MIT
Center for Space Research Chandra X-Ray Center

MEMORANDUM
March 26, 2003

To:	K.Glotfelty/DS Pipeline/Tools Team Lead
From:	D.Huenemoerder/SDS Grating Analysis Scientist
Subject:	Required/Proposed Revisions for HETGS Masks

Summary of Changes Required or Proposed

There is one necessary change which will provide better background analysis support in HETGS spectra. A second revision is being considered, but requires more analysis. A third is needed by the second, but is generic enough that the funtionality should be provided.

1. Bug-fix for `tg_create_mask` parameters: The background binning region used by `tgextract` is larger than the range provided by the Level 1.5 spatial filter created by `tg_create_mask`.

   (***This simple parameter change should be implmented for CIAO 3***)

2. New `tgextract` mask region sign conventions to support large background binning regions. The definition of "_UP" is consistent with respect to `CHIPY`, but not with respect to the "outside" of the HETGS "X". Changes to the way the mask coordintes are defined for tgextract are being considered.

   (*This is post CIAO-3, and pending detailed specifications.*)

3. Customized polygonal masks to define large background regions. A Level 1.5 spatial filter which has larger extent on the "outside" if the "X" will permit large background regions. `tg_resolve_events` currently does not support polygons in the Level 1.5 region filter.

   (*This is for post CIAO-3 consideration.*)

1 Bug in `tg_create_mask`, `tgextract`

The `tg_create_mask` program creates a spatial filter for MEG and HEG regions from calibration data and an error-budget computation of the one-sigma cross-dispersion width (accounting for possible defocus and off-axis position), then scales by a factor. The parameter is `width_factor_hetg`, whose default value is 20. In `tgextract`, the default background parameters, `max_upbkg_tg_d` and `max_downbkg_tg_d`, extend further than the spatial filter. Figure 1 shows the error for HETGS. When the spectrum is binned by `tgextract`, the `BACKSCAL` values are determined from the parameters, not the data, and are thus wrong.

