

## Abstract

Datalink is a new Virtual Observatory protocol designed to facilitate the publication of complex datasets. It provides a standard way to declare relationships between

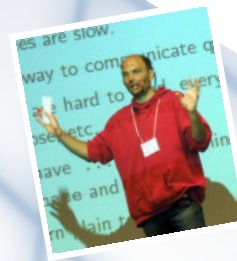
- parts of data sets (e.g., separate data and error files),
- different representations (e.g., previews, derived plots, or data provided in multiple resolutions),
- ancillary calibration data,
- files in provenance trees,
- data services for server-side processing of such data sets (e.g., cutout and rebinning services).

Datalink services can run standalone, where access links are passed around directly, or embedded in VO DAL protocols like ObsTAP or SSAP, where a special annotation of the VOTable service responses allows Datalink-enabled clients advanced access patterns while legacy clients continue to work unchanged.

Datalink is already in use at sites like the CADC or the GAVO data center, with first clients (e.g., SPLAT, TAPHandle) already providing interfaces.

# Introducing Datalink

Markus Demleitner  
Universität Heidelberg, Astronomisches Rechen-Institut  
Mönchhofstraße 12-14, Germany



This poster tries to explain the workings of datalink using the cartoon in the center of the poster. Each item in the cartoon marked with a circled number is explained here, starting with:

1 – a **datalink table**. At the heart of the protocol is a simple table schema, usually transmitted in VOTables. The table has columns

- id – a globally unique dataset identifier, included so descriptions of multiple datasets can reside within a single table.
- access\_url – where to find the linked data
- semantics – a machine-readable description of the data
- content\_type, content\_size – a media type and the size of the data
- a human-readable description (not shown here)
- some administrative columns (not shown here)

## A Scanned Plate and what came of it

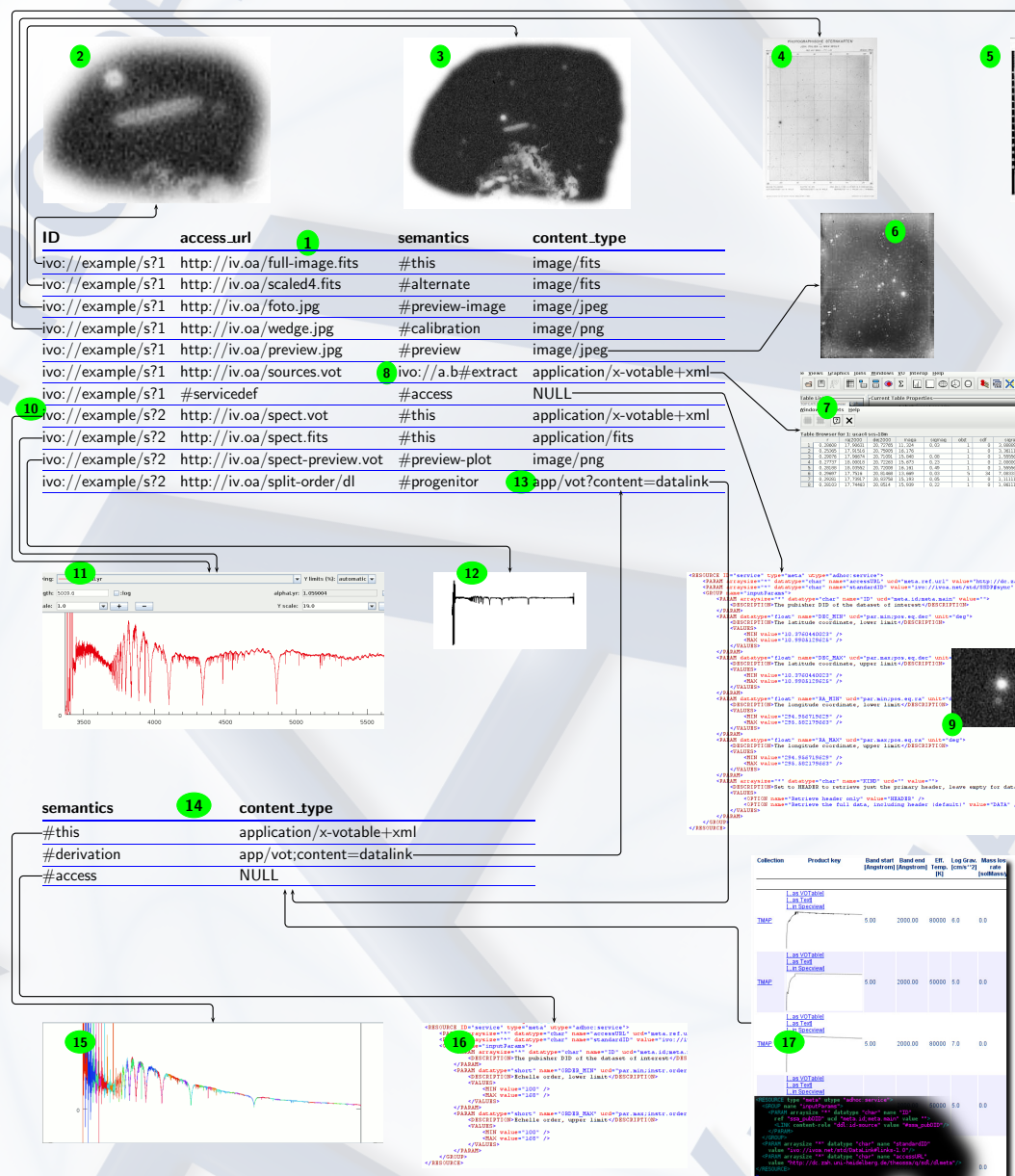
The first seven rows in this example correspond to a scanned plate. A placeholder for the **original dataset** is 2, with semantics #this, i.e., the “main” dataset. A **rebinned version** 3 (the placeholder shows a larger area) is declared as an #alternative representation of the main dataset (inclusion of alternative into the main datalink vocabulary is pending).

