MULLARD SPACE SCIENCE LABORATORY
UNIVERSITY COLLEGE LONDON
L.K.Gilbert

# SWIFT-UVOT STAR CATALOGUE PREPARATION SOFTWARE

| Issue | Date | Comments |
|---|---|---|
| Draft | Nov 2001 | Draft version for comment |
| First release | May 2002 | |
| | | |
| | | |
| | | |

## 1. OVERVIEW

The Swift UVOT star catalogue preparation software is a suite of programs that convert a star catalogue into a format that can be uplinked to Swift and then read by onboard safing software. The initial catalogue will be provided – see documents swift_catalogue.txt and swift_catalogue.extras.

The onboard organization and data compression scheme are outlined in section 2.

The scheme used to label areas of sky is outlined in section 3.

The software suite consists of six main programs

- converter
- pointers
- formatter
- addendum
- locater
- disassembler

and three subroutines and a header file that they all share

- assignmag
- assignpos
- crc_opt
- starcat

A makefile is also included, although altering the code should never be necessary. These programs are detailed in section 4.

With this suite it is possible to regenerate or update the initial catalogue, or create a new catalogue from another dataset.

An archive will also be provided, containing the catalogue from which the onboard catalogue was derived and copies of all intermediate files and data files, as detailed in section 5.

A set of procedures for alteration or recreation of the onboard catalogue, and identification of stored sources is given in section 6.

## 2. THE ONBOARD CATALOGUE

The catalogue itself is divided into three parts in memory. These are

- The pointer table
- The main catalogue
- The addendum

They follow on from each other sequentially and are created separately. The pointer table allows quick access to catalogue data, and the addendum is an area left blank in case any sources need to be added to the catalogue after launch, for example if a supernova occurs during the mission.

The pointer table holds pointers to each sky area and the start of the addendum. It holds a preliminary set of pointers which each point to the start of a pointer list for a declination band, and then the pointers to each sky area in that band are listed sequentially. The final word in the table is a CRC value. This scheme allows rapid access to the main catalogue.

The main catalogue is split into 2524 sky areas, in 44 declination bands. See section 3 for the sky area naming convention. Within each area the position of each source is stored to +/- half an arcminute accuracy in the RA and declination axes, relative to the origin of the area, along with each source's associated magnitude information. If two or more sources are deemed coincident then their magnitudes are combined and the combined information only is stored. Each sky area is followed by a CRC value for memory corruption checking. For more details on the information stored for each source, see sections 2.1, Position Information and 2.2, Magnitude Information.

The addendum is a 304 word long area of memory, which can store up to 100 extra sources. At minimum it will consist of a marker to show that there are no more sources stored, and the remaining memory zero filled, except for the last word, which is a CRC value. If a source is added to the addendum then it will be put at the beginning of the area, and followed by the marker. Any new source will not necessarily be added to the addendum – if it is coincident with a source already in the main catalogue then its magnitude information can be combined with that of the main catalogue source instead.

### 2.1 Position Information

It was required that the position of any source in the *field of view* be stored to +/- half an arcminute accuracy. This implies that declinations must be stored to within +/- half an arcminute, but that the accuracy in right ascension varies with declination. This is because lines of equal right ascension (meridians) converge at the poles. A scheme was chosen to divide the sky into 2524 areas so that within each area a 255 by 255 grid would provide the necessary accuracy in both axes. The position of each source is then taken as the nearest gridpoint to it, and the information stored in the catalogue is the coordinate pair (RA, Dec) of the gridpoint. See section 3, Sky Area Naming Convention, for details of the division of the sky into areas.

## 2.2 Magnitude Information

Magnitude information about each source is used by onboard safing software to protect the UVOT detector. This software uses visual magnitude and colour index as indices to a count rate table, and uses the count rate gained to decide how long, if at all, an observation at this pointing can safely continue. The visual magnitude and colour index of each source is assigned a code.

### 2.2.1 Visual Magnitude Code

If the visual magnitude (Johnson V magnitude) of the source is less than –1.4 (ie extremely bright) then it is assigned code 0.

From there on magnitudes are gathered into bands, each 0.1 of a magnitude wide, and each band is assigned a code number, ascending from 1 to 134. So for example;
A magnitude greater than or equal to –1.4 but less than –1.3 has code 1.
A magnitude greater than or equal to –1.3 but less than –1.2 has code 2.
A magnitude greater than or equal to 11.9 has code 134.
Sources fainter than $12^{th}$ magnitude will not affect the detector and will not be stored in the catalogue – the last division is included for completeness.

