



Imaging Pipeline Reprocessing

Tristan Ashton



Introduction to the ALMA Pipeline

- Used to calibrated ALMA interferometric (IF) and single-dish (SD) data.
- Automated calibration and imaging
- Modular calibration and imaging tasks within CASA, put together based on standard prescriptions or recipes
- Produces a WebLog – a collection of webpages with diagnostic messages, tables, figures and Quality Assurance (QA) scores
- User's guide and other useful documentation:
<https://almascience.nrao.edu/processing/science-pipeline>

Outline

- **Obtain the calibrated measurement set**
 - `scriptForPI.py`
 - `casa_piperestorescript.py`
 - **SRDP**
- How to re-run the calibration pipeline, if needed
 - `casa_pipescript.py`
- How to run the imaging pipeline
 - https://casaguides.nrao.edu/index.php?title=ALMA_Cycle_9_Imaging_Pipeline_Reprocessing
 - `scriptForReprocessing.py`

QA2 Data Products Package: Directory Structure

After untarring the processed data we have a directory tree:

Science
goal

```
2016.1.00164.S ← Project code
├── science_goal.uid__A001_X87a_X9fa
│   ├── group.uid__A001_X87a_X9fb
│   │   └── member.uid  A001 X87a X9fe ←
```

- calibration
- product
- qa
- script

Group OUS: combination
of member OUS's

Member OUS: may contain
12-m array, ALMA Compact
Array (ACA), or Total Power
observation

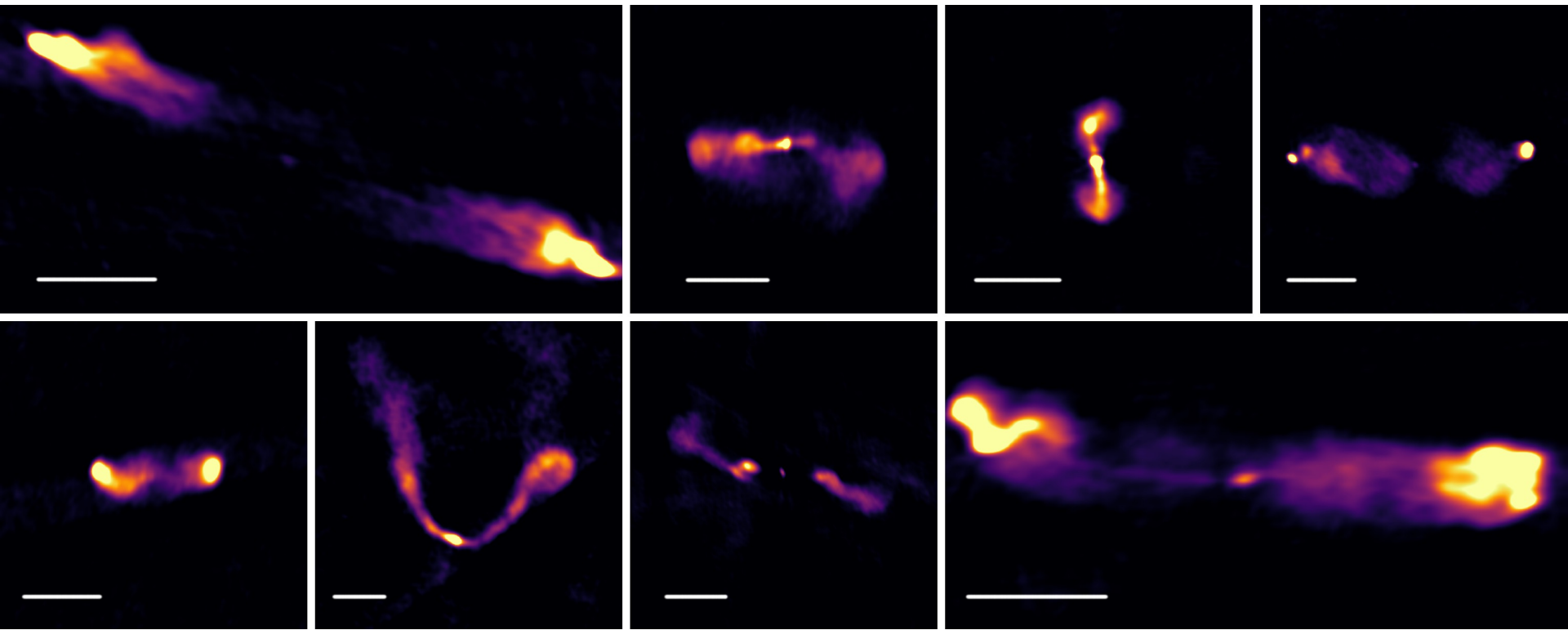
Data delivery products...

Restore calibrated measurement set

scriptForPl.py

- cd into **script/** directory
- Start the correct version of casa (casa --pipeline for PL tasks)
- Run scriptForPl.py: `execfile('*scriptForPl.py')`
- <https://help.almascience.org/index.php?/na/Knowledgebase/Article/View/267>
- But if you do want to use a newer version of CASA, inspect the measurement set carefully to make sure flags were applied correctly.
- A few known issues are posted here:
 - <https://help.almascience.org/index.php?/Knowledgebase/Article/View/379>
 - <https://help.almascience.org/index.php?/Knowledgebase/Article/View/395>

NRAO Science Ready Data Products



B. Kent+NINE program

<https://science.nrao.edu/srdp>

<https://data.nrao.edu>

What is the SRDP

Project?

