Radio Interferometry packages and formats

Anita Richards

UK ALMA Regional Centre

JBCA, University of Manchester

International radio arrays

Omitting specialised e.g. CMB, solar arrays

Space VLBI (Russia/Japan/ Global)

WSRT (NL)

e-MERLIN (UK)

LOFAR (NL/W.Europe)

IRAM (F)

EVN +

VERA (Jap)

VLBA (USA)

KVASAR

SMA, CARMA (USA)

KVN

VLA(USA/Mexico)

GMRT (India)

And more being developed all the time!

ALMA (ESO/N.America /E.Asia/Chile)

ATCA, LBA (Aus)

SKA and pathfinders (S.Africa/Aus/Global; project office UK)

Global Very Long Baseline Interferometry

Summary

MIRIAD

AIPS

GILDAS SOFTWARE

NEWSTAR

GIPSY

DIFMAP

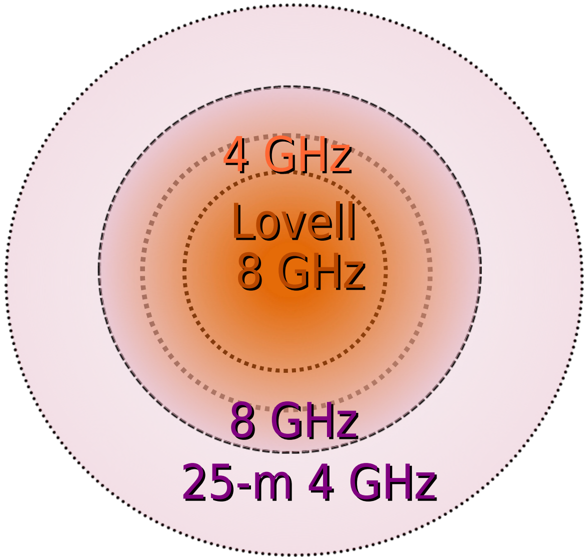
AIPS++

Astronomical Information Processing System

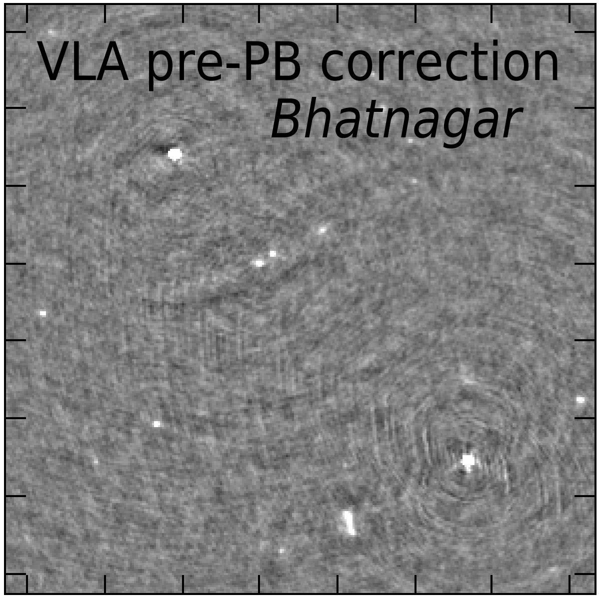
CASA

New-generation array demands

- Wide-field imaging
  - Mixed antenna diameters
  - Narrow channels, short integrations
    - GB - TB data sets
  - Subtract confusing sources
    - 3D faceting and *w*-projection
  - Mosaicing
  - Non-isoplanatic fields – see LOFAR talk
- Huge raw data volumes
  - Pipelines and parallelisation
  - Automate flagging where possible
- Wide-Band imaging
  - Spectral curvature
  - Mixed spectral and continuum configurations

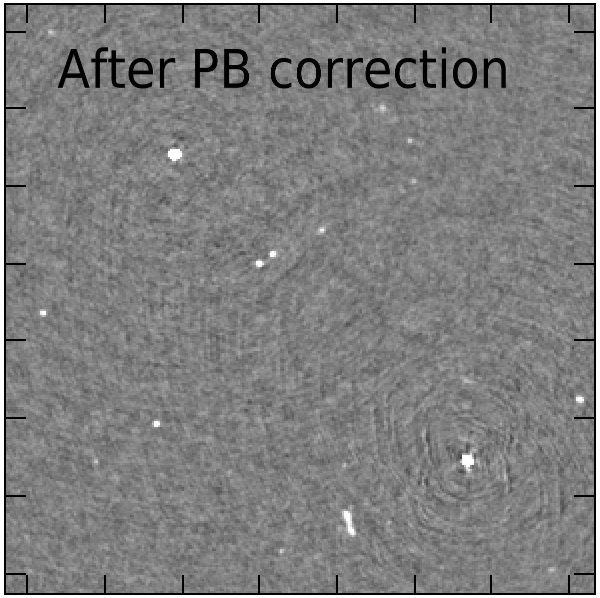


New-generation array demands



- Wide-field imaging
  - Mixed antenna diameters
  - Narrow channels, short integrations
    - GB - TB data sets
  - Subtract confusing sources
    - 3D faceting and *w*-projection
  - Mosaicing
  - Non-isoplanatic fields – see LOFAR talk
- Huge raw data volumes
  - Pipelines and parallelisation
  - Automate flagging where possible
- Wide-Band imaging
  - Spectral curvature
  - Mixed spectral and continuum configurations

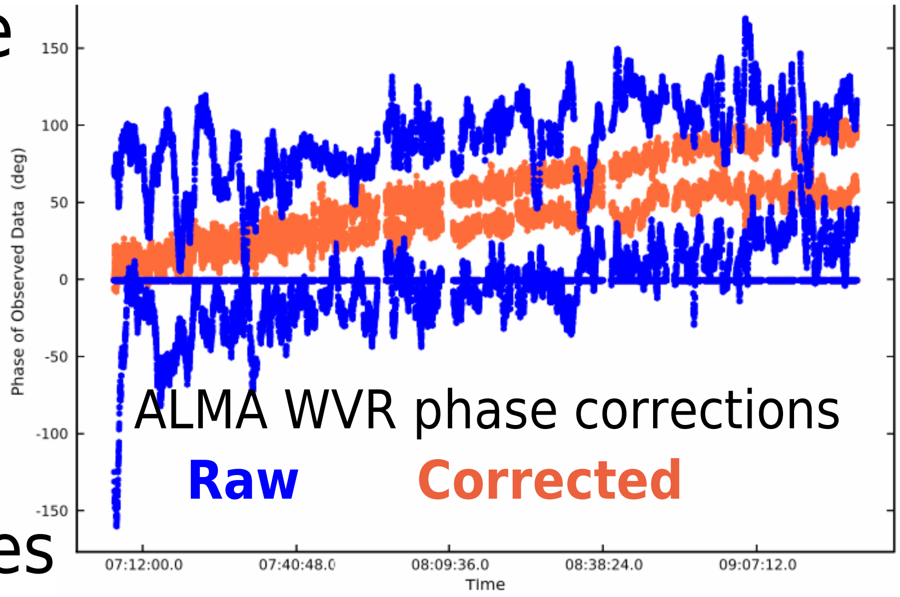
New-generation array demands



- Wide-field imaging
  - Mixed antenna diameters
  - Narrow channels, short integrations
    - GB - TB data sets
  - Subtract confusing sources
    - 3D faceting and *w*-projection
  - Mosaicing
  - Non-isoplanatic fields – see LOFAR talk
- Huge raw data volumes
  - Pipelines and parallelisation
  - Automate flagging where possible
- Wide-Band imaging
  - Spectral curvature
  - Mixed spectral and continuum configurations

Atmospheric correction techniques

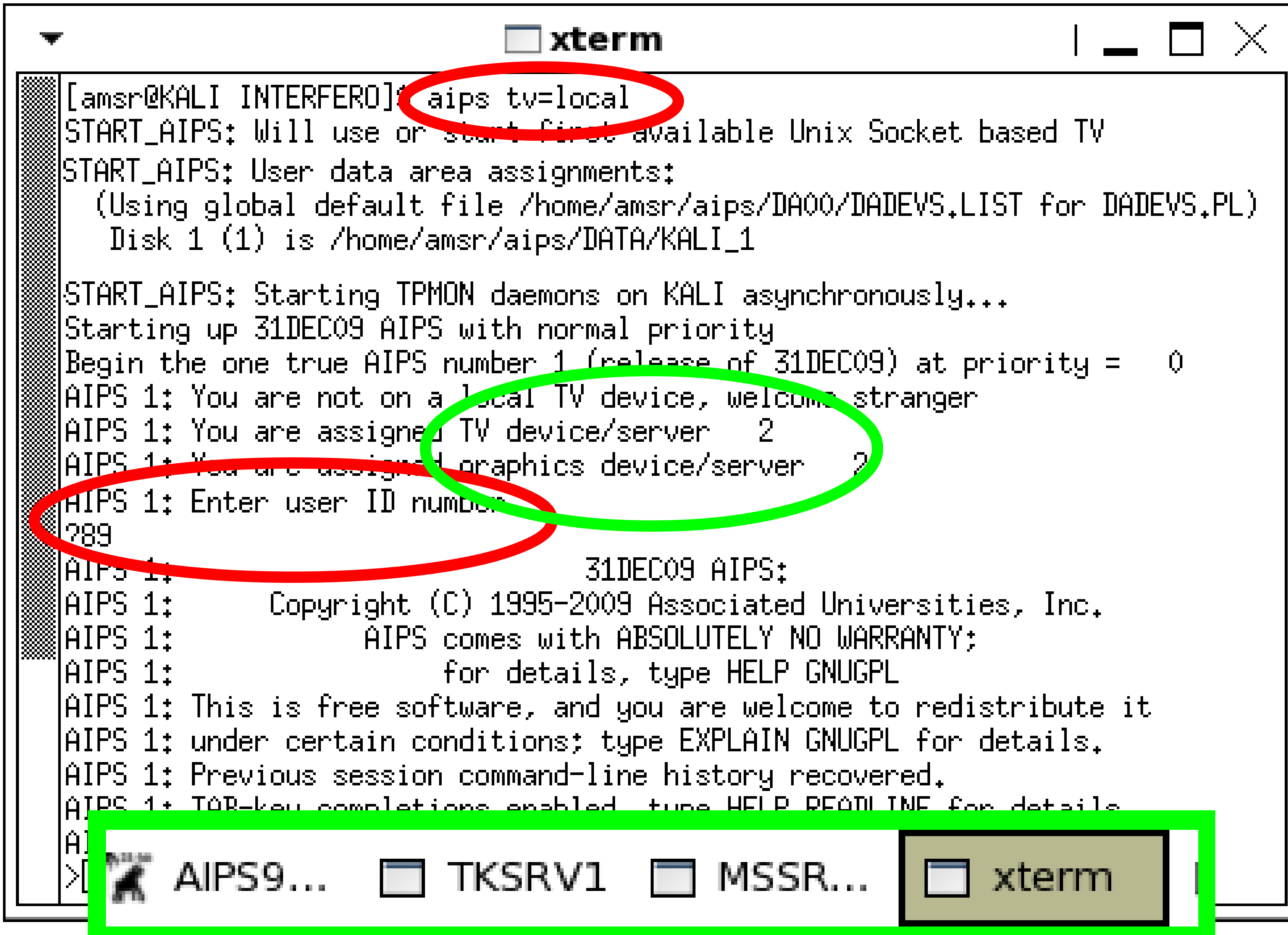
- Low frequencies - ionosphere
  - Polarization affected
    - GPS measurements
- High frequencies – water in troposphere
  - Phase rotated rapidly
  - Absorption affects amplitudes
  - Rapid switching between phase-ref/target
    - Solve for rate or fit polynomials to phases
  - Water vapour (WVR) & T_{sys} measurements
    - Apply correction tables
  - Refractive phase effects $\propto \nu$ - "delay"
- Transfer solutions between data sets

