

schemata — Generic package to aid construction of topical categories*

Charles P. Schaum[†]

Released 2020/12/02

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	2.4.3	Going Big	9
2	Usage	2	2.4.4	Big Groups	19
2.1	Loading and Options	2	2.4.5	Open and Closed	19
2.2	Macro Overview	2	2.5	Final features	22
2.2.1	<code>\schemabox</code>	2	3	Implementation	23
2.2.2	Delimiters	3	3.1	Internal Variables	23
2.2.3	<code>\schema</code>	3	3.2	Package Options	24
2.2.4	<code>\Schema</code>	4	3.3	Macros	24
2.3	Romancing the <code>\schema</code>	6	4	Change History	31
2.4	Tutorial	7	5	Index	32
2.4.1	Starting Off Basic	7			
2.4.2	<i>Loc</i> i 101	8			

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 1.3, last revised 2020/12/02.

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

¹Books 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).

²For 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 $\LaTeX 2_{\epsilon}$. 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`CONTEX`T does not like the way that `schemata` nests math-mode expressions within boxes.

2.2.2 Delimiters

`\DoBraces` 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.

`\DoBrackets` `\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:

$$a \left\{ \begin{array}{c} b \\ c \end{array} \right\} d \quad a \left[\begin{array}{c} b \\ c \end{array} \right] d \quad a \left(\begin{array}{c} b \\ c \end{array} \right) d \quad a \left(\begin{array}{c} b \\ c \end{array} \right) d$$

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

```

1 \DoBrackets
2 \Schema{0ex}{2.4ex}
3   {\schemabox{a}}
4   {\DoParens\Schema[close]{0ex}{2.3ex}
5     {\schemabox{b}\c}}
6     {\schemabox{d}}
7 }
```

$$a \left(\begin{array}{c} b \\ c \end{array} \right) d$$

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

```

1 \makeatletter
2 \newcommand*{\DoVerts}%
3   {\let\@schemata@LD\bracevert%
4     \let\@schemata@RD\bracevert}
5 \makeatother
6 \DoVerts
7 \Schema{0ex}{5ex}
8   {\vskip0.6ex\schemabox{a}}
9   {\Schema[close]{0ex}{5ex}
10    {\vskip0.4ex\schemabox{b}\c\d\l\l\l}
11    {\vskip0.6ex\schemabox{\kern0.1em}f}}
12 }
```

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

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:

$$\boxed{\text{\code{\schema[<type>]{<left side>}{<right side>}}}$$

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:

```

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

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

Any value of $\langle type \rangle$ other than the exact string `open` makes a “closed” schema (the delimiter opens to the left):

```

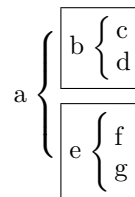
1 \schema[closed]
2   {\NudgeSB\schemabox{b\c}}
3   {\schemabox{a}}

```

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

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:

$$\boxed{\text{\Schema}[\langle type \rangle]\{\langle adjust \rangle\}\{\langle size \rangle\}\{\langle left side \rangle\}\{\langle right side \rangle\}}$$

The $\langle type \rangle$ is `open` by default. As above, any other $\langle type \rangle$ except the exact string `open` will make it a “closed” schema. Both $\langle adjust \rangle$ and $\langle size \rangle$ 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 $\langle adjust \rangle$:⁴

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

Set the delimiter $\langle size \rangle$ 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:

$$\text{main idea} \left\{ \begin{array}{l} \text{part 1} \left\{ \begin{array}{l} \text{detail a} \\ \text{detail b} \end{array} \right\} \\ \text{part 2} \left\{ \begin{array}{l} \text{detail c} \\ \text{detail d} \end{array} \right\} \end{array} \right\} \text{synonym} \left\{ \begin{array}{l} \text{part 3} \left\{ \begin{array}{l} \text{detail e} \\ \text{detail f} \end{array} \right\} \\ \text{part 4} \left\{ \begin{array}{l} \text{detail g} \\ \text{detail h} \end{array} \right\} \end{array} \right.$$

⁴Instead of setting $\langle adjust \rangle$, 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 $\langle adjust \rangle$ keeps that centered look while allowing some adjustments.

The source for that complex schema looks like:

```

1 \Schema[close]{0ex}{5.1ex}
2 {
3   \Schema{0.1ex}{3.8ex}
4   {\SwitchSB\schemabox{main idea}}
5   {
6     \schema{\schemabox{part 1}}
7     {\SwitchSB\NudgeSB\schemabox{detail a\detail b}}\smallskip
8     \schema{\schemabox{part 2}}
9     {\SwitchSB\NudgeSB\schemabox{detail c\detail d}}
10  }
11 }
12 {
13   \Schema{0ex}{3.8ex}
14   {\schemabox{\,synonym}}
15   {
16     \schema{\schemabox{part 3}}
17     {\SwitchSB\schemabox{detail e\detail f}}\smallskip
18     \schema{\schemabox{part 4}}
19     {\SwitchSB\schemabox{detail g\detail h}}
20   }
21 }

```

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

```

1 \schema
2   {\schemabox{a}}
3   {\schemabox{b\c}}
4
5 \schema
6   {\schemabox{d}}
7   {\schemabox{e\f}}

```

$$\begin{array}{l}
 a \left\{ \begin{array}{l} b \\ c \end{array} \right. \\
 d \left\{ \begin{array}{l} e \\ f \end{array} \right.
 \end{array}$$

They can be on a line of text: Does this $\left\{ \begin{array}{l} \text{look} \\ \text{ugly?} \end{array} \right.$

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:

```

1 \def\Box{%
2   \hbox{%
3     \vrule%
4     \vbox to 1cm{\hrule\hbox to 1cm{\hfil}\vfil\hrule}%
5     \vrule%
6   }%
7 }

```

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

```

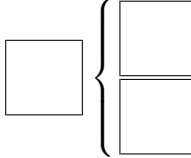
8 \schema{\Box}{\Box}

```

$$\square \left\{ \square \right.$$

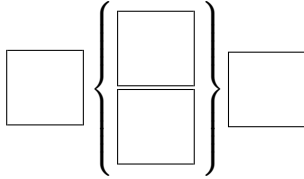
This example is more complex, showing how each side stacks \Boxes vertically:

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



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

```
10 \Schema{-0.2ex}{0.9cm}
11 {\Box}
12 {
13   \Schema[close]
14     {-0.2ex}{0.9cm}
15     {\Box\hbox{\Box\kern0.2em}}
16     {\Box}
17 }
```



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` By default, a `\schemabox` adds a `\strut` to the first line because the topics in
`\UCschema` 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

```
1 \LCschema
2 \Schema{0.1ex}{4.8ex}
3 {\hbox{sensus literalis}}
4 {
5   \schema{\schemabox{sensus\\literalis\\(improprie)}}
6     {\schemabox{e parallelismo clarior\\
7       ex analogia fidei\\ex evidentia rei}}
8     \smallskip\schemabox{sensus literae}
9 }
10 \UCschema
```

$$\text{sensus literalis} \left\{ \begin{array}{l} \text{sensus} \\ \text{literalis} \\ \text{(improprie)} \\ \text{sensus literae} \end{array} \right\} \left\{ \begin{array}{l} \text{e parallelismo clarior} \\ \text{ex analogia fidei} \\ \text{ex evidentia rei} \end{array} \right.$$

⁵Based 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          { c b
```

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 { b
                           c
```

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 { b
```

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

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

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`:

```

1 \schema
2 {
3   \schemabox{Subjectum theo-\
4   logi\ae{}} est Notitia\
5   Dei. Considerat\
6   ergo, Dei, vel}
7 }
8 {
9   \schema
10  {
11   \schemabox{\textsc{Essentiam},}
12  }
13  {
14   \schemabox{Unitate natur\ae{}}.\
15   Trinitate personarum.\
16   Operibus ad intra.}
17 }
18 \schema
19 {
20   \schemabox{\textsc{Voluntatem},\
21   manifestatam in\
22   operibus ad extra;\
23   ut in}
24 }
25 {
26   \schemabox{Creatione.\
27   Sustentatione natur\ae{} laps\ae{}}.\
28   Reparatione.\
29   Conversione.\
30   Justificatione.\
31   Sanctificatione \&\
32   Glorificatione ejusdem.}
33 }
34 }
```

Subjectum theologiae est Notitia Dei. Considerat ergo, Dei, vel	}	<table style="border: none;"> <tr> <td style="vertical-align: middle; padding-right: 10px;">ESSENTIAM,</td> <td style="font-size: 3em; vertical-align: middle; padding-right: 10px;">{</td> <td style="vertical-align: middle;">Unitate naturæ. Trinitate personarum. Operibus ad intra.</td> </tr> <tr> <td style="vertical-align: middle; padding-right: 10px;">VOLUNTATEM, manifestatam in operibus ad extra; ut in</td> <td style="font-size: 3em; vertical-align: middle; padding-right: 10px;">{</td> <td style="vertical-align: middle;">Creatione. Sustentatione naturæ lapsæ. Reparatione. Conversione. Justificatione. Sanctificatione & Glorificatione ejusdem.</td> </tr> </table>	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.
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.						

