Multidimensional (Cube) Data in the VAO: June – Sept 2013

Summary

The issue of science data access to multidimensional (cube) data was extensively discussed in both the spring 2013 IVOA interoperability workshop (Heidelberg, May 2013), as well as in the just concluded US VAO team meeting (Annapolis, June 2013). The Heidelberg workshop featured a special session on multidimensional data including representation from key astronomy projects outside the normal VO community.

The agreement reached with our IVOA partners in Heidelberg was to prototype cube access through the summer in two tracks, one TAP-based and the other SIAV2-based, with both approaches sharing similar components. Functional prototypes with real data are to be completed by the fall Interop to inform discussion and aid further planning. This is consistent with our existing VAO development plans, which call for delivery of a prototype reference implementation of SIAV2 by October 2013.

The activities, functionality, and deliverables described herein cover only the period through the end of September 2013, focused on prototyping and related standards development through the fall IVOA Interop. Based upon our current planning, by 09/20/13 (a week prior to the fall 2013 IVOA interoperability workshop), the VAO will deliver, as a minimum:

- A demonstration (prototype reference implementation) SIAV2 data access service supporting basic discovery and retrieval of real multidimensional datasets, including at least radio and O/IR data. If possible a cutout capability will also be included to demonstrate basic data subsetting for large multidimensional datasets.
- Updated drafts of the SIAV2 and ImageDM specifications, sufficient to support ongoing prototyping, community review, and IVOA discussions.

Time permitting, additional capabilities may be prototyped and demonstrated, including some subset of the following:

- Client-directed, precision interactive access to large cube datasets including capabilities for filtering, subsetting, and 2-D projections of various kinds (a basic subset of the capabilities planned for the full interface).
- Support for output in data formats other than FITS; FITS image being the default, with support for FITS world coordinate system (WCS) specification of the image coordinate systems for both input and output.
- Optional support for STC-S to specify the search region and/or multidimensional filter to be applied for cutout generation.
- Prototype support for data linking (the VO *datalink* standard currently under development) to point to associated data products or other capabilities (such as data access or data reprocessing services) available for a particular dataset or observation.
- Provision of actual (although still prototype) SIAV2 data services at several sites (e.g., NRAO, SAO, IPAC, possibly CDS) to serve up real archive multidimensional image data collections, including 2-D and n-D images and possibly event data.

VAO is also participating in a TAP-based prototyping effort being led by CADC. Automated virtual data generation of an n-D cutout specified via STC-S is the basis of this approach. Aside from the use of TAP for generic data discovery, the essential capabilities required are included in our SIAV2 prototype, in particular the cutout capability and support for STC-S in combination with data linking (a sufficient CADC-like capability could be demonstrated for image data using only capabilities provided by a SIAV2 service, whether or not TAP is available for a particular archive).

Potential end users, such as ALMA, JVLA and JWST, will evaluate the capabilities provided and their feedback will inform the IVOA discussions and a subsequent operational release and related standards. We will also release the VAO white paper on data cubes as a document to help inform the community and to solicit feedback.

Demonstration / Reference Implementation Data Service

From the current SIAV2 specification, a minimal demonstration service must:

- Implement the *queryData* method providing synchronous return of the query response encoded as a VOTable document.
- Support at least the following mandatory SIAV2 query parameters: POS, SIZE, BAND, TIME, and FORMAT (possibly also POL for polarization data).
- Support synchronous retrieval of data via the access reference URL of image datasets referenced in the query response.

Thus a sample client call to discover cube data thus might be:

$*baseURL*/sync?REQUEST=queryData&POS=180,0&SIZE=0.2&BAND=10E-3/12E-3

The query response is a VOTable containing image metadata for matching datasets, including access reference URLs to retrieve the referenced datasets.

The range-list specification of BAND and TIME parameters permits per-axis filtering for data discovery as well as simple automated cutout generation (for services that provide a cutout capability). More powerful capabilities for client-directed slicing and dicing of individual data cubes are provided by the optional SIAV2 *accessData* service operation, if implemented by the service (advanced *accessData* functionality is out of scope for the near term schedule).

Retrieved data products must be available in FITS format. Additional output image formats may be available, such as CASA image table format, HDF5, NDF, JPEG, etc., depending on time and resources.

The service will initially provide access to at least the following test datasets: VLA and ALMA science verification data, mock JWST data, and NED data. Additional M51 HI data[1] as well as data from JCMT and Osiris (mock data) may also be available.

