An open-source toolbox for automated multi-sensor InSAR mass processing and multidimensional deformation time series

Quentin Glaude, Nicolas d'Oreye, Dominique Derauw, Sergey Samsonov, Delphine Smittarello, Maxime Jaspard, Gilles Celli

European Center for Geodynamics and Seismology (ECGS) | National Museum of Natural History, Luxembourg, Centre Spatial de Liege, University of Liege | Natural Resources Canada

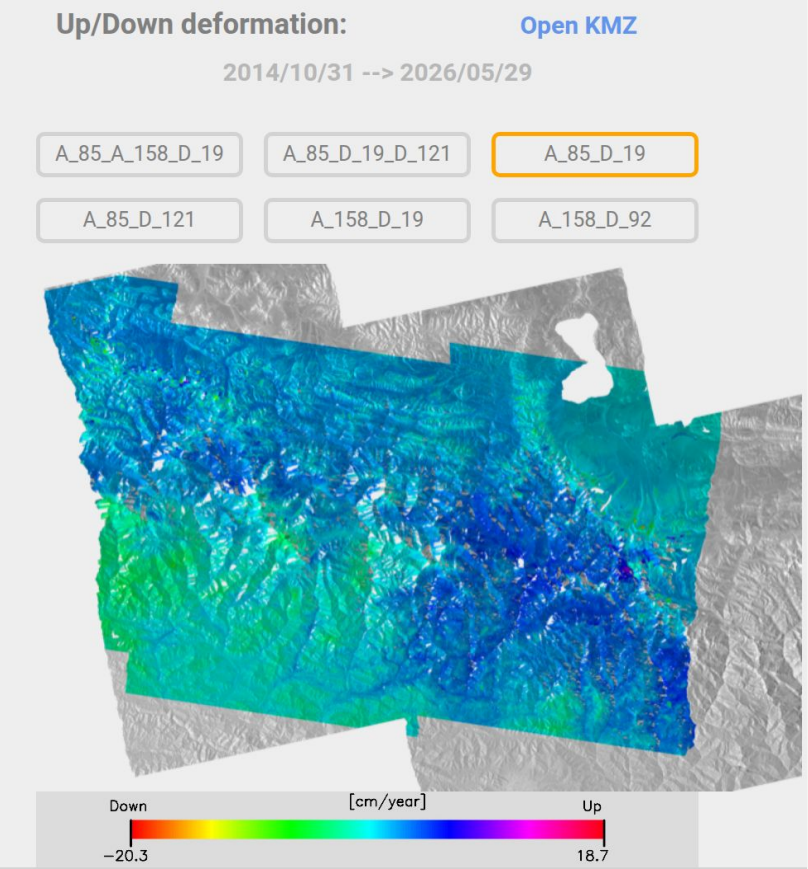
FRINGE 2026 | Processing environments and operational services | 16 June 2026

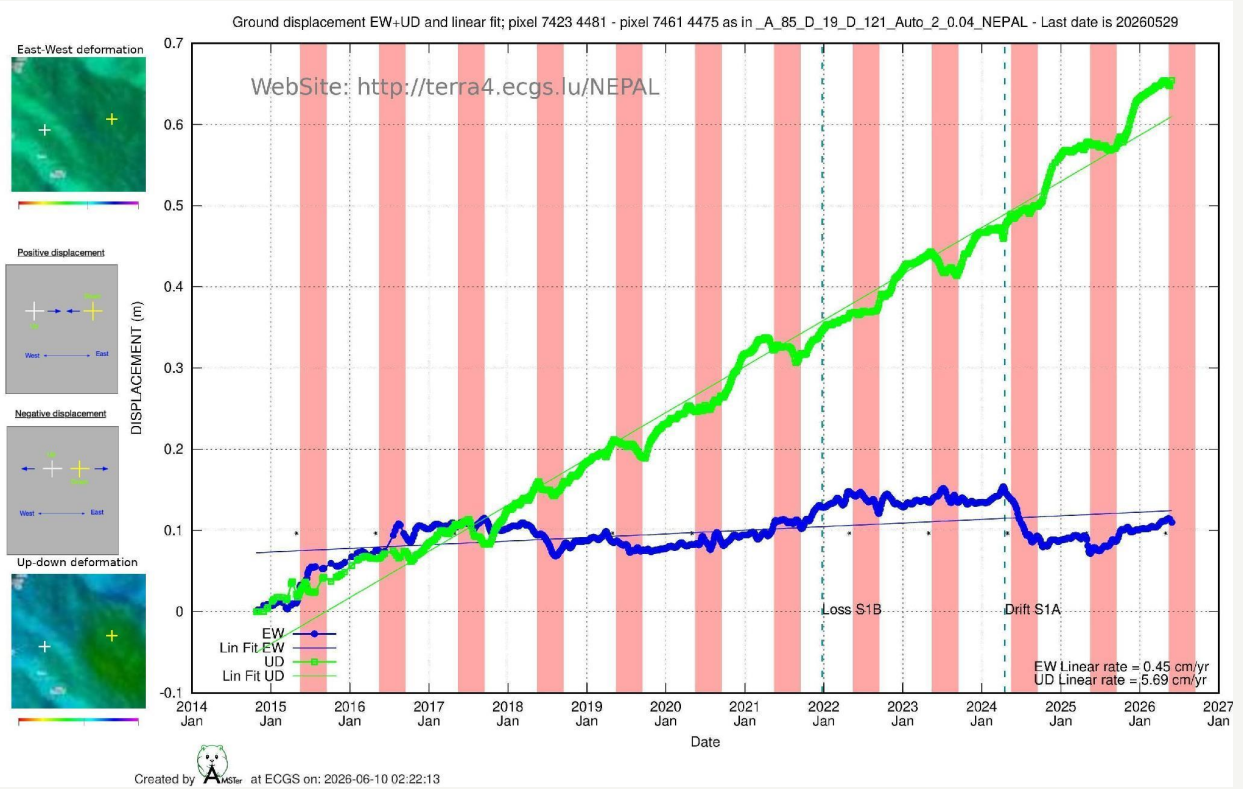
github.com/AMSTerUsers/AMSTer_Distribution

AMSTer User Start of day

Arrival at work, discovering new results computed during the night: deformation map, updated time series, QA reports, geocoded results

➢ https://terra4.ecgs.lu





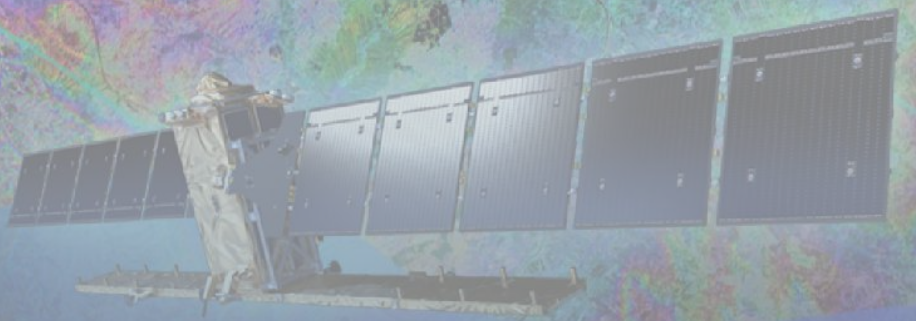
Central Nepal (Himalaya) processed and updated automatically

Ministry of Science and Higher Education
Republic of Poland

JAGIELLONIAN UNIVERSITY
IN KRAKÓW

esa
EUROPEAN SPACE AGENCY

I. Why AMSTer?
II. AMSTer Toolbox Processing Chain
III. Operational services & processing environments
IV. Science studies & validation
V. Resources, community & outlook



github.com/AMSTerUsers/AMSTer_Distribution

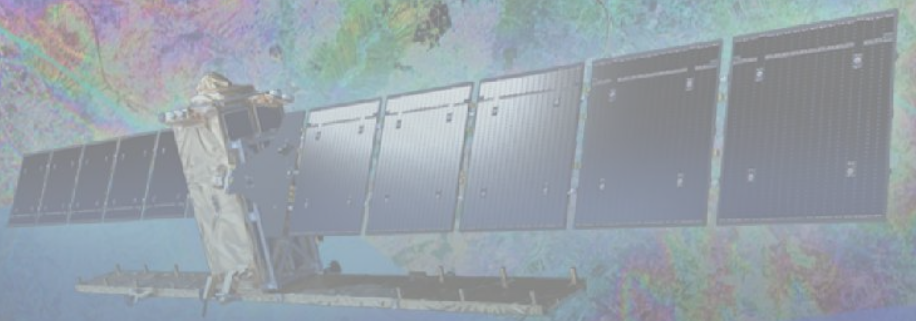
Ministry of Science and Higher Education
Republic of Poland

JAGIELLONIAN UNIVERSITY
IN KRAKÓW

esa
EUROPEAN SPACE AGENCY

I. Why AMSTer?

II. AMSTer Toolbox Processing Chain
III. Operational services & processing environments
IV. Science studies & validation
V. Resources, community & outlook



github.com/AMSTerUsers/AMSTer_Distribution

AMSTer at a glance

Ministry of Science and Higher Education
Republic of Poland

JAGIELLONIAN UNIVERSITY IN KRAKÓW

esa EUROPEAN SPACE AGENCY

AMSTer Engine

C-language core

Coregistration, InSAR generation, filtering, unwrapping, geocoding

MSBAS

Multi-dim inversion

2D (EW+UD) or 3D (EW+UD+NS)
Multi-sensor, multi-geometry

AMSTer Toolbox

Bash + Python wrapper

Mass processing orchestration, cron automation, QC, dissemination

25+
years of development

12+
SAR satellite families

12+
operational targets

AGPL v3
open source license

InSAR processing ecosystem

Tool / Service	Scope & strength	AMSTer's complementary role
DORIS	*Legacy TU Delft processor; v5 still beta*	Maintained open-source multi-sensor chain
GAMMA	Mature commercial suite	Open-source alternative (AGPL v3)
ISCE + MintPy	Flexible research stack	Integrated operational chain incl. dissemination
GMTSAR	GMT ecosystem, free	MSBAS 2D/3D inversion + wide sensor support
FLATSIM / NSBAS	National HPC service (Sentinel-1 only)	Multi-sensor (12+ families), 2D/3D deformation
SNAP	Desktop/GUI reference processor; best educational tool	Automated on-prem mass processing; broader sensor coverage; Lighter memory requirements
openEO / CDSE	Cloud-native, burst-centric	On-prem control; broader sensor coverage
StaMPS / GAMMA-IPTA	Persistent Scatterer InSAR (PSI)	Complementary lane: AMSTer targets DS / SBAS-MSBAS
HyP3 / EGMS	Turnkey products ("turn the key" and it runs)	Full control, custom targets, multi-sensor fusion

AMSTer

- Multi-sensor A-to-Z chain, 12+ sensor families + MSBAS 2D/3D, on-prem, operational, open-source (AGPL v3)
- Built on 25 yr of CSL/ECGS interferometric heritage (CIS → MasTer → AMSTer)
- **Independent implementations let results be cross-checked: science is healthier when not everyone runs the same chain**

Who uses AMSTer?

European Center for Geodynamics and Seismology (ECGS)

Core toolbox development & operational monitoring
Walferdange, Luxembourg

Centre Spatial de Liege (CSL) University of Liege

AMSTer Engine core development
Liège, Belgium

FORM@TER / Pole Terre Solide

French Solid Earth data & services
On-demand InSAR processing (alongside FLATSIM)

Royal Museum for Central Africa

East African Rift geohazard monitoring
Tervuren, Belgium

Geohazards TEP (GEP) (integration under way)

ESA Thematic Exploitation Platform (operated by Terradue)
On-demand InSAR for geohazards

Researchers worldwide

PhD programs, visiting scientists, collaborations

From core development to operational monitoring and the wider research community

12+ SAR satellite families supported

Ministry of Science and Higher Education
Republic of Poland

Jagiellonian University in Kraków

esa EUROPEAN SPACE AGENCY

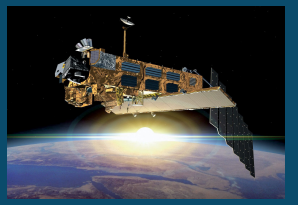
New mission? If the specs and some test data exist, we build a reader .

