Designation: E2560 – 17

Standard Specification for Data Format for Pavement Profile[1]

This standard is issued under the fixed designation E2560; the number immediately following the designation indicates the year of original adoption or, in the case of revision, the year of last revision. A number in parentheses indicates the year of last reapproval. A superscript epsilon (ε) indicates an editorial change since the last revision or reapproval.

1. Scope

1.1 This specification describes a data file format for pavement profile.

1.2 This specification describes the variables and sizes of all data that will be stored in the file. The file is in binary format, and is fully documented in this specification.

1.3 This specification is designed to be independent of hardware platforms, computer languages, and operating system (OS).

1.4 *This standard does not purport to address all of the safety concerns, if any, associated with its use. It is the responsibility of the user of this standard to establish appropriate safety and health practices and determine the applicability of regulatory limitations prior to use.*

1.5 *This international standard was developed in accordance with internationally recognized principles on standardization established in the Decision on Principles for the Development of International Standards, Guides and Recommendations issued by the World Trade Organization Technical Barriers to Trade (TBT) Committee.*

2. Referenced Documents

2.1 *ASTM Standards:*[2]

E867 Terminology Relating to Vehicle-Pavement Systems

2.2 *IEEE Standards:*[3]

IEEE 754–2008 (2008) Floating-Point Arithmetic

3. Terminology

3.1 *Definitions:*

3.1.1 Terminology used in this specification conforms to the definitions included in Terminology E867.

3.2 *Definitions of Terms Specific to This Standard:*

3.2.1 *signed*—integer capable of representing negative values.

3.2.2 *unsigned*—integer only capable of representing non-negative values.

3.2.3 *int8*—data type for an 8-bit, unsigned integer.

3.2.4 *int32*—data type for a 32-bit, signed integer.

3.2.5 *single*—data type for a 32-bit, signed real number, such as, single precision IEEE floating point.

3.2.6 *string*—data type for a variable-length ASCII string. No null character is included at the end of the string. A separate field defines the length of the string.

3.2.7 *3-byte string*—an ASCII string of three characters in length. No null character is included at the end of the string.

3.2.8 *4-byte string*—an ASCII string of four characters in length. No null character is included at the end of the string.

3.2.9 *8-byte string*—an ASCII string of eight characters in length. No null character is included at the end of the string.

3.2.10 *array (numeric data type)*—sequence of data of the specified numeric data type. Only the values are stored; no information about the array is stored.

3.2.11 *array (string)*—ASCII strings separated by a tab. There is no tab after the last string.

3.3 *Symbols:*

3.3.1 *n*—total channels of elevation data.

3.3.2 *m*—total number of test locations (that is, data points).

4. Profile Data Specifications

4.1 *File Structure:*

4.1.1 The general file structure is divided into five sections: *(1)* File Header; *(2)* Metadata; *(3)* Longitudinal Profile Data; *(4)* Transverse Profile Data; and *(5)* File Trailer. The five sections are stored sequentially. (See Fig. 1.)

4.1.2 Each of these portions of the file is described in the following sections, as well as the data types and other descriptors that will be required by the file. The data will be written to the file sequentially, with the offsets listed in the file header as guides to find various portions of the file. It is important to note that all offsets are relative to the beginning of the file. Because offset values may not be known at the time of writing the file header, these values need not be written.

[1] This test method is under the jurisdiction of ASTM Committee E17 on Vehicle - Pavement Systems and is the direct responsibility of Subcommittee E17.31 on Methods for Measuring Profile and Roughness.

Current edition approved June 1, 2017. Published June 2017. Originally approved in 2007. Last previous edition approved in 2013 as E2560 – 13. DOI: 10.1520/E2560-17.

[2] For referenced ASTM standards, visit the ASTM website, www.astm.org, or contact ASTM Customer Service at service@astm.org. For *Annual Book of ASTM Standards* volume information, refer to the standard's Document Summary page on the ASTM website.

[3] Available from Institute of Electrical and Electronics Engineers, Inc. (IEEE), 445 Hoes Ln., P.O. Box 1331, Piscataway, NJ 08854-1331, http://www.ieee.org.

