

GEO-EN Search Service Pilot

Tom Stein , Dan Scholes, Ed Guinness
Jordan Padams, Dan Crichton

Nov 5, 2019

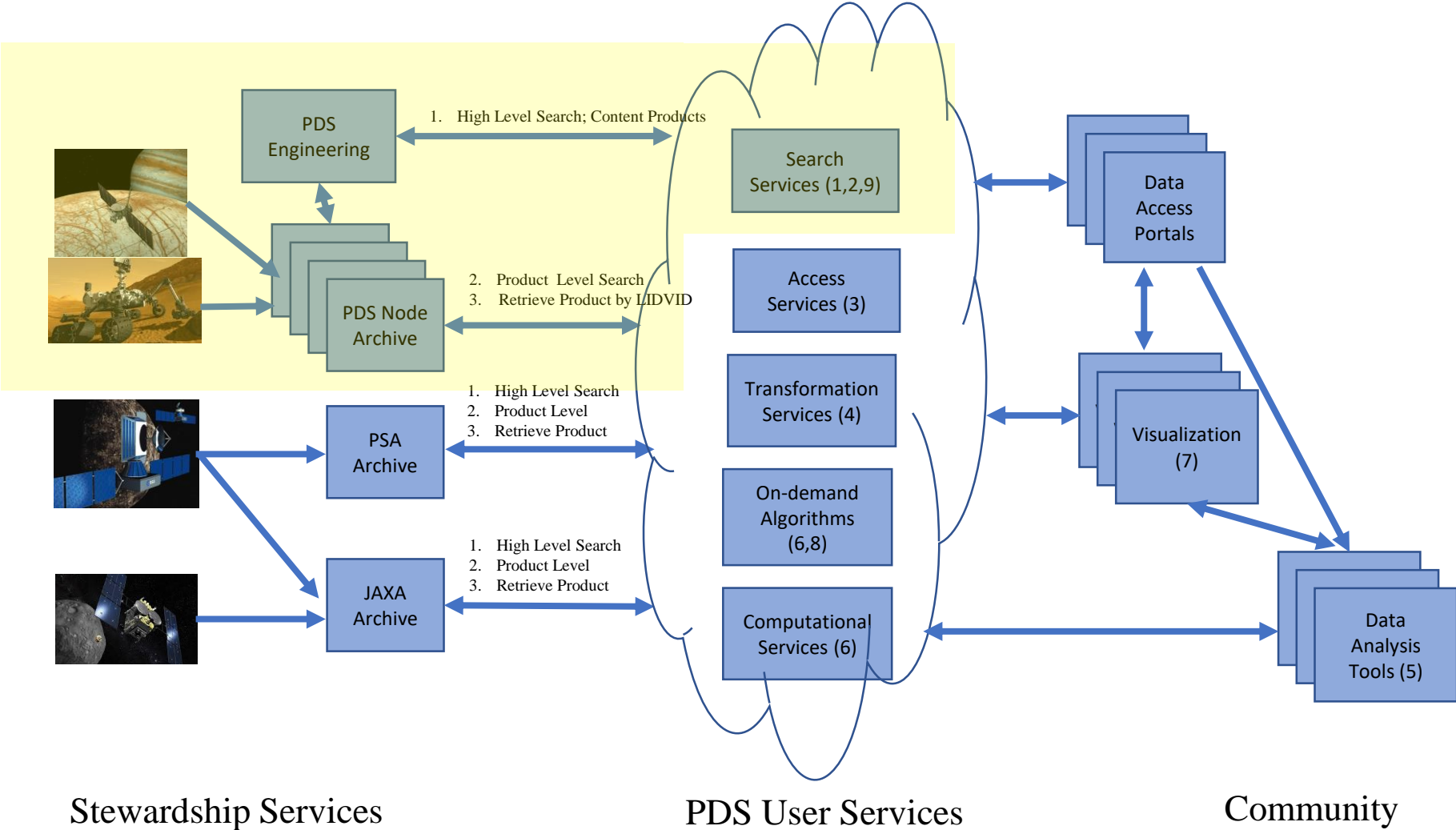
Purpose

- Demonstrate a search data service:
 - Using PDS API
 - Crossing node-EN boundary
 - Supporting node agnostic input
 - Providing machine readable output
- Responds to Roadmap findings
 - Finding 3 – Data discoverability (search services)
 - Finding 7 – Access to data (API development)