- Data from modern radio interferometers such as the VLA and ALMA are both very large in terms of volume, and complicated in terms of what the data model allows.
- Significant barriers now exist for newcomers to data from these instruments, and even individuals with expertise find data processing very burdensome.
- This reduces the scientific output of these facilities – rather than being limited by the technical capabilities of the instruments, scientists are limited by the logistics of data processing

Calibrated MS from NRAO Archive

data.nrao.edu

Q 2019.A.00025.S

Active Search Inputs: Text Search 2019.A.00025.S ✕

▼ Show Search Inputs ▼

View Projects View Observations View Images

	↕ Project	↕ Instrument	Title	↕ First Obs	↕ Last Obs	
[-]	2019.A.00025.S	ALMA	The ALMA view of the Proxima c planet candidate	2021-03-24 04:54	2021-03-26 10:37	5 execution blocks

Title: The ALMA view of the Proxima c planet candidate
Abstract: The aim of this proposal is to obtain a second epoch ALMA image of Proxima Cen to confirm or refute the reality of a secondary 1.3mm source, displaced 1.2 arcsec (1.5 au at 1.3 pc) from the star. Among several other possibilities, it was noted that its flux density was consistent with that of a Saturn-like planetary ring system, and thus, that it could be tracing a yet unknown planet orbiting Proxima Cen. Now, independent evidence for a second planet, Proxima c, has been gathered. Radial velocities suggest a planet orbiting at 1.5 au, the same distance obtained from the ALMA image, and very recent VLT imaging suggests the presence of an optical/near-IR orbiting source coincident with the ALMA source. Given the large proper motion of the star Proxima Cen (4 arcsec/yr), and the orbital period of the planet (about 5.5 yr) the proposed second epoch ALMA observations will easily discriminate between a fake source, an unrelated background object, and material associated with planet Proxima c that is orbiting the star.
PI: Guillem Anglada
Co-Authors: Alice Zurlo, Anita Richards, René Heller, Pedro Amado, Raffaele Gratton, Dino Mesa, Jose-Francisco Gomez, Meredith MacGregor, Maria Lopez-Gonzalez, Paul Kalas, Jose Ortiz, Fabio Del Sordo, Eloy Rodriguez, Mario Damasso, Miguel Ángel Perez-Torres, Mayra Osorio, Itziar de Gregorio-Monsalvo, Guillermo Blazquez-Calero

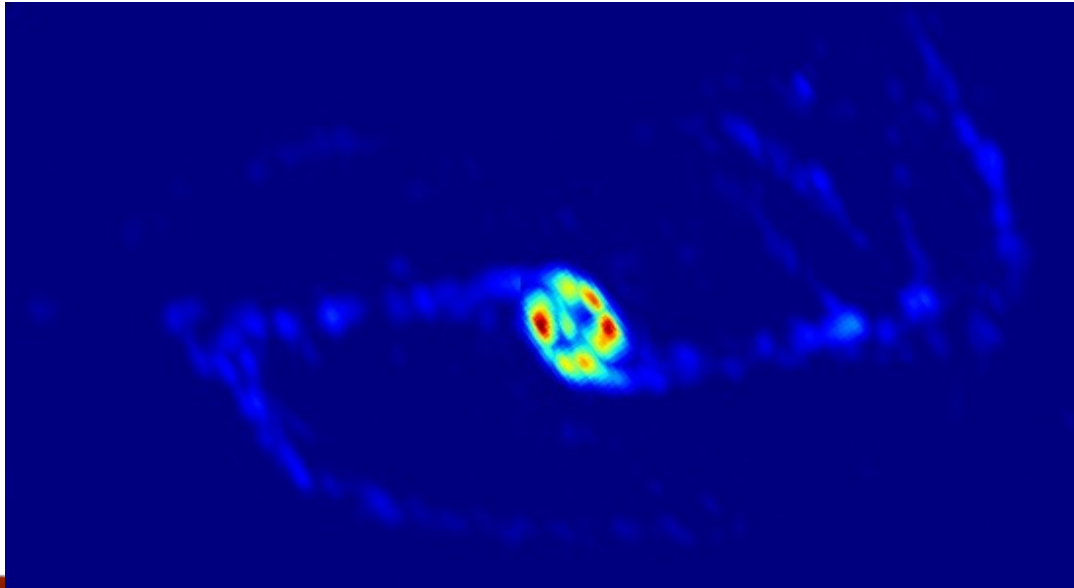
MOUSes Images

	MOUS	↕ Observation Start	↕ Observation Stop	File Size	Array Config	Ang Res	Bands	EBs	
▼	Proxima_a_06_TM1	2021-03-26 08:51	2021-03-26 10:37	644.991 GB	12M	0.269"	06	5	Download Restored MS Re-Imaging



ALMA Imaging

- Users can select their own imaging parameters via a web interface
- Pipeline software will apply the calibration to the raw data, then make the image per the users' request.
- Allows the user to image the part of the cube they want, at the resolution they need.



Outline

- Obtain the calibrated measurement set
 - `scriptForPI.py`
 - `casa_piperestorescript.py`
 - SRDP
- **How to re-run the calibration pipeline, if needed**
 - **`casa_pipescript.py`**
- How to run the imaging pipeline
 - https://casaguides.nrao.edu/index.php?title=ALMA_Cycle_9_1_imaging_Pipeline_Reprocessing
 - `scriptForReprocessing.py`

Re-running the pipeline using `casa_pipescript.py`

- Needed if you want to:
 - Add flags
 - Update fluxes
 - Update baselines
 - Use a new version of pipeline
 - Add or remove ASDM
 - Change reference antennas

Please see the Pipeline User's Guide for examples!

<https://almascience.nrao.edu/processing/science-pipeline>

Pipeline Helper text files

- flux.csv:
 - update the flux of calibrators.
- jyperk.csv:
 - SD pipeline to set the “Kelvin to Jansky” calibration factors which set the overall fluxscale of the data.
- antennapos.csv:
 - update the positions of the antenna elements
- uid*flagtemplate.txt:
 - add additional CASA flagging commands
- uid*flagstemplate.txt:
 - add additional CASA flagging commands that will be applied to the tsys
- uid*flagtargetstemplate.txt:
 - add additional CASA flagging commands that will be applied to the data after the calibration tables are calculated, but before science target imaging is performed.
- cont.dat:
 - used to specify the continuum frequency ranges used for constructing the continuum images and creating the continuum-subtracted cubes.

