Using CASA to Simulate Interferometer Observations

Nuria Marcelino

North American ALMA Science Center

Atacama Large Millimeter/submillimeter Array
Expanded Very Large Array
Robert C. Byrd Green Bank Telescope
Very Long Baseline Array

NRAO

Simulating Interferometer Data

- Take a model image and simulate how it would look if observed by ALMA or the JVLA.
  - Other arrays (e.g., SMA, CARMA, etc.) also included

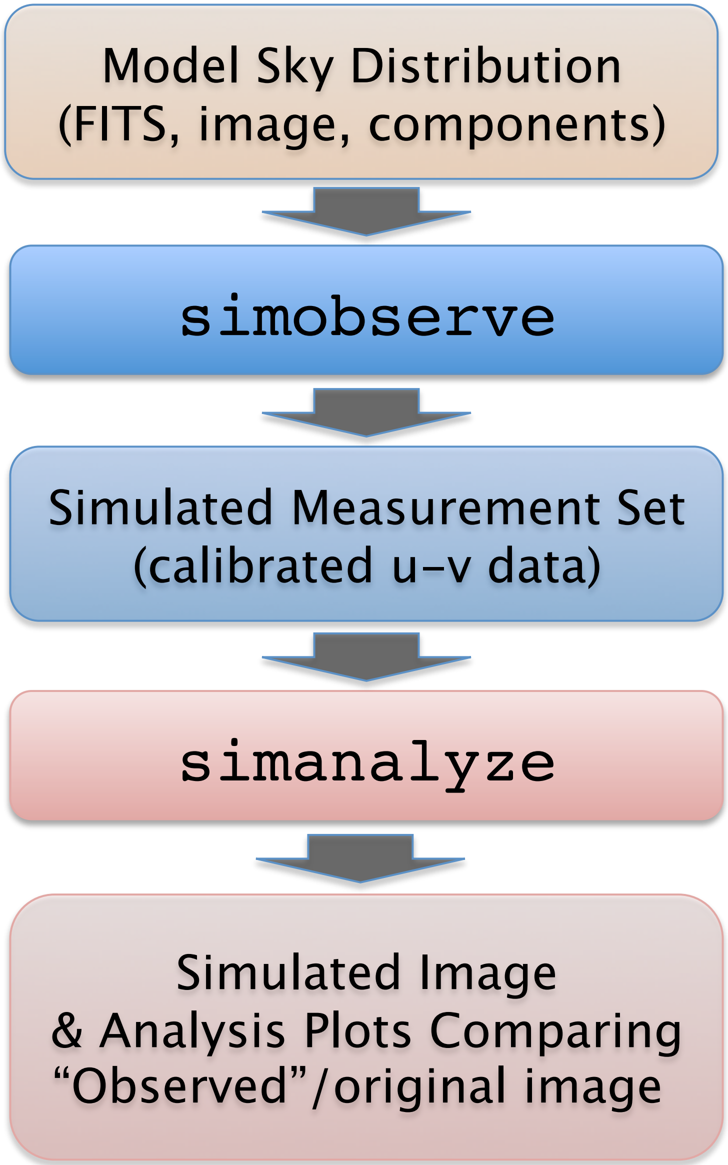
- Explore the effects of:
  - Number of antennas
  - Antenna configuration
  - Length of observation
  - Thermal noise
  - Phase noise

- Functionality included in CASA via tasks `simobserve` and `simanalyze` (nee `simdata`).

- CASAguides includes several walkthroughs: http://casaguides.nrao.edu/index.php?title=Simulating_Observations_in_CASA

Basic Simulation Workflow

In CASA…



Model Sky Distribution
(FITS, image, components)

`simobserve`

Simulated Measurement Set
(calibrated u-v data)

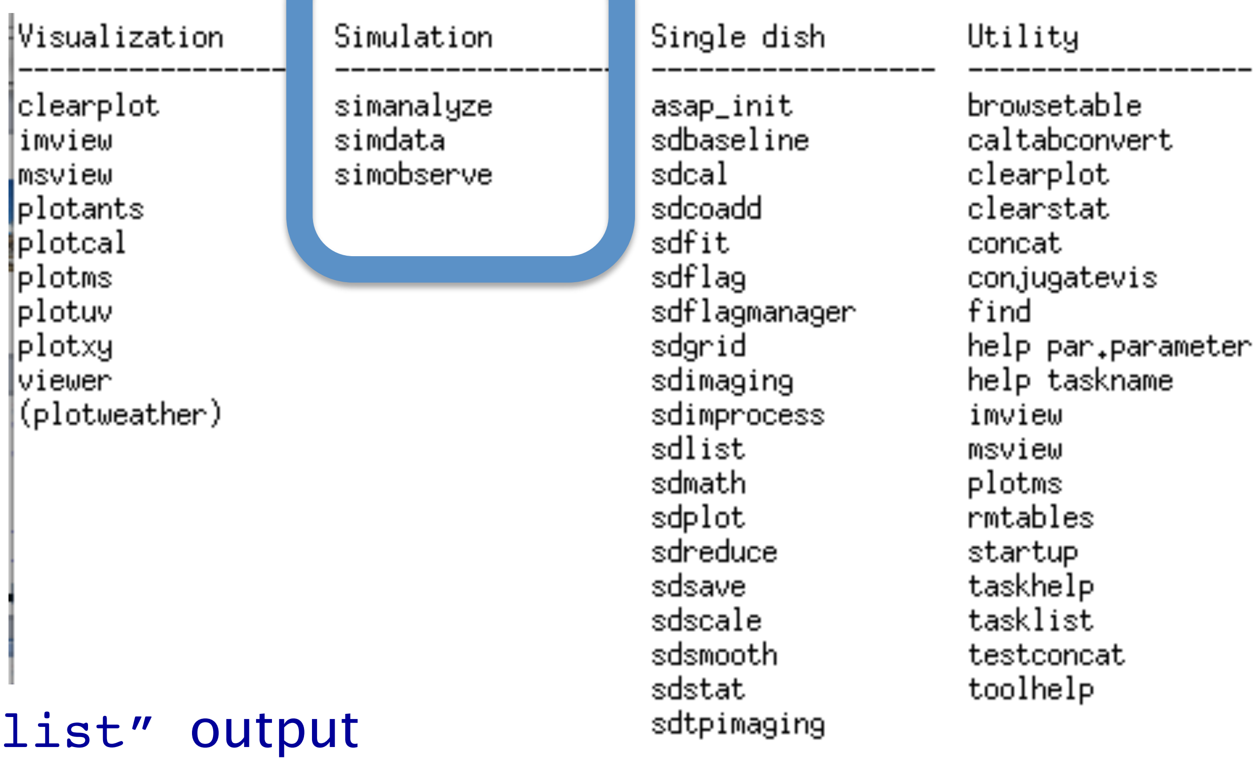
`simanalyze`

Simulated Image
& Analysis Plots Comparing
“Observed”/original image

Simulation Tasks

- `simobserve` simulates interferometric (and single dish) observations of a source.

- `simanalyze` images and analyzes these simulations.

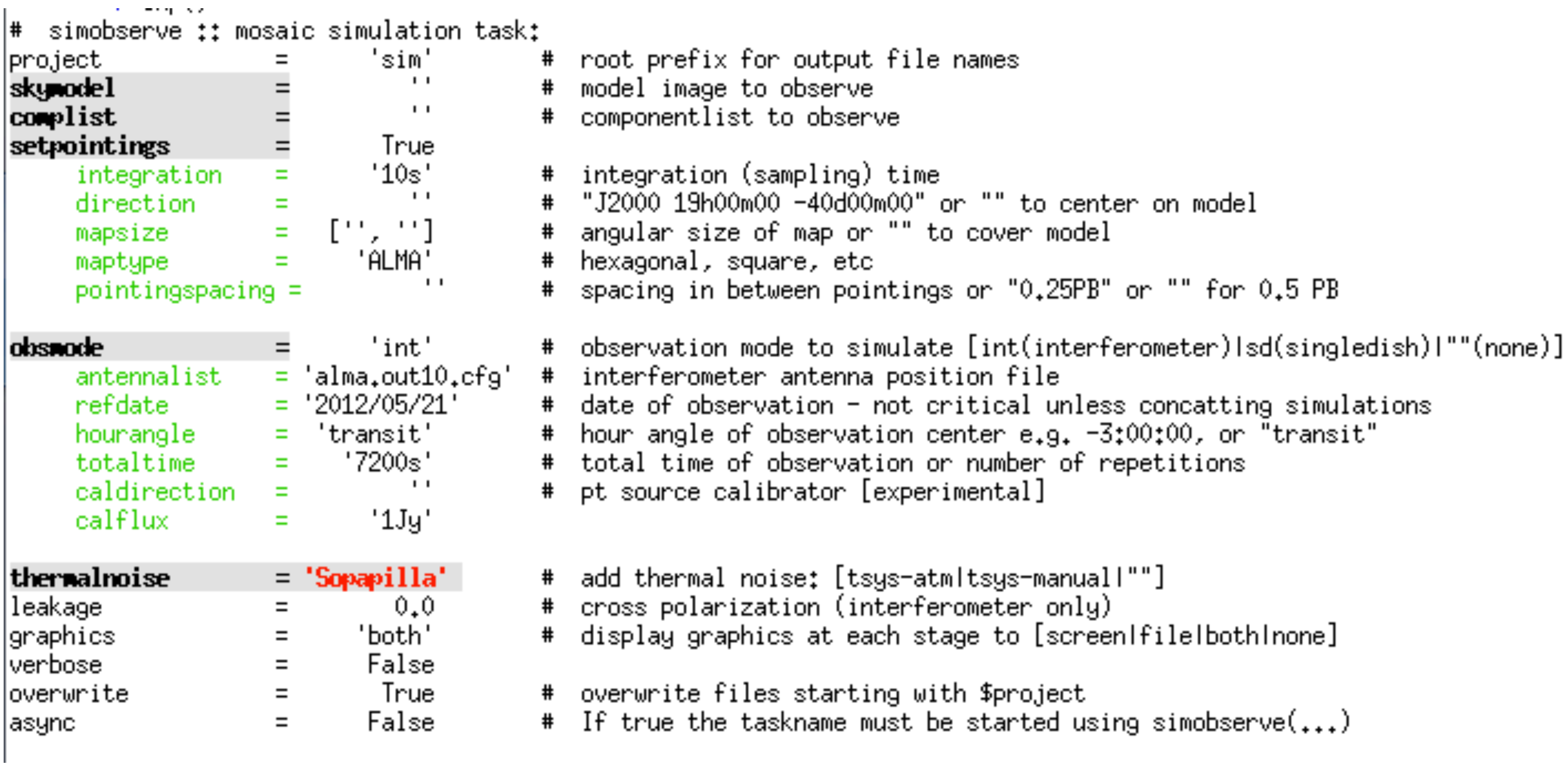


```
Visualization        Simulation           Single dish          Utility
-----------------    -----------------    -----------------    -----------------
clearplot            simanalyze           asap_init            browsetable
imview               simdata              sdbaseline           caltabconvert
msview               simobserve           sdcal                clearplot
plotants                                  sdcoadd              clearstat
plotcal                                   sdfit                concat
plotms                                    sdflag               conjugatevis
plotuv                                    sdflagmanager        find
plotxy                                    sdgrid               help par.parameter
viewer                                    sdimaging            help taskname
(plotweather)                             sdimprocess          imview
                                          sdlist               msview
                                          sdmath               plotms
                                          sdplot               rmtables
                                          sdreduce             startup
                                          sdsave               taskhelp
                                          sdscale              tasklist
                                          sdsmooth             testconcat
                                          sdstat               toolhelp
                                          sdtpimaging
```

“tasklist” output

simobserve

- simulates interferometer observations of a source.

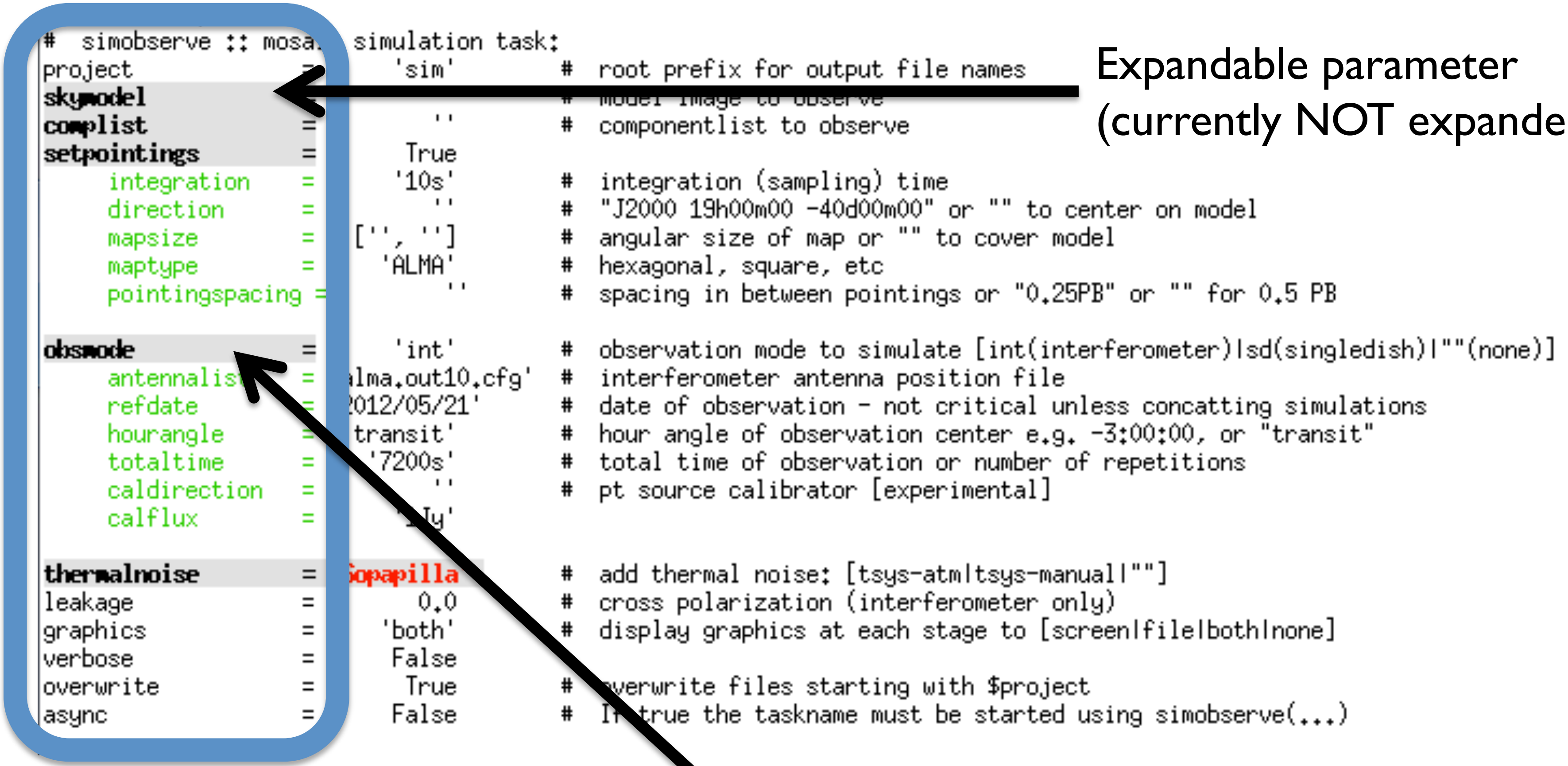


```
#  simobserve :: mosaic simulation task:
project             =      'sim'         #  root prefix for output file names
skymodel            =         ''         #  model image to observe
complist            =         ''         #  componentlist to observe
setpointings        =       True
     integration    =      '10s'         #  integration (sampling) time
     direction      =         ''         #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize        =   ['', '']         #  angular size of map or "" to cover model
     maptype        =     'ALMA'         #  hexagonal, square, etc
     pointingspacing =         ''        #  spacing in between pointings or "0.25PB" or "" for 0.5 PB

obsmode             =      'int'         #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist    = 'alma.out10.cfg'   #  interferometer antenna position file
     refdate        = '2012/05/21'       #  date of observation - not critical unless concatting simulations
     hourangle      =  'transit'         #  hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime      =    '7200s'         #  total time of observation or number of repetitions
     caldirection   =         ''         #  pt source calibrator [experimental]
     calflux        =      '1Jy'

thermalnoise        = 'Sopapilla'        #  add thermal noise: [tsys-atm|tsys-manual|""]
leakage             =        0.0         #  cross polarization (interferometer only)
graphics            =     'both'         #  display graphics at each stage to [screen|file|both|none]
verbose             =      False
overwrite           =       True         #  overwrite files starting with $project
async               =      False         #  If true the taskname must be started using simobserve(...)
```

