Simple Binary Formats

Saving Data to Simple Binary Data Files

EEG data and events can be stored in a Net Station simple binary data file. Unsegmented source files and Viewer window selections that are unsegmented are saved to continuous simple binary format files. Segmented source files are automatically saved as a segmented simple binary file. The structure of each segment includes a mini-header that identifies the category with which the segment is associated, and a time stamp, as explained below.

Users choose the type of simple binary file to save via the **Save Selection...** and **Save A Copy As...** commands on the **File** menu in review mode (see the "Files" chapter of the EGI Net Station 2.0 Technical Manual). In the dialog that results from choosing either of these menu commands, the user is given the choice of both units and precision:

- A/D unit EEG data as integer;
- A/D unit EEG data as single-precision;
- A/D unit EEG data as double-precision;
- Microvolt EEG data as integer;
- Microvolt EEG data as single-precision; and
- Microvolt EEG data as double-precision.

Note: Note: the "Convert to microvolts" checkbox option will be grayed out and unavailable if the source file's EEG data is already in microvolt units. In such an instance, there will be no choice but for the output file's units to be in microvolts.

Byte Order

All simple binary files use big-Endian byte order for the representation of numeric data. This is the native numerics byte order for Apple and UNIX computers. Applications that run on Intel hardware or other little-Endian machines will have to do byte-swapping in order to read the numeric data of a simple binary format correctly. Users are advised that some header fields in simple binary format files contain *character* data, notably the sorted list of event codes and category name array. This data type is "platform-neutral" and byte-swapping should not be performed on these fields.

Headers of the Binary Data Formats

Each simple binary file has a header, a series of bytes having its starting point at the beginning of the file. The headers contain metadata, some values of which are essential to calculating or otherwise determining the file's relevant offsets. In the sections that follow, we refer to values that can be found in these headers by giving their offsets. Consult the tables for each format for a complete explication of the simple binary files' structures:

Table B-7: Net Station Simple Binary Format Without Events

Table B-8: Net Station Simple Binary Format With Events

Table B-9: Net Station Segmented Simple Binary Format Without Events

Table B-10: Net Station Segmented Simple Binary Format With Events

Versions and Types of Simple Binary Files

Net Station Simple Binary Format Version Codes

The term 'file type' is part of Mac OS where it refers to the case-sensitive, four-character code that the Finder uses to identify a file's type. The four-character code for an unsegmented simple binary file is 'UGLY' and for a segmented simple binary file it is 'eGLY' (see Table 6 in Files in the EGI Net Station 2.0 Technical Manual). This method for typing files is only available on Mac OS, since the file type code is not a component of the file's 'data' fork.

For portablility, and to identify the precision of the EEG data contained in a simple binary file, Net Station writes into each file a single digit version code that can be *read as a long integer from the first 4 bytes of the file*. This version code is interpreted in Table B-2. It identifies how in the file the EEG values are stored (integers, single-precision, or double-precision) and also whether the file is structured as continuous data or segments. The header structure for continuous files is mostly similar to that of segmented files, but there are important differences as seen in Table B-7 through Table B-10.

Table B-2: Simple Binary Format File Type Codes

	Type of File	
Data Type	**Continuous (Unsegmented)**	**Segmented**
integer	2	3
single precision	4	5
double precision	6	7

Determining Units

As explained earlier (see page B-5), users have the option of saving their data to simple binary format as either A/D units or microvolts, assuming that the source file data are not already in the form of microvolts. A/D units are raw amplifier values which can be internally converted to microvolt values by Net Station. Users can perform a similar conversion by calculating a conversion factor using values that will be found in the header of the file.

The header of a simple binary file has two fields that need to be consulted to determine whether the EEG data stored in the file are in microvolts or A/D units. *These fields are the '**bits**' field at offset 26 and the '**range**' field at offset 28*. If the values in the '**bits**' and '**range**' fields are both 0, then the file's data is already in the form of microvolts. Users who save to simple binary format should note that this is the only method to determine if a file's data are in A/D units or microvolts.

Converting A/D Units to Microvolts

If the values read from the 'bits' and 'range' fields of the header are not equal to zero, this signifies that the EEG data stored in the file have not been converted to microvolts, but are stored as A/D units.

Converting the A/D units to microvolts can be done programmatically by applying the following formula:

microvolt value = $(\text{range} \,/\, 2^{\text{bits}}) \times$ A/D value,
where 'bits' and 'range' are read from the header of the file.

This formula will not produce as accurate a microvolt value as using gains and zeros (see the "NetAmps USB Device" chapter in the EGI Net Station 2.0 Technical Manual and "Calibration Signal" in CHAPTER 3 of the Net Amps 200 Technical Manual. So for the greatest microvolt conversion accuracy, users who wish a portable format file to have its data in microvolts are advised to check the "Convert to Microvolts" checkbox when generating the portable format file using Net Station. Alternatively, the gains and zeros of the Net Station Recording or Session source file can be obtained in the form of text files using the "Net Station File Exporter" application (see the "Distribution" chapter of the EGI Net Station 2.0 Technical Manual. Once gains and zeros have been obtained, they can be read in to a program and used to convert A/D unit data to microvolt data. See also the entry for "Scaling Factor" in the Glossary of the EGI Net Station 2.0 Technical Manual.