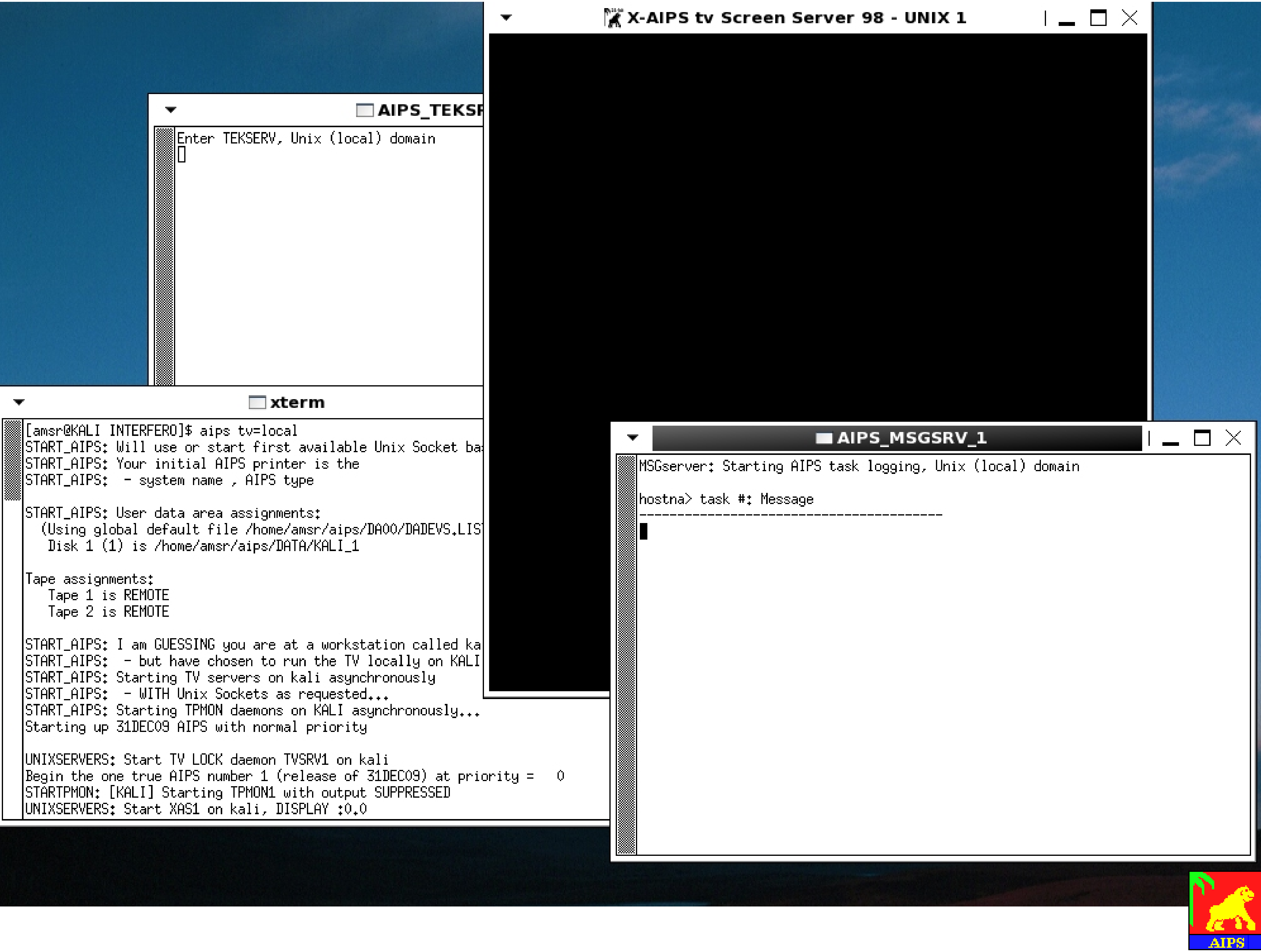


Astronomical Image Processing System

- Originated by NRAO for VLA in 1978
  - Fortran, C
  - Limited built-in scripting/math operations
  - Historically most widely used package for cm-wave
    - VLA, MERLIN, most VLBI ... many more interferometers
    - Some support for single dish and any FITS images
  - Very wide functionality from calibration to analysis
- Especially good for specialised VLBI calibration
- Many sophisticated image analysis tasks
- Python wrapper (Parseltongue) for easier scripting

Starting AIPS




X-AIPS tv Screen Server 98 - UNIX 1
AIPS_TEKSR
Enter TEKSERV, Unix (local) domain
xterm
[amsr@KALI INTERFERO]$ aips tv=local
START_AIPS: Will use or start first available Unix Socket ba
START_AIPS: Your initial AIPS printer is the
START_AIPS: - system name , AIPS type
START_AIPS: User data area assignments:
(Using global default file /home/amsr/aips/DA00/DADEVS.LIS
Disk 1 (1) is /home/amsr/aips/DATA/KALI_1
Tape assignments:
Tape 1 is REMOTE
Tape 2 is REMOTE
START_AIPS: I am GUESSING you are at a workstation called ka
START_AIPS: - but have chosen to run the TV locally on KALI
START_AIPS: Starting TV servers on kali asynchronously
START_AIPS: - WITH Unix Sockets as requested...
START_AIPS: Starting TPMON daemons on KALI asynchronously...
Starting up 31DEC09 AIPS with normal priority
UNIXSERVERS: Start TV LOCK daemon TVSRV1 on kali
Begin the one true AIPS number 1 (release of 31DEC09) at priority = 0
STARTPMON: [KALI] Starting TPMON1 with output SUPPRESSED
UNIXSERVERS: Start XAS1 on kali, DISPLAY :0.0
AIPS_MSGSRV_1
MSGserver: Starting AIPS task logging, Unix (local) domain
hostna> task #: Message
AIPS

AIPS jargon

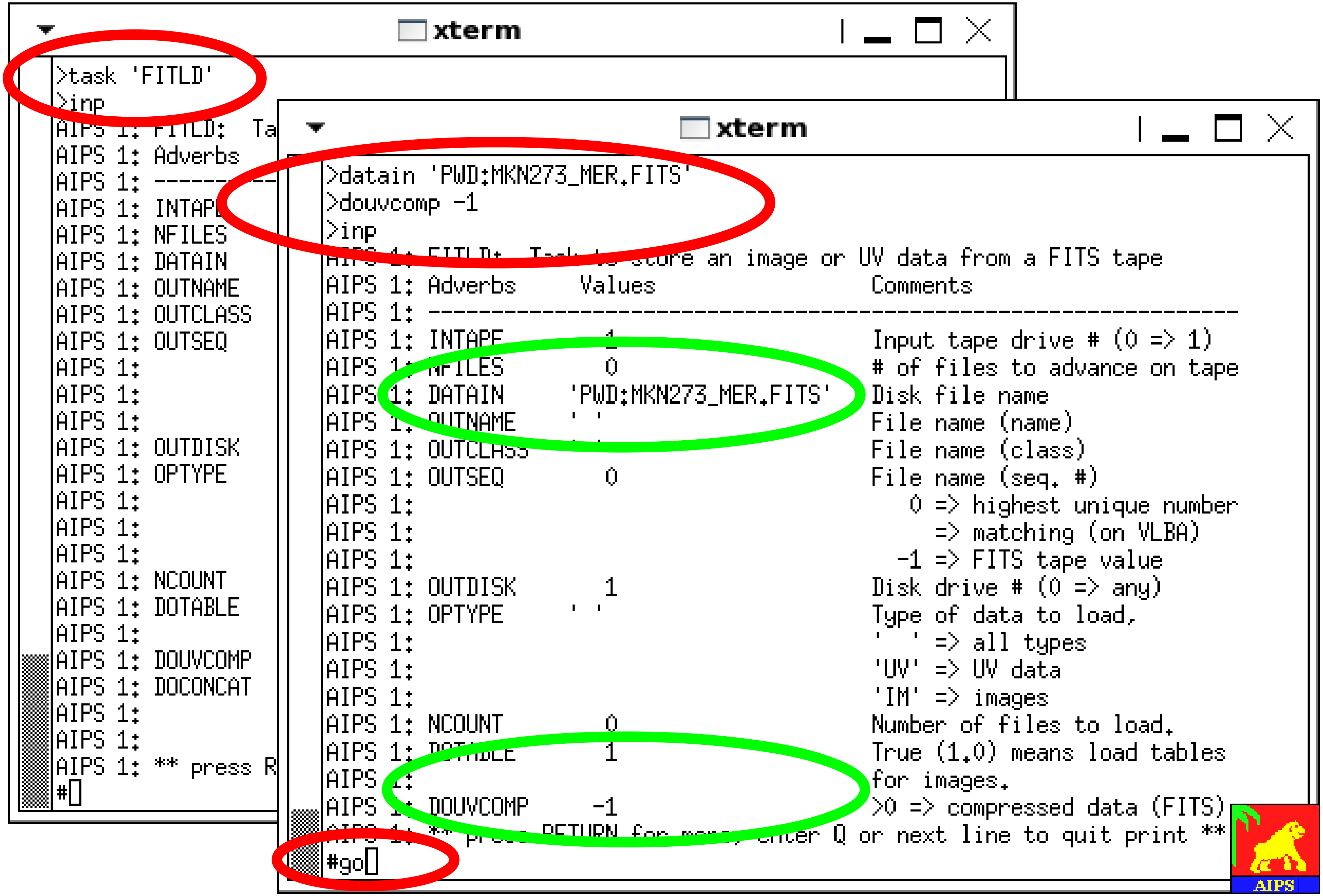
- Major operations are performed using `Tasks`
  - `FITLD` loads data, `CALIB` performs calibration etc.
- Input parameters to `Tasks` are set by `Verbs`
  - >Task 'CALIB'; CALSOUR 'MKN273'; SOLINT 1
  - Words/names in 'inverted commas'; numbers bare
  - *Not* case sensitive, in general
  - Inside AIPS, 12-character limit on file/source names
- To set all defaults: >`RESTORE 0`
  - Beware: will give values typical for VLA data
    - You will have to set parameters suitable for your data
- To exit and kill all AIPS windows: >KLEENEX

