Initial Steps

1. Ensure the latest SAS and CCF are properly installed. Instructions are available at:
http://xmm.esac.esa.int/sas/
2. Download ODF data from archive and untar.
3. It may be necessary to rename files (if they end in .FTZ) and/or uncompress them (if they end in .gz). Assuming you are using csh or tsch:

```
> foreach f (`ls *.FTZ`)
foreach? mv $f $f:r.FIT.gz
foreach? end
> gunzip *.gz
```

4. Set SAS_ODF by 'cd'ing to the ODF directory and running (note back-quotes):

```
> setenv SAS_ODF `pwd`
```

5. Create and enter working directory **repro** and extract a calibration index file by running:

```
> cifbuild
> setenv SAS_CCF `pwd`/ccf.cif
> odfingest
> setenv SAS_ODF `pwd`/`ls -1 *SUM.SAS`
```

Reprocessing

1. In **repro**, run the standard processing chains for MOS, PN, RGS, and OM respectively:

```
> emchain >& emchain.log
> epchain >& epchain.log
> rgsproc >& rgsproc.log
> omichain >& omichain.log (IMAGE data)
> omfchain >& omfchain.log (FAST data)
> omgchain >& omgchain.log (GRISM data)
```

Note: if SAS is installed but a "command not found" occurs, ensure /usr/local/bin/perl exists, a SAS requirement. Some other options for special cases, such as selecting the n-th MOS exposure or PN timing or burst mode:

```
> emchain instruments=M1 exposure=n
> epchain datamode=TIMING (or BURST)
```

3. Optional but convenient soft links useful if there is only one exposure per observation:

```
> ln -s P*M1*MIEV*.FIT m1_evt1.fits
> ln -s P*M2*MIEV*.FIT m2_evt1.fits
> ln -s P*PN*PIEV*.FIT pn_evt1.fits
```

In timing or burst mode, replace PIEV with TIEV.

Lightcurves

Examples for EPIC MOS for both whole field and individual source lightcurves; others are similar.

```
> evselect table=m1_evt1.fits withrateset=yes \
   rateset=m1_ltcrv1.fits maketimecolumn=yes \
   timecolumn=TIME timebinsize=10 makeratecolumn=yes
> evselect table=m2_evt1.fits withrateset=yes \
   rateset=m1_ltcrv_src1.fits maketimecolumn=yes \
   timecolumn=TIME timebinsize=10 makeratecolumn=yes \
   filtertype=expression \
   expression='((X,Y) in CIRLE(24870.5,24149.5,2000))'
```

To make filters based on lightcurves (see below), use:

```
> tabgtigen table=m1_ltcrv2.fits:RATE \
    gtiset=m1_rate.gti expression="RATE<0.35" \
    timecolumn=TIME
> tabgtigen table=m1_ltcrv1.fits:RATE \
    gtiset=m1_time.gti expression="TIME>162594000" \
    timecolumn=TIME
```

Automated lightcurve-based filtering for MOS data, useful for diffuse emission, can be done via:

```
> espfilt eventset=m1_evt1.fits
```

Filtering

To filter out bad events in EPIC MOS & PN (filenames from step 3 of Reprocessing). The `expression` value must be given on a single line. Shown here is the filter for IMAGE mode data; for recommended filters to data obtained in different readout modes, see the SAS Users' Guide and the EPIC Calibration document.

```
> evselect table=m1_evt1.fits withfilteredset=yes \
    filteredset=m1_evt2.fits filtertype=expression \
expression='(PATTERN<=12)&&#XMMEA_EM&&(PI in
[200:15000])' keepfilteroutput=yes
> evselect table=pn_evt1.fits withfilteredset=yes \
   filteredset=pn_evt2.fits filtertype=expression \
   expression='(FLAG==0)&&(PATTERN<=4)&&(PI in
[200:15000])' keepfilteroutput=yes
```

Recommended thresholds to select non-flaring times from MOS and PN lightcurves are:

MOS: `RATE<0.35` for `'(PATTERN==0)&&(PI>10000)'`

PN: `RATE<0.4` for `'(PATTERN==0)&&(PI in [10000:12000])'`

To apply a filter, use:

```
> evselect table=m1_evt2.fits withfilteredset=yes \
    filtertype=expression keepfilteroutput=yes \
    expression='GTI(m1_rate.gti,TIME)' \
    filteredset=m1_flt_evt2.fits
```

Images

To create an EPIC MOS1 image in sky coordinates and exposure map (in units of seconds) between 2000-8000 eV. All EPIC PI channels are in units of eVs.

```
> evselect table=m1_flt_evt2.fits \
   withimageset=yes imageset=m1_img.fits \
   xcolumn=X ycolumn=Y imagebinning=binSize \
   ximagebinsize=32 yimagebinsize=32 \
   filtertype=expression \
   expression='(PI in [2000:8000])'
> atthkgen atthkset=atthk.fits
> eexpmap imageset=m1_img.fits pimin=2000 \
   pimax=8000 attitudeset=atthk.fits \
   eventset=m1_flt_evt2.fits \
   expimageset=m1_expmap_2.0keV.fits
```

To make attractive multicolor images from EPIC data, see the `images` script at:
`http://xmm.esac.esa.int/external/xmm_science/gallery/utils/images.shtml`

Source Finding

To create a list of sources using 2 bands for a PN thin filter observation and display the results. The appropriate ecf values vary with filter and detector; for more information, see:
`http://xmmssc-www.star.le.ac.uk/Catalogue/2XMM/UserGuide_2xmm.html`. The image files must be created beforehand using the method shown in the Images section above.

```
> atthkgen atthkset=atthk.fits
> edetect_chain eventsets=pn_flt_evt2.fits \
   imagesets='image_b1.fits image_b2.fits' \
   pimin='500 2000' pimax='2000 4500' \
   ecf='6.816 2.054' attitudeset=atthk.fits \
   eboxl_list=eboxlist_l.fits esen_mlmin=10 \
   eboxm_list=eboxlist_m.fits \
   esp_nsplinenodes=16 eml_list=emllist.fits
> srcdisplay boxlistset=emllist.fits \
   imageset=pn_image.fits sourceradius=0.01
```

RGS Products

To make a lightcurve of the RGS1 CCDs:

```
> ln -s *R1*EVENLI*.FIT r1_evt1.fits
> ln -s *R1*SRCLI*.FIT  r1_src1.fits
> rgslccorr evlist=r1_evt1.fits \
   srclist=r1_src1.fits timebinsize=100 \
   outputfilename=r1_ltcrv.fits
```

To make an image of the RGS1 CCDs:

```
> evselect table=r1_evt1.fits withimageset=yes \
   imageset=r1_image.fits xcolumn=BETA_CORR \
   ycolumn=XDSP_CORR
> evselect table=r1_evt1.fits withimageset=yes \
   imageset=r1_orderim.fits xcolumn=BETA_CORR \
   expression='region(r1_src1.fits:RGS1_SRC1_
SPATIAL,BETA_CORR,XDSP_CORR)' ycolumn=PI \
   withyranges=yes yimagemin=0 yimagemax=3000
> rgsimplot spatialset=r1_image.fits \
   endispset=r1_orderim.fits device=/VCPS \
   srclistset=r1_src1.fits plotfile=r1_image.ps
```

To make RGS1 response files; RGS2 is similar:

```
> ln -s *R1*SRSPEC1*.FIT r1_o1.pha
> ln -s *R1*SRSPEC2*.FIT r1_o2.pha
> ln -s *R1*BGSPEC1*.FIT r1_o1_bck.pha
> ln -s *R1*BGSPEC2*.FIT r1_o2_bck.pha
> rgsrmfgen rmfset=r1_order1.rsp \
   evlist=r1_evt1.fits withspectrum=yes \
   spectrumset=r1_o1.pha order=1 \
   srclist=r1_src1.fits:RGS1_SRC1_SPATIAL
> rgsrmfgen rmfset=r1_order2.rsp \
   evlist=r1_evt1.fits withspectrum=yes \
   spectrumset=r1_o2.pha order=2 \
   srclist=r1_src1.fits:RGS1_SRC1_SPATIAL
```

