
Learning Standards Digital Representation Specification: JSON Serialization Schemata

Granular Identifiers and Metadata for CCSS (GIM CCSS) Project

**Daniel R Rehak
V 1.0
2013-09-12**



Copyright © 2013, State Educational Technology Directors Association (SETDA)

This work is made available under the terms of the Creative Commons Attribution 3.0 Unported (CC BY 3.0)

<http://creativecommons.org/licenses/by/3.0/>

Contents

Introduction.....	1
Notation.....	2
Informal JSON Model.....	3
Learning Standards Statement.....	7
Learning Standards Statement Schema.....	7
Semantic Constraints.....	13
Implementation Considerations.....	13
Schema Versioning.....	13
IANA Considerations.....	14
Learning Standards Statement Collection.....	15
Learning Standards Statement Collection Schema.....	15
Semantic Constraints.....	16
Implementation Considerations.....	16
Schema Versioning.....	17
IANA Considerations.....	17
Vocabulary.....	18
Vocabulary Schema.....	18
Semantic Constraints.....	18
Implementation Considerations.....	18
Schema Versioning.....	19
IANA Considerations.....	19
Security Considerations.....	20
Conformance.....	21
Examples.....	22
Metadata Example.....	22
Granular Decomposition Example.....	25
State 15% Extension Example.....	32
Versioning Example.....	35
Collection Example.....	39
Linked Data Example.....	41
Schemata Design Decisions.....	43
General Design Decisions.....	43
JSON Schemata Decisions.....	44
Learning Standards Statement Schema Decisions.....	44
Learning Standards Statement Collection Schema Decisions.....	45
Normative References.....	46
Informative References.....	47

Annex: JSON Schema Validation Extensions References	48
GIM CCSS Project Documentation.....	49
Change Log.....	50

Introduction

Note: This section is informative.

This document specifies the schemata for the JSON serialization of a learning standards statement. It includes the:

- **Learning Standards Statement Schema** – Schema for a single learning standards statement.
- **Learning Standards Statement Collection Schema** – Schema for a collection of learning standards statements.
- **Vocabulary Schema** – Schema for a vocabulary.

The schemata permit the Common Core State Standards (CCSS) statements developed in the *Granular Identifiers and Metadata for CCSS Project (GIM CCSS)* to be represented as JSON objects.

The document includes:

- **Notation** – Definitions of normative terms used in the Specification.
- **Informal JSON Model** – The JSON model for the digital representation of Common Core State Standards derived from the information model contained in the *Scope, Technical Requirements, Approaches, and Recommendations* [Scope] (informative).
- **Schemata** – The schemata for learning standards statements, collections of statements and vocabularies. Each schema Specification includes:
 - *Schema* – the JSON Schema specified using [JSON Schema Core, JSON Schema Validation] as extended herein.
 - *Semantic Constraints* – constraints that cannot be specified with [JSON Schema Core, JSON Schema Validation].
 - *Implementation Considerations* – recommended best practices on how to represent or use a JSON instance of the object.
 - *IANA Considerations* – recommendations for media type names for JSON objects conforming to the specification.
- **Security Considerations** – Security considerations for creating, producing or consuming JSON objects for learning standards statements that conform to this Specification.
- **Conformance** – Criteria for an instance of a JSON object to conform to this Specification.
- **Examples** – Sample learning standards statements as JSON objects (informative).
- **Schema Design Decisions** – Choices and rationale in designing the schemata (informative).
- **Normative References** – Normative references to other specifications used in this Specification.
- **Informative References** – References to other documents (informative).
- **Annex: JSON Schema Validation Extensions Reference** – Specification of extensions to [JSON Schema Validation].

Notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this Specification are to be interpreted as described in [RFC 2119].

Unless otherwise noted, sections in this document are normative.

Informal JSON Model

Note: This section is informative.

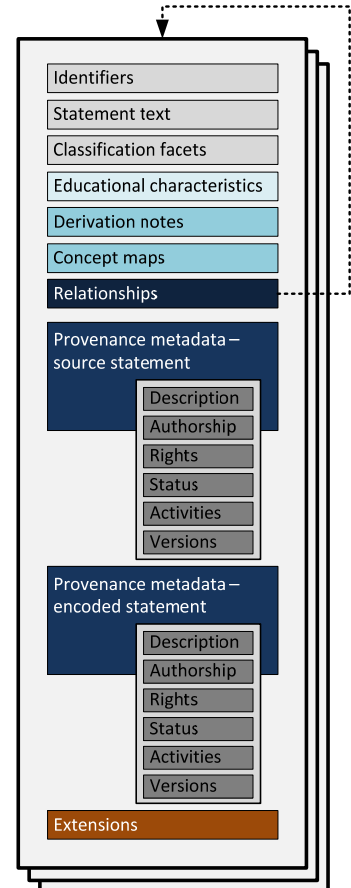
The schemata are designed to support the complete information model contained in the *Scope, Technical Requirements, Approaches, and Recommendations* [Scope].

The information model for a learning standards statement includes (but is not limited to):

- The identifier(s) assigned to the standards statement (required).
- The classification facets/taxons for the standards statement (required, except in a collection).
- The encoding of the text of the standards statement (in HTML 5, required for terminal nodes in the graph, except in a collection).
- Educational descriptors for the standards statement.
- Notes about the derivation of the standards statement.
- Concept maps identifying existing concepts within the standards statement.
- Provenance metadata for the original source statement, including:
 - General description (e.g., title)
 - Authorship
 - History/Activities
 - Versioning
 - Licensing, Rights and Attribution
 - Language
 - Status/State
- Provenance metadata for the digital encoding of the statement, elements as above.
- Relationships between a standards statement and other statements.
- Extensions.

The model supports:

- Multiple arbitrary identifiers and locators per statement.
- Multiple narrative statements.
- Multiple descriptive notes.
- Multiple classifiers per statement.
- Verifiable vocabulary-based classification.
- Metadata, versioning, language translation and licensing information at any level in the hierarchy of statements.
- Multiple relationships between a statement and another statement in any standard. Relationships between statements support representing both tree and graph structures. In the representation of a complete standard, the relationships must produce a rooted graph.



Each statement must have at least one identifier and one classifier. Terminal nodes in the graph (e.g., CCSS *Standard*, CCSS *Components*, or GIM-CCSS *Granular Statements*) must contain at least one narrative element containing the text of the statement. All other items are optional. When used in a collection for exchange, only an identifier that can be used to access the statement is required. Any item may be extended with additional attributes.

The same JSON object model is used to represent information about whole standards and individual statements within standards. For the standard, the JSON object is classified as the root of the standard.

The model supports inheritance of values. The tree can be walked from a statement back to the root to find the value for an attribute that is not included with the statement, e.g., if there is no metadata for a statement, it might be included with a predecessor statement in the hierarchy. The values for a statement override the values of a parent.

While the model supports all of the features listed, a specific implementation need not use all the features. The best practices guide [Best Practices] will profile the overall model and describe how to use selected features to represent CCSS statements:

- As the digital representation of CCSS statements and standards, including the local extensions.
- As the format for exchanging CCSS statements between systems.
- As a model for a data store holding CCSS statements.

An informal description of the key elements of the JSON serialization of a learning standards statement follows.

<code>{ "learningStandardsStatement":</code>	
<code> {</code>	
<code> "identifiers":</code>	Identifiers for the statement
<code> [</code>	Array of identifiers
<code> {</code>	
<code> "identifier":</code>	An identifier for the statement
<code> {</code>	
<code> "idType":</code>	The type of identifier: controlled vocabulary
<code> "id":</code>	The identifier, string
<code> }</code>	
<code> }</code>	
<code>]</code>	
<code> "classifiers":</code>	Taxon classifiers for the statement
<code> {</code>	A set of individual classification objects
<code> <classifier object></code>	A classifier object key-value pair
<code> }</code>	
<code> "statements":</code>	Standards Statements
<code> [</code>	Array of statements
<code> {</code>	
<code> "statement":</code>	An individual statement, in HTML5
<code> {</code>	
<code> "language":</code>	The language of the statement: controlled vocabulary
<code> "text":</code>	The standards statement, string in HTML5
<code> }</code>	
<code> }</code>	
<code>]</code>	
<code> "educationalCharacteristics":</code>	Educational attributes of the statement
<code> {</code>	A set of individual educational characteristic objects
<code> <characteristic object></code>	An educational characteristic object key-value pair
<code> }</code>	
<code> "relationships":</code>	Relationships with other statements
<code> [</code>	Array of relationships
<code> {</code>	
<code> "relationship":</code>	A relationship
<code> {</code>	
<code> "relationType":</code>	Type of relationship: controlled vocabulary
<code> "relatedStatement":</code>	Identifier Object of the related learning standards statement
<code> }</code>	
<code> }</code>	
<code> }</code>	
<code> }</code>	

