**PDS Tech Session Meeting**

**Face-to-Face Meeting**

**28 February -2 March 2011**

**JPL 321-217**

**Attendees (all or parts of three days):**

Chuck Acton (NAIF)

Rafael Alanis (IMG)

Mike Cayanan (EN)

Sheila Chatterjee (Cassini, by phone)

Richard Chen (EN)

Luca Cinquini (JPL Earth System Grid)

Dan Crichton (EN)

Amy Culver (IMG)

Rose Early (SBN, by phone)

Patty Garcia (IMG)

Mitch Gordon (RINGS)

Ed Guinness (GEO)

Sean Hardman (EN, by phone)

John Ho (EN, by phone)

Lyle Huber (ATM)

Steve Hughes (EN)

Joni Johnson (ATM)

Steve Joy (PPI)

Ron Joyner (EN)

Debra Kazden (PPI)

Todd King (PPI)

Emily Law (EN)

Joe Mafi (PPI)

Mike Martin (Management)

Stephanie McLaughlin (SBN, phone)

Tom Morgan (Management)

Lynn Neakrase (ATM)

Carol Neese (SBN, phone)

Jordan Padams (EN, by phone)

Paul Ramirez (EN)

Anne Raugh (SBN)

Mark Rose (UCD)

Elizabeth Rye (EN)

Boris Semenov (NAIF)

Dick Simpson (RS)

Susie Slavney (GEO)

Tom Stein (GEO)

Jesse Stone (SBN)

Betty Sword (EN)

Priya Venkartesan (UCD)

Michael Wendell (SBN)

Kenny Melville (GSFC, by phone)

**Welcome and Logistics (Crichton)**

Dan called the meeting to order at 8:50 AM Monday and asked people to introduce themselves. Goals for the meeting are to educate the PDS community about PDS4 basics: What is the end-to-end process and what are the key parts of the design? We will discuss outstanding technical issues and the plan for transitioning from PDS3 to PDS4.

The meeting was broadcast using WebEx.

**Project Overview (Crichton):**

The current upgrade is larger than anything attempted in PDS during the past couple decades; it should position the system for many years into the future. PDS is currently supporting data from 110 instruments on 15 active missions. Current holdings are about 200 TB, up from about 4 TB 10 years ago. PDS needs to serve as an archive for old data as well as curating and supporting near-real time data streams. There are budget and time pressures; there is now an important international component, and greater integration with the world community is needed.

PDS4 includes an explicit data model; it is implemented using XML. Ad hoc PDS3 product definitions will be replaced by PDS4 definitions that are part of the model; the core set of products is very small, but extensible. The homegrown data dictionary is being replaced by one based on the ISO 11179 RIM standard. A system of integrated registries supports improved tracking and access. The goal is to make PDS4 more useful by providing the capability for plug-in services for both management and science analysis.

The baseline plan includes delivery of Build 1 in October (last year) and an initial operational capability by October 2011 (Build 2) to support ingestion and distribution of PDS4 products. Build 3 will provide ‘user capabilities’ in June 2012 (integration of DN applications and science services, such as order and subscription services). An internal assessment of the documentation as of 15 December (Build 1b) with participation from every node was completed in mid-January; the next major review includes IPDA participation (Build 1c) and will begin in about a month.

There were questions about whether the Build 1b assessment was having an impact on the design and, if so, how and when. Raugh is hoping there will be some outside commentary before significant design changes are attempted; she is not sure that ‘internal’ assessments should be the dominant voice in determining direction. Mike Martin has been compiling a list of shortcomings in the model; he will make these available when he finishes the list.

Martin also asked whether anyone has downloaded and attempted to use PDS4 software; he stressed the importance of software beta testing. Todd King suggested that a review of the software be scheduled, similar to the internal assessment of Build 1b documents.

Crichton posed some questions for the meeting: what are the core PDS4 data types, how do we design XML labels, how do we package data for delivery and distribution, how will data dictionaries work, …? Slavney wondered what we are expecting of data providers that will be different under PDS4. Huber added a question about how the common data dictionary will be populated.

