

1 **IEEE 1484.20.1/Draft 8**
2 **Draft Standard for Learning Technology—**
3 **Data Model for Reusable Competency Definitions**

4 Sponsor
5 Learning Technology Standards Committee
6 of the
7 IEEE Computer Society
8

9 **Abstract:** This Standard defines a data model for describing, referencing, and shar-
10 ing competency definitions, primarily in the context of online and distributed learning.
11 This Standard provides a way to represent formally the key characteristics of a com-
12 petency, independently of its use in any particular context. It enables interoperability
13 among learning systems that deal with competency information by providing a means
14 for them to refer to common definitions with common meanings.

15 **Keywords:** competency, competency definition, reusable competency definition

16

17 Copyright © 2007 by the IEEE.
18 Three Park Avenue
19 New York, New York 10016-5997, USA
20 All rights reserved.

21 This document is an unapproved draft of a proposed IEEE Standard. As such, this
22 document is subject to change. **USE AT YOUR OWN RISK!** Because this is an ap-
23 proved draft, this document must not be utilized for any conformance/compliance
24 purposes. Permission is hereby granted for IEEE Standards Committee participants
25 to reproduce this document for purposes of international standardization considera-
26 tion. Prior to adoption of this document, in whole or in part, by another standards de-
27 velopment organization permission must first be obtained from the Manager,
28 Standards Intellectual Property, IEEE Standards Activities Department. Other entities
29 seeking permission to reproduce this document, in whole or in part, must obtain per-
30 mission from the Manager, Standards Intellectual Property, IEEE Standards Activities
31 Department.

32 IEEE Standards Activities Department
33 Manager, Standards Intellectual Property
34 445 Hoes Lane
35 Piscataway, NJ 08854, USA

36

37 *[NOTE: Information about IEEE LTSC P1484.20 can be found at:*38 <http://www.ieeeltsc.org/>39 *This note will be removed upon reaching the final draft of this IEEE document.]*40 **Introduction**

41 **This introduction is not a part of P1484.20.1, Draft Standard for Learning**
 42 **Technology—Data Model for Reusable Competency Definitions.**

43 This Standard defines a data model for describing, referencing, and sharing competency defini-
 44 tions, primarily in the context of online and distributed learning. This Standard provides a
 45 way to represent formally the key characteristics of a competency, independently of its use in
 46 any particular context. It enables interoperability among learning systems that deal with com-
 47 petency information by providing a means for them to refer to common definitions with
 48 common meanings.

49 **Dedication**

50 The LTSC 1484.20 Working Group dedicates this standard to the memory of Claude Ostyn.
 51 As chair of the LTSC 1484.20 Working Group, he provided initiative, guidance, and technical
 52 insight that made this standard possible. His hard work is much appreciated and will be re-
 53 membered.

54 **Participants**

55 At the time this Standard was completed, the Competency Data Standards Working Group
 56 had the following membership:

Claude Ostyn, <i>Chair</i> Scott Lewis, <i>Technical Editor</i>		
Michael Brown	Chad Kainz	Robby Robson
Mike Collett	Fanny Klett	Christian Stracke
Geoffrey Darnton	Rolf Lindner	Schawn Thropp
Geoffrey A. Frank	Daniel Rehak	Luk Vervenne
Jason Haag		

57 The following individual members of the balloting committee voted on this Standard. Ballot-
58 ters may have voted for approval, disapproval, or abstention. (To be provided by IEEE editor
59 at time of publication.)

60 **Acknowledgements**

61 This Standard is based on the *IMS Reusable Definition of Competency or Educational Objec-*
62 *tive Specification*, Version 1.0, published on October 25, 2002 by the IMS Global Learning
63 Consortium, Inc.

64

65	Contents	
66	1. Overview	1
67	1.1 Scope	1
68	1.2 Purpose	1
69	2. Normative references	1
70	3. Definitions, acronyms, and abbreviations	3
71	3.1 Definitions.....	3
72	3.2 Acronyms and abbreviations.....	3
73	4. Special terms	3
74	5. Conformance	4
75	5.1 Shall and shall not.....	4
76	5.2 RCD instances.....	4
77	5.3 Smallest permitted maximum values.....	4
78	6. Data model	5
79	6.1 General information.....	5
80	6.2 Reusable competency definition.....	6
81	6.2.1 Identifier.....	7
82	6.2.2 Title	7
83	6.2.3 Description.....	8
84	6.2.4 Definition	9
85	6.2.5 Metadata.....	13
86	6.3 Auxiliary data types.....	15
87	6.3.1 Any type.....	16
88	6.3.2 LangString type	16
89	6.3.3 Long identifier type	18
90	Annex A (informative) Bibliography	20
91	Annex B (informative) Conceptual overview	21
92	B.1 Objectives	21
93	B.2 Functional overview	21
94	B.3 Data model overview	22
95	B.4 Taxonomies of reusable competency definitions	23
96	Annex C (informative) An example XML binding schema	24

97

98 **Draft Standard for Learning Technology—** 99 **Data Model for Reusable Competency Definitions**

100 **1. Overview**

101 **1.1 Scope**

102 This Standard defines a data model for describing, referencing, and sharing competency defini-
103 tions, primarily in the context of online and distributed learning. This Standard provides a
104 way to represent formally the key characteristics of a competency, independently of its use in
105 any particular context. It enables interoperability among learning systems that deal with com-
106 petency information by providing a means for them to refer to common definitions with
107 common meanings.

108 This Standard enables information about competencies to be encoded and shared. This Stan-
109 dard does not define whether a competency is a skill, knowledge, ability, attitude, or learning
110 outcome but can be used to capture information about any of these. Policies regarding reusa-
111 ble competency definitions (RCDs), such as the best practice to look for an existing definition
112 to reuse instead of inventing a new one for the same purpose, are outside of the scope of this
113 Standard.

114 **1.2 Purpose**

115 This Standard is based on the existing IMS Global Learning Consortium (IMS) specification
116 entitled *IMS Reusable Definition of Competency or Educational Objective (RDCEO)* [B2]¹.
117 Implementations that conform to the IMS specification will conform to this Standard.

118 **2. Normative references**

119 The following referenced documents are indispensable for the application of this Standard.
120 For dated references, only the edition cited applies. For undated references, the latest edition
121 of the referenced document (including any amendments) applies.

¹ The numbers in brackets correspond to those of the bibliography in Annex A.