### 2.2.2 Colour Index Code

In this case the colour index is taken to mean B-V in Johnson magnitudes. This is split into uneven bands depending on the UVOT detector response to these wavelengths and each band is assigned a code. The detector is more sensitive to blue than to red. Sources that are too red to affect the detector will not be stored in the catalogue. The bands and their codes are as follows;

If the colour index is less than –0.33 the code is 0.
If the colour index is greater than or equal to –0.33 but less than –0.27 the code is 1.
If the colour index is greater than or equal to –0.27 but less than –0.17 the code is 2.
If the colour index is greater than or equal to –0.17 but less than –0.11 the code is 3.
If the colour index is greater than or equal to –0.11 but less than 0.01 the code is 4.
If the colour index is greater than or equal to 0.01 but less than 0.15 the code is 5.
If the colour index is greater than or equal to 0.15 but less than 0.30 the code is 6.
If the colour index is greater than or equal to 0.30 but less than 0.44 the code is 7.

If the colour index is greater than or equal to 0.44 but less than 0.52 the code is 8.
If the colour index is greater than or equal to 0.52 but less than 0.63 the code is 9.
If the colour index is greater than or equal to 0.63 but less than 0.68 the code is 10.
If the colour index is greater than or equal to 0.68 but less than 0.74 the code is 11.
If the colour index is greater than or equal to 0.74 but less than 0.81 the code is 12.
If the colour index is greater than or equal to 0.81 but less than 1.10 the code is 13.
If the colour index is greater than or equal to 1.10 but less than 1.49 the code is 14.
If the colour index is greater than or equal to 1.49 but less than 1.64 the code is 15.
If the colour index is greater than or equal to 1.64 the code is 16.

## 3. SKY AREA NAMING CONVENTION

The convention used in the star catalogue splits the sky into 2524 areas. Each one is given the name of the hemisphere it is in followed by a four digit coded number.

A given area will be named in the format cccccDDRR, where

- ccccc is ‘north’ or ‘south’
- DD is the declination band
- RR is the right ascension division

The two hemispheres are mirror images of each other in this scheme.

The declination bands are numbered 0 to 21, starting at the equator. Each band is 255 arcminutes wide except for polar bands north21 and south21 which are each 45 arcminutes wide.

Each declination band is divided into a number of equal right ascension divisions. This number depends on the way the width of the field of view changes in right ascension with respect to declination.
Bands 0 to 4 have 90 divisions each.
Bands 5 and 6 have 80 divisions each.
Band 7 has 75 divisions.
Bands 8 to 10 have 72 divisions each.
Band 11 has 60 divisions.
Band 12 has 54 divisions.
Band 13 has 50 divisions.
Band 14 has 45 divisions.
Band 15 has 40 divisions.
Band 16 has 36 divisions.
Band 17 has 27 divisions.
Band 18 has 24 divisions.
Band 19 has 15 divisions.
Band 20 has 8 divisions.
Band 21 has 2 divisions.

The right ascension divisions are numbered from 0 upwards, starting at an RA of 0 degrees, and fit exactly into 360 degrees.

So, for example, sky area south1230 would be in the southern hemisphere at declination band 12 (which extends from –51 degrees to –55 degrees 15 arcmin) and right ascension division 30 (which in this band extends from 200 degrees to 206 degrees 40 arcmin).

This scheme maximizes data compression while retaining positional accuracy to within half an arcminute or better in the field of view.

## 4. PROGRAMS

### 4.1 Converter

The converter program converts a star catalogue file into an intermediate format which is then formatted for uplink using the formatter program. This intermediate stage exists to facilitate the splitting of a huge input file into smaller components, as it would be impossible to uplink an entire catalogue in one pass. It also produces a data file used by the pointers program to produce a pointer table for fast access to the onboard catalogue.

The expected input is a star catalogue file called cat.dat. This should be formatted so that each line contains one entry, in the format RA Dec Vmag Bmag, space separated, with
RA - the right ascension of the star, in the range 0 to 360 degrees, to eight decimal places
Dec - the declination of the star, in the range –90 to +90 degrees, to eight decimal places
Vmag - the Johnson V magnitude of the star to three decimal places
Bmag - the Johnson B magnitude of the star to three decimal places

For example, the star Canopus would be described as;
95.98787763 -52.69571799 -0.608 -0.417

The catalogue file used to produce the initial onboard catalogue will be provided in the archive.

The converter takes each entry and assigns it to an area of sky. It then assigns it an RA offset and a Dec offset relative to the origin of that sky area and also assigns a code relating to the star's magnitude and colour index. All stars that are assigned to the same position have their magnitudes added to create a 'pseudostar' which has equivalent properties to the collection of individual stars. A separate file is output for each sky area, giving the coded positions and magnitudes for each star in that area. The output again has one entry per line, in the format RR DD MMCC, with
RR – the RA offset as a two digit hex number
DD – the declination offset as a two digit hex number
MM – the magnitude code as a two digit hex number
CC – the colour index code as a two digit hex number
The final entry in each file will be an end of list marker (FF FF FFFF) used by the formatter program.

