familytree package v3.2

Jiro Senju
jiro1010senju AT gmail DOT com

2026/10/21

Draws a Family Tree.
Defines a box describing an individual, and connects the multiple boxes by lines. The line represents the sibling, the parent-child relation ship, or the marriage.

- Excluding the marriage box, you can get a maleline/patrilineal tree, or a femaleline/matrilineal tree.
- For Japanese, `jlreq.cls` vertical option (`tate`) is supported.

Contents

Introduction 1
Development 2
Structure 2

1 Individual Box (`ft-individual.dtx`)
1.1 Customization 4
1.2 Example 5
1.3 Implementation 5

2 Sibling Box (`ft-sibling.dtx`) **14**
2.1 Customization 14
2.2 Example 14
2.3 Implementation 15

3 Parent-child Relationship Box or Generations Box (`ft-gens.dtx`) **20**
3.1 Example 20
3.2 The order of connecting multiple boxes 21
3.3 Implementation 23

3

4 Marriage Box (`ft-marriage.dtx`) **27**
4.1 Customization 27
4.2 Example 28
4.3 Layout and connecting in a same generation 29
4.4 Implementation 32

5 Library (`ft-lib.dtx`) **36**
5.1 Implementation 36

Introduction

Family Tree is interesting. But also, it can be really complicated, especially including the siblings and marriages.

Graphviz (`dot(1)`) is a good tool to draw a family tree, but I want more straightforward understandability. Here I try developing some macros to draw a family tree easily. I am not a TeXnician, but I hope it helps someone who wants to draw and view a large family tree. Tested on TeX Live 2019. Any comments will be appreciated.

Development

Basically all control sequences have a prefix "ft". But non-prefix names are also defined by \let as an alias/synonym, to improve the usability and the visibility. Obviously only when the name is undefined. If CS name is already defined and familytree pkg cannot define the alias, \message is generated. This document tries using the ft-less alias name, but a few are used with the prefixed name.

There are multiple dtx files, but the generated sty is only one. ft-lib.dtx is described at last of this document, but it comes first in sty.

In the beginning, I was going to implement using \hbox and \vbox or tabular environment. I thought they would be enough. After defining the boxes, I would connect them by lines, then I got a trouble. latex adjusts the position of the boxes by inserting a glue or something, and their connection points are shifted. I could not find a good universal solution. Can TikZ or something define the absolute coordinates and the lines? I don't know. To connect the lines, I had to choose picture environment.

The depth of a character was another trouble, or I don't have enough experience and knowledge. To layout the boxes in picture, I have to consider the depth of the box. To support the Japanese in vertical mode, the depth is very important. In horizontal mode, the depth is very alike of the English alphabets. But in vertical mode, the depth is a half size of a character. I didn't know that, and it took very long time for me.

Structure

Defines every element as a box in the tree, and connects them by lines. In defining a box, we also define its connection points which make the box to be connectable later.

These are the elements.

1. individual box
    - child mark to represent one is adopted or biological child
    - one's title
    - one's name
    - maleline/femaleline for the patrilineal/matrilineal tree
    - additional information
      birth/death year-month-date, nickname, or anything
2. sibling box
    - a line between the child marks
    - interval box to make a space between individuals
3. marriage box
    - marriage line (double line) to connect the husband and the wife
4. parent-child relationship or generations box

1 Individual Box (ft-individual.dtx)

\indvdldef

\indvdldef[⟨*child mark*⟩] {⟨*new box name*⟩} [⟨*title*⟩] {⟨*individual name*⟩} [⟨*additional info*⟩] [⟨*maleline xlength*⟩]

Defines a new box named ⟨*new box name*⟩, and layouts the given arguments in that box.

1. [⟨*child mark*⟩]

   The type of a line which is located ahead of the individual name.

   Specify one of \ftbiological, \ftadopted, \fttop, or \ftblank. Details are discussed later.

2. {⟨*new box name*⟩}

   The name of a new box.

   This is not a control sequence (no backslash).

3. [⟨*title*⟩]

   One's title.

   Put at the shoulder of the individual name with a smaller font (\relsize{-2}).

4. {⟨*individual name*⟩}

   One's name.

5. [⟨*additional info*⟩]

   Additional information.

   Anything can be added such as birth/death year-month-date and alias. Put with a little smaller font (\relsize{-1}) at next to the individual name. Also the attributes (discussed later), \ftmaleline (\ftfemaleline, \ftpatrilineal, \ftmatrilineal), \fthaschild, \ftprivate are specified here.

6. [⟨*maleline xlength*⟩]

   When you draw a line from an individual (instead of a marriage box) to a child, the length of the line can be adjusted by this argument. But in the sibling box, the line length will be adjusted automatically regardless this argument.

Also \indvdldef defines some connection points as CS. The origin is left-bottom of the box and the unit is pt.

⟨*box name*⟩nameX	the tail of the individual name
⟨*box name*⟩nameCX	the center of the width of individual name
⟨*box name*⟩nameCY	the center of the height of individual name

For [⟨*child mark*⟩], you can specify one of these values.

- \biological or \ftbiological

  a single line to represent a biological/real child.

- \adopted or \ftadopted

  a double line to represent an adopted child.

- `\fttop` (or `\top`)

  no line at all.

- `\blank` or `\ftblank`

  put a space whose size is equal to the `\biological` or `\adopted` mark, assuming used for the spouse in a marriage box.

For `[`⟨*additional info*⟩`]`, you can specify any of these attributes.

- `\haschild` or `\fthaschild`

  represents a spouse who has a child. draws a line in a marriage box to their child from the center of the double line which represents a marriage.

- `\private` or `\ftprivate`

  represents the marriage is not official. draws a dashed double line.

- `\maleline` or `\ftmaleline`

  draws a line to a child from the individual name instead of the double line in the marriage box. `\ftfemaleline`, `\ftpatrilineal`, `\ftmatrilineal` are all equivalent.

The individual box consists of a few smaller boxes.

- `\ft@titlebox`
- `\ft@namebox`

  In `\ft@namebox`, there are `\ft@cmarkbox` which represents the ⟨*child mark*⟩ and `\ft@malelinebox` which represents ⟨*maleline*⟩.

- `\ft@optbox`

1.1 Customization

`\nameboxcfg` `\nameboxcfg{`⟨*space from the child mark*⟩`} {`⟨*font*⟩`} {`⟨*space to the maleline*⟩`} {`⟨*maleline length*⟩`}`

`\cmarkboxcfg` `\cmarkboxcfg{`⟨*space between two lines, for adopted*⟩`} {`⟨*line length*⟩`}`

`\titleboxcfg` `\titleboxcfg{`⟨*indent*⟩`} {`⟨*font*⟩`} {`⟨*linestretch*⟩`} {`⟨*vspace to the individual name*⟩`}`

`\optboxcfg` `\optboxcfg{`⟨*vspace from the individual name*⟩`} {`⟨*indent*⟩`} {`⟨*font*⟩`} {`⟨*linestretch*⟩`}`

1.2 Example

1.
```
\indvdldef[\fttop]{Robert}{Robert Crawley}[\maleline]
\fbox{\usebox{\Robert}}
```

Robert Crawley ——

\RobertnameX	71.86122
\RobertnameCX	34.68062
\RobertnameCY	2.5

2.
```
\indvdldef{Robert}[7th]{Robert Crawley}
            [7th Earl of Grantham,\maleline]
\fbox{\usebox{\Robert}}
```

— 7th Robert Crawley ——
7th Earl of Grantham

Jump to next section

1.3 Implementation

1.3.1 Child-mark box

Customization

\ftcmarkboxcfg
\cmarkboxcfg
```
\newlength{\ft@cmarkbox@adopted@sep}
\setlength{\ft@cmarkbox@adopted@sep}{4pt}
\newlength{\ft@cmarkbox@length}
\setlength{\ft@cmarkbox@length}{1\ft@unit}
\newcommand{\ftcmarkboxcfg}[2]{% adopted-sep line-length
  \ft@ifempty{#1}{}{%
    \setlength{\ft@cmarkbox@adopted@sep}{#1}%
  }%
  \ft@ifempty{#2}{}{%
    \setlength{\ft@cmarkbox@length}{#2}%
  }%
}
\ft@alias{cmarkboxcfg}
```

Define a box

\fttop
\top
\ftbiological
\biological
\ftadopted
\adopted
\ftblank
\blank
```
\chardef\fttop=0
\chardef\ftbiological=1
\chardef\ftadopted=2
\chardef\ftblank=3
\ft@alias{top}
\ft@alias{biological}
\ft@alias{adopted}
\ft@alias{blank}
```

```
\ft@indvdl@nameH
\ft@indvdl@nameD
\ft@indvdl@nameCY
  \ft@do@cmarkdef
     \ft@cmarkdef
```

```
\newsavebox{\ft@cmarkbox}
\newlength{\ft@cmarkW}
\newlength{\ft@indvdl@nameH}
\newlength{\ft@indvdl@nameD}
\newlength{\ft@indvdl@nameCY}
\newcommand{\ft@do@cmarkdef}[4]{% cmarkbox-name cmark width line-length
  \ft@savenamebox{#1}{%
    \edef\@w{\strip@pt#3}%
    \edef\@h{\strip@pt\ft@indvdl@nameH}%
    \edef\@l{\strip@pt#4}%
    \unitlength=1pt\relax% coordinates are in pt
    \begin{picture}(\@w,\@h)%
      \ft@dbgframe{\@w,\@h}%
      \ifcase#2% fttop
      \or% ftbiological
        \put(0,\strip@pt\ft@indvdl@nameCY){\line(1,0){\@l}}%
      \or% ftadopted
        \ft@y=\dimexpr\ft@indvdl@nameCY%
          + .5\ft@cmarkbox@adopted@sep\relax%
        \put(0,\strip@pt\ft@y){\line(1,0){\@l}}%
        \advance\ft@y -\ft@cmarkbox@adopted@sep%
        \put(0,\strip@pt\ft@y){\line(1,0){\@l}}%
      \or% \ftblank
      \fi%
    \end{picture}%
  }%
  \ft@nameboxsz{#1}{\ft@indvdl@nameH}{\ft@indvdl@nameD}%
}
\newcommand{\ft@cmarkdef}[4]{%
  % cmarkbox-name cmark cmark-xlength cmark-space
  \ft@savenamebox{#1}{}% empty
  \ifcase#2% fttop
  \else%
    \ft@len=\dimexpr\ft@cmarkbox@length + #3\relax\relax%
    \ifdim\ft@len>0pt%
      \ft@do@cmarkdef{#1}{#2}{\dimexpr\ft@len + #4\relax}{\ft@len}%
    \fi%
  \fi%
  \ft@dbgbox{\@nameuse{#1}}%
}
```

