

Guide to Typst

13.11.2023 - v0.1 - for typist v0.9.0

tschinz

whynotlogic@gmail.com

HEI-Vs

Contents

1 Introduction	4
2 Installation	5
2.1 With cargo	5
2.2 MacOS	5
2.3 Linux	5
2.4 Windows	5
3 Formatting	6
3.1 Markup	6
3.2 Page Formatting	6
3.3 Space	6
3.4 Text Formatting	7
4 Elements	10
4.1 Headings	10
4.2 Lists	10
4.3 Custom Lists	11
4.4 Images	11
4.4.1 Alignment	11
4.4.2 Caption	12
4.4.3 Cluster	12
4.5 Tables	14
4.5.1 Karnaugh Tables	16
4.6 Icon Boxes	17
4.7 Color Boxes	18
4.8 Title Box	19
4.9 Exam Header	20
4.10 Exam Reminder	21
5 References	25
5.1 Links	25
5.2 Crossreferences	25
5.3 External References	25
5.4 Glossary	25
5.5 Acronym	26

6 Code	27
7 Math Equations	28
7.1 Align	28
7.2 Symbols	28
7.2.1 Accents	29
7.2.2 Equals & Operators	29
7.2.3 Scripts	29
7.2.4 Special Elements	29
7.2.5 Alphabeth	30
7.2.6 Logical	31
7.2.7 Operators	31
7.2.8 Arrows	31
7.2.9 Angles	33
7.2.10 Cool Symbols	33
7.2.11 Style	33
8 Emoji Symbols	35
Bibliography	36

Figures

Figure 1: ZNotes Icon	12
Figure 2: Multiple images one caption	12
Figure 3: Multiple images one caption	12
Figure 4: Caption left image	13
Figure 5: Caption right image	13
Figure 6: Caption topleft image	13
Figure 7: Caption topright image	13
Figure 8: Caption bottomleft image	13
Figure 9: Caption bottomright image	13
Figure 14: Some proof	28

Tables

Table 1: Table caption	14
Table 2: Links	25

Listings

Listing 1: Label inserts	25
Listing 2: Rust Code	27

Equations

Equation (1)	28
Equation (2)	28
Equation (3)	28
Equation (4)	28
Equation (5)	30
Equation (6)	30
Equation (7)	30
Equation (8)	30
Equation (9)	30
Equation (10)	30
Equation (11)	30
Equation (12)	30
Equation (13)	30
Equation (14)	30
Equation (15)	30
Equation (16)	30
Equation (17)	30
Equation (18)	30
Equation (19)	30
Equation (20)	30
Equation (21)	30
Equation (22)	30
Equation (23)	30

1 | Introduction

The goal of this document is to have the most common used elements for the markup language typst readily available. A detailed documentation can be found on their website: <https://typst.app/docs> It is to note that these are **my** most common used elements. For some elements custom templates are needed:

- [tablex](#)
- [myref](#)
- all files in the `00-templates/` folder such as
 - `boxes.typ`
 - `constants.typ`
 - `helpers.typ`
 - `items.typ`
 - `metadata.typ`
 - `template-*`

2 | Installation

2.1 With cargo

If you use already the rust programming language then you can use rust to install the latest toolchain.

```
# install rust and cargo
curl https://sh.rustup.rs -sSf | sh

# install typst
cargo install --git https://github.com/typst/typst
```

2.2 MacOS

On MacOS you can use homebrew

```
# install homebrew
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"

# install typst
brew install typst
```

2.3 Linux

In Linux you can use the commonly available package manager

```
brew install typst
pacman -S typst
xbps-install typst
sudo apt-get install typst
```

2.4 Windows

On Windows you can use chocolatey. See: <https://chocolatey.org/install>

```
# install chocolatey
# ensure to use a administrative powershell
Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://community.chocolatey.org/install.ps1'))

# install typst
choco install typst
```

3 | Formatting

3.1 Markup

Name	Example	Raw
Singleline Comment		//
Multiline Comment		/* */
Paragraph break		blankline
Line break		\
bold	bold	*bold*
italic	<i>italic</i>	_italic_
monospaced	monospaced	`monospaced`
math	$x = 1$	$x=1$
lowercase	lower	#lower("LoWeR")
uppercase	UPPER	#upper("UpPeR")
smallcaps	SMALLCAPS	#smallcaps("SmallCaps")
smartquote	“test”	#smartquote()test#smartquote())
overline	<u>overline</u>	#overline("overline")
underline	<u>underline</u>	#underline("underline")
strike	strike	#strike("strike")
sub	Text _{sub}	Text#sub("sub")
super	Text ^{super}	Text#super("super")
Label		<label>
Reference		@label

3.2 Page Formatting

```
#pagebreak() // pagebreak
#parbreak() // parbreak
\ // linebreak
```

3.3 Space

A

B

A #h(5cm) B,

C

D

C #v(0.2cm) D

3.4 Text Formatting

For the custom textsizes and colors you need to import:

```
#import "../01-tail/constants.typ": *
```

Name	Example	Raw
Sizes	8pt text tiny text	text(8pt, "8pt text") text(tiny "tiny text")
	9pt text smaller text	text(9pt, "9pt text") text(smaller "smaller text")
	10pt text small text	text(10pt, "10pt text") text(small "small text")
	11pt text normal text	text(11pt, "11pt text") text(normal "normal text")
	14pt text large text	text(14pt, "14pt text") text(large "large text")
	16pt text larger text	text(16pt, "16pt text") text(larger "larger text")
	24pt text huge text	text(24pt, "24pt text") text(huge "huge text")
	36pt text huger text	text(36pt, "36pt text") text(huger "huger text")
	Fira Sans	text(font:"Fira Sans", "Fira Sans")
	Fira Mono	text(font:"Fira Mono", "Fira Mono")
	Source Sans Pro	text(font:"Source Sans Pro", "Source Sans Pro")
	New Computer Modern	text(font:"New Computer Modern", "New Computer Modern")
Types	New Computer Modern Sans	text(font:"New Computer Modern Sans", "New Computer Modern S

Alignment	start	align(start){start}
	end	align(end){end}
	left	align(left){left}
	center	align(center){center}
	right	align(right){right}
	top	align(top){top}
	horizon	align(horizon){horizon}
	bottom	align(bottom){bottom}
	center + horizon	align(center + horizon){center + horizon}

