

# DFDL WG Call Minutes

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

## Meeting about Meetings\OGF

Project	DFDL 1.0
Meeting Date	14-Jul-22 (Thurs)
Meeting Time	16:00 - 17:00

Created by Steve Hanson on 09-Mar-11

Last Modified by Steve Hanson on 14-Jul-22

## OGF DFDL Working Group Call, 14th July 2022

### Agenda

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

#### 1. Implementations Update

#### 2. dfdl:lengthKind 'pattern' usability - no match does *\*not\** cause a parse error

See email to WG from Mike, Apache Daffodil users find this unintuitive.

#### 3. Need a default dfdl:choiceDispatchKey branch?

See email to WG from Mike asking for this but Steve thinks it might not be necessary. Spec section 15.1.2.

#### 4. DFDL needs more binary number representations

See email to WG from Mike, in summary:

- Clarify the wording that restricts dfdl:binaryDecimalVirtualPoint to xs:decimal and user restrictions
- Add several further reps seen in practice as experimental features (somehow)

#### 5. 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

Steve Hanson  
Mike Beckerle

#### Apologies

#### Minutes

## 1. Implementations Update

Apache Daffodil 3.4.0 in development.

## 2. dfdl:lengthKind 'pattern' usability - no match does *\*not\** cause a parse error

See email to WG from Mike, Apache Daffodil users find this unintuitive.

However it is working exactly as designed. Leave alone for DFDL 1.0.

## 3. Need a default dfdl:choiceDispatchKey branch?

See email to WG from Mike. There is a nested choice use case where there is a need to distinguish a bad choice branch key from a good choice branch key that then fails to parse. To do this requires use of extra discriminators that make the schema unwieldy. Steve to try and reproduce.

**New action 327** raised.

## 4. DFDL needs more binary number representations

See email to WG from Mike, in summary:

a) Clarify the wording that restricts dfdl:binaryDecimalVirtualPoint to xs:decimal and user restrictions

b) Add several further reps seen in practice as experimental features

c) Extend the experimental feature syntax to allow extra enums on existing properties.

**New actions 328 & 329** raised.

## IPR Statement

"I acknowledge that participation in this meeting is subject to the OGF Intellectual Property Policy."

## Meeting closed

17:10 UK

## Next regular call

11th August 2022 @ 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

## Action Items and Other Meeting Documents

Subject	Document Type	Created	Mod

Next action: **330**

### Actions raised at this meeting

No	Action
<b>327</b>	<b>Do we need a default dfdl:choiceDispatchKey branch? (Steve)</b> 14/7/22: To handle a nested choice use case where there is a need to distinguish a bad choice branch key from a good choice branch key that then fails to parse its branch. Steve to reproduce the use case to see if IBM DFDL has a similar problem.
<b>328</b>	<b>Extend the experimental feature syntax to allow extra enums on existing properties (Mike)</b> 14/7/22: The experimental feature document does not cater for additional enums to existing properties. Action 329 is an example of where this would be useful.
<b>329</b>	<b>DFDL 2.0 candidate: Additional binary number reps (Mike)</b> 14/7/22: Apache Daffodil users are encountering a bunch of unsupported binary number formats. Apache Daffodil would like add support for these into DFDL 1.0 as an experimental feature ahead of DFDL 2.0. To do so requires action 328 to extend the experimental syntax to permit this.

