Chapter 13. PDS Objects / Groups

The Planetary Data System has designed a set of standard Objects and Groups to be used for submitting catalog object information as well as for labeling data products. These standard Objects and Groups, along with definitions of individual keywords comprising those Objects and Groups, are defined in the *Planetary Science Data Dictionary*. In addition, Object and Group definitions and examples are also included in Appendix A and Appendix B of this document.

13.1 Generic and Specific Data Object Definitions

For each type of data object that PDS has defined (i.e., IMAGE, TABLE, etc.), there are two categories of definitions: generic and specific. A *generic object definition* is the universal definition of an object, or superset of keywords that can be used. A *specific object definition* is a subset of keywords used for a particular data product to allow effective use of validation tools.

Generic object definitions are designed and approved by the Planetary Data System, and defined in the *Planetary Science Data Dictionary*. Each object definition lists the elements and sub-objects required to be present each time the object is used in a product label. The dictionary definition also provides a list of additional, optional keywords that are frequently used by data preparers. Finally, note that any element defined in the PSDD may be included as an optional element in any object definition, at the discretion of the data preparer.

A specific object definition is defined for a particular data product and is based on a single generic object. The data preparer, in consultation with a data engineer, combines all the required elements of that object with a set of optional elements selected for their relevance to the data at hand. The result is a specific object definition. This definition is subject to approval during a design review.

The following examples illustrate the evolution from the generic IMAGE object to a specific IMAGE object, followed by an instance of that specific IMAGE. Note that when a specific object definition is created and used, the usage must be consistent for all labels using that object.

```
OBJECT                      = GENERIC_OBJECT_DEFINITION
NAME                        = IMAGE
STATUS_TYPE                 = APPROVED
STATUS_NOTE                 = "V2.1  1991-01-20  MDM  New Data Object Definition"
DESCRIPTION                 = "An image object is a regular array of sample
values.  Image objects are normally processed with special display tools to
produce a visual representation of the sample values. This is done by assigning
brightness levels or display colors to the various sample values. Images are
composed of LINES   and SAMPLES.  They may contain multiple bands, in one of
several storage orders.

Note:  Additional engineering values may be prepended or appended to each LINE
of an image, and are stored as concatenated TABLE objects, which must be named
LINE_PREFIX and LINE_SUFFIX.  IMAGE objects may be associated with other
objects, including HISTOGRAMs, PALETTEs, HISTORYs and TABLEs which contain
statistics, display parameters, engineering values or other ancillary data."

SOURCE_NAME                 = "PDS CN/M.MARTIN"
REQUIRED_ELEMENT_SET          =                                {LINE_SAMPLES,
```

```
      LINES, SAMPLE_BITS,
                                 SAMPLE_TYPE}
      OPTIONAL_ELEMENT_SET       =                                {BAND_SEQUENCE,
      BAND_STORAGE_TYPE,
                                 BANDS, CHECKSUM, DERIVED_MAXIMUM,
                                 DERIVED_MINIMUM, DESCRIPTION,
                                 ENCODING_TYPE, FIRST_LINE,
                                 FIRST_LINE_SAMPLE, INVALID,
                                 LINE_PREFIX_BYTES, LINE_SUFFIX_BYTES, MISSING,
                                 OFFSET, SAMPLE_BIT_MASK, SAMPLING_FACTOR,
                                 SCALING_FACTOR, SOURCE_FILE_NAME,
                                 SOURCE_LINES,  SOURCE_LINE_SAMPLES,
                                 SOURCE_SAMPLE_BITS, STRETCHED_FLAG,
                                 STRETCH_MAXIMUM, STRETCH_MINIMUM, PSDD}
      REQUIRED_OBJECT_SET       = "N/A"
      OPTIONAL_OBJECT_SET       = "N/A"

      OBJECT_CLASSIFICATION_TYPE = STRUCTURE

      OBJECT                    = ALIAS
      NAME                      = "N/A"
      USAGE_NOTE                = "N/A"
      END_OBJECT                = ALIAS

      END_OBJECT                = GENERIC_OBJECT_DEFINITION
```

This next example illustrates an IMAGE object definition being used for a specific case.

```
      OBJECT                    = SPECIFIC_OBJECT_DEFINITION
      NAME                      = XYZ_IMAGE
      STATUS_TYPE               = APPROVED
      STATUS_NOTE               = "V2.1 1991-02-10  TMA New specific data object
                                   definition"
      DESCRIPTION               = "The XYZ image is..."

      SOURCE_NAME               = "PDS CN/M.MARTIN"
      REQUIRED_ELEMENT_SET      = {LINE_SAMPLES, LINES, SAMPLE_BITS,
                                   SAMPLE_TYPE, SAMPLING_FACTOR,
                                   SOURCE_FILE_NAME,
                                   SOURCE_LINES,  SOURCE_LINE_SAMPLES,
                                   SOURCE_SAMPLE_BITS, FIRST_LINE,
                                   FIRST_LINE_SAMPLE}

      OBJECT_CLASSIFICATION_TYPE = STRUCTURE

      OBJECT                    = ALIAS
      NAME                      = "N/A"
      USAGE_NOTE                = "N/A"
      END_OBJECT                = ALIAS

      END_OBJECT                = SPECIFIC_ OBJECT_DEFINITION
```

