



International Planetary Data Alliance (IPDA)

Harvest Tool Design Specification

Version 1.0

Editor:

S. Hardman

Authors:

S. Hardman

Abstract

This document describes the design specification for the PDS Harvest-PDAP Tool specifically for the IPDA.

Status of this document

This document was generated by the Registry Development and Coordination Project (2011-2012).

Acknowledgments

This document is based on the PDS Harvest-PDAP Tool Software Requirements and Design Document (SRD/SDD).

Change Log

Revision	Date	Description
1.0	July 2012	Initial release.

Contents

Abstract.....	1
Status of this document	1
Acknowledgments.....	1
Change Log	2
Contents.....	3
1 Executive Summary	4
1.1 Purpose.....	4
1.2 Applicable Documents	4
2 Description	6
3 Use Cases	7
3.1 Register.....	7
3.2 Discover Data Set(s).....	8
3.3 Prepare Metadata	8
3.4 Submit Data Set.....	8
4 Design Philosophy, Assumptions, and Constraints	9
5 Detailed Design.....	10
5.1 Architecture	10
5.2 Interface Design	11
5.3 Data Model.....	11
5.4 Activity Flow	11
Appendix A: Acronyms.....	14
Appendix B: VOTable Example	15
Appendix C: PSA Access.....	17

1 Executive Summary

The multi-disciplinary nature of planetary science and the increasing number of national space agencies involved in planetary exploration suggest the need for a common architecture, standards and shared services to ease discovery, access and use of planetary data by world-wide scientists regardless of which agency is collecting and distributing the data and to ensure access to and exchange of high quality planetary science data products across international boundaries.

“Registries” is an element under the International Planetary Data Alliance (IPDA) Common Architecture. It describes the shared registry services that are used by multiple IPDA members and institution. Registries are catalogs of IPDA service offerings and standard data values that are necessary to enable interoperability. For example, a registry may contain information about services offered within the IPDA (e.g., various access points for getting planetary data from an agency) or it may provide standard data values such as mission names, etc. so they are used consistently across agencies.

This document addresses the use cases and software design of the Harvest-PDAP tool for the IPDA system.

1.1 Purpose

The purpose of this document is to continue the development of the “Registries” element of the IPDA Architecture. This document will convey the resulting design and implementation in a manner that is understandable to the broad spectrum of IPDA stakeholders.

The IPDA, in its level 2 requirements, identified the following requirement that is the driver for the Registry Service:

3.9 IPDA will publish standards for querying planetary data system catalogs including standard query models, protocols, and templates of user interfaces

1.2 Applicable Documents

- 1) IPDA Information Level 1 and 2 Requirements, January 2008, <http://planetarydata.org/standards/ipda-requirements-20080122.pdf>
- 2) Developing a Core Set of Data Standards for the IPDA, Concept White Paper, January 2007, http://planetarydata.org/documents/white-paper-wp/ipda-wp-001_1_0_2007feb07-ipda-developing-a-core-set-of-data-standards-for-the-ipda
- 3) NASA-PDS/ESA-PSA Planetary Data Interoperability, July 2005, http://planetarydata.org/documents/white-paper-wp/IPDA-STC-WP-001_1_0-2005JUL01-NASA%20ESA%20Interoperability.pdf
- 4) IPDA System Architecture Specification, April 2009, http://planetarydata.org/standards/IPDA_SystemArch_20090518.pdf/view
- 5) IPDA Planetary Data Access Protocol, V1.1, September 9, 2011.

- 6) VOTable Format Definition, Version 1.2, November 30, 2009.
- 7) IPDA Registry Service Design Specification, Version 1.0, July 2012.
- 8) IPDA Registry Service Protocol, Version 1.0, July 2012.
- 9) PSA to PDS4 Metadata Mapping, April 18, 2012.

