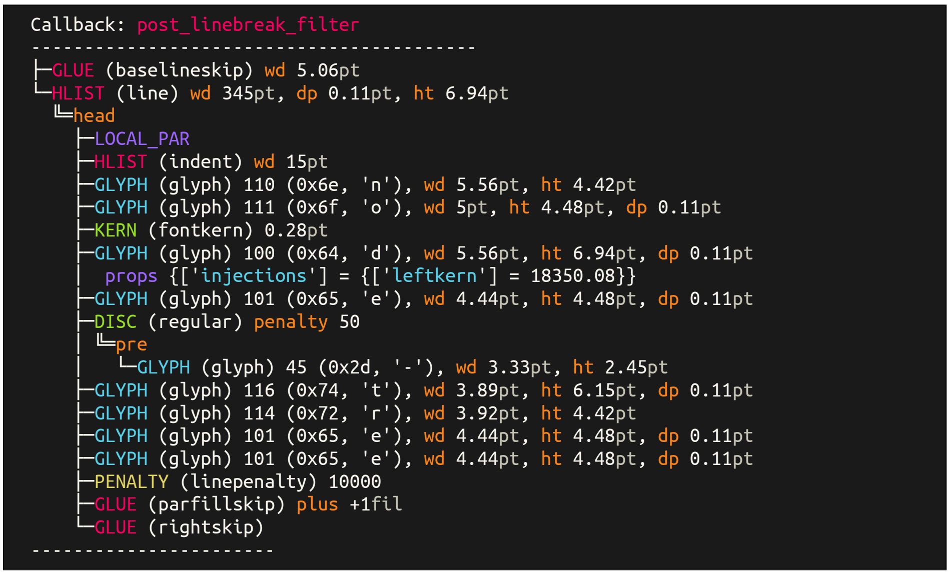
The **nodetree** package

Josef Friedrich
josef@friedrich.rocks
github.com/Josef-Friedrich/nodetree

v2.2.1 from 2022/12/17



```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 5.06pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 6.94pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 110 (0x6e, 'n'), wd 5.56pt, ht 4.42pt
    ├─GLYPH (glyph) 111 (0x6f, 'o'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─KERN (fontkern) 0.28pt
    ├─GLYPH (glyph) 100 (0x64, 'd'), wd 5.56pt, ht 6.94pt, dp 0.11pt
    │  props {['injections'] = {['leftkern'] = 18350.08}}
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    ├─DISC (regular) penalty 50
    │ ╚═pre
    │   └─GLYPH (glyph) 45 (0x2d, '-'), wd 3.33pt, ht 2.45pt
    ├─GLYPH (glyph) 116 (0x74, 't'), wd 3.89pt, ht 6.15pt, dp 0.11pt
    ├─GLYPH (glyph) 114 (0x72, 'r'), wd 3.92pt, ht 4.42pt
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

Contents

1 Abstract

`nodetree` is a development package that visualizes the structure of node lists. `nodetree` shows its debug informations in the consoles' output when you compile a LuaTEX file. It uses a similar visual representation for node lists as the UNIX `tree` command does for a folder tree.

Node lists are the main building blocks of each document generated by the TEX engine *LuaTEX*. The package `nodetree` doesn't change the rendered document. The tree view can only be seen when using a terminal to generate the document.

`nodetree` is inspired by a gist from Patrick Gundlach.

2 Usage

The package `nodetree` has four usage scenarios. It can be used as a standalone Lua module, as a plain LuaTEX, a LuaLATEX package or as package to embed nodetree views in a LuaLATEX document.

2.1 As a plain LuaTeX package

Run `luatex luatex-test.tex` for example to list the nodes using LuaTeX.

```
\input{nodetree.tex}
\NodetreeRegisterCallback{postline}

Lorem ipsum dolor.
\bye
```

2.1.1 Available macros

Macro name	Reference
`\NodetreeRegisterCallback{`⟨*callbacks*⟩`}`	Page 10, Section 3.1
`\NodetreeUnregisterCallback{`⟨*callbacks*⟩`}`	Page 10, Section 3.2
`\NodetreeSetOption[`⟨*option*⟩`]{`⟨*value*⟩`}`	Page 10, Section 3.3
`\NodetreeResetOption{`⟨*option*⟩`}`	Page 10, Section 3.4
`\NodetreeReset`	Page 10, Section 3.6

2.1.2 Available options

Option name	Reference
`callback`	Page 12, Section 5.1
`channel`	Page 12, Section 5.2
`verbosity`	Page 13, Section 5.3
`color`	Page 13, Section 5.4
`unit`	Page 13, Section 5.5
`decimalplaces`	Page 15, Section 5.6

2.2 As a LuaLaTeX package

Or run `lualatex lualatex-test.tex` to show a node tree using LuaLaTeX. In LuaLaTeX you can omit `\NodetreeRegisterCallback{postline}`. `\usepackage{nodetree}` registers automatically the `post_linebreak_filter`. If you don't want debug the `post_linebreak_filter` use `\NodetreeUnregisterCallback{postline}`.

```
\documentclass{article}
\usepackage{nodetree}

\begin{document}
Lorem ipsum dolor.
\end{document}
```

2.2.1 Available macros

Macro name	**Reference**
`\NodetreeRegisterCallback{`⟨*callbacks*⟩`}`	Page 10, Section 3.1
`\NodetreeUnregisterCallback{`⟨*callbacks*⟩`}`	Page 10, Section 3.2
`\NodetreeSetOption[`⟨*option*⟩`]{`⟨*value*⟩`}`	Page 10, Section 3.3
`\NodetreeResetOption{`⟨*option*⟩`}`	Page 10, Section 3.4
`\NodetreeReset`	Page 10, Section 3.6
`\NodetreeSet{`⟨*kv-options*⟩`}`	Page 10, Section 3.5

2.2.2 Available options

Option name	**Reference**
`callback`	Page 12, Section 5.1
`channel`	Page 12, Section 5.2
`verbosity`	Page 13, Section 5.3
`color`	Page 13, Section 5.4
`unit`	Page 13, Section 5.5
`decimalplaces`	Page 15, Section 5.6

2.3 As a Lua module

Import the Lua module of the package inside `\directlua{}` with this command: `local nodetree = require('nodetree')`. Then use the Lua function `nodetree.print(head, options)` to debug nodes inside your Lua code.

```
local nodetree = require('nodetree')

local rule1 = node.new('rule')
rule1.width  = 20 * 65536
rule1.height = 10 * 65536
rule1.depth  = 10 * 65536
nodetree.print(vbox)
```

The function `nodetree.print()` takes as a second argument a Lua table to configure the output.

```
nodetree.print(vbox, { verbosity = 2, unit = 'cm' })
```

This are the default options:

```
options =  {
  callback = 'post_linebreak_filter',
  channel = 'term',
  color = 'colored',
  decimalplaces = 2,
  unit = 'pt',
  verbosity = 1,
}
```

The following code snippet demonstrates the usage in LuaTeX. `head` is the current node.

```
  \directlua{
  local nodetree = require('nodetree')
  local test = function (head)
    nodetree.print(head)
  end
  callback.register('post_linebreak_filter', test)
}

Lorem ipsum dolor.
\bye
```

This example illustrates how the function has to be applied in LuaLaTeX.

```
\documentclass{article}
\usepackage{nodetree}

\begin{document}

\directlua{
  local nodetree = require('nodetree')
  local test = function (head)
    nodetree.print(head)
  end
  luatexbase.add_to_callback('post_linebreak_filter', test, 'test')
}

Lorem ipsum dolor.
```

```
\end{document}
```

2.4 The package nodetree-embed

The single purpose of this auxiliary package is to provide a view similar to a terminal (console) output. This view mimics the output of `nodetree` in a terminal. The view can be embedded in a LuaLaTeX file. You have to compile documents using this embedded view with the option `--shell-escape`. The main environment of this package is `NodetreeEmbed`. Markup inside this environment is written into a temporary LaTeX file. This file is compiled in the background by `latexmk` and the `nodetree` output is embded into this view. The following list shows the single intermediate steps:

1. `jobname.tex`

```
\begin{NodetreeEmbedEnv}
nodetree
\end{NodetreeEmbedEnv}
```

2. `_nodetree-jobname/1.tex`

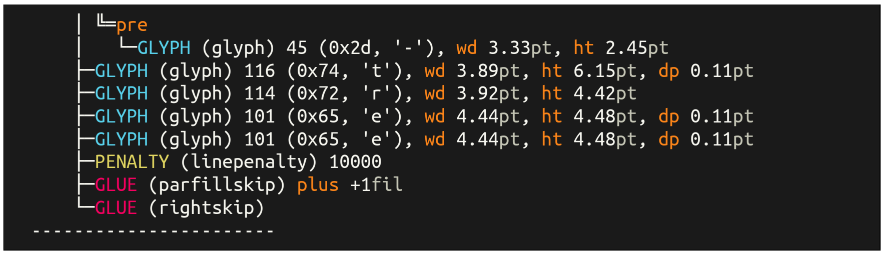
```
%!TEX program = lualatex
\documentclass{article}
\usepackage{nodetree}
\NodetreeSetOption[channel]{tex}
\NodetreeSetOption[verbosity]{1}
\NodetreeSetOption[unit]{pt}
\NodetreeSetOption[decimalplaces]{2}
\NodetreeUnregisterCallback{post_linebreak_filter}
\NodetreeRegisterCallback{post_linebreak_filter}
\begin{document}
nodetree
\end{document}
```

3. `_nodetree-jobname/1.nttex`: This temporary LuaLaTeXfile is compiled using `latexmk` and embedded in the environment `NodetreeEmbed` (the trailing \ character indicates line continuation).

