# 

# Data Format Description Language (DFDL) v1.0

**Experience Document 1**

**Errata for DFDL v1.0 Specification GFD.207**

Status of This Document

Grid Working Document (GWD)

Copyright Notice

Copyright © Open Grid Forum, (2014). Some Rights Reserved. Distribution is unlimited.

Abstract

This document provides experience information to the OGF community on the Data Format Description Language (DFDL) 1.0 specification (GFD-P-R.207).

It lists and describes the non-editorial errata identified in the DFDL 1.0 specification since its publication in September 2014. It contains all errata up to 2014-11-18.

**Contents**

1. Introduction 3

2. Minor Technical Fixes 4

3. Security Considerations 6

4. Contributors 7

5. Intellectual Property Statement 8

6. Disclaimer 9

7. Full Copyright Notice 10

8. References 11

# Introduction

This document has been created to list issues encountered by implementers of the DFDL 1.0 specification GFD.207 [DFDL], and users of implementations of the DFDL 1.0 specification. Specifically, it records all those issues requiring a non-editorial change to the DFDL 1.0 specification, in the form of errata.

The OGF GFD process [GFD] recognises three different kinds of error that may be found in OGF specifications:

*Editorial fixes*. Updates to a document which are not widely announced or publicized.

This category might include headers/footers, spelling, formatting, or simple wording

changes for clarity.

*Minor technical fixes.* Updates to a document which are not simply editorial. For example,

an update to an XML schema or addition to a protocol, to bring the document into

agreement with current practice.

*Major technical fixes.* Such fixes will often require additional technical review and result

in an updated or replaced document.

The following sections of this document list the errata that fall into the last two categories.

All errata are tracked by Redmine Issue trackers [ISSUES].

# Minor Technical Fixes

The following minor technical fixes have been identified.

**2.1**. *Section 13.2.1.* [*https://redmine.ogf.org/issues/230*](https://redmine.ogf.org/issues/230)

Consider a dfdl:escapeScheme annotation with the following properties:

* dfdl:escapeBlockStart="start"
* dfdl:escapeBlockEnd="end"
* dfdl:escapeEscapeCharacter="#"

If this is used to serialize a DFDL Infoset element of type xs:string with value “A hash is a #”, then the value is wrapped with the dfdl:escapeBlockStart and dfdl:escapeBlockEnd, giving simple content "startA hash is a #end".  If this data was parsed, the "#end" will be treated as an escaped escape block end and the parse will fail, reporting that there is no escape block end in the data.   
  
In this scenario, the data is not compliant with the escape scheme, and the DFDL serializer must issue a processing error.

***2.2****. Section 23.* [*https://redmine.ogf.org/issues/231*](https://redmine.ogf.org/issues/231)

XPath 2.0 specification [XPATH2] allows implementation-dependent evaluation of expressions thereby allowing either lazy (sequential) evaluation or full (parallel) evaluation of expressions with OR and AND clauses. This flexibility is not desirable in DFDL 1.0 implementations so the specification is changed to prescribe lazy (sequential) evaluation left-to-right.

**2.3**. *Section 13.2.1.* [*https://redmine.ogf.org/issues/237*](https://redmine.ogf.org/issues/237)

Examples of formats exist where an escape character is active when in front of an in-scope delimiter, but not when in front of another character. Currently DFDL 1.0 will remove all instances of escape characters when parsing regardless of what follows. In order to handle these formats, a new property is added to dfdl:escapeScheme.

