CTEX 宏集手册

CTEX.ORG

2019/05/29 v2.4.16*

简介

CTEX 宏集是面向中文排版的通用 LATEX 排版框架，为中文 LATEX 文档提供了汉字输出支持、标点压缩、字体字号命令、标题文字汉化、中文版式调整、数字日期转换等支持功能，可适应论文、报告、书籍、幻灯片等不同类型的中文文档。

CTEX 宏集支持 LATEX、pdfLATEX、XELATEX、LuaLATEX、upLATEX 等多种不同的编译方式，并为它们提供了统一的界面。主要功能由宏包 ctex 和中文文档类 ctexart、ctexrep、ctexbook 和 ctexbeamer 实现。

目录

第 1 节 介绍 2

第 2 节 简明教程 3
- 2.1 CTEX 宏集的组成 3
- 2.2 CTEX 宏集的安装和更新 . . . 3
- 2.3 使用 CTEX 文档类 4
- 2.4 使用 ctex 宏包 4

第 3 节 宏包选项与 \ctexset 命令 5

第 4 节 编译方式、编码与中文字库 5
- 4.1 编译方式 5
- 4.2 中文编码 6
- 4.3 中文字库 6

第 5 节 排版格式设定 8
- 5.1 文档默认字号 8
- 5.2 章节标题风格 9
- 5.3 排版方案选项 9

第 6 节 文档汉化 11
- 6.1 日期汉化 11
- 6.2 文档标题汉化 11
- 6.3 页面格式设置与汉化 13

第 7 节 章节标题格式设置 14

第 8 节 实用命令 25
- 8.1 字号与间距 25
- 8.2 中文数字转换 26
- 8.3 杂项 26

第 9 节 LuaLATEX 下的中文支持方式 26
- 9.1 LuaLATEX 下替代字体的设置 . 27

第 10 节 CTEX 宏集的配置文件 28
- 10.1 修改宏包默认选项 28
- 10.2 宏包载入后的配置 29
- 10.3 配置标题中文翻译 29
- 10.4 自定义字体集 29

第 11 节 对旧版本的兼容性 29
- 11.1 CTEX 0.8a 及以前的版本 . . . 29
- 11.2 CTEX 0.9–CTEX 1.0d 30
- 11.3 CTEX 1.02c 以后的 SVN 开发版 31
- 11.4 CTEX 2.2 之前的版本 32
- 11.5 CTEX 2.4.1 和 2.4.2 32

第 12 节 宏集依赖情况与手工安装方法 32

第 13 节 开发人员 33

第 14 节 代码实现 35

版本历史 139

代码索引 141

*ctex-kit rev. 735dfe2.

第 1 节　介绍

历史

CTEX 宏集的源头有两个：一是王磊编写的 cjkbook 文档类，二是吴凌云编写的 GB.cap。这些工作没有经过认真、系统的设计，也没有用户文档，不利于维护和改进。

2003 年，吴凌云使用 doc 和 DocStrip 重构了整个工程，并增加了许多新的功能，称为 ctex 宏包。2007 年，oseen 和王越在 ctex 宏包的基础上，增加了对 UTF-8 编码的支持，开发出了 ctexutf8 宏包。

2009 年 5 月，我们在 Google Code 建立了 ctex-kit 项目[1]，对 ctex 宏包及相关脚本进行了整合，并加入了对 XETEX 引擎的支持。在开发新版本时，考虑到合作开发和调试的方便，我们放弃了 doc 和 DocStrip，采取了直接编写宏包代码的方式。

2014 年 3 月，为了适应 LATEX 的最新发展，特别是 LATEX3 的逐渐成熟，李清用 LATEX3 重构了整个宏包的代码，并重新使用 doc 和 DocStrip 工具进行代码的管理，升级版本号为 2.0，并改称 CTEX 宏集。

2015 年 3 月，由于 Google Code 即将停止服务，ctex-kit 项目迁移至 GitHub[2]。

最初，Knuth 在设计开发 TEX 的时候没有考虑到多国语言支持，特别是对多字节的中日韩语言的支持。这使得 TEX 以至后来的 LATEX 对中文的支持一直不是很好。即使在 CJK 宏包解决了中文字符处理的问题以后，中文用户使用 LATEX 仍然要面对许多困难。这些困难里，以章节标题的中文化为最。由于中文和西文语言习惯的差异，用户很难使用标准文档类中的代码结构来表达中文标题。于是，用户不得不对标准文档类做较大的修改。除此之外，日期格式、首行缩进、中文字号和字距等细节问题，也需要精细的调校。我们设计 CTEX 宏集的目的之一就是解决这些 LATEX 文档的汉化难题。

另一方面，随着 TEX 引擎和 LATEX 宏包的不断发展，LATEX 的中文支持方式从早期的专用系统（如 CCT）发展为适用于不同引擎的多种方式[3]。这些方式的适用情况和使用方式有不少细节上的差异，同时操作系统的不同、语言环境的不同等客观情况又进一步带来了更多的细节差异。我们设计 CTEX 宏集的另一个主要目的就是尽可能消除这些差异带来的影响，使用户能够以一个统一的接口来使用不同的中文支持方式，使得同一份文档能够在不同环境下交换使用。

CTEX 宏集的许多实现细节离不开热心朋友们在 bbs.ctex.org 论坛上的讨论，在此对参与讨论的朋友们表示感谢。

关于宏集名字的说明

CTEX 之名是英文单词 China（中国）或 Chinese（中文）的首字母“C”与“TEX”结合而成的。在纯文本环境下，该名字应写作“CTeX”。

CTEX 宏集是由 CTEX 社区 发起并维护的 LATEX 宏包和文档类的集合。社区另有发布名为 CTEX 套装 的 TEX 发行版，与本文档所述的 CTEX 宏集并非是同一事物。

ctex 则是本宏集中的 ctex.sty 的名字。这一完全小写的名称，在过去也被用来指代整个 CTEX 宏集，不过现在则特指 ctex.sty 这一宏包。在不引起歧义的情况下，它也可以沿用过去的习惯，代指整个宏集。

[1] http://code.google.com/p/ctex-kit/

[2] https://github.com/CTeX-org/ctex-kit

[3] 比如：pdfTEX 引擎下的 CJK、zhmCJK 宏包，XETEX 引擎下的 xeCJK 宏包和 LuaTEX 引擎下的 LuaTeX-ja 宏包

第 2 节　简明教程

2.1　CTEX 宏集的组成

为了适应用户不同的需求，我们将 CTEX 宏集的主要功能分散在四个中文文档类和三个宏包当中，具体的组成见表 1。

表 1　CTEX 宏集的组成

类别	文件	说明
文档类	`ctexart.cls`	标准文档类 article 的汉化版本，一般适用于短篇幅的文章
	`ctexrep.cls`	标准文档类 report 的汉化版本，一般适用于中篇幅的报告
	`ctexbook.cls`	标准文档类 book 的汉化版本，一般适用于长篇幅的书籍
	`ctexbeamer.cls`	文档类 beamer 的汉化版本，适用于幻灯片演示
宏包	`ctex.sty`	提供全部功能，但默认不开启章节标题设置功能，需要使用 `heading` 选项来开启
	`ctexsize.sty`	定义和调整中文字号，在 ctex 宏包或 CTEX 中文文档类之外单独调用
	`ctexheading.sty`	提供章节标题设置功能（见 7 节），在 ctex 宏包或 CTEX 中文文档类之外单独调用

2.2　CTEX 宏集的安装和更新

CTEX 宏集依赖的宏包和宏集已被最常见的 TEX 发行版 TEX Live 和 MiKTEX 所收录。如果本地安装 TEX Live 或 MiKTEX 不是完全版本，就可能需要通过这两个发行版提供的宏包管理器来安装宏包。

TEX Live 的宏包管理器是 tlmgr。用户可以在系统命令行中[4]执行

```
tlmgr gui
```

启动管理器的图形界面（Windows 用户也可以通过开始菜单的 TeX Live 2015 ➔ TeX Live Manager 打开）。连接上远程仓库之后，搜索 ctex 安装即可。tlmgr 的图形界面使用 Perl 编写，容易造成系统假死。遇到这种问题的用户，也可以直接在系统命令行执行

```
tlmgr install ctex
```

来安装 CTEX 宏集[5]。

MiKTEX 的宏包管理器是 mpm (MiKTEX Package Manager)。用户可以在开始菜单找到 MiKTeX ➔ Maintenance (Admin) ➔ Package Manager (Admin)，打开管理器，连接上远程仓库之后，搜索 ctex 安装即可。

若希望了解 CTEX 宏集具体的依赖情况和手工安装宏集的方法，请参阅第 12 节。

当宏包发布新版本，并被发行版在远程仓库安装之后，在本地就可以通过宏包管理器来取得新版本。

对于 TEX Live，可以在 tlmgr 的图形界面点击“更新全部已安装的”按钮或者在命令行执行

[4]Windows 系统的命令行是 CMD 命令提示符，你可以使用 Win + R 组合键打开“运行”对话框，然后输入 cmd 确认打开命令提示符窗口。

[5]*nix 用户可能需要超级用户权限才能正确安装宏集。

```
tlmgr update --all
```

来完整更新已安装的宏包。

对于 MiKTEX，在开始菜单找到 MiKTeX ➔ Maintenance (Admin) ➔ Update (Admin)，按照界面说明更新即可。

2.3 使用 CTEX 文档类

如果用户需要在标准文档类的基础上添加中文支持和中文版式支持，我们建议用户使用 CTEX 宏集提供的四个中文文档类。

CTEX 宏集提供了四个中文文档类：ctexart、ctexrep、ctexbook 和 ctexbeamer，分别对应 LATEX 的标准文档类 article、report、book 和 beamer。使用它们的时候，需要将涉及到的所有源文件使用 UTF-8 编码保存[6]。

例 1

```
\documentclass[UTF8]{ctexart}
\begin{document}
中文文档类测试。你需要将所有源文件保存为 UTF-8 编码。

你可以使用 XeLaTeX、LuaLaTeX 或 upLaTeX 编译，也可以使用 (pdf)LaTeX 编译。
推荐使用 XeLaTeX 或 LuaLaTeX 编译。
\end{document}
```

以下是使用 ctexbeamer 文档类编写中文演示文稿的一个示例。

例 2

```
\documentclass[UTF8]{ctexbeamer}
\begin{document}
\begin{frame}{中文演示文档}
\begin{itemize}
  \item 你需要将所有源文件保存为 UTF-8 编码
  \item 你可以使用 XeLaTeX、LuaLaTeX 或 upLaTeX 编译
  \item 也可以使用 (pdf)LaTeX 编译
  \item 推荐使用 XeLaTeX 或 LuaLaTeX 编译
\end{itemize}
\end{frame}
\end{document}
```

2.4 使用 ctex 宏包

用户在使用非标准文档类时，如果需要添加中文支持或中文版式支持，则可以使用 ctex 宏包。

有些文档类是建立在 LATEX 标准文档类之上开发的。这时，给 ctex 宏包加上 heading 选项，可以将章节标题设置为中文风格。

例 3

```
\documentclass{ltxdoc}
\usepackage[UTF8, heading = true]{ctex}
\begin{document}
\section{简介}
章节标题中文化的 \LaTeX{} 手册。
\end{document}
```

[6] 使用 (pdf)LATEX 时也能够使用 GBK 编码，但不推荐。（见 4.2 节）

第 3 节　宏包选项与 \ctexset 命令

CTEX 宏集已经尽可能就中文的行文和版式习惯做了调整和配置，通常而言，这些配置已经够用。因此，除非必要，我们不建议普通用户修改这些默认配置。如果你认为 CTEX 宏集的默认配置还可以完善，可以在项目主页上 提交 issue，向我们反映，我们会酌情在后续版本中予以改进。

不过，CTEX 宏集也提供了一系列选项。用户可以使用这些选项来控制 CTEX 宏集的行为。具体来说，这些选项里，有的以传统的方式提供，也有的以 ⟨*key*⟩=⟨*value*⟩ 的形式提供。对于以键值对形式提供的选项，在下面的说明中使用**粗体**来表示 CTEX 的默认设置。

另一方面，这些选项可以分为以下三类：

- 名字后带有 ☆ 号的选项，只能作为宏包/文档类选项，需要在引入宏包/文档类的时候指定；
- 名字后带有 ⋆ 号的选项，只能通过 CTEX 宏集提供的用户接口 \ctexset 来设定；
- 名字后不带有特殊符号的选项，既可以作为宏包/文档类选项，也可以通过 \ctexset 来设定。

后续文档将在使用说明中对某些特殊的选项加以说明。

\ctexset　　New: 2014-03-18

\ctexset {⟨键值列表⟩}

是 CTEX 宏集的通用控制命令，用来在宏包载入后控制宏包的各项功能。\ctexset 的参数是一个键值列表，以通用的接口完成各项设置。

\ctexset 的参数是一组由逗号分隔的选项列表，列表中的选项通常是一个 ⟨*key*⟩=⟨*value*⟩ 格式的定义。例如设置摘要与参考文献标题名称（6.2 节）就可以使用：

例 4

```
\ctexset{
  abstractname = {本文概要},
  bibname      = {文\quad 献}
}
```

\ctexset 采用 LaTeX3 风格的键值设置，支持不同类型的选项与层次化的选项设置，相关示例见 7 节。

第 4 节　编译方式、编码与中文字库

4.1　编译方式

CTEX 宏集会根据用户使用的编译方式[7]，在底层选择不同的中文支持方式（见表 2）。

表 2　CTEX 宏集的中文支持方式

编译方式	(pdf)LaTeX	XeLaTeX	LuaLaTeX	upLaTeX*
支持宏包	CJK	xeCJK	LuaTeX-ja	原生

* pLaTeX-ng（或称 ApLaTeX）与 upLaTeX 兼容。使用 pLaTeX-ng 编译时，ctex 采用与 upLaTeX 相同的设置。

[7] LaTeX、pdfLaTeX、XeLaTeX、LuaLaTeX 及 upLaTeX。

不同的编译方式和中文支持方式会在一定程度上影响 CTEX 宏集的行为，比如宏包对编码的处理。在用户使用 XELATEX、LuaLATEX 及 upLATEX 编译时，CTEX 宏集使用（且仅能使用）UTF-8 编码；而因为历史原因，在用户使用 LATEX 及 pdfLATEX 编译时，宏包默认使用 GBK 编码。用户需要保证编译方式、源文件编码、宏包编码选项三者一致。关于宏包编码选项，可以参考 4.2 节。

除去文档编码之外，选择不同的编译方式还可能影响 CTEX 宏集对字体选择、空格处理、标点处理的处理。具体的影响将在本文档后续内容中进行阐述。

4.2　中文编码

GBK ☆
UTF8 ☆

指明编写文档时使用的编码格式。CTEX 宏集无法检测用户编写文档时使用的编码格式，因此需要用户自行指定编码。*我们建议用户总是使用 UTF-8 编码，并显式指定 UTF8 选项，并使用 XELATEX、LuaLATEX 或 upLATEX 编译。*

使用 XELATEX、LuaLATEX 或 upLATEX 编译时，CTEX 宏集强制使用 UTF-8 编码，此时 GBK 选项无效。使用 (pdf)LATEX 编译时，CTEX 宏集默认启用 GBK 选项；不过，用户也可以显式声明 UTF8 选项，使 CTEX 宏集工作在 UTF-8 编码下。

4.3　中文字库

以往，为 LATEX 文档配置中文支持是一件相当繁琐的事情。默认情况下，CTEX 宏集能自动检测用户使用的编译方式（参见 4.1 节）和操作系统[8]，选择合适的底层支持和字库，从而简化配置过程。自动配置的情况参见表 3。

表 3　CTEX 宏集自动配置字体策略

	Mac OS X	Windows New[1]	Windows Old[2]	其他
XELATEX	xeCJK 华文字库	xeCJK 中易字库 + 微软雅黑	xeCJK 中易字库	xeCJK Fandol 字库[3]
LuaLATEX[4]	LuaTeX-ja 华文字库	LuaTeX-ja 中易字库 + 微软雅黑	LuaTeX-ja 中易字库	LuaTeX-ja Fandol 字库
pdfLATEX	不可用[5]	CJK + zhmetrics 中易字库 + 微软雅黑	CJK + zhmetrics 中易字库	不可用[5]
LATEX + DVIPDFM*x*	不可用[6]	CJK + zhmetrics 中易字库 + 微软雅黑	CJK + zhmetrics 中易字库	CJK + zhmetrics Fandol 字库
upLATEX + DVIPDFM*x*	不可用[6]	zhmetrics-uptex 中易字库 + 微软雅黑	zhmetrics-uptex 中易字库	zhmetrics-uptex Fandol 字库

1 Windows Vista 及以后的 Windows 操作系统。
2 Windows XP 及以前的 Windows 操作系统。
3 由马起园、苏杰、黄晨成等人开发的开源中文字体，参见：https://github.com/clerkma/fandol-fonts。
4 LuaLATEX 编译时使用 LuaTeX-ja 宏包。对此，第 9 节有特别说明。
5 受 pdfTEX 的限制，无法嵌入 OpenType 字体。
6 目前受 DVIPDFM*x* 的限制，Mac OS X 系统上的黑体和仿宋无法读取。

通常，由 CTEX 宏集进行的自动配置已经足够使用，无需用户手工干预；但是 CTEX 仍然提供了一系列选项，供在 CTEX 的自动选择机制因为意外情况失效，或者在用户有特殊需求的情况下使用。*除非必要，用户不应使用这些选项。*

[8] CTEX 宏集现在能够识别 Mac OS X 系统以及 Windows 系统。

zhmap ☆

Updated: 2014-03-08

zhmap = ⟨**true**|*false*|*zhmCJK*⟩

指定字体映射机制。本选项只在使用 pdfLaTeX/LaTeX 编译时有意义。

true　使用 zhmetrics 宏包，将 CJK 字库通过 \special 命令映射到 .ttf 文件。

false　使用传统的 CJK 字库（Type 1）[9]。

zhmCJK　载入 zhmCJK 宏包[10]，由 zhmCJK 宏包提供从 CJK 字库到 .ttf 的映射。

fontset

New: 2014-03-08

fontset = ⟨*none*|*adobe*|*fandol*|*founder*|*mac*|*macnew*|*macold*|*ubuntu*|*windows*|*windowsnew*|*windowsold*|...⟩

指定 CTeX 宏集加载的字库。

如果没有指定 fontset 的值，CTeX 宏集将自动检测用户使用的操作系统，配置相应的字体（参见表 3）。

CTeX 预定义了以下六种中文字库。

adobe　使用 Adobe 公司的四款中文字体，不支持 *pdfLaTeX*。

fandol　使用 Fandol 中文字体，不支持 *pdfLaTeX*。

founder　使用方正公司的中文字体。

mac　使用 Mac OS X 系统下的字体，不支持 *(pdf)LaTeX*。

macnew　使用 El Capitan 或之后的多字重宋体、苹方黑体。

macold　使用 Yosemite 或之前的华文字库。

ubuntu　使用 Ubuntu 系统下的文泉驿和文鼎字体。

windows　使用简体中文 Windows 系统下的中文字体，自动判断 Windows 系统版本，采用 windowsnew 或 windowsold 的设置。

windowsnew　使用简体中文 Windows Vista 或之后系统下的中易字体和微软雅黑字体。

windowsold　使用简体中文 Windows XP 或之前系统下的中易字体。

注意：使用 (pdf)LaTeX 编译的时候，若设置 zhmap = false（比如需要使用 LaTeX + Dvips 编译），则需要按照传统方式[11]在本地安装好 CJK 字体。

如果不想使用 CTeX 预定义的中文字库，可以设置 fontset 为下述值之一。

none　不配置中文字体，需要用户自己配置。

⟨*name*⟩　这里 ⟨*name*⟩ 为自定义的名字。CTeX 宏集将载入名为 ctex-fontset-⟨*name*⟩.def 的文件作为字体配置文件。因此，请先保证文件的存在。可以在当前工作目录或者本地 TDS 目录树下合适位置建立一个名为 ctex-fontset-⟨*name*⟩.def 的文件，在这个文件里面自定义中文字体。然后通过使用 fontset=⟨*name*⟩ 选项来调用它。字体配置文件的具体写法可以参考 CTeX 宏集 fontset 目录下的字体配置文件。

注意：*如果希望使用 \ctexset 在导言区指定字库，则需要先在宏包/文档类选项中指定 fontset = none。例如：*

例 5

```
\documentclass[fontset = none]{ctexart}
\ctexset{fontset = founder}
\begin{document}
在文档类选项中声明 \verb|fontset = none|，随后在导言区用 \verb|\ctexset|
指定字体。
\end{document}
```

[9] 如果需要使用自定义的字体映射文件，或者希望使用 Type1 字库，请禁用本选项。

[10] zhmCJK 宏包基于 zhmetrics 和 CJK 宏包，提供与 xeCJK 宏包类似的用户接口。

[11] 可以使用 zhmetrics 宏包提供的脚本 CTeXFonts.lua。

CTEX 宏集预定义的中文字库还定义了一些字体命令。除了在 ubuntu 字库中没有 \fangsong 的定义外，所有字库都有以下四个字体命令。

\songti　宋体，CJK 等价命令 \CJKfamily{zhsong}。
\heiti　黑体，CJK 等价命令 \CJKfamily{zhhei}。
\fangsong　仿宋，CJK 等价命令 \CJKfamily{zhfs}。
\kaishu　楷书，CJK 等价命令 \CJKfamily{zhkai}。

在 windows 和 founder 字库中，额外定义了 \lishu 和 \youyuan。

\lishu　隶书，CJK 等价命令 \CJKfamily{zhli}。
\youyuan　圆体，CJK 等价命令 \CJKfamily{zhyou}。

在 windowsnew 和 macnew 字库中，还有 \yahei。其中在 macnew 字库中，\yahei 实际调用苹方黑体。此举乃是为了更好的兼容性而设。

\yahei　微软雅黑，CJK 等价命令 \CJKfamily{zhyahei}。

在 macnew 字库中，还定义了 \pingfang。

\pingfang　苹方黑体，CJK 等价命令 \CJKfamily{zhpf}。

第 5 节　排版格式设定

5.1　文档默认字号

zihao ☆　zihao = ⟨*-4*|*5*|*false*⟩

New: 2015-05-06

将文章默认字号（\normalsize）设置为小四号字或五号字，具体情况见表 4。false 禁用本功能。本选项可以用于四个 CTEX 文档类和 ctex 宏包，也可以用于 ctexsize 宏包。

scheme = chinese 时，对标准文档类默认值为 5，即设置 \normalsize 为五号字；对 beamer 则为 false，使用文档类原有的设置。

表 4　标准字体命令与字号的对应

	zihao = 5		zihao = -4		10pt	11pt	12pt
字体命令	字号	bp	字号	bp	pt	pt	pt
\tiny	七号	5.5	小六	6.5	5	6	6
\scriptsize	小六	6.5	六号	7.5	7	8	8
\footnotesize	六号	7.5	小五	9	8	9	10
\small	小五	9	五号	10.5	9	10	11
\normalsize	五号	10.5	小四	12	10	11	12
\large	小四	12	小三	15	12	12	14
\Large	小三	15	小二	18	14	14	17
\LARGE	小二	18	二号	22	17	17	20
\huge	二号	22	小一	24	20	20	25
\Huge	一号	26	一号	26	25	25	25

10pt ☆
11pt ☆
12pt ☆

CTEX 文档类是在 LATEX 标准文档类之上开发的。因此，除了可以使用 CTEX 宏包定义的字号选项之外，还可以使用标准文档类的同类选项（10pt、11pt 和 12pt）。在使用这些来自标准文档类的选项的时候，CTEX 文档类的字号选项会被抑制。亦即，在 zihao 选项之后设置 10pt 选项，zihao 选项将不再起作用。

标准文档类的其他选项在 CTEX 文档类中依旧有效。例如，设置纸张大小和方向的 a4paper 和 landscape，设置单双面的 oneside 和 twoside 等。CTEX 会将这些选项传给标准文档类[12]。

5.2　章节标题风格

heading ☆

New: 2014-03-08

heading = ⟨*true*|**false**⟩

本选项只能在调用 ctex.sty 时作为宏包选项使用。

CTEX 宏集提供了一套用于修改文档章节标题格式的接口。该选项用于选择是否启用该功能。详细的设置方法请参见 7 节和 6.3 节。

CTEX 宏集提供的四个文档类总是启用该功能。如果在 ctex.sty 下启用该选项，将会检查当前是否使用 LATEX 标准文档类。若然，则该选项将会使得 ctex.sty 宏包的行为和 CTEX 宏集提供的四个中文文档类完全一致；若不然，则会根据 \chapter 是否有定义来使用 ctexbook 或者 ctexart 的标题设置。

sub3section ☆
sub4section ☆

修改 \paragraph 和 \subparagraph 的格式。

默认情况下，\paragraph 和 \subparagraph 会将标题与随后的正文排版在同一个段落。启用 sub3section 会将 \paragraph 的格式修改为类似 \section 的格式，并将 \subparagraph 的格式修改为原本 \paragraph 的格式。启用 sub4section 会将 \paragraph 和 \subparagraph 的格式都修改为类似 \section 的格式。

启用该选项通常需要将计数器 secnumdepth 的值为设置为 4 或 5。

具体格式可参考 7 节中的 runin 和 afterskip 选项。

注意，上述两个选项只有在非 beamer 文档类下 heading 选项启用的时候才有意义。亦即，只有在使用除了 ctexbeamer 的三个 CTEX 文档类或启用了 heading 的 ctex.sty 的时候才有意义。

5.3　排版方案选项

scheme ☆

New: 2015-04-15

scheme = ⟨**chinese**|*plain*⟩

选择文章的排版方案，预设有 chinese 和 plain 两种方案。

chinese　对 beamer 以外的文档类，调整默认字号为五号字，并调整行距为 1.3；汉化文档中的标题名字（如“图”、“表”、“目录”和“参考文献”等，见 6.2 节）；在 heading = true 的情况下[13]（5.2 节），还会将章节标题的风格修改为中文样式（见 7 节）。

当关闭 heading 选项的 ctex 宏包与标准文档类或其衍生文档类联用时，会载入 indentfirst 宏包，以实现章节标题后的段首缩进。

[12] 事实上，LATEX 在文档类中的选项是全局设定的，除了对使用的文档类有影响外，也可能会影响到随后使用的宏包。如果这些宏包中有某些选项出现在文档类的选项列表中，那么该选项将会被自动激活。

[13] 使用 CTEX 文档类，或者使用 ctex 宏包且开启该选项时。

plain　不调整默认字号和行距，不会汉化文档中的标题名字，也不会将章节标题风格修改为中文样式，同时不会调整 \pagestyle，并禁用 autoindent 选项。事实上，此时的 CTEX 宏集只提供了中文支持功能，而不对文章版式进行任何修改。

punct　punct = ⟨**quanjiao**|*banjiao*|*kaiming*|*CCT*|*plain*⟩

Updated: 2014-04-11

设置标点处理格式。预定义好的格式有：

quanjiao　全角式：所有标点占一个汉字宽度，相邻两个标点占 1.5 汉字宽度；
banjiao　半角式：所有标点占半个汉字宽度；
kaiming　开明式：句末点号[14]用占一个汉字宽度，标号和句内点号占半个汉字宽度；
CCT　CCT 式：所有标点符号的宽度略小于一个汉字宽度；
plain　原样（不调整标点间距）。

space　space = ⟨*true*|*false*|**auto**⟩

Updated: 2014-03-08

是否在生成的 PDF 中保留汉字后面的空格。

true　总是保留汉字后的空格。此时，用户需要自行在行尾加上 % 处理换行产生的空格[15]。
false　总是忽略掉汉字后面的空格，不论汉字后是什么（使用 (pdf)LaTeX 编译时）；等同于 auto 的效果（使用 XeLaTeX 编译时）。不建议使用该选项。
auto　根据空格后面的情况决定是否保留：如果空格后面是汉字，则忽略该空格，否则保留。

例如，使用

例 6

```
\ctexset{space=true}
汉字 分词
技术 English
```

将得到“汉字 分词 技术 English”；使用

例 7

```
\ctexset{space=auto}
汉字 分词
技术 English
```

则会得到“汉字分词技术 English”。

使用 LuaLaTeX 及 upLaTeX 编译的时候，该选项无效：汉字间的空格以及汉字与西文字符之间的空格总是有效，不会被忽略，但可以自动忽略掉由换行产生的空格。

linespread ☆　linespread = ⟨数值⟩

New: 2014-04-23

接受一个浮点数值，设置行距倍数。本选项的初始值与 scheme 有关。

scheme = chinese　对标准文档类初始值为 1.3，即 1.3 倍行距。此时，相邻两行的基线（\baselineskip）距离为 $1.3 \times 1.2 = 1.56$ 倍字体高度。对 beamer 不改变行距，即使用默认的单倍行距。
scheme = plain　CTEX 宏集默认不调整行距倍数，文档中的行距由所选文档类和其他宏包或用户设置决定。

autoindent　autoindent = ⟨**true**|*false*|数值|带单位的数值⟩

New: 2014-03-13

在字体大小发生变化时，是否自动调整段首缩进（\parindent）的大小。

[14] 标点符号分为标号与点号。点号分为两类，一共七种：句末点号有句号、问号和叹号；句内点号有逗号、顿号、冒号和分号。
[15] LaTeX 将单个换行视作一个空格。

⟨数值或带单位的数值⟩　用于设置段首缩进的长度。如果不带单位，则默认单位是单个汉字字宽 \ccwd；如果带单位，则使用该单位。

true　等价于设置 autoindent = 2。

false　禁用自动调整功能，可以设置固定长度的段首缩进。如设置每段缩进 40 点：

例 8

```
\ctexset{autoindent=false}
\setlength\parindent{40pt}
```

linestretch ⋆　linestretch = ⟨数值或长度⟩

New: 2014-03-26

linestretch 是一个比较特殊的选项，它用来设置汉字之间弹性间距的弹性程度。如果有单位，则可以在选项中直接写；如果是数字，单位则是汉字宽度 \ccwd 的倍数。

如果行宽不是汉字宽度的整数倍，为了让段落左右两端对齐，自然就要求伸展汉字之间的间距，而 linestretch 选项就是设置每行总的允许伸行量。初始值是允许每行伸行一个汉字的宽度 \ccwd，并且此宽度能根据字号变化动态调整。

过小的 linestretch 可能导致段落文字右侧可能参差不齐；较大的 linestretch 选项则可以帮助拥有较长不可断行内容的复杂段落方便地断行，而不会产生大量编译警告；但很大的 linestretch 则会掩盖段落不良断行产生的坏盒子警告。

如果将 linestretch 选项的值设置为 \maxdimen，则可以禁止按字号自动修改每行的允许伸长量。此时汉字间的弹性间距则固定为 \baselineskip 的 0.08 倍。

第 6 节　文档汉化

6.1　日期汉化

CTEX 宏包对显示当前日期的 \today 命令进行了汉化，使之以中文的方式显示今天的日期。如编译本文档的日期就是“2019 年 5 月 29 日”。

today ⋆　today = ⟨**small**|*big*|*old*⟩

该选项用来控制 \today 命令的输出格式：

small　效果为“2019 年 5 月 29 日”。使用阿拉伯数字和汉字的日期格式。

big　效果为“二〇一九年五月二十九日”。使用全汉字的日期格式。

old　效果为“May 29, 2019”。使用文档原来的（英文）日期格式。

设置日期格式使用 \ctexset 命令完成，例如设置全汉字的日期格式：

例 9

```
\ctexset{today=big}
```

CTEX 宏包的中文日期功能实际上是调用 zhnumber 宏包完成的。如果需要更多有关日期、时间的命令和更复杂的设置，可以查阅 zhnumber 宏包的文档。

6.2　文档标题汉化

这里主要介绍由宏包 scheme 选项（5.3 节）控制的文档标题汉化功能。

设置文档标题名的示例可见例 4。下面的选项（如 contentsname）主要用来重新定义与选项同名的宏（如 \contentsname）的定义。

contentsname ⋆　contentsname = ⟨名字⟩

设置目录标题名 \contentsname。中文默认为“目录”。

listfigurename ⋆　listfigurename = ⟨名字⟩

设置插图目录标题名 \listfigurename。中文默认为“插图”。

listtablename ⋆　listtablename = ⟨名字⟩

设置表格目录标题名 \listtablename。中文默认为“表格”。

figurename ⋆　figurename = ⟨名字⟩

设置图片环境标题名 \figurename。中文默认为“图”。

tablename ⋆　tablename = ⟨名字⟩

设置表格环境标题名 \tablename。中文默认为“表”。

abstractname ⋆　abstractname = ⟨名字⟩

设置摘要 abstract 环境标题名 \abstractname。中文默认为“摘要”。注意 book 类没有摘要，该选项无效。

indexname ⋆　indexname = ⟨名字⟩

设置索引标题名 \indexname。中文默认为“索引”。

appendixname ⋆　appendixname = ⟨名字⟩

设置附录标题名 \appendixname。中文默认为“附录”。

bibname ⋆　bibname = ⟨名字⟩

设置参考文献标题名 \refname（对 article）或 \bibname（对 report、book 和 beamer）。中文默认为“参考文献”。

proofname ⋆　proofname = ⟨名字⟩

设置证明环境的名称 \proofname。中文默认为“证明”。

如果使用 ctexbeamer 文档类或者在 beamer 文档类下使用 ctex 包，还会汉化常用定理类环境的诸如“定义”、“定理”和“引理”等名称。此时，还有下列三个选项。

refname ⋆　refname = ⟨名字⟩

设置参考文献标题名 \refname。中文默认为“参考文献”。

algorithmname ⋆　algorithmname = ⟨名字⟩

设置算法环境标题名 \algorithmname。中文默认为“算法”。

continuation ⋆　continuation = ⟨名字⟩

设置 beamer 可断页的帧在续页标题中的延续标识 \insertcontinuationtext。中文默认为“（续）”。

6.3　页面格式设置与汉化

当使用了 CTEX 的文档类或是用 ctex 宏包加载了 heading 选项时，会设置整个文档的页面格式（page style）为 headings，即相当于设置了

```
\pagestyle{headings}
```

在页眉中显示当前章节的编号与标题。

同时，CTEX 宏包也会对默认的 headings 页面格式进行修改，使之调用 \CTEXthechapter、\CTEXthesection 等宏来正确显示中文的章节编号。

CTEX 宏包的默认页面格式设置是经过汉化的 headings，其基本效果如本文档所示，只在页眉一侧显示章节编号和标题，另一侧显示页码。

更复杂的页面格式可以通过调用 fancyhdr、titleps 等宏包来设置。CTEX 宏包同时也为这些自定义页面格式的包提供了以下宏供使用：

- \CTEXthechapter、\CTEXthesection 等章节编号（见 7 节）。它们用来代替英文文档类中的 \thechapter、\thesection 等宏。
- \leftmark、\rightmark，它们是在使用章节标题命令后，自动设置的宏。它们实际是在与章节标题命令对应的标记命令 \chaptermark、\sectionmark 中调用 \markright 或 \markboth 生成的。

有关 LATEX 页面标记的涵义与使用细节，已经超出了本文档讨论的范围。可以参考 [1, Chapter 23]、[2, §4.3, §4.4] 等书籍。

这里举一个例子，说明通过重定义 \sectionmark，在 ctexart 文档类中的标准 headings 页面格式下控制页眉的方式：

例 10

```
\documentclass{ctexart}
\pagestyle{headings}
\ctexset{section={
    name={第,节},
    number=\arabic{section},
  }
}
\renewcommand\sectionmark[1]{%
  \markright{\CTEXifname{\CTEXthesection——}{}#1}}

\begin{document}

\section{天地玄黄}
\newpage

\section{宇宙洪荒}

\end{document}
```

在上例中，我们设置了页眉的形式是用破折号分开的节编号与节标题，即“第 1 节——天地玄黄”、“第 2 节——宇宙洪荒”。

CTEX 宏包已经对 fancyhdr 宏包进行了补丁，载入 fancyhdr 后，其 fancy 页面格式将使用 \CTEXthechapter 等宏显示中文章节编号。

关于 fancyhdr 的具体用法可以参见其宏包手册。通常也只要像在标准的英文文档类中使用 fancyhdr 一样定义页眉页脚格式即可，并不需要额外的定义。

下面我则给出一个与前例类似而稍复杂的例子，展示如何在文档中设置页眉内容与页眉的格式。

例 11

```
\documentclass{ctexart}
\ctexset{section={
    name={第,节},
    number=\arabic{section},
  }
}
\usepackage{fancyhdr}
\fancyhf{}
\lhead{\textnormal{\kaishu\rightmark}}
\rhead{--\ \thepage\ --}
\pagestyle{fancy}
% \sectionmark 的重定义需要在 \pagestyle 之后生效
\renewcommand\sectionmark[1]{%
  \markright{\CTEXifname{\CTEXthesection——}{}#1}}

\begin{document}

\section{天地玄黄}
\newpage

\section{宇宙洪荒}

\end{document}
```

本例的页眉效果大致如下(有页眉线):

第 1 节——天地玄黄 – 1 –

第 7 节　章节标题格式设置

CTEX 宏包对 LaTeX 的标准文档类(article、report 和 book)和 beamer 进行了扩充。当以 heading 选项调用 CTEX 宏包时(5.2 节),则会启用章节标题的格式设置功能。本节就来介绍有关章节标题的格式选项,所有选项使用 \ctexset 命令设置。

第 6.3 节和本节介绍的功能已经被提取到 ctexheading 宏包之中,可以在 ctex 宏包和 ctexart 等文档类之外独立使用。各选项的默认值与 scheme = plain 时的情形相同。

章节标题的格式选项是分层设置的。顶层的选项是章节标题名称,次一级的选项是章节标题的格式。章节标题名包括 part, chapter, section, subsection, subsubsection, paragraph, subparagraph;而可用的格式包括 numbering, name, number, format, nameformat, numberformat, aftername, titleformat, aftertitle, runin, afterindent, beforeskip, afterskip, fixskip, lotskip, lofskip, indent, hang, pagestyle, break, tocline 等。

注意,对 article 及其衍生的 ctexart 等文档类,没有 chapter 级别的标题。而对于 beamer 文档类,这些选项控制的是由 \partpage, \sectionpage 和 \subsectionpage 产生的标题格式,此时只有 part, section 和 subsection 这三层级别,并且 runin, afterindent, fixskip, hang, break 和 tocline 这六个格式无效。

多级选项之间用斜线分开,例如,part/name 选项设置 \part 标题的在数字前后的名称,而 section/number 选项设置 \section 标题的数字类型。注意,斜线 / 前后不能有空格或者换行。

使用 \ctexset 设置多级选项时,可以在同一个上级选项下设置多个下级选项。

例如,同时设置 part 一级标题的 pagestyle 选项,chapter 一级标题的 format 与 pagestyle 选项和 section 一级标题的 name 与 number 选项:

例 12

```
\ctexset {
  part/pagestyle = empty,
  chapter = {
    format    = \raggedright,
    pagestyle = empty,
  },
  section = {
    name   = {第,节},
    number = \chinese{section},
  }
}
```

part/numbering ⋆
chapter/numbering ⋆
section/numbering ⋆
subsection/numbering ⋆
subsubsection/numbering ⋆
paragraph/numbering ⋆
subparagraph/numbering ⋆

New: 2015-06-21

numbering = **true**|false

控制是否对章节标题编号。对各级标题的默认值均为 true。

我们知道，LaTeX 带星号的章节标题命令（如 \section*）不会对标题编号，但也不会将该没编号的标题编入目录中。本选项控制的是不带星号的标题命令是否编号。设置本选项为 false，除了不对标题编号以外，功能与正常标题一致，比如可以编目录，正确的 hyperref 目录超链接位置和页眉标记。

注意，章节标题的编号深度受 LaTeX 计数器 secnumdepth 的控制。numbering 选项在 secnumdepth 的控制下起作用。

如果没有特别说明，以下将用“...”代表各级章节标题名。

.../name ⋆

Updated: 2014-03-08

name = {⟨前名字⟩,⟨后名字⟩}
name = {⟨前名字⟩}

设置章节的名字。名字可以分为前后两部分，即章节编号前后的词语，两个词之间用一个半角逗号分开；也可以只有一部分，表示只有章节编号之前的名字。例如：

例 13

```
\ctexset{
  chapter/name = {第,章},
  section/name = {\S},
}
```

会使得 \chapter 标题使用形如“第一章”的名字，而 \section 标题则使用形如“§1”的名字。

表 5　name 选项的默认设置

标题名	scheme = chinese	scheme = plain	注
part	{第，部分}	{\partname\space}	原 \partname 为 Part
chapter	{第，章}	{\chaptername\space}	原 \chaptername 为 Chapter
section (beamer)	{}	{\sectionname\space}	原 \sectionname 为 \translate{Section}
section	同右	{}	
subsection (beamer)	{}	{\subsectionname\space}	原 \subsectionname 为 \translate{Subsection}
subsection	同右	{}	
subsubsection	同右	{}	
paragraph	同右	{}	
subparagraph	同右	{}	

.../number ⋆　number = {⟨数字输出命令⟩}

设置章节编号的数字输出格式。⟨数字输出命令⟩ 通常是对应章节编号计数器的输出命令，如 \thesection 或 \chinese{chapter} 之类。

例 14

```
\ctexset{
  section/number = \Roman{section}
}
```

number 选项的定义同时将控制对章节计数器的交叉引用。在引用计数器时，记录在 LaTeX 辅助文件中的是 number 选项的定义。

但是，number 选项不会影响计数器本身的输出。即设置 section/number 不会影响 \thesection 的定义。(但该选项会影响 \CTEXthesection 的定义，见后。)

表 6　number 选项的默认设置

标题名	scheme = chinese	scheme = plain	原 \the⟨标题⟩ 等价定义
part (beamer)	\chinese{part}	\insertromanpartnumber	意义为 \Roman{part}
part	\chinese{part}	\thepart	\Roman{part}
chapter	\chinese{chapter}	\thechapter	\arabic{chapter}
section (beamer)	同右	\insertsectionnumber	意义为 \arabic{section}
section	同右	\thesection	\arabic{section}
subsection (beamer)	\arabic{section}.\arabic{subsection}	\insertsubsectionnumber	意义为 \arabic{subsection}
subsection	同右	\thesubsection	\thesection.\arabic{subsection}
subsubsection	同右	\thesubsubsection	\thesubsection.\arabic{subsubsection}
paragraph	同右	\theparagraph	\thesubsubsection.\arabic{paragraph}
subparagraph	同右	\thesubparagraph	\theparagraph.\arabic{subparagraph}

\CTEXthepart
\CTEXthechapter
\CTEXthesection
\CTEXthesubsection
\CTEXthesubsubsection
\CTEXtheparagraph
\CTEXthesubparagraph

以 \CTEXthe 开头的这组宏给出结合了 name 与 number 选项的章节编号输出格式。例如在 scheme = chinese 时，默认章编号输出格式就是 \CTEXthechapter，形如“第一章”。

这组宏在 CTEX 文档类中将代替 \thechapter 等宏的作用，在章节中引用本章节的完整编号。例如用于帮助定义自定义的目录格式、页眉格式等。

\CTEXifname　\CTEXifname {⟨有名字时的格式命令⟩} {⟨无名字时的格式命令⟩}

New: 2016-09-18

\CTEXifname 用于根据当前章节的名字的有无设置不同的格式。

它可用于 format, titleformat, aftertitle, afterskip, indent 这五个选项和 \chapter 标题 beforeskip 选项的格式设置之中。也可用于帮助定义自定义的目录格式、页眉格式等。

例如，设置章的标题有名字时左对齐，无名字时居中对齐，并且在标题后画一条横线。

例 15

```
\ctexset{
  chapter/format = \CTEXifname{\raggedright}{\centering},
  chapter/aftertitle = \par\CTEXifname{}{\hrule},
}
```

.../format ⋆ format = {⟨格式命令⟩}
.../format+ ⋆ format+= {⟨格式命令⟩}

Updated: 2015-06-30

format 选项用于控制章节标题的全局格式，作用域为章节名字和随后的标题内容。可以用于控制章节标题的对齐方式、整体字体字号等格式。

带加号的 format+ 选项则用于在已有格式之后追加新的格式命令。

例如，设置章格式为无衬线字体左对齐，为节格式增加无衬线字体设置：

例 16

```
\ctexset{
  chapter/format  = \sffamily\raggedright,
  section/format += \sffamily
}
```

表 7　format 选项的默认设置

标题名	scheme = chinese	scheme = plain
part (article)	\Large\bfseries\centering	\raggedright
part (beamer)	同右	\centering
part	\huge\bfseries\centering	\centering
chapter	\huge\bfseries\centering	\raggedright
section (beamer)	同右	\centering
section	\Large\bfseries\centering	\Large\bfseries
subsection (beamer)	同右	\centering
subsection	同右	\large\bfseries
subsubsection	同右	\normalsize\bfseries
paragraph	同右	\normalsize\bfseries
subparagraph	同右	\normalsize\bfseries

.../nameformat ⋆ nameformat = {⟨格式命令⟩}
.../nameformat+ ⋆ nameformat+= {⟨格式命令⟩}

Updated: 2015-06-30

nameformat 用于控制章节名字的格式，作用域为章节名字，包括编号。它一般用于章节名（包括编号）与章节标题的字体、字号等设置不一致的情形。参见 titleformat 选项。

nameformat+ 用于在已有的章节名字格式后附加内容。

nameformat 选项的最后一个格式命令可以是一个有一个参数的命令。这个命令的参数用于接受章节名字和编号，实现特殊效果（见例 21）。

nameformat 选项的默认值，在 scheme 选项的不同取值下相同。

.../numberformat ⋆ numberformat = {⟨格式命令⟩}
.../numberformat+ ⋆ numberformat+= {⟨格式命令⟩}

Updated: 2015-06-19

numberformat 选项用于控制章节编号的格式，作用域仅为编号数字本身。对各级标题默认均为空，当你需要编号的格式和前后的章节名字不一样时可以使用。

numberformat+ 选项用于在原有编号格式后面附加格式命令。

numberformat 选项的最后一个格式命令可以是一个有一个参数的命令。这个命令的参数用于接受编号数字。

例如，我们可以使用 numberformat 特别强调章标题中的数字：

例 17

```
\ctexset{
  chapter/number = \arabic{chapter},
```

表 8　nameformat 选项的默认设置

标题名	scheme = chinese	scheme = plain
part (article)	{}	\Large\bfseries
part (beamer)	同右	\usebeamerfont{part name} \usebeamercolor[fg]{part name}
part	{}	\huge\bfseries
chapter	{}	\huge\bfseries
section (beamer)	同右	\usebeamerfont{section name} \usebeamercolor[fg]{section name}
section	同右	{}
subsection (beamer)	同右	\usebeamerfont{subsection name} \usebeamercolor[fg]{subsection name}
subsection	同右	{}
subsubsection	同右	{}
paragraph	同右	{}
subparagraph	同右	{}

```
  chapter/numberformat = \color{blue}\zihao{0}\itshape,
}
```

上面的代码在 scheme = chinese 时可以做出类似这样的章标题效果：

第 4 章

.../afternamе ⋆　afternamе = {⟨代码⟩}
.../afternamе+ ⋆　afternamе+= {⟨代码⟩}

Updated: 2014-03-08

afternamе 选项的参数 ⟨代码⟩ 将被插入到章节编号与其后的标题内容之间，用于控制格式变换。常用于控制章节编号与标题内容之间的距离，或者控制标题是否另起一行。

afternamе+ 选项用于在原有代码后面附加代码。

表 9　afternamе 选项的默认设置

标题名	scheme = chinese	scheme = plain
part (article)	\quad	\par\nobreak
part (beamer)	同右	\vskip 1em \par
part	同右	\par\vskip 20pt
chapter	\quad	\par\nobreak\vskip 20pt
section (beamer)	同右	\vskip 1em \par
section	同右	\quad
subsection (beamer)	同右	\vskip 1em \par
subsection	同右	\quad
subsubsection	同右	\quad
paragraph	同右	\quad
subparagraph	同右	\quad

.../titleformat ⋆ `titleformat = {⟨格式命令⟩}`
.../titleformat+ ⋆ `titleformat+= {⟨格式命令⟩}`

Updated: 2015-06-30

`titleformat` 选项用于控制标题内容的格式，作用域为章节标题内容。

`titleformat+` 选项用于在原有标题格式后面附加格式命令。

需要注意的是，`titleformat` 选项的最后一个格式命令可以是一个有一个参数的命令。这个命令的参数接受标题内容，用于实现特殊效果。例如，实现多行标题的居中悬挂对齐：

例 18

```
\usepackage{varwidth} %% 提供 varwidth 环境
\ctexset{
  chapter/name = {第,回},
  chapter/titleformat = \chaptertitleformat
}
\newcommand\chaptertitleformat[1]{%% 以标题内容为参数
  \begin{varwidth}[t]{.7\linewidth}#1\end{varwidth}}
......
\chapter{情中情因情感妹妹\\错里错以错劝哥哥}
```

上面的代码可以做出类似这样的章标题效果：

第三十四回　情中情因情感妹妹
错里错以错劝哥哥

表 10　`titleformat` 选项的默认设置

标题名	scheme = chinese	scheme = plain
part (article)	{}	\huge\bfseries
part (beamer)	同右	\usebeamerfont{part title}
part	{}	\Huge\bfseries
chapter	{}	\Huge\bfseries
section (beamer)	同右	\usebeamerfont{section title}
section	同右	{}
subsection (beamer)	同右	\usebeamerfont{subsection title}
subsection	同右	{}
subsubsection	同右	{}
paragraph	同右	{}
subparagraph	同右	{}

.../aftertitle ⋆ `aftertitle = {⟨代码⟩}`
.../aftertitle+ ⋆ `aftertitle+= {⟨代码⟩}`

New: 2015-06-19

`aftertitle` 选项的参数 ⟨代码⟩ 将被插入到章节标题内容之后。

`aftertitle+` 选项用于在原有代码后面附加代码。

`aftertitle` 选项的默认值，在 `scheme` 选项的不同取值下相同。

`sub3section` 或 `sub4section` 宏包选项（见 5.2 节）会影响 `aftertitle` 选项的默认值。

表 11　aftertitle 选项的默认设置

标题名	默认值
part	\par
chapter	\par
section	\@@par
subsection	\@@par
subsubsection	\@@par
paragraph	{}
(sub3section)	\@@par
(sub4section)	同上
subparagraph	{}
(sub4section)	\@@par

表 12　runin 选项的默认设置

标题名	默认值
part	无效
chapter	无效
section	false
subsubsection	false
subsubsection	false
paragraph	true
(sub3section)	false
(sub4section)	同上
subparagraph	true
(sub4section)	false

section/runin ⋆
subsection/runin ⋆
subsubsection/runin ⋆
paragraph/runin ⋆
subparagraph/runin ⋆

New: 2015-06-27

runin = true|false

runin 选项只对 \section 级以下标题有意义。用于确定标题与随后的正文是否排在同一段之上。

runin 选项的默认值，在 scheme 选项的不同取值下相同。

默认情况下，\paragraph、\subparagraph 两级标题是与后面正文排在同一段的，runin 选项为 true；但使用 sub3section 或 sub4section 宏包选项（见 5.2 节）后，将对这两级标题设 runin 选项为 false，这两级标题会改为排在不同段。

.../afterindent ⋆

New: 2015-06-27

afterindent = true|false

afterindent 选项用于设置章节标题后首段的缩进。

book 和 report 类的 \part 标题被单独排在一页之上，afterindent 选项没有意义。

对于 \section 级以下标题，若设置了 runin 选项为 true，即标题与随后正文排在同一段，afterindent 选项也就没有了意义。

表 13　afterindent 选项的默认设置

标题名	scheme = chinese	scheme = plain
part (article)	true	false
part	无效	无效
chapter	true	false
section	true	false
subsection	true	false
subsubsection	true	false
paragraph	true	false
subparagraph	true	false

.../beforeskip ⋆

Updated: 2016-05-10

beforeskip = {⟨弹性间距⟩}

beforeskip 选项用于设置章节标题前的垂直间距。

beforeskip 选项的默认值，在 scheme 选项的不同取值下相同。

表 14　beforeskip 选项的默认设置

标题名	默认值
part (article)	4ex
part (beamer)	0pt
part	0pt plus 1fil
chapter	50pt
section (beamer)	0pt
section	3.5ex plus 1ex minus .2ex
subsection (beamer)	0pt
subsection	3.25ex plus 1ex minus .2ex
subsubsection	3.25ex plus 1ex minus .2ex
paragraph	3.25ex plus 1ex minus .2ex
subparagraph	3.25ex plus 1ex minus .2ex

表 15　afterskip 选项的默认设置

标题名	默认值
part (article)	3ex
part (beamer)	0pt
part	0pt plus 1fil
chapter	40pt
section (beamer)	0pt
section	2.3ex plus .2ex
subsection (beamer)	0pt
subsection	1.5ex plus .2ex
subsubsection	1.5ex plus .2ex
paragraph	1em
(sub3section)	1ex plus .2ex
(sub4section)	同上
subparagraph	1em
(sub4section)	1ex plus .2ex

.../afterskip ⋆　afterskip = {⟨弹性间距⟩}

Updated: 2015-06-27

afterskip 选项控制章节标题与后面下方之间的距离。

对于 \section 级以下标题，runin 选项会影响 afterskip 选项的意义：若 runin 为 true，标题与随后正文排在同一段，⟨弹性间距⟩ 给出水平间距。否则，正文另起一段，⟨弹性间距⟩ 给出的是垂直间距。

afterskip 选项的默认值，在 scheme 选项的不同取值下相同。

sub3section 或 sub4section 宏包选项（见 5.2 节）会影响 aftertitle 选项的默认值。

.../fixskip ⋆　fixskip = true|**false**

New: 2016-06-03

默认情况下，article、book 和 report 类的标题与正文的距离除了由 beforeskip 和 afterskip 选项设置的垂直间距外，还会有一些多余的间距。fixskip 选项用于抑制这些多余间距。

chapter/lofskip ⋆　lofskip = {⟨弹性间距⟩}
chapter/lotskip ⋆　lotskip = {⟨弹性间距⟩}

New: 2016-10-01

lofskip 选项控制插图目录（.lof）中，章之间的插图标题的距离。

同样，lotskip 选项控制表格目录（.lot）中，章之间的表格标题的距离。

目前，这两个选项只在 chapter 标题下有定义。他们的默认值，在 scheme 选项的不同取值下都为 10 pt。

.../indent ⋆　indent = {⟨缩进间距⟩}

Updated: 2015-06-27

indent 选项用于设置章节标题本身的首行缩进。

indent 选项的默认值，在 scheme 选项的不同取值下相同。

如果 indent 的值是以 em、ex 或 \ccwd 为单位，那么缩进间距的大小是相对于 format 中指定的字号大小。

例如，设置 \part 标题缩进三个字、\section 标题缩进 20 pt：

例 19

```
\ctexset{
  part={
    format+=\raggedright,
```

```
    indent=3\ccwd,
  },
  section={
    format=\Large\bfseries,
    indent=20pt,
  }
}
\part{首行缩进的标题}
\noindent 无缩进的正文。
\section{首行缩进的标题}
\noindent 无缩进的正文。
```

表 16　indent 选项的默认设置

标题名	默认值
part	0pt
chapter	0pt
section	0pt
subsection	0pt
subsubsection	0pt
paragraph	0pt
subparagraph	\parindent
(sub3section)	0pt
(sub4section)	同上

表 17　hang 选项的默认设置

标题名	默认值
part	false
chapter	false
section	true
subsection	true
subsubsection	true
paragraph	无意义
(sub3section)	true
(sub4section)	true
subparagraph	无意义
(sub4section)	true

part/hang ⋆
chapter/hang ⋆
section/hang ⋆
subsection/hang ⋆
subsubsection/hang ⋆
paragraph/hang ⋆
subparagraph/hang ⋆

Updated: 2019-03-31

hang = true|false

hang 选项用于设置是否对章节标题实施悬挂缩进（缩进的宽度为名字宽度和 indent 选项设置的宽度之和）。

本选项对 beamer/ctexbeamer 文档类无效。对于 \section 级以下标题，若设置了 runin 选项为 true，即标题与随后正文排在同一段，hang 选项没有意义。

part/pagestyle ⋆
chapter/pagestyle ⋆

New: 2014-03-21

pagestyle = {⟨页面格式⟩}

设置 book/ctexbook 或 report/ctexrep 文档类中，\part 与 \chapter 标题所在页的页面格式（page style）。

表 18　pagestyle 选项的默认设置

标题名	默认值
part (article)	无效
part	plain
chapter	plain

.../break ⋆ `break = {⟨格式命令⟩}`
.../break+ ⋆ `break+= {⟨格式命令⟩}`

New: 2016-09-19

break 选项用于控制章节标题与之前正文的分隔关系。一般用于设置是否在标题之前分页或者设置行间罚点。

带加号的 break+ 选项则用于在已有格式之后追加新的格式命令。

break 选项的默认值，在 scheme 选项的不同取值下相同。

例如，若当前页剩余高度小于正文高度的一半时，则另起一页输出 \section 标题：

例 20

```
\usepackage{needspace}
\ctexset{section/break = \Needspace{.5\textheight}}
```

表 19　break 选项的默认设置

标题名	默认值
part (article)	{}
part	\if@openright\cleardoublepage\else\clearpage\fi
chapter	同上
section	\addpenalty{\@secpenalty}
subsection	同上
subsubsection	同上
paragraph	同上
subparagraph	同上

.../tocline ⋆ `tocline = {⟨格式定义⟩}`

New: 2016-10-25

tocline 选项用于定义章节标题在目录文件（.toc）中的格式。⟨格式定义⟩有两个参数：参数 #1 是 part、chapter 等名字，参数 #2 是标题内容。

表 20　tocline 选项的默认设置

标题名	默认值
part	\CTEXifname{\CTEXthepart\hspace{1em}}{}#2
chapter (chinese)	\CTEXifname{\protect\numberline{\CTEXthechapter\hspace{.3em}}}{}#2
chapter (plain)	\CTEXnumberline{#1}#2
section	\CTEXnumberline{#1}#2
subsection	同上
subsubsection	同上
paragraph	同上
subparagraph	同上

其中 \CTEXnumberline 的意义是若标题 #1 没有名字，则不输出 \numberline{\CTEXthe#1} 等编号：\CTEXifname{\protect\numberline{\csname CTEXthe#1\endcsname}}{}

appendix/numbering ⋆ `numbering = true|false`

New: 2015-06-21

控制是否对附录章（对 book 与 report）或附录节（对 article）进行编号。

用法与普通章节 numbering 选项类似。

appendix/name ⋆
Updated: 2014-03-08

name = {⟨前名字⟩,⟨后名字⟩}
name = {⟨前名字⟩}

设置附录章（对 book 与 report）或附录节（对 article）的名字。

用法与普通章节 name 选项类似。

注意该选项与 appendixname 选项（6.2 节）在意义上有些重叠，但意义不同。appendixname 选项只用来重定义 \appendixname，而不管 \appendixname 如何使用；该选项则决定在章节标题中输出的名字，可以调用 \appendixname 设置。

表 21　appendix/name 选项的默认设置

文档类	影响命令	scheme = chinese	实际定义	scheme = plain	实际定义
article	\section	{}		{}	
book, report	\chapter	\appendixname\space	附录	\appendixname\space	Appendix

appendix/number ⋆

number = {⟨数字输出命令⟩}

设置附录章（对 book 与 report）或附录节（对 article）编号的数字输出格式。

用法与普通章节的 number 选项类似。

该选项也同时控制附录章节计数器的交叉引用。

与普通章节的 number 选项类似，同样需要注意，该选项不会影响计数器本身的输出，即不影响 \thesection 或 \thechapter 的定义。

表 22　appendix/number 选项的默认设置

文档类	影响命令	默认值
article	\section	\Alph{section}
book, report	\chapter	\Alph{chapter}

我们最后举一个稍微复杂的例子，来看看上述选项的综合应用。

例 21

```
\ctexset {
  chapter = {
    beforeskip = 0pt,
    fixskip    = true,
    format     = \Huge\bfseries,
    nameformat = \rule{\linewidth}{1bp}\par\bigskip\hfill\chapternamebox,
    number     = \arabic{chapter},
    aftername  = \par\medskip,
    aftertitle = \par\bigskip\nointerlineskip\rule{\linewidth}{2bp}\par
  }
}
\newcommand\chapternamebox[1]{%
  \parbox{\ccwd}{\linespread{1}\selectfont\centering #1}}
......
\chapter{熟悉 \LaTeX}
```

本例的设置效果大致如下：

第 1 章 熟悉 LaTeX

第 8 节　实用命令

8.1　字号与间距

\zihao

Updated: 2014-03-08

\zihao {⟨字号⟩}

用于调整字号大小。其中 ⟨字号⟩ 的有效值共有 16 个，如表 23 所示。使用 \zihao 命令调整字体大小时，西文字号大小会始终和中文字号保持一致。

表 23　中文字号

⟨字号⟩	大小 (bp)	大小 (pt)	意义
0	42	42.157 49	初号
-0	36	36.135	小初号
1	26	26.097 49	一号
-1	24	24.09	小一号
2	22	22.082 49	二号
-2	18	18.067 49	小二号
3	16	16.06	三号
-3	15	15.056 24	小三号
4	14	14.052 49	四号
-4	12	12.045	小四号
5	10.5	10.539 37	五号
-5	9	9.033 74	小五号
6	7.5	7.528 12	六号
-6	6.5	6.524 37	小六号
7	5.5	5.520 61	七号
8	5	5.018 74	八号

\ziju

Updated: 2014-03-28

\ziju {⟨中文字符宽度的倍数⟩}

用于调整相邻汉字之间的间距，即（在正常中文行文中）前一个汉字的右边缘与后一个汉字的左边缘之间的距离。其中参数可以是任意浮点数值；而中文字符宽度指的是实际汉字的宽度，不包含当前字距。

