



# Calibration, the Weblog, scriptForPI, and running the Pipeline

---

*Luke Maud – ESO Garching - ALMA ARC Scientist*



Calibration, the Weblog, scriptForPI, and running the Pipeline, *European ALMA School at Leiden Observatory – 26.01.2026 to 30.01.2026*



# Before we begin

# What is the Pipeline?

*ALMA calibrate and image all user data*

- Pre-pipeline:
  - People did the calibration and imaging with scripts, like you've learned
  - All ARCs and Arc-Nodes involved
- **Pipeline:**
  - Developed over the past decade or more
  - Now does the calibration and imaging along with heuristics, plots and useful information
  - **Must work for 1000s of datasets**
  - Pipeline are pythonic tasks/stages that are based on CASA core tasks and tools

# Documentation and Guides

*The focus for these sessions will be to use and utilize the ALMA Pipeline*

Pipeline and CASA are both NA deliverables; they are responsible for documentation (with input from all Executives of ALMA, who provide developers and working group members)

- Pipeline: <https://almascience.eso.org/processing/science-pipeline>
  - [User Guide](#)
  - [User Task Reference Manual](#)
  - [Known Issues](#)
  - [Published Paper](#) (The ALMA Interferometric Pipeline Heuristics – Dated 2023 version)
- CASA: [https://casa.nrao.edu/casa\\_obtaining.shtml](https://casa.nrao.edu/casa_obtaining.shtml)
  - [Guides and Documentation](#)

# Documentation and Guides

The focus for these sessions will be to use and utilize the ALMA Pipeline

Tasks,  
Parameters,  
brief overview  
(doc strings)

Detailed  
Explanations

v.1.1 | Oct 2025

**User Support:**  
ALMA Science Pipeline User's Guide for  
Release 2025.1.0.35, CASA 6.6.6-17, python3.10  
Interferometric and Single-Dish Processing



**ALMA**



[www.almascience.org](http://www.almascience.org)

ALMA, an international astronomy facility, is a partnership of ESO (representing its member states), NSF (USA) and NINS (Japan), together with NRC (Canada), NSC and ASIAA (Taiwan), and KASI (Republic of Korea), in cooperation with the Republic of Chile. The Joint ALMA Observatory is operated by ESO, AUI/NRAO and NAOJ.

## Pipeline Tasks Reference Manual

*Release 2025.1.0*

pipeline team

Sep 29, 2025

# What do you want to get out of this tutorial?

*It's possible we cover all of it, but maybe not...*

Join at [menti.com](https://menti.com) | use code **8707 2812**

We'll use Mentimeter – there are **three choices per person**:

- To understand the weblog
- How to **restore** calibrated data
- How to do/re-do continuum subtraction (in Pipeline)
- How to use the pipeline self-calibration process
- How to totally **reprocess** data in the latest Pipeline
- How to use pipeline to re-image



# Getting data to work with as a user

*...this data looks interesting*

As a PI or as an Archive user these are the steps:

- Go via the Archive and search out some data
  - You can download some weblogs without the data – *see already if there is any detection*
- Select what you want to download
  - On the download page **make sure you tick the tarball for RAW data**
- Run the downloaded script
  - Follow the prompts to untar everything
- Investigate the folder structure
- Check the QA and/or weblogs locally
- Start “processing”



# Possible the data for these tutorials

*We can use three datasets, some partially processed*

- Datasets are:
  - 2013.1.00524.S – High Frequency (Nuclear Regions of Nearby LIRGs) – CGCG049
  - 2022.1.00974.S – Massive Star Forming Cores with Masers – G183.348
  - 2024.1.00408.S – Disk Population of Herbig Ae/Be – G304.664-00.966  
PI Miguel Vioque (*kindly allowed us to use these data – will be published next year*)
- I have prepared three tutorials for many different activities later:
  - Allow you to do a full Pipeline run
  - Allow you to do some step-by-step calibration stages
  - Allow you to do a findcont
  - Allow you to test a selfcal process
  - Allow you to do imaging

All ACA data to  
run fast

We'll come  
back and get  
data when we  
need it



**Let's all login and setup a work area**



# Setup as used for the Leiden School

# Setup a work area

Please follow exactly to login

- Allegro instructions
  - Go to this weblink: <https://almaportal.strw.leidenuniv.nl/>
  - Enter your username (`allegroquest<X>`) and password (`XXX`)
- This should not be your first login:
  - if it is, use a 2FA app to scan the code here:  
<https://home.strw.leidenuniv.nl/~alma/workshop/allegroquest<x>.png>
  - else, use you app to enter the 2FA code
- Now use the **Interactive Apps** to launch a desktop

<X> is your assigned user number

Click to launch the VNC

ALMA 4-CPU Workstation for workshop users (284) 1 node | 4 cores | Running

Host: almanode1.strw.leidenuniv.nl Cancel

Created at: 2026-01-20 15:36:37 CET

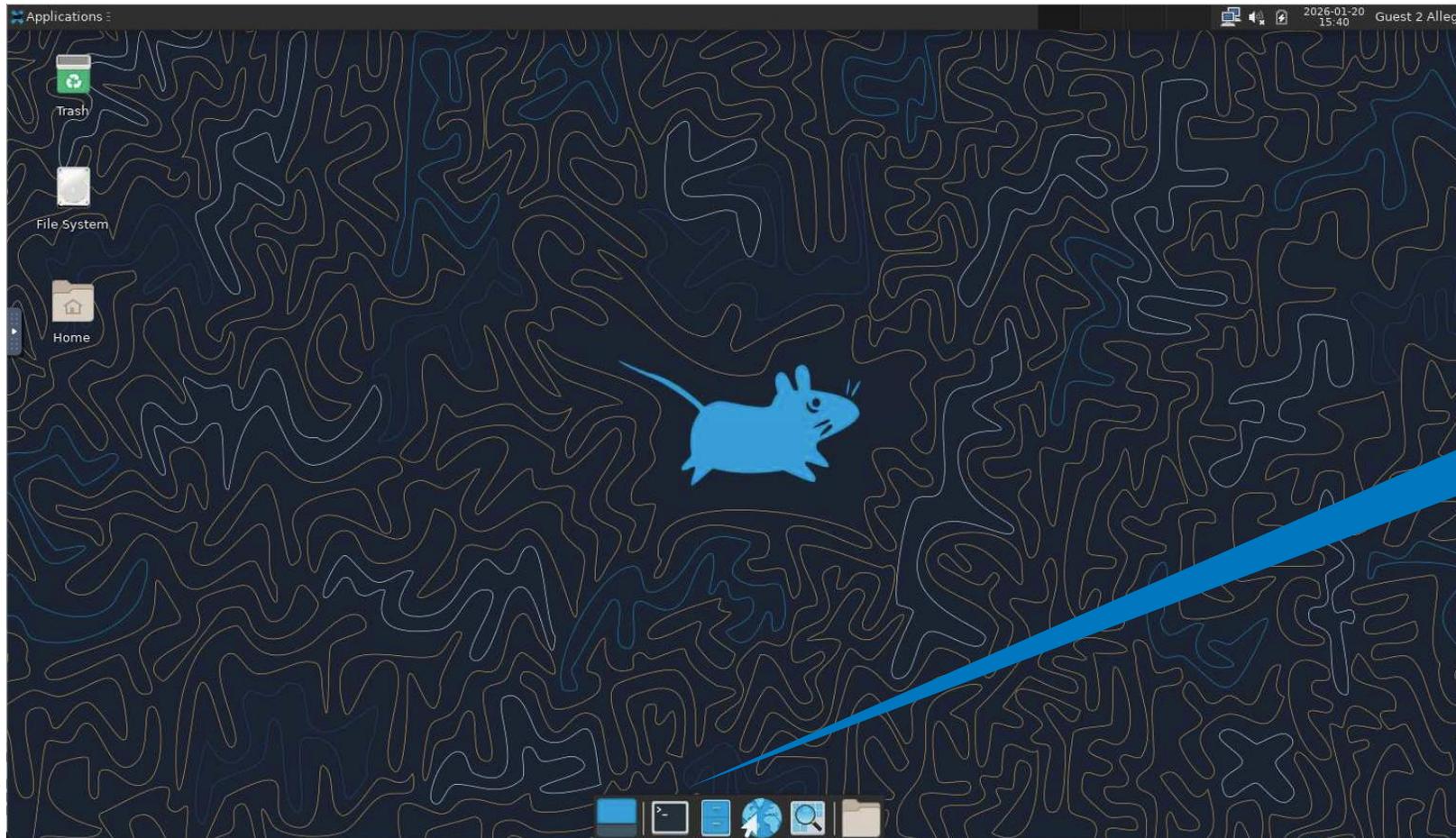
Time Remaining: 71 hours and 57 minutes

Session ID: 4ccd4f5d-e797-4ec7-b08c-59e9e5850b3f

Compression 0 (low) to 9 (high) Image Quality 0 (low) to 9 (high)