|  |  |
| --- | --- |
| **Property Name** | **Description** |
| escapeCharacterPolicy | Enum  Valid values are ‘all’, ‘delimiters’.  Controls when escape characters are removed during parsing, and output during unparsing, when dfdl:escapeKind is 'escapeCharacter'.  When 'all':  During unparsing the following are escaped as described in dfdl:escapeKind when they are in the data.   * Any in-scope terminating delimiter by escaping its first character. * dfdl:escapeCharacter (escaped by dfdl:escapeEscapeCharacter) * any dfdl:extraEscapedCharacters   During parsing, occurrences of dfdl:escapeCharacter and dfdl:escapeEscapeCharacter are interpreted and removed from the data as described in dfdl:escapeKind.  When 'delimiters':  During unparsing the following are escaped as described in dfdl:escapeKind when they are in the data.   * Any in-scope terminating delimiter by escaping its first character. * dfdl:escapeCharacter (escaped by dfdl:escapeEscapeCharacter)   During parsing, occurrences of dfdl:escapeCharacter and dfdl:escapeEscapeCharacter are interpreted and removed from the data as described in dfdl:escapeKind, except that dfdl:escapeCharacter is only removed when it immediately precedes an in-scope terminating delimiter.  Annotation: dfdl:escapeScheme |

Consequential updates to description of dfdl:escapeKind :

“When 'escapeCharacter': On unparsing a single character of the data is escaped by adding a dfdl:escapeCharacter or dfdl:escapeEscapeCharacter immediately before it. The characters to escape are determined by property dfdl:escapeCharacterPolicy.

On parsing any in-scope terminating delimiter encountered in the data is not interpreted as such when it is immediately preceded by the dfdl:escapeCharacter (when not itself preceded by the dfdl:escapeEscapeCharacter). Occurrences of the dfdl:escapeCharacter and dfdl:escapeEscapeCharacter are removed from the data as determined by property dfdl:escapeCharacterPolicy unless the dfdl:escapeCharacter is preceded by the dfdl:escapeEscapeCharacter, or the dfdl:escapeEscapeCharacter does not precede the dfdl:escapeCharacter, respectively.”

**2.4**. *Section 23.5.2.5*. [*https://redmine.ogf.org/issues/239*](https://redmine.ogf.org/issues/239)

The description of fn:exactly-one() does not match the XPath 2.0 specification [XPATH2]. It should state : “Returns the input sequence if it contains exactly one item. Raises an error otherwise.”

**2.5**. *Section 23*. [*https://redmine.ogf.org/issues/240*](https://redmine.ogf.org/issues/240)

XPath 2.0 specification [XPATH2] defines its functions to be in namespace <http://www.w3.org/2005/xpath-functions>. The DFDL specification must bind namespace prefix “fn:” to this namespace.

**2.6**. *Section 5.1*. [*https://redmine.ogf.org/issues/241*](https://redmine.ogf.org/issues/241)

The list of XSDL 1.0 constructs supported by DFDL 1.0 includes the attributes “elementFormDefault” and “form”. These should be explicitly listed.

**2.7**. *Section 13.6.* [*https://redmine.ogf.org/issues/238*](https://redmine.ogf.org/issues/238)

The paragraph in the description of the textNumberPadCharacter property that describes what happens when the pad character is ‘0’ does not cover edge cases such as a sign being present. It is revised as follows:

"When parsing, if the pad character is '0' and dfdl:textTrimKind is 'padChar' then the SimpleContent region is trimmed of the '0' characters as defined by the trimming rules. If at least one '0' character is removed and the trimmed text causes a processing error when parsed, a single '0' character is re-instated and the text is parsed again. This is to handle the case when '0' characters are trimmed away leaving no digits. This rule also applies when the pad character is a DFDL character entity equivalent to '0'. This rule does not apply when the pad character is any other character nor when a pad byte is specified."

**2.8.** *Sections 14.2, 16.* [*https://redmine.ogf.org/issues/243*](https://redmine.ogf.org/issues/243)

Following changes needed to the specification to clarify the behaviour of separator suppression. If it is applicable, the separatorSuppressionPolicy should apply to the whole sequence regardless of the occursCountKind of child elements.

