schemata — Generic package to aid construction of topical categories*

Charles P. Schaum†

Released 2021/02/27

Abstract

The schemata package helps the creation of topical outlines that illustrate the breakdown of concepts and categories in academic texts from the late medieval to early modern periods.

Contents

1 Introduction **1**

2 Usage **2**
2.1 Loading and Options . . . 2
2.2 Macro Overview 2
2.2.1 \schemabox 2
2.2.2 Delimiters 3
2.2.3 \schema 3
2.2.4 \Schema 4
2.3 Romancing the \schema . 6
2.4 Tutorial 7
2.4.1 Starting Off Basic 7
2.4.2 *Loci* 101 8
2.4.3 Going Big 9
2.4.4 Big Groups 19
2.4.5 Open and Closed . 19
2.5 Final features 22

3 Implementation **23**
3.1 Internal Variables 23
3.2 Package Options 24
3.3 Macros 24

4 Change History **31**

5 Index **32**

1 Introduction

This package uses boxes and math mode to typeset *schemata* (plural of τό σχῆμα or *schema*, meaning *form*, *shape*, *appearance*, etc.). One sees them in academic literature from at least the seventeenth through the nineteenth centuries.[1]

Packages like *TikZ*, PSTricks, METAPOST, or other solutions have advantages over this one, especially for those seeking a top-to-bottom diagram.[2] Yet these packages may present challenges if one has to implement both open *and* closed braces in a schema, which math mode allows.

*This file describes version 1.4, last revised 2021/02/27.

†E-mail: charles[dot]schaum@comcast.net

[1]Books that use this package include: Löhe, *The Pastor* [*Der evangelische Geistliche*] (St. Louis, 2015) and Schaum and Collver, *Breath of God, Yet Work of Man* (St. Louis, 2019).

[2]For example: H. Dembowski, *Einführung in die Christologie* (Darmstadt, 1993), 146.

2 Usage

2.1 Package Loading and Options

The schemata package is a minimal "wrapper" for math mode. It can be used with LaTeX or with "generic" formats, including Plain TeX, Eplain, and Lollipop.[3]

For LaTeX invoke:	`\usepackage[`⟨*options*⟩`]{schemata}`
For generic use:	`\input␣schemata.sty`

`\schemataLaTeX` Normally, schemata uses generic TeX macros if the format is not LaTeX 2ε. When using a LaTeX-like format with a different name than `LaTeX2e`, one could insert the following before `\usepackage{schemata}`:

```
\edef\schemataLaTeX{\fmtname}
```

Yet this is usually unneeded. Normally we want `\schemataLaTeX` to be undefined before `schemata.sty` is loaded to get the default value `LaTeX2e`. We recommend not using this macro unless you know what you are doing.

options LaTeX users can choose one among four package options: `braces`, `brackets`, `parens`, and `groups`. These set the defaults for the delimiters. If no options are chosen, the default is `braces`.

2.2 Macro Overview

One can describe schemata as a grouping of boxes that contain content, whose relationships are demonstrated by delimiters. We start with the boxes and their content. Subsequently, we deal with the delimiters, then later, the manner of grouping and arrangement, as well as tweaks and tutorials.

2.2.1 Containers: `\schemabox`

`\schemabox` Schemata contain vertically-centered lists of material in inner vertical mode. When in a `\schema` or a `\Schema` (see below), a `\schemabox` stacks one or more lines of `\hbox`-enclosed text in a `\vbox`. It redefines the macro `\\` to close the current `\hbox` and begin a new one, with some options that can be modified (Section 2.3).

`\schemabox[`⟨*width*⟩`]{`⟨*text*⟩`}`

The ⟨*width*⟩ of a `\schemabox` is a dimension, e.g., `3cm`. No text wrapping (as in a `\parbox`) takes place. If there is more than one line of text, each line of ⟨*text*⟩ must be terminated explicitly by `\\`, except the final line. Usually, the first line of a `\schemabox` inserts a `\strut` for aesthetic reasons.

When not in internal vertical mode, `\schemabox` ignores ⟨*width*⟩, does not redefine `\\`, and prints its argument as text: `\schemabox{line~1\\ line~2}` line 1
line 2. This helps prevent errors.

[3] ConTeXt does not like the way that schemata nests math-mode expressions within boxes.

2.2.2 Delimiters

\DoBraces
\DoBrackets
\DoParens
\DoGroups

Both generic TEX and LATEX users can use these four macros to set or change the type of delimiters. In both generic TEX and LATEX, the default delimiter is braces. `\DoBraces`, `\DoBrackets`, `\DoParens`, and `\DoGroups` do the same thing as the respective package options, except they also change the delimiters when used before `\schema` and `\Schema`. They remain in force until the end of a scope:

$$\mathrm{a}\left\{\begin{matrix}\mathrm{b}\\ \mathrm{c}\end{matrix}\right\}\mathrm{d} \qquad \mathrm{a}\left[\begin{matrix}\mathrm{b}\\ \mathrm{c}\end{matrix}\right]\mathrm{d} \qquad \mathrm{a}\left(\begin{matrix}\mathrm{b}\\ \mathrm{c}\end{matrix}\right)\mathrm{d} \qquad \mathrm{a}\left\langle\begin{matrix}\mathrm{b}\\ \mathrm{c}\end{matrix}\right\rangle\mathrm{d}$$

Additionally, these macros can change the delimiter style within a schema. See Section 2.5, as well as the example below:

```
\DoBrackets
\Schema{0ex}{2.4ex}
  {\schemabox{a}}
  {\DoParens\Schema[close]{0ex}{2.3ex}
    {\schemabox{b\\c}}
    {\schemabox{d}}
}
```

$$\mathrm{a}\left[\begin{matrix}\mathrm{b}\\ \mathrm{c}\end{matrix}\right)\mathrm{d}$$

One can add new types by using eligible math-mode delimiters, e.g.:

```
\makeatletter
\newcommand*{\DoVerts}%
  {\let\@schemata@LD\bracevert%
   \let\@schemata@RD\bracevert}
\makeatother
\DoVerts
\Schema{0ex}{5ex}
  {\vskip0.6ex\schemabox{a}}
  {\Schema[close]{0ex}{5ex}
    {\vskip0.4ex\schemabox{b\\c\\d\\e}}
    {\vskip0.6ex\schemabox{{\kern0.1em}f}}
}
```

$$\mathrm{a}\left|\begin{matrix}\mathrm{b}\\ \mathrm{c}\\ \mathrm{d}\\ \mathrm{e}\end{matrix}\right|\mathrm{f}$$

2.2.3 Leaf Nodes: `\schema`

\schema

A "simple" schema has a left-hand side with vertically-centered vertical material, a brace, and a right-hand side with vertically-centered vertical material:

`\schema[`⟨*type*⟩`]{`⟨*left side*⟩`}{`⟨*right side*⟩`}`

The ⟨*left side*⟩ and ⟨*right side*⟩ are vertical material in order to allow a `\smallskip` or other vertical adjustment as needed.

The ⟨*type*⟩ of a schema is `open` (the delimiter opens to the right) by default:

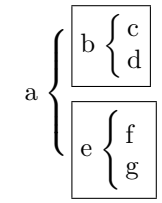
```
\schema
  {\schemabox{a}}
  {\schemabox{b\\c}}
```

$$\mathrm{a}\left\{\begin{matrix}\mathrm{b}\\ \mathrm{c}\end{matrix}\right.$$

Any value of ⟨*type*⟩ other than the exact string `open` makes a "closed" schema (the delimiter opens to the left):

```
\schema[closed]
  {\NudgeSB\schemabox{b\\c}}
  {\schemabox{a}}
```

Using `\NudgeSB` above added a kern of `0.2em` at the right of the `\schemabox` to offset an automatic kern of `-0.2em` that normally pulls the brace slightly closer to the left-hand side when it opens to the right. We cover such tweaks in Section 2.3.



In practice, `\schema` does not nest, so it is only useful for the right-hand "leaves" of a large schema. That makes formatting the "leaves" much faster. Thus, the `\schema` macro is used only in the framed sub-schemata at right.

The automatic sizing of `\schema` changes, depending on the height, depth, and even context of the letters. This can look ugly if uniformity is desired. Use `\Schema` (next section) to enforce uniform schemata.

2.2.4 Branches and Root: `\Schema`

`\Schema` The "complex" form of a schema also has a left-hand side with vertically-centered vertical material, a brace, and a right-hand side of vertically-centered vertical material, along with two arguments that adjust the layout:

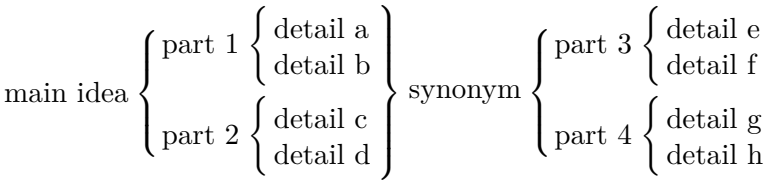
`\Schema[`⟨*type*⟩`]{`⟨*adjust*⟩`}{`⟨*size*⟩`}{`⟨*left side*⟩`}{`⟨*right side*⟩`}`

The ⟨*type*⟩ is `open` by default. As above, any other ⟨*type*⟩ except the exact string `open` will make it a "closed" schema. Both ⟨*adjust*⟩ and ⟨*size*⟩ are dimensions. We recommend expressing them as `ex`. This allows for easier scaling of the schema when changing the font size. Here is how to set ⟨*adjust*⟩:[4]

negative	left side and delimiter up	right side down
positive	left side and delimiter down	right side up

Set the delimiter ⟨*size*⟩ to be a scaled value of ex just a bit larger than the number of lines of text that the delimiter spans.

By using `\Schema` to adjust the delimiter height and centering, one can bypass the shortcomings of `\schema`, but at the cost of time. One has to traverse the schema at least twice to get the desired layout. `\Schema` lets one produce multiple schemata with the same look. This method allows complex layouts:



[4]Instead of setting ⟨*adjust*⟩, one could put vertical skips either before or after `\schemabox`, `\schema`, or `\Schema`. Yet using braces as delimiters tends to draw material toward the center cusp, where ⟨*adjust*⟩ keeps that centered look while allowing some adjustments.