```
Callback: \textcolor{NTEred}{post\_linebreak\_filter}\par
------------------------------------------\par
\mbox{ \textcolor{NTEmagentabright}{GLUE\hspace{0.5em}}\
  \textcolor{NTEyellow}{subtype:} baselineskip, \
  \textcolor{NTEyellow}{width:} 5.06\textcolor{NTEwhite}{pt}}\par
...
```

4. Finally the result:

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 5.06pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 6.94pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 110 (0x6e, 'n'), wd 5.56pt, ht 4.42pt
    ├─GLYPH (glyph) 111 (0x6f, 'o'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─KERN (fontkern) 0.28pt
    ├─GLYPH (glyph) 100 (0x64, 'd'), wd 5.56pt, ht 6.94pt, dp 0.11pt
    │  props {['injections'] = {['leftkern'] = 18350.08}}
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    ├─DISC (regular) penalty 50
```



```
      │  ╚═pre
      │    └─GLYPH (glyph) 45 (0x2d, '-'), wd 3.33pt, ht 2.45pt
      ├─GLYPH (glyph) 116 (0x74, 't'), wd 3.89pt, ht 6.15pt, dp 0.11pt
      ├─GLYPH (glyph) 114 (0x72, 'r'), wd 3.92pt, ht 4.42pt
      ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
      ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
      ├─PENALTY (linepenalty) 10000
      ├─GLUE (parfillskip) plus +1fil
      └─GLUE (rightskip)
  -----------------------
```

2.4.1 Available macros

Macro name	Reference
`\NodetreeRegisterCallback{`⟨*callbacks*⟩`}`	Page 10, Section 3.1
`\NodetreeUnregisterCallback{`⟨*callbacks*⟩`}`	Page 10, Section 3.2
`\NodetreeSetOption[`⟨*option*⟩`]{`⟨*value*⟩`}`	Page 10, Section 3.3
`\NodetreeResetOption{`⟨*option*⟩`}`	Page 10, Section 3.4
`\NodetreeReset`	Page 10, Section 3.6
`\NodetreeSet{`⟨*kv-options*⟩`}`	Page 10, Section 3.5
`\NodetreeEmbedCmd[`⟨*kv-options*⟩`]{`⟨*tex-markup*⟩`}`	Page 10, Section 3.7
`\NodetreeEmbedInput[`⟨*kv-options*⟩`]{`⟨*nttex-file*⟩`}`	Page 10, Section 3.8

2.4.2 Available environment

Environment name	Reference
`\begin{NodetreeEmbedEnv}[`⟨*kv-options*⟩`]`	Page 11, Section 4.1

2.4.3 Available options

Option name	Reference
`callback`	Page 12, Section 5.1
`channel`	Page 12, Section 5.2
`verbosity`	Page 13, Section 5.3
`color`	Page 13, Section 5.4
`unit`	Page 13, Section 5.5
`decimalplaces`	Page 15, Section 5.6
`theme`	Page 16, Section 5.7
`thememode`	Page 16, Section 5.7
`font`	Page 16, Section 5.8
`fontsize`	Page 17, Section 5.9

3 Macros

3.1 \NodetreeRegisterCallback

\NodetreeRegisterCallback \NodetreeRegisterCallback{⟨*callbacks*⟩}: The argument {⟨*callbacks*⟩} takes a comma separated list of callback aliases as described in (→ 5.1).

3.2 \NodetreeUnregisterCallback

\NodetreeUnregisterCallback \NodetreeUnregisterCallback{⟨*callbacks*⟩}: The argument {⟨*callbacks*⟩} takes a comma separated list of callback aliases as described in (→ 5.1).

3.3 \NodetreeSetOption

\NodetreeSetOption \NodetreeSetOption[⟨*option*⟩]{⟨*value*⟩}: (→ 5) This macro sets a single [⟨*option*⟩] to {⟨*value*⟩}.

3.4 \NodetreeResetOption

\NodetreeResetOption \NodetreeResetOption{⟨*option*⟩}: (→ 5) This macro resets a single {⟨*option*⟩} to its default value.

3.5 \NodetreeSet

\NodetreeSet \NodetreeSet{⟨*kv-options*⟩}: This macro sets multiple options a once. It only can be used along with LuaLaTeX. {⟨*kv-options*⟩} are key value pairs.

```
\NodetreeSet{color=no,callbacks={hpack,vpack},verbosity=2}
```

3.6 \NodetreeReset

\NodetreeReset \NodetreeReset: This macro resets multiple options to its default values.

3.7 \NodetreeEmbedCmd

\NodetreeEmbedCmd \NodetreeEmbedCmd[⟨*kv-options*⟩]{⟨*tex-markup*⟩}:

Main macro (cmd) to evaluate some TeX markup and generate a node tree from it. See environment version. (→ 3.7). Uses the xparse +v option to grab the verbatim content. Only available in the package nodetree-embed.

3.8 \NodetreeEmbedInput

\NodetreeEmbedInput \NodetreeEmbedInput[⟨*kv-options*⟩]{⟨*nttex-file*⟩}: The path or filename of *.nt-tex file without the extension. Only available in the package nodetree-embed.

4 Environments

4.1 `NodetreeEmbedEnv`

`NodetreeEmbedEnv` (*env.*) `\begin{NodetreeEmbedEnv}[`⟨*kv-options*⟩`]` … *TEX markup for evaluation* … `\end{NodetreeEmbedEnv}`

Main environment (env) to evaluate some TEX markup and generate a node tree from it. See command version (→ 3.7). Uses the `\detokenize` command to grab the verbatim content. Only available in the package nodetree-embed.

The callbacks are listed in the same order as in the LuaTEXreference manual.

Callback	Alias	Alias (longer)
`contribute_filter`	`contribute`	`contributefilter`
`buildpage_filter`	`buildfilter`	`buildpagefilter`
`build_page_insert`	`buildinsert`	`buildpageinsert`
`pre_linebreak_filter`	`preline`	`prelinebreakfilter`
`linebreak_filter`	`line`	`linebreakfilter`
`append_to_vlist_filter`	`append`	`appendtovlistfilter`
`post_linebreak_filter`	`postline`	`postlinebreakfilter`
`hpack_filter`	`hpack`	`hpackfilter`
`vpack_filter`	`vpack`	`vpackfilter`
`hpack_quality`	`hpackq`	`hpackquality`
`vpack_quality`	`vpackq`	`vpackquality`
`process_rule`	`process`	`processrule`
`pre_output_filter`	`preout`	`preoutputfilter`
`hyphenate`	`hyph`	
`ligaturing`	`liga`	
`kerning`	`kern`	
`insert_local_par`	`insert`	`insertlocalpar`
`mlist_to_hlist`	`mhlist`	`mlisttohlist`

Figure 1: The callback aliases

5 Options

5.1 Option `callback`

The option `callback` is the most important setting of the package. It is possible to specify an alias to select the `callback`. Take a look at the overview of callbacks (→ Figure 1). `nodetree` supports all node related callbacks as listed in the LuaTEX reference manual.

This macros process callback options: `\NodetreeRegisterCallback{`⟨*callbacks*⟩`}`, `\NodetreeUnregisterCallback{`⟨*callbacks*⟩`}`, `\NodetreeSet{`⟨*callback=<callbacks>*⟩`}` and `\usepackage[`⟨*callback=<callbacks>*⟩`]{`⟨*nodetree*⟩`}`.

Use commas to specify mulitple callbacks. Avoid using whitespaces:

```
\NodetreeRegisterCallback{preline,line,postline}
```

Wrap your callback aliases in curly braces for the macro `\NodetreeSet`:

```
\NodetreeSet{callback={preline,line,postline}}
```

The same applies for the macro `\usepackage`:

```
\usepackage{callback={preline,line,postline}}
```

5.2 Option `channel`

You can select the debug output channel with this option. The default value for the option `channel` is `term` which displays the node tree in the current terminal.

