

# LuaTeX-CN Documentation

— High-Quality Classical Chinese Typesetting —

Auto-generated from GitHub Wiki

## Table of Contents

---

[LuaTeX-CN Wiki](#)

[Installation Guide](#)

[Quick Start](#)

[Examples](#)

[Templates and Customization](#)

[Fonts](#)

[Features Overview](#)

[Modern Vertical Books](#)

[Punctuation System](#)

[Judou \(Punctuation Modes\)](#)

[Interlinear & Side Notes](#)

[Annotations & Marginal Notes](#)

[Correction & Decoration](#)

[Elevation \(Taitou\)](#)

[Textbox](#)

[Seals](#)

[Debug Mode](#)

[Changelog](#)

[Development Documentation](#)

[Release Process](#)

# LuaTeX-CN Wiki

---

Welcome to the LuaTeX-CN Wiki. This site provides comprehensive documentation for users and developers to help you create high-quality classic Chinese typesettings using LuaTeX-CN.

**Current Version:** [v0.3.0](#) | **CTAN:** [v0.1.1](#)

## Quick Access

---

- [🚀 Quick Start](#) - Get started with your first project in 5 minutes
- [📦 Installation](#) - Install LuaTeX-CN on your system
- [📖 Examples](#) - Explore "Records of the Grand Historian" (Shiji) and "Dream of the Red Chamber"

## Documentation

---

### 1. Fundamentals

- [Templates & Format](#) - Understand the "Banxin" (Page Center) concept
- [Font Setup](#) - Config Chinese fonts and fallback chains
- [Layout Control](#) - Chapters, TOC entries, and fine-grained layout control

### 2. Document Classes

- [guji \(Ancient Books\)](#) - Traditional vertical typesetting (default class)
- [cn-vbook / tw-vbook \(Modern Books\)](#) - Modern vertical books with punct squeeze and footnotes

### 3. Punctuation & Judou

- [Punctuation System](#) - Modern punct styles (mainland/Taiwan), squeeze, kinsoku
- [Judou](#) - Classical Chinese punctuation (judou/plain/normal modes)

## 4. Specialized Commands

- [Interlinear & Side Notes](#) - Detailed usage of jiazhu and side notes
- [Annotations & Marginal Notes](#) - Floating annotations and marginal notes (MeiPi)
- [Decorations](#) - Underlines, book titles, and correction marks
- [Elevation](#) - Honorific elevation formatting

## 5. Advanced Features

- [Text Boxes](#) - Absolute positioning and adaptive containers
  - [Seals](#) - Adding and positioning electronic seals
  - [Debugging](#) - Grid visualization and layout debugging
- 

## Developers

---

For information on contributing or understanding the internal architecture:

- [Development Guide](#)
  - [Release Process](#)
- 

## Contact

---

- **Contact:** Sheldon Li
- **Email:** sheldonli.dev@gmail.com

# Installation Guide

---

This page describes how to install the `luatex-cn` package.

## Method 1: Via CTAN/Package Manager (Recommended)

---

Install using your TeX distribution's package manager:

```
# TeX Live (Windows/Linux)
tlmgr install luatex-cn

# MiKTeX
# Open MiKTeX Console → Packages → Search for luatex-cn
```

⚠ **Note:** The CTAN version may lag behind GitHub. We recommend using GitHub Release.

## Method 2: Manual Installation from GitHub Release

---

1. **Download:** Go to [GitHub Releases](#) and download the latest `luatex-cn-tex-v*.zip`

2. **Locate texmf directory:**

- **Windows:** `C:\Users\\texmf`
- **macOS/Linux:** `~/texmf`

3. **Place files:** Put the contents of `tex/` into:

```
texmf/tex/latex/luatex-cn/
```

4. **Refresh database:**

```
texhash
```

## Method 3: Direct Usage

---

Place the contents of `tex/` in the same directory as your `.tex` file and compile directly.

## Verify Installation

---

Create a test file `test.tex`:

```
\documentclass[四库全书]{guji}
\begin{document}
\begin{正文}
天地玄黄，宇宙洪荒。
\end{正文}
\end{document}
```

Run:

```
lualatex test.tex
```

If compilation succeeds, installation is complete.

## System Requirements

---

- LuaTeX (TeX Live 2024+ recommended)
- `luaotfload` and `fontspec`

## Font Requirements

---

We recommend installing these fonts for best results:

- Noto Serif CJK SC (Source Han Serif)
- FangSong or KaiTi (for traditional book effects)

# Quick Start

---

This page helps you quickly get started with `luatex-cn` for traditional Chinese book typesetting.

## Basic Template

---

```
\documentclass[四库全书]{guji}
% Optional: specify font
% \setmainfont{Noto Serif SC}

\title{钦定四库全书}
\chapter{史记\\卷一}

\begin{document}
\begin{正文}

This is sample vertical text for ancient book typesetting.

\end{正文}
\end{document}
```

## Common Commands

---

### Interlinear Notes (Double-line Small Text)

```
Main text\\夹注{This is interlinear note content, auto-balances}continued text.
```

### Side Notes

```
Main text\\侧批[yoffset=10pt]{This is side note content}continued text.
```

### Paragraph Indentation

```
\\begin{段落}[indent=2]
天地玄黄\\ \\ \\
```

```
宇宙洪荒  
\end{段落}
```

## Spaces

```
Need\\空格[2]two spaces here.
```

## Annotation Box

```
\\批注[x=5cm, y=2cm, height=6, color={1 0 0}]{Annotation content}
```

## Seal

```
\\印章[page=1, xshift=2cm, yshift=5cm]{seal.png}
```

## Force Column Break

```
End of this column\\\\  
Start of new column
```

## Metadata Setup (vo.2.0+)

```
\title{Book Title}  
\publisher{Publisher Name}  
\ChapterMarker{Volume 1}
```

Or using the setup command:

```
\metadataSetup{  
  title = Book Title,  
  publisher = Publisher Name  
}
```

## Debug Mode (vo.2.2+)

```
\LtcDebugOn % Enable debug ruler, shows grid coordinates  
\LtcDebugOff % Disable debug mode
```



## Preset Styles

---

Choose preset styles via document class options:

Option	Description
四库全书	Qing Dynasty Siku Quanshu style, single fishtail
红楼梦甲戌本	Manuscript style, no fishtail

```
\\documentclass[四库全书]{guji}
% or
\\documentclass[红楼梦甲戌本]{guji}
```

## Next Steps

---

- See [Examples](#) for more complex usage
- Read [Changelog](#) for latest features

# Examples

---

This directory contains multiple usage examples for the `luatex-cn` package, demonstrating different features from traditional ancient book typesetting to modern vertical layouts.

## Example List

---

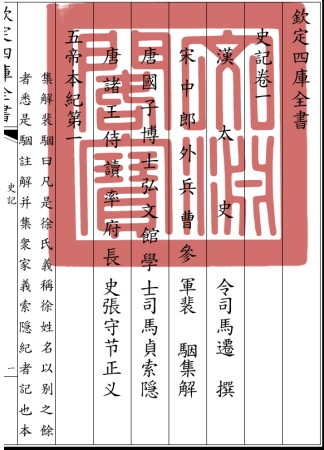
### 1. Records of the Grand Historian - Basic Annals of the Five Emperors

**Features:** Demonstrates highly complex ancient book typesetting.

**Capabilities:**

- Absolutely positioned red seals (overlying text)
- Customized page center text with single fishtail
- Complex interlinear note typesetting
- Traditional black silk frame and indentation

<b>Black &amp; White</b>	<b>Color Preview</b>
	Shiji Color

Black & White	Color Preview
	

[View Source](#)

## 2. Records of the Grand Historian - Table of Contents

**Features:** Following Qing Dynasty "Siku Quanshu" Northern Pavilion style.

**Capabilities:**

- Standard ancient book table of contents layout
- "Eight lines, twenty-one characters" format restriction
- Typical white mouth, double borders on all sides, single fishtail format

[View Source](#)

## 3. Dream of the Red Chamber - Jiaxu Manuscript

**Features:** Simulates manuscript/annotated copy style.

**Capabilities:**

- **Side and marginal notes:** Edge annotations using the sidenote system
- **Double-column small text:** Double-line small annotations in main text