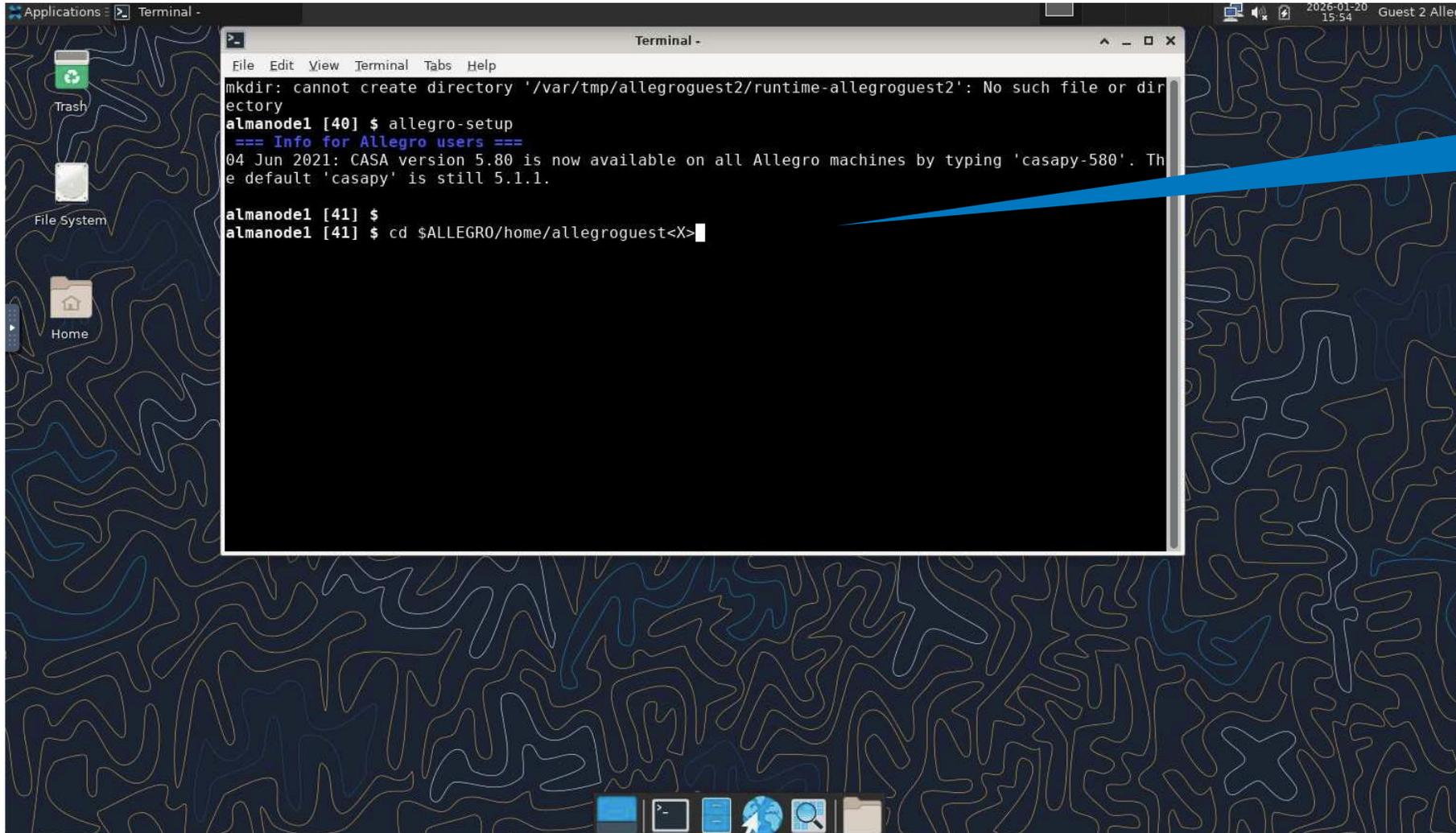
Launch ALMA 4-CPU Workstation for workshop users View Only (Share-able Link)

# Are you on a desktop session?



Click a  
TERMINAL and  
let's sort out the  
directories

# Setup commands



```
Applications | Terminal -
Terminal -
File Edit View Terminal Tabs Help
mkdir: cannot create directory '/var/tmp/allegroquest2/runtime-allegroquest2': No such file or directory
almanode1 [40] $ allegro-setup
=== Info for Allegro users ===
04 Jun 2021: CASA version 5.80 is now available on all Allegro machines by typing 'casapy-580'. The default 'casapy' is still 5.1.1.
almanode1 [41] $
almanode1 [41] $ cd $ALLEGRO/home/allegroquest<X>
```

<X> is your assigned user number

# Setup commands

```
Terminal -
File Edit View Terminal Tabs Help
almanode1 [55] $
almanode1 [55] $ ls
open_ALMA_School_2026 open_alma_school_2025
almanode1 [56] $
almanode1 [56] $ cd open_ALMA_School_2026/
almanode1 [57] $
almanode1 [57] $
almanode1 [57] $ ls
analysis data products
almanode1 [58] $
almanode1 [58] $
```

# Setup commands

```
Terminal -
File Edit View Terminal Tabs Help
almanode1 [59] $ ls
analysis data_products
almanode1 [60] $
almanode1 [60] $ cd analysis/allegroquest<X>
```

Here there will be a directory structure

<X> is your assigned user number

# Setup the working directories and copy data

*Please follow exactly or your directories will be messy and hard to navigate*

```
Terminal -
File Edit View Terminal Tabs Help
almanode1 [112] $ ls
tutorial1-intro_to_interferometry  tutorial5a-vlbi
tutorial2-imaging                  tutorial5b-pipelines
tutorial3a-self_calibration        tutorial6-proposals
tutorial3b-data_combination        tutorial7a-beyond_clean
tutorial4-carta                    tutorial7b-polarisation
almanode1 [113] $
almanode1 [113] $
```

These will be the top level directories

# Setup the working directories and copy data

*Please follow exactly or your directories will be messy and hard to navigate*

```
Terminal -
File Edit View Terminal Tabs Help
almanode1 [122] $
almanode1 [122] $
almanode1 [122] $
almanode1 [122] $
almanode1 [122] $ mkdir 2013.1.00524.S_top
almanode1 [123] $ mkdir 2022.1.00974.S_top
almanode1 [124] $ mkdir 2024.1.00408.S_top
almanode1 [125] $
```

Make these to transfer data into

# Setup the working directories and copy data

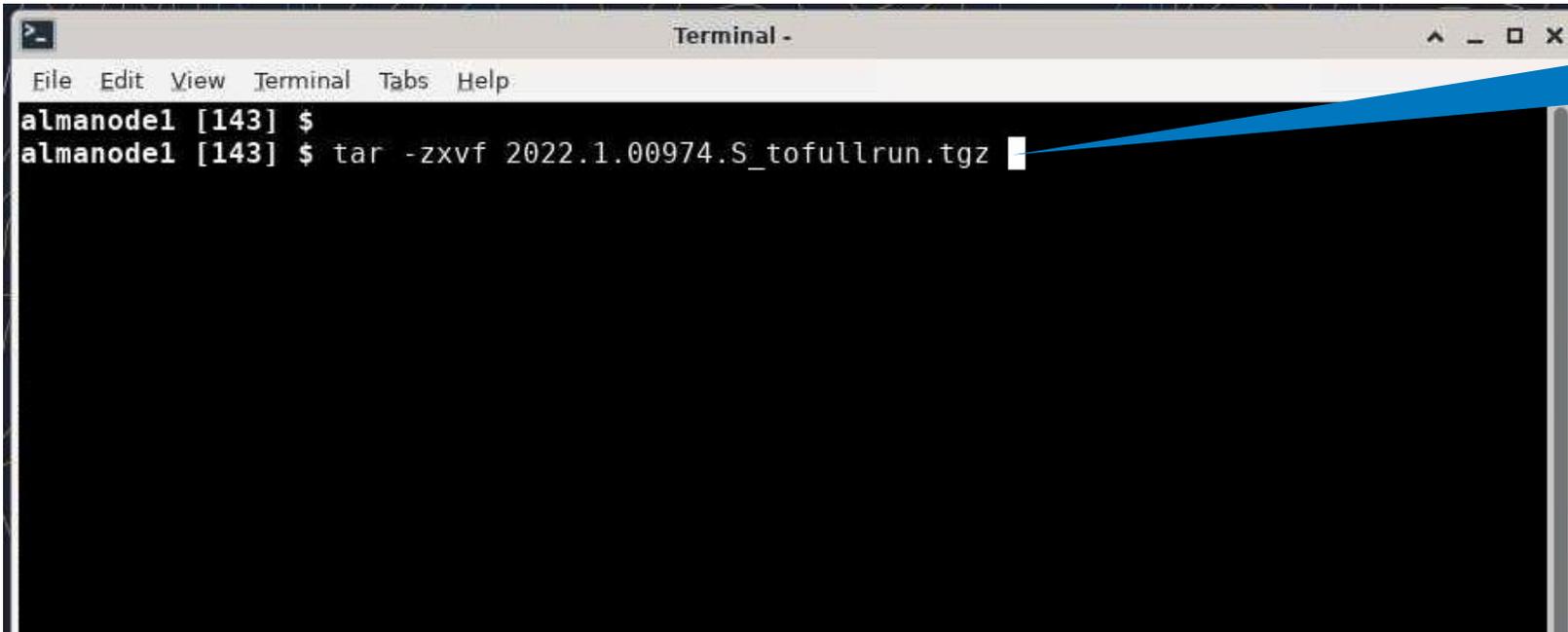
*Please follow exactly or your directories will be messy and hard to navigate*

```
Terminal -
File Edit View Terminal Tabs Help
almanode1 [141] $
almanode1 [141] $
almanode1 [141] $ cd 2022.1.00974.S_top/
almanode1 [142] $
almanode1 [142] $
almanode1 [142] $ cp ../../../../../../data_products/tutorial5b-pipelines/2022.1.00974* .
```

