This document is the proposed revision to the 1975 SEG Y tape format.

The name of the proposed format is SEG Y rev1.

SEG Y rev1 Data Exchange format[1]

SEG Technical Standards Committee[2]

Draft 6

[2] Editors Michael W. Norris and Alan K. Faichney

Content

FIGURES

TABLES

1. Introduction

Since the original SEG Y Data Exchange Format (revision 0, reference page **Error! Bookmark not defined.**) was published in 1975 it has achieved widespread usage within the geophysical industry. This widespread usage has brought about many proprietary variations. Since the publication of SEG Y rev 0, the nature of seismic data acquisition, processing and seismic hardware has changed significantly. The introduction of 3-D acquisition techniques and high speed, high capacity recording media dictate the need for revisions to the SEG Y rev 0 format. The major changes introduced by SEG Y rev 1 are: standardizing the location of header information needed for current processing practices and defining a SEG Y data set as a byte stream format. The SEG Technical Standards Committee strongly encourages producers and users of SEG Y data sets to move to the revised standard in an expeditious fashion.

2. Summary

2.1. Unchanged Items

- EBCDIC encoding is allowed for text.
- The size of the original 3200-byte Textual File Header, 400-byte Binary File Header and 240-byte Trace Headers.
- The data locations for the initial 3200-byte Textual File Header

2.2. Changes from rev 0 to rev 1

- A SEG Y file may be written to any medium that is resolvable to a stream of variable length records.
- The data word formats are expanded to include four-byte, IEEE floating-point and one-byte integer data words.
- A small number of additional fields in the 400-byte Binary File Header and the 240-byte Trace Header are defined and the use of some existing entries is clarified.
- An Extended Textual File Header consisting of additional 3200-byte Textual File Headers blocks is introduced.
- The data in the Extended Textual File Header uses a stanza layout and standard stanzas are defined.
- Trace identification is expanded.
- Engineering conversions are introduced.
- The Textual Header and the Extended Textual Header can be encoded as EBCDIC or ASCII characters.

2.3. Notation

The term CDP as used in this document is used as a synonym for the term CMP.

2.4. Controlling Organization

The SEG Y rev1 is administered by the SEG Technical Standards Committee. Any questions, corrections or problems encountered in the format should be addressed to:

Society of Exploration Geophysicist
P.O. Box 702740
Tulsa, Ok 74170-2740

Attention: SEG Technical Standards Committee

Phone: (918) 497-5500
Fax: (918) 497-5557
Internet site: www.seg.org

2.5. Acknowledgments

The SEG Technical Standards Committee would like to acknowledge the time and effort put forth by a great many individuals and organizations. Special commendation goes to the P.E.S.G.B. Data Management Group. The chairman of P.E.S.G.B. Data Management Group was Jill Lewis. The

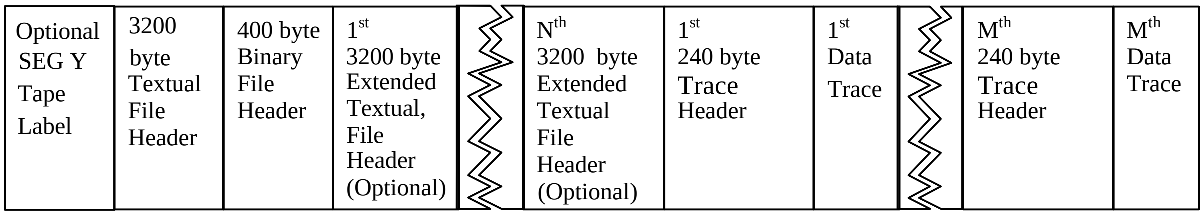
Optional SEG Y Tape Label
3200 byte Textual File Header
400 byte Binary File Header
1st 3200 byte Extended Textual, File Header (Optional)
Nth 3200 byte Extended Textual File Header (Optional)
1st 240 byte Trace Header
1st Data Trace
Mth 240 byte Trace Header
Mth Data Trace

Figure 1 Byte stream structure of a SEGY file with N Extended Textual File Header records and M traces records

principal members were Bob Firth, Eleanor Jack and Jill Holliday. Roger Lott produced the Extended Textual stanzas for Location Data, Bin Grid Definition and Data Extent. Thanks go to Frank Brassil for organizing the response from the Australian Geological Society.

3. SEG Y File Structure

The original SEG Y standard was produced at a time when 9-track tape was the normal storage medium for seismic data, with 800 bpi and 1600 bpi in common usage and 6250 bpi on the horizon. The revised format is intended to be independent of the actual medium on which it is recorded.

The stipulation in the 1975 standard that “no more than one line of seismic data is permitted on any one reel” became impractical long ago. The term “SEG Y file” shall be used in this revision in place of the term “seismic reel” used in the original standard. For this standard, the terms file and data set are synonymous. Both terms are a collection of logically related data traces or ensembles of traces and the associated ancillary data.

3.1. Recording Medium

The SEG Y format described in the 1975 standard defined a data format that was dependent on 9-track tape. With the revised format, a SEG Y file may be written to any medium that supports variable length records. Whatever medium is used, the data must be resolvable to a stream of variable length records. This includes tape devices, such as 9-track tape and 3480 cartridges, which can achieve this in hardware. It also includes high capacity tape devices such as DD2 or 3590, although with these it is desirable to use some kind of blocking and/or logical encapsulation, to use the tape more efficiently and possibly to allow the recording of associated metadata.

A SEG Y file may be written as a logical file to a SEG RODE encapsulated tape. Obviously when seismic data is being exchanged in SEG Y format, the medium and encapsulation scheme used must be acceptable to both the provider and recipient of the data.

One important class of media that does not conform to the variable length record model is the disk file, which is defined on modern systems as a byte stream without any structure. It has become common practice to write SEG Y data to disk, including CD-ROM, for data distribution. Certain rules have to be followed for this to work correctly. Appendix A defines how SEG Y data should be written to a disk file.

In order to make SEG Y consistent with the SEG D Rev 2 standard, Appendix B defines a tape label for SEG Y tapes, using a format based on the RP66 Storage Unit Label. Labels are not mandatory for SEG Y; but their use is highly desirable in environments such as robotic tape libraries and large scale processing centers.

Appendix C defines a simple blocking scheme for SEG Y data to allow more efficient use of high-capacity tape media.

This is based on the scheme defined in the SEG D Rev 2 standard.

3.2. File Structure

Figure 1 illustrates the structure of a SEG Y file. The first 3600-bytes of the file are the Textual File Header and the Binary File Header written as a concatenation of a 3200-byte record and a 400-byte record. This is optionally followed by Extended Textual File Header(s), which consists of zero or more 3200-byte Extended Textual File Header records. The remainder of the SEG Y file contains a variable number of Data Trace records that are each proceeded by a 240-byte Trace Header. The Extended Textual File Header is the only structural change introduced in this revision and while not strictly downward compatible with the 1975 SEG Y format, it has been carefully designed to have minimal impact on existing SEG Y reader software. It should be simple for existing software to be modified to detect the presence of the new header and either process or ignore the Extended Textual File Header. The format of the Extended Textual File Header is described fully in section 6.

3.3. Number Formats

In the 1975 SEG Y standard, all binary values are defined as using “big-endian” byte ordering. This conformed to the IBM tape standard and means that, within the bytes that make up a number, the most significant byte (containing the sign bit) is written closest to the beginning of the file and the least significant byte is written closest to the end of the file. This byte ordering convention is maintained in this revision of the SEG Y format and it should be adhered to for all conforming versions of SEG Y. This is independent of the medium to which a particular SEG Y file is written (i.e. the byte ordering is no different if the file is written to tape on a mainframe or to disk on a PC).

All values in the binary file header and the trace header are two's complement integers, either two bytes or four bytes long. There are no floating-point values defined in the headers.

Trace Data sample values are either two's complement integers or floating-point. This revision adds data sample formats of 8-bit integer and 32-bit IEEE floating-point. Both IBM floating-point (as defined in the original standard) and IEEE floating-point values are written in big-endian byte order (i.e. with the sign/exponent byte written closest to the beginning of the file).

3.4. Varying Trace Lengths

The SEG Y standard specifies fields for sampling interval and number of samples at two separate locations in the file. The Binary File Header contains values that apply to the whole file and the trace headers contain values that apply to the associated trace. The original standard is unclear about how these are to be used together. One view is that variable length traces are supported in SEG Y, with the number of samples in the trace header allowed to vary from trace to trace and to vary from the value in the Binary File Header. An alternate view is that all traces in a SEG Y file will be the same length and the value for the number of samples will be the same in the Binary File Header and all Trace Headers. The data traces are padded or truncated as necessary,

In SEGY rev 1, varying trace lengths in a file are explicitly allowed. The values for sampling interval and number of samples in the Binary File Header should be for the primary set of seismic data traces in the file. This approach allows the Binary File Header to be read and say, for instance, “this is six seconds data sampled at a two-millisecond sampling interval”. The value for the number of samples in each individual Trace Header may vary from the value in the Binary File Header and reflect the actual number of samples in a trace. The number of bytes in each trace record must be consistent with the number of samples in the Trace Header. This is particularly important for SEG Y data written to disk files (see Appendix A).

Allowing variable length traces dictates sequential access and precludes random access in a disk file, since the locations of traces after the first are not known. To enable the option of random access, a new field in the Binary File Header has been defined as a fixed length trace flag. If this flag is set, all traces in the file must have the same length. This will typically be the case for poststack data.

Making the value for number of samples in the Binary File Header the maximum trace length in the file, rather than the length of the primary set of data traces was seriously considered. However, it should be noted that the maximum trace length is typically not known at the time the Binary File Header is written, particularly in a transcription environment. This is the same reason why there are no fields in the Binary File Header such as “first and last record number”. The fixed length trace flag goes some way to ameliorating the problems induced by having variable length traces. If the fixed record length flag is set, the maximum trace length in the file is known (i.e. all traces are the same length).

4. Textual File Header

The first 3200-byte, Textual File Header record contains 40 lines of textual information, providing a human-readable description of the seismic data in the SEG Y file. This information is free form and is the least well defined of the headers in the 1975 standard, although the standard did provide a suggested layout for the first 20 lines. While there would be distinct advantages in making the layout of this header more rigid, it was decided that it would not be practicable to produce a layout that would be universally acceptable in the light of how it is currently used.

The SEG Y rev 1 standard defines a separate textual header with a more comprehensively defined structure, where textual information can be stored in a machine-readable way. This new header will be known as the Extended Textual Header and it is described in detail in section 6. Note that the “traditional” Textual Header is completely separate from the Extended Textual Header and will still be the primary location for human readable information about the contents of the file. In particular, it should contain information about any unusual features in the file, such as if the delay recording time in trace header bytes 109-110 is non-zero. The revision level of the SEG-Y format (Binary File Header bytes 3501-3502) being used **must** be included for all files written in the SEG Y rev 1 format. It is recommended that the SEG Y revision level be included in the Textual File Header. Table 1 is an example Textual File Header with the SEG Y revision level included in the 39th record.

