Enhancing accessibility of structured information via 'Tagged PDF'

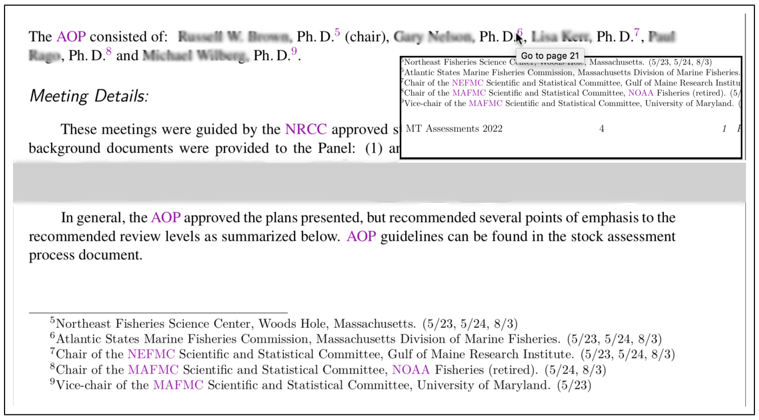
Ross Moore

It is common practice in LaTeX to use coding such as in the following listing, resulting in a layout as shown in Figure 1. (Names are smeared for privacy.)

```
The \AOP\ consisted of:
<-1st member->\footnote{<-affil->} (chair),
<-2nd member->\footnote{<-affil->},
<-3rd member->\footnote{<-affil->}, ...
```

When reading the names one 'glances' down the page to check the affiliation, or in some PDF readers, hovers over the footnote marker link to induce the pop-up, as shown in Figure 1.

Figure 1: sequence of names/authors and affiliations



The AOP consisted of: [redacted], Ph. D.[5] (chair), [redacted], Ph. D.[6], [redacted], Ph. D.[7], [redacted], Ph. D.[8] and [redacted], Ph. D.[9].

[5]Northeast Fisheries Science Center, Woods Hole, Massachusetts. (5/23, 5/24, 8/3)
[6]Atlantic States Marine Fisheries Commission, Massachusetts Division of Marine Fisheries.
[7]Chair of the NEFMC Scientific and Statistical Committee, Gulf of Maine Research Institu
[8]Chair of the MAFMC Scientific and Statistical Committee, NOAA Fisheries (retired). (5/
[9]Vice-chair of the MAFMC Scientific and Statistical Committee, University of Maryland. (

MT Assessments 2022 4 1

Meeting Details:

These meetings were guided by the NRCC approved s
background documents were provided to the Panel: (1) ar

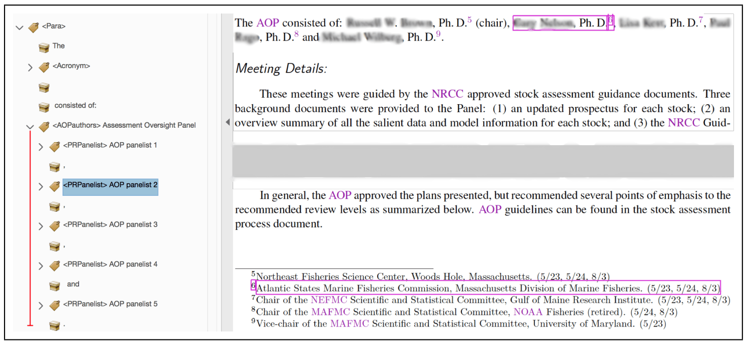
[redacted]

In general, the AOP approved the plans presented, but recommended several points of emphasis to the recommended review levels as summarized below. AOP guidelines can be found in the stock assessment process document.

