

VO-DML Tooling Update

Paul Harrison (JBO)
IVOA Interop Spring 2022



Contents

- ✦ VO-DML Tooling Updates
- ✦ VODSL advert
- ✦ Desired VO-DML changes
- ✦ See also ProposalDM talk

Gradle Tooling

- Introduced the new effort to update the VO-DML tooling at the last Interop
 - Focus on text-represented model-first development with code generation (cf. mapping existing tabular data).
 - currently code does serialisation to XML and Rdb
 - easy to serialise to JSON too (not VOTable)
 - automated TAP service generation a future possibility
 - Allow easy modular development of Data Models.
 - Make collaborative development of DMs less painful.
- Gradle-based tooling now 'standard' (plug-in at version 0.3.7)
 - <https://github.com/ivoa/vo-dml/blob/master/tools/ReadMe.md>
 - Additional functionality and improvements have been driven by use on ProposalDM

Updates since v0.1

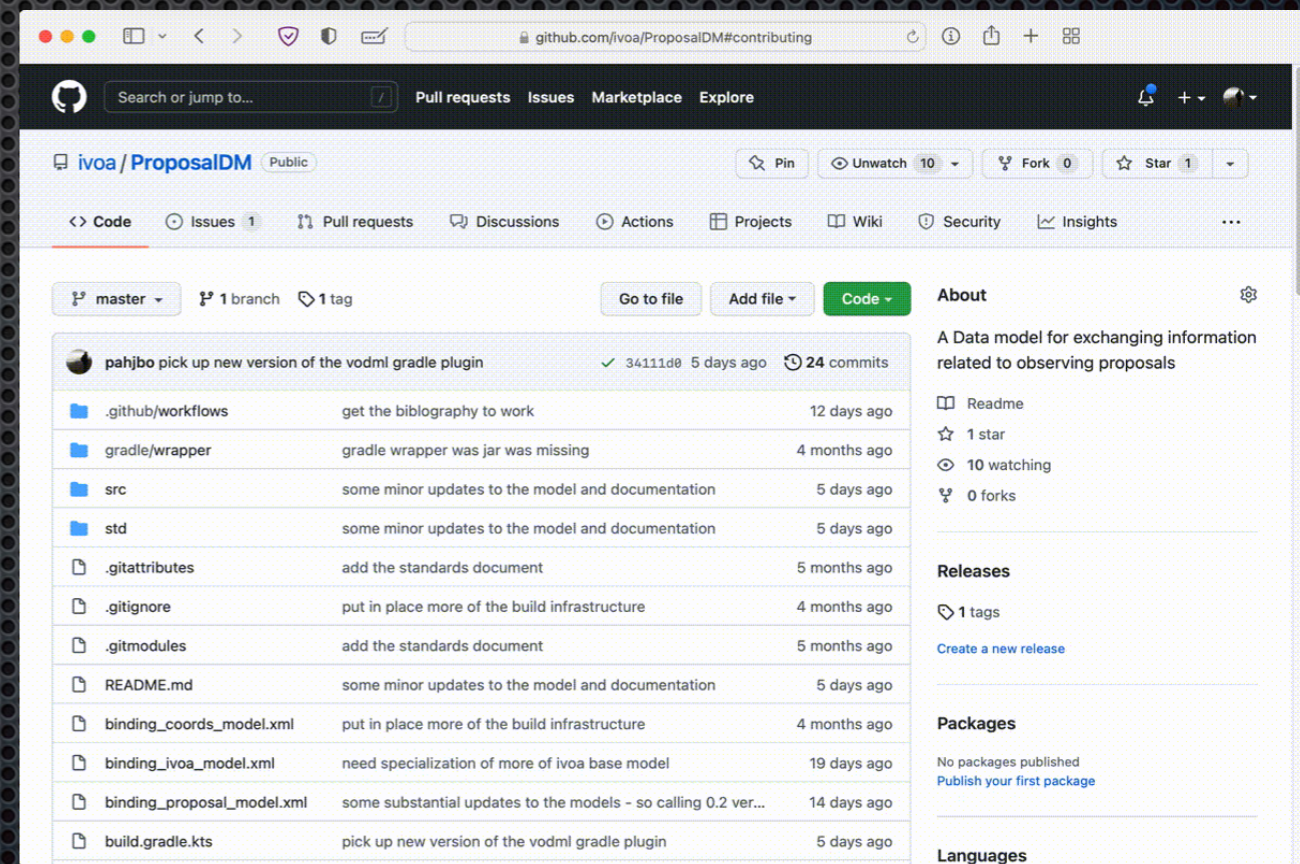
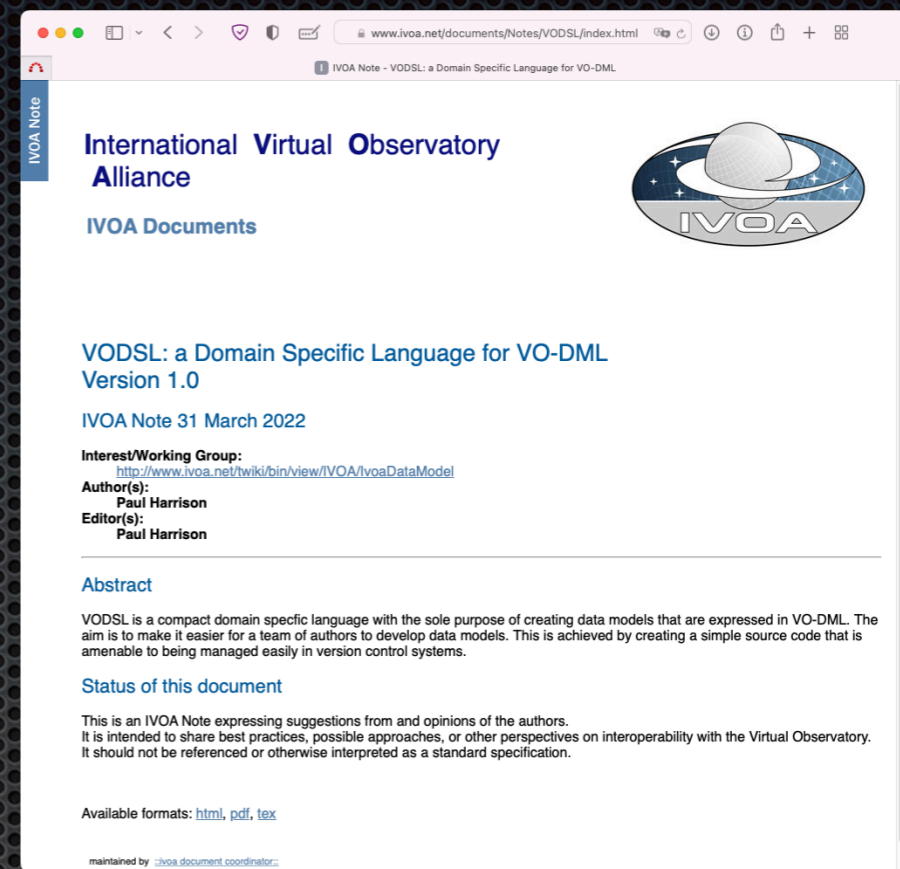
- ✦ Generated Java and associated round-trip serialisations to XML and RDB schema tweaked to work with every model tried so far...
- ✦ Generated Java and VO-DML of model packaged into jar
 - ✦ So can be used as dependency in modular build
 - ✦ Included models are automatically found in jar (reference only by file name)
- ✦ Added an XMI task for UML->VO-DML
- ✦ Added a vodsItoVodml task for VODSL -> VO-DML