The source for that complex schema looks like:

```
\Schema[close]{0ex}{5.1ex}
{
  \Schema{0.1ex}{3.8ex}
  {\SwitchSB\schemabox{main idea}}
  {
    \schema{\schemabox{part 1}}
      {\SwitchSB\NudgeSB\schemabox{detail a\\detail b}}\smallskip
    \schema{\schemabox{part 2}}
      {\SwitchSB\NudgeSB\schemabox{detail c\\detail d}}
  }
}
{
  \Schema{0ex}{3.8ex}
  {\schemabox{\,synonym}}
  {
    \schema{\schemabox{part 3}}
      {\SwitchSB\schemabox{detail e\\detail f}}\smallskip
    \schema{\schemabox{part 4}}
      {\SwitchSB\schemabox{detail g\\detail h}}
  }
}
```

Both `\schema` and `\Schema` will stack vertically if set sequentially as paragraphs in running text:

```
\schema
  {\schemabox{a}}
  {\schemabox{b\\c}}

\schema
  {\schemabox{d}}
  {\schemabox{e\\f}}
```

a { b / c

d { e / f

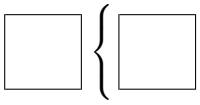
They can be on a line of text: Does this { look / ugly?

Certainly, one need not use a `\schemabox` in either `\schema` or `\Schema`. For example, we make a macro `\Box` below to create one square centimeter of content:

```
\def\Box{%
  \hbox{%
    \vrule%
    \vbox to 1cm{\hrule\hbox to 1cm{\hfil}\vfil\hrule}%
    \vrule%
  }%
}
```

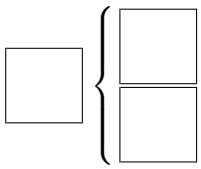
Now we begin with the trivial example of one `\Box` on each side of the delimiter:

```
\schema{\Box}{\Box}
```



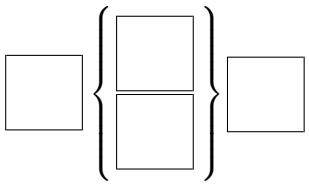
This example is more complex, showing how each side stacks `\Box`es vertically:

```
\schema{\Box}{\Box\Box}
```



Finally we use `\Schema` to get a schema that is both open and closed:

```
\Schema{-0.2ex}{0.9cm}
{\Box}
{
  \Schema[close]
    {-0.2ex}{0.9cm}
  {\Box\hbox{\Box\kern0.2em}}
  {\Box}
}
```



A kern of `0.2em` was added above to compensate for the automatic kern of `-0.2em`, as Section 2.3 explains in more detail. If not expressed in `ex` height, ⟨*size*⟩ should be slightly less than half the height of the contents, e.g., `0.9cm` for a height of `2cm`.

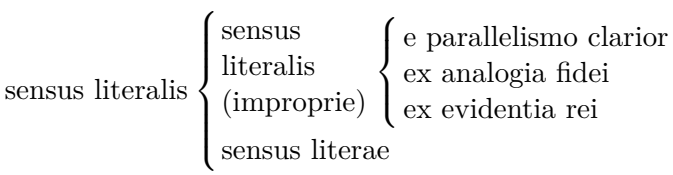
2.3 Romancing the `\schema`

`\LCschema` `\UCschema`

By default, a `\schemabox` adds a `\strut` to the first line because the topics in a schema often start with a capital letter. The `\strut` causes the delimiter of a `\schema` to have the proper size.

If the first letter is not a capital or if the text seems a little off-center, you can turn off this default feature of `\schemabox` by placing `\LCschema` immediately before `\schemabox`. `\LCschema` will prevent all subsequent uses of `\schemabox` from adding `\strut` until you restore the default behavior with `\UCschema`, also best placed before the intended `\schemabox`. Here is an example where an entire schema is in lowercase, so we change the look of the whole thing:[5]

```
\LCschema
\Schema{0.1ex}{4.8ex}
{\hbox{sensus literalis}}
{
  \schema{\schemabox{sensus\\literalis\\(improprie)}}
         {\schemabox{e parallelismo clarior\\
             ex analogia fidei\\ex evidentia rei}}
          \smallskip\schemabox{sensus literae}
}
\UCschema
```



[5] Based on axioms in August Pfeiffer, *Thesaurus Hermeneuticus* (Frankfurt am Main, 1698).

\SwitchSB The macro \SwitchSB is a per-use toggle. It causes a particular \schemabox to do the opposite of whatever \LCschema and \UCschema call for. It should be placed immediately before the \schemabox to be affected and its effect is reset when that particular \schemabox terminates.

Note, however, that mixing lowercase and uppercase-styles of \schemabox may put parts of a schema slightly off-center, meaning that one must ⟨*adjust*⟩ a \Schema by a tenth of an ex, give or take. Also remember that one can add \strut as needed to make manual adjustments.

\NudgeSB The macro \NudgeSB is another "per-use" macro that causes a particular \schemabox to add a default 0.2em kern at the end of every line of text, then is reset thereafter. It "corrects a corrective."

\NudgeSB is meant to be used on the left-hand side of a closed \schema or \Schema. Both macros insert a kern of -0.2em to draw the cusp or flexion point of the delimiter closer to the left-hand side. This corrects the spacing of delimiters that open to the right. When a delimiter opens to the left, the kern may be needed if there is punctuation, or it may throw off the spacing.

\SBNudgeFactor This macro is the kern used by \NudgeSB to make its corrective. Sometimes you feel like a nudge, sometimes you don't, and sometimes you just want a little nudge. We used the example below on page 3 before the schema with two braces, all in a group to localize any changes:

```
\renewcommand\SBNudgeFactor{\kern0.08em}
```

2.4 Tutorial

Now that we have explained what all the macros are supposed to do, let's take a journey together in establishing and practicing a methodology for creating general forms of schemata.

2.4.1 Starting Off Basic

Let's ignore pretty much everything that we learned so far and attempt to typeset a schema with the following:

```
1 \schema{a}{b\\c}
```

a $\left\{\begin{array}{l} \\ \text{b} \\ \text{c} \end{array}\right.$

Oh dear, that went badly. Oh, wait! Schemata hold internal vertical lists. That weird \schemabox thing handles just that case:

```
1 \schema
2   {\schemabox{a}}
3   {\schemabox{b\\c}}
```

a $\left\{\begin{array}{l} \text{b} \\ \text{c} \end{array}\right.$

Now we are getting somewhere! But if we do not have a "big" side we get:

```
1 \schema
2   {\schemabox{a}}
3   {\schemabox{b}}
```

a $\left\{\begin{array}{l} \text{b} \end{array}\right.$

When there is no "big" side of a schema, perhaps use inline math mode:

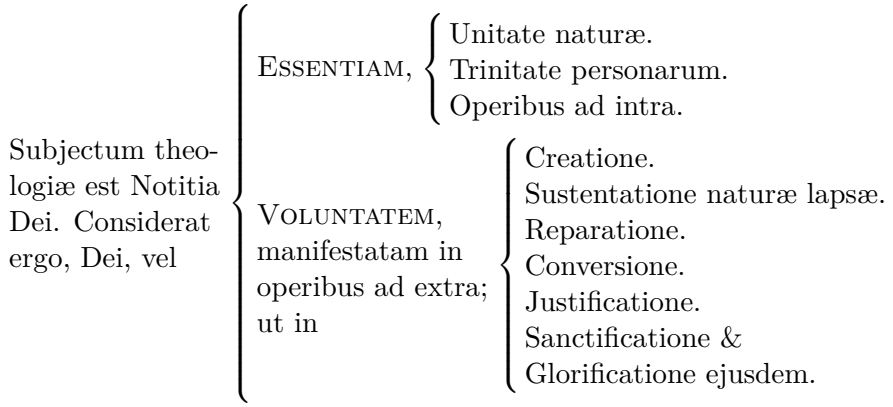
```
\(\hbox{a}\left\{\hbox{\strut gib}\right.\)
```

a $\left\{\text{gib}\right.$

2.4.2 *Loci* 101

We move on from trivial examples to several real-world examples based on published material. Let's try a few examples from *Loci Theologici* by Martin Chemnitz. We begin by using only `\schema`:

```
\schema
{
  \schemabox{Subjectum theo-\\
  logi\ae{} est Notitia\\
  Dei. Considerat\\
  ergo, Dei, vel}
}
{
  \schema
  {
    \schemabox{\textsc{Essentiam},}
  }
  {
    \schemabox{Unitate natur\ae{}.\\
    Trinitate personarum.\\
    Operibus ad intra.}
  }
  \schema
  {
    \schemabox{\textsc{Voluntatem},\\
    manifestatam in\\
    operibus ad extra;\\
    ut in}
  }
  {
    \schemabox{Creatione.\\
    Sustentatione natur\ae{} laps\ae{}.\\
    Reparatione.\\
    Conversione.\\
    Justificatione.\\
    Sanctificatione \&\\
    Glorificatione ejusdem.}
  }
}
```



This is not what we want; `\schema` works for the "leaves" on the right, but not for the "root" on the left. The brace adjusts to the entire right-hand side.

Before we address the brace, we adjust the spacing, starting from the "leaves" at right, going to the "root" on the left. We add a `\smallskip` after a `\schema` to space out the "leaves":[6]

```
17 }\smallskip
```

We have two `\schema` "leaves" and one "root," so we only change one `\schema` into a `\Schema`. We count the lines of text, estimate, then revise. Below we have 8–9 lines of text from "Essentiam" to "ut in." We estimate ⟨*size*⟩ at `8.5ex` and ⟨*adjust*⟩ at `0ex`. The large brace is too low, so we ⟨*adjust*⟩ to `-1ex`, raising the left side and the delimiter, while lowering the right. We then refine ⟨*size*⟩ to `8.7ex`.[7]

```
1 \Schema{-1ex}{8.7ex}
```

After those two line changes, we have the finished schema that now looks like it is supposed to appear:

Subjectum theologiæ est Notitia Dei. Considerat ergo, Dei, vel
- Essentiam,
  - Unitate naturæ.
  - Trinitate personarum.
  - Operibus ad intra.
- Voluntatem, manifestatam in operibus ad extra; ut in
  - Creatione.
  - Sustentatione naturæ lapsæ.
  - Reparatione.
  - Conversione.
  - Justificatione.
  - Sanctificatione &
  - Glorificatione ejusdem.