Copy the relevant tarbals

# Setup the working directories and copy data

*Please follow exactly or your directories will be messy and hard to navigate*

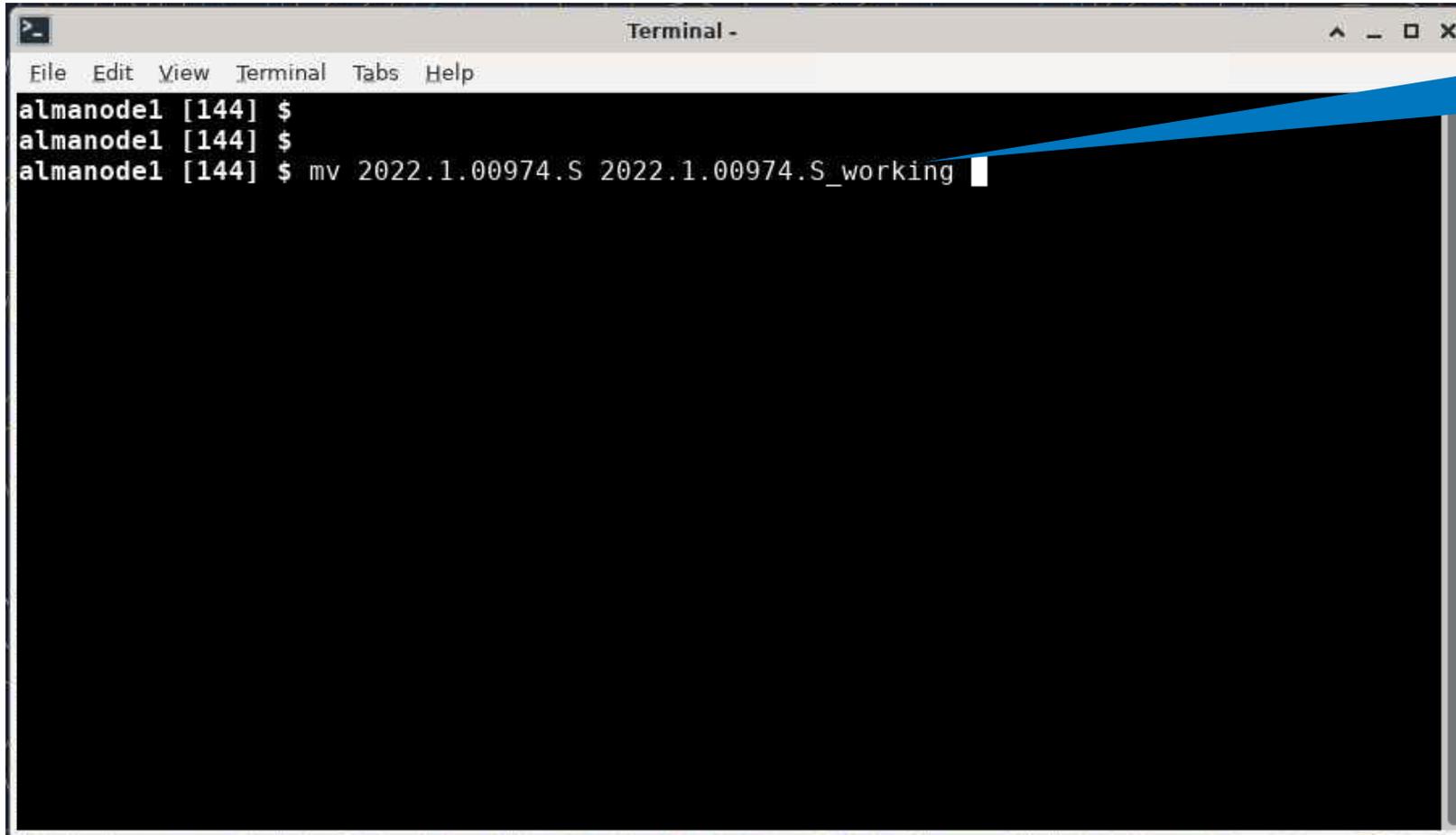


```
Terminal -
File Edit View Terminal Tabs Help
almanode1 [143] $
almanode1 [143] $ tar -zxvf 2022.1.00974.S_tofullrun.tgz
```

For now only  
untar this

# Setup the working directories and copy data

*Please follow exactly or your directories will be messy and hard to navigate*



```
Terminal -
File Edit View Terminal Tabs Help
almanode1 [144] $
almanode1 [144] $
almanode1 [144] $ mv 2022.1.00974.S 2022.1.00974.S_working
```

rename



# Setup on your own machine

# Get data from archive

*Follow relevant archive tutorials*

- **Download and unpack all data as shown in other tutorials**
  - Follow the helper script that will do all of this
  - All ALMA data follow the same data structure



**So, now let's have a quick overview of the project**

# File structure

If you are using your selected Archive data file names will be different

Use your own Archive Download or the example data linked below

- Move down the entire directory tree
  - This is “project code” level,
  - Into the “science goal”
  - Then the “group observing unit set **GOUS**”
  - Finally, “member observing unit set **MOUS**”
- The tree depends on what the original PI (or user) aimed to do with ALMA
  - Can be different bands, different array configurations, difference sources all mixed into various parts

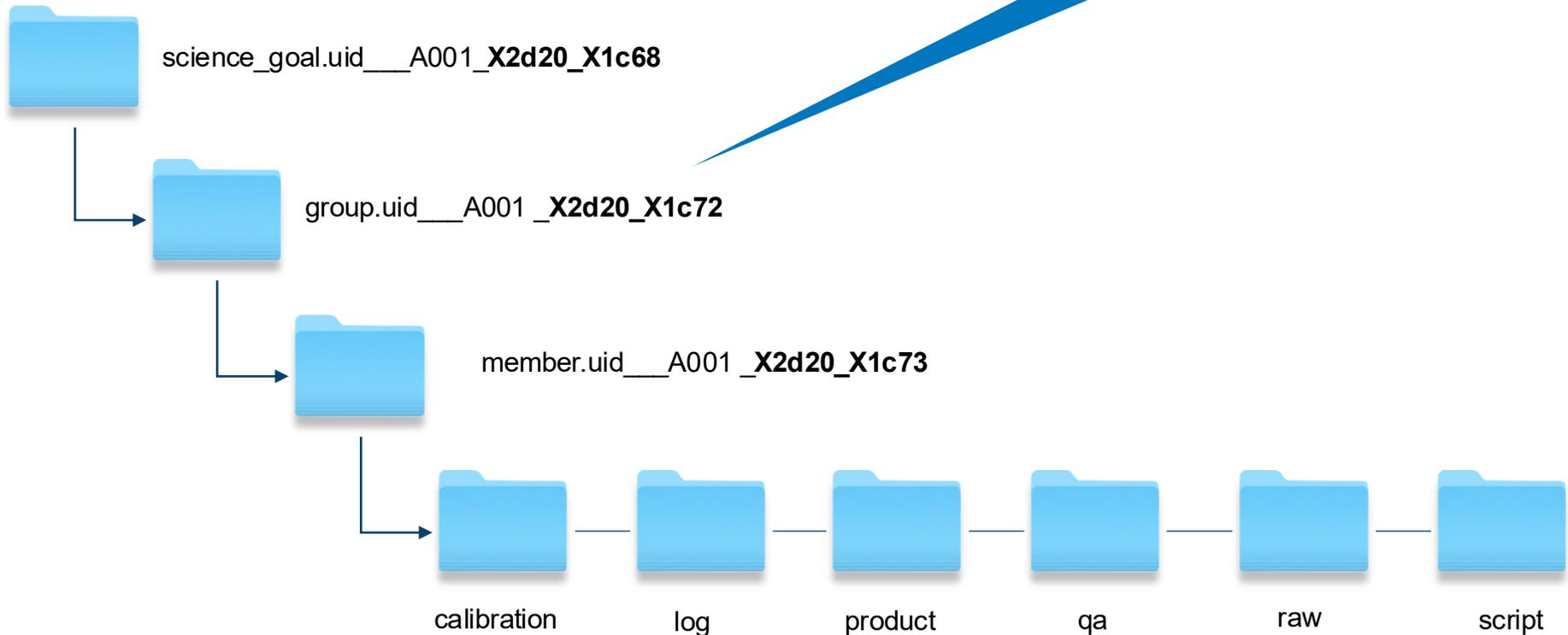
```
almanode1 [116] $
almanode1 [116] $ cd 2022.1.00974.S_working/
almanode1 [117] $
almanode1 [117] $ ls
science_goal.uid__A001_X2d20_X1c68
almanode1 [118] $ cd science_goal.uid__A001_X2d20_X1c68/
almanode1 [119] $
almanode1 [119] $ ls
group.uid__A001_X2d20_X1c72
almanode1 [120] $
almanode1 [120] $
almanode1 [120] $ cd group.uid__A001_X2d20_X1c72/
almanode1 [121] $
almanode1 [121] $ ls
member.uid__A001_X2d20_X1c73
almanode1 [122] $
almanode1 [122] $ cd member.uid__A001_X2d20_X1c73/
almanode1 [123] $
almanode1 [123] $
```