- No fishtail page center, bottom page numbers

Page 2 (Punctuation)	Page 1 (Marginal Notes)
<p>仙鄉總有藝談道富覺石骨 師裏心補那說實打遠格不 聽受攜天性人道人動後高 畢攜帶濟世大自便便談凡 齊歡享弟通間師根心凡快 慈幾子世間祭弟粗也到神 笑當入材利二叢子也到想 道當入材利二叢子也到想 善佩入紅物華心已不人得 善洪濟人仙形道能見禮吐 哉恩那富貴場之弟子過一 紅劫富如榮發非凡質聞這 塵中不溫發發品資向這 有也溫發發品資向這</p>	<p>第一 列位 看官 你道 這書 從何 而來 將此 根由 說與 衆人 聽之 且道 這書 之妙 不可 言喻 其理 之深 不可 測度 其文 之雅 不可 及也 其言 之簡 不可 及也 其意 之遠 不可 及也 其情 之切 不可 及也 其理 之深 不可 測度 其文 之雅 不可 及也 其言 之簡 不可 及也 其意 之遠 不可 及也 其情 之切 不可 及也</p>

[View Source](#)

## 4. Modern Vertical Book

**Features:** Clean modern vertical layout style.

**Capabilities:**

- Demonstrates pure vertical text support without traditional book elements
- Suitable for modern literature or reports requiring vertical typesetting

Modern vertical books are not currently the project focus; more features will be added later.

[View Source](#)

*Note: All ancient book examples are digital reconstructions based on historical document scans, intended to demonstrate `luatex-cn`'s adaptability to various complex vertical typesetting scenarios.*

# Templates and Customization

---

`luatex-cn` provides a flexible template system with built-in and custom templates.

## Built-in Templates

---

Template Name	Document Class	Description
四库全书	<code>guji</code>	Classic official book style (B&W)
四库全书彩色	<code>guji</code>	Classic official book style (Color)
红楼梦甲戌本	<code>guji</code>	Manuscript style with side/top notes
default	<code>cn-vbook</code> / <code>tw-vbook</code>	Default modern vertical style (Taiwan punct)
中华书局	<code>cn-vbook</code>	Zhonghua Shuju modern style (Mainland punct)

## Using Templates

---

Specify the template name in document class options:

```
% guji templates
\documentclass[四库全书彩色]{guji}
\documentclass[红楼梦甲戌本]{guji}

% cn-vbook / tw-vbook templates (v0.2.7+)
\documentclass{cn-vbook}           % Default template
\documentclass[中华书局]{cn-vbook} % Zhonghua Shuju template
```

# Custom Templates

---

## Method 1: Create Configuration File

### guji Template

Create a `luatex-cn-guji-<template-name>.cfg` file in the `configs/` directory:

```
\gujiSetup{ template = default }

\pageSetup{
  paper-width = 1077.2pt,
  paper-height = 1077.2pt,
}

\contentSetup{
  n-column = 12,
  font-size = 30pt,
  border = true,
}

\banxinSetup{
  upper-yuwei = true,
  lower-yuwei = true,
}

\endinput
```

### cn-vbook / tw-vbook Template (vo.2.7+)

Create a `luatex-cn-book-<template-name>.cfg` file in the `configs/` directory:

```
\pageSetup{
  paper-width = 148mm,
  paper-height = 210mm,
  page-number-style = digits,
}

\contentSetup{
  font-size = 11pt,
  line-spacing = 13pt,
  n-column = 0,
  border = false,
}

\punctSetup{punct-style=taiwan}

\endinput
```

## Method 2: Direct Configuration in Document

```

\documentclass{guji}

\gujiSetup{
  book-name = My Book Title,
  chapter-title = Chapter One,
}

\contentSetup{
  n-column = 10,
  font-size = 24pt,
  border = true,
}

\begin{document}
% ...
\end{document}

```

## Configuration Reference Table

Command	Parameter	Description	Default
\gujiSetup	book-name	Book name	-
	chapter-title	Chapter title	-
	template	Template name	default
\contentSetup	n-column	Columns per half page	12
	font-size	Font size	30pt
	line-spacing	Line spacing	45pt
	n-char-per-col	Characters per column	18
	border	Show border	true
	layout-mode	Layout mode (v0.2.7+)	grid
	\pageSetup	paper-width	Paper width
	paper-height	Paper height	-
	margin-*	Margins	-
	page-number-style	Number style (v0.2.7+)	none
\banxinSetup	banxin-upper-ratio	Upper section ratio	0.18

Command	Parameter	Description	Default
	<code>upper-yuwei</code>	Upper fish tail	<code>true</code>
<code>\punctSetup</code>	<code>punct-style</code>	Punct style (vo.2.7+)	<code>taiwan</code>
	<code>punct-squeeze</code>	Squeeze (vo.2.7+)	<code>true</code>
	<code>kinsoku</code>	Line-breaking rules (vo.2.7+)	<code>true</code>

---

🔗 Next: See [Debug Mode](#) or [Features](#)



# Fonts

---

`luatex-cn` provides intelligent cross-platform font detection and a powerful multi-font fallback mechanism to gracefully handle missing characters (rare glyphs).

## 1. Automatic Font Detection

---

By default, the `guji` (or `cn-vbook` / `tw-vbook`) document class automatically selects the most suitable installed fonts based on your operating system:

- **Windows:** Prioritizes SimSun, SimHei, etc.
- **macOS:** Prioritizes Songti SC, PingFang SC, etc.
- **Linux:** Prioritizes Fandol series or Noto CJK series.

To manually trigger and apply the auto-detected font:

```
\ApplyAutoFont
```

## Chinese Font Name Support (v0.2.2+)

Since v0.2.2, font detection supports simplified and traditional Chinese font names:

```
\设置字体{宋体}           % Simplified Chinese name
\设置字体{標楷體}        % Traditional Chinese name
\设置字体{思源宋体}      % Open source font Chinese name
```

The system automatically maps Chinese font names to corresponding system fonts.

## 2. Font Family & Fallback

---

v0.1.5 introduces the concept of **Font Family**. By setting a font family, the system establishes a "fallback chain". When the main font lacks a specific character (e.g., a rare CJK ideograph), it automatically looks for it in the next font in the chain.

## `\setFontFamily`

This command accepts a comma-separated list of fonts. The first font in the list becomes the main font, and the rest serve as fallbacks.

```
% Sets "Source Han Serif SC" as the main font.  
% If a character is missing, it tries "SimSun", then "SimHei".  
\setFontFamily{Source Han Serif SC, SimSun, SimHei}
```

Chinese Alias:

```
\设置字体族{Source Han Serif SC, SimSun, SimHei}
```

To define a named fallback chain for custom usage:

```
% Define a chain named "mybackup"  
  
% Apply this chain to a specific font  
\setmainfont{MyCustomFont}[RawFeature={fallback=mybackup}]
```

## 3. Standard Font Setting

---

If you only need to set a single font without fallback logic:

```
\setmainfont{KaiTi}  
% or  
\设置字体{KaiTi}
```

[!NOTE] For vertical layout, `luatex-cn` generally handles vertical features ( `RawFeature={+vert}` ) automatically.

---

🔗 Next: [Correction](#) or [Features](#).

# Features Overview

---

`luatex-cn` provides a rich set of features for ancient Chinese book typesetting, ranging from core vertical engines to various decorative elements.

## 1. Core Typesetting Engine

---

A grid-based layout system built on LuaTeX for high-precision vertical text flows.

- **RTT Direction:** Native support for top-to-bottom, right-to-left direction.
- **Grid Positioning:** Characters are precisely aligned to a defined grid.
- **Pagination & Column Flow:** Handles complex page splitting and column transitions automatically.

## 2. Document Classes

---

- **guji:** Traditional ancient book typesetting with banxin, fishtail, silk frames.
- **cn-vbook / tw-vbook** (v0.2.7+): Modern vertical books with clean design, built-in footnotes and punctuation system.

## 3. Punctuation Processing

---

- **Punctuation System** (v0.2.7+): Modern punct squeeze, kinsoku, vertical quote replacement with mainland/Taiwan styles.
- **Judou:** Traditional punctuation marks with three modes (normal/judou/none) switchable at any time.