2 Description

This tool provides functionality for capturing and registering data set metadata from a service that supports the Planetary Data Access Protocol (PDAP). Although the tool should support any service with a PDAP interface, the service of interest is the Planetary Science Archive (PSA) of the European Space Agency (ESA). The tool will run locally at the Planetary Data System (PDS) Engineering Node to query the PSA data set registry in order to discover data sets and register associated metadata with the IPDA instance of the Registry service [7]. Unlike the original PDS Harvest tool that crawls a local repository and extracts metadata from product labels, the Harvest-PDAP tool queries the PSA registry via the Planetary Data Access Protocol (PDAP) [5] to retrieve data set metadata which is then registered with a Registry service instance.

3 Use Cases

A use case represents a capability of the service element and how the user (actor) interacts with the system. It should be at a high enough level so as not to reveal or imply the internal structure of the system. An actor is an object (e.g., person, application, etc.) outside the scope of the component but interacts with the component. This section captures the use cases for the Harvest-PDAP tool based on the description of the tool from the previous section. These use cases will be used in the derivation of requirements for the service. The following diagram details the use cases:

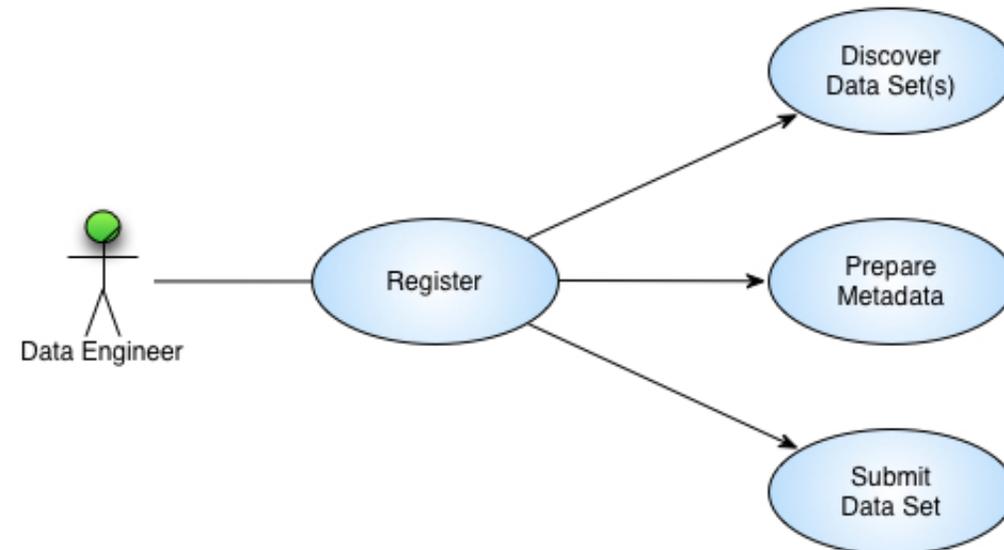


Figure 1: Harvest-PDAP Use Cases

The above diagram identifies the following actors (represented as stick figures):

Data Engineer

This actor represents an engineer that curates the data before and after it enters the IPDA system.

The following sections detail the use cases identified in the above diagram.

3.1 Register

The tool runs in a mode where it performs one query against the PSA registry and registers the data sets discovered. This use case pertains to the Data Engineer actor.

1. Data Engineer executes the Harvest-PDAP tool specifying the configuration file.
2. Harvest-PDAP tool queries for data sets in the PSA catalog (include Discover Data Set(s) use case).
3. Harvest-PDAP tool prepares metadata for each discovered data set (include Prepare Metadata use case).
4. Harvest-PDAP tool registers each data set with the target Registry service instance (include Submit Data Set use case).

3.2 Discover Data Set(s)

Data sets are discovered based on the results returned from a query to the PSA registry. This use case is included as part of the Register use case.

1. Harvest-PDAP tool obtains criteria for accessing the PSA registry from the configuration file.
2. Harvest-PDAP tool queries the PSA registry for the list of data sets and their associated metadata.
3. Harvest-PDAP tool discovers candidate data set(s).

Alternative: Previously Discovered Data Set

At step 3, the tool has already registered the discovered data set product.

- a. Harvest-PDAP tool determines a previous registration for a candidate data set product and skips it.
- b. Return to primary scenario at step 2.