Loading data into AIPS

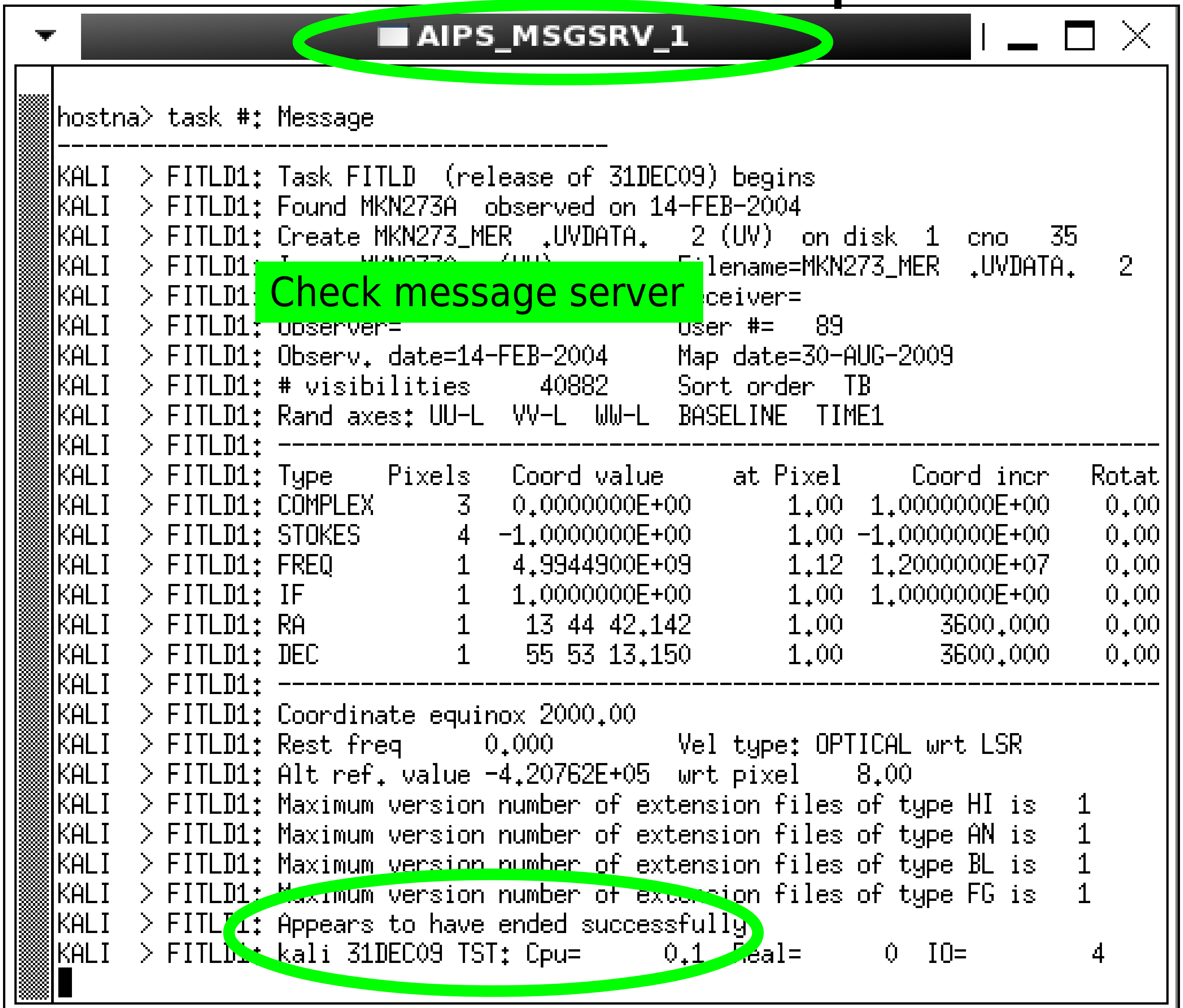
xterm

```
>task 'FITLD'
>inp
AIPS 1: FITLD:  Task to store an image or UV data from a FITS tape
AIPS 1: Adverbs     Values                 Comments
AIPS 1: ----------------------------------------------------------------
AIPS 1: INTAPE        1                    Input tape drive # (0 => 1)
AIPS 1: NFILES        0                    # of files to advance on tape
AIPS 1: DATAIN      *all ' '               Disk file name
AIPS 1: OUTNAME     ' '                    File name (name)
AIPS 1: OUTCLASS    ' '                    File name (class)
AIPS 1: OUTSEQ        0                    File name (seq. #)
AIPS 1:                                       0 => highest unique number
AIPS 1:                                         => matching (on VLBA)
AIPS 1:                                      -1 => FITS tape value
AIPS 1: OUTDISK       1                    Disk drive # (0 => any)
AIPS 1: OPTYPE      ' '                    Type of data to load,
AIPS 1:                                    '  ' => all types
AIPS 1:                                    'UV' => UV data
AIPS 1:                                    'IM' => images
AIPS 1: NCOUNT        0                    Number of files to load.
AIPS 1: DOTABLE       1                    True (1.0) means load tables
AIPS 1:                                    for images.
AIPS 1: DOUVCOMP      1                    >0 => compressed data (FITS)
AIPS 1: DOCONCAT     -1                    >0 -> if VLBA correlator data
AIPS 1:                                    append data to existing
AIPS 1:                                    files, or if no appropriate
AIPS 1: ** press RETURN for more, enter Q or next line to quit print **
#
```

Loading data into AIPS



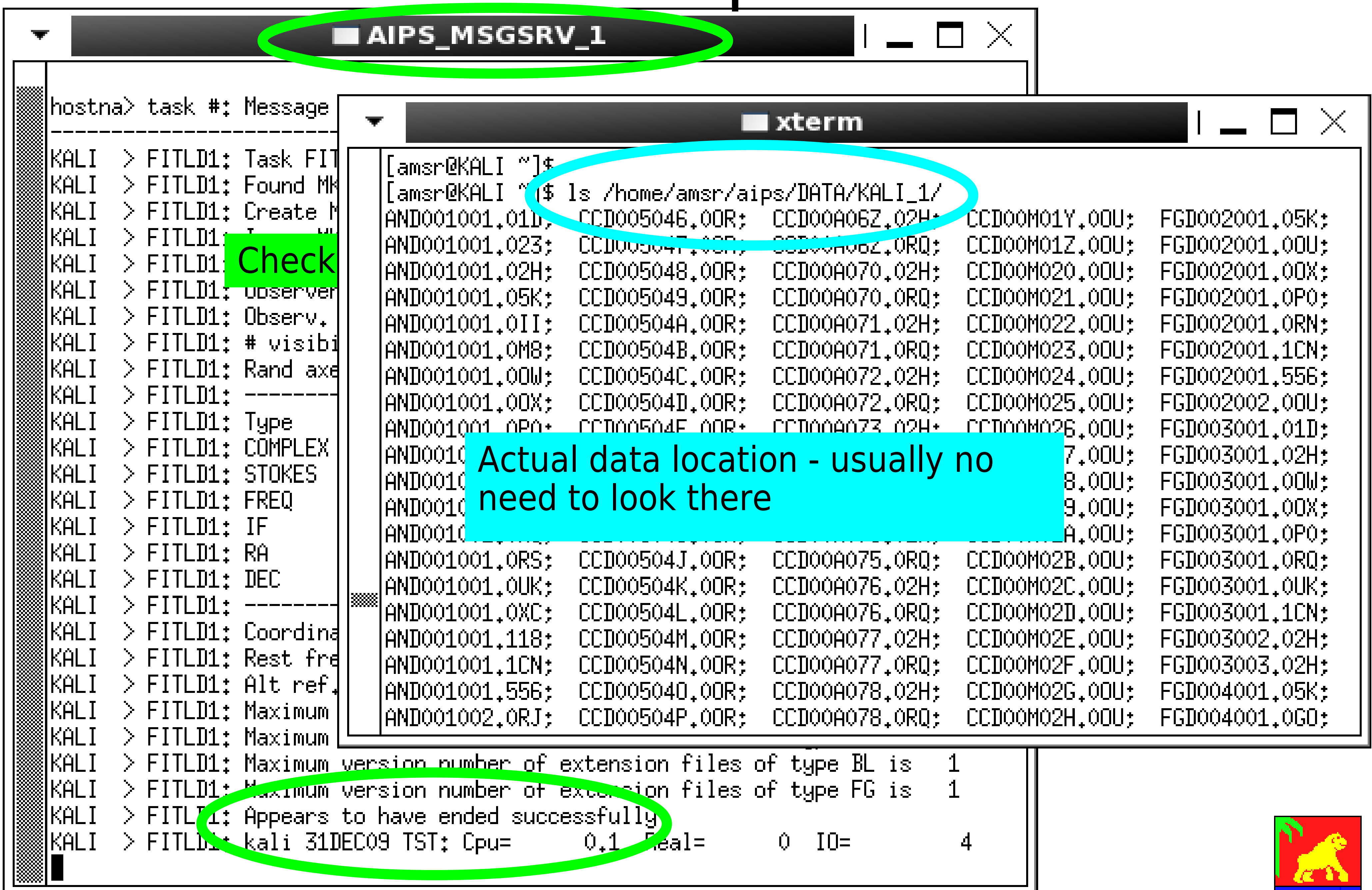
Where does AIPS put data?



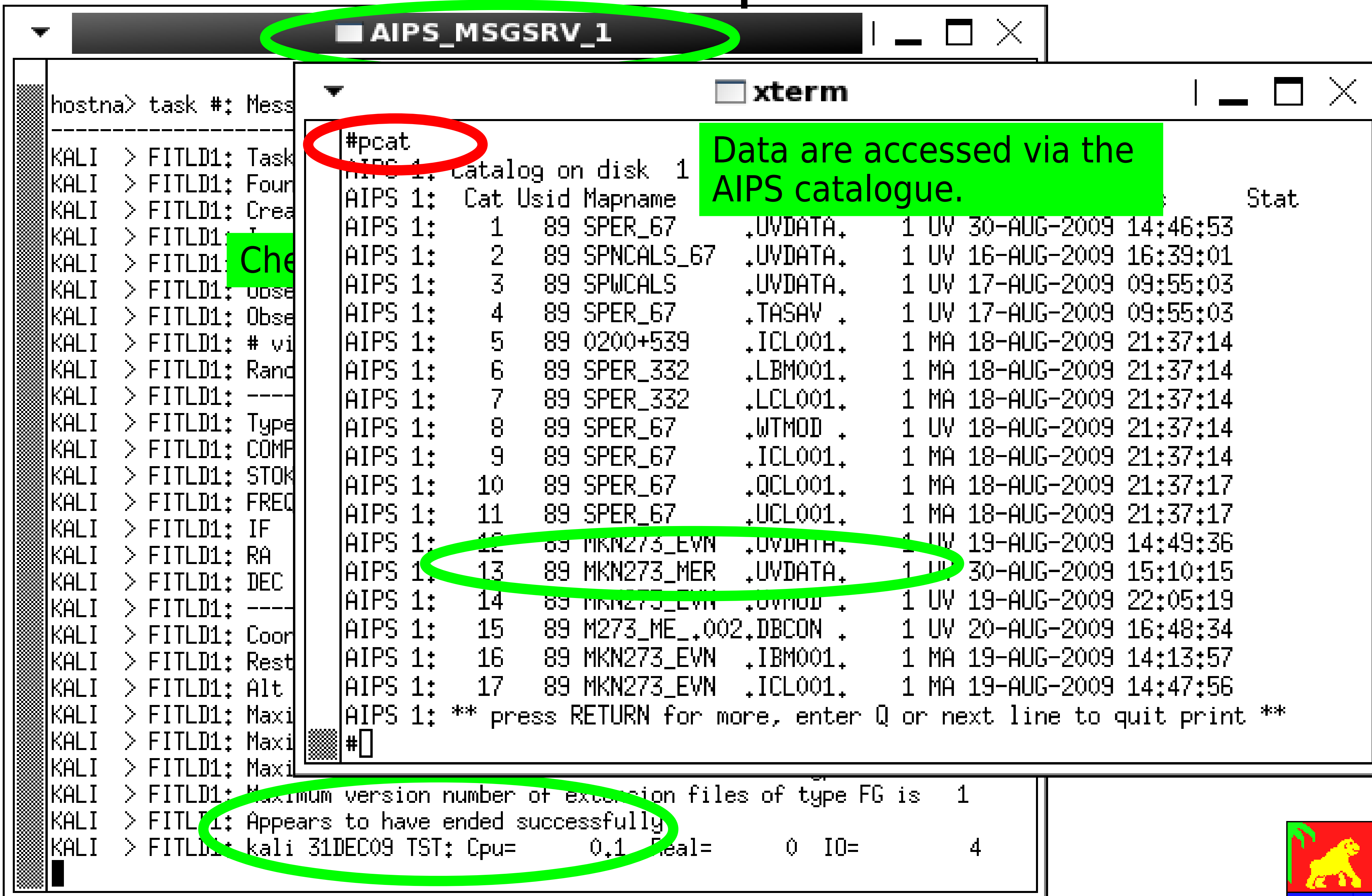
```
AIPS_MSGSRV_1

hostna> task #: Message
----------------------------------------
KALI   > FITLD1: Task FITLD  (release of 31DEC09) begins
KALI   > FITLD1: Found MKN273A  observed on 14-FEB-2004
KALI   > FITLD1: Create MKN273_MER  .UVDATA.   2 (UV)  on disk  1  cno   35
KALI   > FITLD1:                               Filename=MKN273_MER  .UVDATA.   2
KALI   > FITLD1:                               ceiver=
KALI   > FITLD1:                               User #=    89
KALI   > FITLD1: Observ. date=14-FEB-2004      Map date=30-AUG-2009
KALI   > FITLD1: # visibilities      40882     Sort order  TB
KALI   > FITLD1: Rand axes: UU-L  VV-L  WW-L   BASELINE  TIME1
KALI   > FITLD1: --------------------------------------------------------------
KALI   > FITLD1: Type    Pixels   Coord value     at Pixel     Coord incr   Rotat
KALI   > FITLD1: COMPLEX      3   0.0000000E+00       1.00  1.0000000E+00    0.00
KALI   > FITLD1: STOKES       4  -1.0000000E+00       1.00 -1.0000000E+00    0.00
KALI   > FITLD1: FREQ         1   4.9944900E+09       1.12  1.2000000E+07    0.00
KALI   > FITLD1: IF           1   1.0000000E+00       1.00  1.0000000E+00    0.00
KALI   > FITLD1: RA           1    13 44 42.142       1.00       3600.000    0.00
KALI   > FITLD1: DEC          1    55 53 13.150       1.00       3600.000    0.00
KALI   > FITLD1: --------------------------------------------------------------
KALI   > FITLD1: Coordinate equinox 2000.00
KALI   > FITLD1: Rest freq      0.000          Vel type: OPTICAL wrt LSR
KALI   > FITLD1: Alt ref. value -4.20762E+05  wrt pixel    8.00
KALI   > FITLD1: Maximum version number of extension files of type HI is   1
KALI   > FITLD1: Maximum version number of extension files of type AN is   1
KALI   > FITLD1: Maximum version number of extension files of type BL is   1
KALI   > FITLD1: Maximum version number of extension files of type FG is   1
KALI   > FITLD1: Appears to have ended successfully
KALI   > FITLD1: kali 31DEC09 TST: Cpu=      0.1  Real=      0   IO=         4
```