- 122 IEEE 1484.12.1–2002, Standard for Learning Object Metadata.²
- 123 IETF RFC 3986:2005, Uniform Resource Identifiers (URI): Generic Syntax.³
- 124 ISO 639–1, Code for the representation of names of languages – Part 1: Alpha-2 code.⁴
- 125 ISO 639–2, Codes for the representation of names of languages – Part 2: Alpha-3 code.
- 126 ISO 3166–1, Codes for the representation of names of countries and their subdivisions – Part
127 1: Country codes.
- 128 ISO/IEC 10646, Information technology – Universal Multiple-Octet Coded Character Set
129 (UCS).⁵
- 130 ISO/IEC 11404:1996, Information technology – Programming languages, their environments
131 and system software interfaces – Language-independent datatypes.⁶
- 132 W3C Recommendation (28 October 2004), XML Schema Part 2: Datatypes, Second Edition.⁷

² IEEE publications are available from the Institute of Electrical and Electronics Engineers, Inc., 445 Hoes Lane, Piscataway, NJ 08854, USA (<http://standards.ieee.org/>).

³ IETF publications are available from the Internet Engineering Task Force website at <http://ietf.org/rfc.html>.

⁴ ISO publications are available from the ISO Central Secretariat, Case Postale 56, 1 rue de Varembe, CH–1211, Genève 20, Switzerland/Suisse (<http://www.iso.ch/>). ISO publications are also available in the United States from the Sales Department, American National Standards Institute, 25 West 43rd Street, 4th Floor, New York, NY 10036, USA (<http://www.ansi.org/>).

⁵ ISO/IEC publications are available from the ISO Central Secretariat, Case Postale 56, 1 rue de Varembe, CH–1211, Genève 20, Switzerland/Suisse (<http://www.iso.ch/>). ISO/IEC publications are also available in the United States from Global Engineering Documents, 15 Inverness Way East, Englewood, CO 80112, USA (<http://global.ihs.com/>). Electronic copies are available in the United States from American National Standards Institute, 25 West 43rd Street, 4th Floor, New York, NY 10036, USA (<http://www.ansi.org/>).

⁶ ISO/IEC 11404:1996 may be downloaded at no cost from <http://www.iso.org/>.

⁷ W3C publications are available from the World Wide Web Consortium, 32 Vassar Street, Room 32–G515, Cambridge, MA 02139, USA (<http://www.w3.org/>).

133 3. Definitions, acronyms, and abbreviations

134 3.1 Definitions

135 For purposes of this document, the following terms and definitions apply. *The Authoritative*
136 *Dictionary of IEEE Standards Terms* [B1] should be referenced for terms not defined in this
137 Clause.

138 **competency definition record:** An instance of a data structure that represents an identifiable
139 competency definition.

140 **LangString:** A data type that represents one or more character strings. A LangString value
141 may include multiple, semantically equivalent, character strings, such as translations or ex-
142 pressions of a description in different languages.

143 3.2 Acronyms and abbreviations

144 IMS: IMS Global Learning Consortium

145 RCD: reusable competency definition

146 RDCEO: Reusable Definition of Competency or Educational Objective

147 SPM: smallest permitted maximum

148 URI: Uniform Resource Identifier

149 XML: Extensible Markup Language

150 4. Special terms

151 In this Standard, the following special terms and definitions apply.

152 **competency:** Any aspect of competence, such as knowledge, skill, attitude, ability, or
153 learning objective.

154 NOTE—In this Standard, the term “competency” is to be interpreted in the broadest sense to in-
155 clude learning objectives (those things that are sought) as well as competencies (those things that
156 are achieved). The term “competency” is also used to include all classes of things that someone,
157 or potentially something, can be competent in, although some communities of practice use the
158 term with nuance, for example limiting its use to skill and excluding knowledge or understanding.

159 5. Conformance

160 5.1 Shall and shall not

161 In this Standard, “shall” is to be interpreted as a requirement on an implementation; “shall not”
162 is to be interpreted as a prohibition.

163 5.2 RCD instances

164 A conforming RCD instance shall conform to the data model as specified in Clause 5.

165 5.3 Smallest permitted maximum values

166 This standard defines smallest permitted maximum (SPM) values for many data elements. The
167 SPM is the number of entries that a conforming implementation is required to process for a
168 given data element in an RCD instance. An RCD instance may contain more entries for a data
169 element than the SPM. An implementation that receives entries in excess of the SPM shall not
170 produce an error. The implementation may, but is not required to, process those entries. For
171 the purposes of the SPM, an entry may be an element within a collection, such as a bag or
172 record, or it may be a character in a string. SPM values are defined for

- 173 – **Items in a collection:** For example, an SPM is defined for Statement elements
174 within a Definition. If the SPM for the number of Statement elements is n but
175 the number of statements in an RCD instance is $n + 1$, an implementation that
176 receives the Statement elements shall not produce an error. The implementa-
177 tion shall process at least n elements but may ignore the element that exceeds
178 the SPM.
- 179 – **Character strings defined in the types CharacterString and LangString:**
180 For example, if a string found in an element of one of these types in an RCD
181 instance contains more characters than specified by the SPM, an implementa-
182 tion that receives the string shall not produce an error but may ignore the extra
183 characters.

184 NOTE 1—The intent is for the SPM values to cover most cases.

185 NOTE 2—The meaning of “process” in this subclause depends on the nature of the application.

186 NOTE 3—This Standard does not define any provision for whether or how a system may process
187 more than the SPM value for a particular data element.

188 NOTE 4—This Standard does not define failure conditions if a system cannot process the SPM
189 value for a particular data element.

190 NOTE 5—If the SPM for the number of elements in a collection is n , but the number of elements
 191 is $n + 1$, the selection of which n elements an implementation is required to process is left to the
 192 implementation.

193 6. Data model

194 6.1 General information

195 Clause 6 defines the data elements of an RCD. Unless otherwise noted, all components of
 196 records are optional in an RCD instance. The data elements of an RCD are described in 6.2.
 197 The auxiliary data types used in conjunction with the data elements are described in 6.3. (See
 198 Annex B for a conceptual overview of RCDs and the data model.)

199 NOTE 1—The use of ISO/IEC 11404:1996⁸ notation in the synopses in 6.2 and 6.3 is for infor-
 200 mative purposes only and is not meant to prescribe a specific implementation. A complete im-
 201 plementation of the operations defined in ISO/IEC 11404:1996 is not required for conformance.

202 NOTE 2—The ISO/IEC 11404:1996 notation describes the semantics of the language-
 203 independent data types across all bindings (e.g., implementation of a data type as itself, its sub-
 204 types, its subclasses, and its specializations). For example, an ISO/IEC 11404:1996 record may
 205 be implemented as an Extensible Markup Language (XML) complexType; an ISO/IEC
 206 11404:1996 characterstring may be implemented in an encoding (ISO/IEC 646, ISO/IEC 8859-1,
 207 UTF-8, UTF-16, UTF-32, etc., see [B3], [B4], [B5], [B6], [B7]) that supports the repertoire speci-
 208 fied for the characterstring data type.

