

1 **IEEE 1484.20.1/Draft 3**
2 **Draft Standard for Learning Technology—**
3 **Standard 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 The Institute of Electrical and Electronics Engineers, Inc.
18 Three Park Avenue, New York, NY 10016-5997, USA

19 Copyright © 2006 by the Institute of Electrical and Electronics Engineers, Inc.
20 All rights reserved. Published [date to be supplied](#). Printed in the United States of
21 America.

22

23 Three Park Avenue
24 New York, NY 10016-5997, USA

25 This document is an unapproved draft of a proposed IEEE Standard. As such, this
26 document is subject to change. USE AT YOUR OWN RISK! Because this is an un-
27 approved draft, this document must not be utilized for any conformance/compliance
28 purposes. Permission is hereby granted for IEEE Standards Committee participants
29 to reproduce this document for purposes of IEEE standardization activities only. Prior
30 to submitting this document to another standards development organization for stan-
31 dardization activities, permission must first be obtained from the Manager, Standards
32 Licensing and Contracts, IEEE Standards Activities Department. Other entities seek-
33 ing permission to reproduce this document, in whole or in part, must obtain permis-

34 sion from the Manager, Standards Licensing and Contracts, IEEE Standards De-
35 partment.

36

37 IEEE Standards Department
38 Standards Licensing and Contracts
39 445 Hoes Lane, P.O. Box 1331
40 Piscataway, NJ 08855-1331, USA

41

42 *[NOTE: Information about IEEE LTSC P1484.20 can be found at:*

43 <http://www.ieeeltsc.org/>

44 *This note will be removed upon reaching the final draft of this IEEE document.]*

45 **Introduction**

46 (This introduction is not a part of P1484.20.1, Draft Standard for Learning Technology—
47 Standard for Reusable Competency Definitions.)

48 This Standard defines a data model for describing, referencing, and sharing competency defi-
49 nitions, primarily in the context of online and distributed learning. This Standard provides a
50 way to represent formally the key characteristics of a competency, independently of its use in
51 any particular context. It enables interoperability among learning systems that deal with com-
52 petency information by providing a means for them to refer to common definitions with com-
53 mon meanings.

54 **Participants**

55 At the time this Standard was completed, the working group had the following membership:

Claude Ostyn, <i>Chair</i> Scott Lewis, <i>Technical Editor</i>		
Mike Collett	Rolf Lindner	Robby Robson
Geoffrey A. Frank	Dan Rehak	Luk Vervenne
Chad Kainz		

56 The following persons were on the balloting committee: (To be provided by IEEE editor at
57 time of publication.)

58 **Acknowledgements**

59 This Standard is based on the “IMS Reusable Definition of Competency or Educational Ob-
60 jective Specification,” Version 1.0, published on October 25, 2002 by the IMS Global Learn-
61 ing Consortium, Inc.

62 **Contents**

63	1. Overview	1
64	1.1 Scope	1
65	1.2 Purpose	1
66	2. Normative references	1
67	3. Definitions	2
68	3.1 Abbreviations and acronyms.....	3
69	4. Conformance	4
70	4.1 Shall and shall not.....	4
71	4.2 RCD instances.....	4
72	4.3 Smallest permitted maximum values.....	4
73	5. Conceptual overview (informative)	4
74	5.1 Objectives.....	5
75	5.2 Functional overview	5
76	5.3 Data model overview.....	6
77	5.4 Taxonomies of reusable competency definitions.....	7
78	6. Data model	8
79	6.1 General information.....	8
80	6.2 Reusable competency definition.....	8
81	6.2.1 Identifier.....	9
82	6.2.2 Title	10
83	6.2.3 Description.....	10
84	6.2.4 Definition	11
85	6.2.5 Metadata.....	14
86	6.3 Auxiliary data types.....	16
87	6.3.1 Any type.....	17
88	6.3.2 LangString type	17
89	6.3.3 Long identifier type	19
90	Annex A (informative) Bibliography	20
91	Annex B (informative) Sample XML Binding Schema	21
92		

93 **Draft Standard for Learning Technology—** 94 **Reusable Competency Definitions**

95 **1. Overview**

96 **1.1 Scope**

97 This Standard defines a data model for describing, referencing, and sharing competency defi-
98 nitions, primarily in the context of online and distributed learning. This Standard provides a
99 way to represent formally the key characteristics of a competency, independently of its use in
100 any particular context. It enables interoperability among learning systems that deal with com-
101 petency information by providing a means for them to refer to common definitions with com-
102 mon meanings.

103 This standard enables information about competencies to be encoded and shared. It does not
104 define whether a competency is a skill, knowledge, ability, attitude or learning outcome but
105 can be used to capture information about any of these. This Standard does not specify policies
106 regarding RCDs, such as the best practice to look for an existing definition to reuse instead of
107 inventing a new one for the same purpose.

108 **1.2 Purpose**

109 The purpose of this Standard is to publish an IEEE standard based on the existing IMS Global
110 Learning Consortium (IMS) specification for Reusable Definition of Competency or Educa-
111 tional Objective (RDCEO) [B2]¹. This standard is to be defined in such a way that imple-
112 mentations that conform to the IMS specification will be conformant to this Standard.

