

Filling Out the Table Delimited Data Structure

The `<Table_Delimited>` class contains the information needed to parse a string of character bytes in a delimited table format into a table structure in programmatic memory. It is similar to the `<Table_Character>`, the principal difference being on the I/O end of things - `Table_Character`, as a fixed-width table, could potentially allow users to directly access individual rows and fields; `Table_Delimited` allows only serial access to the rows and fields.

N.B.: *PDS has defined a standard comprising parsing rules for extracting tabular data from a text file with delimited fields and records, identified as "PDS DSV 1" in labels. That standard defines the I/O processing for the `<Table_Delimited>` data described below.*

For additional explanation, see the *PDS4 Standards Reference*, or contact your PDS node consultants.

Following are the attributes and subclasses you'll find in `<Table_Delimited>`, in label order.

Note that in the PDS4 master schema, all classes have capitalized names; attributes never do.

`<name>`

OPTIONAL

If you'd like to give your table a name, do it here.

`<local_identifier>`

OPTIONAL

If you need to reference this `<Table_Delimited>` from elsewhere in the same label, give it an identifier here. If the identifier uses the same syntax as an average variable name in a typical programming language, you should be OK syntactically.

`<md5_checksum>`

OPTIONAL

Use this attribute to provide the MD5 checksum of the object *only*. If the object occupies the entire file, then the checksum should be given as an attribute of the `<File>` object. This checksum should be calculated using *only* the bytes defined as being part of this table.

`<offset>`

REQUIRED

This is the offset, in bytes, from the beginning of the file to the beginning of the `Table_Delimited` data. Offsets begin at zero. You must indicate a unit of bytes for this attribute:

```
<offset unit="byte">1234567890</offset>
```

<object_length>

OPTIONAL

If you know the total length of the *Table_Delimited* data, including all delimiters, line break characters and filler space, you can list it here. You must include a unit of bytes for this value. For example:

```
<object_length unit="byte">10240</object_length>
```

<parsing_standard_id>

REQUIRED

This attribute must have the standard value **PDS DSV 1**.

<description>

OPTIONAL

This is a free-format text field for any additional comments you might care to include at this point.

<records>

REQUIRED

This attribute must contain the total number of records in the *Table_Delimited* data.

<record_delimiter>

REQUIRED

This attribute must contain the standard value **Carriage-Return Line-Feed**. Note that the data must have carriage-return and line-feed delimited records.

<field_delimiter>

REQUIRED

This must have one of the standard values:

- **Comma**
- **Horizontal Tab**
- **Semicolon**
- **Vertical Bar**

Note that the parsing rules standard will contain additional information about leading and trailing blanks in the record, possible additional delimiters around text fields, the significance of leading and trailing space in a field, and so on.

<Uniformly_Sampled>

OPTIONAL

If this *Table_Delimited* contains records which are uniformly spaced in some dimension (time, wavelength, distance, etc.), you can use this class to define that dimension and interval rather than including an additional field in each row to hold the value explicitly. The details are on the [Filling Out the Uniformly Sampled Class](#) page.

<Record_Delimited>

REQUIRED

This class defines the repeating series of fields contained in one complete record of the *Table_Delimited* data.

<fields>

REQUIRED

The number of *Field_Delimited* classes directly under (that is, in the first nesting level of) the *Record_Delimited* class. Do **not** count *Field_Delimited* classes nested under *Group_Field_Delimited* classes.

If your *Record_Delimited* contains only one or more *Group_Field_Delimited* classes, this will have a value of zero.

<groups>

REQUIRED

The number of *Group_Field_Delimited* classes directly under (that is, in the first nesting level of) the *Record_Delimited* class. Do **not** count *Group_Field_Delimited* classes nested under other *Group_Field_Delimited* classes.

If your *Record_Delimited* contains only one or more *Field_Delimited* classes, this will have a value of zero.

<maximum_record_length>

OPTIONAL

The length of the longest single record in the *Table_Delimited* data, including all fields, all repetitions of group fields, any space between fields, and the record delimiters. You must specify a unit of bytes for this value:

```
<record_length unit="byte">1234</record_length>
```

A Note about Fields and Group Fields

Records are composed of *Fields* and *Group Fields*. A *Record* must have at least one of those (either will do), and can have an arbitrary number of them, in any order (that is, you can have *Fields* and *Group Fields* interspersed). Note, however, that *Group Fields* are **never** necessary - they are a notational convenience to save writing out large numbers of essentially identical *Field* definitions.

<Field_Delimited>

The class defines a single scalar field.

<name>

REQUIRED

The name of the field. PDS recommends that this be something fairly human-readable that can be easily turned into a variable name for use in applications, or displayed as a meaningful column heading.

