



MDIS 2019 : interfero tutorial with DiapOTB

P. Durand, V. Poughon, G. Usseglio, M. Azzoni

♦ HOW to easily download S1 (or S2 data) from PEPS:
focus on script PEPS_DOWNLOAD.py

♦ DiapOTB

♦ MultiSLC : a script for time series inteferograms or
coregistered SLC on a same MASTER image

peps_download.py : options (*script O. Hagolle*)



A script is often more convenient than interactive download, especially on clusters

how it works : login PEPS needed

Copy python procedure peps_download.py and peps.txt (login) in the same directory of future SAFE data

module load python

```
python peps_download.py -a peps.txt -c S1 -p SLC -m SM -d 2016-08-15 -f 2016-10-01 --  
latmin -21.5 --latmax -21.25 --lonmin 55.5 --lonmax 55.7
```

unzip files

delete polar vh (usually not used - save space)

```
rm ./S1*.SAFE/measurement/s1*-*-vh-*-*-*.tiff
```

peps_download.py : options *(script O. Hagolle)*



-h, --help *show this help message and exit*

-l LOCATION, --location=LOCATION *town name (pick one which is not too frequent to avoid confusions)*

-a AUTH, --auth=AUTH *Peps account and password file MDIS.txt tempo@mdis.fr Mdis2019!*

-w WRITE_DIR, --write_dir=WRITE_DIR *Path where the products should be downloaded*

-c COLLECTION, --collection=COLLECTION *Collection within theia or PEPS collections (S1 S2 etc)*

-p PRODUCT_TYPE, --product_type=PRODUCT_TYPE *GRD, SLC, OCN (for S1) | S2MSI1C S2MSI2A S2MSI2Ap (for S2)*

-m SENSOR_MODE, --sensor_mode=SENSOR_MODE *EW, IW, SM, WV (for S1) | INS-NOBS, INS-RAW (for S3)*

-n, --no_download *Do not download products, just print curl command*

-d START_DATE, --start_date=START_DATE *start date, fmt('2015-12-22')*

-f END_DATE, --end_date=END_DATE *end date, fmt('2015-12-23')*

--json=SEARCH_JSON_FILE *Output search JSON filename*

--windows *For windows usage*

-M=MASTER IMAGE *set the master image file name (.SAFE)*

-t TILE, --tile=TILE *Sentinel-2 tile number*

--lat=LAT *latitude in decimal degrees*

--lon=LON *longitude in decimal degrees*

--latmin=LATMIN *min latitude in decimal degrees*

--latmax=LATMAX *max latitude in decimal degrees*

--lonmin=LONMIN *min longitude in decimal degrees*

--lonmax=LONMAX *max longitude in decimal degrees*

-o ORBIT, --orbit=ORBIT *Orbit Path number*

peps_download.py : examples



example 1 : `python peps_download.py -l 'Toulouse' -a peps.txt -d 2016-12-06 -f 2017-02-01 -c S2ST`

example 2 : `python peps_download.py --lon 1 --lat 44 -a peps.txt -d 2015-11-01 -f 2015-12-01 -c S2`

example 3 : `python peps_download.py --lonmin 1 --lonmax 2 --latmin 43 --latmax 44 -a peps.txt -d 2015-11-01 -f 2015-12-01 -c S2`

example 4 : `python peps_download.py -l 'Toulouse' -a peps.txt -c SpotWorldHeritage -p SPOT4 -d 2005-11-01 -f 2006-12-01`

example 5 : `python peps_download.py -c S1 -p GRD -l 'Toulouse' -a peps.txt -d 2015-11-01 -f 2015-12-01`

Example SM Reunion Island

`python peps_download.py -n -a mdis.txt -c S1 -p SLC -m SM -d 2016-08-15 -f 2016-10-01 --latmin -21.5 --latmax -21.25 --lonmin 55.5 --lonmax 55.7`

peps_download.py : SM Reunion on disk



```
python peps_download.py -n -a mdis.txt -c S1 -p SLC -m SM -d 2016-08-15 -f 2016-10-01 --latmin -21.5 --latmax -21.25 --lonmin 55.5 --lonmax 55.7
```

```
curl -k -o search.json https://peps.cnes.fr/resto/api/collections/S1/search.json?box=55.5,-21.5,55.7,-21.25\&startDate=2016-08-15\&completionDate=2016-10-01\&maxRecords=500\&productType=SLC\&sensorMode=SM
```

```
#####
```

```
6 products to download
```

```
#####
```

% Total	% Received	% Xferd	Average	Speed	Time	Time	Time	Current	Dload	Upload	Total	Spent	Left	Speed
100	36401	0	36401	0	0	82979	0	--:--:--	--:--:--	--:--:--	83107			
S1B_S4_SLC__1SDV_20160929T014610_20160929T014634_002277_003D71_788E	disk													
S1B_S6_SLC__1SDV_20160928T145211_20160928T145235_002270_003D30_C8EC	disk													
S1A_S4_SLC__1SDV_20160923T014652_20160923T014716_013173_014F12_6794	disk													
S1A_S6_SLC__1SDV_20160922T145253_20160922T145317_013166_014EDB_653E	disk													
S1A_S4_SLC__1SDV_20160818T014650_20160818T014715_012648_013DB1_0969	disk													
S1A_S6_SLC__1SDV_20160817T145252_20160817T145315_012641_013D72_EE0D	disk													

peps_download.py : Reunion SM on tape



```
python peps_download.py -n -a mdis.txt -c S1 -p SLC -m SM -d 2015-08-15 -f 2015-10-01 --latmin -21.5 --latmax -21.25 --lonmin 55.5 --lonmax 55.7
```

```
curl -k -o search.json https://peps.cnes.fr/resto/api/collections/S1/search.json?box=55.5,-21.5,55.7,-21.25\&startDate=2015-08-15\&completionDate=2015-10-01\&maxRecords=500\&productType=SLC\&sensorMode=SM
```

```
% Total    % Received % Xferd Average Speed   Time    Time     Time Current
                              Dload  Upload  Total  Spent  Left  Speed
100 13227    0 13227    0    0 44104    0 --:--:-- --:--:-- --:--:-- 44385
S1A_S6_SLC__1SDV_20150928T145246_20150928T145310_007916_00B0E2_30CD tape
S1A_S6_SLC__1SDV_20150823T145245_20150823T145309_007391_00A2AA_7A2B tape
#####
2 products to download
#####
```

DiapOTB Context



Technical context : Port a legacy processing chain for SAR interferometry called Diapason as an Orfeo ToolBox remote module (DiapOTB) funded by CNES.