Colors	black	<code>#text(fill:black)[black]</code>
	red	<code>#text(fill:red)[red]</code>
	green	<code>#text(fill:green)[green]</code>
	blue	<code>#text(fill:blue)[blue]</code>
	purple	<code>#text(fill:purple)[purple]</code>
	gray-80	<code>#text(fill:gray-80)[gray-80]</code>
	gray-70	<code>#text(fill:gray-70)[gray-70]</code>
	gray-60	<code>#text(fill:gray-60)[gray-60]</code>
	gray-50	<code>#text(fill:gray-50)[gray-50]</code>
	gray-40	<code>#text(fill:gray-40)[gray-40]</code>
	gray-30	<code>#text(fill:gray-30)[gray-30]</code>
	gray-20	<code>#text(fill:gray-20)[gray-20]</code>
	gray-10	<code>#text(fill:gray-10)[gray-10]</code>
	hei-orange	<code>#text(fill:hei-orange)[hei-orange]</code>
	hei-blue	<code>#text(fill:hei-blue)[hei-blue]</code>
	hei-pink	<code>#text(fill:hei-pink)[hei-pink]</code>
	hei-yellow	<code>#text(fill:hei-yellow)[hei-yellow]</code>
	hei-green	<code>#text(fill:hei-green)[hei-green]</code>
	spl-green	<code>#text(fill:spl-green)[spl-green]</code>
	spl-blue	<code>#text(fill:spl-blue)[spl-blue]</code>
	spl-pink	<code>#text(fill:spl-pink)[spl-green]</code>
	color-info	<code>#text(fill:color-info)[color-info]</code>
	color-idea	<code>#text(fill:color-idea)[color-idea]</code>
	color-warning	<code>#text(fill:color-warning)[color-warning]</code>
	color-important	<code>#text(fill:color-important)[color-important]</code>
	color-fire	<code>#text(fill:color-fire)[color-fire]</code>
	color-rocket	<code>#text(fill:color-rocket)[color-rocket]</code>
	color-todo	<code>#text(fill:color-todo)[color-todo]</code>
	code-bg	<code>#text(fill:code-bg)[code-bg]</code>
	code-border	<code>#text(fill:code-border)[code-border]</code>

4 | Elements

4.1 Headings

```
= Heading 1
== Heading 1.1
=== Heading 1.1.1
==== Heading 1.1.1.1
...

```

4.2 Lists

- First
- Second
- Third

```
- First
- Second
- Third

```

- First
 - Second
 - Third

```
- First
- Second
- Third

```

- First
- Second
- Third

```
- First
- Second
- Third

```

- First
- Second
- Third

```
list(
  [First],
  [Second],
  [Third],
)

```

1. First
 1. Second
2. Third

```
+ First
+ Second
+ Third
Text
4. Fourth
+ Fifth

```

Text

4. Fourth
5. Fifth

1. First
 - a) Second
2. Third

```
+ First
#set enum(numbering: "a")
+ Second
+ Third
Text

```

Text

4. Fourth
5. Fifth

4. Fourth
+ Fifth

4.3 Custom Lists

```
#import "../00-templates/items.typ": *
```

- ☑ item-list
- 🏷 item-checkbadge
- ⊙ item-checkcircle
- ◻ item-checksquare
- ✓ item-check
- 📁 item-file
- 📁 item-folder
- ⊗ item-xcircle
- ⊗ item-xsquare
- × item-x

```
#item-list(content:"item-list")
#item-checkbadge(content:"item-checkbadge")
#item-checkcircle(content:"item-checkcircle")
#item-checksquare(content:"item-checksquare")
#item-check(content:"item-check")
#item-file(content:"item-file")
#item-folder(content:"item-folder")
#item-xcircle(content:"item-xcircle")
#item-xsquare(content:"item-xsquare")
#item-x(content:"item-x")
```

4.4 Images

4.4.1 Alignment

left



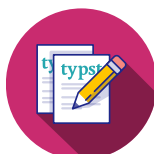
```
#image("../04-resources/icon.svg",
width: 2cm)
```

center



```
#align(center,
image("../04-resources/icon.svg",
width: 2cm)
)
```

right



```
#align(right,
image("../04-resources/icon.svg",
width: 2cm)
)
```

4.4.2 Caption

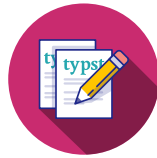


Figure 1: ZNotes Icon

```
#figure(
  image("../04-resources/icon.svg",
    width: 2cm),
  caption: [ZNotes Icon]
) <fig-icon>
```

4.4.3 Cluster

Two images one caption

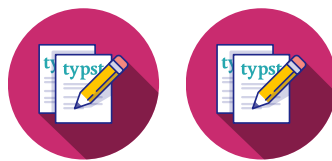


Figure 2: Multiple images **one** caption

```
#figure(
  tablex(
    columns: 2,
    stroke: none,
    align: center + horizon,
    image(icon, width: 2cm), image(icon, width: 2cm)
  ),
  caption: [Multiple images *one* caption]
)
```

Four images one caption

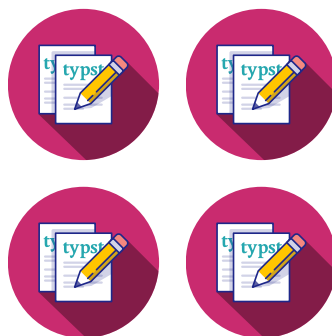


Figure 3: Multiple images **one** caption

```
#figure(
  tablex(
    columns: 2,
    stroke: none,
    align: center + horizon,
```

```

    image(icon, width: 2cm), image(icon, width: 2cm),
    image(icon, width: 2cm), image(icon, width: 2cm),
  ),
  caption: [Multiple images *one* caption]
)

```

Two images two caption

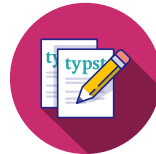
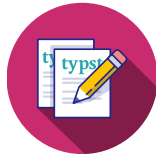


Figure 4: Caption left image Figure 5: Caption right image

```

#align(center,
  tablex(
    columns: 2,
    stroke: none,
    align: center + horizon,
    figure(image(icon, width: 2cm), caption: [Caption left image]), figure(image(icon,
width: 2cm), caption: [Caption right image]),
  ))

