

Print Template Markup

04/30/2025 4:14 pm CDT

Overview

serviceminder.io employs XML for mocking up automated and ad hoc [Print Templates](#). The set of available tags is limited, yet allow for a large amount of flexibility with the right approach.

Introduction to XML

XML models content after a series of descriptive tags, further described by optional attributes. Every tag must be opened and closed, either with a separate open and close tag or a self-closing tag. They must also each begin with a capital letter. Consider the following as a general pattern for each element:

```
<Tag attribute-1="value" attribute-2="value">Content</Tag>
```

or

```
<Tag attribute-1="value" attribute-2="value" />
```

Each print template may begin with the following in order to establish the document's encoding:

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

Following this, the body of the template can be opened with a

```
<Page>
```

tag. Within the

```
<Page>
```

tag, content can be laid out with tags, including the

```
<Newpage />
```

tag.

XML-style comments do not appear in the rendered PDF and allow you to place helpful notes by your code.

```
<Cell height="40">{{ model.description | xml_encode }}</Cell> <!-- Displays the XML-encoded description -->
```

Tags and Attributes

Page

The Page element encompasses the entire body of the document.

Supported attributes:

Attribute	Description
width	Defines the total width of the page (default: 595)
height	Defines the total height of the page (default: 842)
margin-top/bottom/left/right	Offsets the page from given side, e.g. <div>margin-top="5"</div> will push contents 5 pts from the top edge (default top/bottom: ~35)
padding-top/bottom/left/right	Pads the content on given side, e.g. <div>padding-top="5"</div> will pad top edge of contents 5 pts
line-height	Defines the height of each line of text (only has effect if line height is greater than font height)

Newpage

The Newpage element breaks the current page of the document and jumps to the next. If no content follows the Newpage element, a blank page will be placed at the end of the document. This element takes no attributes and is self-closing.

Font

The Font element defines a new font that can be used by other elements by matching the **font-name** attribute to the **name** attribute on the font.

CSS font properties can be used as individual XML attributes. Instead of

```
style="font-family: arial;"
```

use:

```
font-family="arial"
```

You may find a complete list of CSS font properties here: [Font Properties](#)

Supported attributes:

Attribute	Description
name	Sets the name of the element to be referenced by other elements using their font-name attribute.
default	Changes default font when set to "true"

String/Phrase

The String and Phrase elements contain text and do not break the line when closed.

Supported attributes:

Attribute	Description
font-name	Sets the font to be used inside of the element. The font must be defined beforehand.

Paragraph

The Paragraph element contains text and breaks the line when closed.

Supported attributes:

Attribute	Description
align	Sets the alignment of the element. <i>left/center/right</i>
font-name	Sets the font to be used inside of the element. The font must be defined beforehand.

Image

The Image element allows for the insertion of both foreground and background images. This tag is self-closing, e.g.

```
<Image />
```

Supported attributes:

Attribute	Description
width	Defines the width of the image (default: source width)
height	Defines the height of the image (default: source height)
scale	Modifies the size of the image by % (default: 100)
src	Sets the source path for the image, e.g. src="https://example.com/image.jpg"
layer	Pushes image into the background layer with layer="background"
background-color	Provides a fallback color for missing and transparent images. Supports comma-separated RGB (e.g. "255,255,255") and hexadecimal RGB (e.g. "#FFFFFF")

Tables

The Table element allows you to organize content into rows and columns. It is one of the most powerful elements available and one that takes some practice to master.

Within each table declaration, you should define a section for columns, at least one column in that section, a section for cells, and at least one cell.

Example table with all possible attributes:

```

<Table width="525" align="center" cell-borders="0">

  <Columns>

    <Column width="1" />

    <Column width="2" />

  </Columns>

  <Cells>

    <Cell colspan="2" font-name="small-font" line-height="12" border-top="5" border-left="0" border-right="0" border-bottom="5">This cell will span both columns, have a small font, a larger line height, and 5pt borders on the top and bottom.</Cell>

    <Cell height="40" /> <!-- This cell will leave the first column empty, with a maximum cell height of 40pts -->

    <Cell>This cell will be in the second column</Cell>

  </Cells>

</Table>

```

Supported attributes:

Attribute	Description
align	Sets the alignment of the element. <i>left/center/right</i>
width	Sets the total width of the table
cell-borders	Toggles borders for the edges of each cell based on a decimal bitmask. Each side is a bit in a 4-bit sequence as follows: <i>RIGHT LEFT BOTTOM TOP</i> -> <i>1111 (binary)</i> -> <i>8 + 4 + 2 + 1 (decimal)</i> , so no cell borders is 0, and all cell borders is 15.

Columns

The Columns element simply contains the individual column definitions of a table. It takes no attributes.

Column

The Column element defines the individual columns within a table. Each column defined contributes to the total number of cells needed to fill a row. A column's width is not absolute. It represents a fraction of the table's total width, meaning that three columns given any equivalent width will each occupy a third of the table's width. One column of width 20 and one column of width 40 means that the first column will be one third of the table's width and the second column will be two thirds of the table's width.

Attribute	Description
width	Sets the width of the column relative to the widths of the other columns.

Cells

The Cells element simply contains the individual cell definitions of a table. It takes no attributes.

Cell

The Cell element positions content within a row and column of a table. It may contain text or any single element,

but not both. It may also self-close and act as a spacer or separator if given a border.

Supported attributes:

Attribute	Description
colspan	Controls how many columns the cell spans.
align	Sets the alignment of the cell's contents. <i>left/center/right</i>
font-name	Sets the font to be used inside of the element. The font must be defined beforehand.
inherit-font-name	Sets the default font for all plain-text cells in the table that follow it.
line-height	Defines the height of each line of text (only has effect if line height is greater than font height)
border-side	Sets the width of the cell's border on the given side. <i>top/bottom/left/right</i>
height	Forces the cell to a specific height.
padding-bottom	Pushes the following element further down by specified amount.
background-color	Sets the background color of the cell. Supports comma-separated RGB (e.g. "255,255,255") and hexadecimal RGB (e.g. "#FFFFFF")