The Jolly Writer

Joris van der Hoeven

TABLE OF CONTENTS

Preface

The ultimate aim of a typesetter is to compose books in such a perfect manner that nobody will notice it. Nothing is more dishonorable than to make a reader stumble over an ill-placed comma. What the typesetter wants is to bury you cozily in your armchair, with your complete focus on the book, and in a spell not to be broken by trivial typographic details.

If you are a writer instead of a reader, then it takes far more than an armchair to put you in the right creative state of mind. The ideal text editor should help you to maintain this state for as long as possible. You don't want to be torn out of trance by a misguided comma key that suddenly starts inserting apostrophes. The ideal writing tool should behave as your first art pen from school: it should unleash your desire to write, make you feel one with the tool, and occasionally surprise you with the beauty of your own words.

Back in 1996, when I was writing my PhD in computer science, none of the existing text editors were even close to this ideal. General purpose editors such as MICROSOFT WORD produced documents of poor quality and made it difficult to type mathematical formulas. In order to compose professional-looking scientific documents, the main alternative was to use (LA)TEX. Although this solution was nice from the reader's perspective, it forced writers to encode their prose in a technical pseudo-language. (LA)TEX then relied on an akward "compilation" process in order to transform this pseudo-code into a printable and human-readable document.

After my PhD, this unsatisfactory state of the art led me to start the development of GNU TEXMACS, a free office suite for scientists. Before anything else, TEXMACS allows you to create beautiful scientific documents with special types of content, such as mathematical formulas or technical pictures. TEXMACS also provides interfaces for various external systems for symbolic and numeric computations. Recent versions further include a laptop presentation facility, versioning tools, integrated documentation, etc. In the areas of science and education, the ultimate aim is to provide a complete suite for the most frequent tasks on a computer.

One distinctive feature of TEXMACS is that it does not compromise on quality. First of all, the final documents have a professional typesetting quality, similar or even superior to what is achieved by (LA)TEX. Yet, user-friendliness has not been sacrificed, since the system also provides an intuitive wysiwyg (what-you-see-is-what-you-get) graphical interface. Finally, TEXMACS favors the composition of so-called "structured" documents, in contrast to existing wysiwyg interfaces of general purpose office suites, which are mainly "presentation oriented".

The name "GNU TeXmacs" is explained by the facts that TeXmacs is part of the GNU project and that some initial inspiration was drawn from the (LA)TeX systems [36, 38] and GNU EMACS [53]. However, it has become clear over time that this choice of name was one of the biggest mistakes of the project. Indeed, the name incorrectly suggests that TeXmacs is some kind of interface to (LA)TeX. So let me stress once and for all: the current TeXmacs system is completely independent from (LA)TeX. Of course, TeXmacs does provide converters between its native format and LaTeX.

As a free software, the development of TeXmacs has benefitted from the help of a wide community of contributors around the globe. Particular thanks go to the following main co-developers throughout the years: David ALLOUCHE, Miguel DE BENITO DELGADO, Darcy CHEN, Andrey GROZIN, Massimiliano GUBINELLI, Philippe JOYEZ, Grégoire LECERF, Henri LESOURD, François POULAIN, and Denis RAUX. Further thanks go to the many other contributors and supporters; see `http://www.texmacs.org/tmweb/about/authors.en.html` for a more extensive list. I am indebted to Basile AUDOLY and Kostas OIKONOMOU for their careful proofreading.

I also wish to express my gratitude to those who have provided financial support to the TeXmacs project: CNRS, CRI-TECH de Haute-Savoie, DIGITEO, Rennes MÉTROPÔLE, SPRINGER-VERLAG, INRIA, Dan GRAYSON, and Christoph BENZMUELLER.

TeXmacs is built on top of a lot of other software. The QT, GUILE, FREETYPE, and HUMMUS libraries are particularly essential [47, 33, 63, 34]. Recent versions integrate an increasing amount of artwork, such as the TeX Gyre fonts [32] (the main font of this book is PAGELLA), various "Subtle Patterns" [61], more fonts from DAFONT [9], and further freely available pictures from WIKIMEDIA [65].

