



Bugs: mkacisrmf

A list of bugs fixed in CIAO 3.4 is included at the end of this document.

Caveats

1. Lower limit on the energy bounds is 0.243 keV

mkacisrmf checks the energy bounds among the user's input grid, response file and gain file and then picks the output grid that fits all of them. Since the current calibration file does not extend below 0.243 keV, it is not possible for mkacisrmf to create an RMF with bounds lower than that energy.

2. The energy grid of the ARF and RMF files must be the same for use in XSpec. (25 Aug 2006)

Sherpa allows you to use different energy grids for your ARF and RMF files, but XSpec does not. Note that XSpec will still run if the grids do not match, but it issues a warning and sets all values in the ARF to unity (1).

1. Create the RMF first

Since mkacisrmf can change the requested grid to match the calibration data, create the RMF first and then use it to define the energy grid when creating the ARF. This will work for both `mkarf` and `mkwarf`:

```
unix% pset mkarf \  
      engrid="grid(sources_ciao.wrmf[cols ENERG_LO,ENERG_HI])"  
  
or  
  
unix% pset mkwarf \  
      egridspec="grid(sources_ciao.wrmf[cols ENERG_LO,ENERG_HI])"
```

2. Match an existing ARF

If the `specextract`, `psextract` or `acispec` scripts were used, you already have an ARF file for the data. Rather than remake both the RMF and ARF, get the grid information from the history in the ARF file:

```
unix% dmhistory acis_srcl.warf tool=all  
# dmhistory (CIAO 3.4): WARNING: Found "pixlib" library parameters  
  
# dmhistory (CIAO 3.4): WARNING: Found "ardlib" library parameters  
  
mkwarf infile="acis_srcl.[WMAP]" outfile="acis_srcl.warf" weightfile="acis_srcl.wfef"  
spectrumfile="" egridspec="0.3:9.5:0.01" threshold="0" feffile="CALDB"  
mskfile="" mirror="HRMA" detsubsysmod="" ardlibpar="ardlib" geompar="geom"  
clobber="no" verbose="2"
```

Bugs: mkacisrmf – CIAO 3.4

Your file may have been created with `mkarf` instead of `mkwarf`; the `dmhistory tool=all` will show the tool used in either case.

Use the `egridspec` value (or `engrid` in the `mkarf` case) as input for the `energy` parameter in `mkacisrmf`:

```
unix% pset mkacisrmf energy="0.3:9.5:0.01"
```

The [mkacisrmf analysis thread](#) has information on creating the RMF file.

3. **ERROR: Max egridspec energy=10 above max FEF energy=9.886**

`mkwarf` is required to compute and write a `weightfile` output file which contains FEF regions for use by `mkrmf`. If the energy range in the input RMF is greater than that in the FEF files, you get an error like:

```
ERROR: Max egridspec energy=10 above max FEF energy=9.886
```

Although the comparison to the FEF files is unnecessary in this case, there is currently no way to turn it off (e.g. set the `weightfile` to "NONE").

Workaround:

In order to avoid the error, it is necessary to define an energy range for `mkacisrmf` that falls within the boundaries of the FEF files, i.e. approximately 0.28 – 9.8 keV. See the [Creating an RMF to match an extracted spectrum section](#) of the [mkacisrmf analysis thread](#) for an example command.

Bugs

1. *Issue with the WMAP when the source is at a chip edge*

If the observation has large `dy` & `dz` offsets in the aspect solution file and they are quite variable during the observation, the tool will fail with a CALDB error. The large (and varying) offsets cause the mapping from DET to CHIP coordinates to fail and the tool cannot determine which response calibration file to use in creating the RMF.

2. *The tool may fail with messages of this form:*

```
# mkacisrmf (CIAO 3.4): WARNING: 2 CALDB files found. Using the first
# mkacisrmf (CIAO 3.4): ERROR: do not find extension related to
' RESP_TWEAK ' .
```

This occurs when the WMAP file is empty (i.e. all values are zero). You can open the WMAP in `ds9` to view it:

```
unix% ds9 src_pi.fits
```

where `src_pi.fits` contains a WMAP block:

```
unix% dmlist src_pi.fits blocks
-----
Dataset: mkacisrmf_bgfile_srcl.pi
-----
Block Name                                Type                                Dimensions
```

