Introduction to CASA

Aida Ahmadi
ALMA Local Expertise Group (Allegro)

ALMA Data Reduction Training Day
Leiden Observatory
November 27, 2023

Common Astronomy Software Applications

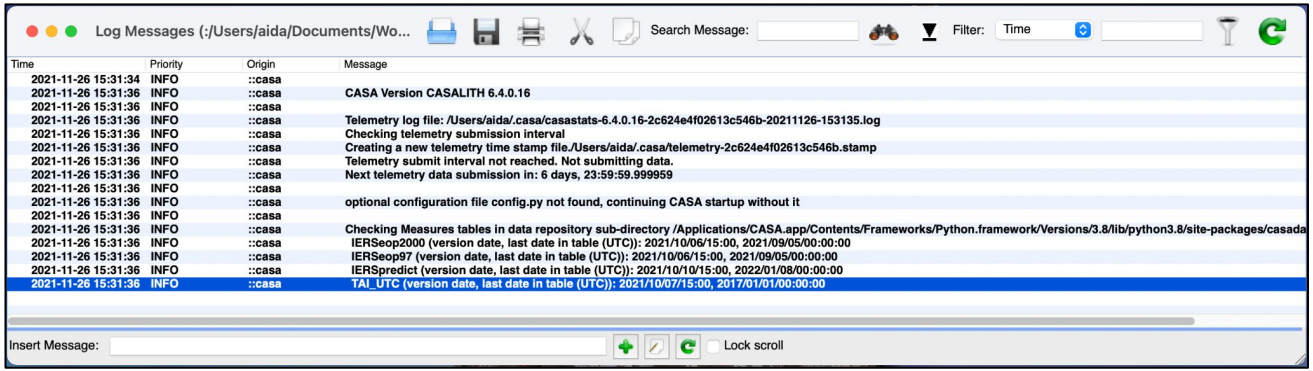
- Can process data from both **single-dish** and **aperture-synthesis telescopes**
- Primary data processing (calibration & imaging) software for **ALMA** & **VLA**
- Python based
  - Versions < 6 use Python 2.7
  - Newer versions use Python 3
- Available for Linux (RedHat) and Mac OS

➔ **Website** – https://casa.nrao.edu/
➔ **Guides** – https://casaguides.nrao.edu/
➔ **Documentation**
  - Versions 6.1 & earlier: https://casa.nrao.edu/casadocs
  - Versions 6.2 & later: https://casadocs.readthedocs.io/en/stable/

Starting CASA

After installation, to open CASA simply type `casa` in the terminal if you set up an alias. Otherwise type the full path.

➔ Starting CASA will open a logger (and a log file):



`casa --nologger`

if you do not need the logger `GUI`

➔ And the terminal prompt:

```
optional configuration file config.py not found, continuing CASA startup without it

IPython 7.15.0 -- An enhanced Interactive Python.

Using matplotlib backend: MacOSX
Telemetry initialized. Telemetry will send anonymized usage statistics to NRAO.
You can disable telemetry by adding the following line to the config.py file in your rcdir (e.g. ~/.casa/config.py):
telemetry_enabled = False
--> CrashReporter initialized.
CASA 6.4.0.16 -- Common Astronomy Software Applications [6.4.0.16]

CASA <1>:
```

CASA Basics

CASA Tasks

Tasks are executed to perform a single job (e.g. loading, plotting, flagging, calibrating)
Each task contains a set of user-defined parameters

List of available tasks

`taskhelp` -> A more exhaustive list of tasks with descriptions
`tasklist()` -> Get an overview of available tasks, organized by category **(removed in CASA 6)**
+ More information about the tasks:
https://casadocs.readthedocs.io/en/stable/api/casatasks.html

Getting help on a task

`inp <taskname>` to get an overview of a given task and its input parameters

`help <taskname>` to get a detailed description of a given task and its input parameters (use arrow keys to continue, press q to exit)

Based on: https://casaguides.nrao.edu/index.php?title=Getting_Started_in_CASA

CASA Basics

Executing a task

Interactively:

`tget <taskname>` -> get the task and its previously set parameters
`inp` -> determine the input parameters needed for the task that was set
(set individual parameters using a Python `<parameter>=<value>` syntax)
`go` -> run the task

