natus.

natus.
neurology

EEG File Format

Project: **NEUROWORKS EEG**

Date: **Aug 31, 2015**

Revision: **Q**

(See revision history at end of document)

CONFIDENTIAL

Table of Contents

1. Introduction

Each EEG study generates four files with the same name, but unique extensions. These four files are known as the patient information, raw data, notes and the table of contents (toc) files with the extensions .eeg, .erd, .ent and .etc respectively.

Each file has two schema numbers which are used to identify version changes to the file format. The information in this document is valid only for the indicated schema numbers.

1.1 Generic File Header: Base Schema 0

Each file begins with a generic file header defined as follows:

Offset (bytes)	Size (bytes)	Name	Type	Description
0	16	m_file_guid	GUID	Globally unique identifier defining the file type. (see below)
16	4	m_file_schema	WORD	Schema number of the file, used to identify version changes.
20	4	m_creation_time	time_t	Creation time in UTC format.
24	4	m_product_version_high	DWORD	NeuroWorks product version – high bytes.
28	4	m_product_version_low	DWORD	NeuroWorks product version – low bytes.
32	80	m_pat_last_name	char [80]	Last name of the patient.
112	80	m_pat_first_name	char [80]	First name of the patient.
192	80	m_pat_middle_name	char [80]	Middle name of the patient.
272	80	m_pat_id	char [80]	ID of the patient.
352				

1.2 Generic File Header : Base Schema 1

Each file begins with a generic file header defined as follows:

Offset (bytes)	Size (bytes)	Name	Type	Description
0	16	m_file_guid	GUID	Globally unique identifier defining the file type. (see below)
16	2	m_file_schema	WORD	Schema number of the file, used to identify version changes.
18	2	m_base_schema	WORD	Schema number of this generic header, common to all headers.
20	4	m_creation_time	time_t	Creation time in UTC format.
24	4	m_patient_id	long	Internal patient ID.
28	4	m_study_id	long	Internal study ID.
32	80	m_pat_last_name	char [80]	Last name of the patient.
112	80	m_pat_first_name	char [80]	First name of the patient.
192	80	m_pat_middle_name	char [80]	Middle name of the patient.
272	80	m_pat_id	char [80]	ID of the patient.
352				

The valid guids are defined as follows:

File Type	GUID (hex)

Patient Information (.eeg)	ca0b0790-d795-11d0-af32-00a0245b54a5
Raw Data (.erd)	ca0b0791-d795-11d0-af32-00a0245b54a5
Table of Contents (.etc)	ca0b0792-d795-11d0-af32-00a0245b54a5
Notes (.ent)	ca0b0793-d795-11d0-af32-00a0245b54a5
Syncronization File (.snc)	6086a9d2-60af-11d3-9860-00104b75c151
Segments Table of Contents (.stc)	e9b71278-a84b-4d59-bf87-10728fb8bd9d

1.3 List Format

Some files encode data in the Excel list format This is a string representation of data in an organized fashion.

A list consists of a series of items. An item can be either another list, or an atomic item. The atomic items contain data in a specific format. The table below outlines the recognized items and their string representations.

List Item	String Representation
Integer **I()**	A signed 4 byte integer number, e.g.: 234 -999
Real **R()**	A signed 8 byte floating point number, e.g.: 3.1415 -4.625E06
String **S()**	A string delimited by quotes. The sequence '\"' is used to indicate a quote within the string. The sequence '\\' is used to indicate a single backslash within the string. E.g.: "Hello" "This character \" is a quote and this \\ is a backslash"
User **U()**	Hex encoded binary data. The first two characters are '0x' followed by 8 hex digits indicating the size of the data. The data immediately follows in hex with two characters per byte. E.g.: 0x0000000401020ab4
List **L()**	A list of items delimited by parentheses and separated by commas. White space characters are ignored. E.g.: (234, 3.1414, "Hello") ("This List", ("Contains", "A List!"))
Key Tree **K(<symbolic name>)**	A hierarchy of name value pairs in list format. A key tree is delimited by parentheses as is a list. The opening parenthesis is followed by the '.' Character. A key tree consists of a list of nodes. Each node is a list where the first item is the key name in string format. The second item is optional and corresponds to the data at that location. The data may be any type of item, including another key tree. E.g.: (. (("Weight", 150), ("Height", 68)))
GUID **G()**	Unique identifier. Elements in GUID structure delimited by forward slash '/'. (."GroupId", -441356710/37853/4565/-91/-111/0/1/2/120/104/11)

2. Patient Information File – Schema 3

The patient information file consists of a single null terminated string following the generic file header. The string encodes all the patient information in a list format defined by a hierarchy of name value pairs.

The patient information consists of a single key tree with the symbolic name PatientInfo. The table below indicates the key name, the data type and a description. Compound key names are separated by '.' Characters. These refer to sub key trees under the given node.

2.1 K(PatientInfo)

Key	Type	Description
Schema	I()	Schema number of the patient info.
Connections.Sink.Host	S()	Computer name of the storage server.
Connections.Source.Host	S()	Computer name of the signal server.
Info.Name.LastName	S()	
Info.Name.MiddleName	S()	
Info.Name.FirstName	S()	
Info.Address.Address1	S()	
Info.Address.Address2	S()	
Info.Address.City	S()	
Info.Address.State	S()	
Info.Address.Country	S()	
Info.Address.ZIP	S()	
Info.Address.StateLabel	S()	
Info.Address.ZIPLabel	S()	
Info.Admin.Telephone	S()	
Info.Admin.ID	S()	
Info.Admin.BillingID	S()	
Info.Admin.TelephoneLabel	S()	
Info.Admin.IDLabel	S()	
Info.Admin.BillingIDLabel	S()	
Info.Personal.Hand	S()	["Left" | "Right" | ""]
Info.Personal.Gender	S()	["Male" | "Female" | ""]
Info.Personal.BirthDate	R()	Microsoft DATE format.
Info.Personal.Height	R()	Units of centimeters.
Info.Personal.Weight	R()	Units of kilograms.
Info.Personal.BirthDateLabel	S()	
Info.Custom.Field1	S()	
Info.Custom.Field2	S()	
Info.Custom.Field3	S()	
Info.Custom.Field4	S()	
Info.Custom.Field5	S()	
Info.Custom.Label1	S()	Title of field1.
Info.Custom.Label2	S()	Title of field2.
Info.Custom.Label3	S()	Title of field3.
Info.Custom.Label4	S()	Title of field4.
Info.Custom.Label5	S()	Title of field5.
Study	L(K(StudyInfo))	List of study information in reverse order of creation time. The head item is the current study.

2.2 K(StudyInfo)

Key	Type	Description
FileName	S()	Full network portable path name.
CreationTime	R()	Microsoft DATE format.
XLTimeZoneInfo	U()	Microsoft TIMEZONEINFO structure recorded with XLCreationTime. Use to convert UTC times to patient locale time.
XLCreationTime	U()	XLTime (currently FILETIME) UTC added in NeuroWorks 2.05

ModificationTime	R()	Microsoft DATE format.
Duration	I()	Milliseconds.
Creator	S()	
Reviewer	S()	Current reviewer.
EegNo	S()	Eeg/Study number
AcquisitionInstrument	S()	Acquisition machine host name.
Headbox	K(Headboxes)	
DSP	K(DspInfo)	
ReviewerInfo	L(K(ReviewerInfo))	List of reviewer information.
Medication	L(K(MedInfo))	List of medication information.
TextField*	K(S(field))	Key tree of user defined text fields in rich text format.
TechField	K(S(field))	Key tree of user defined text fields in rich text format.
PhysField	K(S(field))	Key tree of user defined text fields in rich text format.
Score	K(ScoreChart)	Scoring chart key tree.
ProductVersionHigh	I()	NeuroWorks version – high bytes.
ProductVersionLow	I()	NeuroWorks version – low bytes.
PatientStatus.Normal	I()	BOOL [0	1]
PatientStatus.MentallyChallenged	I()	BOOL [0	1]
PatientStatus.Awake	I()	BOOL [0	1]
PatientStatus.Drowsy	I()	BOOL [0	1]
PatientStatus.Asleep	I()	BOOL [0	1]
PatientStatus.Uncooperative	I()	BOOL [0	1]
PatientStatus.Tense	I()	BOOL [0	1]
PatientStatus.Confused	I()	BOOL [0	1]
PatientStatus.BehaviorDifficulty	I()	BOOL [0	1]
PatientStatus.Aphasic	I()	BOOL [0	1]
PatientStatus.SemiComa	I()	BOOL [0	1]
PatientStatus.Coma	I()	BOOL [0	1]
PatientStatus.StatusEpilepticus	I()	BOOL [0	1]
PatientStatus.Other	S()	
LastAttack	R()	Microsoft DATE format.
LastMeal	R()	Microsoft DATE format.
Electrode.Type	I()	[0:disc	1:collodian	2:other]
Electrode.OtherType	S()	
Electrode.Placement	I()	[0:10-20	1:manual]
Electrode.SpecialLeads	S()	
Electrode.Impedance	S()	
SkullDefects.Background.BitmapHeader	U()	BITMAP structure for the skull picture.
SkullDefects.Foreground.BitmapHeader	U()	BITMAP structure for the defects picture.
SkullDefects.Background.Bitmap	U()	Cbitmap::GetBitmapBits()

---

* No longer defined as of revision C of this document.