```

Four images four caption



Figure 6: Caption topleft image

Figure 7: Caption topright image

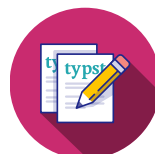
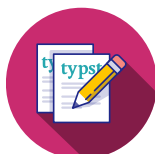


Figure 8: Caption bottomleft image Figure 9: Caption bottomright image

```

#align(center,
  tablex(
    columns: 2,
    stroke: none,
    align: center + horizon,
    figure(image(icon, width: 2cm), caption: [Caption topleft image]),
    figure(image(icon, width: 2cm), caption: [Caption topright image]),
    figure(image(icon, width: 2cm), caption: [Caption bottomleft image]),
    figure(image(icon, width: 2cm), caption: [Caption bottomright image]),
  ))

```

4.5 Tables

For all `#tablex` command the appropriate module nedds to be imported

```
#import "../00-templates/tablex.typ": *
```

Tables with and without caption

	Col1	Col2
Row1	cell-0-0	cell-1-0
Row2	cell-0-1	cell-1-1

	Col1	Col2
Row1	cell-0-0	cell-1-0
Row2	cell-0-1	cell-1-1

Table 1: Table caption

```
tablex(  
  columns: 3,  
  align: center + horizon,  
  []      , [*Col1*] , [*Col2*],  
  [*Row1*], "cell-0-0", "cell-1-0",  
  [*Row2*], "cell-0-1", "cell-1-1",  
)
```

```
figure(  
  tablex(  
    columns: 3,  
    align: center + horizon,  
    []      , [*Col1*] , [*Col2*],  
    [*Row1*], "cell-0-0", "cell-1-0",  
    [*Row2*], "cell-0-1", "cell-1-1",  
  ),  
  kind: table,  
  caption: [Table Caption]  
)
```

Tables with cell spans

	Col1	Col2
Row1	cell-0	cell-1-0
Row2		cell-1-1

	Col1	Col2
Row1	cell-0	
Row2	cell-0-1	cell-1-1

```
tablex(  
  columns: 3,  
  align: center + horizon,  
  []      , [*Col1*] , [*Col2*],  
  [*Row1*], rowspanx(2)[cell-0],  
  "cell-1-0",  
  [*Row2*],                "cell-1-1",  
)
```

```
tablex(  
  columns: 3,  
  align: center + horizon,  
  []      , [*Col1*] , [*Col2*],  
  [*Row1*], colspanx(2)[cell-0],  
  [*Row2*], "cell-0-1", "cell-1-1",  
)
```

Table Design

	Col1	Col2
Row1	cell-0-0	cell-1-0

	Col1	Col2
Row1	cell-0-0	cell-1-0

Row2	cell-0-1	cell-1-1
------	----------	----------

```
tablex(  
  columns: 3,  
  auto-vlines: false,  
  align: center + horizon,  
  [], [*Col1*] , [*Col2*],  
  [*Row1*], "cell-0-0", "cell-1-0",  
  [*Row2*], "cell-0-1", "cell-1-1",  
)
```

Row2	cell-0-1	cell-1-1
------	----------	----------

```
tablex(  
  columns: 3,  
  auto-hlines: false,  
  align: center + horizon,  
  [], [*Col1*] , [*Col2*],  
  [*Row1*], "cell-0-0", "cell-1-0",  
  [*Row2*], "cell-0-1", "cell-1-1",  
)
```

	Col1	Col2
Row1	cell-0-0	cell-1-0
Row2	cell-0-1	cell-1-1

```
#tablex(  
  columns: 3,  
  auto-lines: false,  
  align: center + horizon,  
  (), vlinex(stroke: blue) , vlinex(), (),  
  [], [*Col1*] , [*Col2*], hlinex(stroke: red),  
  [*Row1*], "cell-0-0", "cell-1-0", hlinex(),  
  [*Row2*], "cell-0-1", "cell-1-1",  
)
```

c	b	a	cb	ba	y
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	0	0
0	1	1	0	1	0
1	0	0	0	0	0
1	0	1	0	0	1
1	1	0	1	0	1
1	1	1	1	1	1

```
#tablex(  
  columns: 6,  
  auto-vlines: false,  
  auto-hlines: false,  
  stroke: 0.5pt,  
  align: center+ horizon,  
  (), vlinex(), vlinex(), vlinex(stroke: 1pt) , vlinex(), vlinex(stroke:1pt),  
  [$c$], [$b$], [$a$], [$c b$], [$b a$], [$y$], hlinex(stroke: 1pt),  
  [`0`], [`0`], [`0`], [`0`], [`0`], [`0`], hlinex(stroke: 0.5pt),  
)
```

```

[\`0`], [\`0`], [\`1`], [\`0`],   [\`0`],   [\`1`], hlinex(stroke: 0.5pt),
[\`0`], [\`1`], [\`0`], [\`0`],   [\`0`],   [\`0`], hlinex(stroke: 0.5pt),
[\`0`], [\`1`], [\`1`], [\`0`],   [\`1`],   [\`0`], hlinex(stroke: 1pt),
[\`1`], [\`0`], [\`0`], [\`0`],   [\`0`],   [\`0`], hlinex(stroke: 0.5pt),
[\`1`], [\`0`], [\`1`], [\`0`],   [\`0`],   [\`1`], hlinex(stroke: 0.5pt),
[\`1`], [\`1`], [\`0`], [\`1`],   [\`0`],   [\`1`], hlinex(stroke: 0.5pt),
[\`1`], [\`1`], [\`1`], [\`1`],   [\`1`],   [\`1`],
)

```

4.5.1 Karnaugh Tables

```
#import "../00-templates/karnaugh.typ"
```

	\overline{c}				
	\overline{b}				
y	1	0	1	0	
	1	0	1	1	a

```
#karnaugh(content:((1, 0, 1, 0),
(1, 0, 1, 1),))
```

