# Introduction to CASA

Anita Richards
UK ALMA Regional Centre
JBCA, University of Manchester

With thanks to Danielle Fenech, Dirk Petry, James Miller-Jones and the rest of the JBCA, RadioNet, ESO and NRAO teams

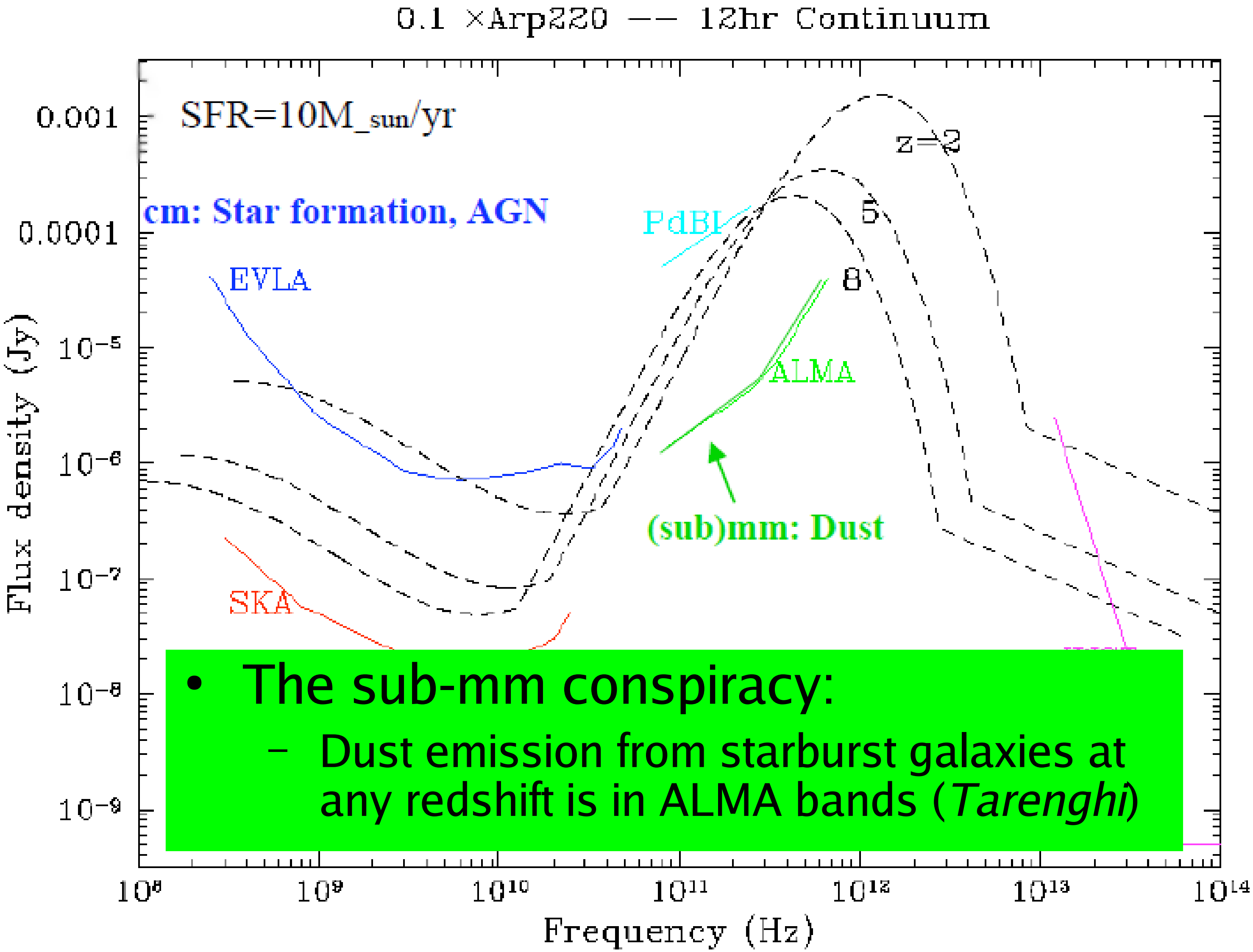
# Next-generation interferometry

- Extended Very Large Array (USA)
  - Optical fibre links, new receivers & correlator etc.
  - 28 antennas, <1 to 36-km baselines
    - 0.05 - 45 arcsec resolution
  - 1 to 50 GHz continuous frequency coverage
  - Up to 8 GHz simultaneous bandwith
  - Full polarization
  - Up to ~4 million spectral channels
  - Continuum sensitivity <1 μJy/hr in central ν range
- Radar, pulsar, solar modes
- Completion 2012
  - Incremental upgrade

# ALMA

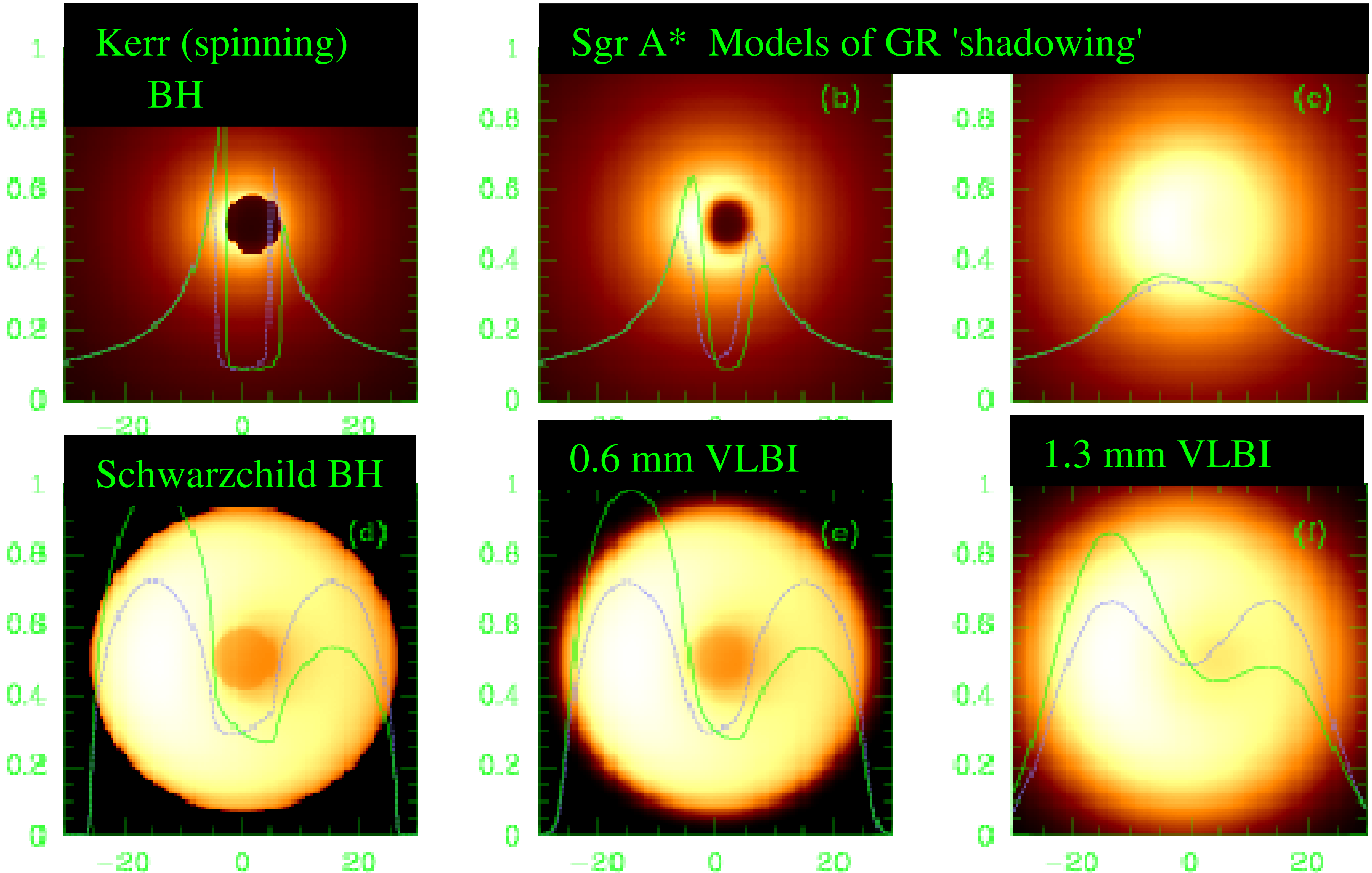
- Atacama Large Millimetre Array
  - Europe (ESO), North America, East Asia, Chile
    - Built on Chajnantur Plateau at 5000 metres
  - 54x12-m, 12x7-m antennas
  - 30 - 950 GHz in 10 atmospheric windows
    - Thousands of molecular lines, dust
    - Collimation regions of continuum jets
    - High-redshift CO, C+, dust
  - Baselines 15 m to 14 km
    - 0.005 - few arcsec resolution
  - Continuum sensitivity ~1 mJy per second
- Sub-arcmin field of view - Mosaicing
- Completion 2012

# ALMA Science from z=8 to 8 kpc




- The sub-mm conspiracy:
  - Dust emission from starburst galaxies at any redshift is in ALMA bands (*Tarenghi*)

