

The most flexible way to create LCARS Interfaces using Typst.

2038503437590223-4398072549052834-3540985420934750392843509872384-945837430859874345-4380543098423
3409554098354334-6045938540945683-4350340985624984575167853218742-897843214842488945-6578964231564
1451657445248941-4587564561234854-5468756486442468515246847915154-457854687431235357-9876514857324
4579854578975213-7478941567354875-5798416575645443213576531457657-597851277421543487-5498445154872

7-172590

1-725900

7-259008

2-590089

5-900897

9-008978

0-897835

8-978355

0-089783

9-783556

7-835568

8-355683

3-556830

5-568305

5-683058

6-830585

8-305858

Single Page Layout Support

Demonstrating versatility for single page Documents or title pages.

0-585813

5-858139

8-581395

5-813953

8-139539

1-395398

3-953987

9-539876

5-398765

9-876514

8-765142

3-987651

7-651422

6-514221

5-142212

1-422128

4-221281

The `single-page-layout()` function allows for easy creation of title pages or single page documents with a special flair.

0-617884

TiefCARS Design System

8-312530

A somewhat complete page layouting system for Typst, imitating the LCARS User Interface (roughly)

– Lena Tauchner

1-281122

Introduction

0-617884

TiefCARS is a highly dynamic, customizable layouting and drawing library to easily generate elements similar to those seen in the Library Computer Access/Retrieval System (LCARS).

2-811220

Aside from its flexibility, TiefCARS is a highly dynamic library. It offers multiple simple ways to generate an initial document, responds to dynamic page sizes and allows theming.

8-112200

With its modular architecture, using pseudo-random number generators for colors and text, the output is both predictable on a per-compile basis and modular.

Reasoning for Development

0-617884

When creating props, one either designs their own LCARS interface via some sort of manual editor (Inkscape, Gimp, ...), which costs time and sometimes results in less than satisfactory outcomes, or they may download preset templates. Either way is not flexible for props that are either custom or in active use (and changing).

1-122008

That's why a library for typst, allowing creative and flexible typesetting, is necessary for any prop designer. Using this library, one can create PADD content, screen elements, ..., without the hassle of placing individual items on a screen.

1-220089

Elements

0-617884

TiefCars ships with a variety of elements, ready to be used. Among the buttons and sweeps, there are also more complex examples like sidebars.

2-200890

Button

The button is a material part of any UI/UX Design, and thus needs to be implemented in TiefCARS. A button is inserted using `#buttons.button(variant)[Content]`

Content

0-089093

A button also allows custom insets: `#buttons.button(3, inset: (left: 12pt, top: 2pt, right: 4pt, bottom: 7pt))[Content]`

Content

0-890931

Sidebar Button

The sidebar of an LCARS interface does not use the usual buttons, instead relying on rectangular buttons with a text in the bottom right. This is displayable via the `#sidebar.sidebar-button(40pt, 100pt)` command:

9-093121

1-211312

0-931211

Sidebar buttons also allow for further customization like the color and the text displayed: `#sidebar.sidebar-button(40pt, 100pt, color: red, elem-text: "TiefCARS <3")`

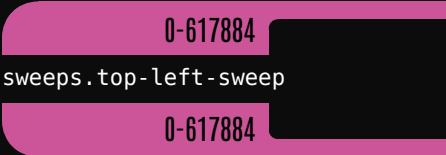
9-312113

TiefCARS <3

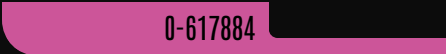
Sweeps

There are four variants of sweeps available, all functioning in the same way:

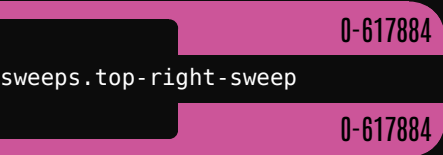
sweeps.bottom-left-sweep



sweeps.top-left-sweep



sweeps.bottom-right-sweep

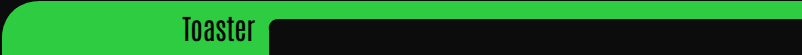


sweeps.top-right-sweep

Each sweep requires three parameters: height, width and color. #sweeps.bottom-left-sweep(20pt, 100pt, theme.accent.a-5)

A sweep may also include the following parameters: sweep-text, sweep-width.

#sweeps.bottom-left-sweep(20pt, 100pt, green, sweep-text: "Toaster", sweep-width: 200pt)



#sweeps.bottom-left-sweep(20pt, 100pt, blue, sweep-text: "Toaster", sweep-width: 20pt)



Other elements

Less relevant/mostly internal elements are: sidebar-bar, sidebar

These are not discussed in this document.

Layouting

Single Page Layout

See Page 1; creates a layout with the iconic horizontal bar. Allows adding text, subtext, and a limited body.

```
#single-page-layout(  
  title: [Your Title],  
  subtitle-text: [  
    Text for your subtitle  
  ]  
,  
  top-height: 5,  
)[  
  Main Body  
]
```

Multi Page Layout

The current layout. Allows multipage-input, dynamically selecting headings and creating a sweep to it.

```
#multi-page-layout(  
  odd-even: false,  
  sidebar-width: 50pt,  
)[  
  Content  
]
```

Or, if only showing one multi page layout in a document:

```
#show: multi-page-layout.with(  
  title: "My amazing document"  
)
```

Content

1-131265
1-312657
3-126570
1-265704
2-657041
6-570417
5-704172
7-041725
0-617884
0-417252
4-172529
1-725292
7-252927
2-529278
5-292780
2-927807
9-278079
2-780795

Flexibility

0-617884

TiefCARS is built for flexibility - different page formats are, of course, supported.

8-079534

You can build beautiful post cards or business cards with TiefCARS!

0-795346

Keep in mind that the layouting is configurable - make it just right for your project using optional parameters.

7-953462

9-534627

Contribution guidelines

There's not really much to consider, other than the request to keep a folder structure. Components go into components/, so on.

Theming can always be expanded and may be a simple entry point, that'd be nice :)

0-617884

3-462772

4-627721

6-277211

2-772115

7-721153

7-211531

2-115316

1-153169

1-531699

5-316999

3-169993

1-699938

6-999383

9-993831

9-938312

9-383125

3-831253