With Events and Without

When the"Net Station Raw EEG File (Without Events)"output option is chosen before generating a portable format file, the length of the header will consist of a fixed number of bytes. If the "With Events"option was chosen, the length of the header will vary, but can be calculated based on certain values that will always be found at fixed offsets in the file. If the file was generated "With Events", event and EEG data will alternate according to a pattern which can be determined using header information.

Single Sample Record (SSR)

We define a Single Sample Record (SSR) as a container for the data of a single sample. An SSR contains EEG data, and may also contain events if events were saved into the file. If there are events, the event states for a given sample occur after the EEG data for that sample, as shown in the tables below, with the precision of each event state being the same as the precision of the EEG data with which it is associated. Hence, event states associated with integer data will be valued at 0 or 1, event states associated with floating point data will be valued at 0.0 or 1.0 at single or double precision.

The EEG and events of a simple data file occur in the file as consecutive SSRs, one for each sample. If the file does not contain events (i.e. the 'Number of Unique Event Codes' field in its header reads 0) then the SSRs in the file will have the structure shown in Table B-3. If the file does contain events (i.e. the 'Number of Unique Event Codes' field in its header is greater than 0), then the SSRs in the file will have the structure shown in Table B-4.

Table B-3: Structure of SSR (Number of Events = 0; Number of Channels = Nc)

Data Type	Structure of Record	Number of Bytes per Record
integer	Consecutive values for channel 1 thru channel Nc	2 x Nc
single precision	"	4 x Nc
double precision	"	8 x Nc

Table B-4: Structure of SSR (Number of Events > 0; Number of Channels = Nc; and Number of Events = Ne)

Data Type	Structure of Record	Number of Bytes per Record
integer	Consecutive values for channel 1 thru channel Nc, followed by event states from state of event 1 thru state of event Ne	(2 x Nc) + (2 x Ne)
single precision	"	(4 x Nc) + (4 x Ne)
double precision	"	(8 x Nc) + (8 x Ne)

Tables Showing Structures of Simple Binary Format Files

The last section of this appendix contains tables that communicate the structures of the simple binary formats. Each table displays the structure of a file's header, followed by the structure of its data:

EEG Data;
EEG Data & Events; or
EEG Segments.

The tables indicate file offsets and how to calculate them when necessary.

Unsegmented Files

For unsegmented files (versions 2, 4, and 6) the EEG Data or EEG Data and Events part of a file consists of consecutive SSRs (defined above). If events have been included in such files, as shown in Table B-8, the end of the header is the location where the identities of the events are stored. These identities are in the form of 4-character codes. The event states in each SSR are in the same order as the 4-character codes.

Segmented Files

For segmented files (versions 3, 5, and 7) the EEG Segments part of a file consists of consecutive ***Segments***. Each segment begins with a category index (short integer) and time stamp (long) which is directly followed by a block of consecutive SSRs (see Table B-5 & Table B-6).

The 1-based category index identifies the category with which the segment is associated. If the value in the index field is 1, then the segment is associated with the first category of the categories array. If it is 2, then with the second category, and so forth. The categories array, which starts at offset 32 in the header of these segmented files, holds the names of the categories to which the segments can belong as a series of pascal strings (pstrings). A pstring is a one dimensional character array with an added byte at the beginning that holds the length of the array. The 4-byte timestamp holds the start time of the segment in milliseconds.

So, in effect, each segment has a mini-header of its own that takes up its first 6 bytes. Following this mini-header are the SSRs, as many of them as there are samples (which can be read from the Number of Samples field).

The segmented file format does not allow variable-length segments, so once the segment size is determined (use Table B-5 or Table B-6 depending on whether the Number of Events field is 0 or not), the offset to the beginning of each segment can be determined after consulting the Number of Segments field.

Table B-5: Structure of a Segment (Number of Events = 0; Number of Channels = Nc; Number of Samples = Ns)

Data Type	Structure of Segment	Number of Bytes per Segment
integer	Index (short), followed by TimeStamp (long), followed by consecutive SSRs	2 + 4 + [(2 x Nc) x Ns]
single precision	"	2 + 4 + [(4 x Nc) x Ns]
double precision	"	2 + 4 + [(8 x Nc) x Ns]

Table B-6: Structure of Segment (Number of Events > 0; Number of Channels = Nc; Number of Samples = Ns Number of Events = Ne)

Data Type	Structure of Segment	Number of Bytes per Segment
integer	Index (short), followed by TimeStamp (long), followed by consecutive SSRs	2 + 4 + [(2 x Nc) + (2 x Ne)] x Ns
single precision	"	2 + 4 + [(4 x Nc) + (4 x Ne)] x Ns
double precision	"	2 + 4 + [(8 x Nc) + (8 x Ne)] x Ns