[5]Northeast Fisheries Science Center, Woods Hole, Massachusetts. (5/23, 5/24, 8/3)
[6]Atlantic States Marine Fisheries Commission, Massachusetts Division of Marine Fisheries. (5/23, 5/24, 8/3)
[7]Chair of the NEFMC Scientific and Statistical Committee, Gulf of Maine Research Institute. (5/23, 5/24, 8/3)
[8]Chair of the MAFMC Scientific and Statistical Committee, NOAA Fisheries (retired). (5/24, 8/3)
[9]Vice-chair of the MAFMC Scientific and Statistical Committee, University of Maryland. (5/23)

However, what if 'glancing' is not possible? For example, the reader is visually-impaired, for whom a pop-up also is not appropriate. Even the concept of 'page' need not be relevant or useful.

Figure 2: capturing structure, using 'Tagged PDF'



```
<Para>
    The
  <Acronym>

    consisted of:
  <AOPauthors> Assessment Oversight Panel
    <PRPanelist> AOP panelist 1
      ,
    <PRPanelist> AOP panelist 2
      ,
    <PRPanelist> AOP panelist 3
      ,
    <PRPanelist> AOP panelist 4
      and
    <PRPanelist> AOP panelist 5
      .
```

The AOP consisted of: [redacted], Ph. D.[5] (chair), [redacted], Ph. D.[6], [redacted], Ph. D.[7], [redacted], Ph. D.[8] and [redacted], Ph. D.[9].

Meeting Details:

These meetings were guided by the NRCC approved stock assessment guidance documents. Three background documents were provided to the Panel: (1) an updated prospectus for each stock; (2) an overview summary of all the salient data and model information for each stock; and (3) the NRCC Guid-

[redacted]

In general, the AOP approved the plans presented, but recommended several points of emphasis to the recommended review levels as summarized below. AOP guidelines can be found in the stock assessment process document.

[5]Northeast Fisheries Science Center, Woods Hole, Massachusetts. (5/23, 5/24, 8/3)
[6]Atlantic States Marine Fisheries Commission, Massachusetts Division of Marine Fisheries. (5/23, 5/24, 8/3)
[7]Chair of the NEFMC Scientific and Statistical Committee, Gulf of Maine Research Institute. (5/23, 5/24, 8/3)
[8]Chair of the MAFMC Scientific and Statistical Committee, NOAA Fisheries (retired). (5/24, 8/3)
[9]Vice-chair of the MAFMC Scientific and Statistical Committee, University of Maryland. (5/23)

In several countries there are now anti-discrimination laws or policies[1] which for the most part embody the principle that "... agencies must give disabled employees and members of the public access to information comparable to the access available to others."

To achieve a result, as in Figure 2, the first step is to rearrange coding as in this next listing.

```
The \AOP\ consisted of:
\begin{AOPpanel}
\AOPchair{<-1st ...->\footnote{<-affil->}}
\AOPmember{<-2nd ...->\footnote{<-affil->}}
 ...
\AOPlastmember{<-5th ...->\footnote{<-affil->}}
\end{AOPpanel}
```

using definitions in a preamble or package, such as:

```
\newenvironment{AOPpanel}{\ignorespaces}{}
\newcommand{\AOPchair}[1]{\PRPpanel{#1} (chair)}
\newcommand{\AOPmember}[1]{, \PRPpanel{#1}}
\newcommand{\AOPlastmember}[1]{ and \PRPpanel{#1}.}
\newcommand{\PRPpanel}[1]{#1}
```

One can see on the right in Figure 2 that the visual layout is the same as in Figure 1. However tagging in the left panel of Figure 2 shows how the punctuation is separated out from the panelist information; with a separate structure element for each. Each use of a macro `\AOPchair`, ..., corresponds to a detail in how the information fits into the single paragraph. `\PRPpanel` handles the real information, combining both the name with the affiliation in a footnote; Figure 3 shows this in greater detail. This is all enclosed in a structure for the `{AOPpanel}` as a whole.

