Installation of AMSTer

(FOR Mac OS DISTROS)

Installation guide by:
N. d'Oreye, D. Smittarello, M. Jaspard, G. Celli – ECGS
B. Smets – RMCA

https://github.com/AMSTerUsers/AMSTer_Distribution

Version 6.2.3: Aug 07 2025

Preliminary information

AMSTer (Automated SAR & InSAR Mass processing Software for Multidimensional Time series) was initially named MasTer. Some publications refer to that former name.

AMSTer is composed of 3 main elements: **AMSTer toolbox**, **AMSTer Engine** and **MSBAS**.

AMSTer toolbox is a set of mostly shell scripts aiming at processing automatically a large number of interferometric pairs and feeding and running the **MSBAS** processor [Samsonov and d'Oreye, 2012, 2017; Samsonov et al., 2017, 2020] in order to obtain the desired deformation maps and time series in vertical and horizontal components. Geocoded amplitude, coherence, interferometric phase and deformation maps are computed using the **AMSTer Engine**, a command line InSAR processor derived from the Centre Spatial de Liege (CSL) InSAR Suite (CIS) [Derauw, 1999; Derauw et al, 2019].

This guide explains the installation procedure for **AMSTer (AMSTer tool, AMSTer Engine, MSBAS)**, on a Mac computer. It does not replace the manual file provided with the software distribution (AMSTer_Manual_x.x.docx), but rather provide much more details to allow any user with less-advanced knowledge of the Mac environment to properly install the software suite.

The installation procedure has been last tested with the **AMSTerEngine_v20241108** package, msbasv3 (20190121), msbasv4 (20201009) and shell scripts dated from November 2024.

Sources and codes are licensed under CC BY-NC-SA 4.0 (Attribution-NonCommercial-ShareAlike 4.0 International) and freely available for noncommercial use from Github:
https://github.com/AMSTerUsers/AMSTer_Distribution
If you are not granted the access to the repository or the discussions group, please send an email to ndo@ecgs.lu.

Download:
Either from the **distribution repository** (all in one package):
- https://github.com/AMSTerUsers/AMSTer_Distribution

Or from the **development (private) repositories**:
- the documentation and manuals from ndoreye/DOCS
- the scripts from ndoreye/SCRIPTS_MT
- the MSBAS sources tuned for AMSTer Toolbox from ndoreye/MSBAS (also available directly from S. Samsonov; see manual)
- the AMSTerEngine sources from ndoreye/_Sources_AE (also available directly from D. Derauw; see manual)

Note: Gitkraken is a convenient interface for Github but is not mandatory.

This installation procedure has been tested on:
- **MacOS Monterey version 12.6 and Sonoma 14.2**

though former installations were operational already on Mojave (V10.14.6) and former MacOS versions.

It is assumed that the user has the administrator rights on the computer and can provide the root privilege when the "sudo" command is executed in the terminal.

In this installation guide, each command line that has to be written and executed in a terminal is written in *italic blue*. Files names are in blue. Paths are given in green.

Note:

A script exists to assist you making the installation (at least for recent Mac OS X such as Monterey). See
…/SAR/AMSTer/SCRIPTS_MT/zz_Utilities_MT/***AMSTer_install.sh***
(in distribution repository), or
…/SAR/AMSTer/SCRIPTS_MT/zz_Utilities_MT/***AMSTer_install.sh***
(in development repository).

Read carefully all the text and questions displayed by the script.

For older Mac OS versions, you may need to adapt the script or proceed manually by adapting the procedure below.

Note that the ***AMSTer_install.sh*** installation script may be run several times (either to correct a problem during the installation or to perform an update of any element at any time). However, if some elements were installed manually (eg Gitkraken, Fiji or QGIS…), the script may not properly detect them as they may have been installed in an expected. This must not be a problem though.

Kind reminder: The software is provided for free. To help us carrying on with the development of the tool, we kindly ask you to cite the **AMSTer Software** in publications that contains results produced by the toolbox thanks to at least the most recent references from below:

For AMSTer

- Derauw, D., N. d'Oreye, M. Jaspard, A. Caselli, and S. Samsonov (2020), Ongoing automated ground deformation monitoring of Domuyo - Laguna del Maule area (Argentina) using Sentinel-1 MSBAS time series: Methodology description and first observations for the period 2015-2020.,
  Journal of South American Earth Sciences, 104, 102850,
  doi:10.1016/j.jsames.2020.102850.
- d'Oreye N., Derauw, D., S. Samsonov, M. Jaspard, and D. Smittarello (2021), MasTer: a full automatic multi-satellite InSAR mass processing tool for rapid incremental 2D ground deformation time series. Proc. IEEE IGARSS21, Brussels, July 2021.
- Smittarello, D., d'Oreye, N., Jaspard, M., Derauw, D., & Samsonov, S. (2022).
  Pair selection optimization for InSAR time series processing. *Journal of Geophysical Research: Solid Earth, 127, e2021JB022825.*
- d'Oreye N., Derauw, D., S. Samsonov, M. Jaspard, D. Smittarello, G. Celli (2023), AMSTer: SAR & InSAR Automated Mass processing Software for Multidimensional time series. Instruction manual V6.0, Nov. 2023. 237 pp.

For AMSTer Engine

- Derauw, D., (1999). Phasimétrie par Radar à Synthèse d'Ouverture; théorie et applications. PhD Dissertation, University of Liège, 141 pages.
- Libert, L., Derauw, D., d'Oreye, N., Barbier, C., Orban, A., (2017). Split-band interferometry-assisted phase unwrapping for the phase ambiguities correction. Remote Sensing, 9(9), 879.