Re-running the calibration

<https://almascience.nrao.edu/processing/science-pipeline>

- Create **rawdata/**, **working/**, and **products/** subdirectories
- Copy **uid*casa_pipescript.py** to **casa_pipescript.py** in the **working/** directory (edit to include PL steps you wish to repeat)
- Copy **flux.csv**, **antennapos.csv** (if present), and **uid*flagtemplate.py** (one flagtemplate.py per execution, modify as needed) to the **working/** directory
- Copy raw ASDMs (rename without suffix .asdm.sdm) to **rawdata/** directory

Re-running the calibration

<https://almascience.nrao.edu/processing/science-pipeline>

- In **working/**
- Modify the pipeline “helper” files
- Edit **casa_pipescript.py** to only include the pipeline steps you wish to repeat (e.g. commenting out the findcont or imaging steps, which are very computationally expensive).
 - See the Pipeline User’s Guide
- Start the version of CASA containing Pipeline using
 - `casa --pipeline`
- You are now ready to run the script by typing `execfile('casa_pipescript.py')`.

Re-running the pipeline using casa_pipescript.py

<https://almascience.nrao.edu/processing/science-pipeline>

```
__rethrow_casa_exceptions = True
context=h_init()
try:
    hifa_importdata(dbsservice=False,
        vis=['uid__A002_X877e41_X452'], session=['session_1'])
        ## Uses flux.csv
    hifa_flagdata(pipeline="automatic")##Uses *flagtemplate.txt
    hifa_fluxcalflag(pipeline="automatic")
    hif_rawflagchans(pipeline="automatic")
    hif_refant(pipeline="automatic")
    h_tsyscal(pipeline="automatic")
    hifa_tsysflag(pipeline="automatic")
    hifa_antpos(pipeline="automatic") ## Uses antennapos.csv
    hifa_vvrgcalflag(pipeline="automatic")
    hif_lowgainflag(pipeline="automatic")
    hif_setmodels(pipeline="automatic")
    hifa_bandpassflag(pipeline="automatic")
    hifa_spwphaseup(pipeline="automatic")
    hifa_gfluxscaleflag(pipeline="automatic")
    hifa_gfluxscale(pipeline="automatic")
    hifa_timegaincal(pipeline="automatic")
    hif_applycal(pipeline="automatic")
    hif_makeimlist(intent='PHASE,BANDPASS,AMPLITUDE')
    hif_makeimages(pipeline="automatic")
    hif_makeimlist(per_eb=True, intent='CHECK')
    hif_makeimages(pipeline="automatic")
    hifa_imageprecheck(pipeline="automatic")
    hif_checkproducts(maxproducts=350.0, maxcubesize=40.0,
maxcubelimit=60.0)
    hifa_exportdata(pipeline="automatic")

# Start of pipeline imaging commands
    hif_mstransform(pipeline="automatic")
    hifa_flagtargets(pipeline="automatic")
        ## Uses *flagtargetstemplate.txt
    hif_makeimlist(specmode='mfs') ## Uses cont.dat
    hif_findcont(pipeline="automatic") ## Modifies cont.dat
    hif_uvcontfit(pipeline="automatic") ## Uses cont.dat
    hif_uvcontsub(pipeline="automatic")
    hif_makeimages(pipeline="automatic")## Uses cont.dat
    hif_makeimlist(specmode='cont') ## Uses cont.dat
    hif_makeimages(pipeline="automatic")## Uses cont.dat
    hif_makeimlist(specmode='cube') ## Uses cont.dat
    hif_makeimages(pipeline="automatic")## Uses cont.dat
    hif_makeimlist(specmode='reFBW') ## Uses cont.dat
    hif_makeimages(pipeline="automatic")## Uses cont.dat

finally:
    h_save()
```

Calibration tasks;
indicates the use of pipeline helper files

Imaging tasks;
indicates the use of pipeline helper files

Outline

- Obtain the calibrated measurement set
 - scriptForPI.py
 - casa_piperestorescript.py
 - SRDP
- How to re-run the calibration pipeline, if needed
 - casa_pipescript.py
- **How to run the imaging pipeline**
 - https://casaguides.nrao.edu/index.php?title=ALMA_Cycle_9_Imaging_Pipeline_Reprocessing
 - scriptForReprocessing.py

Pipeline Image Reprocessing

- Pipeline images are quality assessed but may not be science ready
 - All sources/spws may not be imaged (image mitigation to avoid long PL runs)
 - Change continuum selection
 - Change weighting, channel width, automasking, etc.
 - SRDP re-imaging can handle many of these situations!
- For pipeline calibrated data, see https://casaguides.nrao.edu/index.php/ALMA_Imaging_Pipeline_Reprocessing
- For manually calibrated data, see https://casaguides.nrao.edu/index.php?title=ALMA_Imaging_Pipeline_Reprocessing_for_Manually_Calibrated_Data

Restore Calibration and Prepare for Re-imaging

- Move to directory with calibrated data
- Copy the **uid*flagtargetstemplate.txt** and **cont.dat** from the calibration directory (if you have one)
- Make your changes, either to helper files or **casa_pipescript.py**
- Start casa --pipeline
- `execfile('casa_pipescript.py')`

