Proposed C3D file EVENT storage enhancement

Date of this DRAFT: 4/6/00

Authors: Steven J. Stanhope, Ph.D.
e-mail: Steven_Stanhope@nih.gov

Some material adapted from the AMASS reference manual
and materials by Ms. Pamela Smeder

Overview:

This document outlines an enhanced C3D file format that will support an increased number of EVENT units and the inclusion of a proposed new RANGE variable. EVENT units are used as a general way of designating significant instances in a C3D file (e.g., initiation and/or termination of foot-floor contact). Current EVENT units contain a one to four character EVENT label (e.g. RHS, RTO), and an EVENT time in seconds relative to the first sample (designated as 0.0s) of the C3D file. The proposed new EVENT and RANGE (NER) section will be stored in additional records added to the C3D file. The proposed enhancement is backwardly compatible with known existing computer programs.

Original Format:

The original EVENT storage format accommodates 18 EVENT units. The original format is stored in the C3D file header record which is the first record of the C3D file (see Table 1).

Table 1. Summary of original C3D EVENT format (from AMASS Reference Manual, section 2-2)

Word*	Contents
148	Not used
149	Not used
150	12345 -- keyword to identify presence of event data
151	Number of defined time events- maximum 18.
152	Not used
153->188	(REAL*4) EVENT times in seconds (maximum of 18)
189->198	(BYTE*1) EVENT display switches (0=0n, 1=Off)
199->234	(CHAR) EVENT labels. Each EVENT label has 4 bytes of available storage space

* In this case a Word contains two, eight-bit bytes. This is equivalent to an integer*2 in Fortran.

Opportunities for Improvement:

Changes to the original EVENT format would be beneficial for several reasons:

(1) Increased EVENT information storage capacity – An increasing number of applications require more than 18 EVENTs to analyze files.

(2) Longer EVENT labels could be supported – EVENT labels longer than 4 characters would be more descriptive.

(3) A new variable called RANGE could be created and stored with the C3D file information.

Terminology and Conventions:

C3D file – A binary data file, consisting of multiple complete 512 byte (8 bits/byte) records, that conforms to the original AMASS convention.

C3D file header record – The very first record of 512 eight-bit bytes in a C3D file.

EVENT section – A region of the C3D file that contains EVENT statistics (e.g., the number of EVENT units) and one or more actual EVENT units.

EVENT unit – The total information used to completely characterize one EVENT. (e.g., EVENT label, EVENT time, display switch mode)

EVENT label – A character string (name) used to associate a generalized occurrence (e.g., heel strike, mid stance, toe off) with a discrete instant in C3D file time.

EVENT time – An instance in C3D file time in seconds (where C3D file sample one corresponds to time 0.0s) that is associated with an EVENT label

EVENT display switch – A binary switch (0=on, 1=off) that controls the display of EVENT labels on graphs. Setting an EVENT display switch to 1 should prevent the EVENT label from being displayed during graphics routines.

EVENT instance – The zero based numerical order in which EVENT units having the same EVENT label occur in C3D file time (i.e., the first occurrence of each unique EVENT unit (as determined from the EVENT label) is instance 0).

RANGE unit – The total information used to completely characterize one C3D file RANGE (e.g., RANGE name, start EVENT label flag, start EVENT label…)

RANGE name – A character string (name) used to identify a specific RANGE unit.

RANGE start EVENT label flag – Has a value of 0 or 1. Setting the RANGE start EVENT label flag to 0 specifies that the RANGE start time will be determined using the EVENT label stored in the RANGE start EVENT label location. Therefore, the start time will come from the EVENT unit and not the RANGE start time variable in the RANGE unit. In addition, the RANGE start EVENT instance must be set to the correct value. (note: EVENT instance is zero based, if only one EVENT unit exists with the specified label the instance value is zero. All other instances are determined from the time order of EVENT units having identical labels). Setting the RANGE start EVENT label flag value to 1 indicates that the RANGE start time is stored in the RANGE unit in the RANGE start time variable. In this case, the RANGE start EVENT label and associated RANGE start EVENT instance variables are ignored.

RANGE start EVENT label – A valid EVENT label.

RANGE start EVENT instance – The EVENT instance of the EVENT unit to be used in determination of the RANGE start time.

RANGE start time – Start of the RANGE in seconds (where C3D file sample one corresponds to time 0.0s).

