



International Planetary Data Alliance (IPDA)

## Planetary Data Access Protocol (PDAP)

**Version:** V1.0 09.Jul.2010  
**Latest version:** <http://www.planetarydata.org/>  
**Editor(s):** Pedro Osuna, Jesus Salgado

**Author(s):** Pedro Osuna (ESA/PSA)  
Jesus Salgado (ESA/PSA)  
Steve Hughes (NASA/PDS)  
Sean Kelly (NASA/PDS)  
Yukio Yamamoto (JAXA)  
and the IPDA technical expert group members

## Abstract

This specification defines a protocol for the retrieval of distributed planetary data.

The interface is meant to be reasonably simple to implement by service providers. The basic summarized functionality of the protocol is as follows: a query defining a certain restriction on the available data is used as a first step to query for candidate data. The service returns a list of candidate hits formatted as a VOTable. For each candidate hit an access reference URL may be used to retrieve the real data. Data may be returned in a variety of formats.

This specification describes the “simple” way to access the data through an HTTP protocol. Other types of more complex access, eventually using SQL-like language and more complex transports as SOAP, will be worked on within the IPDA technical experts group.

## Status of This Document

This is the first release of this document.

*This is an IPDA Working Draft for review by IPDA members and other interested parties. It is a draft document and may be updated, replaced, or obsoleted by other documents at any time.*

A list of [current IPDA Recommendations and other technical documents](http://planetarydata.org/technical-reports/) can be found at <http://planetarydata.org/technical-reports/>.

## Contents

1	Overview	4
2	Requirements for Compliance	4
2.1	Compliance	5
3	Planetary Data Service Types	5
4	General Service Metadata Query	6
4.1	Input format	6
4.2	Compulsory input	8
4.2.1	DATA_SET_ID	9
4.2.2	PRODUCT_ID (for requests of RESOURCE_CLASS=PRODUCT)	9
4.2.3	INSTRUMENT_TYPE	9
4.2.4	TARGET_TYPE	10
4.2.5	TARGET NAME	10
4.2.6	INSTRUMENT_NAME	10
4.2.7	START_TIME	10
4.2.8	STOP_TIME	11
4.2.9	RETURN_TYPE	11
4.2.10	RESOURCE_CLASS	11
4.3	SUCCESSFUL OUTPUT	12
4.3.1	Metadata Output – Data Specific Attributes	12
4.3.2	Metadata Output – Data Specific Attributes	13
4.3.3	Metadata output – Reference Attributes	15
5	Map Projected Data Service Metadata Query	16
5.1	Map Projected Data Service extended Input Parameters	16
5.1.1	LONGITUDE	16
5.1.2	LATITUDE	17
5.2	Map Projected Data Service extended Output Fields	17
6	Service Metadata	18
6.1	Registering a Compliant Service	20
7	PDAP version information	20
8	How to design your RESOURCE_CLASS	21
9	Appendix A: Range definition for input parameters	21
10	Appendix B: Valid Map Projected Data Service response in VOTable	22
	References	23

## 1 Overview

The Planetary Data Access Protocol (PDAP) is an International Planetary Data Alliance (IPDA) [1],[2] protocol that defines how to interchange planetary data. This protocol has been developed as a combination of OGC [3] protocols (mainly for map-projected data) and IVOA protocols (for astronomical data but a science level data) like the Simple Image Access Protocol (SIAP) [4].

This close to other protocols definition allows easy integration with other data sources from other Virtual Observatories, but using particular language and characterization of planetary data.

The PDAP is a two steps protocol. That means that two steps need should be performed in order to access the data, in the following sequence:

- Step 1: Metadata Access
- Step 2: Data Access

In the first step, the archive is asked for the metadata pertaining to the data that obey a certain restriction in the input parameters. Among the compulsory output parameters of this metadata is a reference to the real file holding the data. This allows for the second step of the protocol to be executed, as the data are readily made available through the declared link.

## 2 Requirements for Compliance

Compliance with this specification requires that an access web service be maintained with the following characteristics:

1. The **Metadata Query** web method **MUST** be supported as defined in [section 4](#) below.

Through this web method, clients search for available data that match certain client-specified criteria. The response is a table that describes the available data, including metadata and access references (implemented here as URLs) for retrieving them.

2. The **Data Retrieval** web method **MUST** be supported as defined in [section 6](#) below.

This method allows clients to retrieve data through a simple synchronous HTTP GET/POST request using the access reference (URL) returned earlier by the metadata query. The response to the GET/POST request is

the requested data, returned with a MIME type such as "image/fits", "image/jpeg", "image/IMG" and so forth.

3. Finally the service **MUST** be **registered** by providing the information defined in [section 7](#) below.

Registration allows clients to use a central registry service to locate compliant PDAP services and select an optimal subset of services to query, based on the characteristics of each service and the data collections it serves.

## 2.1 Compliance

The keywords "MUST", "REQUIRED", "SHOULD", and "MAY" as used in this document are to be interpreted as described in RFC 2119 [34].