}	
]	
"notes:"	Notes about the derivation of the statement
[Array of notes
{ "note":	A note
{ "language":	The language of the note: vocabulary
"text":	The note, in HTML5
}	
}	
]	
"sourceStatementMetadata":	Metadata for the original statement
{	A set of individual metadata objects
<metadata object>	A metadata object key-value pair
}	
"encodedStatementMetadata":	Metadata for the digital representation
{	A set of individual metadata objects
<metadata object>	A metadata object key-value pair
}	
}	
}	

An example of a simple statement serialization of CCSS.Math.Content.K.CC.A.1 (without any metadata or GIM-CCSS modifications of the existing statement) follows.

```

1  {
2    "learningStandardsStatement": {
3      "identifiers": [
4        {
5          "identifier": {
6            "idType": "CCSS Path",
7            "id": "http://corestandards.org/Math/Content/K/CC/A/1"
8          }
9        },
10       {
11         "identifier": {
12           "idType": "CCSS GUID",
13           "id": "B62C1C106873438AA0126760075A65A3"
14         }
15       }
16     ],
17     "classifiers": {
18       "taxons": {
19         "initiative": "CCSS",
20         "framework": "Math",

```



```

21     "set": "Content",
22     "grade": "KG",
23     "discipline": "Math",
24     "domain": "CC",
25     "cluster": "A",
26     "standard": "1"
27   },
28   "statementType": "Standard"
29 },
30 "statements": [
31   {
32     "statement": {
33       "language": "en-US",
34       "text": "Count to 100 by ones and by tens."
35     }
36   }
37 ],
38 "educationalCharacteristics": {
39   "gradeLevel": "KG"
40 },
41 "relationships": [
42   {
43     "relationship": {
44       "relationType": "child-of",
45       "relatedStatement": {
46         "identifier": {
47           "idType": "CCSS GUID",
48           "id": "4F4106218F834258BCDDB7EB39806880"
49         }
50       }
51     }
52   }
53 ]
54 }
55 }

```

Note: JSON last validated: 20130213. JSON instance schema validation: not validated.

Learning Standards Statement

The learning standards statement schema is used describe a learningStandardsStatement object. The schema is represented using [JSON Schema Core, JSON Schema Validation] incorporating the extensions described in the Annex. The schema does not incorporate duck typing of values; extensions through duck typing are conforming instances.

Learning Standards Statement Schema

The schema for the root-level learningStandardsStatement object.

```

1  {
2  "$schema": "http://json-schema.org/draft-04/schema#",
3  "version": "GIM-CCSS 20130212",
4  "id": "http://publishingagent.tld/schema/learningstandardsstatement.json#",
5  "schemaLicense": "Copyright © 2013, State Educational Technology Directors Association (SETDA). This
6  work is made available under the terms of the Creative Commons Attribution 3.0 Unported (CC BY 3.0)
7  http://creativecommons.org/licenses/by/3.0/",
8  "title": "Learning Standards Statement Collection Schema",
9  "description": "Schema for an individual learning standards statement.",
10 "type": "object",
11 "required": [
12   "learningStandardsStatement"
13 ],
14 "properties": {
15   "learningStandardsStatement": {
16     "id": "#learningStandardsStatement",
17     "title": "Learning Standards Statement Schema",
18     "description": "A single learning standards statement.",
19     "type": "object",
20     "required": [
21       "identifiers",
22       "$schemaVersion",
23       "$schemaLocation"
24     ],
25     "properties": {
26       "$schemaVersion": {
27         "id": "#$schemaVersion",
28         "title": "Schema Version",
29         "description": "The version identifier of the schema used for the instance.",
30         "type": "string"
31       },
32       "$schemaLocation": {
33         "id": "#$schemaLocation",
34         "title": "Schema Location",
35         "description": "The path to the schema used for the instance.",
36         "type": "string"
37       },
38       "identifiers": {
39         "$ref": "#definitions/identifiers"
40     },

```

```

41     "classifiers": {
42         "$ref": "#definitions/classifiers"
43     },
44     "statements": {
45         "$ref": "#definitions/statements"
46     },
47     "educationalCharacteristics": {
48         "$ref": "#definitions/educationalCharacteristics"
49     },
50     "notes": {
51         "$ref": "#definitions/notes"
52     },
53     "relationships": {
54         "$ref": "#definitions/relationships"
55     },
56     "sourceStatementMetadata": {
57         "$ref": "#definitions/metadata"
58     },
59     "encodedStatementMetadata": {
60         "$ref": "#definitions/metadata"
61     }
62 }
63 },
64 },
65 "definitions": {
66     "identifiers": {
67         "id": "#identifiers",
68         "title": "Identifiers",
69         "description": "An array of identifiers for the statement. Required.",
70         "oneOf": [
71             {
72                 "type": "array",
73                 "items": {
74                     "id": "0",
75                     "title": "Identifier",
76                     "description": "A statement identifier.",
77                     "type": "object",
78                     "properties": {
79                         "identifier": {
80                             "id": "#identifier",
81                             "type": "object",
82                             "required": [
83                                 "id"
84                             ],
85                             "properties": {
86                                 "idType": {
87                                     "id": "#idType",
88                                     "title": "idType",
89                                     "description": "The type of the identifier, from a controlled vocabulary.",
90                                     "type": "string"
91                                 },
92                                 "id": {

```

93	"\"id\": \"#id\",
94	"\"title\": \"id\",
95	"\"description\": \"The identifier. Required if the identifier is present. Semantic
96	constraints validate the format of the ID based on its type.\",
97	"\"type\": \"string\"
98	}
99	}
100	}
101	}
102	}
103	},
104	{
105	"\"title\": \"Identifier\",
106	"\"description\": \"A statement identifier.\",
107	"\"type\": \"object\",
108	"\"properties\": {
109	"\"identifier\": {
110	"\"id\": \"#identifier\",
111	"\"type\": \"object\",
112	"\"required\": [
113	"\"id\"
114],
115	"\"properties\": {
116	"\"idType\": {
117	"\"id\": \"#idType\",
118	"\"title\": \"idType\",
119	"\"description\": \"The type of the identifier, from a controlled vocabulary.\",
120	"\"type\": \"string\"
121	},
122	"\"id\": {
123	"\"id\": \"#id\",
124	"\"title\": \"id\",
125	"\"description\": \"The identifier. Required if the identifier is present. Semantic
126	constraints validate the format of the ID based on its type.\",
127	"\"type\": \"string\"
128	}
129	}
130	}
131	}
132	}
133]
134	},
135	"\"classifiers\": {
136	"\"id\": \"#classifiers\",
137	"\"title\": \"Classifiers\",
138	"\"description\": \"A set of multiple statement taxons and classifiers. A classifier is required for a
139	stand-alone statement, but not if the statement is part of a collection instance.\",
140	"\"type\": \"object\",
141	"\"properties\": {
142	"\"taxons\": {
143	"\"id\": \"#taxons\",
144	"\"title\": \"Taxons\",

145	"description": "The statement taxons. A taxon object, typically key-value pairs.",
146	"type": "object"
147	},
148	"statementType": {
149	"id": "#statementType",
150	"title": "Statement Type",
151	"description": "A single statement type classification, from the taxon vocabulary. Should
152	indicate where the statement is in the taxon structure.",
153	"type": "string"
154	}
155	}
156	},
157	"statements": {
158	"id": "#statements",
159	"title": "Statements",
160	"description": "An array of standards statements in natural language. The statement text is
161	required for a stand-alone statement, but not if the statement is part of a collection instance.",
162	"type": "array",
163	"items": {
164	"id": "0",
165	"title": "Statement",
166	"description": "A single standards statement.",
167	"type": "object",
168	"properties": {
169	"statement": {
170	"id": "#statement",
171	"type": "object",
172	"required": [
173	"text"
174],
175	"properties": {
176	"language": {
177	"id": "#language",
178	"title": "Language",
179	"description": "The language of the standards statement, from a defined controlled
180	vocabulary.",
181	"type": "string"
182	},
183	"encoding": {
184	"id": "#encoding",
185	"title": "Encoding",
186	"description": "The encoding format of the standards statement, from a defined
187	controlled vocabulary.",
188	"type": "string"
189	},
190	"text": {
191	"id": "#text",
192	"title": "Text",
193	"description": "The standards statement, typically encoded in a subset of HTML5.
194	Required if the statement is included in the instance.",
195	"type": "string"
196	}

```

197     }
198   }
199 }
200 }
201 },
202 "educationalCharacteristics": {
203   "id": "#educationalCharacteristics",
204   "title": "Educational Characteristics",
205   "description": "A set of educational characteristics.",
206   "type": "object",
207   "properties": {
208     "gradeLevel": {
209       "id": "#gradeLevel",
210       "title": "Grade Level",
211       "description": "The grade levels for the statement, from a controlled vocabulary.",
212       "oneOf": [
213         {
214           "type": "string"
215         },
216         {
217           "type": "array",
218           "items": {
219             "type": "string"
220           }
221         }
222       ]
223     }
224   }
225 },
226 "notes": {
227   "id": "#notes",
228   "title": "Notes",
229   "description": "An array of statement derivation notes in natural language.",
230   "type": "array",
231   "items": {
232     "id": "0",
233     "title": "Note",
234     "description": "A single derivation note.",
235     "type": "object",
236     "properties": {
237       "note": {
238         "id": "#note",
239         "type": "object",
240         "required": [
241           "text"
242         ],
243         "properties": {
244           "language": {
245             "id": "#language",
246             "title": "Language",
247             "description": "The language of the note , from a defined controlled vocabulary.",
248             "type": "string"