2.4.3 Going Big

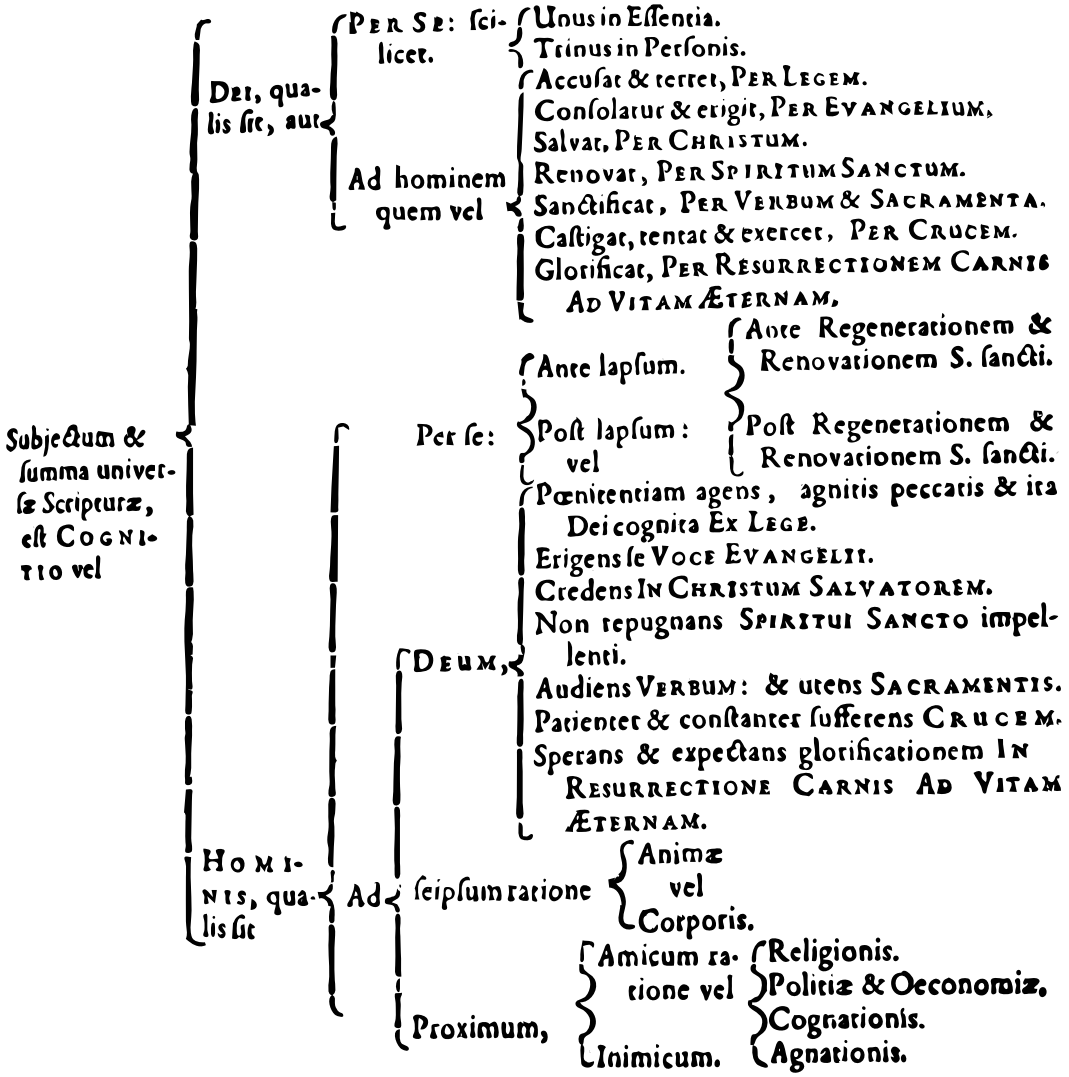
Thus far, we have dealt with many trivial examples. We have amassed a significant body of knowledge:

1. We usually use `\schemabox` for the contents of a schema.
2. Schemata usually "open" from left to right, from "root" to "leaves."
3. We typeset "leaves" with `\schema` to save time.
4. We typeset other parts with `\Schema`.
5. We adjust spacing and delimiter size by working from the "leaves" to the "root" to minimize the number of corrective passes.
6. We may need to consider differences between LaTeX and Plain TeX when using `\vskip`, `\smallskip`, etc., as well as `\newbox`, which is an `\outer` macro in Plain TeX. These differences can cause unexpected errors.
7. We may need to use the tweaking macros `\UCschema`, `\LCschema`, `\SwitchSB`, and `\NudgeSB`.

[6] Using `\vskip` in Plain TeX starts a new paragraph, so `\smallskip` cannot be used within the horizontal mode `\schemabox` when using Plain TeX. In some cases, putting vertical space in the first or last lines of a `\schemabox`, regardless of format, will affect centering.

[7] Changes in TeX distributions can change font metrics and thus, the metrics of your schemata.

Armed with this information, we sally forth to reproduce the following schema found on page 13 of Martin Chemnitz, *Loci Theologici* (Frankfurt, 1653).[8]



- As you see, the braces were composed of various type sorts, mainly smaller rules and assorted curly and bendy bits.
- Because this is Latin we will see roman, italic and small caps, but little of other typefaces. We do see *s-medialis* and many old-style ligatures.
- In the reproduction we will use *s-finalis* only, but we will retain some ligatures.
- We will improve spacing between elements.
- We will not aim for an exact reproduction of line breaks and such.

[8]This image was created from a photograph taken by the author. It is the victim of a few cage transforms, despeckling, color selection and fill, color equalization, cleanup, scaling, and reduction to a two-color indexed palette.

We begin by looking at the "leaves," the rightmost bits of text enclosed by braces. We can use `\schema` in these cases. That results in the following:

```
\schema
{\schemabox{\textsc{Per se}:\\ scilicet.}}
{
  \schemabox{Unus in essentia.}
  \schemabox{Trinus in personis.}
}
```

PER SE: scilicet. { Unus in essentia. / Trinus in personis.

```
\schema
{\schemabox{Ad hominem\\ quem vel}}
{
  \schemabox{Accusat \& terret, \textsc{Per Legem},\\
  Consolatur \& erigit, \textsc{Per Evangelium}.\\
  Salvat, \textsc{Per Christum}.\\
  Renovat, \textsc{Per Spiritum Sanctum}.\\
  Sanctificat, \textsc{Per Verbum} \& \textsc{Sacramenta}.\\
  Castigat, tentat \& exercet, \textsc{Per Crucem}.\\
  Glorificat \textsc{Per Resurrectionem Carnis}\\
  \textsc{\quad Ad Vitam \AE{}ternam}.}
}
```

Ad hominem quem vel {
Accusat & terret, PER LEGEM,
Consolatur & erigit, PER EVANGELIUM.
Salvat, PER CHRISTUM.
Renovat, PER SPIRITUM SANCTUM.
Sanctificat, PER VERBUM & SACRAMENTA.
Castigat, tentat & exercet, PER CRUCEM.
Glorificat PER RESURRECTIONEM CARNIS
AD VITAM ÆTERNAM.

```
\schemabox{Ante lapsum.}

\schema
{\schemabox{Post lapsum:}}
{
  \schemabox{Ante Regenerationem \&\\
  Renovationem S. Sancti.}
  \schemabox{Post Regenerationem \&\\
  Renovationem S. Sancti.}
}
```

Ante lapsum.[9]

Post lapsum: {
Ante Regenerationem &
Renovationem S. Sancti.
Post Regenerationem &
Renovationem S. Sancti.

[9] We delete line 2 after *Ante lapsum* in the large example on page 13 and thereafter.

```
\schema
{\schemabox{\textsc{Deum},}}
{
  \schemabox{P\oe{}nitentia agens, agnitis peccatis \&\\
  ira Dei cognita \textsc{Ex Lege}.\\
  Erigens se \textsc{Voce Evangelii}.\\
  Credens \textsc{In Christum Salvatorem}.\\
  Non repugnans \textsc{Spiritui Sancto} impellenti.\\
  Audiens \textsc{Verbum}: \& utens \textsc{Sacramentis}.\\
  Patienter \& constanter sufferens \textsc{Crucem}.\\
  Sperans \& expectans glorificationem\\
  \textsc{\quad In Resurrectione Carnis}\\
  \textsc{\quad Ad Vitam \AE{}ternam}.}
}
```

Deum, { Pœnitentia agens, agnitis peccatis & ira Dei cognita Ex Lege. / Erigens se Voce Evangelii. / Credens In Christum Salvatorem. / Non repugnans Spiritui Sancto impellenti. / Audiens Verbum: & utens Sacramentis. / Patienter & constanter sufferens Crucem. / Sperans & expectans glorificationem In Resurrectione Carnis Ad Vitam Æternam.

```
\schema
{\schemabox{seipsum ratione}}
{\schemabox{Anim\ae{}\\ vel\\ Corporis}}
```

seipsum ratione { Animæ / vel / Corporis

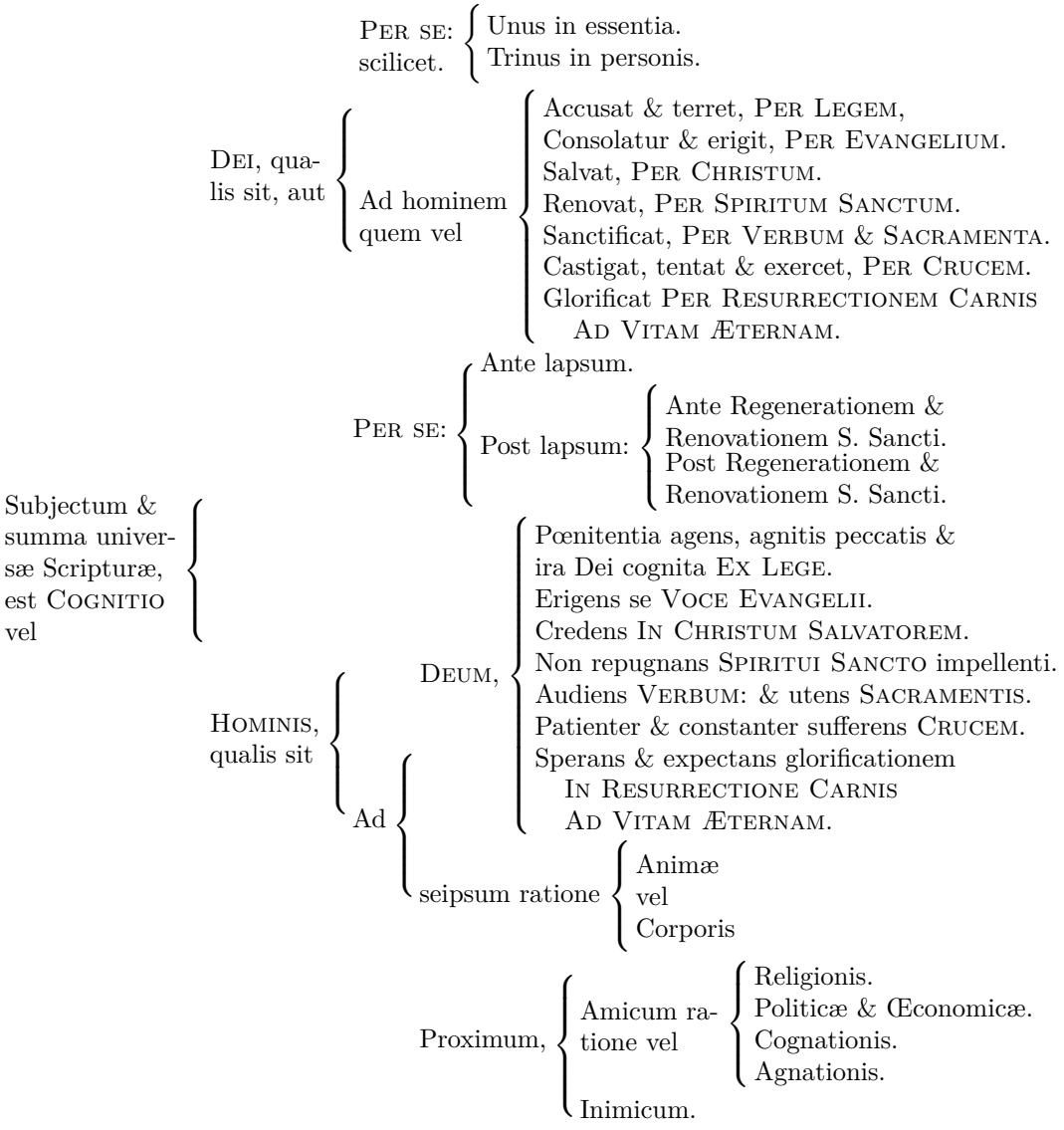
```
\schema
{\schemabox{Amicum ra-\\ tione vel}}
{
  \schemabox{Religionis.\\
  Politic\ae{} \& \OE{}conomic\ae{}.\\
  Cognationis.\\
  Agnationis.}
}

\schemabox{Inimicum.}
```