An implementation is compliant if it satisfies all the MUST or REQUIRED level requirements for the protocols it implements. An implementation that satisfies all the MUST or REQUIRED level and all the SHOULD level requirements for its protocols is said to be "unconditionally compliant"; one that satisfies all the MUST level requirements but not all the SHOULD level requirements for its protocols is said to be "conditionally compliant".

## 3 Planetary Data Service Types

The PDAP protocol is very basic in nature: few input/output parameters are required, making the protocol very flexible and open. This flexibility allows the handling of as different data types as images, spectra, plasma information, atmospheric information, etc.

The general PDAP protocol defines "compulsory" input and output parameters within its working scheme for the general case. However, specific services might be specialised in specific subsystems or instruments, for instance only serving images. In this case, specific parameters are needed to ensure interoperability at the granularity level of the specific service.

Current IPDA PDAP standard will only describe two types of services:

- General Data service
- Map projected data service

The General Data service abides by the input/output parameters described for the general PDAP protocol, whereas the Map projected data Service contains an **extended set** of input/output parameters with its own set of "compulsory" items.

The Map Projected Data Service will be described below as a specific case of the General Data service

Other services, like Spectral services, Atmospheric services, Plasma services, etc., will certainly need to be specified in a near future. Those services shall be added to new versions of the current document as needed.

## 4 General Service Metadata Query

The purpose of the **metadata query web method** is to allow users to search for data for a given set of input parameters (c.f., section 4.1 ff. The service will return references to data which most closely match what is requested. How closely the service can match the specifications of the data requested by the client will depend upon the type of data service as defined in [section 3](#) above. The service returns a table listing all available data satisfying the query. Descriptive metadata is returned for each hit to allow the client to decide which, if any, data it wishes to retrieve. An access reference is provided for each hit to permit subsequent staging and retrieval.

### 4.1 Input format

The metadata query input is transmitted as an HTTP GET/POST request represented by a URL with two parts:

1. A base URL of the form:

[http://<server-address>/<path>? \[<extra arg>&\[...\]\]](http://<server-address>/<path>? [<extra arg>&[...]])

#### Examples:

```
http://psa.esac.esa.int/aio/jsp/metadata.jsp?  
http://starbrite.jpl.nasa.gov/q?
```

Note that when it contains extra GET/POST arguments, the base URL ends in an ampersand, **&**; if there are no extra arguments, then it ends in a question mark, **?**. Every query to a given metadata query service uses the same base URL.

2. Constraints expressed as a list of ampersand-delimited GET/POST arguments, each of the form: `<name>=<value>`

**Examples:**

```
TARGET_NAME=MARS&MISSION_NAME=MARS_EXPRESS&  
DATA_SET_ID=TSK-C-RCC-CAL-RSAWIRTANENV1.0
```

The constraints represent the query parameters which can vary for each successive query.

The baseURL and constraint list are concatenated to form the query.

**Example:**

```
http://psa.esac.esa.int:8000/aio/jsp/metadata.jsp?TARGET\_NAME=MARS&  
MISSION\_NAME=MARS\_EXPRESS&RESOURCE\_CLASS=DATA\_SET
```

## 4.2 PDAP classes description

The RESOURCE\_CLASS input parameter defines the granularity of the PDAP response. As commented before, a valid PDAP response would be, in a general sense, a result table in VOTable format. Every row in the result table contains metadata information of the same kind of abstract object, e.g., the VOTable response could contain a set of datasets (if RESOURCE\_CLASS=DATA\_SET is selected) one row per dataset, a set of products if RESOURCE\_CLASS=PRODUCT is selected) one row per dataset, and so on.

This schema is extensible to other kind of objects, typically extensions of the basic object PRODUCT. In this PDAP specification document, a product extension object called MAP\_PROJECTED is also allowed and described. This map projected product object contains geometrical information so it could accept queries using geometrical constraints and it provide footprint information in the response.

As a summary of the classes described in present document, a data set contains a set of products. Both data sets and products can be downloaded using a data access reference. As one example of product extension, a class map projected product is also included in present specification. This map projected product class has associated a footprint object indirectly described in present specification.