ERS-1/2

since 1991

ENVISAT

since 2002

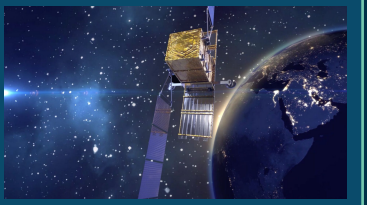


ALOS-1/2/4

since 2006

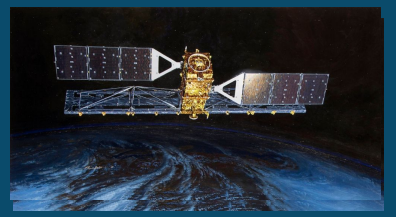
Cosmo-SkyMed

since 2007



Radarsat-1/2/RCM

since 1995



TerraSAR-X / TDX / PAZ

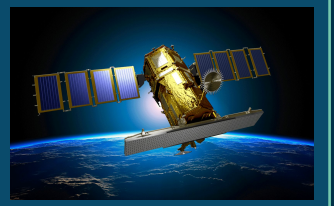
since 2007

Sentinel-1 A/B/C/D

since 2014

Kompsat-5

since 2013



SAOCOM-1A/B

since 2018

ICEYE

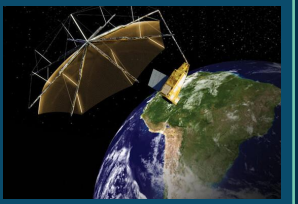
since 2018

NISAR

since 2025

BIOMASS (integration phase)

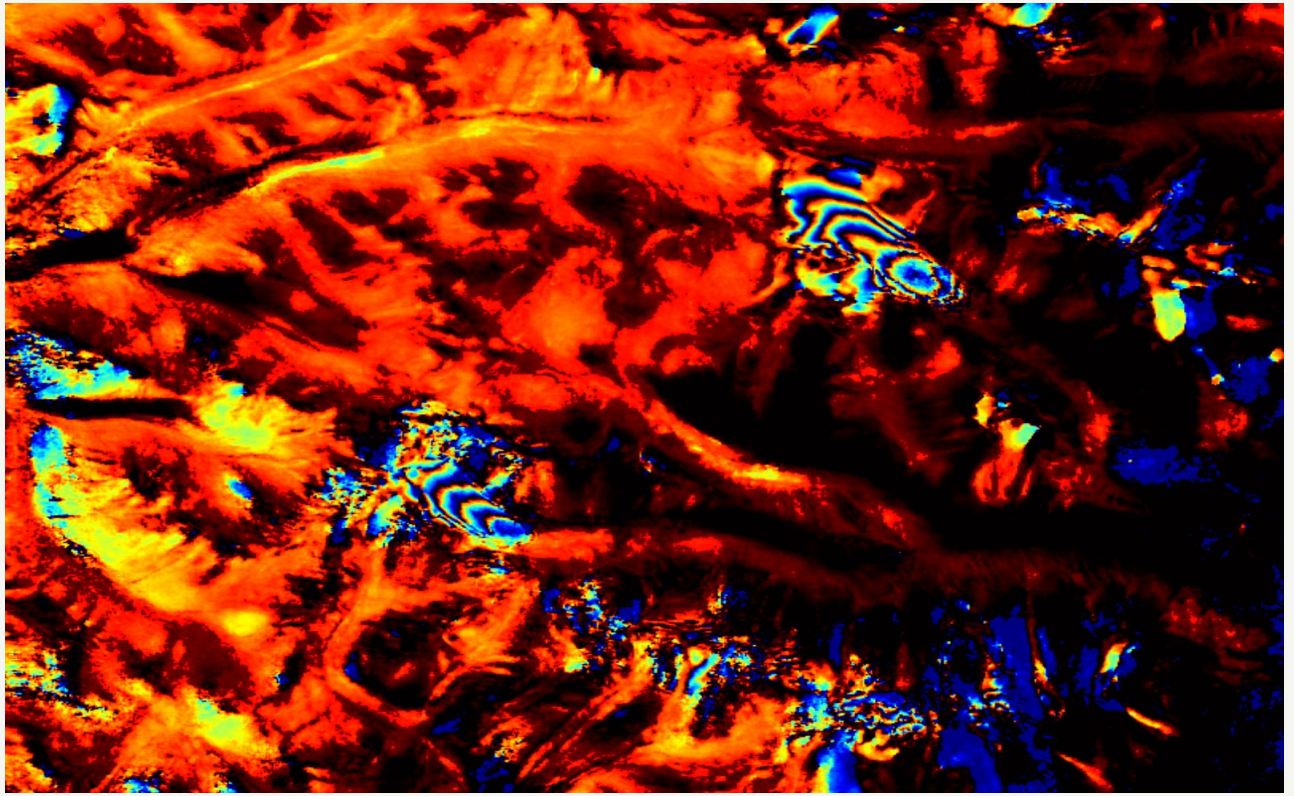
since 2025



Independent in-house data readers: no external dependency (ISCE, SNAP, …). Full control over new-sensor integration.

NISAR joining the family

New `NISARDataReader`, tested on the Feb 2026 NISAR L1 RSLC delivery (acquisitions Nov 2025 - Jan 2026) over Central Nepal, confirming the AMSTer Toolbox capability to process time series alongside other satellites



First Multidimensional Small Baseline Subset Time Series from NISAR L-band InSAR: Processing Workflow Demonstration Using the AMSTer Toolbox

Quentin Glaude[1], Delphine Smittarello[1], Nicolas d'Oreye[1], and Dominique Derauw[2]

[1]European Center for Geodynamics and Seismology (ECGS), Walferdange, Luxembourg
[2]Centre Spatial de Liège, Université de Liège, Belgium

June 8, 2026

Posted on June 8, 2026. The copyright holder is the author/funder. All rights reserved. This is a preprint and has not been peer reviewed. Results may be preliminary.

Abstract

The NASA-ISRO Synthetic Aperture Radar (NISAR) mission, launched in July 2025, joins ALOS-2/4 and SAOCOM in the operational L-band SAR fleet and is the first L-band mission to combine a systematic, globally consistent acquisition plan with a free and open data policy on a regular short-repeat cycle. We demonstrate the first application of Multidimensional Small Baseline Subset processing to NISAR L1 Radar Single Look Complex data acquired over Central Nepal between November 25, 2025, and January 15, 2026. Processing 9 acquisitions distributed across ascending track 98 and descending track 48 enabled generation of line-of-sight deformation time series for both viewing geometries, which were decomposed into vertical and east-west displacement components through MSBAS inversion. First results show that, L-band coherence remained high across the challenging Himalayan terrain. Glacier deformation and other ground motion patterns were clearly observable in the resulting products, with glacier displacement rates exceeding 3 m/yr and slow-moving landslide motion detectable at the centimetre-per-year level. This work establishes a complete processing workflow from NISAR RSLC ingestion through MSBAS decomposition, validating the methodology for NISAR's global systematic coverage mission using the AMSTer toolbox.

1

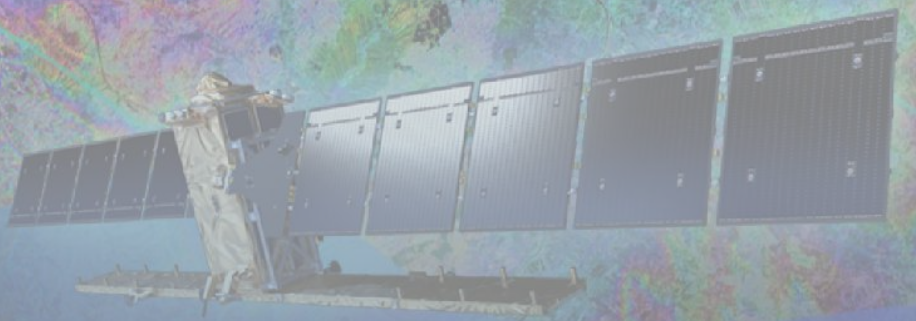
Glaude, Smittarello, d'Oreye & Derauw (2026)
First Multidimensional Small Baseline Subset Time Series from NISAR L-band InSAR: Processing Workflow Demonstration Using the AMSTer Toolbox. ESSOAr preprint, June 2026

Ministry of Science and Higher Education
Republic of Poland

JAGIELLONIAN UNIVERSITY
IN KRAKÓW

esa
EUROPEAN SPACE AGENCY

I. Why AMSTer?
II. AMSTer Toolbox Processing Chain
III. Operational services & processing environments
IV. Science studies & validation
V. Resources, community & outlook



github.com/AMSTerUsers/AMSTer_Distribution

AMSTer: a command-line toolbox

No GUI. Every step is a shell command: high-level Bash scripts orchestrate a compiled C engine

AMSTer Toolbox (Bash orchestration layer) Examples of the commands you usually run

`Read_All_Img.sh` `SuperMasterCoreg.sh` `SuperMaster_MassProc.sh` `Prepa_MSBAS.sh` `MSBAS.sh` `PlotTS_all_comp.sh` `SinglePair.sh`

each bash script runs a fixed sequence of compiled C programs

AMSTerEngine compiled C core (called by the orchestration layer)

`S1DataReader` `S1Coregistration` `InSARProductsGeneration` `InterferogramFiltering` `Geocoding` `…`

Same toolbox, from a quick test to a full multi-year time series:

```
$ SinglePair.sh Primary Secondary LaunchMTparam.txt        # one interferogram, to tune parameters
$ Read_All_Img.sh …
$ SuperMasterCoreg.sh …
$ MSBAS.sh …         # a full time series
```

Written in C for performance; Bash and Python wrappers handle orchestration, parameters and plotting.

Single pair: tune before you scale

One command runs the full chain on a single interferogram, to set parameters before a mass process

```
$ SinglePair.sh Primary Secondary LaunchMTparam.txt

# whole chain, one pair, end to end
```

What you tune in LaunchMTparam.txt

- Multilooking factors (Zoom, ML) and crop
- Adaptive spatial filtering strength
- Unwrapping method: SNAPHU / branch-cut / DetPhun
- Coherence threshold and masking
- Geocoding and output projection (UTM)

Iterate on one interferogram, then reuse the same parameter file for the whole stack

The engine chain `SinglePair.sh` executes

1. **`S1DataReader`** (if Sentinel-1)
   Read SLC into CSL, apply precise orbits
2. **`S1Coregistration`** (if Sentinel-1)
   Align secondary to primary at sub-pixel
3. **`InSARProductsGeneration`**
   Interferogram, coherence, amplitude
4. **`InterferogramFiltering`**
   Adaptive (Goldstein) spatial filtering
5. **Phase unwrapping**
   SNAPHU / branch-cut / DetPhun
6. **`Geocoding`**
   Project to ground geometry (UTM, ENVI format)

From one pair to a full stack

Example: deformation time-serie

➢ six scripts take you from raw archives to ground displacement.

1 `Read_All_Img.sh`
Ingest all SLCs into CSL format

2 `SuperMasterCoreg.sh`
Coregister every image on the Global Primary

3 `Prepa_MSBAS.sh`
Baseline plot, pick Global Primary and pairs

4 `SuperMaster_MassProc.sh`
Generate all compatible interferograms

5 `MSBAS.sh`
Invert the stack into displacement

6 `PlotTS_all_comp.sh`
Time series and rate maps

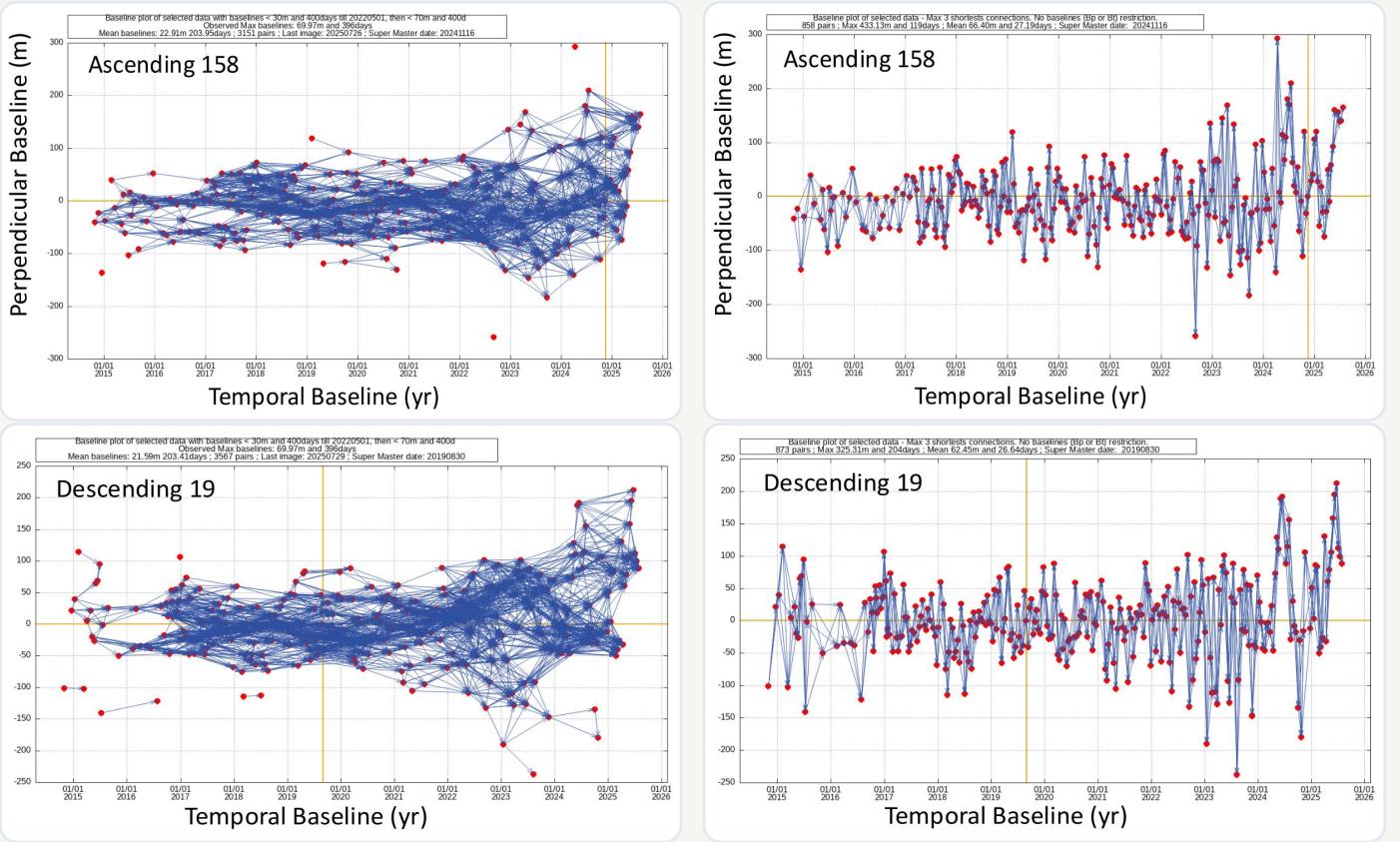
```
$ Read_All_Img.sh          <SAR_DATA_dir> <SAR_CSL_dir>/NoCrop <SAT> <kml> <POL> <RESAMPLED_dir> <MASSPROCESS_dir>
$ SuperMasterCoreg.sh      LaunchMTparam.txt
$ Prepa_MSBAS.sh           <set_dir> <Bp> <Bt> <GlobalPrimary>
$ SuperMaster_MassProc.sh  <pairs_table.txt>  LaunchMTparam.txt
$ MSBAS.sh
$ PlotTS_all_comp.sh       <run_label> <X1> <Y1> <X2> <Y2> [-f -r -t -g] [-events=<file>]

# the commands, run by hand or from a cron job
```

