page-canons.sty

classical-inspired page layouts with margin functionality

Daniel Quigley

May 21, 2026

Contents

Part I

Concepts

Chapter 1

Overview

page-canons implements classical page-layout canons with in-house margin notes and sidenotes. Its one declared external dependency is geometry; all other functionality is self-contained. The package recommends LuaLaTeX.

The implemented canons are:

Van de Graaf medieval proportions; text block matches page aspect;

Villard de Honnecourt parametric family ($N = 3, 6, 9, 12, 15$) with flexible margins;

Tufte asymmetric with wide outer margin for extensive notes;

Canon des Ateliers three styles (ordinary, neater, luxury) from French typographic tradition;

Grid modern $N \times N$ modular system with full control.

page-canons offers four margin modes (`symmetric`, `antisymmetric`, `right`, `left`), gutter support with two calculation modes, a `fullwidth` environment, and exported dimensions for margin-aware content.

Use page-canons when you want canonical proportions and consistent marginalia defaults without adopting a full document class. It sits above geometry: page-canons handles the math; geometry, the implementation. For end-to-end book production, use memoir or KOMA-Script; for a curated editorial idiom, use tufte-book.

Part I fixes concepts; Part II is the user manual; Part III consolidates tests, edge cases, and a quick reference. All compiled pages show geometry frames (`showframe`) such that the reader can see the effect of each option directly.

Chapter 2

Page canons

2.1 What a canon is

Page-construction canons are modern reconstructions of historical text-setting, derived from measuring surviving books and inferring the geometry used to divide a page into proportionate text areas and margins. Popularized in the 20th century by Jan Tschichold after work by J. A. van de Graaf, Raúl Rosarivo, Hans Kayser, and others, these rules still influence contemporary book design, adapted to standardized papers and diverse production needs.

Canon comes via Latin *canon* 'rule; standard' from Greek *kanōn* 'straight rod; measuring rule'. The same sense underlies *canon law* and the musical *canon*.

In practical terms, a canon takes a page's dimensions and non-arbitrarily calculates margin sizes, positions the text block on the page, maintains proportions across paper sizes, and optionally allocates space for marginalia, headers, and footers.

2.2 Formal definition

Let $W = \texttt{\textbackslash paperwidth}$, $H = \texttt{\textbackslash paperheight}$, and g be the gutter width.

Definition 2.1.

A **page canon** is a *scale-equivariant* layout rule

$$\mathcal{C} : (F, \Xi) \longrightarrow \Theta,$$

where F is a rectangular frame, Ξ is an optional set of parameters and constraints, and Θ is a layout solution.

The frame is chosen from:

- the **page frame** $P = [0, W] \times [0, H]$, whose four edges are designated *inner*, *outer*, *top*, *bottom* according to binding orientation;
- the **leaf frame** with gutter g: for recto leaf $L_r = [g, W] \times [0, H]$; for verso leaf $L_v = [0, W - g] \times [0, H]$.

Ξ may include grid divisions N, a baseline step, desired ratios, or style flags. Θ at minimum specifies a textblock rectangle $T \subseteq F$, equivalently given by margins (m_i, m_o, m_t, m_b); it may also include footer and header bands and marginalia measures (`marginparwidth`, `marginparsep`).

Scale-equivariance requires that, for any $\lambda > 0$, $\mathcal{C}(\lambda F, \Xi) = \lambda \mathcal{C}(F, \Xi)$: scaling the frame scales the output proportionally. A canon must declare whether it operates on P or on L.

2.3 Two presentations

The same canon admits two presentations.

Definition 2.2.

The **parametric (algebraic) canon** gives functions of the operative frame dimensions.

Define the operative dimensions W_*, H_*:

$$W_* = \begin{cases} W & (\text{page frame } \texttt{guttermode=geometry}) \\ W - g & (\text{leaf frame } \texttt{guttermode=satzspiegel}) \end{cases}, \qquad H_* = H.$$

Margins are then

$$m_j = f_j(W_*, H_*, \Xi) \cdot \begin{cases} W_* & \text{horizontal,} \\ H_* & \text{vertical,} \end{cases}$$

where each f_j is a dimensionless ratio.

Definition 2.3.

The **constructive (geometric) canon** specifies a procedure on F (subdivide, draw diagonals, intersect lines) that yields a unique textblock T.

The resulting fractions are derived; they may depend on the page aspect H/W and need not be 'nice' rationals. The parametric form is what page-canons implements; the constructive form shows where the proportions come from and helps justify why certain relationships exist: formulas build, constructions justify.

2.4 Requirements of a page canon

A canon, then, is a rule we can test; it should behave predictably under scaling, declare the space it acts on, say where it works and what happens when it does not, produce an unambiguous textblock, and change smoothly when inputs do:

1. all outputs scale by $\lambda > 0$; units do not matter;
2. the canon declares its operative frame (page or leaf) and the recto/verso mapping of inner/outer;
3. aspect ratios and parameters for which the construction is valid are stated, with a fallback policy otherwise;
4. fixed inputs give a unique T, with no eyeball steps;

5. small input changes do not cause discontinuous jumps (piecewise definitions are allowed, but should be noted).

page-canons implements canons as parametric rules, with an optional `guttermode=satzspiegel` that switches the operative frame from page P to leaf L by recomputing on $W_* = W - g$. The Honnecourt family is encoded algebraically (a constructive origin; an algebraic implementation).

2.5 Related work

KOMA-Script [2] typearea chooses a typeblock via `DIV`, `BCOR`, and class-level heuristics sensitive to font details; page-canons fixes canonical ratios and shims the result through geometry. Use KOMA for class-integrated page design; use page-canons for literal canons.

memoir [4] offers a comprehensive layout calculus (`\settypeblocksize`, `\setlrmargins`, diagnostics) and is a full publishing toolkit. For end-to-end book production, memoir is the right hammer; page-canons is a small wrench: canons, gutters, marginalia.

tufte-book [1] embodies a coherent editorial idiom beyond raw geometry: wide outer margins, sidenotes, specialized floats. page-canons can mimic the broad proportions but does not implement the idiom.

Note.

Given W, H and options, margins are fixed by explicit fractions without heuristics. The four margin modes place margin material predictably across book, report, and article. Two gutter modes (invariant textwidth; recomputed on leaf width) are available. Exported lengths (`\marginandtext`, `\fullwidthoverhang`, `\overflowingheadlen`) make margin-aware figures, rules, and floats routine. The package inherits geometry's flexibility for anything it does not explicitly cover.

If you already know your exact margins, geometry alone is simpler; if you want canonical proportions and consistent marginalia defaults, page-canons saves you from re-deriving them.

2.6 Limitations

page-canons deliberately restricts itself to page geometry:

- layout is resolved once at `\AtBeginDocument`; `\pagecanonsetup` can adjust mid-document, but recomputation is global, not per-page or per-chapter;
- the package does not manage line spacing or enforce a baseline grid;
- it is not compatible with classes that manage layout internally (memoir, tufte-book, KOMA-Script); warnings are issued if such classes are detected;
- for Honnecourt and Ateliers, vertical margins are expressed as fractions of W (not H), coupling textblock aspect to paper aspect H/W; a defensible but contested reading;
- in `guttermode=geometry`, large binding allowances can collapse the outer margin to zero (warning issued); use `guttermode=satzspiegel` to preserve proportions;
- the package only computes and applies margins; it does not style headings, floats, front matter, running heads, or captions.

Part II

Using page-canons

Chapter 3

Loading and compatibility

3.1 Syntax

```
\usepackage[<options>]{page-canons}
```

Options fall into three categories: canon selection; margin and marginalia configuration; canon-specific parameters. Unrecognized options are forwarded to geometry.

3.2 Dependencies

page-canons targets LuaLaTeX as the recommended engine for PDF/UA-2 accessibility support. The package loads under any modern engine (pdfLaTeX, XeLaTeX, LuaLaTeX), but emits a warning on non-LuaLaTeX engines; accessibility features may be limited.

Odd/even page detection is implemented in-house. The package checks for ifoddpage at load time and uses it if already present, but does not require it.