Amicum ratione vel { Religionis. / Politicæ & Œconomicæ. / Cognationis. / Agnationis.

Inimicum.[10]

[10] We delete line 9 before *Inimicum* in the large example on page 13 and thereafter.

Below we build all of the "leaves" into the larger schema using `\Schema`. The braces all have dummy values of `0ex` ⟨*adjust*⟩ and `5ex` ⟨*size*⟩. Please do not be alarmed at how bad this looks right now! We will adjust the layout shortly. We just want to see the general look of things:



Below we have the code listing for the schema above. One can see that there is much correlation between the listing and the printed result:

```
\Schema{0ex}{5ex}
{
  \schemabox{Subjectum \&\\
  summa univer-\\
  s\ae{} Scriptur\ae{},\\
  est \textsc{Cognitio}\\
  vel}
}
{
```

```
\Schema{0ex}{5ex}
{
  \schemabox{\textsc{Dei}, qua-\\lis sit, aut}
}
{
  \schema
  {\schemabox{\textsc{Per se}:\\ scilicet.}}
  {
    \schemabox{Unus in essentia.}
    \schemabox{Trinus in personis.}
  }
  \schema
  {\schemabox{Ad hominem\\ quem vel}}
  {
    \schemabox{Accusat \& terret, \textsc{Per Legem},\\
    Consolatur \& erigit, \textsc{Per Evangelium}.\\
    Salvat, \textsc{Per Christum}.\\
    Renovat, \textsc{Per Spiritum Sanctum}.\\
    Sanctificat, \textsc{Per Verbum} \& \textsc{Sacramenta}.\\
    Castigat, tentat \& exercet, \textsc{Per Crucem}.\\
    Glorificat \textsc{Per Resurrectionem Carnis}\\
    \textsc{\quad Ad Vitam \AE{}ternam}.}
  }
}
\Schema{0ex}{5ex}
{
  \schemabox{\textsc{Hominis},\\ qualis sit}
}
{
  \Schema{0ex}{5ex}
  {\schemabox{\textsc{Per se}:}}
  {
    \schemabox{Ante lapsum.}
    \schema
    {\schemabox{Post lapsum:}}
    {
      \schemabox{Ante Regenerationem \&\\
      Renovationem S. Sancti.}
      \schemabox{Post Regenerationem \&\\
      Renovationem S. Sancti.}
    }
  }
  \Schema{0ex}{5ex}
  {\schemabox{Ad}}
  {
    \schema
    {\schemabox{\textsc{Deum},}}
    {
      \schemabox{P\oe{}nitentia agens, agnitis peccatis \&\\
      ira Dei cognita \textsc{Ex Lege}.\\
      Erigens se \textsc{Voce Evangelii}.\\
      Credens \textsc{In Christum Salvatorem}.\\
      Non repugnans \textsc{Spiritui Sancto} impellenti.\\
      Audiens \textsc{Verbum}: \& utens \textsc{Sacramentis}.\\
      Patienter \& constanter sufferens \textsc{Crucem}.\\
      Sperans \& expectans glorificationem\\
```

```
      \textsc{\quad In Resurrectione Carnis}\\
      \textsc{\quad Ad Vitam \AE{}ternam}.}
    }
    \schema
      {\schemabox{seipsum ratione}}
      {\schemabox{Anim\ae{}\\ vel\\ Corporis}}
    \Schema{0ex}{5ex}
    {\schemabox{Proximum,}}
    {
      \schema
      {\schemabox{Amicum ra-\\ tione vel}}
      {
        \schemabox{Religionis.\\
        Politic\ae{} \& \OE{}conomic\ae{}.\\
        Cognationis.\\
        Agnationis.}
      }
      \schemabox{Inimicum.}
    }
  }
}
```

First, we add space between the "leaves" of the tree. If you do not work from right to left, you will waste time revising the "leaves" and "branches." The following lines, shown with some surrounding context, were changed.

Remember that you can add a `\smallskip` within a `\schemabox` in LaTeX, but not in Plain TeX. We have split the text below into two boxes to make it format-agnostic. See also how the second `\smallskip` follows the closing brace of the right-hand side, not the `\schemabox`. That adjusts the entire `\schema`.

```
    {
      \schemabox{Unus in essentia.}\smallskip
      \schemabox{Trinus in personis.}
    }\smallskip
```

Again, below, the skip comes at the close of a `\schema`.

```
      \textsc{\quad Ad Vitam \AE{}ternam}.}
    }\medskip
```

Below, the first skip helps to separate the lone `\schemabox` from the `\schema` beneath it. This illustrates how the internal vertical lists of schemata can contain heterogeneous material.

A medium skip is placed between two `\schemabox`es, which slightly throws off the way the brace spans the boxes. A small skip is put at the end of the last `\schemabox` to correct that. Sometimes putting skips within a `\schema` can be tricky. Then a `\smallskip` is added again at the end of the right-hand side.

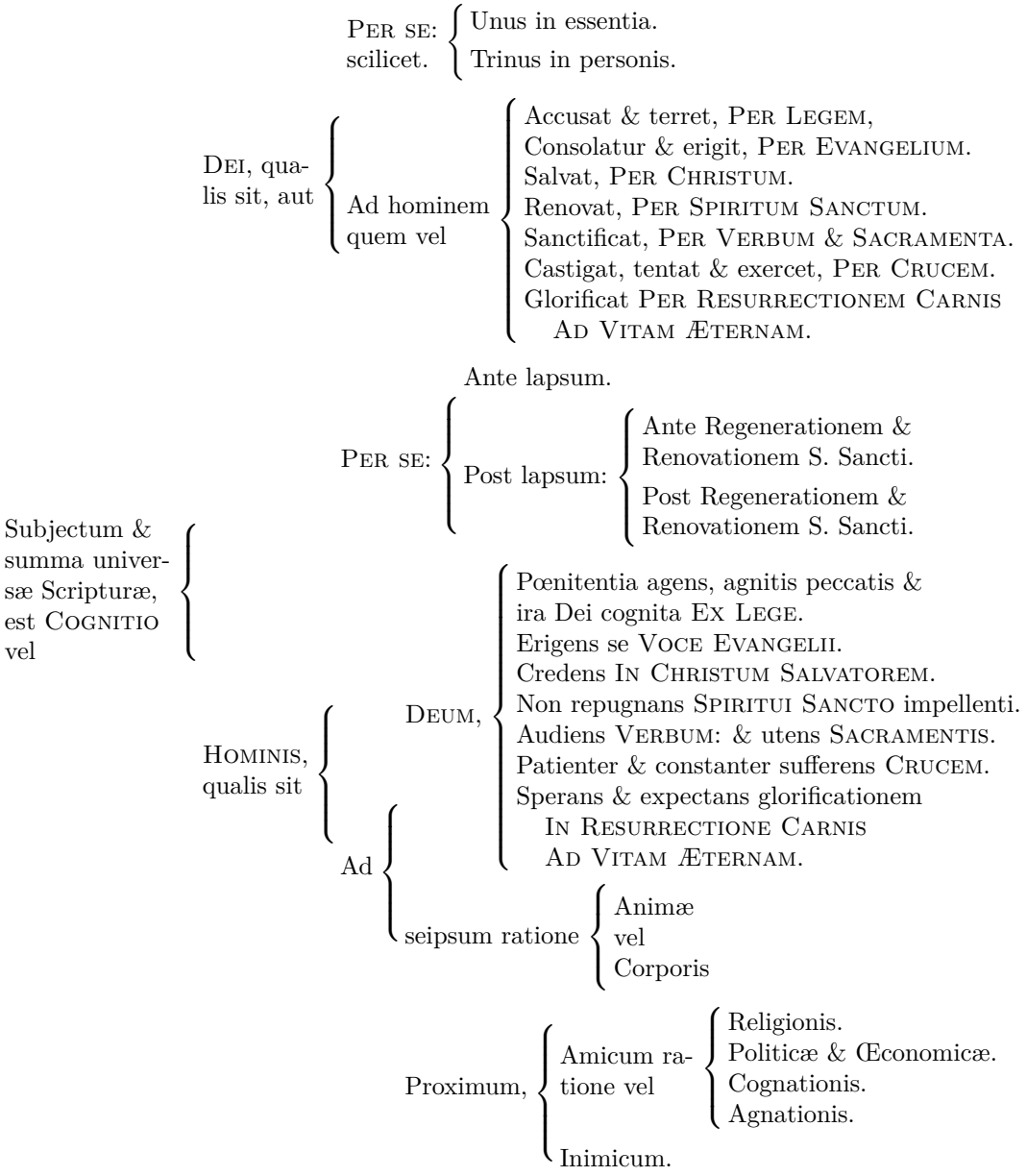
```
    {
      \schemabox{Ante lapsum.}\smallskip
      \schema
      {\schemabox{Post lapsum:}}
      {
        \schemabox{Ante Regenerationem \&\\
        Renovationem S. Sancti.}\medskip
```

```
48          \schemabox{Post Regenerationem \&\\
49          Renovationem S. Sancti.}\smallskip
50        }\smallskip
51      }
```

The skips below generally follow the same patterns that we have seen above.

