

xeCJK 宏包

CTEX.ORG

2026/07/01 v3.10.1*

目录

第 1 节 简介	1	3.7 其他	17
第 2 节 基本用法	2	第 4 节 已知问题和兼容性	17
第 3 节 用户手册	2	4.1 CJK 文字与命令交互时的间距	18
3.1 宏包选项	2	4.2 <code>\char</code> 原语与 <code>interchar</code> 的冲突	19
3.2 字体设置与选择	6	4.3 <code>hyperref</code> 链接与 <code>CJKecglue</code>	19
3.3 CJK 分区字体设置	10	第 5 节 xeCJK 代码实现	19
3.4 设置 CJK 字符范围	11	版本历史	179
3.5 标点符号的处理	12	代码索引	184
3.6 <code>xeCJKfntef</code> 用法说明	14		

第 1 节 简介

xeCJK 是一个 Xe_{La}TeX 宏包, 用于排版中日韩(CJK)文字。主要功能:

1. 分别设置 CJK 和英文字体;
2. 自动忽略 CJK 文字间的空格而保留其他空格, 允许在非标点汉字和英文字母 (a - z, A - Z) 间断行;
3. 提供多种标点处理方式: 全角式、半角式、开明式、行末半角式和 CCT 式;
4. 自动调整中英文间空白。

xeCJK 使用了 Xe_{La}TeX 的一些最新特性, 需要 Xe_{La}TeX 0.9995.0 (2009/06/29) 以后的版本。xeCJK 依赖 L^AT_EX3 项目的宏包套件 `l3kernel` 和 `l3packages`。xeCJK 还需要通过 `fontspec` 宏包来调用系统字体。xeCJK 会自动根据需要载入这些宏包。

xeCJK 的原始作者是孙文昌, 2009 年 5 月起宏包被收入 `ctex-kit` 项目进行维护, 目前主要维护者是刘海洋¹、李清²及黄晨成³。

*`ctex-kit rev. bf040266`.

¹leoliu.pku@gmail.com

²sobenlee@gmail.com

³liamhuang0205@gmail.com

第 2 节 基本用法

与其他 \LaTeX 宏包一样, 引入 `xeCJK` 宏包只要在导言区使用

```
\usepackage{xeCJK}
```

在引入 `xeCJK` 宏包之后, 只要设置 CJK 文字的字体, 就可以在文档中使用中日韩文字了。

可以在各种文档类中使用 `xeCJK` 宏包, 最简单的示例是:

例 1

```
\documentclass{article}
\usepackage{xeCJK}
\setCJKmainfont{SimSun}

\begin{document}
中文 \LaTeX 示例。
\end{document}
```

上述示例设置了中文字体 `SimSun` (宋体)。运行此示例要求系统安装了设置的字体, 源文件用 UTF-8 编码保存, 使用 $X_{\text{Y}}\LaTeX$ 编译。

`xeCJK` 只提供了字体和标点控制等基本 CJK 语言支持。对于中文文档, 可以使用更为高层的 `ctex` 宏包或文档类, 它将自动调用 `xeCJK` 并设置好中文字体, 同时提供了进一步的本地化支持。详细内容参看 `ctex` 宏包套件的说明。

`xeCJK` 提供了大量选项, 可以在宏包调用时作为宏包选项或用 `\xeCJKsetup` 命令进行设置, 详见 3.1 节。除了 `\setCJKmainfont` 命令, `xeCJK` 还提供了许多其他命令设置和选择中文字体, 详见 3.2 节。其他更详细的功能也都将在下面详细说明。在本文档所在的文件夹的 `example` 目录下面也有一些例子可以参考。

第 3 节 用户手册

3.1 宏包选项

`xeCJK` 以 `\langle key \rangle = \langle var \rangle` 的形式提供宏包选项, 你可以在调用宏包的时候直接设置这些选项, 也可以在调用宏包之后使用 `\xeCJKsetup` 来设置这些选项。`xeCJK` 内部调用 `fontspec` 宏包, 可以在调用 `xeCJK` 的时候, 使用它的宏包选项。`xeCJK` 会将 `fontspec` 的选项传递给它。

```
\xeCJKsetup \xeCJKsetup {\langle key_1 \rangle = \langle val_1 \rangle, \langle key_2 \rangle = \langle val_2 \rangle, \dots}
```

其中 `\langle key_1 \rangle`, `\langle key_2 \rangle` 是设置选项, 而 `\langle val_1 \rangle`, `\langle val_2 \rangle` 则是对应选项的设置内容。多个选项可以在一个语句中完成设置。例如

例 2

```
\usepackage[PunctStyle=kaiming]{xeCJK}
```

等价于

例 3

```
\usepackage{xeCJK}
\dots\dots
\xeCJKsetup{PunctStyle=kaiming}
```

带有 \star 或者 \star 标记的选项或命令只能在导言区中使用, 其中 \star 还表示这个选项或命令只影响随后定义的 CJK 字体。其余不带特殊标记的选项或命令, 如果没有特别说明, 则可以在导言区或正文中使用。**粗体**表示 `xeCJK` 的默认设置。

LocalConfig * `LocalConfig = {(true|false|name)}`

是否使用本地配置文件 `xeCJK-⟨name⟩.cfg`。⟨name⟩ 可以是不包含空格的任意使文件名合法的字符串。如果设置为 `true`, 则使用的是 `xeCJK.cfg`; 设置为 `false` 则不载入配置文件。可以把将要在下文介绍到的对 `xeCJK` 的一些设置 (例如设置常用 CJK 字体、修改字符范围和定义新的标点输出格式等) 保存到文件 `xeCJK-⟨name⟩.cfg`。然后把这个文件放在本地的 TDS 目录下的适当位置。使用 \TeX Live 的用户, 可以新建下列目录, 然后再把 `xeCJK-⟨name⟩.cfg` 放在里面:

```
texlive/texmf-local/tex/xelatex/xecjk
```

最后还需要在命令行下执行 `mktexlsr`, 刷新文件名数据库以便 \TeX 系统能够找到它。

请注意, `xeCJK` 宏包中只有上述 `LocalConfig` 选项需要在调用 `xeCJK` 时设置, 而不能通过 `\xeCJKsetup` 来设置。

xeCJKactive `xeCJKactive = (true|false)`

打开/关闭对中文的特殊处理。事实上, 这个选项会打开/关闭 \XeTeX 的整个字符类机制, 依赖这个机制的宏包都会受到影响。

CJKspace `CJKspace = (true|false)`

缺省状态下, `xeCJK` 会忽略 CJK 文字之间的空格, 使用这一选项来保留它们之间的空格。

CJKmath * `CJKmath = (true|false)`

是否支持在数学环境中直接输入 CJK 字符。使用这个选项后, 可以直接在数学环境中输出 CJK 字符。 `url` 宏包将一个 URL 放在一个特殊的数学环境中排版, 所以如果在 `\path` 等命令的路径参数中含有汉字, 则需要启用这个选项, 路径中的汉字才能显示。

CJKglue `CJKglue = {\hskip 0pt plus 0.08\baselineskip}`

设置 CJK 文字之间插入的 `glue`, 上边是 `xeCJK` 的默认值。一般来说, 除非有特殊需要 (例如, 改变文字间距等), 否则不需要设置这个选项, 使用默认值即可。如果要设置这个选项, 为了行末的对齐, 设置的 `glue` 最好有一定的弹性。

CJKecglue `CJKecglue = {⟨glue⟩}`

设置 CJK 文字与西文、CJK 文字与行内数学公式之间的间距, 默认值是一个空格。使用这个选项设置的 ⟨`glue`⟩ 最好也要用一定的弹性。请注意, 这里设置的 ⟨`glue`⟩ 只影响 `xeCJK` 根据需要自动添加的空白, 源文件中直接输入的 CJK 文字与西文之间的空格不受影响 (直接输出)。有时候 `xeCJK` 可能不能正确地调整间距, 需要手动加空格。

xCJKecglue `xCJKecglue = {(true|false|glue)}`

缺省状态下, `xeCJK` 不对源文件中直接输入的 CJK 文字与西文之间的空格进行调整, 如果需要调整, 请使用这个选项。如果使用这个选项, 将使用 `CJKecglue` 替换源文件中直接输入的 CJK 文字与西文之间的空格。

CheckSingle `CheckSingle = (true|false)`

是否避免单个 CJK 文字单独占一个段落的最后一行。需要说明的是, 这个选项只有在段末的最后一个字是 CJK 文字或者标点符号, 并且倒数第二和第三个字都是文字才能正确处理孤字的问题。如果这倒数三个字有作为控制序列的参数情况, 那么一般来说也不能正确处理。

WidowPenalty `WidowPenalty = {⟨penalty|10000⟩}`

使用 `CheckSingle` 选项后, 设置段末三个汉字之间的 `penalty`。初始值为 10000, 即禁止在它们之间折行。

`PlainEquation` `PlainEquation = <true|false>`

如果使用了 `$$...$$` 的形式来输入行间数学公式, 就需要启用本选项, 以便 `CheckSingle` 选项能够正确识别。推荐使用 `\[...]` 的形式来输入行间数学公式。

`NewLineCS` `NewLineCS = { \par \[}`

`NewLineCS+`
`NewLineCS-` 设置造成断行的控制序列, 以便 `CheckSingle` 选项能够正确识别。以上是 `xeCJK` 的初始设置。

`EnvCS` `EnvCS = { \begin \end }`

`EnvCS+`
`EnvCS-` 设置 `LaTeX` 环境开始和结束的控制序列, 以便 `CheckSingle` 选项能够正确识别。以上是 `xeCJK` 的初始设置。

`InlineEnv` `InlineEnv = {<env_1>, <env_2>, <env_3>, ...}`

`InlineEnv+`
`InlineEnv-` 在使用 `CheckSingle` 选项的时候, `xeCJK` 会将 `CJK` 文字后接着的 `LaTeX` 环境的开始 `\begin{...}` 和结束 `\end{...}` 视为断行的地方, 如果有某些特殊的 `LaTeX` 环境没有造成断行, 可以使用这个选项来声明它, 以便 `CheckSingle` 能正确识别。

`AutoFallback` `AutoFallback = <true|false>`

当文档中有个别生僻字时, 可以使用这个选项, 自动使用预先设置好的后备字体来输出这些生僻字。后备字体的设置方法将在 3.2 节中介绍。

`AutoFakeBold` ☆ `AutoFakeBold = <{true|false|数字}>`

全局设定当没有声明对应的粗体时, 是否使用**伪粗体**; 当输入的是数字时, 将使用伪粗体, 并将使用输入的数字作为伪粗体的默认粗细程度。

`AutoFakeSlant` ☆ `AutoFakeSlant = <{true|false|数字}>`

全局设定当没有声明对应的斜体时, 是否使用**伪斜体**; 当输入的是数字时, 将使用伪斜体, 并将使用输入的数字作为伪斜体的默认倾斜程度。倾斜程度的取值范围是 $[-0.999, 0.999]$ 。

`EmboldenFactor` ☆ `EmboldenFactor = <{数字|4}>`

设置伪粗体的默认粗细程度。

`SlantFactor` ☆ `SlantFactor = <{数字|0.167}>`

设置伪斜体的倾斜程度, 范围是 $[-0.999, 0.999]$ 。

`PunctStyle` `PunctStyle = <{quanjiao|banjiao|kaiming|hangmobanjiao|CCT|plain|...}>`

设置标点处理格式。 `xeCJK` 中预先定义好的格式为

`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: 句末点号用全角, 其他半角;

`hangmobanjiao` 行末半角式: 所有标点占一个汉字宽度, 行首行末对齐;

`CCT` CCT 格式: 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)。

可以使用 3.5.2 中介绍的 `\xeCJKDeclarePunctStyle` 定义新的标点格式。

`PunctFamily` `PunctFamily = <{false|family}>`

默认情况下, `CJK` 标点符号的字体与 `CJK` 正文一致, `PunctFamily` 用于单独对标点符号设置字体。 `<family>` 需要使用随后说明的 `\setCJKfamilyfont` 或 `\newCJKfontfamily` 预先定义。 `false` 表示取消本选项的作用, 让标点符号字体与正文一致。

`KaiMingPunct` * `KaiMingPunct = {< . . ? ! >}`

`KaiMingPunct+` * 设置开明(kaiming)标点处理格式时的句末点号,`KaiMingPunct` 后带的 + 与 - 分别表示从已有的开明句末点号中增加或减少标点。
`KaiMingPunct-` *

`LongPunct` * `LongPunct = {< ———·……>}`

`LongPunct+` * 设置长标点,例如破折号”——“与省略号”……”,允许在长标点前后断行,但是禁止在它们之间断行。同时属于 `NoBreakLongPunct` 的长标点则禁止在其前面断行。
`LongPunct-` *

`NoBreakLongPunct` * `NoBreakLongPunct = {< ……>}`

`NoBreakLongPunct+` * 设置禁止在其前面断行的长标点。属于 `NoBreakLongPunct` 的字符必须同时属于 `LongPunct`。
`NoBreakLongPunct-` * 与普通长标点不同, `xeCJK` 禁止在这类标点前面断行,但仍允许在其后面断行,并禁止在相同长标点之间断行。缺省值包含省略号“·”和“…”。

`MiddlePunct` * `MiddlePunct = {< —— · · · ~ = ~ >}`

`MiddlePunct+` * 设置居中显示的标点,例如间隔号“·”。对于在 CJK 文字之间的居中标点, `xeCJK` 会根据不同的标点处理格式,调整居中标点与前后文字之间的空白,保证其确实居中。对于行未出现的居中标点,允许在其后面断行,但禁止在它前面断行。
`MiddlePunct-` *

`PunctWidth` * `PunctWidth = {<length>}`

缺省状态下, `xeCJK` 会根据所选择的标点处理格式自动计算标点所占的宽度,如果对缺省设置不满意,可以通过这一选项来改变它。为了使得标点所占的宽度能够适应字体大小的变化,这里设置的 `length` 的单位最好用 `em` 等相对距离单位,而不建议使用诸如 `pt` 之类的绝对距离单位。这里的设置可用于除了 `plain` 以外的所有标点处理格式。同时,这里的设置对所有的 CJK 标点都生效,如果只要设置部分标点,请使用 3.5.1 节的 `\xeCJKsetwidth`。

`PunctBoundWidth` * `PunctBoundWidth = {<length>}`

与以上选项类似,但设置的是标点符号出现在行首/尾时的宽度。

`AllowBreakBetweenPuncts` `AllowBreakBetweenPuncts = <true|false>`

缺省状态下, `xeCJK` 禁止在相邻 CJK 右标点和 CJK 左标点之间换行,可以使用这一选项改变这一设置。

`RubberPunctSkip` `RubberPunctSkip = <true|false|plus|minus>`

缺省状态下,标点符号前/后的间距有一定的弹性。可以伸长到原始边界宽度,可以收缩到标点另一侧的边界宽度。将本选项设置为 `plus`,将只允许伸长;设置为 `minus` 只允许收缩。设置为 `false` 将禁用这一特性,从而使得前/后的间距为固定值。

`CheckFullRight` `CheckFullRight = <true|false>`

某些控制序列要求不能在它的前面断行。但是在缺省状态下,单个全角右标点的后面总是可以断行的。因此当这些控制序列出现在全角右标点后面时,可能会出现意料之外的断行。此时可以使用这个选项来避免这个情况。

`NoBreakCS` `NoBreakCS = { \footnote \footnotemark \nobreak }`

`NoBreakCS+` * 设置不能在全角右标点后面断行的控制序列。以上是 `xeCJK` 的默认设置。如果这些控制序列在文档中只出现少量几次,也可以不必使用 `CheckFullRight` 选项,而是手工在这些控制序列前面加上 3.7 节介绍的 `\xeCJKnobreak`。
`NoBreakCS-` *

```
experiment/halfright-prebreakpenalty experiment/halfright-prebreakpenalty = <true|false>
```

New: 2026-05-03

这是一个实验性选项。缺省状态下, `xeCJK` 不会禁止在 CJK 文字或全角右标点后面紧跟的半角右标点(如 `. , ; : ! ?)]` 等)之前换行。例如 `. 中, .` 有可能被排在下一行的行首。打开此选项后, `xeCJK` 会在这类位置插入禁则(`penalty 10 000`), 防止半角右标点出现在行首。

此选项的效果类似于 `LuaTeX-ja` 中的 `prebreakpenalty` 机制, 但受限于 `XYTeX interchar` 机制, 粒度为字符级别而非逐字符级别。例如:

```
\xeCJKsetup{experiment/halfright-prebreakpenalty}
```

```
experiment/punct-measure-fix experiment/punct-measure-fix = <true|false>
```

New: 2026-05-17

这是一个实验性选项。缺省状态下, `LATEX` 的 `\para_end:` 在执行 `\tex_par:D` 之前会通过 `\unskip` 移除水平列表末尾的 `glue`。如果段末恰好是全角标点, 其标点补偿 `glue` 也会被移除, 导致 `tabularray` 等使用 `\par` 结束测量段落的宏包得到不正确的宽度。

打开此选项后, `xeCJK` 会在 `para/end` 钩子中插入一个 `\kern`, 补偿被 `\unskip` 移除的标点 `glue` 的自然宽度, 确保测量宽度与排版宽度一致。

注意: 此修复涉及一个权衡。`\kern` 只保留了原 `glue` 的自然宽度, 丢弃了弹性分量 (`stretch/shrink`)。对于段末最后一行, `\parfillskip` 提供的 `0pt plus 1fil` 拉伸量会吸收所有剩余空间, 因此弹性丢失在绝大多数情况下无视觉影响。但在 `CheckSingle` 等强制 `overfull` 的极端场景下, 最后一行的 `overfull` 报告值可能增大。

使用示例:

```
\xeCJKsetup{experiment/punct-measure-fix}
```

```
Verb Verb = <true|false|env|env+>
```

`true` 表示在 `\verb` 命令或 `verbatim` 环境里不自动调整中英文之间的间距。`env` 选项在 `verbatim` 环境里自动计算中西文间距和中文之间的间距, 以便于保持代码的对齐;`env` 选项不调整 `\verb` 里的间距, `env+` 选项还将正文里设置的间距应用到 `\verb` 里。这个选项对使用到 `\verbatim@font` 命令的情形均有效, 更一般的情况可以使用 3.7 节介绍的 `\xeCJKVerbAddon`。`false` 表示不作任何处理。以上选项的值除 `false` 外, 都禁止在汉字之间和汉字与西文之间自动换行。

```
LoadFandol ☆ LoadFandol = <true|false>
```

当没有在导言区设置 CJK 字体时, 是否使用 Fandol 字体。如果启用这个选项, 需要安装 **Fandol** 字体系列。

3.2 字体设置与选择

```
\setCJKmainfont * \setCJKmainfont <{font name}>[<font features>] 或  
\setCJKmainfont [<font features>] <{font name}>
```

设置正文罗马族的 CJK 字体, 影响 `\rmfamily` 和 `\textrm` 的字体。后面两个参数继承自 `fontspec` 宏包, `` 表示字体属性选项, `` 是字体名。字体名可以是字体族名, 也可以是字体的文件名, 查找字体名见 3.2.1 节; 可用的字体属性选项参见 `fontspec` 宏包的文档。需要说明的是 `xeCJK` 修改了 `AutoFakeBold` 和 `AutoFakeSlant` 选项, 以便配合全局伪粗体和伪斜体的设定。

出于兼容性考虑, 字体属性可选项可以放在字体名称前面, 也可以放在后面。如果可选项放在后面, 字体名称与可选项之间不要有空格或者换行。

```
AutoFakeBold AutoFakeBold = <{true|false|数字}>  
AutoFakeSlant AutoFakeSlant = <{true|false|数字}>
```

局部设置当前字体族的伪粗和伪斜属性。如果没有在局部给出这些选项, 将使用全局设定。

`Mapping` Mapping = `{(fullwidth-stop|full-stop|han-trad|han-simp|...)}`

`xeCJK` 提供了以上四个 `TECKit` 映射文件, 可以在设置字体的时候通过 `Mapping` 选项来使用它们。其中 `fullwidth-stop` 用于将正常句号“。”转换成全角实心句号“.”, `full-stop` 的作用相反。`han-trad` 用于将简体中文转换成繁体中文, `han-simp` 的作用相反。需要注意的是, 简繁互换都是简单机械的字字对译, 不能做到完全准确, 使用时要小心。例如简体的“发挥”和“头发”被转换成繁体的“發揮”和“頭髮”, 显然后者应作“頭髮”。也可以根据实际需要, 制作新的映射文件, 请参考 `TECKit` 的文档。

`\setCJKsansfont` * `\setCJKsansfont <{font name}>[]` 或
`\setCJKsansfont [] <{font name}>`

设置正文无衬线族的 CJK 字体, 影响 `\sffamily` 和 `\textsf` 的字体。

`\setCJKmonofont` * `\setCJKmonofont <{font name}>[]` 或
`\setCJKmonofont [] <{font name}>`

设置正文等宽族的 CJK 字体, 影响 `\ttfamily` 和 `\texttt` 的字体。

`\setCJKfamilyfont` * `\setCJKfamilyfont <{family}> <{font name}>[]` 或
`\setCJKfamilyfont <{family}> [] <{font name}>`

声明新的 CJK 字体族 `<family>` 并指定字体。

`\CJKfamily` `\CJKfamily <{family}>`
`\CJKfamily + <{family}>`
`\CJKfamily - <{family}>`

用于在文档中切换 CJK 字体族, `<family>` 必须预先声明。`\CJKfamily` 仅对 CJK 字符类有效, `\CJKfamily+` 对所有字符类均有效, `\CJKfamily-` 对非 CJK 字符类有效。当 `\CJKfamily+` 和 `\CJKfamily-` 的参数为空时, 则使用当前的 CJK 字体族。

`\newCJKfontfamily` * `\newCJKfontfamily [<family>] \<font-switch> <{font name}>[]` 或
`\newCJKfontfamily [<family>] \<font-switch> [] <{font name}>`

声明新的 CJK 字体族 `<family>` 并指定字体, 并定义 `\<font-switch>`, 在文档中可以使用它来切换 CJK 字体族。可以不必指定 `<family>`, 这时候 `<family>` 将等于 `<font-switch>`。

事实上, `\newCJKfontfamily` 是 `\setCJKfamilyfont` 和 `\CJKfamily` 的合并。例如

例 4

```
\newCJKfontfamily[song]\songti{SimSun}
```

等价于

例 5

```
\setCJKfamilyfont{song}{SimSun}
\newcommand*\songti{\CJKfamily{song}}
```

`\CJKfontspec` `\CJKfontspec <{font name}>[]` 或
`\CJKfontspec [] <{font name}>`

在文档中定义新的 CJK 字体族, 并马上使用它。

`\defaultCJKfontfeatures` ☆ `\defaultCJKfontfeatures <{font features}>`

全局设置 CJK 字体族的默认选项。例如, 使用

例 6

```
\defaultCJKfontfeatures{Scale=0.962216}
```

可以将全部 CJK 字体缩小为 0.962216。 `xeCJK` 宏包的初始化设置是

```
\defaultCJKfontfeatures{Script=CJK}
```

```

\addCJKfontfeatures <font features>
\addCJKfontfeatures * <font features>
\addCJKfontfeatures [<block_1, block_2, ...>] <font features>
\addCJKfontfeatures * [<block_1, block_2, ...>] <font features>

```

临时增加当前使用的 CJK 字体的选项。第一条命令, 仅对当前 CJK 主分区字体有效; 第二条对主分区和其他分区的字体都有效; 第三条仅对可选参数中指定的分区有效; 第四条对主分区和可选参数中指定的分区有效。例如, 使用

例 7

```
\addCJKfontfeatures{Scale=1.1}
```

可以将文档中当前使用的 CJK 主分区字体放大为 1.1。

```
\CJKrmdefault
```

保存 `\textrm` 和 `\rmfamily` 所使用的 CJK 字体族, 默认值是 `rm`。

```
\CJKsfdefault
```

保存 `\textsf` 和 `\sffamily` 所使用的 CJK 字体族, 默认值是 `sf`。

```
\CJKttdefault
```

保存 `\texttt` 和 `\ttfamily` 所使用的 CJK 字体族, 默认值是 `tt`。

```
\CJKfamilydefault
```

保存 `\textnormal` 和 `\normalfont` 所使用的 CJK 字体族。类似西文字体的 `\familydefault`。初始值是 `\CJKrmdefault`。如果没有在导言区中修改它, `xeCJK` 会在导言区结束的时候根据西文字体的情况自动更新 `\CJKfamilydefault`。因此, 在导言区里使用

```
\renewcommand\familydefault{\sfdefault}
```

就可以将全文的 CJK 和西文默认字体都改为无衬线字体族。

```

\setCJKmathfont * <font name> [<font features>] 或
\setCJKmathfont [<font features>] <font name>