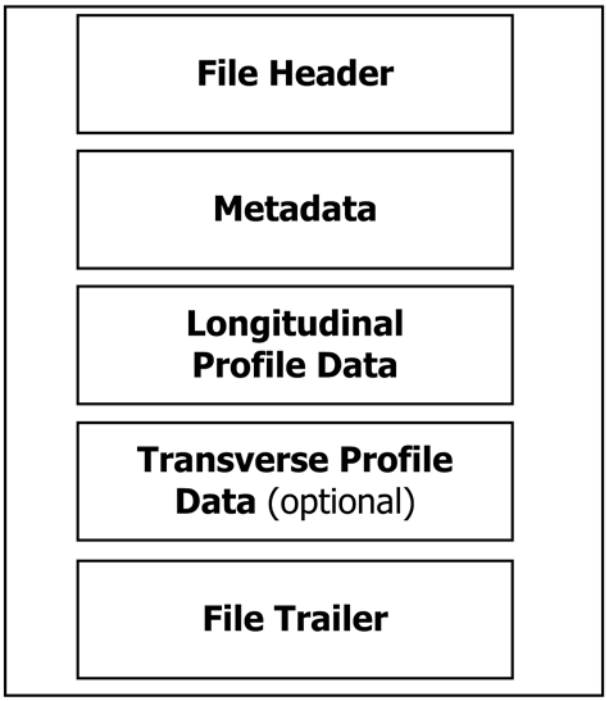

FIG. 1 Layout of the File Structure

However, spare space must still be reserved for the offsets so that values can be updated when known.

4.2 *File Header*—File header contains the information pertaining to the data file type, software version information, and information about the data contained (Table 1).

4.3 *Metadata:*

4.3.1 Metadata is structured, descriptive information about a resource, or data about data. Using metadata in the binary file format will allow generic operating on the data information about which the reader software has no prior knowledge. Also, metadata will allow scalable evolution of the data description without requiring simultaneous upgrades to all reader software.

4.3.2 The first value in the metadata portion will provide the number of metadata entries (MDE) (Table 2). Table 3 shows the information required to construct an appropriate MDE.

4.3.3 The metadata tags are listed in Table 4, and can be used in any number or order. If no metadata tags exist, number of MDEs = 0.

TABLE 2 Metadata

Variable Name	Data Type	Data
Number of MDEs	Int32	Number of MDEs

4.3.4 The names of the standard metadata entries (see Table 3) are not stored in the metadata entry to conserve space and more importantly, to allow for localization, that is, the file is not tied to one written language. User-defined metadata entries cannot be arrays and the data type is always String.

4.3.5 The storage convention for empty arrays is to store a one-byte value of the same data type as the array. For example, an array of singles with no elements would store a value of 0.

4.4 *Longitudinal Profile Data:*

4.4.1 There are two ways to store the profile data: location-wise and array-wise. The first method is appropriate for data recording during profile data collection to prevent data loss, while the other is appropriate for post-processing to speed up software reading and writing.

4.4.2 If the data storage format, from metadata tag #522 specifies location-wise storage, the longitudinal data will be stored as a sequence of current longitudinal distance followed by corresponding elevations of longitudinal sensors at this location, beginning at the left side of the vehicle. The next block of storage will store longitudinal distance and all elevation data for the next location, and so on. However, the location may not need to be stored if a specific data interval is given. (See Fig. 2.)

4.4.2.1 In general, if a location and elevation channels are recorded for each test location, every set of $n+1$ Singles (one distance data and n channels of elevation data) will be read as one profile location. If a specific data interval is included in the metadata, only n Singles will be read for each location. For example, if a standard interval exists and a single channel of profile data is present, only one Single will be read for each location. If two are present, then two Singles will be read per point.

4.4.2.2 The location-wise format is recommended for profiler data acquisition software. Storing the data after every sampling location allows for immediate writing to protect against data loss and reduce memory requirements.

