

Towards DataLink-next

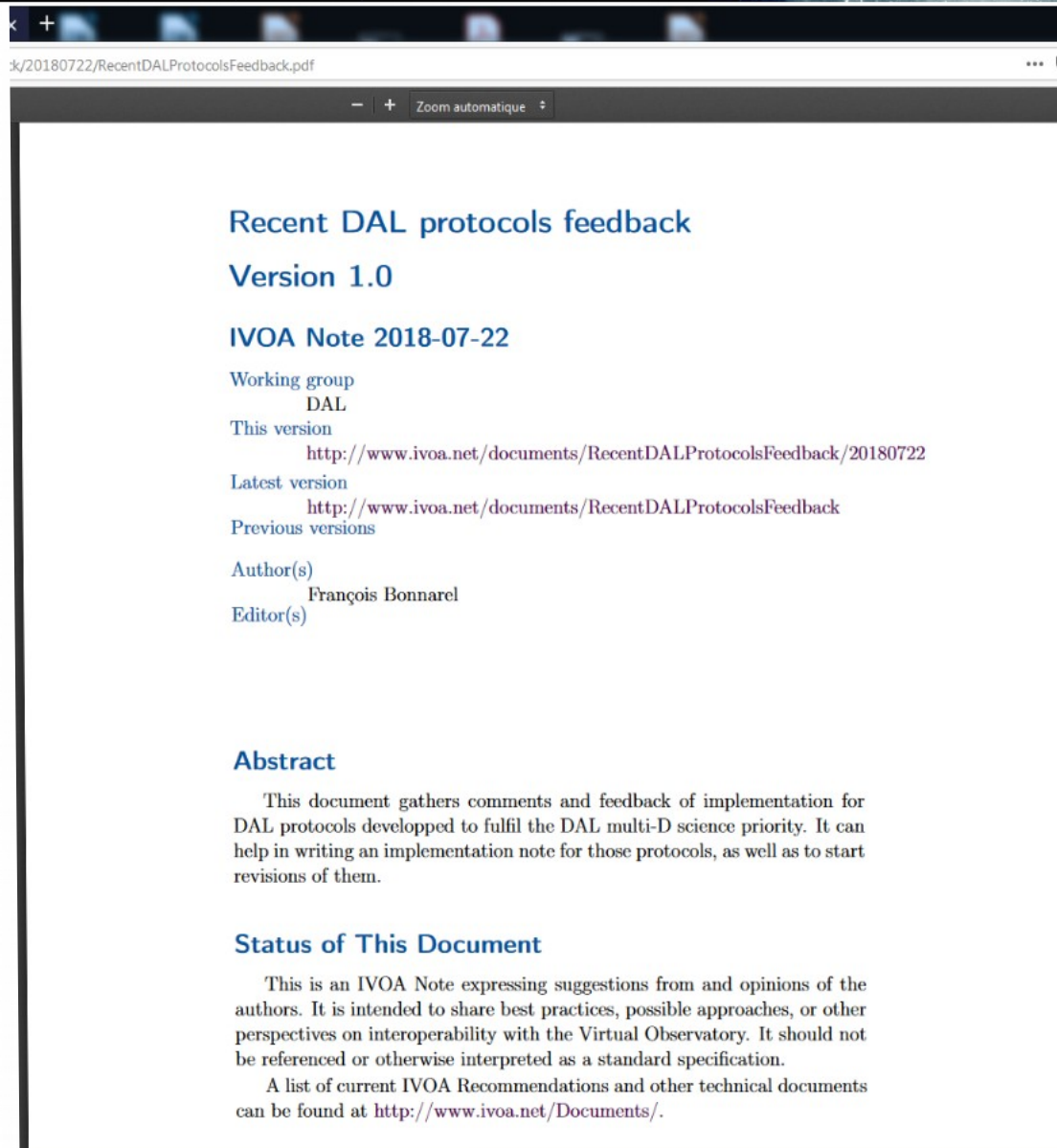


F. Bonnarel (CDS)

acknowledges the whole IVOA DAL Working Group for previous discussions



Ideas and rationale expressed in an IVOA note last summer



The screenshot shows a PDF document viewer with the following content:

Recent DAL protocols feedback
Version 1.0
IVOA Note 2018-07-22

Working group
DAL

This version
<http://www.ivoa.net/documents/RecentDALProtocolsFeedback/20180722>

Latest version
<http://www.ivoa.net/documents/RecentDALProtocolsFeedback>

Previous versions

Author(s)
François Bonnarel

Editor(s)

Abstract

This document gathers comments and feedback of implementation for DAL protocols developed to fulfil the DAL multi-D science priority. It can help in writing an implementation note for those protocols, as well as to start revisions of them.

