

RFA #	Name	Topic	Problem	Recommendation	Disposition	Timeframe	Comments	PLAN	RESOLUTION
RFA_Build2a_001	Neakrase	LADEE & MAVEN Product Development	Problem: We currently do not have cohesive standards, documents, and schema/XML templates to allow LADEE and MAVEN to begin the development of their product pipelines.	Recommendation: We freeze the model and schemas as soon as possible, to allow materials to be distributed to the teams before February so they can begin development of their products and data pipeline for operations (i.e., SIS development begins for LADEE in Close out the information model issues (or at least nearly all of them) on ACR's slide 3 and, from her slide 4, at least the units issue. From her slide 5, settle on LDDs and node interaction (2/3 of first item). Other items are needed, but not for build 2b.	DELIVERED	Build 2b		PDS will freeze the information model and data standards for build 2b in support of LADEE/MAVEN.	Build 2b, Version 0.6.0.0.h was released on January 31, 2012
RFA_Build2a_002	A'Hearn	Information Model Issues (Rough 09)	There is a distressingly large number of known open issues in the information model. Most of these need to be DELIVERED.		DELIVERED	Build 2b		This has been scoped into build 2b and build 2c+ changes.	The recommendations in ACR's IM review associated with the PDS4 label "preamble" and the table_base classes have been implemented. The remaining issues will be address after Build 2b.
RFA_Build2a_003	Anne Raugh	SBNUMD43: Reference_association_type values are jargon	The "has-a" notation for associations is not something we should be requiring data preparers to learn to document what is to them a completely straightforward relationship. PDS should do this relational translation in the background.	Pick values that will help data preparers do the right thing without having to reference additional documentation.	DELIVERED	Build 2b		Value meanings will be assigned to each reference type value to improve usability.	Value meanings have been written and included in the documentation. Tools will ultimately shield the user from these details.
RFA_Build2a_004	Anne Raugh	SBNUMD42: Missing definition for "type"	The "type" attribute is required in the bundle Identification_Area_Bundle but there is no definition for it in that context in the data dictionary, and apparently no standard value list. The generic definition of "type" is too vague to be useful - it amounts to a synonym of the word "type".	Justify the existence of the attribute. If it is required, provide a definition and standard value list relevant to the context.	DELIVERED	Build 2b		Implement the recommendations from ACR's IM review.	The type attribute is included in the Primary_Result_Description to provide the general data type of the results. This was part of the preample update.
RFA_Build2a_005	Anne Raugh	SBNUMD39: field_location is ambiguous	The field_location attribute does not specify what the unit of "location" is. It also has an upper limit, which is probably not wise.	Define units for "location". Either require them to be specified in each instance of the field_location attribute, or fix the unit in the model and propagate it to schemas and through the documentation.	DELIVERED	Build 2b			The definition of field_location has been modified to indicate that the measurement unit is "byte".
RFA_Build2a_006	Anne Raugh	SBNUMD36: Product_Collection_Data has multiple member tables	A collection can have up to three member tables. Why is it necessary to have three separate tables to handle information that could as easily be in a single table with multiple columns, one of which (the version_id) might contain an "N/A"? And if the information for primary members is always required to be in a separate table described by a unique class, why is it necessary to have also have a flag value indicating it is present?	A single table should be sufficient for this purpose. Strong justification needs to be made for complicating the process for data preparers.	DELIVERED	Build 2b		The consensus is that collections will continue to have three types of inventory tables.	The team is looking into replace the three named inventory tables with a single inventory and using schematron rules to validate the names, etc.
RFA_Build2a_007	Anne Raugh	SBNUMD34:contains_primary_member value potentially ambiguous	When it is not completely obvious from the name of an attribute, like this one, that it contains boolean data, the "True" and "False" should be spelled out. It's not clear what this has to do with identification. This is presumably one example of something that may occur in many places.	Require "True" and "False" to be spelled out in boolean attributes when there could be doubt, or change the attribute name so that the standard value meanings are clear to an end-user.	DELIVERED	Build 2b			The ASCII_Boolean data type was updated to match the XML standard. The ASCII_Boolean_TF data type is being deprecated.
RFA_Build2a_008	Anne Raugh	SBNUMD26: Target_Name inconsistent with similar attributes	Target_name is in the Subject_Area, but not in the Observation_Area (or even Cross_Reference) like other high-level observational metadata.	Redesign the top-level required product classes to group related concepts as well and applying rational standards for what is required and what is not.	DELIVERED	Build 2b		Implement the recommendations from ACR's IM review.	The new Target_Identification class addresses target identification and naming issues.