<i>Block</i>	<i>1: WMAP</i>	<i>Image</i>	<i>Int2(1024x1024)</i>	
Block	2: SPECTRUM	Table	4 cols x 1024	rows
Block	3: GTI3	Table	2 cols x 3	rows
Block	4: GTI6	Table	2 cols x 1	rows
Block	5: GTI7	Table	2 cols x 1	rows
Block	6: GTI2	Table	2 cols x 3	rows
Block	7: GTI1	Table	2 cols x 1	rows
Block	8: GTI0	Table	2 cols x 2	rows

3. *mkacisrmf generates incorrect RMF for non-standard channel grid (16 Aug 2006)*

e.g. channel="69:128:1". An example of a standard channel grid is channel="1:4096:1".

4. *The tool will overwrite files even if clobber=no (01 Dec 2006)*

Setting `clobber=no` does not prevent the tool from overwriting existing files.

Bugs fixed in CIAO 3.4

The following is a list of bugs that were fixed in the CIAO 3.4 software release.

1. *# mkacisrmf (CIAO 3.4): ERROR: Not matched ccd_id of wamp header to the pixlib calculated one. (07 Jul 2006)*

Note: This bug fix also resolves the problems seen when working with ACIS "Blank-Sky" Background Files obtained from the ACIS Calibration page. These files have a single ONTIME keyword in the file header, rather than one for each chip that was used in the observation.

This error occurs when there are places in the WMAP that map to a different CCD than the one you are using (i.e. the one the source is on). `mkacisrmf` needs ONTIME header keyword information that isn't available in the WMAP. This error is usually paired with the "do not find extension related to 'RESP_TWEAK'." message; the workaround given below will resolve both errors.

The data is mapping to a different chip because the SIM drift during an observation cannot be taken into account by `mkacisrmf` currently. Since this correction is not applied, the drift combined with the resolution of the WMAP leads to detector positions mapping to inaccurate chip coordinates.

The following CIAO commands may be used to determine which chips `mkacisrmf` is trying to process for the WMAP. The minimum and maximum detector coordinates are obtained with `dmstat`, then `dmcoords` is used to determine on what chip they fall. We introduce the same SIM drift seen in `mkacisrmf` by not providing the aspect solution to `dmcoords`. The region file is the one that was used in creating the spectrum and WMAP. **The syntax used here works in the `tcsh` shell; users of other shells may need to change the set or loop syntax to match their environment.**

```
unix% dmstat "acis_evt2.fits[sky=region(source.reg)][cols det]" sig- med- > /dev/null

unix% set min = `pget dmstat out_min | tr "," " "`
unix% set max = `pget dmstat out_max | tr "," " "`

unix% foreach xx ( $min[1] $max[1] )
foreach?   foreach yy ( $min[2] $max[2] )
foreach?   dmcoords acis_evt2.fits op=det detx=$xx dety=$yy verb=0
foreach?   pget dmcoords chip_id
foreach?   end
```

Bugs: mkacisrmf – CIAO 3.4

```
foreach? end
1
3
1
3
```

We know that the source of interest is on `ccd_id=3` (given by the `source.reg` region file). However, some positions map to `ccd_id=1`. This tells us that `mkacisrmf` needs the `ONTIME` header keyword for `ccd_id=1` in order to run correctly.

Workaround:

Use `dmhedit` to add the `ONTIME` keyword to the WMAP file. The `ONTIME` header keywords are numbered to match the `ccd_id`, so `ONTIME1` must be added for `ccd_id=1` (`ONTIME2` is `ccd_id=2`, and so on). Use the same `ONTIME` value that exists in the header:

```
unix% dmlist "source.pi[wmap]" header |grep ONTIME
0078 ONTIME                40129.3360019030 [s]          Real8          Sum of GTIs
0079 ONTIME3               40129.3360019030 [s]          Real8          Sum of GTIs

unix% dmhedit infile="source.pi[wmap]" filelist="" op=add key=ONTIME1 value=40129.3360019030 unit=s
```

This assumes that the WMAP is an extension in the spectrum file; your WMAP may be a separate image file.

Note that the RMF will contain data from the other CCD (`ccd_id=1`), even though we know only `ccd_id=3` data are in the file. This is the only workaround, as there is currently no way to correct for the SIM drift in `mkacisrmf`.