```

```

249         },
250         "encoding": {
251             "id": "#encoding",
252             "title": "Encoding",
253             "description": "The encoding format of the standards statement, from a defined
254 controlled vocabulary.",
255             "type": "string"
256         },
257         "text": {
258             "id": "#text",
259             "title": "Text",
260             "description": "The derivation note, typically encoded in a subset of HTML5. Required if
261 the note is included in the instance.",
262             "type": "string"
263         }
264     }
265 }
266 }
267 }
268 },
269 "relationships": {
270     "id": "#relationships",
271     "title": "Relationships",
272     "description": "An array of relationships between statements.",
273     "type": "array",
274     "items": {
275         "id": "0",
276         "title": "Relationship",
277         "description": "A single relationship.",
278         "type": "object",
279         "properties": {
280             "relationship": {
281                 "id": "#relationship",
282                 "type": "object",
283                 "required": [
284                     "relatedStatement"
285                 ],
286                 "properties": {
287                     "relationType": {
288                         "id": "#relationType",
289                         "title": "Relation Type ",
290                         "description": "The type of the relationship, from a controlled vocabulary.",
291                         "type": "string"
292                     },
293                     "relatedStatement": {
294                         "id": "#relatedStatement",
295                         "title": "Related Statement ",
296                         "description": "Identifiers of the related statements, using the same definition as
297 statement ids.",
298                         "$ref": "#definitions/identifiers"
299                     }
300                 }
301             }
302         }
303     }
304 }

```

```

301     }
302     }
303   }
304 },
305   "metadata": {
306     "id": "#metadata",
307     "title": "Metadata",
308     "description": "Metadata attributes for a source or encoded statement. Attributes are defined
309 externally to the schema.",
310     "type": "object"
311   }
312 }
313 }

```

Note: JSON schema last validated: 20130213. JSON last validated: 20130213.

To Do: Assign URI for hosting the schema.

Semantic Constraints

A value for identifier `idType` SHOULD come from a controlled vocabulary of identifier types.

A structure for the classifier `taxons` object SHOULD come from a defined classification taxonomy.

A value for the classifier `statementType` SHOULD come from a controlled vocabulary of statement types.

A value for `language` SHALL come from a controlled vocabulary of language codes.

A value for the `gradeLevel` educational characteristic SHOULD come from a controlled vocabulary of grade levels.

A value for the relationship `relationType` SHOULD come from a controlled vocabulary of relation types.

When used as a standalone statement (not part of a collection), the `classifiers` and `statements` objects are required in a JSON instance.

An implementation MAY impose additional constraints on an instance of a conforming JSON object.

Extensions to the schemata SHALL NOT use object names that begin with the `@` (at sign) character (the `@` is reserved for [JSON-LD] objects).

Implementation Considerations

In the representation of a complete learning standard (a set of learning standards statements), the relationships MUST produce a rooted graph based on the vocabulary of statement type classifiers.

A producer SHOULD NOT create NULL or empty string values.

Schema Versioning

The version number for this version of the schema is: GIM-CCSS 20130212.

Any significant change to the schema e.g., addition or deletion of a required schema element, **SHALL** be indicated by a new schema version number. Additions or deletions of other schema elements **SHOULD** be indicated by a new schema version number. An instance of a learningStandardsStatement object using extension objects **SHALL** use the specified schema version number.

IANA Considerations

All learning standards statement JSON document instance **SHOULD** be given a media type (MIME Type).

The proposed MIME media type for the JSON serialization of a learning standards statement is:

- Type name: application
- Subtype name: vnd.ccss.standardstatement+JSON

The media type **SHALL** conform to [RFC4288] and **SHOULD** be registered with IANA [<http://www.iana.org/cgi-bin/mediatypes.pl>] in accordance with [RFC4289].

Learning Standards Statement Collection

The learning standards statement collection schema is used to bundle zero or more learningStandardsStatement objects into a single learningStandardsStatementCollection object. The schema is represented using [JSON Schema Core, JSON Schema Validation] incorporating the extensions described in the Annex. The schema does not incorporate duck typing of values; extensions through duck typing are conforming instances.

Learning Standards Statement Collection Schema

The schema for the root-level learningStandardsStatementCollection object.

```

1  {
2    "$schema": "http://json-schema.org/draft-04/schema#",
3    "version": "GIM-CCSS 20130212",
4    "id": "http://publishingagent.tld/schema/learningstandardsstatementcollection.json#",
5    "schemaLicense": "Copyright © 2013, State Educational Technology Directors Association (SETDA). This
6 work is made available under the terms of the Creative Commons Attribution 3.0 Unported (CC BY 3.0)
7 http://creativecommons.org/licenses/by/3.0/",
8    "title": "Learning Standards Statement Collection Schema",
9    "description": "Schema for an ordered collection of zero or more learning standards statements. The
10 collection includes an optional count of the number of statements in the collection. Each statement is an
11 instance of a learning standards statement JSON object – but only the ID attribute is required.",
12    "type": "object",
13    "required": [
14      "learningStandardsStatementCollection",
15      "$schemaVersion",
16      "$schemaLocation"
17    ],
18    "properties": {
19      "learningStandardsStatementCollection": {
20        "id": "#learningStandardsStatementCollection",
21        "title": "Learning Standards Statement Collection Schema",
22        "description": "An ordered collection of zero or more learning standards statements.",
23        "type": "object",
24        "required": [
25          "statements",
26          "$schemaVersion",
27          "$schemaLocation"
28        ],
29        "properties": {
30          "$schemaVersion": {
31            "id": "#$schemaVersion",
32            "title": "Schema Version",
33            "description": "The version identifier of the schema used for the instance.",
34            "type": "string"
35          },
36          "$schemaLocation": {
37            "id": "#$schemaLocation",
38            "title": "Schema Location",
39            "description": "The path to the schema used for the instance.",
40            "type": "string"

```

```

41     },
42     "totalStatements": {
43         "id": "#totalStatements",
44         "title": "Total Statements",
45         "description": "The total number of learning standards statement objects in the collection.
46 Optional. Must match the size of the collection if present in the instance.",
47         "type": "integer",
48         "minimum": 0
49     },
50     "statements": {
51         "id": "#statements",
52         "title": "Statements",
53         "description": "An array holding the learning standards statement objects. Required.",
54         "type": "array",
55         "minItems": 0,
56         "items": {
57             "id": "schema/learningstandardsstatement.json",
58             "title": "Learning Standards Statement",
59             "description": "An individual learning standards statement. Must conform to the statement
60 schema and all JSON object constraints.",
61             "type": "object",
62             "$ref": "http://publishingagent.tld/schema/learningstandardsstatement.json#"
63         }
64     }
65 }
66 }
67 }
68 }

```

Note: JSON schema last validated: 20130213. JSON last validated: 20130213.

To Do: Assign URI for hosting the schema.

Semantic Constraints

If `totalStatements` is not null, `totalStatements` SHALL be the size of the array of statements.

All semantic constraints for a learning standards statement apply when the statement is included in a collection.

An implementation MAY impose additional constraints on an instance of a conforming JSON object.

Extensions to the schemata SHALL NOT use object names that begin with the @ (at sign) character (the @ is reserved for [JSON-LD] objects).

Implementation Considerations

A producer SHOULD NOT create NULL or empty string values.

A producer SHOULD NOT duplicate statements in the collection; each item SHOULD be unique.

A consumer SHOULD anticipate that a collection contains a large number of statements (2000 or more). If the collection contains a non null `totalStatements`, a consumer MAY use this value to determine if it can process all the items.

A producer SHOULD anticipate that a consumer cannot process a collection with a large number of statements. The producer SHOULD include `totalStatements` to inform the consumer of the size of the collection.

An API design SHOULD anticipate that a consumer cannot process a collection with a large number of statements. The API should support flow control.

Schema Versioning

The version number for this version of the schema is: GIM-CCSS 20130212.