**DDWG Project Update (Hughes):**

The PDS4 Build 1b internal assessment has been completed and results are being absorbed. A team of document editors/authors has formed to oversee the effort. The DDWG had a face-to-face meeting at JPL about 3 weeks ago. DNs have been completing exercises, which test concepts and identify errors. DDWG has been providing test products for system testing and validation.

One focus area for recent attention has been developing topics of special interest to disciplines; an important topic which crosses all nodes is definition of coordinate systems and frames.

The PDS4 Information Model is the foundation for the system; from it the Standards Reference, Data Dictionary, XML Schemas, and Registry Configuration Files can be generated.

The closest thing in PDS4 to the PDS3 data set is the archive bundle. Bundles comprise collections, of which there are several types: investigations, documents, personnel, observations, targets, instruments, and instrument hosts, each of which has products. Products are derived from the four core data structures, but they can be extended to meet the needs of nodes, missions, and other data providers.

DDWG has adopted some external standards, such as the Information Model itself, the registry model (including the data dictionary), and XML.

**PDS4 Data Standards Concepts (Raugh):**

*Data Formats*: The simplest format is an N-dimensional array of homogeneous scalars; issues include the index order in which elements are stored. The next more complicated format is a table; the ‘columns’ may have heterogeneous formats, but every row is the same. Interleaving is prohibited in PDS4. The third structure is a parsable byte stream; examples include spreadsheets and documents. The most complicated format is an encoded byte stream — a computation of some sort is needed in order to extract the information (PDF, MP3, MPEG, etc.). Note that documents, geometry, calibration, etc. are all ‘data’ in PDS4.

*Products*: A digital object is any sequence of bytes that is not an XML label. A product is a label plus all the digital objects that the label describes (could be zero). Each label is in a file of its own; but a single file may contain more than one digital object. A single digital object may not be split across files. Each product has an identifier, which is unique across the registry space. The PDS4 Registry Service tracks products; versioning is tracked at the product level.

*Archive Design*: Groups of similar products are gathered into collections; ‘similar’ is not rigidly defined. Collections are organized into bundles. A large mission archive can be broken into separate bundles. These are logical organizations, but they can also become physical organizations. Slavney asked whether a PDS4 collection is equivalent to a PDS3 data set; there was consensus that ‘data set’ maps better to bundle, but that there are a lot of different ways data can be grouped in PDS4. Technically, a collection is a product (an inventory list); likewise, a bundle is also a product. Mark Rose asked how to find documentation that goes with a data product; in PDS3, there is a DOCUMENT directory. In PDS4, the DN can rearrange things, but Rose noted that one user complaint is that DNs do things differently and ‘hiding’ documentation using variable retrieval algorithms may be negatively received. Gordon said he doesn’t think there is resistance from DNs to adopting a single set of rules that apply across the system.

*Labels*: Users of digital objects need structure information for reading and interpretive information for making sense of the bits. The latter can include mission metadata (about the instrument, its operation, the mission itself), DN metadata, and PDS metadata.

*XML*: The eXtensible Markup Language defines parsing rules, is recommended by W3C, and is supported by 3rd party vendors. Mark Rose asked whether use of the term ‘attribute’ is a good choice, since XML already has a well-defined meaning for ‘attribute’; Gordon noted that PDS4 will use the term ‘XML attribute’ when ‘attribute’ comes up in the XML context.

*Namespace*: A namespace establishes a context for definition. Two items with the same name but from different namespaces generally have different meanings. Namespaces are used to delegate authority for creating attributes used in label sections. Data preparers will have authority to create descriptive attributes and classes in their assigned namespaces.

*XML Schema*: The XML standard defines only the syntax, not what tag names mean. A schema is used to define tags, constrain their content, define standard values, and designate whether attributes are required or optional.