113 **2. Normative references**

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

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

- 117 IEEE 1484.12.1–2002: Standard for Learning Object Metadata.²
- 118 IETF RFC 2396:1998, Uniform Resource Identifiers (URI): Generic Syntax.³
- 119 ISO 639–1, Code for the representation of names of languages – Part 1: Alpha-2 code.⁴
- 120 ISO 639–2, Codes for the representation of names of languages – Part 2: Alpha-3 code.
- 121 ISO/IEC 646:1991, Information technology – ISO 7-bit coded character set for information⁵
122 interchange.
- 123 ISO 3166–1, Codes for the representation of names of countries and their subdivisions – Part
124 1: Country codes.
- 125 ISO/IEC 10646–1, Information technology – Universal Multiple-Octet Coded Character Set
126 (UCS)—Part 1: Architecture and Basic Multilingual Plane.
- 127 ISO/IEC 11404:1996, Information technology – Programming languages, their environments
128 and system software interfaces – Language-independent datatypes.
- 129 W3C Recommendation (28 October 2004), XML Schema Part 2: Datatypes, Second Edition.⁶

130 3. Definitions

- 131 For purposes of this Standard, the following terms and definitions apply. IEEE 100, *The Au-*
132 *thoritative Dictionary of IEEE Standards Terms*, Seventh Edition [B1], should be referenced
133 for terms not defined in this Clause.

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

134 **competency:** In this Standard, any form of knowledge, skill, attitude, ability, or learning ob-
135 jective that can be described in a context of learning, education or training.

136 **competency definition record:** In this Standard, an instance of a data structure that represents
137 a reusable competency definition.

138 NOTE—The term “competency” is to be interpreted in the broadest sense to include learning ob-
139 jectives (those things that are sought) and competency or competencies (those things that are
140 achieved). The term “competency” is also used to include all classes of things that someone, or
141 potentially something, can be competent in, although some communities of practice use the term
142 with nuance, for example limiting its use to skill and excluding knowledge or understanding.

143 **data type:** A property of distinct values, indicating common features of those values and op-
144 erations on those values.

145 **extended data element:** An element of a data structure that is defined outside of a standard
146 and is permitted within an instance of the data structure.

147 **LangString:** A data type that represents one or more character strings. A LangString value
148 may include multiple semantically equivalent character strings, such as translations or expres-
149 sions of a description in different languages. *See also:* **data type**.

150 **value space:** The set of values for a given data type (ISO/IEC 11404:1996).

151 NOTE—In this Standard, a value space is typically either enumerated outright or defined by ref-
152 erence to another standard or specification.

153 **3.1 Abbreviations and acronyms**

154 IMS: IMS Global Learning Consortium

155 RCD: reusable competency definition

156 RDCEO: IMS Reusable Definition of Competency or Educational Objective

157 SPM: smallest permitted maximum

158 URI: Uniform Resource Identifier

159 URN: Uniform Resource Name

160 XML: Extensible Markup Language

161 4. Conformance

162 4.1 Shall and shall not

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

165 4.2 RCD instances

166 A conforming RCD instance shall be an instance of the data model as defined in Clause 6.

167 4.3 Smallest permitted maximum values

168 For data elements that have smallest permitted maximum (SPM) values, an implementation
169 that conforms to this Standard shall accept and process at least that number of entries or char-
170 acters specified by the SPM for the element and may accept and process a larger number.
171 SPM values are defined for

- 172 – **Items with multiple values:** All applications that process RCD instances
173 shall process at least the SPM number of entries. In other words, an applica-
174 tion may impose a maximum on the number of entries it processes for a data
175 element with multiple values, but that maximum shall not be lower than the
176 SPM value.
- 177 – **Data elements with type `CharacterString` or `LangString`:** All applications
178 that process RCD instances shall process at least the SPM length for the
179 `CharacterString` value (either directly or contained in the `LangString`) of that
180 data element. In other words, an application may impose a maximum on the
181 number of characters it processes for the `CharacterString` value of that data
182 element, but that maximum shall not be lower than the SPM value for the data
183 type of the data element.

184 NOTES:

185 1—The intent is for the SPM values to cover most cases.

186 2—The meaning of “process” in this subclause depends on the nature of the application.

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

189 5. Conceptual overview (informative)

190 *This Clause is informative.*

191 5.1 Objectives

192 This Standard is intended to satisfy the following objectives:

- 193 – Provide a data model for competency definition records that can be shared or
194 reused in one or more compatible systems.
- 195 – Reconcile various existing and emerging data models into a widely acceptable
196 model.
- 197 – Provide a way to identify the type and precision of a competency definition.
- 198 – Provide a unique identifier as the means to unambiguously reference an RCD
199 regardless of the setting in which the competency definition is stored, found,
200 retrieved, or used. For example, metadata that describe learning content may
201 contain references to one or more competency definition records that describe
202 learning objectives for the content.
- 203 – Provide a data model for additional information about a competency defini-
204 tion, such as a title, description, and source, compatible with other emerging
205 learning asset metadata standards.
- 206 – Recommend metadata as one of the methods that may be used to express how
207 competency definitions are semantically related.

208 This Standard also addresses the following needs:

- 209 – A common data model that allows the building of various competency mod-
210 els, hierarchies, and maps. However, the definitions of such competency mod-
211 els, hierarchies, and maps are outside of the scope of this Standard.
- 212 – A standard method that allows persistent, long-lived competency definitions
213 to be created, shared among systems, and maintained.
- 214 – A standard method by which competency definitions can be identified as
215 globally unique among conforming systems and repositories.
- 216 – A common data model for the metadata that give an RCD its value in a reuse
217 environment, such as the source of the competency definition, validation in-
218 formation, and other meta information useful to locate an RCD in a repository
219 or collection.