```

设置数学公式中的 CJK 字体族。如果使用了 `CJKmath` 选项, 但是没有使用 `\setCJKmathfont` 设置数学公式中的 CJK 字体, 那么将使用 `\CJKfamilydefault` 作为数学公式中的 CJK 字体。

```

\setCJKfallbackfamilyfont * <family> <font name> [<font features>] 或
\setCJKfallbackfamilyfont <family> [<font features>] <font name>

```

设置 CJK 字体族 `<family>` 的备用字体。例如, 使用

例 8

```

\setCJKmainfont{SimSun}
\setCJKfallbackfamilyfont{\CJKrmdefault}{SimSun-ExtB}

```

可以将 `SimSun-ExtB` 作为 `SimSun` 的备用字体。

```
FallBack FallBack = {[<font features>]}{<font name>}}
```

xeCJK 在 `` 里增加了 FallBack 这个选项。用来在声明主字体的时候,同时设置备用字体。例如,上面的例子等价于:

例 9

```
\setCJKmainfont[FallBack=SimSun-ExtB]{SimSun}
```

如果 FallBack 的值为空,将设置的是备用字体。例如,

例 10

```
\setCJKmainfont[FallBack,AutoFakeBold,Scale=.97]{SimSun-ExtB}
```

等价于

例 11

```
\setCJKfallbackfamilyfont{\CJKrmddefault}[AutoFakeBold,Scale=.97]{SimSun-ExtB}
```

```
\setCJKfallbackfamilyfont * \setCJKfallbackfamilyfont {<family>}
{
  {[<font features_1>] {<font name_1>}} ,
  {[<font features_2>] {<font name_2>}} ,
  .....
} [<common font features>] 或
\setCJKfallbackfamilyfont {<family>} [<common font features>]
{
  {[<font features_1>] {<font name_1>}} ,
  {[<font features_2>] {<font name_2>}} ,
  .....
}
```

\setCJKfallbackfamilyfont 还可以用于设置多层的备用字体。例如,使用

例 12

```
\setCJKmainfont[AutoFakeBold,AutoFakeSlant]{KaiTi_GB2312}
\setCJKfallbackfamilyfont{\CJKrmddefault}[AutoFakeSlant]
{ [BoldFont=SimHei]{SimSun} ,
  [AutoFakeBold] {SimSun-ExtB} }
```

之后,就设置了 SimSun 是 KaiTi_GB2312 的备用字体,而 SimSun-ExtB 是 SimSun 的备用字体。若当前字体族缺字,并没有备用字体,则尝试使用 \CJKfamilydefault 的备用字体。

3.2.1 XeTeX 的字体名查找

由于在 fontspec 宏包文档中缺少关于如何查看 XeTeX 可用字体名的说明,这里略作说明。

XeTeX 通常使用 fontconfig 库查找和调用字体,因此,可以用 fc-list 命令显示可用的字体。在命令行(Windows 的“命令提示符”,Linux 的 Console)下运行以下命令:

```
fc-list > fontlist.txt
```

可以将系统中所有安装的字体列表存入 fontlist.txt 文件中(可能很长)。

fc-list 命令列出的信息很多,而且在安装字体较多的 Windows 系统上的输出将非常庞大,如其中可能包含:

```
Times New Roman:style=cursiva,kurziva,kursiv,Πλάγια,Italic,
Kursivoitu,Italique,Dólt,Corsivo,Cursief,kursywa,Itálico,Курсив,
Ítalik,Poševno,nghiêng,Etzana
Times New Roman:style=Negreta cursiva,tučné kurziva,fed kursiv,
Fett Kursiv,Έντονα Πλάγια,Bold Italic,Negríta Cursiva,
```

```
Lihavoitu Kursivoi,Gras Italique,Félkövéř dőlt,Grassetto Corsivo,
Vet Cursief,Halvfet Kursiv,Pogrubiona kursywa,Negrĭto Itálico,
Полужирный Курсив,Tučná kurzíva,Fet Kursiv,Kalın İtalik,
Крепко роševno,nghiêng đâġ,Lodi etzana
Times New Roman:style=Negreta,tučné,fed,Fett,Έγρονα,Bold,Negrĭta,
Lihavoitu,Gras,Félkövéř,Grassetto,Vet,Halvfet,Pogrubiona,Negrĭto,
Полужирный,Fet,Kalın,Крепко,đâġ,Lodia
Times New Roman:style=Normal,obyčejné,Standard,Κανονικά,Regular,
Normaali,Normal,Normale,Standaard,Normalny,Обычный,Normalne,Navadno,
thườġ,Arrunta
宋体,SimSun:style=Regular
黑体,SimHei:style=Normal,obyčejné,Standard,Κανονικά,Regular,Normaali,
Normál,Normale,Standaard,Normalny,Обычный,Normalne,Navadno,Arrunta
```

在 `fontspec` 或 `xeCJK` 中使用的字体族名是上面列表中冒号前的部分。例如可以使用

例 13

```
\setmainfont{Times New Roman}
\setCJKmainfont{SimSun} % 或者 \setCJKmainfont{宋体}
```

来设置字体。

为了方便起见, `fc-list` 命令也可以加上各种选项控制输出格式, 例如如果只要列出所有的中文字体的字体族名, 可以用命令:

```
fc-list -f "%{family}\n" :lang=zh > zhfont.txt
```

这样就把字体列表保存在文件 `zhfont.txt` 中⁴。这样列出的字体列表就比较简明易用, 如 Windows 下预装的中文字体:

```
Arial Unicode MS
FangSong, 仿宋
KaiTi, 楷体
Microsoft YaHei, 微软雅黑
MingLiU, 細明體
NSimSun, 新宋体
PMingLiU, 新細明體
SimHei, 黑体
SimSun, 宋体
```

要列出日文和韩文的字体, 可以把 `:lang=zh` 选项中的 `zh` 改成 `ja` 或 `ko`。

`fontspec` 和 `xeCJK` 也可以使用字体的文件名访问字体。例如 Windows 下的宋体也可以使用命令:

```
\setCJKmainfont{simsun.ttc}
```

来设置。设置字体文件名的相关选项和语法在 `fontspec` 宏包手册中叙述甚详, 这里不再赘述。有个别字体名不规范的中文字体, `xeCJK` 宏包可能无法正确地通过字体名访问, 那么也可以使用这种方式设置。

3.3 CJK 分区字体设置

众所周知, CJK 文字数量极其庞大, 单一的字体不可能涵盖所有的 CJK 文字。 `xeCJK` 可以在同一 CJK 字体族下, 自动使用不同的字体输出 CJK 字符范围内不同区块里的文字。首先要声明 CJK 子分区。

⁴由于汉字编码原因, Windows 下总需要把字体列表输出的文件中防止乱码。

```
\xeCJKDeclareSubCJKBlock * \xeCJKDeclareSubCJKBlock {<block>} {<block range>}
\xeCJKDeclareSubCJKBlock * {<block>} {<block range>}
```

其中 `<block range>` 是逗号列表, 可以是 CJK 字符的 Unicode 范围, 也可以是单个字符的 Unicode。例如

例 14

```
{ `中 -> `文 , "3400 -> "4DBF , "5000 -> "7000 , `汉 , `字 , "3500 }
```

的形式。需要注意的是, 这里设置的 `<block range>` 除非确实需要 (例如某些特殊字体使用了 Unicode 中的私人使用区的情况), 否则不要超出源代码中预设的 **CJK 文字范围**。使用

例 15

```
\xeCJKDeclareSubCJKBlock{SPUA}{ "E400 -> "E4DA , "E500 -> "E5E8 , "E600 -> "E6CE }
\xeCJKDeclareSubCJKBlock{Ext-B}{ "20000 -> "2A6DF }
```

就声明了 SPUA 和 Ext-B 这两个子分区。同时在 3.2 节介绍的 CJK 字体设置命令的 `` 里新建了 SPUA 和 Ext-B 这两个选项。新建的这两个选项的使用方法跟 3.2 介绍的 FallBack 类似。可以通过它们来设置字体。

例如, 可以使用

例 16

```
\setCJKmainfont [SPUA=SunmanPUA,Ext-B=SimSun-ExtB] {SimSun}
```

设置文档的主字体是 SimSun, SPUA 分区的字体是 SunmanPUA, 而 Ext-B 分区的字体是 SimSun-ExtB。

`\xeCJKDeclareSubCJKBlock` 应该在声明所有的 CJK 字体族之前使用。如果有某个 CJK 字体族没有设置 `<block>` 选项, 将使用 `\CJKfamilydefault` 的 `<block>` 选项作为该 CJK 字体族的 `<block>` 选项。如果希望在使用某 CJK 字体族时, 不在 CJK 主分区与 `<block>` 之间切换字体, 可以使用 `<block>=*` 选项。带星号的命令除了设置 CJK 子分区以外, 还重置标点符号所属的字符类。

```
\xeCJKCancelSubCJKBlock \xeCJKCancelSubCJKBlock {<block_1, block_2, ...>}
\xeCJKCancelSubCJKBlock {<block_1, block_2, ...>}
```

在文档中取消对 CJK 分区的声明。带星号的命令还重置标点符号所属的字符类。

```
\xeCJKRestoreSubCJKBlock \xeCJKRestoreSubCJKBlock {<block_1, block_2, ...>}
\xeCJKRestoreSubCJKBlock * {<block_1, block_2, ...>}
```

在文档中恢复对 CJK 分区的声明。带星号的命令还重置标点符号所属的字符类。

3.4 设置 CJK 字符范围

```
\xeCJKDeclareCharClass * \xeCJKDeclareCharClass {<class>} {<class range>}
\xeCJKDeclareCharClass * {<class>} {<class range>}
```

`<class range>` 的格式和 3.3 节的 `<block range>` 相同。`<class>` 的有效值见源代码(第 5.4 节)。`xeCJK` 已经支持 Unicode 中所有 CJK 文字和标点。一般来说, 不要轻易改变字符类别。带星号的命令除了设置字符类别以外, 为了确保标点处理的正确性, 还重置标点符号所属的字符类。

```
\xeCJKResetCharClass * 用于恢复 xeCJK 对各个字符类别的初始化设置。
```

```
\xeCJKResetPunctClass * 用于重置标点符号所属的字符类。
```

```
\normalspacedchars \normalspacedchars {<char list>}
```

在 `<char list>` 中出现的字符两端不自动添加空格, 初始设置是 /、\、和 - (U+002D)。

3.5 标点符号的处理

`xeCJK` 对标点符号的输出宽度的调整是通过调整其左边或右边的空白宽度来实现的。按照目前的处理方式,对于位于左边的标点符号(如左引号),`xeCJK` 只能调整它左边的空白;对于位于右边的标点符号(如右引号),`xeCJK` 只能调整它右边的空白;对于居中的标点符号,则调整其左右空白,以保证其居中。对于标点符号的相关设置,只能在导言区中进行。

3.5.1 设置特定标点符号的宽度和间距

这里的设置可用于除 `plain` 以外的所有标点处理格式。

```
\xeCJKsetwidth * \xeCJKsetwidth <{标点列表}> <{length}>
\xeCJKsetwidth * <{标点列表}> <{length}>
```

`<标点列表>` 可以是单个标点,也可以是多个标点。例如,

例 17

```
\xeCJKsetwidth{。 ? }{0.7em}
```

将设置句号和问号所占的宽度为 `0.7em`。带星号的命令,设置标点符号出现在行首/尾时的宽度。

```
\xeCJKsetkern * \xeCJKsetkern <{前标点}> <{后标点}> <{length}>
```

`xeCJK` 会根据选定的标点处理格式自动调整相邻的前后两个 `CJK` 标点符号的空白宽度。如果需要个别情况进行特殊调整,可以使用这个命令。例如,

例 18

```
\xeCJKsetkern{: }{ “}{0.3em}
```

将设置冒号与左双引号之间的空白宽度为 `0.3em`。

3.5.2 定义标点符号处理格式

```
\xeCJKDeclarePunctStyle * \xeCJKDeclarePunctStyle <{style}> <{options}>
```

定义新的标点符号处理格式,已经存在的同名格式将被覆盖。可以设置的选项将在下面介绍。

```
\xeJKEditPunctStyle * \xeJKEditPunctStyle <{style}> <{options}>
```

修改已有的标点符号处理格式。

下面是可以设置的标点符号格式选项。其中左边一栏是选项名称,中间是选项的输入值类型,右边则是相关说明。某些选项之间是互斥的,具有优先级关系。要使下一级的选项有效,则需要先禁用上一级的设置:对于 `<boolean>` 类型的选项,将其设置为 `false`,对于 `<length>` 类型的选项,将其设置为 `\maxdimen`,而对于 `<real>` 类型的选项,将其设置为 `nan`。

`enabled-global-setting <boolean>` 是否使用 `\xeCJKsetup` 的 `PunctWidth`、`PunctBoundWidth` 选项和 `\xeCJKsetwidth`、`\xeCJKsetkern` 的设置。默认值是 `true`。

`fixed-punct-width <length>` 设置单个标点符号的宽度。默认值是 `\maxdimen`。

`fixed-punct-ratio <real>` 设置单个标点符号的输出宽度与实际宽度的比例。默认值是 `1.0`。

`mixed-punct-width <length>` 设置句末标点符号的宽度。其中句末标点符号通过 `\xeCJKsetup` 的 `KaiMingPunct` 来设置。默认值是与 `fixed-punct-width` 选项的值相同。

`mixed-punct-ratio <real>` 设置句末标点符号的宽度比例。默认值是与 `fixed-punct-ratio` 选项的值相同。

`middle-punct-width <length>` 设置居中标点符号的宽度。其中居中标点符号通过 `\xeCJKsetup` 的 `MiddlePunct` 来设置。默认值是与 `fixed-punct-width` 选项的值相同。

`middle-punct-ratio` *(real)* 设置居中标点符号的宽度比例。默认值是与 `fixed-punct-ratio` 选项的值相同。

以上几个选项设置的是标点的固定宽度或比例, `xeCJK` 会根据设定的选项计算标点符号左/右的空白宽度。下面的选项设置的是标点符号左/右的空白宽度或比例, 因此不同标点符号的宽度可能会不同。为了使下面的选项生效, 需要先禁用上面的相应选项。优先级自上而下。

`fixed-margin-width` *(length)* 设置标点的左/右空白宽度。默认值是 `\maxdimen`。

`fixed-margin-ratio` *(real)* 设置标点的左/右空白宽度与字体中该标点的相应实际边界宽度的比例。默认值是 1.0。

`mixed-margin-width` *(length)* 设置句末标点的左/右空白宽度。默认值是与 `fixed-margin-width` 的值相同。

`mixed-margin-ratio` *(real)* 设置句末标点的左/右空白宽度的比例。默认值是与 `fixed-margin-ratio` 的值相同。

`middle-margin-width` *(length)* 设置居中标点的两边空白宽度。默认值是与 `fixed-margin-width` 的值相同。

`middle-margin-ratio` *(real)* 设置居中标点的两边空白宽度之和与两边实际两边边界宽度之和的比例。默认值是与 `fixed-margin-ratio` 的值相同。

下面选项设置标点符号出现在行首或者行尾时的宽度或比例。

`bound-punct-width` *(length)* 设置标点符号出现在行首/尾时的宽度。默认值是 `\maxdimen`。

`bound-punct-ratio` *(real)* 设置标点符号出现在行首/尾时的输出宽度与实际宽度的比例。默认值是 `nan`。

`bound-margin-width` *(length)* 设置标点符号出现在行首/尾时的左/右空白宽度。默认值是 `\maxdimen`。

`bound-margin-ratio` *(real)* 设置标点符号出现在行首/尾时的左/右空白宽度与相应实际边界宽度的比例。默认值是 0。

`enabled-hanging` *(boolean)* 当以上选项的计算结果得到的宽度小于标点符号的实际边界宽度时, 是否允许标点符号悬挂出页面边界。默认值是 `false`。

`add-min-bound-to-margin` *(boolean)* 是否在以上计算结果的基础上再加上标点的左右实际边界宽度中的最小值。这个选项对居中的标点无效。默认值是 `false`。

`optimize-margin` *(boolean)* 使用以上设置空白宽度或比例的选项时, 最终输出的标点符号左/右的空白宽度可能大于原来的实际边界宽度。若此时本选项被设置为 `true`, 则使用原来的实际边界宽度。而使用 `fixed-punct-width` 选项计算得出的左/右宽度可能小于该标点的另一侧宽度, 若此时本选项被启用, 则使用该标点的另一侧宽度。默认值为 `false`。

`margin-minimum` *(length)* 指定标点符号左/右的最小空白宽度。当经过以上选项设置的空白宽度小于这个选项的值时, 则使用这个选项的值。默认值是 `0pt`。

下面的选项处理的是前后相邻的两个标点符号之间的空白宽度。这些选项是互斥的, 优先级自上而下。

`enabled-kerning` *(boolean)* 是否调整前后相邻的两个标点之间的空白宽度。如果设置为 `false`, 则每个标点都按原来的输出宽度输出。默认值是 `true`。

`min-bound-to-kerning` *(boolean)* 是否使用当前字体中前面标点实际左右边界的最小值与后面标点实际左右边界的最小值中的最大值作为两个标点之间的空白宽度。默认值是 `false`。

`kerning-total-width` *(length)* 设置两个标点的总共宽度。此时 `xeCJK` 会自动计算两个标点之间的空白宽度。默认值是 `\maxdimen`。

`kerning-total-ratio` *(real)* 设置两个标点的总共输出宽度与实际宽度的比例。默认值是 0.75。

`same-align-margin` *(length)* 前后两个标点位于同侧时, 它们之间的空白宽度。默认值是 `\maxdimen`。

`same-align-ratio` *(real)* 前后两个标点位于同侧时, 它们之间的空白宽度与实际输出宽度的比例。默认值是 `nan`。

`different-align-margin` *(length)* 前后两个标点位于异侧时, 它们之间的空白宽度。默认值是 `\maxdimen`。

`different-align-ratio` *(real)* 前后两个标点位于异侧时, 它们之间的空白宽度与实际输出宽度的比例。默认值是 `nan`。

`kerning-margin-width` *(length)* 设置前后两个标点之间的空白宽度。默认值是 `\maxdimen`。

`kertering-margin-ratio` *(real)* 设置前后两个标点之间的空白宽度与实际输出空白的比例。默认值是 1.0。

`optimize-kertering` *(boolean)* 使用以上选项计算出两个标点之间的空白宽度可能小于通过 `min-bound-to-kertering` 选项得出的结果。当出现这一情况时, 若此选项被设置为 `true`, 则使用该选项的空白宽度。默认值为 `false`。

`kertering-margin-minimum` *(length)* 指定两个标点之间的最小空白宽度。当经过以上选项设置的空白宽度小于这个选项的值时, 则使用这个选项的值。默认值是 `0pt`。

事实上, `xeCJK` 的默认设置就相当于中文全角 (`quanjiao`) 格式。可以使用上面说明的选项定义新的标点处理格式。例如, 使用

例 19

```
\xeCJKDeclarePunctStyle { mine }
{
  fixed-punct-ratio      = nan ,
  fixed-margin-width    = 0 pt ,
  mixed-margin-width    = \maxdimen ,
  mixed-margin-ratio    = 0.5 ,
  middle-margin-width   = \maxdimen ,
  middle-margin-ratio   = 0.5 ,
  add-min-bound-to-margin = true ,
  bound-punct-width     = 0 em ,
  enabled-hanging       = true ,
  min-bound-to-kertering = true ,
  kertering-margin-minimum = 0.1 em
}
```

就定义了一个名为 `mine` 的标点处理格式。可以在导言区通过

```
\xeCJKsetup{PunctStyle=mine}
```

在文档中使用这个格式。它的意义是: 使用标点符号的实际左右边界中的最小值作为其左/右空白的宽度, 对于句末标点和居中标点, 再加上实际边界空白的一半; 当标点出现在行首或行尾时宽度为零, 允许悬挂出页面边界; 使用相邻两个标点的实际边界中的较小值作为它们之间的空白宽度, 并且最小的空白宽度是 `0.1em`。再例如, 使用

例 20

```
\xeCJKEditPunctStyle { hangmoban_jiao } { enabled-global-setting = false }
```

将使得 `\xeCJKsetkern` 等的设置对 `hangmoban_jiao` 这一格式无效。

3.6 xeCJKfntef 用法说明

`xeCJK` 包含有一个子宏包 `xeCJKfntef`, 可以用它来实现汉字加点和可断行的下划线等。它是 `CJKfntef` 宏包在 `X3LaTeX` 下的替换版本, 基本用法完全一致。

`xeCJKfntef` 基于 `ulem` 宏包, 除了兼容 `ulem` 定义的一些命令外, 还进行了一些扩充:

```
\CJKKunderline [*] [-] [(选项)] {(内容)}
```

```
\CJKKunderdblline
```

```
\CJKKunderwave
```

```
\CJKKsout
```

```
\CJKKxout
```

虚室生白, 吉祥止止

虚室生白, 吉祥止止

虚室生白, 吉祥止止

虚室生白, 吉祥止止

虚室生白, 吉祥止止

```
1 \CJKKunderline{虚室生白, 吉祥止止}\\
2 \CJKKunderdblline{虚室生白, 吉祥止止}\\
3 \CJKKunderwave{虚室生白, 吉祥止止}\\
4 \CJKKsout{虚室生白, 吉祥止止}\\
5 \CJKKxout{虚室生白, 吉祥止止}
```

```
1 \CJKKunderline-{\南朝}\CJKKunderline-{\梁}\CJKKunderline-{\劉勰}%
```

```
2 \CJKKunderwave-{\文心雕龍}\CJKKunderwave-{\養氣}\\
```

```
3 \CJKKunderline*[thickness=1pt, hidden=true]{瞻彼閼者, 虚室生白, 吉祥止止}
```

南朝梁劉勰文心雕龍養氣

`\CJKunderdot` `\CJKunderdot` [(选项)] {(内容)}

在汉字下加点, 可以和上述下划线命令嵌套使用。例如

虚室生白, 吉祥止止	1	<code>\CJKunderline{虚室生白, \CJKunderdot{吉祥}止止}\</code>
虚室生白, 吉祥止止	2	<code>\CJKunderdot{虚室生白, \CJKunderline{吉祥}止止}</code>

对上述六种对象, `xeCJKfntef` 提供了一些选项, 设置点或线的位置和颜色。可以用 `\xeCJKsetup` 预先统一设置它们, 也可以在使用时特别设置。

`skip` `\xeCJKsetup { underline/skip = <true|false> }`
`\xeCJKsetup { underline = { skip = <true|false> , ... } }`

默认情况下, 下划线会自动跳过中文标点符号, 可以设置本选项为 `false`, 禁用这一功能。相应下划线命令后加上 `*` 号, 具有相同的效果。

`subtract` 设置本选项为 `true`, 使得下划线的首尾减少一定距离, 避免前后的下划线连在一起, 适用于古籍标点整理中的专名号和书名号。在相应下划线命令后加上 `-` 号, 具有相同的效果。

`hidden` 设置本选项为 `true`, 将隐藏文本内容, 只画下划线。

`format` `\xeCJKsetup { underline/format = \color{red} }`
`\xeCJKsetup { underwave = { format = \color{red}, ... } }`

设置线或点的格式, 比如颜色。

`textformat` 设置下划线或点的正文的格式。例如:

1	<code>\CJKunderline[textformat=\color{blue}]{虚室生白, 吉祥止止}\</code>
2	<code>\CJKunderdot[textformat=\bfseries, format=\color{red}]{虚室生白, 吉祥止止}</code>

虚室生白, 吉祥止止

虚室生白, 吉祥止止

`symbol` 设置 `\CJKunderwave` 或 `\CJKunderdot` 的符号。

例如, 波浪线 `\CJKunderwave` 的符号不会随字号而变化, 在小字号下不好看。我们可以将它改为随字号而变化大小:

瞻彼阙者, 虚室生白, 吉祥止止	1	<code>% \usepackage{fix-cm}</code>
	2	<code>\xeCJKsetup{%</code>
	3	<code>underwave/symbol=</code>
	4	<code>\fontsize{0.5em}{0pt}%</code>
	5	<code>\fontencoding{U}\fontfamily{lasy}\selectfont</code>
	6	<code>\char 58\relax}</code>
	7	<code>\footnotesize</code>
	8	<code>\CJKunderwave{瞻彼阙者, 虚室生白, 吉祥止止}</code>

`thickness` 设置 `\CJKunderline`、`\CJKunderdblline` 和 `\CJKsout` 的线的厚度。初始值是 `\ULthickness`。

`depth` 设置线或点的深度(基线到线或点的顶部的距离)。初始值都是 `0.2em`。

`boxdepth` `\CJKunderdot` 可能会影响到行距, 可以设置本选项进行调整。如果不希望 `\CJKunderdot` 影响行距, 可以将本选项设置为 `0pt`。