### Current Actions:

No	Action
<b>289</b>	<b>Unparsing: expression refers backwards to outputValueCalc which refers beyond it (Mike).</b> 2/8/16: Need to decide if this is allowed and if so if there are any restrictions. 13/9: 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. 11/10: Daffodil has implemented this, Mike to provide scenario. 8/11: Mike couldn't find example, will continue to look

	<p>10/1/17: Mike has realised that all the examples were reworked to avoid using variables, hence why can't be found.</p> <p>7/2: Daffodil will soon be implementing dfdl:newVariableInstance which will bring this up again.</p> <p>...</p> <p>17/4/18: Waiting for Daffodil to implement dfdl:newVariableInstance</p> <p>15/5: 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.</p> <p>...</p> <p>17/10/19: No further progress</p> <p>12/12: Daffodil starting to implement dfdl:newVariableInstance</p> <p>...</p> <p>16/4/20: No further progress</p> <p>30/4: Daffodil has implemented newVariableInstance so now in a position to look at this again.</p> <p>9/7: No update</p> <p>23/7: Daffodil has implemented newVariableInstance so Mike will look at this again</p> <p>3/9: Daffodil team looking at this and related issues surrounding expression evaluation when unparsing.</p> <p>...</p> <p>29/10: No further progress</p> <p>12/11: Mike has a scenario in the PCAP schema that uses several outputValueCalc. To simplify the expressions, he would like to use a variable with a value set by newVariableInstance. The problem is that the rules for variables are different from those for outputValueCalc, specifically variables can not refer forwards (because they are evaluated on both parse and unparse) whereas that is the point of outputValueCalc. To ease this would require a change to the spec. WG agreed this is not necessary for DFDL 1.0 but a candidate for 2.0.</p> <p>10/12: Daffodil to implement as an experimental feature. Added to table below. GitHub issue <a href="https://github.com/OpenGridForum/DFDL/issues/22">https://github.com/OpenGridForum/DFDL/issues/22</a>.</p> <p>7/1/21: Daffodil have started prototype work on this.</p> <p>18/2: Work continuing, but it looks like forward reference variables are needed as anticipated.</p> <p>18/3: Daffodil has implementation of this nearly complete, which Mike will walk through on the next call.</p> <p>15/4: Mike walked through examples of dfdl:newVariableInstance which refer forward and use 'dfdlx:direction=unparseOnly', using the PCAP schema. Will be part of the experience document.</p> <p>8/7: Material assembled for experience document.</p> <p>14/10: Mike to write this up</p> <p>...</p> <p><a href="#">14/7/2022: Write up still in progress</a></p>
322	<p><b>DFDL 2.0 candidate: Handle embedded XML and JSON in a natural way (Mike)</b></p> <p>8/7/2021: This was something that was included in the first DFDL 1.0 spec drafts but dropped.</p> <p>14/10: Mike has created GitHub issue <a href="https://github.com/OpenGridForum/DFDL/issues/27">https://github.com/OpenGridForum/DFDL/issues/27</a>.</p> <p>24/2/2022: Create an experimental feature document</p> <p><a href="#">14/7: Waiting for prototype implementation</a></p>
324	<p><b>Erratum: Clarify semantic of array element assert containing subexpression ".[1]" (All)</b></p> <p>8/7/2021: Erratum needed to state that ".[1]" is equivalent to "../a[1]" and "../a[dfdl:occursIndex()]"</p> <p>14/10: Mike has created GitHub issue</p>

