

1 Keyword Guidelines for OASIS Specifications and Standards
2
3 Date: 7. Nov. 2013
4 Version 6.0
5
6 Status: Approved by TAB for review by Chairs and OASIS Board
7

Table of Contents

1. Introduction.....	1
2. References.....	2
3. Keywords in OASIS TC Specifications and Standards.....	2
3.1 Introduction to Keywords.....	2
3.2 Keywords and Normative Text.....	3
3.3 RFC 2119 Rules.....	3
3.3.1 RFC 2119 Examples.....	3
3.3.1.1 MUST.....	3
3.3.1.2 MUST NOT.....	4
3.3.1.3 SHOULD or RECOMMENDED.....	4
3.3.1.4 SHOULD NOT.....	4
3.3.1.5 MAY.....	4
3.3.1.6 OPTIONAL.....	5
3.4 ISO/IEC Directives, Annex H.....	5
3.4.1 ISO/IEC Directives Examples.....	5
3.4.1.1 SHALL.....	5
3.4.1.2 SHALL NOT.....	5
3.4.1.3 SHOULD.....	5
3.4.1.4 SHOULD NOT.....	6
3.4.1.5 NEED NOT.....	6
3.4.1.6 CAN.....	6
3.4.1.7 CANNOT.....	6
4. OASIS non-normative documents: TC Notes (Non-normative Documents).....	6
5. Appendix A – RFC Keywords.....	7
6. Appendix B – ISO/IEC Keywords.....	8
7. Appendix C - Mapping Table of RFC2119 to ISO keywords and suggested synonyms.....	9
8. FAQ – Frequently Asked Questions.....	10

8

9 1. Introduction

10 Keywords establish the requirements that implementers follow in conforming to OASIS specifications
11 and standards. Careful use of keywords is one part of creating standards that help different
12 implementers to have the same interpretation of these requirements and lead to interoperable
13 applications from different vendors.

14 This guide explains how to use two of the more popular keyword sets, [RFC2119] and [ISO/IEC
15 Directives]. After explaining the basic rules for each keyword set, we provide examples of the
16 keywords in use in OASIS specifications.

17 **2. References**

18 **[ISO/IEC Directives]** ISO/IEC Directives, Part 2 (Fifth Edition) Rules for the structure and drafting of
19 International Standards, International Organization for Standardization and International
20 Electrotechnical Commission, 2004.

21 **[RFC 2119]** Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC
22 2119, March 1997. (<http://www.ietf.org/rfc/rfc2119.txt>)

23 **3. Keywords in OASIS TC Specifications and Standards**

24 **3.1 Introduction to Keywords**

25 Specifications and standards are responsible for your ability to plug in a razor or hair dryer when
26 traveling. Or to use your credit card in any credit card terminal. The relevant standards don't care how
27 your razor works or which credit care you have. The internals don't matter but the ability to interoperate
28 does.

29 The term *keywords* for OASIS specifications or standards means terms specified either by [RFC 2119]
30 or the [ISO/IEC Directives]. Every OASIS specification or standard will choose (and use) one or the
31 other. The two keyword sets are never mixed in a specification or standard.

32 Keywords identify the basis for interoperability as defined by a specification or standard. RFC 2119
33 gives the following guidance on keywords (called “imperatives”)

34 Imperatives of the type defined in this memo must be used with care and sparingly. In
35 particular, they **MUST** only be used where it is actually required for interoperation or to
36 limit behavior which has potential for causing harm (e.g., limiting retransmissions) For
37 example, they must not be used to try to impose a particular method on implementors
38 where the method is not required for interoperability. [RFC2119]

39 For example, ODF 1.2 went to great lengths to say how the format was written and to specify its
40 semantics. However, not one word was said about how an implementation would accomplish
41 that task. It wasn't relevant. It could be an in-memory table, graph, key-value data store, etc.
42 The only thing ODF 1.2 constrains is how to interpret the markup and how to write it back out.

