CASA Tutorial

Cameron Van Eck
2 April 2020

Resources

- VLA Data reduction workshop 2019: https://science.nrao.edu/science/meetings/2019/vla-data-reduction/program
- CASA Documentation: https://casa.nrao.edu/casadocs
- CASA Guides: https://casaguides.nrao.edu/index.php/Main_Page
- CASA cookbook: https://casa.nrao.edu/casa_cookbook.pdf

Introduction

- *Common Astronomy Software Applications* (CASA), the successor to the *Astronomical Image Processing System* (AIPS), is the most common software package for reducing interferometry data
- Provides a unified framework for all tasks, including Python 2 environment and scripting functionality.
- Supports old fashioned AIPS-like task interface and interactivity, or Python scripting capability. Jupyter notebooks not supported (but soon!).

Fun fact: AIPS was built in Fortran, designed to be run on main-frames using tape readers to get data. Did not really fit well with era of personal computers. All tasks limited to 5 letter names, all variables 8 letters.

Introduction

- Standard interface is an iPython command-line, which supports all the usual commands: Python code (import, if-then-else, etc.), % commands (%run, %paste), system commands (cd, ls)
- Visibility data usually stored in *Measurement Sets* (MSs), which store many tables (visibilities, antenna information, pointings, source information, weather, etc)
- Images are stored in proprietary CASA format, but trivially converted to FITS.

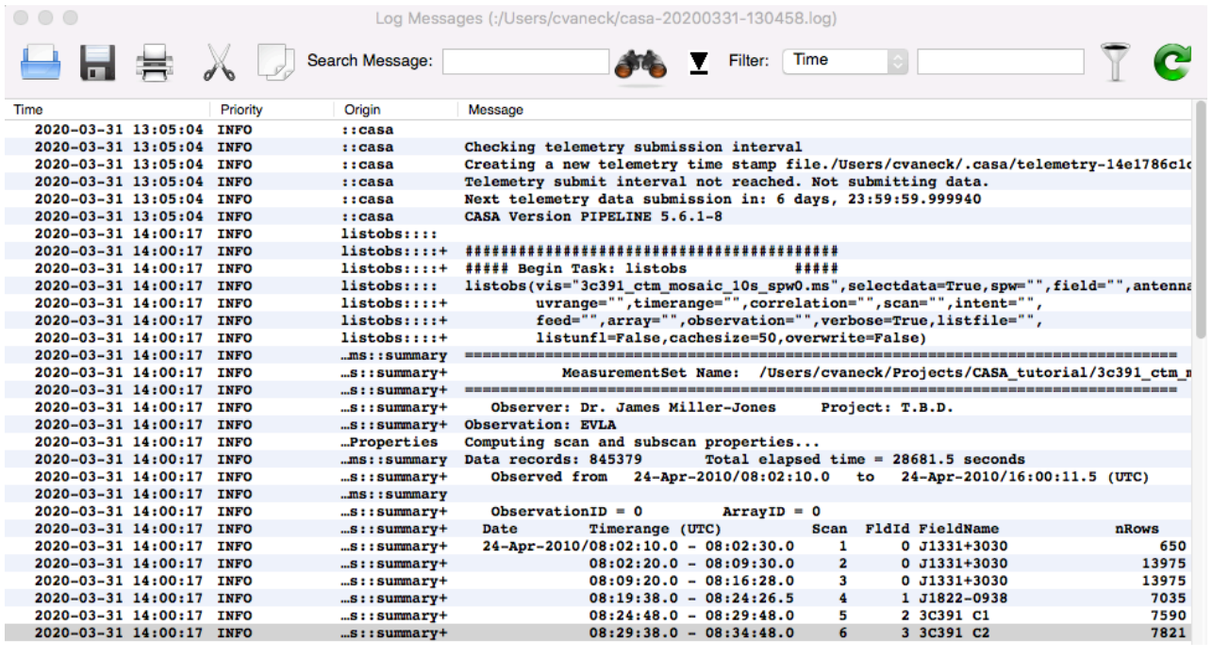
Quick Tips

- MSs store visibilities with 3 columns: DATA, CORRECTED_DATA, and MODEL. Keep track of what is in each column, and when tasks are modifying the columns! It's easy to accidentally overwrite something by accident. Keep a spare copy of the data when possible

- Use interactive/task mode to inspect the data, learn the software, and test different things. Along the way, make a script to replicate everything you've done.

Never believe that you're only going to do something once. Inevitably, things will need to be re-run, and you'll be glad you made a script with all the parameters set, so you don't need to remember what parameters you used.

Logger



The logger is where output is printed. Keep an eye on it every time you run a task! Also, save the log before quitting, because it can be very useful.

Common tasks

- listobs: list observation details
- plotms: visualize visibility data
- flagdata: flag out visibilities
- split/concat: divide/combine measurement sets
- gaincal: solve for Jones matrix parameters (gain and phase)
- applycal: apply calibration tables to visibilities
- tclean: Produce image from visibilities, and Clean
- imview/viewer: look at images
- exportfits: converts image to FITS

Syntax

- Use ‘tasklist’ to see a list of CASA tasks.
  ‘taskhelp’ gives short summaries of each task
  ‘doc’ or ‘doc(‘task’)’ opens CASADocs in browser.
  ‘<task>?’ opens task help

- Task mode:
  ‘inp <task>’ to see list of parameters
  ‘<parameter> = <value>’ to set parameters
  ‘go’ to start task
  ‘tget <task>’ to recover parameters from last run, ‘tput’ to save parameters, ‘default <task>’ to reset.

- Python scripting mode: call like a function.
  <task>(parm1=1, parm2=2, parm3=3,…)

Parameter syntax

- Tilde for ranges: spw='0~3'
- Comma for separate entries: spw='0~3,7,11'
- Colon for sub-ranges, semi-colon for separate entries (channels within spw): spw='7:0~40', spw='11:10;20;30'
- Asterisk for wildcards: field='3C*'
- Exclamation mark for NOT: antenna='!5'
- Ampersand for baselines: antenna='5&12'
- Less than/greater than for ranges: uvrange='<1000'

All of these parameters are strings, because they get passed into CASA's internal parameter parsing system.

Lab report
(for crediting students)

- Reduce 1 data set (can be tutorial or other), up to and including imaging. Maybe make a few measurements with the final image?

- Record what tasks were run, with what parameters. Make some short comments on the purpose of each task, the reason for choosing particular parameter values.

- Submit a lab report with this information and a few screenshots. This will be graded on understanding of the process and explanations for decisions made in the process. Don’t spend a lot of time on it!