SUMER Log Files

John Mariska
Naval Research Laboratory
(Version 2.0, 17 October 1996)

1 Overview

The SUMER instrument is currently producing FITS data files at the rate of several thousand per month. This production rate results in two serious problems. First, since only a limited number of files can be stored on active disk space, many older files must be archived to secondary media before investigators have had an opportunity to examine fully their contents. Second, the large number of files residing on disk at any one time makes it difficult for investigators to identify those data sets that contain observations in which they might be interested.

These problems should eventually be overcome, when a sophisticated database that details all the SUMER observations becomes available. In the meantime, there is a need for a simple database to allow investigators to locate data sets of interest. This document describes a simple database consisting of monthly log files containing minimal data on the contents of each SUMER data file and a small, but easily extensible, set of IDL routines for accessing the database to locate observations of interest. To avoid possible collisions with other IDL routines in the SUMER and SOHO software trees, all the IDL routines associated with the database begin with the prefix `slog_`.

2 Structure and contents of the database

Rather than design a complex database and extensive routines to access it, I have instead taken the approach of trying to place the key information about each SUMER FITS file into a form that reasonably competent IDL users can manipulate to obtain any information that they desire. Thus, the approach I have selected is to define an IDL structure that contains in reasonably compact form all the essential information the I think the user might need. Reasonably competent IDL user therefore currently means any IDL user who knows how to extract information from an array of structures.

The basic file for the database consists of a monthly log file containing an array of these structures. This file is created using the IDL procedure `jtm_sumlog`, which writes the data as an unformatted xdr stream file under VMS. This assures that the log files can be read by any machine which runs

IDL. Each log file is named `yymm.log`, where `yy` is the last two digits of the year and `mm` is the month number (00 to 12).

Each log entry currently contains the following elements:

```
filename : string ; name of the file
dstr     : string ; date/time of observation
obs_seq  : string ; observing sequence
detector : string ; detector (A or B)
slit     : string ; slit as a number
object   : string ; object--often wrong
sci_obj  : string ; science objective--often absent
study_nm : string ; study number
scientis : string ; scientist associated with obs
popudp   : fix    ; pop or udp number
ins_x    : float  ; instrument x coordinate
ins_y    : float  ; instrument y coordinate
xcen     : float  ; x coordinate of center of slit
ycen     : float  ; y coordinate of center of slit
exptime  : float  ; exposure time in seconds
wavemin  : float  ; minimum wavelength
wavemax  : float  ; maximum wavelength
next     : fix    ; number of binary extensions
```

Once one has read a log structure into memory, the simplest way to remember what is in it is to use the IDL `help` command:

```
IDL> help, /str, log_str
```

3 IDL access routines

All the IDL routines described here have minimal, but growing, standard IDL documentation headers. Thus, either the `doc_library` or `xdoc` routines can be used to check for calling syntax and to examine the capabilities of any new routines added to the directory. Generally, if the routine accesses the log structure, that variable will be the first parameter in the calling sequence.

3.1 Log file and IDL routine location

On the SUMER computers the current working version of the database routines described here and the log files they access are located in `science:[software.mariska.slog]`

3.2 Initial access

To access information from the log files, a user must first read the log file for the month in which they he or she is interested. This is done with the procedure slog_rdlog. They syntax is

```
IDL> slog_rdlog, "log_file", log_str
```

Here log_file is the full name of the monthly log file (including path if you are not in the log directory) and log_str is the name you wish to assign to the array of structures which the procedure returns. Note that, except for possible memory limitations, there is nothing to prevent a user from concatenating several months of these arrays together into a single array for searching. This feature may, in fact, be provided as the software evolves.

3.3 Information extraction

Most of the information in the log files is already in a form which is directly useful. Thus the most powerful information extraction routine is probably the IDL where function. For example, to find all the observations at Sun center for a particular month, one could type

```
IDL> ss = where(log_str.xcen eq 0.0 and log_str.ycen eq 0.0)
```

The array ss would then contain the indices of all those structure elements that satisfy the condition that both xcen and ycen be 0. To print a list of file names that matched the criteria given above, one would then type

```
IDL> print, slog_get_file(log_str(ss))
```

If one wanted to just see what was close to Sun center, a better tactic would probably be to use xcen and ycen to construct the radius vector from Sun center and then look for small values of that parameter. Similarly, such a quantity could be used to search for observations with the slit center near or above the limb.

Routines are currently provided to extract some of the information in the files that requires additional processing.

```
slog_get_day - return array of days
slog_get_time - return array of times 3 entries per structure
    (hh mm ss)
slog_get_file - return array of file names (without
```

```
    version numbers)
slog_get_sci - return array of array elements that contain
    given science objective (case sensitive, but partial
    names ok)
slog_get_scientist - return array of elements of structure
    that contain given scientist name (case sensitive, but
    partial names ok)
slog_get_sn - return array of elements of structure that
    contain given study name (case sensitive, but partial
    names ok)
```

3.4 Information display

In trying to use the information in the log files, I have found that it is often useful to get a visual impression of where SUMER was pointed at a particular time, or for all times on a given day. The procedure `slog_splot` will draw a circle representing the Sun as viewed from L1 and place lines indicating the position of the SUMER slit for any structure elements it is provided. For example, to view all the slit positions for a given day, one might use the where function in conjunction with the function `slog_get_day` to obtain an array containing the indices of all the structures in a structure array that match a given day. This information can then be fed to `slog_splot` to display the slit locations. The two lines of IDL commands to accomplish this would be:

```
IDL> ss = where(slog_get_day(log_str eq day))
IDL> slog_splot, log_str(ss), /circ
```

4 Additional developments

Currently, the log files contain only information from the main FITS header of each data file. As I more fully understand the information contained in the binary extension headers, additional elements will be added to the structure definition given above and this document will be modified to include them. I am currently working on adding more information about wavelengths covered in each file and on a visual display of the coverage. The current version of this document resides in that same directory as the log files and IDL access routines as `sumlog.tex` and `sumlog.ps`