Overview of Data Design and Discussion

PDS 2010 Tech Session

June 10, 2009

PDS 2010 Data Design WG

Purpose and Goals
Data Architecture Presentations

1. The tech staff will leave the meeting with a better understanding of the proposed PDS4 data model and its implementation options.

2. The tech staff will provide input to help the working group refine the data model and its options.

3. The tech staff will help identify key additional information that the MC will need in order to make informed decisions in August.

4. Tech reps will make recommendations to their managers on various options.

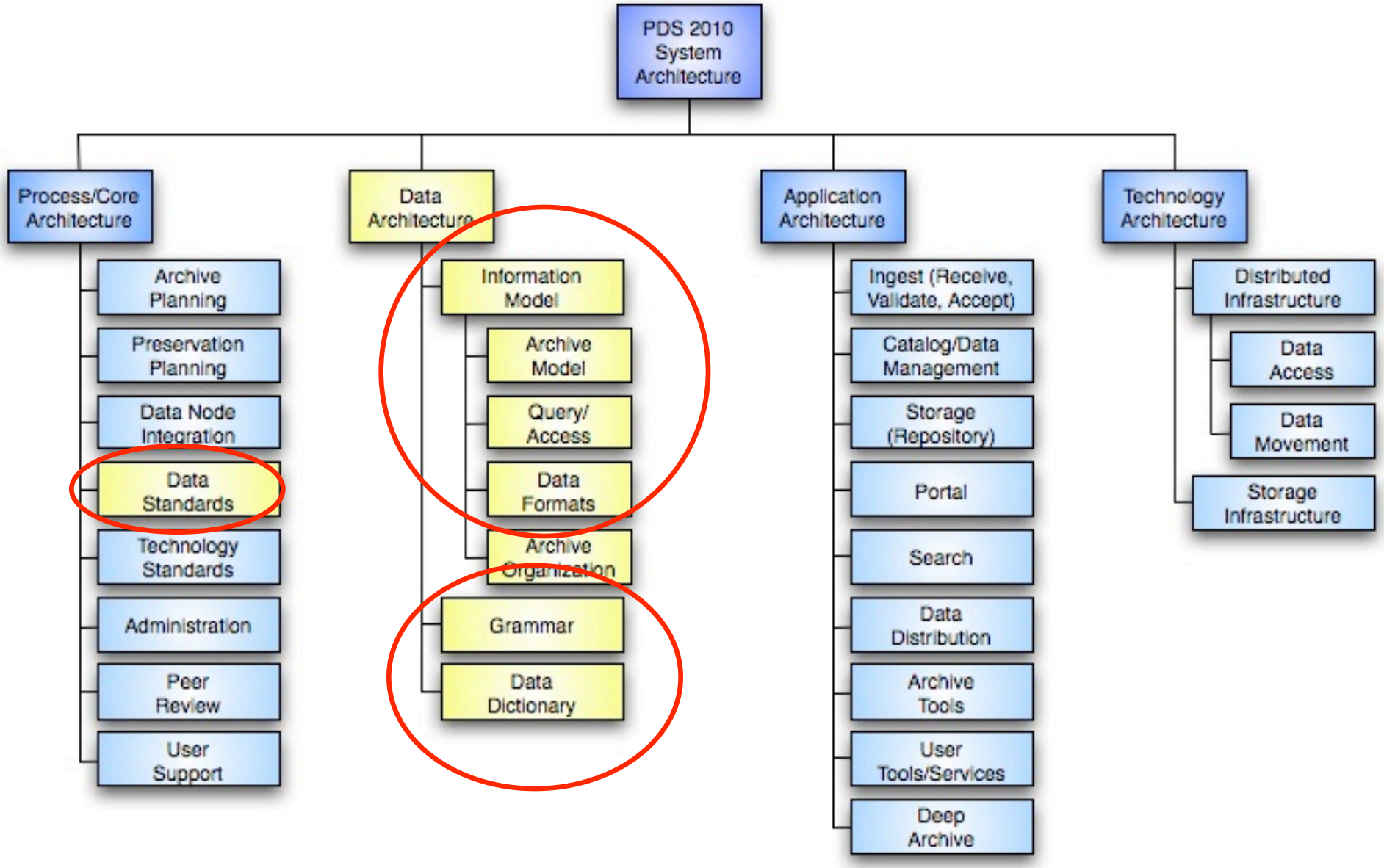
Purpose of this Presentation

- **Present an overview of the PDS4 Data Design task.**
  Design Principles, Goals, and Status.
- **Introduce the design processes.**
- **Present a high level view of the model.**
- **Introduce some issues.**

PDS4 Data Design Working Group

- **Formed in December 2008**
- **Personnel**
  - Steve Hughes (Lead)
  - Mitch Gordon (Rings)
  - Edward Guinness (Geosciences)
  - Lyle Huber (Atmospheres)
  - Ron Joyner (EN)
  - Anne Raugh (SBN)
  - Elizabeth Rye (Imaging, Standards)
  - Steve Joy (PPI)
  - Dick Simpson (Radio Science - Observer)
- **Weekly Telecons**
- **Project Website**
  - Just released version "t" of the model to the PDS engineering node website.
    - http://pds-engineering.jpl.nasa.gov/system_eng/PDS4_Data_Design/index.html
  - The design documents and label schemas are re-generated after each update to the data model database.