13.1.1 Primitive Objects

Generic objects have a subclass called primitive objects that includes the ARRAY, COLLECTION, ELEMENT, and BIT_ELEMENT objects. The primitive objects are used as the building blocks for describing very irregular data that cannot be accommodated by any other

generic object. If at all possible, standard, well-supported generic objects (such as TABLE and IMAGE) should be used to describe archival data.

13.2 Generic and Specific Data Group Definitions

For each type of data Group that PDS has defined (i.e., PARAMETERS, etc.), there are two categories of definitions: generic and specific. A *generic group definition* is the universal definition of a group, or superset of keywords that can be used. A *specific group definition* is a subset of keywords used for a particular data product to allow effective use of validation tools.

As with OBJECTs (see PDS Standards Reference, section 13.1), there are two categories of GROUPs, generic and specific. The generic GROUP is the universal definition of the GROUP, specified in an appendix of the Standards Reference. The specific GROUP is an implementation of the generic GROUP for a particular data set. Shown below is a generic GROUP definition, and then an example of an instance of that GROUP in a data product.

```
OBJECT                      = GENERIC_GROUP_DEFINITION
NAME                        = CAMERA_MODEL
STATUS_TYPE                 = PENDING
STATUS_NOTE                 = "V1.0  2001-07-09 EDR  New Group Definition"
DESCRIPTION                 = "A camera model group is a collection of parameters
necessary to fully describe the geometric characteristics of a camera system."

SOURCE_NAME                 = "PDS IMG/E. RYE"
REQUIRED_ELEMENT_SET          =                              {CAMERA_MODEL_NAME,
                            CAMERA_MODEL_TYPE,
                               CAMERA_MODEL_DESC, CALIBRATION_SOURCE_ID,
                               GEOMETRY_SOURCE_ID, COORDINATE_SYSTEM_NAME,
                               MODEL_COMPONENT_ID, MODEL_COMPONENT_NAME,
                               MODEL_COMPONENT_UNIT_ID}
OPTIONAL_ELEMENT_SET =        {MODEL_COMPONENT_1_VECTOR,
                               MODEL_COMPONENT_2_VECTOR,
                               MODEL_COMPONENT_3_VECTOR,
                               MODEL_COMPONENT_4_VECTOR,
                               MODEL_COMPONENT_5_VECTOR,
                               MODEL_COMPONENT_6_VECTOR,
                               MODEL_COMPONENT_7_VECTOR, PSDD}

OBJECT                      = ALIAS
NAME                        = "N/A"
USAGE_NOTE                  = "N/A"
END_OBJECT                  = ALIAS

END_OBJECT                  = GENERIC_GROUP_DEFINITION
```

An example of using a GROUP follows:

```
GROUP                       = CAMERA_MODEL
CAMERA_MODEL_NAME           = "MIPS-0"
CAMERA_MODEL_TYPE           = "CAHV"
^CAMERA_MODEL_DESC          = "CAHV.ASC"
CALIBRATION_SOURCE_ID       = "UOFA-BACKLASH"
GEOMETRY_SOURCE_ID          = "TELEMETRY"
COORDINATE_SYSTEM_NAME      = "IMP-CAMERA"
MODEL_COMPONENT_ID          = (C, A, H, V)
MODEL_COMPONENT_NAME        = ("CENTER", "AXIS",
                               "HORIZONTAL", "VERTICAL")
```

```
MODEL_COMPONENT_UNIT_ID    = ("m", "none", "pixel", "pixel")
MODEL_COMPONENT_1_VECTOR   = (3.469, 14.593, 8.937)
MODEL_COMPONENT_2_VECTOR   = (0.351, 0.758, 17.932)
MODEL_COMPONENT_3_VECTOR   = (14.020, 15.336, 23.714)
MODEL_COMPONENT_4_VECTOR   = (27.423, 3.719, 16.426)
END_OBJECT                 = CAMERA_MODEL
```