Scientific context : The differential SAR interferometry (DInSAR)

- ◆ Processing of two SAR images of the same portion of the Earth's surface taken at different time.
- ◆ Highlighting differences between SAR images to analyse potential events (earthquake, destruction, ...).

Aims :

- ◆ Launching a complete DInSAR chain with a simple python script or standalone applications.
- ◆ Re-using processing.

Data : Sentinel-1 with SM and IW mode, Cosmo SM and Spotlight, in the near future TerraSAR-X

DiapOTB Applications



General SAR Applications :

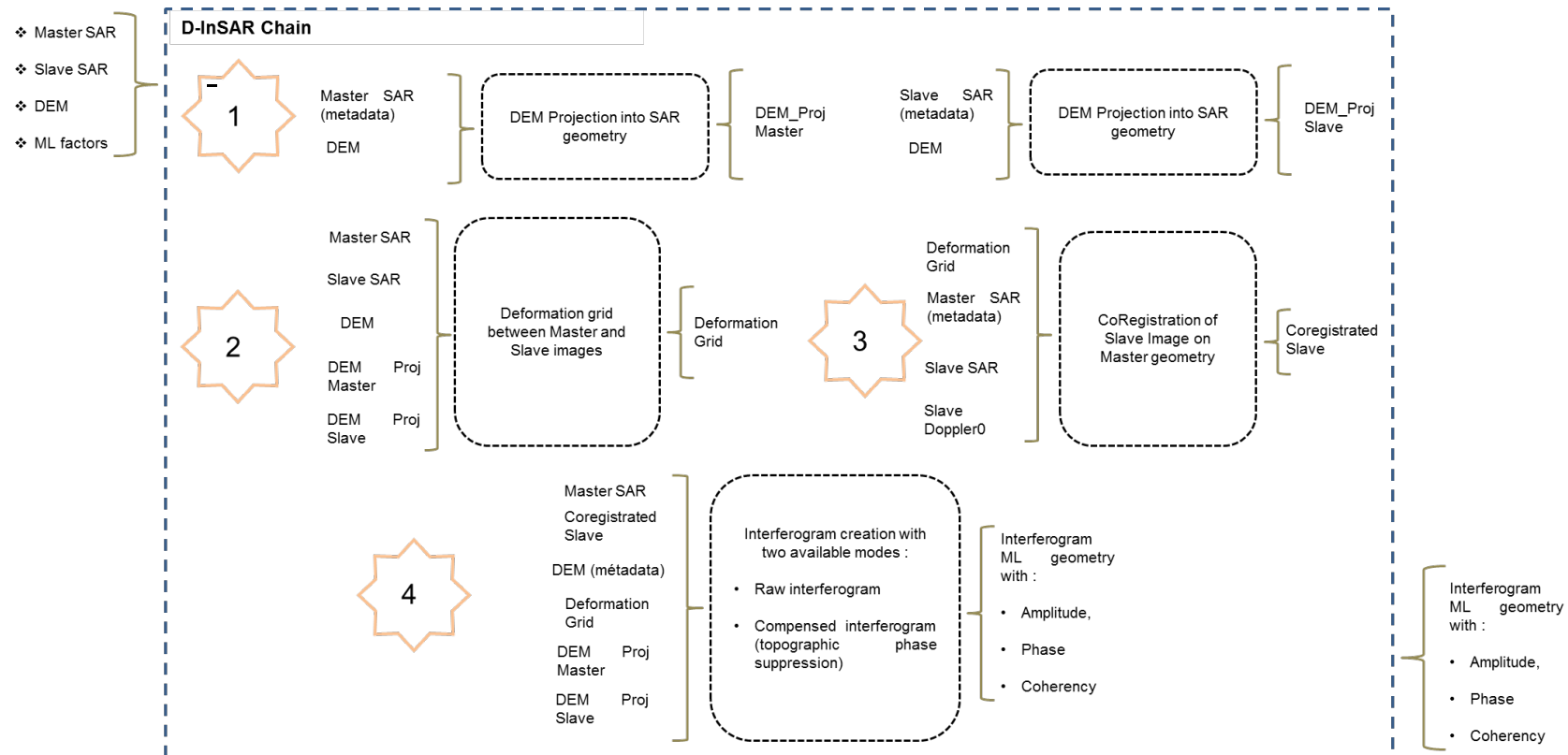
- ◆ Doppler0 estimation
- ◆ MultiLook
- ◆ Simulation of SAR image from Digital Elevation Model
- ◆ Projection of Digital Elevation Model into SAR geometry

Specific Interferometry Applications :