209 NOTE 3—All examples in 6.2 and 6.3 are informative and do not endorse any particular binding.

210 NOTE 4—The following language-independent data types used in this Standard are defined in
 211 ISO/IEC 11404:1996: bag, characterstring, and record.

212 NOTE 5—The labels for data elements and data types in the synopses in 6.2 and 6.3 are for refer-
 213 ence, only. An implementation is not required to use the same labels, as long as the data elements
 214 and data types are semantically equivalent.

215 NOTE 6—This Standard does not define a specific extension mechanism for the data model. Im-
 216 plementers may define bindings that allow additional elements or create additional data models
 217 for competency data. Such models may be used to augment this model to support different appli-
 218 cation domains. Extensions should not duplicate or redefine the semantics of elements defined in
 219 this Standard.

220 NOTE 7—The inclusion of extension elements in an RCD instance may cause interoperability
 221 issues with some implementations or bindings. Therefore, extensions should be provided by in-
 222 cluding the RCD data model inside an extended data model, rather than by inserting extension
 223 elements within the RCD data model itself.

⁸ For information on normative references, see Clause 2.

224 NOTE 8—The RCD data model can be used in profiles and data models defined in particular im-
 225 plementations or in other standards. For example, a profile for learning objectives or assessable
 226 competencies might not define extension elements, but might extend the conformance require-
 227 ments to specify that a Definition element is mandatory in every RCD instance that conforms to
 228 the profile and that the Definition element has to contain several Statement elements with specific
 229 names. Another implementation might define a data model for a collection of RCD instances. Yet
 230 another implementation might define a map or graph to represent a competency model, with the
 231 nodes in the graph referencing RCD instances through their unique identifier.

232 6.2 Reusable competency definition

233 Synopsis

```

234 reusable_competency_definition :
235 record
236 (
237     identifier :
238         long_identifier_type,
239         // Mandatory
240         // Occurs 1 time
241     title :
242         bag of langstring_type(1000),
243         // Mandatory
244         // Occurs 1 time
245         // SPM: 20 instances of langstring_type in the bag
246         // The parameter value is the SPM for the number of
247         // of characters in the string element of the
248         // langstring_type
249     description :
250         bag of langstring_type(4000),
251         // Optional
252         // Occurs 0 or 1 times
253         // SPM: 20 instances of langstring_type in the bag
254         // The parameter value is the SPM for the number of
255         // of characters in the string element of the
256         // langstring_type
257     definition :
258         definition_type,
259         // Optional
260         // Occurs 0 or more times
261         // SPM: 10 instances of definition in a
262         // reusable_competency_definition record
263     metadata :
264         metadata_type,
265         // Optional (implied default values - see 6.2.5)
266         // Occurs 0 or 1 times
267 )

```

268 **Description**

269 The components of `reusable_competency_definition` are defined in 6.2.1 – 6.2.5.
 270 Identifier and Title are mandatory and shall be included in RCD instances. Depending on the
 271 implementation, an instance of `reusable_competency_definition` may include any or
 272 all of the elements Description, Definition, and Metadata.

273 All elements in this data model are intrinsically unordered. The order of the elements in the
 274 data model synopses and the order of the values in a list of values are not significant. For ex-
 275 ample, if an RCD instance includes three Definition elements, their order is not significant.
 276 They may appear in any order without loss of information.

277 NOTE—A binding may impose a particular ordering on RCD instances that conform to that bind-
 278 ing.

279 **6.2.1 Identifier**280 **Synopsis**

```
281     identifier :
282         long_identifier_type,
283         // Mandatory
284         // Occurs 1 time
```

285 **Description**

286 This mandatory data element is a globally unique label that identifies the RCD. The label shall
 287 be unique across all systems that store or process the RCD.

288 `long_identifier_type` is defined in 6.3.3.

289 NOTE 1—This data element uses the same subelements as the Identifier element defined in IEEE
 290 1484.12.1–2002 and consists of two subelements, Catalog and Entry (see 6.3.3).

291 NOTE 2—This Standard does not specify any method to define or test the global uniqueness of
 292 the identifier.

293 **6.2.2 Title**294 **Synopsis**

```
295     title :
296         bag of langstring_type(1000),
297         // Mandatory
298         // Occurs 1 time
299         // SPM: 20 instances of langstring_type in the bag
300         // The parameter value is the SPM for the number of
301         // of characters in the string element of the
302         // langstring_type
```

303 Description

304 This mandatory data element is a single, text label for the RCD. The label is a human-readable
 305 name for the RCD. Because different application domains may coincidentally define the same
 306 title for different competency definitions, the Identifier element (see 6.2.1), not the Title, shall
 307 be used to distinguish among RCDs. The Title element is intended to provide a title, only, and
 308 should not be used to provide descriptive detail in lieu of the Description element (see 6.2.3).

309 `langstring_type` is defined in 6.3.2.

310 NOTE 1—This data element may be expressed in multiple languages. Each translation is
 311 represented by an instantiation of `langstring_type`.

312 NOTE 2—While the Identifier element (see 6.2.1) provides the definitive reference to the defini-
 313 tion, it is typically unintelligible. The Title element provides a convenient, alternative, readable
 314 form. Examples: “English proficiency”, “Schmiblick failure diagnostic level 4”, “Conflict resolu-
 315 tion skills”.

316 6.2.3 Description**317 Synopsis**

```
318     description :
319         bag of langstring_type(4000),
320         // Optional
321         // Occurs 0 or 1 times
322         // SPM: 20 instances of langstring_type in the bag
323         // The parameter value is the SPM for the number of
324         // of characters in the string element of the
325         // langstring_type
```

326 Description

327 This optional data element is a human-readable description of the competency. It is an un-
 328 structured character string meant to be interpretable only by humans.

329 `langstring_type` is defined in 6.3.2.

330 NOTE 1— This data element may be expressed in multiple languages. Each translation is
 331 represented by an instantiation of `langstring_type`.

332 NOTE 2—This data element is typically more explicative than the Title element (see 6.2.2). Ex-
 333 amples: “Proficiency in written and spoken English and use of English for meaningful oral or
 334 written expression”, “Performance of level 4 diagnostic as specified in IETM #SCMBLK007”.

335 **6.2.4 Definition**336 **Synopsis**

```

337     definition :
338         definition_type,
339         // Optional
340         // Occurs 0 or more times
341         // SPM: 10 instances of definition in a
342         // reusable_competency_definition record
343
344     definition_type = record
345     (
346         model_source :
347             characterstring(iso-10646),
348             // Optional
349             // Occurs 0 or 1 times
350             // SPM: 1000 characters
351         statement :
352             statement_type,
353             // Mandatory
354             // Occurs 1 or more times
355             // SPM: 10 instances of statement in a
356             // definition_type record
357     ),

```

358 **Description**

359 This optional data element is a structured description that provides a formal definition of the
360 competency, usually using attributes taken from a specific model of how a competency should
361 be structured or defined. This data element shall contain zero or one Model source elements
362 and at least one Statement element. It may contain multiple Statement elements.

