P1.1.23 Structured Query Language for Virtual Observatory

Yuji Shirasaki

National Astronomical Observatory of Japan,

and Masahiro Tanaka (*NAOJ*),

Satoshi Honda (*NAOJ*), Yoshihiko Mizumoto (*NAOJ*),

Masatoshi Ohishi (*NAOJ*), Naoki Yasuda (*ICRR*),

Yoshifumi Masunaga (*Ochanomizu Univ.*),

Masafumi Oe (*NAOJ*)

1. Abstract

We have designed a multi-purpose Astronomical Query Language for VO (called as JVOQL) on the basis of SQL99 specification.

This poster describes:

- Specification of the JVOQL
  - Construct and Data Type
- JVO SkyNode implementation
  - Functionality test of JVOQL
  - First JVO's interoperability test with other VO.

2. Requirement to the JVOQL

1. Any kind of astronomical data can be queried : Catalog, Image, Spectrum, 3D-Cube, Photon List, Light Curve, ...
2. Support for Astronomy specific query condition
3. Simple enough for easy implementation.
4. High Extendability for describing an efficient query.

3. Design

Req. 1 and 2 → SQL based query language with Astronomical Extension

Most of the astronomical data are stored in the relational database, so it is natural to use the SQL.

Mapping a query parameter a column of the table, any data request can be described in SQL.

Data type describing the region in the sky and comparison operator for the region data type should be defined as an Astronomical Extension.

Req. 3 and 4 → very simple "Basic" syntax + "Enhancement" syntax.

All the data service must support the "Basic" syntax.

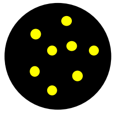
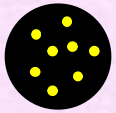
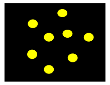
All the data service may support any "Enhancement" syntax as the need arises.

4. Observational Data Request in SQL

How the image data of the interested region is queried in SQL ?

```
SELECT    filtername, exposure, image
FROM      imageData
WHERE     region = region1 and
          spectrumBand = B and
          observationData between
          2001-10-20 and 2003-10-20
```

Any kind of parameters and returnable variables are considered as columns. The Image cutout request is a selection from a virtual table which has infinite number of image selection patterns.

region	spectrumBand	Other search parameters	Image
region1	R	…	
region1	B	…	
region2	R	…	
…	…	…	…

5. Basic Syntax

```
Select   ColumnName [[AS] AliasName] , … | *
From     TableName [[AS] AliasName]
Where    PrimaryCondition [AND PrimaryCondition]
```

- Only column name or “*” is specified in the selection list.
- An algebraic expression is not supported.
- Only one table is specified in “From” part.
- Table name and Column name may have alias name.
- Comparison operators: =, <, >, >=, <=, <>, LIKE, BETWEEN
- Logical operator: AND, NOT (OR is not supported.)
- Region Comparison operator: =, within, contains, overlaps
- Functions: Distance(), Point(), Circle(), Box() can be used in “Where” part.