Table 3.1: Package dependencies

Package	Role
geometry	page layout application

3.3 Document classes

book and report default to `margins=symmetric`. article defaults to `margins=right`. Warnings are issued for memoir and KOMA-Script classes, which manage layout internally.

3.4 This document's layout

This primer is compiled with `canon=vdg`, `margins=symmetric`, `paper=letterpaper`, `showframe`. Exported dimensions at this point in the document:

`\textwidth` = 409.52376pt

`\marginparwidth` = 109.20946pt

`\marginparsep` = 13.64766pt

`\marginandtext` = 532.38087pt

`\fullwidthoverhang` = 122.85712pt

Margin mode: `symmetric`.

Chapter 4

The canons

Each canon is a rule (constructive or parametric) that maps a page or leaf rectangle and optional parameters to a textblock and margins. The default operative frame is the page (`guttermode=geometry`); if `guttermode=satzspiegel` is selected, we recompute on the leaf width $W_* = W - g$ before mapping back.

Canon formulas use W and H for page dimensions. In leaf-frame mode, substitute $W_* = W - g$ for all W in horizontal calculations; vertical calculations use H unchanged. In the default page-frame mode, $W_* = W$ and formulas apply directly.

4.1 Van de Graaf (`vdg`)

The Van de Graaf canon, named after Dutch book designer J. A. van de Graaf, reconstructs medieval manuscript proportions that appear consistently in incunabula and hand-copied texts from the 12th–16th centuries.

The construction begins with the insight that the text block should maintain the same proportions as the page itself:

1. draw both diagonals of the full page spread (verso and recto together);
2. draw the diagonal of a single page;
3. where the single-page diagonal intersects the spread diagonal determines the text block corner;

4. this intersection occurs at exactly 1/9 of the page width and height.

The 1/9 fractions are the outcome of the diagonal construction under the usual single-page-within-spread setup; other historical reconstructions exist. The construction is primary; the fractions, derived. Our implementation codifies the fractions.

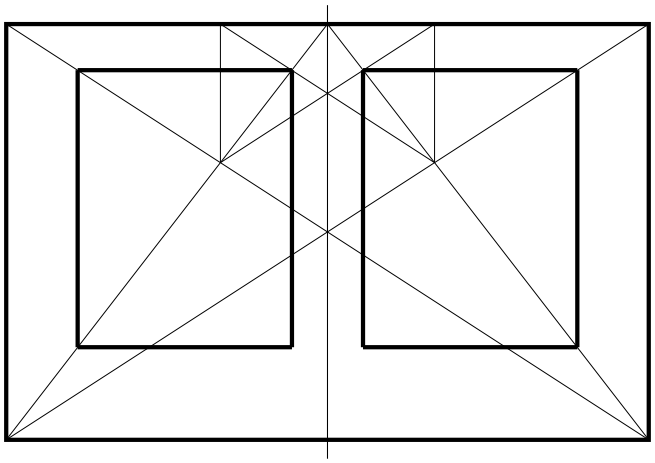


Figure 4.1: Van de Graaf canon; textarea shown with spread.

For a page of width W and height H:

$$m_i = W/9, \quad m_o = 2W/9, \quad m_t = H/9, \quad m_b = 2H/9,$$

yielding `\textwidth` $= \frac{2}{3}W$ and `\textheight` $= \frac{2}{3}H$. Marginalia allocation: `marginparwidth` $\approx 8W/45$, `marginparsep` $\approx W/45$. Footer: `footskip` $= \frac{1}{2}m_b$.

4.2 Villard de Honnecourt (`vdh`)

Villard de Honnecourt was a 13th-century French architect whose sketchbook contains about 250 highly precise drawings, 74 of which are architectural; the remainder span naturalism, allegory, scenes, animals, and machines. His *canon de division harmonieuse* uses recursive subdivision of the page, creating a parametric family of layouts based on N-fold divisions.

Define $u = W/(N+3)$. Then

$$m_i = u, \quad m_o = 2u, \quad m_t = \tfrac{3}{2}u, \quad m_b = 3u,$$

so \textwidth $= \frac{N}{N+3}W$ and \textheight $= H - \frac{9}{2}\frac{W}{N+3}$.

Because vertical margins depend on W, the block aspect varies with H/W.

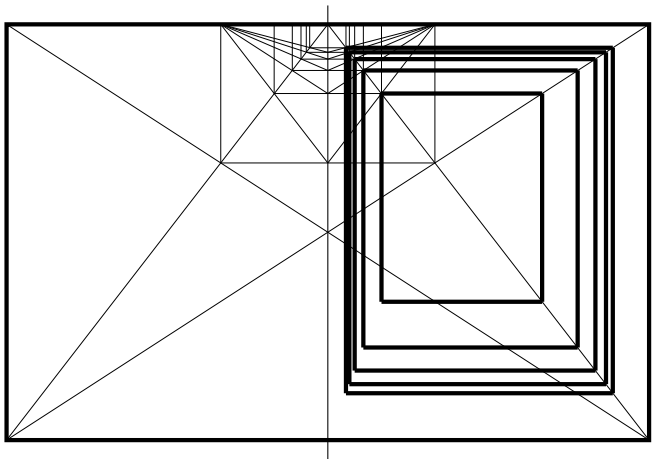


Figure 4.2: Honnecourt canon, textareas drawn for each N; shown with spread.

Configurations are given in Table 4.1. Exact recovery on an integer grid occurs only for special paper aspects.

Table 4.1: Honnecourt configurations.

N	**Inner**	**Outer**	**Top**	**Bottom**	**Margin space**
3	$W/6$	$W/3$	$W/4$	$W/2$	generous
6	$W/9$	$2W/9$	$W/6$	$W/3$	classic
9	$W/12$	$W/6$	$W/8$	$W/4$	moderate
12	$W/15$	$2W/15$	$W/10$	$W/5$	economic
15	$W/18$	$W/9$	$W/12$	$W/6$	compact

Note.

With $N = 3$, division into six reproduces the method Tschichold identifies as Marcus Vencentinus's 15th-century prayer-book scheme. The Honnecourt family subsumes it via parameter choice.

4.3 Tufte (tufte)

Edward Tufte's page design [9, 7, 10, 6, 8] prioritizes generous margins for annotations, figures, and marginalia: wide margins accommodate figures without interrupting text flow; multiple information streams coexist; deliberate asymmetry creates dynamic tension; margins invite reader participation.

This is an algebraic house-style, not a medieval geometric canon; the fractions below are abstracted from Tufte-style layouts.

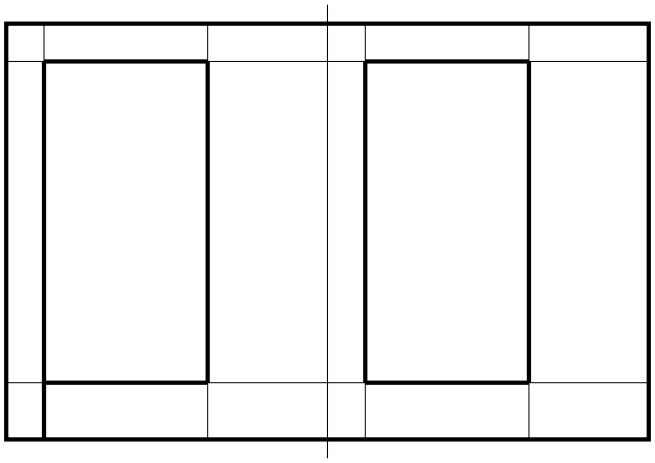


Figure 4.3: Tufte canon; textareas shown with spread.

The ratios are derived directly from the tufte-latex document classes on letterpaper:

- inner margin: $W/8.5 \approx 0.1176\,W$;
- outer margin: $\approx 0.372941\,W$ (empirical from tufte-latex on letterpaper; not exactly $3/8$);
- top: $H/11 \approx 0.0909\,H$;
- bottom: $3H/22 \approx 0.1364\,H$;
- textwidth: $\approx 0.509\,W$;
- marginparwidth: $4W/17 \approx 0.235\,W$;
- marginparsep: $W/26 \approx 0.038\,W$.

The layout exhibits strong asymmetry with extensive annotation space.

