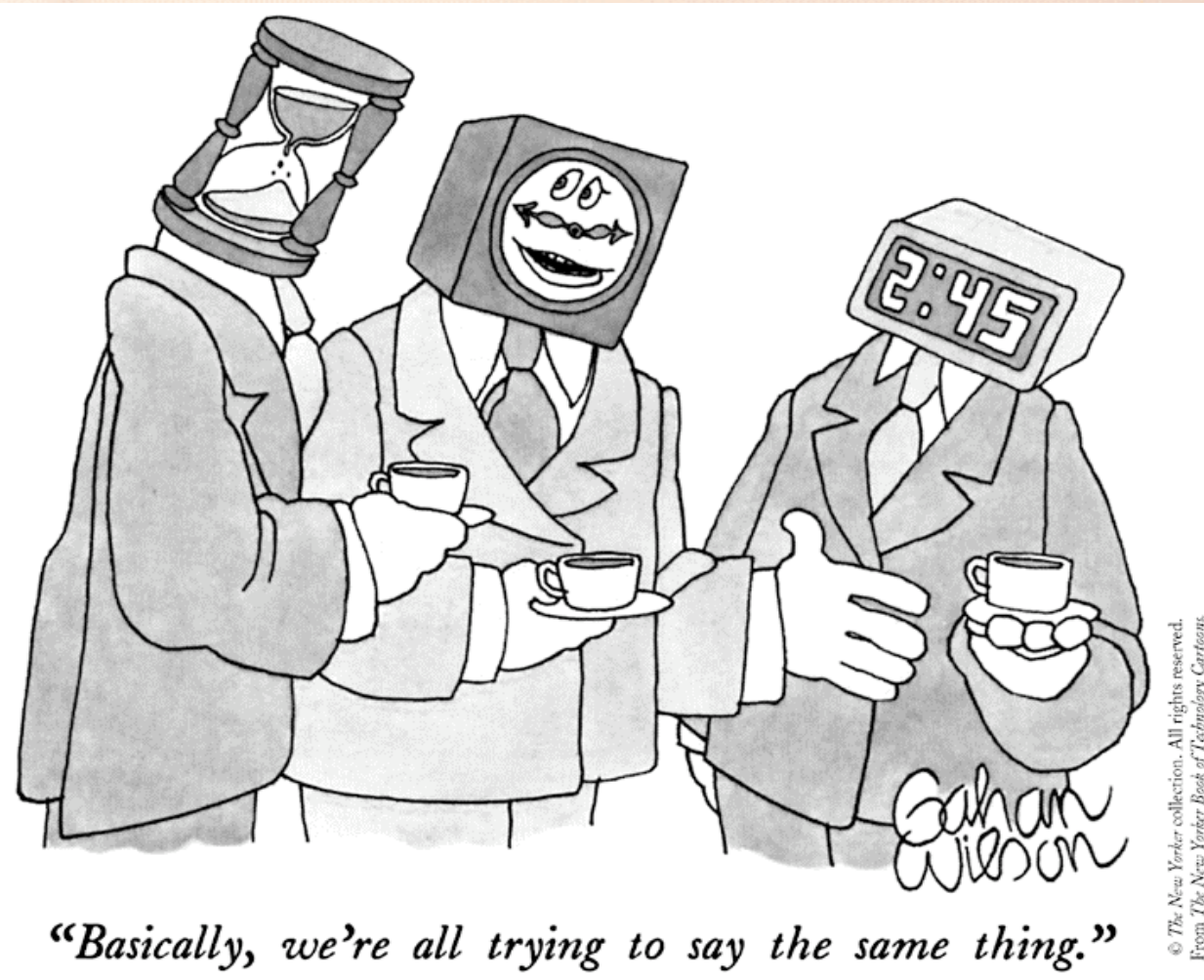


**The PDS4 Data
Model Working
Group
Activity and Status
Report**

**Presented to the PDS Management
Council, Dec 3-4 2007**

- In a nutshell....