	\overline{d}				
	\overline{c}				
y	1	0	0	0	
	1	0	0	1	
	1	1	0	1	a
	1	0	1	1	b

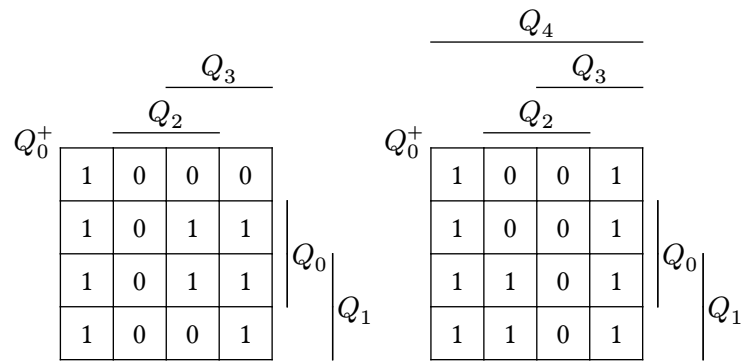
```
#karnaugh(content:((1, 0, 0, 0),
(1, 0, 0, 1),
(1, 1, 0, 1),
(1, 0, 1, 1),))
```

	\overline{d}				
	\overline{c}				
y	1	0	0	0	
	1	0	1	1	
	1	0	1	1	a
	1	0	0	1	b

	\overline{e}				
	\overline{d}				
y	1	0	0	1	
	1	0	0	1	
	1	1	0	1	a
	1	1	0	1	b

```
#karnaugh(content:((1, 0, 0, 0),
(1, 0, 1, 1),
(1, 0, 1, 1),
(1, 0, 0, 1),

(1, 0, 0, 1),
(1, 0, 0, 1),
(1, 1, 0, 1),
(1, 1, 0, 1),))
```



```
#importantbox()["importantbox"]
```



```
#firebox()["firebox"]
```



```
#rocketbox()["rocketbox"]
```



```
#todobox()["todobox"]
```



```
#iconbox(icon:"../04-resources/placeholder.svg", linecolor:
hei-blue)["iconbox"]
```

```
#iconbox(linecolor: hei-pink)["iconbox without icon"]
```

4.7 Color Boxes

```
#import "../00-templates/boxes.typ": *
```

Exercise

Some text

```
#colorbox( title: "Exercise", color:hei-blue)[Some text]
```

Attention

Some text

```
#colorbox( title: "Attention", color:hei-pink)[Some text]
```

Consider

Some text

```
#slantedColorbox( title: "Consider", color:hei-green)[Some text]
```

Information

Some text

```
#slantedColorbox( title: "Information", color:hei-orange)[Some text]
```

4.8 Title Box

```
#import "../00-templates/sections.typ": *
```

Title
Subtitle

```
#titlebox(title:[Title], subtitle:[Subtitle])
```

Title
Subtitle

```
#titlebox(width:50%, radius:0pt, border:1pt, linecolor: hei-blue, titlesize: larger,  
subtitle: large, title:[Title], subtitle:[Subtitle])
```

Title

```
#titlebox(linecolor: hei-green, titlesize: larger, subtitlesize: large, title:[Title])
```

4.9 Exam Header

```
#import "../00-templates/sections.typ": *
```

Name: -----

```
#exam_header(nbrEx:0, lang: "en")
```

Name: -----

Grade

```
#exam_header(nbrEx:1, lang: "en")
```

Name: -----

1	Grade
(10)	

```
#exam_header(nbrEx:2, pts:10, lang: "en")
```

Name: -----

1	2	Grade
(10)	(10)	

```
#exam_header(nbrEx:3, pts:10, lang: "en")
```

Name: -----

1	2	3	Grade
(10)	(10)	(10)	

```
#exam_header(nbrEx:4, pts:10, lang: "en")
```

Name:

1	2	3	4	Grade
(10)	(10)	(10)	(10)	

```
#exam_header(nbrEx:5, pts:10, lang: "en")
```

Name:

1	2	3	4	5	Grade
(10)	(10)	(10)	(10)	(10)	

```
#exam_header(nbrEx:6, pts:10, lang: "en")
```

Name:

1	2	3	4	5	6	Grade
(10)	(10)	(10)	(10)	(10)	(10)	

```
#exam_header(nbrEx:7, pts:10, lang: "en")
```

Name:

1	2	3	4	5	6	7	Grade
(10)	(10)	(10)	(10)	(10)	(10)	(10)	

```
#exam_header(nbrEx:8, pts:10, lang: "en")
```

Name:

1	2	3	4	5	6	7	8	Grade
(10)	(10)	(10)	(10)	(10)	(10)	(10)	(10)	

```
#exam_header(nbrEx:9, pts:10, lang: "en")
```

Name:

1	2	3	4	5	6	7	8	9	Grade
(10)	(10)	(10)	(10)	(10)	(10)	(10)	(10)	(10)	

```
#exam_header(nbrEx:10, pts:10, lang: "en")
```

4.10 Exam Reminder

```
#import "../00-templates/sections.typ": *
```

**Exam Reminder:**

You can only use the following items:

- a laptop without internet connection
- a pocketcalculator
- all paper documents you want

Viel Glück!

```
#exam_reminder_did(lang: "en")
```

**Prüfungserinnerung:**

Sie können nur die folgenden Gegenstände verwenden:

- ein Laptop ohne Internetanschluss
- einen Taschenrechner
- alle Papirdokumente

Viel Glück!

```
#exam_reminder_did(lang: "de")
```

**Rappel d'examen :**

Vous ne pouvez utiliser que les éléments suivants :

- un ordinateur portable sans connexion internet
- une calculatrice de poche
- tous les documents papier que vous souhaitez

Good Luck!

```
#exam_reminder_did(lang: "fr")
```

Exam Reminder:

You can only use the following items:

- the two-page summary you created.
- a pocketcalculator

In addition, properly comment all high-level and assembler code to explain its purpose and how it fits into the program structure.

Viel Glück!

```
#exam_reminder_car(lang: "en")
```

Prüfungserinnerung:

Sie können nur die folgenden Elemente verwenden:

- die zweiseitige Zusammenfassung, die Sie erstellt haben.
- einen Taschenrechner

Kommentieren Sie ausserdem den gesamten High-Level- und Assembler-Code ordnungsgemäss aus, um seinen Zweck und seine Einbindung in die Programmstruktur zu erklären.

Viel Glück!

```
#exam_reminder_car(lang: "de")
```

Rappel d'examen :

Vous ne pouvez utiliser que les éléments suivants :

- le résumé de deux pages que vous avez créé.
- une calculatrice de poche

Commenter également tout le code de haut niveau et le code assembleur de manière appropriée afin d'expliquer son but et son intégration dans la structure du programme.