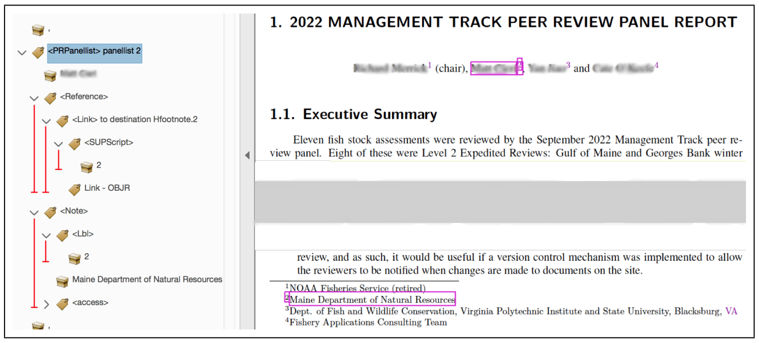
The way the user-defined macros and environment are used, provides a way to capture the 'semantics' of this sequence of panelist names and affiliations.

The reason for using a separate `\PRPpanel` macro, rather than one named `\AOP...` is because there are other similar structures in this same document. Differences in the tagging, and hence the semantics being captured, can be associated with the surrounding environment, whilst `\PRPpanel` can be common to all such situations. In Figure 3 we see part of an earlier page in the same document. The footnote numbers are earlier, and there is no existing paragraph surrounding the `{PRPauthors}` structure.

```
\newenvironment{PRPauthors}%
 {\ignorespaces}{\bigskip\bigskip}
\newcommand{\PRPchair}[1]{\PRPpanel{#1} (chair)}
\newcommand{\PRPauthor}[1]{, \PRPpanel{#1}}
\newcommand{\PRPlastauthor}[1]{ and \PRPpanel{#1}}
```

[1] https://www.section508.gov/manage/laws-and-policies/

Figure 3: tagging of the PRP authors



,
<PRPanellist> panellist 2
[redacted]
<Reference>
<Link> to destination Hfootnote.2
<SUPScript>
2
Link - OBJR
<Note>
<Lbl>
2
Maine Department of Natural Resources
<access>
,

1. 2022 MANAGEMENT TRACK PEER REVIEW PANEL REPORT

[redacted][1] (chair), [redacted][2], [redacted][3] and [redacted][4]

1.1. Executive Summary

Eleven fish stock assessments were reviewed by the September 2022 Management Track peer review panel. Eight of these were Level 2 Expedited Reviews: Gulf of Maine and Georges Bank winter

review, and as such, it would be useful if a version control mechanism was implemented to allow the reviewers to be notified when changes are made to documents on the site.

[1]NOAA Fisheries Service (retired)
[2]Maine Department of Natural Resources
[3]Dept. of Fish and Wildlife Conservation, Virginia Polytechnic Institute and State University, Blacksburg, VA
[4]Fishery Applications Consulting Team

Figure 3 shows the result from `\PRPpanel`, as tagging a `<PRPpanelist>` structure, numbered in sequence. Firstly the name is placed, followed by a `<Reference>` structure, consisting of the superscripted `\footnotemark` as anchor for an active hyperlink. This is followed immediately by the `<Note>` structure coming from the `\footnotetext`, repeating the mark as a `<Lbl>` then placing the text.

For a person reading with Assistive Technology (AT) navigating via structure, the analog of 'glancing down the page' is to simply move by a single element from the `<Reference>` to its following `<Note>`, or to use the hyperlink to get to the same location.

While the first `\footnotemark` uses a custom `<SupScript>` structure, which is 'Role-mapped' to a standard `<Span>`, the second usage does not. There the `<Lbl>` structure has an attribute dictionary:

```
<< /O /Layout  /TextPosition /Sup >>
```

For deriving to HTML (see below) this attribute cannot be applied to the `<Link>` structure; hence the extra `<SupScript>` is used.

Constructing a 'Tagged PDF' document which is structured as shown in Figures 2 and 3 requires a highly sophisticated LaTeX macro system to produce the tagging of both structure and content. The **`tagpdf`** LaTeX package is under development, but not yet ready for real-world documents such as the one shown here. Instead this author and colleagues use coding developed by the author, under the name `tpdf`. A full description of `tpdf` is beyond the scope of this article.