For MSBAS

- Samsonov, S., d'Oreye, N., (2012). Multidimensional time-series analysis of ground deformation from multiple InSAR data sets applied to Virunga Volcanic Province. Geophysical Journal International 191(3), 1095–1108. doi: 10.1111/j.1365-246X.2012.05669.x.
- Samsonov, S., d'Oreye, N., (2017). Multidimensional Small Baseline Subset (MSBAS) for two-dimensional deformation analysis: Case study Mexico City. Canadian Journal of Remote Sensing 82(6), 1–12. doi: 10.1080/07038992.2017.1344926
- Samsonov S., W. Feng, A. Peltier, H. Geirsson, N. d'Oreye, KK. F.Tiampo (2017). Multidimensional Small Baseline Subset (MSBAS) for volcano monitoring in two dimensions: opportunities and challenges. Case study Piton de la Fournaise volcano. Journal of Volcanology and Geothermal Research, Vol 344, 121-138, https://doi.org/10.1016/j.jvolgeores.2017.04.017
- Samsonov, S., A. Dille, O. Dewitte, F. Kervyn, and N. d'Oreye (2020). Satellite interferometry for mapping surface deformation time series in one, two and three dimensions: A new method illustrated on a slow-moving landslide. Engineering Geology, (266), 105471, doi:10.1016/j.enggeo.2019.105471.items

For DORIS

- Kampes, B.M., Hanssen, R.F., Perski, Z., (2003). Radar interferometry with public domain tools. In: Third International Workshop on ERS SAR Interferometry, FRINGE03, Frascati, Italy, 1-5 Dec. 2003, 6 pages.

0) Prepare the work

The tools will need the following directories:

- /opt/local/bin
- $HOME/SAR/EXEC
- $HOME/SAR/AMSTer/DOC
- $HOME/SAR/AMSTer/AMSTerEngine/_Sources_AE/Older
- $HOME/SAR/AMSTer/MSBAS
- $HOME/SAR/AMSTer/SCRIPTS_MT

Where $HOME/ is obviously your home directory.

It is also advised to source your **.barshrc** in your **.bash_profile** as all the config will be set in **.bashrc.**

If you do not have these files, create them and ensure you can write and execute them.

If you have old AMSTer installation parameters in your **.bashrc**, it is advised to remove them and write new ones based on the instructions below.

1) Installing some external components

The following tools and apps are not mandatory but experience shows that it may greatly help the user.

Macports:

- Download latest version from www.macports.org > Available Downloads:
  - E.g., for Monterey, take MacPorts-2.7.2-12-Monterey.pkg
  - Open and install
  - And/or:
    - *sudo port selfupdate*
    - *sudo port upgrade outdated*
  - If your OS is 10.15 or more recent, it is required to change default shell Zsh with bash for scripts compatibility issues:
    chsh -s /bin/bash

GitKraken :

Although not mandatory, Gitkraken is a useful tool to sync with the last versions of AMSTer Toolbox.

- Download GitKraken (from https://www.gitkraken.com) and connect e.g. via Github
- Create profile in GitKraken
- Launch GitKraken
- Clone Repo > with URL :
  - Where to Clone: $HOME/SAR/
  - Repo to clone: URL to AMSTerSoftware_Distribution

To display raster figure: XnView, GIMP or Graphic Converter

- my favourite is Graphic Converter though it is not free…
- To install GIMP:
  - *sudo port install "gimp2"*

A text editor: BBEdit (not free though)

- Get it from https://www.barebones.com/products/bbedit/download.html
  or any other convenient editor that you like

Cyberduck

- Get it from https://cyberduck.io
  or any other convenient ftp software that you like

gnu fortan (not mandatory):

- Get it from https://github.com/fxcoudert/gfortran-for-macOS/releases
- install

GMT and GDAL:

GMT, GDAL and some associated utilities are required for plots, images and GIS manipulations.

- Using MacPorts, at Terminal, type:
  - *sudo port selfupdate* (to be sure)
  - *sudo port install gdal +hdf5 +netcdf +openjpeg*
  - *sudo port install gmt6*
  - *sudo port install graphicsmagick ffmpeg*
  - search for installation location: *which gmt6*
  - link that *gmt6* to /opt/local/bin/*gmt* for portability

  Note: if GDAL is crashing (e.g. after updates when most recent version are not 100% compatible yet), you may try installing homebrew version. This however may require to recompile msbas and/or AMSTer Engine sources after an OS update as homebrew does not manage libraries the same way as MacPorts.

To display geocoded products: QGIS

- Download latest version from https://qgis.org/en/site/forusers/download.html
- note some interesting plugins are:
  - point sampling tool
  - PointConnetor
  - Profile tool
  - Qdraw
  - QuickMapServices
  - RasterDataPlotting
  - Serval
  - Temporal/Spectal Profile Tool
  - Value Tool

To quick look scripts with colours when hitting the space bar: QLColorCode

  - *sudo port install QLColorCode*
  - *reboot*

2) Installing required tools

AMSTer needs some pre-existing software and libraries to work properly. This is what we call the “dependencies”. So, we first need to install them. These installations are explained here, sorted by type of installation procedure.

bashrc:

- Open Terminal
- Type: *touch .bashrc*
- Type: *nano .bashrc*
- Copy/paste infos from example_bashrc.txt eg using nano editor
- Adapt to needs (see below)
- Exit nano by Crtl-X > Y
- Create a .bash_profile that will use your .bashrc. Type: *touch .bash_profile*
- Type: *nano .bash_profile*
- Copy/paste the following lines :

```
if [ -r ~/.bashrc ] ; then
   source ~/.bashrc
fi
```

- Exit nano by Crtl-X > Y
- Type:

```
chmod 700 .bashrc
chmod 700 .bash_profile
```

- Reboot

Note: for recent Mac, default shell is not bash. See here how to get it back to default bash: https://www.howtogeek.com/444596/how-to-change-the-default-shell-to-bash-in-macos-catalina/

Xcode:

- Install Xcode (required for compilations below). In Terminal type: *xcode-select –install*

Libraries & modules:

- Install libraries and modules using macports:
  - *Sudo port install clang-20* (Version 20 is mandatory from AMSTerEngine_2025085)
  - *sudo port install fftw-3-long*
  - *sudo port install fftw-3-single*
  - *sudo port install hdf5*
    Note: At compilation, a note is displayed reading:
    Mac users may need to set the environment variable "HDF5_USE_FILE_LOCKING" to the five-character string "FALSE" when accessing network mounted files. This is an application run-time setting, not a configure or build setting. Otherwise errors such as "unable to open file" or "HDF5 error" may be encountered.
    Not sure what to do out of it yet though…
  - *sudo port install tiff*
  - *sudo port install libgeotiff*
  - *sudo port install libxml2*
  - *sudo port install lapack*
  - *sudo port install libomp-devel*
  - *sudo port install ImageMagick*
  - *sudo port install gdal +libkml*
  - *sudo port install parallel*