# ALMA Science from z=8 to 8 kpc

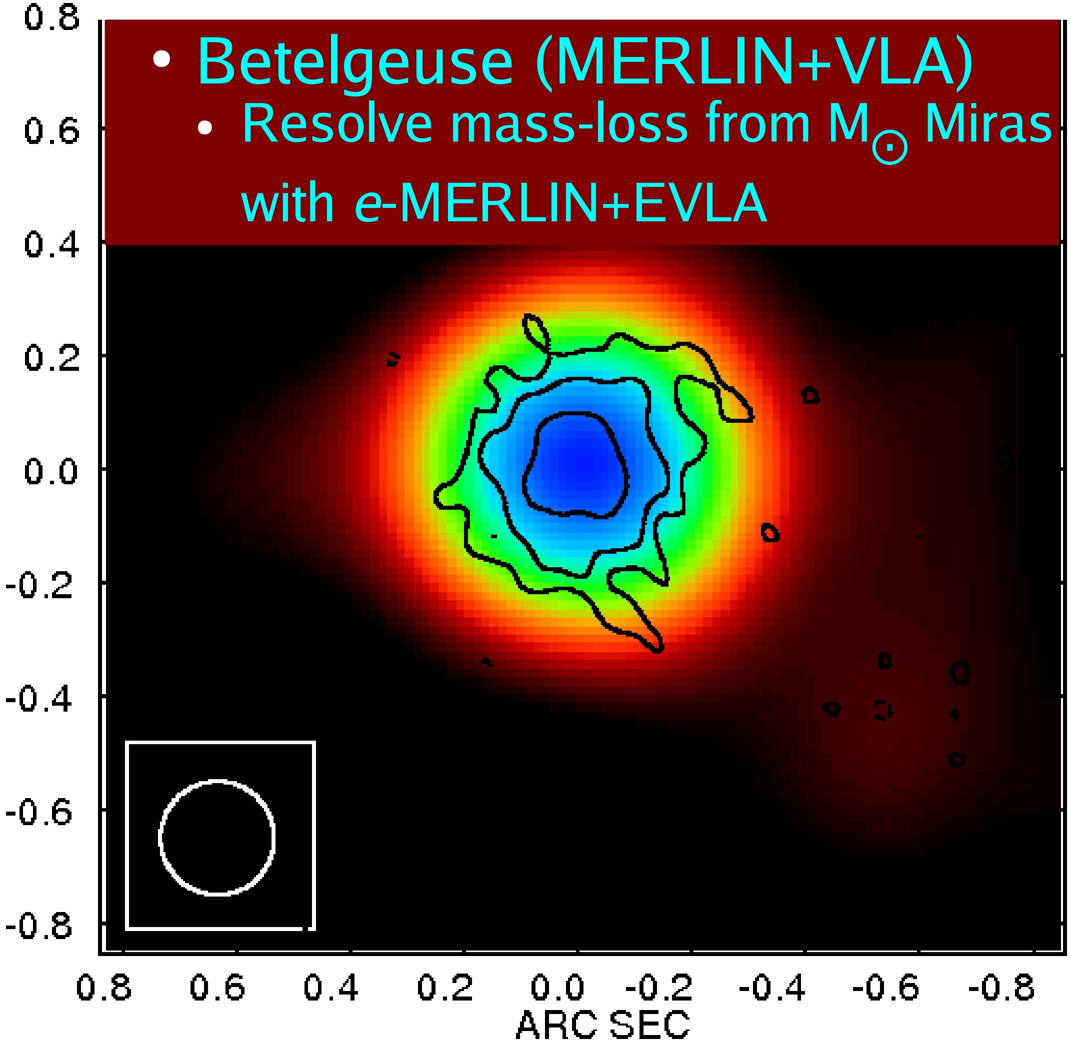


# *e*-MERLIN

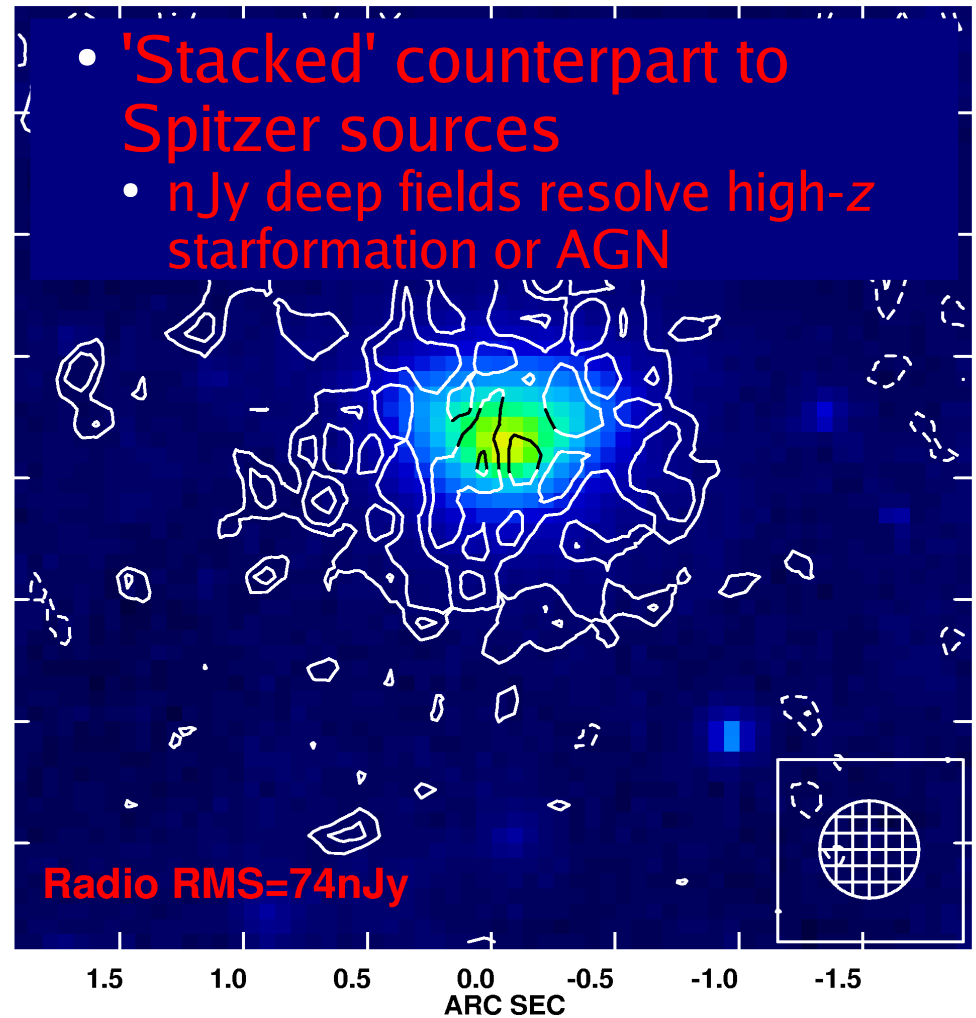
- '*electronic*'-MERLIN (UK)
  - Linking MERLIN telescopes with optical fibres
    - Up to 2 GHz bandwidth
    - Similar correlator as EVLA - up to ~$10^5$ channels
  - 5x25-m antennas, 1x32-m, 75-m Lovell
  - 1.3-1.7, 4-8, 21-26 GHz bands
  - Baselines up to 217 km
    - 0.01 - 0.2 arcsec resolution
  - Continuum sensitivity few μJy per 12 hr
- 5 GHz: 8-arcmin FoV = >$10^8$ pixels
  - All sensitive imaging multi-channel ∵ confusion
  - Integration times 1 sec or less
- Completion 2010

# *e*-MERLIN science

- High resolution and sensitivity



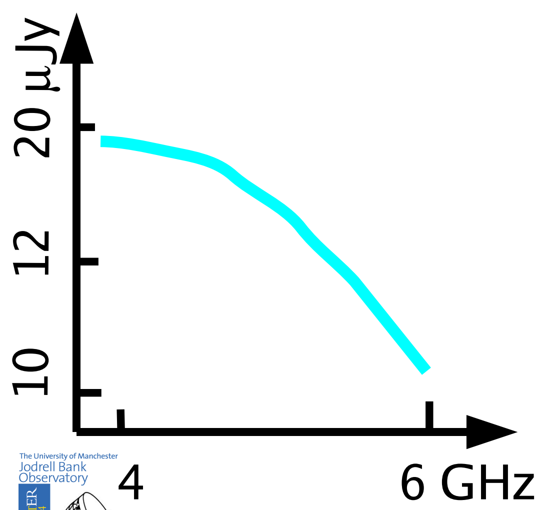

- Betelgeuse (MERLIN+VLA)
  - Resolve mass-loss from $M_\odot$ Miras with *e*-MERLIN+EVLA

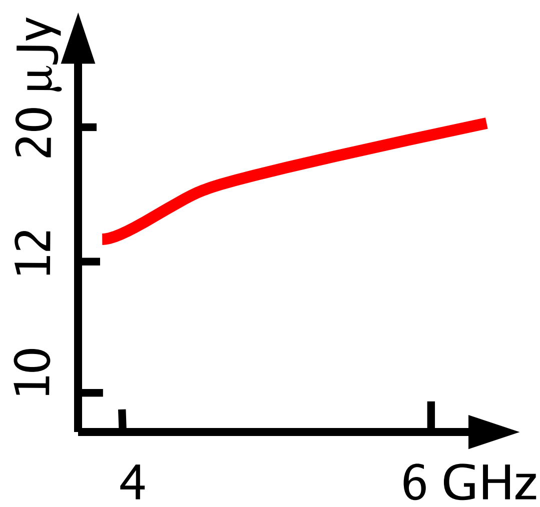


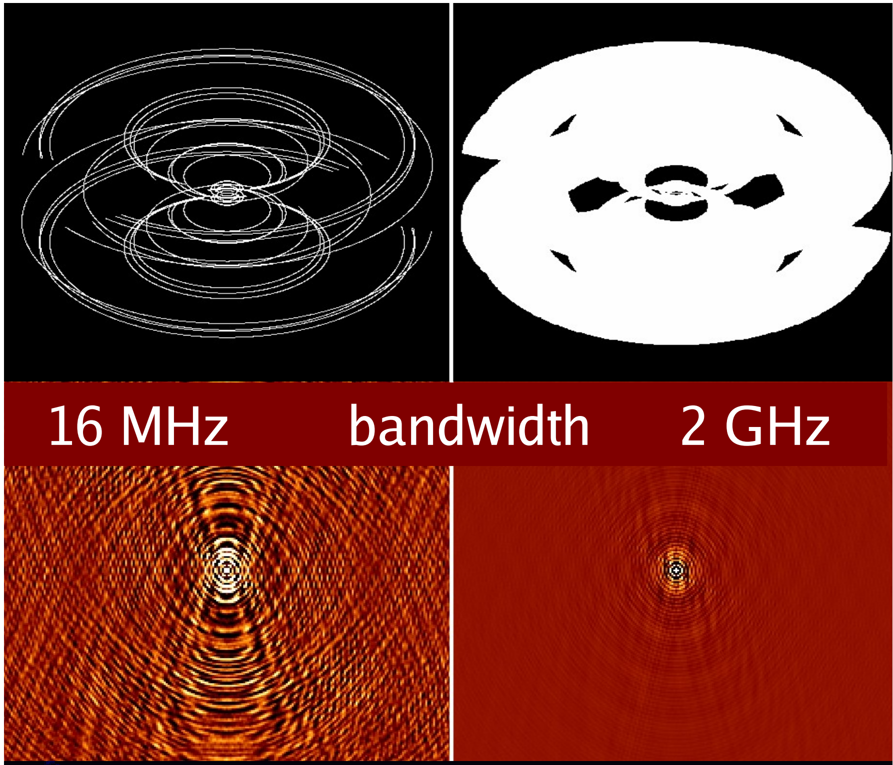

- 'Stacked' counterpart to Spitzer sources
  - nJy deep fields resolve high-*z* starformation or AGN

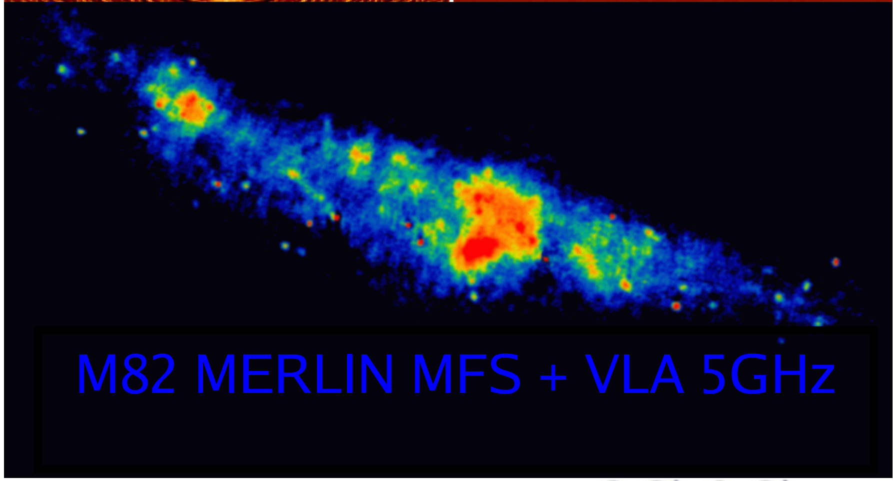
# Wide field, wide band imaging

- ≲50% fractional bandwidth
  - μJy sensitivity
  - Better *uv* plane filling
    - Improved fidelity
- Tens to hundreds of sources per FoV (5'-40' at 8-1 GHz)
  - Map even unwanted confusion
  - Different spectral indicies