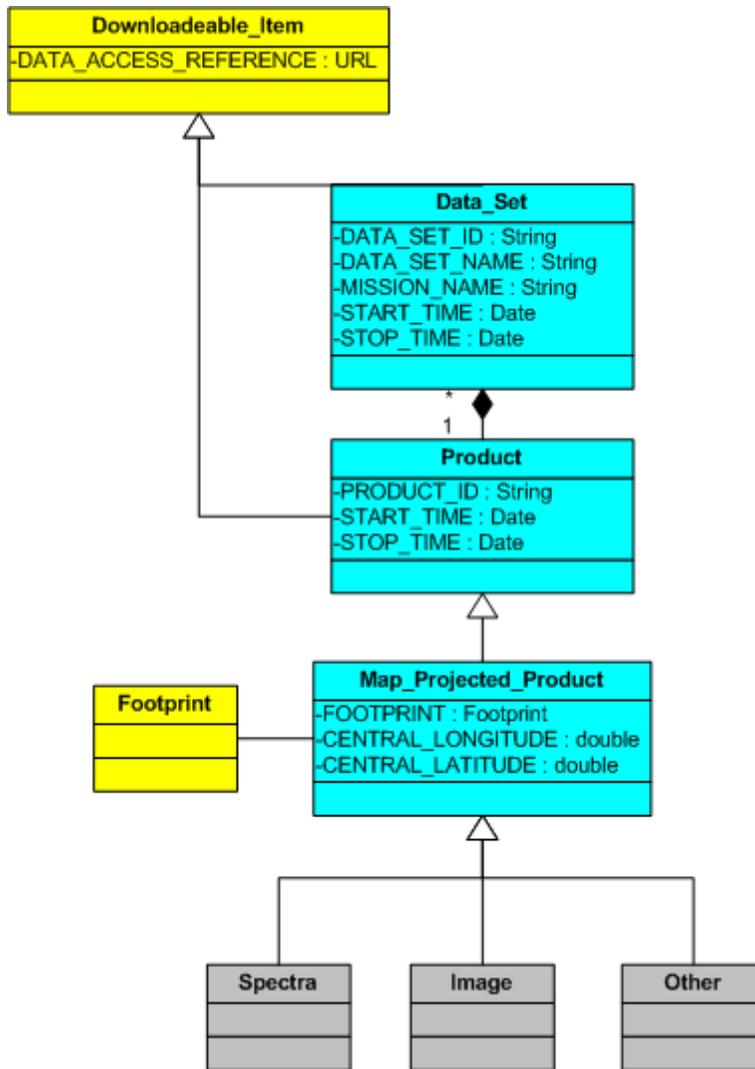


Figure 1: Simplified UML diagram of the resource classes described in the present PDAP specification. In light blue, classes directly described in the specification; in yellow, classes indirectly described in the spec; in gray, examples of possible extensions

Other possible extensions like spectra, plasma data, atmospheric data, etc could be created and easily plugged into the system by the definition of the object data model extension and the direct identification of the new query parameters and output response that this new object could provide. See the Data Specific attributes sections for a detailed description for DATA\_SET, PRODUCT and MAP\_PROJECTED resource classes.

### 4.3 Compulsory input

The following is a list of the compulsory input parameters that any service implementing the current Protocol **MUST** be able to understand. In this context, the “compulsory” attribute means that the service must be able to respond

whenever any of these parameters is input in the “key=<value>” pair of the aforementioned base URL, but it is not needed that the service give compulsorily any result.

At the same time, not all the compulsory input parameters should be present in a valid PDAP URL query.

Values should be URL encoded (e.g. spaces replaced by + or %20)

#### **4.3.1 DATA\_SET\_ID**

The DATA\_SET\_ID element is a unique alphanumeric identifier for a data set or a data PRODUCT. The DATA\_SET\_ID value for a given data set or product is constructed according to flight project naming conventions. In most cases the DATA\_SET\_ID is an abbreviation of the DATA\_SET\_NAME.

**Example:**

MR9/VO1/VO2-M-ISS/VIS-5-CLOUD-V1.0

#### **4.3.2 PRODUCT\_ID (for requests of RESOURCE\_CLASS=PRODUCT)**

The PRODUCT\_ID data element represents a permanent, unique identifier assigned to a data product by its producer within a dataset. That means that the combination DATA\_SET\_ID.PRODUCT\_ID should be unique. The PRODUCT\_ID can describe the lowest-level data object that has a PDS [5] label.

**Example:**

TSK\_C\_RCC\_3\_19970303T165718

#### **4.3.3 INSTRUMENT\_TYPE**

The INSTRUMENT\_TYPE element identifies the type of an instrument.

**Examples:**

POLARIMETER, RADIOMETER, REFLECTANCESPECTROMETER, VIDICON CAMERA

#### 4.3.4 TARGET\_TYPE

The TARGET\_TYPE element identifies the type of a named target.

**Examples:**

PLANET, SATELLITE, RING, REGION, FEATURE, ASTEROID, COMET

#### 4.3.5 TARGET\_NAME

The TARGET\_NAME element identifies a target. The target may be a planet, satellite, ring, region, feature, asteroid or comet.

**Examples:**

GANYMEDE, MARS, 5797 BIVAJ

#### 4.3.6 INSTRUMENT\_NAME

The INSTRUMENT\_NAME element provides the full name of an instrument.