Table 1 Textual File Header

3200-byte Textual File Header							
Cols 1-10 1234567890	**Cols 11-20** 1234567890	**Cols 21-30** 1234567890	**Cols 31-40** 1234567890	**Cols 41-50** 1234567890	**Cols 51-60** 1234567890	**Cols 61-70** 1234567890	**Cols 71-80** 1234567890
C 1 CLIENT			COMPAN	Y		CREW N	O
C 2 LINE		AREA		MA	P ID		
C 3 REEL N	O	DAY-STAR	T OF REEL	YEAR	OBSERV	ER	
C 4 INSTRU	MENT: MFG		MODEL	SE	RIAL NO		
C 5 DATA T	RACES/RECO	RD	AUXILIARY	TRACES/REC	ORD	CDF FOLD	
C 6 SAMPLE	INTERVAL	SA	MPLES/TRAC	E BI	TS/IN	BYTES/SAMP	PLE
C 7 RECORD	ING FORMAT	FO	RMAT THIS	REEL	MEASUREM	ENT SYSTEM	
C 8 SAMPLE	CODE: FLO	ATING PT	FIXED P	T FIXE	D PT-GAIN	CORREL	ATED
C 9 GAIN	TYPE: FIXE	D BINS	RY FLO	ATING POIN	T OTHE	R	
C10 FILTER	S: ALIAS	HZ NOT	CH HZ	BAND	- HZ	SLOPE -	DB/OCT
C11 SOURCE	: TYPE	NU	MBER/POINT	PO	INT INTERV	AL	
C12 PA	TTERN:			LENGTH	WID	TH	
C13 SWEEP:	START	HZ END	HZ LEN	GTH M	S CHANNEL	NO TY	PE
C14 TAPER:	START LEN	GTH	MS END LE	NGTH	MS TYPE		
C15 SPREAD	: OFFSET	MAX	DISTANCE	GROU	P INTERVAL		
C16 GEOPHO	NES: PER G	ROUP S	PACING	FREQUENCY	MFG	MO	DEL
C17 PA	TTERN:			LENGTH	WID	TH	
C18 TRACES	SORTED BY	: RECORD	CDP	OTHER			
C19 AMPLIT	UDE RECOV	RY: NONE	SPHERI	CAL DIV	AGC	OTHER	
C20 MAP PR	OJECTION			ZONE ID	COORDI	NATE UNITS	
C21 PROCES	SING:						
C21 PROCES	SING:						
C23							
C24							
C25							

3200-byte Textual File Header							
Cols 1-10	**Cols 11-20**	**Cols 21-30**	**Cols 31-40**	**Cols 41-50**	**Cols 51-60**	**Cols 61-70**	**Cols 71-80**
1234567890	1234567890	1234567890	1234567890	1234567890	1234567890	1234567890	1234567890
C26							
C27							
C28							
C29							
C30							
C31							
C32							
C33							
C34							
C35							
C36							
C37							
C38							
C39SEG Y rev1[5]							
C40 END EB	CDIC						

5. Binary File Header

The 400-byte Binary File Header record contains binary values that affect the whole SEG Y file. The values in the Binary file Header are defined as two-byte or four-byte, two's complement integers. Certain values in this header are crucial for the processing of the data in the file, particularly the sampling interval, trace length and format code. This revision defines a few additional fields in the optional portion, as well as providing some clarification on the use of some existing entries.

Table 2 Binary File Header

400-byte Binary File Header	
Byte	**Description**
3201-3204	Job identification number.
3205-3208	Line number. For 3-D poststack data, this will typically contain the in-line number.
3209-3212	Reel number.
3213-3214[3]	Number of data traces per ensemble. Mandatory for prestack data.
3215-3216[3]	Number of auxiliary traces per ensemble. Mandatory for prestack data.
3217-3218[4]	Sample interval in microseconds (µs). Mandatory for all data types.
3219-3220	Sample interval in microseconds (µs) of original field recording.
3221-3222[4]	Number of samples per data trace. Mandatory for all types of data. Note: The sample interval and number of samples in the Binary File Header should be for the primary set of seismic data traces in the file.
3223-3224	Number of samples per data trace for original field recording.
3225-3226[4]	Data sample format code. Mandatory for all data. 1 = 4-byte IBM floating-point 2 = 4-byte, two's complement integer 3 = 2-byte, two's complement integer 4 = 4-byte fixed-point with gain (obsolete) 5 = 4-byte IEEE floating-point 6 = Not currently used 7 = Not currently used 8 = 1-byte, two's complement integer
3227-3228[5]	Ensemble fold — The expected number of data traces per trace ensemble (e.g. the CMP fold). Highly recommended for all types of data.

[3] This information is mandatory for prestack data.

[4] This information is mandatory for all data types.

400-byte Binary File Header	
Byte	**Description**
3229-3230[5]	Trace sorting code (i.e. type of ensemble): -1 = Other (should be explained in user Extended Textual File Header stanza 0 = Unknown 1 = As recorded (no sorting) 2 = CDP ensemble 3 = Single fold continuous profile 4 = Horizontally stacked 5 = Common source 6 = Common receiver 7 = Common offset 8 = Common mid-point
3231-3232	Vertical sum code: 1 = no sum, 2 = two sum, ..., N = N sum (N = 1 to 32,767)
3233-3234	Sweep frequency at start.
3235-3236	Sweep frequency at end.
3237-3238	Sweep length (ms).
3239-3240	Sweep type code: 1 = linear 2 = parabolic 3 = exponential 4 = other
3241-3242	Trace number of sweep channel.
3243-3244	Sweep trace taper length in milliseconds at start if tapered (the taper starts at zero time and is effective for this length).
3245-3246	Sweep trace taper length in milliseconds at end (the ending taper starts at sweep length minus the taper length at end).
3247-3248	Taper type: 1 = linear 2 = cos^{2} 3 = other
3249-3250	Correlated data traces: 1 = no 2 = yes
3251-3252	Binary gain recovered: 1 = yes 2 = no

400-byte Binary File Header	
Byte	**Description**
3253-3254	Amplitude recovery method: 1 = none 2 = spherical divergence 3 = AGC 4 = other
3255-3256[5]	Measurement system: Highly recommended for all types of data. If Location Data stanzas are included in the file, this entry must agree with the Location Data stanza. If there is a disagreement, the Location Data stanza is the controlling authority. 1 = Meters 2 = Feet
3257-3258	Impulse signal polarity 1 = Increase in pressure or upward geophone case movement gives negative number on tape. 2 = Increase in pressure or upward geophone case movement gives positive number on tape.
3259-3260	Vibratory polarity code: Seismic signal lags pilot signal by: 1 = 337.5° to 22.5° 2 = 22.5° to 67.5° 3 = 67.5° to 112.5° 4 = 112.5° to 157.5° 5 = 157.5° to 202.5° 6 = 202.5° to 247.5° 7 = 247.5° to 292.5° 8 = 292.5° to 337.5°
3261-3500	Unassigned
3501-3502[4]	SEG Y Format Revision Number. This is a 16-bit unsigned value with a Q-point between the first and second bytes. Thus for SEG Y Revision 1.0, as defined in this document, this will be recorded as 0100_{16}. This field is mandatory for all versions of SEG Y, although a value of zero indicates "traditional" SEG Y conforming to the 1975 standard.
3503-3504[4]	Fixed length trace flag. A value of one indicates that all traces in this SEG Y file are guaranteed to have the same sample interval and number of samples, as specified in Textual File Header bytes 3217-3218 and 3221-3222. A value of zero indicates that the length of the traces in the file may vary and the number of samples in bytes 115-116 of the Trace Header must be examined to determine the actual length of each trace.
3505-3506[4]	Number of 3200-byte Textual Header Extension records following the Binary Header. A value of zero indicates there are no Textual Header Extension records (i.e. this file has no Extended Textual Header). A value of -1 indicates that there are a variable number of Textual Header Extension records and the end of the Extended Textual Header is denoted by an ((EndText)) stanza in the

400-byte Binary File Header	
Byte	**Description**
	final record. A positive value indicates that there are exactly that many Textual Header Extension records. Note that, although the exact number of Textual Header Extension records may be a useful piece of information, it will not always be known at the time the Binary Header is written and it would not be reasonable to insist that a positive value be recorded here.
3507-3600	Unassigned

6. Extended Textual File Header

If bytes 3505-3506 of the Binary File Header are non-zero, then an Extended Textual File Header is present in the SEG Y file. The Extended Textual File Header follows the Binary File Header record and precedes the first Data Trace record. An Extended Textual File Header consists of one or more 3200-byte records and provides additional space for recording required information about the SEG Y file in a flexible but well defined way. The kind of information recorded here will include navigation projections, 3-D bin grids, processing history and acquisition parameters. It is recommended that stanza information be included only once per SEG Y rev 1 file. In the event multiple or conflicting data entries are included in the SEG-Y rev 1 file, the last data entry is assumed to be correct.

The data in the Extended Textual File Header is textual card-image text, organized in the form of stanzas. Appendix D defines a set of predefined stanzas. It is intended that additional stanzas will be defined in the future revisions to this standard. However, the stanza mechanism is intended to be flexible and extensible and it is perfectly acceptable to define private stanzas to hold information for internal use. For the sake of usability and for maximum benefit, a standard stanza should be used if it exists for the type of information required. To avoid clashes of stanza names, some kind of naming convention should be adopted for private stanzas, e.g. prefixing the stanza name with the company name. A SEG Y reader must be capable of ignoring stanzas that it does not comprehend (which may be the whole Extended Textual File Header). The data within stanzas will typically use keywords and values, which can be produced and read by machine, as well as remaining human-readable.

Possible user supplied stanzas which have been suggested are:

General Data Parameters (e.g. License Block, Date, Operator, Line etc.)
General Acquisition Parameters
Geodetic Projection
Survey Outline
3-D Bin Grids
SP to CDP relationship
Usage of Optional parts of Trace Headers
Decoded Binary Header

It is strongly recommended that the SEG Y format be used principally to exchange seismic data. As part of that exchange, the SEG Y file should contain sufficient information to identify the data contained within the file and allow that data to be processed. *The SEG Y file is not intended as an ancillary data exchange format.* Extended Textual Headers provide a means to include almost unlimited ancillary data in the SEG Y file; but restraint should be exercised when selecting ancillary data to be included in the Extended Textual File Headers. If significant amounts of ancillary data need to be exchanged, it is recommended that SEG Ancillary Data Standard data set(s) be used.

6.1. Structure of Extended Textual Header

The Extended Textual File Header consists of one or more 3200-byte records, each record containing 40 lines of textual card-image text. Note that, unlike the Textual File Header, lines in the Extended Textual File Header do not start with the character “C” (EBCDIC C3$_{16}$ or ASCII 43$_{16}$). For processing purposes, all of the Extended Textual File Header records shall be considered as being concatenated into a single logical file (i.e. the gaps between the 3200-byte records are not significant). An exception is for the final 3200-byte record in the Extended Textual File Header, which shall contain a single empty stanza called ((EndText)) (see section 6.3).

Lines of text within the Extended Textual File Header are organized into stanzas. A stanza begins with a stanza header, which is a line containing the name of the stanza and ends with the start of a new stanza, or the end of the Extended Textual File Header. The stanza header begins with double left parentheses ("(", EBCDIC 4D$_{16}$ or ASCII 28$_{16}$) and ends with double right parentheses (")", EBCDIC 5D$_{16}$ or ASCII 29$_{16}$). The first left parentheses at the beginning of a stanza must be in column one. The case of stanza names shall not be significant. To aid readability, spaces (" ", EBCDIC 40$_{16}$ or ASCII 20$_{16}$) within stanza names shall be allowed but ignored. Thus the stanza name ((Recording Parameters)) shall refer to the same stanza as ((RECORDINGPARAMETERS)).