Good Luck!

```
#exam_reminder_car(lang: "fr")
```


5 | References

5.1 Links


Example	Raw
https://example.com	<code>https://example.com</code>
https://example.com	<code>#link("https://example.com")</code>
See example.com	<code>#link("https://example.com")[See example.com]</code>
whynotlogic@gmail.com	<code>#link("mailto:whynotlogic@gmail.com")[whynotlogic\@gmail.com]</code>
	<code>#link("https://tschinz.github.io/znotes")[#image(icon, width:0.5cm)]</code>

Table 2: Links

5.2 Crossreferences

In the document the following references were added.

```
= References <sec-ref>
== Links <sec-links>
#figure(image("../04-resources/icon.svg", width: 2cm)) <fig-icon>
#figure(tablex(...), kind:table) <tab-links>
#figure(align(left, raw(...)) <code-ref>
$ sum_(k=1)^n k = (n(n+1)) / 2 $ <math-eq1> #ref(<math-eq1>)
```

Listing 1: Label inserts

They can be references as follows:

Type	Example	Raw
Section	Section 5	<code>@sec-ref</code>
Subsection	Section 5.1	<code>@sec-links</code>
Table	Table 2	<code>@tab-links</code>
Code	Listing 1	<code>@code-ref</code>

5.3 External References

Example	Raw
[1]	<code>#cite(label("stateoftheArt"))</code>
[1, p.7ff]	<code>#cite(<stateoftheArt>, supplement:[p.7ff])</code>
[1]	<code>@stateoftheArt</code>

5.4 Glossary

The glossary entries need to be defined in 03-tail/glossary.typ. For the glossary functions the “import” of 01-head/helpers.typ is needed.

```
#import "../01-head/helpers.typ": *
#import "../03-tail/glossary.typ": *
```

Example

Scrum

Scrum is an agile process framework for managing complex knowledge work, with an initial emphasis on software development, although it has been used in other fields and is slowly starting to be explored for other complex work, research and advanced technologies.

Raw

```
#gls-scrumm.name
```

```
#gls-scrumm.description
```

5.5 Acronym

The acronym entries need to be defined in 03-tail/glossary.typ. For the acronym functions the “import” of 01-head/helpers.typ is needed.

```
#import "../01-head/helpers.typ": *
#import "../03-tail/glossary.typ": *
```

Example

AR

AR

Augmented Reality

Augmented Reality)

Augmented Reality (AR)

Augmented Reality (AR)

Raw

```
#acr-ar.abr
```

```
#acrshort(acr-ar.abbr)
```

```
#acr-ar.long
```

```
#acrlong(acr-ar)
```

```
#acr-ar.long (#acr-ar.abbr)
```

```
#acrfull(acr-ar)
```

6 | Code

inline monospaced string

```
`inline monospaced string`
```

```
fn main() {println!("Hello world!")}
```

```
raw(lang:"rust",  
  "fn main() {println!(\"Hello world!  
  \")\"  
  )
```

```
fn main() {  
  println!("Hello world!")  
}
```

```
``rust  
fn main() {  
  println!("Hello world!")  
}  
``
```

```
fn main() {  
  println!("Hello world!")  
}
```

```
#figure(  
  align(left,  
    ``rust  
    fn main() {  
      println!("Hello world!")  
    }  
    ``  
  ),  
  caption: [Rust Code],  
)
```

Listing 2: Rust Code

7 | Math Equations

Inline math

Let a and b , and c be the side of a right-angled triangle.

Let a and b , and c be the side of a right-angled triangle.

$$\sum_{k=1}^n k = \frac{n(n+1)}{2}$$

$\sum_{k=1}^n k = (n(n+1)) / 2$,

Fullline math

$$a^2 + b^2 = c^2 \tag{1}$$

$a^2 + b^2 = c^2$ `<math-eq1>`

Math with caption

$$\sum_{k=1}^n k = \frac{n(n+1)}{2} \tag{2}$$

Figure 14: Some proof

```
#figure(
  $ \sum_{k=1}^n k = (n(n+1)) / 2 $,
  caption: [Some proof]
)
```

7.1 Align

Formula

$$\begin{aligned} a_1 &= b_1 + c_1 = z_1 \\ a_2 &= b_2 + c_2 - d_2 + e_2 = z_1 \end{aligned} \tag{3}$$

Raw

```
$
a_1 = b_1 + c_1 = z_1 \ \
a_2 = b_2 + c_2 - d_2 + e_2 = z_1
$
```

$$\begin{aligned} a_1 &= b_1 + c_1 &= z_1 \\ a_2 &= b_2 + c_2 - d_2 + e_2 = z_1 \end{aligned} \tag{4}$$

```
$
a_1 &= b_1 + c_1 &= z_1 \ \
a_2 &= b_2 + c_2 - d_2 + e_2 &= z_1
$
```

7.2 Symbols

This is an incomplete list for all symbols goto [here](#)

Outside of the $\$$ math environment the symbols can be accessed with `.`

7.2.1 Accents

Sym- bol	Raw	Sym- bol	Raw	Sym- bol	Raw
\grave{x}	<code>\$grave(x)\$</code>	\acute{x}	<code>\$acute(x)\$</code>	\hat{x}	<code>\$hat(x)\$</code>
\tilde{x}	<code>\$tilde(x)\$</code>	\breve{x}	<code>\$breve(x)\$</code>	\dot{x}	<code>\$dot(x)\$</code>
\ddot{x}	<code>\$dot.double(x)\$</code>	\ddot{x}	<code>\$dot.triple(x)\$</code>	\ddot{x}	<code>\$dot.quad(x)\$</code>
\ddot{x}	<code>\$diaer(x)\$</code>	$\circ x$	<code>\$circle(x)\$</code>	\ddot{x}	<code>\$acute.double(x)\$</code>
\tilde{x}	<code>\$caron(x)\$</code>	\vec{x}	<code>\$arrow(x)\$</code>	\vec{x}	<code>\$arrow.l(x)\$</code>
\cancel{x}	<code>\$cancel(x)\$</code>	\bar{x}	<code>\$macron(x)\$</code>	\overline{xyz}	<code>\$overline(xyz)\$</code>
\overline{xyz}	<code>\$overline(xyz)\$</code>	\underbrace{xyz}	<code>\$underbrace(xyz)\$</code>	\overbrace{xyz}	<code>\$overbrace(xyz)\$</code>
\underbrace{xyz}	<code>\$underbracket(xyz)\$</code>	\overbrace{xyz}	<code>\$overbracket(xyz)\$</code>	\overbrace{xyz}	<code>\$overbracket(xyz)\$</code>