220 5.2 Functional overview

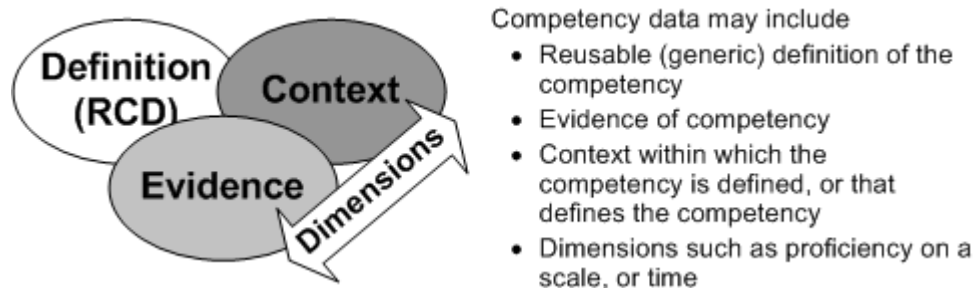
221 This Standard defines a data model for describing, referencing, and sharing competency defi-
222 nitions, primarily in the context of online and distributed learning. The data model provides a
223 formal representation of the key characteristics of a competency, independently of its use in
224 any particular context. It enables interoperability among learning systems that deal with com-
225 petency information by providing a means for them to refer to common definitions with com-
226 mon meanings.

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

230 This Standard provides a means to capture common understandings of competencies that appear as part of learning or career plan, as learning prerequisites, or as learning outcomes. The data model in this Standard can be used to share these definitions between learning systems, human resource systems, learning content, competency or skills repositories, and other relevant systems. This Standard provides unique references to competency definitions for inclusion in other data models, such as personal competency profiles.

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

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



244

245 **Figure 1—RCDs cover only a part of the competency data**

246 5.3 Data model overview

247 The data model is minimalist and extensible. It is neutral with regard to models of and uses of competencies. Competencies are defined and structured in many ways in different communities of practice. This Standard allows communities of practice to share information according to the models they use. Semantics can be “tightened” or “loosened” in the data itself, while conserving the same data model regardless of how strictly a particular organization or institution requires the data to be formulated.

253 The data model contains the following mandatory elements:

- 254 – **Identifier:** A globally unique label that identifies the RCD. This identifier uses the same data elements as the identifier element defined in IEEE 255 1484.12.1–2002, “Standard for Learning Object Metadata,”⁷ and consists of 256

⁷ For information on normative references, see Clause 2.

257 two subelements, Catalog and Entry. The identifier is sufficient to reference
258 the competency in any other system.
259 – **Title:** A text label for the RCD. This is a short, human-readable name. While
260 the identifier provides the definitive reference to the RCD, it is typically unin-
261 tellible. The title provides a convenient, readable alternative. The title may
262 be repeated in multiple languages.

263 The data model contains the following optional elements:

- 264 – **Description:** A human-readable description of the RCD. This is an unstruc-
265 tured character string meant to be interpretable by humans, only. The descrip-
266 tion may be repeated in multiple languages.
- 267 – **Definition:** A structured description that provides a more complete definition
268 of the RCD than the free-form description expressed in the title and descrip-
269 tion, usually using attributes taken from a specific model of how a compe-
270 tency should be structured or defined. Typically, such models define a
271 competency in terms of “statement, conditions, criteria,” “proficiency, criteria,
272 indicators,” “standards, performance indicators, outcomes,” “abilities, basic
273 skills, content, process,” and similar sets of statements.
- 274 – **Metadata:** Embedded metadata that further describe the RCD by, for exam-
275 ple, identifying the author and publication date of the RCD or by specifying
276 known relations to other RCDs.

277 Extensibility can be achieved by defining a specific model structure within the Definition ele-
278 ment (6.2.4) or by including elements defined by IEEE 1484.12.1–2002, Clause 6 in the
279 Metadata element (6.2.5). In addition, implementers may create additional data models that
280 include or reference the RCD data model or RCD instances. Such implementation-specific
281 data models may be used to augment the data model in this Standard to support different ap-
282 plications and communities of practice.

283 NOTE—The identifier, not the title, is used to distinguish between RCDs, because different
284 communities of practice may coincidentally define the same title.

285 **5.4 Taxonomies of reusable competency definitions**

286 This Standard is intended to meet the simple need of referencing and cataloguing a compe-
287 tency, not classifying it. Nonetheless, an implementation might want to include relation and
288 classification information, which can be done by embedding additional metadata as specified
289 in 6.2.5.3. Instances of RCDs also can be referenced by the nodes in a tree or other graph rep-
290 resenting a taxonomy or ontology of competencies.

291 6. Data model

292 6.1 General information

293 This Clause defines the data elements of an RCD. Unless otherwise noted, all components of
294 records are optional in an RCD instance.

295 NOTES:

296 1—The use of ISO/IEC 11404 notation in the synopses in 6.2 and 6.3 is for descriptive purposes
297 only. A complete implementation of the operations defined in ISO/IEC 11404 is not required for
298 conformance.