"inp simobserve" output

CASA Refresher

- `inp` shows parameter names



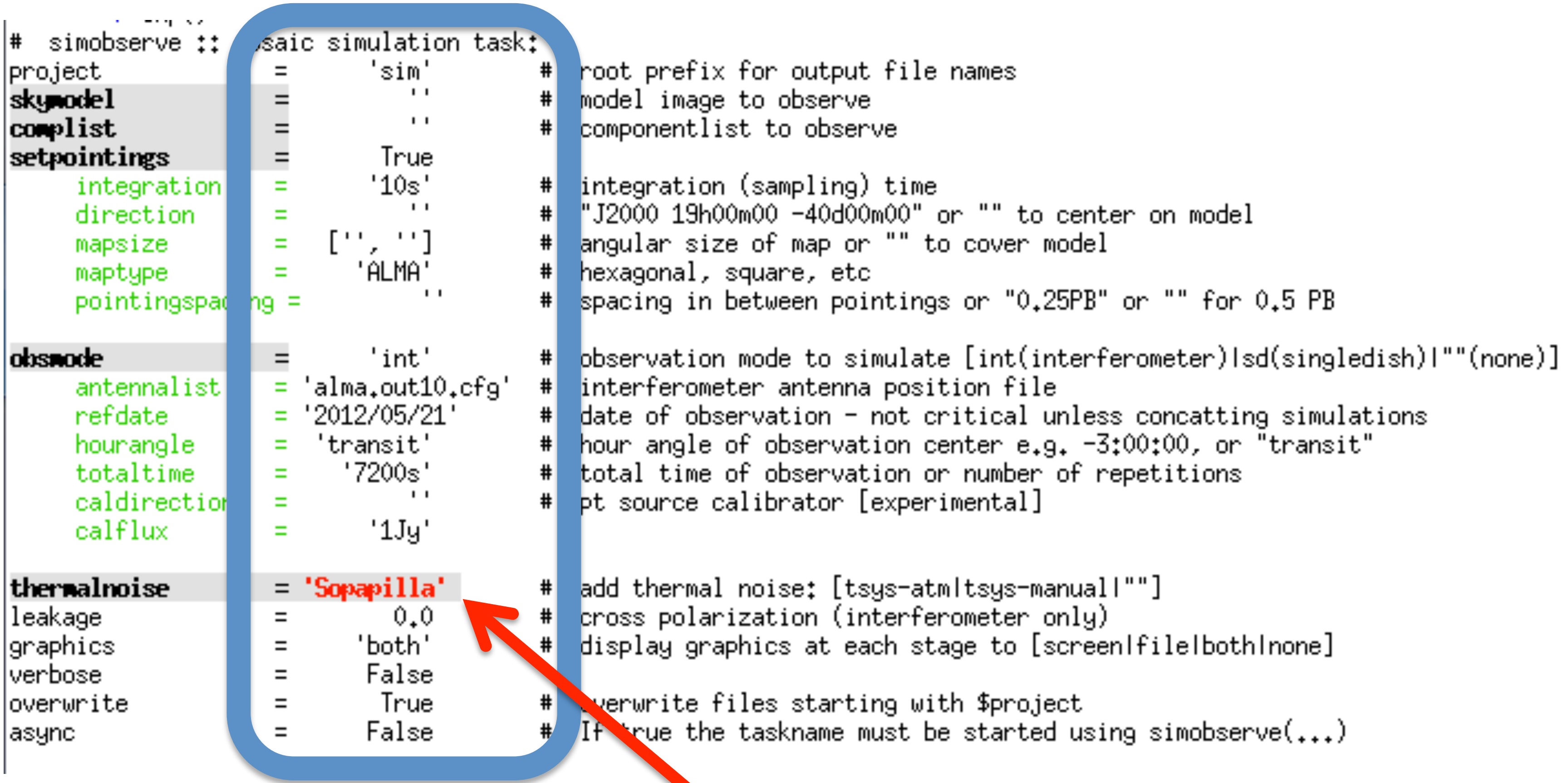
```
#  simobserve :: mosa  simulation task:
project            =      'sim'        #  root prefix for output file names
skymodel           =                   #  model image to observe
complist           =         ''        #  componentlist to observe
setpointings       =       True
     integration   =      '10s'        #  integration (sampling) time
     direction     =         ''        #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize       =   ['', '']        #  angular size of map or "" to cover model
     maptype       =     'ALMA'        #  hexagonal, square, etc
     pointingspacing =        ''       #  spacing in between pointings or "0.25PB" or "" for 0.5 PB

obsmode            =      'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalis    = lma.out10.cfg'   #  interferometer antenna position file
     refdate       = 2012/05/21'      #  date of observation - not critical unless concatting simulations
     hourangle     =  transit'        #  hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime     =    '7200s'        #  total time of observation or number of repetitions
     caldirection  =         ''        #  pt source calibrator [experimental]
     calflux       =      '1Jy'

thermalnoise       = Sopapilla         #  add thermal noise: [tsys-atm|tsys-manual|""]
leakage            =        0.0        #  cross polarization (interferometer only)
graphics           =     'both'        #  display graphics at each stage to [screen|file|both|none]
verbose            =      False
overwrite          =       True        #  overwrite files starting with $project
async              =      False        #  If true the taskname must be started using simobserve(...)
```

Expandable parameter (currently NOT expanded)

Expandable parameter (currently expanded)

CASA Refresher

- `inp` shows current value (change, e.g., by `project = “myproj”`)

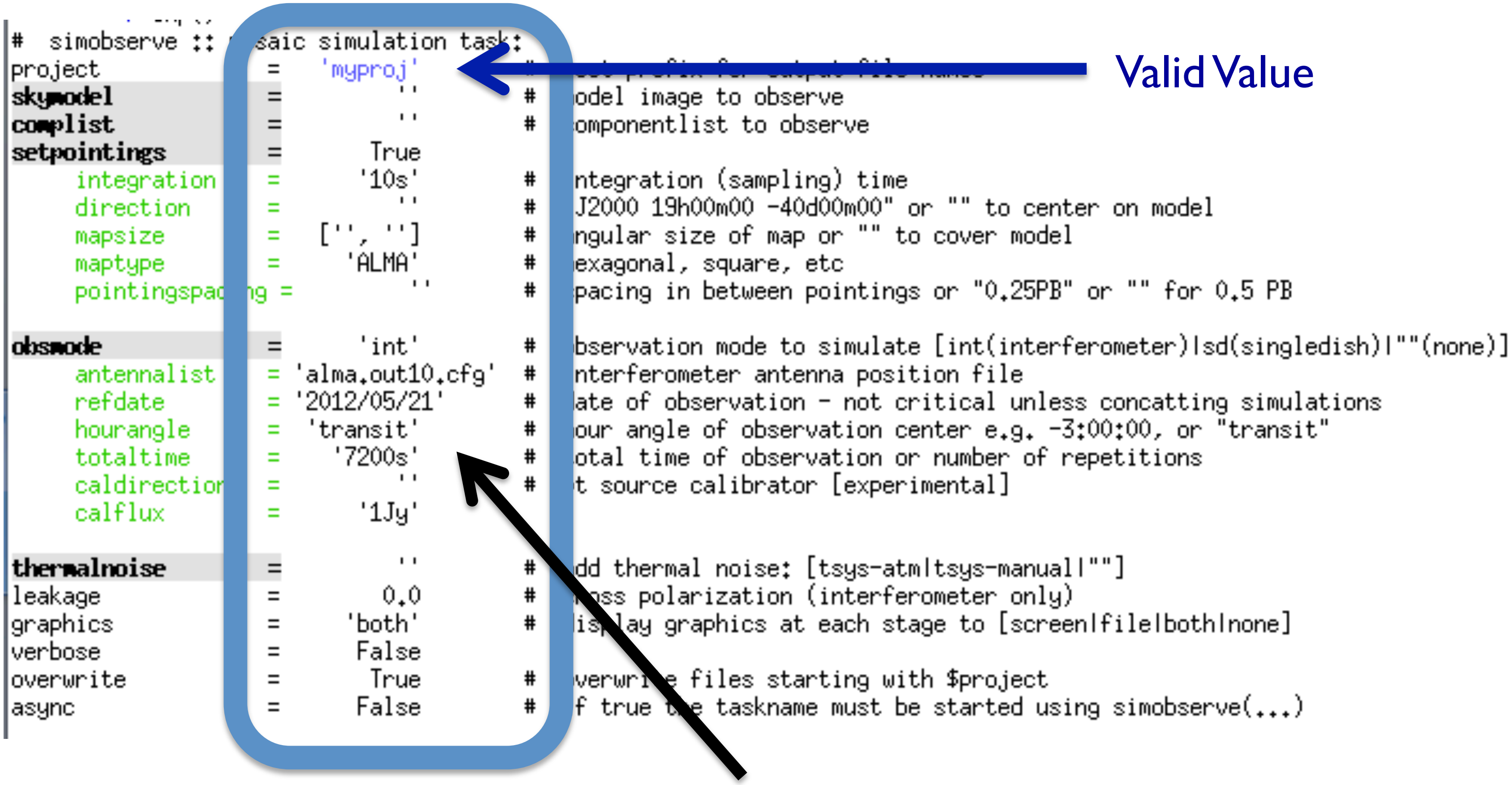


```
#  simobserve :: mosaic simulation task:
project            =       'sim'        #  root prefix for output file names
skymodel           =          ''        #  model image to observe
complist           =          ''        #  componentlist to observe
setpointings       =        True
     integration   =       '10s'        #  integration (sampling) time
     direction     =          ''        #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize       =    ['', '']        #  angular size of map or "" to cover model
     maptype       =      'ALMA'        #  hexagonal, square, etc
     pointingspacing =        ''        #  spacing in between pointings or "0.25PB" or "" for 0.5 PB

obsmode            =       'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist   = 'alma.out10.cfg'   #  interferometer antenna position file
     refdate       = '2012/05/21'       #  date of observation - not critical unless concatting simulations
     hourangle     =   'transit'        #  hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime     =     '7200s'        #  total time of observation or number of repetitions
     caldirection  =          ''        #  pt source calibrator [experimental]
     calflux       =       '1Jy'

thermalnoise       = 'Sopapilla'        #  add thermal noise: [tsys-atm|tsys-manual|""]
leakage            =         0.0        #  cross polarization (interferometer only)
graphics           =      'both'        #  display graphics at each stage to [screen|file|both|none]
verbose            =       False
overwrite          =        True        #  overwrite files starting with $project
async              =       False        #  If true the taskname must be started using simobserve(...)
```

Invalid Value

CASA Refresher

- `inp` shows current value (change, e.g., by `project = “myproj”`)



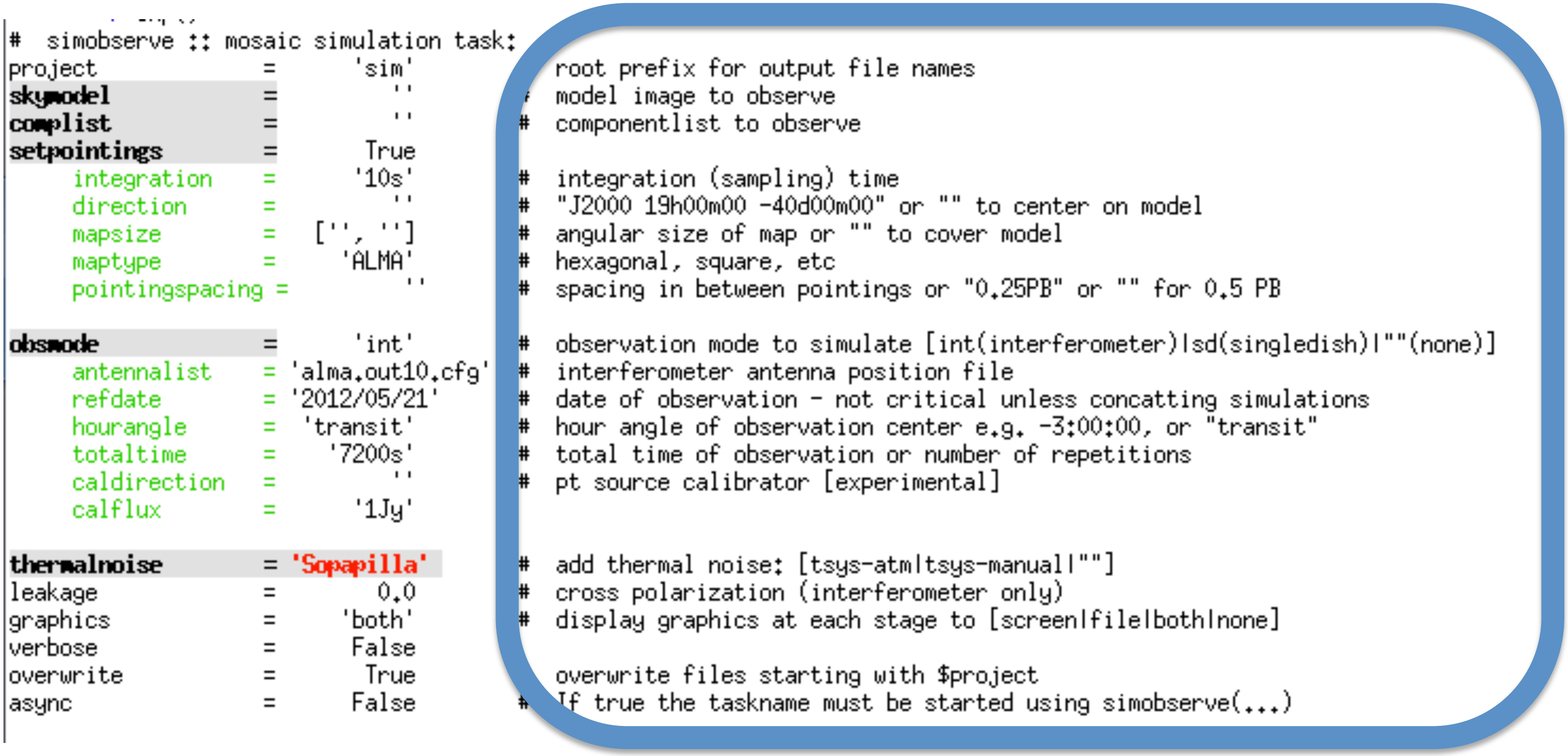
```
#  simobserve ::  saic simulation task:
project            =    'myproj'        #  et prefix for output file names
skymodel           =          ''        #  odel image to observe
complist           =          ''        #  omponentlist to observe
setpointings       =        True
     integration   =       '10s'        #  ntegration (sampling) time
     direction     =          ''        #  J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize       =    ['', '']        #  ngular size of map or "" to cover model
     mapstype      =      'ALMA'        #  exagonal, square, etc
     pointingspacing =        ''        #  pacing in between pointings or "0.25PB" or "" for 0.5 PB

obsmode            =       'int'        #  bservation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist   = 'alma.out10.cfg'   #  nterferometer antenna position file
     refdate       = '2012/05/21'       #  ate of observation - not critical unless concatting simulations
     hourangle     =   'transit'        #  our angle of observation center e.g. -3:00:00, or "transit"
     totaltime     =     '7200s'        #  otal time of observation or number of repetitions
     caldirection  =          ''        #  t source calibrator [experimental]
     calflux       =       '1Jy'

thermalnoise       =          ''        #  dd thermal noise: [tsys-atm|tsys-manual|""]
leakage            =         0.0        #  oss polarization (interferometer only)
graphics           =      'both'        #  isplay graphics at each stage to [screen|file|both|none]
verbose            =       False
overwrite          =        True        #  verwrite files starting with $project
async              =       False        #  f true the taskname must be started using simobserve(...)
```

Valid Value

Default Value

CASA Refresher

- `inp` shows brief description



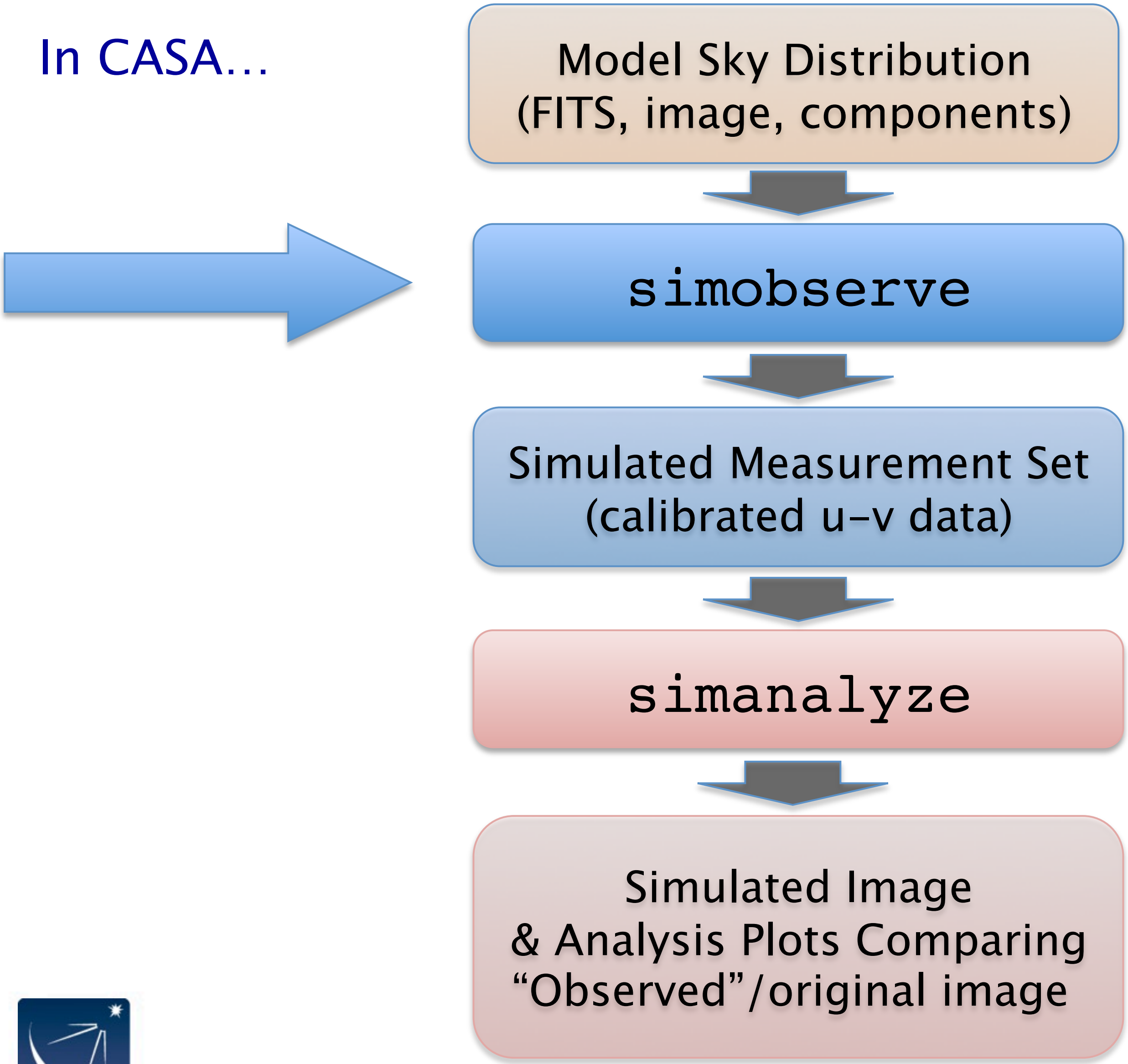
```
#  simobserve :: mosaic simulation task:
project            =      'sim'          root prefix for output file names
skymodel           =         ''        # model image to observe
complist           =         ''        # componentlist to observe
setpointings       =       True
     integration   =      '10s'        # integration (sampling) time
     direction     =         ''        # "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize       =   ['', '']        # angular size of map or "" to cover model
     maptype       =     'ALMA'        # hexagonal, square, etc
     pointingspacing =        ''       # spacing in between pointings or "0.25PB" or "" for 0.5 PB

obsmode            =      'int'        # observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist   = 'alma.out10.cfg'  # interferometer antenna position file
     refdate       = '2012/05/21'      # date of observation - not critical unless concatting simulations
     hourangle     =  'transit'        # hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime     =    '7200s'        # total time of observation or number of repetitions
     caldirection  =         ''        # pt source calibrator [experimental]
     calflux       =      '1Jy'

thermalnoise       = 'Sopapilla'       # add thermal noise: [tsys-atm|tsys-manual|""]
leakage            =        0.0        # cross polarization (interferometer only)
graphics           =     'both'        # display graphics at each stage to [screen|file|both|none]
verbose            =      False
overwrite          =       True          overwrite files starting with $project
async              =      False        # If true the taskname must be started using simobserve(...)
```

CASA Refresher

- When all parameters are set, execute with “`go simobserve`”
- If you get stuck:
  - Type “`tasklist`” to see all tasks
  - Type “`help taskname`” to get help on taskname
  - Type “`default taskname`” to set the default inputs
  - Type “`inp`” to review the inputs of the current task
  - Ask!