*14.2* - updates to paragraphs that describe positional and non-positional sequences

***“Positional sequence*** - Each occurrence in the sequence can be identified by its position in the data. Typically the components of such a sequence do not have an initiator. In some such sequences, the separators for optional zero-length occurrences may or must be omitted when at the end of the group. In DFDL, a sequence is considered positional if it contains only required elements and/or optional and array elements that have dfdl:occursCountKind 'implicit', 'fixed' or 'expression', and it has dfdl:separatorSuppressionPolicy 'never', 'trailingEmptyStrict'  or 'trailingEmpty'.”

***“Non-positional sequence*** - Occurrences in the sequence cannot be identified by their position in the data alone. Often the components of such a sequence have an initiator. Such sequences sometimes allow the separator to be omitted for optional zero-length occurrences anywhere in the sequence. Speculative parsing might need to be employed by to identify each occurrence. In DFDL, a sequence is non-positional if it contains any optional or array elements that have dfdl:occursCountKind 'parsed' or 'stopValue', and/or it has dfdl:separatorSuppressionPolicy 'anyEmpty'.”

*14.2.2* - updates to the last sentences of the 'When dfdl:occursCountKind is ...' paragraphs to match the table and notes.

“When an element is required and is not an array then one occurrence is always expected along with its separator. The dfdl:separatorSuppressionPolicy of the sequence has no effect (nothing is eligible for suppression). Otherwise the behaviour is dependent on dfdl:occursCountKind.  
When dfdl:occursCountKind is 'fixed' then XSDL minOccurs must equal maxOccurs and that many occurrences are always expected along with their separators. The dfdl:separatorSuppressionPolicy of the sequence has no effect (nothing is eligible for suppression).  
When dfdl:occursCountKind is 'expression' the number of occurrences is given by dfdl:occursCount and exactly that many occurrences are always expected along with their separators. The dfdl:separatorSuppressionPolicy of the sequence has no effect (nothing is eligible for suppression).  
When dfdl:occursCountKind is 'parsed' any number of occurrences and their separators are expected. The dfdl:separatorSuppressionPolicy of the sequence must be 'anyEmpty' and it is a schema definition error otherwise.  
When dfdl:occursCountKind is 'stopValue', any number of occurrences and their separators are expected followed by the stop value and its separator. The dfdl:separatorSuppressionPolicy of the sequence has no effect.   
When dfdl:occursCountKind is 'implicit', between XSDL minOccurs and maxOccurs (inclusive) occurrences and their separators are expected, according to the dfdl:separatorSuppressionPolicy of the sequence.”

*14.2.3* - updates to the last sentences of the 'When dfdlk:occursCountKind is ...' paragraphs to match the table and notes.

When an element is required and is not an array then one occurrence is always output along with its separator. The dfdl:separatorSuppressionPolicy of the sequence has no effect (nothing is eligible for suppression).  
Otherwise the behaviour is dependent on dfdl:occursCountKind.  
When dfdl:occursCountKind is 'fixed' or 'expression' the occurrences in the augmented Infoset are always output along with their separators. The dfdl:separatorSuppressionPolicy of the sequence has no effect (nothing is eligible for suppression).  
When dfdl:occursCountKind is 'parsed' non zero-length occurrences in the augmented Infoset are output along with their separators. The dfdl:separatorSuppressionPolicy of the sequence must be 'anyEmpty' and it is a schema definition error otherwise.  
When dfdl:occursCountKind is 'stopValue' the occurrences in the augmented Infoset are output along with their separators followed by the stop value and its separator, according to the dfdl:separatorSuppressionPolicy of the sequence.  
When dfdl:occursCountKind is 'implicit' the occurrences in the augmented Infoset are output along with their separators, according to the dfdl:separatorSuppressionPolicy of the sequence.

*16* - update to occursStopValue property description. The property is a list of logical values, so need to add: "The dfdl:stopValue property must not be empty string."

**2.9**. *Section 16.1.* [*https://redmine.ogf.org/issues/244*](https://redmine.ogf.org/issues/244)

New clauses added to sections 16.1.1 through 16.1.6 where needed to describe the behaviour when maxOccurs or occursCount is zero.

