

Standard Query Language for VO

-- Unification of ADQL and SIAP --

!! This is still under preparation !!
contents might change without notice

Yuji SHIRASAKI

National Astronomical Observatory of Japan

Integration of SkyNode and SIAP node

We need to make standards not only for query language but also supported protocol, the way of metadata query and specification of the returned VOTable.

	SAIP node	SkyNode	UnifiedSkyNode
Data	Image	Catalog	Catalog/Image/Spectrum/...
Query Language	Parameter, name of the parameters are defined.	SQL, no restriction to the name of column	SQL, define mandatory columns
Protocol	Http/Get/Parameter	Http/Put/Soap	Http/Put/Soap ?
Metadata Query	"FORMAT = METADATA" returns supported parameters and columns it will return.	Tables and Columns interface are defined.	Define standard metadata tables which have metadata of table and column. Query is described with SQL.
Query Output	VOTable, Some metadata must be included.	VOTable, no specification for the content	VOTable, FIELD element should have metadata stored in the metadata table.

In the last IVOA meeting @ Cambridge ...

I have shown that image query can be described in SQL syntax by introducing virtual column concept.

SIAP Parameters

POS [mandatory]
SIZE [mandatory]
FORMAT [mandatory]
INTERSECT [option]
...

SIAP query by HTTP/Get parameter request

```
http://jvo.nao.ac.jp/Image?  
Pos=34.3,-5.11&Size=0.01&Format=VOTable
```

SIAP parameters and returned metadata are taken as columns of the virtual table.

Virtual Columns

Pos	Size	Format	ImageURL
(23,+30)	1.0	fits	http://jvo.nao.ac.jp/ Image?id=124214
(23,+23)	0.3	jpeg	http://jvo.nao.ac.jp/ Image?id=124215
...

```
Select imageURL  
From naoj:image  
Where Pos = Point(23,+30)  
and Size = 1.0 and  
Format = 'fits'
```

"Pos" and "Size" columns have infinite number of value, so this table is a virtual table.

Column Type Allowed in the SkyNode

A column is classified into three types:

- **Ordinary Column**: column of the relation table
- **Parameter Column**: constant value of the table
 - e.g. limiting magnitude of the observations relevant to the table, typical error of the coordinate,...

```
Select limitingMagnitude  
From galaxy
```

- **Virtual Column**: continuous value of the table
 - e.g. parameter for specifying the image region,

```
Select imageURL  
From image  
Where region = Box((29,+10),1.0,1.0)
```

Data type allowed in the SkyNode

- **VOTable Data type**
 - boolean, int, long, float, double, character(*) ...
- **Space Data Type**
 - SpacePoint, SpaceRegionCircle, SpaceRegionBox
- **Range Data Type**
 - SpectrumRange, TemporalRange
- **Data Access URL Data Type**
 - ImageAccess, SpectrumAccess, ...

Metadata tables

- SkyNode must have **metadata tables** which describe attributes of the tables and columns of the SkyNode, and a client can query to the metadata tables.
- “**Tables**” metadata table has the following columns:
tableId, tableName, tableType, description
- “**Columns**” metadata table has the following columns:
columnId, columnName, tableId, dataType, coordinate, unit, ucd, columnType, precision, description

Mandatory and Optional Column

Define "Required", "Recommended" and "Optional" column. This is just an example.

ColumnName	UCD (TBD)	Catalog	Image	Comment
Pos	VOX:POS_EQ	Req	Req	Center coordinate of the search region in RA DEC frame. (degree)
DeltaRa	VOX:POS_EQ_DELTA_RA	Req	Req	Box size of the search region in RA coordinate. (degree)
DeltaDec	VOX:POS_EQ_DELTA_DEC	Req	Req	Box size of the search region in Dec coordinate. (degree)
Radius	VOX:POS_RADIUS	Rec	Rec	Radius of the search region. (degree)
DataFormat	VOX:DATA_FORMAT	X	Req	Data format of the observational data. (e.g. FITS, JPEG, HTML...)
OutputFormat	VOX:OUTPUT_FORMAT	Req	Req	Data format of the search result. (VOTable, CSV, HTML, ...)
Region	VOX:POS_REGION	Rec	Rec	Region of Interest in Sky
SpectrumRange	VOX:SPECTRUM_RANGE	Opt	Opt	Region of Interest in Spectrum
TemporalRange	VOX:TEMPORAL_RANGE	Opt	Opt	Region of Interest in Time

Output of Image Query

- According to the current SIAP specification, the following metadata must be included in the output VOTable.

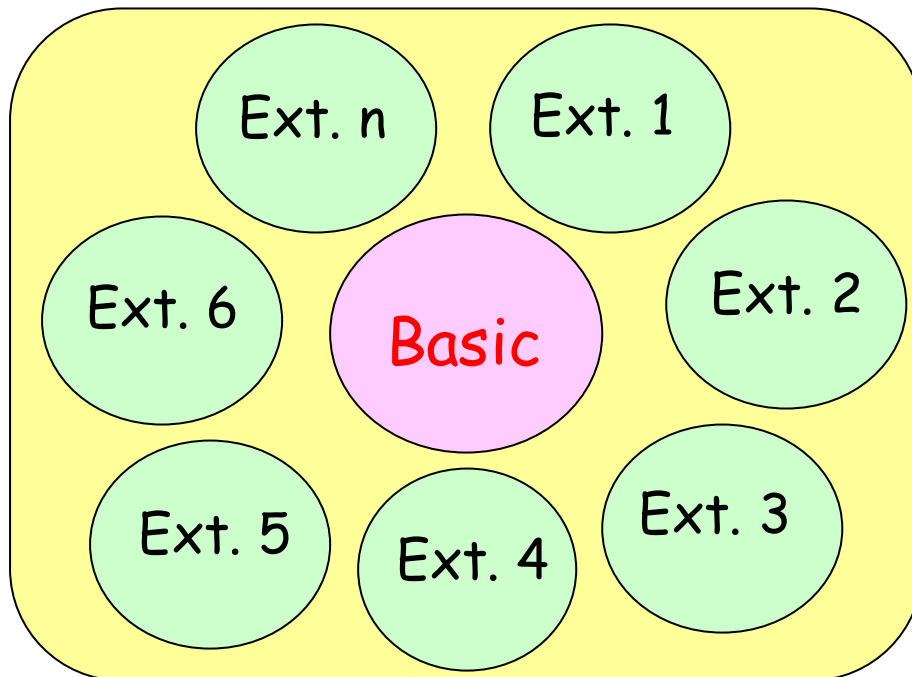
ImageTitle, Coordinate, NAXES, NAXIS, ImageScale, ImageFormat, ImageAccessRef.