Specify `log` and the package creates a log file named `jobname.ntlog`. Specify `tex` and a log file named `jobname.nttex` is creatred. `nt...` stands for `nodetree`. `jobname` is the basename of your file you want to debug. The debug channel is only useful for the auxiliary package `nodetree-embed`. Paste the markup in the environment `NodetreeEmbedView` and you get a terminal like view in your document.

5.3 Option `verbosity`

Higher integer values result in a more verbose output. The default value for this options is `1`. At the moment only verbosity levels `0` to `3` are implemented.

5.3.1 Example: `verbosity=0`

```
Callback: pre_linebreak_filter
------------------------------------------
├─LOCAL_PAR
├─HLIST (indent) wd 15pt
├─GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
├─PENALTY (linepenalty) 10000
└─GLUE (parfillskip) plus +1fil
-----------------------
```

5.3.2 Example: `verbosity=1`

```
Callback: pre_linebreak_filter
------------------------------------------
├─LOCAL_PAR
├─HLIST subtype: indent, width: 15pt
├─GLYPH subtype: glyph, char: 46 (0x2e, '.'), width: 2.78pt, height: 1.06pt
├─PENALTY subtype: linepenalty, penalty: 10000
└─GLUE subtype: parfillskip, stretch: +1fil
-----------------------
```

5.3.3 Example: `verbosity=2`

```
Callback: pre_linebreak_filter
------------------------------------------
├─LOCAL_PAR[9] no: 464
├─HLIST[0] no: 417, subtype: indent[3], width: 15pt
├─GLYPH[29] no: 398, subtype: glyph[256], char: 46 (0x2e, '.'), font: 15, width: 2.78pt, height: 1.06pt
├─PENALTY[14] no: 159, subtype: linepenalty[2], penalty: 10000
└─GLUE[12] no: 470, subtype: parfillskip[15], stretch: +1fil
-----------------------
```

5.3.4 Example: `verbosity=3`

```
Callback: pre_linebreak_filter
------------------------------------------
├─LOCAL_PAR[9] no: 464
├─HLIST[0] no: 417, subtype: indent[3], width: 15pt, prev: 464
├─GLYPH[29] no: 398, subtype: glyph[256], char: 46 (0x2e, '.'), font: 15, width: 2.78pt, height: 1.06pt, prev: 417
├─PENALTY[14] no: 159, subtype: linepenalty[2], penalty: 10000, prev: 398
└─GLUE[12] no: 470, subtype: parfillskip[15], stretch: +1fil, prev: 159
-----------------------
```

5.4 Option `color`

The default option for `color` is `colored`. Use any other string (for example `none` or `no`) to disable the colored terminal output of the package.

```
\usepackage[color=no]{nodetree}
```

5.5 Option `unit`

The option `unit` sets the length unit to display all length values of the nodes. The default option for `unit` is `pt`. See figure 2 and 3 for possible values.

Unit	Description
pt	Point 1/72.27 inch. The conversion to metric units, to two decimal places, is 1 point = 2.85 mm = 28.45 cm.
pc	Pica, 12 pt
in	Inch, 72.27 pt
bp	Big point, 1/72 inch. This length is the definition of a point in PostScript and many desktop publishing systems.
cm	Centimeter
mm	Millimeter
dd	Didot point, 1.07 pt
cc	Cicero, 12 dd
sp	Scaled point, 1/65536 pt

Figure 2: Fixed units

Unit	Description
ex	x-height of the current font
em	Width of the capital letter M

Figure 3: Relative units

5.5.1 Example: unit=pt

```
Callback: pre_linebreak_filter
------------------------------------------
├─LOCAL_PAR
├─HLIST subtype: indent, width: 15pt
├─GLYPH subtype: glyph, char: 76 (0x4c, 'L'), width: 6.25pt, height: 6.83pt
├─GLYPH subtype: glyph, char: 111 (0x6f, 'o'), width: 5pt, height: 4.48pt, depth: 0.11pt
├─GLYPH subtype: glyph, char: 114 (0x72, 'r'), width: 3.92pt, height: 4.42pt
├─GLYPH subtype: glyph, char: 101 (0x65, 'e'), width: 4.44pt, height: 4.48pt, depth: 0.11pt
├─GLYPH subtype: glyph, char: 109 (0x6d, 'm'), width: 8.33pt, height: 4.42pt
├─GLYPH subtype: glyph, char: 46 (0x2e, '.'), width: 2.78pt, height: 1.06pt
├─PENALTY subtype: linepenalty, penalty: 10000
└─GLUE subtype: parfillskip, stretch: +1fil
-----------------------
```

5.5.2 Example: unit=sp

```
Callback: pre_linebreak_filter
------------------------------------------
├─LOCAL_PAR
├─HLIST subtype: indent, width: 983040sp
├─GLYPH subtype: glyph, char: 76 (0x4c, 'L'), width: 409600sp, height: 447611sp
├─GLYPH subtype: glyph, char: 111 (0x6f, 'o'), width: 327680sp, height: 293601sp, depth: 7209sp
├─GLYPH subtype: glyph, char: 114 (0x72, 'r'), width: 256901sp, height: 289669sp
├─GLYPH subtype: glyph, char: 101 (0x65, 'e'), width: 290980sp, height: 293601sp, depth: 7209sp
├─GLYPH subtype: glyph, char: 109 (0x6d, 'm'), width: 545915sp, height: 289669sp
├─GLYPH subtype: glyph, char: 46 (0x2e, '.'), width: 182190sp, height: 69468sp
├─PENALTY subtype: linepenalty, penalty: 10000
└─GLUE subtype: parfillskip, stretch: +1fil
-----------------------
```

5.5.3 Example: unit=cm

```
Callback: pre_linebreak_filter
------------------------------------------
```

```
├─LOCAL_PAR
├─HLIST subtype: indent, width: 0.53cm
├─GLYPH subtype: glyph, char: 76 (0x4c, 'L'), width: 0.22cm, height: 0.24cm
├─GLYPH subtype: glyph, char: 111 (0x6f, 'o'), width: 0.18cm, height: 0.16cm, depth: 0cm
├─GLYPH subtype: glyph, char: 114 (0x72, 'r'), width: 0.14cm, height: 0.16cm
├─GLYPH subtype: glyph, char: 101 (0x65, 'e'), width: 0.16cm, height: 0.16cm, depth: 0cm
├─GLYPH subtype: glyph, char: 109 (0x6d, 'm'), width: 0.29cm, height: 0.16cm
├─GLYPH subtype: glyph, char: 46 (0x2e, '.'), width: 0.1cm, height: 0.04cm
├─PENALTY subtype: linepenalty, penalty: 10000
└─GLUE subtype: parfillskip, stretch: +1fil
-----------------------
```

5.6 Option decimalplaces

The options `decimalplaces` sets the number of decimal places for some node fields. If `decimalplaces` is set to 0 only integer values are shown.

```
\NodetreeSetOption[decimalplaces]{4}
```

5.6.1 Example: decimalplaces=0

```
Callback: pre_linebreak_filter
------------------------------------------
├─LOCAL_PAR
├─HLIST subtype: indent, width: 1cc
├─GLYPH subtype: glyph, char: 76 (0x4c, 'L'), width: 0cc, height: 1cc
├─GLYPH subtype: glyph, char: 111 (0x6f, 'o'), width: 0cc, height: 0cc, depth: 0cc
├─GLYPH subtype: glyph, char: 114 (0x72, 'r'), width: 0cc, height: 0cc
├─GLYPH subtype: glyph, char: 101 (0x65, 'e'), width: 0cc, height: 0cc, depth: 0cc
├─GLYPH subtype: glyph, char: 109 (0x6d, 'm'), width: 1cc, height: 0cc
├─GLYPH subtype: glyph, char: 46 (0x2e, '.'), width: 0cc, height: 0cc
├─PENALTY subtype: linepenalty, penalty: 10000
└─GLUE subtype: parfillskip, stretch: +1fil
-----------------------
```

5.6.2 Example: decimalplaces=2

```
Callback: pre_linebreak_filter
------------------------------------------
├─LOCAL_PAR
├─HLIST subtype: indent, width: 1.17cc
├─GLYPH subtype: glyph, char: 76 (0x4c, 'L'), width: 0.49cc, height: 0.53cc
├─GLYPH subtype: glyph, char: 111 (0x6f, 'o'), width: 0.39cc, height: 0.35cc, depth: 0.01cc
├─GLYPH subtype: glyph, char: 114 (0x72, 'r'), width: 0.31cc, height: 0.34cc
├─GLYPH subtype: glyph, char: 101 (0x65, 'e'), width: 0.35cc, height: 0.35cc, depth: 0.01cc
├─GLYPH subtype: glyph, char: 109 (0x6d, 'm'), width: 0.65cc, height: 0.34cc
├─GLYPH subtype: glyph, char: 46 (0x2e, '.'), width: 0.22cc, height: 0.08cc
├─PENALTY subtype: linepenalty, penalty: 10000
└─GLUE subtype: parfillskip, stretch: +1fil
-----------------------
```

