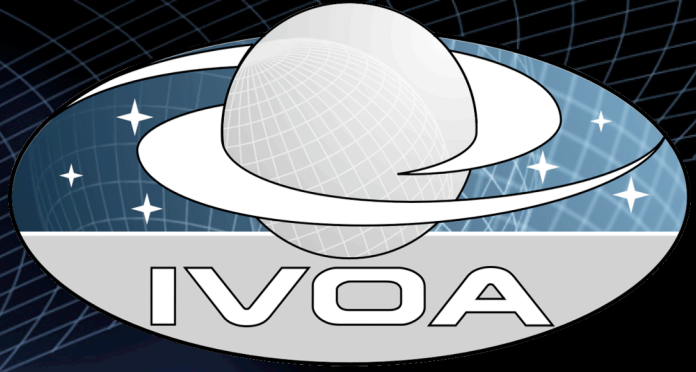
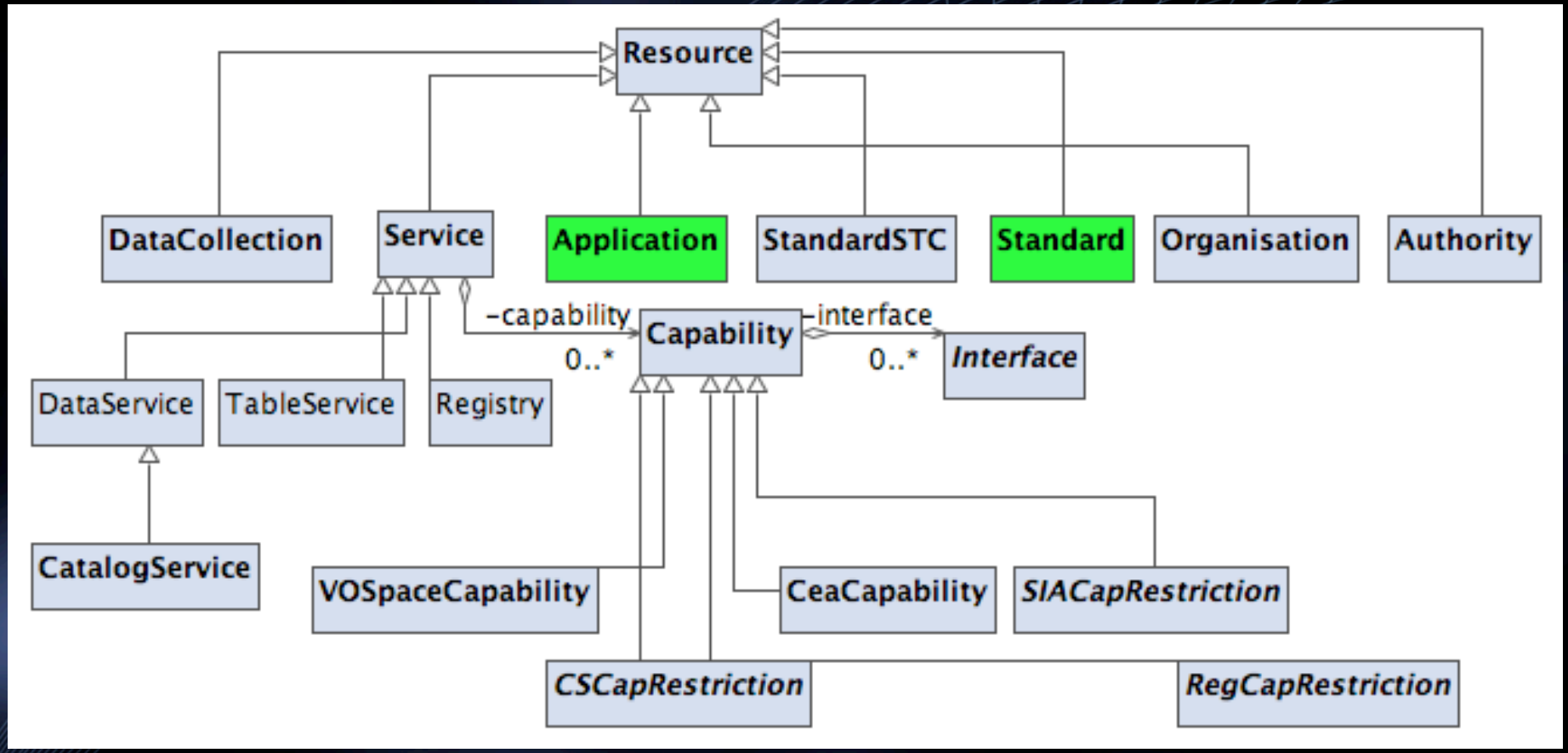


