

如何看待Typst?

但是中文支持约等于0（虽然官方选项里面有中文），字体不统一，选了serif字体，中文字体仍然是无衬线的，而且默认的大小与英文相差较远。两端对齐算法倒是仍然工作良好。英文字母与中文字母之间没有自动调整空白，这是瑕疵。

而且web app似乎十分不稳定。

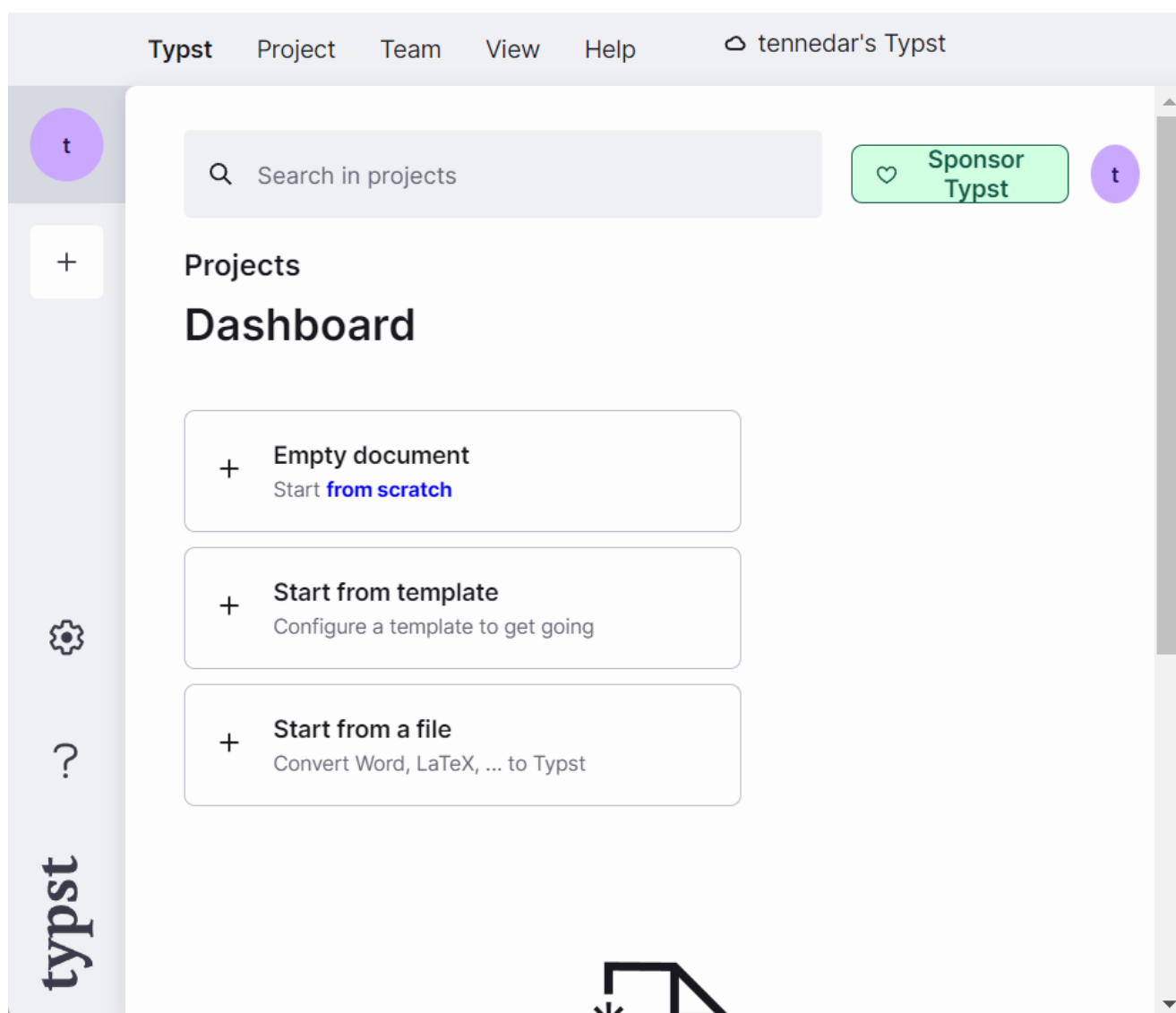
总体来说，typst帮你在背后做了很多优雅的工作，确实让科学写作更容易上手了，也确实能够覆盖大部分场景的使用。而且语义化的公式输入也确实更优雅。但相比于TeX的积累和自定义灵活性，显然还是有距离的。web app的稳定性也需要提升。

