MANCHESTER 1824

The University of Manchester

Introduction to CASA & the Measurement Set (MS)

Adam Avison

ALMA

EUROPEAN ARC

ALMA Regional Centre || UK

What is CASA?

- Common Astronomy Software Applications

- Does everything you need to take raw visibilities from a telescope and turn them into science ready data.

- Aims to support the current and next generation of radio telescopes (ALMA, JVLA, ngVLA and many more).

- Developed by a consortium; NRAO, ESO, NAOJ, CSIRO, and ASTRON.

Obtaining CASA

- CASA homepage: https://casa.nrao.edu/
- Download page: https://casa.nrao.edu/casa_obtaining.shtml
- CASA Documentation: https://casa.nrao.edu/casadocs/latest/

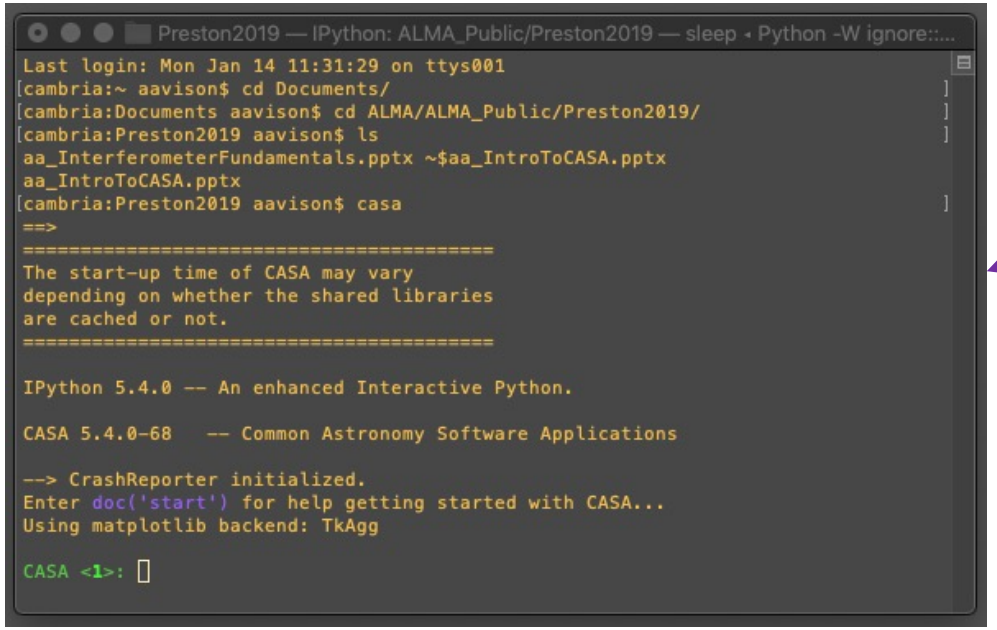
Getting started

- The latest version of CASA is 5.7*
- Versions from Linux (Red Hat only) and macOS.

* There is a separate release track under the version number 6.x but we won’t use these for this session.

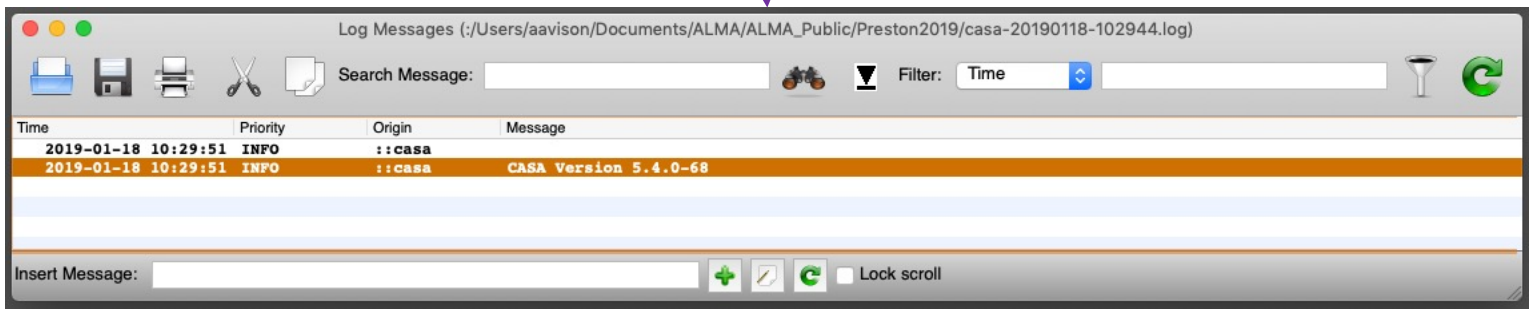
Working with CASA

- Once installed CASA can be started by typing 'casa' in a terminal.
- This will startup the iPython interface in the terminal and launch the Logger GUI



The iPython interface is where the work gets done

Logger will show you *lots* of useful message from the tasks being run.