4.4 Canon des Ateliers (`ateliers`)

The *canon des ateliers* ('workshop canon') follows from French printing tradition. It supplies three distinct styles:

1. **ordinary** (*ordinaire*) maximizes text area while maintaining readability; educational texts and technical manuals;
2. **neater** (*plus soigné*) balanced aesthetic for literary works; increased margins for readability;
3. **luxury** (*luxe*) generous margins for prestigious editions; maximum annotation space.

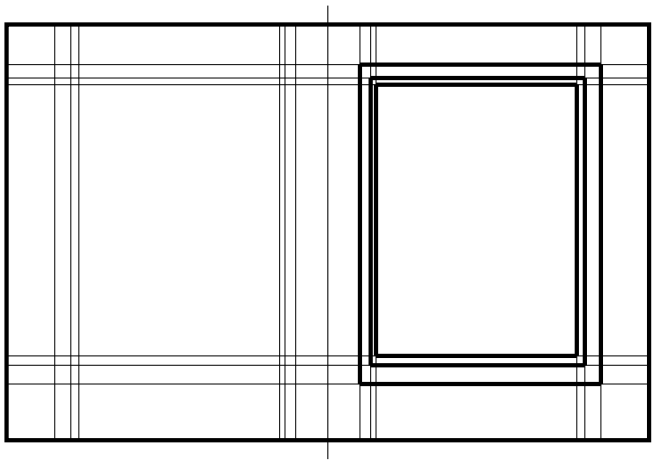


Figure 4.4: *Canon des ateliers*; textareas drawn for each style; shown with spread.

Textwidth is set first:

$$\texttt{textwidth} = \begin{cases} \frac{3}{4}W & \text{(ordinary)}, \\ \frac{2}{3}W & \text{(neater)}, \\ \frac{5}{8}W & \text{(luxury)}. \end{cases}$$

The remaining whitespace $w = W - \texttt{textwidth}$ is then distributed: $m_i = 0.4w$, $m_o = 0.6w$, $m_t = 0.5w$, $m_b = 0.7w$.

The three-tiered system reflects material and economic realities of printing; subject matter, print runs, and material costs determined the style. Luxury editions could afford more white space; higher sale price offset the cost of paper. Better bindings (particularly sewn bindings for luxury editions) accommodated wider inner margins without compromising visibility at the spine; cheaper bindings required narrower inner margins.

4.5 Grid (grid)

The Grid canon is a modern, systematic approach based on modular design popularized by the Swiss International Style. Unlike historical canons, it offers complete parametric control while maintaining modular discipline.

The page is divided into an $N \times N$ grid. For some $N \geq 3$, let $c_w = W_*/N$, $c_h = H/N$. Choose integer cell counts (L, R, T, B) with $L + R < N$ and $T + B < N$. Set

$$m_i = Lc_w, \quad m_o = Rc_w, \quad m_t = Tc_h, \quad m_b = Bc_h.$$

The text block is

$$\texttt{\textwidth} = (N - L - R)c_w, \qquad \texttt{\textheight} = (N - T - B)c_h.$$

The text-area fraction (relative to the operative frame $W_* \times H$) is

$$\frac{\texttt{\textwidth}}{W_*} \cdot \frac{\texttt{\textheight}}{H} = \left(1 - \frac{L+R}{N}\right)\left(1 - \frac{T+B}{N}\right),$$

and the text-block aspect is

$$\frac{\texttt{\textheight}}{\texttt{\textwidth}} = \frac{N - T - B}{N - L - R} \cdot \frac{H}{W_*}.$$

Grid verticals are multiples of H/N, *not* fractions of W; therefore, the block aspect depends on both H/W and the chosen cell counts. Parameters and defaults are given in Table 4.2.

Useful configurations appear in Table 4.3.

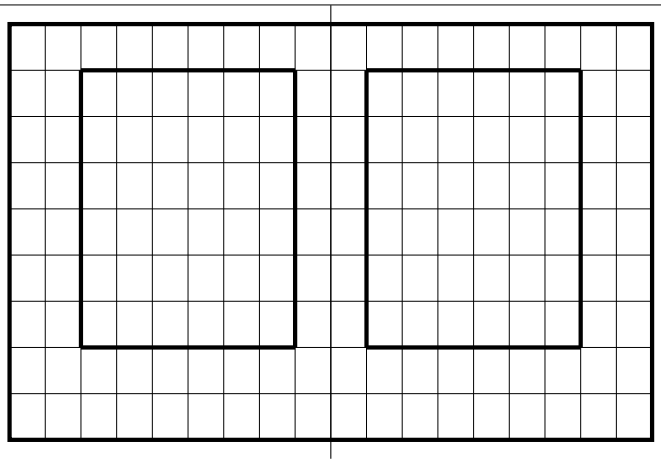


Figure 4.5: Grid canon for $N = 9$; textarea shown with spread.

Table 4.2: Grid canon parameters (*clamped to 1 when $N = 3$).

Parameter	**Default**	**Range**	**Description**
`gridN`	6	≥ 3	grid divisions ($N \times N$)
`gridinner`	1	$< N$	inner margin cells
`gridouter`	2*	$< N$	outer margin cells
`gridtop`	1	$< N$	top margin cells
`gridbottom`	2*	$< N$	bottom margin cells

4.6 Disabled (`canon=false`)

Bypasses all canon calculations. geometry controls layout via its own defaults or any forwarded options. The `margins` key still applies (`reversemarginpar` or `asymmetric` where relevant).

Table 4.3: Common grid configurations.

Style	N	**Inner**	**Outer**	**Top**	**Bottom**	**Description**
minimal	3	1	1	1	1	maximum text area
classic	6	1	2	1	2	balanced proportions
editorial	8	1	3	1	2	wide margin notes
technical	12	1	2	2	3	documentation layout
dense	12	1	1	1	2	maximized content

Chapter 5

Canon relationships

Historical canons encode specific aesthetic judgments about proportion. The Grid canon provides a framework for systematic decision-making and can reproduce some historical relationships exactly, approximate others, and create new ones.

5.1 Grid as a superset

Table 5.1: Classical canons and Grid compared.

Canon	**Computation**	**Parameters**	**Flexibility**
Van de Graaf	fixed fractions	0	none
Villard	formula with N	1 (N)	discrete (5)
Ateliers	whitespace split	1 (style)	discrete (3)
Grid	cell counting	5 ($N + 4$)	continuous

Note.

Van de Graaf is effectively a 9×9 grid with fixed allocations: $m_i/W_* = 1/9$, $m_o/W_* = 2/9$, $m_t/H = 1/9$, $m_b/H = 2/9$. The Grid canon recovers Van de Graaf exactly when N is a multiple of 9 and $(L, R, T, B) = (N/9, 2N/9, N/9, 2N/9)$. If N is not divisible by 9, the result is a best rational approximation in steps of $1/N$.

Table 5.2: Van de Graaf canon and Grid compared.

Aspect	Van de Graaf	Grid equivalent	Grid notation
division	9×9 (implicit)	9×9 (explicit)	`gridN=9`
inner margin	$W_*/9$ (fixed)	1 cell	`gridinner=1`
outer margin	$2W_*/9$ (fixed)	2 cells	`gridouter=2`
top margin	$H/9$ (fixed)	1 cell	`gridtop=1`
bottom margin	$2H/9$ (fixed)	2 cells	`gridbottom=2`
text block	6×6 cells	6×6 cells	computed
user control	none	full	all parameters

To reproduce Van de Graaf with Grid:

```
\usepackage[canon=grid, gridN=9, gridinner=1, gridouter=2, gridtop=1, gridbottom=2]{page-canons}
```

Grid admits variations Van de Graaf does not, e.g.:

```
% same 9x9 grid, but 1:3 horizontal ratio instead of 1:2
\usepackage[canon=grid, gridN=9, gridinner=1, gridouter=3]{page-canons}
```

Note.