To make fluxed spectra from RGS data:

```
> rgsfluxer file=rgs_fluxed1.fits \
   pha='r1_o1.pha r2_o1.pha' \
   bkg='r1_o1_bck.pha r2_o1_bck.pha' \
   rmf='r1_order1.rsp r2_order1.rsp'
```

OM Products

OM data as delivered can usually be used without reprocessing or further work. The tasks `omsource`, `omphotom`, and `omgsource` handle special cases of crowded fields, undetected sources, and overlapping grism spectra, respectively. Standard OM filenames include:

1. Photometry data: `*SWSRLI*` (one exp.), `*OBSMLI*` (all)
2. Light curves (in FAST mode): `*TIMESR*`
3. Spectra (in GRISM mode): `*SPECTR*`

EPIC Products

Shown for PN and MOS1 in IMAGING mode only. Information on timing and burst mode data is in the SAS Guide. Use ds9 or fv to find region coordinates.

```
> evselect table=pn_flt_evt2.fits \
   withspectrumset=yes spectrumset=pn_src.pi \
   energycolumn=PI spectralbinsize=5 \
   withspecranges=yes specchannelmin=0 \
   specchannelmax=20479 filtertype=expression \
   expression='((X,Y) in CIRCLE(26863.5,27700.5,937))
&&(FLAG==0)' keepfilteroutput=yes \
   withfilteredset=yes filteredset=pn_filtered.fits
 > evselect table=m1_evt2.fits \
    withspectrumset=yes spectrumset=m1_src.pi \
    energycolumn=PI spectralbinsize=15 \
    withspecranges=yes specchannelmin=0 \
   specchannelmax=11999 filtertype=expression \
   expression='((X,Y) in CIRCLE(24870.5,24149.5,1000)
)&&(FLAG==0)' withfilteredset=yes \
   keepfilteroutput=yes filteredset=m1_filtered.fits
```

Use similar commands about a source-free circle on the same MOS CCD to get a background spectrum. To scale source spectra properly (do same for bkg):

```
> backscale spectrumset=m1_src.pi \
   badpixlocation=m1_evt2.fits
```

Check pileup:

```
> epatplot set=m1_filtered.fits useplotfile=yes \
   plotfile=m1_epat.ps
```

Generate response (rmf) and effective area (arf) files for a point source including the encircled energy (ee) correction but not out-of-time events (oot):

```
> rmfgen rmfset=m1_src.rmf spectrumset=m1_src.pi
> arfgen arfset=m1_src.arf spectrumset=m1_src.pi \
   withrmfset=yes rmfset=m1_src.rmf \
   badpixlocation=m1_evt2.fits
```

One-step method to do all of the above:

```
> especget filestem=mysrc table=m1_evt2.fits \
   srcexp='((X,Y) IN CIRCLE(26863.5,27700.5,800))' \
   backexp='((X,Y) IN CIRCLE(25800.5,25700.5, 1600))'
```

For More Info

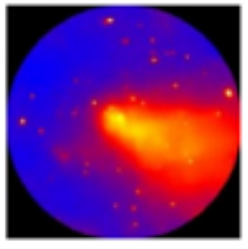
Home page: http://xmm.esac.esa.int/ (HOME)
Top level SAS page: HOME/sas/
For calibration data: HOME/external/xmm_sw_cal/calib/
For threads: HOME/sas/current/documentation/threads/

XMM-Newton Data Analysis:

a Brief Guide

US XMM Guest Observer Facility

NASA/Goddard Space Flight Center
Greenbelt, MD 20771

http://heasarc.gsfc.nasa.gov/docs/xmm/

v. 2.0 for SAS 11.0