

IRAM Memo 2016-?
MRTCAL User Manual

S. Bardeau[1], J. Pety[1,2], A. Sievers[3]

1. IRAM (Grenoble)
2. Observatoire de Paris
3. IRAM (Granada)

July, 06^{th} 2017
Version 0.5

Abstract

Some abstract here.
Related documents: **MRTCAL** Programmer Manual, **CLASS** Associated Arrays.

Contents

1 Quick start guide **4**
1.1 Overview . . . 4
1.2 A simple session (at home) . . . 4
1.3 A simple session (at Pico Veleta) . . . 4

2 Indexing **5**
2.1 Indexing and listing a directory . . . 5
2.2 Entry numbering and versioning . . . 5
2.2.1 Numbering . . . 5
2.2.2 Versioning . . . 6

3 Calibrating **7**
3.1 Processing all a directory . . . 7
3.2 Processing a single IMB-FITS file . . . 7
3.3 Calibration status . . . 8
3.4 Calibration products . . . 8
3.5 Weight array . . . 11

4 Exploring IMB-FITS files (no calibration) **12**
4.1 FITS reader . . . 12
4.2 FITS plots . . . 12
4.2.1 Calibration loads . . . 12
4.2.2 Display MJD values and tables . . . 14
4.2.3 Display antenna position . . . 15

5 MRTCAL Language Internal Help **18**
5.1 Language . . . 18
5.2 CALIBRATE . . . 18
5.3 INDEX . . . 18
5.3.1 INDEX UPDATE . . . 19
5.3.2 INDEX OPEN . . . 19
5.3.3 INDEX OUTPUT . . . 19
5.3.4 INDEX WATCH . . . 20
5.3.5 INDEX CONVERT . . . 20
5.3.6 INDEX /PATTERN . . . 20
5.3.7 INDEX /DATE . . . 20
5.4 MCOPY . . . 21
5.5 MDUMP . . . 21
5.5.1 MDUMP /OUTPUT . . . 22
5.6 MFIND . . . 22
5.6.1 MFIND /SCAN . . . 23
5.6.2 MFIND /DATE . . . 23
5.6.3 MFIND /FRONTEND . . . 23
5.7 MLIST . . . 24
5.7.1 MLIST /TOC . . . 24
5.7.2 MLIST /COLUMNS . . . 25
5.8 MSETUP . . . 25
5.8.1 MSETUP INPUT . . . 26
5.8.2 MSETUP BOOKKEEPING . . . 27
5.8.3 MSETUP CALIBRATION . . . 27
5.8.4 MSETUP SOLVE . . . 29

5.8.5 MSETUP OUTPUT 30
5.8.6 MSETUP PIPELINE 31
5.8.7 MSETUP DEBUG 31
5.9 MUPDATE 31
5.10 MWRITE 32
5.11 PIPELINE 32
5.12 READ 32
5.13 SOLVE 33
5.14 VARIABLE 33
5.14.1 VARIABLE /INDEX 34

1 Quick start guide

1.1 Overview

MRTCAL offers 2 main tasks:

- indexing the IMB-FITS files,
- calibrating the IMB-FITS files.

Indexing is done by creating a database of the IMB-FITS. This offers several advantages (*e.g.* opening the IMB-FITS - which is usually slow - is factorized once at indexing time) and flexibility (*e.g.* the user can select the desired subset of IMB-FITS files). The command to be used for this task is `INDEX` (see `HELP` for details), and the database is usually named *index.mrt* and saved together with the IMB-FITS files.

Calibration of the IMB-FITS files has 2 major modes: per file (command `CALIBRATE`), and for Current indeX (command `PIPELINE`). The pipeline mode is an automated way to calibrate each the files in the Current indeX, selecting the calibration scans, etc.

In addition, **MRTCAL** offers a collection of tuning options with the command `MSETUP`. For a first approach, you may be interested in the groups `MSETUP CALIBRATION` (*i.e.* tune the way the calibration is performed) and `MSETUP OUTPUT` (*i.e.* what results and how they are saved in the **CLASS** file). See dedicated HELP for exhaustive description.

1.2 A simple session (at home)

Assuming the IMB-FITS files are in a local folder with no particular restriction, the simplest **MRTCAL** session which you may think of is the following one:

```
MRTCAL> index build      ! Build the index of IMB-FITS files in the current directory
MRTCAL> mfind            ! Select all the files for calibration
MRTCAL> mlist            ! Optionally list the files to be calibrated
MRTCAL> file out myclassfile.30m single  ! Open a new Class output file
MRTCAL> pipeline         ! Calibrate all the files
```

After the processing, you may want to list again the IMB-FITS files, with their updated calibration status:

```
MRTCAL> mfind            ! Update the list of files
MRTCAL> mlist            ! List the files and their calibration status
```

After the `PIPELINE` step, the **CLASS** file `myclassfile.30m` contains all the *science* spectra (with the scientific target as source name), but also *calibration* spectra (with the source name `CALSKY`). See subsection 3.4 for details. The `PIPELINE` command has run using the current defaults, but there are a number of tunings which can be modified. See `HELP MSET` for details, and type `MSET` to see their current values.

1.3 A simple session (at Pico Veleta)

If you want to recalibrate the IMB-FITS files in your project account at Pico Veleta (i.e. located in `~/observationData/imbfits/`), you must consider that you have no write access to this directory. In this case, you can not put the `index.mrt` file together with the IMB-FITS files. The solution is to create it *elsewhere* (it will be your responsibility to remember the association between the index file and the directory where the IMB-FITS files can be found):

```
! Go to working local directory (with all permissions):
MRTCAL> sic directory some/working/dir/

! Build the index (named myindex.mrt in current working directory)
! for IMB-FITS files found in ~/observationData/imbfits/
MRTCAL> index build ~/observationData/imbfits/ /file myindex.mrt
```

Then the subsequent commands are identical to the previous section. If you need to update or reopen the index file without rebuilding it from scratch, the corresponding commands are respectively:

```
MRTCAL> index update  ~/observationData/imbfits/ /file myindex.mrt

MRTCAL> index open  ~/observationData/imbfits/ /file myindex.mrt
```

2 Indexing

2.1 Indexing and listing a directory

The easiest indexing method is the following one:

```
MRTCAL> index build  ! Build the index of IMB-FITS files in the current directory
```

The `INDEX` command offers many possibilities, such as recursive indexing, restricting to some dates or patterns, indexing read-only directories, waiting for new files to appear... See `HELP INDEX` for details.

Once the index is built, you can access its contents with *e.g.*

```
MRTCAL> index open                  ! (Re)open the index file
MRTCAL> mfind                       ! Select all the entries in the index
MRTCAL> mlist                       ! Sequential list
MRTCAL> mlist /toc                  ! Summarized list
MRTCAL> mlist /column number projid ! Custom list
```

`MFIND` and `MLIST` offer a variety of options so that you can check IMB-FITS contents and setup a list of scans to be calibrated.

2.2 Entry numbering and versioning

2.2.1 Numbering

When **MRTCAL** opens an index file (usually index.mrt), it builds what is called the *input index* in memory. However, for many reasons the contents of the index file may be disordered. In order to give friendly lists to the user, the input index is implicitly sorted by observing date, then scan number, then backend identifier[1]. For example, the following files are ordered like this:

```
MRTCAL> $ls *.fits
iram30m-4mhz-20100930s200-imb.fits  iram30m-fts-20100930s201-imb.fits
iram30m-4mhz-20100930s201-imb.fits  iram30m-wilma-20100930s200-imb.fits
iram30m-fts-20100930s200-imb.fits   iram30m-wilma-20100930s201-imb.fits
MRTCAL> mlist in  ! "in" lists the input index
  N.V ProjId    Source         Date          UT     Scn Backe  ObsType    Swi Calibr
  1.1 054-09   CONTROLD     30-SEP-2010  20:52:47.0 200 4MHZ  CALIBRATE   PSW NONE
  2.1 054-09   CONTROLD     30-SEP-2010  20:52:47.0 200 WILMA CALIBRATE   PSW NONE
```

[1] Backend identifiers are ordered chronologically.

```
  3.1 054-09   CONTROLD     30-SEP-2010  20:52:47.0 200 FTS   CALIBRATE   PSW NONE
  4.1 054-09   CONTROLD     30-SEP-2010  20:53:37.0 201 4MHZ  TRACKED     PSW NONE
  5.1 054-09   CONTROLD     30-SEP-2010  20:53:37.0 201 WILMA TRACKED     PSW NONE
  6.1 054-09   CONTROLD     30-SEP-2010  20:53:37.0 201 FTS   TRACKED     PSW NONE
```

From the input index, the user has to build the *current index*[2] with `MFIND`. They can be identical (`MFIND` called without argument), or the current index can be a subset of the input index. For example:

```
MRTCAL> mfind /backend 4mhz
I-FIND,  2 entries in Current indeX
MRTCAL> mlist  ! No argument: list the current index
  N.V ProjId    Source         Date          UT     Scn Backe  ObsType    Swi Calibr
  1.1 054-09   CONTROLD     30-SEP-2010  20:52:47.0 200 4MHZ  CALIBRATE   PSW NONE
  4.1 054-09   CONTROLD     30-SEP-2010  20:53:37.0 201 4MHZ  TRACKED     PSW NONE
```

The number N which can be seen in the first column is used by several commands to identify the file to be processed (for example, `CALIBRATE 1`). In most of the cases, especially when dealing with a data set which does not evolve, using this number in procedures is acceptable. However, there are some use cases where one needs to pay attention:

- `INDEX UPDATE`: if new files are to be added to an old index file, they will be inserted according to the sorting rule described earlier. If those files were observed after the old ones (*e.g.* if adding a new observation session), they will be appended to the end of the index, so this makes no problem. But if a file has to be inserted before, this will modify the entry numbers.
- `INDEX OPEN /RECURSIVE`: when opening several directories (one index file each), one has to be careful that, again, all the files will be sorted with the same rule. This basically means that the entry numbers which are seen when opening a single directory are not the same when this directory is opened together with other directories.