43 Normative contents don't always use keywords. Often a descriptive or declarative style reads
44 better than an imperative style based on keywords. In that case, such content may still be
45 referred to by a more general statement – e.g. in a conformance clause – where normative
46 keywords are used to clearly indicate what is expected from a conforming implementation.

47 In a very real sense, ODF 1.2 is a collection of statements about elements and attributes, which
48 are then referred to by keywords, should you want to build a particular type of implementation.
49 If all implementations of that type follow the conformance requirements (using keywords), then
50 the results should be interoperable.

51 For example, under 19.402 Presentation Node Type, I would not say:

52 The presentation:node-type attribute **MUST** specify a node type for an animation
53 element.(incorrect)

54 Rather, as the standard reports:

55 The presentation:node-type attribute specifies a node type for an animation element.

56 The defined values for the presentation:node-type attribute are:
57 This has the advantage of freeing the author to write in simple, declarative prose and to save the
58 hard part of keywords for conformance clauses.

59 **3.2 Keywords and Normative Text**

60 “Normative” text includes the parts of a specification or standard that set forth definitions, rules,
61 conformance clauses and other statements that are part of implementing a standard.
62 By way of contrast, “informative” text contains material that may help understand the standard or give
63 examples of its use, but that don't have to be followed in order to implement the specification or
64 standard.
65 The distinction is an important one because keywords cannot appear as keywords in informative text.
66 The reason being that readers might confuse purely informative text with normative text if keywords
67 were found in informative text.
68 Interoperability requires that everyone recognize normative and informative text the same way. Use of
69 keywords in informative text interferes with a uniform reading of text as normative or informative.
70 Some examples of informative text include: notices, tables of contents, introductions, notes, examples,
71 appendices (appendices can be normative if marked), etc.

72 **3.3 RFC 2119 Rules**

73 **[RFC 2119]** keywords are the most common keywords used in OASIS TC specifications and standards
74 to define normative statements and conformance clauses.

75 [RFC 2119] keywords are written in UPPERCASE. When [RFC2119] keywords are written in
76 lowercase, they have only their normal English usage meaning. In lowercase, [RFC2119] keywords do
77 not state normative or conformance requirements.¹

78 Appendix A reproduces the [RFC 2119] definitions of keywords.

79 **3.3.1 RFC 2119 Examples**

80 **3.3.1.1 MUST**

81 “A PullRequest signal message always indicates in its header (see Section
82 5.2.3.1) the MPC on which the message must be pulled. If no MPC is explicitly
83 identified, the default MPC MUST be pulled from. The pulled message sent in
84 response MUST have been assigned to the indicated MPC.” ([OASIS ebXML
85 Messaging Services Version 3.0: Part 1, Core Features](#), 3.4.3. Definition and
86 Usage Requirements.)
87

88 Note that “must” appears in lower and upper case. In the first sentence, “must” only indicates the
89 intended objective or effect one wants to produce. The second and thirds uses, in upper case, are
90 requirements that must be met by an implementation of ebXML.
91

1 According to RFC Style, <http://www.rfc-editor.org/rfc-style-guide/rfc-style>, lowercase keywords: “To simply specify a necessary logical relationship, the normal lower-case words should be used.”, Requirement Words (RFC 2119) section.

92 **3.3.1.2 MUST NOT**

93
94 “OData-defined system query options are prefixed with "\$". Services may
95 support additional query options not defined in the OData specification, but they
96 MUST NOT begin with the "\$" or "@" character.” ([OData Version 4.0 Part 1: Protocol, Committee Specification Draft 02 /, Public Review Draft 02](#), 6.1 Query
97 Option Extensibility.)
98
99

100 Here MUST NOT appears in upper case and announces a requirement for all implementations. (I don't
101 know of any instances of lower case “must not” in OASIS work. If you do, please drop me a note.)
102