Altogether 2524 sky area files are produced. The scheme used to split the sky into areas is given in section 3, Sky Area Naming Convention. Each file has a filename of the format northxxxx.dat or southxxxx.dat where xxxx is a four digit decimal number.

The sky area files used in the preparation of the current onboard catalogue will be provided in the archive.

The converter also produces a data file called distribution.dat. This contains a list of how many sources are in each sky area, with one line per sky area, in the format

H DD RR NNNN, with
H – the hemisphere of the sky area, given as one character, N or S
DD – the declination code of the sky area as a two digit number from 00 to 21
RR – the right ascension code of the sky area as a two digit number from 00 to 89
NNNN – the number of sources in that sky area.
So, for example, the entry N 09 23 67 will indicate that there are 67 sources in sky area north0923.

Finally, the converter outputs some diagnostics to the screen. If the code or the catalogue is corrupted then it may also output a warning to screen.

## 4.2 Pointers

The pointers program takes the distribution file produced by the converter program and uses it to create a pointer table and a data file. The table is formatted for uplink, and is used onboard to access the star catalogue data quickly. It also includes a CRC value for onboard memory corruption checking purposes. The data file is a useful record of where the components of the catalogue are stored onboard.

The expected input is the distribution.dat file from converter.

The output files are called pointers.img and memorymap.dat. The pointers.img file is formatted for uplink by the ITOS system and contains the pointer table and its CRC value. The memorymap.dat file contains general information on where the catalogue is stored on board, and the locations of the first word of data for each sky area, and for the addendum list.

Program pointers also outputs some diagnostics to the screen.

## 4.3 Formatter

The formatter program takes a single sky area file or any number of concatenated sky area files and produces an uplinkable file readable by the onboard software.

The expected input is a file called format.dat, which consists of one or more concatenated sky area files. The files must be concatenated in a particular order, which is the order in which they are stored in onboard memory. The full sequence is from north0000.dat to north2101.dat followed by south0000.dat to south2101.dat, with the files listed in ascending numerical order. Any section of that sequence may be used.

The program also asks the user for the memory location to which the first word of the data will be loaded. This information is held in the memorymap.dat file produced by the pointers program. The address of the first word of the first sky area in the format.dat file is the required location.

The formatter outputs a file called cat.img, formatted for uplink by the ITOS system. This consists of a five line header, and a body of data in ITOS standard uplink format, which consists of eight hex words per line. The data body is simply a repeat of all the entries in the sky area files, slightly reformatted, but with an extra word inserted at the end of each sky area. This extra word is a CRC value, which the formatter inserts in place of each end of list marker. It is used onboard in checks for memory corruption.

Also the formatter outputs some diagnostics to the screen.

## 4.4 Addendum

The addendum program produces an addendum to the onboard catalogue. Initially the addendum will contain no data, but will zero fill the memory allotted to it. In the unlikely event of an extra source having to be added to the catalogue (for example, if a supernova appears during the mission) it may need to go into the addendum list. If it is coincident with a source already in the catalogue then it can be added to the catalogue, but otherwise it will have to go into the addendum list. The uplinkable file will at minimum consist of an end of list marker that is recognized by the onboard software, followed by zeroes and finished with a crc value.

The expected input is a file called addendum.dat. It consists of a list of sources to be added, in the same format as for cat.dat (see section 4.1 above). If there are no sources to add then an empty file, containing only a single carriage return, can be used as input.

The program also asks the user for the memory location to which the first word of the data will be loaded. This information is held in the second to last line of the memorymap.dat file produced by the pointers program.

The addendum program assigns each source a sky area, and offsets within that area and magnitude and colour index codes in the same way as the formatter does. The format for one source entry will be XXYY RRDD MMCC, with
XX – sky area RA
YY – coded sky area declination (coded to include the hemisphere information)
RR – the RA offset as a two digit hex number
DD – the declination offset as a two digit hex number
MM – the magnitude code as a two digit hex number
CC – the colour index code as a two digit hex number
The end of list marker is ‘FFFF FFFF FFFF’. If there are no sources the end of list marker will still be output.

After source and marker data is output the remaining space allotted to the addendum will be filled with zeroes and finished by a CRC value for the addendum area of memory. The final output file is uplinkable and called addendum.img.

Also program addendum outputs some diagnostics to the screen. If the new addendum area created is larger than the space allotted in memory then a warning will be output to the screen. Only 100 sources can be put into the addendum. If more need to be added then it would be preferable to create and load a new main catalogue.

### 4.5 Locater

The locater program is only for use in the event of adding or deleting a source in the onboard catalogue. It is used to decide where and how the alteration needs to be made.

The expected input is a file called source.dat. This contains one line only. When deleting a source the line is that source's catalogue entry copied exactly from cat.dat. When adding a source it is the RA, Dec, Vmag and Bmag information in the same format as for cat.dat (see section 4.1 above).