299 2—The ISO/IEC 11404 notation describes the semantics of the language-independent data types
300 across all bindings (e.g., implementation of a data type as itself, its subtypes, its subclasses, and
301 its specializations). For example, an ISO/IEC 11404 record may be implemented as an SQL table
302 row, or as an Extensible Markup Language (XML) complexType; an ISO/IEC 11404
303 characterstring may be implemented in an encoding (ISO 646, ASCII, ISO 8859-1, UTF-8, UTF-
304 16, UTF-32, etc.) that supports the repertoire specified in the parameter to characterstring data
305 type.

306 3—All examples in 6.2 and 6.3 are informative and do not endorse any particular binding.

307 4—The following language-independent data types used in this Standard are defined in ISO/IEC
308 11404: bag, characterstring, and record.

309 5—The labels for data elements and data types in the synopses in 6.2 and 6.3 are for reference,
310 only. An implementation is not required to use the same labels, as long as the data elements and
311 data types are semantically equivalent.

312 6—This Standard does not define a specific extension mechanism for the data model. Implemen-
313 ters may define bindings that allow additional elements, or create additional data models for com-
314 petency data. Such models may be used to augment this model to support different communities
315 of practice.

316 6.2 Reusable competency definition

317 Synopsis

```
318 reusable_competency_definition :
319 record
320 (
321     identifier :
322         long_identifier_type,
323     title :
324         bag of langstring_type(1000),
325         // SPM: 20 instances of langstring_type in the bag
326         // the parameter value is the SPM for the langstring
```

```

327     description :
328         langstring_type(2000),
329         // the parameter value is the SPM
330     definition :
331         record
332         (
333             model_source :
334                 characterstring(iso-10646-1),
335                 // SPM: 1000 characters
336             statements :
337                 bag of statement_type,
338                 // SPM: 10 statement records in the bag
339         ),
340     metadata :
341         record
342         (
343             rcd_schema :
344                 characterstring(iso-10646-1),
345                 // SPM: 4000 characters
346             rcd_schema_version :
347                 characterstring(iso-10646-1),
348                 // SPM: 1000 characters
349             additional_metadata :
350                 bag of any_type,
351                 // SPM: 10 of any_type in the bag
352         ),
353     )

```

354 **Description**

355 The components of `reusable_competency_definition` are defined in 6.2.1 – 6.2.5.
 356 Identifier and Title are mandatory and shall be included in RCD instances. Depending on the
 357 implementation, an instance of `reusable_competency_definition` shall include zero or
 358 more of the other defined components.

359 All elements in this data model are intrinsically unordered. The order of the elements in the
 360 data model synopses and the order of the values in a list of values are not significant. For ex-
 361 ample, if the model includes three statements, their order is not significant. They may appear
 362 in any order without loss of information.

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

365 **6.2.1 Identifier**

366 **Synopsis**

```

367     identifier :
368         long_identifier_type,

```

369 Description

370 This data element is a globally unique label that identifies the RCD. This data element is suffi-
371 cient to reference the RCD in any conforming system.

372 Subclause 6.3.3 defines `long_identifier_type`.

373 NOTE—This data element uses the same subelements as the identifier element defined in IEEE
374 1484.12.1–2002 and consists of two subelements, Catalogue and Entry.

375 6.2.2 Title**376 Synopsis**

```
377 title : bag of langstring_type(1000),  
378 // SPM: 20 instances of langstring_type in the bag  
379 // the parameter value is the SPM for the langstring
```

380 Description

381 This data element is a single, mandatory, text label for the RCD. The label is a short, human-
382 readable name for the RCD. Because different communities of practice may coincidentally
383 define the same title, the identifier, not the title, shall be used to distinguish among RCDs.

384 Subclause 6.3.2 defines `langstring_type`.

385 NOTES:

386 1—This data element may be repeated in multiple languages.

387 2— While the Identifier element (see 6.2.1) provides the definitive reference to the definition, it
388 is typically unintelligible. The Title element provides a convenient, alternative, readable form.
389 Examples: “English proficiency”, “Schmiblick failure diagnostic level 4”, “Demonstrates conflict
390 resolution skills”.

391 6.2.3 Description**392 Synopsis**

```
393 description : bag of langstring_type(2000),  
394 // SPM: 20 instances of langstring_type in the bag  
395 // the parameter value is the SPM for the langstring
```

396 Description

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

399 Subclause 6.3.2 defines `langstring_type`.

400 NOTES:

401 1—This data element may be repeated in multiple languages.

402 2—This data element is typically more explicative than the Title element (see 6.2.2). Examples:
 403 “Proficiency in written and spoken English and use of English for meaningful oral or written
 404 expression”, “Performance of level 4 diagnostic as specified in IETM #SCMBLK007”.

405 6.2.4 Definition

406 Synopsis

```

407     definition :
408         record
409         (
410             model_source :
411                 characterstring(iso-10646-1),
412                 // SPM: 1000 characters
413             statements :
414                 bag of statement_type,
415                 // SPM: 10 statement records in the bag
416         ),
```

417 Description

418 This data element is an optional, structured description that provides a more complete defini-
 419 tion of the competency, usually using attributes taken from a specific model of how a compe-
 420 tency should be structured or defined. This data element shall contain zero or more model
 421 sources and at least one statement. It may contain multiple statements.