1.3.2 Additional info box

Customization

```
\ftoptboxcfg
  \optboxcfg
```

```
\newlength{\ft@optbox@vsp}
\setlength{\ft@optbox@vsp}{.1\baselineskip}
\newlength{\ft@optbox@indent}
\setlength{\ft@optbox@indent}{1.1\ft@unit}
\newcommand{\ft@optbox@font}{\relsize{-1}}%{\footnotesize}
\newcommand{\ft@optbox@linestretch}{.75}
```

```
\newcommand{\ftoptboxcfg}[4]{% vsp indent font linestretch
  \ft@ifempty{#1}{}{%
    \setlength{\ft@optbox@vsp}{#1}%
  }%
  \ft@ifempty{#2}{}{%
    \setlength{\ft@optbox@indent}{#2}%
  }%
  \ft@ifempty{#3}{}{%
    \renewcommand{\ft@optbox@font}{#3}%
  }%
  \ft@ifempty{#4}{}{%
    \renewcommand{\ft@optbox@linestretch}{#4}%
  }%
}
\ft@alias{optboxcfg}
```

Parsing

\ftmaleline
\maleline
\fthaschild
\haschild
\ftprivate
\private

```
\newcommand{\ftmaleline}{ft@attr0}
\let\ftfemaleline=\ftmaleline
\let\ftpatrilineal=\ftmaleline
\let\ftmatrilineal=\ftmaleline
\newcommand{\fthaschild}{ft@attr1}
\newcommand{\ftprivate}{ft@attr2}
\ft@alias{maleline}
\ft@alias{femaleline}
\ft@alias{patrilineal}
\ft@alias{matrilineal}
\ft@alias{haschild}
\ft@alias{private}
```

\ft@optlist

Extracts the attributes from ⟨*option-list*⟩ (which is ⟨*additional info*⟩ itself) and sets a global flag whose name is generated using ⟨*box-name*⟩. Other than the attributes are appended another list, ⟨*list*⟩ which will be printed later.

```
\newcommand{\ft@optlist}[3]{% list box-name option-list
  \edef\@male{\ftmaleline}%
  \edef\@hasch{\fthaschild}%
  \edef\@priv{\ftprivate}%
  \@for\ft@iter:=#3\do{%
    \edef\@opt{\ft@iter}%
    \ft@dbgmsg{\@opt}%
    \ifx\@opt\@male%
      \global\ft@malelinetrue%
      \ft@namexdef{#2hasmaleline}{\ftmaleline}%
    \else%
      \ifx\@opt\@hasch%
        \ft@namexdef{#2haschild}{\fthaschild}%
      \else%
        \ifx\@opt\@priv%
          \ft@namexdef{#2private}{\ftprivate}%
        \else%
          \ft@list@append{#1}{\@opt}%
```

```
        \fi%
      \fi%
    \fi%
  }%
}
```

Define a box

\ft@opt

```
\newsavebox{\ft@optbox}
\newcommand{\ft@opt}[1]{% option-list
  \savebox{\ft@optbox}{%
    \hspace{\ft@cmarkW}%
    \hspace{\ft@optbox@indent}%
    \vbox{%
      \def\baselinestretch{\ft@optbox@linestretch}%
      \ft@optbox@font%
      \vspace{\ft@optbox@vsp}%
      \@for\ft@iter:=#1\do{%
        \vss%
        \hbox{\ft@iter}%
        \setbox\@tempboxa=\hbox{\ft@iter}%
        \global\ft@depth=\dp\@tempboxa%
      }%
    }%
  }%
  \ft@len=\dimexpr\ht\ft@optbox + \dp\ft@optbox - \ft@depth\relax%
  \ft@boxsz{\ft@optbox}{\ft@len}{\ft@depth}%
  \ft@dbgbox{\ft@optbox}%
}
```

1.3.3 Name box

Customization

\ftnameboxcfg
\nameboxcfg

```
\newlength{\ft@namebox@sp}
\setlength{\ft@namebox@sp}{.25\ft@unit}
\newcommand{\ft@namebox@font}{}% empty
\newlength{\ft@namebox@maleline@sp}
\setlength{\ft@namebox@maleline@sp}{.25\ft@unit}
\newlength{\ft@namebox@maleline@length}
\setlength{\ft@namebox@maleline@length}{1.5\ft@unit}
\newcommand{\ftnameboxcfg}[4]{% sp font maleline-sp maleline-length
  \ft@ifempty{#1}{}{%
    \setlength{\ft@namebox@sp}{#1}%
  }%
  \ft@ifempty{#2}{}{%
    \renewcommand{\ft@namebox@font}{#2}%
  }%
  \ft@ifempty{#3}{}{%
    \setlength{\ft@namebox@maleline@sp}{#3}%
  }%
  \ft@ifempty{#4}{}{%
```

```
    \setlength{\ft@namebox@maleline@length}{#4}%
  }%
}
\ft@alias{nameboxcfg}
```

Define a box

\ft@name

Internally layouts \ft@cmarkbox, ⟨*individual-name*⟩, and \ft@malelinebox. It was difficult to set the length of maleline, ie. where to begin the line. Which is better for the head of maleline, at the box end of the ⟨*individual-name*⟩ or at the box end including all the ⟨*title*⟩, ⟨*individual-name*⟩ and ⟨*additional info*⟩?

The latter looks good, especially when the defined individual box is used alone. But it makes hard for other box definitions to calculate the extra line length to align equal. For such calculation, the former is better since it just has to calculate the difference of the length of names. Finally, I decided to start the line at the end of ⟨*individual-name*⟩, and the default length is \ft@namebox@maleline@length.

In other words, it can happen when an individual box is used alone, the ⟨*title*⟩ or the ⟨*additional info*⟩ may be longer than maleline.

```
\newif\ifft@maleline
\newsavebox{\ft@malelinebox}
\newsavebox{\ft@namebox}
\newcommand{\ft@name}[3]{% box-name individual-name maleline-xlength
  \setbox\@tempboxa=\hbox{\ft@namebox@font#2}%
  \ft@len=\dimexpr\wd\@tempboxa + \ft@cmarkW%
    + \ft@namebox@maleline@sp\relax%
  \ft@namexdefstrip{#1nameX}{\ft@len}%
  %
  \ft@len=\dimexpr\wd\@tempboxa/2 + \ft@cmarkW\relax%
  \ft@namexdefstrip{#1nameCX}{\ft@len}%
  %
  \def\@nm{\ft@namebox@font#2}%
  \ifft@maleline%
    % why are two 'relax'es necessary?
    \@tempskipb=\dimexpr\ft@namebox@maleline@length%
      - \ft@cmarkbox@length + #3\relax\relax%
    \ifdim\@tempskipb>0pt%
      \ft@cmarkdef{ft@malelinebox}{\ftbiological}{\@tempskipb}{0pt}%
      \def\@nm{%
        {\ft@namebox@font#2}%
        \hspace{\ft@namebox@maleline@sp}%
        \usebox{\ft@malelinebox}%
      }%
    \fi%
  \fi%
  \ft@dbgmsg{H \the\ht\@tempboxa, D \the\dp\@tempboxa}%
  \ft@dbgmsg{H \the\ht\ft@cmarkbox, D \the\dp\ft@cmarkbox}%
  \savebox{\ft@namebox}{%
    \usebox{\ft@cmarkbox}%
    \@nm%
  }%
  \ft@boxsz{\ft@namebox}{\ht\@tempboxa}{\dp\@tempboxa}%
  \ft@dbgbox{\ft@namebox}%
```

```
}
```

1.3.4 Title box

Customization

\fttitleboxcfg
\titleboxcfg

```
\newlength{\ft@titlebox@indent}
\setlength{\ft@titlebox@indent}{-.25\ft@unit}
\newcommand{\ft@titlebox@font}{\relsize{-2}}%{\scriptsize}
\newcommand{\ft@titlebox@linestretch}{.25}
\newlength{\ft@titlebox@vsp}
\setlength{\ft@titlebox@vsp}{.1\baselineskip}
\newcommand{\fttitleboxcfg}[4]{% indent font linestretch vsp
  \ft@ifempty{#1}{}{%
    \setlength{\ft@titlebox@indent}{#1}%
  }%
  \ft@ifempty{#2}{}{%
    \renewcommand{\ft@titlebox@font}{#2}%
  }%
  \ft@ifempty{#3}{}{%
    \renewcommand{\ft@titlebox@linestretch}{#3}%
  }%
  \ft@ifempty{#4}{}{%
    \setlength{\ft@titlebox@vsp}{#4}%
  }%
}
\ft@alias{titleboxcfg}
```