**NOTE:** Older ALMA data may not have used the ALMA pipeline but manual calibration scripts

# File structure

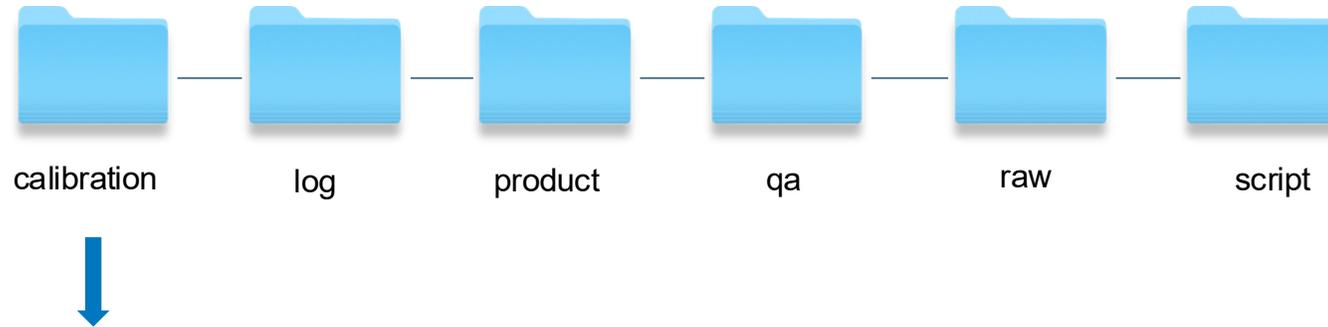
Use your own Archive Download or the example data linked below

If you are using your selected Archive data file names will be different



# File structure

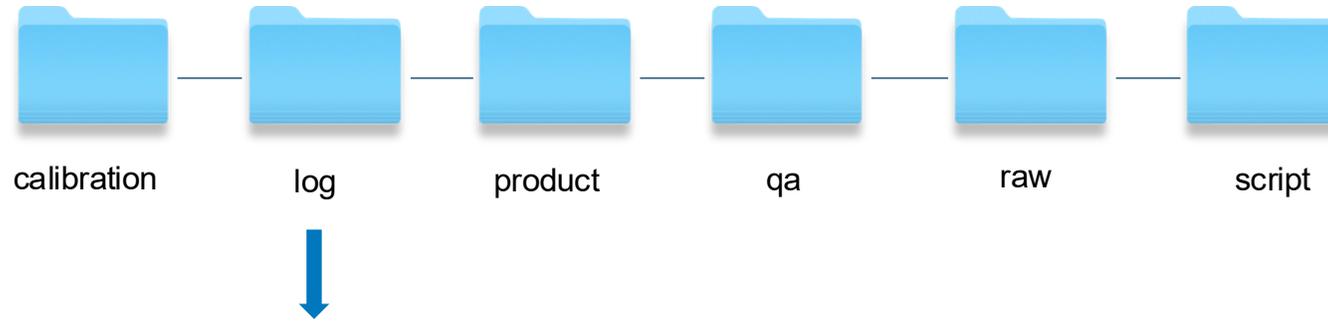
Use your own Archive Download or the example data linked below



- All calibration files done by ALMA
  - All as tarballs
  - Will be automatically untarred and moved in a **restore** process for Pipeline, or a re-run for old manually calibrated data
  - If run was with Pipeline there is a “calapply.txt” showing equivalent manual-like commands for reference only

# File structure

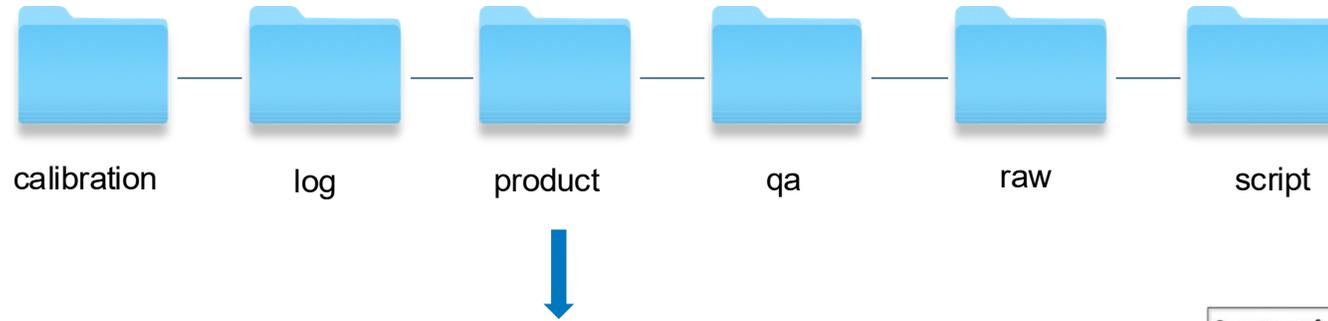
*Use your own Archive Download or the example data linked below*



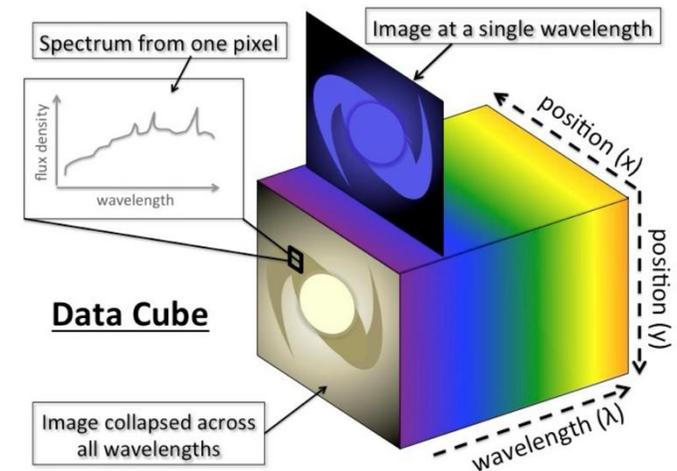
- Any logs made of the run
  - Might be calibration and imaging with ALMA pipeline in one log
  - Separate calibration and imaging
  - Other log files for manual calibrated data

# File structure

Use your own Archive Download or the example data linked below

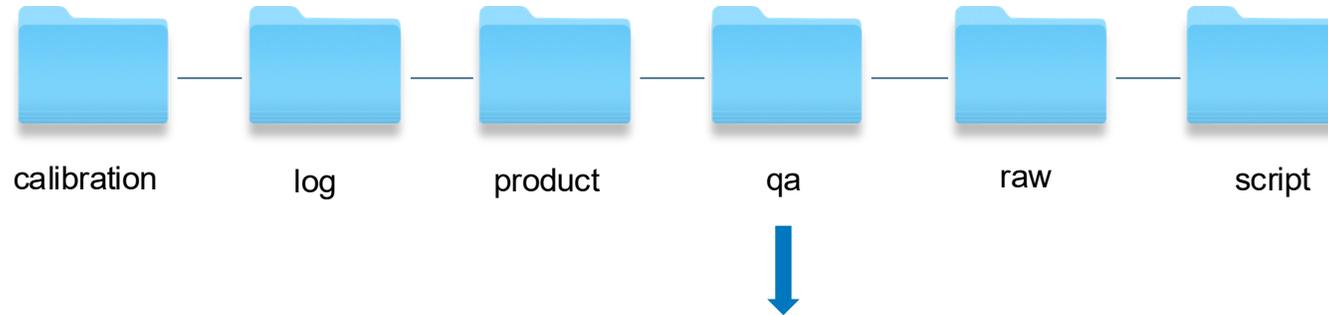


- All **image** products
  - FITS format images for the MOUS
  - Various Targets, SpWs
  - Types are MFS, CONT, CUBE (repBW)
  - Generally primary beam corrected and masks included
  - If Pipeline with selfcal, sometimes selfcal images