Any significant change to the schema e.g., addition or deletion of a required schema element, SHALL be indicated by a new schema version number. Additions or deletions of other schema elements SHOULD be indicated by a new schema version number. An instance of a `learningStandardsStatementCollection` object using extension objects SHALL use the specified schema version number.

IANA Considerations

All learning standards statement collection JSON document instance SHOULD be given a media type (MIME Type).

The proposed MIME media type for the JSON serialization of a collection of learning standards statement is:

- Type name: `application`
- Subtype name: `vnd.ccss.standardcollection+JSON`

The media type SHALL conform to [RFC4288] and SHOULD be registered with IANA [<http://www.iana.org/cgi-bin/mediatypes.pl>] in accordance with [RFC4289].

Vocabulary

The vocabulary schema is used describe a vocabulary object. The schema is represented using [JSON Schema Core, JSON Schema Validation] incorporating the extensions described in the Annex. The schema does not incorporate duck typing of values; extensions through duck typing are conforming instances.

Vocabulary Schema

The schema for the root-level vocabulary object.

Note: The Schema is incomplete.

```

1  {
2    "$schema": "http://json-schema.org/draft-04/schema#",
3    "version": "GIM-CCSS 20130212",
4    "id": "http://publishingagent.tld/schema/learningstandardsvocabulary.json#",
5    "schemaLicense": "Copyright © 2013, State Educational Technology Directors Association (SETDA). This
6 work is made available under the terms of the Creative Commons Attribution 3.0 Unported (CC BY 3.0)
7 http://creativecommons.org/licenses/by/3.0/",
8    "title": "Learning Standards Statement Vocabulary Schema",
9    "description": "Definition of a vocabulary and its terms.",
10   "type": "object",
11   "required": [
12     "vocabulary",
13     "$schemaVersion",
14     "$schemaLocation"
15   ],
16   "name": "vocabulary",
17   "properties": {
18     "...": {},
19     "...": {}
20   }
21 }
```

Note: JSON schema last validated: not validated. JSON last validated: not validated.

To Do: Assign URI for hosting the schema.

Semantic Constraints

An implementation MAY impose additional constraints on an instance of a conforming JSON object.

Extensions to the schemata SHALL NOT use object names that begin with the @ (at sign) character (the @ is reserved for [JSON-LD] objects).

Implementation Considerations

A producer SHOULD NOT create NULL or empty string values.

Schema Versioning

The version number for this version of the schema is: GIM-CCSS 20130212.

Any significant change to the schema e.g., addition or deletion of a required schema element, **SHALL** be indicated by a new schema version number. Additions or deletions of other schema elements **SHOULD** be indicated by a new schema version number. An instance of a vocabulary object using extension objects **SHALL** use the specified schema version number.

IANA Considerations

All learning standards vocabulary JSON document instance **SHOULD** be given a media type (MIME Type).

The proposed MIME media type for the JSON serialization of a learning standards vocabulary is:

- Type name: application
- Subtype name: vnd.ccss.vocabulary+JSON

The media type **SHALL** conform to [RFC4288] and **SHOULD** be registered with IANA [<http://www.iana.org/cgi-bin/mediatypes.pl>] in accordance with [RFC4289].

Security Considerations

Producers including learning standards statement in JSON objects or consumers accessing learning standards statement may want to consider the potential for unsolicited or malicious content and should take preventive measures to recognize such content and either identify it or not include it in their objects.

Producers **SHOULD** take reasonable measures to make sure potentially malicious user input such as cross-site scripting attacks are not included in the learning standards statements or vocabularies they publish.

Services that provide learning standards statements or vocabularies to consumers and other services **MUST** take reasonable measures to make sure potentially malicious ingested input is not distributed or emitted.

Consumers should be aware of the potential for malicious content where the attacker publishes objects with falsified property values with the intent of injecting malicious content, hiding or corrupting legitimate content, or misleading users.

Consumers that make learning standards statements available for crawling by search engines **SHOULD** take reasonable measures to limit any use of their site as a Search Engine Optimization loophole. This **MAY** include converting un-trusted hyperlinks to text or including a `rel="nofollow"` attribute.

JSON objects are subject to the same security considerations described in [RFC 4627]. The JSON objects may include URIs; see [RFC 3986] for security considerations. The JSON objects may include IRIs; see [RFC 3987] for security considerations.

Conformance

To conform to this Specification, an instance of a JSON object:

- SHALL be a valid JSON object.
- SHALL have a root object of `learningStandardsStatement`, `learningStandardsVocabulary` or `learningStandardsStatementCollection`.
- SHALL conform to and validate according to the JSON schemata for the root object and all sub objects.
 - Duck-typed objects are permitted throughout. The JSON schema does not include duck typing of objects. A JSON object SHALL be considered to be valid if it validates when duck typed extensions are removed.
- SHALL meet all required semantic constraints.

Common Core State Standards (CCSS) statements developed in the *Granular Identifiers and Metadata for CCSS (GIM CCSS) Project* SHALL use the schemata as the definition of the digital representation of CCSS statements. These statements SHALL NOT include duck-typed extensions. The JSON object holding the digital representation of the statement SHALL conform to this Specification.

JSON objects used to exchange Common Core State Standards (CCSS) statements between systems SHALL conform to the schemata. These statements SHALL NOT include duck-typed extensions. The JSON object holding the digital representation of the statement SHALL conform to this Specification.

A digital store holding Common Core State Standards (CCSS) statements MAY store the statements in JSON objects that conform to the Specification.

This Specification does not give additional conformance criteria for producers, consumers or processors of JSON objects that conform to the schemata defined herein.

Examples

Note: This section is informative.

Note: The examples are for illustrative purposes only. The granular statements used are examples.

Examples include:

- Metadata for the CCSS Math Standard as a whole.
- The granular decomposition of a CCSS Math *Standard*.
- A local extension to a statement.
- The versioning of a statement.
- A collection of statements.
- A linked-data representation of a statement in [JSON-LD]

Metadata Example

The example shows the serialization of the root of the CCSS Math Standards with illustrative metadata. The example includes:

- The root Math Standards classified as a “Framework”.
- The placement of the Math Standards as a child of the CCSSO standards.
- GIM-CCSS assigned identifiers.
- Grade level designations (K-12).
- The parent (CCSS Initiative) relationship.
- An annotation.
- Illustrative provenance metadata for the source statement.
- Illustrative provenance metadata for the digital representation indicating it is a draft.

The example does not include relationships enumerating the parts (Sets). The example does not include a complete set of authoring and publishing activities.

```

1  {
2  "learningStandardsStatement": {
3  "$schemaVersion": "GIM-CCSS 20130212",
4  "$schemaLocation": "http://publishingagent.tld/schema/learningstandardsstatement.json#",
5  "identifiers": [
6    {
7      "identifier": {
8        "idType": "CCSS Path",
9        "id": "http://corestandards.org/Math"
10     }
11   },
12   {
13     "identifier": {
14       "idType": "CCSS Dot",
15       "id": "CCSS.Math"
16     }
17   },
18   {
19     "identifier": {
20       "idType": "GIM UUID",
21       "id": "cf116eb9f9d1408489aed9b530c34fc9"

```

22	}
23	},
24	{
25	"identifier": {
26	"idType": "GIM Path",
27	"id": "CCSS/Math"
28	}
29	}
30],
31	"classifiers": {
32	"taxons": {
33	"initiative": "CCSS",
34	"framework": "Math"
35	},
36	"statementType": "Framework"
37	},
38	"statements": [
39	{
40	"statement": {
41	"language": "en-US",
42	"encoding": "text/html",
43	"text": "Understanding mathematics These Standards define what students
44	should understand and be able to do in their study of mathematics. ..."
45	}
46	}
47],
48	"educationalCharacteristics": {
49	"gradeLevel": [
50	"KG",
51	"01",
52	"02",
53	"03",
54	"04",
55	"05",
56	"06",
57	"07",
58	"08",
59	"09",
60	"10",
61	"11",
62	"12"
63]
64	},
65	"relationships": [
66	{
67	"relationship": {
68	"relationType": "part-of",
69	"relatedStatement": [
70	{
71	"identifier": {
72	"idType": "GIM UUID",
73	"id": "e2357213c9bb42b19a4710928fcb51bc"
74	}

```

75         },
76         {
77             "identifier": {
78                 "idType": "GIM Path",
79                 "id": "CCSS"
80             }
81         }
82     ]
83 }
84 }
85 ],
86 "notes": [
87     {
88         "note": {
89             "language": "en-US",
90             "encoding": "text/html",
91             "text": "This is an example JSON serialization of the top-level statement in the CCSSO Math
92 Standards"
93         }
94     }
95 ],
96 "sourceStatementMetadata": {
97     "title": "Standards for Mathematical Practice",
98     "creator": "NGA Center and CCSSO",
99     "publisher": "NGA Center and CCSSO",
100    "license": "NGA Center for Best Practices (NGA Center) and the Council of Chief State School
101 Officers (CCSSO) Public License",
102    "licenseHolder": "NGA Center and CCSSO",
103    "status": "published",
104    "version": "1.0",
105    "jurisdiction": "US",
106    "created": "2010-06-01T",
107    "dateValid": "2010-06-01T"
108 },
109 "encodedStatementMetadata": {
110     "title": "Digital Representations of Standards for Mathematical Practice",
111     "creator": "DR",
112     "publisher": "SETDA",
113     "license": "CC-BY-3.0",
114     "licenseHolder": "SETDA",
115     "status": "draft",
116     "version": "1.0",
117     "jurisdiction": "US",
118     "created": "2013-02-11T"
119 }
120 }
121 }

```

Note: JSON last validated: 20130213. JSON instance schema validation: not validated.