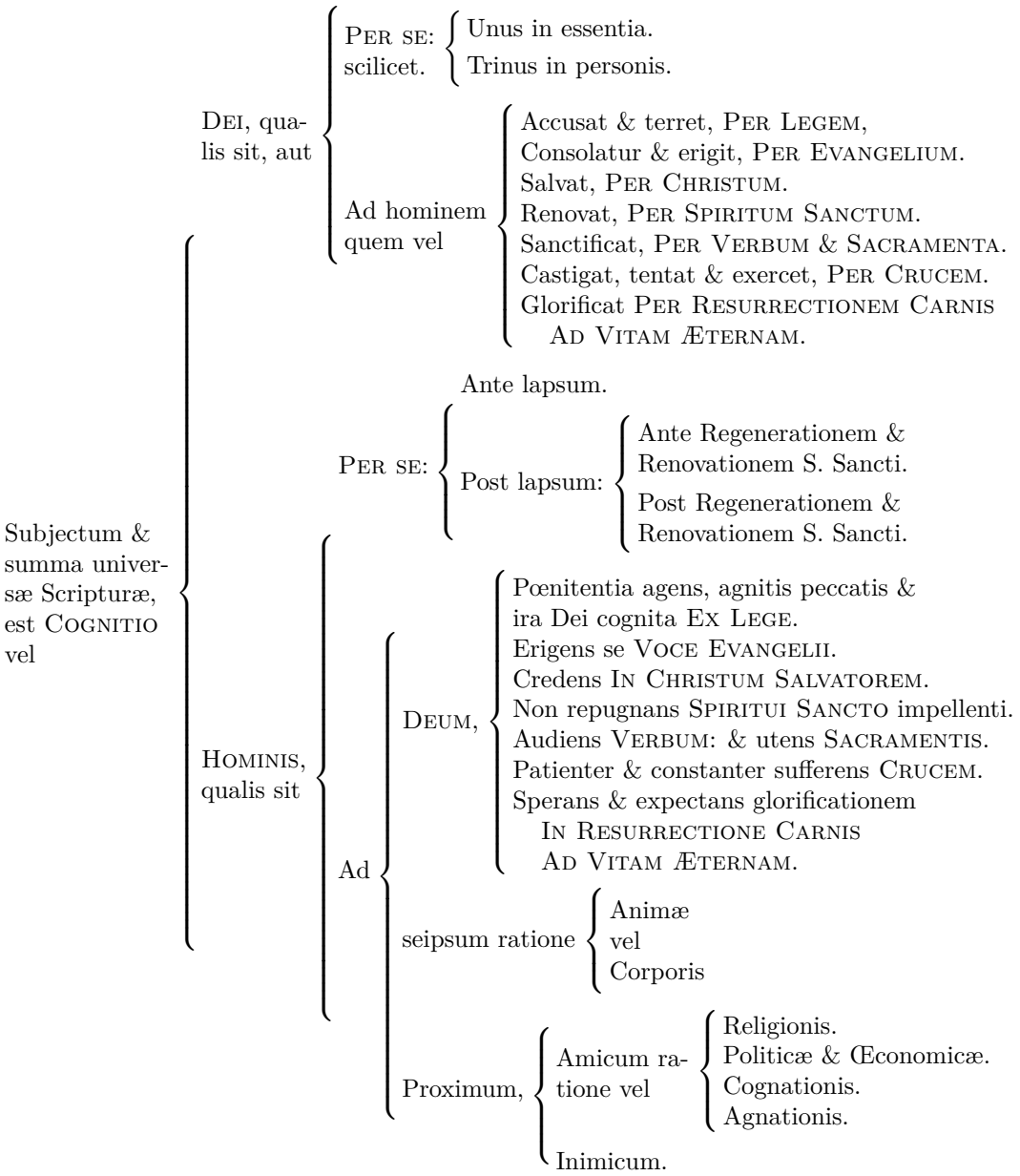
```
67          \textsc{\quad Ad Vitam \AE{}ternam}.}
68        }\smallskip
69        \schema
70          {\schemabox{seipsum ratione}}
71          {\schemabox{Anim\ae{}\\ vel\\ Corporis}}\smallskip

82            }\smallskip
83            \schemabox{Inimicum.}
```

The resulting schema looks better already:



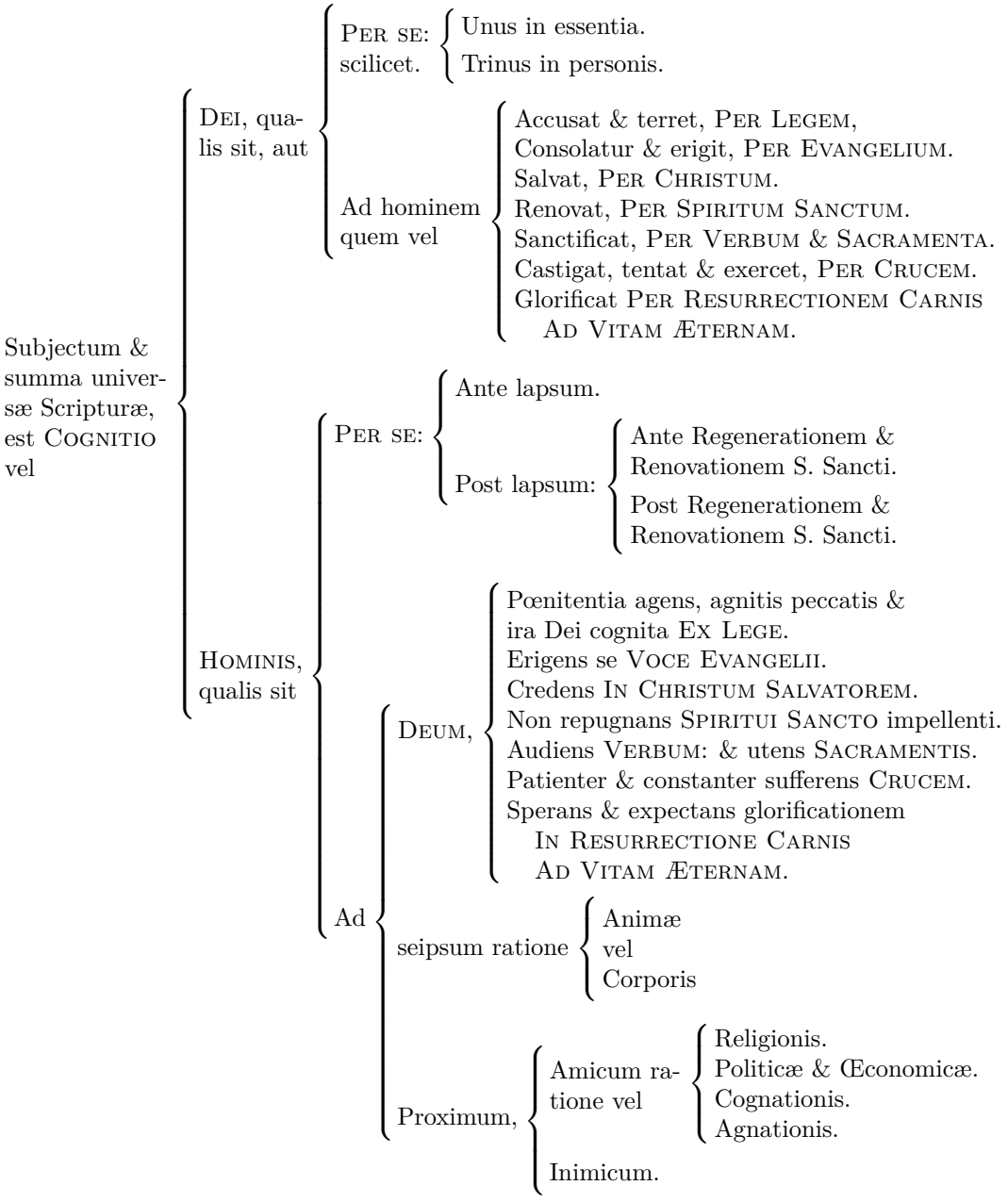
Next we estimate the lines of text and blank lines from the top of a `\Schema` brace to the bottom, e.g., from "Per se:" to "quem vel". We use those "ex" height figures for ⟨*size*⟩. The following lines illustrate our "ball park" figures:

```
1   \Schema{0ex}{23ex}

10  \Schema{0ex}{8ex}

34  \Schema{0ex}{16ex}

39  \Schema{0ex}{5ex}

52  \Schema{0ex}{16ex}

72  \Schema{0ex}{5ex}
```