CASA imaging pipeline script

Without calibration steps!

```
import glob as glob
__rethrow_casa_exceptions = True
context = h_init()

#The following context.set_state() statements are setting ancillary information and indeed not really necessary for imaging.
context.set_state('ProjectSummary', 'proposal_code', 'E2E6.1.00092.S')
context.set_state('ProjectSummary', 'piname', 'unknown')
context.set_state('ProjectSummary', 'proposal_title', 'unknown')
context.set_state('ProjectStructure', 'ous_part_id', 'X118955135')
context.set_state('ProjectStructure', 'ous_title', 'Undefined')
context.set_state('ProjectStructure', 'ppr_file', '../working/PPR_uid__A002_Xd0adbd_X56.xml')
context.set_state('ProjectStructure', 'ps_entity_id', 'uid://A002/Xd0adbd/X51')
context.set_state('ProjectStructure', 'recipe_name', 'hifa_image')
context.set_state('ProjectStructure', 'ous_entity_id', 'uid://A002/Xd0adbd/X4d')
context.set_state('ProjectStructure', 'ousstatus_entity_id', 'uid://A002/Xd0adbd/X55')
#
## Delete uid*_target.ms and flagversions if it exists
os.system('rm -rf uid*_target.ms')
os.system('rm -rf uid*_target.ms.flagversions')

MyVis=glob.glob('*.*ms')

try:
    hifa_importdata(vis=MyVis,dbservice=False)
    hif_mstransform(pipelinemode="automatic")
    hifa_flagtargets(pipelinemode="automatic")
    hifa_imageprecheck(pipelinemode="automatic")
    hif_checkproductsizes(maxproductsizes=350.0, maxcubesize=40.0, maxcubelimit=60.0)
    hif_makeimlist(specmode='mfs')
    hif_findcont(pipelinemode="automatic")
    hif_uvcontfit(pipelinemode="automatic")
    hif_uvcontsub(pipelinemode="automatic")
    hif_makeimages(pipelinemode="automatic")
    hif_makeimlist(specmode='cont')
    hif_makeimages(pipelinemode="automatic")
    hif_makeimlist(specmode='cube')
    hif_makeimages(pipelinemode="automatic")
    hif_makeimlist(specmode='repBW')
    hif_makeimages(pipelinemode="automatic")
    hifa_exportdata(imaging_products_only=True)
finally:
    h_save()
```

22. Image Pre-Check

BACK

Goals From OT:

Representative Target: G353.41

Representative Frequency: 93.1787 GHz (SPW 25)

Bandwidth for Sensitivity: 2000 MHz

Min / Max Acceptable Resolution: 0.760 arcsec / 1.14 arcsec

Maximum expected beam axial ratio (from OT): Not available

Goal PI sensitivity: Not available

Single Continuum: False

Estimated Synthesized Beam and Sensitivities for the Representative Target/Frequency

Estimates are given for four possible values of the tclean robust weighting parameter: robust = 0.0, +0.5 (default), +1.0, and +2.0. **If the "Min / Max Acceptable Resolution" is available (>=Cycle 5 12-m Array data)**, the robust value closest to the default (+0.5) that predicts a beam area (defined as simply major x minor) that is in the range of the PI requested beam areas according to the table row for repBW (Bandwidth for Sensitivity) is chosen. If none of these robust values predict a beam area that is in range, robust=+2.0 is chosen if the predicted beam area is too small, and robust=0.0 is chosen if the predicted beam area is too large. The chosen robust value is highlighted in green and used for all science target imaging. In addition to an estimate for the repBW, an estimate for the aggregate continuum bandwidth (aggBW) is also given assuming NO line contamination but accounting for spw frequency overlap. If the Bandwidth for Sensitivity (repBW) is > the bandwidth of the spw containing the representative frequency (repSPW), then the beam is predicted using all spws, otherwise the beam is predicted for the repSPW alone. A message appears on the "By Task" view if a non-default value of robust (i.e., not +0.5) is chosen. Additionally, if the predicted beam is not within the PI requested range using one of the four robust values, Warning messages appear on this page.

These estimates should always be considered as the BEST CASE SCENARIO. These estimates account for Tsys, the observed uv-coverage, and prior flagging. The estimates DO NOT account for (1) subsequent science target flagging; (2) loss of continuum bandwidth due to the hif_findcont process (i.e. removal of lines and other spectral features from the data used to image the continuum); (3) Issues that affect the image quality like (a) poor match of uv-coverage to image complexity; (b) dynamic range effects; (c) calibration deficiencies (poor phase transfer, residual baseline based effects, residual antenna position errors, etc.). *It is also important to note that both the repBW and aggBW beam calculations are intrinsically multi-frequency synthesis continuum calculations, using the relevant spws as described above. The synthesized beam for a single channel in a cube will typically be larger and can be significantly larger depending on the details of uv-coverage and channel width.*

robust	uvtaper	Synthesized Beam	Cell	Beam Ratio	Bandwidth	BW Mode	Effective Sensitivity
0.0	<input type="checkbox"/>	1.16 x 0.996 arcsec @ 79.2 deg	0.2 x 0.2 arcsec	1.16	2000 MHz	repBW	0.000157 Jy/beam
0.0	<input type="checkbox"/>	1.16 x 0.996 arcsec @ 79.2 deg	0.2 x 0.2 arcsec	1.16	2930 MHz	aggBW	0.000129 Jy/beam
0.5	<input type="checkbox"/>	1.25 x 1.08 arcsec @ 79.1 deg	0.22 x 0.22 arcsec	1.16	2000 MHz	repBW	0.000124 Jy/beam
0.5	<input type="checkbox"/>	1.25 x 1.08 arcsec @ 79.1 deg	0.22 x 0.22 arcsec	1.16	2930 MHz	aggBW	0.000102 Jy/beam
1.0	<input type="checkbox"/>	1.37 x 1.20 arcsec @ 78.6 deg	0.24 x 0.24 arcsec	1.14	2000 MHz	repBW	0.000115 Jy/beam
1.0	<input type="checkbox"/>	1.37 x 1.20 arcsec @ 78.6 deg	0.24 x 0.24 arcsec	1.14	2930 MHz	aggBW	9.53e-05 Jy/beam
2.0	<input type="checkbox"/>	1.43 x 1.25 arcsec @ 77.2 deg	0.25 x 0.25 arcsec	1.14	2000 MHz	repBW	0.000115 Jy/beam
2.0	<input type="checkbox"/>	1.43 x 1.25 arcsec @ 77.2 deg	0.25 x 0.25 arcsec	1.14	2930 MHz	aggBW	9.49e-05 Jy/beam

Pipeline QA

Score	Reason
0.85	Predicted non-default robust=0.0 beam is within the PI requested range

Pipeline QA summary for this task.