*16.1.1* (fixed):  
Parsing: When maxOccurs is 0, no occurrences looked for in the data   
Unparsing: When maxOccurs is 0, no occurrences looked for in the infoset or written

*16.1.2* (implicit):  
Parsing: When maxOccurs is 0, no occurrences looked for in the data   
Unparsing: When maxOccurs is 0, no occurrences looked for in the infoset or written

*16.1.4* (expression):  
Parsing: When dfdl:occursCount is 0, no occurrences looked for in the data

**2.10**. *Sections various.* [*https://redmine.ogf.org/issues/245*](https://redmine.ogf.org/issues/245)

The specification is changed to say that a 'fixed length' element with a non-zero length can never have an empty representation. The emptyValueDelimiterPolicy does not therefore apply and all occurrences of the element will have the fixed length. This is to prevent a catch 22 situation when initiators are present. Spec changes:

Add a definition of 'fixed length element' to the glossary in section 3, being an element of specified length where lengthKind is 'explicit' but length is not an expression, or lengthKind is 'implicit' (of simple type only). (Note that the spec also uses 'fixed length' for choice branches where choiceLengthKind is 'explicit')

Add a paragraph to section 9.2.2 which states that empty representation is not possible for fixed length elements with a non-zero length.

Add a clause in 9.4.2.2 and 9.4.2.3 so that the statement about optional occurrences says "If dfdl:emptyValueDelimiterPolicy **is in effect and** is not 'none'...", as the words today do not cover cases when it is ignored (no initiator or terminator).

Add a new paragraph to the property description of emptyValueDelimiterPolicy in section 12.2 "Ignored if the element is fixed length and length is not zero (no empty representation is possible)."

Update existing paragraph of the property description of emptyValueDelimiterPolicy in section 12.2 "It is a schema definition error if dfdl:emptyValueDelimiterPolicy **is in effect and** is set to 'none' or 'terminator' when the parent xs:sequence has dfdl:initiatedContent 'yes'."

Update existing paragraph of the property description of nilValueDelimiterPolicy in section 13.15 "It is a schema definition error if dfdl:nilValueDelimiterPolicy **is in effect and** is set to 'none' or 'terminator' when the parent xs:sequence has dfdl:initiatedContent 'yes'."

Update existing paragraph of the property description of nilKind in section 13.15 to remove the definition of fixed length.

**2.11**. *Section 23.4.* [*https://redmine.ogf.org/issues/246*](https://redmine.ogf.org/issues/246)

Clarify the meaning of an unqualified path step in a DFDL expression. Add new note after table 57 with accompanying reference.

5. NameTest - These QNames are path steps that refer to elements in the DFDL infoset. If such an element is in a namespace, then the NameTest QName must have a prefix which is bound to the namespace. Specifically, any default namespace is not used to implicitly qualify these NameTest QNames. This behavior is consistent with XPath expression usage in XML Schema [x] such as in the path property of the xs:selector and xs:field elements within xs:key and xs:unique constraints, and in related XML standards such as XSLT. Note however, that this behavior is different from the way QNames are used in other places in XML and DFDL Schemas such as the ref property of an element reference, or the dfdl:ref property of a DFDL format annotation. There a QName with no prefix must always be referring to a global declaration or definition, and so is augmented with the default namespace when needed.

[x] Definitive XML Schema (Walmsley, ISBN 0-13-065567-8) page 390, Section 17.8, Table 17-6 says "A child element-type name which must be prefixed if it is in a namespace".

**2.12**. *Section 12.3.7.2.1.* [*https://redmine.ogf.org/issues/247*](https://redmine.ogf.org/issues/247)

Correct the xs:decimal row in table 22 so that the minimum number of bits is 8. This is because lengthUnits 'bits' is not allowed for xs:decimal. A footnote is added to the table stating this restriction.

