

LOFAR synthesis data handling Miscellaneous Tools

Ger van Diepen
ASTRON

Table Browser

GER.MS

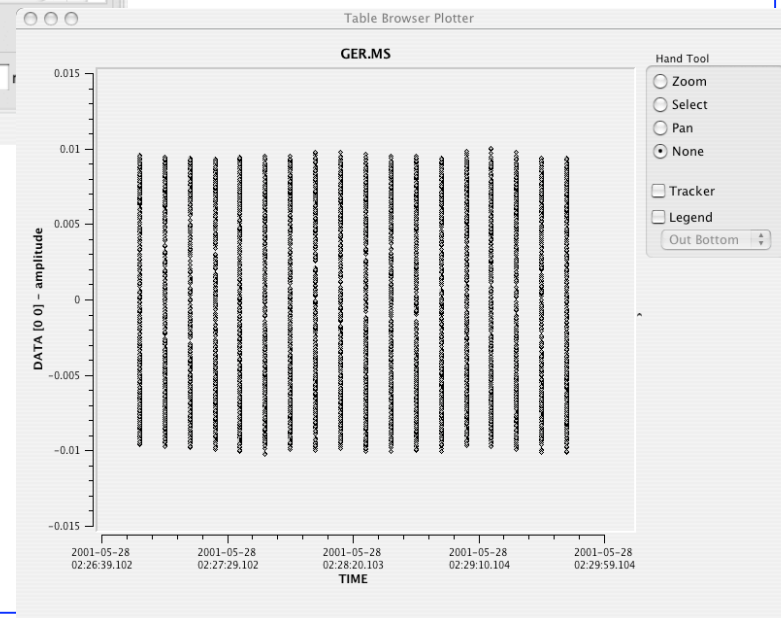
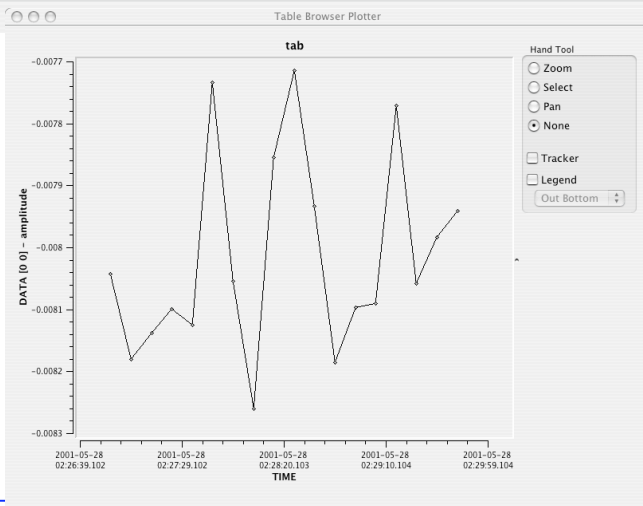
	UVW	ANTENNA1	ANTENNA2	ARRAY_ID	PROCESSOR_ID	EXPOSURE	FEED1	FEED2	FIELD_ID	FLAG_ROW	INTERVAL
0	[183.035, 1386.46, -2337.42]	0	13	0	-1	9.9549184...	0	0	0	0	10
1	[176.584, 1337.6, -2255.05]	0	12	0	-1	9.9549184...	0	0	0	0	10
2	[172.549, 1312.07, -2214.57]	1	13	0	-1	9.9549184...	0	0	0	0	10
3	[166.128, 1263.25, -2132.17]	1	12	0	-1	9.9549184...	0	0	0	0	10
4	[164.871, 1237.6, -2091.55]	2	13	0	-1	9.9549184...	0	0	0	0	10
5	[158.372, 1188.83, -2009.12]	2	12	0	-1	9.9549184...	0	0	0	0	10
6	[153.45, 1164.71, -1967.88]	3	13	0	-1	9.9549184...	0	0	0	0	10
7	[147.022, 1115.93, -1885.46]	3	12	0	-1	9.9549184...	0	0	0	0	10
8	[145.271, 1092.64, -1843.47]	4	13	0	-1	9.9549184...	0	0	0	0	10
9	[138.778, 1043.81, -1761.08]	4	12	0	-1	9.9549184...	0	0	0	0	10
10	[132.635, 1018.26, -1720.78]	5	13	0	-1	9.9549184...	0	0	0	0	10
11	[126.28, 969.486, -1638.35]	5	12	0	-1	9.9549184...	0	0	0	0	10
12	[127.222, 944.644, -1597.08]	6	13	0	-1	9.9549184...	0	0	0	0	10
13	[120.655, 895.891, -1514.65]	6	12	0	-1	9.9549184...	0	0	0	0	10

Restore Columns Resize Headers

PAGE NAVIGATION First << [1 / 13] >> Last 1 Go Loading 1000

```
python
from pyrap.tables import *
t = table('GER.MS')
t1 = t.query('DATA_DESC_ID=0')
t1.browse()
```

casabrowser ('GER.MS')



casaviewer (ms)



```
python
```

```
from pyrap.tables import *  
t = table('GER.MS')  
t1 = t.query('DATA_DESC_ID=0')  
t1.view()
```

```
casaviewer ('GER.MS')
```

Sorting... Done.