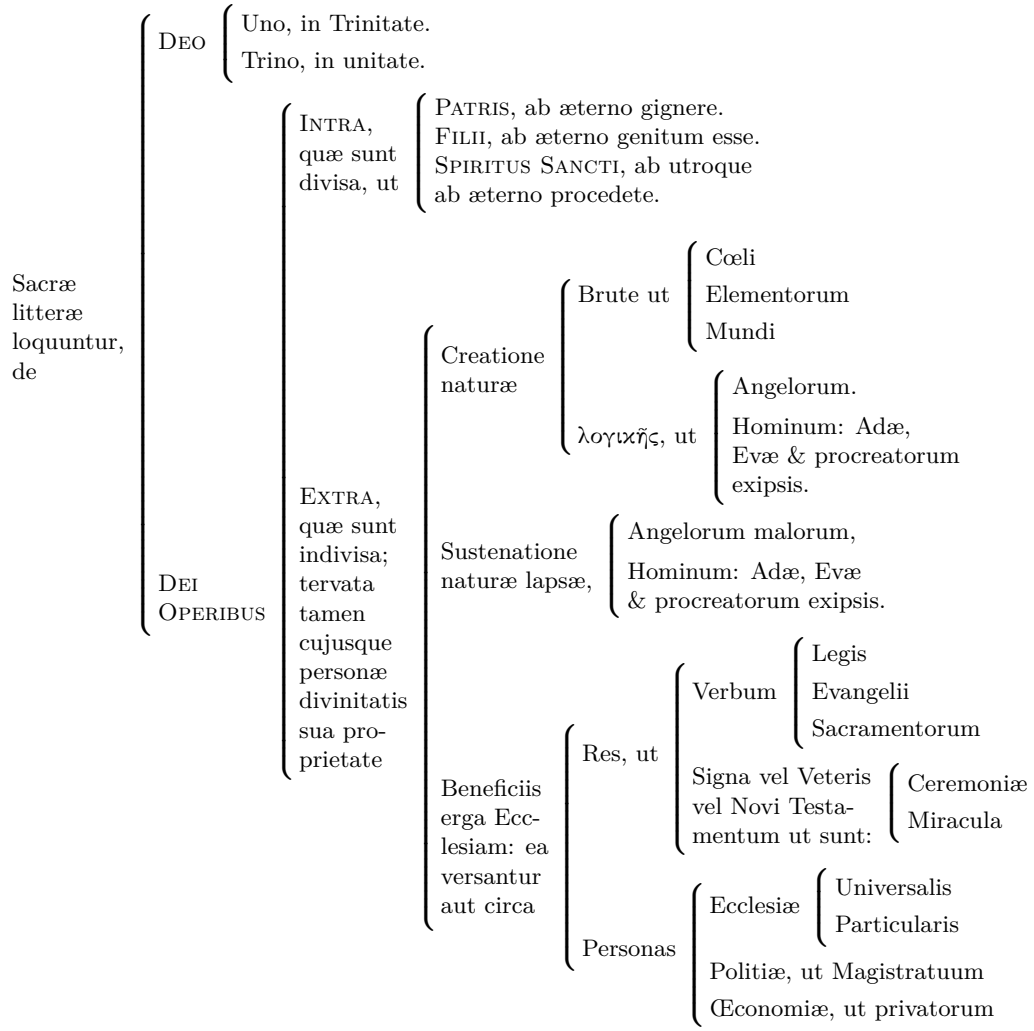
Next we tweak ⟨*adjust*⟩ values by counting the lines (`ex`) in the direction the left side needs to move relative to the right, multiply the result by two, and make it negative for up and positive for down. Using an editor, e.g., `texworks` makes this fairly easy. We also adjust the final ⟨*size*⟩ of the braces. Work from leaves to root:

```
 1  \Schema{-25ex}{20.6ex}% Do this one last. ``Subjectum''

10  \Schema{-6.4ex}{8.5ex}% Do this one first. ``Dei''

34  \Schema{-13.4ex}{17.4ex}% Do this one fifth. ``Hominis''

39  \Schema{-4.4ex}{5ex}% Do this one second. ``Per se'' (lower)

52  \Schema{4.2ex}{14.4ex}% Do this one fourth. ``Ad''

72  \Schema{2ex}{5.1ex}% Do this one third. ``Proximum''
```



2.4.4 Big Schema with Groups

The next example illustrates everything tha we have covered so far, plus `\DoGroups`, all inside a local scope:

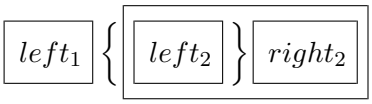


2.4.5 Open and Closed Schemata

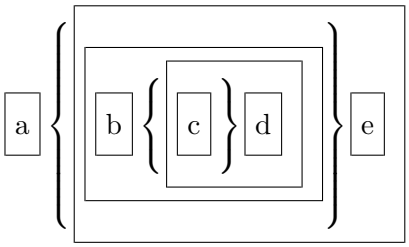
Now we look at schemata that have both open and closed braces. One must use `\Schema` to get delimiters to be the same height. These schemata take the form:

```
\Schema{⟨adjust⟩}{⟨height⟩}
{⟨left1⟩}
{
   \Schema[close]{⟨adjust⟩}{⟨height⟩}
   {⟨left2⟩}
   {⟨right2⟩}
}
```

We use a modified version of our `\Box` macro from above to show how each part nests within the other. Below we do not use `\NudgeSB` from Section 2.3 because we are not using `\schemabox`; instead we directly add the kern: `\hbox{\Box{\,$left_2$}\kern0.2em}` within the closed schema. The result is:

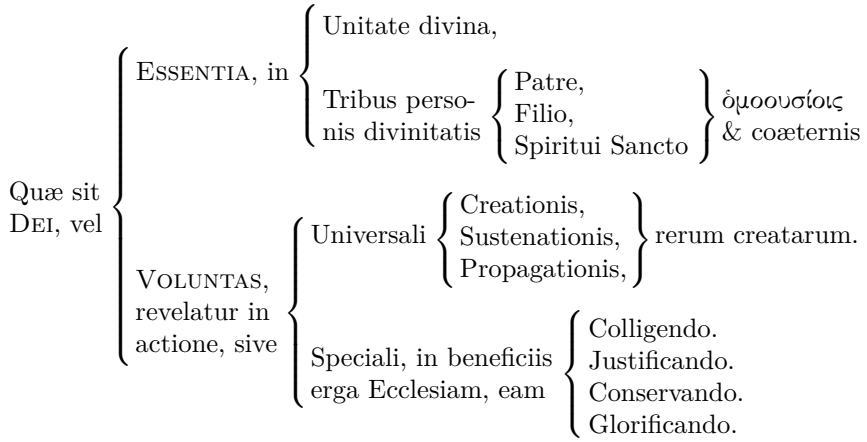


Here is another, more complex example:




```
\newbox\mybox
\def\Box#1{%
  \setbox\mybox=\hbox{\vrule\vbox{\hrule%
    \vfil\hbox{\strut\space #1\space}%
    \vfil\hrule}\vrule}%
  \dimen0=\ht\mybox%
  \advance\dimen0 by2ex%
  \hbox{\vrule\vbox to \dimen0{\hrule%
    \vfil\hbox{{\Large\strut}\space #1\space}%
    \vfil\hrule}\vrule}}
\Schema{0ex}{6ex}
{\Box{a}}
{%
  \Box{%
    \Schema[close]{0ex}{6ex}
    {%
      \Box{%
        \Schema{0ex}{3ex}
        {\Box{b}}
        {%
          \Box{%
            \Schema[close]{0ex}{3ex}
            {\hbox{\Box{c}\kern0.2em}}
            {\Box{d}}
          }
        }
      }
    }
    {\Box{e}}
  }
}
```

This is more of a real-world example. As above, one must use `\Schema` to prevent the opening braces from being slightly larger than the closing braces.

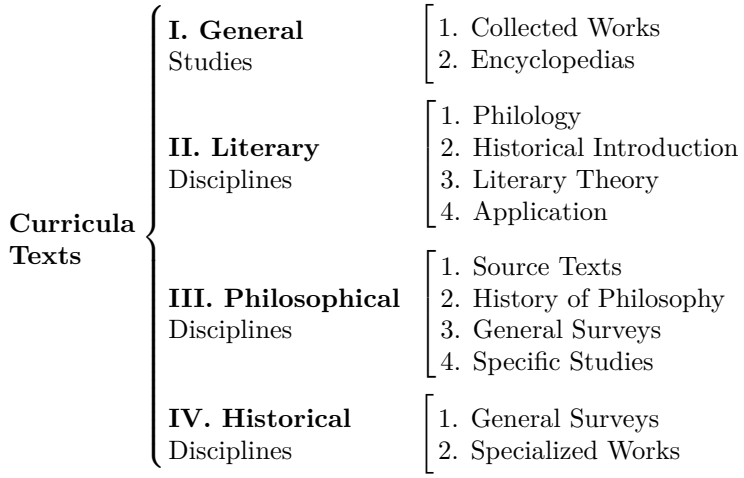


This listing of the example above illustrates closed schemata. The macro `\gk` uses either polyglossia or babel. We cannot show Unicode Greek text in the verbatim environment; we substitute `xxxxx` for ὁμοουσίοις.

```
\Schema{-1.4ex}{10ex}
{\schemabox{Qu\ae{} sit\\ \textsc{Dei}, vel}}
{
  \Schema{-1ex}{5ex}
  {\schemabox{\textsc{Essentia}, in}}
  {
    \vskip1ex\schemabox{Unitate divina,}
    \medskip
    \Schema{0ex}{3.4ex}
    {\schemabox{Tribus perso-\\ nis divinitatis}}
    {
      \Schema[close]{0ex}{3.4ex}
      {\NudgeSB\schemabox{Patre,\\ Filio,\\ Spiritui Sancto}}
      {\schemabox{\gk{xxxxx}{<omoous'iois}\\ \& co\ae{}ternis}}
    }
  }
  \medskip
  \Schema{-0.2ex}{6.4ex}
  {\schemabox{\textsc{Voluntas},\\ revelatur in\\ actione, sive}}
  {
    \Schema{0ex}{3.4ex}
    {\schemabox{Universali}}
    {
      \Schema[close]{0ex}{3.4ex}
      {\schemabox{Creationis,\\ Sustenationis,\\ Propagationis,}}
      {\schemabox{rerum creatarum.}}
    }
    \medskip
    \schema
    {\schemabox{Speciali, in beneficiis\\ erga Ecclesiam, eam}}
    {\schemabox{Colligendo.\\ Justificando.\\
     Conservando.\\ Glorificando.}}
  }
}
```

2.5 Final features

This final example illustrates how one can set the width of a `\schemabox`, and for what sort of use that might be. Below we invoke `\DoBrackets` after the start of the group containing the right-hand side of the first `\Schema`.




```
\Schema{-0.2ex}{14.4ex}
{\schemabox{\bfseries Curricula\\\bfseries Texts}}
{
  \DoBrackets%
  % \newbox here is doable in LaTeX, not in Plain TeX,
  % where it must be used as an \outer macro.
  \newbox\mybox%
  \setbox\mybox=\hbox{\bfseries III. Philosophical }%
  \dimen0=\wd\mybox%
  \schema
    {\schemabox[\dimen0]{\bfseries I. General\\Studies}}
    {\schemabox{1. Collected Works\\2. Encyclopedias}}
  \smallskip
  \schema
    {\schemabox[\dimen0]{\bfseries II. Literary\\Disciplines}}
    {\schemabox{1. Philology\\
     2. Historical Introduction\\
     3. Literary Theory\\
     4. Application}}
  \smallskip
  \schema
    {\schemabox[\dimen0]{\bfseries III. Philosophical\\Disciplines}}
    {\schemabox{1. Source Texts\\
     2. History of Philosophy\\
     3. General Surveys\\
     4. Specific Studies}}
  \smallskip
  \schema
    {\schemabox[\dimen0]{\bfseries IV. Historical\\Disciplines}}
    {\schemabox{1. General Surveys\\
     2. Specialized Works}}
}
```

3 Implementation

Shorter macros are written in both LaTeX and generic TeX. Longer macros implement both a LaTeX and a generic front end with a common back end.

