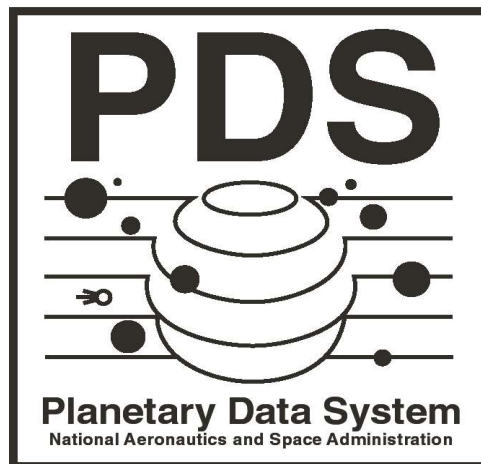


The Planetary Data System

System Requirements Document (SRD)

February 6, 2004

Version 1.2



Jet Propulsion Laboratory
Pasadena, California

JPL D-xxxx

Planetary Data System

System Requirements Document (SRD)

February 6, 2004

Version 1.2

Approved:

LaVerne Hall
PDS Project Manager

Date

Reta Beebe
PDS Project Scientist

Date

Bill Knopf

Date



Jet Propulsion Laboratory
Pasadena, California

JPL D-xxxx

CHANGE LOG

Revision	Date	Description	Author
Candidate Requirements	Oct 21, 2002 – March 25, 2003	Gathered candidates from PDS discipline nodes	PDS DN/CN Collaboration
Binned Requirements	April 18, 2003	Binned requirements list	CN SE Team
Working List of Requirements 1	July 23, 2003	Organized and listed requirements	CN SE Team
Working List of Requirements 2	Sept 8, 2003	Revised and edited list from requirements workshop	Working Group
Release Draft 1.0	10/31/03	First complete draft	J. Wilf
Release Draft 1.1	11/17/03	Draft for MRO Review	J. Wilf
Release Draft 1.2	02/06/04	Revised for MRO	J. Wilf

TABLE OF CONTENTS

Section	Title	Page
1.0 INTRODUCTION		1
1.1	Purpose	1
1.2	Scope	1
1.3	Audience	2
1.4	History	2
1.5	Controlling Documents	3
1.6	Applicable Documents	3
1.7	Document Maintenance	3
2.0 SYSTEM DESCRIPTION		4
2.1	Definitions	4
2.2	Core vs. Extended System	5
3.0 FUNCTIONAL REQUIREMENTS		5
3.1	Ingest Requirements	6
3.2	Validate Requirements	7
3.3	Track Requirements	10
3.4	Search Requirements	11
3.5	Display Requirements	13
3.6	Retrieve Requirements	13
3.7	Geometry Requirements	15
3.8	Data Mining Requirements	16
3.9	Process Requirements	16
3.10	Transfer Requirements	18
3.11	Notify Requirements	18
3.12	Administer Requirements	19
4.0 NON-FUNCTIONAL REQUIREMENTS		20
4.1	Architecture Requirements	20
4.2	Interface Requirements	21
4.3	Capacity Requirements	22
4.4	Performance Requirements	22
4.5	Availability Requirements	22
4.6	Security Requirements	23
4.7	Compatibility Requirements	24
4.8	Standards Support Requirements	24
4.9	Policy Support Requirements	25
4.10	User Support Requirements	26
4.11	Documentation Requirements	26
4.12	Development Requirements	27
5.0 PRIORITIZATION		28

6.0 IMPLEMENTATION.....	28
6.1 Existing Implementation.....	28
6.2New Implmentation.....	30
7.0 TEST METHODS AND ACCEPTANCE CRITERIA.....	30
8.0 CHANGE REQUESTS AND USER SCENARIOS.....	30
APPENDIX A ACRONYMS.....	31
APPENDIX B GLOSSARY.....	33
APPENDIX C TRACEABILITY MATRIX.....	41
APPENDIX D PRIORITY MATRIX.....	48

1.0 INTRODUCTION

1.1 Purpose

The purpose of this document is to provide requirements to the Planetary Data System (PDS), to guide the development of its online information system through the next five years. These requirements will help the PDS meet near-term challenges, including:

- More complex missions and instruments
- Significantly higher data volumes
- User demands to improve data access and useability
- Science that requires correlation across disparate data sets

Ultimately, every requirement included here should improve the ability of the PDS to archive and distribute planetary science data, in accordance with its charter.

1.2 Scope

This document addresses two types of requirements. The first are functional requirements. These define system capabilities – functions that the system must be able to perform. Each functional requirement applies to one of the twelve general system functions: Ingest, Validate, Track, Search, Display, Retrieve, [Calculate] Geometry, [Perform] Data Mining, [Run] Process, Transfer, Notify, and Administer. The requirements for each of these functions are described in detail in **Section 3**.

This document also includes non-functional requirements. These define system characteristics and constraints. The non-functional requirements are divided into twelve general areas of concern: Architecture, [User] Interface, Capacity, Performance, Availability, Security, Compatibility, Standards [Support], Policy [Support], [User] Support, Documentation, and Development. The requirements for each of these areas are described in detail in **Section 4**.

The remaining sections provide additional information and context. **Section 2** defines key terminology and describes the overall system. **Section 5** shows how the requirements are prioritized. **Section 6** discusses the implementation method. **Section 7** describes how the system will be tested to ensure that the requirements have been met. Finally, **Section 8** shows how new requirements will be added through change requests and user scenarios.

Further details can be found in the appendices, including acronyms, glossary, a prioritization matrix, and a matrix that traces requirements to their sources.

1.3 Audience

This document is written primarily for PDS developers, who will use the requirements to guide the design and implementation of the system. The expected audience also includes:

- PDS project management
- PDS discipline and data node management
- PDS science data users
- Project personnel who archive data with the PDS
- NASA personnel at HQ

1.4 History

This document is the result of a requirements gathering effort that began after the delivery of the first online PDS Distribution system (PDS-D), in October, 2002. PDS-D was designed to distribute data from geographically separated repositories, and it successfully met the demands of the Mars Odyssey mission. However, meeting the challenges listed in **Section 1.1** – more complex missions, higher data volumes, better usability, and the need to correlate data across data sets – called for a larger system, both in scale and capability. So the PDS began gathering requirements for the “next” online system.

The process began with the PDS discipline nodes, who generated “candidate requirements” representing the needs of both the data providers and the science user community. Then the PDS central node system engineers captured the candidate requirements in a database, sorted them into categories, and organized them into a draft requirements list.

A PDS-wide Requirements Working Group (RWG) was formed, which discussed and edited the requirements list. The list approved by the RWG is the basis for **Section 3** and **Section 4** of this document.

Concurrently, Dr. Ray Arvidson of the PDS Geosciences node, led an effort to define scenarios for using Mars Reconnaissance Orbiter (MRO) data to achieve scientific objectives. As **Section 8** describes, these user scenarios – and others to come – provide a means for adding new requirements to the system.

1.5 Controlling Documents

[1] Planetary Data System (PDS) Standards Reference, August 1, 2003, Version 3.6, JPL D-7669, Part 2.

[2] Planetary Science Data Dictionary Document, August 28, 2002, Planetary Data System (PDS), JPL D-7116, Rev E.

[3] Planetary Data System (PDS) Archive Life Cycle, In Preparation.

[4] User Scenarios Associated With Searching for and Retrieving Mars Reconnaissance Orbiter and Associated Data in the Planetary Data System, Ray Arvidson and Jim Murphy, Version 1.1, August 22, 2003.

[5] PDS-MRO Delta Peer Review Presentation, January 21, 2004.

1.6 Applicable Documents

[6] Planetary Data System (PDS) Proposers Archiving Guide, June 5, 2003, Version 1.8.

[7] PDS Data Distribution System (PDS-D) Functional Requirements Specification for Delivery #1 (D01), March 5, 2002, Version 1.2.

[8] Architectural Styles and the Design of Network-based Software Architectures, Roy Thomas Fielding, Dissertation, University of California, Irvine, 2000.

1.7 Document Maintenance

During the period covered by this document, new missions and user scenarios from the science community may require new capabilities of the system. This document will be updated periodically to reflect any changes or additions.

2.0 SYSTEM DESCRIPTION

2.1 Definitions

In order to specify requirements properly, definitions of technical and organizational terms have been agreed upon. A complete glossary of these terms is given in **Appendix B**. The terms that are essential for a high-level description of the system are defined in this section.

First, it is necessary to define “the system” itself. All requirements in this document apply to the system, as defined here:

the System (capitalized) – The integrated hardware and software used to support the essential functions of the PDS.

The next definitions help define the contents of the System in a generalized way:

attribute – A property of an object that has a name and a value. For example, the Cassini mission has an attribute named "MISSION_NAME" with the value "Cassini."

resource – Anything that can be described by a set of attributes. Resources in the PDS include data, software, documents, web services, spacecraft, and personnel.

retrievable resource – A resource that can be represented digitally, stored on media, and made accessible to users. When the context is clear, this document refers to retrievable resources simply as **resources**.

resource description – The set of attributes describing a resource. In the PDS, all attributes must be defined in the Planetary Science Data Dictionary (PSDD).

All the elements specified in the PDS Standards [1] as part of a PDS archive volume are retrievable resources or resource descriptions. The retrievable resources include data products, calibration, browse and display files, software, and documentation. Resource descriptions are found in PDS labels, catalog files, and index files.

Generally speaking, all the elements found on a PDS archive volume, will be found on The System in the following places:

repositories – Mountable computer storage that contains retrievable resources.

catalogs – A databases containing an organized collection of attributes, which are used to select and locate resources.

the Data Set Catalog (capitalized) – A special catalog containing the attributes of all PDS data sets and the resources associated with those data sets.

With these definitions in place, the System can be described in terms of the types of resources found on the System, how the System handles those resources, and how the users interact with them.

2.2 Core vs. Extended System

The System can be considered as consisting of two parts:

Core System – Supports only the essentials of distribution and archiving. It ensures that PDS products are properly stored in the system and that they can be found through attribute searches, downloaded by the user, and transferred to physical media for long-term archiving.

Extended System – Improves the usability of the system and data, especially for science users. It calls for advanced interfaces, correlative search across data sets, geometric calculations, on-the-fly data processing, and the use of information embedded in the data.

Dividing the System into Core and Extended components is useful in both prioritizing requirements and managing development. In the sections that follow, requirements for the Extended System are distinguished by an “**EXT**” appended to the requirement identifier.