SkullDefects.Foreground.Bitmap	U()	Cbitmap::GetBitmapBits()

2.3 K(DspInfo)

Key	Type	Description
DSPHWVersion	S()	Hardware version in the format "major_rev.minor_rev"
DSPSWVersion	S()	Software version in the format "major_rev.minor_rev"

2.4 K(Headboxes)

Key	Type	Description
HB0	K(HeadboxInfo)	(optional)
HB1	K(HeadboxInfo)	(optional)
HB2	K(HeadboxInfo)	(optional)
HB3	K(HeadboxInfo)	(optional)

2.5 K(HeadboxInfo)

Key	Type	Description
HBSerialNumber	I()	Headbox serial number.
HBType	I()	Headbox type.
HBSWVersion	S()	Software version in the format "major_rev.minor_rev"
HBIsCalibrated	I()	[TRUE	FALSE]
HBCalDate	I()	UTC format date of last calibration.
HBCalSWVersionHigh	I()	Version of the calibration program. – high bytes.
HBCalSWVersionLow	I()	Version of the calibration program. – low bytes.

2.6 K(ReviewerInfo)

Key	Type	Description
Name	S()	Reviewer's name.
ReviewTime	R()	Time marked as reviewed. Microsoft DATE format.

2.7 K(MedInfo)

Key	Type	Description
Name	S()	Medication name.
Dosage	S()	
StartDate	R()	Microsoft DATE format.
Comment	S()	

2.8 K(ScoreChart)

Key	Type	Description
Entries	L(K(ScoreEntries))	

2.9 K(ScoreEntries)

Key	Type	Description
Time	I()	Time from start of study. [milliseconds]
Title	S()	
Comment	S()	
Duration	I()	[milliseconds]
ShowWaveforms	I()	[TRUE	FALSE]
NoteID	I()	ID of the annotation from which this entry was derived. [> 0]
Montage		K(Montage)

3. Raw Data File

The raw data file (.erd) encodes the sample data in a binary format.

3.1 Raw Data File Header- Schema 5

The raw data file begins with a generic file header followed by a raw data header defined as follows:

Offset (bytes)	Size (bytes)	Name	Type	Description
352	8	m_sample_freq	double	Sample frequency in Hertz.
360	4	m_num_channels	int	Number of channels stored.
364	4	m_deltabits	int	Number of bits of delta information stored per channel. **Currently only a value of 8 is supported.**
368	128	m_phys_chan	int [32]	Storage order to physical channel assignments.
496	16	m_headbox_type	int [4]	Type of the headboxes.
512	16	m_headbox_sn	int [4]	Serial number of the headboxes.
528	40	m_headbox_sw_version	char [4][10]	Headbox software version number: (major_rev.minor_rev).
568	10	m_dsp_hw_version	char [10]	DSP Hardware version number.
578	10	m_dsp_sw_version	char [10]	DSP Software version number.
588	4	m_discardbits	int	Number of least significant bits to discard when storing.
592				

3.2 Raw Data File Header- Schema 6

The raw data file begins with a generic file header followed by a raw data header defined as follows:

Offset (bytes)	Size (bytes)	Name	Type	Description

Offset (bytes)	Size (bytes)	Name	Type	Description
352	8	m_sample_freq	double	Sample frequency in Hertz.
360	4	m_num_channels	int	Number of channels stored.
364	4	m_deltabits	int	Number of bits of delta information stored per channel. **Currently only a value of 8 is supported.**
368	512	m_phys_chan	int [128]	Storage order to physical channel assignments.
880	16	m_headbox_type	int [4]	Type of the headboxes.
896	16	m_headbox_sn	int [4]	Serial number of the headboxes.
912	40	m_headbox_sw_version	char [4][10]	Headbox software version number: (major_rev.minor_rev).
952	10	m_dsp_hw_version	char [10]	DSP Hardware version number.
962	10	m_dsp_sw_version	char [10]	DSP Software version number.
972	4	m_discardbits	int	Number of least significant bits to discard when storing.
976				

3.3 Raw Data File Header- Schema 7

The raw data file begins with a generic file header followed by a raw data header defined as follows:

Offset (bytes)	Size (bytes)	Name	Type	Description
352	8	m_sample_freq	double	Sample frequency in Hertz.
360	4	m_num_channels	int	Number of channels stored.
364	4	m_deltabits	int	Number of bits of delta information stored per channel. **Currently only a value of 8 is supported.**
368	4096	m_phys_chan	int [1024]	Storage order to physical channel assignments.
4464	16	m_headbox_type	int [4]	Type of the headboxes.
4480	16	m_headbox_sn	int [4]	Serial number of the headboxes.
4496	40	m_headbox_sw_version	char [4][10]	Headbox software version number: (major_rev.minor_rev).
4536	10	m_dsp_hw_version	char [10]	DSP Hardware version number.
4546	10	m_dsp_sw_version	char [10]	DSP Software version number.
4556	4	m_discardbits	int	Number of least significant bits to discard when storing.
4560				

The sample data is encoded using the n-bit delta compression algorithm. For each sample, only the lower n bits of the difference between the last and the current sample are stored. If the difference exceeds the range represented by n bits then a full 32 bit value is stored, thereby ensuring lossless compression.

The raw data file consists of a series of sample packets. A sample packet consists of the sampled value of all channels being stored and an event byte

Each sample packet is stored in three parts:

1.   Event Byte

2. Delta Information
3. Absolute Channel Values (if any)

3.3.1 Event Byte

Bit 0 of the event byte indicates the presence of the external trigger during the sample period. At present, a photic stimulator is the only supported device which generates an external trigger.

3.3.2 Delta Information

This is an array of *m_num_channels* of deltas representing the difference between the current and the last channel value. Each element of the array is *n* bits long. The total array is rounded up to an integer number of bytes. If the difference is greater than *max_delta* or less than *min_delta* then the value *abs_delta* is placed in the array.

$$Max\_delta = 2^{n-1} - 1$$
$$min\_delta = 1 - 2^{n-1}$$
$$abs\_delta = 2^{n-1}$$

3.3.3 Absolute Channel Values

For each element of the delta array that is set to *abs_delta* a 32 bit integer absolute channel value is stored. If an element of the delta array is not set to *abs_delta* then there is no absolute channel value stored.

Values written to the raw data file consist of undecimated and unfiltered sample data. The conversion factor to engineering units is variable, dependent upon m_headbox_type and m_headbox_sw_version, and which physical channel number is being examined:

m_headbox_type	m_sw_version	Channel Range	Conversion Factor (microvolts per raw sample unit)
1 (EEG32)	All	All	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
3 (EEG128)	All	All	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
6 (EMU36)	All	0-31	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		32-35	$((5000000./(2^{10} - 0.5)) / 2^6) * 2^{m\_discardbits}$
4 (AMB28)	All	0-23	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		24-27	$((5000000./(2^{10} - 0.5)) / 2^6) * 2^{m\_discardbits}$
9 (MOBEE32)	All	0-32	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		33-34	$(1/2^6) * 2^{m\_discardbits}$
8 (MOBEE24)	All	0-24	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		25-26	$(1/2^6) * 2^{m\_discardbits}$
5 (HYPPO)	All	0-25	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		26-31	$((8711. / (2^{21} - 0.5)) / (159.8/249.5)) * 2^{m\_discardbits}$
		40-41	$(1/2^6) * 2^{m\_discardbits}$
	< 3.4	32-39	$((10000000./(2^{10} - 0.5)) / 2^6) * 2^{m\_discardbits}$
	3.4 +	32-39	$(20000000./65536.) * 2^{m\_discardbits}$

3.4 Raw Data File Header- Schema 8

The raw data file begins with a generic file header followed by a raw data header defined as follows:

Offset (bytes)	Size (bytes)	Name	Type	Description
352	8	m_sample_freq	double	Sample frequency in Hertz.
360	4	m_num_channels	int	Number of channels stored.

364	4	m_deltabits	int	Number of bits of delta information stored per channel. **Currently only a value of 8 is supported.**
368	4096	m_phys_chan	int [1024]	Storage order to physical channel assignments.
4464	16	m_headbox_type	int [4]	Type of the headboxes.
4480	16	m_headbox_sn	int [4]	Serial number of the headboxes.
4496	40	m_headbox_sw_version	char [4][10]	Headbox software version number: (major_rev.minor_rev).
4536	10	m_dsp_hw_version	char [10]	DSP Hardware version number.
4546	10	m_dsp_sw_version	char [10]	DSP Software version number.
4556	4	m_discardbits	int	Number of least significant bits to discard when storing.
4560	2048	m_shorted	Short [1024]	0's indicate recorded channels, 1's are channels being recorded
6608	2048	m_frequency_factor	Short [1024]	Frequency multipliers: SHRT_MAX indicates headbox frequency, 20 – every 20th sample, 4 – every 4th sample etc.
8656				

M_shorted and m_frequency_factor allow us not to record shorted channels and record channels with sampling frequency lower than the headbox sampling frequency thus reducing file size.

The sample data is encoded using the n-bit delta compression algorithm. For each sample, only the lower n or 2*n bits of the difference between the last and the current sample are stored. If the difference exceeds the range represented by 2*n bits then a full 32 bit value is stored, thereby ensuring lossless compression.

The raw data file consists of a series of sample packets. A sample packet consists of the sampled value of all channels being stored and an event byte

Each sample packet is stored in three parts:

4. Event Byte
5. Frequency byte (Present only if at least one of the m_frequency_factor values in the header is different from SHRT_MAX thus indicating that some channels have been recorded with frequency different from headbox sampling frequency)
6. Delta mask
7. Delta Information
8. Absolute Channel Values (if any)

3.4.1 Event Byte

Bit 0 of the event byte indicates the presence of the external trigger during the sample period. At present, a photic stimulator is the only supported device which generates an external trigger.

3.4.2 Frequency Byte

Optional. This byte is written only if at least one of the m_frequency_factor fields in the header is different from SHRT_MAX, thus indicating that this erd file has channels with recording frequency different from the headbox sampling frequency.
Bit7 – all channels are being writte

Bit5 – Channels with recording frequency of 1/50 of the sampling frequency are written (currently not used)

Bit4 – Channels with recording frequency of 1/20 of the sampling frequency are written
Bit3 – Channels with recording frequency of 1/10 of the sampling frequency are written (currently not used)
Bit2 – Channels with recording frequency of 1/5 of the sampling frequency are written (currently not used)
Bit1 – Channels with recording frequency of 1/4 of the sampling frequency are written
Bit0 – Channels with recording frequency of 1/2 of the sampling frequency are written (currently not used)

Currently variable frequency recording is used for SLEEP studies only.