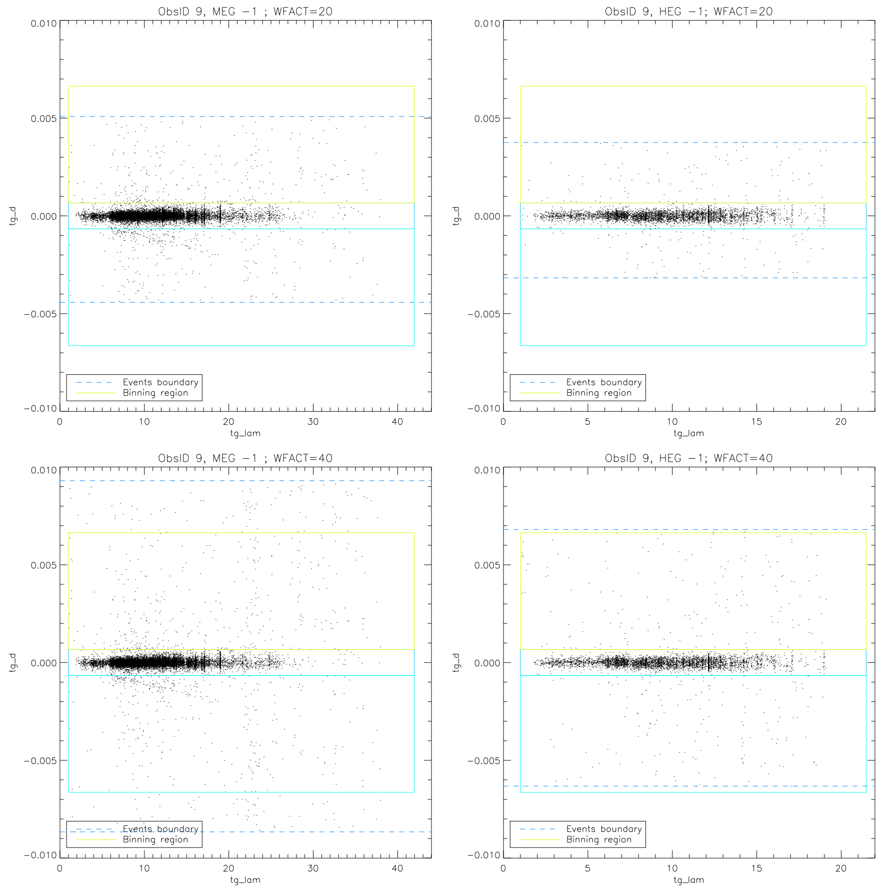

Figure 1: MEG (top left) and HEG (top right) events with the `tgextract` default binning region (solid borders) and event limit from the default spatial mask (`width_factor_hetg` = 20). The central region of events enclosed by solid lines is the default source binning region of `tgextract`. The bottom shows the binning region with a *revised* spatial mask (`width_factor_hetg` = 40).

My experiments have shown that a `width_factor_hetg` = 40 mitigates the problem without compromising event resolution at 6–7 keV. Larger `width_factor_hetg` causes the HEG and MEG regions to overlap too much such that valid short-wavelength events are not resolved. The bottom panels of Figure 1 show the output with the revised parameter.

As an additional safety mechanism, `tgextract` could record the maximum `tg_d` encountered in the background region and compare to the `max_upbkg_tgd` requested. This isn't foolproof because it depends upon the data, but a warning could be issued if the former is less than the latter.

2 Extraction Mask Coordinate Sign Revision for `tgextract`

Figure 2 shows current grating coordinate conventions.

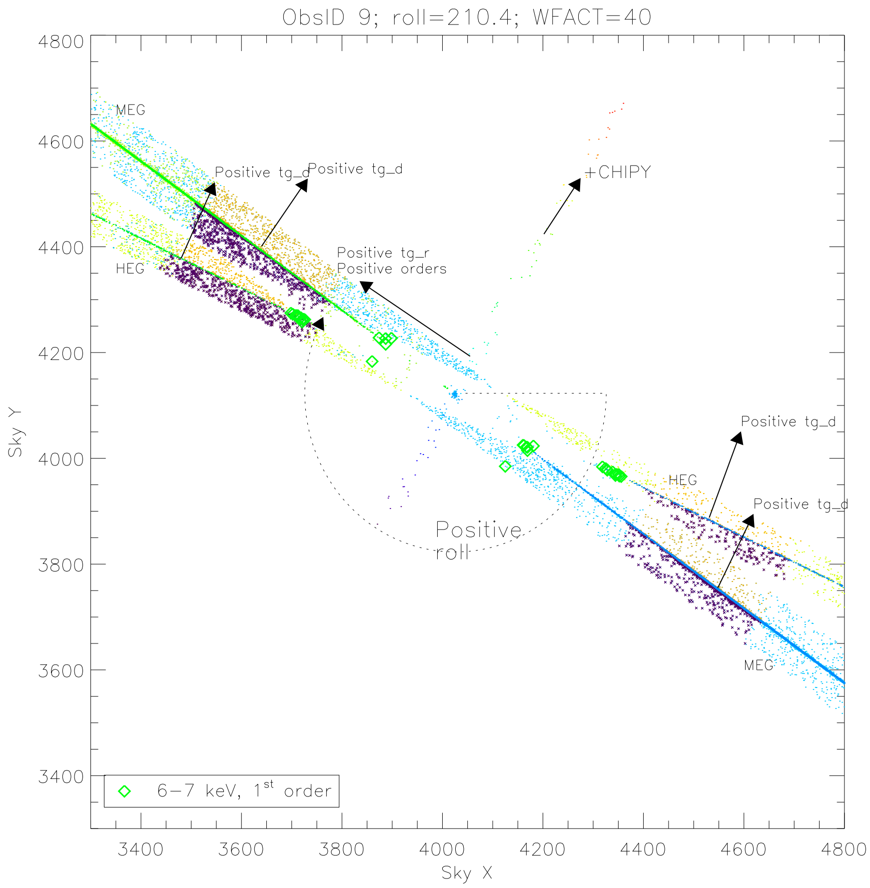

Figure 2: All the salient coordinate conventions are shown by the arrows and labels. (Color and symbol coding were used to interactively verify the conventions.) The important feature here is the direction of `tg_d` relative to the "X". For MEG negative (lower right) and HEG positive (upper left) orders, increasing positive `tg_d` points towards the opposite grating arm. The direction of increasing `tg_d` is defined as "_UP" in the `tgextract` background definition. (Note: I also checked observations with rolls in the first and second quadrants to make sure there were no unexpected sign changes with roll.)