Designing the InSAR network

Choosing the connection graph (the insar pairs to be processed) controls time-series quality and job size

- Links set by perpendicular and temporal baseline limits (Bp, Bt)
- Date-dependent thresholds adapt to gaps (e.g. Sentinel-1B loss, Dec 2021)
- "x shortest connections": keep the N most robust links per image
- Coherence-proxy selection for seasonally decorrelating terrain
- Built by `Prepa_MSBAS.sh` and `Extract_x_Shortest_Connections.sh`

Ref: Smittarello et al. (2022), **J. Geophys. Res. Solid Earth**, 127, e2021JB022825.



Central Nepal: full network (left) vs. 3 shortest connections (right); ascending 158 (top), descending 19 (bottom).

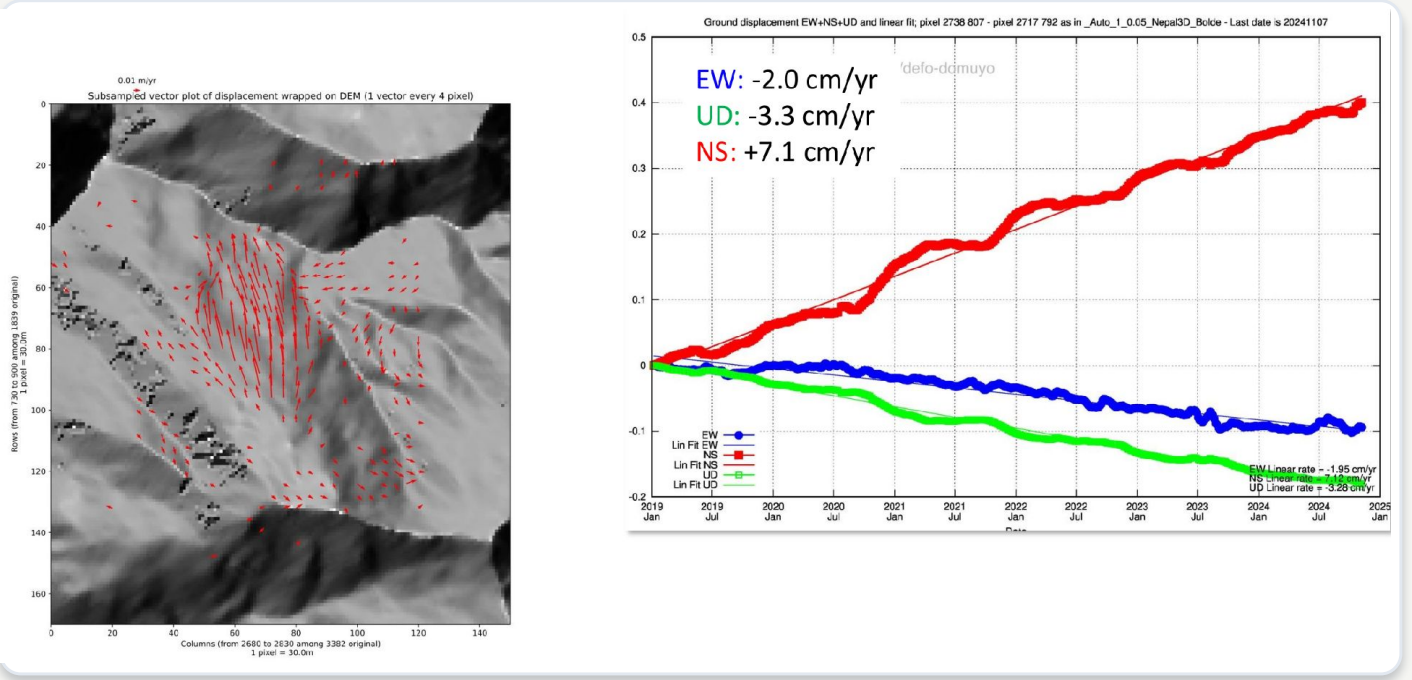
MSBAS: from SLC to 3D displacement

The end goal: combine every geometry and sensor into ground-displacement time series

MSBAS inverts the full stack of unwrapped, geocoded interferograms, across all viewing geometries and sensors, into displacement time series

- Input: unwrapped, geocoded interferograms
- 2D: East-West and Vertical (Up-Down) at every pixel and date
- 3D: adds North-South
    - steepest-slope assumption
    - geometry-diversity constraint

- L-curve regularisation; run by `MSBAS.sh`, plotted by `PlotTS_all_comp.sh`



3D displacement (EW + UD + NS) from MSBAS inversion of multi-geometry Sentinel-1 stacks: vector field and per-component time series.

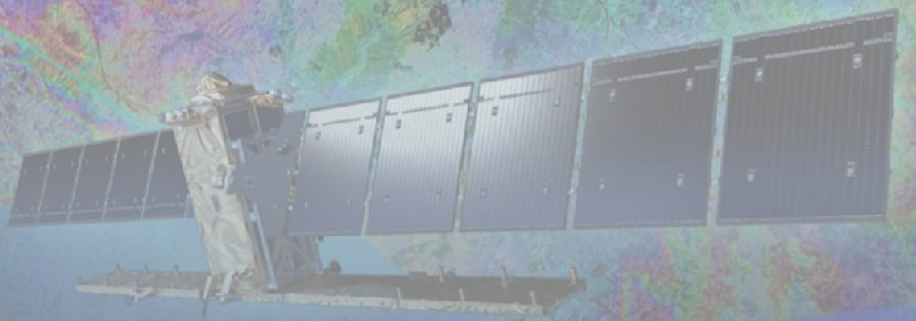
```
Level-1 SLC  →  coregistered InSAR stacks (N geometries)  →  MSBAS.sh  →  2D (EW+UD) / 3D (EW+UD+NS) time series
```

Ministry of Science and Higher Education
Republic of Poland

JAGIELLONIAN UNIVERSITY
IN KRAKÓW

esa
EUROPEAN SPACE AGENCY

I. Why AMSTer?
II. AMSTer Toolbox Processing Chain
III. Operational services & processing environments
IV. Science studies & validation
V. Resources, community & outlook



github.com/AMSTerUsers/AMSTer_Distribution

Operational automation at ECGS

Possible to schedule processing for a geographical target through cron jobs

Example over Central Nepal: 3 scheduled tasks running in sequence

1 **Step 1**
Read, coregister & select pairs

cron job
`NEPAL_S1_Step1_Read_SMCoreg_Pairs.sh`

`Read_All_Img.sh` →
`SuperMasterCoreg.sh` →
`Prepa_MSBAS.sh`

5 modes: A85, A158, D19, D92, D121

2 **Step 2**
Mass interferometric processing

cron job
`NEPAL_S1_Step2_MassProc.sh`

`SuperMaster_MassProc.sh`

Per mode; waits for Step 1; only if new images

3 **Step 3**
MSBAS inversion & dissemination

cron job
`NEPAL_S1_Step3_MSBAS.sh`

`MSBAS.sh` →
`PlotTS_all_comp.sh`

2D time series + automatic web plots

Example over Central Nepal: 5 Sentinel-1 geometries, updated automatically as new images arrive.

Per-target jobs
Each site has its own cron set: download, read + coreg, mass proc, inversion

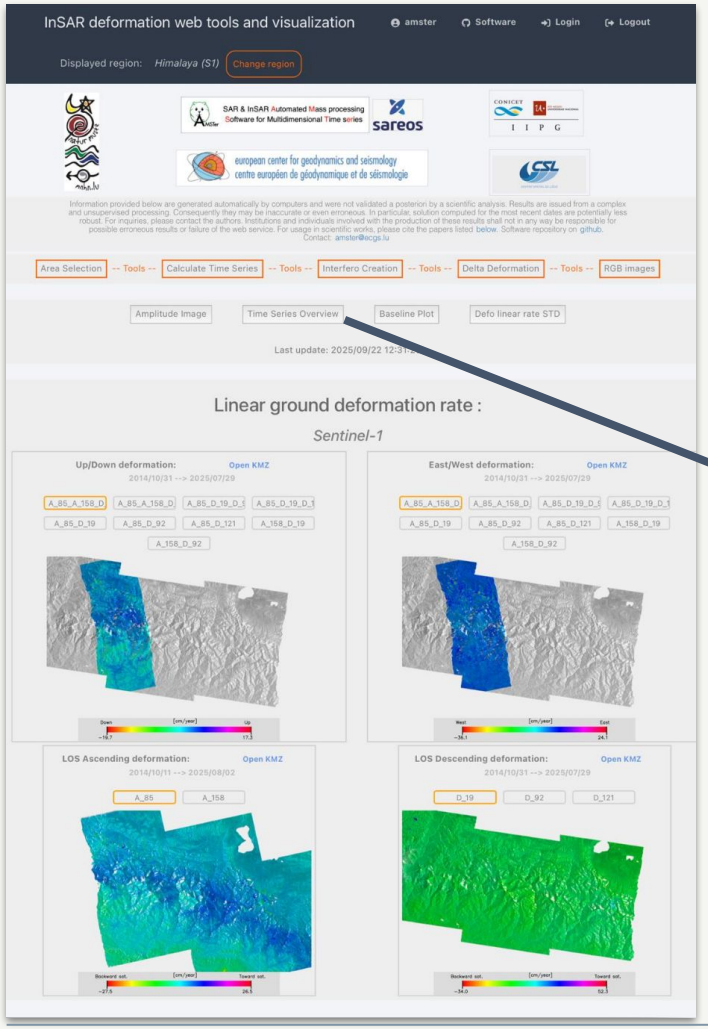
Incremental
Only new acquisitions trigger processing; no full recompute

Guarded
Each step checks the previous finished; skips when no new data

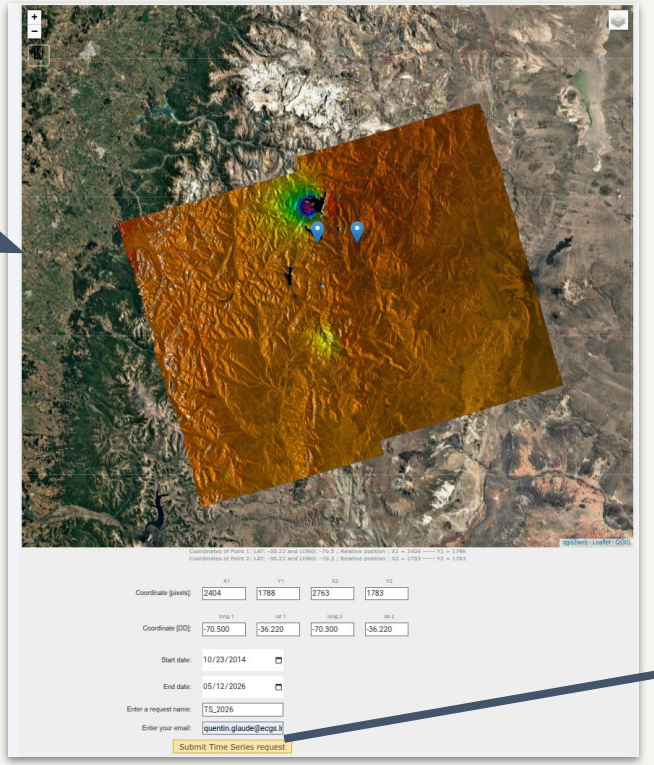
Unattended
Runs for years; results published automatically to the web