Table B-7: Net Station Simple Binary Format Without Events			
HEADER			
Size (bytes)	**File Offset**	**Data Type**	**Description of Data**
4	0	long	Version number: if 2, EEG data are signed short integer; if 4, EEG data are single-precision float; if 6, EEG data are double-precision float.
2	4	short	Recording time: year (4-digit value)
2	6	short	Recording time: month
2	8	short	Recording time: day
2	10	short	Recording time: hour
2	12	short	Recording time: minute
2	14	short	Recording time: second
4	16	long	Recording time: millisecond
2	20	short	Sampling rate (samples per second)
2	22	short	Number of channels
2	24	short	Board gain (1, 2, 4, or 8)
2	26	short	Number of conversion **bits**
2	28	short	Full-scale **range** of amplifier in μV
4	30	long	Number of samples
2	34	short	Number of unique event codes: equals 0 for files without events;
EEG DATA			
size of SSR in bytes x Number of samples	36	varies	Consecutive SSRs (See Table B-3)

Table B-8: Net Station Simple Binary Format With Events			
HEADER			
Size (bytes)	**File Offset**	**Data Type**	**Description of Data**
4	0	long	Version number: if 2, EEG data are signed short integer; if 4, EEG data are single-precision float; if 6, EEG data are double-precision float.
2	4	short	Recording time: year (4-digit value)
2	6	short	Recording time: month
2	8	short	Recording time: day
2	10	short	Recording time: hour
2	12	short	Recording time: minute
2	14	short	Recording time: second
4	16	long	Recording time: millisecond
2	20	short	Sampling rate (samples per second)
2	22	short	Number of channels
2	24	short	Board gain (1, 2, 4, or 8)
2	26	short	Number of conversion **bits**
2	28	short	Full-scale **range** of amplifier in μV
4	30	long	Number of samples
2	34	short	Number of unique event codes: greater than 0 for files with events.
Number of unique event codes x 4	36	contiguous 4-byte values	Sorted list of 4-character event codes. The number of codes is equal to the Number of unique event codes.
EEG DATA & EVENTS			
size of SSR in bytes x Number of samples	36 + (4 x number of unique event codes)	varies	Consecutive SSRs (See Table B-4)

Table B-9: Net Station Segmented Simple Binary Format Without Events			
HEADER			
Size (bytes)	**File Offset**	**Data Type**	**Description of Data**
4	0	long	Version number: if 3, EEG data are signed short integer; if 5, EEG data are single-precision float; if 7, EEG data are double-precision float.
2	4	short	Recording time: year (4-digit value)
2	6	short	Recording time: month
2	8	short	Recording time: day
2	10	short	Recording time: hour
2	12	short	Recording time: minute
2	14	short	Recording time: second
4	16	long	Recording time: millisecond
2	20	short	Sampling rate (samples per second)
2	22	short	Number of channels
2	24	short	Board gain (1, 2, 4, or 8)
2	26	short	Number of conversion **bits**
2	28	short	Full-scale **range** of amplifier in μV
2	30	short	Number of category names
Size in bytes of packed array of category names	32	pstring array	Packed array of category names, each name encoded as a pstring
2	varies	short	Number of segments
4	varies	long	Number of samples per segment

Table B-9: Net Station Segmented Simple Binary Format Without Events (Continued)

2	varies	short	Number of unique event codes: 0 for files without events.
		EEG DATA	
Size of a Segment in bytes x Number of segments	varies	varies	Consecutive Segments (See Table B-5)

Table B-10: Net Station Segmented Simple Binary Format With Events			
HEADER			
Size (bytes)	**File Offset**	**Data Type**	**Description of Data**
4	0	long	Version number: if 3, EEG data are signed short integer; if 5, EEG data are single-precision float; if 7, EEG data are double-precision float.)
2	4	short	Recording time: year (4-digit value)
2	6	short	Recording time: month
2	8	short	Recording time: day
2	10	short	Recording time: hour
2	12	short	Recording time: minute
2	14	short	Recording time: second
4	16	long	Recording time: millisecond
2	20	short	Sampling rate (samples per second)
2	22	short	Number of channels
2	24	short	Board gain (1, 2, 4, or 8)
2	26	short	Number of conversion **bits**
2	28	short	Full-scale **range** of amplifier in μV
2	30	short	Number of category names
size in bytes of the packed array of category names	32	pstring array	Packed array of category names, each name encoded as a pstring
2	varies	short	Number of segments
4	varies	long	Number of samples per segment
2	varies	short	Number of unique event codes: greater than 0 for files with events.

Table B-10: Net Station Segmented Simple Binary Format With Events (Continued)

Number of unique event codes x 4	varies	contiguous 4-byte values	Sorted list of 4-character event codes. The number of codes is equal to the value in the Number of unique event codes field.
EEG DATA & EVENTS			
Size of a Segment in bytes x Number of segments	varies	varies	Consecutive Segments (See Table B-6)