typst

Compose papers faster.

Martin Haug ▪ Laurenz Mädje

20. JULI **2023**

Typst hilft Studierenden und Wissenschaftlern **technische Dokumente schneller** zu schreiben

Louis Vignoli, PhD, Industrial quantum scientist

„I am over the moon with Typst and converted as many of my coworkers as possible.“

Christoph

„This is the most pleasant typesetting software that I have ever used by a million miles.“

Prof. Christopher Métrailler, HES-SO Valais

„Typst is a perfect tool for me and my use cases“

Mordrag

„I love how simple Typst seems, while retaining the features one would expect.“

Donald Knuth erfand **TEX** im Jahr **1978** um den zweiten Band seines Buchs „The Art of Computer Programming“ zu setzen

Makro-basierte Schriftsatzsprache

Computer Modern: Schriftfamilie

DVI: Dokumentausgabeformat

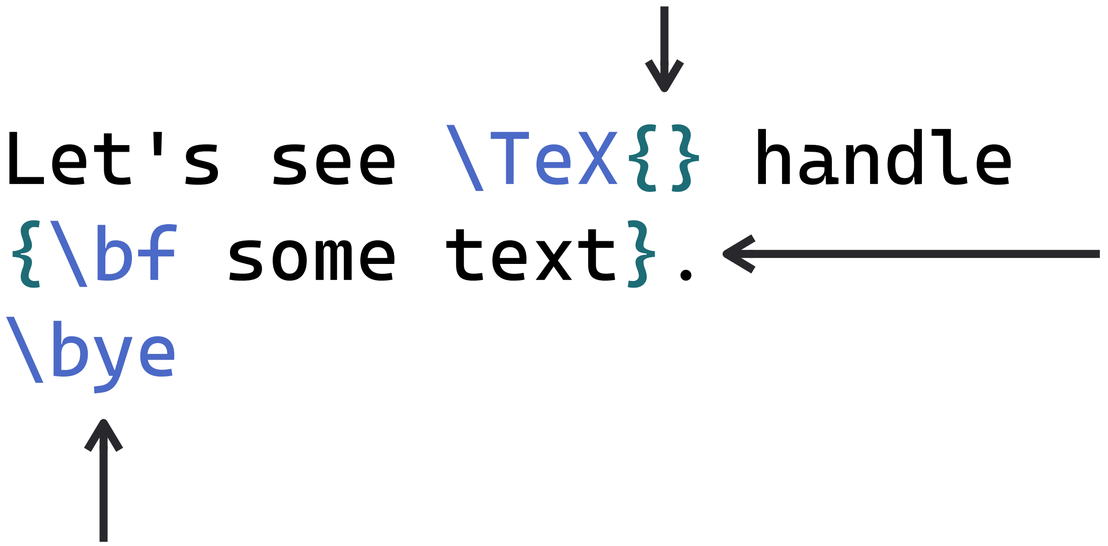
Ein Zeilenumbruchs-Algorithmus

METAFONT: Schrift-Technologie

WEB: Programmiersprache

FÜR TEX ENTWICKELTE TECHNOLOGIEN

Leere Gruppe zur Trennung von
Makro und Leerzeichen

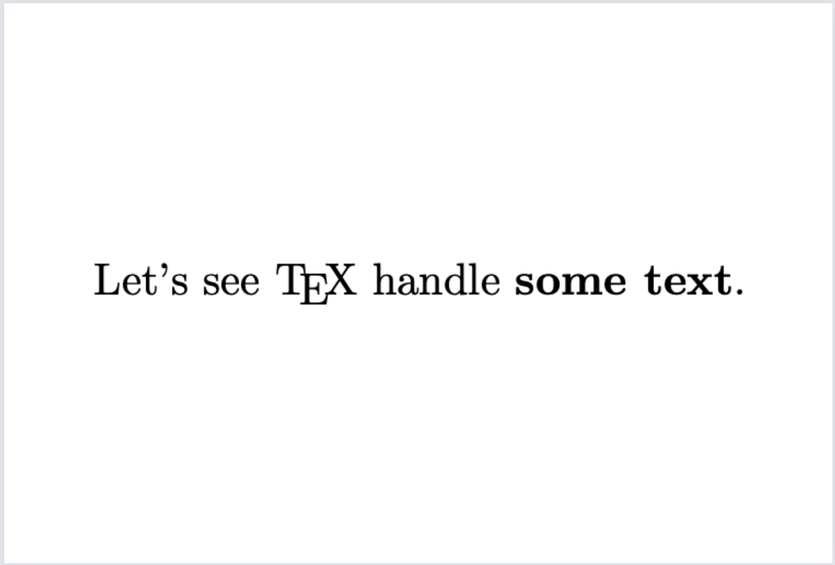


```
Let's see \TeX{} handle
{\bf some text}.
\bye
```