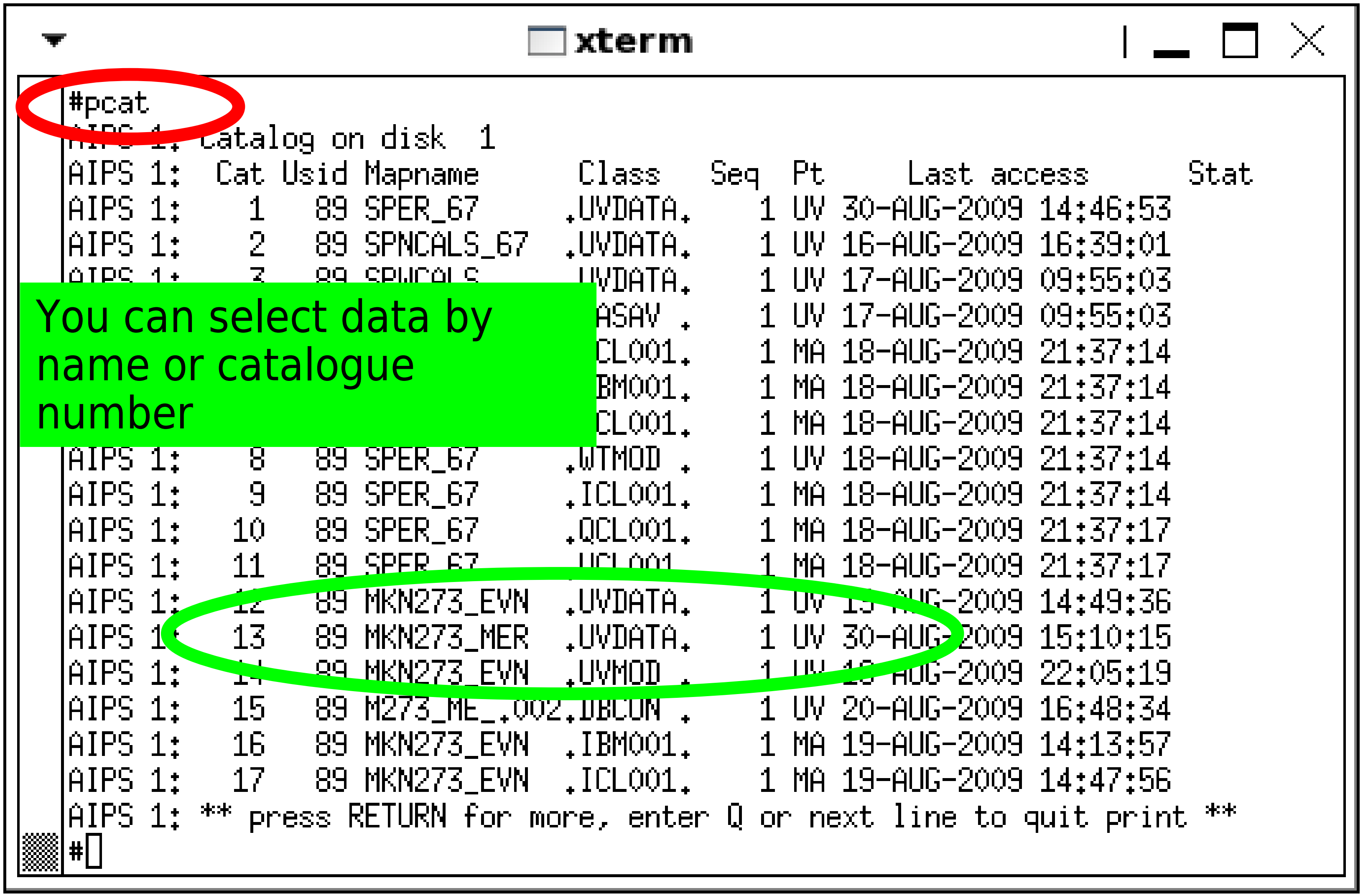
Where does AIPS put data?



Where does AIPS put data?



What's in the data?



What's in the data?

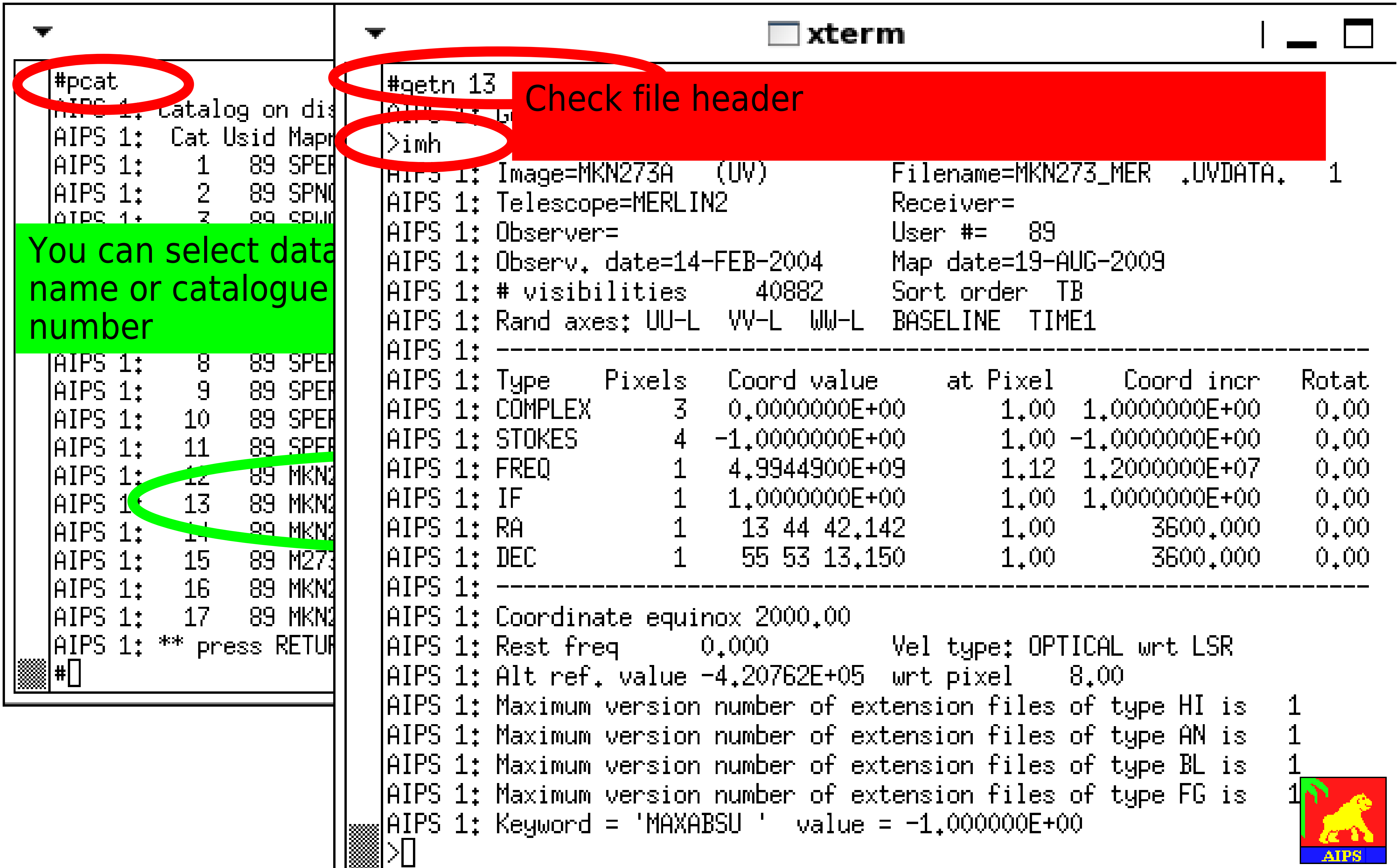
```
#pcat
AIPS 1: Catalog on dis
AIPS 1:  Cat Usid Mapr
AIPS 1:    1   89 SPER
AIPS 1:    2   89 SPN0
AIPS 1:    3   89 SPW0
AIPS 1:    8   89 SPER
AIPS 1:    9   89 SPER
AIPS 1:   10   89 SPER
AIPS 1:   11   89 SPER
AIPS 1:   12   89 MKN2
AIPS 1:   13   89 MKN2
AIPS 1:   14   89 MKN2
AIPS 1:   15   89 M273
AIPS 1:   16   89 MKN2
AIPS 1:   17   89 MKN2
AIPS 1: ** press RETUR
#
```

You can select data name or catalogue number

xterm

```
#getn 13
AIPS 1: Got(1) disk= 1  user=  89   type=UV   MKN273_MER.UVDATA.1
>imh
AIPS 1: Image=MKN273A   (UV)          Filename=MKN273_MER  .UVDATA.   1
AIPS 1: Telescope=MERLIN2             Receiver=
AIPS 1: Observer=                     User #=   89
AIPS 1: Observ. date=14-FEB-2004      Map date=19-AUG-2009
AIPS 1: # visibilities     40882      Sort order  TB
AIPS 1: Rand axes: UU-L  VV-L  WW-L   BASELINE  TIME1
AIPS 1: ----------------------------------------------------------------
AIPS 1: Type    Pixels   Coord value     at Pixel     Coord incr   Rotat
AIPS 1: COMPLEX      3   0.0000000E+00       1.00  1.0000000E+00    0.00
AIPS 1: STOKES       4  -1.0000000E+00       1.00 -1.0000000E+00    0.00
AIPS 1: FREQ         1   4.9944900E+09       1.12  1.2000000E+07    0.00
AIPS 1: IF           1   1.0000000E+00       1.00  1.0000000E+00    0.00
AIPS 1: RA           1    13 44 42.142       1.00       3600.000    0.00
AIPS 1: DEC          1    55 53 13.150       1.00       3600.000    0.00
AIPS 1: ----------------------------------------------------------------
AIPS 1: Coordinate equinox 2000.00
AIPS 1: Rest freq      0.000          Vel type: OPTICAL wrt LSR
AIPS 1: Alt ref. value -4.20762E+05  wrt pixel    8.00
AIPS 1: Maximum version number of extension files of type HI is   1
AIPS 1: Maximum version number of extension files of type AN is   1
AIPS 1: Maximum version number of extension files of type BL is   1
AIPS 1: Maximum version number of extension files of type FG is   1
AIPS 1: Keyword = 'MAXABSU '  value = -1.000000E+00
>
```

What's in the data?