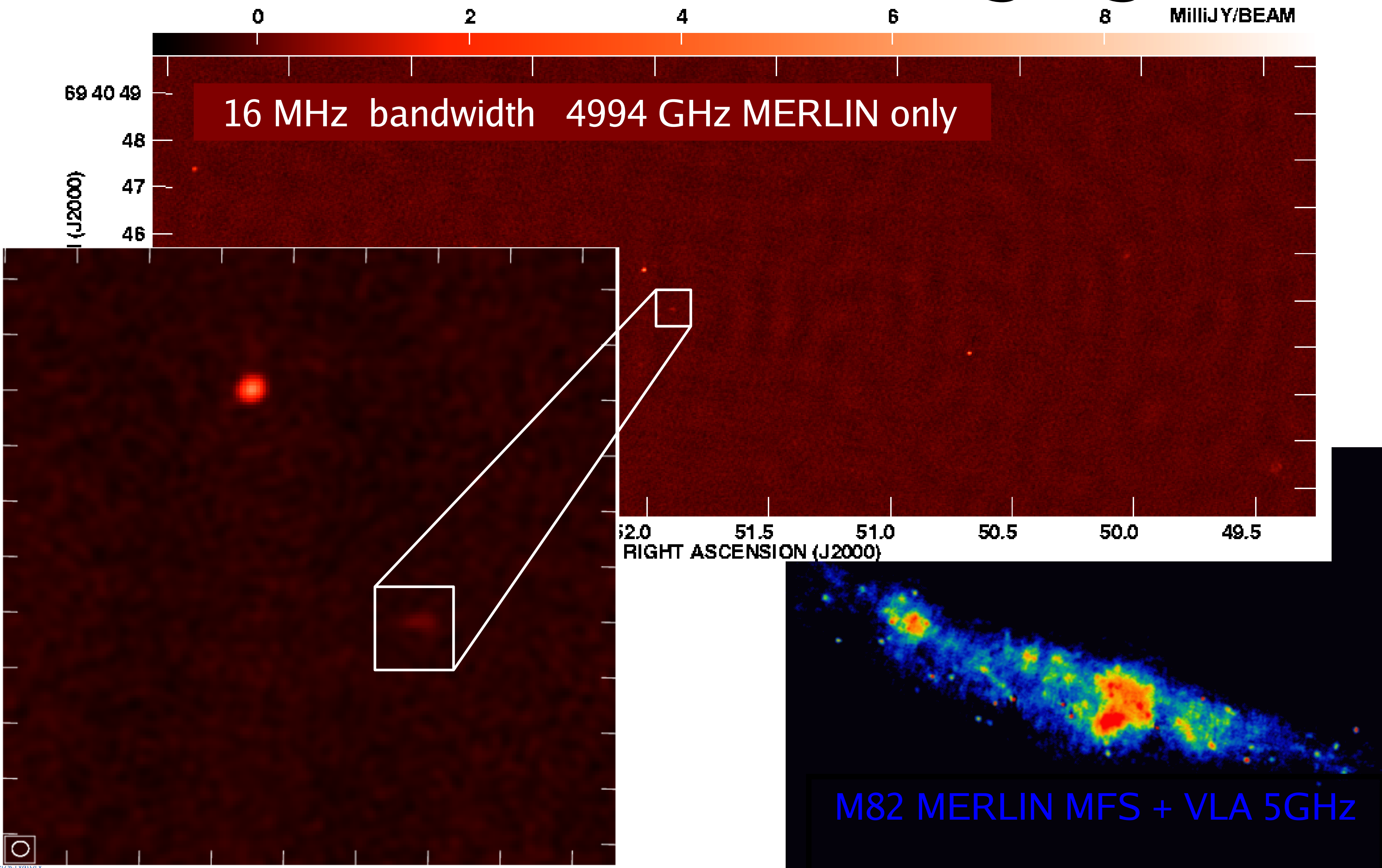


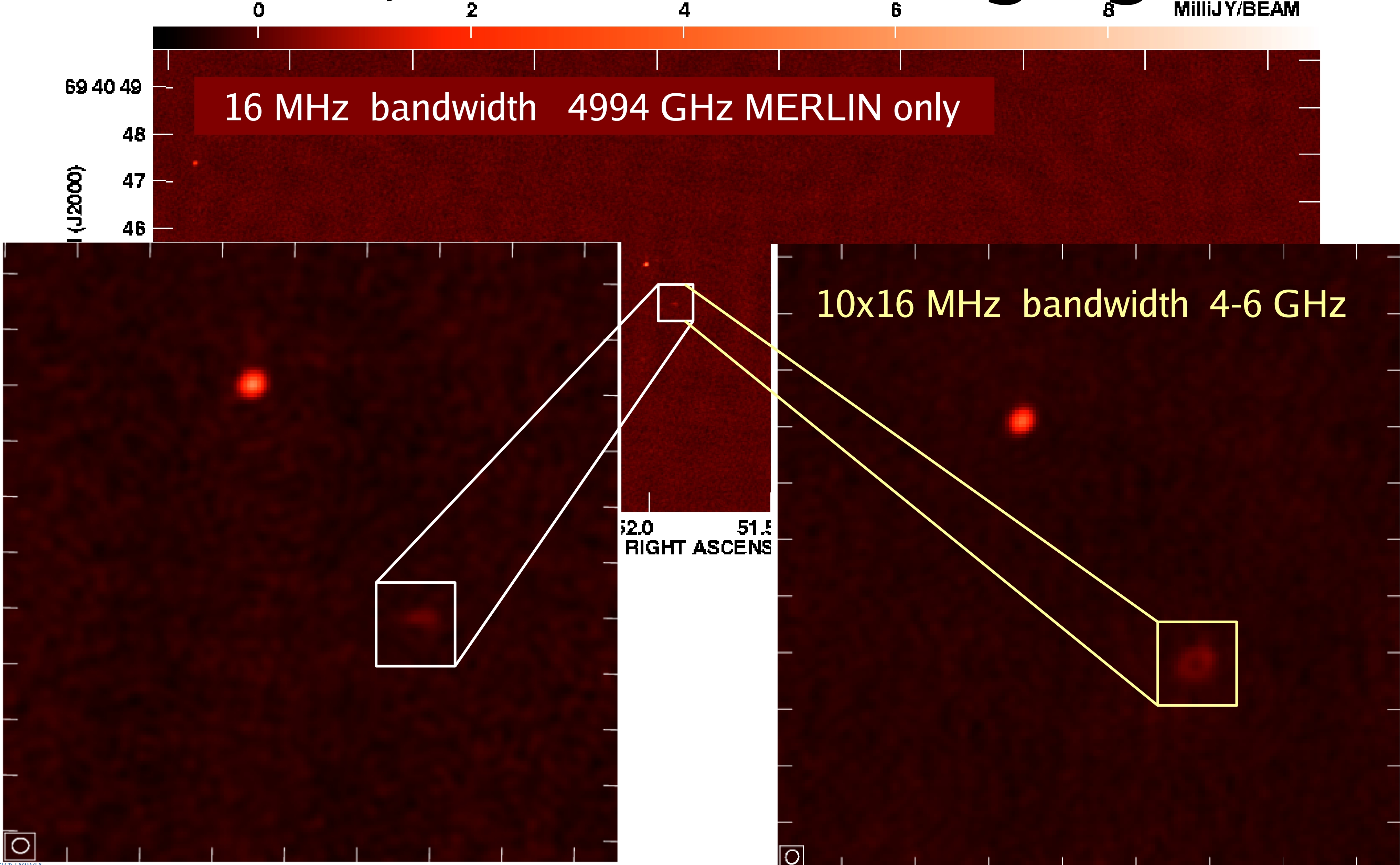



# Wide field, wide band imaging

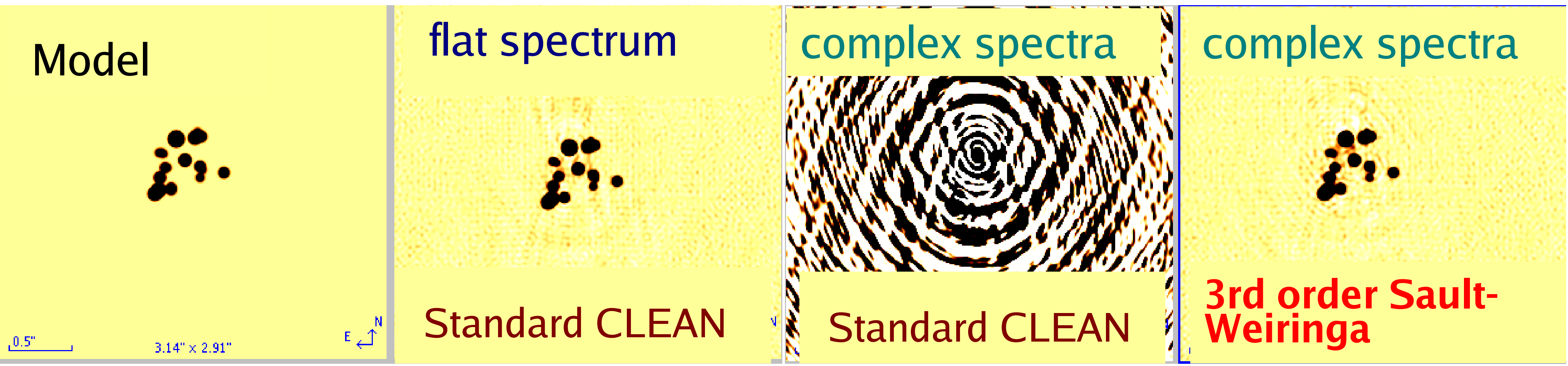


# Wide field, wide band imaging

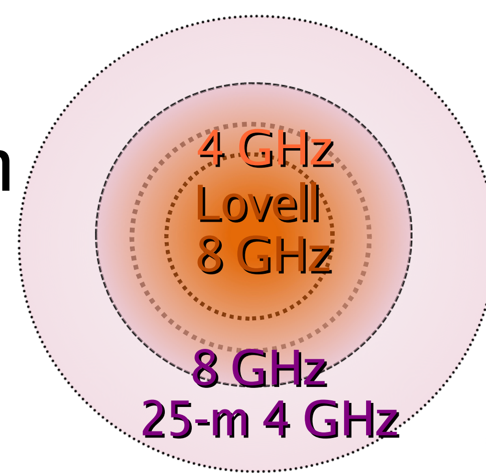


# Wide band, wide field imaging

- Multi-channel data for wide-field imaging
  - Solve for spectral index $\alpha$, maybe curvature
    - Prototypes by *Rau* (CASA), *Stewart & Fenech* (Parseltongue)

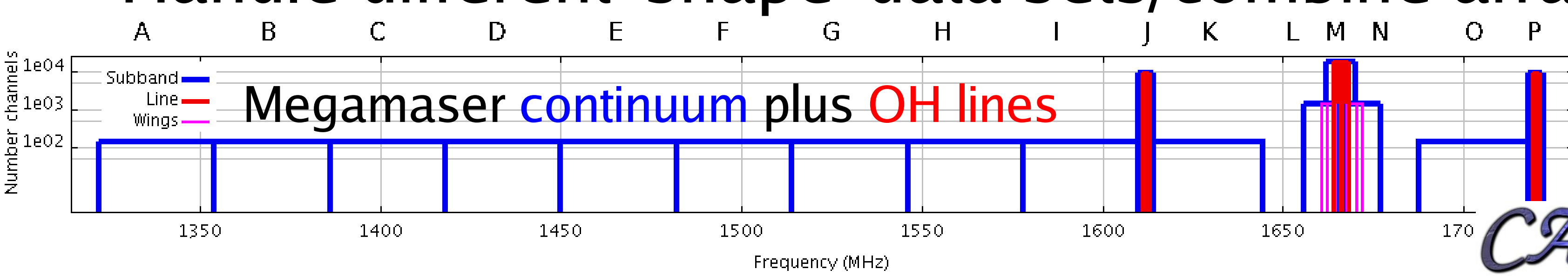



- Primary beam $\propto 1/\nu$
  - MERLIN, VLBI dishes 25-100 m
  - ALMA 7 & 12 m
    - Starting to implement in CASA



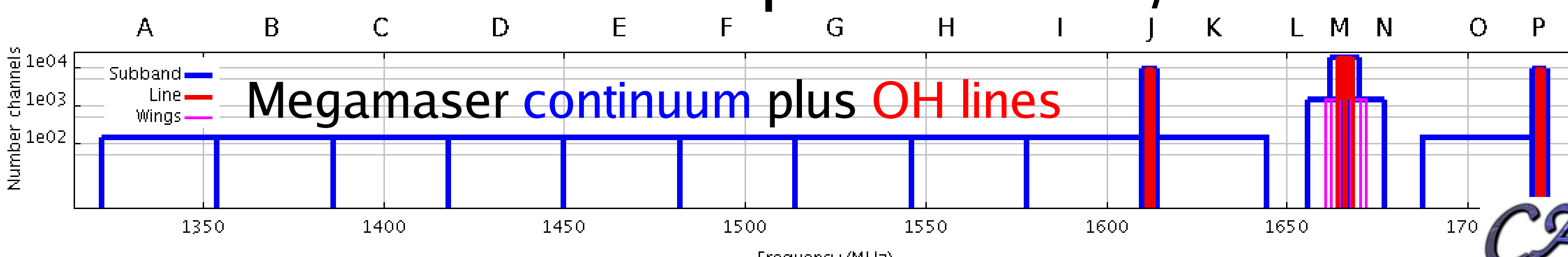
# Data *reduction* challenges

- TeraBytes of data per day
  - Data processing must be parallelizable
- Wide fields - new calibration techniques
  - Changing ionosphere, antenna deformation etc.
- High frequencies - very unstable atmosphere
  - Rapid source-switching, fit polynomials to phases
- Wide bands - new imaging techniques
  - Solve for continuum spectral index/curvature
  - Mosaicing many pointings, sky curvature
- Flexible spectral configurations
  - Handle different 'shape' data sets/combine arrays