103 **3.3.1.3 SHOULD or RECOMMENDED**

104
105 “If the eb:PartyId/@type attribute is not present, the content of the PartyId
106 element MUST be a URI [RFC2396], otherwise the Receiving MSH SHOULD
107 report a "ValueInconsistent" error with severity "error". It is strongly
108 RECOMMENDED that the content of the eb:PartyId element be a URI.” ([OASIS
109 ebXML Messaging Services Version 3.0: Part 1, Core Features](#), .5.2.2.4.
110 eb:Messaging/eb:UserMessage/eb:PartyInfo/eb:From/eb:PartyId.)
111

112 The use of RFC 2119 SHOULD and RECOMMENDED are shown by use of upper case. The example
113 specification has numerous uses of “should” in lower case, i.e., in normal English usage. The “strongly
114 RECOMMENDED” does not require an implementation to “report a ValueInconsistent error,” but
115 doing so improves interoperability.
116

117 **3.3.1.4 SHOULD NOT**

118
119 “OData services SHOULD NOT require any query options to be specified in a
120 request. Services SHOULD fail any request that contains query options that they
121 not understand and MUST fail any request that contains unsupported OData
122 query options defined in the version of this specification supported by the
123 service.”([OData Version 4.0 Part 1: Protocol, Committee Specification Draft 02 /, Public Review Draft 02](#), 6.1 Query Option Extensibility.)
124
125

126 It is recommended that ODATA services not require query options be specified in a request but, after
127 considering all the implications, an implementation may do so.
128

129 **3.3.1.5 MAY**

130
131 “Policy sets MAY be included in an enclosing <PolicySet> element either
132 directly using the <PolicySet> element or indirectly using the
133 <PolicySetIdReference> element. Policies MAY be included in an enclosing
134 <PolicySet> element either directly using the <Policy> element or indirectly
135 using the <PolicyIdReference> element.” ([eXtensible Access Control Markup
136 Language \(XACML\) Version 3.0](#), 5 Syntax (normative, with the exception of the
137 schema fragments)

138
139 A correct usage of MAY as a keyword but also an illustration of designating part of a section as
140 normative. Designation of portions of a specification as normative is a step towards producing a
141 standard that supports interoperability.
142

143 **3.3.1.6 OPTIONAL**

144
145 “The <Response> element encapsulates the authorization decision produced by
146 the PDP. It includes a sequence of one or more results, with one <Result>
147 element per requested resource. Multiple results MAY be returned by some
148 implementations, in particular those that support the XACML Profile for
149 Requests for Multiple Resources [Multi]. Support for multiple results is
150 OPTIONAL.” ([eXtensible Access Control Markup Language \(XACML\) Version](#)
151 [3.0](#), 5 Syntax (normative, with the exception of the schema fragments)
152

153 Be aware of OPTIONAL as an alternative to MAY when required by the text.

154 **3.4 ISO/IEC Directives, Annex H**

155 Unlike [RFC 2119], Annex H of [ISO/IEC Directives] does not distinguish between upper and lower
156 case forms of its keywords. Using the [ISO/IEC Directives], an author can write keywords in upper or
157 lower case. Annex H does define equivalent expressions for keywords, to be used in exceptional cases.

158 Appendix B summarizes the definitions in Annex H.

159 **3.4.1 ISO/IEC Directives Examples**

160 In the ISO/IEC examples note that keywords do not require UPPER case or **bolding** in order to be
161 keywords. They are keywords by definition, not typography. If you need a verbal alternative to “shall,”
162 those are defined in Annex H of [ISO/IEC Directives]. An attempt at non-keyword alternatives to
163 “shall” appears in Appendix C of this document.

