SmarterApp Item Response Format Specification

V 1.00: Public Release

Daniel Rehak
2015-04-22

Introduction

This document describes the format of the student response data for an individual Smarter Balanced assessment item. It includes the description of the response format for a select set of item types:

- Evidence-Based Selected Response Item (EBSR)
- Equation Item (EQ)
- Grid Item (GI)
- Hot Text Item (HTQ)
- Match Item (MI)
- Short Answer Item (SA)
- Table Item (TI)
- Writing Extended Response Item (WER)

Assessment item response data for each of the different assessment item types is in a similar format:

- The data is a string of characters.
- The data starts with a set of attributes that is common to all items. These attributes describe the item (item type and id) and its score.
- These attributes are followed by the detailed item response data.
- The format of this response data is specific to the type of item.

The first section below describes the overall syntax of the response format and the common attributes. The detailed item response data for each of the different types of assessment items are described in subsequent sections.

Note, the document does not describe how the individual item response data is stored in a file or exchanged between systems. It does not describe how collections of responses are organized, stored or exchanged. It only describes the syntax of a (single) actual response.

Common Item Response Format Attributes

The item response is a string of characters (encoded as UTF-8). The response consists of three (3) attributes:

1. The *item identifier* or item key. The item key is a string of characters that uniquely identifies the item.
2. The *item score*. The score is an integer. The range of possible values is item type specific:
   - -1: the item was not scored
   - 0: item score
   - 1: item score
   - 2: item score
   - ...
   - n: item score (the maximum value for the score is item specific)
3. The detailed *item response data*. The format of the response data is specific to the type of item. The data is expressed in XML; either general XML, or a specific XML vocabulary, e.g., HTML or MathML. The details of the XML for the different item types is presented for each item type in individual sections below. The presentation describes the syntax of the response data, illustrates how the response data aligns with an item description and shows several examples of response data.

Item Code	Item Type
EBSR	Evidence-Based Selected Response Item
EQ	Equation Item
GI	Grid Item
HTQ	Hot Text Item
MI	Match Item
SA	Short Answer Item
TI	Table Item
WER	Writing Extended Response Item

The *tilda* character, i.e., "~" (Unicode U+007E, ASCII 126 decimal) is used as a delimiter to separate the attributes.

White space (space, tab, line feed/carriage return) characters are only present in the detailed item response data. No white space is present in the first three attribute values or around the delimiters.

A fictitious example each of the different item responses formats (with partial detailed response data) are shown below.

Item Code	Partial Response Data
EBSR	123-45600~0~<itemResponse>...</itemResponse>
EQ	123-56700~0~0~<response>$...$</response>
GI	123-67800~0~<?xml...><AnswerSet>...</AnswerSet>
HTQ	123-78900~0~<itemResponse>...</itemResponse>
MI	123-89100~0~<itemResponse>...</itemResponse>
SA	123-91200~-1~ ...
TI	123-12300~0~<responseSpec>...</responseSpec>
WER	123-23400~-1~ ...

Evidence-Based Selected Response Item (EBSR) Response Format

The response format for an EBRS item consists of:

- The item identifier
- The item score:
  - 0: item score
  - 1: item score
  - 2: item score
  - ...
  - n: item score (the maximum value for the score is item specific)
- Response data in XML.

Response Data XML
The root of the XML response data is a single `<itemResponse>` element without attributes.

- Nested within `<itemResponse>` is a sequence of 1 or more `<response>` elements. Each `<response>` element has a required `id` attribute; the attribute value is a string that identifies the particular response. The value of the `id` is unique for each `<response>` for the item. The value of the `id` will match the value of the `responseIdentifier` attribute of one of the item's `<choiceInteraction>` element.

  - Nested within the `<response>` element is a sequence of 0 or more `<value>` elements (the `<response>` element may be empty, but if empty it will have both an opening and closing tag). The `<value>` elements will not have any attributes.

    Each `<value>` element will contain a string that is the value of the particular response; the string value will be the `identifier` attribute of the selected `<simpleChoice>` element.

The response XML does not begin with a standard XML declaration. Assume that the XML conforms to XML 1.0 and is encoded in UTF-8, i.e., if present, the XML declaration would be:

```
<?xml version="1.0", encoding="UTF-8"?>
```

Elements and attributes not described above should not appear in the XML. Common XML features, e.g., comments, white space, character encodings, may appear in the XML.

Response Data XML Example
An example of the nested XML element structure is shown below.

```
<itemResponse>
      <response id="EBSR1">
            <value>A</value>
      </response>
      <response id="EBSR2">
            <value>C</value>
            <value>A</value>
      </response>
</itemResponse>
```

In the example:

- For the item interaction `<choiceInteraction responseIdentifier="EBSR1">`
  - the student has selected `<simpleChoice identifier="A">`
- For the item interaction `<choiceInteraction responseIdentifier="EBSR2">`
  - the student has selected `<simpleChoice identifier="C">`
  - the student has selected `<simpleChoice identifier="A">`