## 4. Traditional Book Elements

---

Full support for visual elements required by traditional binding:

- **Banxin & Yuwei:** Support for single/double fishtails, black silk frames, etc.
- **Split Page:** Support for electronic reading and traditional "tube page" binding.
- **Border System:** Double borders, inner borders, and customizable thickness.

## 5. Annotation & Revision System

---

Multiple ways to add comments to the text:

- **Interlinear Notes**: Double-line small text within main body with auto-balancing.
- **Side Notes**: Vertical annotations between text columns.
- **Annotations**: Floating annotation boxes placed anywhere on the page.
- **Marginal Notes** (v0.2.3+): Top-of-page annotations.
- **Footnotes** (v0.2.7+): Endnotes and page-bottom notes with `lujiao/circled` numbering.
- **TOC Entries**: `\条目` (v0.2.5+) with hierarchical indentation.

## 6. Layout Control & Commands

---

- **Force Column**: `\列` (or `\Column`) forces content to start in a new column.
- **Last Column**: `\末列` (or `\LastColumn`) places content in the last column.
- **Flexible Widths**: `\行[width=2cm]{...}` (v0.2.7+) for variable-width columns.
- **Page Breaks**: Supports standard `\newpage` and `\clearpage`.
- **Natural Layout** (v0.2.7+): `layout-mode=natural` for non-uniform spacing.

## 7. Decorations & Utilities

---

- **Seals**: Add collection or personal seals anywhere with opacity support.
- **Fonts**: Auto-detection, font families, and fallback chains. Supports simplified/traditional Chinese font names since v0.2.2.
- **Textbox**: Grid-based layout with auto-compression and filling.
- **Correction & Decoration**: Correction marks and character decoration.
- **Elevation** (v0.2.6+): Honorific elevation formatting (single/double/triple/flat).

## 8. Architecture & Debug Features (v0.2.0+)

---

- **Style Registry**: Multi-attribute style storage with cross-page preservation.
- **Metadata System**: Centralized book metadata management.
- **Debug Mode**: Grid visualization and layout debugging.
- **Plugin Architecture**: Standardized plugin API with three-phase interfaces.

## 9. Template System

---

Preset classical styles available out-of-the-box. See [Templates](#).

- [四库全书](#) (Siku Quanshu): Qing Dynasty imperial style.
  - [四库全书彩色](#) : Colored version.
  - [红楼梦甲戌本](#) : Manuscript style.
  - [中华书局](#) (vo.2.7+): Modern mainland-style vertical layout.
- 

 Next Steps: Check [Templates](#) | [Debug Mode](#) | [Examples](#)

# Modern Vertical Book Classes (v0.2.7+)

---

luatex-cn provides two document classes for modern vertical Chinese book typesetting. Unlike `guji` (ancient book typesetting), they use a clean, modern design: no banxin, no fishtail, no silk frames — suitable for contemporary publications that need vertical layout.

Class	Punct Style	Description
<code>cn-vbook</code> (v0.2.8+)	Mainland	Simplified Chinese vertical books
<code>tw-vbook</code> (v0.2.8+)	Taiwan	Traditional Chinese vertical books
<code>ltc-book</code>	Taiwan	Legacy name, still works

## Basic Usage

---

```
\documentclass{cn-vbook}
\setmainfont{Noto Serif SC}

\begin{document}
\begin{正文}
天地玄黄，宇宙洪荒。日月盈昃，辰宿列张。
\end{正文}
\end{document}
```

## Template System

---

Option	Description	Punct Style
(none)	Default, 16-kai (170x240mm), 12pt font	Taiwan
<code>中华书局</code>	Zhonghua Shuju style, 15pt font, footnotes enabled	Mainland

```
\documentclass{cn-vbook} % Default
\documentclass[中华书局]{cn-vbook} % Zhonghua Shuju
```

## Differences from guji

Feature	cn-vbook / tw-vbook	guji
Design	Modern, clean	Traditional ancient
Banxin/Yuwei	No	Yes
Borders/Silk frames	No	Yes
Layout mode	Free column width ( <code>n-column=0</code> )	Fixed grid ( <code>n-column=8</code> )
Default punct	Modern (normal)	Configurable judou/normal
Footnotes	Built-in	Manual loading
Page numbers	Auto vertical	In banxin center

## Punctuation Style

`cn-vbook` / `tw-vbook` enable the modern punctuation system by default. See [Punctuation System](#).

```
\documentclass{cn-vbook}
\punctSetup{punct-style=mainland} % Override to mainland style
```

Or use `\documentclass[中华书局]{cn-vbook}` which automatically sets mainland style.

You can also switch to judou mode: `\开启句读模式`

## Footnote System

Built-in footnote support with endnote and page-bottom modes.

```
\脚注设置{number-style=lujiao, indent=1em}
天地玄黄\脚注{This is a footnote}。
\输出脚注 % Output at paragraph end
```

## Configuration ( `\footnoteSetup` / `\脚注设置` )

Parameter	Description	Default	Values
<code>mode</code>	Mode	<code>endnote</code>	<code>endnote</code> , <code>page</code>
<code>number-style</code>	Style	<code>lujiao</code>	<code>lujiao</code> (【一】), <code>circled</code> (①)
<code>separator</code>	Separator	<code>blank</code>	<code>blank</code> , <code>none</code>
<code>font-size</code>	Size	<code>0.8em</code>	Any dimension
<code>indent</code>	Indent	<code>1em</code>	Any dimension

## Commands

Command	Chinese	Description
<code>\Footnote{...}</code>	<code>\脚注{...}</code>	Add footnote
<code>\FlushFootnote</code>	<code>\输出脚注</code>	Output endnotes
<code>\footnoteSetup{...}</code>	<code>\脚注设置{...}</code>	Configure
<code>\ResetFootnoteCounter</code>	<code>\重置脚注计数</code>	Reset counter

## Page Numbers

Auto vertical page numbers at lower-left. Configure via `\pageSetup` :

Parameter	Default
<code>page-number-style</code>	<code>digits</code> (also: <code>chinese</code> , <code>arabic</code> , <code>none</code> )
<code>page-number-font-size</code>	<code>9pt</code>
<code>page-number-x</code>	<code>3mm</code>
<code>page-number-y</code>	<code>8mm</code>

## Custom Templates

Create `configs/luatex-cn-book-<name>.cfg` with `\pageSetup`, `\contentSetup`, `\punctSetup`, then use `\documentclass[name]{cn-vbook}`.



---

Next: [Punctuation](#) | [Features](#) | [Templates](#)

# Punctuation System (v0.2.7+)

---

`luatex-cn` provides a complete modern punctuation processing system with squeeze, kinsoku (line-breaking rules), vertical quote replacement, and mainland/Taiwan style support.

## Punctuation System vs Judou System

---

`luatex-cn` has two mutually exclusive punctuation mechanisms:

System	Use Case	punct-mode	Description
<b>Punctuation</b> (this page)	Modern books	<code>normal</code> (default)	Squeeze, kinsoku, quote replacement
<b>[Judou ](#en-judou)</b>	Ancient books	<code>judou</code> or <code>none</code>	Convert punct to circles/dots or remove

When `punct-mode=normal`, the punctuation system is active; when `punct-mode=judou` or `none`, the judou system takes over and the punctuation system is automatically disabled.

## Punctuation Styles

---

### Taiwan Style (Default)

All punctuation marks are centered within the grid cell.

```
\punctSetup{punct-style=taiwan}
% or
\台湾标点风格
```

### Mainland Style

Period (。 ) and comma (， ) marks are offset toward the upper-right corner of the grid cell, matching mainland Chinese publishing conventions.

```
\punctSetup{punct-style=mainland}
% or
\大陆标点风格
```

## Configuration

---

Configure punctuation behavior via `\punctSetup` :

```
\punctSetup{
  punct-style = taiwan,    % Style: taiwan or mainland
  punct-squeeze = true,   % Squeeze: compress punct to half-grid
  kinsoku = true,        % Kinsoku: prevent misplaced punct
  punct-hanging = false, % Hanging (in development)
}
```

Parameter	Description	Default	Values
<code>punct-style</code>	Punctuation style	<code>taiwan</code>	<code>taiwan</code> , <code>mainland</code>
<code>punct-squeeze</code>	Squeeze punctuation	<code>true</code>	<code>true</code> , <code>false</code>
<code>kinsoku</code>	Line-breaking rules	<code>true</code>	<code>true</code> , <code>false</code>
<code>punct-hanging</code>	Hanging punctuation	<code>false</code>	<code>true</code> , <code>false</code>

## Punctuation Squeeze

---

When enabled, periods, commas and other marks are compressed to half the grid cell height, producing a tighter, more elegant layout.

