DFDL WG Call Agenda

This OPEN document will not be filed. It is being kept active.

Meeting about Meetings\OGF

<table>
<thead>
<tr>
<th>Project</th>
<th>DFDL 1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Meeting Date</td>
<td>16-Apr-20 (Thurs)</td>
</tr>
<tr>
<td>Meeting Time</td>
<td>16:00 - 17:00</td>
</tr>
</tbody>
</table>

Created by Steve Hanson on 09-Mar-11
Last Modified by Steve Hanson on 15-Apr-20

OGF DFDL Working Group Call, 16th April 2020

Agenda

Prepare for your meeting by describing the objectives (both immediate and long-term, if appropriate) of the meeting; and describing planning details.

1. Daffodil Open Source Project
   Status update

2. Behavior of \texttt{nilKind literalValue} with respect to \texttt{binaryNumberRep of packed}
   See email thread. Any reason why \texttt{dfdl:}\texttt{nilLiteralCharacter} is not sufficient for this scenario?

3. Clarification: Unparser choice branch selection (section 15.1.3)
   See email thread. Believed addressed by erratum 5.60.

4. AOB

Minutes

Meeting Minutes

Reflect on your meeting as you record all topics and issues discussed, and any tabled conversations. What went well, or what would you do differently next time? Document those so others can take advantage of your learning.

Attendees

Apologies

Minutes

IPR Statement
“I acknowledge that participation in this meeting is subject to the OGF Intellectual Property Policy.”

Meeting closed

Next regular call
30th April 2020 @ 16:00 UK
Create Action Items

Record the to-do's and individuals assigned by entering the appropriate information in the form below. Press the "Create Action Items" button to create specific to-do's that can be tracked in the assignee's Work for Me views. All Action Items will be tracked in the Action Items and Other Meeting Documents tab.

Action Items and Other Meeting Documents

<table>
<thead>
<tr>
<th>Subject</th>
<th>Document Type</th>
<th>Created</th>
<th>Modified</th>
</tr>
</thead>
</table>

Next action: 317

Actions raised at this meeting

<table>
<thead>
<tr>
<th>No</th>
<th>Action</th>
</tr>
</thead>
</table>

Current Actions:

<table>
<thead>
<tr>
<th>No</th>
<th>Action</th>
</tr>
</thead>
</table>
| 250 | **Standardise on a single tdm1 format for DFDL tests (All)**  
5/2/14: Steve has requested permission for IBM to view / use the Daffodil tdm1 files, as a precursor to trying to standardise on a common tdm1 format. Was formerly part of action 066.  
...  
18/2: No further progress  
11/3: Mike and Steve discussing the best way to share and cooperate on tdm1 format.  
25/3: Discussed the creation of an OGF document that will own and define a standardised tdm1 format.  
11/4: Proposal is for the OGF document to define a tdm1 format without Tresys or IBM copyright statement.  
15/4: Draft document on Redmine |
6/5: No further progress

20/5: Mark has read through the document. Particularly concerned with how
namespaces are handled in the infoset.

... 

17/6: No further progress

25/6: Mike has added bit order capability as per action 233.

... 

9/12: No further progress

6/1/15: Mike to resubmit this as Tresys would like to run their tdml suite against both
Daffodil and IBM DF DL.

... 

10/2: No further progress

24/2: Mike updating the Daffodil TDML test runner to handle unparsers (ie, serializer)
tests

... 

14/4: No further progress

28/4: Tresys have enhanced their tdml runner to allow unparsers tests and round-trip
tests (parser->unparser->parser) as well as the new tutorial tag (see action 228)

12/5: Not discussed 

... 

3/11: No progress

5/1/16: No progress. Needs more interoperability between implementations to be really
useful.

... 

25/7/17: No further progress

3/10: No further progress although forthcoming work to add packed/zoned numbers
may force https://redmine.org/issues/339 progress

... 

11/12: Expected to look at this in the next month or so

... 

4/9/18: No further progress

16/10: Mike has started work on a TDML runner that can drive a pluggable DF DL
implementation, in support of interoperability testing, including IBM DF DL.

1/11: Pluggable TDML runner working. On Github at
achieved using its schema resolver feature and pointing it at Daffodil's resolver. IBM
DFDL sample uses mark() on its input stream but IBM believes this is not necessary.

15/11: IBM DFDL and Daffodil have dependencies on different releases of ICU. Forcing
changes to the TDML runner to isolate the implementations under test.