Automated web dissemination

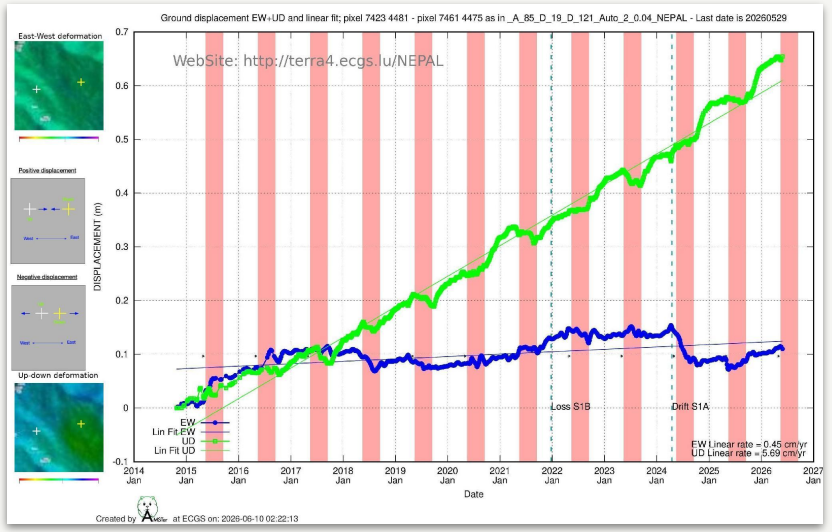
Self-updating, multi-region web portal that publishes AMSTer/MSBAS results automatically



➢ https://terra4.ecgs.lu



- Live & self-updating:
  - new image in → maps refreshed automatically
- Open + partner access:
  - public demo regions; login-protected pages
- Explore it yourself:
  - switch basemaps, read vertical / E-W / LOS velocity
- Click any point
  - → its time series; or two points → custom series by email in ~5 min
- Every layer exports to Google Earth / GIS (KMZ)



AMSTer Engine on HPC

Lightweight C and OpenMP parallelisation → Runs on any cluster with a standard EasyBuild toolchain

```
# On HPC (Rocky 9, EasyBuild 2023a)

module load foss/2023a
module load HDF5/1.14.0-gompi-2023a
module load netCDF/4.9.2-gompi-2023a
module load LibTIFF/4.5.0-GCCcore-12.3.0
module load libgeotiff/1.7.1-GCCcore-12.3.0
module load libxml2/2.11.4-GCCcore-12.3.0
module load FFTW/3.3.10-GCC-12.3.0
module load GSL/2.7-GCC-12.3.0

# AMSTerEngine Complilation
make USEOPENMP=YES
```

```
$ seff 699793
Job ID: 699793   Cluster: lyra   State: COMPLETED
CPU Utilized: 00:23:28    CPU Efficiency: 98.46%
Memory Utilized: 7.81 GB
Job Wall-clock time: 00:23:50
```

- Compiles & runs natively on HPC: standard EasyBuild modules, submitted as ordinary Slurm jobs (tested on CÉCI, Lyra cluster, Rocky 9)
- Lightweight engine: ~98% CPU efficiency, <8 GB RAM per job: runs from a laptop to a cluster node
- Embarrassingly parallel: one Slurm job per interferometric pair fans mass processing across nodes

GEP: platform interface and services

Ministry of Science and Higher Education
Republic of Poland

JAGIELLONIAN UNIVERSITY IN KRAKÓW

esa EUROPEAN SPACE AGENCY

geohazards tep by Terradue

On-demand processing

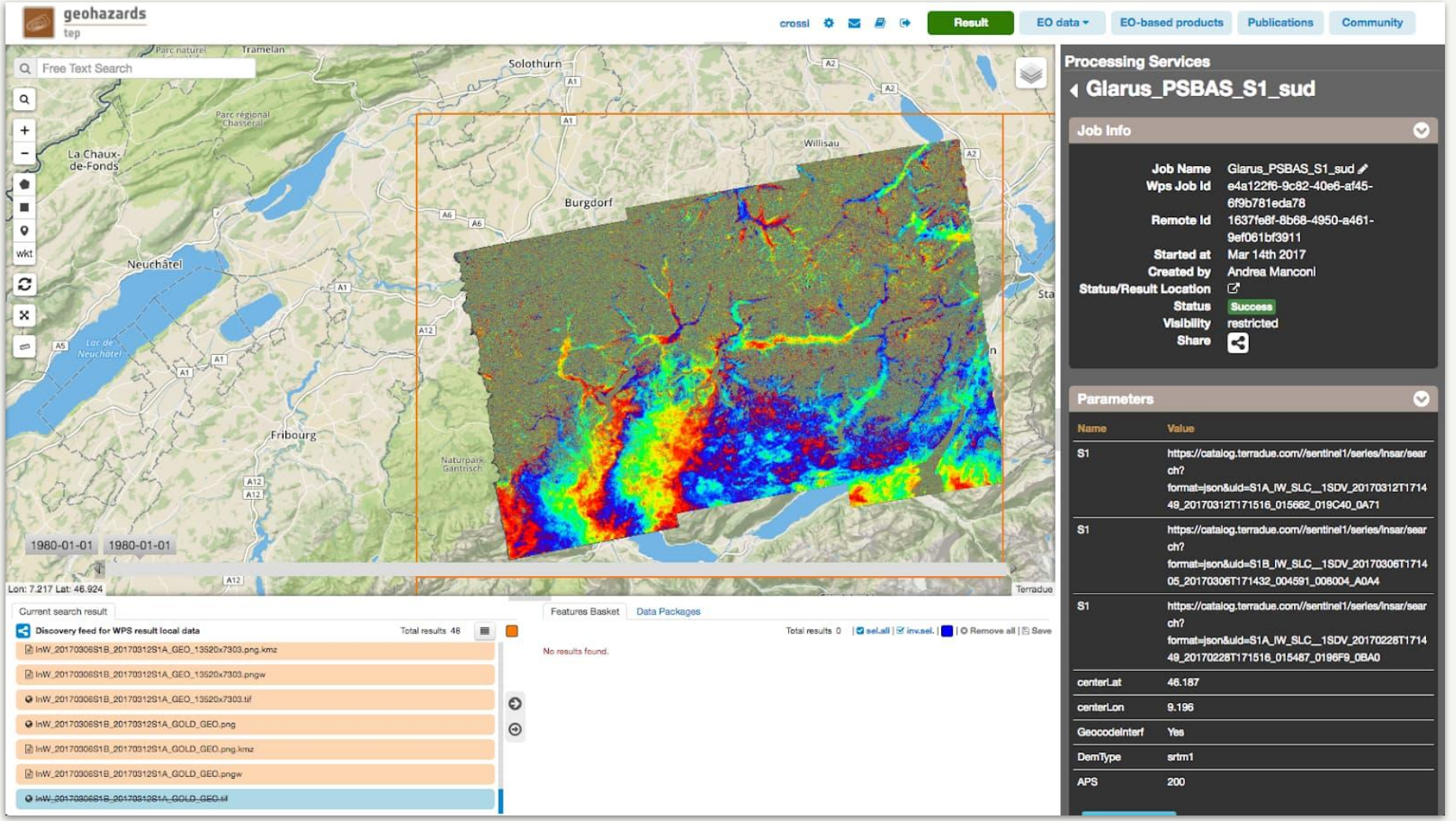
User-defined AOI and time range

Systematic monitoring

Automated scheduled runs

Data sharing

Results accessible to community

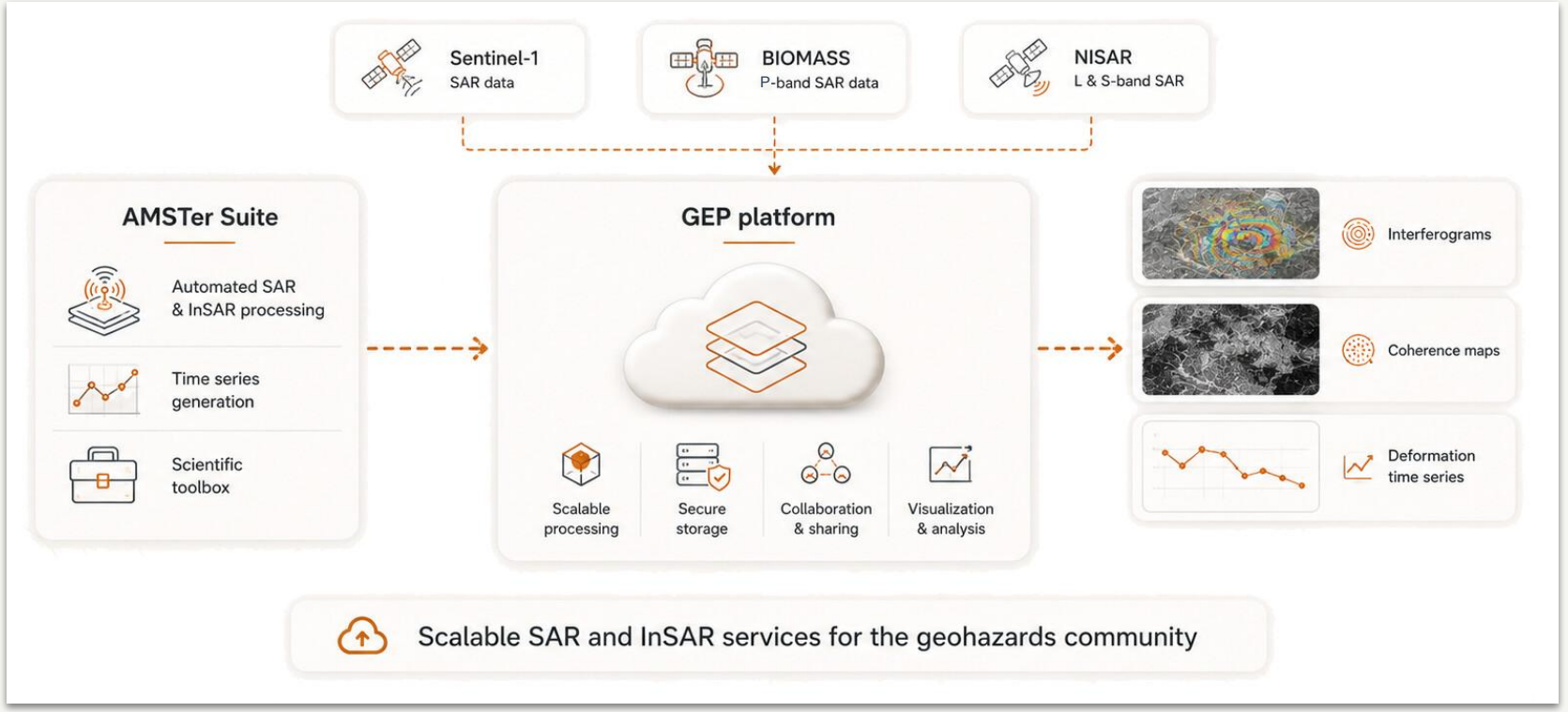


AMSTer on the GEP

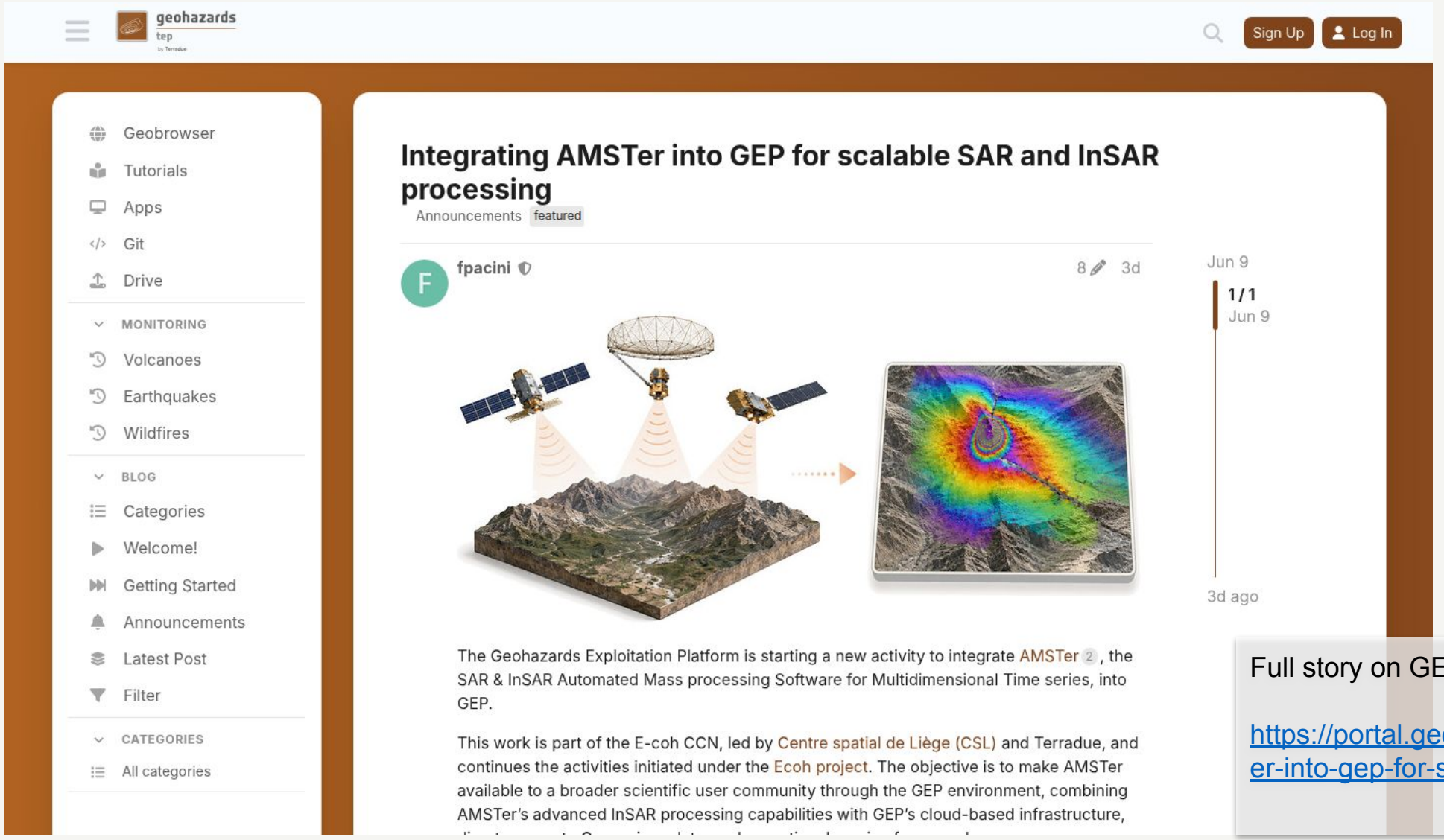
Making AMSTer available to the geohazards community through GEP's cloud