GNU functions and utilities:

Mac version of some tools such as *sed* or *awk* etc do not have exactly the same syntax as their gnu version. For the sake of portability (and because some options are not available for Mac versions), we use here gnu tools:

- At Terminal:
  - *sudo port install gsed*
  - *sudo port install gawk*
    Note: At compilation, a note is displayed reading:
    readline support has been removed from gawk. If you need to run gawk interactively, install rlwrap:
    sudo port install rlwrap
    and run gawk using rlwrap:
    rlwrap gawk ...
  - *sudo port install coreutils* (i.e. for gdate, gstat)
  - *sudo port install findutils* (i.e. for find)
  - *sudo port install grep* (i.e. for ggrep)
  - *sudo port install wget* (needed i.e. to download the S1 orbits)
  - *sudo port install curl*
- Check that all these gnu functions are stored in the /opt/local/bin directory. If not, move them there. That directory MUST be defined as ${PATHGNU} state variable in your .bashrc. At Terminal, type
  - *which gsed*
  - *which gawk*
  - *which gdate*
  - *which gstat*
  - *which gfind*
  - *which ggrep*

  If answer to all these lines is not the same as echo ${PATHGNU}, edit your .bashrc and change ${PATHGNU} state variable. It must be something like
  export PATHGNU=/opt/local/bin
  Don't forget that a reboot (or source .bashrc) is required for changes in .bashrc to be taken into account.
  Note: ensure that /opt/local/bin is in your path before the /usr/bin to ensure that it would in priority take into account your GNU commands in case of doubt.

- Just in case one of these functions would be called without its g-name in a script, it is recommended to have them in ${PATHGNU} with both names, that is with and without heading g (that is **sed** and g**sed**, **awk** and **gawk**, **grep** and **ggrep**, **date** and **gdate**, **stat** and **gstat**. This can be simply done with a link. In Terminal, type:
  - *cd* ${PATHGNU}
  - *ln -s gsed sed*
  - *ln -s gawk awk*
  - *ln -s gdate date*
  - *ln -s gstat stat*
  - *ln -s gfind find*

  And to be sure…(I know it is overkill, but one never know…):
  - *ln -s ggrep grep*
  - *ln -s gseq seq*
  - *ln -s guniq uniq*
  - *ln -s greadlink p readlink*
  - *ln -s gdu du*
  - *ln -s gxargs xargs*

  Check also that *wget* is ${PATHGNU}

For some plotting applications: Java

Install it with :

- *sudo port install jdk20*

or manually:

- see http://www.java.com
- follow instructions to install Java

Notes:

- if some scripts crash with the following message, re-install java and then re-install Fiji:
  The operation couldn't be completed. Unable to locate a Java Runtime that supports (null).

- One may need to define JAVA_HOME state variable in the **.bashrc** for some application using it. Check where it is installed by typing:
  - *java -XshowSettings:properties -version 2>&1 > /dev/null | grep 'java.home' | cut -d = -f2- | cut -d " " -f2-*

  and update the **.bashrc** by adding a line with the answer of the line above:
  - *export JAVA_HOME=PATH_FROM_LINE_ABOVE*

Fiji/ImageJ:

- Download latest version from *https://imagej.net/software/fiji/downloads*
- Install
- In Finder, locate Fiji.app in your /Applications directory. Right click on it and click on "Show Package Contents"
- Navigate in Contents>MacOS and copy the path to executable (must be ImageJ-macosx)
- Update $PATHFIJI in your .bashrc. It must be something like
  export PATHFIJI=/Applications/Fiji.app/Contents/MacOS/

snaphu:

- Download latest version (e.g. snaphu-v2.0.6.tar) from https://web.stanford.edu/group/radar/softwareandlinks/sw/snaphu/
  and store it in in $HOME/SAR/EXEC.
- unzip it e.g. using *tar -zxvf*
- edit the Makefile in the /src directory to specify your compiler and the optimization flags. Ensure that makefile has no static option in following line
  CFLAGS = -O3 # -D NO_CS2
- compile: at Terminal in /src dir, type *make*
  (it seems that you can ignore warnings; most recent compilers are more sensitive)
- Move compiled snaphu from /bin into $HOME/SAR/EXEC
- Download also the snaphu_man1.txt if you need more info about snaphu and its configurations.
- Store the installed source e.g. in $HOME/SAR/EXEC/Sources_Installed.

cpxfiddle:

Not mandatory, but pretty convenient for creating raster files for quick looking at results.

- Download source from https://github.com/TUDelftGeodesy/Doris/tree/master/sar_tools
- Cut/paste source in new document with your favorite editor
- Save the file in $HOME/*SAR/EXEC/cpxfiddle.cc*
- In Terminal:
  - *cd* $HOME/*SAR/EXEC*
  - *make -n cpxfiddle*

  - *g++ -O -c -ocpxfiddle.o cpxfiddle.cc* (you can ignore warning)
  - *g++ -O cpxfiddle.o -o cpxfiddle*
- Clean sources in $HOME/*SAR/EXEC*/cpxfiddle.o
- Store the installed source cpxfiddle.cc e.g. in $HOME/SAR/EXEC/Sources_Installed
- Update $PATHTOCPXFIDDLE in your .bashrc. It must be something like
  export PATHTOCPXFIDDLE=/${HOME}/SAR/EXEC/

gnuplot:

- At Terminal:
  - *sudo port install gnuplot*
  - *which gnuplot*
- Ensure that the path provided by the former command displays the same path as $PATHGNU in your .bashrc. It must be something like
  export PATHGNU=/opt/local/bin

osascript:

- should be by default on your mac. Test at Terminal with : *osascript -h*

python:

- At Terminal:
  - *sudo port install python310*
  - *sudo port select --set python python310* (To make this the default Python or Python 3)
  - *sudo port select --set python3 python310* (To make this the default Python or Python 3)
  - *sudo port install py310-opencv4*
  - *sudo port install py-numpy*
  - *sudo port install py310-scipy*
  - *sudo port install py310-matplotlib*
  - *sudo port install py310-gdal*
  - *sudo port install py310-shapely*
- Check which version is used:
  - at Terminal, type: *python -c 'import sys ; print(sys.path)'*
  - It must answer something like:
    "['', '/opt/local/Library/Frameworks/Python.framework/Versions/3.10/lib/python310.zip', '/opt/local/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10', '/opt/local/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/lib-dynload', '/opt/local/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages']"
  - If a package such as numpy does not load, search where it is stored and check that it is in the same version as your python. For instance, numpy here must be in something like:
    /opt/local/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/site-packages/numpy
    If versions differ, change the default python version with same command as above or try to install an appropriate version of the package.
- Copy python to appropriate directory:
  For security reasons recent Mac OS now install python in /opt/local/bin instead of /usr/bin, and this is a good practice. If for any reason your system did install python in /usr/bin, please link it to /opt/local/bin/python as that is from where it will be called by all the scripts:
  - *which python3* (To know where it is installed, e.g. */usr/bin*)
  - *sudo ln -s /usr/bin*/pyhton3 */opt/local/bin/python3*
    (To make this the default Python, providing that the path is indeed */usr/bin*)
  - *sudo ln -s /usr/bin*/pyhton3 */opt/local/bin/python*
    (Just in case… i.e. to be sure that it will be called either as python or python3, providing of course that python3 is not already in /opt/local/bin)

- Install utm package, which is used ***in _LatLong2UTM.py*** script:
  - You must get python 3.10 and pip. To install pip, type at the Terminal:
    curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py

python3.10 get-pip.py

- Check pip. Since python is probably not in your PATH, type at the Terminal:
  $HOME/Library/Python/3.10/bin/pip --version
  It must answer something like
  pip 22.1.2 from /Users/doris/Library/Python/3.10/lib/python/site-packages/pip (python 3.10)
- Now install it. Type at the Terminal :
  $HOME/Library/Python/3.10/bin/pip install utm

- Install pyqt6 package, which is used eg. in ***AMSTerOrganizer.sh*** script:
  - */opt/local/bin/python3.10 -m pip install pyqt6*
  - */opt/local/bin/python3.10 -m pip install appscript*

- Install networkx package, which is used to analyse baseline plots:
  - */opt/local/bin/python3.10 -m pip install networkx*

- Install geopandas package, which is used to some scripts using kml infos:
  - */opt/local/bin/python3.10 -m pip install* geopandas

- Install scikit-gstat package, which is used to compute variograms:
  - */opt/local/bin/python3.10 -m pip install* scikit-gstat

- Install rasterio package, which is used to compute variograms:
  - */opt/local/bin/python3.10 -m pip install* rasterio

- Add the path to python (*/opt/local/bin/*) in the $PATH in your bashrc. To avoid any risk of using another version of python if badly called, add it in the beginning of the $PATH.

3) Installing AMSTer software

AMSTer consists of several packages that need to be obtained or downloaded from different sources:

- **AMSTerEngine (core parts: InSAR and MSBASTools)**:
  - See **Preliminary information/Sources and codes** above for Github instructions, or
  - contact: dderauw@ecgs.lu
- **AMSTer Toolbox (SCRIPTS_MT: Main Toolbox mass processing scripts)**:
  - See **Preliminary information/Sources and codes** above for Github instructions, or
  - contact ndo@ecgs.lu
- **MSBAS (2D or 3D inversion time series)**:
  - See **Preliminary information/Sources and codes** above for Github instructions, or
  - Download from: https://doi.org/10.4095/313749 or
  - Request code tuned for AMSTer with a make file adapted to Mac and Linux, contact ndo@ecgs.lu

  In case of problem, contact sergey.samsonov@canada.ca

3.1 Compile AMSTerEngine:

If executable files were provided (zip file),

- simply unzip it and copy the binaries in $HOME/SAR/AMSTer/AMSTerEngine
- then store the zipped file in $HOME/SAR/AMSTer/AMSTerEngine/_Sources_AE/Older/V*mostrecent*_AMSTererEngine where *mostrecent* is the date of the last version, which is in the name of the zip file. This date must be as YYYYMMDD and will be used by some scripts to track (and log) the version used during processing.
- Update $PATH in your .bashrc. It must be something like PATH=$PATH:/$HOME/SAR/AMSTer/AMSTerEngine

If you were provided with the source code (zip file), simply unzip it in $HOME/SAR/AMSTer/AMSTerEngine/_Sources_AE/Older/V*mostrecent*_AMSTererEngine where *mostrecent* is the date of the last version, which is in the name of the zip file. This date must be as YYYYMMDD and will be used by some scripts to track (and log) the version used during processing. Then,

- In Terminal, in $HOME/SAR/AMSTer/AMSTerEngine/_Sources_AE/Older/V*mostrecent*_AMSTererEngine/AMSTerEngineYYYYMMDD/InSAR/sources
  type: *make*
- Move all files from $HOME/SAR/AMSTer/AMSTerEngine/_Sources_AE/Older/V*mostrecent*_AMSTererEngine/AMSTerEngineYYYYMMDD/InSAR/bin
  in $HOME/SAR/AMSTer/AMSTerEngine
- In Terminal, in $HOME/SAR/AMSTer/AMSTerEngine/_Sources_AE/Older/V*mostrecent*_AMSTererEngine/AMSTerEngineYYYYMMDD/MSBASTools/sources
  type: *make*
- Move all files from $HOME/SAR/AMSTer/AMSTerEngine/_Sources_AE/Older/V*mostrecent*_AMSTererEngine/AMSTerEngineYYYYMMDD/MSBASTools /bin
  in $HOME/SAR/AMSTer/AMSTerEngine

- Clean (delete) the decompressed directories; keep only the zip file in:
  $HOME/SAR/AMSTer/AMSTerEngine/_Sources_AE/Older/V*mostrecent*_AMSTererEngine
- Update $PATH in your .bashrc. It must be something like PATH=$PATH:/$HOME/SAR/AMSTer/AMSTerEngine

Note that if you are provided with an updated version of AMSTer Engine (source code), you can simply update it by copying the zipped file it in $HOME/SAR/EXEC and run the script /$HOME/SAR/AMSTer/SCRIPTS_MT/zz_Utilities_MT_Ndo/***UpdateAMSTerEngine.sh***
with 2 parameters: the path to the zipped sources file and the date of the version.