The format of the lines of text within a stanza depends on the type of the data contained in the stanza, which is implicitly defined by the name of the stanza. However, most stanzas will contain data organized as keyword/value pairs. The ground rules for stanzas that use this layout are as follows:

- Each line consists of a keyword/value pair in the form “keyword = value”.
- A keyword can contain any printable character except double right or left parentheses or the equal sign. However, the use of punctuation characters in keywords is not recommended.
- The case of a keyword is not significant.
- For readability, spaces within a keyword are allowed but ignored. Thus the keyword “Line Name” refers to the same value as “LINENAME”.
- The value associated with a keyword begins with the first non-blank character following the equal sign and extends to the last non-blank character on the line.
- The value field for a keyword may consist of multiple subfields, separated by commas (",", EBCDIC 6B$_{16}$ or ASCII 2C$_{16}$).
- Blank lines are ignored.
- If the first non-blank character in a line is the hash sign ("#", EBCDIC 7B$_{16}$ or ASCII 23$_{16}$), the line is treated as a comment and ignored.
- If the last non-blank character on a line is an ampersand ("&", EBCDIC 50$_{16}$ or ASCII 16$_{16}$), the next line is considered to be a continuation of this line (i.e. the next line is concatenated with this line, with the ampersand removed).
- Each line in an Extended Textual File Header ends in carriage return and linefeed (EBCDIC 0D25$_{16}$ or ASCII 0D0A$_{16}$)

6.2. Example of Stanza

((Data File Contents))

Line Name Convention = CDA

Line Name = As per convention above

First Trace In Data Set = numeric value

Last Trace In Data Set = numeric value

First Trace In Segment = numeric value

Last Trace In Segment = numeric value

First SP In Data Set = numeric value

Last SP In Data Set = numeric value

First SP In Segment = numeric value

Last SP In Segment = numeric value

First CDP In Data Set = numeric value

Last CDP In Data Set = numeric value

First CDP In Segment = numeric value

Last CDP In Segment = numeric value

6.3. EndText stanza

The stanza ((EndText)) is treated specially. This stanza must appear on its own in the final 3200-byte record in the Extended Textual File Header. The stanza header shall be on the first line in the record and must be the only non-blank text in the record (i.e. the stanza must be empty). This allows the end of the Extended Textual File Header to be located easily by SEG Y readers and simplifies decoding for SEG Y readers that do not wish to process the Extended Textual File Textual Header.

7. Data Traces

7.1. Trace Header

The SEG Y trace header contains trace attributes, which are defined as two-byte or four-byte, two's complement integers. The values in bytes 1-180 were defined in the 1975 standard and these entries remain unchanged, although clarification and extensions may be supplied where appropriate. Bytes 181-240 were for optional information in the 1975 standard and this has been the main area of conflict between different flavors of SEG Y. This revision defines standard locations in bytes 181-240 for certain values that are needed in modern data processing. In particular, standard locations for a shotpoint number and CDP coordinates are defined. Bytes 203 to 210 allow the measurement units for the trace data samples to be defined and transduction constants to be defined. These entries allow the trace data values to be converted to engineering units.

The values included in the Trace Header are limited and intended to provide information that may change on a trace-by-trace basis and the basic information needed to process and identify the trace. The trace headers are not intended to be a repository for significant amounts of ancillary data. If significant amounts of ancillary data need to be exchanged, it is recommended that SEG Ancillary Data Standard data set(s) be used.

Table 3 Trace Header

240-byte Trace Header	
Byte	**Description**
1-4[5]	Trace sequence number within line — Numbers continue to increase if the same line continues across multiple SEG Y files.
5-8	Trace sequence number within SEG Y file — Each file starts with trace sequence one.
9-12[5]	Original field record number.
13-16[5]	Trace number within the original field record.
17-20	Energy source point number — Used when more than one record occurs at the same effective surface location. It is recommended that the new entry defined in Trace Header bytes 197-202 be used for shotpoint number.
21-24	CDP ensemble number
25-28	Trace number within the CDP ensemble — Each ensemble starts with trace number one.

[5] Strongly recommended that this information always be recorded.

240-byte Trace Header

Byte	Description	
29-30[5]	Trace identification code: -1 = Other 0 = Unknown 1 = Seismic data 2 = Dead 3 = Dummy 4 = Time break 5 = Uphole 6 = Sweep 7 = Timing 8 = Waterbreak 9 = Near-field gun signature 10 = Far-field gun signature 11 = Seismic pressure sensor 12 = Multicomponent seismic sensor - Vertical component 13 = Multicomponent seismic sensor - Cross-line component 14 = Multicomponent seismic sensor - In-line component 15 = Rotated multicomponent seismic sensor - Vertical component 16 = Rotated multicomponent seismic sensor - Transverse component 17 = Rotated multicomponent seismic sensor - Radial component 18 = Vibrator reaction mass 19 = Vibrator baseplate 20 = Vibrator estimated ground force 21 = Vibrator reference 22 … N = optional use, (maximum N = 32,767)	
31-32	Number of vertically summed traces yielding this trace. (1 is one trace, 2 is two summed traces, etc.)	
33-34	Number of horizontally stacked traces yielding this trace. (1 is one trace, 2 is two stacked traces, etc.)	
35-36	Data use: 1 = Production 2 = Test	
37-40	Distance from center of the source point to the center of the receiver group (negative if opposite to direction in which line is shot).	
41-44	Receiver group elevation (all elevations above sea level are positive and below sea level are negative).	The scalar in Trace Header bytes 69-70 applies to these values. The units are feet or meters as specified in Binary File Header bytes 3255-3256)
45-48	Surface elevation at source.	
49-52	Source depth below surface (a positive number).	
53-56	Datum elevation at receiver group.	
57-60	Datum elevation at source.	
61-64	Water depth at source.	

240-byte Trace Header

Byte	Description	
65- 68	Water depth at group.	
69-70	Scalar to be applied to all elevations and depths specified in Trace Header bytes 41-68 to give the real value. Scalar = 1, +10, +100, +1000, or +10,000. If positive, scalar is used as a multiplier; if negative, scalar is used as a divisor.	
71-72	Scalar to be applied to all coordinates specified in Trace Header bytes 73-88 and to bytes Trace Header 181-188 to give the real value. Scalar = 1, +10, +100, +1000, or +10,000. If positive, scalar is used as a multiplier; if negative, scalar is used as divisor.	
73-76	Source coordinate - X.	If the coordinate units are in seconds of arc, the X values represent longitude and the Y values latitude. A positive value designates the number of seconds east of Greenwich Meridian or north of the equator and a negative value designates the number of seconds south or west.
77-80	Source coordinate - Y.	
81-84	Group coordinate - X.	
85-88	Group coordinate - Y.	
89-90	Coordinate units: 1 = Length (meters or feet) 2 = Seconds of arc 3 = Decimal degrees 4 = Degrees, minutes, seconds Note: To encode ±DDDMMSS bytes 89-90 equal = ±DDD*10^4 + MM*10^2 + SS with bytes 71-72 set to 1; To encode ±DDDMMSS.ss bytes 89-90 equal = ±DDD*10^6 + MM*10^4 + SS*10^2 with bytes 71-72 set to -100.	
91-92	Weathering velocity. (ft/s or m/s as specified in Binary File Header bytes 3255-3256)	
93-94	Subweathering velocity. (ft/s or m/s as specified in Binary File Header bytes 3255-3256)	
95-96	Uphole time at source in milliseconds.	Time in milliseconds as scaled by the scalar specified in Trace Header bytes 215-216.
97-98	Uphole time at group in milliseconds.	
99-100	Source static correction in milliseconds.	
101-102	Group static correction in milliseconds.	
103-104	Total static applied in milliseconds. (Zero if no static has been applied,)	
105-106	Lag time A — Time in milliseconds between end of 240-byte trace identification header and time break. The value is positive if time break occurs after the end of header; negative if time break occurs before the end of header. Time break is defined as the initiation pulse that may be recorded on an auxiliary trace or as otherwise specified by the recording system.	

240-byte Trace Header

Byte	Description
107-108	Lag Time B — Time in milliseconds between time break and the initiation time of the energy source. May be positive or negative.
109-110	Delay recording time — Time in milliseconds between initiation time of energy source and the time when recording of data samples begins. In SEG Y rev 0 this entry was intended for deep-water work if data recording does not start at zero time. The entry can be negative to accommodate negative start times (i.e. data recorded before time zero, presumably as a result of static application to the data trace). If a non-zero value (negative or positive) is recorded in this entry, a comment to that effect should appear in the Textual File Header.
111-112	Mute time — Start time in milliseconds.
113-114	Mute time — End time in milliseconds.
115-116[5]	Number of samples in this trace.
117-118[5]	Sample interval in microseconds (µs) for this trace. The number of bytes in a trace record must be consistent with the number of samples written in the trace header. This is important for all recording media; but it is particularly crucial for the correct processing of SEG Y data in disk files (see Appendix C). If the fixed length trace flag in bytes 3503-3504 of the Binary File Header is set, the sample interval and number of samples in every trace in the SEG Y file must be the same as the values recorded in the Binary File Header. If the fixed length trace flag is not set, the sample interval and number of samples may vary from trace to trace.
119-120	Gain type of field instruments: 1 = fixed 2 = binary 3 = floating point 4 … N = optional use
121-122	Instrument gain constant.
123-124	Instrument early or initial gain (dB).
125-126	Correlated: 1 = no 2 = yes
127-128	Sweep frequency at start (Hz).
129-130	Sweep frequency at end (Hz).
131-132	Sweep length in milliseconds.

240-byte Trace Header	
Byte	**Description**
133-134	Sweep type: 1 = linear 2 = parabolic 3 = exponential 4 = other
135-136	Sweep trace taper length at start in milliseconds.
137-138	Sweep trace taper length at end in milliseconds.
139-140	Taper type: 1 = linear 2 = $\cos^2$ 3 = other
141-142	Alias filter frequency (Hz), if used.
143-144	Alias filter slope (dB/octave).
145-146	Notch filter frequency (Hz), if used.
147-148	Notch filter slope (dB/octave).
149-150	Low-cut frequency (Hz), if used.
151-152	High-cut frequency (Hz), if used.
153-154	Low-cut slope (dB/octave)
155-156	High-cut slope (dB/octave)
157-158	Year data recorded — The 1975 standard is unclear as to whether this should be recorded as a 2-digit or a 4-digit year and both have been used. For SEG Y revisions beyond rev 0, the year should be recorded as the complete 4-digit Gregorian calendar year (i.e. the year 2001 should be recorded as 2001_{10} ($7D1_{16}$)).
159-160	Day of year.
161-162	Hour of day (24 hour clock).
163-164	Minute of hour.
165-166	Second of minute.
167-168	Time basis code: 1 = Local 2 = GMT (Greenwich Mean Time) 3 = Other, should be explained in a user defined stanza in the Extended Textual File Header 4 = UTC (Coordinated Universal Time)
169-170	Trace weighting factor — Defined as 2^{-N} volts for the least significant bit. (N = 0, 1, …, 32767)
171-172	Geophone group number of roll switch position one.

