The output format of show-pdf-tags.lua

Marcel F. Krüger

July 11, 2025

1 Output format description

The output of `show-pdf-tags.lua` when invoked on a tagged PDF file is a tree structure containing all tags present in the structure hierarchy.

Every Structure Element gets printed in the form

```
<Tag> (<Tag NS>) / <Mapped> (<Mapped NS>):
├─<Meta 1>
├─<Meta 2>
├─<Meta ...>
├─<Meta n>
├─<Child 1>
├─<Child 2>
├─<Child ...>
└─<Child n>
```

Here `<Tag>` is the subtype of the structure element and `<Tag NS>` it's namespace. If the tag does not belong to any namespace than (`<Tag NS>`) is omitted.

In case that the structure element is role mapped then `<Mapped>` (`<Mapped NS>`) similarly describes the role map target. This omits any intermediate mappings. So if A gets mapped to B which in turn is mapped to C, then only A / C is printed, the intermediate step B is ignored to keep the output readable.

The entries `<Meta ?>` contain additional information about the structure element. The possible fields here are

- "`Referenced as object 42`": This is present on any object which is referenced (though `/Ref`) by any other object. The number is an arbitrarily chosen natural number which uniquely identifies the element and serves as a global identifier.

- "`Title: Some title`": "`Some title`" is the title as specified though the `/T` key.

- "`Language: xx-XX`": The structure element specifies language identifier `xx-XX` though `/Lang`.

- "`Expansion: Expanded`": The structure element specifies the expansion Expanded though `/E`.

- "Alternate text: Text": The structure element specifies the alternate text Text though /Alt.

- "Actual text: Text": The structure element specifies the actual text Text though /ActualText.

- "Associated files are present": At least one associated file is specified though /AF. Beside this note associated files are currently ignored.

- "Attributes:": Attributes are present. The attributes are printed in the following lines grouped by attribute owner in the form

```
Attributes:
├<Owner 1>
│├<Attr Name 1>: <Attr value 1>
│├<Attr Name 2>: <Attr value 2>
│└<Attr Name n>: <Attr value n>
├...
└<Owner n>
 ├<Attr Name 1>: <Attr value 1>
 ├<Attr Name 2>: <Attr value 2>
 └<Attr Name n>: <Attr value n>
```

  For attributes owned by a namespace the Owner 1 field is the namespace identifier. For other owners it's a slash followed by the owner name.

- "References object(s) 42, 142, 242": The element references though /Ref the elements which are marked with these identifiers.

Finally Child 1 to Child n describes the child elements. These can have one of three forms:

- Another structure element

- A object reference using OBJR. These are represented as

```
Referenced object of type <Type> on page <Page>
```

  Here <Type> represents the type of the references objct as specified by /Type. " of type <Type>" is omitted if the referenced object does not explicitly specify a type. Page is the page index on which the referenced object appears.

- Marked content. Marked content is represented as

```
Marked content on page <page>: <text>
```

Here <page> is the page index of the page on which the marked content appears and <text> is the text content of the marked content, converted to Unicode though ToUnicode maps and specified Actual-Text. Other content (including but not limited to XObjects and non-text drawing operators) is ignored.

This <text> is provided to help getting a general idea which content is marked and should not be relied upon to get a full understanding of the content of the marked content sequence.

2 Example output

An example for a simple document could be

```
Document (http://iso.org/pdf2/ssn):
└─Section (http://typesetting.eu/test/pdfns) / Sect (http://iso.org/pdf2/ssn):
  ├─H (http://iso.org/pdf2/ssn):
  │ ├─Lbl (http://iso.org/pdf2/ssn):
  │ │ └─Marked content on page 1: 1
  │ └─Marked content on page 1: First section
  └─P (http://iso.org/pdf2/ssn):
    ├──Attributes:
    │  └/Layout:
    │    ├LineHeight: 11
    │    ├SpaceAfter: 4.625
    │    └TextAlign: "Center"
    ├─Marked content on page 1: Some example content
    ├─Lbl (http://iso.org/pdf2/ssn):
    │ └─Marked content on page 1: 1
    ├─FENote (http://iso.org/pdf2/ssn):
    │ └─P (http://iso.org/pdf2/ssn):
    │   ├─Lbl (http://iso.org/pdf2/ssn):
    │   │ └─Marked content on page 1: 1
    │   └─Marked content on page 1: With a footnote
    ├─Marked content on page 1: .
    └─Link (http://iso.org/pdf2/ssn):
      ├──Alternate text: A link to a well known search engine
      └─Link (http://iso.org/pdf2/ssn):
        ├─Marked content on page 1: Google
        └─Referenced object of type Annot on page 1
```