If the format is "LaTeX2e" then the macros use the LaTeX 2_ε front end. Otherwise they use generic TeX, meaning Plain TeX, eplain, and Lollipop — maybe others too, but they are not supported.

In order to support such a diversity of formats and TeX engines, we must avoid newer primitives like `\unless` and `\ifdefined`. Thus, we must revert to the "old" way of testing whether or not a macro is defined.

`\schemataLaTeX` Below we manually duplicate with verbatim material what we put early in the `dtx` file for the versioning information to work. The `\schemataLaTeX` macro normally is undefined until it is assigned the value of `LaTeX2e`, to be compared with `\fmtname`. If we are not using LaTeX 2_ε, we do the equivalent of `\makeatletter` in either Plain TeX or eplain.

```
%<package>\expandafter\ifx \csname schemataLaTeX\endcsname\relax
%<package>  \def\schemataLaTeX{LaTeX2e}\fi
%<package>\ifx\fmtname\schemataLaTeX
%<package>\expandafter\NeedsTeXFormat\expandafter{\schemataLaTeX}[2005/12/01]
%<package>\ProvidesPackage{schemata}
%<*package>
  [2021/02/27 1.4 generic package to aid construction of topical categories]
%</package>
%<package>\else
%<package>\catcode`@=11\relax
%<package>\fi
```

3.1 Internal Variables

`\@schemata@LaTeX` We declare the internal macro `\@schemata@LaTeX` to be the value of `\schemataLaTeX` to safeguard package operation. From this point onward we can display or query `\schemataLaTeX` for user-side tests without affecting package internals.

```
\edef\@schemata@LaTeX{\schemataLaTeX}
```

Two box registers and two dimen registers are used to analyze the left-hand and right-hand vertical sizes of the boxes in a schema. Three more dimen registers are for scratchwork.

```
\newbox\@schemata@rhs%
\newbox\@schemata@lhs%
\newdimen\@schemata@rheight%
\newdimen\@schemata@lheight%
\newdimen\@schemata@one%
\newdimen\@schemata@two%
\newdimen\@schemata@three%
```

Two Boolean flags affect the height of a `\schemabox`, respectively setting and toggling that height for lowercase and uppercase content in order to add or remove space for boxes with only lowercase text.

```
\newif\if@schemata@LCBox%
\newif\if@schemata@SWBox%
```

This Boolean flag determines if a kern should be added to the end of each line in a `\schemabox` (helps with closed braces).

```
\newif\if@schemata@NudgeBox%
```

3.2 Package Options

We set braces to be the default set of delimiters. Apart from LaTeX 2$_\varepsilon$ we ignore the options. Three options are implemented, namely, `braces` (the default), `brackets`, and `parens`. Since the options are used infrequently, we naively process them in whatever order we get, each overwriting the last.

```
\ifx\fmtname\@schemata@LaTeX
  \DeclareOption{braces}%
    {\let\@schemata@LD\lbrace%
     \let\@schemata@RD\rbrace}
  \DeclareOption{brackets}%
    {\let\@schemata@LD\lbrack%
     \let\@schemata@RD\rbrack}
  \DeclareOption{parens}%
    {\let\@schemata@LD(%
     \let\@schemata@RD)}
  \DeclareOption{groups}%
    {\let\@schemata@LD\lgroup%
     \let\@schemata@RD\rgroup}
  \ExecuteOptions{braces}%
  \ProcessOptions\relax
\else
  \let\@schemata@LD\lbrace%
  \let\@schemata@RD\rbrace%
\fi
```

3.3 Macros

`\DoBraces` Set the delimiters to be braces. This is local to a scope, including within a schema.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\DoBraces}%
    {\let\@schemata@LD\lbrace%
     \let\@schemata@RD\rbrace}
\else
  \def\DoBraces%
    {\let\@schemata@LD\lbrace%
     \let\@schemata@RD\rbrace}
\fi
```

`\DoBrackets` Set the delimiters to be brackets. This is local, as above.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\DoBrackets}%
    {\let\@schemata@LD\lbrack%
     \let\@schemata@RD\rbrack}
\else
  \def\DoBrackets%
    {\let\@schemata@LD\lbrack%
     \let\@schemata@RD\rbrack}
\fi
```

`\DoParens` Set the delimiters to be parentheses. This is local, as above.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\DoParens}%
    {\let\@schemata@LD(%
     \let\@schemata@RD)}
```

```
\else
  \def\DoParens%
    {\let\@schemata@LD(%
     \let\@schemata@RD)}
\fi
```

`\DoGroups` Set the delimiters to be parentheses. This is local, as above.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\DoGroups}%
    {\let\@schemata@LD\lgroup%
     \let\@schemata@RD\rgroup}
\else
  \def\DoGroups%
    {\let\@schemata@LD\lgroup%
     \let\@schemata@RD\rgroup}
\fi
```

`\LCschema` Prevent `\schemabox` from adding a `\strut` in the first line.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\LCschema}{\@schemata@LCBoxtrue}
\else
  \def\LCschema{\@schemata@LCBoxtrue}
\fi
```

`\UCschema` Permit `\schemabox` to add a `\strut` in the first line (default).

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\UCschema}{\@schemata@LCBoxfalse}
\else
  \def\UCschema{\@schemata@LCBoxfalse}
\fi
```

`\SwitchSB` Flip the UC/LC settings for one `\schemabox`, which will reset this value on exit.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\SwitchSB}{\@schemata@SWBoxtrue}%
\else
  \def\SwitchSB{\@schemata@SWBoxtrue}
\fi
```

`\NudgeSB` Add a kern to the end of each line in one `\schemabox`. This will be reset on exit from that `\schemabox`.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\NudgeSB}{\@schemata@NudgeBoxtrue}
\else
  \def\NudgeSB{\@schemata@NudgeBoxtrue}
\fi
```

`\SBNudgeFactor` Define the `\kern` to be added to the end of each line in one `\schemabox`. The default is `0.2em`, equal to the horizontal corrective.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand{\SBNudgeFactor}{\kern0.2em}
\else
  \def\SBNudgeFactor{\kern0.2em}
\fi
```

\schemabox If in internal vertical mode, restricted horizontal mode, or math mode, wrap a stack of \hboxes in a \vbox, then put that inside an \hbox. The first argument sets an optional width for those \hboxes. Normally insert a \strut in the first \hbox. The second argument contains the rows of horizontal material, where \\ is redefined to end one \hbox and begin another. When in any other mode mode, just display the second argument as text.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\schemabox}[2][0pt]{\@schemata@schemabox[#1]{#2}}
\else
  \def\schemabox{\futurelet\@schemata@testchar\@schemata@schemab@x}
  \def\@schemata@schemab@x{%
    \ifx[\@schemata@testchar
      \let\next\@schemata@schemabox%
    \else
      \let\next\@schemata@@schemab@x%
    \fi
    \next%
  }%
  \def\@schemata@@schemab@x#1{\@schemata@schemabox[0pt]{#1}}
\fi
\def\@schemata@schemabox[#1]#2{%
  \ifinner
    \if@schemata@LCBox
      \def\@schemata@Adj{}%
      \if@schemata@SWBox\def\@schemata@Adj{\strut}\fi
    \else
      \def\@schemata@Adj{\strut}%
      \if@schemata@SWBox\def\@schemata@Adj{}\fi
    \fi
    \if@schemata@NudgeBox
      \let\@schemata@Nudge\SBNudgeFactor%
    \else
      \def\@schemata@Nudge{}%
    \fi
    \ifdim#1<1pt
      \def\\{\@schemata@Nudge\egroup\hbox\bgroup\ignorespaces }%
      \vbox{\hbox\bgroup%
        \@schemata@Adj\ignorespaces #2\@schemata@Nudge%
        \egroup}%
    \else
      \def\\{\hfil\egroup\hbox to #1\bgroup\ignorespaces }%
      \vbox{\hbox to #1\bgroup%
      \@schemata@Adj\ignorespaces #2\hfil%
      \egroup}%
    \fi
  \else
    #2%
  \fi
  \@schemata@SWBoxfalse%
  \@schemata@NudgeBoxfalse%
}
```

`\schema` This "simple" schema vertically centers two boxes of internal vertical material and puts a "simple" brace between the boxes based on the height of the box and the options passed to the schema.

There is something of a "magic" value for adjusting the height used for the larger side of a `\schema`, namely `1.44265ex`. By using this adjustment, which is slightly larger than $\sqrt{2}$ times the ex-height of the font, the results look more aesthetically pleasing in terms of centering and size of the braces.

By default, a schema has a box to the left, an open delimiter, and a box to the right. If any optional argument other than `open` is used, the schema prints a box to the left, a close brace, and a box to the right.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand{\schema}[3][open]{%
    \@schemata@schema[#1]{#2}{#3}}
\else
  \long\def\schema{\futurelet\@schemata@testchar\@schemata@schem@}
  \long\def\@schemata@schem@{%
    \ifx[\@schemata@testchar
      \let\next\@schemata@schema%
    \else
      \let\next\@schemata@@schem@%
    \fi
    \next%
  }%
  \long\def\@schemata@@schem@#1#2{%
    \@schemata@schema[open]{#1}{#2}}
\fi
\long\def\@schemata@schema[#1]#2#3{%
  \def\@schemata@option{#1}\def\@schemata@open{open}%
  \ifx\@schemata@option\@schemata@open
    \setbox\@schemata@rhs=\vbox{#3}%
    \@schemata@rheight=\ht\@schemata@rhs%
    \advance\@schemata@rheight\dp\@schemata@rhs%
    \advance\@schemata@rheight by 1.44265ex%
    \hbox{$\vcenter{#2}%
      \@schemata@lbrace{\@schemata@rheight}%
      \vcenter{#3}$}%
  \else
    \setbox\@schemata@lhs=\vbox{#2}%
    \@schemata@lheight=\ht\@schemata@lhs%
    \advance\@schemata@lheight\dp\@schemata@lhs%
    \advance\@schemata@lheight by 1.44265ex%
    \hbox{$\vcenter{#2}%
      \kern-0.2em\@schemata@rbrace{\@schemata@lheight}%
      \vcenter{#3}$}%
  \fi
}
```

`\Schema` This is the general-purpose form of schemata. The arguments include whether it is an open or closed schema, the vertical adjustment of the left-hand side and delimiter over against the right-hand side, the size of the brace, and the contents of the left and right-hand sizes. It works about the same as above, but requires manual adjustment of the braces. Again we see the "magic" height adjustment value of `1.44265ex`.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand{\Schema}[5][open]{%
    \@schemata@Schema[#1]{#2}{#3}{#4}{#5}}
\else
  \long\def\Schema{\futurelet\@schemata@testchar\@schemata@Schem@}
  \long\def\@schemata@Schem@{%
    \ifx[\@schemata@testchar
      \let\next\@schemata@Schema%
    \else
      \let\next\@schemata@@Schem@%
    \fi
    \next%
  }%
  \long\def\@schemata@@Schem@#1#2#3#4{%
    \@schemata@Schema[open]{#1}{#2}{#3}{#4}}
\fi
\long\def\@schemata@Schema[#1]#2#3#4#5{%
  \def\@schemata@option{#1}%
  \def\@schemata@open{open}%
  \@schemata@one=#2%
  \ifx\@schemata@option\@schemata@open
    \hbox{$\vcenter{\vskip1.44265\@schemata@one#4}%
      \@schemata@biglbrace{#2}{#3}\vcenter{#5}$}%
  \else
    \hbox{$\vcenter{\vskip1.44265\@schemata@one#4}\kern-0.2em%
      \@schemata@bigrbrace{#2}{#3}\vcenter{#5}$}%
  \fi
}
```