However, it should be clear that part of any such system must include extending or re-defining the expansions of the macros used for user input; that is, the `\AOP...` and `\PRP...` macros, the environments `{AOPpanel}` and `{PRPauthors}` as well as LaTeX internal commands (e.g., for processing footnotes and hyperlinks) and environments generally. With environments there are 'hooks' which allow for additional coding to be executed in 4 different places via `\AddToHook{`⟨hook⟩`}`; so `env/AOPpanel/before` and `env/AOPpanel/after` allow coding to be added before and after the environment itself is processed. These can be used to control how the environment's structure fits into that of the surrounding document. Similarly, hooks `env/AOPpanel/begin` and `env/AOPpanel/end` can place more coding *inside* the environment, affecting how information is handled within the environment itself, without affecting outside. Both pairs of hooks are useful.

Furthermore, one can modify `\AOPpanel` and `\endAOPpanel` from the expansions provided when the environment was set up using `\newenvironment`. There is a `\patchcmd` macro in the **`etoolbox`** package, but often it is easier to redefine the expansions directly, whilst keeping a copy of the original expansion in case needed within the redefined version.

```
\NewCommandCopy\LTX@AOPpanel\AOPpanel
\newcommand\TPDF@AOPpanel[1]{....}
\NewCommandCopy\AOPpanel\TPDF@AOPpanel
```

Use similar coding for adjusting `\endAOPpanel`.

For example, the `\TPDF@PRPpanel` replacement for `\PRPpanel` must implement the following tasks:

- close off any preceding text, leaving the `<Para>` open;
- increment the counter for the kind of panelist;
- start the next `<PRPpanelist>` structure, numbered appropriately;
- process the argument `#1`; that is, place the name as tagged text, then allow a `\footnote{...}` command to be processed with its argument.

It is a modified version of internal commands called by `\footnote` that handles tagging of the footnote markers and text, resulting in the tagging seen in Figure 3. After that is all done, structure is closed to the level of the original `<Para>`, to allow any further panelists to be included, as seen in Figure 2.

PDF reuse as HTML and XML, attributes and classes

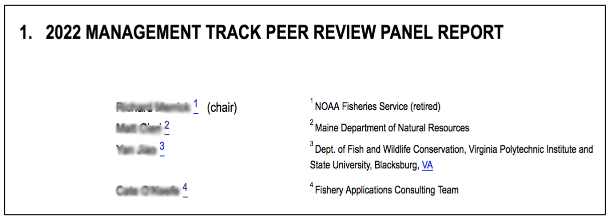
Typically visually impaired people prefer to read documents via HTML, using Assistive Technology (AT) such as screen-readers, Braille keyboards, and more. Previously PDF files have been regarded as inaccessible, compared with HTML. Standards such as PDF/UA are intended to redress this situation; but old habits and preferences cannot be expected to be altered until AT software updates sufficiently to better handle Tagged PDFs built to conform to newer standards.

Figure 4: HTML coding derived from `\PRPpanel` usage

```
<span> </span>
<div class="PRPanelist" data-pdf-se-type-original="PRPanelist" data-pdf-se-type="Div" id="PRPanelist.6">
(-- 2nd author --)<a href="#Note.15" class="Xlink" data-pdf-se-type="Link"
aria-details="Note.15" aria-label="AOP panelist 2 affiliation" id="Link.0994">
<sup data-pdf-se-type-original="SUPScript" data-pdf-se-type="Span" id="SUPScript.06">6</sup></a>
<p class="PRPaffil" data-pdf-se-type="Note" name="Hfootnote.6" id="Note.15">
<sup data-pdf-se-type="Lbl" id="Lbl.242"><span>6&#xa0;</span></sup>
 Atlantic States Marine Fisheries Commission, Massachusetts Division of Marine Fisheries.  (5/23, 5/24, 8/3)
<span class="Access" data-pdf-se-type-original="access" data-pdf-se-type="Span" id="access.242"><span>
</span></span></p></div>
```

Fortunately there is good online software[2] that 'derives' a well-structured single HTML file from a valid PDF/UA document. Figure 5 shows the result of using this with our real-world example PDF, for a sequence of panelists, viewed in the Firefox browser. Being a single HTML file, there is no concept of 'page' to determine where to place footnotes. Instead, the `<Note>` structure is floated to the right using CSS rules, explained below.

