ASKAPsoft tutorial

ASKAPsoft Imaging Tutorial

- Courtesy of (and many thanks to) Wasim Raja
- Wasim has prepared four scripts to:
  - Generate input slurm scripts, parsets and associated files
  - Launch jobs on Galaxy
- The scripts are:
  1. bandpass calibration: **do_cal_1934.sh**
  2. prepare science data: **do_pre_process_ras.sh**
  3. image/selfcal science data (continuum only): **do_selfcal_ras.sh**
  4. form linear mosaic: **do_linmos_ras.sh**
- Also:
  - a script to set up galaxy modules: **setup_modules_on_nodes.sh**
  - a file to configure various parameters: **process_ASKAPdata.config**

setup_modules_on_nodes.sh

```
module use /group/askap/modulefiles
module unload askapsoft
module load askapsoft/0.22.1

module unload askapdata
module load askapdata

module unload askappipeline
module load askappipeline
#module load askapcli

export PMI_NO_PREINITIALIZE=1
export PMI_NO_FORK=1
export PMI_DEBUG=1

module unload askap-cray
module load askap-cray

module unload slurm
module load slurm
```

process_ASKAPdata.config

```
export TRIAL=0                                  # set to 1 to generate files but not run them

export SPLIT_CHAN=1                             # split out a subset of frequency channels
export BCHAN_SPLIT=8192
export ECHAN_SPLIT=8407 #9271

export MY_SBID_BPCAL=5181                       #  scheduling block for band-pass calibration (i.e. the id of the BP calibration
                                                # observation)
export MY_SBID_TARGET=5177                      #  scheduling block for science data (i.e. the id of the science observation)
export MY_FIELD_NAME=COSMOLOGY_T15-2            # name of the science field
export PATH_TO_SETUP_FILE=$PWD                  # change me if running from a different directory
export MY_OUTPATH=ras_data_processing_${this_user}/
mkdir -p ${MY_OUTPATH}msdata/${MY_SBID_TARGET} ${MY_OUTPATH}bpcal_solutions/${MY_SBID_BPCAL}

# Decide which beams you wish to process. Do bandpass calibration for all 36 beams, but restrict imaging and selfcal to 1 or a few
export BBEAM_BPCAL=0 # Must be 0 with the current structure of bptables
export EBEAM_BPCAL=35 # Can be less than maxBeams
export BBEAM=0 # image / selfcal beams 0 to 1
export EBEAM=1

# Some imaging parameters:
export ROBUST=-0.5
export BLOOP_SELFCAL=0
export ELOOP_SELFCAL=1
```

Tutorial — Yesterday

$ mkdir askap_tutorial

$ cd askap_tutorial

$ cp –r /group/askap/dmitchell/askap_tutorial/* .

- “Source” some setup files:

$. setup_modules_on_nodes.sh

$. process_ASKAPdata.config

- process_ASKAPdata.config will set up things like a directory for output and input of scripts: $MY_OUTPATH (set to ras_data_processing_username) and various calibration and imaging parameters

Tutorial — Yesterday

- Generate solutions yourselves:

  $ ./do_cal_1934.sh

  - mssplit — select a subset of channels (to limit the amount of processing)
  - cflag — look for radio frequency interference and set flags
  - cbpcalibrator — run the calibrator for each frequency channel

- Or just copy the solution table that I generated:

  $. process_ASKAPdata.config

  $ mv cbpcal_1934_sb5181_bm0-bm35_refant-1_bp.tab \
  $MY_OUTPATH/bpcal_solutions/5181/

Tutorial — Yesterday

Plot some bandpass calibration solutions. Make sure you have logged in with X11 forwarding:

$ ssh –X username@galaxy.pawsey.org.au

- or:

$ ssh –Y username@galaxy.pawsey.org.au

For the help menu:

$ plot_bandpass.py –h

optional arguments:

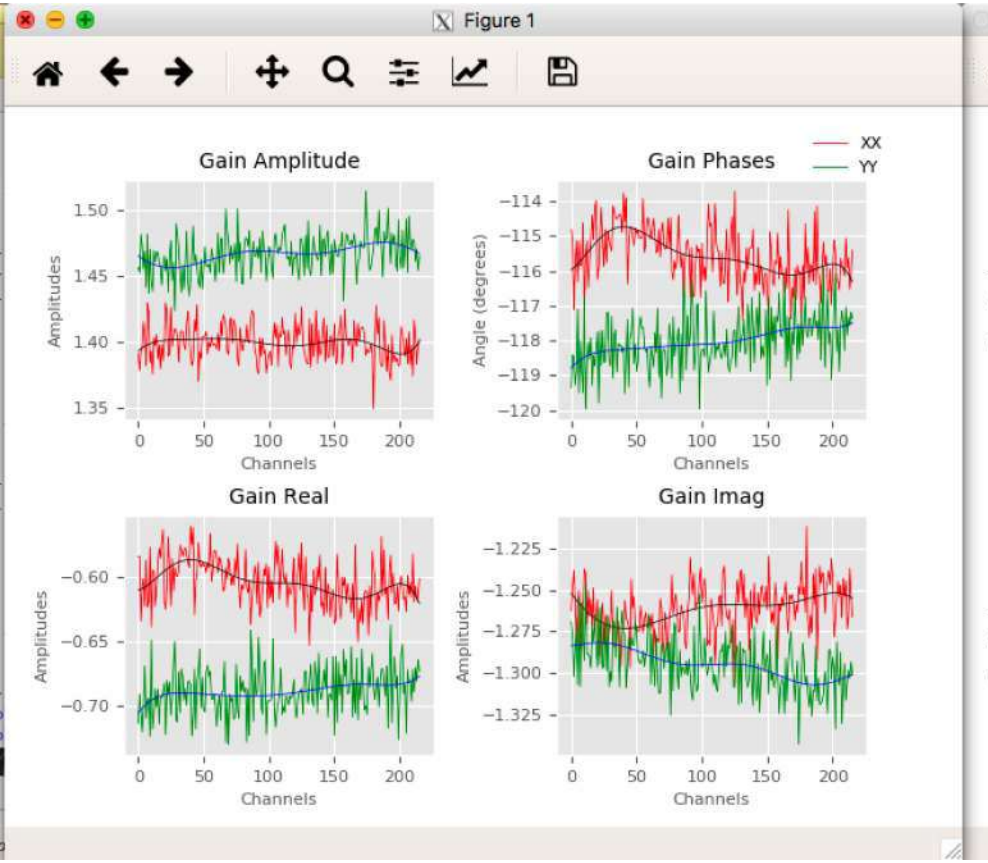
-t BP_TAB, --t BP_TAB	Input Bandpass table (with path)
-ib BEAM_NUM, --ib BEAM_NUM	The beam number you wish to process
-ia ANTE_NUM, --ia ANTE_NUM	The antenna number you wish to process

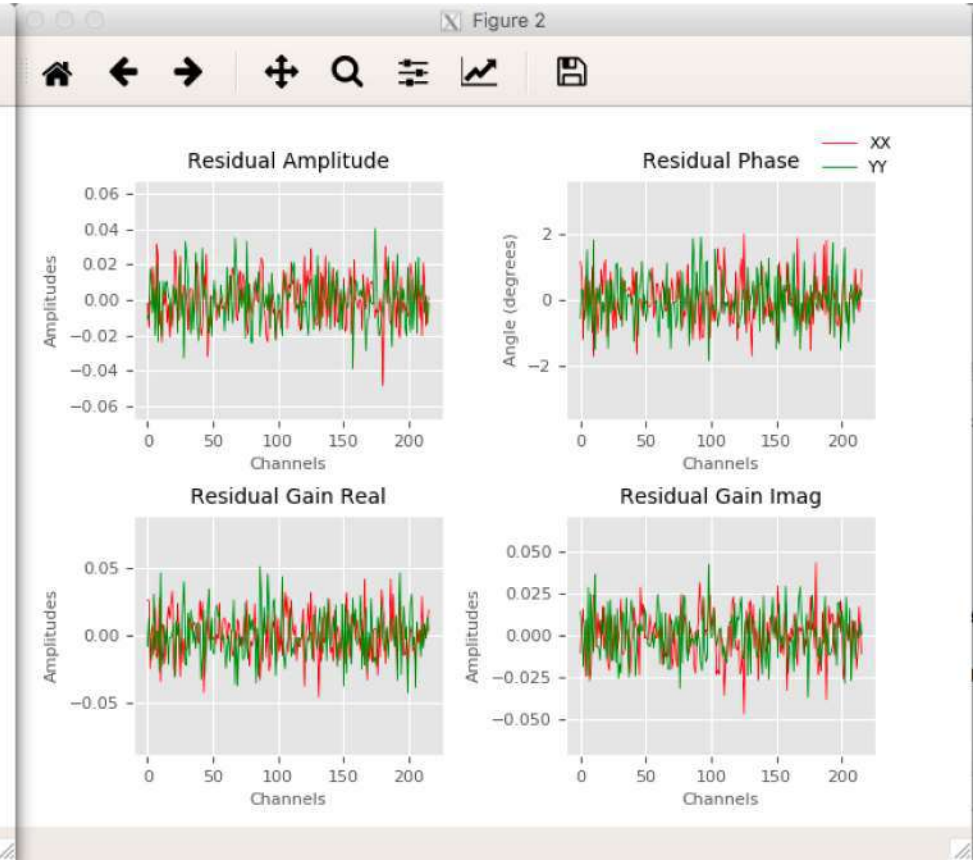
$ plot_bandpass.py -t cbpcal_1934_sb5181_bm0-bm35_refant-1_bp.tab –ia 1

```
Successful readonly open of default-locked table cbpcal_1934_sb5181_bm0-bm35_refant-1_bp.tab: 3 columns, 1 rows
Plotting bandpass solutions for Input table: cbpcal_1934_sb5181_bm0-bm35_refant-1_bp.tab
For:
             Beam Num:  0
             Ante Num:  1
Smooth fits will be derived using:
          Poly Order:  2
          Harm Order:  3
```

Tutorial — Yesterday

plot_bandpass.py -t cbpcal_1934_sb5181_bm0-bm35_refant-1_bp.tab





Tutorial — Today

- Look at visibilities
- Make images
- Look at images
- Mosaic image
- Look at mosaics

- Two options for looking at results:
  - Download (scp) to your local machine and look with casa tools.
  - Use remotevis.pawsey.org.au to use casa remotely on the Zeus cluster.