4.4.3 If the data storage format from metadata tag #522 specifies array-wise storage, then the longitudinal data will be stored as a sequence of the longitudinal distance array followed

TABLE 1 File Header

Variable Name	Data Type	Data	Default Value
Signature	4-byte String	Identifies file as being written in the Standard Pavement Profile Format	"SPPF"
Version	4-byte String	Identifies the version number of the file format. This number is incremented if a change is made that breaks compatibility with previous versions of the format.	"1.05"
SW version	8-byte String	Identifier of the software that produced the file	for example, "TGPA1.00"
Metadata offset	Int32	Offset in bytes from the beginning of the file to the beginning of the metadata	N/A
Longitudinal offset	Int32	Offset in bytes from the beginning of the file to the beginning of the longitudinal profile data	N/A
Transverse data offset	Int32	Offset in bytes from the beginning of the file to the beginning of the transverse profile data	N/A

TABLE 3 Metadata Entries

Variable name	Data type	Data
Tag of MDE	Int32	Metadata tag (see Table 4)
Data type of MDE	Int32	Data type index of MDE (see Table 5)
Array size	Int32	"-1" if not an array. "0" if array is empty. Numbers greater than 0 specify the number of elements in the array. Even though arrays of strings are stored differently than other types of array, an array size should still be specified here.
Count	Int32	For data types "String" and "Array (String)," count = the number of bytes in the string. for other data types, count = 1.
Name length	Int32	For metadata entries listed in Table 4, this is 0. For user-defined entries, this value is the length of the name.
Name	String	Name of the metadata
MDE	varies	Information associated with tag of MDE

by the elevation array of each longitudinal sensor, beginning at the left side of the vehicle for all locations. (See Fig. 3.)

4.4.3.1 In general, if distance and elevation channels are recorded for each test location, *n+1* sets (one distance channel and *n* channels of elevation data) of *m* Singles will be stored in sequence, where *m* is the number of points. If a specific data interval is included in the metadata, only *n* sets of *m* Singles will be stored sequentially, with distance being calculated from the beginning of the test location by the software. For example, if a standard interval exists and a single channel of profile data is present, only one set of Singles will be stored. If two are present, then two sets of *m* Singles will be stored sequentially.

4.4.3.2 This data format is recommended for software that reads and writes the data during post-processing. Data stored as one continuous array (array-wise) can be read and processed much faster than the location-wise storage format.

4.5 *Transverse Profile Data*—The transverse elevation readings are treated the same as the longitudinal data.

4.6 *File Trailer*—The file trailer is used to signal the end of the file. (See Table 15.)

4.7 *Event Markers*—Event markers are defined by tags 528 to 531. These four arrays must all be of the same length.

4.8 *Sections:*

4.8.1 A section is defined by the use of two event markers. The first event marker is the start location of the section and the second event marker is the stop location. Special attention should be paid to lead-in and lead-out event markers. These two markers define the section that is bounded by the lead-in and lead-out. Please note: they do not define the lead-in and lead-out, but the section between them. An example of this follows:

4.8.1.1 A 1000-point profile has a lead-in of 50 points and a lead-out of 40 points. Points 0 to 49 will constitute the lead-in, so the event marker index for lead-in will be 50. Points 960 to 999 constitute the lead-out, so the event marker index for lead-out will be 959.

4.8.2 Tags 526 and 527 were defined before the use of event markers to define sections. These tags are no longer used.

4.8.3 Tag 531 is a recent addition to help ensure the integrity of the sections, even if the event markers are not in order. The tag is not required, but if present, must be the same length as tags 529 and 530. As this tag is new, there is no guarantee that file readers will use it. When writing a file, ensure that events are not sorted, but rather stored in the order created. Use tag 531 only to verify the order. An example follows, containing two sections and one event marker. The key for tags 311 and 531 are random ASCII strings that can be of any length. The only constraint is that the values cannot be duplicated for a given tag.

311 (Section keys)	57A9, GD89
312 (Section names)	Section 1, Section 2
528 (Event marker index)	300, 350, 699, 800, 1000
529 (Event marker text)	(blank), (blank), Event 1, (blank), (blank),
530 (Event marker type)	2, 3, 1, 2, 3
531 (Event marker section-related key)	65UW, 65UW, 7H89, 8GJK

4.8.4 Tags 532, 533, and 534 are used to record the geographical location of events. These values are not required to be set, but the arrays must be the same size as the other event marker arrays.