Last but not least, I wish to thank SYLVIE, JUDITH, and NIELS for their support and patience.

CHAPTER 4
MATHEMATICS

You enter *math mode* by starting a new formula or by placing the cursor inside an existing formula. In math mode, the menus, the icon toolbars, and the keyboard behavior are adapted so as to facilitate the insertion of mathematical symbols and markup. For instance, typing - > inserts the arrow →.

Recent versions of TeXmacs also include optional "semantic editing" facilities that will briefly be described at the end of this chapter (see [26, 29] for more details). When used appropriately, these allow you to write documents in which all formulas are at least correct from a syntactical point of view. A "syntax corrector" is included to assist you with this task. Syntactic correctness is for instance important when using formulas as inputs for a computer algebra system. Syntactically correct documents are also less likely to contain "typos". Further functionalities, such as semantic search and replace, should be developed in the future.

4.1 Incorporating mathematical formulas

TeXmacs provides three main ways to insert mathematical formulas into the main text:

Inline formulas. Short formulas—such as $a^2+b^2=c^2$—are usually embedded directly into the main text flow. Such *inline formulas* can be inserted using $ or Insert ▸ Mathematics ▸ Inline formula.

The typesetter attempts to squeeze inline formulas as much as possible, so that they do not disrupt the general presentation. For example, the presentation $\lim_{n\to\infty} \log n/n=0$ is preferred over $\displaystyle\lim_{n\to\infty} \log n/n=0$, and $x=\frac{1}{2}$ over $x=\dfrac{1}{2}$. Nevertheless, you can force the more voluminous renderings using Format ▸ Display style ▸ on.

Displayed formulas. Big or important formulas are usually displayed on separate lines:

$$x^n+y^n=z^n.$$

Such *displayed formulas* can be inserted using ⌥$ or Insert ▸ Mathematics ▸ Displayed formula and use a less compressed layout than inline formulas. You may turn an inline formula into a displayed one and *vice versa* using ^⇥. Displayed formulas can also be numbered using the toggle ^# or Focus ▸ Numbered.

Equation arrays. For the presentation of multiple equations, it is best to align them as in the following example:

$$x+0 \;=\; x$$

$$\begin{aligned} x+(-x) &= 0 \\ x+y &= y+x \\ (x+y)+z &= x+(y+z). \end{aligned}$$

A similar layout is often required for step by step computations

$$\begin{aligned} (\mathrm{e}^{\sin x}+\sin \mathrm{e}^x)' &= (\mathrm{e}^{\sin x})'+(\sin \mathrm{e}^x)' \\ &= (\sin x)'\,\mathrm{e}^{\sin x}+(\mathrm{e}^x)'\sin \mathrm{e}^x \\ &= \mathrm{e}^{\sin x}\cos x+\mathrm{e}^x\sin \mathrm{e}^x, \end{aligned}$$

in which case several left-hand members are simply left empty.

In order to create so-called *equation arrays* of this type, you may use ⌥& or Insert ▸ Mathematics ▸ Several equations. The typesetter uses a table with three columns for the rendering, where the first column is aligned to the right, the third one to the left, and the middle column is centered. New rows are created by pressing ↩. In fact, since equation arrays are special forms of tables, all table editing rules apply (see chapter 5). In a sense, they are also somewhat superfluous: you may obtain the same layout by creating an appropriate table inside a displayed formula. Nevertheless, equation arrays are so common in mathematical documents that it is nice to have special markup for them.

4.2 Mathematical symbols

Following the basic design principles of the user interface (see section 2.1), TeXmacs allows you to enter most mathematical symbols in at least four ways: using the menus, the icon toolbars, appropriate keyboard shortcuts, or LaTeX commands.

