# S3
Simple Self-described Service protocol.

Carlos Rodrigo Blanco[1]
Enrique Solano[1]
Miguel Cerviño[2]

[1]CAB,INTA-CSIC; SVO
[2]IAA, CSIC; SVO

IVOA interoperability meeting
Urbana, May 21-25, 2012

Introduction
Operations
Conclusions

**Requirements**
Generalization

SVO

## History: Requirements

We wanted a protocol for theoretical data so that it was:

- Similar to other simple protocols.
    - SIAP, SSAP...

- Simple to develop.
    - The simpler the development of the service is, the more people will be willing to implement it $\Rightarrow$ their data in the VO.

- Flexible.
    - The relevant characteristics (parameters, options available, internal data structure) can be very different for different models.

Introduction
Operations
Conclusions

Requirements
Generalization

SVO

## History: Generalization

Idea:

- take the SSAP protocol,

- get the abstract ideas in it,

- apply them to other types of data:
  let services describe what they are offering.

How:

- $\sim$ 2005: TSAP (SSAP), for theoretical spectra

- $\sim$ 2008: S3 (Simple Self-described Service),
  generalization for other types of theoretical data

Introduction    Capabilities
Operations    queryData
Conclusions    getData

SVO

## Operations

Three main operations:

- Service description *(getCapabilities)*:
    - what queries can be done to the service? (valid parameters)

- Seach data query *(queryData)*:
    - Which results (files) are available for a given range of those parameters?

- Give me a particular file *(getData)*.
    - Or preprocess it before downloading.

Introduction
Operations
Conclusions

Capabilities
queryData
getData

# (1) Capabilities: what is the service offering

- What type of data is the service offering
  - SSAP: spectra (time series?)
  - SIAP: images
  - theory/other: depends.

- Which parameters can be used for searching, and what values are allowed for each of them?
  - SSAP, SIAP...: POS, SIZE, BAND...
  - theory/other: those specified by the service.

Introduction
Operations
Conclusions

Capabilities
queryData
getData

# (1) Capabilities: what is the service offering

*http://.../?REQUEST=getCapabilities*

```
<RESOURCE type="capabilities">
 <DESCRIPTION>...</DESCRIPTION>
 <PARAM name="ServiceType" value="Isochrone">
 ...

 <PARAM name="INPUT:age" unit="Gyr" ...>
  <VALUES>
   <MIN value="10"/>
   <MAX value="100"/>
  </VALUES>
 </PARAM>

 <PARAM name="INPUT:Z" unit="" ...>
  <VALUES type="actual">
  <OPTION value="10"/>
  <OPTION value="20"/>
  <OPTION value="50"/>
  </VALUES>
 </PARAM>
```

- Service offers:
  - Isochrones.

- Two parameters for queries:
  - age.
  - Z.

# (2) queryData

What results are available for given (range of) values for the accepted parameters?

The Query      (http request)

- How to build the query
  - http://.../?param1=value1&param2=value1/value2...
  - params specified by the server in getCapabilities.

- How to specify values, ranges, lists of values
  - range: param=value1/value2
  - list: param=value1,value2,value3

Introduction
Operations
Conclusions

Capabilities
queryData
getData

SVO

# (2) queryData

What results are available for given (range of) values for the accepted parameters?

The Answer     (VOTable)

- INFO element (OK, ERROR...)
- Some PARAMS explaining the results
  - GROUP/S of PARAMS explaining the AccessURL to data files (when applicable)

- A Table with the list of results:
  - including a cell for the AccessURL to the particular file(s) (when applicable).
  - (there could be several data files available for the same result)

Introduction
**Operations**
Conclusions

Capabilities
queryData
getData

SVO

# (2) queryData

*http://.../?age=10/30&Z=10,20*

```
<INFO name="QUERY_STATUS" value="OK"/>
<RESOURCE type="results">

 ...
  <TABLE>
   <FIELD NAME="age"/>
   <FIELD NAME="Z"/>
   <FIELD NAME="datalink"/>
   <TABLEDATA>
    <TR>
     <TD>15</TD>
     <TD>10</TD>
     <TD>http://.../s3.php?fileid=12</TD>
    </TR>
    <TR>
     <TD>15</TD>
     <TD>20</TD>
     <TD>http://.../s3.php?fileid=23</TD>
    </TR>

    ...
   </TABLEDATA>
  </TABLE>
```
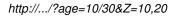
- Two available results for the query.
- Different values of the parameters.
- Link to download the final datafile (the isochrone).