Assessment Item – Response Data Alignment

An example QTI choice interaction within an item is shown below. It illustrates how the identifiers in the response data align with the identifiers in the item. Highlighted identifiers in the item correspond to potential identifiers in the response data.

```
...
<qti spec="itemBody">
       <itemBody>
              <p>Two Interactions</p>
              <choiceInteraction responseIdentifier="EBSR1" shuffle="false"
              maxChoices="0">
                     <prompt><p>Part 1</p>
                     <simpleChoice identifier="A"><p>First choice</p>
                     </simpleChoice>
                     <simpleChoice identifier="B"><p>Second choice</p>
                     </simpleChoice>
                     <simpleChoice identifier="C"><p>Third choice</p>
                     </simpleChoice>
              </choiceInteraction>
              <choiceInteraction responseIdentifier="EBSR2" shuffle="false"
              maxChoices="0">
                     <prompt><p>Part 2</p>
                     <simpleChoice identifier="A"><p>First choice</p>
                     </simpleChoice>
                     <simpleChoice identifier="B"><p>Second choice</p>
                     </simpleChoice>
                     <simpleChoice identifier="C"><p>Third choice</p>
                     </simpleChoice>
              </choiceInteraction>
       </itemBody>
</qti>
...
```

Example Item Responses

Fictitious examples of different item responses for EBSR items are shown below.

```
123-45601~0~<itemResponse><response id="EBSR1"><value>A</value></response>
<response id="EBSR2"><value>B</value></response></itemResponse>

123-45602~0~<itemResponse><response id="EBSR1"><value>A</value></response>
<response id="EBSR2"><value>C</value><value>A</value></response></itemResponse>

123-45603~0~<itemResponse><response id="EBSR1"></response>
<response id="EBSR2"><value>C</value><value>A</value></response></itemResponse>

123-45604~0~<itemResponse><response id="EBSR1"></response>
<response id="EBSR2"></response></itemResponse>
```

Equation Item (EQ) Response Format

The response format for an EQ item consists of:

- The item identifier
- The item score:
  - 0: item score
  - 1: item score
  - 2: item score
  - ...
  - n: item score (the maximum value for the score is item specific)
- MathML representing an equation response data wrapped in XML.

Response Data XML
The root of the XML response data is a single `<response>` element without attributes.

- Nested within `<response>` is 1 or more `<math>` elements. The `<math>` element may include a namespace attribute to specify MathML, e.g., `xmlns="http://www.w3.org/1998/Math/MathML"` Inclusion of the namespace implies that all nested MathML elements do not need to be namespaced.
  - Nested within `<math>` is a single `<mstyle>` element. The `<mstyle>` element may be empty.
    - Nested within `<mstyle>` is the equation that the student entered using any of the MathML elements. MathML elements may include MathJax class attributes.

The response XML does not begin with a standard XML declaration. Assume that the XML conforms to XML 1.0 and is encoded in UTF-8, i.e., if present, the XML declaration would be:
`<?xml version="1.0", encoding="UTF-8"?>`

Elements and attributes not described above should not appear in the XML. Common XML features, e.g., comments, white space, character encodings, may appear in the XML.

Response Data XML Example
An example of the nested XML element structure is shown below:

```
<response>
	<math xmlns="http://www.w3.org/1998/Math/MathML">
		<mstyle displaystyle="true">
			<mn>2</mn>
			<mo>&#x2212;</mo>
			<mn>4</mn>
		</mstyle>
	</math>
</response>
```

The example represents the equation:

$$2 - 4$$

Assessment Item – Response Data Alignment
There are no elements or attributes in the item that are aligned with elements and attributes in the response.

Example Item Responses
Fictitious examples of different item responses for EQ items are shown below.

```
123-56723~0~<response><mathxmlns="http://www.w3.org/1998/Math/MathML"><mstyle>
<mn>2</mn><mo>&#x2212;</mo><mn>4</mn></mstyle></math></response>
<response id="2"><value>5</value></response></itemResponse>

123-56724~0~<response><mathxmlns="http://www.w3.org/1998/Math/MathML"><mstyle>
<mrow><mn>2</mn><mo>+</mo><mn>2</mn><mo>=</mo><mn>3</mn><mo>+</mo><mn>1</mn><mrow>
<mstyle></math></response>

123-56725~0~<response><mathxmlns="http://www.w3.org/1998/Math/MathML"><mstyle>
<mn>0</mn></mstyle></math></response>
```

Grid Item (GI) Response Format

The response format for a GI item consists of:

- The item identifier
- The item score:
  - 0: item score
  - 1: item score
  - 2: item score
  - ...
  - n: item score (the maximum value for the score is item specific)
- Response data wrapped in XML.

Response Data XML
The XML begins with a standard XML declaration `<?xml version="1.0", encoding="UTF-8"?>`

The root of the XML response data is a single `<AnswerSet>` element without attributes.