- ◆ Deformation grids creation
- ◆ CoRegistration
- ◆ Interferogram

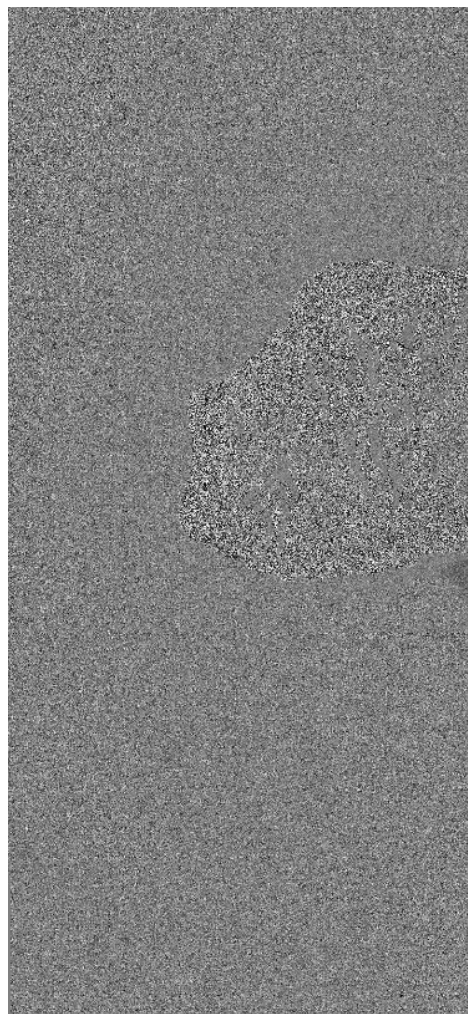
DiapOTB Chain

DInSAR Chain in Python : Use previous applications as parts of a larger processing workflow to get a final interferogram from two SLC (single look complex) images and a digital elevation model.

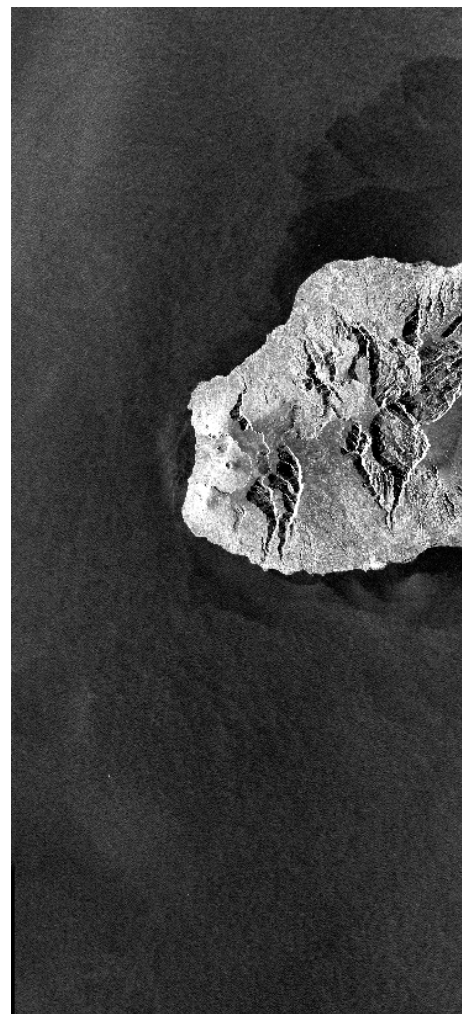


Example of DiapOTB applications : SARMultiLook

SLC
Image

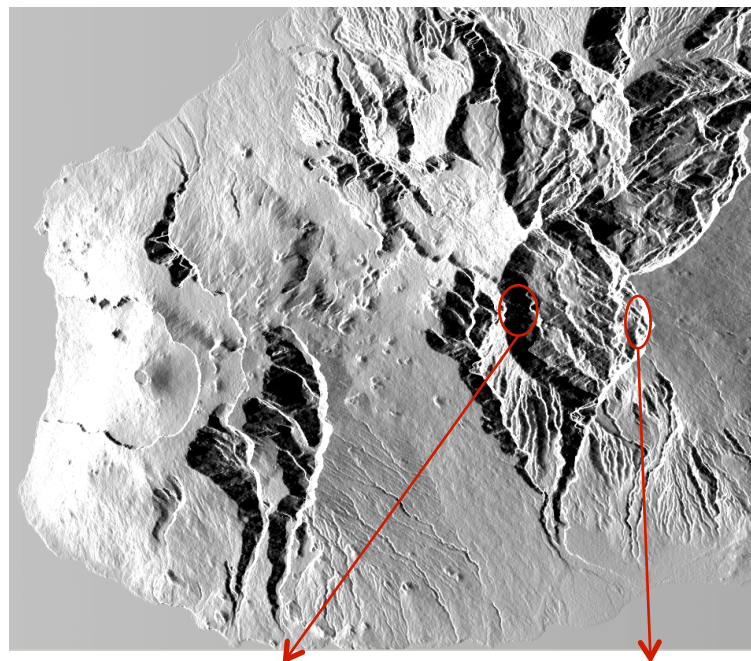
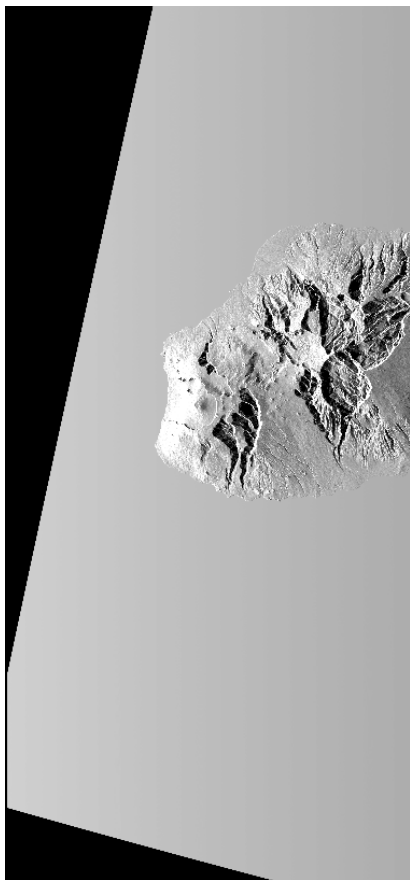


Real image
ML (3x3)



Example of DiapOTB application : SARDEMToAmplitude

Get an amplitude simulated image from only a DEM and SAR metadata



Shadowing

Layover

A generic filter was developed to estimate amplitude. This filter can be easily extended to different purposes such as calculating Shadowing or Layover masks.