UV data header

Type | Name | Class | Seq. No

```
type=UV    MKN273_MER.UVDATA.1

AIPS 1: Image=MKN273A    (UV)            Filename=MKN273_MER  .UVDATA.   1
AIPS 1: Telescope=MERLIN2                Receiver=
AIPS 1: Observer=                        User #=    89
      : Observ. date=14-FEB-2004         Map date=19-AUG-2009
      : # visibilities     40882         Sort order  TB
      : Rand axes: UU-L  VV-L  WW-L      BASELINE  TIME1
AIPS 1: ------------------------------------------------------------------
      : Type    Pixels   Coord value       at Pixel    Coord incr   Rotat
      : COMPLEX      3   0.0000000E+00       1.00   1.0000000E+00   0.00
      : STOKES       4  -1.0000000E+00       1.00  -1.0000000E+00   0.00
      : FREQ         1   4.9944900E+09       1.12   1.2000000E+07   0.00
      : IF           1   1.0000000E+00       1.00   1.0000000E+00   0.00
      : RA           1    13 44 42.142       1.00        3600.000   0.00
      : DEC          1    55 53 13.150       1.00        3600.000   0.00
AIPS 1: ------------------------------------------------------------------
AIPS 1: Coordinate equinox 2000.00
AIPS 1: Rest freq      0.000             Vel type: OP
AIPS 1: Alt ref. value -4.20762E+05      wrt pixel
AIPS 1: Maximum version number of extension files of type HI is   1
AIPS 1: Maximum version number of extension files of type AN is   1
AIPS 1: Maximum version number of extension files of type BL is   1
AIPS 1: Maximum version number of extension files of type FG is   1
AIPS 1: Keyword = 'MAXABSU '  value = -1.000000E+00
```

Axes: Visibilities

Amp, f, weight
LL RR LR RL
Hz
Sub-band
Pos
Pos

Extension tables

Image data

xterm

```
>getn 20;imh
AIPS 1: Got(1)   disk= 1  user=  89   type=MA   MKN273_MER.ICL001.1
AIPS 1: Image=MKN273A   (MA)          Filename=MKN273_MER  .ICL001.   1
AIPS 1: Telescope=MERLIN2             Receiver=
AIPS 1: Observer=                     User #=   89
AIPS 1: Observ. date=14-FEB-2004      Map date=19-AUG-2009
AIPS 1: Minimum=-4.29469685E-04       Maximum= 7.45257037E-03 JY/BEAM
AIPS 1: ----------------------------------------------------------------
AIPS 1: Type    Pixels   Coord value     at Pixel     Coord incr   Rotat
AIPS 1: RA---SIN   512    13 44 42.142     256.00      -0.015000    0.00
AIPS 1: DEC--SIN   512    55 53 13.150     257.00       0.015000    0.00
AIPS 1: FREQ         1   4.9929902E+09       1.00  1.2000000E+07    0.00
AIPS 1: STOKES       1   1.0000000E+00       1.00  1.0000000E+00    0.00
AIPS 1: ----------------------------------------------------------------
AIPS 1: Coordinate equinox 2000.00
AIPS 1: Map type=NORMAL               Number of iterations=    1000
AIPS 1: Conv size= 0.13732 X 0.06835   Position angle= -22.69
AIPS 1: Rest freq      0.000          Vel type: OPTICAL wrt LSR
AIPS 1: Alt ref. value -4.20762E+05  wrt pixel    8.00
AIPS 1: Maximum version number of extension files of type CC is    1
AIPS 1: Maximum version number of extension files of type HI is    1
AIPS 1: Keyword = 'CCFLUX  '  value =  4.341595E-02
AIPS 1: Keyword = 'CCTOTAL '  value =  4.341595E-02
```

Image data

xterm

Type | Name | Class | Seq. No.

```
                              ~=    89    type=MA    MKN273_MER.ICL001.1
                                         Filename=MKN273_MER  .ICL001.   1
                                         Receiver=
AIPS 1: Observer=                        User #=   89
AIPS 1: Observ. date=14-FEB-2004         Map date=19-AUG-2009
AIPS 1: Minimum=-4.29469685E-04          Maximum= 7.45257037E-03 JY/BEAM
AIPS 1: ----------------------------------------------------------------
      : Type      Pixels    Coord value       at Pixel     Coord incr    Rotat
      : RA---SIN    512     13 44 42.142       256.00       -0.015000     0.00
      : DEC--SIN    512     55 53 13.150       257.00        0.015000     0.00
      : FREQ          1    4.9929902E+09         1.00   1.2000000E+07     0.00
      : STOKES        1    1.0000000E+00         1.00   1.0000000E+00     0.00
      : ----------------------------------------------------------------
AIPS 1: Coordinate equinox 2000.00
AIPS 1:
AIPS 1: Conv size= 0.13732 X 0.06835   Position angle= -22.69
AIPS 1: Rest freq      0.000            Vel type: OPTICAL  wrt LSR
AIPS 1: Alt ref. value -4.20762E+05  wrt pixel
AIPS 1: Maximum version number of extension files of type CC is   1
AIPS 1: Maximum version number of extension files of type HI is   1
AIPS 1: Keyword = 'CCFLUX  '  value =  4.341595E-02
AIPS 1: Keyword = 'CCTOTAL '  value =  4.341595E-02
AIPS 1: Keyword = 'PARANGLE'  value = -1.239448E+02
AIPS 1: Keyword = 'ZENANGLE'  value =  6.472005E+00
>tvlo;tvzoom;tvps
```

Axes: Pos, Pos, Hz, 1 = I = total intensity

Restoring beam Maj, Min (arcsec), position angle (degrees)

Extension tables

- Standard astronomical data format:
  - See Greisen, Calabretta & Valdez or FITS web home
  - UVFITS or IDE FITS for visibility data
  - Image files for 1, 2, 3+ D images
    - Unfortunately several dialects
  - AIPS uses FITS
  - CASA can read/export FITS
- Structure of FITS file
  - Header
  - (Binary) data
  - Extension tables

FITS The Astronomical Image and Table Format

FITS Header

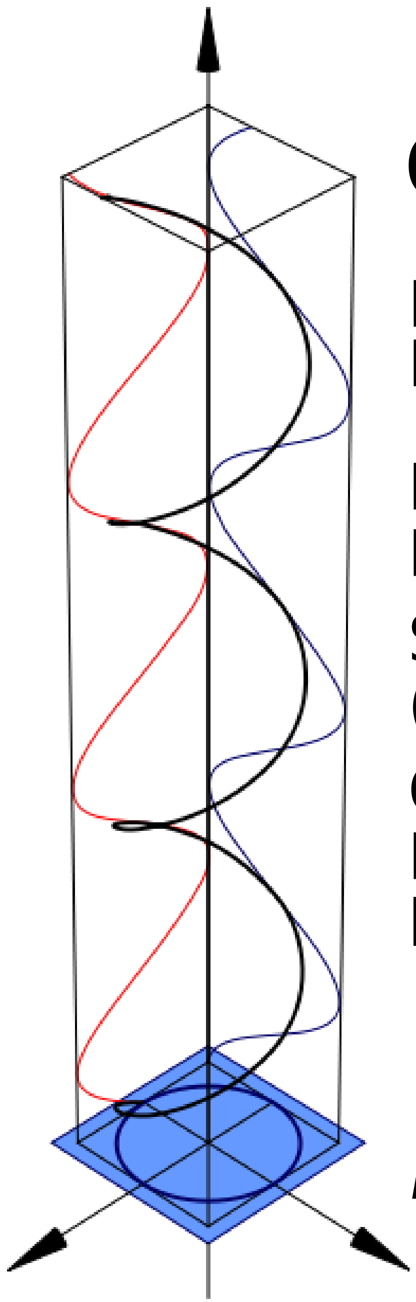
xterm

```
SIMPLE  =                    T /
BITPIX  =                  -32 /
NAXIS   =                    4 /
NAXIS1  =                   66 /
NAXIS2  =                   66 /
NAXIS3  =                  280 /
NAXIS4  =                    1 /
EXTEND  =                    T /Tables following main image
BLOCKED =                    T /Tape may be blocked
OBJECT  = 'SPER    '           /Source name
TELESCOP= 'MERLIN2 '           /
INSTRUME= '        '           /
OBSERVER= '        '           /
DATE-OBS= '1999-05-25'         /Obs start date YYYY-MM-DD
DATE-MAP= '2000-01-11'         /Last processing date YYYY-MM-D
BSCALE  =    1.00000000000E+00 /REAL = TAPE * BSCALE + BZERO
BZERO   =    0.00000000000E+00 /
BUNIT   = 'JY/BEAM '           /Units of flux
EPOCH   =      1.950000000E+03 /Epoch of RA DEC
VELREF  =                  257 />256 RADIO, 1 LSR 2 HEL 3 OBS
ALTRVAL =    1.66710997656E+09 /Altenate FREQ/VEL ref value
ALTRPIX =     -1.390000000E+02 /Altenate FREQ/VEL ref pixel
OBSRA   =    3.48128515485E+01 /Antenna pointing RA
OBSDEC  =    5.83592651738E+01 /Antenna pointing DEC
RESTFREQ=    1.66735906400E+09 /Rest frequency
DATAMAX =      5.355936050E+00 /Maximum pixel value
DATAMIN =     -5.429587513E-02 /Minimum pixel value
```

```
CTYPE1  = 'RA---SIN'           /
CRVAL1  =    3.48128515485E+01 /
CDELT1  =     -1.111111123E-05 /
CRPIX1  =      3.300000000E+01 /
CROTA1  =      0.000000000E+00 /
CTYPE2  = 'DEC--SIN'           /
CRVAL2  =    5.83592651738E+01 /
CDELT2  =      1.111111123E-05 /
CRPIX2  =      3.400000000E+01 /
CROTA2  =      0.000000000E+00 /
CTYPE3  = 'VELO-LSR'           /
CRVAL3  =    6.28035946778E+03 /
CDELT3  =     -1.756092529E+02 /
CRPIX3  =     -1.390000000E+02 /
CROTA3  =      0.000000000E+00 /
CTYPE4  = 'STOKES  '           /
CRVAL4  =    1.00000000000E+00 /
CDELT4  =      1.000000000E+00 /
CRPIX4  =      1.000000000E+00 /
CROTA4  =      0.000000000E+00 /
ISTORY AIPS HEADER2  WTNOISE  =  1.035
-More--(0%)
```

- Fortunately there are tools
  - IMHEAD in AIPS or CASA

Polarization jargon



CIRCULAR

Left-hand
LHC, L, LL

Right-hand
RHC, R, RR

Stokes V =
(RR-LL)/2

Cross hands
LR RL make
linear

LINEAR

Stokes Q = (RL + LR)/2

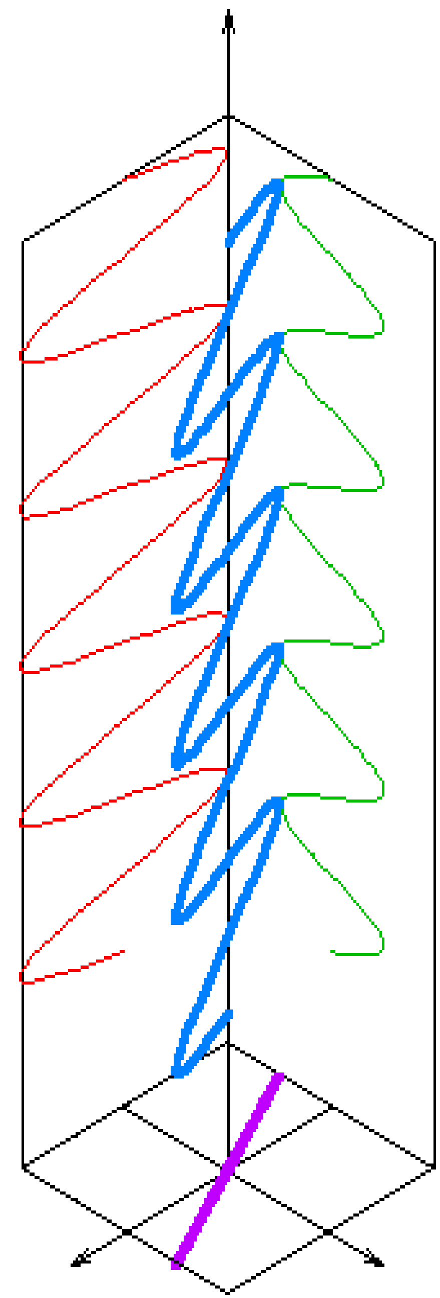
Stokes U = (RL - LR)/2i

Polarized intensity
$P = \sqrt{(Q^2 + U^2 + V^2)}$

Polarization
angle χ = ½ atan2(U/Q)