Granular Decomposition Example

The example takes “CCSS.Math.Content.6.NS.C.7C Understand the absolute value of a rational number as its distance from 0 on the number line; interpret absolute value as magnitude for a positive or negative quantity in a real-world situation” and decomposes it into two statements:

1. Understand the absolute value of a rational number as its distance from 0 on the number line.
2. Interpret absolute value as magnitude for a positive or negative quantity in a real-world situation.

The example includes:

- The JSON serialization of the CCSSO statement.
- The two more granular statements.

Original Statement

- Indicating there are children (the granular statements).
- The metadata for the encoded statement (source statement metadata not included).

```

1  {
2  "learningStandardsStatement": {
3  "$schemaVersion": "GIM-CCSS 20130212",
4  "$schemaLocation": "http://publishingagent.tld/schema/learningstandardsstatement.json#",
5  "identifiers": [
6  {
7  "identifier": {
8  "idType": "CCSS Path",
9  "id": "http://corestandards.org/Math/Content/6/NS/C/7/c"
10 }
11 },
12 {
13 "identifier": {
14 "idType": "CCSS Dot",
15 "id": "CCSS.Math.Content.6.NS.C.7c"
16 }
17 },
18 {
19 "identifier": {
20 "idType": "CCSS GUID",
21 "id": "B62C1C106873438AA0126760075A65A3"
22 }
23 },
24 {
25 "identifier": {
26 "idType": "GIM UUID",
27 "id": "610f436f266e4102a3b469a2b4d63da0"
28 }
29 },
30 {
31 "identifier": {
32 "idType": "GIM Path",
33 "id": "CCSS/Math/Content/./6/Math/NS/C/7/c"
34 }
35 }
36 ]

```

```

37 },
38 "classifiers": {
39   "taxons": {
40     "initiative": "CCSS",
41     "framework": "Math",
42     "set": "Content",
43     "grade": "06",
44     "discipline": "Math",
45     "domain": "NS",
46     "cluster": "C",
47     "standard": "7",
48     "component": "c"
49   },
50   "statementType": "Component"
51 },
52 "statements": [
53   {
54     "statement": {
55       "language": "en-US",
56       "encoding": "text/html",
57       "text": "Understand the absolute value of a rational number as its distance from 0 on the
58 number line; interpret absolute value as magnitude for a positive or negative quantity in a real-world
59 situation. <i>For example, for an account balance of &ndash;30 dollars, write  $|\text{&ndash;30}| = 30$  to
60 describe the size of the debt in dollars</i>."
61     }
62   }
63 ],
64 "educationalCharacteristics": {
65   "gradeLevel": "06"
66 },
67 "relationships": [
68   {
69     "relationship": {
70       "relationType": "child-of",
71       "relatedStatement": [
72         {
73           "identifier": {
74             "idType": "CCSS Path",
75             "id": "http://corestandards.org/Math/Content/6/NS/C/7"
76           }
77         },
78         {
79           "identifier": {
80             "idType": "CCSS Dot",
81             "id": "CCSS.Math.Content.6.NS.C.7"
82           }
83         },
84         {
85           "identifier": {
86             "idType": "CCSS GUID",
87             "id": "71F96ADF450A4133A258FDACE2D3910D"
88           }
89         }
90       ]
91     }
92   }
93 ]

```

90	{
91	"identifier": {
92	"idType": "GIM Path",
93	"id": "CCSS/Math/Content/./6/Math/NS/C/7/c"
94	}
95	}
96]
97	}
98	},
99	{
100	"relationship": {
101	"relationType": "parent-of",
102	"relatedStatement": [
103	{
104	"identifier": {
105	"idType": "GIM UUID",
106	"id": "d39c073f29994ff8b60d172a27040406"
107	}
108	},
109	{
110	"identifier": {
111	"idType": "GIM Path",
112	"id": "CCSS/Math/Content/./6/Math/NS/C/7/c/1"
113	}
114	}
115]
116	}
117	},
118	{
119	"relationship": {
120	"relationType": "parent-of",
121	"relatedStatement": [
122	{
123	"identifier": {
124	"idType": "GIM UUID",
125	"id": "49e99b9b94b447e38dc25a321f9a438a"
126	}
127	},
128	{
129	"identifier": {
130	"idType": "GIM Path",
131	"id": "CCSS/Math/Content/./6/Math/NS/C/7/c/2"
132	}
133	}
134]
135	}
136	}
137	},
138	"notes": [
139	{
140	"note": {
141	"language": "en-US",
142	"encoding": "text/html",

143	"text": "Decomposition of Math/Content/6/NS/C/7/c/ into granular statements"
144	}
145	}
146],
147	"encodedStatementMetadata": {
148	"creator": "DR",
149	"publisher": "SETDA",
150	"license": "CC-BY-3.0",
151	"licenseHolder": "SETDA",
152	"status": "draft",
153	"version": "1.0",
154	"jurisdiction": "US",
155	"created": "2013-02-11T"
156	}
157	}

Granular statement “1”: Understand the absolute value of a rational number as its distance from 0 on the number line.

1	{
2	"learningStandardsStatement": {
3	"\$schemaVersion": "GIM-CCSS 20130212",
4	"\$schemaLocation": "http://publishingagent.tld/schema/learningstandardsstatement.json#",
5	"identifiers": [
6	{
7	"identifier": {
8	"idType": "GIM UUID",
9	"id": "d39c073f29994ff8b60d172a27040406"
10	}
11	},
12	{
13	"identifier": {
14	"idType": "GIM Path",
15	"id": "CCSS/Math/Content/./6/Math/NS/C/7/c/1"
16	}
17	}
18]
19	},
20	"classifiers": {
21	"taxons": {
22	"initiative": "CCSS",
23	"framework": "Math",
24	"set": "Content",
25	"grade": "06",
26	"discipline": "Math",
27	"domain": "NS",
28	"cluster": "C",
29	"standard": "7",
30	"component": "c",
31	"granular A": "1"
32	},
33	"statementType": "Granular A"
34	},

35	"statements": [
36	{
37	"statement": {
38	"language": "en-US",
39	"encoding": "text/html",
40	"text": "Understand the absolute value of a rational number as its distance from 0 on the number
41	line."
42	}
43	}
44],
45	"educationalCharacteristics": {
46	"gradeLevel": "06"
47	},
48	"relationships": [
49	{
50	"relationship": {
51	"relationType": "child-of",
52	"relatedStatement": [
53	{
54	"identifier": {
55	"idType": "CCSS Path",
56	"id": "http://corestandards.org/Math/Content/6/NS/C/7/c"
57	}
58	},
59	{
60	"identifier": {
61	"idType": "CCSS GUID",
62	"id": "B62C1C106873438AA0126760075A65A3"
63	}
64	},
65	{
66	"identifier": {
67	"idType": "GIM UUID",
68	"id": "610f436f266e4102a3b469a2b4d63da0"
69	}
70	},
71	{
72	"identifier": {
73	"idType": "GIM Path",
74	"id": "CCSS/Math/Content/./6/Math/NS/C/7/c"
75	}
76	}
77]
78	}
79	}
80],
81	"encodedStatementMetadata": {
82	"creator": "DR",
83	"publisher": "SETDA",
84	"license": "CC-BY-3.0",
85	"licenseHolder": "SETDA",
86	"status": "draft",
87	"version": "1.0",

88	"jurisdiction": "US",
89	"created": "2013-02-11T"
90	}
91	}

Granular statement “2”: Interpret absolute value as magnitude for a positive or negative quantity in a real-world situation.

