This page provides a description of the TIA Series Data (ESD) file format. This document only applies to files generated by TIA 3.x or later.

The ESD format is very flexible, as it allows for temporally and positionally tagged series of 1-D or 2-D data of any real or complex data type. All ES Vision spectrum and image series (version 3.x or later) are stored in this format.

You can write code to read ESD Files, and subsequently perform custom processing, analysis and presentation.

General Structure

An ESD File is a binary file consisting of six parts:

1. Header – Contains basic information for the Series File, including unique file identifiers, data types, calibrations, number and dimensionality of series file, and file offsets of the Data and Tag Offset Arrays.
2. Dimension Array – Stores information about the dimensionality of the data series.
3. Data Offset Array – Contain file offsets of individual data elements.
4. Tag Offset Array – Contain file offsets of individual data tags.
5. Data Elements – The actual individual data elements, stored as short-header binary objects.
6. Data Tags – The actual individual data tags, stored as short-header binary objects.

1. Header Format

The Header begins at byte 0 in the ESD File. The Header has the following format:

Byte Offset	Data Type	Data Size (Bytes)	Name	Description
0	Integer	2	ByteOrder	Byte ordering indicator – value 0x4949 ('II') indicates little-endian (PC) byte ordering. All series files are written with this byte ordering, where the least-significant byte is stored first.
2	Integer	2	SeriesID	Series identification word – value 0x0197 indicates TIA Series Data File
4	Integer	2	SeriesVersion	Version number word – indicates the version of the Series Data File. Should be 0x0220 for TIA 4.7.3 and above, otherwise 0x0210.
6	Integer	4	DataTypeID	Indicates the type of data object stored at each element in the series. May be one of the following values: 0x4120 – Data elements are 1-dimensional arrays 0x4122 – Data elements are 2-dimensional arrays
10	Integer	4	TagTypeID	Indicates the type of tab stored at each element in the series. May be one of the following values: 0x4152 – Tag is time only 0x4142 – Tag is 2-D position with time
14	Integer	4	TotalNumberElements	Indicates the total number of data elements and tags referred to by the Dimension Array. Equals the product of the dimension sizes, and corresponds to the total number of addressable indices in the series.
18	Integer	4	ValidNumberElements	Indicates the number of valid Data elements and Tags in the series data file. Normally equal to TotalNumberElements, may be less than TotalNumberElements if not all elements of the series file were written
22	Integer	SeriesVersion<=0x210 : 4 **SeriesVersion>=0x220 : 8**	OffsetArrayOffset	Indicates the absolute offset (in bytes) in the Series Data File of the beginning of the Data Offset Array.
26	Integer	4	NumberDimensions	Indicates the number of dimensions of the Series Data File. This indicates the number of dimensions

				of the indices, NOT the number of dimensions of the data.

2. Dimension Array Format

Immediately following the header, beginning at byte offset 30, is the Dimension Array. The Dimension Array consists of one or more immediately adjacent Dimension structures of the following form:

Byte Offset	Data Type	Data Size (Bytes)	Name	Description
0	Integer	4	DimensionSize	Indicates the number of elements in this dimension.
4	Float	8	CalibrationOffset	Indicates the calibration value at element CalibrationElement.
12	Float	8	CalibrationDelta	Indicates the calibration delta between elements of the series.
20	Integer	4	CalibrationElement	Indicates the element in the series which has a calibration value of CalibrationOffset.
24	Integer	4	DescriptionLength	Indicates the length of the Description string.
28	Char	*	Description	String of length DescriptionLength which describes this dimension.
*	Integer	4	UnitsLength	Indicates the length of the Units string.
*	Char	*	Units	String of length UnitsLength which is the name of units in this dimension.

As indicated by the above structure, the length of each Dimension structure is (32 + DescriptionLength + UnitsLength) bytes.