RFA_Build2a_009	Anne Raugh	SBNUMD25: Observing_System Misplaced	Observing Systems is under "Cross_Identification", but it's not related to cross-identifications, it provides the highest-level observational meta data	Redesign the top-level required product classes to group related concepts as well and applying rational standards for what is required and what is not.	DELIVERED	Build 2b		Implement the recommendations from ACR's IM review.	The Observing_System class has been modified and moved to the Observation_Area.
RFA_Build2a_010	Anne Raugh	SBNUMD24: Subject_Area is redundant and misplaced	The contents of this area are not really identification, yet it is inside the "Identification_Area". Most of the information in here is duplicated in "Observing_System"	Redesign the top-level required product classes to group related concepts as well and applying rational standards for what is required and what is not.	DELIVERED	Build 2b		Implement the recommendations from ACR's IM review.	The Subject_Area class has been omitted.

RFA_Build2a_011	Anne Raugh	SBNUMD20: axis_storage_order has no standard values	Storage order should not be allowed to vary in the first place. If it is allowed to vary, it should only be allowed to vary in very specific ways. These need to be defined and mapped to specific enumerated values.		DELIVERED	Build 2b	The recommendations from ACR's IM review are being reviewed by an Imaging team.	Implement the Imagings team recommendations	The new array data structure has been implemented and addresses this issue.
RFA_Build2a_012	Anne Raugh	SBNUMD19: "unit_of_measure" is excessively verbose naming	Does the attribute name really need to be "unit_of_measure"? Are our users so dumb they will think that "unit" refers to anything other than a unit of measure in any of the contexts in which this XML attribute appears?	Do not be verbose for verbosity's sake in any attribute or class names.	CLOSED	Build 2b	This attribute is used only in the Ingest Data Dictionary schema. This schema will not be included in the schema used by data providers. Also this attribute has been previously reviewed and updated by the team as part of the data dictionary scrub.	No Change	
RFA_Build2a_013	Anne Raugh	SBNUMD18: local_identifier required	local_identifier seems to be required wherever it appears. I see no justification for requiring an additional identifier in all, or indeed in any, of these cases.	Justify the requirement for the additional local_identifier, or make it optional.	DELIVERED	Build 2b		Implement the recommendations from ACR's IM review.	The local_identifier attribute was made optional in most classes and in some cases removed.
RFA_Build2a_014	Rose	Vacuous Types Should Be Removed from the Schemas	Comment/Concern: A "vacuous" type (my term) is an XML Schema type that extends some base type but neither extends nor restricts it. In other words, it is a synonym for the base type. The presence of vacuous types in the schemas makes them longer, harder to understand, and harder for data providers to work with. All vacuous types should be removed from the schemas unless there is a strong justification for their existence. In particular, these 52 types in the build 2a schemas are vacuous and should be removed:	Recommendation: Remove all the vacuous types listed in RFA	DELIVERED	Build 2b	There has been a significant effort to remove vacuous classes from the XML Schema. Some vacuous classes remain in the model for organizing purposes, for example the TDO_* classes.	Continue effort to remove vacuous classes.	All "vacuous" classes have been removed from the schemas. The array_base subclasses remain since there are significant updates to these classes planned in the near future.
RFA_Build2a_015	Rose	Elements are multiply defined in "cousin" classes	Comment/Concern: To make it easy to create language bindings from the XML schemas, an element that is in 2 related classes should be introduced in a common base class. However, there are 103 elements that are multiply defined in the build 2a schemas in classes that share a common base class. In particular, these elements are multiply defined: (See RFA)	Recommendation: Adopt a policy that elements should be defined exactly once within the schema. This forces elements to be pushed upward in the type hierarchy to the common base class of those types needing the element.	DELIVERED	Build 2b	There has been a significant effort to address this issue. This issue will continue to be discussed after Build 2b.		Significant changes have been made. Additional changes are being considered in association with the array and table base discussions.
RFA_Build2a_016	Anne Raugh	SBNUMD38: First field_location fixed/not fixed to "1" in collection inventories	The data dictionary states that the first field in an inventory table must start in location 1 (whatever that is) but this is not propagated to the schemas.	Decide what the correct answer is and propagate it to all references	DELIVERED	Build 2b			Assert statements have been added to validate these constraints.
RFA_Build2a_017	Anne Raugh	SBNUMD37: Inventory_LIDVID_Primary and related tables are missing fixed data	The number of fields and the field data types are fixed, and thus should be included in the schemas for the product inventory tables.	Add to schemas and disallow user changes.	DELIVERED	Build 2b			Assert statements have been added to validate these constraints.