5.6.3 Example: decimalplaces=5

```
Callback: pre_linebreak_filter
------------------------------------------
├─LOCAL_PAR
├─HLIST subtype: indent, width: 1.16821cc
├─GLYPH subtype: glyph, char: 76 (0x4c, 'L'), width: 0.48676cc, height: 0.53193cc
├─GLYPH subtype: glyph, char: 111 (0x6f, 'o'), width: 0.3894cc, height: 0.34891cc, depth: 0.00857cc
├─GLYPH subtype: glyph, char: 114 (0x72, 'r'), width: 0.30529cc, height: 0.34423cc
├─GLYPH subtype: glyph, char: 101 (0x65, 'e'), width: 0.34579cc, height: 0.34891cc, depth: 0.00857cc
```

```
├GLYPH subtype: glyph, char: 109 (0x6d, 'm'), width: 0.64875cc, height: 0.34423cc
├GLYPH subtype: glyph, char: 46 (0x2e, '.'), width: 0.21651cc, height: 0.08255cc
├PENALTY subtype: linepenalty, penalty: 10000
└GLUE subtype: parfillskip, stretch: +1fil
----------------------
```

5.7 Option theme and thememode

5.7.1 Example: theme=bwdark thememode=dark

```
Callback: pre_linebreak_filter
------------------------------------------
├LOCAL_PAR
├HLIST (indent) wd 15pt
├GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
├PENALTY (linepenalty) 10000
└GLUE (parfillskip) plus +1fil
-----------------------
```

5.7.2 Example: theme=bwlight thememode=light

```
Callback: pre_linebreak_filter
------------------------------------------
├LOCAL_PAR
├HLIST (indent) wd 15
├GLYPH (glyph) 46 (0x2e, '.'), wd 2.78  , ht 1.06
├PENALTY (linepenalty) 10000
└GLUE (parfillskip) plus +1
-----------------------
```

5.7.3 Example: theme=monokaisoda thememode=dark

```
Callback: pre_linebreak_filter
------------------------------------------
├LOCAL_PAR
├HLIST (indent) wd 15pt
├GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
├PENALTY (linepenalty) 10000
└GLUE (parfillskip) plus +1fil
-----------------------
```

5.7.4 Example: theme=monokaisoda thememode=light

```
Callback: pre_linebreak_filter
------------------------------------------
├LOCAL_PAR
├HLIST (indent) wd 15pt
├GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
├PENALTY (linepenalty) 10000
└GLUE (parfillskip) plus +1fil
-----------------------
```

5.8 Option font

nodetree-embed passes the option font down to the command \setmonofont{} of the fontspec package. The used font should be a monospaced and have some box drawing glyphs (See table UNICODE glyphs 4).

5.8.1 Example: font={Liberation Mono}

```
Callback: post_linebreak_filter
-----------------------------------------
├─GLUE (baselineskip) wd 10.94pt
└─HLIST (line) wd 345pt, ht 1.06pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

5.8.2 Example: font={Ubuntu Mono}

```
Callback: post_linebreak_filter
-----------------------------------------
├─GLUE (baselineskip) wd 10.94pt
└─HLIST (line) wd 345pt, ht 1.06pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

5.9 Option fontsize

5.9.1 Example: \small

```
Callback: pre_linebreak_filter
-----------------------------------------
├─LOCAL_PAR
├─HLIST (indent) wd 15pt
├─GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
├─PENALTY (linepenalty) 10000
└─GLUE (parfillskip) plus +1fil
-----------------------
```

5.9.2 Example: \tiny

```
Callback: pre_linebreak_filter
-----------------------------------------
├─LOCAL_PAR
├─HLIST (indent) wd 15pt
├─GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
├─PENALTY (linepenalty) 10000
└─GLUE (parfillskip) plus +1fil
-----------------------
```

Code	Character	Name
U+2500	─	BOX DRAWINGS LIGHT HORIZONTAL
U+2502	│	BOX DRAWINGS LIGHT VERTICAL
U+2514	└	BOX DRAWINGS LIGHT UP AND RIGHT
U+251C	├	BOX DRAWINGS LIGHT VERTICAL AND RIGHT
U+2550	═	BOX DRAWINGS DOUBLE HORIZONTAL
U+2551	║	BOX DRAWINGS DOUBLE VERTICAL
U+255A	╚	BOX DRAWINGS DOUBLE UP AND RIGHT
U+2560	╠	BOX DRAWINGS DOUBLE VERTICAL AND RIGHT

Figure 4: The UNICODE box drawings glyphs

6 Visual tree structure

6.1 Two different connections

Nodes in LuaTEX are connected. The `nodetree` package distinguishs between the `list` and `field` connections.

- `list`: Nodes, which are double connected by `next` and `previous` fields.
- `field`: Connections to nodes by other fields than `next` and `previous` fields, e. g. `head`, `pre`.

6.2 Unicode characters to show the tree view

The package `nodetree` uses the unicode box drawing symbols. Your default terminal font should contain this characters to obtain the tree view. Eight box drawing characters are necessary.
For `list` connections *light* characters are shown.

```
│ │
│ ├─list1
│ └─list2
└─list3
```

`field` connections are visialized by *Double* characters.

```
║ ║
║ ╠═field1
║ ╚═field2
╚═field3
```

7 Examples

In this section lists some examples of the `nodetree` output.

7.1 The node list of the package name

`nodetree`

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 5.06pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 6.94pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 110 (0x6e, 'n'), wd 5.56pt, ht 4.42pt
    ├─GLYPH (glyph) 111 (0x6f, 'o'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─KERN (fontkern) 0.28pt
    ├─GLYPH (glyph) 100 (0x64, 'd'), wd 5.56pt, ht 6.94pt, dp 0.11pt
    │  props {['injections'] = {['leftkern'] = 18350.08}}
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    ├─DISC (regular) penalty 50
    │ ╚═pre
    │   └─GLYPH (glyph) 45 (0x2d, '-'), wd 3.33pt, ht 2.45pt
    ├─GLYPH (glyph) 116 (0x74, 't'), wd 3.89pt, ht 6.15pt, dp 0.11pt
    ├─GLYPH (glyph) 114 (0x72, 'r'), wd 3.92pt, ht 4.42pt
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.2 The node list of a mathematical formula

`$1+2$`

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 5.56pt
└─HLIST (line) wd 345pt, dp 0.83pt, ht 6.44pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─MATH (beginmath)
    ├─GLYPH (glyph) 49 (0x31, '1'), wd 5pt, ht 6.44pt
    ├─GLUE (medmuskip) wd 2.22pt, plus 1.11pt, minus 2.22pt
    ├─GLYPH (glyph) 43 (0x2b, '+'), wd 7.78pt, ht 5.83pt, dp 0.83pt
    ├─PENALTY (noadpenalty) 700
    ├─GLUE (medmuskip) wd 2.22pt, plus 1.11pt, minus 2.22pt
    ├─GLYPH (glyph) 50 (0x32, '2'), wd 5pt, ht 6.44pt
    ├─MATH (endmath)
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.3 The node list of the word *Office*

The characters *ffi* are deeply nested in a discretionary node.

`Office`

```
  Callback: post_linebreak_filter
  ------------------------------------------
  ├─GLUE (baselineskip) wd 4.95pt
  └─HLIST (line) wd 345pt, dp 0.22pt, ht 7.05pt
    ╚═head
      ├─LOCAL_PAR
      ├─HLIST (indent) wd 15pt
      ├─GLYPH (glyph) 79 (0x4f, 'O'), wd 7.78pt, ht 7.05pt, dp 0.22pt
      ├─DISC (regular) penalty 50
      │ ╠═post
      │ ║ └─GLYPH (glyph ligature) 64257 (0xfb01, 'ﬁ'), wd 5.56pt, ht 7.05pt
      │ ║   ╚═components
      │ ║     ├─GLYPH (glyph) 102 (0x66, 'f'), wd 3.06pt, ht 7.05pt
      │ ║     └─GLYPH (glyph) 105 (0x69, 'i'), wd 2.78pt, ht 6.57pt
      │ ╠═pre
      │ ║ ├─GLYPH (glyph) 102 (0x66, 'f'), wd 3.06pt, ht 7.05pt
      │ ║ └─GLYPH (glyph) 45 (0x2d, '-'), wd 3.33pt, ht 2.45pt
      │ ╚═replace
      │   └─GLYPH (glyph ligature) 64259 (0xfb03, 'ﬃ'), wd 8.33pt, ht 7.05pt
      │     ╚═components
      │       ├─GLYPH (glyph ligature) 64256 (0xfb00, 'ﬀ'), wd 5.83pt, ht 7.05pt
      │       │ ╚═components
      │       │   ├─GLYPH (glyph) 102 (0x66, 'f'), wd 3.06pt, ht 7.05pt
      │       │   └─GLYPH (glyph) 102 (0x66, 'f'), wd 3.06pt, ht 7.05pt
      │       └─GLYPH (glyph) 105 (0x69, 'i'), wd 2.78pt, ht 6.57pt
      ├─GLYPH (glyph) 99 (0x63, 'c'), wd 4.44pt, ht 4.48pt, dp 0.11pt
      ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
      ├─PENALTY (linepenalty) 10000
      ├─GLUE (parfillskip) plus +1fil
      └─GLUE (rightskip)
  -----------------------
```

7.4 Node types

This chapter shows some node types in a `nodetree` view.