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 $\langle size \rangle$ at `8.5ex` and $\langle adjust \rangle$ at `0ex`. The large brace is too low, so we $\langle adjust \rangle$ to `-1ex`, raising the left side and the delimiter, while lowering the right. We then refine $\langle size \rangle$ 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 theo-	{	ESSENTIAM,	{	Unitate naturæ.
logiæ est Notitia		VOLUNTATEM,		Trinitate personarum.
Dei. Considerat	{	manifestatam in	{	Creatione.
ergo, Dei, vel		operibus ad extra;		Sustentatione naturæ lapsæ.
		ut in		Reparatione.
				Conversione.
				Justificatione.
				Sanctificatione &
				Glorificatione ejusdem.

2.4.3 Going Big

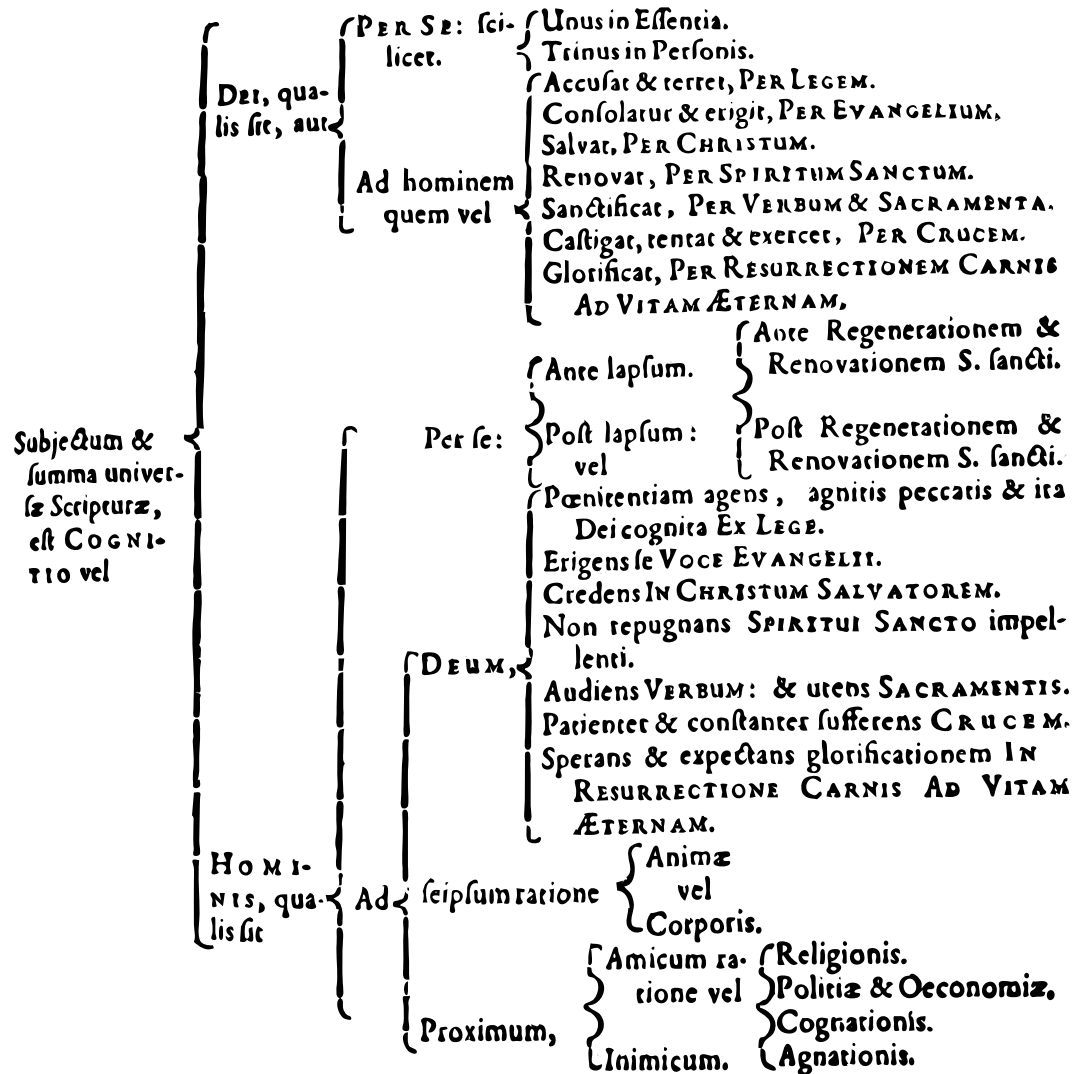
Thus far, we have dealt with many trivial examples. We have amassed a significant body of knowledge:

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 L^AT_EX and P_LA_IN T_EX when using `\vskip`, `\smallskip`, etc., as well as `\newbox`, which is an `\outer` macro in P_LA_IN T_EX. These differences can cause unexpected errors.
7. We may need to use the tweaking macros `\UCschema`, `\LCschema`, `\SwitchSB`, and `\NudgeSB`.

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

⁷Changes in T_EX 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.

⁸This 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:

```

1 \schema
2 {\schemabox{\textsc{Per se}:\ \ scilicet.}}
3 {
4   \schemabox{Unus in essentia.}
5   \schemabox{Trinus in personis.}
6 }

```

$$\text{PER SE: } \left\{ \begin{array}{l} \text{Unus in essentia.} \\ \text{Trinus in personis.} \end{array} \right.$$

```

1 \schema
2 {\schemabox{Ad hominem\ \ quem vel}}
3 {
4   \schemabox{Accusat \& terret, \textsc{Per Legem},\ \
5   Consolatur \& erigit, \textsc{Per Evangelium}.\ \
6   Salvat, \textsc{Per Christum}.\ \
7   Renovat, \textsc{Per Spiritum Sanctum}.\ \
8   Sanctificat, \textsc{Per Verbum} \& \textsc{Sacramenta}.\ \
9   Castigat, tentat \& exercet, \textsc{Per Crucem}.\ \
10  Glorificat \textsc{Per Resurrectionem Carnis}\ \
11  \textsc{\quad Ad Vitam \AE{ }ternam}.}
12 }

```

$$\text{Ad hominem } \left\{ \begin{array}{l} \text{Accusat \& terret, PER LEGEM,} \\ \text{Consolatur \& erigit, PER EVANGELIUM.} \\ \text{Salvat, PER CHRISTUM.} \\ \text{Renovat, PER SPIRITUM SANCTUM.} \\ \text{Sanctificat, PER VERBUM \& SACRAMENTA.} \\ \text{Castigat, tentat \& exercet, PER CRUCEM.} \\ \text{Glorificat PER RESURRECTIONEM CARNIS} \\ \text{AD VITAM \AE{ }TERNAM.} \end{array} \right.$$

```

1 \schemabox{Ante lapsum.}
2
3 \schema
4 {\schemabox{Post lapsum:}}
5 {
6   \schemabox{Ante Regenerationem \&\ \
7   Renovationem S. Sancti.}
8   \schemabox{Post Regenerationem \&\ \
9   Renovationem S. Sancti.}
10 }

```

$$\text{Ante lapsum.}^9$$

$$\text{Post lapsum: } \left\{ \begin{array}{l} \text{Ante Regenerationem \&} \\ \text{Renovationem S. Sancti.} \\ \text{Post Regenerationem \&} \\ \text{Renovationem S. Sancti.} \end{array} \right.$$