RFA_Build2a_018	King	Table Record	The definition of Table_Record does not contain a specification of fields which can be included in a record. Without a list a table record cannot be defined using the base Table class.	Tables can contain a combination of field types (character, binary, bits fields, items). Adding a "choice" with a list of possible field types to "Table_Record" allows all types of tables to be described with one record type, greatly simplifying the model. Fields such as Table_Field_Checksum, Table_Field_File_Specification_Name, Table_Field_LID, Table_Field_LIDVID are not included because each can be described as Table_Character_Field with the content validated using tools such as Schematron.	DELIVERED	Build 2b		Implement the table class recommendations from ACR's IM review.	There are now three tables, Table_Character, Table_Binary, and Table_Delimited. They each have a respective Record_* and the xml schema "choice" construct is used to define the allowed fields.

RFA_Build2a_019	King	Instrument Type	Instrument lacks an Instrument_Type. A common constraint is to look for data which originates from particular type of instrument. For example, data from a "magnetometer". There is a limited set of instrument types and this value should be an enumeration.	Add a Instrument_type attribute to Instrument.	DELIVERED	Build 2b			The Type attribute has been added to the Instrument class. Permissible values have been defined.
RFA_Build2a_020	King	Bundle	Use of "file_specification_name" in Bundle_Member_Entry can become stale very quickly. Location of a collection should be through logical ID only (though LID_reference is optional!). There is a need for persistent Bundles so that collections (data, document, calibration, etc.) can be retained as a deliverable unit. These collections may not reside with the bundle description, so the file path and name should not be set in the description.	Removed "file_specification_name" from Bundle_Member_Entry so that the reference to other products is by identifier only.	CLOSED	Build 2b	No Change		The Bundle/Collection review team has recommended that the file_specification_attribute remain. The recommendation was reviewed and accepted by the DDWG.
RFA_Build2a_021	King	Table Classes	There is no "description" element for the Table_* classes. Tables need description to be more clearly identifiable to a user.	Add a "description" element, perhaps eliminating the "comment" attribute.	DELIVERED	Build 2b		Implement the table class recommendations from ACR's IM review.	The comment attribute has been replaced by the description attribute in all the fundamental structures.
RFA_Build2a_022	King	Cross Area Reference	There is very little direction with regards to how the Cross_Reference_Area is to be used. It could be limited in use to associate two products which are tightly connected (e.g. a data product, and a browse representation of those same data). It could also be used to indicate very loose associations (e.g. documents associated with an instrument, mission, or observation to which a data product belongs). Tight associations should be specific attributes. The looser associations might be better indicated in some other way (e.g. by including the collections to which they belong in the same bundle).	efine tight associations as named attributes.	DELIVERED	Build 2b		Implement the table class recommendations from ACR's IM review.	The cross_reference area have been omitted and replaced by the internal_reference and external_reference classes. In addition the references have been distributed across several classes in order to provide the user with better contextual information
RFA_Build2a_023	Anne Raugh	SBNUMD15: Telemetry_Parametes.xsd in wrong namespace	I was pointed to the Telemetry_Parameters.xsd file as an example of a local dictionary file. However, despite the fact that it is stored in what looks like the Imaging namespace, the namespace identifier in the file is the global one - so this was not a local example, either.	Correct namespace in the Telemetry_Parameters class or move it to the correct schema location. Provide an actual working example of a local dictionary.	DELIVERED	Build 2b			Corrected namespace in Telemetry_Parameters
RFA_Build2a_026	A'Hearn	Release	Release to the broad community of a standard that is likely to change is a mistake. Release of build 2 (a,b,c or whatever) should be limited to nodes, LADEE, and MAVEN. It implies that the data format should be driven by every individual's favorite format. This is against the philosophy of limiting archival formats but providing various favorite output formats.	Limit builds 2x to LADEE, MAVEN, and nodes	DELIVERED	Build 2b	Build 2b is an internal build so PDS can support LADEE/MAVEN.	Build 2b will be released to nodes working with LADEE and MAVEN.	Build 2b was released to nodes working with LADEE and MAVEN.