**PDS4 Glossary (Simpson):**

Simpson went through the current version of the PDS4-wide glossary. It will be available from a PDS web site, and terms should be used consistently throughout all PDS4 documentation. Semenov and King didn’t like examples which showed products constructed from a digital, physical, and conceptual objects because that combination is very unlikely to be found in practice. Todd even proposed banning such triplets as a matter of policy. Mike Martin was apprehensive about the reception of meta-attributes by data providers and users.

**Document Set Overview (Gordon):**

The document set is a work in progress intended to provide help to people starting data migrations over the next year. The goal is to provide a measured introduction to PDS4, starting with the big picture and working into details.

1. *Introduction*: A couple pages that briefly describe each document and the audience that might find it interesting.

2. *Concepts Document*: An overview of PDS4 concepts. Mark Rose asked for a brief description of how data are submitted to PDS.

3. *Jumpstart Document*: A brief description of PDS4 concepts in terms of their PDS3 analogs. It will be of limited use to people with little or no PDS3 experience.

4. *Data Provider’s Handbook* (*DPH*): A cookbook guide giving step-by-step instructions for creating a simple archive.

5. *Standards Reference*: One of two fundamental reference documents (for use with the *DPH*); it is a human readable compilation of the rules embedded in the Information Model.

6. *Data Dictionary*: The second fundamental reference document. There are abridged and unabridged versions.

7. *Enhanced Glossary*: Intended to bridge the document set. There may be more than one version, aimed at different audiences.

8. *Examples*: Samples of various products, collections, bundles, and packages. These are frequently referenced by the DPH and Standards Reference.

Mike Martin asked whether there will be a (short) specification document. Raugh thinks the first section of the Jumpstart Document should provide some background; but Martin wants something short, but technical. Gordon thinks an executive summary of the Standards Reference might do the job; Crichton agreed. Martin believes that a short, readable summary would be evidence that the system designers know what they are doing.

**Local Data Dictionaries (Raugh):**

Raugh stepped through a demonstration of creating a data dictionary using two schema files downloaded from a recently distributed package.

**Label Overview (Mafi):**

There are generic, tailored, and specific schema. Generic comes directly from the Information Model; they can be downloaded from the schema/pds4/common web site. Tailored schema are extensions of the generic schema by adding mission and discipline node components. Specific schema are constrained versions of the first two. Labels are instantiations of the other three to describe a digital or non-digital object.

The label has the following ‘areas’: identification, cross reference, observation, file, data, and alternate data. Todd King asked whether the cross reference information will be tightly or loosely constrained — is every cross reference included, or is the number of cross references so minimal that only the data provider’s work can be reproduced?

**Questions and Answers (Law):**

Mike Martin asked what is the range of acceptable encoded byte streams; the assumption is that support for the encoded format is either widely available or provided by PDS, but what does that mean in practical terms? Gordon replied PDS is not signing up to support a lot of user-provided encoded streams. Raugh said that, if PDS accepts a standard, PDS should purchase a copy of the standard and make it available to users (if purchase is required).

Then Mike asked what happens to PDS3 interleaved image files. There are several options.

The ISIS people have offered to write software to converted ISIS 3 data to PDS4; they are also migrating ISIS 2 data to ISIS 3 formats.

King asked what is the expected return from the Build 1c review? Are the results from previous reviews being folded into the 1c products to the point where there is real improvement? What ‘big ticket items’ are still being debated by the DDWG, and which of those have not really not been updated in the specification yet? Raugh said the IPDA should be aware that Build 1c is still a work in progress; she is worried about the schedule — there is still a lot to decide, and it may be necessary to go to MC and admit that more time is needed to complete the design in a rational way.

**DDWG Splinter (Hughes)**

Primary and Secondary Collections and Bundles: Mitch Gordon led a discussion on inconsistencies in the way primary and secondary collections are defined. He extended the terms ‘primary’ and ‘secondary’ to memberships in collections and bundles and to bundles themselves. Later Mitch distributed a summary of the discussion by e-mail; but many of the issues were not fully resolved.