**Examples:**

FLUXGATE MAGNETOMETER, NEAR\_INFRARED MAPPING  
SPECTROMETER, HIGH RESOLUTION STEREO CAMERA

#### 4.3.7 START\_TIME

The START\_TIME element provides the date and time of the beginning of an event or observation (whether it be a spacecraft, ground-based, or system event) in UTC system

**Formation rule:**

YYYY-MM-DDThh:mm:ss[.fff]

**Example:**

1997-03-03T16:18:39.000

### 4.3.8 STOP\_TIME

The STOP\_TIME element provides the date and time of the end of an event or observation (whether it be a spacecraft, ground-based, or system event) in UTC system

**Formation rule:**

YYYY-MM-DDThh:mm:ss[.fff]

**Example:**

1997-03-11T19:29:36.000

### 4.3.9 RETURN\_TYPE

The return type describes the type of output that should be received after the metadata query. The default will be VOTable (see ref. [6]). Other possible formats are HTML and ASCII.

**Examples:**

HTML, VOTABLE

### 4.3.10 RESOURCE\_CLASS

The RESOURCE\_CLASS input parameter defines the granularity of the PDAP response. See PDAP classes description section for a detailed description.

**Examples:**

DATA\_SET, PRODUCT, MAP\_PROJECTED

## 4.4 SUCCESSFUL OUTPUT

The output returned by a **Metadata Query** is a [VOTable](#) version 1.1, an XML table format, returned with a MIME-type of text/xml. The table lists all the data available to the client that match the query constraints. The following requirements are placed on the contents of the table when the query successfully returns a list of available data:

1. The VOTable **MUST** contain a RESOURCE element, identified with the tag type="results", containing a single TABLE element which contains the results of the query. The VOTable is permitted to contain additional RESOURCE elements, but the usage of any such elements is not defined here. If multiple resources are present it is recommended that the query results be returned in the first resource element.
2. The RESOURCE element SHOULD contain an INFO with name="QUERY\_STATUS". Its value attribute should set to "OK" if the query executed successfully, regardless of whether any matching products were found. All other possible values for the value attribute are described in [section 4.3](#) below.

### Examples:

```
<INFO name="QUERY_STATUS" value="OK"/>
```

```
<INFO          name="QUERY_STATUS"          value="OK">Successful  
Search</INFO>
```

3. Each table row represents a different data available to the client.
4. Each row of the output VOTable MUST contain FIELDS where a certain **utype** (see ref.on IPDA Data Dictionary Keywords) or ID have been set.

### 4.4.1 Metadata Output – Data Specific Attributes RESOURCE\_CLASS=DATA\_SET level

- Exactly one field **MUST** have one *utype*="pds:DATA\_SET.DATA\_SET\_ID" with datatype="char", and arraysize="\*", with obvious contents (see previous definitions).

- Exactly one field **MUST** have one *utype*="pds:DATA\_SET.MISSION\_NAME " with datatype="char", and arraysize="\*", with obvious contents (see previous definitions).
- Exactly one field **MUST** have one *utype*="pds:DATA\_SET.START\_TIME" and one *utype*="pds:DATA\_SET.STOP\_TIME" with datatype="char" and arraysize="\*" with obvious contents (see previous definitions)
- Exactly one field **MUST** have one *ID*="RESOURCE\_CLASS" with datatype="char", and arraysize="\*". For this protocol version, the only ont value accepted is DATA\_SET. See same field description for PRODUCT RESOURCE\_CLASS.
- Exactly one field **MUST** have one *utype*="pds:DATA\_SET.DATA\_SET\_NAME", with datatype="char", and arraysize="\*", containing a short (usually one line) description of the data set. This should concisely describe the data set to a user, typically identifying the data source (e.g., survey name), object name or field coordinates, bandpass/filter, and so forth. Note that a client application may present the user with a combined list of data from many different queries and sources: the title string is important to identify each data to the user.
- Exactly one field **MUST** have *ID*="DATA\_SET.DATA\_ACCESS\_REFERENCE", with datatype="char" and arraysize="\*", specifying the URL to be used to access or retrieve the data. Since the URL will often contain metacharacters the URL is normally enclosed in an XML CDATA section (<![CDATA[...]]>) or otherwise encoded to escape any embedded metacharacters.

**Example:**

```
<![CDATA[
http://psa.esac.esa.int:8000/aio/jsp/product.jsp?dataSetID=MEX-M-
ASPERA3-2-EDR-ELS-V1.0&PROTOCOL=HTTP]]>
```

#### 4.4.2 Metadata Output – Data Specific Attributes RESOURCE\_CLASS=PRODUCT level

- Exactly one field **MUST** have one *utype*="pds:PRODUCT.PRODUCT\_ID" (the combination of DATA\_SET\_ID plus PRODUCT\_ID is required to identify the product) with datatype="char", and arraysize="\*", with obvious contents (see previous definitions).