164 **3.4.1.1 SHALL**

165 “An OpenDocument document shall meet the following requirements.” ([Open](#)
166 [Document Format for Office Applications \(OpenDocument\) Version 1.2](#), 2.2.1
167 OpenDocument Document.)
168

169 **3.4.1.2 SHALL NOT**

170 “OpenDocument extended documents may contain elements and attributes not
171 defined by the OpenDocument schema. Elements and attributes not defined by
172 the OpenDocument schema are called foreign elements and attributes. Foreign
173 elements and attributes shall not be associated with a namespace that is listed in
174 tables 1, 2 or 3 of section 1.5.” ([Open Document Format for Office Applications](#)
175 [\(OpenDocument\) Version 1.2](#), 3.17 Foreign Elements and Attributes.)
176

177 **3.4.1.3 SHOULD**

178

179 “The generator string should allow OpenDocument consumers to distinguish
180 between all released versions of a producer.” ([Open Document Format for Office
181 Applications \(OpenDocument\) Version 1.2, 4.3.2.1 <meta:generator>.](#))
182

183 **3.4.1.4 SHOULD NOT**

184
185 “consumers should not permit characters defined by the [SQL] feature F392 for
186 new or changed names of tables, views, columns, and queries.” ([Open Document
187 Format for Office Applications \(OpenDocument\) Version 1.2, 19.49 db:enable-
188 sql92-check.](#))
189

190 **3.4.1.5 NEED NOT**

191
192 “letters in a custom shape need not have the same height.” ([Open Document
193 Format for Office Applications \(OpenDocument\) Version 1.2, 19.224 draw:text-
194 path-same-letter-heights.](#))
195

196 **3.4.1.6 CAN**

197
198 “The draw:transform attribute specifies a list of transformations that can be
199 applied to a drawing shape.” ([Open Document Format for Office Applications
200 \(OpenDocument\) Version 1.2, 19.228 draw:transform.](#))
201

202 **3.4.1.7 CANNOT**

203
204 “The boslevel value cannot cause entities to be included in the BOS if doing so
205 would exceed the maximum BOS level.” (ISO/IEC 10179:1996 Document Style
206 Semantics and Specification Language, 6.5.2 HyTime BOS control data
207 attributes.)

208 **4. OASIS non-normative documents: TC Notes (Non-normative 209 Documents)**

210 OASIS TC Notes (non-normative documents) do not specify conformance clauses. To avoid confusion
211 with OASIS TC Specifications and Standards, citation of or use of [RFC 2119] or [ISO/IEC
212 Directives] should be avoided in OASIS TC Notes (Non-normative Documents).

213

214 5. Appendix A – RFC Keywords

215 [RFC2119] defines its keywords as follows²:

216 “1. MUST This word, or the terms "REQUIRED" or "SHALL", mean that the
217 definition is an absolute requirement of the specification.

218 2. MUST NOT This phrase, or the phrase "SHALL NOT", mean that the definition
219 is an absolute prohibition of the specification.

220 3. SHOULD This word, or the adjective "RECOMMENDED", mean that there may
221 exist valid reasons in particular circumstances to ignore a particular item, but the full
222 implications must be understood and carefully weighed before choosing a different
223 course.

224 4. SHOULD NOT This phrase, or the phrase "NOT RECOMMENDED" mean that
225 there may exist valid reasons in particular circumstances when the particular behavior
226 is acceptable or even useful, but the full implications should be understood and the
227 case carefully weighed before implementing any behavior described with this label.

228 5. MAY This word, or the adjective "OPTIONAL", mean that an item is truly
229 optional. One vendor may choose to include the item because a particular
230 marketplace requires it or because the vendor feels that it enhances the product while
231 another vendor may omit the same item. An implementation which does not include a
232 particular option MUST be prepared to interoperate with another implementation
233 which does include the option, though perhaps with reduced functionality. In the
234 same vein an implementation which does include a particular option MUST be
235 prepared to interoperate with another implementation which does not include the
236 option (except, of course, for the feature the option provides.)”