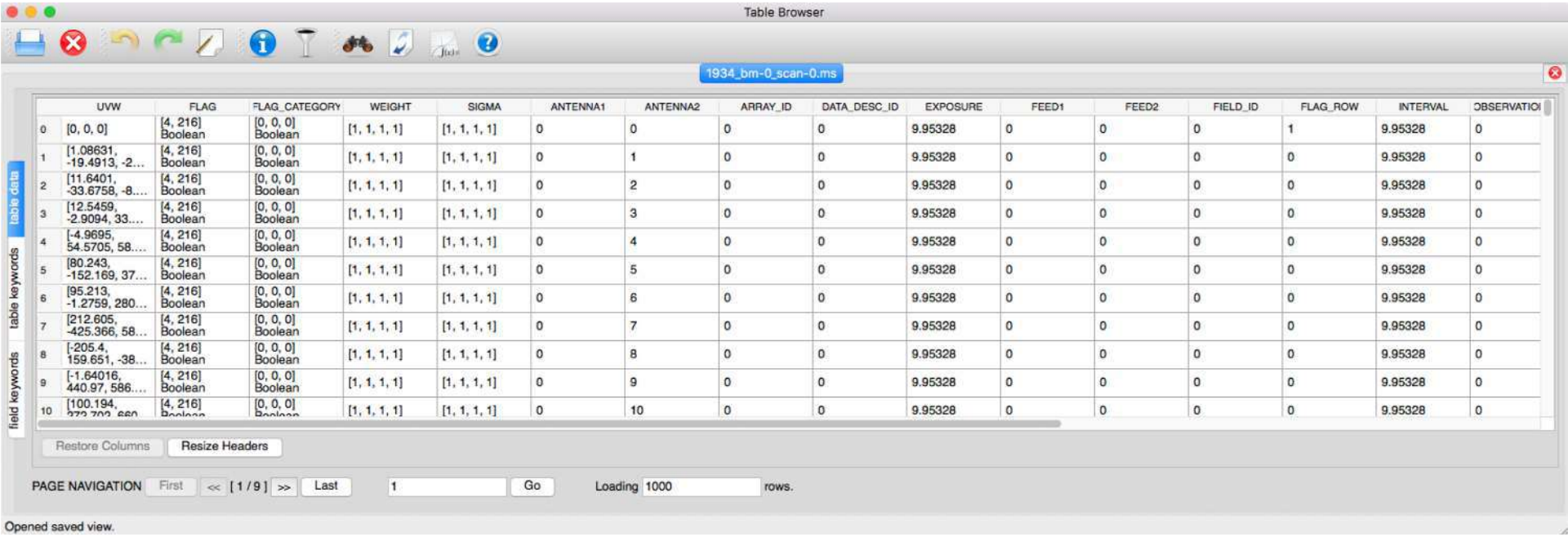
Tutorial — Today

On local machine (replace $MY_OUTPATH with full directory path):

$ scp -r username@hpc-data.pawsey.org.au:$MY_OUTPATH/msdata/5181/FLAGGED_DYNAMIC/1934_bm-0_scan-0.ms .

Have a look at the contents of the measurement set

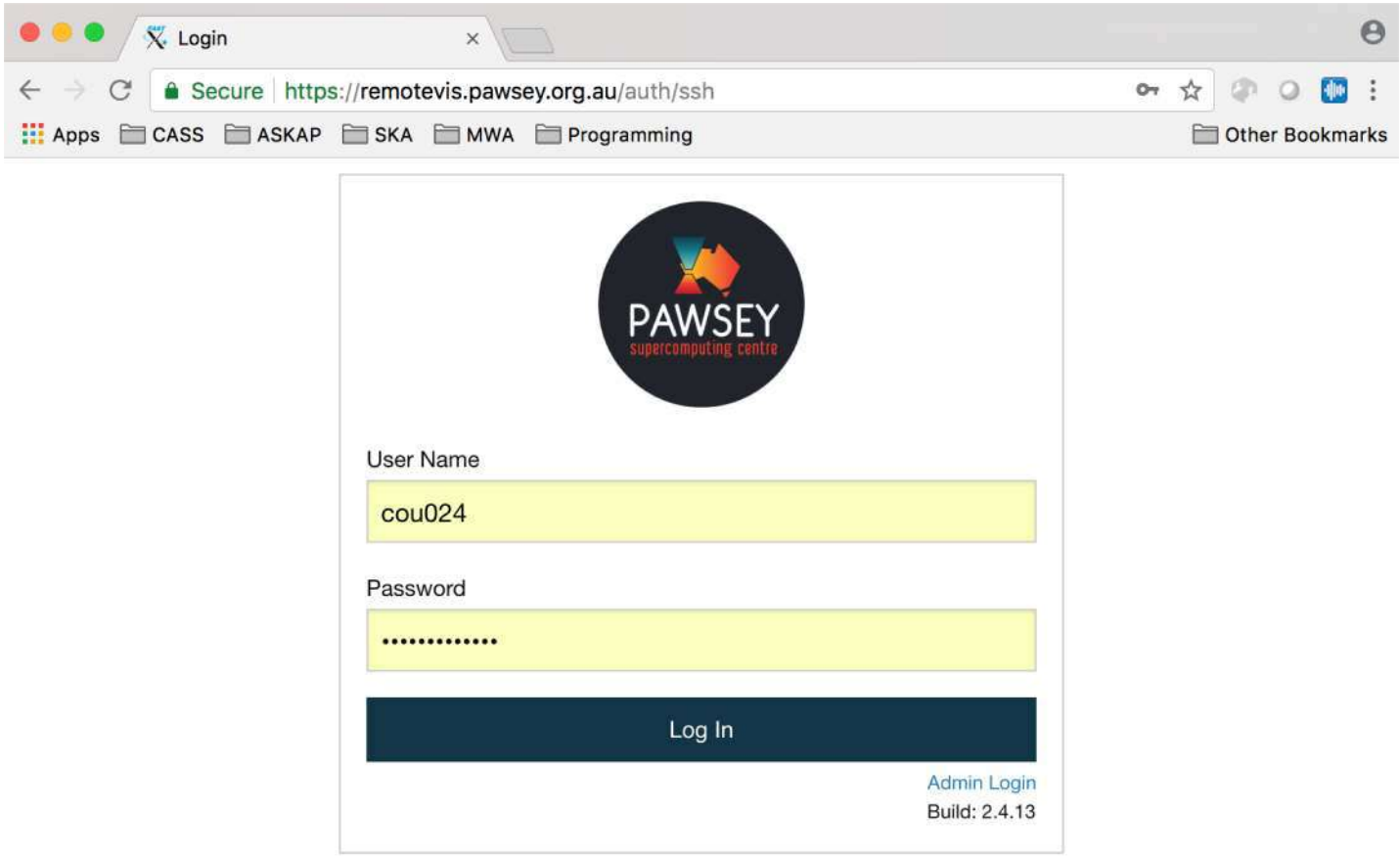
$ casabrowser 1934_bm-0_scan-0.ms



	UVW	FLAG	FLAG_CATEGORY	WEIGHT	SIGMA	ANTENNA1	ANTENNA2	ARRAY_ID	DATA_DESC_ID	EXPOSURE	FEED1	FEED2	FIELD_ID	FLAG_ROW	INTERVAL	OBSERVATIO
0	[0, 0, 0]	[4, 216] Boolean	[0, 0, 0] Boolean	[1, 1, 1, 1]	[1, 1, 1, 1]	0	0	0	0	9.95328	0	0	0	1	9.95328	0
1	[1.08631, -19.4913, -2...	[4, 216] Boolean	[0, 0, 0] Boolean	[1, 1, 1, 1]	[1, 1, 1, 1]	0	1	0	0	9.95328	0	0	0	0	9.95328	0
2	[11.6401, -33.6758, -8....	[4, 216] Boolean	[0, 0, 0] Boolean	[1, 1, 1, 1]	[1, 1, 1, 1]	0	2	0	0	9.95328	0	0	0	0	9.95328	0
3	[12.5459, -2.9094, 33....	[4, 216] Boolean	[0, 0, 0] Boolean	[1, 1, 1, 1]	[1, 1, 1, 1]	0	3	0	0	9.95328	0	0	0	0	9.95328	0
4	[-4.9695, 54.5705, 58....	[4, 216] Boolean	[0, 0, 0] Boolean	[1, 1, 1, 1]	[1, 1, 1, 1]	0	4	0	0	9.95328	0	0	0	0	9.95328	0
5	[80.243, -152.169, 37...	[4, 216] Boolean	[0, 0, 0] Boolean	[1, 1, 1, 1]	[1, 1, 1, 1]	0	5	0	0	9.95328	0	0	0	0	9.95328	0
6	[95.213, -1.2759, 280...	[4, 216] Boolean	[0, 0, 0] Boolean	[1, 1, 1, 1]	[1, 1, 1, 1]	0	6	0	0	9.95328	0	0	0	0	9.95328	0
7	[212.605, -425.366, 58...	[4, 216] Boolean	[0, 0, 0] Boolean	[1, 1, 1, 1]	[1, 1, 1, 1]	0	7	0	0	9.95328	0	0	0	0	9.95328	0
8	[-205.4, 159.651, -38...	[4, 216] Boolean	[0, 0, 0] Boolean	[1, 1, 1, 1]	[1, 1, 1, 1]	0	8	0	0	9.95328	0	0	0	0	9.95328	0
9	[-1.64016, 440.97, 586....	[4, 216] Boolean	[0, 0, 0] Boolean	[1, 1, 1, 1]	[1, 1, 1, 1]	0	9	0	0	9.95328	0	0	0	0	9.95328	0
10	[100.194, 772.702, 660	[4, 216] Boolean	[0, 0, 0] Boolean	[1, 1, 1, 1]	[1, 1, 1, 1]	0	10	0	0	9.95328	0	0	0	0	9.95328	0