2.2.2 Versioning

The first time an IMB-FITS file is indexed, it is given a version number, starting at 1. Then, when it is calibrated, a new version of the entry is added in the index. For example, using the same example as before:

```
MRTCAL> calibrate 1

Calibrating   1.1 054-09   CONTROLD     30-SEP-2010  20:52:47.0 200 4MHZ  CALIBRATE   PSW NONE
[calibration messages]
```

yields

```
MRTCAL> mlist in
  N.V ProjId    Source         Date          UT     Scn Backe  ObsType    Swi Calibr
  1.1 054-09   CONTROLD     30-SEP-2010  20:52:47.0 200 4MHZ  CALIBRATE   PSW NONE
  1.2 054-09   CONTROLD     30-SEP-2010  20:52:47.0 200 4MHZ  CALIBRATE   PSW DONE EOHLI:254 EOVLI:198
  2.1 054-09   CONTROLD     30-SEP-2010  20:52:47.0 200 WILMA CALIBRATE   PSW NONE
  3.1 054-09   CONTROLD     30-SEP-2010  20:52:47.0 200 FTS   CALIBRATE   PSW NONE
  4.1 054-09   CONTROLD     30-SEP-2010  20:53:37.0 201 4MHZ  TRACKED     PSW NONE
  5.1 054-09   CONTROLD     30-SEP-2010  20:53:37.0 201 WILMA TRACKED     PSW NONE
  6.1 054-09   CONTROLD     30-SEP-2010  20:53:37.0 201 FTS   TRACKED     PSW NONE
```

[2]The current index is the list of entries to be processed by the command `PIPELINE`.

One can see that a new entry, numbered 1.2, has been added to the input index. It is identical to the version 1.1, except that its calibration status is now `DONE` (and the calibration results are saved in the index file). The version 1.2 was inserted just after the version 1.1 so that the correct sorting is kept.

Beware that, when a new entry is added to the input index, the current index is NOT udpated. One has to call `MFIND` again. Note that `MFIND` finds only the latest versions of each entry:

```
MRTCAL> mfind /backend 4mhz
I-FIND,  2 entries in Current indeX
MRTCAL> mlist
  N.V ProjId    Source          Date          UT     Scn Backe  ObsType    Swi Calibr
  1.2 054-09   CONTROLD     30-SEP-2010  20:52:47.0 200 4MHZ  CALIBRATE   PSW DONE E0HLI:254 E0VLI:198
  4.1 054-09   CONTROLD     30-SEP-2010  20:53:37.0 201 4MHZ  TRACKED     PSW NONE
```

Consider also that an entry can be calibrated several times, *e.g.* to test the effect of a modified calibration tuning. Each of these calibrations will increase the version number, producing a new entry in the index, memorizing its calibration results.

3 Calibrating

3.1 Processing all a directory

The easiest **MRTCAL** session is:

```
MRTCAL> index build      ! Build the index of IMB-FITS files in the current directory
MRTCAL> mfind            ! Select all the files for calibration
MRTCAL> mlist            ! Optionally list the files to be calibrated
MRTCAL> file out myclassfile.30m single  ! Open a new Class output file
MRTCAL> pipeline         ! Calibrate all the files
```

This calibrates all the IMB-FITS files present in the current directory. The command `MFIND` is used to build the *current index* from all the files in the *input index*. The command `PIPELINE` processes all, and only, the entries in the current index. It chooses automatically how the calibration and science scans are paired. As a result, the calibrated spectra are saved in the named **CLASS** file and the index is updated to reflect the calibration status of the IMB-FITS files.

3.2 Processing a single IMB-FITS file

In order to calibrate a single science scan, one has to use the command `CALIBRATE`:

```
MRTCAL> index build  ! Build index of files in current directory
MRTCAL> mfind        ! Selects all the entries in index
MRTCAL> mlist        ! List the entries in the Current indeX
MRTCAL> file out myclassfile single  ! The usual CLASS command
MRTCAL> calibrate 1  ! Where "1" is the entry number of the calibration scan
MRTCAL> calibrate 2  ! Where "2" is the entry number of the science scan
```

The 2 last lines can also be merged in a single call:

```
MRTCAL> calibrate 2 /with 1
```

As a result, only the 2 named scans are calibrated. The entries in the index file are also updated to reflect that their calibration has been done.

3.3 Calibration status

When an IMB-FITS file is first indexed, its calibration status is set as `NONE`. For example:

```
  N.V ProjId    Source         Date          UT     Scn Backe  ObsType    Swi Calibr
  1.1 054-09   CONTROLD     30-SEP-2010  20:52:47.0 200 4MHZ  CALIBRATE   PSW NONE
```

When it is successfully calibrated, its status is set as `DONE`:

```
  N.V ProjId    Source         Date          UT     Scn Backe  ObsType    Swi Calibr
  1.2 054-09   CONTROLD     30-SEP-2010  20:52:47.0 200 4MHZ  CALIBRATE   PSW DONE E0HLI:254 E0VLI:198
```

However, in case of failure during the calibration, the status is set as `FAILED`. It can be useful to find all the failed calibrations for a closer look. This can be achieved with the command:

```
MRTCAL> MFIND /CALIBRATION FAILED
```

It can also happen that an IMB-FITS file has no useful data. The file structure is correct but either it contains no subscan, or no `ON` subscan, or no data dumps within the susbcans. In these cases, the calibration does not fail (from the computing point of view) but there is nothing relevant which can be done from these files: calibration scans can not be used, and science scans produce no spectra. Their calibration status is set as `EMPTY`.

Note that the `PIPELINE` command chooses automatically which calibration scan it pairs with a science scan. In particular, it ignores all the `FAILED` and `EMPTY` calibration scans, which means either it uses a `DONE` calibration scan, or it processes the desired one if not yet done.

3.4 Calibration products

By default, after processing a calibration scan, **MRTCAL** will save the calibration products in the **CLASS** output file[3]. They will thus be available to the user when he opens the **CLASS** file for reading. For example, after calibrating iram30m-4mhz-20100930s200-imb.fits and iram30m-4mhz-20100930s201-imb.fits one can see:

```
MRTCAL> file in 09054.30m
I-CONVERT,  File is  [Native]
I-INPUT,  09054.30m successfully opened
MRTCAL> find
I-FIND,  14 observations found
MRTCAL> list
Current index contains:
 N;V Source       Line         Telescope      Lambda     Beta Sys  Sca Sub
 1;1 CALSKY       NULL         30ME0HLI-4M1   -600.0     +0.0 Eq   200 1
 2;1 CALSKY       NULL         30ME0VLI-4M2   -600.0     +0.0 Eq   200 1
 3;1 CONTROLD     NULL         30ME0HLI-4M1     +0.0     +0.0 Eq   201 2
 4;1 CONTROLD     NULL         30ME0VLI-4M2     +0.0     +0.0 Eq   201 2
 5;1 CONTROLD     NULL         30ME0HLI-4M1     +0.0     +0.0 Eq   201 4
 6;1 CONTROLD     NULL         30ME0VLI-4M2     +0.0     +0.0 Eq   201 4
 7;1 CONTROLD     NULL         30ME0HLI-4M1     +0.0     +0.0 Eq   201 6
 8;1 CONTROLD     NULL         30ME0VLI-4M2     +0.0     +0.0 Eq   201 6
 9;1 CONTROLD     NULL         30ME0HLI-4M1     +0.0     +0.0 Eq   201 8
10;1 CONTROLD     NULL         30ME0VLI-4M2     +0.0     +0.0 Eq   201 8
11;1 CONTROLD     NULL         30ME0HLI-4M1     +0.0     +0.0 Eq   201 10
```

[3]This can be disabled by setting `MSET CALIBRATION PRODUCT NONE` before processing the calibration scans.

```
12;1 CONTROLD     NULL         30ME0VLI-4M2     +0.0     +0.0 Eq   201 10
13;1 CONTROLD     NULL         30ME0HLI-4M1     +0.0     +0.0 Eq   201 12
14;1 CONTROLD     NULL         30ME0VLI-4M2     +0.0     +0.0 Eq   201 12
```

The calibration products are saved as **CLASS** spectra with source name `CALSKY`. There are as many of them as frontend-backend combination (*telescope* name in **CLASS**). In order to find only the science data, user has to find explicitly his source name, *e.g.*:

```
MRTCAL> find /source controld
I-FIND,  12 observations found
```

A `CALSKY` entry is a spectrum providing the CalSky load as the main data (`RY`), plus a collection of Associated Arrays:

```
MRTCAL> find /source calsky
I-FIND,  2 observations found
MRTCAL> get first
I-GET,  Observation 1; Vers 1 Scan 200
MRTCAL> dump
[...]
 ASSOCIATED ARRAYS ----------------------------------------
  Number of associated arrays: 7
  Associated array #1: CALAMBIENT
    Unit:               Format: -11  Bad:   -1000.000  Dimensions: 1006 x 0
    Data:    10158.63       9447.842       ...      17594.74       15591.32
  Associated array #2: CALCOLD
    Unit:               Format: -11  Bad:   -1000.000  Dimensions: 1006 x 0
    Data:    5958.474       5238.263       ...      7394.842       7605.105
  Associated array #3: TREC
    Unit:               Format: -11  Bad:   -1000.000  Dimensions: 1006 x 0
    Data:    304.3947       304.3947       ...      177.7052       177.7052
  Associated array #4: TCAL
    Unit:               Format: -11  Bad:   -1000.000  Dimensions: 1006 x 0
    Data:    243.2139       243.2139       ...      249.4267       249.4267
  Associated array #5: TSYS
    Unit:               Format: -11  Bad:   -1000.000  Dimensions: 1006 x 0
    Data:    424.9489       423.0339       ...      279.2844       283.7039
  Associated array #6: TAUZEN
    Unit:               Format: -11  Bad:   -1000.000  Dimensions: 1006 x 0
    Data:   6.2096495E-02  6.2096495E-02   ...     5.5034418E-02  5.5034418E-02
  Associated array #7: FLAG
    Unit:               Format: -11  Bad:   -1000.000  Dimensions: 1006 x 0
    Data:    0.000000       0.000000       ...      0.000000       0.000000
```

Those 8 arrays provide useful information as a function of frequency:

- CalSky, CalAmbient, CalCold: the calibration loads,
- Trec, Tcal, Tsys: the receiver, calibration, and system temperatures,
- TauZen: the zenithal opacity,
- Flag: a flag array indicating if the data is reliable (0) or not (1). As of today, **MRTCAL** flags all the channels of a chunk if any error occurs when performing the chopper calibration (e.g. sky greater than hot load).