	<p><a href="https://github.com/OpenGridForum/DFDL/issues/26">https://github.com/OpenGridForum/DFDL/issues/26</a>.</p> <p>24/2/2022: Mike to look back through emails for more information.</p> <p>14/7: <a href="#">No further progress</a></p>
325	<p><b>Submit OGF DFDL 1.0 to ISO for adoption as an ISO standard (Mike, Steve)</b></p> <p>11/11: Alan &amp; Wolfgang attended the call. OGF is looking to re-home its successful standards to more relevant organisations. The proposal is that DFDL moves to ISO. OGF has well-established connections with ISO, and Wolfgang is an attendee at working group meetings. There is a specific ISO process that lets them adopt a public, freely available standard whilst maintaining its public, free status. In contrast to ISO-owned standards which are chargeable. The public, free status continues to apply to ongoing revisions. DFDL feels like a good fit for ISO, noted that DFDL schemas exist for ISO 8583 and EDIFACT. There is no need initially to change DFDL to use ISO templates, this can happen later. Noted that this is not a process for dumping old standards onto ISO, it applies to standards with an active community. The end result is that DFDL continues to evolve and be freely available but there are revisions that are effectively 'blessed' by ISO. An example of this in practice is Open Virtualization Format Specification from DMTF. Adoption of a standard by ISO is ultimately determined by national body votes.</p> <p>Next steps: OGF will write to ISO-IEC JST1 with the proposal. DFDL WG will reach out to the chair of SC38 WG5 which is considered the most likely place DFDL will end up in ISO. The WG5 chair is Fernando Gebara Filho</p> <p>DFDL WG to consider attending the appropriate ISO meetings.</p> <p>Useful links:</p> <p>ISO Process:</p> <p><a href="https://www.iec.ch/members_experts/refdocs/iec/Consolidated_JTC1_Supplement_2020_publication.pdf">https://www.iec.ch/members_experts/refdocs/iec/Consolidated_JTC1_Supplement_2020_publication.pdf</a></p> <p>DMTF ANSI/ISO adoptions: <a href="https://www.dmtf.org/about/register/apresources">https://www.dmtf.org/about/register/apresources</a></p> <p>24/2/2022: OGF are still working on the first step of the process, the application for OGF becoming a PAS submitter. They expect to submit the application letter begin of March. Once OGF will be a recognised PAS submitter there will be an assigned mentor who will guide through the next steps which mostly will have to be taken by the DFDL WG. Wolfgang sent an email with questions that the WG need to answer. Questions discussed and answers sent back to Wolfgang.</p> <p>7/4: Steve to chase OGF for latest position.</p> <p>14/7: <a href="#">From Wolfgang: "The process was a bit slower than foreseen, mostly because of holiday absence both on side the of OGF and JTC1. Regarding the application letter I am waiting for the confirmation from our guide that the last version we have sent him on Monday is well suited for our official application as PAS submitter. If we have his confirmation OGF will officially send the application to JTC1. Thereafter it will subject to a voting of the national bodies which my take additional one or two months." Meantime in parallel Steve will contact the WG5 chair.</a></p>
326	<p><b>Rules for when an assert or discriminator is used on a prefixLengthType simple type (All)</b></p> <p>24/2/2022: Mike and Steve believe this is allowed but the meaning of relative path components need to be clarified. IBM DFDL does not support asserts on global simple types so unable to test if it works. Apache Daffodil ignores them. A good way to look at the prefixLengthType is as a quasi-element, so assert feels like it should be allowed. If so need to decide if the expression can use absolute paths, relative paths other than '.' and variables.</p> <p>7/4: Struggling to think of a use case where the expression is other than '.', meaning the quasi-element. This is a very restricted use, and suitable for an erratum on DFDL 1.0. Mike to write up.</p> <p>Aside: Steve wondered if validation could be used here instead. IBM DFDL has rich enough validation options that effectively make a validation failure behave like a</p>

	<p>parse error. Apache Daffodil does not though.</p> <p>14/7: <a href="#">Mike has created a tracker</a></p> <p><a href="https://github.com/OpenGridForum/DFDL/issues/29">https://github.com/OpenGridForum/DFDL/issues/29</a>. Following from Steve's suggestion, can you use an assert with <code>dfdl:checkConstraints('.')</code> be applied to the type?</p>
--	--