As seen in: Ontology Mapping and Alignment, Natasha Noy, Stanford University

A Important Introduction

- We call ourselves the Data Model group but we're really the Information Model group because:

An information model is an abstract but formal representation of entities including their properties, relationships and the operations that can be performed on them.

- An information model does not constrain how a description is mapped to an actual implementation in software. There may be many mappings of the information model. Such mappings are called Data Models

A Common Goal

- We all shared the same objective:

Provide an improved Information Model to PDS

- We strived to be well informed:
 - We shared our ideas in open discussions
 - E-mail, Telecon, written reports and white papers.
 - We looked at other models
 - Dublin Core, IVOA, OAI-ORE, PDS3, SPASE, SWEET, VSTO.
 - We sought advice from the Management Council
 - Scope and Expectations

The Problems

There are identifiable problems with the PDS3 information model.

- Each time a new target, volume set, volume or data set is added to the system the data dictionary must be updated.
- Object definitions in the data model allow too many options.
- Data dictionary has grown so much that it is becoming difficult to use. (27 ways to say "Temperature", 20 for "Latitude")
- Products consisting of multiple files (Compound products) is very poorly supported.
- Object descriptions (metadata) lack terms to aid in search interfaces.
- Some terms have a specific intent, but have been obscured by lax usage.
- Non-functional document & software labels
- Catalog file hierarchy becoming less applicable to more datasets
- Missing things (objects): movies, SPICE, advanced products

Which leads to:

- Difficulty that providers have in designing PDS-compliant products.
- Difficulty that users have in reading and using PDS-compliant products.

The PDS Data Dictionary contains:

14,458 terms

1643 elements and 81 objects.

12,734 standard values (2,848 target names, 144 volume sets, 1,966 volumes and 1,370 data set IDs)

Level 1-3 Requirements

The PDS Information Model is a critical high level system component which part of fulfilling the following requirements:

- 1.4 Establishing archiving standards,
- 1.5 Providing archiving tools,
- 2.2 Validation,
- 2.3 Peer review,
- 3.1 Search,
- 3.2 Retrieval,
- 4.1 Preservation,
- 4.2 Usability.

Principals

- *Interoperability* – The PDS works to ensure interoperability among planetary science archives by seeking community consensus on a core set of common objects and data elements.
- *Partitioning* - The data model is logically separated into partitions in order to allow for management and evolution of components of the data model independently. For example, the image model is managed by the imaging community.
- *Formal Specification* - The data model is explicitly and unambiguously defined using a formal data engineering notation and/or language.
- *Standards* - PDS applies commonly accepted and documented standards that address fulfill its requirements.
- *Evolvability and Flexibility* – The data model should be extensible and flexible enough to meet new requirements.
- *Model Expressions* – The data model is implementation neutral and can have different expressions to support subsystem functions.