You may also do:

`default(<taskname>)` -> to set the parameters of a task to their default values
set individual parameters using a Python `<parameter>=<value>` syntax

***Note**: you can also simply set parameters without the `default` or `tget` steps but beware that you would be setting parameters globally!*

Programmatically:

```
taskname(parameter1='', parameter2='', ...)
```

Based on: https://casaguides.nrao.edu/index.php?title=Getting_Started_in_CASA

CASA Basics

Parameters

grey: parameter has sub-parameters
green: sub-parameters
red: invalid value
blue: parameter altered from its default

```
CASA <39>: inp tclean
# tclean -- Radio Interferometric Image Reconstruction
vis                 = 'data/sis14_twhya_calibrated_flagged.ms'
                                        # Name of input visibility file(s)
selectdata          = True              # Enable data selection parameters
   field            = ''                # field(s) to select
   spw              = ''                # spw(s)/channels to select
   timerange        = ''                # Range of time to select from data
   uvrange          = ''                # Select data within uvrange
   antenna          = ''                # Select data based on antenna/baseline
   scan             = ''                # Scan number range
   observation      = ''                # Observation ID range
   intent           = ''                # Scan Intent(s)
datacolumn          = ''                # Data column to image(data,corrected)
imagename           = ''                # Pre-name of output images
imsize              = [100]             # Number of pixels
cell                = []                # Cell size
phasecenter         = ''                # Phase center of the image
stokes              = 'I'               # Stokes Planes to make
projection          = 'SIN'             # Coordinate projection
startmodel          = ''                # Name of starting model image
specmode            = 'mfs'             # Spectral definition mode (mfs,cube,cubedata, cubesource)
   reffreq          = ''                # Reference frequency
gridder             = 'standard'        # Gridding options (standard, wproject, widefield, mosaic, awproject)
   vptable          = ''                # Name of Voltage Pattern table
   pblimit          = 0.2               # PB gain level at which to cut off normalizations
deconvolver         = 'hogbom'          # Minor cycle algorithm (hogbom,clark,multiscale,mtmfs,mem,clarkstokes)
restoration         = True              # Do restoration steps (or not)
   restoringbeam    = []                # Restoring beam shape to use. Default is the PSF main lobe
   pbcor            = False             # Apply PB correction on the output restored image
outlierfile         = ''                # Name of outlier-field image definitions
weighting           = 'nat'             # Weighting scheme (natural,uniform,briggs, briggsabs[experimental], briggsbwtaper[experimental])
niter               = 0                 # Maximum number of iterations
usemask             = 'user'            # Type of mask(s) for deconvolution: user, pb, or auto-multithresh
   mask             = ''                # Mask (a list of image name(s) or region file(s) or region string(s) )
   pbmask           = 0.0               # primary beam mask
fastnoise           = True              # True: use the faster (old) noise calculation. False: use the new improved noise calculations
restart             = True              # True : Re-use existing images. False : Increment imagename
savemodel           = 'none'            # Options to save model visibilities (none, virtual, modelcolumn)
calcres             = True              # Calculate initial residual image
calcpsf             = True              # Calculate PSF
   psfcutoff        = 0.35              # All pixels in the main lobe of the PSF above psfcutoff are used to fit a Gaussian beam (the Clean beam).
parallel            = False             # Run major cycles in parallel
```

Data selection syntax

spw='0:5~30;40~55,1:10~25;45~58,2'

Running scripts

In CASA: **`execfile('script_name.py')`**
In the terminal: **`casa -c script_name.py`**

CASA Data formats

- **Raw visibility (uv) data from ALMA** comes in **ALMA Science Data Model** (.asdm) format
- Once imported into CASA, it can be stored as a **Measurement Set** (.ms)

```
importasdm(asdm='rawdata.asdm', vis='visibilities.ms')
```

Measurement sets:

- Measurement sets (.ms files) are **directories** that contain many sub-directories and tables

```
[CASA <1>: ls sis14_twhya_calibrated_flagged.ms/
ANTENNA/          DATA_DESCRIPTION/ POINTING/         STATE/          table.f10*      table.f16*       table.f20*        table.f23*        table.f7*
ASDM_ANTENNA/     FEED/             POLARIZATION/     SYSCAL/         table.f11*      table.f17*       table.f20_TSM0*   table.f23_TSM1*   table.f8*
ASDM_CALWVR/      FIELD/            PROCESSOR/        SYSPOWER/       table.f12*      table.f17_TSM1*  table.f21*        table.f3*         table.f9*
ASDM_RECEIVER/    FLAG_CMD/         SORTED_TABLE/     WEATHER/        table.f13*      table.f18*       table.f21_TSM1*   table.f4*         table.info*
ASDM_STATION/     HISTORY/          SOURCE/           table.dat*      table.f14*      table.f19*       table.f22*        table.f5*         table.lock*
CALDEVICE/        OBSERVATION/      SPECTRAL_WINDOW/  table.f1*       table.f15*      table.f2*        table.f22_TSM1*   table.f6*
```

- Images produced by CASA (.image, .residual, etc.) are also stored in the same format
- You can use the **`exportfits`** task to convert CASA images to FITS format

CASA Data formats

Measurement sets:

```
CASA <1>: ls sis14_twhya_calibrated_flagged.ms/
ANTENNA/          DATA_DESCRIPTION/ POINTING/         STATE/        table.f10*  table.f16*  table.f20*       table.f23*      table.f7*
ASDM_ANTENNA/     FEED/             POLARIZATION/     SYSCAL/       table.f11*  table.f17*  table.f20_TSM0*  table.f23_TSM1* table.f8*
ASDM_CALWVR/      FIELD/            PROCESSOR/        SYSPOWER/     table.f12*  table.f17_TSM1* table.f21*   table.f3*       table.f9*
ASDM_RECEIVER/    FLAG_CMD/         SORTED_TABLE/     WEATHER/      table.f13*  table.f18*  table.f21_TSM1*  table.f4*       table.info*
ASDM_STATION/     HISTORY/          SOURCE/           table.dat*    table.f14*  table.f19*  table.f22*       table.f5*       table.lock*
CALDEVICE/        OBSERVATION/      SPECTRAL_WINDOW/  table.f1*    table.f15*  table.f2*   table.f22_TSM1*  table.f6*
```

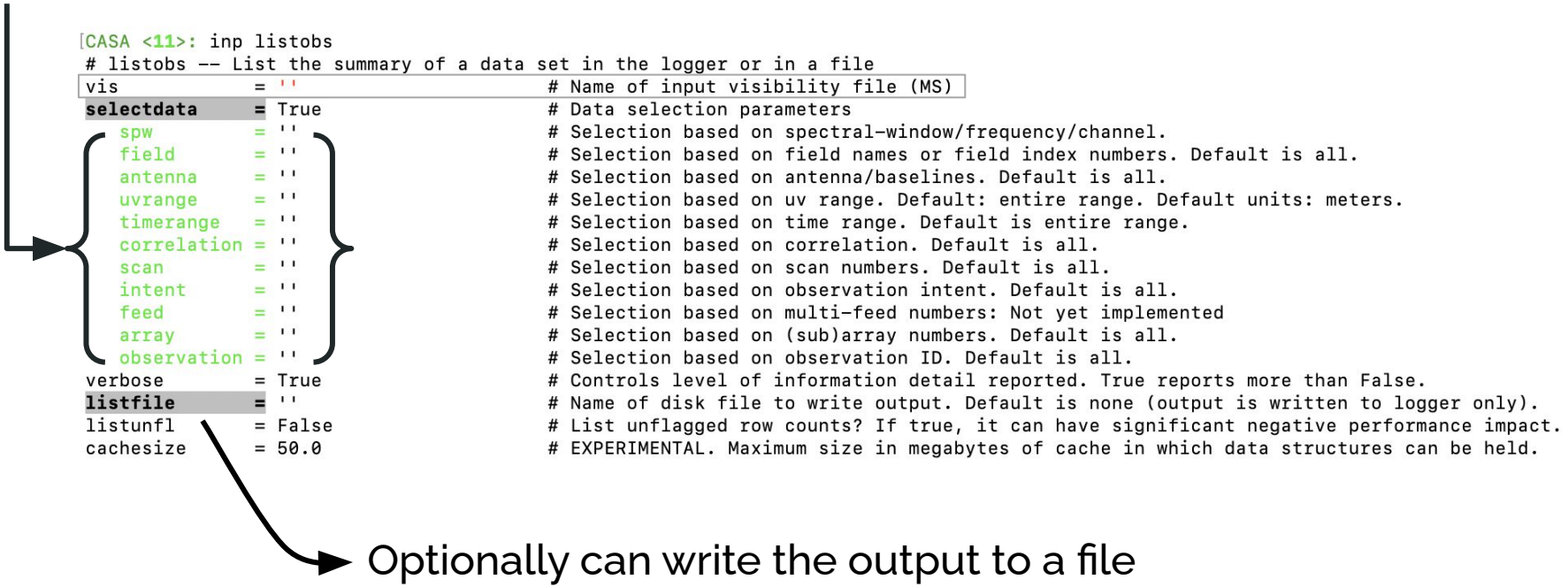
- To copy or remove them in the terminal you need to use the recursive option
  - ➢ `cp -r this_data.ms that_data.ms`
- Safest way to remove a dataset in CASA:
  - ➢ `rmtables('this_data.ms')`
    - Alternatively:
      - ➢ `rm -rf this_data.ms`
    - Or if within a script:
      - ➢ `os.system('rm -r this_data.ms')`