The demonstration/reference SIAV2 service will be hosted by NRAO and will be implemented using the DALServer framework.

[1] http://hdl.handle.net/10904/10230

Standards

The primary standards required for a SIAV2 reference implementation are SIAV2 and the ImageDM, which defines the data model for multidimensional image data. Working drafts of both exist, however updates are required to support development of the prototype service.

Related work by the IVOA DAL working group on the DataLink-1.0 and STC-S-1.0 specifications is also planned, with drafts targeted for June and July and preliminary recommendations in August and October respectively. The DALI (DAL Interface) specification is currently in PR and will be used to help define the standard elements of a SIAV2 service.

DALServer Framework

It is useful to review aspects of the DALServer service framework[2] here in order to understand the work required to prototype SIAV2 within the framework.

DALServer provides generic implementations of the IVOA data access protocols within a common framework, allowing shared components to be reused to implement each protocol. Elements such as input query processing, parameter parsing and management, data model management, query response processing, query output formatting in VOTable or other formats, management of asynchronous processing (UWS), service capability querying (VOSI), etc., are largely the same for all DAL services hence may be abstracted to a common framework.

In addition, in the case of SIAV2, DALServer already has an implementation of the SSA protocol for access to spectral data. While the actual data access for spectrum and image data differs, both SSA and SIAV2 are second-generation DAL protocols, and the VO protocol is very similar for the two classes of data. Much of the SSA implementation (which is mostly generic in any case) can be reused for SIAV2.

The DALServer framework provides a generic implementation of the IVOA data access protocol for each class of data (table, image, spectrum, etc.). To produce an actual data service, one must install the framework locally and then configure a ***service instance***. The new service instance is given a unique servlet name and type so that it can be called externally, and various service parameters are configured. In particular the service instance is pointed at one or more database tables containing metadata describing the data collection to be served.

Given a configured service instance it is possible for the framework to automatically produce a VO service instance capable of serving up static archive data products, e.g., whole image files. More sophisticated services capable of computing cutouts or other virtual data products additionally require some ***back-end processing***. Such back-end processing is similar to conventional astronomical data processing, e.g., for a pipeline, and usually has little or no direct connection to VO; hence generic data processing components can often be used. In more complex use cases the back-end processing will be specific to the data collection being accessed, implemented using custom software specific to the data.

Given a generic implementation of the service protocol, a configured service instance, and possibly some back-end processing to compute virtual data products, one has all the elements required to deploy a VO data service that can be queried programmatically, e.g., as a Web

[2] http://dev.usvao.org/vao/wiki/Products/DALServer

service (HTTP-based). The remaining capability required is a ***Web query interface*** to be able to query the new service instance interactively, via a Web browser.

Since the service framework manages the service instance and has full knowledge of its service class, capabilities and data, a Web query interface can be auto-generated for each new service instance. This includes a conventional query form specific to the class of data (possibly augmented with custom parameters specific to the individual data collection), with the query response displayed in the browser as a table, e.g., using a dynamic table rendering technology such as *voview*. *The Web query interface is especially important for a demonstration service, as this is the main component of the service framework of interest to a prospective user.* If nothing else it helps them to "see" the data collection being served up, and understand what the service provides. It is also essential for a production service deployment.

Ultimately what one ends up with is a DALServer instance deployed at a site, providing programmatic access to one or more data collections via the standard VO protocols. In addition the Web-based user interface may be used to display the data collections (services) available at the site. Each data collection is then queryable via a forms-based Web query, and ultimately individual data products may be previewed, downloaded or otherwise flagged for further processing. By configuring a new data service the user not only gets a reference-grade VO service implementation, they get a conventional browser-based Web query interface for their data collection. In the case of a smaller site that only has one or more image data collections (or catalogs) to publish to the VO, something like TAP would be overkill, while configuring a new SIA service instance is straightforward and can provide more advanced, scalable capabilities for actual image data access (not to sound like a salesman here, but this is what we want to come across with a demonstration service).

In terms of implementation, the generic service framework, machine readable data models, back-end processing if any, and Web query interface are all ***separable components*** (hence implementable separately, possibly in parallel) within the overall DALServer framework, connected together via interfaces that define the configured service instances, the input parameters for a service class or instance, the standard metadata for a given service class, the back-end processing interface, and so on.

Schedule

The task list and schedule for VAO cube data access standards development and prototyping through early October is given in the table below. Due to space limitations it is not possible to fully describe each task directly in the table, however additional task-specific notes follow below the summary table.