3.0 FUNCTIONAL REQUIREMENTS

FR.SYS.1 **The System** shall provide the following functions:

- | | |
|--------------|---|
| 3.1 Ingest | ING – Receive, accept, store, and catalog resources |
| 3.2 Validate | VAL – Determine whether resources are PDS compliant |
| 3.3 Track | TRK – Track resources |
| 3.4 Search | SCH – Find resources |
| 3.5 Display | DSP – Display resources |

3.6 Retrieve	RET – Download resources
3.7 Geometry	GEO – Calculate geometric information
3.8 Data Mining	DMG – Extract information from data resources
3.9 Process	PRC – Execute computer code using resources as input
3.10 Transfer	TRN – Transfer resources to other systems or media
3.11 Notify	NOT – Inform users of new or updated resources
3.12 Administer	ADM – Monitor and update the state of the System

3.1 Ingest Requirements

The **Ingest Function** receives, accepts, stores, and catalogs resources. This makes the resources available to the other system functions.

FR.ING.1 The **Ingest Function** shall be able to receive resources and resource descriptions submitted electronically.

FR.ING.2 The **Ingest Function** shall be able to receive resources and resource descriptions submitted on the following physical media:

- a) External hard disks with EIDE, SCSI, USB, or firewire interfaces
- b) Internal hard disks with EIDE or SCSI interfaces
- c) CD-ROM, CD-R, and DVD-R media
- d) DLT tape

FR.ING.3 The **Ingest Function** shall be able to receive and update resource descriptions submitted independently of the resource itself.

FR.ING.4 The **Ingest Function** shall enable the PDS to accept or reject any resource or resource description it receives. Only accepted resources and resource descriptions are stored or cataloged on the System.

FR.ING.5 The **Ingest Function** shall store all accepted resources. This means placing them on mountable computer storage media.

FR.ING.6 The **Ingest Function** shall catalog all accepted resource descriptions. This means saving the contents of the resource descriptions in a database.

FR.ING.7 The **Ingest Function** shall ensure that for every resource stored online, there is a cataloged resource description that includes a pointer to the online location of the resource.

FR.ING.8 The **Ingest Function** shall enable the PDS to remove resources and resource descriptions from the System.

FR.ING.9 The **Ingest Function** shall be able to obtain resource descriptions from the following standard PDS file types, and use this information to update the following types of catalogs:

- a) Product labels - to update product catalogs
- b) Index tables - to update product catalogs
- c) Catalog files - to update the Data Set Catalog
- d) Data dictionary element definitions - to update the main Planetary Science Data Dictionary and local data dictionaries

FR.ING.10 The **Ingest Function** shall provide an online interface for creating and submitting resource descriptions for the Data Set Catalog.

FR.ING.11 The **Ingest Function** shall provide an online interface for creating and submitting data dictionary element definitions for the main Planetary Science Data Dictionary and local data dictionaries.

FR.ING.12 The **Ingest Function** shall provide the capability to define an association between any resource and a set of other resources. *For example, a calibration program can be associated with a set of data products and calibration files.*

FR.ING.13 The **Ingest Function** shall provide a suite of tools for mission archive production, able to do the following:

- a) Generate index tables from a set of product labels
- b) Generate the labels for index tables
- c) Generate catalog files from information in the Data Set Catalog
- d) Generate product labels from an index table
- e) Generate product labels by merging source data into a label template
- f) Interactively create a product label from a label template

3.2 Validate Requirements

The **Validate Function** determines whether a given resource or resource description is PDS compliant (or "valid"). In other words, the Validate Function tests the resource or resource description to decide whether it satisfies all applicable PDS standards. The standards themselves are defined in the **PDS Standards Reference** [1]. Not every type of resource can be validated, only those for which there are testable PDS standards.

Also note that validation does not ensure that a resource/description is scientifically accurate or useful to the planetary science community. Those issues are decided through the peer review process.

FR.VAL.1 The **Validate Function** shall be able to determine whether any resource description received by the PDS is valid, by testing whether the following are true:

- a) All attributes in the resource description are in the PSDD and have valid values, according to the PSDD
- b) All attributes in the resource description have valid values according to the PSDD
- c) If the resource description describes a resource stored on the System, there is a pointer to the location of the resource
- d) If the resource description describes a resource stored on the System, it properly describes the file format and file properties of the resource (for example, for a binary table, the rows, columns, and start bytes must be given correctly)

FR.VAL.2 The **Validate Function** shall be able to determine whether a PDS label is valid by testing whether the following are true:

- a) The label is valid as a resource description, according to the tests described in **FR.VAL.1**.
- b) The label file has valid syntax according to the grammar of the Object Description Language (ODL)

FR.VAL.3 The **Validate Function** shall be able to determine whether a PDS catalog file is valid by testing whether the following are true:

- a) The catalog file is valid as a PDS label, according to the tests described in **FR.VAL.2**.
- b) The catalog file is formatted correctly, according to the PDS Standards Reference [1]
- c) The resource description in the file can be cataloged in the Data Set Catalog with without violating the referential integrity of the Data Set Catalog.

FR.VAL.4 The **Validate Function** shall be able to determine whether a PDS index table is valid by testing whether the following are true:

- a) The tables are properly formed as fixed-field, ASCII tables
- b) Every row in the index table corresponds to a PDS data product
- c) Every row in the table contains one or more fields that point to the location of the product on the System

FR.VAL.5 The **Validate Function** shall be able to determine whether a PDS volume is valid by testing whether the following are true:

- a) The volume has one of the standard directory structures given in the PDS Standards Reference [1]
- b) All required directories and files are present and named according to the naming standards given in the PDS Standards Reference [1]
- c) All label files are valid, according to the tests described in FR.VAL.2
- d) The catalog files are valid, according to the tests described in FR.VAL.3
- e) All index files are valid, according to the tests described in FR.VAL.4
- f) Every file on the volume is pointed to by a PDS compliant label
- g) Every data product on the volume has an entry in the index table
- h) All the text files on the volume are properly formed (for example, with no illegal characters and abiding by line length standards)

FR.VAL.6 The **Validate Function** shall be able to determine whether a PDS data set release is valid by testing whether the following is true:

- a) All the volumes comprising the release are valid, according to the tests described in **FR.VAL.5**
- b) The release has a valid resource description, as defined in the DATA_SET_RELEASE object in the PSDD

FR.VAL.7 The **Validate Function** shall provide a web-based interface that enable users to run all validation tests on resources stored on remote repositories.

FR.VAL.8 The **Validate Function** shall provide a set of validation tools that enable users to run all validation tests on resources stored locally, on any PDS-supported platform, in one of the following modes:

- a) Interactively, through a graphical user interface.
- b) In batch mode, through a command line interface.

FR.VAL.9 The **Validate Function** shall enable users to specify whether validation tests return a simple pass/fail indication or a detailed summary report.

FR.VAL.10 The **Validate Function** shall enable users to specify one or more email addresses to which the results of validation tests will be automatically forwarded.

FR.VAL.11 The **Validation Tools** that use the PSDD shall be able to select whether to use only the global data dictionary, or to include any of the available local data dictionaries.

3.3 Track Requirements

The **Track Function** reports the status of a resource – the values of attributes representing the state of a resource at a given point in time. Any set of attributes can be used for this purpose and thus "tracked" by the System. However, a special RESOURCE_STATUS is defined in order to track the ingestion and availability of resources in general.

FR.TRK.1 The **Track Function** shall be able to report the RESOURCE_STATUS of any resource submitted to the PDS, as having one of the following values:

- a) Expected - PDS expects to receive the resource
- b) Received - PDS has received the resource
- c) Accepted - PDS has agreed to ingest the resource into the System
- d) Rejected - PDS has decided NOT to ingest the resource into the System
- e) Withdrawn - The submitter has decided they DON'T want the resource ingested
- f) Available - The resource has been stored and its description cataloged
- g) Superseded - A new version of the resource is available
- h) Removed - The resource has been removed from the System

FR.TRK.2 The **Track Function** shall alert the submitter by email when a submitted resource has been assigned a status of Received, Accepted, Rejected, Available, Superseded, or Removed.

FR.TRK.3 For every active mission, the **Track Function** shall be able to report the expected and actual completion dates for the following archive planning milestones (described in the PDS Archive LifeCycle [3]):

- a) The Mission has received a PDS orientation
- b) Mission archive funding is in place
- c) The Memorandum of Understanding (MOU) between the Mission and the PDS has been signed
- d) Data archive working groups have been formed
- e) A Draft Data Management Plan (DMP) or Archive Plan (AP) has been approved
- f) All Interface Control Documents (ICDs) have been received
- g) All Data Set information "skeletons" have been cataloged

FR.TRK.4 For every data release, the **Track Function** shall be able to report the following properties of the release:

- a) Data Set ID

- b) Release ID
- c) Release version ID
- d) Release description
- e) Data provider (i.e., instrument team)
- f) Data distributor (i.e., PDS data node, discipline node, or central node)
- g) RESOURCE_STATUS of the release
- h) ARCHIVE_STATUS of the release
- i) Whether the release has been validated
- j) Expected delivery date (when it should be received by the PDS)
- k) Actual delivery date
- l) Expected release date (when it should be available to the community)
- m) Actual release date
- n) Number of files in the release
- o) Size of the release in bytes
- p) Product type(s) in the release
- q) Release coverage (e.g., a list of start/stop times)
- r) Liens against the release
- s) Release errata

FR.TRK.5 The **Track Function** shall alert the data provider by email when the due date for a data delivery is two weeks away and when it has passed without receipt of data.

FR.TRK.6 The **Track Function** shall provide an online mechanism for generating a list of all the data deliveries and releases that are late.

FR.TRK.7 The **Track Function** shall provide an online mechanism for generating reports showing all the data releases for a given mission, instrument, data set, or within a given time frame.

3.4 Search Requirements

The **Search Function** locates resources based on their cataloged attributes.

FR.SCH.1 The **Search Function** shall provide the ability to locate any resource on the System based on its cataloged attributes.

FR.SCH.2 The **Search Function** shall be able to take as input any set of attributes defined in the PSDD as a search criteria and return the following:

- a) Set of resource descriptions that match the criteria
- b) Set of resources whose descriptions match the criteria
- c) The number of resources matching the criteria

FR.SCH.3 The **Search Function** shall provide a mechanism for locating data sets through the following set of attributes:

- a) Target
- b) Mission
- c) Instrument host (e.g., spacecraft)
- d) Instrument name
- e) Instrument type
- f) Data object type (e.g., image)
- g) Curating node
- h) Start/stop times
- i) Data set ID
- j) Citation ID

FR.SCH.4-EXT The **Search Function** shall provide the capability to determine whether a search is correlated. That is, the search function will be able to determine if an attribute to be used in the search has the exact same meaning across all contexts involved in the search.

FR.SCH.5-EXT The **Search Function** shall support searches across all NAIF-supported coordinate systems. *See Geometry requirements.*

FR.SCH.6 The **Search Function** shall provide a method for locating resources that are associated with a selected resource. *Note: Associations between resources are created through the **Ingest Function**.*

FR.SCH.7 The **Search Function** shall be able to perform substring searches on values of character keywords.