For example, the binary relation "$\leqslant$" can both be found in the Insert ▸ Symbol ▸ Binary relation menu and in the toolbar menu under the icon $\prec$. When hovering the mouse pointer for a while over the $\leqslant$ in either of these menus, the corresponding keyboard shortcut < = will appear in a help balloon. Users who are familiar with LaTeX, may also enter the symbol using \ l e q s l a n t ↩.

The keyboard shortcuts for mathematical symbols were designed according to a small number of simple rules, which make them easy to remember. Before we go into more details, it should also be noticed that symbols carry a precise syntactical semantics in TeXmacs. In particular, TeXmacs carefully distinguishes between so-called *homoglyphs*, which are distinct symbols that look the same. For more information, see sections 4.9 and 4.11 below.

For example, the vertical bar $|$ can be used as a separator for defining sets $R^{>}=\{x\in R\,|\,x>0\}$, but also as the binary relation "divides" (e.g. $11|1001$), or for restricting the domain of a function: $f|_E$. Such homoglyphs have different binding forces and often come with a different spacing. The most annoying ambiguity is between invisible multiplication $x y$ and function application $\sin x$, which are respectively entered using the shortcuts * and ␣.

As a general rule, we also note that TeXmacs takes care of the spacing inside mathematical formulas. For instance, you enter $a+b$ by typing a + b, not

α	a ⇥	β	b ⇥	γ	g ⇥	δ	d ⇥	ε	e ⇥	ϵ	e ⇥ ⇥ ⇥
ζ	z ⇥	η	h ⇥	θ	j ⇥	ϑ	j ⇥ ⇥ ⇥	ι	i ⇥	κ	k ⇥
$\varkappa$	k ⇥ ⇥	λ	l ⇥	μ	m ⇥	ν	n ⇥	ξ	x ⇥	o	o ⇥
π	p ⇥	ϖ	p ⇥ ⇥ ⇥	ρ	r ⇥	ϱ	r ⇥ ⇥	σ	s ⇥	ς	s ⇥ ⇥
τ	t ⇥	υ	u ⇥	φ	f ⇥	ϕ	f ⇥ ⇥	ψ	y ⇥	χ	q ⇥
ω	w ⇥										
A	A ⇥	B	B ⇥	Γ	G ⇥	Δ	D ⇥	E	E ⇥	Z	Z ⇥
H	H ⇥	Θ	J ⇥	I	I ⇥	K	K ⇥	Λ	L ⇥	M	M ⇥
N	N ⇥	Ξ	X ⇥	O	O ⇥	Π	P ⇥	P	R ⇥	Σ	S ⇥
T	T ⇥	Υ	U ⇥	Φ	F ⇥	Ψ	Y ⇥	X	Q ⇥	Ω	W ⇥

Table 4.1. Keyboard shortcuts for Greek characters.

a␣+␣b. The space bar is reserved for function application: typing s i n ␣ x produces $\sin x$.

4.2.1 Letter-like symbols

Mathematicians like to use Greek characters as well as letters in several special fonts for particular purposes. For instance, the set of natural numbers is typically denoted by $\mathbb{N}$ and a maximal ideal by $\mathfrak{m}$.

Greek characters can be obtained using the Tab key ⇥, as variants of the usual Roman letters. For instance, you may enter β and Λ using b ⇥ and L ⇥. Table 4.1 shows keyboard shortcuts for the complete Greek alphabet. Notice that the Greek letters ε, θ, κ, π, ρ, σ, and φ admit alternative renderings ϵ, ϑ, $\varkappa$, ϖ, ϱ, ς, and ϕ.

TeXmacs reserves the function keys F5, F6, F7, and F8 for entering characters in special mathematical fonts:

F5. This is the keyboard prefix for upright mathematical symbols. For example, F5 S produces S instead of S, whereas F5 l ⇥ produces λ instead of λ.

F6. This keyboard prefix is used for producing bold letters such as $\boldsymbol{v}$ (F6 v) or $\boldsymbol{S}$ (F6 S). You may combine it with the other prefixes F5, F7, and F8. For example, F6 F5 v yields **v** and F6 F7 X yields $\boldsymbol{\mathcal{X}}$.