363 NOTE 1—Typically, the models that underlie this data element define competencies in terms of
364 “statement, conditions, criteria,” “proficiency, criteria, indicators,” “standards, performance
365 indicators, outcomes,” “abilities, basic skills, content, process,” and similar sets of statements.

366 NOTE 2—This data element provides a structure for including an arbitrary collection of state-
367 ments that describe a competency. The author of an RCD is free to use this data element in any
368 way that best describes the competency.

369 **6.2.4.1 Model source**370 **Synopsis**

```

371     model_source :
372         characterstring(iso-10646),
373         // Optional
374         // Occurs 0 or 1 times
375         // SPM: 1000 characters

```

376 **Description**

377 This optional data element is the source of the model used for the competency definition. The
 378 characters in the string shall belong to the repertoire of ISO/IEC 10646 as allowed by IETF
 379 RFC 3986:2005.

380 NOTE—The value of this data element should be sufficiently specific to avoid conflict with other
 381 source names; therefore, the value should be a Uniform Resource Identifier (URI). If the value of
 382 this data element is a URI, it may point to an actual document that defines the source formally.
 383 Examples: “<http://example.org/3-part-learning-objective>”, “<http://example.edu/ref/los.xml>”.

384 **6.2.4.2 Statement**385 **Synopsis**

```

386     statement :
387         statement_type,
388         // Mandatory
389         // Occurs 1 or more times
390
391     statement_type = record
392     (
393         statement_id :
394             long_identifier_type,
395             // Conditionally mandatory (see description)
396             // Occurs 0 or 1 times
397         statement_name :
398             characterstring(iso-10646),
399             // Conditionally mandatory (see description)
400             // Occurs 0 or 1 times
401             // SPM: 1000 characters
402         statement_text :
403             bag of langstring_type(1000),
404             // Conditionally mandatory (see description)
405             // Occurs 0 or 1 times
406             // SPM: 20 instances of langstring_type in the bag
407             // The parameter value is the SPM for the number of
408             // of characters in the string element of the
409             // langstring_type

```

```

410     statement_token :
411         vocabulary_type,
412         // Conditionally mandatory (see description)
413         // Occurs 0 or 1 times
414     ),

```

415 **Description**

416 Each instance of this data element is a description of a single characteristic of a Definition
 417 element (see 6.2.4). At least one Statement element is mandatory if a Definition element ex-
 418 ists. A record of type `statement_type` shall contain one or more of the elements defined in
 419 `statement_type`. None of these elements is specifically required, but at least one shall be
 420 present.

421 For example, a particular learning-objective model might require a list of specific statement
 422 strings, each of which has a specific name, such as “Condition”, “Performance” and “Stan-
 423 dard”. A Statement element matching this model would use the components Statement name
 424 and Statement text (see 6.2.4.2.2 and 6.2.4.2.3).

425 **6.2.4.2.1 Statement ID**

426 **Synopsis**

```

427     statement_id :
428         long_identifier_type,
429         // Conditionally mandatory (see 6.2.4.2)
430         // Occurs 0 or 1 times

```

431 **Description**

432 This conditionally mandatory data element is a label for the statement. This label shall be
 433 unique at least within the scope of the definition.

434 `long_identifier_type` is defined in 6.3.3.

435 NOTE—This Standard does not specify how IDs are created, assigned, or resolved.

436 **6.2.4.2.2 Statement name**

437 **Synopsis**

```

438     statement_name :
439         characterstring(iso-10646),
440         // Conditionally mandatory (see 6.2.4.2)
441         // Occurs 0 or 1 times

```

442 // SPM: 1000 characters

443 **Description**

444 This conditionally mandatory data element is a name for the statement. This name shall be
 445 unique at least within the scope of the definition. Examples: “Condition”, “Action”, “Stan-
 446 dard”, “Outcome”, “Criteria”. The characters in the string shall belong to the repertoire of
 447 ISO/IEC 10646.

448 NOTE—This Standard does not specify how names are created, assigned, or resolved.

449 **6.2.4.2.3 Statement text**

450 **Synopsis**

```
451 statement_text :
452     bag of langstring_type(1000),
453     // Conditionally mandatory (see 6.2.4.2)
454     // Occurs 0 or 1 times
455     // SPM: 20 instances of langstring_type in the bag
456     // The parameter value is the SPM for the number of
457     // of characters in the string element of the
458     // langstring_type
```

459 **Description**

460 This conditionally mandatory data element is an unstructured, textual description of those as-
 461 pects of the RCD referred to by the Statement name element (see 6.2.4.2.2). Example: “Given
 462 a set of integer numbers in the range 1 to 49.”

463 NOTE—This data element may be expressed in multiple languages. Each translation is
 464 represented by an instantiation of langstring_type

465 **6.2.4.2.4 Statement token**

466 **Synopsis**

```
467 statement_token :
468     vocabulary_type,
469     // Conditionally mandatory (see 6.2.4.2)
470     // Occurs 0 or 1 times
471
472     vocabulary_type = record
473     (
474         source :
475             characterstring(iso-10646),
476             // Mandatory
477             // Occurs 1 time
478             // SPM: 1000 characters
479         value :
480             characterstring(iso-10646),
481             // Mandatory
```

```

482         // Occurs 1 time
483         // SPM: 1000 characters
484     ),

```

485 Description

486 This conditionally mandatory data element consists of a vocabulary token, along with an identifier of the token's source. This allows the use of controlled terms (vocabularies) instead of, 487 or along with, free-form statement text (see 6.2.4.2.3). 488

489 A record of type `vocabulary_type` shall contain both a Source element and a Value element. The Source element indicates the source of the token value. The Source element may be 490 a URI that identifies a formal vocabulary definition. Example: 491 “`http://www.example.org/OSList`”. The Value element is the actual token value from a list of 492 tokens defined in the source. For example, the token might be `MRS_15`. 493

494 NOTE 1—This approach to controlled terms (vocabularies) follows that used in metadata standards such as IEEE 1484.12.1–2002. In this Standard, the vocabulary token is just a string value 495 associated with the specification of a source. It is assumed that the source defines a context in 496 which the value is meaningful. The value does not have to be a human-language word and does 497 not have to be meaningful out of context. The source typically defines the meaning of the token, 498 either by reference to a standard or by the fact that the data in the source element is a URI for a 499 human- or machine-readable description of the vocabulary tokens. 500

501 NOTE 2—This Standard does not define what a source is, only that the source has an identifier. 502 For example, a source may be another standard, a policy document, or a formal vocabulary.

