Current status of VO compliant data service in Japanese Virtual Observatory

Y. Shirasaki, Y. Komiya, M. Ohishi , Y. Mizumoto (NAOJ),
Y. Ishihara, J. Tsutsumi, T. Hiyama (Fujitsu),
H. Nakamoto, and M. Sakamoto (SEC)

jvo 検索

http://jvo.nao.ac.jp/portal

Abstract

In these years, standards to build a Virtual Observatory (VO) data service have been established with the efforts in the International Virtual Observatory Alliance (IVOA). We applied these newly established standards (SSAP, TAP) to our VO service toolkit which was developed to implement earlier VO standards SIAP and (deprecated) SkyNode. The toolkit can be easily installed and provides a GUI interface to construct and manage the VO service. In this poster, we shows the architecture of our toolkit and how it can be used to build a VO service.

Architecture of VO Service Toolkit

The architecture of the toolkit is show in Figure 1. Five kinds of access interface are placed behind the http interface, which correspond IVOA standards Table Access Protocol (TAP), Simple Spectrum Access Protocol (SSAP), and Simple Image Access Protocol (SIAP), Simple Cone Search Protocol (SCSP) and data request interface, respectively. ADQL is constructed from the request parameters and is passed to the VOQuery Executor. The VOQuery Exeuctor translates the ADQL to a DBMS's native SQL, and execute the query. Search result (ResultSet object) is converted to a VOTable object by consulting column metadata to a metadata database.