我个人认为，web做编辑器，始终离原生的桌面应用还是差了些，但考虑到协作性这个需求，可能这也是typst暂时的最优解吧。

Typst中文文档 快捷符号

1.基本语法

主页



文本格式

= #标题

_ _ #斜体

+ #有序列表

- #无序列表

可以嵌套

段落中断 空行

着重强调 ***strong***

代码段 ``print(1)``

链接 <https://typst.app/>

术语列表 / Term: description

行中断 \

转义字符 Tweet at us \#ad

注释 `/* block */, // line`

```
5 = 标题
6 =标题
7 这是一个_斜体_测试
8 这是一个_斜体_测试
9 test_aaa_test
10 test_aaa_test
11 + aaa
12 + bbb
13 - adfsf
14 - adfljsaejfka
15 + ccc
```

测试 Test 테스트 마지막

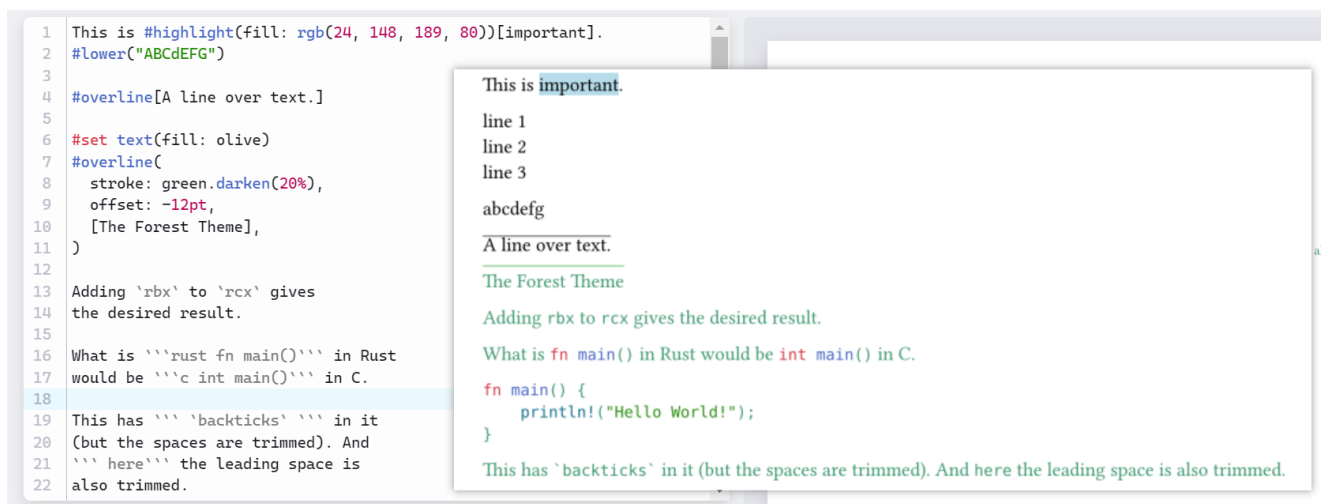
标题

=标题 这是一个斜体测试 这是一个斜体测试 test aaa test test_aaa_test

1. aaa
2. bbb
 - adfsf
 - adfljsaejfka
3. ccc

/ **Ligature:** A merged glyph.
/ **Kerning:** A spacing adjustment
between two adjacent letters.