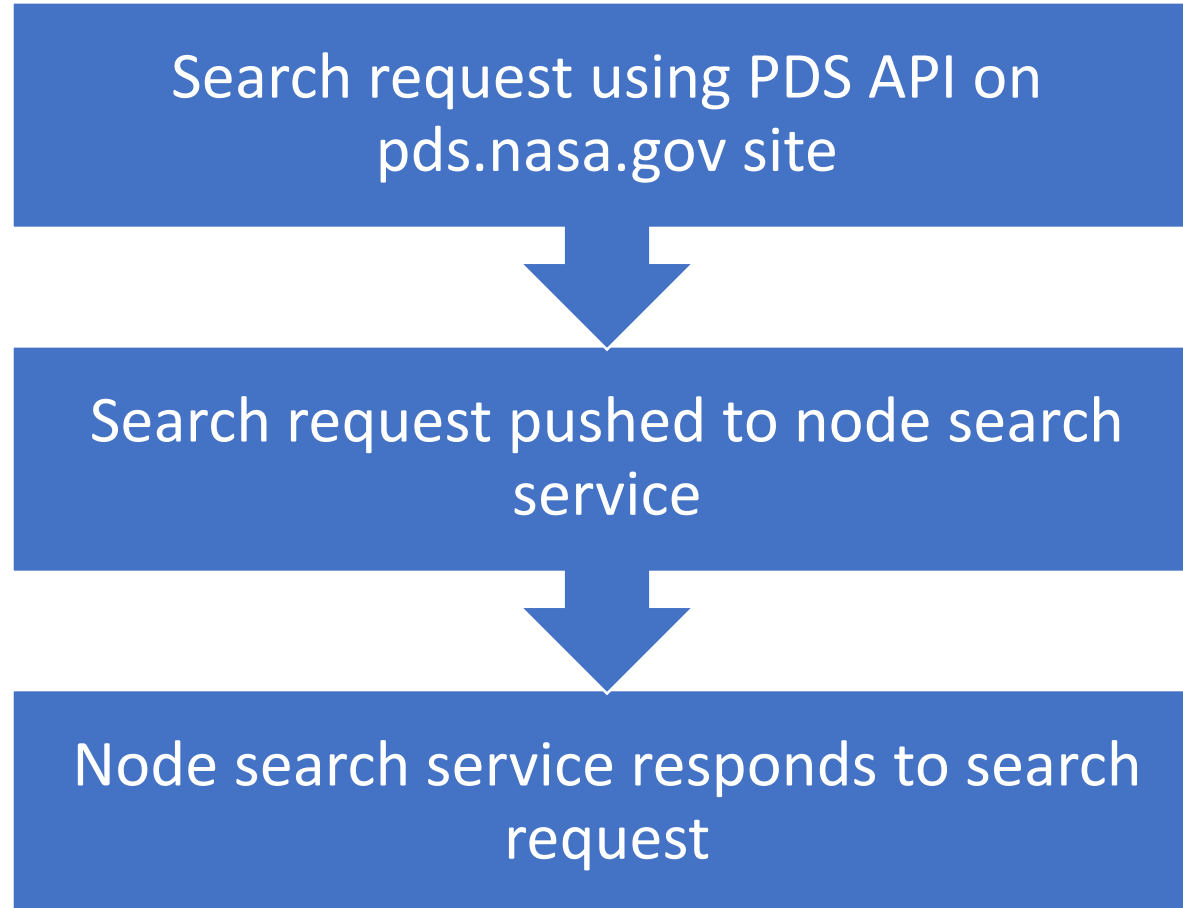
Purpose (2)

- But...the pilot is just a pilot.
 - It is a notional implementation that WILL change.
 - It is not a fixed template for future projects.
 - All nodes and additional stakeholders ultimately will be required to design the PDS API.