## Closed actions

No	Action

## Deferred actions

No	Action
250	<p><b>Standardise on a single tdml format for DFDL tests (All)</b></p> <p>5/2/14: Steve has requested permission for IBM to view / use the Daffodil tdml files, as a precursor to trying to standardise on a common tdml format. Was formerly part of action 066.</p> <p>...</p> <p>18/2: No further progress</p> <p>11/3: Mike and Steve discussing the best way to share and cooperate on tdml format.</p> <p>25/3: Discussed the creation of an OGF document that will own and define a standardised tdml format.</p> <p>11/4: Proposal is for the OGF document to define a tdml format without Tresys or IBM copyright statement.</p> <p>15/4: Draft document on Redmine</p> <p>...</p> <p>6/5: No further progress</p> <p>20/5: Mark has read through the document. Particularly concerned with how namespaces are handled in the info set.</p> <p>...</p> <p>17/6: No further progress</p> <p>25/6: Mike has added bit order capability as per action 233.</p> <p>...</p> <p>9/12: No further progress</p> <p>6/1/15: Mike to resurrect this as Tresys would like to run their tdml suite against both Daffodil and IBM DFDL.</p> <p>...</p> <p>10/2: No further progress</p> <p>24/2: Mike updating the Daffodil TDML test runner to handle unparser (ie, serializer) tests</p> <p>...</p> <p>14/4: No further progress</p> <p>28/4: Tresys have enhanced their tdml runner to allow unparser tests and round-trip tests (parser-&gt;unparser-&gt;parser) as well as the new tutorial tag (see action 228)</p> <p>12/5: Not discussed</p> <p>...</p> <p>3/11: No progress</p> <p>5/1/16: No progress. Needs more interoperability between implementations to be really useful.</p> <p>...</p> <p>25/7/17: No further progress</p> <p>3/10: No further progress although forthcoming work to add packed/zoned numbers</p>

	<p>may force <a href="https://redmine.ogf.org/issues/339">https://redmine.ogf.org/issues/339</a> progress</p> <p>...</p> <p>11/12: Expected to look at this in the next month or so</p> <p>...</p> <p>4/9/18: No further progress</p> <p>16/10: Mike has started work on a TDML runner that can drive a pluggable DFDL implementation, in support of interoperability testing, including IBM DFDL.</p> <p>1/11: Pluggable TDML runner working. On Github at <a href="https://github.com/OpenDFDL/ibmDFDLCrossTester">https://github.com/OpenDFDL/ibmDFDLCrossTester</a>. Schema resolution for IBM DFDL 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.</p> <p>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.</p> <p>29/11: Good progress on the TDML runner, see email from Mike. The ibmCrossTestRig is not part of Daffodil (because it links against IBM DFDL), but is open source Apache License v2, and is currently in review at <a href="https://github.com/OpenDFDL/ibmDFDLCrossTester/pull/1">https://github.com/OpenDFDL/ibmDFDLCrossTester/pull/1</a>. Steve needs to talk to IBM legal to check this is ok as it currently modifies IBM DFDL sample code.</p> <p>10/1/19: Daffodil have removed the modified IBM DFDL sample code. Steve to ask whether IBM can donate tests from the existing IBM DFDL test suite.</p> <p>7/2: If permitted, the tests from IBM can be used to see how the IBM and Daffodil tdmls have diverged.</p> <p>5/4: IBM are permitted to send Daffodil some example tdmls. Steve to send some to Mike for next call.</p> <p>...</p> <p>31/5: No progress</p> <p>27/6: Steve has sent example TDML files to Mike, under the existing IBM Grant of Copyright agreement with Apache.</p> <p>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.</p> <p>...</p> <p>29/8: No further progress</p> <p>17/10: Noted that this action had its number accidentally swapped with 242 in October 2016. Renumbered back to 250.</p> <p>...</p> <p>23/7/20: No further progress</p> <p>3/9: Nothing likely to happen for this action in medium term, so moving to deferred actions</p>
<b>316</b>	<p><b>Proposed new experimental feature 'User defined functions' (Mike)</b></p> <p>9/1/20: Version 2.5.0. of Daffodil will include this, needs writing up. Does XPath 2.0 provide for this?</p> <p>16/4: Nothing in core XPath 2.0 that allows user-defined functions, that is left to XSL etc. A write-up is here: <a href="https://cwiki.apache.org/confluence/display/DAFFODIL/Proposal%3A+Feature+to+SUPPORT+User+Defined+Functions">https://cwiki.apache.org/confluence/display/DAFFODIL/Proposal%3A+Feature+to+SUPPORT+User+Defined+Functions</a>. Mike has an example where values need to be normalised for later comparison. Bradd has an example where floating point precision needs adjustment. Steve is concerned that DFDL is straying too far into the transformation layer, maybe "user-defined-functions-for-DFDL" should be a separate spec?</p> <p>...</p> <p>3/9: No further progress.</p> <p>17/9: Nothing likely to happen for this action in medium term, so moving to deferred actions.</p>