Linear feeds X,XX, Y,YY

Cross hands XY YX
make circular



Stokes I = (LL + RR)/2 = (XX + YY)/2

beware, some packages' definitions differ by x2

Diagrams thanks to Wikipaedia

FITS axes labels

- Axes contain one+ pixels
- Quantization of physical variable e.g.
  - Position in RA
  - Frequency
  - Label
    - Types of polarization ⇒
      - I (one 'pixel')
      - IQUV (four 'pixels')

CASA

- Polarizations also termed correlations

Polarization type	Label	FITS code
Total	I	1
Linear	Q	2
Linear	U	3
Circular	V	4
Circular	RR	-1
Circular	LL	-2
Linear	RL	-3
Linear	LR	-4
Linear	XX	-5
Linear	YY	-6
Circular	XY	-7
Circular	YX	-8
Undef	UNDEF	---
Linear	POLI	5
Linear	POLA	6

CASA developed to meet NG needs

- `aips++` development in c++ started in ~1994
  - Easier to maintain/develop/parallelise
- User-friendly python wrapper since 2007
  - Common Astronomy Software Application
  - 'Task' interface or scripting
  - Underlying aips++ toolkit available
- Measurement Set data format
  - *uv* data and images in subdirectories
    - In working directory or wherever you want
- Prime motivation (& funding) for ALMA and EVLA
  - ALBiUS (RadioNet) for interoperability with AIPS
- ***Easy to install***

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

Starting point

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

Goal

$\underline{I}$ mage

$\underline{A}$ dditive baseline error

Scalars

Methods

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

l,m image plane coords

u,v Fourier plane coords

i,j telescope pair

Jones Matrices

Hazards

Multiplicative baseline error

Bandpass response

Generalised electronic gain

Dterm (pol. leakage)

E (antenna voltage pattern)

Parallactic angle

Tropospheric effects

Faraday rotation

Using the Measurement Equation

- *Hamaker, Bregman & Sault 1996*
  - Decompose into relevant calibration components e.g.
- $\underline{V}_{ij}^{obs} = \mathbf{B}_{ij}\mathbf{G}_{ij}\mathbf{D}_{ij}\mathbf{P}_{ij}\mathbf{T}_{ij}\mathbf{F}_{ij}\underline{V}_{ij}^{ideal}$
  - Chose one (or a few) at a time
    - Usually solve fastest-varying first
      - (so averaging over slower-varying)
    - Might have to iterate
  - Linearise and solve by χ^2 (or other) minimization
  - (Same principles as AIPS etc. gain calibration)
- Visibility data are stored in Measurement Sets
  - Accessible directories of tables

CASA

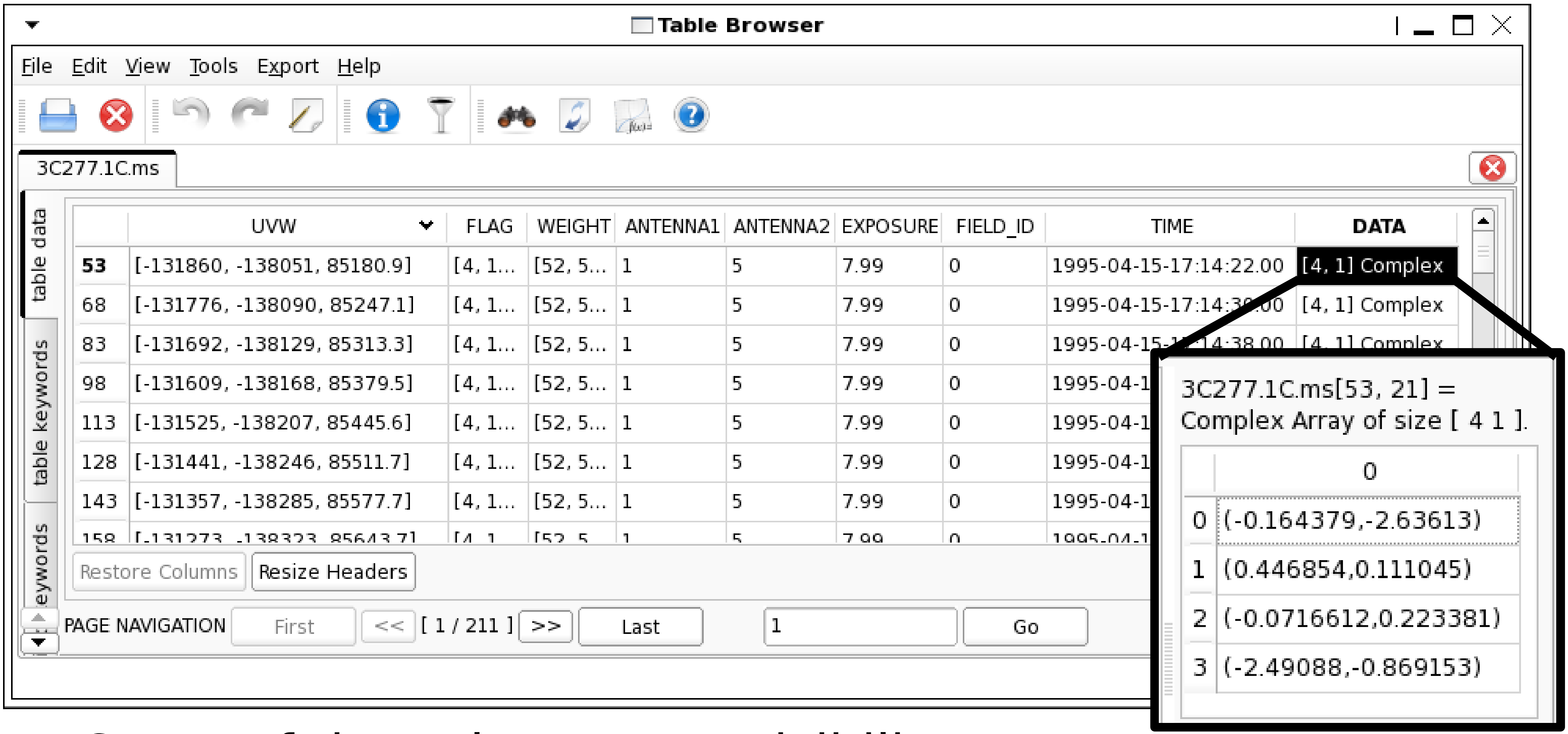
Measurement Set visibility data

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

```
> tree jupiterallcal.split.ms
```

```
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
    - Inspect in CASA `browsetable` (rarely necessary)

Visibility data: Measurement Set format

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

- Unix-like directory structure with binary data and ascii metadata files arranged in subdirectories
- Additional tables in MS and free-standing:
  - *Admin*: Antenna, Source etc.
  - *Processing*: calibration, flags, etc.
- ~interconvertible with FITS; similar image format

Science data model format

```
uid___A002_X10e085_X148
|-- ASDM.xml
|-- ASDMBinary
|   |-- uid___A002_X10e085_X149
|   |-- uid___A002_X10e085_X14a
|   |-- uid___A002_X10e085_X14b
.
.
|   |-- uid___A002_X10e085_X1e9
|   `-- uid___A002_X10e085_X1ea
|-- AlmaRadiometer.xml
|-- Antenna.xml
|-- CalAmpli.xml
|-- CalBandpass.xml
|-- CalData.xml
|-- CalDevice.xml
|-- CalPhase.xml
|-- CalReduction.xml
|-- CalWVR.xml
|-- ConfigDescription.x
|-- CorrelatorMode.xml
```

- Native format for ALMA, EVLA etc.
  - Compact, static binary data
  - Accessible xml metadata
    - CASA converts to MS

```
|-- DataDescription.xml
|-- ExecBlock.xml
|-- Feed.xml
|-- Field.xml
|-- Focus.xml
|-- FocusModel.xml
|-- Main.xml
|-- Pointing.bin
|-- Pointing.xml
|-- PointingModel.x
|-- Polarization.xm
|-- Processor.xml
```

```
|-- Receiver.xml
|-- SBSummary.xml
|-- Scan.xml
|-- Source.xml
|-- SpectralWindow.xml
|-- State.xml
|-- Station.xml
|-- Subscan.xml
|-- SwitchCycle.xml
`-- Weather.xml