Modularity Demo

- ✦ e.g. Integrating in with PhotDM
 - ✦ fork and make changes to use tooling
 - ✦ gradle vodmlValidate
 - ✦ gradle vodmlDoc
 - ✦ gradle vodmlGenerateJava
 - ✦ *Note:* no interaction with the VO-DML GitHub repository

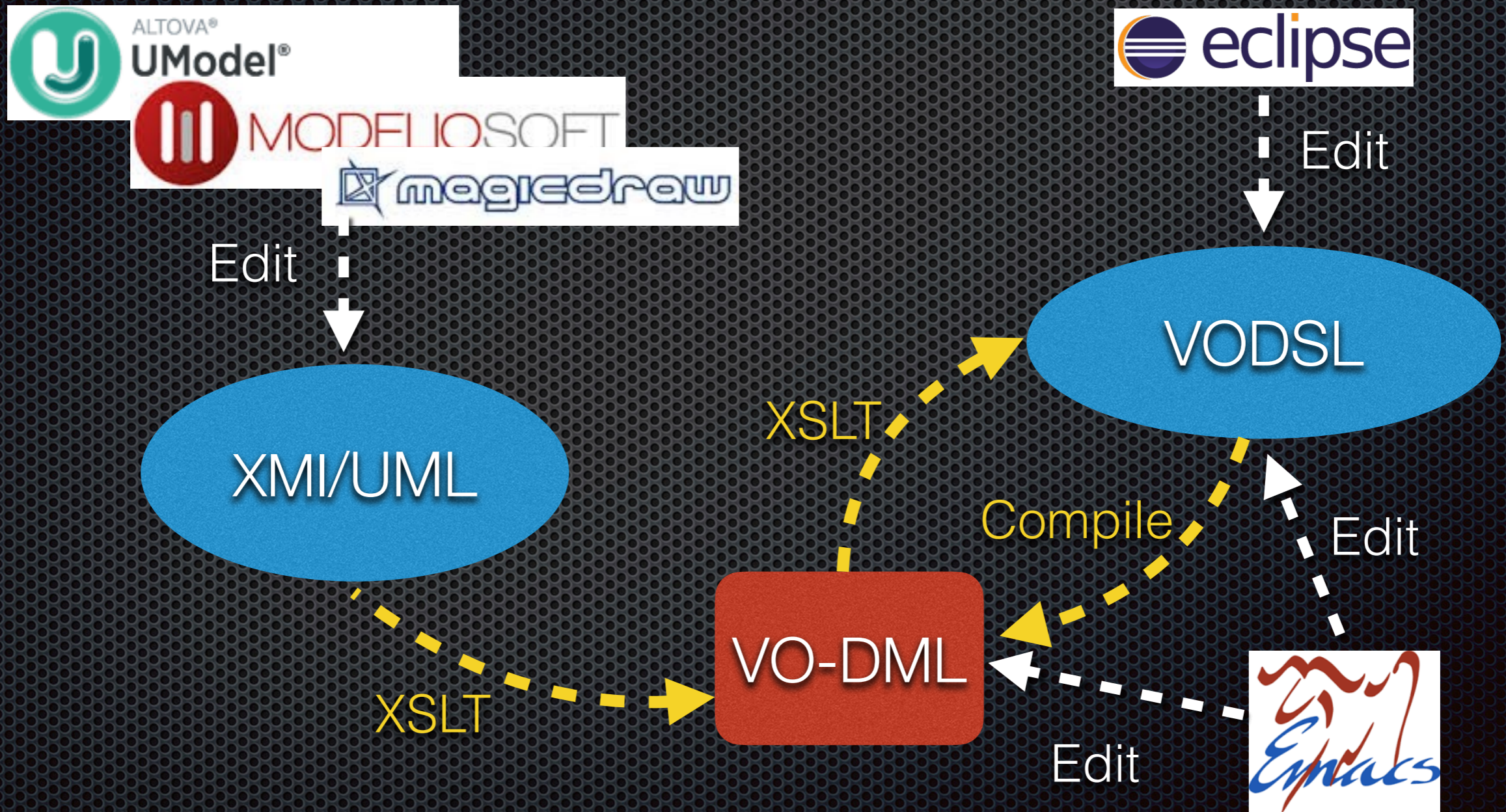
VODSL

- ✦ A concise language for creating VO-DML
 - ✦ with an associated Eclipse plug-in
- ✦ Introduced 8(!) years ago, but recently...
 - ✦ Published a note
 - ✦ Eclipse editor install much easier
 - ✦ “command line” support in the VO-DML gradle plugin.