# File structure

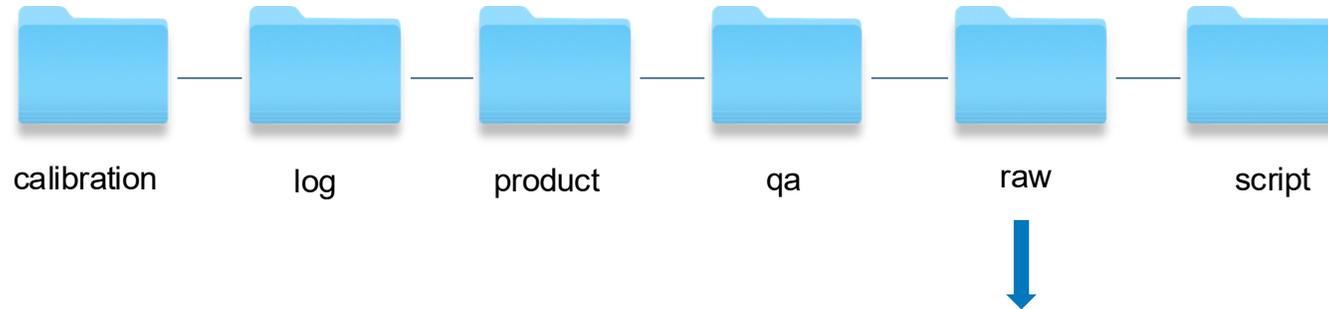
Use your own Archive Download or the example data linked below



- Reports from **quality assessment**
  - Newest data should include QA0 report
  - Tarball of the **weblog** from Pipeline processing
  - PNG or PDF for manually processed data

# File structure

Use your own Archive Download or the example data linked below

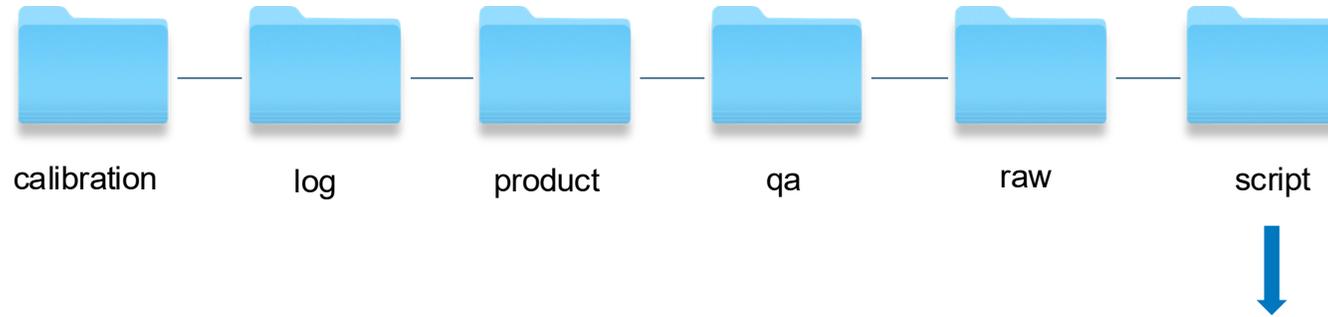


- Data

- The ASDMs (ALMA Science Data Model) are the raw data needed to be converted by CASA into measurement sets for calibration and imaging
- These are **required** if you want to do any type of reprocessing, imaging, restore etc

# File structure

Use your own Archive Download or the example data linked below



- Scripts

- **scriptForPI.py** is a wrapper to automatically do everything to get your data in a state to image
- **pipescirpt.py** and **piperestorescript.py** are Pipeline related scripts for **full run** and **restore** respectively
- XML files are ancillary from pipeline runs by ALMA
- Other .py could be from manual calibration or imaging



# What is this weblog?

# Overview of Quality Assurance

*The ALMA project guarantee the quality of the data – so have many QA steps*

- QA0:
  - Check if data are recorded correctly and all source/fields are included
  - Checks occur at the telescope in Chile
- QA2:
  - Data are calibrated and imaged
  - Must meet the requested **Beam Size** and **Noise RMS**
  - Up to October 2023 all data calibrated have always been checked by a human – data analyst (DA), and the data reduction manager (DRM)

Old data could be manually calibrated – or – with old PL and old weblogsc

# Weblog

## *Interface to the whole calibration and imaging process*

- In the recent years the majority (97% in the last 2 cycles) of ALMA data are calibrated and imaged by the ALMA pipeline
- The ALMA pipeline is **primarily** run at **JAO** (Joint ALMA Observatory) in Chile, but also the ALMA Regional Centers (**ARCs**) in Europe, North America, East Asia
- A weblog is produced as part of the Pipeline processing
  - Serves as the **interface** for QA, for the DA to the data
  - Shows **everything** about how data was **processed**, including plots and tables for the data quality to be assessed
  - Latest versions have trustworthy scores on a traffic light scheme (**Green**, **Blue**, **Yellow**, **Red**)

**NOTE: Older ALMA data may not have used the ALMA pipeline NO WEBLOGS**

# Weblog – why should you care

*First look at data and processing – is everything what you wished for...*

- Good calibration is the root of good imaging
  - Possible calibration error could lead to imaging errors
- Pipeline has continually evolved, better heuristics and processing logic
  - Old weblogs might be limited or hint at things that could be improved
- Is there something useful in the data
  - Images and spectra can be seen along with beam and noise parameters

Given ALMA's QA process, everything should be good, to a level not adversely effecting Science Output

i.e. the point of these tutorials. - showing you how to redo

# Weblog – take a look

*Is there anything interesting inside?*

- Main landing page – **HOME** – top taps govern main navigation
- All things in **BLUE** are a link

The weblog will show you everything you need to know, and more

The screenshot shows the ALMA web interface. At the top, there is a navigation bar with 'Home' highlighted in a green box, 'By Topic', 'By Task', and 'Main Tabs'. The version number '2016.1.00484.L' is in the top right. Below the navigation bar, there are three main sections:

- Observation Overview:** A table with fields: Project (uid://A001/X5ac/X43f), Principal Investigator (sandrews), OUS Status Entity id (uid://A001/Xbd4641/X23), Observation Start (2017-05-09 04:28:16 UTC), Observation End (2017-05-09 05:23:29 UTC), and Number of Execution Blocks (1).
- Pipeline Summary:** A table with fields: Pipeline Version (2023.1.0.124 (documentation)), CASA Version (6.5.4.9 (environment)), IERSeop2000 Version (0001.0179 (last date: 2024-02-08 00:00:00)), IERSpredict Version (0623.1483 (last date: 2024-06-07 00:00:00)), Pipeline Start (2024-03-11 10:34:25 UTC), and Execution Duration (1 day, 6:48:49).
- Observation Summary:** A table with columns: Measurement Set, Receivers, Num Antennas, Time (UTC) (Start, End, On Target), Baseline Length (Min, Max, RMS), and Size. It includes a status bar and three rows of observation data.



Project Code

2016.1.00484.L

## Observation Overview

Project	uid://A001/X5ac/X43f
Principal Investigator	sandrews
OUS Status Entity id	uid://A001/Xbd4641/X23
Observation Start	2017-05-09 04:28:16 UTC
Observation End	2017-05-09 05:23:29 UTC
Number of Execution Blocks	1

PI

Dates

## Pipeline Summary

Pipeline Version	2023.1.0.124 (documentation)
CASA Version	6.5.4.9 (environment)
IERSseop2000 Version	0001.0179 (last date: 2024-02-08 00:00:00)
IERSpredict Version	0623.1483 (last date: 2024-06-07 00:00:00)
Pipeline Start	2024-03-11 10:34:25 UTC
Execution Duration	1 day, 6:48:49

Versions

## Observation Summary

Measurement Set	Receivers	Num Antennas	Time (UTC)			Baseline Length			Size
			Start	End	On Target	Min	Max	RMS	
Observing Unit Set Status: uid://A001/Xbd4641/X23 Scheduling Block ID: uid://A001/Xbd4641/X17									
Session: ... Build Version: 201608-CYCLE4-ON-R-2017-04-26-00-00-00									
uid__A002_Xc02418_X29c8.ms		16	2017-05-09 05:23:29		0:25:19	15.1 m	1.1 km	476.0 m	32.6 GiB
uid__A002_Xc02418_X29c8_targets.ms	ALMA Band 6	45	2017-05-09 04:46:58	2017-05-09 05:22:08	0:25:12	15.1 m	1.1 km	476.0 m	17.1 GiB
uid__A002_Xc02418_X29c8_targets_line.ms	ALMA Band 6	45	2017-05-09 04:46:58	2017-05-09 05:22:08	0:25:12	15.1 m	1.1 km	476.0 m	17.1 GiB

the Measurement Set

useful parameters



### QA Scores: Lowest by Topic

## Quick overview of low scores

Topic	Lowest Score	Task
Data Sets	26	hifa_renorm: Renorm
Calibration	17	hifa_timegaincal: Gain calibration
Flagging	19	hif_applycal: Apply calibrations from context
Imaging	23	hif_makeimages: Make check source images
Miscellaneous	5	hif_refant: Select reference antennas