240-byte Trace Header	
Byte	**Description**
173-174	Geophone group number of trace number one within original field record.
175-176	Geophone group number of last trace within original field record.
177-178	Gap size (total number of groups dropped).
179-180	Over travel associated with taper at beginning or end of line: 1 = down (or behind) 2 = up (or ahead)
181-184	X coordinate of CDP position of this trace (scalar in Trace Header bytes 71-72 applies).
185-188	Y coordinate of CDP position of this trace (scalar in bytes Trace Header 71-72 applies).
189-192	For 3-D poststack data, this field should be used for the in-line number. If one in-line per SEG Y file is being recorded, this value should be the same for all traces in the file and the same value will be recorded in bytes 3205-3208 of the Binary File Header.
193-196	For 3-D poststack data, this field should be used for the cross-line number. This will typically be the same value as the CDP number in Trace Header bytes 21-24, but this doesn't have to be the case.
197-200	Shotpoint number — This is probably only applicable to 2-D poststack data. Note that it is assumed that the shotpoint number refers to the source location nearest to the CDP location for a particular trace. If this is not the case, there should be a comment in the Textual Header explaining what the shotpoint number actually refers to.
201-202	Scalar to be applied to the shotpoint number in Trace Header bytes 197-200 to give the real value. If positive, scalar is used as a multiplier; if negative as a divisor; if zero the shotpoint number is not scaled (i.e. it is an integer. A typical value will be -10, allowing shotpoint numbers with one place of decimals).
203-204	Trace value measurement unit: -1 = Other (should be described in Data Sample Measurement Units Stanza) 0 = Unknown 1 = Pascal (Pa) 2 = Volts (v) 3 = Millivolts (mV) 4 = Amperes (A) 5 = Meters (m) 6 = Meters per second (m/s) 7 = Meters per second squared (m/s^2) 8 = Newton (N) 9 = Watt (W)

240-byte Trace Header	
Byte	**Description**
205-210	Transduction Constant — The multiplicative constant used to convert the Data Trace samples to the Transduction Units (specified in Trace Header bytes 211-212). The constant is encoded as a four-byte, two's complement integer (bytes 205-208) which is the mantissa and a two-byte, two's complement integer (bytes 209-210) which is the power of ten exponent (i.e. Bytes 205-208 * 10**Bytes 209-210).
211-212	Transduction Units — The unit of measurement of the Data Trace samples after they have been multiplied by the Transduction Constant specified in Trace Header bytes 205-210. -1 = Other (should be described in Data Sample Measurement Units stanza) 0 = Unknown 1 = Pascal (Pa) 2 = Volts (v) 3 = Millivolts (mV) 4 = Amperes (A) 5 = Meters (m) 6 = Meters per second (m/s) 7 = Meters per second squared (m/s^2) 8 = Newton (N) 9 = Watt (W)
213-214	Device/Trace Identifier — The unit number or id number of the device associated with the Data Trace (i.e. 4368 for vibrator serial number 4368 or 20316 for gun 16 on string 3 on vessel 2). This field allows traces to be associated across trace ensembles independently of the trace number (Trace Header bytes 25-28).
215-216	Scalar to be applied to times specified in Trace Header bytes 95-114 to give the true time value in milliseconds. Scalar = 1, +10, +100, +1000, or +10,000. If positive, scalar is used as a multiplier; if negative, scalar is used as divisor. A value of zero is assumed to be a scalar value of 1.
217-240	Unassigned — For optional information.

7.2. Trace Data

Trace Data follows each Trace Header. The seismic data in a SEG Y file is organized into ensembles of traces or as a series of stacked traces. When the trace data is organized into ensembles of traces, it is strongly recommended that the ensemble type be identified (Binary File Header bytes 3229-3230).

Appendix A. Writing SEG Y Data to a Disk File

On modern UNIX and PC systems, a disk file is defined at the operating system level as a byte stream without any structure. It has become common practice for SEG Y data to be streamed into a disk file, without any kind of encapsulation. Such a disk file can only be read by software that comprehends the SEG Y format, since it must use certain values in the SEG Y headers to reconstruct the original record stream and thus this must really be regarded as a special form of encapsulation for SEG Y. This appendix describes the rules that must be followed when SEG Y data is written to a disk file (i.e. fixed disk, floppy disk, CD-ROM, MO disk, etc.).

The first 3600-bytes of the file are the “traditional” SEG Y File Header (i.e. the 3200-byte Textual File Header followed by the 400-byte Binary Header). The Binary Header may be followed by zero or more 3200-byte Textual File Header Extension records, as indicated in bytes 3505-3506 of the Binary Header.

The first Data Trace record, beginning with the 240-byte Trace Header, immediately follows the Binary File Header or if supplied, the last Extended Textual File Header. The number of bytes of Data Trace sample values that follow the Trace Header is determined from the value for number of samples in bytes 115-116 in the Trace Header, together with the sample format code in bytes 3225-3226 of the Binary Header. For format codes 1, 2, 4 and 5 the number of bytes of sample data is four times the number of samples. For format code 3, the number of bytes of sample data is twice the number of samples. For format code 8, the number of bytes of sample data is the same as the number of samples.

The Trace Header for the second Data Trace in the file follows immediately after the sample data for the first trace and so on for subsequent traces in the file.

All values are written to the disk file using “big-endian” byte ordering, the same as if the file were being written to tape. Data written to disk using “little-endian” byte ordering will not be SEG Y rev 1 compatible. For exchange purposes the Textual File Header and Extended Textual File Headers may be written in EBCDIC or ASCII character code.

Appendix B. SEG Y Tape Labels

In order to bring SEG Y into line with SEG D Rev 2, a label may be written at the front of a SEG Y file on unformatted, removable media (e.g. magnetic tape). This is a single record consisting of 128 bytes of ASCII characters and is very similar to the RP66 Storage Unit Label. A SEG Y tape label is optional and is only valid on SEG Y files written to unformatted, removable media. However, a label must be present if the blocking scheme described in Appendix C is being used. In this case the label must appear as a separate 128-byte record at the beginning of the file. There must be no file mark between the label record and the first data record.

If the recording medium supports multiple partitions, each partition is treated in isolation as if it were a separate unit. Thus, if labels are being used, each partition must begin with a label. Data from one partition can not "run-over" into a subsequent partition. Each partition must be capable of being decoded in isolation. On one recording medium, it is permissible to mix partitions containing SEG Y data with partitions containing non-SEG Y formatted information.

The format of a SEG Y Tape Label is summarized in Table 4.

Table 4 SEG Y Tape Label

Field	Description	Bytes	Start - end byte
1	Storage Unit Sequence Number	4	1 - 4
2	SEG Y Revision	5	5 - 9
3	Storage Unit Structure (fixed or variable)	6	10 - 15
4	Binding Edition	4	16 - 19
5	Maximum Block Size	10	20 - 29
6	API Producer Organization Code	10	30 - 39
7	Creation Date	11	40 - 50
8	Serial Number	12	51 - 62
9	Reserved	6	63 - 68
10	Storage Set Identifier	60	69 - 128

Field 1
The Storage Unit Sequence Number is an integer in the range 1 to 9999 that indicates the order in which the current storage unit occurs in the storage set. The first storage unit of a storage set has sequence number 1, the second 2 and so on. This number is represented using the characters 0 to 9, right justified with leading blanks if needed to fill out the field (no leading zeros). The right-most character is in byte 4 of the label. *This field is optional.* If not used, it must be blank (filled with blank characters). This implies that this is the only storage unit within the storage set. Separate storage sets should be used for different data types.

Field 2
The SEG Y Revision field indicates which revision of SEG Y was used to record the data on this tape. SY1.0 indicates that the data was recorded using SEG Y Revision 1. *This field is required.*

Field 3
Storage Unit Structure is a name indicating the record structure of the storage unit. This name is left justified with trailing blanks if needed to fill out the field. The leftmost character is in byte 10 of the label. For SEG Y Revision 1 tapes, this field must contain "RECORD". *This field is required.*

"RECORD" - Records may be of variable length, ranging up to the block size length specified in the maximum block size field of the storage unit label (if not zero). If the maximum block size specified is zero, records may be of any length.

Field 4
Binding Edition is the character B in byte 16 of the label followed by a positive integer in the range 1 to 999 (no leading zeros), left justified with trailing blanks if needed to fill out the field. The integer value corresponds to the edition of the Part 3 of the API RP66 standard used to describe the physical binding of the logical format to the storage unit. *This field is required.*

Field 5
Maximum Block Size is an integer in the range of 0 to 4,294,967,295 (2^{32}-1), indicating the maximum block length for the

storage unit, or 0 (zero) if undeclared. This number is represented using the characters 0 to 9, right justified, with leading blanks if necessary to fill out the field (no leading zeros). The rightmost character is byte 29 of the label. *A valid value or 0 (zero) must be recorded.*

Field 6
Producer Organization Code is an integer in the range of 0 to 4,294,967,295 (2^{32}-1) indicating the organization code of the storage unit producer. This number is represented using the characters 0 to 9, right justified, with leading blanks if necessary to fill out the field (no leading zeros). The rightmost character is byte 39 of the label*. This field is required.*

Organization codes are assigned and maintained by POSC. To request a new organization code contact:

POSC
9801 Westheimer Road, Suite 450
Houston, Texas 77042

Telephone (713) 267-5109
Website www.POSC.org

Field 7
Creation date is the earliest date that any current information was recorded on the storage unit. The date is represented in the form dd-MMM-yyyy, where yyyy is the year (e.g. 1996), MMM is one of (JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC) and dd is the day of the month in the range 1 to 31. Days 1 to 9 may have one leading blank. The separator is a hyphen (code 45_{10}). *This field is required.*

Field 8
Serial Number is an ID used to distinguish the storage unit from other storage units in an archive of an enterprise. The specification and management of serial numbers is delegated to organizations using this standard. This field may be empty (i.e. may contain all blanks, in which case no serial number is specified).

Field 9
This field is reserved and should be recorded as all blanks (code 32_{10}).

Field 10
Storage Set Identifier is a descriptive name for the storage set. Every storage unit in the same storage set shall have the same value for the storage set identifier in its storage unit label. A value may have embedded blanks and is non-blank if at least one character is different from blank (code 32_{10}). This field is intended to distinguish the storage set from other storage sets, but is not required to be unique. *A non-blank value shall be recorded.*

Appendix C. Blocking of SEG Y Files on Tape

This appendix describes a simple blocking scheme for writing SEG Y files to tape, based on the blocking scheme described in the SEG D Rev 2 standard. This is effectively a special encapsulation layer for SEG Y and may be necessary with certain tape devices that require a large block size to use the tape efficiently. Note however that this is not the only way to achieve this and it may be preferable to use another encapsulation scheme such as SEG RODE.

In the following explanation, a SEG Y record means a record defined in the SEG Y standard (i.e. a 3200-byte Textual File Header record, a 400-byte Binary Header record, a 3200-byte Extended Textual File Header record or a 240-byte trace record with its associated Data Trace). A tape record means a variable length physical record written to the tape device.

A tape containing SEG Y data written using this blocking scheme must begin with a SEG Y tape label, as described in Appendix

B. The label must be written as a separate tape record 128 bytes long. If the tape medium supports partitioning, each partition is treated in isolation and must have its own label.

