

1 Introduction

The `\Configure` command is the most powerful user command in \TeX 4ht system. It helps to add various kinds of hooks to insert target markup code into the desired locations in and around various types of content. For instance, take the example of `\section{...}` in \LaTeX . A typical example will be:

```
\section{Introduction}
```

`\Configure` for `\section` provides four (4) hooks in the following style:

```
\Configure{section}
  {<beginning of section>} {<end of the section>}
  {<before section heading>}{<after section heading>}
```

Suppose our target markup requires the following pattern of markup for a section heading:

```
<section id="sec1">
  <title>Introduction</title>
  <para>
    ...
  </para>
</section>
```

We will accomplish the above target by configuring the section macros in the following manner:

```
\Configure{section}
  {\EndP\IgnorePar\Tg<section id="sec\thesection">}
  {\EndP\Tg</section>}
  {\Tg<title>}
  {\Tg</title>\ShowPar}
```

The above code will help translate any \LaTeX section heading into the required markup scheme as provided in the example. Macros like `\EndP`, `\IgnorePar` and `\ShowPar` will be discussed a little later. These are hooks to insert begin and end of paragraph code into the content appropriately. Here, in this document, we shall list most of the `\Configure` hooks available in \TeX 4ht system and describe briefly how those can be used for configuring most of the \LaTeX commands popularly used in documents.

```
\Configure{PROLOG}.....1
```

#1 Comma separated list of hooks to appear before HTML. Each hook E is declared to be configurable by an instruction of the form `\NewConfigure {E}{1}`. A star * prefix calls for accumulative configurations.

Example

```
\Configure{PROLOG}{VERSION,DOCTYPE,*XML-STYLESHEET}
\Configure{VERSION}
  {\HCode{<?xml version="1.0"?>}}
```

`\Configure{ext}.....1`

#1: default extension name for target files (recorded in `\:html`). Can also be requested through a command line option `ext=...`

`\Preamble.....0`

Records the list of the requested options. Defined upon entering the environment `\Preamble{...}....\EndPreamble`, to replace the earlier version of `\Preamble`.

`\ifOption 3`

#1 Argument to be checked whether it is a given option.

#2 True part

#3 False part

2 Wrapper for the Document

`\Configure{DOCTYPE}.....1`
`\Configure{HTML}.....2`
`\Configure{HEAD}.....2`
`\Configure{@HEAD}.....1`
`\Configure{BODY}.....2`
`\Configure{TITLE+}.....1`
`\Configure{TITLE}.....2`
`\Configure{@TITLE}.....1`
`\Configure{Preamble}.....2`

```
<DOCTYPE>
<HTML 1>
  <HEAD 1>
    <TITLE 1>
      <@TITLE>
      <TITLE+>
    <TITLE 2>
    <@HEAD>
  <HEAD 2>
  <BODY 1>
  .....
  <BODY 2>
</HTML 2>
```

The `\Configure{@HEAD}{...}` command is additive, concatenating the content of all of its appearances. An empty parameter requests the cancellation of the earlier contributions.

For instance,

```
\Configure{@HEAD}{A}
\Configure{@HEAD}{}
\Configure{@HEAD}{B}
\Configure{@HEAD}{C}
```

contributes ‘BC’.

The `\Configure{TITLE+}` provides the content for the title, `\Configure{TITLE}` sets the envelop, and `\Configure{@TITLE}` acts as a hook for introducing localized configurations. As is the case for `\Configure{@HEAD}`, the contribution of `\Configure{@TITLE}` is also additive.

These configurations should be introduced early enough in the compilation. For instance, in the case of LaTeX, between `\Preamble` and `\begin{document}` of a local configuration file.

```
\Preamble
  %% here %%
\begin{document}
  ...
\EndPreamble

\Configure{@BODY}.....1
\Configure{@/BODY}.....1
```

Variants of `\Configure{@HEAD}` which contribute their content, respectively, after `<body>` and before `</body>`.

```
\Configure{CutAtTITLE+}.....1
\Configure{HPageTITLE+}.....1
```

#1 An insertion just before the content of `<TITLE>`. If **#1** is a one parametric macro, it gets the title content for an argument.

3 Support for Sectioning Commands

```
\Configure{unit-name} .....4
```

- #1** start
- #2** end
- #3** before title
- #4** after title

Example

```
\Configure{section}
  {\HCode{<section>}}    {\HCode{</section>}}
  {\HCode{<title>}}     {\HCode{</title>}}
\ConfigureMark{unit-name}.....1
```

Defines a macro `<unit-name>HMark` to hold the given argument. Upon entering the unit, `\TitleMark` gets the content of this macro.

