

Template 1.0

The API Specification for Template

Editor's Draft (September 07, 2009)

Editor: Markku Laine, TKK <mplaine@cc.hut.fi>

Abstract

The Template markup language is an intermediate language for Service Management UI (XFormsDB IDE). It is used for describing reusable, parameterized parts of a UI (e.g., components), which are written in XHTML+XFormsDB, and utilizing them on a Web page.

Status of this Document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. The latest status of this document series is maintained at the TKK.

Please send comments about this document to mplaine@cc.hut.fi.

Table of Contents

1	About the Template Specification.....	3
1.1	Document Conventions.....	3
1.2	Namespace for Template	3
2	The <code>template:webpage</code> Element	3
2.1	Usage Example.....	3
3	The <code>template:component</code> Element.....	4
3.1	Usage Example.....	4
4	The <code>template:container</code> Element.....	4
4.1	Usage Example.....	4
5	The <code>template:head</code> Element	4
5.1	Usage Example.....	4
6	The <code>template:body</code> Element	5
6.1	Usage Example.....	5
7	The <code>template:meta</code> Element	5
7.1	Usage Example.....	5
8	The <code>template:call-component</code> Element.....	5

8.1	Usage Example.....	5
9	The <code>template:param</code> Element	6
9.1	Usage Example.....	6
10	The <code>template:option</code> Element	6
10.1	Usage Example.....	6
11	The <code>template:with-param</code> Element.....	7
11.1	Usage Example.....	7
12	The <code>template:theme</code> Attribute.....	7
12.1	Usage Example.....	7

1 About the Template Specification

The Template markup language needs to be used with the XHTML+XFormsDB markup languages.

Please find more information about the aforementioned languages from the following Websites:

- <http://www.w3.org/TR/xhtml1/>
- <http://www.w3.org/TR/xforms11/>
- <http://www.tml.tkk.fi/Opinnot/T-111.5360/api/>

1.1 Document Conventions

Throughout this document, the following namespace prefixes and corresponding namespace identifiers are used:

- **template:** The Template namespace <http://www.tml.tkk.fi/2009/template>
- **xforms:** The XForms namespace <http://www.w3.org/2002/xforms>
- **xhtml:** The XHTML namespace <http://www.w3.org/1999/xhtml>

1.2 Namespace for Template

The namespace URI for Template is <http://www.tml.tkk.fi/2009/template>.

2 The `template:webpage` Element

The `template:webpage` element acts as a root element for an XHTML+XFormsDB based Web page i.e. replaces the `xhtml:html` root element of a Web page. In addition, the element contains two child elements: the `template:head` element and the `template:body` element, which replace the `xhtml:head` element and the `xhtml:body` element, respectively.

2.1 Usage Example

```
<template:webpage xml:lang="en" lang="en"
  xmlns:ev="http://www.w3.org/2001/xml-events"
  xmlns:exforms="http://www.exforms.org/exf/1-0"
  xmlns:template="http://www.tml.tkk.fi/2009/template"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xforms="http://www.w3.org/2002/xforms"
  xmlns:xformsdb="http://www.tml.tkk.fi/2007/xformsdb"
  xmlns:xhtml="http://www.w3.org/1999/xhtml"
  xmlns:xxforms="http://orbeon.org/oxf/xml/xforms">
  <template:head>
    ...
  </template:head>
  <template:body>
    ...
  </template:body>
</template:webpage>
```

3 The `template:component` Element

The `template:component` element acts as a root element for an XHTML+XFormsDB based Component. The element contains two child elements: the `template:head` element and the `template:body` element.

3.1 Usage Example

```
<template:component xml:lang="en" lang="en"
  xmlns:ev="http://www.w3.org/2001/xml-events"
  xmlns:exforms="http://www.exforms.org/exf/1-0"
  xmlns:template="http://www.tml.tkk.fi/2009/template"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xforms="http://www.w3.org/2002/xforms"
  xmlns:xformsdb="http://www.tml.tkk.fi/2007/xformsdb"
  xmlns:xhtml="http://www.w3.org/1999/xhtml"
  xmlns:xxforms="http://orbeon.org/oxf/xml/xforms">
  <template:head>
    ...
  </template:head>
  <template:body>
    ...
  </template:body>
</template:component>
```

4 The `template:container` Element

Optional child element of the `template:body` element (does not need to be a direct parent element, though) that acts as a placeholder for Components to be called/included to a Web page. The element contains two child elements: the `template:head` element and the `template:body` element.