7.2.2 Equals & Operators

Sym- bol	Raw	Sym- bol	Raw	Sym- bol	Raw
$=$	<code>\$=\$</code>	$=$	<code>\$eq\$</code>	\neq	<code>\$eq.not\$</code>
\neq	<code>\$!=\$</code>	\equiv	<code>\$equiv\$</code>	\neq	<code>\$equiv.not\$</code>
\simeq	<code>\$tilde.eq\$</code>	\napprox	<code>\$tilde.eq.not\$</code>	$=$	<code>\$eq.small\$</code>
\geq	<code>\$gt.eq\$</code>	\nlessgtr	<code>\$gt.eq.not\$</code>	\leq	<code>\$lt.eq\$</code>
\nlessgtr	<code>\$lt.eq.not\$</code>	\approx	<code>\$approx\$</code>	\cong	<code>\$approx.eq\$</code>
\napprox	<code>\$approx.not\$</code>	$:$	<code>\$colon\$</code>	$:=$	<code>\$colon.eq\$</code>
$=:$	<code>\$eq.colon\$</code>	$::=$	<code>\$colon.double.eq\$</code>	$+$	<code>\$+\$</code>
$+$	<code>\$plus\$</code>	$+$	<code>\$plus.small\$</code>	\pm	<code>\$plus.minus\$</code>
\oplus	<code>\$plus.circle\$</code>	$-$	<code>\$-\$</code>	$-$	<code>\$minus\$</code>
\mp	<code>\$minus.plus\$</code>	\ominus	<code>\$minus.circle\$</code>		

7.2.3 Scripts

Sym- bol	Raw	Sym- bol	Raw	Sym- bol	Raw
x_1	<code>\$x_1\$</code>	x_{12}	<code>\$x_(12)\$</code>	x_1	<code>\$scripts(x)_1\$</code>
x_1	<code>\$x_1\$</code>	x_{12}	<code>\$x_(12)\$</code>	x_1	<code>\$scripts(x)_1\$</code>
x_1^2	<code>\$x_1^2\$</code>	x_{12}^{34}	<code>\$x_(12)^(34)\$</code>	x_1^2	<code>\$scripts(x)_1^2\$</code>
x_1^2	<code>\$x_1^2\$</code>	x_{12}^{34}	<code>\$x_(12)^(34)\$</code>	x_1^2	<code>\$scripts(x)_1^2\$</code>

7.2.4 Special Elements

Symbol

Raw

$$\binom{n}{k} \quad (5) \quad \$ \text{binom}(n, k) \$$$

$$\left\lfloor \frac{1}{2} \right\rfloor \quad (7) \quad \$ \text{round}(1, 2) \$$$

$$\begin{pmatrix} 1 & 2 & \dots & 10 \\ 2 & 2 & \dots & 10 \\ \vdots & \vdots & \ddots & \vdots \\ 10 & 10 & \dots & 10 \end{pmatrix} (9)$$

```
$ mat(
  1, 2, ..., 10;
  2, 2, ..., 10;
  ...,
  dots.v,
  dots.v,
  dots.down,
  dots.v;
  10, 10, ...,
  10;
) $
```

$$\sum_{k=0}^n a_k \quad (11) \quad \$ \text{sum_}(k=0)^n \text{ a_k} \$$$

$$\sqrt[3]{x} \quad (13) \quad \$ \text{root}(3, x) \$$$

```
$ f(x, y) := cases(
  1 "if" (x dot y)/2 <= 0,
  2 "if" x "is even",
  3 "if" x in NN,
  4 "else",
) $
```

$$\frac{1}{2} \quad (16) \quad \$ \text{frac}(1,2) \$$$

$$\frac{(x+1)}{(x+2)} \quad (18) \quad \$ ((x+1))/(x+2)) \$$$

$$n! = \prod_{k=1}^n k \quad (20) \quad \$ \text{product_}(k=1)^n \text{ k} =$$

$$\int \quad (22) \quad \$ \text{integral} \$$$

Symbol

Raw

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad (6) \quad \$ \text{vec}(1, 2, \text{delim: "["}) \$$$

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \quad (8) \quad \$ \text{mat}(1,2; 3,4) \$$$

$$\sum a_k \quad (10) \quad \$ \text{sum} \text{ a_k} \$$$

$$\sum_{k=0}^n a_k \quad (12) \quad \$ \text{scripts}(\text{sum})_{(k=0)}^n \text{ a_k} \$$$

$$f(x, y) := \begin{cases} 1 & \text{if } \frac{x \cdot y}{2} \leq 0 \\ 2 & \text{if } x \text{ is even} \\ 3 & \text{if } x \in \mathbb{N} \\ 4 & \text{else} \end{cases} \quad (14)$$

$$\frac{1}{2} \quad (15) \quad \$ 1/2 \$$$

$$\frac{x+1}{x+2} \quad (17) \quad \$ (x+1)/(x+2) \$$$

$$\prod \quad (19) \quad \$ \text{product} \$$$

$$n! = \prod_{k=1}^n k \quad (21) \quad \$ n! = \text{scripts}(\text{product})_{(k=1)}^n \text{ k} \$$$