The plate is part of an early survey which has been published in book form. A JPEG photo of the book page corresponding to the plate is declared as #preview-image here.

Datalink is ideal for keeping files from a dataset’s provenance chain. Here, we include a **PNG grey wedge** 5 from the scan with #calibration semantics.

Finally, a **thumbnail** 6 of the image is included as a #preview

## A Cartoon of a Datalink Table and the Things it Links



7 illustrates a **table of extracted sources** from the image. No corresponding semantics is given in the core datalink vocabulary yet, but data providers can introduce new terms as URIs. Datalink semantics are defined in RDF, so data providers using this feature can introduce relationships to existing terms (as in 8 ivo://a.b#extract is a derivation) immediately or later.

Semantics URIs are supposed to be interpreted relative to

<http://www.ivoa.net/rdf/datalink/>; the leading hashes simply are fragment identifiers into that vocabulary. The next row defines a **cutout service** 9. Datalink allows a straightforward declaration of the parameters for server-side data manipulation services within the VOTables that return datalink metadata. If you decipher the XML, you will see that this is sufficient not only to operate the service but also produce attractive UIs by declaring units, UCDs, and ranges of the pertinent parameters.

## An Echelle Spectrum

Starting at 10, a few datalinks for a **different dataset** start, as you can see from them having a different ID. The semantics #this should already be familiar; it corresponds to a **spectrum** 11 here, except that we provide this spectrum both as a VOTable and a FITS file, and hence there are two rows for #this.

The **preview** 12 in this case is a plot, which is reflected in the different semantics. A client consulting the RDF of the core vocabulary would figure out that #preview-plot actually is a #preview.

The last datalink 13 shows **recursive datalink**: its file has the media type

application/x-votable+xml;content=datalink

that designates datalink documents (and can be used in protocols like ObsTAP, too). In this case, the datalink is for a #progenitor in the provenance chain, which in this case is a file with unmerged Echelle orders.

We only give semantics and content\_type for this **second datalink result table** 14. In it, you will again find the datalink link to the **unmerged orders** 15. In addition, there is the backlink to the #derivation datalink result for ivo://example/s?2 we just looked at, and again a **16 processing service**. Inspecting its metadata, you will see that it allows the retrieval of selected orders only.

## DAL Responses

Finally, 17 to symbolises an **SSAP response**; within responses of VO data access layer (DAL) protocols, operators can include a brief piece of XML denoting a column containing values of Datalink’s ID parameter as shown in the black inset. Clients can then retrieve datalink documents from the service indicated instead of using the access URL given in the DAL response directly. The latter, most likely directly linking to the dataset, will ensure compatibility with older clients, while modern, Datalink-enabled clients can unlock the wealth of information conveyed through Datalink.

Server-side support for Datalink is already included in the DaCHS VO server package. In it from a single code base both datalink metadata (the dlmeta renderer) and processed data products (the dlget renderer) are generated. The behaviour of the services is defined through a combination of a descriptor generator (obtain dataset metadata), meta makers (generate data links or processing parameters), data functions (manipulate data during server-side data processing), and formatters (bring data into the format promised to the client). Extensive documentation is available in the DaCHS reference manual.

Client support for datalink is not yet great; for instance, the SPLAT spectral analysis client uses datalink features to allow server-side processing of spectra.

Until the major clients have satisfying Datalink UIs, an interim solution is to let web browsers display Datalink documents. The beginnings of a corresponding stylesheet are already part of DaCHS, other developers are welcome to re-use and improve the material. Note, however, that to make this work with stock browsers, Datalink documents need to be delivered with a text/xml media type in mild violation of the standard. Using the request’s Accept: HTTP header, this can safely be done.

Information on and the texts of the IVOA standards mentioned is found at <http://ivoa.net/documents>, with the Datalink specification residing at <http://ivoa.net/documents/DataLink>.

DaCHS (discussed in more detail in 2014A&C.....7...27D) can be obtained from <http://soft.g-vo.org>. Documentation on defining datalink services is available from <http://docs.g-vo.org/DaCHS/ref.html#datalink-cores>.