Figure 5: HTML version of the PRP authors



1. 2022 MANAGEMENT TRACK PEER REVIEW PANEL REPORT

[redacted] 1 (chair)	1 NOAA Fisheries Service (retired)
[redacted] 2	2 Maine Department of Natural Resources
[redacted] 3	3 Dept. of Fish and Wildlife Conservation, Virginia Polytechnic Institute and State University, Blacksburg, VA
[redacted] 4	4 Fishery Applications Consulting Team

To better understand Figure 5, one needs to look at part of the raw HTML coding, as shown above in Figure 4. One sees that each HTML tag (`<div`, `<sup`, `<p`, `<span`) has a unique `id="..."` attribute, inheriting the corresponding structure's unique name from the PDF. Namely, the paragraph with `id="Note.15"`, which comes from the `<Note>` structure having the footnote text as can be seen in Figure 1, becomes the target for the hyperlink from `<a href="#Note.15"`.

Also shown, as lighter colored attributes and values, are the PDF structure-element names, with `-original` indicating a 'Custom' structure name. This maps to a 'Standard' name as shown in the attribute which follows. Such user-defined HTML attributes would be ignored by web browsers, unless specially set up to handle them.

For layout on the HTML page, the important attributes are `class="..."`. The specified names and affiliations are subject to CSS style rules:

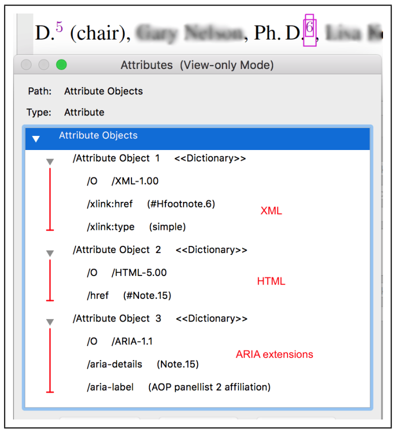
```
.PRPanelist { clear:right; }
.PRPaffil { float:right; font-size:small;
  margin:0 0 0 5px; width:60%; }
```

[2] ngPDF: `https://ngpdf.com/`

This use of `float` and `clear` is what turns an otherwise horizontal sequence into what appears to be a 2-column vertical listing.

For Accessibility, having anchor text being just a single digit, say '2', is not very informative. Hence the `aria-label="..."` tells what kind of information is found at the link target. A longer description, via `aria-details="..."`, is at the target location itself. With such attributes for all internal hyperlinks, WCAG Level A Success Criterion 2.4.4[3] is fulfilled for this document. This uses the concept of 'Accessible Name',[4] here built from the anchor-text and aria-label value.

Figure 6: link attributes



D.5 (chair), [redacted], Ph. D.6, [redacted]

Attributes (View-only Mode)

Path: Attribute Objects
Type: Attribute

```
▼ Attribute Objects
    /Attribute Object 1    <<Dictionary>>
        /O    /XML-1.00
        /xlink:href    (#Hfootnote.6)
        /xlink:type    (simple)
    /Attribute Object 2    <<Dictionary>>
        /O    /HTML-5.00
        /href    (#Note.15)
    /Attribute Object 3    <<Dictionary>>
        /O    /ARIA-1.1
        /aria-details    (Note.15)
        /aria-label    (AOP panellist 2 affiliation)
```

Figure 6 shows the array of attribute dictionaries specified for the hyperlink within the PDF document, as shown in figures 2 and 5. These control how the link works when exported into either HTML or XML formats. Notice how the XML version uses a target of `Hfootnote.6`, which is the name of the PDF 'Destination'. On the other hand, HTML requires the 'Structure Destination' of `Note.15`.