Basic Simulation Workflow

In CASA…



Model Sky Distribution
(FITS, image, components)

`simobserve`

Simulated Measurement Set
(calibrated u-v data)

`simanalyze`

Simulated Image
& Analysis Plots Comparing
“Observed”/original image

What Defines a Simulation?

Model Sky Distribution
(Required)

What does the sky really look like in your field?

Telescope
(Required)

Number of Antennas, Configuration, Diameter

Observation
(Required)

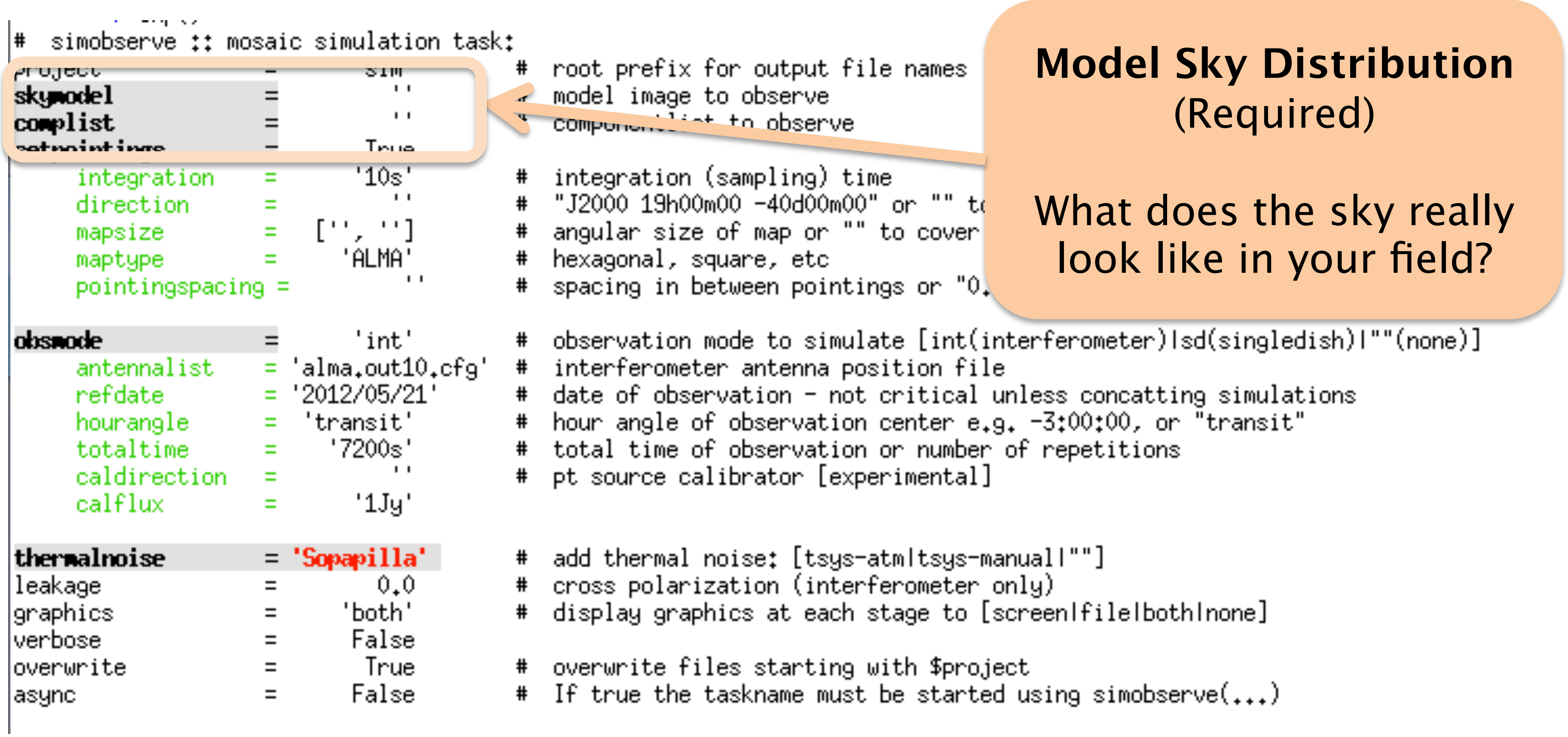
Integration time, scan length, pointing centers

Corruption
(Optional)

Thermal noise, phase noise, polarization leakage

simobserve

- Model sky distribution as FITS file or “component list”



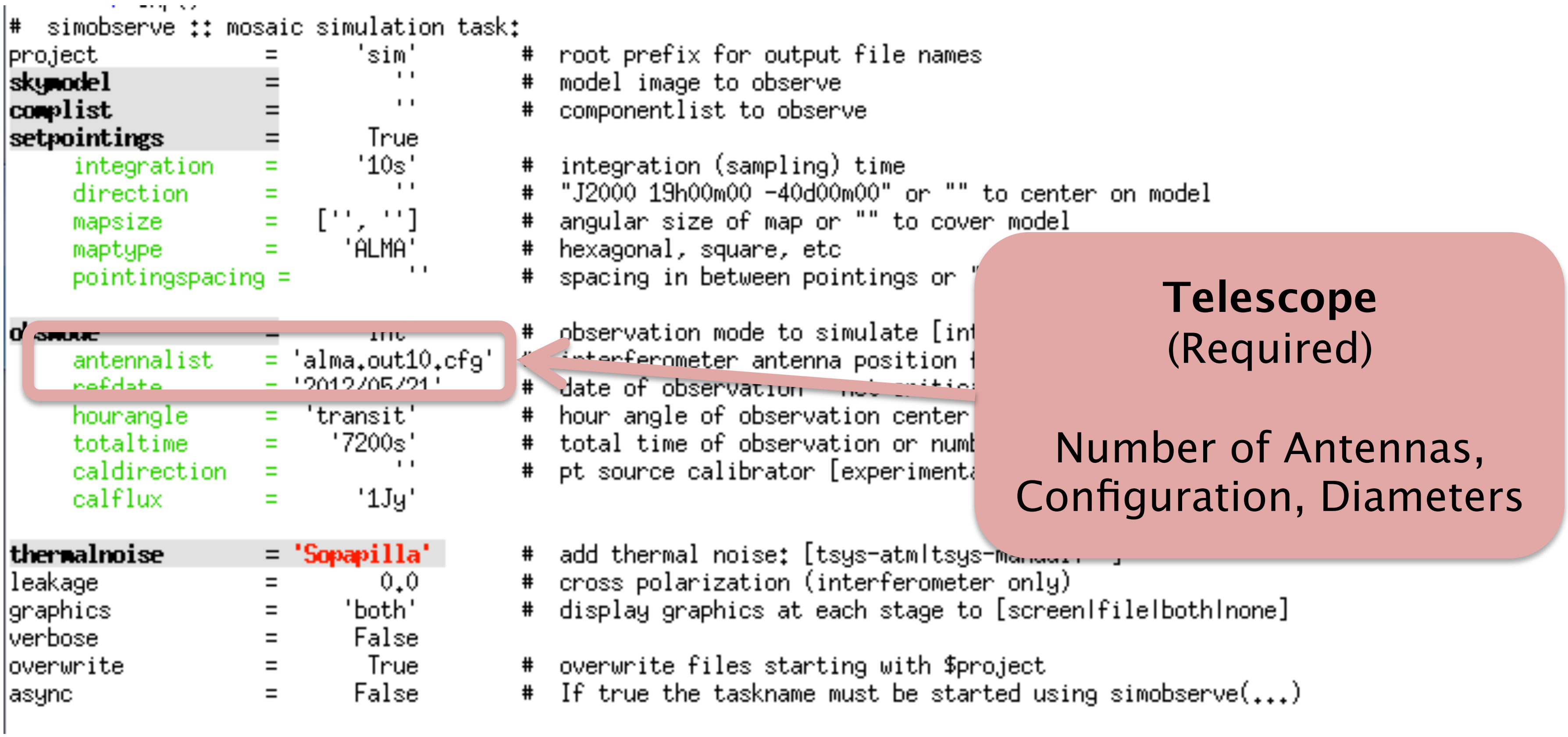
```
#  simobserve :: mosaic simulation task:
project             =      'sim'        #  root prefix for output file names
skymodel            =         ''        #  model image to observe
complist            =         ''        #  componentlist to observe
setpointings        =       True
     integration    =      '10s'        #  integration (sampling) time
     direction      =         ''        #  "J2000 19h00m00 -40d00m00" or "" to
     mapsize        =   ['', '']        #  angular size of map or "" to cover
     maptype        =     'ALMA'        #  hexagonal, square, etc
     pointingspacing =         ''       #  spacing in between pointings or "0.

obsmode             =      'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist    = 'alma.out10.cfg'  #  interferometer antenna position file
     refdate        = '2012/05/21'      #  date of observation - not critical unless concatting simulations
     hourangle      =  'transit'        #  hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime      =    '7200s'        #  total time of observation or number of repetitions
     caldirection   =         ''        #  pt source calibrator [experimental]
     calflux        =      '1Jy'

thermalnoise        = 'Sopapilla'       #  add thermal noise: [tsys-atm|tsys-manual|""]
leakage             =        0.0        #  cross polarization (interferometer only)
graphics            =     'both'        #  display graphics at each stage to [screen|file|both|none]
verbose             =      False
overwrite           =       True        #  overwrite files starting with $project
async               =      False        #  If true the taskname must be started using simobserve(...)
```

Model Sky Distribution
(Required)

What does the sky really look like in your field?

simobserve

- Telescope via configuration file.



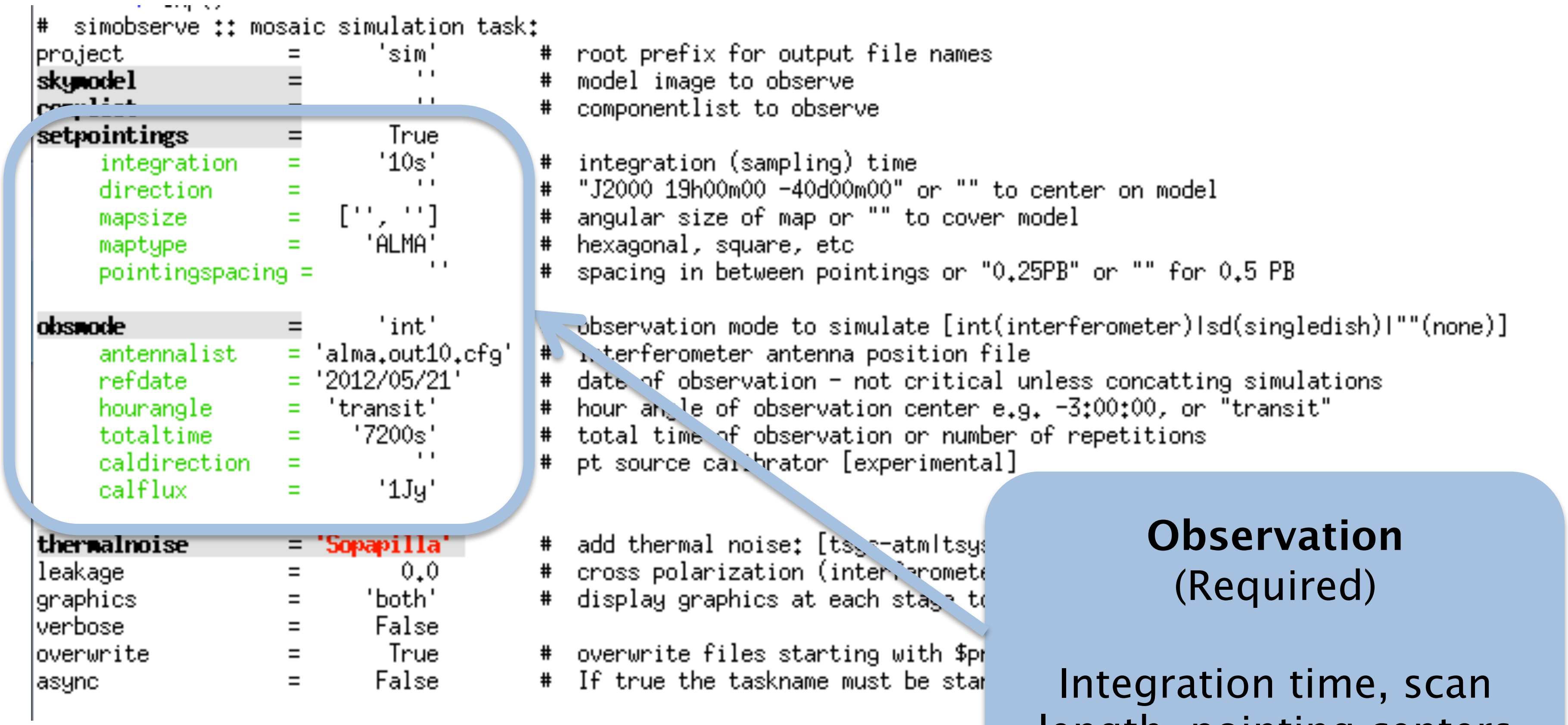
```
#  simobserve :: mosaic simulation task:
project            =      'sim'        #  root prefix for output file names
skymodel           =         ''        #  model image to observe
complist           =         ''        #  componentlist to observe
setpointings       =       True
     integration   =      '10s'        #  integration (sampling) time
     direction     =         ''        #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize       =   ['', '']        #  angular size of map or "" to cover model
     maptype       =     'ALMA'        #  hexagonal, square, etc
     pointingspacing =        ''       #  spacing in between pointings or '

obsmode            =      'int'        #  observation mode to simulate [int
     antennalist   = 'alma.out10.cfg'  #  interferometer antenna position f
     refdate       = '2012/05/21'      #  date of observation - not critic
     hourangle     =  'transit'        #  hour angle of observation center
     totaltime     =    '7200s'        #  total time of observation or numb
     caldirection  =         ''        #  pt source calibrator [experimenta
     calflux       =      '1Jy'

thermalnoise       = 'Sopapilla'       #  add thermal noise: [tsys-atm|tsys-manual|'']
leakage            =        0.0        #  cross polarization (interferometer only)
graphics           =     'both'        #  display graphics at each stage to [screen|file|both|none]
verbose            =      False
overwrite          =       True        #  overwrite files starting with $project
async              =      False        #  If true the taskname must be started using simobserve(...)
```

Telescope
(Required)