3.3 Prepare Metadata

Metadata is prepared for a discovered data set based on the associated metadata returned from the PSA registry. This use case is included as part of the Register use case.

1. Harvest-PDAP tool determines the metadata for a data set based on the associated metadata returned from the PSA registry.
2. Harvest-PDAP tool retrieves the VOLDESC.CAT file from the PSA repository to determine the name of the data set catalog file.
3. Harvest-PDAP tool retrieves the data set catalog file from the PSA repository and extracts additional metadata from that file for the data set.
4. Harvest-PDAP tool formats the metadata for submission to the Registry service.

3.4 Submit Data Set

A data set and its associated metadata are submitted to the target instance of the Registry service. This use case is included as part of the Register use case.

1. Harvest-PDAP tool authenticates for access to the Registry service API.
2. Harvest-PDAP tool submits the associated metadata for a product for registration via the Registry service API.
3. Registry service responds with a successful status regarding the registration and the global unique identifier for the product.
4. Harvest-PDAP tool logs the registration.

Alternative: Product Registration Fails

At step 2, the product registration fails for any number of reasons.

- a. Registry service returns an exception with cause of failure.
- b. Harvest-PDAP tool logs the exception.

4 Design Philosophy, Assumptions, and Constraints

The intent of the Harvest-PDAP tool is to provide a simple solution for querying the PSA registry for the purpose of registering data set products into the federated system of registries.

The PSA registry offers a couple of different interfaces for obtaining metadata, but we chose to use the PDAP interface. PDAP is a REST-based API for interacting with the PSA registry and is based on an International Planetary Data Alliance standard.

5 Detailed Design

The design covers the component breakdown within the tool, external/internal interfaces and the associated data model.

5.1 Architecture

The following diagram details the component breakdown for the Harvest-PDAP tool:

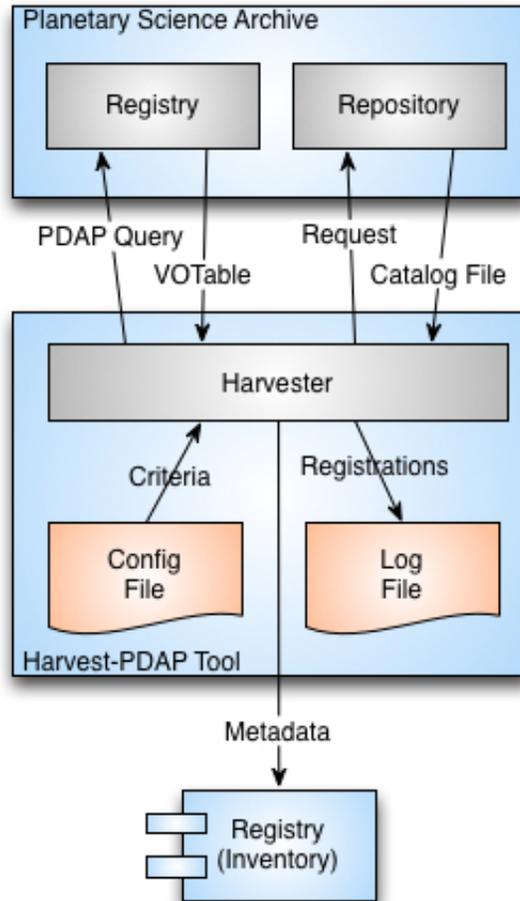


Figure 2: Harvest-PDAP Tool Architecture

The Harvest-PDAP tool consists simply of a Harvester component that receives its configuration from a local configuration file as well as from command-line parameters. It queries the PSA registry via the PDAP interface and receives metadata for candidate data set products in VOTable [6] format. See Appendix B for an example VOTable structure. The Harvester may also retrieve the associated data set catalog file from the PSA repository to gather additional metadata regarding the data set. The Harvester then registers those data set products with the target Registry service instance using the REST-based Registry service API. The IPDA Registry Service Protocol [8] documents the API in detail. A local log file captures each registration.

5.2 Interface Design

The following sections describe the Harvest-PDAP tool interfaces.