3.4.3 Delta mask

Bit-mask of a size int(number_of_channels / 8 + 0.5). Each 1 in the mask indicates that corresponding channel has 2*n bit delta, 0 means that corresponding channel has n bit delta.

3.4.4 Delta Information

This is an array of (*m_num_channels – number_of_ shortedchannels – channels_with_different_ recording_ frequencies_ that_ are_ not_ recorded_at_ this_ time)* of deltas representing the difference between the current and the last channel value. Each element of the array is *n* or 2*n bits long. The total array is rounded up to int(number_of_channels / 8 + 0.5). If the difference is greater than *max_delta* or less than *min_delta* then the value *abs_delta* is placed in the array.

$$Max\_delta = 2^{n*2-1} - 1$$
$$min\_delta = 1 - 2^{n*2-1}$$
$$abs\_delta = FFFF$$

3.4.5 Absolute Channel Values

For each element of the delta array that is set to *abs_delta* a 32 bit integer absolute channel value is stored. If an element of the delta array is not set to *abs_delta* then there is no absolute channel value stored.

Values written to the raw data file consist of undecimated and unfiltered sample data. The conversion factor to engineering units is variable, dependent upon m_headbox_type and m_headbox_sw_version, and which physical channel number is being examined:

m_headbox_type	m_sw_version	Channel Range	Conversion Factor (microvolts per raw sample unit)
1 (EEG32)	All	All	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
3 (EEG128)	All	All	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
6 (EMU36)	All	0-31	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		32-35	$((5000000./(2^{10} - 0.5)) / 2^6) * 2^{m\_discardbits}$
4 (AMB28)	All	0-23	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		24-27	$((5000000./(2^{10} - 0.5)) / 2^6) * 2^{m\_discardbits}$
9 (MOBEE32)	All	0-32	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		33-34	$(1/2^6) * 2^{m\_discardbits}$
8 (MOBEE24)	All	0-24	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		25-26	$(1/2^6) * 2^{m\_discardbits}$
5 (HYPPO)	All	0-25	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		26-31	$((8711. / (2^{21} - 0.5)) / (159.8/249.5)) * 2^{m\_discardbits}$
		40-41	$(1/2^6) * 2^{m\_discardbits}$
	< 3.4	32-39	$((10000000./(2^{10} - 0.5)) / 2^6) * 2^{m\_discardbits}$
	3.4 +	32-39	$(20000000./65536.) * 2^{m\_discardbits}$

The *m_phys_chan* array (see Raw Data File Header) indicates the physical channel number of the *m_num_channels* sample values. That is, the first delta value in the delta array corresponds to the physical channel m_phys_chan[0]. The second delta value corresponds to physical channel m_phys_chan[1], etc.

The physical channels are labeled as follows depending on *m_headbox_type[0]* in the ERD file header. In case of unknown headbox (not 1,3,4,5,6,8, or 9) a generic label (C*n*) should be used where *n* is channel index (1-based).

Channel index	EEG32 Type=1	EEG128 Type=3	EMU36 Type=6	AMB28 Type=4	MOBEE32 Type=9	MOBEE24 Type=8	HYPPO Type=5	EMU256 Quantum I Type=20	EMU512 Quantum II Type=20
0	C3	C1	AC1	AC1	Ref	Ref	C3	C1	C1
1	C4	C2	AC2	AC2	Fp1	Fp1	C4	C2	C2
2	CZ	C3	Ref	Ref	F7	F7	O1	C3	C3
3	F3	C4	Fp1	Fp1	T3	T3	O2	C4	C4
4	F4	C5	F7	F7	A1	A1	A1	C5	C5
5	F7	C6	T3	T3	T5	T5	A2	C6	C6
6	F8	C7	T5	T5	O1	O1	LOC	C7	C7
7	FZ	C8	O1	O1	F3	F3	ROC	C8	C8
8	FP1	C9	F3	F3	C3	C3	CHIN1	C9	C9
9	FP2	C10	C3	C3	P3	P3	CHIN2	C10	C10
10	FPZ	C11	P3	P3	Fpz	Fpz	ECGL	C11	C11
11	O1	C12	Fz	Fz	Fz	Fz	ECGR	C12	C12
12	O2	C13	Cz	Cz	Cz	Cz	LAT1	C13	C13
13	P3	C14	Pz	Pz	Pz	Pz	RAT1	C14	C14
14	P4	C15	F4	F4	Fp2	Fp2	LAT2	C15	C15
15	PZ	C16	C4	C4	F8	F8	RAT2	C16	C16
16	T3	C17	P4	P4	T4	T4	X1	C17	C17
17	T4	C18	Fp2	Fp2	A2	A2	X2	C18	C18
18	T5	C19	F8	F8	T6	T6	X3	C19	C19
19	T6	C20	T4	T4	O2	O2	X4	C20	C20
20	AUX1	C21	T6	T6	F4	F4	X5	C21	C21
21	AUX2	C22	O2	O2	C4	C4	X6	C22	C22
22	AUX3	C23	AC23	AC23	P4	P4	X7	C23	C23
23	AUX4	C24	AC24	AC24	X1	X1	X8	C24	C24
24	AUX5	C25	AC25	DC1	X2	X2	X9	C25	C25
25	AUX6	C26	AC26	DC2	X3		X10	C26	C26
26	AUX7	C27	AC27	DC3	X4		CHEST	C27	C27
27	AUX8	C28	AC28	DC4	X5		ABD	C28	C28
28	PG1	C29	AC29		X6		FLOW	C29	C29
29	PG2	C30	AC30		X7		SNORE	C30	C30
30	A1	C31	AC31		X8		DIF1	C31	C31
31	A2	C32	AC32		X9		DIF2	C32	C32
32		C33	DC1		X10		POSITION	C33	C33
33		C34	DC2				PRES	C34	C34
34		C35	DC3				DC1	C35	C35

35		C36	DC4				DC2	C36	C36
36		C37					DC3	C37	C37
37		C38					DC4	C38	C38
38		C39					DC5	C39	C39
39		C40					DC6	C40	C40
40		C41					OSAT	C41	C41
41		C42					PR	C42	C42
42		C43						C43	C43
43		C44						C44	C44
44		C45						C45	C45
45		C46						C46	C46
46		C47						C47	C47
47		C48						C48	C48
48		C49						C49	C49
49		C50						C50	C50
50		C51						C51	C51
51		C52						C52	C52
52		C53						C53	C53
53		C54						C54	C54
54		C55						C55	C55
55		C56						C56	C56
56		C57						C57	C57
57		C58						C58	C58
58		C59						C59	C59
59		C60						C60	C60
60		C61						C61	C61
61		C62						C62	C62
62		C63						C63	C63
63		C64						C64	C64
64		C65						C65	C65
65		C66						C66	C66
66		C67						C67	C67
67		C68						C68	C68
68		C69						C69	C69
69		C70						C70	C70
70		C71						C71	C71
71		C72						C72	C72
72		C73						C73	C73
73		C74						C74	C74
74		C75						C75	C75
75		C76						C76	C76
76		C77						C77	C77
77		C78						C78	C78
78		C79						C79	C79
79		C80						C80	C80

80		C81						C81	C81
81		C82						C82	C82
82		C83						C83	C83
83		C84						C84	C84
84		C85						C85	C85
85		C86						C86	C86
86		C87						C87	C87
87		C88						C88	C88
88		C89						C89	C89
89		C90						C90	C90
90		C91						C91	C91
91		C92						C92	C92
92		C93						C93	C93
93		C94						C94	C94
94		C95						C95	C95
95		C96						C96	C96
96		C97						C97	C97
97		C98						C98	C98
98		C99						C99	C99
99		C100						C100	C100
100		C101						C101	C101
101		C102						C102	C102
102		C103						C103	C103
103		C104						C104	C104
104		C105						C105	C105
105		C106						C106	C106
106		C107						C107	C107
107		C108						C108	C108
108		C109						C109	C109
109		C110						C110	C110
110		C111						C111	C111
111		C112						C112	C112
112		C113						C113	C113
113		C114						C114	C114
114		C115						C115	C115
115		C116						C116	C116
116		C117						C117	C117
117		C118						C118	C118
118		C119						C119	C119
119		C120						C120	C120
120		C121						C121	C121
121		C122						C122	C122
122		C123						C123	C123
123		C124						C124	C124
124		C125						C125	C125

125		C126						C126	C126
126		C127						C127	C127
127		C128						C128	C128
128		OSAT						C129	C129
129		PR						C130	C130
130		C131						C131	C131
131		C132						C132	C132
132		C133						C133	C133
133		C134						C134	C134
134		C135						C135	C135
135		C136						C136	C136
136		C137						C137	C137
137		C138						C138	C138
138		C139						C139	C139
139		C140						C140	C140
140		C141						C141	C141
141		C142						C412	C412
142		C143						C143	C143
143		C144						C144	C144
144		C145						C145	C145
145		C146						C146	C146
146		C147						C147	C147
147		C148						C148	C148
148		C149						C149	C149
149		C150						C150	C150
150		C151						C151	C151
151		C152						C152	C152
152		C153						C153	C153
153		C154						C154	C154
154		C155						C155	C155
155		C156						C156	C156
156		C157						C157	C157
157		C158						C158	C158
158		C159						C159	C159
159		C160						C160	C160
160		C161						C161	C161
161		C162						C162	C162
162		C163						C163	C163
163		C164						C164	C164
164		C165						C165	C165
165		C166						C166	C166
166		C167						C167	C167
167		C168						C168	C168
168		C169						C169	C169
169		C170						C170	C170