## Kinsoku (Line-Breaking Rules)

---

Follows Chinese typesetting standards (CLREQ) to prevent marks from appearing in inappropriate positions:

- **Line-start forbidden:** Closing brackets, periods, commas, middle marks cannot appear at column start
- **Line-end forbidden:** Opening brackets cannot appear at column end

## Vertical Quote Replacement

---

The system automatically replaces horizontal bracket/quote forms with Unicode Vertical Presentation Forms (e.g., 「 $\rightarrow$   $\neg$ ,」  $\rightarrow$   $\neg$ , ( $\rightarrow$   $\curvearrowright$ ,)  $\rightarrow$   $\curvearrowright$ , 《 $\rightarrow$   $\curvearrowright$ ,》  $\rightarrow$   $\curvearrowright$ ,  $\rightarrow$  | , ... $\rightarrow$   $\vdots$ ).

## Using with cn-vbook / tw-vbook

---

The `cn-vbook` / `tw-vbook` document classes enable the punctuation system by default. Choose styles via templates:

```
% Default template: Taiwan style
\documentclass{cn-vbook}

% Zhonghua Shuju template: Mainland style
\documentclass[中华书局]{cn-vbook}
```

## Using with guji

---

The `guji` class also supports the punctuation system. When `punct-mode=normal`, it activates automatically:

```
\documentclass[四库全书]{guji}
\正常标点模式 % Use modern punctuation (not judou)
\punctSetup{punct-style=mainland}
```

## Switching Between Modes

---

You can dynamically switch between the punctuation and judou systems:

```
\正常标点模式 % Modern punctuation
\句读模式 % Judou mode (circles/dots)
\无标点模式 % Plain text (no punctuation)
```

## Technical Details

---

Implemented via `luatex-cn-core-punct.lua` in three stages:

1. **Flatten**: Classify punctuation characters and replace with vertical forms

2. **Layout:** Apply squeeze and kinsoku processing
  3. **Render:** Apply mainland/Taiwan style position offsets
- 

🔗 Next: [Judou](#) | [Modern Vertical](#) | [Features](#)

# Judou (Punctuation Modes)

---

**Judou** (jù dòu) refers to the traditional Chinese system of punctuation and pausing marks. `luatex-cn` can automatically convert modern punctuation into traditional styles suitable for ancient book typesetting.

## Punctuation Mode Overview

---

`luatex-cn` provides three punctuation modes ( `punct-mode` ), switchable via `\judouSetup` or convenience commands:

Mode	Description	System
<code>normal</code> (default)	Modern full-width punctuation with squeeze/kinsoku	[Punctuation ](#en-punctuation)
<code>judou</code>	Convert to traditional Ju/Dou marks (circles/dots)	Judou system
<code>none</code>	Remove all punctuation (plain text / "Baiwen")	None

**Note:** The punctuation system and judou system are **mutually exclusive**. When `punct-mode=normal`, the punctuation system handles squeeze/kinsoku; when `punct-mode=judou` or `none`, the judou system takes over.

## What is Judou Mode?

---

Traditional ancient books typically do not use modern full-width punctuation marks. Instead, they use small circles ("Ju") or dots ("Dou") attached to the characters.

- **Ju** (句): Corresponds to periods (.), question marks (?), and exclamation marks (!). Rendered as a small circle (○).
- **Dou** (读): Corresponds to commas (,), semicolons (;), and enumeration commas (、). Rendered as a small dot or tick (●).

## Supported Punctuation (v0.2.1+)

Since v0.2.1, the range of supported punctuation has been extended to include:

- Basic punctuation: periods, commas, enumeration commas, semicolons, colons
- Paired punctuation: quotation marks (「」 『』 """"), book title marks (《》 〈〉 ), parentheses
- Special punctuation: dashes, ellipses, interpuncts, etc.

## Usage

---

### 1. Switching Punctuation Modes

```
% Enable judou mode
\JudouOn
\开启句读模式
\句读模式

% Return to modern punctuation
\JudouOff
\关闭句读模式
\正常标点模式

% Plain text mode (no punctuation)
\无标点模式
\开启无标点模式
```

### 2. Using with Different Document Classes

#### guji (Ancient Books)

```
\documentclass[四库全书]{guji}
\开启句读模式 % Use traditional judou

\begin{document}
\begin{正文}
人之初，性本善。
% Commas become dots, periods become circles at bottom-right
\end{正文}
\end{document}
```

## cn-vbook / tw-vbook (Modern Books)

```
\documentclass{cn-vbook}
% cn-vbook defaults to modern punctuation (punct-mode=normal)

\begin{document}
\begin{正文}
天地玄黄，宇宙洪荒。 % Modern punct with squeeze/kinsoku

\句读模式
寒来暑往，秋收冬藏。 % Switch to judou
\end{正文}
\end{document}
```

### 3. Dynamic Switching

```
\句读模式
人之初，性本善。 % Judou marks

\正常标点模式
现代文：他说：“一二三。” % Modern punctuation

\无标点模式
子曰学而时习之不亦说乎 % Plain text (Baiwen)
```

## Parameters

Fine-tune Judou appearance via `\judouSetup` :

Key	Description	Default
<code>punct-mode</code>	Punctuation mode	<code>normal</code>
<code>judou-color</code>	Color of the marks	<code>red</code>
<code>judou-size</code>	Size of the marks	<code>1em</code>
<code>judou-pos</code>	Position relative to character	<code>right-bottom</code>

```
\judouSetup{
  punct-mode = judou,
  judou-color = black,
  judou-size = 0.8em,
}
```



## Command Reference

---

Command	Chinese Alias	Description
<code>\JudouOn</code>	<code>\开启句读模式</code>	Enable judou mode
<code>\JudouOff</code>	<code>\关闭句读模式</code>	Disable judou mode
<code>\NormalPunctuationMode</code>	<code>\正常标点模式</code>	Switch to modern punct
<code>\JudouPunctuationMode</code>	<code>\句读模式</code>	Switch to judou mode
<code>\NonePunctuationMode</code>	<code>\无标点模式</code>	Switch to plain text
<code>\judouSetup{...}</code>	-	Configure judou parameters

## Technical Details

---

Implemented via `luatex-cn-guji-judou.lua`:

- 1. Identification:** Scans node stream for punctuation Unicode code points.
- 2. Conversion:** In `judou` mode, punctuation nodes are removed and replaced by a decorate marker node attached to the preceding CJK character.
- 3. Rendering:** The renderer draws the circle/dot at the specified anchor point (usually bottom-right) of the character.

---

🔗 Next: [Punctuation](#) | [Features](#)

# Interlinear & Side Notes

---

`luatex-cn` offers a comprehensive annotation system, including in-text interlinear notes and marginal side notes.

## Interlinear Notes (Jiazhu / 夹注)

---

Commonly seen in ancient books as double-line small characters between main body text.

### Usage

```
Main text\\夹注{This is interlinear note content. If it is long, it will automatically split into two col
```

### Key Features

- **Auto-Balancing Algorithm:** The system calculates the remaining space in the current column and splits the note content evenly into right and left sub-columns.
- **Cross-Column/Page Flow:** When a note doesn't fit in one column, it flows to the next column or page automatically.
- **Alignment Control:** Supports `left`, `right`, `center`, `inward`, and `outward` (default) alignment modes.
- **Balance Control (vo.2.6+):** Use `auto-balance=false` to disable the automatic balancing algorithm. Useful for manual line breaks or replicating specific historical manuscript styles.