6. Geometry Data Type

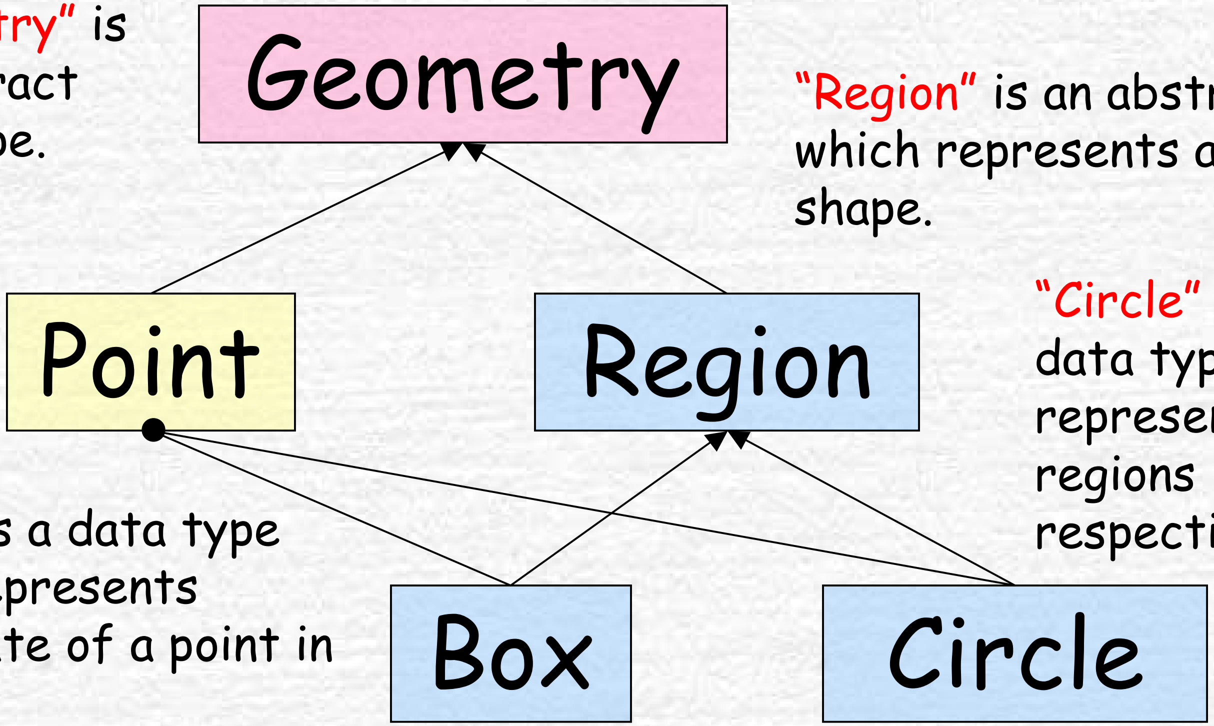
- Special to the astronomical query

  → Search on a region in the sky.

- A point is expressed by a pair of two numbers. Region is expression by a collection of points and the region size

  → Define structured data type (Geometry).

“Geometry” is an abstract data type.



Geometry

Point

Region

Box

Circle

“Region” is an abstract data type which represents a region of any shape.

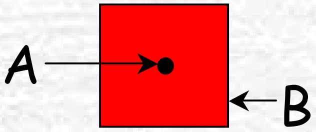
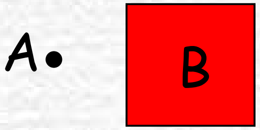
“Circle” and “Box” are data types which represent Circle and Box regions in the sky, respectively.

“Point” is a data type which represents coordinate of a point in the sky.

7. Region Comparison 1

[NOT] <SpacePoint> <RegionCompOper> <SpaceRegion>

[NOT] <SpaceRegion> <RegionCompOper> <SpacePoint>

Region Comparison	Meaning	Figure
A within B	Point A is within Region B.	
B contains A	Region B contains Point A.	
NOT A within B	Point A is outside Region B.	
NOT B contains A	Region B excludes Point A.	

<SpacePoint> ::= [Point](x, y , [‘frame’])

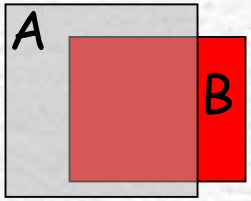
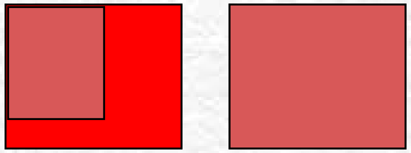
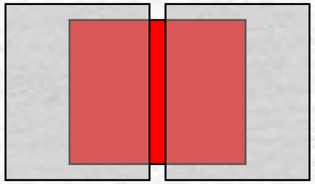
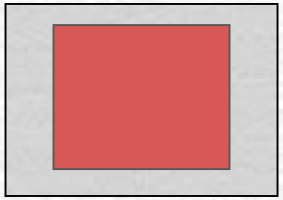
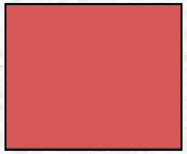
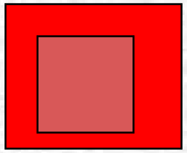
e.g. Point(13.2,-34.5), Point(32.1, -12.5, ‘ICRS’), (34.7, -26, ‘Gala’)

<SpaceRegion> ::= [Circle](<SpacePoint>, radius)
| [Box](<SpacePoint>, xsize, ysize)

e.g. Circle((23.7,-0.3), 2.3), Box((58.3,+1.2), 3.3, 3.3)