Ligature A merged glyph.
Kerning A spacing adjustment between two
adjacent letters.



图像

函数

```
#image("image.jpg")
#image("image.jpg",width:80%) 1cm 0.7in
```

this is a image aka @label

```
#figure(
  image("image.jpg",width:70%),
  caption:[
    Explanation
  ]
)<label>
```

[内容块]可包含文本、标记、函数调用
"字符串"

参考文献

在撰写报告时，您需要引用资料来支持您的一些论证。 您可以使用 bibliography 函数向文档添加参考文献。 此函数需要参考文献文件的路径。

Typst 的原生参考文献格式是 Hayagriva，但为了兼容性，您也可以使用 BibLaTeX 文件。由于您的同学已经完成了文献调查并向您发送了 .bib 文件，因此您将直接使用该文件。请在文件面板中上传文件，以在 Typst App 中访问它。

```
#bibliography("test.bib")
```

公式

$\$表达式\$$

$\$A=b+c\$$

公式是内联排版的，与周围的文本在同一行上。如果你想把它放在它自己的新行上，你应该在其开头和结尾插入一个空格：

```
$ A=b+c $
```

多个字母将被视作为符号、变量或函数名称。要隐式地表明这是单个字母之间的乘法，请在它们之间加入空格。

如果你想要一个由多个字母组成的变量，你可以用引号括起来：

```
$Q = rho A b + "time offset" $
```

```
sum_(i=0)^n A/B #求和  
_ #下标（圆括号）  
^ #上标（圆括号）  
/ #分数
```

列向量 `vec(x_1,x_2,x_3)`

```
1 $x \ y$  
2  
3 $x \&= 2 \&= 3$  
4  
5 $pi$  
6  
7 $arrow.r.long$  
8  
9 $->, !=$快捷符号  
10  
11 $floor(x)$数学函数调用  
12  
13 $#rect(width: 1cm)$代码表达式  
14  
15 $x\^2$转义字符  
16  
17 $/* comment */$
```

x
 y
 $x = 2$
 $= 3$
 π
 \rightarrow
 \rightarrow, \neq 快捷符号
 $[x]$ 数学函数调用
代码表达式
 x^2 转义字符

2. 格式调整

句/段/页

```
#set text(
  font: "New Computer Modern",
  size: 10pt
)
#set page(
  paper: "a6",
  margin: (x: 1.8cm, y: 1.5cm),
)
#set par(
  justify: true,
  leading: 0.52em,
)
```

= Introduction

In this report, we will explore the various factors that influence fluid dynamics in glaciers and how they contribute to the formation and behavior of these natural structures.

...

```
#align(center + bottom)[
  #image("glacier.jpg", width: 70%)

  *Glaciers form an important
  part of the earth's climate
  system.*
]
```

首先是 `page` 的 Set 规则，它接收两个参数：页面大小和页面边距。页面大小为字符串，Typst 接受 [许多标准页面大小](#)，但您也可以指定自定义页面大小。边距为一个 [字典](#)，字典是键值对的集合。在本例中，键为 `x` 和 `y`，值分别为水平边距和垂直边距。我们还可以通过传递带有键 `left`、`right`、`top` 和 `bottom` 的字典来为每边指定单独的边距。

我们还设置了行间距（又名行距）：它被指定为 `length` 值，我们使用 `em` 单位来指定相对于字体大小的行距：`1em` 相当于当前字体大小（默认为 `11pt`）。

最后，我们通过加入中心对齐和垂直对齐来对图像进行底部对齐。垂直和水平对齐可以与 `+` 运算符结合使用，以生成 2D 对齐。

标题

```
#set heading(numbering: "1.")
```

```
= Introduction  
#lorem(10)
```

```
== Background  
#lorem(12)
```

```
== Methods  
#lorem(15)
```