⇧F6. You may use this prefix for "blackboard bold" fonts. The classical sets $\mathbb{C}$, $\mathbb{N}$, $\mathbb{Q}$, $\mathbb{R}$, and $\mathbb{Z}$ can for example be obtained using the shortcuts ⇧F6 C, ⇧F6 N, ⇧F6 Q, ⇧F6 R, and ⇧F6 Z. An even easier way to obtain these symbols is using C C, N N, Q Q, R R, and Z Z, i.e. by typing twice the same uppercase letter.

F7. This is the keyboard prefix for calligraphic symbols such as $\mathcal{A}$ (F7 A) and $\mathcal{P}$ (F7 P). Notice that not all fonts provide calligraphic variants $\mathcal{a}$, $\mathcal{b}$, $\mathcal{c}$, ... for lowercase letters. Whenever such variants are missing for a given font, then TeXmacs will use a system-dependent substitute font instead (see section 3.8.1).

d	d ⇥ ⇥	∂	d ⇥ ⇥ ⇥	e	e ⇥ ⇥	i	i ⇥ ⇥
γ	g ⇥ ⇥	λ	l ⇥ ⇥	π	p ⇥ ⇥	Γ	G ⇥ ⇥ ⇥
$\hbar$	h ⇥ ⇥	$\imath$	i ⇥ ⇥ ⇥	$\jmath$	j ⇥ ⇥	ℓ	l ⇥ ⇥ ⇥
D	D ⇥ ⇥	∇	V ⇥ ⇥	Δ	D ⇥ ⇥ ⇥ ⇥	∞	@ @
$\Re$	R E	$\Im$	I M	∁	C ⇥ ⇥ ⇥	Ⅎ	G ⇥ ⇥ ⇥ ⇥
$\forall$	A ⇥ ⇥	$\exists$	E ⇥ ⇥	$\nexists$	E ⇥ ⇥ ⇥	$\wp$	P ⇥ ⇥
$\aleph$	A ⇥ ⇥ ⇥	$\beth$	B ⇥ ⇥ ⇥	$\gimel$	G ⇥ ⇥	$\daleth$	D ⇥ ⇥ ⇥
$\backepsilon$	e ⇥ ⇥ ⇥ ⇥	ɷ	W ⇥ ⇥	Ǝ	E ⇥ ⇥ ⇥ ⇥	$\mho$	W ⇥ ⇥

Table 4.2. Keyboard shortcuts for several letter-like symbols.

- **F8**. This keyboard prefix corresponds to the "Fraktur" font. For example, the shortcuts F8 m and F8 I produce $\mathfrak{m}$ and $\Im$.

In fact, we note that $\mathrm{T_EX_{MACS}}$ does not consider mathematical letter-like symbols like $\boldsymbol{v}$, $\mathbb{N}$, $\mathcal{A}$, and $\mathfrak{m}$ as being typeset in a separate font. Instead, such symbols are interpreted as special characters in an extended (UNICODE) alphabet. In particular, the font selector (see section 3.8.2) does not contain any entry for—say—the "blackboard bold" font.

There exist a few more letter-like symbols that cannot be obtained through the systematic mechanisms from above. First of all, the important mathematical constants i, e, and π are entered using i ⇥ ⇥, e ⇥ ⇥, and p ⇥ ⇥. Notice that $\mathrm{T_EX_{MACS}}$ uses an upright rendering for these constants, which allows you to distinguish them from the letters i, e, and π. The d and λ from differential- and lambda-calculus are entered similarly, using d ⇥ ⇥ and l ⇥ ⇥. Table 4.2 shows the list of keyboard shortcuts for these and other letter-like mathematical symbols.