503 NOTE 3—This Standard does not specify how vocabularies are created, assigned, or resolved.

504 6.2.5 Metadata

505 Synopsis

```

506     metadata :
507         metadata_type,
508         // Optional
509         // Occurs 0 or one times
510
511     metadata_type = record
512     (
513         rcd_schema :
514             characterstring(iso-10646),
515             // Optional, defined default value
516             // Occurs 0 or 1 times
517             // SPM: 1000 characters
518         rcd_schema_version :
519             characterstring(iso-10646),
520             // Optional, defined default value
521             // Occurs 0 or 1 times
522             // SPM: 1000 characters

```

```

523     additional_metadata :
524         any_type,
525         // Optional
526         // Occurs 0 or more times
527         // SPM: 10 of instances of additional_metadata
528         // in a metadata_type record
529     ),

```

530 **Description**

531 This optional data element consists of embedded metadata about the RCD. This data element
 532 does not preclude the use of external metadata about the RCD. Such external metadata are not
 533 defined by this Standard.

534 `any_type` is defined in 6.3.1.

535 NOTE—Application profiles may specify additional metadata requirements.

536 **6.2.5.1 RCD schema**

537 **Synopsis**

```

538     rcd_schema :
539         characterstring(iso-10646),
540         // Optional, defined default value
541         // Occurs 0 or 1 times
542         // SPM: 1000 characters

```

543 **Description**

544 This optional data element is a label for the schema that defines and controls the RCD in-
 545 stance. The characters in the string shall belong to the repertoire of ISO/IEC 10646. If this data
 546 element is omitted from the RCD instance, then a value of “iee.org/1484.20.1/2007” shall be
 547 assumed.

548 NOTE 1—Different values may be used to indicate that RCD instances conform to particular ap-
 549 plication profiles but should not be used to replicate the purpose of other elements such as Model
 550 source (see 6.2.4.1).

551 NOTE 2—This data element is not a label for the schema of the embedded metadata defined in
 552 6.2.5.3. Every instance of embedded metadata, if any, should include its own schema description
 553 or identifier.

554 **6.2.5.2 RCD schema version**

555 **Synopsis**

```

556     rcd_schema_version :
557         characterstring(iso-10646),
558         // Optional

```

559 // Occurs 0 or 1 times
 560 // SPM: 1000 characters

561 **Description**

562 This optional data element describes the version of the schema described by the RCD schema
 563 element (see 6.2.5.1). The characters in the string shall belong to the repertoire of ISO/IEC
 564 10646. If this element is omitted then a value of “1.0” shall be assumed.

565 **6.2.5.3 Additional metadata**

566 **Synopsis**

```
567       additional_metadata :
568           any_type,
569                // Optional
570                // Occurs 0 or more times
571
```

572 **Description**

573 Zero or more instances of additional, embedded metadata that describe the RCD instance may
 574 be included in the metadata element. If additional metadata are present, the actual type of each
 575 element shall be defined by an application profile. An implementation shall accept any addi-
 576 tional metadata record, but it is not required to interpret such additional metadata records.

577 any_type is defined in 6.3.1.

578 NOTE 1—If only one additional metadata record is included, the record should conform to IEEE
 579 1484.12.1–2002. Useful metadata defined in IEEE 1484.12.1–2002 include additional identifica-
 580 tion as an entry in one or more catalogs, information about the author, the creation date, and so
 581 on. The IEEE 1484.12.1–2002 Relation element may be used to relate a competency definition to
 582 a prior version of the definition. One or more IEEE 1484.12.1–2002 Classification elements may
 583 be used to indicate where this particular definition fits in a taxonomy of competencies.

584 NOTE 2—A particular binding specification or application profile may impose additional restric-
 585 tions or requirements.

586 NOTE 3—Each additional metadata record should contain information about the name and ver-
 587 sion of its schema. For example, in IEEE 1484.12.1–2002 metadata records, this is specified in
 588 the meta-metadata element of the metadata record.

589 **6.3 Auxiliary data types**

590 The data types described in 6.3.1 – 6.3.3 are used in conjunction with the data elements de-
 591 scribed in 6.2.

592 6.3.1 Any type

593 Synopsis

```
594     type any_type = (unspecified);
```

595 Description

596 This data type represents any type not specified in this Standard. This Standard does not re-
597 quire an implementation to process data elements of this type when encountered in an RCD
598 instance.

599 NOTE—If implementations specify or require data elements for which the type is defined in this
600 Standard as `any_type`, the implementations should provide the means to interpret and validate
601 the implementation-specific data. For example, an implementation that uses an XML binding
602 should include a valid XML schema that can be referenced in RCD instances bound in XML doc-
603 uments, and the schema should be documented.

604 6.3.2 LangString type

605 Synopsis

```
606     type langstring_type(length) = record
607     (
608         language :
609             characterstring(iso-10646),
610             // Mandatory
611             // Occurs 1 time
612             // SPM: 250 characters
613         string :
614             characterstring(iso-10646),
615             // Mandatory
616             // Occurs 1 time
617             // SPM: the length parameter
618     );
```

619 Description

620 This data type consists of a language specification for a localized string and the string itself.

621 Examples

622 The following are two examples of localized strings: "Information Technology" in
623 French and "localization" in British English.

```
624     "fr", "Technologies de l'information"
625     "en-GB", "localisation"
```

626 **6.3.2.1 Language**

627 **Synopsis**

```
628     language :
629         characterstring(iso-10646) ,
630         // Mandatory
631         // Occurs 1 time
632         // SPM: 250 characters
```

633 **Description**

634 This data element shall be a character string consisting of a required language code followed
635 by multiple, optional, hyphen-prefixed subcodes.

636 The following constraints apply to the language code part of the character string :

- 637 – Two-letter codes are defined by ISO 639–1.
- 638 – Three-letter codes are defined by ISO 639–2.
- 639 – The value prefix "i" is reserved for registrations defined by the Internet As-
640 signed Numbers Authority (IANA).
- 641 – The value prefix "x" is reserved for private use.

642 The following constraints apply to the first subcode part of the character string :

- 643 – Two-letter subcodes are ISO 3166–1 alpha-2 country codes.
- 644 – Subcodes of from three to eight letters are registered with IANA.

645 Constraints for additional subcodes are unspecified.

646 The character string shall conform to the syntax defined for a valid language code by the XML
647 Schema derived data type language (see W3C Recommendation XML Schema, Part 2 [28
648 October 2004]).

649 ISO 639–2 specifies two code sets, one for bibliographic applications (ISO 639–2/B) and one
650 for terminology applications (ISO 639–2/T). Either code set may be used.

651 NOTE 1—The language code is normally given in lower case and the subcodes (if any) in upper
652 case. However, the values are case insensitive.

653 NOTE 2—The syntax defined by the XML Schema derived data type language does not en-
654 force all constraints on this language code.