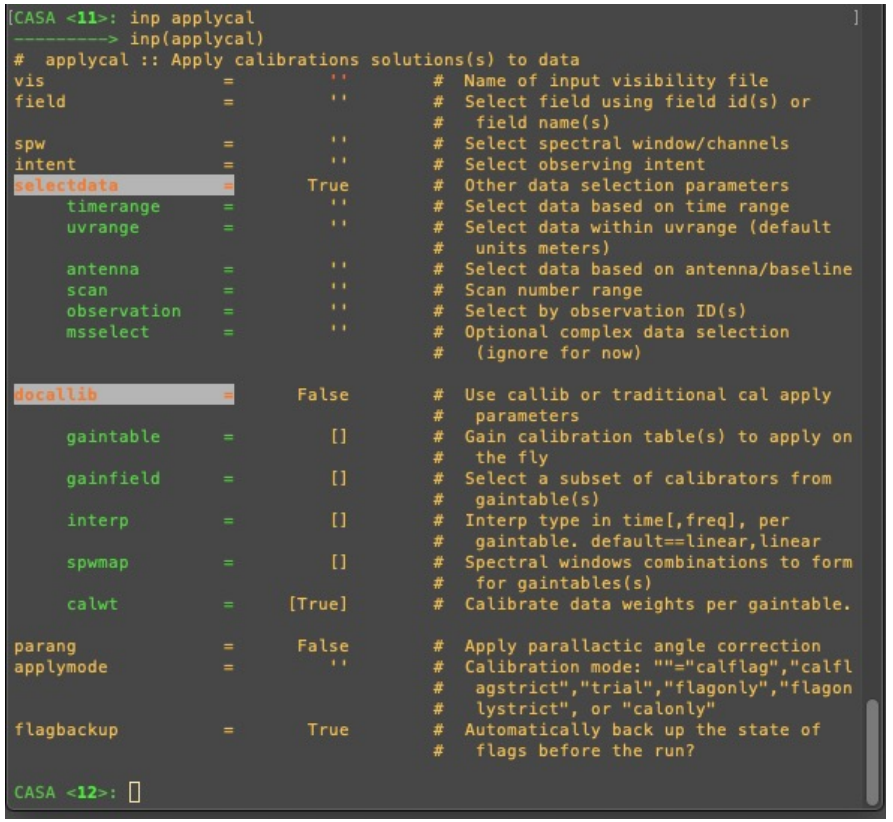


CASA tasks

- Tasks in CASA are the commands which are used to preform a specific function.
- Each contain a set of user definable parameters.
- To see what parameters a task has we can use the `inp` command.

Example: applycal

- applycal is the task used to apply calibration tables to the data.



- Typing just inp will give you the inputs for the last CASA task you used.

Getting more information

- For *most* parameters within a task you can get more information on what it wants by typing help(par.<param_name>)

```
[CASA <17>: help par.field                                                    ]
---------> help(par.field)
Help on function field in module parameter_dictionary:

field()
    field -- Select field using field id(s) or field name(s).
             [run listobs to obtain the list ids or names]
    default: 0  (for sdimaging)
             '' = all fields (for the other ASAP tasks)

    If field string is a non-negative integer, it is assumed a field index
    otherwise, it is assumed a field name
             field='0~2'; field ids 0,1,2
             field='0,4,5~7'; field ids 0,4,5,6,7
             field='3C286,3C295'; field named 3C286 adn 3C295
             field = '3,4C*'; field id 3, all names starting with 4C
    This selection is in addition to scanlist, iflist, and pollist.

    See help par.selectdata for additional syntax.
    See specific task for any additional details.
[(END)_                                                                       ]
```

Getting even more information

- CASA can take you straight to the CASA documentation webpage for a given task.
- For the table of contents you can type `doc('toc')`
- For some specific tasks you can type e.g. `doc('applycal')`
- Alternately, a Google search for 'CASAdocs NRAO <task name>' should bring up the relevant page. *(Beware Google seems to preferentially point to the ~2010 docs so make sure you get the more recent versions)*