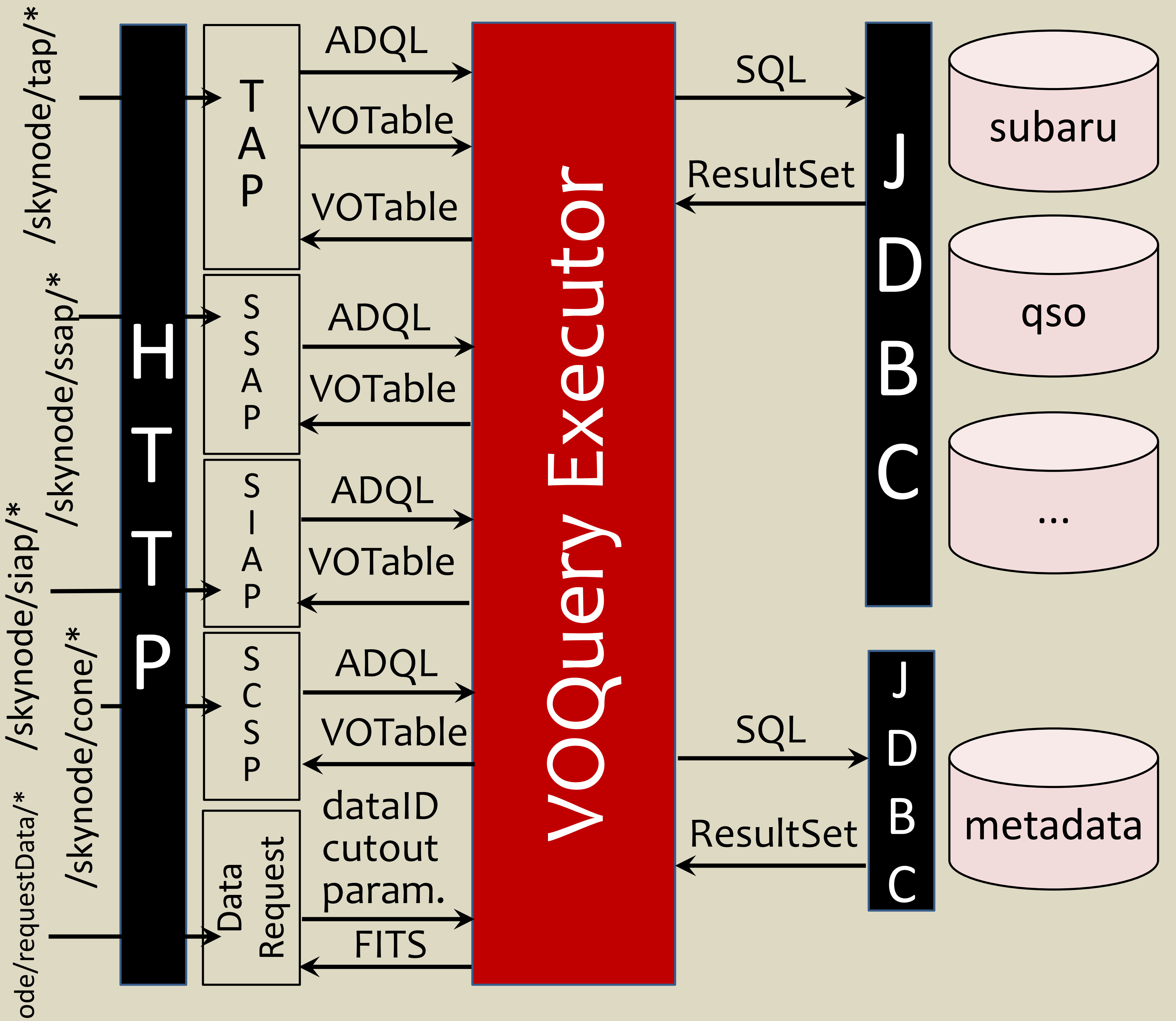


Figure 1. Architecture of the VO service toolkit.

VO Services hosted at JVO

- Subaru Suprime-Cam archive (TAP/SIA)
- Subaru MOIRCS archive (TAP/SIA)
- Subaru HDS archive (TAP/SSA)
- Subaru Deep Survey (SDF, SXDS) (TAP/SIA)
- Stellar Abundances for Galactic Archeology Database (TAP)
- etc…
- AKARI DR1 (TAP)
- IRSF LMC survey (TAP/SIA)
- Kiso UV-X Galaxies Catalog (TAP)

Software

Java DK

Software development kit

Apache Tomcat

web server
Servlet container

Apache Struts

web application framework

Postgresql

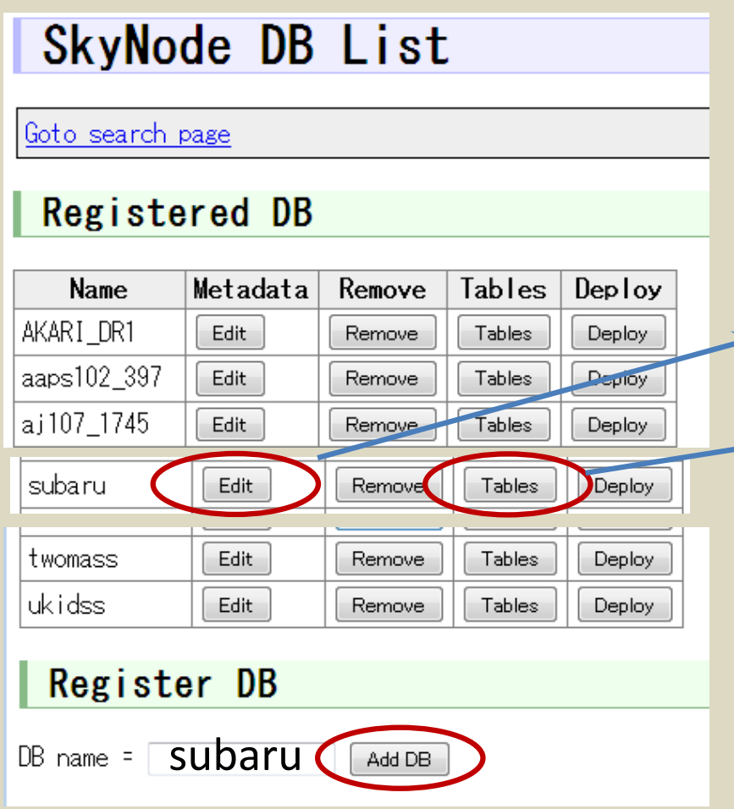
object-relational DBMS

Apache Tomcat is used for a web container. Struts framework is used for invoking an appropriate Java method on a http request and returning a result file (VOTable, html, fits, graphics …) . Although Postgresql is taken as a target implementation for an backend DBMS, any DBMS which supports JDBC can be easily used as an backend DBMS.

Search with a celestial region condition is implemented by using HTM index and/or PGSphere extension library for Postgresql.