Our Honnecourt family sets verticals as fractions of W: $m_t = \frac{3}{2}u$, $m_b = 3u$ with $u = W/(N+3)$. Grid fixes verticals in steps of H/N. Exact equality would require

$$T = \frac{3N}{2(N+3)} \cdot \frac{W}{H}, \qquad B = \frac{3N}{N+3} \cdot \frac{W}{H},$$

which are rarely integers unless the paper aspect is specially tuned. Grid $\rightarrow$ Honnecourt is therefore generally an approximation, good for large N.

Note.

Ateliers distributes a fixed whitespace fraction by ratios $0.4 : 0.6 : 0.5 : 0.7$. Integer cell counts cannot reproduce these exactly; choose large N and round.

Grid can reproduce exactly:

- Van de Graaf with $N = 9$, margins $1 : 2 : 1 : 2$;

Table 5.3: Non-integer values (*) mark where Grid cannot exactly reproduce Ateliers.

Style	**Text width**	**Whitespace**	**Distribution**	**Grid approximation**
ordinary	75%	25%	$0.4 : 0.6 : 0.5 : 0.7$	$N = 20$, cells $2 : 3 : 2.5^* : 3.5^*$
neater	66.7%	33.3%	$0.4 : 0.6 : 0.5 : 0.7$	$N = 15$, cells $2 : 3 : 2.5^* : 3.5^*$
luxury	62.5%	37.5%	$0.4 : 0.6 : 0.5 : 0.7$	$N = 16$, cells $2.4^* : 3.6^* : 3 : 4.2^*$

- simple ratios like $1 : 2$, $1 : 3$, $2 : 3$.

It can approximate:

- Honnecourt's $1.5 : 3$ vertical ratio (needs $N = 6k$ for a $k : 2k$ approximation);
- Ateliers' $0.4 : 0.6$ split (needs large N).

It can express novel configurations: asymmetric layouts (e.g., $1 : 4 : 2 : 1$), golden-ratio approximations with appropriate N, Fibonacci margins.

Chapter 6

Margin modes

The `margins` key controls how inner/outer asymmetry is applied and which side receives marginalia. All four modes compose with any canon.

6.1 `symmetric` (default for book/report)

Standard two-sided layout. Marginalia appear in the outer margin. Fullwidth content extends into the outer margin and shifts direction on verso pages.

6.2 `antisymmetric`

Two-sided layout with marginalia on the *inner* (spine-side) margin. Unusual; useful where annotations face the gutter in spread view.

6.3 `right`

Single-sided. Every page has the wider margin on the right. Default for article.

6.4 `left`

Single-sided. Every page has the wider margin on the left.

6.5 Geometry flags

Each mode maps to geometry as follows:

Table 6.1: Margin-mode placement

Mode	Geometry flags	Effect
`symmetric`	(default)	two-sided; notes on outer edge
`antisymmetric`	`reversemarginpar`	two-sided; notes on inner edge
`right`	`asymmetric`	one-sided; notes on right
`left`	`asymmetric`	one-sided; notes on left

Chapter 7

Gutters

The `gutterval` key adds binding allowance. Two application modes are available via `guttermode`.

7.1 `geometry` (default)

Adds the gutter to the inner margin and subtracts it from the outer margin, preserving page width but shifting the text block toward the outer edge. Since `\textwidth` $= W - (m_i + m_o)$ and the change is $+g$ inner, $-g$ outer, the sum is unchanged.

Caveat: large binding allowances can collapse the outer margin to zero (warning issued).

7.2 `satzspiegel`

Reduces the effective leaf width to $W_* = W - g$ before computing the canon, then adds the gutter back to the inner margin. This shrinks the text block rather than stealing from the outer margin: Δ`\textwidth` $= -\alpha\, g$ with $\alpha = \frac{2}{3}$ (vdg), $\frac{N}{N+3}$ (vdh), ≈ 0.509412 (tufte), k (ateliers), or $\frac{N-L-R}{N}$ (grid).

Chapter 8

Marginalia

page-canons provides margin notes and sidenotes with in-house collision avoidance, font isolation, and position-aware placement. Implementation writes position labels to the `.aux` file on the first pass and reads them on the second; **two compilation passes are required** for accurate vertical positioning. On the first pass, notes appear at the call-site baseline without collision adjustment.

Margin notes always land in the outer margin, following the current margin mode. On two-sided symmetric documents, the outer margin alternates sides.

8.1 Margin notes

The functionality of margin notes follows from [3].

8.1.1 Syntax

```
\marginnote{text}            % auto-positioned, collision-aware
\marginnote{text}[2cm]       % shifted 2cm below call-site baseline
\marginnote{text}[-1cm]      % shifted 1cm above call-site baseline
```

The optional `voffset` is *additive* to the collision accumulator. A note that would already be pushed down by collision avoidance is pushed further by a positive offset.

8.1.2 Font isolation

Margin notes always typeset in upright, medium-weight roman regardless of the call-site font; this is enforced by `\normalfont` prepended before `\marginfont` in the content builder. A margin note inside italic or bold body text does not inherit the variant.

8.1.3 Collision avoidance

Adjacent notes on the same page accumulate vertically. If the notes exceed the available margin depth they overflow the page foot; no silent truncation occurs.

8.2 Sidenotes

Sidenotes are numbered annotations: a mark appears inline in the body, and the note text appears in the outer margin at the corresponding vertical position. The functionality of side notes follows from [5].

8.2.1 Syntax

```
\sidenote{text}               % auto-number, auto-position
\sidenote{text}[2cm]          % auto-number, shifted 2cm down
\sidenote[3]{text}            % explicit number 3, auto-position
\sidenote[3]{text}[-1cm]      % explicit number, shifted up

\sidenotemark                 % place mark only (steps counter)
\sidenotemark[3]              % place mark with explicit number
\sidenotetext{text}           % place note text (does not step counter)
\sidenotetext{text}[1cm]      % place note text, shifted down
```

The optional leading `[num]` overrides the sidenote counter. The optional trailing `[voffset]` shifts the note from the call-site baseline; it is additive to collision avoidance, identical in mechanics to `\marginnote`'s offset.

8.2.2 Separated mark and text

Use `\sidenotemark` and `\sidenotetext` when the mark and the note body need to appear at different call sites, for instance when the mark must land inside an environment incompatible with `\vadjust` such as `tabular`.

8.2.3 Mark formatting hooks

Two commands control rendering:

```
\canonssidenotemarkformat{num}   % inline mark (default: superscript)
\canonssidenotelabelformat{num}  % margin label (default: superscript + thin space)
```

Redefine to change mark style document-wide:

```
\renewcommand*{\canonssidenotemarkformat}[1]{$^{#1}$}
\renewcommand*{\canonssidenotelabelformat}[1]{\textbf{#1}\enspace}
```

8.2.4 Numbering modes

The `numbering` key controls counter resets. Set at load time via the package option or in the preamble via `\pagecanonsetup`.

Table 8.1: Numbering options

Value	**Behavior**
`global` (default)	one counter for the entire document
`persection`	reset at each `\section`
`perchapter`	reset at each `\chapter` (book/report only)
`perpage`	reset at each page (requires two passes)

8.3 Shared configuration

Size, line spread, and justification apply to both margin notes and sidenotes via unified keys; split keys override the unified setting.

Table 8.2: Shared options

Unified	Split overrides
`size=`	`marginnotesize=`, `sidenotesize=` (with `marginsize=false`)
`justification=`	`marginnotejustify=`, `sidenotejustify=` (with `marginjustify=false`)
`notespread=`	(no split)

Changes via `\pagecanonsetup` take effect immediately on the next note call; no page break is triggered because the underlying macros are evaluated at typesetting time.

8.4 Disabling marginalia and deferment

The marginalia engine can be disabled at load time:

```
\usepackage[nomarginalia]{page-canons}
```

With `nomarginalia`, page-canons does not redefine `\marginnote`, does not install the sidenote commands, and does not integrate with sidenotes if loaded. The canon, margin mode, gutter, and fullwidth machinery are unaffected; only marginalia is suppressed. Use this when another package owns marginal placement and page-canons is wanted purely for its geometry.

Furthermore, page-canons checks at `\begindocument` whether `\marginnote` is already defined; if so, it leaves the existing definition intact and emits a `PackageInfo` log entry. This handles loading another marginalia provider (marginnote [3], marginalia, similar) alongside page-canons without an explicit `nomarginalia` flag.

sidenotes [5] is detected separately and integrated rather than stepped aside from; the in-house sidenote system defers to it when present.