Example of results for DiapOTB chain



Data : Sentinel-1 StripMap mode (Reunion)

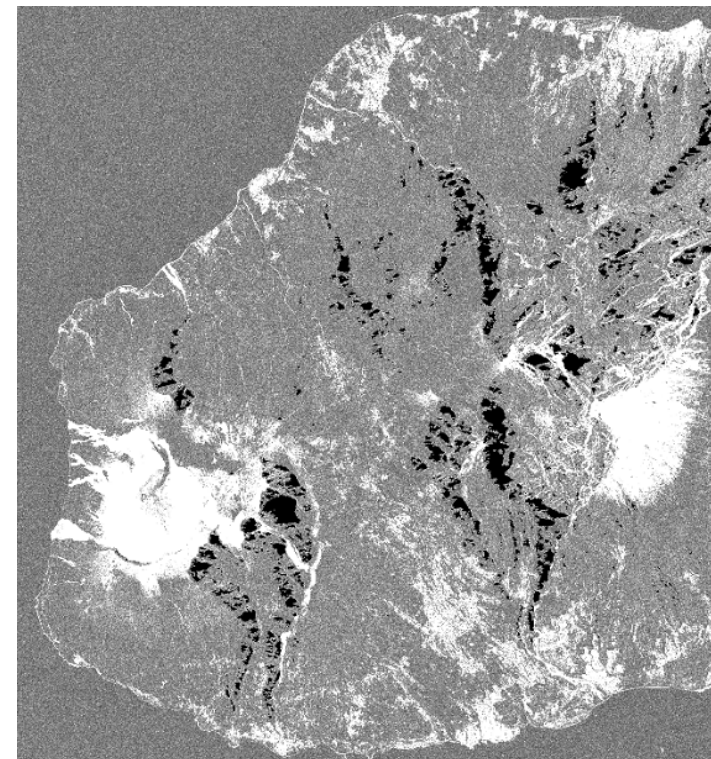
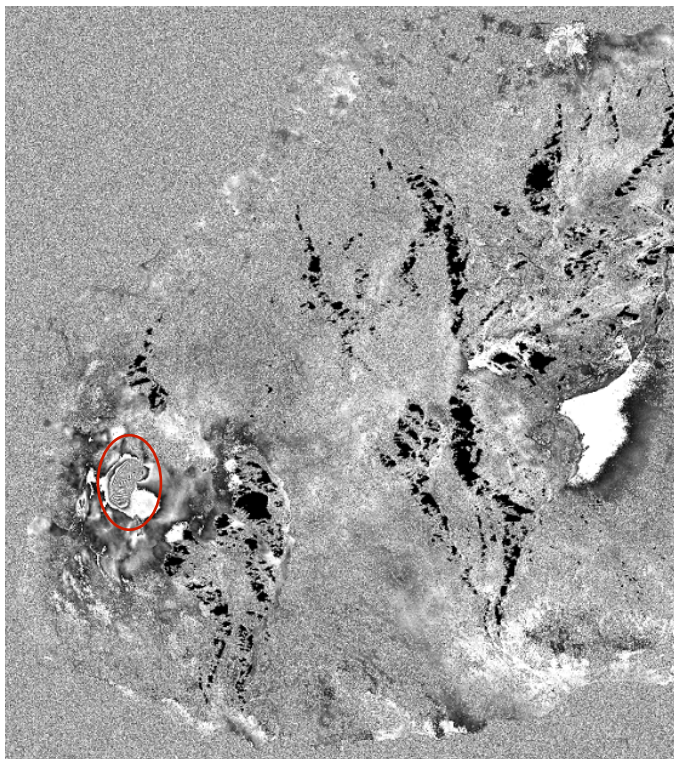
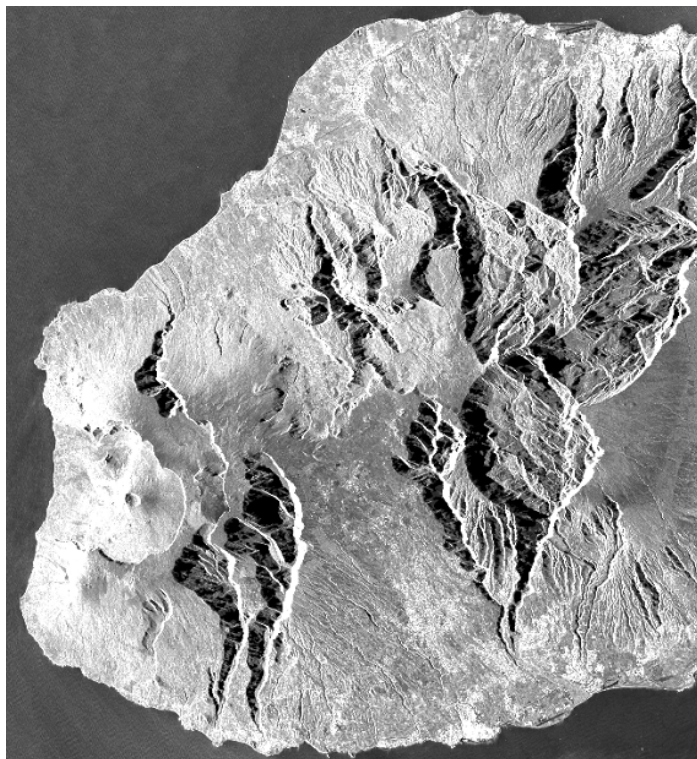
Results : Interferogram with 3x3 factors for averaging and representing by a Vector Image :

- ◆ Band 1 : Amplitude
- ◆ Band 2 : Phase
- ◆ Band 3 : Coherency
- ◆ Band 4 : IsData boolean to indicate if output pixels are consistent