Working with CASA

- To execute a task:

1. Default the task parameters with `default(taskname)`
2. Fill in all the parameters you need
3. Do an `inp` to check you've filled everything in the right format
4. Type the task name to execute it

```
[CASA <38>: default(applycal)                                                ]
```

```
[CASA <39>: vis='myvis.ms'                                                   ]

[CASA <40>: field='G123.45'                                                  ]

[CASA <41>: spw=0                                                            ]
```

```
[CASA <42>: inp                                                              ]
---------> inp()
#  applycal :: Apply calibrations solutions(s) to data
vis                 = 'myvis.ms'         #  Name of input visibility file
field               =  'G123.45'         #  Select field using field id(s) or
                                         #   field name(s)
spw                 =          0         #  Select spectral window/channels
```

```
[CASA <43>: spw='0'                                                          ]
```

```
[CASA <45>: inp                                                              ]
---------> inp()
#  applycal :: Apply calibrations solutions(s) to data
vis                 = 'myvis.ms'         #  Name of input visibility file
field               =  'G123.45'         #  Select field using field id(s) or
                                         #   field name(s)
spw                 =        '0'         #  Select spectral window/channels
```

```
[CASA <46>: applycal                                                         ]
---------> applycal()
```

Here CASA checks your input is of the right format. It will appear red if it is wrong

You can also enter the command and a list of parameters in one line e.g.

```
applycal(vis='myVis.ms', field='G123.45', spw='0')
```