- With the SQL specification, however, to get those metadata we must explicitly specify them in the "Select" list.
- I propose not to specify the content of output metadata, instead specify that
"Virtual image table must have columns related to ImageTitle, Coordinate,"

SQL Syntax Specification for VO

- VOQL should have **scalable** syntax:
 - Small size DBs → very simple syntax for easy implementation.
 - Large size DBs → sophisticated syntax for efficient data search.
 - We propose to define a minimal **basic syntax** as a standard and **optional syntax extension**.

hybrid syntax structure



"Basic" syntax **must** be implemented by all the VO data service.
Any "Extension-n" syntax **may** be implemented by each VO data service.

Registry will have information which extensions are implemented at each data service, or data service itself returns the information by "voqlSpec" interface, or querying to the **metadata table**. (TBD)

Basic Specification

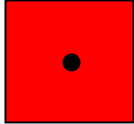
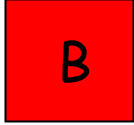
```
Select  ColumnName [[AS] AliasName] , ... | *  
From    TableName [[AS] AliasName]  
Where   Condition [AND Condition]
```

- Only column name or "*" is specified in the selection list.
- An algebraic expression is not supported.
- Only one table is specified at "From" part.
- Table name and Column name can 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()

Region Comparison Operator

[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.	A• 
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')

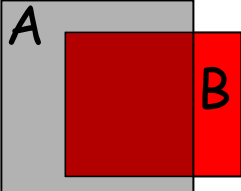
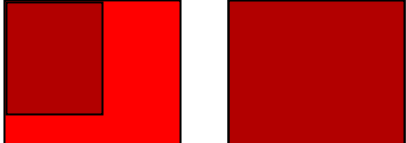
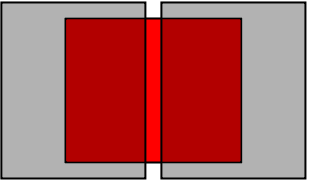
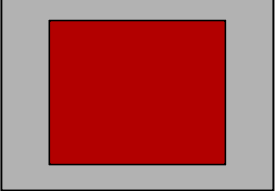

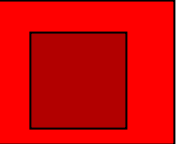
<SpaceRegion> ::= [Circle](<SpacePoint>, radius)

| [Box](<SpacePoint>, xsize, ysize)

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

Region Comparison

[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 $A = B$

Example for basic spec. VOQL

Select 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

Select image of 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

c.f. <http://jvo.nao.ac.jp/imageData?POS=24.2,5.0&SIZE=0.2>

Extension Packages of VOQL

Ext.1 An algebraic expression in "Select" and "Where" part.

Ext.2 Multiple tables in "From" part.

Ext.3 Join predicate at "From" part.

Ext.4 Logical operator "OR".

Ext.5 Data type Extension.

Ext.6 VOTable in "From" part and cross match with VOTables.

Ext.7 UCD (Portal). UCDs used as representative of column name are resolved from the column metadata.

Ext.8 Unit (Portal). Unit of the column is obtained from the column metadata and the value is translated accordingly.

Ext.9 Use of Identifier for Table name (Portal): To identify a table in the VO uniquely.

Ext.10 Omission of From part (Portal): Tables to be searched are determined from the condition described at Where part.

Use of Identifier for Table Name

<TableName> ::=

[AuthorityName:][CatalogDataPath.]TableName

e.g. "galaxy" table of the data resource "ivo://naoj/subaryu/spcam/" is specified as "naoj:subaru.spcam.galaxy".

External Table in From part

<ExternalTableName> ::=

EXT:(<Number> | [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:selectedGalaxy as vot

Where img.region = Box((vot.ra, vot.dec), 0.1, 0.1)

Xmatch Predicate

In addition to the ADQL 0.8.1 specification, column name can be specified.

Xmatch (table1, table2.(ra, dec), !table3, ... , sigma)

e.g. Try cross match between sdss and twomass catalog.

```
Select o.objId, o.ra, o.r, o.type, t.objId
```

```
From SDSS.photoPrimary o, TWOMASS.PhotoPrimary t
```

```
Where Xmatch(o.(ra, dec), t.(ra, dec), 3.5)
```


Example for extension syntax

Image query using catalog table.

```
Select  cat.ra, cat.dec, img.filter, img.ImageURL
From    galaxyCatalog cat, imageTable img
Where   img.region = Circle((cat.ra, cat.dec), 30 [arcsec])
        and Point(cat.ra, cat.dec) within Circle(234, +10, 30 [arcmin])
```

Image query using external VOTable file.

```
Select  img.filter, img.ImageURL
From    ext:galaxyCatalog cat, imageTable img
Where   img.region = Circle(cat.ra, cat.dec)
```

↑
"galaxyCatalog" table in an external file (VOTable)