Define a box

\ft@title

```
\newsavebox{\ft@titlebox}
\newcommand{\ft@title}[1]{% title
  \ft@len=\dimexpr\ft@cmarkW + \ft@titlebox@indent\relax%
  \savebox{\ft@titlebox}{%
    \vbox{%
      \def\baselinestretch{\ft@titlebox@linestretch}%
      \ft@titlebox@font%
      %
      \setbox\@tempboxa=\hbox{#1}%
      \ft@dbgmsg{H \the\ht\@tempboxa, D \the\dp\@tempboxa}%
      \global\ft@depth=\dp\@tempboxa%
      \hbox{\hspace{\ft@len}#1}%
      \vspace{\ft@titlebox@vsp}%
      \global\advance\ft@depth \ft@titlebox@vsp%
    }%
  }%
  \ft@dbgmsg{H \the\ht\ft@titlebox, D \the\dp\ft@titlebox}%
  \ft@len=\dimexpr\ht\ft@titlebox + \dp\ft@titlebox - \ft@depth\relax%
  \ft@boxsz{\ft@titlebox}{\ft@len}{\ft@depth}%
  \ft@dbgbox{\ft@titlebox}%
}
```

1.3.5 Combine the boxes — core

Generate the boxes

\ft@indvdlbox@gen
```
\newcommand{\ft@indvdlbox@gen}[4]{%
  % box-name title individual-name maleline-xlength
  \ft@width=0pt%
  \IfValueT{#2}{%
      \ft@title{#2}%
      \ft@width=\wd\ft@titlebox%
  }%
  \global\@tempswafalse%
  \ifx\ft@indvdl@opts\empty\else%
    \global\@tempswatrue%
    \ft@opt{\ft@indvdl@opts}%
    \ifdim\ft@width<\wd\ft@optbox%
      \ft@width=\wd\ft@optbox%
    \fi%
  \fi%
  \ft@name{#1}{#3}{#4}%
}
```

Calculate the size of a name box

\ft@indvdlbox@calc
```
\newcommand{\ft@indvdlbox@calc}[2]{% box-name title
  \global\ft@width=0pt%
  \global\ft@height=0pt%
  \global\ft@depth=0pt%
  \def\ft@dpri##1{\ft@dbgmsg{##1 W \the\ft@width, H \the\ft@height,%
      D \the\ft@depth}}%
  \ft@dpri{h0}%
  \IfValueT{#2}{%
    \global\ft@width=\wd\ft@titlebox%
    \global\ft@height=\dimexpr\ht\ft@titlebox + \dp\ft@titlebox\relax%
    \ft@dpri{h1}%
  }%
  %
  \global\advance\ft@height \ht\ft@namebox%
  %
  % mark this point of the height, to invert it later
  \ft@y=\dimexpr\ft@height - \ft@indvdl@nameCY\relax%
  %
  \ifdim\ft@width<\wd\ft@namebox%
    \global\ft@width=\wd\ft@namebox%
  \fi%
  \if@tempswa%
    \global\advance\ft@height \dp\ft@namebox%
  \else%
    \global\ft@depth=\dp\ft@namebox%
  \fi%
  \ft@dpri{h2}%
  %
  \if@tempswa%
    \ifdim\ft@width<\wd\ft@optbox%
      \global\ft@width=\wd\ft@optbox%
```

```
    \fi%
    \global\advance\ft@height \ht\ft@optbox%
    \global\ft@depth=\dp\ft@optbox%
    \ft@dpri{h3}%
  \fi%
  \ft@dpri{h4}%
  %
  % invert the y
  \ft@len=\dimexpr\ft@height - \ft@y\relax%
  \ft@namexdefstrip{#1nameCY}{\ft@len}%
}
```

Layout them by `picture`

\ft@indvdlbox@layout

```
\newcommand{\ft@indvdlbox@layout}[2]{% box-name title
  \ft@newnamebox{#1}{%
    \edef\@w{\strip@pt\ft@width}%
    \edef\@h{\strip@pt\ft@height}%
    \unitlength=1pt\relax% coordinates are in pt
    \begin{picture}(\@w,\@h)%
      \ft@dbgframe{\@w,\@h}%
      \ft@dbgplot{0,\@nameuse{#1nameCY}}%
      %
      \IfValueT{#2}{%
        \advance\ft@height -\dimexpr\ht\ft@titlebox%
          + \dp\ft@titlebox\relax%
        \put(0,\strip@pt\ft@height){\usebox{\ft@titlebox}}%
      }%
      \advance\ft@height -\ht\ft@namebox%
      \put(0,\strip@pt\ft@height){\usebox{\ft@namebox}}%
      \advance\ft@height -\dp\ft@namebox%
      \if@tempswa%
        \advance\ft@height -\ht\ft@optbox%
        \put(0,\strip@pt\ft@height){\usebox{\ft@optbox}}%
      \fi%
    \end{picture}%
  }%
  % height should hold the original value
  \ft@nameboxsz{#1}{\ft@height}{\ft@depth}%
}
```

Main function to combine the boxes

\ft@indvdlbox

```
\newcommand{\ft@indvdlbox}[4]{%
  % box-name title individual-name  maleline-xlength
  \ft@indvdlbox@gen{#1}{#2}{#3}{#4}%
  \ft@indvdlbox@calc{#1}{#2}%
  \ft@indvdlbox@layout{#1}{#2}%
  %
  \ft@dbgbox[\ft@dbgplot{0,\@nameuse{#1nameCY}}%
    \ft@dbgplot{\@nameuse{#1nameCX},\strip@pt\ft@height}%
    \ft@dbgplot{\@nameuse{#1nameX},\strip@pt\ft@height}%
  ]{\@nameuse{#1}}%
}
```

1.3.6 Individual box — interface

\ftindvdldef
\indvdldef

```
% [child-mark] box-name [title] individual-name
%       [option-list...] [maleline-xlength]
\NewDocumentCommand{\ftindvdldef}{O{\ftbiological}momoO{Opt}}{%
  \setbox\@tempboxa=\hbox{\ft@namebox@font#4}%
  \global\ft@indvdl@nameH=\ht\@tempboxa%
  \global\ft@indvdl@nameD=\dp\@tempboxa%
  \if@tate%
    \global\ft@indvdl@nameCY%
      =\dimexpr(\ft@indvdl@nameH - \ft@indvdl@nameD)/2\relax%
  \else%
    \@ifundefined{jlreqsetup}{%
      % not jlreq
      \global\ft@indvdl@nameCY=\dimexpr(\ft@indvdl@nameH%
        - \ft@indvdl@nameD) / 2\relax\relax%
    }{%
      % jlreq
      % magic number! depends on font?
      \global\ft@indvdl@nameCY=\dimexpr\ft@indvdl@nameH/2 - .66pt\relax%
    }%
  \fi%
  \ft@dbgmsg{H \the\ft@indvdl@nameH, D \the\ft@indvdl@nameD,%
    CY \the\ft@indvdl@nameCY}%
  %
  \ft@cmarkdef{ft@cmarkbox}{#1}{Opt}{\ft@namebox@sp}%
  \ifcase#1% fttop
    \ft@namexdef{#2hascmark}{#1}%
  \or% ftbiological
    \ft@namexdef{#2hascmark}{#1}%
  \or% ftadopted
    \ft@namexdef{#2hascmark}{#1}%
  \fi%
  \global\ft@cmarkW=\wd\ft@cmarkbox%
  %
  \global\ft@malelinefalse%
  \xdef\ft@indvdl@opts{}%
  \IfValueT{#5}{%
    \ft@optlist{\ft@indvdl@opts}{#2}{#5}%
    \ft@dbgmsg{opt \ft@indvdl@opts}%
  }%
  %
  \ft@indvdlbox{#2}{#3}{#4}{#6}%
}
\ft@alias{indvdldef}
```

2 Sibling Box (ft-sibling.dtx)

`\sblngdef` `\sblngdef{⟨new box name⟩} {⟨name list of individual boxes⟩}`

`{⟨name list of individual boxes⟩}` is the comma separated box names which are defined by `\indvdldef`. They are aligned and connected by a line. All names are NOT control sequence (no backslash).

If any of the siblings has a `\maleline` attribute, then the length of all lines are set to the longest one.

Like `\indvdldef`, `\sblngdef` defines a few connection points (CS) to be used later. The origin is left-bottom of the box and the unit is `pt`.

- ⟨*box name*⟩`nameCY`

  Center of the line which connects all the siblings.

  The line begins at the head of the child-mark of the first element of the given list, and ends at the last element.

- ⟨*box name*⟩⟨*individual box name*⟩`nameCY`

  Center of the height for each individual name.

  In other words, shifted ⟨*individual box name*⟩`nameCY` which `\indvdldef` defined.

`\ivaldef` `\ivaldef{⟨new box name⟩} {⟨length⟩}`

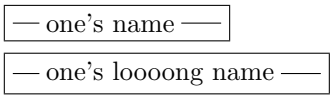
Sometimes an extra space is necessary between the siblings who have many descendants. For such spaces, you can define an interval box by `\ivaldef`. It defines an blank box who has a specified size. There are three pre-defined interval boxes, `\ival`, `\ivali`, and `\ivalii`. They have the size of `0.5em`, `1em`, `2em` for each.