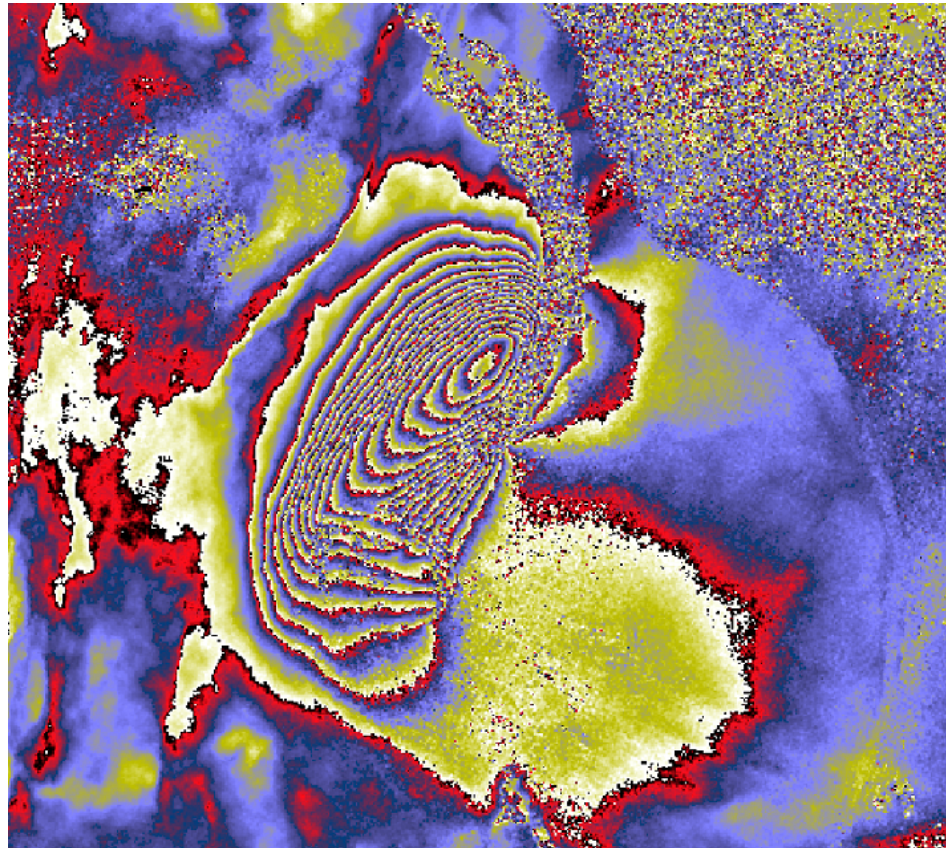
Performances :

- ◆ DiapOTB applications are multi-threaded applications
- ◆ Runs on CNES HPC cluster (HAL)
- ◆ Around 8 minutes for the complete StripMap chain on 1 node with 24 Cpus

Example of results for DiapOTB chain



Example of results for DiapOTB chain



Displacement near piton de la fournaise between our input images taken on 2016/08/18 and 2016/09/29 with S1-A and S1-B respectively.

DiapOTB prospects



- Available Now

gitlab.orfeo-toolbox.org/remote_modules/diapotb

Support for S1 SM and S1 IW and Cosmo SM and Spotlight

- Phase unwrapping not available yet : any willing person welcome !

DiapOTB scripts examples



example :

Multi-look

```
qsub -l -X -l select=1:ncpus=12:mem=60000Mb -l walltime=10:00:00
```

to affect RAM and CPUs

```
export MODULEPATH=/Modules_DiapOTB/modulefiles:$MODULEPATH
```

```
module load diapOTB
```

```
export ITK_GLOBAL_DEFAULT_NUMBER_OF_THREADS=12
```

for multithreading

```
cd /working_directory/
```

Configure parameters in .json

```
python $DIAPOTB_HOME/python_src/diapOTB_S1IW.py config.json
```

for launching

Boîte à outils



Orfeo Toolbox

_ Outils de traitements raster et vecteur.



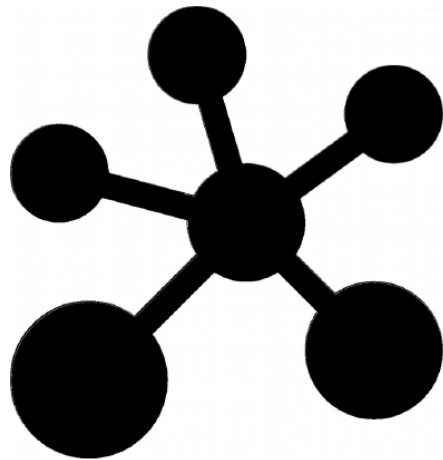
GDAL

_ Bibliothèque d'outils pour traitements Raster/Vecteur.
_ API Python



Python

_ Langage d'écriture du script



Cluster Linux

_ Cluster CNES, Grappe de serveurs CNES, outils de calculs puissants.
_ 24 cpus, 120Go, 24 thread.



PEPS

_ Plateforme du CNES destinée au téléchargement des images Sentinel.



Diap OTB

_ Issu de la chaîne Diapason.
_ Outils de traitements interférométriques.