- Exactly one field **MUST** have one *utype*="pds:DATA\_SET.MISSION\_NAME " with datatype="char", and arraysize="\*", with same contents than previous definition.
- Exactly one field **MUST** have one *utype*="pds:PRODUCT.START\_TIME" and one field **MUST** have *utype*="pds:PRODUCT.STOP\_TIME", with obvious contents (see previous definitions)
- Exactly one field **MUST** have one *ID*="RESOURCE\_CLASS" with datatype="char", and arraysize="\*". Even when there is a query parameter **RESOURCE\_CLASS** to filter the query output, this field could contain different heterogeneous values, e.g. IMAGE and SPECTRUM (if these resource classes are defined in a external document) for a PRODUCT query. The use of this parameter would be to help clients to handle products in a better way, using e.g. specific helper applications.

**Examples of valid values:**

PRODUCT, MAP\_PROJECTED

- Exactly one field **MUST** have one *utype*="pds:PRODUCT.PRODUCT\_ID", with datatype="char", and arraysize="\*", containing a short (usually one line) description of the product. Note that a client application may present the user with a combined list of data from many different queries and sources: the title string is important to identify each data to the user.
- Exactly one field **MUST** have *ID*="DATA\_ACCESS\_REFERENCE", with datatype="char" and arraysize="\*", specifying the URL to be used to access or retrieve the data. Since the URL will often contain metacharacters the URL is normally enclosed in an XML CDATA section (<![CDATA[...]]>) or otherwise encoded to escape any embedded metacharacters.

**Example:**

```
<![CDATA[http://psa.esac.esa.int:8000/aio/jsp/product.jsp?dataSetID=MEX-M-HRSC-3-RDRV2.0&productID=H1483\_000\_P22.IMG&PROTOCOL=HTTP ] ]>
```

- Exactly one field **COULD** have `ID="ICON_ACCESS_REFERENCE "`, with `datatype="char"` and `arraysize=""`, specifying the URL to be used to access to an ICON representation of the product. Since the URL will often contain metacharacters the URL is normally enclosed in an XML CDATA section (`<![CDATA[...]]>`) or otherwise encoded to escape any embedded metacharacters.

**Example:**

```
<![CDATA[http://psa.esac.esa.int:8000/aio/jsp/product.jsp?dataSetID=MEX-M-HRSC-3-RDRV2.0&productID=H1483\_000\_P22.IMG&rule=retrieveBrowseImage&compression=lose&PROTOCOL=HTTP] ]>
```

#### 4.4.3 Metadata output – Reference Attributes

- Exactly one field **MUST** have one `utype="pds:<resource_class>.REFERENCE_FORMAT"` with `datatype="char"`, and `arraysize=""`, where `<resource_class>` could be `DATA_SET` or `PRODUCT`
- Exactly one field **MUST** have one `utype="pds:<resource_class>.PUBLISHER"` with `datatype="char"`, and `arraysize=""`, where `<resource_class>` could be `DATA_SET` or `PRODUCT`. A *PUBLISHER* is an entity responsible for making the resource available. Examples of a Publisher include a person, an organisation, or a service. Typically, the name of a Publisher should be used to indicate the entity.
- Exactly one field **MUST** have one `utype="pds:<resource_class>.CONTRIBUTOR"` with `datatype="char"`, and `arraysize=""`. A *CONTRIBUTOR* is an entity responsible for making contributions to the content of the resource. Examples of a Contributor include a person, an organisation, or a service. Typically, the name of a Contributor should be used to indicate the entity
- Exactly one field **MUST** have one `utype="pds:<resource_class>.PUBLISHING_DATE"` with `datatype="char"`, and `arraysize=""`.
- Exactly one field **MUST** have one `utype="pds:<resource_class>.RIGHTS"` with `datatype="char"`, and `arraysize=""`. The *RIGHTS* entry represent a person or organization

owning or managing rights over the resource. Recommended best practice is to use the URI or name of the Rights Holder to indicate the entity.

## 5 Map Projected Data Service Metadata Query (for requests of RESOURCE\_CLASS=MAP\_PROJECTED)

As already mentioned in point 3, the case of the map projected data access is a very special one in the Inter-planetary science world, as a lot of the information from Planetary science has historically come from e.g. images from the planetary bodies with associated geometrical information.

Rather than defining a new protocol for map projected data access, we have decided to include it in the more general PDAP protocol as a specific case. However, specific input/output parameters are required for this type of data. In particular, the Region Of Interest (ROI) of the observation must be defined for the map projected data access in order to characterize the geometrical information attached to the data.

Although we agree that the definition of these parameters might be difficult due to the very nature of the different planetary bodies, etc., we define a four-corners ROI search as compulsory input parameters, and a so called FOOTPRINT for the output parameters. This definition is quite close to OpenGIS Map Services but using specific planetary data characterization through PDS keywords and describing a standard way to access scientific data.