170		C171						C171	C171
171		C172						C172	C172
172		C173						C173	C173
173		C174						C174	C174
174		C175						C175	C175
175		C176						C176	C176
176		C177						C177	C177
177		C178						C178	C178
178		C179						C179	C179
179		C180						C180	C180
180		C181						C181	C181
181		C182						C182	C182
182		C183						C183	C183
183		C184						C184	C184
184		C185						C185	C185
185		C186						C186	C186
186		C187						C187	C187
187		C188						C188	C188
188		C189						C189	C189
189		C190						C190	C190
190		C191						C191	C191
191		C192						C192	C192
192		C193						C193	C193
193		C194						C194	C194
194		C195						C195	C195
195		C196						C196	C196
196		C197						C197	C197
197		C198						C198	C198
198		C199						C199	C199
199		C200						C200	C200
200		C201						C201	C201
201		C202						C202	C202
202		C203						C203	C203
203		C204						C204	C204
204		C205						C205	C205
205		C206						C206	C206
206		C207						C207	C207
207		C208						C208	C208
208		C209						C209	C209
209		C210						C210	C210
210		C211						C211	C211
211		C212						C212	C212
212		C213						C213	C213
213		C214						C214	C214
214		C215						C215	C215

215		C216						C216	C216
216		C217						C217	C217
217		C218						C218	C218
218		C219						C219	C219
219		C220						C220	C220
220		C221						C221	C221
221		C222						C222	C222
222		C223						C223	C223
223		C224						C224	C224
224		C225						C225	C225
225		C226						C226	C226
226		C227						C227	C227
227		C228						C228	C228
228		C229						C229	C229
229		C230						C230	C230
230		C231						C231	C231
231		C232						C232	C232
232		C233						C233	C233
233		C234						C234	C234
234		C235						C235	C235
235		C236						C236	C236
236		C237						C237	C237
237		C238						C238	C238
238		C239						C239	C239
239		C240						C240	C240
240		C241						C241	C241
241		C242						C242	C242
242		C243						C243	C243
243		C244						C244	C244
244		C245						C245	C245
245		C246						C246	C246
246		C247						C247	C247
247		C248						C248	C248
248		C249						C249	C249
249		C250						C250	C250
250		C251						C251	C251
251		C252						C252	C252
252		C253						C253	C253
253		C254						C254	C254
254		C255						C255	C255
255		C256						C256	C256
256								DC1	C257
257								DC2	C258
258								DC3	C259
259								DC4	C260

260								DC5	C261
261								DC6	C262
262								DC7	C263
263								DC8	C264
264								DC9	C265
265								DC10	C266
266								DC11	C267
267								DC12	C268
268								DC13	C269
269								DC14	C270
270								DC15	C271
271								DC16	C272
272								TRIG	C273
273								OSAT	C274
274								PR	C275
275								Pleth	C276
276									C277
277									C278
278									C279
279									C280
280									C281
281									C282
282									C283
283									C284
284									C285
285									C286
286									C287
287									C288
288									C289
289									C290
290									C291
291									C292
292									C293
293									C294
294									C295
295									C296
296									C297
297									C298
298									C299
299									C300
300									C301
301									C302
302									C303
303									C304
304									C305

305								C306
306								C307
307								C308
308								C309
309								C310
310								C311
311								C312
312								C313
313								C314
314								C315
315								C316
316								C317
317								C318
318								C319
319								C320
320								C321
321								C322
322								C323
323								C324
324								C325
325								C326
326								C327
327								C328
328								C329
329								C330
330								C331
331								C332
332								C333
333								C334
334								C335
335								C336
336								C337
337								C338
338								C339
339								C340
340								C341
341								C342
342								C343
343								C344
344								C345
345								C346
346								C347
347								C348
348								C349
349								C350

350									C351
351									C352
352									C353
353									C354
354									C355
355									C356
356									C357
357									C358
358									C359
359									C360
360									C361
361									C362
362									C363
363									C364
364									C365
365									C366
366									C367
367									C368
368									C369
369									C370
370									C371
371									C372
372									C373
373									C374
374									C375
375									C376
376									C377
377									C378
378									C379
379									C380
380									C381
381									C382
382									C383
383									C384
384									C385
385									C386
386									C387
387									C388
388									C389
389									C390
390									C391
391									C392
392									C393
393									C394
394									C395

395									C396
396									C397
397									C398
398									C399
399									C400
400									C401
401									C402
402									C403
403									C404
404									C405
405									C406
406									C407
407									C408
408									C409
409									C410
410									C411
411									C412
412									C413
413									C414
414									C415
415									C416
416									C417
417									C418
418									C419
419									C420
420									C421
421									C422
422									C423
423									C424
424									C425
425									C426
426									C427
427									C428
428									C429
429									C430
430									C431
431									C432
432									C433
433									C434
434									C435
435									C436
436									C437
437									C438
438									C439
439									C440

440								C441
441								C442
442								C443
443								C444
444								C445
445								C446
446								C447
447								C448
448								C449
449								C450
450								C451
451								C452
452								C453
453								C454
454								C455
455								C456
456								C457
457								C458
458								C459
459								C460
460								C461
461								C462
462								C463
463								C464
464								C465
465								C466
466								C467
467								C468
468								C469
469								C470
470								C471
471								C472
472								C473
473								C474
474								C475
475								C476
476								C477
477								C478
478								C479
479								C480
480								C481
481								C482
482								C483
483								C484
484								C485

485								C486
486								C487
487								C488
488								C489
489								C490
490								C491
491								C492
492								C493
493								C494
494								C495
495								C496
496								C497
497								C498
498								C499
499								C500
500								C501
501								C502
502								C503
503								C504
504								C505
505								C506
506								C507
507								C508
508								C509
509								C510
510								C511
511								C512
512								DC1
513								DC2
514								DC3
515								DC4
516								DC5
517								DC6
518								DC7
519								DC8
520								DC9
521								DC10
522								DC11
523								DC12
524								DC13
525								DC14
526								DC15
527								DC16
528								TRIG
529								OSAT

530									PR
531									Pleth

3.5 Raw Data File Header – Schema 9

The raw data file begins with a generic file header followed by a raw data header defined as follows:

Offset (bytes)	Size (bytes)	Name	Type	Description
352	8	m_sample_freq	double	Sample frequency in Hertz.
360	4	m_num_channels	int	Number of channels stored.
364	4	m_deltabits	int	Number of bits of delta information stored per channel. **Currently only a value of 8 is supported.**
368	4096	m_phys_chan	int [1024]	Storage order to physical channel assignments.
4464	16	m_headbox_type	int [4]	Type of the headboxes.
4480	16	m_headbox_sn	int [4]	Serial number of the headboxes.
4496	40	m_headbox_sw_version	char [4][10]	Headbox software version number: (major_rev.minor_rev).
4536	10	m_dsp_hw_version	char [10]	DSP Hardware version number.
4546	10	m_dsp_sw_version	char [10]	DSP Software version number.
4556	4	m_discardbits	int	Number of least significant bits to discard when storing.
4560	2048	m_shorted	Short [1024]	0's indicate recorded channels, 1's are channels being recorded
6608	2048	m_frequency_factor	Short [1024]	Frequency multipliers: SHRT_MAX indicates headbox frequency, 20 – every 20th sample, 4 – every 4th sample etc.
8656				

M_shorted and m_frequency_factor allow us not to record shorted channels and record channels with sampling frequency lower than the headbox sampling frequency thus reducing file size.

The sample data is encoded using the n-bit delta compression algorithm. For each sample, only the lower n or 2*n bits of the difference between the last and the current sample are stored. If the difference exceeds the range represented by 2*n bits then a full 32 bit value is stored, thereby ensuring lossless compression.

The raw data file consists of a series of sample packets. A sample packet consists of the sampled value of all channels being stored and an event byte

Each sample packet is stored in three parts:

9.   Event Byte
10. Frequency byte   (Present only if at least one of the m_frequency_factor values in the header is different from SHRT_MAX thus indicating that some channels have been recorded with frequency different from headbox sampling frequency)
11. Delta mask
12. Delta Information
13. Absolute Channel Values (if any)

3.5.1 Event Byte

Bit 0 of the event byte indicates the presence of the external trigger during the sample period. At present, a photic stimulator is the only supported device which generates an external trigger.

3.5.2 Frequency Byte

Optional. This byte is written only if at least one of the m_frequency_factor fields in the header is different from SHRT_MAX, thus indicating that this erd file has channels with recording frequency different from the headbox sampling frequency.
Bit7 – all channels are being writte

Bit5 – Channels with recording frequency of 1/50 of the sampling frequency are written (currently not used)
Bit4 – Channels with recording frequency of 1/20 of the sampling frequency are written
Bit3 – Channels with recording frequency of 1/10 of the sampling frequency are written (currently not used)
Bit2 – Channels with recording frequency of 1/5 of the sampling frequency are written (currently not used)
Bit1 – Channels with recording frequency of 1/4 of the sampling frequency are written
Bit0 – Channels with recording frequency of 1/2 of the sampling frequency are written (currently not used)

Currently variable frequency recording is used for SLEEP studies only.

3.5.3 Delta mask

Bit-mask of a size int(number_of_channels / 8 + 0.5). Each 1 in the mask indicates that corresponding channel has 2*n bit delta, 0 means that corresponding channel has n bit delta.