How to start VO service

1. create your db
   $ createdb subaru
2. create a table and insert data
   $ psql subaru
   postgres=# create table image (…)
   postgres=# copy image from '…' delimiter …
3. register the db on the web GUI (figure 2)
4. register db metadata (figure 3)
5. select table and edit the metadata (figure4,5)
6. edit column metadata (figure6)



SkyNode DB List

Goto search page

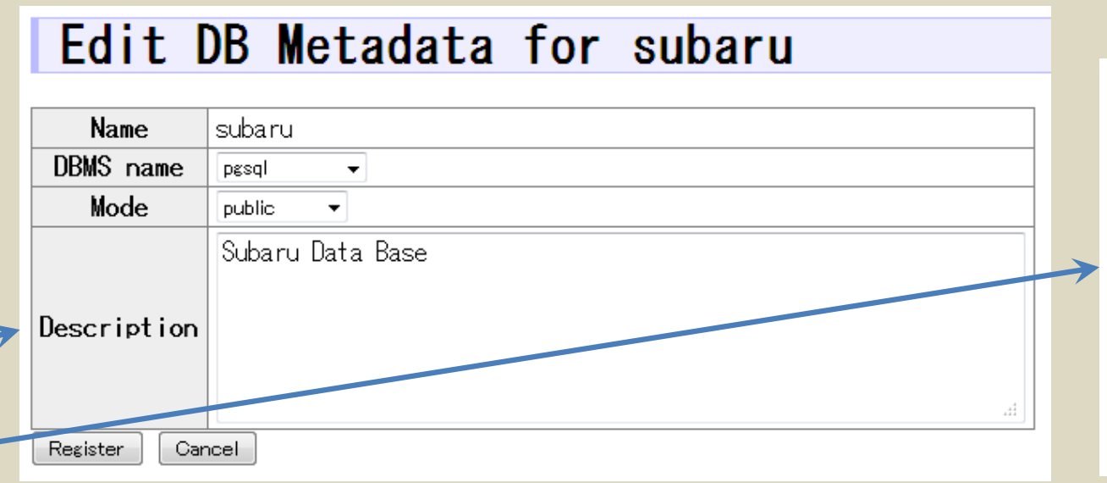
Registered DB

Name	Metadata	Remove	Tables	Deploy
AKARI_DR1	Edit	Remove	Tables	Deploy
aaps102_397	Edit	Remove	Tables	Deploy
aj107_1745	Edit	Remove	Tables	Deploy
subaru	Edit	Remove	Tables	Deploy
twomass	Edit	Remove	Tables	Deploy
ukidss	Edit	Remove	Tables	Deploy

Register DB

DB name = subaru Add DB

Figure 2. GUI for registering a database to metadata DB.