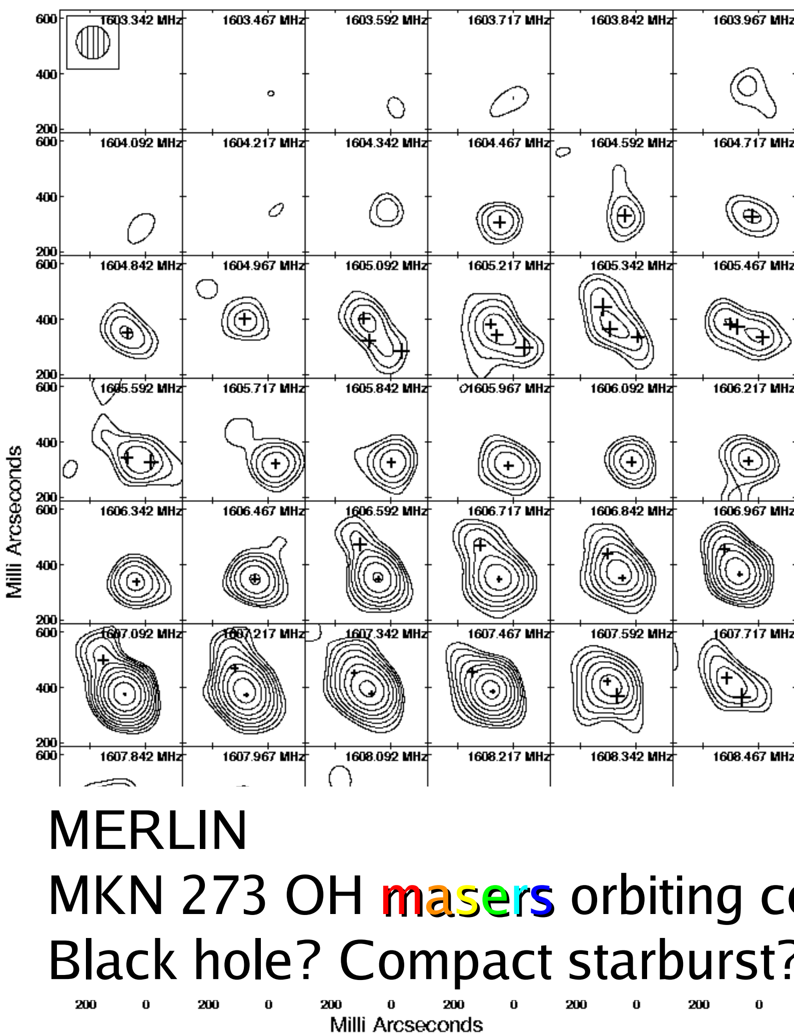


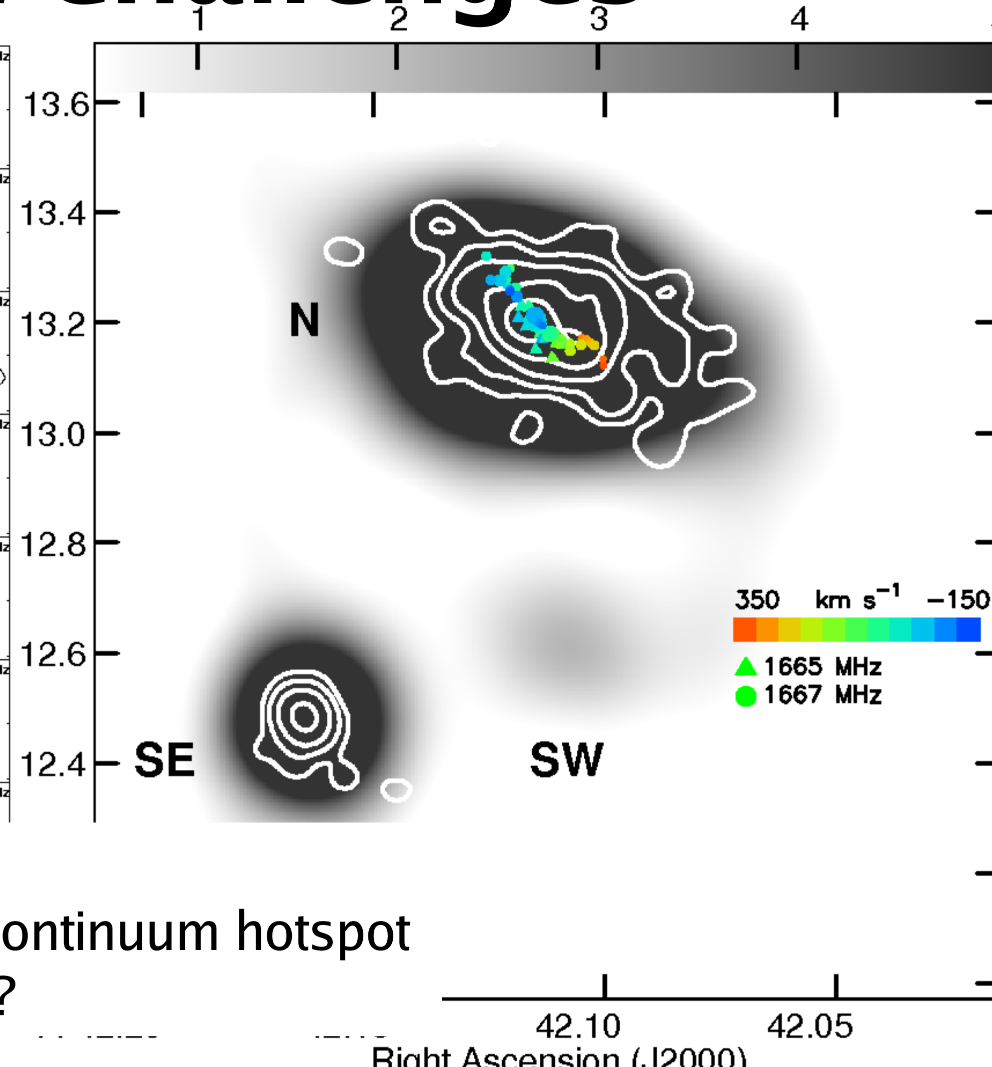
# Data *reduction* challenges

- TeraBytes of data per day
  - Data processing must be parallelizable
- Wide fields - new calibration techniques
  - Changing ionosphere, antenna deformation etc.
- High frequencies - very unstable atmosphere
  - Rapid source-switching, fit polynomials to phases
- Wide bands - new imaging techniques
  - Solve for continuum spectral index/curvature
  - Mosaicing many pointings, sky curvature
- Flexible spectral configurations
  - Handle different 'shape' data sets/combine arrays



# Data *reduction* challenges

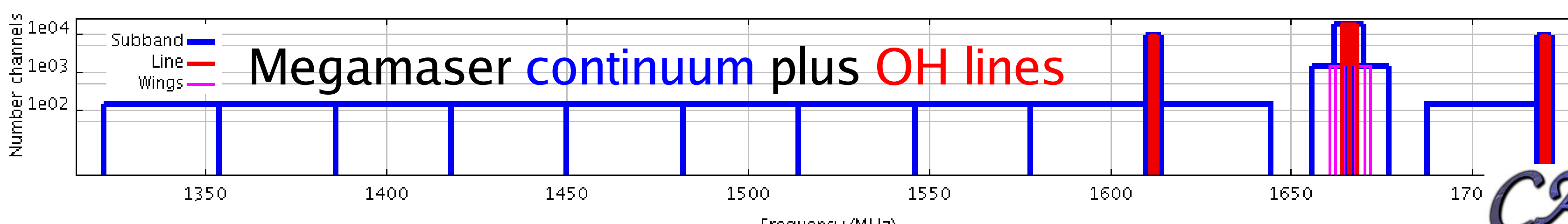



MERLIN

MKN 273 OH masers orbiting continuum hotspot

Black hole? Compact starburst?




Megamaser continuum plus OH lines

# CASA developed to meet NG needs

- Primary drivers: EVLA and ALMA needs
- Currently >20 people working on CASA:
  - 14 NRAO+Virginia, 3 NAOJ, 2 ESO, 1 Calgary,
  - Fractions at Paris, ASTRON, ATNF ....
- Beta releases publicly available
  - Good for VLA, ATCA, CARMA, some *e*-MERLIN
    - Don't blindly copy Cookbook parameter values!
- Complements classic AIPS
  - CASA cannot yet handle all VLBI-like processing
  - ALBiUS (RadioNet) developing interoperability
- Python scripting provides 'task' interface
  - Underlying aips++ (c++) toolkit also available

# Libraries use Measurement Equation

$$\underline{V}_{ij} = \mathbf{M}_{ij}\mathbf{B}_{ij}\mathbf{G}_{ij}\mathbf{D}_{ij}\int \mathbf{E}_{ij}\mathbf{P}_{ij}\mathbf{T}_{ij}\mathbf{F}_{ij} S \underline{I}_{\nu}(l,m) e^{-i2\pi(u_{ij}l+v_{ij}m)} dl dm + \underline{A}_{ij}$$

## *Vectors*

$\underline{V}$ isibility = *f(u,v)*

$\underline{I}$ mage to be calculated

$\underline{A}$ dditive baseline error

## Scalars

$S$ (mapping $\underline{I}$ to observer pol.)

$l,m$ image plane coords

$u,v$ Fourier plane coords

$i,j$ telescope pair

## Jones Matrices

**M**ultiplicative baseline error

**B**andpass response

**G**eneralised electronic gain

**D**term (pol. leakage)

**E** (antenna voltage pattern)

**P**arallactic angle

**T**ropospheric effects

**F**araday rotation

# Using the Measurement Equation

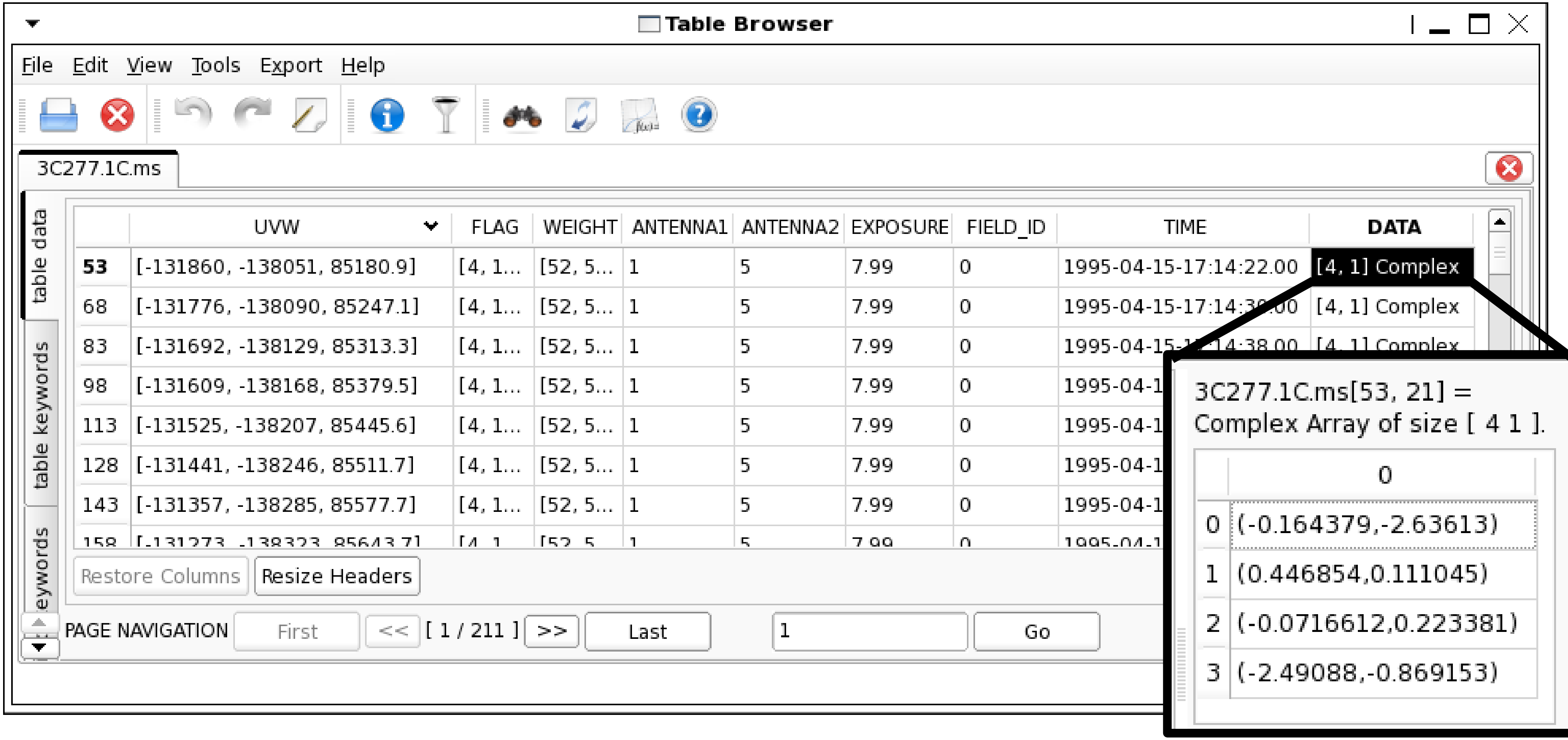
- *Hamaker, Bregman & Sault 1996*
  - Decompose into individual calibration components e.g.
- $\underline{V}_{ij}^{obs} = \mathbf{B}_{ij}\mathbf{G}_{ij}\mathbf{D}_{ij}\mathbf{P}_{ij}\mathbf{T}_{ij}\mathbf{F}_{ij}\underline{V}_{ij}^{ideal}$
  - Linearise and solve by $\chi^2$ minimization
- Same principles as any gain calibration
- Other terms added as required
  - e.g. $\zeta$ Jones matrix (© Jan Nordham)
- Visibility data are stored in Measurement Sets
  - Accessible directories of tables

# Measurement Set visibility data

- Directory of Tables
- MAIN table
  - One row per integration per baseline per spectral window
    - Cells hold complex visibilities and weights

```
jupiterallcal.split.ms
|-- ANTENNA
|    |-- table.dat
|    |-- table.f0
|    |-- table.info
|    `-- table.lock
|-- DATA_DESCRIPTIO
|    |-- table.dat
|    |-- table.f0
|    |-- table.info
|    `-- table.lock
|-- FEED
|    |-- table.dat
|    |-- table.f0
|    |-- table.f0i
|    |-- table.info
|    `-- table.lock
|-- FIELD
|    |-- table.dat
|    |-- table.f0
|    |-- table.f0i
|    |-- table.info
|    `-- table.lock
|-- FLAG_CMD
|    |-- table.dat
|    |-- table.f0
|    |-- table.info
|    `-- table.lock
|-- HISTORY
|    |-- table.dat
|    |-- table.f0
|    |-- table.info
|    `-- table.lock
|-- OBSERVATION
|    |-- table.dat
|    |-- table.f0
|    |-- table.info
|    `-- table.lock
|-- POINTING
|    |-- table.dat
|    |-- table.f0
|    |-- table.f0i
|    |-- table.f1
|    |-- table.info
|    `-- table.lock
|-- POLARIZATION
|    |-- table.dat
|    |-- table.f0
|    |-- table.f0i
|    |-- table.info
|    `-- table.lock
|-- PROCESSOR
|    |-- table.dat
|    |-- table.f0
|    |-- table.info
|    `-- table.lock
|-- SOURCE
|    |-- table.dat
|    |-- table.f0
|    |-- table.f0i
|    |-- table.info
|    `-- table.lock
|-- SPECTRAL_WINDOW
|    |-- table.dat
|    |-- table.f0
|    |-- table.f0i
|    |-- table.info
|    `-- table.lock
|-- STATE
|    |-- table.dat
|    |-- table.f0
|    |-- table.info
|    `-- table.lock
|-- table.dat
|-- table.f0
|-- table.f1
|-- table.f2
|-- table.f2_TSM1
|-- table.f3
|-- table.f3_TSM1
|-- table.f4
|-- table.f5
|-- table.f6
|-- table.f6_TSM0
|-- table.f7
|-- table.f7_TSM1
|-- table.f8
|-- table.f8_TSM1
|-- table.info
`-- table.lock
```

# Measurement Set MAIN table




- Some of the columns per visibility
  - **Data**: Complex value for each of 4 correlations (LL RR LR RL) per spectral channel

# What's in the Measurement Set?

| MAIN | Model, e.g.: | Corrected data | Flags |
|---|---|---|---|
| **Original visibility data** | *FT of image made from MS*<br><br>*FT of supplied model image*<br><br>*FT of calibrator flux density* | *Copy of visibilities with calibration tables applied*<br><br>(Used in imaging but not calibration) | (Edits are stored here first; backup tables can be made and used to modify) |

- Additional tables:
  - *Admin*: Antenna, Source etc.
  - *Processing*: calibration, flags, etc.