Gruppe mit geschweiften Klammern
scoped das \bf-Makro

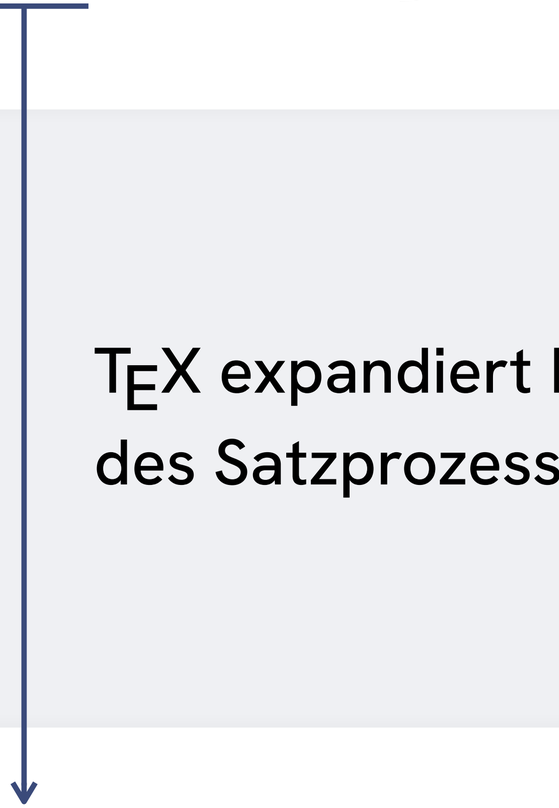
Dieses Makro sorgt dafür, dass die
letzte Seite ausgegeben wird.

```
Let's see \TeX{} handle
{\bf some text}.
\bye
```



```
{\bf writing something}
```

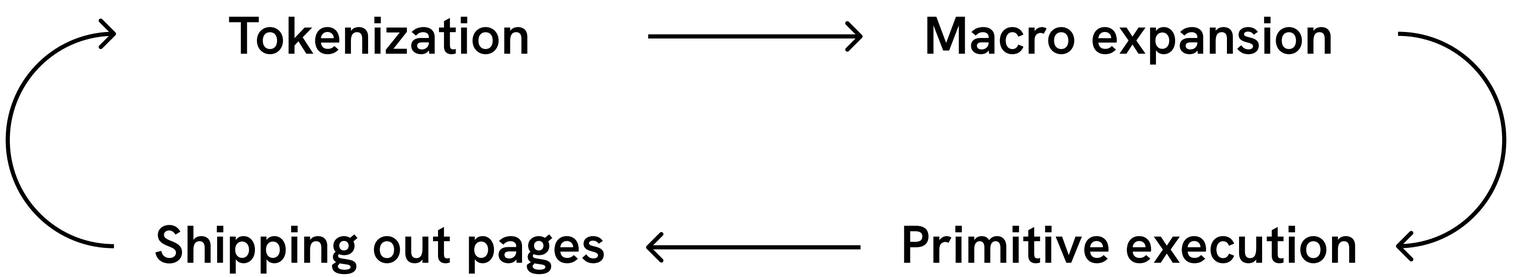
Makro aus plain.tex



TEX expandiert Macros während des Satzprozesses

```
{\fam\bffam\tenbf writing …}
```

\radical \moveleft \message \lccode \lastbox \pagegoal
athaccent \noalign \omit \mark \leftskip \kern
\multiply \maxdeadcycles \mathop \jobname \meani
\medmuskip \everycr \futurelet \fontdimen \delcode \immediate \clead
ness \leqno \displaywidowpenalty \everyjob \above \errhelp \globa
ders
uage \abovedisplayshortskip \begingroup \crcr \floatingpena
\batchmode \inputlineno \csname \dump \endlinechar
gafter \everypar \holdinginsert
tcode \hfuzz \ifeof \ifcase \ifx \badness \aftergroup
\hfilneg \belowdisplayskip \chardef \firstmark \exhyphe
enout \divide \copy \adjdemerits
r \baselineskip \botmark \hss \prevgraf \expandafter
\abovedisplayshortskip \countdef \deadcycles \dump



TEX ließt den Quellcode von oben nach unten

`*.aux`

Vorrausschauen mit Hilfsdatei und mehrfachem Kompilieren

`\catcode`

Dynamische Redefinition der Syntax möglich

TEX ist sehr low-level und arbeitet mit formatierungsgestützten Atomen

„Damit ein Dokument einfach zu lesen ist, muss die visuelle Struktur die logische Struktur widerspiegeln“

— Leslie Lamport, Erfinder von LATEX, 1985

LATEX ist eine Makro-Sammlung für TEX, die Struktur in den Vordergrund stellt

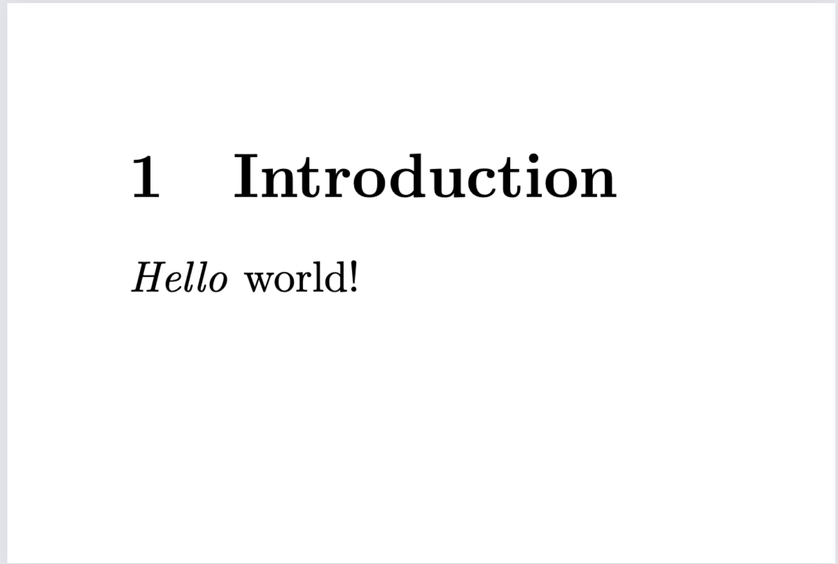
```
\documentclass{article}

\begin{document}

\section{Introduction}
\emph{Hello} world!

\end{document}
```