PDS 2010 Architecture



PDS4 Key Goals

- **Enable a stable and usable long-term archive.**
- **Enable more efficient archive preparation for data providers.**
- **Enable services for the data consumer to find the specific data they need and provide the formats they require.**

PDS4 Design Principles

- The data model:
  - is defined in a formal language
  - is independent of implementation
  - defines a few fundamental data structures that do not evolve over time
  - is extensible enabling it to handle more complex data formats

- The archive data formats shall be designed independent of data provider and data consumer formats.

- The data architecture shall include a standard data dictionary model.

DDWG Deliverables

- **PDS4 Information Model**
  - The Information Model defines PDS object classes. This includes data structures, formats, and products as well as data sets, documents, missions.

- **PDS4 Data Dictionary Model**
  - The Data Dictionary Model provides the schema for the PDS data dictionary. The data dictionary documents the data elements used in the PDS4 Information Model.

- **PDS Standards Reference V4.0**
  - The PDS Standards Reference V4.0 will be written in the format and tone of a standards reference document.

- **Grammar Options**
  - The Grammar is used to capture PDS archive metadata for product labels.

PDS4 Data Design Accomplishments - General

• Project and Project Members Defined • Principles and Drivers updated	**Done**
• General Data Model (Draft) • Product Data Model (Draft) • Data Dictionary Model (Draft) • Grammar Options • PDS Standards Reference (Outline)	**Substantial Progress**
• Data Dictionary Model (Final) • Grammar Decision • PDS Standards Reference (Draft)	**Next 3 Months**

- PDS and community wide review
- General Data Model (Final)
- Product Data Model (Final)
- PDS Standards Reference (Final)

Elapsed Time ~ 6 months – Additional duty for most WG members

PDS4 Data Design Accomplishments - Details

- **Four basic data structures**
  - Homogeneous N-dimensional array of scalars – Array_Base
  - Heterogeneous repeating record structure of scalars – Table_Base
  - Unencoded byte stream
  - Encoded byte stream

- **Array_Base**
  - Array_2D
    - Image_Grayscale
    - Spectrum_2D
  - Array_3D
    - Image_3D
    - Movie
- **Table_Base**
  - Table_Character
  - Table_Binary
  - Table_Binary_Grouped

- **Identifiable**
  - Digital Product
    - Data Product
      - Product_Image_Grayscale
      - …Image_3D
      - …Movie
      - …Table_Character
      - …Table_Binary
      - …Table_Binary_Grouped
    - Document Set
    - Software_Set
  - *Non-Digital Product*
    - *Mission*
    - *Instrument*
    - *Resource*

- **Data Type**
  - Binary Data Type
    - Decimal Integer
    - …
  - Character Literals
    - Character Integer
    - …