Should primary collection identification be in data product labels? At present, the primary collection association is an optional part of the cross reference area; perhaps it should be moved to the identification area and made required. If the collection were to be divided, many labels would have to be revised. If a collection were to be divided, why not close the first collection and open a second?

Packaging: Checksums are included in a manifest which accompanies an archive bundle; but it is external to the bundle. Gordon will explain tomorrow.

Name Resolution: A topic proposed several weeks ago by Anne Raugh, queued on this agenda, but not discussed.

**Adjourn and Reconvene (Crichton):**

The Monday session was adjourned at 6:30 PM; the Tuesday session began at 8:35 AM.

**Data Set Migration - Overview (Gordon):**

Mitch proposed migrating a simple, ground-based PDS3 data set to PDS4. The original data set had catalog files, an index file, two types of data (‘easy’ and ‘source’), document files, and SPICE files. The PDS4 archive bundle contains five collections: two data collections, a document collection, a schema collection, and a SPICE collection.

LIDs can be formed starting at the bundle level. He proposed ASTROM\_0101, which must be checked for uniqueness within PDS; everything below that can be named by the DN, which ensures uniqueness within the bundle.

Bundle URN:NASA:PDS:ASTROM\_0101)

Data collections URN:NASA:PDS:ASTROM\_0101:DATA\_DERIVED and

URN:NASA:PDS:ASTROM\_0101:DATA\_SOURCE

Document collection URN:NASA:PDS:ASTROM\_0101:DOCUMENT

What happens to PDS3 files that don’t fit within the structure above? AAREADME.TXT can be optionally incorporated into the bundle product (or referenced by the label), ERRATA.TXT and INDEX.TAB can go to the document collection, VOLDESC.CAT goes to the trash, and CATALOG files can selectively go to context and description areas. There are at least two options for PDS3 label files: they can be retained as part of new product definitions or they can all be collected in a PDS3-LABELS subdirectory in the document collection (with TBD PDS4 labeling). Sending PDS3 labels to the trash may also be an option, but the pros and cons are still being evaluated.

Mike Martin and Mark Rose asked about the possibility of including HTML labels; Rose noted that HTML sometimes provides ‘rich’ content, which could be valuable if saved. Raugh argued that ‘rich’ may not be a good criterion and that HTML is not a good standard; but XHTML might be possible. Semenov said current NAIF volumes have AAREADME.HTM files in the root.

What new files are needed for PDS4? At the bundle level, the bundle label (including the inventory list) and an optional README.TXT are needed. For data sets, which may not be final, ‘update’ and/or ‘release notes’ files may be added to the document collection. There is an open question of whether metadata should be updated during a migration; if so, does the migrated data bundle have a version number which distinguishes it from the original (unmigrated) data set? At the collection level, there are labels and inventories (and optional README.TXT files [when did README.TXT files at the collection level become possible? — RAS]). At the product level, there are new PDS4 XML label files; and the schemata go into the SCHEMA collection.

How is the bundle delivered? An MD5 checksum manifest lists every file in the delivery package; the package can be zipped or tarred before shipment. If the bundle is too large for delivery in one package, it can be divided into smaller pieces. A spirited discussion on whether checksums should be a *required* part of the archive followed.

Mark Rose said it would be nice to have an example of a complete archive on-line by the time of the Build 1c review.

**Data Set Migration - Example (Neakrase):**

ATM has employed two students (Shannon Rees and Matias Roybal — one in physics, one in engineering; neither with any computer background needed for either PDS3 or PDS4) to migrate PDS3 data from five Phoenix instruments to PDS4. ODL doesn’t look anything like XML; parsing PDS4-appropriate values from PDS input is non-trivial. The bulk of the work has gone into dealing with documents, catalog files, etc. The goal has been to prototype using the Phoenix data but to develop tools that can be generalized.