这个命令会影响 \ccwd 的值，但不会影响英文字距。

\ccwd
Updated: 2014-03-27

当前汉字的字宽保存在长度寄存器 \ccwd 之中。汉字字宽是相邻两个汉字中心之间的距离，包含字距在内。因此修改字距会间接修改字宽。

8.2 中文数字转换

CTeX 宏集的中文数字转换功能实际上是调用 zhnumber 宏包来完成。下面只介绍一些基本的用法，更高级的用法可以查阅 zhnumber 宏包的文档。

\chinese
Updated: 2016-05-01

\chinese {⟨*counter*⟩}
\pagenumbering {chinese}

\chinese 命令与 \roman 等命令的用法类似，作用在一个 LaTeX 计数器上，将计数器的值以中文数字的形式输出。

\zhnumber
New: 2014-03-08

\zhnumber {⟨*number*⟩}

以中文格式输出数字。这里的数字可以是整数、小数和分数。

\zhdigits
New: 2014-03-08

\zhdigits {⟨*number*⟩}

将阿拉伯数字转换为中文数字串。

\CTEXnumber

\CTEXnumber \⟨*macro*⟩ {⟨*number*⟩}

\⟨*macro*⟩ 必须是一个 TeX 宏，不需预先定义。\CTEXnumber 通过 \zhnumber 将 ⟨*number*⟩ 转为中文数字，最后将结果存储在 \⟨*macro*⟩ 里。对 \⟨*macro*⟩ 的定义是局部的，将它展开一次就可以得到转换结果。

一般来说，并不需要使用 \CTEXnumber，直接使用 \zhnumber 即可。但是，如果在文档中需要多次使用同一个数字 ⟨*number*⟩ 的中文形式，就可以先用 \CTEXnumber 将结果保存起来备用，而不是每次使用时都用 \zhnumber 现场转换一次。

\CTEXdigits

\CTEXdigits \⟨*macro*⟩ {⟨*number*⟩}

\CTEXdigits 与 \CTEXnumber 类似，但其转换的结果是中文数字串，而不是中文数字。

8.3 杂项

\CTeX

用于显示 CTeX 标志。

第 9 节 LuaLaTeX 下的中文支持方式

在 LuaLaTeX 下，CTeX 宏集依赖 LuaTeX-ja 宏包来完成中文支持。该宏包是日本 TeX 社区的北川弘典、前田一贵、八登崇之等人开发的，设计目的主要是在 LuaTeX 引擎下实现日本 pTeX 引擎的（大部分）功能。它为了兼容 pLaTeX 的使用习惯，对 LaTeX 2_ε 的 NFSS 作了不少修改和扩充。这对于简体中文用户来说不是必要的，因而 CTeX 禁用了它在 LaTeX 格式下的大部分设置，只保留了必要的部分。同时修改了它的字体设置方式，使得相关命令与 xeCJK 宏包大致相同。

20150420 版以后的 LuaTeX-ja 宏包开始支持竖排，但 CTeX 暂不支持竖排。

9.1 LuaLaTeX 下替代字体的设置

AlternateFont
New: 2014-04-14

```
\setCJKfamilyfont {⟨family⟩}
  [
    AlternateFont =
      {
        {⟨character range_1⟩} [⟨alternate font features_1⟩] {⟨alternate font name_1⟩} ||
        {⟨character range_2⟩} [⟨alternate font features_2⟩] {⟨alternate font name_2⟩} ||
        ......
      } ,
    ⟨base font features⟩
  ] {⟨base font name⟩}
```

在设置字体族 ⟨*family*⟩ 的时候，同时设置该字体族在字符范围 ⟨*character range*$_n$⟩ 内，对应字形的替代字体。

CharRange
New: 2014-04-14

```
\setCJKfamilyfont {⟨family⟩}
  [
    CharRange = {⟨character range⟩} ,
    ⟨alternate font features⟩
  ] {⟨alternate font name⟩}
```

只设置字体族 ⟨*family*⟩ 在字符范围 ⟨*character range*⟩ 内，对应字形的替代字体。

一个 \setCJKfamilyfont 里只能使用一次 CharRange 或者 AlternateFont，但可以将它们分开重叠使用。例如下面的方式是有效的。

例 22

```
\setCJKmainfont[AlternateFont={...}{...}, ...]{...}
\setCJKmainfont[CharRange={"4E00->"67FF,-2}, ...]{...}
\setCJKmainfont[CharRange={"6800->"9FFF}, ...]{...}
```

declarecharrange ⋆
New: 2014-04-14

```
\ctexset
  {
    declarecharrange =
      {
        {⟨name_1⟩} {⟨character range_1⟩} ,
        {⟨name_2⟩} {⟨character range_2⟩} ,
        ...
      }
  }
```

预先声明字符范围。声明字符范围 ⟨*name*⟩ 之后，它的名字 ⟨*name*⟩ 可以用在 AlternateFont 和 CharRange 选项的 ⟨*character range*⟩ 之中，表示对应的字符范围。

在声明字符范围 ⟨*name*⟩ 的同时，还为 \setCJKmainfont 等字体设置命令定义了选项 ⟨*name*⟩，用于设置对应字符的替代字体：

```
⟨name⟩ = [⟨alternate font features⟩] {⟨alternate font name⟩}
```

⟨*name*⟩ 选项可以与 AlternateFont 共同使用，但不能与 CharRange 一起使用。如果没有给 ⟨*name*⟩ 设置值，则等价于设置 CharRange=⟨*name*⟩，即只设置 ⟨*name*⟩ 对应的字符范围的替代字体。

clearalternatefont ⋆
resetalternatefont ⋆

New: 2014-04-15

```
\ctexset
  {
    clearalternatefont = {⟨family1, family2, ...⟩} ,
    resetalternatefont = {⟨family1, family2, ...⟩} ,
    clearalternatefont ,
    resetalternatefont
  }
```

清除与重置 CJK 字体族 ⟨*family*⟩ 的替换字体设置。如果没有给定值，则作用于当前 CJK 字体族。清除与重置操作总是全局的。

第 10 节 CTEX 宏集的配置文件

CTEX 宏集提供了不同的配置文件，可以通过修改配置文件来改变 CTEX 宏集的默认行为。

在多数情况下，并不需要修改配置文件，CTEX 宏集的默认设置已经能满足大多数用户的需要。不恰当地修改 CTEX 宏集的默认行为也可能导致同一文件在别处无法正常编译或排版效果完全不同，因此修改应该慎重。

但在一些情况下，直接修改配置文件仍是必要的，例如：

- 系统没有安装默认设置的字体文件，无法编译。
- 需要经常编译来自其他系统的中文 TEX 文件，但对方的操作系统或默认设置与本机不同。

与 CTEX 宏集的源代码一样，配置文件采用 LATEX3 的语法编写。

CTEX 宏集的配置文件随宏包其他文件一起安装在 TEX 系统 TDS 目录树中，文件后缀是 `.cfg`。为了避免本地配置文件内容因 CTEX 宏集的更新而丢失，不要直接修改系统 TDS 目录树中的配置文件，而应该将系统自带的配置文件复制到本地的或用户私有的 TDS 目录树中修改，并运行 `texhash` 命令刷新文件名数据库。

例如对于 TEX Live，系统自带的配置文件就在 TEX Live 安装目录下的 `texmf-dist/tex/latex/ctex/config/` 子目录下，可以修改它的副本，保存在本地 TDS 树的 `texmf-local/tex/latex/ctex/` 目录下，或者用户 TDS 树的 `~/.texlive2015/texmf-var/tex/latex/ctex/` 目录下，作为本地/用户专有的配置文件。复制配置文件后需要运行 `texhash` 命令使本地配置文件生效。

MiKTEX 的配置文件也保存在类似的目录结构中，MiKTEX 管理的几个 TDS 根目录可以在 MiKTEX Options 设置项中查看到，这里不再赘述。

除了修改本地 TEX 系统中的配置文件，对于特定文档，也可以将修改过的配置文件保存在文档的工作目录下。此时配置文件就只对工作目录下的所有文档生效。

10.1 修改宏包默认选项

配置文件 `ctexopts.cfg` 可以用来修改宏包的默认选项。随系统安装的配置文件除了文件信息声明外没有实际的内容，但在注释中给出了一个简单的示例，只要取消注释就可以生效。

例 23

```
% 系统自带 ctexopts.cfg 注释中的示例语句，固定默认字体集为 windowsnew。
% 该设置可以用在安装了 Windows 字体的非 Windows 系统中。
\keys_set:nn { ctex / option } { fontset = windowsnew }
```

如上例所示，宏包选项通常使用 LaTeX3 的 \keys_set:nn 命令完成键值设置，第一个参数是固定的子模块 ctex/option，第二个参数中是用户定义的新的默认宏包选项。

ctexopts.cfg 中的设置将在 CTeX 宏集的开始处，定义过宏包选项之后，\ProcessKeysOptions 命令之前生效。最好只使用此配置文件修改宏包默认选项。

10.2　宏包载入后的配置

配置文件 ctex.cfg 将在宏包的末尾被载入生效。可以用它完成任意的设置，或是覆盖已有的定义。随系统安装的配置文件除版本信息外没有实际内容，注意配置文件中也使用 LaTeX3 语法。

例 24

```
% 简单的 ctex.cfg 内容示例。
% 修改默认的页面格式设置。
\pagestyle{plain}
```

例 25

```
% 略复杂的 ctex.cfg 内容示例：禁止段末孤字成行。
% 在使用 XeTeX 编译时，打开 xeCJK 的 CheckSingle 选项。
\sys_if_engine_xetex:T
  {
    \xeCJKsetup { CheckSingle }
  }
% 在使用 LuaTeX 编译时，设置 LuaTeX-ja 的 jcharwidowpenalty 参数。
\sys_if_engine_luatex:T
  {
    \ltjsetparameter { jcharwidowpenalty = 10000 }
  }
```

10.3　配置标题中文翻译

由于 CTeX 宏集需要同时支持 GBK 和 UTF-8 两种编码，因此对标题的中文翻译写在两个配置文件当中：ctex-name-gbk.cfg 和 ctex-name-utf8.cfg。两个文件的设置相同，只是编码不同。

为了同一文档在不同电脑上编译效果的一致性，通常不建议修改默认的中文翻译。

10.4　自定义字体集

4.3 节介绍的用于 fontset 选项的自定义字库文件，类似于 CTeX 宏集的配置文件，也应该与其他本地配置文件一起保存在本地 TDS 目录树下，并可以配合 ctexopts.cfg 等配置文件使用。

第 11 节　对旧版本的兼容性

11.1　CTeX 0.8a 及以前的版本

在 ctex-kit 项目成立之前，CTeX 宏包的最后一个版本是 CTeX 0.8a（2007/05/06）。

第 2 版未考虑对这些很早版本的兼容性。

11.2　CTEX 0.9–CTEX 1.0d

在 2009 年在 ctex-kit 项目成立后，新增了 XETEX 引擎的支持，并增加了不少控制字体的命令和选项。

这里主要介绍新版本 CTEX 宏包相对 1.02d 版本（2014/06/09）的兼容性。

第 2 版的 CTEX 宏包已尽力保证对 1.0x 版本的兼容性，原有为 1.0x 编写的代码，在第 2 版的 CTEX 宏包下保证仍能编译，并且在大多数情况下保持编译效果不变。

CTEX 宏包在 0.8a 以前的版本支持以 CCT 作为底层中文支持方式，从 0.9 版之后即不再推荐使用，只保留向后兼容。在 CTEX 宏包第 2 版中则完全不再支持 CCT。

下面这些是在旧版本 CTEX 宏包中存在，而在新版本中已不建议使用的选项和命令，在未来版本中可能会删去它们的支持。

在多数情况下它们的功能仍将保留，但也有部分选项命令功能已失效。

`cs4size`
`c5size`

分别相当于 `zihao=-4` 和 `zihao=5`，过时选项。

`CCT`
`CCTfont`

相关选项已删除。

`indent`
`noindent`

`indent` 和 `noindent` 什么也不做，过时选项。

在中文版式下，ctex 宏包的相关功能在与标准文档类及其衍生文档类联用时默认打开。CTEX 文档类的相关功能由章节标题的 `afterindent` 选项的值来确定。

`zhmap`
`nozhmap`

`zhmap` 宏包选项增加了参数，扩充了功能，除了支持真假值参数外，还支持选择 zhmCJK 作为底层中文处理宏包。（4.3 节）

`nozhmap` 选项相当于 `zhmap=false`。过时选项。

`winfonts`
`adobefonts`
`nofonts`

宏包选项 `winfonts` 相当于 `fontset=windows`，`adobefonts` 相当于 `fontset=adobe`，`nofonts` 相当于 `fontset=none`。这几个选项是过时选项，对于新文档，应使用 `fontset` 选项设置不同字体集。

另外，第 2 版 CTEX 宏包的默认字体不再是 Windows 系统字体，而是根据检测到的操作系统选择使用 Windows、Mac 的系统字体还是 Fandol 字体（4.3 节）。

`punct`
`nopunct`

旧版本中宏包 `punct` 选项没有参数，现在可以用参数设定标点风格（5.3 节）。原有无参形式的 `punct` 选项相当于 `punct=quanjiao`。

旧版宏包中 `nopunct` 选项的效果大致相当于 `punct=plain`。过时选项，不推荐使用。

`cap`
`nocap`

原有的 `cap` 和 `nocap` 选项由新的 `scheme` 选项代替。（5.3 节）

`cap` 选项相当于 `scheme = chinese`，`nocap` 选项相当于 `scheme = plain`。它们均已过时，仅因兼容性而保留。

`space`
`nospace`

新版本宏包 `space` 选项增加真假值参数。（5.3 节）

`nospace` 选项相当于 `space=false`，成为过时选项。

`fancyhdr`

新版本宏包中总是自动处理对 fancyhdr 宏包的兼容性，而由用户自己使用 `\usepackage` 载入 fancyhdr 宏包。

`fancyhdr` 选项过时，因兼容性保留，功能是载入 fancyhdr 宏包。

hyperref

新版本宏包中总是自动处理对 hyperref 宏包的兼容性，而由用户自己使用 \usepackage 载入 hyperref 宏包。

hyperref 选项过时，因兼容性保留，功能是在导言区末尾载入 hyperref 宏包。

fntef

旧版本的 fntef 选项用于统一 CCTfntef 与 CJKfntef 的界面，新版本 CTEX 宏集不再支持 CCT，而是直接载入 CJKfntef 或 xeCJKfntef 宏包并禁用其彩色设置。该选项是过时选项。

\CTEXunderdot
\CTEXunderline
\CTEXunderdblline
\CTEXunderwave
\CTEXsout
\CTEXxout
\CTEXfilltwosides

在调用 fntef 宏包选项的同时，旧版本 CTEX 宏包由于需要支持 CCT 系统，会将以 \CJK 开头的 \CJKunderline 等宏换名为以 \CTEX 开头的 \CTEXunderline 等宏。此功能在新版本的 CTEX 宏集中已失去意义。

此外，在 pdfTEX 引擎下，用于设置格式的 \CJKunderdotbasesep 等宏也被换名为 \CTEXunderdotbasesep 等宏。

在新版本中，上述由 fntef 衍生的相关宏都成为过时命令。

\CTEXsetfont

更新当前的中文字体信息，包括当前字距（\ccwd）和段首缩进（\parindent）。一般来说，用户无需使用这个命令。

\CTEXindent

更新 \ccwd 宽度后设置 \parindent=2\ccwd。过时命令。

\CTEXnoindent

设置 \parindent=0pt。过时命令。

\CTEXsetup

\CTEXsetup[⟨选项⟩]{⟨标题⟩}

相当于设置了 \ctexset{ ⟨标题⟩ = {⟨选项⟩} }。过时命令。

\CTEXoptions

\CTEXoptions[⟨选项⟩]

相当于设置了 \ctexset{⟨选项⟩}。过时命令。

\Chinese

\Chinese{⟨*counter*⟩}

新版宏集中 \chinese 统一了旧版本中 \chinese 和 \Chinese 的功能。因此，该命令已过时。

captiondelimiter

原为 \CTEXoptions 命令的选项，用于控制 \caption 编号后面的标点。此选项已过时，并在新版本的 CTEX 宏包中失效。

可以使用 caption 宏包的 labelsep 选项来完成同样的功能。

例 26

```
% 代替 \CTEXoptions[captiondelimiter={:}]
\usepackage{caption}
\captionsetup{labelsep=colon}
```

11.3　CTEX 1.02c 以后的 SVN 开发版

CTEX 宏包在 1.02c 版本（2011/03/11）之后在 Google code 上的 SVN 开发版本，内部版本号一直升到 1.11 版，但从未正式发布。SVN 开发版在 1.02c 版本的基础上新增的功能在第 2 版中大多继承了过来，但新增的命令与选项都不再保持兼容。

CTEX 宏包第 2 版不保证对未发布的 SVN 开发版兼容。

11.4　CTEX 2.2 之前的版本

part/beforeskip
chapter/beforeskip
section/beforeskip
subsection/beforeskip
subsubsection/beforeskip
paragraph/beforeskip
subparagraph/beforeskip

在 CTEX 2.2 之前的版本中，beforeskip 选项的符号还用于确定章节标题后首段的缩进。当 beforeskip 是负值时，章节标题后的第一段按英文文档的排版习惯，没有首行缩进，否则保留首行缩进。

这一特性在 2.2 版和后续版本中不再保留，相应的功能通过新的 afterindent 选项来设置。如果原先设置 beforeskip 为负值，在新版本中需要改为正值，并设置相应的 afterindent 选项为 false。

section/afterskip
subsection/afterskip
subsubsection/afterskip
paragraph/afterskip
subparagraph/afterskip

在 CTEX 2.2 之前的版本中，对于 \section 级以下标题，afterskip 选项的符号用于确定标题与随后正文是否排在同一段。如果是正值，则正文另起一段，否则标题与随后正文排在同一段，afterskip 的绝对值给出水平间距。

这一特性在 2.2 版和后续版本中不再保留，相应的功能通过新的 runin 选项来设置。如果原先设置 afterskip 为负值，在新版本中需要改为正值，并设置相应的 runin 选项为 true。

11.5　CTEX 2.4.1 和 2.4.2

part/fixbeforeskip
chapter/fixbeforeskip

这两个选项已经被删除，相应功能由新的选项 fixskip 提供。

第 12 节　宏集依赖情况与手工安装方法

本节介绍 CTEX 宏集的依赖情况，并介绍手工编译安装的具体方法。通常用户只需参照第 2.2 节介绍的方法，使用发行版自带的宏包管理器安装本宏集。

CTEX 宏集有两个源文件：ctex.dtx、ctexpunct.spa。使用不同的编译方式时，CTEX 依赖的宏包略有不同。在手工安装 CTEX 宏集之前，请确保你的 TEX 发行版中已经正确安装了这些宏包。CTEX 依赖宏包的详情叙述如下：

- expl3、xparse 和 l3keys2e 宏包。它们属于 l3kernel 和 l3packages 宏集。
- indentfirst 宏包，属于 tools 宏集。
- everysel 宏包，属于 ms 宏集。
- ulem 宏包。
- zhnumber 宏包。

➥ 以上是各种编译方式都必需的依赖项。

- CJK 宏集。
- CJKpunct 宏包。
- xCJK2uni 宏包。
- zhmetrics 宏包。
- zhmCJK 宏包，它还依赖 oberdiek 宏集。

➥ 以上是使用 pdfLATEX 或 LATEX + DVIPDFM*x* 的编译方式所需要的依赖项，其中 zhmCJK 是可选的。

- xeCJK 宏集，它还依赖
  - xtemplate 宏包，它属于 l3packages 宏集。
  - fontspec 宏包。
- environ 宏包，它还依赖 trimspaces 宏包。

➥ 以上是使用 XƎLATEX 编译时的依赖项。

- luatexja 宏包，它还依赖
  - adobemapping 宏包。
  - lualibs 宏包。
  - luaotfload 宏包。
  - luatexbase 宏包，它还依赖 ctablestack 宏包。
  - oberdiek 宏集。
  - xkeyval 宏包。
  - etoolbox 宏包。
- fontspec 宏包。
- xunicode-addon 宏包，属于 xeCJK 宏集，它还依赖
  - xunicode 宏包，它还依赖
    - graphics 宏集。
    - graphics-cfg 宏包。
    - graphics-def 宏包。

➥ 以上是使用 LuaLaTeX 编译时的依赖项。

- pxeverysel 宏包，属于 platex-tools 宏集。
- zhmetrics-uptex 宏包。

➥ 以上是使用 upLaTeX 编译时的依赖项。

出于一些原因，zhmCJK 尚未被收入 TeX Live 和 MiKTeX。因此，若你希望使用 zhmCJK 作为 CTeX 宏集的底层中文支持方式，那么你需要自行安装该宏包。zhmCJK 的安装较为复杂。我们建议你

1. 从 CTAN 下载 zhmCJK 宏包的 TDS 安装包，
2. 按目录结构将文件复制到 TeX 发行版的本地 TDS 根目录，
3. 最后执行 `texhash` 刷新 TeX 发行版的 ls-R 数据库以完成安装。

其他细节，可参照其 宏包手册 中第 3 节的指导。

CTeX 宏集已被 TeX Live 和 MiKTeX 收录，若无特别理由，我们强烈建议用户使用宏包管理器安装本宏集。

若要手工安装，请遵循如下步骤：

1. 从 CTAN 下载 CTeX 宏集的 TDS 安装包，
2. 按目录结构将文件复制到 TeX 发行版的本地 TDS 根目录，
3. 最后执行 `texhash` 刷新 TeX 发行版的 ls-R 数据库以完成安装。

第 13 节　开发人员

- 吴凌云 (`aloft@ctex.org`)
- 江疆 (`gzjjgod@gmail.com`)
- 王越 (`yuleopen@gmail.com`)
- 刘海洋 (`LeoLiu.PKU@gmail.com`)
- 李延瑞 (`LiYanrui.m2@gmail.com`)
- 陈之初 (`zhichu.chen@gmail.com`)
- 李清 (`sobenlee@gmail.com`)
- 黄晨成 (`liamhuang0205@gmail.com`)

目前比较活跃的开发维护人员是刘海洋、李清和黄晨成。

参考文献

[1] Donald Ervin Knuth. *The TEXbook, Computers & Typesetting*, volume A. Addison-Wesley, 1986

[2] Frank Mittelbach and Michel Goossens. *The LATEX Companion*. Tools and Techniques for Computer Typesetting. Boston: Addison-Wesley, second edition, 2004

第 14 节　代码实现

```
⟨@@=ctex⟩
```

宏包载入检查。

```
⟨*class|ctex⟩
\tl_const:Nx \c__ctex_version_tl
  { \cs_if_exist_use:cF { ver@ \@currname . \@currext } { 9999/99/99 } }
⟨*class⟩
\cs_new_eq:cN { ver@ctex.        \@pkgextension } \c__ctex_version_tl
\cs_new_eq:cN { ver@ctexcap.     \@pkgextension } \c__ctex_version_tl
\cs_new_eq:cN { ver@ctexsize.    \@pkgextension } \c__ctex_version_tl
\cs_new_eq:cN { ver@ctexheading. \@pkgextension } \c__ctex_version_tl
⟨/class⟩
⟨*ctex⟩
\msg_new:nnnn { ctex } { subpackage-loaded }
  { Package~`#1'~can~not~be~loaded~with~`ctex'. }
  {
    `#1'~is~actually~a~part~of~`ctex'.\\
    It~is~not~necessary~to~load~it~separately.
  }
\@ifpackageloaded { ctexsize }
  { \msg_error:nnn { ctex } { subpackage-loaded } { ctexsize } }
  { \cs_new_eq:cN { ver@ctexsize. \@pkgextension } \c__ctex_version_tl }
\@ifpackageloaded { ctexheading }
  { \msg_error:nnn { ctex } { subpackage-loaded } { ctexheading } }
  { \cs_new_eq:cN { ver@ctexheading. \@pkgextension } \c__ctex_version_tl }
⟨/ctex⟩
⟨/class|ctex⟩
```

```
⟨*class|style⟩
\RequirePackage { xparse , l3keys2e }
⟨/class|style⟩
```

```
⟨*class|ctex⟩
```

检查 expl3 和 l3keys2e 的版本。

```
\msg_new:nnnn { ctex } { l3-too-old }
  { Support~package~`#1'~too~old. }
  {
    Please~update~an~up~to~date~version~of~the~bundles\\\\
    `l3kernel'~and~`l3packages'\\\\
    using~your~TeX~package~manager~or~from~CTAN.
  }
\@ifpackagelater { expl3 } { 2019/03/05 } { }
  { \msg_error:nnn { ctex } { l3-too-old } { expl3 } }
⟨*class⟩
\@ifpackagelater { l3keys2e } { 2015/12/20 } { }
  { \msg_error:nnn { ctex } { l3-too-old } { l3keys2e } }
⟨/class⟩
```

\c__ctex_engine_str
\c__ctex_engine_file_str

引擎检查。目前 LaTeX3 将 ApTeX 识别为 upTeX。

```
\str_const:Nx \c__ctex_engine_str
  { \cs_if_exist:NTF \ngostype { aptex } { \c_sys_engine_str } }
\msg_new:nnnn { ctex } { engine-not-supported }
  { Engine~`#1'~is~not~yet~supported,~ctex~will~abort! }
  { You~can~switch~to~xelatex,~lualatex,~pdflatex,~uplatex,~or~aplatex. }
\file_if_exist:nTF { ctex-engine- \c__ctex_engine_str .def }
  {
    \str_const:Nx \c__ctex_engine_file_str
      { ctex-engine- \c__ctex_engine_str .def }
  }
  { \msg_critical:nnx { ctex } { engine-not-supported } { \c__ctex_engine_str } }
```

```
⟨/class|ctex⟩
```

```
⟨*class|ctex|ctexheading⟩
```

```
\RequirePackage { ctexhook , ctexpatch }
⟨!ctexheading⟩ \RequirePackage { fix-cm , everysel }
```

14.1　内部函数与变量

\l__ctex_tmp_tl
\l__ctex_tmp_int
\l__ctex_tmp_box
\l__ctex_tmp_dim

临时变量。

```
\tl_clear_new:N \l__ctex_tmp_tl
\int_new:N \l__ctex_tmp_int
\box_new:N \l__ctex_tmp_box
⟨!ctexheading⟩ \dim_new:N \l__ctex_tmp_dim
```

\ctex_file_wrapper:nnn

设置文件操作的 \catcode 环境，参数 #1 是设置，#2 是文件操作，#3 是恢复。默认关闭 LaTeX3 语法环境，并设置 @ 的 \catcode 为 11。

```
\cs_new_protected:Npn \ctex_file_wrapper:nnn #1#2#3
  {
    \use:x
      {
        \ExplSyntaxOff
        \char_set_catcode_letter:n { 64 }
        #1
        \exp_not:n {#2}
        \bool_if:NTF \l__kernel_expl_bool
          { \ExplSyntaxOn }
          { \ExplSyntaxOff }
        \char_set_catcode:nn { 64 } { \char_value_catcode:n { 64 } }
        #3
      }
  }
```

\ctex_file_input:n

输入文件。

```
\cs_new_protected_nopar:Npn \ctex_file_input:n #1
  { \ctex_file_wrapper:nnn { } { \file_input:n {#1} } { } }
```

\ctex_scheme_input:n

输入 scheme 文件。先查找当前文档类下的 ⟨*scheme*⟩，找不到再查找一般的文件。

```
\cs_new_protected_nopar:Npn \ctex_scheme_input:n #1
  {
    \ctex_file_wrapper:nnn
      { }
      {
        \tl_if_exist:NTF \c__ctex_class_tl
          {
            \file_if_exist_input:nF { ctex-scheme- #1 - \c__ctex_class_tl .def }
              { \file_input:n  { ctex-scheme- #1 .def } }
          }
          { \file_input:n  { ctex-scheme- #1 .def } }
      }
      { }
  }
\cs_generate_variant:Nn \ctex_scheme_input:n { o }
```

\g__ctex_section_depth_int

若大于 3，则 \paragraph 和 \subparagraph 标题单独占一行；若为 3，则 \paragraph 单独占一行。

```
⟨*!beamer⟩
\int_new:N \g__ctex_section_depth_int
\int_gset:Nn \g__ctex_section_depth_int { 2 }
⟨/!beamer⟩

⟨/class|ctex|ctexheading⟩
⟨*class|ctex⟩
```

对旧版本的宏包给出错误信息。

```
\msg_new:nnnn { ctex } { package-too-old }
  { Support~package~`#1'~too~old. }
  {
    Please~update~an~up~to~date~version~of~the~package~`#1'\\
    using~your~TeX~package~manager~or~from~CTAN.
  }
```

\ifctexpdf　在 zhmetrics 映射文件中使用。

```
\sys_if_output_pdf:TF
  { \cs_new_eq:NN \ifctexpdf \if_true: }
  { \cs_new_eq:NN \ifctexpdf \if_false: }
```

\ctex_if_preamble:TF　测试是否在 LaTeX 2ε 的导言区。在宏包内部初始为真，文档最开始位置再设置为假。注意，钩子 \ctex_after_end_preamble:n 在 \AtBeginDocument 之后执行，可以与 \@onlypreamble 的行为一致。

```
\cs_new_eq:NN \ctex_if_preamble:TF \use_i:nn
\ctex_after_end_preamble:n { \cs_set_eq:NN \ctex_if_preamble:TF \use_ii:nn }
```

\ctex_set_default_ccwd:Nn　若参数 #2 带长度单位，则设置它为 tl 变量 #1 的值，否则以 \ccwd 为单位。

```
\cs_new_protected:Npn \ctex_set_default_ccwd:Nn #1#2
  { \tl_set:Nx #1 { \__ctex_default_ccwd_aux:n {#2} } }
\cs_new:Npn \__ctex_default_ccwd_aux:n #1
  {
    \exp_not:n {#1}
    \exp_after:wN \__ctex_default_ccwd_aux:w
      \dim_use:N \tex_dimexpr:D #1 pt \scan_stop: \q_stop
  }
\exp_last_unbraced:NNNNo
  \cs_new:Npn \__ctex_default_ccwd_aux:w #1 { \tl_to_str:n { pt } } #2 \q_stop
    { \tl_if_empty:nT {#2} { \ccwd } }
```

\l__ctex_encoding_tl　(pdf)LaTeX 初始化编码为 GBK，其它则是 UTF8。

```
\tl_new:N \l__ctex_encoding_tl
\tl_set:Nx \l__ctex_encoding_tl
  { \sys_if_engine_pdftex:TF { GBK } { UTF8 } }
```

\g__ctex_zhmCJK_bool　是否使用 zhmCJK 宏包。

```
\bool_new:N \g__ctex_zhmCJK_bool
```

\l__ctex_autoindent_tl　保存 autoindent 选项的值，空值表示不自动调整首行缩进。

```
\tl_new:N \l__ctex_autoindent_tl
```

\ctex_if_autoindent_touched:F　检查 autoindent 选项是否被用户设置。

```
\cs_new_eq:NN \ctex_if_autoindent_touched:F \use:n
```

\ctex_zhmap_case:nnn　参数 #1 是 zhmCJK 的内容，#2 是 zhmetrics。

```
\cs_new_eq:NN \ctex_zhmap_case:nnn \use_ii:nnn
```

\ctex_at_end:n　区分 \AtEndOfClass 和 \AtEndOfPackage，虽然它们的意思都是一样的。

```
⟨class⟩ \cs_new_protected_nopar:Npn \ctex_at_end:n { \AtEndOfClass }
⟨ctex⟩ \cs_new_protected_nopar:Npn \ctex_at_end:n { \AtEndOfPackage }
```

\g__ctex_std_options_clist　保存传递给标准文档类的选项。

```
⟨*class⟩
\clist_new:N \g__ctex_std_options_clist
⟨/class⟩
```

对无效选项给出警告。

```
\msg_new:nnn { ctex } { invalid-option }
  { Option~`\l_keys_key_tl'~is~invalid~in~current~mode. }
\msg_new:nnn { ctex } { invalid-value }
  { Value~`#1'~is~invalid~for~the~key~`\l_keys_key_tl'. }
```

对过时选项或命令给出警告。

```
\msg_new:nnn { ctex } { deprecated-option }
  { Option~ `\l_keys_key_tl'~ is~ deprecated.\\ #1 }
\msg_new:nnn { ctex } { deprecated-command }
  { Command~ #1 is~ deprecated.\\ #2 }
\msg_new:nnn { ctex } { deprecated-environment }
  { Environment~ `#1'~ is~ deprecated.\\ #2 }
```

```
⟨/class|ctex⟩
```

\g__ctex_font_size_int　0 表示修改默认字体大小为五号，1 为小四号，大于 1 则不作修改。初始值 -1 表示 zihao 选项未初始化，会在将来根据文档类决定初值。

```
⟨*class|ctex|ctexsize⟩
\int_new:N \g__ctex_font_size_int
\int_set:Nn \g__ctex_font_size_int { -1 }
⟨/class|ctex|ctexsize⟩
```

14.2　宏包选项

```
⟨*class|style⟩
\keys_define:nn { ctex / option }
  {
⟨/class|style⟩
```

zihao

```
⟨*class|ctex|ctexsize⟩
    zihao .choice: ,
    zihao .value_required:n = true ,
    zihao /     5  .code:n = { \int_gset:Nn \g__ctex_font_size_int { 0 } } ,
    zihao /    -4  .code:n = { \int_gset:Nn \g__ctex_font_size_int { 1 } } ,
    zihao / false  .code:n = { \int_gset:Nn \g__ctex_font_size_int { 2 } } ,
  }
⟨/class|ctex|ctexsize⟩
⟨*class|ctex⟩
    c5size  .code:n =
      {
        \msg_warning:nnn { ctex } { deprecated-option }
          { Option~ `zihao=5'~ is~ set. }
        \keys_set:nn { ctex / option } { zihao = 5 }
      } ,
    cs4size  .code:n =
      {
        \msg_warning:nnn { ctex } { deprecated-option }
          { Option~ `zihao=-4'~ is~ set. }
        \keys_set:nn { ctex / option } { zihao = -4 }
      } ,
    c5size  .value_forbidden:n = true ,
    cs4size .value_forbidden:n = true ,
```

linespread　行距初始值为标志 nan，用于检查用户是否设置了 linespread 选项。

```
    linespread  .fp_set:N = \l__ctex_line_spread_fp ,
```

```
linespread .initial:n = { \c_nan_fp } ,
linespread .value_required:n = true ,
```

autoindent　自动调整段落的首行缩进功能。

```
autoindent .choice: ,
autoindent .default:n = { true } ,
autoindent / true    .code:n =
  {
    \tl_set:Nn \l__ctex_autoindent_tl { 2 \ccwd }
    \cs_set_eq:NN \ctex_if_autoindent_touched:F \use_none:n
  } ,
autoindent / false   .code:n =
  {
    \tl_clear:N \l__ctex_autoindent_tl
    \cs_set_eq:NN \ctex_if_autoindent_touched:F \use_none:n
  } ,
autoindent / unknown .code:n =
  {
    \ctex_set_default_ccwd:Nn \l__ctex_autoindent_tl {#1}
    \cs_set_eq:NN \ctex_if_autoindent_touched:F \use_none:n
  } ,
```

indent　仅为兼容性保留，已过时。

```
indent .code:n =
  {
    \msg_warning:nnn { ctex } { deprecated-option }
      {
        The~ functionality~ has~ been~ removed.\\
        It's~ better~ to~ set~ the~ heading~ styles~ via~ `afterindent'~
        options.~
      }
  } ,
indent .value_forbidden:n = true ,
noindent .code:n =
  {
    \msg_warning:nnn { ctex } { deprecated-option }
      {
        The~ functionality~ has~ been~ removed.\\
        It's~ better~ to~ set~ the~ heading~ styles~ via~ `afterindent'~
        options.
      }
  } ,
noindent .value_forbidden:n = true ,
```

GBK
UTF8

```
GBK  .code:n = { \tl_set:Nn \l__ctex_encoding_tl { GBK } } ,
UTF8 .code:n = { \tl_set:Nn \l__ctex_encoding_tl { UTF8 } } ,
GBK  .value_forbidden:n = true ,
UTF8 .value_forbidden:n = true ,
```

fontset　初始值为空。若用户未指定，则根据操作系统载入对应字体配置，可以区分 Windows、Mac OS X 和其它。

```
fontset    .tl_gset:N = \g__ctex_fontset_tl ,
nofonts    .code:n =
  {
    \msg_warning:nnn { ctex } { deprecated-option }
      {
        Option~ `fontset=none'~ is~ set.~ It~ is~ better~ to~ use~
        fontset~ option.
      }
    \keys_set:nn { ctex / option } { fontset = none }
  } ,
```

```
adobefonts .code:n =
  {
    \msg_warning:nnn { ctex } { deprecated-option }
      {
        Option~ `fontset=adobe'~ is~ set.~ It~ is~ better~ to~ use~
        fontset~ option.
      }
    \keys_set:nn { ctex / option } { fontset = adobe }
  } ,
winfonts   .code:n =
  {
    \msg_warning:nnn { ctex } { deprecated-option }
      {
        Option~ `fontset=windows'~ is~ set.~ It~ is~ better~ to~ use~
        fontset~ option.
      }
    \keys_set:nn { ctex / option } { fontset = windows }
  } ,
nofonts    .value_forbidden:n = true ,
winfonts   .value_forbidden:n = true ,
adobefonts .value_forbidden:n = true ,
```

zhmap

```
zhmap .choice: ,
zhmap .default:n = { true } ,
zhmap / zhmCJK .code:n =
  {
    \bool_gset_true:N \g__ctex_zhmCJK_bool
    \cs_gset_eq:NN \ctex_zhmap_case:nnn \use_i:nnn
  } ,
zhmap / true   .code:n =
  {
    \bool_gset_false:N \g__ctex_zhmCJK_bool
    \cs_gset_eq:NN \ctex_zhmap_case:nnn \use_ii:nnn
  } ,
zhmap / false  .code:n =
  {
    \bool_gset_false:N \g__ctex_zhmCJK_bool
    \cs_gset_eq:NN \ctex_zhmap_case:nnn \use_iii:nnn
  } ,
nozhmap   .code:n =
  {
    \msg_warning:nnn { ctex } { deprecated-option }
      { Option~ `zhmap=false'~ is~ set. }
    \keys_set:nn { ctex / option } { zhmap = false }
  } ,
nozhmap   .value_forbidden:n = true ,
```

punct　设置标点符号输出格式。

```
punct   .tl_set:N = \l__ctex_punct_tl ,
punct  .default:n = { quanjiao } ,
punct  .initial:n = { quanjiao } ,
nopunct   .code:n =
  {
    \msg_warning:nnn { ctex } { deprecated-option }
      { Option~ `punct=plain'~ is~ set.  }
    \keys_set:nn { ctex / option } { punct = plain }
  } ,
nopunct   .value_forbidden:n = true ,
```

space

```
space .choices:nn =
  { true , auto , false }
  {
    \exp_args:Nx \ctex_at_end:n
```

```
          { \keys_set:nn { ctex } { space = \l_keys_choice_tl } }
      } ,
    space  .default:n = { true } ,
    nospace   .code:n =
      {
        \msg_warning:nnn { ctex } { deprecated-option }
          { Option~ `space=false'~ is~ set. }
        \keys_set:nn { ctex / option } { space = false }
      } ,
    nospace   .value_forbidden:n = true ,
```

heading

```
    heading .bool_set:N = \l__ctex_heading_bool ,
```

```
⟨/class|ctex⟩
⟨*class|ctex|ctexheading⟩
```

sub3section
sub4section

```
⟨*!beamer⟩
    sub3section .code:n =
      { \int_gset:Nn \g__ctex_section_depth_int { 3 } } ,
    sub4section .code:n =
      { \int_gset:Nn \g__ctex_section_depth_int { 4 } } ,
    sub3section .value_forbidden:n = true ,
    sub4section .value_forbidden:n = true ,
⟨/!beamer⟩
```

scheme

```
    scheme .tl_set:N  = \l__ctex_scheme_tl ,
⟨*ctexheading⟩
    scheme .default:n = { plain } ,
    scheme .initial:n = { plain }
  }
⟨/ctexheading⟩
⟨*!ctexheading⟩
    scheme .default:n = { chinese } ,
    scheme .initial:n = { chinese } ,
⟨/!ctexheading⟩
```

```
⟨/class|ctex|ctexheading⟩
⟨*class|ctex⟩
```

```
    cap    .code:n    =
      {
        \msg_warning:nnn { ctex } { deprecated-option }
          { Option~ `scheme = chinese' ~ is~ set. }
        \keys_set:nn { ctex / option } { scheme = chinese }
      } ,
    nocap  .code:n    =
      {
        \msg_warning:nnn { ctex } { deprecated-option }
          { Option~ `scheme = plain' ~ is~ set. }
        \keys_set:nn { ctex / option } { scheme = plain }
      } ,
    cap    .value_forbidden:n = true ,
    nocap  .value_forbidden:n = true ,
```

fntef
fancyhdr
hyperref

这些都是过时的宏包兼容选项，原选项功能总是打开的。

```
    fntef    .code:n =
      {
        \msg_warning:nnn { ctex } { deprecated-option }
          { `(xe)CJKfntef'~ package~ is~ always~ loaded. }
      } ,
    fancyhdr .code:n =
      {
        \msg_warning:nnn { ctex } { deprecated-option }
          { `fancyhdr'~ package~ is~ loaded. }
```

```
        \RequirePackage { fancyhdr }
      } ,
    hyperref .code:n =
      {
        \msg_warning:nnn { ctex } { deprecated-option }
          { `hyperref'~ package~ will~ be~ loaded. }
        \ctex_at_end:n
          {
            \cs_if_exist:NF \hypersetup
              { \cs_new_eq:NN \hypersetup \ctex_hypersetup:n }
          }
        \ctex_at_end_preamble:n { \RequirePackage { hyperref } }
      } ,
  }
⟨/class|ctex⟩
⟨*class|ctex|ctexsize⟩
```

10pt
11pt
12pt

使 ctex 和 ctexsize 可以接受文档类的全局选项，不修改默认字体大小。在文档类下还将参数传给标准文档类。

```
\tl_clear_new:N \l__ctex_tmp_tl
\clist_map_inline:nn
  {
    10pt , 11pt , 12pt ,
     8pt ,  9pt , 14pt , 17pt , 20pt , 25pt , 30pt , 36pt , 48pt , 60pt
  }
  {
    \tl_put_right:Nn \l__ctex_tmp_tl
      {
        #1 .code:n =
⟨*!class⟩
          { \int_gset:Nn \g__ctex_font_size_int { 2 } } ,
⟨/!class⟩
⟨*class⟩
          {
            \int_gset:Nn \g__ctex_font_size_int { 2 }
            \clist_gput_right:Nn \g__ctex_std_options_clist {#1}
          } ,
⟨/class⟩
        #1 .value_forbidden:n = true ,
      }
  }
\use:x { \keys_define:nn { ctex / option } { \exp_not:o { \l__ctex_tmp_tl } } }
\tl_clear:N \l__ctex_tmp_tl
```

将未知选项传给标准文档类。

```
⟨*class⟩
\keys_define:nn { ctex / option }
  {
    unknown .code:n =
      { \clist_gput_right:No \g__ctex_std_options_clist { \CurrentOption } }
  }
⟨/class⟩
```

⟨!ctexsize⟩

```
\ctex_file_input:n { ctexopts.cfg }
⟨/class|ctex|ctexsize⟩
⟨*class|style⟩
\ProcessKeysOptions { ctex / option }
⟨/class|style⟩
⟨*class⟩
```

五号字使用标准文档类的 10pt 字体大小设置，小四号字则使用 12pt。

```
\if_case:w \g__ctex_font_size_int
```

```
  \clist_gput_right:Nn \g__ctex_std_options_clist { 10pt }
\or:
  \clist_gput_right:Nn \g__ctex_std_options_clist { 12pt }
\fi:
```

使用 \PassOptionsToClass 是为了预防可能存在的选项冲突。

```
⟨*article⟩
\tl_const:Nn \c__ctex_class_tl { article }
\PassOptionsToClass { \g__ctex_std_options_clist } { article }
\LoadClass { article }
⟨/article⟩
⟨*book⟩
\tl_const:Nn \c__ctex_class_tl { book }
\PassOptionsToClass { \g__ctex_std_options_clist } { book }
\LoadClass { book }
⟨/book⟩
⟨*report⟩
\tl_const:Nn \c__ctex_class_tl { report }
\PassOptionsToClass { \g__ctex_std_options_clist } { report }
\LoadClass { report }
⟨/report⟩
⟨*beamer⟩
\tl_const:Nn \c__ctex_class_tl { beamer }
\PassOptionsToClass { \g__ctex_std_options_clist } { beamer }
\LoadClass { beamer }
⟨/beamer⟩

⟨/class⟩
```

14.3　特定引擎支持与设置

14.3.1　ctex-engine-pdftex.def

\ctex_set_zhmap:n　设置 upTEX 字体映射，同时作用于 \AtBeginDvi 与 \AtBeginShipoutFirst。该宏对 pdfTEX 和 upTEX 均有用。

```
⟨*pdftex|uptex|aptex⟩
\cs_new_protected_nopar:Npn \ctex_set_zhmap:n #1
  {
    \AtBeginDvi {#1}
    \ctex_at_end_package:nn { atbegshi }
      { \AtBeginShipoutFirst {#1} }
  }
\@onlypreamble \ctex_set_zhmap:n
⟨/pdftex|uptex|aptex⟩

⟨*pdftex⟩
```

\c__ctex_cmap_encoding_seq　需要加上 CMap 的 CJK 字体编码。

```
\seq_const_from_clist:Nn \c__ctex_cmap_encoding_seq
  { C19 , C10 , C00 , C09 , C40 , C60 }
```

\ctex_family_cmap:nn　在 \DeclareFontFamily 的 ⟨*loading-settings*⟩ 中给 CJK 字体族加上 CMap。

```
\cs_new_protected_nopar:Npn \ctex_family_cmap:nn #1#2
  {
    \cs_if_free:cF { #1 + #2 }
      {
        \seq_if_in:NnT \c__ctex_cmap_encoding_seq {#1}
          { \tl_gput_right:cn { #1 + #2 } { \ctex_add_cmap:n {#1} } }
      }
  }
\cs_generate_variant:Nn \ctex_family_cmap:nn { x }
\cs_new_eq:NN \CTEX@Family@CMap \ctex_family_cmap:xn
```

\ctex_add_cmap:n　给 #1 编码的 CJK 字体加上 CMap。

```
\cs_new_protected_nopar:Npn \ctex_add_cmap:n #1
  {
    \cs_if_free:NF \CJK@plane
      { \ctex_add_cmap:cn { __ctex_add_cmap_ #1 \CJK@plane : } {#1} }
  }
\cs_new_protected_nopar:Npn \ctex_add_cmap:Nn #1#2
  {
    \cs_if_exist:NF #1 { \__ctex_save_cmap:Nn #1 {#2} }
    #1
  }
\cs_generate_variant:Nn \ctex_add_cmap:Nn { c }
\cs_new_protected_nopar:Npn \__ctex_save_cmap:Nn #1#2
  {
    \tl_set:Nx \l__ctex_tmp_tl { \str_lower_case:n {#2} \CJK@plane }
    \tex_immediate:D \tex_pdfobj:D stream ~ file { \l__ctex_tmp_tl .cmap }
    \cs_new_protected_nopar:Npx #1
      {
        \exp_not:N \tex_pdffontattr:D \exp_not:N \tex_font:D
          { /ToUnicode ~ \int_use:N \tex_pdflastobj:D \c_space_tl 0 ~ R }
      }
  }
```

\DeclareFontFamily　只在 pdfLaTeX 下加 CMap。如 cmap 宏包被引入，则不重复设置。

```
\group_begin:
\char_set_catcode_other:N \#
\sys_if_output_pdf:TF
  {
    \group_end:
    \ctex_appto_cmd:NnnTF \DeclareFontFamily { \ExplSyntaxOff }
      { \CTEX@Family@CMap {#1} {#2} }
      {
        \ctex_at_end_package:nn { cmap }
          { \cs_gset_eq:NN \CTEX@Family@CMap \use_none:nn }
      }
      { \ctex_patch_failure:N \DeclareFontFamily }
  }
  { \group_end: }
```

首先检查选项，决定是否载入 zhmCJK 宏包。

```
\if_bool:N \g__ctex_zhmCJK_bool
  \PassOptionsToPackage { encoding = \l__ctex_encoding_tl } { zhmCJK }
  \RequirePackage { zhmCJK }
```

不载入 zhmCJK 宏包时直接调用 CJK 及相关宏包。

```
\else:
  \str_if_eq:onTF { \l__ctex_encoding_tl } { GBK }
    { \RequirePackage { CJK } }
    { \RequirePackage { CJKutf8 } }
  \RequirePackage { CJKpunct , CJKspace }
```

\ctex_load_zhmap:nnnn　载入 zhmetrics 的字体映射文件，同时设置 \CJKrmdefault 等。

```
  \cs_new_protected_nopar:Npn \ctex_load_zhmap:nnnn #1#2#3#4
    {
      \tl_set:Nn \CJKrmdefault {#1}
      \tl_set:Nn \CJKsfdefault {#2}
      \tl_set:Nn \CJKttdefault {#3}
      \ctex_set_zhmap:n { \ctex_zhmap_input:n {#4} }
    }
  \@onlypreamble \ctex_load_zhmap:nnnn
```

\ctex_zhmap_input:n

载入字体映射文件时，确认 % 和 \ 的 \catcode。

```
\cs_new_protected_nopar:Npn \ctex_zhmap_input:n #1
  {
    \ctex_file_wrapper:nnn
      {
        \char_set_catcode_comment:n { 37 } % %
        \char_set_catcode_escape:n  { 92 } % \
      }
      { \file_input:n {#1} }
      {
        \char_set_catcode:nn { 37 } { \char_value_catcode:n { 37 } }
        \char_set_catcode:nn { 92 } { \char_value_catcode:n { 92 } }
      }
  }
```

```
\tl_if_exist:NF \CJKfamilydefault
  { \tl_const:Nn \CJKfamilydefault { \CJKrmdefault } }
\tl_if_exist:NF \CJKrmdefault { \tl_new:N \CJKrmdefault }
\tl_if_exist:NF \CJKsfdefault { \tl_new:N \CJKsfdefault }
\tl_if_exist:NF \CJKttdefault { \tl_new:N \CJKttdefault }
\ctex_preto_cmd:NnnTF \rmfamily { \ExplSyntaxOff }
  { \CJKfamily { \CJKrmdefault } }
  { }
  { \ctex_patch_failure:N \rmfamily }
\ctex_preto_cmd:NnnTF \sffamily { \ExplSyntaxOff }
  { \CJKfamily { \CJKsfdefault } }
  { }
  { \ctex_patch_failure:N \sffamily }
\ctex_preto_cmd:NnnTF \ttfamily { \ExplSyntaxOff }
  { \CJKfamily { \CJKttdefault } }
  { }
  { \ctex_patch_failure:N \ttfamily }
\ctex_preto_cmd:NnnTF \normalfont { \ExplSyntaxOff }
  { \CJKfamily { \CJKfamilydefault } }
  { \cs_set_eq:NN \reset@font \normalfont }
  { \ctex_patch_failure:N \normalfont }
```

zhmCJK 判断结束。

```
\fi:
```

\ctex_CJK_input:n
\CJK@input

breqn 包可能会在正文中将 ^ 的 \catcode 改为 12 或 13，这将破坏 CJK 对汉字的首字节的定义（\CJK@loadBinding 和 \CJK@loadEncoding）。因此需要确保载入 .enc 和 .bdg 文件时，^ 的 \catcode 为 7。

```
\cs_new_protected_nopar:Npn \ctex_CJK_input:n #1
  {
    \ctex_file_wrapper:nnn
      {
        \char_set_catcode_other:n            { 60 } % <
        \char_set_catcode_math_superscript:n { 94 } % ^
        \int_set:Nn \tex_endlinechar:D { -1 }
      }
      { \file_input:n {#1} }
      {
        \char_set_catcode:nn { 60 } { \char_value_catcode:n { 60 } }
        \char_set_catcode:nn { 94 } { \char_value_catcode:n { 94 } }
        \int_set:Nn \tex_endlinechar:D { \int_use:N \tex_endlinechar:D }
      }
  }
\cs_set_eq:NN \CJK@input \ctex_CJK_input:n
```

\ctex_plane_to_utfxvibe:Nn
\CJK@surr

fancyhdr 宏包的 \nouppercase 会将 \uppercase 定义为 \relax，而 \CJK@surr 需要用它将 \CJK@plane 转化成大写字母，这就造成了冲突[16]。我们在这里给出 \CJK@surr 的一个不依赖

[16] https://github.com/CTeX-org/ctex-kit/issues/146

\uppercase 的实现。

```
\if_cs_exist:N \CJK@surr
  \cs_new_protected_nopar:Npn \ctex_plane_to_utfxvibe:Nn #1#2
    {
      \tl_set:Nx \l__ctex_tmp_tl {#2}
      \int_set:Nn \l__ctex_tmp_int
        { \exp_args:No \int_from_hex:n { \l__ctex_tmp_tl } }
      \int_compare:nNnTF \l__ctex_tmp_int < { 256 }
        { \tl_gset:Nx #1 { \int_to_Hex:n { \l__ctex_tmp_int } } }
        {
          \int_sub:Nn \l__ctex_tmp_int { 256 }
          \tl_gset:Nx #1
            {
              \int_to_Hex:n
                { \int_div_truncate:nn { \l__ctex_tmp_int } { 4 } + "D800 }
              \int_to_Hex:n
                { \int_mod:nn { \l__ctex_tmp_int } { 4 } + "DC }
            }
        }
    }
  \cs_set_eq:NN \CJK@surr \ctex_plane_to_utfxvibe:Nn
\fi:
```

CJKpunct 宏包会在 \AtBeginDocument 的里设置标点格式为 quanjiao。

```
\AtBeginDocument
  {
    \str_if_eq:eeF { \l__ctex_punct_tl } { quanjiao }
      { \punctstyle { \l__ctex_punct_tl } }
  }
```

在导言区末尾更新 \CJKfamilydefault，注意要在 \CJK@envStart 之前使用。

```
\ctex_at_end_preamble:n { \ctex_update_default_family: }
```

启用中文字符功能。GBK 编码时，将汉字的首字节设置为活动字符，并对这些字符初始化；UTF8 编码时，上游宏包已经处理好。\CJK@makeActive 应该先于 ctex-name-gbk.cfg 等文件的载入。注意 \CJK@loadBinding 需要调用补丁后的 \CJK@input。使用 zhmCJK 时，此功能已经被启用。

```
\reverse_if:N \if_bool:N \g__ctex_zhmCJK_bool
  \str_if_eq:onF { \l__ctex_encoding_tl } { UTF8 }
    {
      \CJK@makeActive
      \CJK@loadBinding { standard }
    }
```

在导言区结束时调用 \CJK@envStart 启用完整的中文功能。

\CJK@envStart 的定义是

```
\def\CJK@envStart#1#2#3{
  \CJK@upperReset
  \ifCJK@lowercase@
    \CJK@lowerReset
  \fi%
  \CJK@makeActive%
  \CJK@global\let\CJK@selectFamily \CJK@selFam
  \CJK@global\let\CJK@selectEnc \CJK@selEnc%
  \def\CJK@@@enc{#2}
  \ifx\CJK@@@enc \@empty
    \PackageInfo{CJK}{
      no encoding parameter given,\MessageBreak
      waiting for \protect\CJKenc\space commands}
  \else
    \CJKenc{#2}
  \fi
```

```
\CJKfontenc{#2}{#1}
\CJKfamily{#3}
\def\CJK@series{\f@series}
\def\CJK@shape{\f@shape}%
\csname CJKhook\endcsname}
```

\CJK@upperReset 可能会有一定风险，因此我们直到导言区末尾才使用 \CJK@envStart。这样可以避免将 CJK 环境内置入 document 环境的最里层，最后也就不需要 \clearpage。zhmCJK 已经提供类似功能。注意先使用 \ctex_update_default_family: 更新 \CJKfamilydefault。

```
\exp_args:Nx \ctex_at_end_preamble:n
  {
    \exp_not:N \CJK@envStart
      { } { \l__ctex_encoding_tl } { \exp_not:N \CJKfamilydefault }
    \exp_not:N \CJKtilde
  }
```

zhmCJK 判断结束。

```
\fi:
```

\ctex_auto_ignorespaces:　关闭名字空间，保存 \CJK@@ignorespaces 的定义，方便使用。

```
⟨@@=⟩
\cs_new_eq:NN \ctex_auto_ignorespaces: \CJK@@ignorespaces
```

恢复名字空间，要把它放在一个 macrocode 环境中，l3doc 才能正确工作。

```
⟨@@=ctex⟩
```

\ctex_ignorespaces_case:N
\ctex_set_ignorespaces:　设置忽略空格的的方式。根据 space 选项的值重定义 \CJK@ignorespaces，并保存起来供 \CJKhook 备用。

```
\cs_new_protected_nopar:Npn \ctex_ignorespaces_case:N #1
  {
    \cs_set_protected_nopar:Npn \ctex_set_ignorespaces:
      { \cs_set_eq:NN \CJK@ignorespaces #1 }
    \ctex_set_ignorespaces:
  }
\cs_new_protected_nopar:Npn \ctex_set_ignorespaces:
  { \cs_set_eq:NN \CJK@ignorespaces \ctex_auto_ignorespaces: }
```

\CJKhook　CJK 和 CJK* 环境都会重新定义 \CJK@ignorespaces。我们在 CJK 宏包提供的 \CJKhook 里重新设置它，让这两个环境忽略空格的方式都受 space 选项的控制。这对 zhmCJK 是必要的。

```
\ctex_gadd_hook:Nn \CJKhook { \ctex_set_ignorespaces: }
```

\ctex_punct_set:n　设置 CJK 族对应到实际的字体。#1 是 fontset 的名字。

```
\cs_new_protected_nopar:Npn \ctex_punct_set:n #1
  {
    \clist_map_inline:Nn \c__ctex_punct_family_clist
      {
        \cs_if_free:cF { c__ctex_ #1 ##1 _punct_spaces_tl }
          {
            \cs_set_eq:cc
              { CJKpunct@ ##1 @spaces }
              { c__ctex_ #1 ##1 _punct_spaces_tl }
          }
      }
  }
\clist_const:Nn \c__ctex_punct_family_clist
  {
    zhsong , zhhei , zhfs , zhkai , zhli , zhyou ,
    zhsongb , zhheil , zhheib , zhyoub , zhyahei , zhyaheib
  }
```

\ctex_punct_map_family:nn　CJK 族 #1 使用族 #2 的边界信息。

```
\cs_new_protected_nopar:Npn \ctex_punct_map_family:nn #1#2
  {
    \cs_if_free:cF { CJKpunct@ #2 @spaces }
      { \cs_set_eq:cc { CJKpunct@ #1 @spaces } { CJKpunct@ #2 @spaces } }
  }
```

\ctex_punct_map_bfseries:nn　CJK 族 #1 的 \bfseries 使用族 #2 的边界信息。

```
\cs_new_protected_nopar:Npn \ctex_punct_map_bfseries:nn #1#2
  {
    \clist_map_inline:nn {#1}
      {
        \ctex_punct_map_series:nnn { ##1 } { b } {#2}
        \ctex_punct_map_series:nnn { ##1 } { bx } {#2}
      }
  }
\cs_new_protected_nopar:Npn \ctex_punct_map_series:nnn #1#2#3
  {
    \CJKpunctmapfamily { C19 } {#1} {#2} { m }  {#3}
    \CJKpunctmapfamily { C19 } {#1} {#2} { it } {#3}
    \CJKpunctmapfamily { C19 } {#1} {#2} { sl } {#3}
    \CJKpunctmapfamily { C70 } {#1} {#2} { m }  {#3}
    \CJKpunctmapfamily { C70 } {#1} {#2} { it } {#3}
    \CJKpunctmapfamily { C70 } {#1} {#2} { sl } {#3}
  }
```

\ctex_punct_map_itshape:nn　CJK 族 #1 的 \itshape 使用族 #2 的边界信息。

```
\cs_new_protected_nopar:Npn \ctex_punct_map_itshape:nn #1#2
  {
    \CJKpunctmapfamily { C19 } {#1} { m }  { it } {#2}
    \CJKpunctmapfamily { C19 } {#1} { b }  { it } {#2}
    \CJKpunctmapfamily { C19 } {#1} { bx } { it } {#2}
    \CJKpunctmapfamily { C70 } {#1} { m }  { it } {#2}
    \CJKpunctmapfamily { C70 } {#1} { b }  { it } {#2}
    \CJKpunctmapfamily { C70 } {#1} { bx } { it } {#2}
  }
```

\ctex_punct_space:nn
\ctexspadef　定义标点的边界信息。

```
\cs_new_protected_nopar:Npn \ctex_punct_space:nn #1#2
  { \tl_const:cn { c__ctex_ #1 _punct_spaces_tl } {#2} }
\cs_new_eq:NN \ctexspadef \ctex_punct_space:nn
```

载入边界信息文件。

```
\ctex_file_input:n { ctexspa.def }
```

⟨/pdftex⟩

14.3.2 ctex-engine-xetex.def

⟨*xetex⟩

```
\RequirePackage { xeCJK }
\exp_args:Nx \xeCJKsetup
  {
    LoadFandol   = false ,
    PunctStyle   = \l__ctex_punct_tl
  }
```

最新版本的 fontspec 默认对 \rmfamily 和 \sffamily 设置 Ligatures=TeX，对 \ttfamily 设置 WordSpace={1,0,0} 和 PunctuationSpace=WordSpace。

```
\@ifpackagelater { fontspec } { 2014/05/25 } { }
```

```
  { \msg_error:nnn { ctex } { package-too-old } { fontspec } }
⟨/xetex⟩
```

14.3.3 ctex-engine-luatex.def

```
⟨*luatex⟩
```

LuaTeX-ja 为了兼容 pLaTeX 的使用习惯，对 LaTeX 2ε 的 NFSS 作了不少修改和扩充，这对于简体中文用户来说不是必要的。我们在这里禁用它。

```
\msg_new:nnn { ctex } { luatexja-loaded }
  {
    Package~`luatexja'~can~not~be~loaded~before~`ctex'.\\
    Loading~file~`#1'~will~abort!
  }
\@ifpackageloaded { luatexja }
  { \msg_critical:nnx { ctex } { luatexja-loaded } { \g_file_curr_name_str } }
  { \tl_const:cn { ver@ltj-latex.\@pkgextension } { 9999/99/99 } }
\RequirePackage { luatexja }
\@ifpackagelater { luatexja } { 2015/09/21 } { }
  { \msg_error:nnn { ctex } { package-too-old } { luatexja } }
\RequirePackage { fontspec }
\@ifpackagelater { fontspec } { 2014/05/25 } { }
  { \msg_error:nnn { ctex } { package-too-old } { fontspec } }
```