1. Introduction

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

1.1. Background

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

1.2. Methods

Lorem ipsum dolor sit amet, consectetur

Show规则

3.高级样式

创建标题和摘要

现在，让我们添加标题和摘要，我们将从标题开始。我们将其居中对齐，并通过将其括在 `*星号*` 中以将其加粗。

```
#align(center, text(17pt)[  
  *A fluid dynamic model  
  for glacier flow*  
])
```

由于我们是与我们的导师一起撰写这篇论文的，我们将添加我们自己和他们名字。

```
#grid(
  columns: (1fr, 1fr),
  align(center)[
    Therese Tungsten \
    Artos Institute \
    #link("mailto:tung@artos.edu")
  ],
  align(center)[
    Dr. John Doe \
    Artos Institute \
    #link("mailto:doe@artos.edu")
  ]
)
```

使用 `grid`，我们可以准确控制每列的大小以及哪些内容放到哪个单元格。`columns` 参数接受 **relative lengths** 或 **fractions** 的数组。在本例中，我们向它传递了两个相等的 `fractions`，告诉它将可用空间分成两个相等大小的列。然后，我们将两个内容块参数传递给 `grid` 函数。第一个是我们自己的信息，第二个是我们的导师的信息。我们再次使用 `align` 函数将列内的内容居中。网格采用任意数量的内容块参数来指定单元格。行是自动添加的，但也可以使用 `rows` 参数手动调整它们的大小。

会议希望摘要居中，且左右不对齐。值得注意的是，我们在 `align` 的内容参数中使用了一个新的 Set 规则来禁用摘要的对齐。即便它是在第一个 Set 规则之后指定的，这也不会影响文档的其余部分。这是因为内容块拥有 *局部作用域* 样式。内容块中设置的任何规则只会影响该内容块中的内容。

```
...

#align(center)[
  #set par(justify: false)
  *Abstract* \
  #lorem(80)
]
```

变量

另一个调整可以是将论文标题保存在一个变量中，这样我们就不必多次输入标题。我们可以使用 `let` 关键字来做到这一点：

```
#let title = [
  A fluid dynamic model
  for glacier flow
]

...

#set page(
  header: align(
    right + horizon,
```

```

      title
    ),
    ...
  )

  #align(center, text(17pt)[
    *#title*
  ])

  ...

```

添加列和标题

我们需要将 `columns` 函数应用于整个文档。我们可以使用“所有内容” Show 规则。

```

...

#show: rest => columns(2, rest)

= Introduction
#lorem(300)

= Related Work
#lorem(200)

```

唯一剩下的问题是，现在所有标题看起来都一模一样。“Motivation”和“Problem Statement”子小节应该是斜体标题，但现在，它们看起来与小节标题没有区别。我们可以通过在设置规则上使用 `where` 选择器来解决这个问题：这是一个 `method`，我们可以标题（和其他元素）上调用，允许我们按 `level` 过滤它们。我们可以用它来区分小节和子小节标题：

```

#show heading.where(
  level: 1
): it => block(width: 100%)[
  #set align(center)
  #set text(12pt, weight: "regular")
  #smallcaps(it.body)
]

#show heading.where(
  level: 2
): it => text(
  size: 11pt,
  weight: "regular",
  style: "italic",
  it.body + [.,
)

```


4.制作模板

其他

如何使用 VS Code 进行本地编辑？

1. 在 VS Code 中打开任意工作目录。
2. 在 VS Code 中安装 Typst LSP 和 Typst Preview 插件。前者负责语法高亮和错误检查，后者负责预览。
 - 也推荐下载 Typst Companion 插件，其提供了例如 `Ctrl + B` 进行加粗等便捷的快捷键。
 - 还可以下载 Typst Sync 和 Typst Sympy Calculator 插件，前者提供了本地包的云同步功能，后者提供了基于 Typst 语法的科学计算器功能。
3. 新建一个 `test.typ` 文件，写入内容 `# Hello World`。
4. 按下 `Ctrl + K V`，即可同步增量渲染与预览，还提供了光标双向定位功能。