4.8.5 There are two ways to define the original geographical coordinates associated with a profile. Tags 318 to 323 define the start and stop coordinates for a profile. Or, tags 535, 536, and 541 can be used by the profiler to record the geographical location of multiple points in the profile. If there is not a coordinate for each profile point, tag 538 is required to associate each coordinate with a distance on the profile. Tags 539, 540, and 542 provide storage for a processed route. With either of these methods, there may be too little or too much data to create a route suitable for use. Because converting these methods into a usable route can take time, these tags allow for storage of the processed route.

4.9 *Images:*

4.9.1 Tag 305 can be used to store an image of the profile. This provides a method to quickly view the general shape of the profile without the need to draw the profile. It is recommended that the image be 48 pixels wide by 48 pixels high in PNG (portable network graphics) format. The image is stored as a byte array.

4.9.2 Tag 325 can be used to store an image of the geographical route of the profile. It is recommended that the image be 250 pixels wide by 250 pixels high in PNG format. The image is stored as a byte array.

5. Keywords

5.1 longitudinal profile; pavement profile; profile data specifications; transverse profile

TABLE 4 Metadata Tags and Descriptions

Tag	Name	Data Type	Notes
256 – 511: General Profiler and Location Information			
258	Title	String	Required
259	Profiler trade name and model number	String	
260	Vehicle identification	String	
261	Date data was collected—(yyyymmdd)	String	
262	Time data was collected—(hhmmss)	String	
263	Profiler operator name	String	
264	Average vehicle speed associated with data	Single	
265	Original filename before import	String	Read-only[B]
271	Agency district name	String	
272	Agency district number	Int32	
273	County name	String	
274	County number	Int32	
275	Nearby city name	String	
281	Roadway designation	String	
282	Lane identification	String	
283	Station number of beginning point	String	
284	Reference marker or milepost of beginning point	String	No longer used[C]
285	Pavement surface type (See Table 6)	Int32	
286	Direction of travel	String	Such as “northbound”
287	Station number of ending point	String	
288	Reference marker or milepost of ending point	String	No longer used
291	Ambient temperature	String	
292	Surface temperature	String	
293	Climactic conditions (see Table 7)	Int32	
294	Data history	String	Read-only
295	Date file last modified—(yyyymmdd)	String	Read-only
296	Time file last modified—(hhmmss)	String	Read-only
297	Date file imported from original file format—(yyyymmdd)	String	Read-only
298	Time file imported from original file format—(hhmmss)	String	Read-only
299	Run number (multiple runs—same location on the same day)	Int32	
300	Profiler type (see Table 8)	Int32	
301	Country name	String	
302	State/Province Name	String	
303	Wind speed	Single	
304	Wind direction	String	
305	Thumbnail image	Array (Int8)	Read-only
306	Start milepost	Single	
307	Stop milepost	Single	
308	Profiler direction (see Table 9)	Int32	
309	File key	String	Read-only
310	Profile keys	Array (String)	Read-only
311	Section keys	Array (String)	Read-only
312	Section names	Array (String)	Read-only
313	Comments	String	
314	Default section key	String	
315	Original file key	String	Read-only
316	Coordinate system (see Table 10)	Int32	
317	UTM zone	Int32	
318	Start longitude	Single	Units defined by Table 10
319	Start latitude	Single	Units defined by Table 10
320	Start elevation	Single	In metres
321	Stop longitude	Single	Units defined by Table 10
322	Stop latitude	Single	Units defined by Table 10
323	Stop elevation	Single	In metres
325	Route Image	Array (Int8)	
512 – 767 Longitudinal and Transverse Profile Information			
512	Number of longitudinal elevation channels	Int32	Required
513	Number of transverse elevation channels	Int32	Required
514	Number of longitudinal data points	Int32	Required
515	Number of transverse profiles data points	Int32	Required
516	Longitudinal distance between longitudinal data points	Single	
517	Longitudinal distance between transverse profiles	Single	