- **Draft PDS Data Dictionary Model**

- **Grammar – ODL+, PVL, and XML Labels**

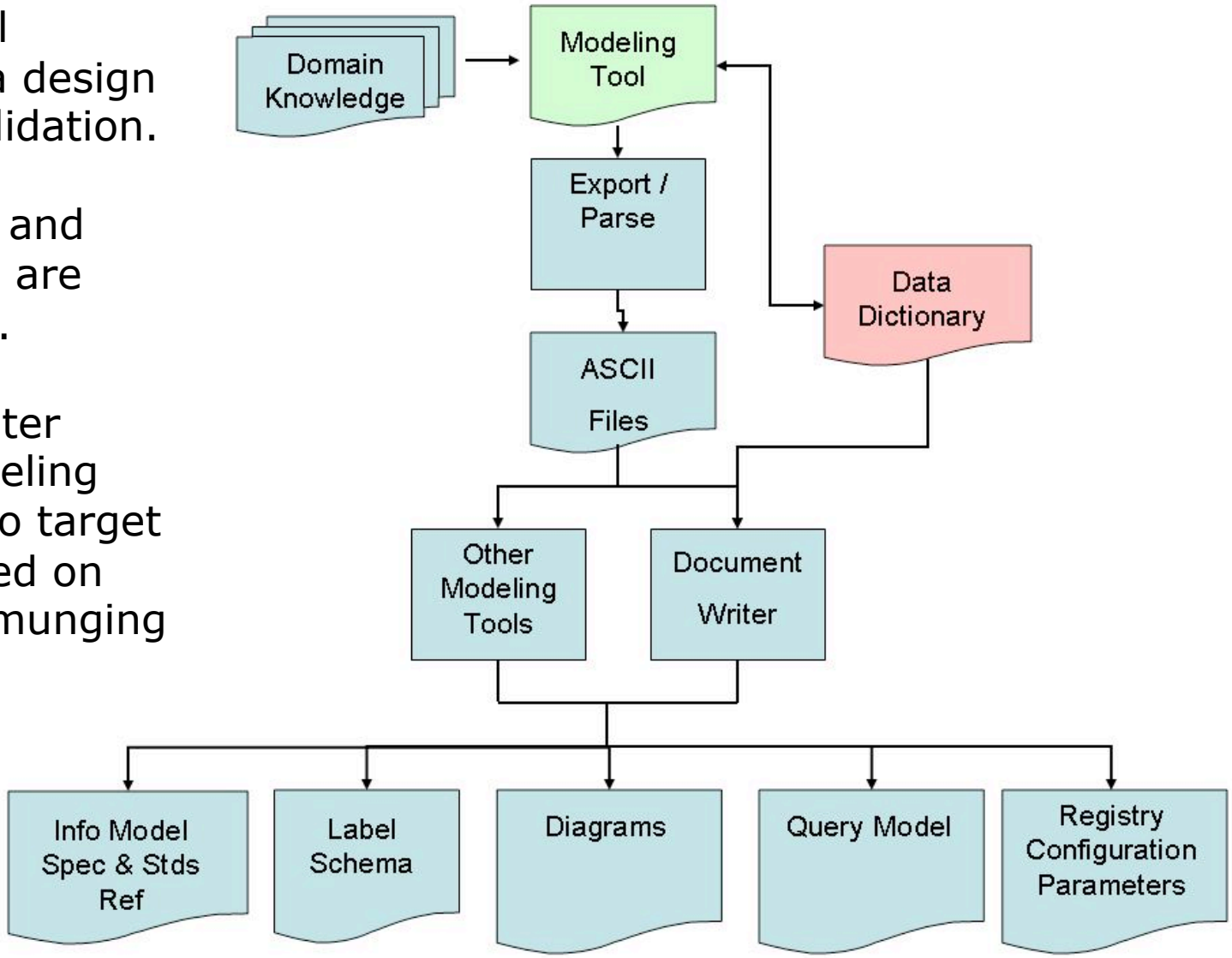
The Model Design Process

• Master model constrains data design and defines validation.

•Master model and data dictionary are tightly coupled.

•Document writer translates modeling information into target languages based on grammar and munging rules.

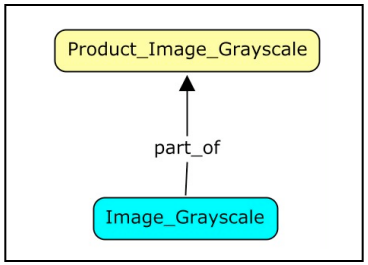
•Updates to master model are reflected quickly in the documents.



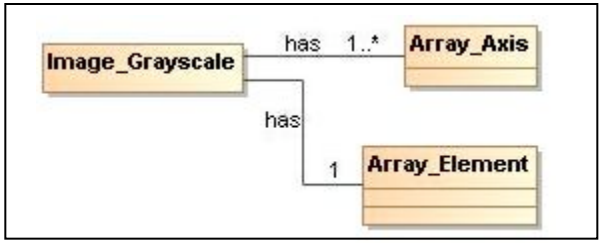
Example Results

Image _Grayscale

Concept Map



UML Class Diagram



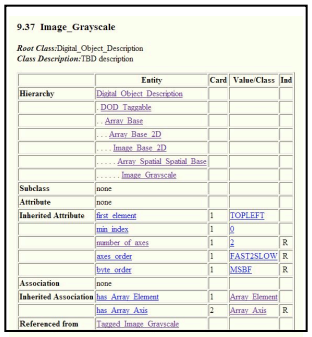
XML Schema

```
<!-- PDS4 XML/Schema for Product_Image_Grayscale -->
<xs:complexType name="Image_Grayscale_Type">
  <xs:sequence>
    <xs:element name="axes_order" type="axes_order_Type"
```

PVL Label Template

```
/* ******* Label Template - Product_Image_Grayscale
Object = Product_Image_Grayscale;
      Object = Image_Grayscale;
        local_identifier      = ${local_identifier};
```

