URL: http://cxc.harvard.edu/sherpa3.4/bugs/ut_eflux.html
Last modified: 21 September 2006

Utility Bugs: eflux

Bugs

1. ***In ARF−less analysis, it is possible to calculate fluxes but not flux densities.***

```
sherpa> dataspace (0.1:10:0.01) histogram
sherpa> paramprompt off
Model parameter prompting is off
sherpa> source = powlaw1d * xsphabs
sherpa> eflux (0.5:2.0)
Flux for source dataset 1: 1.92335e−09 counts
sherpa> eflux (0.5)
Error: applying the current filter yields an empty dataset.
```

create a dummy ARF filled with zeroes:

```
sherpa> erase all
sherpa> x = [0.1:10.0:0.01]
sherpa> arf = struct { _filename, ENERG_LO, ENERG_HI, SPECRESP }
sherpa> arf._filename = "a1"
sherpa> arf.ENERG_LO = typecast( x[[:−2]], Float_Type );
sherpa> arf.ENERG_HI = typecast( x[[1:]], Float_Type );
sherpa> arf.SPECRESP = arf.ENERG_LO * 0.0f;
sherpa> () = load_arf("a1",arf )
sherpa> instrument 1 = a1

(continue from above)
```

Note that this ARF adds no value to the session since it equals 0.0 at each energy; it is only used to allow the flux−density to be calculated.

2. ***Problems calculating flux and counts over a range.***

   The behavior described in this bug applies to the following commands:

   - `[B]EFLUX`,
   - `[B]PFLUX`,
   - `[B]DCOUNTS`,
   - `[B]MCOUNTS`,
   - and the corresponding "`get`" commands (e.g. `get eflux`).

   Given x > 0:

   1. "`EFLUX (0:x)`" returns the same value as "`EFLUX`" (i.e. the flux over the full range of the dataset), when it should return the flux integrated from the start of the dataset to x.
   2. "`EFLUX (x:0)`" returns the same value as "`EFLUX (x)`." This should return an error since x > 0, so the range isn't sensible.

Workaround:

replace 0 with a small decimal, e.g. 0.001. Then case 1 gives the correct answer (flux from the start of the dataset to x), and case 2 gives 0 (which at least tells you that you did something wrong).

URL: http://cxc.harvard.edu/sherpa3.4/bugs/ut_eflux.html
Last modified: 21 September 2006