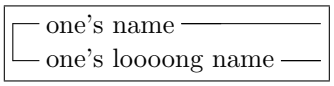
2.1 Customization

`\sblngboxcfg` `\sblngboxcfg{⟨space between the siblings⟩}`

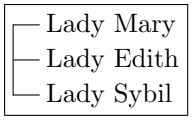
2.2 Example

1. ```
   \indvdldef{A}{one's name}[\maleline]
   \indvdldef{B}{one's loooong name}[\maleline]
   \sblngdef{ABbro}{A,B}
   ```

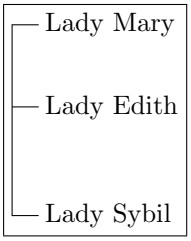
   


2. ```
   \sblngboxcfg{1ex}
   \indvdldef{youngSybil}{Lady Sybil}
   \indvdldef{youngEdith}{Lady Edith}
   \indvdldef{youngMary}{Lady Mary}
   \sblngdef{youngSis}{youngMary,youngEdith,youngSybil}
   \fbox{\usebox{\youngSis}}
   ```



\youngSisnameCY	15.69443
\youngSisyoungMarynameCY	28.88885
\youngSisyoungEdithnameCY	15.69443
\youngSisyoungSybilnameCY	2.5

3. `\sblngdef{youngSis}{youngMary,ivali,youngEdith,ivalii,youngSybil}`
   `\fbox{\usebox{\youngSis}}`



Jump to next section

2.3 Implementation

2.3.1 Interval box

\ftivaldef
\ivaldef
```
\newcommand{\ftivaldef}[2]{% box-name length
  \ft@newnamebox{#1}{\vbox to #2{\hsize=1pt}}%
  \ft@len=#2%
  \ft@namexdefstrip{#1ival}{\ft@len}% just a flag
  \divide\ft@len 2%
  \ft@namexdefstrip{#1nameCY}{\ft@len}%
}
\ft@alias{ivaldef}
```

Pre-defined interval boxes

\ftival
\ival
```
\ftivaldef{ftival}{.5\ft@unit}
\ft@alias{ival}
\ft@alias{ivalnameCY}
\ft@alias{ivalival}
```

\ftivali
\ivali
```
\ftivaldef{ftivali}{1\ft@unit}
\ft@alias{ivali}
\ft@alias{ivalinameCY}
\ft@alias{ivaliival}
```

\ftivalii
\ivalii
```
\ftivaldef{ftivalii}{2\ft@unit}
\ft@alias{ivalii}
\ft@alias{ivaliinameCY}
\ft@alias{ivaliiival}
```

Customization

\ftsblngboxcfg
\sblngboxcfg

```
\newlength{\ft@sblng@vsp}
\setlength{\ft@sblng@vsp}{.5\baselineskip}%
\newcommand{\ftsblngboxcfg}[1]{% space-length
  \global\ft@sblng@vsp=#1%
}
\ft@alias{sblngboxcfg}
```

2.3.2 Sibling box — core

If any of the siblings has an attributes \maleline or alike, then finds the longest one and sets its length to all others. Between the name and \maleline, insert a space \ft@namebox@maleline@sp.

```
\newcommand{\ft@sblng@maleline}[2]{% sibling y
  \@ifundefined{#1hasmaleline}{}{%
    \ft@x=\@nameuse{#1nameX}pt\relax%
    \ifdim\ft@x<\ft@width%
      \put(\strip@pt\ft@x,\strip@pt#2){%
        \line(1,0){\strip@pt\dimexpr\ft@width - \ft@x}%
      }%
    \fi%
  }%
}
```

\ft@sblng@connect

```
\newlength{\ft@c}
\newcommand{\ft@sblng@connect}[1]{% box-name
  %
  % draw a line to connect all the siblings
  % length = eldest CY - youngest CY
  % and calculate nameCY of the box
  % nameCY = length/2 + youngest CY
  %
  \ft@y=\@nameuse{#1\ft@lastcmark nameCY}pt%
  \ft@dbgplot{1,\strip@pt\ft@y}%
  \ft@yy=\@nameuse{#1\ft@firstcmark nameCY}pt%
  \ft@dbgplot{1,\strip@pt\ft@yy}%
  \ft@len=\dimexpr\ft@yy - \ft@y\relax%
  %
  \ft@c=\dimexpr\ft@len/2 + \ft@y\relax%
  \ft@namexdefstrip{#1nameCY}{\ft@c}%
  \ft@dbgplot{1,\strip@pt\ft@c}%
  %
  \ifnum\@nameuse{\ft@lastcmark hascmark}=\ftadopted%
    \advance\ft@y -\dimexpr\ft@cmarkbox@adopted@sep/2\relax%
    \advance\ft@len \dimexpr\ft@cmarkbox@adopted@sep/2\relax%
  \fi%
  \ifnum\@nameuse{\ft@firstcmark hascmark}=\ftadopted%
    \advance\ft@len \dimexpr\ft@cmarkbox@adopted@sep/2\relax%
  \fi%
  %
  \ifdim\ft@len<2pt%
```

```
      %\ft@len=\@nameuse{\ft@firstcmark nameCY}pt\relax%
    \else%
      \advance\ft@y -\dimexpr\arrayrulewidth/2\relax%
      \advance\ft@len \arrayrulewidth%
      \put(0,\strip@pt\ft@y){\line(0,1){\strip@pt\ft@len}}%
    \fi%
  }
```

\ft@sblng@layout

```
  \newcommand{\ft@sblng@layout}[2]{% box-name individual-name-list
    \ft@newnamebox{#1}{%
      \edef\@w{\strip@pt\ft@width}%
      \edef\@h{\strip@pt\ft@height}%
      \unitlength=1pt\relax% coordinates are in pt
      \begin{picture}(\@w,\@h)%
        \ft@dbgframe{\@w,\@h}%
        %
        \@for\ft@iter:=#2\do{%
          \edef\ft@sblng@name{\ft@iter}%
          \ft@dbgmsg{H \the\ft@height,%
            \ft@sblng@name nameCY \@nameuse{\ft@sblng@name nameCY}pt,%
            \the\ft@y}%
          %
          % calculate the nameCY for each
          \advance\ft@height -\ht\@nameuse{\ft@sblng@name}%
          \global\ft@y=\dimexpr\@nameuse{\ft@sblng@name nameCY}pt%
            + \ft@height\relax%
          \ft@dbgplot{0,\strip@pt\ft@y}%
          \ft@namexdefstrip{#1\ft@sblng@name nameCY}{\ft@y}%
          %
          % align the malelines
          \ft@sblng@maleline{\ft@sblng@name}{\ft@y}%
          %
          % place the individual boxes
          \put(0,\strip@pt\ft@height){\usebox{\@nameuse{\ft@sblng@name}}}%
          \advance\ft@height -\dimexpr\dp\@nameuse{\ft@sblng@name}%
            + \ft@sblng@vsp\relax%
        }%
        %
        % connect them
        \ifx\ft@firstcmark\relax\else%
          \ifx\ft@lastcmark\relax\else%
            \ifx\ft@firstcmark\ft@lastcmark\else%
              \ft@sblng@connect{#1}%
            \fi%
          \fi%
        \fi%
      \end{picture}%
    }%
    % height should hold the original value
    \ft@nameboxsz{#1}{\ft@height}{\ft@depth}%
  }
```

2.3.3 Sibling box — interface

\ftsblngdef
\sblngdef

```
\newcommand{\ftsblngdef}[2]{% box-name comma-separated-individuals
  %
  % calculate the size of the box
  \ft@width=0pt%
  \ft@height=0pt%
  \ft@theight=0pt%
  \ft@box@has@malelinefalse%
  \@tempswatrue%
  \let\ft@firstcmark=\relax%
  \let\ft@lastcmark=\relax%
  \def\ft@dpri##1{\ft@dbgmsg{##1 W \the\ft@width, H \the\ft@height,%
      D \the\ft@depth}}%
  \ft@dpri{h0}%
  \@for\ft@iter:=#2\do{%
    \if@tempswa%
      \xdef\ft@eldest{\ft@iter}%
      \@tempswafalse%
    \fi%
    \xdef\ft@youngest{\ft@iter}%
    \@ifundefined{ft@firstcmark}{%
      \@ifundefined{\ft@youngest hascmark}{}{%
        \global\let\ft@firstcmark=\ft@youngest%
      }%
    }{%
      \@ifundefined{\ft@youngest hascmark}{}{%
        \global\let\ft@lastcmark=\ft@youngest%
      }%
    }%
    \@ifundefined{\ft@youngest hasmaleline}{}{%
      \global\ft@box@has@malelinetrue%
    }%
    %
    \setlength{\ft@len}{\wd\@nameuse{\ft@youngest}}%
    \ifdim\ft@width<\ft@len%
      \global\ft@width=\ft@len%
      \@ifundefined{\ft@youngest hasmaleline}{%
        \global\ft@widest@has@no@malelinetrue%
      }{%
        \global\ft@widest@has@no@malelinefalse%
      }%
    \fi%
    \global\advance\ft@theight \dimexpr\ht\@nameuse{\ft@youngest}%
      + \dp\@nameuse{\ft@youngest} + \ft@sblng@vsp\relax%
    \ft@dpri{\ft@youngest}%
  }%
  \advance\ft@theight -\ft@sblng@vsp%
  \ft@depth=\dp\@nameuse{\ft@youngest}%
  \ft@height=\ft@theight%
  \advance\ft@height -\ft@depth%
  \ifft@widest@has@no@maleline%
    \ifft@box@has@maleline%
```

```
      \global\advance\ft@width \dimexpr\ft@namebox@maleline@sp%
        + \ft@namebox@maleline@length\relax%
    \fi%
  \fi%
  %
  % layout the all boxes
  \ft@sblng@layout{#1}{#2}%
  \ft@dbgbox{\@nameuse{#1}}%
}
\ft@alias{sblngdef}
```

3 Parent-child Relationship Box or Generations Box (ft-gens.dtx)

\pcdef `\pcdef{⟨new box name⟩} {⟨parent box name⟩} {⟨child box name⟩}`

Defines a parent-child relationship box. Connects the given ⟨*parent box*⟩ and ⟨*child box*⟩ by a line, and creates a new box ⟨*new box name*⟩.