`\documentclass` legt Art des Dokuments fest

Emphasis-Makro anstatt kursive Schrift zu setzen

LaTeX-eigene Start- und End-Makros

```latex
\documentclass{article}

\begin{document}

\section{Introduction}
\emph{Hello} world!


\end{document}
```



Große Akzeptanz bei Journals

Fokus auf Struktur statt Format erlaubt einfache Neuformatierung

6.380

Verfügbare LaTeX-Pakete auf CTAN

Vorteile von LaTeX

Verleiht Street Cred

Läuft auf IBM 4300 Mainframe

245.905

Fragen zu (LA)TeX auf tex.stackexchange.com

Ausschließlich
für PDFs/Print

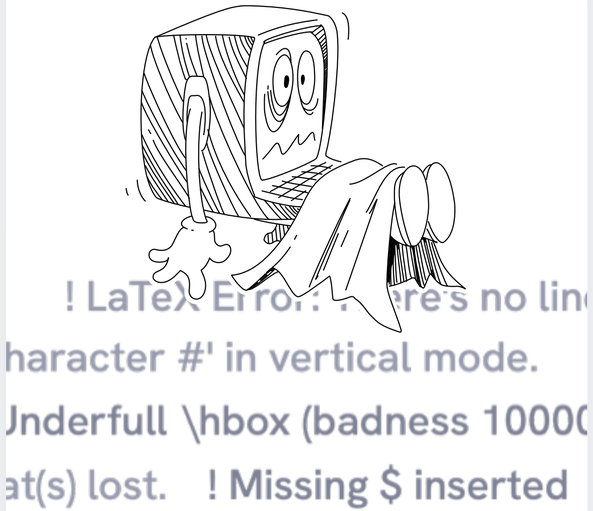
!

Stil-Anpassungen erfordern Expertenwissen und sind langwierig

`graphicx`
`tabularx`
`babel`
`xcolor`
`amsmath`

Viele grundlegende Aufgaben erfordern Pakete

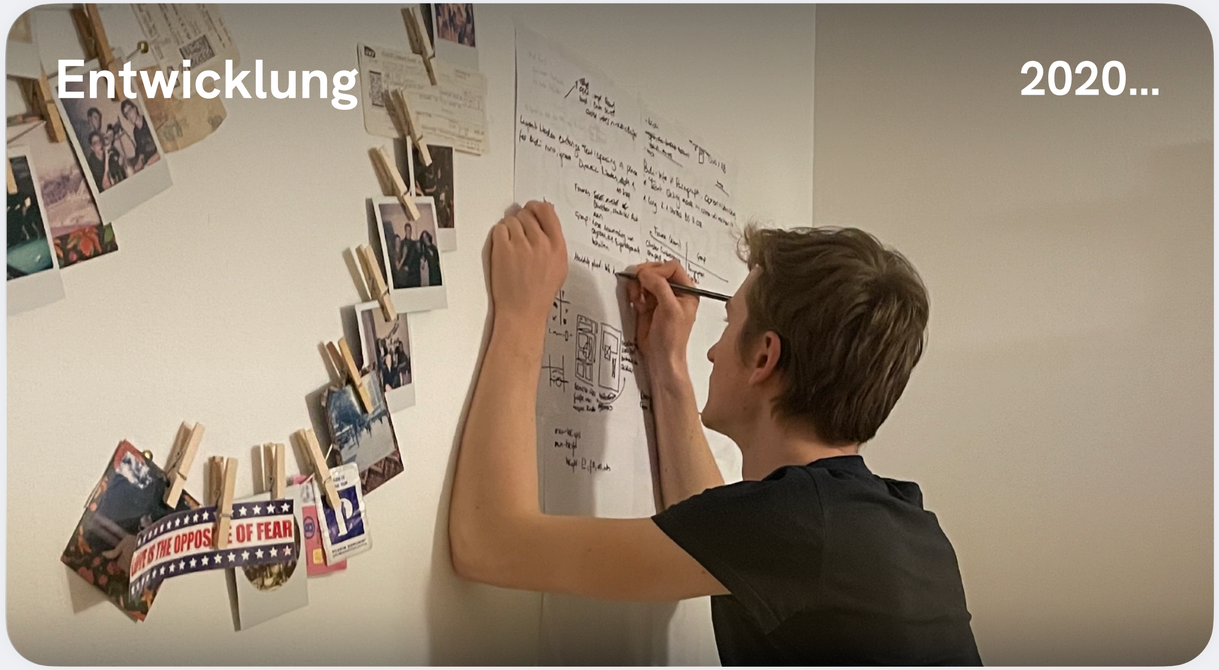
Nachteile von LaTeX



Wenig hilfreiche Fehlermeldungen

typst


TU Berlin: Idee
2019
TECHNISCHE UNIVERSITÄT


Entwicklung
2020...