TABLE 4 *Continued*

Tag	Name	Data Type	Notes
518	Longitudinal sensor spacing from vehicle center (negative values to the left of vehicle center, positive to the right)	Array (Single)	Required
519	Transverse sensor spacing from vehicle center (negative values to the left of vehicle center, positive to the right)	Array (Single)	
520	Names for longitudinal sensors	Array (String)	Required
521	Names for transverse sensors	Array (String)	
522	Longitudinal data storage format (See Table 11)	Int32	Required
523	Channel type for each longitudinal profile (see Table 12)	Array (Int32)	
525	Profile offset (if linear distance adjustment or correlation is performed)	Single	
526	Profile start index to define the location of lead-in	Int32	No longer used
527	Profile stop index to define the location of lead-out	Int32	No longer used
528	Event marker index	Array (Int32)	
529	Event marker text	Array (String)	
530	Event marker type (See Table 13)	Array (Int32)	
531	Event marker section-related key	Array (String)	
532	Event marker longitude	Array (Single)	
533	Event marker latitude	Array (Single)	
534	Event marker altitude	Array (Single)	In metres
535	Logged Coordinate X	Array (Single)	Units defined by Table 10
536	Logged Coordinate Y	Array (Single)	Units defined by Table 10
538	Logged Coordinate Distance	Array (Single)	
539	Route Coordinate X	Array (Single)	Units defined by Table 10
540	Route Coordinate Y	Array (Single)	Units defined by Table 10
541	Logged Coordinate Z	Array (Single)	In meters
542	Route Coordinate Z	Array (Single)	In meters
768 – 1023: Measurement Units Information			
768	Units for longitudinal distances (see Table 14)	Int32	Required
769	Units for elevation data (see Table 14)	Int32	Required
770	Units of speed (see Table 14)	Int32	
771	Units of temperature (see Table 14)	Int32	
772	Units of sensor spacing (see Table 14)	Int32	
773	Do not use		
774	Do not use		
1024 – 2047[A]	Reserved for user defined metadata entries	String	

[A] Any tags not listed below 1024 are reserved for future use.
[B] "Read-only" indicates that the user should not be allowed to edit the value. Only the software or profiler should be allowed to change the value.
[C] Items no longer used are so indicated.

TABLE 5 Data Types

Index[A]	Data Type	Size (bytes)	Description
8	String	varies	one byte ASCII (no unicode support)
17	Int8	1	8-bit unsigned integer
3	Int32	4	32-bit signed integer
4	Single	4	Single precision IEEE floating point

[A] Data type index values follow Microsoft programming conventions.

TABLE 6 Pavement Surface Types

Value of Tag 285	Pavement Surface Type
0	Undefined
1	Portland cement concrete
2	Asphalt
3	Unpaved

TABLE 7 Current Climactic Conditions

Value of Tag 293	Climactic Conditions
0	Undefined
1	Sunny
2	Hazy/fog
3	Partly cloudy
4	Mostly cloudy
5	Overcast
6	Light rain/snow
7	Moderate rain
8	Heavy rain

TABLE 8 Profiler Type

Value of Tag 300	Description
1	High speed
2	Light weight
3	Manual

TABLE 9 Direction of Travel

Value of Tag 308	Description
1	Forward (in traffic flow)
2	Reverse (against traffic flow)

TABLE 10 Coordinate System

Value of Tag 316	Description
1	Universal Transverse Mercator (UTM) (metres)
3	World Geodetic System (WGS 84) (decimal degrees)
5	Spherical Mercator (metres)

TABLE 11 Data Storage Format

Value of Tag 522	Data Storage Format
1	Location-wise
2	Array-wise

TABLE 12 Channel Location

Value of Tag 523	Description
1	Left wheel path
2	Right wheel path
3	Centerline

TABLE 13 Event Marker Type

Value of Tag 530	Description
1	Generic marker
2	Section start
3	Section stop
4	Leave-out start
5	Leave-out stop
6	Lead-in (first point after lead-in stops)
7	Lead-out (last point before lead-out stops)

TABLE 14 Units

Value of Tags 768 – 1024	Unit
Distance and Elevation	
73	Mils
1	Inches
2	Feet
4	Miles
5	Milimetres
6	Centimetres
7	Metres
8	Kilometres
Speed	
24	Feet/second
28	Miles/hour
27	Metres/second
26	Kilometres/hour
Temperature	
35	Degrees Fahrenheit
33	Degrees Centigrade
Time	
36	Seconds

TABLE 15 File Trailer

Variable Name	Data type	Data	Default Value
End of file	3-byte String	indicates the end of the file	"@@@"

