EDOC061

EUROGAM PROJECT

NSF DATA ACQUISITION SYSTEM

Unified Spectrum Format

Edition 2.2
Jan 1992

NSF Software Systems Group
UK Science and Engineering Research Council
Daresbury Laboratory

Introduction

Previous discussion documents have proposed structures for a spectrum database and for data storage. It has also previously been suggested that the same format be used for on-line spectra, disc based spectra and for spectra archived on tape. The requirements for a spectrum format for both on-line and off-line use have been specified in the document EDOC060. This document describes a format which meets the requirements of EDOC060 and can be used for on-line spectra in the histogrammer or sorter and for off-line spectra held on disc or tape. Requirements for implementation are defined.

The document EDOC060 should be read for formal specification of the requirements. The document EDOC062 should be read for details of program access to the spectrum data.

Document History

Version 1.0 Dec 1990 - Initial version

Version 2.0 May 1991 - re-order fields in the header and reduce data arrays to just spectrum and an optional error spectrum. Unused pointers now contain -1 so that pointers may have the value 0.

Version 2.1 Sept 1991 - Removal of graph type arrays, change to data array descriptor, n dimensional arrays stored following C rather than Fortran ordering convention.

Version 2.2 Jan 1992 - Number of Information strings increased to 32.

Terminology

Throughout this document several terms in common usage are used which are defined here to avoid misinterpretation.
Spectrum - any collection of data items which includes the conventional 1D histogram and 2D matrices in common use.
Channel - an individual data item in a spectrum.
On-line - spectra held in memory (normally VME global memory or a processor memory) which is potentially being incremented by either hardware or software. This includes spectra being generated by a post experiment data replay.
Off-line - spectra held on secondary storage (normally disc) which may be read or mapped into processor memory. The data contents are not normally being changed.

General Format

Each spectrum consists of three parts; a header, an string space and a counts space. For on-line spectra the components may not be contiguous but would be allocated within a large memory space. For off-line spectra the components would normally be contiguous within a space defined by the disc file holding the spectrum with the header as the first component in the file.

The spectrum header is a fixed length 512 byte data structure which is used to access all items in the string and counts spaces. The associated string space and counts space are allocated in units of 256 bytes.

The string space is used to hold character strings which contain name, title, annotation and calibration information etc. Each character string is allocated one (or more) units of 256 bytes from the string space. The character string is represented using the standard XDR format which is a 32 bit integer field containing the number of significant characters which follow. The characters are padded to fill the allocated space with trailing null characters.

Note: The minimum space allocation allows for strings of upto 252 characters in length but any required length is possible by allocating the necessary number of contiguous 256 byte fragments.

The address of each character string expressed as an offset into the string space is held as a pointer in the spectrum header. Unused pointers which do not have string space allocated are set to -1.

The counts space is used to hold data arrays which contain the spectrum counts and is used and referenced in the same way as the string space.

Header Format

Offset 0 - *magic number*. A 32 bit integer field set to the value 412900921 (decimal). The *magic number* can be used as a test that a disc file probably does contain a spectrum.

Offset 4 - *header version number*. A 32 bit integer field set to the value 1. Available to allow other header formats in the future. This document describes version 1 of the header.

Offset 8 - *spectrum name*. A 32 byte field containing the name of the spectrum padded where necessary with trailing null characters. In the on-line mode this field is used to access the spectrum. In the off-line mode spectra are accessed only by the disc file name and this field may be ignored (set to all null characters) unless the spectrum was previously an on-line spectrum in which case the name should be preserved but it will not be used for spectrum access.

Offset 40 - *dimension*. A 32 bit integer field which contains the number of dimensions used to describe this spectrum. This version of the header allows for a maximum of 8 dimensions.

Offset 44 - *creation time and date*. A 20 byte field containing the time and date at which the spectrum was first created as ASCII text in the form "nn-mmm-yyyy hh:mm:ss" (e.g. 06-Dec-1990 12:07:00). When an on-line spectrum is written to disc this field is preserved and the disc file creation time/date used to record when the spectrum was written to disc.

Offset 64 - *modification time and date*. A 20 byte field containing the time and date at which the spectrum was last modified as ASCII text in the form "nn-mmm-yyyy hh:mm:ss". When an on-line spectrum is written to disc this field is preserved.

Offset 84 - *dimension 1 base*. A 32 bit integer field containing the coordinate of the first channel for dimension 1 of the spectrum.

Offset 88 to 112. Base information for dimensions 2 to 8. Unused fields should be set to -1.

magic number
header version number
spectrum name
dimension
creation time and date
modification time and date
dimension 1 base
:
dimension 8 base
dimension 1 range
:
dimension 8 range
information 1 pointer
:
information 32 pointer
annotation 1 pointer
:
annotation 8 pointer
calibration 1 pointer
:
calibration 8 pointer
efficiency 1 pointer
:
efficiency 8 pointer
data array 1 descriptor
data array 2 descriptor
base address of string space
string free space
top of string space
base address of counts space
counts free space
top of counts space
unused

Offset 116 - *dimension 1 range.* A 32 bit integer field containing the number of channels for dimension 1 of the spectrum.

Offset 120 to 144. Range information for dimensions 2 to 8. Unused fields should be set to -1.

