

Typst でスライドを書く

Writing slides in Typst

2025-08-27

サイボウズ・ラボユース 夏合宿 @マホロバマインズ 三浦

Norimasa TAKANA

はじめに

目次

1. Typst とは	3	3.5 progress bar	16
1.1 Typst って何?	4	3.6 インジケータ	17
1.2 Typst のここがすごい!	5	3.7 作図	19
1.3 Typst のここが険しい	6	3.8 簡単なアニメーション	21
2. Touying	7	4. GUI のツールとの比較	22
2.1 Touying とは	8	4.1 プレゼンはどうするの?	23
2.2 色々できるぞ Touying + α	9	4.2 有利不利	24
3. Touying で出来ること	10		
3.1 コード表示	11		
3.2 矢印	12		
3.3 目次	13		
3.4 Box	14		

はじめに

自己紹介

名前: 高名 典雅 (Norimasa TAKANA)

所属: 筑波大学 理工情報生命学術院

Twitter: @_Alignof

雑な経歴:

- SecHack365 '19 思索駆動
- seccamp '20 Y-III (C コンパイラゼミ)
- Cybozu labyouth 12th (2022)
- seccamp/nextcamp '23 L3,N7,N11 講師
- 2024 年度未踏 IT 人材発掘・育成事業 (スーパークリエイター)
- seccamp connect '25 講師



図 1: 最近の画像

1. Typst とは

Typst とは

Typst って何？

Typst¹ (/taɪpst/) は近年登場した OSS の組版ツール。

CUI のバイナリだけでなく，（僕は試したことはないけど）オンラインでも書ける。

<https://typst.app/play/>

● これを使うと L^AT_EX に戻れなくなるため注意が必要。



¹<https://typst.app/>

Typst とは

Typst のここがすごい！

ここがすごいぞ Typst

- Rust 製
- バイナリー発で環境構築が楽.
- コンパイルが鬼のように速い.
- プログラマブルな構文.
- そこそこまともなエコシステム.
- **コンパイルを 2 回しなくて済む.**

Typst とは

Typst のここが険しい

正直そんなに無い。

ここが険しいぞ Typst

- CJK で微妙に詰めが甘い部分がある。
 - ▶ 例えばソースコードで改行すると本文にもスペースが入ってしまう。
- 型がかなりガバいのでそこそこ凝ったことをしようとすると険しい。

開発は活発なので今後に期待。

2. Touying

Touying

Touying とは

Typst を使ってスライドを生成するライブラリ。

各 subsection に対応する部分が自動でスライドとして描画される。

```
slide.typ
```

```
1 == Typst のここが険しい
2 正直そんなに無い。
3 #mybox(title: "ここが険しいぞ Typst", color: red)[
4   - CJK で微妙に詰めが甘い部分がある。
5   - 例えばソースコードで改行すると本文にもスペースが入ってしまう。
6   - 型がかなりガバいのでそこそこ凝ったことをしようとすると険しい。
7 ]
8
9 開発は活発なので今後に期待。
```

書き味は普通の文書と変わらない。

Touying

色々できるぞ Touying + α

- 目次
- Box
- progress bar
- 簡単なアニメーション
- 作図
- コード表示

3. Touying で出来ること

Touying で出来ること

コード表示

zebraw を使うことでいい感じにコード表示ができる（ライブラリ使わなくても結構標準機能でもいける）。

```
main.rs
```

```
1 fn main() {  
2     println!("Hello, world!");  
    > ↑にハイライトとコメント  
3 }
```

Touying で出来ること

矢印

pinit というライブラリを使うと、**ハイライト**したり、

その部分に矢印をつけたり出来る。

Touying で出来ること

目次

↓これ

1. Typst とは	3	3.4 Box	14
1.1 Typst って何?	4	3.5 progress bar	16
1.2 Typst のここがすごい!	5	3.6 インジケータ	17
1.3 Typst のここが険しい	6	3.7 作図	19
2. Touying	7	3.8 簡単なアニメーション	21
2.1 Touying とは	8	4. GUI のツールとの比較	22
2.2 色々できるぞ Touying + α	9	4.1 プレゼンはどうするの?	23
3. Touying で出来ること	10	4.2 有利不利	24
3.1 コード表示	11		
3.2 矢印	12		
3.3 目次	13		

Touying で出来ること

Box

タイトル

内容

用語: 用語

用語の説明

強調の内容

定義 3.1 (定義の例)

定義の内容

⚠ 注意

これらには **theorion** が必要です。

Touying で出来ること

progress bar

↑ ここにあるコイツ。

style.typ

```
1 let header(self) = {
2   set std.align(top)
3   block(width: 100%, height: 74pt, {
4     grid(
5       rows: (auto, auto),
6       // ...
7       if self.store.progress-bar {
8         components.progress-bar(height: 4pt, self.colors.secondary,
self.colors.dark_primary)
9       },
10    )
11  })
12 }
```

