1. Exploring the test data set

The primary requirement for analyzing interferogram stacks with GIAnT, is that all the input data should be coregistered on a common grid. We could design our interferogram processing workflows to generated coregistered output products in radar coordinates (i.e, along-track, slant range space) or in geo-coordinates (i.e, latitude, longitude space). For this exercise, we will consider actual COSMO SkyMed data acquired over Kilauea volcano in Hawaii, and processed to a common geocoded output grid using ISCE with the approach outlined in Lab 8. Let us peruse the contents of the directory containing the test dataset, to try and understand the inputs needed by GIAnT.

We'll start in this directory:

```
> cd /home/ubuntu/data/giant/kilaeau
```
(sorry for typo in directory name)

```
> ls
GIAnT insar prepGIAnT.py
```

"insar" directory contains all the interferograms processed using the approach described in Lab 8. We will proceed with our time-series analysis in "GIAnT" directory. "prepGIAnT.py" is a simple python script that was used to set up a GIAnT analysis run using outputs from ISCE. We will now describe the contents of the "insar" and "GIAnT" directories in detail.

2. “insar” - directory with unwrapped phase and coherence files

In the “insar” directory, we have compiled all the required interferograms and coherence files as float32 binary files. Information from each pair is stored in an individual subdirectory.

```
> cd insar
> ls
20101228_20101220  20110319_20101220  20110403_20110326
20110129_20101228  20110319_20110222  20110404_20101212
20110214_20101212  20110319_20110318  20110404_20110214
20110214_20110206  20110322_20101220  20110404_20110302
20110222_20101220  20110322_20110129  20110404_20110303
…
```

Each individual interferogram sub-directory contains the geocoded unwrapped phase and coherence files. Remember that in lab 8, we geocoded all the products to the same output grid using the “geocode bounding box” tag in the input XML files.

```
> cd 20101228_20101220
> ls
filt_topophase.unw.geo      insarProc.xml      topophase.cor.geo.xml
filt_topophase.unw.geo.xml  topophase.cor.geo
```

GIAnT needs the following conditions to be satisfied:

- All the unwrapped phase files should be of the same size. All the coherence files should be of the same size.
- The top left pixel of each file lines up exactly.
- A consistent format, i.e, data type, interleaving scheme, should be followed for unwrapped phase and coherence files.

You can view the contents of each of these files using mdx.py. NOTE: you will need to run mdx.py from the graphical desktop.

```
> cd insar/20101228_20101220
> mdx.py filt_topophase.unw.geo topophase.cor.geo
```

3. “GIAnT” - directory for time-series analysis

We will now proceed to the GIAnT analysis directory.

```
> cd ../../GIAnT
> ls
example.xml  ifg.list  prepdataxml.py  prepsbasxml.py  userfn.py
```

This lab will focus on describing the two files: “ifg.list” and “example.xml” . The python scripts are used in the actual time-series analysis and will be explained in detail in Lab 9.1

Note:

The XML file format used by GIAnT is different from that used by ISCE. Maybe in the future, we will converge on a common format.

4. ifg.list - Interferogram network description

"ifg.list" is a four column text file that describes our interferogram network in a simple fashion.

```
> less ifg.list
20110310   20101212   -90.8372435038   CSK
20110322   20101220   -53.7819100954   CSK
20110407   20110326   183.599267061   CSK
20101228   20101220   -259.347791029   CSK
20110322   20110318   -112.975218835   CSK
20110407   20110303   156.871731578   CSK
20110318   20101220   59.2300242801   CSK
...
```

Each line of ifg.list is of the following form:
Masterdate Slavedate Bperp_in_m Sensor

- Master and slave dates can be provided in yyyymmdd or yymmdd format. We used the yyyymmdd format for this example.
- Perpendicular baseline is provided in meters. This can be used to generate the time-baseline plots. The baseline values can be found in insarProc.xml files generated by ISCE.
- Sensor information is used if users want to build stacks that combines information from multiple sensors. This is particularly useful if your dataset includes acquisitions from multiple sensors on the same date.
- GIAnT does not insist on a particular temporal ordering of the master-slave acquisition in an interferogram. GIAnT can decode the order and build time-series design matrices accordingly. However it would be easier for the users if they stuck to one consistent order - earliest-latest or latest-earliest.

5. “example.xml” - common metadata

ISCE generates a detailed progress log file (insarProc.xml) that contains information related to each of the processing steps. Since we use a stack of coregistered interferograms, any one resource file that contains information regarding the dimensions of the dataset is sufficient to describe the entire dataset.

Let’s look at our example.xml file:

```
> less example.xml
<insarProc>
    <insarProc>
        <ISCE_VERSION>Release: 2.0.0, svn-1541, 20140724. Current
svn-exported</ISCE_VERSION>
    </insarProc>
    <planet>
        <GM>398600448073000.0</GM>
        <NAME>Earth</NAME>
        <SPINRATE>7.292106590880652e-05</SPINRATE>
        <ellipsoid>
            <SEMIMAJOR_AXIS>6378137.0</SEMIMAJOR_AXIS>
         <ECCENTRICITY_SQUARED>0.0066943799901</ECCENTRICITY_SQUARED>
            <MODEL>WGS-84</MODEL>
        </ellipsoid>
    </planet>
```

Note that only some of these fields, that are common to the stack of interferograms, are used in the processing. In particular, the following fields are gathered:

Parameter	**Source in example.xml**
Width of image	runGeocode.outputs.GEO_WIDTH
Length of image	runGeocode.outputs.GEO_LENGTH
Wavelength	runTopo.inputs.RADAR_WAVELENGTH
UTC	master.frame.SENSING_START
Heading	runTopo.inputs.PEG_HEADING

We have now gathered all the information associated with the interferogram stack and can proceed to the time-series analysis labs.