### Jiazhu Configuration

```
\jiazhuSetup{
  font-size = 12pt,
  font = {FandolSong-Regular},
  font-color = red,
  auto-balance = true, % Whether to auto-balance sub-column lengths
  align = outward     % Alignment mode
}
```

## Side Notes (SideNode / 侧批)

---

Small annotations placed between text columns or at page margins.

### Usage

Three syntax options:

```
% Basic usage
Main text\侧批{This is the side note content}continued text.

% Positional parameter (auto-converts to em)
Main text\侧批[5pt]{Note with adjusted height}continued text.
Main text\侧批[0.5]{Equivalent to 0.5em}continued text.

% Named parameters
Main text\侧批[yshift=10pt, color=red]{Both color and position}continued text.
```

### Parameters

- `yshift` : Vertical offset with positional parameter support. Pure numbers auto-convert to em units (e.g., `0.5` → `0.5em`).
- `color` : Annotation color (default: red).
- `font-size` : Font size for the note (default: 10pt).
- `grid-height` : Custom grid height for the note block (default: equals font size).
- `border-padding-top` : Top border padding (default: opt).
- `border-padding-bottom` : Bottom border padding (default: opt).

### Global Configuration

```
\sidenoteSetup{
  color = red,
  font-size = 19pt,
  grid-height = 20pt,
  yshift = 0em,
  border-padding-top = 0.5em,
  border-padding-bottom = 0.5em
}
```

### Aliases

- Setup command: `\sidenoteSetup` / `\CePiSetup` / `\侧批设置` (Simplified) / `\側批設置` (Traditional)

- Vertical shift parameter: `yshift` / 纵移 (Simplified) / 縱移 (Traditional)

## Cross-Page Color Preservation (v0.2.1+)

v0.2.1 introduced the style registry mechanism. Side note color settings now persist correctly across pages. Even when side note content spans multiple pages, colors remain consistent.

---

## Technical Implementation (For Developers)

---

- **Interlinear Notes:** Uses LuaTeX attributes to mark nodes. During layout, the system detects these attributes, pauses normal typesetting, and calls `core_textflow.lua` for column splitting and re-mapping to the grid. Migrated to style stack architecture since v0.2.1.
  - **Side Notes:** Managed by `core_sidenote.lua`. It captures the content as an independent box and calculates the offset based on the current reference character's coordinates. Supports Phase 2 style registry for color preservation since v0.2.1.
  - **Style Registry:** `luatex-cn-style-registry.lua` provides multi-attribute style storage, supporting deduplication and cross-page preservation for color, `font_size`, `grid_height`, and other attributes.
- 

🔗 Next Steps: Check [Annotations](#) or [Judou](#).

# Annotations (PiZhu / 批注)

---

The `\批注` command allows placing floating annotation boxes anywhere on the page, simulating post-publication notes or commentaries in ancient manuscripts.

## Usage

---

```
\批注[x=5cm, y=2cm, height=6, color={1 0 0}]{Annotation content}
```

## Parameters

---

- `x` : Horizontal position of the top-left corner (relative to page center).
- `y` : Vertical position of the top-left corner (relative to page center).
- `height` : Number of grid rows (height) allocated for the box.
- `color` : Color in RGB format (e.g., `{1 0 0}` for red).
- `font-size` : Font size inside the annotation.
- `grid-width` : Custom grid cell width for the box.
- `grid-height` : Custom grid cell height for the box.

## Features

---

- **Absolute Positioning**: Does not occupy main body space; overlaid directly at specified coordinates.
- **Auto-Columnizing**: If content exceeds the set `height`, it automatically starts a new column within the box.
- **Clip Prevention**: Position calculations account for Banxin (page center) to prevent content being cut off by borders.
- **Multi-column Support**: Internal columns can be precisely controlled via `n-cols`.
- **Independent Indentation** (v0.2.2+): Annotation content no longer inherits paragraph indentation from main text (fix #37).

## Technical Implementation

---

Under the hood, annotations are based on `core_textbox.lua`. Content is rendered into an independent vertical grid system. This produced box is then overlaid during the page rendering stage using `shipout` logic in `render_page.lua`.

---

# Marginal Notes (MeiPi / 眉批) v0.2.3+

---

The `\眉批` command places annotations at the top of the page (above the main text), a common format in traditional Chinese book annotations. Unlike `\批注`, MeiPi automatically calculates positions, arranging from right to left to avoid overlap.

## Usage

---

```
% Auto-positioning (recommended)
\眉批[height=5]{Marginal note content}

% Manual Y coordinate
\眉批[height=5, y=50pt]{Marginal note content}

% Fully manual positioning
\眉批[height=5, x=100pt, y=50pt]{Marginal note content}
```

## Parameters

---

- `height` : Number of grid rows (height) for the note, **required**.
- `x` : Horizontal position. Auto-calculated if not specified, arranging right-to-left.
- `y` : Vertical position. Auto-calculated if not specified, aligning bottom edge to main text top.
- `color` : Color, default red `{1 0 0}`.
- `font-size` : Font size, default 18pt.
- `grid-width` : Grid width, default 20pt.
- `grid-height` : Grid height, default 19pt.

## Features

---

- **Auto-arrangement** (v0.2.3+): Multiple MeiPi notes automatically arrange right-to-left. Spacing configurable via `\meipiSetup{spacing=...}`.
- **Smart Positioning**: Y coordinate auto-calculated so bottom edge aligns with main text top.

- **Cross-page Support** (v0.2.3+): MeiPi correctly appears on the page where its anchor is located, not the previous page.
- **Independent Indentation**: MeiPi content does not inherit paragraph indentation from main text.

## Configuration

---

```
\meipiSetup{
  spacing = 10pt, % Spacing between MeiPi notes
  gap = 0pt      % Gap between MeiPi bottom and main text top
}
```

## Technical Implementation

---

MeiPi is based on `guji/luatex-cn-guji-meipi.lua`. It inserts anchor nodes into the node list to record positions, then during layout phase determines which page the MeiPi should appear on based on the **first node with layout information after** the anchor.

---

🔗 Next Steps: Check [Seals](#) or [Judou](#).



# Correction & Decoration

---

`luatex-cn` provides tools for classical text correction and decoration, including correction marks, emphasis marks, and underlines (proper name marks / book title marks).

## Correction (GaiZi)

---

The `\fix` (or `\改`) command marks corrections in classical text editing. It retains the original character with a deletion mark (、) and places the replacement character at the bottom right.

### Usage

```
\fix{replacement}  
% or Chinese alias  
\改{替换字}
```

### Example

```
This has a \fix{correct}wrong text.
```

In this example, "wrong" is marked with a deletion symbol, and "correct" is placed at its bottom-right corner at 60% size.

### Technical Details

- **Deletion Mark:** Automatically overlays a deletion point (、) on the character.
- **Replacement Text:** Placed at bottom-right corner at 60% of original font size.
- **Non-destructive:** Does not affect the original grid layout.

---

## Text Decoration

---

Preset commands for marking text according to classical or modern standards.

## 1. Underline Marks

Command	Alias	Description	Use Case
<code>\Underline</code>	<code>\下划线</code>	Straight line on the left	General highlighting
<code>\专名号</code>	-	Straight line on the left	Proper names (Person, Place)
<code>\WavyUnderline</code>	<code>\波浪线</code>	Wavy line on the left	General highlighting
<code>\书名号</code>	-	Wavy line on the left	Book/Article titles

### Options ( [options] ):

- `color` : Line color (default: `red`).
- `offset` : Distance from text (default: `0.6em`).

### Example:

```
\专名号[color=black]{Confucius}
\书名号{The Analects}
```

**Technical Improvement (v0.2.6+):** Underlines have been migrated from glyph-based ( | ) to **PDF Graphics** implementation. This solves:

- Line discontinuity at the end of columns/pages.
- Strong coupling between line thickness/position and specific fonts.
- Support for use within sidenotes, jiazhu, and textboxes.

## Elevation (Taitou / 抬头) **v0.2.6+**

`luatex-cn` provides full support for "Taitou" (Elevation/Rising). See: [Elevation](#).