`sep` 设置 `\CJKunderdot` 与 `\CJKunderline`、`\CJKunderdblline` 或 `\CJKunderwave` 嵌套使用时, 点与线或者线与点的距离。

`\CJKKunderdblline` 的两条线之间的距离。初始值是 1.1pt。

`\CJKsout` 的高度(线的中心到基线的距离)。初始值是 0.35em。

例如,我们可以设置 `\CJKsout` 的厚度和颜色,让它具有类似高亮的效果:

```
1 \CJKsout*[thickness=2.5ex, format=\color{yellow}]{瞻彼阙者, 虚室生白, 吉祥止止}
```

瞻彼阙者, 虚室生白, 吉祥止止

`\xeCJKfntef` 还提供给自定义下划线和符号的 `\CJKKunderanyline` 和 `\CJKKunderanysymbol`。

`\CJKKunderanyline` `\CJKKunderanyline` [*] [-] [(选项)] {<深度>} {<下划内容>} {<文本内容>}

`\xeCJKfntef` 先将 <下划内容> 放进一个盒子(`\xeCJKfntefbox`)里,然后向下移动 <深度> 给定的距离,再用于填充。可用的 <选项> 是 `textformat`、`skip`、`hidden`、`subtract`、`sep` 和 `boxdepth`。选项 `sep` 和 `boxdepth` 的初始值是空,表示禁用该选项的功能。可以在 `\xeCJKsetup` 中通过对对象 `ulem` 来设置。

例如,高亮效果也可以如下实现:

```
1 \CJKKunderanyline*{0.5ex}{\color{yellow}\rule{2pt}{2.5ex}}{虚室生白, 吉祥止止}
```

虚室生白, 吉祥止止

`\CJKKunderanysymbol` `\CJKKunderanysymbol` [(选项)] {<深度>} {<符号>} {<文本内容>}

`\xeCJKfntef` 将 <符号> 放进一个盒子(`\xeCJKfntefbox`)里。<深度> 参数用于设置盒子顶部的深度(基线到盒子顶部的距离)。可用的 <选项> 是 `textformat`、`sep` 和 `boxdepth`, 意义与 `\CJKKunderdot` 的相同。

例如,给汉字加三角形,可以如下设置:

```
1 \CJKKunderanysymbol[sep=0.1em]{0.2em}{\tiny$\triangle$}
2 {瞻彼阙者, 虚室生白, \CJKKunderline{吉祥止止}}
```

瞻彼阙者, 虚室生白, 吉祥止止
△ △ △ △ △ △ △ △ △ △ △ △ △ △

`\xeCJKfntefon` `\xeCJKfntefon` [*] [-] [(选项)]

功能与用法 `ulem` 宏包的 `\ULon` 相同, 扩展了可选参数符号 `*` 和 `-`, 可用的 <选项> 是 `textformat`、`skip`、`hidden` 和 `subtract`。这四个选项对 `ulem` 宏包定义的 `\uline` 等命令也有效,需要在 `\xeCJKsetup` 中通过对对象 `ulem` 来设置。例如

```
1 \xeCJKsetup{ulem={textformat=\bfseries\color{red}, skip=true}}
2 \uline{虚室生白, 吉祥止止}
```

虚室生白, 吉祥止止

`\xeCJKfntef` 的下划线和删除线命令内部使用 `ulem` 宏包的 `\hbox` 机制排版文字。汉字 `\CJKsout{文}` 中 `\CJKsout` 前后的源码空格会被 `\xeCJK` 自动处理为 CJK 文字间距 (`\CJKglue`), 与汉字 文字 汉字 的效果一致。关于 `\xeCJKfntef` 命令与颜色命令的嵌套组合等更多细节,请参见“已知问题和兼容性”一节中关于“CJK 文字与命令交互时的间距”的说明。

此外, `\xeCJKfntef` 还提供了指定宽度, 让汉字分散对齐的环境 `\CJKfilltwosides` 和 `\CJKfilltwosides*`。

```
CJKfilltwosides \begin{CJKfilltwosides} [<位置>] {<宽度>}
                文本内容\\
                文本内容
                \end{CJKfilltwosides}
```

环境中的内容被放入垂直盒子中, 可选参数 `<位置>` 指定盒子的基线位置。可以使用 `t` (顶部)、`c` (居中) 和 `b` (底部), 默认是 `c`。 `<宽度>` 参数指定盒子的宽度。 `CJKfilltwosides*` 环境与 `CJKfilltwosides` 的区别是, 当 `<宽度>` 不大于零或者不大于盒子的自然宽度时, 就取盒子的自然宽度。例如

```
瞻    彼    阒    者,
虚 室 生 白, 吉 祥 止 止
```

```
1 \begin{CJKfilltwosides}{.8\linewidth}
2   瞻彼阒者, \\
3   虚室生白, 吉祥止止
4 \end{CJKfilltwosides}
```

```
瞻 彼 阒 者,
虚室生白, 吉祥止止
```

```
1 \begin{CJKfilltwosides*}{0pt}
2   瞻彼阒者, \\
3   虚室生白, 吉祥止止
4 \end{CJKfilltwosides*}
```

3.7 其他

```
\xeCJKVerbAddon
\xeCJKOffVerbAddon
```

调整文字间距以便于让 CJK 字符占的宽度等于西文等宽字体中两个空格的宽度。如果这两个空格的宽度小于当前 CJK 正常文字的宽度, 将对 CJK 字体进行适当地缩小。这有利于等宽字体的代码对齐等情形。需要注意的是, `\xeCJKVerbAddon` 对 `xeCJK` 的内部进行了比较大的修改, 使用它之后, 将禁止在 CJK 字符类之间自动换行, 这与西文在抄录环境中的情况是一致的。所以不应该单独使用, 应该放在分组里限制其作用域, 否则是无效的。当然它可以和其他关于代码抄录的宏包配合使用。例如, 可以使用于 `fancyvrb` 宏包的 `formatcom` 选项。此时设置的西文字体应该确实是等宽的以保证对齐。若西文等宽字体发生变动(包括字体大小), 则需要在其后面使用 `\xeCJKVerbAddon`, 重新计算间距的宽度。`\xeCJKOffVerbAddon` 用于在使用 `\xeCJKVerbAddon` 的环境中局部取消它的作用。由于 `listings` 宏包有自己的代码对齐机制, 所以 `\xeCJKVerbAddon` 在由 `listings` 定义的代码环境中无效。

```
\xeCJKKnobreak ……汉字。 \xeCJKKnobreak\footnote{脚注}
```

`\xeCJKKnobreak` 用在全角标点符号后面, 目的是确保不能在此处断行。如果已经启用了前面介绍的 `CheckFullRight` 选项, 则不需要再用此命令。

```
\xeCJKShipoutHook
```

`xeCJK` 在正文中的一些特殊设置 (汉字下加点、在 `verbatim` 或 `lstlisting` 环境中分页) 可能会影响到 \TeX 的输出例程序 (output routine) 中的内容 (比如页眉和页脚)。`\xeCJKShipoutHook` 用于恢复正文中的普通设置。`xeCJK` 已经处理了页眉和页脚的情况, 其他的就需要根据情况自行调用。比如若使用 `eso-pic` 或者 `atbegshi` 实现文字水印, 并且正文中使用了以上所列的特殊形式, 就需要在命令 `\AtBeginShipout` 的参数的前面使用 `\xeCJKShipoutHook`。

第 4 节 已知问题和兼容性

根据 `unicode-data` 宏包, \XeTeX 将所有 CJK 表意文字的 `\catcode` 设置为 11。因此汉字可以直接用作控制序列的名字, 但是当汉字出现在控制序列后面的时候, 要用空格分隔开, 否则就会出现“! Undefined control sequence.”的错误。

`xeCJK` 使用并重新定义了 CJK 宏包的部分宏命令, 如 `\CJKfamily`、`\CJKsymbol` 和 `\CJKglue` 等。需要指出, `xeCJK` 不需要 CJK 的支持, 并且 `xeCJK` 自动禁止在它之后载入 CJK 宏包。可以在 `xeCJK` 之后载入 `CJKnumb` 宏包, 实现数字的中文化, 也可以用功能更完善的 `zhnumber` 宏包。

xeCJK 进行了一些处理,使得在使用 XeTeX 时 listings 宏包可以支持 Unicode,因此在 listings 定义的代码环境中可以直接使用中文,不再需要通过 escapechar。

新版本(3.x)的 xecjk 完全使用 LaTeX3 的语法来编写。LaTeX3 放弃了 \outer 宏的概念,因此相关工具在遇到 \outer 宏时可能会存在问题。按照目前 xecjk 的实现方式,在 CJK 文字后面遇到 \outer 宏时会出现类似

```
! Forbidden control sequence found while scanning use of \use_i:nn
```

的错误。目前已知的有 cprotect 宏包提供的 \cprotect。它的定义是

```
\outer\long\def\cprotect{\icprotect}
```

因此,这时可以暂时用 \icprotect 代替 \cprotect。事实上,当 cprotect 被引入时,xeCJK 将使用

```
\let\cprotect\icprotect
```

来取消 \cprotect 的外部宏限制。但由于 \cprotect 的特殊性,应该只在外部使用它,即不要让它出现在任何宏的参数中。其他 \outer 宏的情况,可以在它前面加上 \relax 来回避上面的错误。

xeCJK 依赖 XeTeX 的 \XeTeXinterchartoks 机制,与使用相同机制的宏包(例如 polyglossia 和 xesearch)可能会存在大小不一的冲突。xeCJK 虽然为此作了一些处理,但与它们共同使用时应该小心。

4.1 CJK 文字与命令交互时的间距

xeCJK 利用 XeTeX 的 interchar 机制,在 CJK 文字之间自动插入 CJKglue,在 CJK 文字与西文之间自动插入 CJKecglue,并自动忽略 CJK 文字间的源码空格。当 CJK 文字被 LaTeX 命令包裹时,命令引入的分组、盒子或特殊节点可能干扰这一机制。xeCJK 的预期行为是:

```
前 中 后          % 直接排版
前 \foo{中} 后    % 用命令包裹
```

两种写法应产生相同的 CJK 间距。

目前 xecjk 已确保以下命令在此场景下的间距行为正确:

- 字体切换命令: \textbf、\textit、\emph 等。这些命令仅切换字体,对 interchar 机制透明。
- 颜色命令: \textcolor、\colorbox 等(基于 LaTeX_{2_ε} 颜色机制的命令)。
- xecjkfntef 命令: \CJKunderdot、\CJKunderline、\CJKsout、\CJKunderwave、\CJKunderdblline、\CJKxout。
- 盒子命令: \mbox。
- hyperref 的链接注释: \ref、\hyperref 等。
- 以上命令的嵌套组合,例如:

```
前 \CJKunderline{\textcolor{red}{中}} 后
前 \textcolor{red}{\CJKunderline{中}} 后
```

需要注意的是,xeCJK 的间距兼容是针对上述命令的内部行为逐一实现的,并非通用的自动检测机制。未列出的第三方命令如果在 CJK 文字间引入特殊的分组或节点结构,仍可能产生多余或缺失的间距。已知的限制包括:使用 LaTeX₃ 颜色后端(l3color)的命令尚未覆盖。如果遇到间距异常,欢迎到 [GitHub Issues](#) 提交兼容请求。

4.2 \char 原语与 interchar 的冲突

X_YTeX 的 interchar 机制工作在 token 层面, 无法区分 \char 原语的输出和直接字符输入。因此当第三方宏包在内部使用 \char 从非 CJK 字体取字形时, 如果该字符编码恰好属于 xeCJK 的 interchar 字符分类, 就会被拦截并切换到 CJK 字体, 产生错误的字形。例如 mpro2 的 \overbrace 通过 \char183 取花括号字形, 而 U+00B7(MIDDLE DOT)在 xeCJK 中属于 FullRight 类, 导致花括号变成中文间隔号。

为此, xeCJK 提供 \xeCJKchar 命令, 用法与 \char 完全相同, 但在输出字符前临时关闭 XeTeXinterchartokenstate。用户可以使用 \xeCJKchar"00B7 代替 \char"00B7 来避免 interchar 拦截。

xeCJK 已为以下宏包的命令提供了自动兼容补丁:

- mpro2:\overbrace、\underbrace
- pifont:\Pifont(U 编码字体切换)

如果遇到其他宏包的类似问题, 可以在导言区手动包装有冲突的命令:

```
% 方法一：直接用 \xeCJKchar 替换 \char
\hbox{\myfont\xeCJKchar183}

% 方法二：包装第三方命令，关闭 interchar
\let\origProblemCmd\problemCmd
\renewcommand\problemCmd[1]{%
  \makexeCJKinactive % 关闭 interchar
  \origProblemCmd{#1}}
```

\makexeCJKinactive 是赋值操作, 受 TeX 分组约束——在 {...}、\$...\$ 或 \begingroup... \endgroup 内使用时, 结束后自动恢复。

4.3 hyperref 链接与 CJKecglue

hyperref 的链接注释命令(如 \ref、\hyperref 等)在内部会产生 pdf:bann / pdf:eann 特殊节点。xeCJK 对此做了兼容处理, 确保链接注释不影响正常的 CJKecglue 插入。例如, 以下写法的排版效果与不加 \ref 时等价:

```
图\ref{fig}图 % 等价于"图1图"
```

无论是否加载 hyperref, \ref 前后都会自动插入适当的 CJKecglue。

第 5 节 xeCJK 代码实现

```
1 <^*package>
2 <@=@xeCJK>
```

5.1 运行环境检查

xeCJK 必须使用 X_YTeX 引擎的支持。

```
3 \msg_new:nnn { xeCJK } { Require-XeTeX }
4 {
5   The~xeCJK~package~requires~XeTeX~to~function.\\
6   You~must~change~your~typesetting~engine~to~"xelatex" \\
7   instead~of~"plain~"~"latex"~or~"pdflatex"~or~"lualatex".\\
8   Loading~xeCJK~will~abort!
9 }
10 \sys_if_engine_xetex:F { \msg_critical:nn { xeCJK } { Require-XeTeX } }
```

应该使用较新版本的 `expl3` 宏包。

```

11 \msg_new:nnn { xeCJK } { l3-too-old }
12 {
13   Support~package~`#1'~too~old. \\\
14   Please~update~an~up~to~date~version~of~the~bundles\\\
15   `l3kernel'~and~`l3packages'\\\
16   using~your~TeX~package~manager~or~from~CTAN.\
17   \str_if_eq:nnT {#1} { expl3 } { Loading~xeCJK~will~abort! }
18 }
19 \@ifpackagelater { expl3 } { 2025/10/09 } { }
20 { \msg_critical:nnn { xeCJK } { l3-too-old } { expl3 } }

```

`\ctex_disable_package:n` 由 `ctexhook` 提供。

```
21 \RequirePackage { ctexhook }
```

判断宏包是否被引入, 可用于文档正文中。

```

\xeCJK_if_package_loaded_p:n
\xeCJK_if_package_loaded:nTF
22 \prg_new_conditional:Npnn \xeCJK_if_package_loaded:n #1 { p , T , F , TF }
23 {
24   \tl_if_exist:cTF { ver@ #1 . \c__xeCJK_package_ext_tl }
25   { \prg_return_true: } { \prg_return_false: }
26 }
27 \tl_const:Ne \c__xeCJK_package_ext_tl { \@pkgextension }

```

下面这些 CJK 系列宏包不应该被使用。

```

28 \msg_new:nnn { xeCJK } { after-package }
29 {
30   The~`#1'~package~and~xeCJK~are~incompatible.\\\
31   Please~load~it~after~xeCJK.
32 }
33 \clist_map_inline:nn { CJKnumb }
34 {
35   \xeCJK_if_package_loaded:nT {#1}
36   { \msg_error:nnn { xeCJK } { after-package } {#1} }
37 }
38 \clist_map_inline:nn
39 { CJKulem , CJKvert , CJKpunct , CJKutf8 , CJK }
40 { \ctex_disable_package:n {#1} }

```

将 `CJKfntef` 包替换为 `xeCJKfntef` 包。

```

41 \ctex_if_format_at_least:nTF { 2020/10/01 }
42 { \ctex_replace_package:nn { CJKfntef } { xeCJKfntef } }
43 { \ctex_disable_package:n { CJKfntef } }

44 \cs_if_exist:NF \NewDocumentCommand
45 { \RequirePackage { xparse } }
46 \RequirePackage { xtemplate }

```

5.2 内部工具

分配临时变量。

```

47 \tl_new:N \l__xeCJK_tmp_tl
48 \tl_new:N \l__xeCJK_tmpb_tl
49 \int_new:N \l__xeCJK_tmp_int
50 \box_new:N \l__xeCJK_tmp_box
51 \dim_new:N \l__xeCJK_tmp_dim
52 \bool_new:N \l__xeCJK_tmp_bool
53 \skip_new:N \l__xeCJK_tmp_skip
54 \clist_new:N \l__xeCJK_tmp_clist

```

```

\__xeCJK_msg_new:nn 各种信息函数的缩略形式。
\__xeCJK_error:n
\__xeCJK_error:ne
\__xeCJK_warning:ne
\__xeCJK_info:nee
55 \cs_new_protected:Npn \__xeCJK_msg_new:nn { \msg_new:nnn { xeCJK } }
56 \cs_new_protected:Npn \__xeCJK_msg_new:nnn { \msg_new:nnnn { xeCJK } }
57 \cs_new_protected:Npn \__xeCJK_error:n { \msg_error:nn { xeCJK } }
58 \cs_new_protected:Npn \__xeCJK_error:ne { \msg_error:nne { xeCJK } }
59 \cs_new_protected:Npn \__xeCJK_warning:n { \msg_warning:nn { xeCJK } }
60 \cs_new_protected:Npn \__xeCJK_warning:ne { \msg_warning:nne { xeCJK } }
61 \cs_new_protected:Npn \__xeCJK_warning:nee { \msg_warning:nnee { xeCJK } }
62 \cs_new_protected:Npn \__xeCJK_warning:neee { \msg_warning:nneee { xeCJK } }
63 \cs_new_protected:Npn \__xeCJK_info:nee { \msg_info:nnee { xeCJK } }

\xeCJK_allow_break: 64 \cs_new_protected:Npn \xeCJK_allow_break:
\xeCJK_no_break: 65 { \tex_penalty:D \c_zero_int }
66 \cs_new_protected:Npn \xeCJK_no_break:
67 { \tex_penalty:D \c__xeCJK_nobreak_penalty_int }
68 \int_const:Nn \c__xeCJK_nobreak_penalty_int { 10 000 }

```

`__xeCJK_at_end_preamble:n` 在 `\document` 前后和宏包后加上各种钩子, 依赖 `ctexhook`。

```

\__xeCJK_after_preamble:n
\__xeCJK_after_end_preamble:n
\__xeCJK_package_hook:nn
69 \AtBeginDocument { \xeCJK@document@hook }
70 \ctex_at_end_preamble:n { \xeCJK@document@left@hook }
71 \ctex_after_end_preamble:n { \xeCJK@document@right@hook }
72 \cs_new_protected:Npn \xeCJK@document@hook
73 { \tl_use:N \g__xeCJK_after_preamble_hook_tl }
74 \cs_new_protected:Npn \xeCJK@document@left@hook
75 { \tl_use:N \g__xeCJK_at_end_preamble_hook_tl }
76 \cs_new_protected:Npn \xeCJK@document@right@hook
77 { \tl_use:N \g__xeCJK_after_end_preamble_hook_tl }
78 \cs_new_protected:Npn \__xeCJK_at_end_preamble:n
79 { \tl_gput_right:Nn \g__xeCJK_at_end_preamble_hook_tl }
80 \cs_new_protected:Npn \__xeCJK_after_preamble:n
81 { \tl_gput_right:Nn \g__xeCJK_after_preamble_hook_tl }
82 \cs_new_protected:Npn \__xeCJK_after_end_preamble:n
83 { \tl_gput_right:Nn \g__xeCJK_after_end_preamble_hook_tl }
84 \cs_new_protected:Npn \__xeCJK_package_hook:nn
85 { \ctex_at_end_package:nn }
86 \tl_new:N \g__xeCJK_at_end_preamble_hook_tl
87 \tl_new:N \g__xeCJK_after_preamble_hook_tl
88 \tl_new:N \g__xeCJK_after_end_preamble_hook_tl

```

`\xeCJKShipoutHook` 在 `\shipout` 盒子里加钩子, 可以影响到页眉页脚。`\AtBeginDvi` 将参数保存在盒子中, 而 `atbegshi` 的 `\AtBeginShipout` 在 `\shipout` 盒子构建好之后才起作用, 所以它们都影响不到页眉页脚。我们通过往 `\@begindvi` 里加入钩子来完成。注意, 第一次使用 `\@begindvi` 之后, 它会将自身定义为 `\@empty`。

```

89 \__xeCJK_after_preamble:n
90 { \tl_put_right:Nn \@begindvi { \xeCJK@first@begindvi } }
91 \cs_new_protected:Npn \xeCJK@first@begindvi
92 {
93   \xeCJKShipoutHook
94   \cs_if_exist:NTF \@begindvi
95     { \tl_gput_right:Nn }
96     { \tl_const:Nn }
97   \@begindvi { \xeCJKShipoutHook }
98 }
99 \NewDocumentCommand \xeCJKShipoutHook { }
100 {
101   \bool_if:NF \l__xeCJK_shipout_hook_bool
102   {
103     \bool_set_true:N \l__xeCJK_shipout_hook_bool
104     \tl_use:N \l__xeCJK_shipout_hook_tl
105   }
106 }

```

`\xeCJK_add_to_shipout:n` 往 `\shipout` 盒子中加入钩子。

```
107 \cs_new_protected:Npn \xeCJK_add_to_shipout:n
108   { \tl_put_right:Nn \l__xeCJK_shipout_hook_tl }
109 \tl_new:N \l__xeCJK_shipout_hook_tl
110 \bool_new:N \l__xeCJK_shipout_hook_bool
```

`\xeCJK_tl_remove_outer_braces:n` 去掉 #1 外层的分组括号。

```
111 \cs_new:Npn \xeCJK_tl_remove_outer_braces:n #1
112   {
113     \exp_last_unbraced:Ne
114     \__xeCJK_tl_remove_outer_braces:w { \tl_trim_spaces:n {#1} } \s_stop
115   }
116 \cs_new:Npn \__xeCJK_tl_remove_outer_braces:w #1 \s_stop
117   {
118     \tl_if_single:nTF {#1}
119     {
120       \tl_if_head_is_N_type:nTF {#1}
121       { \tl_trim_spaces:n }
122       { \xeCJK_tl_remove_outer_braces:n }
123     }
124     { \tl_trim_spaces:n }
125     {#1}
126   }
```

`\xeCJK_cs_clear:N` 让控制序列的意义为空。

```
\xeCJK_cs_gclear:N
127 \cs_new_protected:Npn \xeCJK_cs_clear:N #1
128   { \cs_set_eq:NN #1 \prg_do_nothing: }
129 \cs_new_protected:Npn \xeCJK_cs_gclear:N #1
130   { \cs_gset_eq:NN #1 \prg_do_nothing: }
```

`\xeCJK_swap_cs:NN` 交换 #1 和 #2 的意义。

```
131 \cs_new_protected:Npn \xeCJK_swap_cs:NN #1#2
132   {
133     \cs_set_eq:NN \__xeCJK_swap_cs_aux:w #1
134     \cs_set_eq:NN #1 #2
135     \cs_set_eq:NN #2 \__xeCJK_swap_cs_aux:w
136     \cs_undefine:N \__xeCJK_swap_cs_aux:w
137   }
```

`\xeCJK_font_gset_to_current:N` #1 是控制序列的名字, 令它等于当前字体命令。

```
138 \cs_new_protected:Npn \xeCJK_font_gset_to_current:N
139   { \exp_after:wN \__xeCJK_font_gset_to_current_aux:NN \tex_the:D \tex_font:D }
140 \cs_new_protected:Npn \__xeCJK_font_gset_to_current_aux:NN #1#2
141   { \cs_if_eq:NNF #1 \tex_nullfont:D { \cs_gset_eq:NN #2#1 } }
```

`\xeCJK_glyph_if_exist_p:N` 判断当前字体中是否含有字符 #1。fontspec 中的类似函数在判断为真的时候, 会留有一个
`\xeCJK_glyph_if_exist:NTF` `\scan_stop:`, 造成不必要的边界, 同时也不完全可展。因此, 我们重新定义它。

```
142 \prg_new_conditional:Npnn \xeCJK_glyph_if_exist:N #1 { p , T , F , TF }
143   {
144     \tex_iffontchar:D \tex_font:D `#1 \exp_stop_f:
145     \prg_return_true: \else: \prg_return_false: \fi:
146   }
```