# Starting CASA

- See web links for downloads (or http://casa.nrao.edu)
  - Don't forget the Cookbook!
- Start by typing **casapy**
  - This starts the iPython environment
    - Interactive input to tasks in the xterm
    - Logger (see toolbar for display, export options)
  - Access to shell
    - Direct simple commands e.g. ls
    - Prefix any unix command with ! e.g. !more file
- Python
  - Take care with indentation
  - Case sensitive
  - Zero indexed (e.g. 27 antennas numbered 0~26)
    - **Run any scripts or functions you want**

# Using CASA

- Use **inp taskname** to view inputs
  - Greyed parameters are expandable

```
xterm

CASA <37>: inp gaincal
---------> inp(gaincal)
#  gaincal :: Determine temporal gains from calibrator observations
vis                 = '3C277.1C.ms'     #  Name of input visibility file
caltable            =         ''        #  Name of output gain
                                        #   calibration table
field               =         ''        #  Select field using field
                                        #   id(s) or field name(s)
spw                 =         ''        #  Select spectral
                                        #   window/channels
selectdata          =      False        #  Other data selection
                                        #   parameters
solint              =      'inf'        #  Solution interval: egs.
                                        #   'inf', '60s' (see help)
```

# Using CASA

xterm

```
CASA <38>: selectdata = True

CASA <39>: inp gaincal
----------> inp(gaincal)
#  gaincal :: Determine temporal gains from calibrator observations
vis                 = '3C277.1C.ms'     #  Name of input visibility file
caltable            =         ''        #  Name of output gain
                                        #   calibration table
field               =         ''        #  Select field using field
                                        #   id(s) or field name(s)
spw                 =         ''        #  Select spectral
                                        #   window/channels
selectdata          =       True        #  Other data selection
                                        #   parameters
     timerange      =         ''        #  Select data based on time
                                        #   range
     uvrange        =         ''        #  Select data within uvrange
                                        #   (default units meters)
     antenna        =         ''        #  Select data based on
                                        #   antenna/baseline
     scan           =         ''        #  Scan number range
     msselect       =         ''        #  Optional complex data
                                        #   selection (ignore for now)

solint              =      'inf'        #  Solution interval: egs.
                                        #   'inf', '60s' (see help)
```

# Using CASA

- Simplest input to tasks is `param=value`
  - In this mode, variables are global
    - `solint='1min'` will appear in all tasks until reset
  - `default(gaincal)` resets default values
  - `tget gaincal` restores last *successful* execution
  - `saveinputs(gaincal,'gctry1')` saves inputs at any stage
  - `execfile('gctry1')` restores
    - gctry1 is a text file, view using e.g. `!more gctry1`
- `Help('gaincal')` for more details
  - Use the Cookbook for fuller examples

# Running tasks

- In interactive mode
  - Just type e.g. `gaincal`
  - Tasks are normally run sequentially per session
  - See the logger for progress
- Assign measurements to variables
  - e.g. `noise_target = imstat()`
    - Python syntax examples in scripts or cookbook
    - `rms_target=noise_target['rms'][0]`
- Beware re-assigning/mistyping task params
  - `molint = '1sin'` won't give an error
  - `calmode = 'delay'` does show up in red

# Data handling

- CASA converts visibility data to MS
  - Recognised formats (see Cookbook for loading) :
    - VLA Export format
    - UVFITS (not calibration, flagging, leakage tables)
      - So apply these first if importing e.g. AIPS UVFITS
    - Science Data Models like AlmaSDM
  - Data can be exported as UVFITS
    - Apply calibration etc. first
    - SPLIT out different spectral configurations
- CASA images are also directories
  - Import/export from FITS
- NB won't overwrite file of the same name

# CASA demos 1: VLA data

- NGC5921 - HI spectral line observations
  - FITS data and script included in CASA tarball
  - See School web site or search directory
    - ngc5921_demo.py, NGC5921.fits
  - Script contains flowchart, helpful comments
    - NB It starts by deleting all previous data (unless you #)
- Jupiter - polarized radio continuum
  - Script in tarball, see web page for data link
  - Additional practice:
    - Flagging
    - Calibration of polarization leakage and pol. angle
    - Imaging/visualising linear polarization
- Use by cut and paste/type yourself

# CASA demos 2: MERLIN & ATCA

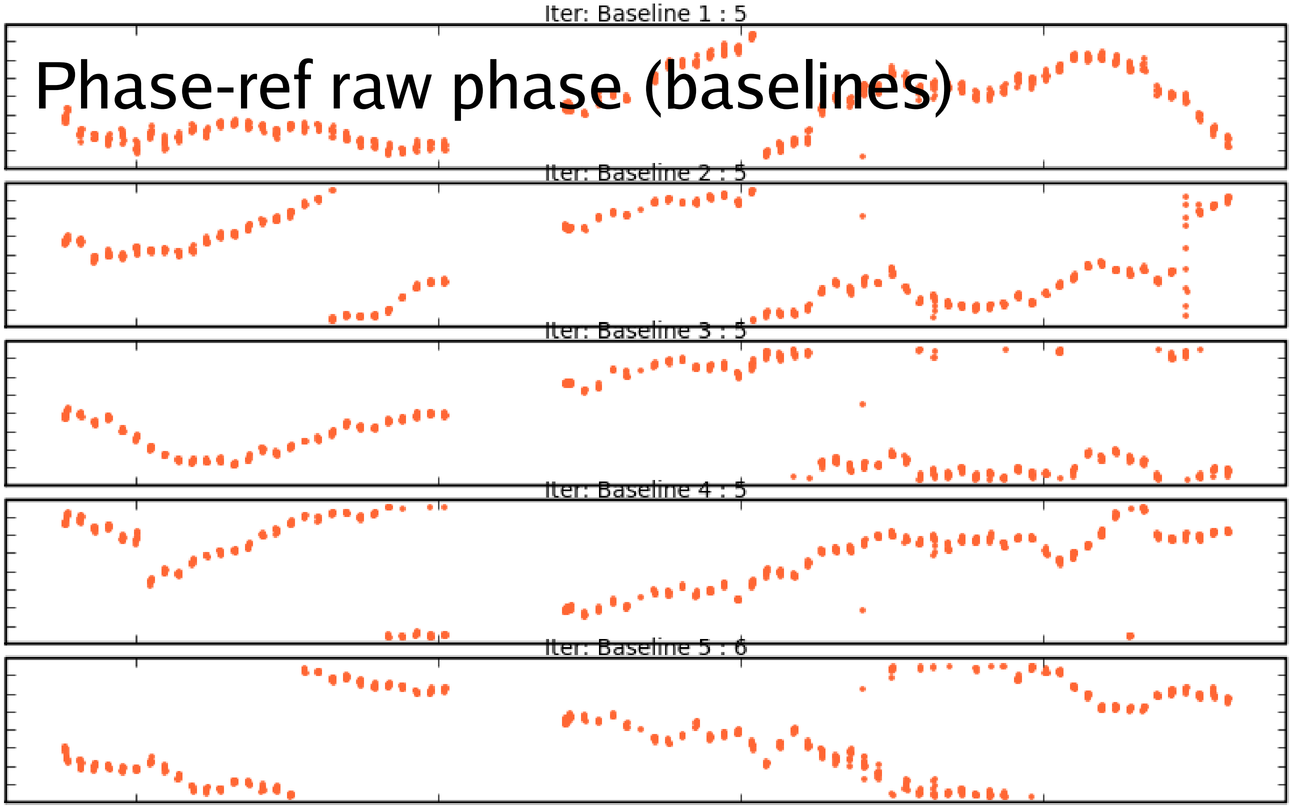
- If you have time - see web site for links
  - FITS data, AIPS & CASA scripts
- MERLIN: 3C277.1 polarized continuum
  - Higher resolution
  - Faster phase rate
    - More information this afternoon
- ATCA: J2342-44 HI spectral line
  - Lower resolution
  - Many sources in continuum field
- Scripts provide examples of strategies for a range of array parameters

# What to look for in calibration

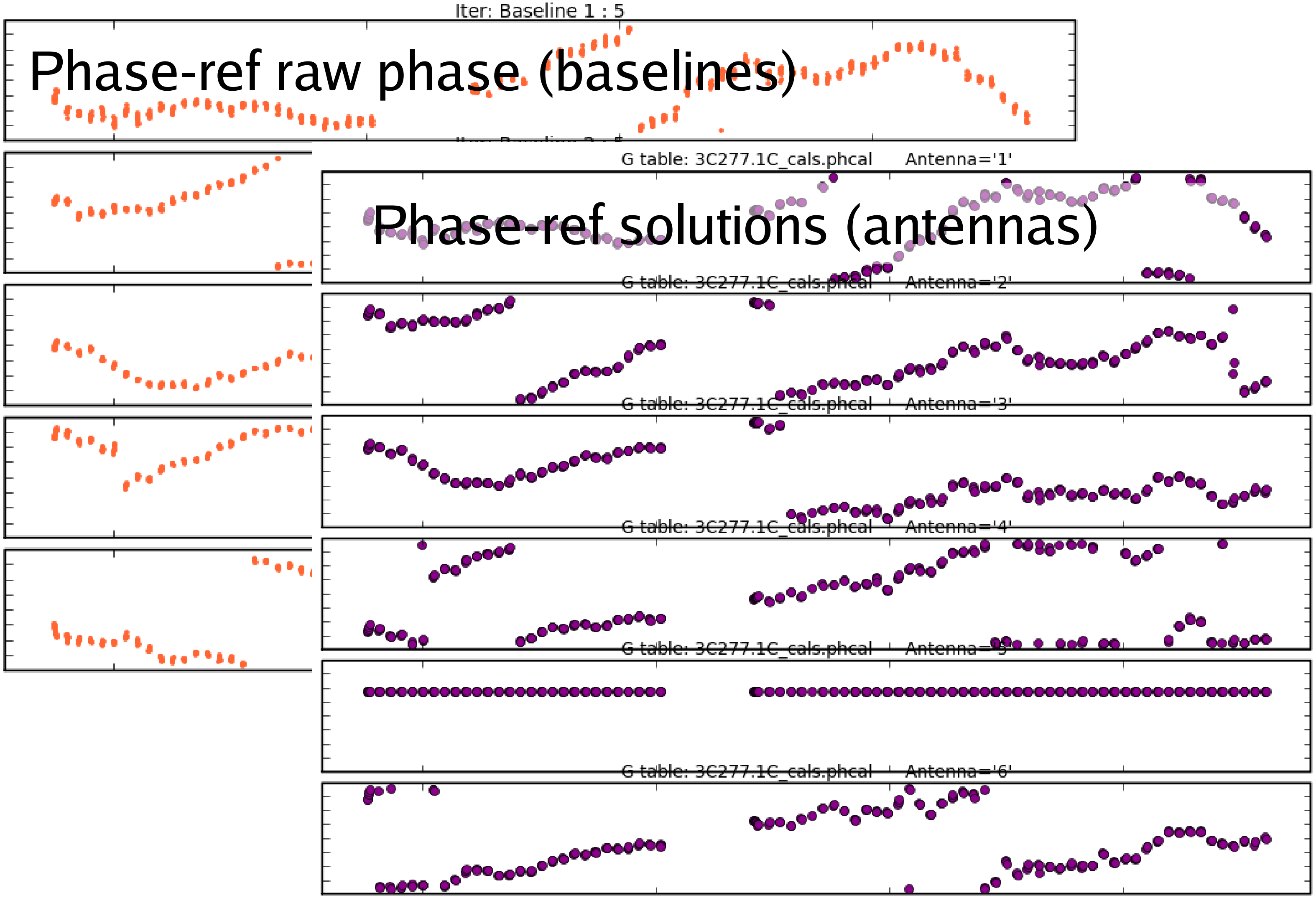
- Tasks like `gaincal` compare the visibility data with a model (FT of image clean components)
  - $\underline{\mathbf{V}}_{true} = \underline{\mathbf{V}}_{observed} \times \Sigma \textit{corruption}$
  - You are trying to find $[\Sigma \textit{corruption}]^{-1}$
    - Usually a least-squares or least-difference fit (Cornelia and Katherine's talks)
- Always check your solutions and think whether they make sense in relation to the data
  - tasks `plotxy, plotcal`