- Nested within `<AnswerSet>` is a single `<Question>` element. Each `<Question>` element has an `id` attribute that may be empty. The `id` attribute can be ignored.
  - Nested within `<Question>` is 1 or more `<QuestionPart>` elements. Each `<QuestionPart>` element has a required `id` attribute; the attribute value is a string that identifies the particular response. The value of the `id` is unique for each `<QuestionPart>` for the item. The value of the `id` will match the value of the `id` attribute of the item's `<QuestionPart>` element.
    - Nested within `<QuestionPart>` is a single `<ObjectSet>` element without attributes. The `<ObjectSet>` element will be the first element within `<QuestionPart>.`
      - Nested within `<ObjectSet>` is an optional `<RegionGroupObject>` element with 2 required attributes:
        - `name` – the name of the RegionGroup. The value of the `name` is unique for each `<RegionGroupObject>` for the item. The value of the `name` will match the value of the `name` attribute of one of the item's `<RegionGroup>` element.
        - `numselected` – the number of selected Regions. The value of `numselected` will be the same as the number of `<RegionObject>` elements with an attribute value of `isselected="true"`
        - Nested within `<RegionGroupObjects>` is 1 or more `<RegionObject>` elements. Each `<RegionObject>` element is empty and has two required attributes:
          - `name` – the name of the Region. The value of the `name` is unique for each `<RegionObject>` for the item. The value of the `name` will match the value of the `name` attribute of one of the item's `<Region>` element and will match the value of the `region` attribute of one of the `<Include>` elements in the selected `<RegionGroup>` element.
          - `isselected` – indicates if the Region was selected. The value is `"true"`or `"false"`.

- Nested within <ObjectSet> is 0 or more <AtomicObject> elements without attributes.
    - The <AtomicObject> element will contain a string defining the atomic object.
        - The string contains the *name* of the object followed by the the x-y coordinate pairs of the location of the object.
        - The value of the *name* will match the value of one of the <Label> elements in the item's <IconSpec> element.
        - The coordinate pair consists of an *x* and *y* coordinate, each an integer value (pixels) of the coordinate location on the grid, delimited by a comma (Unicode U+002C, ASCII 44 decimal, , ,).
        - NOTE: Y is measured **upward**.
        - The coordinate pair is enclosed in parenthesis () (Unicode U+0028, ASCII 40 decimal, (and Unicode U+0029, ASCII 41 decimal,)).
        - The entire string is enclosed in curly braces {} (Unicode U+007B, ASCII 123 decimal, } and Unicode U+007D, ASCII 123 decimal, }).
        - Example: {3(234,45)}
        - The <AtomicObject> string matches the regular expression {\w+\([+|-]?\d+,[+|-]?\d+\)}
- Nested within <ObjectSet> is 0 or more <Object> elements without attributes.
    - Nested within <Object> is a single <PointVector> element without attributes.
        - The <PointVector> element will contain a string defining all points within the object.
            - The coordinate pair consists of an *x* and *y* coordinate, each an integer value (pixels) of the coordinate location on the grid, delimited by a comma (Unicode U+002C, ASCII 44 decimal, , ,).
            - The pair is enclosed in parenthesis () (Unicode U+0028, ASCII 40 decimal, (and Unicode U+0029, ASCII 41 decimal,)).
            - The entire string is enclosed in curly braces {} (Unicode U+007B, ASCII 123 decimal, } and Unicode U+007D, ASCII 123 decimal, }).
            - Example: {(343,213)}
            - The <PointVector> string matches the regular expression {(\(\d+,\d+\))*}

- Nested within <Object> is a single <EdgeVector> element without attributes.
    - The <EdgeVector> element will contain a string defining the end points of the edges of the object.
        - The end point coordinate pair consists of an *x* and *y* coordinate, each an integer value (pixels) of the coordinate location on the grid, delimited by a comma (Unicode U+002C, ASCII 44 decimal, , ,).
        - The pair is enclosed in parenthesis () (Unicode U+0028, ASCII 40 decimal, (and Unicode U+0029, ASCII 41 decimal,)).
        - The edge vector may be empty.
        - The entire string is enclosed in curly braces {} (Unicode U+007B, ASCII 123 decimal, } and Unicode U+007D, ASCII 123 decimal, }).
        - The <EdgeVector> string matches the regular expression `{(\(\d+,\d+\))*}`
- Nested within <Object> is a single <LabelList> element without attributes.
    - The <LabelList> element will contain a string defining a list of labels. The <LabelList> element is not currently used.
        - The list is empty.
        - The entire string is enclosed in curly braces {} (Unicode U+007B, ASCII 123 decimal, } and Unicode U+007D, ASCII 123 decimal, }).
        - The <LabelList> string matches the regular expression `{}`
- Nested within <Object> is a single <ValueList> element without attributes.
    - The <ValueList> element will contain a string defining a list of values for labels. The <ValueList> element is not currently used.
        - The list is empty.
        - The entire string is enclosed in curly braces {} (Unicode U+007B, ASCII 123 decimal, } and Unicode U+007D, ASCII 123 decimal, }).
        - The {ValueList> string matches the regular expression `{}`

