

Hetvid: A Typst template for lightweight notes

itpyi

Xijing Ci'en Institute of Translation, Tang Empire

DATE CREATED: 2025-03-27

DATE MODIFIED: 2025-04-22

Abstract

This is a template designed for writing scientific notes.

Contents

1. Usage	1
2. Fonts	3
2.1. Basic settings	3
2.2. On math fonts	4
2.3. On weight of New Computer Modern font	4
3. Headings	5
4. Headings	5
4.1. Second-level headings	5
4.1.1. Third-level headings	5
5. Paragraphs and Indentation	5
6. Quotes	6
7. Equation	7
8. Theorems	7
9. Figures and tables	8
9.1. In-paragraph figures and tables	8
9.2. Standalone figures and tables	8
9.3. Texts in figures and tables	10
10. Bibliography and citation	10
11. Designing philosophy and goals	10
Bibliography	11

In this doc, upcoming features are represented by *light font*.

1. Usage

You can import this template in 3 ways.

- Copy the files (including `hetvid.typ` and `dingli.typ`) to your working directory, and import it by

```
1 #import "hetvid.typ": *
```

- Copy this repository to your directory for local packages, then import it by

```
1 #import "@local/hetvid:0.1.0": *
```

- After we have published this to Typst Universe, you can import it by

```
1 #import "@local/hetvid:0.1.0": *
```

We recommend the second way.

Specifically, the directory for local packages is `{data-dir}/typst/packages/local/`, where `{data-dir}` is

- Linux: `$XDG_DATA_HOME` or `~/.local/share`;
- MacOS: `~/Library/Application Support`;
- Windows: `%APPDATA%`, where `%APPDATA%` is a variable, usually being

```
1 C:\Users\USERNAME\AppData\Roaming
```

You can check this in cmd by command

```
1 $ echo The value of ^%AppData^% is %AppData%
2 The value of %AppData% is C:\Users\USERNAME\AppData\Roaming
```

See <https://superuser.com/questions/632891/what-is-appdata>.

The user should copy this directory to this directory for local packages. For example for Windows, there should be such a directory in the end:

```
1 C:\Users\USERNAME\AppData\Roaming\typst\packages\local\hetvid\0.1.0\
```

Then the user can `#import` this template.

After importing, the user can set basic information by the following code. Note that default values have been given to variables affecting the format. If you do not need to change these default values, you do not need to write them when you use this template.

```
1 #show: hetvid.with(
2   // Information for the title
3   // Metadata
4   title: [Title],
5   author: "The author",
6   affiliation: "The affiliation",
7   header: "",
8   date-created: datetime.today().display(),
9   date-modified: datetime.today().display(),
10  abstract: [],
11  toc: true,
12
13  // Language
14  lang: "en",
15
16  // Information for format, only write the term you need to change
17  // Paper size
18  paper-size: "a4",
```

```

19
20 // Fonts
21 body-font: ("New Computer Modern", "Libertinus Serif", "TeX Gyre Termes",
22 "Songti SC", "Source Han Serif SC", "STSong", "Simsun", "serif"),
23 raw-font: ("DejaVu Sans Mono", "Cascadia Code", "Menlo", "Consolas", "New
24 Computer Modern Mono", "PingFang SC", "STHeiti", "华文细黑", "Microsoft
25 YaHei", "微软雅黑"),
26 heading-font: ("Helvetica", "Tahoma", "Arial", "PingFang SC", "STHeiti",
27 "Microsoft YaHei", "微软雅黑", "sans-serif"),
28 math-font: ("New Computer Modern Math", "Libertinus Math", "TeX Gyre Termes
29 Math"),
30 emph-font: ("New Computer Modern", "Libertinus Serif", "TeX Gyre Termes",
31 "Kaiti SC", "KaiTi_GB2312"),
32 body-font-size: 11pt,
33 body-font-weight: "regular", // set it to 450 if you want book-weight of
34 NewCM fonts
35 raw-font-size: 9pt,
36 caption-font-size: 10pt,
37 heading-font-weight: "regular",
38
39 // Colors
40 link-color: link-color,
41 muted-color: muted-color,
42 block-bg-color: block-bg-color,
43
44 // indention
45 ind: 1.5em,
46 justify: true,
47
48 // bibliography style, if lang == "zh", default to be "gb-7714-2015-numeric"
49 bib-style: (
50   en: "springer-mathphys",
51   zh: "gb-7714-2015-numeric",
52 ),
53
54 // Numbering level of theorems
55 thm-num-lv: 1,
56 )