The locater program does not produce any files. Its only output is to screen. If it has found that the source in source.dat is coincident with a source in the main catalogue it will indicate this and give the sky area of the source. If the source is not coincident it will indicate that instead.

### 4.6 Disassembler

This program is used to create a reference file from which any star returned by the onboard software can be identified in the initial swift_catalogue.dat and swift_catalogue.extras catalogues. This will not include stars later put into the addendum.

The expected input is the same cat.dat file as used by program converter above.

The output file is called disassembledcat.dat. It runs on a line by line basis and has three types of entry, all space separated.

For a lone source the entry runs;
Line number of source in initial catalogue,
RA from initial catalogue,
Declination from initial catalogue,
V magnitude from initial catalogue,

B magnitude from initial catalogue
Assigned sky area in onboard catalogue,
Assigned RA offset in onboard catalogue,
Assigned declination offset in onboard catalogue,
Assigned V magnitude code in onboard catalogue,
Assigned colour index code in onboard catalogue.

For a source that is a component of a combined source, by reason of coincidence, the entry runs;
Line number of source in initial catalogue,
RA from initial catalogue,
Declination from initial catalogue,
V magnitude from initial catalogue,
B magnitude from initial catalogue,
Assigned component number within the combined source,
Combined source number (which includes sky area).

For each combined source the entry runs;
Combined source number (which includes sky area),
Assigned RA offset in onboard catalogue,
Assigned declination offset in onboard catalogue,
Assigned combined V magnitude code in onboard catalogue,
Assigned combined colour index code in onboard catalogue.

## 5. ARCHIVES

When the software is delivered an archive will also be handed over, containing information on what has been loaded into memory relating to the star catalogue. A file called archive.log will be provided, containing initial information. Any change to the onboard catalogue should be noted in this log.

The archive provided will consist of

- Cat.dat, the catalogue used to produce the first onboard catalogue.
- Swift_catalogue.dat and swift_catalogue.extras, the data sources for cat.dat.
- 2524 files entitled northxxxx.dat or southxxxx.dat (where xxxx is a four digit number) which each represent one sky area of the catalogue. These are not uploadable but uploadable versions can be made with the formatter program.
- Cat.img, the catalogue file actually loaded onto Swift before launch
- Distribution.dat, the file used to create the pointers list
- Memorymap.dat, the file detailing the locations of components of the catalogue in memory
- Pointers.img, the pointer table loaded onto Swift before launch
- Addendum.dat, the file used to generate the addendum area to the catalogue
- Addendum.img, the addendum to the catalogue loaded onto Swift before launch
- Disassembledcat.dat, the catalogue reference file

This archive should be kept in one directory, not the same one that contains the suite of programs (to prevent overwriting), and should never be touched. This is particularly important in the case of distribution.dat and memorymap.dat, which are used when making additions to or deletions from the main catalogue.

If an addition or deletion is made then the relevant files should be stored in a second archive directory and named so that the latest version of any file is clearly marked. Files that may change and should be archived are

- Cat.dat
- Any one of the sky area files, and the .img files produced from them.
- Addendum.dat
- Addendum.img
- Disassembledcat.dat

## 6. PROCEDURES

Throughout these procedures there are diagnostic checks to be made. The converter and addendum programs may also output error messages. If an error is indicated then check the formats of all the input files involved, and also check that no corruption of the code has occurred. Then rerun the procedure.

### 6.1 Adding a Source to the Onboard Catalogue.

There are two ways in which a source can be added to the catalogue. If it is very close to a source already there it will be considered coincident and 'added' to the main catalogue. In this case the magnitudes of the two sources are combined to produce a 'pseudostar' with the same properties as the two sources together. If the source is not close to another it will be put into the addendum instead. This method may seem awkward but is done to maximize data storage with respect to available memory.

First, decide which method of addition to use.

Go to the program suite directory.

Create a file called source.dat. This should contain one line of information on the source to be added, in the format RA Dec Vmag Bmag, space separated, with
RA - the right ascension of the star, in the range 0 to 360 degrees, to eight decimal places
Dec - the declination of the star, in the range –90 to +90 degrees, to eight decimal places
Vmag - the Johnson V magnitude of the star to three decimal places
Bmag - the Johnson B magnitude of the star to three decimal places.

For example, the star Canopus would be described as;
95.98787763 -52.69571799 -0.608 -0.417

Also copy the latest version of the catalogue file into this directory and call it cat.dat.

Run the locater program with the command
./locater

This will output some information to screen. If it says 'No coincident sources found in main catalogue' go to procedure 6.1.1, Adding a Source to the Addendum. Otherwise make a note of the sky area given and go to procedure 6.1.2, Adding a Source to the Main Catalogue.

### 6.1.1 Adding a Source to the Addendum

Make a copy of the latest version of addendum.dat.

Concatenate this and the source.dat file you have just created and call the resulting file addendum.dat.