The directions of `tg_d` are consistent with respect to `CHIPY`, but this is not optimum with respect to background binning. If the binning masks `tg_d` sign convention were such that positive `tg_d` were directed to the outside of the "X", then specification of the `max_upbkg_tg_d` in `tgextract` would be consistent among gratings and orders and would allow a large value without collision with the other arm.

It is not desirable to alter the `tg_d` coordinates themselves, since there are applications which depend on the current definition (extended sources; LETG spectra).

Detailed specifications for `tgextract` are pending.

3 Custom Level 1.5 Region File

To take advantage of the modified `tgextract` background region sign convention and ability of `tgextract` to apply background limits, we need to provide a wide spatial filter for use in `tg_resolve_events`. We cannot simply use large `width_factor_hetg` values because then the HEG and MEG arms would overlap for too much of their extent, and cause valid events to be unresolved. It is possible to define HEG-only and MEG-only masks and perform multiple passes to extract the spectra. But in the current system, it requires four passes to do each of the HEG and MEG positive and negative orders separately. With modified extraction region sign convention, it would still take two passes, one for MEG and one for HEG, with large rectangular masks.

With the support of polygon region types in `tg_resolve_events`, we could do this in one pass. Currently, `tg_resolve_events` only supports rotated boxes for the HEG and MEG filters.

A "windmill"-shaped mask with 8 points for each of MEG and HEG would suffice for providing narrow limits on the "`X`" inside, and wide limits on the outside. An example is shown in Figure 3.

I have written S-Lang scripts which will read a current Level 1.5 mask file with standard `rotbox` regions, replace these with `polygon` regions, and write a new region file. The polygons are defined in un-rotated coordinates, then rotated by the `ROLL_NOM` of the observation (from the input file) and clocking angle of the grating, then translated to the zero-order centroid (also read from the input region file itself). A simple template is defined which is independent of the observation, and which will be valid for most HETGS observations which are done near the nominal aim point. The S-Lang script, `tgmodmask.sl` defines the functions, and `tgmodmask` is an executable file which invokes `isis-script`.

