

Running the executable

the following command(s) will convert a pml file to pdf

```
./pml2pdf.sh --cli --infile /path/to/file.pml --outfile /path/to/file.pdf  
./pml2pdf.cmd --cli --infile /path/to/file.pml --outfile /path/to/file.pdf
```

Prerequisites

- Java Runtime, Version 8 or higher
- properly setup JAVA_HOME environment variable

Linux Recommendations

for Linux it is recommended to install java privately using SDKMAN [<https://sdkman.io/>]

Page Markdown Layout (PML)

Basic Document Structure

```
<?xml version="1.0" encoding="UTF-8" ?>  
<document>  
  <?...instruction... ?>  
  <page ...attributes...>  
    ...page level objects...  
  </page>  
</document>
```

Attribute Types

Boolean

usually boolean can take the following values "true|false|on|off|yes|no" and are specified like below:

```
"_boolean_"
```

Factors ie. Percentages

usually Factors are integers in the rangs of 0 ... 100, sometimes -1 means diable feature and are specified like below:

```
"_factor_"
```

File/Path

usually Files/Paths must be actully exist on the filesystem and are specified like below:

```
"_file_"  
"_path_"
```

Examples:

```
"file:///tmp/some/path/to/file.ext"  
"./some/path/to/file.ext"  
"/some/path/to/file.ext"  
"/some/path/to/file.ext"  
  
"./some/path"  
"/some/path"  
"/some/path"
```

Resources

usually Resource can be Files or references to mounts and the classpath and are specified like below:

```
"_resource_"
```

Examples:

```
"mnt:/tmp/some/path/to/file.ext"  
"mnt:tmp/some/path/to/file.ext"  
"zip:/tmp/some/path/to/file.ext"  
"zip:tmp/some/path/to/file.ext"  
"cp:tmp/some/path/to/file.ext"
```

Document Level Elements

Instructions

Options

JPEG Compression

specifies the jpeg compression factor in percent (1 = max compression, min quality; 100 = min compression, max quality)

```
<?option jpeg-compression="_factor_" ?>
```

Attribute lookup debug

```
<?option print-attribute-access="_boolean_" ?>
```

include

include the specified file for document processing.

```
<?include src="_file_" ?>
```

info

set pdf document info attributes

```
<?info  
  author="_text_"  
  creator="_text_"  
  producer="_text_"  
  subject="_text_"  
  keywords="_text_"  
  title="_text_" ?>
```

mount

adds zip files or directories to the global search path.

```
<?mount src="_file_" ?>  
<?mount dir="_path_" ?>
```

Examples:

```
<?mount src="/path/to/file.zip" ?>  
<?mount dir="/some/path" ?>
```

define

defines values for attribute lookup

```
<?define key="_value_" ?>
```

defaults

like define but loads from a properties file

```
<?defaults src="_resource_" ?>
```

Examples:

```
<?defaults src="/path/to/attr.properties" ?>  
<?defaults src="./path/to/attr.properties" ?>  
<?defaults src="../path/to/attr.properties" ?>  
<?defaults src="cp:path/to/attr.properties" ?>  
<?defaults src="mnt:path/to/attr.properties" ?>  
<?defaults src="zip:path/to/attr.properties" ?>
```

font

loads a font and defines id for referencing in attributes

```
<?font id="_id_"  
name|src="_resource_|_ref_"  
charset|encoding="_charset_"  
icon-map="_resource_" ?>
```

the following special references can be used as resources:

- "pdf:name" --- one of the base 14 pdf font names:
 - helvetica, helvetica-bold, helvetica-boldoblique, helvetica-oblique
 - courier, courier-bold, courier-oblique, courier-boldoblique
 - times-roman, times-bold, times-bolditalic, times-italic
 - symbol
 - zapf-dingbats
- "awt:name" --- a registered name of a awt font resource linked to a system or user installed font

the following character maps are defined (default: pdfdoc):