3.2 Compile MSBAS:

- In Terminal, in $HOME/SAR/AMSTer/MSBAS/3Dsbas_Archives/msbasv4
  decompress msbas_20201009_wExtract_Unified_20220919_Optimized_v1.2_Gilles.zip
- In
  $HOME/SAR/AMSTer/MSBAS/3Dsbas_Archives/msbasv4/msbas_20201009_wExtract_Unified_20220919_Optimized_v1.2_Gilles
  type: *make all*
  (it seems that you can ignore warnings; most recent compilers are more sensitive)
- Move *msbasv4* in: $HOME/SAR/AMSTer/MSBAS
- In
  $HOME/SAR/AMSTer/MSBAS/3Dsbas_Archives/msbasv4/msbas_20201009_wExtract_Unified_20220919_Optimized_v1.2_Gilles/msbas_extract
  type: *make all*
- Move *msbas_extract* in: $HOME/SAR/AMSTer/MSBAS
- Clean (delete) the decompressed directories; keep the sources in
  $HOME/SAR/AMSTer/EXEC/Sources_Installed

- Update $PATH in your .bashrc. It must be something like
  PATH=$PATH:$HOME/SAR/AMSTer/MSBAS

Note:
The version msbas_20201009_wExtract_Unified_20220919_Optimized_v1.3_Full3D.zip is dedicated to very specific usage, that is when enough diversity in viewing geometries is achieved to perform full 3D inversion. Usually, this can only be achieved when data are acquired along Ascending and Descending orbits by sensors with right and left looking capabilities. See AMSTer manual.

3.3 Get the AMSTer Toolbox scripts at the right place:

- Ensure that the scripts (obtained from Github or from unzipped file) are stored at the right place:
  $HOME/SAR/AMSTer/SCRIPTS_MT

- Update $PATH in your .bashrc. It must be something like
  PATH=$PATH:$HOME/SAR/AMSTer/SCRIPTS_MT
  PATH=$PATH:$HOME/SAR/AMSTer/SCRIPTS_MT/zz_Utilities_MT
  PATH=$PATH:$HOME/SAR/AMSTer/SCRIPTS_MT/zz_Utilities_MT_NdO
  PATH=$PATH:$HOME/SAR/AMSTer/SCRIPTS_MT/_cron_scripts
  PATH=$PATH:/home/YourAccount/SAR/AMSTer/SCRIPTS_MT/optimtoolbox
  PATH=$PATH:/home/YourAccount/SAR/AMSTer/SCRIPTS_MT/diagtoolbox
  PATH=$PATH:$HOME/SAR/AMSTer/SCRIPTS_MT/AMSTerOrganizer

- If scripts were receiver from a zipped file, it can be stored in
  $HOME/SAR/AMSTer/EXEC/Sources_Installed

3.4 Define the environmental variables:

AMSTerEngine, MSBAS and AMSTer Toolbox requires environmental variables that will be called from the scripts. These are :

PATH_SCRIPTS=/${HOME}/SAR/AMSTer : path to where is located the directory SCRIPTS_MT (i.e. the directory containing all the scripts)

PATHGNU=/opt/local/bin : path to where the gnu utilities are stored (better to have it in the $PATH as well)

PATHFIJI=/Applications/Fiji.app/Contents/MacOS/: path to where Fiji application is

PATHCONV=/opt/local/bin : path to where the convert command is (from ImageMagick)

PATHTOCPXFIDDLE=/${HOME}/SAR/EXEC/ : path to where the cpxfiddle command is (from TU Delft)

The path to some disks where data, intermediate results and final results will be stored must also be defined by state variable. At least 5 variables must be defined, that is:
PATH_1650, PATH_3600, PATH_3601, PATH_3602 and PATH_DataSAR.

Names are inherited from architecture at ECGS and will be called throughout the processing.
For security reason, it is a good practice to have them pointing to different physical drives, if possible, but this is not mandatory; they can also point toward directories instead of disks, e.g.:

PATH_1650=/Volumes/hp-1650-Data_Share1 (disk or dir where data or results will be stored)
PATH_3600=/Volumes /hp-D3600-Data_Share1 (disk or dir where data or results will be stored)
PATH_3601=/Volumes /hp-D3601-Data_RAID6 (disk or dir where data or results will be stored)
PATH_3602=/Volumes /hp-D3602-Data_RAID5 (disk or dir where data or results will be stored)
PATH_DataSAR=/Volumes /DataSAR/ (disk or dir where auxiliary data will be automatically stored such as orbits, geoid etc… in a dir named SAR_AUX_FILES)

Note also that more disks can be defined and used e.g. for sharing the computation through several disks when intensive computation is required.

When processing ENVISAT or Sentinel 1 data, AMSTerEngine will require orbital information. It will usually pick them for you and store them automatically in dedicated directories where it will get them when required. These directories must be defined by the following state variable names:

S1_ORBITS_DIR=${PATH_DataSAR}/SAR_AUX_FILES/ORBITS/S1_ORB
ENVISAT_PRECISES_ORBITS_DIR=${PATH_DataSAR}/SAR_AUX_FILES/ORBITS/ENV_ORB/vor_gdr_d

To create DEM corrected from geoid height, the following state variable must be defined:
EARTH_GRAVITATIONAL_MODELS_DIR=$PATH_DataSAR/SAR_AUX_FILES/EGM
A directory named EGM96 must exist in that directory and it must contain the geoid file WW15MGH.DAC. It can be downloaded from (link checked 05 10 2022):
https://web.archive.org/web/20130314064801/http://earth-info.nga.mil/GandG/wgs84/gravitymod/egm96/binary/WW15MGH.DAC

To create DEM, EXTERNAL_DEMS_DIR state variable need to be defined in your **.bashrc** (.e.g.: EXTERNAL_DEMS_DIR=*/YourPath/DEMS/*).

When you create your DEM (e.g. using **getSRTMDEM** function since AMSTerEngine V20241010– see manual), it is strongly advised to move (and rename if needed) the created DEM to a directory from where you will call it in the ParametersFile.txt files. By default, the DEM will have the same name than the given kml.

