Self described service
Post DataLink solutions

F.Bonnarel

EURO VO CoSADIE Collaborative and Sustainable Astronomical Data Infrastructure for Europe

cnrs INSU Observer & comprendre

IVOA

UNIVERSITÉ DE STRASBOURG

CENTRE DE DONNÉES ASTRONOMIQUES DE STRASBOURG

Self described services
Two solutions

- Parameters described in the DataLink response
  - Find a way to associate to corresponding record (VO-DML ?)
  - May be difficult to parse
  - DataLink should not depend on a complex still undecided protocol
  - DataLink and Services definition should remain independant
- Parameters described by a specific answer of the service, eg :
  - Michel's et al solution
  - S3 solution

The TapHandle solution
Laurent Michel

- An xml document defines and describes the input parameters
- Is retrieved in case of « empty » query to the service
- Could be rewritten in Json
- Easy to parse, local xml

```
!   !   <description>Number of counts per bin</description>!!
!   !   <unit>none</description>!    !inkform servicetype="download" identifier="PowerLaw" format="image/gif">
!   <!--
!   !   database name: ThreeXMM
!   !   database url: http://obs-he-lm:8888/3XMM
!   !   oid: 1160803203386703876
!   !   model: PowerLaw
!   -->
!   <description>
!   Apply a power law model on a XMM-Newton EPIC spectrum
!   </description>
!   <baseurl>
!   http://obs-he-lm:8888/3XMM/fitmodelonspectrum?oid=1160803203386703876&model=powlaw
!   </baseurl>
!   <parameter identifier="binsize" mandatory="true"
!   ucd="spect.binSize" datatype="int">!
!   !   <range type="enum">!!
!   !   !  <value>1</value>!   !    !
!   !   !  <value>5</value>!   !    !
!   !   !  <value type="default">10</value>!    !  !
!   !   !  <value>25</value>! !    !
!   !   !  <value>50</value>! !
!   !   </range>!   !
!   </parameter>
!   <parameter identifier="nh" mandatory="true" !
!   ucd=phys.abund.X" datatype="real">!
!   !   <description>Galactical NH</description>
!   !   <unit>1e22cm-2</description>
!   !   <range type="range">!   !    !
!   !   !  <value type="min">0</value>!!   !
!   !   !  <value type="max">1</value>!!   !
!   !   !  <value type="default">0.01</value>! !   !
!   !   </range>!
!   </parameter>
!
!   <parameter identifier="alpha" mandatory="true" !
!     ucd="meta.code;spect.index“ datatype="real">
!
!   <description>Photon index of power law</description>!
!   !   <unit>none</description>!    !
!   !   <range type="range">!   !
!   !   !  <value type="min">1</value>!!   !
!   !   !  <value type="max">9</value>!!   !
!   !   !  <value type="default">1.7</value>! !    !
!   !   </range>!
!   </parameter>
</linkform>
```

S3 solution

- INPUT parameters are described in a VOTABLE
- Description retrieved when service queried via FORMAT=METADATA
- DESCRIPTION and possible VALUES

```
<VOTABLE version="1.1"
xsi:schemaLocation="http://www.ivoa.net/xml/VOTable/v1.1">
<RESOURCE type="meta">
    <DESCRIPTION>
        Theoretical Isochrones for the MyModel model.
    </DESCRIPTION>
    <PARAM name="INPUT:age:range" ucd="phys.age" unit=”Gyr”>
        <DESCRIPTION>
            Age of the star in Gyr.
        </DESCRIPTION>
        <VALUES type="actual">
            <OPTION value="1"/>
            <OPTION value="2"/>
            <OPTION value="12"/>
        </VALUES>
    </PARAM>
    <PARAM name="INPUT:metallicity" ucd="..." unit=””>
        <DESCRIPTION>
            Metallicity of the star defined as Fe/H
        </DESCRIPTION>
        <VALUES type="actual">
            <OPTION value="0"/>
            <OPTION value="0.5"/>
            <OPTION value="1"/>
        </VALUES>
    </PARAM>
    <PARAM name="INPUT:logg:fixed" ucd="phys.gravity" unit=”” value=”1”/>
</RESOURCE>
</VOTABLE>
```

Idea

- Use the VOSI syntax of parameter description
- Provide an xml document consistent with
- Extension to vr:InputParameter to be defined (ranges, defaut values, etc ...)
- VOTABLE and Json serialisation possible ?

vs:BaseParam Type Schema Definition

```
<xs:complexType name="BaseParam">
   <xs:sequence>
      <xs:element name="name" type="xs:token" minOccurs="0"/>
      <xs:element name="description" type="xs:token" minOccurs="0"/>
      <xs:element name="unit" type="xs:token" minOccurs="0"/>
      <xs:element name="ucd" type="xs:token" minOccurs="0"/>
      <xs:element name="utype" type="xs:token" minOccurs="0"/>
   </xs:sequence>
   <xs:anyAttribute namespace="##other" />
</xs:complexType>
```

----

vs:InputParam Type Schema Definition

```
<xs:complexType name="InputParam">
   <xs:complexContent>
      <xs:extension base="vs:BaseParam">
         <xs:sequence>
            <xs:element name="dataType" type="vs:SimpleDataType"
                        minOccurs="0"/>
         </xs:sequence>
         <xs:attribute name="use" type="vs:ParamUse" default="optional"/>
         <xs:attribute name="std" type="xs:boolean" default="true"/>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>
```