14.3.3.1 LuaTeX-ja 的默认设置

```
\ExplSyntaxOff
```

以下设置抄录自 `lltjdefs.sty`，略有改动。

U+2460–U+24FF (Enclosed Alphanumerics) 原属于字符范围 6，是 JAchar，我们把它们归入字符范围 3，改成 ALchar。

```
\ltjdefcharrange{1}{"80-"36F, "1E00-"1EFF}
\ltjdefcharrange{2}{"370-"4FF, "1F00-"1FFF}
\ltjdefcharrange{3}{%
  "2000-"206F, "2070-"243F, "2460-"24FF, "2500-"27BF, "2900-"29FF,
  "2B00-"2BFF}
\ltjdefcharrange{4}{%
   "500-"10FF, "1200-"1DFF, "2440-"245F, "27C0-"28FF, "2A00-"2AFF,
  "2C00-"2E7F, "4DC0-"4DFF, "A4D0-"A82F, "A840-"ABFF, "FB00-"FE0F,
  "FE20-"FE2F, "FE70-"FEFF, "10000-"1FFFF, "E000-"F8FF} % non-Japanese
\ltjdefcharrange{5}{"D800-"DFFF, "E0000-"E00FF, "E01F0-"10FFFF}
\ltjdefcharrange{6}{%
  "2E80-"2EFF, "3000-"30FF, "3190-"319F, "31F0-"4DBF,
  "4E00-"9FFF, "F900-"FAFF, "FE10-"FE6F, "20000-"2FFFF, "E0100-"E01EF}
\ltjdefcharrange{7}{%
  "1100-"11FF, "2F00-"2FFF, "3100-"31EF, "A000-"A4CF,
  "A830-"A83F, "AC00-"D7FF}
\ltjdefcharrange{8}{"A7, "A8, "B0, "B1, "B4, "B6, "D7, "F7}
```

将引号、破折号等中西文公用的标点符号归入字符范围 9，将他们设置为 JAchar。

```
\ltjdefcharrange{9}{%
  "2018, "2019, "201C, "201D, "2013, "2014, "2025, "2026, "2027, "2E3A}
```

LuaTeX-ja 默认把字符范围 2 和 3 设置为 JAchar，我们这里把它们都改成 ALchar。

```
\ltjsetparameter{jacharrange={-1, -2, -3, -4, -5, +6, +7, -8, +9}}
\directlua{for x=128,255 do luatexja.math.is_math_letters[x] = true end}
```

以下设置抄录自 `ltj-latex.sty`。

```
\directlua{
  local s = kpse.find_file('ltj-kinsoku.lua', 'tex')
  luatexja.stack.charprop_stack_table[0] = s and dofile(s) or {}
}
\ltjsetparameter{kanjiskip=\z@ plus .4pt minus .5pt,
  xkanjiskip=.25\zw plus 1pt minus 1pt,
```

```
  autospacing, autoxspacing, jacharrange={-1},
  yalbaselineshift=\z@, yjabaselineshift=\z@,
  jcharwidowpenalty=500, differentjfm=paverage
}
\ExplSyntaxOn
```

14.3.3.2 LuaTeX-ja 的补丁

```
⟨@@=ctex_ltj⟩
```

在 LaTeX 下，LuaTeX-ja 对 fontspec、xunicode、unicode-math 和 listings 打了补丁。其中前三个是把 `\char` 换成 `\ltjalchar`，确保字符是 ALchar 类。我们这里用 xunicode-addon 来处理 xunicode。

```
\RequirePackage { xunicode-addon }
\AtBeginUTFCommand
  {
    \group_begin:
    \lua_now:e { tex.globaldefs = 0 }
    \ltj@allalchar
  }
\AtEndUTFCommand { \group_end: }
```

`\fontspec_visible_space:`

我们不使用 luatexja 对 fontspec 的补丁，直接处理。

```
\cs_set_protected:Npn \fontspec_visible_space:
  {
    \tex_iffontchar:D \tex_font:D "2423 \exp_stop_f:
      \ltjalchar "2423 \exp_stop_f:
    \else:
      \fontspec_visible_space_fallback:
    \fi:
  }
```

对 listings 的补丁是让代码环境支持 JAchar 类。LuaTeX-ja 的补丁会将代码目录标题改为日文，我们不需要。

```
\ctex_at_end_package:nn { listings }
  {
    \use:x
      {
        \exp_not:N \RequirePackage { lltjp-listings }
        \tl_set:Nn \exp_not:N \lstlistingname
          { \exp_not:o { \lstlistingname } }
        \tl_set:Nn \exp_not:N \lstlistlistingname
          { \exp_not:o { \lstlistlistingname } }
      }
  }
```

14.3.3.3 字体切换方式

`\ctex_ltj_select_font:`
`\CJK@family`

`\CJK@family` 保存的是当前 CJK 实际的字体族名，如果为空表示没有设置过字体。

```
\cs_new_protected_nopar:Npn \ctex_ltj_select_font:
  {
    \cs_if_exist_use:cF { \l__ctex_ltj_current_font_tl }
      { \tl_if_empty:NF \CJK@family { \__ctex_ltj_select_font_aux: } }
  }
\tl_new:N \CJK@family
\tl_new:N \l__ctex_ltj_current_font_tl
\tl_set:Nn \l__ctex_ltj_current_font_tl
  { \CJK@encoding / \CJK@family / \f@series / \f@shape / \f@size }
```

`\__ctex_ltj_select_font_aux:`

使用 `\pickup@font` 取得字体名称前，总需要先设置 `\font@name`。在这里将 `\f@family` 换成

CJK 字体族，并确保编码正确。

```
\cs_new_protected_nopar:Npn \__ctex_ltj_select_font_aux:
  {
    \group_begin:
      \tl_set_eq:NN \f@encoding \CJK@encoding
      \tl_set_eq:NN \f@family \CJK@family
      \__ctex_ltj_push_fontname:n { \use:c { \curr@fontshape / \f@size } }
      \ctex_ltj_pickup_font:
    \group_end:
    \font@name
    \__ctex_ltj_pop_fontname:
```

当字形未定义的时候，NFSS 就会启动替换机制（`\wrong@fontshape`）。第一次启动后，`\l__ctex_ltj_current_font_tl` 还是没有定义。为此，我们再次选择字体，确保它有定义和指向正确的 `font.id`。这对 `AlternateFont` 的设置特别重要。

```
    \cs_if_exist:cF { \l__ctex_ltj_current_font_tl }
      { \__ctex_ltj_select_font_aux: }
  }
\cs_new_protected_nopar:Npn \__ctex_ltj_push_fontname:n #1
  {
    \cs_gset_eq:NN \__ctex_ltj_save_fontname:w \font@name
    \cs_gset_nopar:Npx \font@name {#1}
  }
\cs_new_protected_nopar:Npn \__ctex_ltj_pop_fontname:
  { \cs_gset_eq:NN \font@name \__ctex_ltj_save_fontname:w }
```

`\ctex_ltj_pickup_font:`

替换 `\define@newfont` 内部调用的 `\extract@font` 和 `\do@subst@correction`。

```
\cs_new_protected_nopar:Npn \ctex_ltj_pickup_font:
  {
    \exp_after:wN \cs_if_exist:NF \font@name
      {
        \group_begin:
          \cs_set_eq:NN \extract@font \ctex_ltj_extract_font:
          \cs_set_eq:NN \do@subst@correction \ctex_ltj_subst_font:
          \define@newfont
        \group_end:
      }
  }
\cs_new_eq:NN \pickup@jfont \ctex_ltj_pickup_font:
```

`\ctex_ltj_extract_font:`

LuaTeX-ja 的 `\globaljfont` 在 luatexja-core 中定义：

```
%%%%%%%% \jfont\CS={...:...;jfm=metric;...}, \globaljfont
\protected\def\jfont#1{%
  \afterassignment\ltj@@jfont
  \directlua{luatexja.jfont.jfontdefX(false, 'yoko','\luatexluaescapestring{\noexpand#1}')}}
\protected\def\globaljfont#1{%
  \afterassignment\ltj@@jfont
  \directlua{luatexja.jfont.jfontdefX(true,  'yoko','\luatexluaescapestring{\noexpand#1}')}}
\def\ltj@@jfont{\directlua{luatexja.jfont.jfontdefY()}}
```

`jfontdefX` 函数的作用是把 `\CS` 定义为其后的字体，`jfontdefY` 的作用是更新 JFM 和记录相关字体信息。最后的工作是：

```
tex.sprint(cat_lp, global_flag, '\\protected\\expandafter\\def\\csname ',
  (cstemp==' ') and '\\space' or cstemp, '\\endcsname{\\ltj@cur'..
  (jfm_dir == 'yoko' and 'j' or 't') .. 'fnt', fn, '\\relax}')
```

`\CS` 的作用就是把 `\ltj@curjfnt` 设置为刚才定义的字体的 `font.id`。

```
\cs_new_protected_nopar:Npn \ctex_ltj_extract_font:
  {
    \get@external@font
    \ctex_ltj_if_alternate_shape_exist:nT { \curr@fontshape }
```

```
      {
        \tl_set:Nx \external@font
          { \exp_after:wN \__ctex_ltj_patch_external_font:w \external@font }
      }
    \exp_after:wN \globaljfont \font@name \external@font \scan_stop:
```

这里 \font@name 不会直接改变当前字体，而 \DeclareFontFamily 和 \DeclareFontShape 的最后一个参数通常要使用 \font 来引用当前字体。为此，我们在分组内启用之前定义的字体，以便能得到正确的 \font。对字体参数的赋值总是全局的，不会受到分组的影响。

```
    \font@name
    \lua_now:e { font.current(tex.getattribute('ltj@curjfnt')) }
    \use:c { \f@encoding + \f@family }
    \use:c { \curr@fontshape }
  }
```

\ctex_ltj_subst_font:

\do@subst@correction 在设置通过 sub 或者 ssub 函数定义的字体时会用到。如果没有设置 SlantedFont，fontspec 会设置 \itdefault 作为 \sldefault 的替代字形，因而会用到这个函数。它的本来定义是：

```
\def\do@subst@correction{%
    \xdef\subst@correction{%
       \font@name
       \global\expandafter\font
         \csname \curr@fontshape/\f@size\endcsname
         \noexpand\fontname\font
        \relax}%
    \aftergroup\subst@correction
}
```

我们在这里不需要定义新字体，而是设置对应字体的命令。

```
\cs_new_protected_nopar:Npn \ctex_ltj_subst_font:
  {
    \ctex_ltj_if_alternate_shape_exist:nF { \curr@fontshape }
      {
        \group_begin:
        \tl_set_eq:NN \CJK@family \f@family
        \cs_if_exist:cF { \l__ctex_ltj_current_font_tl  }
          {
            \cs_gset_protected_nopar:Npx \subst@correction
              {
                \cs_new_eq:NN
                  \exp_not:c { \l__ctex_ltj_current_font_tl }
                  \font@name
              }
            \group_insert_after:N \group_insert_after:N
            \group_insert_after:N \subst@correction
          }
        \group_end:
      }
  }
```

\ctex_ltj_if_alternate_shape_exist:n*TF*

即 LuaTeX-ja 中的 \ltj@@does@alt@set，判断是否存在替代字体。

```
\prg_new_conditional:Npnn \ctex_ltj_if_alternate_shape_exist:n #1 { T , F , TF }
  {
    \lua_now:e { luatexja.jfont.does_alt_set ('\lua_escape:e {#1}') }
      \prg_return_true: \else: \prg_return_false: \fi:
  }
```

__ctex_ltj_patch_external_font:w

若对字体的定义完全相同，则它们有相同的 font.id。因此如果字形是由 NFSS 的替换机制定义的，它们就有相同的 font.id。print_aftl_address 函数的定义是

```
function print_aftl_address()
```

```
  tex.sprint(cat_lp, ';ltjaltfont' .. tostring(aftl_base):sub(8))
end
```

主要目的是，如果当前字形有替代字体，则往字形的定义中加入一些标志，确保 font.id 唯一。

```
\cs_new_nopar:Npn \__ctex_ltj_patch_external_font:w #1 ~ at
  { #1 \lua_now:e { luatexja.jfont.print_aftl_address() } ~ at }
```

\ctex_ltj_select_alternate_font:

在 \selectfont 中更新替代字体。

```
\cs_new_protected_nopar:Npn \ctex_ltj_select_alternate_font:
  {
    \ctex_ltj_if_alternate_shape_exist:nT { \l__ctex_ltj_current_shape_tl }
      {
        \lua_now:e
          {
            luatexja.jfont.output_alt_font_cmd
              ('y', '\lua_escape:e { \l__ctex_ltj_current_shape_tl }')
          }
        \lua_now:e { luatexja.jfont.pickup_alt_font_a ('\f@size') }
      }
  }
\tl_new:N \l__ctex_ltj_current_shape_tl
\tl_set:Nn \l__ctex_ltj_current_shape_tl
  { \CJK@encoding / \CJK@family / \f@series / \f@shape }
```

\ltj@pickup@altfont@auxy

被用在函数 output_alt_font_cmd 中，作用是定义替代字体。

```
\cs_new_protected_nopar:Npn \ltj@pickup@altfont@auxy #1
  {
    \cs_if_exist:cF { #1/\f@size }
      {
        \group_begin:
          \use:x { \exp_not:N \split@name #1 / \f@size } \@nil
          \__ctex_ltj_push_fontname:n { \use:c { \curr@fontshape / \f@size } }
          \ctex_ltj_pickup_font:
        \group_end:
        \__ctex_ltj_pop_fontname:
      }
  }
```

⟨@@=⟩

\ltj@pickup@altfont@copy

被用在函数 pickup_alt_font_a 中。\ltj@@getjfontnumber 的作用是将字体命令 #1 对应的 font.id 保存到 \ltj@tempcntc 中。

```
\cs_new_protected_nopar:Npn \ltj@pickup@altfont@copy #1#2
  {
    \ltj@@getjfontnumber #1
    \lua_now:e
      {
        luatexja.jfont.pickup_alt_font_b
          ( \int_use:N \ltj@tempcntc, '\lua_escape:e {#2}' )
      }
  }
```

14.3.3.4　数学字体族

⟨@@=ctex_ltj⟩

以下内容来自 lltjfont.sty，目的是让汉字可以在数学环境中直接使用。

\ctex_ltj_if_jfont:nTF

参数 #1 是一个 LaTeX 2ε 编码名称或者字体命令。LaTeX 2ε 字体命令的一般形式是：

\⟨*encoding*⟩/⟨*family*⟩/⟨*series*⟩/⟨*shape*⟩

通过截取名字中的 ⟨*encoding*⟩ 来判断是否是 jfont。最后会设置 \ifin@ 为对应的 \iftrue 或者 \iffalse。

```
\cs_new:Npn \ctex_ltj_if_jfont:nTF #1
  {
    \lua_now:e
      { luatexja.jfont.is_kenc( string.match('\lua_escape:e {#1}', '[^/]+') ) }
    \ifin@ \exp_after:wN \use_i:nn \else: \exp_after:wN \use_ii:nn \fi:
  }
```

\ctex_ltj_if_jfont_math:NTF

#1 是一个形式为 \M@⟨*encoding*⟩ 的命令，它由 \DeclareFontEncoding 的第三个参数来定义。

```
\cs_new:Npn \ctex_ltj_if_jfont_math:NTF #1
  { \exp_after:wN \__ctex_ltj_if_jfont_math:w \token_to_str:N #1 \q_stop }
\group_begin:
  \char_set_catcode_other:N M
  \cs_new:Npn \__ctex_ltj_if_jfont_math:w #1 M #2#3 \q_stop
    { \ctex_ltj_if_jfont:nTF {#3} }
\group_end:
```

```
⟨@@=⟩
```

\getanddefine@fonts
\ctex_ltj_get_and_define_fonts:nN

在使用的场合，\escapechar 已经被设置成 −1，使用 \token_to_str:N 就可以得到名字，不必使用 \cs_to_str:N。

```
\cs_new_protected_nopar:Npn \ctex_ltj_get_and_define_fonts:nN #1#2
  {
    \ctex_ltj_if_jfont:nTF { \token_to_str:N #2 }
      { \ctex_ltj_get_and_define_fonts_ja:nN }
      { \ctex_ltj_get_and_define_fonts_al:nN }
      {#1} #2
  }
\cs_new_eq:NN \ctex_ltj_get_and_define_fonts_al:nN \getanddefine@fonts
\cs_set_eq:NN \getanddefine@fonts \ctex_ltj_get_and_define_fonts:nN
\cs_new_protected_nopar:Npn \ctex_ltj_get_and_define_fonts_ja:nN #1#2
  {
    \tl_gset:Nx \font@name { \use:c { \token_to_str:N #2 / \tf@size } }
    \ctex_ltj_pickup_font: \tl_set_eq:NN \textfont@name \font@name
    \tl_gset:Nx \font@name { \use:c { \token_to_str:N #2 / \sf@size } }
    \ctex_ltj_pickup_font: \tl_set_eq:NN \scriptfont@name \font@name
    \tl_gset:Nx \font@name { \use:c { \token_to_str:N #2 / \ssf@size } }
    \ctex_ltj_pickup_font:
    \tl_put_right:Nx \math@fonts
      {
        \ltj@setpar@global
        \ltj@@set@stackfont #1 , \textfont@name   \c_colon_str { MJT }
        \ltj@@set@stackfont #1 , \scriptfont@name \c_colon_str { MJS }
        \ltj@@set@stackfont #1 , \font@name       \c_colon_str { MJSS }
      }
  }
```

```
⟨@@=ctex_ltj⟩
```

\use@mathgroup
\ctex_ltj_use_math_group:Nn

在使用 unicode-math 宏包时，\ctex_ltj_math_group_hook: 将被重定义。

```
\cs_new_protected_nopar:Npn \ctex_ltj_use_math_group:Nn #1#2
  {
    \mode_if_math:T
      {
        \math@bgroup
          \cs_if_eq:cNF { M@ \f@encoding } #1 {#1}
          \ctex_ltj_math_group_hook:
          \ctex_ltj_if_jfont_math:NTF #1
            { \jfam } { \mathgroup } #2 \scan_stop:
        \math@egroup
      }
  }
```

```
\cs_new_eq:NN \ctex_ltj_math_group_hook: \prg_do_nothing:
\cs_set_eq:NN \use@mathgroup \ctex_ltj_use_math_group:Nn
```

对 unicode-math 的补丁主要是将 `unicode-math-table.tex` 中的数学符号设置为 luatexja 中的数学字母。本段代码应放在 `\ctex_ltj_math_group_hook:` 的定义之后，避免因宏包载入顺序而造成的编译错误。

```
\cs_new_protected:Npn \ctex_ltj_set_math_letter:NN #1#2
  {
    \group_begin:
      \cs_set_protected:Npn #1 ##1##2##3
        { \ltjsetmathletter { ##1 } }
      #2
    \group_end:
  }
\ctex_at_end_package:nn { unicode-math }
  {
    \cs_if_exist:NTF \um_input_math_symbol_table:
      {
        \ctex_ltj_set_math_letter:NN
          \um_sym:nnn
          \um_input_math_symbol_table:
      }
      {
        \cs_set_eq:NN \use@mathgroup \ctex_ltj_use_math_group:Nn
        \cs_set_protected_nopar:Npn \ctex_ltj_math_group_hook:
          { \__um_switchto_literal: }
        \ctex_ltj_set_math_letter:NN
          \__um_sym:nnn
          \__um_input_math_symbol_table:
      }
  }
```

14.3.3.5　字体族的定义与使用

`\ctex_mono_jfm:n`
`\l__ctex_ltj_jfm_tl`

LuaTeX-ja 中与标点格式 `plain` 对应的 JFM 是 `mono`。

```
\cs_new_protected_nopar:Npn \ctex_mono_jfm:n #1
  {
    \str_if_eq:nnTF {#1} { plain }
      { \tl_set:Nn \l__ctex_ltj_jfm_tl { mono } }
      { \tl_set:Nn \l__ctex_ltj_jfm_tl {#1} }
  }
\tl_new:N \l__ctex_ltj_jfm_tl
\cs_generate_variant:Nn \ctex_mono_jfm:n { o }
\ctex_mono_jfm:o { \l__ctex_punct_tl }
```

`\CJK@encoding`
`\__ctex_ltj_change_encoding:`

在 LaTeX 下， LuaTeX-ja 依赖字体编码来实现特殊设置。例如上述的 `\ctex_ltj_if_jfont:nTF` 就是通过判断编码来实现的， 它在设置数学字体时会用到。所以不应该与西文共用 EU2。定义字体族 song 为 `\CJK@encoding` 的默认替换字体。下划线 _ 不在 `\nfss@catcodes` 里，可以放心使用。

```
\tl_const:Nn \CJK@encoding { LTJY3 }
\DeclareFontEncoding { \CJK@encoding } { } { }
\use:x
  {
    \exp_not:N \DeclareFontSubstitution
      { \CJK@encoding } { song } { \mddefault } { \updefault }
  }
\lua_now:e { luatexja.jfont.add_kyenc_list('\CJK@encoding') }
\cs_new_protected_nopar:Npn \__ctex_ltj_change_encoding:
  { \tl_set_eq:NN \g_fontspec_encoding_tl \CJK@encoding }
\DeclareFontFamily { \CJK@encoding } { song } { }
\DeclareFontShape { \CJK@encoding } { song } { \mddefault } { \updefault }
```

```
  { <-> psft:SimSun:cid=Adobe-GB1-5;jfm=\l__ctex_ltj_jfm_tl } { }
\DeclareFontShape { \CJK@encoding } { song } { \bfdefault } { \updefault }
  { <-> psft:SimHei:cid=Adobe-GB1-5;jfm=\l__ctex_ltj_jfm_tl } { }
\tl_const:Nn \c__ctex_ltj_math_tl { CJKmath }
\DeclareSymbolFont { \c__ctex_ltj_math_tl }
  { \CJK@encoding } { song } { \mddefault } { \updefault }
\SetSymbolFont { \c__ctex_ltj_math_tl } { bold }
  { \CJK@encoding } { song } { \bfdefault } { \updefault }
\int_const:Nn \c__ctex_ltj_math_fam_int { \use:c { sym \c__ctex_ltj_math_tl } }
\jfam \c__ctex_ltj_math_fam_int
```

这是 luatexja-fontspec 中新增的一些字体选项。

```
\newfontfeature { CID }     {    cid = #1 }
\newfontfeature { JFM }     {    jfm = #1 }
\newfontfeature { JFM-var } { jfmvar = #1 }
```

在新版本的 fontspec 中，`\__fontspec_fontname_wrap:n` 变成了私有函数。

```
\keys_define:nn { fontspec-preparse-external }
  {
    NoEmbed .code:n =
      { \cs_set_eq:NN \__fontspec_fontname_wrap:n \__ctex_ltj_noembed_wrap:n }
  }
\cs_new:Npn \__ctex_ltj_noembed_wrap:n #1 { psft: #1 }
```

`\ctex_ltj_set_family:nnn` 将自定义的字体族名与 fontspec 实际设置的名字对应起来。

```
\cs_new_protected:Npn \ctex_ltj_set_family:nnn #1#2#3
  {
    \group_begin:
    \clist_clear:N \l__ctex_ltj_char_range_clist
    \seq_clear:N \l__ctex_ltj_alternate_seq
    \tl_set:Nn \l__ctex_ltj_base_CJKfamily_tl {#1}
    \keys_set_known:nnN { ctex_ltj / fontspec } {#2} \l__ctex_ltj_tmp_tl
    \clist_set:No \l__ctex_ltj_font_options_clist { \l__ctex_ltj_tmp_tl }
    \ctex_ltj_set_alternate_family:nnF {#1} {#3}
      {
        \prop_gput:Nnn \g__ctex_ltj_family_font_name_prop {#1} {#3}
        \prop_gput:Nno \g__ctex_ltj_family_font_options_prop
          {#1} { \l__ctex_ltj_font_options_clist }
        \__ctex_ltj_update_family_uid:N \l__ctex_ltj_font_options_clist
        \__ctex_ltj_use_global_options:N \l__ctex_ltj_font_options_clist
        \__ctex_ltj_gset_family_cs:nn {#1} {#3}
      }
    \group_end:
  }
\tl_new:N \l__ctex_ltj_base_CJKfamily_tl
\clist_new:N \l__ctex_ltj_font_options_clist
\cs_new_protected_nopar:Npn \__ctex_ltj_use_global_options:N #1
  {
    \clist_concat:NNN #1 \g__ctex_ltj_default_features_clist #1
    \clist_put_left:Nx #1 { JFM = \l__ctex_ltj_jfm_tl }
  }
```

`\g__ctex_ltj_family_name_prop`
`\g__ctex_ltj_family_font_name_prop`
`\g__ctex_ltj_family_font_options_prop`

分别保存 fontspec 设置的字体族名、字体名称和字体选项。

```
\prop_new:N \g__ctex_ltj_family_name_prop
\prop_new:N \g__ctex_ltj_family_font_name_prop
\prop_new:N \g__ctex_ltj_family_font_options_prop
```

`\__ctex_ltj_check_family:n` 删除重复的定义，清除替代字体的先前设置。

```
\cs_new_protected_nopar:Npn \__ctex_ltj_check_family:n #1
  {
    \prop_gpop:NnNT \g__ctex_ltj_family_font_name_prop {#1} \l__ctex_ltj_tmp_tl
      {
```

```
        \cs_undefine:c { \__ctex_ltj_family_csname:n {#1} }
        \cs_undefine:c { \__ctex_ltj_alternate_cs:n {#1} }
       \prop_gpop:NnNT \g__ctex_ltj_family_name_prop {#1} \l__ctex_ltj_base_family_tl
         {
           \use:c { \__ctex_ltj_alternate_cs:n { clear / #1 } }
           \cs_undefine:c { \__ctex_ltj_alternate_cs:n { clear / #1 } }
           \cs_undefine:c { \__ctex_ltj_alternate_cs:n { reset / #1 } }
           \prop_gremove:Nn \g__ctex_ltj_reset_alternate_prop {#1}
         }
        \msg_warning:nnxx { ctex } { redefine-family } {#1} { \l__ctex_ltj_tmp_tl }
      }
  }
\tl_new:N \l__ctex_ltj_tmp_tl
\msg_new:nnn { ctex } { redefine-family }
  { Redefining~CJKfamily~`\__ctex_ltj_msg_family_map:n {#1}'~(#2). }
```

__ctex_ltj_gset_family_cs:nn　在设置字体时，实际上并不是马上就定义。而是只保存相关参数，在通过 \CJKfamily 第一次使用时才定义。需要注意将编码改为 \CJK@encoding。

```
\cs_new_protected_nopar:Npn \__ctex_ltj_gset_family_cs:nn #1#2
  {
    \cs_gset_protected_nopar:cpx { \__ctex_ltj_family_csname:n {#1} }
      {
        \group_begin:
        \__ctex_ltj_change_encoding:
        \exp_not:n { \cs_set_eq:NN \CJKfamily \use_none:n }
        \exp_not:n { \cs_set_eq:NN \CTEX@fontfamily \use_none:n }
        \exp_not:n { \fontspec_set_family:Nnn \g__ctex_ltj_fontspec_family_tl }
          { \exp_not:o { \l__ctex_ltj_font_options_clist } } {#2}
        \prop_gput:Nno \exp_not:N \g__ctex_ltj_family_name_prop {#1}
          { \exp_not:N \g__ctex_ltj_fontspec_family_tl }
        \tl_gset_eq:NN \exp_not:N \g__ctex_ltj_fontspec_family_tl
          \exp_not:N \g__ctex_ltj_fontspec_family_tl
        \__ctex_ltj_set_alternate_family:n {#1}
        \group_end:
      }
  }
\tl_new:N \l__ctex_ltj_base_family_tl
\tl_new:N \g__ctex_ltj_fontspec_family_tl
\cs_new_nopar:Npn \__ctex_ltj_family_csname:n #1 { ctex_ltj/family/#1 }
\cs_new_protected_nopar:Npn \__ctex_ltj_set_alternate_family:n #1
  {
    \tl_set:Nn \l__ctex_ltj_base_CJKfamily_tl {#1}
    \tl_set_eq:NN \l__ctex_ltj_base_family_tl \g__ctex_ltj_fontspec_family_tl
    \cs_if_exist_use:c { \__ctex_ltj_alternate_cs:n { reset / #1 } }
    \cs_if_exist_use:c { \__ctex_ltj_alternate_cs:n {#1} }
  }
\cs_new:Npn \__ctex_ltj_alternate_cs:n #1 { ctex_ltj/alternate_family/#1 }
```

\CJKfamily　切换字体。

```
\NewDocumentCommand \CJKfamily { m }
  { \ctex_ltj_switch_family:x {#1} \tex_ignorespaces:D }
\cs_new_protected_nopar:Npn \ctex_ltj_switch_family:n #1
  {
    \ctex_ltj_family_if_exist:nNTF {#1} \CJK@family
      {
        \tl_set:Nn \l_ctex_ltj_family_tl {#1}
        \selectfont
      }
      { \__ctex_ltj_family_unknown_warning:n {#1} }
  }
\tl_new:N \l_ctex_ltj_family_tl
\cs_generate_variant:Nn \ctex_ltj_switch_family:n { x }
```

\ctex_ltj_family_if_exist:nNTF 判断 CJK 字体族 #1 是否存在，若存在则把实际族名保存到 #2 中。

```
\prg_new_protected_conditional:Npnn \ctex_ltj_family_if_exist:nN #1#2 { T , F , TF }
  {
    \prop_get:NnNTF \g__ctex_ltj_family_name_prop {#1} #2
      { \prg_return_true: }
      {
        \cs_if_exist_use:cTF { \__ctex_ltj_family_csname:n {#1} }
          {
            \tl_set_eq:NN #2 \g__ctex_ltj_fontspec_family_tl
            \prg_return_true:
          }
          { \prg_return_false: }
      }
  }
\prg_generate_conditional_variant:Nnn \ctex_ltj_family_if_exist:nN { x } { T , F , TF }
```

__ctex_ltj_family_unknown_warning:n

```
\cs_new_protected_nopar:Npn \__ctex_ltj_family_unknown_warning:n #1
  {
    \prop_if_empty:NF \g__ctex_ltj_family_font_name_prop
      {
        \seq_if_in:NnF \g__ctex_ltj_unknown_family_seq {#1}
          {
            \seq_gput_right:Nn \g__ctex_ltj_unknown_family_seq {#1}
            \msg_warning:nnn { ctex } { family-unknown } {#1}
          }
      }
  }
\seq_new:N \g__ctex_ltj_unknown_family_seq
\msg_new:nnn { ctex } { family-unknown }
  {
    Unknown~CJK~family~`\__ctex_ltj_msg_family_map:n {#1}'~is~being~ignored.\\
    Try~to~use~`\__ctex_ltj_msg_def_family_map:n {#1}'~to~define~it.
  }
\cs_new_nopar:Npn \__ctex_ltj_msg_def_family_map:n #1
  {
    \str_case_e:nnF {#1}
      {
        \CJKrmdefault { \token_to_str:N \setCJKmainfont }
        \CJKsfdefault { \token_to_str:N \setCJKsansfont }
        \CJKttdefault { \token_to_str:N \setCJKmonofont }
      }
      { \token_to_str:N \setCJKfamilyfont \{ #1 \} }
    [...]\{...\}
  }
\cs_new_nopar:Npn \__ctex_ltj_msg_family_map:n #1
  {
    \str_case_e:nnF {#1}
      {
        \CJKrmdefault { \token_to_str:N \CJKrmdefault }
        \CJKsfdefault { \token_to_str:N \CJKsfdefault }
        \CJKttdefault { \token_to_str:N \CJKttdefault }
      }
      {#1}
  }
```

\ctex_ltj_fontspec:nn

```
\cs_new_protected_nopar:Npn \ctex_ltj_fontspec:nn #1#2
  {
    \prop_get:NnNTF \g__ctex_ltj_fontspec_prop
      { CJKfontspec/#1/#2/id } \l_ctex_ltj_family_tl
      { \ctex_ltj_switch_family:x { \l_ctex_ltj_family_tl } }
      {
        \int_gincr:N \g__ctex_ltj_family_int
        \__ctex_ltj_fontspec:xnn
          { CJKfontspec ( \int_use:N \g__ctex_ltj_family_int ) }
          {#1} {#2}
      }
```

```
  }
\cs_new_protected_nopar:Npn \ctex_ltj_fontspec:xx #1#2
  { \use:x { \ctex_ltj_fontspec:nn {#1} {#2} } }
\cs_new_protected_nopar:Npn \__ctex_ltj_fontspec:nnn #1#2#3
  {
    \bool_if:NT \l__ctex_ltj_add_alternate_bool
      {
        \cs_if_free:cF
          { \__ctex_ltj_alternate_cs:n { reset / \l_ctex_ltj_family_tl } }
          {
            \cs_gset_eq:cc
              { \__ctex_ltj_alternate_cs:n { reset / #1 } }
              { \__ctex_ltj_alternate_cs:n { reset / \l_ctex_ltj_family_tl } }
            \cs_gset_eq:cc
              { \__ctex_ltj_alternate_cs:n { clear / #1 } }
              { \__ctex_ltj_alternate_cs:n { clear / \l_ctex_ltj_family_tl } }
          }
        \bool_set_false:N \l__ctex_ltj_add_alternate_bool
      }
    \prop_gput:Nnn \g__ctex_ltj_fontspec_prop { CJKfontspec/#2/#3/id } {#1}
    \ctex_ltj_set_family:nnn {#1} {#2} {#3}
    \ctex_ltj_switch_family:n {#1}
  }
\cs_generate_variant:Nn \__ctex_ltj_fontspec:nnn { x }
\prop_new:N \g__ctex_ltj_fontspec_prop
```

\ctex_ltj_add_font_features:n
\ctex_ltj_add_font_features:nn

```
\cs_new_protected_nopar:Npn \ctex_ltj_add_font_features:n #1
  { \ctex_ltj_add_font_features:xn { \l_ctex_ltj_family_tl } {#1} }
\cs_new_protected_nopar:Npn \ctex_ltj_add_font_features:nn #1#2
  {
    \prop_get:NnNTF \g__ctex_ltj_family_font_name_prop
      {#1} \l__ctex_ltj_tmp_tl
      {
        \prop_get:NnN \g__ctex_ltj_family_font_options_prop
          {#1} \l__ctex_ltj_font_options_clist
        \clist_put_right:Nn \l__ctex_ltj_font_options_clist {#2}
        \bool_set_true:N \l__ctex_ltj_add_alternate_bool
        \ctex_ltj_fontspec:xx
          { \exp_not:o { \l__ctex_ltj_font_options_clist } }
          { \exp_not:o { \l__ctex_ltj_tmp_tl } }
      }
      { \msg_warning:nn { ctex } { addCJKfontfeature-ignored } }
  }
\bool_new:N \l__ctex_ltj_add_alternate_bool
\cs_generate_variant:Nn \ctex_ltj_add_font_features:n  { x }
\cs_generate_variant:Nn \ctex_ltj_add_font_features:nn { x }
\msg_new:nnn { ctex } { addCJKfontfeature-ignored }
  {
    \token_to_str:N \addCJKfontfeature (s)~ignored.\\
    It~cannot~be~used~with~a~font~that~wasn't~selected~by~ctex.
  }
```

__ctex_ltj_pass_args:nnnn

为了支持字体属性可选项在前在后两种语法，给出两个辅助工具，自带展开功能。

```
\cs_new_protected:Npn \__ctex_ltj_pass_args:nnnn #1#2#3#4
  {
    \IfNoValueTF {#2}
      { \__ctex_ltj_post_arg:w {#1} {#3} {#4} }
      {
        \use:x { #1 {#2} {#3} }
        #4
      }
  }
\NewDocumentCommand \__ctex_ltj_post_arg:w { m m m O { } }
  {
    \use:x { #1 {#4} {#2} }
    #3
```

```
}

\setCJKfamilyfont
\newCJKfontfamily
\CJKfontspec
\addCJKfontfeatures
\NewDocumentCommand \setCJKfamilyfont { m o m }
  {
    \__ctex_ltj_pass_args:nnnn
      { \ctex_ltj_set_family:nnn {#1} } {#2} {#3}
      { }
  }
\NewDocumentCommand \newCJKfontfamily { o m o m }
  {
    \tl_set:Nx \l__ctex_ltj_tmp_tl
      { \IfNoValueTF {#1} { \cs_to_str:N #2 } {#1} }
    \cs_new_protected_nopar:Npx #2
      { \ctex_ltj_switch_family:n { \l__ctex_ltj_tmp_tl } }
    \__ctex_ltj_pass_args:nnnn
      { \ctex_ltj_set_family:nnn { \l__ctex_ltj_tmp_tl } } {#3} {#4}
      { }
  }
\NewDocumentCommand \CJKfontspec { o m }
  {
    \__ctex_ltj_pass_args:nnnn
      { \ctex_ltj_fontspec:nn } {#1} {#2}
      { \tex_ignorespaces:D }
  }
\NewDocumentCommand \addCJKfontfeatures { m }
  {
    \ctex_ltj_add_font_features:x {#1}
    \tex_ignorespaces:D
  }
\cs_new_eq:NN \addCJKfontfeature \addCJKfontfeatures

\setCJKmainfont
\setCJKsansfont
\setCJKmonofont
\setCJKmathfont
\defaultCJKfontfeatures
\NewDocumentCommand \setCJKmainfont { o m }
  {
    \__ctex_ltj_pass_args:nnnn
      { \ctex_ltj_set_family:nnn { \CJKrmdefault } } {#1} {#2}
      { \normalfont }
  }
\cs_new_eq:NN \setCJKromanfont \setCJKmainfont
\NewDocumentCommand \setCJKsansfont { o m }
  {
    \__ctex_ltj_pass_args:nnnn
      { \ctex_ltj_set_family:nnn { \CJKsfdefault } } {#1} {#2}
      { \normalfont }
  }
\NewDocumentCommand \setCJKmonofont { o m }
  {
    \__ctex_ltj_pass_args:nnnn
      { \ctex_ltj_set_family:nnn { \CJKttdefault } } {#1} {#2}
      { \normalfont }
  }
\NewDocumentCommand \setCJKmathfont { o m }
  {
    \__ctex_ltj_pass_args:nnnn
      { \ctex_ltj_set_family:nnn { \c__ctex_ltj_math_tl } } {#1} {#2}
      { }
  }
\NewDocumentCommand \defaultCJKfontfeatures { m }
  { \clist_gset:Nn \g__ctex_ltj_default_features_clist {#1} }
\clist_new:N \g__ctex_ltj_default_features_clist
\@onlypreamble \setCJKmainfont
\@onlypreamble \setCJKsansfont
\@onlypreamble \setCJKmonofont
\@onlypreamble \setCJKmathfont
\@onlypreamble \setCJKromanfont
\@onlypreamble \defaultCJKfontfeatures

\tl_if_exist:NF \CJKfamilydefault
```

```
  { \tl_const:Nn \CJKfamilydefault { \CJKrmdefault } }
\tl_if_exist:NF \CJKrmdefault { \tl_const:Nn \CJKrmdefault { rm } }
\tl_if_exist:NF \CJKsfdefault { \tl_const:Nn \CJKsfdefault { sf } }
\tl_if_exist:NF \CJKttdefault { \tl_const:Nn \CJKttdefault { tt } }
```

\fontfamily
\CTEX@fontfamily

修改 \fontfamily，使主要 CJK 字体族能随西文主要字体更新。

```
\RenewDocumentCommand \fontfamily { m }
  {
    \tl_set:Nx \f@family {#1}
    \CTEX@fontfamily {#1}
  }
\cs_new_protected_nopar:Npn \CTEX@fontfamily #1
  {
    \str_if_eq:nnTF {#1} { \familydefault }
      { \CJKfamily { \CJKfamilydefault } }
      { \__ctex_ltj_update_family_aux: }
  }
\cs_new_protected_nopar:Npn \__ctex_ltj_update_family_aux:
  {
    \str_case_e:nn { \f@family }
      {
        { \rmdefault }     { \CJKfamily { \CJKrmdefault } }
        { \sfdefault }     { \CJKfamily { \CJKsfdefault } }
        { \ttdefault }     { \CJKfamily { \CJKttdefault } }
        { \familydefault } { \CJKfamily { \CJKfamilydefault } }
      }
  }
```

在导言区末尾更新 \CJKfamilydefault。

```
\ctex_at_end_preamble:n { \ctex_update_default_family: }
```

\ctex_ltj_ensure_default_family:

在导言区结束确认 \CJKfamilydefault 确实存在。

```
\cs_new_protected_nopar:Npn \ctex_ltj_ensure_default_family:
  {
    \prop_if_empty:NF \g__ctex_ltj_family_font_name_prop
      {
        \ctex_ltj_family_if_exist:xNF { \CJKfamilydefault } \l__ctex_ltj_tmp_tl
          {
            \str_if_eq:eeTF { \CJKfamilydefault } { \CJKrmdefault }
              { \use:n }
              {
                \ctex_ltj_family_if_exist:xNTF { \CJKrmdefault } \l__ctex_ltj_tmp_tl
                  { \tl_gset:Nn \CJKfamilydefault { \CJKrmdefault } \use_none:n }
                  { \use:n }
              }
              {
                \prop_map_inline:Nn \g__ctex_ltj_family_font_name_prop
                  {
                    \prop_map_break:n
                      { \tl_gset_rescan:Nnn \CJKfamilydefault { } { ##1 } }
                  }
              }
          }
        \normalfont
        \ctex_ltj_update_mathfont:
      }
  }
```

\ctex_ltj_update_mathfont:

更新数学字体为实际的字体。

```
\cs_new_protected_nopar:Npn \ctex_ltj_update_mathfont:
  {
    \ctex_ltj_family_if_exist:xNTF { \c__ctex_ltj_math_tl } \l__ctex_ltj_tmp_tl
      { \ctex_ltj_update_mathfont:n { \l__ctex_ltj_tmp_tl } }
```

```
      {
        \ctex_ltj_family_if_exist:xNT { \CJKfamilydefault } \l__ctex_ltj_tmp_tl
          { \ctex_ltj_update_mathfont:n { \l__ctex_ltj_tmp_tl } }
      }
  }
\cs_new_protected_nopar:Npn \ctex_ltj_update_mathfont:n #1
  {
    \tl_const:Nx \c__ctex_ltj_math_family_tl {#1}
    \DeclareSymbolFont { \c__ctex_ltj_math_tl } { \CJK@encoding }
      { \c__ctex_ltj_math_family_tl } { \mddefault } { \updefault }
    \cs_if_free:cTF
      { \CJK@encoding/\c__ctex_ltj_math_family_tl/\bfdefault/\updefault }
      {
        \SetSymbolFont { \c__ctex_ltj_math_tl } { bold } { \CJK@encoding }
          { \c__ctex_ltj_math_family_tl } { \mddefault } { \updefault }
      }
      {
        \SetSymbolFont { \c__ctex_ltj_math_tl } { bold } { \CJK@encoding }
          { \c__ctex_ltj_math_family_tl } { \bfdefault } { \updefault }
      }
  }
```

14.3.3.6　替代字体的设置

AlternateFont
CharRange

设置替代字体的选项。

```
\keys_define:nn { ctex_ltj / fontspec }
  {
    AlternateFont  .code:n = \ctex_ltj_set_alternate_seq:n {#1} ,
    AlternateFont  .value_required:n = true ,
    CharRange .clist_set:N = \l__ctex_ltj_char_range_clist ,
    CharRange .value_required:n = true
  }
```

\ctex_ltj_set_alternate_seq:n

我们使用 || 作为替代字体序列的分隔标志。它可能被设置为活动字符，为此需要先“消毒”，同时过滤掉空元素。

```
\group_begin:
  \char_set_catcode_other:N \|
  \cs_set:Npn \__ctex_ltj_tmp:w #1
    {
      \cs_new_protected:Npn \ctex_ltj_set_alternate_seq:n ##1
        {
          \clist_if_empty:NT \l__ctex_ltj_char_range_clist
            {
              \tl_set:Nn \l__ctex_ltj_tmp_tl { ##1 }
              \tl_replace_all:Nnn \l__ctex_ltj_tmp_tl {#1} { || }
              \seq_set_split:NnV \l__ctex_ltj_tmp_seq { || } \l__ctex_ltj_tmp_tl
              \seq_set_filter:NNn \l__ctex_ltj_tmp_seq \l__ctex_ltj_tmp_seq
                { ! \tl_if_blank_p:n { ####1 } }
              \seq_concat:NNN \l__ctex_ltj_alternate_seq
                \l__ctex_ltj_alternate_seq \l__ctex_ltj_tmp_seq
            }
        }
    }
  \char_set_catcode_active:N \|
  \__ctex_ltj_tmp:w { || }
\group_end:
\seq_new:N \l__ctex_ltj_tmp_seq
\seq_new:N \l__ctex_ltj_alternate_seq
```

\ctex_ltj_set_alternate_family:nnF

如果在字体的选项中设置了 CharRange，则只设置替代字体。

```
\cs_new_protected_nopar:Npn \ctex_ltj_set_alternate_family:nnF #1#2#3
  {
```

```
    \clist_if_empty:NTF \l__ctex_ltj_char_range_clist
      {
        \__ctex_ltj_check_family:n {#1}
        \seq_if_empty:NF \l__ctex_ltj_alternate_seq
         { \ctex_ltj_save_alternate_seq:cn { \__ctex_ltj_alternate_cs:n {#1} } {#2} }
        #3
      }
      { \ctex_ltj_set_alternate_family:nn {#1} {#2} }
  }
```

\ctex_ltj_save_alternate_seq:Nn
\ctex_ltj_save_alternate_seq:Nnnwn

保存由 AlternateFont 设置的替代字体序列。

```
\cs_new_protected_nopar:Npn \ctex_ltj_save_alternate_seq:Nn #1#2
  {
    \seq_map_inline:Nn \l__ctex_ltj_alternate_seq
      { \ctex_ltj_save_alternate_seq:Nnnwnw #1 {#2} ##1 { } \q_stop }
  }
\cs_generate_variant:Nn \ctex_ltj_save_alternate_seq:Nn { c }
\NewDocumentCommand \ctex_ltj_save_alternate_seq:Nnnwnw
  { m m m +O{ } m u{ \q_stop } }
  {
    \clist_set:Nn \l__ctex_ltj_char_range_clist {#3}
    \clist_set:Nn \l__ctex_ltj_alternate_options_clist {#4}
    \__ctex_ltj_use_global_options:N \l__ctex_ltj_alternate_options_clist
    \tl_if_blank:nTF {#5}
      { \tl_set:Nn \l__ctex_ltj_tmp_tl {#2} }
      {
        \tl_set:Nn \l__ctex_ltj_tmp_tl {#5}
        \tl_replace_all:Nnn \l__ctex_ltj_tmp_tl { * } {#2}
      }
    \use:x
      {
        \ctex_ltj_save_alternate_family:Nnnn \exp_not:N #1
          { \exp_not:o { \l__ctex_ltj_char_range_clist } }
          { \exp_not:o { \l__ctex_ltj_alternate_options_clist } }
          { \exp_not:o { \l__ctex_ltj_tmp_tl } }
      }
  }
\clist_new:N \l__ctex_ltj_alternate_options_clist
```

\ctex_ltj_set_alternate_family:nn

设置选项 CharRange 范围内的替代字体。如果已经定义了主字体，我们也马上定义替代字体，否则只保存起来备用。

```
\cs_new_protected_nopar:Npn \ctex_ltj_set_alternate_family:nn #1#2
  {
    \__ctex_ltj_update_family_uid:N \l__ctex_ltj_font_options_clist
    \__ctex_ltj_use_global_options:N \l__ctex_ltj_font_options_clist
    \ctex_ltj_set_alternate_family:coonn
      { \__ctex_ltj_alternate_cs:n {#1} }
      { \l__ctex_ltj_char_range_clist }
      { \l__ctex_ltj_font_options_clist } {#2} {#1}
  }
\cs_new_protected_nopar:Npn \ctex_ltj_set_alternate_family:Nnnnn #1#2#3#4#5
  {
    \prop_get:NnNT \g__ctex_ltj_family_name_prop {#5} \l__ctex_ltj_base_family_tl
      { \ctex_ltj_set_alternate_family:nnn {#2} {#3} {#4} }
    \ctex_ltj_save_alternate_family:Nnnn #1 {#2} {#3} {#4}
  }
\cs_generate_variant:Nn \ctex_ltj_set_alternate_family:Nnnnn { coo }
```

\ctex_ltj_save_alternate_family:Nnnn

保存替代字体序列的定义，以备定义主字体时使用。

```
\cs_new_protected_nopar:Npn \ctex_ltj_save_alternate_family:Nnnn #1#2#3#4
  {
    \cs_if_exist:NF #1 { \cs_set_eq:NN #1 \prg_do_nothing: }
    \cs_gset_protected_nopar:Npx #1
```

```
      { \exp_not:o { #1 \ctex_ltj_set_alternate_family:nnn {#2} {#3} {#4} } }
  }
```

\ctex_ltj_set_alternate_family:nnn　实际定义替代字体族。

```
\cs_new_protected_nopar:Npn \ctex_ltj_set_alternate_family:nnn #1#2#3
  {
    \group_begin:
    \__ctex_ltj_change_encoding:
    \cs_set_eq:NN \CJKfamily \use_none:n
    \ctex_ltj_swap_cs:NN
      \DeclareFontShape@ \ctex_ltj_declare_alternate_shape:nnnnnn
    \tl_set:Nn \l__ctex_ltj_char_range_clist {#1}
    \fontspec_set_family:Nnn \l__ctex_ltj_alternate_family_tl {#2} {#3}
    \group_end:
  }
\tl_new:N \l__ctex_ltj_alternate_family_tl
```

\ctex_ltj_swap_cs:NN　交换两个控制序列的意义。

```
\cs_new_protected:Npn \ctex_ltj_swap_cs:NN #1#2
  {
    \cs_set_eq:NN \__ctex_ltj_tmp:w #1
    \cs_set_eq:NN #1 #2
    \cs_set_eq:NN #2 \__ctex_ltj_tmp:w
    \cs_undefine:N \__ctex_ltj_tmp:w
  }
```

LTJFONTUID
__ctex_ltj_update_family_uid:N　fontspec 在一个字体族的选项和字体名称相同的时候，就不定义新字体。为了避免混淆替代字体的设置，我们新定义一个虚拟的选项 LTJFONTUID，确保 fontspec 对 CJK 字体族总是定义新字体。

```
\keys_define:nn { fontspec } { LTJFONTUID .code:n = }
\cs_new_protected_nopar:Npn \__ctex_ltj_update_family_uid:N #1
  {
    \int_gincr:N \g__ctex_ltj_family_int
    \clist_put_right:Nx #1 { LTJFONTUID = \int_use:N \g__ctex_ltj_family_int }
  }
\int_new:N \g__ctex_ltj_family_int
```

\ctex_ltj_declare_alternate_shape:nnnnnn　在定义替代字体的字形时，通过字符范围与主字体的对应字形关联起来。\DeclareFontShape@ 一个有六个参数，我们只需要使用它的第三个参数 ⟨*series*⟩ 和第四个参数 ⟨*shape*⟩。

```
\cs_new_protected:Npn \ctex_ltj_declare_alternate_shape:nnnnnn #1#2#3#4#5#6
  {
    \ctex_ltj_declare_alternate_shape:nnnnnn {#1} {#2} {#3} {#4} {#5} {#6}
    \ctex_ltj_set_alternate_shape:Nnnnnnn \l__ctex_ltj_char_range_clist
      { \l__ctex_ltj_base_family_tl } {#3} {#4}
      { \l_fontspec_family_tl } {#3} {#4}
  }
```

\ctex_ltj_set_alternate_shape:Nnnnnnn　与 LuaTeX-ja 的 \DeclareAlternateKanjiFont 的功能类似，区别是固定编码为 \CJK@encoding。这个设置总是全局的。

```
\cs_new_protected_nopar:Npn \ctex_ltj_set_alternate_shape:Nnnnnnn #1#2#3#4#5#6#7
  {
    \clist_map_inline:Nn #1
      {
       \prop_get:NnNTF \g__ctex_ltj_char_range_prop { ##1 } \l__ctex_ltj_char_range_tl
          {
            \ctex_ltj_set_alternate_shape:nnN { #2/#3/#4 } { #5/#6/#7 }
              \l__ctex_ltj_char_range_tl
          }
          { \ctex_ltj_set_alternate_shape:nnn { #2/#3/#4 } { #5/#6/#7 } { ##1 } }
      }
```

```
    \__ctex_ltj_save_alternate_shape:cnn
      { \__ctex_ltj_alternate_cs:n { clear / \l__ctex_ltj_base_CJKfamily_tl } }
      { luatexja.jfont.clear_alt_font_latex }
      { '\lua_escape:e { \CJK@encoding/#2/#3/#4 }' }
  }
```

\ctex_ltj_set_alternate_shape:nnn

我们使用 -> 而不是像 LuaTeX-ja 一样使用 - 作为区间的分隔符。LuaTeX-ja 支持使用负数来引用由 JFM 设置的字符类。如果使用 - 作为分隔符，那么负数单独使用时，就需要把它放在两层花括号之内（例如 {{-1}}），或者使用类似 {-1}-{-1} 的形式才不会解释错误。

```
\NewDocumentCommand \ctex_ltj_set_alternate_shape:nnn
  { m m > { \SplitArgument { 1 } { -> } } m }
  { \ctex_ltj_set_alternate_shape:nnnn {#1} {#2} #3 }
\cs_new_protected_nopar:Npn \ctex_ltj_set_alternate_shape:nnnn #1#2#3#4
  {
    \ctex_ltj_set_alternate_shape:n
      {
        \IfNoValueTF {#4}
          { \int_eval:n {#3} , \int_eval:n {#3} , }
          {
            \int_eval:n { \tl_if_blank:nTF {#3} { "80 } {#3} } ,
            \int_eval:n { \tl_if_blank:nTF {#4} { "10FFFF } {#4} } ,
          }
        '\lua_escape:e { \CJK@encoding/#2 }' ,
        '\lua_escape:e { \CJK@encoding/#1 }'
      }
  }
\cs_new_protected_nopar:Npn \ctex_ltj_set_alternate_shape:n #1
  {
    \lua_now:e { luatexja.jfont.set_alt_font_latex ( #1 ) }
    \__ctex_ltj_save_alternate_shape:cnn
      { \__ctex_ltj_alternate_cs:n { reset / \l__ctex_ltj_base_CJKfamily_tl } }
      { luatexja.jfont.set_alt_font_latex } {#1}
  }
```

\ctex_ltj_set_alternate_shape:nnN

若字符范围预先由 declarecharrange 声明，则可以直接使用。

```
\cs_new_protected_nopar:Npn \ctex_ltj_set_alternate_shape:nnN #1#2#3
  {
    \tl_map_inline:Nn #3
      {
        \ctex_ltj_set_alternate_shape:n
          {
            ##1 ,
            '\lua_escape:e { \CJK@encoding/#2 }' ,
            '\lua_escape:e { \CJK@encoding/#1 }'
          }
      }
  }
```

__ctex_ltj_save_alternate_shape:Nnn

将实际设置的替换字形保存起来用于清除或恢复。暂时令 \l__ctex_ltj_base_family_tl 为 \scan_stop: 是让它不被展开，使得替换字体的设置可以在 \addCJKfontfeature 中直接使用。

```
\cs_new_protected_nopar:Npn \__ctex_ltj_save_alternate_shape:Nnn #1#2#3
  {
    \group_begin:
    \cs_if_exist:NF #1 { \cs_set_eq:NN #1 \prg_do_nothing: }
    \cs_set_eq:NN \l__ctex_ltj_base_family_tl \scan_stop:
    \cs_set_eq:NN \lua_escape:e \scan_stop:
    \cs_gset_protected_nopar:Npx #1
      { \exp_not:o {#1} \exp_not:N \lua_now:e { #2 ( #3 ) } }
    \group_end:
  }
\cs_generate_variant:Nn \__ctex_ltj_save_alternate_shape:Nnn { c }
```

clearalternatefont
resetalternatefont

清除和重置操作总是全局的。

```
\keys_define:nn { ctex }
  {
    clearalternatefont    .code:n =
      { \clist_map_function:xN {#1} \ctex_ltj_clear_alternate_font:n } ,
    resetalternatefont    .code:n =
      { \clist_map_function:xN {#1} \ctex_ltj_reset_alternate_font:n } ,
    clearalternatefont .default:n = \l_ctex_ltj_family_tl ,
    resetalternatefont .default:n = \l_ctex_ltj_family_tl
  }
\cs_new_protected_nopar:Npn \ctex_ltj_clear_alternate_font:n #1
  {
    \group_begin:
      \ctex_ltj_family_if_exist:xNTF {#1} \l__ctex_ltj_base_family_tl
        {
          \cs_if_exist_use:cT { \__ctex_ltj_alternate_cs:n { clear / #1 } }
            {
              \prop_gput:Nno \g__ctex_ltj_reset_alternate_prop
                {#1} { \l__ctex_ltj_base_family_tl }
              \tl_set_eq:NN \CJK@family \l__ctex_ltj_base_family_tl
              \selectfont
            }
        }
        { \__ctex_ltj_family_unknown_warning:n {#1} }
    \group_end:
  }
\cs_new_protected_nopar:Npn \ctex_ltj_reset_alternate_font:n #1
  {
    \group_begin:
      \prop_gpop:NnNT \g__ctex_ltj_reset_alternate_prop {#1} \CJK@family
        {
          \tl_set_eq:NN \l__ctex_ltj_base_family_tl \CJK@family
          \use:c { \__ctex_ltj_alternate_cs:n { reset / #1 } }
          \selectfont
        }
    \group_end:
  }
\prop_new:N \g__ctex_ltj_reset_alternate_prop
\cs_generate_variant:Nn \clist_map_function:nN { x }
```

declarecharrange

预先声明字符范围。

```
\keys_define:nn { ctex }
  {
    declarecharrange .code:n = \ctex_ltj_declare_char_range:x {#1} ,
    declarecharrange .value_required:n = true
  }
\cs_new_protected_nopar:Npn \ctex_ltj_declare_char_range:n #1
  { \clist_map_inline:nn {#1} { \__ctex_ltj_declare_char_range:nn ##1 } }
\cs_generate_variant:Nn \ctex_ltj_declare_char_range:n { x }
\cs_new_protected_nopar:Npn \__ctex_ltj_declare_char_range:nn #1#2
  { \use:x { \ctex_ltj_declare_char_range:nn { \tl_trim_spaces:n {#1} } } {#2} }
```

\ctex_ltj_declare_char_range:nn
\g__ctex_ltj_char_range_prop

#1 是名字，#2 是范围。

```
\cs_new_protected_nopar:Npn \ctex_ltj_declare_char_range:nn #1#2
  {
    \tl_clear:N \l__ctex_ltj_char_range_tl
    \clist_map_function:nN {#2} \ctex_ltj_save_char_range:n
    \prop_gput:Nno \g__ctex_ltj_char_range_prop {#1} { \l__ctex_ltj_char_range_tl }
    \ctex_ltj_def_char_range_key:n {#1}
    \tl_clear:N \l__ctex_ltj_char_range_tl
  }
\tl_new:N \l__ctex_ltj_char_range_tl
\prop_new:N \g__ctex_ltj_char_range_prop
```

\ctex_ltj_save_char_range:n　预先解释字符区间的意义。

```
\NewDocumentCommand \ctex_ltj_save_char_range:n
  { > { \SplitArgument { 1 } { -> } } m }
  { \ctex_ltj_save_char_range:nn #1 }
\cs_new_protected_nopar:Npn \ctex_ltj_save_char_range:nn #1#2
  {
    \tl_put_right:Nx \l__ctex_ltj_char_range_tl
      { {
          \IfNoValueTF {#2}
            { \int_eval:n {#1} , \int_eval:n {#1} }
            {
              \int_eval:n { \tl_if_blank:nTF {#1} { "80 } {#1} } ,
              \int_eval:n { \tl_if_blank:nTF {#2} { "10FFFF } {#2} }
            }
      } }
  }
```

\ctex_ltj_def_char_range_key:n　在字体设置选项中定义字符范围键。

```
\cs_new_protected_nopar:Npn \ctex_ltj_def_char_range_key:n #1
  {
    \keys_if_exist:nnF { ctex_ltj / fontspec } {#1}
      {
        \keys_define:nn { ctex_ltj / fontspec }
          { #1 .code:n = \ctex_ltj_char_range_key:nn {#1} { ##1 } }
      }
  }
```

\ctex_ltj_char_range_key:nn　如果字符范围键没有值，则只设置的这个字符范围内的替代字体。

```
\cs_new_protected:Npn \ctex_ltj_char_range_key:nn #1#2
  {
    \tl_if_blank:nTF {#2}
      { \tl_set:Nn \l__ctex_ltj_char_range_clist {#1} }
      {
        \clist_if_empty:NT \l__ctex_ltj_char_range_clist
          {
            \tl_set:Nn \l__ctex_ltj_tmp_tl { {#1} }
            \__ctex_ltj_char_range_parse_feature:w #2 \q_stop
          }
      }
  }
```

__ctex_ltj_char_range_parse_feature:w　可以使用加方括号的方式，通过文件名来调用字体。这容易与字体选项混淆。例如，需要将 [simsun.ttc] 设置为 range 的主字体，就需要使用 range={{[simsun.ttc]}} 或者 []{[simsun.ttc]}。下面的目的是，支持直接使用 [simsun.ttc] 和 [...][simsun.ttc]。

```
\NewDocumentCommand \__ctex_ltj_char_range_parse_feature:w
  { +o o u { \q_stop } }
  {
    \exp_args:NNf \tl_put_right:Nn \l__ctex_ltj_tmp_tl
      {
        \IfNoValueTF {#1} { {#3} }
          {
            \IfNoValueTF {#2}
              { \tl_if_blank:nTF {#3} { { [#1] } } { [ {#1} ] {#3} } }
              { [ {#1} ] { [#2] } }
          }
      }
    \seq_put_right:No \l__ctex_ltj_alternate_seq { \l__ctex_ltj_tmp_tl }
  }
```

14.3.3.7　其它设置

在抄录环境中禁用 autospacing 和 autoxspacing。然而，LuaTeX-ja 还是会使 JAchar 自动折行。没有看到有简单的禁用折行的办法，可能需要设置所有的 JAchar 的 prebreakpenalty 或 postbreakpenalty 为 10000：

```
\directlua
  {
    luatexja.isglobal = tex.globaldefs > 0 and "global" or ""
    for i = 0x80, 0x10FFFF do
      if luatexja.charrange.jcr_table_main[i] > 0 and
         luatexja.charrange.jcr_table_main[i] < 218 and
         luatexja.charrange.is_japanese_char_curlist(i) then
        luatexja.stack.set_stack_table(luatexja.stack_table_index.PRE + i, 10000)
      end
    end
  }
```

```
\AtBeginDocument
  {
    \ctex_appto_cmd:NnnTF \verbatim@font
      { \char_set_catcode_letter:n { 64 } }
      { \CTEX@verbatim@font@hook }
      { }
      { \ctex_patch_failure:N \verbatim@font }
  }
\cs_new_protected_nopar:Npn \CTEX@verbatim@font@hook
  { \ltjsetparameter { autospacing = false , autoxspacing = false } }
```