Example Cases for Imaging Pipeline changes

- Aggregate continuum Images with all channels
- Revise continuum ranges
- Remake cubes for a subset of spw and fields with channel binning, new robust, and uv-taper

- All Examples in CASA Guide:
https://casaguides.nrao.edu/index.php?title=ALMA_Cycle_8_Imaging_Pipeline_Reprocessing

Aggregate Continuum Image With All Channels

```
## Edit the USER SET INPUTS section below and then execute
## this script (note it must be in the 'calibrated/working' directory.

import glob as glob
__rethrow_casa_exceptions = True
pipelinemode='automatic'
context = h_init()

#####
## USER SET INPUTS

## Select a title for the weblog
context.project_summary.proposal_code='NEW AGGREGATE CONT'

## Delete uid*_target.ms and flagversions if it exists
os.system('rm -rf uid*_target.ms')
os.system('rm -rf uid*_target.ms.flagversions')

#####

## Move cont.dat to another name if it exists
os.system('mv cont.dat original.cont.dat')

## Make a list of all uv-datasets appended with *.ms
MyVis=glob.glob('*.ms')

try:
    ## Load the *.ms files into the pipeline
    hifa_importdata(vis=MyVis,dbservice=False,pipelinemode=pipelinemode)

    ## Split off the science target data into its own ms (called
    ## *_target.ms) and apply science target specific flags
    hif_mstransform(pipelinemode=pipelinemode)
    hifa_flagtargets(pipelinemode=pipelinemode)

    ## calculate the synthesized beam and estimate the sensitivity
    ## for the aggregate bandwidth and representative bandwidth
    ## for three values of the robust parameter.
    hifa_imageprecheck(pipelinemode="automatic")

    ## check the imaging product size and adjust the relevant
    ## imaging parameters (channel binning, cell size and image size)
    ## User can comment this out if they don't want size mitigation.
    hif_checkproductsize(maxproductsize=350.0, maxcubecellsize=40.0, maxcubelimit=60.0)

    ## Skip the continuum subtraction steps and make an aggregate
    ## continuum image with all unflagged channels (file named
    ## cont.dat should NOT be present in directory).
    hif_makeimlist(specmode='cont',pipelinemode=pipelinemode)
    hif_makeimages(pipelinemode=pipelinemode)

    ## Export new images to fits format if desired.
    hifa_exportdata(pipelinemode=pipelinemode)

finally:
    h_save()
```

Change name so pipeline does not pick up the file. All unflagged channels will be used in continuum image.

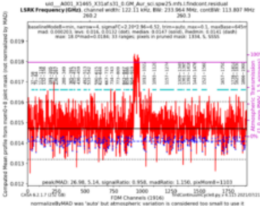
No findcont step to make new cont.dat!

Revise Continuum Range

See Section 7.6 of Pipeline User's guide for more cont.dat

- Edit or create a cont.dat file in the working directory

31. Find Continuum

Field	Spw	Continuum Frequency Range		Frame	Status	Average spectrum
		Start	End			
GM_Aur	25	260.13811 GHz	260.14079 GHz	LSRK	NEW	
		260.14250 GHz	260.14397 GHz			
		260.14726 GHz	260.14812 GHz			
		260.14971 GHz	260.15276 GHz			
		260.15691 GHz	260.15874 GHz			
		260.16009 GHz	260.16070 GHz			
		260.16241 GHz	260.16631 GHz			
		260.16876 GHz	260.17633 GHz			
		260.17816 GHz	260.18146 GHz			
		260.18304 GHz	260.18402 GHz			
		260.18683 GHz	260.18976 GHz			
		260.19367 GHz	260.19635 GHz			
		260.19867 GHz	260.19989 GHz			
		260.20563 GHz	260.20881 GHz			

Field: GM_Aur

SpectralWindow: 25

```

260.1381059540~260.1407923828GHz LSRK
260.1425019284~260.1439672532GHz LSRK
260.1472642340~260.1481190068GHz LSRK
260.1497064420~260.1527592020GHz LSRK
260.1569109556~260.1587426116GHz LSRK
260.1600858260~260.1606963780GHz LSRK
260.1624059236~260.1663134564GHz LSRK
260.1687556644~260.1763265092GHz LSRK
260.1781581652~260.1814551460GHz LSRK
260.1830425812~260.1840194644GHz LSRK
260.1868280036~260.1897586532GHz LSRK
260.1936661860~260.1963526148GHz LSRK
260.1986727124~260.1998938164GHz LSRK
    
```

Revise Continuum Range

```
#####  
## USER SET INPUTS  
  
## Select a title for the weblog  
context.project_summary.proposal_code = 'NEW CONTSUB'  
  
## Delete uid*_target.ms and flagversions if it exists  
os.system('rm -rf uid*_target.ms')  
os.system('rm -rf uid*_target.ms.flagversions')  
  
## Select spw(s) that have new cont.dat parameters  
## If all spws have changed use MySpw=''  
MySpw='17'  
  
#####  
  
## Make a list of all uv-datasets appended with *.ms  
MyVis=glob('*.ms')  
  
try:  
    ## Load the *.ms files into the pipeline  
    hifa_importdata(vis=MyVis,dbservice=False,pipelinemode=pipelinemode)  
  
    ## Split off the science target data into its own ms (called  
    ## *_target.ms) and apply science target specific flags  
    hif_mstransform(pipelinemode=pipelinemode)  
    hifa_flagtargets(pipelinemode=pipelinemode)  
  
    ## Fit and subtract the continuum using revised cont.dat for all spws  
    hif_makeimlist(specmode='mfs',spw=MySpw)  
    hif_uvcontfit(pipelinemode=pipelinemode)  
    hif_uvcontsub(pipelinemode=pipelinemode)  
    hif_makeimages(pipelinemode=pipelinemode)  
  
    ## calculate the synthesized beam and estimate the sensitivity  
    ## for the aggregate bandwidth and representative bandwidth  
    ## for three values of the robust parameter.  
    hifa_imageprecheck(pipelinemode=pipelinemode)  
  
    ## check the imaging product size and adjust the relevant  
    ## imaging parameters (channel binning, cell size and image size)  
    ## User can comment this out if they don't want size mitigation.  
    hif_checkproductsize(maxproductsize=350.0, maxcubesize=40.0, maxcubelimit=60.0)  
  
    ## Make new aggregate cont  
  
    hif_makeimlist(specmode='cont',pipelinemode=pipelinemode)  
    hif_makeimages(pipelinemode=pipelinemode)  
  
    ## Make new continuum subtracted cube for revised spw(s)  
    hif_makeimlist(specmode='cube',spw=MySpw,pipelinemode=pipelinemode)  
    hif_makeimages(pipelinemode=pipelinemode)  
  
    ## Export new images to fits format if desired.  
    hifa_exportdata(pipelinemode=pipelinemode)  
  
finally:  
    h_save()
```