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

```

1 \schema
2 {\schemabox{\textsc{Deum},}}
3 {
4   \schemabox{P\oe{}nitentia agens, agnitis peccatis \&\
5   ira Dei cognita \textsc{Ex Lege}.\}
6   Erigens se \textsc{Voce Evangelii}.\}
7   Credens \textsc{In Christum Salvatorem}.\}
8   Non repugnans \textsc{Spiritui Sancto} impellenti.\}
9   Audiens \textsc{Verbum}: \& utens \textsc{Sacramentis}.\}
10  Patienter \& constanter sufferens \textsc{Crucem}.\}
11  Sperans \& expectans glorificationem\
12  \textsc{\quad In Resurrectione Carnis}\}
13  \textsc{\quad Ad Vitam \AE{}ternam}.)}
14 }

```

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.

```

1 \schema
2 {\schemabox{seipsum ratione}}
3 {\schemabox{Anim\ae{} \ vel \ Corporis}}

```

seipsum ratione {

- Animæ
- vel
- Corporis

```

1 \schema
2 {\schemabox{Amicum ra-\ tione vel}}
3 {
4   \schemabox{Religionis.\}
5   Politic\ae{} \& \OE{}conomic\ae{}.\}
6   Cognationis.\}
7   Agnationis.}
8 }
9
10 \schemabox{Inimicum.}

```

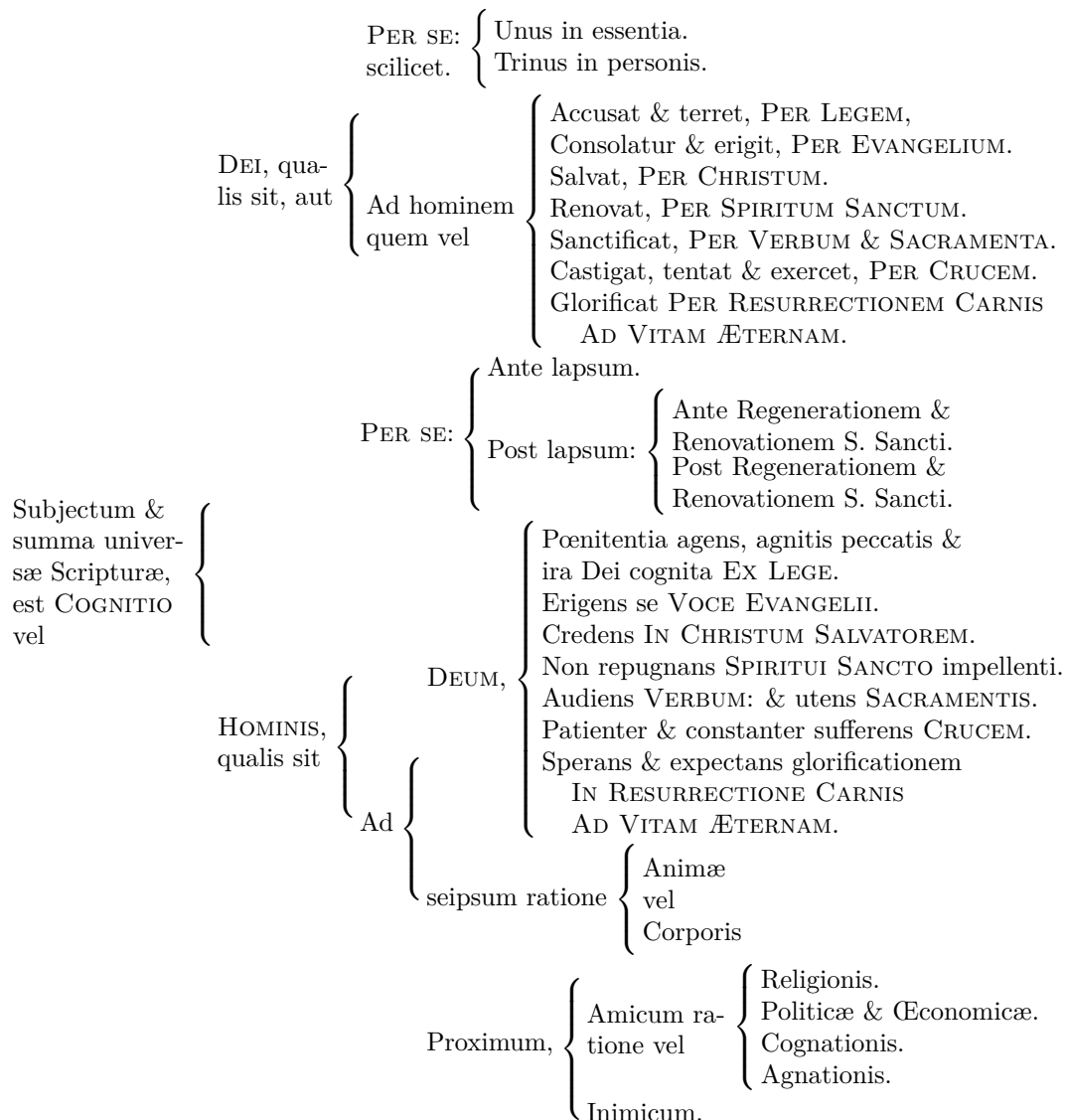
Amicum ra- {

- Religionis.
- Politicae & Economicae.
- Cognitionis.
- Agnationis.

Inimicum.¹⁰

¹⁰We 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:

```

1 \Schema{0ex}{5ex}
2 {
3   \schemabox{Subjectum \&\}
4   summa univer-\}
5   s\ae{} Scriptur\ae{},\}
6   est \textsc{Cognitio}\}
7   vel}
8 }
9 {

```

```

10 \Schema{0ex}{5ex}
11 {
12   \schemabox{\textsc{Dei}, qua-\llis sit, aut}
13 }
14 {
15   \schema
16   {\schemabox{\textsc{Per se}:\ll scilicet.}}
17   {
18     \schemabox{Unus in essentia.}
19     \schemabox{Trinus in personis.}
20   }
21   \schema
22   {\schemabox{Ad hominem\ll quem vel}}
23   {
24     \schemabox{Accusat \& terret, \textsc{Per Legem},\ll
25     Consolatur \& erigit, \textsc{Per Evangelium}.\ll
26     Salvat, \textsc{Per Christum}.\ll
27     Renovat, \textsc{Per Spiritum Sanctum}.\ll
28     Sanctificat, \textsc{Per Verbum} \& \textsc{Sacramenta}.\ll
29     Castigat, tentat \& exercet, \textsc{Per Crucem}.\ll
30     Glorificat \textsc{Per Resurrectionem Carnis}\ll
31     \textsc{\quad Ad Vitam \AE{}ternam}.}
32   }
33 }
34 \Schema{0ex}{5ex}
35 {
36   \schemabox{\textsc{Hominis},\ll qualis sit}
37 }
38 {
39   \Schema{0ex}{5ex}
40   {\schemabox{\textsc{Per se}:}}
41   {
42     \schemabox{Ante lapsum.}
43     \schema
44     {\schemabox{Post lapsum:}}
45     {
46       \schemabox{Ante Regenerationem \&\ll
47       Renovationem S. Sancti.}
48       \schemabox{Post Regenerationem \&\ll
49       Renovationem S. Sancti.}
50     }
51   }
52   \Schema{0ex}{5ex}
53   {\schemabox{Ad}}
54   {
55     \schema
56     {\schemabox{\textsc{Deum},}}
57     {
58       \schemabox{P\oe{}nitentia agens, agnitis peccatis \&\ll
59       ira Dei cognita \textsc{Ex Lege}.\ll
60       Erigens se \textsc{Voce Evangelii}.\ll
61       Credens \textsc{In Christum Salvatorem}.\ll
62       Non repugnans \textsc{Spiritui Sancto} impellentii.\ll
63       Audiens \textsc{Verbum}: \& utens \textsc{Sacramentis}.\ll
64       Patienter \& constanter sufferens \textsc{Crucem}.\ll
65       Sperans \& expectans glorificationem\ll

```

```

66     \textsc{\quad In Resurrectione Carnis}\}
67     \textsc{\quad Ad Vitam \AE{}ternam}.)}
68   }
69   \schema
70     {\schemabox{seipsum ratione}}
71     {\schemabox{Anim\ae{} \} \} vel \} Corporis}}
72   \Schema{0ex}{5ex}
73   {\schemabox{Proximum,}}
74   {
75     \schema
76     {\schemabox{Amicum ra-\} tione vel}}
77   {
78     \schemabox{Religionis.\}
79     Politic\ae{} \& \OE{}conomic\ae{}.\}
80     Cognationis.\}
81     Agnationis.}
82   }
83   \schemabox{Inimicum.}
84 }
85 }
86 }
87 }

```

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`.

```

17   {
18     \schemabox{Unus in essentia.}\smallskip
19     \schemabox{Trinus in personis.}
20   }\smallskip

```

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

```

31     \textsc{\quad Ad Vitam \AE{}ternam}.)}
32   }\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 `\schemaboxes`, 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.