Some built-in configurations of \TeX 4ht require an argument for the `<unit-name>HMark` commands. For safety, these commands should always be followed by a, possibly empty, argument. The argument should be a separator between the title mark and its content.

Example

```
\Configure{section}
  {}{}
  {\HCode{<h3>}\TitleMark\space}{\HCode{</h3>}}
\ConfigureMark{section}{\thesection}
```

3.1 ToC link

```
\Configure{toTocLink}.....2
```

Each unit title contains a `\Link{...}{...}...EndLink` command. The first argument of `\Link` points to the first table of contents referencing the title. The second argument provides an anchor for references to the title (mainly from tables of contents).

The package option `section+` requests the inclusion of the title within the anchor. Without this option, the link command resides between the title mark and its content.

The `\Configure{toTocLink}` command is provided for configuring the `\Link` and `\EndLink` instructions. In the default setting, when the `sections+` option is not activated, the `\Link` command is altered to replace its first argument with an empty argument.

Example

```
\Configure{toTocLink}
  {\Link}
  {\ifx \TitleMark\sectionHMark
    \Picture[\up]{haut.jpg align="right"}%
    \EndLink
    \TitleMark\space
  \else \EndLink \fi
}
\def\up{[up]}

\Configure{toToc}.....2
```

#1 unit type

#2 desired contents type (if empty, 'unit type' is assumed)

Example

```
\Configure{toToc}{chapter}{likechapter}
```

Introduces chapter as likechapter into toc

#1 empty: stop adding entries of 'unit type' to toc

@: add entries of 'unit type' to toc

?: resume mode in effect before the last stop

#2 unit type

Example

```
\Configure{toToc}{chapter}
\chapter{...}
\Configure{toToc}{@}{chapter}

\Configure{writetoc}.....1
```

#1 Configuration material for the insertion instruction. New configurations are added to those request earlier by the command. An empty argument cancels the earlier contributions.

```
\NoLink.....1
```

Ignore option `section+` for sections of type #1.

```
\TitleCount
```

Count of entries submitted to the `toc` file

```
\Configure{NoSection}.....2
```

Insertions around the parameters of sectioning commands, applied when the parameters are not used to create titles for the divisions.

```
\CutAt{#1,#2,#3,...}
```

#1 section type to be placed in a separate web page.
#2,#3,... end delimiting section types, other than #1, for the web pages A + before #1 requests hypertext buttons for the web pages.

Example

```
\CutAt{mychapter,myappendix,mypart}
\CutAt{+myappendix,mychapter,mypart}
```

Cut points at arbitrary points can be introduced by introducing section-like commands in a manner similar to

```
\NewSection\mysection{}
\CutAt{mysection}
\Configure{+CutAt}.....3
```

#1 sectioning type

#2 before

#3 after

Requests delimiters for the `\CutAt` buttons of the specified sectioning type.

Example

```
\Configure{+CutAt}{mysection}{{}}{}}
```

```
\PauseCutAt{#1}  
\ContCutAt{#1}
```

#1 section type

```
\Configure{CutAt-filename} .....1
```

A 2-parameter hook for tailoring section-based filenames. The section type is available through #1. The section title is accessible through #2.

Example

```
\Configure{CutAt-filename}{\NextFile{#1-#2.html}}
```

4 Tables of Contents

Created from the entries collected in the previous compilation within a `jobname.4tc` file.

```
\ConfigureToc{unit-name} .....4
```

#1 before unit number

#2 before content

#3 before page number

#4 at end

- Empty arguments request the omission of the corresponding field.
- `\TocCount` Specifies the entry count withing the `jobname.4tc` file.
- `\TitleCount` Count of entries submitted to the toc file.
- An alternative to `\ConfigureToc{unit-name}`:

```
\def\toc<unit-name>#1#2#3{<before unit number>#1<before content>#2%  
  <before page number>#3<at end>}
```

Example

```
\ConfigureToc{section}  
{}  
{\Picture[*]{pic.jpg width="13" height="13"}~}  
{}  
{\HCode{<br />}}
```

```
\Configure{TocLink}.....4
```

Configures the link offered in the third arguments of `\ConfigureToc`.

Example

```
\Configure{TocLink}{\Link{#2}{#3}#4\EndLink}

\TocAt{#1,#2,#3,...}
```

#1 section type for which local tables of contents `\Toc #1` are requested.

#2,#3,... sectioning types to be included in the tables of contents.

The non-leading arguments may be preceded by slashes `/`, in which cases the arguments specify end points for the tables.