Will use new cont.dat in uvcontsub

Will make Continuum images
using cont.dat selection

Remake cubes with select spw, bins, fields, robust

Set parameters

```
## USER SET INFO

## Select a title for the weblog
context.project_summary.proposal_code = 'SUBSET CUBE IMAGING'

## Delete uid*_target.ms and flagversions if it exists
os.system('rm -rf uid*_target.ms')
os.system('rm -rf uid*_target.ms.flagversions')

## Select spw(s) to image and channel binning for each spcified
## MySpw. All spws listed in MySpw must have a corresponding MyNbins
## entry, even if it is 1 for no binning.
MySpw='17,23'
MyNbins='17:8,23:2'

## Select subset of sources to image by field name.
## To select all fields, set MyFields=''
MyFields='CoolSource1,CoolSource2'

## Select Briggs Robust factor for data weighting (affects angular
## resolution of images)
MyRobust=1.5

#####

## Make a list of all uv-datasets appended with *.ms
MyVis=glob.glob('*.*ms')
```

Check Pipeline Reference Manual for parameters of pipeline tasks!

Include parameters in task calls

```
try:
## Load the *.ms files into the pipeline
hifa_importdata(vis=MyVis, dbservice=False, pipelinemode=pipelinemode)

## Split off the science target data into its own ms (called
## *target.ms) and apply science target specific flags
## In this example we split off all science targets and science
## spws, however hif_mstransform could also contain the spw and field
## selections
hif_mstransform(pipelinemode=pipelinemode)
hifa_flagtargets(pipelinemode=pipelinemode)

## Fit and subtract the continuum using existing cont.dat
## for selected spws and fields only.
hif_makeimlist(specmode='mfs')
hif_uvcontfit(spw=MySpw, field=MyFields, pipelinemode=pipelinemode)
hif_uvcontsub(spw=MySpw, field=MyFields, pipelinemode=pipelinemode)
hif_makeimages(pipelinemode=pipelinemode)

## calculate the synthesized beam and estimate the sensitivity
## for the aggregate bandwidth and representative bandwidth
## for three values of the robust parameter.
## Don't need to run this task if you will use a different robust value anyway.
## hifa_imageprecheck(pipelinemode=pipelinemode)

## check the imaging product size and adjust the relevant
## imaging parameters (channel binning, cell size and image size)
## User can comment this out if they don't want size mitigation.
hif_checkproductsize(maxproductsize=350.0, maxcubesize=40.0, maxcubelimit=60.0)

## Make new continuum subtracted cube for selected spw(s) and fields
hif_makeimlist(specmode='cube', spw=MySpw, nbins=MyNbins, field=MyFields, robust=MyRobust, pipelinemode=pipelinemode)
hif_makeimages(pipelinemode=pipelinemode)

## Export new images to fits format if desired.
hifa_exportdata(pipelinemode=pipelinemode)

finally:
h_save()
```

Imaging pipeline reprocessing

A weblog is always created!

Home
Home
By Topic
By Task

NEW AGGREGATE CONT

Tasks in execution **Tasks in execution order**

1. hifa_importdata 1. hifa_importdata
2. hif_mstransform 2. hif_mstransform
3. hifa_flagtargets 3. hifa_flagtargets
4. hifa_imageprecheck 4. hifa_imageprecheck
5. hif_checkproducts 5. hif_checkproducts
6. hif_makeimlist (cont) 6. hif_makeimlist (cont)
7. hif_makeimages (cont) 7. hif_makeimages (cont)
8. hifa_exportdata 8. hifa_exportdata

Task notifications

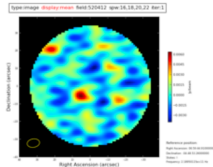
Warning! No continuum frequency selection for Target Field '520412' SPW 16

Warning! No continuum frequency selection for Target Field '520412' SPW 18

Warning! No continuum frequency selection for Target Field '520412' SPW 20

Warning! No continuum frequency selection for Target Field '520412' SPW 22

Image Details

Field	Spw	Pol	Image details	Image result
520412 (TARGET)	16, 18, 20, 22 / X1222766920#ALMA_RB_06#BB_1#SW-01, X1222766920#ALMA_RB_06#BB_2#SW-01, X1222766920#ALMA_RB_06#BB_3#SW-01, X1222766920#ALMA_RB_06#BB_4#SW-01	I	<p>centre frequency of image 218.9501GHz (LSRK)</p> <p>beam 7.10 x 4.77 arcsec</p> <p>beam p.a. -80.6deg</p> <p>final theoretical sensitivity 0.00071 Jy/beam</p> <p>cleaning threshold 0.0022 Jy/beam Dirty DR: 8.9 DR correction: 1.5</p> <p>clean residual peak / scaled MAD 4.71</p> <p>non-pbcor image RMS 0.0012 Jy/beam</p> <p>pbcor image max / min 0.0169 / -0.0106 Jy/beam</p> <p>fractional bandwidth / nterms 1.9% / 1</p> <p>aggregate bandwidth 5 GHz (LSRK)</p> <p>score 1.00</p> <p>image file oussid.s7_0_520412_sci.spw16_18_20_22.cont.l.iter1.image</p>	 <p>View other QA images...</p>

Outline

- Obtain the calibrated measurement set
 - scriptForPI.py
 - casa_piperestorescript.py
 - SRDP
- How to re-run the calibration pipeline, if needed
 - casa_pipescript.py
- **How to run the imaging pipeline**
 - https://casaguides.nrao.edu/index.php?title=ALMA_Cycle_9_Imaging_Pipeline_Reprocessing
 - scriptForReprocessing.py

scriptFo

- Include Cycle 9
- calibrated

ALMA Cycle 9 Project

Agent reply

AA API Agent 1 week ago

Dear ,

You should have recently received an email announcing that data for member ObsUnitSet from your Science Goal : , containing the Scheduling Block in your project with the title:

are now available for download through the ALMA Science Portal Request Handler. For your convenience, the North American ALMA Science Center provides a suite of added-value products, **which has been updated for Cycle 9:**