Introduction
**Operations**
Conclusions

Capabilities
queryData
getData

SVO

# (2) queryData

*http://.../?age=10/30&Z=10,20*

```
...
<TABLE>
 <GROUP ...>
  <FIELDref id="link1"/>
  <PARAM NAME="dataxis1" value="M" .../>
  <PARAM NAME="dataxis2" value="Teff" .../>
  <PARAM NAME="dataxis3" value="Logg" .../>
 </GROUP>

 ...
<FIELD NAME="age" .../>
<FIELD NAME="Z" .../>
<FIELD ID=link1 NAME="datalink" .../>
<TABLEDATA>
 <TR>
  <TD>15</TD>
  <TD>10</TD>
  <TD>http://.../s3.php?fileid=12</TD>
 </TR>
</TABLEDATA>
</TABLE>
```

Extra info:

- The available file will contain (at least) three columns (fields):
  - M
  - Teff
  - Logg

- Some preprocessing can be requested on those columns before downloading.

Introduction
**Operations**
Conclusions

Capabilities
queryData
getData

# (2) queryData

*http://.../?age=10/30&Z=10,20*

```
...
<TABLE>
 <GROUP ...>
  <FIELDref id="link1"/>
  <PARAM NAME="dataaxis1" value="M" ...>
   <VALUES>
    <MIN value="10"/>
    <MAX value="1000"/>
   </VALUES>
  <PARAM NAME="dataaxis2" value="Teff" .../>
  <PARAM NAME="dataaxis3" value="Logg" .../>
 </GROUP>

 ...
 <FIELD NAME="age" .../>
 <FIELD NAME="Z" .../>
 <FIELD ID=link1 NAME="datalink" .../>
 <TABLEDATA>
  <TR>
   <TD>15</TD>
   <TD>10</TD>
   <TD>http://.../s3.php?fileid=12</TD>
  </TR>
```

Extra info:

- M range in the file is:
    - min: 10
    - max: 1000

Introduction    Capabilities
**Operations**    queryData
Conclusions    **getData**

SVO

# (3) getData

### Give me a particular file

- Just use the URL to download a file
  - (obtained in the queryData operation).
  - http://.../s3.php?fileid=12

- Ask for some preprocessing before downloading
  - change resolution, cutout, etc
  - http://.../s3.php?fileid=12&CUTOUT=Teff,500/700
  - Other preprocessings: resolution, resample, graphmap, etc.

## Useful approach

> 100 different services implemented with this approach

- $\sim$ **25** services for **theoretical spectra**.
- $\sim$ **4** services for **observational templates**.
- $\sim$ **20** services for **synthetic photometry**.
- $\sim$ **20** services for **chi-square fitting** of observed photometry with theoretical data.
- $\sim$ **20** services for **bayesian analysis** of observed photometry compared with theoretical data.
- $\sim$ **8** services for **isochrones** and evolutionary tracks
- $\sim$ **4** services for complex **asteroseismology** models (stellar structure and oscillation properties)
- **1 Filter profile** service providing filter transmission curves and calibration properties.

## Useful approach

- VOSA application uses these services

  - to get theoretical spectra (with cutout and resampling) and synthetic photometry to make plots,
  - to chi-square fit observed photometry with theoretical models and estimate physical properties of the objects,
  - to make a bayesian analysis of the parameter space,
  - to get filter properties to transform magnitudes,
  - to build HR diagrams and estimate object age and mass,
  - etc

- other applications could use the same services.

- or VOSA could use similar services implemented by other people.

# IVOA Note

- IVOA Note in 2008 about S3.
- Some improvements made since them.

## Summary

- Easy to develop different types of services.

- This talk is just a summary, data models (SpecDM, SimDM), vocabularies and other details would enrich the services.

- Useful for microsimulations (even complex).

- Maybe useful for other types of data.

- Some discussion/inputs would be nice.

$\Rightarrow$ We propose to start the discussion to promoted it to a formal specification.

# THANK YOU!