FR.SCH.8 The **Search Function** shall be able to return resource descriptions given a PDS compliant bibliographic citation of that resource (typically a data set or product). In addition, this capability shall be associated with a permanent URL, which can then be given as part of the citation.

The following requirements are specific to resource type or data submitter (i.e., these are domain-specific searches):

FR.SCH.9-EXT The **Search Function** shall enable searching for landing sites or observational target areas using attributes considered high priority according to program or mission goals, for example those identified by the Mars Exploration Payload Analysis Group (MEPAG).

FR.SCH.10-EXT The **Search Function** should be able to filter search results on rings for a variety of geometric factors:

- a) Longitude relative to periapse

- b) Proximity to the planet's shadow
- c) Proximity to the ansa
- d) For a specified phase range
- e) Separation from a known body

FR.SCH.11-EXT The **Search Function** will provide a standard utility to accept a specific instrument, instrument host, and time window, return a list of data products that fall within the time window.

FR.SCH.12-EXT The **Search Function** will provide a standard utility to accept a specific instrument, instrument host and spatial window on a target body, return a list of data products that fall within the spatial window.

3.5 Display Requirements

The **Display Function** provides representations of resources in the system, which can be viewed by user interfaces.

FR.DSP.1 The **Display Function** shall be able to display any resource description in the System.

FR.DSP.2 The **Display Function** shall provide method(s) for displaying any resource in the System. *This means providing a representation of the resource , which can be displayed, for example the browse version of an image.*

FR.DSP.3 Where the resource is a repository, the **Display Function** shall provide a mechanism to navigate through the directories of a repository and display individual files.

FR.DSP.4 The **Display Function** shall provide an online mechanism for previewing a representation of the data in order to determine whether it is appropriate for further actions, such as downloading.

FR.DSP.5 The **Display Function** shall have the capability to generate a default web interface to the data when new data are delivered to the PDS.

3.6 Retrieve Requirements

The **Retrieve Function** makes a located resource available to the user locally.

FR.RET.1 The **Retrieve Function** shall provide a mechanism to deliver any collection of resources stored on the System.

FR.RET.2 The **Retrieve Function** shall enable the user to download resources through a web browser, provided the size of the download does not exceed a System-defined size limit.

FR.RET.3 The **Retrieve Function** shall be able to deliver resources to the user through one of the following methods, subject to PDS policy on their use:

- a) Download through a Java client that connects directly to the repository
- b) Delivery on any physical media supported by the PDS

FR.RET.4 The **Retrieve Function** shall inform the user if the delivery size exceeds the System-defined size limit and enable the user to order resource delivery through the methods given in **FR.RET.3**.

FR.RET.5 The **Retrieve Function** shall inform the user of the estimated download size and allow the user to cancel the download if the user deems it too large.

FR.RET.6 The **Retrieve Function** shall accept a unique resource identifier and deliver the resource, regardless of its location on the system.

FR.RET.7 The **Retrieve Function** shall enable the automatic retrieval of all files that are part of a multi-file resource.

FR.RET.8 The **Retrieve Function** shall enable the automated packaging of a resource and all associated resources for delivery in one of the following formats:

- a) ZIP
- b) TAR
- c) Gzipped TAR

FR.RET.9 The **Retrieve Function** shall enable the building of PDS compliant volumes, combining selected resources with others required on a PDS compliant volume.

FR.RET.10 The **Retrieve Function** shall enable the user to create and download standard PDS catalog files based on the latest information in the PDS Data Set Catalog.

FR.RET.11-EXT The **Retrieve Function** shall enable the reformatting of a data product to any of the supported set of formats appropriate to that data type. *Example formats include: simple binary tables with fixed-length fields, simple ASCII tables, simple images.*

FR.RET.12-EXT The **Retrieve Function** shall support the delivery of a user defined subset of a data product as a PDS-labeled product.

3.7 Geometry Requirements

The **Geometry Function** refers to the discovery, calculation and use of the geometric relationships between instrument hosts, instruments, and targets of observation. Geometric operations include coordinate conversion, calculation and recalculation of ephemerides, time calculations as well as computing basic location, pointing and orientation information.

Notes:

1) *These requirements only apply to those those PDS data sets, instruments, instrument hosts and targets that are supported by NAIF SPICE kernels and software.*

2) *For the purposes of the following requirements, a "target" may be a body, a feature on a body, or any area or phenomenon of interest for which SPICE support or gazetteer type information is available.*

3) *A spatial window is an area defined on a target body or the celestial sphere. On the target body it is defined in terms of an appropriate surface or host-centered coordinate system. On the celestial sphere it is defined in an appropriate celestial coordinate system.*

FR.GEO.1 The **Geometry Function** shall be able to recalculate indexed geometric values produced using archived software when updated SPICE kernels become available.

FR.GEO.2-EXT The **Geometry Function** shall be able to accept a specific instrument, instrument host, and time window and return a list of known (i.e., NAIF-supported) targets within the instrument field of view.

FR.GEO.3-EXT The **Geometry Function** shall be able to transform coordinates in any format between any two supported reference frames.

FR.GEO.4-EXT The **Geometry Function** shall be able to transform a given time between standard time systems. *NOTE: NAIF provides the CHRONOS utility, which can be run on the client side to accomplish this.*

FR.GEO.5-EXT The **Geometry Function** shall be able to accept a specific instrument, instrument host, and a three-dimensional range with respect to a specific coordinate system and return the time range(s) when the instrument host is within the specified range.

FR.GEO.6-EXT The **Geometry Function** will provide a standard utility to accept

a specific instrument, instrument host, time window and target, return a list of data products for which the target lies within the field of view.

3.8 Data Mining Requirements

FR.DMG.1-EXT The **Data Mining Function** shall provide the capability to extract elevation information from MGS MOLA data.

3.9 Process Requirements

The **Process Function** enables the execution of routines using resources and resource descriptions as input. The Process requirements specify how the routines are invoked and executed, and which set of standard routines must be provided by the System.

FR.PRC.1-EXT The **Process Function** shall provide a mechanism for running routines on any collection of resources or resource descriptions, performing the appropriate set of transformations or manipulations, and returning the results.

FR.PRC.2-EXT The **Process Function** shall provide a mechanism for adding user-created routines to the ones offered by the System.

FR.PRC.3-EXT The **Process Function** shall enable routines to be invoked remotely through the following methods:

- a) XML-encoded message over HTTP
- b) URL-encoded query over HTTP
- c) Java API through RMI (Remote Method Invokation)

FR.PRC.4-EXT The **Process Function** shall supply local processes for transforming the following PDS data to the given formats.

- a) PDS image and table data to FITS format
- b) PDS image data to PNG and JPEG formats
- c) Binary tables to ASCII tables

FR.PRC.5-EXT The **Process Function** shall supply local processes for transforming PDS image and QUB data as follows:

- a) Geometrically correcting images
- b) Radiometrically correcting images
- c) Mapping images to any PDS-supported coordinate system
- d) Re-projecting images to any PDS-supported map projection

- e) Applying transformations available from the ISIS system

FR.PRC.6-EXT The **Process Function** shall supply local processes for subsetting the following data types:

- a) Selecting individual bands from spectral QUB data
- b) Selecting rectangular sub-regions of images
- c) Selecting specified rows and columns from tabular data
- d) selecting specified intervals from time series

FR.PRC.7-EXT Where the resources is an image, the **Process Function** shall enable the following image display controls:

- a) Contrast stretch
- b) Zoom and pan
- c) Histogram display
- d) Line/sample location and DN value under cursor

FR.PRC.8-EXT The **Process Function** shall provide a mechanism for overlaying image data with the following:

- a) Textual information
- b) Feature information
- c) Topographic information
- d) Ground tracks of other data
- e) Latitude/longitude grids

FR.PRC.9-EXT The **Process Function** shall provide a mechanism for searching and retrieving information necessary for registering all geo-referenced data for the same planetary target to the same, user-selected coordinate system.

FR.PRC.10-EXT The **Process Function** shall provide a standard utility to take the name of an attribute and indices, and indicate:

- a) If that attribute appears in both indices
- b) If the attribute values in both indices be directly compared (i.e. without conversion)

The following are domain or mission-specific processes:

FR.PRC.11 The **Process Function** shall provide a standard utility for subsetting and creating reduced-resolution images, by supplying the appropriate coefficients to JPEG2000 images.

3.10 Transfer Requirements

The **Transfer Function** makes located resources available to remote systems or on physical media. This is server-to-server or server-to-media transfer, as opposed to the **Retrieve Function**, which is restricted to sending resources from the server tier to the user.

FR.TRN.1 The **Transfer Function** shall provide, on request, a mechanism for writing any collection of data products and ancillary files onto physical media.

FR.TRN.2 The **Transfer Function** shall provide, on request, a mechanism for generation of PDS compliant archive volumes onto physical media from a PDS repository.

FR.TRN.3 The **Transfer Function** shall provide a mechanism for delivering the PDS compliant archive volumes covering each data set to the following organizations:

- a) Central node
- b) The discipline node curator for that data set
- c) NSSDC

FR.TRN.4 The **Transfer Function** shall enable full or partial mirroring of all system repositories.

3.11 Notify Requirements

The **Notify Function** notifies users that resources, to which they have subscribed, have been added or changed.

FR.NOT.1 The **Notify Function** shall provide an online mechanism for creating subscriber accounts, with each account requiring the following attributes:

- a) User full name
- b) User last name
- c) Password
- d) Institution
- e) Work telephone
- f) Email address
- g) Mailing address
- h) Whether user is funded by NASA Code S

FR.NOT.2 The **Notify Function** shall provide an online mechanism for users to

subscribe to receive email notifications of any of the following events:

- a) Data release for a subscribed instrument
- b) Release of a new version of the PDS standards
- c) Release of a new build of the PSDD
- d) Release of a new version for any PDS supported software
- e) General PDS news and announcements

FR.NOT.3 The **Notify Function** shall provide an online mechanism for subscribers to change their account information, change their notification requests, delete their accounts, and cancel their subscriptions.

FR.NOT.4 The **Notify Function** shall maintain a web site to which subscribers may be referred for detailed information.

FR.NOT.5 All notifications will be logged and will be accessible for viewing by subscribers.

3.12 Administer Requirements

The **Administer Function** enables system administrators to monitor, update, backup, and perform other housekeeping tasks on the system.

FR.ADM.1 The **Administer Function** shall provide a mechanism for monitoring the availability and performance of every server on the system.

FR.ADM.2 The **Administer Function** shall provide a mechanism for capturing, displaying, and summarizing the following metrics from all possible delivery and feedback mechanisms:

- a) Number of user accesses, per server and time period
- b) Number of data requests, by repository
- c) Number of bytes transferred, by repository
- d) Transfer performance, by server and time period, including bandwidth usage and average load over time
- e) Number and type of support requests and user comments

FR.ADM.3 The **Administer Function** shall provide a mechanism for updating any system component.