- Led by CSL and Terradue
- InSAR and DInSAR pair processing and coherence tracking, for sensors AMSTer already handles
- MSBAS time-series processing: multi-sensor deformation (East-West and vertical), geoprojected outputs
- Enabling workstreams: ETAD integration, plus BIOMASS and NISAR data readiness

Objective: Establish AMSTer as a persistent scientific suite within GEP



AMSTer on the GEP



Full story on GEP portal:

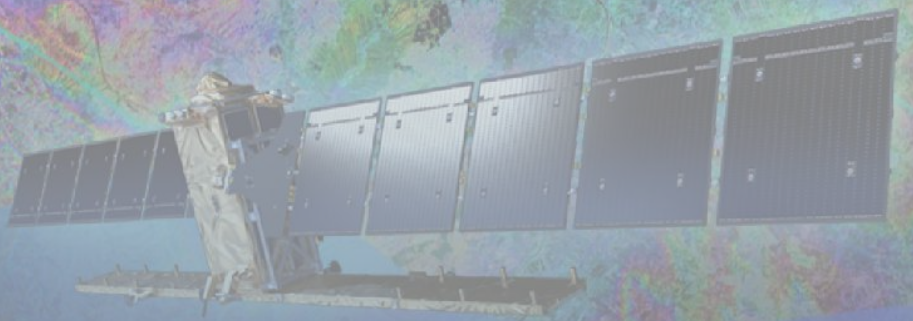
https://portal.geohazards-tep.eu/t/integrating-amster-into-gep-for-scalable-sar-and-insar-processing/

Ministry of Science and Higher Education
Republic of Poland

JAGIELLONIAN UNIVERSITY
IN KRAKÓW

EUROPEAN SPACE AGENCY

I. Why AMSTer?
II. AMSTer Toolbox Processing Chain
III. Operational services & processing environments
IV. Science studies & validation
V. Resources, community & outlook



github.com/AMSTerUsers/AMSTer_Distribution

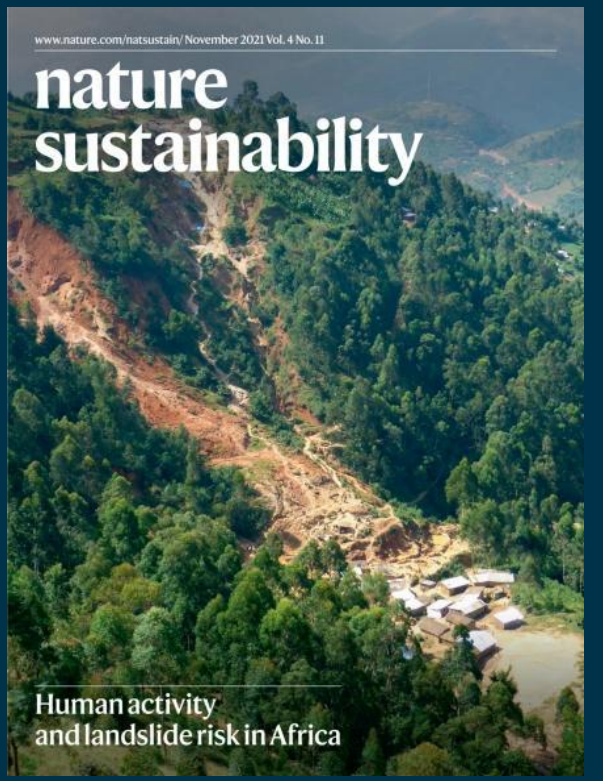
Monitoring & study sites worldwide

Domuyo • Laguna del Maule / Laguna Fea • Piton de la Fournaise • Virunga/Nyiragongo • Karthala • Himalaya
Guadeloupe • Galeras • Luxembourg • Brussels-Mons • South Kivu/Bukavu • Antarctica

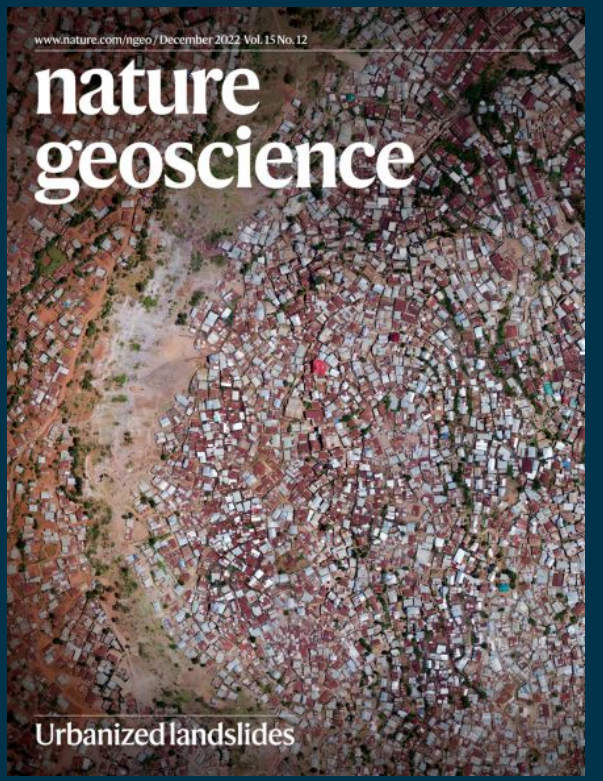
Behind some high-impact studies

The AMSTer toolbox was one of the tools that helped get there

D. Smittarello (2022)
Mount Nyiragongo, RDC



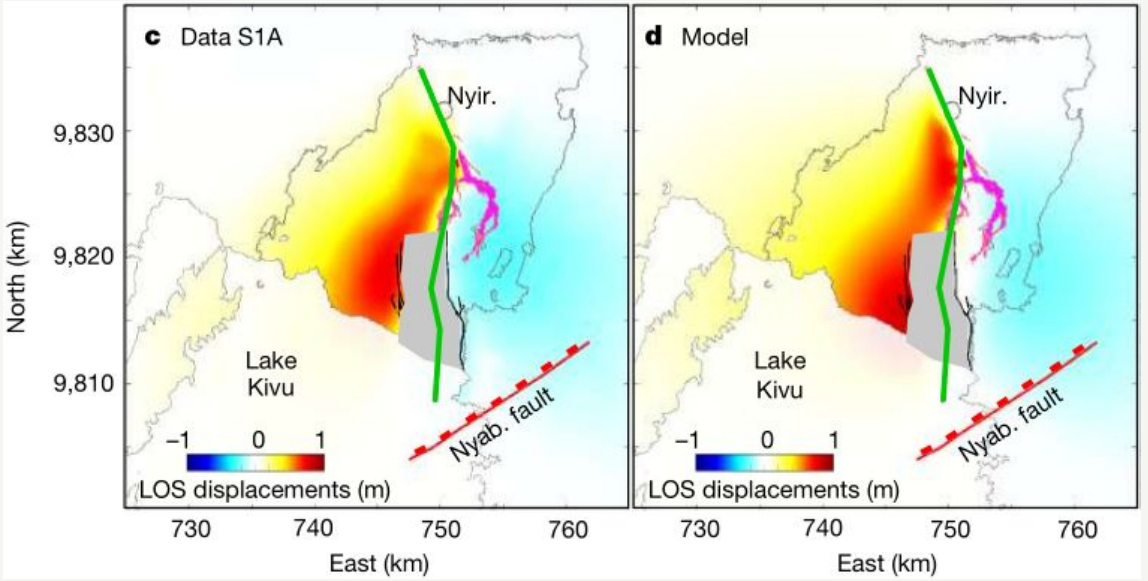
A. Depicker (2021)
Landslide risk in Africa, RDC



A. Dille (2022)
Urban landslides in Bukavu, RDC

Nyiragongo eruption, DRC 2021

Crisis response: tracking a shallow dyke beneath a city



InSAR vs Modelling of post eruptive ground deformation

Nyiragongo eruption summarized (May 2021, DR Congo)

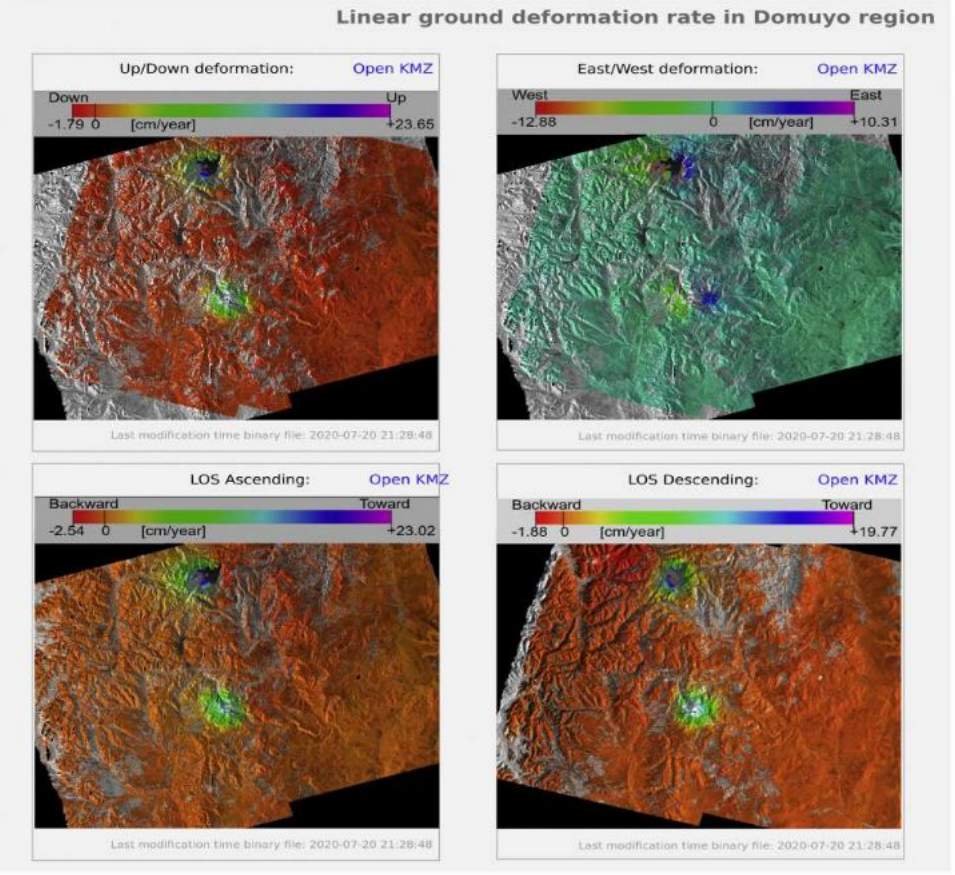
- Permanent lava lake, with no warning signs
- In May 2021 it erupted suddenly, with no precursor
- Unusually, the eruption came first, then magma spread sideways underground as a 25 km "dyke"
- Reversed sequence, eruption then a ~25 km dyke propagating south at <500 m depth, beneath Goma, Gisenyi and Lake Kivu.
- The magma travelled less than 500 m deep, directly under Goma.
- Satellite radar (S1, ALOS-2, CSK) measured the ground movement; a model of the dyke reproduces it

AMSTer Contribution: Co-eruptive Sentinel-1 and COSMO-SkyMed interferograms computed with MasTer (now AMSTer), automatically stitching 17/16 bursts across two frames to capture the full deformation footprint, plus the precursor-free pre-eruptive interferogram and multi-sensor crater-collapse tracking (8 satellites, 15 geometries).

Smittarello, Smets, Barrière et al. (2022).
Precursor-free eruption triggered by edifice rupture at Nyiragongo volcano. Nature 609, 83–88.