Number of Antennas, Configuration, Diameters

simobserve

- Observations defined via `setpointings` and `obsmode`



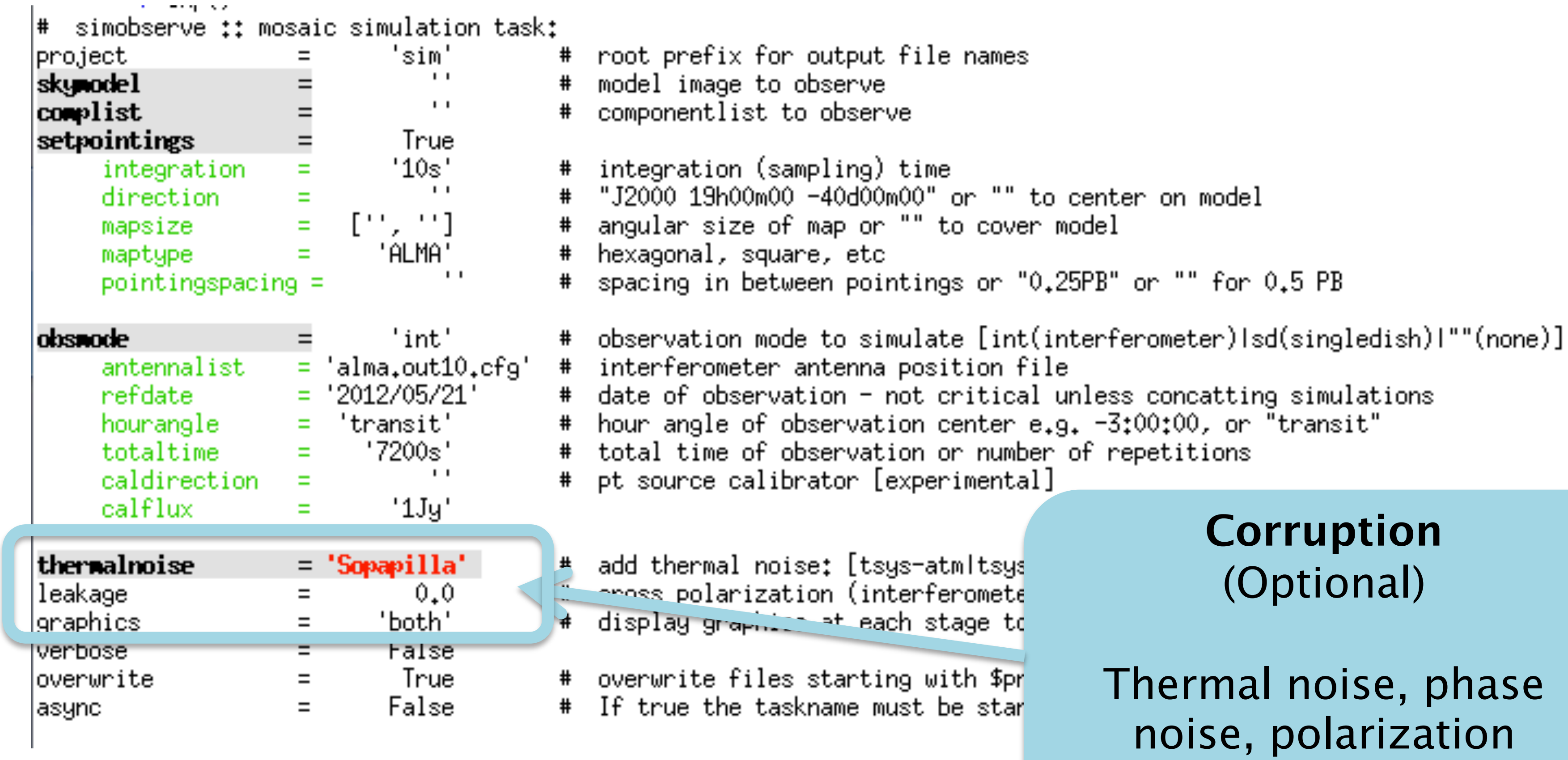
```
#  simobserve :: mosaic simulation task:
project            =      'sim'        #  root prefix for output file names
skymodel           =         ''        #  model image to observe
complist           =         ''        #  componentlist to observe
setpointings       =       True
     integration   =      '10s'        #  integration (sampling) time
     direction     =         ''        #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize       =   ['', '']        #  angular size of map or "" to cover model
     maptype       =     'ALMA'        #  hexagonal, square, etc
     pointingspacing =        ''       #  spacing in between pointings or "0.25PB" or "" for 0.5 PB

obsmode            =      'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist   = 'alma.out10.cfg'  #  interferometer antenna position file
     refdate       = '2012/05/21'      #  date of observation - not critical unless concatting simulations
     hourangle     =  'transit'        #  hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime     =    '7200s'        #  total time of observation or number of repetitions
     caldirection  =         ''        #  pt source calibrator [experimental]
     calflux       =      '1Jy'

thermalnoise       = 'Sopapilla'       #  add thermal noise: [tsys-atm|tsys
leakage            =        0.0        #  cross polarization (interferomete
graphics           =     'both'        #  display graphics at each stage to
verbose            =      False
overwrite          =       True        #  overwrite files starting with $pr
async              =      False        #  If true the taskname must be star
```

Observation
(Required)

Integration time, scan length, pointing centers

simobserve

- Corruption with `thermalnoise` & toolkit



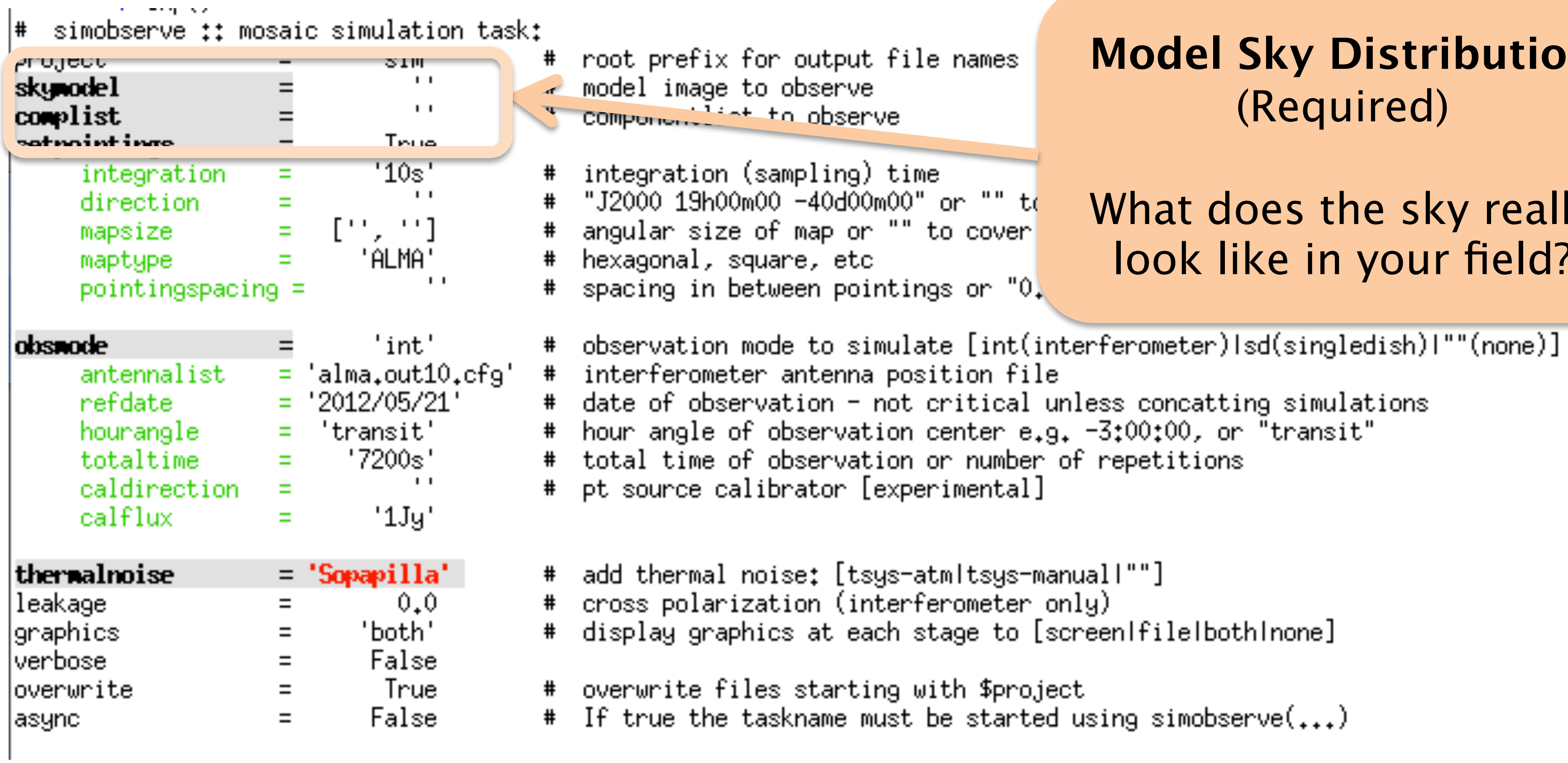
```
#  simobserve :: mosaic simulation task:
project            =      'sim'        #  root prefix for output file names
skymodel           =         ''        #  model image to observe
complist           =         ''        #  componentlist to observe
setpointings       =       True
     integration   =      '10s'        #  integration (sampling) time
     direction     =         ''        #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize       =   ['', '']        #  angular size of map or "" to cover model
     maptype       =     'ALMA'        #  hexagonal, square, etc
     pointingspacing =        ''       #  spacing in between pointings or "0.25PB" or "" for 0.5 PB

obsmode            =      'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist   = 'alma.out10.cfg'  #  interferometer antenna position file
     refdate       = '2012/05/21'      #  date of observation - not critical unless concatting simulations
     hourangle     =  'transit'        #  hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime     =    '7200s'        #  total time of observation or number of repetitions
     caldirection  =         ''        #  pt source calibrator [experimental]
     calflux       =      '1Jy'

thermalnoise       = 'Sopapilla'       #  add thermal noise: [tsys-atm|tsys
leakage            =        0.0        #  cross polarization (interferomete
graphics           =     'both'        #  display graphics at each stage to
verbose            =      False
overwrite          =       True        #  overwrite files starting with $pr
async              =      False        #  If true the taskname must be star
```

Corruption
(Optional)

Thermal noise, phase noise, polarization leakage

simobserve

- Model sky distribution as FITS file or “component list”



```
#  simobserve :: mosaic simulation task:
project            =      'sim'        #  root prefix for output file names
skymodel           =         ''        #  model image to observe
complist           =         ''        #  componentlist to observe
setpointings       =       True
     integration   =      '10s'        #  integration (sampling) time
     direction     =         ''        #  "J2000 19h00m00 -40d00m00" or "" to
     mapsize       =    ['', '']       #  angular size of map or "" to cover
     maptype       =     'ALMA'        #  hexagonal, square, etc
     pointingspacing =         ''      #  spacing in between pointings or "0.

obsmode            =      'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist   = 'alma.out10.cfg'  #  interferometer antenna position file
     refdate       = '2012/05/21'      #  date of observation - not critical unless concatting simulations
     hourangle     =  'transit'        #  hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime     =    '7200s'        #  total time of observation or number of repetitions
     caldirection  =         ''        #  pt source calibrator [experimental]
     calflux       =      '1Jy'

thermalnoise       = 'Sopapilla'       #  add thermal noise: [tsys-atm|tsys-manual|""]
leakage            =        0.0        #  cross polarization (interferometer only)
graphics           =     'both'        #  display graphics at each stage to [screen|file|both|none]
verbose            =      False
overwrite          =       True        #  overwrite files starting with $project
async              =      False        #  If true the taskname must be started using simobserve(...)
```

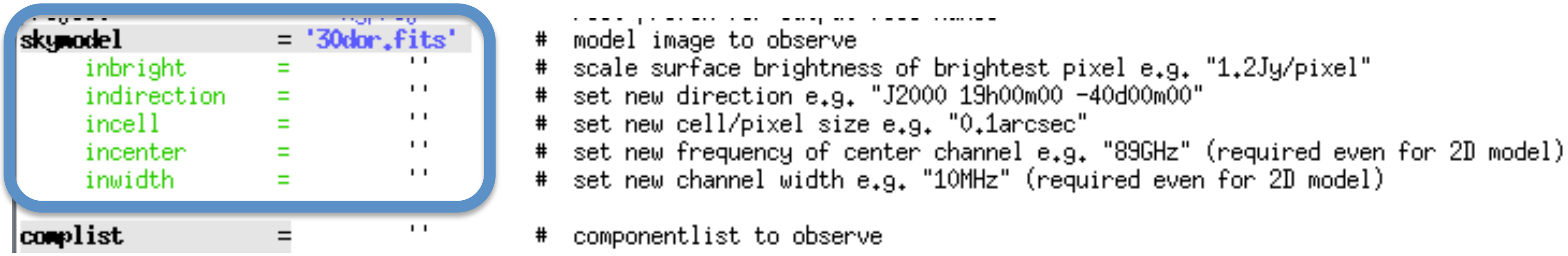
Model Sky Distribution
(Required)

What does the sky really look like in your field?

Input Sky Model

- Model sky distribution as FITS file. `simobserve` needs:
  - Coordinates
  - Brightness units
  - Pixel scale (angular and spectral)
  - Stokes axis (optional)

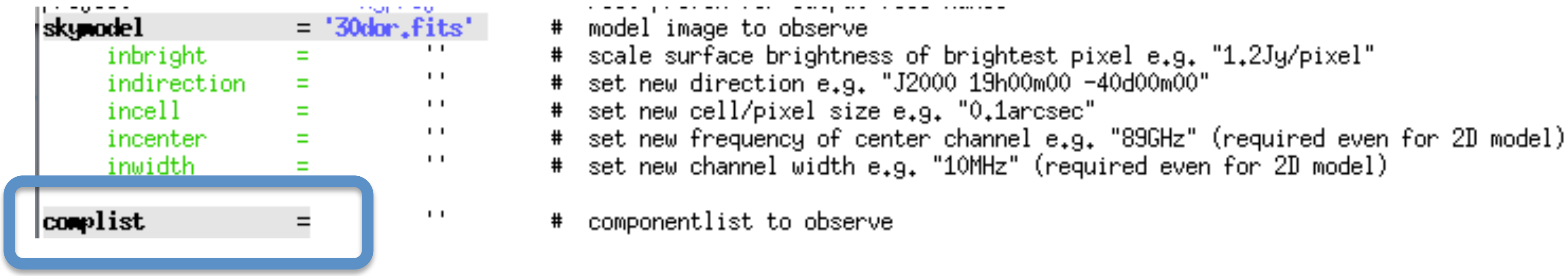
- These may be specified in your FITS header or supplied/over-written by `simobserve`.



```
skymodel        =  '30dor.fits'        #  model image to observe
     inbright   =             ''       #  scale surface brightness of brightest pixel e.g. "1.2Jy/pixel"
     indirection =            ''       #  set new direction e.g. "J2000 19h00m00 -40d00m00"
     incell     =             ''       #  set new cell/pixel size e.g. "0.1arcsec"
     incenter   =             ''       #  set new frequency of center channel e.g. "89GHz" (required even for 2D model)
     inwidth    =             ''       #  set new channel width e.g. "10MHz" (required even for 2D model)

complist        =             ''       #  componentlist to observe
```

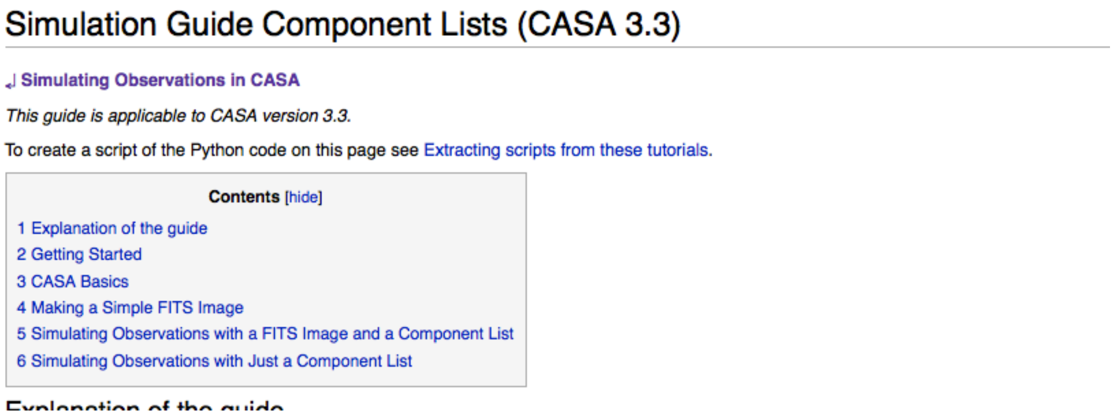
Input Sky Model

- Alternatively, supply a Gaussian “component list.”



```
skymodel        =  '30dor.fits'   #  model image to observe
     inbright   =            ''   #  scale surface brightness of brightest pixel e.g. "1.2Jy/pixel"
     indirection =           ''   #  set new direction e.g. "J2000 19h00m00 -40d00m00"
     incell     =            ''   #  set new cell/pixel size e.g. "0.1arcsec"
     incenter   =            ''   #  set new frequency of center channel e.g. "89GHz" (required even for 2D model)
     inwidth    =            ''   #  set new channel width e.g. "10MHz" (required even for 2D model)

complist        =            ''   #  componentlist to observe
```

- Example at: http://casaguides.nrao.edu/index.php?title=Simulation_Guide_Component_Lists_(CASA_3.3)



Simulation Guide Component Lists (CASA 3.3)

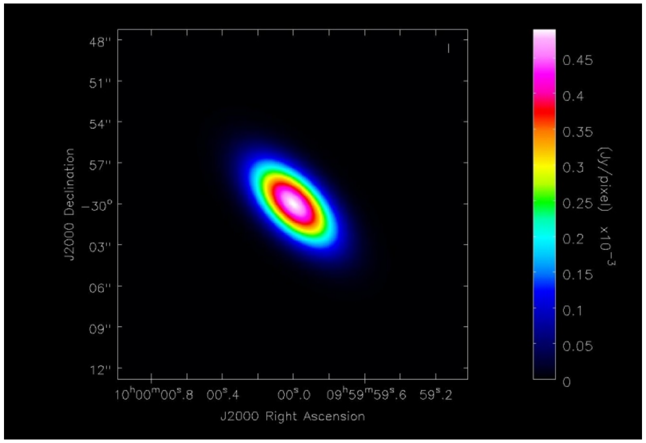
↲ **Simulating Observations in CASA**

This guide is applicable to CASA version 3.3.

To create a script of the Python code on this page see Extracting scripts from these tutorials.