7.4.1 Type: `hlist(0)` Subtype: `line(1)`

`Lorem`

```
  Callback: post_linebreak_filter
  ------------------------------------------
  ├─GLUE (baselineskip) wd 5.17pt
  └─HLIST (line) wd 345pt, dp 0.11pt, ht 6.83pt
    ╚═head
      ├─LOCAL_PAR
      ├─HLIST (indent) wd 15pt
      ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 6.25pt, ht 6.83pt
      ├─GLYPH (glyph) 111 (0x6f, 'o'), wd 5pt, ht 4.48pt, dp 0.11pt
      ├─GLYPH (glyph) 114 (0x72, 'r'), wd 3.92pt, ht 4.42pt
      ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
      ├─GLYPH (glyph) 109 (0x6d, 'm'), wd 8.33pt, ht 4.42pt
      ├─PENALTY (linepenalty) 10000
      ├─GLUE (parfillskip) plus +1fil
      └─GLUE (rightskip)
  -----------------------
```

7.4.2 Type: `hlist(0)` Subtype: `box(2)`

`L\hbox to 40pt{ore}m`

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 5.17pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 6.83pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 6.25pt, ht 6.83pt
    ├─HLIST (box) wd 40pt, dp 0.11pt, ht 4.48pt
    │ ╚═head
    │   ├─GLYPH (glyph) 111 (0x6f, 'o'), wd 5pt, ht 4.48pt, dp 0.11pt
    │   ├─GLYPH (glyph) 114 (0x72, 'r'), wd 3.92pt, ht 4.42pt
    │   └─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    ├─GLYPH (glyph) 109 (0x6d, 'm'), wd 8.33pt, ht 4.42pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.3 Type: `hlist(0)` Subtype: `indent(3)`

`\setlength {\parindent }{5cm} I`

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 0.18cm
└─HLIST (line) wd 12.13cm, ht 0.24cm
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 5cm
    ├─GLYPH (glyph) 73 (0x49, 'I'), wd 0.13cm, ht 0.24cm
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.4 Type: `vlist(1)`

`L\vbox to 40pt{O}L`

```
Callback: post_linebreak_filter
- groupcode: vbox
------------------------------------------
└─HLIST (line) wd 12.13cm, dp 0.01cm, ht 0.25cm
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 0.53cm
    ├─GLYPH (glyph) 79 (0x4f, 'O'), wd 0.27cm, ht 0.25cm, dp 0.01cm
    ├─PENALTY (linepenalty) 10000
  ├─GLUE (parfillskip) plus +1fil
  └─GLUE (rightskip)
-----------------------
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (lineskip) wd 0.04cm
└─HLIST (line) wd 12.13cm, dp 0.01cm, ht 1.41cm
 ╚═head
  ├─LOCAL_PAR
  ├─HLIST (indent) wd 0.53cm
  ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 0.22cm, ht 0.24cm
  ├─VLIST (unknown) wd 12.13cm, dp 0.01cm, ht 1.41cm
  │ ╚═head
  │  └─HLIST (line) wd 12.13cm, dp 0.01cm, ht 0.25cm
  │   ╚═head
  │    ├─LOCAL_PAR
  │    ├─HLIST (indent) wd 0.53cm
  │    ├─GLYPH (glyph) 79 (0x4f, 'O'), wd 0.27cm, ht 0.25cm, dp 0.01cm
  │    ├─PENALTY (linepenalty) 10000
  │    ├─GLUE (parfillskip) plus +1fil
  │    └─GLUE (rightskip)
  ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 0.22cm, ht 0.24cm
  ├─PENALTY (linepenalty) 10000
  ├─GLUE (parfillskip) plus +1fil
  └─GLUE (rightskip)
-----------------------
```

7.4.5 Type: rule(2)

`\rule [-2mm]{10mm}{4mm}`

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 2.22mm
└─HLIST (line) wd 121.25mm, dp 2mm, ht 2mm
 ╚═head
  ├─LOCAL_PAR
  ├─HLIST (indent) wd 5.27mm
  ├─HLIST (box) wd 10mm, dp 2mm, ht 2mm
  │ ╚═head
  │  └─RULE (normal) wd 10mm, dp 2mm, ht 2mm
  ├─PENALTY (linepenalty) 10000
  ├─GLUE (parfillskip) plus +1fil
  └─GLUE (rightskip)
-----------------------
```

7.4.6 Type: mark(4)

`\mark {Lorem}.`

```
Callback: pre_output_filter
- direction: TLT
- groupcode: output
- packtype: exactly
- maxdepth: 327680
- size: 36044800
------------------------------------------
├─WHATSIT (write) stream 129, data
```

```
├─MARK  table: 0x82db3a0
├─GLUE (topskip) wd 3.14mm
├─HLIST (line) wd 121.25mm, ht 0.37mm
│ ╚═head
│   ├─LOCAL_PAR
│   ├─HLIST (indent) wd 5.27mm
│   ├─GLYPH (glyph) 46 (0x2e, '.'), wd 0.98mm, ht 0.37mm
│   ├─PENALTY (linepenalty) 10000
│   ├─GLUE (parfillskip) plus +1fil
│   └─GLUE (rightskip)
└─GLUE (userskip) plus +1fil
-----------------------
Callback: pre_output_filter
- direction: TLT
- groupcode: output
- packtype: exactly
- maxdepth: 327680
- size: 36044800
------------------------------------------
├─WHATSIT (write) stream 129, data
├─GLUE (topskip) wd 3.51mm
└─VLIST (unknown)
-----------------------
Callback: pre_output_filter
- direction: TLT
- groupcode: output
- packtype: exactly
- maxdepth: 327680
- size: 36044800
------------------------------------------
├─WHATSIT (write) stream 129, data
├─GLUE (topskip) wd 3.51mm
├─HLIST (unknown) wd 121.25mm
└─GLUE (userskip) plus +1fill
-----------------------
```

7.4.7 Type: disc(7) Subtype: discretionary(0)

`L\-O\-L`

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 4.95pt
└─HLIST (line) wd 345pt, dp 0.22pt, ht 7.05pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 6.25pt, ht 6.83pt
    ├─DISC (explicit) penalty 50
    │ ╚═pre
    │   └─GLYPH (glyph) 45 (0x2d, '-'), wd 3.33pt, ht 2.45pt
    ├─GLYPH (glyph) 79 (0x4f, 'O'), wd 7.78pt, ht 7.05pt, dp 0.22pt
    ├─DISC (explicit) penalty 50
    │ ╚═pre
    │   └─GLYPH (glyph) 45 (0x2d, '-'), wd 3.33pt, ht 2.45pt
    ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 6.25pt, ht 6.83pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.8 Type: disc(7) Subtype: regular(3)

Office

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 4.95pt
└─HLIST (line) wd 345pt, dp 0.22pt, ht 7.05pt
  ╚head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 79 (0x4f, 'O'), wd 7.78pt, ht 7.05pt, dp 0.22pt
    ├─DISC (regular) penalty 50
    │ ╠post
    │ ║ └─GLYPH (glyph ligature) 64257 (0xfb01, 'ﬁ'), wd 5.56pt, ht 7.05pt
    │ ║   ╚components
    │ ║     ├─GLYPH (glyph) 102 (0x66, 'f'), wd 3.06pt, ht 7.05pt
    │ ║     └─GLYPH (glyph) 105 (0x69, 'i'), wd 2.78pt, ht 6.57pt
    │ ╠pre
    │ ║ ├─GLYPH (glyph) 102 (0x66, 'f'), wd 3.06pt, ht 7.05pt
    │ ║ └─GLYPH (glyph) 45 (0x2d, '-'), wd 3.33pt, ht 2.45pt
    │ ╚replace
    │   └─GLYPH (glyph ligature) 64259 (0xfb03, 'ﬃ'), wd 8.33pt, ht 7.05pt
    │     ╚components
    │       ├─GLYPH (glyph ligature) 64256 (0xfb00, 'ﬀ'), wd 5.83pt, ht 7.05pt
    │       │ ╚components
    │       │   ├─GLYPH (glyph) 102 (0x66, 'f'), wd 3.06pt, ht 7.05pt
    │       │   └─GLYPH (glyph) 102 (0x66, 'f'), wd 3.06pt, ht 7.05pt
    │       └─GLYPH (glyph) 105 (0x69, 'i'), wd 2.78pt, ht 6.57pt
    ├─GLYPH (glyph) 99 (0x63, 'c'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.9 Type: whatsit(8) Subtype: pdfaction(22)

