Introduction to CASA

Elisabetta Liuzzo
INAF- Istituto di Radioastronomia
Italian node of European ALMA Regional Centre
Credits to Kazi Rygl

UniTo – April 2018
The **ALMA** measurement set

CASA is a set of C++ application libraries running through a python interface. The software has been designed to calibrate, edit, image, and analyze interferometric and single dish data, in particular the ALMA and JVLA data.

The native uvdata format is the measurement set (*.ms).

A measurement set with a large number of tables, containing the observed visibilities, various calibration tables, flags and models.
The ALMA measurement set

Organized in a MAIN table with four columns:

Data  -  Model  -  Corrected data  -  Flags

The original visibilities are never overwritten when calibration or flags are applied, but copied with applied correction/edit to corrected data column. The ms structure stays always the same.

<table>
<thead>
<tr>
<th>MAIN</th>
<th>Model, e.g.:</th>
<th>Corrected data</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original visibility data</td>
<td>FT of image made from MS</td>
<td>Copy of visibilities with calibration tables applied</td>
<td>(Edits are stored here first; backup tables can be made and used to modify)</td>
</tr>
<tr>
<td></td>
<td>FT of supplied model image</td>
<td>(Used in imaging but not calibration)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FT of calibrator flux density</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Starting Casa

• Start Casa by:

  $ casapy  

     for versions < 4.7.0

  $ casa  

     for versions > 4.7.0

When starting Casa, a logger window opens up - in which the history and the output of the tasks is written.

When exiting casa the logger content gets written into casapy-*.log and all the commands given will finish in ipython*.log
Most common commands

to obtain the list of tasks

$> \text{tasklist}

to see the input of a task

$> \text{inp listobs}

launch task

$> \text{go listobs}

to run a file

$> \text{execfile('name\_of\_file')}

to obtain the previously used parameters

$> \text{tget concat}

to obtain the task in default setting

$> \text{default('split')}

to see the summary info of a task and about its adverbs

$> \text{help flagdata}

open help menu with $> \text{help}$ to get info on modules (casa tasks), keywords (python keywords, such as if, from and while) and topics (python objects such as floats, classes)


Casa tasks

Inp flagdata shows the arguments of the task

blue - not a default value, needs to be set
black - default value
red - erroneous value (here, missing "")
green - sub parameters

```
CASA <37>: inp
-------------> inp()
# flagdata :: All-purpose flagging task based on data-selections and flagging modes/algorithms.
vis = 'uid_A602_Xa9b40d_X3cb8.ms.split.cal' # Name of MS file or calibration table to flag
mode = 'manual' # Flagging mode
  field = 18 # Field names or field index numbers: '' ==> all, field='0-2,3C286'
  spw = 1 # Spectral-window/frequency/channel: '' ==> all, spw='0:17-19'
  antenna = '' # Antenna/baselines: '' ==> all, antenna = '3,VA04'
  timewindow = '' # Time range: '' ==> all,timewindow='09:14:0-09:54:0'
  correlation = '' # Correlation: '' ==> all, correlation='XX,YY'
  scan = 8 # Scan numbers: '' ==> all
  intent = '' # Observation intent: '' ==> all, intent='CAL*POINT*'
  array = '' # (Sub)array numbers: '' ==> all
  uvrange = '' # UV range: '' ==> all; uvrange = '0-100klambda', default units=meters
  observation = '' # Observation ID: '' ==> all
  feed = '' # Multi-feed numbers: Not yet implemented
  autocorr = False # Flag only the auto-correlations
action = 'apply' # Action to perform in MS and/or in inpfile (none/apply/calculate)
  display = '' # Display data and/or end-of-MS reports at runtime (data/report/both).
  flagbackup = True # Back up the state of flags before the run
savepars = False # Save the current parameters to the FLAG_CMD table or to a file
async = False # If true the taskname must be started using flagdata(...)

CASA <38>: 
CASA <39>: 
```
Data selection

field=‘Tita*’ - use * as a wildcard
field=‘2’  - use field_id as name

spw=""  - selects all spw
spw=‘1~3’  - selects spw 1,2,3 (~ gives a range)
spw=‘2:100~200,3’  - selects chan 100 to 200 from spw2, and spw3

antenna=‘DV01’ - selects all baselines with DV01
antenna =‘!DV01’ - selects all baselines except those with DV01

timerange = ’06:24:00~06:30:00’ - selects timerange
timerange = ‘<06:30:00’ - selects times before a given time
Visualization of visibilities and calibration tables: plotms

Important to check your calibration results, e.g. quality of data
visualization of images and image analysis: viewer

Useful for data analysis and statistics
The right Casa version

- On the Casa website there are many versions

- ALMA Archive data comes in raw format with a calibration script, which was created during quality assurance analysis of the dataset (ALMA data are delivered calibrated and with the proposed rms and resolution verified)

- To reduce your Archive data, you need to run the calibration script (scriptForPI.py) in the same version of Casa.

  $$> \text{execfile('scriptForPI.py')}$$

- If the data calibration was done by the pipeline (pipeline reduced data have a PPR*xml file in the script directory), then you need to download the pipeline version of CASA

  - start casa pipeline version:

    $$> \text{casapy --pipeline}$$
Casa online tutorials & documentation

Casaguides for ALMA: explain not only Casa commands but also ALMA data reduction (https://casaguides.nrao.edu/index.php/ALMAguides)