- **README: A full explanation of the new added-value data product structure**
- **calibrated_final.tgz:** Calibrated measurement sets for each Execution Block, plus calibration tables, to facilitate reprocessing with the ALMA imaging pipeline
- **scriptForReprocessing.py:** Allows quick and easy use of the imaging pipeline, with several options for continuum subtraction and imaging
- **Weblogs:** Available for one-click remote viewing in your browser
- **QA2 and QA0 information:** Comments from the data reducer about the calibration
- **ADMIT products:** A quick-look for interferometric data, including moment maps and line identification for well-known chemical species

CASAGuides illustrating the use of the ALMA imaging pipeline can be accessed at https://casaguides.nrao.edu/index.php?title=ALMA_Imaging_Pipeline_Reprocessing

Unlike the data obtained through the Request Handler, **these added value products will only be available for download for the next 30 days.**

Your data may be found here: <https://bulk.cv.nrao.edu/almadata/proprietary>.

Please use your ALMA username and password to login. For more information about data calibration, imaging, ADMIT, and how to request Face-to-Face support, please see the following Knowledgebase article: <https://help.almascience.org/kb/articles/where-can-i-get-additional-information-for-my-na-added-value-data-products>

With kind regards,
The North American ALMA Archive at the NAASC
message sent at 2023-

[Resolve ticket](#)

Ticket Status

Awaiting User

Created 6 AM

Reference

Assigned agent

GP George Privon

CCs 2 CC+

RJ

JL

[Click here to scroll down to the form.](#)

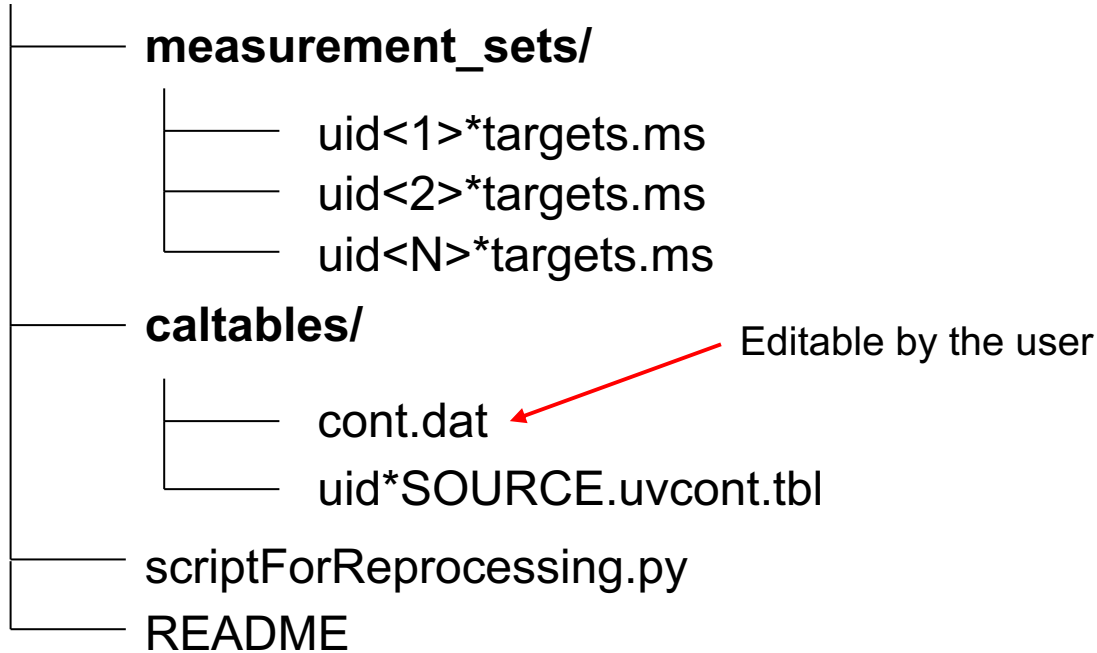
Ticket Properties

Department

Cycle 9 Observing Programs (NA)

calibrated_final.tgz

calibrated_final/



Reproduce Pipeline Images

- By default, identical images to those seen in the original weblog are produced
- cd to **calibrated_final/**
- Run casa --pipeline
 - `execfile('scriptForReprocessing.py [flags]')`
- Or: `casa --pipeline -c scriptForReprocessing.py [flags]`

Script flags

<code>-h, --help</code>	show this help message and exit
<code>--contsub</code>	Fit and subtract continuum using the channel ranges from the local <code>cont.dat</code> file. Generates new <code>*uvcont.tbl</code> tables in <code>working_reprocess/</code> directory and <code>uid*targets_line.ms</code> in <code>measurement_sets/</code>
<code>--contsub_fast</code>	Continuum subtract data via <code>uvcontsub</code> and the local <code>*uvcont.tbl</code> files, but only using the CASA commands rather than pipeline calls (faster). Generates <code>uid*targets_line.ms</code> in <code>measurement_sets/</code>
<code>--image [IMAGE]</code>	Run the imaging pipeline and place images in the specified directory (default='images'). NOTE: unless <code>cont.dat</code> or the imaging options in this script are modified, the images produced will be identical to those on the ALMA Science Archive
<code>--cleanup</code>	Remove <code>working_reprocess/</code> directory and log files after any other options are executed. WARNING: removes weblogs inside of <code>working_reprocess/</code>
<code>--weblog [WEBLOG]</code>	Launches a browser to view weblog after other tasks are run. By default ('latest'), displays the latest weblog generated locally. Other options are to use the specific pipeline folder name (e.g. 'pipeline-20221010T192458')
<code>--calibrated_final</code>	Concatenate <code>uid*targets.ms</code> to produce <code>calibrated_final.ms</code> in <code>measurement_sets/</code>
<code>--calibrated_final_line</code>	Concatenate <code>uid*targets_line.ms</code> (if they exist) to produce <code>calibrated_final_line.ms</code> in <code>measurement_sets/</code>