FR.ADM.4 The **Administer Function** shall provide a mechanism for rolling back changes to any system component.

FR.ADM.5 The **Administer Function** shall provide mechanisms to ensure that

mission archives are always preserved in their entirety, from the time they are received by the PDS, including:

- a) Automated, scheduled backup for all repositories
- b) Automated, scheduled backup for all catalogs
- c) Offsite storage of backup or archive media

FR.ADM.6 The **Administer Function** shall maintain an inventory of all deep archive volumes.

4.0 NON-FUNCTIONAL REQUIREMENTS

NR.SYS.1 The **System** shall be characterized by the following non-functional requirements:

- | | |
|--------------------|---|
| 4.1 Architecture | ARC - system architecture |
| 4.2 Interface | INT - user interface and look and feel |
| 4.3 Capacity | CAP - storage needs |
| 4.4 Performance | PER - data access, processing, and transfer |
| 4.5 Availability | AVA - uptime requirements |
| 4.6 Security | SEC - security requirements |
| 4.7 Compatibility | CMP - platform support and interoperability |
| 4.8 Standards | STD - standards to support system operation |
| 4.9 Policies | POL - policies to support system operation |
| 4.10 Documentation | DOC - documentation required for the system |
| 4.11 Support | SUP - technical and user service support |
| 4.12 Development | DEV - development and system evolution |

4.1 Architecture Requirements

Architectural Requirements cover characteristics and constraints on the overall construction of the system.

NR.ARC.1 The **System** shall have an open architecture, where all changes and enhancements can be made in a collaborative manner, and all Application Programming Interfaces (APIs) are fully documented and shared.

NR.ARC.2 The **System** shall have non-proprietary core components. This includes all components that enable communication and transfer of data from one location to another.

NR.ARC.3 The **System** shall be modular and capable of being built

incrementally.

NR.ARC.4 The **System** shall be service based, where each service can be built independently of each other and are capable of evolving independently.

NR.ARC.5 The **System** shall be a distributed system. It shall provide the ability to locate data, resources, and services from multiple physical locations.

4.2 Interface Requirements

Interface Requirements cover user interface and look-and-feel issues.

NR.INT.1 The **System** shall provide a web-based interface as the primary means through which users interact with the system. A user shall be able to access all the functions and services through a common web browser.

NR.INT.2 The **System** shall provide an Application Programming Interface (API) as a means for developers to access all functions and services provided by the system.

NR.INT.3 The **User Interface** shall provide access to online help for all functions, as well as easy access to all applicable documentation.

NR.INT.4 The **User Interface** shall provide intuitive interaction with the system. This means that most users should be able to figure out how to use basic system functions without reading further documentation.

NR.INT.5 The **User Interface** shall provide "context sensitive" searching for data sets. That is, once the user sets a value for a search parameter, the available values for the other parameters must be consistent with the first parameter/value pair. *For example, if TARGET is set to "JUPITER", then the interface will not allow MISSION to be set to "PATHFINDER", since PATHFINDER did not observe Jupiter.*

NR.INT.6 All **web form** input shall be validated to reduce the possibility of erroneous results.

NR.INT.7 All **web pages** and email announcements shall appear neat and of professional quality.

NR.INT.8 The **System** shall promote a common look-and-feel for the web interface by providing interface elements to developers of data set interfaces.

4.3 Capacity Requirements

Capacity Requirements cover the storage needs of the system.

NR.CAP.1 The **System** shall provide online storage for 120 TB of data and ensure that all data are accessible to all system functions and services.

NR.CAP.2 The **System** shall provide local backup storage for 120 TB of data, which may be restored in the event of data loss from the primary storage system.

NR.CAP.3 The **System** shall provide geographically separated backup storage for 120 TB of data, which may be restored in the event of data loss from both the primary storage system and the local backup system.

4.4 Performance Requirements

Performance Requirements specify the speed and throughput characteristics for all data transfers.

NR.PER.1 The **System** shall be able to receive and ingest 5 TB of data over a two-week period. (*Note: This is derived from the maximum expected MRO data volume per delivery during the nominal mission.*)

NR.PER.2 The **System** shall improve performance of internet-based data distribution by providing load-balanced mirror sites.

FR.PER.3 The **System** shall improve performance and availability of internet-based distribution by providing automatic failover for the search, view, retrieve, transfer, subscribe functions.

4.5 Availability Requirements

Availability Requirements cover constraints on system uptime and continuity of service.

NR.AVA.1 All **servers** on the system shall meet the availability standards of their hosting institution.

NR.AVA.2 Best efforts shall be made to minimize hardware system downtime.

NR.AVA.3 Best efforts shall be made to minimize software system downtime.

4.6 Security Requirements

Security Requirements address the need to limit system access in order to protect system and data integrity.

NR.SEC.1 The **System** shall provide unrestricted access to all data, documentation, and software released to the public.

NR.SEC.2 The **System** shall provide role-based authentication for controlled access to data or system functions and shall recognize the following roles:

- a) Authenticated
 - 1. Data provider for a specified mission
 - 2. Data distributor for a specified mission
 - 3. Central Node data engineer for a specified mission
 - 4. Subscriber (Science Community member)
 - 5. Administrator

- b) Anonymous
 - 1. General Public
 - 2. Educational Community

NR.SEC.3 The **System** shall restrict access to a mission's pre-release data to the following roles:

- a) Data provider for the mission
- b) Data distributor for the mission
- c) Central node data engineer for the mission
- d) System administrator

NR.SEC.4 The **System** shall restrict the ability to perform ingest, validate, and transfer functions on a mission's data to the following roles:

- a) Data distributor for the mission
- b) Central node data engineer for the mission
- c) System administrator

NR.SEC.5 The **System** shall restrict access to subscription information to the subscriber and the system administrator.

NR.SEC.6 The **System** shall restrict the ability to perform the administer function to the system administrator.

4.7 Compatibility Requirements

Compatibility Requirements describe the manner in which system components must be able to interoperate in a heterogeneous computing environment.

NR.CMP.1 The **System's** server-side components shall be portable across the following platforms:

- a) Solaris
- b) Linux
- c) Windows 2000, XP, Windows Server 2003 family
- d) Mac OS X

NR.CMP.2 The **System's** client-side tools shall be portable across the following platforms:

- a) Solaris
- b) Linux
- c) Windows 2000, XP, 2003 family
- d) Mac OS X

NR.CMP.3 The **System** shall enable the following languages to use APIs to client-side libraries:

- a) C
- b) Perl
- c) Java
- d) IDL

4.8 Standards Support Requirements

Standards Requirements describe the standards needed to support system operation.

NR.STD.1 The **PDS** shall ensure that all system functions are consistent with PDS standards as documented in the PDS Standards Reference.

NR.STD.2 The **PDS** shall establish standards for treating all labeled ancillary files as products.

NR.STD.3 The **PDS** shall establish standards for associating any product of any other product.

NR.STD.4 The **PDS** shall establish standards for applying version control to

metadata and data products.

NR.STD.5 The PDS shall establish standards for enabling the use of a more sophisticated target model.

4.9 Policy Support Requirements

Policy Requirements describe the policies needed to support system operation.

NR.POL.1 The **PDS** shall provide a way of determining a consistent set of standards that apply to a mission/volume set and do not change over the life of the mission/volume set.

NR.POL.2 The **PDS** shall provide a policy for enabling instrument teams to become data nodes.

NR.POL.3 The **PDS** shall provide a policy for conducting peer reviews.

NR.POL.4 The **PDS** shall provide a policy for acceptable means of electronic data delivery: where files are delivered, who maintains the area where files are delivered, how online validation and peer review are done, what must take place before data are made available.

NR.POL.5 The **PDS** shall provide a security policy that takes into account security mandates from NASA, the department of Interior, discipline node home institutions, and other PDS stake-holders.

NR.POL.6 The **PDS** shall provide a policy for enabling mirror sites.

NR.POL.7 The **PDS** shall provide a policy for ensuring backup of data that has been distributed online but not yet written to permanent archive.

NR.POL.8 The **PDS** shall provide a policy on the scope of its holdings that includes highly derived products, such as parameter catalogs.

NR.POL.9 The **PDS** shall provide a policy for responding to requests for physical media (e.g., CD, DVD, external disk, etc.), including those created by the system on demand.

NR.POL.10 The **PDS** shall provide a policy on the creation of long-term archive volumes, including:

- a) Whether the PDS will continue to create archive volumes
- b) Whether the PDS will continue to deliver archive volumes to the PDS Central Node and curating PDS Discipline Nodes; or whether the PDS

- will deliver data solely to the NSSDC for long-term data preservation
- c) If the PDS continues to create archive volumes, which physical media (e.g., CD, DVD, tape, etc.) are acceptable as archive media

NR.POL.11 The **PDS** shall specify which physical media (e.g., CD, DVD, tape, etc.) are acceptable as long-term archive volumes.

4.10 User Support Requirements

Support Requirements indicate the personnel support needed to maintain the System and provide help to users.

NR.SUP.1 The **PDS** shall provide at least one full-time system administrator, responsible for the following:

- a) Keeping servers up and running properly
- b) Maintaining and upgrading system software
- c) Performing system backups
- d) Providing technical support where needed

NR.SUP.2 The **PDS** shall provide a single e-mail address for all user feedback and requests for technical support.

4.11 Documentation Requirements

Documentation Requirements describe the documentation needed to support system operation.

NR.DOC.1 The **System** shall include a document describing the entire PDS archive lifecycle.

NR.DOC.2 The **System** shall include a Proposers Archive Guide to help proposers understand and describe how to meet PDS archive requirements in Announcements of Opportunity (AO).

NR.DOC.3 The **System** shall include templates and examples of the following archive planning and design documents:

- a) Memorandum of Understanding (MOU) between the mission and the PDS, defining roles and responsibilities
- b) Archive Plan
- c) Interface Control Document (ICD), describing the archive data products and volumes for a particular instrument

NR.DOC.4 The **System** shall include an Operations Plan, including information on the the following topics:

- a) Running system functions including failover
- b) Maintaining and upgrading system software and hardware
- c) Performing system backup and restore
- d) Ingesting, validating, distributing, and archiving data
- e) Scheduled failover testing

NR.DOC.5 The **System** shall include a development plan, including information on the following topics:

- a) Development goals
- b) Technology infusion plans
- c) Tools plan
- d) System evolution

NR.DOC.6 The **System** shall include complete documentation for tools and system APIs.

NR.DOC.7 The **System** shall include documentation for all interfaces, data structures and standards used for online as well as hard media distribution (e.g., DVD and external disk).

4.12 Development Requirements

Development Requirements describe the characteristics and constraints on the development and evolution of the system.

NR.DEV.1 The **PDS** shall use strict configuration control on all components of the system.