- Nested within <ObjectSet> is 0 or more <TerminatedEdgeObject> elements without attributes.
    - The <TerminatedEdgeObject> element will contain a string defining an arrow.
        - The string contains a pair of coordinate points, each defining one of the end points of the edge followed by a term defining the type of the edge.
        - The coordinate pair consists of an *x* and *y* coordinate, each an integer value (pixels) of the coordinate location on the grid, delimited by a comma (Unicode U+002C, ASCII 44 decimal, , ,).
        - The coordinate pair is enclosed in parenthesis () (Unicode U+0028, ASCII 40 decimal, (and Unicode U+0029, ASCII 41 decimal,)).
        - The type of edge comes from a vocabulary of values.
            - Type-1 is a line with a single arrowhead at the second coordinate point.
            - Type-2 is a line with a arrowhead at each coordinate point.
        - A comma (Unicode U+002C, ASCII 44 decimal, , ,) delimits the type from the second coordinate pair.
        - Example: (234,435),(323,298),Type-2
        - The <TerminatedEdgeObject> string matches the regular expression \([+|-]?\d+,[+|-]?\d+\),\([+|-]?\d+,[+\-]?\d+\),[\s\S]+

- Nested within <QuestionPart> is a single <SnapPoint> element without attributes. The <SnapPoint> element will be the second element within <QuestionPart>.
    - The <SnapPoint> element will contain a string defining snap points.
        - The snap points replicate the snap points in the original item.
        - The string contains the value of the snap radius (pixels) followed by the "@" (Unicode U+0040, ASCII 64 decimal, @ @) followed by a list of x-y coordinate pairs.
        - The coordinate pairs are delimited by a semi colon (Unicode U+003B, ASCII 59 decimal, ; ;).
        - The coordinate pair consists of an *x* and *y* coordinate, each a non-negative integer value (pixels) of the coordinate location on the grid, delimited by a comma (Unicode U+002C, ASCII 44 decimal, , ,).
        - Example: 10@17,34;138,220
    - The <SnapPoint> string matches the regular expression \d+@(\d+,\d+)(;\d+,\d+)*
    - The <SnapPoint> element may be empty.

- Nested within <QuestionPart> is 0 or 1 <Lines> elements without attributes. The <Lines> element will be the third element within <QuestionPart>.

    - Nested within <Lines> is a sequence of 1 or more <Line> elements defining a line segment. Each <Line> element is empty and has six required attributes:
        - sourceX – the x coordinate of one end of the line, a non-negative integer value (pixels) of the coordinate location.
        - sourceY – the y coordinate of one end of the line, a non-negative integer value (pixels) of the coordinate location.
        - targetX – the x coordinate of the other end of the line, a non-negative integer value (pixels) of the coordinate location.
        - targetY – the y coordinate of the other end of the line, a non-negative integer value (pixels) of the coordinate location.
        - dir – the direction of an arrow, a vocabulary of values.
            - both is a line with arrows at both ends.
            - forward is a line with an arrows at the second end point.
            - none is a line without arrows.
        - style – the style of the line, a vocabulary of values.
            - dashed is a dashed line.
            - solid is a solid line.

Elements and attributes not described above should not appear in the XML. Common XML features, e.g., comments, white space, character encodings, may appear in the XML.

Response Data XML Example
An example of the nested XML element structure is shown below.

```
<?xml version="1.0", encoding="UTF-8"?>
<AnswerSet>
      <Question id="">
            <QuestionPart id="1">
                  <ObjectSet>
                        <RegionGroupObject id="group" numselected="1">
                              <RegionGroup name="a1"
                                    isselected="true"/>
                              <RegionGroup name="a2"
                                    isselected="false"/>
                        </RegionGroupObject>
                        <AtomicObject>{3(234,45)}</AtomicObject>
                        <AtomicObject>{4(67,156)}</AtomicObject>
                        <Object>
                              <PointVector>{(343,213)}</PointVector>
                              <EdgeVector>{}</EdgeVector>
                              <LabelList>{}</LabelList>
                              <ValueList>{}</ValueList>
                        </Object>
                        <Object>
                              <PointVector>{(93,73)}</PointVector>
                              <EdgeVector>{}</EdgeVector>
                              <LabelList>{}</LabelList>
                              <ValueList>{}</ValueList>
                        </Object>
                        <TerminatedEdgeObject>(200,202),(300,303),
                              Type-2</TerminatedEdgeObject>
                        <TerminatedEdgeObject>(234,435),(323,298),
                              Type-2</TerminatedEdgeObject>
                  </ObjectSet>
                  <SnapPoint>10@17,34;138,220</SnapPoint>
```

```
                        <Lines>
                              <Line sourceX="34" sourceY="258" targetX="88"
                              targetY="314" dir="both" style="solid" />
                              <Line sourceX="443" sourceY="154" targetX="542"
                              targetY="439" dir="both" style="solid" />

                        </Lines>
                  </QuestionPart>
            </Question>
      </AnswerSet>
```

In the example:

- For <QuestionPart id="1">
  - the student has selected 1 object for <RegionGroup name="group"> from the grid item.
    - the student has selected <RegionObject name="a1"> for <RegionGroup name="group"> from the grid item.
    - the student has not selected <RegionObject name="a2"> for <RegionGroup name="group"> from the grid item.
  - the student has placed the item's <IconSpec> element with <Label>3</Label> at point *x*=234, *y*=45 on the grid.
  - the student has placed the item's <IconSpec> element with <Label>4</Label> at point *x*=67, *y*=156 on the grid.
  - the student has placed a point vector 343, 213.
  - the student has created a point vector 93, 73.
  - the student has drawn a Type-2 terminated edge object, a double headed arrow, from *x*=200, *y*=202 to *x*=300, *y*=303.
  - the student has drawn a Type-2 terminated edge object, a double headed arrow, from *x*=234, *y*=435 to *x*=323, *y*=298.
  - the item has a 10 pixel radius snap point at *x*=17, *y*=34 and *x*=138, *y*=220.
  - the student has drawn a solid line from *x*=34, *y*=258 to *x*=88, *y*=314.
  - the student has drawn a solid line from *x*=443, *y*=154 to *x*=542, *y*=439.