A *Map Projected Data Service* **MUST** implement the general PDAP input/output parameters plus the following ones:

### 5.1 Map Projected Data Service extended Input Parameters

An Map Projected Data Service **MUST** support the following parameters to define a Region Of Interest (ROI) for the product search:

#### 5.1.1 LONGITUDE

A service **MUST** implement a parameter to specify a LONGITUDE range (For range definition, please refer to appendix A)

For a Planetocentric, body-fixed, rotating coordinate system, longitude is defined as: The angle increasing eastward between the prime meridian and the vector from the coordinate system origin to the point of interest, projected into the

equatorial plane. This is a right-handed coordinate system. For a Planetographic, body-fixed, rotating coordinate system, longitude is defined as: The angle between the prime meridian and the vector from the coordinate system origin to the point of interest, projected into the equatorial plane. Planetographic longitudes are defined to increase with time for a distant observer. Thus, they increase to the west for prograde rotators, and to the east for retrograde rotators. For the Earth, Moon and Sun, PDS also supports the traditional use of the range (-180,180)

### 5.1.2 LATITUDE

A service **MUST** implement a parameter to specify a LATITUDE range (For range definition, please refer to appendix A). Combined with the LONGITUDE range, both parameters define a region.

For a Planetocentric, body-fixed, rotating coordinate system, latitude is defined as: The angle between the equatorial plane and a vector connecting the point of interest and the origin of the planetocentric coordinate system. Positive in the hemisphere north of the equator (i.e., hemisphere to the north of the solar system invariant plane) and negative in the southern hemisphere. For a Planetographic, body-fixed, rotating coordinate system, latitude is defined as: The angle between the equatorial plane and a vector through the point of interest that is normal to a biaxial ellipsoid reference surface. Positive in the hemisphere north of the equator (i.e., hemisphere to the north of the solar system invariant plane) and negative in the southern hemisphere.

## 5.2 Map Projected Data Service extended Output Fields

- Exactly one field **MUST** have one *ID*="PRODUCT.FOOTPRINT" with datatype="char", and arraysize="\*". The **FOOTPRINT** will be specified as follows:

- **Polygonal** regions

POLYGON(LON\_1:LAT\_1,LON\_2:LAT\_2,....,LON\_n:LAT\_n)

- **Circular** regions:

CIRCLE(CENTER\_LON:CENTER\_LAT,RADIUS)

- **Landmarks:**

POINT(LON:LAT)

- Exactly one field **MUST** have one *utype*="pds:PRODUCT.CENTER\_LONGITUDE" with *datatype*="char", and *arraysize*="\*".  
The *CENTER\_LONGITUDE* element provides a reference longitude for certain map projections. For example, in an Orthographic projection, the *center\_longitude* along with the *center\_latitude* defines the point or tangency between the sphere of the planet and the plane of the projection. The *map\_scale* (or *map\_resolution*) is typically defined at the *center\_latitude* and *center\_longitude*.
- Exactly one field **MUST** have one *utype*="pds:PRODUCT.CENTER\_LATITUDE" with *datatype*="char", and *arraysize*="\*".  
The *CENTER\_LATITUDE* element provides a reference latitude for certain map projections. For example, in an Orthographic projection, the *center\_latitude* along with the *center\_longitude* defines the point or tangency between the sphere of the planet and the plane of the projection. The *map\_scale* (or *map\_resolution*) is typically defined at the *center\_latitude* and *center\_longitude*.

According to the above set of parameters, a possible PDAP map projected data query could be as follows:

**Example:**

```
http://psa.esac.esa.int:8000/aio/jsp/metadata.jsp?TARGET_NAME=MARS
&LATITUDE=25/30&LONGITUDE=35/40&
RESOURCE_CLASS=MAP_PROJECTED&
RETURN_TYPE=HTML
```

See Appendix B for an example valid Map Projected Service response in VOTable (with positional parameters).

## 6 Service Metadata

This type of query gives as a result the list of parameters that the server accepts as inputs, which would be an expanded set composed of the compulsory parameters plus others that the server would understand, which are considered "OPTIONAL".

Due to the very nature of the data, different Data Sets might have different metadata optional values associated. This would mean that a

“RESOURCE\_CLASS=METADATA” query might need a further specification in the input parameters to do a finer grain filtering of the sought metadata parameters.

For the time being we will only specify a general metadata query, and will leave for further discussions which parameters might be used to filter the different cases depending on the Data Sets.

According to the aforementioned points, a prototypical query of this type would be like:

[http://psa.sciops.esa.int/PAIO/piap.jsp?RESOURCE\\_CLASS=METADATA](http://psa.sciops.esa.int/PAIO/piap.jsp?RESOURCE_CLASS=METADATA)