`\c_xeCJK_space_skip_tl` 当前字体状态下, 一个字间空格产生的 `glue` 的长度, 包括伸展和收缩部分。

```
147 \tl_const:Nn \c_xeCJK_space_skip_tl
148   {
149     \int_compare:nNnTF \g__xeCJK_space_factor_int = { 1000 }
150     {
151       \skip_if_eq:nnTF \tex_spaceskip:D \c_zero_skip
152       {
153         \tex_fontdimen:D 2 ~ \tex_font:D
154         plus \tex_fontdimen:D 3 ~ \tex_font:D
```

```

155         minus \tex_fontdimen:D 4 ~ \tex_font:D
156     }
157     { \tex_spaceskip:D }
158 }
159 {
160     \skip_if_eq:nnTF \tex_spaceskip:D \c_zero_skip
161     {
162         \int_compare:nNnTF \g__xeCJK_space_factor_int < { 2000 }
163         {
164             \__xeCJK_space_skip_scale:nnn
165             { \tex_fontdimen:D 2 ~ \tex_font:D }
166         }
167         {
168             \skip_if_eq:nnTF \tex_xspaceskip:D \c_zero_skip
169             {
170                 \__xeCJK_space_skip_scale:nnn
171                 {
172                     \tex_fontdimen:D 2 ~ \tex_font:D +
173                     \tex_fontdimen:D 7 ~ \tex_font:D
174                 }
175             }
176             { \tex_xspaceskip:D \use_none:nn }
177         }
178         { \tex_fontdimen:D 3 ~ \tex_font:D }
179         { \tex_fontdimen:D 4 ~ \tex_font:D }
180     }
181     {
182         \int_compare:nNnTF \g__xeCJK_space_factor_int < { 2000 }
183         { \__xeCJK_space_skip_scale:nnn { \tex_spaceskip:D } }
184         {
185             \skip_if_eq:nnTF \tex_xspaceskip:D \c_zero_skip
186             {
187                 \__xeCJK_space_skip_scale:nnn
188                 {
189                     \tex_spaceskip:D +
190                     \tex_fontdimen:D 7 ~ \tex_font:D
191                 }
192             }
193             { \tex_xspaceskip:D \use_none:nn }
194         }
195         { \tex_gluestretch:D \tex_spaceskip:D }
196         { \tex_glueshrink:D \tex_spaceskip:D }
197     }
198 }
199 }
200 \cs_new:Npn \__xeCJK_space_skip_scale:nnn #1#2#3
201 {
202     \dim_eval:n {#1}
203     plus \fp_eval:n { \g__xeCJK_space_factor_int / 1000 } #2
204     minus
205     \int_div_truncate:nn
206     { 1000 * \int_value:w #3 } { \g__xeCJK_space_factor_int } sp
207 }

```

`\xeCJK_reset_space_factor:` 在 `\XeTeXinterchartoks` 里, `\spacefactor` 已经被重置为 1000。我们需要在 `Default` 类里保存 `\spacefactor` 用于计算空格宽度。

```

208 \int_new:N \g__xeCJK_space_factor_int
209 \cs_new_protected:Npn \xeCJK_reset_space_factor:
210 { \int_gset:Nn \g__xeCJK_space_factor_int { 1000 } }
211 \xeCJK_reset_space_factor:

```

`\xeCJK_glue_to_skip:nN` 取得一个 `glue` 的长度, 包括伸展和收缩部分。如果参数不是 `glue`, 则取其宽度。

```

212 \cs_new_protected:Npn \xeCJK_glue_to_skip:nN #1#2
213 {
214     \group_begin:
215     \hbox_set:Nw \l__xeCJK_tmp_box #1 \scan_stop:

```

```

216     \__xeCJK_if_last_glue:TF
217     {
218         \exp_args:NNNo \hbox_set_end:
219         \skip_set:Nn #2 { \skip_use:N \tex_lastskip:D }
220     }
221     {
222         \exp_args:NNNo \hbox_set_end:
223         \skip_set:Nn #2 { \dim_use:N \box_wd:N \l__xeCJK_tmp_box }
224     }
225     \exp_args:NNNo \group_end:
226     \skip_set:Nn #2 { \skip_use:N #2 }
227 }

```

`\xeCJK_int_until_do:nn` 由于定义较为简单, 可以比 `\int_until_do:nNnn` 稍微快一点点。
`__xeCJK_int_until_do:wn`

```

228 \cs_new_protected:Npn \xeCJK_int_until_do:nn #1#2
229 {
230     \__xeCJK_int_until_do:wn \use_none:n
231     { \reverse_if:N \if_int_compare:w #1#2 }
232 }
233 \cs_new_protected:Npn \__xeCJK_int_until_do:wn \use_none:n #1
234 { #1 \exp_after:wN \__xeCJK_int_until_do:wn \fi: \use_none:n {#1} }
235 \int_new:N \l__xeCJK_begin_int
236 \int_new:N \l__xeCJK_end_int

```

`\xeCJK_peek_catcode_ignore_spaces:NTF` 我们在里面设置了一个变量 `\l__xeCJK_peek_ignore_spaces_bool` 用于标识后面的空格是否被省略掉了。

```

237 \cs_new_protected:Npn \xeCJK_peek_catcode_ignore_spaces:NTF #1#2#3
238 {
239     \cs_set_eq:NN \l__xeCJK_peek_search_token #1 \scan_stop:
240     \cs_set_protected:Npe \__xeCJK_peek_catcode_true:w
241     { \exp_not:N \group_align_safe_end: \exp_not:n {#2} }
242     \cs_set_protected:Npe \__xeCJK_peek_catcode_false:w
243     { \exp_not:N \group_align_safe_end: \exp_not:n {#3} }
244     \bool_set_false:N \l__xeCJK_peek_ignore_spaces_bool
245     \group_align_safe_begin:
246     \peek_after:Nw \__xeCJK_peek_catcode_ignore_spaces_branches:w
247 }
248 \cs_new_protected:Npn \__xeCJK_peek_catcode_ignore_spaces_branches:w
249 {
250     \if_meaning:w \l_peek_token \c_space_token
251     \bool_set_true:N \l__xeCJK_peek_ignore_spaces_bool
252     \exp_after:wN \peek_after:Nw
253     \exp_after:wN \__xeCJK_peek_catcode_ignore_spaces_branches:w
254     \tex_romannumeral:D 0
255     \else:
256     \if_catcode:w
257     \exp_not:N \l_peek_token \exp_not:N \l__xeCJK_peek_search_token
258     \exp_after:wN \exp_after:wN
259     \exp_after:wN \__xeCJK_peek_catcode_true:w
260     \else:
261     \exp_after:wN \exp_after:wN
262     \exp_after:wN \__xeCJK_peek_catcode_false:w
263     \fi:
264     \fi:
265 }
266 \cs_new_eq:NN \l__xeCJK_peek_search_token ?
267 \cs_new_eq:NN \__xeCJK_peek_catcode_true:w \prg_do_nothing:
268 \cs_new_eq:NN \__xeCJK_peek_catcode_false:w \prg_do_nothing:
269 \bool_new:N \l__xeCJK_peek_ignore_spaces_bool

```

`\xeCJK_token_value_class:N` 用于取得记号 #1 所在的 X_YTeX 字符类。#1 应为 `\catcode` 为 11 或 12 的显性或隐性记号。

```

270 \cs_new:Npn \xeCJK_token_value_class:N #1
271 { \tex_XeTeXcharclass:D \xeCJK_token_value_charcode:N #1 }

```

`\xeCJK_token_value_charcode:N` 当记号 #1 的 charcode 大于等于 0x10000 时, XeTeX 0.9999.0 版以前的 `\meaning` 的返回结果比较特殊⁵, 需要特别处理。0.9999.0 版以后的 XeTeX 的 `\meaning` 对于超出 BMP 的字符, 会返回两个字符, 分别对应于其 UTF-16 编码的首尾代理⁶。这一 Bug 在 TeX Live 2015 的 0.99992 版中得到修复⁷。

```

272 \cs_new:Npn \xeCJK_token_value_charcode:N #1
273   { \exp_after:wN \__xeCJK_get_charcode:w \token_to_meaning:N #1 \q_stop }
274 \group_begin:
275   \cs_set:Npn \__xeCJK_tmp:w #1 ~ #2 ~ #3#4#5 \q_stop
276     {
277       \tl_if_empty:nTF { #4#5 }
278         {
279           \cs_new:Npn \__xeCJK_get_charcode:w ##1 ~ ##2 ~ ##3 \q_stop
280             { \int_eval:n { `##3 } }
281         }
282       {
283         \tl_if_empty:nTF {#5}
284           {
285             \cs_new:Npn \__xeCJK_get_charcode:w ##1 ~ ##2 ~ ##3##4 \q_stop
286               {
287                 \int_eval:n
288                   {
289                     \tl_if_empty:nTF { ##4 }
290                       { `##3 }
291                       { ( `##3 - "D800 ) * "400 + ( `##4 - "DC00 ) + "10000 }
292                   }
293             }
294           }
295         {
296           \cs_new:Npn \__xeCJK_get_charcode:w ##1 ~ ##2 ~ ##3##4 \q_stop
297             { \int_eval:n { \tl_if_empty:nTF { ##4 } { `##3 } { "20000 } } }
298         }
299       }
300   }
301   \exp_after:wN \__xeCJK_tmp:w \token_to_meaning:N ~~~~~20000 { } \q_stop
302 \group_end:

```

`\xeCJK_if_CJK_class_p:N` 判断字符 #1 是否为 CJK 字符类, 包括文字和标点符号。

```

\__xeCJK_if_CJK_class:NNTF
303 \prg_new_conditional:Npnn \xeCJK_if_CJK_class:N #1 { p , T , F , TF }
304   {
305     \if_cs_exist:w
306       \__xeCJK_CJK_class_tl:n { \xeCJK_token_value_class:N #1 }
307     \cs_end:
308     \prg_return_true: \else: \prg_return_false: \fi:
309   }
310 \cs_new:Npn \__xeCJK_CJK_class_tl:n #1
311   { c__xeCJK_CJK_class_ \int_eval:n {#1} _tl }

```

`\xeCJK_if_same_class_p:NN` 判断两个字符是否同属于一个字符类。

```

\__xeCJK_if_same_class:NNTF
312 \prg_new_conditional:Npnn \xeCJK_if_same_class:NN #1#2 { p , T , F , TF }
313   {
314     \if_int_compare:w \xeCJK_token_value_class:N #1 =
315                       \xeCJK_token_value_class:N #2 \exp_stop_f:
316     \prg_return_true: \else: \prg_return_false: \fi:
317   }

```

`\xeCJK_make_boundary:` 利用 `\scan_stop:` 结束 CJK 分组, 用于恢复字体等。

```

318 \cs_new_protected:Npn \xeCJK_make_boundary:
319   { \bool_if:NT \l__xeCJK_CJK_group_bool { \scan_stop: } }

```

⁵ 参见 <http://tug.org/pipermail/xetex/2013-January/023967.html> 和 <http://tex.stackexchange.com/a/64848>。

⁶ 参见 <http://tug.org/pipermail/xetex/2013-June/024543.html>。

⁷ 参见 <http://tug.org/pipermail/xetex/2015-May/025941.html>。

5.3 功能开关

`xeCJKActive` 事实上, 将开启或关闭 Xe_{La}TeX 的整个字符类机制。

```

320 \keys_define:nn { xeCJK / options }
321   {
322     xeCJKActive .choice: ,
323     xeCJKActive / true .code:n = { \makeXeCJKActive } ,
324     xeCJKActive / false .code:n = { \makeXeCJKInactive } ,
325     xeCJKActive .default:n = { true }
326   }

\makeXeCJKActive 327 \NewDocumentCommand \makeXeCJKActive { }
\makeXeCJKInactive 328 { \tex_ XeTeXinterchartokenstate:D = \c_one_int }
329 \NewDocumentCommand \makeXeCJKInactive { }
330 { \tex_ XeTeXinterchartokenstate:D = \c_zero_int }

```

抑制 BOM 和零宽格式字符, 避免触发错误的 inter-class toks 插入。

```

331 \char_set_catcode_ignore:n { "FEFF }
332 \char_set_catcode_ignore:n { "200B }
333 \char_set_catcode_ignore:n { "200C }
334 \char_set_catcode_ignore:n { "200D }
335 \char_set_catcode_ignore:n { "2060 }

```

5.4 字符类别设定

`\g__xeCJK_class_seq` 和 `\g__xeCJK_new_class_seq` 分别用于记录在 xeCJK 中使用的字符类别名称和新建的字符类别的编号。

```

336 \seq_new:N \g__xeCJK_class_seq
337 \seq_new:N \g__xeCJK_new_class_seq

```

`\xeCJK_new_class:n` 新建一个字符类别。#1 是自定义名称。

```

338 \cs_new_protected:Npn \xeCJK_new_class:n #1
339   {
340     \int_if_exist:cTF { \__xeCJK_class_csname:n {#1} }
341     { \__xeCJK_error:ne { class-already-defined } {#1} }
342     {
343       \exp_args:Nc \newXeTeXintercharclass
344       { \__xeCJK_class_csname:n {#1} }
345       \clist_new:c { g__xeCJK_#1_range_clist }
346       \seq_gput_right:Nn \g__xeCJK_class_seq {#1}
347       \seq_gput_right:Nv \g__xeCJK_new_class_seq
348       { \__xeCJK_class_csname:n {#1} }
349     }
350   }

```

`\xeCJK_save_class:nn` 保存 Xe_{La}TeX 预定义的字符类别。#1 是自定义名称, #2 是编号。

```

351 \cs_new_protected:Npn \xeCJK_save_class:nn #1#2
352   {
353     \int_if_exist:cTF { \__xeCJK_class_csname:n {#1} }
354     { \__xeCJK_error:ne { class-already-defined } {#1} }
355     {
356       \int_const:cn { \__xeCJK_class_csname:n {#1} } {#2}
357       \clist_new:c { g__xeCJK_#1_range_clist }
358       \seq_gput_right:Nn \g__xeCJK_class_seq {#1}
359     }
360   }

```

`__xeCJK_class_csname:n` 字符类名称对应的控制序列名字。

```

361 \cs_new:Npn \__xeCJK_class_csname:n #1 { c__xeCJK_#1_class_int }
362 \cs_new_eq:cN { \__xeCJK_class_csname:n { Others } } \l__xeCJK_tmp_int
363 \__xeCJK_msg_new:nn { class-already-defined }
364   {
365     XeTeX~character~class~`#1!~has~been~already~defined.\\\
366     Please~take~another~name. \
367   }

```

xeCJK 需要以下字符类别用于字符输出。其中 Default、CJK、FullLeft、FullRight、Boundary 为 X_YTeX 中预定义的类别，xeCJK 新增加了 HalfLeft、HalfRight、NormalSpace 和 CM。其中异体字选择符 (Ideographic Variation Selectors)⁸ 需要 X_YTeX 0.9999.0 以上的版本⁹和相关字体的支持。

类别	说明	例子
Default	西文一般符号	abc123
CJK	CJK 表意符号	汉字 あいう
FullLeft	全角左标点	(《: “
FullRight	全角右标点	, 。) 》”
HalfLeft	半角左标点	([{
HalfRight	半角右标点	, . ?)] }
NormalSpace	前后原始间距的符号	/
Boundary	边界	空格
CM	组合标识	异体字选择符
HangulJamo	朝鲜文字母	ㅁ . ㅂ ㅅ

```

Default 368 \xeCJK_save_class:n { Default } { 0 }
CJK
FullLeft XYTeX 0.99994 将字符类总数扩大到 409610。
FullRight
Boundary 369 \str_const:Nn \c__xeCJK_xetex_version_str
370 { \int_use:N \tex_XeTeXversion:D \tex_XeTeXrevision:D }
371 \fp_compare:nNnTF { \c__xeCJK_xetex_version_str } > { 0.99993 }
372 { \xeCJK_save_class:n { Boundary } { 4095 } }
373 { \xeCJK_save_class:n { Boundary } { 255 } }

```

L^AT_EX 2_ε 2016/02/01 不再预设置 CJK 字符类。

```

374 \int_compare:nNnTF { \tex_XeTeXcharclass:D "4E00 } = \c_one_int
375 {
376   \xeCJK_save_class:n { CJK } { 1 }
377   \xeCJK_save_class:n { FullLeft } { 2 }
378   \xeCJK_save_class:n { FullRight } { 3 }
379   \int_const:Nn \c__xeCJK_class_begin_int { 3 }
380 }
381 {
382   \xeCJK_new_class:n { CJK }
383   \xeCJK_new_class:n { FullLeft }
384   \xeCJK_new_class:n { FullRight }
385   \int_const:Nn \c__xeCJK_class_begin_int { 0 }
386 }

```

```

HalfLeft 新增西文半角左/右标点、前后原始间距的符号和异体字选择符类。
HalfRight
NormalSpace 387 \xeCJK_new_class:n { HalfLeft }
CM 388 \xeCJK_new_class:n { HalfRight }
HangulJamo 389 \xeCJK_new_class:n { NormalSpace }
390 \xeCJK_new_class:n { CM }
391 \xeCJK_new_class:n { HangulJamo }

```

西文半角左/右标点和前后原始间距的字符类。

```

\c__xeCJK_HalfLeft_chars_clist
\c__xeCJK_HalfRight_chars_clist
\c__xeCJK_NormalSpace_chars_clist
392 \clist_const:Nn \c__xeCJK_HalfLeft_chars_clist
393 { "28 , "5B , "60 , "7B , "2329 }
394 \clist_const:Nn \c__xeCJK_HalfRight_chars_clist
395 { "21 , "22 , "25 , "27 , "29 , "2C , "2E , "3A , "3B , "3F , "5D , "7D , "232A }
396 \clist_const:Nn \c__xeCJK_NormalSpace_chars_clist { "2D , "2F , "5C }

```

⁸<http://www.unicode.org/reports/tr37/>

⁹<http://tug.org/pipermail/xetex/2013-March/024118.html>

¹⁰<http://tug.org/pipermail/xetex/2016-February/026363.html>

以下对全角标点符号的归类来源于 Xe_{La}TeX 的脚本 `unicode-char-prep.pl` 和 Unicode 数据库¹¹。

```
\c__xeCJK_OP_chars_clist
```

Open Punctuation (OP)													
U+2018	‘	U+201C	“	U+3008	◁	U+300A	◁◁	U+300C	⌈	U+300E	⌈⌈	U+3010	【
U+3014	〔	U+3016	〔〔	U+3018	〔〔	U+301A	〔〔	U+301D	˘	U+FE17	⌋	U+FE35	(
U+FE37	˘	U+FE39	⌋	U+FE3B	⌋	U+FE3D	⌋	U+FE3F	⌋	U+FE41	⌋	U+FE43	⌋
U+FE47	⌋	U+FE59	(U+FE5B	{	U+FE5D	{	U+FF08	(U+FF3B	[U+FF5B	{
U+FF5F	〔	U+FF62	⌈										

以下代码的第一行是中西文共用的左引号。

```
397 \clist_const:Nn \c__xeCJK_OP_chars_clist
398 {
399   "2018 , "201C ,
400   "3008 , "300A , "300C , "300E , "3010 , "3014 , "3016 , "3018 , "301A , "301D ,
401   "FE17 , "FE35 , "FE37 , "FE39 , "FE3B , "FE3D , "FE3F , "FE41 , "FE43 , "FE47 ,
402   "FE59 , "FE5B , "FE5D , "FF08 , "FF3B , "FF5B , "FF5F , "FF62
403 }
```

```
\c__xeCJK_PR_chars_clist
```

Prefix Numeric (PR)									
U+FE69	\$	U+FF04	\$	U+FFE1	£	U+FFE5	¥	U+FFE6	₩

```
404 \clist_const:Nn \c__xeCJK_PR_chars_clist
405 { "FE69 , "FF04 , "FFE1 , "FFE5 , "FFE6 }
```

以上两类标点符号出现在文字的左边, 不应出现在行尾位置。

```
\c__xeCJK_FullLeft_chars_clist
```

```
406 \clist_const:Nn \c__xeCJK_FullLeft_chars_clist
407 {
408   \exp_not:V \c__xeCJK_OP_chars_clist ,
409   \exp_not:V \c__xeCJK_PR_chars_clist
410 }
```

```
\c__xeCJK_CL_chars_clist
```

Close Punctuation (CL)													
U+00B7	·	U+2019	’	U+201D	”	U+2013	—	U+2014	—	U+2025	⋯	U+2026	⋯
U+2027	·	U+2E3A	⋯	U+3001	、	U+3002	。	U+3009	〉	U+300B	〉〉	U+300D	」
U+300F	』	U+3011	〕	U+3015	〕	U+3017	〕	U+3019	〕	U+301B	〕	U+301E	〕
U+301F	˘	U+FE11	˘	U+FE12	˘	U+FE18	˘	U+FE36	˘	U+FE38	˘	U+FE3A	˘
U+FE3C	˘	U+FE3E	˘	U+FE40	˘	U+FE42	˘	U+FE44	˘	U+FE48	˘	U+FE50	˘
U+FE52	˘	U+FE5A	˘	U+FE5C	˘	U+FE5E	˘	U+FF09	˘	U+FF0C	˘	U+FF0E	˘
U+FF3D	〕	U+FF5D	〕	U+FF60	〕	U+FF61	。	U+FF63	〕	U+FF64	、		

以下代码的第一行是中西文共用的一些标点符号。

```
411 \clist_const:Nn \c__xeCJK_CL_chars_clist
412 {
413   "00B7 , "2019 , "201D , "2013 , "2014 , "2025 , "2026 , "2027 , "2E3A ,
414   "3001 , "3002 , "3009 , "300B , "300D , "300F , "3011 , "3015 , "3017 , "3019 ,
415   "301B , "301E , "301F , "FE11 , "FE12 , "FE18 , "FE36 , "FE38 , "FE3A , "FE3C ,
416   "FE3E , "FE40 , "FE42 , "FE44 , "FE48 , "FE50 , "FE52 , "FE5A , "FE5C , "FE5E ,
417   "FF09 , "FF0C , "FF0E , "FF3D , "FF5D , "FF60 , "FF61 , "FF63 , "FF64
418 }
```

```
\c__xeCJK_NS_chars_clist
```

Nonstarter (NS)													
U+30FB	•	U+FE54	;	U+FE55	:	U+FF1A	:	U+FF1B	;	U+FF65	•	U+16FE0	☒

Hyphens (cl-03)

U+301C	～	U+30A0	=	U+FF5E	～
--------	---	--------	---	--------	---

Iteration marks (cl-09)

U+3005	々	U+303B	ゞ	U+309D	ゞ	U+309E	ゞ	U+30FD	ゞ	U+30FE	ゞ
--------	---	--------	---	--------	---	--------	---	--------	---	--------	---

¹¹<http://www.unicode.org/reports/tr14/>

根据 W3C 的资料¹², c1-03 和 c1-09 在非常松散的情况下可以没有禁则。我们将全角浪线 U+FF5E 等连接号归入 FullRight 类并在宏包末尾设为 LongPunct。

```

419 \clist_const:Nn \c__xeCJK_hyphens_chars_clist
420 { "301C , "30A0 , "FF5E }
421 \clist_const:Nn \c__xeCJK_iteration_marks_chars_clist
422 { "3005 , "303B , "309D , "309E , "30FD , "30FE }
423 \clist_const:Nn \c__xeCJK_NS_chars_clist
424 { "30FB , "FE54 , "FE55 , "FF1A , "FF1B , "FF65 , "16FE0 }
425 \AtEndOfPackage
426 {
427   \cs_set:Npn \__xeCJK_tmp:w #1
428     { \char_generate:nn {#1} { 12 } }
429   \__xeCJK_add_special_punct:nn { long }
430     { \clist_map_function:NN \c__xeCJK_hyphens_chars_clist \__xeCJK_tmp:w }
431   \cs_undefine:N \__xeCJK_tmp:w
432 }

```

\c__xeCJK_EX_chars_clist Exclamation/Interrogation (EX)

| U+FE15 | ! | U+FE16 | ? | U+FE56 | ? | U+FE57 | ! | U+FF01 | ! | U+FF1F | ? |

```

433 \clist_const:Nn \c__xeCJK_EX_chars_clist
434 { "FE15 , "FE16 , "FE56 , "FE57 , "FF01 , "FF1F }

```

\c__xeCJK_IS_chars_clist Infix Numeric Separator (IS)

| U+FE10 | ' | U+FE13 | : | U+FE14 | ; |

```

435 \clist_const:Nn \c__xeCJK_IS_chars_clist { "FE10 , "FE13 , "FE14 }

```

\c__xeCJK_CJ_chars_clist Conditional Japanese Starter (CJ)。这类字符的禁则是可选的¹³, 为实现的简单计, 我们把它们归入 CJK 类, 即没有禁则。