Within the PDF, the hyperlink is implemented as an 'Annotation' of subtype 'Link', as shown in

[3] Web Content Accessibility Guidelines: `https://www.w3.org/TR/WCAG21/#link-purpose-in-context`

[4] `https://www.w3.org/WAI/ARIA/apg/practices/names-and-descriptions/`

Figure 7: Link annotation dictionary

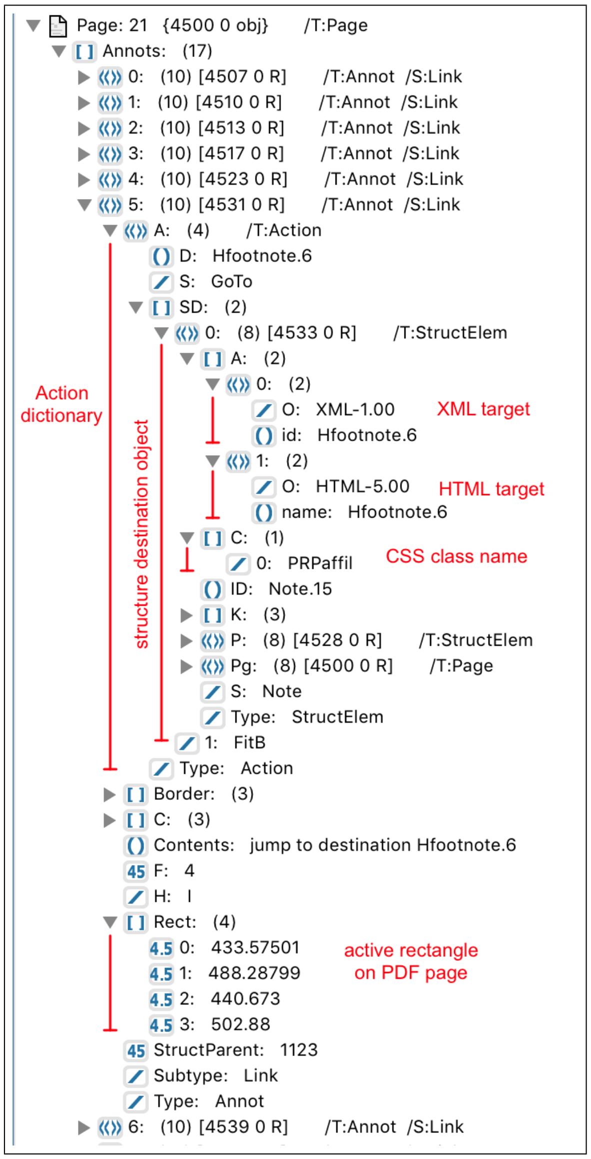
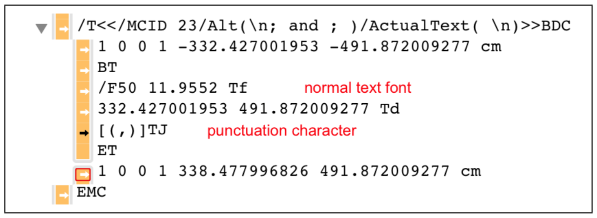

```
Page: 21  {4500 0 obj}     /T:Page
  Annots:  (17)
    0:  (10) [4507 0 R]     /T:Annot  /S:Link
    1:  (10) [4510 0 R]     /T:Annot  /S:Link
    2:  (10) [4513 0 R]     /T:Annot  /S:Link
    3:  (10) [4517 0 R]     /T:Annot  /S:Link
    4:  (10) [4523 0 R]     /T:Annot  /S:Link
    5:  (10) [4531 0 R]     /T:Annot  /S:Link
      A:  (4)     /T:Action
        D:  Hfootnote.6
        S:  GoTo
        SD:  (2)
          0:  (8) [4533 0 R]     /T:StructElem
            A:  (2)
              0:  (2)
                O:  XML-1.00
                id:  Hfootnote.6
              1:  (2)
                O:  HTML-5.00
                name:  Hfootnote.6
            C:  (1)
              0:  PRPaffil
            ID:  Note.15
            K:  (3)
            P:  (8) [4528 0 R]     /T:StructElem
            Pg:  (8) [4500 0 R]     /T:Page
            S:  Note
            Type:  StructElem
          1:  FitB
        Type:  Action
      Border:  (3)
      C:  (3)
      Contents:  jump to destination Hfootnote.6
      F:  4
      H:  I
      Rect:  (4)
        0:  433.57501
        1:  488.28799
        2:  440.673
        3:  502.88
      StructParent:  1123
      Subtype:  Link
      Type:  Annot
    6:  (10) [4539 0 R]     /T:Annot  /S:Link
```