__ctex_ltjitaliccorr

LaTeX 的倾斜校正也要重新定义。

```
⟨@@=⟩
\cs_set_eq:NN \@@italiccorr \/
⟨@@=ctex⟩
```

\ctex_ltj_set_kanjiskip:N
\ctex_ltj_set_xkanjiskip:N

\ltjsetkanjiskip 和 \ltjsetxkanjiskip 是相应的 \ltjsetparameter 的快捷方式，在使用他们时，要注意先使用 \ltj@setpar@global。

```
\cs_new_protected_nopar:Npn \ctex_ltj_set_kanjiskip:N
  { \ltj@setpar@global \ltjsetkanjiskip }
\cs_new_protected_nopar:Npn \ctex_ltj_set_xkanjiskip:N
  { \ltj@setpar@global \ltjsetxkanjiskip }
```

```
⟨/luatex⟩
```

14.3.4　ctex-engine-uptex.def

```
⟨*uptex|aptex⟩
```

按 CJK 的命名习惯模拟部分命令，并设置默认字体。

```
\tl_set:Nn \CJKrmdefault { zhrm }
\tl_set:Nn \CJKsfdefault { zhsf }
\tl_set:Nn \CJKttdefault { zhtt }
\tl_set:Nn \CJKfamilydefault { \CJKrmdefault }
\tl_set:Nn \kanjifamilydefault { \CJKfamilydefault }
\RenewDocumentCommand \rmfamily { }
  {
    \not@math@alphabet \rmfamily \mathrm
    \romanfamily \rmdefault
    \kanjifamily \CJKrmdefault
    \selectfont
  }
\RenewDocumentCommand \sffamily { }
  {
```

```
    \not@math@alphabet \sffamily \mathsf
    \romanfamily \sfdefault
    \kanjifamily \CJKsfdefault
    \selectfont
  }
\RenewDocumentCommand \ttfamily { }
  {
    \not@math@alphabet \ttfamily \mathtt
    \romanfamily \ttdefault
    \kanjifamily \CJKttdefault
    \selectfont
  }
\NewDocumentCommand \CJKfamily { m }
  {
    \kanjifamily {#1}
    \selectfont
  }
```

将 upLaTeX 的默认字体由 mc 改为 zhrm，并启用 \jfam。

```
\DeclareErrorKanjiFont{JY2}{zhrm}{m}{n}{10}
\DeclareKanjiSubstitution{JY2}{zhrm}{m}{n}
\DeclareKanjiSubstitution{JT2}{zhrm}{m}{n}
\DeclareSymbolFont{mincho}{JY2}{zhrm}{m}{n}
\SetSymbolFont{mincho}{bold}{JY2}{zhrm}{bx}{n}
\jfam \symmincho
```

在导言区末尾更新 \CJKfamilydefault。

```
\ctex_at_end_preamble:n { \ctex_update_default_family: }
```

使修改立刻生效，保证导言区字体族正确。

```
\normalfont
```

\em　取消 upLaTeX 对 \em 使用 \mcfamily、\gtfamily 命令的重定义，恢复 LaTeX 2ε 对 \em 的原始定义。如果用户已经重定义了 \em，则新定义保持不变。upLaTeX 2016/05/07u00 的定义有所变化，这一行为可以由用户通过 platexrelease 包改变，需要分支处理。

```
\ctex_patch_cmd_once:NnnnTF \em
  { \ExplSyntaxOff }
  { \eminnershape \else \gtfamily \itshape }
  { \eminnershape \else \itshape }
  { }
  {
    \ctex_patch_cmd:Nnn \em
      { \mcfamily \upshape \else \gtfamily \itshape }
      { \eminnershape \else \itshape }
  }
\cs_set_nopar:Npn \eminnershape { \upshape }
```

\ctex_set_upfamily:nnn　将 NFSS 字体族 #1 设置为 JFM 字体名 #2，粗体形式字体名 #3。其中字体名形如 upzhserif，不包括表示方向的后缀 -h 与 -v。粗体字体名为空时不设置该字形。本命令不设置字体映射，需要复用已有的字体映射或另行设置。

```
\cs_new_protected_nopar:Npn \ctex_set_upfamily:nnn #1 #2 #3
  {
    \DeclareKanjiFamily{JY2}{#1}{}
    \DeclareKanjiFamily{JT2}{#1}{}
    \DeclareFontShape{JY2}{#1}{m}{n}{<->~ #2-h}{}
    \DeclareFontShape{JT2}{#1}{m}{n}{<->~ #2-v}{}
    \tl_if_empty:nF { #3 }
      {
        \DeclareFontShape{JY2}{#1}{bx}{n}{<->~ #3-h}{}
        \DeclareFontShape{JT2}{#1}{bx}{n}{<->~ #3-v}{}
      }
  }
```

```
\@onlypreamble \ctex_set_upfamily:nnn
```

\ctex_set_upmap:nnn

设置 upTEX 字体映射。#1 是形如 upserif 的 PS TFM 字体名，不带表示粗体的后缀 b 与表示排版方向的后缀 -h 与 -v。#2 与 #3 是普通与粗体的实际字体名。

```
\cs_new_protected_nopar:Npn \ctex_set_upmap:nnn #1 #2 #3
  {
    \ctex_set_zhmap:n
      {
        \special{ pdf:mapline~ #1-h~   UniGB-UTF16-H~ #2 }
        \special{ pdf:mapline~ #1-v~   UniGB-UTF16-V~ #3 }
        \tl_if_empty:nF { #3 }
          {
            \special{ pdf:mapline~ #1b-h~  UniGB-UTF16-H~ #3 }
            \special{ pdf:mapline~ #1b-v~  UniGB-UTF16-V~ #3 }
          }
      }
  }
\@onlypreamble \ctex_set_upmap:nnn
```

\ctex_set_upfonts:nnnnnn

设置 upTEX 基本字体映射，按 zhmetrics-uptex 的定义，依次设置衬线体正、粗、意大利，无衬线体正、粗，等宽体正——共 6 种字体，并分横排及直排。

```
\cs_new_protected_nopar:Npn \ctex_set_upfonts:nnnnnn #1 #2 #3 #4 #5 #6
  {
    \ctex_set_upmap:nnn { upserif } { #1 } { #2 }
    \ctex_set_upmap:nnn { upserifit } { #3 } {}
    \ctex_set_upmap:nnn { upsans } { #4 } { #5 }
    \ctex_set_upmap:nnn { upmono } { #6 } {}
  }
\@onlypreamble \ctex_set_upfonts:nnnnnn
```

everysel 宏包（2011/10/28）未考虑 upLaTeX 对 \selectfont 的修改，需要引入 pxeverysel 宏包。

```
\RequirePackage { pxeverysel }
```

⟨/uptex|aptex⟩

14.3.5　调整 \CJKfamilydefault

\ctex_update_default_family:

在导言区结束，如果 \CJKfamilydefault 没有被更改，则在此时根据西文字体的情况更新 \CJKfamilydefault。xeCJK 已经有这个功能，不需要再调整。

```
⟨*pdftex|luatex|uptex|aptex⟩
\cs_new_protected_nopar:Npn \ctex_update_default_family:
  {
    \tl_if_eq:NNT \CJKfamilydefault \l__ctex_family_default_init_tl
      {
        \group_begin:
          \cs_set_eq:NN \__ctex_family_default_wrap:n \exp_not:n
          \tl_gset:Nx \CJKfamilydefault
            {
              \str_case:onF { \familydefault }
                {
                  { \rmdefault } { \exp_not:N \CJKrmdefault }
                  { \sfdefault } { \exp_not:N \CJKsfdefault }
                  { \ttdefault } { \exp_not:N \CJKttdefault }
                }
                { \CJKfamilydefault }
            }
        \group_end:
      }
```

使用 LuaLaTeX 时，自动调整得到的 \CJKfamilydefault 可能没有定义，需要确认它的存在性。使用 CJK 宏包或 upLaTeX 时，C19rm、JY2rm 等总是有定义的，不需要确认。

```
⟨*luatex⟩
    \ctex_ltj_ensure_default_family:
⟨/luatex⟩
  }
```

\l__ctex_family_default_init_tl

往 \CJKfamilydefault 中加入标志，用于判断它是否被更改。

```
\tl_new:N \l__ctex_family_default_init_tl
\cs_new_eq:NN \__ctex_family_default_wrap:n \use:n
\tl_set:Nx \l__ctex_family_default_init_tl
  {
    \exp_not:N \__ctex_family_default_wrap:n
      { \exp_not:o { \CJKfamilydefault } }
  }
\tl_gset_eq:NN \CJKfamilydefault \l__ctex_family_default_init_tl
⟨/pdftex|luatex|uptex|aptex⟩
```

14.3.6　操作系统的判断

\ctex_detected_platform:

在 LuaTeX 下直接用调用 os.name 来判断。

```
⟨*luatex⟩
\cs_new_protected_nopar:Npn \ctex_detected_platform:
  {
    \tl_gset:Nx \g__ctex_fontset_tl
      {
        \lua_now:e
          {
            if ~ os.name == 'windows' then ~
              tex.sprint ( 'windows' )
            elseif ~ os.name == 'macosx' then ~
              tex.sprint ( 'mac' )
            else ~
              tex.sprint ( 'fandol' )
            end
          }
      }
  }
⟨/luatex⟩
```

ApTeX 可以使用 \ngostype 来判断。

```
⟨*aptex⟩
\cs_new_protected_nopar:Npn \ctex_detected_platform:
  {
    \tl_gset:Nx \g__ctex_fontset_tl
      {
        \str_case:onF { \ngostype }
          {
            { Win32 }  { windows }
            { Win64 }  { windows }
            { Darwin } { mac }
          }
          { fandol }
      }
  }
⟨/aptex⟩
```

pdfTeX 和 XeTeX 下则依据 /dev/null 和 nul: 的存在性以及文件系统的大小写敏感性来判断。Mac OS X 的大小写敏感性在安装时是可选的。为了保险起见，这里的判断很繁琐，最多要进行 4 次文件操作！

```
⟨*xetex|pdftex|uptex⟩
```

```
\cs_new_protected_nopar:Npn \ctex_detected_platform:
  {
    \file_if_exist:nTF { /dev/null }
      {
        \file_if_exist:nTF { nul: }
          {
            \file_if_exist:nTF { \c__ctex_upper_case_file_str }
              { \ctex_if_macosx:TF { mac } { windows } }
              { \ctex_if_macosx:TF { mac } { fandol } }
          }
          { \ctex_if_macosx:TF { mac } { fandol } }
      }
      { \tl_gset:Nn \g__ctex_fontset_tl { windows } }
  }
\str_const:Nx \c__ctex_upper_case_file_str
  { \exp_args:No \str_upper_case:n { \g_file_curr_name_str } }
```

\ctex_if_macosx:TF　以 /Library/Fonts/Songti.ttc 为特征文件判断 Mac OS X。

```
\cs_new_protected_nopar:Npn \ctex_if_macosx:TF #1#2
  {
    \file_if_exist:nTF { \c__ctex_macosx_file_str }
      { \tl_gset:Nn \g__ctex_fontset_tl {#1} }
      { \tl_gset:Nn \g__ctex_fontset_tl {#2} }
  }
\str_const:Nn \c__ctex_macosx_file_str { /Library/Fonts/Songti.ttc }
⟨/xetex|pdftex|uptex⟩
```

14.3.7　hyperref 兼容性处理

现在处理各个引擎下的 PDF 中文书签问题。根据编译引擎与文件编码的不同，ctex 向 hyperref 传递适当的参数，完成中文书签的正确设置。用户仍需要自己载入 hyperref 宏包。

\ctex_hypersetup:n　如果已经载入 hyperref 宏包，则直接使用其定义设置选项；否则 \ctex_hypersetup:n 的效果与 \PassOptionsToPackage 一致，只传递宏包参数。如果用户不载入 hyperref 宏包，相关参数即被丢弃。

```
⟨*class|ctex⟩
\@ifpackageloaded { hyperref }
  {
    \cs_new_protected_nopar:Npn \ctex_hypersetup:n #1
      { \hypersetup {#1} }
  }
  {
    \cs_new_protected_nopar:Npn \ctex_hypersetup:n #1
      { \PassOptionsToPackage {#1} { hyperref } }
  }
⟨/class|ctex⟩
```

在 pdfTEX 下使用 GBK 编码，DVIPDFM*x* 驱动可以直接用它的 \special 命令，其它模式用 xCJK2uni 宏包处理。使用 UTF-8 编码时，CJKutf8 已经处理了书签问题，但仍需要设置 pdfencoding 为 unicode，目的是在书签的开头写入 BOM (\376\377)，提示这是 UTF-16BE 字节流。

```
⟨*pdftex⟩
\ctex_hypersetup:n { driverfallback = dvipdfmx }
\str_if_eq:onTF { \l__ctex_encoding_tl } { GBK }
  {
    \ctex_hypersetup:n { CJKbookmarks = true }
    \sys_if_output_pdf:TF
      { \ctex_at_end_package:nn { hyperref } { \RequirePackage { xCJK2uni } } }
      {
        \ctex_at_end_package:nn { hyperref }
```

```
          {
            \str_if_eq:onTF { \Hy@driver } { hdvipdfm }
              {
                \AtBeginShipoutFirst
                  { \special { pdf:tounicode~GBK-EUC-UCS2 } }
              }
              { \RequirePackage { xCJK2uni } }
          }
      }
  }
  { \ctex_hypersetup:n { pdfencoding = unicode } }
⟨/pdftex⟩
```

在 XeTeX 下，hyperref 在处理带有非 ASCII 字符和 \% 的书签时有问题[17]。事实上，hyperref 在驱动文件 hxetex.def 中设置了 \Hy@unicodetrue，从而书签总是会被 \HyPsd@ConvertToUnicode 转化成 UTF-16BE 编码的形式（抄录自 \pdfstringdef的定义）：

```
\ifHy@unicode
  \HyPsd@ConvertToUnicode#1%
  \ifx\HyPsd@pdfencoding\HyPsd@pdfencoding@auto
    \ltx@IfUndefined{StringEncodingConvertTest}{%
    }{%
      \EdefUnescapeString\HyPsd@temp#1%
      \ifxetex
        \let\HyPsd@UnescapedString\HyPsd@temp
        \StringEncodingConvertTest\HyPsd@temp\HyPsd@temp
                                  {utf16be}{ascii-print}{%
          \EdefEscapeString\HyPsd@temp\HyPsd@temp
          \global\let#1\HyPsd@temp
          \HyPsd@EscapeTeX#1%
          \Hy@unicodefalse
        }{%
         \HyPsd@ToBigChars#1%
        }%
```

通过宏包选项 pdfencoding=unicode 设置 \HyPsd@pdfencoding 为 unicode，可以避免随后再将书签从 UTF-16BE 字节流转化回正常字符（其中使用的 \HyPsd@ToBigChars 没有考虑书签中含有 \% 的情况）。Heiko Oberdiek 在 README 中说明了将书签转化回正常字符的意图：避免 XDVIPDFMX 的警告[18]：

```
** WARNING ** Failed to convert input string to UTF16...
```

XeTeX 的维护者 Khaled Hosny 已经注意到了这个问题[19]。需要注意的是，hxetex.def 重载了宏包选项 unicode，目的是不能设置它为 false，但也导致它不会改变 \HyPsd@pdfencoding。如果 hyperref 先于 CTeX 被载入，那么 unicode 选项是没有意义的。因此要通过意义相同但在 XeTeX 下更保险的 pdfencoding 选项来设置。为了与 XeTeX 下的行为一致（使用 \HyPsd@LoadUnicode 载入 puenc.def），在 LuaTeX 下也启用这个选项。

```
⟨*xetex|luatex⟩
\ctex_hypersetup:n { pdfencoding = unicode }
⟨/xetex|luatex⟩
```

我们假定 upTeX 使用 DVIPDFM*x* 驱动输出，于是使用与 pdfTeX 类似的设置。注意 upTeX 需要使用 UTF8-UTF16 的编码转换。

```
⟨*uptex|aptex⟩
\ctex_hypersetup:n { driverfallback = dvipdfmx }
\ctex_at_end_package:nn { hyperref }
  { \AtBeginShipoutFirst { \special { pdf:tounicode~ UTF8-UTF16 } } }
⟨/uptex|aptex⟩

⟨*pdftex|xetex|luatex|uptex|aptex⟩
```

[17] https://github.com/CTeX-org/ctex-kit/issues/39
[18] http://project.ktug.org/dvipdfmx/mailman/dvipdfmx/2009-December/000153.html
[19] http://tug.org/pipermail/tex-live/2013-December/034613.html

14.3.8 CJKfntef、xeCJKfntef 相关设置

CTEX 宏集对 pdfTEX 与 XETEX 引擎，分别载入 CJKfntef 或 xeCJKfntef 宏包，并关闭宏包默认的彩色等多余格式。

载入 CJKfntef 或 xeCJKfntef 并做适当格式设置。有关 \CTEX 开头的宏定义是过时命令，仅做兼容性保留。

```
⟨*pdftex⟩
\RequirePackage { CJKfntef }
\normalem
\cs_new_protected_nopar:Npn \__ctex_clear_fntef_color:n #1
  { \tl_clear:c { CJK#1color } }
⟨/pdftex⟩
⟨*xetex⟩
\RequirePackage { xeCJKfntef }
\@ifpackagelater { xeCJKfntef } { 2014/11/04 }
  {
    \cs_new_protected_nopar:Npn \__ctex_clear_fntef_color:n #1
      { \xeCJKsetup { #1 / format = { } } }
  }
  {
    \cs_new_protected_nopar:Npn \__ctex_clear_fntef_color:n #1
      { \tl_clear:c { CJK#1color } }
  }
⟨/xetex⟩
⟨*luatex|uptex|aptex⟩
\msg_new:nnn { ctex } { fntef-not-available }
⟨luatex⟩  { Functions~ of~ `CJKfntef'~ is~ not~ available~ in~ LuaLaTeX. }
⟨uptex⟩  { Functions~ of~ `CJKfntef'~ is~ not~ available~ in~ upLaTeX. }
⟨aptex⟩  { Functions~ of~ `CJKfntef'~ is~ not~ available~ in~ ApLaTeX. }
\msg_warning:nn { ctex } { fntef-not-available }
⟨/luatex|uptex|aptex⟩
\clist_map_inline:nn
  { underdot , underline , underdblline , underwave , sout , xout }
⟨*pdftex|xetex⟩
  {
    \__ctex_clear_fntef_color:n {#1}
    \cs_new_protected_nopar:cpx { CTEX#1 }
      {
        \msg_warning:nnnn { ctex } { deprecated-command } { \exp_not:c { CTEX#1 } }
          { You~ can~ use~ the~ command~ with~ prefix~ \exp_not:N \CJK~ instead. }
        \exp_not:c { CJK#1 }
      }
  }
\cs_new_protected_nopar:Npn { \CTEXfilltwosides }
  {
    \msg_warning:nnnn { ctex } { deprecated-environment } { CTEXfilltwosides }
      { You~ can~ use~ `CJKfilltwosides'~ environment~ instead. }
    \CJKfilltwosides
  }
\cs_new_protected_nopar:Npn { \endCTEXfilltwosides } { \endCJKfilltwosides }
⟨/pdftex|xetex⟩
⟨*luatex|uptex|aptex⟩
  { \cs_new_eq:cN { CTEX#1 } \use:n }
\cs_new_eq:NN \CTEXfilltwosides \use_none:n
\cs_new_eq:NN \endCTEXfilltwosides \prg_do_nothing:
⟨/luatex|uptex|aptex⟩
⟨*pdftex⟩
\clist_map_inline:nn
  {
    underdotbasesep ,   underdotsep ,     underlinebasesep ,
    underlinesep ,      underdbllinesep , underdbllinebasesep ,
    underwavebasesep ,  underwavesep ,    southeight ,
    underdotcolor ,     underwavecolor ,  underlinecolor ,
    underdbllinecolor , soutcolor ,       xoutcolor
```

```
  }
  {
    \cs_new_eq:cc { CTEX#1 } { CJK#1 }
    \cs_set_nopar:cpx { CJK#1 } { \exp_not:c { CTEX#1 } }
  }
⟨/pdftex⟩
```

14.3.9 \ccwd 的更新

\ctex_update_ccwd:
\ccwd

```
\cs_new_protected_nopar:Npn \ctex_update_ccwd:
⟨*pdftex|xetex⟩
  {
    \hbox_set:Nn \l__ctex_tmp_box { \CJKglue }
    \dim_set:Nn \ccwd { \box_wd:N \l__ctex_tmp_box + \f@size \p@ }
  }
⟨/pdftex|xetex⟩
⟨*luatex⟩
  { \skip_set:Nn \ccwd { \ltjgetparameter { kanjiskip } + \zw } }
⟨/luatex⟩
⟨*uptex|aptex⟩
  { \skip_set:Nn \ccwd { 1zw + \tex_kanjiskip:D } }
⟨/uptex|aptex⟩
\dim_new:N \ccwd
```

\ctex_update_ccglue:

更新字间距。

```
\cs_new_protected_nopar:Npn \ctex_update_ccglue:
⟨*pdftex|xetex⟩
  {
    \cs_set_protected_nopar:Npn \CJKglue
      { \skip_horizontal:N \l__ctex_ccglue_skip }
  }
⟨/pdftex|xetex⟩
⟨*luatex⟩
  { \ctex_ltj_set_kanjiskip:N \l__ctex_ccglue_skip }
⟨/luatex⟩
⟨*uptex|aptex⟩
  { \skip_set_eq:NN \tex_kanjiskip:D \l__ctex_ccglue_skip }
⟨/uptex|aptex⟩
\skip_new:N \l__ctex_ccglue_skip
```

\ctex_if_ccglue_touched_p:
\ctex_if_ccglue_touched:*TF*

检查用户是否修改过汉字间距。

```
\prg_new_conditional:Npnn \ctex_if_ccglue_touched: { TF }
  {
⟨*pdftex|xetex⟩
    \if_meaning:w \CJKglue \__ctex_ccglue:
      \prg_return_false: \else: \prg_return_true: \fi:
⟨/pdftex|xetex⟩
⟨*luatex⟩
    \skip_if_eq:nnTF { \l__ctex_ccglue_skip } { \ltjgetparameter { kanjiskip } }
      { \prg_return_false: } { \prg_return_true: }
⟨/luatex⟩
⟨*uptex|aptex⟩
    \skip_if_eq:nnTF { \l__ctex_ccglue_skip } { \tex_kanjiskip:D }
      { \prg_return_false: } { \prg_return_true: }
⟨/uptex|aptex⟩
  }
```

注意下面的标记不能用 %<pdftex|xetex>，它会导致旧版本的 l3docstrip 不能替换 @@。

```
⟨*pdftex|xetex⟩
\ctex_at_end:n { \cs_new_eq:NN \__ctex_ccglue: \CJKglue }
⟨/pdftex|xetex⟩
```

\ctex_update_em_unit:

将当前汉字的宽度保存到 \ccwd 中备用。不采用 1em，因为这时的 1em 实际上来自西文字体

的信息，未必等于汉字的宽度，这似乎在传统的 .tfm 字体上表现更明显。在 pdfTEX 和 XETEX 下，直接使用 \f@size\p@ 作为汉字的宽度，这应该对大多数汉字字体都成立，但不适用于诸如“方正兰亭黑长”之类的特殊字体。在 XETEX 可以用 \fontcharwd 来改进。而在 pdfTEX 下，若使用 zhmetrics 技术，所有的汉字共享同一个 .tfm，\fontcharwd 也就没有意义。在 LuaTEX 下，LuaTeX-ja 总是按照 JFM 中的设置输出汉字的宽度，可以直接用 \zw 作为汉字宽度。upTEX 可以直接使用原生的长度单位 zw。

```
\cs_new_protected_nopar:Npn \ctex_update_em_unit:
  { \dim_set:Nn \ccwd { \f@size \p@ } }
  { \dim_set:Nn \ccwd { \zw } }
  { \dim_set:Nn \ccwd { 1zw } }
```

⟨pdftex|xetex⟩ ⟨luatex⟩ ⟨uptex|aptex⟩

14.3.10　其它

\ctex_add_to_selectfont:n
\CTEX@selectfont@hook

\EverySelectfont 直到文档开始时才有效。为了 \ccwd 和 LuaTeX-ja 的字体设置在导言区也可用，我们还需要在这里手工修改 \selectfont。everysel 宏包会用 \CheckCommand 来检查 \selectfont 是否为标准定义。我们修改了 \selectfont，所以会给出一个警告。为了消除这个警告，在它检查之前，还原本来定义。pxeverysel 宏包取消了检查，但也需要恢复定义，避免重复使用钩子。

```
\cs_new_protected:Npn \ctex_add_to_selectfont:n #1
  {
    \cs_set_protected_nopar:Npx \CTEX@selectfont@hook
      { \exp_not:o { \CTEX@selectfont@hook #1 } }
  }
\cs_new_eq:NN \CTEX@selectfont@hook \prg_do_nothing:
\if_cs_exist:N \@EverySelectfont@Init
  \group_begin:
    \cs_set:Npn \__ctex_tmp:N #1
      {
        \tl_set:Nn \l__ctex_tmp_tl {#1}
        \cs_new_eq:NN \CTEX@selectfont@save #1
        \cs_new_protected_nopar:Npn \__ctex_restore_selectfont:
          {
            \tl_put_left:Nn \@EverySelectfont@Init
              { \let #1 \CTEX@selectfont@save }
            \cs_undefine:N \__ctex_restore_selectfont:
          }
      }
    \ctex_parse_name:NN \__ctex_tmp:N \selectfont
  \exp_last_unbraced:NNo \group_end:
  \ctex_patch_cmd_once:NnnnTF { \l__ctex_tmp_tl }
    { \ExplSyntaxOff }
    { \size@update }
    { \CTEX@selectfont@hook \size@update }
    { \__ctex_restore_selectfont: }
    { \ctex_patch_failure:N \selectfont }
\fi:
```

\CJK@plane 有定义，说明处于 CJK 宏包的 \CJKsymbol 之内，不必使用钩子。

```
⟨*pdftex⟩
\EverySelectfont { \cs_if_exist:NF \CJK@plane { \CTEX@selectfont@hook } }
⟨/pdftex⟩
⟨*xetex|luatex|uptex|aptex⟩
\EverySelectfont { \CTEX@selectfont@hook }
⟨/xetex|luatex|uptex|aptex⟩
```

Attribute 寄存器 \ltj@curjfnt 的初始值是 −1，必须把它设置为一个有效的 font.id，否则编译时会直接退出。

```
⟨*luatex⟩
\ctex_add_to_selectfont:n
```

```
  {
    \ctex_ltj_select_font:
    \ctex_ltj_select_alternate_font:
  }
\tl_set:Nn \CJK@family { song } \selectfont
\tl_clear:N \CJK@family
</luatex>
```

\ctex_update_xkanjiskip:
\l__ctex_xkanjiskip_skip

upTEX 和 LuaTeX-ja 对 \xkanjiskip 都是即时赋值。单位 zw 与字体相关，因此需要每次 \selectfont 的时候更新一次 \xkanjiskip。如果用户设置过 \xkanjiskip，就不更新。注意，同 TEX 的 \baselineskip 一样，如果在一个段落内多次设置了 \kanjiskip 或 \xkanjiskip，只有最后的设置会影响全段。

```
<*luatex|uptex|aptex>
\cs_new_protected_nopar:Npn \ctex_update_xkanjiskip:
  {
    \skip_if_eq:nnT
      { \ltjgetparameter { xkanjiskip } } { \l__ctex_xkanjiskip_skip }
      { \tex_xkanjiskip:D } { \l__ctex_xkanjiskip_skip }
      {
        \skip_set:Nn \l__ctex_xkanjiskip_skip { \l__ctex_xkanjiskip_tl }
        \ctex_ltj_set_xkanjiskip:N \l__ctex_xkanjiskip_skip
        \skip_set_eq:NN \tex_xkanjiskip:D \l__ctex_xkanjiskip_skip
      }
  }
\tl_new:N \l__ctex_xkanjiskip_tl
\tl_set:Nn \l__ctex_xkanjiskip_tl
  { .25\zw plus 1pt minus 1pt }
  { .25zw plus 1pt minus 1pt }
\skip_new:N \l__ctex_xkanjiskip_skip
\skip_set:Nn \l__ctex_xkanjiskip_skip
  { \ltjgetparameter { xkanjiskip } }
  { \tex_xkanjiskip:D }

\ctex_add_to_selectfont:n { \ctex_update_xkanjiskip: }
</luatex|uptex|aptex>
```

\cht
\cdp
\cwd

分别从 .jfm 中读取字符高度、深度和宽度，目前仅考虑横排的情况。

```
<*luatex>
\dim_new:N \cht
\dim_new:N \cdp
\dim_new:N \cwd
\newluafunction \g__ctex_kanjisize_func
\group_begin:
\char_set_catcode_space:n { 32 }
\lua_now:e
  {
    local nulltable = { }
    local t = lua.get_functions_table()
    local fmt = luatexja.jfont.font_metric_table
    local getattribute = tex.getattribute
    local setdimen = tex.setdimen
    t[\int_use:N \g__ctex_kanjisize_func] = function ()
      local ft = fmt[getattribute('ltj@curjfnt')] or nulltable
      local ft = ft and ft.char_type or nulltable
      local fk = ft and ft[0] or nulltable
      setdimen('cht', fk.height or 0)
      setdimen('cdp', fk.depth or 0)
      setdimen('cwd', fk.width or ft.zw or 0)
    end
  }
\group_end:
\cs_new_protected_nopar:Npn \ctex_update_kanjisize:
 { \tex_luafunction:D \g__ctex_kanjisize_func }
```

```
\ctex_add_to_selectfont:n { \ctex_update_kanjisize: }
⟨/luatex⟩
```

space　在导言区或正文中设置忽略空格方式。pdfTEX 和 XETEX 下初始设置为 auto，LuaTEX、upTEX 下是无效选项。

```
\keys_define:nn { ctex }
  {
⟨*pdftex|xetex⟩
    space .choice: ,
    space / true  .code:n =
⟨pdftex⟩      { \ctex_ignorespaces_case:N \prg_do_nothing: } ,
⟨xetex⟩       { \xeCJKsetup { CJKspace = true } } ,
    space / auto  .code:n =
⟨pdftex⟩      { \ctex_ignorespaces_case:N \ctex_auto_ignorespaces: } ,
⟨xetex⟩       { \xeCJKsetup { CJKspace = false } } ,
    space / false .code:n =
⟨pdftex⟩      { \ctex_ignorespaces_case:N \tex_ignorespaces:D } ,
⟨xetex⟩       { \xeCJKsetup { CJKspace = false } } ,
    space .default:n = { true } ,
    space .initial:n = { auto }
⟨/pdftex|xetex⟩
⟨*luatex|uptex|aptex⟩
    space .code:n =
      { \msg_warning:nn { ctex } { invalid-option } }
⟨/luatex|uptex|aptex⟩
  }
```

punct　在导言区或正文中设置标点符号输出格式。LuaTeX-ja 设置的是字体的默认 JFM，只会影响到之后设置的字体。upTEX 暂时无效。

```
\keys_define:nn { ctex }
  {
    punct .code:n =
      {
        \tl_set:Nx \l__ctex_punct_tl { #1 }
⟨pdftex⟩        \punctstyle { \l__ctex_punct_tl }
⟨xetex⟩         \xeCJKsetup { PunctStyle = \l__ctex_punct_tl }
⟨luatex⟩        \ctex_mono_jfm:o { \l__ctex_punct_tl }
⟨uptex|aptex⟩   \msg_warning:nn { ctex } { invalid-option }
      } ,
    punct .default:n = { quanjiao } ,
  }
```

XELATEX、LuaLATEX 和 upLATEX 总是使用 UTF8 编码。

```
⟨*xetex|luatex|uptex|aptex⟩
\tl_set:Nn \l__ctex_encoding_tl { UTF8 }
⟨/xetex|luatex|uptex|aptex⟩
```

```
⟨/pdftex|xetex|luatex|uptex|aptex⟩
```

14.3.11　载入引擎定义文件

最后载入各个编译引擎的定义文件。

```
⟨class|ctex⟩ \ctex_file_input:n { \c__ctex_engine_file_str }
```

14.4　用户设置接口

\ctexset

```
⟨*class|ctex|ctexheading⟩
\NewDocumentCommand \ctexset { } { \keys_set:nn { ctex } }
⟨/class|ctex|ctexheading⟩
```

\CTEXsetup
\CTEXoptions

过时命令。出于历史原因，\CTEXoptions 需要在 pxeverysel 宏包之后定义。

```
⟨*class|ctex⟩
\NewDocumentCommand \CTEXsetup { +o > { \TrimSpaces } m }
  {
    \msg_warning:nnnn { ctex } { deprecated-command } { \CTEXsetup }
      { \ctexset~ {~ #2~ =~ {~ #1~ }~ }~ is~ set. }
    \IfNoValueF {#1} { \keys_set:nn { ctex / #2 } {#1} }
  }
\NewDocumentCommand \CTEXoptions { +o }
  {
    \msg_warning:nnnn { ctex } { deprecated-command } { \CTEXoptions }
      { \ctexset~ {~ #1~ }~ is~ set. }
    \IfNoValueF {#1} { \keys_set:nn { ctex } {#1} }
  }
```

14.5　字距与缩进

autoindent

autoindent 也是可以用在正文中的选项，意义与宏包选项 option/autoindent 相同。

```
\keys_define:nn { ctex }
  {
    autoindent .choice: ,
    autoindent .default:n = { true } ,
    autoindent / true    .code:n =
      {
        \tl_set:Nn \l__ctex_autoindent_tl { 2 \ccwd }
        \ctex_select_size:
      } ,
    autoindent / false   .code:n =
      { \tl_clear:N \l__ctex_autoindent_tl } ,
    autoindent / unknown .code:n =
      {
        \ctex_set_default_ccwd:Nn \l__ctex_autoindent_tl {#1}
        \ctex_select_size:
      }
  }
```

\CTEXsetfont

无论字体大小是否变化都更新相关信息。

```
\NewDocumentCommand \CTEXsetfont { } { \ctex_select_size: }
\cs_new_protected_nopar:Npn \ctex_select_size:
  { \cs_if_free:NTF \size@update { \ctex_update_size: } { \selectfont } }
```

\ctex_update_size:

在字号变化时更新 \ccwd、\parindent 和汉字间距。字距为零则恢复正常设置。

```
\cs_new_protected_nopar:Npn \ctex_update_size:
  {
    \tl_if_eq:NNTF \l__ctex_ziju_tl \c__ctex_zero_tl
      {
        \ctex_update_stretch:
        \ctex_update_parindent:
      }
      { \ctex_update_ziju: }
  }
\tl_const:Nx \c__ctex_zero_tl { \fp_use:N \c_zero_fp }
\tl_new:N \l__ctex_ziju_tl
\tl_set_eq:NN \l__ctex_ziju_tl \c__ctex_zero_tl
```

在 \selectfont 中，若 \size@update 为 \relax，说明字体大小没有变化，我们也就不用更新相关参数。

```
\ctex_add_to_selectfont:n
  { \cs_if_free:NF \size@update { \ctex_update_size: } }
```

linestretch　若行宽不是汉字宽度的整数倍，自然要求伸展它们之间的差。这里设置的是在此基础上的额外伸展量。初始化为一个汉字的宽度。若设置为 \maxdimen，则禁用此功能。参数的默认单位是汉字的宽度 \ccwd。

```
\keys_define:nn { ctex }
  {
    linestretch .code:n =
      {
        \ctex_set_default_ccwd:Nn \l__ctex_line_stretch_tl {#1}
        \ctex_select_size:
      } ,
    linestretch .value_required:n = true
  }
\tl_new:N \l__ctex_line_stretch_tl
\tl_set:Nn \l__ctex_line_stretch_tl { \ccwd }
```

\ctex_update_stretch:　首先计算一行上汉字的字数，\CJKglue 相当于将 \linewidth 与汉字总宽度之差均匀地填充到汉字之间。ε-TEX 的除法是四舍五入，而我们这里应该用截断。由于没有可展性的要求，直接用原语 \tex_divide:D 要比 \int_div_truncate:nn 快一些。下面的算法还兼顾到了 \linewidth 不为汉字字宽的整数倍的情况。若用户禁用 linestretch 并且修改过 \CJKglue，则只更新 \ccwd，否则设置伸展量为 0.08 倍 \baselineskip。注意 everysel 的钩子位于 \size@update 之前，\baselineskip 还未更新，不能直接使用它。

```
\cs_new_protected_nopar:Npn \ctex_update_stretch:
  {
    \ctex_update_em_unit:
    \dim_set:Nn \l__ctex_tmp_dim { \l__ctex_line_stretch_tl }
    \dim_compare:nNnTF \l__ctex_tmp_dim = \c_max_dim
      {
        \ctex_if_ccglue_touched:TF
          { \ctex_update_ccwd: }
          {
            \dim_set:Nn \l__ctex_tmp_dim
              { \baselinestretch \tex_glueexpr:D \f@baselineskip \scan_stop: }
            \skip_set:Nn \l__ctex_ccglue_skip
              { \c_zero_dim plus .08 \l__ctex_tmp_dim }
            \ctex_update_ccglue:
          }
      }
      {
        \int_set:Nn \l__ctex_tmp_int
          { \tex_dimexpr:D \linewidth - \ccwd - \l__ctex_tmp_dim \scan_stop: }
        \tex_divide:D \l__ctex_tmp_int \ccwd
        \int_compare:nNnTF \l__ctex_tmp_int > \c_zero_int
          {
            \skip_set:Nn \l__ctex_ccglue_skip
              {
                \c_zero_dim plus \dim_eval:n
                  {
                    ( \linewidth - \ccwd - \l__ctex_tmp_int \ccwd ) /
                    \l__ctex_tmp_int
                  }
              }
          }
          { \skip_zero:N \l__ctex_ccglue_skip }
        \ctex_update_ccglue:
      }
  }
```

\ctex_update_parindent:　更新段落首行缩进。此函数在字号变化时调用。

```
\cs_new_protected_nopar:Npn \ctex_update_parindent:
  {
    \tl_if_empty:NF \l__ctex_autoindent_tl
      {
```

```
        \dim_compare:nNnF \parindent = \c_zero_dim
          { \dim_set:Nn \parindent { \l__ctex_autoindent_tl } }
      }
  }
```

\ziju　若参数为 0，则恢复正常间距。

```
\NewDocumentCommand \ziju { m }
  { \exp_args:Nx \ctex_ziju:n {#1} \tex_ignorespaces:D }
\cs_new_protected_nopar:Npn \ctex_ziju:n #1
  {
    \tl_set:Nx \l__ctex_ziju_tl { \fp_eval:n {#1} }
    \ctex_select_size:
  }
```

\ctex_update_ziju:　更新字距。若字距不大于 -1，即 \ccwd 为非正值，则不计算伸缩值。否则，首先假定汉字的宽度为正常宽度加上字距，看一行上能正常放下多少个汉字。

```
\cs_new_protected_nopar:Npn \ctex_update_ziju:
  {
    \ctex_update_em_unit:
    \dim_set:Nn \l__ctex_ziju_dim { \l__ctex_ziju_tl \ccwd }
    \dim_add:Nn \ccwd { \l__ctex_ziju_dim }
    \dim_compare:nNnTF \ccwd > \c_zero_dim
```

伸展量保证行内的剩余空白能够被均匀地填充到汉字之间，收缩的最大限度是让当前行还能够再挤下一个汉字并且不会出现负间距。由 TEX 决定伸展还是收缩。

```
      {
        \dim_set:Nn \l__ctex_tmp_dim
          { \linewidth - \ccwd + \l__ctex_ziju_dim }
        \int_set:Nn \l__ctex_tmp_int { \l__ctex_tmp_dim }
        \tex_divide:D \l__ctex_tmp_int \ccwd
        \dim_sub:Nn \l__ctex_tmp_dim { \l__ctex_tmp_int \ccwd }
```

由于 \parindent 是一个固定值，并不参与伸缩，容易导致第一行出现坏盒子。我们在这里将字数减去 2，以此放大伸缩值。

```
        \dim_compare:nNnF \parindent = \c_zero_dim
          {
            \int_compare:nNnF \l__ctex_tmp_int < 3
              { \int_sub:Nn \l__ctex_tmp_int { 2 } }
          }
        \skip_set:Nn \l__ctex_ccglue_skip
          {
            \l__ctex_ziju_dim
            plus  \dim_eval:n { \l__ctex_tmp_dim / \l__ctex_tmp_int }
            minus \dim_min:nn { \dim_abs:n { \l__ctex_ziju_dim } }
              { ( \ccwd - \l__ctex_tmp_dim ) / ( \l__ctex_tmp_int + 1 ) }
          }
      }
      { \skip_set:Nn \l__ctex_ccglue_skip { \l__ctex_ziju_dim } }
    \ctex_update_ccglue:
```

字距设置得比较大时，为了尽量保证段首缩进能够与下一行对齐，应该需要相应地加上或者减去伸缩值。但是这里并不清楚 TEX 是伸展还是收缩，之前以“当前行是否还放得下一个汉字”为标准加上或减去伸缩值的做法也未必与实际结果一致，所以只好还是设置为 2\ccwd。

```
    \ctex_update_parindent:
  }
\dim_new:N \l__ctex_ziju_dim
```

\CTEXindent
\CTEXnoindent　过时命令。

```
\NewDocumentCommand \CTEXindent { }
```

```
  {
    \msg_warning:nnnn { ctex } { deprecated-command } { \CTEXindent }
      { \parindent is~ set~ to~ 2\ccwd. }
    \ctex_update_ccwd: \dim_set:Nn \parindent { 2 \ccwd }
  }
\NewDocumentCommand \CTEXnoindent { }
  {
    \msg_warning:nnnn { ctex } { deprecated-command } { \CTEXnoindent }
      { \parindent is~ set~ to~ 0pt. }
    \dim_zero:N \parindent
  }
```

14.6　中文数字与日期

```
\PassOptionsToPackage { encoding = \l__ctex_encoding_tl } { zhnumber }
\RequirePackage { zhnumber }
```

\chinese

```
\cs_new_nopar:Npn \chinese { \zhnum_counter:n }
\cs_new_eq:NN \@chinese \@zhnum
\cs_new_eq:NN \Chinese \chinese
\cs_new_eq:NN \CTEXcounter \use_none:n
```

给 enumitem 宏包注册 \chinese、\zhnum 和 \zhdig。

```
\ctex_at_end_package:nn { enumitem }
  {
    \cs_if_free:NF \AddEnumerateCounter
      {
        \AddEnumerateCounter * { \zhnum }   { \@zhnum } { 1 }
        \AddEnumerateCounter * { \zhdig }   { \@zhdig } { 1 }
        \AddEnumerateCounter * { \chinese } { \@chinese } { 1 }
      }
  }
```

\CTEXnumber
\CTEXdigits

```
\NewDocumentCommand \CTEXnumber { m m }
  { \protected@edef #1 { \zhnumber {#2} } }
\NewDocumentCommand \CTEXdigits { m m }
  { \protected@edef #1 { \zhdigits {#2} } }
```

today

```
\cs_set_eq:NN \CTEX@todayold \today
\keys_define:nn { ctex }
  {
    today .choice: ,
    today / old     .code:n =
      { \cs_set_eq:NN \today \CTEX@todayold } ,
    today / small   .code:n =
      {
        \cs_set_eq:NN \today \zhtoday
        \zhnumsetup { time = Arabic }
      } ,
    today / big     .code:n =
      {
        \cs_set_eq:NN \today \zhtoday
        \zhnumsetup { time = Chinese }
      } ,
    today / unknown .code:n =
      { \msg_error:nnx { ctex } { today-undef } {#1} }
  }
\msg_new:nnnn { ctex } { today-undef }
  { Today~format~`#1'~is~undefined. }
  { Available~today~formats~are~`old',~`small',~and~`big'. }
```

14.7　其它中文标题定义

\proofname　\proofname 未在标准文档类中定义，需要确保它非空。

```
\tl_if_exist:NF \proofname
```

```
  {
    \tl_new:N \proofname
    \tl_set:Nn \proofname { Proof }
  }

\keys_define:nn { ctex }
  {
    contentsname   .tl_set:N = \contentsname ,
    listfigurename .tl_set:N = \listfigurename ,
    listtablename  .tl_set:N = \listtablename ,
    figurename     .tl_set:N = \figurename ,
    tablename      .tl_set:N = \tablename ,
    abstractname   .tl_set:N = \abstractname ,
    indexname      .tl_set:N = \indexname ,
    appendixname   .tl_set:N = \appendixname ,
    proofname      .tl_set:N = \proofname ,
⟨article⟩   bibname        .tl_set:N = \refname
⟨book|report⟩   bibname        .tl_set:N = \bibname
⟨*beamer⟩
    algorithmname  .tl_set:N = \algorithmname ,
    bibname        .tl_set:N = \bibname ,
    refname        .tl_set:N = \refname ,
    continuation   .tl_set:N = \insertcontinuationtext
⟨/beamer⟩
  }

⟨*ctex⟩
\msg_new:nnn { ctex } { ctexbibname }
  {
    Neither~`\token_to_str:N \bibname'~nor~`\token_to_str:N \refname'~can~be~found.\\
    The~key~`bibname'~will~set~`\token_to_str:N \ctexbibname'~to~the~given~value.
  }
\tl_if_exist:NTF \insertcontinuationtext
  {
    \keys_define:nn { ctex }
      {
        algorithmname .tl_set:N = \algorithmname ,
        bibname       .tl_set:N = \bibname ,
        refname       .tl_set:N = \refname ,
        continuation  .tl_set:N = \insertcontinuationtext
      }
  }
  {
    \tl_if_exist:NTF \bibname
      { \keys_define:nn { ctex } { bibname .tl_set:N = \bibname } }
      {
        \tl_if_exist:NTF \refname
          { \keys_define:nn { ctex } { bibname .tl_set:N = \refname } }
          {
            \msg_warning:nn { ctex } { ctexbibname }
            \keys_define:nn { ctex } { bibname .tl_set:N = \ctexbibname }
          }
      }
  }
⟨/ctex⟩

⟨/class|ctex⟩
```

14.8　中文化的标题结构

本节内容在 CTEX 文档类或打开 heading 选项下生效。

```
⟨*class|heading⟩
```

14.8.1　定义标题格式选项

\c__ctex_section_headings_seq　保存 \section 级以下标题名字。

```
⟨*article|book|report⟩
\seq_const_from_clist:Nn \c__ctex_section_headings_seq
  { section , subsection , subsubsection , paragraph , subparagraph }
⟨/article|book|report⟩
```

\c__ctex_headings_seq

```
⟨*article|book|report⟩
\seq_new:N \c__ctex_headings_seq
\seq_gset_eq:NN \c__ctex_headings_seq \c__ctex_section_headings_seq
⟨book|report⟩ \seq_gput_left:Nn \c__ctex_headings_seq { chapter }
\seq_gput_left:Nn \c__ctex_headings_seq { part }
⟨/article|book|report⟩
⟨*beamer⟩
\seq_const_from_clist:Nn \c__ctex_headings_seq
  { part , section , subsection }
⟨/beamer⟩
```

__ctex_initial_heading:n

```
\cs_new_protected_nopar:Npn \__ctex_initial_heading:n #1
  {
    \tl_new:c { CTEX@pre#1 }
    \tl_new:c { CTEX@post#1 }
    \tl_const:cx { CTEXthe#1 }
      {
        \exp_not:c { CTEX@pre#1 }
        \exp_not:c { CTEX@the#1 }
        \exp_not:c { CTEX@post#1 }
      }
    \tl_const:cx { CTEX@#1name }
      {
        \group_begin:
          \exp_not:c { CTEX@#1@nameformat }
            {
              \exp_not:c { CTEX@pre#1 }
              \exp_not:N \tl_if_empty:NTF
              \exp_not:c { CTEX@#1@numberformat }
                { \exp_not:c { CTEX@the#1 } }
                {
                  \group_begin:
                    \exp_not:c { CTEX@#1@numberformat }
                    \exp_not:c { CTEX@the#1 }
                  \group_end:
                }
              \exp_not:c { CTEX@post#1 }
            }
        \group_end:
      }
  }
```

__ctex_def_heading_keys:n

```
\cs_new_protected_nopar:Npn \__ctex_def_heading_keys:n #1
  {
    \tl_put_right:Nx \l__ctex_tmp_tl
      {
        #1                    .meta:nn = { ctex / #1 } { ####1 } ,
        #1 / name              .code:n =
          { \ctex_assign_heading_name:nn {#1} { ####1 } } ,
        #1 / number         .tl_set:N = \exp_not:c { CTEX@the#1 } ,
        #1 / beforeskip     .tl_set:N = \exp_not:c { CTEX@#1@beforeskip } ,
        #1 / afterskip      .tl_set:N = \exp_not:c { CTEX@#1@afterskip} ,
        #1 / indent         .tl_set:N = \exp_not:c { CTEX@#1@indent } ,
        #1 / numbering    .bool_set:N = \exp_not:c { CTEX@#1@numbering } ,
        #1 / numbering     .initial:n = true ,
        #1 / beforeskip    .initial:n = \c_zero_skip ,
```

```
        #1 / afterskip    .initial:n = \c_zero_skip ,
        #1 / indent       .initial:n = \c_zero_dim ,
        #1 / beforeskip   .value_required:n = true ,
        #1 / afterskip    .value_required:n = true ,
        #1 / indent       .value_required:n = true ,
⟨*article|book|report⟩
        #1 / afterindent .bool_set:N = \exp_not:c { CTEX@#1@afterindent } ,
        #1 / fixskip     .bool_set:N = \exp_not:c { CTEX@#1@fixskip } ,
        #1 / hang        .bool_set:N = \exp_not:c { CTEX@#1@hang } ,
        #1 / hang         .initial:n = true ,
        #1 / runin       .bool_set:N = \exp_not:c { CTEX@#1@runin } ,
        #1 / tocline         .code:n =
          {
            \cs_set:Npn \exp_not:c { CTEX@#1@tocline}
                        \exp_not:n { ####1####2 } { ####1 }
          } ,
        \__ctex_plus_key_aux:nn {#1} { break } ,
⟨/article|book|report⟩
        \__ctex_plus_key_aux:nn {#1} { format } ,
        \__ctex_plus_key_aux:nn {#1} { nameformat } ,
        \__ctex_plus_key_aux:nn {#1} { numberformat } ,
        \__ctex_plus_key_aux:nn {#1} { titleformat } ,
        \__ctex_plus_key_aux:nn {#1} { aftername } ,
        \__ctex_plus_key_aux:nn {#1} { aftertitle } ,
      }
  }
\cs_new_nopar:Npn \__ctex_plus_key_aux:nn #1#2
  {
    #1 / #2   .tl_set:N = \exp_not:c { CTEX@#1@#2 } ,
    #1 / #2 +   .code:n =
      { \tl_put_right:Nn \exp_not:c { CTEX@#1@#2 } { ####1 } } ,
    #1 / #2 ~ + .code:n =
      { \tl_put_right:Nn \exp_not:c { CTEX@#1@#2 } { ####1 } }
  }
```

\ctex_assign_heading_name:nn
__ctex_assign_heading_name:nnn

name 的值是一个至多两个元素的逗号分隔列表。由于 LaTeX3 的 clist 总是会自动忽略空元素，所以设置 name={, 章} 后，第一个元素将会是“章”，必须用空的分组保护空元素：name={{}, 章}，这在使用中有些许不便。我们可以改用 seq 或者手写函数解析参数来加以改进。为实现的简单起见，这里用了 xparse 的 \SplitArgument，它带有参数的长度检查。

```
\NewDocumentCommand \ctex_assign_heading_name:nn
  { m > { \SplitArgument { 1 } { , } } +m }
  { \__ctex_assign_heading_name:nnn {#1} #2 }
\cs_new_protected:Npn \__ctex_assign_heading_name:nnn #1#2#3
  {
    \tl_set:cn { CTEX@pre#1 } {#2}
    \IfNoValueTF {#3}
      { \tl_clear:c { CTEX@post#1 } }
      { \tl_set:cn { CTEX@post#1 } {#3} }
  }
```

part/pagestyle
chapter/pagestyle
chapter/lofskip
chapter/lotskip

只在 ctexbook 和 ctexrep 下有定义。

```
\group_begin:
⟨*book|report⟩
\tl_set:Nn \l__ctex_tmp_tl
  {
    part    / pagestyle .tl_set:N = \CTEX@part@pagestyle ,
    chapter / pagestyle .tl_set:N = \CTEX@chapter@pagestyle ,
    chapter / lofskip   .tl_set:N = \CTEX@chapter@lofskip ,
    chapter / lotskip   .tl_set:N = \CTEX@chapter@lotskip ,
    chapter / lofskip  .initial:n = \c_zero_skip ,
    chapter / lotskip  .initial:n = \c_zero_skip ,
    chapter / lofskip  .value_required:n = true ,
    chapter / lotskip  .value_required:n = true ,
```

```
  }
⟨/book|report⟩
⟨*article|beamer⟩
\tl_clear:N \l__ctex_tmp_tl
⟨/article|beamer⟩
```

定义标题键值选项。

```
\seq_map_inline:Nn \c__ctex_headings_seq
  {
    \__ctex_initial_heading:n {#1}
    \__ctex_def_heading_keys:n {#1}
  }
\use:x
  {
    \group_end:
    \keys_define:nn { ctex } { \exp_not:o { \l__ctex_tmp_tl } }
  }
⟨*article|book|report⟩
```

14.8.2　标准标题命令的修改

\CTEX@fixtopskip　修正 book 和 report 类的 \part 和 \chapter 标题之前的多余空行。

```
⟨*book|report⟩
\cs_new_protected_nopar:Npn \CTEX@fixtopskip
  {
    \CTEX@fixheadingskip
    \dim_compare:nNnF \tex_pagegoal:D < \c_max_dim
      { \skip_sub:Nn \l__ctex_heading_skip { \tex_topskip:D } }
  }
⟨/book|report⟩
```

\CTEX@fixheadingskip　抑制行间粘连，修正标题前后的多余间距。事实上，减掉 \parskip，有一定的风险。如果接下来的内容不会进入水平模式（例如在 format 选项中使用 \hrule 或者 \hbox），TEX 就不会加上 \parskip。这时候就需要用户把 \parskip 加到 beforeskip 或者 afterskip 作为修正。

```
\cs_new_protected_nopar:Npn \CTEX@fixheadingskip
  {
    \par
    \dim_set:Nn \tex_prevdepth:D { -1000pt }
    \skip_sub:Nn \l__ctex_heading_skip { \tex_parskip:D }
  }
\skip_new:N \l__ctex_heading_skip
\cs_new_protected_nopar:Npn \CTEX@setheadingskip
  { \skip_set:Nn \l__ctex_heading_skip }
\cs_new_eq:NN \CTEX@headingskip \l__ctex_heading_skip
```

\partmark　提供 \partmark。

```
\ProvideDocumentCommand \partmark { m }
  { \markboth { } { } }
```

\CTEXifname
\CTEX@ifnametrue
\CTEX@ifnamefalse　用于判断当前标题是否有编号。

```
\cs_new_eq:NN \CTEXifname \use_ii:nn
\cs_new_protected_nopar:Npn \CTEX@ifnametrue
  { \cs_set_eq:NN \CTEXifname \use_i:nn }
\cs_new_protected_nopar:Npn \CTEX@ifnamefalse
  { \cs_set_eq:NN \CTEXifname \use_ii:nn }
```

\CTEX@addloflotskip　往插图和表格目录中加入额外间距。如果间距为零，则不加入。

```
⟨*book|report⟩
\cs_new_protected_nopar:Npn \CTEX@addloflotskip #1
  {
    \skip_set:Nn \l__ctex_heading_skip { \use:c { CTEX@#1@lofskip } }
    \skip_if_eq:nnF { \l__ctex_heading_skip } { \c_zero_skip }
      {
        \addtocontents { lof }
          { \protect \addvspace { \skip_use:N \l__ctex_heading_skip } }
      }
    \skip_set:Nn \l__ctex_heading_skip { \use:c { CTEX@#1@lotskip } }
    \skip_if_eq:nnF { \l__ctex_heading_skip } { \c_zero_skip }
      {
        \addtocontents { lot }
          { \protect \addvspace { \skip_use:N \l__ctex_heading_skip } }
      }
  }
⟨/book|report⟩
```

\CTEX@addtocline

```
\cs_new_protected:Npn \CTEX@addtocline #1#2
  { \addcontentsline { toc } {#1} { \use:c { CTEX@#1@tocline } {#1} {#2} } }
```

\CTEX@disableautoindent　禁用自动调整首行缩进。

```
\cs_new_protected_nopar:Npn \CTEX@disableautoindent
  { \tl_clear:N \l__ctex_autoindent_tl }
```

14.8.2.1　part 的标题

```
⟨@@=⟩
```

\part

```
⟨*article⟩
\renewcommand\part{%
   \if@noskipsec \leavevmode \fi
   \par
   \CTEX@part@break
%  \addvspace{4ex}%
   \CTEX@setheadingskip \CTEX@part@beforeskip
   \ifodd \CTEX@part@fixskip \CTEX@fixheadingskip \fi
   \addvspace \CTEX@headingskip
   \ifodd \CTEX@part@afterindent
     \@afterindenttrue
   \else
     \@afterindentfalse
   \fi
   \secdef\@part\@spart}
⟨/article⟩
⟨*book|report⟩
\renewcommand\part{%
% \if@openright
%   \cleardoublepage
% \else
%   \clearpage
% \fi
  \CTEX@part@break
% \thispagestyle{plain}%
  \thispagestyle{\CTEX@part@pagestyle}%
  \if@twocolumn
    \onecolumn
    \@tempswatrue
  \else
    \@tempswafalse
  \fi
% \null\vfil
  \CTEX@setheadingskip \CTEX@part@beforeskip
```

```
  \ifodd \CTEX@part@fixskip \CTEX@fixtopskip \fi
  \vspace*{\CTEX@headingskip}%
  \secdef\@part\@spart}
⟨/book|report⟩
```

\@part

```
⟨*article⟩
\def\@part[#1]#2{%
  \ifnum \c@secnumdepth >\m@ne
    \ifodd \CTEX@part@numbering
      \CTEX@ifnametrue
      \refstepcounter{part}%
%     \addcontentsline{toc}{part}{\thepart\hspace{1em}#1}%
    \else
      \CTEX@ifnamefalse
      \CTEX@makeanchor{part*}%
%     \addcontentsline{toc}{part}{#1}%
    \fi
  \else
    \CTEX@ifnamefalse
    \CTEX@makeanchor{part*}%
%   \addcontentsline{toc}{part}{#1}%
  \fi
  \CTEX@gettitle{#1}%
  \CTEX@addtocline{part}{#1}%
  {\interlinepenalty \@M
%  \normalfont \parindent \z@ \raggedright
   \CTEX@disableautoindent
   \normalfont \CTEX@part@format
%  \ifnum \c@secnumdepth >\m@ne
%    \Large\bfseries\partname\nobreakspace\thepart\par\nobreak
%  \fi
   \CTEX@hangindent{part}%
     {\CTEXifname{\CTEX@partname\CTEX@part@afternameame}{}}%
%  \huge\bfseries #2%
   \CTEX@part@titleformat{#2}%
%  \markboth{}{}%
   \partmark{#1}%
   \CTEX@part@aftertitle}%
  \nobreak
% \vskip 3ex
  \CTEX@setheadingskip \CTEX@part@afterskip
  \ifodd \CTEX@part@fixskip \CTEX@fixheadingskip \fi
  \vskip \CTEX@headingskip
  \@afterheading}
⟨/article⟩
⟨*book|report⟩
\def\@part[#1]#2{%
  \ifnum \c@secnumdepth >-2\relax
    \ifodd \CTEX@part@numbering
      \CTEX@ifnametrue
      \refstepcounter{part}%
%     \addcontentsline{toc}{part}{\thepart\hspace{1em}#1}%
    \else
      \CTEX@ifnamefalse
      \CTEX@makeanchor{part*}%
%     \addcontentsline{toc}{part}{#1}%
    \fi
  \else
    \CTEX@ifnamefalse
    \CTEX@makeanchor{part*}%
%   \addcontentsline{toc}{part}{#1}%
  \fi
  \CTEX@gettitle{#1}%
  \CTEX@addtocline{part}{#1}%
%  \markboth{}{}%
   \partmark{#1}%
  {\interlinepenalty \@M
```

```
%  \normalfont \centering
   \CTEX@disableautoindent
   \normalfont \CTEX@part@format
%  \ifnum \c@secnumdepth >-2\relax
%    \huge\bfseries\partname\nobreakspace\thepart\par\vskip 20\p@
%  \fi
   \CTEX@hangindent{part}%
     {\CTEXifname{\CTEX@partname\CTEX@part@aftername}{}}%
%  \Huge\bfseries #2\par}%
   \CTEX@part@titleformat{#2}%
   \CTEX@part@aftertitle}%
  \@endpart}
⟨/book|report⟩
```

\@spart

```
⟨*article⟩
\def\@spart#1{%
    \CTEX@ifnamefalse
    \CTEX@makeanchor@spart{part*}%
    \CTEX@gettitle{#1}%
    {\interlinepenalty \@M
%    \normalfont \parindent \z@ \raggedright
     \CTEX@disableautoindent
     \normalfont \CTEX@part@format
     \CTEX@hangindent{part}{}%
%    \huge \bfseries #1\par}%
     \CTEX@part@titleformat{#1}%
     \CTEX@part@aftertitle}%
     \nobreak
%    \vskip 3ex
     \CTEX@setheadingskip \CTEX@part@afterskip
     \ifodd \CTEX@part@fixskip \CTEX@fixheadingskip \fi
     \vskip \CTEX@headingskip
     \@afterheading}
⟨/article⟩
⟨*book|report⟩
\def\@spart#1{%
    \CTEX@ifnamefalse
    \CTEX@makeanchor@spart{part*}%
    \CTEX@gettitle{#1}%
    {\interlinepenalty \@M
%    \normalfont \centering
     \CTEX@disableautoindent
     \normalfont \CTEX@part@format
     \CTEX@hangindent{part}{}%
%    \Huge \bfseries #1\par}%
     \CTEX@part@titleformat{#1}%
     \CTEX@part@aftertitle}%
    \@endpart}
⟨/book|report⟩
```