Berliner Startup Stipendium
Jan-Jun 2023

Demo

Typst File Edit View Text Structure Layout Code Help

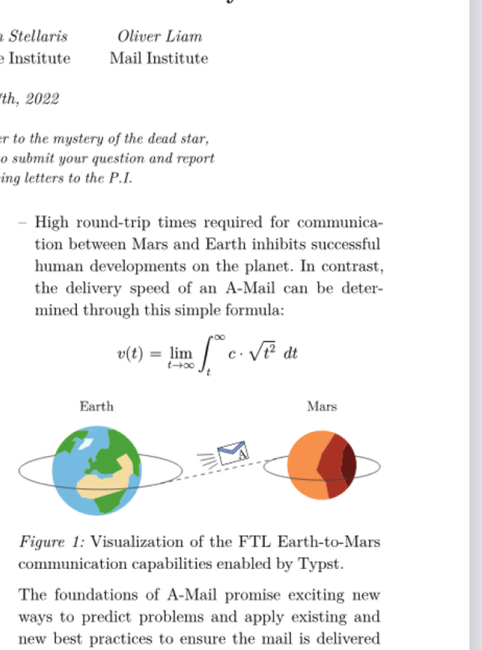
Johanna's Typst › Space Mail

Latin Modern Roman | B I | H | Σ | @ | 100% | Share

```
= Introduction
Our concept suggests three ways that A-Mail can be best utilized.

- First is to reduce the probability of the failure of a space mission. This
  problem is known as the Mars problem and suggest problems with human
  communication.

- High round-trip times required for communication between Mars and Earth inhibits
  successful human developments on the planet. In contrast, the delivery speed of
  an A-Mail can be determined through this simple formula:
  $[ v(t) = lim(t -> inf) integrate(t..inf) c * sqrt(t^2) dt ]$

#figure(
  image("a-mail.svg"),
  caption: [
    Visualization of the FTL Earth-Mars communication
    capabilities enabled by Typst.
  ],
)

The foundations of A-Mail promise exciting new ways to predict problems and apply
existing and new best practices to ensure the mail is delivered without any
issues. We call this extension AI-Mail. AI-Mail is a new concept designed and
```

Egon Johanna Nicole

typst



Towards Faster Interstellar Mail Delivery

Johanna Swift, Delivery Institute · Egon Stellaris, Space Institute · Oliver Liam, Mail Institute

May 17th, 2022

Until there is a definitive answer to the mystery of the dead star, please use the old postal system to submit your question and report the location of missing letters to the P.I.

Abstract

Recent advances in space-based document processing have enabled faster mail delivery between different planets of a solar system. Given the time it takes for a message to be transmitted from one planet to the next, its estimated that even a one-way trip to a distant destination could take up to one year. During these periods of interplanetary mail delivery there is a slight possibility of mail being lost in transit. This issue is considered so serious that space management employs P.I. agents to track down and retrieve lost mail. We propose A-Mail, a new anti-matter based approach that can ensure that mail loss occurring during interplanetary transit is unobservable and therefore potentially undetectable. Going even further, we extend A-Mail to predict problems and apply existing and new best practices to ensure the mail is delivered without any issues. We call this extension AI-Mail. Integrating quantum computing could open up even more possible applications, ranging from inter-galactic delivery to mind reading.

Reference: Johanna Swift, Egon Stellaris, Oliver Liam. *Towards Faster Interstellar Mail Delivery*. https://doi.org/10.7891/1209348510

– High round-trip times required for communication between Mars and Earth inhibits successful human developments on the planet. In contrast, the delivery speed of an A-Mail can be determined through this simple formula:

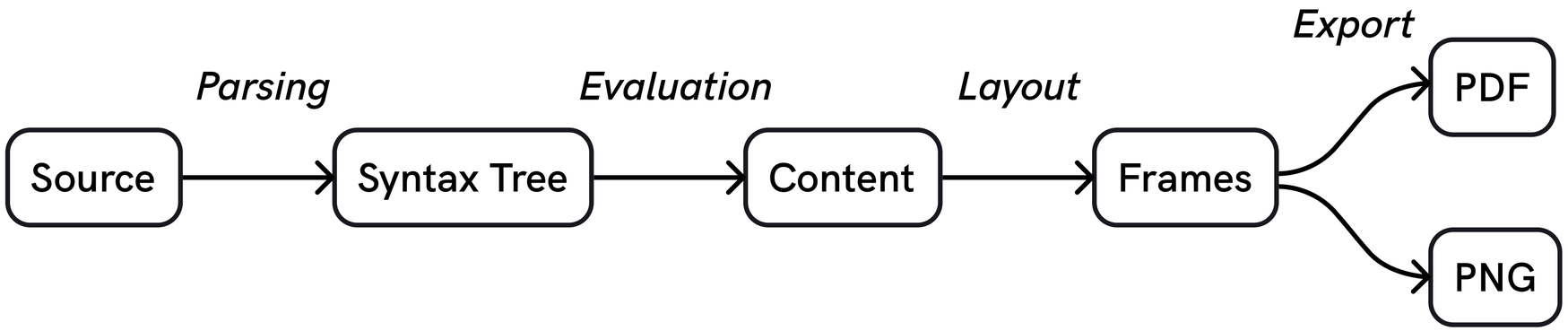
$$v(t) = \lim_{t \to \infty} \int_t^{\infty} c \cdot \sqrt{t^2}\, dt$$

Earth Mars

Figure 1: Visualization of the FTL Earth-to-Mars communication capabilities enabled by Typst.

The foundations of A-Mail promise exciting new ways to predict problems and apply existing and new best practices to ensure the mail is delivered

Compiler-Architektur



Introspection

Query-System

Inkrementalität

Schnelle Rekompilation

Recent Developments Within Glaciers

July 20, 2023

Laurenz
laurenz.maedje@typst.app

Martin
martin.haug@typst.app

Outline

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit.