8. Region Comparison 2

[NOT] <SpaceRegion> <RegionCompOper> <SpaceRegion>

Region Comparison	Meaning	Image Atlas Data Service	Image Cutout Service
A = B	Region A is the smallest region which overlaps the largest part of B.		
A overlaps B	Region A is the smallest region which overlaps B.		Same as A = B
A contains B	Region A is the smallest region which contains B.		
A within B	Region A is the largest region which is contained in B.		Same as A = B

9. Examples of Basic Syntax

Query of catalog data for the specified region.

```
Select    ra, dec, mag_r
From      galaxy
Where     Point(ra, dec)  within  Circle((24.3, +5.0), 2.0))
           and mag_r < 24
```

can be omitted if it is trivial

Query of image retrieval URL for the specified region and the corresponding filter name.

```
Select    filter, imageURL
From      imageData
Where     region  =  Box((24.3, +5.0), 0.2))
```

"Pos = Point(24.2,5.0) and DeltaRa = 0.2 and DeltaDec = 0.2" is also valid syntax.

10. Enhanced Syntax

1. An algebraic expression in “Select” and “Where” part.
2. Logical operator “OR”.
3. Unit support.
4. Structured Data Type Enhancement.
5. Multiple tables in “From” part.
6. Join predicate at “From” part.
7. VOTable in “From” part and cross match with VOTables.
8. Use of Identifier for Table name (Portal): To identify a table in the VO uniquely.
9. UCD (Portal). UCDs used as representative of column name are resolved from the column metadata.
10. Omission of From part (Portal): Tables to be searched are determined from the condition described in “Where” part.

11.Table name identifier expression

Query to the multiple data services should identify the specified tables uniquely in the VO. → Use the VO standard on the identifier of resources.

```
<TableName> ::=
      [AuthorityName:][CatalogPath.]TableName
```

e.g.1 ivo://naoj/subaryu/spcam/galaxy
→ naoj:subaru.spcam.galaxy

e.g.2 ivo://adil.ncsa/surveys/96.JC.01
→ adil.ncsa:surveys.96\.JC\.01

Periods in the catalog path and table name must be escaped by a backslash.

12. External Table in "From" part

This is a syntax enhancement to describe join between a DB table and a VOTable.

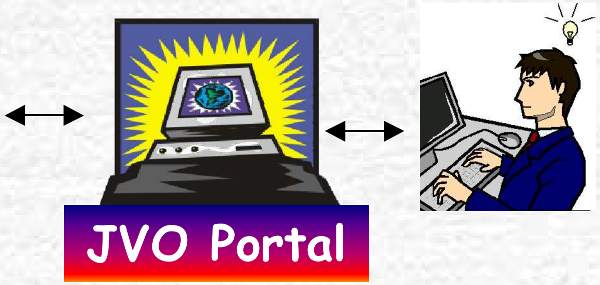
```
<ExternalTableName> ::=

EXT::<FileNumber>[ (.ResourceName)* .TableName ]
```

e.g. Search images corresponding to objects listed in a VOTable.

```
Select  vot.ra, vot.dec, img.imageURL
From    image as img, EXT::1.selectedGalaxy as vot
Where   img.region = Box((vot.ra, vot.dec), 0.1, 0.1)
```