Figure 7. This dictionary includes a clickable 'Rectangle' on the specific page, as well as specifying the 'Action' to be taken when clicked. The D field value of `Hfootnote.6` names the target within the PDF, while `SD` provides a reference to the structure, having object number 4533 and named as `Note.15`, found at that location; that is, a 'Structure Destination'. Notice how this object also provides its name for export to both XML and HTML formats, specifying the respective 'Attribute' type (either `id` or `name`). Also there is the 'Class' name of `PRPaffil`, used for CSS styling as discussed earlier.

One further detail, seen in Figure 5, is that the punctuation has been suppressed to become simply `<span> </span>`. This is achieved in the PDF content stream as shown in Figure 8. Appearing as a comma within the PDF using the 'show string' of `[(,)]TJ`, the `/Alt(...)` and `/ActualText(...)` provide alternatives that can be spoken by a screen-reader or used with text extraction.

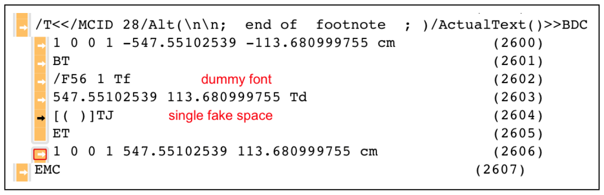
Figure 8: handling variants for punctuation


```
/T<</MCID 23/Alt(\n; and ; )/ActualText( \n)>>BDC
  1 0 0 1 -332.427001953 -491.872009277 cm
  BT
  /F50 11.9552 Tf
  332.427001953 491.872009277 Td
  [(,)]TJ
  ET
  1 0 0 1 338.477996826 491.872009277 cm
EMC
```

Access tags

A special kind of structure and content is provided by so-called 'Access' tags, whose presence can be seen by the `<access>` in Figure 3, and near the bottom of the HTML listing in Figure 5. These use `/Alt` and `/ActualText` similarly to the above, having a 'show string' of `[( )]TJ` (see Figure 9) which places a space character, but using the 'dummy font' having width 0.001 pt, as obtained with the `\pdffakespace` primitive. This space is imperceptible within the typeset PDF, yet is still selectable. With an empty string for `/ActualText` it adds no content to the HTML and XML exports.

Figure 9: 'Access Tag' after footnote


```
/T<</MCID 28/Alt(\n\n;  end of  footnote  ; )/ActualText()>>BDC
  1 0 0 1 -547.55102539 -113.680999755 cm
  BT
  /F56 1 Tf
  547.55102539 113.680999755 Td
  [( )]TJ
  ET
  1 0 0 1 547.55102539 113.680999755 cm
EMC
```

However, a non-empty `/Alt(...)` string, as in Figure 9, affects what is spoken by some (but not all) screen-reading software, including Adobe Reader's 'Read Out Loud' facility. With ' ; ' creating a slight pause, the descriptive markup conveyed via these 'Access' tags is separated from actual content being read. Furthermore, the complete stream that is read can be exported as 'Text (Accessible)' from Adobe software. In Figure 10 we see how the sequence of names shown in Figures 1 and 2 will be spoken or exported. The judicious use of `\n\n` inserts line-breaks to help break up the text stream nicely upon export.