4.2.2 Other symbols

There are a few simple rules that allow you to enter most non-letter-like mathematical symbols using "natural" key-combinations. The most important rule is juxtaposition: - > yields $\rightarrow$, - - > yields $\longrightarrow$, and > = yields $\geqslant$. Similarly, | ⇥ - yields $\vdash$, | - > yields $\mapsto$, and - > < - yields $\rightleftarrows$. The other rules are all based on the use of a special key:

- **⇥**. This is the main key for obtaining *variants* (see section 2.3.2). Some symbols have many variants. For example, < yields $<$, < ⇥ yields $\in$, < ⇥ ⇥ yields $\subset$, < ⇥ ⇥ ⇥ yields $\prec$, < ⇥ ⇥ ⇥ ⇥ yields $\sqsubset$, < ⇥ ⇥ ⇥ ⇥ ⇥ yields $\langle\rangle$, and < ⇥ ⇥ ⇥ ⇥ ⇥ ⇥ returns to $<$. You may "cycle back" among the variants using ⇧⇥, so that < ⇥ ⇥ ⇧⇥ is equivalent to < ⇥. For symbols with many variants, the last variants are obtained most efficiently by cycling back. For example, quick shortcuts for $\langle\rangle$ and $\sqsubset$ are < ⇧⇥ and < ⇧⇥ ⇧⇥.

  The variant mechanism is particularly powerful when used in conjunction with juxtaposition. For example, < =, < = ⇥, < = ⇥ ⇥, and < = ⇥ ⇥ ⇥ respectively yield $\leqslant$, $\leq$, $\leqq$, and $\Leftarrow$. Notice that the juxtapositions are horizontal for the first three shortcuts, but vertical for the last one. Similarly, the shortcuts < = =, < ⇥ ⇥ =, and < < < = produce $\Longleftarrow$, $\subseteq$, and $\lll$.

Chapter 9

Laptop presentations

9.1 Introduction

TeXmacs provides a special `beamer` style for the creation of laptop presentations. A beamer presentation consist of a series of screens instead of pages. During a presentation, the screens are displayed one after another, via a remote controller or special keyboard shortcuts. Using special markup, the screens can be further subdivided, so as to make part of their content appear progressively or in alternation. The presentation may also contain dynamic elements such as animations or on-the-fly mathematical computations.

There exist two main types of screens: textual and graphical ones. *Textual* screens are simply pages of the size of your screen and similar formatting rules apply as on ordinary paper. In particular, a consistent layout is guaranteed throughout the presentation. *Graphical* screens are pictures of the size of your screen with an optional title. Text can be placed anywhere inside such pictures and freely moved around. It is also easy to insert other graphical objects such as polygons, circles, and more general curves.

During an actual presentation, it is important to use the entire screen and remove all menu bars, toolbars, and other clutter. This is the purpose of the *presentation mode*, which can be activated and deactivated using View ▸ Presentation mode or `^F9`. The main keyboard shortcuts for traversing your slides forward and backward are `F11` and `F10`. The keys `F9` and `F12` are used to go to the start and end of the presentation, whereas `⇟` and `⇞` allow you to directly move forward and backward by one screen at a time. TeXmacs also provides a *panorama mode* which can be toggled using View ▸ Show panorama or `^F10`. This mode provides a global overview of the entire presentation and allows you to quickly move the cursor to any position.

During live performances, a remote controller is often preferred for navigating through the presentation. TeXmacs supports several types of remote controllers. Some of them (such as Apple infrared controllers) should work out of the box. Others map the buttons on the remote controller to certain keys on your keyboard, and you will need to toggle View ▸ Remote control in order to remap these keys to the right actions during presentations. If necessary, the appropriate mappings can be specified in Edit ▸ Preferences ▸ Keyboard ▸ Remote control.