Read the original archived memorymap.dat file, to find out the address of the first word of the addendum data in memory. Make a note of this address.

Run the addendum program with the command
./addendum

The program will ask you to enter the address of the data. This is the address that you read from the memorymap.dat file. It should be entered as given – the program is expecting a hex number, and there is no need to precede it with 0x, 16# or any other kind of hex identifier.

The program will output to screen the number of records that it has read in. Check that this figure is the same as the number of lines in the new addendum.dat file. If it is not then an error has occurred.

A file called addendum.img will be produced. This file is ready for uplink. Uplink it.

Archive the files addendum.dat and addendum.img and update archive.log.

The source will now be visible to the onboard safing software.

### 6.1.2 Adding a Source to the Main Catalogue

Concatenate the cat.dat file that is now in this directory with the source.dat file you have just created and call the resulting file cat.dat.

Run the converter program with the command
./converter

The program will output to screen the number of records that it has read in. Check that this figure is the same as the number of lines in the new cat.dat file. If it is not then an error has occurred.

2524 sky area files will be produced. Only one should differ from its previous counterpart, and that one should be the area given by the locater program earlier. The difference should only be one word. If this is not the case then an error has occurred.

Archive the one different sky area file and also make a copy of it called format.dat.

Now delete all of the sky area files from this directory, to avoid being swamped.

A distribution.dat file will also have been produced. Make sure that there is no difference between this one and the original archived version. If it has changed then an error has occurred. If it has not changed then delete it.

Read the original archived memorymap.dat file, to find out the address of the first word of the data from this sky area in memory. Make a note of this address.

Run the formatter program with the command
./formatter

The program will ask you to enter the address of the data. This is the address that you read from the memorymap.dat file. It should be entered as given – the program is expecting a hex number, and there is no need to precede it with 0x, 16# or any other kind of hex identifier.

The program will output to screen the number of records that it has read in, and the number of end of area markers. Check that only one end of area marker has been read, and that the number of lines in format.dat equals the number of records read plus the one marker. If not then an error has occurred.

A file called cat.img will be produced. This file is ready for uplink. Uplink it.

Run the disassembler program with the command
./disassembler

This will produce an updated disassembledcat.dat file.

Archive the files cat.dat, disassembledcat.dat and cat.img (which should be renamed to the relevant sky area name) and update archive.log.

The source will now be visible to the onboard safing software.

## 6.2 Deleting a Source from the Onboard Catalogue

If the source to be deleted is in the addendum go to procedure 6.2.1, Deleting a Source from the Addendum. Otherwise go to procedure 6.2.2, Deleting a Source from the Main Catalogue.

### 6.2.1 Deleting a Source from the Addendum

Go to the program suite directory.

Copy into this directory the latest version of addendum.dat and call it addendum.dat.

Delete from the copy you've just made of addendum.dat the line containing the source that is to be deleted.

Read the original archived memorymap.dat file, to find out the address of the first word of the addendum data in memory. Make a note of this address.

Run the addendum program with the command
./addendum

The program will ask you to enter the address of the data. This is the address that you read from the memorymap.dat file. It should be entered as given – the program is expecting a hex number, and there is no need to precede it with 0x, 16# or any other kind of hex identifier.

The program will output to screen the number of records that it has read in. Check that this figure is the same as the number of lines in the new addendum.dat file. If it is not then an error has occurred.

A file called addendum.img will be produced. This file is ready for uplink. Uplink it.

Archive the files addendum.dat and addendum.img and update archive.log.

The source will now have been removed from the onboard catalogue.

### 6.2.2 Deleting a Source from the Main Catalogue.

This is an awkward procedure as the source to be deleted may or may not be 'added' to another coincident source. If it is one of a group of coincident sources added together then all that needs to be changed is the combined magnitude of the group. However, if the deleted source was not coincident with others then after its removal the onboard memory must be padded out, to avoid leaving a gap. Since the source is 'hidden' rather than totally erased, the disassembledcat.dat file should be left as it is – the source will still be present in the initial catalogue.

In either case, begin by going to the suite directory.

Copy the latest version of the catalogue file into this directory and call it cat.dat.

Create a file called source.dat. Copy into it the one line entry from cat.dat of the source you wish to delete.

Run the locater program with the command
./locater

This will output to screen the sky area in which the source is located. Make a note of this area.

Now delete the one line entry for the source from cat.dat.

Run the converter program with the command
./converter

The program will output to screen the number of records that it has read in. Check that this figure is the same as the number of lines in the new cat.dat file, and one less than the number of lines in the previous cat.dat file. If it is not then an error has occurred.

2524 sky area files will be produced, and so will a new distribution.dat file. Delete all the sky area files except for the one identified by the locater program above.