Status of This Document

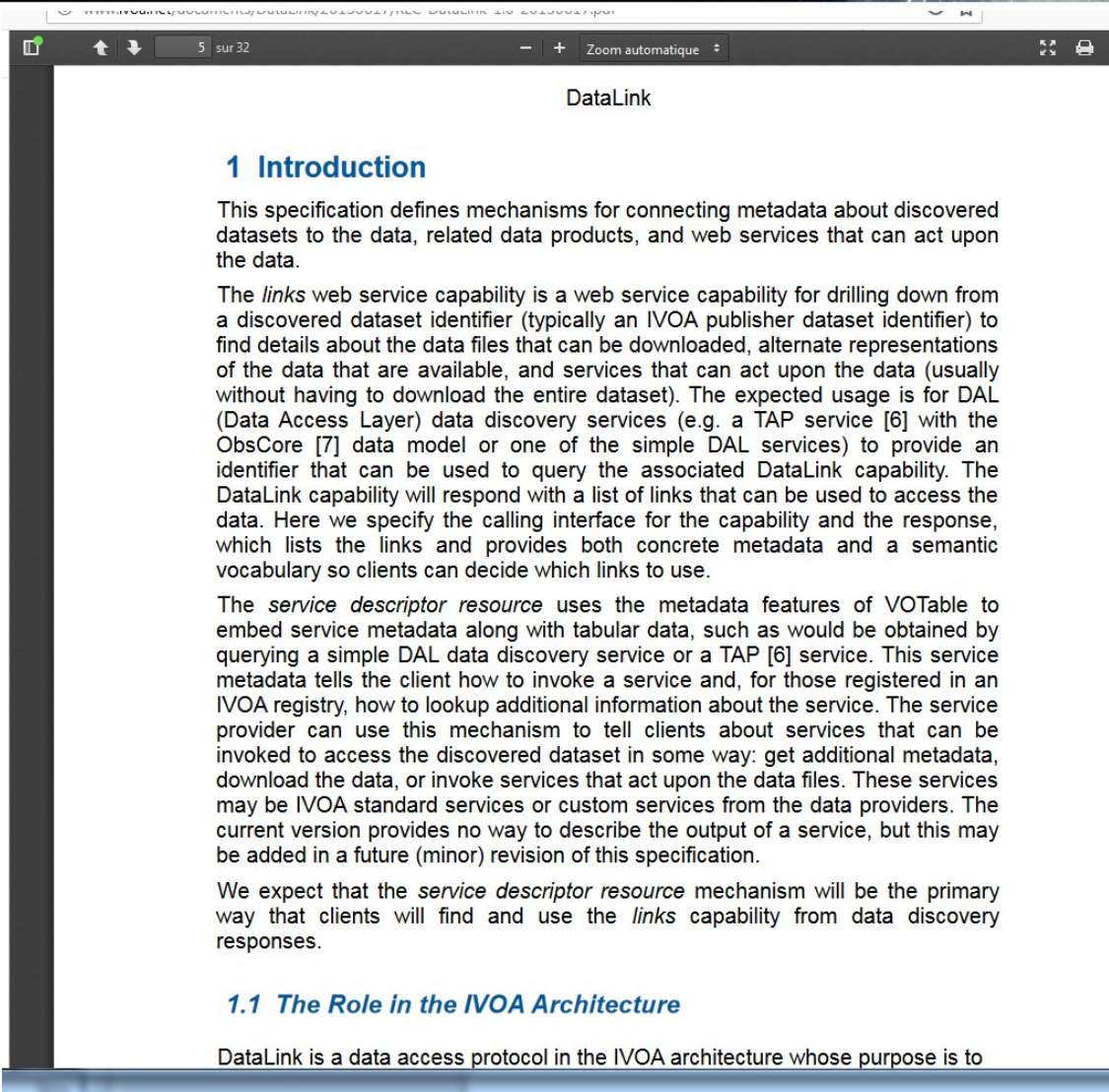
This is an IVOA Note expressing suggestions from and opinions of the authors. It is intended to share best practices, possible approaches, or other perspectives on interoperability with the Virtual Observatory. It should not be referenced or otherwise interpreted as a standard specification.

A list of current IVOA Recommendations and other technical documents can be found at <http://www.ivoa.net/Documents/>.

Section 2 of this document summarizes all the changes possibly wished for DataLink from feedback in the last two years



Change 1



The screenshot shows a web browser window with the address bar containing a URL. The page title is "DataLink". The main content is titled "1 Introduction" and contains several paragraphs of text. The text discusses mechanisms for connecting metadata to data, the role of the *links* web service capability, and the use of *service descriptor resource* metadata. A sub-section "1.1 The Role in the IVOA Architecture" is also visible, followed by the start of a paragraph: "DataLink is a data access protocol in the IVOA architecture whose purpose is to".

DataLink

1 Introduction

This specification defines mechanisms for connecting metadata about discovered datasets to the data, related data products, and web services that can act upon the data.

The *links* web service capability is a web service capability for drilling down from a discovered dataset identifier (typically an IVOA publisher dataset identifier) to find details about the data files that can be downloaded, alternate representations of the data that are available, and services that can act upon the data (usually without having to download the entire dataset). The expected usage is for DAL (Data Access Layer) data discovery services (e.g. a TAP service [6] with the ObsCore [7] data model or one of the simple DAL services) to provide an identifier that can be used to query the associated DataLink capability. The DataLink capability will respond with a list of links that can be used to access the data. Here we specify the calling interface for the capability and the response, which lists the links and provides both concrete metadata and a semantic vocabulary so clients can decide which links to use.