\@endpart

```
⟨*book|report⟩
\def\@endpart{%
%              \vfil
               \CTEX@setheadingskip \CTEX@part@afterskip
               \ifodd \CTEX@part@fixskip \CTEX@fixheadingskip \fi
               \vskip \CTEX@headingskip
               \newpage
               \if@twoside
                \if@openright
                 \null
                 \thispagestyle{empty}%
                 \newpage
                \fi
               \fi
               \if@tempswa
                 \twocolumn
               \fi}
```

```
⟨/book|report⟩
```

14.8.2.2 chapter 的标题

```
⟨*book|report⟩
```

\chapter

```
\renewcommand\chapter{%
%                    \if@openright\cleardoublepage\else\clearpage\fi
%                    \thispagestyle{plain}%
                     \CTEX@chapter@break
                     \thispagestyle{\CTEX@chapter@pagestyle}%
                     \global\@topnum\z@
%                    \@afterindentfalse
                     \ifodd \CTEX@chapter@afterindent
                       \@afterindenttrue
                     \else
                       \@afterindentfalse
                     \fi
                     \secdef\@chapter\@schapter}
```

\@chapter

```
\def\@chapter[#1]#2{%
  \ifnum \c@secnumdepth >\m@ne
⟨*book⟩
    \if@mainmatter
⟨/book⟩
      \ifodd \CTEX@chapter@numbering
        \CTEX@ifnametrue
        \refstepcounter{chapter}%
%       \typeout{\@chapapp\space\thechapter.}%
        \typeout{\CTEXthechapter}%
%       \addcontentsline{toc}{chapter}
%         {\protect\numberline{\thechapter}#1}%
      \else
        \CTEX@ifnamefalse
        \CTEX@makeanchor{\Hy@chapapp*}%
%       \addcontentsline{toc}{chapter}{#1}%
      \fi
⟨*book⟩
    \else
      \CTEX@ifnamefalse
      \CTEX@makeanchor@chapter{\Hy@chapapp*}%
%     \addcontentsline{toc}{chapter}{#1}%
    \fi
⟨/book⟩
  \else
    \CTEX@ifnamefalse
    \CTEX@makeanchor@chapter{\Hy@chapapp*}%
%   \addcontentsline{toc}{chapter}{#1}%
  \fi
  \CTEX@gettitle{#1}%
  \CTEX@addtocline{chapter}{#1}%
  \chaptermark{#1}%
% \addtocontents{lof}{\protect\addvspace{10\p@}}%
% \addtocontents{lot}{\protect\addvspace{10\p@}}%
  \CTEX@addloflotskip{chapter}%
  \if@twocolumn
    \@topnewpage[\@makechapterhead{#2}]%
  \else
    \@makechapterhead{#2}%
  \@afterheading
  \fi}
```

\@schapter

```
\def\@schapter#1{%
  \CTEX@ifnamefalse
  \CTEX@makeanchor@schapter{\Hy@chapapp*}%
  \CTEX@gettitle{#1}%
  \if@twocolumn
```

```
    \@topnewpage[\@makeschapterhead{#1}]%
  \else
    \@makeschapterhead{#1}%
    \@afterheading
  \fi}
```

\@makechapterhead

```
\def\@makechapterhead#1{%
% \vspace*{50\p@}%
  \CTEX@setheadingskip \CTEX@chapter@beforeskip
  \ifodd \CTEX@chapter@fixskip \CTEX@fixtopskip \fi
  \vspace*{\CTEX@headingskip}%
% {\normalfont \parindent \z@ \raggedright
  {\CTEX@disableautoindent
   \normalfont \CTEX@chapter@format
   \interlinepenalty\@M
%  \ifnum \c@secnumdepth >\m@ne
%    \if@mainmatter
%      \huge\bfseries\@chapapp\space\thechapter\par\nobreak\vskip 20\p@
%    \fi
%  \fi
   \CTEX@hangindent{chapter}%
     {\CTEXifname{\CTEX@chaptername\CTEX@chapter@afternname}{}}%
%  \Huge \bfseries #1\par\nobreak
   \CTEX@chapter@titleformat{#1}%
   \CTEX@chapter@aftertitle
   \nobreak
%  \vskip 40\p@
   \CTEX@setheadingskip \CTEX@chapter@afterskip
   \ifodd \CTEX@chapter@fixskip \CTEX@fixheadingskip \fi
   \vskip \CTEX@headingskip
  }}
```

\@makeschapterhead

```
\def\@makeschapterhead#1{%
% \vspace*{50\p@}%
  \CTEX@setheadingskip \CTEX@chapter@beforeskip
  \ifodd \CTEX@chapter@fixskip \CTEX@fixtopskip \fi
  \vspace*{\CTEX@headingskip}%
% {\normalfont \parindent \z@ \raggedright
  {\CTEX@disableautoindent
   \normalfont \CTEX@chapter@format
   \interlinepenalty\@M
   \CTEX@hangindent{chapter}{}%
%  \Huge \bfseries  #1\par\nobreak
   \CTEX@chapter@titleformat{#1}%
   \CTEX@chapter@aftertitle
   \nobreak
%  \vskip 40\p@
   \CTEX@setheadingskip \CTEX@chapter@afterskip
   \ifodd \CTEX@chapter@fixskip \CTEX@fixheadingskip \fi
   \vskip \CTEX@headingskip
  }}
```

⟨/book|report⟩

14.8.2.3　section 类的标题

\@startsection

```
\def\@startsection#1#2#3#4#5#6{%
  \if@noskipsec \leavevmode \fi
  \par
% \@tempskipa #4\relax
% \@afterindenttrue
% \ifdim \@tempskipa <\z@
%   \@tempskipa -\@tempskipa \@afterindentfalse
% \fi
  \CTEX@update@sectionformat@n{#1}%
  \ifodd \CTEX@afterindent
    \@afterindenttrue
```

```
  \else
    \@afterindentfalse
  \fi
  \if@nobreak
    \everypar{}%
  \else
%   \addpenalty\@secpenalty\addvspace\@tempskipa
    \csname CTEX@#1@break\endcsname
    \CTEX@setheadingskip{#4}%
    \ifodd \CTEX@fixskip \CTEX@fixheadingskip \fi
    \addvspace \CTEX@headingskip
  \fi
  \@ifstar
    {\CTEX@makeanchor@ssect{#1*}\@ssect{#3}{#4}{#5}{#6}}%
    {\@dblarg{\@sect{#1}{#2}{#3}{#4}{#5}{#6}}}}
```

\@seccntformat

```
\def\@seccntformat#1{%
% \csname the#1\endcsname\quad}%
  \csname CTEX@#1name\endcsname
  \csname CTEX@#1@aftername\endcsname}
```

\@sect

```
\def\@sect#1#2#3#4#5#6[#7]#8{%
  \ifnum #2>\c@secnumdepth
    \CTEX@ifnamefalse
    \CTEX@makeanchor@sect{#1*}%
    \let\@svsec\@empty
  \else
    \ifodd \csname CTEX@#1@numbering\endcsname
      \CTEX@ifnametrue
      \refstepcounter{#1}%
      \protected@edef\@svsec{\@seccntformat{#1}\relax}%
    \else
      \CTEX@ifnamefalse
      \CTEX@makeanchor{#1*}%
      \let\@svsec\@empty
    \fi
  \fi
  \CTEX@gettitle{#7}%
% \@tempskipa #5\relax
% \ifdim \@tempskipa>\z@
  \unless \ifodd \CTEX@runin
    \begingroup
      #6{%
        \CTEX@hangfrom{\hskip\glueexpr #3\relax\@svsec}%
%       \interlinepenalty \@M #8\@@par}%
        \interlinepenalty \@M
        \csname CTEX@#1@titleformat\endcsname{#8}%
        \csname CTEX@#1@aftertitle\endcsname}%
    \endgroup
    \csname #1mark\endcsname{#7}%
%   \addcontentsline{toc}{#1}{%
%     \ifnum #2>\c@secnumdepth \else
%       \protect\numberline{\csname the#1\endcsname}%
%     \fi
%     #7}%
    \CTEX@addtocline{#1}{#7}%
  \else
    \def\@svsechd{%
    #6{\hskip\glueexpr #3\relax
%     \@svsec #8}%
      \@svsec
      \csname CTEX@#1@titleformat\endcsname{#8}%
      \csname CTEX@#1@aftertitle\endcsname}%
    \csname #1mark\endcsname{#7}%
%   \addcontentsline{toc}{#1}{%
%     \ifnum #2>\c@secnumdepth \else
%       \protect\numberline{\csname the#1\endcsname}%
```

```
%     \fi
%     #7}%
    \CTEX@addtocline{#1}{#7}}%
  \fi
  \@xsect{#5}}
```

\@ssect

```
\def\@ssect#1#2#3#4#5{%
  \CTEX@ifnamefalse
  \CTEX@gettitle{#5}%
% \@tempskipa #3\relax
% \ifdim \@tempskipa>\z@
  \unless \ifodd \CTEX@runin
    \begingroup
      #4{%
        \CTEX@hangfrom{\hskip\glueexpr #1\relax}%
%         \interlinepenalty \@M #5\@@par}%
          \interlinepenalty \@M
          \CTEX@titleformat@n{#5}%
          \CTEX@aftertitle}%
    \endgroup
  \else
%   \def\@svsechd{#4{\hskip\glueexpr #1\relax #5}}%
    \def\@svsechd{#4{\hskip\glueexpr #1\relax
                     \CTEX@titleformat@n{#5}\CTEX@aftertitle}}%
  \fi
  \@xsect{#3}}
```

\@xsect

```
\def\@xsect#1{%
% \@tempskipa #1\relax
% \ifdim \@tempskipa>\z@
  \unless \ifodd \CTEX@runin
    \par \nobreak
%   \vskip \@tempskipa
    \CTEX@setheadingskip{#1}%
    \ifodd \CTEX@fixskip \CTEX@fixheadingskip \fi
    \vskip \CTEX@headingskip
    \@afterheading
  \else
    \@nobreakfalse
    \global\@noskipsectrue
    \everypar{%
      \if@noskipsec
        \global\@noskipsecfalse
       {\setbox\z@\lastbox}%
        \clubpenalty\@M
        \begingroup \@svsechd \endgroup
        \unskip
%       \@tempskipa #1\relax
%       \hskip -\@tempskipa
        \hskip\glueexpr #1\relax
      \else
        \clubpenalty \@clubpenalty
        \everypar{}%
      \fi}%
  \fi
  \ignorespaces}
```

```
⟨@@=ctex⟩
```

\CTEX@hangindent　用于实现 \part 和 \chapter 标题的 indent 和 hang 选项。

```
\cs_new_protected:Npn \CTEX@hangindent #1#2
  {
    \dim_set:Nn \parindent { \use:c { CTEX@#1@indent } }
    \bool_if:cTF { CTEX@#1@hang }
      { \@hangfrom { \__ctex_indent_aux: #2 } }
      {#2}
  }
```

```
\cs_new_protected_nopar:Npn \__ctex_indent_aux:
  {
    \dim_compare:nNnF \parindent = \c_zero_dim
      { \skip_horizontal:n { \parindent } }
  }
```

\CTEX@hangfrom　hang 选项控制是否采用悬挂缩进。

```
\cs_new_protected_nopar:Npn \CTEX@hangfrom
  {
    \bool_if:NTF \CTEX@hang
      { \@hangfrom }
      { \noindent \use:n }
  }
```

\CTEX@update@sectionformat@n　在 \@startsection 中设置 \CTEX@titleformat@n 等为相应函数。

```
\cs_new_protected_nopar:Npn \CTEX@update@sectionformat@n #1
  {
    \cs_set_eq:Nc \CTEX@titleformat@n { CTEX@#1@titleformat }
    \cs_set_eq:Nc \CTEX@aftertitle    { CTEX@#1@aftertitle }
    \cs_set_eq:Nc \CTEX@afterindent   { CTEX@#1@afterindent }
    \cs_set_eq:Nc \CTEX@fixskip       { CTEX@#1@fixskip }
    \cs_set_eq:Nc \CTEX@hang          { CTEX@#1@hang }
    \cs_set_eq:Nc \CTEX@runin         { CTEX@#1@runin }
  }
\cs_new_eq:NN \CTEX@titleformat@n \use:n
\cs_new_eq:NN \CTEX@aftertitle \prg_do_nothing:
\cs_new_eq:NN \CTEX@afterindent \c_true_bool
\cs_new_eq:NN \CTEX@fixskip \c_false_bool
\cs_new_eq:NN \CTEX@hang \c_true_bool
\cs_new_eq:NN \CTEX@runin \c_false_bool
```

\CTEX@part@tocline
\CTEX@chapter@tocline

```
\cs_new:Npn \CTEX@part@tocline #1#2
  {
    \CTEXifname
      { \CTEXthepart \hspace { 1em } }
      { }
    #2
  }
⟨*book|report⟩
\cs_new:Npn \CTEX@chapter@tocline #1#2
  {
    \CTEXifname
      { \protect \numberline { \CTEXthechapter \hspace { .3em } } }
      { }
    #2
  }
⟨/book|report⟩
```

\CTEXnumberline

```
\cs_new_nopar:Npn \CTEXnumberline #1
  {
    \CTEXifname
      { \protect \numberline { \use:c { CTEXthe #1 } } }
      { }
  }

\int_zero:N \l__ctex_tmp_int
\seq_map_inline:Nn \c__ctex_section_headings_seq
  {
    \int_incr:N \l__ctex_tmp_int
    \cs_gset_protected_nopar:cpx  {#1}
      {
        \exp_not:N \@startsection {#1}
          { \int_use:N \l__ctex_tmp_int }
          { \exp_not:c { CTEX@#1@indent } }
          { \exp_not:c { CTEX@#1@beforeskip } }
```

```
          { \exp_not:c { CTEX@#1@afterskip } }
          { \exp_not:N \normalfont \exp_not:c { CTEX@#1@format } }
      }
    \cs_new:cpn { CTEX@#1@tocline } ##1##2
      { \CTEXnumberline { ##1 } ##2 }
  }
```

14.8.2.4　附录标题

appendix/name
appendix/number
appendix/numbering

```
\keys_define:nn { ctex }
  {
    appendix                .meta:nn = { ctex / appendix } {#1} ,
    appendix / name          .code:n =
      { \ctex_assign_heading_name:nn { appendix } {#1} } ,
    appendix / number      .tl_set:N = \CTEX@appendix@number ,
    appendix / numbering .bool_set:N = \CTEX@appendix@numbering ,
    appendix / numbering  .initial:n = true
  }
\tl_new:N \CTEX@preappendix
\tl_new:N \CTEX@postappendix
```

\appendix

```
\cs_new_eq:NN \CTEX@save@appendix \appendix
\cs_gset_protected_nopar:Npn \appendix
  {
    \CTEX@save@appendix
⟨*article⟩
    \gdef \CTEX@presection { \CTEX@preappendix }
    \gdef \CTEX@thesection { \CTEX@appendix@number }
    \gdef \CTEX@postsection { \CTEX@postappendix }
    \gdef \CTEX@section@numbering { \CTEX@appendix@numbering }
⟨/article⟩
⟨*book|report⟩
    \gdef \CTEX@prechapter { \CTEX@preappendix }
    \gdef \CTEX@thechapter { \CTEX@appendix@number }
    \gdef \CTEX@postchapter { \CTEX@postappendix }
    \gdef \CTEX@chapter@numbering { \CTEX@appendix@numbering }
⟨/book|report⟩
  }
```

14.8.2.5　设置 hyperref 宏包的标题锚点

\CTEX@makeanchor

设置超链接跳转锚点，在 hyperref 载入后才有意义。

```
\cs_new_protected_nopar:Npn \CTEX@makeanchor #1
  { }
```

\c__ctex_headings_cs_seq

保存内部标题命令的 CTEX 定义，用于随后比较。

```
\seq_const_from_clist:Nn \c__ctex_headings_cs_seq
⟨article⟩  { part , spart , sect , ssect }
⟨book|report⟩  { part , spart , chapter , schapter , sect , ssect }
\seq_map_inline:Nn \c__ctex_headings_cs_seq
  {
    \cs_new_eq:cc { CTEX@ #1 } { @ #1 }
    \cs_new_eq:cN { CTEX@makeanchor@ #1 } \CTEX@makeanchor
  }
```

\CTEX@hyperheadinghook

hyperref 会重定义内部标题命令，目的在于为没有编号的标题设置锚点（这一功能受他的 `implicit` 选项的控制）。我们在上面对标题命令的修改已经包含这一功能，如果这些标题命令在 hyperref 载入之前没有被修改过，则恢复 CTEX 的定义。

```
\cs_new_protected_nopar:Npn \CTEX@hyperheadinghook
  {
    \group_begin:
```

```
      \ifHy@implicit
        \cs_set_eq:NN \H@old@chapter \Hy@org@chapter
        \seq_map_inline:Nn \c__ctex_headings_cs_seq
          {
            \cs_if_eq:ccT { H@old@ ##1 } { CTEX@ ##1 }
              {
                \cs_gset_eq:cc { @ ##1 } { CTEX@ ##1 }
                \cs_gset_eq:cN { CTEX@makeanchor@ ##1 } \CTEX@makeanchor
              }
          }
      \else:
        \seq_map_inline:Nn \c__ctex_headings_cs_seq
          { \cs_gset_eq:cN { CTEX@makeanchor@ ##1 } \CTEX@makeanchor }
      \fi:
    \group_end:
  }
\ctex_at_end_package:nn { hyperref }
  {
    \cs_gset_protected_nopar:Npn \CTEX@makeanchor #1
      {
        \Hy@MakeCurrentHrefAuto {#1}
        \Hy@raisedlink
          {
            \hyper@anchorstart { \@currentHref }
            \hyper@anchorend
          }
      }
    \CTEX@hyperheadinghook
  }
```

14.8.2.6　兼容 nameref 宏包

\CTEX@gettitle　在 nameref 载入后才有意义，与上述 hyperref 的处理类似。

```
\cs_new_protected:Npn \CTEX@gettitle #1
  { }
\ctex_at_end_package:nn { nameref }
  {
    \cs_gset_protected_nopar:Npn \CTEX@gettitle { \NR@gettitle }
    \seq_map_inline:Nn \c__ctex_headings_cs_seq
      {
        \cs_if_eq:ccT { NR@ #1 } { CTEX@ #1 }
          { \cs_gset_eq:cc { @ #1 } { CTEX@ #1 } }
      }
  }
```

14.8.2.7　兼容 titlesec 宏包

我们修改了 `\@startsection` 的定义，它的第四个（⟨*beforeskip*⟩）和第五个（⟨*afterskip*⟩）参数的符号不再有特殊意义，改由相应的选项 `afterindent` 和 `runin` 来控制。

引入 titlesec 宏包，并且未设置它的 `loadonly` 选项时，titlesec 会展开 section 类标题获取它们的参数，进行初始设置。我们需要进行一些调整。

\ctex_titlesec_hook:　`\titleformat` 的设置保存在名为 `\ttlf@`⟨*section*⟩ 的宏中备用，它的内容是

`\ttlh@`⟨*shape*⟩{⟨*format*⟩}{⟨*label*⟩}{⟨*sep*⟩}{⟨*before*⟩}{⟨*after*⟩}

我们这里的 ⟨*shape*⟩ 为 `hang` 或者 `runin`。`\titlespacing` 的设置保存在 `\ttls@`⟨*section*⟩ 之中，它的内容是

{⟨*left*⟩}{⟨*right*⟩}{⟨*before*⟩}{⟨*after*⟩}{⟨*afterindent*⟩}

其中 ⟨*afterindent*⟩ 为 1 或 0，分别对应是否保留段首缩进。我们需要根据 CTEX 的 runin 和 afterindent 选项调整 \ttlh@⟨*shape*⟩ 和 ⟨*afterindent*⟩。注意，由 \ttl@extract 得的 ⟨*before*⟩ 和 ⟨*after*⟩ 的值总是非负的，而 CTEX 的 beforeskip 和 afterskip 是可以取负值的，但我们不打算调整它们了。如果使用了 titlesec 的 indentafter 等选项，也不需要调整 \ttls@⟨*section*⟩。

```
\cs_new_protected_nopar:Npn \ctex_titlesec_hook:
  {
    \@ifpackagewith { titlesec } { explicit }
      {
        \cs_set_eq:NN \__ctex_titlesec_format:Nn
                      \__ctex_titlesec_format_explicit:Nn
      }
      { }
    \clist_map_inline:nn
      { indentafter , noindentafter , indentfirst , nonindentfirst }
      {
        \@ifpackagewith { titlesec } { ##1 }
          {
            \clist_map_break:n
              { \cs_set_eq:NN \__ctex_titlesec_hook:n \__ctex_titlesec_format:n }
          }
          { }
      }
    \seq_map_function:NN \c__ctex_section_headings_seq \__ctex_titlesec_hook:n
  }
\cs_new_protected_nopar:Npn \__ctex_titlesec_hook:n #1
  {
    \__ctex_titlesec_format:n {#1}
    \exp_args:Nc \__ctex_titlesec_spacing:Nn { ttls@#1 } {#1}
  }
\cs_new_protected_nopar:Npn \__ctex_titlesec_format:n #1
  {
    \cs_if_free:cF { ttlf@#1 }
      { \exp_args:Nc \__ctex_titlesec_format:Nn { ttlf@#1 } {#1} }
  }
\cs_new_protected_nopar:Npn \__ctex_titlesec_format:Nn #1#2
  {
    \tl_set:Nx #1
      {
        \bool_if:cTF { CTEX@#2@runin }
          { \exp_not:N \ttlh@runin }
          { \exp_not:N \ttlh@hang }
        \tl_tail:N #1
      }
  }
\cs_new_protected_nopar:Npn \__ctex_titlesec_format_explicit:Nn #1#2
  {
    \cs_set_nopar:Npx #1 ##1
      {
        \bool_if:cTF { CTEX@#2@runin }
          { \exp_not:N \ttlh@runin }
          { \exp_not:N \ttlh@hang }
        \exp_args:No \tl_tail:n { #1 { } }
      }
  }
\cs_new_protected_nopar:Npn \__ctex_titlesec_spacing:Nn #1#2
  { \tl_set:Nx #1 { \exp_after:wN \__ctex_titlesec_spacing:nnnnnn #1 {#2} } }
\cs_new:Npn \__ctex_titlesec_spacing:nnnnnn #1#2#3#4#5#6
  {
    \exp_not:n { {#1} {#2} {#3} {#4} }
    { \bool_if:cTF { CTEX@#6@afterindent } { \@ne } { \z@ } }
  }

\@ifpackageloaded { titlesec }
  { }
```

```
{
  \ctex_at_end_package:nn { titlesec }
    {
      \@ifpackagewith { titlesec } { loadonly }
        { }
        { \ctex_titlesec_hook: }
    }
}
```

让编译时终端显示 \CTEXthechapter，目录使用 \CTEXtheXXX 编号。

```
\ctex_at_end_package:nn { titlesec }
  {
⟨*book|report⟩
    \tl_set:Nn \ttl@chapterout { \typeout { \CTEXthechapter } }
⟨/book|report⟩
    \cs_if_free:NF \ttl@tocpart
      {
        \cs_set_protected_nopar:Npn \ttl@tocpart
          { \tl_set:Nn \ttl@a { \CTEXthepart \hspace { 1em } } }
      }
    \seq_map_inline:Nn \c__ctex_headings_seq
      {
        \cs_if_exist:cF { ttl@toc #1 }
          {
            \cs_new_protected_nopar:cpx { ttl@toc #1 }
              {
                \tl_set:Nn \exp_not:N \ttl@a
                  {
                    \exp_not:N \protect
                    \exp_not:N \numberline { \exp_not:c { CTEXthe #1 } }
                  }
              }
          }
      }
  }
```

14.8.2.8　兼容 titleps 宏包

按照 titleps 宏包的实现机制，\CTEXtheXXX 等宏直到页眉排版时才会被展开，这可能会造成问题[20]。

\ctex_titleps_hook:

我们修改 titleps 包的内部命令 \ttl@settopmark 和 \ttl@setsubmark，将 \CTEXtheXXX 等加入更新队列中。

```
\group_begin:
\char_set_catcode_other:N \#
\cs_new_protected_nopar:Npn \ctex_titleps_hook:
  {
    \ctex_patch_cmd:Nnn \ttl@settopmark
      { \protect \@namedef { the#1 } { \@nameuse { the#1 } } }
      {
        \protect \@namedef { the#1 } { \@nameuse { the#1 } }
        \CTEX@titlepslabel@set {#1}
      }
    \ctex_patch_cmd:Nnn \ttl@setsubmark
      { \protect \@namedef { the#1 } { } }
      {
        \protect \@namedef { the#1 } { }
        \CTEX@titlepslabel@clear {#1}
      }
    \ctex_patch_cmd:Nnn \ttl@setsubmark
      { \protect \@namedef { the#2 } { \@nameuse { the#2 } } }
```

[20] https://github.com/CTeX-org/ctex-kit/issues/217

```
      {
        \protect \@namedef { the#2 } { \@nameuse { the#2 } }
        \CTEX@titlepslabel@set {#2}
      }
  }
\group_end:
```

\CTEX@titlepslabel@set
\CTEX@titlepslabel@clear

这两个函数要在随后被 \xdef 展开来获得 \CTEXtheXXX 的内容，不应该用 \protected 来定义。

```
\cs_new_nopar:Npn \CTEX@titlepslabel@set #1
  {
    \cs_if_free:cF { CTEXthe#1 }
      { \protect \@namedef { CTEXthe#1 } { \@nameuse { CTEXthe#1 } } }
  }
\cs_new_nopar:Npn \CTEX@titlepslabel@clear #1
  {
    \cs_if_free:cF { CTEXthe#1 }
      { \protect \@namedef { CTEXthe#1 } { } }
  }
```

titleps 宏包的功能可以由 titlesec 的选项 pagestyles 引入。

```
\ctex_at_end_package:nn { titlesec }
  { \cs_if_free:NF \ttl@settopmark { \ctex_titleps_hook: } }
\ctex_at_end_package:nn { titleps } { \ctex_titleps_hook: }
```

除此之外，也可以使用 titleps 提供的命令 \newtitlemark 来完成：

```
\newtitlemark { \CTEXthechapter }
\newtitlemark { \CTEXthesection }
```

但 \newtitlemark 不包含章节间的层次信息，功能上不及修改内部命令完整。

\ttl@setifthe

使 \iftheXXX 等命令在页眉设置中可用。

```
\ctex_at_end_package:nn { titleps }
  {
    \cs_set_protected_nopar:Npn \ttl@setifthe #1
      {
        \exp_args:Nco \cs_set_nopar:Npn { ifthe #1 }
          {
            \CTEXifname
              { \protect \@firstoftwo }
              { \protect \@secondoftwo }
          }
      }
    \seq_map_function:NN \c__ctex_headings_seq \ttl@setifthe
  }
```

14.8.3　目录标签的宽度

\numberline

```
\cs_new_protected:Npn \CTEX@toc@width@n #1
  {
    \hbox_set:Nn \l__ctex_tmp_box {#1}
    \dim_set:Nn \@tempdima
      {
        \dim_max:nn { \@tempdima }
          { \box_wd:N \l__ctex_tmp_box + \f@size \p@ / 2 }
      }
  }
\group_begin:
\char_set_catcode_other:N \#
\use:n
  {
```

```
    \group_end:
    \ctex_preto_cmd:NnnTF \numberline { \ExplSyntaxOff }
      { \CTEX@toc@width@n {#1} }
      { }
      { \ctex_patch_failure:N \numberline }
    \@ifpackageloaded { tocloft }
      { }
      {
        \ctex_at_end_package:nn { tocloft }
          {
            \ctex_preto_cmd:NnnTF \numberline
              { \char_set_catcode_letter:n { 64 } }
              { \CTEX@toc@width@n {#1} }
              { }
              { \ctex_patch_failure:N \numberline }
          }
      }
  }
```

14.8.4　页眉信息的修改

\ps@headings

```
⟨*article⟩
\if@twoside
  \ctex_patch_cmd:Nnn \ps@headings
    { \ifnum \c@secnumdepth > \z@ \thesection \quad \fi }
    { \CTEXifname { \CTEXthesection \quad } { } }
  \ctex_patch_cmd:Nnn \ps@headings
    { \ifnum \c@secnumdepth > \@ne \thesubsection \quad \fi }
    { \CTEXifname { \CTEXthesubsection \quad } { } }
\else:
```

不知为何，标准文档类此处对 secnumdepth 的判断为 0，与 \section 的层次 1 不符。

```
  \ctex_patch_cmd:Nnn \ps@headings
    { \ifnum \c@secnumdepth > \m@ne \thesection \quad \fi }
    { \CTEXifname { \CTEXthesection \quad } { } }
\fi:
⟨/article⟩
⟨*book|report⟩
\ctex_patch_cmd:Nnn \ps@headings
  {
⟨book⟩    \ifnum \c@secnumdepth > \m@ne \if@mainmatter
⟨report⟩  \ifnum \c@secnumdepth > \m@ne
      \@chapapp \ \thechapter . ~ \ %
⟨report⟩  \fi
⟨book⟩    \fi \fi
  }
  { \CTEXifname { \CTEXthechapter \quad } { } }
\if@twoside
  \ctex_patch_cmd:Nnn \ps@headings
    { \ifnum \c@secnumdepth > \z@ \thesection . ~ \ \fi }
    { \CTEXifname { \CTEXthesection \quad } { } }
\fi:
⟨/book|report⟩
```

\ps@fancy

这里对 fancyhdr 宏包打补丁。原来 fancyhdr 宏包中使用 \thesection 等宏表示页眉中的章节编号，这里改用 ctex 包所用的 \CTEXthesection 系列宏。

```
\ctex_at_end_package:nn { fancyhdr }
  {
⟨*article⟩
    \ctex_patch_cmd:Nnn \ps@fancy
      { \ifnum \c@secnumdepth > \z@ \thesection \hskip 1em \relax \fi }
      { \CTEXifname { \CTEXthesection \quad } { } }
    \ctex_patch_cmd:Nnn \ps@fancy
      { \ifnum \c@secnumdepth > \@ne \thesubsection \hskip 1em \relax \fi }
      { \CTEXifname { \CTEXthesubsection \quad } { } }
```

```
⟨/article⟩
⟨*book|report⟩
    \ctex_patch_cmd:Nnn \ps@fancy
      { \ifnum \c@secnumdepth > \m@ne \@chapapp \ \thechapter . ~ \ \fi }
      { \CTEXifname { \CTEXthechapter \quad } { } }
    \ctex_patch_cmd:Nnn \ps@fancy
      { \ifnum \c@secnumdepth > \z@ \thesection . ~ \ \fi }
      { \CTEXifname { \CTEXthesection \quad } { } }
⟨/book|report⟩
  }

⟨/article|book|report⟩
```

14.8.5 beamer 标题页模板的修改

```
⟨*beamer⟩
\ExplSyntaxOff
```

对应 \partpage。

```
\defbeamertemplate*{part page}{CTEX}[1][]{%
  \begingroup
    \CTEX@disableautoindent
%     \centering
%     {\usebeamerfont{part name}%
%      \usebeamercolor[fg]{part name}\partname~\insertromanpartnumber}
%     \vskip1em\par
    \par \addvspace{\glueexpr\CTEX@part@beforeskip\relax}%
    \CTEX@part@format
    \parindent \dimexpr \CTEX@part@indent \relax
    \ifodd \CTEX@part@numbering
      \CTEX@partname \CTEX@part@aftername
    \fi
    \begin{beamercolorbox}[sep=16pt,center,#1]{part title}
%       \usebeamerfont{part title}\insertpart\par
      \CTEX@part@titleformat \insertpart \CTEX@part@aftertitle
    \end{beamercolorbox}%
    \par \addvspace{\glueexpr\CTEX@part@afterskip\relax}%
  \endgroup
}
```

对应 \sectionpage。

```
\defbeamertemplate*{section page}{CTEX}[1][]{%
  \begingroup
    \CTEX@disableautoindent
%     \centering
%     {\usebeamerfont{section name}%
%      \usebeamercolor[fg]{section name}\sectionname~\insertsectionnumber}
%     \vskip1em\par
    \par \addvspace{\glueexpr\CTEX@section@beforeskip\relax}%
    \CTEX@section@format
    \parindent \dimexpr \CTEX@section@indent \relax
    \ifodd \CTEX@section@numbering
      \CTEX@sectionname \CTEX@section@aftername
    \fi
    \begin{beamercolorbox}[sep=12pt,center,#1]{part title}
%       \usebeamerfont{section title}\insertsection\par
      \CTEX@section@titleformat \insertsection \CTEX@section@aftertitle
    \end{beamercolorbox}%
    \par \addvspace{\glueexpr\CTEX@section@afterskip\relax}%
  \endgroup
}
```

对应 \subsectionpage。

```
\defbeamertemplate*{subsection page}{CTEX}[1][]{%
  \begingroup
```

```
    \CTEX@disableautoindent
%    \centering
%    {\usebeamerfont{subsection name}%
%     \usebeamercolor[fg]{subsection name}\subsectionname~\insertsubsectionnumber}
%    \vskip1em\par
    \par \addvspace{\glueexpr\CTEX@subsection@beforeskip\relax}%
    \CTEX@subsection@format
    \parindent \dimexpr \CTEX@subsection@indent \relax
    \ifodd \CTEX@subsection@numbering
      \CTEX@subsectionname \CTEX@subsection@aftername
    \fi
    \begin{beamercolorbox}[sep=8pt,center,#1]{part title}
%      \usebeamerfont{subsection title}\insertsubsection\par
      \CTEX@subsection@titleformat \insertsubsection \CTEX@subsection@aftertitle
    \end{beamercolorbox}%
    \par \addvspace{\glueexpr\CTEX@subsection@afterskip\relax}%
  \endgroup
}
```

将 beamer 的默认模板重定向为 CTEX 模板。

```
\defbeamertemplatealias{part page}{default}{CTEX}
\defbeamertemplatealias{section page}{default}{CTEX}
\defbeamertemplatealias{subsection page}{default}{CTEX}
\ExplSyntaxOn
⟨/beamer⟩
```

14.8.6 标签引用数字的汉化

\refstepcounter

对标题进行引用时，设置标签为通过 number 选项设置的形式。

```
\cs_new_protected_nopar:Npn \CTEX@setcurrentlabel@n #1
  {
    \protected@edef \@currentlabel
      {
        \cs_if_exist:cTF { CTEX@the#1 }
          { \exp_args:cc { p@#1 } { CTEX@the#1 } }
          { \exp_not:o { \@currentlabel } }
      }
  }
```

\ctex_varioref_hook:

关于标签引用的宏包可能会修改 \refstepcounter。其中 cleveref 和 hyperref 宏包都会保存之前的定义，并且它们都要求尽可能晚的被载入，所以对我们上述的修改影响不大。需要注意的是 varioref 宏包，如果它在 CTEX 之后被载入，我们之前的修改将会被覆盖。

```
\cs_new_protected_nopar:Npn \ctex_varioref_hook:
  {
    \seq_map_inline:Nn \c__ctex_headings_seq
      { \ctex_fix_varioref_label:n { ##1 } }
  }
```

__ctex_fix_varioref_label:n

varioref 宏包的 \labelformat 实际上是定义一个以 \the<#1> 为参数的宏 \p@<#1>。LaTeX 在定义计数器 <#1> 时，都会将 \p@<#1> 初始化为 \@empty。如果这个宏非空，说明用户自定义了标签格式，我们就不再修改。这里不能使用 \exp_args:Nnc，因为 c 这种展开格式不会将参数放在花括号内。而 \labelformat 的定义是

```
\def\labelformat#1{\expandafter\def\csname p@#1\endcsname##1}
```

它的第二个参数必须放在花括号内，否则将会被作为宏的定界符号。

```
\cs_new_protected_nopar:Npn \ctex_fix_varioref_label:n #1
  {
    \tl_if_empty:cT { p@#1 }
      { \exp_args:Nno \labelformat {#1} { \cs:w CTEX@the#1 \cs_end: } }
  }
```

如果 varioref 已经被载入，则使用它来设置。

```
\@ifpackageloaded { varioref }
  { \ctex_varioref_hook: }
  {
    \cs_new_eq:NN \CTEX@save@refstepcounter \refstepcounter
    \RenewDocumentCommand \refstepcounter { m }
      {
        \CTEX@save@refstepcounter {#1}
        \CTEX@setcurrentlabel@n {#1}
      }
    \ctex_at_end_package:nn { varioref } { \ctex_varioref_hook: }
  }
```

14.8.7　载入 ⟨*scheme*⟩ 文件

```
\ctex_scheme_input:o { \l__ctex_scheme_tl }
⟨/class|heading⟩
```

14.8.8　标题格式的 scheme 定义

下面使用 CTEX 文档类的设置方式，plain 模拟标准文档类直接定义或以 \@startsection 设定的章节标题格式，chinese 汉化的标题格式。

```
⟨*scheme&(article|book|report|beamer)⟩

\keys_set:nn { ctex / part }
  {
    aftertitle  = \par ,
⟨*article|book|report⟩
    hang        = false ,
⟨/article|book|report⟩
⟨*plain⟩
    name        = \partname \space ,
⟨*article|book|report⟩
    number      = \thepart ,
⟨/article|book|report⟩
⟨*beamer⟩
    number      = \insertromanpartnumber ,
⟨/beamer⟩
⟨/plain⟩
⟨*chinese⟩
    number      = \chinese { part } ,
⟨/chinese⟩
⟨*article⟩
    beforeskip  = 4ex ,
    afterskip   = 3ex ,
⟨*plain⟩
    format      = \raggedright ,
    nameformat  = \Large \bfseries ,
    aftername   = \par \nobreak ,
    titleformat = \huge \bfseries ,
    afterindent = false
⟨/plain⟩
⟨*chinese⟩
    format      = \Large \bfseries \centering ,
    aftername   = \quad ,
    afterindent = true
⟨/chinese⟩
⟨/article⟩
⟨*book|report⟩
    aftername   = \par \vskip 20 \p@ ,
    beforeskip  = 0pt \@plus 1fil ,
    afterskip   = 0pt \@plus 1fil ,
    pagestyle   = plain ,
    break       = \if@openright \cleardoublepage \else \clearpage \fi ,
```

```
⟨*plain⟩
    format      = \centering ,
    nameformat  = \huge \bfseries ,
    titleformat = \Huge \bfseries
⟨/plain⟩
⟨*chinese⟩
    format      = \huge \bfseries \centering
⟨/chinese⟩
⟨/book|report⟩
⟨*beamer⟩
    format      = \centering ,
    nameformat  = \usebeamerfont { part ~ name }
                  \usebeamercolor [fg] { part ~ name } ,
    aftername   = \vskip 1em \par ,
    titleformat = \usebeamerfont { part ~ title }
⟨/beamer⟩
  }
```

```
⟨*book|report⟩
\keys_set:nn { ctex / chapter }
  {
    pagestyle   = plain ,
    aftertitle  = \par ,
    hang        = false ,
    beforeskip  = 50 \p@ ,
    afterskip   = 40 \p@ ,
    lofskip     = 10 \p@ ,
    lotskip     = 10 \p@ ,
    break       = \if@openright \cleardoublepage \else \clearpage \fi ,
⟨*plain⟩
    name        = \chaptername \space ,
    number      = \thechapter ,
    format      = \raggedright ,
    nameformat  = \huge \bfseries ,
    aftername   = \par \nobreak \vskip 20 \p@ ,
    titleformat = \Huge \bfseries ,
    afterindent = false ,
    tocline     = \CTEXnumberline {#1} #2
⟨/plain⟩
⟨*chinese⟩
    number      = \chinese { chapter } ,
    format      = \huge \bfseries \centering ,
    aftername   = \quad ,
    afterindent = true
⟨/chinese⟩
  }
⟨/book|report⟩
```

```
⟨@@=⟩
```

```
\keys_set:nn { ctex / section }
  {
⟨*article|book|report⟩
    number      = \thesection ,
    aftername   = \quad ,
    aftertitle  = \@@par ,
    beforeskip  = 3.5ex \@plus 1ex \@minus .2ex ,
    afterskip   = 2.3ex \@plus .2ex ,
    runin       = false ,
    break       = \addpenalty \@secpenalty ,
⟨*plain⟩
    format      = \Large \bfseries ,
    afterindent = false
⟨/plain⟩
⟨*chinese⟩
    format      = \Large \bfseries \centering ,
    afterindent = true
⟨/chinese⟩
```

```
⟨/article|book|report⟩
⟨*beamer⟩
⟨*plain⟩
    name        = \sectionname \space ,
⟨/plain⟩
    format      = \centering ,
    number      = \insertsectionnumber ,
    nameformat  = \usebeamerfont { section ~ name }
                  \usebeamercolor [fg] { section ~ name } ,
    aftername   = \vskip 1em \par ,
    titleformat = \usebeamerfont { section ~ title } ,
    aftertitle  = \par
⟨/beamer⟩
  }
\keys_set:nn { ctex / subsection }
  {
⟨*article|book|report⟩
    number      = \thesubsection ,
    format      = \large \bfseries ,
    aftername   = \quad ,
    aftertitle  = \@@par ,
    beforeskip  = 3.25ex \@plus 1ex \@minus .2ex ,
    afterskip   = 1.5ex  \@plus .2ex ,
    runin       = false ,
    break       = \addpenalty \@secpenalty ,
⟨*plain⟩
    afterindent = false
⟨/plain⟩
⟨*chinese⟩
    afterindent = true
⟨/chinese⟩
⟨/article|book|report⟩
⟨*beamer⟩
⟨*plain⟩
    name        = \subsectionname \space ,
    number      = \insertsubsectionnumber ,
⟨/plain⟩
⟨*chinese⟩
    number      = \arabic { section } . \arabic { subsection } ,
⟨/chinese⟩
    format      = \centering ,
    nameformat  = \usebeamerfont { subsection ~ name }
                  \usebeamercolor [fg] { subsection ~ name } ,
    aftername   = \vskip 1em \par ,
    titleformat = \usebeamerfont { subsection ~ title } ,
    aftertitle  = \par
⟨/beamer⟩
  }
⟨*article|book|report⟩
\keys_set:nn { ctex / subsubsection }
  {
    number      = \thesubsubsection ,
    format      = \normalsize \bfseries ,
    aftername   = \quad ,
    aftertitle  = \@@par ,
    beforeskip  = 3.25ex \@plus 1ex \@minus .2ex ,
    afterskip   = 1.5ex \@plus .2ex ,
    runin       = false ,
    break       = \addpenalty \@secpenalty ,
⟨*plain⟩
    afterindent = false
⟨/plain⟩
⟨*chinese⟩
    afterindent = true
⟨/chinese⟩
```

```
  }
\keys_set:nn { ctex / paragraph }
  {
    number      = \theparagraph ,
    format      = \normalsize \bfseries ,
    aftername   = \quad ,
    beforeskip  = 3.25ex \@plus 1ex \@minus .2ex ,
    break       = \addpenalty \@secpenalty ,
⟨*plain⟩
    afterindent = false
⟨/plain⟩
⟨*chinese⟩
    afterindent = true
⟨/chinese⟩
  }
\keys_set:nn { ctex / subparagraph }
  {
    number      = \thesubparagraph ,
    format      = \normalsize \bfseries ,
    aftername   = \quad ,
    beforeskip  = 3.25ex \@plus 1ex \@minus .2ex ,
    break       = \addpenalty \@secpenalty ,
⟨*plain⟩
    afterindent = false
⟨/plain⟩
⟨*chinese⟩
    afterindent = true
⟨/chinese⟩
  }
```

处理 sub3section 与 sub4section 的格式。

```
\int_compare:nNnTF \g__ctex_section_depth_int > 2
  {
    \keys_set:nn { ctex / paragraph }
      {
        aftertitle  = \@@par ,
        afterskip   = 1ex \@plus .2ex ,
        runin       = false
      }
  }
  {
    \keys_set:nn { ctex / paragraph }
      {
        afterskip   = 1em ,
        runin       = true
      }
  }
\int_compare:nNnTF \g__ctex_section_depth_int > 3
  {
    \keys_set:nn { ctex / subparagraph }
      {
        aftertitle  = \@@par ,
        afterskip   = 1ex \@plus .2ex ,
        runin       = false
      }
  }
  {
    \keys_set:nn { ctex / subparagraph }
      {
        afterskip   = 1em ,
        runin       = true
      }
  }
\int_compare:nNnTF \g__ctex_section_depth_int > 2
  { \keys_set:nn { ctex / subparagraph } { indent = \c_zero_dim } }
```

```
  { \keys_set:nn { ctex / subparagraph } { indent = \parindent } }
```

⟨@@=ctex⟩

处理附录的格式。

```
\keys_set:nn { ctex / appendix }
⟨*article⟩
  { number      = \@Alph \c@section }
⟨/article⟩
⟨*book|report⟩
  {
    name        = \appendixname \space ,
    number      = \@Alph \c@chapter
  }
⟨/book|report⟩
```

⟨/article|book|report⟩

⟨/scheme&(article|book|report|beamer)⟩

14.8.9 ctex.sty 的 heading 选项

⟨*ctex|ctexheading⟩

\c__ctex_std_class_tl 用于记录被引入的标准文档类。

```
\clist_map_inline:nn { article , book , report , beamer }
  {
    \@ifclassloaded {#1}
      { \clist_map_break:n { \tl_const:Nn \c__ctex_std_class_tl {#1} } }
      { }
  }
```

若标准文档类被引入，则载入对应的标题定义文件。否则视 \chapter 是否有定义来引入 book 或者 article。

```
\msg_new:nnn { ctex } { not-standard-class }
  {
    None~of~the~standard~document~classes~was~loaded.\\
    Heading~`#1'~is~selected.\\
    ctex~may~not~work~as~expected.
  }
⟨ctex⟩ \bool_if:NTF \l__ctex_heading_bool
⟨ctexheading⟩ \use:n
  {
    \tl_if_exist:NTF \c__ctex_std_class_tl
      { \cs_new_eq:NN \c__ctex_class_tl \c__ctex_std_class_tl }
      {
        \cs_if_exist:NTF \chapter
          {
            \cs_if_exist:NF \if@mainmatter
              { \cs_new_eq:NN \if@mainmatter \tex_iftrue:D }
            \tl_const:Nn \c__ctex_class_tl { book }
          }
          { \tl_const:Nn \c__ctex_class_tl { article } }
        \msg_warning:nnx { ctex } { not-standard-class } { \c__ctex_class_tl }
      }
    \ctex_file_input:n { ctex- \c__ctex_class_tl .def }
  }
⟨ctex⟩ { \ctex_scheme_input:o { \l__ctex_scheme_tl } }
```

⟨/ctex|ctexheading⟩

14.8.10 标题配置文件

⟨*name⟩

```
\keys_set_known:nn { ctex }
  {
    contentsname   = 目录 ,
    listfigurename = 插图 ,
    listtablename  = 表格 ,
    figurename     = 图 ,
    tablename      = 表 ,
    abstractname   = 摘要 ,
    indexname      = 索引 ,
    bibname        = 参考文献 ,
    appendixname   = 附录 ,
    proofname      = 证明 ,
    algorithmname  = 算法 ,
    refname        = 参考文献 ,
    continuation   = （续） ,
    part    / name = { 第 , 部分 } ,
    chapter / name = { 第 , 章 }
  }
```

⟨/name⟩

14.9 chinese 方案的其他设置

⟨*scheme&chinese⟩

chinese 在标准文档类下的页面格式总采用 headings。

⟨article|book|report⟩
```
\pagestyle { headings }
```

日期格式。

```
\keys_set:nn { ctex } { today = small }
```

若用户未设置宏包选项 autoindent，则自动调整首行缩进。

```
\ctex_if_autoindent_touched:F
  { \keys_set:nn { ctex } { autoindent = true } }
```

使用标题定义时的设置。首先是命题名字汉化。beamer 需要汉化定理名称。

```
⟨*!generic⟩
\str_if_eq:onTF { \l__ctex_encoding_tl } { GBK }
⟨*beamer⟩
  {
    \uselanguage { ChineseGBK }
    \languagealias { chinese } { ChineseGBK }
    \ctex_file_input:n { ctex-name-gbk.cfg }
  }
  {
    \uselanguage { ChineseUTF8 }
    \languagealias { chinese } { ChineseUTF8 }
    \ctex_file_input:n { ctex-name-utf8.cfg }
  }
```

让 translator 包优先查找中文翻译。

```
\clist_put_left:Nn \trans@languagepath { chinese }
⟨/beamer⟩
⟨*!beamer⟩
  { \ctex_file_input:n { ctex-name-gbk.cfg } }
  { \ctex_file_input:n { ctex-name-utf8.cfg } }
```

对 beamer 以外的文档类，若用户未设置宏包选项 zihao，则设置 \normalsize 为五号字。beamer 不调整默认字体大小。

```
\int_compare:nNnF \g__ctex_font_size_int > { -1 }
  { \int_gset:Nn \g__ctex_font_size_int { 0 } }
```

对 beamer 以外的文档类，若用户未设置宏包选项 linespread，则设置行距初始值为 $1.3 \times 1.2 = 1.56$ 倍字体大小。beamer 不调整行距。

```
\fp_compare:nNnT { \l__ctex_line_spread_fp } ? { \c_zero_fp }
```

```
  { \fp_set:Nn \l__ctex_line_spread_fp { 1.3 } }
⟨/!beamer⟩
⟨/!generic⟩
```

不使用标题定义时的通用设置。

```
⟨*generic⟩
\tl_set:Nn \l__ctex_tmp_tl { beamer }
\tl_if_eq:NNTF \c__ctex_std_class_tl \l__ctex_tmp_tl
  {
    \str_if_eq:onTF { \l__ctex_encoding_tl } { GBK }
      {
        \uselanguage { ChineseGBK }
        \languagealias { chinese } { ChineseGBK }
        \ctex_file_input:n { ctex-name-gbk.cfg }
      }
      {
        \uselanguage { ChineseUTF8 }
        \languagealias { chinese } { ChineseUTF8 }
        \ctex_file_input:n { ctex-name-utf8.cfg }
      }
    \clist_put_left:Nn \trans@languagepath { chinese }
  }
  {
    \str_if_eq:onTF { \l__ctex_encoding_tl } { GBK }
      { \ctex_file_input:n { ctex-name-gbk.cfg } }
      { \ctex_file_input:n { ctex-name-utf8.cfg } }
    \int_compare:nNnF \g__ctex_font_size_int > { -1 }
      { \int_gset:Nn \g__ctex_font_size_int { 0 } }
    \fp_compare:nNnT { \l__ctex_line_spread_fp } ? { \c_zero_fp }
      { \fp_set:Nn \l__ctex_line_spread_fp { 1.3 } }
```

若 ctex 宏包与标准文档类及其衍生文档类联用，则将载入 indentfirst 宏包，实现章节标题后首个段落的段首缩进。

```
    \tl_if_exist:NT \c__ctex_std_class_tl
      { \RequirePackage { indentfirst } }
  }
⟨/generic⟩
⟨/scheme&chinese⟩
```

14.10　中文字号

```
⟨*class|ctex|ctexsize⟩
```

\zihao

```
\NewDocumentCommand \zihao { m }
  { \exp_args:Nx \ctex_zihao:n {#1} \tex_ignorespaces:D }
```

\ctex_zihao:n

```
\cs_new_protected_nopar:Npn \ctex_zihao:n #1
  {
    \prop_get:NnNTF \c__ctex_font_size_prop {#1} \l__ctex_font_size_tl
      { \exp_after:wN \fontsize \l__ctex_font_size_tl \selectfont }
      { \msg_error:nnn { ctex } { fontsize } {#1} }
  }
\msg_new:nnnn { ctex } { fontsize }
  { Undefined~Chinese~font~size~`#1'~in~command~\token_to_str:N \zihao.}
  {
    The~old~font~size~is~used~if~you~continue.\\
    The~available~font~sizes~are~listed~as~follow.\\
    \seq_use:Nnnn \c__ctex_font_size_seq { ~and~ } { ,~ } { ,~and~ }.
  }
```

14.10.1　定义中文字号

\c__ctex_font_size_prop
__ctex_save_font_size:nn

基础行距是字号的 1.2 倍，采用 ε-TEX 的 scaling 运算得到的结果要比简单的 `1.2\dimexpr` 精确[21]。

```
\prop_new:N \c__ctex_font_size_prop
\seq_new:N \c__ctex_font_size_seq
\cs_new_protected_nopar:Npn \__ctex_save_font_size:nn #1#2
  {
    \use:x
      {
        \prop_gput:Nnn \exp_not:N \c__ctex_font_size_prop {#1}
          {
            { \dim_to_decimal:n {#2} }
            { \dim_to_decimal:n { (#2) * 6 / 5 } }
          }
      }
    \seq_gput_right:Nn \c__ctex_font_size_seq {#1}
  }
\clist_map_inline:nn
  {
    {  8 } { 5    bp } ,
    {  7 } { 5.5  bp } ,
    { -6 } { 6.5  bp } ,
    {  6 } { 7.5  bp } ,
    { -5 } { 9    bp } ,
    {  5 } { 10.5 bp } ,
    { -4 } { 12   bp } ,
    {  4 } { 14   bp } ,
    { -3 } { 15   bp } ,
    {  3 } { 16   bp } ,
    { -2 } { 18   bp } ,
    {  2 } { 22   bp } ,
    { -1 } { 24   bp } ,
    {  1 } { 26   bp } ,
    { -0 } { 36   bp } ,
    {  0 } { 42   bp }
  }
  { \__ctex_save_font_size:nn #1 }
```

\ctex_declare_math_sizes:nnnn

```
\cs_new_protected_nopar:Npn \ctex_declare_math_sizes:nnnn #1#2#3#4
  {
    \__ctex_get_font_sizes:Nn \l__ctex_font_size_tl { {#1} {#2} {#3} {#4} }
    \exp_after:wN \DeclareMathSizes \l__ctex_font_size_tl
  }
```

__ctex_get_font_sizes:Nn

```
\cs_new_protected_nopar:Npn \__ctex_get_font_sizes:Nn #1#2
  {
    \tl_clear:N #1
    \tl_map_inline:nn {#2}
      {
        \prop_get:NnNTF \c__ctex_font_size_prop {##1} \l__ctex_tmp_tl
          { \tl_put_right:Nx #1 { { \tl_head:N \l__ctex_tmp_tl } } }
          { \tl_put_right:Nx #1 { { \dim_to_decimal:n { ##1 } } } }
      }
  }

\clist_map_inline:nn
  {
    {  8 }{  8 }{ 5pt }{ 5pt } ,
    {  7 }{  7 }{ 5pt }{ 5pt } ,
    { -6 }{ -6 }{ 5pt }{ 5pt } ,
    {  6 }{  6 }{ 5pt }{ 5pt } ,
    { -5 }{ -5 }{ 6pt }{ 5pt } ,
    {  5 }{  5 }{ 7pt }{ 5pt } ,
```

[21] http://thread.gmane.org/gmane.comp.tex.latex.latex3/3190

```
    { -4 }{ -4 }{ 8pt }{ 6pt } ,
    {  4 }{  4 }{  5 }{  6 } ,
    { -3 }{ -3 }{ -4 }{ -5 } ,
    {  3 }{  3 }{  4 }{  5 } ,
    { -2 }{ -2 }{ -3 }{ -4 } ,
    {  2 }{  2 }{  3 }{  4 } ,
    { -1 }{ -1 }{ -2 }{ -3 } ,
    {  1 }{  1 }{  2 }{  3 } ,
    { -0 }{ -0 }{ -1 }{ -2 } ,
    {  0 }{  0 }{  1 }{  2 }
  }
  { \ctex_declare_math_sizes:nnnn #1 }
```

14.10.2　修改默认字号大小

\ctex_set_font_size:Nnn

```
\cs_new_protected:Npn \ctex_set_font_size:Nnn #1#2#3
  {
    \prop_get:NnNTF \c__ctex_font_size_prop {#2} \l__ctex_font_size_tl
      { \exp_after:wN \__ctex_set_font_size:nnNn \l__ctex_font_size_tl #1 {#3} }
      { \msg_error:nnn { ctex } { fontsize } {#2} }
  }
\cs_new_protected:Npn \__ctex_set_font_size:nnNn #1#2#3#4
  { \cs_set_protected_nopar:Npn #3 { \@setfontsize #3 {#1} {#2} #4 } }
```

```
\if_case:w \g__ctex_font_size_int
  \ctex_file_input:n { ctex-c5size.clo }
\or:
  \ctex_file_input:n { ctex-cs4size.clo }
\fi:
```

⟨/class|ctex|ctexsize⟩

⟨*c5size⟩

```
\ctex_set_font_size:Nnn \normalsize { 5 }
  {
    \abovedisplayskip 10\p@ \@plus2\p@ \@minus5\p@
    \abovedisplayshortskip \z@ \@plus3\p@
    \belowdisplayshortskip 6\p@ \@plus3\p@ \@minus3\p@
    \belowdisplayskip \abovedisplayskip
    \let\@listi\@listI
  }
\ctex_set_font_size:Nnn \small { -5 }
  {
    \abovedisplayskip 8.5\p@ \@plus3\p@ \@minus4\p@
    \abovedisplayshortskip \z@ \@plus2\p@
    \belowdisplayshortskip 4\p@ \@plus2\p@ \@minus2\p@
    \def\@listi{\leftmargin\leftmargini
                \topsep 4\p@ \@plus2\p@ \@minus2\p@
                \parsep 2\p@ \@plus\p@ \@minus\p@
                \itemsep \parsep}
    \belowdisplayskip \abovedisplayskip
  }
\ctex_set_font_size:Nnn \footnotesize { 6 }
  {
    \abovedisplayskip 6\p@ \@plus2\p@ \@minus4\p@
    \abovedisplayshortskip \z@ \@plus\p@
    \belowdisplayshortskip 3\p@ \@plus\p@ \@minus2\p@
    \def\@listi{\leftmargin\leftmargini
                \topsep 3\p@ \@plus\p@ \@minus\p@
                \parsep 2\p@ \@plus\p@ \@minus\p@
                \itemsep \parsep}
    \belowdisplayskip \abovedisplayskip
  }
\ctex_set_font_size:Nnn \scriptsize { -6 } { }
\ctex_set_font_size:Nnn \tiny  {  7 } { }
\ctex_set_font_size:Nnn \large { -4 } { }
\ctex_set_font_size:Nnn \Large { -3 } { }
```

```
\ctex_set_font_size:Nnn \LARGE { -2 } { }
\ctex_set_font_size:Nnn \huge  {  2 } { }
\ctex_set_font_size:Nnn \Huge  {  1 } { }
⟨/c5size⟩
⟨*cs4size⟩
\ctex_set_font_size:Nnn \normalsize { -4 }
  {
    \abovedisplayskip 12\p@ \@plus3\p@ \@minus7\p@
    \abovedisplayshortskip \z@ \@plus3\p@
    \belowdisplayshortskip 6.5\p@ \@plus3.5\p@ \@minus3\p@
    \belowdisplayskip \abovedisplayskip
    \let\@listi\@listI
  }
\ctex_set_font_size:Nnn \small { 5 }
  {
    \abovedisplayskip 11\p@ \@plus3\p@ \@minus6\p@
    \abovedisplayshortskip \z@ \@plus3\p@
    \belowdisplayshortskip 6.5\p@ \@plus3.5\p@ \@minus3\p@
    \def\@listi{\leftmargin\leftmargini
                \topsep 9\p@ \@plus3\p@ \@minus5\p@
                \parsep 4.5\p@ \@plus2\p@ \@minus\p@
                \itemsep \parsep}
    \belowdisplayskip \abovedisplayskip
  }
\ctex_set_font_size:Nnn \footnotesize { -5 }
  {
    \abovedisplayskip 10\p@ \@plus2\p@ \@minus5\p@
    \abovedisplayshortskip \z@ \@plus3\p@
    \belowdisplayshortskip 6\p@ \@plus3\p@ \@minus3\p@
    \def\@listi{\leftmargin\leftmargini
                \topsep 6\p@ \@plus2\p@ \@minus2\p@
                \parsep 3\p@ \@plus2\p@ \@minus\p@
                \itemsep \parsep}
    \belowdisplayskip \abovedisplayskip
  }
\ctex_set_font_size:Nnn \scriptsize { 6 } { }
\ctex_set_font_size:Nnn \tiny  { -6 } { }
\ctex_set_font_size:Nnn \large { -3 } { }
\ctex_set_font_size:Nnn \Large { -2 } { }
\ctex_set_font_size:Nnn \LARGE {  2 } { }
\ctex_set_font_size:Nnn \huge  { -1 } { }
\ctex_set_font_size:Nnn \Huge  {  1 } { }
⟨/cs4size⟩
```

⟨ctexsize⟩

```
\normalsize
⟨*class|ctex⟩
```

14.11　更新行距

\l__ctex_line_spread_fp 被设置了才有必要更新行距和 \footnotesep。

```
\fp_compare:nNnF { \l__ctex_line_spread_fp } ? { \c_zero_fp }
  {
    \exp_args:Nx \linespread { \fp_use:N \l__ctex_line_spread_fp }
```

\footnotesep　我们调整了行距，可能导致脚注的间距与行距不协调，需要调整 \footnotesep。标准文档类对 \footnotesep 的设置是，字体大小为 \footnotesize 时 \strutbox 的高度（默认值是 .7\baselineskip）。我们沿用这个设置方法，只需要更新具体的大小。

```
    \group_begin: \footnotesize \exp_args:NNNo \group_end:
    \dim_set:Nn \footnotesep { \dim_use:N \box_ht:N \strutbox }
  }
```

激活默认字体大小，更新行距、\parindent 和 \CJKglue。

```
\normalsize
```

14.12　其它功能

\CTeX　ctex-faq.sty 中的定义是

```
\DeclareRobustCommand\CTeX{$\mathbb{C}$\kern-.05em\TeX}
```

然而 \mathbb 未必有定义，这里就不采用它了，只定义最简单的形式。CTEX 可以直接用在 PDF 书签中。

```
\NewDocumentCommand \CTeX { }
  { C \TeX }
\ctex_at_end_package:nn { hyperref }
  { \pdfstringdefDisableCommands { \tl_set:Nn \CTeX { CTeX } } }
```

captiondelimiter　过时选项。

```
\keys_define:nn { ctex }
  {
    captiondelimiter .code:n =
      {
        \msg_warning:nnn { ctex } { deprecated-option }
          { You~can~load~the~package~`caption'~to~get~its~functionality. }
      }
  }
```

```
⟨/class|ctex⟩
```