The Figure 1 shows an example of these arrays.

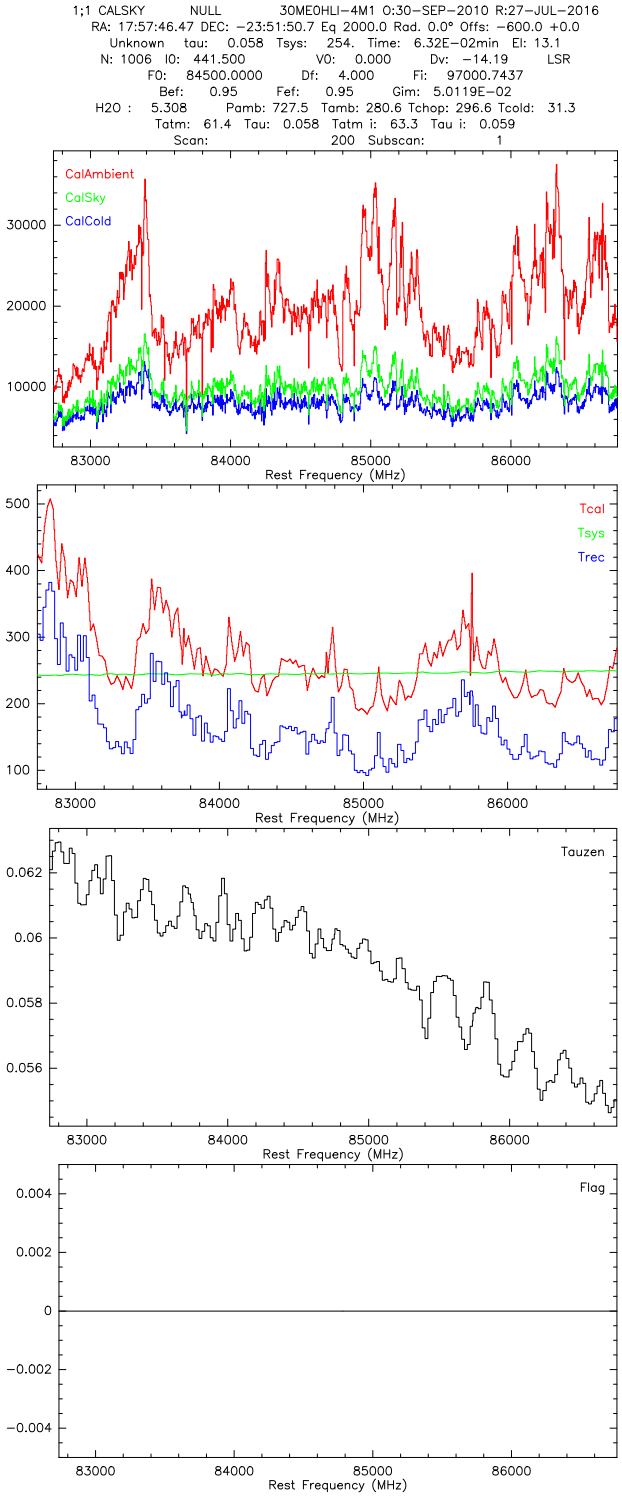


Figure 1: Calibration products as found after calibrating the file iram30m-4mhz-20100930s200-imb.fits. The default calibration bandwidth was used (20 MHz).

3.5 Weight array

MRTCAL offers the possibility to associate a weight array (Associated Array W) together with the spectrum intensities (`RY`). This weight varies in frequency. The weight at channel i is:

$$W(i) = \frac{t \times |\delta f|}{(T_{sys}(i) \times e^{A\tau(i)})^2} \quad (1)$$

where t, δf and A are respectively the integration time (seconds), the frequency resolution (MHz) and the airmass (same scalar values shared by all channels), and $\tau(i)$ is the opacity at channel i. $T_{sys}(i)$ is the *zenithal* system temperature, which can be found in the calibration products from a calibration scan (see subsection 3.4).

By default, **MRTCAL** does not save the weight array. This is activated by the command:

```
MRTCAL> MSET OUTPUT WEIGHT YES
```

Beware that if enabled, this writes twice more data per spectrum, which typically doubles the **CLASS** file size! An example of a weight array is plotted in the Figure 2, produced with the following commands:

```
LAS> file in 09054
I-CONVERT,  File is  [Native]
I-INPUT,  09054.30m successfully opened
LAS> find /source controld
I-FIND,  12 observations found
LAS> get first
I-GET,  Observation 3; Vers 1 Scan 201
LAS> plot w
```

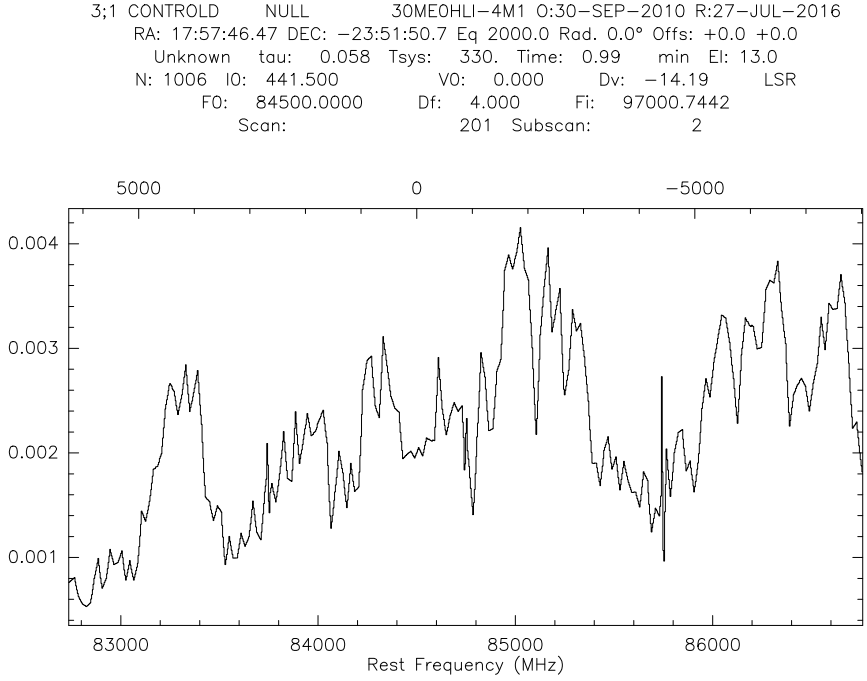


Figure 2: Weight array associated to a spectrum produced from the file iram30m-4mhz-20100930s200-imb.fits.

4 Exploring IMB-FITS files (no calibration)

4.1 FITS reader

Using **MRTCAL** as a FITS reader can be done with a few commands, *e.g.*:

```
MRTCAL> read iram30m-fts-20120128s233-imb.fits /file /subscan 2 ! Load the HDUs for subscan 2
MRTCAL> mdump                   ! Summary of the file HDUs
MRTCAL> mdump antslow           ! Details of the AntSlow HDU for the current subscan
MRTCAL> mdump antslow longoff   ! The LONGOFF table for the current subscan
MRTCAL> variable                ! Create the related Sic variables, e.g.
MRTCAL> examine subs%antslow%table%longoff  ! the same as Sic array
```

See `HELP` for details and options about these commands. The procedures in the next subsections are extensively based on this IMB-FITS reader.

4.2 FITS plots

MRTCAL offers a serie of procedures which allow plotting various **IMBFITS** contents. They are intended to help understanding the data and diagnose possible problems.

4.2.1 Calibration loads

The loads of a calibration scan can be displayed in a plot with *e.g.* the command:

```
MRTCAL> @ plot-loads iram30m-4mhz-20100930s200-imb.fits
```

The result is shown in the Figure 3. It shows the load values as found in the `DATA` column of each subscan. They can be averaged (*i.e.* display a single average load) or not (*i.e.* display all the single dumps) thanks to the option `/AVERAGE YES|NO` (default is `YES`). Under normal conditions, we expect the load intensities to arrange in the order Cold < Sky < Hot for a proper calibration.

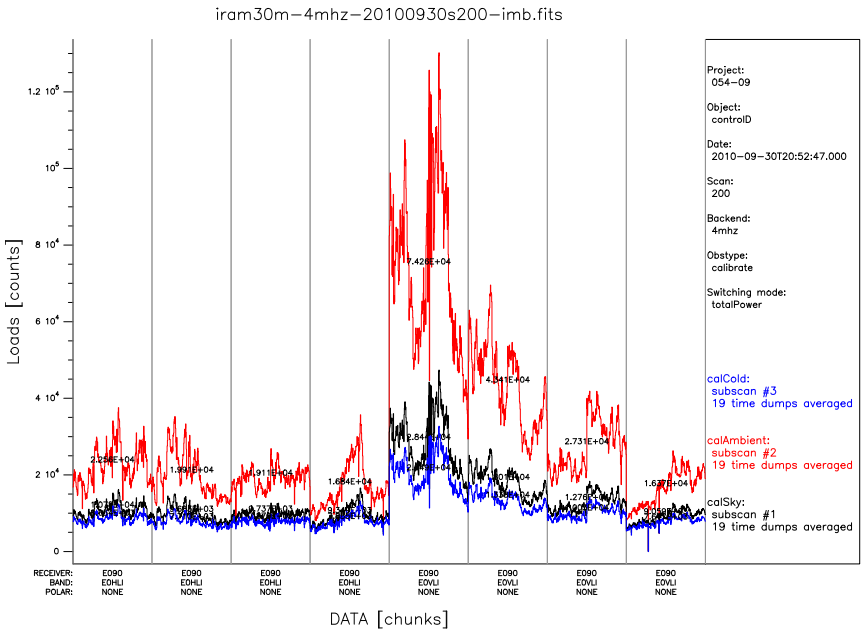


Figure 3: Resulting plot of the procedure `@ plot-loads`. X axis is the output of the backend, *i.e.* a collection of backend "parts" (*e.g.* E0HLI, E0VLI, and so on) glued together; they need to be split and reassembled in the correct order to produce spectra with sky frequencies. It displays i) the 3 loads Cold, Sky, and Hot, and ii) vertical grey lines showing the range of channels to be used; outer channels are discarded (none here). The mean load value per chunk is also displayed in black, overlaid to each chunk load curve.