$$\int_a^b f(x) \quad (23) \quad \$ \text{integral} \$$$

7.2.5 Alphabeth

Symbol

Raw

$\alpha\beta\gamma\delta\epsilon\zeta\eta\theta\iota\kappa\lambda\mu\nu\xi\omicron\rho\sigma\tau\upsilon\varphi\chi\psi\omega$	<code>\$alpha beta gamma delta epsilon zeta eta theta iota kappa lambda mu nu xi omicron pi rho sigma tau upsilon phi chi psi omega\$</code>
$\mathrm{A}\mathrm{B}\mathrm{G}\mathrm{\Delta}\mathrm{E}\mathrm{Z}\mathrm{H}\mathrm{\Theta}\mathrm{I}\mathrm{K}\mathrm{A}\mathrm{M}\mathrm{N}\mathrm{\Xi}\mathrm{O}\mathrm{I}\mathrm{P}\mathrm{\Sigma}\mathrm{T}\mathrm{T}\mathrm{\Phi}\mathrm{X}\mathrm{\Psi}\mathrm{\Omega}$	<code>\$Alpha Beta Gamma Delta Epsilon Zeta Eta Theta Iota Kappa Lambda Mu Nu Xi Omicron Pi Rho Sigma Tau Upsilon Phi Chi Psi Omega\$</code>
$\mathrm{A}\mathrm{B}\mathrm{C}\mathrm{D}\mathrm{E}\mathrm{F}\mathrm{G}\mathrm{H}\mathrm{I}\mathrm{J}\mathrm{K}\mathrm{L}\mathrm{M}\mathrm{N}\mathrm{O}\mathrm{P}\mathrm{Q}\mathrm{R}\mathrm{S}\mathrm{T}\mathrm{U}\mathrm{V}\mathrm{W}\mathrm{X}\mathrm{Y}\mathrm{Z}$	<code>\$AA BB CC DD EE FF GG HH II JJ KK LL MM NN OO PP QQ RR SS TT UU VV WW XX YY ZZ\$</code>

7.2.6 Logical

Symbol	Raw	Symbol	Raw	Symbol	Raw
\wedge	<code>\$and\$</code>	\bigwedge	<code>\$and.big\$</code>	$\&$	<code>\$amp\$</code>
\vee	<code>\$or\$</code>	$ $	<code>\$bar.v\$</code>	$*$	<code>\$ast.op\$</code>
$*$	<code>\$ast.basic\$</code>	$*$	<code>\$ast.low\$</code>	\oplus	<code>\$plus.circle\$</code>
\oplus	<code>\$plus.circle.big\$</code>				

7.2.7 Operators

Sym- bol	Raw	Sym- bol	Raw	Sym- bol	Raw
$\sin x$	<code>\$sin x\$</code>	$\cos x$	<code>\$cos x\$</code>	$\tan x$	<code>\$tan x\$</code>
$\arcsin x$	<code>\$arcsin x\$</code>	$\arccos x$	<code>\$arccos x\$</code>	$\arctan x$	<code>\$arctan x\$</code>
$\sinh x$	<code>\$sinh x\$</code>	$\cosh x$	<code>\$cosh x\$</code>	$\tanh x$	<code>\$tanh x\$</code>
$\arg x$	<code>\$arg x\$</code>	$\csc x$	<code>\$csc x\$</code>	$\deg x$	<code>\$deg x\$</code>
$\det x$	<code>\$det x\$</code>	$\dim x$	<code>\$dim x\$</code>	$\exp x$	<code>\$exp x\$</code>
$\bmod x$	<code>\$mod x\$</code>	$\inf x$	<code>\$inf x\$</code>	$\log x$	<code>\$log x\$</code>
$\lim x$	<code>\$lim x\$</code>	$\liminf x$	<code>\$liminf x\$</code>	$\limsup x$	<code>\$limsup x\$</code>
$\min x$	<code>\$min x\$</code>	$\max x$	<code>\$max x\$</code>	$\sup x$	<code>\$sup x\$</code>

7.2.8 Arrows

SymRaw	SymRaw	SymRaw	
Arrows right			
\rightarrow	<code>\$arrow\$</code>	\mapsto	<code>\$arrow.bar\$</code>
\mapsto	<code>\$arrow.bar.long\$</code>	\Rightarrow	<code>\$arrow.double.long\$</code>
\Rightarrow	<code>\$arrow.double.bar\$</code>	\Rightarrow	<code>\$arrow.quad\$</code>
\Rightarrow	<code>\$arrow.stroked\$</code>	\dashrightarrow	<code>\$arrow.dashed\$</code>
\hookrightarrow	<code>\$arrow.curve\$</code>	\rightsquigarrow	<code>\$arrow.squiggly\$</code>
Arrows left			

\leftarrow	<code>\$arrow.l\$</code>	\longleftarrow	<code>\$arrow.l.long\$</code>	$\bar{\leftarrow}$	<code>\$arrow.l.bar\$</code>
\longleftarrow	<code>\$arrow.l.bar.long\$</code>	\Leftrightarrow	<code>\$arrow.l.double\$</code>	\Longleftrightarrow	<code>\$arrow.l.double.long\$</code>
$\bar{\leftarrow}$	<code>\$arrow.l.double.bar\$</code>	\Longleftrightarrow	<code>\$arrow.l.double.bar.long\$</code>	\Leftrightarrow	<code>\$arrow.l.quad\$</code>
$\bar{\leftarrow}$	<code>\$arrow.l.stroked\$</code>	\blackleftarrow	<code>\$arrow.l.filled\$</code>	\dashleftarrow	<code>\$arrow.l.dashed\$</code>
\curvearrowleft	<code>\$arrow.l.curve\$</code>	\squigarrowleft	<code>\$arrow.l.squiggly\$</code>	\looparrowleft	<code>\$arrow.l.loop\$</code>

Double Arrows Left Right

\leftrightarrow	<code>\$arrow.l.r\$</code>	\nleftrightarrow	<code>\$arrow.l.r.not\$</code>	\longleftrightarrow	<code>\$arrow.l.r.long\$</code>
\Leftrightarrow	<code>\$arrow.l.r.double\$</code>	\Longleftrightarrow	<code>\$arrow.l.r.double.long\$</code>	\nleftrightarrow	<code>\$arrow.l.r.double.not\$</code>
\Leftrightarrow	<code>\$arrow.l.r.stroked\$</code>	\blackleftrightarrow	<code>\$arrow.l.r.filled\$</code>	\loopleftrightarrow	<code>\$arrow.l.r.wave\$</code>

Arrows Top

\uparrow	<code>\$arrow.t\$</code>	$\bar{\uparrow}$	<code>\$arrow.t.bar\$</code>	\Uparrow	<code>\$arrow.t.double\$</code>
\Uparrow	<code>\$arrow.t.triple\$</code>	\Updownarrow	<code>\$arrow.t.quad\$</code>	\uparrow	<code>\$arrow.t.stroked\$</code>
\Uparrow	<code>\$arrow.t.filled\$</code>	\uparrow	<code>\$arrow.t.dashed\$</code>	\rightarrow	<code>\$arrow.t.curve\$</code>