Data Inspection with CASA

listobs	list the contents of measurement set
plotants	plot the location of antennas
plotms	inspect/flag visibilities interactively
imview	view/inspect images interactively

listobs: lists the contents of measurement set

Can select a subset of the measurement set



listobs: lists the contents of measurement set

Example > listobs(vis='sis14_twhya_calibrated_flagged.ms')

sequence of observations

```
##########################################
##### Begin Task: listobs          #####
listobs( vis='sis14_twhya_calibrated_flagged.ms', selectdata=True, spw='', field='', antenna='', uvrange='', timerange='', correlation='', scan='', intent='', feed='', array='', observation='', verbose=
================================================================================
           MeasurementSet Name:  /Users/aida/Documents/Work/Leiden/Allegro/Events/202111_Data_Reduction_Day/data/sis14_twhya_calibrated_flagged.ms      MS Version 2
================================================================================
   Observer: cqi     Project: uid://A002/X327408/X6f
Observation: ALMA
Computing scan and subscan properties...
Data records: 80563       Total elapsed time = 5647.68 seconds
   Observed from   19-Nov-2012/07:36:57.0   to   19-Nov-2012/09:11:04.7 (UTC)

   ObservationID = 0         ArrayID = 0
  Date        Timerange (UTC)          Scan  FldId FieldName             nRows     SpwIds   Average Interval(s)    ScanIntent
 19-Nov-2012/07:36:57.0 - 07:39:13.1     4      0 J0522-364               4200  [0]  [6.05] [CALIBRATE_BANDPASS#ON_SOURCE,CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
          07:44:45.2 - 07:47:01.2     7      2 Ceres                   3800  [0]  [6.05] [CALIBRATE_AMPLI#ON_SOURCE,CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
          07:52:42.0 - 07:53:47.6    10      3 J1037-295               1900  [0]  [6.05] [CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
          07:56:23.5 - 08:02:11.3    12      5 TW Hya                  8514  [0]  [6.05] [OBSERVE_TARGET#ON_SOURCE]
          08:04:36.3 - 08:05:41.9    14      3 J1037-295               1900  [0]  [6.05] [CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
          08:08:09.6 - 08:13:57.3    16      5 TW Hya                 10360  [0]  [6.05] [OBSERVE_TARGET#ON_SOURCE]
          08:16:20.6 - 08:17:26.2    18      3 J1037-295               2100  [0]  [6.05] [CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
          08:19:53.9 - 08:25:41.7    20      5 TW Hya                 10321  [0]  [6.05] [OBSERVE_TARGET#ON_SOURCE]
          08:28:17.1 - 08:29:22.6    22      3 J1037-295               2100  [0]  [6.05] [CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
          08:32:00.5 - 08:37:48.2    24      5 TW Hya                 10324  [0]  [6.05] [OBSERVE_TARGET#ON_SOURCE]
          08:40:11.9 - 08:41:17.4    26      3 J1037-295               2100  [0]  [6.05] [CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
          08:43:45.6 - 08:49:33.4    28      5 TW Hya                  9462  [0]  [6.05] [OBSERVE_TARGET#ON_SOURCE]
          08:51:57.1 - 08:53:02.6    30      3 J1037-295               1900  [0]  [6.05] [CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
          08:58:12.0 - 09:00:28.1    33      6 3c279                   3402  [0]  [6.05] [CALIBRATE_BANDPASS#ON_SOURCE,CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
          09:01:35.7 - 09:02:41.2    34      3 J1037-295               1900  [0]  [6.05] [CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
          09:05:15.6 - 09:07:31.6    36      5 TW Hya                  4180  [0]  [6.05] [OBSERVE_TARGET#ON_SOURCE]
          09:09:59.1 - 09:11:04.7    38      3 J1037-295               2100  [0]  [6.05] [CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
        (nRows = Total number of rows per scan)
```