Chaîne de traitements D-InSar

Principe de la chaîne D-InSar_MultiSlc

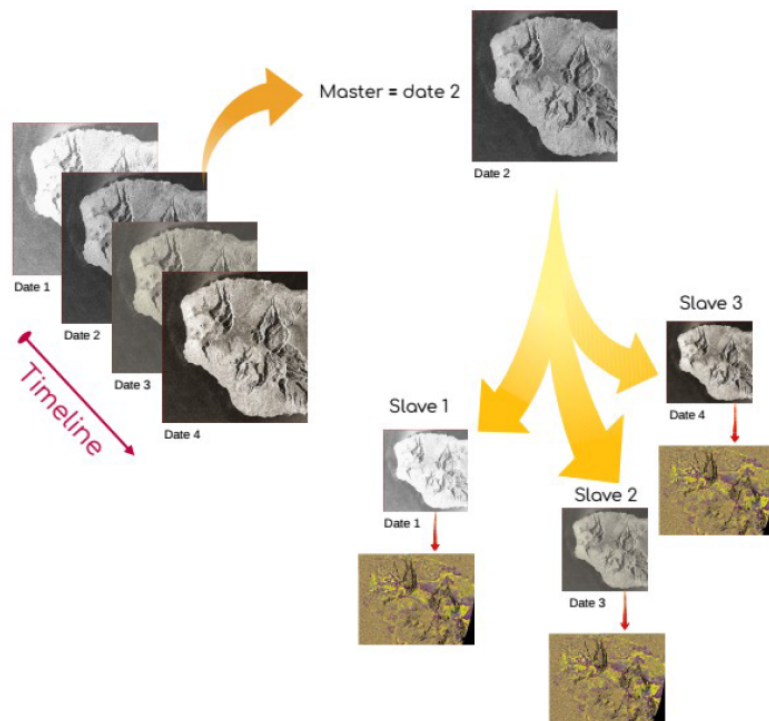
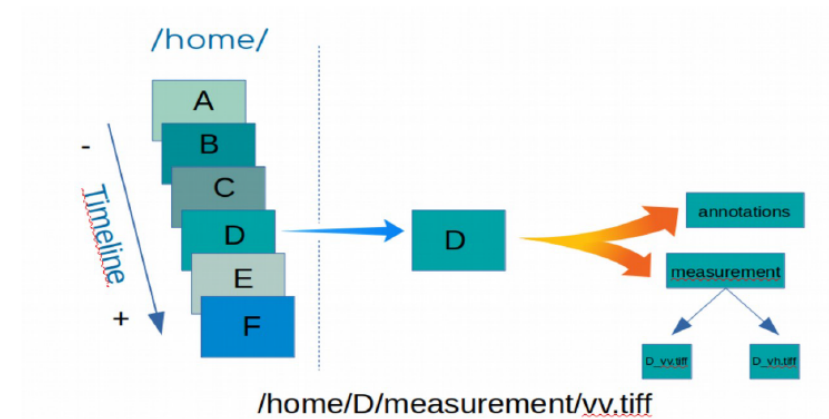


Illustration d'un chemin de fichier pour Sentinel 1



Notre script doit pouvoir retrouver le chemin d'après le nom seul.

Liste des options du script :

-d : date de début

-f : date de fin

-e : dates à exclure

-m : image maîtresse

-b : bursts à traiter **IW**

-roi : Définir une ROI (ft : ulx uly lrx lry)

-i : interférogramme (activée par défaut)

-o : Orthorectification

-l : Light version (activée par défaut)

-ram : Définir la ram (4Go à par défaut)

-spacingxy : Résolution spatiale pour l'orthorectification (rééchantillonnage 0.0001° par défaut)

-ml : Multilook range x azimuth

Principes de fonctionnement

. Lancement en ligne de commande

. Le script se lance en duo avec un fichier de configuration

Ml Range Azimut

Grid Threshold
Grid Gap
Gain [...]

. Choix date début, date fin, et si besoin des dates à exclure

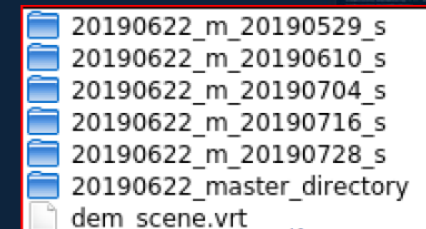
. Choix de la maîtresse

Organisation du dossier de sortie

. Sortie interférogramme ou non.

. Orthorectification ou non.

. Light mode ou full content



20190622_m_20190529_s
20190622_m_20190610_s
20190622_m_20190704_s
20190622_m_20190716_s
20190622_m_20190728_s
20190622_master_directory
dem_scene.vrt

Création automatique du MNT de la zone

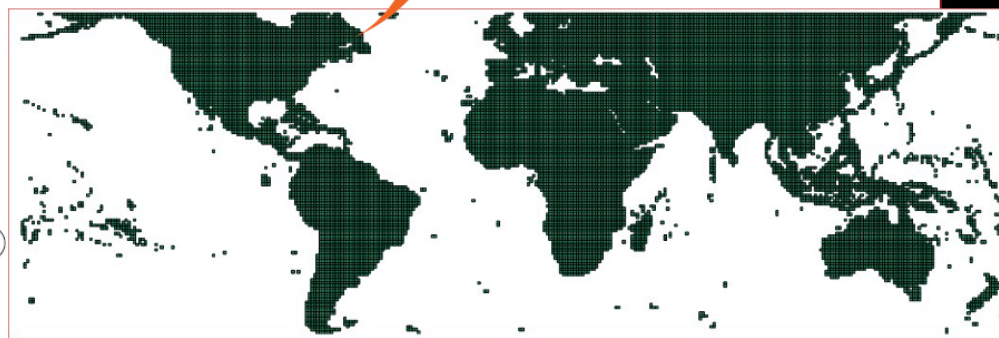
Le script intègre une fonction de S1Tilling* pour la création d'un Modèle Numérique de Terrain