Assessment Item – Response Data Alignment

An example Grid item is shown below. It illustrates how the identifiers in the response data align with the identifiers in the item. Highlighted identifiers, names and values in the item correspond to potential information in the response data.

```
...
<QuestionPart id="1">
      ...
      <ObjectMenuIcons>
            <IconSpec>
                  <FileSpec>file_a.png</FileSpec>
                  <Label>1</Label>
            </IconSpec>
            <IconSpec>
                  <FileSpec>file_b.png</FileSpec>
                  <Label>2</Label>
            </IconSpec>
            <IconSpec>
                  <FileSpec>file_c.png</FileSpec>
                  <Label>3</Label>
            </IconSpec>
            <IconSpec>
                  <FileSpec>file_d.png</FileSpec>
                  <Label>4</Label>
            </IconSpec>
            <IconSpec>
                  <FileSpec>file_e.png</FileSpec>
                  <Label>5</Label>
            </IconSpec>
```

```
            </ObjectMenuIcons>
            ...
            <Regions>
                  <Region name="a1" ...>...</Region>
                  <Region name="a2" ...>...</Region>
                  <Region name="a3" ...>...</Region>
                  <Region name="a4" ...>...</Region>
            </Regions>
            <RegionGroups>
                  <RegionGroup name="group" min="0" max="1">
                        <Include region="a1"/>
                        <Include region="a2"/>
                  </RegionGroup>
                  <RegionGroup name="other" min="0" max="1">
                        <Include region="a3"/>
                        <Include region="a4"/>
                  </RegionGroup>
            </RegionGroups>
            ...
      </QuestionPart>
      ...
```

Example Item Responses

Fictitious examples of different item responses for GI items are shown below.

```
123-67826~0~<?xml version="1.0", encoding="UTF-8"?><AnswerSet><Question id="">
<QuestionPart id="1"><ObjectSet><RegionGroupObject id="group" numselected="1">
<RegionGroup name="a1" isselected="true"/><RegionGroup name="a2"
isselected="false"/></RegionGroupObject></ObjectSet><SnapPoint></SnapPoint>
</QuestionPart></Question></AnswerSet>

123-67827~0~<?xml version="1.0", encoding="UTF-8"?><AnswerSet><Question id="">
<QuestionPart id="1"><ObjectSet><AtomicObject>{3(234,45)}</AtomicObject>
<AtomicObject>{4(67,156)}</AtomicObject></ObjectSet>
<SnapPoint>10@17,34;138,220</SnapPoint></QuestionPart></Question></AnswerSet>

123-67828~0~<?xml version="1.0", encoding="UTF-8"?><AnswerSet><Question id="">
<QuestionPart id="1"><ObjectSet><Object><PointVector>{(343,213)}</PointVector>
<EdgeVector>{}</EdgeVector><LabelList>{}</LabelList><ValueList>{}</ValueList>
</Object><Object><PointVector>{(93,73)}</PointVector><EdgeVector>{}</EdgeVector>
<LabelList>{}</LabelList><ValueList>{}</ValueList></Object></ObjectSet><SnapPoint>
</SnapPoint></QuestionPart></Question></AnswerSet>

123-67829~0~<?xml version="1.0", encoding="UTF-8"?><AnswerSet><Question id="">
<QuestionPart id="1"><ObjectSet><Object><PointVector>{(343,213)}</PointVector>
<EdgeVector>{}</EdgeVector><LabelList>{}</LabelList><ValueList>{}</ValueList>
</Object><TerminatedEdgeObject>(200,202),(300,303),Type-2</TerminatedEdgeObject>
<TerminatedEdgeObject>(234,435),(323,298),Type-2</TerminatedEdgeObject>
</ObjectSet><SnapPoint></SnapPoint><Lines><Line sourceX="34" sourceY="258"
targetX="88" targetY="314" dir="both" style="solid" /><Line sourceX="443"
sourceY="154" targetX="542" targetY="439" dir="both" style="solid" /></Lines>
</QuestionPart></Question></AnswerSet>

123-67830~0~<?xml version="1.0", encoding="UTF-8"?><AnswerSet><Question id="">
<QuestionPart id="1"><ObjectSet>
...
</ObjectSet><SnapPoint>...</SnapPoint><Lines>...</Lines>
</QuestionPart></Question></AnswerSet>

123-67831~0~<?xml version="1.0", encoding="UTF-8"?><AnswerSet><Question id="">
<QuestionPart id="1"><ObjectSet>
...
</ObjectSet><SnapPoint>...</SnapPoint><Lines>...</Lines>
</QuestionPart></Question></AnswerSet>
```

Hot Text Item (HTQ) Response Format

The response format for a HTQ item consists of:

- The item identifier
- The item score:
  - 0: item score
  - 1: item score
  - 2: item score
  - ...
  - n: item score (the maximum value for the score is item specific)
- Response data wrapped in XML.