```

41   {
42     \schemabox{Ante lapsum.}\smallskip
43     \schema
44     {\schemabox{Post lapsum:}}
45     {
46       \schemabox{Ante Regenerationem \&\}
47       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.

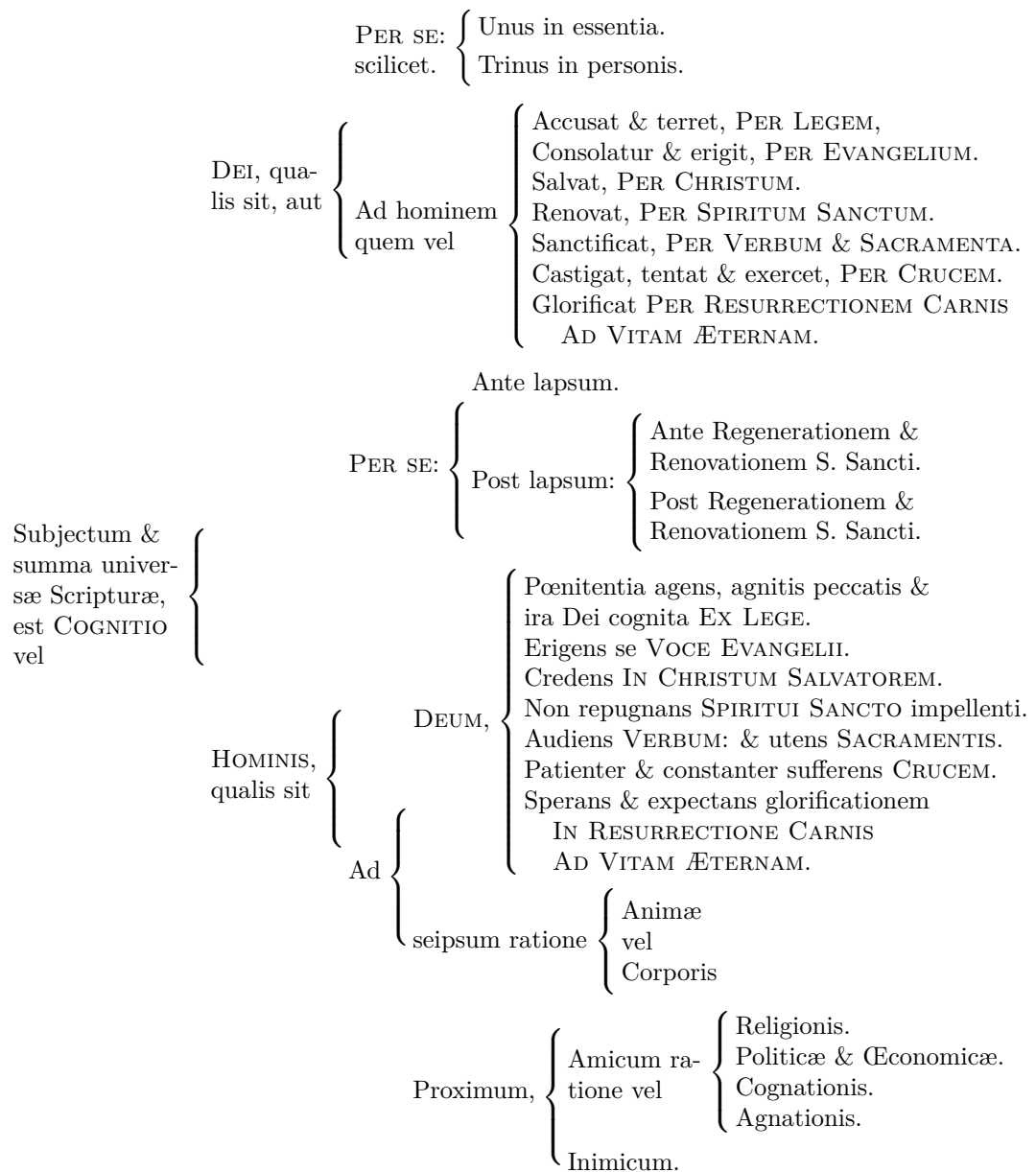
```

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 $\langle size \rangle$. The following lines illustrate our “ball park” figures:

```

1 \Schema{0ex}{23ex}

10 \Schema{0ex}{8ex}

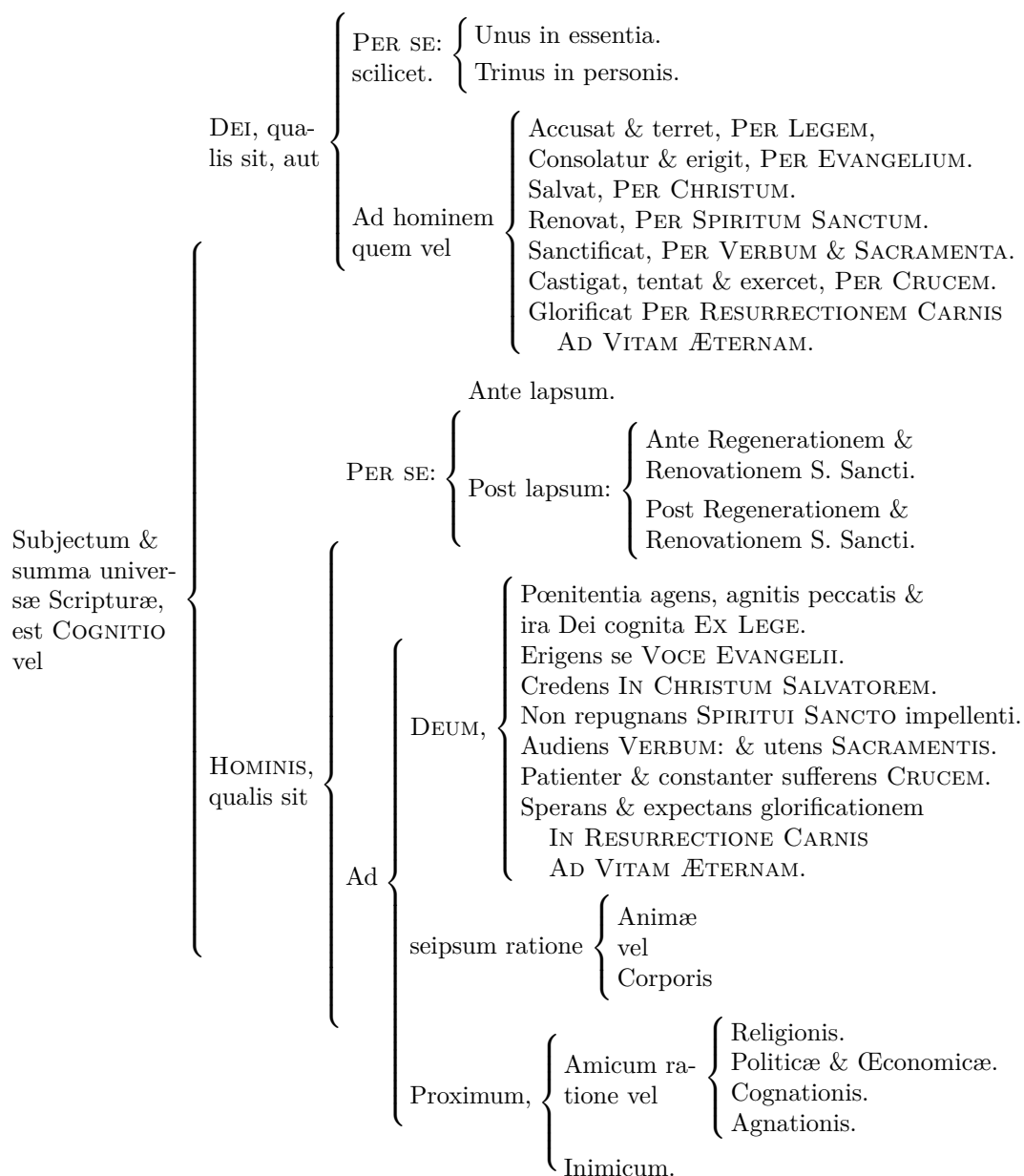
34 \Schema{0ex}{16ex}

39 \Schema{0ex}{5ex}

52 \Schema{0ex}{16ex}

72 \Schema{0ex}{5ex}

```

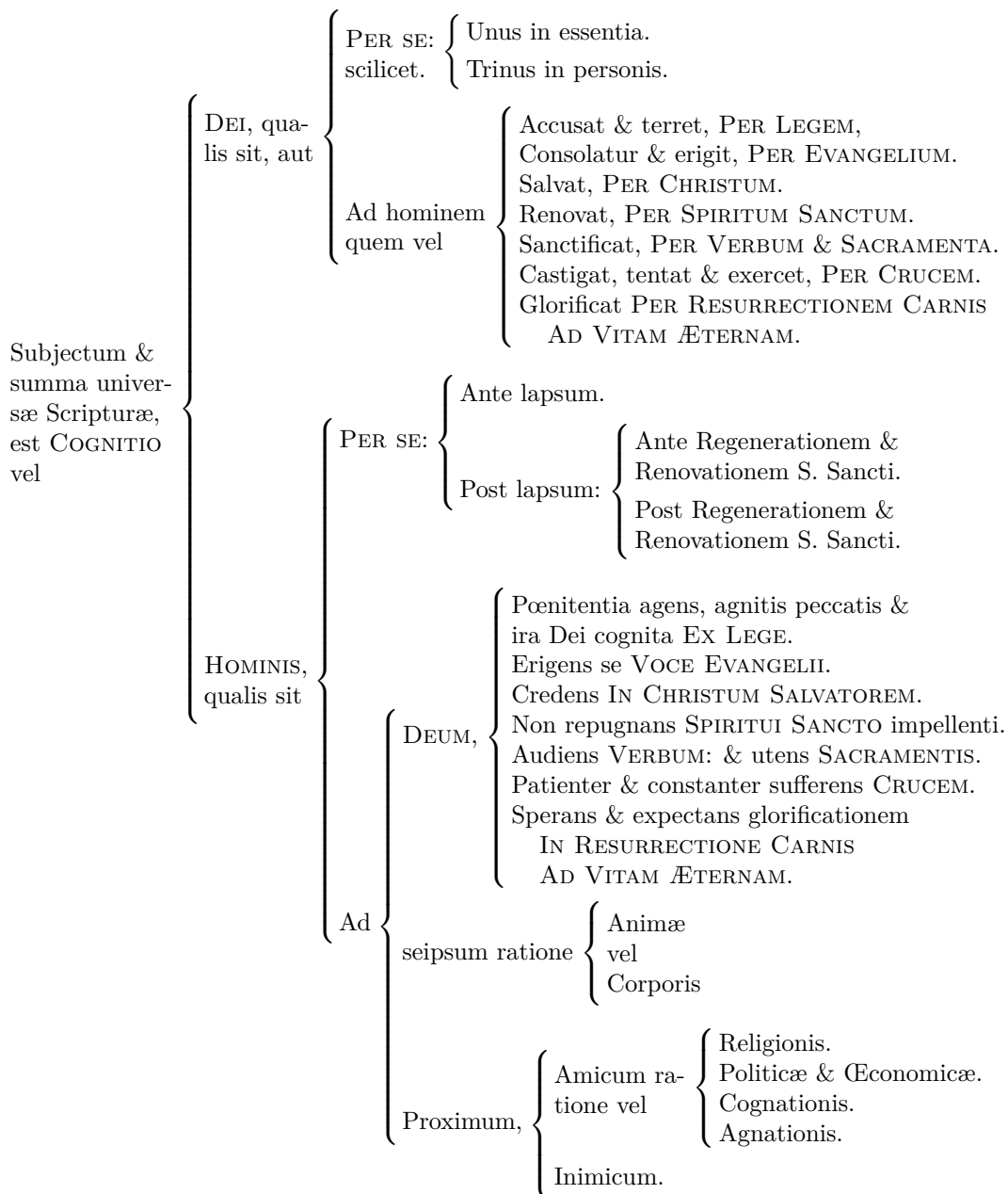


Next we tweak $\langle adjust \rangle$ values by counting the lines (ex) in the direction the left side needs to move relative to the right, multiply the result by two, and make it negative for up and positive for down. Using an editor, e.g., `texworks` makes this fairly easy. We also adjust the final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

```

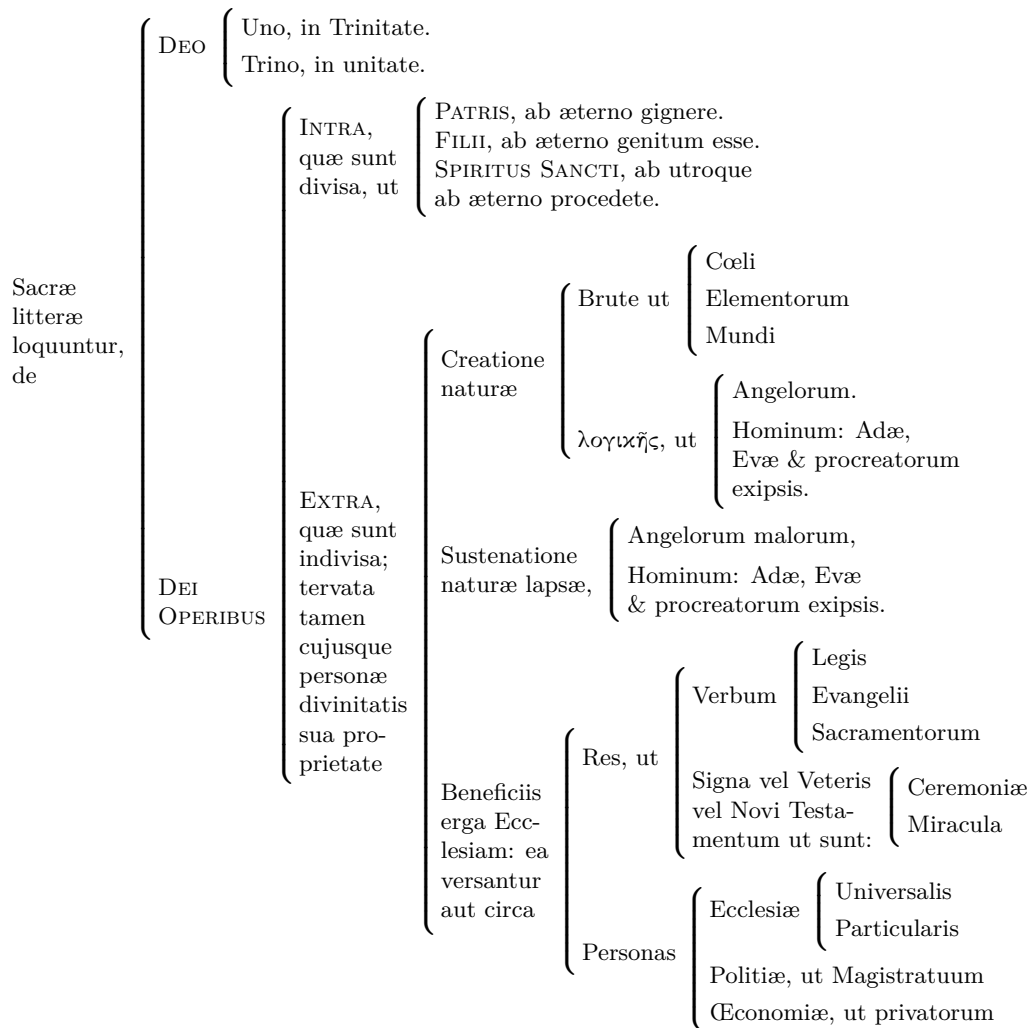
1 \Schema{-25ex}{20.6ex}% Do this one last. ‘‘Subjectum’’
10 \Schema{-6.4ex}{8.5ex}% Do this one first. ‘‘Dei’’
34 \Schema{-13.4ex}{17.4ex}% Do this one fifth. ‘‘Hominis’’
39 \Schema{-4.4ex}{5ex}% Do this one second. ‘‘Per se’’ (lower)
52 \Schema{4.2ex}{14.4ex}% Do this one fourth. ‘‘Ad’’
72 \Schema{2ex}{5.1ex}% Do this one third. ‘‘Proximum’’

```



2.4.4 Big Schema with Groups

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



2.4.5 Open and Closed Schemata

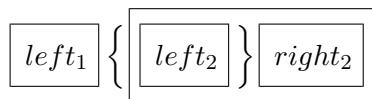
Now we look at schemata that have both open and closed braces. One must use `\Schema` to get delimiters to be the same height. These schemata take the form:

```

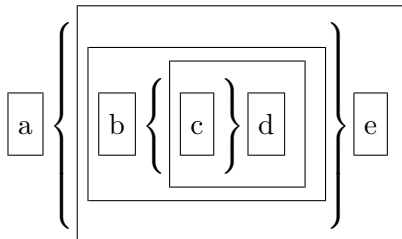
\Schema{<adjust>}{<height>}
{<left1>}
{
  \Schema[close]{<adjust>}{<height>}
  {<left2>}
  {<right2>}
}

```

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:



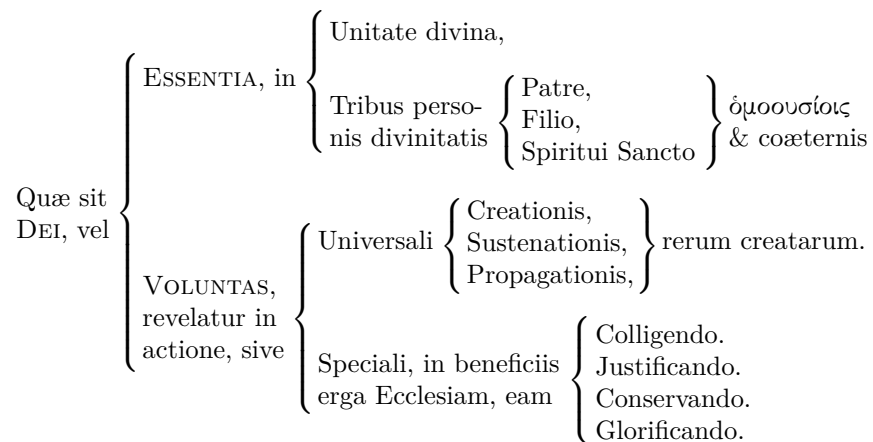
Here is another, more complex example:



```

1 \newbox\mybox
2 \def\Box#1{%
3   \setbox\mybox=\hbox{\vrule\vbox{\hrule%
4     \vfil\hbox{\strut\space #1\space}%
5     \vfil\hrule}\vrule}%
6   \dimen0=\ht\mybox%
7   \advance\dimen0 by2ex%
8   \hbox{\vrule\vbox to \dimen0{\hrule%
9     \vfil\hbox{\Large\strut}\space #1\space}%
10    \vfil\hrule}\vrule}}
11 \Schema{0ex}{6ex}
12 {\Box{a}}
13 {%
14   \Box{%
15     \Schema[close]{0ex}{6ex}
16     {%
17       \Box{%
18         \Schema{0ex}{3ex}
19         {\Box{b}}
20         {%
21           \Box{%
22             \Schema[close]{0ex}{3ex}
23             {\hbox{\Box{c}\kern0.2em}}
24             {\Box{d}}
25           }
26         }
27       }
28     }
29   {\Box{e}}
30 }
31 }
```

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 `ὁμοουσίους`.

```

1 \Schema{-1.4ex}{10ex}
2 {\schemabox{Qu\ae{ } sit\ \ \textsc{Dei}, vel}}
3 {
4   \Schema{-1ex}{5ex}
5   {\schemabox{\textsc{Essentia}, in}}
6   {
7     \vskip1ex\schemabox{Unitate divina,}
8     \medskip
9     \Schema{0ex}{3.4ex}
10    {\schemabox{Tribus perso-\ \ nis divinitatis}}
11    {
12      \Schema[close]{0ex}{3.4ex}
13      {\NudgeSB\schemabox{Patre,\ \ Filio,\ \ Spiritui Sancto}}
14      {\schemabox{\gk{xxxxx}{<omooous'iois}\ \ \& co\ae{ }ternis}}
15    }
16  }
17  \medskip
18  \Schema{-0.2ex}{6.4ex}
19  {\schemabox{\textsc{Voluntas},\ \ revelatur in\ \ actione, sive}}
20  {
21    \Schema{0ex}{3.4ex}
22    {\schemabox{Universali}}
23    {
24      \Schema[close]{0ex}{3.4ex}
25      {\schemabox{Creationis,\ \ Sustentionis,\ \ Propagationis,}}
26      {\schemabox{rerum creatarum.}}
27    }
28    \medskip
29    \schema
30    {\schemabox{Speciali, in beneficiis\ \ erga Ecclesiam, eam}}
31    {\schemabox{Colligendo.\ \ Justificando.\ \
32      Conservando.\ \ Glorificando.}}
33  }
34 }

```

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`.

Curricula Texts	}	I. General Studies	[1. Collected Works 2. Encyclopedias
	}	II. Literary Disciplines	[1. Philology 2. Historical Introduction 3. Literary Theory 4. Application
	}	III. Philosophical Disciplines	[1. Source Texts 2. History of Philosophy 3. General Surveys 4. Specific Studies
	}	IV. Historical Disciplines	[1. General Surveys 2. Specialized Works

```

1 \Schema{-0.2ex}{14.4ex}
2 {\schemabox{\bfseries Curricula\\\bfseries Texts}}
3 {
4   \DoBrackets%
5   % \newbox here is doable in LaTeX, not in Plain TeX,
6   % where it must be used as an \outer macro.
7   \newbox\mybox%
8   \setbox\mybox=\hbox{\bfseries III. Philosophical }%
9   \dimen0=\wd\mybox%
10  \schema
11    {\schemabox[\dimen0]{\bfseries I. General\\Studies}}
12    {\schemabox{1. Collected Works\\2. Encyclopedias}}
13  \smallskip
14  \schema
15    {\schemabox[\dimen0]{\bfseries II. Literary\\Disciplines}}
16    {\schemabox{1. Philology\\
17      2. Historical Introduction\\
18      3. Literary Theory\\
19      4. Application}}
20  \smallskip
21  \schema
22    {\schemabox[\dimen0]{\bfseries III. Philosophical\\Disciplines}}
23    {\schemabox{1. Source Texts\\
24      2. History of Philosophy\\
25      3. General Surveys\\
26      4. Specific Studies}}
27  \smallskip
28  \schema
29    {\schemabox[\dimen0]{\bfseries IV. Historical\\Disciplines}}
30    {\schemabox{1. General Surveys\\
31      2. Specialized Works}}
32 }

```

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 “ $\LaTeX 2\epsilon$ ” then the macros use the $\LaTeX 2\epsilon$ 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 $\LaTeX 2\epsilon$, we do the equivalent of `\makeatletter` in either `PLAIN \TeX` or `eplain`.

```
1 %<package>\expandafter\ifx \curname schemataLaTeX\endcurname\relax
2 %<package> \def\schemataLaTeX{LaTeX2e}\fi
3 %<package>\ifx\fmtname\schemataLaTeX
4 %<package>\expandafter\NeedsTeXFormat\expandafter{\schemataLaTeX}[2005/12/01]
5 %<package>\ProvidesPackage{schemata}
6 %<*package>
7 [2020/12/02 1.3 generic package to aid construction of topical categories]
8 %</package>
9 %<package>\else
10 %<package>\catcode'@=11\relax
11 %<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.

```
12 \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.

```
13 \newbox\@schemata@rhs%
14 \newbox\@schemata@lhs%
15 \newdimen\@schemata@rheight%
16 \newdimen\@schemata@lheight%
17 \newdimen\@schemata@one%
18 \newdimen\@schemata@two%
19 \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.

```
20 \newif\if@schemata@LCBox%
21 \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).

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

3.2 Package Options

We set braces to be the default set of delimiters. Apart from $\text{\LaTeX} 2_{\epsilon}$ 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.