```
\usepackage{hyperref}
\begin{document}
\url{http:/luatex.org}
\end{document}
```

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 5.06pt
└─HLIST (line) wd 345pt, dp 2.29pt, ht 6.94pt
  ╚head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─WHATSIT (pdf_start_link) wd -16384pt, dp -16384pt, ht -16384pt, objnum 4, link_attr /Border[
    │ ╚action
    │   └─WHATSIT (pdf_action) action_type 3, file , data /Subtype/Link/A<</Type/Action/S/URI/URI(
    ├─MATH (beginmath)
    ├─GLYPH (glyph) 104 (0x68, 'h'), wd 5.25pt, ht 6.11pt
    ├─GLYPH (glyph) 116 (0x74, 't'), wd 5.25pt, ht 5.54pt, dp 0.06pt
    ├─GLYPH (glyph) 116 (0x74, 't'), wd 5.25pt, ht 5.54pt, dp 0.06pt
    ├─GLYPH (glyph) 112 (0x70, 'p'), wd 5.25pt, ht 4.37pt, dp 2.22pt
    ├─GLUE (thickmuskip)
    ├─GLYPH (glyph) 58 (0x3a, ':'), wd 5.25pt, ht 4.31pt
    ├─PENALTY (noadpenalty) 500
    ├─GLUE (thickmuskip)
    ├─GLYPH (glyph) 47 (0x2f, '/'), wd 5.25pt, ht 6.94pt, dp 0.83pt
    ├─GLYPH (glyph) 108 (0x6c, 'l'), wd 5.25pt, ht 6.11pt
    ├─GLYPH (glyph) 117 (0x75, 'u'), wd 5.25pt, ht 4.31pt, dp 0.06pt
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5.25pt, ht 4.4pt, dp 0.06pt
    ├─GLYPH (glyph) 116 (0x74, 't'), wd 5.25pt, ht 5.54pt, dp 0.06pt
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 5.25pt, ht 4.4pt, dp 0.06pt
    ├─GLYPH (glyph) 120 (0x78, 'x'), wd 5.25pt, ht 4.31pt
    ├─GLUE (medmuskip)
    ├─GLYPH (glyph) 46 (0x2e, '.'), wd 5.25pt, ht 1.25pt
    ├─PENALTY (noadpenalty) 700
    ├─GLUE (medmuskip)
    ├─GLYPH (glyph) 111 (0x6f, 'o'), wd 5.25pt, ht 4.4pt, dp 0.06pt
    ├─GLYPH (glyph) 114 (0x72, 'r'), wd 5.25pt, ht 4.37pt
    ├─GLYPH (glyph) 103 (0x67, 'g'), wd 5.25pt, ht 4.42pt, dp 2.29pt
    ├─MATH (endmath)
    ├─WHATSIT (pdf_end_link)
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.10 Type: whatsit(8) Subtype: pdfcolorstack(28)

```
\usepackage{color}
\begin{document}
Lo\textcolor{red}{re}m.
\end{document}
```

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 5.17pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 6.83pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 6.25pt, ht 6.83pt
    ├─GLYPH (glyph) 111 (0x6f, 'o'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─WHATSIT (pdf_colorstack) data 1 0 0 rg 1 0 0 RG
    ├─GLYPH (glyph) 114 (0x72, 'r'), wd 3.92pt, ht 4.42pt
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    ├─WHATSIT (pdf_colorstack) data
    ├─GLYPH (glyph) 109 (0x6d, 'm'), wd 8.33pt, ht 4.42pt
    ├─GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.11 Type: glue(12) Subtype: baselineskip(2)

`\baselineskip=5cm Lorem Lorem`

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 4.76cm
└─HLIST (line) wd 12.13cm, dp 0cm, ht 0.24cm
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 0.53cm
    ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 0.22cm, ht 0.24cm
    ├─GLYPH (glyph) 111 (0x6f, 'o'), wd 0.18cm, ht 0.16cm, dp 0cm
    ├─GLYPH (glyph) 114 (0x72, 'r'), wd 0.14cm, ht 0.16cm
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 0.16cm, ht 0.16cm, dp 0cm
    ├─GLYPH (glyph) 109 (0x6d, 'm'), wd 0.29cm, ht 0.16cm
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 4.76cm
└─HLIST (line) wd 12.13cm, dp 0cm, ht 0.24cm
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 0.53cm
    ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 0.22cm, ht 0.24cm
    ├─GLYPH (glyph) 111 (0x6f, 'o'), wd 0.18cm, ht 0.16cm, dp 0cm
    ├─GLYPH (glyph) 114 (0x72, 'r'), wd 0.14cm, ht 0.16cm
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 0.16cm, ht 0.16cm, dp 0cm
    ├─GLYPH (glyph) 109 (0x6d, 'm'), wd 0.29cm, ht 0.16cm
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.12 Type: glue(12) Subtype: parskip(3)

`\parskip=5cm Lorem Lorem`

```
Callback: pre_output_filter
- groupcode: output
- packtype: exactly
- size: 36044800
- direction: TLT
- maxdepth: 327680
------------------------------------------
├─WHATSIT (write) stream 129, data
├─GLUE (topskip) wd 3.17pt
├─HLIST (line) wd 345pt, dp 0.11pt, ht 6.83pt
│ ╚═head
│   ├─LOCAL_PAR
│   ├─HLIST (indent) wd 15pt
│   ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 6.25pt, ht 6.83pt
│   ├─GLYPH (glyph) 111 (0x6f, 'o'), wd 5pt, ht 4.48pt, dp 0.11pt
│   ├─GLYPH (glyph) 114 (0x72, 'r'), wd 3.92pt, ht 4.42pt
│   ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
│   ├─GLYPH (glyph) 109 (0x6d, 'm'), wd 8.33pt, ht 4.42pt
│   ├─PENALTY (linepenalty) 10000
│   ├─GLUE (parfillskip) plus +1fil
│   └─GLUE (rightskip)
├─GLUE (parskip) wd 142.26pt
├─GLUE (parskip)
├─GLUE (baselineskip) wd 5.06pt
├─HLIST (line) wd 345pt, dp 0.11pt, ht 6.83pt
│ ╚═head
│   ├─LOCAL_PAR
│   ├─HLIST (indent) wd 15pt
│   ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 6.25pt, ht 6.83pt
│   ├─GLYPH (glyph) 111 (0x6f, 'o'), wd 5pt, ht 4.48pt, dp 0.11pt
│   ├─GLYPH (glyph) 114 (0x72, 'r'), wd 3.92pt, ht 4.42pt
│   ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
│   ├─GLYPH (glyph) 109 (0x6d, 'm'), wd 8.33pt, ht 4.42pt
│   ├─PENALTY (linepenalty) 10000
│   ├─GLUE (parfillskip) plus +1fil
│   └─GLUE (rightskip)
├─GLUE (userskip) wd -0.11pt
└─GLUE (userskip) plus +1fil
-----------------------
Callback: pre_output_filter
- groupcode: output
- packtype: exactly
- size: 36044800
- direction: TLT
- maxdepth: 327680
------------------------------------------
├─WHATSIT (write) stream 129, data
├─GLUE (topskip) wd 10pt
└─VLIST (unknown)
-----------------------
Callback: pre_output_filter
- groupcode: output
- packtype: exactly
- size: 36044800
- direction: TLT
- maxdepth: 327680
------------------------------------------
├─WHATSIT (write) stream 129, data
├─GLUE (topskip) wd 10pt
├─HLIST (unknown) wd 345pt
└─GLUE (userskip) plus +1fill
-----------------------
```

7.4.13 Type: glue(12) Subtype: spaceskip(13)