listobs: lists the contents of measurement set

List of fields & spectral windows

Fields: 5

ID	Code	Name	RA	Decl	Epoch	SrcId	nRows
0	none	J0522-364	05:22:57.984648	-36.27.30.85128	J2000	0	4200
2	none	Ceres	06:10:15.950590	+23.22.06.90668	J2000	2	3800
3	none	J1037-295	10:37:16.079736	-29.34.02.81316	J2000	3	16000
5	none	TW Hya	11:01:51.796000	-34.42.17.36600	J2000	4	53161
6	none	3c279	12:56:11.166576	-05.47.21.52464	J2000	5	3402

Spectral Windows: (1 unique spectral windows and 1 unique polarization setups)

SpwID	Name	#Chans	Frame	Ch0(MHz)	ChanWid(kHz)	TotBW(kHz)	CtrFreq(MHz)	BBC Num	Corrs
0	ALMA_RB_07#BB_2#SW-01#FULL_RES	384	TOPO	372533.086	610.352	234375.0	372649.9688	2	XX YY

Sources: 5

ID	Name	SpwId	RestFreq(MHz)	SysVel(km/s)
0	J0522-364	0	-	-
1	Ceres	0	-	-
2	J1037-295	0	-	-
3	TW Hya	0	-	-
4	3c279	0	-	-

Antenna names & positions

Antennas: 21:

ID	Name	Station	Diam.	Long.	Lat.	Offset from array center (m) East	Offset from array center (m) North	Offset from array center (m) Elevation	ITRF Geocentric coordinates (m) x	ITRF Geocentric coordinates (m) y	ITRF Geocentric coordinates (m) z
1	DA42	A050	12.0 m	-067.45.16.2	-22.53.29.3	43.0352	-744.9713	21.6702	2225079.880016	-5440041.377534	-2481724.598031
2	DA44	A068	12.0 m	-067.45.20.6	-22.53.25.7	-82.4232	-631.7828	23.5810	2224981.097784	-5440131.250387	-2481621.066374
3	DA45	A070	12.0 m	-067.45.11.9	-22.53.29.3	166.1833	-743.4934	19.8811	2225193.450167	-5439993.764157	-2481722.540534
4	DA46	A067	12.0 m	-067.45.12.7	-22.53.27.2	142.4097	-678.7318	20.1280	2225181.070532	-5440026.290790	-2481662.975103
5	DA48	A046	12.0 m	-067.45.17.0	-22.53.29.3	21.4267	-742.7987	21.6757	2225060.202580	-5440050.344436	-2481722.598651
6	DA49	A029	12.0 m	-067.45.18.2	-22.53.25.8	-12.9134	-636.4552	22.1350	2225044.239583	-5440102.022535	-2481624.808405
7	DA50	A045	12.0 m	-067.45.17.9	-22.53.30.1	-5.4183	-767.4398	22.6034	2225032.051652	-5440052.426015	-2481745.660003
9	DV02	A077	12.0 m	-067.45.10.1	-22.53.25.9	217.6299	-637.5333	15.8376	2225255.259272	-5440008.987869	-2481623.352052
11	DV05	A082	12.0 m	-067.45.08.3	-22.53.29.2	269.0433	-740.9521	15.7832	2225287.593766	-5439952.243679	-2481718.605314
12	DV06	A037	12.0 m	-067.45.17.5	-22.53.28.8	6.7403	-727.3003	21.2086	2225048.729287	-5440061.085777	-2481708.139136
14	DV08	A021	12.0 m	-067.45.17.2	-22.53.27.0	14.3196	-672.8108	21.3420	2225063.814715	-5440077.948261	-2481657.992572
15	DV10	A071	12.0 m	-067.45.19.9	-22.53.23.5	-60.7887	-563.2541	23.3799	2225011.141945	-5440147.560932	-2481557.855663
16	DV13	A072	12.0 m	-067.45.12.6	-22.53.24.0	147.1742	-580.5887	18.1825	2225199.254375	-5440058.161494	-2481571.803699
17	DV15	A074	12.0 m	-067.45.12.1	-22.53.32.0	161.8159	-828.6196	18.7688	2225176.483514	-5439963.820451	-2481800.529842
18	DV16	A069	12.0 m	-067.45.21.3	-22.53.30.2	-101.4797	-770.1047	23.2972	2224942.993176	-5440088.421459	-2481748.384855
19	DV17	A138	12.0 m	-067.45.17.1	-22.53.34.4	19.1461	-901.2603	26.0137	2225036.269025	-5439997.853009	-2481870.267607
20	DV18	A053	12.0 m	-067.45.17.3	-22.53.31.2	12.5939	-802.9941	21.5281	2225043.111690	-5440031.889497	-2481777.995870
21	DV19	A008	12.0 m	-067.45.15.4	-22.53.26.8	67.5592	-667.6872	20.9574	2225113.709955	-5440059.310545	-2481653.122797
22	DV20	A020	12.0 m	-067.45.17.8	-22.53.28.0	-2.9649	-703.4389	21.6629	2225043.419055	-5440073.737929	-2481686.333574
24	DV22	A011	12.0 m	-067.45.14.4	-22.53.28.4	95.9131	-716.5005	21.0898	2225132.810230	-5440031.115405	-2481698.143589
25	DV23	A007	12.0 m	-067.45.15.1	-22.53.27.3	74.0152	-681.2926	21.3231	2225117.809276	-5440052.280005	-2481665.799049

Task listobs complete. Start time: 2021-11-29 22:42:19.113889 End time: 2021-11-29 22:42:19.209607