U+3041	あ	U+3043	い	U+3045	う	U+3047	え	U+3049	お	U+3063	つ	U+3083	や
U+3085	ゆ	U+3087	よ	U+308E	わ	U+3095	か	U+3096	け	U+30A1	ア	U+30A3	イ
U+30A5	ウ	U+30A7	エ	U+30A9	オ	U+30C3	ツ	U+30E3	ヤ	U+30E5	ユ	U+30E7	ヨ
U+30EE	ワ	U+30F5	カ	U+30F6	ケ	U+30FC	一	U+31F0	ク	U+31F1	シ	U+31F2	ス
U+31F3	ト	U+31F4	ヌ	U+31F5	ハ	U+31F6	ヒ	U+31F7	フ	U+31F8	ヘ	U+31F9	ホ
U+31FA	ム	U+31FB	ラ	U+31FC	リ	U+31FD	ル	U+31FE	レ	U+31FF	ロ	U+FF67	ア
U+FF68	イ	U+FF69	ウ	U+FF6A	エ	U+FF6B	オ	U+FF6C	ヤ	U+FF6D	ユ	U+FF6E	ヨ
U+FF6F	ツ	U+FF70	ー										

```

436 \clist_const:Nn \c__xeCJK_CJ_chars_clist
437 {
438   "3041 , "3043 , "3045 , "3047 , "3049 , "3063 , "3083 , "3085 , "3087 , "308E ,
439   "3095 , "3096 , "30A1 , "30A3 , "30A5 , "30A7 , "30A9 , "30C3 , "30E3 , "30E5 ,
440   "30E7 , "30EE , "30F5 , "30F6 , "30FC , "31F0 , "31F1 , "31F2 , "31F3 , "31F4 ,
441   "31F5 , "31F6 , "31F7 , "31F8 , "31F9 , "31FA , "31FB , "31FC , "31FD , "31FE ,
442   "31FF , "FF67 , "FF68 , "FF69 , "FF6A , "FF6B , "FF6C , "FF6D , "FF6E , "FF6F ,
443   "FF70
444 }

```

\c__xeCJK_PO_chars_clist Postfix Numeric (PO)

| U+FE6A | % | U+FF05 | % | U+FFE0 | ¢ |

```

445 \clist_const:Nn \c__xeCJK_PO_chars_clist { "FE6A , "FF05 , "FFE0 }

```

\c__xeCJK_FullRight_chars_clist 以上六类标点符号出现在文字的右边, 不应出现在行首位置。

```

446 \clist_const:Ne \c__xeCJK_FullRight_chars_clist
447 {
448   \exp_not:V \c__xeCJK_CL_chars_clist ,
449   \exp_not:V \c__xeCJK_NS_chars_clist ,

```

¹²<http://www.w3.org/TR/jlreq/>

¹³<https://github.com/CTeX-org/ctex-kit/issues/165>

```

450 \exp_not:V \c__xeCJK_EX_chars_clist ,
451 \exp_not:V \c__xeCJK_IS_chars_clist ,
452 \exp_not:V \c__xeCJK_PO_chars_clist ,
453 \exp_not:V \c__xeCJK_hyphens_chars_clist
454 }

```

\c__xeCJK_CJK_chars_clist CJK 字符类, 包括文字和标点符号。

```

455 \clist_const:Nn \c__xeCJK_CJK_chars_clist
456 {

```

- 闽南语、客家话阴去和阳去声调标记

```
457 "02EA -> "02EB ,
```

- CJK Radicals Supplement (中日韩部首补充)

```
458 "2E80 -> "2EFF ,
```

- Kangxi Radicals (康熙字典部首)

```
459 "2F00 -> "2DFD ,
```

- Ideographic Description Characters (表意文字描述符)

```
460 "2FF0 -> "2FFF ,
```

- CJK Symbols and Punctuation (中日韩符号和标点)

```
461 "3000 -> "303F ,
```

- Hiragana (日文平假名)

```
462 "3040 -> "309F ,
```

- Katakana (日文片假名)

```
463 "30A0 -> "30FF ,
```

- Bopomofo (注音字母)

```
464 "3100 -> "312F ,
```

- Hangul Compatibility Jamo (谚文兼容字母)

```
465 "3130 -> "318F ,
```

- Kanbun (象形字注释标志)

```
466 "3190 -> "319F ,
```

- Bopomofo Extended (注音字母扩展)

```
467 "31A0 -> "31BF ,
```

- CJK Strokes (中日韩笔画)

```
468 "31C0 -> "31EF ,
```

- Katakana Phonetic Extensions (日文片假名语音扩展)

```
469 "31F0 -> "31FF ,
```

- Enclosed CJK Letters and Months (带圈中日韩字母和月份)

```
470 "3200 -> "32FF ,
```

- CJK Compatibility (中日韩兼容)

```
471 "3300 -> "33FF ,
```

- CJK Unified Ideographs Extension-A (中日韩统一表意文字扩展 A)

```
472 "3400 -> "4DBF ,
```

- Yijing Hexagrams Symbols (易经六十四卦符号)
473 "4DC0 -> "4DFF ,
- CJK Unified Ideographs (中日韩统一表意文字)
474 "4E00 -> "9FFF ,
- Yi Syllables (彝文音节)
475 "A000 -> "A48F ,
- Yi Radicals (彝文字根)
476 "A490 -> "A4CF ,
- Hangul Syllables (谚文音节)
477 "AC00 -> "D7AF ,
- CJK Compatibility Ideographs (中日韩兼容表意文字)
478 "F900 -> "FAFF ,
- Vertical Forms (竖排形式)
479 "FE10 -> "FE1F ,
- CJK Compatibility Forms (中日韩兼容形式)
480 "FE30 -> "FE4F ,
- Halfwidth and Fullwidth Forms (半角及全角形式)
481 "FF00 -> "FFEF ,
- Ideographic Symbols and Punctuation (表意文字符号及标点)
482 "16FE0 -> "16FFF ,
- Tangut (西夏文)
483 "17000 -> "187FF ,
- Tangut Components (西夏文部首)
484 "18800 -> "18AFF ,
- Khitan Small Script (契丹小字)
485 "18B00 -> "18CFF ,
- Tangut Supplement (西夏文增补)
486 "18D00 -> "18D7F ,
- Tangut Components Supplement (西夏文部首增补)
487 "18D80 -> "18DFF ,
- Kana Extended-B (日文假名扩展 B)
488 "1AFF0 -> "1AFFF ,
- Kana Supplement (日文假名增补)
489 "1B000 -> "1B0FF ,
- Kana Extended-A (日文假名扩展 A)
490 "1B100 -> "1B12F ,
- Small Kana Extension (小型日文假名扩展)
491 "1B130 -> "1B16F ,
- Nushu (女书)
492 "1B170 -> "1B2FF ,

- Enclosed Ideographic Supplement (带圈表意文字增补)


```
493 "1F200 -> "1F2FF ,
```
- CJK Unified Ideographs Extension-B (中日韩统一表意文字扩展 B)


```
494 "20000 -> "2A6DF ,
```
- CJK Unified Ideographs Extension-C (中日韩统一表意文字扩展 C)


```
495 "2A700 -> "2B73F ,
```
- CJK Unified Ideographs Extension-D (中日韩统一表意文字扩展 D)


```
496 "2B740 -> "2B81F ,
```
- CJK Unified Ideographs Extension-E (中日韩统一表意文字扩展 E)


```
497 "2B820 -> "2CEAF ,
```
- CJK Unified Ideographs Extension-F (中日韩统一表意文字扩展 F)


```
498 "2CEB0 -> "2EBEF ,
```
- CJK Unified Ideographs Extension-I (中日韩统一表意文字扩展 I)


```
499 "2EBF0 -> "2EE5F ,
```
- CJK Compatibility Ideographs Supplement (中日韩兼容表意文字增补)


```
500 "2F800 -> "2FA1F ,
```
- CJK Unified Ideographs Extension-G (中日韩统一表意文字扩展 G)


```
501 "30000 -> "3134F ,
```
- CJK Unified Ideographs Extension-H (中日韩统一表意文字扩展 H)


```
502 "31350 -> "323AF ,
```
- CJK Unified Ideographs Extension-J (中日韩统一表意文字扩展 J)


```
503 "323B0 -> "3347F
504 }
```

`\c__xeCJK_CM_chars_clist` 包括日文假名浊点和异体字选择符。组合标识最好是归入 256 类,即透明类,不会影响状态。但也会产生一定问题。比如下面的例子,位于行尾的“二”造成分组不匹配。

```
\XeTeXinterchartokenstate=1
\XeTeXcharclass`二=256
\XeTeXinterchartoks 255 1 = {\bgroup}
\XeTeXinterchartoks 1 255 = {\egroup}
\XeTeXinterchartoks 1 1 = {x}
\font\zhfont="SimSun"
\zhfont
一二三二
\bye

505 \clist_const:Nn \c__xeCJK_CM_chars_clist
506 {
```

- Diacritics (音调符号)


```
507 "302A -> "302F ,
```
- 日文假名浊点


```
508 "3099 -> "309A ,
```
- Variation Selectors (异体字选择符)


```
509 "FE00 -> "FE0F ,
```
- Variation Selectors Supplement (异体字选择符增补)


```
510 "E0100 -> "E01EF
```

```
511 }
```

\c__xeCJK_HangulJamo_chars_clist 朝鲜文字母。

```
512 \clist_const:Nn \c__xeCJK_HangulJamo_chars_clist
513 {
```

- Hangul Jamo (谚文字母)

```
514 "1100 -> "11FF ,
```

- Hangul Jamo Extended-A (谚文扩展 A)

```
515 "A960 -> "A97F ,
```

- Hangul Jamo Extended-B (谚文扩展 B)

```
516 "D7B0 -> "D7FF
```

```
517 }
```

5.5 字符类别处理

\xeCJK_class_num:n #1 为字符类别名称,用于取得字符类别对应的编号。

```
518 \cs_new:Npn \xeCJK_class_num:n #1
519 { \use:c { \__xeCJK_class_csname:n {#1} } }
```

```
\xeCJKDeclareCharClass 520 \NewDocumentCommand \xeCJKDeclareCharClass { s > { \TrimSpaces } m m }
521 {
522   \xeCJK_declare_char_class:nn {#2} {#3}
523   \IfBooleanT {#1} { \xeCJKResetPunctClass }
524 }
```

\xeCJK_declare_char_class:nn 用于设置字符所属的类别, #1 为类别名称, #2 为字符的 Unicode, 相邻字符用半角逗号隔开, 支持类似 "1100 -> "11FF 起止范围的使用方式。

\xeCJK_declare_char_class:nN
_xeCJK_set_char_class_aux:Nnw

```
525 \cs_new_protected:Npn \xeCJK_declare_char_class:nn #1#2
526 {
527   \clist_set:Nc \l__xeCJK_tmp_clist {#2}
528   \xeCJK_declare_char_class:nN {#1} \l__xeCJK_tmp_clist
529 }
530 \cs_new_protected:Npn \xeCJK_declare_char_class:nN #1#2
531 {
532   \clist_gconcat:ccN
533     { g__xeCJK_#1_range_clist } { g__xeCJK_#1_range_clist } #2
534   \clist_map_inline:Nn #2
535     {
536       \str_if_eq:nnF {##1} { -> }
537       {
538         \__xeCJK_set_char_class_aux:Nnw \xeCJK_set_char_class:nnn {##1}
539         { \xeCJK_class_num:n {#1} }
540       }
541     }
542   \xeCJK_set_char_class:nnn { "3099 } { "309A } { \xeCJK_class_num:n { CM } }
543 }
544 \NewDocumentCommand \__xeCJK_set_char_class_aux:Nnw
545 { m > { \SplitArgument { 1 } { -> } } m } { #1 #2 }
546 \cs_generate_variant:Nn \clist_gconcat:NNN { cc }
```

_xeCJK_check_num_range:nnNN

```
547 \cs_new_protected:Npn \__xeCJK_check_num_range:nnNN #1#2#3#4
548 {
549   \bool_lazy_or:nnTF
550     { \tl_if_blank_p:n {#1} }
551     { \tl_if_blank_p:n {#2} }
552     {
553       \int_set:Nn #3 { \tl_if_blank:nTF {#1} {#2} {#1} }
554       \int_set_eq:NN #3 #4
555     }
556 }
```

```

557     \int_set:Nn #3 { \int_min:nn {#1} { \tl_if_novalue:nTF {#2} {#1} {#2} } }
558     \int_set:Nn #4 { \int_max:nn {#1} { \tl_if_novalue:nTF {#2} {#1} {#2} } }
559   }
560 }

561 \token_if_letter:NF ^^^ac00
562 {
563   \int_set:Nn \l__xeCJK_begin_int { "AC00 }
564   \int_set:Nn \l__xeCJK_end_int   { "D7A3 }
565   \xeCJK_int_until_do:nn { \l__xeCJK_begin_int > \l__xeCJK_end_int }
566   {
567     \char_set_catcode_letter:n { \l__xeCJK_begin_int }
568     \int_incr:N \l__xeCJK_begin_int
569   }
570 }

```

`\xeCJK_set_char_class:nnn` 设置字符类别, #1 和 #2 为字符类别起止的 Unicode, #3 为类别名称对应编号。

```

571 \cs_new_protected:Npn \xeCJK_set_char_class:nnn #1#2#3
572 {
573   __xeCJK_check_num_range:nnNN {#1} {#2} \l__xeCJK_begin_int \l__xeCJK_end_int
574   \int_set:Nn \l__xeCJK_tmp_int {#3}
575   \xeCJK_int_until_do:nn { \l__xeCJK_begin_int > \l__xeCJK_end_int }
576   {
577     \tex_XeTeXcharclass:D \l__xeCJK_begin_int = \l__xeCJK_tmp_int
578     \int_incr:N \l__xeCJK_begin_int
579   }
580 }

```

`__xeCJK_set_char_class_eq:nn` 将字符类 #1 中的字符全部设置成字符类 #2。只适用于 #1 的字符类范围为离散的逗号列表的情况。

```

581 \cs_new_protected:Npn \__xeCJK_set_char_class_eq:nn #1#2
582 {
583   \int_set:Nn \l__xeCJK_tmp_int { \xeCJK_class_num:n {#2} }
584   \clist_map_inline:cn { c__xeCJK_#1_chars_clist }
585     { \tex_XeTeXcharclass:D ##1 = \l__xeCJK_tmp_int }
586 }

```

`\normalspacedchars` 声明前后不加间距的字符。

```

587 \NewDocumentCommand \normalspacedchars { m }
588 {
589   \tl_map_inline:nn {#1}
590     { \tex_XeTeXcharclass:D `##1 = \xeCJK_class_num:n { NormalSpace } }
591 }

```

`\xeCJKResetPunctClass` 用于重置标点符号所属的字符类。

```

592 \NewDocumentCommand \xeCJKResetPunctClass { }
593 {
594   \clist_gclear:N \g__xeCJK_HalfLeft_range_clist
595   \clist_gclear:N \g__xeCJK_HalfRight_range_clist
596   \clist_gclear:N \g__xeCJK_FullLeft_range_clist
597   \clist_gclear:N \g__xeCJK_FullRight_range_clist
598   \xeCJK_declare_char_class:nN { HalfLeft } \c__xeCJK_HalfLeft_chars_clist
599   \xeCJK_declare_char_class:nN { HalfRight } \c__xeCJK_HalfRight_chars_clist
600   \xeCJK_declare_char_class:nN { FullLeft } \c__xeCJK_FullLeft_chars_clist
601   \xeCJK_declare_char_class:nN { FullRight } \c__xeCJK_FullRight_chars_clist
602 }

```

`\xeCJKResetCharClass` 用于恢复 xeCJK 对字符类别的设置。

```

603 \NewDocumentCommand \xeCJKResetCharClass { }
604 {
605   \clist_gclear:N \g__xeCJK_CJK_range_clist
606   \clist_gclear:N \g__xeCJK_NormalSpace_range_clist
607   \clist_gclear:N \g__xeCJK_CM_range_clist

```

```

608 \clist_gclear:N \g__xeCJK_HangulJamo_range_clist
609 \xeCJK_declare_char_class:nN { CJK } \c__xeCJK_CJK_chars_clist
610 \xeCJK_declare_char_class:nN { NormalSpace } \c__xeCJK_NormalSpace_chars_clist
611 \xeCJK_declare_char_class:nN { CM } \c__xeCJK_CM_chars_clist
612 \xeCJK_declare_char_class:nN { HangulJamo } \c__xeCJK_HangulJamo_chars_clist
613 \xeCJKResetPunctClass
614 }

```

设置字符类别。

```
615 \xeCJKResetCharClass
```

`\xeCJK_inter_class_toks:nnn` 在相邻类别之间插入内容。

```

616 \cs_new_protected:Npn \xeCJK_inter_class_toks:nnn #1#2#3
617 {
618   \tex_XeTeXinterchartoks:D \xeCJK_class_num:n {#1} ~
619   \xeCJK_class_num:n {#2} = {#3}
620 }
621 \cs_generate_variant:Nn \xeCJK_inter_class_toks:nnn { nne }

```

`\xeCJK_get_inter_class_toks:nn` 取出相邻类别之间的内容。

```

622 \cs_new:Npn \xeCJK_get_inter_class_toks:nn #1#2
623 {
624   \tex_the:D \tex_XeTeXinterchartoks:D \xeCJK_class_num:n {#1} ~
625   \xeCJK_class_num:n {#2}
626 }

```

`\xeCJK_clear_inter_class_toks:nn` 清除相邻类别之间的内容。注意，直接赋空值可能会导致 XeTeX 崩溃。例如

```

\XeTeXinterchartokenstate = 1
\XeTeXcharclass`A=10
\XeTeXinterchartoks 10 10 = {xx}
\begingroup
  \XeTeXinterchartoks 10 10 = {} AA
\endgroup
\bye

```

如果把上述例子中的分组 `\begingroup` 和 `\endgroup` 去掉，则结果正常，甚为怪异。此处 XeTeX 的 bug 已经在 0.999992 版中修复¹⁴。

```

627 \cs_new_protected:Npn \xeCJK_clear_inter_class_toks:nn #1#2
628 { \xeCJK_inter_class_toks:nnn {#1} {#2} { \prg_do_nothing: } }

```

`\xeCJK_pre_inter_class_toks:nnn` 在相邻类别之间已有的内容前增加内容。

```

629 \cs_new_protected:Npn \xeCJK_pre_inter_class_toks:nnn #1#2#3
630 {
631   \xeCJK_inter_class_toks:nne {#1} {#2}
632   { \exp_not:n {#3} \xeCJK_get_inter_class_toks:nn {#1} {#2} }
633 }
634 \cs_generate_variant:Nn \xeCJK_pre_inter_class_toks:nnn { nne }

```

`\xeCJK_app_inter_class_toks:nnn` 在相邻类别之间已有的内容后追加内容。

```

635 \cs_new_protected:Npn \xeCJK_app_inter_class_toks:nnn #1#2#3
636 {
637   \xeCJK_inter_class_toks:nne {#1} {#2}
638   { \xeCJK_get_inter_class_toks:nn {#1} {#2} \exp_not:n {#3} }
639 }
640 \cs_generate_variant:Nn \xeCJK_app_inter_class_toks:nnn { nne }

```

`\xeCJK_copy_inter_class_toks:nnnn` 将 #3 和 #4 之间的内容复制到 #1 和 #2 之间。

```

641 \cs_new_protected:Npn \xeCJK_copy_inter_class_toks:nnnn #1#2#3#4
642 {
643   \tl_set:Ne \l__xeCJK_tmp_tl
644   { \xeCJK_get_inter_class_toks:nn {#3} {#4} }

```

¹⁴<http://tug.org/svn/texlive?view=revision&revision=53880>

```

645 \tl_if_empty:NTF \l__xeCJK_tmp_tl
646 {
647   \tl_set:Ne \l__xeCJK_tmp_tl
648   { \xeCJK_get_inter_class_toks:nn {#1} {#2} }
649   \tl_if_empty:NF \l__xeCJK_tmp_tl
650   { \xeCJK_clear_inter_class_toks:nn {#1} {#2} }
651 }
652 { \xeCJK_inter_class_toks:nne {#1} {#2} { \exp_not:o \l__xeCJK_tmp_tl } }
653 }

```

\xeCJK_replace_inter_class_toks:nmmn 将 #1 和 #2 之间出现的 #3 用 #4 替换。

```

654 \cs_new_protected:Npn \xeCJK_replace_inter_class_toks:nmmn #1#2#3#4
655 {
656   \tl_set:Ne \l__xeCJK_tmp_tl
657   { \xeCJK_get_inter_class_toks:nn {#1} {#2} }
658   \tl_if_empty:NF \l__xeCJK_tmp_tl
659   {
660     \tl_replace_all:Nnn \l__xeCJK_tmp_tl {#3} {#4}
661     \xeCJK_inter_class_toks:nne {#1} {#2}
662     { \exp_not:o \l__xeCJK_tmp_tl }
663   }
664 }

```

\xeCJK_clear_Boundary_and_CJK_toks: 清除边界与 CJK 文字、全角左右标点之间的内容。

```

665 \cs_new_protected:Npn \xeCJK_clear_Boundary_and_CJK_toks:
666 { }
667 \cs_new_protected:Npn \__xeCJK_update_clear_toks:n #1
668 {
669   \cs_gset_protected:Npe \xeCJK_clear_Boundary_and_CJK_toks:
670   {
671     \exp_not:o { \xeCJK_clear_Boundary_and_CJK_toks: }
672     \tex_XeTeXinterchartoks:D
673     \xeCJK_class_num:n { Boundary } ~
674     \xeCJK_class_num:n {#1} = { \exp_not:N \prg_do_nothing: }
675   }
676 }

```

\g__xeCJK_base_class_seq 保存宏包预先定义的字符类。

```

\g__xeCJK_non_CJK_class_seq
\g__xeCJK_CJK_class_seq
677 \seq_new:N \g__xeCJK_base_class_seq
678 \seq_gset_eq:NN \g__xeCJK_base_class_seq \g__xeCJK_class_seq
679 \seq_new:N \g__xeCJK_non_CJK_class_seq
680 \seq_gset_from_clist:Nn \g__xeCJK_non_CJK_class_seq
681 { Default , HalfLeft , HalfRight , NormalSpace , Boundary }
682 \seq_new:N \g__xeCJK_CJK_class_seq
683 \cs_new_protected:Npn \__xeCJK_save_CJK_class:n #1
684 {
685   \seq_gput_right:Nn \g__xeCJK_CJK_class_seq {#1}
686   \tl_const:cn
687   { \__xeCJK_CJK_class_tl:n { \use:c { \__xeCJK_class_csname:n {#1} } } }
688   {#1}
689   \__xeCJK_update_clear_toks:n {#1}
690 }
691 \clist_map_function:nN
692 { CJK , FullLeft , FullRight , CM , HangulJamo } \__xeCJK_save_CJK_class:n

```

5.6 字符输出规则

	Default	CJK	FullL	FullR	HalfL	HalfR	Normal	Bound	CM
Default		✓	✓	✓				✓	✓
CJK	✓	✓	✓	✓	✓	✓	✓	✓	
FullLeft	✓	✓	✓	✓	✓	✓	✓	✓	✓
FullRight	✓	✓	✓	✓	✓	✓	✓	✓	✓
HalfLeft		✓	✓	✓					✓
HalfRight		✓	✓	✓				✓	✓
NormalSpace		✓	✓	✓				✓	✓
Boundary	✓	✓	✓	✓	✓		✓		✓
CM	✓	✓	✓	✓	✓	✓	✓	✓	✓

在 CJK 类开始时, 设置 `\XeTeXdashbreakstate` 为零, 避免破折号之间的折行。

`\xeCJK_class_group_begin:`
`\xeCJK_class_group_end:`

```

693 \cs_new_protected:Npn \xeCJK_class_group_begin:
694 {
695   \c_group_begin_token
696   \bool_set_true:N \l__xeCJK_CJK_group_bool
697   \xeCJK_reset_space_factor:
698   \int_zero:N \tex_XeTeXdashbreakstate:D
699 }
700 \bool_new:N \l__xeCJK_CJK_group_bool
701 \cs_new_eq:NN \xeCJK_class_group_end: \c_group_end_token

```

CM 字符类与 CJK 字符类基本相同, 只是从 CJK 转移到 CM 时, 不加入任何内容。

```

702 \AtEndOfPackage
703 {
704   \seq_map_inline:Nn \g__xeCJK_class_seq
705   {
706     \str_if_eq:nnTF {#1} { CM }
707     { \xeCJK_copy_inter_class_toks:nmmm { CM } {#1} { CJK } { CJK } }
708     {
709       \xeCJK_copy_inter_class_toks:nmmm { CM } {#1} { CJK } {#1}
710       \str_if_eq:nnF {#1} { CJK }
711       { \xeCJK_copy_inter_class_toks:nmmm {#1} { CM } {#1} { CJK } }
712     }
713   }
714 }