Création d'une empreinte shapefile de l'image

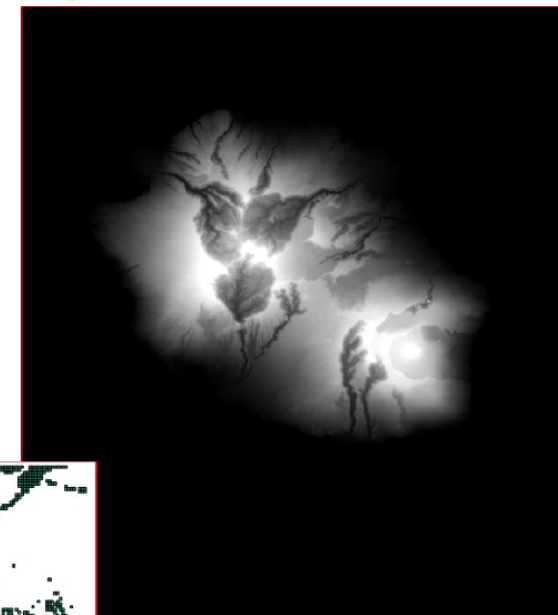


Intersection de l'image sur les tuiles SRTM



N390009.bgt	N390020.bgt	N390002.bgt	N390009.bgt	S10E118.bgt
N390010.bgt	N390021.bgt	N390003.bgt	N390010.bgt	S10E119.bgt
N390011.bgt	N390022.bgt	N390004.bgt	N390011.bgt	S10E120.bgt
N390012.bgt	N390023.bgt	N390005.bgt	N390012.bgt	S10E121.bgt
N390013.bgt	N390024.bgt	N390006.bgt	N390013.bgt	S10E122.bgt
N390014.bgt	N390025.bgt	N390007.bgt	N390014.bgt	S10E123.bgt
N390015.bgt	N390026.bgt	N390008.bgt	N390015.bgt	S10E124.bgt
N390016.bgt	N390027.bgt	N390009.bgt	N390016.bgt	S10E125.bgt
N390017.bgt	N390028.bgt	N390010.bgt	N390017.bgt	S10E126.bgt
N390018.bgt	N390029.bgt	N390011.bgt	N390018.bgt	S10E127.bgt
N390019.bgt	N390030.bgt	N390012.bgt	N390019.bgt	S10E128.bgt
N390020.bgt	N390031.bgt	N390013.bgt	N390020.bgt	S10E129.bgt
N390021.bgt	N390032.bgt	N390014.bgt	N390021.bgt	S10E130.bgt
N390022.bgt	N390033.bgt	N390015.bgt	N390022.bgt	S10E131.bgt
N390023.bgt	N390034.bgt	N390016.bgt	N390023.bgt	S10E132.bgt
N390024.bgt	N390035.bgt	N390017.bgt	N390024.bgt	S10E133.bgt
N390025.bgt	N390036.bgt	N390018.bgt	N390025.bgt	S10E134.bgt
N390026.bgt	N390037.bgt	N390019.bgt	N390026.bgt	S10E135.bgt
N390027.bgt	N390038.bgt	N390020.bgt	N390027.bgt	S10E136.bgt
N390028.bgt	N390039.bgt	N390021.bgt	N390028.bgt	S10E137.bgt
N390029.bgt	N390040.bgt	N390022.bgt	N390029.bgt	S10E138.bgt
N390030.bgt	N390041.bgt	N390023.bgt	N390030.bgt	S10E139.bgt
N390031.bgt	N390042.bgt	N390024.bgt	N390031.bgt	S10E140.bgt
N390032.bgt	N390043.bgt	N390025.bgt	N390032.bgt	S10E141.bgt
N390033.bgt	N390044.bgt	N390026.bgt	N390033.bgt	S10E142.bgt
N390034.bgt	N390045.bgt	N390027.bgt	N390034.bgt	S10E143.bgt
N390035.bgt	N390046.bgt	N390028.bgt	N390035.bgt	S10E144.bgt
N390036.bgt	N390047.bgt	N390029.bgt	N390036.bgt	S10E145.bgt
N390037.bgt	N390048.bgt	N390030.bgt	N390037.bgt	S10E146.bgt
N390038.bgt	N390049.bgt	N390031.bgt	N390038.bgt	S10E147.bgt
N390039.bgt	N390050.bgt	N390032.bgt	N390039.bgt	S10E148.bgt
N390040.bgt	N390051.bgt	N390033.bgt	N390040.bgt	S10E149.bgt
N390041.bgt	N390052.bgt	N390034.bgt	N390041.bgt	S10E150.bgt
N390042.bgt	N390053.bgt	N390035.bgt	N390042.bgt	S10E151.bgt
N390043.bgt	N390054.bgt	N390036.bgt	N390043.bgt	S10E152.bgt
N390044.bgt	N390055.bgt	N390037.bgt	N390044.bgt	S10E153.bgt
N390045.bgt	N390056.bgt	N390038.bgt	N390045.bgt	S10E154.bgt
N390046.bgt	N390057.bgt	N390039.bgt	N390046.bgt	S10E155.bgt
N390047.bgt	N390058.bgt	N390040.bgt	N390047.bgt	S10E156.bgt
N390048.bgt	N390059.bgt	N390041.bgt	N390048.bgt	S10E157.bgt
N390049.bgt	N390060.bgt	N390042.bgt	N390049.bgt	S10E158.bgt
N390050.bgt	N390061.bgt	N390043.bgt	N390050.bgt	S10E159.bgt
N390051.bgt	N390062.bgt	N390044.bgt	N390051.bgt	S10E160.bgt
N390052.bgt	N390063.bgt	N390045.bgt	N390052.bgt	S10E161.bgt
N390053.bgt	N390064.bgt	N390046.bgt	N390053.bgt	S10E162.bgt
N390054.bgt	N390065.bgt	N390047.bgt	N390054.bgt	S10E163.bgt
N390055.bgt	N390066.bgt	N390048.bgt	N390055.bgt	S10E164.bgt
N390056.bgt	N390067.bgt	N390049.bgt	N390056.bgt	S10E165.bgt
N390057.bgt	N390068.bgt	N390050.bgt	N390057.bgt	S10E166.bgt
N390058.bgt	N390069.bgt	N390051.bgt	N390058.bgt	S10E167.bgt
N390059.bgt	N390070.bgt	N390052.bgt	N390059.bgt	S10E168.bgt
N390060.bgt	N390071.bgt	N390053.bgt	N390060.bgt	S10E169.bgt
N390061.bgt	N390072.bgt	N390054.bgt	N390061.bgt	S10E170.bgt
N390062.bgt	N390073.bgt	N390055.bgt	N390062.bgt	S10E171.bgt
N390063.bgt	N390074.bgt	N390056.bgt	N390063.bgt	S10E172.bgt