```

2. Fonts

2.1. Basic settings

We have defined several types of fonts, with their default values shown in the code in Section 1. The default values are font families, which in Typst are implemented as an `array` type variable containing several fonts. For each character, the compiler will display it using the first font in the family that supports that character.

The font types are as follows:

Body font (`body-font`) Mainly used for the main text. Its size and weight are determined by `body-font-size` and `body-font-weight`.

Monospace font (`raw-font`) Used for plain text content, such as code. The default values are all monospaced fonts. Since the size of monospaced fonts may not match regular fonts of the same size, we provide the `raw-font-size` variable for users to adjust the size of monospace fonts.

Heading font (`heading-font`) Used for document titles and section headings. The default value is sans-serif fonts (e.g., Heiti), making the headings distinct from the main text and more modern. Users can also modify it to bold serif fonts. The weight of the headings can be adjusted by setting `heading-font-weight`.

Math font (`math-font`) Used for mathematical formulas. See Section 2.2 for details.

Emphasis font (`emph-font`) For documents in Chinese language, not relevant here.

Additionally, we provide the `caption-font-size` parameter to control the font size of captions.

2.2. On math fonts

From a self-consistent perspective, we recommend setting the math font and the body font to the same font (its math variant and regular form). For example, in the equations, we often encounter characters with textual meaning displayed in upright form. We do not want these characters to appear in different fonts inside and outside the equation. Similarly, numbers frequently appear in formulas and in the main text, possibly referring to the same objects. Therefore, we also do not want them to appear in different fonts.

The default math font is New Computer Modern Math, which corresponds to the body font New Computer Modern. This is the default font for \LaTeX .

2.3. On weight of New Computer Modern font

The default body font used in the template, New Computer Modern, and the math font, New Computer Modern Math, offer two weight options: the regular weight (400), which is the default in our template, and a slightly heavier book weight (450). If you wish to use the latter, set `body-font-weight` to 450.

The default behavior of the New Computer Modern font in Typst (version 0.13.1) can be slightly inconsistent. For example, if you set the fonts as follows:

```
1 #set text(font: "New Computer Modern")
2 #show math.equation: set text(font: "New Computer Modern Math")
```

the body text will use the New Computer Modern font with a default weight of regular (400), while the equations will use the New Computer Modern Math font with a default weight of book (450). As a result, the equations may appear slightly bolder than the body text. Our template avoids this issue by explicitly specifying the weight for both fonts.

3. Headings

4. Headings

The font size of first-level headings is 1.4 em. There is a vertical space of 2 em (relative to the heading font size, the same below) above the first-level heading and 1.2 em below it.

4.1. Second-level headings

The font size of second-level headings is 1.2 em. There is a vertical space of 1.5 em above the second-level heading and 1.2 em below it.

4.1.1. Third-level headings

The font size of third-level headings is 1.1 em. There is a vertical space of 1.5 em above the third-level heading and 1.2 em below it.

Although Typst provides functionality for higher-level headings, the template author strongly discourages using headings with too many levels, so no special customization has been made for their behavior. Generally, notes under 100 pages rarely use third-level headings. For example, Kitaev's articles [1–3] and Witten's lecture notes [4–6] do not use third-level headings.

5. Paragraphs and Indentation

Users can adjust whether to justify text alignment by setting `justify: true | false`, with the default of this template being justified alignment. Users can adjust the indentation amount using the `ind` parameter.