```

HangulJamo 字符类与 CJK 字符类基本相同, 只是 HangulJamo 类之间不加入任何内容。

```

715 \AtEndOfPackage
716 {
717   \seq_map_inline:Nn \g__xeCJK_class_seq
718   {
719     \str_if_eq:nnF {#1} { HangulJamo }
720     {
721       \xeCJK_copy_inter_class_toks:nmmm { HangulJamo } {#1} { CJK } {#1}
722       \xeCJK_copy_inter_class_toks:nmmm {#1} { HangulJamo } {#1} { CJK }
723     }
724   }
725 }

726 \clist_map_inline:nn { Default , HalfLeft , HalfRight , NormalSpace }
727 {
728   \xeCJK_inter_class_toks:nnn {#1} { CJK }
729   {
730     \xeCJK_class_group_begin:
731     \xeCJK_select_font:
732     \xeCJK_clear_inter_class_toks:nn {#1} { CJK }
733     \xeCJK_clear_Boundary_and_CJK_toks:
734     \xeCJK_fallback_symbol:NN

```

```

735     \CJKsymbol
736   }
737   \xeCJK_inter_class_toks:nnn { CJK } {#1} { \xeCJK_class_group_end: }
738 }

739 \clist_map_inline:nn { Default , HalfLeft }
740 {
741   \xeCJK_inter_class_toks:nnn { Boundary } {#1}
742   { \xeCJK_Boundary_and_Default: }
743   \xeCJK_app_inter_class_toks:nnn { CJK } {#1}
744   { \CJKecglue }
745 }

\xeCJK_Boundary_and_Default: 746 \cs_new_protected:Npn \xeCJK_Boundary_and_Default:
747 { \xeCJK_check_for_ecglue: }
748 \cs_new_protected:Npn \__xeCJK_check_for_xecglue:
749 {
750   \__xeCJK_if_last_glue:TF
751   { \__xeCJK_replace_space: }
752   { \__xeCJK_check_for_ecglue: }
753 }
754 \cs_new_protected:Npn \__xeCJK_check_for_ecglue:
755 {
756   \xeCJK_if_last_node:nTF { CJK }
757   { \use_i:nn }
758   { \xeCJK_if_last_node:nTF { CJK-widow } }
759   { \xeCJK_remove_node: \skip_horizontal:N \l__xeCJK_ecglue_skip }
760   {
761     \xeCJK_if_last_node:nTF { CJK-space }
762     { \xeCJK_remove_node: \skip_horizontal:N \l__xeCJK_ecglue_skip }
763     { }
764   }
765 }
766 \cs_new_eq:NN \xeCJK_check_for_ecglue: \__xeCJK_check_for_ecglue:

```

`__xeCJK_replace_space:` 将空格替换为 `\CJKecglue`。注意由 `\leaders` 等产生的 glue，并不能正确地还回去。好在 $\text{\LaTeX}2_{\epsilon}$ 中常用的 `\hrulefill` 和 `\dotfill` 定义末尾都有 `\kern\z@` 保护。

```

767 \cs_new_protected:Npn \__xeCJK_replace_space:
768 {
769   \skip_set_eq:NN \l__xeCJK_last_skip \tex_lastskip:D
770   \tex_unskip:D
771   \xeCJK_if_last_node:nTF { CJK-space }
772   { \xeCJK_remove_node: \skip_horizontal:N \l__xeCJK_ecglue_skip }
773   {
774     \xeCJK_if_last_node:nTF { CJK }
775     {
776       \skip_if_eq:nnTF
777       { \l__xeCJK_last_skip }
778       { \c_xeCJK_space_skip_tl }
779       { \xeCJK_remove_node: \skip_horizontal:N \l__xeCJK_ecglue_skip }
780       { \skip_horizontal:N \l__xeCJK_last_skip }
781     }
782     { \skip_horizontal:N \l__xeCJK_last_skip }
783   }
784 }
785 \skip_new:N \l__xeCJK_last_skip

786 \clist_map_inline:nn { Default , HalfRight }
787 {
788   \xeCJK_inter_class_toks:nnn {#1} { Boundary }
789   {
790     \int_gset_eq:NN \g__xeCJK_space_factor_int \tex_spacefactor:D
791     \peek_meaning_remove:NTF \tex_italiccorrection:D
792     {
793       \tex_italiccorrection:D
794       { \xeCJK_make_node:n { default } }
795     }

```

```

796     {
797         \token_if_space:NTF \l_peek_token
798         { \xeCJK_make_space_node: }
799         { { \xeCJK_make_node:n { default } } }
800     }
801 }
802 \xeCJK_pre_inter_class_toks:nnn {#1} { CJK } { \CJKecglue }
803 }

804 \xeCJK_inter_class_toks:nnn { Boundary } { NormalSpace }
805 { \xeCJK_Boundary_and_NormalSp: }

\xeCJK_Boundary_and_NormalSp: 806 \cs_new_protected:Npn \xeCJK_Boundary_and_NormalSp:
807 { \xeCJK_check_for_ecglue_normalsp: }
808 \cs_new_protected:Npn \__xeCJK_check_for_xecglue_normalsp:
809 {
810     \__xeCJK_if_last_glue:TF
811     { \__xeCJK_replace_space: }
812     { \__xeCJK_check_for_ecglue_normalsp: }
813 }
814 \cs_new_protected:Npn \__xeCJK_check_for_ecglue_normalsp:
815 {
816     \xeCJK_if_last_node:nT { CJK-space }
817     { \xeCJK_remove_node: \xeCJK_space_or_xecglue: }
818 }
819 \cs_new_eq:NN \xeCJK_check_for_ecglue_normalsp:
820     \__xeCJK_check_for_ecglue_normalsp:

821 \xeCJK_inter_class_toks:nnn { NormalSpace } { Boundary }
822 {
823     \int_gset_eq:NN \g__xeCJK_space_factor_int \tex_spacefactor:D
824     \peek_meaning_remove:NTF \tex_italiccorrection:D
825     {
826         \tex_italiccorrection:D
827         { \xeCJK_make_node:n { normalspace } }
828     }
829     {
830         \token_if_space:NTF \l_peek_token
831         { \xeCJK_make_space_node: }
832         { { \xeCJK_make_node:n { normalspace } } }
833     }
834 }

835 \xeCJK_inter_class_toks:nnn { Boundary } { CJK }
836 {
837     \xeCJK_check_for_glue:
838     \xeCJK_class_group_begin:
839     \xeCJK_clear_Boundary_and_CJK_toks:
840     \xeCJK_select_font:
841     \xeCJK_fallback_symbol:NN
842     \CJKsymbol
843 }

\xeCJK_check_for_glue: 844 \cs_new_protected:Npn \xeCJK_check_for_glue:
845 {
846     \__xeCJK_if_last_kern:TF
847     { \__xeCJK_check_for_glue_auxi: }
848     {
849         \__xeCJK_if_last_glue:TF
850         { \__xeCJK_check_for_glue_skip: }
851         {
852             \__xeCJK_if_last_whatshit:TF
853             { \__xeCJK_recover_glue_whatshit: }
854             {
855                 \__xeCJK_if_last_math:TF
856                 { \xeCJK_remove_node: \skip_horizontal:N \l__xeCJK_ecglue_skip }
857                 { \__xeCJK_check_for_glue_auxii: }
858             }
859         }

```

```

860     }
861 }

```

xeCJKfntef 的命令内部使用 ulem 的 \hbox 排版 CJK 文字, 导致 XeTeX 的 interchar class 在命令结束后不是 CJK 而是 Boundary。源码空格因此产生 inter-word glue (而非被吃掉), 叠在 CJK kern pair 标记上方。本函数通过 \unskip 探测 glue 下方的 kern pair 来解决。kern pair 的探测由 \g_@@_glue_check_pending_bool 门控: 仅在 xeCJKfntef 组结束或颜色恢复 (\reset@color) 之后才启用, 以避免在 space=true 模式下误替换用户输入的空格。该标志分别由 @@_ulem_group_end:n 和 \reset@color 补丁设置。

对于 \mbox、\colorbox 等产生 hbox 的命令, hbox 节点隔在 kern pair 和 glue 之间, \lastkern 无法穿透。此时检查 \unskip 后的最后节点是否为 hlist, 若是则通过全局 \g_@@_last_node_tl 判断前方内容类型。该变量由 \xeCJK_make_node:n 设置, 能穿透 hbox/group 边界保持 CJK 标记。hlist 分支本身不设门控: 其安全性依赖于 \g_@@_last_node_tl 仅在 CJK 输出路径中被设置, 非 CJK 上下文中该变量为空, @@_check_for_glue_skip_hlist_aux: 会直接走 fallback。对于 \colorbox 等带颜色的 hbox 命令, \reset@color 在 hbox 之后插入 color pop whatsit, 遮蔽了 hlist 检测。此时通过专用标志 \g_@@_reset_color_pending_bool 门控 whatsit 回退路径: 仅当 \reset@color 已标记 hlist 上下文时才启用, 避免 \write 等无关 whatsit 的干扰。

listings 等宏包在 CJK 字符间放置 fil 级别的 glue 用于列对齐, 通过 \skip_if_finite:nTF 将其排除。标题格式中的 \quad 等有意放置的间距没有 shrink 分量, 通过检测 \tex_glueshrink:D 来排除。

本函数仅在 @@_if_last_glue:TF 的 true 分支中调用, 因此入口处 \tex_lastskip:D 保证为非零 glue。

```

862 \cs_new_protected:Npn \__xeCJK_check_for_glue_skip:
863 {
864   \skip_set_eq:NN \l__xeCJK_last_skip \tex_lastskip:D
865   \skip_if_finite:nTF { \l__xeCJK_last_skip }
866   {
867     \dim_compare:nNnTF
868     { \tex_glueshrink:D \l__xeCJK_last_skip } > \c_zero_dim
869     {
870       \tex_unskip:D
871       \__xeCJK_if_last_kern:TF
872       { \__xeCJK_check_for_glue_skip_kern: }
873       { \__xeCJK_check_for_glue_skip_nonkern: }
874     }
875     { \__xeCJK_check_for_glue_skip_consume_and_fallback: }
876   }
877   { \__xeCJK_check_for_glue_skip_consume_and_fallback: }
878 }
879 \cs_new_protected:Npn \__xeCJK_check_for_glue_skip_kern:
880 {
881   \bool_if:NTF \g__xeCJK_glue_check_pending_bool
882   {
883     \bool_gset_false:N \g__xeCJK_glue_check_pending_bool
884     \dim_case:nnF { \tex_lastkern:D }
885     {
886       { \__xeCJK_node:n { CJK } }
887       { \xeCJK_remove_node: \CJKglue }
888       { \__xeCJK_node:n { CJK-space } }
889       { \xeCJK_remove_node: \__xeCJK_ccglue_or_space: }
890       { \__xeCJK_node:n { CJK-widow } }
891       { \xeCJK_remove_node: \xeCJK_widow_penalty: \CJKglue }
892     }
893     { \__xeCJK_check_for_glue_skip_restore: }
894   }
895   { \__xeCJK_check_for_glue_skip_restore: }
896 }
897 \cs_new_protected:Npn \__xeCJK_check_for_glue_skip_nonkern:

```

```

898 {
899   \bool_if:NT \g__xeCJK_glue_check_pending_bool
900   { \bool_gset_false:N \g__xeCJK_glue_check_pending_bool }
901   \__xeCJK_if_last_hlist:TF
902   { \__xeCJK_check_for_glue_skip_hlist_aux: }
903   {
904     \bool_if:NTF \g__xeCJK_reset_color_pending_bool
905     {
906       \bool_gset_false:N \g__xeCJK_reset_color_pending_bool
907       \__xeCJK_if_last_whatsit:TF
908       { \__xeCJK_check_for_glue_skip_hlist_aux: }
909       { \__xeCJK_check_for_glue_skip_restore: }
910     }
911     { \__xeCJK_check_for_glue_skip_restore: }
912   }
913 }
914 \cs_new_protected:Npn \__xeCJK_check_for_glue_skip_restore:
915 {
916   \skip_horizontal:N \l__xeCJK_last_skip
917   \__xeCJK_check_for_glue_auxii:
918 }
919 \cs_new_protected:Npn \__xeCJK_check_for_glue_skip_consume_and_fallback:
920 {
921   \bool_if:NT \g__xeCJK_glue_check_pending_bool
922   { \bool_gset_false:N \g__xeCJK_glue_check_pending_bool }
923   \__xeCJK_check_for_glue_auxii:
924 }
925 \cs_new_protected:Npn \__xeCJK_check_for_glue_skip_hlist_aux:
926 {
927   \tl_if_eq:NnTF \g__xeCJK_last_node_tl { CJK }
928   { \tl_gclear:N \g__xeCJK_last_node_tl \CJKglue }
929   {
930     \tl_if_eq:NnTF \g__xeCJK_last_node_tl { CJK-space }
931     { \tl_gclear:N \g__xeCJK_last_node_tl \__xeCJK_ccglue_or_space: }
932     {
933       \tl_if_eq:NnTF \g__xeCJK_last_node_tl { CJK-widow }
934       {
935         \tl_gclear:N \g__xeCJK_last_node_tl
936         \xeCJK_widow_penalty: \CJKglue
937       }
938       {
939         \skip_horizontal:N \l__xeCJK_last_skip
940         \__xeCJK_check_for_glue_auxii:
941       }
942     }
943   }
944 }
945 \cs_new_protected:Npn \__xeCJK_check_for_glue_auxi:
946 {
947   \dim_case:nn { \tex_lastkern:D }
948   {
949     { \__xeCJK_node:n { CJK } }
950     { \xeCJK_remove_node: \CJKglue }
951     { \__xeCJK_node:n { CJK-space } }
952     { \xeCJK_remove_node: \__xeCJK_ccglue_or_space: }
953     { \__xeCJK_node:n { CJK-widow } }
954     { \xeCJK_remove_node: \xeCJK_widow_penalty: \CJKglue }
955     { \__xeCJK_node:n { default } }
956     { \xeCJK_remove_node: \skip_horizontal:N \l__xeCJK_ecglue_skip }
957   }
958 }
959 \cs_new_protected:Npn \__xeCJK_check_for_glue_auxii:
960 {
961   \xeCJK_if_last_punct:TF
962   { \__xeCJK_check_for_glue_auxiii: }
963   { \xeCJK_check_for_xglue: }
964 }

```

```

965 \cs_new_protected:Npn \__xeCJK_check_for_glue_auxiii:
966 {
967   \bool_if:NT \l__xeCJK_last_penalty_bool
968     { \tex_penalty:D \l__xeCJK_last_penalty_int }
969   \skip_horizontal:N \l__xeCJK_last_skip
970   \tl_if_eq:NNF \l__xeCJK_aligni_tl \c__xeCJK_left_tl { \CJKglue }
971 }
972 \cs_new_eq:NN \xeCJK_check_for_xglue: \prg_do_nothing:
973 \cs_new_protected:Npn \__xeCJK_check_for_xglue:
974 {
975   \__xeCJK_if_last_glue:T
976   {
977     \skip_set_eq:NN \l__xeCJK_last_skip \tex_lastskip:D
978     \tex_unskip:D
979     \xeCJK_if_last_node:nTF { CJK-space }
980     { \xeCJK_remove_node: \__xeCJK_ccglue_or_space: }
981     {
982       \xeCJK_if_last_node:nTF { default-space }
983       { \xeCJK_remove_node: \skip_horizontal:N \l__xeCJK_ecglue_skip }
984       { \__xeCJK_check_for_xglue_aux: }
985     }
986   }
987 }
988 \cs_new_protected:Npn \__xeCJK_check_for_xglue_aux:
989 {
990   \skip_if_eq:nnTF
991     { \l__xeCJK_last_skip }
992     { \c__xeCJK_space_skip_tl }
993     {
994       \xeCJK_if_last_node:nTF { CJK }
995       { \xeCJK_remove_node: \__xeCJK_ccglue_or_space: }
996       {
997         \xeCJK_if_last_node:nTF { default }
998         { \xeCJK_remove_node: \skip_horizontal:N \l__xeCJK_ecglue_skip }
999         {
1000           \__xeCJK_if_last_math:TF
1001             { \skip_horizontal:N \l__xeCJK_ecglue_skip }
1002             { \skip_horizontal:N \l__xeCJK_last_skip }
1003         }
1004       }
1005     }
1006     { \skip_horizontal:N \l__xeCJK_last_skip }
1007 }
1008 \cs_new_protected:Npn \__xeCJK_ccglue_or_space:
1009 { \CJKglue }

```

__xeCJK_if_last_none:TF 一些 ϵ -TeX 结点判定函数。

```

\__xeCJK_if_last_hlist:TF
\__xeCJK_if_last_math:TF
\__xeCJK_if_last_glue:TF
\__xeCJK_if_last_kern:TF
\__xeCJK_if_last_penalty:TF
1010 \group_begin:
1011 \cs_set:Npn \__xeCJK_tmp:nn #1
1012 {
1013   \exp_args:Ncc \__xeCJK_tmp_aux:NNn
1014     { __xeCJK_if_last_#1 : }
1015     { c__xeCJK_#1_node }
1016 }
1017 \cs_set:Npn \__xeCJK_tmp_aux:NNn #1#2#3
1018 {
1019   \int_const:Nn #2 {#3}
1020   \prg_new_conditional:Npnn #1 { T , F , TF }
1021   {
1022     \if_int_compare:w \tex_lastnodetype:D = #2
1023     \prg_return_true: \else: \prg_return_false: \fi:
1024   }
1025 }
1026 \__xeCJK_tmp:nn { none } { -1 }
1027 \__xeCJK_tmp:nn { hlist } { 1 }
1028 \__xeCJK_tmp:nn { math } { 10 }

```

```

1029 \_xeCJK_tmp:nn { glue } { 11 }
1030 \_xeCJK_tmp:nn { kern } { 12 }
1031 \_xeCJK_tmp:nn { penalty } { 13 }
1032 \_xeCJK_tmp:nn { whatsit } { 9 }
1033 \group_end:

\XeCJK_if_last_node_p:n 1034 \prg_new_conditional:Npnn \xeCJK_if_last_node:n #1 { p , T , F , TF }
\XeCJK_if_last_node:nTF 1035 {
1036   \if_dim:w
1037     \cs_if_exist_use:cTF { c__xeCJK_#1_node_dim }
1038     { = \tex_lastkern:D }
1039     { \use:c { c__xeCJK_#1_node_skip } = \tex_lastskip:D }
1040     \prg_return_true: \else: \prg_return_false: \fi:
1041   }

```

\XeCJK_declare_node:n 用于判断插入的各种 kern 和 glue。

```

\XeCJK_make_node:n 1042 \cs_new_protected:Npn \xeCJK_declare_node:n #1
1043 {
1044   \int_gincr:N \g__xeCJK_node_int
1045   \dim_if_exist:cTF { c__xeCJK_#1_node_dim }
1046   { \dim_gset:cn } { \dim_const:cn }
1047   { c__xeCJK_#1_node_dim } { \g__xeCJK_node_int sp }
1048 }
1049 \cs_new_protected:Npn \xeCJK_declare_glue_node:n #1
1050 {
1051   \int_gincr:N \g__xeCJK_node_int
1052   \skip_if_exist:cTF { c__xeCJK_#1_node_skip }
1053   { \skip_gset:cn } { \skip_const:cn }
1054   { c__xeCJK_#1_node_skip } { \g__xeCJK_node_int sp }
1055 }
1056 \int_new:N \g__xeCJK_node_int
1057 \int_gset:Nn \g__xeCJK_node_int { 10 }
1058 \tl_new:N \g__xeCJK_last_node_tl
1059 \cs_new_protected:Npn \xeCJK_make_node:n #1
1060 {
1061   \exp_args:Nc \_xeCJK_make_node:N { c__xeCJK_#1_node_dim }
1062   \tl_gset:Nn \g__xeCJK_last_node_tl {#1}
1063 }
1064 \cs_new:Npn \_xeCJK_node:n #1
1065 { \use:c { c__xeCJK_#1_node_dim } }
1066 \cs_new:Npn \_xeCJK_glue_node:n #1
1067 { \use:c { c__xeCJK_#1_node_skip } }
1068 \cs_new_protected:Npn \_xeCJK_make_node:N #1
1069 {
1070   \tex_kern:D - #1
1071   \tex_kern:D #1
1072 }
1073 \cs_new_protected:Npn \xeCJK_remove_node:
1074 {
1075   \_xeCJK_if_last_kern:TF
1076   { \tex_unkern:D \tex_unkern:D }
1077   {
1078     \_xeCJK_if_last_glue:T
1079     { \tex_unskip:D \tex_unskip:D }
1080   }
1081   \tl_gclear:N \g__xeCJK_last_node_tl
1082 }
1083 \cs_new_protected:Npn \_xeCJK_recover_ecglue_whatsit:
1084 {
1085   \_xeCJK_if_last_whatsit:T
1086   {
1087     \tl_if_eq:NnTF \g__xeCJK_last_node_tl { CJK }
1088     { \xeCJK_remove_node: \skip_horizontal:N \l__xeCJK_ecglue_skip }
1089     {
1090       \tl_if_eq:NnTF \g__xeCJK_last_node_tl { CJK-widow }
1091       { \xeCJK_remove_node: \skip_horizontal:N \l__xeCJK_ecglue_skip }

```

```

1092         {
1093             \tl_if_eq:NnT \g__xeCJK_last_node_tl { CJK-space }
1094             { \xeCJK_remove_node: \skip_horizontal:N \l__xeCJK_ecglue_skip }
1095         }
1096     }
1097 }
1098 }
1099 \cs_new_protected:Npn \__xeCJK_recover_glue_whatsit:
1100 {
1101     \__xeCJK_if_last_whatsit:T
1102     {
1103         \tl_if_eq:NnTF \g__xeCJK_last_node_tl { CJK }
1104         { \xeCJK_remove_node: \CJKglue }
1105         {
1106             \tl_if_eq:NnTF \g__xeCJK_last_node_tl { CJK-space }
1107             { \xeCJK_remove_node: \__xeCJK_ccglue_or_space: }
1108             {
1109                 \tl_if_eq:NnTF \g__xeCJK_last_node_tl { CJK-widow }
1110                 { \xeCJK_remove_node: \xeCJK_widow_penalty: \CJKglue }
1111                 {
1112                     \tl_if_eq:NnT \g__xeCJK_last_node_tl { default }
1113                     { \xeCJK_remove_node: \skip_horizontal:N \l__xeCJK_ecglue_skip }
1114                 }
1115             }
1116         }
1117     }
1118 }
1119 \xeCJK_declare_node:n { CJK }
1120 \xeCJK_declare_node:n { CJK-space }
1121 \xeCJK_declare_node:n { default }
1122 \xeCJK_declare_node:n { CJK-widow }
1123 \xeCJK_declare_node:n { normalspace }
1124 \xeCJK_declare_glue_node:n { default-space }

1125 \bool_new:N \g__xeCJK_glue_check_pending_bool
1126 \bool_new:N \g__xeCJK_reset_color_pending_bool

```

`\xeCJK_make_space_node:` 用于判断插入空格之前的 `node`, 默认为空, 只有用户设置了 `xCJKecglue` 选项才有意义。需要使用 `glue` 来标记, 使用 `kern` 会影响 `character protrusion` 功能。

```

1127 \cs_new_eq:NN \xeCJK_make_space_node: \prg_do_nothing:
1128 \cs_new_protected:Npe \__xeCJK_make_space_node:
1129 {
1130     \tex_hskip:D - \__xeCJK_glue_node:n { default-space }
1131     \tex_hskip:D \__xeCJK_glue_node:n { default-space }
1132 }

```

CJKglue CJK 文字之间插入的 `glue`。

```

1133 \keys_define:nn { xeCJK / options }
1134 {
1135     CJKglue .code:n =
1136     {
1137         \cs_set_protected:Npn \CJKglue {#1}
1138         \xeCJK_glue_to_skip:nN {#1} \l__xeCJK_ccglue_skip
1139     }
1140 }
1141 \skip_new:N \l__xeCJK_ccglue_skip

```

CJKecglue CJK 与西文和数学行内数学公式之间自动添加的空白。
xCJKecglue

```

1142 \keys_define:nn { xeCJK / options }
1143 {
1144     CJKecglue .code:n =
1145     {
1146         \cs_set_protected:Npn \CJKecglue {#1}