N390064.bgt	N390075.bgt	N390057.bgt	N390064.bgt	S10E173.bgt
N390065.bgt	N390076.bgt	N390058.bgt	N390065.bgt	S10E174.bgt
N390066.bgt	N390077.bgt	N390059.bgt	N390066.bgt	S10E175.bgt
N390067.bgt	N390078.bgt	N390060.bgt	N390067.bgt	S10E176.bgt
N390068.bgt	N390079.bgt	N390061.bgt	N390068.bgt	S10E177.bgt
N390069.bgt	N390080.bgt	N390062.bgt	N390069.bgt	S10E178.bgt
N390070.bgt	N390081.bgt	N390063.bgt	N390070.bgt	S10E179.bgt
N390071.bgt	N390082.bgt	N390064.bgt	N390071.bgt	S10E180.bgt
N390072.bgt	N390083.bgt	N390065.bgt	N390072.bgt	S10E181.bgt
N390073.bgt	N390084.bgt	N390066.bgt	N390073.bgt	S10E182.bgt
N390074.bgt	N390085.bgt	N390067.bgt	N390074.bgt	S10E183.bgt
N390075.bgt	N390086.bgt	N390068.bgt	N390075.bgt	S10E184.bgt
N390076.bgt	N390087.bgt	N390069.bgt	N390076.bgt	S10E185.bgt
N390077.bgt	N390088.bgt	N390070.bgt	N390077.bgt	S10E186.bgt
N390078.bgt	N390089.bgt	N390071.bgt	N390078.bgt	S10E187.bgt
N390079.bgt	N390090.bgt	N390072.bgt	N390079.bgt	S10E188.bgt
N390080.bgt	N390091.bgt	N390073.bgt	N390080.bgt	S10E189.bgt
N390081.bgt	N390092.bgt	N390074.bgt	N390081.bgt	S10E190.bgt
N390082.bgt	N390093.bgt	N390075.bgt	N390082.bgt	S10E191.bgt
N390083.bgt	N390094.bgt	N390076.bgt	N390083.bgt	S10E192.bgt
N390084.bgt	N390095.bgt	N390077.bgt	N390084.bgt	S10E193.bgt
N390085.bgt	N390096.bgt	N390078.bgt	N390085.bgt	S10E194.bgt
N390086.bgt	N390097.bgt	N390079.bgt	N390086.bgt	S10E195.bgt
N390087.bgt	N390098.bgt	N390080.bgt	N390087.bgt	S10E196.bgt
N390088.bgt	N390099.bgt	N390081.bgt	N390088.bgt	S10E197.bgt
N390089.bgt	N390100.bgt	N390082.bgt	N390089.bgt	S10E198.bgt
N390090.bgt	N390101.bgt	N390083.bgt	N390090.bgt	S10E199.bgt
N390091.bgt	N390102.bgt	N390084.bgt	N390091.bgt	S10E200.bgt
N390092.bgt	N390103.bgt	N390085.bgt	N390092.bgt	S10E201.bgt
N390093.bgt	N390104.bgt	N390086.bgt	N390093.bgt	S10E202.bgt
N390094.bgt	N390105.bgt	N390087.bgt	N390094.bgt	S10E203.bgt
N390095.bgt	N390106.bgt	N390088.bgt	N390095.bgt	S10E204.bgt
N390096.bgt	N390107.bgt	N390089.bgt	N390096.bgt	S10E205.bgt
N390097.bgt	N390108.bgt	N390090.bgt	N390097.bgt	S10E206.bgt
N390098.bgt	N390109.bgt	N390091.bgt	N390098.bgt	S10E207.bgt
N390099.bgt	N390110.bgt	N390092.bgt	N390099.bgt	S10E208.bgt
N390100.bgt	N390111.bgt	N390093.bgt	N390100.bgt	S10E209.bgt
N390101.bgt	N390112.bgt	N390094.bgt	N390101.bgt	S10E210.bgt
N390102.bgt	N390113.bgt	N390095.bgt	N390102.bgt	S10E211.bgt
N390103.bgt	N390114.bgt	N390096.bgt	N390103.bgt	S10E212.bgt
N390104.bgt	N390115.bgt	N390097.bgt	N390104.bgt	S10E213.bgt
N390105.bgt	N390116.bgt	N390098.bgt	N390105.bgt	S10E214.bgt
N390106.bgt	N390117.bgt	N390099.bgt	N390106.bgt	S10E215.bgt
N390107.bgt		N390100.bgt	N390107.bgt	S10E216.bgt
N390108.bgt		N390101.bgt	N390108.bgt	S10E217.bgt

Récupération des tuiles dans un datalake



Création d'un raster virtuel sous GDAL



* S1Tilling : Script de Thierry Koleček (CNES - CESBIO)

Script SAR_MultiSlc activation



```
python SAR_MultiSlc.py SAR_MultiSlc.json -d 20160607 -f 20160930 -m s1a-s6-slc-  
vv-20160817t145252-20160817t145315-012641-013d72-002.tiff -o yes
```

```
python SAR_MultiSlc.py SAR_MultiSlc.json -d 20160607 -f 20160930 -m s1a-s4-slc-  
vv-20160818t014650-20160818t014715-012648-013db1-002.tiff -o yes
```

```
python QuickMlView.py -in s1a-s6-slc-vv-20160817t145252-20160817t145315-012641-013d72-002.tiff -ml 3 3
```

```
python QuickMlView.py -in s1a-s4-slc-vv-20160818t014650-20160818t014715-012648-013db1-002.tiff -ml 3 3
```