Diff the new distribution.dat file against the original archived version. If there is no difference then a coincident source has been deleted. Go to procedure 6.2.2.1, Deleting a Coincident Source. Otherwise the only difference should be in the line relating to the source's sky area, which you noted above. The number of sources in that area should have decreased by one. In this case go to procedure 6.2.2.2, Deleting a Non-Coincident Source. If anything else has occurred there has been an error.

#### 6.2.2.1 Deleting a Coincident Source

As a final check, diff the new sky area file against the previous version. The difference should be one word only. If not then an error has occurred.

Archive the new sky area file and make a copy of it called format.dat.

Read the original archived memorymap.dat file, to find out the address of the first word of the data from this sky area in memory. Make a note of this address.

Run the formatter program with the command
./formatter

The program will ask you to enter the address of the data. This is the address that you read from the memorymap.dat file. It should be entered as given – the program is expecting a hex number, and there is no need to precede it with 0x, 16# or any other kind of hex identifier.

The program will output to screen the number of records that it has read in, and the number of end of area markers. Check that only one end of area marker has been read, and that the number of lines in format.dat equals the number of records read plus the one marker. If not then an error has occurred.

A file called cat.img will be produced. This file is ready for uplink. Uplink it.

Archive the files cat.dat and cat.img (which should be renamed to the relevant sky area name) and update archive.log.

The source will now have been removed from the onboard catalogue.

#### 6.2.2.2 Deleting a Non-Coincident Source

As a final check, diff the new sky area file against the previous version. The difference should be that the new version has one line less than the previous version. If not then an error has occurred.

To keep the onboard memory in order the deleted source must be replaced with a placeholder. This is added as follows.

Edit the new sky area file, to add at the beginning the line
FF FF 0000
in that format exactly ie, two capital Fs, then another two, then four zeroes. This is the placeholder, which will be ignored by onboard safing software.

Archive the new, edited sky area file and make a copy of it called format.dat.

Read the original archived memorymap.dat file, to find out the address of the first word of the data from this sky area in memory. Make a note of this address.

Run the formatter program with the command
./formatter

The program will ask you to enter the address of the data. This is the address that you read from the memorymap.dat file. It should be entered as given – the program is expecting a hex number, and there is no need to precede it with 0x, 16# or any other kind of hex identifier.

The program will output to screen the number of records that it has read in, and the number of end of area markers. Check that only one end of area marker has been read, and that the number of lines in format.dat equals the number of records read plus the one marker. If not then an error has occurred.

A file called cat.img will be produced. This file is ready for uplink. Uplink it.

Archive the files cat.dat and cat.img (which should be renamed to the relevant sky area name) and update archive.log.

The source will now have been effectively removed from the onboard catalogue.

## 6.3 Replacing Corrupted Areas of the Onboard Catalogue

If the onboard catalogue becomes corrupted then an NHK packet will be issued. This packet will notify that error 146, corrupted EEPROM data, has occurred. The first parameter it gives will have value 4, the code for the star catalogue, and the second parameter will indicate in which area the corruption has occurred.

The second number will be a four digit hex number.
If the number is 0xEEEE then the corruption is in the pointer table. Go to procedure 6.3.1, Reloading the Pointer Table.
If the number is anything below 0x4E20 then the corruption is in one of the sky areas. Make a note of the hex number and go to procedure 6.3.2, Reloading a Sky Area.
If the number is 0xAAAA then the corruption is in the addendum. Go to procedure 6.3.3, Reloading the Addendum.

### 6.3.1 Reloading the Pointer Table

Uplink the latest version of pointers.img.

Make a note of this in archive.log.

The corruption should now be corrected.

### 6.3.2 Reloading a Sky Area

Take the second parameter in the NHK message and convert it from hex to decimal. Left pad it with zeroes to make a five digit number, eg 101 becomes 00101. Now it can be used to determine which sky area is corrupted.

If the first digit is 1 then the area is in the southern hemisphere. If it is 0 then it is in the northern hemisphere. The second and third digits give the declination band of the sky area, and the fourth and fifth give the right ascension division.
For example, 10000 gives sky area south0000, and 01216 gives sky area north1216.

Determine the sky area to reload.

If this area has previously had a source added to or deleted from it then a .img file should be archived. Uplink the latest version of this .img file and note this in archive.log. The corruption should now be corrected and this procedure finishes here.

If the area has been untouched since launch then a .img file must be regenerated for it. Copy the sky area file from the original archive into the suite directory, and call it format.dat.

Read the original archived memorymap.dat file, to find out the address of the first word of the data from this sky area in memory. Make a note of this address.

Run the formatter program with the command
./formatter

The program will ask you to enter the address of the data. This is the address that you read from the memorymap.dat file. It should be entered as given – the program is expecting a hex number, and there is no need to precede it with 0x, 16# or any other kind of hex identifier.

The program will output to screen the number of records that it has read in, and the number of end of area markers. Check that only one end of area marker has been read, and that the number of lines in format.dat equals the number of records read plus the one marker. If not then an error has occurred.