Response Data XML
The root of the XML response data is a single `<itemResponse>` element without attributes.

- Nested within `<itemResponse>` is a sequence of 1 or more `<response>` elements. Each `<response>` element has a required `id` attribute; the attribute value is a string that identifies the particular response. The value of the id is unique for each `<response>` for the item. The value of the `id` will match the value of the `data-its-group` attribute of one of the item's `<span>` elements that has a class attribute value of `interaction selectable` in the item's `<stem>` .

  - Nested within the `<response>` element is a sequence of 0 or more `<value>` elements (the `<response>` element may be empty, but if empty it will have both an opening and closing tag). The `<value>` elements will not have any attributes.

    Each `<value>` element will contain a string that is the value of the particular response; the string value will be the `data-its-identifier` attribute of the selected `<span>` element.

The response XML does not begin with a standard XML declaration. Assume that the XML conforms to XML 1.0 and is encoded in UTF-8, i.e., if present, the XML declaration would be:

```
<?xml version="1.0", encoding="UTF-8"?>
```

Elements and attributes not described above should not appear in the XML. Common XML features, e.g., comments, white space, character encodings, may appear in the XML.

Response Data XML Example
An example of the nested XML element structure is shown below.

```
<itemResponse>
      <response id="1">
            <value>4</value>
            <value>6</value>
      </response>
      <response id="2">
            <value>3</value>
      </response>
</itemResponse>
```

In the example:

- The student has selected `<span class="interaction selectable" data-its-identifier="4" data-its-group="1">...</span>`
- The student has selected `<span class="interaction selectable" data-its-identifier="6" data-its-group="1">...</span>`
- The student has selected `<span class="interaction selectable" data-its-identifier="3" data-its-group="2">...</span>`

Assessment Item – Response Data Alignment

An example QTI hottext interaction within an item is shown below. It illustrates how the identifiers in the response data align with the identifiers in the item. Highlighted identifiers in the item correspond to identifiers in the response data.

```
<stem>
	<p>Part 1</p>
		<span class="interaction selectable" data-its-identifier="1"
		data-its-group="1">Hot Text A</span>
		<span class="interaction selectable" data-its-identifier="2"
		data-its-group="1">Hot Text B</span>
		<span class="interaction selectable" data-its-identifier="3"
		data-its-group="1">Hot Text C</span>
		<span class="interaction selectable" data-its-identifier="4"
		data-its-group="1">Hot Text D</span>
		<span class="interaction selectable" data-its-identifier="5"
		data-its-group="1">Hot Text E</span>
		<span class="interaction selectable" data-its-identifier="6"
		data-its-group="1">Hot Text F</span>
	<p>Part 2</p>
		<span class="interaction selectable" data-its-identifier="1"
		data-its-group="2">Hot Text A</span>
		<span class="interaction selectable" data-its-identifier="2"
		data-its-group="2">Hot Text B</span>
		<span class="interaction selectable" data-its-identifier="3"
		data-its-group="2">Hot Text C</span>
		<span class="interaction selectable" data-its-identifier="4"
		data-its-group="2">Hot Text D</span>
</stem>
...
```

Example Item Responses

Fictitious examples of different item responses for HTQ items are shown below.

```
123-78905~0~<itemResponse><response id="1"><value>3</value></response>
<response id="2"><value>5</value></response></itemResponse>

123-78906~0~<itemResponse><response id="1"><value>3</value><value>2</value>
<value>5</value></response><response id="2"></response></itemResponse>

123-78907~0~<itemResponse><response id="1"><value>3</value><value>2</value>
<value>5</value></response></itemResponse>

123-78908~0~<itemResponse><response id="1"></response></itemResponse>
```

Match Item (MI) Response Format

The response format for a MI item consists of:

- The item identifier
- The item score:
  - 0: item score
  - 1: item score
  - 2: item score
  - ...
  - n: item score (the maximum value for the score is item specific)
- Response data wrapped in XML.

Response Data XML
The root of the XML response data is a single `<itemResponse>` element without attributes.

- Nested within `<itemResponse>` is a sequence of 1 or more `<response>` elements. Each `<response>` element has a required `id` attribute; the attribute value is a string that identifies the particular response. The value of the id is unique for each `<response>` for the item. The value of the `id` will match the value of the `responseIdentifier` attribute of one of the item's `<matchInteraction>` elements.
  - Nested within the `<response>` element is a sequence of 0 or more `<value>` elements (the `<response>` element may be empty, but if empty it will have both an opening and closing tag). The `<value>` elements will not have any attributes.
  - Each `<value>` element will contain a string that is the value of the particular response. The string is a pair of values indicating the match. The first value in the string will be the `identifier` attribute of the selected `<simpleAssociableChoice>` element from the first `<simpleMatchSet>` element of the item. The second value in the string will be the `identifier` attribute of the selected `<simpleAssociableChoice>` element from the second `<simpleMatchSet>` element of the item.