**Work items:**

No	Item	Owner	Target	Status

**Experimental Features:**

No	Title	Action	GitHub Tracker	Implementor	Experience Document	Release
1	Experimental feature mechanism	301	N/A	Apache Daffodil	<a href="https://github.com/OpenGridForum/DFDL/blob/master/docs/current/gwdi-experimental-features-conventions.docx">https://github.com/OpenGridForum/DFDL/blob/master/docs/current/gwdi-experimental-features-conventions.docx</a>	1.0
2	Elements of type 'xs:hexBinary' with lengthUnits 'bits'	292	<a href="https://github.com/OpenGridForum/DFDL/issues/12">https://github.com/OpenGridForum/DFDL/issues/12</a>	Apache Daffodil		2.0 candidate
3	Variable Path Step in DFDL expression	287	<a href="https://github.com/OpenGridForum/DFDL/issues/13">https://github.com/OpenGridForum/DFDL/issues/13</a>	ESA DFDL4S		2.0 candidate
4	BLOBs as URIs in infoset	312	<a href="https://github.com/OpenGridForum/DFDL/issues/15">https://github.com/OpenGridForum/DFDL/issues/15</a>	Apache Daffodil		2.0 candidate
5	Data streaming layers	304	<a href="https://github.com/OpenGridForum/DFDL/issues/11">https://github.com/OpenGridForum/DFDL/issues/11</a>	Apache Daffodil		2.0 candidate
6	Interoperability	307	N/A	Apache Daffodil	<a href="https://github.com/OpenGridForum/DFDL/blob/master/docs/current/gwde-dfdl-experience-6-v0.1-interoperability.docx">https://github.com/OpenGridForum/DFDL/blob/master/docs/current/gwde-dfdl-experience-6-v0.1-interoperability.docx</a>	1.0
7	Empty element parse policy	306	N/A	Apache Daffodil	<a href="https://github.com/OpenGridForum/DFDL/blob/master/docs/current/gwde-dfdl-experience-7-emptyElementParsePolicy.docx">https://github.com/OpenGridForum/DFDL/blob/master/docs/current/gwde-dfdl-experience-7-emptyElementParsePolicy.docx</a>	1.0
8	Integer enums as strings in infoset	294	<a href="https://github.com/OpenGridForum/DFDL/issues/14">https://github.com/OpenGridForum/DFDL/issues/14</a>	Apache Daffodil		2.0 candidate
10	Indirection using pointers	309	<a href="https://github.com/OpenGridForum/DFDL/issues/17">https://github.com/OpenGridForum/DFDL/issues/17</a>	z/TPF DFDL	<a href="https://github.com/OpenGridForum/DFDL/blob/master/docs/current/gwde-dfdl-experience-8-experimental-indirection.docx">https://github.com/OpenGridForum/DFDL/blob/master/docs/current/gwde-dfdl-experience-8-experimental-indirection.docx</a>	2.0 candidate
11	User-defined functions	316	<a href="https://github.com/OpenGridForum/DFDL/issues/21">https://github.com/OpenGridForum/DFDL/issues/21</a>	Apache Daffodil		2.0 candidate
12	Unparser forward reference variables	289	<a href="https://github.com/OpenGridForum/DFDL/issues/22">https://github.com/OpenGridForum/DFDL/issues/22</a>	Apache Daffodil		2.0 candidate
13	Handle embedded XML & JSON in natural way	322	#27	Apache Daffodil		2.0 candidate