⟨*parent box*⟩ is a box who has only one line from an individual name to one's child. For example, the box created by `\indvdldef` with `\maleline` attribute (and equivalent) is specified. Obviously, ⟨*child box*⟩ is a box who has a line to one's parent. For example, the box created by `\indvdldef` with `\biological` or `\adopted` is specified as a child mark.

`\pcdef` is a simplified version of `\gensdef`, which is discussed next.

\gensdef `\gensdef{⟨new box name⟩} {⟨parent box name⟩} {⟨list of connection-pair⟩}`

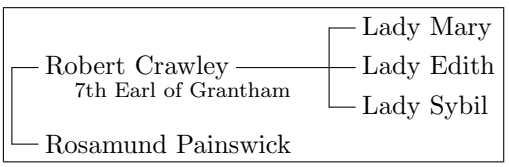
```
connection-pair :=
        {⟨individual box name in the parent box⟩}
        {⟨child box name⟩}
```

Defines a two-generations box. Same to `\pcdef`, ⟨*child box*⟩ is a box who has only one line to the parent, but ⟨*parent box*⟩ can have multiple lines to one's child. It is ⟨*connection-pair*⟩ that makes it clear which parent connects to which child box.

3.1 Example

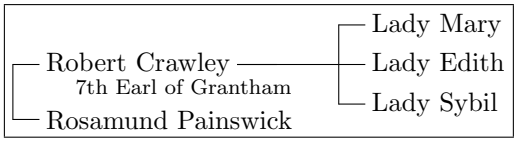
1. `\sblngdef` for daughters, `\pcdef`, and then `\sblngdef` for their parent generation.

```
\indvdldef{youngSybil}{Lady Sybil}
\indvdldef{youngEdith}{Lady Edith}
\indvdldef{youngMary}{Lady Mary}
\sblngdef{youngSis}{youngMary,youngEdith,youngSybil}

\indvdldef{Robert}{Robert Crawley}
          [7th Earl of Grantham,\maleline][2em]
\pcdef{RobertDaughters}{Robert}{youngSis}

\indvdldef{Rosamund}{Rosamund Painswick}
\sblngdef{RobertRosamund}{RobertDaughters,Rosamund}
\fbox{\usebox{\RobertRosamund}}
```



2. two \sblngdef, and then \gensdef. The result is essentially same. One difference is the space between the siblings which was automatically adjusted in previous example.

```
\sblngdef{youngSis}{youngMary,youngEdith,youngSybil}
\sblngdef{RobertRosamund}{Robert,Rosamund}
\gensdef{family}{RobertRosamund}{
  {Robert}{youngSis}
}
\fbox{\usebox{\family}}
```



3.2 The order of connecting multiple boxes

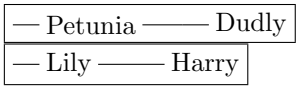
If we get \sblngdef as a tool to align the individual boxes in column, then \pcdef and \gensdef are the tool to align the boxes in row. When the siblings have their child for each, then there are multiple parent-child relationships, so it is better to call it generations box rather than parent-child box.

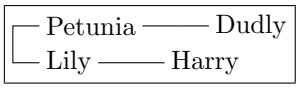
There are two ways to draw such tree. One is to define parent-child first and then define the siblings of the parent generation. The other is in the reverse order, eg. to define the siblings of the parent generation first and then define the parent-child relationship for each.

Let's consider about these two ways.

1. define two parent-child relationships, and then define the siblings.

```
\indvdldef{Dudly}{Dudly}
\indvdldef{Petunia}{Petunia}[\matrilineal]
\pcdef{Dursleys}{Petunia}{Dudly}

\indvdldef{Harry}{Harry}
\indvdldef{Lily}{Lily}[\matrilineal]
\pcdef{Potters}{Lily}{Harry}

\sblngdef{sis}{Dursleys,Potters}
```

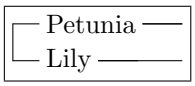


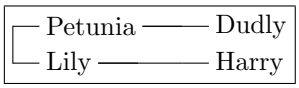


2. define the sisters, and then define the parent-child for each.

```
...
\sblngdef{sis}{Petunia,Lily}
\gensdef{twofam}{sis}{%
  {Petunia}{Dudly},%
```

```
  {Lily}{Harry}%
}
```



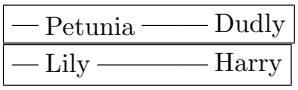


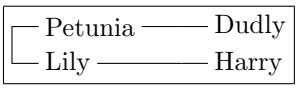
As you see, by the 1st method the length of two lines to their child differs and the positions (in horizontal) of the child generation are not equal. That makes the tree uneasy to understand straightforward. It is because that the feature of `\sblngdef` to set the line length to the longest one didn't work. The argument passed to `\sblngdef` were already connected to the child, so if `\sblngdef` extended the line it would be much worse result.

On the other hand, by the 2nd method, the argument passed to `\sblngdef` were not connected to the child. So it is harmless if `\sblngdef` extends the line.

Even if you took the 1st method, there still exists to make the line length equal. Using `\indvdldef` feature to adjust the line length, set the length of Lily's `\femaleline` (`\matrilineal`) to the one of Petunia's. To achieve this, calculate the difference of the name length of these sisters and give an optional argument of `\indvdldef`. The result should be same to above.

```
...
\newsavebox{\boxB}
\savebox{\boxA}{\hbox{Petunia}}
\savebox{\boxB}{\hbox{Lily}}
\indvdldef{Lily}{Lily}[\matrilineal][\dimexpr\wd\boxA - \wd\boxB\relax]
\pcdef{Potters}{Lily}{Harry}

\sblngdef{sis}{Dursleys,Potters}
```



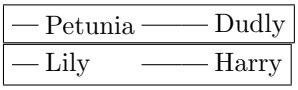


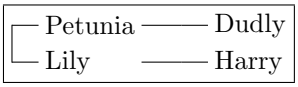
You can get the same result if you use `\nameboxcfg` since it has a feature to set the length of a line to child. But it is not a good idea to use `\nameboxcfg` every time when you `\indvdldef`. The value set by `\nameboxcfg` should be applied wider, and it is not supposed to use for a single `\indvdldef`. It is better to append an optional argument to `\indvdldef`.

There is one more option. It is to set the length of Lily's name to Petunia's. By this method, the space between Lily's name and the line to child becomes wider and the length of lines become equal.

```
...
\savebox{\boxA}{\hbox{Petunia}}
\indvdldef{Lily}{\hbox to \wd\boxA{Lily}}[\matrilineal]
```

```
\pcdef{Potters}{Lily}{Harry}

\sblngdef{sis}{Dursleys,Potters}
```





The sequence or the order to define and connect the boxes is important. In connecting the boxes, this package considers the size of the being connected individual boxes. For example, the sibling box considers the height of the individual box and makes the boxes to be never overlapped. But in connecting a child to the already defined sibling box, this feature doesn't work. So the children of the siblings may be overlapped. In this case, you need to insert the interval box between the siblings manually.

Jump to next section

3.3 Implementation

3.3.1 Generations box — core

Connection pair

`\ft@getpair` Extracts a connection-pair from the given list, defines the connection-point in the parent box (the former of the pair) as `\ft@cpoint`, and defines the child box name (the latter of the pair) as `\ft@kids`.

```
\def\ft@getpair#1#2#3{% connection-pair parent-box-name
  \ft@dbgmsg{args #1, #2, #3}%
  \@ifundefined{#3#1nameCY}{%
    \@ifundefined{#3#1mrrgCY}{%
      \xdef\ft@cpoint{#1nameCY}%
    }{%
      \xdef\ft@cpoint{#3#1mrrgCY}%
    }%
  }{%
    \xdef\ft@cpoint{#3#1nameCY}%
  }%
  \xdef\ft@kids{#2}%
}
```

Top margin

`\def@calc@xtop`

```
\newcommand{\ft@calc@xtop}[4]{% name parent cpoint kids
  % top half of kids
  \ft@len=\dimexpr\ht\@nameuse{#4} - \@nameuse{#4nameCY}pt\relax\relax%
  % top half of parent cpoint
  \@tempskipa=\dimexpr\ht\@nameuse{#2} - \@nameuse{#3}pt\relax\relax%
  %
  \ifdim\ft@len<\@tempskipa%
    \ft@len=0pt%
```

```
    \else%
      \advance\ft@len -\@tempskipa%
    \fi%
    \global#1=\ft@len%
  }
```

Bottom margin

`\ft@calc@xbottom`

```
  \newcommand{\ft@calc@xbottom}[4]{% name parent cpoint kids
    % bottom half of kids
    \ft@len=\@nameuse{#4nameCY}pt\relax%
    % bottom half of parent cpoint
    \@tempskipa=\@nameuse{#3}pt\relax%
    %
    \ifdim\ft@len=\@tempskipa%
      \ft@len=0pt%
      \global\setlength{\ft@depth}{\dp\@nameuse{#2}}%
      \ifdim\ft@depth<\dp\@nameuse{#4}%
        \global\setlength{\ft@depth}{\dp\@nameuse{#4}}%
      \fi%
    \else%
      \ifdim\ft@len<\@tempskipa%
        \ft@len=0pt%
        \global\setlength{\ft@depth}{\dp\@nameuse{#2}}%
      \else%
        \advance\ft@len -\@tempskipa%
        \global\setlength{\ft@depth}{\dp\@nameuse{#4}}%
      \fi%
    \fi%
    \global#1=\ft@len%
  }
```