1	{
2	"learningStandardsStatement": {
3	"\$schemaVersion": "GIM-CCSS 20130212",
4	"\$schemaLocation": "http://publishingagent.tld/schema/learningstandardsstatement.json#",
5	"identifiers": [
6	{
7	"identifier": {
8	"idType": "GIM UUID",
9	"id": "49e99b9b94b447e38dc25a321f9a438a"
10	}
11	},
12	{
13	"identifier": {
14	"idType": "GIM Path",
15	"id": "CCSS/Math/Content/./6/Math/NS/C/7/c/2"
16	}
17	}
18]
19	},
20	"classifiers": {
21	"taxons": {
22	"initiative": "CCSS",
23	"framework": "Math",
24	"set": "Content",
25	"grade": "06",
26	"discipline": "Math",
27	"domain": "NS",
28	"cluster": "C",
29	"standard": "7",
30	"component": "c",
31	"granular A": "2"
32	},
33	"statementType": "Granular A"
34	},
35	"statements": [
36	{
37	"statement": {
38	"language": "en-US",
39	"encoding": "text/html",
40	"text": "Interpret absolute value as magnitude for a positive or negative quantity in a real-world
41	situation."
42	}
43	}
44],
45	"educationalCharacteristics": {

```

46   "gradeLevel": "06"
47   },
48   "relationships": [
49     {
50       "relationship": {
51         "relationType": "child-of",
52         "relatedStatement": [
53           {
54             "identifier": {
55               "idType": "CCSS Path",
56               "id": "http://corestandards.org/Math/Content/6/NS/C/7/c"
57             }
58           },
59           {
60             "identifier": {
61               "idType": "CCSS GUID",
62               "id": "B62C1C106873438AA0126760075A65A3"
63             }
64           },
65           {
66             "identifier": {
67               "idType": "GIM UUID",
68               "id": "610f436f266e4102a3b469a2b4d63da0"
69             }
70           },
71           {
72             "identifier": {
73               "idType": "GIM Path",
74               "id": "CCSS/Math/Content/./6/Math/NS/C/7/c"
75             }
76           }
77         ]
78       }
79     }
80   ],
81   "encodedStatementMetadata": {
82     "creator": "DR",
83     "publisher": "SETDA",
84     "license": "CC-BY-3.0",
85     "licenseHolder": "SETDA",
86     "status": "draft",
87     "version": "1.0",
88     "jurisdiction": "US",
89     "created": "2013-02-11T"
90   }
91 }

```

Note: JSON last validated: 20130213. JSON instance schema validation: not validated.

State 15% Extension Example

The example takes the Kansas ELA Literacy Learning Anchor Standard addition – “2. Use meta-cognitive strategies to monitor literacy learning progress” – and places it as a state extension to the CCSS ELA Anchor Standards.

The example includes:

- The JSON serialization of the Kansas statement (the KS ID is unknown).
- The relationship showing the statement is an extension of the original CCSS statements.
- Ownership and rights to the statement are held by the State.
- The metadata for the source and encoded statement.

```

1  {
2  "learningStandardsStatement": {
3    "$schemaVersion": "GIM-CCSS 20130212",
4    "$schemaLocation": "http://publishingagent.tld/schema/learningstandardsstatement.json#",
5    "identifiers": [
6      {
7        "identifier": {
8          "idType": "GIM UUID",
9          "id": "5b37819c9b0d43eaafca4a86fb7091e7"
10       }
11     },
12     {
13       "identifier": {
14         "idType": "GIM Path",
15         "id": "KSCCS/ELA-Literacy/CCRA/KS-LL/K-12/././2"
16       }
17     },
18     {
19       "identifier": {
20         "idType": "KS ID",
21         "id": ""
22       }
23     }
24   ],
25   "classifiers": {
26     "taxons": {
27       "initiative": "KSCCS",
28       "framework": "ELA-Literacy",
29       "set": "CCRA",
30       "strand": "KS-LL",
31       "grade": "K-12",
32       "standard": "2"
33     },
34     "statementType": "Standard"
35   },
36   "statements": [
37     {
38       "statement": {
39         "language": "en-US",
40         "encoding": "text/html",

```

41	"text": "Use meta-cognitive strategies to monitor literacy learning progress."
42	}
43	}
44],
45	"educationalCharacteristics": {
46	"gradeLevel": [
47	"KG",
48	"01",
49	"02",
50	"03",
51	"04",
52	"05",
53	"06",
54	"07",
55	"08",
56	"09",
57	"10",
58	"11",
59	"12"
60]
61	},
62	"relationships": [
63	{
64	"relationship": {
65	"relationType": "child-of",
66	"relatedStatement": [
67	{
68	"identifier": {
69	"idType": "CCSS Path",
70	"id": "http://corestandards.org/ELA-Literacy/CCRA"
71	}
72	},
73	{
74	"identifier": {
75	"idType": "GIM UUID",
76	"id": "7598208fc0734f068992fa179ab10f39"
77	}
78	},
79	{
80	"identifier": {
81	"idType": "GIM Path",
82	"id": "CCSS/ELA-Literacy/CCRA"
83	}
84	}
85]
86	}
87	}
88],
89	"sourceStatementMetadata": {
90	"author": "KSDE",
91	"creator": "KSDE",
92	"publisher": "KSDE",
93	"license": "© 2011 KSDE",

94	"licenseHolder": "KSDE",
95	"status": "published",
96	"version": "1.0",
97	"jurisdiction": "KS",
98	"created": "2012-03-26T"
99	},
100	"encodedStatementMetadata": {
101	"creator": "DR",
102	"publisher": "SETDA",
103	"license": "CC-BY-3.0",
104	"licenseHolder": "SETDA",
105	"status": "draft",
106	"version": "1.0",
107	"jurisdiction": "KS",
108	"created": "2013-02-11T"
109	}
110	}
111	}

Note: JSON last validated: 20130219. JSON instance schema validation: not validated.

Versioning Example

Note: The example is contrived.

The example shows how a new version of a statement is defined.

- The metadata for the source statement shows the new version number (but the encoding is a new version 1.0).
- There are multiple relations from the current version to the prior version.

```

1  {
2  "learningStandardsStatement": {
3    "$schemaVersion": "GIM-CCSS 20130212",
4    "$schemaLocation": "http://publishingagent.tld/schema/learningstandardsstatement.json#",
5    "identifiers": [
6      {
7        "identifier": {
8          "idType": "CCSS Path",
9          "id": "http://corestandards.org/ELA-Literacy/RL/7/5"
10       }
11     },
12     {
13       "identifier": {
14         "idType": "CCSS Dot",
15         "id": "CCSS.ELA-Literacy.RL.7.5"
16       }
17     },
18     {
19       "identifier": {
20         "idType": "CCSS GUID",
21         "id": "62289E99C7BB4ED9BA8B995F9ED54DB8"
22       }
23     },
24     {
25       "identifier": {
26         "idType": "GIM UUID",
27         "id": "4691b0642c3a4d42a4264b11ee0f8be5"
28       }
29     },
30     {
31       "identifier": {
32         "idType": "GIM Path",
33         "id": "CCSS/ELA-Literacy/Content/RL/7/./././5"
34       }
35     }
36   ],
37   "classifiers": {
38     "taxons": {
39       "initiative": "CCSS",
40       "framework": "ELA-Literacy",
41       "set": "Content",
42       "strand": "RL",
43       "grade": "07",

```

44	"standard": "5"
45	},
46	"statementType": "Standard"
47	},
48	"statements": [
49	{
50	"statement": {
51	"language": "en-US",
52	"encoding": "text/html",
53	"text": "Analyze and explain how the form or structure (e.g., soliloquy, sonnet) of a drama or
54	poem contributes to its meaning"
55	}
56	}
57],
58	"educationalCharacteristics": {
59	"gradeLevel": "07"
60	},
61	"relationships": [
62	{
63	"relationship": {
64	"relationType": "child-of",
65	"relatedStatement": [
66	{
67	"identifier": {
68	"idType": "CCSS Path",
69	"id": "http://corestandards.org/ELA-Literacy/RL/7"
70	}
71	},
72	{
73	"identifier": {
74	"idType": "CCSS Dot",
75	"id": "CCSS.ELA-Literacy.RL.7"
76	}
77	},
78	{
79	"identifier": {
80	"idType": "CCSS GUID",
81	"id": "48611D0AFB9C47e594C4BDC4349248D6"
82	}
83	},
84	{
85	"identifier": {
86	"idType": "GIM Path",
87	"id": "CCSS/ELA-Literacy/Content/RL/7"
88	}
89	}
90]
91	}
92	},
93	{
94	"relationship": {
95	"relationType": "version-of",
96	"relatedStatement": [

97	{
98	"identifier": {
99	"idType": "CCSS GUID",
100	"id": "0FAC40650F514a459951A657F942B845"
101	}
102	},
103	{
104	"identifier": {
105	"idType": "GIM UUID",
106	"id": "5c9d6ea279b54a2a811ac8a1b39ce556"
107	}
108	}
109]
110	}
111	},
112	"notes": [
113	{
114	"note": {
115	"language": "en-US",
116	"encoding": "text/html",
117	"text": "Newer version of CCSS/ELA-Literacy/Content/RL/7/5"
118	}
119	}
120],
121	"sourceStatementMetadata": {
122	"title": "Standards for English Language Arts & Literacy in History/Social Studies, Science, and
123	Technical Subjects",
124	"creator": "NGA Center and CCSSO",
125	"publisher": "NGA Center and CCSSO",
126	"license": "NGA Center for Best Practices (NGA Center) and the Council of Chief State School
127	Officers (CCSSO) Public License",
128	"licenseHolder": "NGA Center and CCSSO",
129	"status": "published",
130	"version": "2.0",
131	"jurisdiction": "US",
132	"created": "2012-05-18T",
133	"dateValid": "2012-06-11T"
134	},
135	"encodedStatementMetadata": {
136	"creator": "DR",
137	"publisher": "SETDA",
138	"license": "CC-BY-3.0",
139	"licenseHolder": "SETDA",
140	"status": "draft",
141	"version": "1.0",
142	"jurisdiction": "US",
143	"created": "2013-02-11T"
144	}
145	}
146	}
147	}

Note: JSON last validated: 20130213. JSON instance schema validation: not validated.