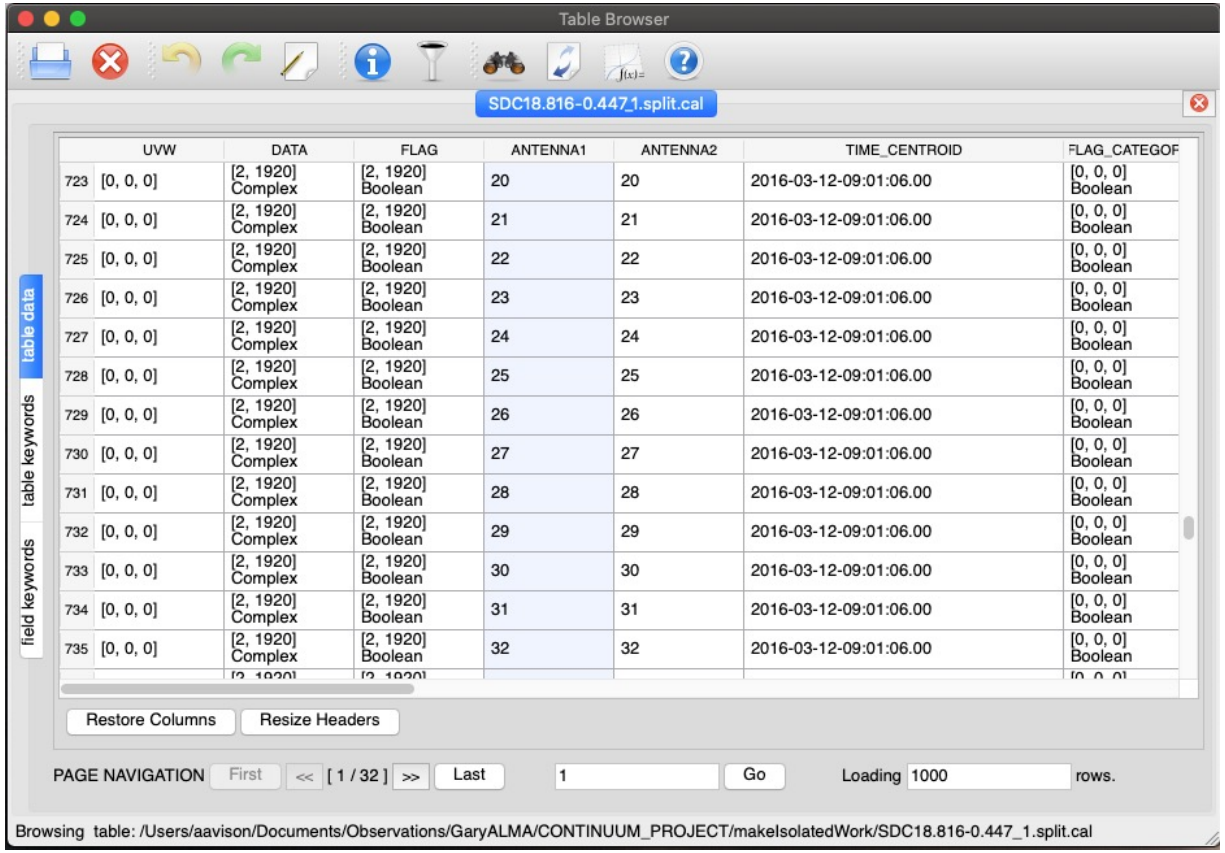
CASA Data

- CASA runs on visibilities and images which are themselves a directory structure.
- Raw visibilities come in ASDM (archival science data model) format.
- After import into CASA these visibilities in the MeasurementSet (MS) format.

```
cambria:TWHYA_tutorial aavison$ ls sis14_twhya_calibrated_flagged.ms
ANTENNA             PROCESSOR           table.f14           table.f23
ASDM_ANTENNA        SORTED_TABLE        table.f15           table.f23_TSM1
ASDM_CALWVR         SOURCE              table.f16           table.f24
ASDM_RECEIVER       SPECTRAL_WINDOW     table.f17           table.f24_TSM1
ASDM_STATION        STATE               table.f17_TSM1      table.f3
CALDEVICE           SYSCAL              table.f18           table.f4