Calculate the box size

`\ft@gens@size`

```
  \newlength{\ft@xtop}
  \newlength{\ft@xbottom}
  \newcommand{\ft@gens@size}[2]{% parent-box connect-pair-list
    \@tempswatrue%
    \ft@width=0pt%
    \@for\ft@iter:=#2\do{%
      \expandafter\ft@getpair\ft@iter{#1}%
      \ft@dbgmsg{\ft@cpoint and \ft@kids}%
      \if@tempswa%
        \ft@calc@xtop{\ft@xtop}{#1}{\ft@cpoint}{\ft@kids}%
        \@tempswafalse%
      \fi%
      \setlength{\ft@len}{\wd\@nameuse{\ft@kids}}%
      \ifdim\ft@width<\ft@len%
        \global\ft@width=\ft@len%
      \fi%
    }%
```

```
    \ft@calc@xbottom{\ft@xbottom}{#1}{\ft@cpoint}{\ft@kids}%
    \ft@dbgmsg{xtop \the\ft@xtop, xbottom \the\ft@xbottom}%
    %
    %\ft@x=\dimexpr\wd\@nameuse{#1}% - \ft@cmarkbox@length\relax%
    \ft@x=\wd\@nameuse{#1}\relax% - \ft@cmarkbox@length\relax%
    \ft@dbgmsg{x \the\ft@x}%
    \advance\ft@width \ft@x%
    \ft@dbgmsg{w \the\ft@width}%
    \ft@height=\dimexpr\ht\@nameuse{#1} + \ft@xtop + \ft@xbottom\relax%
    \ft@dbgmsg{kids H \the\ht\@nameuse{\ft@kids}}%
    \ft@dbgmsg{H \strip@pt\ft@height, D \strip@pt\ft@depth}%
  }
```

Layout

\ft@gens@layout

```
  \newcommand{\ft@gens@layout}[3]{%
    % box-name parent-box-name {{parent-name} {child-name}, ...}
    \ft@newnamebox{#1}{%
      \edef\@w{\strip@pt\ft@width}%
      \edef\@h{\strip@pt\ft@height}%
      \unitlength=1pt\relax% coordinates are in pt
      \begin{picture}(\@w,\@h)%
        \ft@dbgframe{\@w,\@h}%
        %
        \ft@y=\ft@xbottom%
        \ft@dbgplot{0,\strip@pt\ft@y}%
        \put(0,\strip@pt\ft@y){\usebox{\@nameuse{#2}}}%
        \advance\ft@y \@nameuse{#2nameCY}pt%
        \ft@namexdefstrip{#1nameCY}{\ft@y}%
        %\ft@namexdefstrip{#1#2nameCY}{\ft@y}%
        %
        \@for\ft@iter:=#3\do{%%
          \expandafter\ft@getpair\ft@iter{#2}%
          \ft@dbgmsg{\ft@cpoint and \ft@kids}%
          %
          \ft@y=\dimexpr\ft@xbottom + \@nameuse{\ft@cpoint}pt\relax%
          \ft@dbgmsg{parent cpoint \the\ft@y}%
          \ft@dbgplot{\strip@pt\ft@x,\strip@pt\ft@y}%
          %
          \advance\ft@y  -\@nameuse{\ft@kids nameCY}pt%
          \ft@dbgmsg{final child y \the\ft@y}%
          \put(\strip@pt\ft@x,\strip@pt\ft@y){%
            \usebox{\@nameuse{\ft@kids}}}%
          \ft@namexdefstrip{#1\ft@kids Y}{\ft@y}%
        }%
      \end{picture}%
    }%
  }
```

3.3.2 Generations box — interface

\ftgensdef

```
\newcommand{\ftgensdef}[3]{%
  % box-name parent-box-name {{parent-name} {child-name}, ...}
  %
  % calculate the size of the new box
  \ft@gens@size{#2}{#3}%
  %
  % draw them all
  \ft@gens@layout{#1}{#2}{#3}%
  %
  \@ifundefined{#2hascmark}{}{%
    \ft@namexdef{#1hascmark}{\@nameuse{#2hascmark}}%
  }%
  \ft@nameboxsz{#1}{\ft@height}{\ft@depth}%
}
\ft@alias{gensdef}
```

3.3.3 Parent-child box — interface

\ftpcdef

```
\newcommand{\ftpcdef}[3]{% box-name parent-box-name child-box-name
  \ftgensdef{#1}{#2}{{#2}{#3}}%
}
\ft@alias{pcdef}
```

4 Marriage Box (ft-marriage.dtx)

\mrrgdef `\mrrgdef{⟨new box name⟩} {⟨spouse list A⟩} {⟨oneself⟩} {⟨spouse list B⟩} [⟨childline xlength⟩]`

Defines a marriage box with a specified name ⟨*new box name*⟩. To support remarrying and the concubines, the spouses are specified by a list. The element of the list is a box name defined by `\indvdldef`. ⟨*spouse list A*⟩ is placed upper side of ⟨*oneself*⟩, and ⟨*spouse list B*⟩ is lower side. All box names are NOT control sequence (no backslash).

Aligns them in the same column, and connects them by a double line if the marriage is official. If the marriage is not official (`\private` attribute), uses a dashed double line. Those double line is placed at the center of the length of the name of ⟨*oneself*⟩.

If a spouse has a child (`\haschild` attribute), then the line to their child is drawn from the center of the double line.

Like `\sblngdef`, the interval box can be inserted if you want more spaces.

Like `\indvdldef`, some connection points are defined. Their origin is left-bottom of the box and the unit is `pt`.

- ⟨*box name*⟩`nameCY`

  the center of the height of the name of ⟨*oneself*⟩

- ⟨*box name*⟩⟨*individual box name*⟩`nameCY`

  the center of the height of the name of who has any child-mark

- ⟨*box name*⟩⟨*individual box name*⟩`mrrgCY`

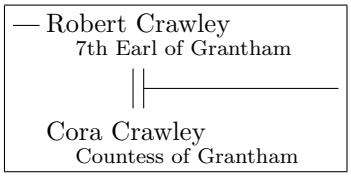
  the center of the double line when any spouse has `\haschild` attribute

4.1 Customization

\mrrgboxcfg `\mrrgboxcfg{⟨space between two lines⟩} {⟨space between name and the line⟩} {⟨line length⟩}`

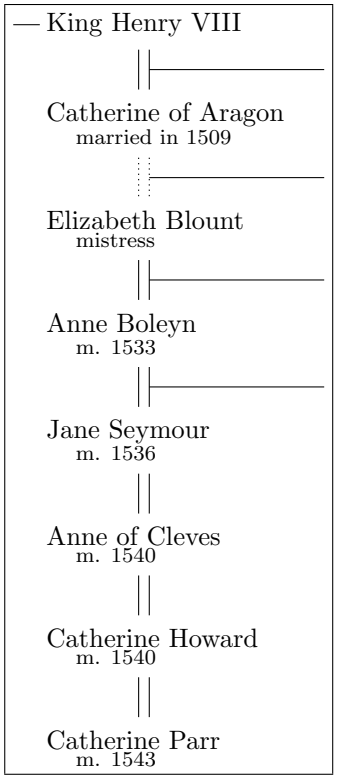
4.2 Example

1.
```
\indvdldef{Robert}{Robert Crawley}[7th Earl of Grantham]
\indvdldef[\ftblank]{Cora}{Cora Crawley}
          [Countess of Grantham,\haschild]
\mrrgdef{seven}{}{Robert}{Cora}
\fbox{\usebox{\seven}}
```



\sevennameCY	51.8446
\sevenRobertnameCY	51.8446
\sevenCoramrrgCY	28.14452

2.
```
\indvdldef{HenryVIII}{King Henry VIII}[]

\indvdldef[\ftblank]{CatherineofAragon}{Catherine of Aragon}
          [married in 1509,\haschild]
\indvdldef[\ftblank]{Elizabeth}{Elizabeth Blount}
          [mistress,\haschild,\private]
\indvdldef[\ftblank]{AnneBoleyn}{Anne Boleyn}
          [m. 1533,\haschild]
\indvdldef[\ftblank]{Jane}{Jane Seymour}
          [m. 1536,\haschild]
\indvdldef[\ftblank]{AnneofCleves}{Anne of Cleves}
          [m. 1540]
\indvdldef[\ftblank]{CatherineHoward}{Catherine Howard}
          [m. 1540]
\indvdldef[\ftblank]{CatherineParr}{Catherine Parr}
          [m. 1543]

\mrrgdef{HenryVIIIWives}{}{HenryVIII}{CatherineofAragon,Elizabeth,%
  AnneBoleyn,Jane,AnneofCleves,CatherineHoward,CatherineParr}
\fbox{\usebox{\HenryVIIIWives}}
```



4.3 Layout and connecting in a same generation

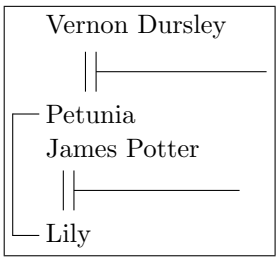
It is not a good idea to put everything in a single family tree.

For example, see King Henry VIII and his wives. Catherine of Aragon, his first wife was actually a wife of Henry's brother, Arthur. If we put King's siblings to this tree, how would

it be looked? It's just ugly and hard to understand in a glance. Let's think more using an example from 3.2 "The order of connecting multiple boxes" again.