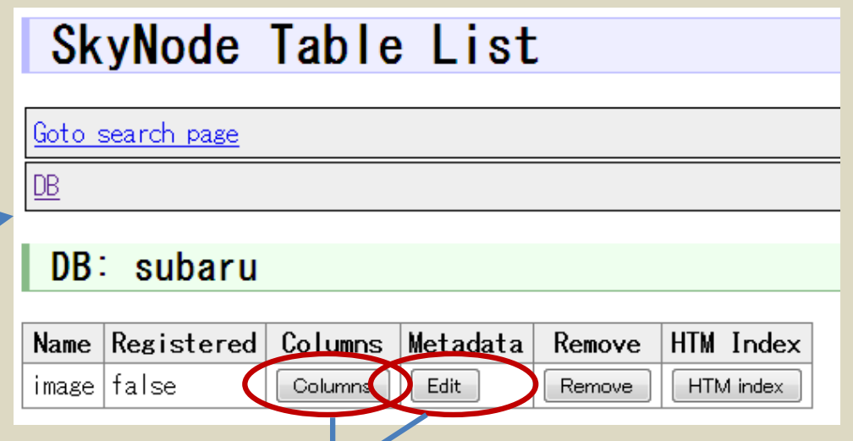


Edit DB Metadata for subaru

Name	subaru
DBMS name	pgsql
Mode	public
Description	Subaru Data Base

Register Cancel

Figure 3. edit database access parameters and its description



SkyNode Table List

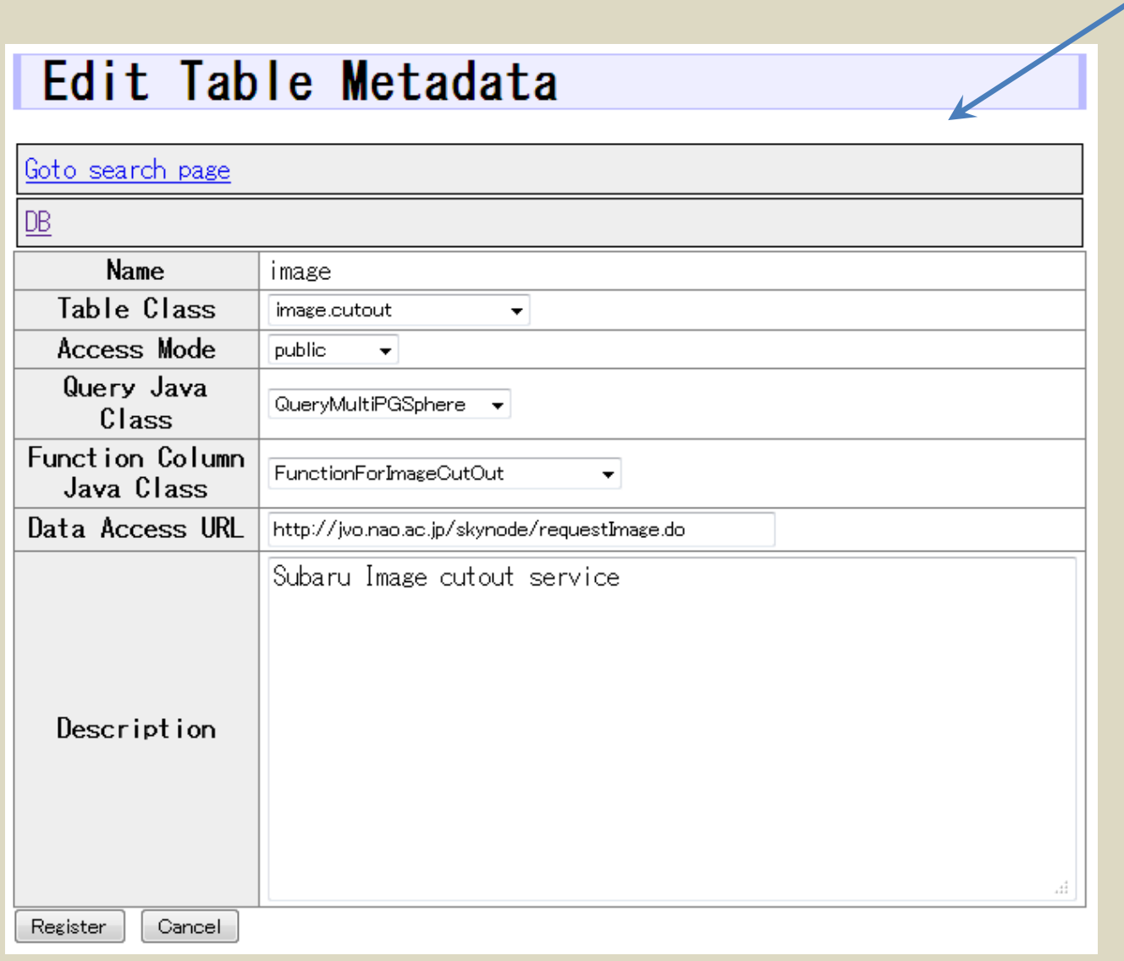
Goto search page

DB

DB: subaru

Name	Registered	Columns	Metadata	Remove	HTM Index
image	false	Columns	Edit	Remove	HTM index