The default setting requests automatic insertion of the local tables immediately after the sectioning heads.

A star `*` character may be introduced, between the `\TocAt` and the left brace, to request the appearances of the tables of contents at the end of the units' prefaces.

A hyphen `-` character, on the other hand, disables the automatic insertions of the local tables.

In case of a single argument, the command removes the existing definition of `\Toc#1`.

Example

```
\TocAt{mychapter,mysection,mysubsection,/myappendix,/mypart}
\TocAt-{mysection,mysubsection,/mylikesection}
\section{...}\Tocmysection
```

The definition of the local table of contents can be redefined within `\csname Toc#1\endcsname`.

Example

```
\TocAt{section}
\def\Tocsection{\TableOfContents[section]}

\Css{div.sectionTOCS {
    width : 30%;
    float : right;
    text-align : left;
    vertical-align : top;
    margin-left : 1em;
    font-size : 85%;
    background-color : #DDDDDD;
}}
```

Example

Table of content before the section title.

```

\Configure{section}{}{}
  {\Tocsection \let\saveTocsection=\Tocsection
   \def\Tocsection{\let\Tocsection=\saveTocsection}%
   \ifvmode \IgnorePar\fi \EndP\IgnorePar
   \HCode{<h3 class="sectionHead">\TitleMark\space\HTMLParOff}
   {\HCode{</h3>\HTMLParOn\ShowPar \IgnoreIndent \par}

```

```

\Configure{TocAt}.....2
\Configure{TocAt*}.....2

```

- #1 before the tables of contents
- #2 after the tables of contents

5 Navigation Links for Sectioning Divisions

```

\Configure{crosslinks}.....8

```

- #1 left delimiter
- #2 right delimiter
- #3 next
- #4 previous
- #5 previous-tail
- #6 front
- #7 tail
- #8 up

The content to be displayed in the pointers.

```

\Configure{crosslinks*}.....1--7

```

Links to be included and their order. Available options: `next`, `prev`, `prevtail`, `tail`, `front`, `up`. The last argument must be empty.

Default:

```

\Configure{crosslinks*}
  {next}
  {prev}
  {prevtail}
  {tail} {front}
  {up}
  {}

```

```

\Configure{crosslinks+}.....4

```

- #1 before top menu
- #2 after top menu
- #3 before bottom menu

#4 after bottom menu

The top cross links are omitted, if both #1 and #2 are empty. The bottom cross links are omitted, if both #3 and #4 are empty.

`\Configure{next}.....1`

#1 the anchor of the next button of the front page.

Default: The value provided in `\Configure{crosslinks}`.

`\Configure{next+}.....2`

#1 before the next button of the front page, when the `next` option is active.

#2 after the button

Default: The values provided in `\Configure{crosslinks}`.

`\Configure{crosslinks:next}.....1`
`\Configure{crosslinks:prev}.....1`
`\Configure{crosslinks:prevtail}.....1`
`\Configure{crosslinks:tail}.....1`
`\Configure{crosslinks:front}.....1`
`\Configure{crosslinks:up}.....1`

#1 local configurations for the delimiters and hooks

`\Configure{crosslinks-}.....2`

Asks to show linkless buttons with the following insertions.

#1 before

#2 after

The default values are used, if both #1 and #2 are empty.

Example

```
\Configure{crosslinks-}{}{}

\Configure{crosslinks-}
  {\HCode{<span class="hidden">}}
  {\HCode{</span>}}
\Css{span.hidden {visibility:hidden;}}
```

6 Paragraphs

`\Configure{HtmlPar}.....4`

#1 content at the start non-indented paragraphs

#2 content at the start indented paragraphs

#3 insertion into `\EndP` , at the start of non-indented paragraphs

#4 insertion into `\EndP` , at the start of indented paragraphs

`\HtmlParOff`
`\HtmlParOn`

`\IgnorePar` Asks to ignore the next paragraph
`\ShowPar` Asks to take into account the following paragraphs

`\IgnoreIndent` asks to ignore indentation in the next paragraph
`\ShowIndent` asks to check indentation in the following paragraphs

`\SaveEndP` Saves the content of `\EndP`, and sets it to empty content
`\RecallEndP`

`\SaveHtmlPar`
`\RecallHtmlPar`

Example

```
\Configure{@BODY}
  {\ifvmode \IgnorePar\fi \EndP
   \HCode{<div>\par\ShowPar}
\Configure{@/BODY}
  {\ifvmode \IgnorePar\fi \EndP
   \HCode{</div>}}
```