Each subsequent tape record may contain one or more SEG Y records concatenated together. Each tape record must contain an integral number of SEG Y records (i.e. the start of a tape record must coincide with the start of a SEG Y record). The first tape record following the SEG Y Tape Label must begin with the 3200-byte Textual File Header record. For all tape records in a file, the record length must be less than the maximum record length for the tape medium being used.

A SEG Y reader program that comprehends this blocking scheme must unblock the data to reproduce the original SEG Y record stream. In particular, it must examine the number of samples recorded in each trace header to determine the actual length of the trace record.

When this blocking scheme is being used, it is permitted to end one SEG Y file and start a new one either with or without an intervening file mark. If a file mark is present, it signifies the end of a SEG Y file and a file mark must be followed by either a tape record beginning with a 3200-byte Textual File Header record or another file mark. Alternatively, a new SEG Y file can be identified by a tape record beginning with a 'C' character ($C3_{16}$ in EBCDIC or ASCII 43_{16}), which is taken as the start of a 3200-byte Textual File Header record. It follows that the start of a SEG Y file must start on a tape record boundary (i.e. any tape record contains data from only one SEG Y file). In either case, a double file mark signifies end of data.

Appendix D. Extended Textual Stanzas

All stanza keywords are required unless the stanza definition specifically notes that the inclusion of a keyword is optional.

D-1. Location Data

D-1.1 Stanza for Location Data

The Location Data stanza identifies the system to which the coordinates for the source, group or CDP location given in trace header bytes 73-88 or 181-188 are referenced. Without this identification these coordinates are ambiguous.

Table 5 Stanza for Location Data

Stanza Header and Keyword	**Format**	**Comment**
((Location Data))	Text	Stanza name
The following keywords apply to all Coordinate Reference Systems (CRS):		
CRS type =	From enumerated list:	Projected = map grid. Geographic = latitude, longitude; and in the case of a 3-D CRS

Stanza Header and Keyword	Format	Comment
	projected geographic vertical compound	additionally ellipsoidal height. Vertical = gravity-related height system. Compound = a quasi-three-dimensional set of coordinates comprised of either a geographical 2-D or a projected CRS with a vertical CRS.
CRS name =	Text	The name of the Coordinate Reference System.
Geodetic Datum name =	Text	The name of the Geodetic Datum.
Prime Meridian (PM) name =	Text	Mandatory if not “Greenwich”. (Note: most, but not all, Coordinate Reference Systems use Greenwich as the prime meridian).
PM Greenwich value =	Real Number	The longitude of the CRS’s prime meridian relative to the Greenwich meridian, positive if east of Greenwich. Not required if Prime Meridian name = “Greenwich”.
PM Greenwich value unit name =	Text	Not required if Prime Meridian name = “Greenwich”.
Ellipsoid name =	Text	
Ellipsoid semi-major axis =	Real Number	
Semi-major axis unit name =	Text	
Ellipsoid inverse flattening =	Real Number	
Coordinate System (CS) axis 1 name =	Text	The name or abbreviation of the axis for the coordinates in trace header bytes 73-76, 81-84 and 181-184. For example: easting, X, E, or longitude.
CS axis 1 orientation =	Text	The positive direction for axis 1. For example: “east”, or “north”.
Coordinate System (CS) axis 2 name =	Text	The name or abbreviation of the axis for the coordinates in trace header bytes 77-80, 85-88 and 185-188. For example: northing, Y, N, or latitude.
CS axis 2 orientation =	Text	The positive direction for axis 2. For example: “north” or “east”.

Stanza Header and Keyword	**Format**	**Comment**
The following keywords are additionally required for Projected Coordinate Reference Systems:		
Projection zone name =	Text	
Projection method name =	Text	For example: “Transverse Mercator”
Projection parameter 1 name =	Text	The number and type of parameters is dependent upon the projection method. For Transverse Mercator the five parameters required are: latitude of natural origin longitude of natural origin scale factor at natural origin false easting false northing
Projection parameter 1 value =	Real Number	
Projection parameter 1 unit name =	Text	
Projection parameter 2 name =	Text	
Projection parameter 2 value =	Real Number	
Projection parameter 2 unit name =	Text	
…		
Projection parameter 7 name =	Text	
Projection parameter 7 value =	Real Number	
Projection parameter 7 unit name =	Text	

D-1.2 Example Stanza for Location Data

((Location Data))

CRS type =	projected
CRS name =	NAD27 / Texas South Central
Geodetic Datum name =	North American Datum 1927
Ellipsoid name =	Clarke 1866
Ellipsoid semi-major axis =	6378206.4
Semi-major axis unit name =	meter
Ellipsoid inverse flattening =	294.9786982
Coordinate System axis 1 name =	Y
CS axis 1 orientation =	north
Coordinate System axis 2 name =	X
CS axis 2 orientation =	east
Projection zone name =	Texas CS27 South Central zone
Projection method name =	Lambert Conic Conformal (2SP)
Projection parameter 1 name =	latitude of first standard parallel
Projection parameter 1 value =	28.23
Projection parameter 1 unit name =	DDD.MMSSsss
Projection parameter 2 name =	latitude of second standard parallel
Projection parameter 2 value =	30.17
Projection parameter 2 unit name =	DDD.MMSSsss
Projection parameter 3 name =	latitude of false origin

Projection parameter 3 value =	27.5
Projection parameter 3 unit name =	DDD.MMSSsss
Projection parameter 4 name =	longitude of first standard parallel
Projection parameter 4 value =	-99
Projection parameter 4 unit name =	DDD.MMSSsss
Projection parameter 5 name =	easting at false origin
Projection parameter 5 value =	2000000.0
Projection parameter 5 unit name =	US survey foot
Projection parameter 6 name =	northing at false origin
Projection parameter 6 value =	0.0
Projection parameter 6 unit name =	US survey foot

D-2. Bin Grid Definition

D-2.1 Stanza for Bin Grid Definition

The Bin Grid Definition stanza defines a bin grid including its relationship to a projected coordinate reference system (map grid). The projected coordinate reference system must be defined in a Location Data stanza (see section D-1). The content of this Bin Grid Definition stanza follows the provisions of the UKOOA P6/94 and P6/98 formats.

The bin grid is the relative coordinate framework which defines a matrix of evenly spaced points referred to as the bin nodes. The term bin node is used instead of the term bin center and refers to the locations where the bin grid lines intersect.

The bin grid is defined by a pair of orthogonal axes designated the I and the J axes, with the I axis rotated 90 degrees clockwise from the J axis. The order of specifying bin grid coordinates will be the I value followed by the J value (I , J). The choice of I , J axes is made to avoid any confusion between bin grid (I,J) and map grid (E,N) coordinates. Axes may be labeled by users as they wish within their own software, including such terms as In-line and Cross-line, Row and Column, x and y, Line and Trace. There is no industry accepted common terminology for axis labeling and terms such as In-line and Cross-line are used in contradictory ways by different users. For the purpose of data exchange through SEG-Y the only reference is to the I and J axes.

Coordinates of three check nodes are required to permit numerical checks on the bin grid definition parameters. These nodes should be distributed distant from each other within the area of coverage.

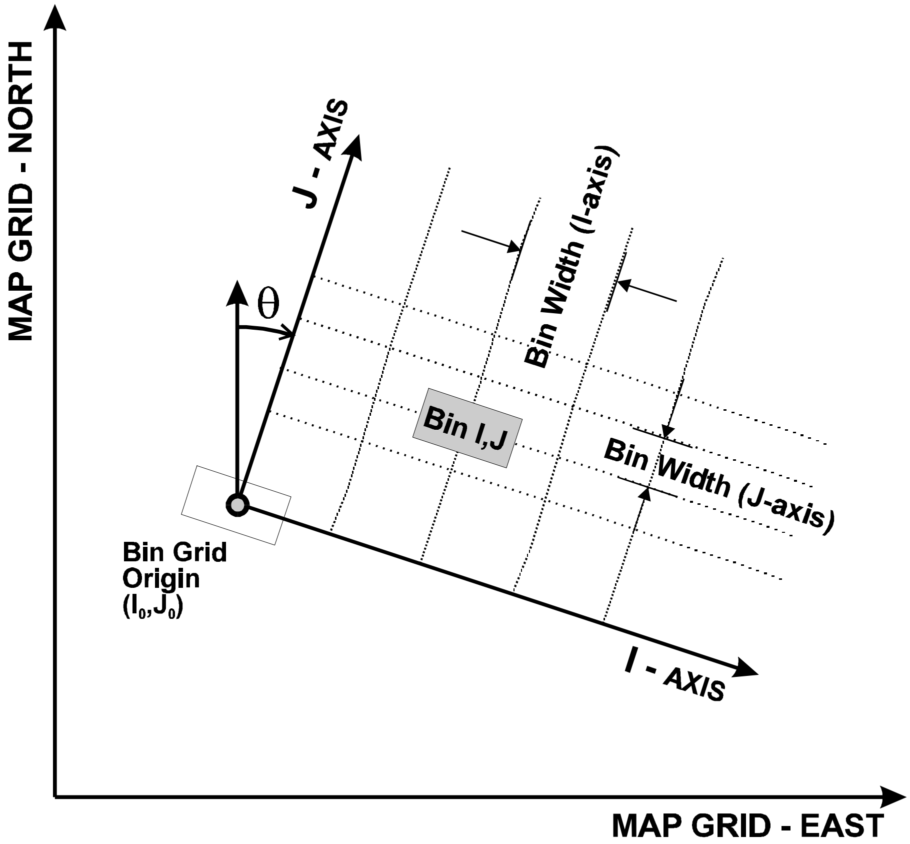

Figure 2 Bin grid definition

Table 6 Stanza for Bin Grid Definition

Stanza Header and Keyword	Format	Comment
((Bin Grid Definition))	Text	Stanza name
Bin grid name =	Text	Text description of the defined bin grid.
Bin grid origin I coordinate =	Real Number	Bin grid I coordinate at the bin grid origin.
Bin grid origin J coordinate =	Real Number	Bin grid J coordinate at the bin grid origin. The positive J axis is orientated 90 degrees counter clockwise from the positive I axis.
Bin grid origin Easting =	Real Number	Map grid Easting coordinate at the bin grid origin.
Bin grid origin Northing =	Real Number	Map grid Northing coordinate at the bin grid origin.
Scale factor of bin grid =	Real Number	Map grid scale factor at any bin node within the bin grid, preferably the center of the area of coverage.
Scale factor node I coordinate =	Real Number	Bin grid I coordinate of the bin node at which the scale factor (above) has been defined.