# Phase calibration

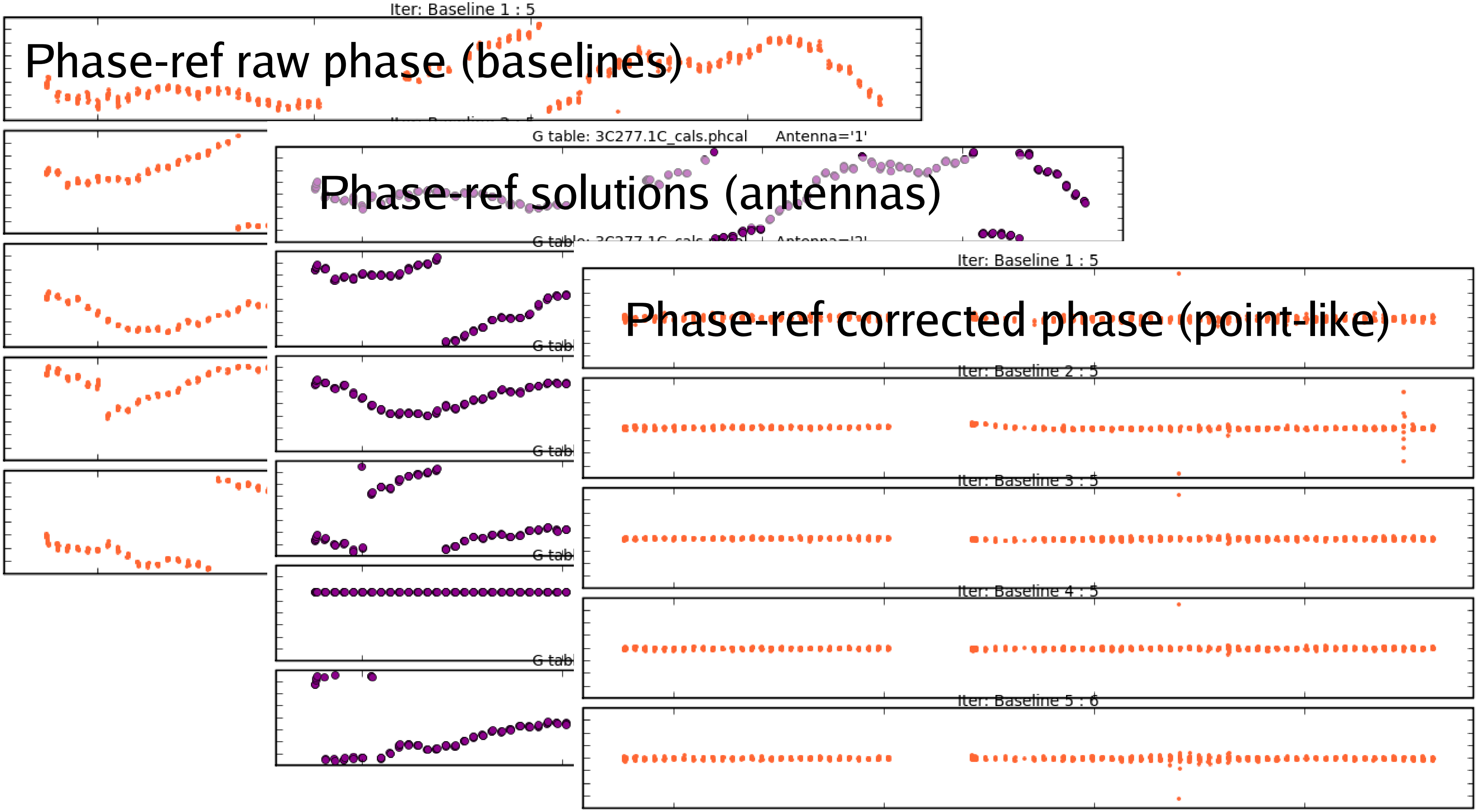
Phase-ref raw phase (baselines)