A generic schema was downloaded from the EN web site for character tables; it was edited to add constraints and to add mission and node information. A single schema was sufficient for all five instruments; the table formats are identical although the contents vary. The ‘mission dictionary’ allows tailoring for each of the five table types.

ATM is using Python 2.6.5-2.7.1 (2.x.x); there is a separate development tree 3.x.x, which should not be used for PDS4 (its syntax is completely different and it appears to be less useful in the PDS4 context). ATM has a Python ‘dictionary’ which maps PDS3 values to appropriate PDS4 counterparts; but there are problems when keywords are used more than once in the same label. There is a GUI, but the routines can also be run automatically. Migrating a single LIDAR file takes 15-20 seconds.

Document schemas are being upgraded; there are Python scripts to handle the migration. Bundle scripts need to be written for each instrument since the contents differ.

Hardman and Joyner visited NMSU and installed registration software; ATM has not used the software much, but Sean has registered a few things remotely from EN. The server requires a current version of Java.

ATM is now looking at binary tables and FITS data.

**Questions and Answers (Rye):**

Mike Martin asked whether array\_base and table\_base could be changed to simpler terms — ‘array’ and ‘table’, which are what most people would call them. Rye does not think array\_base and table\_base will be used in normal conversation; she thinks Table\_2D and similar terms will be the ones used. She argued that Table\_2D is the extension that most data providers will need. A lengthy discussion followed on the relative costs and benefits of having small numbers of base classes and small numbers of class extensions. Steve Hughes equated this to flexibility in the model versus flexibility in the schema.

**System Overview (Hardman):**

The PDS system upgrade includes catalog data base, data search, subscription, and phone book changes. Product registration, search, and preparation tools will add new functionality. Although everything will be upgraded, basic functions needed by data providers will not be changed significantly; there will still be design, generation, and validation tools.

The system design is planned to occur over three phases: ingestion, distribution, and user services. Level 4-5 requirements have been traced back to Level 1-3 PDS Requirements. Specifications have been completed and reviewed for registry, harvest, security, and report services; specifications are in progress for preparation (tools) and search; and specifications will be drafted in the near future for the operator portal and monitoring services.

Build 1 included a prototype ingestion subsystem that performs registry, harvest, and security services (Lightweight Data Access Protocol, LDAP). Although not related to ingestion, the report service was also included in this build. There was also an initial data provider tool suite — a validation tool and user guides for selected off-the-shelf parts of the design tool.

The plan for Build 2 includes a prototype distribution system (a search component and updated data search interface at EN); further development of the ingestion subsystem; configuration of the report component; selection of an off-the-shelf monitor component; and development of tracking, subscription, and phone book interfaces. The plan for Build 3 includes tools for transformation and visualization, integration of new components with existing node software and infrastructure, and incorporation of findings of ongoing research into data movement and storage solutions.

There are half a dozen EN staff working on system design; the System Design Working Group includes about 8 people from across PDS.

**System Design Concepts (Hardman):**

Design concepts include use of common, extensible software; technology that allows for future expansion; minimization of tight coupling so that phased development is easier; and use of COTS software where it is cost-effective. Goals include increased ingestion efficiency, improved tracking and greater archive integrity, facilitation of data search across nodes, improved delivery to data users and the deep archive, increased integration of services across the system, and simplification. Design considerations include local governance for data and metadata, estimation of future data volumes, and minimization of unnecessary data transfers because of bandwidth limitations.

Interfaces are REST-based (Representational State Transfer) over HTTP; this applies mainly to registry and search services. Libraries are being developed using Java and Python; portions of the library will be open for contributions from the PDS community.

A registry provides services for sharing content and metadata. A federated registry allows cooperating registries to appear and act as a single virtual registry. This provides seamless information sharing while preserving local governance. The registry provides tracking, auditing, locating, and maintaining ‘artifacts’ within the system. The search service serves as the publicly available information for information held in the registry (which itself is not publicly accessible); the service allows for metadata ‘annotation’ — the possibility of updating geometry, feature identification, etc.