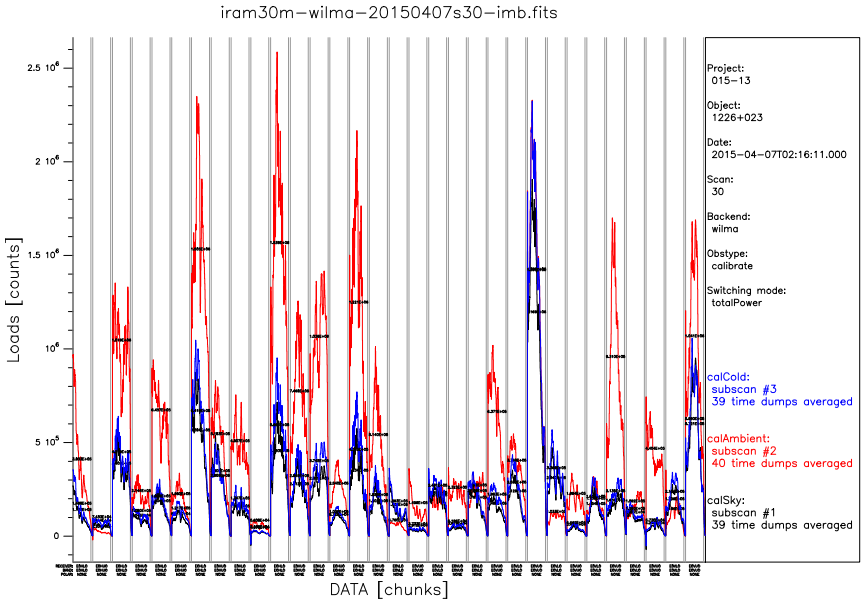


Figure 4: Same as Fig. 3, but this calibration scan has incorrect loads (note the E0VLO backend part in the center with Hot < Sky < Cold).

4.2.2 Display MJD values and tables

The headers and tables found in the IMB-FITS file rely heavily on time-stamps described with Modified Julian Date (MJD). This includes subscan start and stop, instantaneous antenna positions during the subscan, dump sampling during integration. These quantities can be displayed in a graphical way with *e.g.* the following command:

```
MRTCAL> @ plot-mjd iram30m-4mhz-20100930s200-imb.fits
```

The result is shown in the Figure 5. Under normal conditions, there should be backend data and antenna positions during the whole subscan, *i.e.* the markers should start before and end after the box edges. It is a sane behaviour to have a bit more of data before and after the subscan limits.

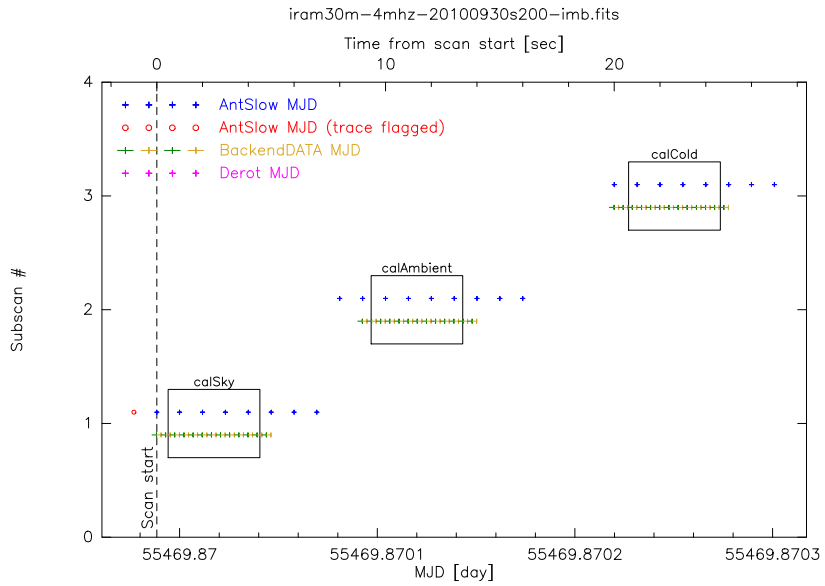


Figure 5: Resulting plot of the procedure `@ plot-mjd`. X axis is time, Y axis is the subscan number. It displays i) the "on-source" time coverage of each subscan (subscan-start and subscan-stop as boxes), ii) the time coverage of the backend data in each subscan (green+gold horizontal segments), iii) the time coverage of the antenna slow positions (blue or red markers), iv) the time stamps of the derotator dumps if relevant (magenta crosses). Each DATA dump is displayed as an horizontal segment showing its actual integration time. The associated time stamp is displayed as a vertical segment (the time stamp can be found anywhere from the beginning to the end of the integration, according to the header value `TSTAMPED`); colors are alternated to avoid confusion.

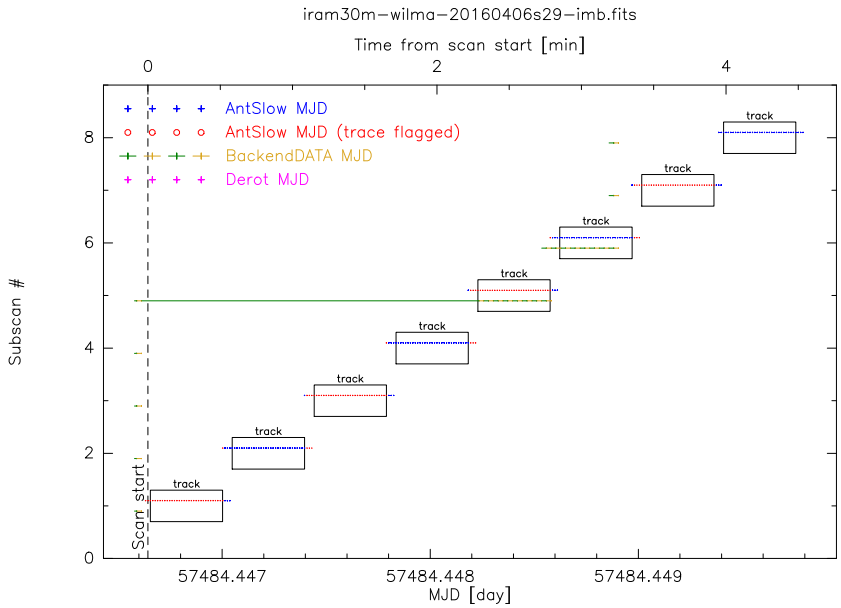


Figure 6: Same as Fig. 5, but the backend tables are incorrect (a lot of subscans have no backend data in the subscan range).

4.2.3 Display antenna position

Observing modes typically involve the antenna to track a given position (per subscan) or drift on sky (also per subscan), and switching between these track/on-the-fly subscans. Given the fact that the sky rotates in all cases, this results in typical patterns when looking at the antenna position. It can be displayed with

```
MRTCAL> @ plot-drifting-positions iram30m-wilma-20100930s201-imb.fits
```

The result is shown in the Figure 7. Under normal conditions, tracked subscans should track a single position during the subscan. On the other hand, on-the-fly subscans continuously drift on the sky. Depending on the observing mode, we can see ON (tracked or OTF) alterning with OFF (tracked) subscans.

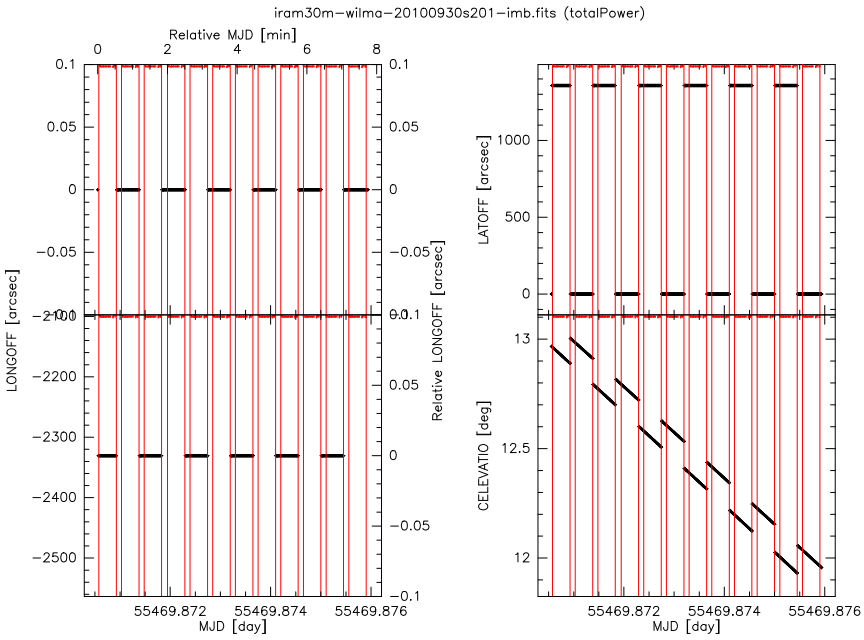


Figure 7: Resulting plot of the procedure `@ plot-drifting-positions`. X axis is time, Y axes are longitude, latitude, or commanded elevation. It displays i) the `LONGOFF` (relative longitude) of the antenna during the scan - the plot is divided in 2 parts as it is a common behavior to alternate between 2 longitudes from one subscan to another - , ii) the `LATOFF` (relative latitude) of the antenna during the scan, iii) the `CELEVATIO` (commanded elevation) during the scan, iv) the "on-source" time coverage of each subscan (as vertical red lines). This example is a typical tracked scan with 12 subscans alterning between ON and OFF positions (`LONGOFF` and `LATOFF` are constant in each subscan, but `CELEVATIO` drifts as Earth rotates.).