The stages that pipeline identified as lowest



Overview of worst scores – in this one, nothing is yellow or red

### Task Notifications: Warnings and Errors

Stage	Task	Message
2	hifa_flagdata	Warning: Undefined representative bandwidth for data set uid__A002_Xc02418_X29c8.ms
31	hif_findcont	Warning: Undefined representative bandwidth for data set uid__A002_Xc02418_X29c8.ms

stages with warnings

### Flagging Summaries

uid\_\_A002\_Xc02418\_X29c8.ms

flagging overview

Flagging percentages for Source name: AS\_209, Intents: ATMOSPHERE,TARGET

spw	DA41	DA42	DA44	DA45	DA46	DA47	DA48	DA49	DA51	DA52	DA53	DA54	DA55	DA56	DA57	DA58	DA59	DA60	DA61	DA62	DA63	DA65	DV01	DV03	DV04	DV05	DV06	DV07
19	43.157	43.966	43.630	43.238	44.496	44.077	44.380	42.660	44.068	43.665	44.541	44.203	44.676	43.869	43.768	43.215	44.043	56.505	44.126	44.135	43.872	44.595	43.478	44.438	44.523	100.000	43.999	
21	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	39.129	40.082	39.129	39.129
23	41.899	41.899	41.899	41.899	41.899	41.899	41.899	41.899	41.899	41.899	41.899	41.899	100.000	41.899	41.899	41.899	41.899	41.899	41.899	41.899	41.899	41.899	41.899	41.899	41.899	42.808	100.000	41.899
25	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	2.347	3.876	2.347	2.347

Flagging percentages for Source name: J1517-2422, Intents: ATMOSPHERE,BANDPASS,POINTING,WVR

spw	DA41	DA42	DA44	DA45	DA46	DA47	DA48	DA49	DA51	DA52	DA53	DA54	DA55	DA56	DA57	DA58	DA59	DA60	DA61	DA62	DA63	DA65	DV01	DV03	DV04	DV05	DV06	DV07
19	39.713	40.569	44.616	39.799	44.775	42.158	43.931	42.073	42.146	41.694	47.794	40.826	44.053	43.234	47.562	43.283	41.241	44.799	57.062	44.823	44.885	41.816	42.574	41.522	43.882	41.694	100.000	43.454
21	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	35.446	36.913	35.446
23	38.380	38.380	38.380	38.380	38.380	38.380	38.380	38.380	38.380	38.380	38.380	38.380	100.000	38.380	38.380	38.380	38.380	38.380	38.380	38.380	38.380	38.380	38.380	38.380	38.380	38.380	100.000	38.380
25	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222	2.222





# Task Summaries

Task	QA Score	Duration
1. <b>hifa_importdata</b> : Register measurement sets with the pipeline	1.00	
2. <b>hifa_flagdata</b> : ALMA deterministic flagging	1.00	0:41:03
3. <b>hifa_fluxcalflag</b> : Flag spectral features in solar system flux calibrators	1.00	0:00:03
4. <b>hif_rawflagchans</b> : Flag channels in raw data	1.00	0:07:53
5. <b>hif_refant</b> : Select reference antennas	1.00	0:00:23
6. <b>h_tsyscal</b> : Calculate Tsys calibration	1.00	0:07:19
7. <b>hifa_tsysflag</b> : Flag Tsys calibration	1.00	0:09:24
8. <b>hifa_antpos</b> : Correct for antenna position offsets	Nonzero antenna position offsets 0.90	0:00:06
9. <b>hifa_wvrgcalflag</b> : Calculate and flag WVR calibration	1.00	0:17:46
10. <b>hif_lowgainflag</b> : Flag antennas with low gain	1.00	0:10:00
11. <b>hif_setmodels</b> : Set calibrator model visibilities	1.00	0:09:32
12. <b>hifa_bandpassflag</b> : Phase-up bandpass calibration and flagging	0.99	0:25:24
13. <b>hifa_bandpass</b> : Phase-up bandpass calibration	0.99	0:18:17
14. <b>hifa_spwphaseup</b> : Spw phase offsets calibration	1.00	0:05:15
15. <b>hifa_gfluxscaleflag</b> : Phased-up flux scale calibration + flagging	1.00	0:25:08
16. <b>hifa_gfluxscale</b> : Transfer fluxscale from amplitude calibrator	1.00	0:23:34
17. <b>hifa_timegaincal</b> : Gain calibration	Potential phase offset outliers 0.80	0:33:53
18. <b>hifa_targetflag</b> : Target outlier flagging	1.00	0:14:44
19. <b>hif_applycal</b> : Apply calibrations from context	Phase vs frequency slope outliers 0.90	0:43:48

"hifa" – ALMA specific interferometric stages

"h" – Pipeline stage can be used for any instrument, single dish, or interferometry

"hif" –specific interferometric stages

# Weblog – take a look

*Is there anything interesting inside?*

- **Show of hands** – have you looked at a Weblog in detail before ?
- Open a browser and navigate to this weblog:

**[https://almascience.eso.org/arcdistribution/LeidenSchool/G183\\_QA2\\_delivered\\_weblog/html/](https://almascience.eso.org/arcdistribution/LeidenSchool/G183_QA2_delivered_weblog/html/)**

Links will be deleted mid-2026, you can download the weblog from the ALMA archive

# Weblog – take a look

*Is there anything interesting inside?*

- Let's look for some things:
  - Names of the **traditional calibration** stages
  - What **stage** (in order) can you first see if molecular lines are detected in the source?
  - Did any Spectral Windows (SpWs) have **a low signal to noise**?
  - Was **self-calibration** performed?

# Weblog – take a look

*Is there anything interesting inside?*

- Let's look for some things:
  - Names of the **traditional calibration** stages

Note, all stages in Pipeline are required to do calibration, they are running flagging and heuristic setups, NOT, just these few “traditional” stages

***h\_tsyscal** (combined with **hifa\_tsysflag**): Tsys gaintable and flags outliers*

***hifa\_wvrgcalflag**: makes the WVR solutions and flags bad antennas*

***hifa\_bandpass**: makes the bandpass solutions*

***hifa\_gfluxscale**: corrects the flux scaling using an amplitude calibrator*

***hifa\_timegaincal**: does all temporal gains, amp and phase*

***hif\_applycal**: applies all previously made solutions*

# Weblog – take a look

*Is there anything interesting inside?*

- Let's look for some things:
  - What **stage** (in order) can you first see if molecular lines are detected in the source?

## hif\_applycal

- **hif\_applycal**: sorts and applies all calibrations to the calibrators and targets
- Many plots are made of phase, amplitude, uvdistance, frequency
- Some amp vs. freq. could show molecular lines in the target, but it's a average of the visibilities
- Take care as more noise (positive amplitude noise) near ATM features (low transmission) look like lines – but are not

# This stage is preferable though



## Weblog – take a look

*Is there anything interesting inside?*

- Let's look for some things:
  - What **stage** (in order) can you first see if molecular lines are detected in the source?

## hif\_findcont

- **hif\_findcont**: this is the first stage to **actually run target imaging**
- It makes some dirty cubes, behind the scenes, and checks channel by channel for molecular lines
- This is required so Pipeline knows later what to use for continuum subtraction

# Weblog – take a look

*Is there anything interesting inside?*

- Let's look for some things:
  - Did any Spectral Windows (SpWs) have **a low signal to noise**?

# YES

- **hifa\_spwphaseup**: this stage governs how calibrations proceed, if there is low SNR a SpW can be mapped to another one, or if many SpWs have low SNR they can all be combined
- **SpW 28** is low SNR (16.5 where 32.0 is required and is mapped to 30)
- Since **PL2025** (released Oct 2025) all INTENTs (i.e. calibrator fields) can be assessed for mapping and combine

# Weblog – take a look

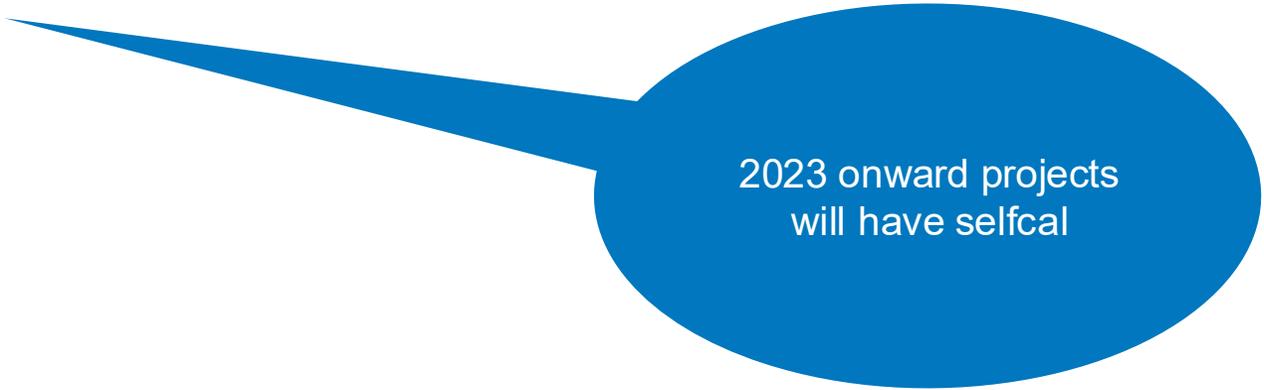
*Is there anything interesting inside?*

- Let's look for some things:
  - Was **self-calibration** performed?