655 **Examples**

```
656     "en-GB"
657     "de"
658     "fr-CA"
659     "it"
660     "i-bnn" (IANA Bunun)
```

661 **6.3.2.2 String**662 **Synopsis**

```

663     string :
664         characterstring(iso-10646),
665         // Mandatory
666         // Occurs 1 time
667         // SPM: The length parameter

```

668 **Description**

669 This data element contains the text of the localized string. The characters in the string shall
670 belong to the repertoire of ISO/IEC 10646.

671 **6.3.3 Long identifier type**672 **Synopsis**

```

673     type long_identifier_type = record
674     (
675         catalog: characterstring(iso-10646),
676         // Mandatory
677         // Occurs 1 time
678         // SPM: 4000 characters
679         entry: characterstring(iso-10646),
680         // Mandatory
681         // Occurs 1 time
682         // SPM: 4000 characters
683     );

```

684 **Description**

685 This data type is an identifier (a label) that is intended to be unique among all contexts in
686 which the identifier may be used. The Catalog element is the name or designator of the identi-
687 fication or cataloging scheme for this entry, in other words, a namespace-scheme. The Entry
688 element is the value of the identifier within the identification or cataloging scheme that desig-
689 nates or identifies this RCD, in other words, a namespace-specific string. The characters in the
690 strings shall belong to the repertoire of ISO/IEC 10646.

691 Values for `long_identifier_type` as a whole shall conform to the syntax for URIs as de-
692 fined by IETF RFC 3986:2005. The catalog and entry values may be concatenated as a single
693 character string in an application profile or binding. If the catalog and entry values are conca-
694 tenated, the resulting character string shall be a globally unique identifier in the form of a URI.
695 The identifier shall be unique across all systems that store or process the RCD. The total
696 length of a concatenated identifier shall be no longer than 4000 characters.

697 NOTE—This Standard does not specify how to constrain the length of the Catalog and Entry
698 elements in order to limit the concatenated identifier to 4000 characters. A binding may require

699 one or more characters for concatenation of the strings contained in catalog and entry. Concatena-
700 tion into a URI may require at least seven additional characters. RCD instances should allow for
701 the possibility that the Catalog and Entry elements may be split and concatenated if the RCD
702 record is transformed for use a different binding. Therefore, in practice the total length of the Cat-
703 alog and Entry elements, exclusive of concatenation characters, should be constrained to at most
704 3990 characters.

705

706 **Annex A**

707 (informative)

708 **Bibliography**

709 [B1] IEEE 100, The Authoritative Dictionary of IEEE Standards Terms, Seventh Edition.

710 [B2] IMS Specification (25 October 2002), IMS Reusable Definition of Competency or Edu-
711 cational Objective, Version 1.0.

712 [B3] ISO/IEC 646, Information technology – ISO 7-bit coded character set for information
713 interchange.

714 [B4] ISO/IEC 8859–1, Information Technology — 8-bit single-byte coded graphic character
715 sets — Part 1: Latin alphabet No. 1

716 [B5] The Internet Society (November 2003) [RFC 3629](#), UTF-8, a transformation format of
717 ISO 10646.

718 [B6] The Internet Society (February 2000) [RFC 2781](#), UTF-16, an encoding of ISO 10646.

719 [B7] The Unicode Consortium, *The Unicode Standard, Version 4.0*, Boston, MA, Addison-
720 Wesley Developers Press, 2003.

721 *Reference [B7] describes UTF-32.*

722

723 **Annex B**

724 (informative)

725 **Conceptual overview**

726 **B.1 Objectives**

727 This Standard is intended to satisfy the following objectives:

- 728 – Provide a data model for competency definition records that can be shared or
729 reused among compatible systems.
- 730 – Reconcile various existing and emerging data models into a widely acceptable
731 model.
- 732 – Provide a way to identify the type and precision of a competency definition.
- 733 – Provide a unique identifier as the means to unambiguously reference an RCD
734 regardless of the application in which the competency definition is stored,
735 found, retrieved, or used. For example, metadata that describe learning content
736 may contain references to one or more competency definition records that de-
737 scribe learning objectives for the content.
- 738 – Provide a data model for additional information about a competency defini-
739 tion, such as a title, description, and source, that is compatible with other
740 emerging metadata standards.
- 741 – Recommend metadata as one of the methods that may be used to express how
742 competency definitions are semantically related.

743 This Standard also addresses the following needs:

- 744 – A common data model that allows the building of various competency mod-
745 els, hierarchies, and maps. However, the definitions of such competency mod-
746 els, hierarchies, and maps are outside of the scope of this Standard.
- 747 – A standard method that allows persistent, long-lived competency definitions
748 to be created, shared among systems, and maintained.
- 749 – A standard method by which competency definitions can be uniquely identi-
750 fied among conforming systems and repositories.
- 751 – A common data model for the metadata that give an RCD its value in a reuse
752 environment, such as the source of the competency definition, validation in-
753 formation, and other meta information useful to locate an RCD in a repository
754 or collection.

755 **B.2 Functional overview**

756 This Standard defines a data model for describing, referencing, and sharing competency defi-
757 nitions, primarily in the context of online and distributed learning. The data model provides a

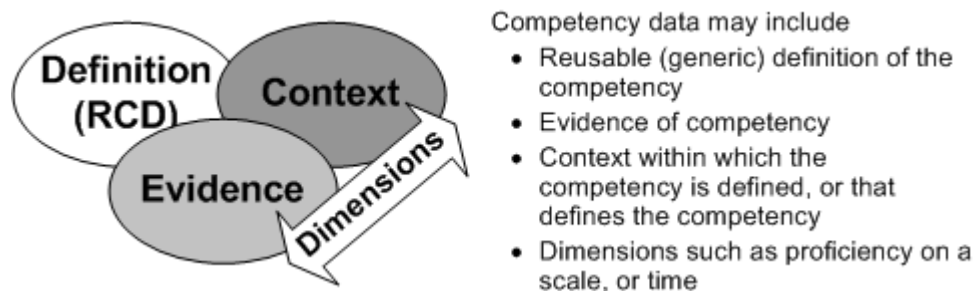
758 formal representation of the key characteristics of a competency, independently of its use in
 759 any particular context. It enables interoperability among systems that deal with competency
 760 information by providing a means for them to refer to common definitions with common
 761 meanings.

762 The core information in an RCD is an unstructured textual definition of the competency that
 763 can be referenced through a globally unique identifier. This definition may be refined using a
 764 user-defined model of the structure of a competency.

765 This Standard provides a means to capture common understandings of competencies that ap-
 766 pear as part of learning and career plans, as learning prerequisites, or as learning outcomes.
 767 The data model in this Standard can be used to share these definitions among learning sys-
 768 tems, human resource systems, learning content, competency or skills repositories, and other
 769 relevant systems. This Standard provides unique references to competency definitions for in-
 770 clusion in other data models, such as personal competency profiles.