Touying で出来ること

インジケータ

← 左のやつ。現在の章をハイライトで表示する。

```
style.typ
```

```
1 let side-bar = place(  
2   top + left,  
3   dy: 74pt,  
4   block(  
5     width: 25pt,  
6     height: 85%,  
7     fill: self.colors.cyan.lighten(80%),  
8  
9     std.align(  
10      center + horizon,  
11      grid(  
12        rows: (auto, auto, auto),  
13        row-gutter: .5em,
```

Touying で出来ること

インジケータ

```
14     context for chapter in array.range(counter(heading).final().at(0)) {
15         circle(
16             radius: 8pt,
17             fill: if chapter + 1 == counter(heading).get().at(0) {
18                 self.colors.secondary
19             } else {
20                 self.colors.dark_primary
21             }
22         )
23     }
24 )
25 )
26 )
27 )
```

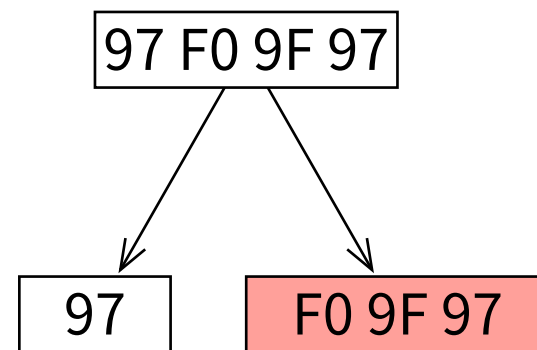
Touying で出来ること

作図

どっかのゼミ資料で使った例.

⚠ 注意

原則，目的の byte 以外も**有効な命令である必要がある**。
ただし，ジャンプ命令の間はその限りではない。



Touying で出来ること

作図

用語: stack overflow

確保されたメモリの範囲を超えて書き込みができる脆弱性. これによりユーザが**スタックに任意のデータを書き込む**ことが可能になる.

図 2 の例では, `int array[0]` から大きさ 3 を超えて値を書き込むことで灰色の部分のスタック領域を上書きしている.

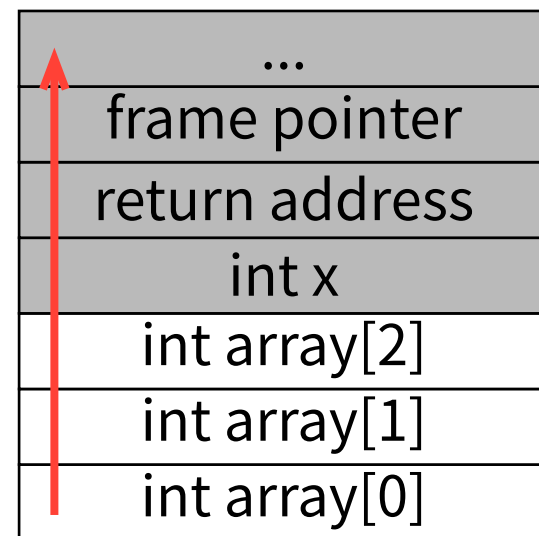


図 2: stack overflow の例

Touying で出来ること

簡単なアニメーション

文字を などの機能がある。



Touying で出来ること

簡単なアニメーション

文字を出現させるなどの機能がある。

内容をこれからに変えることもできる。

- 1 == 簡単なアニメーション
- 2 文字を`#pause` 出現させる `#meanwhile` などの機能がある。
- 3
- 4 `#pause`
- 5
- 6 内容を`#alternatives`[これから][`#highlight`[これ]]に変えることもできる。

Touying で出来ること

簡単なアニメーション

文字を出現させるなどの機能がある。

内容を **これ** に変えることもできる。

- 1 == 簡単なアニメーション
- 2 文字を `#pause` 出現させる `#meanwhile` などの機能がある。
- 3
- 4 `#pause`
- 5
- 6 内容を `#alternatives`[これから][`#highlight`[これ]] に変えることもできる。

4. GUI のツールとの比較

GUI のツールとの比較

プレゼンはどうするの？

Typst + touying で出来るのはスライドの pdf を生成できるだけ。

→ **プレゼン**はどうするか？

pympress というツールを使うと pdf を使っていい感じのプレゼンができる。

- レーザーポインタ
- ペンでの書き込み
- 発表者モード (2 画面表示)
- スピーカーノート
- 動画再生¹

¹私はまだ試したこと無いです。

GUI のツールとの比較

有利不利

良いところ

- 好きなエディタで書ける。
- ファイル分割もバージョン管理も自由。
- 装飾をライブラリが勝手にやってくれる。
- 電車の中でも書ける。

悪いところ

- 写真をベタベタ貼るタイプのスライドは Google slides のほうが良いかも。
- GitHub に push し忘れると**すべてが終わる**。