Class Definition Table

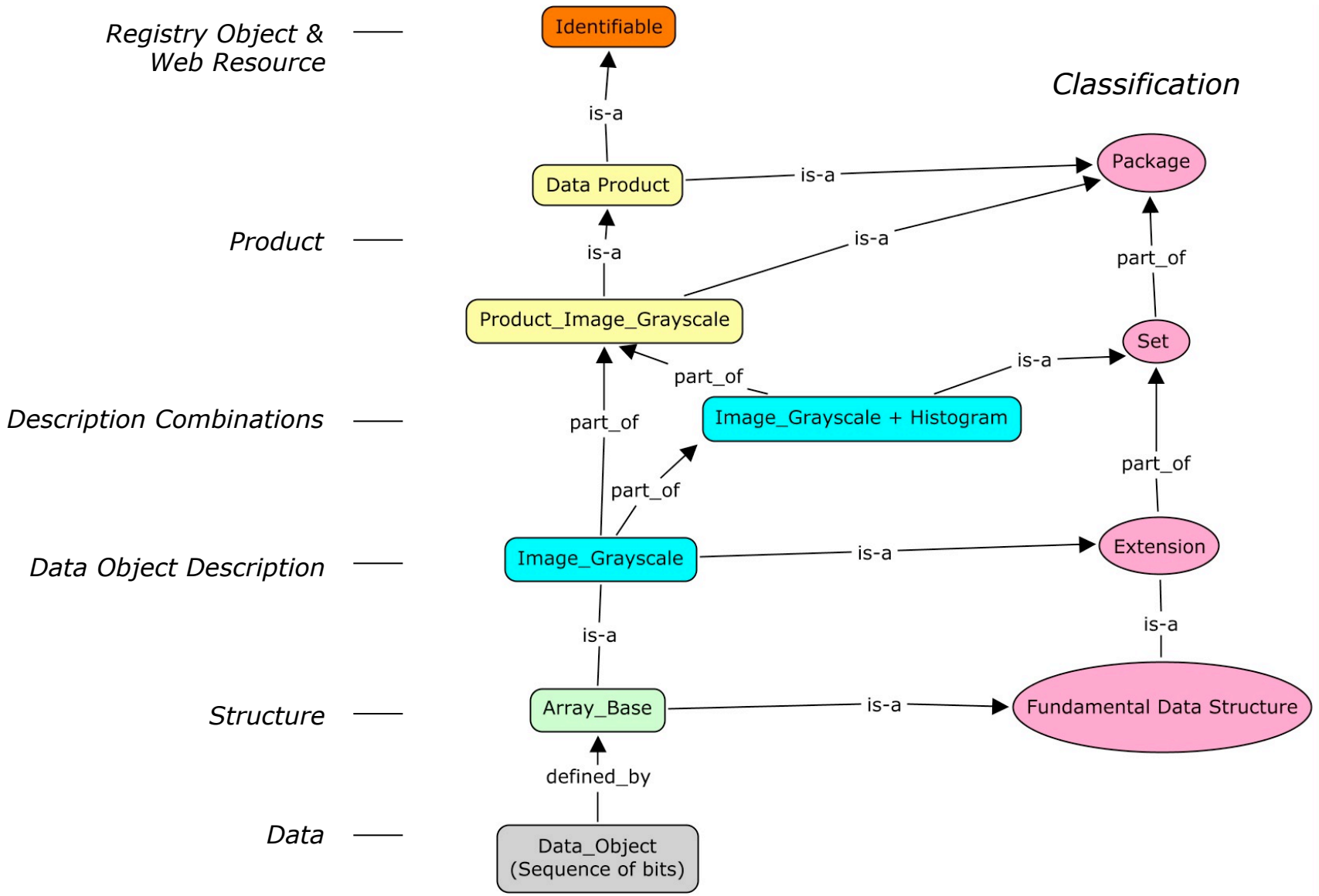


9.37 Image_Grayscale

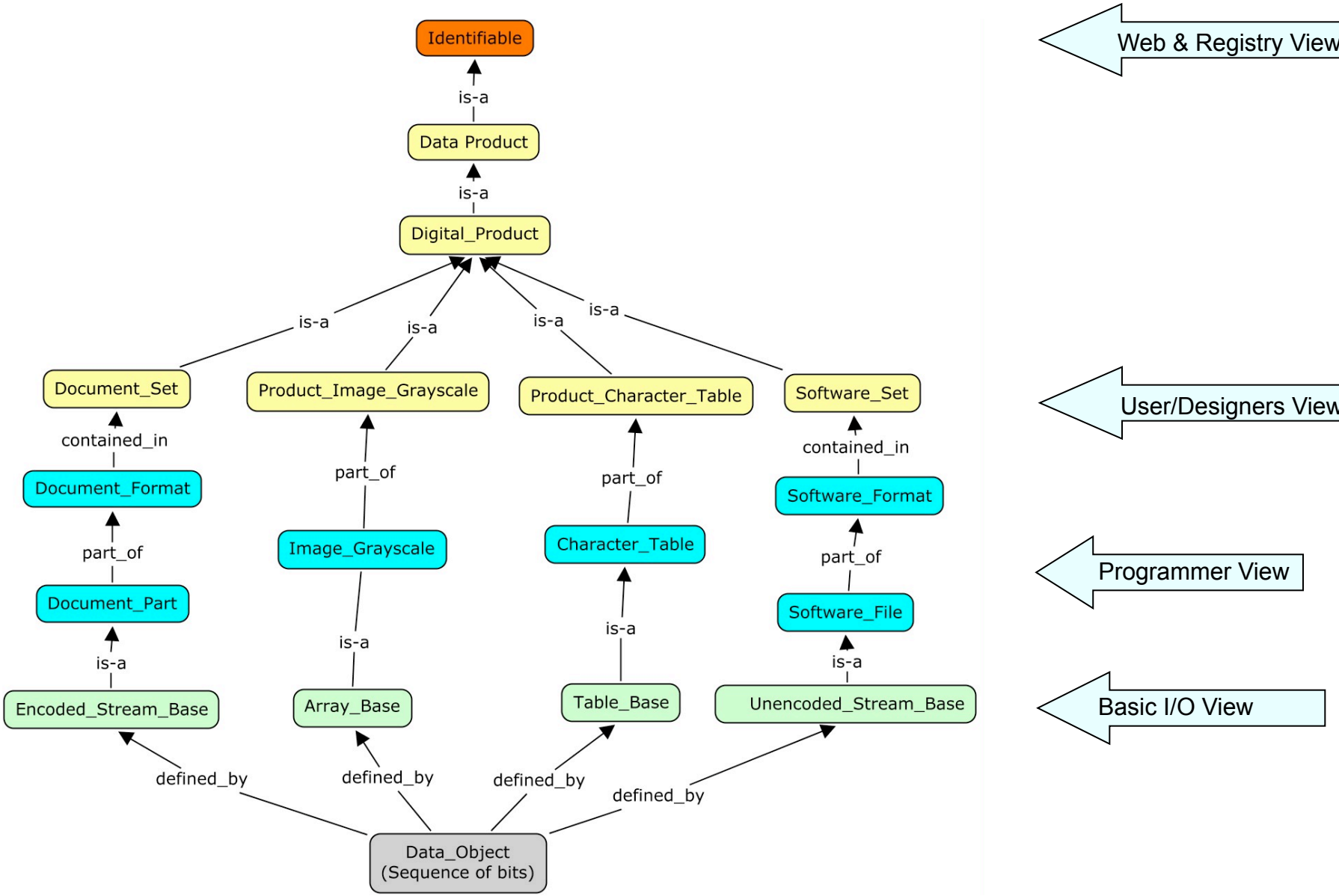
*Root Class:*Digital_Object_Description
*Class Description:*TBD description

	Entity	Card	Value/Class	Ind
Hierarchy	Digital_Object_Description			
	. DOD_Taggable			
	. . Array_Base			
	. . . Array_Base_2D			
	. . . . Image_Base_2D			
	. . . . . Array_Spatial_Spatial_Base			
	. . . . . . Image_Grayscale			
Subclass	none			
Attribute	none			
Inherited Attribute	first_element	1	TOPLEFT	
	min_index	1	0	
	number_of_axes	1	2	R
	axes_order	1	FAST2SLOW	R
	byte_order	1	MSBF	R
Association	none			
Inherited Association	has_Array_Element	1	Array_Element	
	has_Array_Axis	2	Array_Axis	R
Referenced from	Tagged_Image_Grayscale			