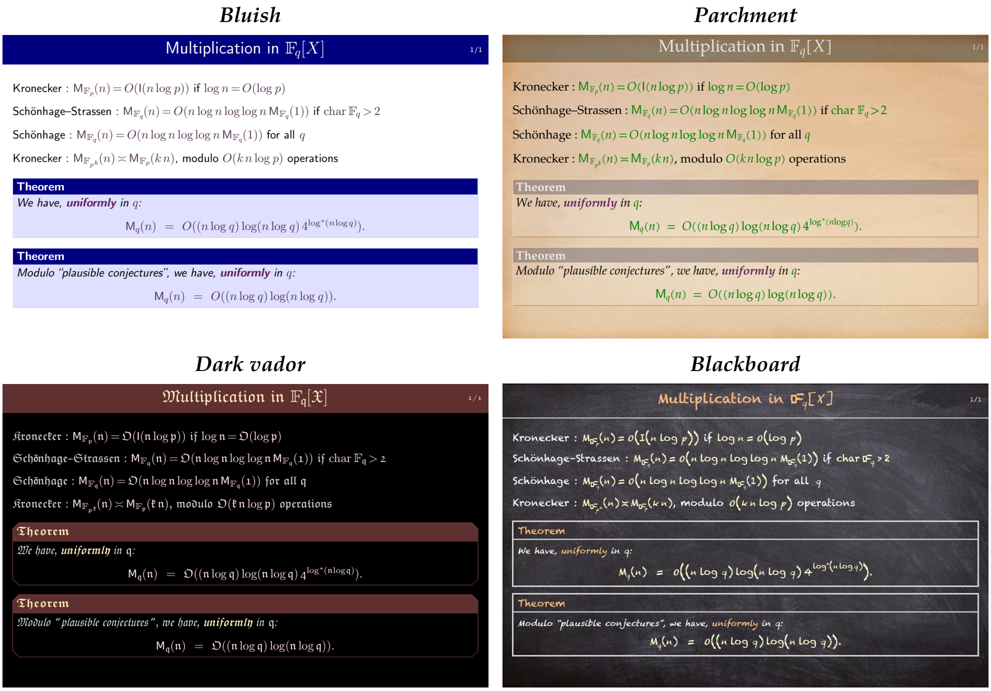

Figure 9.1. Various beamer themes.

By activating the debugging tool Tools ▸ Debugging tool and Debug ▸ keyboard, you may determine the particular mappings used by your remote control.

9.2 Themes and style options

You may select one of the beamer *themes* in order to specify the global look and feel of your presentation. Such a theme mainly corresponds to a collection of colors and background patterns for titles, mathematical formulas, theorems, etc., and the presentation itself. Assuming that the focus is on your entire presentation, the available themes can be found in Focus ▸ Beamer theme; see Figure 9.1 for a few examples.

In addition to the main theme, we recall from section 2.7 that the rendering of individual markup elements can be further customized via the Focus ▸ Preferences menu. For instance, inside a screen title, you may select Focus ▸ Preferences ▸ Framed titles as an alternative for the default title bar rendering at the top of the screen. Similarly, inside a theorem, you may select the alternative Hanging theorems rendering. Yet another example is the use of an alternative color for mathematical formulas. See Figure 9.2 for the effect of such customizations.

Default non-framed theorems *Hanging theorems*

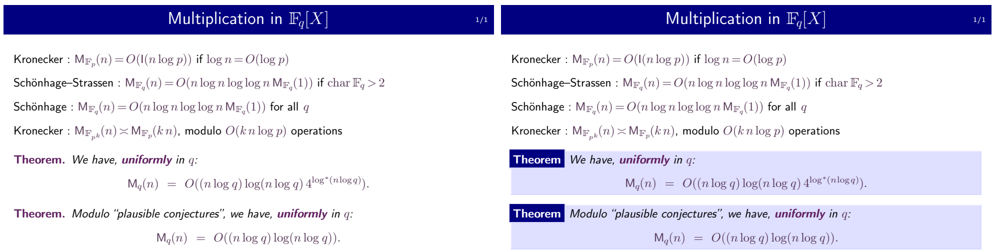

Figure 9.2. Customized rendering for various markup elements.