1. Introduction

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

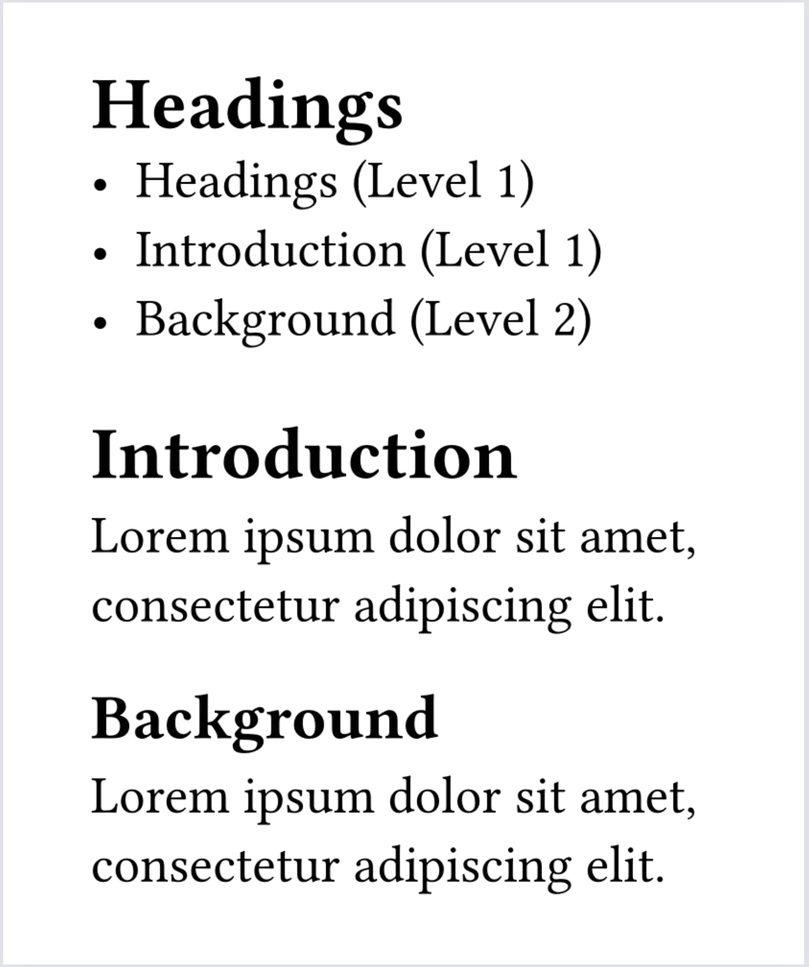
Figure 1: A glacier which might not exist for much longer.

2. Related Work

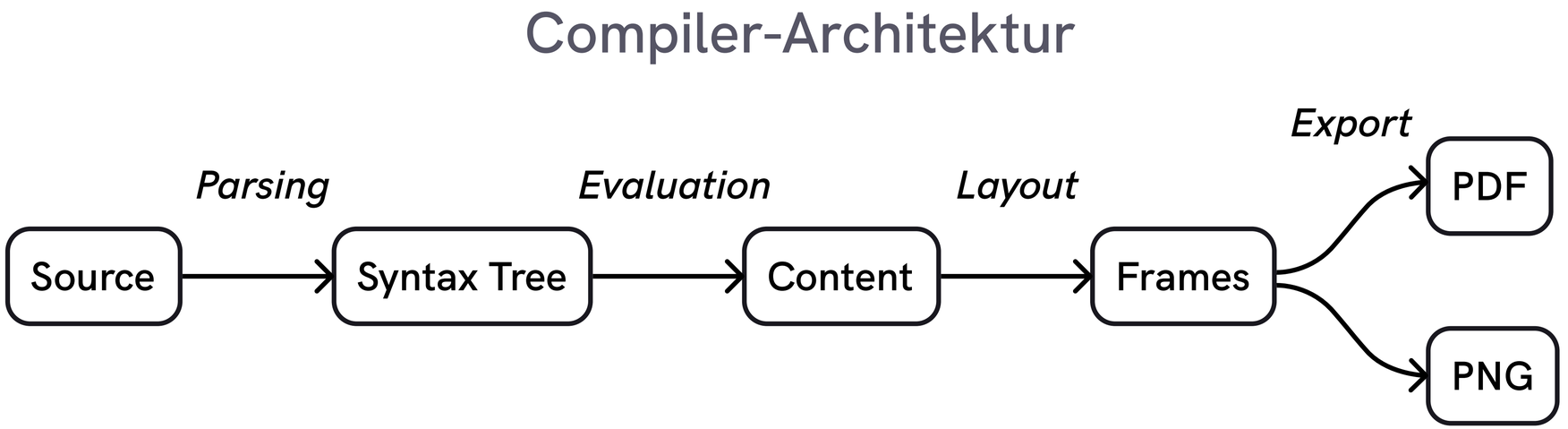
In Figure 1, we can clearly observe: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum

Introspection

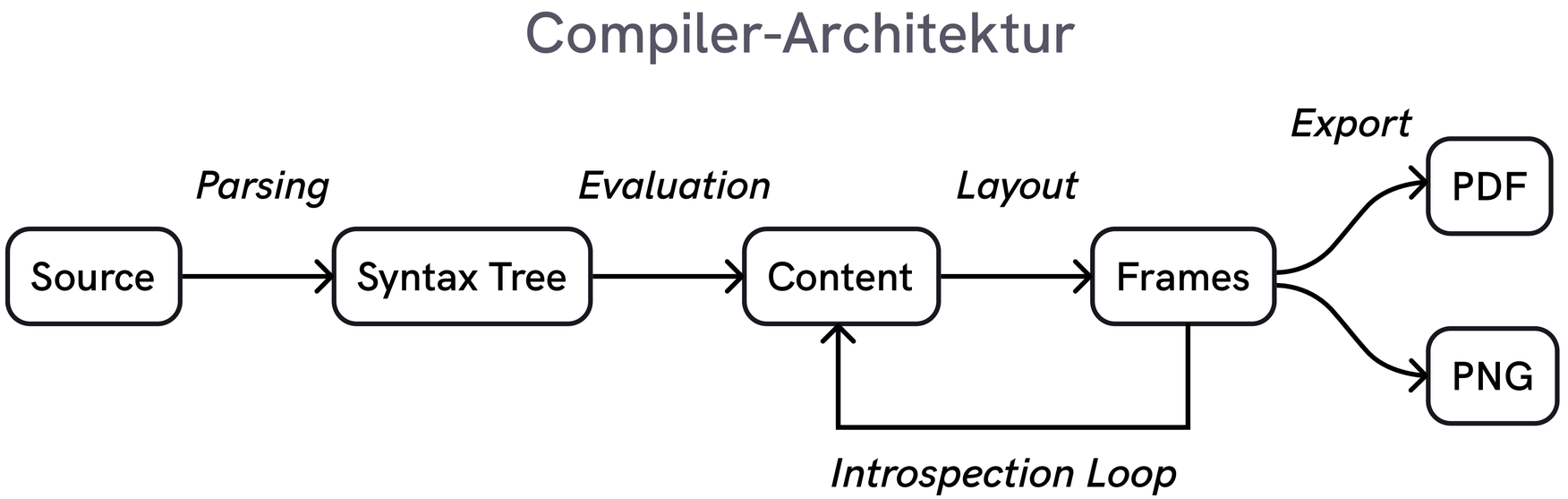
```
= Headings
#locate(loc => {
  let headings = query(heading, loc)
  for elem in headings [
    - #elem.body (Level #elem.level)
  ]
})

= Introduction
#lorem(8)

== Background
#lorem(8)
```