The *service descriptor resource* uses the metadata features of VOTable to embed service metadata along with tabular data, such as would be obtained by querying a simple DAL data discovery service or a TAP [6] service. This service metadata tells the client how to invoke a service and, for those registered in an IVOA registry, how to lookup additional information about the service. The service provider can use this mechanism to tell clients about services that can be invoked to access the discovered dataset in some way: get additional metadata, download the data, or invoke services that act upon the data files. These services may be IVOA standard services or custom services from the data providers. The current version provides no way to describe the output of a service, but this may be added in a future (minor) revision of this specification.

We expect that the *service descriptor resource* mechanism will be the primary way that clients will find and use the *links* capability from data discovery responses.

1.1 The Role in the IVOA Architecture

DataLink is a data access protocol in the IVOA architecture whose purpose is to

- General introduction :
 - extend the scope of "DataLinking" outside the original niche of resources attached to dataset.
 - → any VO resource ?



Change 2

1.2 Motivating Use Cases

Below are some of the more common use cases that have motivated the development of the DataLink specification. While this is not complete, it helps to understand the problem area covered by this specification.

- 6 -

DataLink

1.2.1 Multiple Files per Dataset

It is very common for a single dataset to be physically manifest as multiple files of various types. With a DataLink web service, the client can drill down using a discovered dataset identifier and obtain links to download one or more data files. For static data files, the DataLink service will be able to provide a URL as well as the content-type and content-length (file size) for each download.

1.2.2 Progenitor Dataset

In some cases, the data provider may wish to provide one or more links to progenitor (input) datasets; this would enable the users to drill down to input data in order to better understand the content of the product dataset, possibly reproduce the product to evaluate the processing, or reprocess it with different parameters or software.

1.2.3 Alternate Representations

For some datasets (large ones) it is useful to be able to access preview data (either precomputed or generated on-the-fly) and use it to determine if the entire dataset should be downloaded (e.g. in an interactive session). A DataLink service can provide links to previews as a URL with a specific relationship to the dataset and include other metadata like content-type (e.g. image/png) and content-length

- as a matter of consequence of change 1 :
- add use case in 1.2 such as
 - attaching datasets (Images, spectra, time series ...) to sources in a catalog.
 - and probably others?



Change 3

in separate specifications.

1.2.7 Recursive DataLink

In some cases, a dataset may contain many files (as in 1.2.1 above) and the provider may wish to make some files directly accessible and other (less important) files only accessible via additional calls. Such organisation of links could be accomplished by including a link to another DataLink service in the initial DataLink response (e.g. recursive DataLink). This service link would be described with both a service type (as in 1.2.4) and content type.

- 8 -

2 Resources

The DataLink web service capability is implemented as an HTTP REST [11] web service capability that conforms to the DALI-sync resource [1] description. The {links} resource is described fully below (see 2.1). The values for the identifiers are typically found using a data discovery service (e.g. TAP with ObsCore). Data discovery services that work with an associated DataLink service should include a resource in the discovery response to describe the DataLink service. The mechanism for doing this is described below (see 4).

The requirements for a standalone DataLink service are given in the table below.

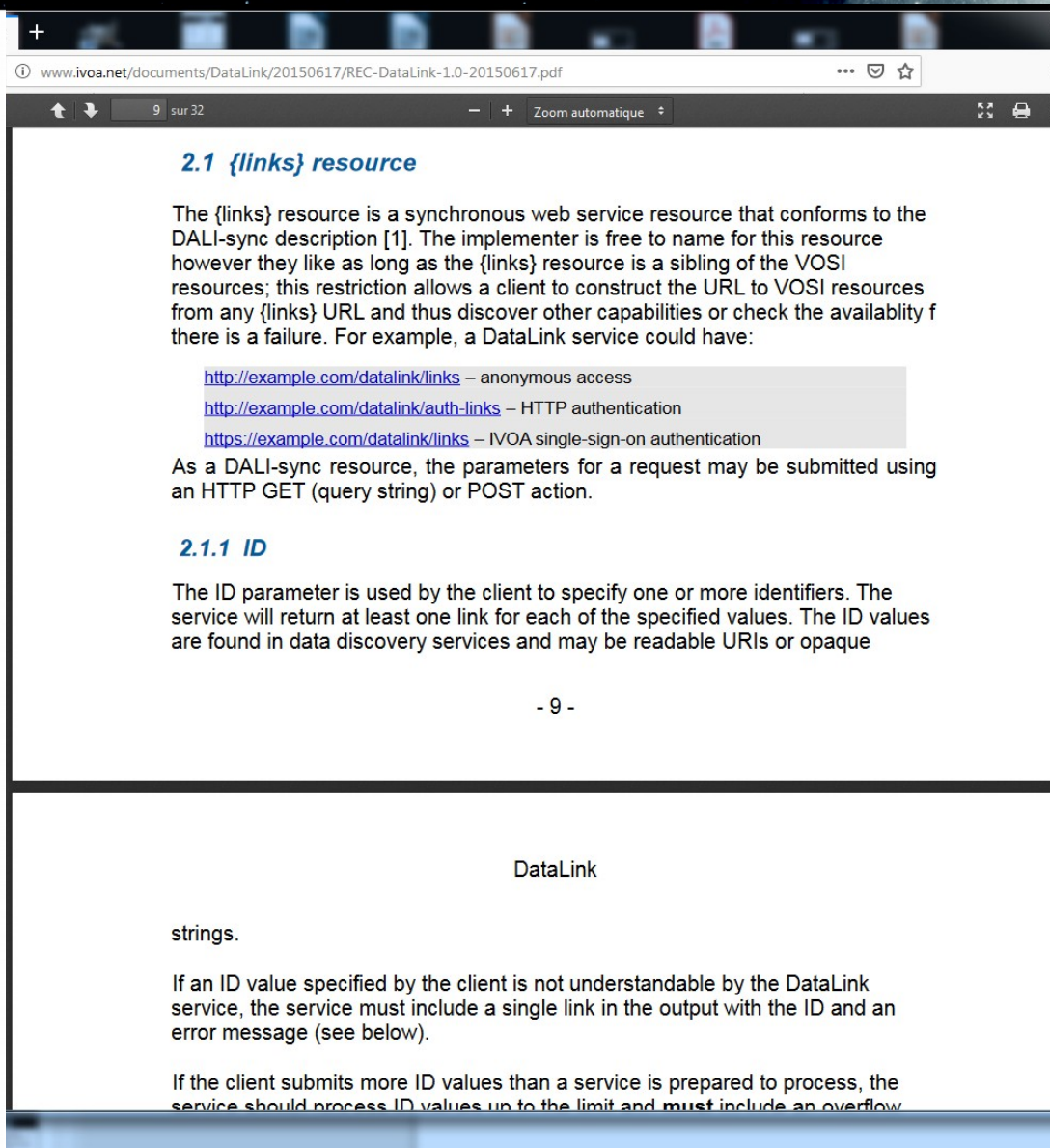
resource type	resource name	required
---------------	---------------	----------