Domuyo & Laguna del Maule

Operational monitoring: unattended 2D deformation



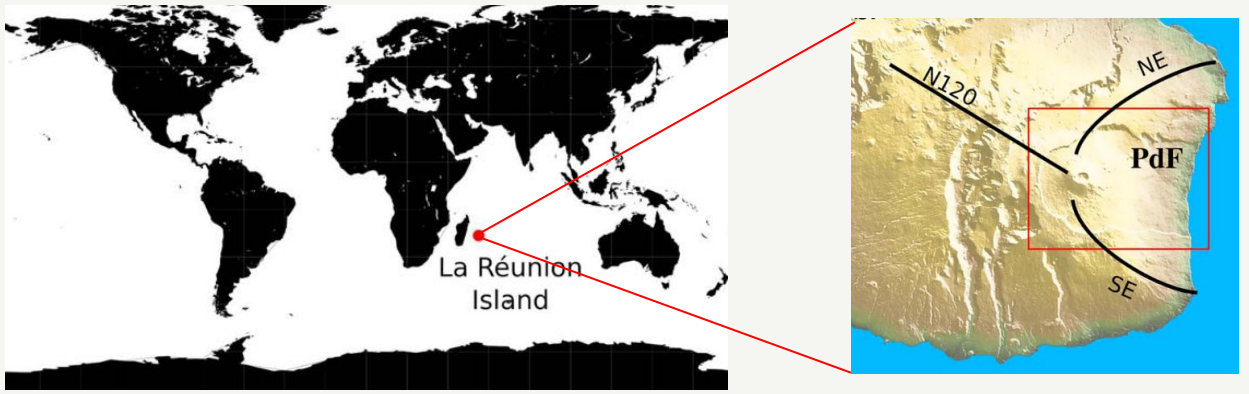
Domuyo & Laguna del Maule — Southern Andes

- Two slowly inflating volcanoes in the Southern Andes
- AMSTer monitors them automatically: it downloads each new image, processes it, and updates a web page every few days
- Combining two viewing angles gives both vertical and east-west ground motion
- Six years of data: Laguna del Maule rising up to 27 cm/year, Domuyo ~13 cm/year
- The long record shows the uplift speeds up and slows down over time, and revealed movement nobody had mapped before

AMSTer Contribution: This study is the demonstration of the MasTer (now AMSTer) automated processing chain — the operational, hands-off mode behind the monitoring sites shown earlier.

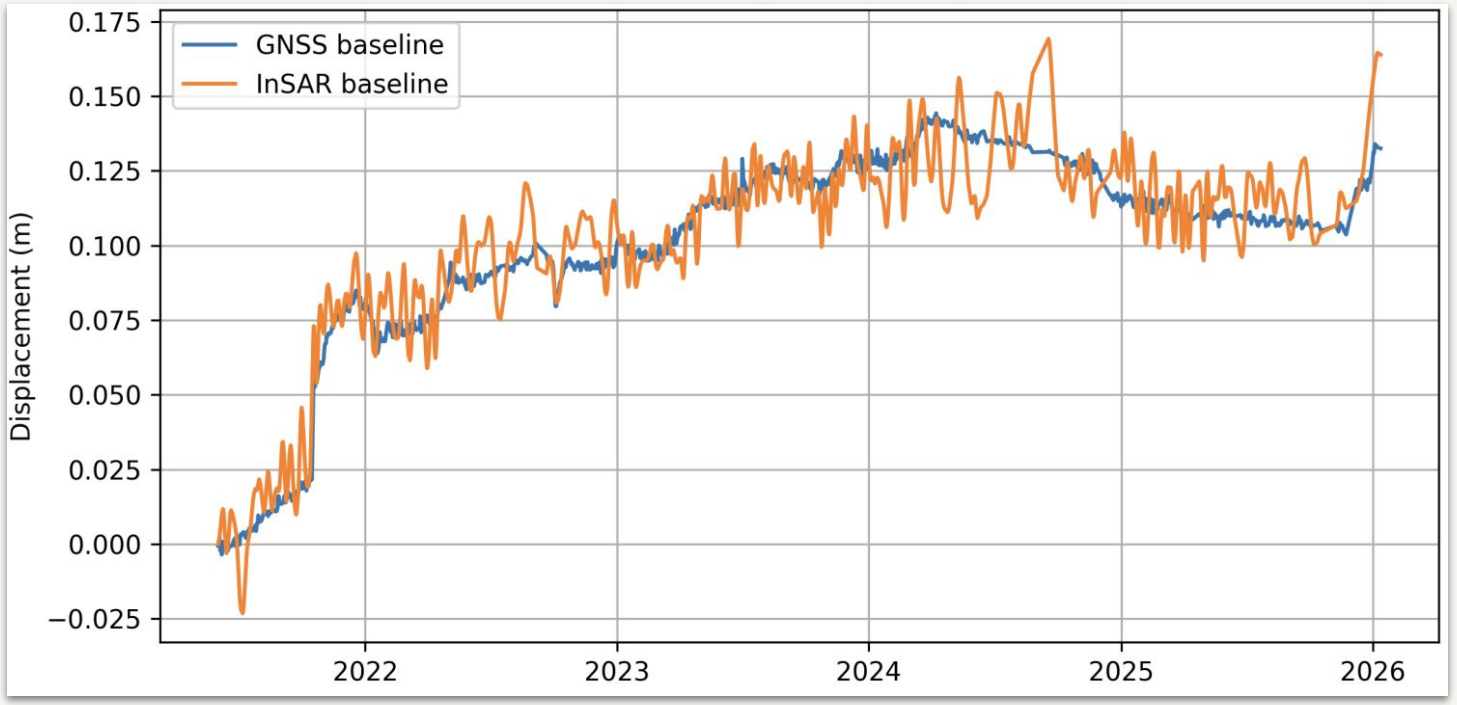
Derauw, d'Oreye, Jaspard, Caselli & Samsonov (2020).
Ongoing automated ground deformation monitoring of Domuyo–Laguna del Maule area (Argentina) using Sentinel-1 MSBAS time series. J. South Am. Earth Sci. 104, 102850.

AMSTer-derived movements vs GNSS

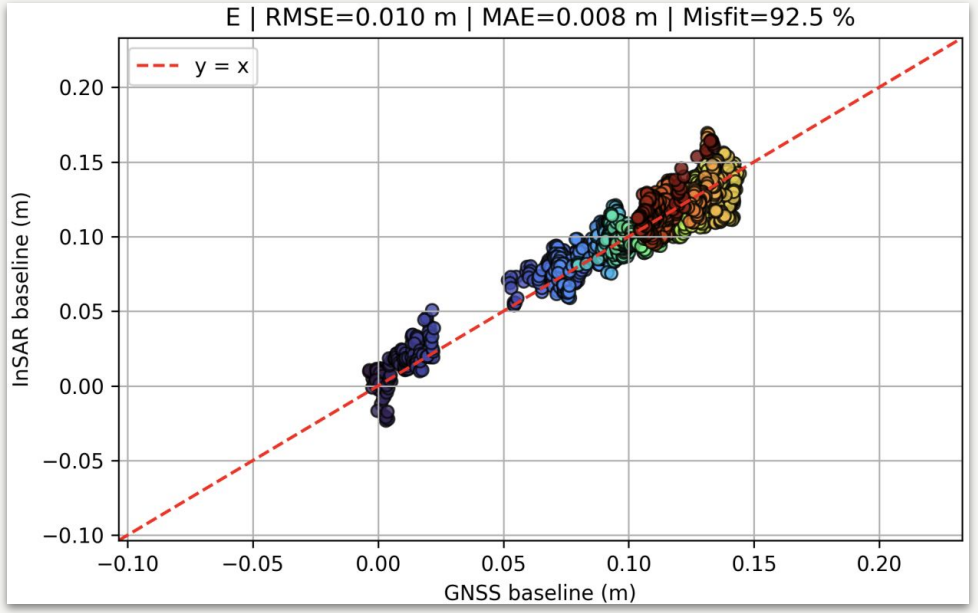


InSAR validation with GNSS: Piton de la Fournaise

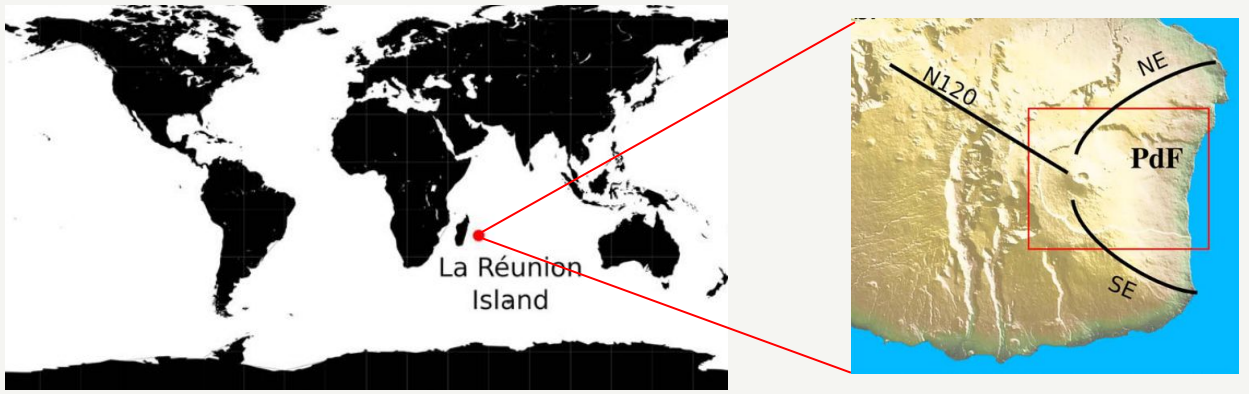
- Piton de la Fournaise is one of the most active volcanoes on Earth
- Dense GNSS network available
- Validation benchmark for InSAR-derived deformation:
  - InSAR vs GNSS time series
  - InSAR vs GNSS scatterplots

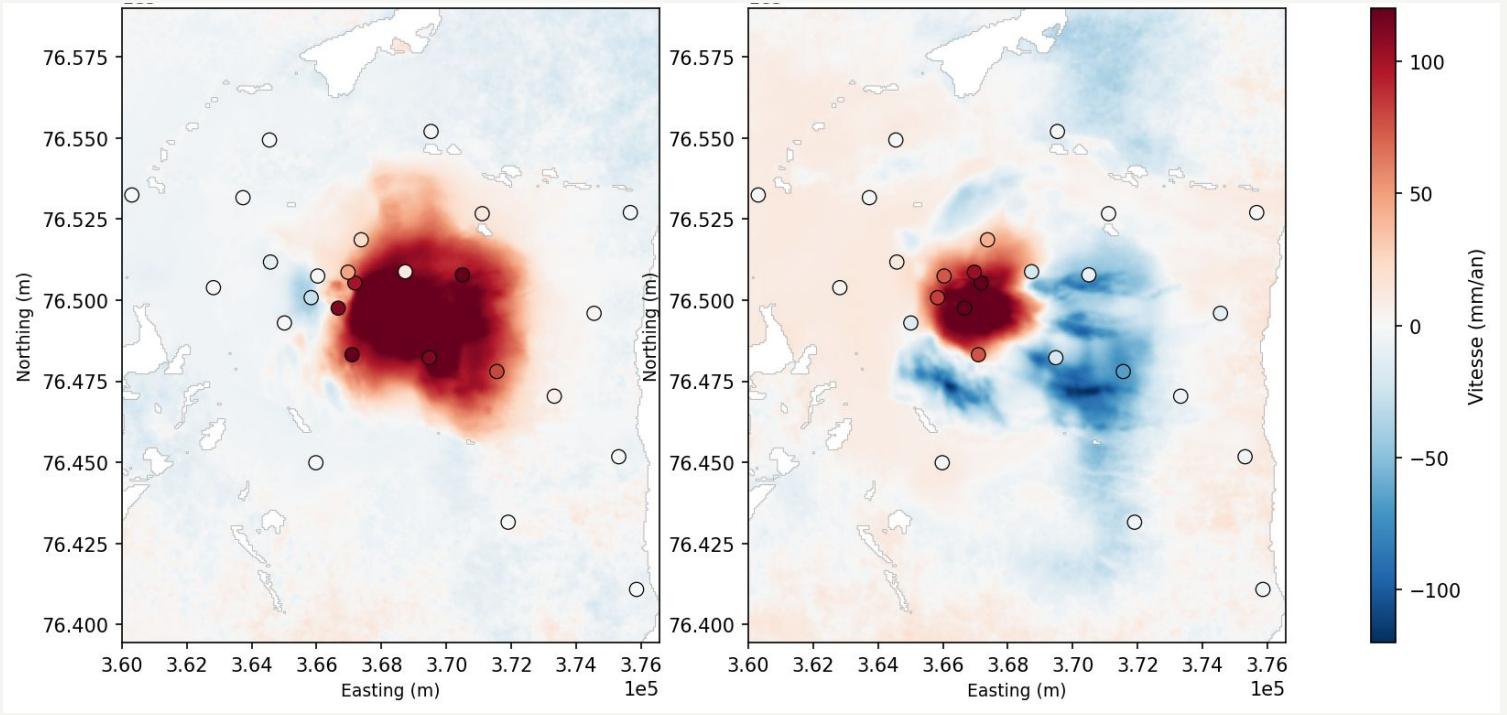


InSAR vs GNSS baseline time series (FJAG–BOMG, East component, 2021–2026)



AMSTer-derived movements vs GNSS





EW and UD velocity maps (Sentinel-1, 2016 to 2026) with co-located GNSS stations

InSAR validation with GNSS: Piton de la Fournaise

- Piton de la Fournaise is one of the most active volcanoes on Earth
- Dense GNSS network available
- Validation benchmark for InSAR-derived deformation:
  - InSAR vs GNSS time series
  - InSAR vs GNSS scatterplots

- Average velocity at each GNSS station
  - GNSS stations projected onto InSAR velocity maps
  - If coloured dots blend in → agreement
  - (new stations near crater without sufficient data)

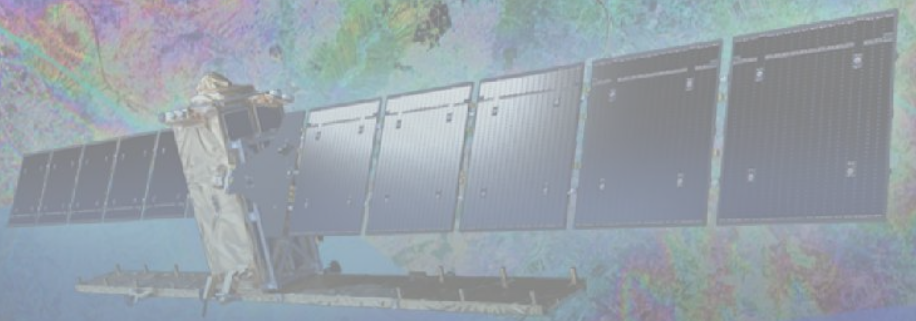
Smittarello, D (2026).
In prep.