29/11: Good progress on the TDML runner, see email from Mike. The ibmCrossTestRig
is not part of Daffodil (because it links against IBM DF DL), but is open source Apache
License v2, and is currently in review at
https://github.com/OpenDFDL/ibmDFDLCrossTester/pull/1. Steve needs to talk to IBM
legal to check this is ok as it currently modifies IBM DF DL sample code.

10/1/19: Daffodil have removed the modified IBM DF DL sample code. Steve to ask
whether IBM can donate tests from the existing IBM DF DL test suite.

7/12: If permitted, the tests from IBM can be used to see how the IBM and Daffodil tdmls
have diverged.

5/4: IBM are permitted to send Daffodil some example tdmls. Steve to send some to
Mike for next call.

... 

31/5: No progress

27/6: Steve has sent example TDML files to Mike, under the existing IBM Grant of
Copyright agreement with Apache

11/7: Daffodil to adapt their test runner so that it works with the IBM TDML file variant.
Action can be closed when this is done.

...
<table>
<thead>
<tr>
<th>Date</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>29/8</td>
<td>No further progress</td>
</tr>
<tr>
<td>17/10</td>
<td>Noted that this action had its number accidentally swapped with 242 in October 2016. Renumbered back to 250.</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>9/1/20</td>
<td>No further progress</td>
</tr>
<tr>
<td>289</td>
<td><strong>Unparsing: expression refers backwards to outputValueCalc which refers beyond it.</strong></td>
</tr>
<tr>
<td>2/8/16</td>
<td>Need to decide if this is allowed and if so if there are any restrictions.</td>
</tr>
<tr>
<td>13/9</td>
<td>Motivating scenario is where a variable is being set to a length element using dfdl:setVariable, which on unparse is set using dfdl:outputValueCalc. So although the variable is referring backwards to the length element, it is effectively forward referencing so must block. Mike believes this is unavoidable.</td>
</tr>
<tr>
<td>11/10</td>
<td>Daffodil has implemented this. Mike to provide scenario.</td>
</tr>
<tr>
<td>8/11</td>
<td>Mike couldn't find example, will continue to look</td>
</tr>
<tr>
<td>10/1/17</td>
<td>Mike has realised that all the examples were reworked to avoid using variables, hence why can't be found.</td>
</tr>
<tr>
<td>7/2</td>
<td>Daffodil will soon be implementing dfdl:newVariableInstance which will bring this up again.</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>17/4/18</td>
<td>Waiting for Daffodil to implement dfdl:newVariableInstance</td>
</tr>
<tr>
<td>15/5</td>
<td>Daffodil team have supplied an example of this from the PCAP schema. Likely to require a flag on newVariableInstance (or maybe variable declaration) to indicate whether needed on parse, unparse or both.</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>17/10/19</td>
<td>No further progress</td>
</tr>
<tr>
<td>12/12</td>
<td>Daffodil starting to implement dfdl:newVariableInstance</td>
</tr>
<tr>
<td>9/1/20</td>
<td>No further progress</td>
</tr>
<tr>
<td>309</td>
<td><strong>Create example scenarios to illustrate offset &amp; pointer requirements (Bradd)</strong></td>
</tr>
<tr>
<td>5/4/19</td>
<td>Daffodil have a draft proposal for offset support, TPF have experimental implementation for pointer support. Need examples to show the requirement, especially unparsing.</td>
</tr>
<tr>
<td>2/5</td>
<td>Bradd supplied an example of pointers. On parsing the pointer is used as an absolute address to a piece of accessible memory, and the element is parsed from that location. On unparsing memory is allocated and unparsing of the element occurs into that location and the pointer set to the location (memory allocation is implementation-defined). Note the pointer value does <em>not</em> appear in the infoset. Looks like a useful and workable addition to DFDL. Could solve the parsing requirements for TIFF image files. Bradd also has extension for offset, which is like pointer but uses relative location instead of absolute. Both are examples of indirection. A further example could be specifying a file to read. Contrast this with what DFDL has used the term ‘offset’ for in the past, namely as an alternative property to alignment/skip which allows the parser/unparser to jump directly to a point in the current buffer. These are orthogonal concepts. Noted that parsing of ZIP files may need both. Secure implementations may need to disallow use of pointers and/or offsets unless they can guarantee to fill everywhere with the fill byte. Implementations should also be deterministic. Agreed that recursion not needed to implement this. Bradd mentioned a further concept ‘overflows’, an example being an array unparsed into a linked list. Pointers proposal needs to be written up as an experimental feature.</td>
</tr>
<tr>
<td>31/5</td>
<td>Bradd to write up pointers proposal as an experimental feature.</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>11/7</td>
<td>No update</td>
</tr>
<tr>
<td>8/8</td>
<td>Bradd aiming to get this written up for next time. Also needs issue tracker raising.</td>
</tr>
<tr>
<td>29/8</td>
<td>Bradd unable to make the call.</td>
</tr>
<tr>
<td>17/10</td>
<td>Written up for review and sent to WG but not as an experimental feature document. Mike also noted <a href="http://www.binarydom.com/sdk/doc/bdrl.shtml">http://www.binarydom.com/sdk/doc/bdrl.shtml</a>. Mike has reviewed and commented on the write-up, Steve needs to do the same, then send back to Bradd. Main discussion was around unparsing, eg, buffering implications,</td>
</tr>
</tbody>
</table>
whether to try and format exactly or canonically. In parallel, Bradd to create an
experimental feature document (see table below).