Contents [hide]

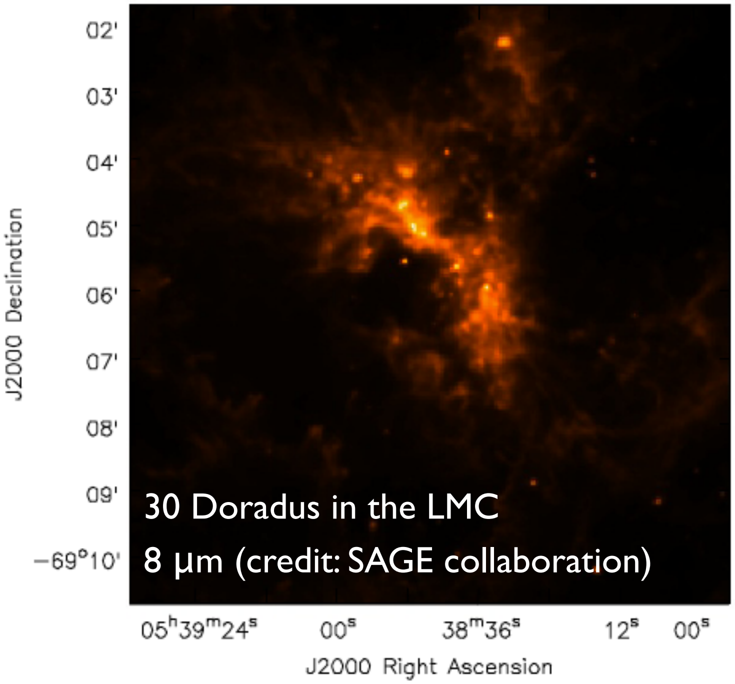
Explanation of the guide



Simple Example

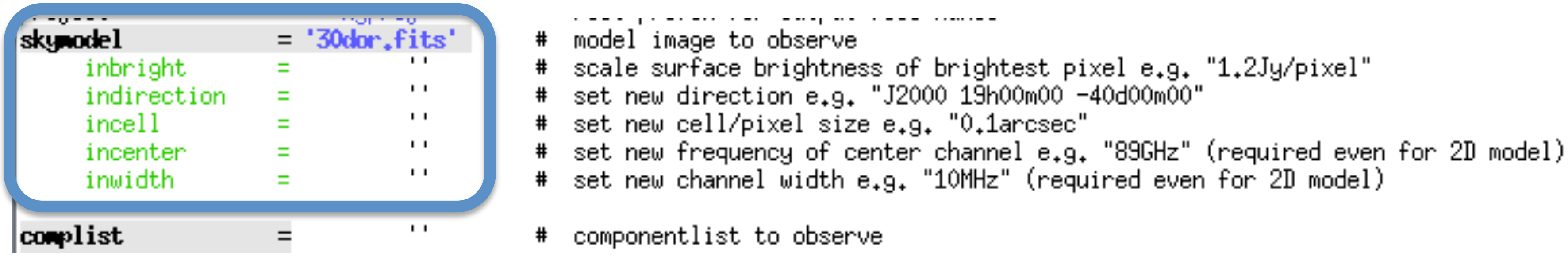
- Simulate observing 1mm dust continuum in a 30-Doradus (LMC)-like region at the distance of M31/M33 (800 kpc).
- We have a near-IR image of 30 Doradus, will need to:
  - Scale the brightness and observing frequency
  - Adjust the pixel scale (move it from 50-800 kpc)
  - Set a new position
  - Define the observations INTEGRATION TIME, TELESCOPE, ETC.

Our Model



30 Doradus in the LMC
8 µm (credit: SAGE collaboration)

Simple Example



```
skymodel         =  '30dor.fits'   #  model image to observe
     inbright    =           ''    #  scale surface brightness of brightest pixel e.g. "1.2Jy/pixel"
     indirection =           ''    #  set new direction e.g. "J2000 19h00m00 -40d00m00"
     incell      =           ''    #  set new cell/pixel size e.g. "0.1arcsec"
     incenter    =           ''    #  set new frequency of center channel e.g. "89GHz" (required even for 2D model)
     inwidth     =           ''    #  set new channel width e.g. "10MHz" (required even for 2D model)

complist         =           ''    #  componentlist to observe
```

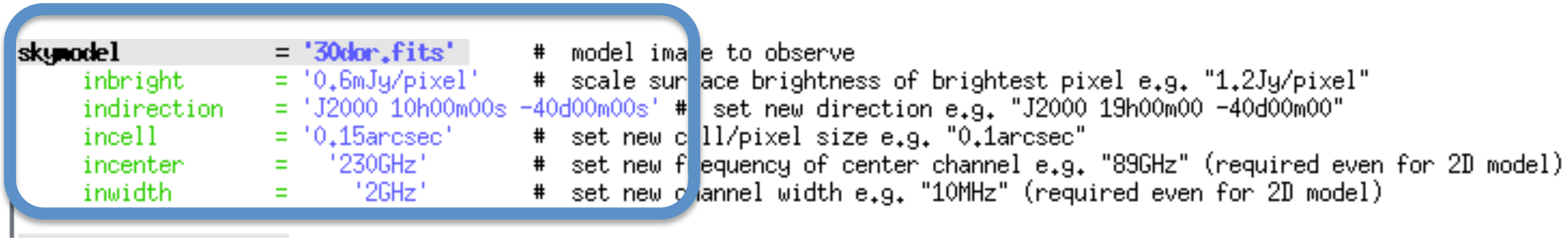
- inbright = “0.6mJy/pixel”
  Requires spectral model/other knowledge to estimate (Science!)

- Indirection = “J2000 10h00m00s -40d00m00s”

- incell=“0.15arcsec”
  native cell size = 2.3”, moving from 50 kpc→800 kpc scale by 50/800

- incenter=“230GHz”, inwidth=“2GHz”
  Need to supply observing frequency & bandwidth (here 1mm dust continuum)

Simple Example



```
skymodel        = '30dor.fits'        #  model image to observe
     inbright        = '0.6mJy/pixel'     #  scale surface brightness of brightest pixel e.g. "1.2Jy/pixel"
     indirection     = 'J2000 10h00m00s -40d00m00s' #  set new direction e.g. "J2000 19h00m00 -40d00m00"
     incell          = '0.15arcsec'       #  set new cell/pixel size e.g. "0.1arcsec"
     incenter        =   '230GHz'         #  set new frequency of center channel e.g. "89GHz" (required even for 2D model)
     inwidth         =     '2GHz'         #  set new channel width e.g. "10MHz" (required even for 2D model)
```

- inbright = “0.6mJy/pixel”
  Requires spectral model/other knowledge to estimate (Science!)

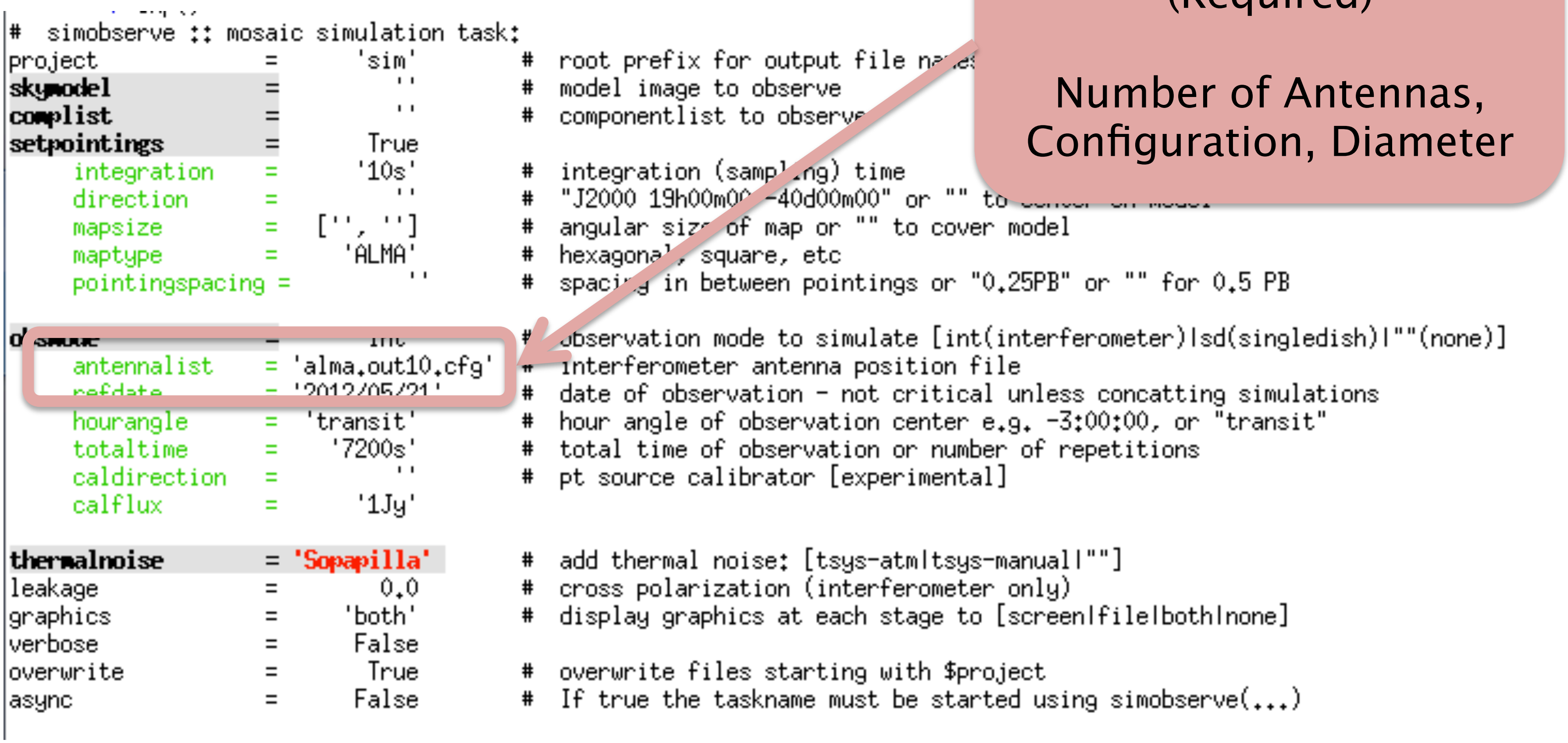
- Indirection = “J2000 10h00m00s -40d00m00s”

- `incell`=“0.15arcsec”
  native cell size = 2.3”, moving from 50 kpc→800 kpc scale by 50/800

- `incenter`=“230GHz”, `inwidth`=“2GHz”
  Need to supply observing frequency & bandwidth (here 1mm dust continuum)

simobserve

- Telescope via configuration file.



```
#  simobserve :: mosaic simulation task:
project            =      'sim'        #  root prefix for output file names
skymodel           =         ''        #  model image to observe
complist           =         ''        #  componentlist to observe
setpointings       =       True
     integration   =      '10s'        #  integration (sampling) time
     direction     =         ''        #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize       =   ['', '']        #  angular size of map or "" to cover model
     maptype       =     'ALMA'        #  hexagonal, square, etc
     pointingspacing =        ''       #  spacing in between pointings or "0.25PB" or "" for 0.5 PB

obsmode            =      'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist   = 'alma.out10.cfg'  #  interferometer antenna position file
     refdate       = '2012/05/21'      #  date of observation - not critical unless concatting simulations
     hourangle     =  'transit'        #  hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime     =    '7200s'        #  total time of observation or number of repetitions
     caldirection  =         ''        #  pt source calibrator [experimental]
     calflux       =      '1Jy'

thermalnoise       = 'Sopapilla'       #  add thermal noise: [tsys-atm|tsys-manual|""]
leakage            =        0.0        #  cross polarization (interferometer only)
graphics           =     'both'        #  display graphics at each stage to [screen|file|both|none]
verbose            =      False
overwrite          =       True        #  overwrite files starting with $project
async              =      False        #  If true the taskname must be started using simobserve(...)
```

Telescope
(Required)

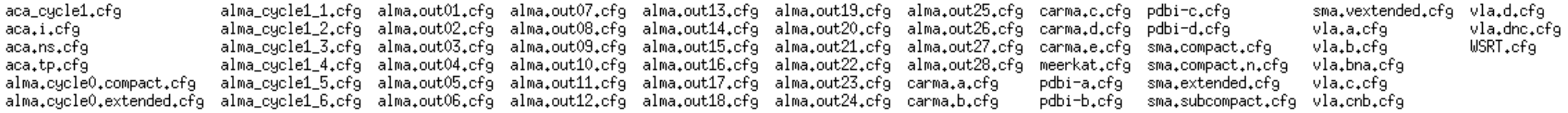
Number of Antennas,
Configuration, Diameter

Configuration Files

- Define telescope array for `simobserve`.

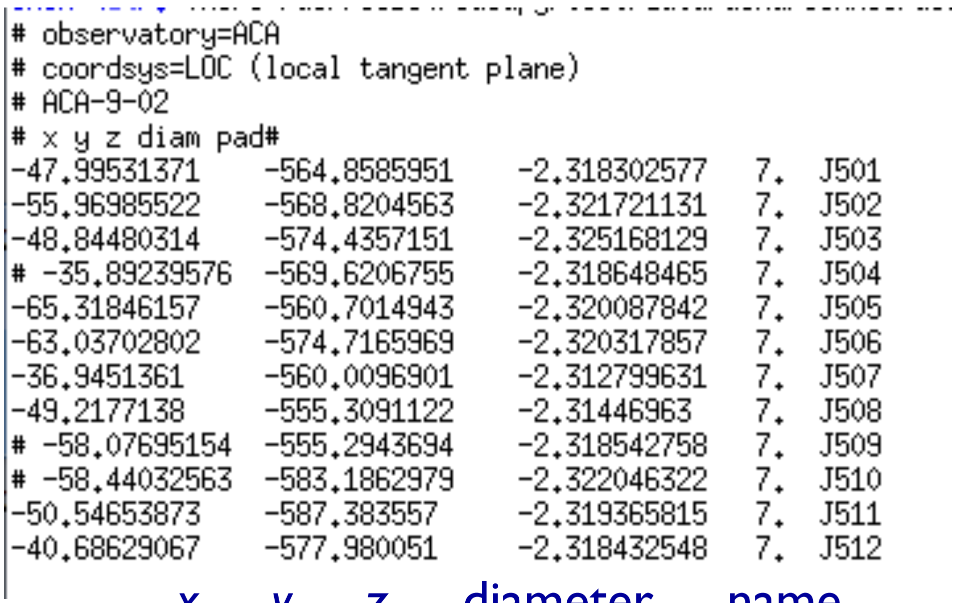
Config Files in CASA Already

ALMA, JVLA, CARMA, SMA, etc.



```
aca_cycle1.cfg              alma_cycle1_1.cfg  alma.out01.cfg  alma.out07.cfg  alma.out13.cfg  alma.out19.cfg  alma.out25.cfg  carma.c.cfg  pdbi-c.cfg           sma.vextended.cfg  vla.d.cfg
aca.i.cfg                   alma_cycle1_2.cfg  alma.out02.cfg  alma.out08.cfg  alma.out14.cfg  alma.out20.cfg  alma.out26.cfg  carma.d.cfg  pdbi-d.cfg           vla.a.cfg          vla.dnc.cfg
aca.ns.cfg                  alma_cycle1_3.cfg  alma.out03.cfg  alma.out09.cfg  alma.out15.cfg  alma.out21.cfg  alma.out27.cfg  carma.e.cfg  sma.compact.cfg      vla.b.cfg          WSRT.cfg
aca.tp.cfg                  alma_cycle1_4.cfg  alma.out04.cfg  alma.out10.cfg  alma.out16.cfg  alma.out22.cfg  alma.out28.cfg  meerkat.cfg  sma.compact.n.cfg    vla.bna.cfg
alma.cycle0.compact.cfg     alma_cycle1_5.cfg  alma.out05.cfg  alma.out11.cfg  alma.out17.cfg  alma.out23.cfg  carma.a.cfg     pdbi-a.cfg   sma.extended.cfg     vla.c.cfg
alma.cycle0.extended.cfg    alma_cycle1_6.cfg  alma.out06.cfg  alma.out12.cfg  alma.out18.cfg  alma.out24.cfg  carma.b.cfg     pdbi-b.cfg   sma.subcompact.cfg   vla.cnb.cfg
```

Example Config File: ALMA Cycle 1 ACA

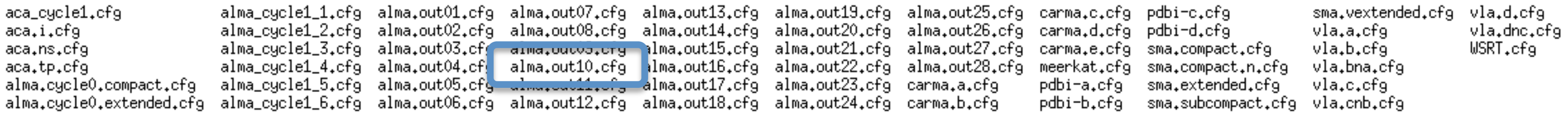


```
# observatory=ACA
# coordsys=LOC (local tangent plane)
# ACA-9-02
# x y z diam pad#
-47.99531371    -564.8585951    -2.318302577    7.   J501
-55.96985522    -568.8204563    -2.321721131    7.   J502
-48.84480314    -574.4357151    -2.325168129    7.   J503
# -35.89239576  -569.6206755    -2.318648465    7.   J504
-65.31846157    -560.7014943    -2.320087842    7.   J505
-63.03702802    -574.7165969    -2.320317857    7.   J506
-36.9451361     -560.0096901    -2.312799631    7.   J507
-49.2177138     -555.3091122    -2.31446963     7.   J508
# -58.07695154  -555.2943694    -2.318542758    7.   J509
# -58.44032563  -583.1862979    -2.322046322    7.   J510
-50.54653873    -587.383557     -2.319365815    7.   J511
-40.68629067    -577.980051     -2.318432548    7.   J512
```