Introspection

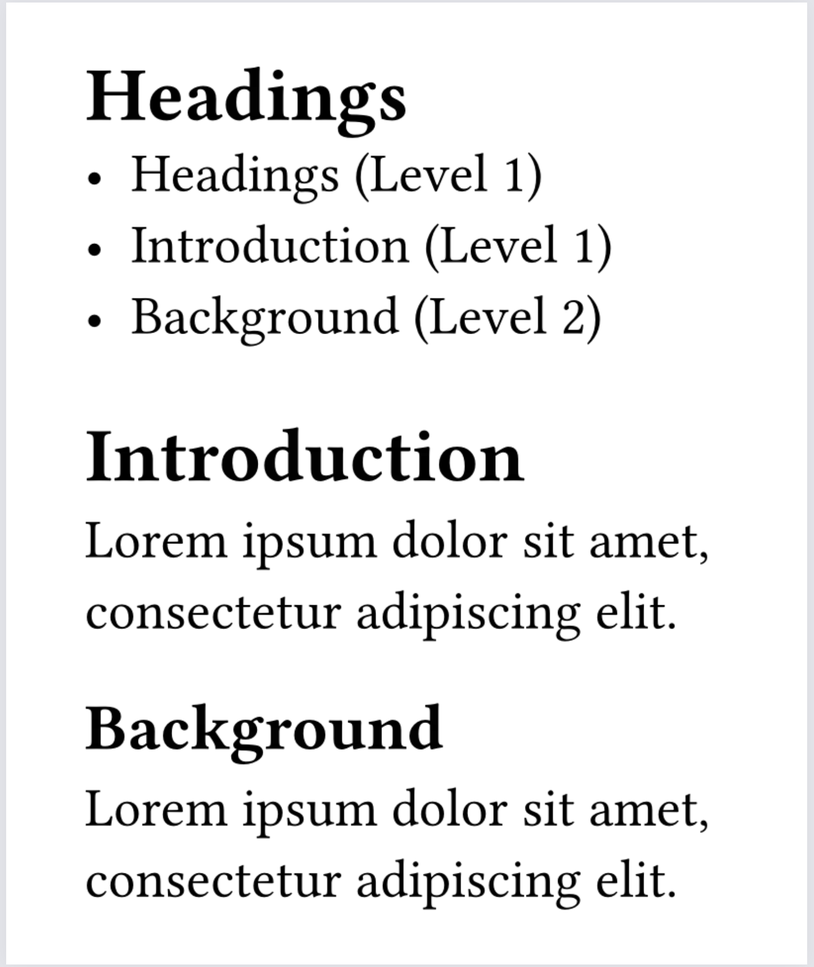


Introspection



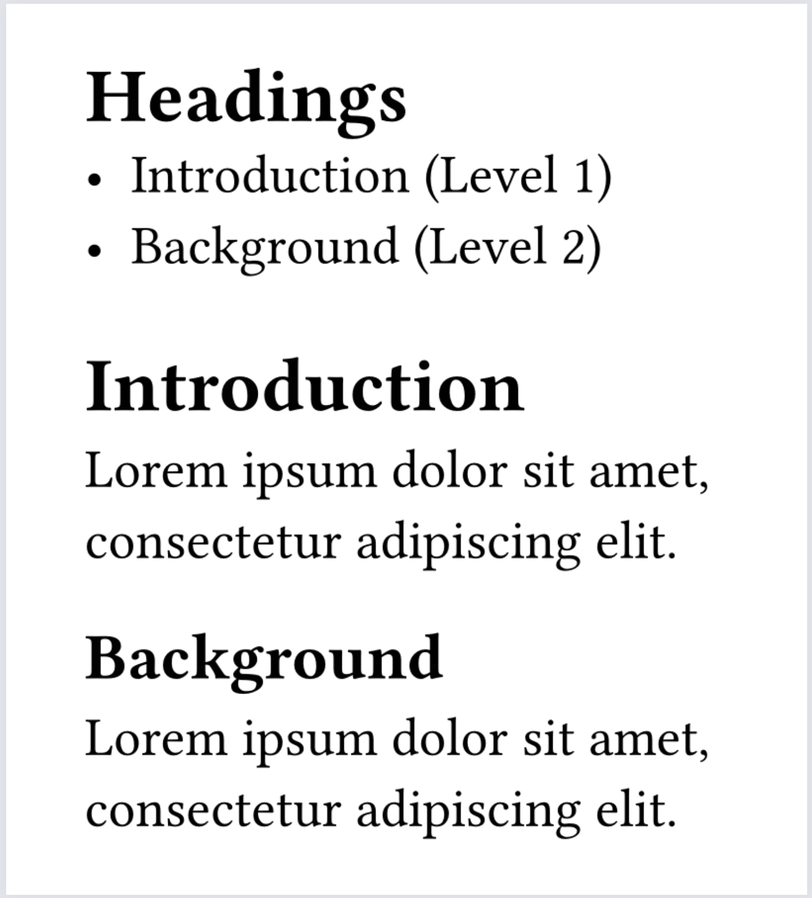
Introspection

```
= Headings
#locate(loc => {
  let headings = query(heading, loc)
  for elem in headings [
    - #elem.body (Level #elem.level)
  ]
})

= Introduction
#lorem(8)

== Background
#lorem(8)
```



Introspection

```
= Headings
#locate(loc => {
  let headings = query(
    selector(heading).after(loc),
    loc,
  )
  for elem in headings [
    - #elem.body (Level #elem.level)
  ]
})

= Introduction
#lorem(8)

== Background
#lorem(8)
```



Introspection

```
#locate(loc => {
  let headings = query(heading, loc)
  [= Heading] * (headings.len() + 1)
})
```


Heading

Heading

Heading

Heading

Heading

1

Inkrementalität

alpha.calc

```
1 + 2 + beta.calc
```

beta.calc

```
gamma.calc + 4
```

gamma.calc

```
8 + 1
```

calc.rs

```rust
fn evaluate(script: &str, files: &Files) -> i32 {
    script
        .split('+')
        .map(str::trim)
        .map(|item| match item.parse::<i32>() {
            Ok(num) => num,
            Err(_) => evaluate(&files.get(item), files),
        })
        .sum()
}
```

Inkrementalität

alpha.calc

```
1 + 2 + beta.calc
```

beta.calc

```
gamma.calc + 4
```

gamma.calc

```
8 + 1
```

calc.rs

```rust
#[comemo::memoize]
fn evaluate(script: &str, files: Tracked<Files>) -> i32 {
    script
        .split('+')
        .map(str::trim)
        .map(|item| match item.parse::<i32>() {
            Ok(num) => num,
            Err(_) => evaluate(&files.get(item), files),
        })
        .sum()
}
```

Inkrementalität

alpha.calc

```
1 + 2 + beta.calc
```

beta.calc

```
gamma.calc + 4
```

gamma.calc

```
8 + 1
```

calc.rs

```rust
#[comemo::memoize]
fn evaluate(script: &str, files: Tracked<Files>) -> i32 {
    script
        .split('+')
        .map(str::trim)
        .map(|item| match item.parse::<i32>() {
            Ok(num) => num,
            Err(_) => evaluate(&files.get(item), files),
        })
        .sum()
}

#[comemo::track]
impl Files {
    fn get(&self, path: &str) -> String {
        /* load and cache file */
    }
}
```

Inkrementalität

```
= Headings
#locate(loc => {
  let headings = query(heading, loc)
  for elem in headings [
    - #elem.body (Level #elem.level)
  ]
})

= Introduction
#lorem(8)

== Background
#lorem(8)
```

Headings

- Headings (Level 1)
- Introduction (Level 1)
- Background (Level 2)