```
23 \ifx\fmtname\schemata@LaTeX
24   \DeclareOption{braces}%
25     {\let\schemata@LD\lbrace%
26       \let\schemata@RD\rbrace}
27   \DeclareOption{brackets}%
28     {\let\schemata@LD\lbrack%
29       \let\schemata@RD\rbrack}
30   \DeclareOption{parens}%
31     {\let\schemata@LD(%
32       \let\schemata@RD)}
33   \DeclareOption{groups}%
34     {\let\schemata@LD\lgroup%
35       \let\schemata@RD\rgroup}
36   \ExecuteOptions{braces}%
37   \ProcessOptions\relax
38 \else
39   \let\schemata@LD\lbrace%
40   \let\schemata@RD\rbrace%
41 \fi
```

3.3 Macros

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

```
42 \ifx\fmtname\schemata@LaTeX
43   \newcommand*\DoBraces%
44     {\let\schemata@LD\lbrace%
45       \let\schemata@RD\rbrace}
46 \else
47   \def\DoBraces%
48     {\let\schemata@LD\lbrace%
49       \let\schemata@RD\rbrace}
50 \fi
```

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

```
51 \ifx\fmtname\schemata@LaTeX
52   \newcommand*\DoBrackets%
53     {\let\schemata@LD\lbrack%
54       \let\schemata@RD\rbrack}
55 \else
56   \def\DoBrackets%
57     {\let\schemata@LD\lbrack%
58       \let\schemata@RD\rbrack}
59 \fi
```

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

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



```

64 \else
65   \def\DoParens%
66     {\let\schemata@LD(%
67       \let\schemata@RD)}
68 \fi

```

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

```

69 \ifx\fmtname\schemata@LaTeX
70   \newcommand*\DoGroups%
71     {\let\schemata@LD\lgroup%
72       \let\schemata@RD\rgroup}
73 \else
74   \def\DoGroups%
75     {\let\schemata@LD\lgroup%
76       \let\schemata@RD\rgroup}
77 \fi

```

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

```

78 \ifx\fmtname\schemata@LaTeX
79   \newcommand*\LCschema{\schemata@LCBoxtrue}
80 \else
81   \def\LCschema{\schemata@LCBoxtrue}
82 \fi

```

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

```

83 \ifx\fmtname\schemata@LaTeX
84   \newcommand*\UCschema{\schemata@LCBoxfalse}
85 \else
86   \def\UCschema{\schemata@LCBoxfalse}
87 \fi

```

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

```

88 \ifx\fmtname\schemata@LaTeX
89   \newcommand*\SwitchSB{\schemata@SWBoxtrue}%
90 \else
91   \def\SwitchSB{\schemata@SWBoxtrue}
92 \fi

```

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

```

93 \ifx\fmtname\schemata@LaTeX
94   \newcommand*\NudgeSB{\schemata@NudgeBoxtrue}
95 \else
96   \def\NudgeSB{\schemata@NudgeBoxtrue}
97 \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.

```

98 \ifx\fmtname\schemata@LaTeX
99   \newcommand*\SBNudgeFactor{\kern0.2em}
100 \else
101   \def\SBNudgeFactor{\kern0.2em}
102 \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.

```

103 \ifx\fmtname\@schemata@LaTeX
104   \newcommand*{\schemabox}[2][Opt]{\@schemata@schemabox[#1]{#2}}
105 \else
106   \def\schemabox{\futurelet\@schemata@testchar\@schemata@schemab@x}
107   \def\@schemata@schemab@x{%
108     \ifx[\@schemata@testchar
109       \let\next\@schemata@schemabox%
110     \else
111       \let\next\@schemata@@schemab@x%
112     \fi
113     \next%
114   }%
115   \def\@schemata@@schemab@x#1{\@schemata@schemabox[Opt]{#1}}
116 \fi
117 \def\@schemata@schemabox[#1]#2{%
118   \ifinner
119     \if@schemata@LCBox
120       \def\@schemata@Adj{}%
121       \if@schemata@SWBox\def\@schemata@Adj{\strut}\fi
122     \else
123       \def\@schemata@Adj{\strut}%
124       \if@schemata@SWBox\def\@schemata@Adj{ }\fi
125     \fi
126     \if@schemata@NudgeBox
127       \let\@schemata@Nudge\@SBNudgeFactor%
128     \else
129       \def\@schemata@Nudge{ }%
130     \fi
131     \ifdim#1<1pt
132       \def\{\{\@schemata@Nudge\egroup\hbox\bgroup\ignorespaces }%
133       \vbox{\hbox\bgroup%
134         \@schemata@Adj\ignorespaces #2\@schemata@Nudge%
135         \egroup}%
136     \else
137       \def\{\{\hfil\egroup\hbox to #1\bgroup\ignorespaces }%
138       \vbox{\hbox to #1\bgroup%
139         \@schemata@Adj\ignorespaces #2\hfil%
140         \egroup}%
141     \fi
142   \else
143     #2%
144   \fi
145   \@schemata@SWBoxfalse%
146   \@schemata@NudgeBoxfalse%
147 }

```

`\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.

```

148 \ifx\fmtname\@schemata@LaTeX
149   \newcommand{\schema}[3][open]{%
150     \@schemata@schema[#1]{#2}{#3}}
151 \else
152   \long\def\schema{\futurelet\@schemata@testchar\@schemata@schem@}
153   \long\def\@schemata@schem@{%
154     \ifx[\@schemata@testchar
155       \let\next\@schemata@schema%
156     \else
157       \let\next\@schemata@@schem@%
158     \fi
159     \next%
160   }%
161   \long\def\@schemata@@schem@#1#2{%
162     \@schemata@schema[open]{#1}{#2}}
163 \fi
164 \long\def\@schemata@schema[#1]#2#3{%
165   \def\@schemata@option{#1}\def\@schemata@open{open}%
166   \ifx\@schemata@option\@schemata@open
167     \setbox\@schemata@rhs=\vbox{#3}%
168     \@schemata@rheight=\ht\@schemata@rhs%
169     \advance\@schemata@rheight\dp\@schemata@rhs%
170     \advance\@schemata@rheight by 1.44265ex%
171     \hbox{\$ \vcenter{#2}%
172           \@schemata@lbrace{\@schemata@rheight}%
173           \vcenter{#3}$}%
174   \else
175     \setbox\@schemata@lhs=\vbox{#2}%
176     \@schemata@lheight=\ht\@schemata@lhs%
177     \advance\@schemata@lheight\dp\@schemata@lhs%
178     \advance\@schemata@lheight by 1.44265ex%
179     \hbox{\$ \vcenter{#2}%
180           \kern-0.2em\@schemata@rbrace{\@schemata@lheight}%
181           \vcenter{#3}$}%
182   \fi
183 }
```

`\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 sides. 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.

```

184 \ifx\fmtname\@schemata@LaTeX
185   \newcommand{\Schema}[5][open]{%
186     \@schemata@Schema[#1]{#2}{#3}{#4}{#5}}
187 \else
188   \long\def\Schema{\futurelet\@schemata@testchar\@schemata@Schem@}
189   \long\def\@schemata@Schem@{%
190     \ifx[\@schemata@testchar
191       \let\next\@schemata@Schema%
192     \else
193       \let\next\@schemata@@Schem@%
194     \fi
195     \next%
196   }%
197   \long\def\@schemata@@Schem@#1#2#3#4{%
198     \@schemata@Schema[open]{#1}{#2}{#3}{#4}}
199 \fi
200 \long\def\@schemata@Schema[#1]#2#3#4#5{%
201   \def\@schemata@option{#1}%
202   \def\@schemata@open{open}%
203   \@schemata@one=#2%
204   \ifx\@schemata@option\@schemata@open
205     \hbox{${\vcenter{\vskip1.44265\@schemata@one#4}}%
206       \@schemata@biglbrace{#2}{#3}\vcenter{#5}$}%
207   \else
208     \hbox{${\vcenter{\vskip1.44265\@schemata@one#4}\kern-0.2em%
209       \@schemata@bigrbrace{#2}{#3}\vcenter{#5}$}%
210   \fi
211 }

```

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

```

212 \ifx\fmtname\@schemata@LaTeX
213   \newcommand*{\@schemata@lbrace}[1]{%
214     \ifmmode
215       \left.\vcenter{\vbox to #1{\vfil}}\right\@schemata@LD%
216     \fi
217   }
218 \else
219   \def\@schemata@lbrace#1{%
220     \ifmmode
221       \left.\vcenter{\vbox to #1{\vfil}}\right\@schemata@LD%
222     \fi
223   }
224 \fi

```

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