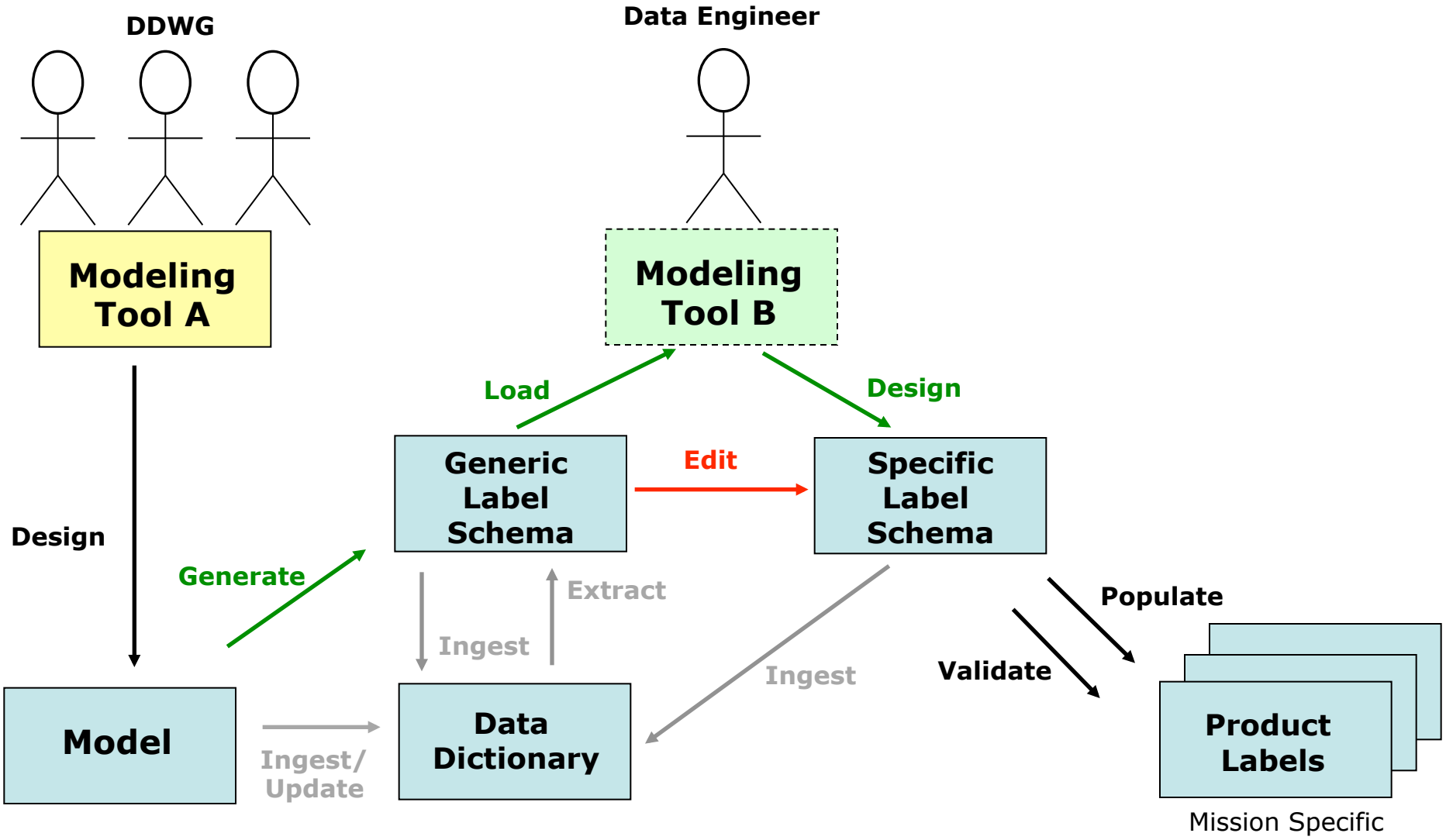
PDS4 Data Product Model Components



PDS4 Data Product Concept Map



PDS4 Product Label Creation



Current using editor
Current using Oxygen/Future Design Tool
Proposed

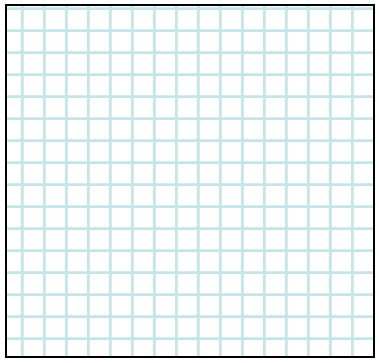
Issue - Ambiguity

- **A PDS3 Image object only requires the number of lines and line samples in a simple image.**
  - However the 2 dimensional structure becomes a 3 dimensional structure with the addition of the BANDS keyword.
  - A two dimensional structure is assumed by the omission of the BANDS keyword.

Image Description

```
OBJECT = IMAGE
  LINES = 800
  LINE_SAMPLES = 800
  SAMPLE_TYPE = UNSIGNED_INTEGER
  SAMPLE_BITS = 8
END_OBJECT = IMAGE
```

Simple Image

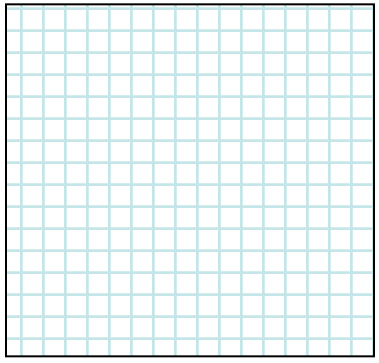


Problem
Data Structure is not Rigorously Defined

- **Current Image Description**
  - Where is the first logical pixel?
  - Are the pixels in row or column major order?
  - How many axes exist?

```
OBJECT = IMAGE
  LINES = 800
  LINE_SAMPLES = 800
  SAMPLE_TYPE = UNSIGNED_INTEGER
  SAMPLE_BITS = 8
END_OBJECT = IMAGE
```

Simple Image



Solution
Rigorously Define Data Structures

- **Array_Base Structure**
  - Where is the first logical pixel? **TOPLEFT**
  - Are the pixels in row or column major order? **FAST2SLOW**
  - How many axes exist? **<#axes>**

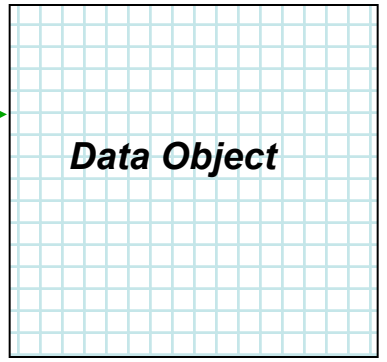
  - **Ambiguity cleared up.**

Abstract Data Object Description

```
Object                = Array_Base
    Data_Location     = (filename, 255);
    first_element     = TOPLEFT
    min_index         = 0
    number_of_axes    = #axes
    axes_order        = FAST2SLOW
    byte_order        = MSBF
    element_bytes             = 1
    element_type      = DECIMAL_INTEGER
    axis_length       = (#first, #second, …)
    axis_name         = (first, second, …)
End_Object            = Array_Base
```