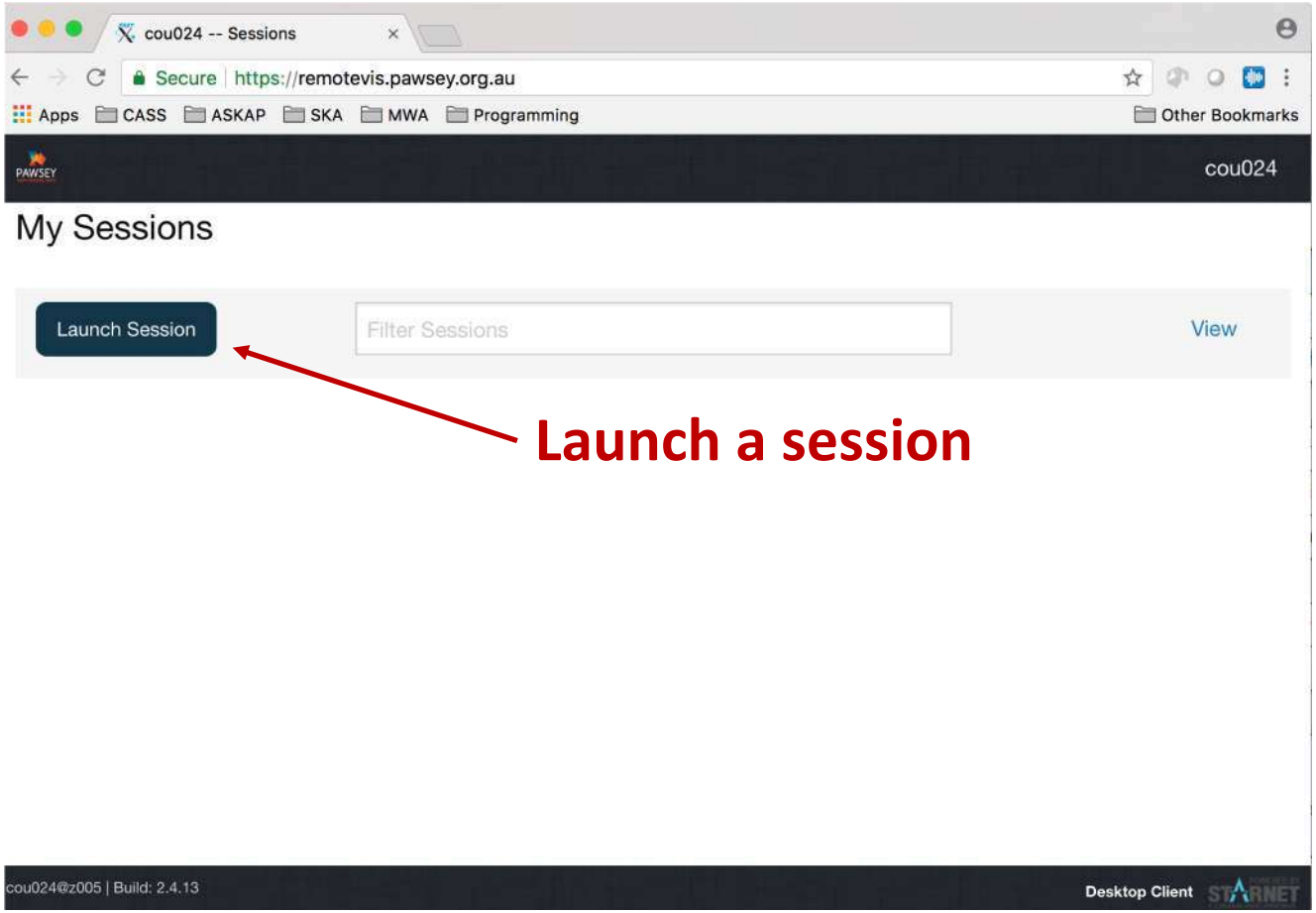
Tutorial — Today

remotely on the Zeus cluster using remotevis.pawsey.org.au



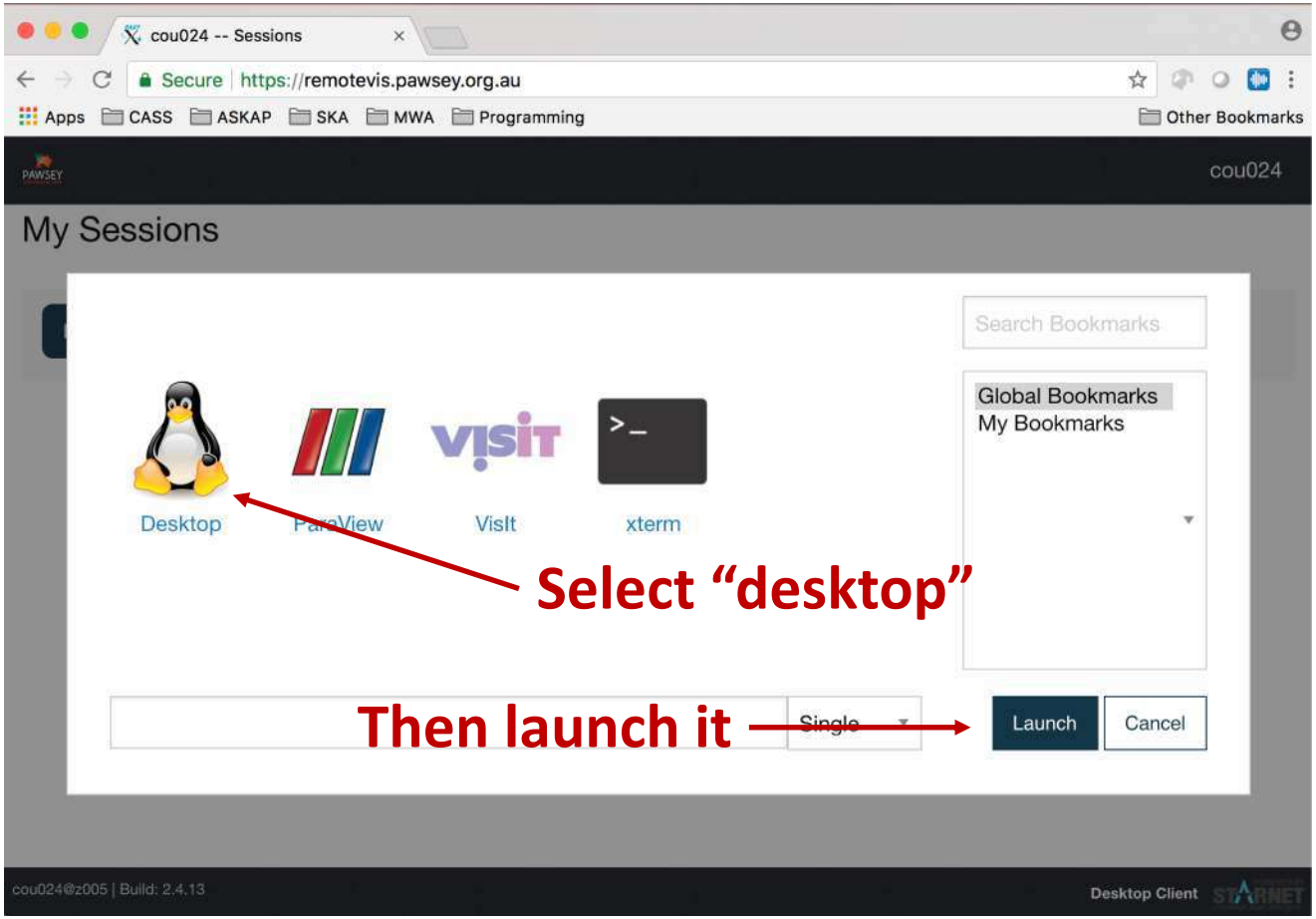
Tutorial — Today

remotely on the Zeus cluster using remotevis.pawsey.org.au