Ministry of Science and Higher Education
Republic of Poland

JAGIELLONIAN UNIVERSITY
IN KRAKÓW

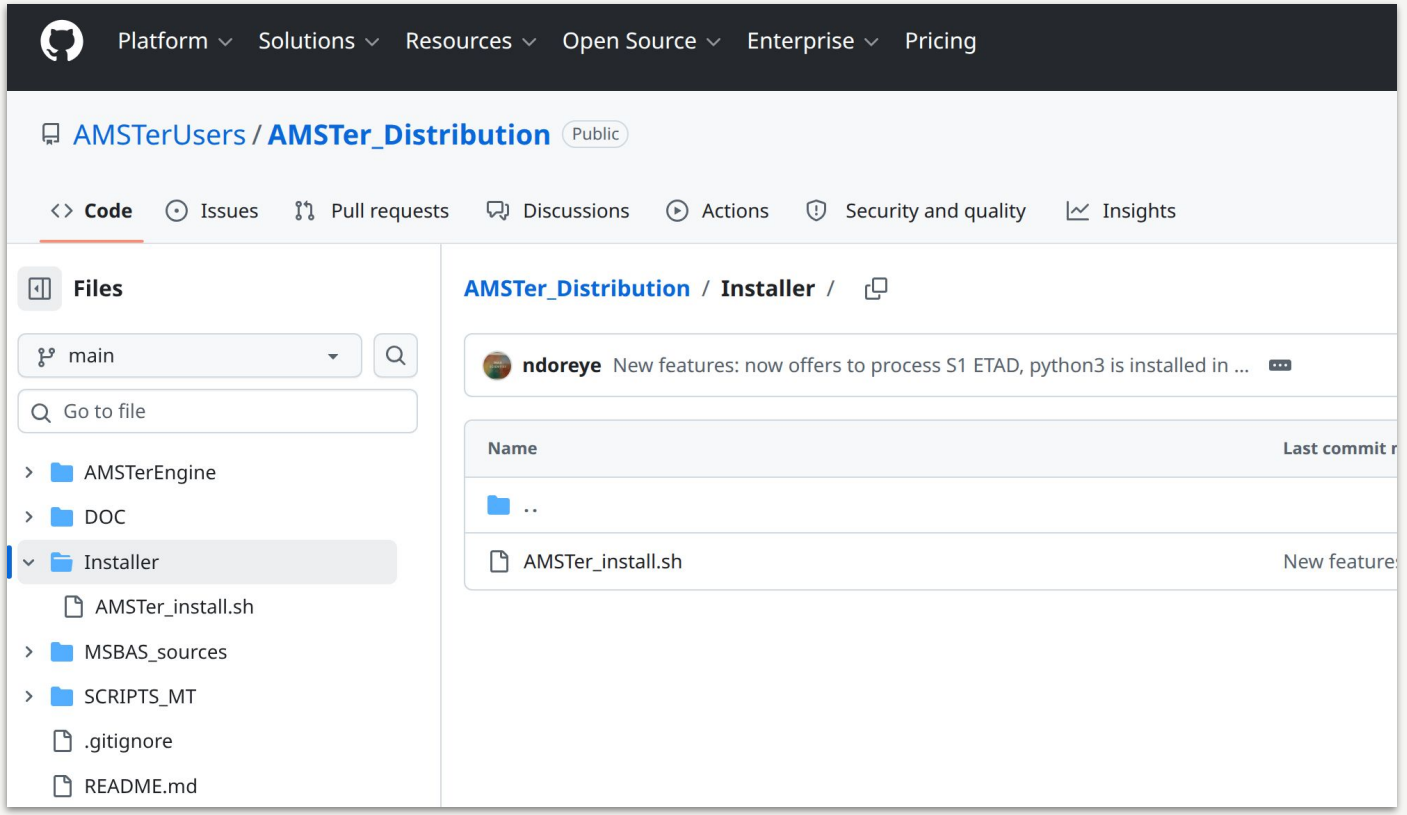
esa
EUROPEAN SPACE AGENCY

I. Why AMSTer?
II. AMSTer Toolbox Processing Chain
III. Operational services & processing environments
IV. Science studies & validation
V. Resources, community & outlook



github.com/AMSTerUsers/AMSTer_Distribution

Comprehensive documentation

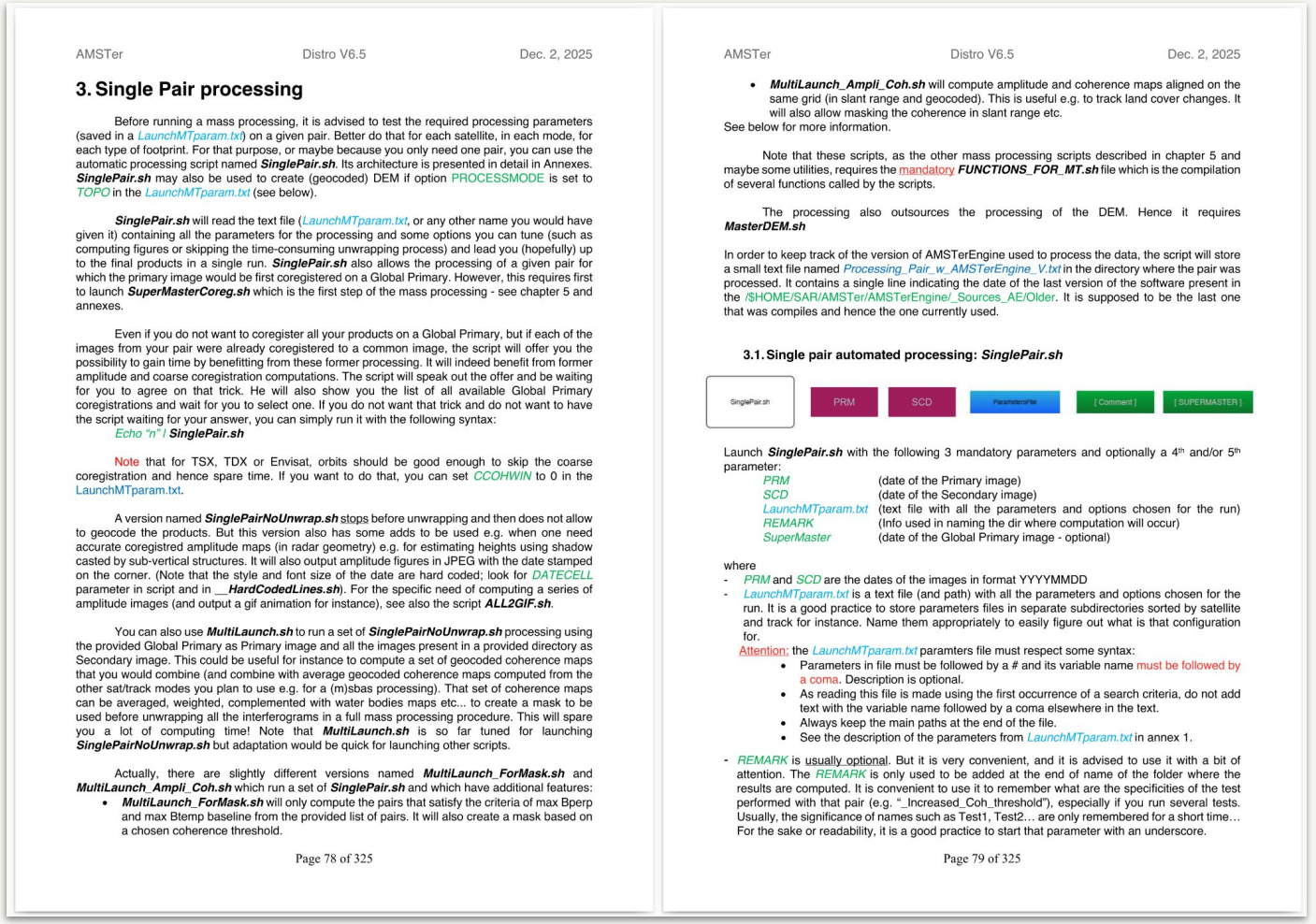


github.com/AMSTerUsers/AMSTer_Distribution

Installer & examples

Script-based install for Linux & macOS, worked examples per sensor

Comprehensive documentation



AMSTer Distro V6.5 Dec. 2, 2025

3. Single Pair processing

Before running a mass processing, it is advised to test the required processing parameters (saved in a *LaunchMTparam.txt*) on a given pair. Better do that for each satellite, in each mode, for each type of footprint. For that purpose, or maybe because you only need one pair, you can use the automatic processing script named ***SinglePair.sh***. Its architecture is presented in detail in Annexes. ***SinglePair.sh*** may also be used to create (geocoded) DEM if option PROCESSMODE is set to *TOPO* in the *LaunchMTparam.txt* (see below).

SinglePair.sh will read the text file (*LaunchMTparam.txt*, or any other name you would have given it) containing all the parameters for the processing and some options you can tune (such as computing figures or skipping the time-consuming unwrapping process) and lead you (hopefully) up to the final products in a single run. ***SinglePair.sh*** also allows the processing of a given pair for which the primary image would be first coregistered on a Global Primary. However, this requires first to launch ***SuperMasterCoreg.sh*** which is the first step of the mass processing - see chapter 5 and annexes.

Even if you do not want to coregister all your products on a Global Primary, but if each of the images from your pair were already coregistered to a common image, the script will offer you the possibility to gain time by benefitting from these former processing. It will indeed benefit from former amplitude and coarse coregistration computations. The script will speak out the offer and be waiting for you to agree on that trick. He will also show you the list of all available Global Primary coregistrations and wait for you to select one. If you do not want that trick and do not want to have the script waiting for your answer, you can simply run it with the following syntax:

*Echo "n" | **SinglePair.sh***

Note that for TSX, TDX or Envisat, orbits should be good enough to skip the coarse coregistration and hence spare time. If you want to do that, you can set *CCOHWIN* to 0 in the LaunchMTparam.txt.

A version named ***SinglePairNoUnwrap.sh*** stops before unwrapping and then does not allow to geocode the products. But this version also has some adds to be used e.g. when one need accurate coregistred amplitude maps (in radar geometry) e.g. for estimating heights using shadow casted by sub-vertical structures. It will also output amplitude figures in JPEG with the date stamped on the corner. (Note that the style and font size of the date are hard coded; look for *DATECELL* parameter in script and in ***__HardCodedLines.sh***). For the specific need of computing a series of amplitude images (and output a gif animation for instance), see also the script ***ALL2GIF.sh***.

You can also use ***MultiLaunch.sh*** to run a set of ***SinglePairNoUnwrap.sh*** processing using the provided Global Primary as Primary image and all the images present in a provided directory as Secondary image. This could be useful for instance to compute a set of geocoded coherence maps that you would combine (and combine with average geocoded coherence maps computed from the other sat/track modes you plan to use e.g. for a (m)sbas processing). That set of coherence maps can be averaged, weighted, complemented with water bodies maps etc... to create a mask to be used before unwrapping all the interferograms in a full mass processing procedure. This will spare you a lot of computing time! Note that ***MultiLaunch.sh*** is so far tuned for launching ***SinglePairNoUnwrap.sh*** but adaptation would be quick for launching other scripts.

Actually, there are slightly different versions named ***MultiLaunch_ForMask.sh*** and ***MultiLaunch_Ampli_Coh.sh*** which run a set of ***SinglePair.sh*** and which have additional features:

- ***MultiLaunch_ForMask.sh*** will only compute the pairs that satisfy the criteria of max Bperp and max Btemp baseline from the provided list of pairs. It will also create a mask based on a chosen coherence threshold.

Page 78 of 325

AMSTer Distro V6.5 Dec. 2, 2025

- ***MultiLaunch_Ampli_Coh.sh*** will compute amplitude and coherence maps aligned on the same grid (in slant range and geocoded). This is useful e.g. to track land cover changes. It will also allow masking the coherence in slant range etc.

See below for more information.