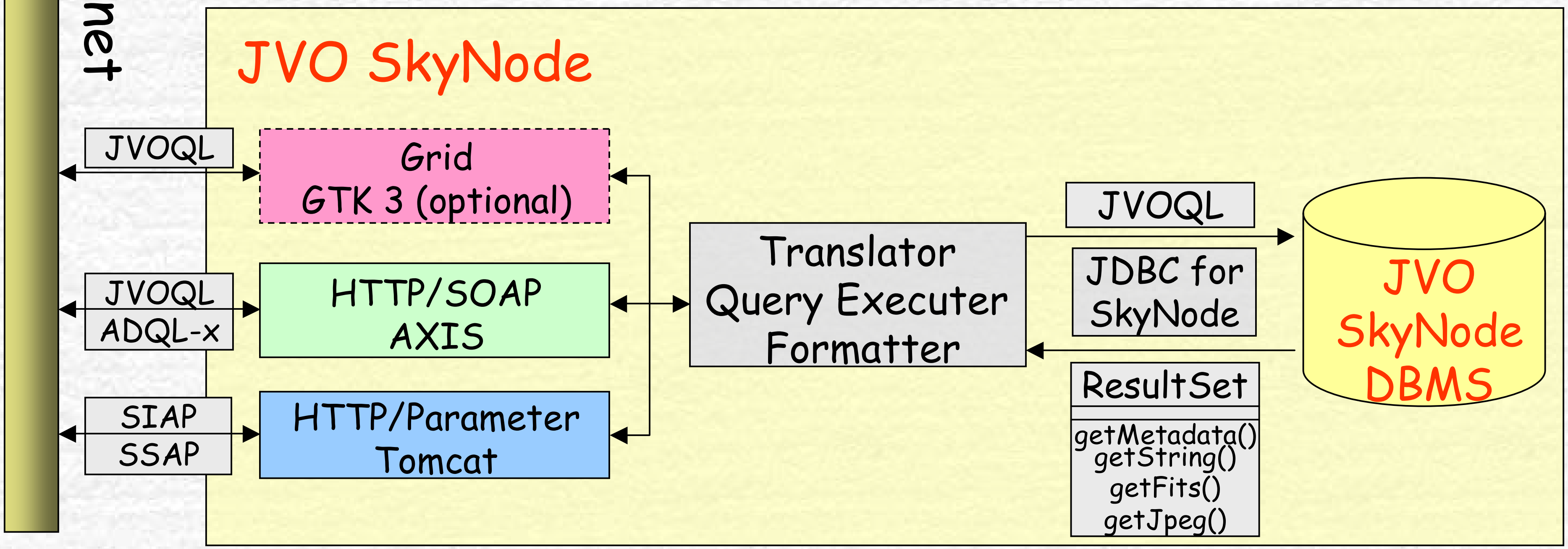
13. JVO SkyNode Architecture



JVO Portal

- JVO SkyNode is now under development to test the functionality of the JVOQL.
- Four kinds of query languages are converted to java class SelectSQL and query is executed through JDBC-like interface.
- Query results is obtained as ResultSet, and table data is formatted to VOTable or CSV file.

Internet



JVO SkyNode

JVOQL

Grid
GTK 3 (optional)

JVOQL
ADQL-x

HTTP/SOAP
AXIS

SIAP
SSAP

HTTP/Parameter
Tomcat

Translator
Query Executer
Formatter

JVOQL

JDBC for
SkyNode

ResultSet

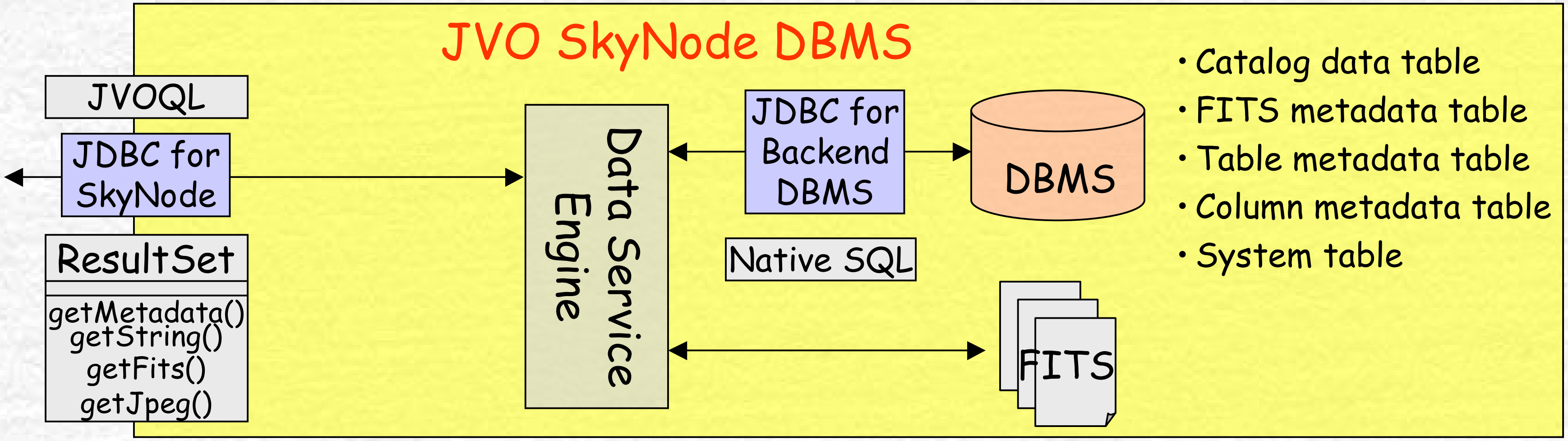
getMetadata()
getString()
getFits()
getJpeg()