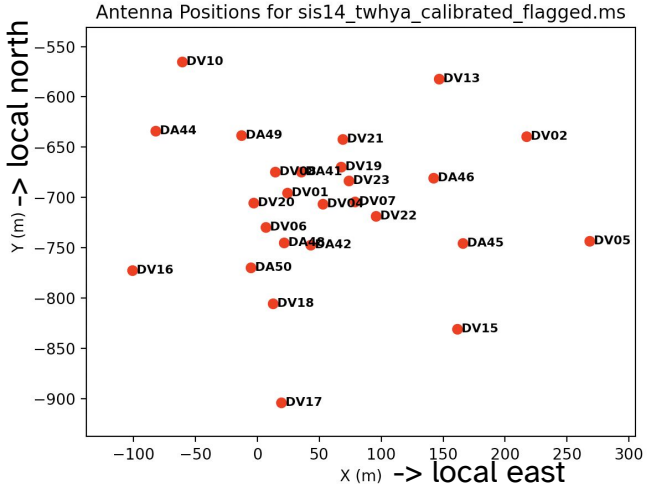
End Task: listobs

##

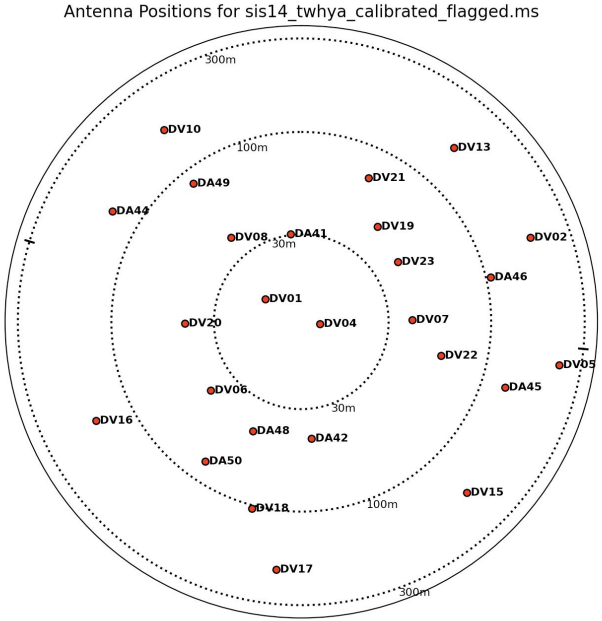
##

plotants: plot the location of antennas

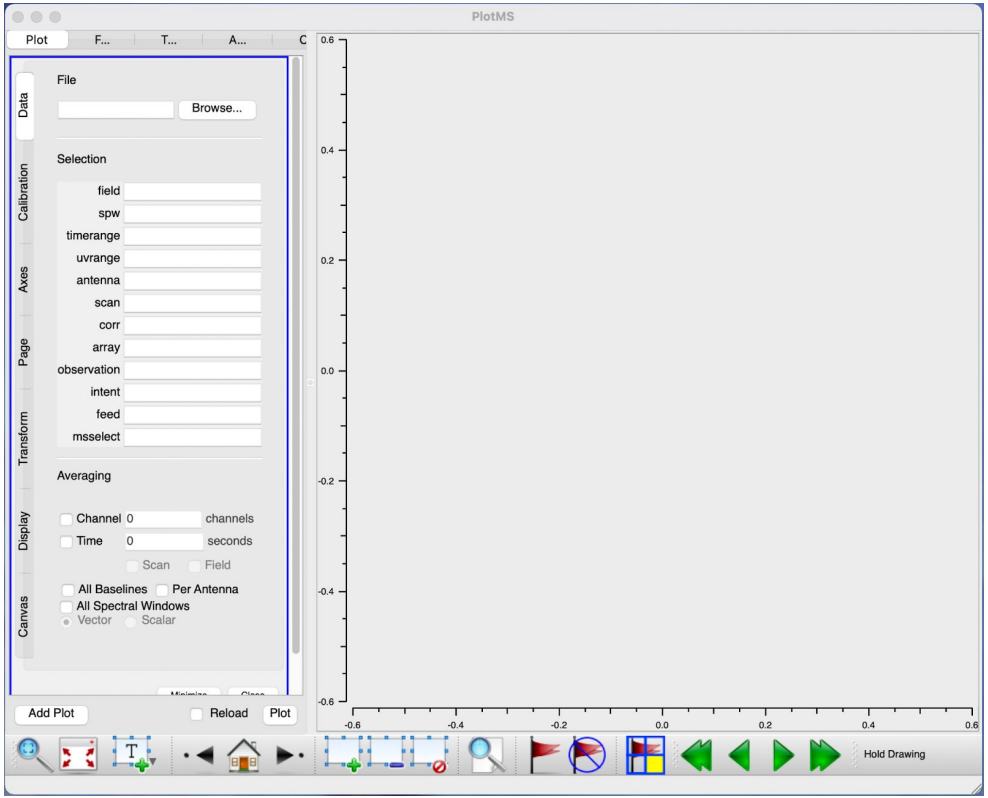
Example > plotants(vis='sis14_twhya_calibrated_flagged.ms', showgui=True, logpos=True)