How can we represent the Petunia – Lily sisters tree including their husbands. As a first step, define two marriage boxes, and then define the sibling box.

```
\indvdldef{Petunia}{Petunia}
\indvdldef[\blank]{Vernon}{Vernon Dursley}[\haschild]
\mrrgdef{Dursleys}{Vernon}{Petunia}{}

\indvdldef{Lily}{Lily}
\indvdldef[\blank]{James}{James Potter}[\haschild]
\mrrgdef{Potters}{James}{Lily}{}

\sblngdef{sis}{Dursleys,Potters}
\fbox{\usebox{\sis}}
```

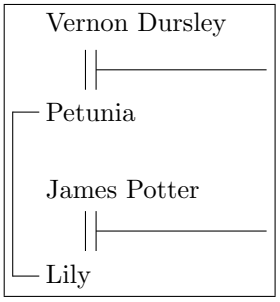


Why is this tree so ugly? There are three points to consider.

1. The position of two double lines differ from each other.
2. The length of a line to their child differs too. If we connected the child, the ugliness would be improved.
3. James interrupts into between Petunia and Lily. It makes the understandability worse.

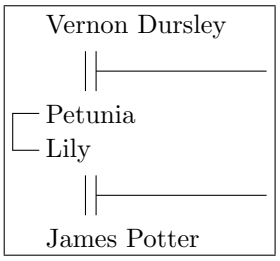
On fixing the first point, the position of the double line, the second point will be fixed automatically. The solution is the one already suggested in 3.2 "The order of connecting multiple boxes", set the width of Lily box to Petunia's. For the third point, the position of James, how about expanding the space as a first step?

```
...
\savebox{\boxA}{\hbox{Petunia}}
\indvdldef{Lily}{\hbox to \wd\boxA{Lily}}
\indvdldef[\blank]{James}{James Potter}[\haschild]
\mrrgdef{Potters}{James}{Lily}{}[\dimexpr\wd\Vernon - \wd\James\relax]

\sblngdef{sis}{Dursleys,ivali,Potters}
\fbox{\usebox{\sis}}
```



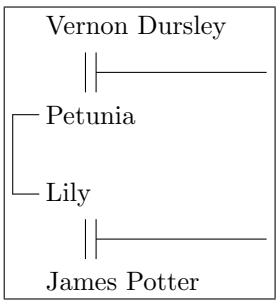
Even spreading the space wider, James is still interrupting those two sisters. Does it look better? If we want more, the last way is to switch the position of James and Lily.

```
...
\mrrgdef{Potters}{}{Lily}{James}[\dimexpr\wd\Vernon - \wd\James\relax]

\sblngdef{sis}{Dursleys,Potters}
\fbox{\usebox{\sis}}
```



Moreover spreading the blank is a good option.

```
...
\sblngdef{sis}{Dursleys,ivali,Potters}
\fbox{\usebox{\sis}}
```



Is this best looking? The easiness of looking is subject to one's opinion or taste. Personally I feel resistance in the order of husband and wife. But also I admit that as long as the main purpose of this tree is to represent those sisters, this position of James is not bad.

Jump to next section

4.4 Implementation

Customization

\ftmrrgboxcfg
\mrrgboxcfg

```
\newlength{\ft@mrrgline@sep}
\setlength{\ft@mrrgline@sep}{4pt}
\newlength{\ft@mrrgline@sp}
\setlength{\ft@mrrgline@sp}{.5\ft@unit}
\newlength{\ft@mrrgline@length}
\setlength{\ft@mrrgline@length}{1.5\ft@unit}
\newcommand{\ftmrrgboxcfg}[3]{% sep space length
  \ft@ifempty{#1}{}{%
    \setlength{\ft@mrrgline@sep}{#1}%
  }%
  \ft@ifempty{#2}{}{%
    \setlength{\ft@mrrgline@sp}{#2}%
  }%
  \ft@ifempty{#3}{}{%
    \setlength{\ft@mrrgline@length}{#3}%
  }%
}
\ft@alias{mrrgboxcfg}
```

Parsing

```
\newcommand{\ft@mrrg@parse}[1]{% spouse-list
  \global\ft@height=0pt%
  \global\ft@width=0pt%
  \global\ft@box@has@malelinefalse%
  \@for\ft@iter:=#1\do{%
    \ifx\ft@iter\empty\else%
      \xdef\ft@spouse{\ft@iter}%
      \@ifundefined{\ft@iter ival}{%
        \@ifundefined{\ft@spouse haschild}{}{%
          \global\ft@box@has@malelinetrue%
        }%
        \setlength{\ft@len}{\wd\@nameuse{\ft@spouse}}%
        \ifdim\ft@width<\ft@len%
          \global\ft@width=\ft@len%
        \fi%
        \ft@dbgmsg{\ft@spouse, W \the\wd\@nameuse{\ft@spouse},%
          H \the\ht\@nameuse{\ft@spouse},%
          D \the\dp\@nameuse{\ft@spouse}}%
        \global\advance\ft@height \dimexpr\ft@mrrgline@length%
          + 2\ft@mrrgline@sp\relax%
        \ft@dbgmsg{\ft@spouse, H \the\ft@height}%
      }{}%
      \global\advance\ft@height \dimexpr\ht\@nameuse{\ft@spouse}%
        + \dp\@nameuse{\ft@spouse}\relax%
      \ft@dbgmsg{\ft@spouse, h H \the\ft@height}%
    \fi%
  }%
  %
  \ifft@box@has@maleline%
```

```
    \global\advance\ft@width \ft@namebox@maleline@length%
  \fi%
  %
  \global\ft@depth=\dp\@nameuse{\ft@spouse}%
  \global\advance\ft@height \dimexpr -2\ft@mrrgline@sp%
    - \ft@mrrgline@length - \ft@depth\relax%
  \ft@dbgmsg{final H \the\ft@height, D \the\ft@depth}%
}
```

The double line

\ft@mrrg@line

```
\newlength{\ft@mrrg@chlen}
\newcommand{\ft@mrrg@line}[5]{% box-name spouse cx sp length
  \ft@x=#3%
  \global\advance\ft@height -#4%
  \@tempskipb=\dimexpr\ft@mrrgline@sep/2\relax%
  \edef\@y{\strip@pt\ft@height}%
  \@ifundefined{#2private}{%
    \ft@len=#5\relax%
    \edef\@l{\strip@pt\ft@len}%
    \put(\strip@pt\dimexpr\ft@x - \@tempskipb, \@y){\line(0,-1){\@l}}%
    \put(\strip@pt\dimexpr\ft@x + \@tempskipb, \@y){\line(0,-1){\@l}}%
  }{%
    % this divisor should match the delta_y for multiput
    \ft@len=#5\relax%
    \ft@len=\dimexpr\ft@len/2 + .5pt\relax%
    \@tempcnta=\dimexpr\ft@len/65536\relax%
    \multiput(\strip@pt\dimexpr\ft@x - \@tempskipb, \@y)%
      (0,-2){\@tempcnta}{\line(0,-1){.5}}%
    \multiput(\strip@pt\dimexpr\ft@x + \@tempskipb, \@y)%
      (0,-2){\@tempcnta}{\line(0,-1){.5}}%
  }%
  \@ifundefined{#2haschild}{}{%
    \ft@len=#5\relax%
    \ft@y=\dimexpr\ft@height - \ft@len/2\relax%
    \put(\strip@pt\dimexpr\ft@x + \@tempskipb,\strip@pt\ft@y)%
        {\line(1,0){\strip@pt\ft@mrrg@chlen}}%
    \ft@dbgplot{\strip@pt\ft@x,\strip@pt\ft@y}%
    \ft@namexdefstrip{#1#2mrrgCY}{\ft@y}%
  }%
  \ft@len=#5\relax%
  \@tempskipa=#4\relax%
  \global\advance\ft@height \dimexpr -\ft@len - \@tempskipa\relax%
  \ft@dbgmsg{line #2 H \the\ft@height}%
}
```

Layout the names

\ft@mrrg@name

```
\newcommand{\ft@mrrg@name}[2]{% box-name individual-name
  \global\advance\ft@height -\ht\@nameuse{#2}%
  \put(0,\strip@pt\ft@height){\usebox{\@nameuse{#2}}}%
```

```
  \ft@dbgframe[0,\strip@pt\ft@height]%
              {\strip@pt\wd\@nameuse{#2},\strip@pt\ht\@nameuse{#2}}%
  %
  \@ifundefined{#2hasmaleline}{}{%
    \ft@x=\@nameuse{#2nameX}pt%
    \ft@y=\dimexpr\ft@height + \@nameuse{#2nameCY}pt\relax%
    \ft@len=\dimexpr\ft@width - \@nameuse{#2nameX}pt%
      %- \ft@namebox@maleline@sp%
      \relax%
    \put(\strip@pt\ft@x,\strip@pt\ft@y){\line(1,0){\strip@pt\ft@len}}%
    \ft@namexdefstrip{#1#2nameCY}{\ft@y}%
  }%
  %
  \@ifundefined{#2hascmark}{}{%
    \ft@len=\dimexpr\ft@height + \@nameuse{#2nameCY}pt\relax%
    \ft@namexdefstrip{#1#2nameCY}{\ft@len}%
    \ft@dbgplot{0,\strip@pt\ft@len}%
  }%
  \global\advance\ft@height -\dp\@nameuse{#2}%
  \ft@dbgmsg{name #2 H \the\ft@height}%
}
```

4.4.1 Layout and connect the individuals — core

\ft@mrrg@spouse

```
\newlength{\ft@mrrg@ival}
\newcommand{\ft@mrrg@spouse}[2]{% box-name list
  \global\ft@mrrg@ival=0pt%
  \@for\ft@iter:=#2\do{%
    \@ifundefined{\ft@iter ival}{%
      \@tempskipa=\dimexpr\ft@mrrgline@length + \ft@mrrg@ival\relax%
      \if@tempswa%
        \ft@mrrg@name{#1}{\ft@iter}%
        \ft@mrrg@line{#1}{\ft@iter}{\ft@xx}{\ft@mrrgline@sp}%
                     {\@tempskipa}%
      \else%
        \ft@mrrg@line{#1}{\ft@iter}{\ft@xx}{\ft@mrrgline@sp}%
                     {\@tempskipa}%
        \ft@mrrg@name{#1}{\ft@iter}%
      \fi%
      \global\ft@mrrg@ival=0pt%
    }{%
      \global\advance\ft@mrrg@ival%
        \dimexpr\ht\@nameuse{\ft@iter}%
        + \dp\@nameuse{\ft@iter}\relax%
    }%
  }%
}
```

