

Margin Notes

Setup

Unfortunately typst doesn’t expose margins to calling functions, so you’ll need to set them explicitly. This is done using `set-page-properties` **before you place any content**:

```
// At the top of your source file
// Of course, you can substitute any margin numbers you prefer
// provided the page margins match what you pass to `set-page-properties`
#import "../drafting.typ": *
#let (l-margin, r-margin) = (1in, 2in)
#set page(
  margin: (left: l-margin, right: r-margin, rest: 0.1in),
  paper: "us-letter",
)
#set-page-properties(margin-left: l-margin, margin-right: r-margin)
```

The basics

```
#lorem(20)
#margin-note[Hello, world!]

#lorem(25)
#margin-note[When notes are about to overlap, they're automatically shifted]
#margin-note(stroke: aqua + 3pt)[Notice the lack of collision]
#lorem(25)
#margin-note(side: left, dy: -50pt)[You can also explicitly specify the offset]

#let caution-rect = rect.with(inset: 1em, radius: 0.5em, fill: orange.lighten(80%))
#inline-note(rect: caution-rect)[
  Be aware that notes will stop automatically avoiding collisions if 4 or more notes
  overlap. This is because `typst` warns when the layout doesn't resolve after 5 attempts
  (initial layout + adjustment for each note)
]
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aequi doleamus. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aequi doleamus.

Be aware that notes will stop automatically avoiding collisions if 4 or more notes overlap. This is because typst warns when the layout doesn't resolve after 5 attempts (initial layout + adjustment for each note)

Hello, world!

When notes are about to overlap, they’re automatically shifted

Notice the lack of collision

Adjusting the default style

All function defaults are customizable through updating the module state:

```
#lorem(4) #margin-note[Default style]
#set-margin-note-defaults(stroke: orange, side: left)
#lorem(4) #margin-note[Updated style]
```

Lorem ipsum dolor sit. Lorem ipsum dolor sit.

Even deeper customization is possible by overriding the default rect:

```
#import "@preview/colorful-boxes:1.1.0": stickybox

#let default-rect(stroke: none, fill: none, ..args, content) = {
  stickybox(rotation: -20deg, ..args, content)
}
#set-margin-note-defaults(rect: default-rect, stroke: none, side: right)

#lorem(10)
#margin-note(dy: -25pt)[Why not use sticky notes in the margin?]
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do.

Why not use sticky notes in the margin?

Multiple document reviewers

```
#let reviewer-a = margin-note.with(stroke: blue)
#let reviewer-b = margin-note.with(stroke: purple)
#lorem(20)
#reviewer-a[Comment from reviewer A]
#lorem(15)
#reviewer-b(side: left)[Comment from reviewer B]
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore.

Comment from reviewer A

Inline Notes

```
#lorem(10)
#inline-note[The default inline note will split the paragraph at its location]
#lorem(10)
#inline-note(par-break: false, stroke: (paint: orange, dash: "dashed"))[
  But you can specify `par-break: false` to prevent this
]
#lorem(10)
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do.

The default inline note will split the paragraph at its location

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do. But you can specify `par-break: false` to prevent this Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do.

Hiding notes for print preview

```
#set-margin-note-defaults(hidden: true)

#lorem(20)
#margin-note[This will respect the global "hidden" state]
#margin-note(hidden: false, dy: -2.5em)[This note will never be hidden]
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat.

This note will never be hidden

Feature roadmap

- Incorporate logic from <https://github.com/typst/typst/issues/763> when it’s resolved to avoid users explicitly calling `drafting.set-page-properties`

Positioning

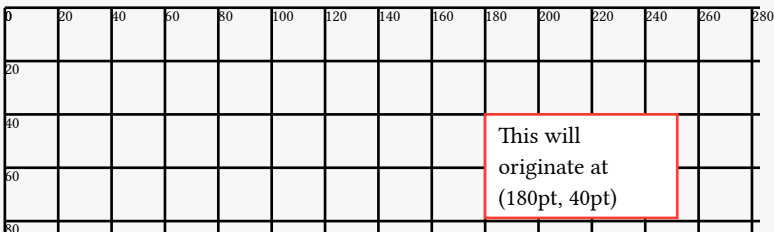
Precise placement: rule grid

Need to measure space for fine-tuned positioning? You can use `rule-grid`. Just note that dimensions can’t be specified using %:

```
#rule-grid(width: 10cm, height: 3cm, spacing: 20pt)
#place(
  dx: 180pt,
  dy: 40pt,
  rect(fill: white, stroke: red, width: 1in, "This will originate at (180pt, 40pt)")
)

// Optionally specify divisions of the smallest dimension to automatically calculate
// spacing
#rule-grid(dx: 10.5cm, width: 3cm, height: 1cm, divisions: 3, color: green)

// The rule grid doesn't take up space, so add it explicitly
#v(3cm + 1em)
```



Absolute positioning

What about absolutely positioning something regardless of margin and relative location? `absolute-place` is your friend. You can put content anywhere:

```
#locate(loc => {
  let dx, dy = (3.5in, loc.position().y - 1.5in)
  let content-str = (
    "This absolutely-placed box will originate at (" + repr(dx) + ", " + repr(dy) + ")"
    + " in page coordinates"
  )
  absolute-place(
    dx: dx, dy: dy,
    rect(
      fill: green.lighten(60%),
      radius: 0.5em,
      width: 1.5in,
      [#content-str]
    )
  )
})
Look for the green box to the right
```

This absolutely-placed box will originate at (252pt, 1803.12pt) in page coordinates

Look for the green box to the right

The “rule-grid” also supports absolute placement at the top-left of the page by passing `relative: false`. This is helpful for “rule”-ing the whole page.

Feature roadmap

- Allow percentage-based dimensions for `rule-grid`
- Allow independent (x, y) divisions or spacing rather than forcing square units