

# Virtual Observatory Standards Overview

This document provides an overview of some of the standards LSST should consider adopting along with any issues that may cause us difficulty.

## Simple Cone Search

<http://www.ivoa.net/documents/REC/DAL/ConeSearch-20080222.html>

SCS is a scheme for querying a catalog given a position on the sky and a circular aperture. Just requires RA/Dec and radius all in degrees. Service must return results in VOTable format. Three mandatory fields in the returned table: ID string, RA/Dec as doubles in ICRS.

Resource metadata requires an epoch of the observations as a free-form string. This resource metadata seems to be a survey wide definition rather than relating to the individual results of a query.

It is entirely reasonable for SCS to be a thin frontend to a TAP service.

## Simple Image Access

Query a service for 2D image data. Simple 2D image access has been around for a long time. Version 2.0 adds support for query and retrieval of spectral data cubes and the ability to do simple remote analysis of image subsets; the latter being of particular importance for large spectral cubes. The V2 draft document has not been updated in a year and is still open to debate. A [VAO poster from 2014 AAS](#) is attached. Image retrieval with cutouts is already required for LSST.

## Table Access Protocol

<http://www.ivoa.net/documents/TAP/>

Used to query tables. This can mean observation information (ObsTAP) or source catalogs or time-series catalogs. Must be able to handle ADQL and must be able to return VOTable. Accepting other query languages is allowed, as is returning results in other table formats. TAP supports synchronous or asynchronous queries. Tracking progress of a query is done using the Universal Worker Service (UWS) scheme.

TAP looks fairly straightforward. The main issue being ADQL and support of asynchronous queries.

## Astronomical Data query Language

<http://www.ivoa.net/documents/latest/ADQL.html>

This standard has been around since 2008 and is based around SQL92. Some trigonometrical functions have been added but also geometrical functions to allow region queries via STC-S strings (e.g. area, polygon, distance, circle, region). A BNF grammar is provided.

## DataLink

<http://www.ivoa.net/documents/DataLink/20150414/PR-DataLink-1.0-20150413.html>

This is used to link a resource found in one service to resources provided by other services. For example, given a stacked image DataLink could be used to return all the individual visits that went into that image. A source in a catalog returned by Simple Cone Search could be linked to the source footprint. An image could be linked to a variance. A preview image (such as a JPEG file) could be linked to the real data file.

This seems like a very useful functionality that could be integrated nicely into the SUI tools (SUIT?).

## Common Archive Observation Model (CAOM)

<http://www.cadc-ccda.hia-ihp.nrc-cnrc.gc.ca/caom2/>

This is a model for representing observations and compound observations (ie deep stacks) and the artefacts (files) relating to those observations. It is not an IVOA standard but is a model that builds on IVOA standards to provide a single schema simplifying creation of IVOA services.

CAOM has nothing to say about catalog data other than how rows in a table may relate to a catalog file (artefact). A TAP to ADQL converter for Qserv would still need to be written, as would a Simple Cone Search.

CAOM and the related software make VO interfaces like ObsCore, SIA, image cutout service, DataLink and "footprint services" (where footprint here means the instrument footprint [area covered by an observation] not the footprint of a source) almost trivial. CADR have a layered design such that SIA is implemented by using TAP on top of ObsCore. ObsTAP (VO discovery of observations) would also be trivial.

We could consider designing our observation/image archive tables to match the CAOM design or provide a completely distinct set of database tables specifically for CAOM. CAOM-compatible views are also an option.

The database access layer in the CADC stack is pluggable. CADC have software implementing TAP, SIA and other services all written in Java on top of CAOM using a Postgres database.

## MAST PanSTARRS

MAST use CAOM to represent all their archive observation data. They are very happy with it. They use the python CAOM library from CADC to generate data in the correct format but do not use any of the CADC Java layer. They have designed their database schemas to be CAOM-compliant. They are writing their own TAP library to access the catalogs because they use .Net and MS SQL Server. MAST also support a Simple Cone Search interface which can return multiple formats (VOTable, TSV, CSV etc). They currently use CASjobs for handling long-lived queries (at least until the TAP system is deployed).