Defines

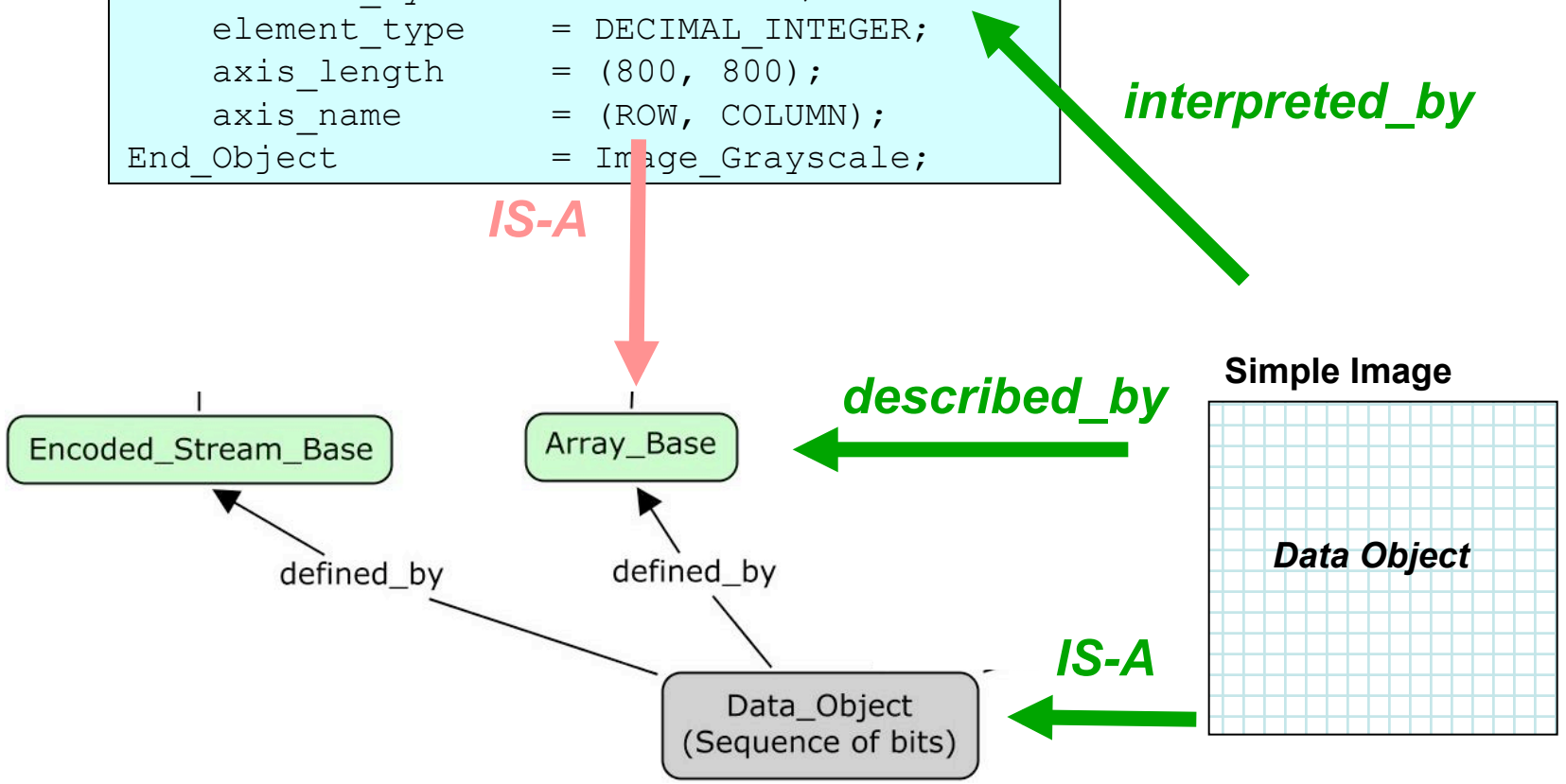
Simple Image



Extend a Data Structure

Data Object Description

```
Object                = Image_Grayscale;
    Data_Location     = (filename, 255);
    first_element     = TOPLEFT;
    min_index         = 0;
    number_of_axes    = 2;
    axes_order        = FAST2SLOW;
    byte_order        = MSBF;
    element_bytes             = 1;
    element_type      = DECIMAL_INTEGER;
    axis_length       = (800, 800);
    axis_name         = (ROW, COLUMN);
End_Object            = Image_Grayscale;
```