```

```

1147     \xeCJK_glue_to_skip:nN {#1} \l__xeCJK_ecglue_skip
1148   } ,
1149   xCJKecglue .choice: ,
1150   xCJKecglue / true .code:n =
1151   {
1152     \bool_set_true:N \l__xeCJK_xecglue_bool
1153     \cs_set_eq:NN \xeCJK_space_or_xecglue: \CJKecglue
1154     \cs_set_eq:NN \xeCJK_make_space_node: \__xeCJK_make_space_node:
1155     \cs_set_eq:NN \xeCJK_check_for_xglue: \__xeCJK_check_for_xglue:
1156     \cs_set_eq:NN \xeCJK_check_for_ecglue: \__xeCJK_check_for_xecglue:
1157     \cs_set_eq:NN
1158       \xeCJK_check_for_ecglue_normalsp:
1159       \__xeCJK_check_for_xecglue_normalsp:
1160   } ,
1161   xCJKecglue / false .code:n =
1162   {
1163     \bool_set_false:N \l__xeCJK_xecglue_bool
1164     \cs_set_eq:NN \xeCJK_space_or_xecglue: \xeCJK_space_glue:
1165     \xeCJK_cs_clear:N \xeCJK_make_space_node:
1166     \xeCJK_cs_clear:N \xeCJK_check_for_xglue:
1167     \cs_set_eq:NN \xeCJK_check_for_ecglue: \__xeCJK_check_for_ecglue:
1168     \cs_set_eq:NN
1169       \xeCJK_check_for_ecglue_normalsp:
1170       \__xeCJK_check_for_ecglue_normalsp:
1171   } ,
1172   xCJKecglue / unknown .code:n =
1173   {
1174     \bool_set_true:N \l__xeCJK_xecglue_bool
1175     \cs_set_protected:Npn \CJKecglue {#1}
1176     \xeCJK_glue_to_skip:nN {#1} \l__xeCJK_ecglue_skip
1177     \cs_set_eq:NN \xeCJK_space_or_xecglue: \CJKecglue
1178     \cs_set_eq:NN \xeCJK_make_space_node: \__xeCJK_make_space_node:
1179     \cs_set_eq:NN \xeCJK_check_for_xglue: \__xeCJK_check_for_xglue:
1180     \cs_set_eq:NN \xeCJK_check_for_ecglue: \__xeCJK_check_for_xecglue:
1181     \cs_set_eq:NN
1182       \xeCJK_check_for_ecglue_normalsp:
1183       \__xeCJK_check_for_xecglue_normalsp:
1184   } ,
1185   xCJKecglue .default:n = { true }
1186 }
1187 \cs_new_eq:NN \xeCJK_space_glue: \c_space_tl
1188 \skip_new:N \l__xeCJK_ecglue_skip
1189 \bool_new:N \l__xeCJK_xecglue_bool

```

CJKspace 是否保留 CJK 文字间的空白, 默认不保留。

```

1190 \keys_define:nn { xeCJK / options }
1191 {
1192   CJKspace .choice: ,
1193   CJKspace / true .code:n =
1194   {
1195     \bool_set_true:N \l__xeCJK_reserve_space_bool
1196     \cs_set_protected:Npn \__xeCJK_ccglue_or_space:
1197       { \xeCJK_space_glue: }
1198   } ,
1199   CJKspace / false .code:n =
1200   {
1201     \bool_set_false:N \l__xeCJK_reserve_space_bool
1202     \cs_set_protected:Npn \__xeCJK_ccglue_or_space:
1203       { \CJKglue }
1204   } ,
1205   CJKspace .default:n = { true } ,
1206   space .meta:n = { CJKspace = true } ,
1207   nospace .meta:n = { CJKspace = false }
1208 }
1209 \bool_new:N \l__xeCJK_reserve_space_bool
1210 \xeCJK_inter_class_toks:nnn { CJK } { Boundary } { \xeCJK_CJK_and_Boundary:w }

```

`\xeCJK_CJK_and_Boundary:w` 当边界是 `\relax` 的时候,它可能是由 `\csname ... \endcsname` 的形式产生的,这样就可能出现问题¹⁵。原来是都在未定义控制序列前都加上 `\exp_not:N`,现在是采用分组结束后手工恢复的方式。

```

1211 \cs_new_protected:Npn \xeCJK_CJK_and_Boundary:w
1212 {
1213   \xeCJK_peek_catcode_ignore_spaces:NTF \c_math_toggle_token
1214   {
1215     \bool_if:NTF \l__xeCJK_peek_ignore_spaces_bool
1216     { \xeCJK_class_group_end: \xeCJK_space_or_xecglue: }
1217     { \xeCJK_class_group_end: \CJKeckglue }
1218   }
1219   {
1220     \group_align_safe_begin:
1221     \bool_if:NTF \l__xeCJK_peek_ignore_spaces_bool
1222     {
1223       \token_if_macro:NTF \l_peek_token
1224       { \__xeCJK_boundary_reserve_space: }
1225       { \__xeCJK_boundary_group_end:n { CJK-space } }
1226     }
1227     {
1228       \token_if_eq_meaning:NNTF \l_peek_token \scan_stop:
1229       { \__xeCJK_CJK_and_Boundary_relax:N }
1230       {
1231         \token_if_group_end:NTF \l_peek_token
1232         {
1233           \bool_gset_true:N \g__xeCJK_glue_check_pending_bool
1234           \__xeCJK_boundary_group_end:n { CJK }
1235         }
1236         { \__xeCJK_boundary_group_end:n { CJK } }
1237       }
1238     }
1239   }
1240 }
1241 \cs_new_protected:Npn \__xeCJK_boundary_reserve_space:
1242 { \__xeCJK_boundary_group_end:n { CJK-space } }
1243 \cs_new_protected:Npn \__xeCJK_CJK_and_Boundary_relax:N #1
1244 {
1245   \__xeCJK_boundary_group_end:n { CJK }
1246   \token_if_eq_meaning:NNTF #1 \scan_stop:
1247   {#1} { \cs_set_eq:NN #1 \scan_stop: #1 }
1248 }
1249 \cs_new_protected:Npn \__xeCJK_boundary_group_end:n #1
1250 {
1251   \group_align_safe_end:
1252   \xeCJK_class_group_end:
1253   \xeCJK_glue_to_skip:nN { \CJKeckglue } \l__xeCJK_ecglue_skip
1254   { \xeCJK_make_node:n {#1} }
1255 }

```

```

\xeCJK_ignore_spaces:w 1256 \cs_new_protected:Npn \xeCJK_ignore_spaces:w
1257 {
1258   \xeCJK_peek_catcode_ignore_spaces:NTF \c_math_toggle_token
1259   {
1260     \bool_if:NTF \l__xeCJK_peek_ignore_spaces_bool
1261     { \xeCJK_space_or_xecglue: } { \CJKeckglue }
1262   }
1263   {
1264     \bool_if:NT \l__xeCJK_peek_ignore_spaces_bool
1265     {
1266       \dim_case:nn { \tex_lastkern:D }
1267       {
1268         { \__xeCJK_node:n { CJK } }
1269         { \xeCJK_remove_node: \xeCJK_make_node:n { CJK-space } }
1270         { \__xeCJK_node:n { default } }

```

¹⁵参见 <http://bbs.ctex.org/forum.php?mod=viewthread&tid=71563>。

```

1271         { \xeCJK_remove_node: \xeCJK_make_space_node: }
1272     }
1273     \group_align_safe_begin:
1274     \token_if_macro:NTF \l_peek_token
1275     { \__xeCJK_reserve_space_aux: }
1276     { \group_align_safe_end: }
1277 }
1278 }
1279 }
1280 \cs_new_protected:Npn \__xeCJK_reserve_space_aux:
1281 {
1282     \group_align_safe_end:
1283     \xeCJK_space_or_xecglue:
1284 }
1285 \xeCJK_inter_class_toks:nmn { CJK } { CJK }
1286 { \xeCJK_CJK_and_CJK:N }
\xeCJK_CJK_and_CJK:N 1287 \cs_new_protected:Npn \xeCJK_CJK_and_CJK:N
1288 {
1289     \CJKglue
1290     \xeCJK_fallback_symbol:NN
1291     \CJKsymbol
1292 }
1293 \xeCJK_inter_class_toks:nmn { FullLeft } { CJK }
1294 {
1295     \xeCJK_FullLeft_and_CJK:
1296     \xeCJK_fallback_symbol:NN
1297     \CJKsymbol
1298 }
1299 \xeCJK_inter_class_toks:nmn { FullRight } { CJK }
1300 {
1301     \xeCJK_FullRight_and_CJK:
1302     \xeCJK_fallback_symbol:NN
1303     \CJKsymbol
1304 }
1305 \seq_map_inline:Nn \g__xeCJK_non_CJK_class_seq
1306 {
1307     \clist_map_inline:nn { FullLeft , FullRight }
1308     {
1309         \xeCJK_inter_class_toks:nne {#1} {##1}
1310         { \exp_not:c { xeCJK_Default_and_##1:nN } {#1} }
1311         \xeCJK_inter_class_toks:nne {##1} {#1}
1312         { \exp_not:c { xeCJK_##1_and_Default: } }
1313     }
1314 }
\l__xeCJK_halfright_nobreak_bool 1315 \bool_new:N \l__xeCJK_halfright_nobreak_bool

```

在 CJK → HalfRight 和 FullRight → HalfRight 的过渡中,条件性地插入禁则 penalty。

```

1316 \xeCJK_app_inter_class_toks:nmn { CJK } { HalfRight }
1317 {
1318     \bool_if:NT \l__xeCJK_halfright_nobreak_bool
1319     { \xeCJK_no_break: }
1320 }

```

FullRight → HalfRight 需要将 penalty 插入 punct_glue 之前,因此覆写而非追加。

```

1321 \xeCJK_inter_class_toks:nmn { FullRight } { HalfRight }
1322 {
1323     \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
1324     \xeCJK_class_group_end:
1325     \bool_if:NT \l__xeCJK_halfright_nobreak_bool
1326     { \xeCJK_no_break: }
1327     \__xeCJK_punct_glue:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
1328 }

```

```

1329 \xeCJK_inter_class_toks:nnn { Boundary } { FullLeft }
1330 { \xeCJK_Boundary_and_FullLeft:N }
1331 \xeCJK_inter_class_toks:nnn { Boundary } { FullRight }
1332 { \xeCJK_Boundary_and_FullRight:N }

1333 \xeCJK_inter_class_toks:nnn { FullLeft } { Boundary }
1334 { \xeCJK_FullLeft_and_Boundary: }
1335 \xeCJK_inter_class_toks:nnn { FullRight } { Boundary }
1336 { \xeCJK_FullRight_and_Boundary: }

\xeCJK_FullLeft_and_Boundary: 1337 \cs_new_protected:Npn \xeCJK_FullLeft_and_Boundary:
1338 {
1339   \__xeCJK_punct_if_middle:NTF \g__xeCJK_last_punct_tl
1340   {
1341     \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
1342     \xeCJK_class_group_end:
1343     \exp_after:wN \xeCJK_punct_node:N \g__xeCJK_last_punct_tl
1344     \xeCJK_no_break:
1345     \__xeCJK_punct_glue:NN \c__xeCJK_left_tl \g__xeCJK_last_punct_tl
1346     \__xeCJK_punct_boundary_guard:
1347   }
1348   {
1349     \xeCJK_class_group_end:
1350     \exp_after:wN \xeCJK_punct_node:N \g__xeCJK_last_punct_tl
1351     \__xeCJK_nobreak_zero_glue:
1352   }
1353   \tex_ignorespaces:D
1354 }

```

```

\xeCJK_FullRight_and_Boundary: 1355 \cs_new_protected:Npn \xeCJK_FullRight_and_Boundary:
1356 {
1357   \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
1358   \xeCJK_class_group_end:
1359   \exp_after:wN \xeCJK_punct_node:N \g__xeCJK_last_punct_tl
1360   \__xeCJK_punct_glue:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
1361   \__xeCJK_punct_boundary_guard:
1362   \tex_ignorespaces:D
1363 }

```

`__xeCJK_punct_boundary_guard:` 在 restricted horizontal mode (tabular 单元格、\hbox 等) 中，\结束行时通过 `\unskip` 移除最后一个 glue。这会吞掉标点的补偿 glue，导致单元格宽度错误。此函数在 inner mode 下于 glue 之后插入 `\penalty 0`，使最后节点不再是 glue，从而保护它。

在段落模式下，当 `experiment/punct-measure-fix` 选项启用时，记录 glue 的自然宽度，并通过 `para/end` 钩子插入等宽 `\kern` 恢复。使用 `\kern` 而非 `\hskip`：`\kern` 不会被 `\unskip` 移除（无需额外保护），也不构成合法断行点（避免 `CheckSingle` 等场景下产生空行）。

```

1364 \bool_new:N \g__xeCJK_par_guard_bool
1365 \dim_new:N \g__xeCJK_par_guard_dim
1366 \cs_new_protected:Npn \__xeCJK_punct_boundary_guard:
1367 {
1368   \mode_if_inner:TF
1369   { \tex_penalty:D \c_zero_int }
1370   {
1371     \bool_if:NT \g__xeCJK_punct_measure_fix_bool
1372     {
1373       \bool_gset_true:N \g__xeCJK_par_guard_bool
1374       \dim_gset:Nn \g__xeCJK_par_guard_dim { \tex_lastskip:D }
1375     }
1376   }
1377 }

```

当 Boundary 之后紧跟其他字符类时，说明标点不在段末，需要重置保护标志。

```

1378 \clist_map_inline:nn
1379 {
1380   Default , HalfLeft , HalfRight , NormalSpace ,
1381   CJK , FullLeft , FullRight , CM , HangulJamo

```

```

1382 }
1383 {
1384   \xeCJK_pre_inter_class_toks:nnn { Boundary } {#1}
1385   { \bool_gset_false:N \g__xeCJK_par_guard_bool }
1386 }

```

在 para/begin 中重置标志, 避免跨段落残留状态。在 para/end 中, 若标志为真则插入 \kern 补偿被 \unskip 移除的标点 glue 的自然宽度。

```

1387 \AddToHook { para/begin }
1388   { \bool_gset_false:N \g__xeCJK_par_guard_bool }
1389 \AddToHook { para/end }
1390   {
1391     \bool_if:NT \g__xeCJK_par_guard_bool
1392     {
1393       \tex_kern:D \g__xeCJK_par_guard_dim
1394       \bool_gset_false:N \g__xeCJK_par_guard_bool
1395     }
1396   }

```

\xeCJK_punct_node:N 保存标点的当前边界宽度和字符码, 通过插入 \kern 实现。

```

1397 \cs_new_protected:Npn \xeCJK_punct_node:N #1
1398   {
1399     \__xeCJK_punct_bound_unitization:NN #1 \l__xeCJK_tmp_dim
1400     \__xeCJK_make_node:N \l__xeCJK_tmp_dim
1401     \dim_set:Nn \l__xeCJK_tmp_dim { `#1 sp }
1402     \__xeCJK_make_node:N \l__xeCJK_tmp_dim
1403   }

```

__xeCJK_punct_bound_unitization:NN 我们不想出现过大的 \kern, 因此当边界大于1pt 时, 以 \c_max_dim 为标准对其进行“单位化”。

```

1404 \cs_new_protected:Npn \__xeCJK_punct_bound_unitization:NN #1#2
1405   {
1406     \dim_set:Nn #2
1407     {
1408       \dim_max:nn
1409       { \c_zero_dim }
1410       { \__xeCJK_use_punct_dim:nNN { bound } \c__xeCJK_right_tl #1 }
1411     }
1412     \dim_compare:nNnF {#2} < { 1pt }
1413     { \dim_set:Nn #2 { -1pt * \dim_ratio:nn {#2} { \c_max_dim } } }
1414   }

```

```

\xeCJK_punct_bound_kern:N 1415 \cs_new_protected:Npn \xeCJK_punct_bound_kern:N #1
__xeCJK_punct_bound_kern:NN 1416   {
1417     \exp_after:wN \__xeCJK_punct_bound_kern:NN
1418     \g__xeCJK_last_punct_tl #1
1419   }
1420 \cs_new_protected:Npn \__xeCJK_punct_bound_kern:NN #1#2
1421   {
1422     \xeCJK_get_punct_bounds:NN \l__xeCJK_aligni_tl #1
1423     \xeCJK_get_punct_kerning:NN #1 #2
1424     \__xeCJK_punct_bound_unitization:NN #1 \l__xeCJK_tmp_dim
1425     \skip_set:Nn \l__xeCJK_punct_kern_skip
1426     { \__xeCJK_use_dim_or_skip:nNN { bound_kern } #1 #2 }
1427     \dim_compare:nNnF \l__xeCJK_tmp_dim = \l__xeCJK_last_bound_dim
1428     { \__xeCJK_punct_bound_kern_ratio:NN #1 #2 }
1429     \bool_if:NTF \l__xeCJK_last_penalty_bool
1430     {
1431       \tex_penalty:D \l__xeCJK_last_penalty_int
1432       \skip_horizontal:N
1433     }
1434     { \__xeCJK_punct_bound_kern_aux:NNN #1 #2 }
1435     \l__xeCJK_punct_kern_skip
1436   }
1437 \skip_new:N \l__xeCJK_punct_kern_skip

```

_xeCJK_punct_bound_kern_ratio:NN 当标点前后的字体情况不一致时,按一定的比例进行压缩。

```

1438 \cs_new_protected:Npn \_xeCJK_punct_bound_kern_ratio:NN #1#2
1439 {
1440   \dim_set:Nn \l__xeCJK_bound_dim
1441     { \_xeCJK_use_punct_dim:nNN { bound_width } #1 #2 }
1442   \dim_compare:nNnT \l__xeCJK_bound_dim > \c_zero_dim
1443     {
1444       \dim_compare:nNnF \l__xeCJK_last_bound_dim > \c_zero_dim
1445         {
1446           \dim_set:Nn \l__xeCJK_last_bound_dim
1447             {
1448               - \l__xeCJK_last_bound_dim *
1449               \dim_ratio:nn { \c_max_dim } { 1pt }
1450             }
1451         }
1452       \_xeCJK_punct_bound_kern_ratio_aux:N #2
1453     }
1454 }
1455 \cs_new_protected:Npn \_xeCJK_punct_bound_kern_ratio_aux:N #1
1456 {
1457   \skip_set:Nn \l__xeCJK_punct_kern_skip
1458     {
1459       \l__xeCJK_punct_kern_skip *
1460       \dim_ratio:nn
1461         {
1462           \l__xeCJK_last_bound_dim
1463           + \_xeCJK_use_punct_dim:nNN { bound } \c__xeCJK_left_tl #1
1464         }
1465       { \l__xeCJK_bound_dim }
1466     }
1467 }

\_xeCJK_nobreak_hskip:N 1468 \cs_new_protected:Npn \_xeCJK_nobreak_hskip:N
\_xeCJK_nobreak_hskip:n 1469 { \xeCJK_no_break: \skip_horizontal:N }
\_xeCJK_punct_bound_kern:N 1470 \cs_new_protected:Npn \_xeCJK_nobreak_hskip:n
\_xeCJK_punct_bound_breakable_kern:N 1471 { \xeCJK_no_break: \skip_horizontal:n }
1472 \cs_new_eq:NN \_xeCJK_punct_bound_kern:N \_xeCJK_nobreak_hskip:N
1473 \cs_new_protected:Npn \_xeCJK_punct_bound_breakable_kern:N
1474 {
1475   \tl_if_eq:NNTF \l__xeCJK_aligni_tl \c__xeCJK_right_tl
1476     {
1477       \tl_if_eq:NNTF \l__xeCJK_alignii_tl \c__xeCJK_left_tl
1478         { \skip_horizontal:N }
1479         { \_xeCJK_nobreak_hskip:N }
1480     }
1481   { \_xeCJK_nobreak_hskip:N }
1482 }
1483 \cs_new_protected:Npn \_xeCJK_punct_bound_kern_aux:NNN #1#2
1484 {
1485   \str_if_eq:nnTF {#1} {#2}
1486     { \_xeCJK_nobreak_hskip:N }
1487     {
1488       \_xeCJK_punct_if_long:NTF #1
1489         { \skip_horizontal:N }
1490         {
1491           \_xeCJK_punct_if_long:NTF #2
1492             { \skip_horizontal:N }
1493             { \_xeCJK_punct_bound_kern:N }
1494         }
1495     }
1496 }

1497 \clist_map_inline:nn { CJK , FullLeft , FullRight }
1498 {
1499   \clist_map_inline:nn { FullLeft , FullRight }
1500     {
1501       \xeCJK_inter_class_toks:nne {#1} {##1}

```

```

1502     { \exp_not:c { xeCJK_#1_and_##1:N } }
1503   }
1504 }

```

_xeCJK_punct_bound_rule:NN 用于抹去标点符号的全部左/右空白。

```

1505 \cs_new_protected:Npn \_xeCJK_punct_bound_rule:NN #1#2
1506 {
1507   \tex_vrule:D
1508   width - \_xeCJK_use_punct_dim:nNN { bound } #1 #2 ~
1509   depth \c_zero_dim
1510   height \c_zero_dim \scan_stop:
1511 }

```

_xeCJK_punct_rule:NN 用于减少标点符号的左/右空白。

```

1512 \cs_new_protected:Npn \_xeCJK_punct_rule:NN #1#2
1513 {
1514   \tex_vrule:D
1515   width \_xeCJK_use_punct_dim:nNN { rule } #1 #2 ~
1516   depth \c_zero_dim
1517   height \c_zero_dim \scan_stop:
1518 }

```

_xeCJK_punct_glue:NN 根据所选的标点处理方式在标点符号左/右增加的空白。

```

1519 \cs_new_protected:Npn \_xeCJK_punct_glue:NN #1#2
1520 { \_xeCJK_punct_hskip:n { \_xeCJK_use_dim_or_skip:nNN { glue } #1 #2 } }
1521 \cs_new_eq:NN \_xeCJK_punct_hskip:n \skip_horizontal:n

```

\xeCJK_punct_kern:NN 相邻两个标点之间的间距, 总允许长标点与其他标点之间折行, 但 NoBreakLongPunct 在右侧时禁止折行。

_xeCJK_punct_kern:NN

```

1522 \cs_new_protected:Npn \_xeCJK_punct_kern:NN #1#2
1523 {
1524   \str_if_eq:eeTF {#1} {#2}
1525   { \_xeCJK_punct_nobreak_kern:NN }
1526   {
1527     \_xeCJK_punct_if_long:NTF #1
1528     { \_xeCJK_punct_breakable_kern:NN }
1529     {
1530       \_xeCJK_punct_if_long:NTF #2
1531       {
1532         \_xeCJK_punct_if_nobreak_long:NTF #2
1533         { \_xeCJK_punct_nobreak_kern:NN }
1534         { \_xeCJK_punct_breakable_kern:NN }
1535       }
1536       { \_xeCJK_punct_nobreak_kern:NN }
1537     }
1538   }
1539   #1 #2
1540 }
1541 \cs_new_eq:NN \xeCJK_punct_kern:NN \_xeCJK_punct_kern:NN

```

_xeCJK_punct_nobreak_kern:NN

```

1542 \cs_new_protected:Npn \_xeCJK_punct_nobreak_kern:NN #1#2
1543 { \_xeCJK_nobreak_hskip:n { \_xeCJK_use_dim_or_skip:nNN { kern } #1 #2 } }

```

_xeCJK_punct_breakable_kern:NN

```

1544 \cs_new_protected:Npn \_xeCJK_punct_breakable_kern:NN #1#2
1545 {
1546   \exp_after:wN \_xeCJK_punct_if_right:NT #1
1547   { \_xeCJK_punct_rule:NN \c__xeCJK_right_tl #1 }
1548   \_xeCJK_punct_breakable_kern:n
1549   { \_xeCJK_use_dim_or_skip:nNN { bound_kern } #1 #2 }
1550   \_xeCJK_punct_if_right:NF #2
1551   { \_xeCJK_punct_rule:NN \c__xeCJK_left_tl #2 }
1552 }
1553 \cs_new_eq:NN \_xeCJK_punct_breakable_kern:n \skip_horizontal:n

```

`\g__xeCJK_last_punct_tl` 用于记录当前的标点符号。

```
1554 \tl_new:N \g__xeCJK_last_punct_tl
```

```
\xeCJK_FullLeft_and_CJK: 1555 \cs_new_protected:Npn \xeCJK_FullLeft_and_CJK:
1556 {
1557   \__xeCJK_punct_if_middle:NTF \g__xeCJK_last_punct_tl
1558   {
1559     \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
1560     \xeCJK_no_break:
1561     \__xeCJK_punct_glue:NN \c__xeCJK_left_tl \g__xeCJK_last_punct_tl
1562   }
1563   { }
1564   \__xeCJK_select_font:
1565 }
```