14.12.1　列表环境的缩进

\verse
\quotation　只在使用文档类的时候修改诗歌和引用环境的缩进。

```
⟨*scheme&chinese&(article|book|report)⟩
\ctex_patch_cmd:Nnn \verse { -1.5em } { -2 \ccwd }
\ctex_patch_cmd:Nnn \verse {  1.5em } {  2 \ccwd }
\ctex_patch_cmd:Nnn \quotation { 1.5em } { 2 \ccwd }
⟨/scheme&chinese&(article|book|report)⟩
```

```
⟨*class|ctex⟩
```

14.13　载入中文字体

\ctex_fontset_error:n　字库不可用时给出紧急错误信息，停止读取定义文件。

```
\cs_new_protected_nopar:Npn \ctex_fontset_error:n #1
  { \msg_critical:nnn { ctex } { fontset-unavailable } {#1} }
\msg_new:nnn { ctex } { fontset-unavailable }
  { CTeX~fontset~`#1'~is~unavailable~in~current~mode. }
```

\ctex_load_fontset:　如果用户没有指定字体，则探测操作系统，载入相应的字体配置。

```
\cs_new_protected_nopar:Npn \ctex_load_fontset:
  {
    \tl_if_empty:NTF \g__ctex_fontset_tl
      {
        \ctex_detected_platform:
        \ctex_file_input:n { ctex-fontset- \g__ctex_fontset_tl .def }
      }
      {
        \file_if_exist:nTF { ctex-fontset- \g__ctex_fontset_tl .def }
          { \ctex_file_input:n { ctex-fontset- \g__ctex_fontset_tl .def } }
          {
            \use:x
              {
```

```
                \ctex_detected_platform:
                \msg_error:nnxx { ctex } { fontset-not-found }
                  { \g__ctex_fontset_tl } { \exp_not:N \g__ctex_fontset_tl }
              }
            \ctex_file_input:n { ctex-fontset- \g__ctex_fontset_tl .def }
          }
      }
  }
\@onlypreamble \ctex_load_fontset:
\msg_new:nnnn { ctex } { fontset-not-found }
  {
    CTeX~fontset~`#1'~could~not~be~found.\\
    Fontset~`#2'~will~be~used~instead.
  }
  { You~may~run~`mktexlsr'~firstly. }
```

fontset　在导言区通过 \ctexset 载入中文字库的选项。

```
\keys_define:nn { ctex }
  {
    fontset .code:n =
      {
        \ctex_if_preamble:TF
          {
            \str_if_eq:eeTF {#1} { none }
              { \msg_warning:nnn { ctex } { invalid-value } {#1} }
              {
                \str_if_eq:onTF { \g__ctex_fontset_tl } { none }
                  {
                    \tl_gset:Nx \g__ctex_fontset_tl {#1}
                    \ctex_load_fontset:
                  }
                  {
                    \msg_error:nnxx { ctex } { fontset-loaded }
                      { \g__ctex_fontset_tl } {#1}
                  }
              }
          }
          { \msg_error:nn { ctex } { fontset-only-preamble } }
      }
  }
\msg_new:nnnn { ctex } { fontset-loaded }
  {
    CTeX~fontset~`#1'~has~been~loaded.
    \str_if_eq:nnF {#1} {#2} { \\ Fontset~`#2'~will~be~ignored. }
  }
  { Only~one~fontset~can~be~loaded~in~the~preamble. }
\msg_new:nnn { ctex } { fontset-only-preamble }
  {
    The~ `fontset'~ option~ can~ be~ used~ only~ in~ preamble.
  }
```

载入中文字库。

```
\str_if_eq:onF { \g__ctex_fontset_tl } { none }
  { \ctex_load_fontset: }
```

14.14　宏包配置文件

14.14.1　ctex.cfg

```
\ctex_at_end:n { \ctex_file_input:n { ctex.cfg } }
```

⟨/class|ctex⟩

⟨*config⟩

```
%%
```

```
⟨/config⟩
```

14.14.2 ctexopts.cfg

这里仅为配置文件示例：使用 Windows Vista 或以后版本的字体设置。

```
⟨*ctexopts⟩
%%
%% \keys_set:nn { ctex / option } { fontset = windowsnew }
⟨/ctexopts⟩
```

14.15 字体定义文件

14.15.1 传统定义方式

```
⟨*c19|c70⟩
%%
%% Chinese characters
%%
⟨c19⟩ %% character set: GBK (extension of GB 2312)
⟨c70⟩ %% character set: Unicode
%% font encoding: Unicode
%%
⟨/c19|c70⟩
```

CJK 宏包使用的字体族。

```
⟨rm&c19⟩ \DeclareFontFamily{C19}{rm}{\hyphenchar\font\m@ne}
⟨rm&c70⟩ \DeclareFontFamily{C70}{rm}{\hyphenchar\font\m@ne}
⟨sf&c19⟩ \DeclareFontFamily{C19}{sf}{\hyphenchar\font\m@ne}
⟨sf&c70⟩ \DeclareFontFamily{C70}{sf}{\hyphenchar\font\m@ne}
⟨tt&c19⟩ \DeclareFontFamily{C19}{tt}{\hyphenchar\font\m@ne}
⟨tt&c70⟩ \DeclareFontFamily{C70}{tt}{\hyphenchar\font\m@ne}
```

upLaTeX 使用的字体族。upLaTeX 在 NFSS 下使用字体编码 JY2 和 JT2 来分别表示横排与直排的日文。

```
⟨rm&jy2⟩ \DeclareKanjiFamily{JY2}{zhrm}{}
⟨rm&jt2⟩ \DeclareKanjiFamily{JT2}{zhrm}{}
⟨sf&jy2⟩ \DeclareKanjiFamily{JY2}{zhsf}{}
⟨sf&jt2⟩ \DeclareKanjiFamily{JT2}{zhsf}{}
⟨tt&jy2⟩ \DeclareKanjiFamily{JY2}{zhtt}{}
⟨tt&jt2⟩ \DeclareKanjiFamily{JT2}{zhtt}{}
```

```
⟨*rm⟩
⟨*c19⟩
\DeclareFontShape{C19}{rm}{m}{n}{<-> CJK * gbksong}{\CJKnormal}
\DeclareFontShape{C19}{rm}{b}{n}{<-> CJK * gbkhei}{\CJKnormal}
\DeclareFontShape{C19}{rm}{bx}{n}{<-> CJK * gbkhei}{\CJKnormal}
\DeclareFontShape{C19}{rm}{m}{sl}{<-> CJK * gbksongsl}{\CJKnormal}
\DeclareFontShape{C19}{rm}{b}{sl}{<-> CJK * gbkheisl}{\CJKnormal}
\DeclareFontShape{C19}{rm}{bx}{sl}{<-> CJK * gbkheisl}{\CJKnormal}
\DeclareFontShape{C19}{rm}{m}{it}{<-> CJK * gbkkai}{\CJKnormal}
\DeclareFontShape{C19}{rm}{b}{it}{<-> CJKb * gbkkai}{\CJKbold}
\DeclareFontShape{C19}{rm}{bx}{it}{<-> CJKb * gbkkai}{\CJKbold}
⟨/c19⟩
⟨*c70⟩
\DeclareFontShape{C70}{rm}{m}{n}{<-> CJK * unisong}{\CJKnormal}
\DeclareFontShape{C70}{rm}{b}{n}{<-> CJK * unihei}{\CJKnormal}
\DeclareFontShape{C70}{rm}{bx}{n}{<-> CJK * unihei}{\CJKnormal}
\DeclareFontShape{C70}{rm}{m}{sl}{<-> CJK * unisongsl}{\CJKnormal}
\DeclareFontShape{C70}{rm}{b}{sl}{<-> CJK * uniheisl}{\CJKnormal}
\DeclareFontShape{C70}{rm}{bx}{sl}{<-> CJK * uniheisl}{\CJKnormal}
\DeclareFontShape{C70}{rm}{m}{it}{<-> CJK * unikai}{\CJKnormal}
\DeclareFontShape{C70}{rm}{b}{it}{<-> CJKb * unikai}{\CJKbold}
\DeclareFontShape{C70}{rm}{bx}{it}{<-> CJKb * unikai}{\CJKbold}
⟨/c70⟩
```

```
⟨*jy2⟩
\DeclareFontShape{JY2}{zhrm}{m}{n}{<-> upzhserif-h}{}
\DeclareFontShape{JY2}{zhrm}{m}{it}{<-> upzhserifit-h}{}
\DeclareFontShape{JY2}{zhrm}{bx}{n}{<-> upzhserifb-h}{}
⟨/jy2⟩
⟨*jt2⟩
\DeclareFontShape{JT2}{zhrm}{m}{n}{<-> upzhserif-v}{}
\DeclareFontShape{JT2}{zhrm}{m}{it}{<-> upzhserifit-v}{}
\DeclareFontShape{JT2}{zhrm}{bx}{n}{<-> upzhserifb-v}{}
⟨/jt2⟩
⟨/rm⟩
⟨*sf⟩
⟨*c19⟩
\DeclareFontShape{C19}{sf}{m}{n}{<-> CJK * gbkyou}{\CJKnormal}
\DeclareFontShape{C19}{sf}{b}{n}{<-> CJKb * gbkyou}{\CJKbold}
\DeclareFontShape{C19}{sf}{bx}{n}{<-> CJKb * gbkyou}{\CJKbold}
\DeclareFontShape{C19}{sf}{m}{sl}{<-> CJK * gbkyousl}{\CJKnormal}
\DeclareFontShape{C19}{sf}{b}{sl}{<-> CJKb * gbkyousl}{\CJKbold}
\DeclareFontShape{C19}{sf}{bx}{sl}{<-> CJKb * gbkyousl}{\CJKbold}
\DeclareFontShape{C19}{sf}{m}{it}{<-> CJK * gbkyou}{\CJKnormal}
\DeclareFontShape{C19}{sf}{b}{it}{<-> CJKb * gbkyou}{\CJKbold}
\DeclareFontShape{C19}{sf}{bx}{it}{<-> CJKb * gbkyou}{\CJKbold}
⟨/c19⟩
⟨*c70⟩
\DeclareFontShape{C70}{sf}{m}{n}{<-> CJK * uniyou}{\CJKnormal}
\DeclareFontShape{C70}{sf}{b}{n}{<-> CJKb * uniyou}{\CJKbold}
\DeclareFontShape{C70}{sf}{bx}{n}{<-> CJKb * uniyou}{\CJKbold}
\DeclareFontShape{C70}{sf}{m}{sl}{<-> CJK * uniyousl}{\CJKnormal}
\DeclareFontShape{C70}{sf}{b}{sl}{<-> CJKb * uniyousl}{\CJKbold}
\DeclareFontShape{C70}{sf}{bx}{sl}{<-> CJKb * uniyousl}{\CJKbold}
\DeclareFontShape{C70}{sf}{m}{it}{<-> CJK * uniyou}{\CJKnormal}
\DeclareFontShape{C70}{sf}{b}{it}{<-> CJKb * uniyou}{\CJKbold}
\DeclareFontShape{C70}{sf}{bx}{it}{<-> CJKb * uniyou}{\CJKbold}
⟨/c70⟩
⟨*jy2⟩
\DeclareFontShape{JY2}{zhsf}{m}{n}{<-> upzhsans-h}{}
\DeclareFontShape{JY2}{zhsf}{bx}{n}{<-> upzhsansb-h}{}
⟨/jy2⟩
⟨*jt2⟩
\DeclareFontShape{JT2}{zhsf}{m}{n}{<-> upzhsans-v}{}
\DeclareFontShape{JT2}{zhsf}{bx}{n}{<-> upzhsansb-v}{}
⟨/jt2⟩
⟨/sf⟩
⟨*tt⟩
⟨*c19⟩
\DeclareFontShape{C19}{tt}{m}{n}{<-> CJK * gbkfs}{\CJKnormal}
\DeclareFontShape{C19}{tt}{b}{n}{<-> CJKb * gbkfs}{\CJKbold}
\DeclareFontShape{C19}{tt}{bx}{n}{<-> CJKb * gbkfs}{\CJKbold}
\DeclareFontShape{C19}{tt}{m}{sl}{<-> CJK * gbkfssl}{\CJKnormal}
\DeclareFontShape{C19}{tt}{b}{sl}{<-> CJKb * gbkfssl}{\CJKbold}
\DeclareFontShape{C19}{tt}{bx}{sl}{<-> CJKb * gbkfssl}{\CJKbold}
\DeclareFontShape{C19}{tt}{m}{it}{<-> CJK * gbkfs}{\CJKnormal}
\DeclareFontShape{C19}{tt}{b}{it}{<-> CJKb * gbkfs}{\CJKbold}
\DeclareFontShape{C19}{tt}{bx}{it}{<-> CJKb * gbkfs}{\CJKbold}
⟨/c19⟩
⟨*c70⟩
\DeclareFontShape{C70}{tt}{m}{n}{<-> CJK * unifs}{\CJKnormal}
\DeclareFontShape{C70}{tt}{b}{n}{<-> CJKb * unifs}{\CJKbold}
\DeclareFontShape{C70}{tt}{bx}{n}{<-> CJKb * unifs}{\CJKbold}
\DeclareFontShape{C70}{tt}{m}{sl}{<-> CJK * unifssl}{\CJKnormal}
\DeclareFontShape{C70}{tt}{b}{sl}{<-> CJKb * unifssl}{\CJKbold}
\DeclareFontShape{C70}{tt}{bx}{sl}{<-> CJKb * unifssl}{\CJKbold}
\DeclareFontShape{C70}{tt}{m}{it}{<-> CJK * unifs}{\CJKnormal}
\DeclareFontShape{C70}{tt}{b}{it}{<-> CJKb * unifs}{\CJKbold}
\DeclareFontShape{C70}{tt}{bx}{it}{<-> CJKb * unifs}{\CJKbold}
⟨/c70⟩
```

```
⟨*jy2⟩
\DeclareFontShape{JY2}{zhtt}{m}{n}{<-> upzhmono-h}{}
⟨/jy2⟩
⟨*jt2⟩
\DeclareFontShape{JT2}{zhtt}{m}{n}{<-> upzhmono-v}{}
⟨/jt2⟩
⟨/tt⟩
⟨*fontset⟩
```

14.15.2 ctex-fontset-windows.def 等

ctex 包利用 `C:\bootfont.bin` 文件的存在性来判断是否使用 Windows XP 版本，分别载入新旧字体设置。（注：ctex 包不支持 Windows 2000 以前使用 `simsun.ttf` 的旧宋体文件名。）

```
⟨*windows⟩
\file_if_exist:nTF { C:/bootfont.bin }
  { \ctex_file_input:n { ctex-fontset-windowsold.def } }
  { \ctex_file_input:n { ctex-fontset-windowsnew.def } }
⟨/windows⟩
```

旧的 Windows 字体设置使用黑体作为无衬线体，楷体和仿宋是 GB2312 编码；新的 Windows 字体设置使用微软雅黑作为无衬线体，楷体和仿宋是大字库。Windows 8 以后，微软雅黑由原来的 `.ttf` 后缀改为 `.ttc` 后缀，需要加以区分。

```
⟨*windowsnew⟩
\tl_new:N \l__ctex_msyh_suffix_tl
\tl_set:Nn \l__ctex_msyh_suffix_tl { .ttc }
\file_if_exist:nF { C:/Windows/Fonts/msyh.ttc }
  {
    \file_if_exist:nF { msyh.ttc }
      { \tl_set:Nn \l__ctex_msyh_suffix_tl { .ttf } }
  }
⟨/windowsnew⟩
⟨*windowsnew|windowsold⟩
\sys_if_engine_pdftex:TF
  {
    \ctex_zhmap_case:nnn
      {
        \ctex_punct_set:n { windows }
        \setCJKmainfont
          [ BoldFont = simhei.ttf , ItalicFont = simkai.ttf ] { simsun.ttc }
⟨*windowsold⟩
        \setCJKsansfont { simhei.ttf }
        \ctex_punct_map_family:nn { \CJKsfdefault } { zhhei }
⟨/windowsold⟩
⟨*windowsnew⟩
        \setCJKsansfont
          [ BoldFont = msyhbd\l__ctex_msyh_suffix_tl ] { msyh\l__ctex_msyh_suffix_tl }
        \setCJKfamilyfont { zhyahei }
          [ BoldFont = msyhbd\l__ctex_msyh_suffix_tl ] { msyh\l__ctex_msyh_suffix_tl }
        \ctex_punct_map_family:nn { \CJKsfdefault } { zhyahei }
        \ctex_punct_map_bfseries:nn { \CJKsfdefault , zhyahei } { zhyaheib }
⟨/windowsnew⟩
        \setCJKmonofont { simfang.ttf }
        \setCJKfamilyfont { zhkai }  { simkai.ttf }
        \setCJKfamilyfont { zhfs }   { simfang.ttf }
        \setCJKfamilyfont { zhsong } { simsun.ttc }
        \setCJKfamilyfont { zhhei }  { simhei.ttf }
        \setCJKfamilyfont { zhli }   { simli.ttf }
        \setCJKfamilyfont { zhyou }  { simyou.ttf }
        \ctex_punct_map_family:nn { \CJKrmdefault } { zhsong }
        \ctex_punct_map_family:nn { \CJKttdefault } { zhfs }
        \ctex_punct_map_itshape:nn { \CJKrmdefault } { zhkai }
        \ctex_punct_map_bfseries:nn { \CJKrmdefault } { zhhei }
      }
```

```
      {
        \ctex_load_zhmap:nnnn { rm } { zhhei } { zhfs } { zhwindowsfonts }
        \ctex_punct_set:n { windows }
        \ctex_punct_map_family:nn { \CJKrmdefault } { zhsong }
        \ctex_punct_map_bfseries:nn { \CJKrmdefault } { zhhei }
        \ctex_punct_map_itshape:nn { \CJKrmdefault } { zhkai }
      }
      {
        \tl_set:Nn \CJKrmdefault { rm }
        \tl_set:Nn \CJKsfdefault { sf }
        \tl_set:Nn \CJKttdefault { tt }
      }
  }
  {
    \sys_if_engine_uptex:TF
      {
        \ctex_set_upfonts:nnnnnn
          {simsun.ttc} {simhei.ttf} {simkai.ttf}
          {msyh\l__ctex_msyh_suffix_tl} {msyhbd\l__ctex_msyh_suffix_tl}
          {simhei.ttf} {simhei.ttf}
          {simfang.ttf}
        \ctex_set_upfamily:nnn { zhsong } { upzhserif } {}
        \ctex_set_upfamily:nnn { zhhei } { upzhserifb } {}
        \ctex_set_upfamily:nnn { zhfs } { upzhmono} {}
        \ctex_set_upfamily:nnn { zhkai } { upzhserifit } {}
        \ctex_set_upfamily:nnn { zhyahei } { upzhsans } { upzhsansb }
        \ctex_set_upfamily:nnn { zhli } { upschrm } {}
        \ctex_set_upmap:nnn { upstsl } { simli.ttf } {}
        \ctex_set_upfamily:nnn { zhyou } { upschgt } {}
        \ctex_set_upmap:nnn { upstht } { simyou.ttf } {}
      }
      {
⟨*windowsold⟩
        \setCJKmainfont
          [ BoldFont = SimHei , ItalicFont = KaiTi_GB2312 ] { SimSun }
        \setCJKsansfont { SimHei }
        \setCJKmonofont { FangSong_GB2312 }
        \setCJKfamilyfont { zhkai } { KaiTi_GB2312 }
        \setCJKfamilyfont { zhfs }  { FangSong_GB2312 }
⟨/windowsold⟩
⟨*windowsnew⟩
        \setCJKmainfont
          [ BoldFont = SimHei , ItalicFont = KaiTi ] { SimSun }
        \setCJKsansfont
          [ BoldFont = { *~Bold } ] { Microsoft~YaHei }
        \setCJKmonofont { FangSong }
        \setCJKfamilyfont { zhkai } { KaiTi }
        \setCJKfamilyfont { zhfs }  { FangSong }
⟨/windowsnew⟩
        \setCJKfamilyfont { zhsong }  { SimSun }
        \setCJKfamilyfont { zhhei }   { SimHei }
        \setCJKfamilyfont { zhli }    { LiSu }
        \setCJKfamilyfont { zhyou }   { YouYuan }
⟨*windowsnew⟩
        \setCJKfamilyfont { zhyahei }
          [ BoldFont = { *~Bold } ] { Microsoft~YaHei }
⟨/windowsnew⟩
      }
  }
⟨/windowsnew|windowsold⟩
```

14.15.3 ctex-fontset-adobe.def

```
⟨*adobe⟩
```

\pdfmapline 不支持 OpenType 字体，因而 adobe 字体集在 pdf 模式下就没有定义。

fandol 的情况类似。

```
\sys_if_engine_pdftex:TF
  {
    \sys_if_output_pdf:TF
      { \ctex_fontset_error:n { adobe } }
      {
        \ctex_zhmap_case:nnn
          {
            \setCJKmainfont
              [
                      cmap = UniGB-UTF16-H ,
                  BoldFont = AdobeHeitiStd-Regular.otf ,
                ItalicFont = AdobeKaitiStd-Regular.otf
              ] { AdobeSongStd-Light.otf }
            \setCJKsansfont [ cmap = UniGB-UTF16-H ] { AdobeHeitiStd-Regular.otf }
            \setCJKmonofont [ cmap = UniGB-UTF16-H ] { AdobeFangsongStd-Regular.otf }
            \setCJKfamilyfont { zhsong }
              [ cmap = UniGB-UTF16-H ] { AdobeSongStd-Light.otf }
            \setCJKfamilyfont { zhhei }
              [ cmap = UniGB-UTF16-H ] { AdobeHeitiStd-Regular.otf }
            \setCJKfamilyfont { zhkai }
              [ cmap = UniGB-UTF16-H ] { AdobeKaitiStd-Regular.otf }
            \setCJKfamilyfont { zhfs }
              [ cmap = UniGB-UTF16-H ] { AdobeFangsongStd-Regular.otf }
            \ctex_punct_set:n { adobe }
            \ctex_punct_map_family:nn { \CJKrmdefault } { zhsong }
            \ctex_punct_map_family:nn { \CJKsfdefault } { zhhei }
            \ctex_punct_map_family:nn { \CJKttdefault } { zhfs }
            \ctex_punct_map_itshape:nn { \CJKrmdefault } { zhkai }
            \ctex_punct_map_bfseries:nn { \CJKrmdefault } { zhhei }
          }
          {
            \ctex_load_zhmap:nnnn { rm } { zhhei } { zhfs } { zhadobefonts }
            \ctex_punct_set:n { adobe }
            \ctex_punct_map_family:nn { \CJKrmdefault } { zhsong }
            \ctex_punct_map_bfseries:nn { \CJKrmdefault } { zhhei }
            \ctex_punct_map_itshape:nn { \CJKrmdefault } { zhkai }
          }
          { \ctex_fontset_error:n { adobe } }
      }
  }
  {
    \sys_if_engine_uptex:TF
      {
        \ctex_set_upfonts:nnnnnn
       {AdobeSongStd-Light.otf} {AdobeHeitiStd-Regular.otf} {AdobeKaitiStd-Regular.otf}
          {AdobeHeitiStd-Regular.otf} {AdobeHeitiStd-Regular.otf}
          {AdobeFangsongStd-Regular.otf}
        \ctex_set_upfamily:nnn { zhsong } { upzhserif } {}
        \ctex_set_upfamily:nnn { zhhei } { upzhsans } {}
        \ctex_set_upfamily:nnn { zhfs } { upzhmono} {}
        \ctex_set_upfamily:nnn { zhkai } { upzhserifit } {}
      }
      {
        \setCJKmainfont
          [
              BoldFont = AdobeHeitiStd-Regular ,
            ItalicFont = AdobeKaitiStd-Regular
          ] { AdobeSongStd-Light }
        \setCJKsansfont { AdobeHeitiStd-Regular}
        \setCJKmonofont { AdobeFangsongStd-Regular}
        \setCJKfamilyfont { zhsong } { AdobeSongStd-Light }
        \setCJKfamilyfont { zhhei }  { AdobeHeitiStd-Regular }
        \setCJKfamilyfont { zhfs }   { AdobeFangsongStd-Regular }
        \setCJKfamilyfont { zhkai }  { AdobeKaitiStd-Regular }
      }
  }
```

```
⟨/adobe⟩
⟨*fandol⟩
```

14.15.4 ctex-fontset-fandol.def

```
\sys_if_engine_pdftex:TF
  {
    \sys_if_output_pdf:TF
      { \ctex_fontset_error:n { fandol } }
      {
        \ctex_zhmap_case:nnn
          {
            \setCJKmainfont
              [
                      cmap = UniGB-UTF16-H ,
                  BoldFont = FandolSong-Bold.otf ,
                ItalicFont = FandolKai-Regular.otf
              ] { FandolSong-Regular.otf }
            \setCJKsansfont
              [
                    cmap = UniGB-UTF16-H ,
                BoldFont = FandolHei-Bold.otf
              ] { FandolHei-Regular.otf }
            \setCJKmonofont [ cmap = UniGB-UTF16-H ] { FandolFang-Regular.otf }
            \setCJKfamilyfont { zhsong }
              [
                    cmap = UniGB-UTF16-H ,
                BoldFont = FandolSong-Bold.otf
              ] { FandolSong-Regular.otf }
            \setCJKfamilyfont { zhhei }
              [
                    cmap = UniGB-UTF16-H ,
                BoldFont = FandolHei-Bold.otf
              ] { FandolHei-Regular.otf }
            \setCJKfamilyfont { zhfs }
              [ cmap = UniGB-UTF16-H ] { FandolFang-Regular.otf }
            \setCJKfamilyfont { zhkai }
              [ cmap = UniGB-UTF16-H ] { FandolKai-Regular.otf }
            \ctex_punct_set:n { fandol }
            \ctex_punct_map_family:nn { \CJKrmdefault } { zhsong }
            \ctex_punct_map_family:nn { \CJKsfdefault } { zhhei }
            \ctex_punct_map_family:nn { \CJKttdefault } { zhfs }
            \ctex_punct_map_itshape:nn { \CJKrmdefault } { zhkai }
            \ctex_punct_map_bfseries:nn { \CJKrmdefault , zhsong } { zhsongb }
            \ctex_punct_map_bfseries:nn { \CJKsfdefault , zhhei } { zhheib }
          }
          {
            \ctex_load_zhmap:nnnn { rm } { zhhei } { zhfs } { zhfandolfonts }
            \ctex_punct_set:n { fandol }
            \ctex_punct_map_family:nn { \CJKrmdefault } { zhsong }
            \ctex_punct_map_bfseries:nn { \CJKrmdefault } { zhhei }
            \ctex_punct_map_itshape:nn { \CJKrmdefault } { zhkai }
          }
          { \ctex_fontset_error:n { fandol } }
      }
  }
  {
    \sys_if_engine_uptex:TF
      {
        \ctex_set_upfonts:nnnnnn
          {FandolSong-Regular.otf} {FandolSong-Bold.otf} {FandolKai-Regular.otf}
          {FandolHei-Regular.otf} {FandolHei-Bold.otf}
          {FandolFang-Regular.otf}
        \ctex_set_upfamily:nnn { zhsong } { upzhserif } { upzhserifb }
        \ctex_set_upfamily:nnn { zhhei } { upzhsans } { upzhsansb }
        \ctex_set_upfamily:nnn { zhfs } { upzhmono} {}
        \ctex_set_upfamily:nnn { zhkai } { upzhserifit } {}
```

```
      }
      {
        \setCJKmainfont
          [
            Extension = .otf ,
            BoldFont = FandolSong-Bold , ItalicFont = FandolKai-Regular
          ]
          { FandolSong-Regular }
        \setCJKsansfont
          [ Extension = .otf , BoldFont = FandolHei-Bold ] { FandolHei-Regular }
        \setCJKmonofont [ Extension = .otf ] { FandolFang-Regular }
        \setCJKfamilyfont { zhsong }
          [ Extension = .otf , BoldFont = FandolSong-Bold ] { FandolSong-Regular }
        \setCJKfamilyfont { zhhei }
          [ Extension = .otf , BoldFont = FandolHei-Bold ] { FandolHei-Regular }
        \setCJKfamilyfont { zhfs }  [ Extension = .otf ] { FandolFang-Regular }
        \setCJKfamilyfont { zhkai } [ Extension = .otf ] { FandolKai-Regular }
      }
  }
⟨/fandol⟩
```

14.15.5 ctex-fontset-mac.def 等

按 Issue 351 的讨论，以 El Capitan 为分界，分别设置 macold（El Capitan 之前）和 macnew（El Capitan 及之后）。检测方式则以 El Capitan 及之后的苹方字体为准。

```
⟨*mac⟩
\file_if_exist:nTF { /System/Library/Fonts/PingFang.ttc }
  { \ctex_file_input:n { ctex-fontset-macnew.def } }
  { \ctex_file_input:n { ctex-fontset-macold.def } }
⟨/mac⟩
```

macold 的设置参考了 OS X Mavericks (10.9) 预装的主要简体中文字体列表。

在 DVIPDFM*x* 下，可以通过下述方式使用华文宋体和华文楷体：

```
\special{pdf:mapline unisong@Unicode@ unicode :4:Songti.ttc}
\special{pdf:mapline unikai@Unicode@  unicode :4:Kaiti.ttc}
```

而 \pdfmapline 似乎不支持带索引的 ttc 字体，Songti.ttc 默认使用的是 Songti SC Black，Kaiti.ttc 默认使用的是 Kaiti SC Black。华文黑体不能通过这种方式使用：

```
\special{pdf:mapline unihei@Unicode@ unicode \detokenize{华文黑体}.ttf}
```

DVIPDFM*x* 不能生成 PDF，报下述错误：

```
** WARNING ** UCS-4 TrueType cmap table...
** ERROR ** Unable to read OpenType/TrueType Unicode cmap table.
```

如果将 CMap 改为 UniGB-UTF16-H，错误信息是

```
** WARNING ** No usable TrueType cmap table found for font "华文黑体.ttf".
** WARNING ** CID character collection for this font is set to "Adobe-GB1"
** ERROR ** Cannot continue without this...
```

在 pdfTEX 下生成的 PDF 只有方框[22]。华文细黑和华文仿宋的情况类似。这确认为 DVIPDFM*x* 的 bug[23]。

macnew 在默认字体设置方面，引入了多字重的宋体作为罗马字族，以及引入了苹方黑体作为无衬线字族。

由于 Songti SC Light 的字重与 STSong 及 Windows 上的 SimSun 更接近，故默认字重使用 Songti SC Light，而不带后缀的正常字重事实上没有使用。下方关于 zhsong 的字族设

[22] http://www.newsmth.net/bbscon.php?bid=460&id=312640

[23] https://github.com/clerkma/ptex-ng/blob/master/texk/libdpx/cidtype2.c#L597

置也有同样的情况。

```
⟨*macold|macnew⟩
\sys_if_engine_pdftex:TF
  { \ctex_fontset_error:n { mac } }
  {
    \sys_if_engine_uptex:TF
      { \ctex_fontset_error:n { mac } }
      {
⟨*macold⟩
        \setCJKmainfont [ BoldFont = STHeiti , ItalicFont = STKaiti ] { STSong }
        \setCJKsansfont [ BoldFont = STHeiti ] { STXihei }
⟨/macold⟩
⟨*macnew⟩
        \setCJKmainfont
          [
               UprightFont = *~Light ,
                  BoldFont = *~Bold ,
                ItalicFont = Kaiti~SC ,
            BoldItalicFont = Kaiti~SC~Bold
          ] { Songti~SC }
        \setCJKsansfont { PingFang~SC }
⟨/macnew⟩
        \setCJKmonofont { STFangsong }
⟨*macold⟩
        \setCJKfamilyfont { zhsong } { STSong }
        \setCJKfamilyfont { zhhei }  { STHeiti }
⟨/macold⟩
⟨*macnew⟩
        \setCJKfamilyfont { zhsong }
          [
               UprightFont = *~Light ,
                  BoldFont = *~Bold ,
          ] { Songti~SC }
        \setCJKfamilyfont { zhhei }  { Heiti~SC }
        \setCJKfamilyfont { zhpf }   { PingFang~SC }
⟨/macnew⟩
        \setCJKfamilyfont { zhfs }   { STFangsong }
⟨macold⟩ \setCJKfamilyfont { zhkai }  { STKaiti }
⟨macnew⟩ \setCJKfamilyfont { zhkai }  { Kaiti~SC }
      }
  }
⟨/macold|macnew⟩
```

14.15.6 ctex-fontset-founder.def

```
⟨*founder⟩

\sys_if_engine_pdftex:TF
  {
    \ctex_zhmap_case:nnn
      {
        \setCJKmainfont
          [ BoldFont = FZXBSK.TTF , ItalicFont = FZKTK.TTF ] { FZSSK.TTF }
        \setCJKsansfont [ BoldFont = FZHTK.TTF ] { FZXH1K.TTF }
        \setCJKmonofont { FZFSK.TTF }
        \setCJKfamilyfont { zhsong } [ BoldFont = FZXBSK.TTF ] { FZSSK.TTF }
        \setCJKfamilyfont { zhhei }  { FZHTK.TTF }
        \setCJKfamilyfont { zhkai }  { FZKTK.TTF }
        \setCJKfamilyfont { zhfs }   { FZFSK.TTF }
        \setCJKfamilyfont { zhli }   { FZLSK.TTF }
        \setCJKfamilyfont { zhyou } [ BoldFont = FZY3K.TTF ] { FZY1K.TTF }
        \ctex_punct_set:n { founder }
        \ctex_punct_map_family:nn { \CJKrmdefault } { zhsong }
        \ctex_punct_map_family:nn { \CJKsfdefault } { zhheil }
        \ctex_punct_map_family:nn { \CJKttdefault } { zhfs }
        \ctex_punct_map_itshape:nn { \CJKrmdefault } { zhkai }
```

```
        \ctex_punct_map_bfseries:nn { \CJKrmdefault , zhsong } { zhsongb }
        \ctex_punct_map_bfseries:nn { \CJKsfdefault } { zhhei }
        \ctex_punct_map_bfseries:nn { zhyou } { zhyoub }
      }
      {
        \ctex_load_zhmap:nnnn { rm } { zhhei } { zhfs } { zhfounderfonts }
        \ctex_punct_set:n { founder }
        \ctex_punct_map_family:nn { \CJKrmdefault } { zhsong }
        \ctex_punct_map_bfseries:nn { \CJKrmdefault } { zhhei }
        \ctex_punct_map_itshape:nn { \CJKrmdefault } { zhkai }
      }
      { \ctex_fontset_error:n { founder } }
  }
  {
    \sys_if_engine_uptex:TF
      {
        \ctex_set_upfonts:nnnnnn
          { FZSSK.TTF } { FZXBSK.TTF } { FZKTK.TTF }
          { FZXH1K.TTF } { FZHTK.TTF }
          { FZFSK.TTF }
        \ctex_set_upfamily:nnn { zhsong } { upzhserif } { upzhserifb }
        \ctex_set_upfamily:nnn { zhhei } { upzhsans } { upzhsansb }
        \ctex_set_upfamily:nnn { zhfs } { upzhmono} {}
        \ctex_set_upfamily:nnn { zhkai } { upzhserifit } {}
        \ctex_set_upfamily:nnn { zhli } { upschrm } {}
        \ctex_set_upmap:nnn { upstsl } { FZLSK.TTF } {}
        \ctex_set_upfamily:nnn { zhyou } { upschgt } {}
        \ctex_set_upmap:nnn { upstht } { FZY1K.TTF } {}
      }
      {
        \setCJKmainfont
          [ BoldFont = FZXiaoBiaoSong-B05 , ItalicFont = FZKai-Z03 ] { FZShuSong-Z01 }
        \setCJKsansfont [ BoldFont = FZHei-B01 ] { FZXiHeiI-Z08 }
        \setCJKmonofont { FZFangSong-Z02 }
       \setCJKfamilyfont { zhsong } [ BoldFont = FZXiaoBiaoSong-B05 ] { FZShuSong-Z01 }
        \setCJKfamilyfont { zhhei }  { FZHei-B01 }
        \setCJKfamilyfont { zhkai }  { FZKai-Z03 }
        \setCJKfamilyfont { zhfs }   { FZFangSong-Z02 }
        \setCJKfamilyfont { zhli }   { FZLiShu-S01 }
        \setCJKfamilyfont { zhyou } [ BoldFont = FZZhunYuan-M02 ] { FZXiYuan-M01 }
      }
  }
⟨/founder⟩
⟨*ubuntu⟩
```

14.15.7 ctex-fontset-ubuntu.def

以下根据 Ubuntu 12.04 的中文字体情况设置。CMap 不清楚应该是什么，指定为 UniGB-UTF16-H 还是有警告：

```
** WARNING ** UCS-4 TrueType cmap table...
```

需要注意的是 `uming.ttc` 和 `ukai.ttc` 看起来像有四种字形的样子，但其实只有“令”和“骨”这区区两个字有新字形，其余都取旧字形[24]。

```
\sys_if_engine_pdftex:TF
  {
    \ctex_zhmap_case:nnn
      {
        \setCJKmainfont
          [ BoldFont = wqy-zenhei.ttc , ItalicFont = ukai.ttc ] { uming.ttc }
        \setCJKsansfont { wqy-zenhei.ttc }
        \setCJKmonofont { uming.ttc }
        \setCJKfamilyfont { zhsong } { uming.ttc }
        \setCJKfamilyfont { zhhei }  { wqy-zenhei.ttc }
```

[24] http://www.freedesktop.org/wiki/Software/CJKUnifonts/Download/

```
        \setCJKfamilyfont { zhkai }  { ukai.ttc }
        \ctex_punct_set:n { ubuntu }
        \ctex_punct_map_family:nn { \CJKrmdefault } { zhsong }
        \ctex_punct_map_family:nn { \CJKsfdefault } { zhhei }
        \ctex_punct_map_family:nn { \CJKttdefault } { zhsong }
        \ctex_punct_map_itshape:nn { \CJKrmdefault } { zhkai }
        \ctex_punct_map_bfseries:nn { \CJKrmdefault } { zhhei }
      }
      {
        \ctex_load_zhmap:nnnn { rm } { zhhei } { zhsong } { zhubuntufonts }
        \ctex_punct_set:n { ubuntu }
        \ctex_punct_map_family:nn { \CJKrmdefault } { zhsong }
        \ctex_punct_map_bfseries:nn { \CJKrmdefault } { zhhei }
        \ctex_punct_map_itshape:nn { \CJKrmdefault } { zhkai }
      }
      { \ctex_fontset_error:n { ubuntu } }
  }
  {
    \sys_if_engine_uptex:TF
      {
        \ctex_set_upfonts:nnnnnn
          { uming.ttc } { wqy-zenhei.ttc } { ukai.ttc }
          { wqy-zenhei.ttc } { wqy-zenhei.ttc }
          { uming.ttc }
        \ctex_set_upfamily:nnn { zhsong } { upzhserif } {}
        \ctex_set_upfamily:nnn { zhhei } { upzhsans } {}
        \ctex_set_upfamily:nnn { zhkai } { upzhserifit } {}
      }
      {
        \setCJKmainfont
        [ BoldFont = WenQuanYi~Zen~Hei , ItalicFont = AR~PL~UKai~CN ] { AR~PL~UMing~CN }
        \setCJKsansfont { WenQuanYi~Zen~Hei }
        \setCJKmonofont { AR~PL~UMing~CN }
        \setCJKfamilyfont { zhsong } { AR~PL~UMing~CN }
        \setCJKfamilyfont { zhhei }  { WenQuanYi~Zen~Hei }
        \setCJKfamilyfont { zhkai }  { AR~PL~UKai~CN }
      }
  }
```

⟨/ubuntu⟩

14.15.8　中文字体命令

```
⟨*!windows&!mac⟩
\NewDocumentCommand \songti   { } { \CJKfamily { zhsong } }
\NewDocumentCommand \heiti    { } { \CJKfamily { zhhei } }
⟨!ubuntu⟩ \NewDocumentCommand \fangsong { } { \CJKfamily { zhfs } }
\NewDocumentCommand \kaishu   { } { \CJKfamily { zhkai } }
⟨*windowsnew|windowsold|founder⟩
\NewDocumentCommand \lishu    { } { \CJKfamily { zhli } }
\NewDocumentCommand \youyuan  { } { \CJKfamily { zhyou } }
⟨/windowsnew|windowsold|founder⟩
⟨windowsnew⟩ \NewDocumentCommand \yahei    { } { \CJKfamily { zhyahei } }
⟨*macnew⟩
\NewDocumentCommand \yahei    { } { \CJKfamily { zhpf } }
\NewDocumentCommand \pingfang { } { \CJKfamily { zhpf } }
⟨/macnew⟩
⟨/!windows&!mac⟩
⟨/fontset⟩
⟨*zhmap⟩
```

14.15.9　zhmetrics 的字体映射

确认 \catcode，没有重复载入检查。

```
\begingroup\catcode61\catcode48\catcode32=10\relax%
```

```
  \catcode 35=6 % #
  \catcode123=1 % {
  \catcode125=2 % }
  \toks0{\endlinechar=\the\endlinechar\relax}%
  \toks2{\endlinechar=13 }%
  \def\x#1 #2 {%
    \toks0\expandafter{\the\toks0 \catcode#1=\the\catcode#1\relax}%
    \toks2\expandafter{\the\toks2 \catcode#1=#2 }}%
  \x  13  5 % carriage return
  \x  32 10 % space
  \x  35  6 % #
  \x  40 12 % (
  \x  41 12 % )
  \x  45 12 % -
  \x  46 12 % .
  \x  47 12 % /
  \x  58 12 % :
  \x  60 12 % <
  \x  61 12 % =
  \x  64 11 % @
  \x  91 12 % [
  \x  93 12 % ]
  \x 123  1 % {
  \x 125  2 % }
  \edef\x#1{\endgroup%
    \edef\noexpand#1{%
      \the\toks0 %
      \let\noexpand\noexpand\noexpand#1%
          \noexpand\noexpand\noexpand\undefined%
      \noexpand\noexpand\noexpand\endinput}%
    \the\toks2}%
\expandafter\x\csname ctex@zhmap@endinput\endcsname
```

```
\begingroup\expandafter\endgroup
\expandafter\let\csname ifzhmappdf\expandafter\endcsname\csname
  \expandafter\ifx\csname ifctexpdf\endcsname\relax
    \expandafter\ifx\csname pdfoutput\endcsname\relax
      iffalse\else\ifnum\pdfoutput < 1 iffalse\else iftrue\fi\fi
  \else ifctexpdf\fi
\endcsname
```

提供非 LaTeX 格式下的 \ProvidesFile。

```
\begingroup
\expandafter\ifx\csname ProvidesFile\endcsname\relax
  \long\def\x#1\ProvidesFile#2[#3]{%
    #1%
    \immediate\write-1{File: #2 #3}%
    \expandafter\xdef\csname ver@#2\endcsname{#3}}
  \expandafter\x%
\fi
\endgroup
```

14.15.9.1 zhwindowsfonts.tex

```
⟨*windows⟩
\ProvidesFile{zhwindowsfonts.tex}%
  [2019/05/29 v2.4.16 Windows font map loader for pdfTeX and DVIPDFMx (CTEX)]

\ifzhmappdf
  \pdfmapline{=gbk@UGBK@     <simsun.ttc}
  \pdfmapline{=gbksong@UGBK@ <simsun.ttc}
  \pdfmapline{=gbkkai@UGBK@  <simkai.ttf}
  \pdfmapline{=gbkhei@UGBK@  <simhei.ttf}
  \pdfmapline{=gbkfs@UGBK@   <simfang.ttf}
  \pdfmapline{=gbkli@UGBK@   <simli.ttf}
  \pdfmapline{=gbkyou@UGBK@  <simyou.ttf}
```

```
  \pdfmapline{=cyberb@Unicode@  <simsun.ttc}
  \pdfmapline{=unisong@Unicode@ <simsun.ttc}
  \pdfmapline{=unikai@Unicode@  <simkai.ttf}
  \pdfmapline{=unihei@Unicode@  <simhei.ttf}
  \pdfmapline{=unifs@Unicode@   <simfang.ttf}
  \pdfmapline{=unili@Unicode@   <simli.ttf}
  \pdfmapline{=uniyou@Unicode@  <simyou.ttf}

  \pdfmapline{=gbksongsl@UGBK@ <simsun.ttc}
  \pdfmapline{=gbkkaisl@UGBK@  <simkai.ttf}
  \pdfmapline{=gbkheisl@UGBK@  <simhei.ttf}
  \pdfmapline{=gbkfssl@UGBK@   <simfang.ttf}
  \pdfmapline{=gbklisl@UGBK@   <simli.ttf}
  \pdfmapline{=gbkyousl@UGBK@  <simyou.ttf}

  \pdfmapline{=unisongsl@Unicode@ <simsun.ttc}
  \pdfmapline{=unikaisl@Unicode@  <simkai.ttf}
  \pdfmapline{=uniheisl@Unicode@  <simhei.ttf}
  \pdfmapline{=unifssl@Unicode@   <simfang.ttf}
  \pdfmapline{=unilisl@Unicode@   <simli.ttf}
  \pdfmapline{=uniyousl@Unicode@  <simyou.ttf}

\else
  \special{pdf:mapline gbk@UGBK@     unicode :0:simsun.ttc -v 50}
  \special{pdf:mapline gbksong@UGBK@ unicode :0:simsun.ttc -v 50}
  \special{pdf:mapline gbkkai@UGBK@  unicode simkai.ttf -v 70}
  \special{pdf:mapline gbkhei@UGBK@  unicode simhei.ttf -v 150}
  \special{pdf:mapline gbkfs@UGBK@   unicode simfang.ttf -v 50}
  \special{pdf:mapline gbkli@UGBK@   unicode simli.ttf -v 150}
  \special{pdf:mapline gbkyou@UGBK@  unicode simyou.ttf -v 60}

  \special{pdf:mapline cyberb@Unicode@  unicode :0:simsun.ttc -v 50}
  \special{pdf:mapline unisong@Unicode@ unicode :0:simsun.ttc -v 50}
  \special{pdf:mapline unikai@Unicode@  unicode simkai.ttf -v 70}
  \special{pdf:mapline unihei@Unicode@  unicode simhei.ttf -v 150}
  \special{pdf:mapline unifs@Unicode@   unicode simfang.ttf -v 50}
  \special{pdf:mapline unili@Unicode@   unicode simli.ttf -v 150}
  \special{pdf:mapline uniyou@Unicode@  unicode simyou.ttf -v 60}

  \special{pdf:mapline gbksongsl@UGBK@ unicode :0:simsun.ttc -s .167 -v 50}
  \special{pdf:mapline gbkkaisl@UGBK@  unicode simkai.ttf -s .167 -v 70}
  \special{pdf:mapline gbkheisl@UGBK@  unicode simhei.ttf -s .167 -v 150}
  \special{pdf:mapline gbkfssl@UGBK@   unicode simfang.ttf -s .167 -v 50}
  \special{pdf:mapline gbklisl@UGBK@   unicode simli.ttf -s .167 -v 150}
  \special{pdf:mapline gbkyousl@UGBK@  unicode simyou.ttf -s .167 -v 60}

  \special{pdf:mapline unisongsl@Unicode@ unicode :0:simsun.ttc -s .167 -v 50}
  \special{pdf:mapline unikaisl@Unicode@  unicode simkai.ttf -s .167 -v 70}
  \special{pdf:mapline uniheisl@Unicode@  unicode simhei.ttf -s .167 -v 150}
  \special{pdf:mapline unifssl@Unicode@   unicode simfang.ttf -s .167 -v 50}
  \special{pdf:mapline unilisl@Unicode@   unicode simli.ttf -s .167 -v 150}
  \special{pdf:mapline uniyousl@Unicode@  unicode simyou.ttf -s .167 -v 60}

⟨/windows⟩
```

14.15.9.2 zhadobefonts.tex

```
⟨*adobe⟩
\ProvidesFile{zhadobefonts.tex}%
  [2019/05/29 v2.4.16 Adobe font map loader for DVIPDFMx (CTEX)]

\ifzhmappdf
%% pdfTeX does not support OTF fonts

\else
  \special{pdf:mapline gbk@UGBK@     UniGB-UTF16-H AdobeSongStd-Light.otf}
  \special{pdf:mapline gbksong@UGBK@ UniGB-UTF16-H AdobeSongStd-Light.otf}
```

```
  \special{pdf:mapline gbkkai@UGBK@  UniGB-UTF16-H AdobeKaitiStd-Regular.otf}
  \special{pdf:mapline gbkhei@UGBK@  UniGB-UTF16-H AdobeHeitiStd-Regular.otf}
  \special{pdf:mapline gbkfs@UGBK@   UniGB-UTF16-H AdobeFangsongStd-Regular.otf}

  \special{pdf:mapline cyberb@Unicode@  UniGB-UTF16-H AdobeSongStd-Light.otf}
  \special{pdf:mapline unisong@Unicode@ UniGB-UTF16-H AdobeSongStd-Light.otf}
  \special{pdf:mapline unikai@Unicode@  UniGB-UTF16-H AdobeKaitiStd-Regular.otf}
  \special{pdf:mapline unihei@Unicode@  UniGB-UTF16-H AdobeHeitiStd-Regular.otf}
  \special{pdf:mapline unifs@Unicode@   UniGB-UTF16-H AdobeFangsongStd-Regular.otf}

  \special{pdf:mapline gbksongsl@UGBK@ UniGB-UTF16-H AdobeSongStd-Light.otf -s .167}
  \special{pdf:mapline gbkkaisl@UGBK@  UniGB-UTF16-H AdobeKaitiStd-Regular.otf -s .167}
  \special{pdf:mapline gbkheisl@UGBK@  UniGB-UTF16-H AdobeHeitiStd-Regular.otf -s .167}
  \special{pdf:mapline gbkfssl@UGBK@  UniGB-UTF16-H AdobeFangsongStd-Regular.otf -s .167}

  \special{pdf:mapline unisongsl@Unicode@ UniGB-UTF16-H AdobeSongStd-Light.otf -s .167}
  \special{pdf:mapline unikaisl@Unicode@  UniGB-UTF16-H AdobeKaitiStd-Regular.otf -s .167}
  \special{pdf:mapline uniheisl@Unicode@  UniGB-UTF16-H AdobeHeitiStd-Regular.otf -s .167}
  \special{pdf:mapline unifssl@Unicode@  UniGB-UTF16-H AdobeFangsongStd-Regular.otf -s .167}

⟨/adobe⟩
```

14.15.9.3 zhfandolfonts.tex

```
⟨*fandol⟩
\ProvidesFile{zhfandolfonts.tex}%
  [2019/05/29 v2.4.16 Fandol font map loader for DVIPDFMx (CTEX)]

\ifzhmappdf
%% pdfTeX does not support OTF fonts

\else
  \special{pdf:mapline gbk@UGBK@     UniGB-UTF16-H FandolSong-Regular.otf}
  \special{pdf:mapline gbksong@UGBK@ UniGB-UTF16-H FandolSong-Regular.otf}
  \special{pdf:mapline gbkkai@UGBK@  UniGB-UTF16-H FandolKai-Regular.otf}
  \special{pdf:mapline gbkhei@UGBK@  UniGB-UTF16-H FandolHei-Regular.otf}
  \special{pdf:mapline gbkfs@UGBK@   UniGB-UTF16-H FandolFang-Regular.otf}

  \special{pdf:mapline cyberb@Unicode@  UniGB-UTF16-H FandolSong-Regular.otf}
  \special{pdf:mapline unisong@Unicode@ UniGB-UTF16-H FandolSong-Regular.otf}
  \special{pdf:mapline unikai@Unicode@  UniGB-UTF16-H FandolKai-Regular.otf}
  \special{pdf:mapline unihei@Unicode@  UniGB-UTF16-H FandolHei-Regular.otf}
  \special{pdf:mapline unifs@Unicode@   UniGB-UTF16-H FandolFang-Regular.otf}

  \special{pdf:mapline gbksongsl@UGBK@ UniGB-UTF16-H FandolSong-Regular.otf -s .167}
  \special{pdf:mapline gbkkaisl@UGBK@  UniGB-UTF16-H FandolKai-Regular.otf -s .167}
  \special{pdf:mapline gbkheisl@UGBK@  UniGB-UTF16-H FandolHei-Regular.otf -s .167}
  \special{pdf:mapline gbkfssl@UGBK@   UniGB-UTF16-H FandolFang-Regular.otf -s .167}

  \special{pdf:mapline unisongsl@Unicode@ UniGB-UTF16-H FandolSong-Regular.otf -s .167}
  \special{pdf:mapline unikaisl@Unicode@  UniGB-UTF16-H FandolKai-Regular.otf -s .167}
  \special{pdf:mapline uniheisl@Unicode@  UniGB-UTF16-H FandolHei-Regular.otf -s .167}
  \special{pdf:mapline unifssl@Unicode@  UniGB-UTF16-H FandolFang-Regular.otf -s .167}

⟨/fandol⟩
```

14.15.9.4 zhfounderfonts.tex

```
⟨*founder⟩
\ProvidesFile{zhfounderfonts.tex}%
  [2019/05/29 v2.4.16 Founder font map loader for pdfTeX and DVIPDFMx (CTEX)]

\ifzhmappdf
  \pdfmapline{=gbk@UGBK@     <FZSSK.TTF}
  \pdfmapline{=gbksong@UGBK@ <FZSSK.TTF}
  \pdfmapline{=gbkkai@UGBK@  <FZKTK.TTF}
  \pdfmapline{=gbkhei@UGBK@  <FZHTK.TTF}
  \pdfmapline{=gbkfs@UGBK@   <FZFSK.TTF}
```

```
  \pdfmapline{=gbkli@UGBK@   <FZLSK.TTF}
  \pdfmapline{=gbkyou@UGBK@  <FZY1K.TTF}

  \pdfmapline{=cyberb@Unicode@  <FZSSK.TTF}
  \pdfmapline{=unisong@Unicode@ <FZSSK.TTF}
  \pdfmapline{=unikai@Unicode@  <FZKTK.TTF}
  \pdfmapline{=unihei@Unicode@  <FZHTK.TTF}
  \pdfmapline{=unifs@Unicode@   <FZFSK.TTF}
  \pdfmapline{=unili@Unicode@   <FZLSK.TTF}
  \pdfmapline{=uniyou@Unicode@  <FZY1K.TTF}

  \pdfmapline{=gbksongsl@UGBK@ <FZSSK.TTF}
  \pdfmapline{=gbkkaisl@UGBK@  <FZKTK.TTF}
  \pdfmapline{=gbkheisl@UGBK@  <FZHTK.TTF}
  \pdfmapline{=gbkfssl@UGBK@   <FZFSK.TTF}
  \pdfmapline{=gbklisl@UGBK@   <FZLSK.TTF}
  \pdfmapline{=gbkyousl@UGBK@  <FZY1K.TTF}

  \pdfmapline{=unisongsl@Unicode@ <FZSSK.TTF}
  \pdfmapline{=unikaisl@Unicode@  <FZKTK.TTF}
  \pdfmapline{=uniheisl@Unicode@  <FZHTK.TTF}
  \pdfmapline{=unifssl@Unicode@   <FZFSK.TTF}
  \pdfmapline{=unilisl@Unicode@   <FZLSK.TTF}
  \pdfmapline{=uniyousl@Unicode@  <FZY1K.TTF}

\else
  \special{pdf:mapline gbk@UGBK@     unicode FZSSK.TTF}
  \special{pdf:mapline gbksong@UGBK@ unicode FZSSK.TTF}
  \special{pdf:mapline gbkkai@UGBK@  unicode FZKTK.TTF}
  \special{pdf:mapline gbkhei@UGBK@  unicode FZHTK.TTF}
  \special{pdf:mapline gbkfs@UGBK@   unicode FZFSK.TTF}
  \special{pdf:mapline gbkli@UGBK@   unicode FZLSK.TTF}
  \special{pdf:mapline gbkyou@UGBK@  unicode FZY1K.TTF}

  \special{pdf:mapline cyberb@Unicode@  unicode FZSSK.TTF}
  \special{pdf:mapline unisong@Unicode@ unicode FZSSK.TTF}
  \special{pdf:mapline unikai@Unicode@  unicode FZKTK.TTF}
  \special{pdf:mapline unihei@Unicode@  unicode FZHTK.TTF}
  \special{pdf:mapline unifs@Unicode@   unicode FZFSK.TTF}
  \special{pdf:mapline unili@Unicode@   unicode FZLSK.TTF}
  \special{pdf:mapline uniyou@Unicode@  unicode FZY1K.TTF}

  \special{pdf:mapline gbksongsl@UGBK@ unicode FZSSK.TTF -s .167}
  \special{pdf:mapline gbkkaisl@UGBK@  unicode FZKTK.TTF -s .167}
  \special{pdf:mapline gbkheisl@UGBK@  unicode FZHTK.TTF -s .167}
  \special{pdf:mapline gbkfssl@UGBK@   unicode FZFSK.TTF -s .167}
  \special{pdf:mapline gbklisl@UGBK@   unicode FZLSK.TTF -s .167}
  \special{pdf:mapline gbkyousl@UGBK@  unicode FZY1K.TTF -s .167}

  \special{pdf:mapline unisongsl@Unicode@ unicode FZSSK.TTF -s .167}
  \special{pdf:mapline unikaisl@Unicode@  unicode FZKTK.TTF -s .167}
  \special{pdf:mapline uniheisl@Unicode@  unicode FZHTK.TTF -s .167}
  \special{pdf:mapline unifssl@Unicode@   unicode FZFSK.TTF -s .167}
  \special{pdf:mapline unilisl@Unicode@   unicode FZLSK.TTF -s .167}
  \special{pdf:mapline uniyousl@Unicode@  unicode FZY1K.TTF -s .167}

⟨/founder⟩
```

14.15.9.5 zhubuntufonts.tex

```
⟨*ubuntu⟩
\ProvidesFile{zhubuntufonts.tex}%
  [2019/05/29 v2.4.16 Ubuntu font map loader for pdfTeX and DVIPDFMx (CTEX)]

\ifzhmappdf
  \pdfmapline{=gbk@UGBK@     <uming.ttc}
  \pdfmapline{=gbksong@UGBK@ <uming.ttc}
  \pdfmapline{=gbkkai@UGBK@  <ukai.ttc}
```

```
  \pdfmapline{=gbkhei@UGBK@  <wqy-zenhei.ttc}
  \pdfmapline{=gbkfs@UGBK@   <uming.ttc}
  \pdfmapline{=gbkyou@UGBK@  <wqy-zenhei.ttc}

  \pdfmapline{=cyberb@Unicode@  <uming.ttc}
  \pdfmapline{=unisong@Unicode@ <uming.ttc}
  \pdfmapline{=unikai@Unicode@  <ukai.ttc}
  \pdfmapline{=unihei@Unicode@  <wqy-zenhei.ttc}
  \pdfmapline{=unifs@Unicode@   <uming.ttc}
  \pdfmapline{=uniyou@Unicode@  <wqy-zenhei.ttc}

  \pdfmapline{=gbksongsl@UGBK@ <uming.ttc}
  \pdfmapline{=gbkkaisl@UGBK@  <ukai.ttc}
  \pdfmapline{=gbkheisl@UGBK@  <wqy-zenhei.ttc}
  \pdfmapline{=gbkfssl@UGBK@   <uming.ttc}
  \pdfmapline{=gbkyousl@UGBK@  <wqy-zenhei.ttc}

  \pdfmapline{=unisongsl@Unicode@ <uming.ttc}
  \pdfmapline{=unikaisl@Unicode@  <ukai.ttc}
  \pdfmapline{=uniheisl@Unicode@  <wqy-zenhei.ttc}
  \pdfmapline{=unifssl@Unicode@   <uming.ttc}
  \pdfmapline{=uniyousl@Unicode@  <wqy-zenhei.ttc}

\else
  \special{pdf:mapline gbk@UGBK@     unicode :0:uming.ttc}
  \special{pdf:mapline gbksong@UGBK@ unicode :0:uming.ttc}
  \special{pdf:mapline gbkkai@UGBK@  unicode :0:ukai.ttc}
  \special{pdf:mapline gbkhei@UGBK@  unicode :0:wqy-zenhei.ttc}
  \special{pdf:mapline gbkfs@UGBK@   unicode :0:uming.ttc}

  \special{pdf:mapline cyberb@Unicode@  unicode :0:uming.ttc}
  \special{pdf:mapline unisong@Unicode@ unicode :0:uming.ttc}
  \special{pdf:mapline unikai@Unicode@  unicode :0:ukai.ttc}
  \special{pdf:mapline unihei@Unicode@  unicode :0:wqy-zenhei.ttc}
  \special{pdf:mapline unifs@Unicode@   unicode :0:uming.ttc}

  \special{pdf:mapline gbksongsl@UGBK@ unicode :0:uming.ttc -s .167}
  \special{pdf:mapline gbkkaisl@UGBK@  unicode :0:ukai.ttc -s .167}
  \special{pdf:mapline gbkheisl@UGBK@  unicode :0:wqy-zenhei.ttc -s .167}
  \special{pdf:mapline gbkfssl@UGBK@   unicode :0:uming.ttc -s .167}

  \special{pdf:mapline unisongsl@Unicode@ unicode :0:uming.ttc -s .167}
  \special{pdf:mapline unikaisl@Unicode@  unicode :0:ukai.ttc -s .167}
  \special{pdf:mapline uniheisl@Unicode@  unicode :0:wqy-zenhei.ttc -s .167}
  \special{pdf:mapline unifssl@Unicode@   unicode :0:uming.ttc -s .167}

⟨/ubuntu⟩
\fi

\ctex@zhmap@endinput
⟨/zhmap⟩
```

14.15.10　制作 spa 文件

我们通过 XeTeX 的 \XeTeXglyphbounds 取得字体中标点符号的边界信息，为 CJKpunct 宏包制作 spa。

```
⟨*spa⟩
⟨*macro⟩
\input expl3-generic %
\ExplSyntaxOn
\sys_if_engine_xetex:F
  {
    \msg_new:nn { ctex } { xetex }
      { XeTeX~is~required~to~compile~this~document! }
    \msg_fatal:nn { ctex } { xetex }
```

```
}
```

