International
Virtual
Observatory
Alliance

VODML Mapping Lite Syntax

Version 1.0

IVOA Note 2020-04-22

Working group
DM
This version
http://www.ivoa.net/documents/cab-msd/20200422
Latest version
http://www.ivoa.net/documents/cab-msd
Previous versions
This is the first public release
Author(s)
François Bonnarel, Gilles Landais, Laurent Michel, Jesus Salgado
Editor(s)
Laurent Michel

Abstract

???? Abstract ????

Status of this document

This is an IVOA Note expressing suggestions from and opinions of the authors. It is intended to share best practices, possible approaches, or other perspectives on interoperability with the Virtual Observatory. It should not be referenced or otherwise interpreted as a standard specification.

A list of current IVOA Recommendations and other technical documents can be found at http://www.ivoa.net/documents/.

Contents

Acknowledgments

???? Or remove the section header ????

Conformance-related definitions

The words "MUST", "SHALL", "SHOULD", "MAY", "RECOMMENDED", and "OPTIONAL" (in upper or lower case) used in this document are to be interpreted as described in IETF standard RFC2119 (**?**).

The *Virtual Observatory (VO)* is a general term for a collection of federated resources that can be used to conduct astronomical research, education, and outreach. The International Virtual Observatory Alliance (IVOA) is a global collaboration of separately funded projects to develop standards and infrastructure that enable VO applications.

1 Introduction

???? Write something ????

1.1 Role within the VO Architecture

Fig. 1 shows the role this document plays within the IVOA architecture (**?**).

???? and so on, LaTeX as you know and love it. ????

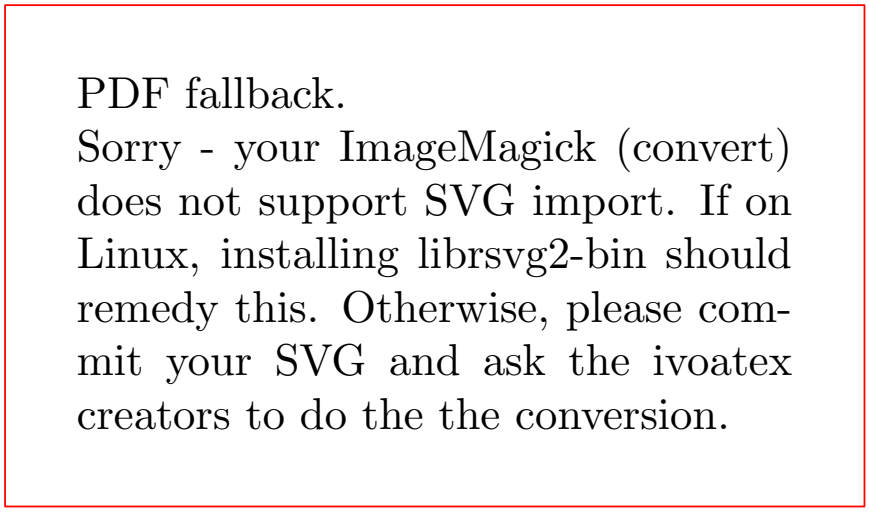

Figure 1: Architecture diagram for this document

2 Use Cases and Requirement

2.1 Shy Annotations

- Must not alter the original VOTable content
- Reasonably compact
- Reasonably human readable
- Must be parsable without breaking working code

2.2 Faithful Annotations

- The structure of the annotation is faithful to that of the model

2.3 Different Usage Levels

- Easy to ignore
- Just detect the model on which data are mapped
- Just get the metadata
- Get full model instances for each table row

2.4 Easy to Build

- The mapping structure is independant of the data structure
- Possibility of building components and templates

2.5 Complex Data Mapping

- Multitable
- Column filtering
- Data grouping

This alternative mapping syntax keeps the same basics as the original proposal:

3 Syntax

- This mapping syntax support directive for the clients which are not part of the model (e.g. aggregation request or data filters).
- The mapping as an explicit entry point, telling to the client what is the VOTable content.
- Processing this alternate syntax require for the client to apply rules not states in the VOTable itself.
- The mapping is located in a `<VODML>` block, child of `<VOTABLE>`.
- The mapping elements reflect the model structure.
- The `<VODML>` block starts with a list of implemented models.
- There is one `<TEMPLATES>` per mapped `<TABLE>`.
- There is one `<GLOBALS>` block containing data shared by the whole mapping.