3. Data Offset Array Format

Beginning at byte offset OffsetArrayOffset is the Data Offset Array. This consists of TotalNumberElements Integers corresponding to the byte offsets of the individual data elements. The Integers are 4-byte for SeriesVersion <= 0x0210 and **8-byte for SeriesVersion >= 0x0220.**

4. Tag Offset Array Format

Immediately following the Data Offset Array is the Tag Offset Array. This consists of TotalNumberElements Integers corresponding to the byte offsets of the individual data tags. The Integers are 4-byte for SeriesVersion <= 0x0210 and **8-byte for SeriesVersion >= 0x0220.**

5. Data Element Format

The data is stored in one of the following formats:

1-D data

For 1-D data (DataTypeID = 0x4120), the data is stored in a structure of the following form:

Byte Offset	Data Type	Data Size (Bytes)	Name	Description
0	Float	8	CalibrationOffset	Indicates the calibration value at element CalibrationElement.
8	Float	8	CalibrationDelta	Indicates the calibration delta between elements of the array.

16	Integer	4	CalibrationElement	Indicates the element in the array which has a calibration value of CalibrationOffset.
20	Integer	2	DataType	Indicates the type of data stored at each element of the array. May be one of the following values: 1 – Unsigned 1-byte integer 2 – Unsigned 2-byte integer 3 – Unsigned 4-byte integer 4 – Signed 1-byte integer 5 – Signed 2-byte integer 6 – Signed 4-byte integer 7 – 4-byte float 8 – 8-byte float 9 – 8-byte complex 10 – 16-byte complex
22	Integer	4	ArrayLength	Indicates the number of elements in the array.
26	*	*	Data	The actual data values.

2D Data

For 2-D data (DataTypeID = 0x4122), the data is stored in a structure of the following form:

Byte Offset	Data Type	Data Size (Bytes)	Name	Description
0	Float	8	CalibrationOffsetX	Indicates the calibration value at element CalibrationElement in the X-direction.
8	Float	8	CalibrationDeltaX	Indicates the calibration delta between elements of the array in the X-direction.
16	Integer	4	CalibrationElementX	Indicates the element in the array in the X-direction which has a calibration value of CalibrationOffset.
20	Float	8	CalibrationOffsetY	Indicates the calibration value at element CalibrationElement in the Y-direction.
28	Float	8	CalibrationDeltaY	Indicates the calibration delta between elements of the array in the Y-direction.
36	Integer	4	CalibrationElementY	Indicates the element in the array in the Y-direction which has a calibration value of CalibrationOffset.
40	Integer	2	DataType	Indicates the type of data stored at each element of the array. May be one of the following values: 1 – Unsigned 1-byte integer 2 – Unsigned 2-byte integer 3 – Unsigned 4-byte integer 4 – Signed 1-byte integer 5 – Signed 2-byte integer 6 – Signed 4-byte integer 7 – 4-byte float 8 – 8-byte float 9 – 8-byte complex 10 – 16-byte complex
42	Integer	4	ArraySizeX	Indicates the number of elements in the array in the X-direction (the array width).
46	Integer	4	ArraySizeY	Indicates the number of elements in

				the array in the Y-direction (the array height).
50	*	*	Data	The actual data values.

6. Data Tag Format

The tags are stored in one of the following formats:

Time-only

For time-only tags (TagTypeID = 0x4152) the tag is stored in a structure of the following form:

Byte Offset	Data Type	Data Size (Bytes)	Name	Description
0	Integer	2	TagTypeID	Indicates the type of the tag – should have value 0x4152
2	Float	4	Time	Indicates the time in ANSI-standard time as the number of seconds elapsed since Jan. 1, 1970.

Time and position

For time and position tags (TagTypeID = 0x4142) the tag is stored in a structure of the following form:

Byte Offset	Data Type	Data Size (Bytes)	Name	Description
0	Integer	2	TagTypeID	Indicates the type of the tag – should have value 0x4142
2	Float	4	Time	Indicates the time in ANSI-standard time as the number of seconds elapsed since Jan. 1, 1970.
6	Float	8	PositionX	Indicates the position in the X-direction of the tag.
14	Float	8	PositionY	Indicates the position in the Y-direction of the tag.

Reading the File

Reading from an ESD file generally consists of the following steps:

1. Check the SeriesID and SeriesVersion words in the header to confirm that the file is a Series Data File of the appropriate version.
2. Read the total number of elements and the relevant information from the Dimension Array in order to determine the size and calibration of the series.
3. Read the OffsetArrayOffset word from the header to determine the location of the Data and Tag Offset Arrays.
4. Read in the Data and/or Tag Offset Arrays.

Read in the actual data and/or tags using the offsets in the Offset Arrays.