The response XML does not begin with a standard XML declaration. Assume that the XML conforms to XML 1.0 and is encoded in UTF-8, i.e., if present, the XML declaration would be:
`<?xml version="1.0", encoding="UTF-8"?>`

Elements and attributes not described above should not appear in the XML. Common XML features, e.g., comments, white space, character encodings, may appear in the XML.

Response Data XML Example
An example of the nested XML element structure is shown below.

```
<itemResponse>
      <response id="RESPONSE">
            <value>1 a</value>
            <value>2 c</value>
            <value>2 a</value>
            <value>3 d</value>
      </response>
</itemResponse>
```

In the example:

- For the interaction `<matchInteraction responseIdentifier="RESPONSE">`
  - the student has selected that `<simpleAssociableChoice identifier="1">` matches `<simpleAssociableChoice identifier="a">`

- the student has selected that <simpleAssociableChoice identifier="2"> matches <simpleAssociableChoice identifier="c">
- the student has selected that <simpleAssociableChoice identifier="2"> matches <simpleAssociableChoice identifier="a">
- the student has selected that <simpleAssociableChoice identifier="3"> matches <simpleAssociableChoice identifier="d">

Assessment Item – Response Data Alignment

An example QTI match interaction within an item is shown below. It illustrates how the identifiers in the response data align with the identifiers in the item. Highlighted identifiers in the item correspond to identifiers in the response data.

```
...
<qti spec="itemBody">
       <matchInteraction responseIdentifier="RESPONSE" minAssociations="1"
       maxAssociations="0">
              <prompt />
              <simpleMatchSet>
                     <simpleAssociableChoice identifier="1"
                     matchMax="0"><p>1</p></simpleAssociableChoice>
                     <simpleAssociableChoice identifier="2"
                     matchMax="0"><p>2</p></simpleAssociableChoice>
                     <simpleAssociableChoice identifier="3"
                     matchMax="0"><p>3</p></simpleAssociableChoice>
                     <simpleAssociableChoice identifier="4"
                     matchMax="0"><p>4</p></simpleAssociableChoice>
              </simpleMatchSet>
              <simpleMatchSet>
                     <simpleAssociableChoice identifier="a"
                     matchMax="0"><p>True</p></simpleAssociableChoice>
                     <simpleAssociableChoice identifier="b"
                     matchMax="0"><p>False</p></simpleAssociableChoice>
                     <simpleAssociableChoice identifier="c"
                     matchMax="0"><p>Null</p></simpleAssociableChoice>
                     <simpleAssociableChoice identifier="d"
                     matchMax="0"><p>Empty</p></simpleAssociableChoice>
              </simpleMatchSet>
       </matchInteraction>
</qti>
...
```

Example Item Responses

Fictitious examples of different item responses for MI items are shown below.

```
123-89109~0~<itemResponse><response id="RESPONSE"><value>1 a</value></response>
<response id="2"><value>5</value></response></itemResponse>

123-89110~0~<itemResponse><response id="RESPONSE"><value>1 b</value>
<value>2 c</value><value>2 d</value></response></itemResponse>

123-89111~0~<itemResponse><response id="RESPONSE"></response></itemResponse>
```

Short Answer Item (SA) Response Format

The response format for a SA item consists of:

- The item identifier
- The item score:
  - -1: the item was not scored
- Response data in HTML.

Response Data HTML
The response data is the student's written response encoded in HTML. The HTML may contain most of the HTML *grouping* elements (e.g., paragraphs, lists) and *text-level semantics* (e.g., bold, underline). It may include special characters and embedded styles. The HTML may span multiple lines, i.e., includes line breaks and other white space around the HTML elements. The HTML should not include any *document metadata* elements, *section* elements (e.g., body), *embedded content* elements (e.g., iframes), *link* elements, *media* elements, *tabular data* elements, *form* elements or *script* elements. An HTML element may include a language attribute (`lang`) to designate the natural language of the response.

The response HTML does not begin with a standard HTML declaration. Assume that the HTML conforms to HTML5 and is encoded in UTF-8, i.e., if present, the HTML declaration would be:

```
<!DOCTYPE html><html><meta charset="utf-8">
```

Assessment Item – Response Data Alignment
There are no elements or attributes in the item that are aligned with elements and attributes in the response.

Example Item Responses
Fictitious examples of different item responses for SA items are shown below.

```
123-91212~-1~<p>Paragraph</p><p>Paragraph</p>

123-91213~-1~<ul><li>List item</li><li>List item</li><li>List item</li></ul>

123-91214~-1~<p> </p>

123-91215~-1~<p style="margin-left: 10px"><em>Text</em><br />
<u>Text</u></p>
```

Table Item (TI) Response Format

The response format for a TI item consists of:

- The item identifier
- The item score:
  - 0: item score
  - 1: item score
  - 2: item score
  - ...
  - n: item score (the maximum value for the score is item specific)
- Response data describing a table wrapped in XML. Elements are a subset of the HTML table elements.

Response Data XML
The root of the XML response data is a single <responseSpec> element without attributes.