VOApplication and VOStandard Schema Progress

Paul Harrison ESO

Context



- Attempt to show Important Resource Classes
- VOApplication and VOStandard 1st level extensions

Motivation

- NOT to register every application and standard for the sake of it but…
  - Register applications that are either e.g.
    - Remotely executable as a service e.g. CEA
    - Locally able to participate in VO enabled interaction -e.g. PLASTIC
  - Register standards because
    - VOResource has many IVOA ID references to “standards” - self-consistency.
    - Some standards require extra metadata or defining instances to be created - e.g. STC, VOSpace.

STC Coordinate System

```
- <resource xsi:type="vs:StandardSTC" status="active" created="2000-01-01T09:0
  xsi:schemaLocation="http://www.ivoa.net/xml/VOResource/v1.0 VOResource-v1.0
  http://www.ivoa.net/xml/STC/stc-v1.30.xsd stc-v1.30.xsd http://www.w3.org/1999/xlir
    <validationLevel validatedBy="ivo://nvo.ncsa/registry"> 4 </validationLevel>
    <title>Standard Space-time Coordinate Systems</title>
    <shortName>STC</shortName>
    <identifier>ivo://STClib/CoordSys</identifier>
  + <curation></curation>
  + <content></content>
  - <stc:STCResourceProfile>
    - <AstroCoordSystem id="UTC-FK5-TOPO">
      - <TimeFrame>
          <TimeScale>UTC</TimeScale>
          <TOPOCENTER/>
        </TimeFrame>
      - <SpaceFrame>
        - <FK5>
            <Equinox>J2000.0</Equinox>
          </FK5>
          <TOPOCENTER/>
          <SPHERICAL coord_naxes="2"/>
        </SpaceFrame>
      </AstroCoordSystem>
    </stc:STCResourceProfile>
</resource>
```

VOSpace

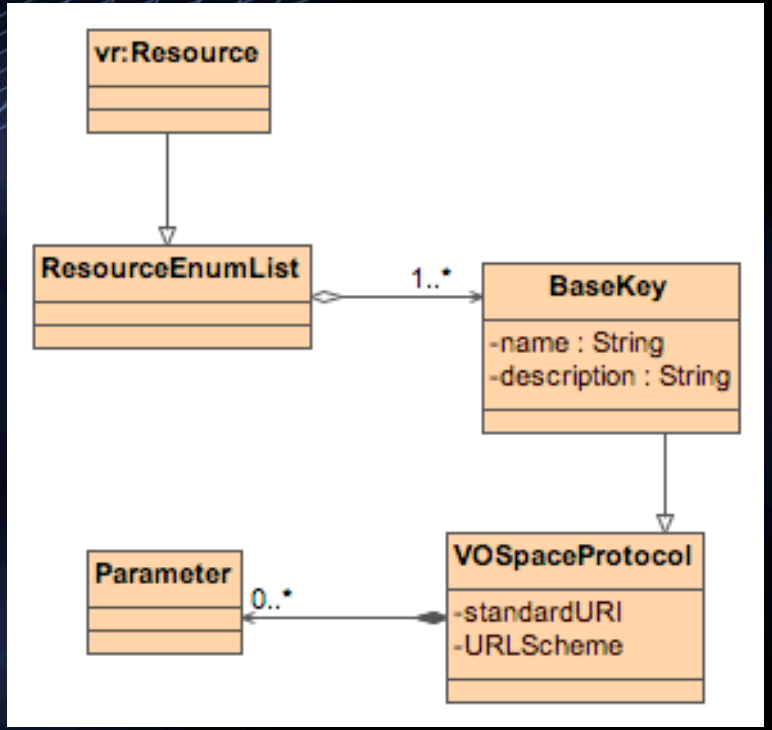
- Registry entries are essential for correct functioning of VOSpace
  - Discovery of the correct VOSpace server to invoke from the vos: URL.
  - Discovery of the capabilities of VOSpace services
    - Data transports available
    - Support for optional functionality
      - containers
      - links