Arrows Bottom

\downarrow	<code>\$arrow.b\$</code>	$\bar{\downarrow}$	<code>\$arrow.b.bar\$</code>	\Downarrow	<code>\$arrow.b.double\$</code>
\Downarrow	<code>\$arrow.b.triple\$</code>	\Downarrow	<code>\$arrow.b.quad\$</code>	\downarrow	<code>\$arrow.b.stroked\$</code>
\Downarrow	<code>\$arrow.b.filled\$</code>	\downarrow	<code>\$arrow.b.dashed\$</code>	\rightarrow	<code>\$arrow.b.curve\$</code>

Double Arrows Top Bottom

\Updownarrow	<code>\$arrow.t.b\$</code>	\Updownarrow	<code>\$arrow.t.b.double\$</code>	\Updownarrow	<code>\$arrow.t.b.stroked\$</code>
\Updownarrow	<code>\$arrow.t.b.filled\$</code>				

Arrows Diagonal Top Right

\nearrow	<code>\$arrow.tr\$</code>	\nearrow	<code>\$arrow.tr.double\$</code>	\nearrow	<code>\$arrow.tr.stroked\$</code>
\nearrow	<code>\$arrow.tr.filled\$</code>	\nearrow	<code>\$arrow.tr.hook\$</code>		

Arrows Diagonal Bottom Right

\searrow	<code>\$arrow.br\$</code>	\searrow	<code>\$arrow.br.double\$</code>	\searrow	<code>\$arrow.br.stroked\$</code>
\searrow	<code>\$arrow.br.filled\$</code>	\searrow	<code>\$arrow.br.hook\$</code>		

Arrows Diagonal Bottom Left

\swarrow	<code>\$arrow.bl\$</code>	\swarrow	<code>\$arrow.bl.double\$</code>	\swarrow	<code>\$arrow.bl.stroked\$</code>
\swarrow	<code>\$arrow.bl.filled\$</code>	\swarrow	<code>\$arrow.bl.hook\$</code>		





Arrows Diagonal Top Left

\nwarrow	<code>\$arrow.tl\$</code>	\nwarrow	<code>\$arrow.tl.double\$</code>	\nwarrow	<code>\$arrow.tl.stroked\$</code>
\nwarrow	<code>\$arrow.tl.filled\$</code>	\nwarrow	<code>\$arrow.tl.hook\$</code>		



















Double Arrows Diagonal

\nearrow	<code>\$arrow.tl.br\$</code>	\nearrow	<code>\$arrow.tr.bl\$</code>
------------	------------------------------	------------	------------------------------

Other Arrows

	<code>\$arrow.cw\$</code>		<code>\$arrow.cw.half\$</code>		<code>\$arrow.ccw\$</code>
	<code>\$arrow.ccw.half\$</code>				

7.2.9 Angles

Sym- bol	Raw	Sym- bol	Raw	Sym- bol	Raw
	<code>\$angle\$</code>		<code>\$angle.rev\$</code>		<code>\$angle.acute\$</code>
	<code>\$angle.acute\$</code>		<code>\$angle.arc\$</code>		<code>\$angle.arc.rev\$</code>
	<code>\$angle.l\$</code>		<code>\$angle.r\$</code>		<code>\$angle.l.double\$</code>
	<code>\$angle.r.double\$</code>		<code>\$angle.right\$</code>		<code>\$angle.right.rev\$</code>
	<code>\$angle.right.arc\$</code>		<code>\$angle.right.dot\$</code>		<code>\$angle.right.sq\$</code>
	<code>\$angle.spheric\$</code>		<code>\$angle.spheric.rev\$</code>		<code>\$angle.spheric.top\$</code>

7.2.10 Cool Symbols

Sym- bol	Raw	Sym- bol	Raw	Sym- bol	Raw
@	<code>\$at\$</code>	%	<code>\$co\$</code>	©	<code>\$copyright\$</code>
©	<code>\$copyright.sound\$</code>	°C	<code>\$degree.c\$</code>	€	<code>\$euro\$</code>
\$	<code>\$dollar\$</code>	£	<code>\$pound\$</code>	₩	<code>\$won\$</code>
¥	<code>\$yen\$</code>	฿	<code>\$bitcoin\$</code>	°F	<code>\$degree.f\$</code>
!	<code>\$excl\$</code>	¡	<code>\$excl.inv\$</code>	!!	<code>\$excl.double\$</code>
!?	<code>\$excl.quest\$</code>	↯	<code>\$arrow.zigzag\$</code>	⊗	<code>\$ast.circle\$</code>
⋆	<code>\$ast.triple\$</code>	χ	<code>\$chi\$</code>	☒	<code>\$floral\$</code>
⌘	<code>\$maltese\$</code>	¶	<code>\$pilcrow\$</code>	ℏ	<code>\$planck\$</code>
♣	<code>\$suit.club\$</code>	♦	<code>\$suit.diamond\$</code>	♥	<code>\$suit.heart\$</code>
♠	<code>\$suit.spade\$</code>	△	<code>\$triangle.stroked.nested2\$</code>		

7.2.11 Style

Symbol	Raw	Symbol	Raw
ABC123	<code>\$sans(A B C 1 2 3)\$</code>	𝒶𝒷𝒸123	<code>\$frak(A B C 1 2 3)\$</code>
ABC123	<code>\$mono(A B C 1 2 3)\$</code>	ABC123	<code>\$bb(A B C 1 2 3)\$</code>
ABC123	<code>\$cal(A B C 1 2 3)\$</code>		

Symbol	Raw
--------	-----

$$\sum_{i \in \mathbb{N}} 1 + i$$

```
#show math.equation: set text(font: "Fira Math")  
$sum_(i in NN) 1 + i$,
```

8 | Emoji Symbols

This is an incomplete list for all emoji goto [here](#)

If the emoji module is imported the #emoji can be removed

```
#import emoji: *
```

Sym	Raw	Sym	Raw
	#emoji.face		

```
#bibliography("../03-tail/bibliography.bib", style:"apa")  
#bibliography("../03-tail/bibliography.bib", style:"chicago-author-date")  
#bibliography("../03-tail/bibliography.bib", style:"chicago-notes")  
#bibliography("../03-tail/bibliography.bib", style:"ieee")  
#bibliography("../03-tail/bibliography.bib", style:"mla")
```

Bibliography

- [1] P. Fettke, “State-of-the-Art Des State-of-the-Art”, *Wirtschaftsinformatik*, pp. 257–266, 2006, doi: 10.1007/s11576-006-0057-3.