**System Ingestion (Hardman):**

Ingestion involves preparation of and registration of PDS products; tools will be provided to assist data providers. XML editors can validate syntax of PDS4 labels; but how do you verify that each data object is as described? This is on the EN development list, but it does not appear to have high priority (partly because the Information Model remains in flux).

There was discussion about how to identify PDS3 data within PDS4. The DATA\_SET\_ID and PRODUCT\_ID should uniquely identify a PDS3 data set and its products; but how do you identify the data set after it has been fully *migrated* to PDS4? The PDS4 ‘data set’ is not the same as the PDS3 version. There was some support for distinguishing using the version identifier; but there were also objections for multiple reasons.

The harvest tool crawls through data capturing and registering product metadata; this is the first step in tracking products. The tool can be operated in modes that allow for both periodic or on-demand registration of products. It can be configured for registering both PDS3 and PDS4 archives. It is designed so as to be easily integrated into existing node operations.

The registry will compute an MD5 checksum if none has been supplied; if a checksum has been provided, the registry will verify it (and report errors). Guinness warned that Mini-RF deliveries require days of CPU time to calculate checksums. Hardman has not decided how to handle deliveries with errors; it would not be good to ingest only parts of a defective delivery. How to synchronize the DN registries with the central registry is TBD.

Registry entry types include non-digital objects, digital objects, and relationships. Sean anticipates a large number of associations, and he is trying to decide whether to include all of these in the registry and how to display the chosen ones to users.

The harvest service pulls metadata from the Identification Area. The logical ID plus the version ID serve as the unique identifier for each product. ‘Title’ becomes the display name; it is not necessarily unique. ‘Product class’ is used to classify the product type; it is not clear whether this is useful if there are two or more objects (of different types) in the product. There was discussion about whether using a single value is misleading in these cases; it implicitly designates one of several objects as being ‘primary’ and users may disagree on how those are chosen. It is not clear how this can be resolved.

In the Cross Reference Area two registered products are linked through an association.

Registry configuration is based on files derived from the Information Model; this is where harvesting requirements (*e.g*., product type and common metadata for the product types) are specified. Slavney asked how to add more information to the registry after the initial harvest; Hardman replied that EN will develop a procedure so that errors can be corrected (but this does not correct errors in the original metadata). How to supplement the initial harvest result is TBD.

A status “submitted” indicates successful registration; but registered products are not visible to the public until ‘approved’. The criteria for ‘approval’ will be set by PDS policy. An operator issues the approval; an operator can also set status to “deprecated” or “deleted”. Deprecated entries will not be accessible, but they can be tracked.

EN will periodically pull information from the node registries to create an aggregated registry. Only entries with status “approved” will be aggregated. This allows nodes to maintain governance while facilitating PDS-wide tracking and reporting. There may be slightly more up-to-date information in node registries; how to decide which to query is TBD. Some nodes may be interested in data ingested through others (for example, RINGS interest in IMG data); there is a question about how to add metadata of interest to the second node that has not been harvested by the first.

**System Ingestion Demonstration (Ramirez and Hardman):**

The demo runs under Apache Tomcat.

There was interest in some of the software tools used; Paul offered to send a list so that attendees could try them.

**Adjourn and Reconvene (Hardman):**

The Tuesday session was adjourned at 5 PM; the Wednesday session began at 8:30 AM.

**System Distribution (Hardman):**

Distribution involves discovery and retrieval. The SDWG and DDWG are in the process of identifying and capturing search scenarios — to build the infrastructure for facilitating all levels of product discovery. The order and subscription services are farther in the future.

The search service returns a set of matching results to queries. The search service provides a second level of harvesting from metadata; generation of search indices from registry metadata supports multiple query formats. The service is deployable, which could speed searches that are DN specific.

Gordon asked whether DN search services should be REST-based. Sean replied that REST is the recommended direction, but he doesn’t think PDS can make this a requirement. He would like to talk to anyone planning to develop such a capability in the next 1-2 years. Tom Stein said GEO will not be implementing a fully REST-based system; GEO already has some REST-based services, but there isn’t enough labor available to convert the entire node.