JVO
SkyNode
DBMS

14. JVO SkyNode DBMS

JVO SkyNode DBMS :

- is an astronomical database system which accept JVOQL syntax and return observation data as well as tabular data ,
- includes DBMS which is used to store catalog data, FITS file metadata, and system information,
- can access to observational data of FITS files which are managed by unix file system,
- implemnts a JDBC-like interface, search request can be executed by Statement object and result is returned as ResultSet object.



JVO SkyNode DBMS

- Catalog data table
- FITS metadata table
- Table metadata table
- Column metadata table
- System table

15. Region Search using HTM index

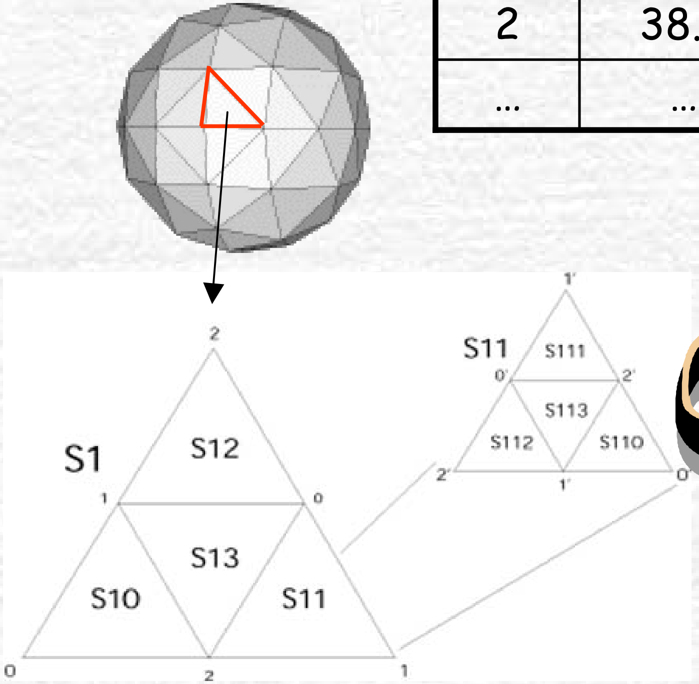
Region search is a common search criterion for an astronomical database. For efficient search data should be properly indexed on the objects' coordinates.

Catalog table

id	ra	dec	mag
1	12.3	-23.4	18.4
2	38.5	+34.2	16.5
…	…	…	…

HTM Index table

id	htm
1	16522516
2	16754765
…	…



http://www.sdss.jhu.edu/htm/

```
Select   ra, dec, mag
From     Catalog
Where    Point(ra,dec) within Box((20,+15), 1.0)
```

```
Select   c.ra, c.dec, c.mag
From     Catalog as c
         Natural Left Join htmIndex as i
Where    i.htm between 16522500 and 16522512
         OR
         i.htm between 16522500 and 16522512
         …
```

16. Implementation of Virtual Table

A Virtual Table become a real table by making a join of these tables.

Virtual column table is a collection of parameters specified by a requester (cutout region, spectrum range, …) and is dynamically created on each request.

Virtual Table = Virtual Column Table 1 × Virtual Column Table 2 × …

Metadata Table is a collection of metadata of the observation data.

× Table parameter Table 1 × Table parameter Table 2 × …

× Metadata Table

Table parameter table is a collection of parameters which characterize the table (telescope name, filter name,…) and exists as an real table.

17. Summary

- JVOQL is designed to be used as a VO standard query language.

- JVOQL can describe a query to get any kind of astronomical data.

- “Basic“ syntax specification is defined. All the data service must support this syntax.

- Optional “Enhancement“ syntax specifications are defined for describing a efficient query to the large DB.