A SAS/xmmselect/evselect versus XANADU/xselect/fselect comparison

M. Cappi (ITeSRE – CNR; Bologna)

Outline:

Comparison of extracting standard products through

1) Command lines (fselect vs. evselect)

2) GUIs Tools (flaunch vs sas)

3) Standard User–Interface Tools
   (xselect vs. xmmselect)

XANADU–ftools: V5.0.1
SAS: V5.0 (Dec. 3rd)
### Extracting Lightcurves

<table>
<thead>
<tr>
<th><strong>Left: XANADU</strong></th>
<th><strong>Right: SAS</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Commands</strong></td>
<td><strong>Commands</strong></td>
</tr>
<tr>
<td><strong>parameters</strong></td>
<td><strong>parameters</strong></td>
</tr>
<tr>
<td><strong>actions</strong></td>
<td><strong>actions</strong></td>
</tr>
</tbody>
</table>

- **Left: XANADU**
  - `fselect`  
    - PI, etc.  
    - `.evt` $\rightarrow$ `.evt`
  - `lcurve`  
    - binsize etc.  
    - `.evt` $\rightarrow$ `.qdp`

- **Right: SAS**
  - `evselect`  
    - PI, binsize, etc.  
    - `.evt` $\rightarrow$ `.ds`
  - `dsplot`  
    - `.ds` $\rightarrow$ `.agr`

### Problems

- **Left: XANADU**
  - **Wrong Exposure**
    - (~2% high) (=TSTOP–TSTART)

- **Right: SAS**
  - **Grace less known/used than QDP**

### Time Consumed

- **Left: XANADU**
  - (User–CPU) Time consumed: ~1.4 s

- **Right: SAS**
  - (User–CPU) Time consumed: ~11 s
**Extracting Images**

**Left: XANADU**

- **Commands**
  - Parameters
  - Actions

**Right: SAS**

- **Commands**
  - Parameters
  - Actions

---

**fselect**

- PATTERN, PI, etc.
- \( \text{.evt} \rightarrow \text{.evt} \)

**evselect**

- PATTERN, PI, binsize, etc.
- \( \text{.evt} \rightarrow \text{.ds} \)

**f2dhisto**

- binsize etc.
- \( \text{.evt} \rightarrow \text{.ima} \)

**ds9**

- \( \text{.evt} \rightarrow \text{.ds} \)

**ds9**

---

**(User–CPU) Time consumed:**

- ~5 s

**Problems:** None found

---

**(User–CPU) Time consumed:**

- ~19 s

**Problems:** None found
Left: XANADU

Commands

parameters

actions

Right: SAS

Commands

parameters

actions

Extracting Spectra

\[ \text{xselect @script} \quad \text{PATTERN, PI, region, etc.} \]
\[ \text{fselect} \quad \text{.evt} \rightarrow \text{.pi} \]
\[ \text{fmodehead} \quad \text{TLMAX1} \]
\[ \text{pi} \rightarrow \text{.pi} \]
\[ \text{grppha} \]

(User–CPU) Time consumed:

\[ \sim 30 \text{ s} \]

**Problems:** counts \(\sim 1\%\) larger due to different region extraction algorithm

\[ \text{evselect} \quad \text{PATTERN, PI, region, binsize, etc.} \]
\[ \text{.evt} \rightarrow \text{.ds} \]
\[ \text{fmodehead} \quad \text{TLMAX1} \]
\[ \text{.ds} \rightarrow \text{.ds} \]
\[ \text{grppha} \]

(User–CPU) Time consumed:

\[ \sim 18 \text{ s} \]

**Problems:** BACKSCAL wrong (always 3.6e5)
\( \Rightarrow \) Similar capabilities... (e.g. both allow scripts); but of course SAS is mandatory for XMM
Currently, two advantages of `xselect`:
already known to most users + allows easy scripts
The Standard User–Interface/interactive Tools is the most important for most GOs

<table>
<thead>
<tr>
<th></th>
<th><strong>xselect</strong></th>
<th><strong>xmmselect</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Beginners</td>
<td>Typing is not easy, but already known + easy to &quot;save&quot;</td>
<td>Clicking is easy, but not known + impossible to &quot;save&quot;</td>
</tr>
<tr>
<td>Typical</td>
<td>Scripts are user–friendly (extract curve, plot image...)</td>
<td>Scripts are user–UNfriendly (#!/usr/bin/sh ...+ very long strings of commands)</td>
</tr>
<tr>
<td>Experts</td>
<td>Scripts are powerfull + commands are fast</td>
<td>Scripts are powerfull but commands are slower</td>
</tr>
</tbody>
</table>

⇒ My suggestion is:
We shall really add "xmmselect logging" (for beginners) or/and "xmmselect scripting" (for typical&experts) output or/and make file scripts (for standard analysis) available to GO community.