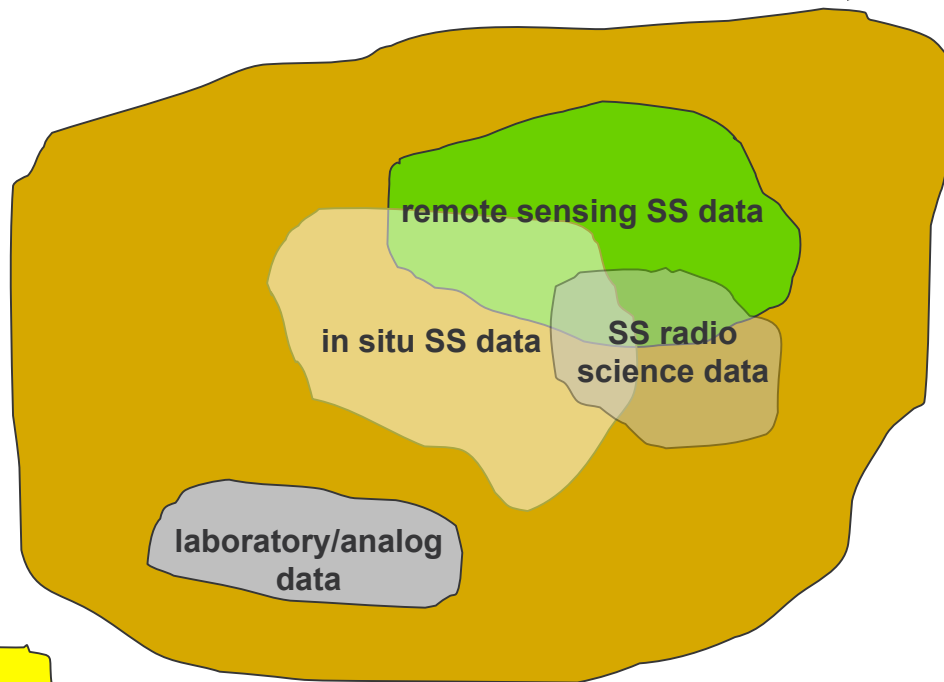
Information Modeling 101

1. Identify your domain
2. Determine the partitions of domain.
3. Identify the entities in each partition
 - Identify relationships of entities
4. Identify the attributes for each entity.

Planetary Domain

Data produced by or relevant to NASA's planetary missions, research programs, and data analysis programs.

-PDS Mission Statement 2006-03-02



What distinguishes the subjects within the boundary from those outside?

What distinguishes one data product from another inside the boundary?

What descriptions of data products are needed if they are to be used?

solar physics

Are these outliers part of PDS responsibility?

extra-SS data

linguistics

literature (novels)

Courtesy of Dick Simpson

Partitions and Entities (Conceptual Model)

- **Participants**

- Mission
- Observatory
- Instrument
 - Detector
- Person
- Reference
- Target

- **Product**

- Sample (Physical)
- Data Structure (Digital)
 - Catalog (record collection)
 - Table (row, column)
 - Image (x, y, z)
 - Movie (x, y, z, t)
 - n-Array
 - Compound Structure (?)
- Documents

- **Resource**

- Repository
- Registry
- Service
- Web Link

- **Collections**

- Dataset
- Event
- Campaign

- **Query**

Defining Attributes

Always

Adopt, adapt, develop (in that order)

So...

Look at PDS3 for a equivalent entity. Adopt if one is found:

1. Evaluate attributes, select appropriate
2. Transfer entities into PDS4 model
3. Define new entities as needed

Note: Looking to PDS3 first will help carry “corporate” knowledge forward.

Otherwise

1. Look to other models (adapt)
2. Define a new set of attributes (develop)

An Example

INSTRUMENT (PDS3)

- INSTRUMENT_HOST_ID
- INSTRUMENT_ID
- INSTRUMENT_INFORMATION
 - INSTRUMENT_DESC
 - INSTRUMENT_NAME
 - INSTRUMENT_TYPE
- INSTRUMENT_REFERENCE_INFO
 - REFERENCE_KEY_ID

INSTRUMENT (PDS4)

- OBSERVATORY_ID
- INSTANCE_ID
- DESC
- NAME
- INSTRUMENT_TYPE
- REFERENCE_ID

Note

REFERENCE_ID can have multiple occurrences.
INSTANCE_ID is a unique identifier for the instance.

Recommendations

M/C approve level 1 & 2 entities.

Vet with techs first?

Have the PDS4 Data Model Group:

- **Define the relationships of each entity.**
- **Define the attributes for all entities**
- **Vet complete model with a wider community.**

Then we have a base PDS4 Information Model.
(except for the archive format decisions)

Backup Material

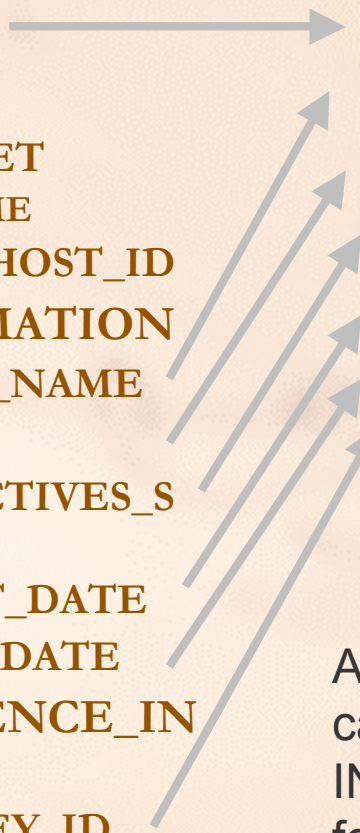
Common Attributes

- Do all entities share some common attributes?
- **InstanceID:** A unique identifier for the instance.
- **Name:** A textual tag for the instance.
- **Description:** A brief narrative about the instance.