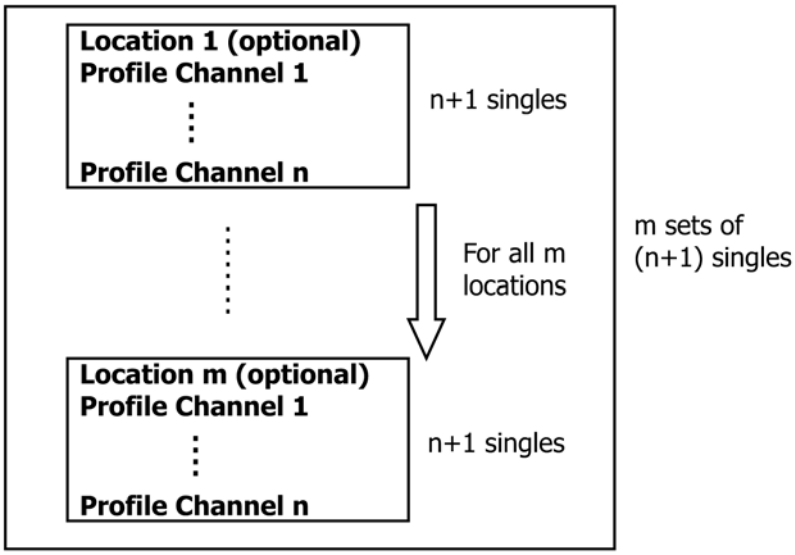

FIG. 2 Location-wise Storage

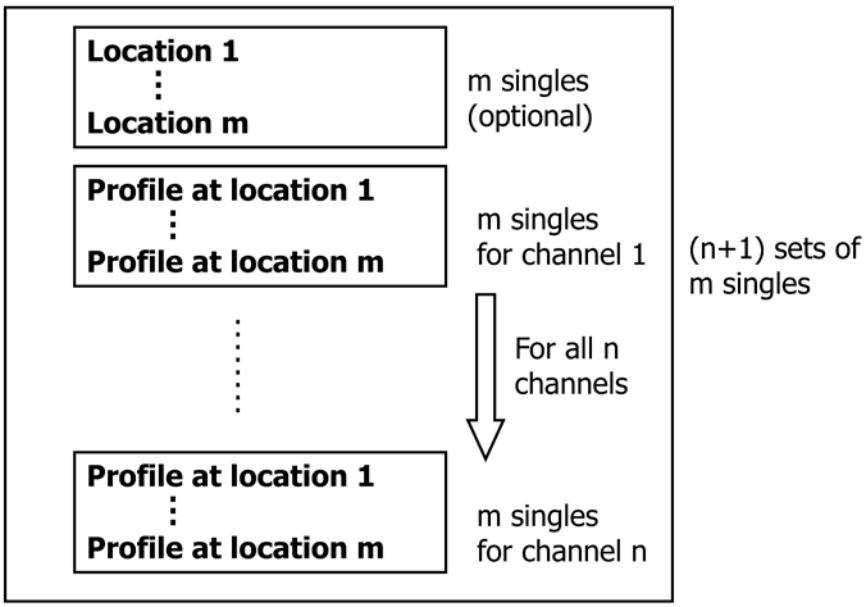

FIG. 3 Array-wise Storage

APPENDIX

X1. SAMPLE GUIDE

X1.1 A Sample File

X1.1.1 Instructions to prepare a sample file are given below. (See Table X1.1).

X1.1.2 A binary sample file (PPF-Sample.ppf) can be obtained from ASTM as an adjunct file to this standard.

X1.1.3 A comparison sample file with respect to the binary sample in Table X1.1 is given in ERD format in Fig. X1.1.

X1.1.4 The Profile Viewing and Analysis (ProVAL)[4] can be used to validate any ASTM E2560 files generated by users' programs.

[4] Available from www.RoadProfile.com.