and a prototypical output could be something like:

```
<?xml version="1.0" ?>
<!DOCTYPE VOTABLE (View Source for full doctype...)>
-<VOTABLE version="1.1">
  -<RESOURCE type="results">
    <INFO name="PDAP_VERSION" value="0.3" />
    <DESCRIPTION>ESA Planetary Science Archive PDAP Query
      Service</DESCRIPTION>
    -<PARAM name=" INPUT:CENTER_LONGITUDE">
      <DESCRIPTION>Planetocentric Center longitude center of Region of
        Interest</DESCRIPTION>
      </PARAM>
    -<PARAM name=" INPUT:CENTER_LATITUDE">
      <DESCRIPTION>Planetocentric Center latitude center of Region of
        Interest</DESCRIPTION>
      </PARAM>
      [...]etc.... all compulsory input parameters should appear here

    -<PARAM name="INPUT:SUB_SOLAR_LONGITUDE">
      <DESCRIPTION>" The sub_solar_longitude element provides the
        longitude of the subsolar point. The subsolar point is that point on
        a body's reference surface where a line from the body center to the
        sun center intersects that surface. Note: The
        coordinate_system_type data element should be used in
        conjunction with this data element." </DESCRIPTION>
      </PARAM>

    -<PARAM name="INPUT:SUB_SOLAR_LATITUDE">
      <DESCRIPTION>"The sub_solar_latitude element provides the latitude
        of the subsolar point. The subsolar point is that point on a body's
        reference surface where a line from the body center to the sun
        center intersects that surface" </DESCRIPTION>
      </PARAM>

    [...] etc., any optional input search parameters accepted by the server.

  </RESOURCE>
</VOTABLE>
```

This type of query serves the purpose of unveiling the type of information that can be queried to the server. Smart clients can use this feature to build on-the-fly User Interfaces to fill in required parameters to perform complex queries to the server.

## 6.1 Registering a Compliant Service

Registration of PDAP services will be done through a standard IPDA registration service or a external registry associated to other VO initiatives. Although the technical details of the registration of PDAP services is out of the scope of the present document, the following metadata would be required for registration:

- **Title:** The name of the PDAP service. There are no restrictions on the format of the value; however, it is recommended that if the service is associated with a particular data collection, the service Title contain a recognizable form of the collection's name.
- **Publisher:** The entity responsible for making the service available. This is usually the organization or data center name (e.g. "Space Telescope Science Institute/MAST"); however, it can be a person's name.
- **Contact.Name:** The name of a party (i.e. person or group) that can be contacted regarding the content, implementation, or maintenance of the service.
- **Contact.Email:** The email address of a party (i.e. person or group) that can be contacted regarding the content, implementation, or maintenance of the service.
- **ReferenceURL:** A URL pointing to additional information about the service or the PDAP data provided by the service. This can be a human-readable document specifically describing this service, or it can just be the home page for the data provider.

## 7 PDAP version information

A PDAP compliant output **SHOULD** contain an INFO tag with name PDAP\_VERSION and value the PDAP specification version implemented by the service. This INFO tag must be inside the RESOURCE tag due to VOTable compatibility. (see RESOURCE\_CLASS=METADATA output as an example). This information should appear in a RESOURCE\_CLASS=METADATA discovery query.

This tag has two different uses:

- Provide information to client applications about the version of the PDAP specification implemented by the PDAP server so clients could prevent parsing problems.
- Allow automatically discover version information by registry servers.

**Example:**

```
<INFO name="PDAP_VERSION" value="0.5"/>
```

See section 6 for a more detailed example.

## 8 How to design your RESOURCE\_CLASS

PDAP is based in the resource class concept, i.e. a specific class (data set, product, map\_projected, etc) requires specific metadata to be handled properly at client side. However, some other extensions are possible and to include all of them in current specification would require a permanent living document.

In order to allow future extensions, the following rules should be followed to create a new resource class that better describe your data:

- Select a new resource class identifier. Current reserved values are DATA\_SET, PRODUCT and MAP\_PROJECTED
- Identify PDS keywords that could be used to characterize the data
- A standard resource class document extension should be created and raised to IPDA level for review. If considered of general use, this new resource class definition would be incorporated as a standard IPDA specification.

## 9 Appendix A: Range definition for input parameters

For a simple HTTP GET/POST service, the constraints in the queries for the input parameters need to be done in a valid URL.

However, the URL standard is not very specific defining special constraints as, e.g., ranges. We describe here some examples of accepted range syntax for URL PDAP services:

<b>param = x</b>	Equality	Equivalent to param=x
<b>param = x/y</b>	Closed range	Equivalent to param>=X AND param<=y
<b>param = x/</b>	Open range	Equivalent to param>=x
<b>param = x,y</b>	List	Equivalent to param=x OR param=y

<b>param = x/a,b/y</b>	Range list	Equivalent to [param<=x AND param<=a] OR [param>=b and param<=y]
------------------------	------------	--