422 NOTES:

423 1—Typically, the models that underlie this data element define competencies in terms of
 424 “statement, conditions, criteria”, “proficiency, criteria, indicators”, “standards, performance
 425 indicators, outcomes”, “abilities, basic skills, content, process”, and similar sets of statements.

426 2—This data element provides a structure for including an arbitrary collection of statements that
 427 determine a competency. The author of an RCD is free to use this data element in any way that
 428 best describes the competency.

429 **6.2.4.1 Model source**430 **Synopsis**

```

431     model_source :
432         characterstring(iso-10646-1),
433         // SPM: 1000 characters

```

434 **Description**

435 This data element is the source of the model used for the competency definition. The charac-
 436 ters in the string shall belong to the repertoire of ISO/IEC 10646–1:2000, as allowed by IETF
 437 RFC 2396.

438 NOTE—The value of this data element should be specific enough to avoid conflict with other
 439 source names; therefore, it is recommended that the value be a uniform resource identifier (URI).
 440 If the value of this data element is a URI, it may point to an actual document that defines the
 441 source formally. However, this is not required. Examples: “3-part-learning-objective”,
 442 “http://foo.edu/ref/los.xml”.

443 **6.2.4.2 Statements**444 **Synopsis**

```

445     statements :
446         bag of statement_type,
447         // SPM: 10 statement records in the bag
448
449     statement_type = record
450         // SPM: 10 statement records in the bag
451     (
452         statement_id :
453             long_identifier_type,
454         statement_name :
455             characterstring(1000),
456         statement_text : bag of langstring_type(1000),
457             // SPM: 20 instances of langstring_type in the bag
458             // the parameter value is the SPM for the langstring
459         statement_token :
460             vocabulary_type,
461     ),

```

462 **Description**

463 Each record in this data element is a description of a single characteristic of a Definition ele-
 464 ment (see 6.2.4). A record of type `statement_type` shall contain one or more elements.

465 Although no specific component of a statements element is required, the element shall contain
 466 at least one of these components. For example, a particular learning-objective model might
 467 require a list of specific statement strings, each of which has a specific name, such as

468 “Condition”, “Performance” and “Standard”. A statement element matching this model would
469 use the components Statement name and Statement text (see 6.2.4.2.2 and 6.2.4.2.3).

470 **6.2.4.2.1 Statement ID**

471 **Synopsis**

```
472     statement_id :  
473         long_identifier_type,
```

474 **Description**

475 This data element is a label for the statement. This label shall be unique at least within the
476 scope of the definition.

477 Subclause 6.3.3 defines `long_identifier_type`.

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

479 **6.2.4.2.2 Statement name**

480 **Synopsis**

```
481     statement_name :  
482         characterstring,
```

483 **Description**

484 This data element is a name for the statement. This name shall be unique at least within the
485 scope of the definition. Examples: “Condition”, “Action”, “Standard”, “Outcome”, “Criteria”.

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

487 **6.2.4.2.3 Statement text**

488 **Synopsis**

```
489     statement_text : bag of langstring_type(1000),  
490         // SPM: 20 instances of langstring_type in the bag  
491         // the parameter value is the SPM for the langstring
```

492 **Description**

493 This data element is an unstructured, textual description of those aspects of the RCD referred
494 to by the statement name element. Example: “Given a set of integer numbers in the range 1 to
495 49.”.

496 NOTE—This data element may be repeated in multiple languages.

497 **6.2.4.2.4 Statement token**498 **Synopsis**

```

499     statement_token :
500         vocabulary_type ,
501
502     vocabulary_type = record
503     (
504         source :
505             characterstring(iso-10646-1) ,
506             // SPM: 1000 characters
507         value :
508             characterstring(iso-10646-1) ,
509             // SPM: 1000 characters
510     ) ,

```

511 **Description**

512 This data element consists of a vocabulary token, along with an identifier of its source. This
 513 allows the use of controlled terms (vocabularies) instead of, or along with, free-form statement
 514 text (see 6.2.4.2.3).

515 The source element indicates the source of the token value. The source element may be a URI
 516 that identifies a formal vocabulary definition. Example:
 517 “<http://www.vocabularies.org/OSList>”.

518 The value element is the actual token value from a list of tokens defined in the source. For ex-
 519 ample, the token might be MRS_15.

520 **NOTES:**

521 1—This approach to controlled terms (vocabularies) follows that used in metadata standards such
 522 as IEEE 1484.12.1–2002. In this Standard, the token is just a string; it does not have to be a hu-
 523 man-language word and does not have to be meaningful. The source typically defines the mean-
 524 ing of the token, either by reference to a standard or by the fact that the data in the source element
 525 is a URL to a human- or machine-readable description of the vocabulary tokens.

526 2—This Standard does not define what a source is, only that the source has an identifier. For ex-
 527 ample, a source may be another standard, a policy document, or a formal vocabulary.

528 3—This Standard does not specify how vocabularies are created, assigned, or resolved.

529 **6.2.5 Metadata**530 **Synopsis**

```

531     metadata :
532         record
533     (
534         rcd_schema :

```

```

535         characterstring(iso-10646-1),
536         // SPM: 1000 characters
537     rcd_schema_version :
538         characterstring(iso-10646-1),
539         // SPM: 1000 characters
540     additional_metadata :
541         bag of any_type,
542         // SPM: 10 of any type in the bag
543     ),

```

544 **Description**

545 This data element consists of embedded metadata about the RCD. This data element does not
 546 preclude the use of external metadata about the RCD. Such external metadata are not defined
 547 by this Standard.