`<field_number>`

OPTIONAL

This is the sequential number of the *Field_Delimited* definition. For PDS data products, the *field_number* is intended to be a help to human readers trying to map field definitions to columns in a print-out of the *Table*.

The *Standards Reference* lays out rules for using the *field_number* in cases where there are *Group_Field_Delimiteds* present which can be useful in programmatic contexts, but not so much in the visual-inspection case.

`<data_type>`

REQUIRED

The type of the values in the field. This must be one of the values listed in the [Standard Values Quick Reference](#).

`<maximum_field_length>`

OPTIONAL

The greatest number of bytes in the longest instance of this field in the delimited table. You must specify the unit:

```
<maximum_field_length unit="byte">12</maximum_field_length>
```

Note: This value is slightly ambiguous within the context of a scalar field within a `<Group_Field_Delimited>` class. It should be the maximum length of a single scalar value, not the maximum length of all the repetitions of the value within a group.

`<field_format>`

OPTIONAL

The value of this attribute is a string representing the read/print format for the data in the field, using a subset of the POSIX print conventions defined in the *Standards Reference*, and also described on the [PDS4 Field Format Conventions](#) page.

`<unit>`

OPTIONAL

If the value in this field has an associated unit, this is where it goes. This value is case sensitive, and you may use characters from the UTF-8 character set (like the Angstrom symbol) where appropriate.

Note: If a field contains a unitless value, then there should be no `<unit>` attribute.

NEVER include a null *unit* value, or even worse, this: `<unit>N/A</unit>`.

`<scaling_factor>`

OPTIONAL

If the data in this field are scaled, this attribute should contain the value by which the data must be multiplied to get back to the original value. Scaling factors are applied prior to adding any offset.

<value_offset>

OPTIONAL

If the values in the field have been shifted by an offset, this attribute should contain the value that must be added to each field value to get back to the original value. Offsets are added *after* the scaling factor, if any.

<description>

OPTIONAL

Free-format text describing the content of the field.

Note: While not required, SBN expects to see a useful definition for every *Field*, as do both reviewers and users. Omit this attribute at your peril.

<Special_Constants>

OPTIONAL

This class defines flag values used to indicate that a particular field value is unknown for one reason or another. It is identical to the <Special_Constants> class used in the *Array* classes. For details, check the [Filling Out the Array 2D Data Structure](#) page. Here is a quick list of the special constants available in this class:

- *saturated_constant*
- *missing_constant*
- *error_constant*
- *invalid_constant*
- *unknown_constant*
- *not_applicable_constant*
- *valid_maximum*
- *high_instrument_saturation*
- *high_representation_saturation*
- *valid_minimum*
- *low_instrument_saturation*
- *low_representation_saturation*

<Field_Statistics>

OPTIONAL

If you want to include things like extrema, mean value, and such for all the values that occur in this field through all the records in the table, this is the place to do it. This class is identical for all *Field* types. For details, see [Filling Out the Field Statistics Class](#). Here is a quick list of the field statistics available in this class:

- *maximum*
- *minimum*
- *mean*
- *standard_deviation*
- *median*

<Group_Field_Delimited>

This class defines a set of *Field_Delimited* and nested *Group_Field_Delimited* classes that repeats a given number of times in each record.

<name>

OPTIONAL

If you'd like to give your group a name, this is the place to do it. Names are often useful for helping users quickly understand what relationship the repeating fields have with each other.

<group_number>

OPTIONAL

Analogous to *field_number* for scalar fields, this is a sequential number useful for referencing *Group_Field_Delimited* classes at a single nesting level of a complex *Record_Delimited* definition.

<repetitions>

REQUIRED

The number of times the complete set of *Field_Delimiteds* and *Group_Field_Delimiteds* comprising this <Group_Field_Character> repeats.

<fields>

REQUIRED

The count of *Field_Delimited* classes directly under (i.e., at the first nesting level below) the *Group_Field_Delimited* definition. This will be zero if the group contains no *Field_Delimited* classes.

<groups>

REQUIRED

The count of *Group_Field_Delimited* classes directly under (i.e., at the first nesting level below) the present *Group_Field_Delimited* definition. This will be zero if the group contains no nested *Group_Field_Delimited* classes.

<description>

OPTIONAL

This free-format text field is available to provide additional text about why this group exists or what it represents.

Fields and Nested Groups

As in the *Record_Delimited*, the *Group_Field_Delimited* may contain either *Field_Delimited* classes, or *Group_Field_Delimited* classes, or both intermixed. *Group_Field_Delimited* classes may be nested arbitrarily deeply. The requirements for these data structure classes inside a <Group_Field_Delimited> are identical to those above.