Introduce a new DataLinkrecognition section between section 1 and section 2 helping readers to

- figure out how it works and the relationships between main response, links resource and service descriptors.
- Identify the 3 solutions of recognition according to their best suited context: Acces.reference/Access.format doublet, service descriptor and the new LINK mechanism. Clarify various context of usage of Service descriptors.



Change 4



The screenshot shows a PDF document titled "DataLink-1.0-20150617.pdf" from the IVOA website. The document is displayed in a browser window with a toolbar at the top showing navigation and zoom controls. The main content of the page is as follows:

2.1 {links} resource

The {links} resource is a synchronous web service resource that conforms to the DALI-sync description [1]. The implementer is free to name for this resource however they like as long as the {links} resource is a sibling of the VOSI resources; this restriction allows a client to construct the URL to VOSI resources from any {links} URL and thus discover other capabilities or check the availability of there is a failure. For example, a DataLink service could have:

- <http://example.com/datalink/links> – anonymous access
- <http://example.com/datalink/auth-links> – HTTP authentication
- <https://example.com/datalink/links> – IVOA single-sign-on authentication

As a DALI-sync resource, the parameters for a request may be submitted using an HTTP GET (query string) or POST action.

2.1.1 ID

The ID parameter is used by the client to specify one or more identifiers. The service will return at least one link for each of the specified values. The ID values are found in data discovery services and may be readable URIs or opaque strings.

If an ID value specified by the client is not understandable by the DataLink service, the service must include a single link in the output with the ID and an error message (see below).

If the client submits more ID values than a service is prepared to process, the service should process ID values up to the limit and **must** include an overflow

- In section 3 « {links} Response »
 - insist on the occurrence of an INFO tag
 - identifying the response as DataLink consistent
 - reference to DALI.

- 9 -

DataLink

strings.

If an ID value specified by the client is not understandable by the DataLink service, the service must include a single link in the output with the ID and an error message (see below).

If the client submits more ID values than a service is prepared to process, the service should process ID values up to the limit and **must** include an overflow



Change 5

resource. This resource describes a service as specified in section 4 . For example, if the response document includes this resource to describe a service:

```
<RESOURCE type="service" utype="ad hoc:service" ID="srv1">
...
</RESOURCE>
```

then the `service_def` column would contain `srv1` to indicate that a resource with XML ID `srv1` in the same document describes the service. Note that service descriptors do not always require an XML ID value; it is only the reference from `service_def` that warrants adding an ID to the descriptor.

3.2.4 error_message

The `error_message` column is used when no `accessURL` can be generated for an input identifier. If an `error_message` is included in the output, the only other columns with values should be the ID column and the semantics column; all others should be null.

3.2.5 description

The `description` column should contain a human-readable description of the link; it is intended for display by interactive applications.

3.2.6 semantics

The semantics column contains a single term from an external RDF vocabulary that describes the meaning of this linked resource relative to the identified dataset. The semantics column is intended to be machine-readable and assist automating data retrieval and processing.