x   *y*   *z*   diameter   name

Configuration Files

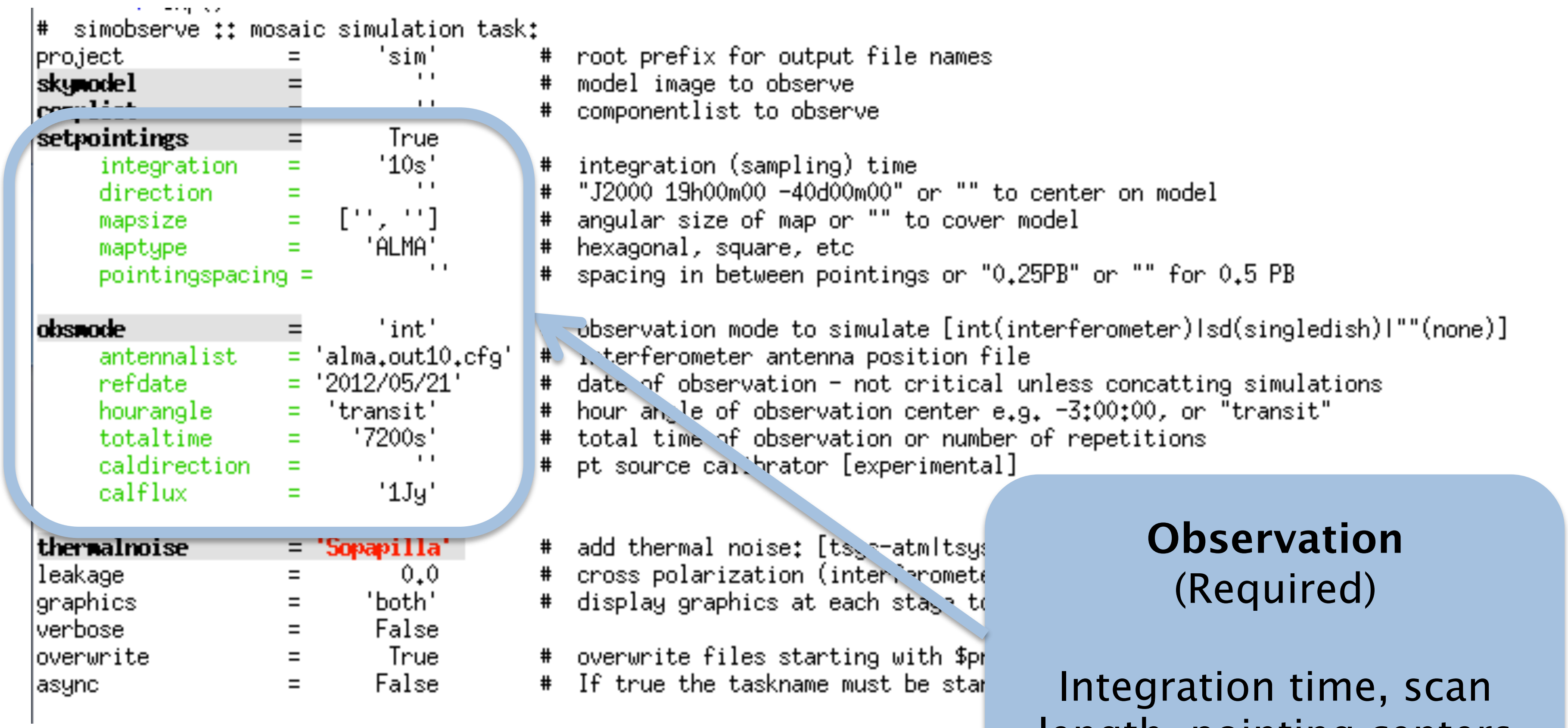
- Pick an intermediate-extent full-ALMA configuration



```
aca_cycle1.cfg           alma_cycle1_1.cfg alma.out01.cfg alma.out07.cfg alma.out13.cfg alma.out19.cfg alma.out25.cfg carma.c.cfg pdbi-c.cfg         sma.vextended.cfg vla.d.cfg
aca.i.cfg                alma_cycle1_2.cfg alma.out02.cfg alma.out08.cfg alma.out14.cfg alma.out20.cfg alma.out26.cfg carma.d.cfg pdbi-d.cfg         vla.a.cfg         vla.dnc.cfg
aca.ns.cfg               alma_cycle1_3.cfg alma.out03.cfg alma.out09.cfg alma.out15.cfg alma.out21.cfg alma.out27.cfg carma.e.cfg sma.compact.cfg    vla.b.cfg         WSRT.cfg
aca.tp.cfg               alma_cycle1_4.cfg alma.out04.cfg alma.out10.cfg alma.out16.cfg alma.out22.cfg alma.out28.cfg meerkat.cfg sma.compact.n.cfg  vla.bna.cfg
alma.cycle0.compact.cfg  alma_cycle1_5.cfg alma.out05.cfg alma.out11.cfg alma.out17.cfg alma.out23.cfg carma.a.cfg    pdbi-a.cfg  sma.extended.cfg   vla.c.cfg
alma.cycle0.extended.cfg alma_cycle1_6.cfg alma.out06.cfg alma.out12.cfg alma.out18.cfg alma.out24.cfg carma.b.cfg    pdbi-b.cfg  sma.subcompact.cfg vla.cnb.cfg
```

simobserve

- Observations defined via `setpointings` and `obsmode`



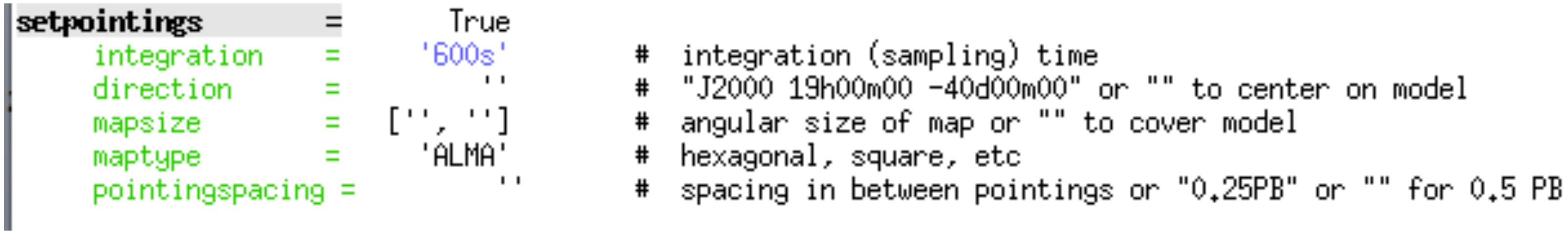
```
#  simobserve :: mosaic simulation task:
project            =      'sim'        #  root prefix for output file names
skymodel           =         ''        #  model image to observe
complist           =         ''        #  componentlist to observe
setpointings       =       True
     integration   =      '10s'        #  integration (sampling) time
     direction     =         ''        #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize       =   ['', '']        #  angular size of map or "" to cover model
     maptype       =     'ALMA'        #  hexagonal, square, etc
     pointingspacing =        ''       #  spacing in between pointings or "0.25PB" or "" for 0.5 PB

obsmode            =      'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist   = 'alma.out10.cfg'  #  interferometer antenna position file
     refdate       = '2012/05/21'      #  date of observation - not critical unless concatting simulations
     hourangle     =  'transit'        #  hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime     =    '7200s'        #  total time of observation or number of repetitions
     caldirection  =         ''        #  pt source calibrator [experimental]
     calflux       =      '1Jy'

thermalnoise       = 'Sopapilla'       #  add thermal noise: [tsys-atm|tsys
leakage            =        0.0        #  cross polarization (interferomete
graphics           =     'both'        #  display graphics at each stage to
verbose            =      False
overwrite          =       True        #  overwrite files starting with $pr
async              =      False        #  If true the taskname must be star
```

Observation
(Required)

Integration time, scan length, pointing centers

setpointings

- setpointings dictates field, integration time, mosaic



```
setpointings        =       True
     integration    =     '600s'        #  integration (sampling) time
     direction      =         ''        #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize        =   ['', '']        #  angular size of map or "" to cover model
     maptype        =     'ALMA'        #  hexagonal, square, etc
     pointingspacing =        ''        #  spacing in between pointings or "0.25PB" or "" for 0.5 PB
```

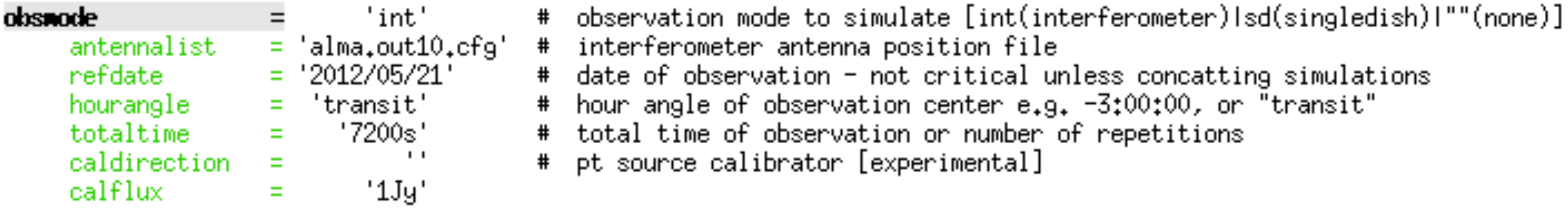
- integration sets data averaging (and field visit) time
  HERE AVERAGING 600S (10M) ENSURES A QUICK INITIAL EXECUTION

- direction sets field or map center

- mapsize, maptype, pointingspacing define a mosaic
  BY DEFAULT IT WILL COVER THE MODEL, HERE THAT MEANS A 9-POINT MOSAIC

obsmode

- `obsmode` sets total time, date, observing sequence



```
obsmode            =      'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist   = 'alma.out10.cfg'  #  interferometer antenna position file
     refdate       = '2012/05/21'      #  date of observation - not critical unless concatting simulations
     hourangle     =  'transit'        #  hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime     =    '7200s'        #  total time of observation or number of repetitions
     caldirection  =         ''        #  pt source calibrator [experimental]
     calflux       =      '1Jy'
```

- `totaltime` sets total observation time
  Here 7200s (2h) is the default value

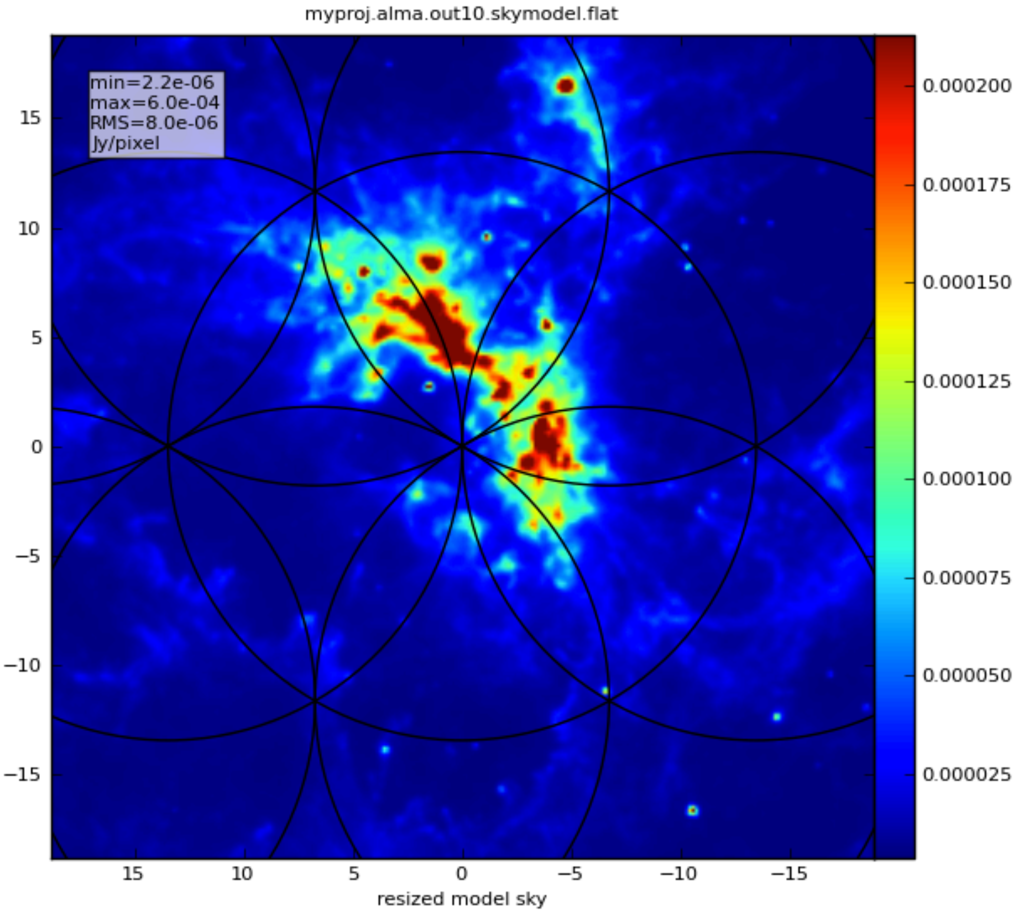
- Optionally specify the date, LST, and a calibrator sequence.

`go simobserve`

simobserve creates a measurement set (MS) in `projectname/projectname.ms`

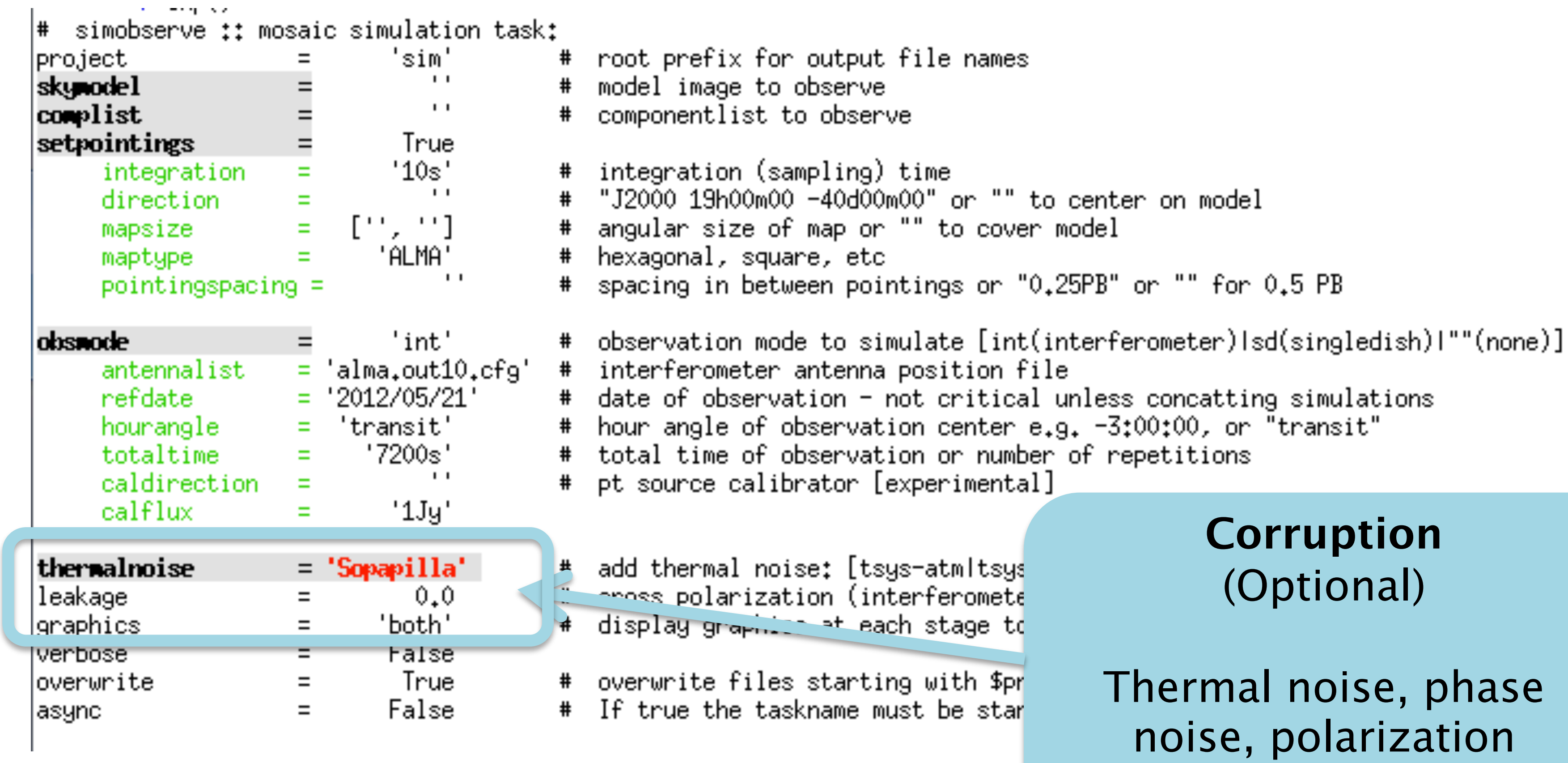
skymodel image

- `simobserve` outputs several files to project directory:
  - projectname.alma.ou
  - projectname.alma.ou
  - projectname.alma.ou
    TEXT FILES SHOW THE LOCA
  - projectname.alma.ou
  - projectname.alma.ou
  - projectname.alma.ou
  - projectname.alma.ou



simobserve

- Corruption with `thermalnoise` & toolkit



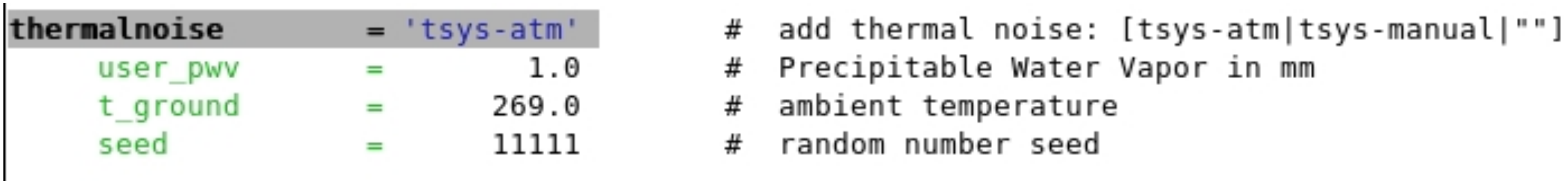
```
#  simobserve :: mosaic simulation task:
project            =      'sim'        #  root prefix for output file names
skymodel           =         ''        #  model image to observe
complist           =         ''        #  componentlist to observe
setpointings       =       True
     integration   =      '10s'        #  integration (sampling) time
     direction     =         ''        #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize       =   ['', '']        #  angular size of map or "" to cover model
     maptype       =     'ALMA'        #  hexagonal, square, etc
     pointingspacing =       ''        #  spacing in between pointings or "0.25PB" or "" for 0.5 PB

obsmode            =      'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist   = 'alma.out10.cfg'  #  interferometer antenna position file
     refdate       = '2012/05/21'      #  date of observation - not critical unless concatting simulations
     hourangle     =  'transit'        #  hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime     =    '7200s'        #  total time of observation or number of repetitions
     caldirection  =         ''        #  pt source calibrator [experimental]
     calflux       =      '1Jy'

thermalnoise       = 'Sopapilla'       #  add thermal noise: [tsys-atm|tsys
leakage            =        0.0        #  cross polarization (interferomete
graphics           =     'both'        #  display graphics at each stage to
verbose            =      False
overwrite          =       True        #  overwrite files starting with $pr
async              =      False        #  If true the taskname must be star
```

Corruption
(Optional)

Thermal noise, phase noise, polarization leakage

thermalnoise

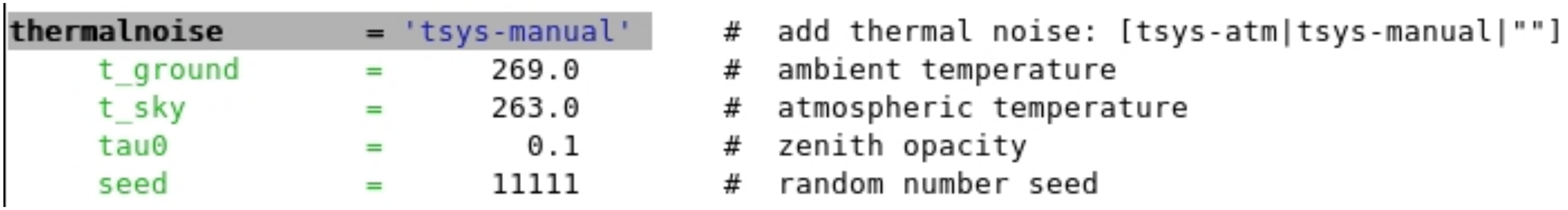
- Set observing conditions to add random noise to image



```
thermalnoise        = 'tsys-atm'        #  add thermal noise: [tsys-atm|tsys-manual|""]
     user_pwv       =          1.0      #  Precipitable Water Vapor in mm
     t_ground       =        269.0      #  ambient temperature
     seed           =        11111      #  random number seed
```

ATM model specific for ALMA site !