In order to facilitate the inclusion of multiple instances of keywords within data product labels without requiring a whole host of new GROUPs, there is a special GROUP called the PARAMETERS GROUP. It has no required elements, and the set of all elements in the PSDD as its optional element set.

```
OBJECT                     = GENERIC_GROUP_DEFINITION
NAME                       = PARAMETERS
STATUS_TYPE                = PENDING
STATUS_NOTE                = "V1.0  2001-07-09 EDR  New Group Definition"
DESCRIPTION                = "The parameters group provides a mechanism for
                             Grouping multiple sets of related parameters
                             within a data product label."

SOURCE_NAME                = "PDS IMG/E. RYE"
REQUIRED_ELEMENT_SET         =                                {}
OPTIONAL_ELEMENT_SET =       {PSDD}

OBJECT                     = ALIAS
NAME                       = "N/A"
USAGE_NOTE                 = "N/A"
END_OBJECT                 = ALIAS

END_OBJECT                 = GENERIC_GROUP_DEFINITION
```

For example:

```
GROUP                      = COMMANDED_INST_PARAMETERS
 SHUTTER_MODE              = "BOTSIM"
 FILTER_NUMBER             = 5
 FILTER_NAME               = "L570-R570"
 EXPOSURE_DURATION         = 1.05
END_OBJECT                 = COMMANDED_INST_PARAMETERS

GROUP                      = TELEMETRY_INST_PARAMETERS
 SHUTTER_MODE              = "AUTO"
 FILTER_NUMBER             = 0
 FILTER_NAME               = "CLEAR"
 EXPOSURE_DURATION         = 0.773
END_OBJECT                 = TELEMETRY_INST_PARAMETERS
```

13.2.1 Implementation of Group Statements

PDS applies the following restrictions to the use of GROUPS:

1. The GROUP structure may only be used in a data product label which also contains one or more data OBJECT definitions.
2. The GROUP statement must contain only attribute assignment statements, include pointers, or related information pointers (i.e., no data location pointers).
3. GROUP statements may not be nested.

4. GROUP statements may not contain OBJECT definitions.
5. Only PSDD elements may appear within a GROUP statement.
6. The keyword contents associated with a specific GROUP identifier (e.g., CAMERA_MODEL) must be identical across all labels of a single data set.

Usage of a GROUP structure must be coordinated with and approved by the responsible PDS discipline Node.

Descriptors may be pre-pended to any generic Group name to produce, and distinguish between, specific instances of the generic group (i.e., any generic Group name may be preceded with a qualifier to uniquely identify the specific instance of the generic Group). For example, the generic PARAMETERS Group could have specific instances of "A_PARAMETERS", "B_PARAMETERS", etc. Pre-pending a descriptor to the generic instances allows multiple instances of the Group to be repeated within a single label.

The specific GROUP is an implementation of the generic GROUP for a particular data set and must be consistent in its structure (i.e., use the same set of keywords) across the data set. For example, the PARAMETERS Group may consist of any keywords defined within the PSDD.

In the following examples, the TELEMETRY_GEOMETRY_PARAMETERS Group consists of three keywords and the CORRECTED_GEOMETRY_PARAMETERS Group consists of three keywords. In this case, both specific instances use the same keywords but could consist of different sets of keywords. Both instances can be collocated within a single data product label. But, each instance across the dataset must contain the same set of keywords.

```
GROUP                    = TELEMETRY_GEOMETRY_PARAMETERS
 GEOMETRY_SOURCE_ID      = "TELEMETRY"
 INSTRUMENT_AZIMUTH      = 35.6 <DEGREES>
 INSTRUMENT_ELEVATION    = -15.4 <DEGREES>
END_OBJECT               = TELEMETRY_GEOMETRY_PARAMETERS

GROUP                    = CORRECTED_GEOMETRY_PARAMETERS
 GEOMETRY_SOURCE_ID      = "MIPS_MPFMOS"
 INSTRUMENT_AZIMUTH      = 35.9 <DEGREES>
 INSTRUMENT_ELEVATION    = -15.5 <DEGREES>
END_OBJECT               = CORRECTED_GEOMETRY_PARAMETERS

GROUP                    = CORRECTED_GEOMETRY_PARAMETERS
 GEOMETRY_SOURCE_ID      = "UOFA-BACKLASH"
 INSTRUMENT_AZIMUTH      = 35.8 <DEGREES>
 INSTRUMENT_ELEVATION    = -15.6 <DEGREES>
END_OBJECT               = CORRECTED_GEOMETRY_PARAMETERS
```

In the near term, the only validation requirements for GROUPs will be that all the elements present in a GROUP must be present in the PDS Data Dictionary. In the future, it is hoped that the contents of the GROUPs will also be validated against their generic GROUP specifications. This would be to ascertain that all the required elements of a particular GROUP are present and that no elements are present that are not specified in the set of required and optional elements.

(This page intentionally left blank.)