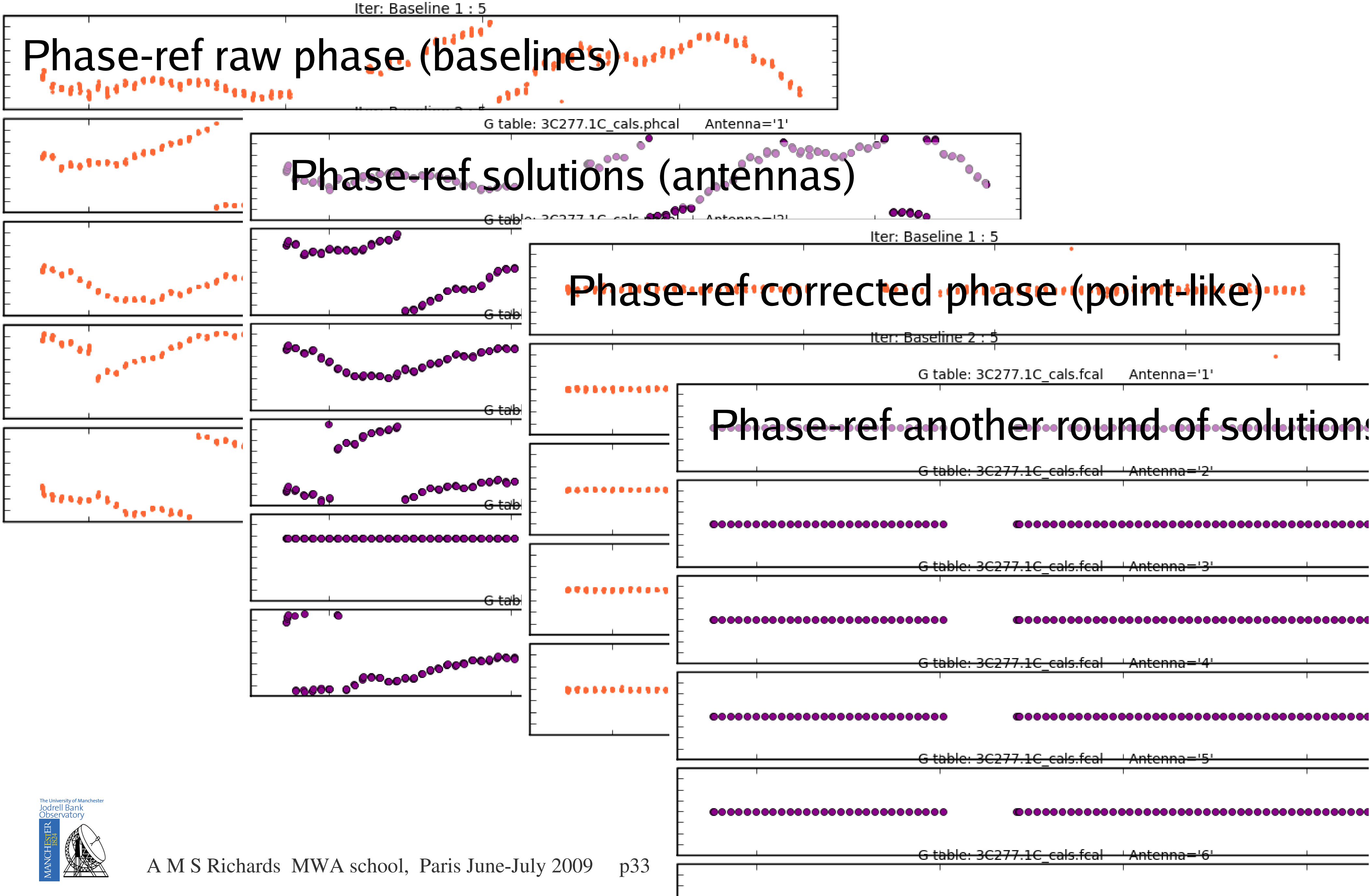
# Phase calibration



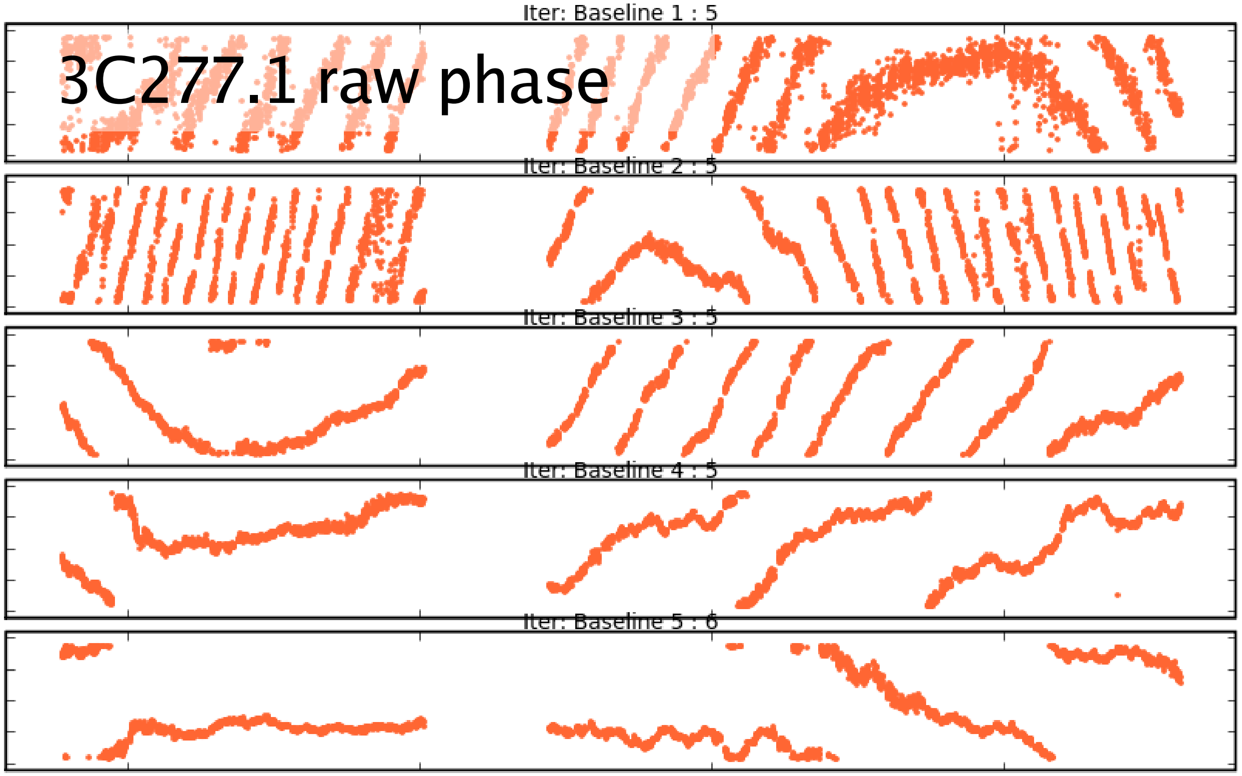
# Phase calibration



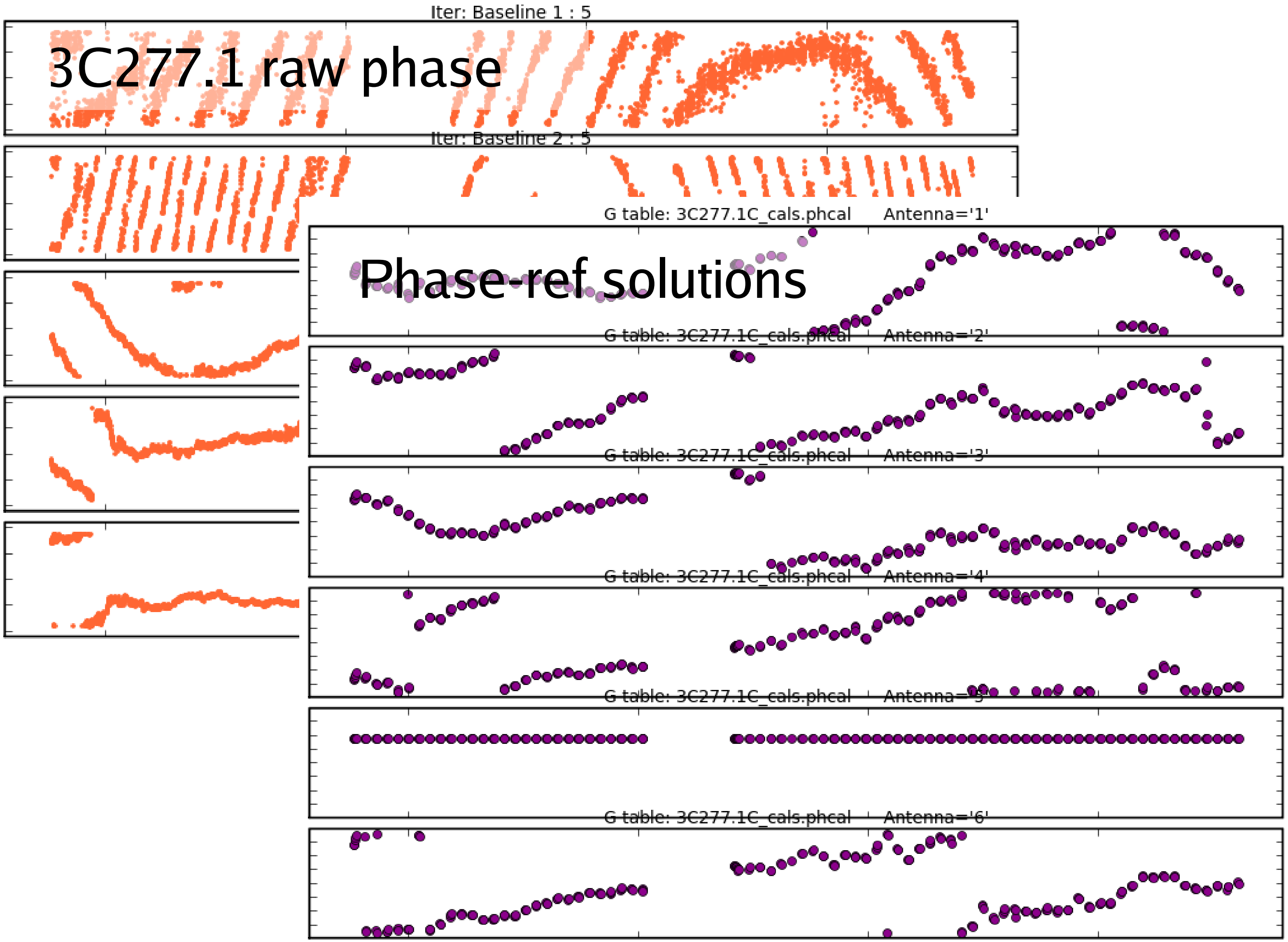
# Phase calibration



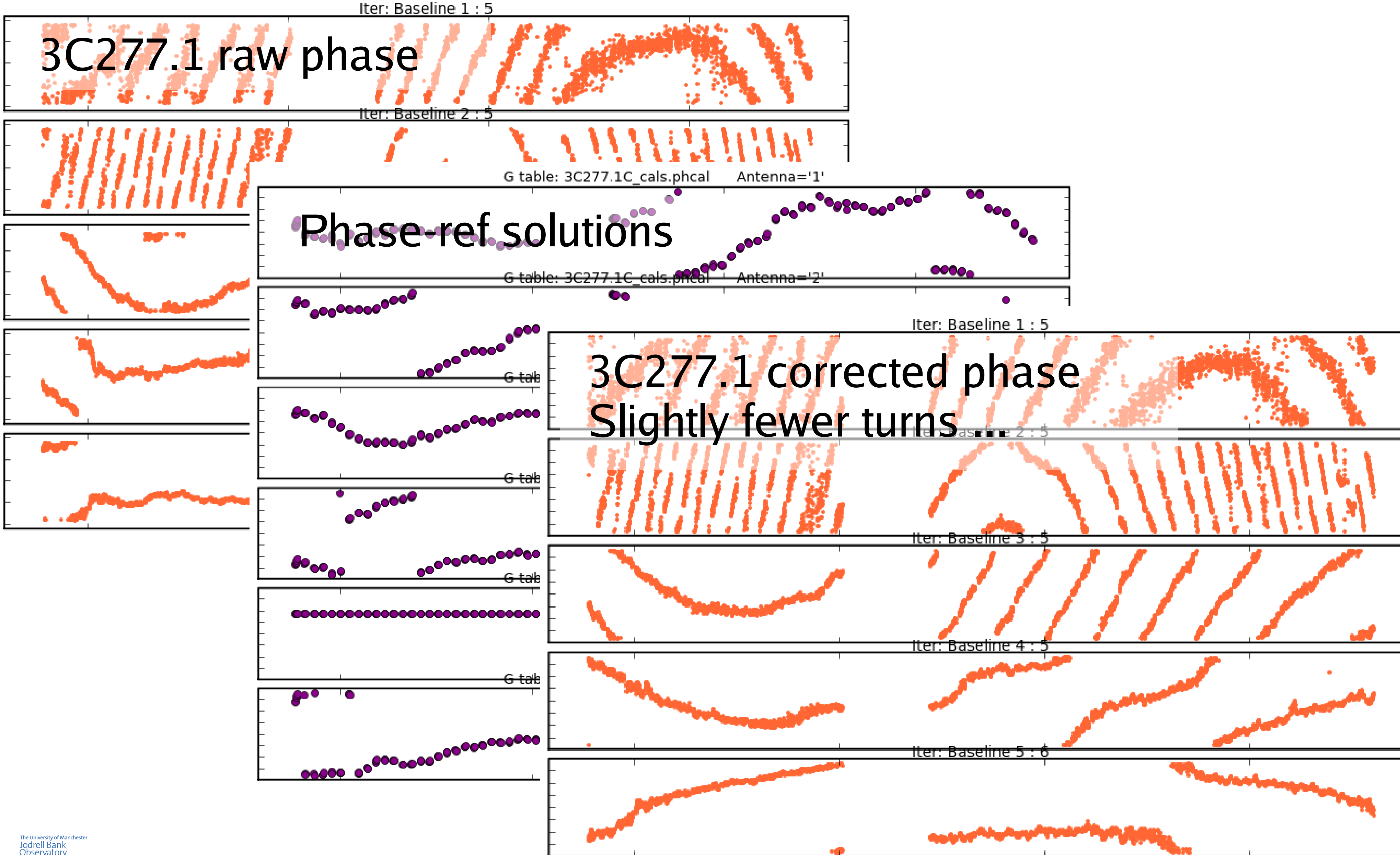
# Interpolated onto target:



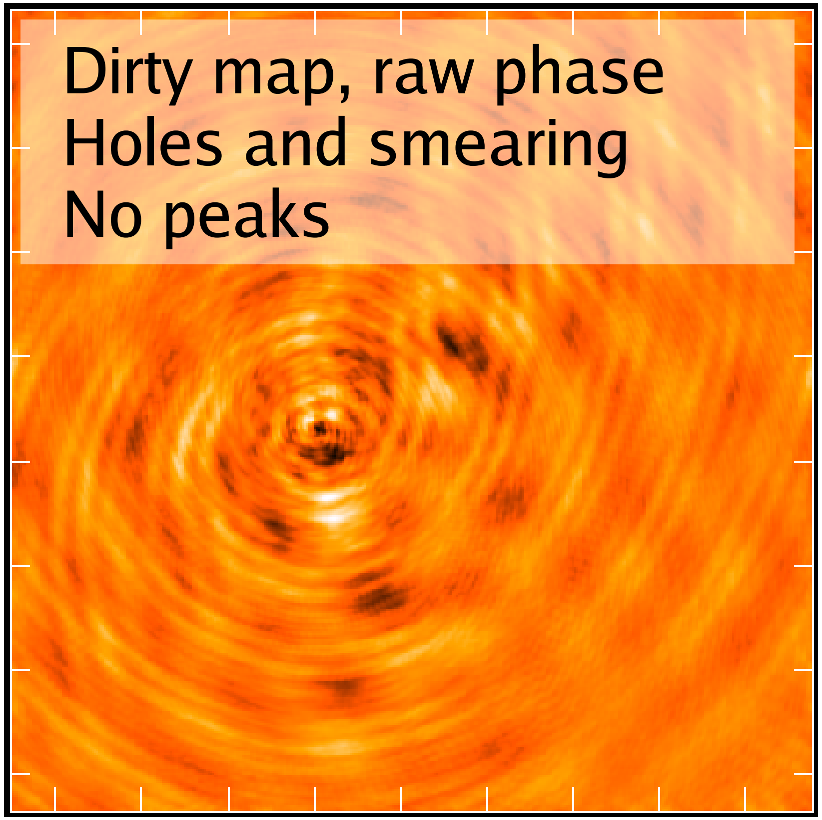
# Interpolated onto target:



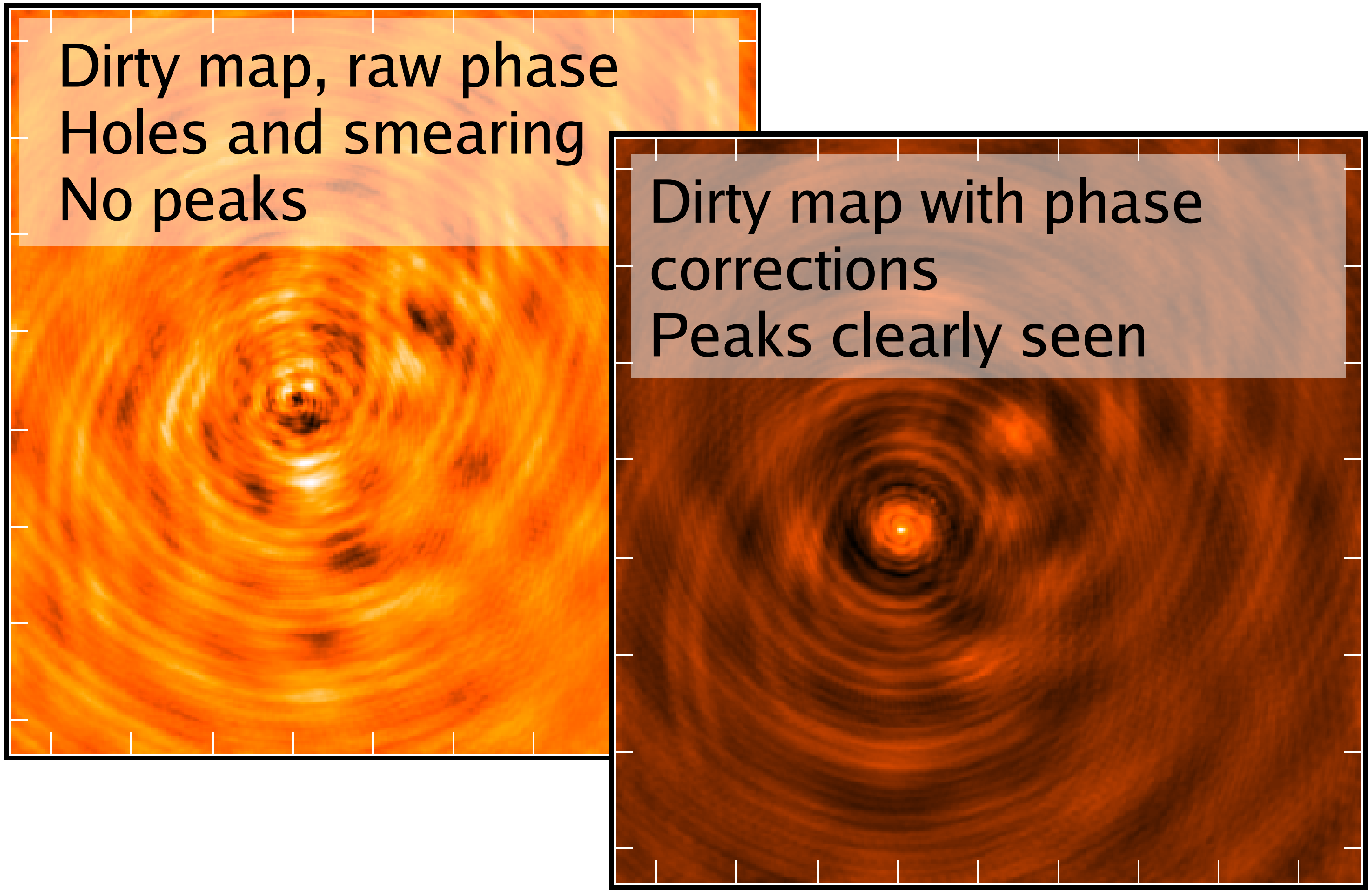
# Interpolated onto target:



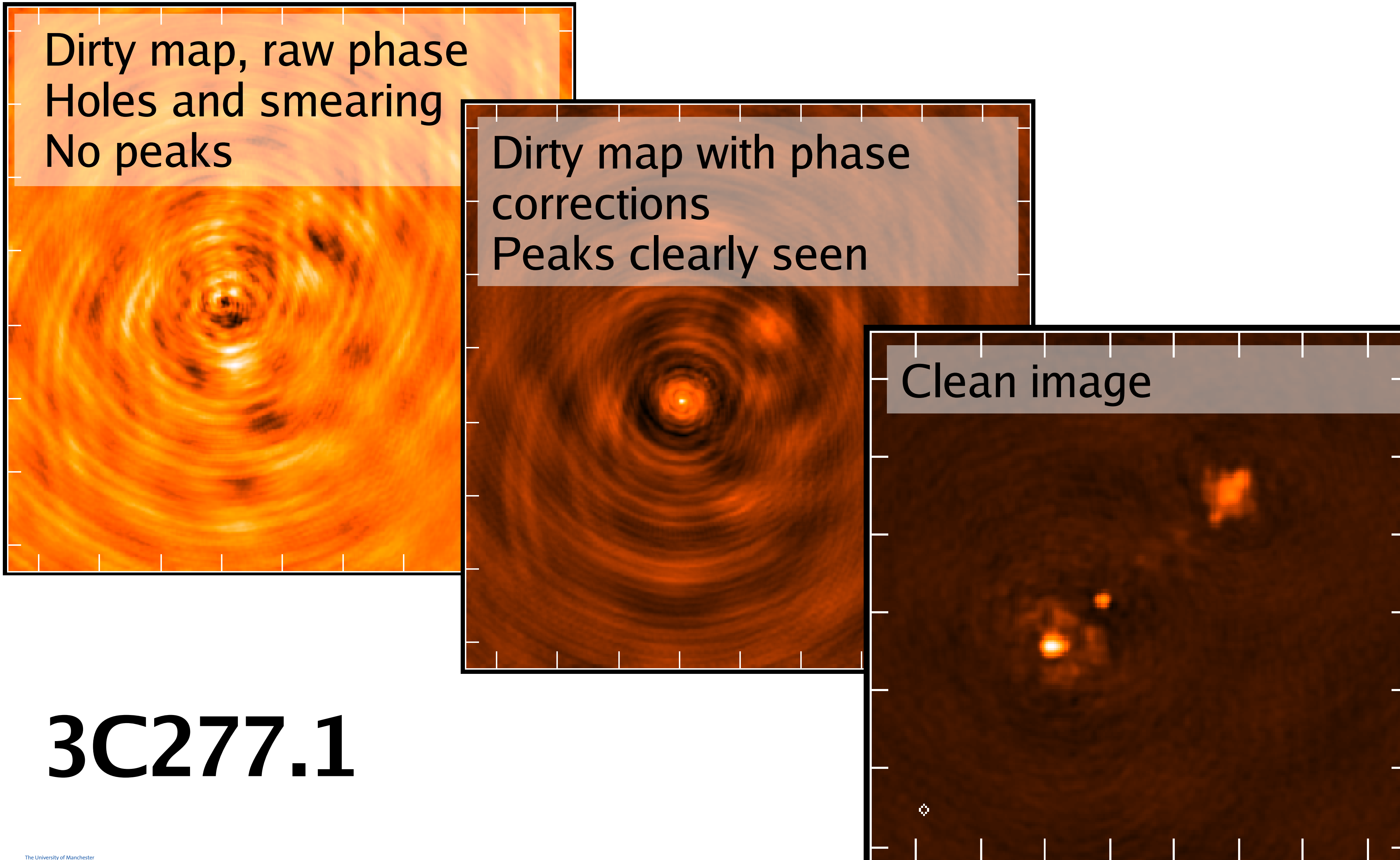
# Makes all the difference...