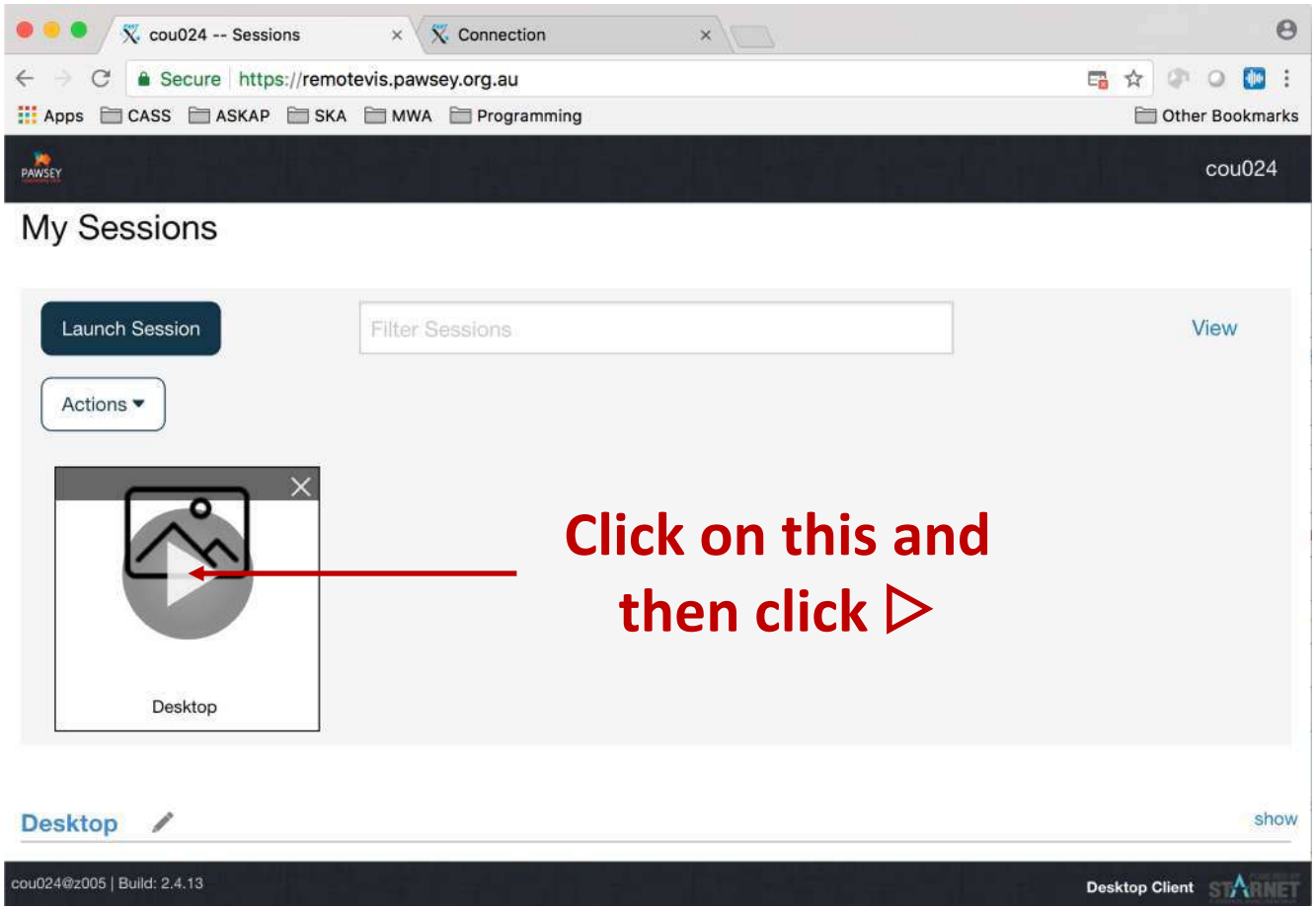
Tutorial — Today

remotely on the Zeus cluster using remotevis.pawsey.org.au



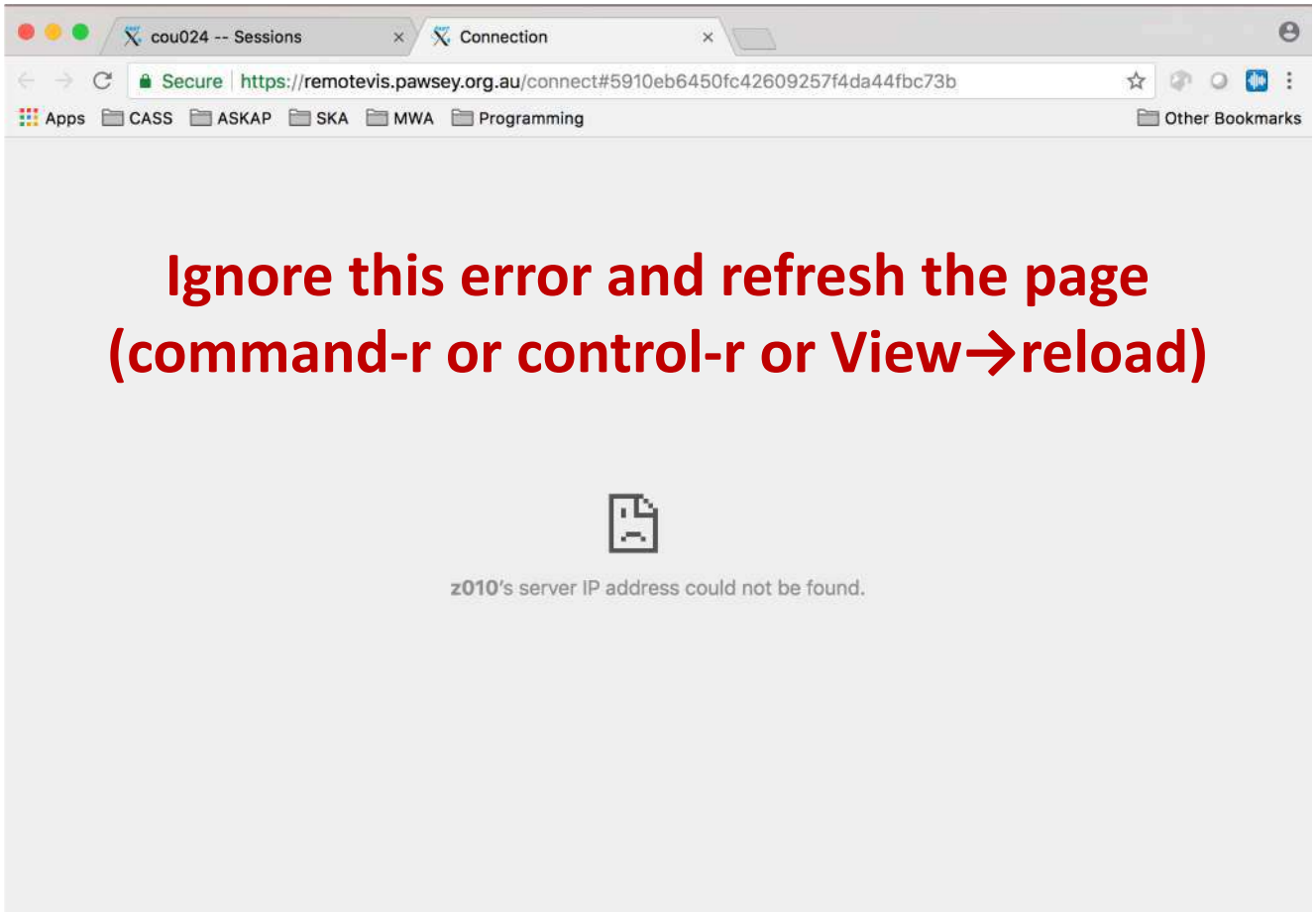
Tutorial — Today