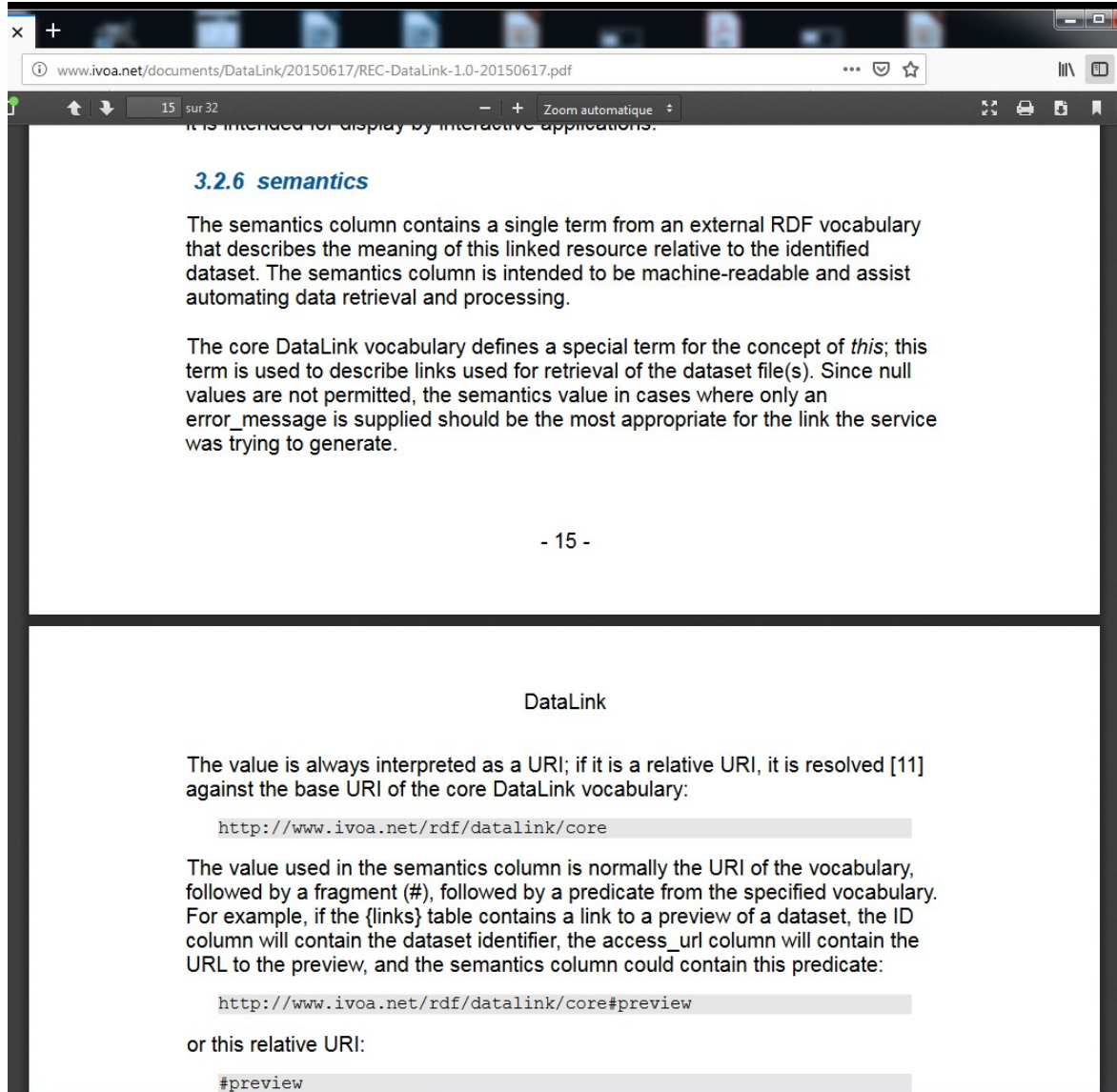
The core DataLink vocabulary defines a special term for the concept of *this*; this term is used to describe links used for retrieval of the dataset file(s). Since null values are not permitted, the semantics value in cases where only an `error_message` is supplied should be the most appropriate for the link the service was trying to generate.

15

- 3.2.5 : description :
- Add a SHOULD for this field to be given a content.



Change 7



www.ivoa.net/documents/DataLink/20150617/REC-DataLink-1.0-20150617.pdf

3.2.6 semantics

The semantics column contains a single term from an external RDF vocabulary that describes the meaning of this linked resource relative to the identified dataset. The semantics column is intended to be machine-readable and assist automating data retrieval and processing.

The core DataLink vocabulary defines a special term for the concept of *this*; this term is used to describe links used for retrieval of the dataset file(s). Since null values are not permitted, the semantics value in cases where only an `error_message` is supplied should be the most appropriate for the link the service was trying to generate.

- 15 -

DataLink

The value is always interpreted as a URI; if it is a relative URI, it is resolved [11] against the base URI of the core DataLink vocabulary:

```
http://www.ivoa.net/rdf/datalink/core
```

The value used in the semantics column is normally the URI of the vocabulary, followed by a fragment (#), followed by a predicate from the specified vocabulary. For example, if the {links} table contains a link to a preview of a dataset, the ID column will contain the dataset identifier, the `access_url` column will contain the URL to the preview, and the semantics column could contain this predicate:

```
http://www.ivoa.net/rdf/datalink/core#preview
```