4.1 Usage Example

```
<template:container>
  <template:head>
    ...
  </template:head>
  <template:body>
    ...
  </template:body>
</template:container>
```

5 The `template:head` Element

Required child element of the `template:webpage` element, the `template:component` element, and the `template:container` element that contains meta information about its parent as well as all other information related to the parent (e.g., CSS, JavaScript, and XForms models) except UI controls.

5.1 Usage Example

```
<template:head>
  ...
</template:head>
```

6 The `template:body` Element

Required child element of the `template:webpage` element, the `template:component` element, and the `template:container` element that contains all the contents of the parent (e.g., UI controls).

However, the `template:body` element of the `template:container` element can only contain the `template:call-component` elements.

6.1 Usage Example

```
<template:body>
  ...
</template:body>
```

7 The `template:meta` Element

Required child element of the `template:head` element that provides metadata about the `template:webpage` element, the `template:component` element, or the `template:container` element. Default value is the value of this element.

The metadata can be used by XFormsDB IDE (XIDE) or other web services, which have access rights to the files.

Attributes of the `template:meta` element:

name

Required attribute specifying the name of a metadata, whose legal values are: `title`, `description`, `id`, and `tags` (this value is for the `template:component` and `template:webpage` elements only). For the `template:container` and `template:webpage` elements, the `id` attribute must be unique within the Web application. For the `template:component` element, the `id` attribute must be unique within all components in XIDE.

7.1 Usage Example

```
<template:meta name="description">
  My component description.
</template:metadadata>
```

8 The `template:call-component` Element

Optional child element of the `template:body` element of the `template:container` element for calling a named Component.

Attributes of the `template:call-component` element:

name

Required attribute specifying the name of a Component to be called.

8.1 Usage Example

```
<template:call-component name="myComponent">
  ...
</template:call-component>
```

9 The `template:param` Element

Optional child element of the `template:head` element that declares the parameter to be used within a Web page or Component. The parameter can be typed and a default value can be assigned to it. Default value is the value of this element. The parameter can be further used only within attributes (cf. Usage Example).

Attributes of the `template:param` element:

name

Required attribute specifying the name of a parameter.

description

Optional attribute specifying the description of a parameter. Default value is an empty string.

type

Optional attribute specifying the type of a parameter, whose legal values are: `text` (default) and `binding`.

9.1 Usage Example

```
<template:param name="myBindingParameter" description="My parameter
description." type="binding">
  instance( &apos;my-default-instance&apos; )/my-default-element
</template:param>
...
<xforms:output ref="$myBindingParameter" />
```

10 The `template:option` Element

Optional child element of the `template:param` element that offers one or more pre-defined options for the parameter value. Default value is the value of the first `template:option` element, or in case the `selected="true"` attribute exists in one of the `template:option` elements, then the value of that element.

Attributes of the `template:option` element:

description

Optional attribute specifying the description of an option. Default value is an empty string.

selected

Optional attribute specifying that this option is pre-selected.

10.1 Usage Example

```
<template:param name="myTypeParameter" description="My type parameter
description." type="text">
  <template:option description="Date">
    xs:date
  </template:option>
```

```

<template:option description="Integer">
  xs:integer
</template:option>
<template:option description="String" selected="true">
  xs:string
</template:option>
</template:param>

```

11 The `template:with-param` Element

Optional child element of the `template:call-component` element, whose value is linked to the parameter declared within a Component. The parameter declared within a Component must have the same name as the one specified in the `name` attribute of this element. Default value is the value of this element.

Attributes of the `template:with-param` element:

name

Required attribute specifying the name of the parameter , which has been declared within a Component to be called.

11.1 Usage Example

```

<template:with-param name="myBindingParameter">
  instance( &apos;my-instance&apos; )/my-element[ 1 ]
</template:with-param>

```

12 The `template:theme` Attribute

Optional attribute of the `xhtml:link` element that specifies whether this is a theme stylesheet. The difference between Web application stylesheets and theme stylesheets (e.g., `reset_theme.css`) is that theme stylesheets must be placed before any other stylesheets on the Web page.

12.1 Usage Example

```

<xhtml:link rel="stylesheet" type="text/css"
  href="css/reset_theme.css" media="all" template:theme="true" />

```