CJKpunct 定义的标点符号是：

‘ “ 「 『 （ （ [{ 〈 《 〖 【
—… 、 。 ， . ： ； ！ ？ ％ ） ）] } 〉 》 〗 】 ’ ” 」 』

注意顺序不能改变。

```
\seq_const_from_clist:Nn \c__ctex_punct_seq
  {
    "2018 , "201C , "300C , "300E , "3014 , "FF08 , "FF3B , "FF5B ,
    "3008 , "300A , "3016 , "3010 ,
    "2014 , "2026 , "3001 , "3002 , "FF0C , "FF0E , "FF1A , "FF1B ,
    "FF01 , "FF1F , "FF05 , "3015 , "FF09 , "FF3D , "FF5D , "3009 ,
    "300B , "3017 , "3011 , "2019 , "201D , "300D , "300F
  }
```

\ctex_make_spa:nn　#1 是 spa 文件名，#2 是由 CJK 族名与字体构成的逗号列表。

```
\cs_new_protected_nopar:Npn \ctex_make_spa:nn #1#2
  {
    \iow_open:Nn \g__ctex_spa_iow {#1}
    \clist_map_inline:nn {#2}
      { \__ctex_write_family:nn ##1 }
    \iow_close:N \g__ctex_spa_iow
  }
\iow_new:N \g__ctex_spa_iow
\cs_new_eq:NN \MAKESPA \ctex_make_spa:nn
```

```
\cs_new_protected:Npn \__ctex_write_family:nn #1#2
  {
    \group_begin:
      \tex_font:D \l__ctex_punct_font = "#2" ~ at ~ 100 pt \scan_stop:
      \l__ctex_punct_font
      \clist_clear:N \l__ctex_punct_bounds_clist
      \seq_map_inline:Nn \c__ctex_punct_seq
       { \exp_args:No \__ctex_save_bounds:n { \int_use:N \tex_XeTeXcharglyph:D ##1 } }
      \iow_now:Nx \g__ctex_spa_iow
        {
          \token_to_str:N \ctexspadef {#1}
```

最后这三个逗号对 CJKpunct 来说是必要的。

```
            { \l__ctex_punct_bounds_clist , , , }
        }
    \group_end:
  }
\cs_new_protected_nopar:Npn \__ctex_save_bounds:n #1
  {
    \clist_put_right:Nx \l__ctex_punct_bounds_clist
      {
        \__ctex_calc_bounds:nn { 1 } {#1} ,
        \__ctex_calc_bounds:nn { 3 } {#1}
      }
  }
\clist_new:N \l__ctex_punct_bounds_clist
```

CJKpunct 要求的格式是边界空白宽度与 1 em 的比值的一百倍。

```
\cs_new_nopar:Npn \__ctex_calc_bounds:nn #1#2
  {
    \fp_eval:n
      {
        round
          (
            \dim_to_decimal_in_unit:nn
              { 100 \tex_XeTeXglyphbounds:D #1 ~ #2 }
              { 1 em }
```

```
          )
      }
  }
\ExplSyntaxOff
⟨/macro⟩
```

下面是 CTEX 定义的一些字体。

```
⟨*make⟩
\input ctexspamacro %

\MAKESPA {ctexpunct.spa}
  {
    {adobezhsong}     {AdobeSongStd-Light} ,
    {adobezhhei}      {AdobeHeitiStd-Regular} ,
    {adobezhkai}      {AdobeKaitiStd-Regular} ,
    {adobezhfs}       {AdobeFangsongStd-Regular} ,
    {fandolzhsong}    {FandolSong} ,
    {fandolzhsongb}   {FandolSong-Bold} ,
    {fandolzhhei}     {FandolHei} ,
    {fandolzhheib}    {FandolHei-Bold} ,
    {fandolzhkai}     {FandolKai} ,
    {fandolzhfs}      {FandolFang} ,
    {founderzhsong}   {FZShuSong-Z01} ,
    {founderzhsongb}  {FZXiaoBiaoSong-B05} ,
    {founderzhhei}    {FZHei-B01} ,
    {founderzhheil}   {FZXiHeiI-Z08} ,
    {founderzhkai}    {FZKai-Z03} ,
    {founderzhfs}     {FZFangSong-Z02} ,
    {founderzhli}     {FZLiShu-S01} ,
    {founderzhyou}    {FZXiYuan-M01} ,
    {founderzhyoub}   {FZZhunYuan-M02} ,
    {ubuntuzhsong}    {AR PL UMing CN} ,
    {ubuntuzhhei}     {WenQuanYi Zen Hei} ,
    {ubuntuzhkai}     {AR PL UKai CN} ,
    {windowszhsong}   {SimSun} ,
    {windowszhhei}    {SimHei} ,
    {windowszhkai}    {KaiTi} ,
    {windowszhfs}     {FangSong} ,
    {windowszhli}     {LiSu} ,
    {windowszhyou}    {YouYuan} ,
    {windowszhyahei}  {Microsoft YaHei} ,
    {windowszhyaheib} {Microsoft YaHei Bold}
  }

\primitive\end
⟨/make⟩
⟨/spa⟩
```

14.16 translator 宏包的中文字典

```
⟨*dict⟩
```

包括 ChineseGBK 和 ChineseUTF8 两种形式，目前只翻译 beamer 宏包需要的定理环境名称。

```
⟨*theorem⟩
\providetranslation{Comments}{评论}
\providetranslation{comments}{评论}
\providetranslation{Comment}{评论}
\providetranslation{comment}{评论}
\providetranslation{Corollaries}{推论}
\providetranslation{corollaries}{推论}
\providetranslation{Corollary}{推论}
\providetranslation{corollary}{推论}
\providetranslation{Definitions}{定义}
\providetranslation{definitions}{定义}
```

```
\providetranslation{Definition}{定义}
\providetranslation{definition}{定义}
\providetranslation{Examples}{例}
\providetranslation{examples}{例}
\providetranslation{Example}{例}
\providetranslation{example}{例}
\providetranslation{Exercises}{练习}
\providetranslation{exercises}{练习}
\providetranslation{Exercise}{练习}
\providetranslation{exercise}{练习}
\providetranslation{Facts}{事实}
\providetranslation{facts}{事实}
\providetranslation{Fact}{事实}
\providetranslation{fact}{事实}
\providetranslation{Key Lemmas}{关键引理}
\providetranslation{key lemmas}{关键引理}
\providetranslation{Key Lemma}{关键引理}
\providetranslation{key lemma}{关键引理}
\providetranslation{Key Observations}{关键观察}
\providetranslation{key observations}{关键观察}
\providetranslation{Key Observation}{关键观察}
\providetranslation{key observation}{关键观察}
\providetranslation{Lemmas}{引理}
\providetranslation{lemmas}{引理}
\providetranslation{Lemma}{引理}
\providetranslation{lemma}{引理}
\providetranslation{Main Theorems}{主要定理}
\providetranslation{main theorems}{主要定理}
\providetranslation{Main Theorem}{主要定理}
\providetranslation{main theorem}{主要定理}
\providetranslation{Observations}{观察}
\providetranslation{observations}{观察}
\providetranslation{Observation}{观察}
\providetranslation{observation}{观察}
\providetranslation{Problems}{问题}
\providetranslation{problems}{问题}
\providetranslation{Problem}{问题}
\providetranslation{problem}{问题}
\providetranslation{Proofs}{证明}
\providetranslation{proofs}{证明}
\providetranslation{Proof}{证明}
\providetranslation{proof}{证明}
\providetranslation{Proof Sketch}{证明提要}
\providetranslation{Proof sketch}{证明提要}
\providetranslation{proof sketch}{证明提要}
\providetranslation{Proof Sketches}{证明提要}
\providetranslation{Proof sketches}{证明提要}
\providetranslation{proof sketches}{证明提要}
\providetranslation{Sketch of Proof}{证明提要}
\providetranslation{Sketch of Proofs}{证明提要}
\providetranslation{Sketch of proof}{证明提要}
\providetranslation{Sketch of proofs}{证明提要}
\providetranslation{sketch of proof}{证明提要}
\providetranslation{sketch of proofs}{证明提要}
\providetranslation{Propositions}{命题}
\providetranslation{propositions}{命题}
\providetranslation{Proposition}{命题}
\providetranslation{proposition}{命题}
\providetranslation{Remarks}{注}
\providetranslation{remarks}{注}
\providetranslation{Remark}{注}
\providetranslation{remark}{注}
\providetranslation{Solutions}{解}
\providetranslation{solutions}{解}
\providetranslation{Solution}{解}
\providetranslation{solution}{解}
\providetranslation{Theorems}{定理}
```

```
\providetranslation{theorems}{定理}
\providetranslation{Theorem}{定理}
\providetranslation{theorem}{定理}
⟨/theorem⟩
⟨/dict⟩
```

14.17 ctexcap 宏包

```
⟨*ctexcap⟩
```

ctexcap 是过时宏包。

```
\clist_new:N \l__ctex_ctexcap_options_clist
\clist_set:Nx \l__ctex_ctexcap_options_clist
  { \exp_not:v { opt@ \@currname . \@currext } , heading }
\msg_new:nnn { ctexcap } { deprecated }
  {
    Package~`ctexcap'~is~deprecated.\\
    Please~use~package~`ctex'~with~option~`#1'~instead: \\\\
    \iow_indent:n { \token_to_str:N \usepackage [#1] \{ ctex \} } \\
  }
\msg_warning:nnx { ctexcap } { deprecated }
  { \clist_use:Nn \l__ctex_ctexcap_options_clist { , ~ } }
```

ctexcap 是默认打开 heading 选项的 ctex。

```
\PassOptionsToPackage { heading = true } { ctexcap }
\RequirePackageWithOptions { ctex }
⟨/ctexcap⟩
```

14.18 ctexhook 宏包

```
⟨*ctexhook⟩
```

\ctex_at_end_preamble:n
\ctex_after_end_preamble:n

实现 etoolbox 宏包的 \AtEndPreamble 和 \AfterEndPreamble。

```
\cs_new_protected:Npn \ctex_at_end_preamble:n #1
  { \tl_gput_right:Nn \g__ctex_end_preamble_hook_tl {#1} }
\cs_new_protected:Npn \ctex_after_end_preamble:n #1
  { \tl_gput_right:Nn \g__ctex_after_end_preamble_hook_tl {#1} }
\cs_new_protected_nopar:Npn \CTEX@document@left@hook
  { \group_end: \g__ctex_end_preamble_hook_tl \group_begin: }
\cs_new_protected_nopar:Npn \CTEX@document@right@hook
  { \scan_stop: \g__ctex_after_end_preamble_hook_tl \tex_ignorespaces:D }
\cs_set_nopar:Npx \document
  {
    \CTEX@document@left@hook
    \exp_not:o { \document }
    \CTEX@document@right@hook
  }
\tl_new:N \g__ctex_end_preamble_hook_tl
\tl_new:N \g__ctex_after_end_preamble_hook_tl
```

\ctex_at_end_package:nn

与 filehook 的 \AtEndOfPackageFile* 类似，如果原来没有在载入宏包则在宏包末尾执行语句，否则立即执行。

```
\cs_new_protected:Npn \ctex_at_end_package:nn #1#2
  {
    \@ifpackageloaded {#1}
      {#2}
      { \ctex_gadd_hook:cn { g__ctex_at_end_ #1 _hook_tl } {#2} }
  }
```

\ctex_gadd_hook:Nn
\ctex_gadd_hook:cn

给钩子附加内容。

```
\cs_new_protected:Npn \ctex_gadd_hook:Nn #1#2
  {
```

```
    \tl_if_exist:NF #1 { \tl_new:N #1 }
    \tl_gput_right:Nn #1 {#2}
  }
\cs_generate_variant:Nn \ctex_gadd_hook:Nn { c }
```

\ctex_package_end_hook:n
\ctex_package_end_hook:o

宏包末尾钩子，只执行一次，用后清除。

```
\cs_new_protected_nopar:Npn \ctex_package_end_hook:n #1
  {
    \cs_if_exist_use:cT { g__ctex_at_end_ #1 _hook_tl }
      { \cs_undefine:c { g__ctex_at_end_ #1 _hook_tl } }
  }
\cs_generate_variant:Nn \ctex_package_end_hook:n { o }
```

对 \@popfilename 做补丁来实现 \ctex_at_end_package:nn 的功能。

```
\tl_put_left:Nn \@popfilename
  {
    \cs_if_eq:NNT \@currext \@pkgextension
      { \ctex_package_end_hook:o { \@currname } }
  }
```

⟨/ctexhook⟩

14.19 ctexpatch 宏包

⟨*ctexpatch⟩

```
\cs_if_exist:NF \str_new:N { \RequirePackage { l3str } }
```

\ctex_patch_cmd_once:NnnnTF

只进行第一次匹配进行替换。参数 #2 是宏重建时的 \catcode 设置。

```
\cs_new_protected:Npn \ctex_patch_cmd_once:NnnnTF #1#2
  {
    \ctex_patch_boot:NNnnTF \__ctex_patch_cmd:Nnnnnw #1
      { once } {#2} { \use_i:nn } { \use_ii:nn }
  }
```

\ctex_patch_cmd_all:NnnnTF

替换所有匹配到的文本。

```
\cs_new_protected:Npn \ctex_patch_cmd_all:NnnnTF #1#2
  {
    \ctex_patch_boot:NNnnTF \__ctex_patch_cmd:Nnnnnw #1
      { all } {#2} { \use_i:nn } { \use_ii:nn }
  }
```

\ctex_patch_cmd:Nnn

快捷方式，在补丁的时候关闭 LaTeX3 语法和设置 @ 为字母类，补丁失败时给出警告。

```
\cs_new_protected:Npn \ctex_patch_cmd:Nnn #1
  {
    \ctex_patch_boot:NNnnTF \__ctex_patch_cmd:Nnnnnw #1
      { once }
      {
        \ExplSyntaxOff
        \char_set_catcode_letter:n { 64 }
      }
      { }
      { \ctex_patch_failure:N #1 }
  }
\cs_new_protected:Npn \ctex_patch_failure:N #1
  { \msg_warning:nnx { ctex } { patch-failure } { \token_to_str:N #1 } }
\msg_new:nnn { ctex } { patch-failure }
  { Oops!~Command~`#1'~is~NOT~patchable.\\ }
```

\ctex_preto_cmd:NnnTF

在宏的原本定义前面增加钩子。

```
\cs_new_protected:Npn \ctex_preto_cmd:NnnTF #1#2
```

```
  {
    \ctex_patch_boot:NNnnTF \__ctex_hookto_cmd:Nnnnw #1
      { left } {#2} { \use_i:nn } { \use_ii:nn }
  }
```

\ctex_appto_cmd:NnnTF　在宏的原本定义后面追加钩子。

```
\cs_new_protected:Npn \ctex_appto_cmd:NnnTF #1#2
  {
    \ctex_patch_boot:NNnnTF \__ctex_hookto_cmd:Nnnnw #1
      { right } {#2} { \use_i:nn } { \use_ii:nn }
  }
```

\ctex_patch_boot:NNnnTF　参数记号 # 作为宏的参数被读入时，总是会双写，会影响随后的字符串替换。需要先将它转换为普通符号。

```
\cs_new_protected:Npn \ctex_patch_boot:NNnnTF #1#2#3#4#5#6
  {
    \tl_set:Nn \__ctex_patch_true:w {#5}
    \tl_set:Nn \__ctex_patch_false:w {#6}
    \group_begin:
      \char_set_catcode_other:n { 35 }
      \ctex_parse_name:NN #1 #2 {#3} {#4}
  }
\tl_new:N \__ctex_patch_true:w
\tl_new:N \__ctex_patch_false:w
```

\ctex_parse_name:NN　用 \DeclareRobustCommand 定义的宏或者由 \newcommand 或 \newrobustcmd 定义的带一个可选参数的宏第一次展开的结果都不是其实际定义，实际定义被保存在另外的宏中。由这些命令定义的宏的第一次展开结果可以有下面的形式（细节可查阅 xpatch 的文档）：

```
\protect␣\xaa␣␣                    % \DeclareRobustCommand\xaa[1]{...}
\protect␣\xab␣␣                    % \DeclareRobustCommand\xab[1][]{...}
\@protected@testopt␣\xac␣\\xac␣{}  % \newcommand\xac[1][]{...}
\@testopt␣\\xad␣{}                 % \newrobustcmd\xad[1][]{...}
\x@protect␣\1\protect␣\1␣␣         % \DeclareRobustCommand\1[1]{...}
\x@protect␣\2\protect␣\2␣␣         % \DeclareRobustCommand\2[1][]{...}
\@protected@testopt␣\3\\3␣{}       % \newcommand\3[1][]{...}
\@testopt␣\\4␣{}                   % \newrobustcmd\4[1][]{...}
```

ctexpatch 的主要原理是先对宏的 \meaning 作字符串替换，然后再用 \scantokens 来重建它。我们希望对宏的实际定义打补丁，为此需要先得到对应的名字。letltxmacro、show2e 和 xpatch 宏包中都有类似的工作。

```
\cs_new_protected:Npn \ctex_parse_name:NN #1#2
  { \ctex_parse_name:NNx #1#2 { \cs_to_str:N #2 } }
\group_begin:
\cs_set_protected:Npn \__ctex_tmp:w #1#2#3
  {
    \cs_new_protected:Npn \ctex_parse_name:NNn ##1##2##3
      {
        \bool_lazy_or:nnTF
          { \cs_if_exist_p:c { ##3 ~ } }
          { \cs_if_exist_p:c { #1##3 } }
          {
            \group_begin:
            \use:x
              {
                \group_end:
                \__ctex_parse_name:nNNNnN
                  { \cs_replacement_spec:N ##2 }
                  \exp_not:N ##2
                  \exp_not:c { ##3 ~ }
                  \exp_not:c { #1##3 }
              } {##3} ##1
```

```
          }
          { ##1##2 }
      }
    \cs_new_protected:Npn \__ctex_parse_name:nNNNnN ##1##2##3##4##5##6
      {
        \exp_args:Nc ##6
          {
            \str_case:nnTF {##1}
              {
                { \protect ##3 } { }
                { \x@protect ##2 \protect ##3 } { }
              }
              {
                \str_if_eq:eeTF
                  { \exp_not:n { #1@protected@ ##3 #1##3 } }
                  {
                    \exp_last_unbraced:Nf \__ctex_parse_name:w
                      \cs_replacement_spec:N ##3 #3 ~ #2 \q_stop
                  }
                  { #1##5 ~ } { ##5 ~ }
              }
              {
                \str_case:onTF { \__ctex_parse_name:w ##1 #3 ~ #2 \q_stop }
                  {
                    { #1@protected@ ##2 ##4 } { }
                    { #1@ ##4 } { }
                  }
                  { #1##5 } {##5}
              }
          }
      }
    \cs_new:Npn \__ctex_parse_name:w ##1 #3 ~ ##2 #2 ##3 \q_stop { ##1##2 }
  }
\use:x
  {
    \__ctex_tmp:w
      { \c_backslash_str }
      { \c_left_brace_str }
      { \tl_to_str:n { testopt } }
  }
\group_end:
\cs_generate_variant:Nn \ctex_parse_name:NNn { NNx }
```

\l__ctex_prefix_str
\l__ctex_parameter_str
\l__ctex_replacement_str

分别保存宏的 \meaning 中的前缀、参数文本和替换文本。

```
\str_new:N \l__ctex_prefix_str
\str_new:N \l__ctex_parameter_str
\str_new:N \l__ctex_replacement_str
```

\ctex_get_macro_meaning:NTF
__ctex_get_macro_meaning:w

解构待补丁宏的 \meaning。若命令不是宏，则走向 false 分支。

```
\group_begin:
  \cs_set_protected:Npn \__ctex_tmp:w #1
    {
      \prg_new_protected_conditional:Npnn
        \ctex_get_macro_meaning:N ##1 { TF }
        {
          \exp_after:wN \__ctex_get_macro_meaning:w
            \token_to_meaning:N ##1 \q_mark #1 -> \q_mark \q_stop
        }
      \cs_new_protected:Npn \__ctex_get_macro_meaning:w
          ##1 #1 ##2 -> ##3 \q_mark ##4 \q_stop
        {
          \tl_if_empty:nTF { ##4 }
            { \prg_return_false: }
            {
              \str_set:Nn \l__ctex_prefix_str      { ##1 }
```

```
              \str_set:Nn \l__ctex_parameter_str   { ##2 }
              \str_set:Nn \l__ctex_replacement_str { ##3 }
              \prg_return_true:
            }
        }
    }
  \exp_args:No \__ctex_tmp:w { \tl_to_str:n { macro: } }
\group_end:
```

\ctex_if_rescanable:NnTF　检查宏是否可以重建。

```
\cs_new_protected:Npn \ctex_if_rescanable:NnTF #1#2#3#4
  {
    \ctex_get_macro_meaning:NTF #1
      {
        \__ctex_patch_rebuild:Nn \__ctex_rebuild_cmd:w {#2}
        \cs_if_eq:NNTF #1 \__ctex_rebuild_cmd:w {#3} {#4}
      }
      {#4}
  }
\cs_new_eq:NN \__ctex_rebuild_cmd:w \prg_do_nothing:
```

__ctex_patch_rebuild:Nn　使用 \tl_rescan:nn 来重新记号化 \meaning 字符串。

```
\cs_new_protected:Npn \__ctex_patch_rebuild:Nn #1#2
  {
    \__ctex_patch_rescan:NNn \l__ctex_prefix_tl      \l__ctex_prefix_str      {#2}
    \__ctex_patch_rescan:NNn \l__ctex_parameter_tl   \l__ctex_parameter_str   {#2}
    \__ctex_patch_rescan:NNn \l__ctex_replacement_tl \l__ctex_replacement_str {#2}
    \use:x
      {
        \exp_not:o { \l__ctex_prefix_tl } \tex_def:D \exp_not:N #1
          \exp_not:o { \l__ctex_parameter_tl }
            { \exp_not:o { \l__ctex_replacement_tl } }
      }
  }
\cs_new_protected:Npn \__ctex_patch_rescan:NNn #1#2#3
  {
    \str_if_empty:NTF #2
      { \tl_clear:N #1 }
      { \tl_set_rescan:Nno #1 {#3} {#2} }
  }
\tl_new:N \l__ctex_prefix_tl
\tl_new:N \l__ctex_parameter_tl
\tl_new:N \l__ctex_replacement_tl
```

__ctex_patch_cmd:Nnnnnw　对宏的替换文本进行字符串替换，然后重建。

```
\cs_new_protected:Npn \__ctex_patch_cmd:Nnnnnw #1#2#3#4#5
  {
    \group_end:
    \ctex_if_rescanable:NnTF #1 {#3}
      {
        \use:x
          {
            \__ctex_patch_replace:nnnTF {#2}
              { \tl_to_str:n {#4} }
              { \tl_to_str:n {#5} }
          }
          {
            \__ctex_patch_rebuild:Nn #1 {#3}
            \__ctex_patch_true:w
          }
          { \__ctex_patch_false:w }
      }
      { \__ctex_patch_false:w }
  }
```

__ctex_patch_replace:nnnTF　替换前先检查原文本是否存在。

```
\cs_new_protected:Npn \__ctex_patch_replace:nnnTF #1#2#3#4
  {
    \tl_if_in:NnTF \l__ctex_replacement_str {#2}
      { \use:c { tl_replace_ #1 :Nnn } \l__ctex_replacement_str {#2} {#3} #4 }
  }
```

__ctex_hookto_cmd:Nnnnw　在宏的前/后附加钩子。

```
\cs_new_protected:Npn \__ctex_hookto_cmd:Nnnnw #1#2#3#4
  {
    \group_end:
    \ctex_get_macro_meaning:NTF #1
      {
        \str_if_empty:NTF \l__ctex_parameter_str
          { \__ctex_hookto_cmd_parameterless:Nnnnw }
          { \__ctex_hookto_cmd_parameter:Nnnnw }
          #1 {#2} {#3} {#4}
      }
      { \__ctex_patch_false:w }
  }
```

__ctex_hookto_cmd_parameterless:Nnnnw　如果宏没有参数，可以直接进行附加操作。注意保持宏的前缀。

```
\cs_new_protected:Npn \__ctex_hookto_cmd_parameterless:Nnnnw #1#2#3#4
  {
    \str_if_empty:NF \l__ctex_prefix_str
      { \tl_rescan:no {#3} { \l__ctex_prefix_str } }
    \tex_edef:D #1
      {
        \use:c { __ctex_ #2 _hook_aux:nn }
          { \exp_not:o {#1} }
          { \exp_not:n {#4} }
      }
    \__ctex_patch_true:w
  }
\cs_generate_variant:Nn \tl_rescan:nn { no }
\cs_new:Npn \__ctex_left_hook_aux:nn #1#2 { #2#1 }
\cs_new_eq:NN \__ctex_right_hook_aux:nn \use:nn
```

__ctex_hookto_cmd_parameter:Nnnnw　如果宏有参数，需要在字符串中进行附加，然后再重建。

```
\cs_new_protected:Npn \__ctex_hookto_cmd_parameter:Nnnnw #1#2#3#4
  {
    \__ctex_patch_rebuild:Nn \__ctex_rebuild_cmd:w {#3}
    \cs_if_eq:NNTF #1 \__ctex_rebuild_cmd:w
      {
        \use:c { str_put_ #2 :Nn } \l__ctex_replacement_str {#4}
        \__ctex_patch_rebuild:Nn #1 {#3}
        \__ctex_patch_true:w
      }
      { \__ctex_patch_false:w }
  }
```

⟨/ctexpatch⟩

版本历史

v2.0 (2014/03/06 – 2015/05/06)

General: c5size, cs4size 是过时选项。 38
captiondelimiter 是过时选项。 113
fancyhdr 成为过时选项，原选项功能总是打开。 41
fntef 成为过时选项，原选项功能总是打开。 41
hyperref 成为过时选项，原选项功能总是打开。 41
indent, noindent 是过时选项。 39
nofonts, adobefonts, winfonts 是过时选项。 39
nopunct 是过时选项。 40
nospace 是过时选项。 40
nozhmap 是过时选项。 40
punct 选项可以设置标点格式。 40
ctex 宏包新增 heading 选项。 41
\CTEXindent, \CTEXnoindent 是过时命令。 81
\CTEXsetup, \CTEXoptions 是过时命令。 14, 79
\CTEXunderdot, \CTEXunderline, \CTEXunderdblline, \CTEXunderwave, \CTEXsout, \CTEXxout 是过时命令；CTEXfilltwosides 是过时环境。 74
标题设置新增 pagestyle 选项。 85
调整 \footnotesep 的大小，以适合行距的变化。 112
兼容 extsizes 宏包、beamer、memoir 等提供的更多字号选项。 42
将标题汉化功能加入 ctex.sty。 82
将中文字号功能提取到可以独立使用的 ctexsize。 . . . 109
解决 etoolbox 与 breqn 关于 \end 的冲突。 113
默认关闭 CJKfntef 或 xeCJKfntef 的彩色设置。 74
删除 c19gbsn.fd 和 c19gkai.fd。 1
通过 LuaTeX-ja 宏包支持 LuaLaTeX。 49
新增 autoindent 选项。 39
新增 fontset 选项。 39
新增 linespread 选项。 38
新增 linestretch 选项。 79
新增 scheme 选项，并将 cap 和 nocap 列为过时选项。 . . 41
新增 zhmCJK 支持选项。 40
新增 zihao 选项。 38
新增统一设置接口 \ctexset。 78
应用 LaTeX3 重新整理代码。 1
中文字号不再采用近似值。 110
自动检测操作系统，载入对应的字体配置。 71
\CJK@surr: 解决与 \nouppercase 的冲突。 45

v2.0.1 (2015/05/15)

General: 修复 10pt、11pt 等选项无效的问题。 42

v2.0.2 (2015/05/16)

General: 修复加载 ctex 宏包后章节标题后第一段无段首缩进的问题。 109

v2.1 (2015/05/18 – 2015/06/19)

General: format+, nameformat+ 等带加号的选项，加号与前面的文字之间可以有可选的空格。 84
nameformat 可以接受章节名字为参数。 84
不依赖 ifpdf 宏包。 37
不再设置 hyperref 宏包的 colorlinks 选项。 72
给 pdfLaTeX 下的非 UTF8 编码 CJK 字体族加上 CMap。 43
将章节标题设置功能提取到可以独立使用的宏包 ctexheading 中。 1
新的标题格式选项 aftertitle。 84
修复 ctexbook 和 ctexrep 类的中文 part/number 选项初值为空的错误。 103
\ctex_if_macosx:TF: 改用 /Library/Fonts/Songti.ttc 为特征文件。 72
hyperref: 补充定义 \hypersetup。 41

v2.2 (2015/06/21 – 2015/06/30)

General: beforeskip 和 afterskip 选项的符号不再有特殊意义。 84
beforeskip、afterskip 和 indent 选项支持表达式。 . . 87
不再依赖 etoolbox 宏包。 1
非 ctexart 类的 part/beforeskip 和 part/afterskip 选项有意义。 87
给 enumitem 宏包注册 \chinese 和 \zhnum。 82
将文档开头和宏包末尾钩子提取到 ctexhook 宏包中。 . 133
将中文版式下的 part 和 chapter 标题的 nameformat 和 titleformat 选项的初值合并到 format 中。 103
删去 etoolbox 与 breqn 的兼容补丁。 113
新的标题格式选项 afterindent。 84
新的标题格式选项 numbering。 84
新的标题格式选项 runin。 84
新增子宏包 ctexpatch 实现给宏打补丁功能。 134

v2.3 (2015/09/17 – 2016/01/05)

General: .value_required: 和 .value_forbidden: 已过时。 38
代码实现避免使用 \lowercase 技巧（Joseph Wright）。 . 37
更新 LuaTeX-ja 支持（20150922.0）。 49
更新 unicode-math 宏包补丁。 50
兼容 titleps 宏包。 98
修复 nameformat 作用域问题。 84
与 LaTeX3 (2015/12/20) 同步。 35

v2.4 (2015/02/19 – 2016/04/25)

General: 初步支持 upLaTeX。 68
加强 beamer 宏包支持。 1
提供 translator 宏包的中文定理名称翻译。 131
提供 upLaTeX 的 NFSS 字体定义。 115
正确更新 CJK 包的 \CJKfamilydefault。 70
正确设置 upTeX 下字体命令。 69
\ctex_parse_name:NN: 修复宏名解析错误。 135

v2.4.1 (2016/04/26 – 2016/05/14)

General: beamer 不调整默认行距。 108
beamer 不调整默认字体大小。 108
使用 bootfont.bin 判断 Windows XP 以避免权限问题。 117
随字体更新 upTeX 的 \xkanjiskip。 77
新的标题格式选项 part/fixbeforeskip 和 chapter/fixbeforeskip。 85
正确更新 upLaTeX 的 \CJKfamilydefault。 70
\ccwd: 正确设置 upTeX 下的 \ccwd。 75
\chinese: 支持 \pagenumbering。 82
zihao: 不允许无参 zihao 选项。 38

v2.4.2 (2016/05/15)
General: 恢复 luatexja 对 \emshape 和 \eminnershape 的重定义。 50
\em: 兼容 upLaTeX 2016/05/07u00 的定义。 69
v2.4.3 (2016/06/03 – 2016/08/26)
General: 更新 unicode-math 补丁。 55
简化 fontspec 补丁。 50
确保 \proofname 非空。 82
删除选项 part/fixbeforeskip 和 chapter/fixbeforeskip。 85
新的标题格式选项 fixskip。 84
v2.4.4 (2016/09/09 – 2016/09/19)
General: 不再默认设置 xeCJK 的伪粗体。 48
改进 hyperref 宏包的标题锚点设置。 95
解决 zhmap 文件的 \catcode 问题。 44
使用 titlesec 时，章节目录也使用 CTeX 的编号。 98
提供 \CTEXifname。 86
提供 \partmark。 86
新的标题格式选项 break。 84
v2.4.5 (2016/10/01 – 2016/10/25)
General: 新的标题格式选项 chapter/lofskip 和 chapter/lotskip。 85
新的标题格式选项 hang。 84
新的标题格式选项 tocline。 84
\ps@headings: 修复补丁失败。 100
v2.4.6 (2016/10/31 – 2016/11/20)
General: \CTEXifname 初始为假。 86
支持字体属性可选项在后的新语法。 60
重新初始化 \ifthechapter 等。 99
v2.4.7 (2016/12/23 – 2016/12/27)
General: 依赖 pxeverysel 宏包。 70
\ps@headings: 修复 ctexrep 类的 \chaptermark 汉化错误。 100
v2.4.8 (2017/02/23)
\CTEX@fontfamily: 解决与 fontspec 2017/01/24 v2.5d 的字体族匹配兼容问题。 61
v2.4.9 (2017/02/27)
General: 调整 unicode-math 补丁的代码顺序。 55
v2.4.10 (2017/07/19 – 2017/07/23)
General: 常数 \c_minus_one 已过时。 35
定义 \cht，\cdp 和 \cwd。 77
使用 lazy 函数对 Boolean 表达式进行最小化运算（LaTeX3 2017/07/19）。 35
v2.4.11 (2017/08/17 – 2017/11/21)
General: 不把 Enclosed Alphanumerics 设置为 JAchar。 . 49
不把希腊和西里尔字母设置为 JAchar。 49
因上游 l3keys 变化，重新定义 format␣+ 等带空格加号的选项。 84
\ps@headings: 补充页眉空格。 100
v2.4.12 (2017/12/05 – 2018/01/27)
General: 同步 LaTeX3 2017/12/16。 1
修正 \ctexset 在 ctexheading 包中无定义的错误（曾祥东）。 78
正确使用 \ltjsetkanjiskip 和 \ltjsetxkanjiskip。 . 68
\CTeX: 不依赖 \ifincsname。 113
v2.4.13 (2018/03/23)
General: 修正导言区 \selectfont 钩子位置。 76
v2.4.14 (2018/05/01)
General: 配置 macnew 的默认字体设置。 121
区分 macold 及 macnew。 121
为 macnew 配置字体命令。 124
v2.4.15 (2019/01/29 – 2019/04/05)
General: 定义 part/hang 和 chapter/hang。 87
将 JY2 和 JT2 编码的字体定义提取到单独的文件中。 . 115
将 upLaTeX 的默认字体由 mc 改为 zhrm，并启用 \jfam。 69
局部指定 autoindent 为 false，并交换 \CTEX@XXX@indent 与 \CTEX@XXX@format 的顺序。 . . . 101
同步 LaTeX3 2019/03/05。 1
统一“方正细黑一_GBK”的名称为 FZXiHeiI-Z08。 . . 122
显式补丁 upLaTeX 的 \rmfamily 等字体命令。 68
修正 part/indent 和 chapter/indent 的实现方法。 . . 87
v2.4.16 (2019/05/11 – 2019/05/29)
General: 更好地兼容 nameref 宏包。 96
修正 part/indent 和 chapter/indent 的实现方法，在其标题内部禁用 autoindent。 87
允许设置 autoindent 为 0。 37

代码索引

意大利体的数字表示描述对应索引项的页码；带下划线的数字表示定义对应索引项的代码行号；罗马字体的数字表示使用对应索引项的代码行号。

Symbols

\ 502, 3287, 3294, 3310, 3313
\# 468, 3188, 3247
.../afterindent *20*
.../aftername *18*
.../aftername+ *18*
.../afterskip *21*
.../aftertitle *19*
.../aftertitle+ *19*
.../beforeskip *20*
.../break *23*
.../break+ *23*
.../fixskip *21*
.../format *17*
.../format+ *17*
.../indent *21*
.../name *15*
.../nameformat *17*
.../nameformat+ *17*
.../number *16*
.../numberformat *17*
.../numberformat+ *17*
.../titleformat *19*
.../titleformat+ *19*
.../tocline *23*
\/ 1638
\\ 15, 33, 34, 103, 139, 141, 143, 200, 210, 666, 1096, 1178, 2331, 3674, 3675, 3788, 3789, 4009, 4039, 5107, 5108, 5109, 5183
\{ 1107, 1108, 5109
\} 1107, 1108, 5109
10pt *9*, 361
11pt *9*, 361
12pt *9*, 361

A

\abovedisplayshortskip . 3879, 3887, 3898, 3918, 3926, 3937
\abovedisplayskip 3878, 3881, 3886, 3893, 3897, 3904, 3917, 3920, 3925, 3932, 3936, 3943
\abstractname 2315
abstractname *12*
\addCJKfontfeature 1178, 1222
\addCJKfontfeatures 1195
\addcontentsline 2532, 2580, 2584, 2589, 2620, 2624, 2629, 2726, 2731, 2737, 2743, 2871, 2885
\AddEnumerateCounter 2270, 2272, 2273, 2274
\addpenalty 2829, 3514, 3547, 3581, 3595, 3609
\addtocontents 2520, 2526, 2748, 2749
\addvspace 2521, 2527, 2541, 2544, 2748, 2749, 2829, 2833, 3327, 3337, 3347, 3357, 3367, 3377
adobefonts *30*
\algorithmname 2322, 2338
algorithmname *12*
AlternateFont *27*, 1334
\appendix 3025
appendix/name *24*, 3014
appendix/number *24*, 3014
appendix/numbering *23*, 3014
\appendixname 2317, 3659
appendixname *12*
\arabic 3561
\AtBeginDocument 569, 1627
\AtBeginDvi 427
\AtBeginShipoutFirst 429, 1846, 1861
\AtBeginUTFCommand 713
\AtEndOfClass 129
\AtEndOfPackage 130
\AtEndUTFCommand 719
autoindent *10*, 179, 2125

B

\baselinestretch 2180
\begin 3333, 3353, 3373
\begingroup 2862, 2899, 2931, 3321, 3341, 3361, 4608, 4641, 4648
\belowdisplayshortskip . 3880, 3888, 3899, 3919, 3927, 3938
\belowdisplayskip 3881, 3893, 3904, 3920, 3932, 3943
\bfdefault 960, 966, 1324, 1331
\bfseries 2598, 2602, 2640, 2644, 2659, 2679, 2778, 2783, 2802, 3441, 3443, 3447, 3460, 3461, 3464, 3490, 3492, 3498, 3516, 3520, 3541, 3575, 3592, 3606
\bibname 2320, 2323, 2331, 2339, 2345, 2346
bibname *12*
bool commands:
  \bool_gset_false:N 260, 265
  \bool_gset_true:N 255
  \bool_if:NTF . 70, 1136, 2946, 2957, 3129, 3139, 3150, 3678
  \bool_lazy_or:nnTF 5211
  \bool_new:N 125, 1173
  \bool_set_false:N 1148
  \bool_set_true:N 1166
  \c_false_bool 2973, 2975
  \c_true_bool 2972, 2974
box commands:
  \box_ht:N 3959
  \box_new:N 60
  \box_wd:N 1932, 3243

C

c5size *30*
cap *30*
captiondelimiter *31*, 3966

\catcode 4608, 4609, 4610, 4611, 4615, 4616
CCT 30
CCTfont 30
\ccwd 21, 26, 121, 183, 1928, 1975, 1976, 1977, 2131, 2169, 2188, 2189, 2196, 2223, 2224, 2225, 2228, 2230, 2231, 2242, 2253, 2254, 3976, 3977, 3978
\cdp 2043
\centering 2636, 2675, 3323, 3343, 3363, 3447, 3459, 3464, 3468, 3498, 3520, 3528, 3563
\chapter 2703, 3684
chapter/beforeskip 32
chapter/fixbeforeskip 32
chapter/hang 22
chapter/lofskip 21, 2461
chapter/lotskip 21, 2461
chapter/numbering 15
chapter/pagestyle 22, 2461
\chaptermark 2747
\chaptername 3487
char commands:
\char_set_catcode:nn 73, 506, 507, 542, 543
\char_set_catcode_active:N 1359
\char_set_catcode_comment:n 501
\char_set_catcode_escape:n 502
\char_set_catcode_letter:n 67, 1630, 3261, 5175
\char_set_catcode_math_superscript:n 537
\char_set_catcode_other:N . 468, 868, 1342, 3188, 3247
\char_set_catcode_other:n 536, 5199
\char_set_catcode_space:n 2049
\char_value_catcode:n 73, 506, 507, 542, 543
CharRange 27, 1334
\Chinese 31, 2266
\chinese 26, 2264, 2274, 3434, 3497
\cht 2043
\CJK 1897
\CJKbold 4087, 4088, 4098, 4099, 4115, 4116, 4118, 4119, 4121, 4122, 4126, 4127, 4129, 4130, 4132, 4133, 4147, 4148, 4150, 4151, 4153, 4154, 4158, 4159, 4161, 4162, 4164, 4165
\CJKfamily 516, 520, 524, 528, 1032, 1055, 1270, 1277, 1278, 1279, 1280, 1428, 1672, 4592, 4593, 4594, 4595, 4597, 4598, 4600, 4602, 4603
\CJKfamilydefault 510, 511, 528, 584, 1257, 1258, 1270, 1280, 1288, 1290, 1294, 1301, 1314, 1649, 1650, 1736, 1740, 1748, 1761, 1763
\CJKfilltwosides 1905
\CJKfontspec 1195
\CJKglue 1931, 1945, 1959, 1972
\CJKhook 599
\CJKnormal 4080, 4081, 4082, 4083, 4084, 4085, 4086, 4091, 4092, 4093, 4094, 4095, 4096, 4097, 4114, 4117, 4120, 4125, 4128, 4131, 4146, 4149, 4152, 4157, 4160, 4163
\CJKpunctmapfamily 632, 633, 634, 635, 636, 637, 641, 642, 643, 644, 645, 646
\CJKrmdefault 491, 511, 512, 516, 1103, 1114, 1226, 1258, 1259, 1277, 1290, 1293, 1294, 1646, 1649, 1655, 1744, 4216, 4218, 4219, 4224, 4225, 4226, 4229, 4306, 4309, 4310, 4315, 4316, 4317, 4384, 4387, 4388, 4394, 4395, 4396, 4494, 4497, 4498, 4505, 4506, 4507, 4554, 4557, 4558, 4563, 4564, 4565
\CJKsfdefault 492, 513, 520, 1104, 1115, 1233, 1260, 1278, 1647, 1662, 1745, 4199, 4206, 4207, 4230, 4307, 4385, 4389, 4495, 4499, 4555
\CJKtilde 585
\CJKttdefault 493, 514, 524, 1105, 1116, 1239, 1261, 1279, 1648, 1669, 1746, 4217, 4231, 4308, 4386, 4496, 4556
clearalternatefont 28, 1520
\cleardoublepage 2555, 2704, 3457, 3485
\clearpage 2557, 2704, 3457, 3485
clist commands:
\clist_clear:N 981, 4944
\clist_concat:NNN 1001
\clist_const:Nn 612
\clist_gput_right:Nn 377, 389, 399, 401
\clist_gset:Nn 1249
\clist_if_empty:NTF 1347, 1366, 1606
\clist_map_break:n 3108, 3669
\clist_map_function:nN 1523, 1525, 1557, 1571
\clist_map_inline:Nn 602, 1459
\clist_map_inline:nn 362, 624, 1564, 1889, 1915, 3103, 3666, 3806, 3841, 4933
\clist_new:N 132, 998, 1250, 1401, 4962, 5102
\clist_put_left:Nn 1002, 3735, 3761
\clist_put_right:Nn 1165, 1447, 4956
\clist_set:Nn 985, 1384, 1385, 5103
\clist_use:Nn 5112
\clubpenalty 2930, 2937
\contentsname 2310
contentsname 12
continuation 12
cs commands:
\cs:w 3402
\cs_end: 3402
\cs_generate_variant:Nn 93, 444, 456, 945, 1067, 1154, 1174, 1175, 1380, 1417, 1519, 1557, 1565, 5144, 5150, 5266, 5373
\cs_gset_eq:NN 256, 261, 266, 476, 763, 767, 1141, 1144, 3061, 3062, 3067, 3092
\cs_gset_nopar:Npx 764
\cs_gset_protected_nopar:Npn 3026, 3073, 3088
\cs_gset_protected_nopar:Npx 802, 1028, 1421, 1515, 3002
\cs_if_eq:NNTF 904, 3059, 3091, 5153, 5299, 5379
\cs_if_exist:NTF 44, 353, 453, 758, 770, 800, 838, 923, 1420, 1512, 2007, 3174, 3389, 3684, 3686, 5158
\cs_if_exist_p:N 5212, 5213
\cs_if_exist_use:N 1051, 1052
\cs_if_exist_use:NTF 4, 741, 1073, 1534, 5147
\cs_if_free:NTF 438, 448, 604, 619, 1138, 1323, 2144, 2158, 2270, 3122, 3167, 3213, 3218, 3222
\cs_new:Npn 113, 120, 859, 865, 869, 977, 1054, 2976, 2984, 3011, 3147, 5256, 5374

\cs_new_eq:NN ... 6, 7, 8, 9, 20, 23, 107, 108, 109, 127, 128, 354, 445, 589, 650, 779, 804, 880, 911, 1222, 1229, 1757, 1910, 1911, 1912, 1924, 1972, 1983, 1989, 2265, 2266, 2267, 2506, 2509, 2970, 2971, 2972, 2973, 2974, 2975, 3025, 3049, 3050, 3407, 3682, 3687, 4938, 5303, 5375
\cs_new_nopar:Npn 819, 1046, 1099, 1110, 2264, 2443, 2992, 3211, 3216, 4963
\cs_new_protected:Npn 62, 111, 913, 978, 1181, 1345, 1436, 1450, 1601, 1978, 2454, 2531, 2943, 3084, 3237, 3861, 3867, 4939, 5117, 5119, 5133, 5139, 5159, 5164, 5169, 5180, 5184, 5189, 5194, 5204, 5209, 5228, 5279, 5294, 5304, 5316, 5325, 5344, 5349, 5361, 5376
\cs_new_protected_nopar:Npn 77, 79, 129, 130, 425, 436, 446, 451, 457, 489, 497, 532, 549, 591, 597, 600, 617, 622, 630, 639, 648, 739, 748, 761, 766, 768, 780, 794, 821, 836, 849, 873, 882, 899, 938, 955, 999, 1007, 1026, 1047, 1057, 1082, 1120, 1132, 1134, 1156, 1158, 1267, 1273, 1284, 1309, 1318, 1364, 1375, 1402, 1411, 1418, 1424, 1444, 1457, 1476, 1490, 1497, 1509, 1529, 1545, 1563, 1566, 1568, 1581, 1593, 1635, 1640, 1642, 1696, 1709, 1723, 1734, 1766, 1784, 1799, 1815, 1826, 1830, 1867, 1874, 1878, 1901, 1907, 1928, 1942, 1974, 1990, 2022, 2067, 2143, 2145, 2170, 2205, 2215, 2220, 2373, 2403, 2490, 2497, 2504, 2510, 2512, 2515, 2533, 2950, 2955, 2961, 3042, 3052, 3095, 3115, 3120, 3125, 3135, 3145, 3189, 3385, 3394, 3399, 3779, 3794, 3826, 3831, 3981, 3985, 4930, 4954, 5121, 5123, 5145
\cs_new_protected_nopar:Npx ... 461, 1205, 1894, 3176
\cs_replacement_spec:N 5220, 5242
\cs_set:Npn 1343, 1986, 2430
\cs_set_eq:NN 110, 184, 189, 194, 529, 547, 567, 594, 598, 606, 620, 773, 774, 881, 912, 930, 975, 1032, 1033, 1420, 1428, 1438, 1439, 1440, 1512, 1513, 1514, 1638, 1739, 2281, 2286, 2289, 2294, 2511, 2513, 2963, 2964, 2965, 2966, 2967, 2968, 3056, 3099, 3109
\cs_set_nopar:Npn 1695, 3228
\cs_set_nopar:Npx 1925, 3137, 5125
\cs_set_protected:Npn 720, 916, 5207, 5271
\cs_set_protected_nopar:Npn 593, 931, 1945, 3169, 3226, 3868
\cs_set_protected_nopar:Npx 1980
\cs_to_str:N 54, 1204, 5205
\cs_undefine:N . 1011, 1012, 1016, 1017, 1441, 1994, 5148
cs4size 30
\csname 2830, 2839, 2840, 2841, 2848, 2867, 2868, 2870, 2873, 2882, 2883, 2884, 2887, 4640, 4642, 4643, 4644, 4649, 4653
\CTeX 26, 3962
ctex commands:
\ctex_add_cmap:n 441, 446
\ctex_add_cmap:Nn 449, 451, 456
\ctex_add_to_selectfont:n 1978, 2013, 2041, 2069, 2157
\ctex_after_end_preamble:n 37, 110, 5117
\ctex_appto_cmd:NnnTF 472, 1629, 5189
\ctex_assign_heading_name:nn 2409, 2451, 3018
\ctex_at_end:n 129, 288, 351, 1972, 4048
\ctex_at_end_package:nn 134, 428, 475, 728, 921, 1840, 1842, 1860, 2268, 3071, 3086, 3155, 3162, 3221, 3223, 3224, 3258, 3298, 3413, 3964, 5133
\ctex_at_end_preamble:n 356, 574, 581, 1283, 1683, 5117
\ctex_auto_ignorespaces: 588, 598, 2079
\ctex_CJK_input:n 532
\ctex_declare_math_sizes:nnnn 3826, 3860
\ctex_detected_platform: 1765, 3989, 3998
\ctex_family_cmap:nn 436, 445
\ctex_file_input:n 77, 392, 651, 2108, 3693, 3728, 3733, 3738, 3739, 3754, 3759, 3765, 3766, 3870, 3872, 3990, 3994, 4002, 4048, 4177, 4178, 4434, 4435
\ctex_file_wrapper:nnn 62, 78, 81, 499, 534
\ctex_fix_varioref_label:n 3397, 3399
\ctex_fontset_error:n 3981, 4285, 4319, 4353, 4398, 4439, 4442, 4509, 4567
\ctex_gadd_hook:Nn 599, 5137, 5139
\ctex_get_macro_meaning:N 5274
\ctex_get_macro_meaning:NTF 5270, 5296, 5352
\ctex_hypersetup:n 72, 354, 1823, 1835, 1838, 1853, 1856, 1859
\ctex_if_autoindent_touched:TF 127, 184, 189, 194, 3720
\ctex_if_ccglue_touched:TF 1956, 2176
\ctex_if_ccglue_touched_p: 1956
\ctex_if_macosx:TF 1806, 1807, 1809, 1815
\ctex_if_preamble:TF 109, 4017
\ctex_if_rescanable:NnTF 5294, 5328
\ctex_ignorespaces_case:N 591, 2076, 2079, 2082
\ctex_load_fontset: 3985, 4025, 4047
\ctex_load_zhmap:nnnn 489, 4222, 4313, 4392, 4503, 4561
\ctex_ltj_add_font_features:n 1156, 1219
\ctex_ltj_add_font_features:nn 1156, 1157
\ctex_ltj_char_range_key:nn 1598, 1601
\ctex_ltj_clear_alternate_font:n 1523, 1529
\ctex_ltj_declare_alternate_shape:nnnnnn 1430, 1450
\ctex_ltj_declare_char_range:n 1560, 1563, 1565
\ctex_ltj_declare_char_range:nn 1567, 1568
\ctex_ltj_def_char_range_key:n 1573, 1593
\ctex_ltj_ensure_default_family: 1284, 1753
\ctex_ltj_extract_font: 773, 780
\ctex_ltj_family_if_exist:nNTF 1059, 1068, 1288, 1293, 1311, 1314, 1532
\l_ctex_ltj_family_tl 1061, 1066, 1123, 1124, 1139, 1143, 1146, 1157, 1526, 1527
\ctex_ltj_fontspec:nn 1120, 1132, 1167, 1214
\ctex_ltj_get_and_define_fonts:nN 873
\ctex_ltj_get_and_define_fonts_al:nN 877, 880
\ctex_ltj_get_and_define_fonts_ja:nN 876, 882
\ctex_ltj_if_alternate_shape_exist:nTF 783, 796, 814, 823
\ctex_ltj_if_jfont:nTF 55, 859, 870, 875
\ctex_ltj_if_jfont_math:NTF 865, 906
\ctex_ltj_math_group_hook: 54, 55, 905, 911, 931
\ctex_ltj_pickup_font: ... 754, 768, 843, 885, 887, 889
\ctex_ltj_reset_alternate_font:n 1525, 1545
\ctex_ltj_save_alternate_family:Nnnn 1395, 1415, 1418

\ctex_ltj_save_alternate_seq:Nn 1370, 1375
\ctex_ltj_save_alternate_seq:Nnnwn 1375
\ctex_ltj_save_alternate_seq:Nnnwnw 1378, 1381
\ctex_ltj_save_char_range:n 1571, 1578
\ctex_ltj_save_char_range:nn 1580, 1581
\ctex_ltj_select_alternate_font: 821, 2016
\ctex_ltj_select_font: 739, 2015
\ctex_ltj_set_alternate_family:nn 1373, 1402
\ctex_ltj_set_alternate_family:nnn . 1414, 1422, 1424
\ctex_ltj_set_alternate_family:Nnnnn 1406, 1411, 1417
\ctex_ltj_set_alternate_family:nnTF 986, 1364
\ctex_ltj_set_alternate_seq:n 1336, 1341
\ctex_ltj_set_alternate_shape:n ... 1478, 1490, 1501
\ctex_ltj_set_alternate_shape:nnN 1463, 1497
\ctex_ltj_set_alternate_shape:nnn 1466, 1473
\ctex_ltj_set_alternate_shape:nnnn 1475, 1476
\ctex_ltj_set_alternate_shape:Nnnnnnn .. 1453, 1457
\ctex_ltj_set_family:nnn 978, 1151, 1198, 1208, 1226, 1233, 1239, 1245
\ctex_ltj_set_kanjiskip:N 1640, 1950
\ctex_ltj_set_math_letter:NN 913, 925, 933
\ctex_ltj_set_xkanjiskip:N 1640, 2029
\ctex_ltj_subst_font: 774, 794
\ctex_ltj_swap_cs:NN 1429, 1436
\ctex_ltj_switch_family:n 1056, 1057, 1067, 1124, 1152, 1206
\ctex_ltj_update_mathfont: 1306, 1309
\ctex_ltj_update_mathfont:n 1312, 1315, 1318
\ctex_ltj_use_math_group:Nn 899, 930
\ctex_make_spa:nn 4930
\ctex_mono_jfm:n 938, 946, 2099
\ctex_package_end_hook:n 5145, 5154
\ctex_parse_name:NN 1997, 5200, 5204
\ctex_parse_name:NNn 5205, 5209, 5266
\ctex_patch_boot:NNnnTF 5161, 5166, 5171, 5186, 5191, 5194
\ctex_patch_cmd:Nnn 1691, 3191, 3197, 3203, 3270, 3273, 3277, 3283, 3293, 3301, 3304, 3309, 3312, 3976, 3977, 3978, 5169
\ctex_patch_cmd_all:NnnnTF 5164
\ctex_patch_cmd_once:NnnnTF 1685, 1999, 5159
\ctex_patch_failure:N 478, 518, 522, 526, 530, 1633, 2004, 3254, 3264, 5178, 5180
\ctex_plane_to_utfxvibe:Nn 548
\ctex_preto_cmd:NnnTF 515, 519, 523, 527, 3251, 3260, 5184
\ctex_punct_map_bfseries:nn 622, 4207, 4219, 4225, 4310, 4316, 4388, 4389, 4395, 4498, 4499, 4500, 4506, 4558, 4564
\ctex_punct_map_family:nn 617, 4199, 4206, 4216, 4217, 4224, 4306, 4307, 4308, 4315, 4384, 4385, 4386, 4394, 4494, 4495, 4496, 4505, 4554, 4555, 4556, 4563
\ctex_punct_map_itshape:nn 639, 4218, 4226, 4309, 4317, 4387, 4396, 4497, 4507, 4557, 4565
\ctex_punct_map_series:nnn 626, 627, 630
\ctex_punct_set:n 600, 4194, 4223, 4305, 4314, 4383, 4393, 4493, 4504, 4553, 4562
\ctex_punct_space:nn 648
\ctex_scheme_input:n 79, 3415, 3695
\ctex_select_size: .. 2132, 2139, 2142, 2143, 2164, 2218
\ctex_set_default_ccwd:Nn 111, 193, 2138, 2163
\ctex_set_font_size:Nnn 3861, 3876, 3884, 3895, 3906, 3907, 3908, 3909, 3910, 3911, 3912, 3915, 3923, 3934, 3945, 3946, 3947, 3948, 3949, 3950, 3951
\ctex_set_ignorespaces: 591, 599
\ctex_set_upfamily:nnn 1696, 4242, 4243, 4244, 4245, 4246, 4247, 4249, 4329, 4330, 4331, 4332, 4408, 4409, 4410, 4411, 4518, 4519, 4520, 4521, 4522, 4524, 4576, 4577, 4578
\ctex_set_upfonts:nnnnnn 1723, 4237, 4325, 4404, 4514, 4572
\ctex_set_upmap:nnn 1709, 1725, 1726, 1727, 1728, 4248, 4250, 4523, 4525
\ctex_set_zhmap:n 424, 494, 1711
\ctex_titleps_hook: 3187, 3222, 3223
\ctex_titlesec_hook: 3095, 3159
\ctex_update_ccglue: 1942, 2183, 2202, 2246
\ctex_update_ccwd: 1928, 2177, 2254
\ctex_update_default_family: 47, 574, 1283, 1683, 1733
\ctex_update_em_unit: 1974, 2172, 2222
\ctex_update_kanjisize: 2067, 2069
\ctex_update_parindent: 2150, 2205, 2247
\ctex_update_size: 2144, 2145
\ctex_update_stretch: 2149, 2170
\ctex_update_xkanjiskip: 2021, 2041
\ctex_update_ziju: 2152, 2220
\ctex_varioref_hook: 3394, 3405, 3413
\ctex_zhmap_case:nnn 128, 256, 261, 266, 4192, 4287, 4355, 4481, 4544
\ctex_zhmap_input:n 494, 497
\ctex_zihao:n 3778, 3779
\ctex_ziju:n 2214, 2215

ctex internal commands:

\g__ctex_after_end_preamble_hook_tl 5120, 5124, 5132
__ctex_assign_heading_name:nnn 2451
\l__ctex_autoindent_tl 126, 183, 188, 193, 2131, 2135, 2138, 2207, 2210, 2534
__ctex_calc_bounds:nn 4958, 4959, 4963
__ctex_ccglue: 1959, 1972
\l__ctex_ccglue_skip 1946, 1950, 1953, 1955, 1963, 1967, 2181, 2192, 2201, 2237, 2245
\c__ctex_class_tl 84, 86, 404, 409, 414, 419, 3682, 3688, 3690, 3691, 3693
__ctex_clear_fntef_color:n ... 1867, 1874, 1878, 1893
\c__ctex_cmap_encoding_seq 434, 440
\l__ctex_ctexcap_options_clist 5102, 5103, 5112
__ctex_def_heading_keys:n 2403, 2481
__ctex_default_ccwd_aux:n 112, 113
__ctex_default_ccwd_aux:w 116, 120
\l__ctex_encoding_tl 122, 216, 217, 482, 485, 576, 584, 1836, 2105, 2262, 3723, 3750, 3764
\g__ctex_end_preamble_hook_tl 5118, 5122, 5131
\c__ctex_engine_file_str 43, 2108

\c__ctex_engine_str 43
\l__ctex_family_default_init_tl 1736, 1756
__ctex_family_default_wrap:n 1739, 1757, 1760
__ctex_fix_varioref_label:n 3399
\g__ctex_font_size_int 145, 156, 157, 158, 372, 376, 398, 3740, 3741, 3767, 3768, 3869
\c__ctex_font_size_prop 3781, 3792, 3836, 3863
\c__ctex_font_size_seq 3790, 3793, 3804
\l__ctex_font_size_tl 3781, 3782, 3828, 3829, 3863, 3864
\g__ctex_fontset_tl 220, 1768, 1786, 1811, 1818, 1819, 3987, 3990, 3993, 3994, 4000, 4002, 4022, 4024, 4029, 4046
__ctex_get_font_sizes:Nn 3828, 3831
__ctex_get_macro_meaning:w 5270
\l__ctex_heading_bool 299, 3678
\l__ctex_heading_skip 2494, 2501, 2503, 2505, 2506, 2517, 2518, 2521, 2523, 2524, 2527
\c__ctex_headings_cs_seq 3044, 3057, 3066, 3089
\c__ctex_headings_seq ... 2363, 2478, 3172, 3235, 3396
__ctex_hookto_cmd:Nnnnw 5186, 5191, 5349
__ctex_hookto_cmd_parameter:Nnnnw 5356, 5376
__ctex_hookto_cmd_parameterless:Nnnnw . 5355, 5361
__ctex_indent_aux: 2947, 2950
__ctex_initial_heading:n 2373, 2480
\g__ctex_kanjisize_func 2047, 2057, 2068
__ctex_left_hook_aux:nn 5374
\l__ctex_line_spread_fp 112, 176, 3742, 3743, 3769, 3770, 3955, 3957
\l__ctex_line_stretch_tl 2163, 2168, 2169, 2173
\l__ctex_ltj_add_alternate_bool 1136, 1148, 1166, 1173
__ctex_ltj_alternate_cs:n 1012, 1015, 1016, 1017, 1051, 1052, 1054, 1139, 1142, 1143, 1145, 1146, 1370, 1407, 1469, 1494, 1534, 1551
\l__ctex_ltj_alternate_family_tl 1432, 1435
\l__ctex_ltj_alternate_options_clist 1385, 1386, 1397, 1401
\l__ctex_ltj_alternate_seq 982, 1354, 1355, 1363, 1369, 1377, 1625
\l__ctex_ltj_base_CJKfamily_tl 983, 997, 1049, 1469, 1494
\l__ctex_ltj_base_family_tl 65, 1013, 1044, 1050, 1413, 1454, 1513, 1532, 1537, 1538, 1550
__ctex_ltj_change_encoding: 947, 1031, 1427
\l__ctex_ltj_char_range_clist 981, 1338, 1347, 1366, 1384, 1396, 1408, 1431, 1453, 1604, 1606
__ctex_ltj_char_range_parse_feature:w . 1609, 1613
\g__ctex_ltj_char_range_prop 1461, 1568
\l__ctex_ltj_char_range_tl 1461, 1464, 1570, 1572, 1574, 1576, 1583
__ctex_ltj_check_family:n 1007, 1368
\l__ctex_ltj_current_font_tl 51, 741, 745, 746, 758, 800, 805
\l__ctex_ltj_current_shape_tl 823, 828, 833, 834
__ctex_ltj_declare_char_range:nn 1564, 1566
\g__ctex_ltj_default_features_clist 1001, 1249, 1250
__ctex_ltj_family_csname:n ... 1011, 1028, 1046, 1073
\g__ctex_ltj_family_font_name_prop 988, 1004, 1009, 1084, 1160, 1286, 1298
\g__ctex_ltj_family_font_options_prop 989, 1004, 1163
\g__ctex_ltj_family_int . 1126, 1128, 1446, 1447, 1449
\g__ctex_ltj_family_name_prop 1004, 1013, 1036, 1070, 1413
__ctex_ltj_family_unknown_warning:n 1064, 1082, 1542
\l__ctex_ltj_font_options_clist 985, 990, 991, 992, 998, 1035, 1164, 1165, 1168, 1404, 1405, 1409
__ctex_ltj_fontspec:nnn 1127, 1134, 1154
\g__ctex_ltj_fontspec_family_tl 1034, 1037, 1038, 1039, 1045, 1050, 1075
\g__ctex_ltj_fontspec_prop 1122, 1150, 1155
__ctex_ltj_gset_family_cs:nn 993, 1026
__ctex_ltj_if_jfont_math:w 866, 869
\l__ctex_ltj_jfm_tl 938, 959, 961, 1002
\c__ctex_ltj_math_fam_int 967, 968
\c__ctex_ltj_math_family_tl 1320, 1322, 1324, 1327, 1331
\c__ctex_ltj_math_tl 962, 963, 965, 967, 1245, 1311, 1321, 1326, 1330
__ctex_ltj_msg_def_family_map:n 1097, 1099
__ctex_ltj_msg_family_map:n 1025, 1096, 1110
__ctex_ltj_noembed_wrap:n 975, 977
__ctex_ltj_pass_args:nnnn 1181, 1197, 1207, 1213, 1225, 1232, 1238, 1244
__ctex_ltj_patch_external_font:w 786, 819
__ctex_ltj_pop_fontname: 757, 766, 845
__ctex_ltj_post_arg:w 1184, 1190
__ctex_ltj_push_fontname:n 753, 761, 842
\g__ctex_ltj_reset_alternate_prop 1018, 1536, 1548, 1556
__ctex_ltj_save_alternate_shape:Nnn 1468, 1493, 1509
__ctex_ltj_save_fontname:w 763, 767
__ctex_ltj_select_font_aux: 742, 748
__ctex_ltj_set_alternate_family:n 1040, 1047
__ctex_ltj_tmp:w 1343, 1360, 1438, 1440, 1441
\l__ctex_ltj_tmp_seq 1351, 1352, 1355, 1362
\l__ctex_ltj_tmp_tl 984, 985, 1009, 1020, 1023, 1161, 1169, 1203, 1206, 1208, 1288, 1293, 1311, 1312, 1314, 1315, 1349, 1350, 1351, 1388, 1390, 1391, 1398, 1608, 1616, 1625
\g__ctex_ltj_unknown_family_seq ... 1086, 1088, 1093
__ctex_ltj_update_family_aux: 1271, 1273
__ctex_ltj_update_family_uid:N 991, 1404, 1443
__ctex_ltj_use_global_options:N 992, 999, 1386, 1405
\c__ctex_macosx_file_str 1817, 1821
\l__ctex_msyh_suffix_tl 4181, 4182, 4186, 4203, 4205, 4239
\l__ctex_parameter_str 5267, 5286, 5307, 5354
\l__ctex_parameter_tl 5307, 5312, 5323
__ctex_parse_name:nNNNnN 5219, 5228
__ctex_parse_name:w 5241, 5247, 5256
__ctex_patch_cmd:Nnnnnw 5161, 5166, 5171, 5325
__ctex_patch_false:w 5197, 5203, 5340, 5342, 5359, 5385
__ctex_patch_rebuild:Nn 5298, 5304, 5337, 5378, 5382