`\@schemata@lbrace` Draw an on-center delimiter to the left of a simple box.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\@schemata@lbrace}[1]{%
    \ifmmode
      \left.\vcenter{\vbox to #1{\vfil}}\right\@schemata@LD%
    \fi
  }
\else
  \def\@schemata@lbrace#1{%
    \ifmmode
      \left.\vcenter{\vbox to #1{\vfil}}\right\@schemata@LD%
    \fi
  }
\fi
```

`\@schemata@rbrace` Draw an on-center delimiter to the right of a simple box.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\@schemata@rbrace}[1]{%
    \ifmmode
      \left\@schemata@RD\vcenter{\vbox to #1{\vfil}}\right.%
    \fi
  }
\else
  \def\@schemata@rbrace#1{%
    \ifmmode
      \left\@schemata@RD\vcenter{\vbox to #1{\vfil}}\right.%
    \fi
  }
\fi
```

`\@schemata@biglbrace` Draw a vertically-adjustable delimiter to the left of a complex assortment of boxes. Again we see the "magic" height adjustment value of `1.44265ex`, but both positive and negative.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\@schemata@biglbrace}[2]{%
    \@schemata@@biglbrace{#1}{#2}}
\else
  \def\@schemata@biglbrace#1#2{%
    \@schemata@@biglbrace{#1}{#2}}
\fi
\def\@schemata@@biglbrace#1#2{%
  \@schemata@one=#1%
  \@schemata@two=#2%
  \@schemata@three=-\@schemata@two%
  \ifdim\@schemata@three>\@schemata@two%
    \@schemata@two=\@schemata@three\fi
  \ifdim\@schemata@one<0pt
    \ifmmode\vcenter{\hbox{$\left.%
      \vbox to 1.44265\@schemata@two{\vfil}%
      \right\@schemata@LD%
      \atop\vbox to -1.44265\@schemata@one{\vfil}$}}\fi
  \else
    \ifmmode\vcenter{\hbox{$\vbox to 1.44265\@schemata@one{\vfil}%
      \atop\left.%
      \vbox to 1.44265\@schemata@two{\vfil}%
      \right\@schemata@LD$}}\fi
  \fi
}
```

`\@schemata@bigrbrace` Draw a vertically-adjustable delimiter to the right of a complex assortment of boxes. Again we see the "magic" height adjustment value of `1.44265ex`, but both positive and negative.

```
\ifx\fmtname\@schemata@LaTeX
  \newcommand*{\@schemata@bigrbrace}[2]{%
    \@schemata@@bigrbrace{#1}{#2}%
  }
\else
  \def\@schemata@bigrbrace#1#2{%
    \@schemata@@bigrbrace{#1}{#2}%
  }
\fi
\def\@schemata@@bigrbrace#1#2{%
  \@schemata@one=#1%
  \@schemata@two=#2%
  \@schemata@three=-\@schemata@two%
  \ifdim\@schemata@three>\@schemata@two%
    \@schemata@two=\@schemata@three\fi
  \ifdim\@schemata@one<0pt
    \ifmmode\vcenter{\hbox{$\left.%
      \vbox to 1.44265\@schemata@two{\vfil}%
      \right\@schemata@RD%
      \atop\vbox to -1.44265\@schemata@one{\vfil}$}}\fi
  \else
    \ifmmode\vcenter{\hbox{$\vbox to 1.44265\@schemata@one{\vfil}%
      \atop\left.%
      \vbox to 1.44265\@schemata@two{\vfil}%
      \right\@schemata@RD$}}\fi
  \fi
}
```

If we are not using LaTeX 2_ε, we do the equivalent of `\makeatother`.

```
\ifx\fmtname\@schemata@LaTeX\else
  \catcode`@=12\relax
\fi
```

4 Change History

0.5
General: Initial version 1
0.6
\DoBraces: Added macro 24
\DoBrackets: Added macro 24
\DoParens: Added macro 24
\LCschema: Added macro 25
\SwitchSB: Added macro 25
\UCschema: Added macro 25
\schemabox: Added lowercase tweaks . . . 26
General: Added brackets and parens as well as braces 24
Added features 1
Added UC/LC tweaks. 23
0.7
General: Changed contact info 1
0.8
\@schemata@biglbrace: Renamed; use absolute value of brace size 29
\@schemata@bigrbrace: Renamed; Use absolute value of brace size 30
\@schemata@lbrace: Renamed 28
\@schemata@rbrace: Renamed 29
\NudgeSB: Added macro 25
\schemabox: Added nudge feature; fix errors when not in internal vertical mode . . . 26
General: Rename box/dimen registers . . . 23
Renamed internal macros 23
Rewrote manual 1
1.0
\@schemata@biglbrace: ensure short; create front- and back-end 29
\@schemata@bigrbrace: ensure short; create front- and back-end 30
\@schemata@lbrace: ensure short 28
\@schemata@rbrace: ensure short 29
\DoBraces: ensure short 24
\DoBrackets: ensure short 24
\DoGroups: Added macro 25
\DoParens: ensure short 24
\LCschema: ensure short 25
\NudgeSB: ensure short 25
\SBNudgeFactor: Added macro 25
\Schema: create front- and back-end; ensure long 28
\SwitchSB: ensure short 25
\UCschema: ensure short 25
\schema: create front- and back-end; ensure long 27
\schemabox: Create front- and back-end; ensure short 26
\schemataLaTeX: Added format-specific features 23
General: Added group option 24
Ensure better multi-format operation . . . 1
Rename box/dimen registers 23
1.1
General: Fix issue with dtx guards 1
1.2
\Schema: Fix namespace 28
\schema: Fix namespace 27
\schemabox: Fix namespace 26
General: Updates to Readme.md, Makefile, schematest.tex, combine Readme.md and schematest.tex files in dtx 1
1.3
\@schemata@LaTeX: Added 23
\schemataLaTeX: Fix format detection bug 23
1.4
General: Updates to Readme.md 1

5 Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols

\@schemata@@Schem@ 193, 197
\@schemata@@biglbrace 240, 243, 245
\@schemata@@bigrbrace 265, 269, 272
\@schemata@@schem@ 157, 161
\@schemata@@schemab@x 111, 115
\@schemata@Adj 120, 121, 123, 124, 134, 139
\@schemata@LCBoxfalse 84, 86
\@schemata@LCBoxtrue . 79, 81
\@schemata@LD 25, 28, 31, 34, 39, 44, 48, 53, 57, 62, 66, 71, 75, 215, 221, 254, 260
\@schemata@LaTeX 12, 23, 42, 51, 60, 69, 78, 83, 88, 93, 98, 103, 148, 184, 212, 225, 238, 263, 290
\@schemata@Nudge 127, 129, 132, 134
\@schemata@NudgeBoxfalse 146
\@schemata@NudgeBoxtrue 94, 96
\@schemata@RD 26, 29, 32, 35, 40, 45, 49, 54, 58, 63, 67, 72, 76, 228, 234, 281, 287
\@schemata@SWBoxfalse . . 145
\@schemata@SWBoxtrue . 89, 91
\@schemata@Schem@ . 188, 189
\@schemata@Schema 186, 191, 198, 200
\@schemata@biglbrace 206, 238
\@schemata@bigrbrace 209, 263
\@schemata@lbrace . 172, 212
\@schemata@lheight 16, 176, 177, 178, 180
\@schemata@lhs 14, 175, 176, 177
\@schemata@one 17, 203, 205, 208, 246, 251, 255, 257, 273, 278, 282, 284
\@schemata@open 165, 166, 202, 204
\@schemata@option 165, 166, 201, 204
\@schemata@rbrace . 180, 225
\@schemata@rheight 15, 168, 169, 170, 172
\@schemata@rhs 13, 167, 168, 169
\@schemata@schem@ . 152, 153
\@schemata@schema 150, 155, 162, 164
\@schemata@schemab@x 106, 107
\@schemata@schemabox 104, 109, 115, 117
\@schemata@testchar 106, 108, 152, 154, 188, 190
\@schemata@three 19, 248, 249, 250, 275, 276, 277
\@schemata@two . . 18, 247, 248, 249, 250, 253, 259, 274, 275, 276, 277, 280, 286
\\ 132, 137

A

\advance . . 169, 170, 177, 178
\atop 255, 258, 282, 285

C

\catcode 291

D

\DoBraces *3*, 42
\DoBrackets *3*, 51
\DoGroups *3*, 69
\DoParens *3*, 60
\dp 169, 177

E

\edef 12

F

\fmtname 23, 42, 51, 60, 69, 78, 83, 88, 93, 98, 103, 148, 184, 212, 225, 238, 263, 290

H

\ht 168, 176

I

\if@schemata@LCBox . 20, 119
\if@schemata@NudgeBox 22, 126
\if@schemata@SWBox 21, 121, 124
\ifinner 118

K

\kern 99, 101, 180, 208

L

\lbrace 25, 39, 44, 48
\lbrack 28, 53, 57
\LCschema *6*, 78
\left 215, 221, 228, 234, 252, 258, 279, 285
\lgroup 34, 71, 75
\long 152, 153, 161, 164, 188, 189, 197, 200

N

\NudgeSB *7*, 93

R

\rbrace 26, 40, 45, 49
\rbrack 29, 54, 58
\rgroup 35, 72, 76
\right 215, 221, 228, 234, 254, 260, 281, 287

S

\SBNudgeFactor . . . *7*, 98, 127
\Schema *4*, 184
\schema *3*, 148
\schemabox *2*, 103
\schemataLaTeX 1, *2*, 12
\strut 121, 123
\SwitchSB *7*, 88

U

\UCschema *6*, 83