1 directory, 196 files
```

Starting CASA

- See web links for downloads (or http://casa.nrao.edu)
  - Don't forget the Cookbook!
- Start by typing **casapy** (or set up your choice)
  - 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**

CASA

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'     #  Nome 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

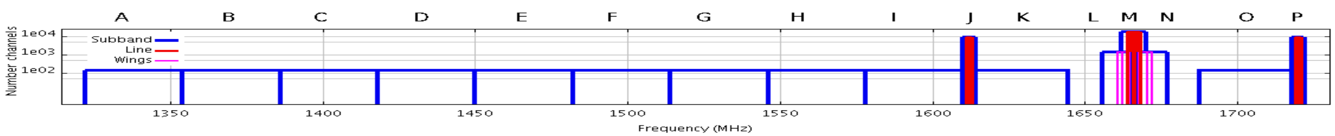
- Simplest input to tasks is `param=value`
  - In this mode, variables are global
    - `solint='1min'` will appear in all tasks until reset
  - `default(gaincal)` resets default values
  - `tget gaincal` restores last *successful* execution
  - `saveinputs(gaincal,'gctry1')` saves inputs at any stage
  - `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
- Export data as FITS files
  - Apply all calibration/flagging first

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 *your variables*
  - e.g. `noise_target = imstat()`
    - Task `imstat` gives python array of image measurements
    - `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

AIPS or CASA? (either? both?)

- Either package for straightforward data
  - Might need 'native' package in early stages
- Raw data format:
  - CASA for SDM, MS, UVFITS, IDI FITS
    - Good for combining different data shapes



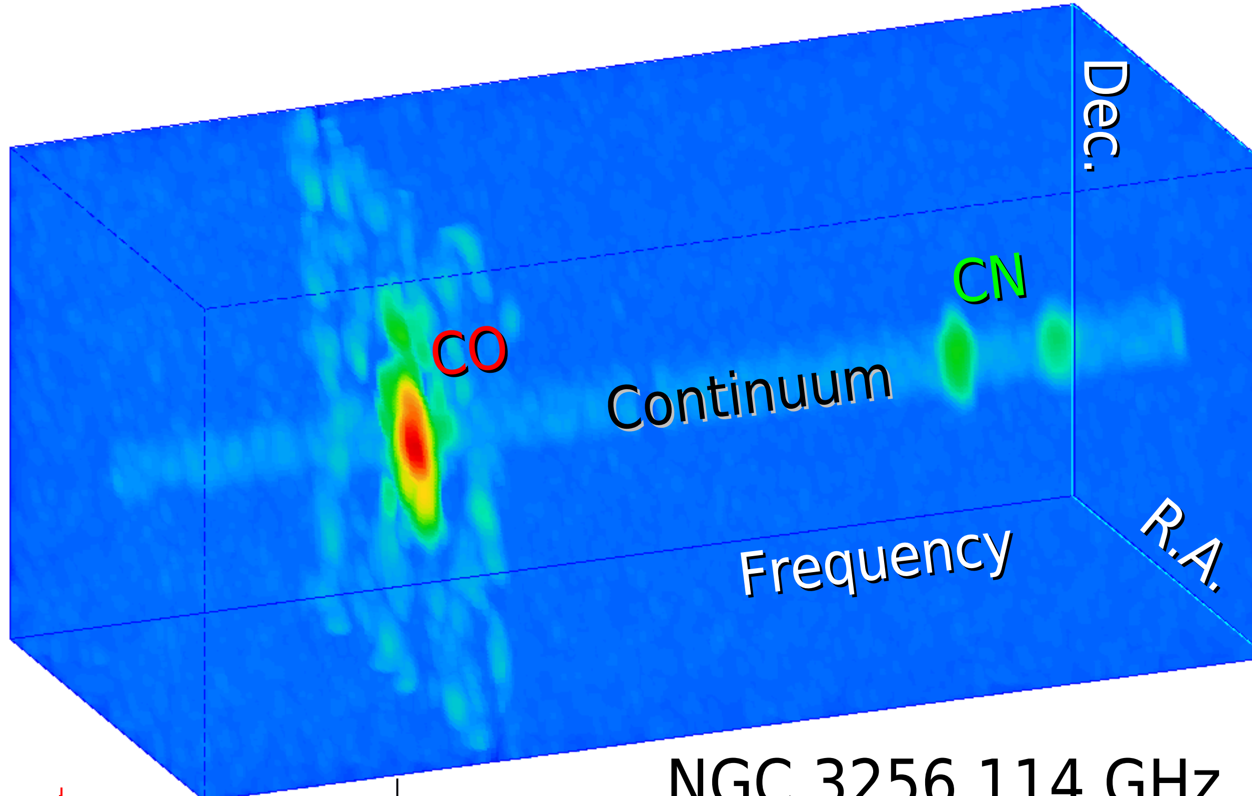
  - AIPS also for FITS (but harder for linearly pol. feeds)
- Calibration
  - CASA especially for EVLA and ALMA
    - Apply Water Vapour Radiometry etc.corrections
    - Flexible bandpass and polarization calibration
      - Simple delay corrections
  - AIPS for combined delay and rate calibration
    - ALMA task being developed

AIPS or CASA? (or either or both)

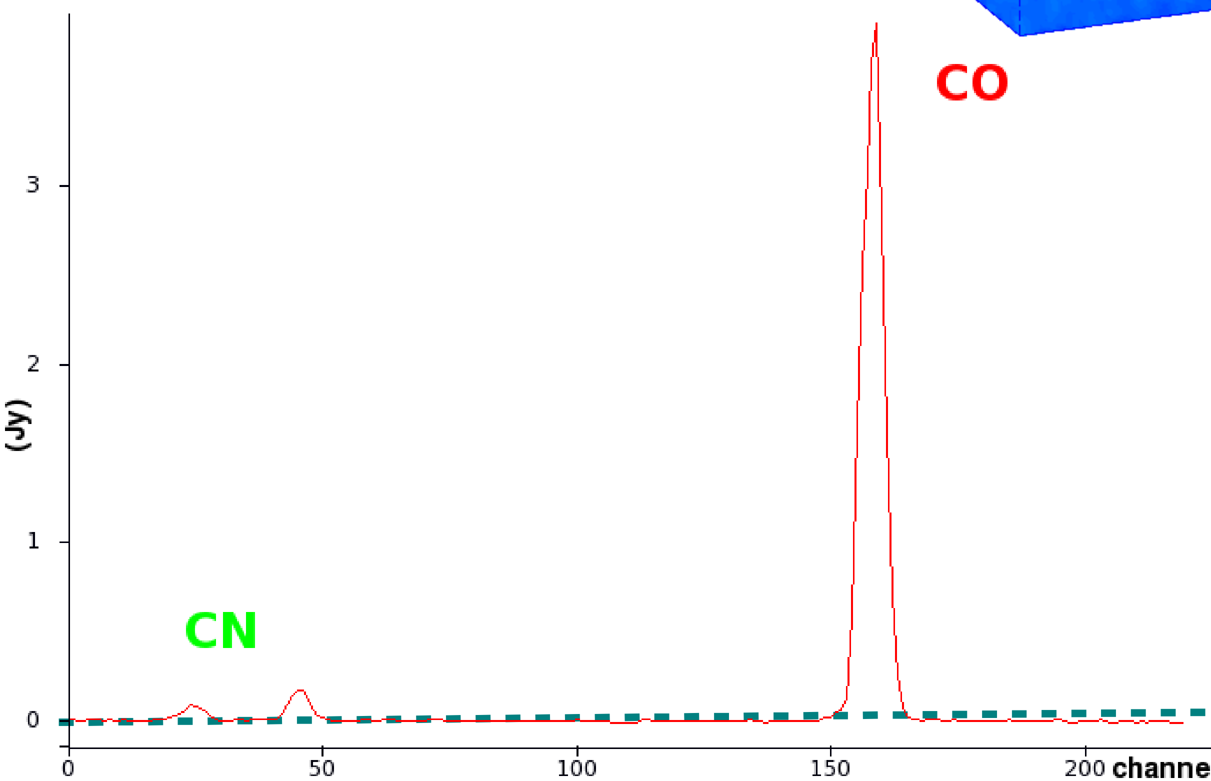
- Imaging
  - CASA
    - Wide-band MFS with spectral index/curvature
    - *w*-projection and/or faceting (3-D sky)
    - Multi-scale clean
    - Heterogenous primary beams
    - Mosaicing
  - AIPS
    - Faster, especially sparse multi-facet wide-field images
    - Maximum entropy methods
    - At present, more measurement/analysis tools
- Interoperability
  - Script both in Python
  - Easy to swap data but apply calibration/flags first
    - Most extension tables lost
    - May need to re-calculate/apply weights

... and more

- 'Small' field of view (single star, galaxy...)
  - Easy to identify line-free channels

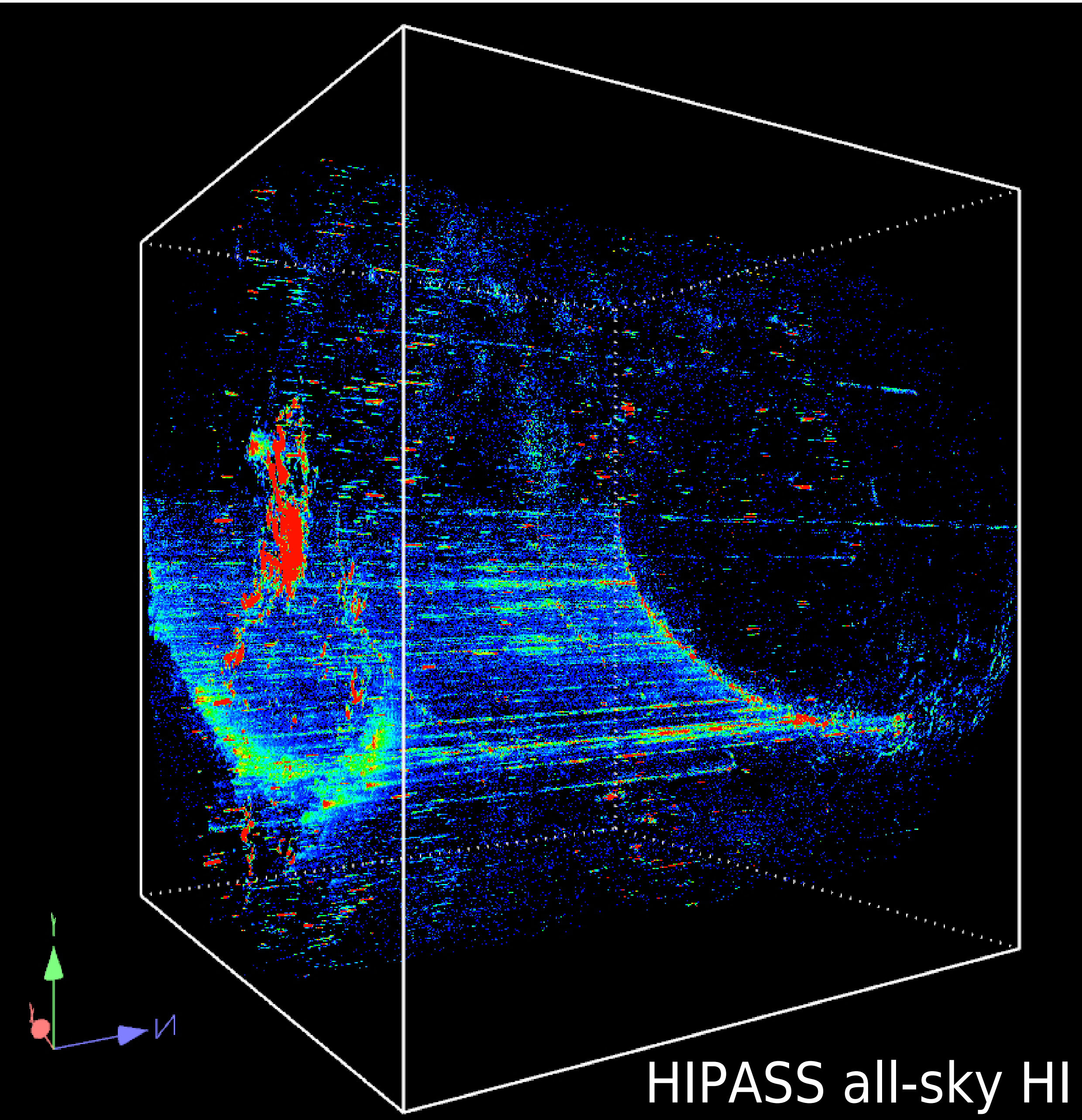


NGC 3256 114 GHz



- Subtract continuum in *uv* plane
  - CASA or AIPS
    - CASA better for wide fields

Groeningen Image Processing SYstem

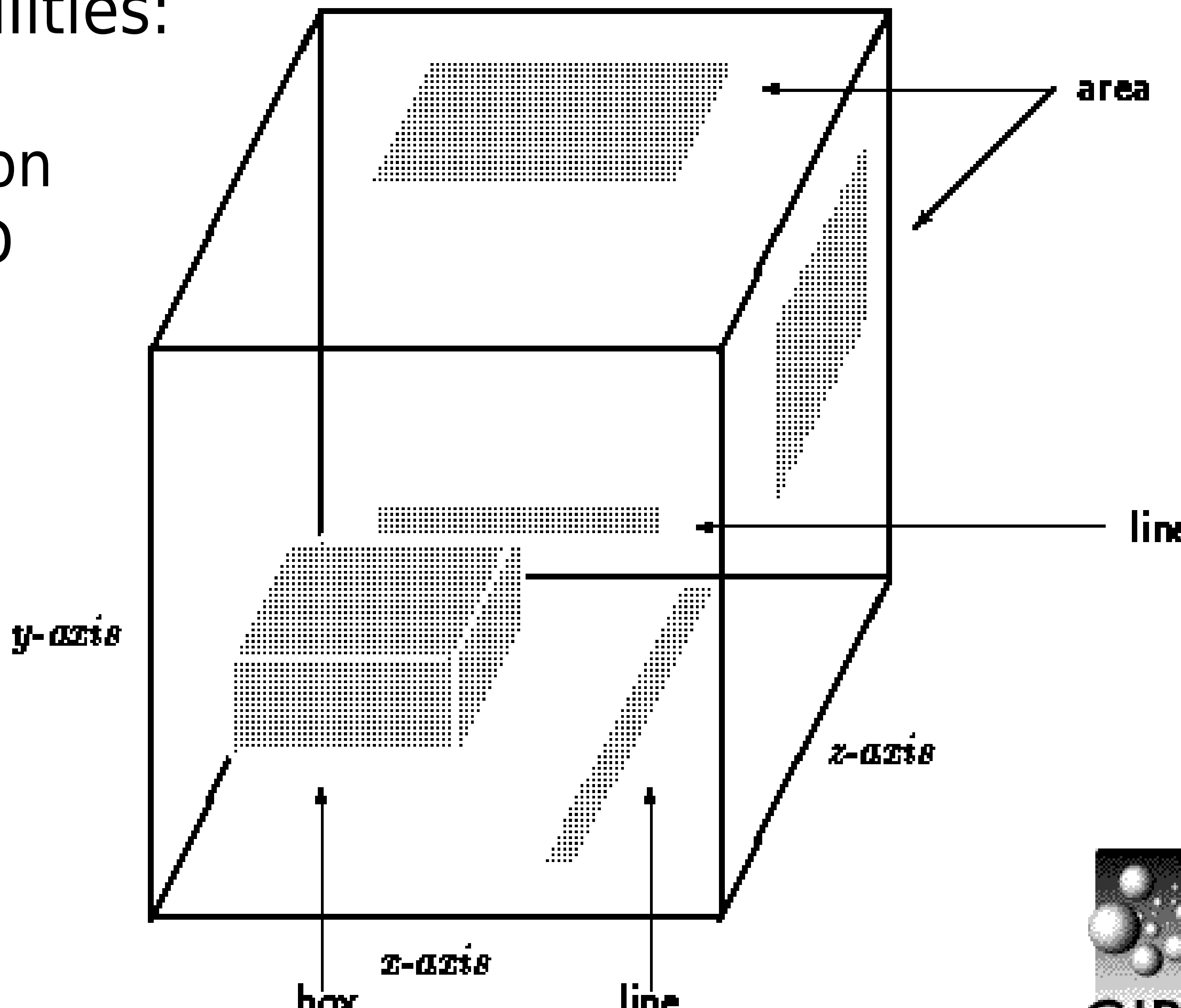