In Typst, first-line indentation can be controlled using the `all` option to determine whether all paragraphs are indented. However, its behavior is somewhat complex. In summary:

- The default behavior is `all: false`. In this case, for a paragraph, if the preceding block-level element is also a paragraph, the current paragraph will be indented; otherwise, it will not. Non-paragraph block-level elements include:
 - Empty elements: As a result, the first paragraph of block-level elements (e.g., block quotes, theorems) is not indented, which aligns with English typography conventions.
 - Section headings: The first paragraph of each section is not indented, which aligns with English typography conventions.
 - Displayed equations: No indentation follows displayed equations, which is generally expected. However, if a paragraph ends with a displayed equation, special handling is required.
 - Block quotes: Similar to displayed equations, this is generally expected. However, if a paragraph ends with a block quote, special handling is required (see Section 6).
 - Unordered and ordered lists: Similar to displayed equations, this is generally expected when lists appear within a paragraph. However, if a paragraph ends with a list, special handling is required.

- Figures: Paragraph following a non-floating figure is not indented. This is expected only when the figure is part of the paragraph. Other cases require special handling (see Section 9).
- Other block-level elements, such as theorem environments used in this template.

Overall, this setting generally aligns with conventions but requires special handling in some cases.

- Setting `all: true` indents all paragraphs. This setting also introduces some issues requiring special handling, such as paragraphs following displayed equations or block quotes. Usually we do not mean to start a new paragraph in these cases, but the contents are always indented. We can avoid this by enclosing for example the equations in a box.

The template author believes that maintaining the simplicity of Typst’s native syntax for displayed equations is crucial. Therefore, we adopt the first approach, i.e., not setting all paragraphs to be indented, while applying special handling to paragraphs that require indentation.

The implementation of this special handling is straightforward: a virtual paragraph is added before paragraphs that need indentation but are not indented by default. This makes the compiler treat it as a block-level “paragraph” element, but it does not appear in the compiled output. To achieve this, we provide the `par-vir` function to create such virtual paragraphs. In the template, we have already added virtual paragraphs where adjustments are needed for Chinese documents. In most cases, users do not need to worry about indentation behavior. Only when a user-written paragraph ends with a non-paragraph block-level element (e.g., equations, quotes, lists) and the following content needs to start a new paragraph, the user must insert:

```
1 #par-vir
```

to achieve correct indentation. See the example below.

Example 5.1. This paragraph ends with an equation,

$$1 + 1 = 2. \tag{1}$$

We want to start a new paragraph.

Without adding a virtual paragraph, we cannot start a new paragraph after the equation, even if we add a blank line.

$$1 + 1 = 2. \tag{2}$$

We failed to start a new paragraph. Of course, this blank line does not affect the vertical spacing.

6. Quotes

The template customizes the behavior of block quotes: they are indented by $2 \times \text{ind}$ on both sides, where “ind” takes the value given to `ind` by the user, or takes the default value 1.5 em. There is a vertical spacing of 1.5 em above and below the block quote, and the subsequent content is treated as a continuation of the previous paragraph, with no first-line indentation.

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.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aequ.

Lorem ipsum

As you see here, by default, the template does not create a new paragraph (that is, make an indentation) after a quote. If you need to start a new paragraph and want the subsequent content to have a first-line indentation, you can insert a virtual paragraph.

7. Equation

Displayed equations are numbered by default, such as

$$\mathcal{F}f(k) = \frac{1}{2\pi i} \int dk e^{ikx} f(x), \quad (3)$$

and can be referenced by their numbers, such as Eq. (3). When referencing equations, we provide the custom command `@{KEY}` to display only the number, leaving the prefix and parentheses to the user. This is because users may use different prefixes in the same document, such as Equation (1), the integral (1), or the isomorphism (1), and may also reference multiple equations simultaneously, such as properties (1-3). On the other hand, we provide a command `#eqref` for referring equations with abbreviation “Eq.” or “Eqs.”, depending on the number of keys in the argument. For example, `#eqref(<eq:fourier>)` outputs Eq. (3), while `#eqref(<eq:fourier>, <eq:trivial>)` outputs Eqs. (3, 4). We do this because the space after the dot in “Eq.” is supposed to be small and non-breaking, which is tedious for the user to type.