## 2. Emphasis Marks

Command	Alias	Description
<code>\EmphasisMark</code>	<code>\着重号</code>	Add small circles ( ° ) to the right of each char
<code>\圈点</code>	-	Same as above

## Example:

```
\EmphasisMark{Very Important}
```

---

## Generic Decoration

The `\decorate` (or `\装饰`) command allows overlaying arbitrary content on **each character**.

## Usage

```
\decorate[options]{content}
```

## Parameters

Key	Description	Default
<code>char</code>	Decoration character	<code>o</code>
<code>color</code>	Color	<code>red</code>
<code>xoffset</code>	Horizontal offset (xshift)	<code>0pt</code>
<code>yoffset</code>	Vertical offset (yshift)	<code>0pt</code>
<code>scale</code>	Scale factor	<code>1.0</code>
<code>font-size</code>	Force specific font size	<code>auto</code>

## Example

```
% Add a red circle on each character in "important"  
\decorate[char=o, scale=0.8, color=red]{important}
```

---

➤ Next: [Textbox](#) or [Features](#).

# Elevation (Taitou / 抬头) v0.2.6+

---

`luatex-cn` provides full support for "Taitou" (Elevation/Rising), a traditional formatting used in ancient Chinese texts to show respect by starting a new line and raising the text height.

## Usage

---

Command	Effect	Description
<code>\抬头[N]</code>	New line, raised by N cells	Core command, N can be pos/neg
<code>\平抬</code>	New line, top-aligned	Equivalent to <code>\抬头[0]</code>
<code>\单抬</code>	New line, 1 cell higher	Equivalent to <code>\抬头[1]</code>
<code>\双抬</code>	New line, 2 cells higher	Equivalent to <code>\抬头[2]</code>
<code>\三抬</code>	New line, 3 cells higher	Equivalent to <code>\抬头[3]</code>
<code>\相对抬头[N]</code>	Raise by N cells relative to indent	Used in indented contexts like Jiazhu

## Automatic Border Wrapping

---

Taitou is deeply integrated with the Banxin (page frame) system. When text "breaks out" of the normal layout into the margin, `luatex-cn` automatically draws stepped border lines to ensure the silk frames correctly wrap the elevated text.

## Example

---

```
The Great\单抬 Sage Confucius
```

## Technical Details

---

The Elevation feature is defined in `luatex-cn-core-paragraph.sty` and implemented across `luatex-cn-core-textflow.lua` and `luatex-cn-layout-grid.lua`. When an elevation command is invoked, the layout engine adjusts the starting row (`cur_row`) for the current column and records the elevation height. The rendering layer in `luatex-cn-core-render-border.lua` then uses this data to draw the stepped border lines.

---

🔗 Next: [Templates & Format](#) or [Features](#).

# Textbox

---

`luatex-cn` provides powerful vertical text box capabilities with grid alignment.

## Basic Textbox

---

The `\TextBox` (or `\文本框`) command creates an independent vertical text area.

### Usage

```
\TextBox[options]{Content}
```

### Options

Key	Description	Default
<code>height</code>	Height in grid units. 0 for auto-extension.	0
<code>n-cols</code>	Number of columns.	0
<code>border</code>	Show border.	false
<code>background-color</code>	Background color.	None
<code>floating</code>	Enable floating mode (needs x, y).	false

### Features

- **Auto-Extension:** If `height` is 0, the box extends naturally.
- **Compression (Flat Text):** If `height` is fixed and content is too long, text is vertically compressed (v0.1.5).
- **Independent Indentation** (v0.2.2+): Textbox content no longer inherits paragraph indentation from main text, ensuring layout independence (fix #37).

## Fill Textbox

---

`\FillTextBox` (or `\填充文本框`) is a variant of `\TextBox` with default fill alignment.

```
\FillTextBox[height=10]{Content}
```

---

➤ Next: [Features](#).

# Seals (YinZhang / 印章)

---

The `\印章` command adds images like collection marks or personal seals to the page, supporting absolute positioning and opacity.

## Usage

---

```
\印章[page=1, xshift=2cm, yshift=5cm, opacity=0.8, width=3cm]{seal.png}
```

## Parameters

---

- `page` : Page number where the seal appears (default: 1).
- `xshift` / `yshift` : Offsets from the page center or margin.
- `opacity` : Opacity (0.0 to 1.0). For ancient books, 0.7-0.9 is recommended to simulate ink absorption.
- `width` : Display width of the seal image.
- `color` : (Optional) Color for vector-based seals.

## Features

---

- **Layer Management:** Seals are rendered during the `shipout/background` hook by default, appearing below the text. This can be customized if foreground placement is needed.
- **Multi-point Layout:** Multiple seal commands can be used on the same page; they are positioned independently based on their coordinates.
- **Absolute Positioning:** Uses TikZ's coordinate system for pinpoint accuracy on the page.

## Technical Implementation

---

Seals leverage LaTeX `shipout` hooks. Each call to `\印章` registers a `tikzpicture` block inside the appropriate hook, wrapped in an `\ifnum\value{page}=N` conditional to ensure images are output on the correct page and at the correct layer.



---

🔗 Next Steps: Check [Judou](#) or [Features Overview](#).

# Debug Mode

---

`luatex-cn` provides comprehensive debugging features to help troubleshoot layout issues.

## Enable/Disable Debug

---

```
\LtcDebugOn    % or \开启调试  
\LtcDebugOff  % or \关闭调试
```

## Module-level Debug

---

Enable debugging for specific modules:

```
\LtcDebugModuleOn{vertical} % or \开启调试模块{vertical}  
\LtcDebugModuleOff{vertical} % or \关闭调试模块{vertical}
```

## Display Helper Tools

---

### Show Page Frame

```
\LtcShowFrame % or \显示边框
```

### Show Grid Coordinates

Overlay grid coordinates on the page for element positioning:

```
% Show grid (default unit: cm)  
\LtcShowGrid          % or \显示网格  
  
% Specify unit  
\LtcShowGrid[measure=cm] % or \显示网格[measure=cm]  
\LtcShowGrid[measure=pt]  
\LtcShowGrid[measure=mm]
```

```
% Hide grid
\LtcHideGrid           % or \隐藏网格
```

Supported units: `cm` (default), `pt`, `mm`

## Show Coordinates

`\显示坐标` is an alias for `\显示网格`:

```
\显示坐标[measure=pt]
\隐藏坐标
```

## Debug Color Settings

---

Set different debug colors for different modules:

```
\LtcDebugColor{vertical}{blue}
\LtcDebugColor{banxin}{red}
```

## Usage Example

---

```
\documentclass[四库全书彩色]{guji}

% Enable debug mode
\LtcDebugOn
\LtcShowGrid[measure=cm]

\begin{document}
\begin{正文}
Debug content...
\end{正文}
\end{document}
```

## Debug Output

---

When debug is enabled, the compilation log outputs detailed layout information including:

- Grid position calculations
- Character positioning
- Banxin rendering process
- Plugin invocation order

---

 Back to: [Features](#) | [Home](#)

# Changelog

---

All notable changes to this project will be documented here.

## [0.3.0] - 2026-02-18

---

Project is now stable for full-book typesetting.

- **🔧 User Experience Improvements:** Organized parameter naming conventions, updated Wiki and README documentation
- **🔧 Template System:** Added templates folder with ready-to-use typesetting templates
- **🐛 Paragraph Fix:** Fixed `bottom-indent` default value setting (fix #xx)
- **🐛 Banxin Fix:** Fixed banxin chapter alignment handling

**Note:** Before v1.0, commands and parameters may change. Please keep your installation up to date.

## [0.2.9] - 2026-02-18

---

Complete typesetting of *Qinding Siku Quanshu Jianming Mulu (Volume 1)*. Added "digitization" mode for classical text OCR and proofreading.