4.4.2 Marriage box — interface

\ftmrrgdef
\mrrgdef

```
\NewDocumentCommand{\ftmrrgdef}{mmmmO{0pt}}{%
```

```
  % box-name spouse-listA oneself spouse-listB [xline]
  \ft@xx=\@nameuse{#3nameCX}pt\relax%
  \ft@mrrg@parse{#2,#3,#4}%
  %
  \advance\ft@width #5%
  \global\ft@mrrg@chlen=\dimexpr\ft@width - \ft@xx%
    - \ft@mrrgline@sep/2\relax%
  %
  \ft@theight=\ft@height%
  \ft@newnamebox{#1}{%
    \edef\@w{\strip@pt\ft@width}%
    \edef\@h{\strip@pt\ft@height}%
    \unitlength=1pt\relax% coordinates are in pt
    \begin{picture}(\@w,\@h)%
      \ft@dbgframe{\@w,\@h}%
      %
      \ft@ifempty{#2}{}{%
        \@tempswatrue%
        \ft@mrrg@spouse{#1}{#2}%
      }%
      %
      \ft@mrrg@name{#1}{#3}%
      \@ifundefined{#1#3nameCY}{}{%
        \ft@len=\@nameuse{#1#3nameCY}pt\relax%
        \ft@dbgplot{1,\strip@pt\ft@len}%
        \ft@namexdefstrip{#1nameCY}{\ft@len}%
      }%
      \@ifundefined{#3hascmark}{}{%
        \ft@namexdef{#1hascmark}{\@nameuse{#3hascmark}}%
      }%
      %
      \ft@ifempty{#4}{}{%
        \@tempswafalse%
        \ft@mrrg@spouse{#1}{#4}%
      }%
    \end{picture}%
  }%
  \ft@nameboxsz{#1}{\ft@theight}{\ft@depth}%
}
\ft@alias{mrrgdef}
```

5 Library (ft-lib.dtx)

While this is the last section of this document, ft-lib.dtx is located top of sty.

5.1 Implementation

```
\RequirePackage{relsize}
\RequirePackage{xparse}
```

\ft@unit Represents a width of a single character.

\if@tate Supports Japanese vertical mode (jlreq.cls). Other than Japanese, this dummy \if@tate is always false.

```
\global\newlength{\ft@unit}%
\@ifundefined{zw}{%
  \ifx\kanjiskip\@undefined% not pTeX (\kanjiskip is its primitive)
    \global\ft@unit=1em%
  \else%
    \global\ft@unit=1zw%
  \fi%
}{%
  \global\ft@unit=1\zw% jlreq (any engine), LuaTeX-ja
}%
\@ifundefined{if@tate}{%
  \global\newif\if@tate%
}{}%
```

\ft@alias Makes an alias with prefix-less (ft). I don't know why such many \expandafters are necessary.

```
\newcommand{\ft@alias}[1]{% ft-less-name
  \@ifundefined{#1}{%
    \global\expandafter\expandafter\expandafter%
      \let\expandafter\expandafter\csname#1\endcsname\expandafter%
      =\csname ft#1\endcsname%
  }{%
    \message{skip alias to ft#1}%
  }%
}
```

5.1.1 utility

\ftymd \ftymd{⟨*year*⟩} [⟨*month*⟩] [⟨*date*⟩]

\ymd Arranges and prints year-month-date. Japanese vertical mode is supported. Assumes to be used in ⟨*additional info*⟩ of \indvdldef.

```
\NewDocumentCommand{\ftymd}{moo}{% year [month] [date]
  \if@tate%
  \tatechuyoko{#1}%
  \IfValueT{#2}{\tatechuyoko{#2}}%
  \IfValueT{#3}{\tatechuyoko{#3}}%
  \else%
  #1%
  \IfValueT{#2}{\slash#2}%
  \IfValueT{#3}{\slash#3}%
  \fi%
```

```
}
\ft@alias{ymd}
```

`\ftundef` `\ftundef{`⟨*box name*⟩`}`

Undefines all internal control sequencees who has ⟨*box name*⟩ in its name. For this macro, we don't define the prefix-less alias.

```
\newcommand{\ftundef}[1]{%
  \@for\ft@iter:=#1\do{%
    \global\expandafter\let\csname\ft@iter\endcsname=\relax%
    \global\expandafter\let\csname\ft@iter nameCY\endcsname=\relax%
    \global\expandafter\let\csname\ft@iter mrrgCY\endcsname=\relax%
    \global\expandafter\let\csname\ft@iter hasmaleline\endcsname%
      =\relax%
  }%
}
```

5.1.2 internal library

global variable

```
\newif\ifft@box@has@maleline
\newif\ifft@widest@has@no@maleline
\newlength{\ft@len}
\newlength{\ft@width}
\newlength{\ft@height}
\newlength{\ft@depth}
\newlength{\ft@theight}
\newlength{\ft@x}
\newlength{\ft@xx}
\newlength{\ft@y}
\newlength{\ft@yy}
```

utility function

`\ft@ifempty` `\ft@ifempty{`⟨*arg*⟩`}{`⟨*true*⟩`}{`⟨*false*⟩`}`

When ⟨*arg*⟩ is empty, executes ⟨*true*⟩. Otherwise runs ⟨*false*⟩.

```
\newcommand{\ft@ifempty}[1]{% arg {true} {false}
  \if\relax\detokenize{#1}\relax%
    \expandafter\@firstoftwo%
  \else%
    \expandafter\@secondoftwo%
  \fi%
}
```

`\ft@namexdef` `\ft@namexdefstrip`

```
\newcommand{\ft@namexdef}[2]{% name value
  \global\expandafter\edef\csname#1\endcsname{#2}%
  %\expandafter\xdef\csname#1\endcsname{#2}%
  \ft@dbgmsg{xdef #1 #2}%
}
\newcommand{\ft@namexdefstrip}[2]{% name length
  \ft@namexdef{#1}{\strip@pt#2}%
}
```

```
\ft@savenamebox \newcommand{\ft@savenamebox}[2]{% name contents
 \ft@newnamebox   \expandafter\savebox\expandafter{\csname#1\endcsname}{#2}%
                 }
                 \newcommand{\ft@newnamebox}[2]{% name contents
                   \global\expandafter\newsavebox\expandafter{\csname#1\endcsname}%
                   \ft@savenamebox{#1}{#2}%
                 }
  \ft@nameboxsz  \newcommand{\ft@nameboxsz}[4][]{% [width] name ht-value dp-value
      \ft@boxsz    \ft@ifempty{#1}{}{%
                     \expandafter\wd\csname#2\endcsname=#1\relax%
                   }%
                   \expandafter\ht\csname#2\endcsname=#3%
                   \expandafter\dp\csname#2\endcsname=#4%
                   \ft@dbgmsg{#2 box, W \the\wd\@nameuse{#2}, H \the\ht\@nameuse{#2},%
                     D \the\dp\@nameuse{#2}}%
                 }
                 \newcommand{\ft@boxsz}[3]{% box ht-value dp-value
                   \expandafter\ht#1=#2%
                   \expandafter\dp#1=#3%
                   \ft@dbgmsg{#2 box, W \the\wd#1, H \the\ht#1, D \the\dp#1}%
                 }
\ft@list@append  \newcommand{\ft@list@append}[2]{% list element
                   \ifx#1\empty%
                   \xdef#1{#2}%
                   \else%
                   \xdef#1{#1,#2}%
                   \fi%
                 }
                 \iffalse
                 \newcommand{\ft@detokenize}[1]{%
                   {%
                     \escapechar=‘\ %
                     \catcode‘\ =9%
                     %\string#1%
                     \detokenize{#1}%
                   }%
                 }
                 \fi
```

debugging

```
      \ifftdbg  \newif\ifftdbg
    \ft@dbgmsg  \newcommand{\ft@dbgmsg}[1]{%
  \ft@dbgframe    \ifftdbg%
   \ft@dbgplot      \message{#1}%
    \ft@dbgbox    \fi%
                 }
                 \newcommand{\ft@dbgframe}[2][0,0]{% [point x,y] width,height
                   \ifftdbg%
                     \put(#1){\framebox(#2){}}%
                   \fi%
                 }
                 \newcommand{\ft@dbgplot}[2][]{%
```

```
  \ifftdbg%
    \put(#2){\circle{1}{\tiny#1}}%
  \fi%
}%
\newcommand{\ft@dbgbox}[2][]{% [extra-put] box
  \ifftdbg\begingroup%
    \unitlength=1pt\relax%
    \ft@len=\dimexpr\ht#2 + \dp#2\relax%
    \begin{picture}(\strip@pt\wd#2,\strip@pt\ht#2)(0,-\strip@pt\dp#2)%
      \ft@dbgframe[0,-\strip@pt\dp#2]{\strip@pt\wd#2,\strip@pt\ft@len}%
      %\put(0,0){{\tiny\ft@detokenize{#2}}}%
      %\put(0,-\strip@pt\dp#2){\usebox{#2}}%
      \put(0,0){\usebox{#2}}%
      #1%
    \end{picture}%
  \endgroup\fi%
}
```