Archive Formats

- One format for each Product type?
- One format for everything (HDF)?
- Should there be a policy What if the format specification can be archived with data?
- 30 sample “best of” datasets were provided to Steve Hughes by the discipline nodes.

An Example

- **MISSION (PDS3)**
 - MISSION_NAME
 - MISSION_HOST
 - MISSION_TARGET
 - TARGET_NAME
 - INSTRUMENT_HOST_ID
 - MISSION_INFORMATION
 - MISSION_ALIAS_NAME
 - MISSION_DESC
 - MISSION_OBJECTIVES_SUMMARY
 - MISSION_START_DATE
 - MISSION_STOP_DATE
 - MISSION_REFERENCE_INFORMATION
 - REFERENCE_KEY_ID
 - **MISSION (PDS4)**
 - NAME
 - ALIAS_NAME
 - DESC
 - OBJECTIVES_SUMMARY
 - START_DATE
 - STOP_DATE
 - REFERENCE_ID
 - INSTANCE_ID
- 

Note

ALIAS_NAME and REFERENCE_ID can have multiple occurrences. INSTANCE_ID is a unique identifier for the mission. MISSION_INFORMATION is actually used to describe mission phases.

The Seven Core Questions

1. How will PDS-4 enable "one-stop shopping", i.e., seamless access to data that reside at multiple nodes?

Adopting a common information model makes distributed searches more achievable, integration of services more possible and interpretation of data easier. Seamless access requires a seamless model. The model should specify the metadata needed to support science, engineering, administration and operations. We need to capture the context in which data are acquired to aid in its interpretation. We need to preserve information relevant to the data.

2. How will PDS-4 help users by delivering derived data products in the format, coordinate system, and map projection the user requests?

The information model must include a model for supported formats, coordinate systems and map projections. The information model must support the delivery, conversion and transformation services required to respond to a user request. The information model should be based on stated requirements for supported formats, coordinates systems and map projections. An efficient and lean data model containing those terms that add value and benefit the system is the goal.

The Seven Core Questions (cont.)

3. How will PDS-4 help data providers by automating the design, production, and delivery of PDS data sets?

A formally defined model for specifying entities and relationships will make creating compliant data sets easier and more efficient. Expressing the data model in application appropriate languages (XML Schema, ODL, OWL) will enable adoption in a variety of environments. Any tools developed which rely on the formal specification should be portable (useful) to other providers.

4. How will PDS-4 ensure that PDS standards are simple, straightforward, and consistent so that data providers and users can easily understand and apply them?

The Information Model should be expressed as a formal specification in a data engineering language. The formal specification must be translated into a version that users can easily understand and apply. The Information Model should be prescriptive with an accommodation to allow descriptive information (additional terms) to be included, but clearly separated from the PDS Information Model. The PDS Information Model should include only those terms with a well specified purpose and clear intent.

The Seven Core Questions (cont.)

5 How will PDS-4 ensure that data sets can be safely and efficiently archived in NSSDC and retrieved on demand?

While the information model aids in archiving by preserving information using standardized terms, it does not aid directly with NSSDC.

6. How will PDS-4 improve the data transfer, data integrity, and maintenance of PDS data sets?

7. Should PDS4 be required to be backwards compatible?

No. Backward compatibility places undue constraints on PDS4. Some of the simplest reasons are: If we add a new required element to an object we break backward compatibility. If we remove any elements we break backward compatibility. More complex reasons are: If alter the relationship of objects we break backward compatibility. Doing any one of this may be the right solution.