

# The Get Structured Text Action Group

## The Get Structured Text Action Group

Provides a basic set of functions for simple access to bits of the XML structured text in its native markup language. There is an assumption that the markup includes at least rudimentary structural definition. These functions provide users or applications with the basic building blocks to begin using the text in its native markup if desirable. The available functions mirror those of the View Structured Text Action Group and provide a means to: obtain a well-formed XML summary of the logical units of content contained within a text (a table of contents), obtain a well-formed XML fragment of any descriptive or explanatory information contained in the document header of the text, and obtain a well-formed XML fragment of any individual logical unit of content contained within a text.

Response Type: text/xml

### getTOC

Returns a logical table of contents for a text as well-formed XML (based on a generic <div> structure). The table of contents contains actionable URLs for each unit of content of a text. These URLs could be fully formed calls to **getChunk**.

The structure of the well-formed XML is as follows:

```
<div type="toc" id="optional local text identifier">
<head>The label or title of the text</head>
  <div xlink="actionable URL to a chunk">
    <head>The label or title of the chunk</head>
  <div>
</div>
```

[AssetActionsDevelopment:View a complete stand-alone example of a getTOC response.](#)

Some notes about the XML structure:

1. <div> tags are unnumbered and can be nested as appropriate
2. The value of the 'type' attribute for a **getTOC** response is always "toc"
3. The 'id' attribute is optional
4. The <head> element nested in the <response> element contains the title/label of the object
5. The <head> element nested in <div> elements contains the title/label of the chunk

### Proposal to change well-formed XML structure wrapper

```
<response type="toc" id="optional local text identifier">
  <head>The label or title of the object</head>
  <div xlink="actionable URL to a chunk">
    <head>The label or title of the chunk</head>
  <div>
</response>
```

- \*The wrapper becomes a <response> element that contains the document header markup
- \*The @n attribute has been removed

### getHeader

Returns the well-formed native markup language header (for example, TEI, EAD, DOCBOOK) for an entire text as a whole. It is assumed that this header will provide information about the text as a whole.

The well-formed native markup language header is contained in the following wrapper:

```
<div type="header" n="format of the native XML" id="optional local text identifier">
  ... native markup ...
</div>
```

[AssetActionsDevelopment:View a complete stand-alone example of a getHeader response.](#)

Some notes about the wrapper:

1. the header is wrapped in the same top level <div> + attributes structure as the TOC example; this <div> is followed by the actual document header markup.
2. the value of the 'type' attribute is always "header"

### Proposal to change well-formed XML structure wrapper

```
<response type="header" id="optional local text identifier">
<head>The label or title of the object</head>
... native markup ...
</reponse>
```

- \*The wrapper becomes a <response> element that contains the document header markup
- \*The @n attribute has been removed
- \*A <head> element has been added; it contains the label/title of the object

### getChunk

Returns a logical chunk of a text, such as a chapter, as native XML in a generic <div> wrapper. This assumes that a request parameter is passed (the *chunkID*).

Format:

The well-formed XML in native markup language is contained the following wrapper:

```
<div type="chunk" n="optional title or label of the chunk" id="optional parent text identifier">
<head>The label or title of the chunk</head>
    ... native markup ...
<div>
</div>
```

[AssetActionsDevelopment:View a complete stand-alone example of a getChunk response.](#)

Some notes about the wrapper:

1. the chunk is wrapped in the same top level <div> + attributes structure as the TOC example; this <div> is followed by the actual chunk markup.
2. the value of the 'type' attribute is always "chunk"

### Proposal to change well-formed XML structure wrapper

```
<response type="chunk" id="optional local text identifier">
  <head>The label or title of the chunk</head>
  ... native markup ...
</response>
```

- \*The wrapper becomes a <response> element that contains the document header markup
- \*The @n attribute has been removed