Data services pilot focus



Pilot search service request flow



Example use case

- User request for search results

Return the product ID and a URL to a web page showing detail (as JSON) for MRO CRISM TRDRs over Gale Crater on Mars restricted to the 10 most recently acquired products sorted by acquisition date, most recent first

Request example

- For use case

Return the **product metadata including URL to a web page showing detail** as **JSON**

for **MRO CRISM TRDRs over Gale Crater on Mars**

restricted to the 10 most recently acquired products sorted by acquisition date, most recent first

- Request

- **Request item** – Archive metadata (Product ID, detail page web URL)

- **Return type** – JSON

- **Search constraints** - MRO CRISM TRDRs over Gale Crater on Mars

- **Output constraints** – restricted to the 10 most recently acquired products sorted by acquisition date, most recent first

Request items (conceptual)

Archive metadata	Information specific to the content of the archive (e.g., the number of products, a list of LIDVIDs)
Browse	An image in common format (e.g., JPG, PNG, GeoTIFF) or data in common format (e.g., CSV, Excel spreadsheet) representative of a data product; the result may optionally be modified from the original: images by contrast stretching, band subsampling, and/or resizing, data by number of rows and/or columns
Data object	A PDS data object file
Data product	A PDS data product (data object(s) and label file(s))
Label	A PDS label file
Product metadata	Information specific to a data product (e.g., start time, footprint); may include values from the label or may be derived from the label or another source; content varies by node, collection, and service
Web detail page	A web page that provides detailed information about a data product. It is likely that the return will be a URL that will point into a web application like AN, ODE, Opus, etc.
Documentation	Document(s) from an archive volume or collection
Web tool search	A web-based search tool showing the search form with the given parameters filled in
Web tool results	A web-based search tool showing the results for given parameters

Return types

count	Number
files	A single file
	A single file that is a package of multiple files (zip or tar)
	Data that can be consumed as a data blob
metadata	ASCII formatted string <ul style="list-style-type: none">• CSV (Comma Separated Value)• JSON (JavaScript Object Notation)• ODL (Object Description Language)• XML (eXtensible Markup Language)

Request example

- For use case

Return the **product metadata including URL to a web page showing detail** as **JSON**

for **MRO CRISM TRDRs over Gale Crater on Mars**

restricted to the **10 most recently acquired products**

sorted by acquisition date, most recent first

- API request

<https://pilot.rsl.wustl.edu/api/v1/search/products/metadata/mars/mro/?instrument=crism&prodType=trdr&featuretype=crater&feature=gale&offset=0&page Size=10&sortkey=-observationEndUtc,productid&outputformat=json>

Examples - LIDVID

Get count of products with matching LID or LIDVID

https://pilot.rsl.wustl.edu/api/v1/search/products/lidvids/count/?lidvid=urn:nasa:pds:mess_mla_calibrated:data_gdr:hdec_45n_500m&outputformat=json

https://pilot.rsl.wustl.edu/api/v1/search/products/lidvids/count/?lidvid=urn:nasa:pds:mess_mla_calibrated:data_gdr:hdec_45n_500m::1.0&outputformat=json

Get metadata using LID or LIDVID

https://pilot.rsl.wustl.edu/api/v1/search/products/lidvids/metadata/?lidvid=urn:nasa:pds:mess_mla_calibrated:data_gdr:hdec_45n_500m&outputformat=json

https://pilot.rsl.wustl.edu/api/v1/search/products/lidvids/metadata/?lidvid=urn:nasa:pds:mess_mla_calibrated:data_gdr:hdec_45n_500m::1.0&outputformat=json

List product files using LIDVID

https://pilot.rsl.wustl.edu/api/v1/search/products/lidvids/metadata/files?lidvid=urn:nasa:pds:mess_mla_calibrated:data_gdr:hdec_45n_500m::1.0&outputformat=json

https://pilot.rsl.wustl.edu/api/v1/search/products/lidvids/metadata/files?lidvid=urn:nasa:pds:mess_mla_calibrated:data_gdr:hdec_45n_500m::1.0&outputformat=json&filetype=label