TABLE X1.1 Sample File

Description	Value	Byte Length	Comment
	File Header		
Signature	SPPF	4	
Version	1.05	4	
Software Version	Writer01	8	
Metadata Offset	28	4	
Longitudinal Offset	401	4	
Transverse Offset	-1	4	
	Metadata Count		
	12	4	12 metadata entries
	General Metadata		
Tag	258	4	Title
Data type	8	4	
Array size	-1	4	
Count	51	4	
Name length	0	4	
Name		0	
MDE	"1993 RPUG Study, Dipstick, Section 1, Measurement 1"	51	
	Profile Metadata		
Tag	512	4	Longitudinal Profile Count
Data type	3	4	
Array size	-1	4	
Count	1	4	
Name length	0	4	
Name		0	
MDE	2	4	
Tag	513	4	Transverse Profile Count
Data type	3	4	
Array size	-1	4	
Count	1	4	
Name length	0	4	
Name		0	
MDE	0	4	
Tag	514	4	Longitudinal Point Count
Data type	3	4	
Array size	-1	4	
Count	1	4	
Name length	0	4	
Name		0	
MDE	10	4	
Tag	515	4	Transverse Point Count
Data type	3	4	
Array size	-1	4	
Count	1	4	
Name length	0	4	
Name		0	
MDE	0	4	
Tag	516	4	Longitudinal distance between longitudinal points
Data type	4	4	
Array size	-1	4	
Count	1	4	
Name length	0	4	
Name		0	
MDE	1	4	1 in.
Tag	518	4	Longitudinal sensor spacing
Data type	4	4	
Array size	2	4	
Count	1	4	

TABLE X1.1 *Continued*

Description	Value	Byte Length	Comment
Name length	0	4	
Name		0	
MDE	0 0	8	Values not specified in ERD, but required in PPF
Tag	520	4	Longitudinal sensor names
Data type	8	4	
Array size	2	4	
Count	30	4	
Name length	0	4	
Name		0	
MDE	“Left Elevation” [TAB] “Right Elevation”	30	
Tag	522	4	Longitudinal data storage format
Data type	3	4	
Array size	-1	4	
Count	1	4	
Name length	0	4	
Name		0	
MDE	2	4	Array-wise
Tag	523	4	Longitudinal profile type
Data type	3	4	
Array size	2	4	
Count	1	4	
Name length	0	4	
Name		0	
MDE	1 2	8	Left Right
	Units Metadata		
Tag	768	4	Longitudinal distance unit
Data type	4	4	
Array size	-1	4	
Count	1	4	
Name length	0	4	
Name		0	
MDE	2	4	Inches
Tag	769	4	Elevation unit
Data type	4	4	
Array size	-1	4	
Count	1	4	
Name length	0	4	
Name		0	
MDE	2	4	Inches
	Longitudinal Data		
Left	0.000000E+00 4.166670E-04 4.166670E-04 6.666670E-04 1.333330E-03 7.500000E-04 -3.000000E-03 -5.583330E-03 -6.250000E-03 -7.750000E-03	40	

TABLE X1.1 *Continued*

Description	Value	Byte Length	Comment
Right	0.000000E+00 -1.416670E-03 5.833330E-04 9.166670E-04 1.333330E-03 -1.666670E-03 -4.583330E-03 -5.000000E-03 -6.583330E-03 -8.250000E-03	40	
	Transverse Data		
		0	No transverse data
	File Trailer		
	@@@	3	

```
ERDFILEV2.00
      2,    529,     -1,     -1,      5,
1.00000    ,     -1,
TITLE   1993 RPUG Study, Dipstick, Section
1, Measurement 1
SHORTNAMLElev.  RElev.
LONGNAMELeft Elevation
Right Elevation
UNITSNAMft      ft
GENNAME Profile Elevation
Profile Elevation
XLABEL  Distance
XUNITS  ft
FORMAT  (2G14.6)
PROFINSTDipstick
HISTORY Converted to ERD format at 23:46,
Oct. 23, 1994
END
  0.000000      0.000000
  0.416667E-03 -0.141667E-02
  0.416667E-03  0.583333E-03
  0.666667E-03  0.916667E-03
  0.133333E-02  0.133333E-02
  0.750000E-03 -0.166667E-02
 -0.300000E-02 -0.458333E-02
 -0.558333E-02 -0.500000E-02
 -0.625000E-02 -0.658333E-02
 -0.775000E-02 -0.825000E-02
```

FIG. X1.1 A Comparison Sample File in ERD Format.