Mark Rose said that the user survey showed PDS customers prefer a ‘federated’ search in the sense that they want to find data across DN boundaries. He acknowledged that this may not be feasible or cost-effective; but it is what they are requesting. There were several objections, largely on the grounds that ‘federated search’ is itself discipline specific — different users have different expectations of what a PDS-wide search would produce, depending on their interests and backgrounds. Cinquini and King both suggested procedures that queries could be posed with more parameters than any DN would likely answer itself; but each node would respond as best it could.

There was a question about how much storage is required to support the search system; if PDS has 20M products (the estimate 2-3 years ago) and each label requires 2K storage, then a disk with 40 GB capacity should be sufficient. Although the total storage is significant, it does not appear to be unreasonable given devices currently available.

SDWG is looking at adding externally produced metadata to the data base of search indices; this could make searches more effective, but handlers for the external data would need to be designed and integrated.

The SDWG needs consistent population of the Identification and Cross-Reference Areas. There is a recognized problem with product\_class in the first case; cross-references are less well-defined (*e.g*., Investigation) and Subject Area is imprecise at best.

**System Integration (Hardman):**

One goal in development of the new system has been to minimize the impact on nodes; this does not mean no impact, and impact will depend on the node.

The Build 1 deployment included services at EN and some registry services at ATM; but the ATM installation is not integrated with the EN system.

Mike Martin asked whether the REST-based search and REST-based registry services are equally accessible to the public. Sean doesn’t like that idea (as noted above, the search service provides an interface to the registry not direct access to the registry); but Mike thinks it would OK, and it fits better with a service oriented architecture. There could be valuable user-added features over time. Mike used the example of NAIF Cassini sequencing information, which will probably not be easily accessible through a search interface.

Through Build 2, the impact will be mostly on node processes rater than node systems. There will be some installations at DNs, but these will mostly be to integrate harvest and registry services.

Guinness asked about installations at data nodes. Sean said there should be no differences; the data nodes can be handled in the same way as the discipline nodes.

Slavney asked whether there is going to be a master list for tracking PDS3-to-PDS4 migrations. There was a suggestion that MIWG could be responsible for this. There are questions about what should be tracked in PDS3 data sets before they are migrated to PDS4.

Gordon asked whether the archive status history is going to be tracked at the product level. Sean said the specifications need to be clarified, but he thinks the history can be tracked at the granularity needed — such as that to meet requirements set by the ROSES/PMDAP announcement. King noted that the tracking product does not have to be ‘versioned’; it should be possible to ‘annotate’ an original tracking product.

King suggested ‘ghosting’ the pieces of Slide 5 which are not deployed in Build 1 and then again at Build 2. This could make the figure better if it is to be presented to the Management Council.

**Schedule/Deployment/Operations (Law):**

Build 1 was in October 2010, Build 2 will be in October 2011, and Build 3 will be in June 2012. Build 2 will include an updated Information Model and new versions of primary documents and schemata; tools and service packages will be updated. Nodes will have the harvest tool, local registry service, and an optional search service. Nodes should identify an appropriate server — one that can be accessed remotely from EN.

Ron Joyner has migrated about 85 percent of the central catalog into the PDS4 registry; this process will be completed by Build 2. The plan is to decommission the central catalog at that point; but there were strong objections. There will be people still delivering PDS3 data sets who need access to previously delivered DATASET.CAT files, for example. Emily said all of the PDS3 information will be available even though it may be in a different format. EN agreed to keep the catalog available until the DNs agreed it was no longer necessary. Tom Morgan asked whether there was a cost associated with keeping the PDS3 catalog alive; Crichton replied that there would be no additions to the catalog after October, so the cost should be very minimal. One residual question is whether Joyner has saved NSSDC IDs in the PDS4 registry.

