VOTable Future

Mark Taylor (Bristol)

IVOA Interop Meeting
NCSA Urbana IL

21 May 2012

`$Id: votable-plenary.tex,v 1.8 2012/05/21 04:30:55 mbt Exp $`

Outline

- Issue: NULL representation
- History
- Suggested solutions
  - TABLEDATA
  - BINARY2
- Discussion: requirements and opinions
- Decisions and way forward

Issue: NULL representation

Representing NULLs when writing VOTables is a bit tricky.

- Summary:
  - ▷ **Integer columns:** “magic” bad value must be specified up front
    `<FIELD datatype="short" name="COUNT"><VALUES null="-32768"/></FIELD>`
  - ▷ **Float columns:** NULL and NaN not distinguished
  - ▷ **Array columns:** NULL and empty array not distinguished

- Context:
  - ▷ It’s a subtle point
    - ○ Capability of VOTable to represent NULL values is not in question
    - ○ Existing numeric model has served VOTable for 8+ years, FITS BINTABLE for 20+ years
  - ▷ It has come up now because of TAP: streaming query result to VOTable
    - ○ Want to represent DB content model accurately (preserve type, NULL≠NaN)
    - ○ Don’t know in advance what integer values are not used in data
  - ▷ Any changes should be cautious
    - ○ VOTable is widely used — backward compatibility is important
    - ○ VOTable WG is dormant
  - ▷ Boring but (maybe?) important

Interested Parties

- TAP service implementors (and other VOTable producers)
- TAP client implementors (and other VOTable consumers)
- VOTable toolkit implementors

  . . . and their users

History

- Summer 2011
  - Raised on DAL list + VOTableIssues wiki page (initially by Tom McGlynn)
- Pune Interop Oct 2011
  - Raised at TCG
  - Special Apps splinter session scheduled
  - Some discussion and provisional conclusions, not very wide participation
- Post-Pune Oct 2011
  - Posted summary and call for comments on Interop list
  - ... no response
- Euro-VO May 2012:
  - Discussions, more thought → suggestions
- This week May 2012:
  - Consideration in TCG meeting Sunday

VOTable DATA Encoding Refresher

VOTable has three alternative data encoding mechanisms:

- TABLEDATA *(widely used)*:

```
<DATA>
  <TABLEDATA>
    <TR> <TD>M51</TD> <TD>202.43</TD> <TD>47.22</TD> </TR>
    <TR> <TD>M97</TD> <TD>168.63</TD> <TD>55.03</TD> </TR>
  </TABLEDATA>
</DATA>
```

- BINARY *(not so much used)*:

```
<DATA>
  <BINARY>
    <STREAM encoding="base64">
       TTUxAAAAAAAAAEBpTcKPXCj2QEecKPXCj1xNOTcAAAAAAAAAQGUUKPXCj1xAS4PX
       Cj1wpA==
    </STREAM>
  </BINARY>
</DATA>
```

- FITS *(hardly ever used?)*:

```
<DATA>
  <FITS>
    <STREAM href="fcat-2.fits"/>
  </FITS>
</DATA>
```

These encode exactly the same data

Suggested Fix: TABLEDATA

New rule: Empty `<TD/>` element means NULL

- Details:
  - ▷ Integer columns: empty TD no longer illegal, no magic value required
  - ▷ Float columns: empty TD no longer means NaN
  - ▷ Array/string columns: empty TD no longer means empty array/string
- Impact: *Low*
  - ▷ Integer columns: many VOTable producers already (illegally) use this convention, most VOTable consumers already accept it to mean NULL *(any counterexamples?)*
  - ▷ Float columns: little or no VOTable-based software relies on NULL/NaN distinction
  - ▷ Array/string columns: most VOTable software already conflates NULL/empty arrays
  - ▷ VOTable data becomes dependent on encoding (conversion may not be lossless)

  ⇒ **Little or no code change required**

Suggested Fix: BINARY2

New encoding BINARY2

- Details:
  - ▷ Like BINARY, but with a per-cell bitmask indicating NULL-ness
- Impact: *Medium*
  - ▷ New code required in VOTable clients/servers/libraries to benefit
  - ▷ Additional machinery required for negotiating VOTable output from services
    - ○ Must ensure that old clients don't receive BINARY2 VOTables
    - ○ Add "`serialization`" parameter to `application/x-votable+xml` MIME type
  - ▷ Will introduce VOTable files which older software is unable to understand

Options

0. Do nothing

- Problems remain for TAP service implementors & other VOTable producers
- Existing illegal usages (empty `<TD/>` for integer) continue
- Life goes on.

1. Adopt TABLEDATA fix only

- New VOTable 1.3, minor changes of wording
- TAP service implementors can write TABLEDATA (only) output easily
- Data content capability of TABLEDATA and BINARY no longer equal (no roundtrip)

2. Adopt TABLEDATA fix and new BINARY2 encoding

- New VOTable 1.3, some significant (though localised) changes
- Changes to MIME type parameterisation required
- TAP service implementors can write TABLEDATA & BINARY-like output easily
- Faithful representation of RDBMS state in VOTable serialization
- More work for client/server/library implementors
- Benefits only seen when clients and servers upgrade appropriately
- BINARY2 tables incomprehensible to older software

Questions

- Is the existing NULL representation a problem that needs fixing?
- Do any VOTable consumers forsee trouble from `<TD/>`=NULL rule?
  - newly legal empty integer TDs
  - adjusted semantics for empty float/array/string TDs
- Do people actually want to use a BINARY-like VOTable encoding?
  - data providers (esp. TAP service implementors/deployers)?
  - data consumers (esp. TAP clients)?
  - Would compressed TABLEDATA be a better solution?

    (BINARY: 1; TABLEDATA: 2.1±1.1; gzip-BINARY: 0.37±.17; gzip-TABLEDATA 0.32±.20)
- Should BINARY be deprecated altogether?
  - No longer contains equivalent data to TABLEDATA
  - JavaScript clients tend to require TABLEDATA (treat VOTable as DOM)