Conclusions

Here we have attempted to show, using a 'real-world' example document, how with a relatively simple

Figure 10: Accessible Text for screen-reading

```
The A O P consisted of: ; start of AOPauthors block ;  (-- 1st author --) , Ph.D.
; refer to footnote 5 ;
; start of footnote 5 ; Northeast Fisheries Science Center, Woods Hole, Massachusetts. (5/23, 5/24, 8/3)
; end of footnote ; (chair)
; and ; (-- 2nd author --), Ph.D.
; refer to footnote 6 ;
; start of footnote 6 ; Atlantic States Marine Fisheries Commission, Massachusetts Division of Marine
 Fisheries. (5/23, 5/24, 8/3)
; end of footnote ;
; and ; (-- 3rd author --), Ph.D.
 ...
```

adjustment of source coding, the semantics of structured information can be captured explicitly within a 'Tagged PDF' document. Each use of a user-defined macro or environment provides a place where extra meaning can be captured for presentation, not just visually on a page, but preserved for export into other formats. The various figures show how and where this information can be stored inside the PDF, and also where it then appears within derived HTML and Text-only export formats.

By giving consideration also to CSS styling for the exported HTML, and WCAG recommendations for enhancing Accessibility, one can indeed provide "disabled ... access ... comparable to ... others". Even the tiniest details, such as punctuation, can be handled so as to enhance each alternative view, without compromising the high-quality visual layout that is (LA)TEX's hallmark.

Technical notes

All the images and figures used in this article originate from a 'real-world' document intended for publication and release into the public domain.[5] Researchers' names have been deliberately blurred, so as not to be easily extractable. Similarly, names have been replaced by generic place-holder strings in the textual outputs of Figures 5 and 10. Those authors and panel members have nothing whatsoever to do with the techniques of 'Tagged PDF' production and the export into other formats being discussed here.

Figure 1 is a (doctored) view using Apple's 'Preview' application, using 'TEXshop'[6] on a MacBook; most of the page's content was removed and the rest compacted to better display the relevant parts. Figures 2, 3 and 6–9 are views using Adobe's 'Acrobat Pro' application,[7] involving various panels to display visual content, 'Tags' tree, an 'Attribute' array, an internal view of a portion of the 'Structure Tree', and internal parts of a 'Page Content' stream in raw PDF. All graphic editing of screen-shots was done using 'GraphicConverter'[8] for macOS.

Figure 4 is a screenshot, taken while using the 'Firefox' browser, of the HTML webpage derived by 'ngPDF' at the website[2] stated earlier. Figure 5 is from a text-editor application ('Erbele'[9] for macOS) which was used to work with the HTML document source. With Figure 10 Apple's 'TextEdit'[10] was used to present the Text (Accessible), exported from 'Acrobat Pro'; but any text-editing software would be sufficient for this. Other textual code-listings were done using either 'Erbele' or the default editor for 'TEXshop', which was the application for composing this article. It was also the application used to build the "real-world" 185-page 'Tagged PDF' document that has supplied the examples, built using LATEX. All software used was running on an Apple 'MacBook Pro'[11] device.

◇ Ross Moore
School of Mathematical and Physical Sciences
Macquarie University
Sydney, Australia
ross.moore (at) mq dot edu dot au
https://researchers.mq.edu.au/en/persons/ross-moore

[5] at http://www.science.mq.edu.au/~ross/TaggedPDF/FallMT2022/ as FallMT2022.html and FallMT2022.pdf.

[6] TEXshop: https://pages.uoregon.edu/koch/texshop/

[7] Acrobat Pro: https://www.adobe.com/products/acrobat-pro-cc.html

[8] GraphicConverter: https://www.lemkesoft.de/en/products/graphicconverter/

[9] Erbele text editor: https://apps.apple.com/de/app/erbele/id1595456360

[10] TextEdit: https://support.apple.com/en-au/guide/textedit/welcome/mac

[11] MacBook Pro: https://www.apple.com/macbook-pro/