12/12: Bradd sent an updated document. WG will review for next meeting. There was
some discussion about the use in the document of the term ‘empty’ and whether that
really meant ‘missing’. This led to an in-depth discussion about the different use cases
for default values, it is likely that DFDL 2.0 will introduce support for some of these,
specifically:
- Item exists in Infoset with default value, so unparsable empty rep (the mirror of parsing,
as practised by GPB)
- Item exists in data with default value, so remove from Infoset post-validation (the
  mirror of unparsing, and a requirement from z/TPF who have a post-parse option to do this)

9/1/20: No progress, still needs reviewing.

<table>
<thead>
<tr>
<th>311</th>
<th>Move DFDL 1.0 spec to Grid Recommendation as per GFDL.152 (All)</th>
</tr>
</thead>
</table>
| 27/6/19: Steve to create experience document for IBM DFDL usage. Mike to do same
  for Daffodil. Steve to reach out to the DFDL4S team at ESA to do the same.
  Mike to send email to OGF to inform them of the WG’s intent and to establish who to
  deal with going forward. |
| 11/7: Mike has sent email to Steve for review. |
| 8/8: Mike emailed OGF and Alan Sill has replied. He has forwarded the email to the
  Grid Forum Steering Committee with a request for responses and asked Jens Jensen
  as VP for Standards to help the WG work through the process. Agreed that a draft
  should be created that fixes all the known typographical bugs (as noted in tracker 233)
  - Mike will do this. Mike will follow-up with Jens Jensen also. |
| 29/8: DFDL WG needs to find an External Reviewer as there is no Data Area chair
  currently. Needs to be independent from IBM and Tresys. Also, need to establish
  whether the existing document can be edited or a new document is needed - the latter
  would entail a 6 month delay. Need to get the current errata experience document
  updated with the set of trackers that we need in 1.0. |
| 17/10: External reviewer is to be Martin Westhead, who was a founding member of
  the DFDL WG. Assumption is that we can use the existing spec and add the errata to it,
  although the number of errata is large. Mike is working through the issue trackers,
  adding them to the errata document at
  https://redmine.org/dmsf_files/13384?download=, and updating the spec if they can
  simply be added in without further discussion or approval. Several require approval
  and Steve has reviewed those. Mike has responded to Steve’s comments with further
  changes, ready for another review cycle. One erratum requiring more thought is 5.39
  (issue 299) as it introduces the new concept of a choice branch with zero occurrences.
| 12/12: Steve has reviewed latest errata, no changes required. Note 5.39 is correct as
  written, no more discussion needed. Mike has incorporated the errata into the DFDL
  spec. Steve has reviewed the spec, and Mike has further updated it. Current version at
  https://redmine.org/dmsf_files/13601?download=. Martin Westhead thinks he can
  review by end of year. Mike will create an html version of the current version, marked
  as a draft, for users for convenience, and will investigate best place to host it. |
| 9/1/20: The html version of the updated spec is available, and Martin Westhead has
  started to review the spec and is providing comments. He has concerns about the first
  few sections of the spec, particularly with regard to concepts being mentioned before
  explanation, so requiring multiple passes to understand. Mike considering the best way
  to fix this. Martin is much happier with the later sections. Mike has found a few more
  minor typographical errors and is fixing |