NR.DEV.2 The **PDS** shall develop the System in accordance with an iterative phased-development life cycle, with each iteration including the following steps:

- a) System definition
- b) Requirements analysis
- c) Design
- d) Implementation
- e) Testing
- f) Integration
- g) Deployment

5.0 PRIORITIZATION

The requirements must be prioritized in order to decide which requirements must be implemented in a given development cycle. For each requirement, there are three main considerations:

- Is the requirement part of the Core System or the Extended System?
- How important is the requirement?
- Has the requirement already been implemented?

First, all requirements are assigned membership in the core or the extended system. Then, within that classification, all requirements are assigned one of three priority levels, defined as follows:

- **Priority 1** – Critical
- **Priority 2** – Important
- **Priority 3** – Desirable

Critical requirements are needed for the PDS to perform its basic functions. Important requirements are used in day-to-day operations. And desirable requirements are useful, but not used as often or as broadly as the ones considered “important.”

Finally, it must be considered whether the requirement has already been implemented. As noted in **Section 1.4**, the current PDS-D system supports a subset of the functions needed for the new System, and existing code should be reused whenever possible. Also, since the requirements will be implemented over multiple cycles, it is necessary to track which requirements were implemented in previous cycles.

A table showing the priorities for each of the requirements can be found in **Appendix D**.

6.0 IMPLEMENTATION

6.1 Existing Implementation

As noted above, existing implementations of the requirements will be used whenever possible. In particular, the new System will inherit capabilities from the PDS Distribution System (PDS-D), which has been operational since October 1,

2002.

PDS-D was designed to distribute data from geographically separated repositories through a common web-based interface. The system primarily supports three functions:

- **Search** – Locating resources based on their cataloged attributes
- **Retrieve** – Making a located resource available to users
- **Notify** – Informing users that new or updated resources are available

The Search function in PDS-D applies to two resource types: data sets and data products. Data Sets are located by searching attributes in the Data Set Catalog. Once a data set is selected, the appropriate product catalog can be searched for data products, by attributes. The entire process is mediated through a web-based interface, typically using fill-in forms or clickable maps to enter search parameters.

The Retrieve function in PDS-D supports the downloading of any PDS label, file, or data product from any repository on the system. Any collection of these resources can also be downloaded.

Finally, PDS-D enables users to request an email notification when new or updated data from a specified instrument become available. The email contains links to the data online.

There is limited support for several other functions. For example, the web interface has a **display** function that enables viewing of browse images where available. But for the most part, PDS-D concentrates on the three functions listed above. These fulfill the basic needs for users to find and retrieve online PDS data, and they translate directly to the new System.

6.2 New Implementation

New implementation will be done through a phased delivery cycle, with deliveries coincident with the end of the fiscal year. Each delivery will deploy a working system with increasing capability and maturity. Within the delivery cycle, the traditional development phases will apply:

- **Requirements** – Selecting requirements for implementation
- **Design** – System design and design review
- **Implementation** – Coding and unit testing
- **Integration** – Deployment to test environment
- **System Test** – Complete pre-delivery regression testing
- **Delivery** – Deployment to operational system and acceptance review

Further details will be found in the PDS System Development Plan (TBD).

7.0 TEST METHODS AND ACCEPTANCE CRITERIA

For every requirement, there must be a method of testing whether it has been successfully implemented. Typically, the testing involves one or more of the following methods:

- **Demonstration:** The operation of the system that relies on observable functional operation not requiring the use of instrumentation, special test equipment, or subsequent analysis.
- **Test:** The operation of the system using instrumentation or other special test equipment to collect data for later analysis.
- **Analysis:** The processing of accumulated data obtained from other qualification methods. Examples are reduction, interpretation, or extrapolation of test results.
- **Inspection:** The visual examination of system source code, documentation, and so on.

The actual tests will be specified in detail in a section of the PDS System Development Plan (TBD).

8.0 CHANGE REQUESTS AND USER SCENARIOS

As mentioned in **Section 1**, this is a living document, meant to evolve over time, as new missions and the needs of the planetary science community dictate.

There are two main avenues for changing the requirements:

- **System Change Request (SCR)** - Direct
- **User Scenarios** – Indirect

An SCR is a direct request for a new feature or modification of current system behavior. The request can come from NASA HQ, a project, a PDS discipline node, or science user. The PDS System Engineering team will analyze the request, generate a new or modified requirement based on the analysis, and present the requirement to the PDS Requirements Working Group (RWG) for approval. If the RWG approves, the requirements described in this document will be modified accordingly.

User Scenarios provide an indirect way to develop requirements. These are short descriptions of how PDS data might be used to achieve a specified goal. Typically, the goal is the analysis of a scientific problem – although other goals, such as finding a promising landing site on Mars, for example, are also included.

User Scenarios will usually be generated by science users or PDS discipline nodes on their behalf. As with the SCR, the PDS System Engineering team will analyze the User Scenario, generate a new or modified requirement based on the analysis, and present the requirement to the PDS RWG for approval. Again, if the RWG approves, the contents of this document will be modified accordingly.

The current set of User Scenarios can be found in reference [4].

APPENDIX A ACRONYMS

Acronyms pertaining to this document:

ADS	Astrophysics Data System (NASA bibliographic database)
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange (plain text)
ATMOS	Atmospheres Discipline Node (PDS)
CN	Central Node (PDS)
CODMAC	Committee on Data Management, Archiving, and Computing
CRISM	Compact Reconnaissance Imaging Spectrometer for Mars (MRO)
CTX	Context Imager (MRO)
DVD	Digital Versatile Disc
EDR	Experiment Data Record
FITS	Flexible Image Transport System (data format)
FR	Functional Requirement
FTP	File Transfer Protocol
GB	Giga Bytes

GCM	General/Global Circulation Model
GEO	Geosciences Discipline Node (PDS)
HiRISE	High Resolution Imaging Science Experiment (MRO)
HQ	Headquarters (usually NASA HQ)
HRSC	High Resolution Stereo Camera (MRO)
HSSC	HRSC Super-resolution Stereo Camera (MRO)
HTTP	HyperText Transfer Protocol
IAU	International Astronomical Union
ICD	Interface Control Document
ID	Identifier
IDL	Interactive Data Language
IMG	Imaging Support Node (PDS)
ISIS	Integrated Software for Imagers and Spectrometers
JPEG	Joint Photographic Experts Group
JPL	Jet Propulsion Laboratory
Kernel	Fancy name used within SPICE for "file"
MEP	Mars Exploration Program
MEPAG	Mars Exploration Program Analysis Group
MGS	Mars Global Surveyor
MOC	Mars Orbiter Camera
MOC NA	MOC Narrow Angle Camera
MOC WA	MOC Wide Angle Camera
MOLA	Mars Orbital Laser Altimeter
MOS	Mission Operations Systems
MOU	Memorandum of Understanding
MRO	Mars Reconnaissance Orbiter
NAIF	Navigation and Ancillary Information Facility
NASA	National Aeronautics and Space Administration
NSSDC	National Space Science Data Center
OAL	Object Access Library
OODT	Object-Oriented Data Technology
OS	Operating System
PDR	Preliminary Design Review
PDS	Planetary Data System
PDS-A	PDS Automation
PDS-D	PDS Distribution System
PDS-M	PDS Data Model
PI	Principal Investigator
PNG	Portable Network Graphics (format)
PSDD	Planetary Science Data Dictionary
PSG	Project Science Group
RDR	Reduced Data Record
RICE	Rice Compression (name for Robert Rice)
RWG	Requirements Working Group
SAN	Storage Across Network(s)
SBN	Small Bodies Node (PDS)

SHARAD	Shallow Radar (MRO)
SIS	Software Interface Specification
SPICE	Spacecraft, Planet, Instrument, C-matrix, Events
SQL	Structured Query Language
Tbits	Tera (1E12) bits
TES	Thermal Emission Spectrometer
TB	Tera Bytes
TBD	To Be Determined
TDS	Telemetry/Tracking Data Delivery System
TES	MGS Thermal Emission Imaging System
THEMIS	Odyssey Thermal Emission Imaging System
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
USGS	United States Geological Survey
UTC	Universal Time Coordinated
UV	Ultraviolet
UVIS	Cassini UV Instrument
VIS	Visible
WG	Working Group

APPENDIX B GLOSSARY

Accept (verb) - To accept a resource is to agree to store it on the system.

Administer - The System function that enables system administrators to monitor, backup, repair, upgrade, and reconfigure the System.

ancillary information - Information deemed critical to understanding, interpreting, or processing a specified data set. In the PDS, this typically takes the form of calibration information, documentation, or software. A file containing ancillary information is referred to as an **ancillary file**.

archive (verb) - To preserve data sets and ancillary files in a PDS compliant volume structure on long-term storage media, such as CD-ROM and DVD (see data set, ancillary file, and volume).

Archive, The - The collection of all resources that are preserved as part of the PDS mandate.

access (verb) - To access a resource is to view or download a representation of that resource (see **representation**). A resource that can be accessed is **accessible**.

associated information - Information deemed related to a specified resource. The relationship between the associated information and the resource is completely at the discretion of the party making the association. The relationship could simply be that the "information is associated with the resource." Ancillary information is a special case of associated information (see **ancillary information**). A file containing associated information is referred to as an **associated file**.

attribute - A property (characteristic) that has a name and a value. For example, the Cassini mission has an attribute named "MISSION_NAME" with the value "Cassini".

authenticate (verb) - To determine whether a user logging into the system can access restricted resources.

available - A resource is available when it is accessible (see **access**). A repository or catalog is available when it is online and the resources contained in them can be accessed.

body (planetary) - A general way to refer to planets, moons, asteroids, comets, and other physical objects in the solar system.

browse (verb) - To view resources online through a web browser.

browse product - A resource representation created for viewing (see **resource representation**).

catalog (noun) - A database containing an organized collection of attributes used to select and locate resources.

catalog (verb) - To save the contents of a resource description in a database.

CHRONOS - A NAIF-supplied tool for handling transformations from one time system to another.

client - A program used to access the System from a user's computer. The most common client is a web browser. Computation done by client is referred to as being done clientside.

Core System, the - The part of the System that supports only the essentials of distribution and archiving. It ensures that PDS products are properly stored in the system and that they can be found through attribute searches, downloaded by the user, and transferred to physical media for long-term archiving.

correlate - To correlate an attribute is to determine whether it is valid to directly compare values of that attribute from two different data sets. For example, if the

keyword LONGITUDE is used in the labels of data products from two different data sets, LONGITUDE is correlated if exactly the same definition of LONGITUDE is used in both data sets.

curating node - The PDS data or discipline node responsible for distributing a specified data set.

data - The raw or processed results of scientific observations.

data node - An organization outside the PDS that distributes PDS data through the System. The organization is typically an instrument team, for example the Mars Odyssey THEMIS team, which manages the THEMIS data node at Arizona State University.