3.5.4 Delta Information

This is an array of (*m_num_channels – number_of_ shortedchannels – channels_with_different_ recording_ frequencies_ that_ are_ not_ recorded_at_ this_ time)* of deltas representing the difference between the current and the last channel value. Each element of the array is *n* or 2*n bits long. The total array is rounded up to int(number_of_channels / 8 + 0.5). If the difference is greater than *max_delta* or less than *min_delta* then the value *abs_delta* is placed in the array.

$$\text{Max\_delta} = 2^{n*2-1} - 1$$
$$\text{min\_delta} = 1 - 2^{n*2-1}$$
$$\text{abs\_delta} = FFFF$$

3.5.5 Absolute Channel Values

For each element of the delta array that is set to *abs_delta* a 32 bit integer absolute channel value is stored. If an element of the delta array is not set to *abs_delta* then there is no absolute channel value stored.

Values written to the raw data file consist of undecimated and unfiltered sample data. The conversion factor to engineering units is variable, dependent upon m_headbox_type and m_headbox_sw_version, and which physical channel number is being examined:

m_headbox_type	m_sw_version	Channel Range	Conversion Factor (microvolts per raw sample unit)
1 (EEG32)	All	All	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
3 (EEG128)	All	All	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
6 (EMU36)	All	0-31	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		32-35	$((5000000./(2^{10} - 0.5)) / 2^{6}) * 2^{m\_discardbits}$

4 (AMB28)	All	0-23	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		24-27	$((5000000./(2^{10} - 0.5)) / 2^6) * 2^{m\_discardbits}$
9 (MOBEE32)	All	0-32	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		33-34	$(1 / 2^6) * 2^{m\_discardbits}$
8 (MOBEE24)	All	0-24	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		25-26	$(1 / 2^6) * 2^{m\_discardbits}$
5 (HYPPO)	All	0-25	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		26-31	$((8711. / (2^{21} - 0.5)) / (159.8/249.5)) * 2^{m\_discardbits}$
		40-41	$(1 / 2^6) * 2^{m\_discardbits}$
	< 3.4	32-39	$((10000000./(2^{10} - 0.5)) / 2^6) * 2^{m\_discardbits}$
	3.4 +	32-39	$((20000000./65536.) / 2^6) * 2^{m\_discardbits}$
14 (Connex)	All	0-37	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		38-47	$((10800000./65536.) / 2^6) * 2^{m\_discardbits}$
		48-49	$(1 / 2^6) * 2^{m\_discardbits}$
15 (Trex)	All	0-23	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		24-27	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		28-31	$((10000000. / 65536.) / 2^6) * 2^{m\_discardbits}$
		32-33	$(1 / 2^6) * 2^{m\_discardbits}$
17 (EMU40)	All	0-39	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		40-43	$((10800000. / 65536.) / 2^6) * 2^{m\_discardbits}$
		44-45	$(1 / 2^6) * 2^{m\_discardbits}$
19 (EEG32U)	All	All	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
20 (Quantum)	All	0 … m_num_channels[1] - 20	$(-8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		(m_num_channels – 20) … m_num_channels – 3)	$((-10800000. / 65536.) / 2^6) * 2^{m\_discardbits}$
		m_num_channels – 3) … m_num_channels	$(1 / 2^6) * 2^{m\_discardbits}$
21 (NeuroLink IP)	All	0-127	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		128-129	$(1 / 2^6) * 2^{m\_discardbits}$
		130-255	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
22 (Netlink)	All	0-31	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		32-39	$((10800000. / 65536.) / 2^6) * 2^{m\_discardbits}$
		40-41	$(1 / 2^6) * 2^{m\_discardbits}$
		42	$((10800000. / 65536.) / 2^6) * 2^{m\_discardbits}$
23 (Traveler)	All	0-31	$(8711. / (2^{21} - 0.5)) * 2^{m\_discardbits}$
		32-35	$((10800000. / 65536.) / 2^6) * 2^{m\_discardbits}$
		36-37	$(1 / 2^6) * 2^{m\_discardbits}$
		38	$((10800000. / 65536.) / 2^6) * 2^{m\_discardbits}$

---

[1] Number of channels recorded in the study. The number can be found in the Raw Data file header.

The *m_phys_chan* array (see Raw Data File Header) indicates the physical channel number of the *m_num_channels* sample values. That is, the first delta value in the delta array corresponds to the physical channel m_phys_chan[0]. The second delta value corresponds to physical channel m_phys_chan[1], etc.

The physical channels are labeled as in schema 8, with the addition of the new headboxes listed below, depending on *m_headbox_type[0]* in the ERD file header. In case of unknown headbox (not 1,3,4,5,6,8,9,14,15,17,19,20,21,22 or 23) a generic label (C*n*) should be used where *n* is channel index (1-based).

Chan. Index	Connex EEG/Sleep Type=14	Trex Ambulatory Type=15	EMU40 Ambulatory Type=17	EEG32U (same as EEG32 – Type 1 above) Type=19	NeuroLink IP Type=21	Netlink Type=22	Traveler Ambulatory Type=23	EMU256 Quantum I Type=20	EMU512 Quantum II Type=20
0	C3	Fp1	Fp1	C3	C1	Fp1	Pg1	C1	C1
1	C4	F7	F7	C4	C2	Fp2	Pg2	C2	C2
2	O1	T3	T3	CZ	C3	F7	Eogl	C3	C3
3	O2	A1	T5	F3	C4	F3	Fp1	C4	C4
4	A1	T5	O1	F4	C5	Fz	Fp2	C5	C5
5	A2	O1	F3	F7	C6	F4	Eogr	C6	C6
6	Cz	F3	C3	F8	C7	F8	T1	C7	C7
7	F3	C3	P3	FZ	C8	T3	F7	C8	C8
8	F4	P3	A1	FP1	C9	C3	F3	C9	C9
9	F7	Fpz	Fz	FP2	C10	Cz	Fz	C10	C10
10	F8	Fz	Cz	FPZ	C11	C4	F4	C11	C11
11	Fz	Cz	Fpz	O1	C12	T4	F8	C12	C12
12	Fp1	Pz	Pz	O2	C13	T5	T2	C13	C13
13	Fp2	Fp2	X1	P3	C14	P3	A1	C14	C14
14	Fpz	F8	X2	P4	C15	Pz	T3	C15	C15
15	P3	T4	X3	PZ	C16	P4	C3	C16	C16
16	P4	A2	X4	T3	C17	T6	Cz	C17	C17
17	Pz	T6	X5	T4	C18	O1	C4	C18	C18
18	T3	O2	X6	T5	C19	O2	T4	C19	C19
19	T4	F4	X7	T6	C20	A1	A2	C20	C20
20	T5	C4	X8	AUX1	C21	A2	T5	C21	C21
21	T6	P4	X9	AUX2	C22	Pg1	P3	C22	C22
22	LOC	X1	Fp2	AUX3	C23	Pg2	Pz	C23	C23
23	ROC	X2	X10	AUX4	C24	Oz	P4	C24	C24
24	CHIN1	DIF1	F8	AUX5	C25	X1	T6	C25	C25
25	CHIN2	DIF2	X11	AUX6	C26	X2	O1	C26	C26
26	ECGL	DIF3	T4	AUX7	C27	X3	Oz	C27	C27
27	ECGR	DIF4	X12	AUX8	C28	X4	O2	C28	C28
28	LAT1	DC1	T6	PG1	C29	X5	X1	C29	C29
29	LAT2	DC2	X13	PG2	C30	X6	X2	C30	C30
30	RAT1	DC3	O2	A1	C31	X7	X3	C31	C31
31	RAT2	DC4	X14	A2	C32	X8	X4	C32	C32
32	CHEST	OSAT	F4		C33	DC1	DC1	C33	C33

33	ABD	PR	X15		C34	DC2	DC2	C34	C34
34	FLOW		C4		C35	DC3	DC3	C35	C35
35	SNORE		X16		C36	DC4	DC4	C36	C36
36	DIF5		P4		C37	DC5	OSAT	C37	C37
37	DIF6		X17		C38	DC6	PR	C38	C38
38	POS		A2		C39	DC7	POS	C39	C39
39	DC2		X18		C40	DC8		C40	C40
40	DC3		DC1		C41	OSAT		C41	C41
41	DC4		DC2		C42	PR		C42	C42
42	DC5		DC3		C43	POS		C43	C43
43	DC6		DC4		C44			C44	C44
44	DC7		OSAT		C45			C45	C45
45	DC8		PR		C46			C46	C46
46	DC9				C47			C47	C47
47	DC10				C48			C48	C48
48	OSAT				C49			C49	C49
49	PR				C50			C50	C50
50					C51			C51	C51
51					C52			C52	C52
52					C53			C53	C53
53					C54			C54	C54
54					C55			C55	C55
55					C56			C56	C56
56					C57			C57	C57
57					C58			C58	C58
58					C59			C59	C59
59					C60			C60	C60
60					C61			C61	C61
61					C62			C62	C62
62					C63			C63	C63
63					C64			C64	C64
64					C65			C65	C65
65					C66			C66	C66
66					C67			C67	C67
67					C68			C68	C68
68					C69			C69	C69
69					C70			C70	C70
70					C71			C71	C71
71					C72			C72	C72
72					C73			C73	C73
73					C74			C74	C74
74					C75			C75	C75
75					C76			C76	C76
76					C77			C77	C77
77					C78			C78	C78

78					C79			C79	C79
79					C80			C80	C80
80					C81			C81	C81
81					C82			C82	C82
82					C83			C83	C83
83					C84			C84	C84
84					C85			C85	C85
85					C86			C86	C86
86					C87			C87	C87
87					C88			C88	C88
88					C89			C89	C89
89					C90			C90	C90
90					C91			C91	C91
91					C92			C92	C92
92					C93			C93	C93
93					C94			C94	C94
94					C95			C95	C95
95					C96			C96	C96
96					C97			C97	C97
97					C98			C98	C98
98					C99			C99	C99
99					C100			C100	C100
100					C101			C101	C101
101					C102			C102	C102
102					C103			C103	C103
103					C104			C104	C104
104					C105			C105	C105
105					C106			C106	C106
106					C107			C107	C107
107					C108			C108	C108
108					C109			C109	C109
109					C110			C110	C110
110					C111			C111	C111
111					C112			C112	C112
112					C113			C113	C113
113					C114			C114	C114
114					C115			C115	C115
115					C116			C116	C116
116					C117			C117	C117
117					C118			C118	C118
118					C119			C119	C119
119					C120			C120	C120
120					C121			C121	C121
121					C122			C122	C122
122					C123			C123	C123

