

AimQL – AIM Query Language

by Hakan BULU
hakan.bulu@deu.edu.tr
(August 2012)

1. Introduction

AimQL is a simple SQL like query language to fetch the AIM Annotations.

2. Capabilities

- Although XQuery is key sensitive, AimQL is not a key sensitive language.
- In the WHERE clause, it supports inner and outer “AND” and “OR” conjunctions infinitely.
- It supports basic syntax checking.
- For incorrect AIM class and attributes usage, it throws informative exceptions.
- It supports same usage of the SQL’s “LIKE” operator. For example;
 - person.name Like ‘%h%’
 - person.name Like ‘%h’
 - person.name Like ‘h%’
- Supported operators are;

Function	Meaning
=	Equal
<>	Different From
>	Grater
<	Smaller
>=	Grater and Equal
<=	Smaller and Equal
LIKE	SQL’s Like

3. Structure

Basically, structure of the AimQL is as follows;

```
SELECT FROM [collectionName] WHERE [aimClassName].[aimAttributeName] [function] [value]  
AND/OR [aimClassName].[aimAttributeName] [function] [value] AND/OR ...
```

4. Usage

- AimQL is available in the AIM API. You can use
getImageAnnotationsFromServerWithAimQuery method of AnnotationGetter
class.

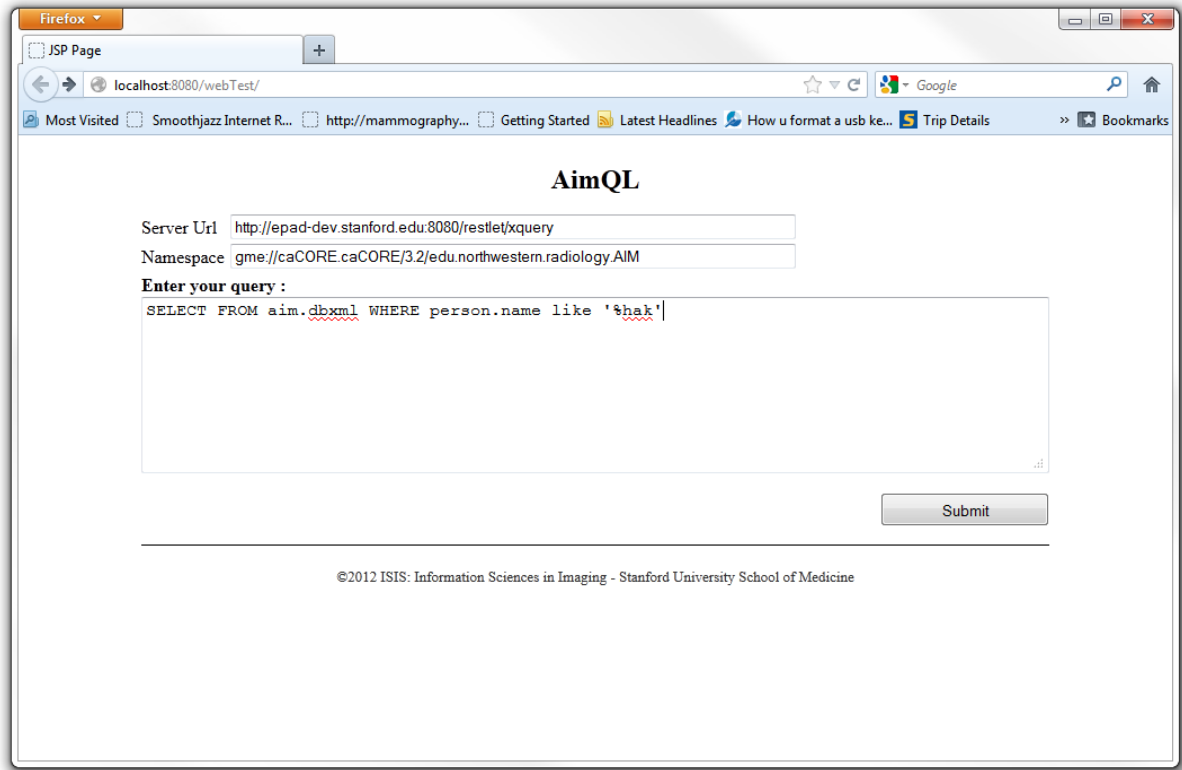
```
List<ImageAnnotation> testList =  
AnnotationGetter.getImageAnnotationsFromServerWithAimQuery(serverUrlDownload,  
namespace, "SELECT FROM aim.dbxml WHERE (person.Name Like '%h')");
```

```

for (int i = 0; i < testList.size(); i++) {
System.out.println(testList.get(i).getListPerson().get(0).getName());
}

```

- AimQL is available in the web page.



5. Examples

- Fetch all ImageAnnotation objects where uniqueIdentifier equal to "ID-12345-x"

```
SELECT FROM aim.dbxml WHERE (ImageAnnotation.uniqueIdentifier = 'ID-12345-x')
```

- Fetch all ImageAnnotation objects where *person name* starts with "hak"

```
SELECT FROM aim.dbxml WHERE person.name like '%hak'
```

- Fetch all ImageAnnotation objects where *person name* starts with "hak" or "cem" and *person cagridId* greater than 4

```
SELECT FROM aim.dbxml WHERE (person.name like '%hak' or person.name like '%cem') and person.cagridId > 4
```

- Fetch all ImageAnnotation objects where *ImagingObservation codeMeaning* contains "Mass"

```
SELECT FROM aim.dbxml WHERE ImagingObservation.codeMeaning LIKE '%Mass%'
```