Offset 148 - *information 1 pointer.* A 32 bit integer field containing the offset from the base of the string space of a character string holding information about this spectrum. If unused should be set to -1.

Offset 152 to 272. Further information pointers. See later for the use of the information fields. Unused pointers should be set to -1.

Offset 276 - *annotation 1 pointer.* A 32 bit integer field containing the offset from the base of the string space of a character string holding annotation information for dimension 1. If unused should be set to -1.

Offset 280 to 304. Annotation pointers for dimensions 2 to 8. Unused pointers should be set to -1.

Offset 308 - *calibration 1 pointer.* A 32 bit integer field containing the offset from the base of the string space of a character string holding calibration information for dimension 1. If unused should be set to -1.

Offset 312 to 336. Calibration pointers for dimensions 2 to 8. Unused fields should be set to -1.

Offset 340 - *efficiency 1 pointer.* A 32 bit integer field containing the offset from the base of the string space of a character string holding efficiency information for dimension 1. If unused should be set to -1.

Offset 344 to 368. Efficiency pointers for dimensions 2 to 8. Unused fields should be set to -1.

Offset 372 - *data array 1 descriptor.* A 20 byte field containing a description of the format of the data array for this spectrum.

Offset 392 - *data array 2 descriptor.* A 20 byte field containing a description of the format of an error spectrum associated with this spectrum. If unused this field should be set to all one bits.

Offset 412 - *base address of string space.* A 32 bit integer field which contains the address of the base of the string space. For on-line spectra it will be an absolute VME address. For off-line spectra it should be set to the offset of the string space from the base of the header.

Offset 416 - *string free space.* A 32 bit integer field containing the offset from the base of the string space of the first unused location.

Offset 420 - *top of string space.* A 32 bit integer field containing the offset from the base of the string space of the last available location.

Offset 424 - *base address of counts space.* A 32 bit integer field which contains the address of the base of the counts space. For on-line spectra it will be an absolute VME address. For off-line spectra it should be set to the offset of the counts space from the base of the header.

Offset 428 - *counts free space.* A 32 bit integer field containing the offset from the base of the counts space of the first unused location.

Offset 432 - *top of counts space.* A 32 bit integer field containing the offset from the base of the counts space of the last available location.

For further description of the uses of the space address and offset fields see the section on Implementation.

Offset 436 to 512. Unused

Data Array Descriptor

array layout
array type
reserved for future use
reserved for future use
pointer

Offset 0 - *array layout.* A 32 bit integer field which is used to define the format of channel items within a data array (see below). This field will be -1 for an unused array descriptor.

Offset 4 - *array type.* A 32 bit integer field which is used to define the format of the data item within a channel item (see below).

Offset 16 - *pointer.* A 32 bit integer field containing the offset from the base of the counts space of the data array. This field will be -1 for an unused array descriptor.

The *array layout* defines the format of each channel item and the ordering of channel items within the data array.

array layout =	0	histogram/matrix
	1	histogram/half matrix

For a spectrum having dimension n
histogram/matrix - each channel item consists only of a data item ordered following the C language convention for a n dimensional array.
histogram/half matrix only stores the upper diagonal of the array.

The *array type* defines the format of the data item component of each channel item within the data array.

array type =	0	8 bit unsigned integer
	1	8 bit signed integer
	2	16 bit unsigned integer
	3	16 bit signed integer
	4	32 bit unsigned integer
	5	32 bit signed integer
	6	32 bit floating point

Signed integers are held in 2s complement format and floating point numbers in IEEE format.

Information strings

Up to 32 information strings are available per spectrum.
String 1 is allocated for spectrum title information.
String 2 is allocated for information about the experiment which created the spectrum - for example experiment name, beam energy, beam ion species and target species etc.
String 3 is allocated for information about the run within the experiment - for example

run name, run number etc.
Strings 4 and 5 are available for a general comment indicating the content of data arrays 1 and 2.
The remaining strings are available for general user comments or additional information.
All information strings are free format text and may contain any printable characters including LF (UNIX end of line).

Annotation strings

One string per dimension is available to allow for a description of the "significance of the dimension". For example it could contain the "units" (MeV , cm etc) of the calibration.

Calibration strings

One string per dimension is available to hold calibration information for that dimension. The information is held free format in character representation. The first field in the string is the name of the calibration method and the remaining fields are parameters to that calibration routine.

Efficiency strings

One string per dimension is available to hold efficiency information for that dimension. The information is held free format in character representation. The first field in the string is the name of the efficiency calculation algorithm and the remaining fields are parameters to that algorithm.

Data Interchange

This document describes a common format for spectra held in memory or on disc. On-line spectra will at the end of a run be 'saved' and become off-line spectra held on disc. The disc files can be copied to any available magnetic tape medium in the same format for backup and archive purposes using standard system utilities (for UNIX one would use *tar* or *dump*). Spectra can be moved in this format between the UNIX workstations in the Eurogam colaboration using the utility *tar* or by a network connection.

For interchange of data with laboratories outside the Eurogam colaboration and for those running other data analysis systems which may wish to import the Eurogam data a more system independant format which can be used for writing spectra onto tape is desired. This format can trade efficiency of access and compactness of the data for maximum easy of interchange. A suitable format for this purpose is described in the document *EDOC080*.

Implementation

Suggestions as to methods of implementation to be provided.