RFA_Build2a_027	Susie Slavney	PDS4 Build 2b requirements traceability matrix	To be able to judge the readiness of PDS 4 Build 2b we should see a list of requirements that were intended to apply to 2b, and a matrix showing how they have been met. If a requirement has not been met yet, we need to know the amount of work remaining to be done to know if it is reasonable to expect it to be done by the deadline.		DELIVERED	Build 2b		Tracability Matrix posted to the EN site will be linked to this RFA response.	The tracability matrix and a link to the presentations for build 2b has been posted to the PDS EN website under build 2b ORR

RFA_Build2a_028	A'Hearn	Requirements	I and others have lost track of the suite of requirements.	Provide a traceability matrix between requirements and builds.	DELIVERED	Build 2b		Tracability Matrix posted to the EN site will be linked to this RFA response.	The tracability matrix and a link to the presentations for build 2b has been posted to the PDS EN website under build 2b ORR
RFA_Build2a_029	Anne Raugh	SBNUMD50: Errors in PDS2010 documentation	At least two actual errors were encountered in the documentation for the registry, harvest and validate utilities: bad syntax for a "ping" command which caused a spurious error message; and directions to rename a directory that was, in fact, correctly named. In addition, there were multiple cases of instructions to rename things that did need to be renamed - but this is something that should have been done in the software prior to release!	Correct documentation.	DELIVERED	Build 2b		The software documentation will be updated to correct each issue found during the Build 1d test exercise at the Nodes as well as any additional issues encountered in Build 2a.	The updated documentation is available in the Build 2b software release available on the PDS EN site at http://pds-engineering.jpl.nasa.gov/pds2010/development/2.1.0/ .
RFA_Build2a_030	Anne Raugh	SBNUMD33: No standards versions in generic schemas	The generic schemas do not contain either data dictionary or standards reference version numbers. Since the schemas are dependent on the standards and are contained in the data dictionary, this information should be included in the schemas, even if it is possible for users to update it later.	Insert current dictionary and standard version numbers into all generic schema as they are generated	DELIVERED	Build 2b		Implement the preamble recommendations from ACR's IM review.	The information_model_version attribute now requires the version number of the information model. An "assert" statement validates the value.
RFA_Build2a_031	Anne Raugh	SBNUMD12: Generic Schemas contain local references	Every generic schema I opened had a local file reference in the schemaLocation attribute. This is not transportable or viable in the archive, or even with the disk organization mandated by the SR. These locations need to be relocatable - URL-style identifiers, for example.	investigate recommendations by standards organizations for creating transportable schema. Adopt generic schemaLocation values that support these conventions and promote long-term archive stability. Document the rules PDS follows and require data designers to do the same in their tailored and specific schema. Include a test for relocatable schemaLocation values in all schema archived in the PDS.	DELIVERED	Build 2b			Local references have been removed from the master (aka extension) schemas.
RFA_Build2a_032	Anne Raugh	SBNUMD04: Version number in namespace strings	Namespaces should not have versioning information encoded into them (e.g., the "v05" in the PDS4 namespace of this test). This undermines the long-term stability and usability of the schemas. It also makes me think there is no configuration control in effect.	If there is, in fact, no configuration control - that is, no separation between development and production environments - institute it and use it. Remove versioning information from all namespaces and reference only the production environment in generic schemas.	CLOSED	Build 2b	The namespace review team has recommended that the Version be indicated in a schema namespace using Vn, where n is the major version number. This recommendation has been reviewed and accepted by the DDWG.	Temporarily before the release of Version 1.0, the version id V0n is being used to indicate the beta version release, for example v06.	No Change
RFA_Build2a_033	King	Document Schemas	With the "master" schema it is possible to define a valid document with undesirable root elements. The preferred root elements needs to be explicitly defined to avoid misuse of the master (type) schema.	Define a top level document schema to reduce the number of specific or local schema. A top level document schema can be used for most data products descriptions and make the overall process of generation and validations more lightweight and less costly.	DELIVERED	Build 2b		Define top level "elements".	Top level elements have been defined. A root element has been included in Build 2b as a prototype for testing. The final decision will be made after build 2.
RFA_Build2a_034	King	File Area Observational	Elements under File_Area_Observational must be described in alphabetical order. This is a very arbitrary. The ability to define the elements in logical order will make the documents easier to evaluated and construct by a user.	This change will allow selected elements to appear in any order rather than the current alphabetical order. This change addresses a potential cost driver. This change will substantially ease migration and contain costs because it allows a more convenient mapping of current PDS3 labels to PDS4 equivalents. Without this change additional processing of metadata during the migration will be necessary in order to produce compliant PDS4 XML documents. It will also ease the burden on data engineers while inspecting transformed documents and performing validation.	DELIVERED	Build 2b		Implement the xsd:choice construct to allow the data objects to be listed in any order.	Implemented the xsd:choice construct to allow the data objects to be listed in any order.

SUMMARY	
OPEN	0
PENDING	0
DELIVERED	29
CLOSED	3
Total	32