Pass `nomarginalia` explicitly when intent is to silence the `PackageInfo` entry as well.

8.5 Accessibility and tagging

Note.

Design for accessibility is a work-in-progress. Feedback and constructive criticism is welcome here! I outline the general architecture; whether this is correct or how this can be improved with specific feedback would be much appreciated.

page-canons wraps margin notes and sidenotes in PDF `Aside` structure elements when document-level tagging is active. PDF/UA-2 designates `Aside` as the structure type for content tangential to the main reading flow; margin notes and sidenotes fit this categorization.

page-canons does not enable tagging itself; tagging is enabled via `\DocumentMetadata`. This manual is compiled with

```
\DocumentMetadata{
  pdfversion=2.0,
  pdfstandard={UA-2,a-4f},
  tagging=on,
  tagging-setup={activate-all},
  ...
}
```

declared before `\documentclass`. At `\begindocument`, page-canons tests for an active tagging session via `\tag_if_active:TF` (with a fallback path for older tagpdf releases that exposed the state as `\g__tag_active_bool`). If tagging is active, then marginalia hooks install and emit a confirming `PackageInfo`; if not, the hooks remain no-ops.

Note.

Disabling marginalia (or deferring to another marginalia provider) also disables the `Aside` hooks: the hooks are invoked only from page-canons's own `\marginnote` implementation. If marginalia comes from another package, that package is responsible for any tagging.

Margin notes called in horizontal mode (inline within paragraphs) are not nested inside the surrounding paragraph element. Doing so would place tangential content inside the main reading flow, contradicting the semantics of `Aside`. Instead, the implementation ends the current marked-content sequence, walks the tagpdf structure stack upward for

the nearest `Part` or `Sect` ancestor, begins an `Aside` as a child of that ancestor, places the note content, closes the `Aside`, and resumes the marked-content sequence.

Margin notes called in vertical mode begin an `Aside` in the local context without parent rerouting; vertical placement is already outside the inline flow.

Geometric placement uses `\vadjust`, `\rlap`/`\llap`, and zero-dimension boxes whose internals are positioning machinery. `page-canons` suspends tagging during these operations and resumes after, preventing the placement scaffolding from affecting the structure tree.

Chapter 9

Fullwidth and captions

9.1 The `fullwidth` environment

```
\begin{fullwidth}[skip=<dim>, justification=<value>]
  <content>
\end{fullwidth}
```

Extends content into the outer margin by setting a minipage of width `\linewidth` + `\fullwidthoverhang`. Does not float, does not break across pages. Inside the environment, `\linewidth` gives the full container width; `\textwidth` is reset to the minipage width.

Options:

Table 9.1: Fullwidth options

Key	Values	Default
`skip`	dimension	`0.5\baselineskip`
`justification`	`centering`, `raggedright`, `raggedleft`, `justified`	`centering`

9.2 Captions inside fullwidth

Inside `fullwidth`, captions align to the text-block width rather than the full minipage width, on the side matching the current margin mode (symmetric: outer; antisymmetric: opposite; right: flush left; left: indented by overhang). The package installs its own `\@makecaption` when caption is not loaded; if caption is present, its formatting takes precedence.

Note.

This is a design decision encoded into the source code.

Chapter 10

Runtime reconfiguration

`\pagecanonsetup{key=value, …}` reconfigures the layout at any point. Mid-document calls that change layout keys invoke `\newgeometry` internally and therefore trigger a page break. Margin-related keys (`size=`, `notespread=`, etc.) take effect immediately without a page break.

10.1 Two-pass key processing

The first pass scans for `canon=` and resets all canon-specific parameters (`vdhN`, `ateliersstyle`, `gridN`, `gridinner`, `gridouter`, `gridtop`, `gridbottom`) to compiled defaults if the canon value is changing. The second pass processes all keys left-to-right. Key order is therefore irrelevant when switching canons.

10.2 Reset semantics

1. Canon changes reset canon-specific parameters. Supply overrides in the same call.
2. Same-canon calls do not reset. Existing parameters persist, enabling incremental adjustment.
3. Omitting `canon=` never resets anything.

10.3 Preamble safety

`\pagecanonsetup` is safe in the preamble. Layout keys are stored and applied at `\begin{document}`. Margin-related keys are also applied at load time, so their defaults are in effect before `\begin{document}`.

Part III

Reference and validation

Chapter 11

Test suite

Tests live here rather than in the feature chapters. Each test is a minimal check: the preamble gives the configuration, the body exercises the feature, and expected output is stated briefly. Visual tests rely on the page's `showframe` rules; numerical tests print computed dimensions.

11.1 Basic dimensions

Test 1: Exported dimensions (VdG, letterpaper)

`\textwidth` = 409.52376pt

`\marginparwidth` = 109.20946pt

`\marginparsep` = 13.64766pt

`\marginandtext` = 532.38087pt

`\marginandsep` = 122.85712pt

`\textandsep` = 423.17142pt

`\fullwidthoverhang` = 122.85712pt

`\overflowingheadlen` = 532.38087pt

Test 2: VdG text block proportion

Expected: `\textwidth` $\approx 2/3 \times 8.5\,\text{in} \approx 408\,\text{pt}$.

Computed: 409.52376pt.

11.2 Canon switching

Test 3: VdH N=9

`\textwidth` = 460.72127pt

`\marginparwidth` = 86.62894pt

Frames should show narrower margins than the preceding VdG pages. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Test 4: Tufte

`\textwidth` = 312.9309pt

`\marginparwidth` = 144.53778pt

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Test 5: Ateliers luxury

`\textwidth` = 383.93439pt

`\marginparwidth` = 65.24821pt

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Test 6: Grid 9×9 with custom cells

`\textwidth` = 341.27498pt

`\marginparwidth` = 182.01334pt

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Test 7: `canon=false`

`\textwidth` = 430.00462pt

Expected: geometry default (≈ 345 pt on letterpaper).

11.3 Margin notes

Test 8: Single note

Single note. Should appear in the outer margin at the call-site baseline. Width: 109.20946pt.

This sentence contains a margin note.The note should appear in the outer margin without disturbing the flow of body text.

Nunc sed pede. Praesent vitae lectus. Praesent neque justo, vehicula eget, interdum id, facilisis et, nibh. Phasellus at purus et libero lacinia dictum. Fusce aliquet. Nulla eu ante placerat leo semper dictum. Mauris metus. Curabitur lobortis. Curabitur sollicitudin hendrerit nunc. Donec ultrices lacus id ipsum.

Test 9: Three adjacent notes (collision)

First note.

Second note: pushed below first by collision avoidance.

Third note: pushed below second.

Three notes are placed in quick succession. All three should appear without overlapping. If the notes exceed the available margin depth they overflow the page foot.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Test 10: Positive offset

Call site for a note pushed 2 cm down.

Test 11: Negative offset

Shifted 2 cm below baseline.

Shifted 1 cm above baseline.

Call site for a note pulled 1 cm up.

Test 12: Zero explicit offset

Call site for a note with explicit zero offset. Behaviorally identical to the auto path.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Test 13: Font isolation inside italic

This sentence is *entirely italic and contains a margin notemid-run*, and the italic continues after.

Upright despite italic context.

Test 14: Font isolation inside bold

Bold sentence with a margin noteinside.

Normal weight despite bold context.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Test 15: Small-font notes

Zero offset: same as auto.

Small-font note.

Margin note text is now typeset in `small`. Donec a nibh ut elit vestibulum tristique. Integer at pede. Cras volutpat varius magna. Phasellus eu wisi. Praesent risus justo, lobortis eget, scelerisque ac, aliquet in, dolor. Proin id leo. Nunc iaculis, mi vitae accumsan commodo, neque sem lacinia nulla, quis vestibulum justo sem in eros. Quisque sed massa. Morbi lectus ipsum, vulputate a, mollis ut, accumsan placerat, tellus. Nullam in wisi. Vivamus eu ligula a nunc accumsan congue. Suspendisse ac libero. Aliquam erat volutpat. Donec augue. Nunc venenatis fringilla nibh. Fusce accumsan pulvinar justo. Nullam semper, dui ut dignissim auctor, orci libero fringilla massa, blandit pulvinar pede tortor id magna. Nunc adipiscing justo sed velit tincidunt fermentum.