Stanza Header and Keyword	Format	Comment
Scale factor node J coordinate =	Real Number	Bin grid J coordinate of the bin node at which the scale factor (above) has been defined.
Nominal bin width on I axis =	Real Number	Nominal separation of bin nodes in the I-axis direction. Units are those of the projected coordinate reference system (map grid).
Nominal bin width on J axis =	Real Number	Nominal separation of bin nodes in the J-axis direction. Units are those of the projected coordinate reference system (map grid).
Grid bearing of bin grid J axis =	Real Number	Bearing of the positive direction of the bin grid J-axis defined as a map grid bearing, measured clockwise from map grid north.
Grid bearing unit name =	Text	The name of the angle unit for the bin grid bearing.
Bin node increment on I axis =	Real Number	Increment value between adjacent bin grid nodes in the I-axis direction.
Bin node increment on J axis =	Real Number	Increment value between adjacent bin grid nodes in the J-axis direction.
First check node I coordinate =	Real Number	
First check node J coordinate =	Real Number	
First check node Easting =	Real Number	
First check node Northing =	Real Number	
Second check node I coordinate =	Real Number	
Second check node J coordinate =	Real Number	
Second check node Easting =	Real Number	
Second check node Northing =	Real Number	
Third check node I coordinate =	Real Number	
Third check node J coordinate =	Real Number	
Third check node Easting =	Real Number	
Third check node Northing =	Real Number	

D-2.2 Example for Bin Grid Definition

((Bin Grid Definition))

Bin grid name =	Marine X final migrated volume
Bin grid origin I coordinate =	1.0
Bin grid origin J coordinate =	1.0
Bin grid origin Easting =	456781.0
Bin grid origin Northing =	5836723.0
Scale factor of bin grid =	0.99984
Scale factor node I coordinate =	1.0

Scale factor node J coordinate =	1.0
Nominal bin width on I axis =	25.0
Nominal bin width on J axis =	12.5
Grid bearing of bin grid J axis =	20
Grid bearing unit name =	degree
Bin node increment on I axis =	1
Bin node increment on J axis =	1
First check node I coordinate =	334.0000
First check node J coordinate =	235.0000
First check node Easting =	465602.94
First check node Northing =	5836624.30
Second check node I coordinate =	1352.0000
Second check node J coordinate =	955.0000
Second check node Easting =	492591.98
Second check node Northing =	5836377.16
Third check node I coordinate =	605.0000
Third check node J coordinate =	955.0000
Third check node Easting =	475046.03
Third check node Northing =	5842763.36

D-3. Data Geographic Extent & Coverage Perimeter

The Data Geographic Extent stanza describes the geographical extent of data in geographical (latitude/longitude), map grid and/or bin grid coordinates. The Coverage Perimeter stanza describes the perimeter of a 3-D data set. The geographical and projected (map grid) coordinate reference systems must be defined in Location Data stanza (see section D-1). The bin grid must be defined in a Bin Grid Definition stanza (see section D-2). The content of the data extent and coverage perimeter stanzas follow the provisions of the UKOOA P6/98 format.

The Coverage Perimeter stanza allows for the description of the following coverages:

- The total coverage of all data within the data set through the coordinates of a series of points describing the perimeter of the total coverage.
- Full-fold coverage through the coordinates of a series of points describing the outer perimeter of the full-fold coverage.
- Islands within the full-fold coverage with less than full-fold through the coordinates of a series of points describing the outer perimeter of the null full-fold coverage.
- Islands within the total coverage within which there is no coverage through the coordinates of a series of points describing the outer perimeter of the null fold coverage.

Figure 3 describes these concepts by showing the various data extents and coverage perimeters for a seismic survey encompassing a platform undershoot.

For processed data sets (near-trace cubes, migrated volumes, etc.), the fold will be affected by various processing steps (trace summation, offset rejection, migration, etc.). These processed data sets can be represented by either a Total Coverage Perimeter or a Full-fold Perimeter. The type of processed data set should be stated using the coverage extent comment keyword.

Wherever a detailed perimeter is known for a data set, the perimeter should be included in the exchange file. Bin grid and/or map grid coordinates may be given for each node of each perimeter. The data set extent can then be easily derived from the detailed perimeter. However, given the practical importance of the data set extent (e.g. used for loading of data onto workstations), the extent should also be defined explicitly in bin grid, map grid and/or latitude and longitude terms through a data extent stanza.

The Data Set Extent provides the user with a simple representation of the area covered by the survey for mapping and data management purposes, rather than a precise representation of the fold of coverage of a binning system or process.

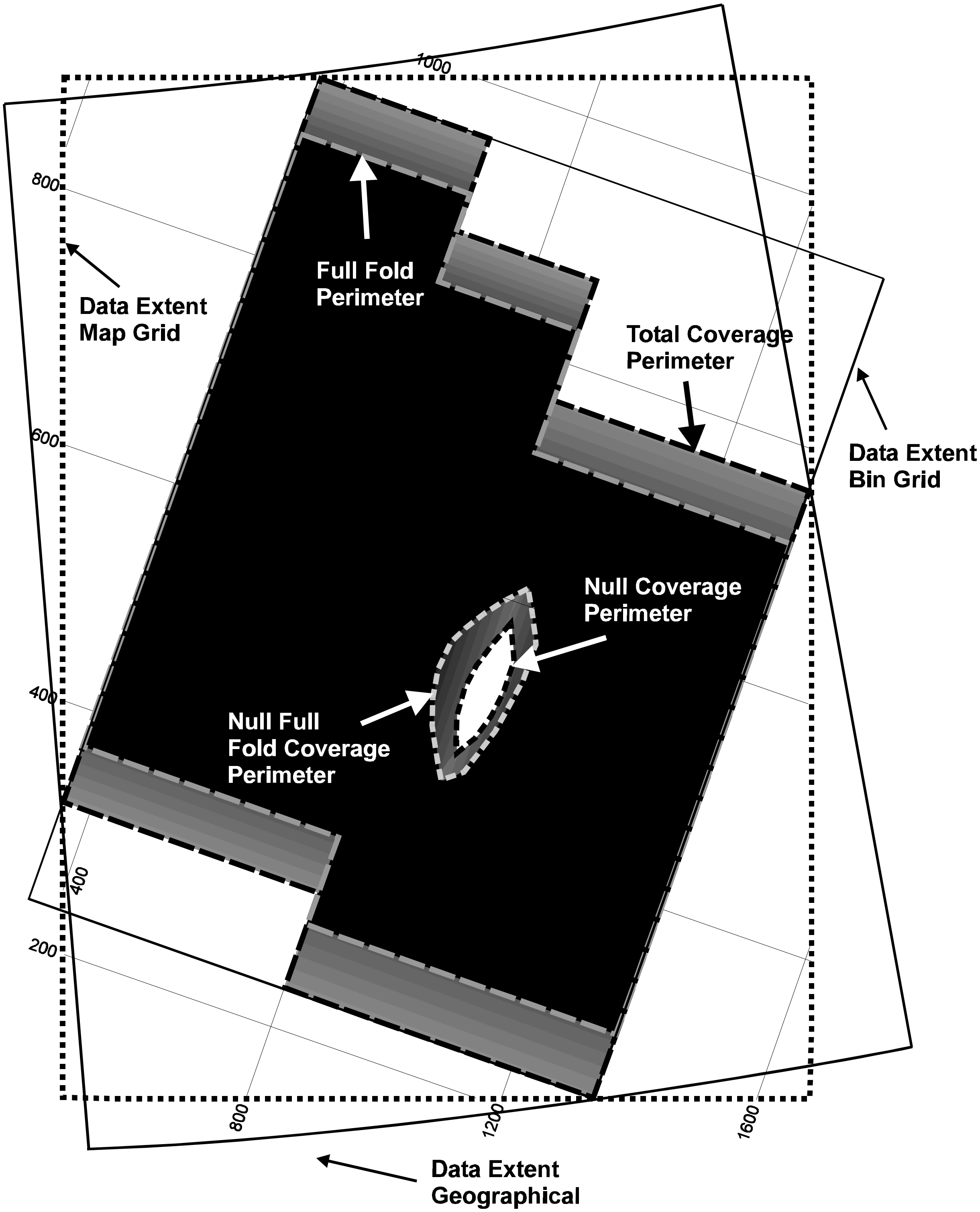

Figure 3 Various data extents and coverage perimeters for a seismic survey

D-3.1 Stanzas for Data Geographic Extent

The Data Geographic Extent stanza describes the geographical extent of data in geographical (latitude/longitude), map grid and/or bin grid coordinates.

Table 7 Stanza for Data Geographic Extent

Stanza Header and Keyword	Format	Comment
((Data Geographic Extent))	Text	Stanza name
Minimum Easting =	Real Number	Map grid Easting of the bin node with the minimum value of map grid Easting.
Maximum Easting =	Real Number	Map grid Easting of the bin node with the maximum value of map grid Easting.
Minimum Northing =	Real Number	Map grid Northing of the bin node with the minimum value of map grid Northing.
Maximum Northing =	Real Number	Map grid Northing of the bin node with the maximum value of map grid Northing.
Minimum Latitude =	Real Number	Latitude of the bin node with the minimum latitude value.
Maximum Latitude =	Real Number	Latitude of the bin node with the maximum latitude value.
Minimum Longitude =	Real Number	Longitude of the bin node with the minimum longitude value.
Maximum Longitude =	Real Number	Longitude of the bin node with the maximum longitude value.
Geographical coordinate unit name =	Text	The name of the units in which the maximum and minimum latitude and longitude are given.
Minimum I coordinate =	Real Number	Bin grid I coordinate of the bin node with the minimum value of I coordinate.
Maximum I coordinate =	Real Number	Bin grid I coordinate of the bin node with the maximum value of I coordinate.
Minimum J coordinate =	Real Number	Bin grid J coordinate of the bin node with the minimum value of J coordinate.
Maximum J coordinate =	Real Number	Bin grid J coordinate of the bin node with the maximum value of J coordinate.
Data Extent comment =	Text	May be repeated as necessary.

D-3.2 Example for Data Geographic Extent

```
 ((Data Geographic Extent))
Minimum Easting =                 465966.28
Maximum Easting =                 491792.63
Minimum Northing =                5827921.28
Maximum Northing =                5845080.18
```

Minimum Latitude =	52.4516782
Maximum Latitude =	52.3604359
Minimum Longitude =	2.3209385
Maximum Longitude =	2.5243181
Geographical coordinate unit name =	DDD.MMSSsss
Minimum I coordinate =	334.0000
Maximum I coordinate =	1352.0000
Minimum J coordinate =	235.0000
Maximum J coordinate =	955.0000
Data Extent comment =	UKOOA P6/98 example. The CRS referred to (WGS 84 / UTM zone 31N) is not consistent with the example shown in section D-1.1.

D-3.3 Stanza for Coverage Perimeter

The Coverage Perimeter stanza describes the perimeter of a 3-D data set.

Table 8 Stanza for Coverage Perimeter

Stanza Header and Keyword	Format	Comment
((Coverage Perimeter))	Text	Stanza name
Coverage type =	From enumerated list: total full-fold null full-fold null fold	See section preamble for description of enumerated coverage types.
Perimeter coordinate type =	From enumerated list: I,J E,N I,J,E,N	The coverage perimeter may be described by bin grid and/or map grid coordinates.
Perimeter node number =	Integer	The number of nodes describing the perimeter. For an n-sided perimeter the perimeter node number should be n.
Perimeter node coordinates =	2 or 4 comma-separated real numbers	I, J and/or Easting, Northing coordinates of the bin node. Repeat the first point at the end of the list: for an n-sided perimeter the perimeter node coordinate record count should be n+1.
Coverage Perimeter comment =	Text	May be repeated as necessary.