## 10 Appendix B: Valid Map Projected Data Service response in VOTable

```

<?xml version="1.0" ?>
<!DOCTYPE VOTABLE (View Source for full doctype...)>
<VOTABLE version="1.1">
  <RESOURCE type="results">

    <INFO name="PDAP_VERSION" value="0.3"/>

    <DESCRIPTION>PSA Metadata Query Service</DESCRIPTION>
    <INFO name="QUERY_STATUS" value="OK" />
    <TABLE>

<FIELD ID="PRODUCT.PRODUCT_ID" ucd="PRODUCT_ID" utype="pds:PRODUCT.PRODUCT_ID"
datatype="char" arraysize="*" />

<FIELD ID="PRODUCT.DATA_ACCESS_REFERENCE" ucd="DATA_ACCESS_REFERENCE"
datatype="char" arraysize="*" />

<FIELD ID="PRODUCT.FOOTPRINT" ucd="FOOTPRINT"
datatype="char" arraysize="*" />

<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.MISSION_NAME" ucd="MISSION_NAME" utype="pds:DATA_SET.MISSION_NAME"
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="PRODUCT.TARGET_NAME" ucd="TARGET_NAME" utype="pds:PRODUCT.TARGET_NAME"
datatype="char" arraysize="*" />

<FIELD ID="PRODUCT.TARGET_TYPE" ucd="TARGET_TYPE" utype="pds:PRODUCT.TARGET_TYPE"
datatype="char" arraysize="*" />

<FIELD ID="PRODUCT.INSTRUMENT_ID" ucd="INSTRUMENT_ID" utype="pds:PRODUCT.INSTRUMENT_ID"
datatype="char" arraysize="*" />

<FIELD ID="PRODUCT.INSTRUMENT_NAME" ucd="INSTRUMENT_NAME"
utype="pds:PRODUCT.INSTRUMENT_NAME" datatype="char" arraysize="*" />

<FIELD ID="PRODUCT.START_TIME" ucd="START_TIME" utype="pds:PRODUCT.START_TIME"
datatype="char" arraysize="*" />

<FIELD ID="PRODUCT.STOP_TIME" ucd="STOP_TIME" utype="pds:PRODUCT.STOP_TIME"
datatype="char" arraysize="*" />

<FIELD ID="PRODUCT.ICON_ACCESS_REFERENCE" ucd="ICON_ACCESS_REFERENCE" datatype="char"
arraysize="*" />

<FIELD ID="RESOURCE_CLASS" ucd="RESOURCE_CLASS" datatype="char" arraysize="*" />

<DATA>

```

```

<TABLEDATA>
<TR>
<TD>H1428_0000_BL2.IMG</TD>
</TD>
<![CDATA[
http://psa.esac.esa.int:8000/aio/jsp/product.jsp?dataSetID=MEX-M-HRSC-3-RDR-
V2.0&productID=H1428_0000_BL2.IMG&PROTOCOL=HTTP
]]>
</TD>

<TD>POLYGON(30.941999435424805:50.5369987487793,30.961999893188477:50.29100036621094,30.9
81000900268555:50.04399871826172,31.000999450683594:49.797000885009766,31.020000457763672
: [more polygon data]
84000015258789:49.2239990234375,28.820999145507812:49.47600173950195,28.802000045776367:4
9.724998474121094,28.783000946044922:49.97100067138672,28.76300048828125:50.2169990539550
8,28.742000579833984:50.4630012512207)</TD>
<TD>MEX-M-HRSC-3-RDR-V2.0</TD>
<TD>HRSC REDRS</TD>
<TD>MARS EXPRESS</TD>
<TD>Deutsches Zentrum fuer Luft und Raumfahrt (DLR), Institut fu</TD>
<TD>DLR</TD>
<TD />
<TD>MARS</TD>
<TD>SATELLITE,SATELLITE,PLANET</TD>
<TD>HRSC</TD>
<TD>HIGH RESOLUTION STEREO CAMERA</TD>
<TD>2005-02-26 14:36:19.316666666</TD>
<TD>2005-02-26 14:46:09.236666666</TD>
= <TD>
- <![CDATA[
http://psa.esac.esa.int:8000/aio/jsp/product.jsp?dataSetID=MEX-M-HRSC-3-RDR-
V2.0&productID=H1428_0000_BL2.IMG&rule=retrieveBrowseImage&compression=loose&protocol=HTT
P
]]>
</TD>
<TD>MAP_PROJECTED</TD>
</TR>
</TABLEDATA>
</DATA>
</TABLE>
</RESOURCE>
</VOTABLE>

```

## References

- [1] IPDA International Planetary Data Alliance,  
<http://planetarydata.org/>
- [2] Osuna P. et al, NASA-PDS/ESA-PSA Planetary Data Interoperability White Paper,  
<http://planetarydata.org/documents/white-paper-wp/>
- [3] Jeff de la Beaujardiere et al, OGC Web Map Service Interface. OGC standard,  
<http://www.opengeospatial.org/standards/wms>
- [4] Tody D, Plante R., Simple Image Access Protocol, IVOA Recommendation,  
<http://www.ivoa.net/Documents/SIA/>
- [5] NASA PDS, Planetary Data System Standards Reference,  
<http://pds.nasa.gov/tools/standards-reference.shtml>
- [6] Ochsenbein F, Williams R., VOTable Format Definition, IVOA Recommendation,  
<http://www.ivoa.net/Documents/VOTable/>