Test 16: Wider line spread

Line spacing increased to 1.2. Integer placerat. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Sed in massa. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Phasellus tempus aliquam risus. Aliquam rutrum purus at metus. Donec posuere odio at erat. Nam non nibh. Phasellus ligula. Quisque venenatis lectus in augue. Sed vestibulum dapibus neque.

Wider spread: 1.2. Lines in this note are more widely spaced.

Test 17: Ragged-left notes

Note justification changed to `raggedleft`. Mauris tempus eros at nulla. Sed quis dui dignissim mauris pretium tincidunt. Mauris ac purus. Phasellus ac libero. Etiam dapibus iaculis nunc. In lectus wisi, elementum eu, sollicitudin nec, imperdiet quis, dui. Nulla viverra neque ac libero. Mauris urna leo, adipiscing eu, ultrices non, blandit eu, dui. Maecenas dui neque, suscipit sit amet, rutrum a, laoreet in, eros. Ut eu nibh. Fusce nec erat tempus urna fringilla tempus. Curabitur id enim. Sed ante. Cras sodales enim sit amet wisi. Nunc fermentum consequat quam.

Ragged left note. Text is flush right inside the margin column.

11.4 Sidenotes

Test 18: Auto-numbered sidenote

The sidenote command attaches a mark in the text[1] and places the numbered annotation in the margin.

[1] First sidenote. Auto-numbered from the global counter. Should appear in the outer margin.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Test 19: Sidenote with positive offset

Sidenote pushed 2 cm down from baseline.[2]

Test 20: Sidenote with negative offset

Sidenote pulled 1 cm up.[3]

[3] Shifted 1cm above baseline.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Test 21: Separated mark and text

[4] The mark was placed here by `\sidenotemark`.

[4] This note text was placed by `\sidenotetext` immediately after the mark, with a 1 cm push.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Test 22: Independent margin-note and sidenote sizes

Small margin note.

[5] Footnotesize sidenote.

Text with a sidenote[5] next to a margin note. The margin note uses `small`; the sidenote uses `footnotesize`.

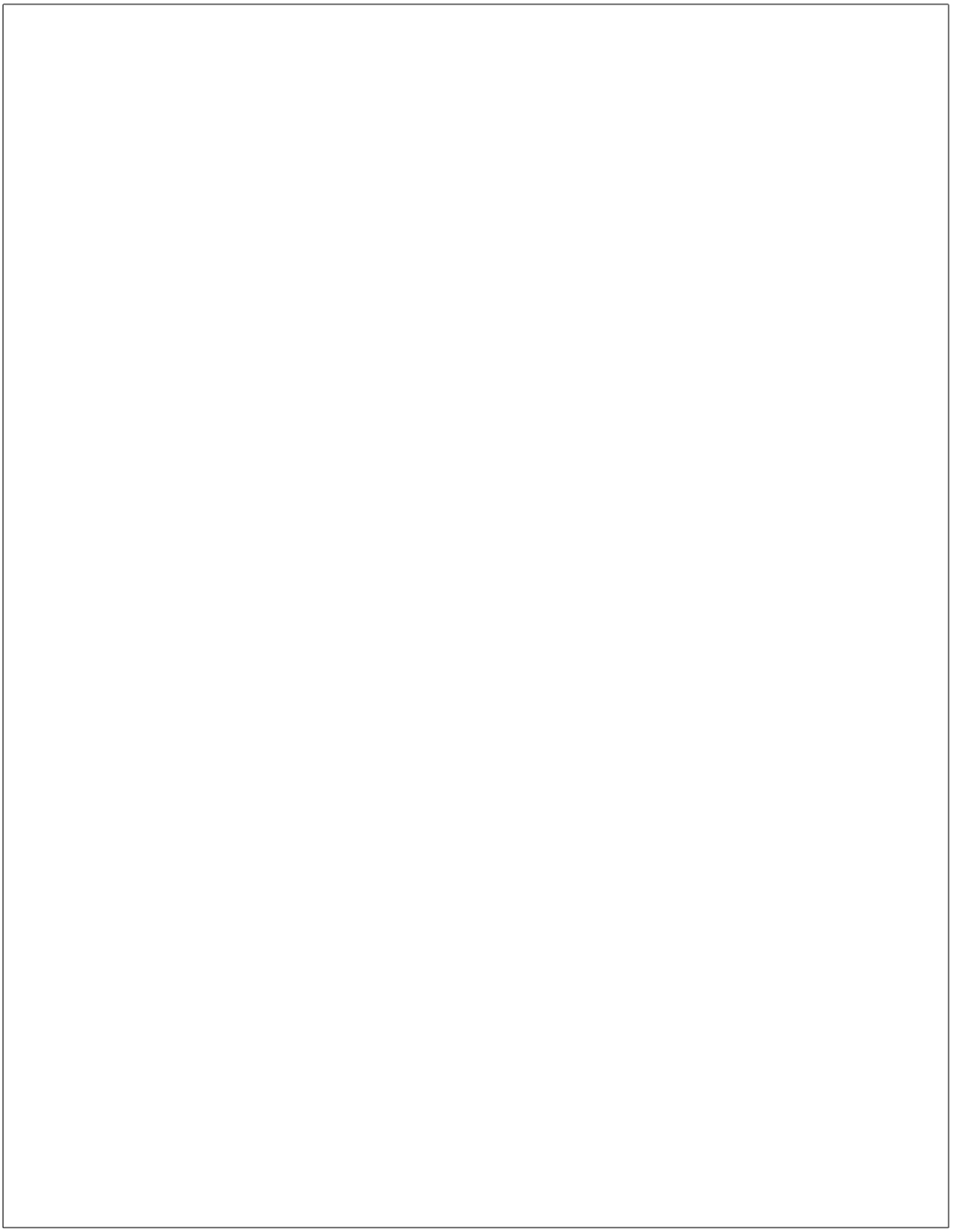
Donec a nibh ut elit vestibulum tristique. Integer at pede. Cras volutpat varius magna. Phasellus eu wisi. Praesent risus justo, lobortis eget, scelerisque ac, aliquet in, dolor. Proin id leo. Nunc iaculis, mi vitae accumsan commodo, neque sem lacinia nulla, quis vestibulum justo sem in eros. Quisque sed massa. Morbi lectus ipsum, vulputate a, mollis ut, accumsan placerat, tellus. Nullam in wisi. Vivamus eu ligula a nunc accumsan congue. Suspendisse ac libero. Aliquam erat volutpat. Donec augue. Nunc venenatis fringilla nibh. Fusce accumsan pulvinar justo. Nullam semper, dui ut dignissim auctor, orci libero fringilla massa, blandit pulvinar pede tortor id magna. Nunc adipiscing justo sed velit tincidunt fermentum.