Combine Data Structures - Sets

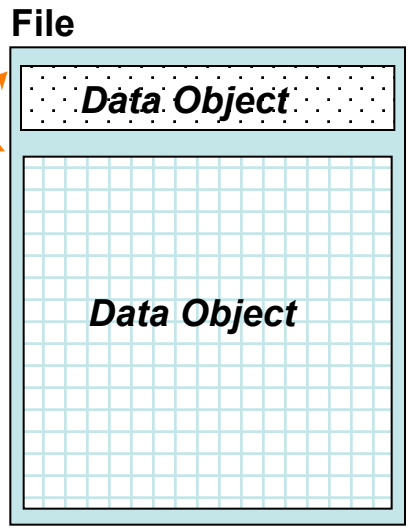
Data Object Descriptions

```
Object = Tagged_Image_Grayscale_Set;

  Object              = Image_Grayscale;
    Data_Location     = (filename, 255);
    first_element     = TOPLEFT;
    min_index         = 0;
    number_of_axes    = 2;
    axes_order        = FAST2SLOW;
    byte_order        = MSBF;
    element_bytes             = 1;
    element_type      = DECIMAL_INTEGER;
    axis_length       = (800, 800);
    axis_name         = (ROW, COLUMN);
  End_Object          = Image_Grayscale;

  Object = Header;
    Data_Location     = (filename, 0);
    comment           = "This Header…";
    bytes             = 255;
  End_Object = Header;

End_Object = Tagged_Image_Grayscale_Set;
```



Package a Product

Product

```
Object = Product_Image_Grayscale
```

Identification Metadata

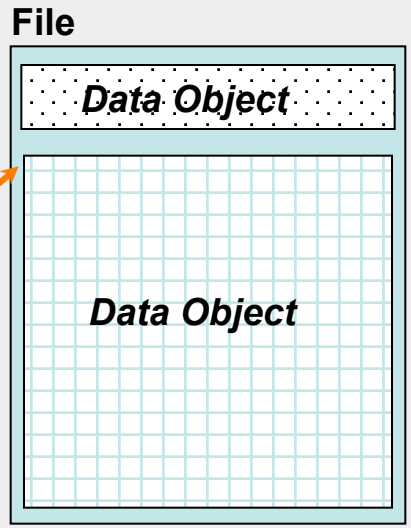
```
Object = Identification_Section;
  Identifier            = "PDS4_IMG_IMAGE_...
  URN                   = "http://URN:MPFL-M-IMP-2...
  Title                 = "MARS PATHFINDER LANDER ...
  Version               = "1.0";
  Label_Revision_Note   = "20090101:1.0 - initial ...
  DD_Version_Id         = "DD_Version_Id";
  PDS_Version_Id        = "PDS4.0";
  Product_Creation_Time = 1998-07-14T00:36:08.000;
End_Object = Identification_Section;
```

Descriptive Metadata

```
Object = Circumstances_Of_Observation_Section;
  Spacecraft_Clock_Start_Count = "1246943630";
    ...
```

Structural Metadata

```
Object = Tagged_Image_Grayscale_Set;
  Object              = Image_Grayscale;
    Data_Location     = (filename, 255);
    first_element     = TOPLEFT;
    ...
```



Query Model

- A formal model consisting of classes, attributes, and relations that are appropriate for use as search constraints
  - Types of query models include
    - Data Set
    - Product
      - Data Product
      - Document Product
      - Software Product
    - Any PDS4 Identifiable
- The query models are subsets of the archive model
  - Can be augmented with external metadata (e.g. LDD)
- Example query constraints
  - general parameters - time, target, mission, instrument host, instrument
  - any metadata defined in the archive model
  - any association between two classes
    - e.g. documents associated with data sets
  - class type and hierarchy
  - geometry - (lat, lon), (az, el), (ra, dec), (x,y,z).

Benefits of the PDS4 Data Model

- **The data model is managed in a data modeling tool.**
  - The model is formally defined.
  - The model can be validated and tested.
- **Defines a few simple fundamental data structures.**
  - Fundamental data structures may be extended and combined to form more complex data formats
- **The overall architecture is model driven.**
  - Disentangles the model from its implementation.
  - Model can evolve over time as research domain changes.
  - Drives the generation of documentation, label schema, and other model dependent artifacts.
- **The data dictionary uses a standard data dictionary model.**

Road Map for Remainder of Session
Data Design

- Remaining presentations with follow-up discussions and questions

  - Detailed PDS4 Data Model Discussion
    - Image_Grayscale – Elizabeth R.
    - Table_Character – Lyle H.
    - Table_Character_Grouped – Ed G.
    - Software – Anne R.

  - Data Transformations – Anne R.
  - Data Dictionary Discussion and Planning- Steve H.
  - Grammar Discussion – Paul R.
  - Standards Reference Plans – Elizabeth R.
  - Data Transformation Options and Discussion – Mitch G.

Backup