```
\spaceskip =5cm a a
```

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 7.52pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 4.48pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─GLUE (spaceskip) wd 142.26pt
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.14 Type: glue(12) Subtype: leaders(100)

a \leavevmode \leaders \hbox { . }\hfill \kern 0pt a

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 7.52pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 4.48pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─GLUE (spaceskip) wd 3.33pt, plus 1.66pt, minus 1.11pt
    ├─GLUE (leaders) plus +1fill
    │ ╚═leader
    │   └─HLIST (box) wd 10.55pt, ht 1.06pt
    │     ╚═head
    │       ├─GLUE (spaceskip) wd 3.33pt, plus 1.66pt, minus 1.11pt
    │       ├─GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
    │       └─GLUE (spaceskip) wd 4.44pt, plus 4.99pt, minus 0.37pt
    ├─KERN (userkern)
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.15 Type: glue(12) Subtype: cleaders(101)

a \leavevmode \cleaders \hbox { . }\hfill \kern 0pt a

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 7.52pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 4.48pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─GLUE (spaceskip) wd 3.33pt, plus 1.66pt, minus 1.11pt
    ├─GLUE (cleaders) plus +1fill
    │ ╚═leader
    │   └─HLIST (box) wd 10.55pt, ht 1.06pt
    │     ╚═head
    │       ├─GLUE (spaceskip) wd 3.33pt, plus 1.66pt, minus 1.11pt
    │       ├─GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
    │       └─GLUE (spaceskip) wd 4.44pt, plus 4.99pt, minus 0.37pt
    ├─KERN (userkern)
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.16 Type: glue(12) Subtype: xleaders(102)

`a \leavevmode \xleaders \hbox { . }\hfill \kern 0pt a`

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 7.52pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 4.48pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─GLUE (spaceskip) wd 3.33pt, plus 1.66pt, minus 1.11pt
    ├─GLUE (xleaders) plus +1fill
    │ ╚═leader
    │   └─HLIST (box) wd 10.55pt, ht 1.06pt
    │     ╚═head
    │       ├─GLUE (spaceskip) wd 3.33pt, plus 1.66pt, minus 1.11pt
    │       ├─GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
    │       └─GLUE (spaceskip) wd 4.44pt, plus 4.99pt, minus 0.37pt
    ├─KERN (userkern)
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.17 Type: glue(12) Subtype: gleaders(102)

`a \leavevmode \gleaders \hbox { . }\hfill \kern 0pt a`

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 7.52pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 4.48pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─GLUE (spaceskip) wd 3.33pt, plus 1.66pt, minus 1.11pt
    ├─GLUE (gleaders) plus +1fill
    │ ╚═leader
    │   └─HLIST (box) wd 10.55pt, ht 1.06pt
    │     ╚═head
    │       ├─GLUE (spaceskip) wd 3.33pt, plus 1.66pt, minus 1.11pt
    │       ├─GLYPH (glyph) 46 (0x2e, '.'), wd 2.78pt, ht 1.06pt
    │       └─GLUE (spaceskip) wd 4.44pt, plus 4.99pt, minus 0.37pt
    ├─KERN (userkern)
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
```

```
-----------------------
```

7.4.18 Type: kern(13) Subtype: userkern(0)

a\kern 2pt

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 7.52pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 4.48pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─KERN (userkern) 2pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.19 Type: kern(13) Subtype: fontkern(1)

Ve

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 5.17pt
└─HLIST (line) wd 345pt, dp 0.22pt, ht 6.83pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 86 (0x56, 'V'), wd 7.5pt, ht 6.83pt, dp 0.22pt
    ├─KERN (fontkern) -0.83pt
    ├─GLYPH (glyph) 101 (0x65, 'e'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    │  props {['injections'] = {['leftkern'] = -54394.88}}
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.20 Type: kern(13) Subtype: accentkern(2)

\accent 96 a

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 4.85pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 7.15pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─KERN (accentkern)
    ├─HLIST (unknown) wd 5pt, ht 6.98pt, shift -0.17pt
    │ ╚═head
    │   └─GLYPH (glyph) 96 (0x60, '`'), wd 5pt, ht 6.98pt
    ├─KERN (accentkern) -5pt
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.21 Type: kern(13) Subtype: italiccorrection(3)

\textit {L}\/OL

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 4.95pt
└─HLIST (line) wd 345pt, dp 0.22pt, ht 7.05pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 6.27pt, ht 6.83pt
    ├─KERN (italiccorrection) 0.17pt
    ├─GLYPH (glyph) 79 (0x4f, 'O'), wd 7.78pt, ht 7.05pt, dp 0.22pt
    ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 6.25pt, ht 6.83pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.22 Type: penalty(14)

L \penalty 23 OL

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 4.95pt
└─HLIST (line) wd 345pt, dp 0.22pt, ht 7.05pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 6.25pt, ht 6.83pt
    ├─GLUE (spaceskip) wd 3.33pt, plus 1.66pt, minus 1.11pt
    ├─PENALTY (userpenalty) 23
    ├─GLYPH (glyph) 79 (0x4f, 'O'), wd 7.78pt, ht 7.05pt, dp 0.22pt
    ├─GLYPH (glyph) 76 (0x4c, 'L'), wd 6.25pt, ht 6.83pt
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.23 Type: glyph(29)

abc

```
Callback: post_linebreak_filter
------------------------------------------
├─GLUE (baselineskip) wd 5.06pt
└─HLIST (line) wd 345pt, dp 0.11pt, ht 6.94pt
  ╚═head
    ├─LOCAL_PAR
    ├─HLIST (indent) wd 15pt
    ├─GLYPH (glyph) 97 (0x61, 'a'), wd 5pt, ht 4.48pt, dp 0.11pt
    ├─GLYPH (glyph) 98 (0x62, 'b'), wd 5.56pt, ht 6.94pt, dp 0.11pt
    ├─KERN (fontkern) 0.28pt
    ├─GLYPH (glyph) 99 (0x63, 'c'), wd 4.44pt, ht 4.48pt, dp 0.11pt
    │  props {['injections'] = {['leftkern'] = 18350.08}}
    ├─PENALTY (linepenalty) 10000
    ├─GLUE (parfillskip) plus +1fil
    └─GLUE (rightskip)
-----------------------
```

7.4.24 Type: attributelist(40)

`{\attribute 0=1 A}`

```
Callback: hpack_filter
- size: 0
- groupcode: hbox
- attributelist: <node nil < 83 > 149 : attribute_list 0>
- packtype: additional
- direction: TLT
------------------------------------------
└─RULE (normal) dp 3.6pt, ht 8.4pt
-----------------------
Callback: hpack_filter
- size: 0
- groupcode: hbox
- attributelist: <node nil < 145 > 147 : attribute_list 0>
- packtype: additional
- direction: TLT
------------------------------------------
└─RULE (normal) dp 3.6pt, ht 8.4pt
-----------------------
Callback: hpack_filter
- size: 0
- groupcode: hbox
- attributelist: <node 12 < 513 > 594 : attribute_list 0>
- packtype: additional
- direction: TLT
------------------------------------------
└─RULE (normal) dp 3.6pt, ht 8.4pt
-----------------------
Callback: hpack_filter
- size: 0
- groupcode: adjusted_hbox
- attributelist: <node nil < 511 > 605 : attribute_list 0>
- packtype: additional
- direction: TLT
------------------------------------------
└─HLIST (box) wd 345pt
-----------------------
Callback: hpack_filter
- size: 22609920
- groupcode: hbox
- attributelist: <node nil < 684 > 691 : attribute_list 0>
- packtype: exactly
- direction: TLT
------------------------------------------
├─GLUE (userskip) plus +1fil
```

```
├─GLYPH (glyph) 49 (0x31, '1'), wd 5pt, ht 6.66pt
└─GLUE (userskip) plus +1fil
-----------------------
Callback: hpack_filter
- size: 0
- groupcode: adjusted_hbox
- attributelist: <node nil < 684 > 691 : attribute_list 0>
- packtype: additional
- direction: TLT
------------------------------------------
└─HLIST (box) wd 345pt, ht 6.66pt
  ╚═head
    ├─GLUE (userskip) plus +1fil
    ├─GLYPH (glyph) 49 (0x31, '1'), wd 5pt, ht 6.66pt
    └─GLUE (userskip) plus +1fil
-----------------------
```

8 Implementation

8.1 The file nodetree.tex

```
\directlua{
  nodetree = require('nodetree')
}
```

\NodetreeSetOption

```
\def\NodetreeSetOption[#1]#2{
  \directlua{
    nodetree.set_option('#1', '#2')
  }
}
\let\nodetreeoption\NodetreeSetOption
```

\NodetreeResetOption

```
\def\NodetreeResetOption#1{
  \NodetreeSetOption[#1]{%
    \directlua{
      tex.print(nodetree.get_default_option('#1'))
    }%
  }%
}
```

\NodetreeReset

```
\def\NodetreeReset{
  \NodetreeResetOption{callback}
  \NodetreeResetOption{channel}
  \NodetreeResetOption{color}
  \NodetreeResetOption{decimalplaces}
  \NodetreeResetOption{unit}
  \NodetreeResetOption{verbosity}
}
\let\nodetreereset\NodetreeReset
```

\NodetreeRegisterCallback

```
\def\NodetreeRegisterCallback#1{
  \directlua{
    nodetree.set_option('callback', '#1')
    nodetree.register_callbacks()
  }
}
\let\nodetreeregister\NodetreeRegisterCallback
```