548 Subclause 6.3.1 defines any_type.

549 NOTE—Application profiles may specify additional metadata requirements.

550 **6.2.5.1 RCD schema**

551 **Synopsis**

```

552     rcd_schema :
553         characterstring(iso-10646-1),
554         // SPM: 1000 characters

```

555 **Description**

556 This data element is a label for the schema that defines and controls the RCD instance.

557 NOTES:

558 1—If this data element is omitted then a value of “ieee.org/1484.20.1/2006” should be assumed.
 559 Different values may be used to signal application profiles but should not be used to replicate the
 560 purpose of other elements such as model source (see 6.2.4.1).

561 2—This data element is not a label for the schema of the embedded metadata defined in 6.2.5.3.
 562 Every instance of embedded metadata, if any, should include its own schema description or iden-
 563 tifier.

564 **6.2.5.2 RCD schema version**

565 **Synopsis**

```

566     rcd_schema_version :
567         characterstring(iso-10646-1),
568         // SPM: 1000 characters

```

569 **Description**

570 This data element describes the version of the schema described by the Rcd schema element
571 (see 6.2.5.1).

572 NOTE—If this element is omitted then a value of “1.0” should be assumed.

573 **6.2.5.3 Additional metadata**

574 **Synopsis**

```
575     additional_metadata :
576         bag of any_type,
577         // SPM: 10 of any type in the bag
```

578 **Description**

579 This data element contains optional, additional, embedded metadata describing the RCD. If
580 additional metadata are present, the actual type shall be defined by an application profile.

581 If an additional metadata record is included, the record should conform to IEEE 1484.12.1–
582 2002.

583 More than one additional metadata record is allowed in the bag, but if the bag contains more
584 than one record, each record should conform to a different metadata specification. An imple-
585 mentation shall accept any metadata record that it cannot interpret, but it is not required to in-
586 terpret such metadata records.

587 NOTES:

588 1—Useful metadata defined in IEEE 1484.12.1–2002 include additional identification as an entry
589 in one or more catalogues, information about the author, the creation date, and so on. The IEEE
590 1484.12.1–2002 Relation element may be used to relate a definition to a prior version of the defi-
591 nition, and one or more IEEE 1484.12.1–2002 Classification elements may be used to indicate
592 where this particular definition fits in a taxonomy of competencies.

593 2—A particular binding specification or application profile may impose additional restrictions or
594 requirements.

595 3—Each additional metadata record should contain information about the name and version of its
596 schema. For example, in IEEE 1484.12.1–2002 metadata records, this is specified in the meta-
597 metadata element of the metadata record.

598 **6.3 Auxiliary data types**

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

601 **6.3.1 Any type**

602 **Synopsis**

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

604 **Description**

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

608 NOTE—If implementations specify or require data elements for which the type is defined in this
609 Standard as `any_type`, the implementations should provide the means to interpret and validate
610 the implementation-specific data. For example, an implementation that uses an XML binding
611 should include a valid XML schema that can be referenced in RCD instances bound in XML
612 documents, and the schema should be documented.

613 **6.3.2 LangString type**

614 **Synopsis**

```
615     type langstring_type(length) =  
616         record  
617         (  
618             language :  
619                 language_type,  
620             string :  
621                 characterstring(iso-10646-1),  
622                 // SPM: the length parameter  
623         );
```

624 **Description**

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

626 **Examples**

627 The following are three examples of localized strings: "Information Technology" in
628 French, "localization" in British English, and "xxx" in Japanese hiragana.

```
629     "fr", "Technologies de l'information"  
630     "en-GB", "localisation"  
631     "jp-JP-jisx208", "xxx"
```

632 **6.3.2.1 Language**

633 **Synopsis**

```
634     language :  
635         characterstring( iso-646 ),  
636         // SPM: 250 characters
```

637 **Description**

638 The `language` data element shall be a character string consisting of a required language code
639 followed by multiple, optional, hyphen-prefixed subcodes.

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

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

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

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

649 Constraints for additional subcodes are unspecified.

650 The value held by the character string shall be a valid language code as defined by the XML
651 Schema derived data type `language` (see XML Schema, Part 2).

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

654 NOTES:

655 1—The language code is normally given in lower case and the subcodes (if any) in upper case.
656 However, the values are case insensitive.

657 2—The XML Schema derived data type `language` does not enforce all constraints on this lan-
658 guage code.

659 **Examples**

```
660     "en-GB"  
661     "de"  
662     "fr-CA"  
663     "it"  
664     "i-bnn" (IANA Bunun)
```

665 **6.3.2.2 String**

666 **Synopsis**

```
667     string :  
668         characterstring(iso-10646-1),  
669         // SPM: The length parameter
```

670 **Description**

671 This data element contains the text of the localized string.

672 **6.3.3 Long identifier type**

673 **Synopsis**

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

682 **Description**

683 This data type is an identifier (a label) that is intended to be unique within the context of the
684 RCD. The catalog element is the name or designator of the identification or cataloging scheme
685 for this entry, in other words, a namespace-scheme. The entry element is the value of the identifier
686 within the identification or cataloging scheme that designates or identifies this RCD, in
687 other words, a namespace-specific string.