Introduction

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

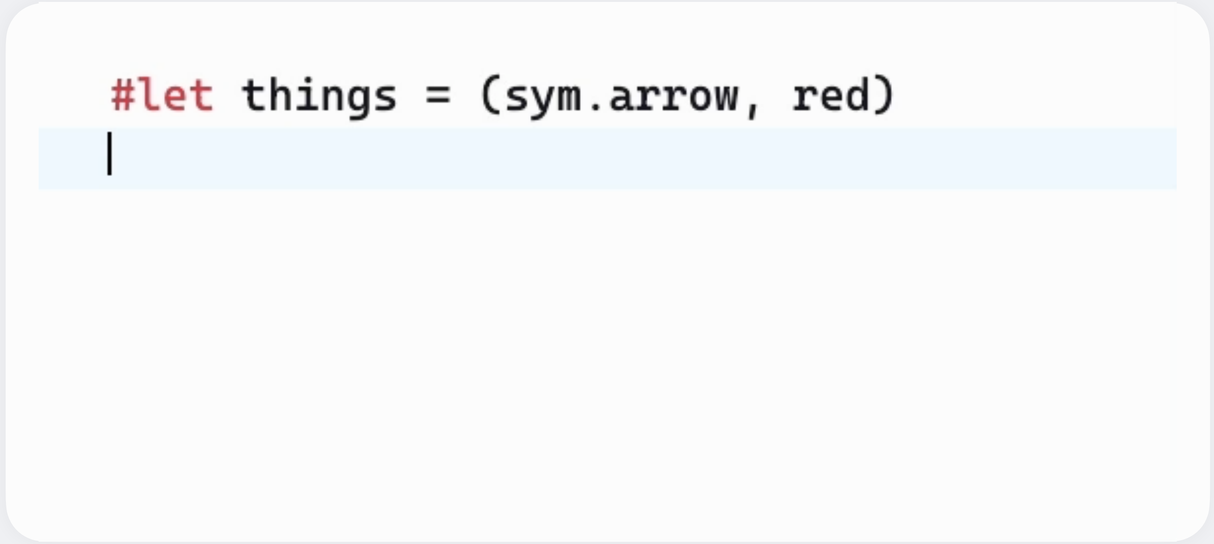
Background

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Inkrementalität

eval/mod.rs

```rust
#[comemo::memoize]
fn eval(
    world: Tracked<dyn World + '_>,
    route: Tracked<Route>,
    tracer: TrackedMut<Tracer>,
    source: &Source,
) -> SourceResult<Module> {
    ...
}
```



Auto-Vervollständigung

```
#let things = (sym.arrow, red)
```

Auto-Vervollständigung

Hover Tooltips

```
#for val in range(15, step: 4) {
  str(val) + " and "
}
```

```
#let things = (sym.arrow, red)
#things
```

1. Relevante Ausdrücke markieren
2. Dokument vollständig kompilieren, bei markierten Ausdrücken Wert speichern
3. Auto-Vervollständigungen basierend auf beobachteten Werten bestimmen

Relevante Ausdrücke

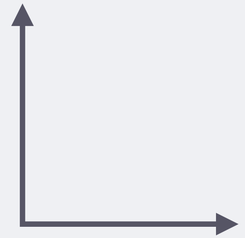
```
#for val in range(15, step: 4) {
  str(val) + " and "
}
```

Web App

Im Browser

UI
TypeScript
React
Sass

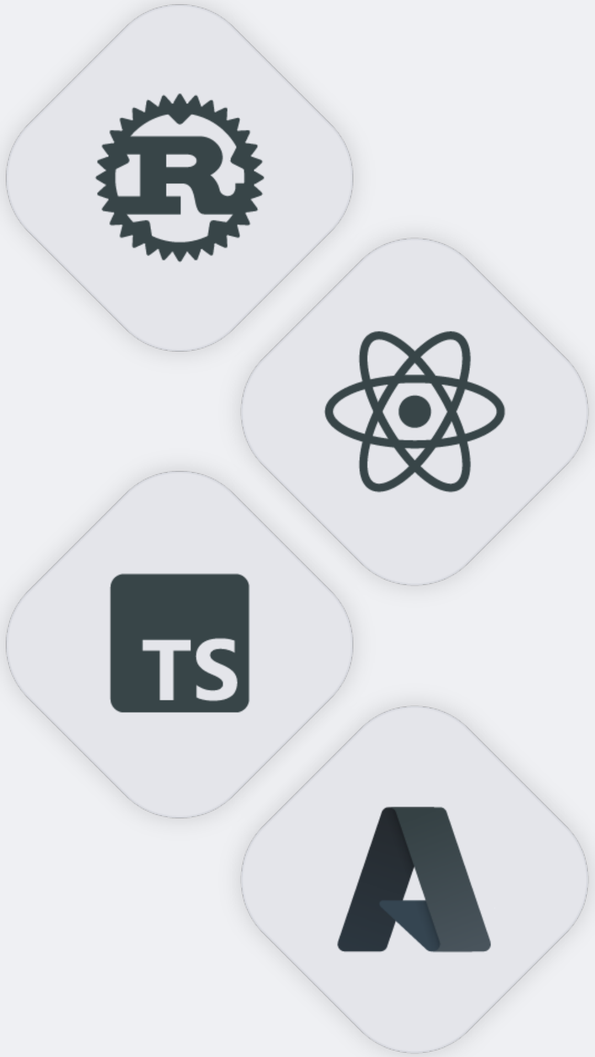
Compiler
WebAssembly
Web Worker



API
TypeScript
Serverless
PostgreSQL

Collaboration
TypeScript
Docker

Microsoft Azure

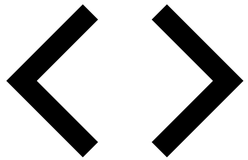


Lokaler Compiler

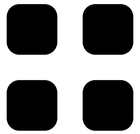
```
$ typst watch hello.typ
watching hello.typ
writing to hello.pdf

[18:35:33] compiled successfully
```

```
$ typst watch hello.typ
watching hello.typ
writing to hello.pdf

[18:35:33] compiled with errors

error: expected length, found color
   ┌─ /hello.typ:1:16
   │
1  │ #set text(size: red)
   │                 ^^^
```

HTML-Export

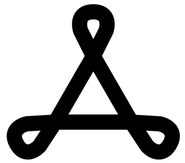


Export der selben Datei als HTML, PDF und ePUB. Veröffentlichung auf typst.site



Flexibleres Layout

Roadmap



Barrierefreie PDFs

Kommentare Review-Tools

Rechtschreib-korrektur

Versionierung Git-Integration

Registriere dich für die Typst Web App

Alle, die diesen Link benutzen, erhalten ein exklusives TU Berlin-Theme

https://typst.app/signup?code=freitagsrunde

Typst Compiler auf GitHub

typst/typst

Kommandozeilen-Compiler im Releases-Tab verfügbar, PRs herzlich willkommen!

typst

Community-Tools

VS Code Plugins (LSP, Preview), qjcg/awesome-typst, typst.cn, …

Folge uns auf Twitter

@typstapp

Trete unserer Community bei

3569 Mitglieder

typst

Compose papers faster.

Martin Haug
Laurenz Mädje