- Data cubes with many 3-D sources
  - No spectral channel is free of a line somewhere in the spatial field of view
- Have to make 'dirty' image cube first
  - Select sub-cubes
  - Subtract continuum in image plane
  - Deconvolve beam
- *Possible* in AIPS/CASA
- **GIPSY** *specialised* for this and further analysis

GIPSY

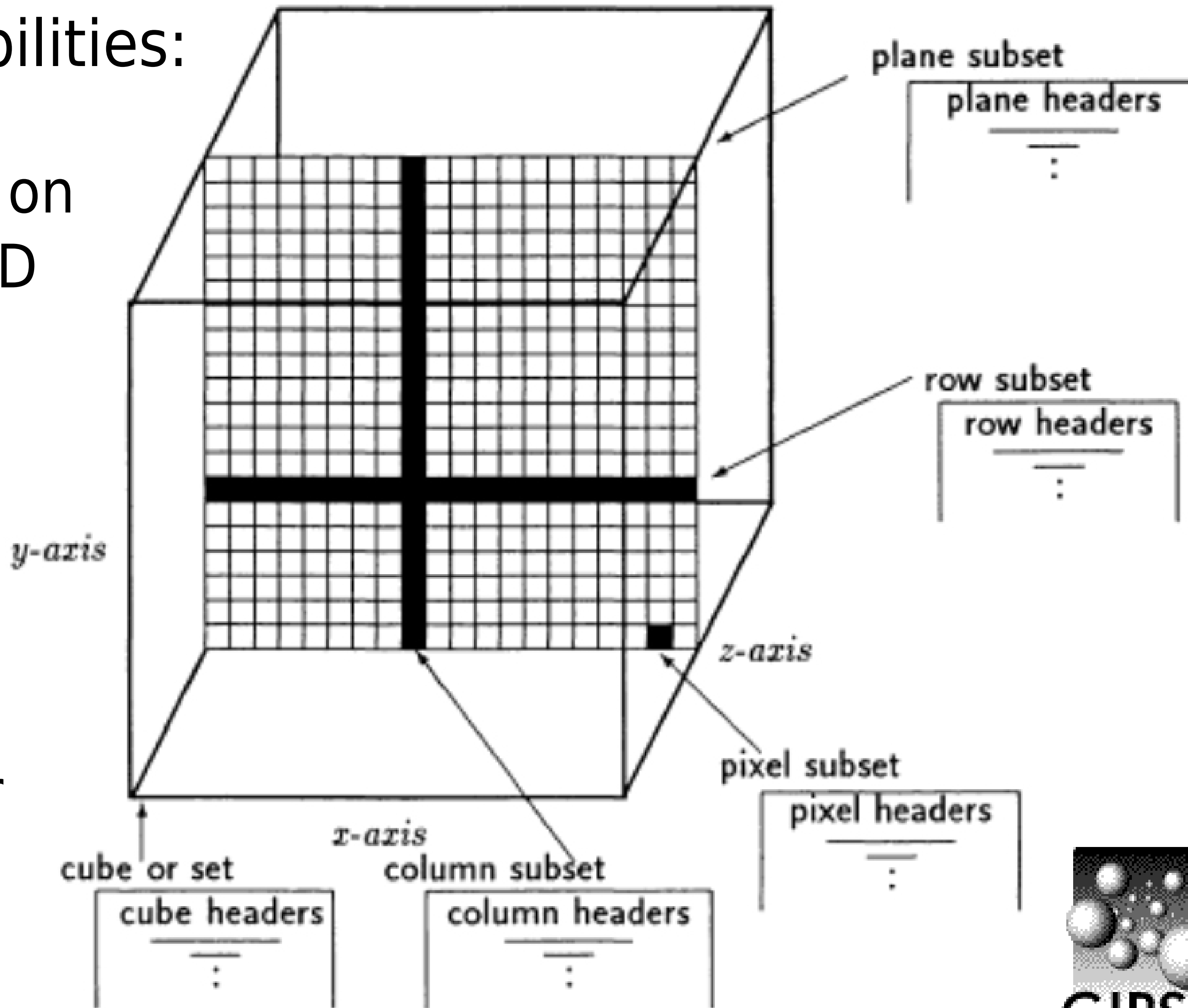
GIPSY overview

- (see website and M Verheyen, Fri am for details)
- Unique capabilities:
  - Operations on 1-, 2- or 3-D subsets (frames) without splitting



GIPSY overview

- (see M Verheyen, Friday am for details)
- Unique capabilities:
  - Operations on 1-, 2- or 3-D subsets (frames) without splitting
  - Heirachical headers for easy selection



GIPSY

What you can use GIPSY for

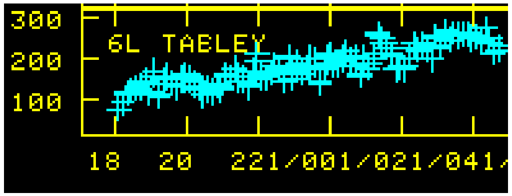
- Operates on image-plane data
  - Apply all calibration, FT (clean) in favourite package
- Input FITS cubes, preferably:
  - 32-bit
  - SIN or NCP projection
  - Frequency units on spectral axis
    - but conversions are possible
- Continuum subtraction
  - Start from dirty image and beam cubes
    - Input must be $2x2^n$ pixels/side for output 2^n pixels/side
  - Select, subtract, clean
- Spectral analysis of subtracted, cleaned cubes
  - Fit Gaussian profiles, ellipses etc.
  - Make moments
  - Compare models with data   etc. etc.

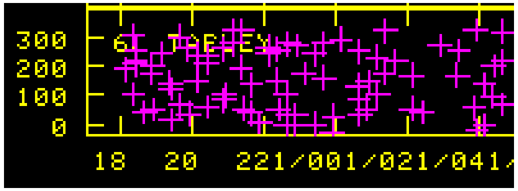
Keep sight of the physics

- Brain gets filled with package jargon
  - `task 'CALIB'; calsour 'phaseref'; solint 0.5; docal 100; aparm(7) 3; gainuse 5; solmo 'p'`
- Rember this means
  - Take the visibility data for the phase ref and apply existing calibration table 5; minimum snr 3
  - If no other model is given, a point source at the field centre will be used
  - Compare the data with the model phase and calculate the corrections needed
- That way you will know to expect
  - and what to check if you get

Keep sight of the physics

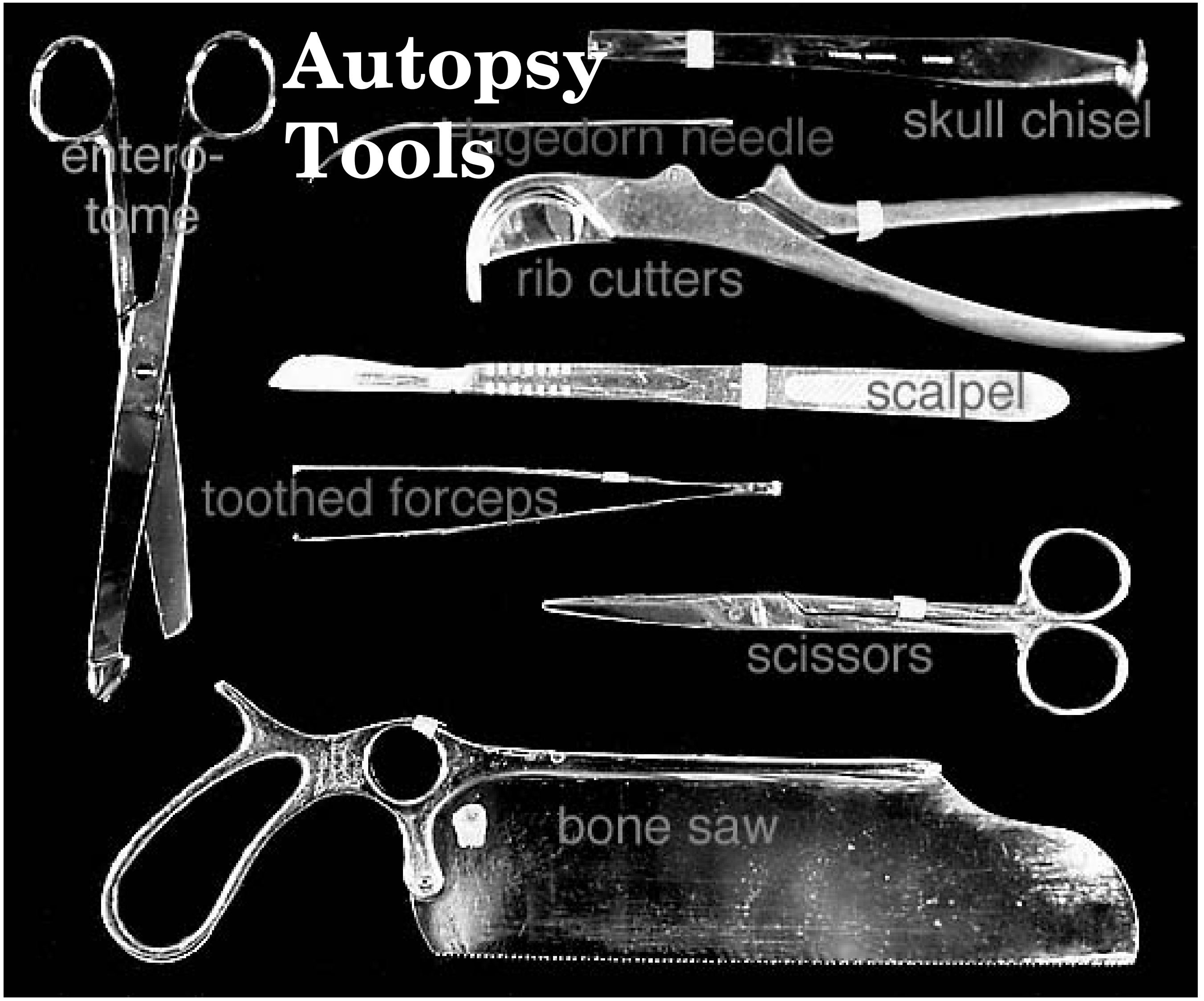
- Brain gets filled with package jargon
  - `task 'CALIB'; calsour 'phaseref'; solint 0.5; docal 100; aparm(7) 3; gainuse 5; solmo 'p'`
- Rember this means
  - Take the visibility data for the phase ref and apply existing calibration table 5; minimum snr 3
  - If no other model is given, a point source at the field centre will be used
  - Compare the data with the model phase and calculate the corrections needed
- That way you will know to expect
  - and what to check if you get





Keep a full processing history

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



An experienced radio astronomer