**2.13***. Section 13.7.1.1.* [*https://redmine.ogf.org/issues/248*](https://redmine.ogf.org/issues/248)

The paragraphs in section 13.7.1.1 are incorrect and should read as follows:

“For both parsing and unparsing, the bit string that represents the content region for a base-2 binary number is converted to/from an Infoset value by a calculation that involves the length and the dfdl:byteOrder and dfdl:bitOrder properties.

When parsing, DFDL specifies how an unsigned integer of unbounded magnitude is computed from a bit string based on its length, and the dfdl:byteOrder and dfdl:bitOrder properties. For signed types, this unbounded integer is converted into a signed value by way of the well-known twos-complement scheme, and for the xs:decimal type, the dfdl:binaryDecimalVirtualPoint property can be used to convert this integer into a decimal value with an integer and a fractional component

A DFDL implementation can use any conversion technique consistent with this description.”

**2.14**. *Section 13.7.* [*https://redmine.ogf.org/issues/249*](https://redmine.ogf.org/issues/249)

In the description of binaryDecimalVirtualPoint, clarify that when unparsing, if the property value is not sufficient to remove the decimal point from the infoset value, it is a processing error. This is true even if the resultant number can be converted into an integer (that is, all digits after the decimal point are zero) because it is an example of excess precision.

**2.15***. Section 12.3.* [*https://redmine.ogf.org/issues/253*](https://redmine.ogf.org/issues/253)

Correct the description of lengthUnits to state it can have the value ‘bits’ for packed calendars.

“- 'bits' may only be used for xs:boolean, xs:byte, xs:short, xs:int, xs:long, xs:unsignedByte, xs:unsignedShort, xs:unsignedInt and xs:unsignedLong simple types with binary representation, and for calendar simple types with binary packed representation. ”

1. Security Considerations

Security considerations are dealt with in the corresponding sections of the DFDL 1.0 specification [DFDL].

No additional security issues have been raised.

1. Contributors

Stephen M. Hanson,

IBM Software Group,

Hursley,

Winchester,UK

[smh@uk.ibm.com](mailto:smh@uk.ibm.com)

Michael J. Beckerle,

Tresys Technology,

Columbia, MD, USA

[mbeckerle@tresys.com](mailto:mbeckerle@tresys.com)

We greatly acknowledge the contributions made to this document by the following people.

Tim Kimber, IBM Software Group, Hursley, UK

Alex Wood, IBM Software Group, Hursley, UK

Mark Frost, IBM Software Group, Hursley, UK

Andrew Edwards, IBM Software Group, Hursley, UK

1. Intellectual Property Statement

The OGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the OGF Secretariat.

The OGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this recommendation. Please address the information to the OGF Executive Director.

1. Disclaimer

This document and the information contained herein is provided on an “As Is” basis and the OGF disclaims all warranties, express or implied, including but not limited to any warranty that the use of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

1. Full Copyright Notice

Copyright (C) Open Grid Forum (2014). Some Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included as references to the derived portions on all such copies and derivative works. The published OGF document from which such works are derived, however, may not be modified in any way, such as by removing the copyright notice or references to the OGF or other organizations, except as needed for the purpose of developing new or updated OGF documents in conformance with the procedures defined in the OGF Document Process, or as required to translate it into languages other than English. OGF, with the approval of its board, may remove this restriction for inclusion of OGF document content for the purpose of producing standards in cooperation with other international standards bodies.

The limited permissions granted above are perpetual and will not be revoked by the OGF or its successors or assignees.

1. References

[DFDL] OGF DFDL 1.0 specification

<http://www.ogf.org/documents/GFD.207.pdf/>

[GFD] OGF Document Process and Requirements <http://www.ogf.org/documents/GFD.152.pdf/>

[XPATH2] XPath 2.0

http://www.w3.org/TR/xpath20/

[ISSUES] OGF Redmine Issue Trackers

<http://redmine.ogf.org/projects/dfdl-wg/issues>