Data Mining function - The System function that extracts and uses information from the data product itself, rather than the attributes of that product.

data product - A resource consisting of a PDS compliant label and one or more files containing data. The label provides a resource description of the data files (see **data** and **label**).

data provider - The organization responsible for delivering mission archives to the PDS (see **mission archives**).

data set - A collection of related data products. The data products typically (but not always) come from a series of observations from an instrument and have been processed in the same manner (see **data product**).

Data Set Catalog, the - The PDS catalog that allows resources to be located through association with a specific data set (see **data set** and **catalog**).

deep archive - The collection of PDS archive volumes held by the National Space Science Data Center (NSSDC).

deliver data (verb phrase) - To submit mission archives to the PDS (see **submit** and **mission archive**). Typically, for large data sets, only a portion of the mission archive is submitted in a single **delivery**.

Display function - The System function that provides viewable representations of resources.

distribute (verb) - To distribute a resource is to enable users to locate and download it, along with all associated resources.

Extended System, the - The part of the System that improves the usability of the system and data, especially for science users. It calls for advanced

interfaces, correlative search across data sets, geometric calculations, on-the-fly data processing, and the use of information embedded in the data.

failover - The ability of the System to route requests from a component that has failed (e.g., a hard disk that has crashed) to one that can provide the same functionality or resources (e.g., another disk with the same contents).

gazetteer - A database that associates the topographical features of a body with their location on the body's surface (see **body**).

Geometry Function - The System function that deals with the geometric relationships between instrument hosts, instruments, and targets of observation. The Geometry Function works with properties including location, orientation, instrument pointing, time systems, ephemerides, and illumination angles.

Geo-referenced - Pertaining Where every pixel in an image is mapped to coordinates on the surface of a body in a specified map projection (see **body**).

Interface Control Document (ICD) - A document in which an instrument team describes all the mission archives that will be produced from their instrument and the schedule by which these will be delivered to the PDS (see **mission archives**).

index - Refers to a PDS index table, a fixed-field-length, comma-delimited table, where each row contains a product ID and the set of attributes describing that product. The attributes are typically the same ones found in the product's label (see **label** and **product**).

Ingest function - The System function that receives, stores, and catalogs resources (see **receive**, **store**, and **catalog**).

keyword - An element of the Planetary Science Data Dictionary (PDSDD) that defines a named property of an object. The keyword plus its value is an attribute (see **attribute** and **object**).

label (product label) - A resource description stored in a file. If the label is in the same file as the resource, it is called an **attached label**. If it is in a separate file, it is called a **detached label**.

lifecycle - The lifecycle of a resource is the set of distinct changes that a resource undergoes from its creation to its final place, if any, on the System and in the Archive.

local data dictionary - The set of definitions for all attribute names that are valid for use in resource descriptions, in a specified context. For example, all the attribute names that can be used only in the context of the Rosetta mission. In

the PDS, local data dictionaries are implemented as separate namespaces in the Planetary Science Data Dictionary (see **Planetary Science Data Dictionary**).

location - The location of a resource on the System is its Uniform Resource Locator (URL).

Management Council (PDS) - The governing body of the PDS, consisting of the project manager, project scientist, project engineer, mission interface lead, and managers of all the PDS discipline nodes.

metadata - In the general sense, metadata is any information about data that is not the data itself. In the PDS, metadata refers specifically to the resource description.

namespace - A set of names that are all unique relative to each other. Typically, there is a single “controlling authority” that controls the names in the namespace. For example, a mission or discipline node may control all the names in a local data dictionary (see **local data dictionary**).

metrics - Useful statistics about the operation of the System, for example, the number of bytes downloaded from each repository, average data rates for downloads, which products were accessed most often, etc.

mirroring (verb) - Setting up a duplicate repository on the network, in case the original repository fails or becomes overloaded.

mission archive - A data set, along with all its ancillary files, organized in a PDS-compliant volume structure (see **data set**, **ancillary information**, and **volume**).

nearline storage - Computer storage that can be placed online on demand, for example a robotic tape system that can load and unload tapes. Near-line storage is used by data warehouses as an inexpensive, scalable way to store large volumes of data.

non-functional requirements - Requirements that describe characteristics of or constraints on the System.

Notify function - The System function that enables users to receive an email informing them when selected resources have been added or modified.

Object Access Library (OAL) - A library of C language routines for reading, writing, and manipulating PDS labels and objects (see **label** and **object**).

object - An element of the Planetary Science Data Dictionary (PDSDD) that names and defines a resource (see **Planetary Science Data Dictionary**).

offline storage - Computer storage that is not accessible through the System, for example a library of DVDs.

on demand - A System function is available on demand when a user can start, stop, and control that function from the user's computer.

online - A resource is online when a user can access it from the user's computer (see **access**).

package (noun) - A set of retrievable resources and resource descriptions collected into a single file, typically for download.

physical media - Computer storage media that can be removed from a computer system and physically transferred and stored in another location. Examples are CD-ROM , DVD, and magnetic tape media. If the physical media is approved for long-term storage of PDS archives, it is referred to as **archive media**.

Planetary Science Data Dictionary (PSDD) - The set of definitions for all attribute names that are valid for use in resource descriptions, across the PDS. Note that the PSDD has been implemented as part of the PDS Data Set Catalog.

PDS compliant - A resource is PDS compliant when it adheres to all applicable PDS standards for that resource. Resources for which PDS compliance is particularly important are volumes, products, index tables, and labels (see also **validation**).

PDS node - A generic way to refer to any of the PDS data nodes, discipline nodes, or central node.

pointing information - Information giving the position and orientation of an instrument so one may determine its field of view.

policy - A rule that governs some aspect of PDS operations.

process (noun) - Computer code that can be called or executed by the System.

Process function - The System function that calls or executes computer code, using resources or resource descriptions as input.

product - A retrievable resource and its resource description.

priority - A measure of the importance of a requirement.

radiometrically corrected - The relative brightness of pixels in an image have

been corrected to the proper values.

referential integrity - A database has referential integrity if it is always true that when a record in a table refers to a corresponding record in another table, that corresponding record exists.

release (noun) - A subset of a mission archive that is delivered to the PDS and made accessible to the public as a single unit (see **mission archive**).

release (verb) - To make a resource accessible to the public (see **access**).

receive (verb) - The PDS receives a resource when a PDS node is given physical media containing the resource or when the resource is transferred over the network to a PDS node (see **submit**).

repository - mountable computer storage media that contains retrievable resources.

representation (of a resource) - The actual bits and bytes returned by a computer program that mediates between the user and the resource. For example, a program that lets the user request an image, might return a byte stream containing the image data, a zipped file containing the image, or a JPEG formatted image for display in a browser. Technically speaking, the phrase "the user views or downloads a resource" really means "the user views or downloads the representation of a resource."

resource - Anything that can be described by a set of attributes. Resources in the PDS include data, software, documents, web services, spacecraft, and personnel.

resource description - The set of attributes describing a resource. In the PDS, all attributes must be defined in the Planetary Science Data Dictionary (PSDD).

Retrieve function - The System function that enables the user to download resources to the user's computer.

review - The process of determining whether the content of a resource is scientifically usable and the description is sufficient to support its use. **Peer review** is the process of reviewing a data set.

role based authentication - Access to restricted areas of the System is determined by the user's assigned role. In the PDS, roles may include Subscriber, Data Engineer, System Administrator, etc.

search (verb) - To compare cataloged attributes, within or across contexts, to locate resources that meet user criteria. Searching includes, for example,

locating a subset of data products from within a data set, as well locating data products across instruments or missions. A search may be direct, correlated or calculated. A **direct search** ignores issues of context and compares only cataloged values. A **correlated search** will correlate the attributes to be tested before performing the the search. A **calculated search** involves computing one or more of the search criteria.

Search function - The System function that supports the location of resources based on their cataloged attributes (see **search**).

SPICE System - A means for providing scientists with ancillary observation geometry data and events, and related tools, useful in the planning and interpretation of science instrument observations returned from planetary spacecraft. SPICE stands for Spacecraft, Planet, Instrument, C-Matrix, and Events (see **Geometry function**).

status (of a resource) - The attributes that represent the state of a resource at a given point in time. Any set of attributes can be defined for this purpose and thus be “tracked” by System. However, it is natural to use the **_STATUS** keywords, which define the state of a resource with respect to a given lifecycle. For example, the **ARCHIVE_STATUS** of a data set tells which phase of the archive lifecycle the data set is in (see **lifecycle** and **Track Function**).

store (verb) - To place a resource on computer storage media in the System, so that it can be used.

submit (verb) - An organization submits a resource to the PDS when it sends physical media containing the resource to a PDS node (to **submit physically**) or transfers the resource over a network to a computer belonging to a PDS node (to **submit electronically**). Note: a special case of submitting resources is **delivering data** to the PDS (see **deliver data**).

subscribe (verb) - To subscribe to a resource means to request email notification whenever that resource is released or updated. Typically, users will subscribe to be notified when new data from a selected instrument are released.

System, the - The integrated hardware and software used to support the essential functions of the PDS.

tool - A computer program that is distributed by the System, but executes on a user's computer, independent of the System. This is in contrast with a client, which communicates with the System (see **client**).

Track function -The System function that reports on the status of a resource, starting from the time it is submitted to the PDS and continuing throughout its lifecycle (see **status** and **lifecycle**).

Transfer function - The System function that makes resources available to remote systems or writes it to removable storage media.

user - Any person who uses or accesses resources on the System.

Validate function - The System function that determines whether resources received by the PDS are PDS compliant (see **PDS compliant** and **validation**, below).

validation - The process of determining whether a resource and its description is PDS compliant. Typically, this means it is well-formatted, complete, and capable of being stored or cataloged and made accessible. A resource that passes validation is also said to be **validated** (see **PDS compliant**).

view (verb) - To view a resource is to look at a representation of the resource through a browser, another client, or display tool (see **representation**).

volume - A directory structure for storing data sets and ancillary files. A **PDS compliant volume** is one that is consistent with the structure defined in the PDS Standards Reference. Note that a volume may reside in an online repository, where it is referred to as an **online volume**. Or it may be written to archive media, where it is referred to as an **archive volume**.

APPENDIX C TRACEABILITY MATRIX

Every functional and non-functional requirement in this document can be traced to one or more sources – for example, a mission archive plan, a request from a PDS discipline node, a directive from NASA Headquarters, or a decision made by the PDS Systems Engineering team. These sources have been given identifiers and stored in a database, referred to as the “CORE database,” since it is part of the CORE (TM) Systems Engineering software package.