Task	*Description*	*Start*	*End*	*Who*
Update ImageDM, SIAV2 drafts to support prototyping			30 Jun	
1	Updated ImageDM WD			
1.1	Inventory remaining differences ObsCore, SDM			FB,MC
1.2	Review ImageDM architecture			MC
1.3	List of Utypes for ImageDM			
1.4	AccessData model			
1.5	Update June version of WD sufficient to support prototyping			all

2	Updated SIAV2 WD			
2.1	Tweak SIAV2 interface summary			DT,FB
2.2	Reworked SIAV2 WD sufficient to support prototyping			DT,FB
Provision of test datasets			30 Jun	
3	Deliver test data to NRAO			MG
Initial SIAV2 reference implementation at NRAO		01Jul	15 Aug	
4.1	Assemble test datasets (both 2D, ND)			
4.2	Update framework design			DT,MC
4.3	Core framework updates			DT
4.4	New auto-generated Web query interface			MC
4.5	SIAV2 generic reference impl. within updated framework			DT
4.6	Configure service instances for test data collections			MC,DT
Documentation and testing of reference implementation		15 Aug	31 Aug	
5.1	Document how to use demo/test interface			
5.2	Web page description of project (VAO project pages)			
5.3	Basic testing of reference service query interfaces			BB
Add actual data service instances at several sites		15 Aug	15 Sep	
6.1	NRAO data collection(s)			DT+
6.2	SAO data collection(s)			AR+
6.3	Other sites TBD			
Update Reference Implementation				
7	Update reference implementation as 5, 6 proceed	15 Aug	15 Sep	DT,MC
Other Tasks				
8	Release of VAO Whitepaper on data cubes v1.0		15 Sep	
9	Delivery of demonstration services for Interop		20 Sep	
10	Update ImageDM, SIAV2 WD for Interop		23 Sep	
Fall IVOA Interoperability workshop		26 Sep	28 Sep	

Additional notes on the above tasks follow below, indexed by task number.

1.3	**List of Utypes**. Most DM Utypes are common with other Observation-derived data models, e.g., ObsCore and SDM. We need to define a standard place to put metadata specific to the dataset class, Image in this case. Mapping needs to be reviewed and a final decision made on what approach to take for the prototype.
1.4	**AccessData Model**. This needs to be present in the architecture and drafts (it already is) and there may be time to prototype it, however this represents advanced functionality and much of this will need to be deferred until the fall.
1.5	**ImageDM WD**. To support the prototype this requires a data model architecture and a decomposition of the data model into a list of ImageDM Utypes (this is what is used to drive the DALServer framework, and to populate a query response). Rigorous specification of each individual data model element is primarily to support users and is not required for to support prototyping.
2.2	**SIAV2 WD**. For prototyping what is most required is an updated interface summary. An updated SIAV2 draft (newer than late 2009) is required to reflect the current state of the project. In particular this must reference recent relevant standards such as DALI, VOSI, Datalink and so forth (SSA can serve as a stand-in for PQL). It is not necessary to fully flesh out the working draft at this point.
4.2	**Update framework design**. This includes the key interfaces between the components, e.g., list of configured service instances, metadata defined for a service instance, back-end interface, and so forth. These interfaces are required to allow the components to be developed independently.

5.1	**Documentation**. Since this is a prototype and not an actual product, only minimal documentation is required. This can consist of Web-viewable documentation on usage, included directly in the DALServer implementation, possibly augmented by a Web page describing the prototype implementation and what is provided.
5.2	**Project Description**. A VAO project page describing the cube data access prototype as a project.
6.1	This would be a separate deployment of DALServer to an NRAO VO server, interfaced directly to the NRAO archive, e.g., to expose a full image data collection (possibly 2-D).
6.2	This could include pass-through of event data, filtered to match the SIA region of interest.
8	**Delivery of test services**. This includes the SIAV2 reference implementation with integrated test data collections, any additional SIAV2 data service instances serving real data collections, and (time permitting) a service interface compatible with the CADC approach, but possibly using SIAV2 instead of ObsTAP for data discovery.

Note that only tasks 1-5, 9 are required to be completed during the schedule period, as noted earlier. The remaining tasks will be addressed on a time-available basis, but are highly desirable, time permitting. In particular it will be helpful to move beyond just the reference implementation of SIAV2, and have deployments for actual production data collections at several sites to help demonstrate the SIAV2 capabilities on real data collections. In the case of schedule slippage the later tasks can be deferred until after the Interop.