3.1 GLOBALS

Contains `INSTANCE`s with fixed values that can be used everywhere in the `VODML`.

```
<GLOBALS>
  <INSTANCE ID="SpaceCoordFrame" dmrole="">
    <INSTANCE dmrole="coords:SpaceFrame.refPosition" dmtype="coords:StdRefLocation">
      <VALUE dmrole="coords:StdRefLocation.position" dmtype="ivoa:string" value="NoS
    </INSTANCE>
    <VALUE dmrole="coords:SpaceFrame.spaceRefFrame" dmtype="ivoa:string" value="ICRS
    <VALUE dmrole="coords:SpaceFrame.equinox" dmtype="coords:Epoch" value="NoSet"/>
  </INSTANCE>
  <INSTANCE >
```

```
    ...
  </INSTANCE>
  ...
</GLOBALS>
```

Listing 1: INSTANCE bloc example

Child	Role
`INSTANCE`	`GLOBALS` children must be `INSTANCE` .

Table 1: Supported `GLOBALS` children

`GLOBALS` has no attributes.

3.2 INSTANCE

Mapping for either object type or a datatype instances.

```
<INSTANCE dmrole="ds:dataset.Dataset.dataID" dmtype="ds:dataset.DataID" ID="_ds_">
    <VALUE dmrole="ds:dataset.DataID.title" value="Gaia TS Mapping Test" />
    <VALUE dmrole="ds:dataset.DataID.datasetID" value="ivoa://gaia/ts/12345" />
    <VALUE dmrole="ds:dataset.DataID.creatorDID" value="ivoa://esa/gaia/" />
    <VALUE dmrole="ds:dataset.DataID.version" value="0.0" />
    <VALUE dmrole="ds:dataset.DataID.date" value="2018:11:11" />
    <VALUE dmrole="ds:dataset.DataID.creationType" value="LiteMappingTest" />
    <INSTANCE dmrole="ds:dataset.DataID.creator" dmtype="ds:dataset.Creator">
        <INSTANCE dmrole="ds:party.Role.party" dmtype="ds:party.Individual">
            <VALUE dmrole="ds:party.Party.name" value="VODML-Team" />
       </INSTANCE>
    </INSTANCE>
</INSTANCE>
```

Listing 2: INSTANCE bloc example

Child	Role
`INSTANCE`	Another embedded instance .
`VALUE`	Primitive attribute .
`COMPOSITION`	Composition with a limited set of `INSTANCE` e.g. author list
`ARRAY`	Composition with a set of `INSTANCE` corresponding each to one row of the data table.
`FILTER`	TbC

Table 2: Supported `INSTANCE` children

Attribute	Requ. level	Role
@dmrole	Mandatory	VODML role of the instance. May be empty for instances child of `GLOBALS`
@dmtype	Mandatory except for reference	VODML type of the instance.
@dmref	Mandatory for reference	reference to another instance in the mapping bloc.
@ID	Mandatory if the instance is referenced by other instances	Unique identifier of the instance.

Table 3: Supported attributes for `INSTANCE`

@dmrole	@dmref	@dmtype	use case
yes	yes		Reference to another instance. The element must have no child
yes		yes	Instance serialization The element must enclose the instance content

Table 4: Supported attribute patterns for `INSTANCE`

3.3 VALUE

Mapping for primitive attributes. `VALUE` are the model leaves that point onto real data.

```
<INSTANCE dmrole="model:value.example" dmtype="model:value.Example">
    <VALUE dmrole="model:preset.value" value="Preset Value" />
    <VALUE dmrole="model:ref.value" ref="fieldID" />
    <VALUE dmrole="model:reforpreset.value" value="Preset Value" ref="fieldID" />
 </INSTANCE>
```

Listing 3: VALUE examples

`VALUE`s have no children.

A Changes from Previous Versions

No previous versions yet.

Attribute	Requ. level	Role
@dmrole	MUST	VODML role of the instance attribute.
@dmtype	MUST	VODML type of the instance attribute.
@value	MUST if no `@ref` element attribute. MAY if `@ref` element attribute	Value of the instance attribute. If `VALUE` has also a `@ref`, `@ref` MUST be resolved first. `VALUE` MUST be taken when `@ref` cannot be resolved
@ref	MUST if no `@value` element attribute. MAY if `@value` element attribute	Reference of the data element (`FIELD` or `PARAM`). MUST refer to an element of the `TABLE` referenced by the current `TEMPLATE` The client MUST first look for a `FIELD` matching `@ref`. In case of failure, it MUST look for a `PARAM`

Table 5: Supported attributes for `VALUE`

@dmrole	@dmtype	@ref	@value	Role
yes	yes	yes		The instance attribute must take the value pointed by `@ref`
yes	yes		yes	The instance attribute must take the value set in `@value`
yes	yes	yes	yes	The instance attribute must take the value pointed by `@ref` and the this set in `@value` if `@ref` cannot be resolved

Table 6: Supported attribute patterns for `VALUE`