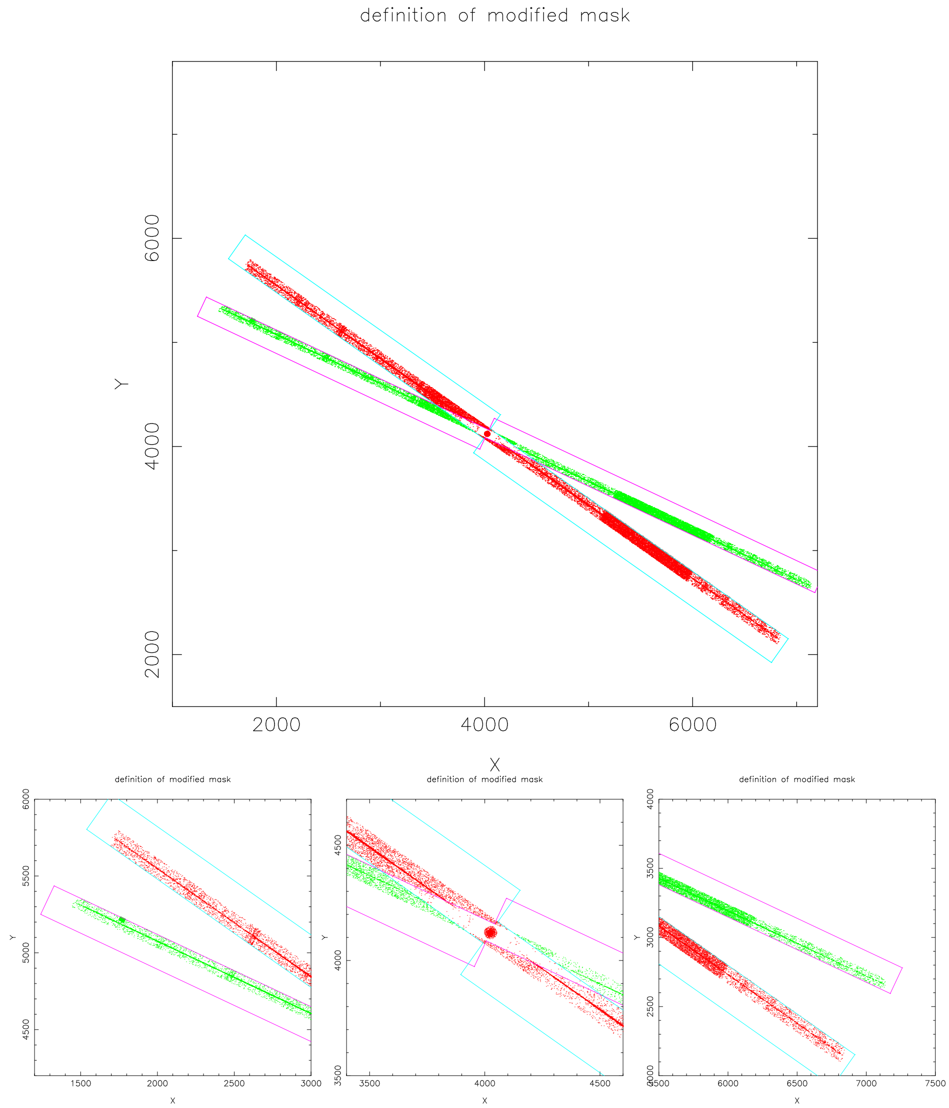

Figure 3: “Windmill” mask definition. The top shows the whole field for the same event file of Figure 2. The bottom shows detail for ends and center of the spectrum. The red are MEG and green are HEG events, filtered by the current `tg_create_mask` rectangular masks. The new definition maintains the inner boundary such that short wavelength events are still resolved at Fe K.

The core mask specification (used in the figures) is as follows:

```
define hetg_region(type)
{
% define a "windmill" shaped polygonal region, centered on 0,0,
% with no rotation.
  variable p = struct{ x, y } ;
  variable left, right, hi, lo;

  % ad hoc, based on mask w/ width_factor=40
  % and doubling the extent on the "outside" of the "x"
  %
  left = -3500.;
  right = 3000.;

  switch ( type )
  {
    case MEG:
    lo = 56.;
    hi = 4*lo;
  }
  % HEG is MEG flipped in y, and a bit narrower (astig less):
  % but astigmatism really irrelevant at this level. Keep for
  % consistency w/ default masks.
  {
    case HEG:
    lo = -41.;
    hi = 4*lo;
  }
  {
    verror("Invalid grating type index, %d\n", type );
    return NULL;
  }
  % define polygon vertices, traverse ccw:
  %
  p.x = [ 0, left, left,   0,   0, right, right, 0,   0 ];
  p.y = [hi,   hi,  -lo, -lo, -hi,   -hi,    lo, lo, hi ];

  return p;
}
```

4 Background Level

It should be noted that HETGS background rejection via order-sorting and spatial filtering is quite high. The background support here will mainly benefit very faint source observations. The background level estimated from 155ks of Capella spectra is approximately 2.2 counts $\mathrm{ks}^{-1}\,\text{Å}^{-1}$ (in a single order). Furthermore, some of these counts are really source photons from the frame-shift streak, in the proportion of 1.3% (0.041s/3.2s), but in any bin it is source-model dependent because it combines both MEG and HEG counts. Some other fraction comes from the PSF wings. (This is why the issue is hedged in the help files, as to the background region counts being "related to the background".)