- **🔧 Helper Commands:** Added `\臣`, `\xshift` and other typesetting auxiliary commands
- **🔧 Page Processing:** Added cover page, collation page, and blank page processing
- **🔧 Digital Mode:** New `guji-digital` document class for OCR-based classical text digitization and proofreading (experimental)
- **🔧 SideNote Enhancement:** `\SideNode` now supports positional parameters and numeric auto-conversion (`\SideNode[5pt] = \SideNode[0.5]` auto-converts to em)
- **🔄 Parameter Renaming:**
  - Renamed: `font-color` Chinese alias `字色` → `字体颜色` (`\Style`, `\Paragraph`, `\SideNode`, etc.)
  - Renamed: `yoffset` → `yshift` (`\SideNode`, Chinese alias `纵偏` → `纵移`)

- Renamed: `distribute` → `fill` (`\TextBox`, Chinese alias 均分 → 填充)
- Renamed: `font-name` → `font` (`\Content`, Chinese alias 字体名 → 字体)
- Improved: Paragraph `first-indent` / `bottom-indent` defaults to empty, automatically follows `indent` parameter
- 🐛 **Taitou Fixes:** Fixed numerous taitou indentation and line-breaking issues
- 🐛 **Judou Fix:** Fixed judou mode behavior inside jiazhu (interlinear notes)
- 🐛 **Banxin Fix:** Fixed single-page banxin processing
- 🐛 **Background Color Fix:** Fixed background color inheritance and override logic

## [0.2.8] - 2026-02-14

---

- ✨ **Free Layout Mode:** New automatic column width and flexible column spacing support
- ✨ **Style Command:** Added `\样式` command and `Style` environment for inline style overrides (font size, color, grid height, etc.)
- ✨ **Cover Feature:** Added `Cover` environment and volume/chapter name commands for ancient book covers
- ✨ **Command Renaming:** Document classes unified to `guji`, `cn-vbook`, `tw-vbook`; commands fully support English/simplified/traditional Chinese names
- 🐛 **Bug Fixes:** Improved modern traditional Chinese vertical typesetting; fixed footnote and heading layout issues
- 🔄 **Refactoring:** Major refactoring of layout-grid module, unified grid/natural engines, rewrote unit tests

## [0.2.7] - 2026-02-09

---

- ✨ **Modern Punctuation System:** Full modern punctuation support (squeeze, kinsoku line-breaking, vertical quote replacement) with mainland/Taiwan style options
- ✨ **Ite-book Document Class:** New `lte-book` class for modern vertical Chinese book typesetting with template system (default/zhonghuashuju)
- ✨ **Footnote System:** Added `\Footnote` command with endnote and page-bottom modes, lujiao/circled numbering styles
- ✨ **Natural Layout Mode:** Added `layout-mode=natural` for non-uniform character spacing with half-grid punctuation and bottom-aligned columns
- ✨ **Flexible Column Widths:** `\Column` now supports `width` parameter for variable-width column layouts

- **🚩 Title Page Column Widths:** TitlePage supports `column-widths` parameter for flexible layouts
- **🐛 Bug Fix:** Fixed content loss when `\newpage` follows auto-pagination at column end (fix #63)
- **🐛 Bug Fix:** Fixed `\Space` command being skipped at column start (fix #62)
- **🐛 Bug Fix:** Fixed `\挪抬` command trailing spaces producing extra whitespace
- **♻️ Refactoring:** Split `render-page.lua` into two modules, reorganized regression tests into three suites

## [0.2.6] - 2026-02-07

---

- **🚩 Taitou Support:** Added `\抬头[N]` system (Single/Double/Triple/Flat/Relative Taitou) with automatic border wrapping
- **🚩 Proper Name/Book Title Marks:** Re-implemented using PDF graphics, solving connectivity and font compatibility issues (fix #53, #56, #57)
- **🚩 Jiazhu Balance Control:** Added `auto-balance` option for non-balanced layouts
- **🐛 Glyph Rotation Fix:** Added automatic 90° rotation for ellipsis and em dash when font lacks vertical forms
- **🐛 Bug Fix:** Fixed sidenote color leakage across page boundaries (fix #44)
- **🐛 Bug Fix:** Fixed `jiazhu` column switching logic

## [0.2.5] - 2026-02-05

---

- **🚩 Catalog Entry Command:** Added `\条目` command with hierarchical indentation for table of contents
- **🚩 Column Control Commands:** Added `\列` (Column) and `\末列` (LastColumn) commands for better layout control (fix #30)
- **🐛 Bug Fix:** Fixed decoration underlines (专名号/书名号) not drawn on last character at column end (fix #54)
- **🐛 Bug Fix:** Fixed character centering offset for some characters (e.g., "漢"), main text now uses width-based centering (fix #55)
- **🐛 Bug Fix:** Fixed sidenote color leakage to `banxin` when crossing page boundaries (fix #48)

## [0.2.4] - 2026-02-03

---

- ✦ **Underline Support:** Added `\Underline` / `\下划线` commands, implemented with the `|` character (fix #5)
- ✦ **Book Title & Proper Name Marks:** Added `\书名号` (wavy line) and `\专名号` (straight line) commands (fix #16)
- 🐛 **Bug Fix:** Fixed sidenote column alignment when switching columns (fix #47)
- 🐛 **Bug Fix:** Fixed sidenote going to second column when crossing pages (fix #49)
- 🐛 **Bug Fix:** Fixed jiazhu alignment issue (fix #51)
- 🐛 **Bug Fix:** Fixed `\newgeometry` not supporting paperwidth/paperheight (fix #52)
- 🐛 **Bug Fix:** Fixed sidenote color leakage causing banxin to turn red when crossing pages

## [0.2.3] - 2026-02-01

---

- ✦ **MeiPi (Marginal Notes):** Added `\眉批` command with auto-positioning and manual coordinate support (fix #46)
- ✦ **Special TextBox:** TextBox frame feature with border shape customization (fix #25)
- ✦ **Publisher Display:** Added publisher display below banxin with customizable font (fix #26)
- 🐛 **Bug Fix:** Fixed annotation and textbox incorrectly inheriting paragraph indentation (fix #37)
- 🐛 **Bug Fix:** Fixed MeiPi appearing on wrong page when crossing page boundaries
- 🐛 **Bug Fix:** Fixed font detection support for simplified/traditional Chinese names (fix #35)
- ♻️ **Refactoring:** Refactored border style system with style stack management for `border`, `outer_border` attributes

## [0.2.2] - 2026-01-31

---

- 🐛 **Bug Fix:** Fixed preamble chapter display on first page (fix #45)
- 🐛 **Bug Fix:** Fixed jiazhu column jumping at column end (fix #42)
- ✦ **Judou Enhancement:** Support for more Chinese punctuation symbols (fix #40)
- 🐛 **Bug Fix:** Fixed TextBox and annotation inheriting paragraph indent (fix #37)



- 🚩 **Debug Feature:** Added debug ruler showing page grid coordinates (fix #31)
- 🚩 **Font Detection:** Support simplified/traditional Chinese font names (fix #35)
- ♻️ **Refactoring:** Consolidated banxin text rendering logic
- ♻️ **Refactoring:** Moved title/publisher commands to metadata.sty
- ☑️ **Module Cleanup:** Cleaned up banxin, guji, yinzhang module dependencies

## [0.2.1] - 2026-01-31

---

- 🚩 **Color Preservation:** Sidenote color preservation across pages (fix #38, #44)
- 🚩 **Style Stack Architecture:** Implemented Phase 3 style stack for multi-attribute support
- 🚩 **Style Registry:** Phase 2 multi-attribute style storage (color, font\_size, grid\_height)
- 🐛 **Bug Fix:** Fixed sidenote rendering order, now correctly displays above silk columns (fix #38)
- 🐛 **Bug Fix:** Punctuation mode now correctly applies to sidenote content (fix #39)
- ♻️ **Jiazhu Migration:** Migrated jiazhu functionality to style stack architecture
- ♻️ **Refactoring:** Unified color registration mechanism

## [0.2.0] - 2026-01-31

---

- 🚩 **Plugin Architecture:** Refactored to standardized plugin API for consistent component behavior
- 🚩 **Global State Management:** Unified configuration via global tables for all modules
- 🚩 **Metadata System:** Centralized book metadata management (title, publisher, chapter markers)
- ♻️ **Module Namespace:** Clear separation of core/, guji/, banxin/, util/, decorate/
- ♻️ **Major Refactoring:** Plugin system refactoring with standardized module interfaces

## [0.1.5] - 2026-01-28

---

- 🚩 **Multi-font Support:** Added multi-font fallback chain and Chinese font setting commands (fix #23).
- 🚩 **New Features:** Implemented `\decorate` and `\fix` functionalities, supporting vertical typesetting correction marks (fix #12).