Download the data file/label for a product using LIDVID

https://pilot.rsl.wustl.edu/api/v1/search/products/lidvids/files?lidvid=urn:nasa:pds:mess_mla_calibrated:data_gdr:hdec_45n_500m::1.0&filetype=data

https://pilot.rsl.wustl.edu/api/v1/search/products/lidvids/files?lidvid=urn:nasa:pds:mess_mla_calibrated:data_gdr:hdec_45n_500m::1.0&filetype=labelpds3

https://pilot.rsl.wustl.edu/api/v1/search/products/lidvids/files?lidvid=urn:nasa:pds:mess_mla_calibrated:data_gdr:hdec_45n_500m::1.0&filetype=labelpds4

Examples – Planet features

List the feature types for Mars

<https://pilot.rsl.wustl.edu/api/v1/search/featuretypes/mars>

List features for Mercury

<https://pilot.rsl.wustl.edu/api/v1/search/features/mercury?featuretype=Albedo Feature&feature=Australia>

https://pilot.rsl.wustl.edu/api/v1/search/features/mercury?featuretype=Albedo Feature&feature=A*

<https://pilot.rsl.wustl.edu/api/v1/search/features/mercury/Albedo Feature>

Examples – CRISM data with geographic constraint

Get product counts

<https://pilot.rsl.wustl.edu/api/v1/search/products/count/mars/mro/crism/trdr/?minlat=20&maxlat=25&westernlon=30&easternlon=40>

<https://pilot.rsl.wustl.edu/api/v1/search/products/count/mars?mission=mro&instrument=crism&prodType=trdr&minlat=20&maxlat=25&westernlon=30&easternlon=40>

<https://pilot.rsl.wustl.edu/api/v1/search/products/count/mars/?mission=mro&instrument=crism&prodType=trdr&minlat=20&maxlat=25&westernlon=30&easternlon=40>

<https://pilot.rsl.wustl.edu/api/v1/search/products/count/mars/?mission=mro&instrument=crism&prodType=trdr&featuretype=crater&feature=gale>

Get product metadata

<https://pilot.rsl.wustl.edu/api/v1/search/products/metadata/mars/mro/crism/trdr/?offset=0&pageSize=100&minlat=20&maxlat=25&westernlon=30&easternlon=40&sortkey=productid&outputformat=json>

<https://pilot.rsl.wustl.edu/api/v1/search/products/metadata/mars/mro/?instrument=crism&prodType=trdr&offset=0&pageSize=100&minlat=20&maxlat=25&westernlon=30&easternlon=40&sortkey=productid&outputformat=json>

<https://pilot.rsl.wustl.edu/api/v1/search/products/metadata/mars/mro/?instrument=crism&prodType=trdr&featuretype=crater&feature=gale&offset=0&pageSize=10&sortkey=-observationEndUtc&productid&outputformat=json>

Examples – forward to search tool

Access Search tool – MSL AN Search

<https://pilot.rsl.wustl.edu/api/v1/search/products/searchTools/mars/?mission=msl&instrument=Mastcam&sol=102>

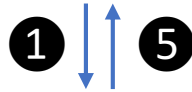
Bonus –

Download large file (5GB) Example

<https://pilot.rsl.wustl.edu/api/v1/search/downloadtest>

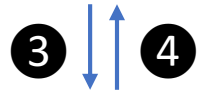
Information Flow

MRO CRISM
TRDRs over
Gale Crater
on Mars



Engineering
Node

2



GEO Node



- 1 Parse user's text query
- 2 Run Named Entity Recognizer
MRO -> Mission
CRISM -> Instrument
TRDRs -> Product Type
Gale Crater -> Feature
Mars -> Target
- 3 Make web service call(s) to GEO Node
- 4 Process GEO response
- 5 Generate user response (JSON)

Search Geo API

Test service through PDS gamma site (requires tunneling into JPL VPN)