**NO**

- **hif\_selfcal**: global interferometric task for self-calibration **available from Cycle 10 only**

**G183 is older data, 2022 project, the Pipeline used does not have self-cal, or latest heuristics - we can reprocess!!!**

A blue speech bubble with a tail pointing towards the main text.

2023 onward projects  
will have selfcal

# Weblog – other ways to look

*Is there anything interesting inside?*

- ALMA Archive is now providing direct access to some weblogs



Previews for AGAL\_G304.664-00.966

ALMA

[README](#) [QA2 report](#) [Weblog](#)

**Combined continuum** : 215.571.234.946GHz, kHz, XX YY  
 member\_uid\_\_A001\_X3788\_Xctb0\_AGAL\_G304.664-00.966\_sci.spw24\_26\_28\_30.cont.l.pbcor.fits 56 KB

Band: 6  
 Frequency type: line  
 Frequency range: 215.571.234.946  
 Continuum sensitivity: 0.187  
 Line sens. (10km/s): 2.584 mJy/beam  
 Line sens. (native): 0.187 uJy/beam  
 Polarisations: XX YY  
 Array: 7m

**SPW 0** : 215.571.217.445GHz, 1,128.906 kHz, XX YY  
 member\_uid\_\_A001\_X3788\_Xctb0\_AGAL\_G304.664-00.966\_sci.spw24.cube.l.selfcal.pb.fits.gz 928 KB

Band: 6  
 Frequency type: line  
 Frequency range: 215.571.217.445  
 Frequency resolution: 1,128.906 kHz  
 Continuum sensitivity: 0.187  
 Line sens. (10km/s): 5.283 mJy/beam  
 Line sens. (native): 0.328 uJy/beam  
 Polarisations: XX YY

2024.1.00408.S	V_star_RR_Tau	05:39:30.509 +26:22:26.85 6	0.2027	215.563.234.936 GHz	2025-12-03	0	5.592	0.092	7m	30.713	44.316
2024.1.00408.S	EM_star_LkHA_338	06:10:47.124 -06:12:50.59; 6	0.3671	215.621.234.998 GHz	2025-12-04	0	4.768	0.092	7m	29.497	44.304
2024.1.00408.S	Gaia_DR3_59616908...	17:34:04.624 -39:23:41.391 6	0.4640	215.572.234.948 GHz	2025-12-04	0	4.653	0.092	7m	31.285	44.315
2024.1.00408.S	V_star_V486_Sco	17:34:37.860 -32:36:19.291 6	0.4636	215.571.234.946 GHz	2025-12-04	0	4.593	0.092	7m	32.211	44.314

Click to download from Archive

# Weblog – other ways to look

*Is there anything interesting inside?*

- ALMA Archive is now providing direct access to **some** weblogs

- Click to download

- Untar

```

Downloads — -bash — 115x5
-rw-r--r--@  1 lmaud  5000   100224283 Jan 16 17:40 member.uid___A001_X3788_Xc1b0.hifa_calimage.weblog.tgz
[MLH2P7LNHNY:Downloads lmaud$
[MLH2P7LNHNY:Downloads lmaud$
[MLH2P7LNHNY:Downloads lmaud$
[MLH2P7LNHNY:Downloads lmaud$ tar -zxvf member.uid___A001_X3788_Xc1b0.hifa_calimage.weblog.tgz

```

- Run **ls -ltr** to see last file name

- Move to the “html” folder

- Open via local server with Python

```

html — Python -m http.server 8085 --bind 127.0.0.1 — 116x7
-rw-r--r--@  1 lmaud  5000   100224283 Jan 16 17:40 member.uid___A001_X3788_Xc1b0.hifa_calimage.weblog.tgz
drwxr-xr-x   3 lmaud  5000         96 Jan 16 17:44 pipeline-20241203T120736
[MLH2P7LNHNY:Downloads lmaud$ cd pipeline-20241203T120736
[MLH2P7LNHNY:pipeline-20241203T120736 lmaud$ cd html/
[MLH2P7LNHNY:html lmaud$ python3 -m http.server 8085 --bind 127.0.0.1
Serving HTTP on 127.0.0.1 port 8085 (http://127.0.0.1:8085/) ...

```

Port “8085” can be changed, if doing more than one weblogs



# What are our options for processing?



# Options on what to

*Right into science, or do I need to reconsider the calibration or imaging?*

- How to **restore** calibrated data
- How to do/re-do continuum subtraction (in Pipeline)
- How to use the pipeline self-calibration process
- How to totally **reprocess** data in the latest Pipeline
- How to use pipeline to re-image



# What to keep in mind

*Which CASA? Do I need a pipeline version?*

- There is a compatibility matrix for both CASA and the ALMA Pipeline

PROJECT CODE	CYCLE	CASA	Release	
<=2013	0,1,2	4.4.0, 4.3.1, 4.2.2, 4.2.1, 4.2.0, 3.4, 3.3.	Redhat, Mac OS 10.7, 10.8	later some cal PL
2015	3	4.6.0, 4.5.2, 4.5.1	RH 5	
2016	4	4.7.2, 4.7.0-1, 4.7.0	RH6/7, MacOS 10.10, 10.11	image PL too
2017	5	5.4.0-70, 5.4.0-68, 5.3.0, 5.1.1	RH6/7 MacOS 10.12	
2018	6	5.6.1-8, 5.4.0-70	RH6/7, Mac OS 10.13, 10.14	
2019	7	6.1.1-15, 5.6.1-8	RH 6-9, Mac OS 10.15	
2021	8	6.4.1-12, 6.2.1-7	RH 6-9, Mac OS 11	
2022	9	6.4.1-12 + patch	RH 6-9, Mac OS 11	
2023	10	6.5.4-9	RH 6-9, Mac OS 11	
2024	11	6.6.1-17	RH 6-9, Mac OS 12	
2025	12	6.6.6-17	RH 6-9, Mac OS 13, 14	

Pipeline now has selfcal, these are best versions

Most manually calibrated, no Pipeline, no weblog

Pipeline here onwards is really very good

Pipeline has best lowSNR heuristics



# What to keep in mind

## Which CASA? Do I need a pipeline version?

- There is a compatibility matrix for both CASA and the ALMA Pipeline

CASA versions accepted for ALMA data processing:

CASA version	Pipeline branch and version	Pipeline Documentation	Description	used in operations	tarball for most modern OS available for each	versions that can be used to restore these data <sup>1</sup>
6.6.6-17	2025.1.0.35	<a href="#">User's Guide</a> <a href="#">Reference Manual 2025</a> <a href="#">Pipeline 2025 known issues</a>	Cycle 12	2025-09-29	<a href="#">casa-6.6.6-17-pipeline-2025.1.0.35-py3.10.el8.tar.xz</a> <a href="#">casa-6.6.6-17-pipeline-2025.1.0.35-14.0-arm64-py310.dmg</a>	most recent
6.6.1-17	2024.1.0.8	<a href="#">User's Guide</a> <a href="#">Reference Manual 2024</a> <a href="#">Pipeline 2024 known issues</a>	Cycle 11	2024-09-30 ~ 2025-09-29	<a href="#">casa-6.6.1-17-pipeline-2024.1.0.8-py3.8.el8.tar.xz</a> <a href="#">casa-6.6.1-17-pipeline-2024.1.0.8-12.0-py38.dmg</a>	most recent
6.5.4-9	2023.1.0.124	<a href="#">User's Guide</a> <a href="#">Reference Manual 2023</a> <a href="#">Pipeline 2023 known issues</a>	Cycle 10	2023-09-30 ~ 2024-09-29	<a href="#">casa-6.5.4-9-pipeline-2023.1.0.124-py3.8.tar.xz</a> <a href="#">casa-6.5.4-9-pipeline-2023.1.0.124-11.0-py38.dmg</a>	most recent
6.4.1-12	2022.2.0.68	<a href="#">User's Guide</a> <a href="#">Reference Manual 2022.2</a> <a href="#">Pipeline 2022 known issues</a>	Cycle 9 patch	2023-04-18 ~ 2023-09-30	<a href="#">casa-6.4.1-12-pipeline-2022.2.0.68-py3.6.tar.xz</a> <a href="#">casa-6.4.1.12-pipeline-2022.2.0.68-11.0-py36-py36.dmg</a>	most recent
6.4.1-12	2022.2.0.64	<a href="#">User's Guide</a> <a href="#">Reference Manual 2022.2</a> <a href="#">Pipeline 2022 known issues</a>	Cycle 9	2022-09-27 ~ 2023-04-18	<a href="#">casa-6.4.1-12-pipeline-2022.2.0.64-py3.6.tar.xz</a>	most recent
6.2.1-7	2021.2.0.128	<a href="#">User's Guide</a> <a href="#">Reference Manual 2021.2</a> <a href="#">Pipeline 2021 known issues</a>	Cycle 8	2021-10-01 ~ 2022-09-26	<a href="#">casa-6.2.1-7-pipeline-2021.2.0.128.tar.xz</a> <a href="#">casa-6.2.1.7-pipeline-2021.2.0.128-10.15-py36.dmg</a>	most recent
6.1.1-15	2020.1.0-40	<a href="#">Users Guide 2020.1</a> <a href="#">Reference Manual 2020.1</a> <a href="#">Pipeline 2020 known issues</a>	Cycle 7 reprise	2021-05-10 ~ 2021-10-01	<a href="#">casa-6.1.1-15-pipeline-2020.1.0.40.tar.xz</a> <a href="#">casa-6.1.1.15-pipeline-2020.1.0.40-10.15.dmg</a>	casa-6.1.1-15-pipeline-2020.1.0.40 or most recent
5.6.1-8	Pipeline-CASA56-P1-B, r42866	<a href="#">Users Guide 5.6.1</a>	Cycle 7	2019-10-01 ~ 2021-05-10	<a href="#">casa-pipeline-release-5.6.1-8.el7.tar.gz</a> <a href="#">casa-pipeline-release-5.6.1-8-10.14.dmg</a>	casa-pipeline-release-5.6.1
5.4.0-70	Pipeline-CASA54-P1-B, r42254 and r42866	<a href="#">Users Guide 5.4</a> <a href="#">Reference Manual 5.4</a>	Cycle 6	2018-10-01 ~ 2019-10-01	<a href="#">casa-release-5.4.0-70.el7.tar.gz</a>	casa-pipeline-release 5.4.0 or 5.6.1