Figure 4. edit database access parameters and its description



Edit Table Metadata

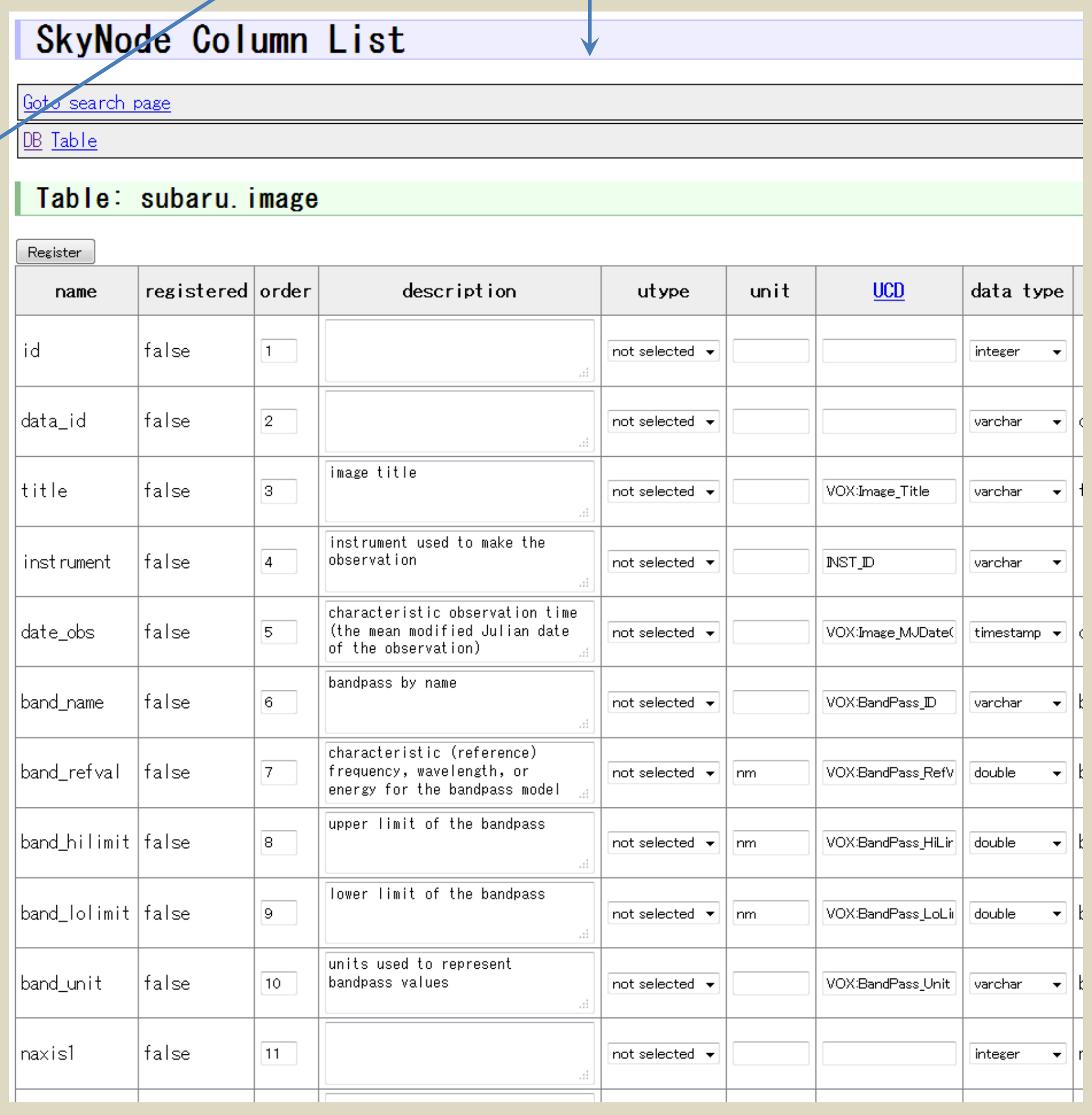
Goto search page

DB

Name	image
Table Class	image.cutout
Access Mode	public
Query Java Class	QueryMultiPGSphere
Function Column Java Class	FunctionForImageCutOut
Data Access URL	http://jvo.nao.ac.jp/skynode/requestImage.do
Description	Subaru Image cutout service

Register Cancel

Figure 5. edit table metadata



SkyNode Column List

Goto search page

DB Table

Table: subaru. image

Register

name	registered	order	description	utype	unit	UCD	data type
id	false	1		not selected			integer
data_id	false	2		not selected			varchar
title	false	3	image title	not selected		VOX:Image_Title	varchar
instrument	false	4	instrument used to make the observation	not selected		INST_ID	varchar
date_obs	false	5	characteristic observation time (the mean modified Julian date of the observation)	not selected		VOX:Image_MJDate(	timestamp
band_name	false	6	bandpass by name	not selected		VOX:BandPass_ID	varchar
band_refval	false	7	characteristic (reference) frequency, wavelength, or energy for the bandpass model	not selected	nm	VOX:BandPass_RefV	double
band_hilimit	false	8	upper limit of the bandpass	not selected	nm	VOX:BandPass_HiLir	double
band_lolimit	false	9	lower limit of the bandpass	not selected	nm	VOX:BandPass_LoLi	double
band_unit	false	10	units used to represent bandpass values	not selected		VOX:BandPass_Unit	varchar
naxis1	false	11		not selected			integer

Figure 6. edit column metadata