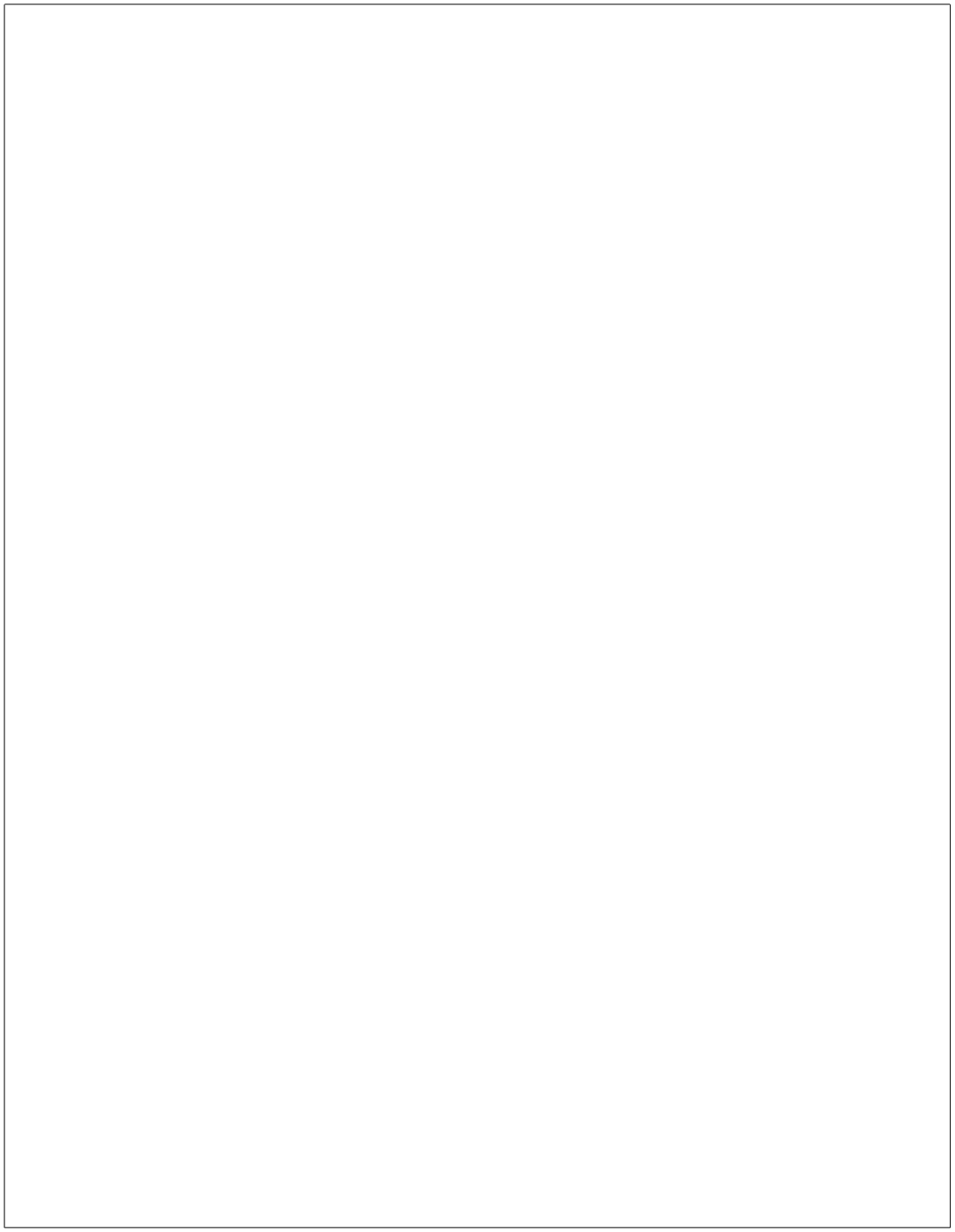
11.5 Margin modes

Test 23: Symmetric fullwidth on odd page

Symmetric, odd page: fullwidth extends into the right (outer) margin.

Test 24: Symmetric fullwidth on forced even page





Symmetric, even page: fullwidth extends leftward into the outer margin.	

Test 25: Antisymmetric margin note and fullwidth

Margin note should appear on the inner margin.

Antisymmetric: note is in the inner (spine-side) margin.

Antisymmetric: fullwidth extends into the inner margin.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Test 26: Right-margin fullwidth

Right-margin mode: fullwidth always extends rightward.		
Test 27: Left-margin fullwidth		

Left-margin mode: fullwidth always extends leftward.

11.6 Gutters

Test 28: Gutter in `geometry` mode

`\textwidth` = 409.52376pt

Text width unchanged from zero-gutter VdG; inner margin is 12 mm wider.

Test 29: Gutter in `satzspiegel` mode

`\textwidth` = 386.76192pt

Text width narrower: canon computed on reduced leaf width $W - 12\,\mathrm{mm}$.

11.7 Fullwidth environment

Test 30: Default fullwidth

Default: 0.5\baselineskip skip, centered. Width = 523.58089pt.

Test 31: Custom skip and justification

2\baselineskip skip above. Ragged right.

Test 32: Justified

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Test 33: Zero skip

Zero skip.

Test 34: Figure with caption in fullwidth

Figure: Caption confined to text column width. On odd pages it is flush left; on even pages it is indented by the overhang distance.

11.8 Runtime reconfiguration

Test 35: Chained reconfiguration

`\textwidth` = 409.52376pt

After Tufte + right: `\textwidth` = 312.9309pt `\marginparwidth` = 144.53778pt		
Tufte + right: fullwidth extends rightward on every page.		

After Ateliers neater + symmetric: `\textwidth` = 409.53314pt		
Ateliers neater + symmetric. Narrower overhang than Tufte.		

Test 36: Canon switch resets stale parameters

Grid 9×9, outer=3: `\textwidth` = 341.27498pt

VdG (grid params discarded): `\textwidth` = 409.52376pt

Grid defaults (should be 6×6, outer=2): `\textwidth` = 307.14748pt

The last width must match the 6×6 default, not the prior 9×9 outer=3 value.

Test 37: Key order independence

Order `gridN, canon, gridouter`: `\textwidth` = 409.52998pt

Order `canon, gridouter, gridN`: `\textwidth` = 409.52998pt

Both widths must be identical.

Test 38: Same-canon persistence

Grid 9×9, outer=3: `\textwidth` = 341.27498pt

Same canon, no reset: `\textwidth` = 341.27498pt

Both widths must be identical.

Test 39: Recovery from `canon=false`

\textwidth = 430.00462pt

`\textwidth` = 312.9309pt

Should match earlier Tufte values; `canon=false` must not corrupt state.

Chapter 12

Edge cases and invariants

12.1 Margin guard

For non-grid canons, `marginparwidth` + 2 `marginparsep` ≤ `outermargin` is enforced. If violated, `marginparwidth` is shrunk with a warning.

Test 40: Margin guard invariant

`marginparwidth` + 2 `marginparsep` = 136.50478*pt*.

Must not exceed the outer margin width visible in the `showframe` rules.

12.2 Grid $N = 3$ failsafe

When `gridN=3` with default cell counts (inner=1, outer=2, top=1, bottom=2), the text area would be zero. The package adjusts to 1:1 allocation.

Test 41: Grid $N = 3$ failsafe

`\textwidth` = 204.765pt

Expected: $\approx 1/3$ of paper width (one cell of three).

12.3 Font isolation

`\normalfont` is prepended before `\marginfont` at every content-building site. Notes cannot inherit italic, bold, or any other font variant from the call site. See the marginnotes test section.

12.4 Twocolumn

Loading page-canons in twocolumn mode emits a package warning; marginalia placement is not guaranteed. Not tested here.

12.5 Unknown canon

Passing an unrecognized `canon` value triggers a warning and falls back to `vdg`.

Chapter 13

Quick reference

13.1 Package options

Table 13.1: Canon selection options

Key	Values	Default
`canon`	`vdg`, `vdh`, `tufte`, `ateliers`, `grid`, `false`	`vdg`
`vdhN`	3, 6, 9, 12, 15	6
`atelierssstyle`	`ordinary`, `neater`, `luxury`	`ordinary`
`gridN`	integer ≥ 3	6
`gridinner`	integer	1
`gridouter`	integer	2
`gridtop`	integer	1
`gridbottom`	integer	2

13.2 Fullwidth options

13.3 Commands

13.4 Exported lengths

Table 13.2: Layout selection options

Key	Values	Default
`margins`	`symmetric`, `antisymmetric`, `right`, `left`	class-dep.
`gutterval`	dimension	`0mm`
`guttermode`	`geometry`, `satzspiegel`	`geometry`
`showframe`	bare boolean	off
`landscape`	bare boolean	off
`debug`	bare boolean	off

Table 13.3: Marginalia options; unified

Key	Values	Default
`size`	font size name	`footnotesize`
`justification`	`default`, `raggedright`, `raggedleft`, `centered`, `justified`	`default`
`notespread`	factor	`1.05`
`numbering`	`global`, `persection`, `perchapter`, `perpage`	`global`
`nomarginalia`	bare boolean	off

Table 13.4: Marginalia options; split (override unified)