Note that these scripts, as the other mass processing scripts described in chapter 5 and maybe some utilities, requires the mandatory ***FUNCTIONS_FOR_MT.sh*** file which is the compilation of several functions called by the scripts.

The processing also outsources the processing of the DEM. Hence it requires ***MasterDEM.sh***

In order to keep track of the version of AMSTerEngine used to process the data, the script will store a small text file named *Processing_Pair_w_AMSTerEngine_V.txt* in the directory where the pair was processed. It contains a single line indicating the date of the last version of the software present in the /$HOME/SAR/AMSTer/AMSTerEngine/_Sources_AE/Older. It is supposed to be the last one that was compiles and hence the one currently used.

3.1. Single pair automated processing: *SinglePair.sh*

Launch ***SinglePair.sh*** with the following 3 mandatory parameters and optionally a 4th and/or 5th parameter:

PRM	(date of the Primary image)
SCD	(date of the Secondary image)
LaunchMTparam.txt	(text file with all the parameters and options chosen for the run)
REMARK	(Info used in naming the dir where computation will occur)
SuperMaster	(date of the Global Primary image - optional)

where

- *PRM* and *SCD* are the dates of the images in format YYYYMMDD
- *LaunchMTparam.txt* is a text file (and path) with all the parameters and options chosen for the run. It is a good practice to store parameters files in separate subdirectories sorted by satellite and track for instance. Name them appropriately to easily figure out what is that configuration for.

  Attention: the *LaunchMTparam.txt* paramters file must respect some syntax:
  - Parameters in file must be followed by a # and its variable name must be followed by a coma. Description is optional.
  - As reading this file is made using the first occurrence of a search criteria, do not add text with the variable name followed by a coma elsewhere in the text.
  - Always keep the main paths at the end of the file.
  - See the description of the parameters from *LaunchMTparam.txt* in annex 1.
- *REMARK* is usually optional. But it is very convenient, and it is advised to use it with a bit of attention. The *REMARK* is only used to be added at the end of name of the folder where the results are computed. It is convenient to use it to remember what are the specificities of the test performed with that pair (e.g. "_Increased_Coh_threshold"), especially if you run several tests. Usually, the significance of names such as Test1, Test2… are only remembered for a short time… For the sake or readability, it is a good practice to start that parameter with an underscore.

Page 79 of 325

github.com/AMSTerUsers/AMSTer_Distribution

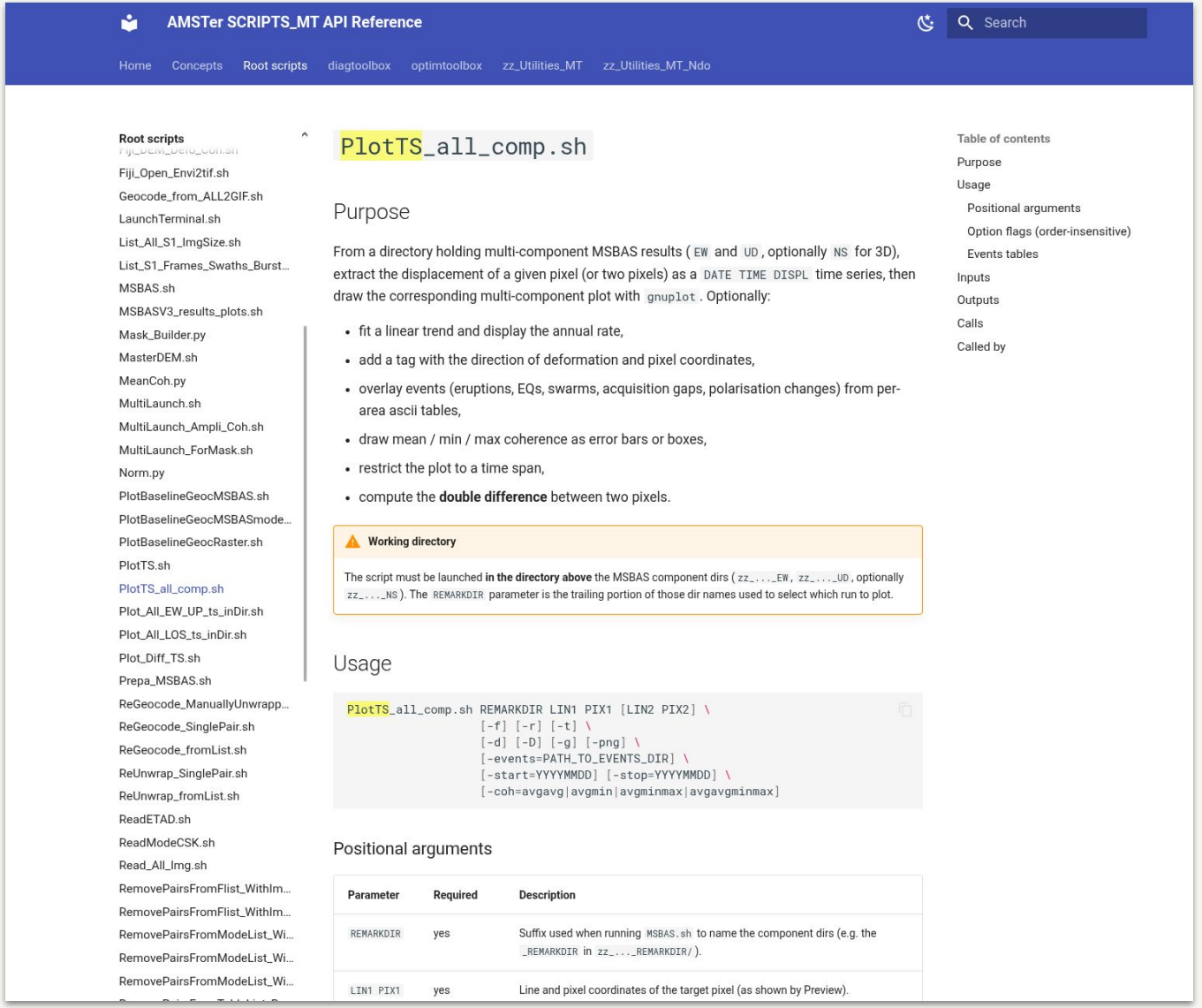
Installer & examples

Script-based install for Linux & macOS, worked examples per sensor

650+ page user manual

AMSTer Manual v6.5 for stable release, comprehensive guide to all processing steps

Comprehensive documentation



github.com/AMSTerUsers/AMSTer_Distribution

Installer & examples

Script-based install for Linux & macOS, worked examples per sensor

650+ page user manual

AMSTer Manual v6.5 for stable release, comprehensive guide to all processing steps

Online API reference

mkdocs-material site: per-script documentation for ~400 scripts; includes: Purpose, usage, arguments, input, output, dependencies.

Availability and collaboration

Ministry of Science and Higher Education
Republic of Poland

JAGIELLONIAN UNIVERSITY IN KRAKÓW

esa EUROPEAN SPACE AGENCY

Welcoming new users

AMSTer welcomes new users, but is designed for researchers with SAR / InSAR knowledge. It is a powerful research-grade toolbox, not a push-button solution. Best-effort support is provided through GitHub Issues and direct contact.

On-site training

ECGS can welcome visiting researchers and students
for hands-on training and collaborative work.
Availability is limited, contact us to discuss.

Active user community

Students, PhD candidates, post-docs, and institutional researchers
across Europe and beyond are using AMSTer in their work.

AGPL v3, open source

Free for research use. Service use requires source disclosure.
GitHub repository with documentation and installer scripts.

Training and community building

Organization of summer schools and trainings

AMSTer Summer School training students, PhD researchers, and institutional partners on InSAR processing, at the Chamber of Commerce, Luxembourg

Summer School in InSAR, time series processing and deformation modelling

“If you need InSAR... MasTer it!"

22–26 May 2023
Training at Chambre of Commerce, Luxembourg (20 participants)

MDIS-FormaTerre 2026

Mesure de la Deformation par Imagerie Satellitaire

14–15 Sept 2026
Training at ENSG Nancy, France (85 participants)

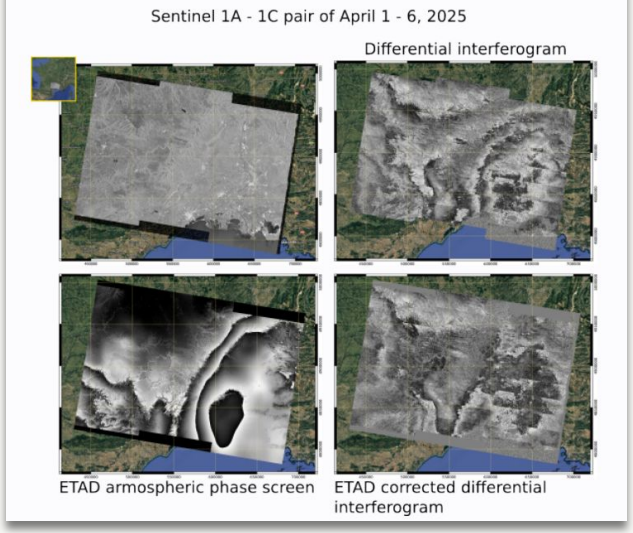
16–18 Sept 2026
Scientific symposium, Xonrupt-Longemer (Vosges, France) (115 speakers, guests, and scientists)

Co-organised by OTELo (Nancy) & ECGS (Luxembourg)

mdis-2026.sciencesconf.org

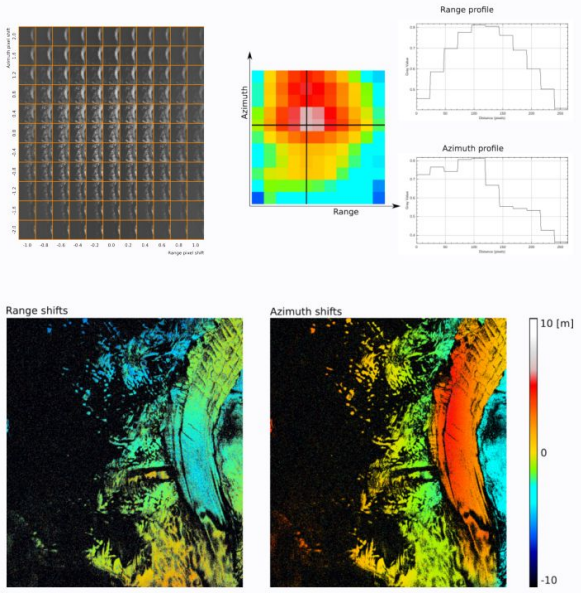
See also at FRINGE 2026: Today

Example of **ETAD** management for atmospheric phase screen removal (ETAD v2.0)



ETAD atmospheric phase screen removal

Coherence mapping and tracking results



Coherence mapping and tracking results

Recent developments in AMSTer Engine

ETAD and coherence tracking integration in AMSTer Toolbox

D. Derauw et al.

- Sentinel-1 ETAD atmospheric phase screen removal for geodetic-quality corrections.
- Coherent cross-correlation for 2D displacement retrieval, complementing interferometric approaches.

Meet Dominique Derauw, lead developer of AMSTer Engine!

POSTER SESSION I
Tuesday 4:30 pm – 7:00 pm

See also at FRINGE 2026: Tomorrow

Case study enabled by AMSTer

Resolving 3D Deformation of Piton de la Fournaise

D. Smittarello et al.

Enabled by Dense Multi-Geometry ALOS-2 Acquisitions

- 22 geometries
- 1,600 ALOS-2 images
- 7,700 interferograms
- True 3D MSBAS inversion

Volcanoes & volcanic hazards 1
Wednesday 9:20 am – 9:40 am

Access, roadmap, and sustainability

github.com/AMSTerUsers/AMSTer_Distribution

AGPL v3

Free for research; service use requires source disclosure

Documentation

650+ page manual + mkdocs API reference

Installer

Script-based install for Linux & macOS

Roadmap

- BIOMASS integration (under way)
- NISAR next batch preparation
- Coherence Tracking revamp (see Derauw's poster)
- Full API documentation online
- GEP integration (ongoing, under E-coh CCN)
- Revamped Extended Spectral Diversity
- Next workshop preparation and associated training material

Actively maintained and evolving

Ministry of Science and Higher Education
Republic of Poland

JAGIELLONIAN UNIVERSITY
IN KRAKÓW

esa
EUROPEAN SPACE AGENCY

- **An open, automated, multi-sensor InSAR chain**
- **from download to multidimensional time series and web dissemination**
- **running operationally at ECGS on a dozen targets worldwide.**

BREADTH	DEPTH	OPERATIONAL	OPEN
12+ satellite families ERS-1 to NISAR	2D/3D MSBAS multi-dim inversion	12+ targets 25+ years	AGPL v3 GitHub + docs

github.com/AMSTerUsers/AMSTer_Distribution

Ministry of Science and Higher Education
Republic of Poland

JAGIELLONIAN UNIVERSITY
IN KRAKÓW

esa
EUROPEAN SPACE AGENCY

Questions?

Quentin Glaude

quentin.glaude@ecgs.lu

European Center for Geodynamics and Seismology, Luxembourg

Co-authors: N. d'Oreye, D. Derauw, S. Samsonov, D. Smittarello, M. Jaspard, G. Celli

github.com/AMSTerUsers/AMSTer_Distribution