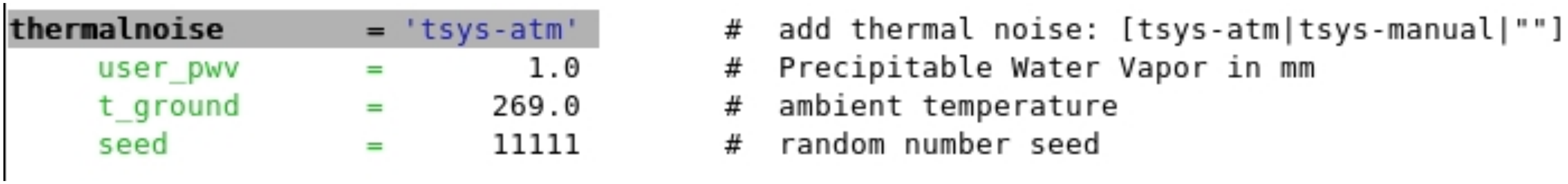
- Use instead…



```
thermalnoise        = 'tsys-manual'     #  add thermal noise: [tsys-atm|tsys-manual|""]
     t_ground       =        269.0      #  ambient temperature
     t_sky          =        263.0      #  atmospheric temperature
     tau0           =          0.1      #  zenith opacity
     seed           =        11111      #  random number seed
```

thermalnoise

- Set observing conditions to add random noise to image



```
thermalnoise        = 'tsys-atm'      #  add thermal noise: [tsys-atm|tsys-manual|""]
     user_pwv       =        1.0      #  Precipitable Water Vapor in mm
     t_ground       =      269.0      #  ambient temperature
     seed           =      11111      #  random number seed
```

ATM MODEL SPECIFIC FOR ALMA SITE !

- See CASAguides and toolkit manual for other ways to corrupt data. (E.G., PHASE NOISE)

  http://casaguides.nrao.edu/index.php?title=Corrupt

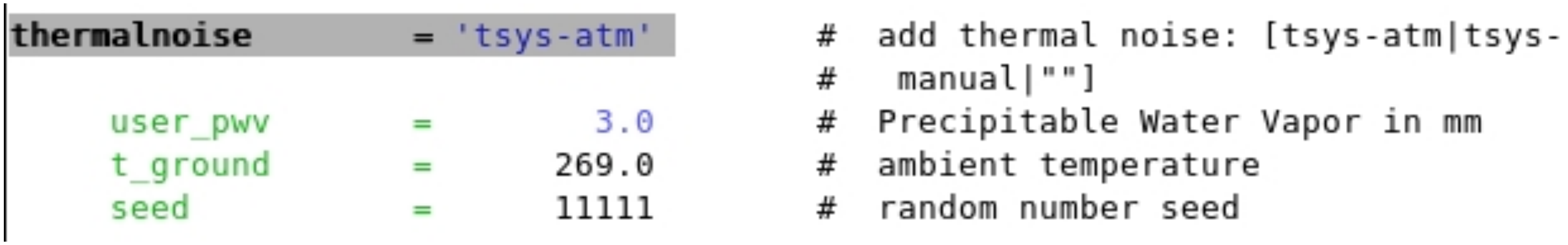
  http://casa.nrao.edu/docs/casaref/CasaRef.html (Simulator tool, sm)

go simobserve

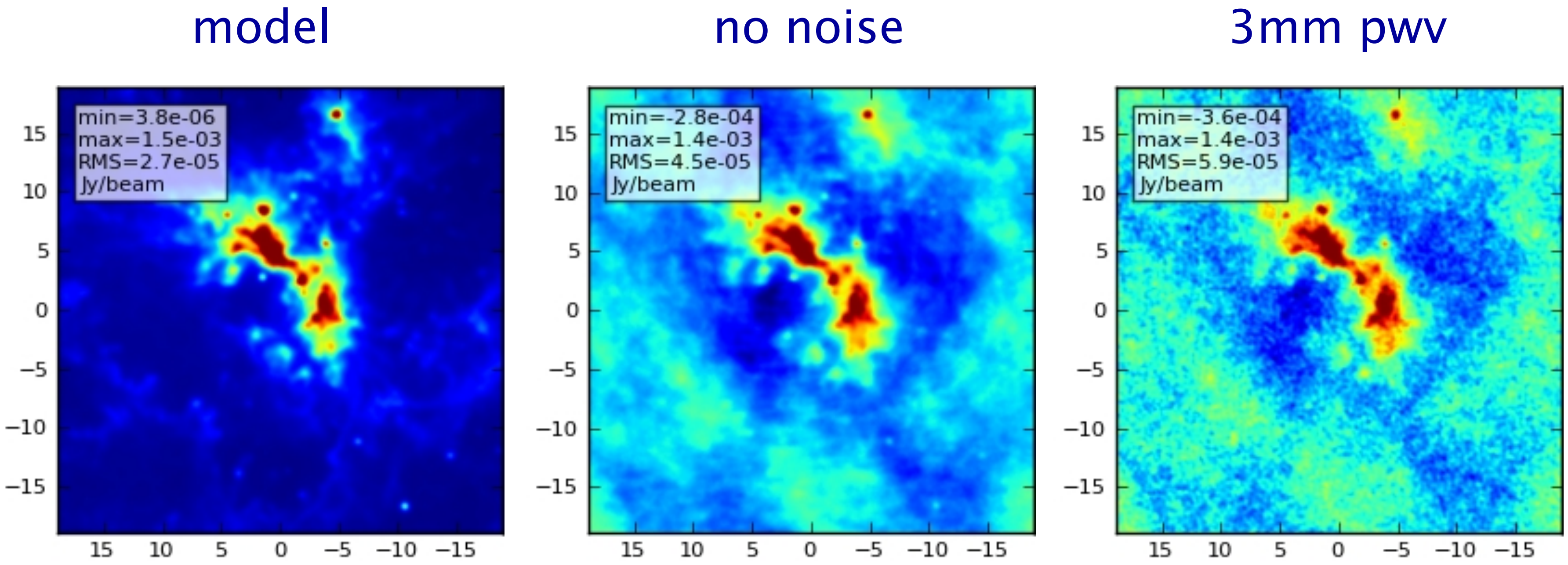
SIMOBSERVE CREATES A NOISY MEASUREMENT SET (MS) IN projectname/ projectname.noisy.ms

thermalnoise

- Set observing conditions to add random noise to image



```
thermalnoise        =  'tsys-atm'          #  add thermal noise: [tsys-atm|tsys-
                                           #   manual|""]
     user_pwv       =         3.0          #  Precipitable Water Vapor in mm
     t_ground       =       269.0          #  ambient temperature
     seed           =       11111          #  random number seed
```



model

no noise

3mm pwv

Multiple sets of observations

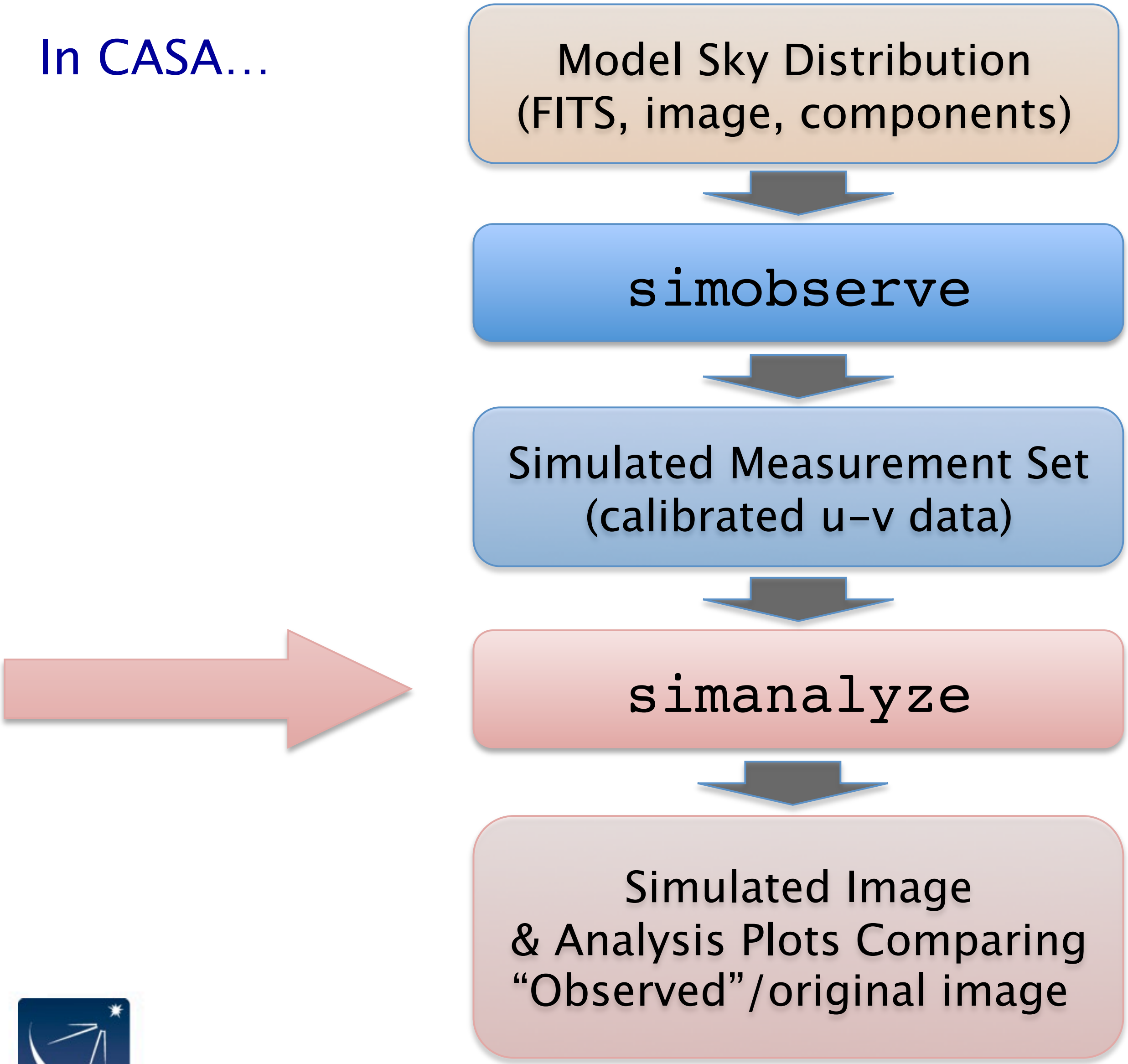
- One can simulate multiple sets of observations with multiple calls to simobserve, to:
  - simulate combining data with different hour angles
  - simulate combining data from different configurations (JVLA A+D), or arrays (ALMA 12m+ACA)
  - simulate combining data from interferometers and single dish telescopes (JVLA+GBT)

- The CLEAN task can take multiple measurement sets to combine interferometric observations

- The FEATHER task can combine single dish and interferometric observations

Basic Simulation Workflow

In CASA…



Model Sky Distribution (FITS, image, components)

`simobserve`

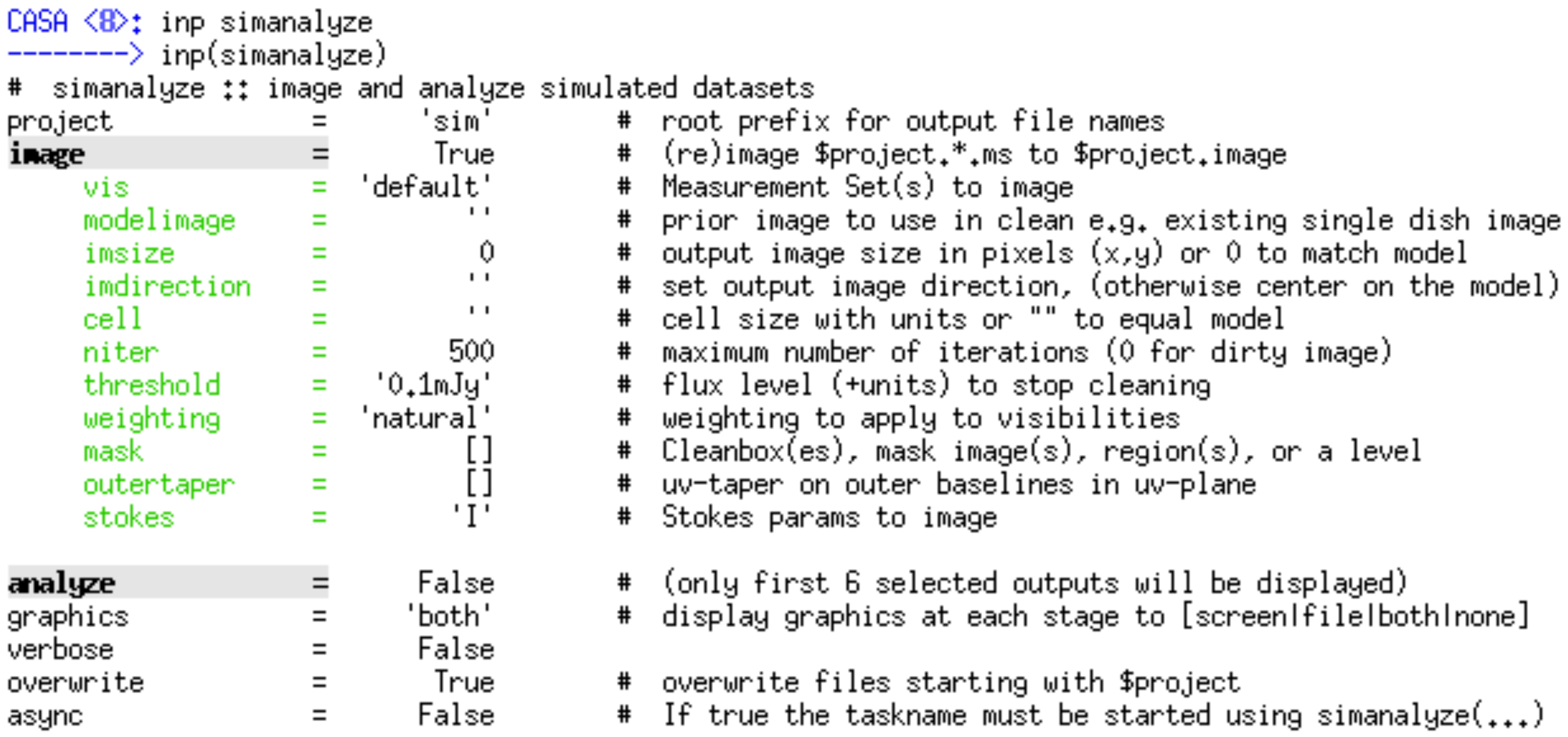
Simulated Measurement Set (calibrated u-v data)

`simanalyze`

Simulated Image & Analysis Plots Comparing “Observed”/original image

simanalyze

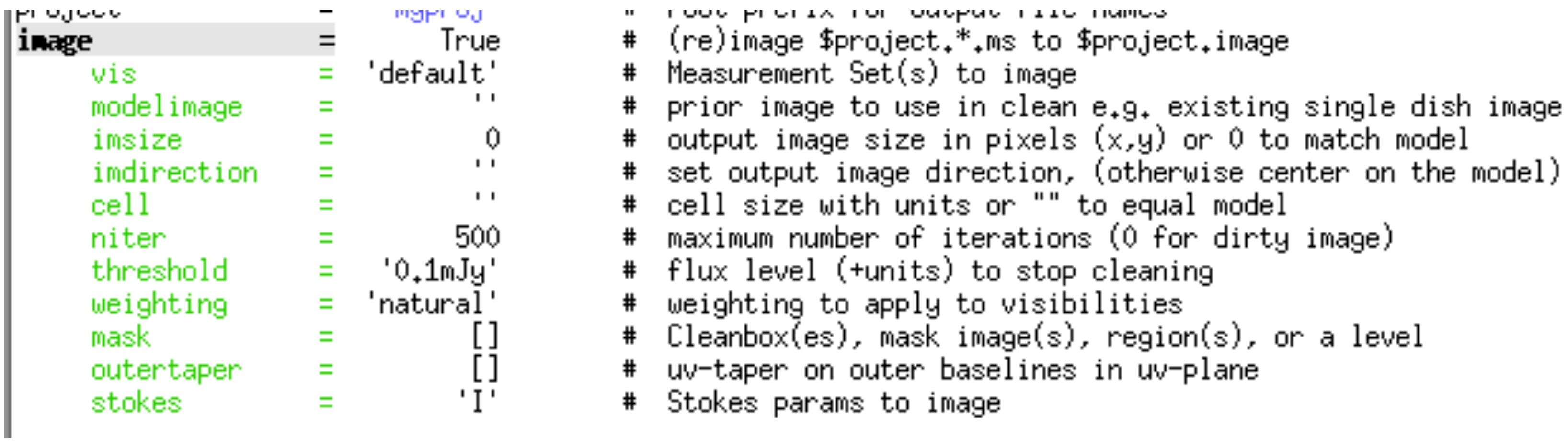
- Image and analyze `simobserve` output



```
CASA <8>: inp simanalyze
--------> inp(simanalyze)
#  simanalyze :: image and analyze simulated datasets
project             =      'sim'        #  root prefix for output file names
image               =       True        #  (re)image $project.*.ms to $project.image
     vis            =  'default'        #  Measurement Set(s) to image
     modelimage     =         ''        #  prior image to use in clean e.g. existing single dish image
     imsize         =          0        #  output image size in pixels (x,y) or 0 to match model
     imdirection    =         ''        #  set output image direction, (otherwise center on the model)
     cell           =         ''        #  cell size with units or "" to equal model
     niter          =        500        #  maximum number of iterations (0 for dirty image)
     threshold      =   '0.1mJy'        #  flux level (+units) to stop cleaning
     weighting      =  'natural'        #  weighting to apply to visibilities
     mask           =         []        #  Cleanbox(es), mask image(s), region(s), or a level
     outertaper     =         []        #  uv-taper on outer baselines in uv-plane
     stokes         =        'I'        #  Stokes params to image

analyze             =      False        #  (only first 6 selected outputs will be displayed)
graphics            =     'both'        #  display graphics at each stage to [screen|file|both|none]
verbose             =      False
overwrite           =       True        #  overwrite files starting with $project
async               =      False        #  If true the taskname must be started using simanalyze(...)
```

image

- Grid, invert, and CLEAN the simulated data set.