237

2 This is a direct quote from RFC 2119, <http://www.ietf.org/rfc/rfc2119.txt>

238 6. Appendix B – ISO/IEC Keywords

239 ISO/IEC keywords are defined in Annex H of the [ISO/IEC Directives].

240 TCs that use ISO/IEC keywords should consult Annex H for the normative definitions of those
241 keywords. For use with the mapping table in Appendix C, a synopsis of Annex H reads as follows:

- 242 • **shall** – to indicate requirements strictly to be followed in order to conform to the standard and
243 in which no deviation is permitted. Do not use “must” as an alternative for “shall”.
- 244 • **shall not** - converse of shall. Do not use “must not” instead of “shall not”.
- 245 • **should** – to indicate that among several possibilities one is recommended as particularly
246 suitable, without mentioning or excluding others.
- 247 • **should not** – converse of should.
- 248 • **may** – to indicate a course of action permissible within the limits of the standard. Do not use
249 “can” instead of “may”
- 250 • **need not** – to indicate a course of action is not required. (converse of may)
- 251 • **can** – statement of possibility and capability, whether material, physical, or causal.
- 252 • **cannot** – converse of can.

253 Annex H also defines equivalent expressions for keywords, to be used in exceptional cases. See Annex
254 H, [ISO/IEC Directives] for the details.

255 **7. Appendix C - Mapping Table of RFC2119 to ISO keywords and**
 256 **suggested synonyms**

257 Table 1 lists semantic equivalents between RFC2119 and ISO keywords. Where there is an empty cell,
 258 this means there is no equivalent in that set. If specification writers restrict themselves to keywords that
 259 have semantic equivalents, conversion between RFC2119 and ISO, or vice versa will be easier.

260 Note for the purpose of this exercise we consider RFC2119 treatment of interoperability to be narrow,
 261 and interpret requirements as broadly as possible in the manner interpreted by ISO. The third column in
 262 the table lists suggested non-normative synonyms that should be considered when specification writers
 263 are trying to avoid using one the formal keywords.

264 **Table 1**

RFC 2119	Annex H (ISO)	Non-NormativeSynonyms
MUST, SHALL, REQUIRED	shall	will
MUST NOT, SHALL NOT, REQUIRED	shall not	will not, “it is not possible that”
SHOULD/RECOMMENDED	should	ought to
SHOULD NOT/NOT RECOMMENDED	should not	ought not to
MAY/OPTIONAL	may	could
* ³	need not	might not
*	can	is capable of
*	cannot	is not capable of

265

3 A “*” indicates no equivalent for *need not*, *can* or *can not* appears in RFC2119. The “?” indicates the TAB's inability to suggest good synonyms for them.

266 8. FAQ – Frequently Asked Questions

267 1. When are RFC 2119 keywords (or other keywords) required?

268 All TC work products that will become OASIS TC specifications or OASIS standards, should
269 use [RFC 2119] keywords.

270 2. When are RFC 2119 keywords (or other keywords) to be avoided?

271 When a TC is writing a TC Note, also known as a “non-standards track” work product, it should
272 not use [RFC 2119], to avoid confusion with OASIS TC Specifications and Standards.

273 3. Do keywords only appear in conformance clauses?

274 No. Keywords appear in normative parts of a document that are then referred to by clauses in a
275 conformance clause.

276 4. As an editor, why would I use ISO keywords instead of RFC in a specification?

277 If you are planning on submitting an OASIS TC Specification or Standard to ISO/IEC, you can
278 use [RFC 2119] keywords on a first submission.⁴ However, on subsequent submissions, you
279 will be required to conform to [ISO/IEC Directives], which will require use of ISO keywords.

280 5. Is “MAY NOT” a keyword?

281 No, although there is an example of “MAY NOT” in an OASIS specification.

282 Note: When evaluating and, or, or n-of, it MAY NOT be necessary to attempt a
283 full evaluation of each argument in order to determine whether the evaluation of
284 the argument would result in "Indeterminate".

285
286 In a note, which is non-normative text, keywords should not appear. Instead of “MAY NOT,”
287 the text should read: “may not.”

4 As of August, 2013. ISO/IEC rules change so verify the current rules before choosing your keywords.