DATA_DESCRIPTION    SYSPOWER            table.f19           table.f5
FEED                WEATHER             table.f2            table.f6
FIELD               table.dat           table.f20           table.f7
FLAG_CMD            table.f1            table.f20_TSM0      table.f8
HISTORY             table.f10           table.f21           table.f9
OBSERVATION         table.f11           table.f21_TSM1      table.info
POINTING            table.f12           table.f22           table.lock
POLARIZATION        table.f13           table.f22_TSM1
```

The Measurement Set

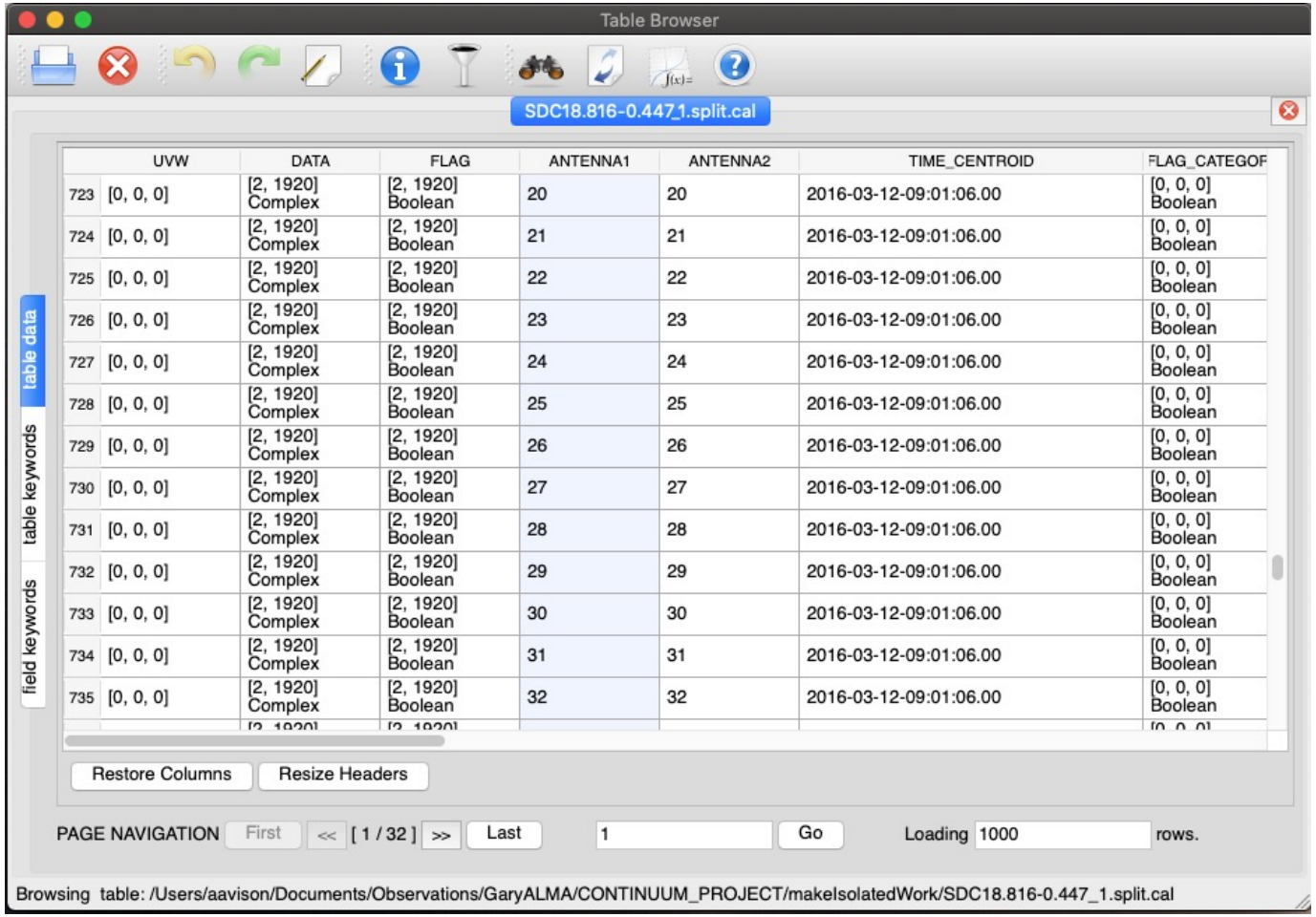
- As viewed from a terminal an MS appears as a directory containing subdirectories and data in binary format. (i.e. the image on the last slide)
- To get a look at that data we need to use the CASA task browsetable