123					C124			C124	C124
124					C125			C125	C125
125					C126			C126	C126
126					C127			C127	C127
127					C128			C128	C128
128					PR			C129	C129
129					OSAT			C130	C130
130					C131			C131	C131
131					C132			C132	C132
132					C133			C133	C133
133					C134			C134	C134
134					C135			C135	C135
135					C136			C136	C136
136					C137			C137	C137
137					C138			C138	C138
138					C139			C139	C139
139					C140			C140	C140
140					C141			C141	C141
141					C142			C412	C412
142					C143			C143	C143
143					C144			C144	C144
144					C145			C145	C145
145					C146			C146	C146
146					C147			C147	C147
147					C148			C148	C148
148					C149			C149	C149
149					C150			C150	C150
150					C151			C151	C151
151					C152			C152	C152
152					C153			C153	C153
153					C154			C154	C154
154					C155			C155	C155
155					C156			C156	C156
156					C157			C157	C157
157					C158			C158	C158
158					C159			C159	C159
159					C160			C160	C160
160					C161			C161	C161
161					C162			C162	C162
162					C163			C163	C163
163					C164			C164	C164
164					C165			C165	C165
165					C166			C166	C166
166					C167			C167	C167
167					C168			C168	C168

168					C169			C169	C169
169					C170			C170	C170
170					C171			C171	C171
171					C172			C172	C172
172					C173			C173	C173
173					C174			C174	C174
174					C175			C175	C175
175					C176			C176	C176
176					C177			C177	C177
177					C178			C178	C178
178					C179			C179	C179
179					C180			C180	C180
180					C181			C181	C181
181					C182			C182	C182
182					C183			C183	C183
183					C184			C184	C184
184					C185			C185	C185
185					C186			C186	C186
186					C187			C187	C187
187					C188			C188	C188
188					C189			C189	C189
189					C190			C190	C190
190					C191			C191	C191
191					C192			C192	C192
192					C193			C193	C193
193					C194			C194	C194
194					C195			C195	C195
195					C196			C196	C196
196					C197			C197	C197
197					C198			C198	C198
198					C199			C199	C199
199					C200			C200	C200
200					C201			C201	C201
201					C202			C202	C202
202					C203			C203	C203
203					C204			C204	C204
204					C205			C205	C205
205					C206			C206	C206
206					C207			C207	C207
207					C208			C208	C208
208					C209			C209	C209
209					C210			C210	C210
210					C211			C211	C211
211					C212			C212	C212
212					C213			C213	C213

213					C214			C214	C214
214					C215			C215	C215
215					C216			C216	C216
216					C217			C217	C217
217					C218			C218	C218
218					C219			C219	C219
219					C220			C220	C220
220					C221			C221	C221
221					C222			C222	C222
222					C223			C223	C223
223					C224			C224	C224
224					C225			C225	C225
225					C226			C226	C226
226					C227			C227	C227
227					C228			C228	C228
228					C229			C229	C229
229					C230			C230	C230
230					C231			C231	C231
231					C232			C232	C232
232					C233			C233	C233
233					C234			C234	C234
234					C235			C235	C235
235					C236			C236	C236
236					C237			C237	C237
237					C238			C238	C238
238					C239			C239	C239
239					C240			C240	C240
240					C241			C241	C241
241					C242			C242	C242
242					C243			C243	C243
243					C244			C244	C244
244					C245			C245	C245
245					C246			C246	C246
246					C247			C247	C247
247					C248			C248	C248
248					C249			C249	C249
249					C250			C250	C250
250					C251			C251	C251
251					C252			C252	C252
252					C253			C253	C253
253					C254			C254	C254
254					C255			C255	C255
255					C256			C256	C256
256								DC1	C257
257								DC2	C258

258							DC3	C259
259							DC4	C260
260							DC5	C261
261							DC6	C262
262							DC7	C263
263							DC8	C264
264							DC9	C265
265							DC10	C266
266							DC11	C267
267							DC12	C268
268							DC13	C269
269							DC14	C270
270							DC15	C271
271							DC16	C272
272							TRIG	C273
273							OSAT	C274
274							PR	C275
275							Pleth	C276
276								C277
277								C278
278								C279
279								C280
280								C281
281								C282
282								C283
283								C284
284								C285
285								C286
286								C287
287								C288
288								C289
289								C290
290								C291
291								C292
292								C293
293								C294
294								C295
295								C296
296								C297
297								C298
298								C299
299								C300
300								C301
301								C302
302								C303
303								C304
304								C305

305									C306
306									C307
307									C308
308									C309
309									C310
310									C311
311									C312
312									C313
313									C314
314									C315
315									C316
316									C317
317									C318
318									C319
319									C320
320									C321
321									C322
322									C323
323									C324
324									C325
325									C326
326									C327
327									C328
328									C329
329									C330
330									C331
331									C332
332									C333
333									C334
334									C335
335									C336
336									C337
337									C338
338									C339
339									C340
340									C341
341									C342
342									C343
343									C344
344									C345
345									C346
346									C347
347									C348
348									C349
349									C350
350									C351
351									C352

352									C353
353									C354
354									C355
355									C356
356									C357
357									C358
358									C359
359									C360
360									C361
361									C362
362									C363
363									C364
364									C365
365									C366
366									C367
367									C368
368									C369
369									C370
370									C371
371									C372
372									C373
373									C374
374									C375
375									C376
376									C377
377									C378
378									C379
379									C380
380									C381
381									C382
382									C383
383									C384
384									C385
385									C386
386									C387
387									C388
388									C389
389									C390
390									C391
391									C392
392									C393
393									C394
394									C395
395									C396
396									C397
397									C398
398									C399

399									C400
400									C401
401									C402
402									C403
403									C404
404									C405
405									C406
406									C407
407									C408
408									C409
409									C410
410									C411
411									C412
412									C413
413									C414
414									C415
415									C416
416									C417
417									C418
418									C419
419									C420
420									C421
421									C422
422									C423
423									C424
424									C425
425									C426
426									C427
427									C428
428									C429
429									C430
430									C431
431									C432
432									C433
433									C434
434									C435
435									C436
436									C437
437									C438
438									C439
439									C440
440									C441
441									C442
442									C443
443									C444
444									C445
445									C446

446									C447
447									C448
448									C449
449									C450
450									C451
451									C452
452									C453
453									C454
454									C455
455									C456
456									C457
457									C458
458									C459
459									C460
460									C461
461									C462
462									C463
463									C464
464									C465
465									C466
466									C467
467									C468
468									C469
469									C470
470									C471
471									C472
472									C473
473									C474
474									C475
475									C476
476									C477
477									C478
478									C479
479									C480
480									C481
481									C482
482									C483
483									C484
484									C485
485									C486
486									C487
487									C488
488									C489
489									C490
490									C491
491									C492
492									C493

493									C494
494									C495
495									C496
496									C497
497									C498
498									C499
499									C500
500									C501
501									C502
502									C503
503									C504
504									C505
505									C506
506									C507
507									C508
508									C509
509									C510
510									C511
511									C512
512									DC1
513									DC2
514									DC3
515									DC4
516									DC5
517									DC6
518									DC7
519									DC8
520									DC9
521									DC10
522									DC11
523									DC12
524									DC13
525									DC14
526									DC15
527									DC16
528									TRIG
529									OSAT
530									PR
531									Pleth

4. Note File – Schema 3

The note file contains notes entered during acquisition and review. Some notes are automatically generated by the NeuroWorks software, and some are entered directly by the user.

The note file begins with a generic EEG file header. After this is a series of variable length records. Each record begins with a fixed length header defined as follows:

Offset (bytes)	Size (bytes)	Name	Type	Description
0	4	Type	int	One of: END_OF_NOTES_TYPE = 0, KEYTREE_TYPE = 1
4	4	Length	long	length of the record including the header in bytes.
8	4	prev_length	long	length of the last record including the header.
12	4	Id	long	**This field is unused. The unique ID is now stored under the "GUID" key in the note key tree.**

The last record in the file consists only of a note header with type of 0 (END_OF_NOTES_TYPE). All other records have type 1 (KEYTREE_TYPE). For key tree records, the data which follows the header is in Excel list format. This is a string representation of a key tree with the following format:

4.1 K(Note)

Key	Type	Description
Type	S()	"Annotation"
GUID	S()	Unique note id number within the study.
Text	S()	Note title.
Comment	S()	Note comment.
Stamp	I()	Sample stamp of the note. This is a count of samples from the start of the study indicating the time of the note.
Origin	S()	"Review" or "Acquisition"
Host	S()	The name of the computer on which the note was generated.
ReadOnly	I()	TRUE or FALSE
Data	K(NoteData)	Additional note data.

4.2 K(NoteData)

4.2.1 Base Note Data

All notes have the following note data fields:

Key	Type	Description
Type	S()	"", "Montage", "Montage Delta", "Impedance", "Reference"
User	S()	User login name of the note creator.
CreationTime	R()	Microsoft DATE format.
ModificationTime	R()	Microsoft DATE format.
Deleted	I()	TRUE or FALSE. May not be present if note is not deleted.

Depending on the note data type additional fields are present, as outlined below.