You can also create automatic water body mask using **getSRTMDEM** function since AMSTerEngine V20241010 (see manual). For that, you will need to have a state variable EXTERNAL_MASKS_DIR.

If an EXTERNAL_MASKS_DIR environmental variable is defined in your **.bashrc** (.e.g.: EXTERNAL_MASKS_DIR=*/YourPath/MASKS/*), the waterBody mask will be saved in that EXTERNAL_MASKS_DIR. In that case, it is also strongly advised to move (and rename if needed) the created mask to a directory from where you will call it in the ParametersFile.txt files.

4) Directory structure for data, auxiliary data and results

In addition to the structure of the installation directories, it is important to store your data, auxiliary data (DEM, masks, kml, orbits, geoid, parameters files…), intermediate results and final results in a structure as described below.

AMSTer expects a directory structure similar as what is shown below. Most of the directories where the intermediate and final results are stored will be created by the scripts based on the information provided in the LaunchMTparameters.txt file.

For those which must be created by the user (in full or based on name provided in the LaunchMT-parameters.txt file), it is required to follow the following architecture and directories naming.
In *italic green* is the information that must be adapted to your satellite(s), orbital mode(s) and region(s). These directories can be located in the disk(s) you want.
Note that, for safety reasons, it is advised to store e.g. raw data and data in csl format (i.e. from AMSTerEngine) on different hard drives, or the mass processed results and the msbas results on different hard drives etc….
Names in red are mandatory.

Disk 1 (i.e. PATH_1650):

PATH_1650/kml : where to store kml for cropping, or for coherence check
PATH_1650/ Param_files : where to store your LaunchMTparameters.txt file by */SAT/TRK*….
PATH_1650/SAR_CSL/*SAT/TRK*/NoCrop : where data in csl format are stored (incl. crops)
PATH_1650/SAR_SM/AMPLITUDES/*SAT/TRK/Region* : where amplitudes gif will be computed
PATH_1650/SAR_SM/MSBAS/*Region* : where compatible pairs for msbas are computed
PATH_1650/SAR_SM/RESAMPLED/*SAT/TRK* : where data resampled on Global Primary are saved

Disk 2 (i.e. PATH_3600):

PATH_3600/SAR_DATA/*SAT/…* : where raw data are stored (safer to keep them on a different hard drive than 1650)

Disk 3 (i.e. PATH_3601):

PATH_3601/SAR_MASSPROCESS/*SAT/TRK/Crop* : where directories with each pair computation details and results are stored
PATH_3601/SAR_MASSPROCESS/*SAT/TRK/Crop*/Geocoded : where geocoded maps are stored in sub dir Amp, Coh, Defo etc…
PATH_3601/SAR_MASSPROCESS/*SAT/TRK/Crop*/GeocodedRasters : where rasters of geocoded maps (for quick look) are stored

Disk 4 (i.e. PATH_3602):

PATH_3602/MSBAS_RESULTS/*Region* : where msbas series are computed

Maybe in one of these disks, or somewhere else where enough storage room is available:

$PATH_DataSAR/SAR_AUX_FILES/DEM : where DEMs are stored
$PATH_DataSAR/SAR_AUX_FILES/EGM96 : where geoid is stored
$PATH_DataSAR/SAR_AUX_FILES/MASKS/*SAT/TRK* : where coherence masks are computed
$PATH_DataSAR/SAR_AUX_FILES/ORBITS/*SAT* : where orbits are stored

Note that $PATH_DataSAR is the state variable defined in .bashrc !!

5) Check installation

5.1 The directory structure

At the end, you should have AMSTer installed in a folder named “SAR” as shown in **Fig. 1**.

```
/$HOME/SAR/EXEC
                |--Executables…
                |--Sources_Installed

/$HOME/SAR/AMSTer
            |--DOC
            |     |----Biblio
            |     |----AMSTerEngine
            |     |----AMSTer_Toolbox
            |     |----MSBAS
            |
            |--AMSTerEngine
            |     |--_Sources_AE/Older
            |     |                  |--[several sources of old versions]
            |     |                  |--V20241108_AMSTerEngine
            |     |--Executables…
            |
            |--MSBAS
            |     |----3Dsbas_Archives
            |     |              |--msbasv4
            |     |--Executables…
            |
            |-- SCRIPTS_MT
                  |-- _cron_scripts
                  |-- AMSTerOrganizer
                  |-- diagtoolbox
                  |-- optimtoolbox
                  |-- TemplatesForPlots
                  |-- TSCombiFiles
                  |-- zz_Utilities_MT
                  |-- zz_Utilities_MT_Ndo
                  A lot of scripts…
```

Figure 1 – Directory structure required for the proper usage of AMSTer.

Notes:

1) The folders /$HOME/SAR/AMSTer/AMSTerEngine/_Sources_AE/Older is used to store the archives and the most recent sources of both components of AMSTerEngine, that is InSAR and MSBASTools.

Their latest executable version, when compiled, must be moved to /$HOME/SAR/AMSTer/AMSTerEngine/.

Hence, when a new version of these packages is available, no need to change the paths of the environmental variables.

2) Similarly, the folders /$HOME/SAR/AMSTer/MSBAS/_3Dsbas_Archives is used to store the archives and the most recent sources of both components of MSBAS, that is MSBAS_EXTRACT and MSBASv*i* (where *i* stands for the version). Their latest executable version, when compiled, must be moved to /$HOME/SAR/AMSTer/MSBAS/.

Hence, when a new version of these packages is available, no need to change the paths of the environmental variables.

3) You can use the script ***Check_Installation.sh*** to check your installation. See script and manual.

You should also have auxiliary files stored in a folder named "SAR_AUX_FILES" in DataSAR as shown in **Fig. 2**.

```
$PATH_DataSAR/SAR_AUX_FILES/
                              |--DEM
                              |
                              |--EGM96
                              |     |----WW15MGH.DAC
                              |
                              |--MASKS
                              |
                              |--ORBITS
                                   |----ENV_ORB
                                   |----S1_ORB
```

Figure 2 – Directory structure required for the storage of auxiliary files required by AMSTer Toolbox.

Notes:

1) The folders /$PATH_DataSAR/SAR_AUX_FILES/DEM/ is used to store the DEMs. That is where you store and sort manually your DEMs according to your needs.