- adobe-standard, adobe-symbol, adobe-zapf-dingbats
- cp1250, cp1251, cp1252, cp1253, cp1254, cp1255, cp1256, cp1257, cp1258
- hp-roman8
- ibm437, ibm850, ibm851, ibm852, ibm855, ibm857
- iso-8859-1, iso-8859-2, iso-8859-3, iso-8859-4, iso-8859-5, iso-8859-6, iso-8859-7, iso-8859-8, iso-8859-9, iso-8859-13, iso-8859-15
- koi8-r, koi8-ru, koi8-u (cyrillic)
- macintosh (mac roman)
- microsoft-dingbats (0xF0xx)
- pdfdoc (pdf doc encoding extension to latin1)
- texnansi (TeX ansi encoding extension to latin1)
- text (groff ascii encoding extension to latin1)

the following aliases for encodings are supported:

- hp-roman, hproman
- mac-roman, macroman
- latin1 ... latin9, latin-1 ... latin-9
- iso-cyrillic, iso-arabic, iso-greek, iso-hebrew, iso-thai, iso-celtic
- ms-dingbats

the following special encoding can be used:

- unicode, allows ttf/otf fonts to use the entire 16-bit unicode glyph-space
- icons, allows ttf/otf based icon fonts for entity-lookup (see below)

the attribute "icon-map" allows loading a icon-entity specifier map

entity/icon

allows defining ids for referencing entities/icons

```
<?entity name="_name_" code="_code_" ?>  
<?icon font="_ref_" name="_name_" code="_code_" ?>
```

Examples:

```
<?entity name="non-breaking-space" code="0xA0" ?>  
<?icon font="fa" name="refresh" code="0xf123" ?>
```

class

allows specifying predefined classes of attribute sets for referencing

```
<?class id="_id_" key1="_value1_" ... keyN="_valueN_" ?>
```

Examples:

```
<?class id="a4-portrait" page-mediabox="595,842" ?>  
<?class id="us-letter" page-mediabox="612,792" ?>  
<?class id="a4-landscape" page-mediabox="842,595" ?>
```

image

```
<?image id="_id_"  
  src="_resource_"  
  compress="jpg|jpeg|index|indexed|mono|grey|dct-gray"  
  transparency="alpha|_boolean_"  
  alpha="_factor_" ?>
```

svg

```
<?svg id="_id_"  
  src="_resource_"  
  render="_factor_"  
  compress="jpg|jpeg|index|indexed|mono|grey|dct-gray"  
  transparency="alpha|_boolean_"  
  alpha="_factor_" ?>
```

header/footer

```
<?header* ... ?>  
<?footer* ... ?>
```

page element

```
<page mediabox="_bb_" rotate="_rot_">
... page level elements ...
</page>
```

Page Level Elements

image/svg

```
<svg id="_id_" ref="_id_" pos="_x,_y_" width="_pts_" height="_pts_" background="_boolean_" />

<svg id="_id_" pos="_x,_y_" width="_pts_" height="_pts_" background="_boolean_"
src="_resource_"
render="_factor_"
compress="jpg|jpeg|index|indexed|mono|grey|dct-gray"
transparency="alpha|_boolean_"
alpha="_factor_" />
```

```
<image id="_id_" ref="_id_" pos="_x,_y_" width="_pts_" height="_pts_" background="_boolean_" />

<image id="_id_" pos="_x,_y_" width="_pts_" height="_pts_" background="_boolean_"
src="_resource_"
compress="jpg|jpeg|index|indexed|mono|grey|dct-gray"
transparency="alpha|_boolean_"
alpha="_factor_" />
```

h/heading

layouts heading(s) at position and creates outlines for the pdf outline tree (see outline/chapter/section)

```
<heading id="_id_" pos="_x,_y_" level="_level_" ...attributes... >
...heading text...
</heading>

<h id="_id_" pos="_x,_y_" level="_level_" outline="_text_" ...attributes... >
...heading text...
</h>
```