remotely on the Zeus cluster using remotevis.pawsey.org.au

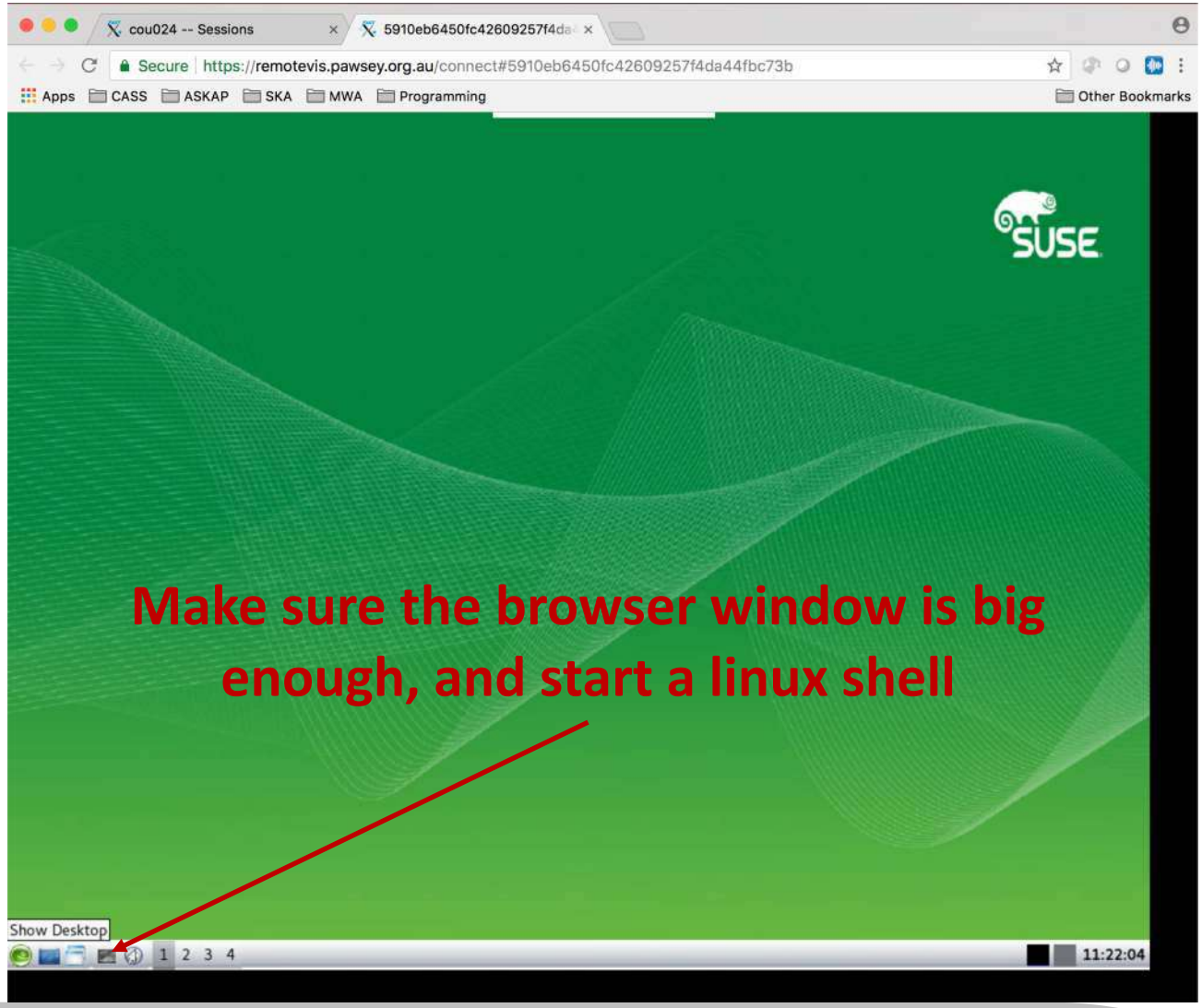


Tutorial — Today

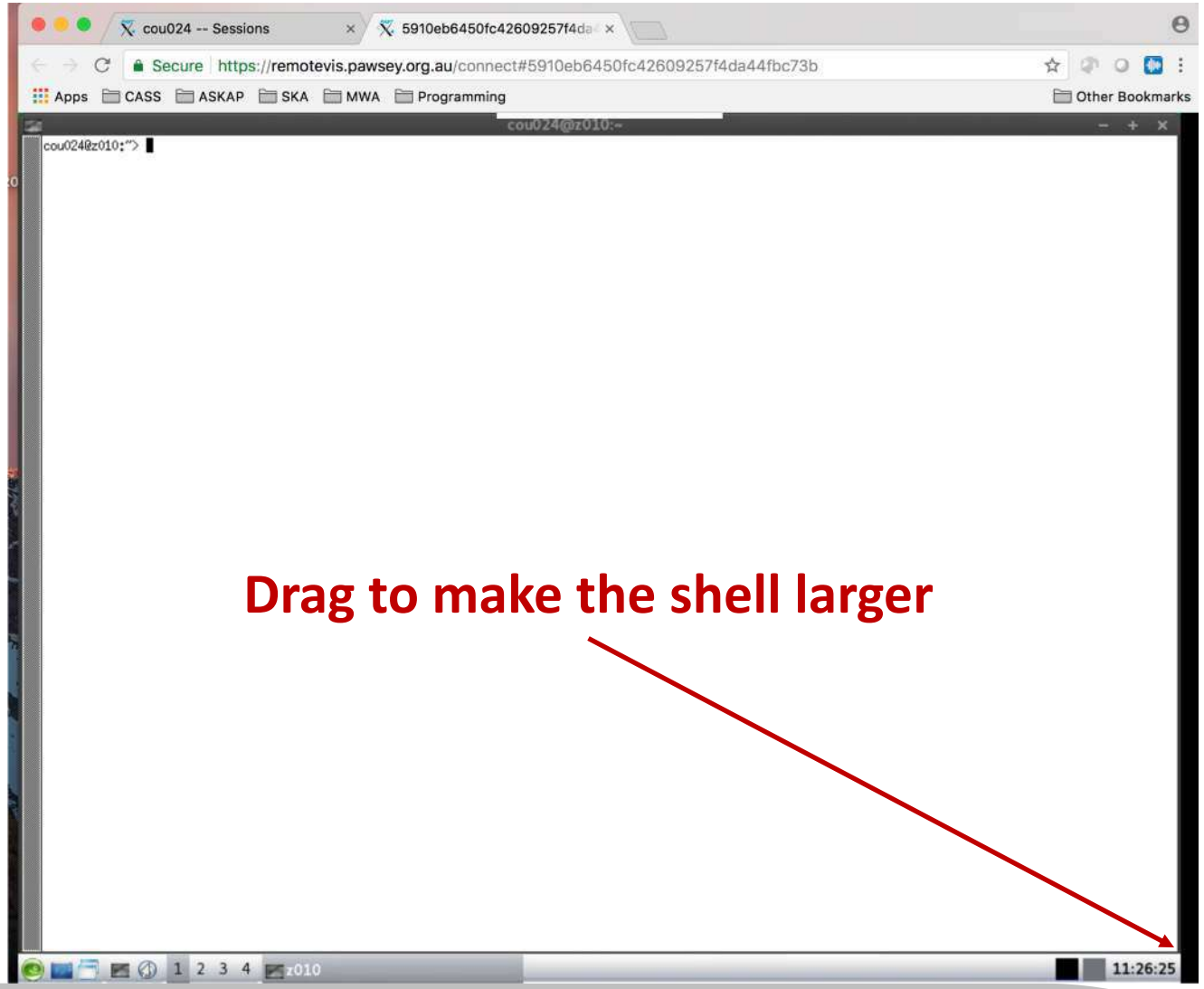
remotely on the Zeus cluster using remotevis.pawsey.org.au