771 RCD instances that conform to this Standard are intended for interchange by machines, but
 772 most of the information they contain is intended for human interpretation.

773 This Standard does not address the aggregation of smaller competencies into larger competen-
 774 cies (e.g., “throws” plus “catches” equals “plays ball”) nor does it address how competencies
 775 are to be assessed, validated, certified, recorded, or used as part of a process, such as instruc-
 776 tional design or knowledge management. It also does not specify how records of competencies
 777 associated with an individual are structured, stored, or shared. Figure B.1 shows how an RCD
 778 integrates with competency data.



780 **Figure B.1—RCD integration with competency data**

781 **B.3 Data model overview**

782 The data model is minimal and extensible. It is neutral with regard to models of and uses of
 783 competencies. Competencies are defined and structured in many ways in different application
 784 domains. This Standard allows implementers to share information according to the models
 785 they use. Semantics can be “tightened” or “loosened” in the data itself, while conserving the
 786 same data model regardless of how strictly a particular organization or institution requires the
 787 data to be formulated.

788 The data model contains the following mandatory elements:

- 789 – **Identifier:** A globally unique label that identifies the RCD. This identifier
790 uses the same data elements as the identifier element defined in IEEE
791 1484.12.1–2002 and consists of two subelements, Catalog and Entry. The
792 identifier is sufficient to reference the competency in any system that stores or
793 process the RCD.
- 794 – **Title:** A text label for the RCD. This is a human-readable name. While the
795 identifier provides the definitive reference to the RCD, it is typically unintel-
796 ligible. The title provides a convenient, readable alternative. The title may be
797 repeated in multiple languages.

798 The data model contains the following optional elements:

- 799 – **Description:** A human-readable description of the RCD. This is an unstruc-
800 tured character string meant to be interpretable by humans, only. The descrip-
801 tion may be repeated in multiple languages.
- 802 – **Definition:** A structured description that provides a more complete definition
803 of the RCD than the free-form description expressed in the title and descrip-
804 tion, usually using attributes taken from a specific model of how a competen-
805 cy should be structured or defined. Typically, such models define a
806 competency in terms of “statement, conditions, criteria,” “proficiency, criteria,
807 indicators,” “standards, performance indicators, outcomes,” “abilities, basic
808 skills, content, process,” and similar sets of statements.
- 809 – **Metadata:** Embedded metadata that further describe the RCD by, for exam-
810 ple, identifying the author and publication date of the RCD or by specifying
811 known relations to other RCDs.

812 Extensibility can be achieved by defining a specific model structure within the Definition ele-
813 ment (6.2.4) or by including elements defined by IEEE 1484.12.1–2002, Clause 6 in the Me-
814 tadata element (6.2.5). In addition, implementers may create additional data models that
815 include or reference the RCD data model or RCD instances. Such implementation-specific
816 data models may be used to augment the data model in this Standard to support different ap-
817 plications in various application domains.

818 NOTE—The identifier, not the title, is used to distinguish between RCDs, because different
819 RCDs may coincidentally use the same title while referring to different competencies.

820 **B.4 Taxonomies of reusable competency definitions**

821 This Standard is intended to meet the simple need of referencing and cataloging a competen-
822 cy, not classifying it. Nonetheless, an implementation might want to include relation and clas-
823 sification information, which can be done by embedding additional metadata as specified in
824 6.2.5.3. Instances of RCDs also can be referenced by the nodes in a tree or other structure
825 representing a taxonomy or ontology of competencies.

826

827 **Annex C**

828 (informative)

829 **An example XML binding schema**

830 This Standard does not define any specific binding for the data model. However, related stan-
831 dards may reference this Standard and define bindings.

832 The example in Figure C.1 illustrates existing practice using an XML schema defined by the
833 IMS Global Learning Consortium [B2].

834 NOTE 1—The example schema uses “rdceo” as the label for the root data element. It is not re-
835 quired that an implementation of this Standard use the same labels for data-element or type labels
836 as those used in the example below, as long as the elements and types themselves are semantical-
837 ly equivalent.

838 NOTE 2—The example schema conforms to this Standard. However, it allows extensions in a
839 form different from that recommended by this Standard.

840

```

841 <?xml version="1.0" encoding="UTF-8"?>
842 <xs:schema
843 targetNamespace="http://www.imsglobal.org/xsd/imsrdceo_rootv1p0"
844 xmlns:xs="http://www.w3.org/2001/XMLSchema"
845 xmlns="http://www.imsglobal.org/xsd/imsrdceo_rootv1p0"
846 elementFormDefault="qualified"
847 attributeFormDefault="unqualified">
848   <xs:group name="extelement">
849     <xs:annotation>
850       <xs:documentation>extension mechanism for elements</xs:documentation>
851     </xs:annotation>
852     <xs:sequence>
853       <xs:any namespace="##other" processContents="strict" maxOc-
854 curs="unbounded"/>
855     </xs:sequence>
856   </xs:group>
857   <xs:element name="rdceo">
858     <xs:annotation>
859       <xs:documentation>A single definition of a competence, educational objec-
860 tive etc</xs:documentation>
861     </xs:annotation>
862     <xs:complexType>
863       <xs:sequence>
864         <xs:element ref="identifier" minOccurs="1" maxOccurs="1"/>
865         <xs:element ref="title"/>
866         <xs:element ref="description" minOccurs="0"/>
867         <xs:element ref="definition" minOccurs="0" maxOccurs="unbounded"/>
868         <xs:element ref="metadata" minOccurs="0"/>
869         <xs:sequence minOccurs="0">
870           <xs:group ref="extelement"/>
871         </xs:sequence>
872       </xs:sequence>