<https://pds-gamma.jpl.nasa.gov/data-services/geo-demo/>

Test input searches

MRO CRISM data

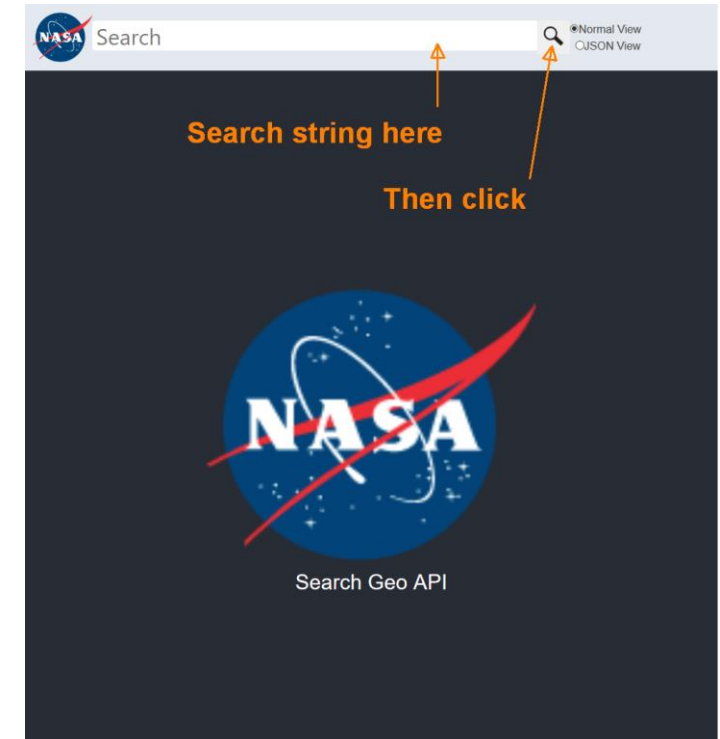
MRO CRISM TRDRs over Gale Crater on Mars

MESSENGER LIDs

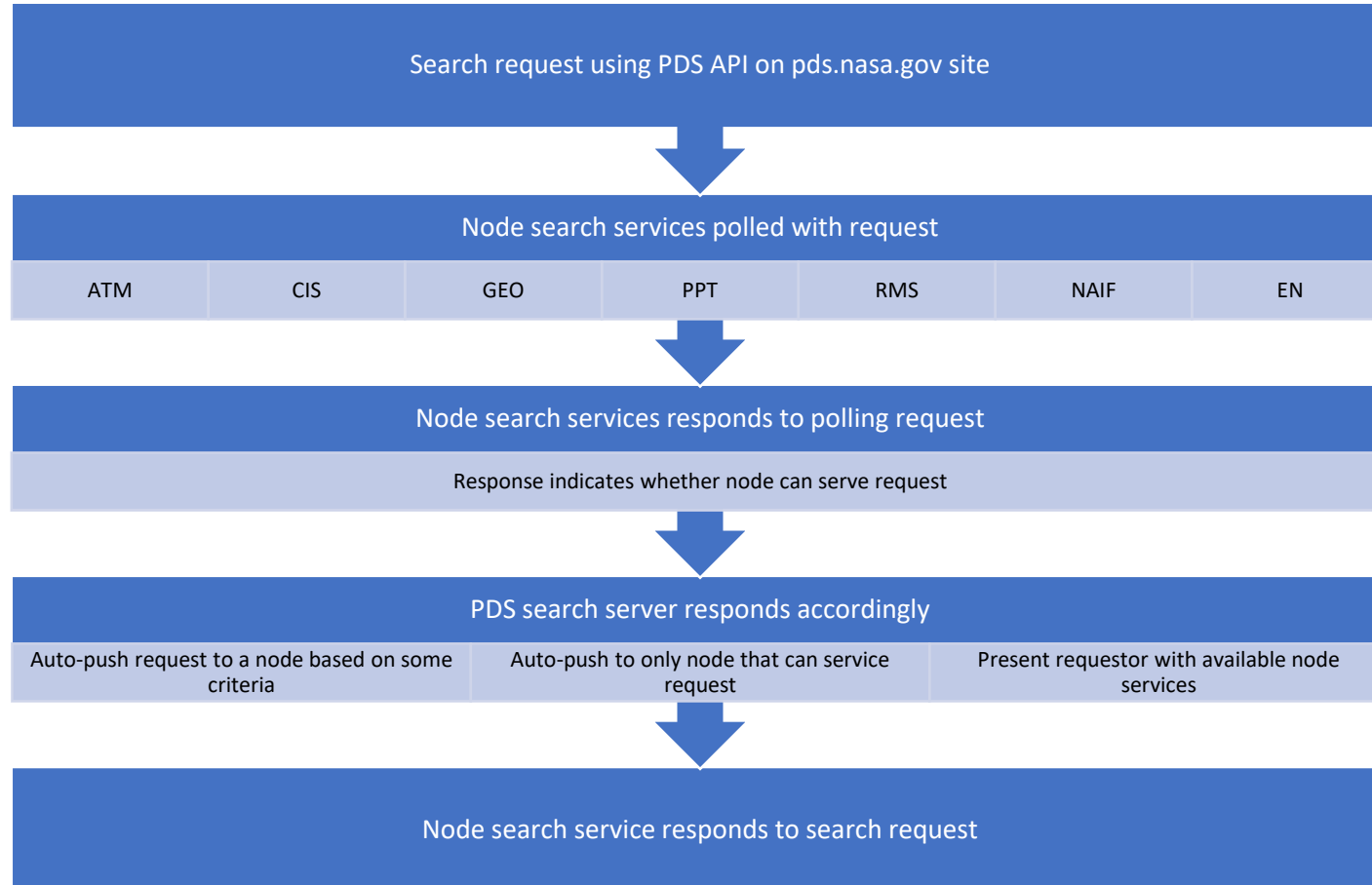
urn:nasa:pds:mess_grs_derived:data_cdr:grs_cs22011081zzz_dat