```
225 \ifx\fmtname\@schemata@LaTeX
226   \newcommand*\@schemata@rbrace}[1]{%
227     \ifmmode
228       \left\@schemata@RD\vcenter{\vbox to #1{\vfil}}\right.%
229     \fi
230   }
231 \else
232   \def\@schemata@rbrace#1{%
233     \ifmmode
234       \left\@schemata@RD\vcenter{\vbox to #1{\vfil}}\right.%
235     \fi
236   }
237 \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.

```
238 \ifx\fmtname\@schemata@LaTeX
239   \newcommand*\@schemata@biglbrace}[2]{%
240     \@schemata@@biglbrace{#1}{#2}}
241 \else
242   \def\@schemata@biglbrace#1#2{%
243     \@schemata@@biglbrace{#1}{#2}}
244 \fi
245 \def\@schemata@@biglbrace#1#2{%
246   \@schemata@one=#1%
247   \@schemata@two=#2%
248   \@schemata@three=-\@schemata@two%
249   \ifdim\@schemata@three>\@schemata@two%
250     \@schemata@two=\@schemata@three\fi
251   \ifdim\@schemata@one<0pt
252     \ifmmode\vcenter{\hbox{\$}\left.%
253       \vbox to 1.44265\@schemata@two{\vfil}}%
254       \right\@schemata@LD%
255       \atop\vbox to -1.44265\@schemata@one{\vfil}$}}\fi
256   \else
257     \ifmmode\vcenter{\hbox{\$}\vbox to 1.44265\@schemata@one{\vfil}}%
258       \atop\left.%
259       \vbox to 1.44265\@schemata@two{\vfil}}%
260       \right\@schemata@LD$}}\fi
261   \fi
262 }
```

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

```

263 \ifx\fmtname\@schemata@LaTeX
264   \newcommand*\@schemata@bigbrace}[2]{%
265     \@schemata@@bigbrace{#1}{#2}%
266   }
267 \else
268   \def\@schemata@bigbrace#1#2{%
269     \@schemata@@bigbrace{#1}{#2}%
270   }
271 \fi
272 \def\@schemata@@bigbrace#1#2{%
273   \@schemata@one=#1%
274   \@schemata@two=#2%
275   \@schemata@three=-\@schemata@two%
276   \ifdim\@schemata@three>\@schemata@two%
277     \@schemata@two=\@schemata@three\fi
278   \ifdim\@schemata@one<0pt
279     \ifmode\vcenter{\hbox{\$left.%
280       \vbox to 1.44265\@schemata@two{\vfil}%
281       \right\@schemata@RD%
282       \atop\vbox to -1.44265\@schemata@one{\vfil}$}}\fi
283   \else
284     \ifmode\vcenter{\hbox{\$vbox to 1.44265\@schemata@one{\vfil}%
285       \atopleft.%
286       \vbox to 1.44265\@schemata@two{\vfil}%
287       \right\@schemata@RD$}}\fi
288   \fi
289 }

```

If we are not using $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$, we do the equivalent of `\makeatother`.

```

290 \ifx\fmtname\@schemata@LaTeX\else
291   \catcode'\@=12\relax
292 \fi

```

4 Change History

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

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	<code>\@schemata@open</code>	H
<code>\@schemata@@Schem@</code> 193, 197 165, 166, 202, 204	<code>\ht</code> 168, 176
<code>\@schemata@@biglbrace</code>	<code>\@schemata@option</code>	I
. 240, 243, 245 165, 166, 201, 204	<code>\if@schemata@LCBox</code> . 20, 119
<code>\@schemata@@bigrbrace</code>	<code>\@schemata@rbrace</code> . 180, <u>225</u>	<code>\if@schemata@NudgeBox</code> 22, 126
. 265, 269, 272	<code>\@schemata@rheight</code>	<code>\if@schemata@SWBox</code>
<code>\@schemata@@schem@</code> 157, 161 15, 168, 169, 170, 172 21, 121, 124
<code>\@schemata@@schemab@x</code>	<code>\@schemata@rhs</code>	<code>\ifinner</code> 118
. 111, 115 13, 167, 168, 169	K
<code>\@schemata@Adj</code> 120,	<code>\@schemata@schem@</code> . 152, 153	<code>\kern</code> 99, 101, 180, 208
121, 123, 124, 134, 139	<code>\@schemata@schema</code>	L
<code>\@schemata@LCBoxfalse</code> 84, 86 150, 155, 162, 164	<code>\lbrace</code> 25, 39, 44, 48
<code>\@schemata@LCBoxtrue</code> . 79, 81	<code>\@schemata@schemab@x</code> 106, 107	<code>\lbrack</code> 28, 53, 57
<code>\@schemata@LD</code>	<code>\@schemata@schemab@x</code>	<code>\LCschema</code> 6, <u>78</u>
. 25, 28, 31, 34, 39, 104, 109, 115, 117	<code>\left</code> 215, 221,
44, 48, 53, 57, 62, 66,	<code>\@schemata@testchar</code> 106,	228, 234, 252, 258, 279, 285
71, 75, 215, 221, 254, 260	108, 152, 154, 188, 190	<code>\lgroup</code> 34, 71, 75
<code>\@schemata@LaTeX</code> <u>12</u> ,	<code>\@schemata@three</code> 19,	<code>\long</code> 152, 153,
23, 42, 51, 60, 69, 78,	248, 249, 250, 275, 276, 277	161, 164, 188, 189, 197, 200
83, 88, 93, 98, 103, 148,	<code>\@schemata@two</code> 18, 247,	N
184, 212, 225, 238, 263, 290	248, 249, 250, 253, 259,	<code>\NudgeSB</code> 7, <u>93</u>
<code>\@schemata@Nudge</code>	274, 275, 276, 277, 280, 286	R
. 127, 129, 132, 134	<code>\@schemata@two</code> 132, 137	<code>\rbrace</code> 26, 40, 45, 49
<code>\@schemata@NudgeBoxfalse</code> 146	A	<code>\rbrack</code> 29, 54, 58
<code>\@schemata@NudgeBoxtrue</code>	<code>\advance</code> 169, 170, 177, 178	<code>\rgroup</code> 35, 72, 76
. 94, 96	<code>\atop</code> 255, 258, 282, 285	<code>\right</code> 215, 221,
<code>\@schemata@RD</code>	C	228, 234, 254, 260, 281, 287
. 26, 29, 32, 35, 40,	<code>\catcode</code> 291	S
45, 49, 54, 58, 63, 67,	D	<code>\SBNudgeFactor</code> 7, <u>98</u> , 127
72, 76, 228, 234, 281, 287	<code>\DoBraces</code> 3, <u>42</u>	<code>\Schema</code> 4, <u>184</u>
<code>\@schemata@SWBoxfalse</code> . . 145	<code>\DoBrackets</code> 3, <u>51</u>	<code>\schema</code> 3, <u>148</u>
<code>\@schemata@SWBoxtrue</code> . 89, 91	<code>\DoGroups</code> 3, <u>69</u>	<code>\schemabox</code> 2, <u>103</u>
<code>\@schemata@Schem@</code> . 188, 189	<code>\DoParens</code> 3, <u>60</u>	<code>\schemataLaTeX</code> <u>1</u> , 2, 12
<code>\@schemata@Schema</code>	<code>\dp</code> 169, 177	<code>\strut</code> 121, 123
. 186, 191, 198, 200	E	<code>\SwitchSB</code> 7, <u>88</u>
<code>\@schemata@biglbrace</code> 206, <u>238</u>	<code>\edef</code> 12	U
<code>\@schemata@bigrbrace</code> 209, <u>263</u>	F	<code>\UCschema</code> 6, <u>83</u>
<code>\@schemata@lbrace</code> . 172, <u>212</u>	<code>\fmtname</code> 23,	
<code>\@schemata@lheight</code>	42, 51, 60, 69, 78, 83,	
. 16, 176, 177, 178, 180	88, 93, 98, 103, 148,	
<code>\@schemata@lhs</code>	184, 212, 225, 238, 263, 290	
. 14, 175, 176, 177		
<code>\@schemata@one</code> 17,		
203, 205, 208, 246, 251,		
255, 257, 273, 278, 282, 284		