__ctex_patch_replace:nnnTF 5332, 5344
__ctex_patch_rescan:NNn 5306, 5307, 5308, 5316
__ctex_patch_true:w 5196, 5202, 5338, 5371, 5383
__ctex_plus_key_aux:nn 2433, 2435, 2436, 2437, 2438, 2439, 2440, 2443
\l__ctex_prefix_str 5267, 5285, 5306, 5363, 5364
\l__ctex_prefix_tl 5306, 5311, 5322
\l__ctex_punct_bounds_clist ... 4944, 4950, 4956, 4962
\c__ctex_punct_family_clist 602, 612
\l__ctex_punct_font 4942, 4943
\c__ctex_punct_seq 4922, 4945
\l__ctex_punct_tl 275, 571, 572, 658, 946, 2096, 2097, 2098, 2099
__ctex_rebuild_cmd:w ... 5298, 5299, 5303, 5378, 5379
\l__ctex_replacement_str 5267, 5287, 5308, 5346, 5347, 5381
\l__ctex_replacement_tl 5308, 5313, 5324
__ctex_restore_selectfont: 1990, 1994, 2003
__ctex_right_hook_aux:nn 5375
__ctex_save_bounds:n 4946, 4954
__ctex_save_cmap:Nn 453, 457
__ctex_save_font_size:nn 3792
\l__ctex_scheme_tl 310, 3415, 3695
\g__ctex_section_depth_int 94, 304, 306, 3617, 3633, 3649
\c__ctex_section_headings_seq . 2359, 2365, 2999, 3113
__ctex_set_font_size:nnNn 3864, 3867
\g__ctex_spa_iow 4932, 4935, 4937, 4947
\c__ctex_std_class_tl ... 3666, 3681, 3682, 3748, 3771
\g__ctex_std_options_clist 131, 377, 389, 399, 401, 405, 410, 415, 420
__ctex_titlesec_format:n 3109, 3117, 3120
__ctex_titlesec_format:Nn 3099, 3123, 3125
__ctex_titlesec_format_explicit:Nn 3100, 3135
__ctex_titlesec_hook:n 3109, 3113, 3115
__ctex_titlesec_spacing:Nn 3118, 3145
__ctex_titlesec_spacing:nnnnnn 3146, 3147
__ctex_tmp:N 1986, 1997
__ctex_tmp:w 5207, 5260, 5271, 5292
\l__ctex_tmp_box 58, 1931, 1932, 3239, 3243
\l__ctex_tmp_dim 58, 2173, 2174, 2179, 2182, 2188, 2227, 2229, 2231, 2240, 2242
\l__ctex_tmp_int 58, 552, 554, 555, 557, 561, 563, 2187, 2189, 2190, 2196, 2197, 2229, 2230, 2231, 2234, 2235, 2240, 2242, 2998, 3001, 3005
\l__ctex_tmp_tl 58, 361, 368, 383, 384, 459, 460, 551, 553, 1988, 1999, 2405, 2463, 2476, 2486, 3747, 3748, 3836, 3837
\c__ctex_upper_case_file_str 1805, 1813
\c__ctex_version_tl 3, 6, 7, 8, 9, 20, 23
__ctex_write_family:nn 4934, 4939
\l__ctex_xkanjiskip_skip 2021
\l__ctex_xkanjiskip_tl 2028, 2033, 2034
\c__ctex_zero_tl 2147, 2154, 2156
\g__ctex_zhmCJK_bool 125, 255, 260, 265, 481, 575
\l__ctex_ziju_dim 2223, 2224, 2228, 2239, 2241, 2245, 2249
\l__ctex_ziju_tl 2147, 2155, 2156, 2217, 2223
\ctexbibname 2332, 2352
\CTEXcounter 2267
\CTEXdigits *26*, 2277
\CTEXfilltwosides *31*, 1901, 1911
\CTEXifname *16*, 2509, 2601, 2643, 2782, 2978, 2986, 2994, 3230, 3272, 3275, 3279, 3291, 3295, 3303, 3306, 3311, 3314
\CTEXindent *31*, 2250
\CTEXnoindent *31*, 2250
\CTEXnumber *26*, 2277
\CTEXnumberline 2992, 3012, 3494
\CTEXoptions *31*, 2112
\ctexset *5*, 2109, 2116, 2122
\CTEXsetfont *31*, 2142
\CTEXsetup *31*, 2112
\CTEXsout *31*
\ctexspadef 648, 4949
\CTEXthechapter *16*, 2725, 2987, 3165, 3291, 3311
\CTEXtheparagraph *16*
\CTEXthepart *16*, 2979, 3170
\CTEXthesection *16*, 3272, 3279, 3295, 3303, 3314
\CTEXthesubparagraph *16*
\CTEXthesubsection *16*, 3275, 3306
\CTEXthesubsubsection *16*
\CTEXunderdblline *31*
\CTEXunderdot *31*
\CTEXunderline *31*
\CTEXunderwave *31*
\CTEXxout *31*
\CurrentOption 389
\cwd 2043

D

declarecharrange *27*, 1558
\DeclareErrorKanjiFont 1677
\DeclareFontEncoding 948
\DeclareFontFamily 467, 957, 4066, 4067, 4068, 4069, 4070, 4071
\DeclareFontShape 958, 960, 1700, 1701, 1704, 1705, 4080, 4081, 4082, 4083, 4084, 4085, 4086, 4087, 4088, 4091, 4092, 4093, 4094, 4095, 4096, 4097, 4098, 4099, 4102, 4103, 4104, 4107, 4108, 4109, 4114, 4115, 4116, 4117, 4118, 4119, 4120, 4121, 4122, 4125, 4126, 4127, 4128, 4129, 4130, 4131, 4132, 4133, 4136, 4137, 4140, 4141, 4146, 4147, 4148, 4149, 4150, 4151, 4152, 4153, 4154, 4157, 4158, 4159, 4160, 4161, 4162, 4163, 4164, 4165, 4168, 4171
\DeclareFontSubstitution 951
\DeclareKanjiFamily 1698, 1699, 4072, 4073, 4074, 4075, 4076, 4077
\DeclareKanjiSubstitution 1678, 1679
\DeclareMathSizes 3829
\DeclareSymbolFont 963, 1321, 1680
\def 2575, 2615, 2650, 2670, 2685, 2716, 2757, 2767, 2792, 2812, 2838, 2842, 2878, 2893, 2908, 2909, 2913, 3889, 3900, 3928, 3939, 4614, 4650

\defaultCJKfontfeatures 1223
\defbeamertemplate 3320, 3340, 3360
\defbeamertemplatealias 3380, 3381, 3382
dim commands:
\dim_abs:n 2241
\dim_add:Nn 2224
\dim_compare:nNnTF . . 2174, 2209, 2225, 2232, 2493, 2952
\dim_eval:n 2194, 2240
\dim_max:nn 3242
\dim_min:nn 2241
\dim_new:N 61, 1941, 2044, 2045, 2046, 2249
\dim_set:Nn 1932, 1975, 1976, 1977, 2173, 2179, 2210, 2223, 2227, 2254, 2500, 2945, 3240, 3959
\dim_sub:Nn 2231
\dim_to_decimal:n 3800, 3801, 3838
\dim_to_decimal_in_unit:nn 4969
\dim_use:N 117, 3959
\dim_zero:N 2260
\c_max_dim 2174, 2493
\c_zero_dim 2182, 2194, 2209, 2225, 2232, 2418, 2952, 3650
\dimexpr 3329, 3349, 3369
\directlua 699, 700
\document 5125, 5128

E

\edef 4633, 4634
\else . 1687, 1688, 1692, 1693, 2547, 2556, 2565, 2581, 2586, 2621, 2626, 2704, 2712, 2728, 2734, 2740, 2753, 2763, 2823, 2828, 2847, 2852, 2872, 2877, 2886, 2907, 2923, 2936, 3457, 3485, 4645, 4646, 4692, 4731, 4762, 4821, 4884
else commands:
\else: 484, 724, 817, 863, 1960, 3065, 3276
\em 1685
\eminnershape 1687, 1688, 1693, 1695
\end 3336, 3356, 3376, 5014
\endCJKfilltwosides 1907
\endcsname 2830, 2839, 2840, 2841, 2848, 2867, 2868, 2870, 2873, 2882, 2883, 2884, 2887, 4640, 4642, 4643, 4644, 4647, 4649, 4653
\endCTEXfilltwosides 1907, 1912
\endgroup 2869, 2906, 2931, 3338, 3358, 3378, 4633, 4641, 4656
\endinput 4638
\endlinechar 4612, 4613
\everypar 2827, 2926, 2938
\EverySelectfont 2007, 2010
exp commands:
\exp_after:wN 116, 770, 786, 788, 863, 866, 3146, 3782, 3829, 3864, 5276
\exp_args:cc 3390
\exp_args:Nc 3118, 3123, 5230
\exp_args:Nco 3228
\exp_args:Nnc 102
\exp_args:NNf 1616
\exp_args:NNNo 3958
\exp_args:Nno 3402
\exp_args:No 553, 1814, 3142, 4946, 5292
\exp_args:Nx 288, 581, 655, 2214, 3778, 3957
\exp_last_unbraced:Nn 5241
\exp_last_unbraced:NNn 1998
\exp_last_unbraced:NNNNn 119
\exp_not:N 463, 583, 584, 585, 732, 733, 735, 805, 841, 951, 1036, 1037, 1038, 1039, 1395, 1516, 1744, 1745, 1746, 1760, 1896, 1897, 1898, 1925, 2379, 2380, 2381, 2386, 2388, 2389, 2390, 2391, 2394, 2395, 2398, 2410, 2411, 2412, 2413, 2414, 2423, 2424, 2425, 2427, 2430, 2445, 2447, 2449, 3004, 3006, 3007, 3008, 3009, 3130, 3131, 3140, 3141, 3178, 3180, 3181, 3798, 4000, 5221, 5222, 5223, 5311
\exp_not:n 69, 115, 383, 734, 736, 1032, 1033, 1034, 1035, 1168, 1169, 1396, 1397, 1398, 1422, 1516, 1739, 1761, 1981, 2431, 2486, 3149, 3391, 5104, 5128, 5239, 5311, 5312, 5313, 5368, 5369
\exp_stop_f: 722, 723
\expandafter 4615, 4616, 4640, 4641, 4642, 4643, 4644, 4649, 4653, 4654
\ExplSyntaxOff 66, 72, 472, 515, 519, 523, 527, 678, 1686, 2000, 3251, 3319, 4975, 5174
\ExplSyntaxOn 71, 710, 3383, 4915

F

\familydefault 1269, 1280, 1742
fancyhdr 30, 336
\fangsong 4594
\fi . . 2538, 2543, 2549, 2558, 2567, 2570, 2585, 2590, 2599, 2610, 2625, 2630, 2641, 2665, 2688, 2696, 2697, 2700, 2704, 2714, 2732, 2738, 2744, 2756, 2766, 2770, 2779, 2780, 2789, 2795, 2808, 2813, 2819, 2825, 2832, 2834, 2856, 2857, 2874, 2888, 2891, 2911, 2920, 2939, 2940, 3271, 3274, 3278, 3288, 3289, 3294, 3302, 3305, 3310, 3313, 3332, 3352, 3372, 3457, 3485, 4645, 4646, 4655, 4908
fi commands:
\fi: 402, 531, 568, 587, 726, 817, 863, 1960, 2005, 3068, 3280, 3296, 3873
\figurename 2313
figurename 12
file commands:
\g_file_curr_name_str 670, 1814
\file_if_exist:nTF 48, 1801, 1803, 1805, 1817, 3993, 4176, 4183, 4185, 4433
\file_if_exist_input:nTF 86
\file_input:n 78, 87, 89, 504, 540
fntef 31, 336
\font 4066, 4067, 4068, 4069, 4070, 4071
\fontfamily 1262
fontset 7, 220, 4013
\fontsize 3782
fontspec commands:
\g_fontspec_encoding_tl 956
\l_fontspec_family_tl 1455
\fontspec_set_family:Nnn 1034, 1432
\fontspec_visible_space: 720
\fontspec_visible_space_fallback: 725

fontspec internal commands:
__fontspec_fontname_wrap:n 56, 975
\footnotesep 3958
\footnotesize 3895, 3934, 3958
fp commands:
\fp_compare:nNnTF 3742, 3769, 3955
\fp_eval:n 2217, 4965
\fp_set:Nn 3743, 3770
\fp_use:N 2154, 3957
\c_nan_fp 177
\c_zero_fp 2154, 3742, 3769, 3955

G

GBK 6, 216
\gdef 3030, 3031, 3032, 3033, 3036, 3037, 3038, 3039
\global 2708, 2925, 2928
\globaljfont 788
\glueexpr 2864, 2879, 2901, 2908, 2909, 2935, 3327, 3337, 3347, 3357, 3367, 3377
group commands:
\group_begin: 467, 715, 750, 772, 798, 840, 867, 915, 980, 1030, 1341, 1426, 1511, 1531, 1547, 1738, 1985, 2048, 2385, 2393, 2461, 3054, 3187, 3246, 3958, 4941, 5122, 5198, 5206, 5215, 5270
\group_end: 471, 480, 719, 755, 776, 811, 844, 871, 919, 995, 1041, 1361, 1433, 1517, 1543, 1554, 1750, 1998, 2066, 2396, 2400, 2485, 3069, 3210, 3250, 3958, 4952, 5122, 5218, 5265, 5293, 5327, 5351
\group_insert_after:N 808, 809
\gtfamily 1687, 1692

H

hbox commands:
\hbox_set:Nn 1931, 3239
heading 9, 299
\heiti 4593
\hskip ... 2864, 2879, 2901, 2908, 2909, 2934, 2935, 3302, 3305
\hspace 2580, 2620, 2979, 2987, 3170
\Huge 2644, 2679, 2783, 2802, 3461, 3492, 3912, 3951
\huge 2602, 2640, 2659, 2778, 3443, 3460, 3464, 3490, 3498, 3911, 3950
hyperref 31, 336
\hypersetup 353, 354, 1827
\hyphenchar 4066, 4067, 4068, 4069, 4070, 4071

I

if commands:
\if_bool:N 481, 575
\if_case:w 398, 3869
\if_cs_exist:N 548, 1984
\if_false: 108
\if_meaning:w 1959
\if_true: 107
\ifctexpdf 106
\ifdim 2817, 2860, 2897, 2915
\IfNoValueF 2117, 2123
\IfNoValueTF 1183, 1204, 1480, 1585, 1618, 1620, 2457
\ifnum 2576, 2597, 2616, 2639, 2717, 2776, 2843, 2872, 2886, 3271, 3274, 3278, 3285, 3286, 3294, 3302, 3305, 3310, 3313, 4645
\ifodd 2543, 2545, 2570, 2577, 2610, 2617, 2665, 2688, 2710, 2721, 2770, 2789, 2795, 2808, 2821, 2832, 2848, 2861, 2898, 2916, 2920, 3330, 3350, 3370
\ifx 4643, 4644, 4649
\ifzhmappdf 4661, 4728, 4759, 4790, 4857
\ignorespaces 2941
\immediate 4652
indent 30, 196
\indexname 2316
indexname 12
\input 4914, 4978
\insertcontinuationtext 2325, 2334, 2341
\insertpart 3334, 3335
\insertromanpartnumber 3325, 3430
\insertsection 3354, 3355
\insertsectionnumber 3345, 3529
\insertsubsection 3374, 3375
\insertsubsectionnumber 3365, 3558
int commands:
\int_compare:nNnTF 554, 2190, 2234, 3617, 3633, 3649, 3740, 3767
\int_const:Nn 967
\int_div_truncate:nn 80, 561
\int_eval:n 1481, 1483, 1484, 1586, 1588, 1589
\int_from_hex:n 553
\int_gincr:N 1126, 1446
\int_gset:Nn 96, 156, 157, 158, 304, 306, 372, 376, 3741, 3768
\int_incr:N 3001
\int_mod:nn 563
\int_new:N 59, 95, 146, 1449
\int_set:Nn 147, 538, 544, 552, 2187, 2229
\int_sub:Nn 557, 2235
\int_to_Hex:n 555, 560, 562
\int_use:N ... 464, 544, 855, 1128, 1447, 2057, 3005, 4946
\int_zero:N 2998
\c_zero_int 2190
\interlinepenalty 2593, 2635, 2654, 2674, 2775, 2800, 2865, 2866, 2902, 2903
iow commands:
\iow_close:N 4935
\iow_indent:n 5109
\iow_new:N 4937
\iow_now:Nn 4947
\iow_open:Nn 4932
\itemsep 3892, 3903, 3931, 3942
\itshape 1687, 1688, 1692, 1693

J

\jfam 907, 968, 1682

K

\kaishu 4595
\kanjifamily 1655, 1662, 1669, 1674
\kanjifamilydefault 1650

kernel internal commands:
\l__kernel_expl_bool 70
keys commands:
\l_keys_choice_tl 289
\keys_define:nn 150, 383, 386, 972, 1334, 1443, 1520, 1558, 1597, 2071, 2092, 2125, 2159, 2282, 2308, 2336, 2346, 2349, 2352, 2486, 3014, 3966, 4013
\keys_if_exist:nnTF 1595
\l_keys_key_tl 135, 137, 139
\keys_set:nn 29, 166, 172, 228, 237, 246, 272, 282, 289, 296, 326, 332, 2110, 2117, 2123, 3418, 3476, 3505, 3537, 3572, 3589, 3603, 3619, 3627, 3635, 3643, 3650, 3651, 3653, 3719, 3721, 4055
\keys_set_known:nn 3698
\keys_set_known:nnN 984

L

\labelformat 3402
\languagealias 3727, 3732, 3753, 3758
\LARGE 3910, 3949
\Large 2598, 3441, 3447, 3516, 3520, 3909, 3948
\large 3541, 3908, 3947
\lastbox 2929
\leavevmode 2538, 2813
left commands:
\c_left_brace_str 5262
\leftmargin 3889, 3900, 3928, 3939
\leftmargini 3889, 3900, 3928, 3939
\let 1993, 2846, 2855, 3882, 3921, 4636, 4642
\linespread 3957
linespread 10, 176
linestretch 11, 2159
\linewidth 2188, 2196, 2228
\lishu 4597
\listfigurename 2311
listfigurename 12
\listtablename 2312
listtablename 12
\LoadClass 406, 411, 416, 421
\long 4650
\lstlistingname 733, 734
\lstlistlistingname 735, 736
\ltjalchar 723
\ltjdefcharrange . 679, 680, 681, 684, 688, 689, 692, 695, 696
LTJFONTUID 1443
\ltjgetparameter 1936, 1963, 2025, 2039
ltjitaliccorr internal commands:
__ctex_ltjitaliccorr 1637
\ltjsetkanjiskip 1641
\ltjsetmathletter 917
\ltjsetparameter 698, 704, 1636
\ltjsetxkanjiskip 1643
lua commands:
\lua_escape:e 816, 828, 855, 862, 1471, 1486, 1487, 1504, 1505, 1514
\lua_now:e 716, 790, 816, 820, 825, 830, 852, 861, 954, 1492, 1516, 1770, 2050

M

\MAKESPA 4938, 4980
\markboth 2508, 2604, 2633
\mathgroup 907
\mathrm 1653
\mathsf 1660
\mathtt 1667
\mcfamily 1692
\mddefault 952, 958, 964, 1322, 1327
mode commands:
\mode_if_math:TF 901
msg commands:
\msg_critical:nnn 53, 670, 3982
\msg_error:nn 4033
\msg_error:nnn 19, 22, 38, 41, 661, 674, 677, 2298, 3783, 3865
\msg_error:nnnn 3999, 4028
\msg_fatal:nn 4920
\msg_new:nn 4918
\msg_new:nnn 134, 136, 138, 140, 142, 664, 1024, 1094, 1176, 1883, 2329, 3672, 3983, 4042, 5105, 5182
\msg_new:nnnn 12, 30, 45, 100, 2300, 3785, 4007, 4036
\msg_warning:nn 1171, 1887, 2089, 2100, 2351
\msg_warning:nnn 164, 170, 198, 208, 223, 232, 241, 270, 280, 294, 324, 330, 338, 343, 349, 1089, 3691, 3970, 4020, 5111, 5181
\msg_warning:nnnn 1020, 1896, 1903, 2115, 2121, 2252, 2258

N

\newCJKfontfamily 1195
\NewDocumentCommand 1055, 1190, 1195, 1201, 1211, 1217, 1223, 1230, 1236, 1242, 1248, 1381, 1473, 1578, 1613, 1672, 2110, 2113, 2119, 2142, 2213, 2250, 2256, 2277, 2279, 2451, 3777, 3962, 4592, 4593, 4594, 4595, 4597, 4598, 4600, 4602, 4603
\newfontfeature 969, 970, 971
\newluafunction 2047
\newpage 2690, 2695
\ngostype 44, 1788
\nobreak 2598, 2607, 2662, 2778, 2783, 2786, 2802, 2805, 2917, 3442, 3491
\nobreakspace 2598, 2640
nocap 30
\noexpand 4634, 4636, 4637, 4638
nofonts 30
\noindent 2959
noindent 30
nopunct 30
\normalem 1866
\normalfont 527, 529, 530, 1227, 1234, 1240, 1305, 1684, 2594, 2596, 2636, 2638, 2655, 2657, 2675, 2677, 2772, 2774, 2797, 2799, 3009
\normalsize 3575, 3592, 3606, 3876, 3915, 3953, 3961
nospace 30
nozhmap 30

\null 2568, 2693
\numberline 2727, 2873, 2887, 2987, 2995, 3181, 3237

O

\onecolumn 2563
or commands:
\or: 400, 3871

P

\pagestyle 3718
\par 2499, 2539, 2598, 2640, 2644, 2659, 2679, 2778, 2783, 2802, 2814, 2917, 3326, 3327, 3334, 3337, 3346, 3347, 3354, 3357, 3366, 3367, 3374, 3377, 3420, 3442, 3453, 3471, 3479, 3491, 3532, 3534, 3566, 3568
paragraph/afterskip 32
paragraph/beforeskip 32
paragraph/hang 22
paragraph/numbering 15
paragraph/runin 20
\parindent 2209, 2210, 2232, 2253, 2254, 2259, 2260, 2594, 2655, 2772, 2797, 2945, 2952, 2953, 3329, 3349, 3369, 3651
\parsep 3891, 3892, 3902, 3903, 3930, 3931, 3941, 3942
\part 2536
part/beforeskip 32
part/fixbeforeskip 32
part/hang 22
part/numbering 15
part/pagestyle 22, 2461
\partmark 2507, 2605, 2634
\partname 2598, 2640, 3325, 3425
\PassOptionsToClass 405, 410, 415, 420
\PassOptionsToPackage 482, 1831, 2262, 5113
\pdfmapline 4662, 4663, 4664, 4665, 4666, 4667, 4668, 4670, 4671, 4672, 4673, 4674, 4675, 4676, 4678, 4679, 4680, 4681, 4682, 4683, 4685, 4686, 4687, 4688, 4689, 4690, 4791, 4792, 4793, 4794, 4795, 4796, 4797, 4799, 4800, 4801, 4802, 4803, 4804, 4805, 4807, 4808, 4809, 4810, 4811, 4812, 4814, 4815, 4816, 4817, 4818, 4819, 4858, 4859, 4860, 4861, 4862, 4863, 4865, 4866, 4867, 4868, 4869, 4870, 4872, 4873, 4874, 4875, 4876, 4878, 4879, 4880, 4881, 4882
\pdfoutput 4645
\pdfstringdefDisableCommands 3965
\pingfang 4603
prg commands:
\prg_do_nothing: 911, 1420, 1512, 1912, 1983, 2076, 2971, 5303
\prg_generate_conditional_variant:Nnn 1081
\prg_new_conditional:Npnn 814, 1956
\prg_new_protected_conditional:Npnn 1068, 5273
\prg_return_false: . . . 817, 1078, 1960, 1964, 1968, 5283
\prg_return_true: 817, 1071, 1076, 1960, 1964, 1968, 5288
\primitive 5014
\ProcessKeysOptions 395
\proofname 2303, 2318
proofname 12
prop commands:
\prop_get:NnN 1163
\prop_get:NnNTF 1070, 1122, 1160, 1413, 1461, 3781, 3836, 3863
\prop_gpop:NnNTF 1009, 1013, 1548
\prop_gput:Nnn . . . 988, 989, 1036, 1150, 1536, 1572, 3798
\prop_gremove:Nn 1018
\prop_if_empty:NTF 1084, 1286
\prop_map_break:n 1300
\prop_map_inline:Nn 1298
\prop_new:N 1004, 1005, 1006, 1155, 1556, 1577, 3792
\protect 2521, 2527, 2727, 2748, 2749, 2873, 2887, 2987, 2995, 3180, 3192, 3194, 3198, 3200, 3204, 3206, 3214, 3219, 3231, 3232, 5234, 5235
\ProvideDocumentCommand 2507
\ProvidesFile 4650, 4658, 4725, 4756, 4787, 4854
\providetranslation 5019, 5020, 5021, 5022, 5023, 5024, 5025, 5026, 5027, 5028, 5029, 5030, 5031, 5032, 5033, 5034, 5035, 5036, 5037, 5038, 5039, 5040, 5041, 5042, 5043, 5044, 5045, 5046, 5047, 5048, 5049, 5050, 5051, 5052, 5053, 5054, 5055, 5056, 5057, 5058, 5059, 5060, 5061, 5062, 5063, 5064, 5065, 5066, 5067, 5068, 5069, 5070, 5071, 5072, 5073, 5074, 5075, 5076, 5077, 5078, 5079, 5080, 5081, 5082, 5083, 5084, 5085, 5086, 5087, 5088, 5089, 5090, 5091, 5092, 5093, 5094, 5095, 5096, 5097, 5098
punct 10, 30, 275, 2092
\punctstyle 572, 2097

Q

\quad 2839, 3271, 3272, 3274, 3275, 3278, 3279, 3291, 3295, 3303, 3306, 3311, 3314, 3448, 3499, 3509, 3542, 3576, 3593, 3607
quark commands:
\q_mark 5277, 5280
\q_stop 117, 120, 866, 869, 1378, 1382, 1609, 1614, 5242, 5247, 5256, 5277, 5280
\quotation 3975

R

\raggedright 2594, 2655, 2772, 2797, 3440, 3489
\refname 2319, 2324, 2331, 2340, 2348, 2349
refname 12
\refstepcounter . . . 2579, 2619, 2723, 2850, 3385, 3407, 3408
\relax 2616, 2639, 2815, 2851, 2859, 2864, 2879, 2896, 2901, 2908, 2909, 2914, 2933, 2935, 3302, 3305, 3327, 3329, 3337, 3347, 3349, 3357, 3367, 3369, 3377, 4608, 4612, 4615, 4643, 4644, 4649
\renewcommand 2537, 2553, 2703
\RenewDocumentCommand 1262, 1651, 1658, 1665, 3408
\RequirePackage 27, 56, 57, 345, 356, 483, 486, 487, 488, 654, 672, 675, 712, 732, 1731, 1840, 1849, 1865, 1871, 2263, 3772, 5158
\RequirePackageWithOptions 5114
resetalternatefont 28, 1520
reverse commands:
\reverse_if:N 575

\rmdefault 1277, 1654, 1744
\rmfamily 515, 518, 1651, 1653
\romanfamily 1654, 1661, 1668

S

scan commands:
\scan_stop: 65, 117, 788, 907, 1513, 1514, 2180, 2188, 4942, 5124
scheme 9, 310
\scriptsize 3906, 3945
\secdef 2550, 2572, 2715
section/afterskip 32
section/beforeskip 32
section/hang 22
section/numbering 15
section/runin 20
\sectionname 3345, 3526
\selectfont 1062, 1539, 1552, 1656, 1663, 1670, 1675, 1997, 2004, 2018, 2144, 3782
seq commands:
\seq_clear:N 982
\seq_concat:NNN 1354
\seq_const_from_clist:Nn . 434, 2360, 2370, 3044, 4922
\seq_gput_left:Nn 2366, 2367
\seq_gput_right:Nn 1088, 3804
\seq_gset_eq:NN 2365
\seq_if_empty:NTF 1369
\seq_if_in:NnTF 440, 1086
\seq_map_function:NN 3113, 3235
\seq_map_inline:Nn 1377, 2478, 2999, 3047, 3057, 3066, 3089, 3172, 3396, 4945
\seq_new:N 1093, 1362, 1363, 2364, 3793
\seq_put_right:Nn 1625
\seq_set_filter:NNn 1352
\seq_set_split:Nnn 1351
\seq_use:Nnnn 3790
\setbox 2929
\setCJKfamilyfont 1107, 1195, 4204, 4210, 4211, 4212, 4213, 4214, 4215, 4258, 4259, 4267, 4268, 4270, 4271, 4272, 4273, 4275, 4297, 4299, 4301, 4303, 4342, 4343, 4344, 4345, 4369, 4374, 4379, 4381, 4423, 4425, 4427, 4428, 4460, 4461, 4464, 4469, 4470, 4472, 4473, 4474, 4487, 4488, 4489, 4490, 4491, 4492, 4532, 4533, 4534, 4535, 4536, 4537, 4550, 4551, 4552, 4585, 4586, 4587
\setCJKmainfont 1103, 1223, 4195, 4254, 4262, 4289, 4335, 4357, 4414, 4445, 4449, 4483, 4528, 4546, 4581
\setCJKmathfont 1223
\setCJKmonofont 1105, 1223, 4209, 4257, 4266, 4296, 4341, 4368, 4422, 4458, 4486, 4531, 4549, 4584
\setCJKromanfont 1229, 1255
\setCJKsansfont 1104, 1223, 4198, 4202, 4256, 4264, 4295, 4340, 4363, 4420, 4446, 4456, 4485, 4530, 4548, 4583
\SetSymbolFont 965, 1326, 1330, 1681
\sfdefault 1278, 1661, 1745
\sffamily 519, 522, 1658, 1660
skip commands:
\skip_horizontal:N 1946
\skip_horizontal:n 2953
\skip_if_eq:nnTF 1963, 1967, 2024, 2518, 2524
\skip_new:N 1955, 2037, 2503
\skip_set:Nn 1936, 1939, 2028, 2038, 2181, 2192, 2237, 2245, 2505, 2517, 2523
\skip_set_eq:NN 1953, 2030
\skip_sub:Nn 2494, 2501
\skip_use:N 2521, 2527
\skip_zero:N 2201
\c_zero_skip 2416, 2417, 2469, 2470, 2518, 2524
\small 3884, 3923
\songti 4592
\space 2724, 2778, 3425, 3487, 3526, 3557, 3659
space 10, 30, 285, 2071
\special 1713, 1714, 1717, 1718, 1847, 1861, 4693, 4694, 4695, 4696, 4697, 4698, 4699, 4701, 4702, 4703, 4704, 4705, 4706, 4707, 4709, 4710, 4711, 4712, 4713, 4714, 4716, 4717, 4718, 4719, 4720, 4721, 4732, 4733, 4734, 4735, 4736, 4738, 4739, 4740, 4741, 4742, 4744, 4745, 4746, 4747, 4749, 4750, 4751, 4752, 4763, 4764, 4765, 4766, 4767, 4769, 4770, 4771, 4772, 4773, 4775, 4776, 4777, 4778, 4780, 4781, 4782, 4783, 4822, 4823, 4824, 4825, 4826, 4827, 4828, 4830, 4831, 4832, 4833, 4834, 4835, 4836, 4838, 4839, 4840, 4841, 4842, 4843, 4845, 4846, 4847, 4848, 4849, 4850, 4885, 4886, 4887, 4888, 4889, 4891, 4892, 4893, 4894, 4895, 4897, 4898, 4899, 4900, 4902, 4903, 4904, 4905
\SplitArgument 1474, 1579, 2452
str commands:
\c_backslash_str 5261
\c_colon_str 893, 894, 895
\str_case:nnTF 1742, 1788, 5232, 5247
\str_case_e:nn 1275
\str_case_e:nnTF 1101, 1112
\str_const:Nn 43, 50, 1813, 1821
\str_if_empty:NTF 5318, 5354, 5363
\str_if_eq:eeTF 571, 1290, 4019, 5238
\str_if_eq:nnTF 485, 576, 940, 1269, 1836, 1844, 3723, 3750, 3764, 4022, 4039, 4046
\str_lower_case:n 459
\str_new:N 5158, 5267, 5268, 5269
\str_set:Nn 5285, 5286, 5287
\str_upper_case:n 1814
\strutbox 3959
sub3section 9, 302
sub4section 9, 302
subparagraph/afterskip 32
subparagraph/beforeskip 32
subparagraph/hang 22
subparagraph/numbering 15
subparagraph/runin 20
subsection/afterskip 32
subsection/beforeskip 32
subsection/hang 22

subsection/numbering 15
subsection/runin 20
\subsectionname 3365, 3557
subsubsection/afterskip 32
subsubsection/beforeskip 32
subsubsection/hang 22
subsubsection/numbering 15
subsubsection/runin 20
\symmincho 1682
sys commands:
\c_sys_engine_str 44
\sys_if_engine_pdftex:TF 124, 4190, 4282, 4350, 4438, 4479, 4542
\sys_if_engine_uptex:TF 4235, 4323, 4402, 4441, 4512, 4570
\sys_if_engine_xetex:TF 4916
\sys_if_output_pdf:TF 106, 469, 1839, 4284, 4352

T

\tablename 2314
tablename 12
\TeX 3963
TEX and LATEX 2ε commands:
\@@italiccorr 1638
\@@par 2865, 2902, 3510, 3543, 3577, 3621, 3637
\@afterheading 2612, 2667, 2755, 2765, 2922
\@afterindentfalse 2548, 2709, 2713, 2818, 2824
\@afterindenttrue 2546, 2711, 2816, 2822
\@Alph 3655, 3660
\@chapapp 2724, 2778, 3287, 3310
\@chapter 2715, 2716
\@chinese 2265, 2274
\@clubpenalty 2937
\@currentHref 3078
\@currentlabel 3387, 3391
\@currext 4, 5104, 5153
\@currname 4, 5104, 5154
\@dblarg 2837
\@empty 102, 2846, 2855
\@endpart 2647, 2682, 2684
\@EverySelectfont@Init 1984, 1992
\@firstoftwo 3231
\@hangfrom 2947, 2958
\@ifclassloaded 3668
\@ifpackagelater 37, 40, 660, 673, 676, 1872
\@ifpackageloaded 18, 21, 669, 1824, 3152, 3255, 3404, 5135
\@ifpackagewith 3097, 3106, 3157
\@ifstar 2835
\@listI 3882, 3921
\@listi 3882, 3889, 3900, 3921, 3928, 3939
\@M 2593, 2635, 2654, 2674, 2775, 2800, 2865, 2866, 2902, 2903, 2930
\@makechapterhead 2752, 2754, 2767
\@makeschapterhead 2762, 2764, 2792
\@minus 3511, 3544, 3578, 3594, 3608, 3878, 3880, 3886, 3888, 3890, 3891, 3897, 3899, 3901, 3902, 3917, 3919, 3925, 3927, 3929, 3930, 3936, 3938, 3940, 3941
\@namedef . 3192, 3194, 3198, 3200, 3204, 3206, 3214, 3219
\@nameuse 3192, 3194, 3204, 3206, 3214
\@ne 3150, 3274, 3305
\@nil 841
\@nobreakfalse 2924
\@noskipsecfalse 2928
\@noskipsectrue 2925
\@onlypreamble 37, 431, 496, 1251, 1252, 1253, 1254, 1255, 1256, 1708, 1722, 1730, 4006
\@part 2550, 2572, 2574
\@pkgextension 6, 7, 8, 9, 20, 23, 671, 5153
\@plus 3454, 3455, 3511, 3512, 3544, 3545, 3578, 3579, 3594, 3608, 3622, 3638, 3878, 3879, 3880, 3886, 3887, 3888, 3890, 3891, 3897, 3898, 3899, 3901, 3902, 3917, 3918, 3919, 3925, 3926, 3927, 3929, 3930, 3936, 3937, 3938, 3940, 3941
\@popfilename 134, 5151
\@schapter 2715, 2757
\@seccntformat 2838, 2851
\@secondoftwo 3232
\@secpenalty 2829, 3514, 3547, 3581, 3595, 3609
\@sect 2837, 2842
\@setfontsize 3868
\@spart 2550, 2572, 2649
\@ssect 2836, 2893
\@startsection 94, 96, 103, 2812, 3004
\@svsec 2846, 2851, 2855, 2864, 2880, 2881
\@svsechd 2878, 2908, 2909, 2931
\@tempdima 3240, 3242
\@tempskipa 2815, 2817, 2818, 2829, 2859, 2860, 2896, 2897, 2914, 2915, 2918, 2933, 2934
\@tempswafalse 2566
\@tempswatrue 2564
\@topnewpage 2752, 2762
\@topnum 2708
\@xsect 2892, 2912, 2913
\@zhdig 2273
\@zhnum 2265, 2272
\abstractname 12
\addCJKfontfeature 65
\AfterEndPreamble 133
\algorithmname 12
\appendixname 12, 24
\AtBeginDocument 37, 46
\AtBeginDvi 43
\AtBeginShipoutFirst 43
\AtEndOfClass 37
\AtEndOfPackage 37
\AtEndOfPackageFile* 133
\AtEndPreamble 133
\baselineskip 10, 11, 77, 80
\bfseries 48
\bibname 12

\c@chapter 3660
\c@secnumdepth 2576, 2597, 2616, 2639, 2717, 2776, 2843, 2872, 2886, 3271, 3274, 3278, 3285, 3286, 3294, 3302, 3305, 3310, 3313
\c@section 3655
\caption 31
\catcode 36, 45, 45, 124, 134
\ccwd . . . 11, 11, 25, 26, 31, 31, 37, 75, 75, 76, 79, 80, 80, 81
\chapter 9, 15, 16, 22, 24, 24, 86, 93, 107
\chaptermark 13
\chaptername 15
\char 50
\CheckCommand 76
\Chinese 31
\chinese 26, 31, 82
\CJK@@ignorespaces 47, 589
\CJK@encoding 55, 57, 64, 747, 751, 835, 947, 1321, 1324, 1326, 1330, 1471, 1486, 1487, 1504, 1505
\CJK@envStart 46, 46, 47, 583
\CJK@family 50, 739, 752, 799, 835, 1059, 1538, 1548, 1550, 2018, 2019
\CJK@ignorespaces 47, 47, 594, 598
\CJK@input 46, 532
\CJK@loadBinding 45, 46, 579
\CJK@loadEncoding 45
\CJK@makeActive 46, 578
\CJK@plane 45, 76, 448, 449, 459, 2007
\CJK@surr 45, 548
\CJK@upperReset 47
\CJKfamily 57
\CJKfamilydefault 46, 47, 61, 61, 69, 70, 70, 71, 71
\CJKglue 80, 112
\CJKhook 47, 47
\CJKrmdefault 44
\CJKsymbol 76
\CJKunderdotbasesep 31
\CJKunderline 31
\clearpage 47
\contentsname 11, 12
\CS 51
\CTEX@addloflotskip 2514, 2750
\CTEX@addtocline . . . 2531, 2592, 2632, 2746, 2876, 2890
\CTEX@afterindent 2821, 2965, 2972
\CTEX@aftertitle 2905, 2910, 2964, 2971
\CTEX@appendix@number 3019, 3031, 3037
\CTEX@appendix@numbering 3020, 3033, 3039
\CTEX@chapter@afterindent 2710
\CTEX@chapter@aftername 2782
\CTEX@chapter@afterskip 2788, 2807
\CTEX@chapter@aftertitle 2785, 2804
\CTEX@chapter@beforeskip 2769, 2794
\CTEX@chapter@break 2706
\CTEX@chapter@fixskip 2770, 2789, 2795, 2808
\CTEX@chapter@format 2774, 2799
\CTEX@chapter@lofskip 2467
\CTEX@chapter@lotskip 2468
\CTEX@chapter@numbering 2721, 3039
\CTEX@chapter@pagestyle 2466, 2707
\CTEX@chapter@titleformat 2784, 2803
\CTEX@chapter@tocline 2976
\CTEX@chaptername 2782
\CTEX@disableautoindent 2533, 2595, 2637, 2656, 2676, 2773, 2798, 3322, 3342, 3362
\CTEX@document@left@hook 5121, 5127
\CTEX@document@right@hook 5123, 5129
\CTEX@Family@CMap 445, 473, 476
\CTEX@fixheadingskip 2492, 2497, 2543, 2610, 2665, 2688, 2789, 2808, 2832, 2920
\CTEX@fixskip 2832, 2920, 2966, 2973
\CTEX@fixtopskip 2489, 2570, 2770, 2795
\CTEX@fontfamily 1033, 1262
\CTEX@gettitle 2591, 2631, 2653, 2673, 2745, 2760, 2858, 2895, 3084
\CTEX@hang 2957, 2967, 2974
\CTEX@hangfrom 2864, 2901, 2955
\CTEX@hangindent 2600, 2642, 2658, 2678, 2781, 2801, 2943
\CTEX@headingskip 2506, 2544, 2571, 2611, 2666, 2689, 2771, 2790, 2796, 2809, 2833, 2921
\CTEX@hyperheadinghook 3052, 3082
\CTEX@ifnamefalse 2509, 2582, 2587, 2622, 2627, 2651, 2671, 2729, 2735, 2741, 2758, 2844, 2853, 2894
\CTEX@ifnametrue 2509, 2578, 2618, 2722, 2849
\CTEX@makeanchor 2583, 2588, 2623, 2628, 2730, 2854, 3042, 3050, 3062, 3067, 3073
\CTEX@makeanchor@chapter 2736, 2742
\CTEX@makeanchor@schapter 2759
\CTEX@makeanchor@sect 2845
\CTEX@makeanchor@spart 2652, 2672
\CTEX@makeanchor@ssect 2836
\CTEX@part@afterindent 2545
\CTEX@part@aftername 2601, 2643, 3331
\CTEX@part@afterskip 2609, 2664, 2687, 3337
\CTEX@part@aftertitle . . . 2606, 2646, 2661, 2681, 3335
\CTEX@part@beforeskip 2542, 2569, 3327
\CTEX@part@break 2540, 2559
\CTEX@part@fixskip 2543, 2570, 2610, 2665, 2688
\CTEX@part@format 2596, 2638, 2657, 2677, 3328
\CTEX@part@indent 3329
\CTEX@part@numbering 2577, 2617, 3330
\CTEX@part@pagestyle 2465, 2561
\CTEX@part@titleformat . . 2603, 2645, 2660, 2680, 3335
\CTEX@part@tocline 2976
\CTEX@partname 2601, 2643, 3331
\CTEX@postappendix 3024, 3032, 3038
\CTEX@postchapter 3038
\CTEX@postsection 3032
\CTEX@preappendix 3023, 3030, 3036
\CTEX@prechapter 3036
\CTEX@presection 3030
\CTEX@runin 2861, 2898, 2916, 2968, 2975
\CTEX@save@appendix 3025, 3028
\CTEX@save@refstepcounter 3407, 3410

\CTEX@section@aftername 3351
\CTEX@section@afterskip 3357
\CTEX@section@aftertitle 3355
\CTEX@section@beforeskip 3347
\CTEX@section@format 3348
\CTEX@section@indent 3349
\CTEX@section@numbering 3033, 3350
\CTEX@section@titleformat 3355
\CTEX@sectionname 3351
\CTEX@selectfont@hook 1978, 2007, 2010
\CTEX@selectfont@save 1989, 1993
\CTEX@setcurrentlabel@n 3385, 3411
\CTEX@setheadingskip 2504, 2542, 2569, 2609, 2664, 2687, 2769, 2788, 2794, 2807, 2831, 2919
\CTEX@subsection@aftername 3371
\CTEX@subsection@afterskip 3377
\CTEX@subsection@aftertitle 3375
\CTEX@subsection@beforeskip 3367
\CTEX@subsection@format 3368
\CTEX@subsection@indent 3369
\CTEX@subsection@numbering 3370
\CTEX@subsection@titleformat 3375
\CTEX@subsectionname 3371
\CTEX@thechapter 3037
\CTEX@thesection 3031
\CTEX@titleformat@n 94, 2904, 2910, 2963, 2970
\CTEX@titlepslabel@clear 3201, 3211
\CTEX@titlepslabel@set 3195, 3207, 3211
\CTEX@toc@width@n 3237, 3252, 3262
\CTEX@todayold 2281, 2286
\CTEX@update@sectionformat@n 2820, 2961
\CTEX@verbatim@font@hook 1631, 1635
\ctex@zhmap@endinput 4910
\CTEXdigits 26
\CTEXifname 16
\CTEXnumber 26, 26
\CTEXnumberline 23
\CTEXoptions 31, 31, 79
\ctexset 5, 5, 7, 11, 14, 27, 28, 31, 31, 114
\CTEXsetup 31
\CTEXthechapter 13, 16, 98
\CTEXthesection 13, 16, 100
\CTEXunderdotbasesep 31
\CTEXunderline 31
\curr@fontshape 753, 783, 792, 796, 842
\DeclareAlternateKanjiFont 64
\DeclareFontEncoding 54
\DeclareFontFamily 43, 52
\DeclareFontShape 52
\DeclareFontShape@ 64, 1430
\DeclareRobustCommand 135
\define@newfont 51, 775
\do@subst@correction 51, 52, 774
\em 69
\escapechar 54
\EverySelectfont 76
\external@font 785, 786, 788
\extract@font 51, 773
\f@baselineskip 2180
\f@encoding 751, 791, 904
\f@family 50, 752, 791, 799, 1264, 1275
\f@series 747, 835
\f@shape 747, 835
\f@size 747, 753, 830, 838, 841, 842, 1932, 1975, 3243
\fangsong 8
\figurename 12
\font 52
\font@name 50, 52, 756, 763, 764, 767, 770, 788, 789, 806, 884, 885, 886, 887, 888, 895
\fontcharwd 76
\fontfamily 61
\footnotesep 112, 112
\footnotesize 112
\get@external@font 782
\getanddefine@fonts 873
\globaljfont 51
\H@old@chapter 3056
\hbox 86
\heiti 8
\hrule 86
\Hy@chapapp 2730, 2736, 2742, 2759
\Hy@driver 1844
\Hy@MakeCurrentHrefAuto 3075
\Hy@org@chapter 3056
\Hy@raisedlink 3076
\Hy@unicodetrue 73
\hyper@anchorend 3079
\hyper@anchorstart 3078
\HyPsd@ConvertToUnicode 73
\HyPsd@LoadUnicode 73
\HyPsd@pdfencoding 73
\HyPsd@ToBigChars 73
\if@mainmatter 2719, 2777, 3285, 3686, 3687
\if@nobreak 2826
\if@noskipsec 2538, 2813, 2927
\if@openright 2554, 2692, 2704, 3457, 3485
\if@tempswa 2698
\if@twocolumn 2562, 2751, 2761
\if@twoside 2691, 3269, 3292
\iffalse 54
\ifHy@implicit 3055
\ifin@ 54, 863
\iftrue 54
\indexname 12
\insertcontinuationtext 12
\itdefault 52
\itshape 48
\jfam 69
\kaishu 8
\kanjiskip 77
\labelformat 102
\leftmark 13

\linewidth 80
\lishu 8
\listfigurename 12
\listtablename 12
\ltj@@does@alt@set 52
\ltj@@getjfontnumber 53, 851
\ltj@@set@stackfont 893, 894, 895
\ltj@allalchar 717
\ltj@curjfnt 51, 76
\ltj@pickup@altfont@auxy 836
\ltj@pickup@altfont@copy 849
\ltj@setpar@global 68, 892, 1641, 1643
\ltj@tempcntc 53, 855
\ltjalchar 50
\ltjsetkanjiskip 68
\ltjsetparameter 68
\ltjsetxkanjiskip 68
\m@ne 2576, 2597, 2717, 2776, 3278, 3285, 3286, 3310, 4066, 4067, 4068, 4069, 4070, 4071
\markboth 13
\markright 13
\math@bgroup 903
\math@egroup 908
\math@fonts 890
\mathbb 113
\maxdimen 11, 80
\meaning 135, 136, 136, 137
\newcommand 135
\newrobustcmd 135
\newtitlemark 99
\nfss@catcodes 55
\ngostype 71
\normalsize 8, 108
\not@math@alphabet 1653, 1660, 1667
\nouppercase 45
\NR@gettitle 3088
\p@ 1932, 1975, 2640, 2748, 2749, 2768, 2778, 2787, 2793, 2806, 3243, 3453, 3481, 3482, 3483, 3484, 3491, 3878, 3879, 3880, 3886, 3887, 3888, 3890, 3891, 3897, 3898, 3899, 3901, 3902, 3917, 3918, 3919, 3925, 3926, 3927, 3929, 3930, 3936, 3937, 3938, 3940, 3941
\pagenumbering 26
\pagestyle 10
\paragraph 9, 20, 36
\parindent 10, 31, 79, 81, 112
\parskip 86
\part 14, 20, 21, 22, 86, 93
\partmark 86
\partname 15
\partpage 14, 101
\PassOptionsToClass 43
\PassOptionsToPackage 72
\pdfmapline 118, 121
\pdfstringdef 73
\pickup@font 50
\pickup@jfont 779
\pingfang 8
\ProcessKeysOptions 29
\proofname 12, 82
\protected 99
\protected@edef 2278, 2280, 2851, 3387
\ProvidesFile 125
\ps@fancy 3298
\ps@headings 3268
\refname 12, 12
\refstepcounter 102
\relax 45, 79
\reset@font 529
\rightmark 13
\rmfamily 48
\roman 26
\scantokens 135
\scriptfont@name 887, 894
\section 9, 14, 15, 20, 20, 21, 21, 22, 23, 24, 24, 32, 84, 100
\section* 15
\sectionmark 13
\sectionname 15
\sectionpage 14, 101
\selectfont 53, 70, 76, 77, 79
\setCJKfamilyfont 27, 27
\setCJKmainfont 27
\sf@size 886
\sffamily 48
\size@update 79, 80, 2001, 2002, 2144, 2158
\sldefault 52
\songti 8
\special 7, 72
\split@name 841
\SplitArgument 85
\ssf@size 888
\strutbox 112
\subparagraph 9, 20, 36
\subsectionname 15
\subsectionpage 14, 101
\subst@correction 802, 809
\tablename 12
\textfont@name 885, 893
\tf@size 884
\thechapter 13, 16, 24
\thesection 13, 16, 24, 100
\titleformat 96
\titlespacing 96
\today 11, 11
\trans@languagepath 3735, 3761
\ttfamily 48
\ttl@a 3170, 3178
\ttl@chapterout 3165
\ttl@extract 97
\ttl@setifthe 3224
\ttl@setsubmark 98, 3197, 3203
\ttl@settopmark 98, 3191, 3222
\ttl@tocpart 3167, 3169

\ttlh@hang 3131, 3141
\ttlh@runin 3130, 3140
\uppercase *45, 46*
\use@mathgroup 899, 930
\usepackage *30, 31*
\verbatim@font 1629, 1633
\wrong@fontshape *51*
\x@protect 5235
\xdef *99*
\XeTeXglyphbounds *129*
\xkanjiskip *77*
\yahei *8*
\youyuan *8*
\z@ 704, 707, 2594, 2655, 2708, 2772, 2797, 2817, 2860, 2897, 2915, 2929, 3150, 3271, 3294, 3302, 3313, 3879, 3887, 3898, 3918, 3926, 3937
\zhdig *82*
\zhdigits *26*
\zhnum *82*
\zhnumber *26, 26*
\zihao *25*
\ziju *25*
\zw *76*

tex commands:
\tex_def:D 5311
\tex_dimexpr:D 117, 2188
\tex_divide:D *80*, 2189, 2230
\tex_edef:D 5365
\tex_endlinechar:D 538, 544
\tex_font:D 463, 722, 4942
\tex_glueexpr:D 2180
\tex_iffontchar:D 722
\tex_iftrue:D 3687
\tex_ignorespaces:D 1056, 1215, 1220, 2082, 2214, 3778, 5124
\tex_immediate:D 460
\tex_kanjiskip:D 1939, 1953, 1967
\tex_luafunction:D 2068
\tex_pagegoal:D 2493
\tex_parskip:D 2501
\tex_pdffontattr:D 463
\tex_pdflastobj:D 464
\tex_pdfobj:D 460
\tex_prevdepth:D 2500
\tex_topskip:D 2494
\tex_XeTeXcharglyph:D 4946
\tex_XeTeXglyphbounds:D 4970
\tex_xkanjiskip:D 2026, 2030, 2040
\the 4612, 4615, 4616, 4635, 4639
\thechapter 2724, 2727, 2778, 3287, 3310, 3488
\theparagraph 3591
\thepart 2580, 2598, 2620, 2640, 3427
\thesection 3271, 3278, 3294, 3302, 3313, 3508
\thesubparagraph 3605
\thesubsection 3274, 3305, 3540
\thesubsubsection 3574
\thispagestyle 2560, 2561, 2694, 2705, 2707
\tiny 3907, 3946

tl commands:
\c_space_tl 464
\tl_clear:N 188, 384, 1570, 1574, 1868, 1879, 2019, 2135, 2458, 2476, 2534, 3833, 5319
\tl_clear_new:N 58, 361
\tl_const:Nn 3, 404, 409, 414, 419, 511, 649, 671, 947, 962, 1258, 1259, 1260, 1261, 1320, 2154, 2377, 2383, 3669, 3688, 3690
\tl_gput_right:Nn 441, 5118, 5120, 5142
\tl_gset:Nn 555, 558, 884, 886, 888, 1294, 1740, 1768, 1786, 1811, 1818, 1819, 4024
\tl_gset_eq:NN 1038, 1763
\tl_gset_rescan:Nnn 1301
\tl_head:N 3837
\tl_if_blank:nTF 1387, 1483, 1484, 1588, 1589, 1603, 1621
\tl_if_blank_p:n 1353
\tl_if_empty:NTF 742, 2207, 2389, 3401, 3987
\tl_if_empty:nTF 121, 1702, 1715, 5282
\tl_if_eq:NNTF 1736, 2147, 3748
\tl_if_exist:NTF 84, 510, 512, 513, 514, 1257, 1259, 1260, 1261, 2303, 2334, 2345, 2348, 3681, 3771, 5141
\tl_if_in:NnTF 5346
\tl_map_inline:Nn 1499
\tl_map_inline:nn 3834
\tl_new:N 122, 126, 512, 513, 514, 744, 745, 833, 944, 997, 1023, 1044, 1045, 1066, 1435, 1576, 1756, 2033, 2155, 2168, 2305, 2375, 2376, 3023, 3024, 4181, 5131, 5132, 5141, 5202, 5203, 5322, 5323, 5324
\tl_put_left:Nn 1992, 5151
\tl_put_right:Nn 368, 890, 1583, 1616, 2405, 2447, 2449, 3837, 3838
\tl_replace_all:Nnn 1350, 1391
\tl_rescan:nn *137*, 5364, 5373
\tl_set:Nn 112, 123, 183, 216, 217, 459, 491, 492, 493, 551, 733, 735, 746, 785, 834, 941, 942, 983, 1049, 1061, 1203, 1264, 1349, 1388, 1390, 1431, 1604, 1608, 1646, 1647, 1648, 1649, 1650, 1758, 1988, 2018, 2034, 2096, 2105, 2131, 2169, 2217, 2306, 2456, 2459, 2463, 3127, 3146, 3165, 3170, 3178, 3747, 3965, 4182, 4186, 4229, 4230, 4231, 5196, 5197
\tl_set_eq:NN 751, 752, 799, 885, 887, 956, 1050, 1075, 1538, 1550, 2156
\tl_set_rescan:Nnn 5320
\tl_tail:N 3132
\tl_tail:n 3142
\tl_to_str:n 120, 5263, 5292, 5333, 5334
\tl_trim_spaces:n 1567
\today 2281, 2286, 2289, 2294
today *11*, 2281

token commands:
\token_to_meaning:N 5277
\token_to_str:N *54*, 866, 875, 884, 886, 888, 1103, 1104, 1105, 1107, 1114, 1115, 1116, 1178, 2331, 2332, 3786, 4949, 5109, 5181

\toks 4612, 4613, 4615, 4616, 4635, 4639
\topsep 3890, 3901, 3929, 3940
\TrimSpaces 2113
\ttdefault 1279, 1668, 1746
\ttfamily 523, 526, 1665, 1667
\twocolumn 2699
\typeout 2724, 2725, 3165

U

um commands:
\um_input_math_symbol_table: 923, 927
\um_sym:nnn 926
um internal commands:
__um_input_math_symbol_table: 935
__um_switchto_literal: 932
__um_sym:nnn 934
\undefined 4637
\unless 2861, 2898, 2916
\unskip 2932
\updefault . . . 952, 958, 960, 964, 966, 1322, 1324, 1327, 1331
\upshape 1692, 1695
use commands:
\use:N 753, 791, 792, 842, 884, 886, 888, 967, 1015, 1551, 2517, 2523, 2532, 2945, 2995, 5347, 5367, 5381
\use:n 64, 127, 383, 730, 841, 949, 1133, 1186, 1192, 1291, 1295, 1393, 1567, 1757, 1910, 2483, 2959, 2970, 3248, 3679, 3796, 3996, 5216, 5258, 5309, 5330
\use:nn 5375
\use_i:nn 109, 863, 2511, 5162, 5167, 5187, 5192
\use_i:nnn 256
\use_ii:nn . . . 110, 863, 2509, 2513, 5162, 5167, 5187, 5192
\use_ii:nnn 128, 261
\use_iii:nnn 266
\use_none:n 184, 189, 194, 1032, 1033, 1294, 1428, 1911, 2267
\use_none:nn 476
\usebeamercolor 3325, 3345, 3365, 3470, 3531, 3565
\usebeamerfont 3324, 3334, 3344, 3354, 3364, 3374, 3469, 3472, 3530, 3533, 3564, 3567
\uselanguage 3726, 3731, 3752, 3757
\usepackage 5109
UTF8 . . . . . *6*, 216

V

\verse 3975
\vfil 2568, 2686
\vskip 2608, 2611, 2640, 2663, 2666, 2689, 2778, 2787, 2790, 2806, 2809, 2918, 2921, 3326, 3346, 3366, 3453, 3471, 3491, 3532, 3566
\vspace 2571, 2768, 2771, 2793, 2796

W

winfonts *30*
\write 4652

X

\x 4614, 4617, 4618, 4619, 4620, 4621, 4622, 4623, 4624, 4625, 4626, 4627, 4628, 4629, 4630, 4631, 4632, 4633, 4640, 4650, 4654
\xdef 4653
\xeCJKsetup 655, 1875, 2077, 2080, 2083, 2098

Y

\yahei 4600, 4602
\youyuan 4598

Z

\zhdig 2273
\zhdigits *26*, 2280
zhmap *7*, *30*, 251
\zhnum 2272
zhnum commands:
\zhnum_counter:n 2264
\zhnumber *26*, 2278
\zhnumsetup 2290, 2295
\zhtoday 2289, 2294
\zihao *25*, 3777, 3786
zihao *8*, 153
\ziju *25*, 2213
\zw 705, 1936, 1976, 2035