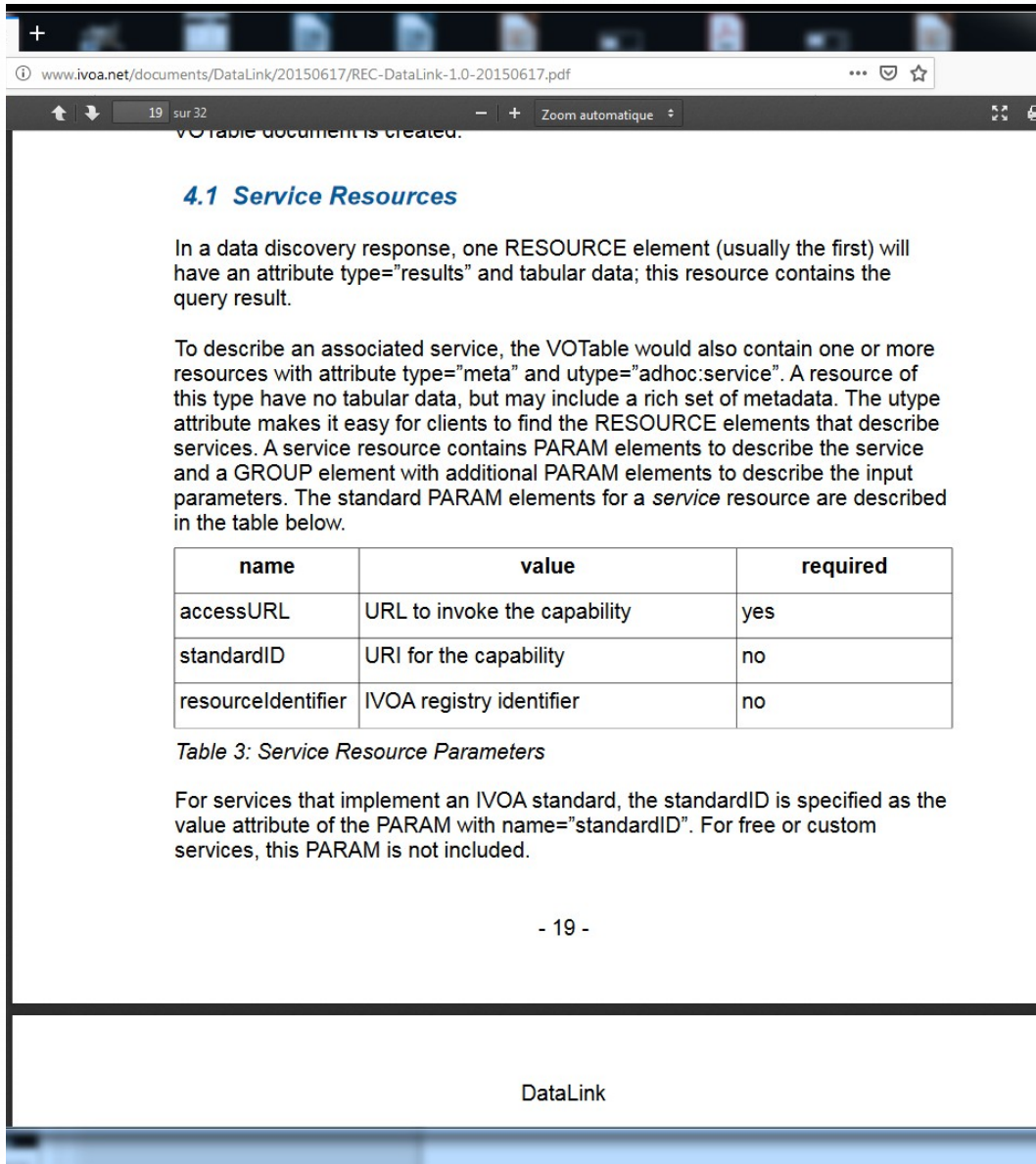
or this relative URI:

```
#preview
```

extend semantics

- managed by semantics in the vocabulary Web Page
 - metadata : information associated to the dataset
 - fault : semantic field for rows with error messages
 - science_alternate : transformed dataset (eg : rebinned)
 - alt : "this" in a different format
 - proc : definition change from "service-side data processing result" to "reference to a server-side processing service"
 - New associated_data branch
 - associated_image
 - associated_spectrum

Change 8



www.ivoa.net/documents/DataLink/20150617/REC-DataLink-1.0-20150617.pdf

VOtable document is created.

4.1 Service Resources

In a data discovery response, one RESOURCE element (usually the first) will have an attribute type="results" and tabular data; this resource contains the query result.

To describe an associated service, the VOTable would also contain one or more resources with attribute type="meta" and utype="ad hoc:service". A resource of this type have no tabular data, but may include a rich set of metadata. The utype attribute makes it easy for clients to find the RESOURCE elements that describe services. A service resource contains PARAM elements to describe the service and a GROUP element with additional PARAM elements to describe the input parameters. The standard PARAM elements for a *service* resource are described in the table below.

name	value	required
accessURL	URL to invoke the capability	yes
standardID	URI for the capability	no
resourceIdentifier	IVOA registry identifier	no

Table 3: Service Resource Parameters

For services that implement an IVOA standard, the standardID is specified as the value attribute of the PARAM with name="standardID". For free or custom services, this PARAM is not included.

- 19 -

DataLink

- 4.1 Service resources.
 - Impose the service descriptor to be upfront the RESOURCE it is related to.
 - Nest the RESOURCES in case of multi "type=results" Responses to avoid ambiguity.



Change 9

www.ivoa.net/documents/DataLink/20150617/REC-DataLink-1.0-20150617.pdf

VOTable document is created.

4.1 Service Resources

In a data discovery response, one RESOURCE element (usually the first) will have an attribute type="results" and tabular data; this resource contains the query result.