4.2.2 Impedance Note Data

Key	Type	Description
ThresholdIndex	I()	25000 Kohm = 0, 10000 Kohm = 1, 5000 Kohm = 2, 2500 Kohn = 3
IsDoneScanning	I()	TRUE or FALSE
CurrentChannel	I()	
LockedChannel	I()	
Chan*x*	I()	Impedance value for channel number *x* in ohms.
SD*x*	R()	Standard deviation of the impedance.

4.2.3 Montage Note Data

Key	Type	Description
FileName	S()	Full path name of the montage file.
Schema	I()	2
Name	S()	Montage name.
MontgageType	S()	"10-20", "Grid"
HalfMontageChannels	L(K(HalfMontageChannel))	
Channels	L(K(ChanNode))	
ChanNames	L(S())	List of channel names.
SpecResolution	I()	Spectrum resolution.
SpecDecimation	I()	Spectrum decimation.
SpecNumBins	I()	Number of spectral bins.
Bin0 … Bin5	K(SpecBinData)	

4.2.3.1 K(SpecBinData)

Key	Type	Description
LowFreq	R()	Lower bin frequency [Hz].
HighFreq	R()	Higher bin frequency [Hz].
Color	I()	Bin color. (COLORREF)

4.2.3.2 K(ChanNode)

Key	Type	Description
ChanIndex	I()	Output channel index.
Channel	K(MontageChannel)	

4.2.3.3 K(MontageChannel)

Key	Type	Description
ChanType	S()	"Waveform", "Amplitude Spectrum [µV]", "Power Spectrum [uV²]" "Power Spectrum [dB]"
Group	I()	

Page	I()	
Color	I()	COLORREF
Gain	S()	Gain string (number followed by a unit)
LffCutoff	R()	[Hz]
HffCutoff	R()	[Hz]
NotchCutoff	R()	[Hz]
From_Name	S()	Input2 half montage channel name.
To_Name	S()	Input1 half montage channel name
ChanProcType	S()	Used to indicate if event detection is active for this channel. Possible values: "Enabled", (used for detection) "Disabled", (not used) "Artifact" (used to detect eye blinks, etc.)
PositiveUp	I()	0 / 1
GroupId	G()	
Set	I()	
IchannelId	G()	
ItypeId	G()	
IsiteId	G()	
IinputId	G()	
SamplingFreq	R()	
Calibration	K()	
OchannelId	G()	
OtypeId	G()	

4.2.3.4 K(HalfMontageChannel)

Key	Type	Description
Name	S()	
Readonly	I()	TRUE or FALSE. HMC's that correspond to physical channels are read only. The Laplacian HMC's are writeable.
Coefficients	L(K(Coefficients))	

4.2.3.5 K(Coefficients)

Key	Type	Description
Chan	I()	Input channel Id.
Coeff	R()	Channel weight.

4.2.4 Montage Delta Note Data

Key	Type	Description
0 … N	K(ChannelDelta)	Change in channel N.

4.2.4.1 K(ChannelData)

Key	Type	Description
Gain	R()	Gain in uV/mm. Only present if changed.
LffCutoff	R()	[Hz] . Only present if changed.
HffCutoff	R()	[Hz] . Only present if changed.

NotchCutoff	R()	[Hz] . Only present if changed.

4.2.5 Event and Episode, or Spike and Seizure Note

Key	Type	Description
CreationTime	R()	Microsoft DATE format
UseCreator	I()	1, if created by a processor. 0 if manually creator
ChannelNumber	I()	The channel number of the event/episode if applicable.
StartTime	_int64 (FileTime)	The start time of the note saved in FileTime format.
EndTime	_int64 (FileTime)	The end time of the note, if applicable, saved in FileTime format.
EndStamp	I()	End stamp of the note, if applicable, -1 otherwise
AnalysisId	U()	GUID that identifies which batch analysis run the note belongs to. Only present for notes generated by batch analyzer.
AnalysisContext	I()	Indicates which application created the note. Set to one of: 1 – Batch Analyzer 2 – Ambulatory Upload 3 – Wave, Acquisition 4 – Wave, Review At the time of writing only the batch analyzer creates and sets this key. It is used by the batch analyzer to distinguish between notes created by it and the other applications.

5. Table Of Contents File

The Table Of Contents file is an index into the raw data file used for mapping samples to times, and also to facilitate quick searches. This file ends in the extension '.toc'. The file starts with a generic EEG file header, and is followed by a series of fixed length records called toc entries.

Each toc entry provides an offset into the raw data file to the start of a sample packet. The packet at that location is guaranteed to contain all abs_delta values in the delta information, followed by absolute sample values.

A toc entry is added to the toc file whenever recording is started and also for every 1000 samples.

Note: The byte alignment for the toc entry structure is set at 4 bytes.

5.1 TOC Entry – File Schema 2

Offset (bytes)	Size (bytes)	Name	Type	Description
0	4	offset	DWORD	Offset in bytes into the '.erd' file to a sample packet.
4	4	timestamp	long	Timestamp in milliseconds from the start of the study. The timestamp keeps incrementing even when not recording.
8	4	sample_num	long	Number of samples recorded to this point.
12	4	sample_span	short	Number of samples from this toc entry to the next.

5.2 TOC Entry – File Schema 3

Offset (bytes)	Size (bytes)	Name	Type	Description
0	4	offset	DWORD	Offset in bytes into the '.erd' file to a sample packet.
4	4	samplestamp	long	Sample count from the start of the study. Once the study begins, the sample count keeps incrementing at the sampling frequency, even when not recording. The sampling frequency is constant throughout a study, so the sample stamp can be readily converted to time.
8	4	sample_num	long	Number of samples recorded to this point.
12	4	sample_span	short	Number of samples from this toc entry to the next.

6. Synchronization File (.SNC)

The synchronization file is used to calculate a FILETIME given a sample stamp (and vise-versa). Theoretically, it is possible to calculate a sample stamp's FILETIME given the FILETIME of sample stamp zero (when sampling started) and the sample rate. However, because the sample rate cannot be represented with full precision the accuracy of the FILETIME calculation is affected.

To compensate for the lack of accuracy, the synchronization file maintains a sample stamp-to-computer time (called, MasterTime) mapping. Interpolation is then used to calculate a FILETIME given a sample stamp (and vise-versa).

6.1 SNC file entry – Schema 0

The synchronization file begins with a generic file header followed by 1 or more synchronization entries:

Size (bytes)	Name	Type	Description
4	sampleStamp	long	Sample number from start of study.
8	sampleTime	FILETIME	File time representation of sampleStamp.

The attributes, sampleStamp and sampleTime, are used to predict (using interpolation) the FILETIME based upon a given sample stamp (and vise-versa). Currently, the only use for this conversion process is to enable correlation of EEG (sample_stamp) data with other sources of data such as Video (which works in FILETIME).

7. Segment Table of Contents File (.STC)

The Segment Table of Contents file is an index into pairs of (raw data file / table of contents file). It is used for mapping samples file segments. EEG raw data is split into segments in order to break a single file size limit (used to be 2GB) while still allowing quick searches. This file ends in the extension '**.stc**'. Default segment size (size of ERD file after which it is closed and new [ERD / ETC] pair is opened) is 50MB and may be tweaked by **[HKCU]\Software\ExcelTech\Storage\Settings\SegmentSize<Low | High>** registry settings. The file starts with a generic EEG file header, and is followed by a series of fixed length records called the STC entries. ERD segments are named according to the following schema:

<FIRST_NAME>, <LAST_NAME>_<GUID>.ERD (first)
<FIRST_NAME>, <LAST_NAME>_<GUID>.ETC (first)
<FIRST_NAME>, <LAST_NAME>_<GUID>_<INDEX>.ERD (second and subsequent files)
<FIRST_NAME>, <LAST_NAME>_<GUID>_<INDEX>.ETC (second and subsequent files)

<INDEX> is formatted with "%03d" format specifier and starts at 1 (initial value being 0 and omitted for compatibility with the previous versions).

Each STC entry provides an index of the ERD / ETC pair for specific range of samples.
An STC entry is added to the STC file whenever a new segment is created.

Note: The byte alignment for the STC entry structure is set at 4 bytes.

7.1 STC File Header- Schema 1

The stc data file begins with a generic file header followed by a stc data header defined as follows:

Offset (bytes)	Size (bytes)	Name	Type	Description
352	4	m_next_segment	int	Sample frequency in Hertz.
356	4	m_final	int	Number of channels stored.
360	48	m_padding	int[12]	Padding
402				

7.2 STC Entry – File Schema 1

STC entry is very similar to the ETC file entry so that current algorithms that rely on last ETC entry to get end of study stamp can still work based on information in STC file.

Offset (bytes)	Size (bytes)	Name	Type	Description
0	4	segment_name	char[256]	Name of ERD / ETC file segment.
256	4	start_stamp	long	First sample stamp that is found in the ERD / ETC pair.
260	4	end_stamp	long	Last sample stamp that is still found in the ERD / ETC pair.
264	4	sample_num	long	Number of samples recorded to the point that corresponds to **start_stamp**. This number accumulates over ERD / ETC pairs and is equal to **sample_num** of the <u>first</u> entry in the ETC file referenced by this STC entry.
12	4	sample_span	long	Number of samples from this stc entry to the next.

8. Sleep Data File (.EPO)

The Sleep Data file stores all sleep specific data. It stores individual data items related to a study (such as epoch length, all fields will be discussed further below), as well as channels storing data for each epoch. The file contains a header, which has all of the individual data items, and a group of individual data channels, each containing as many data items as the number of epochs in the whole study (calculated from start of recording to end of recording, independent of the lights off time). The sleep data file uses structured storage to store its data, which is similar to a file system, with "Storages" representing directories, and "streams" representing files. It uses the following format:

<root> (storage)
        <GlobalHeader> (storage)
                <Schema> (stream)
                <EpochInfo> (stream)
                <StudyInfo> (stream)
        <Epoch Data Channel> (storage)
                <ChannelInfo> (stream)
                <ChannelRawData> (stream)
        <Epoch Data Channel> (storage)
        <Epoch Data Channel> (storage)
        …

8.1 EPO File Header (Schema 3.4.0.0)

The epo file header stores individual data items, separated into 3 separate streams: Schema (which stores the file schema), StudyInfo (which stores the patient and study guid associated with this file), and epoch info (which stores all other sleep related individual data items). These data items are always written/read from the file in the exact same order, as seen in the EpochDataBridge project.