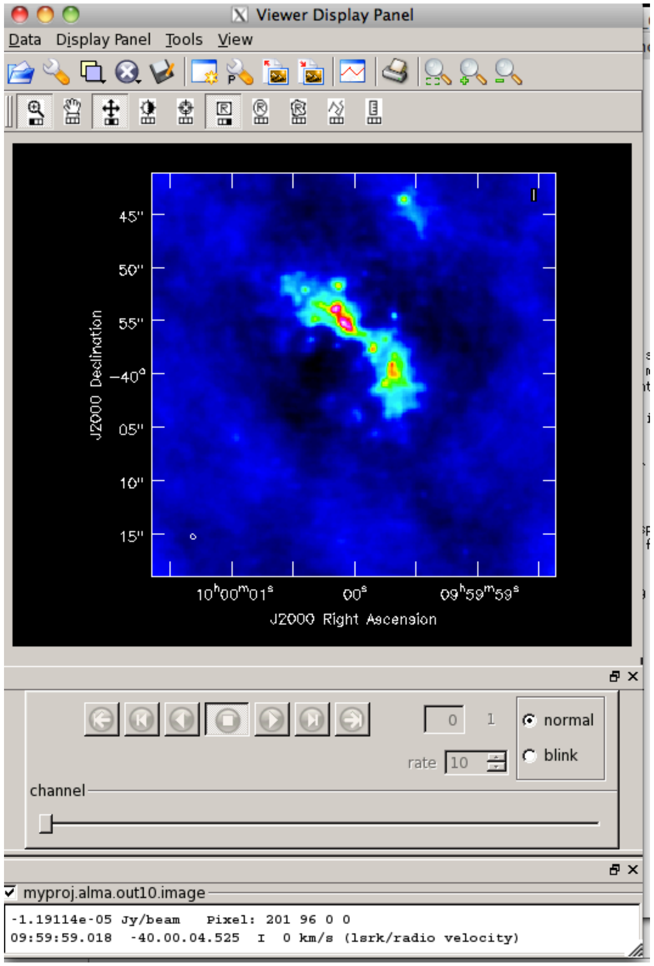
```
project        =   myproj      #  root prefix for output file names
image          =     True      #  (re)image $project.*.ms to $project.image
     vis       =  'default'    #  Measurement Set(s) to image
     modelimage =        ''    #  prior image to use in clean e.g. existing single dish image
     imsize    =         0     #  output image size in pixels (x,y) or 0 to match model
     imdirection =       ''    #  set output image direction, (otherwise center on the model)
     cell      =         ''    #  cell size with units or "" to equal model
     niter     =       500     #  maximum number of iterations (0 for dirty image)
     threshold =   '0.1mJy'    #  flux level (+units) to stop cleaning
     weighting =  'natural'    #  weighting to apply to visibilities
     mask      =        []     #  Cleanbox(es), mask image(s), region(s), or a level
     outertaper =       []     #  uv-taper on outer baselines in uv-plane
     stokes    =       'I'     #  Stokes params to image
```

- Similar but reduced options compared to CLEAN.
  DEFAULTS ARE "SMART", INFORMED BY THE MODEL.

- You can also image the simulated observations with CLEAN.
  THEY ARE A NORMAL CASA MEASUREMENT SET FOR ALL PURPOSES

image

- Output files can be examined with the CASA viewer.
  IN CASA 3.4 THESE LIVE IN `projectname/projectname.image`

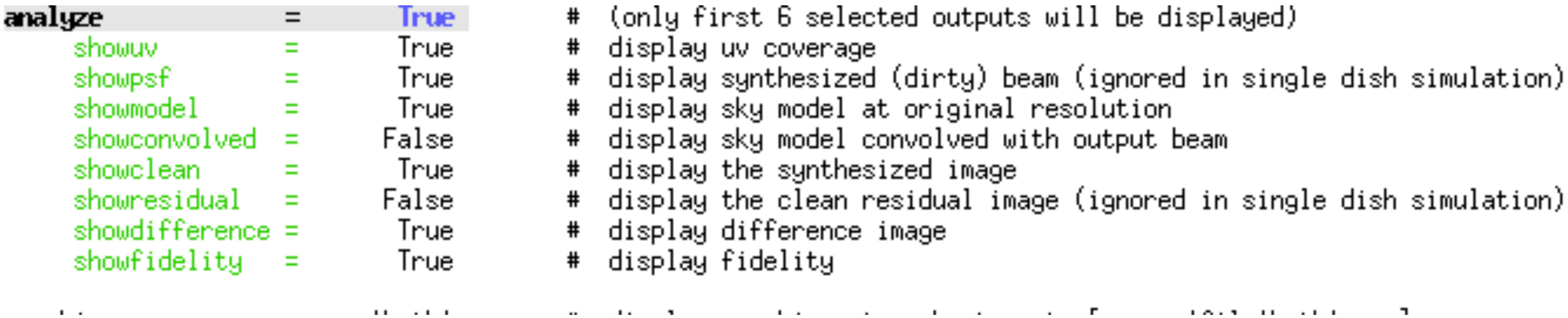


☑ myproj.alma.out10.image

```
-1.19114e-05 Jy/beam   Pixel: 201 96 0 0
09:59:59.018  -40.00.04.525  I  0 km/s (lsrk/radio velocity)
```

analyze

- Create diagnostic plots based on `simobserve` and image



```
analyze            =       True        #  (only first 6 selected outputs will be displayed)
     showuv        =       True        #  display uv coverage
     showpsf       =       True        #  display synthesized (dirty) beam (ignored in single dish simulation)
     showmodel     =       True        #  display sky model at original resolution
     showconvolved =      False        #  display sky model convolved with output beam
     showclean     =       True        #  display the synthesized image
     showresidual  =      False        #  display the clean residual image (ignored in single dish simulation)
     showdifference =      True        #  display difference image
     showfidelity  =       True        #  display fidelity
```

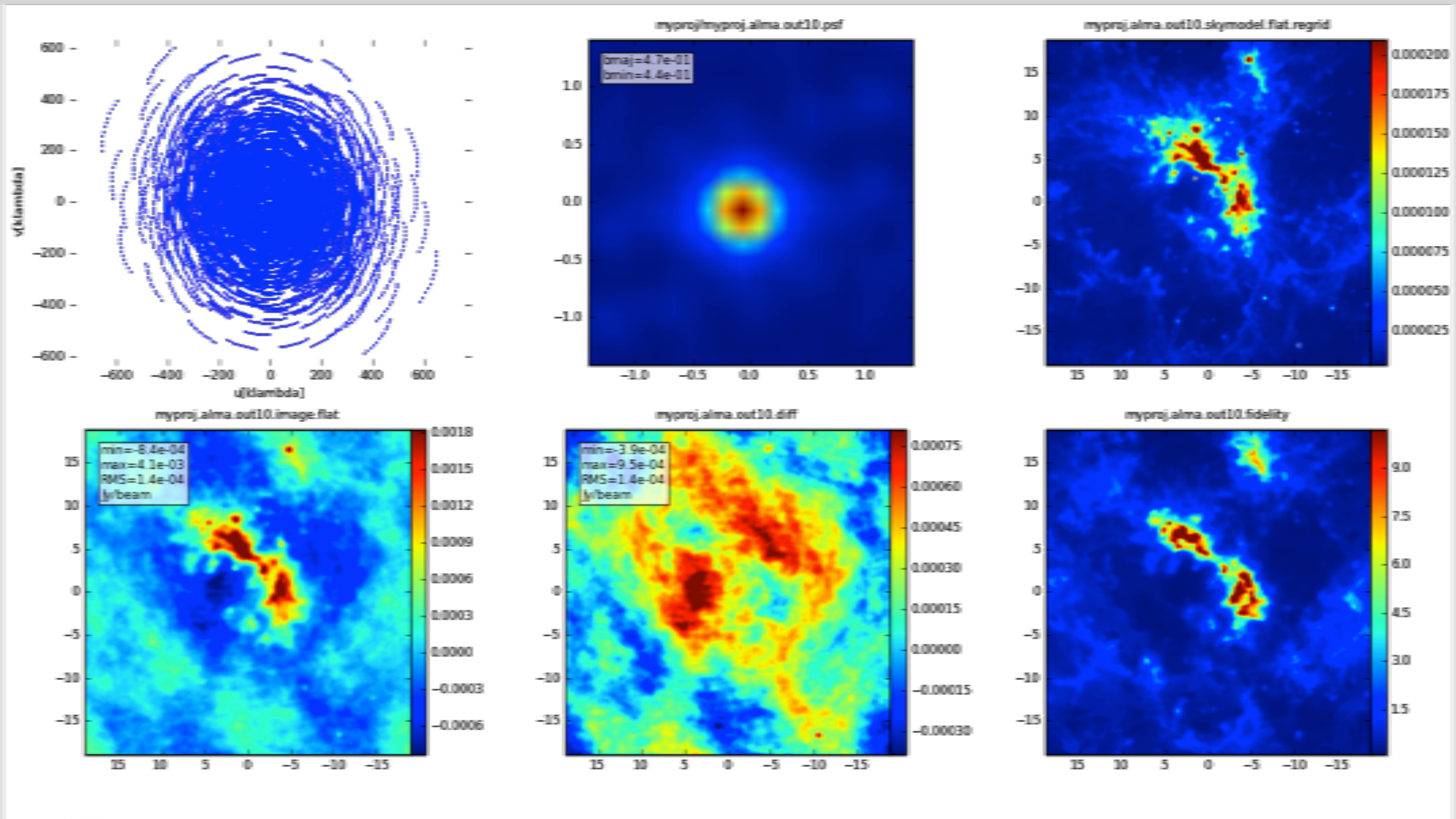
- Pick up to 6 of these.

`go simanalyze`

SIMANALYZE CREATES IMAGES AND DIAGNOSTIC PLOTS IN `projectname/`

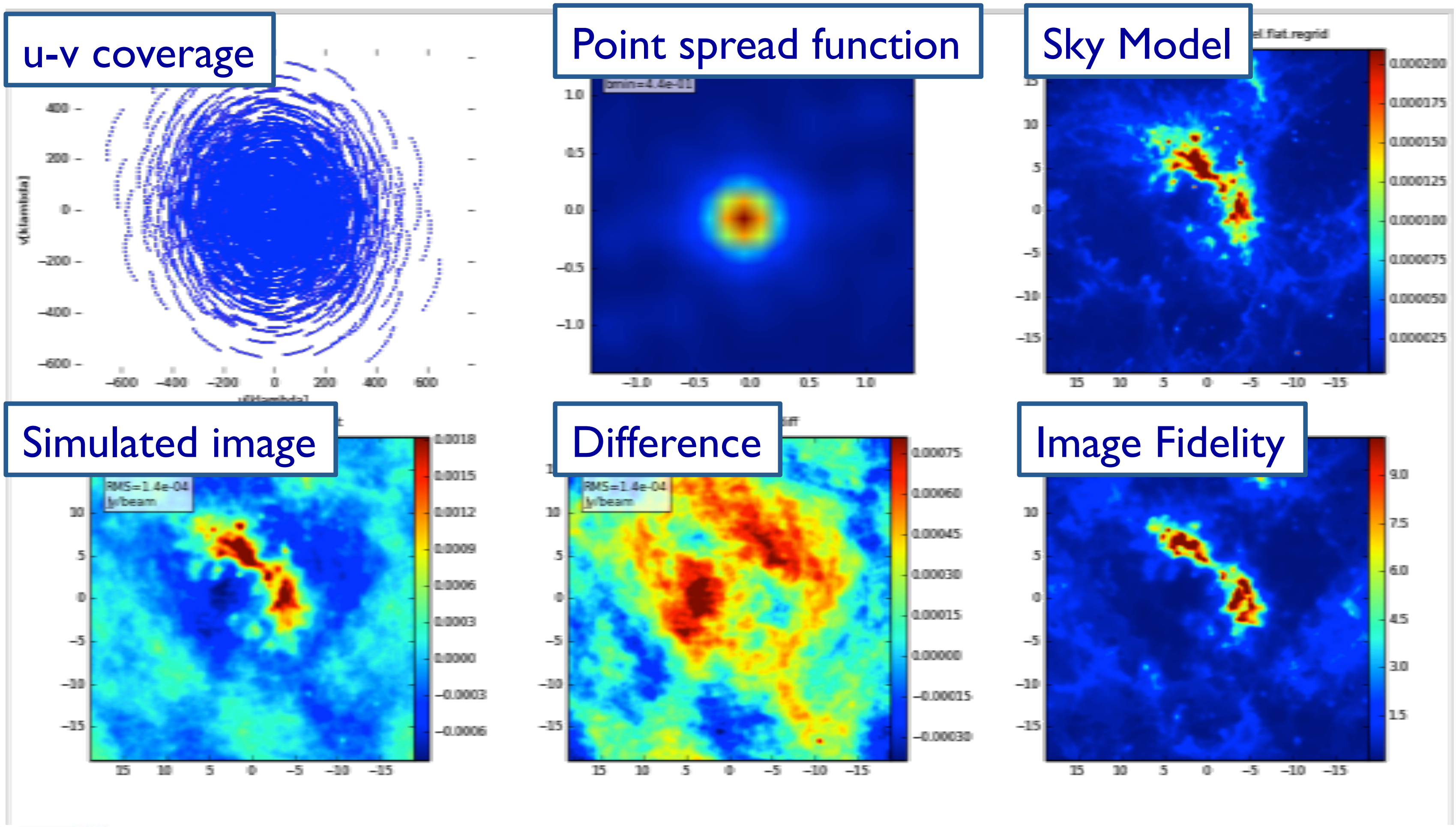
analyze

- Create diagnostic plots based on `simobserve` and image



analyze

- Create diagnostic plots based on `simobserve` and image



u-v coverage

Point spread function

Sky Model

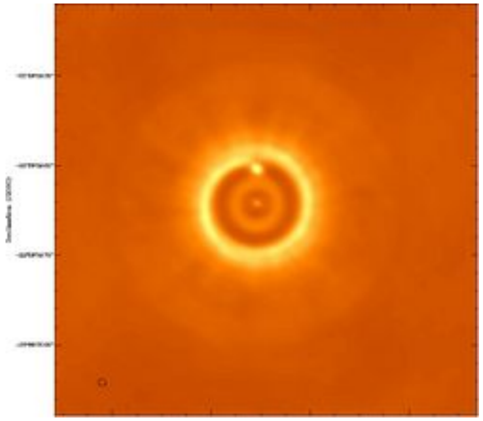
Simulated image

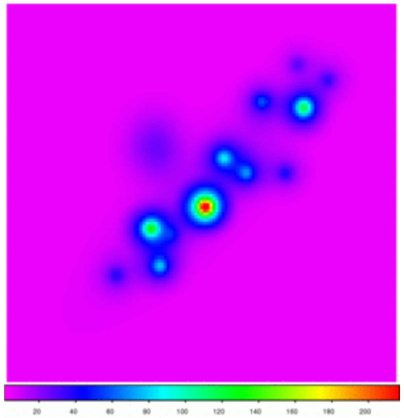
Difference

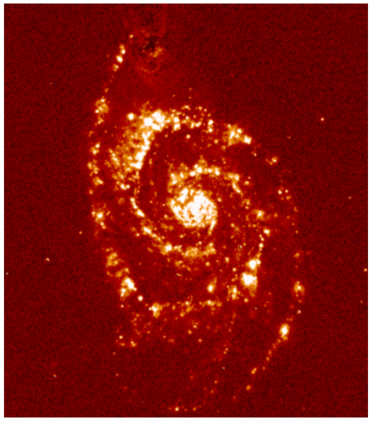
Image Fidelity

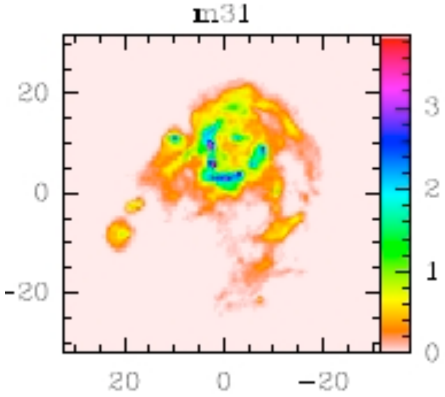
Try It Yourself!

- Simulate one of the suite of model images at http://casaguides.nrao.edu/index.php?title=Sim_Inputs








m31
20
0
-20
20
0
−20
3
2
1
0