Collection Example

The example shows a collection containing a statement and its granular decomposition. The statements have been shortened to include only one identifier, the type classifier and the statement text for brevity in presentation.

```

1  {
2    "learningStandardsStatementCollection": {
3      "$schemaVersion": "GIM-CCSS 20130212",
4      "$schemaLocation": "http://publishingagent.tld/schema/learningstandardsstatementcollection.json",
5      "totalStatements": 3,
6      "statements": [
7        {
8          "learningStandardsStatement": {
9            "identifiers": [
10             {
11               "identifier": {
12                 "idType": "GIM Path",
13                 "id": "CCSS/Math/Content/.6/Math/NS/C/7/c"
14               }
15             }
16           ],
17           "classifiers": {
18             "statementType": "Component"
19           },
20           "statements": [
21             {
22               "statement": {
23                 "text": "Understand the absolute value of a rational number as its distance from 0 on the
24 number line; interpret absolute value as magnitude for a positive or negative quantity in a real-world
25 situation. <i>For example, for an account balance of &ndash;30 dollars, write |&ndash;30| = 30 to describe
26 the size of the debt in dollars</i>."
27               }
28             }
29           ]
30         }
31       ],
32       {
33         "learningStandardsStatement": {
34           "identifiers": [
35             {
36               "identifier": {
37                 "idType": "GIM Path",
38                 "id": "CCSS/Math/Content/.6/Math/NS/C/7/c/1"
39               }
40             }
41           ],
42           "classifiers": {
43             "statementType": "Granular A"
44           },
45           "statements": [
46             {
47               "statement": {

```

```

48         "text": "Understand the absolute value of a rational number as its distance from 0 on the
49 number line."
50     }
51 }
52 ]
53 }
54 },
55 {
56     "learningStandardsStatement": {
57         "identifiers": [
58             {
59                 "identifier": {
60                     "idType": "GIM Path",
61                     "id": "CCSS/Math/Content/.6/Math/NS/C/7/c/2"
62                 }
63             }
64         ],
65         "classifiers": [
66             {
67                 "classifier": {
68                     "statementType": "Granular A"
69                 }
70             }
71         ],
72         "statements": {
73             "statement": {
74                 "text": "Interpret absolute value as magnitude for a positive or negative quantity in a real-
75 world situation."
76             }
77         }
78     }
79 }
80 ]
81 }
82 }
83

```

Note: JSON last validated: 20130213. JSON instance schema validation: not validated.

Linked Data Example

The example shows a basic statement with [JSON-LD] markup (JSON-LD names are prefixed with "@"):

- The "@context" object links to the JSON-LD definition of the standards statement schema.
- The second "@context" object links to SKOS.
- The top-level "@id" object provides an IRI for the standards statement.
- The "@type" object declares that the JSON object is a learning standards statement.
- The "statement" object uses "@language" and "@value" in the description of the "text" object instead of a separate "language" object.
- The metadata includes a "@context" object link to the definition of Dublin Core terms.
- The "dc." prefix indicates which objects in the metadata come from the Dublin Core schema.
- The "skos." prefix indicates the item comes from the SKOS vocabulary.
- The embedded "@id" object in the metadata provides a separate IRI for the metadata.

```

1  {
2  "context": {
3    "skos": "http://skos.tld/"
4  },
5  "@id": "http://publishingagent.tld/statement/730d6682f09846f8a959b58251fcc8e2",
6  "@type": "learningStandardsStatement",
7
8  "learningStandardsStatement": {
9    "$schemaVersion": "GIM-CCSS 20130212",
10   "$schemaLocation": "http://publishingagent.tld/schema/learningstandardsstatement.json#",
11   "identifiers": [
12     {
13       "identifier": {
14         "idType": "CCSS Path",
15         "id": "http://corestandards.org/Math/Content/K/CC/A/1"
16       }
17     },
18     {
19       "identifier": {
20         "idType": "CCSS GUID",
21         "id": "B62C1C106873438AA0126760075A65A3"
22       }
23     }
24   ],
25   "classifiers": {
26     "taxons": {
27       "initiative": "CCSS",
28       "framework": "Math",
29       "set": "Content",
30       "grade": "KG",
31       "discipline": "Math",
32       "domain": "CC",
33       "cluster": "A",
34       "standard": "1"
35     },
36     "statementType": "Standard"
37   },
38   "statements": [

```

```

39     {
40         "statement": {
41             "text": {
42                 "@value": "Count to 100 by ones and by tens.",
43                 "@language": "en-US"
44             }
45         }
46     },
47 ],
48 "educationalCharacteristics": {
49     "gradeLevel": "KG"
50 },
51 "relationships": [
52     {
53         "relationship": {
54             "relationType": "skos.child-of",
55             "relatedStatement": {
56                 "identifier": {
57                     "idType": "CCSS GUID",
58                     "id": "4F4106218F834258BCDDB7EB39806880"
59                 }
60             }
61         }
62     }
63 ],
64 "encodedStatementMetadata": {
65     "@context": {
66         "dc": "http://purl.org/dc/elements/1.1"
67     },
68     "@id": "http://publishingagent.tld/statement/62367d5f3ea64211-8c0a29d2ee0447f8",
69     "dc.creator": "DR",
70     "dc.publisher": "SETDA",
71     "license": "CC-BY-3.0",
72     "licenseHolder": "SETDA",
73     "status": "draft",
74     "version": "1.0",
75     "jurisdiction": "US",
76     "dc.created": "2012-12-01T"
77 }
78 }
79 }

```

Note: JSON last validated: 20130211. JSON instance schema validation: not validated.

Note: The JSON-LD Context has not been defined. JSON-LD has not been verified.

Schemata Design Decisions

Note: This section is informative.

General Design Decisions

Identifiers, statement text, notes and relationships are arrays, permitting multiple values, consistent with the information model.

An identifier is a structured object consisting of the type and value. Typing the identifier adds information needed to process the identifier. The type comes from a controlled vocabulary to permit validation. The format of the value depends on the type.

Statement classification is an object with an arbitrary set of classification key-value pairs. Values may be simple values, arrays or objects. The flexible object structure permits any type of classification.

Educational characteristics is an object with an arbitrary set of key-value pairs, one per characteristic. Values may be simple values, arrays or objects. The flexible object structure permits any type of characteristic.

A relationship is a structured object consisting of the type and the related statement identifier. Typing the relationship provides the semantics to define the relationship. The type comes from a controlled vocabulary to permit validation. The related statement identifier may be a single identifier, or an array of values. If the target statement of the relationship has multiple identifiers, these should be placed into the array for a single relationship – this implies the identifiers all refer to the same statement and are functionally equivalent (avoids additional data indicating that the identifiers refer to the same statement). Different targets of the same type (e.g., different children) should be placed in different relationship objects.

There are two sets of metadata, one for the source statement and one for the encoded statement. Each is an object with an arbitrary set of metadata key-value pairs. Values may be simple values, arrays or objects. The flexible object structure permits any type of metadata. Both sets should use the same attributes.

Required items define a minimal description of a statement: the identifier, the classifier and the statement text. In a collection, only an identifier is needed, assuming the identifier can be resolved into a locator to return the rest of the statement.

JSON object names are formatted in lower camel case as convention.

JSON semantics of ordered arrays are maintained.

Arrays contain a list of named objects (i.e., "array": [{"object": {"a": 1, "b": 2}}, {"object": {"c": 1, "d": 2}}] not "array": [{"a": 1, "b": 2}, {"c": 1, "d": 2}]). Adding objects defines types.

All items are duck typed – anything that is a simple value could be replaced by an object. Converting a simple value to an object provides extensibility.

If [JSON-LD] is adopted, languages should be specified through using a "@language" property.

A schema version is included in an instance of a JSON object to support verification.

Additional metadata about the JSON object (e.g., producer, when produced) is included in the encoded statement metadata, avoiding the need for additional top-level metadata attributes.