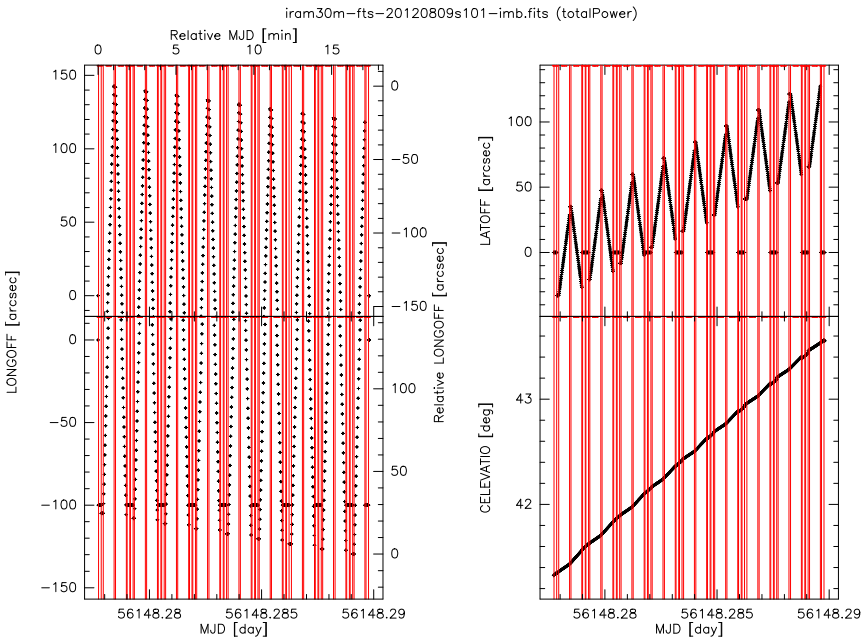


Figure 8: Same as Fig. 7, but for a typical On-The-Fly scan with 35 subscans alterning short OFF position, long scanning in a direction, long scanning in the opposite direction, short OFF position, and so on (ROOR repeated sequence). Short OFF subscans have $(-100, 0)$ relative coordinates.

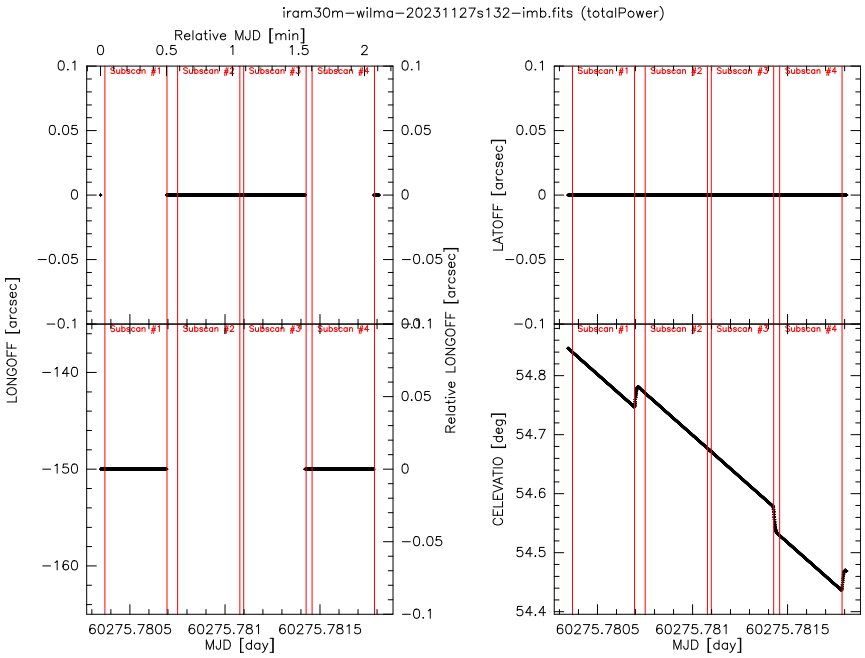


Figure 9: Same as Fig. 7, but this tracked scan has 2 different position in a subscan (when zooming closely one can see 2 faulty dumps at the end of each subscan).

5 MRTCAL Language Internal Help

5.1 Language

```
                    MRTCAL\ Command Language Summary

CALIBRATE: Calibrate a single file
INDEX    : Build/read the index of the IMB-FITS files
MCOPY    : Copy the index to a file
MDUMP    : DUMP the content of the Mrtcal buffers
MFIND    : Select entries from index file
MLIST    : List the files selected by the FIND command
MSETUP   : Tune some program behaviours
MUPDATE  : Update the index and IMB-FITS files from buffers
MWRITE   : WRITE the class data format
PIPELINE : Calibrate the Current indeX
READ     : READ IMBFITS file
SOLVE    : SOLVE for the pointing/focus/skydip parameters
VARIABLE : Add Sic variables mapping internal buffers
```

5.2 CALIBRATE

```
    [MRTCAL\]CALIBRATE Num[.Ver] [/WITH Num[.Ver]]

Calibrate the file indentified by its Number. The calibration scan is
chosen implicitly by the command, except if explicitly specified through
the option /WITH.

Boths file numbers can be added a Version number. If not given, the lat-
est version is selected.
```

5.3 INDEX

```
    [MRTCAL\]INDEX [BUILD|UPDATE|OPEN|APPEND|WATCH|CONVERT  [DirName]]
[/FILE IndexFile] [/RECURSIVE] [/PATTERN String] [/DATE DateString]
[/TIMEOUT Value]

This commands creates, updates, or opens a file index (database) of IMB-
FITS files, for later use by the commands MFIND and/or CALIBRATE.
  - BUILD creates or rebuilds the index file from scratch, overwriting
    the previous one if any,
  - UPDATE creates or updates the index file if it already exists,
  - OPEN opens the index file,
  - APPEND appends in memory another index file,
  - OUTPUT select the index file used for writing new entries,
  - WATCH waits for new files to appear in the IMB-FITS directory, and
    index them (INDEX UDPATE) as soon as one new is found.
  - CONVERT converts the index file to the latest known version (if rel-
    evant).
BUILD and UPDATE will implicitly OPEN the index file after creating it.
The default is to open the index of the current working directory.

By default, the index file created/searched as 'index.mrt' in the direc-
```

tory where the IMB-FITS files are (DirName). However, if user has no right (e.g. write access) to put the index file in this directory, he can override this default and give any IndexFile name (with its path). In this case, it is his responsibility to ensure a correct DirName/IndexFile match. For clarity reasons, the default should be prefered when possible.

5.3.1 INDEX UPDATE

```
[MRTCAL\]INDEX UPDATE [DirName] [/PATTERN String] [DATE DateString]
```

Build or update the index of the IMB-FITS files. The index is stored in a binary file named 'index.mrt'. The command implicitly performs an INDEX OPEN right after the UPDATE to allow direct use of the index.

The index lists all the files named 'iram30m-*-*s*-imb.fits' in the directory, including unreadable and incomplete (in terms of number of subscans) ones. Selection is made from this list with the various options and features of the commands using this list.

The command can be called several times. The index file is updated if needed, in particular if the file readability has improved (e.g. scan has been completed).

By default, the command searches for the IMB-FITS files in the current working directory. This can be overriden thanks to the option /DIRECTORY. Note that the index file is always created in the same directory of the files it lists (write access is thus needed). Directory search is not recursive.

5.3.2 INDEX OPEN

```
[MRTCAL\]INDEX OPEN [DirName]
```

Load the index in memory, for subsequent use by the commands FIND and/or CALIBRATE. Default is to open the index of the current working directory.

5.3.3 INDEX OUTPUT

```
[MRTCAL\]INDEX OUTPUT [DirName] [/FILE IndexFile]
```

Append a new index file in memory (i.e. appended to the Input indeX) and make it the default for writing new entries (i.e. entries saved after a calibration). This index file is thus in-out.

By default, the new entries are saved in the same index file they come from. If you want to keep the original index file unmodified (e.g. if you have no write access), use this command to select another output file. This output index destination is reset by commands which fully reset the Input indeX, namely INDEX OPEN and INDEX BUILD.

Note that if you mix several directories in the Input indeX, it becomes

a non-sense to use a unique index file as output (you will get lost when reopening it). This command is basically protected against this situation.

5.3.4 INDEX WATCH

[MRTCAL\]INDEX WATCH [/TIMEOUT Value]

Wait for new files added in the directory associated to the input index. As soon as new data is detected, the index is updated (same as INDEX UPDATE) and the command returns. There are 3 major waiting modes:
- INDEX WATCH: wait indefinitely,
- INDEX WATCH /TIMEOUT Value>0: wait until this time (in seconds) is elapsed, and then return normally. Note that you can mimic an infinite loop by setting a huge value (1 year is ~3e7 seconds),
- INDEX WATCH /TIMEOUT Value<=0: does not wait. This can be useful when writing generic code including time-outs.

The waiting loop can also be stopped with CTRL-C. In this case the command will return with an error status.

5.3.5 INDEX CONVERT

[MRTCAL\]INDEX CONVERT [/FILE IndexFile]

Check if the index file version is the latest possible. If not, the associated file on disk is automatically converted. The original file is backup'ed to another name.

While index file version has very seldom evolutions, when this happens it is necessary to convert 'old' files opened in UPDATE mode beforehand, as MRTCAL writes only in latest version files.