Key	Values	Default
`marginsize`	`true`/`false`	`true`
`marginnotesize`	font size name	follows `size`
`sidenotesize`	font size name	follows `size`
`marginjustify`	`true`/`false`	`true`
`marginnotejustify`	justification value	follows `justification`
`sidenotejustify`	justification value	follows `justification`

Table 13.5: Options for fullwidth

Key	Values	Default
`skip`	dimension	`0.5\baselineskip`
`justification`	`centering`, `raggedright`, `raggedleft`, `justified`	`centering`

Table 13.6: Options for commands

Command	Action
`\pagecanonsetup{…}`	reconfigure; layout keys trigger page break
`\pagecanonmargins`	expands to current margin mode
`\marginnote{text}`	margin note at call-site baseline
`\marginnote{text}[dim]`	margin note with explicit voffset
`\sidenote{text}`	numbered sidenote, auto-position
`\sidenote{text}[dim]`	numbered sidenote with voffset
`\sidenote[n]{text}`	sidenote with explicit number
`\sidenote[n]{text}[dim]`	explicit number and voffset
`\sidenotetext{text}`	note text only (no mark, no step)
`\sidenotetext{text}[dim]`	note text with voffset
`\sidenotemark`	inline mark only (steps counter)
`\sidenotemark[n]`	inline mark with explicit number
`\canonssidenotemarkformat{n}`	format hook: inline mark
`\canonssidenotelabelformat{n}`	format hook: margin label
`\canonsswitchmargin`	toggle current marginpar side
`\canonsresetmargin`	restore marginpar side to mode default

Table 13.7: Exported lengths

Length	Value
`\marginandtext`	`\textwidth` + `\marginparwidth` + `\marginparsep`
`\marginandsep`	`\marginparwidth` + `\marginparsep`
`\textandsep`	`\textwidth` + `\marginparsep`
`\fullwidthoverhang`	`\marginparwidth` + `\marginparsep`
`\overflowingheadlen`	same as `\marginandtext`

Bibliography

[1] Bil Kleb, Bill Wood, and Kevin Godby. *tufte-latex. Document classes inspired by the work of Edward Tufte*. Version 3.5.2. Repository: `https://tufte-latex.github.io/tufte-latex/`. CTAN. URL: `https://ctan.org/pkg/tufte-latex` (visited on 10/04/2025).

[2] Markus Kohm. *KOMA-Script. A bundle of versatile classes and packages*. Version 3.48. Available at `https://komascript.de`. CTAN. Sept. 2025. URL: `https://ctan.org/pkg/koma-script` (visited on 10/04/2025).

[3] Markus Kohm. *marginnote. Notes in the margin, even where* `\marginpar` *fails*. Version 2023-09-07. Repository: `https://github.com/komascript/marginnote.git`. CTAN. Sept. 2023. URL: `https://ctan.org/pkg/marginnote` (visited on 10/04/2025).

[4] Lars Madsen and Peter R. Wilson. *memoir. Typeset fiction, non-fiction and mathematical books*. Version 3.8.4. CTAN. Oct. 2025. URL: `https://ctan.org/pkg/memoir` (visited on 10/04/2025).

[5] Andy Thomas. *sidenotes. Typeset notes containing rich content, in the margin*. Version 1.20. CTAN. Sept. 2024. URL: `https://ctan.org/pkg/sidenotes` (visited on 10/04/2025).

[6] E.R. Tufte. *Beautiful Evidence*. Graphics Press, 2006. ISBN: 9780961392178.

[7] E.R. Tufte. *Envisioning Information*. Graphics Press, 1990. ISBN: 9781930824140.

[8] E.R. Tufte. *Seeing with Fresh Eyes: Meaning, Space, Data, Truth*. Graphics Press, 2020. ISBN: 9780961392192.

[9] E.R. Tufte. *The Visual Display of Quantitative Information*. Graphics Press, 2001. ISBN: 9781930824133.

[10] E.R. Tufte. *Visual Explanations: Images and Quantities, Evidence and Narrative*. Graphics Press, 1997. ISBN: 9781930824157.

Appendix A

Formulary

Frames, symbols, order

$W = \texttt{\textbackslash paperwidth}, \quad H = \texttt{\textbackslash paperheight}, \quad W_* = \begin{cases} W & \text{(default)} \\ W - g & \text{if } \texttt{guttermode=satzspiegel} \end{cases}$

Order: (1) paper → (2) canon → (3) gutter mode → (4) margin mode → (5) export lengths.

Canon summaries

Van de Graaf (vdg). $m_i = W_*/9$, $m_o = 2W_*/9$, $m_t = H/9$, $m_b = 2H/9$. $\texttt{\textbackslash textwidth} = \frac{2}{3}W_*$, $\texttt{\textbackslash textheight} = \frac{2}{3}H$. Not centered; outer/bottom > inner/top.

Villard de Honnecourt (vdh, $N \in \{3, 6, 9, 12, 15\}$**).** With $u = W_*/(N+3)$: $m_i = u$, $m_o = 2u$, $m_t = 1.5u$, $m_b = 3u$. $\texttt{\textbackslash textwidth} = \frac{N}{N+3}W_*$, $\texttt{\textbackslash textheight} = H - 4.5u$. Verticals depend on W_*, so block aspect varies with H/W.

Tufte (tufte). $m_i \approx 2W/17$, $m_o \approx 3W/8$ (code $0.372941W$), $m_t = H/11$, $m_b = 3H/22$. $\texttt{\textbackslash textwidth} \approx 0.509412\,W$. Notes: $\texttt{marginparwidth} \approx 4W/17$, $\texttt{marginparsep} \approx W/26$.

Canon des Ateliers (ateliers). With $k \in \{\frac{3}{4}, \frac{2}{3}, \frac{5}{8}\}$ (ordinary/neater/luxury) and $w = (1-k)W$: $m_i = 0.4w$, $m_o = 0.6w$, $m_t = 0.5w$, $m_b = 0.7w$.

Explicitly:

$$\begin{aligned} \text{ordinary}: &\ (m_i, m_o, m_t, m_b) = (\tfrac{1}{10}, \tfrac{3}{20}, \tfrac{1}{8}, \tfrac{7}{40})W \\ \text{neater}: &\ (\tfrac{2}{15}, \tfrac{1}{5}, \tfrac{1}{6}, \tfrac{7}{30})W \\ \text{luxury}: &\ (\tfrac{3}{20}, \tfrac{9}{40}, \tfrac{3}{16}, \tfrac{21}{80})W \end{aligned}$$

`\textwidth` $= kW$. Aspect varies with H/W.

Grid (grid). Pick $N \geq 3$. Cell size $c_w = W_*/N$, $c_h = H/N$. Choose integers $L, R, T, B \geq 0$ with $L + R < N$ and $T + B < N$: $m_i = Lc_w$, $m_o = Rc_w$, $m_t = Tc_h$, $m_b = Bc_h$. `\textwidth` $= (N{-}L{-}R)c_w$, `\textheight` $= (N{-}T{-}B)c_h$.

Area fraction (relative to $W_* \times H$):

$$\frac{\texttt{\textwidth}}{W_*} \cdot \frac{\texttt{\textheight}}{H} = \left(1 - \tfrac{L+R}{N}\right)\left(1 - \tfrac{T+B}{N}\right).$$

VdG exactly if N multiple of 9 and $(L, R, T, B) = (N/9, 2N/9, N/9, 2N/9)$. Honnecourt is generally approximated; Ateliers, rounded.

Footers (all canons). `footskip` $= \frac{1}{2}m_b$.

Gutter modes

`geometry` $\text{inner}_{\text{final}} = m_i + g$, $\text{outer}_{\text{final}} = m_o - g$. Preserves `\textwidth`; may clamp outer to 0 if g is large.

`satzspiegel` Compute canon on $W_* = W - g$, then add $+g$ to inner. Preserves proportions; Δ`\textwidth` $= -\alpha\, g$ with $\alpha = \frac{2}{3}$ (vdg), $\frac{N}{N+3}$ (vdh), ≈ 0.509412 (tufte), k (ateliers), $\frac{N-L-R}{N}$ (grid).