To describe an associated service, the VOTable would also contain one or more resources with attribute type="meta" and utype="ad hoc:service". A resource of this type have no tabular data, but may include a rich set of metadata. The utype attribute makes it easy for clients to find the RESOURCE elements that describe services. A service resource contains PARAM elements to describe the service and a GROUP element with additional PARAM elements to describe the input parameters. The standard PARAM elements for a *service* resource are described in the table below.

name	value	required
accessURL	URL to invoke the capability	yes
standardID	URI for the capability	no
resourceIdentifier	IVOA registry identifier	no

Table 3: Service Resource Parameters

For services that implement an IVOA standard, the standardID is specified as the value attribute of the PARAM with name="standardID". For free or custom services, this PARAM is not included.

- 19 -

DataLink

- 4.1 again.
- Recommend (SHOULD status) to add a free name to the service descriptor to help identifying the meaning.



Change 10

www.ivoa.net/documents/DataLink/20150617/REC-DataLink-1.0-20150617.pdf

19 sur 32

VOTable document is created.

4.1 Service Resources

In a data discovery response, one RESOURCE element (usually the first) will have an attribute type="results" and tabular data; this resource contains the query result.

To describe an associated service, the VOTable would also contain one or more resources with attribute type="meta" and utype="ad hoc:service". A resource of this type have no tabular data, but may include a rich set of metadata. The utype attribute makes it easy for clients to find the RESOURCE elements that describe services. A service resource contains PARAM elements to describe the service and a GROUP element with additional PARAM elements to describe the input parameters. The standard PARAM elements for a *service* resource are described in the table below.

name	value	required
accessURL	URL to invoke the capability	yes
standardID	URI for the capability	no
resourceIdentifier	IVOA registry identifier	no

Table 3: Service Resource Parameters

For services that implement an IVOA standard, the standardID is specified as the value attribute of the PARAM with name="standardID". For free or custom services, this PARAM is not included.

- 19 -

DataLink

- 4.1 table :
- add the « content-type » RESOURCE PARAM with value « media type » as an optional feature in the table.



Change 11

www.ivoa.net/documents/DataLink/20150617/REC-DataLink-1.0-20150617.pdf

19 sur 32

VOTable document is created.

4.1 Service Resources

In a data discovery response, one RESOURCE element (usually the first) will have an attribute type="results" and tabular data; this resource contains the query result.

To describe an associated service, the VOTable would also contain one or more resources with attribute type="meta" and utype="ad hoc:service". A resource of this type have no tabular data, but may include a rich set of metadata. The utype attribute makes it easy for clients to find the RESOURCE elements that describe services. A service resource contains PARAM elements to describe the service and a GROUP element with additional PARAM elements to describe the input parameters. The standard PARAM elements for a *service* resource are described in the table below.

name	value	required
accessURL	URL to invoke the capability	yes
standardID	URI for the capability	no
resourceIdentifier	IVOA registry identifier	no

Table 3: Service Resource Parameters

For services that implement an IVOA standard, the standardID is specified as the value attribute of the PARAM with name="standardID". For free or custom services, this PARAM is not included.

- 19 -

DataLink

- 4.1 :
- add a DESCRIPTION element (SHOULD status) in the service descriptor.



Change 12

4.6 Example: Self-Describing Service

A service may include a service descriptor that describes itself with its normal output. This usage is comparable to prototype work on S3 (see [12]) and when combined with calling a service with no input parameters (e.g as allowed in 2.1.1) will make it easy for clients to obtain a description of both standard and custom features.