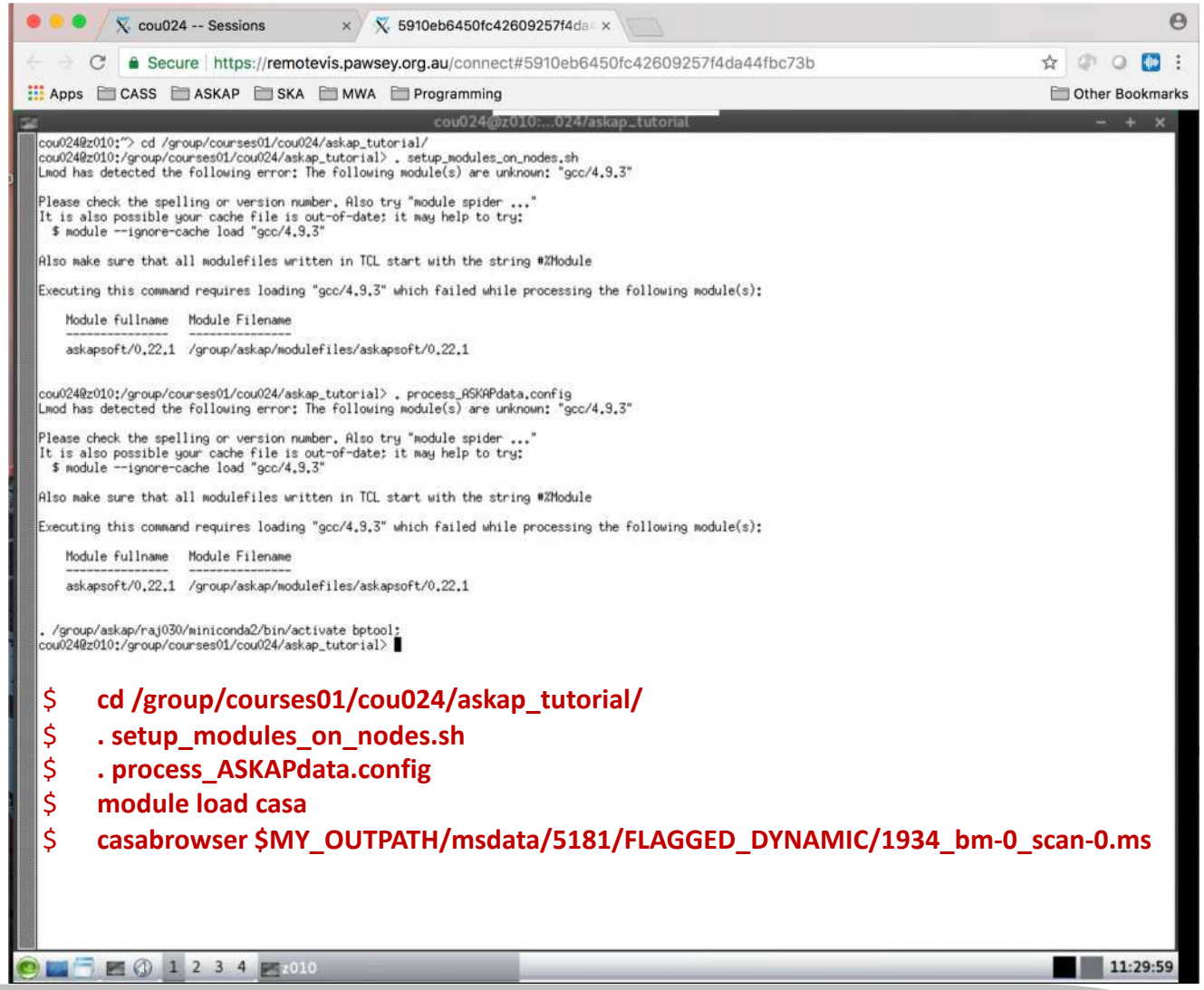
Tutorial — Today



Tutorial — Today

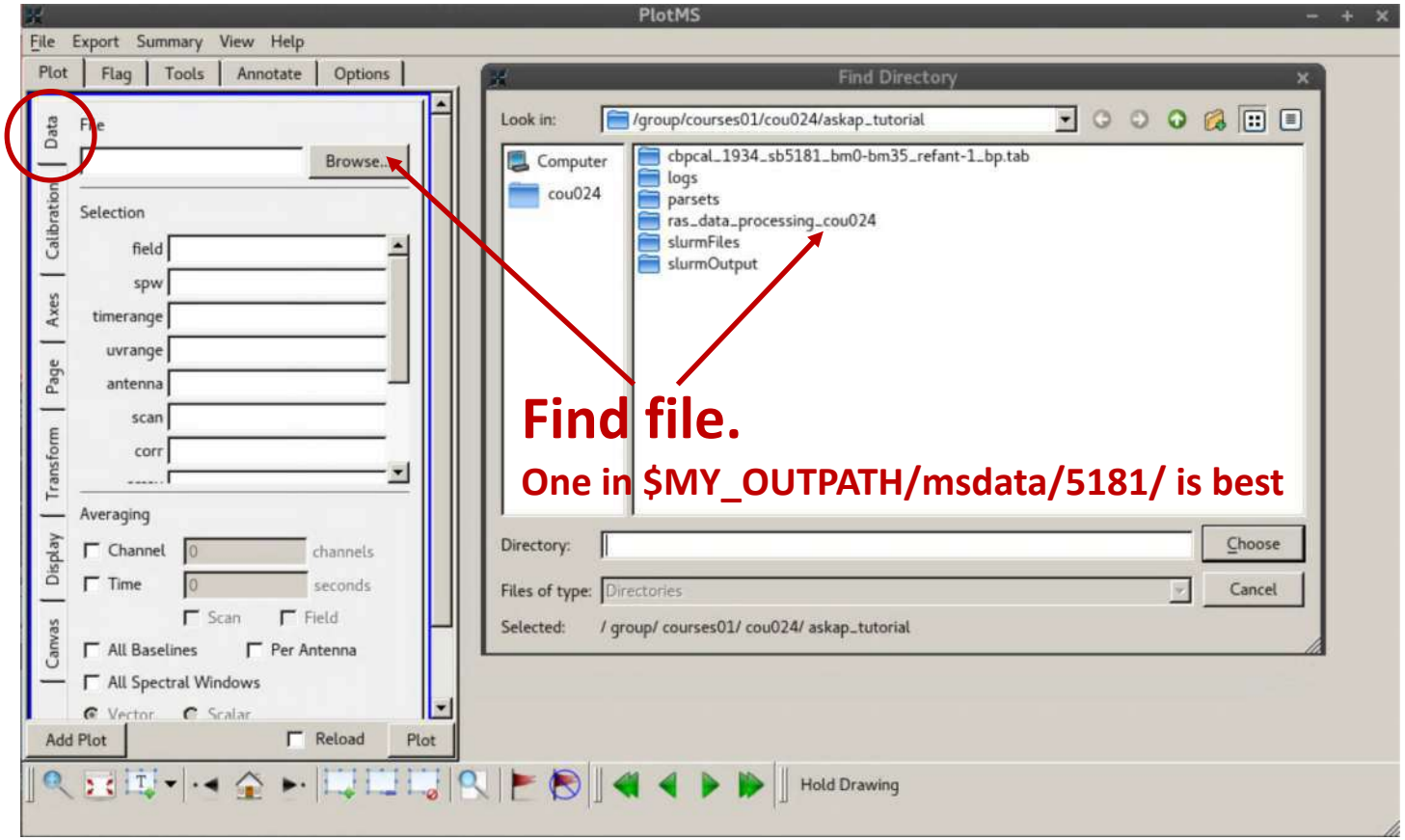


Tutorial — Today



Tutorial — Today

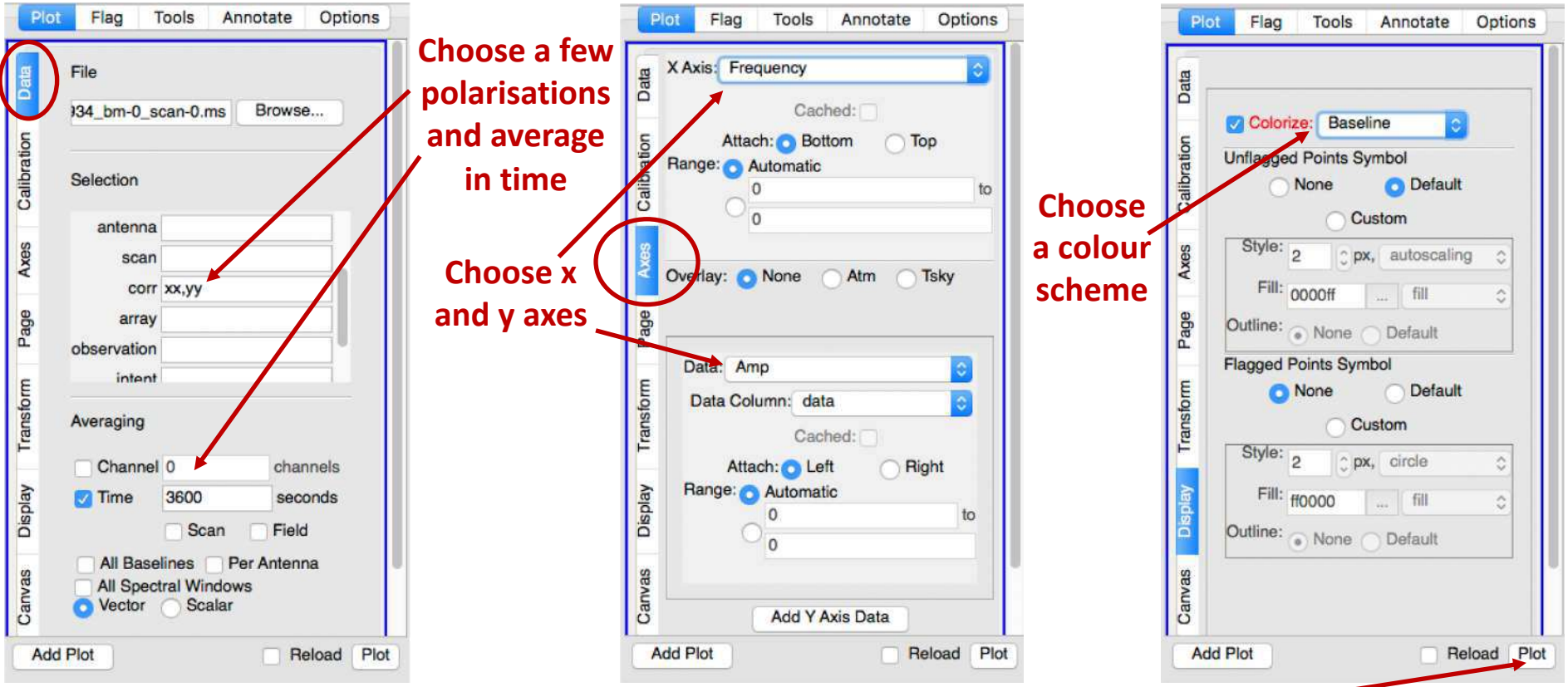
On local machine or remotevis.pawsey.org.au, plot data in the measurement set:



Tutorial — Today

On local machine or remotevis.pawsey.org.au, plot data in the measurement set:

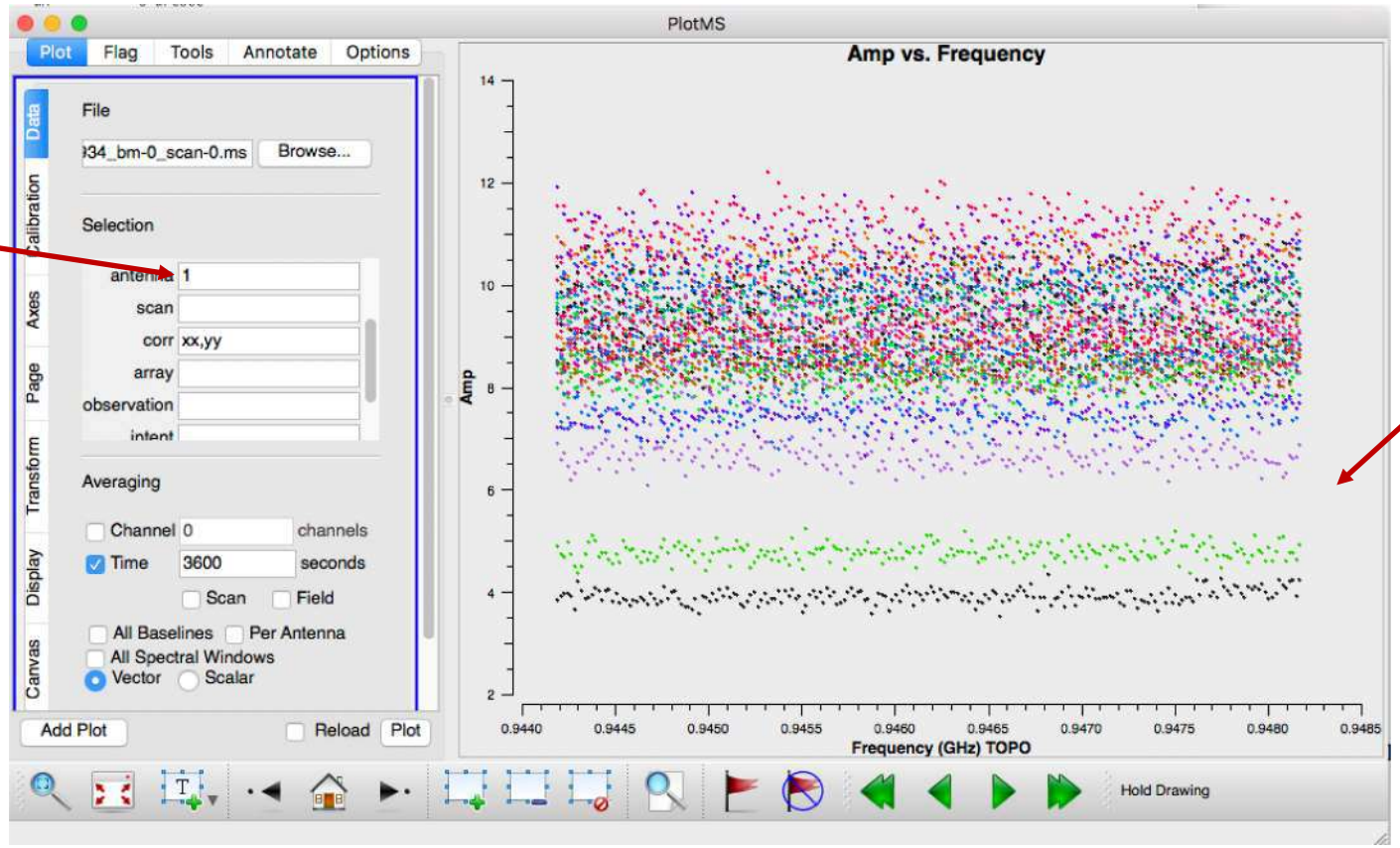
$ casaplotms



Tutorial — Today

On local machine or remotevis.pawsey.org.au, plot data in the measurement set:

$ casaplotms



Just plot for antenna 1

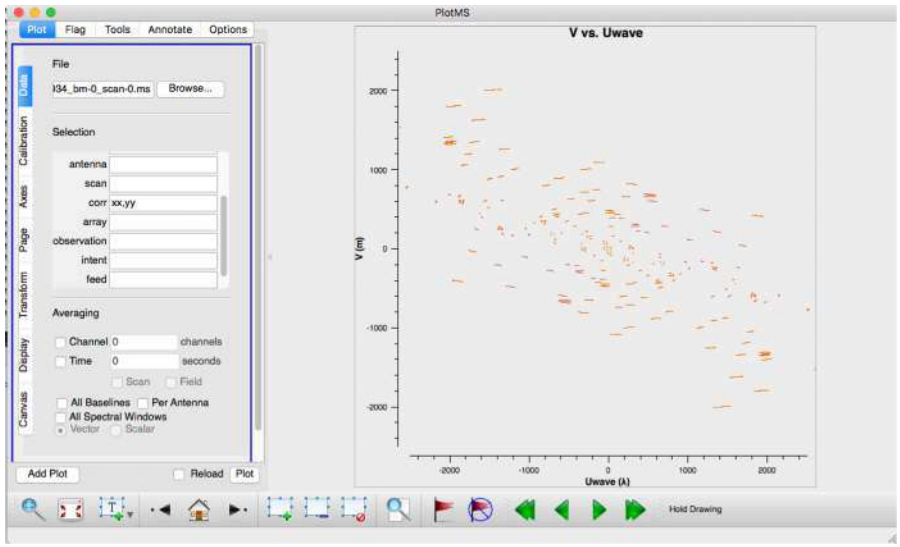
Before calibration

Tutorial — Today

On local machine or remotevis.pawsey.org.au, plot data in the measurement set:

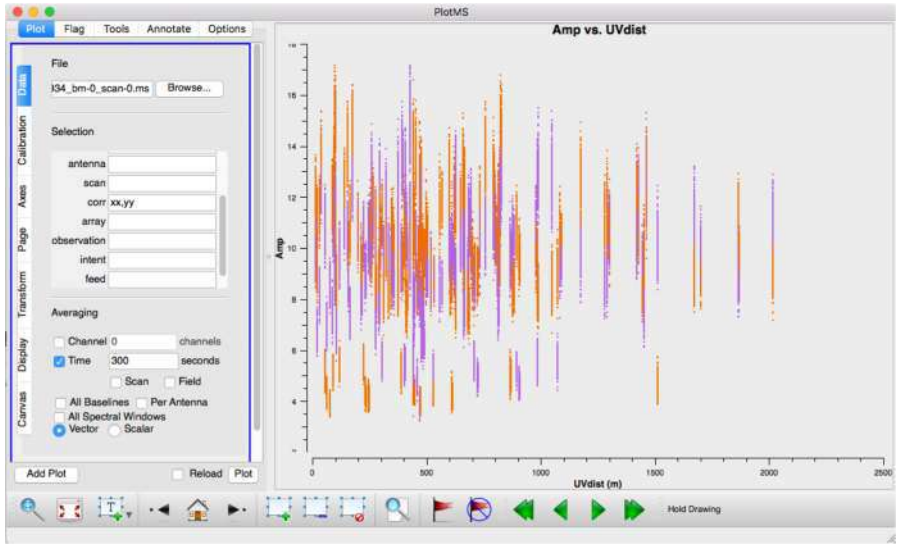
$ casaplotms

Plot the uv coverage



Plot amplitude versus uvdist

"uv distance" = $\sqrt{u^2 + u}$



Calibrate the Calibrator!

- **Generate a new file apply_cal.in:**

```
Ccalapply.dataset                          = test_cal.ms
Ccalapply.calibaccess                      = table
Ccalapply.calibaccess.table.maxant         = 16
Ccalapply.calibaccess.table.maxbeam        = 36
Ccalapply.calibaccess.table.maxchan        = 216
Ccalapply.calibaccess.table                = cbpcal_1934_sb5181_bm0-bm35_refant-1_bp.tab
```

- **Generate a new file apply_cal.sbatch:**

```
#!/usr/bin/env bash
#SBATCH --partition=workq
#SBATCH --time=00:05:00
#SBATCH --ntasks=20
#SBATCH --ntasks-per-node=20
#SBATCH --job-name=apply_cal
#SBATCH --account=courses01
#SBATCH --reservation=courseq
#SBATCH --export=ALL
srun --ntasks=19 --ntasks-per-node=19 ccalapply -c apply_cal.in > apply_cal.log
```

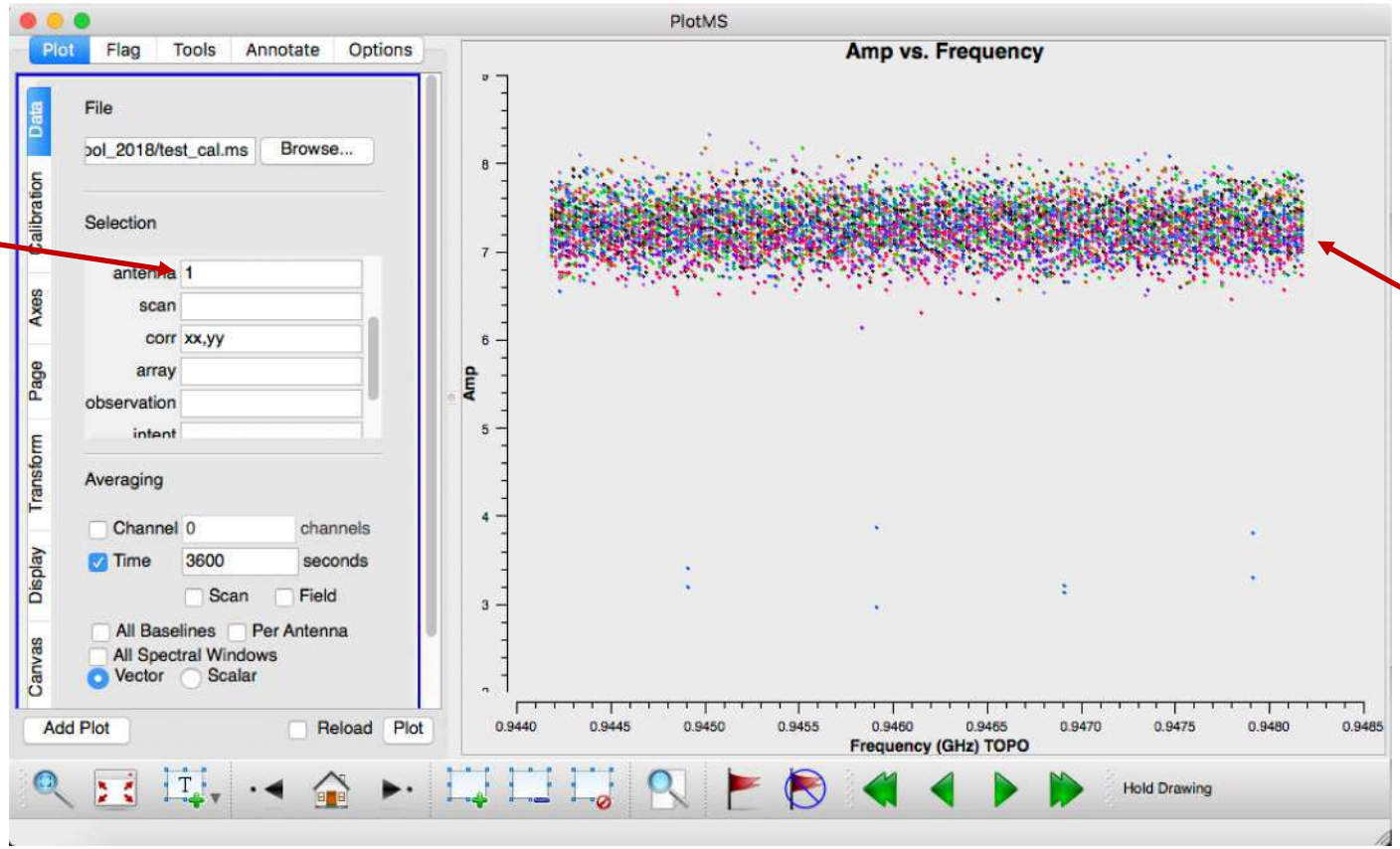
- **Run:**

```
cp -r $MY_OUTPATH/msdata/5181/1934_bm-0_scan-0.ms test_cal.ms
cp -r $MY_OUTPATH/bpcal_solutions/5181/cbpcal_1934_sb5181_bm0-bm35_refant-1_bp.tab .
sbatch apply_cal.sbatch
```