D-3.4 Example Stanza for Coverage Perimeter

This example is based on Figure 3.

```
((Coverage Perimeter))
Coverage type =                      total
Perimeter coordinate type =          I,J,E,N
Perimeter node number =              10
Perimeter node coordinates =         334.0000,955.0000,468680.63,5845080.18
Perimeter node coordinates =         654.0000,955.0000,476196.97,5842344.46
Perimeter node coordinates =         654.0000,875.0000,475855.00,5841404.91
Perimeter node coordinates =         900.0000,875.0000,481633.18,5839301.83
Perimeter node coordinates =         900.0000,768.0000,481175.81,5838045.19
Perimeter node coordinates =         1352.0000,768.0000,491792.63,5834180.98
Perimeter node coordinates =         1352.0000,235.0000,489514.29,5827921.28
Perimeter node coordinates =         802.0000,235.0000,476595.58,5832623.30
Perimeter node coordinates =         802.0000,320.0000,476958.92,5833621.57
Perimeter node coordinates =         334.0000,320.0000,465966.28,5837622.56
Perimeter node coordinates =         334.0000,955.0000,468680.63,5845080.18
[Coverage Perimeter]
Coverage type =                      full-fold
Perimeter coordinate type =          I,J
Perimeter node number =              10
Perimeter node coordinates =         334.0000,908.0000
Perimeter node coordinates =         654.0000,908.0000
Perimeter node coordinates =         654.0000,833.0000
Perimeter node coordinates =         900.0000,833.0000
Perimeter node coordinates =         900.0000,721.0000
Perimeter node coordinates =         1352.0000,721.0000
Perimeter node coordinates =         1352.0000,289.0000
Perimeter node coordinates =         802.0000,289.0000
Perimeter node coordinates =         802.0000,368.0000
Perimeter node coordinates =         334.0000,368.0000
Perimeter node coordinates =         334.0000,908.0000
Coverage Perimeter comment =         48 fold data
[Coverage Perimeter]
Coverage type =                      null full-fold
Perimeter coordinate type =          E,N
Perimeter node number =              9
Perimeter node coordinates =         482101.92,5835620.00
Perimeter node coordinates =         482874.75,5834820.00
Perimeter node coordinates =         482067.29,5834063.19
Perimeter node coordinates =         481388.11,5833804.99
Perimeter node coordinates =         480572.36,5833902.39
Perimeter node coordinates =         479705.57,5834736.58
Perimeter node coordinates =         479274.40,5835452.12
Perimeter node coordinates =         479633.25,5835707.21
Perimeter node coordinates =         480739.50,5835823.27
Perimeter node coordinates =         482101.92,5835620.00
```

```
[Coverage Perimeter]
Coverage type =                    null fold
Perimeter coordinate type =        I,J,E,N
Perimeter node number =            8
Perimeter node coordinates =       958.0000,579.0000,481730.25,5835329.67
Perimeter node coordinates =       978.0000,552.0000,482084.61,5834841.59
Perimeter node coordinates =       980.0000,512.0000,481960.60,5834354.72
Perimeter node coordinates =       958.0000,481.0000,481311.34,5834178.73
Perimeter node coordinates =       946.0000,468.0000,480973.91,5834128.64
Perimeter node coordinates =       900.0000,498.0000,480021.67,5834874.23
Perimeter node coordinates =       920.0000,522.0000,480594.03,5834985.11
Perimeter node coordinates =       958.0000,582.0000,481743.07,5835364.90
Perimeter node coordinates =       958.0000,579.0000,481730.25,5835329.67
```

D-4. Data Sample Measurement Units

D-4.1 Stanza for Data Sample Measurement Units

The Data Sample Measurement Units stanza provides a means of defining a measurement unit other than the measurement units defined in the Trace Header bytes 203-204.

Table 9 Stanza for Data Sample Measurement Units

Stanza and Keyword	Format	Comment
((Measurement Units))	Text	Stanza name
Data Sample Measurement Unit =	Text 60	A textual description of the measurement unit used for the data samples (i.e. millivolts, meters)
Volt conversion =	Real Number	The multiplicative constant that converts the Data Sample Measurement Unit to Volts.

D-4.2 Example stanza for Data Sample Measurement Units

```
((Measurement Units))
Data Sample Measurement Unit =     Millivolts
Volt conversion =                  0.001
```

D-5. Processing History

The Processing History stanza provides a means to track the processing history of the seismic data traces.

D-5.1 Stanza for Processing History

Table 10 Stanza for Processing History

Stanza and Keyword	Format	Comment
((Processing History))	Text	Stanza name
The following six entries are repeated as needed to define all processing steps applied to the data traces.		
Processing Company =	Text 60	
Processing Software =	Text 60	
Input Data Set =	Text 60	Data set name or data set id of the data traces being processed
Processing Date =	Text 60	Date in YYYYMMDD-HHMMSS format
Process Applied =	Text 60	Name of the algorithm or program being applied to the data traces
Process Parameters =	Text 60	

D-5.2 Example stanza for Processing History

((Processing History))	
Processing Company =	Expert Processing Inc.
Processing Software =	Omega
Input Data Set =	$ADBigDo_FieldSeq463
Processing Date =	20010519-231643
Processing Applied =	SEG-D edit
Process Parameters =	MP factor applied
Processing Company =	Expert Processing Inc.
Processing Software =	Omega
Input Data Set =	$ADBigDo_FieldSeq463
Processing Date =	20010519-231643
Processing Applied =	Trace select/sort
Process Parameters =	Data traces, Common Rcv Sort
Processing Company =	Expert Processing Inc.
Processing Software =	Omega
Input Data Set =	$ADBigDo_FieldSeq463_Edit
Processing Date =	20010520-115959
Processing Applied =	Predictive deconvolution
Process Parameters =	Surface consistent, 130 ms, 3 windows

In this example, a field data set was edited and sorted into common receiver order as an initial process. In a second step, the edited data was deconvolved using a surface consistent deconvolution operator.

Appendix E. Data Word Format

Code 1 — 4-byte hexadecimal exponent data (i.e. IBM single precision floating point)

Bit	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Byte 1	S	C_6	C_5	C_4	C_3	C_2	C_1	C_0
Byte 2	Q_{-1}	Q_{-2}	Q_{-3}	Q_{-4}	Q_{-5}	Q_{-6}	Q_{-7}	Q_{-8}
Byte 3	Q_{-9}	Q_{-10}	Q_{-11}	Q_{-12}	Q_{-13}	Q_{-14}	Q_{-15}	Q_{-16}
Byte 4	Q_{-17}	Q_{-18}	Q_{-19}	Q_{-20}	Q_{-21}	Q_{-22}	Q_{-23}	0

S = sign bit. — (One = negative number).

C = excess 64 hexadecimal exponent. — This is a binary exponent of 16. The exponent has been biased by 64 such that it represents $16^{(ccccccc-64)}$ where CCCCCCC can assume values from 0 to 127.

Q_{1-23} = magnitude fraction. — This is a 23-bit positive binary fraction (i.e., the number system is sign and magnitude). The radix point is to the left of the most significant bit (Q_{-1}) with the MSB being defined as 2^{-1}. The sign and fraction can assume values from ($1 - 2^{-23}$ to $-1 + 2^{-23}$). It must always be written as a hexadecimal left justified number. If this fraction is zero, the sign and exponent must also be zero (i.e., the entire word is zero. Note that bit 7 of Byte 4 must be zero in order to guarantee the uniqueness of the start of scan.

Value = S.QQQQ,QQQQ,QQQQ,QQQQ,QQQQ,QQQ x $16^{(ccccccc-64)}$

Code 2 — 4-byte, two's complement integer

Bit	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Byte 1	S	I_{30}	I_{29}	I_{28}	I_{27}	I_{26}	I_{25}	I_{24}
Byte 2	I_{22}	I_{22}	I_{21}	I_{20}	I_{19}	I_{18}	I_{17}	I_{16}
Byte 3	I_{15}	I_{14}	I_{13}	I_{12}	I_{11}	I_{10}	I_9	I_8
Byte 4	I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0

Value = S ($I_{30}*2^{30} + I_{29}*2^{29} + \ldots + I_1*2^1 + I_0*2^0$)

Code 3 — 2-byte, two's complement integer

Bit	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Byte 3	S	I_{14}	I_{13}	I_{12}	I_{11}	I_{10}	I_9	I_8
Byte 4	I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0

Value = S $(I_{14}*2^{14} + I_{13}*2^{13} + \ldots + I_1*2^1 + I_0*2^0)$

Code 4 — 32-bit fixed point with gain values (Obsolete)

Bit	0	1	2	3	4	5	6	7
Byte 1	0	0	0	0	0	0	0	0
Byte 2	G_7	G_6	G_5	G_4	G_3	G_2	G_1	G_0
Byte 3	S	I_{14}	I_{13}	I_{12}	I_{11}	I_{10}	I_9	I_8
Byte 4	I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0

Value = S $(I_{14}*2^{14} + I_{13}*2^{13} + \ldots + I_1*2^1 + I_0*2^0) * 2^{GGGGGGGG}$

Code 5 — 4-byte, IEEE Floating Point

The IEEE format is fully documented in the IEEE standard, "ANSI/IEEE Std 754 - l985", available from the IEEE. The IEEE format is summarized as follows:

Bit	0	1	2	3	4	5	6	7
Byte 1	S	C_7	C_6	C_5	C_4	C_3	C_2	C_1
Byte 2	C_0	Q_{-1}	Q_{-2}	Q_{-3}	Q_{-4}	Q_{-5}	Q_{-6}	Q_{-7}
Byte 3	Q_{-8}	Q_{-9}	Q_{-10}	Q_{-11}	Q_{-12}	Q_{-13}	Q_{-14}	Q_{-15}
Byte 4	Q_{-16}	Q_{-17}	Q_{-18}	Q_{-19}	Q_{-20}	Q_{-21}	Q_{-22}	Q_{-23} (Note 1)

The value (v) of a floating-point number represented in this format is determined as follows:

if e = 255 & f = 0. .v = NaN	Not-a-Number (see Note 2)
if e = 255 & f = 0. .v = $(-1)^s *\infty$	Overflow
if 0 < e < 255. . . .v = $(-1)^s *2^{e-127} *(1.f)$	Normalized
if e = 0 & f ¹ 0. . .v = $(-1)^s *2^{e-126} *(0.f)$	Denormalized
if e = 0 & f = O. . .v = $(-1)^s *0$	± zero

where e = binary value of all C's (exponent)
f = binary value of all Q's (fraction)

NOTES:

1. Bit 7 of byte 4 must be zero to guarantee uniqueness of the start of scan in the Multiplexed format (0058). It may be non zero in the demultiplexed format (8058).
2. A Not-a-Number (NaN) is interpreted as an invalid number. All other numbers are valid and interpreted as described above.