```

```

873     <xs:anyAttribute namespace="##other" processContents="strict"/>
874   </xs:complexType>
875 </xs:element>
876 <xs:element name="langstring">
877   <xs:annotation>
878     <xs:documentation>A string in a human language</xs:documentation>
879   </xs:annotation>
880   <xs:complexType>
881     <xs:simpleContent>
882       <xs:extension base="xs:string">
883         <xs:anyAttribute namespace="##other" processContents="strict"/>
884       </xs:extension>
885     </xs:simpleContent>
886   </xs:complexType>
887 </xs:element>
888 <xs:element name="title">
889   <xs:annotation>
890     <xs:documentation>A title for the definition</xs:documentation>
891   </xs:annotation>
892   <xs:complexType>
893     <xs:sequence>
894       <xs:element ref="langstring" maxOccurs="unbounded"/>
895       <xs:sequence minOccurs="0">
896         <xs:group ref="extelement"/>
897       </xs:sequence>
898     </xs:sequence>
899     <xs:anyAttribute namespace="##other" processContents="strict"/>
900   </xs:complexType>
901 </xs:element>
902 <xs:element name="identifier">
903   <xs:annotation>
904     <xs:documentation>Catenated form of the identifier of a
905 RDCEO</xs:documentation>
906   </xs:annotation>
907   <xs:complexType>
908     <xs:simpleContent>
909       <xs:extension base="xs:anyURI">
910         <xs:anyAttribute namespace="##other" processContents="strict"/>
911       </xs:extension>
912     </xs:simpleContent>
913   </xs:complexType>
914 </xs:element>
915 <xs:element name="description">
916   <xs:annotation>
917     <xs:documentation>A description for the definition</xs:documentation>
918   </xs:annotation>
919   <xs:complexType>
920     <xs:sequence>
921       <xs:element ref="langstring" maxOccurs="unbounded"/>
922       <xs:sequence minOccurs="0">
923         <xs:group ref="extelement"/>
924       </xs:sequence>
925     </xs:sequence>
926     <xs:anyAttribute namespace="##other" processContents="strict"/>
927   </xs:complexType>
928 </xs:element>
929 <xs:element name="definition">
930   <xs:annotation>
931     <xs:documentation>A structured form of the definition</xs:documentation>
932   </xs:annotation>
933   <xs:complexType>
934     <xs:sequence>
935       <xs:element ref="model" minOccurs="0"/>

```

```

936     <xs:element ref="statement" maxOccurs="unbounded"/>
937     <xs:sequence minOccurs="0">
938         <xs:group ref="extelement"/>
939     </xs:sequence>
940 </xs:sequence>
941     <xs:anyAttribute namespace="##other" processContents="strict"/>
942 </xs:complexType>
943 </xs:element>
944 <xs:element name="model">
945     <xs:annotation>
946         <xs:documentation>The model identification for the structured defini-
947 tion</xs:documentation>
948     </xs:annotation>
949     <xs:complexType>
950         <xs:simpleContent>
951             <xs:extension base="xs:string">
952                 <xs:anyAttribute namespace="##other" processContents="strict"/>
953             </xs:extension>
954         </xs:simpleContent>
955     </xs:complexType>
956 </xs:element>
957 <xs:element name="statement">
958     <xs:annotation>
959         <xs:documentation>A component part of a structured defini-
960 tion</xs:documentation>
961     </xs:annotation>
962     <xs:complexType>
963         <xs:sequence>
964             <xs:choice>
965                 <xs:element ref="statementtext"/>
966                 <xs:element ref="statementtoken"/>
967             </xs:choice>
968             <xs:sequence minOccurs="0">
969                 <xs:group ref="extelement"/>
970             </xs:sequence>
971         </xs:sequence>
972         <xs:attribute name="statementid" type="xs:ID"/>
973         <xs:attribute name="statementname" type="xs:string"/>
974         <xs:anyAttribute namespace="##other" processContents="strict"/>
975     </xs:complexType>
976 </xs:element>
977 <xs:element name="statementtext">
978     <xs:annotation>
979         <xs:documentation>Used for statements with free-form
980 text</xs:documentation>
981     </xs:annotation>
982     <xs:complexType>
983         <xs:sequence>
984             <xs:element ref="langstring" maxOccurs="unbounded"/>
985             <xs:sequence minOccurs="0">
986                 <xs:group ref="extelement"/>
987             </xs:sequence>
988         </xs:sequence>
989     </xs:complexType>
990 </xs:element>
991 <xs:element name="source">
992     <xs:annotation>
993         <xs:documentation>Source identification for a vocabulary to-
994 ken</xs:documentation>
995     </xs:annotation>
996     <xs:complexType>
997         <xs:simpleContent>
998             <xs:extension base="xs:string">

```

```

999      <xs:anyAttribute namespace="##other" processContents="strict"/>
1000    </xs:extension>
1001  </xs:simpleContent>
1002 </xs:complexType>
1003 </xs:element>
1004 <xs:element name="value">
1005   <xs:annotation>
1006     <xs:documentation>A vocabulary token</xs:documentation>
1007   </xs:annotation>
1008   <xs:complexType>
1009     <xs:simpleContent>
1010       <xs:extension base="xs:string">
1011         <xs:anyAttribute namespace="##other" processContents="strict"/>
1012       </xs:extension>
1013     </xs:simpleContent>
1014   </xs:complexType>
1015 </xs:element>
1016 <xs:element name="metadata">
1017   <xs:annotation>
1018     <xs:documentation>A container for metadta</xs:documentation>
1019   </xs:annotation>
1020   <xs:complexType>
1021     <xs:sequence>
1022       <xs:element ref="rdceoschema" minOccurs="0"/>
1023       <xs:element ref="rdceoschemaversion" minOccurs="0"/>
1024       <xs:sequence minOccurs="0">
1025         <xs:group ref="extelement"/>
1026       </xs:sequence>
1027     </xs:sequence>
1028     <xs:anyAttribute namespace="##other" processContents="strict"/>
1029   </xs:complexType>
1030 </xs:element>
1031 <xs:element name="statementtoken">
1032   <xs:annotation>
1033     <xs:documentation>Used for statements with token values (vocabulary
1034 use)</xs:documentation>
1035   </xs:annotation>
1036   <xs:complexType>
1037     <xs:sequence>
1038       <xs:element ref="source"/>
1039       <xs:element ref="value"/>
1040       <xs:sequence minOccurs="0">
1041         <xs:group ref="extelement"/>
1042       </xs:sequence>
1043     </xs:sequence>
1044     <xs:anyAttribute namespace="##other" processContents="strict"/>
1045   </xs:complexType>
1046 </xs:element>
1047 <xs:element name="rdceoschema">
1048   <xs:annotation>
1049     <xs:documentation>The identity of the RDCEO schema - assumed to be IMS
1050 RDCEO if absent.</xs:documentation>
1051   </xs:annotation>
1052   <xs:complexType>
1053     <xs:simpleContent>
1054       <xs:extension base="xs:string">
1055         <xs:anyAttribute namespace="##other" processContents="strict"/>
1056       </xs:extension>
1057     </xs:simpleContent>
1058   </xs:complexType>
1059 </xs:element>
1060 <xs:element name="rdceoschemaversion">
1061   <xs:annotation>

```

```
1062     <xs:documentation>The version of the RDCEO schema - assumed to be 1.0 if
1063 absent</xs:documentation>
1064   </xs:annotation>
1065   <xs:complexType>
1066     <xs:simpleContent>
1067       <xs:extension base="xs:string">
1068         <xs:anyAttribute namespace="##other" processContents="strict"/>
1069       </xs:extension>
1070     </xs:simpleContent>
1071   </xs:complexType>
1072 </xs:element>
1073 </xs:schema>
```

1074

Figure C.1—An example schema