Nodes will need to provide log files in specific formats to support the report service. Raugh asked what this really means; SBN limits its logging because there are serious CPU impacts and the existing file sizes are GB each month so they cannot be routinely delivered as e-mail attachments. GEO uses a format which is not what EN needs; but Hardman said there is a simple conversion.

Nodes should plan to start using the harvest tool to populate the local PDS4 registry with data from PDS3 labels once the software has been deployed.

Nodes should begin developing data migration plans that can be implemented after the Build 2 release. Amy Culver suggested some direction (and coordination) from Management Council as to which data sets have highest priority for migration. Morgan asked why this is not something nodes will want to do; but Raugh noted that there are resource issues. Future Tech Sessions will focus heavily on data migration, including what the process is, what the steps are, how they should be carried out, and what tools are needed to facilitate the migration.

**Adjourn (Law):**

The Wednesday Tech Session was adjourned at 11:45 AM.

**DDWG Splinter (Hughes):**

Mike Martin asked for a 2-3 page introduction to PDS4. Elizabeth Rye has drafted a legal-size double-sided brochure for this purpose. There was positive reaction. She will distribute the draft; comments should be returned within a week.

There is a system review on about 25 May. The next DDWG F2F should be in the late-May early-June time frame. Hughes will request input using another Doodle.

After lunch: Kazden, Simpson, Rye, Hughes, Joyner, Raugh, Gordon

Documents need to be revised with the following priority and with expected dates for completion of the next releasable version. Authors should forward copies for DDWG or Tech Session review before release.

1) Introduction (Gordon) – 15 March

2) Glossary (Simpson) – distribute for review 4 March, ready for IPDA 11 March

3) Data Standards Concepts (Raugh) – DFR 15 March

4) Jumpstart Guide (Simpson) – DFR 10 March

5) DPH (Joyner and Gordon) – rough version 4 April, ready for IPDA 16 April

6) Standards Reference (Rye) – 4 April, ready for IPDA 16 April

7) Data Dictionary Tutorial (Hughes) – DFR 9 March, ready for IPDA 16 April

8) Data Dictionary (TBD; Abridged, and algorithm for creating)

algorithm for generation from unabridged – 15 March

 introduction rewrite – 15 March

 regenerated – 4 April

9) Examples (Joyner and or Chen; need document, collection, archive\_bundle, context, table character, image, …) – 21 March

10) Information Model Specification (Hughes) - 15 March

 Freeze and generate schemas – 21 March

11) Brochure (Rye)

12) Registry documents for operations (Hardman?)

Data Dictionary Scrubbing:

Hughes will provide Simpson with a spreadsheet version of the current DD; Dick (in consultation with Guinness and Huber) will edit the spreadsheet and submit the changes once. Keep name changes to a minimum (how to handle field\_bytes vs bytes should be discussed as an extra-DD issue).

 Check meta-attributes and allowable values

 Check classes

 Check attributes

Glossary: 1) drop “data element”; use attribute and/or class where appropriate. 2) Is there a problem with “elements”? Possibly but any solution impacts the model, so no change proposed. 3) Rye concerned about prominence of “information object” but no changes proposed.

There was discussion about whether has\_document\_collection is also needed in product cross-reference areas; currently only has\_document is available. There is a related issue in that the model for the registry does not allow associations with collections; Hardman can implement a work-around.

A few additional notes / suggestions captured by Emily:

* Formalize system and beta test
* A few pages of high level PDS4 principles / rules
* Increase the base of PDS4 assessments
* Push Build 1c assessment out to May
* What/how to prepare the data providers and users
* Standard directory structure for all nodes for ease of data browse
* Common set of storage rule
* Exclusion of all 3 types of objects in a product in the model
* Estimate size of database needed
* Hardware needs
* Web and FTP logs formats needed sooner rather than later
* MIWG, end to end process to track what has been migrated, MIWG should not be the group to work subscription category
* A table/diagram that depicts what have been delivered/deployed and future delivery (both system and data)