Code 8 — 1-byte, two's complement integer

Bit	0	1	2	3	4	5	6	7
Byte 4	S	I_6	I_5	I_4	I_3	I_2	I_1	I_0

Value = S ($I_6*2^6 + I_5*2^5 + \ldots + I_1*2^1 + I_0*2^0$)

Appendix F. EBCDIC and ASCII Codes

Table 11 IBM 3270 Char Set Ref Ch 10, GA27-2837-9, April 1987

Char	EBCDIC	ASCII	Description
	(hex)	(hex)	
NU	x00	x00	Null (nul)
SH	x01	x01	Start of heading (soh)
SX	x02	x02	Start of text (stx)
EX	x03	x03	End of text (etx)
ST	x04	x9C	String terminator (st)
HT	x05	x09	Character tabulation (ht)
SA	x06	x86	Start of selected area (ssa)
DT	x07	7F	Delete (del)
EG	x08	x97	End of guarded area (epa)
RI	x09	x8D	Reverse line feed (ri)
S2	x0A	x8E	Single-shift two (ss2)
VT	x0B	x0B	Line tabulation (vt)
FF	x0C	x0C	Form feed (ff)
CR	x0D	x0D	Carriage return (cr)
SO	x0E	x0E	Shift out (so)
SI	x0F	x0F	Shift in (si)
DL	x10	x10	Datalink escape (dle)
D1	x11	x11	Device control one (dc1)
D2	x12	x12	Device control two (dc2)
D3	x13	x13	Device control three (dc3)
OC	x14	x9D	Operating system command (osc)
NL	x15	x85	Next line (nel)
BS	x16	x08	Backspace (bs)
ES	x17	x87	End of selected area (esa)
CN	x18	x18	Cancel (can)
EM	x19	x19	End of medium (em)
P2	x1A	x92	Private use two (pu2)
S3	x1B	x8F	Single-shift three (ss3)
FS	x1C	x1C	File separator (is4)
GS	x1D	x1D	Group separator (is3)
RS	x1E	x1E	Record separator (is2)
US	x1F	x1F	Unit separator (is1)
PA	x20	x80	Padding character (pad)
HO	x21	x81	High octet preset (hop)
BH	x22	x82	Break permitted here (bph)
NH	x23	x83	No break here (nbh)
IN	x24	x84	Index (ind)
LF	x25	x0A	Line feed (lf)
EB	x26	x17	End of transmission block (etb)
EC	x27	x1B	Escape (esc)
HS	x28	x88	Character tabulation set (hts)
HJ	x29	x89	Character tabulation with justification (htj)
VS	x2A	x8A	Line tabulation set (vts)
PD	x2B	x8B	Partial line forward (pld)
PU	x2C	x8C	Partial line backward (plu)

Char	EBCDIC	ASCII	Description
	(hex)	(hex)	
EQ	x2D	x05	Enquiry (enq)
AK	x2E	x06	Acknowledge (ack)
BL	x2F	x07	Bell (bel)
DC	x30	x90	Device control string (dcs)
P1	x31	x91	Private use one (pu1)
SY	x32	x16	Synchronous idle (syn)
TS	x33	x93	Set transmit state (sts)
CC	x34	x94	Cancel character (cch)
MW	x35	x95	Message waiting (mw)
SG	x36	x96	Start of guarded area (spa)
ET	x37	x04	End of transmission (eot)
SS	x38	x98	Start of string (sos)
GC	x39	x99	Single graphic character introducer (sgci)
SC	x3A	x9A	Single character introducer (sci)
CI	x3B	x9B	Control sequence introducer (csi)
D4	x3C	x14	Device control four (dc4)
NK	x3D	x15	Negative acknowledge (nak)
PM	x3E	x9E	Privacy message (pm)
SB	x3F	x1A	Substitute (sub)
SP	x40	x20	Space, Blank
¢	x4A	xA2	Cent sign
.	x4B	x2E	Full stop, Period
<	x4C	x3C	Less-than sign
(	x4D	x28	Left parenthesis
+	x4E	x2B	Plus sign
|	x4F	x7C	Vertical line, Logical OR
&	x50	x26	Ampersand
!	x5A	x21	Exclamation mark
$	x5B	x24	Dollar sign
*	x5C	x2A	Asterisk
)	x5D	x29	Right parenthesis
;	x5E	x3B	Semicolon
¬	x5F	xAC	Not sign
-	x60	x2D	Hyphen, Minus
/	x61	x2F	Solidus, Forward slash
BB	x6A	xA6	Broken bar
,	x6B	x2C	Comma
%	x6C	x25	Percent sign
_	x6D	x5F	Low line, Underline, Underscore
>	x6E	x3E	Greater-than sign
?	x6F	x3F	Question mark
`	x79	x60	Grave accent
:	x7A	x3A	Colon
#	x7B	x23	Number sign, Pound sign, hash mark
@	x7C	x40	Commercial at
'	x7D	x27	Apostrophe
=	x7E	x3D	Equals sign
"	x7F	x22	Quotation mark
a	x81	x61	Latin small letter A
b	x82	x62	Latin small letter B
c	x83	x63	Latin small letter C
d	x84	x64	Latin small letter D
e	x85	x65	Latin small letter E
f	x86	x66	Latin small letter F
g	x87	x67	Latin small letter G
h	x88	x68	Latin small letter H
i	x89	x69	Latin small letter I
j	x91	x6A	Latin small letter J
k	x92	x6B	Latin small letter K
l	x93	x6C	Latin small letter L
m	x94	x6D	Latin small letter M
n	x95	x6E	Latin small letter N
o	x96	x6F	Latin small letter O
p	x97	x70	Latin small letter P
q	x98	x71	Latin small letter Q
r	x99	x72	Latin small letter R
~	xA1	x7E	Tilde
s	xA2	x73	Latin small letter S
t	xA3	x74	Latin small letter T

Char	EBCDIC	ASCII	Description
	(hex)	(hex)	
u	xA4	x75	Latin small letter U
v	xA5	x76	Latin small letter V
w	xA6	x77	Latin small letter W
x	xA7	x78	Latin small letter X
y	xA8	x79	Latin small letter Y
z	xA9	x7A	Latin small letter Z
{	xC0	x7B	Left curly bracket
A	xC1	x41	Latin capital letter A
B	xC2	x42	Latin capital letter B
C	xC3	x43	Latin capital letter C
D	xC4	x44	Latin capital letter D
E	xC5	x45	Latin capital letter E
F	xC6	x46	Latin capital letter F
G	xC7	x47	Latin capital letter G
H	xC8	x48	Latin capital letter H
I	xC9	x49	Latin capital letter I
}	xD0	x7D	Right curly bracket
J	xD1	x4A	Latin capital letter J
K	xD2	x4B	Latin capital letter K
L	xD3	x4C	Latin capital letter L
M	xD4	x4D	Latin capital letter M
N	xD5	x4E	Latin capital letter N
O	xD6	x4F	Latin capital letter O
P	xD7	x50	Latin capital letter P
Q	xD8	x51	Latin capital letter Q
R	xD9	x52	Latin capital letter R
\	xE0	x5C	Reverse solidus, Back slash
S	xE2	x53	Latin capital letter S
T	xE3	x54	Latin capital letter T
U	xE4	x55	Latin capital letter U
V	xE5	x56	Latin capital letter V
W	xE6	x57	Latin capital letter W
X	xE7	x58	Latin capital letter X
Y	xE8	x59	Latin capital letter Y
Z	xE9	x5A	Latin capital letter Z
0	xF0	x30	Digit zero
1	xF1	x31	Digit one
2	xF2	x32	Digit two
3	xF3	x33	Digit three
4	xF4	x34	Digit four
5	xF5	x35	Digit five
6	xF6	x36	Digit six
7	xF7	x37	Digit seven
8	xF8	x38	Digit eight
9	xF9	x39	Digit nine
AC	xFF	x9F	Application program command (apc)
NUL	x00	x00	Nul
SOH	x01	x01	Start of heading (soh)
STX	x02	x02	Start of text (stx)
ETX	x03	x03	End of text (etx)
EOT	x37	x04	End of transmission (eot)
ENQ	x2D	x05	Enquiry (enq)
ACK	x2E	x06	Acknowledge (ack)
alert	x2F	x07	Bell (bel)
BEL	x2F	x07	Bell (bel)
backspace	x16	x08	Backspace (bs)
tab	x05	x09	Character tabulation (ht)
newline	x25	x0A	Line feed (lf)
vertical-tab	x0B	x0B	Line tabulation (vt)
form-feed	x0C	x0C	Form feed (ff)
carriage-return	x0D	x0D	Carriage return (cr)
DLE	x10	x10	Datalink escape (dle)
DC1	x11	x11	Device control one (dc1)
DC2	x12	x12	Device control two (dc2)
DC3	x13	x13	Device control three (dc3)
DC4	x3C	x14	Device control four (dc4)
NAK	x3D	x15	Negative acknowledge (nak)
SYN	x32	x16	Synchronous idle (syn)
ETB	x26	x17	End of transmission block (etb)
CAN	x18	x18	Cancel (can)

Char	EBCDIC	ASCII	Description
	(hex)	(hex)	
SUB	x3F	x1A	Substitute (sub)
ESC	x27	x1B	Escape (esc)
IS4	x1C	x1C	File separator (is4)
IS3	x1D	x1D	Group separator (is3)
intro	x1D	x1D	Group separator (is3)
IS2	x1E	x1E	Record separator (is2)
IS1	x1F	x1F	Unit separator (is1)
DEL	x07	x7F	Delete (del)
space	x40	x20	Space
!	x5A	x21	Exclamation mark
"	x7F	x22	Quotation mark
#	x7B	x23	Number sign
$	x5B	x24	Dollar sign
%	x6C	x25	Percent sign
&	x50	x26	Ampersand
'	x7D	x27	Apostrophe
(	x4D	x28	Left parenthesis
)	x5D	x29	Right parenthesis
*	x5C	x2A	Asterisk
+	x4E	x2B	Plus sign
,	x6B	x2C	Comma
-	x60	x2D	Hyphen, Minus
.	x4B	x2E	Full stop, Period
/	x61	x2F	Solidus, Slash
0	xF0	x30	Digit Zero
1	xF1	x31	Digit one
2	xF2	x32	Digit two
3	xF3	x33	Digit three
4	xF4	x34	Digit four
5	xF5	x35	Digit five
6	xF6	x36	Digit six
7	xF7	x37	Digit seven
8	xF8	x38	Digit eight
9	xF9	x39	Digit nine
:	x7A	x3A	Colon
;	x5E	x3B	Semicolon
<	x4C	x3C	Less-than sign
=	x7E	x3D	Equals sign
>	x6E	x3E	Greater-than sign
?	x6F	x3F	Question mark
@	x7C	x40	Commercial at
[	X00	x5B	Left square bracket
\	xE0	x5C	Reverse solidus, Backslash
]	x00	x5D	Right square bracket
^	x00	x5E	Circumflex, Caret
_	x6D	x5F	Low line, Underscore
`	x79	x60	Grave accent
{	xC0	x07B	Left curly bracket
|	x4F	x7C	Vertical line
}	xD0	x7D	Right curly bracket
~	xA1	x7E	Tilde

Appendix G. References

Barry, K. M., Cavers, D. A. and Kneale, C. W., 1975, Report on recommended standards for digital tape formats: Geophysics, 40**,** no. 02, 344-352.

Booth, Algan, Duke, Guyton, Norris, Stainsby, Theriot, Wildgoose and Wilhelmsen, SEG Rode Format Record Oriented Data Encapsulation: Geophysics, 61, no. 05, 1545-1558.

Northwood, E. J., Wisinger, R. C., and Bradley, J. J., 1967, Recommended standards for digital tape formats: Geophysics, v32, p. 1073-1084

SEG Comm. Field Tape Std., 1994, Digital field tape standards - SEG-D, revision 1 (special report): Geophysics, 59, no. 04, 668-684.

Wood, Behn, Borresen, Guyton, Miles, O'Neill, Oishi and Scales, SEG-D, Rev 2 SEG Dield Tape Standards, Geophysics, 62, no. 03, 1004-1031