# Makes all the difference...



# Makes all the difference...




# 3C277.1

# Other ways to steer tasks

- Interactive input into tasks is usually like
  - `default gaincal`
    - or else previous values are used
  - `solint='1min'`
- Scripts - see examples e.g. Jupiter.py
  - Execute inside CASA using `execfile('jupiter.py')`
- Call tasks as functions
  - `gaincal(vis='3C277.1C.ms',calmode='p', solint='1min')`
  - In this mode, unset values are always defaulted

# Toolkit

- Access to functions using Python
- NGC 5921 example
  - Assign variable to name of MS to operate on
    - `srcsplitms = vis + 'contsub'`
  - Use **ms** tool and Python functions
    - `ms.open(srcsplitms)`
    - `thistest_src=max(ms.range(['amplitude']).get('amplitude'))`
    - `ms.close()`
  - This measures the maximum amplitude in the MS
    - `print thistest_src`
      - rerturns 46.0655708313 (peak visibility amplitude, Σchannels)
- Many tools will eventually become tasks
  - `toolhelp` to see list

# AIPS or CASA? (or either or both)

- (E)VLA, ATCA, CARMA etc., eventually ALMA
  - CASA for wide-field, wide band capabilities
    - Recognises X/Y as well as L/R polarizations
    - Polynomial interpolation in calibration
  - Cookbook recipes fit mid-range (E)VLA well
- *e*-MERLIN, VLBI, EVLA high $\nu$/extended array
  - Shorter solution intervals (10s - 10min)
    - Correct phase-only before phase and amplitude
  - Choose image pixel size $\lesssim$1/3 beam size
  - FRINGE to solve for phase rate/delay
  - Heterogenous arrays
    - Weight by antenna sensitivity not sample variance
    - Uniform weighting/strong tapering increases noise
  - CASA for wide-field, wide band imaging
    - Mixed antenna diameters for PB scaling

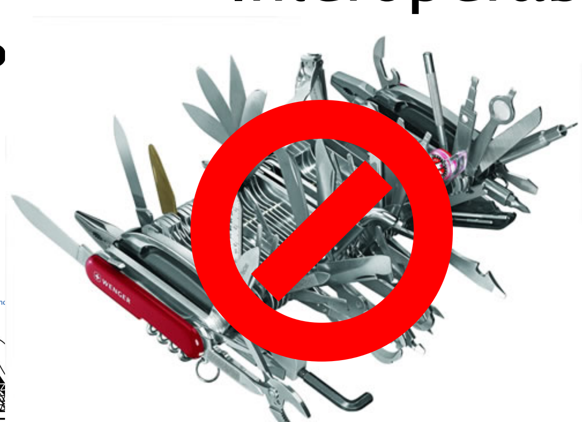
# Choose the right tool for the job

- Innovative projects or observatory staff developer
  - Use toolkit or write your own software
  - Construct pipelines
    - Document for future use

The University of Manchester
Jodrell Bank Observatory

# Choose the right tool for the job

- Innovative projects or observatory staff developer
  - Use toolkit or write your own software
  - Construct pipelines
    - Document for future use
- Experienced radio interferometer user
  - Use pipelined *uv* cal, guided by proposal metadata
  - Refinine self-calibration/editing
    - Adapt recipe parameters for array, custom products
  - Prefer to use a familiar package
    - Interoperability essential

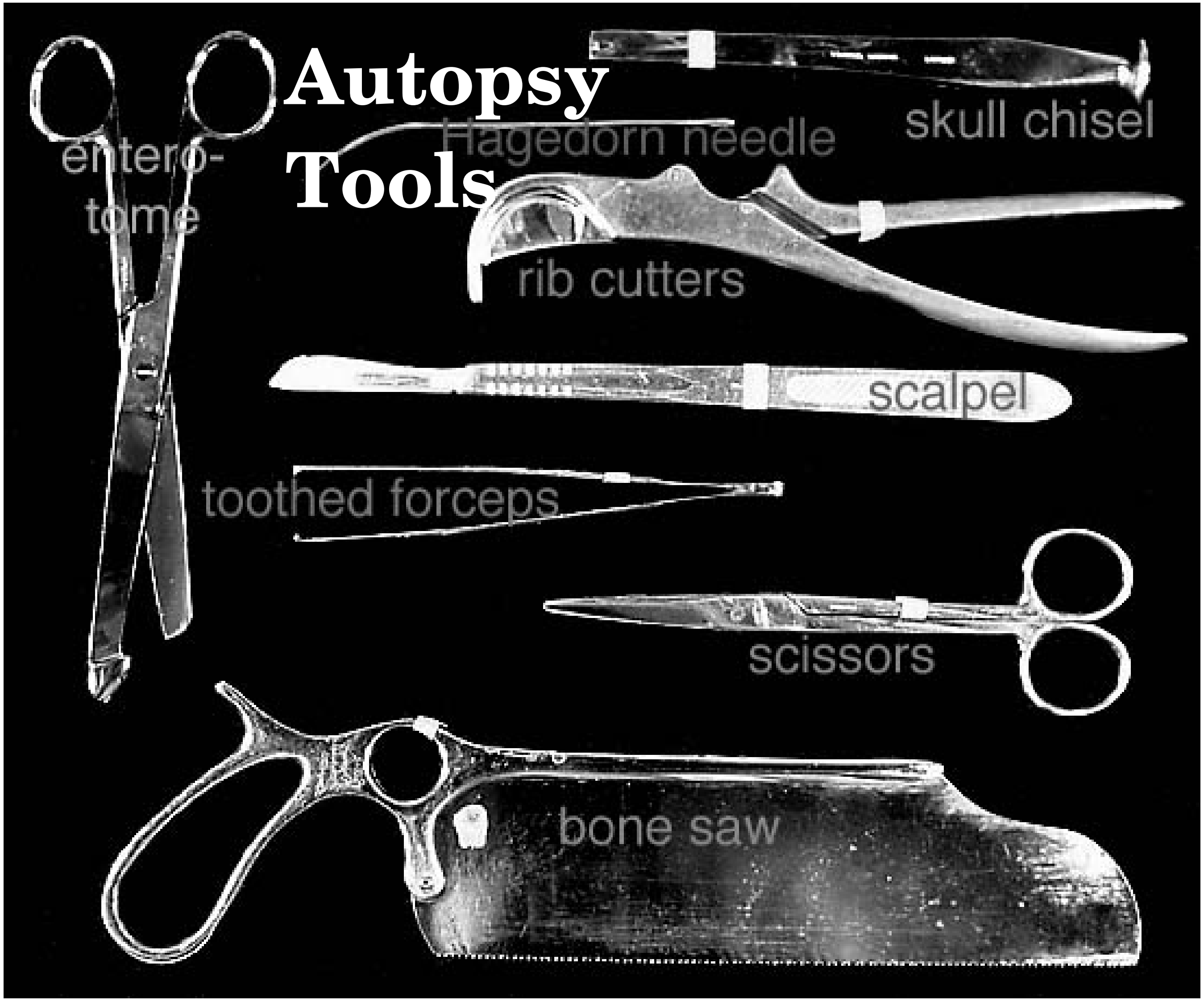


# Choose the right tool for the job

- Innovative projects or observatory staff developer
  - Use toolkit or write your own software
  - Construct pipelines
    - Document for future use
- Experienced radio interferometer user
  - Use pipelined *uv* cal, guided by proposal metadata
  - Refinine self-calibration/editing
    - Adapt recipe parameters for array, custom products
  - Prefer to use a familiar package
    - Interoperability essential
- Beginner - or just in a hurry
  - Jargon-free pipeline interface
    - Steer within instrumental constraints
  - Clear explanation of pitfalls/artefacts

# Keep a full processing history

- Use scripts, or
- Note parameter values
  - Examples for further processing
  - Troubleshooting postmortem



# CASA demos 2: MERLIN & ATCA

- If you have time - see web site for links
  - FITS data, AIPS & CASA scripts
- MERLIN: 3C277.1 polarized continuum
  - Higher resolution
  - Faster phase rate
    - More information this afternoon
- ATCA: J2342-44 HI spectral line
  - Lower resolution
  - Many sources in continuum field
- Scripts provide examples of strategies for a range of array parameters

# MERLIN CASA calibration flowchart

Approx flux and bandpass scaling pre-applied

Set known fluxes (allowing for resolution if necessary)

Phase-cal calibration sources

A&P self-cal for phase ref, bp cal source

Derive bandpass cal if required

Fluxscale if required

Solve for pol. leakage, usually with phase ref source, apply

Correct pol angle, usually with 3C286, apply

Apply phase-ref solutions and image target

Phase self-cal if enough SNR

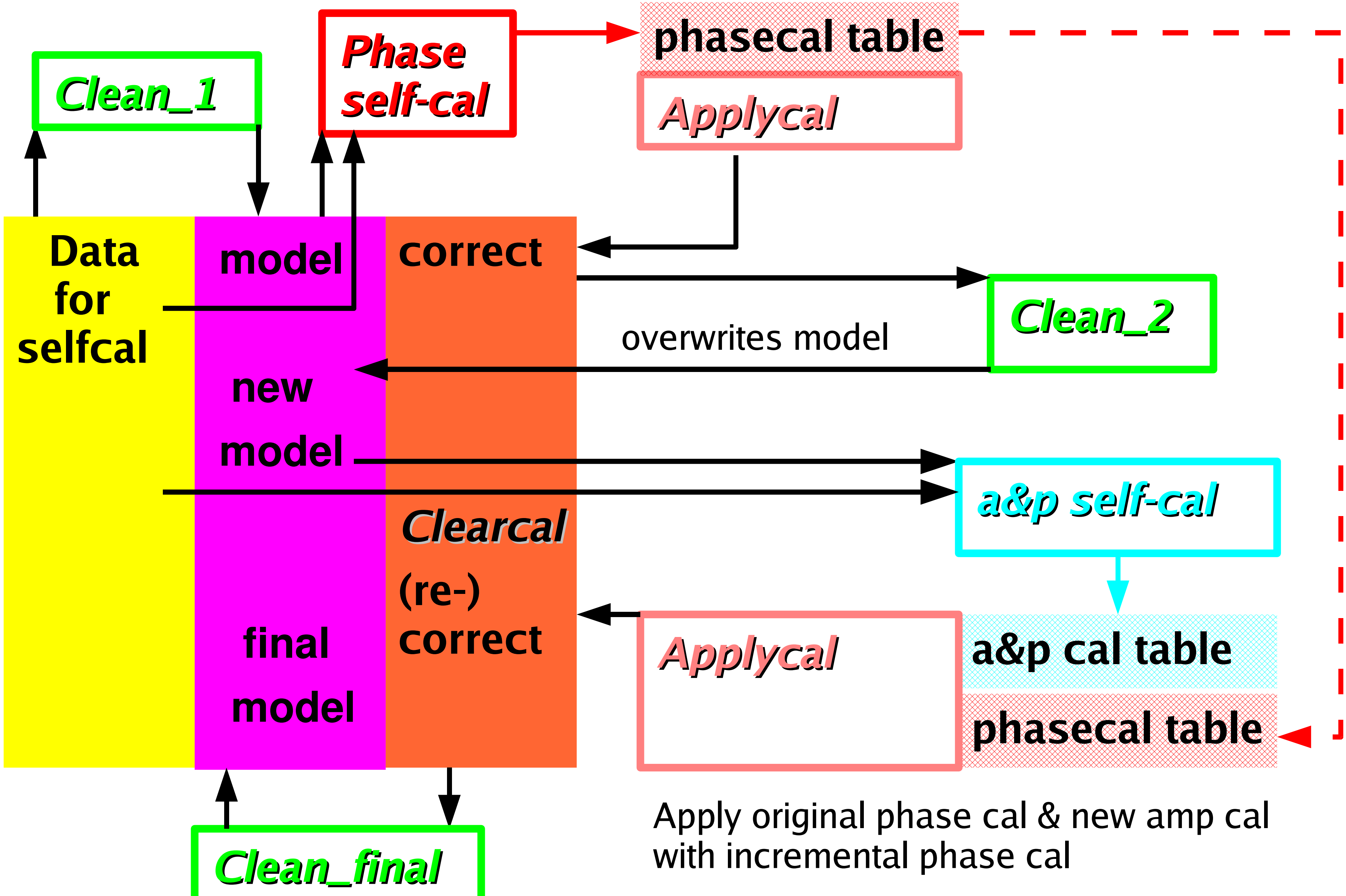
Image target, A&P self-cal if enough SNR

Final target imaging

Calibration is incremental. Apply as required.

Inspect data and solutions regularly. Flag if required.

# Self-Calibration minimising data volume



# More about interferometry

- European Radio Interferometry School
  - Annual event supported by RadioNet
    - http://www.radionet-eu.org
  - Alternates between long and short wavelengths
    - but a bit of everything at each event
- ERIS 2009 cm-wavelength focussed
  - e-MERLIN, EVN, EVLA etc.
    - AIPS and CASA
  - Oxford, UK, 7-11 September 2009
    - Registration open (but nearly full)
- http://astrowiki.physics.oc.ac.uk/ERIS2009
  - or just Google ERIS2009