   It is advised to use another path than the one defined in the state variable $EXTERNAL_DEMS_DIR. This last one is used e.g. to create automatically the DEM using the AMSTerEngine command **getSRTMDEM** (see manual). However, for the automation with AMSTer Toolbox, to avoid possible confusion, we recommend to move the DEM created in $EXTERNAL_DEMS_DIR to /$PATH_DataSAR/SAR_AUX_FILES/DEM/ and indicate that last full path in the LaunchMTparameters.txt file for each of your run. If you move the DEM, remember to change the path indicated in the *DEM*.txt file accordingly to the new place where the DEM is stored.

   The only exception when you may need to have DEMs in $EXTERNAL_DEMS_DIR is if you intend to use AMSTerEngine without the AMSTer Toolbox scripts (e.g. using LazInSAR command). Indeed, in that case, unless you manually change the configuration text files, it will expect the DEM in that directory.

2) The folders /$PATH_DataSAR/SAR_AUX_FILES/EGM must contain the geoid file in EGM96/WW15MGH.DAC.

3) The folders /$PATH_DataSAR/SAR_AUX_FILES/MASKS is not mandatory, but it is a convenient place to store the mask you would build and used during your processing.

   It is advised to use another path than the one defined in the state variable $EXTERNAL_MASKS_DIR. This last one is used e.g. to create automatically the water body mask when creating the DEM using the AMSTerEngine command **getSRTMDEM** (see manual). However, for the automation with AMSTer Toolbox, to avoid possible confusion, we recommend to move the masks created in $EXTERNAL_MASKS_DIR to

/$PATH_DataSAR/SAR_AUX_FILES/MASKS/ and indicate that last full path in the LaunchMTparameters.txt file for each of your run. If you move the mask, remember to change the path indicated in the *Mask*WB.txt file accordingly to the new place where the mask is stored.

4) The folders /$PATH_DataSAR/SAR_AUX_FILES/ORBITS must contain the orbits for ENVISAT (if required) or Sentinel 1 (if required). These directories must be associated to the state variable $ENVISAT_PRECISES_ORBITS_DIR and $S1_ORBITS_DIR respectively.

5) To automatically create DEM using the AMSTerEngine command **getSRTMDEM** (since AMSTer Engine from October 2024), you need to have the following lines in your .nertc (see paragraph 7 below to create a netrc):

machine e4ftl01.cr.usgs.gov login *yourlogin* password *yourpwd*

5.2 The environmental variables in .bashrc

At the end, your ./bashrc must look like:

```
#
# .bashrc
#

[…]

# User specific environment and startup programs
PATH=/opt/local/bin:$PATH
PATH="$PATH:/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin"

BASH_ENV=$HOME/.bashrc
USERNAME=""
LANG=C

# export EDITOR=/usr/bin/edit
export EDITOR=/usr/bin/nano
export PAGER=/usr/bin/less

export USERNAME LANG BASH_ENV PATH

export JAVA_HOME=/Library/Internet\ Plug-Ins/JavaAppletPlugin.plugin/Contents/Home


# AMSTer PATHS
###################

PATH=$PATH:/Users/YourAccount/SAR/EXEC
PATH=$PATH:/Users/YourAccount/SAR/AMSTer/MSBAS
PATH=$PATH:/Users/YourAccount/SAR/AMSTer/AMSTerEngine
PATH=$PATH:/Users/YourAccount/SAR/AMSTer/SCRIPTS_MT
PATH=$PATH:/Users/YourAccount/SAR/AMSTer/SCRIPTS_MT/zz_Utilities_MT_Ndo
PATH=$PATH:/Users/YourAccount/SAR/AMSTer/SCRIPTS_MT/zz_Utilities_MT
PATH=$PATH:/Users/YourAccount/SAR/AMSTer/SCRIPTS_MT/_cron_scripts
PATH=$PATH:/Users/YourAccount/SAR/AMSTer/SCRIPTS_MT/optimtoolbox
PATH=$PATH:/Users/YourAccount/SAR/AMSTer/SCRIPTS_MT/diagtoolbox
PATH=$PATH:/Users/YourAccount/SAR/AMSTer/SCRIPTS_MT/AMSTerOrganizer

# AMSTer VARIABLES
###################

export PATH_HOMEDATA=/Users/YourAccount/
export PATH_3602=/Volumes/hp-D3602-Data_RAID5/
export PATH_3601=/Volumes/hp-D3601-Data_RAID6/
export PATH_3600=/Volumes/hp-D3600-Data_Share1/
export PATH_1650=/Volumes/hp-1650-Data_Share1/
export PATH_DataSAR=/Volumes/DataSAR/

export EARTH_GRAVITATIONAL_MODELS_DIR=${PATH_DataSAR}/SAR_AUX_FILES/EGM
export ENVISAT_PRECISES_ORBITS_DIR=${PATH_DataSAR}/SAR_AUX_FILES/ORBITS/ENV_ORB
export S1_ORBITS_DIR=${PATH_DataSAR}/SAR_AUX_FILES/ORBITS/S1_ORB

export EXTERNAL_DEMS_DIR=/APath/WhereToCreate/DEMs
export EXTERNAL_MASKS_DIR=/APath/WhereToCreate/MASKs

export PATH_SCRIPTS=/Users/YourAccount/SAR/AMSTer
export PATHCONV=/opt/local/bin
export PATHFIJI=/Applications/Fiji.app/Contents/MacOS/
export PATHGNU=/opt/local/bin
export PATHTOCPXFIDDLE=/Users/YourAccount/SAR/EXEC/
export PATH=$PATH

#export OMP_NUM_THREADS=10,8,4
#export OMP_NUM_THREADS=4,3,2
```

6) Operating with cron jobs

If you want to operate **AMSTer** from cron jobs, pay attention to the following:

- It is advised to launch scripts by redirecting the messages to /dev/null to avoid filling up the mailbox by ending the command line with