A file called cat.img will be produced. This file is ready for uplink. Uplink it.

Archive the file cat.img (which should be renamed to the relevant sky area name) and update archive.log.

The corruption should now be corrected.

### 6.3.3 Reloading the Addendum

Uplink the latest version of addendum.img.

Make a note of this in archive.log.

The corruption should now be corrected.

## 6.4 Reloading the Entire Onboard Catalogue

This should be a procedure of last resort, as it will take many passes to uplink the entire catalogue. Note that the catalogue consists of approximately 1Mb of data.

First the main catalogue must be split into manageable chunks for uplink.

Decide on an amount of data that can be uplinked during one pass.

Read the original archived file memorymap.dat. This gives the starting address (in words, not bytes) in memory for each sky area – ignore the pointer table and addendum information. Since the main catalogue is continuous in memoryy ou can work out that the length of any sky area in words is its starting address subtracted from the starting address of the following sky area (or for the last area, south2101, its starting address subtracted from the starting address of the addendum). From this, the uplink data rate and the amount of time available per pass, you should be able to determine how many and which sky areas can be uplinked in one pass. It is important that they are formatted for uplink in the same sequence as given in the memorymap.dat file.

For each pass;

- Make copies of the latest versions of the selected sky area files and concatenate them in the order given by memorymap.dat, ie north files in ascending numerical order followed by south files in ascending numerical order. Call this concatenated file format.dat.

- Read the original archived memorymap.dat file, to find out the address of the first word of the data from the first sky area in format.dat. Make a note of this address.

- Run the formatter program with the command ./formatter

- The program will ask you to enter the address of the data. This is the address that you read from the memorymap.dat file. It should be entered as given – the program is expecting a hex number, and there is no need to precede it with 0x, 16# or any other kind of hex identifier.

- The program will output to screen the number of records that it has read in, and the number of end of area markers. Check that the number of end of area markers read equals the number of files concatenated to make format.dat, and that the number of lines in format.dat equals the number of records read plus the number of markers. If not then an error has occurred.

- A file called cat.img will be produced. This file is ready for uplink. Uplink it and archive it.

Uplink the latest version of pointers.img.

Uplink the latest version of addendum.img.

The entire catalogue should now be correctly stored on board.

## 6.5 Regenerating the Archive

If the archive is lost it may be recreated using only the software suite provided and the original cat.dat file.

Make a copy of the cat.dat file in the suite directory.

Run the converter program with the command
./converter

This will output to screen the number of records that it has read in, and the number of coincident sources. Check that the number of records is the same as the number of lines in the file cat.dat. If not then an error has occurred.
The program will produce all the sky area files and distribution.dat. Archive them all.

Run the pointers program with the command
./pointers

This will output to screen the number of entries that it has read in from all the sky areas combined. Check that this is the same as the number of lines in the file cat.dat minus the number of coincident sources.
The program will produce pointers.img and memorymap.dat. Archive them both.

Create an empty file (ie a file containing only one carriage return) called addendum.dat.

Read the memorymap.dat file, to find out the address of the first word of the addendum data in memory. Make a note of this address.

Run the addendum program with the command
./addendum

The program will ask you to enter the address of the data. This is the address that you read from the memorymap.dat file. It should be entered as given – the program is expecting a hex number, and there is no need to precede it with 0x, 16# or any other kind of hex identifier.

This will output to screen the number of records that it has read in. Check that this is zero. If not then an error has occurred.
The program will produce addendum.img. Archive it and addendum.dat.

Concatenate all of the sky area files, in order (north files in ascending numerical order followed by south files in ascending numerical order), into a file called format.dat. For example, on Linux you would do this with the command
cat north*.dat south*.dat > format.dat
As a check, the number of lines in the concatenated file should be equal to the number of entries as given by the pointers program, plus 2524 end of file markers.

Read the memorymap.dat file, to find out the address of the first word of the data from the north0000 sky area in memory. Make a note of this address.

Run the formatter program with the command
./formatter

The program will ask you to enter the address of the data. This is the address that you read from the memorymap.dat file. It should be entered as given – the program is expecting a hex number, and there is no need to precede it with 0x, 16# or any other kind of hex identifier.

The program will output to screen the number of records and end of area markers that it has read in. Check that the number of records is the same as the number of lines in the file cat.dat and that the number of markers is 2524. If not then an error has occurred.
The program will produce cat.img. Archive it.

Run the disassembler program with the command
./disassembler

This will produce a disassembledcat.dat file. Archive it.

The original archive has now been recreated.

## 6.6 Creating a New Catalogue

If a totally new catalogue is required then create a new archive for it. Make sure that the new initial catalogue file is formatted in the same way as the original cat.dat (see section 4.1 above), and call it cat.dat.