8.1.1 EPO File Header Schema

Name	Type	Size (bytes)	Description
Major	Int	4	Major version number
Minor	Int	4	Minor version number
Revision	Int	4	Revision number
Build	int	4	Build number

8.1.2 EPO File Header StudyInfo

Name	Type	Size (bytes)	Description
StudyGuid	GUID	9	Study guid associated with this file
PatientGuid	GUID	9	Patient guid associated with this file

8.1.3 EPO File Header EpochInfo

Name	Type	Size (bytes)	Description
EpochLength	Long	4	Epoch length for the study (in seconds)
TotalEpochs	Long	4	Number of epochs in the study (from lights off to end of study)

CurrentEpoch	Long	4	The current epoch being looked at while reviewing the study
StartTime	XLTIME_T	4	The sample stamp corresponding to the start of epoch #1 in the study
SleepTimeBase	Long	4	The timebase used for sleep studies (obsolete)
Valid	BYTE	1	The valid flag to determine whether or not all data in the file is complete.
CreationTime	XLTIME_T	4	Unused
CurrentStageName	BSTR	Variable	The "friendly name" of the current staging set being written to/read from.
SegmentStartStamp	Long	4	Unused (originally intended to be used with merging and auto-recovery)
StudyStartStamp	Long	4	The sample stamp corresponding to the start of the study
EpochOffset	Long	4	The number of epochs in the study before lights off
RecoverEpochOffset	Long	4	Unused (another variable intended for use with merging and auto-recovery)
NextFileNum	Long	4	Next file number

8.2 EPO File Epoch Data Channel (Schema 3.4.0.0)

The storages corresponding to Epoch Data Channels have GUIDS for names to ensure that there are no attempts to create storages with the same name (since channel friendly names could theoretically be the same). Thus, each storage has 2 streams within it, one to store channel information and one to store the actual data for the channel.

8.2.1 EPO File Epoch Data Channel Basic ChannelInfo Stream

The following fields are found in the ChannelInfo stream for ALL epoch data channels

Name	Type	Size (bytes)	Description
FriendlyName	BSTR	Variable	The human readable channel name
ChannelType	enChannelType	4	The type of channel (Header, Staging Set, Marking Set, Event Summary, or Data Summary)
StartEpoch	Long	4	Unused
EndEpoch	Long	4	Unused
Hidden	VARIANT_BOOL	2	Determines whether or not the channel is hidden
Creator	BSTR	Variable	Unused (but would appear that it is supposed to be the name of the person who created the channel)
Password	BSTR	Variable	Password used for protecting channels from unwanted changes
Description	BSTR	Variable	Unused (intended as a text description of the channel)

8.2.2 EPO File Epoch Data Channel DataSummary ChannelInfo Stream

The following fields are found in the ChannelInfo stream for DataSummary channels, in addition to the basic fields found in section 8.2.1

Name	Type	Size (bytes)	Description
DataType	enDataType	4	The type of data found in this channel (EKG, O2SAT, HeartRate, Body Position or CPAP)
SummaryType	enSummaryType	4	The type of summary data (count, duration, average, min, max, position, Inspiratory CPAP, Expiratory CPAP, or Breath Rate)

8.2.3 EPO File Epoch Data Channel EventSummary ChannelInfo Stream

The following fields are found in the ChannelInfo stream for EventSummary channels, in addition to the basic fields found in section 8.2.1. This channel type is specifically used for storing sleep event information.

Name	Type	Size (bytes)	Description
Represents	enEventRepType	4	The type of event data that this represents (event count, duration, maximum amplitude, average amplitude, or minimum amplitude)
TypeID	BSTR	Variable	The type ID of the sleep event
EventTypeDesc	BSTR	Variable	Unused (intended as a text description of the event type)

8.2.4 EPO File Epoch Data Channel RawData Stream

The RawData stream in the Epoch Data Channel is a continuous byte stream (of size (TotalEpochs + EpochOffset) * sizeof(datatype) according to the header) which can be interpreted as a stream of varying types, depending on the type of channel. The following table explains how to interpret the data based on the channel type.

ChannelType	Type of Data in Stream	Size
Header	See header description	N/A
Staging Set	Byte	1
Marking Set	Byte	1
Data Summary	Double	8
Event Summary	Double	8

9. Video Table of Contents File (.VTC)

VTC files may have two possible extensions:

Ordinary VTC file - *.vtc
Extracted video files - *.new.vtc[2]

VTC file consists of a header and zero or more records. Structure of a VTC file can be expressed as:

VtcFile := VtcHeader { VtcRecord }

VtcHeader :=   GUID_VTC,   // 16 bytes
                 LONG_Schema   // 4 bytes

The header is packed on a 1-byte boundary (although it naturally falls on a 4-byte boundary as well).

Recent introduction of MPEG-4 video encoding support has lead to a new (MPEG-4) schema. The product supports both schemas (using the VTC header to distinguish between the two).

Video files are named sequentially using a _XXXX format (e.g., _0001, _0002, etc. or _1, _2, … _9999). Because the new MPEG-4 video files are shorter than the legacy MPEG-1 files, there are more of them per study (about 5 times as many).

9.1 Legacy (MPEG-1) VTC Schema

The legacy video schema incorporates a table-of-contents (VTC) file pointing to zero or more video files with the .MPG extension. These video files are MPEG-1 system stream containers. Each video file is typically 10 minutes in duration (but may be shorter if the study has been edited).

VtcRecord :=   STRING_MpgFileName, // 261 bytes (MAX_PATH + 1 for terminating NUL char)
                 INT_View,   // 4 bytes (not currently used)
                 FILETIME_StartTime,   // 8 bytes
                 FILETIME_EndTime   // 8 bytes

Note that the structure aligns to a 4 byte boundary so the string actually takes up 264 bytes. In addition the start/end time are forced to 4 byte boundaries using a pragma pack (4) compiler directive.

Where
    GUID_VTC = {5E01C1F1-F9A9-11D2-ADF3-00104B8CC65E}
    LONG_Schema = 1
    STRING_MpgFileName is a 261 bytes character array
    INT_View = 0
    FILETIME_StartTime and FILETIME_EndTime are time of the first and the last video frame stored in
        MPG file referred by the VtcRecord.

---

[2] It is an additional *.vtc file that has links to an optional set of mpg files. This set is created when user uses "Extract Video" in pruning clips dialog. It is represented by a red line beneath a blue one that represents the original video.

9.2 New (MPEG-4) VTC Schema

The new (MPEG-4) video schema differs from the legacy schema as follows:

1. 'View' is replaced by a 'Location' GUID
2. The video files themselves are AVI containers in which DivX© encoded video is multiplexed with a PCM 8-bit mono audio stream (the audio stream is the master)
3. The AVI files are shorter (typically 2 minutes per file)
4. Header GUID and schema have been changed (see below)
5. The VtcRecord is aligned to a single byte boundary (rather than the 4-byte alignment of the legacy schema)

```
VtcRecord :=      STRING_MpgFileName,  // 261 bytes ( MAX_PATH + 1 for terminating NUL char)
                  GUID_Location,        // 16 bytes
                  FILETIME_StartTime,   // 8 bytes
                  FILETIME_EndTime      // 8 bytes
```

Where
  GUID_VTC = { 0x5be3aca1, 0xa230, 0x4ac0, { 0xaa, 0x94, 0x49, 0x65, 0xf, 0x2c, 0xf6, 0xa0 } }
  LONG_Schema = 0xXXX10001 (where most significant 3 nibbles are reserved)
  STRING_MpgFileName is a 261 bytes character array (no path, assume file is local to VTC file)
  GUID_Location = {0x5ef8feff, 0xdcfc, 0x44e5,{0x8f, 0xae, 0x19, 0xf5, 0xd6, 0x22, 0xb6, 0xd4}}
  FILETIME_StartTime and FILETIME_EndTime are time of the first and the last video frame stored in AVI file referred by the VtcRecord.

Notes:

Dec 7, 2005   - Changes to schema interpretation for MPEG-4 VTC (as of EEGWorks v5.4)

- The most significant 3 nibbles (12 bits) of the schema are now reserved for internal use. As a result, one must mask the schema with 0x000fffff in order to read whether this is a new or legacy VTC file. The reserved bits are used internally by XLTEK video acquisition and rendering to track post processing alterations to the VTC content. For example, the first bit (of the 12) denotes whether or not the end time of each entry has been validated against the reported media length.

- This change only applies to the new MPEG-4 VTC schema.

Revision History

Rev	Date		Description
L	Oct. 6, 2004	SA	Documented new MPEG-4 VTC schema
M	Dec. 7, 2005	SA	Documented change to MPEG-4 VTC schema for EEGWorks v5.4. Added this revision history table and added a table of contents to this document.
N	Jan. 26, 2006	SA	Corrected conversion factors for Trex referential and differential channels (0-27). What was in document did not match what was used in code (EEGWORKS\SOURCE\EEGCOMMON\AmbulatoryUSBHeadbox..h) For EMU40, added conversion factors and channel mapping.
O	Jun. 30, 2009	EF	Added EMU40, Netlink, NeuroLink IP, EEG32U and Traveler headboxes
P	Aug. 31, 2009	EF	Added NeuroLink IP 256 headboxes
Q	Aug. 31, 2015	AZ	Added Quantum I and Quantum II