RANGE stop EVENT label flag – Has a value of 0 or 1. Setting the RANGE stop EVENT label flag to 0 specifies that the RANGE stop time will be determined using the EVENT label stored in the RANGE stop EVENT label location. Therefore, the stop time will come from the EVENT unit and not the RANGE stop time variable in the RANGE unit. In addition, the RANGE stop EVENT instance must be set to the correct value. Setting the RANGE stop EVENT label flag value to 1 indicates that the RANGE stop time is stored in the RANGE unit in the RANGE stop time variable. In this case, the

RANGE stop EVENT label and associated RANGE stop EVENT instance variables are ignored.

RANGE stop EVENT label – A valid EVENT label.

RANGE stop EVENT instance – The EVENT instance of the EVENT unit to be used in determination of the end RANGE time.

RANGE stop time – Time in seconds that designates the end of the RANGE (where C3D file sample one corresponds to time 0.0s).

Proposed New Format:

Additions to the C3D File Header Section

Existence of the NER section in a C3D file will be designated using Word 148 of the C3D file header record. Specifically, Word 148 must contain the keyword '12345'. The location of the first NER record will be stored in Word 149 of the C3D file header record (see Table 2).

Table 2. Proposed additions (bolded entries) to the C3D file header record.

Word (BYTE*2)	Contents
148	**12345 – keyword to identify new label and range section**
149	**Record number of start of new label / range section**
150	12345 -- keyword to identify presence of event data
151 (N)	Number of defined time events- maximum 18.
152	Not used
153->188	(REAL*4) Event times in seconds (maximum of 18)
189->198	(BYTE*1) Event display switches (0=0n,1=Off)
199->234	(CHAR) Event time names. Each time event name is composed of 4 characters.

Format of New EVENT and RANGE section

The NER section will consist of one or more C3D file records of data. No other data other than NER information can be contained in the NER section. The initial record of the NER will contain a header block (bytes 1 through 11). The NER header block will contain important statistics describing the locations and dimensions of NER data.

EVENT and RANGE header section

Table 3. Format of the new EVENT and RANGE header section:

BYTE location*	Contents
1	90 decimal **(updated format)**
2,3	Total number of records in the NER section **(updated format)**
4,5 (M)	Number of EVENT units contained in the NER section This variable is used to determine the size of variable "M" in equation 2 below
6,7 (L)	Number of bytes available to store each EVENT label This variable would be typically set to the longest EVENT label. It is also used to determine the size of variable "L" in Table 4 and equation 1 below.
8,9 (R)	Number of RANGE units contained in the NER section
10,11 (P)	Number of bytes available to store each RANGE label This variable is also used to determine variable "P" in Table 5 below
12 to the end of EVENT units	EVENT units followed by RANGE information are stored from byte 12 on. See format presented below in tables 4 and 5. Note: EVENT and RANGE unit information (including two byte words) may be split across C3D records.

* Byte locations are relative to first byte in the first record of the NER section

EVENT unit format

The first EVENT unit begins at byte 12 in the first record of the NER section. Subsequent EVENT units are stored sequentially from lowest to highest based on the EVENT times. The order of the EVENT units is used to determine the EVENT instance when more that one EVENT having the same label are stored in the C3D file. It is important to note that the EVENT instance numbering convention is zero based (the first EVENT is instance 0).

Table 4. Composition of an EVENT unit

BYTE location	**Contents**
12 to L+11	(BYTE*L) EVENT label. Where each has 'L' bytes of available storage space
L+12 to L+15	(REAL*4) EVENT time in seconds
L+16	(BYTE) Event display switch (0=0n,1=Off)
L+17 to L+20	(BYTE*4) Intentionally not used

Total length in bytes of one EVENT unit is

L + 4 +1+4 = L+9 equation 1

Where L is the amount of storage space (in bytes) available to each EVENT label. The total length of all EVENT units in bytes (E) is determined from equation 2.

E = (L+9)*M equation 2

Where M is the number of EVENT units (see Table 3 above).

RANGE unit format

The first RANGE unit starts immediately following the unused byte of the last EVENT unit. This is equivalent to byte E+1 from the start of the EVENT units and 11 + E +1 bytes from the beginning of the NER header block.