The top-level focus menu also allows you to specify an aspect ratio for your presentation. Traditional projectors use a ratio of 4:3 or 5:4. When connecting your laptop to a widescreen television or certain recent projectors, you may need to select a ratio of 16:9 or 8:5 instead. In case of doubt, we recommend to opt for a 4:3 or 8:5 ratio.

When preparing a presentation, it is sometimes nice to see all screens in succession, rather than editing them one by one. This can be achieved by selecting the Paper page mode from the ☐ icon menu on the focus toolbar or using Document ▸ Page ▸ Format ▸ Page rendering ▸ paper. Alternatively, the Panorama mode allows you to see miniature versions of all your slides on a single screen. The paper and panorama page modes are also useful whenever you need to reorganize your presentation. Indeed, they allow you to select ranges of screens and to copy and paste them.

9.3 Switching, folding, and unrolling

The root of a presentation consists of a screens tag. This tag is an example of a *switch*: it contains an arbitrary number of screens as its arguments, but only one screen is displayed at each time. The Insert ▸ Fold ▸ Switch menu provides several other types of switches, each of which allows you to show different pieces of text in a successive and mutually exclusive manner. The most common Standard switch is used for paragraph-wide content (similar to "displayed" formulas), whereas you should use Tiny switches for inline text (similar to "inline" formulas).

Inside a switch, new *branches* can be inserted after or before the currently visible branch using Focus ▸ Insert argument after or Insert argument before; you may also use ⌥→, ⌥←, or one of the icons ⊕ or ⊕ on the focus toolbar. We also note that switches can be nested inside each other. For example, it is perfectly allowed to insert, say, a standard switch inside a screens switch. When traversing a presentation forwards using F11, you will only move to the next screen of the screens switch after traversing all branches of the innermost switch.

Another popular way to traverse a presentation is to progressively unroll content. This can be done by inserting one of the *unrolling* tags from Insert ▸ Fold ▸ Unroll:

Compressed. When using this default unrolling style, the branches that have not been unrolled are simply ignored. As a consequence, if the unroll tag is followed by other text, then this subsequent text will move downwards while unrolling.

Phantoms. When using this style, the branches that have not yet been unrolled are "displayed" using invisible ink. If the unroll tag has not completely been unrolled, then there will be blank lines between the unroll tag and subsequent text (if any).

Greyed. For this style, the branches that have not yet been unrolled are "pre-displayed" using lightly visible ink.

These primitive unrolling tags can be combined with the various kinds of item lists described in section 3.6. For instance, Insert ▸ Fold ▸ Unroll ▸ Itemize inserts a standard item list that can be unrolled progressively according to the default, compressed unrolling style. Using the standard mechanism of structured variants (see section 2.7), one may next replace the itemize and unroll tags by any of their variants.

Inside a list that can be unrolled, pressing ↩ creates a new unrollable branch with a list item. Sometimes you may want to unroll more than one item at once; in that case, you have to manually insert additional items to the branch, e.g. by typing \ i t e m ↩.

A variant of unrolling is *unfolding*, in which case there are exactly two branches, one of which is always visible, and one of which can be "folded". Different folding styles are available through Insert ▸ Fold ▸ Folded. Some of the rendering styles display a button that may be pushed in order to fold or unfold. The input-output fields inside computer algebra sessions are also foldable.

Yet another variant of folded structures is available through Insert ▸ Fold ▸ Summarize. The tags in this menu are switches with two branches, again with different rendering styles.

When using TeXmacs in combination with an external plug-in, such as a computer algebra system, you will notice that all input-output fields in sessions are foldable. In addition, you can create so-called "executable switches" using the items in the Insert ▸ Fold ▸ Executable submenu. This allows you to switch back and forth between a given input and the corresponding output.

We already explained that switches may be nested in a natural way. The same holds for the other tags that we have just described. Moreover, in the Insert ▸ Fold ▸ Traversal menu, you may specify whether unrolled and unfolded structures should be folded back after traversal.