JSON Schemata Decisions

To support evolution of the schemata, a schema version attribute is included in each schema.

To provide access to the schema, the id attribute is included in each schema. The value is the URI of where the schema is hosted by the publishing agent.

For documentation, all items include the title and description attribute.

For referencing, all items include an id attribute.

There are no specified constraints on string or array sizes. Implementations may impose constraints. Strings, especially those holding identifiers, URIs, or IRIs should support at least 4000 characters (UTF8 character encoding).

There are no specific constraints on attribute uniqueness. Uniqueness may be enforced by semantic or implementation constraints.

Default values are not included. Default values are meaningless for the required attributes.

[JSON reference] is used to decompose the entire schema into parts permitting reuse.

Learning Standards Statement Schema Decisions

To permit reuse and avoid defining two similar schemata, the only element required by the schema is the identifier element. The schema is included by reference in the Learning Standards Statement Collection schema. A learning standards statement included in a learning standards statement collection only requires an identifier. A standalone learning standards statement must include a statement and a classifier; these are modeled as additional requirements outside of the schema.

For simplicity, both the source statement metadata and encoded statement metadata use the same subschema.

To provide a flexible structure, the definition of the properties of the metadata subschema is not specified in the schema. A separate metadata schema used for with GIM-CCSS statements is defined in [Best Practices].

For simplicity, the vocabulary for the identifier type is not specified in the schema. A vocabulary used for the GIM-CCSS statements is defined in [Best Practices].

To provide a meaningful standards statement, an identifier must include an id.

For simplicity, the structure of the taxon object for classifiers and a vocabulary for taxon values are not specified in the schema. An object structure and vocabulary used for the GIM-CCSS statements are defined in [Best Practices].

For simplicity, the vocabulary for statement types is not specified in the schema. A vocabulary used for the GIM-CCSS statements is defined in [Best Practices].

For simplicity, only one educational characteristic, grade level, is defined in the schema. Other attributes are permitted.

For simplicity, the vocabulary for grade level is not specified in the schema. A vocabulary used for the GIM-CCSS statements is defined in [Best Practices].

For flexibility, grade level is defined to be either a single value or an array of values.

To provide a meaningful standards statement, if the statement element is present, it must contain a text attribute.

To provide a meaningful standards statement, if the note element is present, it must contain a text attribute.

For simplicity, the vocabulary for language is not specified in the schema. A vocabulary used for the GIM-CCSS statements is defined in [Best Practices].

For simplicity, the vocabulary for statement text encoding is not specified in the schema. A vocabulary used for the GIM-CCSS statements is defined in [Best Practices].

For simplicity, constraints on the encoding of a text attribute (e.g., that it be HTML5) is not included. A profile of HTML5 used for the GIM-CCSS statements is defined in [Best Practices].

To provide a meaningful standards statement, if a relationship element is present, it must contain at least one id of a related statement.

For simplicity in a JSON instance, the identifier element permits either a single identifier or an array of identifiers.

For simplicity, the vocabulary for relation type is not specified in the schema. A vocabulary used for the GIM-CCSS statements is defined in [Best Practices].

Learning Standards Statement Collection Schema Decisions

There is no requirement that the statements in the collection be unique.

Normative References

For dated references, only the edition cited applies. For undated references, the most recent edition applies.

[JSON Schema Core] Galiegue, F., Zyp, K., and Court, G., “JSON Schema: core definitions and terminology json-schema-core”, January 2013.

[<http://json-schema.org/latest/json-schema-core.html>]

[JSON Schema Validation] Galiegue, F., Zyp, K., and Court, G., “JSON Schema: interactive and non interactive validation json-schema-validation”, January 2013.

[<http://json-schema.org/latest/json-schema-validation.html>]

[RFC 2119] Bradner, S., “Key words for use in RFCs to Indicate Requirement Levels”, IETF RFC 2119, March 1997.

[<http://tools.ietf.org/html/rfc2119>]

[RFC 3339] Klyne, G., “Date and Time on the Internet: Timestamps”, IETF RFC 3339, July 2002.

[<http://tools.ietf.org/html/rfc3339>]

[RFC 3986] Berners-Lee, T., Fielding, R., and Masinter, L., “Uniform Resource Identifier (URI)”, IETF RFC 3986, January 2005.

[<http://tools.ietf.org/html/rfc3986>]

[RFC 3987] Duerst, M., and Suignard, M., “Internationalized Resource Identifiers (IRIs)”, IETF RFC 3987, January 2005.

[<http://tools.ietf.org/html/rfc3987>]

[RFC 4288] Freed, N., and Klensin, J., “Media Type Specifications and Registration Procedures”, IETF RFC 4288, December 2005.

[<http://tools.ietf.org/html/rfc4288>]

[RFC 4289] Freed, N., and Klensin, J., “Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures”, IETF RFC 4289, December 2005.

[<http://tools.ietf.org/html/rfc4289>]

[RFC 4627] Crockford, D., “The application/json Media Type for JavaScript Object Notation (JSON)”, IETF RFC 4627, July 2006.

[<http://tools.ietf.org/html/rfc4627>]

Informative References

Note: This section is informative.

[APIs] *Learning Standards Digital Representation Specification: RESTful APIs*, Granular Identifiers and Metadata for CCSS (GIM CCSS) Project.

[JSON LD] *JSON-LD Syntax 1.0*.
[<http://json-ld.org/spec/latest/json-ld-syntax/>]

[JSON Pointer] Bryan, P., Zyp, K., Nottingham, M., “JSON Pointer”, IETF Internet-Draft, September 2012.
[<http://tools.ietf.org/html/draft-ietf-appsawg-json-pointer-04>]

[JSON Reference] Bryan, P., Zyp, K., “JSON Reference”, IETF Internet-Draft, September 2012.
[<http://tools.ietf.org/html/draft-pbryan-zyp-json-ref-03>]

[JSON Schema V3] Zyp, K., and Court, G., “A JSON Media Type for Describing the Structure and Meaning of JSON Documents”, IETF Internet-Draft, November 2010.
[<http://tools.ietf.org/html/draft-zyp-json-schema-03>]

[JSON Schema Generator] *JSON Schema.net*.
[<http://www.jsonschema.net/>]

[JSON Schema Validator] *JSON Schema Validator*.
[<http://json-schema-validator.herokuapp.com/syntax.html>]

[JSON Validator] *JSONLint – The JSON Validator*.
[<http://jsonlint.com/>]

[Scope] *Scope, Technical Requirements, Approaches, and Recommendations*, Granular Identifiers and Metadata for CCSS (GIM CCSS) Project.

Annex: JSON Schema Validation Extensions References

To model the schemata, the [JSON Schema Validation] has been extended.

The additional version attribute is added:

- The attribute SHALL be "version"
- The value MUST be a string.
- The version is similar to the other metadata keyword attributes.

The additional schema license attribute is added:

- The attribute SHALL be "schemaLicense"
- The value MUST be a string
- The schema license is similar to the other metadata keyword attributes.

The set of formats is extended to include:

- RFC 3987 IRI
 - An IRI that conforms to [RFC 3987].
 - This format includes URIs and URLs and extends the URL format defined in [JSON Schema Validation] which supports [RFC 3986].
 - The format attribute SHALL be "iri".
 - A string instance is valid for this attribute if it is a valid IRI, according to [RFC3986].
- point in time
 - A point in time that conforms to [RFC 3339] with additional pattern constraints:
 - A UTC time with pattern YYYY-MM-DDThh:mm:ss.SZ
 - A local time with pattern YYYY-MM-DDThh:mm:ss.s+/-HH:MM
 - For either the UTC or local time the fractional part of the second [.s] is optional.
 - For local time, the minute part of the offset [:MM] is optional.
 - The format attribute SHALL be "pointInTime".
 - A string instance is valid for this attribute if it is a valid pattern described above.
- start-end date range
 - An inclusive time range specified by the start and end time points.
 - A range with pattern point in time / point in time
 - The format attribute SHALL be "dateRange".
 - A string instance is valid for this attribute if it is a valid pattern as described above.

GIM CCSS Project Documentation

Note: This section is informative.

Scope, Technical Requirements, Approaches, and Recommendations

Learning Standards Digital Representation Specification: JSON Serialization Schemata [this document]

Learning Standards Digital Representation Specification: XML Serialization Schemata

Learning Standards Digital Representation Specification: RESTful API

Learning Standards Digital Representation Specification: Publishing Agent Design and Requirements

Change Log

Date	Version	Author	Notes
20121202	0.3	DR	Partial draft, unedited. Missing complete statement and vocabulary schemata. Missing examples. Release to management group.
20121204	0.4	DR	Partial draft, edited. Release to working group for review.
20130303	0.6	DR	Draft specification. Vocabulary schema omitted. Draft for Public review.
20130912	1.0	DR	V1.0 Public release.