688 Values for this data type shall conform to the syntax for URIs as defined by IETF RFC 2396.
689 The catalog and entry values may be concatenated as a single character string in an application
690 profile or binding. If the catalog and entry values are concatenated, the resulting character
691 string shall conform to the syntax for URIs as defined by IETF RFC 2396.

692 **NOTES**

693 1—This Standard recommends that if an application profile or binding specifies a concatenated
694 format for the identifier, the result be in the form of a globally unique identifier in the form of a
695 Uniform Resource Name (URN) (see IETF RFC 2141 [B3]).

696 2—This Standard recommends that the catalog and entry lengths be limited so that the total
697 length of a concatenated identifier is never longer than 4000 characters.

698 **Annex A**

699 (informative)

700 **Bibliography**

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

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

704 [B3] Network Working Group (May 1997) RFC 2141, URN Syntax.

705 **Annex B**

706 (informative)

707 **Sample XML binding schema**

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

710 The example in Figure B.1 illustrates existing practice using an XML schema defined by the
711 IMS Global Learning Consortium [B2].

712 NOTE—The sample schema uses “rdceo, n ” as the label for the root data element. It is not re-
713 quired that an implementation of this Standard use the same labels for data-element or type labels
714 as those used in the example below, as long as the elements and types themselves are semanti-
715 cally equivalent.

716

```

717 <?xml version="1.0" encoding="UTF-8"?>
718 <xs:schema
719   targetNamespace="http://www.imsglobal.org/xsd/imsrdceo_rootv1p0"
720   xmlns:xs="http://www.w3.org/2001/XMLSchema"
721   xmlns="http://www.imsglobal.org/xsd/imsrdceo_rootv1p0"
722   elementFormDefault="qualified" attributeFormDefault="unqualified">
723   <xs:group name="extelement">
724     <xs:annotation>
725       <xs:documentation>extension mechanism for
726 elements</xs:documentation>
727     </xs:annotation>
728     <xs:sequence>
729       <xs:any namespace="##other" processContents="strict"
730 maxOccurs="unbounded"/>
731     </xs:sequence>
732   </xs:group>
733   <xs:element name="rdceo">
734     <xs:annotation>
735       <xs:documentation>A single definition of a competence,
736 educational objective etc</xs:documentation>
737     </xs:annotation>
738     <xs:complexType>
739       <xs:sequence>
740         <xs:element ref="identifier" minOccurs="1" maxOccurs="1"/>
741         <xs:element ref="title"/>
742         <xs:element ref="description" minOccurs="0"/>
743         <xs:element ref="definition" minOccurs="0"
744 maxOccurs="unbounded"/>
745         <xs:element ref="metadata" minOccurs="0"/>
746         <xs:sequence minOccurs="0">
747           <xs:group ref="extelement"/>
748         </xs:sequence>

```



```

749         </xs:sequence>
750         <xs:anyAttribute namespace="##other"
751 processContents="strict"/>
752     </xs:complexType>
753 </xs:element>
754 <xs:element name="langstring">
755     <xs:annotation>
756         <xs:documentation>A string in a human
757 language</xs:documentation>
758     </xs:annotation>
759     <xs:complexType>
760         <xs:simpleContent>
761             <xs:extension base="xs:string">
762                 <xs:anyAttribute namespace="##other"
763 processContents="strict"/>
764             </xs:extension>
765         </xs:simpleContent>
766     </xs:complexType>
767 </xs:element>
768 <xs:element name="title">
769     <xs:annotation>
770         <xs:documentation>A title for the
771 definition</xs:documentation>
772     </xs:annotation>
773     <xs:complexType>
774         <xs:sequence>
775             <xs:element ref="langstring" maxOccurs="unbounded"/>
776             <xs:sequence minOccurs="0">
777                 <xs:group ref="extelement"/>
778             </xs:sequence>
779         </xs:sequence>
780     <xs:anyAttribute namespace="##other"
781 processContents="strict"/>
782 </xs:complexType>
783 </xs:element>
784 <xs:element name="identifier">
785     <xs:annotation>
786         <xs:documentation>Catenated form of the identifier of an
787 RDCEO</xs:documentation>
788     </xs:annotation>
789     <xs:complexType>
790         <xs:simpleContent>
791             <xs:extension base="xs:anyURI">
792                 <xs:anyAttribute namespace="##other"
793 processContents="strict"/>
794             </xs:extension>
795         </xs:simpleContent>
796     </xs:complexType>
797 </xs:element>
798 <xs:element name="description">
799     <xs:annotation>
800         <xs:documentation>A description for the
801 definition</xs:documentation>
802     </xs:annotation>
803     <xs:complexType>
804         <xs:sequence>
805             <xs:element ref="langstring" maxOccurs="unbounded"/>