- ✨ **Textbox Enhancements:** Textbox supports vertical compression (flattening) to fit grid positions (fix #24).
- 🐛 **Bug Fixes:** Fixed layout bugs including opt dimension parsing error, Judou positioning offsets, and long title squeezing (fix #19, #20).
- ☑️ **Regression Tests:** Introduced automated regression testing program based on PDF rendering comparison.
- ♻️ **Refactoring:** Refactored core rendering logic and improved overall code stability.

## [0.1.4] - 2026-01-27

---

- ✨ Added Judou (punctuation marks) functionality
- 🐛 Fixed paragraph indentation and side note column switching issues
- ♻️ Major code refactoring for better readability and maintainability
- ☑️ Added more test files

## [0.1.3] - 2026-01-25

---

- 🐛 Fixed inaccurate page breaks and page number positioning
- ✨ Annotation coordinates now more accurate, skipping page center to prevent clipping
- 📖 Completely typeset Dream of the Red Chamber (Jiaxu manuscript) Chapter 1
- ♻️ Better compliance with Expl3 syntax
- ☑️ Added l3build test files for regression testing
- ☑️ Established comprehensive Lua unit testing framework with `l3build test` command support
- 🐛 Completely fixed Mock logic and API inconsistency issues in unit tests

## [0.1.2] - 2026-01-22

---

- ♻️ Renamed src to tex
- 🐛 Completely fixed CRLF and BOM character issues on Windows
- 📦 Adopted l3build, see build.lua
- 🐛 Fixed package require paths to ensure release version works out of the box

## [0.1.1] - 2026-01-19

---

### Features

- ✨ Added annotation command, still using floating text box underneath

### Fixes

- 🐛 Fixed CRLF and BOM character issues on Windows

### Documentation

- 📖 Updated documentation with bilingual support

## [0.1.0] - 2026-01-18

---

### Major Features

- **Native Geometric Vertical Typesetting Engine:** Deep Lua customization with grid layout system designed for ancient book typesetting
- **Traditional Book Core Elements:** Complete support for Banxin (page center), Yuwei (fishtail), and traditional silk frames
- **Comprehensive Annotation System:** Flexible interlinear notes and side note functionality
- **Ready-to-use Template System:** Preset visual styles including Siku Quanshu and Dream of the Red Chamber manuscripts
- **Cross-platform Font Auto-detection:** Smart recognition of Windows, macOS, and Linux system fonts
- **Electronic Scroll/Traditional Page Support:** Traditional "tube page" logic and electronic reading adaptation
- **Absolute Positioning Seal System:** Support for placing collection seals anywhere on the page

### Project Infrastructure

- Automated CTAN packaging and release workflow
- Comprehensive bilingual user manuals and multiple classic ancient book reproduction examples

# Development Documentation

---

This document summarizes the core design philosophy, modular architecture, and the complete development workflow for the `luatex-cn` typesetting engine.

## 1. Core Architecture: Three-Layer Design

---

`luatex-cn` utilizes a three-layer architecture to decouple layout logic from rendering:

- LaTeX Interface Layer ( `.sty` , `.cls` ):** Defines user commands; sets attributes and parameters via `luatexbase`.
- Coordination Layer ( `core_*.lua` ):** Manages logic flow. For instance, `core_main` orchestrates the main process, while `core_textflow` handles interlinear note splitting.
- Processing Layer ( `layout_*.lua` , `render_*.lua` ):**
  - **Flattening:** Converts TeX node lists into linear Lua tables.
  - **Layout:** Calculates grid coordinates for each character and its associated elements.
  - **Rendering:** Reassembles content based on coordinates and outputs PDF instructions.

## Module Directory Structure (v0.2.0+)

```
tex/
├─ core/           # Core vertical engine (26 files)
├─ guji/          # Ancient book specific features (6 files)
├─ banxin/        # Page center system (5 files)
├─ splitpage/     # Tube page splitting (2 files)
├─ fonts/         # Font auto-detection
├─ util/          # Utility modules (including style registry)
├─ decorate/     # Text decoration
├─ debug/        # Debug tools
└─ configs/      # Preset configuration files
```

## Plugin System (v0.2.0+)

v0.2.0 introduced a standardized plugin API where each module implements a three-phase interface:

```
plugin = {
  initialize = function(params, engine_ctx) end, -- Initialization
  layout = function(list, layout_map, engine_ctx, context) end, -- Layout
  render = function(head, layout_map, params, context, ...) end -- Rendering
}
```

## Style Registry (v0.2.1+)

`luatex-cn-style-registry.lua` provides multi-attribute style storage:

```
-- Register a style, returns unique ID
local style_id = style_registry.register({
  color = {1, 0, 0},
  font_size = 19,
  grid_height = 20
})

-- Retrieve style by ID
local style = style_registry.get(style_id)
```

Supported attributes: `color`, `font_size`, `grid_height`, `font`, etc.

## 2. Participating in Development

---

We welcome contributions and Pull Requests (PRs).

### 2.1 Fork & Workflow

1. **Fork the Repo:** Click the Fork button on GitHub.
2. **Clone:** Clone your forked repository locally.
3. **Coding:** Main source code is located in the `tex/` directory.

[!TIP] VS Code with LaTeX Workshop is recommended for development.

### 2.2 Testing

Please ensure all tests pass before submitting a PR.

1. **Install Dependencies:** You need `l3build` installed.
2. **Unit Tests:** Run Lua unit tests:

```
l3build test
```

This invokes `test/run_all.lua`.

- 3. Regression Tests:** We use automated regression testing based on PDF rendering comparison to prevent layout regressions.

```
l3build check
```

This compiles `.lvt` files in `testfiles/` and compares the output logs (`.tlg`) against baselines. If your changes affect layout, ensure these pass. If layout changes are intentional, update baselines:

```
l3build save test-name
```

## 2.3 Release

For release procedures, please refer to [Release](#).

## 3. Key Technical Details

---

### Vertical Direction (RTT)

Leverages LuaTeX's `dir RTT` attribute. In RTT mode, text flows top-to-bottom, and lines stack right-to-left.

### Attribute Management

Uses LuaTeX attributes to pass metadata across languages (e.g., "this glyph belongs to an interlinear note").

```
[!WARNING] \selectfont clears all active attributes. Ensure attributes are reset  
AFTER setting the font size.
```

### Node Ownership

Passing nodes to TeX via `tex.box[n] = node` transfers ownership. Use `node.copy_list()` if you need to reuse nodes.

## 4. Development Lessons (LEARNING)

---

- **Color Commands:** Must use normalized RGB (e.g., `0 0 0 rg`), not names.
- **Rendering Order:** PDF follows the "painter's model"—later content overlays earlier content. Insert backgrounds at the head of the list to ensure they are at the bottom.
- **Module Loading:** Standardize on `require()` and utilize `package.loaded` for caching.

# Release Process

---

`luatex-cn` follows an automated release workflow to ensure compliance with CTAN (Comprehensive TeX Archive Network) standards.

## Development Workflow

---

Building and testing are powered by `l3build`:

```
# Run all Lua unit tests
l3build test

# Install to local TEXMFHOME
l3build install

# Create release bundle (.zip)
l3build bundle
```

## Automated Publishing (CI/CD)

---

GitHub Actions handles the packaging:

1. **Source Sync:** The `main` branch is used for active development.
2. **Packaging:**
  - Extracts `src/`, `文档/` (docs), and `示例/` (examples).
  - Translates folders and filenames to English as required by CTAN, using `scripts/file_name_translation.json`.
  - Generates the standardized directory structure (doc, example, tex).
3. **Release Branch:** The packaged content is pushed to the `ctan` branch and published via GitHub Releases.

## Versioning

---

- Version numbers are stored in the root `VERSION` file.
- Significant changes are recorded in the [Changelog](#).
- We follow Semantic Versioning (SemVer) principles.



---

🔗 Next Steps: Check [Development Documentation](#).