Tutorial — Today

On local machine or remotevis.pawsey.org.au, plot data in the measurement set:

$ casaplotms



do_pre_process_ras.sh

$ ./do_pre_process_ras.sh

- mssplit — select the same subset of channels from the science dataset
- ccalapply — apply calibration solutions to the science data
- cflag — look for radio frequency interference and set flags
- mssplit — average in frequency
- cflag — a final round of flagging

do_selfcal_ras.sh

$ ./do_selfcal_ras.sh

- ccalibrator — run calibration using a model of this field
- cimager — image and deconvolve the field with the new calibration solutions
- selavy — run relatively shallow source finder on the restored image
- cmodel — generate a model image from the selavy catalogue

- 1st run: set BLOOP_SELFCAL=0 & ELOOP_SELFCAL=0: imaging with no selfcal

$ squeue -u username

```
•  JOBID    USER     ACCOUNT                NAME EXEC_HOST ST     REASON   START_TIME     END_TIME  TIME_LEFT NODES   PRIORITY
•  5055128  dmitchel askaprt         IMG-5177-0A.I  nid00217  R       None     08:36:54     14:36:54    5:56:54     1      10001
•  5055129  dmitchel askaprt         IMG-5177-1A.I  nid00299  R       None     08:36:54     14:36:54    5:56:54     1      10001
```

do_selfcal_ras.sh

$ ls -ld ${MY_OUTPATH}/image/5177/weight*

image/5177/weights.I.COSMOLOGY_T15-2A_bm-0_iter-0

image/5177/weights.I.COSMOLOGY_T15-2A_bm-1_iter-0

$ ls -ld ${MY_OUTPATH}/image/5177/image*restored

image/5177/image.I.COSMOLOGY_T15-2A_bm-0_iter-0.restored

image/5177/image.I.COSMOLOGY_T15-2A_bm-1_iter-0.restored

$ ls -ld ${MY_OUTPATH}/image/5177/image*restored.cmodel

image/5177/image.I.COSMOLOGY_T15-2A_bm-0_iter-0.restored.cmodel

image/5177/image.I.COSMOLOGY_T15-2A_bm-1_iter-0.restored.cmodel

$ ls -ld ${MY_OUTPATH}/image/5177/psf*

image/5177/psf.I.COSMOLOGY_T15-2A_bm-0_iter-0

image/5177/psf.I.COSMOLOGY_T15-2A_bm-1_iter-0

image/5177/psf.image.I.COSMOLOGY_T15-2A_bm-0_iter-0

image/5177/psf.image.I.COSMOLOGY_T15-2A_bm-1_iter-0

do_linmos_ras.sh

$ ./do_linmos_ras.sh

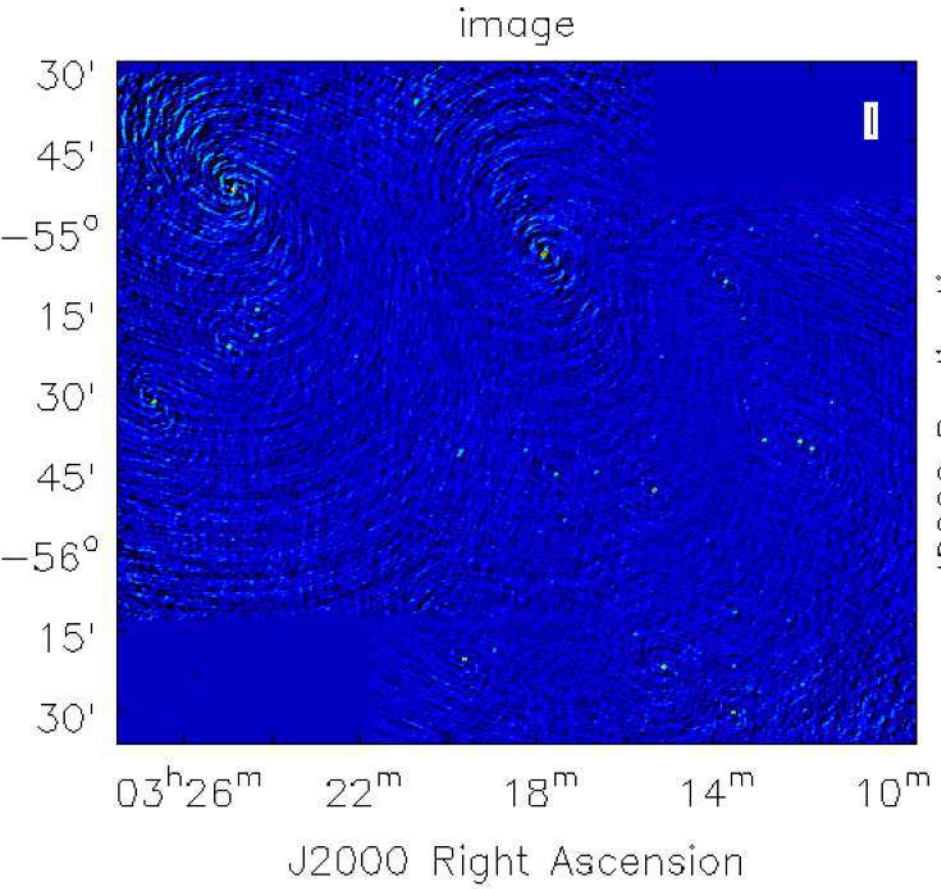
- linmos — form a linear mosaic of the final images

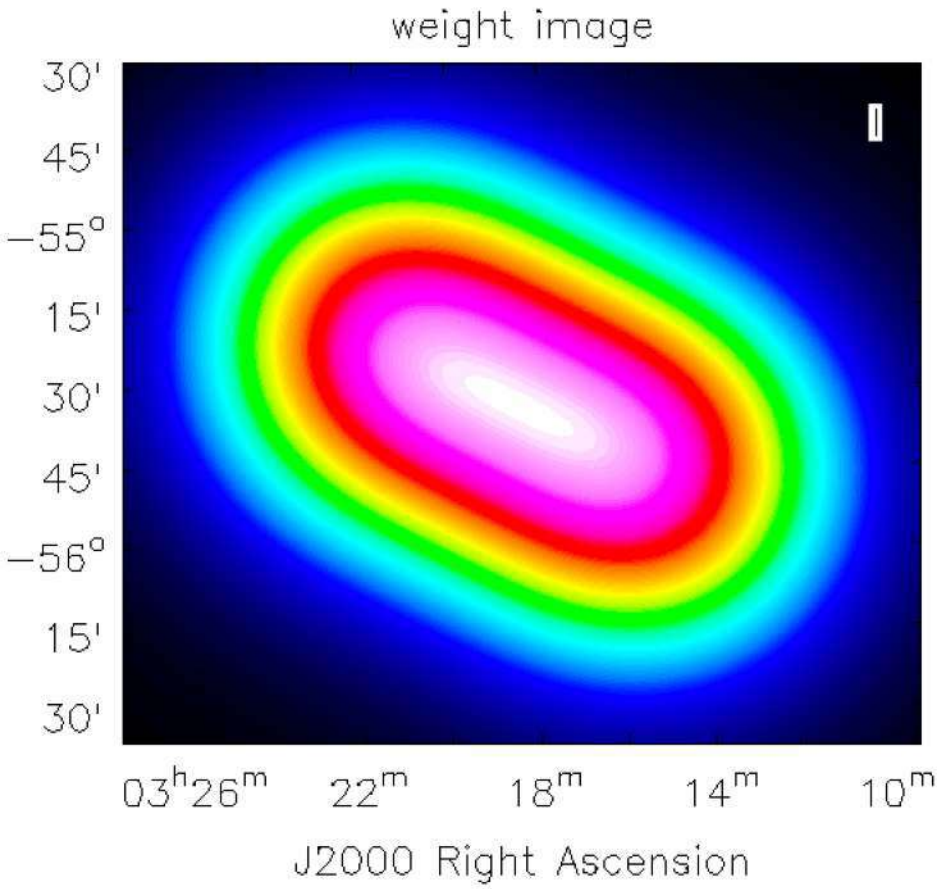
On local machine or remotevis.pawsey.org.au

$ casaviewer dir/image.I.COSMOLOGY_T15-2iter-0.linmosRAS_5177

- dir = $MY_OUTPATH/image/5177

casaviewer image.I.COSMOLOGY_T15-2iter-0.linmosRAS_5177





One loop of self-cal

- 2nd run: set BLOOP_SELFCAL=1 & ELOOP_SELFCAL=1: imaging with a selfcal update

```
$ . process_ASKAPdata.config
$ ./do_selfcal_ras.sh
$ ./do_linmos_ras.sh

$ ls -l $MY_OUTPATH/linmos/5177/

$ scp -r username@hpc-data.pawsey.org.au:$MY_OUTPATH/linmos/5177/\*iter-1\* .

$ casaviewer image.I.COSMOLOGY_T15-2iter-1.linmosRAS_5177
```