\NodetreeUnregisterCallback

```
\def\NodetreeUnregisterCallback#1{
  \directlua{
    nodetree.set_option('callback', '#1')
    nodetree.unregister_callbacks()
  }
}
\let\nodetreeunregister\NodetreeUnregisterCallback
```

8.2 The file nodetree.sty

```
\input{nodetree}

\RequirePackage{kvoptions}

\SetupKeyvalOptions{
  family=NT,
  prefix=NTK@
}

\DeclareStringOption[term]{channel}
\define@key{NT}{channel}[]{\NodetreeSetOption[channel]{#1}}

\DeclareStringOption[postlinebreak]{callback}
\define@key{NT}{callback}[]{\NodetreeSetOption[callback]{#1}}

\DeclareStringOption[0]{verbosity}
\define@key{NT}{verbosity}[]{\NodetreeSetOption[verbosity]{#1}}

\DeclareStringOption[colored]{color}
\define@key{NT}{color}[]{\NodetreeSetOption[color]{#1}}

\DeclareStringOption[1]{unit}
\define@key{NT}{unit}[]{\NodetreeSetOption[unit]{#1}}

\DeclareStringOption[1]{decimalplaces}
\define@key{NT}{decimalplaces}[]{\NodetreeSetOption[decimalplaces]{#1}}

\ProcessKeyvalOptions{NT}
\directlua{
  nodetree.register_callbacks()
}
```

\NodetreeSet

```
\newcommand{\NodetreeSet}[1]{%
  \setkeys{NT}{#1}%
}
\let\nodetreeset\NodetreeSet

\NeedsTeXFormat{LaTeX2e}[1994/06/01]
\ProvidesPackage{nodetree-embed}
  [2022/12/17 v2.2.1 Embed node trees into a LaTeX document]

\RequirePackage{xcolor,mdframed,expl3,xparse,fontspec}

\input{nodetree}
```

```
\RequirePackage{kvoptions}
\SetupKeyvalOptions{
  family=NTE,
  prefix=NTEK@
}

\directlua{
  nodetree = require('nodetree')
  nodetree.check_shell_escape()
}

\define@key{NTE}{callback}[]{\NodetreeSetOption[callback]{#1}}

\DeclareStringOption[0]{verbosity}
\define@key{NTE}{verbosity}[]{\NodetreeSetOption[verbosity]{#1}}

\DeclareStringOption[colored]{color}
\define@key{NTE}{color}[]{\NodetreeSetOption[color]{#1}}

\DeclareStringOption[1]{unit}
\define@key{NTE}{unit}[]{\NodetreeSetOption[unit]{#1}}

\DeclareStringOption[1]{decimalplaces}
\define@key{NTE}{decimalplaces}[]{\NodetreeSetOption[decimalplaces]{#1}}

\DeclareStringOption[monokaisoda]{theme}

\DeclareStringOption[dark]{thememode}

\DeclareStringOption[Ubuntu Mono]{font}

\DeclareStringOption[\footnotesize]{fontsize}

\DeclareBoolOption{showmarkup}

\ProcessKeyvalOptions{NTE}
```

\NTE@colors

```
\ExplSyntaxOn
\def\NTE@colors{
  \str_case_e:nn{\NTEK@theme}{
    {bwdark}{
      \definecolor{NTEblack}{gray}{0}
      \definecolor{NTEred}{gray}{1}
      \definecolor{NTEgreen}{gray}{1}
      \definecolor{NTEyellow}{gray}{1}
      \definecolor{NTEblue}{gray}{1}
      \definecolor{NTEmagenta}{gray}{1}
      \definecolor{NTEcyan}{gray}{1}
      \definecolor{NTEwhite}{gray}{1}
      \definecolor{NTEblackbright}{gray}{0}
      \definecolor{NTEredbright}{gray}{1}
      \definecolor{NTEgreenbright}{gray}{1}
      \definecolor{NTEyellowbright}{gray}{1}
      \definecolor{NTEbluebright}{gray}{1}
      \definecolor{NTEmagentabright}{gray}{1}
      \definecolor{NTEcyanbright}{gray}{1}
      \definecolor{NTEwhitebright}{gray}{1}
    }
    {bwlight}{
      \definecolor{NTEblack}{gray}{0}
      \definecolor{NTEred}{gray}{0}
      \definecolor{NTEgreen}{gray}{0}
      \definecolor{NTEyellow}{gray}{0}
      \definecolor{NTEblue}{gray}{0}
      \definecolor{NTEmagenta}{gray}{0}
      \definecolor{NTEcyan}{gray}{0}
      \definecolor{NTEwhite}{gray}{1}
      \definecolor{NTEblackbright}{gray}{0}
      \definecolor{NTEredbright}{gray}{0}
      \definecolor{NTEgreenbright}{gray}{0}
      \definecolor{NTEyellowbright}{gray}{0}
      \definecolor{NTEbluebright}{gray}{0}
      \definecolor{NTEmagentabright}{gray}{0}
      \definecolor{NTEcyanbright}{gray}{0}
      \definecolor{NTEwhitebright}{gray}{1}
    }
    {monokaisoda}{
      \definecolor{NTEblack}{HTML}{1a1a1a}
      \definecolor{NTEred}{HTML}{f4005f}
      \definecolor{NTEgreen}{HTML}{98e024}
      \definecolor{NTEyellow}{HTML}{fa8419}
      \definecolor{NTEblue}{HTML}{9d65ff}
      \definecolor{NTEmagenta}{HTML}{f4005f}
      \definecolor{NTEcyan}{HTML}{58d1eb}
      \definecolor{NTEwhite}{HTML}{c4c5b5}
      \definecolor{NTEblackbright}{HTML}{625e4c}
      \definecolor{NTEredbright}{HTML}{f4005f}
      \definecolor{NTEgreenbright}{HTML}{98e024}
      \definecolor{NTEyellowbright}{HTML}{e0d561}
      \definecolor{NTEbluebright}{HTML}{9d65ff}
      \definecolor{NTEmagentabright}{HTML}{f4005f}
      \definecolor{NTEcyanbright}{HTML}{58d1eb}
      \definecolor{NTEwhitebright}{HTML}{f6f6ef}
    }
  }
  \str_case_e:nn{\NTEK@thememode}{
    {dark}{
      \definecolor{NTEbackground}{named}{NTEblack}
      \definecolor{NTEfont}{named}{NTEwhitebright}
    }
    {light}{
      \definecolor{NTEbackground}{named}{NTEwhitebright}
      \definecolor{NTEfont}{named}{NTEblack}
    }
  }
```

```
}
\ExplSyntaxOff
```

\NTE@fonts

```
\def\NTE@fonts{
  \bfseries%
  \NTEK@fontsize%
  \setmonofont{\NTEK@font}%
  \ttfamily%
  \setlength{\parindent}{0pt}%
  \setlength{\parskip}{-0.9pt}%
}
```

\NodetreeSet Same definition as in nodetree.sty. Only implement this command if not already registers.

```
\providecommand{\NodetreeSet}[1]{%
  \setkeys{NTE}{#1}%
}
```

```
\newenvironment{NodetreeEmbedView}[1][]{
  \setkeys{NTE}{#1}
  \NTE@colors
  \begin{mdframed}[
    linecolor=black,
    backgroundcolor=NTEbackground,
    fontcolor=NTEfont,
  ]%
  \NTE@fonts
}{
  \end{mdframed}%
}
```

NodetreeEmbedEnv *(env.)*

```
\NewDocumentEnvironment { NodetreeEmbedEnv } { O{} +b } {
  \setkeys{NTE}{#1}
  \ifNTEK@showmarkup
    \noindent
    \texttt{\detokenize{#2}}
  \else
  \fi
  \NTE@colors
  \begin{NodetreeEmbedView}
    \directlua{
      nodetree.compile_include('\luaescapestring{\unexpanded{#2}}')
    }
  \end{NodetreeEmbedView}
}{}
```

\NodetreeEmbedCmd

```
\NewDocumentCommand { \NodetreeEmbedCmd } { O{} +v } {
  \setkeys{NTE}{#1}
  \ifNTEK@showmarkup
    \noindent
    \texttt{#2}
  \else
  \fi
  \NTE@colors
  \begin{NodetreeEmbedView}
    \directlua{
      nodetree.compile_include('\luaescapestring{\unexpanded{#2}}')
    }
  \end{NodetreeEmbedView}
}
```

\NodetreeEmbedInput

```
\newcommand{\NodetreeEmbedInput}[2][]{
  \setkeys{NTE}{#1}
  \begin{NodetreeEmbedView}
  \input{#2.nttex}
  \end{NodetreeEmbedView}
}
\let\nodetreeterminalemulator\NodetreeEmbedInput
```

8.3 The file `nodetree.lua`

There is a source code documentation of the file nodetree.lua compiled with Ldoc on Github: http://josef-friedrich.github.io/nodetree/