- Nested within the <responseSpec> element is a single <responseTable> element without attributes.
  - Nested within the <responseTable> element is a set of 2 or more table rows, each in a <tr> element.
    - The first table row has 1 or more table column headers, each in a <th> element. Each <th> element is empty and has a required id attribute; the attribute value is a string that identifies the particular column of the table. The value of the id is unique for each <th> for the item.

      The value of the id will match the value of the data-its-identifier attribute of one of the item's <th> elements within the item's <thead> element in the item's <table> element (in the <stem>). The number of <th> elements in the response will be the same as the number of <th> elements in the item.
    - The second and all subsequent table rows in the response will have 1 or more table data entries, each in a <td> element. The number of data entries will be the same as the number of columns specified in the table header row. The content of the <td> element will be a string containing the student response. A data value may be empty (e.g., <td/>). The number of data rows will be the same as the number of <tr> row elements specified in the in <tbody> element in the item's <table> element.

The response XML does not begin with a standard XML declaration. Assume that the XML conforms to XML 1.0 and is encoded in UTF-8, i.e., if present, the XML declaration would be:

```
<?xml version="1.0", encoding="UTF-8"?>
```

Elements and attributes not described above should not appear in the XML. Common XML features, e.g., comments, white space, character encodings, may appear in the XML.

Response Data XML Example
An example of the nested XML element structure is shown below.

```
<responseSpec>
      <responseTable>
            <tr>
                  <th id="col0"/>
                  <th id="col1"/>
```

```
                </tr>
                <tr>
                        <td>3</td>
                        <td>4</td>
                </tr>
                <tr>
                        <td>abc</td>
                        <td>7.8</td>
                </tr>

                <tr>
                        <td/>
                        <td>0</td>
                </tr>
        </responseTable>
</responseSpec>
```

The example represents the table shown, specified in the item as containing 2 columns (*col0* and *col1*), and 3 rows of response values.

col0	col1
3	4
abc	7.8
	0

Assessment Item – Response Data Alignment
An example table within an item is shown below. It illustrates how the identifiers in the response data align with the identifiers in the item. Highlighted identifiers in the item correspond to identifiers in the response data.

```
...
<stem>
        <p>Enter data in the table</p>
        <table>
                <thead>
                        <tr>
                                <th data-its-identifier="col0"><p>Column 0</p></th>
                                <th data-its-identifier="col1"><p>Column 1</p></th>
                        </tr>
                </thead>
                <tbody>
                        <tr>
                                <td/><td/>
                        </tr>
                        <tr>
                                <td/><td/>
                        </tr>
                        <tr>
                                <td/><td/>
                        </tr>
                </tbody>
        </table>
</stem>
...
```

Example Item Responses
Fictitious examples of different item responses for TI items are shown below.

```
123-12320~0~<responseSpec><responseTable><tr><th id="col0"/>
<thid="col1"/></tr><tr><td>3</td><td>4</td></tr><tr><td>abc</td><td>7.8</td></tr>
<tr><td/><td>0</td></tr></responseTable></responseSpec>
```

```
123-12321~0~<responseSpec><responseTable><tr><th id="col0"/>
<thid="col1"/></tr><tr><td/><td>17</td></tr><tr><td/><td>22.5</td></tr>
</responseTable></responseSpec>

123-12322~0~<responseSpec><responseTable><tr><th id="col0"/>
<thid="col1"/></tr><tr><td/><td/></tr><tr><td/><td></tr><tr><td/><td></tr>
</responseTable></responseSpec>
```

Writing Extended Response Item (WER) Response Format

The response format for a WER item consists of:

- The item identifier
- The item score:
  - -1: the item was not scored
- Response data in HTML.

Response Data HTML
The response data is the student's written response encoded in HTML. The HTML may contain most of the HTML *grouping* elements (e.g., paragraphs, lists) and *text-level semantics* (e.g., bold, underline). It may include special characters and embedded styles. The HTML may span multiple lines, i.e., includes line breaks and other white space around the HTML elements. The HTML should not include any *document metadata* elements, *section* elements (e.g., body), *embedded content* elements (e.g., iframes), *link* elements, *media* elements, *tabular data* elements, *form* elements or *script* elements. An HTML element may include a language attribute (`lang`) to designate the natural language of the response.

The response HTML does not begin with a standard HTML declaration. Assume that the HTML conforms to HTML5 and is encoded in UTF-8, i.e., if present, the HTML declaration would be:

```
<!DOCTYPE html><html><meta charset="utf-8">
```

Assessment Item – Response Data Alignment
There are no elements or attributes in the item that are aligned with elements and attributes in the response.

Example Item Responses
Fictitious examples of different item responses for WER items are shown below.

```
123-23416~-1~<p>Paragraph</p><p>Paragraph</p>
```

```
123-23417~-1~<ol><li>List item</li><li>List item</li><li>List item</li></ol>
```

```
123-23418~-1~<p> </p>
```

```
123-23419~-1~<p style="margin-left: 10px"><u>Text</u>
<br /><strong>Text</strong></p>
```

Change Log

Date	Version	Author	Notes
20141222	0.50	DR	Internal Draft for Comment Resolution
20150309	0.80	DR	Public Draft Release
20150422	1.00	DR	Public Release