```

```

806         <xs:sequence minOccurs="0">
807             <xs:group ref="extelement" />
808         </xs:sequence>
809     </xs:sequence>
810     <xs:anyAttribute namespace="##other"
811 processContents="strict" />
812 </xs:complexType>
813 </xs:element>
814 <xs:element name="definition">
815     <xs:annotation>
816         <xs:documentation>A structured form of the
817 definition</xs:documentation>
818     </xs:annotation>
819     <xs:complexType>
820         <xs:sequence>
821             <xs:element ref="model" minOccurs="0" />
822             <xs:element ref="statement" maxOccurs="unbounded" />
823             <xs:sequence minOccurs="0">
824                 <xs:group ref="extelement" />
825             </xs:sequence>
826         </xs:sequence>
827         <xs:anyAttribute namespace="##other"
828 processContents="strict" />
829     </xs:complexType>
830 </xs:element>
831 <xs:element name="model">
832     <xs:annotation>
833         <xs:documentation>The model identification for the structured
834 definition</xs:documentation>
835     </xs:annotation>
836     <xs:complexType>
837         <xs:simpleContent>
838             <xs:extension base="xs:string">
839                 <xs:anyAttribute namespace="##other"
840 processContents="strict" />
841             </xs:extension>
842         </xs:simpleContent>
843     </xs:complexType>
844 </xs:element>
845 <xs:element name="statement">
846     <xs:annotation>
847         <xs:documentation>A component part of a structured
848 definition</xs:documentation>
849     </xs:annotation>
850     <xs:complexType>
851         <xs:sequence>
852             <xs:choice>
853                 <xs:element ref="statementtext" />
854                 <xs:element ref="statementtoken" />
855             </xs:choice>
856             <xs:sequence minOccurs="0">
857                 <xs:group ref="extelement" />
858             </xs:sequence>
859         </xs:sequence>
860         <xs:attribute name="statementid" type="xs:ID" />
861         <xs:attribute name="statementname" type="xs:string" />

```

```

862         <xs:anyAttribute namespace="##other"
863 processContents="strict"/>
864     </xs:complexType>
865 </xs:element>
866 <xs:element name="statementtext">
867     <xs:annotation>
868         <xs:documentation>Used for statements with free-form
869 text</xs:documentation>
870     </xs:annotation>
871 </xs:complexType>
872     <xs:sequence>
873         <xs:element ref="langstring" maxOccurs="unbounded"/>
874         <xs:sequence minOccurs="0">
875             <xs:group ref="extelement"/>
876         </xs:sequence>
877     </xs:sequence>
878 </xs:complexType>
879 </xs:element>
880 <xs:element name="source">
881     <xs:annotation>
882         <xs:documentation>Source identification for a vocabulary
883 token</xs:documentation>
884     </xs:annotation>
885 </xs:complexType>
886     <xs:simpleContent>
887         <xs:extension base="xs:string">
888             <xs:anyAttribute namespace="##other"
889 processContents="strict"/>
890         </xs:extension>
891     </xs:simpleContent>
892 </xs:complexType>
893 </xs:element>
894 <xs:element name="value">
895     <xs:annotation>
896         <xs:documentation>A vocabulary token</xs:documentation>
897     </xs:annotation>
898 </xs:complexType>
899     <xs:simpleContent>
900         <xs:extension base="xs:string">
901             <xs:anyAttribute namespace="##other"
902 processContents="strict"/>
903         </xs:extension>
904     </xs:simpleContent>
905 </xs:complexType>
906 </xs:element>
907 <xs:element name="metadata">
908     <xs:annotation>
909         <xs:documentation>A container for metadata</xs:documentation>
910     </xs:annotation>
911 </xs:complexType>
912     <xs:sequence>
913         <xs:element ref="rdceoschema" minOccurs="0"/>
914         <xs:element ref="rdceoschemaversion" minOccurs="0"/>
915         <xs:sequence minOccurs="0">
916             <xs:group ref="extelement"/>
917         </xs:sequence>
918     </xs:sequence>

```

```

919         <xs:anyAttribute namespace="##other"
920 processContents="strict"/>
921     </xs:complexType>
922 </xs:element>
923 <xs:element name="statementtoken">
924     <xs:annotation>
925         <xs:documentation>Used for statements with token values
926 (vocabulary use)</xs:documentation>
927     </xs:annotation>
928     <xs:complexType>
929         <xs:sequence>
930             <xs:element ref="source"/>
931             <xs:element ref="value"/>
932             <xs:sequence minOccurs="0">
933                 <xs:group ref="extelement"/>
934             </xs:sequence>
935         </xs:sequence>
936         <xs:anyAttribute namespace="##other"
937 processContents="strict"/>
938     </xs:complexType>
939 </xs:element>
940 <xs:element name="rdceoschema">
941     <xs:annotation>
942         <xs:documentation>The identity of the RDCEO schema - assumed
943 to be IMS RDCEO if absent.</xs:documentation>
944     </xs:annotation>
945     <xs:complexType>
946         <xs:simpleContent>
947             <xs:extension base="xs:string">
948                 <xs:anyAttribute namespace="##other"
949 processContents="strict"/>
950             </xs:extension>
951         </xs:simpleContent>
952     </xs:complexType>
953 </xs:element>
954 <xs:element name="rdceoschemaversion">
955     <xs:annotation>
956         <xs:documentation>The version of the RDCEO schema - assumed to
957 be 1.0 if absent</xs:documentation>
958     </xs:annotation>
959     <xs:complexType>
960         <xs:simpleContent>
961             <xs:extension base="xs:string">
962                 <xs:anyAttribute namespace="##other"
963 processContents="strict"/>
964             </xs:extension>
965         </xs:simpleContent>
966     </xs:complexType>
967 </xs:element>
968 </xs:schema>

```

969

Figure B.1—An example schema