Imaging with different parameters

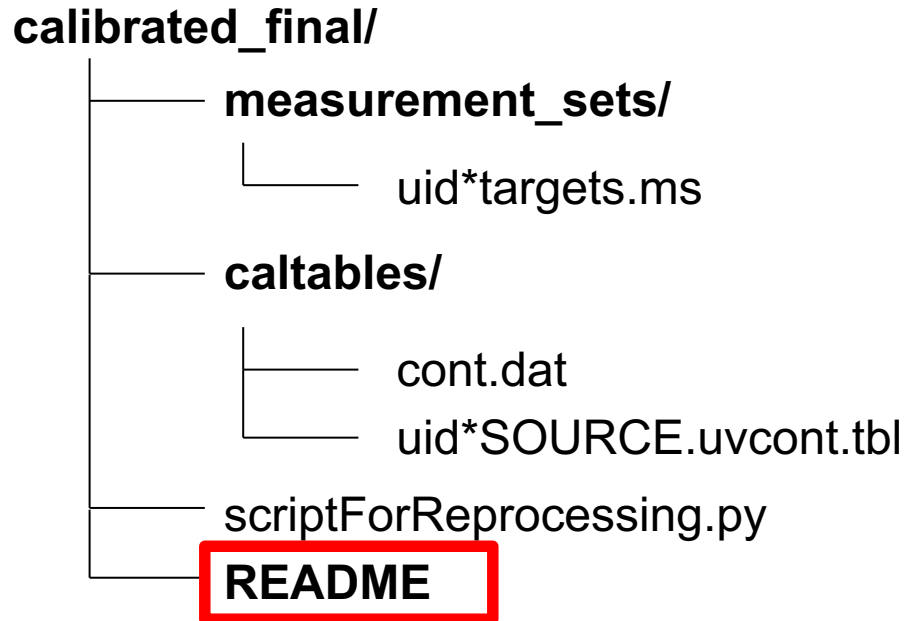
```
make_mfs_images = True      # generate mfs (per spw) images
make_cont_images = True    # generate aggregate continuum images
make_cube_images = True    # generate cube images
make_repBW_images = True   # generate images corresponding to the requested representative bandwidth
mitigate = True            # run hif_checkproductsize() and mitigate created products if necessary.
                           # Set to false if you want all spws and all targets imaged at full resolution.
                           # WARNING: turning off mitigation may result in very large disk usage. Consider
                           # adjusting other mitigation parameter first, or manually selecting the target/spw
                           # combinations you want.
```

```
# for all values below, see Pipeline Users Guide and Reference Manual for detailed descriptions:
# https://almascience.nrao.edu/processing/science-pipeline
```

```
#
# any parameters not selected here can be edited manually in relevant section of the script below
maxproductsize = 350.      # for mitigation; in GB
maxcubesize = 40.         # for mitigation; in GB
maxcubelimit = 60.        # for mitigation; in GB

field = None               # String specifying fields to be imaged; default is all (pending mitigation)
                           # Example: '3C279, M82'
phasecenter = None        # Direction measure or field id of the image center. The default phase center is
                           # set to the mean of the field directions of all fields that are to be image together.
                           # Examples: 'ICRS 13:05:27.2780 -049.28.04.458', "TRACKFIELD" (for ephemeris)
spw = None                 # Spw(s) to image; default is all spws
                           # Example: '17, 23'
uvrange = None            # Select a set of uv ranges to image; default is all
                           # Examples: '0~1000klambda', ['0~100klambda','300~1000klambda']
hm_imsz = None            # Image x and y size in pixels or PB level; default is automatically determined
                           # Examples: '0.3pb', [120, 120]
hm_cell = None            # Image cell size; default is automatically determined
                           # Examples: '3ppb', ['0.5arcsec', '0.5arcsec']
nbins = None              # Channel binning factor for each spw; default is none
                           # Format: 'spw1:nb1,spw2:nb2,...' with optional wildcards: '*:nb'
                           # Examples: '9:2,11:4,13:2,15:8', '*:2'
robust = None             # Robust value to image with; default is automatically determined
                           # Example: 0.5
uvtaper = None            # Uvtaper to apply to data; default is none
                           # Example: ['1arcsec']
```

calibrated_final.tgz



<https://help.almascience.org/kb/articles/where-can-i-get-additional-information-for-my-na-added-value-data-products>

<https://help.almascience.org/kb/articles/what-can-i-do-with-my-na-added-value-data-products>

 **Historical Articles (25)**

- Why are flux scaling and restoring beam calculations in the tas...
- Why must I justify data rates higher than 40 MB/s in my Cycle ...
- OT will not start due to a security exception
- What is a Cycle 7 Solar data issue I should be aware of?
- I heard that images generated with CASA<5.1.1 are affected by ...

[View all articles in Historical Articles >](#)

 **Archive & Data Retrieval (32)**

- How do I obtain a file of calibrated visibilities ("measurement s...
- How are ALMA data products packaged?
- Can I download individual data products or images?
- Can I search for a list of sources at the same time?
- Is it possible to resume interrupted downloads?

[View all articles in Archive & Data Retrieval >](#)

Where can I get additional information for my NA added value data products?

 Jul 14, 2023


 Andrew Lipnicky Published: Jan 22, 2019

What are the current QA2 requirements?

 Jul 11, 2022

 Sarah Wood Published: Nov 19, 2019

How can I solve this astroquery problem?

 Apr 18, 2023

 Felix Stoehr Published: May 27, 2021

What can I do with my NA added value data products?

 Jul 7, 2023

 Tristan Ashton Published: Jul 6, 2023



www.nrao.edu
science.nrao.edu
public.nrao.edu

*The National Radio Astronomy Observatory is a facility of the National Science Foundation
operated under cooperative agreement by Associated Universities, Inc.*