Ecosystem

- VODSL complements rather than competes with VO-DML



VODSL route vs UML route

VODSL

UML

Easier to perform global
refactoring

vs

Easier to visualise the whole
model

Instantaneous validation

vs

Full validation only after
XSLT transformation of XMI

Easier to merge
contributions from two
authors textually

vs

Rely on UML tool to have
model merging facility

VO-DML issues

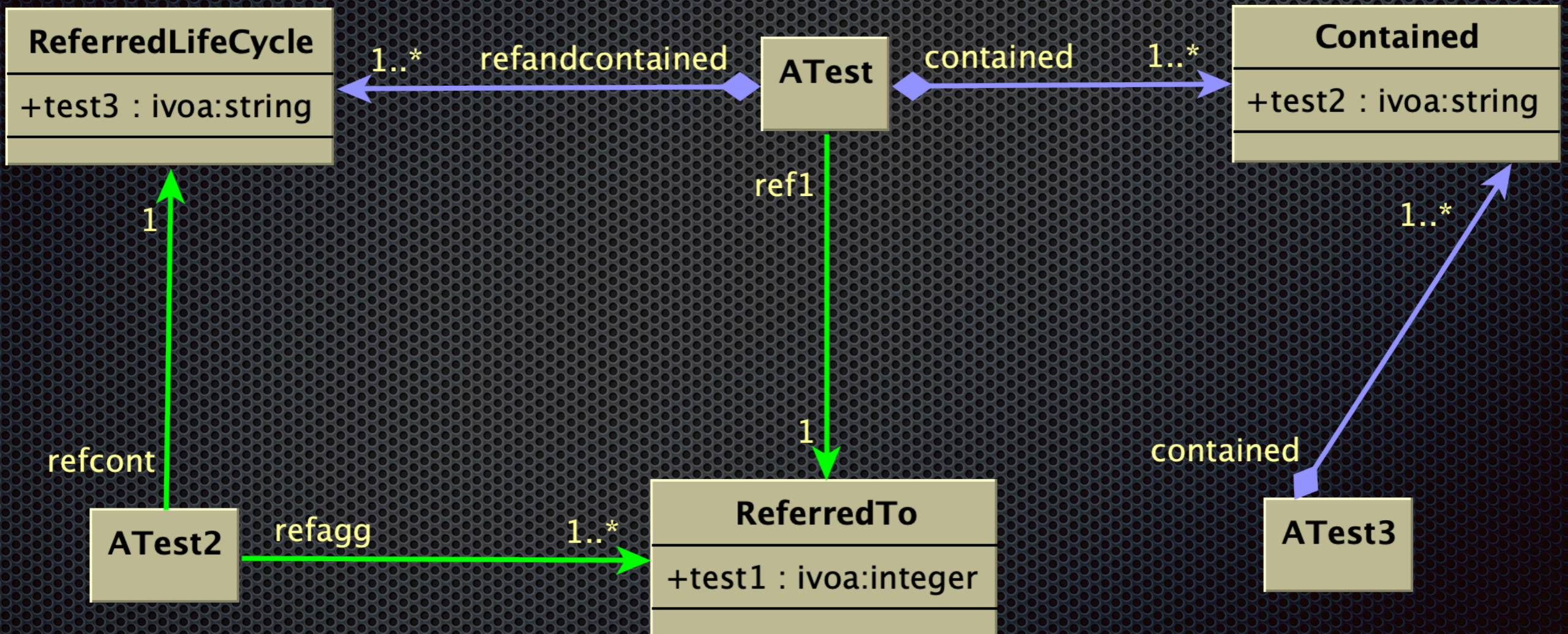
- ✦ There are some VO-DML updates that I would make - all backwards compatible
 - ✦ remove (make optional) the redundant <name> element from <import> (it was used by schematron rules)
 - ✦ I think the same is true for <documentationURL>
 - ✦ review some of the schematron rules
 - ✦ e.g. used twice in composition

VO-DML extension - Natural Keys

- ✦ ORM uses surrogate keys widely - however, in the model it is sometimes better to use a “natural key” i.e. an existing attribute - often the case for the target of “references”.

```
<xsd:complexType name="NaturalKey">
  <xsd:annotation>
    <xsd:documentation>
      This constraint is used to indicate that an attribute is a natural key for its owning ObjectType, meaning that the
      attribute value should be globally unique. This may be applied multiple times to indicate that only a composition
      of several attributes make the globally unique key.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Constraint">
      <xsd:sequence>
        <xsd:element name="Position" type="xsd:positiveInteger">
          <xsd:annotation>
            <xsd:documentation>In the case where multiple attribute values make up the natural key, this
            value indicates the ordinal number of this particular key in the compound key.</xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```


VO-DML lifecycles



TODO (volunteers?)

- Formal changes to the VO-DML standard and schema
 - Making optional some of the repeated information in VO-DML
 - the “Natural Keys” extension..
- Would be good to have an updated DM Designers’ Cookbook.
- Python code generation, ? C++...
 - would need to be able to interoperate with same serialization schema as Java code.
- Integrate in the “VOTable mapping” serialisation code.
- TODOs actually managed as usual with GitHub issues