	UVW	DATA	FLAG	ANTENNA1	ANTENNA2	TIME_CENTROID	FLAG_CATEGOF
723	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	20	20	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
724	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	21	21	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
725	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	22	22	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
726	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	23	23	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
727	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	24	24	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
728	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	25	25	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
729	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	26	26	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
730	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	27	27	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
731	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	28	28	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
732	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	29	29	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
733	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	30	30	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
734	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	31	31	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
735	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	32	32	2016-03-12-09:01:06.00	[0, 0, 0] Boolean

This is an example MS MAIN table. Each **Column** contains information of a specific type. Here we see the *uvw* values, Visibility Data, antenna numbers, time values and flagging information. There are a lot more columns than shown here.

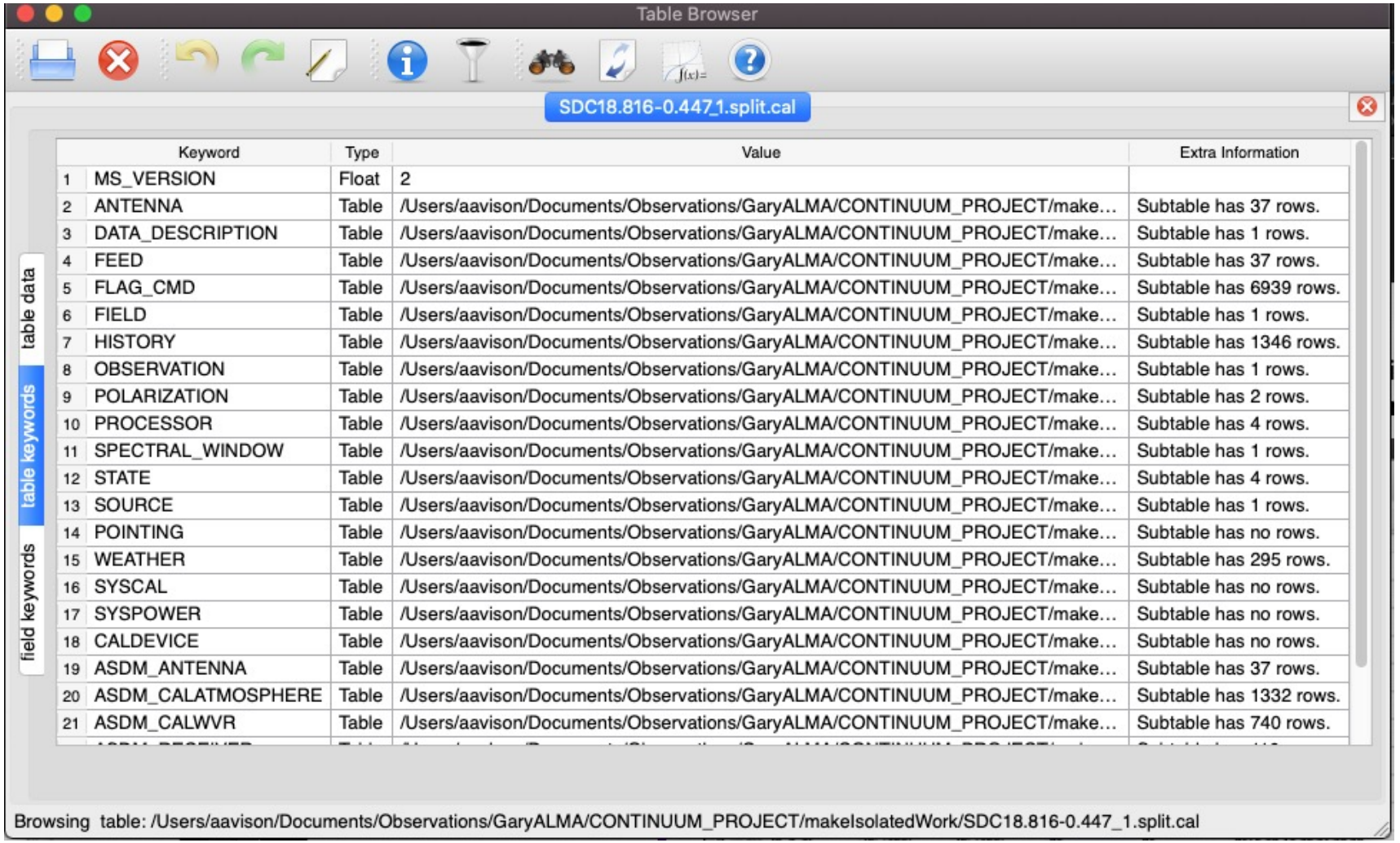
TIME



	UVW	DATA	FLAG	ANTENNA1	ANTENNA2	TIME_CENTROID	FLAG_CATEGOF
723	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	20	20	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
724	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	21	21	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
725	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	22	22	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
726	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	23	23	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
727	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	24	24	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
728	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	25	25	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
729	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	26	26	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
730	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	27	27	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
731	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	28	28	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
732	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	29	29	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
733	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	30	30	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
734	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	31	31	2016-03-12-09:01:06.00	[0, 0, 0] Boolean
735	[0, 0, 0]	[2, 1920] Complex	[2, 1920] Boolean	32	32	2016-03-12-09:01:06.00	[0, 0, 0] Boolean

Each Row gives you the information for a given Column at a unique **Baseline** at a unique **Time**.
Time increases as you scroll down the table.

In Addition to the MAIN table there are many subtables in an MS which can be navigated using browsetable



	Keyword	Type	Value	Extra Information
1	MS_VERSION	Float	2	
2	ANTENNA	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 37 rows.
3	DATA_DESCRIPTION	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 1 rows.
4	FEED	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 37 rows.
5	FLAG_CMD	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 6939 rows.
6	FIELD	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 1 rows.
7	HISTORY	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 1346 rows.
8	OBSERVATION	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 1 rows.
9	POLARIZATION	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 2 rows.
10	PROCESSOR	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 4 rows.
11	SPECTRAL_WINDOW	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 1 rows.
12	STATE	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 4 rows.
13	SOURCE	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 1 rows.
14	POINTING	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has no rows.
15	WEATHER	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 295 rows.
16	SYSCAL	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has no rows.
17	SYSPOWER	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has no rows.
18	CALDEVICE	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has no rows.
19	ASDM_ANTENNA	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 37 rows.
20	ASDM_CALATMOSPHERE	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 1332 rows.
21	ASDM_CALWVR	Table	/Users/aavison/Documents/Observations/GaryALMA/CONTINUUM_PROJECT/make…	Subtable has 740 rows.

Finally, some handy things in CASA:

- In the CASA terminal pressing the up arrow key will show you previous commands you have used.
- If you type a letter or part of a word and press the up arrow it will scroll through previously entered commands which start with those letters.
- The CASA terminal also “tab completes” so if you start typing something and hit the Tab key you’ll be given a list of options which match the text you have written so far.
- If you run a task and exit CASA, then restart CASA later in the same directory you can type `tget <taskname>` . This will restore the previous parameters from your last *successful* run of that task.
  (They are stored in the <taskname>.last file you will see popping up as you run CASA)

Which software to use?