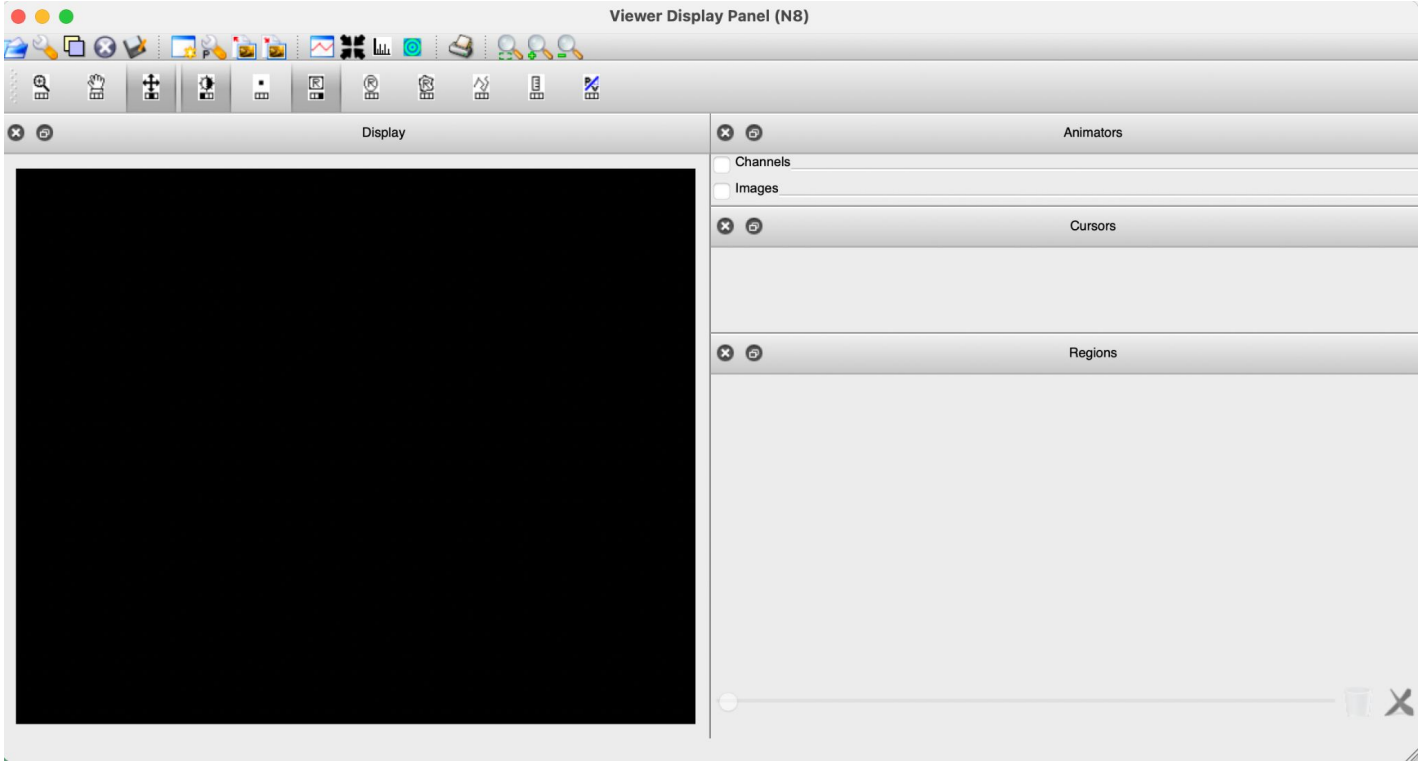
Plotting logarithmic positions helps display the center of the array



plotms: inspect/flag data interactively



imview: view/inspect images interactively



Thanks!

Questions?

Getting started on the Allegro computers

1. Go to the analysis folder in the project directory

```
> cd
/allegro1/allegro/home/your_username/open_ALMA_DRT2023/analysis/your_username
```

2. Make two folders

```
> mkdir imaging
> mkdir analysis_tools
```

3. Copy data from the 'archive' folder to your own folder

```
> cp -r
../../archive/DRT2023/TW_hydra/sis14_twhya_calibrated_flagged.ms.contsub
imaging/.
> cp -r ../../archive/DRT2023/TW_hydra/twhya_n2hp.image analysis_tools/.
> cp -r ../../archive/DRT2023/TW_hydra/sis14_twhya_cont.image analysis_tools/.
> cp -r ../../archive/DRT2023/TW_hydra/*.fits analysis_tools/.
```

4. Copy scripts from the 'scripts' folder to your own folder

```
> cp ../../scripts/Imaging*.py imaging/.
> cp ../../scripts/analysis*.py analysis_tools/.
```

5. Go to the `imaging` folder and open CASA

```
> cd imaging
> nice +10 env -u PYTHONPATH -u LD_LIBRARY_PATH casapy-660
```