Displayed equations have vertical spacing of 1.2 em above and below, consistent with the default behavior of L^AT_EX. Without this spacing, the layout would be too crowded, as shown in the example below.

Example 7.1. If the text before and after the equation fills an entire line, and there is no spacing between the equation and the surrounding text, such as

$$1 + 1 = 2, \quad (4)$$

the visual effect will be very crowded. We want to avoid such an effect, so we add spacing above and below the equation.

8. Theorems

Since existing theorem packages are unsatisfactory in detail, the author has developed the [dingli package](#) to implement theorem environments. As the dingli package has not yet been published, for user convenience, we have copied the package into this template, i.e., the `dingli.typ` file. Functions are defined within to implement theorems, definitions, lemmas, corollaries, examples, proofs, etc. Examples in this document have used the provided `#example` function.

Our theorem package customizes the following details:

- Indentation issues, including the indentation of the paragraph following the theorem and the indentation of the theorem itself in Chinese documents.
- Vertical spacing with the surrounding context. This template sets it to 1.5 em.

- Numbering level issues. You can choose not to number by headings, number by first-level headings, or number by higher-level headings. This can be adjusted via `thm-num-lv: 0|1|...`. The template defaults to numbering by first-level headings.

Theorem 8.1. This is a theorem. In Chinese environments, the theorem still indents the first line by 2 characters, does not indent as a whole, and is not flush with the left margin.

This is a paragraph following the theorem, note the vertical spacing.

Theorem 8.2 (*s*-Theorem). This is a named theorem. Note the realization of the theorem name and separator.

Theorem 8.3. This is a theorem. The vertical spacing before and after the theorem is set to weak, so there is no extra spacing.

Proof. This is a proof.

$$1 + 1 = 2 \tag{5}$$

Does this really need proof?

This is another paragraph in the proof. ■

Other types include lemmas, corollaries, definitions, etc.

Lemma 8.1. This is a lemma.

Corollary 8.1. This is a corollary.

Definition 8.1. This is a definition.

You can reference theorems using Typst's general syntax, such as [Definition 8.1](#).

9. Figures and tables

9.1. In-paragraph figures and tables

Sometimes we insert figures or tables directly into the text: they are closely related to the context and do not have much standalone value. Therefore, we want them to be part of the paragraph rather than independent figures with captions. For such figures or tables, we recommend using

```
1 #align(center, image("{SOURCE}")) //figure
2 #align(center)[#table({...}]] //table
```

rather than `#figure` command to insert. This is because `#figure` is mainly designed for floating figures, and its default behavior may not match the requirements for in-paragraph figures or tables.

9.2. Standalone figures and tables

We suggest adding captions to all standalone figures and tables. For such figures, we provide a customized `#figure` function. Compared to Typst's default `figure`, it has the following features:

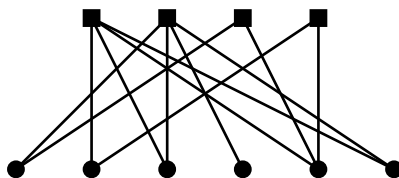


Figure 3. This figure is placed at the top of the page. Note that in Typst, the numbering of floating figures depends on their position in the source code, not their position in the generated document.

1. If the caption is short (fits in one line), it is centered; if it is long (spans multiple lines), it is left-aligned. See Figure 1 and Figure 2. This is implemented by detecting the number of lines in the caption, following the approach in [Typst Examples Book: Multiline detection](#).
2. The caption label (e.g., “Figure 1”, “Table 1”) is bold.
3. The caption font size is slightly smaller than the body text, controlled by `caption-font-size`.
4. There is vertical space of 2 em above and below the figure to separate it from the main text.

Features 3 and 4 are intended to avoid mixing of the caption and the body.

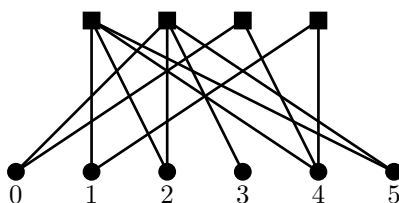


Figure 1. Short caption is centered.