  > /dev/null 2>&1

- If bash is not your default shell (as for the recent Mac OS X) and you want to run scripts from cronjobs, call the scripts in your crontab by calling bash as follow:

  hh mm * * * /bin/bash *YourScript.sh* > /dev/null 2>&1

Example of crontab:

```
# Always end command line launched by cron with > /dev/null 2>&1 to mute output and avoid filling up the mailbox. It will also prevent wrong count of running processes if any...
## TO GET THE SCRIPTS RUNNING USING CMD LINES AND SCRIPTS STORED IN PATH AVAILABLE IN BASHRC RATHER THAN IN COMMON
## /user/local/bin FOR INSTANCE, DO NOT FORGET TO SOURCE THE BASHRC AT THE BEGINNING OF THE SCRIPT !!
# MM HH DoMonth Month DoWeek(0=sunday)

## ********************************************************************************************************************************
## ****************************************** CONGO VVP ****************************************************************************
## ********************************************************************************************************************************

## Step 1: S1 & CSK
45 04 * * * /bin/bash /Users/doris/SAR/SCRIPTS_MT/_cron_scripts/VVP_S1_Step1_Read_SMCoreg_Pairs.sh > /dev/null 2>&1
30 12 * * * /bin/bash /Users/doris/SAR/SCRIPTS_MT/_cron_scripts/VVP_CSK_Step1_Read_SMCoreg_Pairs.sh > /dev/null 2>&1

## Step 2: S1 & CSK
31 06 * * * /bin/bash /Users/doris/SAR/SCRIPTS_MT/_cron_scripts/VVP_S1_Step2_Read_MassProc.sh > /dev/null 2>&1
31 07 * * * /bin/bash /Users/doris/SAR/SCRIPTS_MT/_cron_scripts/VVP_CSK_Step2_MassProc.sh > /dev/null 2>&1

## Step 3: S1
02 21 * * * /bin/bash /Users/doris/SAR/SCRIPTS_MT/_cron_scripts/VVP_S1_Step3_MSBAS.sh > /dev/null 2>&1
02 02 * * * /bin/bash /Users/doris/SAR/SCRIPTS_MT/_cron_scripts/VVP_CSK_Step3_MSBAS.sh > /dev/null 2>&1

## Website Update (once a week)
00 06 * * 0 /bin/bash /Users/doris/SAR/SCRIPTS_MT/_cron_scripts/rsync_satellite_images_to_virunga_volcanoes.sh > /dev/null 2>&1
```

7) Downloading the Sentinel-1 orbits

Processing Sentinel-1 data requires orbits files, either the Near Real Time predicted (PRE) orbits available 180 minutes after the acquisition, or the Non Time Critical Precises (POE) orbits available typically 20 days after the acquisition.

These can be downloaded automatically using the AMSTer Engine function ***updateS1Orbits*** as a transparent step in the processing, either from ESA or from ASF (Alaska SAR Facility).

In both cases, it requires to have at least AMSTerEngine V20231213 and a login and a password to be obtained from ESA or ASF, which must be stored in a file named .netrc in your home directory. That file must contain one line for either ESA and/or ASF login as follow:

```
machine identity.dataspace.copernicus.eu login <YourLogin> password <YourPassword>
machine urs.earthdata.nasa.gov login <YourLogin> password <YourPassword>
```

The first line is dedicated to the access the ESA server https://dataspace.copernicus.eu (this dataspace server replaces the https://scihub.copernicus.eu server that was closed in November 2023), while the second line is dedicated to the access the ASF server.

Of course, `<YourLogin>` and `<YourPassword>` must be replaced (without the `< >` symbols) with the login and password obtained from the provider:

- For ESA, visit https://dataspace.copernicus.eu and proceed to the registration by following the instructions provided on the web site;
- For ASF, visit urs.earthdata.nasa.gov and proceed to the registration by following the instructions provided on the web site. When registered and logged in, you must also:
  - click on “My Profile”, then menu “Applications > Authorized Apps”.
  - At the bottom, click on button “APPROVE MORE APPLICATIONS”
  - In the “Search” bar, type “Sentinel”
  - Click on “AUTHORIZE” button next to each application named with Sentinel 1.

  Then in a terminal on your computer type:

  ```
  > cd ~
  > touch .netrc
  > echo "machine urs.earthdata.nasa.gov login <YourLogin> password <YourPassword>" > .netrc
  > chmod 0600 .netrc
  ```

  Then

  ```
  > cd ~
  > touch .urs_cookies
  ```

  For more information, see: https://wiki.earthdata.nasa.gov/display/EL/How+To+Access+Data+With+cURL+And+Wget.

With such a .netrc file, providing that your destination directory *S1_ORBITS_DIR* is set up and defined as state variable in your *.bashrc*, you can download the orbits from ESA or ASF by launching either ***updateS1Orbits*** or ***updateS1Orbits*** *-ASF* respectively. This is usually done from the scripts and you do not have to worry about it.

8) Updates in this manual

V 4.1 (October 26 2022):
- add installation of gnu parallel
- add chapter “7) Updates in this manual”
- correct typo in Version number in header

V 4.3 (April 20 2023):
- Refer to last versions of software and scripts
- Check against installer script

V 5.0 (Aug 30 2023):
- Major changes in dirs and files naming
- Take into account new AMSTer Engine V20230828 which is mandatory

V 6.0 (Oct 25 2023):
- Major changes related to the renaming of MasTer as AMSTer
- Information about public GitHub access to source code and licensing
- Typos and varia

V 6.0.1 (Nov 14 2023):
- Update Macport jdk19 as jdk20

V 6.1 (Dec 15 2023):
- Data server change at ESA in Nov. 2023 requires new .netrc file for allowing the download of Sentinel-1 orbits.

V 6.1.1 (Jan 03 2024):
- Update msbas zip source code V1.2 to avoid compilation error with most recent Mac OSX version

V 6.2.1 (Oct 10 2024):
- Update some python instructions
- Add DEMS and MASKS state variables to cope with new function getSRTMDEM from AMSTerEngine from October 2024
- Add info about .netrc for SRTM download

V 6.2.2 (Nov 13 2024):
- Information about msbas_20201009_wExtract_Unified_20220919_Optimized_v1.3_Full3D.zip
- Cosmetic

V 6.2.3 (Aug 07 2025):
- New diagnostic toolbox (by Delphine Smittarello)
- Python packages scikit-gstat and geopandas
- Info about python 3 system-wide installation
- Clang version 20 is mandatory from AMSTerEngine_2025085