text/label

layouts simple text(s)/label(s) at position

```
<label id="_id_" link="_id_" pos="_x,_y_" ...attributes... >
...simple label text...
</label>

<text id="_id_" link="_id_" pos="_x,_y_" ...attributes... ><![CDATA[
...simple text line 1...
...simple text line 2...
...simple text line 3...
]]></text>
```

p/paragraph

layouts simple paragraph(s) justified at position with width

```
<p id="_id_" pos="_x_"_y_" width="_pts_" ...attributes... >
...simple text paragraph 1...

...simple text paragraph 2...

...simple text paragraph 3...
]]&gt;&lt;/p&gt;</pre></div><div data-bbox="94 208 140 222" data-label="Section-Header"><h2>table</h2></div><div data-bbox="94 233 333 248" data-label="Text"><p>layouts a table at position with width</p></div><div data-bbox="108 272 596 337" data-label="Text"><pre>&lt;table id="_id_" pos="_x_"_y_" width="_pts_" align="..." widths="..." &gt;
&lt;cell&gt;...&lt;/cell&gt;
&lt;markdown&gt;...&lt;/markdown&gt;
&lt;cell&gt;...&lt;/cell&gt;
&lt;/table&gt;</pre></div><div data-bbox="94 363 183 378" data-label="Section-Header"><h2>markdown</h2></div><div data-bbox="94 390 727 405" data-label="Text"><p>layouts flexmark/commonmark/markdown at position with width, optionally loading from resource.</p></div><div data-bbox="108 428 630 493" data-label="Text"><pre>&lt;markdown id="_id_" pos="_x_"_y_" width="_pts_" src="_res_" ...attributes... /&gt;
&lt;markdown id="_id_" pos="_x_"_y_" width="_pts_" ...attributes... &gt;<![CDATA[
...flexmark syntax...
]]&gt;&lt;/markdown&gt;</pre></div><div data-bbox="94 520 280 536" data-label="Section-Header"><h2>outline/chapter/section</h2></div><div data-bbox="94 547 542 561" data-label="Text"><p>creates outlines for the pdf outline tree and references the actual page</p></div><div data-bbox="108 585 371 625" data-label="Text"><pre>&lt;outline text="_text_" level="_level_" /&gt;
&lt;chapter text="_text_" /&gt;
&lt;section text="_text_" /&gt;</pre></div><div data-bbox="118 649 281 676" data-label="List-Group"><ul><li>• chapter implies level 0</li><li>• section implies level 1</li></ul></div><div data-bbox="94 685 440 699" data-label="Text"><p>outline/chapter/section is a sort of "invisible" heading.</p></div><div data-bbox="94 709 175 724" data-label="Section-Header"><h2>draw-rect</h2></div><div data-bbox="94 735 425 751" data-label="Text"><p>draws a rectangle at position with width and height</p></div><div data-bbox="108 774 309 879" data-label="Text"><pre>&lt;draw-rect id="_id_"
  pos="_x_"_y_"
  width="_pts_"
  height="_pts_"
  fill-color="_color_"
  stroke-color="_color_"
  stroke="_pts_"
  background="_boolean_" /&gt;</pre></div>
```

Layers

the following page elements support the **layer** attribute for putting the object on the specified *pdf-layer*:

- text, label, p, h
- image, svg
- markdown
- draw-rect
- table

Examples

```
<text layer="_layer_name_" ... > ... content ... </text>  
<label layer="_layer_name_" ... > ... content ... </label>  
<p layer="_layer_name_" ... > ... content ... </p>  
<h layer="_layer_name_" ... > ... content ... </h>
```

```
<image layer="_layer_name_" ... />  
<svg layer="_layer_name_" ... />
```

```
<markdown layer="_layer_name_" ... > ... content ... </h>
```

```
<draw-rect layer="_layer_name_" ... attributes ... />
```

```
<table layer="_layer_name_" ... > ... cells ... </h>
```


Entities & Font-Icons

TBD.

Scripting using JEXL

TBD.