5.2.1 External Interface Design

The external interface for the Harvest-PDAP tool is limited to the command-line interface and the configuration file. The tool utilizes Apache's CLI (Command-Line Interface) library for accepting options on the command-line. The command-line interface accepts the following options:

- File specification for the configuration file
- File specification for the log file
- User name and password for registering with a secured Registry service

The configuration file utilizes an XML structure for specifying additional information pertaining to a specific harvest execution. The following information is typical:

- End point for the PDAP interface.
- End point for the Registry Service.
- Static metadata to be registered with each Data Set.

5.2.2 Internal Interface Design

The Harvest-PDAP tool does not have any internal interfaces of consequence.

5.3 Data Model

The Harvest-PDAP tool does not have an associated data model but the metadata that passes to the Registry service for data set product registration is subject to the IPDA Information Model.

5.4 Activity Flow

This section offers a more detailed look at certain aspects of the Harvest-PDAP Tool design. The following diagram details the activity flow of the software:

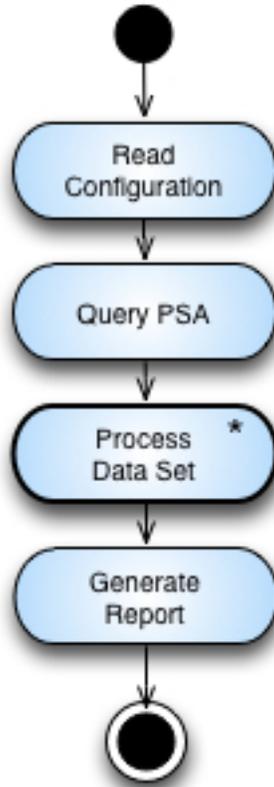


Figure 3: Harvest-PDAP Tool Activity (Overview)

The activity titled “Process Data Set *” in the diagram above represents an iteration over all candidate data set products identified in the query results from the “Query PSA” activity. Example URLs for the PSA interface can be found in Appendix C.

5.4.1 Process Data Set

The following diagram details the activity flow for this activity:

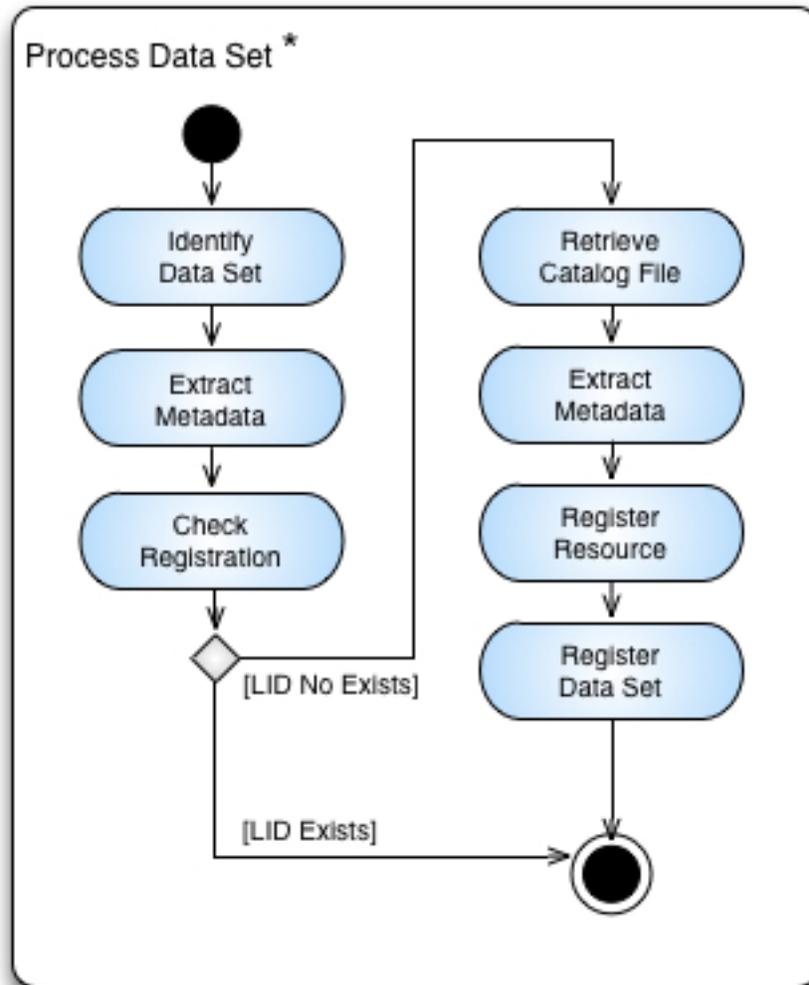


Figure 6: Harvest-PDAP Tool Activity (Process Data Set)

The first “Extract Metadata” activity above involves reading the VOTable entry for the target data set and mapping [10] the appropriate fields to the Product_Data_Set_PDS3 product from the PDS4 Information Model [9]. The second “Extract Metadata” activity above involves reading the data set catalog file and mapping the additional fields to the Product_Data_Set_PDS3 product as well.

In order to support the current PDS Data Search capability, the tool also needs to register an associated Product_Resource product containing the URL to the HTML page for a given data set.

Appendix A: Acronyms

The following acronyms pertain to this document:

API	Application Programming Interface
CLI	Command-Line Interface
ESA	European Space Agency
HTTP	Hypertext Transfer Protocol
JPL	Jet Propulsion Laboratory
LID	Logical Identifier
NASA	National Aeronautics and Space Administration
PDS	Planetary Data System
PDS4	Version 4 of the PDS Standards
PSA	Planetary Science Archive
REST	Representational State Transfer
SDD	Software Design Document
SRD	Software Requirements Document
URL	Uniform Resource Locator
XML	Extensible Markup Language