/Users/diepen/3C343.MS

Selected MS: Time slots: 1437 Baselines (incl. gaps): 15
Correlations: 4 Channels: 64 Spectral Windows: 1

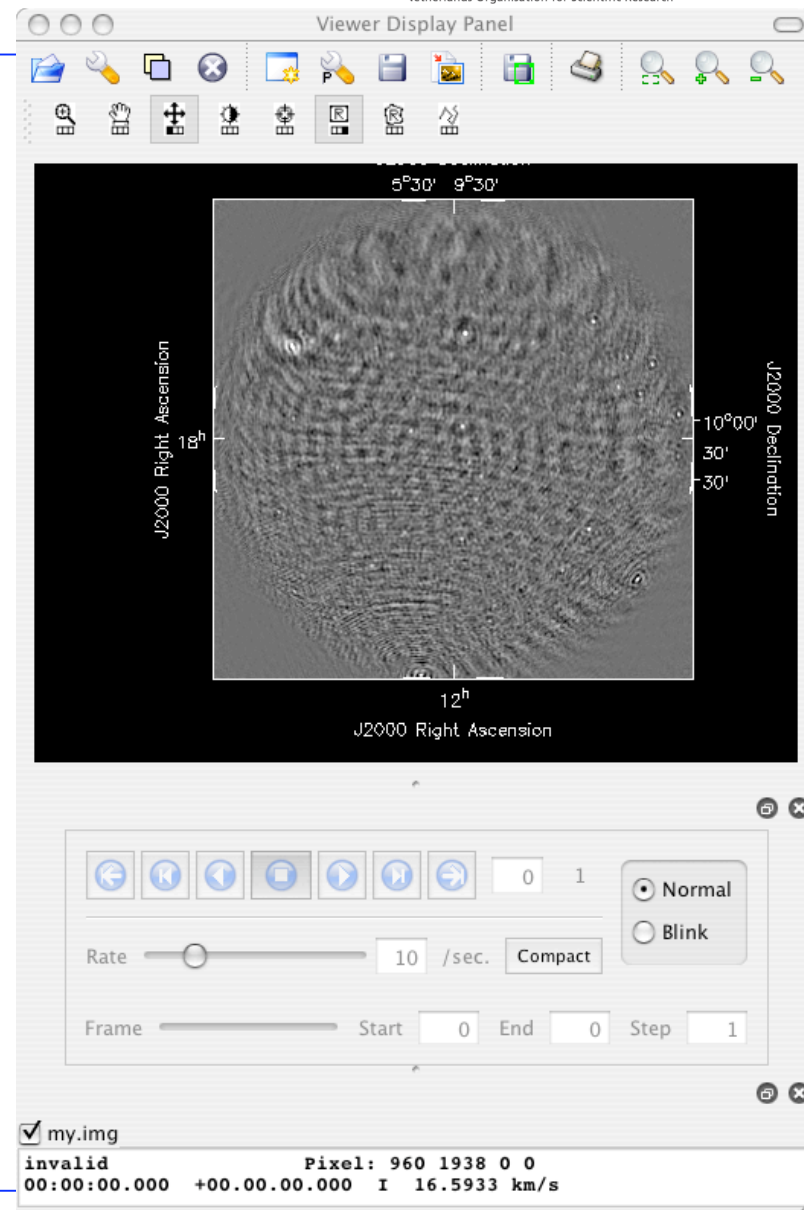
Loading MS vis. data: 28% 47% 80% 89% Done.

Resorting MS vis. data: 38% 95% Done.

casaviewer (image)

```
python
from pyrap.images import *
im = image('my.img')
im.view()
```

```
casaviewer ('my.img')
```



msselect

- select baselines from an MS using CASA's syntax

msoverview in=`msname` [`verbose=T`]

- summary of a MeasurementSet

showtablelock my.ms

- show which process holds a lock on table my.ms

makems

makebeamtables

- make a MeasurementSet with `data=0` (for simulations)
 - See <http://www.lofar.org/wiki/lib/exe/fetch.php?media=software:makems.pdf>
- Attach the beam info tables (needed for BBS)
 - See <http://www.lofar.org/wiki/doku.php?id=engineering:software:tools:makebeamtables>

lwimager

- Make an image from an MS (using CASA classes)
 - See `lwimager -h`
 - `operation=empty` makes a zero image with correct coordinates

cexecms

- Execute a command on all subbands of an observation

See also <http://www.lofar.org/operations/doku.php?id=engineering:software:tools>

Select baselines from an MS

- uses CASA-style baseline selection
- can make shallow or deep copy
- See <http://www.lofar.org/wiki/lib/exe/fetch.php?media=software:msselection.pdf>

```
msselect in=in.ms out=out.ms baselines='CS*'
```

creates out.ms (as RefTable) containing all baselines with core stations.

```
msselect in=in.ms out=out.ms baselines='CS* &'
```

creates out.ms (as RefTable) containing core-core baselines only.

- Uses cexec to execute a command for matching file names on nodes in a cluster
- Replaces placeholders like <FN> with actual file name
- Deduces cluster name from head node (Ice: or locus:).
 - Use -c to override like
 - c 'locus: 0-5'
- Execute without arguments to get help info

```
cexecms "ls -d <FN>" "/data/scratch/pipeline/L23276/L23276*"
```

finds all subbands for the given observation (using a wildcarded file name).

Note: quotes have to be used around command and file name pattern.

```
cexecms "taql 'update <FN>/ANTENNA set MOUNT=\"X-Y\"'"
"/data/scratch/pipeline/L2011_23138/*.dppp"
```

fixes the MOUNT column in the gives MSs.

Note: 3 levels of quotes are needed (X-Y, taql command, and cexecms command).

This command only works for bash. In (t)csh the inner quotes have to be escaped differently like:

```
cexecms "taql 'update <FN>/ANTENNA set MOUNT=\"\"X-Y\"\"'"
"/data/scratch/pipeline/L2011_23138/*.dppp"
```

- Use -s option to execute a script
 - can be python script, parset file, ...

For example:

count flagged data in all (raw) MSs of an observation

```
use LofIm
```

```
cexecms -s $HOME/nd.pset NDPPP '/data/L42791/*.MS' > count.log
```

with nd.pset containing:

```
msin=<FN>
```

```
msout=
```

```
steps=[count]
```