Check the number of entries in the new cat.dat – this should be the same as the number of lines in the file.
If it is greater than 259806 then **proceed cautiously**. This will only fit into available memory if some sources are coincident. If not it will overwrite part of the deferred command store. The converter program will issue a warning to screen if the catalogue is too large.

Make a copy of the cat.dat file in the suite directory.

Run the converter program with the command
./converter

This will output to screen the number of records that it has read in. Check that this is the same as the number of lines in the file cat.dat. If not then an error has occurred.
The program will produce all the sky area files and distibution.dat. Archive them all.

Run the pointers program with the command
./pointers

This will output to screen the number of entries that it has read in from all the sky areas combined. Check that this is the same as the number of lines in the file cat.dat. If not then an error has occurred.
The program will produce pointers.img and memorymap.dat. Archive them both.

Create an empty file (ie a file with no content) called addendum.dat.

Read the memorymap.dat file, to find out the address of the first word of the addendum data in memory. Make a note of this address.

Run the addendum program with the command
./addendum

The program will ask you to enter the address of the data. This is the address that you read from the memorymap.dat file. It should be entered as given – the program is expecting a hex number, and there is no need to precede it with 0x, 16# or any other kind of hex identifier.

This will output to screen the number of records that it has read in. Check that this is zero. If not then an error has occurred.

The program will produce addendum.img. Archive it and addendum.dat.

Run the disassembler program with the command
./disassembler

This will produce a disassembledcat.dat file. Archive it.

The main catalogue must be split into manageable chunks for uplink.

Decide on an amount of data that can be uplinked during one pass.

Read the file memorymap.dat. This gives the starting address (in words, not bytes) in memory for each sky area – ignore the pointer table and addendum information. Since the main catalogue is continuous in memoryy ou can work out that the length of any sky area in words is its starting address subtracted from the starting address of the following sky area (or for the last area, south2101, its starting address subtracted from the starting address of the addendum). From this, the uplink data rate and the amount of time available per pass, you should be able to determine how many and which sky areas can be uplinked in one pass. It is important that they are formatted for uplink in the same sequence as given in the memorymap.dat file.

For each pass;

- Make copies of the latest versions of the selected sky area files and concatenate them in the order given by memorymap.dat, ie north files in ascending numerical order followed by south files in ascending numerical order. Call this concatenated file format.dat.

- Read the memorymap.dat file, to find out the address of the first word of the data from the first sky area in format.dat. Make a note of this address.

- Run the formatter program with the command ./formatter

- The program will ask you to enter the address of the data. This is the address that you read from the memorymap.dat file. It should be entered as given – the program is expecting a hex number, and there is no need to precede it with 0x, 16# or any other kind of hex identifier.

- The program will output to screen the number of records that it has read in, and the number of end of area markers. Check that the number of end of area markers read equals the number of files concatenated to make format.dat, and that the number of lines in format.dat equals the number of records read plus the number of markers. If not then an error has occurred.

- A file called cat.img will be produced. This file is ready for uplink. Uplink it and archive it.

Uplink pointers.img.

Uplink addendum.img.

The new catalogue is now onboard and archived.

### 6.7 Identifying a given source

When the onboard software finds a star in the requested field of view of the UVOT it sends an NHK message to ground. This message includes a coded version of the sky area and offsets of the source.

If the sky area is 0xAAAA then the star was found in the addendum and the number contained in ‘offsets’ is actually the line number of that star in the file addendum.dat. This identifies the star.

Otherwise, first decode the sky area, following these steps;

a) convert it to a decimal value
b) *left* pad it with zeroes to make a 5 digit number
c) if the first digit is 1, it is a southern hemisphere area, otherwise it is northern
d) the second and third digits are the declination band
e) the fourth and fifth digits are the right ascension division

so for example, 0x04C0 becomes decimal 01216, which indicates sky area north1216.

Decode the offsets by splitting the number into two 2 digit hex numbers and converting them to decimals, eg 6A07 becomes 106, 7. These are the RA and declination offsets respectively within the sky area.

For most sources this combination of sky area and offsets will be unique, and a simple grep of the disassembledcat.dat file will locate them. The first number on the line entry for the source in disassembledcat.dat will be its line number in the initial swift_catalogue.dat file (see note below). This identifies the star.

If the combination of sky area and offsets is identified as belonging to a combined source, then the components of that source can be located with another grep of the disassembledcat.dat file, this time searching on the combined source name, which is unique. The first number on the line entry for each component of the combined source in disassembledcat.dat will be its line number in the initial swift_catalogue.dat file (see note below). This identifies all the stars in the combined source.

NOTE: The initial catalogue, swift_catalogue.dat, has had two extra sources added to it to allow for the effect of extended sources with extended bright nuclei. These two extra sources are stored in swift_catalogue.extras and form the last two lines of cat.dat. This means that line numbers 239853 and 239854 actually refer to lines 1 and 2 respectively in swift_catalogue.extras. All other line numbers are for swift_catalogue.dat.