	Factory Integrated	1			

SWAZILAND REVENUE AUTHORITY
CUSTOMS - LEGISLATIVE
11 MAR 2021
6
DEPARTMENT OF CUSTOMS

Delivery term: DAP Incoterms 2020
Delivery method: Air freight

PACKING LIST

Packing list no.: 28976301
Page no.: 2
Date: 09-03-2021
Our ref.: 140-289763
Your ref.: UNDP1-0000099644

Consignee:
UNDP Eswatini (Swaziland)
MBABANE HH H100
Swaziland

Deliver to:

Danoffice IT

Delivery overview

Item	Product no.	Description	Ordered	Delivered	Shippede	Outstanding
	P9Q38A	HPE G2 Basic 3.6kVA/C13 C19 WW PDU	2			
	P9Q38A#0D1	Factory Integrated	2			
	BW932A	HPE 600mm Jb Rack Stabilizer Kit	1			
	BW932A#B01	Include with complete system	1			
	867959-B21	HPE DL360 Gen10 8SFF CTO Server	1			
	867959-B21#B19	HPE Europe - Multilingual Localization	1			
	P02580-L21	HPE DL360 Gen10 Xeon-S 4214 FIO Kit	1			
	P00922-B21	HPE 16GB 2Rx8 PC4-2933Y-R Smart Kit	2			
	P00922-B21#0D1	Factory Integrated	2			
	P18420-B21	HPE 240GB SATA 6G RI SFF SC 3yr Wty SSD	2			
	P18420-B21#0D1	Factory Integrated	2			
	825110-B21	HPE InfiniBand EDR/Ethernet 100Gb 1-port 840QSFP28 Adapter	1			
	825110-B21#0D1	Factory Integrated	1			
	804326-B21	HPE Smart Array E208i-a SR Gen10 12G SAS Modular Controller	1			
	804326-B21#0D1	Factory Integrated	1			
	700759-B21	HP FlexFabric 10Gb 2P 533FLR-T Adptr	1			
	700759-B21#0D1	Factory Integrated	1			
	865408-B21	HPE 500W Flex Slot Platinum Hot Plug Low Halogen PS Kit	2			
	865408-B21#0D1	Factory Integrated	2			
	BD505A	HPE iLO Advanced 1-server LTU with 3 years support	1			
	BD505A#0D1	Factory Integrated	1			
	874543-B21	HPE 1U Gen10 SFF Easy Install Rail Kit	1			
	874543-B21#0D1	Factory Integrated	1			
	P07244-B21	HPE Apollo 4200 Gen10 24LFF CTO Svr	1			
	P07244-B21#B19	HPE Apollo 4200 Gen10 24LFF CTO Svr Europe -Multilingual Loc	1			
	P07908-L21	HPE Apollo 4200 Gen10 Xeon-Gold 5215 FIO Processor Kit	1			
	P00920-K21	HPE 16GB 1Rx4 PC4-2933Y-R Smart Kit	2			
	P00920-K21#0D1	Factory Integrated	2			
	P07248-B21	HPE Apollo 4200 Gen10 Rear 2SFF/PCIe Kit	1			
	P07248-B21#0D1	Factory Integrated	1			
	P18420-K21	HPE 240GB SATA RI SFF SC MV SSD	2			
	P18420-K21#0D1	Factory Integrated	2			
	P09149-K21	HPE 10TB SAS 7.2K LFF LP He 512e DS HDD	24			
	P09149-K21#0D1	Factory Integrated	24			
	656596-B21	HP Ethernet 10Gb 2P 530T Adptr	1			

SWAZILAND REVENUE AUTHORITY
CUSTOMS - LEGISLATIVE
11 MAR 2021
6
DEPARTMENT OF CUSTOMS

Delivery term: DAP Incoterms 2020
Delivery method: Air freight

PACKING LIST

DanofficeIT

Packing list no.: 28976301
Page no.: 3
Date: 09-03-2021
Our ref.: 140-289763
Your ref.: UNDP1-0000099644

Consignee:
UNDP Eswatini (Swaziland)
MBABANE HH H100
Swaziland

Deliver to:

Delivery overview

Item	Product no.	Description	Ordered	Delivered	Shippede	Outstanding
	656596-B21#0D1	Factory Integrated	1			
	P01367-B21	HPE 96W Smart Storage Battery with 260mm Cable Kit	1			
	P01367-B21#0D1	Factory Integrated	1			
	804394-B21	HPE Smart Array E208i-p SR Gen10 Ctrlr	1			
	804394-B21#0D1	Factory Integrated	1			
	869081-B21	HPE Smart Array P408i-a SR Gen10 12G SAS Modular LH Ctrl	1			
	869081-B21#0D1	Factory Integrated	1			
	825110-B21	HPE InfiniBand EDR/Ethernet 100Gb 1-port 840QSFP28 Adapter	1			
	825110-B21#0D1	Factory Integrated	1			
	865414-B21	HPE 800W FS Plat Hot Plug LH Power Supply Kit	2			
	865414-B21#0D1	Factory Integrated	2			
	BD505A	HPE ILO Advanced 1-server LTU with 3 years support	1			
	BD505A#0D1	Factory Integrated	1			
	822731-B21	HPE 2U Shelf-Mount Adjustable Rail Kit	1			
	822731-B21#0D1	Factory Integrated	1			
	P09656-B21	HPE Smart Array E/P SR FIO Controller Mode for Rear Storage	1			
	JL581A	HPE Aruba 8320 48 T/6 40 X472 5 2 Bundle	1			
	JL581A#ACE	HPE Aruba 8320 48 T/6 40 X472 5 2 Bundle DK ENG	1			
	JL483B	HPE Aruba X474 4-post Rack Kit	1			
	834979-B22	Mellanox IB EDR v2 36p RAF Managed Swch	1			
	HA113A1	HP Installation Service	1			
	HA113A1#58Y	HPE Apollo 2000/4200 Install Service	1			
	834973-B22	HPE 1M IB EDR QSFP Copper cable	4			
	834973-B23	HPE 1.5M IB EDR QSFP Copper cable	2			
	SFTP702R	MicroConnect RJ45 patch cord S/FTP (PiMF),	15			
	SFTP702B	MicroConnect RJ45 patch cord S/FTP (PiMF),	10			
	Installation	Installation in Rack	1			
		Packing	1			
		Air freight and DAP charges (Consignee must provide duty/tax exemption documents prior to arrival to minimize risk of delays and storage charges)	1			
		Insurance	1			

SWAZILAND REVENUE AUTHORITY
CUSTOMS - LEGISLATIVE
11 MAR 2021
6
DEPARTMENT OF CUSTOMS

Delivery term: DAP Incoterms 2020
Delivery method: Air freight