<table>
<thead>
<tr>
<th>313</th>
<th>Investigate dfd:textNumberCheckPolicy='lax' and plus sign behaviour (All)</th>
</tr>
</thead>
</table>
| 29/8/19: IBM DFDL and Daffodil look like they behave differently. Spec suggests IBM
  DFDL is correct. Yet both use ICU under the covers! More investigation needed. |
| 17/10: Looks like ICU changed behaviour in a recent release, thought to be 61 around
  Dec 2017. Daffodil is picking up a later release. IBM DFDL is back on 51.2. We can’t
  have this moving around, so the spec needs to pin down the ICU release(s) that are
  eligible for implementations to use. Daffodil will change to pick up an earlier release. |
Potential issue going forward if we have to move to a later ICU for security or bug-fix reasons. Steve to raise a ticket against ICU to (provide an option to) revert the behaviour.

12/12: ICU issue raised - https://unicode-org.atlassian.net/browse/ICU-20896, Response from ICU is that the change was deliberate, is working as designed, and users should not rely on a precise specification of 'lax' as it is intended to be tolerant for handling user input. Discussed, and the feeling in the WG is to try and weaken the definition of 'lax' so that it becomes implementation-dependent. Steve will look at this further and propose an erratum.

9/1/20: No further progress.

### 315
Determine the behaviour of fn:count() and fn:exists() (All)
17/10/19: Not clear from the spec. Need to consider existing concept of 'knownToExist' and the existing rules about when expressions are evaluated.

... 9/1/20: No further progress.

### 316
 Proposed new experimental feature 'User defined functions' (Mike)
9/1/20: Version 2.5.0. of Daffodil will include this, needs writing up. Does XPath 2.0 provide for this?

---

### Closed actions

<table>
<thead>
<tr>
<th>No</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Deferred actions

<table>
<thead>
<tr>
<th>No</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Work items:

<table>
<thead>
<tr>
<th>No</th>
<th>Item</th>
<th>Owner</th>
<th>Target</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Experimental Features:

<table>
<thead>
<tr>
<th>No</th>
<th>Title</th>
<th>Original Action</th>
<th>Issue Tracker</th>
<th>Implementor</th>
<th>Experience Document</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Elements of type xs:hexBinary' with lengthUnits 'bits'</td>
<td>292</td>
<td><a href="https://redmine.ogf.org/issues/344">https://redmine.ogf.org/issues/344</a></td>
<td>Daffodil</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Variable Path Step in DFDL expression</td>
<td>287</td>
<td><a href="https://redmine.ogf.org/issues/353">https://redmine.ogf.org/issues/353</a></td>
<td>DFDL4S</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>BLOBs as URIs in infoset</td>
<td>312</td>
<td><a href="https://redmine.ogf.org/issues/356">https://redmine.ogf.org/issues/356</a></td>
<td>Daffodil</td>
<td>In progress</td>
</tr>
<tr>
<td>5</td>
<td>Data streaming layers</td>
<td>304</td>
<td><a href="https://redmine.ogf.org/issues/341">https://redmine.ogf.org/issues/341</a></td>
<td>Daffodil</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Interoperability</td>
<td>307</td>
<td>N/A</td>
<td>Daffodil</td>
<td><a href="https://redmine.ogf.org/dmsf">https://redmine.ogf.org/dmsf</a>_</td>
</tr>
<tr>
<td>---</td>
<td>----------------------------</td>
<td>-----</td>
<td>----------------------------------------------------------------</td>
<td>--</td>
<td>--------------------------</td>
</tr>
<tr>
<td>8</td>
<td>Integer enums as strings in infoSet</td>
<td>294</td>
<td><a href="https://redmine.ogf.org/issues/354">https://redmine.ogf.org/issues/354</a></td>
<td>Daffodil</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Bi-directional support (moved out of DFDL 1.0)</td>
<td>241</td>
<td><a href="https://redmine.ogf.org/issues/357">https://redmine.ogf.org/issues/357</a></td>
<td>TBD</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Indirection using pointers</td>
<td>309</td>
<td>TBC</td>
<td>IBM DFDL (z/TPF)</td>
<td>In progress</td>
</tr>
<tr>
<td>11</td>
<td>User-defined functions</td>
<td>316</td>
<td>TBC</td>
<td>Daffodil</td>
<td></td>
</tr>
</tbody>
</table>