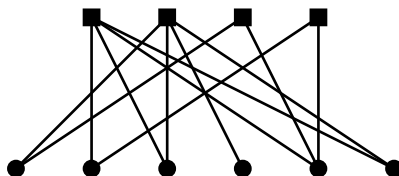


Figure 2. If the caption is longer than one line, it is left-aligned. This is a bipartite graph with 4 nodes on the top, 6 nodes on the bottom, and each top node is connected to an even number of bottom nodes.

Contents after a figure becomes a new paragraph with first-line indentation. Due to the difference of the font size and a vertical separation, the text does not visually blend with a long figure caption.

One can also place a figure at a fixed position on a page rather than position relative to texts. See Figure 3. Note that in Typst, the numbering of floating figures depends on their position in the source code, not their position in the generated document. This ensures that figure numbering follows the logical flow of the text, which is reasonable. However, if a document mixes figures placed in-flow and figures floated to the top or bottom of the page, the resulting figure numbers may appear out of order. Our suggestion is to always place figures at the top or bottom of the page for consistency.

9.3. Texts in figures and tables

The font size of texts in tables is the same as `body-font-size`. For figures, since this template is primarily intended for academic notes and assumes users will use the `cetz` package for drawing, the recommended font size for text inside figures matches the caption font size. This template customizes the default text size in a figure and therefore in `canvas` from the `cetz` package to align with `caption-font-size`. See Figure 1 for an example.

10. Bibliography and citation

The style of bibliography and citations can be set through the `bib-style` parameter, which is a dictionary mapping language codes to bibliography styles. For example, for English documents, the default is `"springer-mathphys"`, and for Chinese documents, it is `"gb-7714-2015-numeric"`. You can customize these styles as needed by modifying the `bib-style` dictionary when calling `hetvid.with`.

Please refer to [7] for available bibliography styles.

11. Designing philosophy and goals

The design philosophy of this template is to provide a lightweight, clear, and aesthetically pleasing environment for writing scientific notes. The main goals are:

- **Simplicity:** Avoid unnecessary complexity in both appearance and usage. The template adheres to common typesetting conventions and minimizes decorative elements, relying on indentation and spacing to separate content.
- **Readability:** Prioritize legibility by using appropriate font choices, sizes, and weights. The template ensures that headings, body text, captions, and mathematical content are visually distinct yet harmonious.
- **Consistency:** Maintain consistent formatting for similar elements (e.g., headings, theorems, figures) throughout the document, reducing cognitive load for both writers and readers.
- **Customizability:** Allow users to easily adjust key parameters (fonts, colors, spacing, numbering) to suit their needs, while providing sensible defaults for most use cases.
- **Academic focus:** Support features commonly needed in scientific writing, such as theorem environments, equation numbering, citation management, and figure/table handling.

By following these principles, the template aims to help users focus on content creation without being distracted by formatting issues.

The name `hetvid` of this template derives from the Latin transcription of Sanskrit word *hetuvidyā*. *Hetuvidyā* is the Sanskrit term for Buddhist logic and debate, introduced to China by the monk Xuanzang.

Bibliography

1. Kitaev, A.Y.: Quantum Computations: Algorithms and Error Correction. Russian Mathematical Surveys. 52, 1191–1249 (1997). <https://doi.org/10.1070/RM1997v052n06ABEH002155>
2. Kitaev, A.: Anyons in an Exactly Solved Model and Beyond. Annals of Physics. 321, 2–111 (2006). <https://doi.org/10.1016/j.aop.2005.10.005>
3. Kitaev, A.: Almost-Idempotent Quantum Channels and Approximate C^* -Algebras, (2025)
4. Witten, E.: Notes on Some Entanglement Properties of Quantum Field Theory. Reviews of Modern Physics. 90, 45003 (2018). <https://doi.org/10.1103/RevModPhys.90.045003>
5. Witten, E.: A Mini-Introduction To Information Theory. La Rivista del Nuovo Cimento. 43, 187–227 (2020). <https://doi.org/10.1007/s40766-020-00004-5>
6. Witten, E.: Introduction to Black Hole Thermodynamics, (2025)
7. Bibliography Function – Typst Documentation