Detail - schema enumerations

- Rather than encode enumerations in the interface schema, these are encoded in the registry - cf. “tagging”
  - E.g. property names, protocol names, views etc.
- Pros
  - Easier to extend - do not have to issue new version of interface schema when a new enumeration value is required - simply edit the registry entry.
  - Easier for individual implementation to publish details of ‘non-standard’ enumeration values in a way that can be semi-automatically understood - e.g. by GUI tools to display a message to user.
- Cons
  - Allowed values are not enforced directly by the interface - up to the programmer to read registry.

Detail - schema enumerations(2)

- Aim is to produce URI
- Multiple keys per registry entry
  - Only one copy of the Dublin core
  - Standard prefix
  - Use fragment # separator to indicate the enumeration key

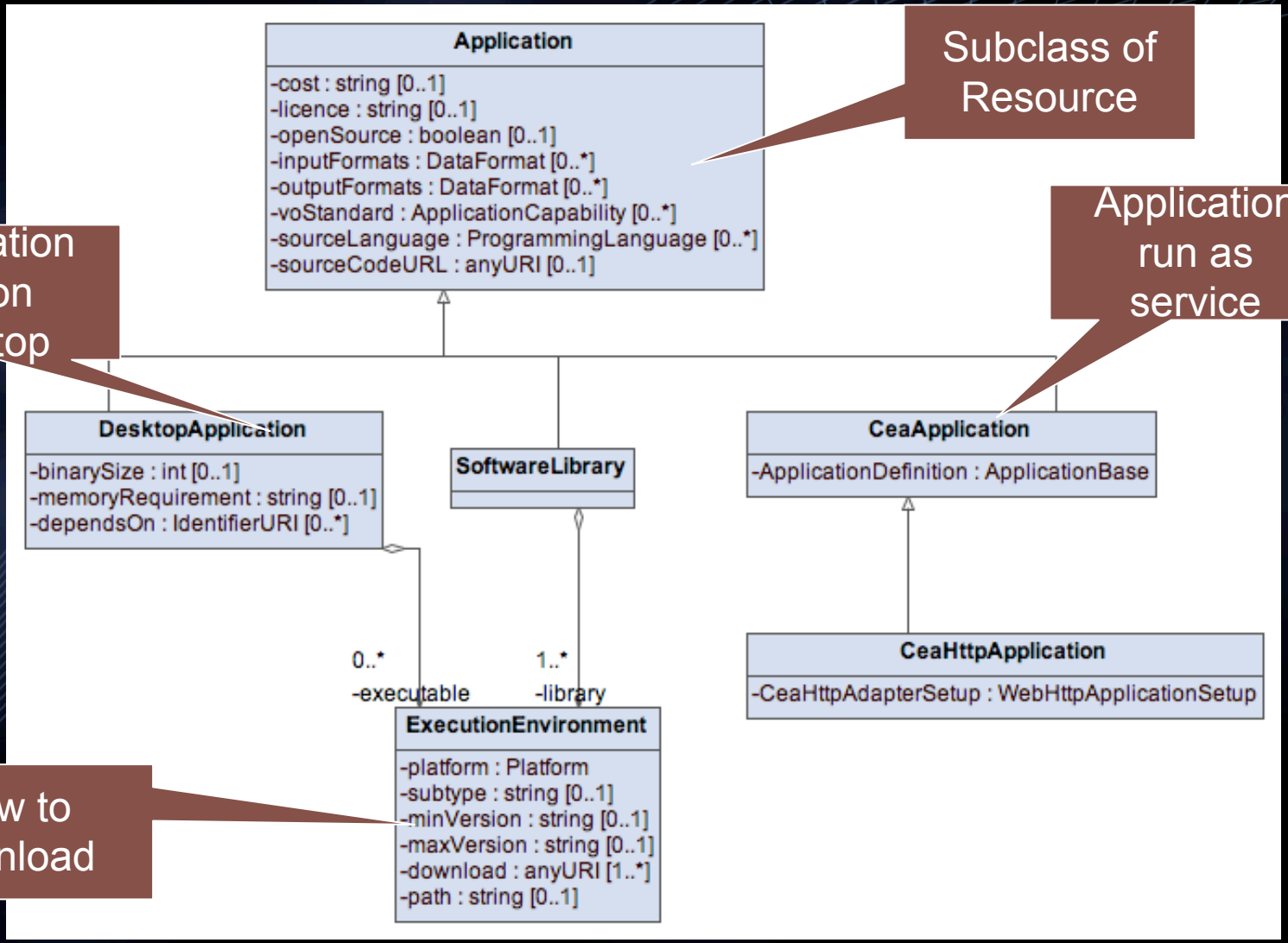


ivo://net.ivoa.vospace/protocols#http-get

Detail - schema enumerations(3)

```
<ri:Resource updated="2005-09-09T12:28:16" xsi:type="vsp:ResourceEnumList" created="200
  <title>VOSpace standard protocols</title>
  <shortName>VOSpace Protocol</shortName>
  <identifier>ivo://net.ivoa.vospace/protocols</identifier>
+ <curation></curation>
+ <content></content>
  <!-- now the actual protocol metadata -->
  <!-- needs to be completed -->
- <key id="http-1.1-get" xsi:type="vsp:VOSpaceProtocol">
    <description>http 1.1 get</description>
   - <standardUrl>
      http://www.w3.org/Protocols/rfc2616/rfc2616.html
    </standardUrl>
