

# **URI UTypes and RDF**

Norman Gray : AstroGrid, Leicester & Glasgow, UK

IVOA Interop, Trieste 2008 May 19

---

## characterisation utypes

- Defines sample types such as `cha:SpatialAxis.accuracy.sysError.ErrorRefval` – based on a path through the `char'n` model, with parameters
- “The IVOA needs to define a single and robust rule to define this concept”
- There is danger in inventing new syntax: requires parsers, validation, error recovery, libraries...
- Intended to be a unique UType for each element in a serialisation, but that’s not guaranteed. Cue UFIs?

- RDF (Resource Description Framework) is an existing (debugged, well-defined, library-supported) syntax and query language
- It's a framework for describing resources... which sounds good!
- Like UCDs, Char'n UTYPES appear to be an extended type system
- RDF is about 'explaining' what a field or column is

- RDF focuses on *properties*, which necessarily have subject and object (and types)
- Allows distinction between 'Y has calibration status Z' and 'Y has calibration status in the field named Z' (for example)
- `<#ra> cha:accuracy [ cha:ErrorRefVal [ cha:Error <#delta> ] ]`.
- Syntax is initially more complicated (more punctuation) than Char'n UTYPES, but arbitrarily extensible

- `SELECT ?r ?v WHERE { ?r phot:mag.r ?v }`
- SPARQL can query RDF generally, across multiple resources and multiple structure within them

- Here, `cha:accuracy` and `cha:ErrorRefVal` are the UTypes, and are URIs
- IVOA Note: <http://www.ivoa.net/Documents/latest/utype-uri.html> proposes systematic association between UType URIs, documentation, and lightweight semantics for interoperability

---

## other possibilities

For free: flexible description of relation between resources

```
<#e0> a eg:DarkFrame;  
  dc:created "2008-03-20T00:00:00".
```

```
<#e1> a eg:Image;  
  dc:created "2008-03-20T12:00:00";  
  eg:darkFrame <#e0>;  
  eg:errorMap <#e2>;  
  eg:qualityMap <#e3>;  
  cha:hasAxis <#a1>, <#a2>.
```

```
<#e2> a eg:ErrorMap;  
  eg:forImageData <#e1>.
```

```
<#a1> a cha:SpatialAxis; ...
```