Appendix B: VOTable Example

This is an example VOTable structure returned from the PSA registry that contains a single data set. Structures with multiple data sets will have additional <TR> blocks under the <TABLEDATA> block.

```
<?xml version="1.0"?>
<!DOCTYPE VOTABLE SYSTEM "http://us-vo.org/xml/VOTable.dtd">
<VOTABLE version="1.1">
  <RESOURCE type="results">
    <DESCRIPTION>PSA Metadata Query Service</DESCRIPTION>
    <INFO name="QUERY_STATUS" value="OK" />
    <TABLE>
      <FIELD ID="DATA_SET.DATA_SET_ID" ucd="DATA_SET_ID"
utype="pds:DATA_SET.DATA_SET_ID" datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.DATA_SET_NAME" ucd="DATA_SET_NAME"
utype="pds:DATA_SET.DATA_SET_NAME" datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.DATA_ACCESS_REFERENCE"
ucd="DATA_ACCESS_REFERENCE" datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.XML_DESCRIPTION" ucd="XML_DESCRIPTION"
datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.PRODUCER.FULL_NAME" ucd="FULL_NAME"
utype="pds:DATA_SET.PRODUCER.FULL_NAME" datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.PRODUCER.INSTITUTION_NAME"
ucd="INSTITUTION_NAME" utype="pds:DATA_SET.PRODUCER.INSTITUTION_NAME"
datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.PRODUCER.NODE_NAME" ucd="NODE_NAME"
utype="pds:DATA_SET.PRODUCER.NODE_NAME" datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.START_TIME" ucd="START_TIME"
utype="pds:DATA_SET.START_TIME" datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.STOP_TIME" ucd="STOP_TIME"
utype="pds:DATA_SET.STOP_TIME" datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.NPRODUCTS" ucd="NPRODUCTS"
utype="pds:DATA_SET.NPRODUCTS" datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.MISSION_NAME" ucd="MISSION_NAME"
utype="pds:DATA_SET.MISSION_NAME" datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.INSTRUMENT_ID" ucd="INSTRUMENT_ID"
utype="pds:DATA_SET.INSTRUMENT_ID" datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.INSTRUMENT_NAME" ucd="INSTRUMENT_NAME"
utype="pds:DATA_SET.INSTRUMENT_NAME" datatype="char" arraysize="*" />
      <FIELD ID="DATA_SET.TARGET_NAME" ucd="TARGET_NAME"
utype="pds:DATA_SET.TARGET_NAME" datatype="char" arraysize="*" />
      <FIELD ID="RESOURCE_CLASS" ucd="RESCLASS" datatype="char"
arraysize="*" />
    <DATA>
      <TABLEDATA>
        <TR>
          <TD>AIRUB-C-PHOTOCAM-2-EDR-HALLEY-1986-V1.0</TD>
          <TD><![CDATA[AIRUB-HALLEY-PHOTOGRAPHIC-PROJECT-EDR-1986-
V1.0]]></TD>
          <TD><![CDATA[http://psa.esac.esa.int:8000/aio/jsp/ \
product.jsp?dataSetID=AIRUB-C-PHOTOCAM-2-EDR-HALLEY-1986-
V1.0&compression=tar&protocol=HTTP]]></TD>
          <TD><![CDATA[http://psa.esac.esa.int:8000/aio/jsp/ \
fileXML.jsp?DATA_SET_ID=AIRUB-C-PHOTOCAM-2-EDR-HALLEY-1986-V1.0]]></TD>
          <TD><![CDATA[WERNER E. CELNIK]]></TD>
          <TD><![CDATA[ASTRONOMISCHES INSTITUT DER RUHR-UNIVERSITAET
BOCHUM]]></TD>
        </TR>
      </TABLEDATA>
    </DATA>
  </RESOURCE>
</VOTABLE>
```

```

<TD></TD>
<TD>1986-02-16 00:00:00.0</TD>
<TD>1986-04-18 00:00:00.0</TD>
<TD>1833</TD>
<TD><![CDATA[EARTH]]></TD>
<TD>300,FFC,HBL,HUV,RUV</TD>
<TD><![CDATA[HASSELBLAD-ZEISS-PLANAR-F2-110MM,HASSELBLAD-
ZEISS-UV-SONNAR-F4.3-105MM,LICHTENKNECKER-FLAT-FIELD-CAMERA-F4-
760MM,PENTACON-OPTICS-F4-300MM,ROLLEI-ZEISS-UV-SONNAR-F4.3-
105MM]]></TD>
<TD>1P/HALLEY,M83</TD>
<TD>DATA_SET</TD>
</TR>
</TABLEDATA>
</DATA>
</TABLE>
</RESOURCE>
</VOTABLE>

```

Appendix C: PSA Access

This appendix provides example URLs for accessing the PSA registry and repository.

- Main page for the PSA Archive InterOperability System
<http://psa.esac.esa.int:8000/aio/doc/>
- PDAP query that returns all data sets in VOTable XML format
http://psa.esac.esa.int:8000/aio/jsp/metadata.jsp?RETURN_TYPE=VOTABLE
- PDAP query that returns a single data set in VOTable XML format
http://psa.esac.esa.int:8000/aio/jsp/metadata.jsp?DATA_SET_ID=AIRUB-C-PHOTOCAM-2-EDR-HALLEY-1986-V1.0&RETURN_TYPE=VOTABLE
- PDAP query that returns a single data set in HTML format (this will be the resource link)
http://psa.esac.esa.int:8000/aio/jsp/metadata.jsp?DATA_SET_ID=AIRUB-C-PHOTOCAM-2-EDR-HALLEY-1986-V1.0&RETURN_TYPE=HTML
- HTTP request for a volume description catalog file
<http://psa.esac.esa.int:8000/aio/jsp/product.jsp?dataSetID=AIRUB-C-PHOTOCAM-2-EDR-HALLEY-1986-V1.0&productID=&path=/&fileName=VOLDESC.CAT&protocol=HTTP>
- HTTP request for a data set catalog file
<http://psa.esac.esa.int:8000/aio/jsp/product.jsp?dataSetID=AIRUB-C-PHOTOCAM-2-EDR-HALLEY-1986-V1.0&productID=&path=CATALOG/&fileName=DATASET.CAT&protocol=HTTP>