5.3.6 INDEX /PATTERN

[MRTCAL\]INDEX [BUILD|UPDATE|WATCH [DirName]] [/FILE IndexFile] [/RECURSIVE] /PATTERN String

The option /PATTERN allows to indicate which files should be indexed. Default is 'iram30m-*-*s*-imb.fits', i.e. all the IMB-FITS files in the directory, but this can be restricted with other patterns. Note that file names matter in Mrtcal: basically you can not index a file which do not match the default pattern above.

For best portability, you should use a sh-compatible syntax.

5.3.7 INDEX /DATE

[MRTCAL\]INDEX [BUILD|UPDATE|WATCH [DirName]] [/FILE IndexFile] [/RECURSIVE] /DATE [List1] [List 2] ... [ListN]

The option /DATE allows to restrict the files to be indexed for a date or list of dates.

```
A list has the form "Date1 [TO Date2]". Typical uses are:
  /DATE Date: select this unique date,
  /DATE Date1 Date2 Date3: select those 3 scalar dates,
  /DATE Date1 TO Date2: select this range of dates,
  /DATE Date1 Date2 Date3 TO Date4: select these dates (Date1,
    Date2) or range of dates (Date3 to Date4).

Accepted formats are:
    - DD-MMM-YYYY (MMM as letters),
    - YYYYMMDD (all numbers),
    - Keyword "TODAY" or "YESTERDAY".
```

5.4 MCOPY

```
    [MRTCAL\]MCOPY [IN|CURRENT] /FILE IndexFile

Copy the named index (default current) from memory to the given file  to
disk.  Any  pre-existing  file  is  overwritten, except that you can not
overwrite one of the input index files involved during this operation.

This command is useful for a few main purposes:
  - extract a subset from an index file, by combining the commands INDEX
    OPEN, MFIND (with criteria), and MCOPY,
  - merge  several  index  files,  by combining INDEX OPEN, INDEX APPEND
    (repeated), MFIND, MCOPY,
  - convert an old format file to latest format, simply by using INDEX
     OPEN, MCOPY IN.
```

5.5 MDUMP

```
    [MRTCAL\]MDUMP [/OUTPUT LogFile]
    [MRTCAL\]MDUMP ALL|SUMMARY|SUBSCANS|HduName|DATA [/OUTPUT LogFile]
    [MRTCAL\]MDUMP HduName ColumnName [/OUTPUT LogFile]

Display the content of all or one section from the internal memory  buf-
fers. Default section is ALL sections.

One of the following section can be used:
  - ALL is a shortcut for SUMMARY + SUBSCANS + the HDUs list
  - SUMMARY  displays  a short summary of the file contents, and equiva-
    lence classes of the subscans (i.e. grouped by categories).
  - SUBSCANS shows a list of the subscans with their observing types,
  - HduName can be one of the following names:
      PRIMARY, SCAN, FRONTEND, BACKEND, DEROT
      BACKDATA, ANTSLOW, ANTFAST (of the current subscan)
  - DATA shows a summary of the current subset of the  DATA  column  for
    the  current  subscan.  It  also  shows how this data is mapped into
    chunksets.

If an HDU name is requested, the header keys are displayed (key +  value
+  comment),  and the table columns are summarized (key + a few values +
```

comment from associated TTYPE). If the HDU name is followed by a column name, the full column contents is displayed.

Note that the internal IMB-FITS reader may apply patches at read time (e.g. old headers are patched to mimic the latest IMB-FITS format). In other words, what MDUMP shows in not exactly the FITS contents, but what the calibration engines use. In case of patch, the comments are replaced by a warning message with a brief explanation.

5.5.1 MDUMP /OUTPUT

```
[MRTCAL\]MDUMP [Topic] [SubTopic] /OUTPUT LogFile
```

The option /OUTPUT will redirect the display to the given file. Default is on screen (STDOUT).

5.6 MFIND

[MRTCAL\]MFIND [/DATE Date] [/SCAN List] [/BACKEND List] [/FRONTEND List] [/OBSTYPE List] [/SWITCHMODE List] [/SOURCE Name] [/PROJID Id] [/COMPLETENESS Key] [/CALIBRATED List] [/POLARIMETRY YES|NO]

Select IMB-FITS files from the index file and build the Current Index in memory. With no options, all files are selected. The options allow to add one or more selection criteria. Using a wildcard '*' as the option value makes it uneffective. The number of selected entries is stored in the variable MFOUND, and the index arrays are available in the structure MDX%.

- /DATE: the date list, various formats are supported (see subtopic for details),
- /SCAN: the scan list. See subtopic for details
- /BACKEND: the backend name(s) as known in the IMB-FITS file.
- /FRONTEND: the receiver name(s), in single or dual band. See subtopic for details.
- /OBSTYPE: the observation type(s) as known in the IMB-FITS file.
- /SWITCHMODE: the switching mode(s) in the list PSW, WSW, FSW, BSW, and UNK(nown).
- /POLARIMETRY: select files with polarimetry data (YES|NO),
- /SOURCE: the source name. Wildcards are allowed in the name.
- /PROJID: the project identifier (character string). Wildcards are allowed in the string.
- /COMPLETENESS: one of the following:
  - YES: select only complete files,
  - NO: select unreadable, empty (0 subscan) or partial (missing subscans) files,
  - READABLE: select partial or complete files,
  - UNREADABLE: select only unreadable or empty files
- /CALIBRATED: select by one or several calibration statuses, in NONE, DONE, EMPTY, FAILED, or SKIPPED.

5.6.1 MFIND /SCAN

```
[MRTCAL\]MFIND /SCAN [List1] [List 2] ... [ListN]
```

```
Scan is selected if its number is found in one of the given lists, where
one list has the form "Val1 [TO Val2]". Typical uses are:
  /SCAN Val: select this unique value,
  /SCAN Val1 Val2 Val3: select those 3 scalar values,
  /SCAN Val1 TO Val2: select this range of values,
  /SCAN Val1 Val2 Val3 TO Val4: select these values (Val1, Val2) or
    range of values (Val3 to Val4).
```

5.6.2 MFIND /DATE

```
[MRTCAL\]MFIND /DATE [List1] [List 2] ... [ListN]
```

```
The file is selected if its scan date, as found in the file name, match-
es  one of the the given lists. The dates in the lists should be strings
in one of the following formats:
    - DD-MMM-YYYY (MMM as letters),
    - YYYYMMDD (all numbers),
    - Keyword "TODAY" or "YESTERDAY".

A list has the form "Date1 [TO Date2]". Typical uses are:
  /DATE Date: select this unique date,
  /DATE Date1 Date2 Date3: select those 3 scalar dates,
  /DATE Date1 TO Date2: select this range of dates,
  /DATE Date1 Date2 Date3 TO Date4: select these dates (Date1,
    Date2) or range of dates (Date3 to Date4).

Note that if the scan was started just before  midnight,  there  can  be
subscans  on  the  next  day,  i.e. calibrated data may have a different
date.
```

5.6.3 MFIND /FRONTEND

```
[MRTCAL\]MFIND /FRONTEND [Pattern1] ... [PatternN]
```

```
The pattern PatternI accepts 3 forms:
  Value
    A single receiver name. Find  single-band  observations  which  used
    this receiver.
  Value1&Value2
    Two  receiver  names with ampersand (&). Find dual-band observations
    which used these two receivers (order does not matter).
  Value1|Value2
    Two receiver names with pipe (|). Find dual-band observations  which
    used at least one of these two receivers (order does not matter).
The Values can be the exact receiver name (e.g. E090) or some wildcarded
pattern (e.g. E0* matches E090). Use * to match any receiver.

Several patterns can be passed to the option to  find  entries  matching
several criteria. For example:
```

```
  *           : Find all (and only) single-band observations.
  *&*         : Find all (and only) dual-band observations.
  * *&*       : Find all single-band and all dual-band observations (i.e.
                everything). This is the default.
  E090        : Find single-band observations with E090.
  E090&*      : Find dual-band observations with E090.
  E090 E090&* : Find all observations with E090.
  E090 E230   : Find single-band observations with E090 or E230.
  E090|E230   : Find dual-band observations with E090 or E230.
  E090&E230   : Find dual-band observations with E090 and E230.
```

5.7 MLIST

```
    [MRTCAL\]MLIST  [IN|CURRENT]  [/COLUMNS List] [/PAGE] [/FILE Output-
File]
    [MRTCAL\]MLIST [IN|CURRENT] /TOC Keys  [/PAGE]  [/VARIABLE  VarName]
[/FILE OutputFile]
    [MRTCAL\]MLIST [IN|CURRENT] /POINTING [/FILE OutputFile]
```

MLIST the content of the given index. MLIST CURRENT refers to the selection made with the command MFIND. MLIST IN lists all the files known in the memory index. Default is MLIST CURRENT.

MLIST offers 3 main kind of lists:

- Basic (no dedicated option): list the entries in index (one per line). The columns can be customized with /COLUMNS.
- Table-Of-Contents (/TOC): summarize the entries per group of properties. See subtopic for details.
- Pointing results (/POINTING): list all the pointing drift solutions found in the index. There can be 0 (not a pointing) to many solutions per entries.

The output can be redirected to an ASCII file thanks to the option /FILE. Default is print to STDOUT, continuously (scroll mode, default) or page by page (option /PAGE).

5.7.1 MLIST /TOC

```
    MRTCAL\MLIST  [IN|CURRENT] /TOC [Key1 Key2 ... KeyN] [/VARIABLE Var-
Name]
```

MLIST /TOC displays the Table Of Contents of the current index, or of the input (MLIST IN) index file. A summary of the different values of each attribute is made, plus a final summary of all the combinations (setups). The results are accessible in the structure MTOC%. This name can be customized thanks to the option /VARIABLE. The attributes can be:

- NUMBER: observation number
- VERSION: observation version
- TELESCOPE
- ID
- SOURCE
- OBSERVED

- SCAN
- FRONTEND
- BACKEND
- OBSTYPE
- SWITCHMODE
- POLARIMETRY: file contains polarimetry data
- CALIBRATED: calibration status
- DIRECTORY

By default, OBSERVED and SCAN are used.