The following table traces every requirement to the source identifiers given in the CORE database. The columns of the table are as follows:

- **Requirement** – the requirement, as numbered in this document
- **Description** – short text to help the reader identify the requirement
- **Source** – the CORE identifier of the source of the requirement
- **Source Notes** – additional note about the source

Requirement	Description	Source	Source Note
FR.ING.1	Receive resources electronically	Working Group	From workshop
FR.ING.2	Receive resources on physical media	Working Group	From workshop
FR.ING.3	Receive resource descriptions	Working Group	From workshop
FR.ING.4	Accept or reject resources	Cassini Tracker	From mission
FR.ING.5	Store all accepted resources	CN	Cleanup edit
FR.ING.6	Catalog all resource descriptions	xCR.13.17	From nodes
FR.ING.7	Ensure pointers to resources	CN	Cleanup edit
FR.ING.8	Enable resource removal	Cassini Tracker	From mission
FR.ING.9	Ingest standard PDS files	CN	Cleanup edit
FR.ING.10	Ingest through Catalog interface	Working Group	From workshop
FR.ING.11	Ingest through PSDD interface	Geosciences	From workshop
FR.ING.12	Enable associations between resources	SBN	From workshop
FR.ING.13	Tools for mission archive production	FR.1.1.2,xCR.114.2 .17	From nodes
FR.VAL.1	Validate resource descriptions	Central node	From central node
FR.VAL.2	Validate any PDS label	FR.1.1.2,OO.1.6, xCR.18.50.4	From nodes
FR.VAL.3	Validate catalog files	FR.1.1.2,OO.1.6, xCR.18.50.4	From nodes
FR.VAL.4	Validate index tables	FR.1.1.2,OO.1.6, xCR.18.50.4	From nodes
FR.VAL.5	Validate PDS volumes	FR.1.1.2,OO.1.6, xCR.18.50.4	From nodes
FR.VAL.6	Validate data set releases	FR.1.1.2,OO.1.6, xCR.13.30.5	From nodes
FR.VAL.7	Run validation tests via web interface	Workshop	From working group
FR.VAL.8	Run validation tests via local tools	Workshop	From working group
FR.VAL.9	Specify contents of validation report	Workshop	From working group

Requirement	Description	Source	Source Note
FR.VAL.10	Send the report anywhere	Workshop	From working group
FR.VAL.11	Enable use of local data dictionaries	Workshop	From working group
FR.TRK.1	Report status for any resource	Cassini Tracker	From mission
FR.TRK.2	Alert submitter to change of status	Cassini Tracker	From mission
FR.TRK.3	Report status of mission archive planning	JPL Program Office	From JPL Program Office
FR.TRK.4	Report status of data releases	Cassini Tracker, FR.1.2.6,xCR.10.1	From mission and nodes
FR.TRK.5	Alert submitter to late delivery	Cassini Tracker	From mission
FR.TRK.6	Report all releases that are late	Cassini Tracker	From mission
FR.TRK.7	Report all releases for a mission, etc.	Cassini Tracker	From mission
FR.SCH.1	Must be able to locate any resource	Workshop	From working group
FR.SCH.2	Locate all resources from attributes	FR.1,xCR.8.4,xCR.17.3, Workshop	From nodes and working group
FR.SCH.3	Locate data sets from DS attributes	XCR.13.8	From nodes
FR.SCH.4-EXT	Determine whether search is correlated	Workshop	From working group
FR.SCH.5-EXT	Search across coordinate systems	FR.1.2.5, xCR.14.1.1	From nodes
FR.SCH.6-EXT	Locate associated resources	XCR.8.2	From nodes
FR.SCH.7	Use substrings to search characters	Workshop	From working group
FR.SCH.8	Standard URL for citation searches	XCR.18.6	From nodes
FR.SCH.9-EXT	Use MEPAG goals to find landing site	XCR.28.13	From nodes
FR.SCH.10-EXT	Geometric filter for Rings search	Workshop, added by Rings node	From nodes
FR.SCH.11-EXT	Locate all data in time window	Workshop	From working group
FR.SCH.12-EXT	Locate all data in spatial window	Workshop	From working group
FR.-DSP.1	Enable display of all res. descriptions	Workshop	From working group

Requirement	Description	Source	Source Note
FR.DSP.2	Enable display of all resources	OO.1.6,xCR.13.24, xCR.8.11	From nodes
FR.DSP.3	Display repositories as filesystem	xCR.15.17	From nodes
FR.DSP.4	Display preview representation	xCR.17.10	From nodes
FR.DSP.5	Generate interface to new data	xCR.16.2,xCR.8.27	From nodes
FR.RET.1	Deliver any set of resources	Central Node	From central node
FR.RET.2	Download right-sized resources	xCR.13.11, xCR.8.1, OO.2.6	From nodes
FR.RET.3	Use bulk download and hard media	xCR.13.14, Workshop	From nodes and working group
FR.RET.4	Inform user of download size	xCR.13.4, xCR.17.11	From nodes
FR.RET.5	Offer alternatives to user	xCR.13.4, xCR.17.11	From nodes
FR.RET.6	Retrieve resource given resource ID	xCR.14.1.7	From nodes
FR.RET.7	Retrieve all files in multipart resource	xCR.8.3	From nodes
FR.RET.8	Package all associated files	FR.1.2.2,xCR.13.10 xCR.13.14	From nodes
FR.RET.9	Build PDS-compliant volumes	FR.TRN.X1, xCR.13.9	From nodes
FR.RET.10	Retrieve most recent catalog files	xCR.17.2	From nodes
FR.RET.11-EXT	Retrieve reformatted files	xCR.17.2,xCR.18.3	From nodes
FR.RET.12-EXT	Retrieve user-defined subset	Workshop, Eric Eliason – HiRise	From mission
FR.GEO.1	Update pointing info in product catalogs	Workshop, Atmospheres	From working group
FR.GEO.2-EXT	Get targets in field of view	Workshop	From working group
FR.GEO.3-EXT	Transform between coordinate frames	Workshop	From working group
FR.GEO.4-EXT	Transform between time systems	Workshop	From working group
FR.GEO.5-EXT	Calculate when inst host within range	Workshop	From working group
FR.GEO.6-EXT	Calculate data products within view	Workshop	From working group

Requirement	Description	Source	Source Note
FR.DMG.1-EXT	Extract elevation from MOLA data	Workshop, from Geosciences	From nodes
FR.PRC.1-EXT	Provide mechanism for running routines	Workshop	From working group
FR.PRC.2-EXT	Enable plug-in of new routines	xCR.31.5	From nodes
FR.PRC.3-EXT	Invoke routines as web services	FR.12.2,xCR.31.5, xCR.14.12	From nodes
FR.PRC.4-EXT	Provide format transformations	xCR.14.1.8, xCR.13.4	From nodes
FR.PRC.5-EXT	Provide coordinate transformations	xCR.14.1.16,xCR.17.7,xCR.13.21	From nodes
FR.PRC.6-EXT	Provide subsetting capability	xCR.31.5,xCR.8.8	From nodes
FR.PRC.7-EXT	Provide image display controls	Workshop	From working group
FR.PRC.8-EXT	Provide image display overlays	FR.1.22,xCR.1.4.1.3,xCR.14.4	From nodes
FR.PRC.9-EXT	Register geo-referenced data	xCR.13.7, xCR.14.1.6	From nodes
FR.PRC.10-EXT	Compare attributes across catalogs	Workshop	From working group
FR.PRC.11	Create reduced JPEG2000 images	Workshop, MRO HiRISE	From MRO
FR.TRN.1	Transfer to physical media	xCR.13.9	From nodes
FR.TRN.2	Create PDS compliant volumes	OO.2.13,xCR.16.1	From nodes
FR.TRN.3	Deliver archive volumes	FR.1.22,xCR.17.5	From nodes
FR.TRN.4	Full or partial mirroring	Central node	From central node
FR.NOT.1	Enable subscriber accounts	XCR.10.3	From nodes
FR.NOT.2	Provide subscriptions	xCR.10.3	From nodes
FR.NOT.3	Enable subscribers to modify accounts	Workshop	From working group
FR.NOT.4	Web site for notification details	Workshop	From working group
FR.NOT.5	Keep notification history	Workshop	From working group
FR.ADM.1	Monitor server availability, etc.	xCR.32.13	From nodes

Requirement	Description	Source	Source Note
FR.ADM.2	Keep and report metrics	OO.1.9,xCR.13.28, xCR.31.6	From nodes
FR.ADM.3	Ability to update components	xCR.10.8, xCR.10.9	From nodes
FR.ADM.4	Ability to rollback updates	xCR.10.8, xCR.10.9	From nodes
FR.ADM.5	Preserve all data received	Central node	From central node
FR.ADM.6	Maintain inventory of archive volumes	FR.1.2.2,xCR.17.5	From nodes
NR.ARC.1	Open and shared architecture	xCR.17.13	From nodes
NR.ARC.2	Non-proprietary core components	xCR.17.14	From nodes
NR.ARC.3	Modularized architecture	xCR.17.14	From nodes
NR.ARC.4	Service-based architecture	xCR.17.14	From nodes
NR.ARC.5	Distributed architecture	xCR.17.12	From nodes
NR.INT.1	Web-based interface	xCR.17.17,FR.3.1	From nodes
NR.INT.2	Provide API for developers	xCR.32.12	From nodes
NR.INT.3	Provide online help for all functions	xCR.14.1.13	From nodes
NR.INT.4	Provide intuitive interaction	xCR.14.1.10	From nodes
NR.INT.5	Context sensitive data set searches	xCR.14.1.10	From nodes
NR.INT.6	Validate all web form input	xCR.10.6	From nodes
NR.INT.7	Web pages of professional quality	FR.1.2.1,xCR.15.16	From nodes
NR.INT.8	Provide look-and-feel elements	Workshop	From working group
NR.CAP.1	Capable of storing 120TB online	xCR.28.16	From nodes
NR.CAP.2	Capable of backing up 120TB online	MRO Review	From MRO review
NR.PER.1	Support ingestion of 5TB/biweekly	MRO Review	From MRO review
NR.PER.2	Support load-balanced mirrors	xCR.17.16	From nodes

Requirement	Description	Source	Source Note
NR.PER.3	Provide automatic failovers	xCR.14.1.9	From nodes
NR.AVA.1	Meet standards of hosting institution	Workshop	From working group
NR.AVA.2	Minimize hardware downtime	Workshop	From working group
NR.AVA.3	Minimize software downtime	Workshop	From working group
NR.SEC.1	Unrestricted access to public releases	xCR.28.19, OO.1.10,xCR.14.2	From nodes
NR.SEC.2	Use role-based authentication	FR.1.2.3,xCR.13.29	From nodes
NR.SEC.3	Restrict access to pre-release data	xCR.28.19	From nodes
NR.SEC.4	Restrict access to system functions	xCR.28.19	From nodes
NR.SEC.5	Restrict access to account information	FR.1.2.3, xCR.13.29	From nodes
NR.SEC.6	Restrict access to admin functions	Workshop	From working group
NR.CMP.1	Platforms for server-side tools	xCR.21.2.1	From nodes
NR.CMP.2	Platforms for client-side tools	xCR.4.1.10, xCR.8.10	From nodes
NR.CMP.3	Languages for client-side APIs	Central Node	From central node
NR.STD.1	Consistent with current standards	Workshop	From working group
NR.STD.2	Treat ancillary files as products	SBN	From nodes
NR.STD.3	Add standards for associations	SBN	From nodes
NR.STD.4	Add standards for version control	SBN	From nodes
NR.STD.5	Create better target model	SBN	From nodes
NR.STD.6	Canonical lat/lon definitions	xCR.14.1.11	From nodes
NR.POL.1	Link standards with volume set	xCR.14.2.3	From nodes
NR.POL.2	Policy for becoming data node	xCR.28.18, xCR.13.32	From nodes
NR.POL.3	Policy for peer reviews	xCR.4.1.4	From nodes