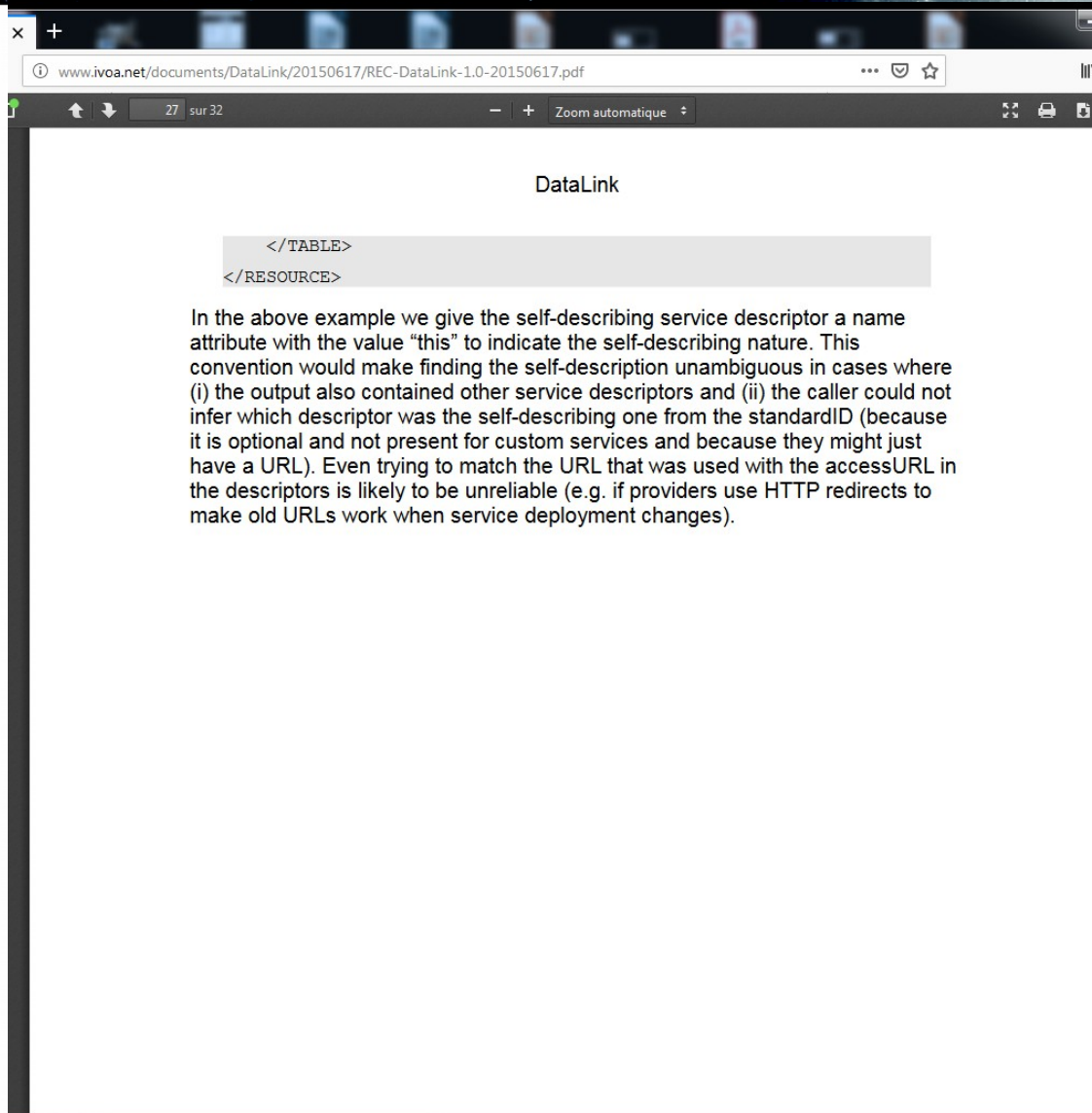
The output of a {links} capability with no input ID would include the self-describing service descriptor and an empty results table:

```
<RESOURCE type="meta" utype="ad hoc:service" name="this">
  <PARAM name="standardID" datatype="char" arraysize="*"
    value="ivo://ivoa.net/std/DataLink#links-1.0" />
  <PARAM name="accessURL" datatype="char" arraysize="*"
    value="http://example.com/mylinks" />
  <GROUP name="inputParams">
    <PARAM name="ID" datatype="char" arraysize="*"
      value="" ref="primaryID"/>
  </GROUP>
</RESOURCE>
<RESOURCE type="results">
  <TABLE>
    <FIELD name="ID" datatype="char" arraysize="*"
      ucd="meta.id;meta.main" />
    <FIELD name="access_url" datatype="char" arraysize="*"
      ucd="meta.ref.url" />
    <FIELD name="service_def" datatype="char" arraysize="*"
      ucd="meta.ref" />
    <FIELD name="error_message" datatype="char" arraysize="*"
      ucd="meta.code.error" x/>
    <FIELD name="semantics" datatype="char" arraysize="*"
      ucd="meta.code" />
    <FIELD name="description" datatype="char" arraysize="*" />
  </TABLE>
</RESOURCE>
```

- 4.6 :
- extend the description of self-describing services to responses attached to a given ID



Change 13



The screenshot shows a web browser window with the address bar containing the URL `www.ivoa.net/documents/DataLink/20150617/REC-DataLink-1.0-20150617.pdf`. The page title is "DataLink". The main content area displays the following XML code:

```
</TABLE>
</RESOURCE>
```

Below the code, there is a paragraph of text explaining the convention:

In the above example we give the self-describing service descriptor a name attribute with the value "this" to indicate the self-describing nature. This convention would make finding the self-description unambiguous in cases where (i) the output also contained other service descriptors and (ii) the caller could not infer which descriptor was the self-describing one from the standardID (because it is optional and not present for custom services and because they might just have a URL). Even trying to match the URL that was used with the accessURL in the descriptors is likely to be unreliable (e.g. if providers use HTTP redirects to make old URLs work when service deployment changes).

- section 4
 - add a new subsection for description of Restful services.
 - It cannot be limited to input PARAMS.
 - Add a templating mechanism ?



Change 14

www.ivoa.net/documents/DataLink/20150617/REC-DataLink-1.0-20150617.pdf

28 sur 32

Zoom automatique

- 27 -

DataLink

5 Changes

This is the initial version of this document.

5.1 *PR-DataLink-1.0-20150413*

Restricted the {links} resource path so that it must be a sibling of the VOSI resources in order to allow discovery of VOSI resources from a {links} URL.

Changed ID parameter to allow caller to invoke service with no ID values and get an empty result table; this is actually easier to implement than a special error case. Added reference to previous work on S3 and an example section where an empty links response has a self-describing service descriptor and an empty result.

Fixed URL to DALI document in the references section.

Fixed namespace prefix in example capabilities document to use recommended value.

5.2 *PR-DataLink-1.0-20140930*

Re-organised introduction to introduce the links capability and distinguish it from the service descriptor more clearly. Explicitly noted that service descriptors do not

- New section :
 - explain the new LINK recognition mechanism as in the IVOA note.



Question

SHOULD we start writing this now ?