5.7.2 MLIST /COLUMNS

```
MRTCAL\MLIST [IN|OUT] /COLUMNS [Attr1 Attr2 ... AttrN]
```

MLIST /COLUMNS allows to customize the columns displayed. They can be repeated as many time as desire. Valid values are:

- NUMBER
- VERSION
- DIRECTORY
- FITSFILE
- MRTFILE: the index file name
- RECORD: the position in the MRT index file
- WORD: the position in the record
- TELESCOPE
- PROJID
- SOURCE
- OBSERVED
- UT
- LST
- AZIMUTH
- ELEVATION
- POSITION: Lontigude, Latitude, Coordinate system, Equinox
- FRONTEND
- SCAN
- BACKEND
- OBSTYPE
- SWITCHMODE
- POLARIMETRY: file contains polarimetry data
- COMPLETE
- CALIBRATED
- ITIME (last indexation time)
- WITH: for a science scan, the Date, Scan, Backend, and Version of the associated calibration scan used when calibrating.
- RESULTS: results of pointing, focus, calibration scans
- DEFAULT: alias for the default columns

5.8 MSETUP

[MRTCAL\]MSETUP [INPUT|BOOKKEEPING|CALIBRATION|SOLVE|OUTPUT|PIPELINE|DEBUG]

Tune global 6 different kinds of behaviour for the READ, CALIBRATE and

```
PIPELINE commands:
    INPUT: how the command READ reads the data
    BOOKKEEPING: how the input data is handled
    CALIBRATION: how the calibration is performed
    SOLVE: how the solutions are computed
    OUTPUT: how the results are exported
    PIPELINE: how the pipeline works
    DEBUG: which debugging messages are displayed.

The MSETUP command without arguments will display the current status  of
all these categories. MSETUP CATEGORY without further argument will dis-
play the current status of the named category.
```

5.8.1 MSETUP INPUT

```
    [MRTCAL\]MSETUP INPUT Keyword Switch

[MRTCAL\]MSETUP INPUT BAD YES|NO

Choose if the command READ should read the bad time dumps  (in  addition
to  the  good  ones) from the backendXXX tables (including the DATA col-
umn). A bad time dump has a null ISWITCH value, and its associated  data
(e.g. MJD, INTEGTIM) are not expected to be correct. Default is YES.

[MRTCAL\]MSETUP INPUT BANDWIDTH Value

Setup  the  desired reading bandwidth in MHz. The nearest value that al-
lows the division of the good spectra channels of each chunkset in equal
virtual chunk sizes will be used. Exceptions are:
  - if value is 0, use the native hardware chunk size (no subdivision),
  - if  value  is larger than the native hardware chunk size, the latter
    is used instead (same as 0),
  - if value is lower than the channel spacing, the latter is  used  in-
    stead (all channels calibrated individually, use with caution).
Default is native chunk size.

[MRTCAL\]MSETUP INPUT DATA NONE|ONTRACK|ALL

Select how Mrtcal should read the DATA column:
  - NONE:  no data is read. Useful to read the other IMB-FITS tables and
    avoiding the DATA column which may be large (save time and/or  memo-
    ry).
  - ONTRACK:  read  only the on-track part of the DATA column, according
    to the subscan start and stop MJDs.
  - ALL: real the whole DATA column, including the off-track parts.
Default is ONTRACK.

[MRTCAL\]MSETUP INPUT MJDINTER YES|NO

Choose if the command READ should intersect the  DATE-OBS  and  DATE-END
(on-track  range)  of the subscan with the MJD columns in the tables An-
tenna Slow, Antenna Fast, and Backend Data. It may happen  that  one  of
```

these columns are too short and the data dumps can not be processed. Setting YES means that the associated dumps in this missing range won't be considered on-track anymore. Default is NO.

[MRTCAL\]MSETUP INPUT TOCHUNK YES|NO

Choose if Mrtcal should map the DATA column into chunksets at read time. NO means that you won't be able to do anything but READing the raw DATA into the internal buffers, and accessing from the VARIABLEs. YES is needed for calibration. This is the default.

5.8.2 MSETUP BOOKKEEPING

[MRTCAL\]MSETUP BOOKKEPPING SPACE value

Set the amount (in MB) of RAM memory used to buffer the input backend data. This allows the user to handle huge data amount on a small computer, even though it will be slower.

5.8.3 MSETUP CALIBRATION

[MRTCAL\]MSETUP CALIBRATION Keyword Switch

[MRTCAL\]MSETUP CALIBRATION ANTSLOW_MJD_SHIFT Value

Apply (add) a shift in the antenna slow MJD traces of the given value (in seconds, can be positive or negative). As a result the antenna will appear earlier or later at the associated position (namely LONGOFF, LATOFF, CAZIMUTH, CELEVATI). The subscan start and stop timestamps are NOT shifted. The antenna fast traces are NOT shifted neither. Default is 0, i.e. no shift is applied.

[MRTCAL\]MSETUP CALIBRATION BAD YES|NO

Choose if the bad time dumps in the BackendXXX tables should be calibrated (in addition to the good ones). A bad time dump has a null ISWITCH value, and its associated data (e.g. MJD, INTEGTIM) are not expected to be correct. If YES is requested, you will likely be exposed to problems (e.g. if MJD column has incorrect values, searches in this column may fail without advise). Default is NO.

[MRTCAL\]MSETUP CALIBRATION BANDWIDTH Value

Setup the desired calibration bandwidth in MHz. The nearest value that allows the division of the good spectra channels of each chunkset in equal virtual chunk sizes will be used. Exceptions are:
- if value is 0, use the native hardware chunk size (no subdivision),
- if value is larger than the native hardware chunk size, the latter is used instead (same as 0),
- if value is lower than the channel spacing, the latter is used instead (all channels calibrated individually, use with caution).

Default is 20 MHz.

[MRTCAL\]MSETUP CALIBRATION CHOPPER STRICT|TOLERANT

Setup the chopper calibration engine. By default (strict mode), any error occuring during this process (e.g. cold load greater than hot load) blanks the resulting data at the chunk level (i.e. the surrounding chunks are calibrated if they have no problem). In tolerant mode, the engine goes to the end of process, even if this implies non physical results (e.g. negative Trec). This second mode is dangerous and should be used for testing purposes only.
In both modes, a FLAG array is added to the calibration products (see HELP MSET OUTPUT CALIBRATION). It is 0 where no problem occured, and 1 otherwise.

[MRTCAL\]MSETUP CALIBRATION FEEDBACK PIXEL|SET|ELEMENT

Tune the granurality of the calibration results (Tsys, Tcal, Trec, ...). The median value per pixel, per chunkset, or per chunk element will be printed in the PIXEL, SET, or ELEMENT mode, respectively. One value per chunk element may result in very long lists when using virtual chunk sizes (see MSETUP CALIBRATION BANDWIDTH). Default is SET.

[MRTCAL\]MSETUP CALIBRATION INTERVAL ErrLimit [WarnLimit]

Setup the time limits (in minutes) used when searching automatically for a calibration scan to be associated to a science scan. Beyond ErrLimit, an error is raised if no calibration scan is found. Beyond WarnLimit, only a warning is displayed. Automatic search is used by the commands PIPELINE and CALIBRATE without the option /WITH. The calibration scan always can be forced with CALIBRATE /WITH. Default ErrLimit is 30 minutes, and WarnLimit defaults to 2/3 of ErrLimit.

[MRTCAL\]MSETUP CALIBRATION MATCH Value

Define the position matching tolerance when checking the subscan position(s) on sky. This tolerance is used 1) when checking that a 'track' subscan effectively provides dumps at the same position, and 2) when grouping the subscans per ON or OFF positions, and checking that each group effectively provides the same position. Default is 4 centi-arcseconds.

[MRTCAL\]MSETUP CALIBRATION MJDINTER YES|NO

Choose if Mrtcal should intersect the DATE-OBS and DATE-END (on-track range) of the subscans with the MJD columns in the tables Antenna Slow, Antenna Fast, and Backend Data. It may happen that one of these columns are too short and the data dumps can not be calibrated. Setting YES means that the associated dumps in this missing range won't be considered on-track anymore, and won't be calibrated. Default is NO.

[MRTCAL\]MSETUP CALIBRATION OFF NEAREST|LINEAR

Define how to select the OFF subscan matching the current ON subscan data (OTF PSW only). In NEAREST mode, the OFF subscan nearest in time will be used. In LINEAR mode, the algorithm will linearly interpolate between the two nearest OFF spectra. Default is LINEAR.

[MRTCAL\]MSETUP CALIBRATION PRODUCTS LINEAR|NEAREST|SPLINE

One set of calibration products Trec, Tcal, Tsys, precipitable water vapor, and zenithal opacity is computed per chunk, at the middle of the chunk (chunk bandwidth is ruled by MSET CALIB BANDWIDTH). In NEAREST mode, the intermediate values on the frequency axis use the nearest computed values (flat value per chunk/bandwidth, possible discontinuities). In LINEAR mode, the intermediate values are interpolated between two middle of chunks (possible discontinuity of the derivatives). In SPLINE mode, the intermediate values are cubic spline interpolated between two middle of chunks (ensuring continuity of the first and second derivatives). Default is LINEAR.

[MRTCAL\]MSETUP CALIBRATION SCAN NEAREST|LINEAR

Define how to select the calibration scan used calibrate science data. In NEAREST mode, the scan nearest in time will be used. In LINEAR mode, the algorithm will linearly interpolate calibration parameters between the two nearest calibration scans. Default is NEAREST.

[MRTCAL\]MSETUP CALIBRATION WATER SET
[MRTCAL\]MSETUP CALIBRATION WATER ELEMENT
[MRTCAL\]MSETUP CALIBRATION WATER FIXED Value

Define if the precipitable water vapour amount should be minimized or not:
- SET means one single value for the whole bandwidth,
- ELEMENT means one independent value per calibration bandwidth,
- FIXED Value will force the pwv amount to the given value without minimization.