Table 5. Composition of a RANGE unit

BYTE location	**Contents**
1 to P	(BYTE*P) RANGE label. Each label has 'P' bytes of available storage space
P+1	(BYTE*1) RANGE start EVENT label flag. Set to (0) if the RANGE start EVENT label is used to determine the start time. Set to (1) if the RANGE start time in seconds is used
P+2 to P+L+1	(BYTE*L) RANGE start EVENT label note: The EVENT label has L bytes of available storage space.
P+L+2 to P+L+3	(BYTE*2) RANGE Start EVENT instance note: The EVENT instance is based on the order of occurrence of EVENT units. Instance is zero based.
P+L+4 to P+L+7	(BYTE*4) RANGE start time - The start time in seconds where C3D sample one is equal to 0.0 seconds
P+L+8	(BYTE*1)RANGE stop EVENT label flag Set to (0) if the RANGE stop EVENT label is used to determine the end time. Set to (1) if the RANGE stop time in seconds is used
P+L+9 to P+2L+8	(BYTE*L) RANGE stop EVENT label note: The EVENT label has L bytes of available storage space.
P+2L+9 to P+2L+10	(BYTE*2) RANGE stop EVENT instance note: The EVENT instance is based on the order of occurrence of EVENT units. Instance is zero based.
P+2L+11 to P+2L+14	(BYTE*4) RANGE stop time – The stop time in seconds where C3D sample one is equal to 0.0 seconds

Total length of one RANGE unit in bytes = P+2L+14
Total length of all RANGE units in bytes = (P+2L+14)R

Total length of data in the NER section (BYTES) = 11 + (L+9)M + (P+2L+14)R **
** Round this figure up to the closest 512-byte record for total required storage.

Implementation Notes:

New EVENT units are stored sequentially in time order starting immediately after the NER header section (first eleven bytes of the NER section). RANGE units are stored sequentially, in RANGE start time order, immediately following the RANGE units. The location of the first record of the NER section is specified in word 149 of the C3D file header section. If the required storage space of the NER section exceeds one record, additional records are added immediately following the first. The NER section is best located after the C3D file parameter section (see Figure 1). In this case, the starting record is limited by a two-byte word. If all fields of an EVENT or RANGE unit do not fit into the same record, the EVENT or RANGE unit will be split between two 512-byte records (note: this means 16 and 32 bit words may be split across records). Additional records are added as necessary to store all the data. Unused storage space in the last NER record is padded with zeros.

Like EVENT units, implementation of RANGE units is intended to be literal in nature. This means, one RANGE unit must exist for every desired range that is based on EVENT units. The NER format is not intended to support implied RANGEs. For example, a C3D file may contain 10 EVENT units all having the EVENT label “right heel strike”. From these ten EVENT units a person can create nine RANGE units, each having the same label “right stride” and each being defined to start and stop on subsequent instances of EVENT units labeled “right heel strike”. The valid format of the NER is for the nine RANGE units to literally exist as separate definitions. Creation of only one RANGE unit with the label “right stride” and *implying* that application programs should evaluate the EVENT units to determine the number of instances this RANGE unit might exist is NOT a valid use of the NER format.

Figure 1. Sample C3D file layout with “suggested” location of NER section

H_1	P_1	P_2	…	P_N	NER_1	NER_2	...	NER_N	D_1	D_2	…	D_N

(H= C3D header record, P= Parameter section records, R = New EVENT and RANGE section records, D= Data section records)

Other aspects of the C3D file must be updated if their starting block location is changed by the inclusion of NER section. These are:

(1) The location of the first parameter record field (BYTE 1 of the C3D file header section) will require updating if the NER section is inserted prior to the Parameter section. The first byte of the C3D file parameter section must be updated (like BYTE 1 of the C3D header section) to indicate the location of the first parameter record.

(2) The start of the Data section in records (WORD 9 in the C3D file header section) will require updating if the NER section is inserted before the Data section. In addition, the like variable in the C3D file Parameter section that indicates the data start record must also be updated (Group: "POINT"; Parameter: "DATA_ST")

Compatibility with other programs:

The new event and range data format is completely compatible with all previous versions of the NIH software. However, compatibility with other existing programs has not been established. No changes were made to the original EVENT storage format and both EVENT storage formats may coexist in a C3D file. The NER section should be ignored by existing versions of software. However, Software that is used to make C3D files from existing C3D file data (such as making .SMT files) may not copy the NER data to the generated files.