    <urlScheme>http:</urlScheme>
  </key>
- <key id="http-1.1-put" xsi:type="vsp:VOSpaceProtocol">
    <description>http 1.1 put</description>
   - <standardUrl>
      http://www.w3.org/Protocols/rfc2616/rfc2616.html
    </standardUrl>
    <urlScheme>http:</urlScheme>
  </key>
```

Registry Application DM



Application Metadata

- OpenSource - Yes/No
- SourceURL - where to obtain the source code.
- License - Name of license and descriptive text.
- Cost - Free; or description of pricing, if not freely available.
- SourceLanguage(s).
- Dependencies - other registered components.
- NetworkRequired -Essential/Useful/Limited/Unnecessary.
- BinarySize - Typical size of the executable.
- MemorySize - Typical memory requirements.
- VOStandards - standards that the application supports.
- InputFormats/OutputFormats - List of supported input data formats, e.g., FITS, HDF, VOTable.

Application model

- Contains just enough information to identify an application and launch automatically if possible, or simply download.
- Simple searches on a few properties possible
- “Find me an application that does X” style queries are best left to ontologies
  - There is connection with UWS, Application Messaging, Theory, and Semantics WGs. Tiger team?

Conclusions

- Schema are stable (if you agree about general model).
- New since Moscow - there is now accompanying Draft Standard documentation.
  - Wiki pages
    - http://www.ivoa.net/twiki/bin/view/IVOA/RegDMApplications
    - http://www.ivoa.net/twiki/bin/view/IVOA/RegDMStandards