Older means then old CASA and old Pipeline

# Options on what to do – (1)

*Get to the imaging as fast as possible*

To get to imaging we need to **restore** the previous calibration **done by ALMA**

- **scriptForPI.py** is the wrapper that does all the work
- You can only do a restore with **compatible versions** of CASA and Pipeline
  - Likely need older OS machine for anything but latest data – **this can be a problem**
  - Can use the **CaIMS** service
  - Pipeline from circa 2021-2022 with CASA 6.2.1 **can** restore using **newer** versions (*care with options for e.g. dosplit, docontsub*)

# Options on what to do – (1)

*Get to the imaging as fast as possible*

To get to imaging we need to **restore** the previous calibration **done by ALMA**

- For a **Pipeline** run, **you only restore** existing calibration tables. You can flag before imaging, but you have **no access to any of the calibration process**
- **Nuance** – **manually calibrated data** runs the calibration script, again, in CASA
  - This remakes and applies calibration
- After a restore, you **can use the latest** CASA and Pipeline for imaging
  - If using Pipeline stages for imaging (very) old data, there may be some warnings or necessity to hard code inputs

## Options on what to do – (2)

### *Improve the calibration and flagging*

This means you want to **reprocess** the data, effectively **redoing** the QA2 from ALMA

- **scriptForPI.py** can help with this for **Pipeline** processed data
- You might have to be careful of some required input files (e.g. getting correct source reference fluxes)
  - If you use the older CASA and Pipeline version, **helper** scripts and files need to be copied
  - When using the scriptForPI.py it will do this copying
- *For manual calibration you can edit the calibration script, but you are fixed to the old CASA versions due to format structure and options*
- Best option is to **reprocess with the latest ALMA Pipeline**

# Options on what to do – (3)

*Want to manually improve the continuum selection*

This means you are happy with the previous calibration but want to edit the continuum ranges

- **hif\_findcont** is part of the **imaging** stages
  - You can look at the existing weblog and judge the continuum and lines
  - You can investigate the existing images to judge the continuum and lines
- You can make a new, or edit an existing **cont.dat** file
  - Care must be taken as the units in frequency are LSRK
  - You can start from restored data, or Pipeline calibrated data (without imaging), or even add the cont.dat for a full Pipeline reprocessing (*findcont will be skipped in this case*)

# Options on what to do – (4)

*Only try to do self-calibration in Pipeline*

This means you are happy with the previous calibration but want to do automatic self-calibration

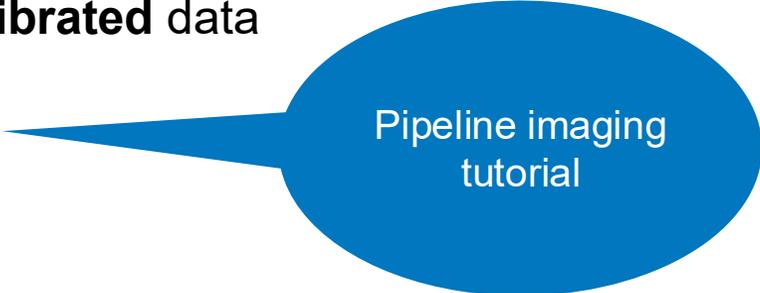
- **hif\_selfcal** is part of the **imaging** stages
  - You can use restored data to image and do self-calibration
  - Data earlier than 2021 might not work properly – self-cal needs some metadata that might not exist
    - You can test this first and see if selfcal completes
    - another option is to **reprocess** with the latest ALMA Pipeline
- You can edit some self-cal options as to not be limited by “PI” requirements
  - E.g. the beam may change during self-calibration, it is a problem for QA2, but maybe not for your science

# Options on what to do – (5)

## *Image only with the Pipeline*

Not really interested in the calibration parts

- You **must** have run a **restore** or **reprocessing** to have the **calibrated** data on disk
- You can use the **calMS** service and download the copy of the **calibrated** data
- You have to follow (or script) the set pipeline imaging tasks



Pipeline imaging tutorial



**Move to TUTORIAL - *restore***



# The Pipeline Context System



**Wait...what? Context....**

# What is the Pipeline Context?

*The Pipeline tracks where it is, and what was done*

The ALMA Pipeline must store information about the data processing and imaging

- The **context** is not an official interface, it is used internally by the pipeline
  - i.e. the Pipeline developers and working group can change parameters with various releases
- The **context** is a Domain Object
  - Stores metadata about the dataset, spectral setup, fields etc.
  - Stores information about running stages
  - Stores information about calibration tables

# The Context and the Pipeline

*There is a sequence, and messy things happen if it is broken*

The ALMA Pipeline, is a **Pipeline**, to run, stage-by-stage, automatically

- The **context** will correctly store stage-by-stage everything required for calibration and imaging to proceed and for each stage to have access to information it needs
- The Pipeline has to be run in order, which in order will store to the **context**
  - Repeating (some) stages will likely result in appending successive information over-and-over
  - e.g. repeating the **hifa\_gfluxscale** will continue to apply previous known corrections as told by the context – this is invalid
- Exiting Pipeline without correct context **save points** will render the data state and the context **incompatible**
  - Without saving, the context thinks processing (or imaging) is not as far as it is, stages will be saved overlapping to the same number
  - If in doubt begin processing/imaging again in a clean/new directory

For example, in manual running of Pipeline stages



# Pipeline Context from the User perspective

*Following good practices will save time*

Users don't really need to know what the context is, or what's inside, but

- Users **must** initiate the context
- Users **must** save the context at appropriate points
- Users **can restore** a context and **continue** a processing or imaging job, provided saving rules were followed
- Users can move/copy the context and all data run to a given saved point to another directory (with another name) as to restart a Pipeline run from that point again

# Pipeline Context from the User perspective

## A hypothetical example

Run session 1:

```
h_init()
hifa_importdata(vis='uid___A002_X123_Xabc')
h_save()
> exit
```

In CASA with Pipeline, you only **imported** data, then had to stop the job and leave

Run session 2:

```
h_resume("pipeline-20260119T173936.context")
hif_mstransform()
hifa_flagtargets()
hifa_imageprecheck()
hif_makeimlist(specmode='mfs')
hif_fincont()
hif_uvcontsub()
hif_makeimages()
h_save()
```

Now you **resume** the session, and don't need to import again the context knows you did it

Here you continue with some imaging commands for continuum subtractions and MFS imaging

# Pipeline Context from the User perspective

*Following good practices will save time*

Experience users can save to multiple context files, but this becomes very complex

- My advice for interactive sessions:
  - Copy a dataset/directory before starting so you can always restart
  - Make a habit to repeat **h\_save()** regularly after tasks
  - If a mess happens or stages fail, start fresh
- If you are doing a calibration or long run, also copy a full directory out at various intervals
  - If you are halfway through a long calibration or imaging and are worried you might make a mistake or something might crash
  - This moves the entire data, context, calibration and images into another directory which can be **resumed** at the same spot again

We will do this later



**Back to TUTORIAL**