Requirement	Description	Source	Source Note
NR.POL.4	Policies to support data delivery	xCR.4.1.7	From nodes
NR.POL.5	Policy for security mandates	OO.1.0,xCR.14.2.9	From nodes
NR.POL.6	Policy for becoming a mirror site	OO.1.2,xCR.13.22	From nodes
NR.POL.7	Policy for backup and recovery	xCR.15.21	From nodes
NR.POL.8	Policy on scope of holdings	OO.2.8,xCR.8.6	From nodes
NR.POL.9	Policy on requests for physical media	OO.3.1,xCR.14.2.7	From nodes
NR.POL.10	Policy on creation of archive volumes	MRO Review	From MRO Review
NR.POL.11	Policy on archive media specs	MRO Review	From MRO Review
NR.SUP.1	Full-time system administrator	Workshop	From working group
NR.SUP.2	Provide single feedback address	NASA HQ	From NASA HQ
NR.DOC.1	Provide archive lifecycle document	xCR.13.50.2	From nodes
NR.DOC.2	Provide proposers archive guide	Management Council	From PDS MC
NR.DOC.3	Provide document templates	OO.1.1,xCR.13.32,xCR.13.33,FR.1.1.2	From nodes
NR.DOC.4	Provide Operations Plan	xCR.15.3,xCR.13.27	From nodes
NR.DOC.5	Provide Development Plan	OO.1.3,xCR.15.14,xCR.15.5,xCR.31.2	From nodes
NR.DOC.6	Document all tools and APIs	OO.1.6,xCR.4.1.10	From nodes
NR.DOC.7	Document internal structures	OO.2.13,xCR.14.2.14	From nodes
NR.DEV.1	Strict configuration management	OO.1.7,xCR.8.18	From nodes
NR.DEV.2	Use iterative phased development	Central node	From central node

APPENDIX D PRIORITY MATRIX

Every functional requirement in this document is assigned membership to the

Core System or the Extended System, then prioritized according to the method described in **Section 5**. The following table provides the prioritization. The columns of the table are as follows:

- **Requirement** – the requirement, as numbered in this document
- **Description** – short text to help the reader identify the requirement
- **Priority** – the assigned priority level
- **Implementation Status** – current state of implementation

Core System

Requirement	Description	Priority	Implementation Status
FR.ING.1	Receive resources electronically	Level 1	Not implemented
FR.ING.2	Receive resources on physical media	Level 1	Not implemented
FR.ING.3	Receive resource descriptions	Level 1	Not implemented
FR.ING.4	Accept or reject resources	Level 1	Not implemented
FR.ING.5	Store all accepted resources	Level 1	Not implemented
FR.ING.6	Catalog all resource descriptions	Level 1	Not implemented
FR.ING.7	Ensure pointers to resources	Level 1	Not implemented
FR.ING.8	Enable resource removal	Level 1	Not implemented
FR.ING.9	Ingest standard PDS files	Level 1	Partly implemented (Ingestion tools)
FR.ING.10	Ingest through Catalog interface	Level 2	Not implemented
FR.ING.11	Ingest through PSDD interface	Level 3	Not implemented
FR.ING.12	Enable associations between resources	Level 2	Not implemented
FR.ING.13	Tools for mission archive production	Level 2	Partly implemented (Various tools)
FR.VAL.1	Validate resource descriptions	Level 1	Not implemented
FR.VAL.2	Validate any PDS label	Level 1	Partly implemented (LVTOOL)
FR.VAL.3	Validate catalog files	Level 1	Partly implemented (Validation tools)

Requirement	Description	Priority	Implementation Status
FR.VAL.4	Validate index tables	Level 1	Partly implemented (Various tools)
FR.VAL.5	Validate PDS volumes	Level 1	Partly implemented (Validation tools)
FR.VAL.6	Validate data set releases	Level 1	Partly implemented (Validation tools)
FR.VAL.7	Run validation tests via web interface	Level 2	Not implemented
FR.VAL.8	Run validation tests via local tools	Level 1	Partly implemented (Validation tools)
FR.VAL.9	Specify contents of validation report	Level 2	Partly implemented (Validation tools)
FR.VAL.10	Send the report anywhere	Level 3	Not implemented
FR.VAL.11	Enable use of local data dictionaries	Level 1	Partly implemented (LVTOOL)
FR.TRK.1	Report status for any resource	Level 1	Not implemented
FR.TRK.2	Alert submitter to change of status	Level 2	Not implemented
FR.TRK.3	Report status of missed reports	Level 2	Not implemented
FR.TRK.4	Report status of archive planning	Level 2	Not implemented
FR.TRK.5	Report status of data releases	Level 1	Not implemented
FR.TRK.6	Alert submitter to late delivery	Level 3	Not implemented
FR.TRK.7	Report all releases that are late	Level 2	Not implemented
FR.TRK.8	Report all releases for a mission, etc.	Level 2	Not implemented
FR.SCH.1	Must be able to locate any resource	Level 1	Not implemented
FR.SCH.2	Locate all resources from attributes	Level 1	Not implemented
FR.SCH.3	Locate data sets from DS attributes	Level 1	Implemented D01
FR.SCH.7	Use substrings to search characters	Level 3	Not implemented
FR.SCH.8	Standard URL for citation searches	Level 3	Implemented D02
FR.-DSP.1	Enable display of all res. descriptions	Level 1	Not Implemented
FR.DSP.2	Enable display of all resources	Level 1	Partly implemented (NASAView)

Requirement	Description	Priority	Implementation Status
FR.DSP.3	Display repositories as filesystem	Level 1	Implemented D02
FR.DSP.4	Display preview representation	Level 1	Partly implemented (Image Atlas)
FR.DSP.5	Generate interface to new data	Level 2	Partly implemented (Basic Browser)
FR.RET.1	Deliver any set of resources	Level 1	Partly implemented D01
FR.RET.2	Download right-sized resources	Level 1	Partly implemented D01
FR.RET.3	Use bulk download and hard media	Level 1	Partly implemented D02
FR.RET.4	Inform user of download size	Level 1	Partly implemented D02
FR.RET.5	Offer alternatives to user	Level 1	Not implemented
FR.RET.6	Retrieve resource given resource ID	Level 2	Not implemented
FR.RET.7	Retrieve all files in multipart resource	Level 1	Partly implemented D01
FR.RET.8	Package all associated files	Level 2	Not implemented
FR.RET.9	Build PDS-compliant volumes	Level 2	Not implemented
FR.RET.10	Retrieve most recent catalog files	Level 2	Implemented (PDS catalog tool)
FR.GEO.1	Update pointing info in product catalogs	Level 2	Not implemented
FR.PRC.11	Create reduced JPEG2000 images	Level 1	Not implemented
FR.TRN.1	Transfer to physical media	Level 1	Not implemented
FR.TRN.2	Create PDS compliant volumes	Level 2	Not implemented
FR.TRN.3	Deliver archive volumes	Level 3	Not implemented
FR.TRN.4	Full or partial mirroring	Level 1	Not implemented
FR.NOT.1	Enable subscriber accounts	Level 1	Implemented D02
FR.NOT.2	Provide subscriptions	Level 1	Partly implemented D01, D02
FR.NOT.3	Enable subscribers to modify accounts	Level 1	Implemented D02

Requirement	Description	Priority	Implementation Status
FR.NOT.4	Web site for notification details	Level 2	Partly implemented D02
FR.NOT.5	Keep notification history	Level 3	Not implemented
FR.ADM.1	Monitor server availability, etc.	Level 1	Partly implemented OODT
FR.ADM.2	Keep and report metrics	Level 1	Partly implemented OODT
FR.ADM.3	Ability to update components	Level 2	Not implemented
FR.ADM.4	Ability to rollback updates	Level 2	Not implemented
FR.ADM.5	Preserve all data received	Level 1	Partly implemented (backup system)
FR.ADM.6	Maintain inventory of archive volumes	Level 1	Partly implemented (server cluster)

Extended System

Requirement	Description	Priority	Implementation Status
FR.SCH.4-EXT	Determine whether search is correlated	Level 2	Not implemented
FR.SCH.5-EXT	Search across coordinate systems	Level 1	Not implemented
FR.SCH.6-EXT	Locate associated resources	Level 1	Not implemented
FR.SCH.9-EXT	Use MEPAG goals to find landing site	Level 2	Not implemented
FR.SCH.10-EXT	Geometric filter for Rings search	Level 3	Not implemented
FR.SCH.11-EXT	Locate all data in time window	Level 1	Not implemented
FR.SCH.12-EXT	Locate all data in spatial window	Level 1	Not implemented
FR.RET.11-EXT	Retrieve reformatted files	Level 1	Partly implemented D01
FR.RET.12-EXT	Retrieve user-defined subset	Level 2	Not implemented
FR.GEO.2-EXT	Get targets in field of view	Level 1	Not implemented

Requirement	Description	Priority	Implementation Status
FR.GEO.3-EXT	Transform between coordinate frames	Level 1	Not implemented
FR.GEO.4-EXT	Transform between time systems	Level 1	Partly implemented (NAIF toolkit)
FR.GEO.5-EXT	Calculate when inst host within range	Level 3	Not implemented
FR.GEO.6-EXT	Calculate data products within view	Level 1	Not implemented
FR.DMG.1-EXT	Extract elevation from MOLA data	Level 2	Not implemented
FR.PRC.1-EXT	Provide mechanism for running routines	Level 1	Not implemented
FR.PRC.2-EXT	Enable plug-in of new routines	Level 2	Not implemented
FR.PRC.3-EXT	Invoke routines as web services	Level 2	Not implemented
FR.PRC.4-EXT	Provide format transformations	Level 1	Partly implemented
FR.PRC.5-EXT	Provide coordinate transformations	Level 1	Partly implemented
FR.PRC.6-EXT	Provide subsetting capability	Level 1	Not implemented
FR.PRC.7-EXT	Provide image display controls	Level 2	Not implemented
FR.PRC.8-EXT	Provide image display overlays	Level 2	Not implemented
FR.PRC.9-EXT	Register geo-referenced data	Level 2	Not implemented
FR.PRC.10-EXT	Compare attributes across catalogs	Level 3	Not implemented

[END OF DOCUMENT]