MSL Mastcam data

msl mastcam images from sol 102



Search service request flow for multiple nodes



PDS API development

- PDS Search API Specification
 - <https://github.com/NASA-PDS-Incubator/pds-api-server/wiki/PDS-Search-API-Specification>
 - Pilot project serves as catalyst to developing PDS-wide API.
 - EN drafted initial guidelines and approach.
 - Follow common best practices.
 - Make easy and intuitive.
 - Adhere to Open API specs.

Query parameter guidelines

- Due to the varying types of data available in the PDS, the applicability and use of certain query parameters will vary dramatically. However, the goals are:
 - To define a common syntax and best practice for defining query parameters
 - To define a *common set of query parameters* for top-level search criteria with the widest-ranging applicability across all PDS data
 - To leverage the PDS4 Information Model as a central driver and repository for the definition of query parameters.
 - To define a set of best practices and procedures for *extending the common set of query parameters* with node-specific query parameters
 - To provide a *managed central location* for describing and documenting all query parameters across the PDS.



Challenges

- Many common query parameters are not PDS4 attributes at this time.
 - Example: a user querying for “target=Mars” is a combination of Target_Identification.type = planet and Target_Identification.name = Mars
 - Although node to node queries probably would use the Mars LID, we cannot expect all outside users to do so.
- Designing for compatibility with PDS4 IM
 - IM uses underscores to separate words in attributes, though API best practice is to use camelCase.
- Metadata returned for products will vary by node, mission, instrument, and product type.
 - User will have to be informed that variations exist and why.
 - How do we capture features in the IM / Registry?
 - How do we capture other metadata other nodes are producing that are not in the IM / Registry?

Search API considerations

- Even simple use cases can have complex looking API calls
- How to handle versioning
- What objects should API call return
 - Blocks with header, input, data?
- What to return for null set / not found / cannot serve query
- Error handling
- GET length limit

Resources

PDS Search API Specification

<https://github.com/NASA-PDS-Incubator/pds-api-server/wiki/PDS-Search-API-Specification>

Pilot API base URL

<https://pilot.rsl.wustl.edu/>

Pilot API documentation

<https://pilot.rsl.wustl.edu/Help>