Default is ELEMENT.

5.8.4 MSETUP SOLVE

[MRTCAL\]MSETUP SOLVE Keyword Switch

[MRTCAL\]MSETUP SOLVE POINTING Mode

Control how the individual pointing drift entries are combined before minimizing for the best solution. Possible modes are:
- ENTRY: unique entry i.e. no mixing,
- DRIFT: unique telescope and unique drift i.e. mixed directions (e.g. 0+180 degrees),
- FESBDIREC: unique frontend, unique sideband and unique direction i.e. mixed polarizations,
- FESBDRIFT: unique frontend, unique sideband and unique drift i.e.

```
       mixed polarizations and mixed directions,
  FEDIREC: unique frontend and unique direction i.e. mixed polarizations
       and sidebands.
  FEDRIFT: unique frontend and unique drift  i.e.  mixed  polarizations,
       sidebands, and directions,

            FE               POLAR  SB     DIREC  DRIFT
  ENTRY     090|250|230|345  H|V    L|U    +|-    Az|El
  DRIFT     090|250|230|345  H|V    L|U    Mixed  Az|El
  FESBDIREC 090|250|230|345  Mixed  L|U    +|-    Az|El  (default)
  FESBDRIFT 090|250|230|345  Mixed  L|U    Mixed  Az|El
  FEDIREC   090|250|230|345  Mixed  Mixed  +|-    Az|El
  FEDRIFT   090|250|230|345  Mixed  Mixed  Mixed  Az|El
```

5.8.5 MSETUP OUTPUT

```
    [MRTCAL\]MSETUP OUTPUT Keyword Switch

[MRTCAL\]MSETUP OUTPUT CALIBRATION NONE|SPECTRA|ASSOCIATED

If NONE, the calibration products (CalSKy, Tsys, ...) are not saved into
the current output Class file. If SPECTRA, they are exported as separat-
ed  Class observations. If ASSOCIATED, they are exported in a single ob-
servation with CalSky as the main spectrum and the other ones as Associ-
ated Arrays. Default is SPECTRA.

[MRTCAL\]MSETUP OUTPUT CHUNK ELEMENT|SET

Write  one  CLASS spectrum per CHUNK ELEMENT or per CHUNKSET. Default is
CHUNKSET.

[MRTCAL\]MSETUP OUTPUT INTEGRATION CYCLE|SUBSCAN|SCAN|*

Write one CLASS spectrum per cycle, or per subscan, or per  scan.  Wild-
card  *  will  automatically  select the best default choice from one of
these, depending on observing mode. All modes are not available  in  all
cases:
- TRACKED PSW: subscan CYCLE or SCAN (default)
- TRACKED FSW or WSW: phase CYCLE, SUBSCAN, or SCAN (default)
- OTF: phase CYCLE (default)
- POINTING: phase CYCLE, or SUBSCAN (default)
Default is *.

[MRTCAL\]MSETUP OUTPUT SPECTRA YES|NO

Indicate whether the calibration spectra should be written in the output
CLASS file. Default is YES, i.e., write the results.

[MRTCAL\]MSETUP OUTPUT USERSECTION YES|NO

Indicate whether the MRTCAL user section  should  be  written  for  each
CLASS  observation.  Default is NO, as this is very experimental to test
```

```
future use CLASS, i.e., we do not warrants its stability.

[MRTCAL\]MSETUP OUTPUT VODIR DirName

Define the target directory for VO XML files. The files will be  written
in:
  -  DirName/YYYYMMDD/scans/NN/  if this directory exists (with YYYYMMDD
    the scan date and NN the scan number), else
  - DirName if this directory exists, else
  - ./ (current working directory) as fallback.
Default for DirName is .

[MRTCAL\]MSETUP OUTPUT VOXML YES|NO

Should the VO XML file describing the calibration results be produced at
end  end  of  the  processing  of the calibration scan (one xml file per
backend). Default is NO.

[MRTCAL\]MSETUP OUTPUT WEIGHT YES|NO

Write or not the weight array (time*abs(fres)/Tsys**2, Tsys varying as a
function  of  frequency)  in the Class output file. If yes, the array is
saved as the Associated Array named W, shipped together with  the  usual
RY spectrum. Beware that twice more data will be written, which typical-
ly doubles the Class file size on disk. Default is NO.
```

5.8.6 MSETUP PIPELINE

```
    [MRTCAL\]MSETUP PIPELINE ONERROR CONTINUE|STOP

Set the error recovery mode of the PIPELINE command. In STOP  mode,  any
raised  error  will immediately stop the pipeline. In CONTINUE mode, the
pipeline will ontinue to iterate with the next index item when an  error
is raised on the current item. Default is CONTINUE.
```

5.8.7 MSETUP DEBUG

```
    [MRTCAL\]MSETUP DEBUG [Topic [Subtopic]] ON|OFF

Turn ON or OFF debugging messages for the given topics:
    - IMBFITS [ALLOCATION|OTHERS]: for the IMB-FITS reader,
    - INDEX [ALLOCATION|OTHERS]: for the index.mrt reading/writing,
    - SYNCHRONIZATION: time synchronization of the tables,
    - CALIBRATION [ALLOCATION|BOOKKEEPING|OTHERS]: for the calibration
  process.

Use MSET DEBUG ON|OFF to turn ON or OFF all the debug messages.
```

5.9 MUPDATE

```
    [MRTCAL\]MUPDATE

Update  in-place the relevant files from the internal buffers. This com-
```

mand makes sense after modifying the structures created by the command VARIABLE in WRITE mode:
- the index file is updated after MHEAD% has been modified,
- the IMB-FITS file is updated after IMBF%, SUBS%, or IMBDATA% have been modified (NOT YET IMPLEMENTED).

After this action, the files are updated on disk but not in memory. They should be re-read if needed.

5.10 MWRITE

[MRTCAL\]MWRITE

Convert the chunks as Class observations and WRITE them in the current Class output file. The chunks are written one by one or are stitched by chunk set before writing according to the MRTCAL\SET BYCHUNK rule.

5.11 PIPELINE

[MRTCAL\]PIPELINE [/SHOW [NONE|DONE|EMPTY|FAILED]]

Calibrate iteratively all the elements in the Current indeX.

The selection of the calibration scan paired to a science scan is automatic. It is selected according to the following rules:
- search in the past the nearest matching calibration scan, up to the limit defined by MSETUP CALIBRATION INTERVAL,
- if no matching calibration scan is found in the past, search for the nearest one in the future, up to the same limit. This is useful in case of offline reprocessing.

A calibration scan matches if:
- its calibration status is NONE or DONE. If NONE, it will be calibrated before use.
- its project name and backend are identical to the science scan.

If those rules do not meet his needs, one can use CALIBRATE /WITH to force pairing a calibration and a science scans.

The option /SHOW will update the Current indeX at the end of the process, and MLIST it. The optional argument indicates to populate CX only with the elements with the corresponding calibration status.

5.12 READ

[MRTCAL\]READ [FileName|FileNum|FIRST|LAST|NEXT|PREV] [/FILENAME] [/SUBSCAN Isub]
[MRTCAL\]READ ZERO

Load a file in the Mrtcal buffers:
- the 4 first HDU in the IMBFITS file (namely Primary, IMBF-scan, IMBF-frontend and IMBF-backend), and then read
- the 3 HDU IMBFITS-backendXXX, IMBFITS-antenna and IMBFITS-subreflector (this latter not implemented) for the subscan number Isub (de-

fault 1)
- the on-track part of the DATA column of the HDU IMBFITS-backendXXX is read (take care that it may be large).

Without argument, the command will try to read again the current filename. Else the command argument can be either
- the file name when the /FILENAME option is present,
- the file number (and optionally its version) in the Input index, of the form Num (or Num.Ver),
- the FIRST|LAST file in Current indeX,
- the NEXT|PREVIOUS file in Current indeX, considered from previous call to the command.

The reference file (for NEXT or PREVIOUS) in Current indeX is reset by the command MFIND. READ ZERO also reset the reference file and does not read any file.

5.13 SOLVE

[MRTCAL\]SOLVE Num[.Ver] [/WITH Num[.Ver]]

Solve the pointing/focus/skydip parameters for the file indentified by its Number. To do this, it first calibrate the pointing/focus/skydip data. The calibration scan is chosen implicitly by the command, except if explicitly specified through the option /WITH.

Boths file numbers can be added a Version number. If not given, the latest version is selected.

5.14 VARIABLE

[MRTCAL\]VARIABLE [*|What [READ|WRITE]] [/INDEX]

Create one or all the Sic structures mapping the Mrtcal internal buffers, namely:
- IMBF%PRIM, IMBF%SCAN, IMBF%FRONT, IMBF%BACK, IMBF%DEROT map the FITS HDUs for the scan,
- SUBS%BACKDATA, SUBS%ANTSLOW, SUBS%ANTFAST map the FITS HDUs for one subscan,
- IMBDATA%TIME and IMBDATA%IMBF% describe a portion of data of the DATA table,
- MHEAD%KEY, MHEAD%PRI, MHEAD%CAL, MHEAD%SCI, MHEAD%POI map the sections of the index entry which was read.

Default is *, i.e. the 4 structures will be created, but only one structure can be created with one of the keywords IMBF, SUBS, IMBDATA, MHEAD.

The structures can be created read-only (keyword READ) or read-write (keyword WRITE) for further update with the commands MUPDATE/MWRITE. Default is read-only.

Note that these variables are not updated when the buffers are changed. For safety they should be updated by recalling the command.

5.14.1 VARIABLE /INDEX

```
[MRTCAL\]VARIABLE /INDEX
```

Populate the MDX%HEAD% Sic structure with all the current index sections. Components are:

- MDX%HEAD%KEY
- MDX%HEAD%PRI
- MDX%HEAD%SCI
- MDX%HEAD%POI

Each individual component is appended a "number of entries" (i.e. size MFOUND) trailing dimension compared to the MHEAD% equivalent structure for one entry.