`\xeCJK_FullLeft_and_Default:` `__xeCJK_nobreak_zero_glue:` 用于确保 FullLeft 类后的西文单词可以断词。

```
1566 \cs_new_protected:Npn \xeCJK_FullLeft_and_Default:
1567 {
1568   \__xeCJK_punct_if_middle:NTF \g__xeCJK_last_punct_tl
1569   {
1570     \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
1571     \xeCJK_class_group_end: \xeCJK_no_break:
1572     \__xeCJK_punct_glue:NN \c__xeCJK_left_tl \g__xeCJK_last_punct_tl
1573   }
1574   {
1575     \xeCJK_class_group_end:
1576     \__xeCJK_nobreak_zero_glue:
1577   }
1578 }
1579 \cs_new_protected:Npn \__xeCJK_nobreak_zero_glue:
1580 {
1581   \tex_penalty:D \c__xeCJK_nobreak_penalty_int
1582   \skip_horizontal:N \c_zero_skip
1583 }
1584 \cs_new_protected:Npn \__xeCJK_zero_glue:
1585 { \skip_horizontal:N \c_zero_skip }
```

```
\xeCJK_FullRight_and_CJK: 1586 \cs_new_protected:Npn \xeCJK_FullRight_and_CJK:
1587 {
1588   \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
1589   \__xeCJK_punct_glue:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
1590   \__xeCJK_select_font:
1591   \CJKglue
1592 }
```

```
\xeCJK_FullRight_and_Default: 1593 \cs_new_protected:Npn \xeCJK_FullRight_and_Default:
1594 {
1595   \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
1596   \xeCJK_class_group_end:
1597   \__xeCJK_punct_glue:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
1598 }
```

```
\xeCJK_Default_and_FullLeft:nN 1599 \cs_new_protected:Npn \xeCJK_Default_and_FullLeft:nN #1#2
1600 {
1601   \xeCJK_get_punct_bounds:NN \c__xeCJK_left_tl #2
1602   \__xeCJK_Default_and_FullLeft_glue:N #2
1603   \xeCJK_class_group_begin:
1604   \xeCJK_select_punct_font:
1605   \xeCJK_clear_inter_class_toks:nn {#1} { FullLeft }
1606   \xeCJK_clear_Boundary_and_CJK_toks:
1607   \tl_gset:Nn \g__xeCJK_last_punct_tl {#2}
1608   \__xeCJK_punct_rule:NN \c__xeCJK_left_tl #2
1609   \xeCJK_fallback_punct_symbol:NN
1610   \CJKpunctsymbol #2
1611 }
1612 \cs_new_protected:Npn \__xeCJK_Default_and_FullLeft_glue:N #1
1613 { \__xeCJK_punct_glue:NN \c__xeCJK_left_tl #1 }
```

```

\xeCJK_CJK_and_FullLeft:N 1614 \cs_new_protected:Npn \xeCJK_CJK_and_FullLeft:N #1
1615 {
1616   \xeCJK_get_punct_bounds:NN \c__xeCJK_left_tl #1
1617   \__xeCJK_CJK_and_FullLeft_glue:N #1
1618   \tl_gset:Nn \g__xeCJK_last_punct_tl {#1}
1619   \__xeCJK_punct_rule:NN \c__xeCJK_left_tl #1
1620   \__xeCJK_select_punct_font:
1621   \xeCJK_fallback_punct_symbol:NN
1622   \CJKpunctsymbol #1
1623 }
1624 \cs_new_protected:Npn \__xeCJK_CJK_and_FullLeft_glue:N #1
1625 {
1626   \CJKglue
1627   \__xeCJK_punct_glue:NN \c__xeCJK_left_tl #1
1628 }

```

```

\xeCJK_Boundary_and_FullLeft:N 1629 \cs_new_protected:Npn \xeCJK_Boundary_and_FullLeft:N #1
1630 {
1631   \xeCJK_get_punct_bounds:NN \c__xeCJK_left_tl #1
1632   \__xeCJK_Boundary_and_FullLeft_glue:N #1
1633   \xeCJK_class_group_begin:
1634   \xeCJK_select_punct_font:
1635   \xeCJK_clear_Boundary_and_CJK_toks:
1636   \tl_gset:Nn \g__xeCJK_last_punct_tl {#1}
1637   \__xeCJK_punct_rule:NN \c__xeCJK_left_tl #1
1638   \xeCJK_fallback_punct_symbol:NN
1639   \CJKpunctsymbol #1
1640 }

```

`__xeCJK_Boundary_and_FullLeft_glue:N` 根据 `\etex_lastnodetype:D` 的值进行分别处理。

```

1641 \cs_new_protected:Npn \__xeCJK_Boundary_and_FullLeft_glue:N #1
1642 {
1643   \tl_set_eq:NN \l__xeCJK_alignii_tl \c__xeCJK_left_tl
1644   \group_begin: \exp_args:Nnc \group_end: \cs_if_exist_use:NTF
1645     { \__xeCJK_bound_type_ \int_use:N \etex_lastnodetype:D _glue:Nn }
1646     {#1}
1647     { \use:n }
1648     { \__xeCJK_punct_glue:NN \c__xeCJK_left_tl #1 }
1649 }
1650 \tl_new:N \c__xeCJK_alignii_tl

```

`__xeCJK_bound_type_-1_glue:Nn` `\etex_lastnodetype:D` 为 `-1` 表示 empty list, 常出现在盒子的起始位置, 在段落前使用 `\noindent` 就是这种情况。

```

1651 \cs_new_protected:cpn { \__xeCJK_bound_type_-1_glue:Nn } #1#2
1652 { \__xeCJK_zero_glue: }

```

`__xeCJK_bound_type_1_glue:Nn` `1` 表示 hlist node, 在这里用来判断是否位于段首。基于正常情况下, TeX 会在段落开头插入宽度为 `\parindent` 的水平盒子用于缩进。

```

1653 \cs_new_protected:cpn { \__xeCJK_bound_type_ 1_glue:Nn } #1
1654 {
1655   \int_do_while:nNnn \etex_lastnodetype:D = \c__xeCJK_hlist_node
1656     { \__xeCJK_bound_hbox_auxi: }
1657   \__xeCJK_if_last_none:TF
1658     {
1659       \dim_case:nnF { \box_wd:N \l__xeCJK_indent_box }
1660       {
1661         { \etex_parindent:D } { \__xeCJK_bound_hbox_auxii:nn }
1662         { \c_zero_dim } { \use_i:nn }
1663       }
1664       { \use:nn }
1665     }
1666     { \use:nn }
1667     { \hbox_unpack_drop:N \l__xeCJK_indent_box }
1668 }
1669 \cs_new_protected:Npn \__xeCJK_bound_hbox_auxi:

```

```

1670 {
1671   \box_set_to_last:N \l__xeCJK_tmp_box
1672   \hbox_set:Nn \l__xeCJK_indent_box
1673   {
1674     \box_use:N \l__xeCJK_tmp_box
1675     \hbox_unpack:N \l__xeCJK_indent_box
1676   }
1677 }
1678 \cs_new_protected:Npn \__xeCJK_bound_hbox_auxii:nn
1679 {
1680   \dim_compare:nNnTF
1681     { \box_ht:N \l__xeCJK_tmp_box } = \c_zero_dim
1682     { \use_i:nn }
1683     { \use:nn }
1684 }
1685 \box_new:N \l__xeCJK_indent_box

```

`__xeCJK_bound_type_11_glue:Nn` 11 表示 glue node, 这里判断的目的是当全角左标点出现在 L^AT_EX 表格的非 p 列行首时, 能够对齐到单元格的边界。判断基于标准 L^AT_EX 表格的列格式 (`\@tabclassz`) 定义中, 在 l 列和 r 列前为了防止 `\tabcolsep` 被无意 `\unskip` 掉, 都加了 `\hskip1sp`, 而 c 列前则有 `\hfil`。enumitem 宏包修改了 description 环境中使用的 `\item(\enit@postlabel@i)`, 在这里起到影响作用的是 `\penalty\z@ \hskip\labelsep`。

```

1686 \cs_new_protected:cpn { __xeCJK_bound_type_ 11 _glue:Nn } #1#2
1687 {
1688   \skip_if_finite:nTF { \tex_lastskip:D }
1689     { \__xeCJK_bound_glue_auxi:Nn #1 {#2} }
1690     { \__xeCJK_zero_glue: }
1691 }
1692 \cs_new_protected:Npn \__xeCJK_bound_glue_auxi:Nn #1#2
1693 {
1694   \__xeCJK_if_last_punct_glue:TF
1695     { \xeCJK_punct_bound_kern:N #1 }
1696     { \__xeCJK_bound_glue_auxii:n {#2} }
1697 }
1698 \cs_new_protected:Npn \__xeCJK_bound_glue_auxii:n #1
1699 {
1700   \skip_set_eq:NN \l__xeCJK_last_skip \tex_lastskip:D
1701   \skip_if_eq:nnTF { \l__xeCJK_last_skip } { 1sp }
1702     { \__xeCJK_zero_glue: }
1703     {
1704       \skip_if_eq:nnTF { \l__xeCJK_last_skip } { \labelsep }
1705         {
1706           \tex_unskip:D
1707           \__xeCJK_if_last_penalty:TF
1708             {
1709               \int_compare:nNnTF \tex_lastpenalty:D = \c_zero_int
1710                 { \skip_horizontal:N \l__xeCJK_last_skip }
1711                 { \skip_horizontal:N \l__xeCJK_last_skip #1 }
1712             }
1713             { \skip_horizontal:N \l__xeCJK_last_skip #1 }
1714         }
1715         {#1}
1716     }
1717 }

```

`__xeCJK_bound_type_12_glue:Nn` 12 表示 kern node, 用于判断之前的字符是否是 CJK 类, 如果是, 则插入 `\CJKglue`。

```

1718 \cs_new_protected:cpn { __xeCJK_bound_type_ 12 _glue:Nn } #1#2
1719 {
1720   \xeCJK_if_last_node:nF { CJK }
1721     { \xeCJK_if_last_node:nF { CJK-space } { \use_none:nn } }
1722   \xeCJK_remove_node: \CJKglue
1723   #2
1724 }

```

`_xeCJK_bound_type_13_glue:n` 13 表示 `penalty node`, 这里判断的目的是全角左标点出现在 `LATEX` 列表环境的 `\item` 后面时, 能对齐到边界。判断基于 `\item` 的内部定义 `\@item` 对 `\everypar` 进行了修改, 在这里起到影响作用的是 `\box\@labels \penalty\z@`。以上判断都比较粗略, 暂时也没有想起更好的办法。

```

1725 \cs_new_protected:cpn { \_xeCJK_bound_type_13_glue:Nn } #1#2
1726 {
1727   \_xeCJK_if_last_punct_penalty:TF
1728   { \xeCJK_punct_bound_kern:N #1 }
1729   {
1730     \int_compare:nNnTF \tex_lastpenalty:D = \c_zero_int
1731     {
1732       \tex_unpenalty:D
1733       \_xeCJK_if_last_hlist:TF
1734       { \tex_penalty:D \c_zero_int }
1735       { \tex_penalty:D \c_zero_int #2 }
1736     }
1737     {#2}
1738   }
1739 }

```

```

\_xeCJK_Default_and_FullRight:nN 1740 \cs_new_protected:Npn \xeCJK_Default_and_FullRight:nN #1#2
1741 {
1742   \xeCJK_get_punct_bounds:NN \c__xeCJK_right_tl #2
1743   \_xeCJK_Default_and_FullRight_glue:N #2
1744   \xeCJK_class_group_begin:
1745   \xeCJK_select_punct_font:
1746   \xeCJK_clear_inter_class_toks:nn {#1} { FullRight }
1747   \xeCJK_clear_Boundary_and_CJK_toks:
1748   \tl_gset:Nn \g__xeCJK_last_punct_tl {#2}
1749   \xeCJK_FullRight_symbol:N #2
1750 }

```

```

\_xeCJK_Boundary_and_FullRight:N 1751 \cs_new_protected:Npn \xeCJK_Boundary_and_FullRight:N #1
1752 {
1753   \xeCJK_get_punct_bounds:NN \c__xeCJK_right_tl #1
1754   \xeCJK_if_last_punct:TF
1755   {
1756     \tl_set_eq:NN \l__xeCJK_alignii_tl \c__xeCJK_right_tl
1757     \xeCJK_punct_bound_kern:N
1758   }
1759   { \_xeCJK_Default_and_FullRight_glue:N }
1760   #1
1761   \xeCJK_class_group_begin:
1762   \xeCJK_select_punct_font:
1763   \xeCJK_clear_Boundary_and_CJK_toks:
1764   \tl_gset:Nn \g__xeCJK_last_punct_tl {#1}
1765   \xeCJK_FullRight_symbol:N #1
1766 }

```

```

\_xeCJK_CJK_and_FullRight:N 1767 \cs_new_protected:Npn \xeCJK_CJK_and_FullRight:N #1
1768 {
1769   \xeCJK_get_punct_bounds:NN \c__xeCJK_right_tl #1
1770   \_xeCJK_CJK_and_FullRight_glue:N #1
1771   \tl_gset:Nn \g__xeCJK_last_punct_tl {#1}
1772   \_xeCJK_select_punct_font:
1773   \xeCJK_FullRight_symbol:N #1
1774 }

```

`\xeCJK_if_last_punct:TF` 判断之前是否是一个标点符号。

```

1775 \cs_new_protected:Npn \xeCJK_if_last_punct:TF
1776 {
1777   \bool_set_false:N \l__xeCJK_last_penalty_bool
1778   \_xeCJK_if_last_glue:TF
1779   { \_xeCJK_if_last_punct_glue:TF }
1780   {

```

```

1781     \__xeCJK_if_last_penalty:TF
1782     { \__xeCJK_if_last_punct_penalty:TF }
1783     { \use_ii:nn }
1784   }
1785 }
1786 \cs_new_protected:Npn \__xeCJK_if_last_punct_glue:TF
1787 {
1788   \prop_get:NoNTF \g__xeCJK_punct_skip_prop
1789   { \skip_use:N \tex_lastskip:D } \l__xeCJK_tmp_tl
1790   { \__xeCJK_if_last_punct_glue_auxi:TF }
1791   { \__xeCJK_if_last_punct_glue_auxii:TF }
1792 }
1793 \cs_new_protected:Npn \__xeCJK_if_last_punct_glue_auxi:TF
1794 {
1795   \skip_set_eq:NN \l__xeCJK_last_skip \tex_lastskip:D
1796   \tex_unskip:D
1797   \int_compare:nNnTF \tex_lastpenalty:D = \c__xeCJK_nobreak_penalty_int
1798   { \__xeCJK_if_last_punct_auxi:TF { \use_i:nn } }
1799   {
1800     \xeCJK_if_last_node:TF
1801     { \__xeCJK_if_last_punct_auxii:TF { \use_i:nn } }
1802     { \use:n }
1803   }
1804   { \skip_horizontal:N \l__xeCJK_last_skip \use_ii:nn }
1805 }
1806 \cs_new_protected:Npn \__xeCJK_if_last_punct_glue_auxii:TF
1807 {
1808   \group_begin:
1809   \g__xeCJK_space_factor_int \tex_spacefactor:D
1810   \skip_if_eq:nnTF { \tex_lastskip:D } { \c_xeCJK_space_skip_tl }
1811   { \group_end: \__xeCJK_if_last_punct_glue_auxiii:TF }
1812   { \group_end: \use_ii:nn }
1813 }
1814 \cs_new_protected:Npn \__xeCJK_if_last_punct_glue_auxiii:TF
1815 {
1816   \skip_set_eq:NN \l__xeCJK_tmp_skip \tex_lastskip:D
1817   \tex_unskip:D
1818   \__xeCJK_if_last_glue:TF
1819   {
1820     \prop_get:NoNTF \g__xeCJK_punct_skip_prop
1821     { \skip_use:N \tex_lastskip:D } \l__xeCJK_tmp_tl
1822     { \__xeCJK_if_last_punct_glue_auxi:TF { \use_i:nn } }
1823     { \use:n }
1824   }
1825   { \use:n }
1826   { \skip_horizontal:N \l__xeCJK_tmp_skip \use_ii:nn }
1827 }
1828 \cs_new_protected:Npn \__xeCJK_if_last_punct_penalty:TF
1829 {
1830   \int_set_eq:NN \l__xeCJK_last_penalty_int \tex_lastpenalty:D
1831   \tex_unpenalty:D
1832   \bool_set_true:N \l__xeCJK_last_penalty_bool
1833   \__xeCJK_if_last_glue:TF
1834   { \__xeCJK_if_last_punct_glue:TF { \use_i:nn } }
1835   { \use:n }
1836   { \__xeCJK_last_punct_penalty_false:nn }
1837 }
1838 \cs_new_protected:Npn \__xeCJK_last_punct_penalty_false:nn #1#2
1839 {
1840   \bool_set_false:N \l__xeCJK_last_penalty_bool
1841   \tex_penalty:D \l__xeCJK_last_penalty_int
1842   #2
1843 }
1844 \cs_new_protected:Npn \__xeCJK_if_last_punct_auxi:TF
1845 {
1846   \tex_unpenalty:D
1847   \bool_if:NF \l__xeCJK_last_penalty_bool

```

```

1848     {
1849         \bool_set_true:N \l__xeCJK_last_penalty_bool
1850         \int_set_eq:NN \l__xeCJK_last_penalty_int \c__xeCJK_nobreak_penalty_int
1851     }
1852     \xeCJK_if_last_node:TF
1853     { \__xeCJK_if_last_punct_auxii:TF { \use_i:nn } }
1854     { \use:n }
1855     { \xeCJK_no_break: \use_ii:nn }
1856 }
1857 \cs_new_protected:Npn \__xeCJK_if_last_punct_auxii:TF
1858 {
1859     \dim_compare:nNnTF \l__xeCJK_last_kern_dim > \c_zero_dim
1860     { \__xeCJK_if_last_punct_auxiii:TF }
1861     { \__xeCJK_make_node:N \l__xeCJK_last_kern_dim \use_ii:nn }
1862 }
1863 \cs_new_protected:Npn \__xeCJK_if_last_punct_auxiii:TF
1864 {
1865     \int_case:nnTF { \tex_XeTeXcharclass:D \l__xeCJK_last_kern_dim }
1866     {
1867         { \xeCJK_class_num:n { FullRight } }
1868         { \tl_set_eq:NN \l__xeCJK_aligni_tl \c__xeCJK_right_tl }
1869         { \xeCJK_class_num:n { FullLeft } }
1870         { \tl_set_eq:NN \l__xeCJK_aligni_tl \c__xeCJK_left_tl }
1871     }
1872     { \__xeCJK_if_last_punct_auxiv:TF }
1873     { \use_ii:nn }
1874 }
1875 \cs_new_protected:Npn \__xeCJK_if_last_punct_auxiv:TF
1876 {
1877     \dim_set_eq:NN \l__xeCJK_tmp_dim \l__xeCJK_last_kern_dim
1878     \xeCJK_if_last_node:TF
1879     {
1880         \tl_gset:Ne \g__xeCJK_last_punct_tl
1881         { \tex_Uchar:D \l__xeCJK_tmp_dim }
1882         \dim_set_eq:NN \l__xeCJK_last_bound_dim \l__xeCJK_last_kern_dim
1883         \use_i:nn
1884     }
1885     { \__xeCJK_make_node:N \l__xeCJK_tmp_dim \use_ii:nn }
1886 }
1887 \tl_new:N \l__xeCJK_aligni_tl
1888 \tl_new:N \l__xeCJK_alignii_tl
1889 \int_new:N \l__xeCJK_last_penalty_int
1890 \dim_new:N \l__xeCJK_last_bound_dim
1891 \bool_new:N \l__xeCJK_last_penalty_bool

\xeCJK_if_last_node:TF 1892 \cs_new_protected:Npn \xeCJK_if_last_node:TF #1#2
1893 {
1894     \__xeCJK_if_last_kern:TF
1895     {
1896         \dim_set_eq:NN \l__xeCJK_last_kern_dim \tex_lastkern:D
1897         \tex_unkern:D
1898         \__xeCJK_if_last_kern:TF
1899         {
1900             \dim_compare:nNnTF \tex_lastkern:D = { - \l__xeCJK_last_kern_dim }
1901             { \tex_unkern:D #1 }
1902             { \tex_kern:D \l__xeCJK_last_kern_dim #2 }
1903         }
1904         { \tex_kern:D \l__xeCJK_last_kern_dim #2 }
1905     }
1906     {#2}
1907 }
1908 \dim_new:N \l__xeCJK_last_kern_dim

\_xeCJK_CJK_and_FullRight_glue:N 1909 \cs_new_protected:Npn \__xeCJK_CJK_and_FullRight_glue:N #1
\_xeCJK_Default_and_FullRight_glue:N 1910 {
1911     \__xeCJK_punct_if_long:NTF #1
1912     {

```

```

1913     \__xeCJK_punct_if_nobreak_long:NTF #1
1914     { \xeCJK_no_break: }
1915     { \xeCJK_allow_break: }
1916   }
1917   { \xeCJK_no_break: }
1918   \__xeCJK_punct_if_middle:NT #1
1919   {
1920     \CJKglue
1921     \__xeCJK_punct_glue:NN \c__xeCJK_right_tl #1
1922     \__xeCJK_punct_rule:NN \c__xeCJK_left_tl #1
1923   }
1924 }
1925 \cs_new_protected:Npn \__xeCJK_Default_and_FullRight_glue:N #1
1926 {
1927   \__xeCJK_punct_if_long:NTF #1
1928   {
1929     \__xeCJK_punct_if_nobreak_long:NTF #1
1930     { \xeCJK_no_break: }
1931     { \xeCJK_allow_break: }
1932   }
1933   { \xeCJK_no_break: }
1934   \__xeCJK_punct_if_middle:NT #1
1935   {
1936     \__xeCJK_punct_glue:NN \c__xeCJK_right_tl #1
1937     \__xeCJK_punct_rule:NN \c__xeCJK_left_tl #1
1938   }
1939 }

\xeCJK_FullLeft_and_FullLeft:N 1940 \cs_new_protected:Npn \xeCJK_FullLeft_and_FullLeft:N #1
1941 {
1942   \xeCJK_get_punct_bounds:NN \c__xeCJK_left_tl #1
1943   \xeCJK_get_punct_kerning:oN \g__xeCJK_last_punct_tl #1
1944   \__xeCJK_punct_kern:NN \g__xeCJK_last_punct_tl #1
1945   \tl_gset:Nn \g__xeCJK_last_punct_tl {#1}
1946   \xeCJK_fallback_punct_symbol:NN
1947   \CJKpunctsymbol #1
1948 }

\xeCJK_FullLeft_and_FullRight:N 1949 \cs_new_protected:Npn \xeCJK_FullLeft_and_FullRight:N #1
1950 {
1951   \xeCJK_get_punct_bounds:NN \c__xeCJK_right_tl #1
1952   \xeCJK_get_punct_kerning:oN \g__xeCJK_last_punct_tl #1
1953   \__xeCJK_punct_kern:NN \g__xeCJK_last_punct_tl #1
1954   \tl_gset:Nn \g__xeCJK_last_punct_tl {#1}
1955   \xeCJK_FullRight_symbol:N #1
1956 }

\xeCJK_FullRight_and_FullLeft:N 1957 \cs_new_protected:Npn \xeCJK_FullRight_and_FullLeft:N #1
1958 {
1959   \xeCJK_get_punct_bounds:NN \c__xeCJK_left_tl #1
1960   \xeCJK_get_punct_kerning:oN \g__xeCJK_last_punct_tl #1
1961   \xeCJK_punct_kern:NN \g__xeCJK_last_punct_tl #1
1962   \tl_gset:Nn \g__xeCJK_last_punct_tl {#1}
1963   \xeCJK_fallback_punct_symbol:NN
1964   \CJKpunctsymbol #1
1965 }

\xeCJK_FullRight_and_FullRight:N 1966 \cs_new_protected:Npn \xeCJK_FullRight_and_FullRight:N #1
1967 {
1968   \xeCJK_get_punct_bounds:NN \c__xeCJK_right_tl #1
1969   \xeCJK_get_punct_kerning:oN \g__xeCJK_last_punct_tl #1
1970   \__xeCJK_punct_kern:NN \g__xeCJK_last_punct_tl #1
1971   \tl_gset:Nn \g__xeCJK_last_punct_tl {#1}
1972   \xeCJK_FullRight_symbol:N #1
1973 }

```

5.7 全角右标点后的断行

CheckFullRight 选项设置。

```

1974 \keys_define:nn { xeCJK / options }
1975 {
1976   CheckFullRight .choice: ,
1977   CheckFullRight / true .code:n =
1978   {
1979     \cs_if_eq:NNF \xeCJK_FullRight_and_Boundary: \xeCJK_check_FullRight:
1980     {
1981       \cs_set_eq:NN \__xeCJK_save_FullRight_check:
1982         \xeCJK_FullRight_and_Boundary:
1983       \cs_set_eq:NN \__xeCJK_save_FullRight_symbol:N
1984         \xeCJK_FullRight_symbol:N
1985       \cs_set_eq:NN \xeCJK_FullRight_and_Boundary:
1986         \xeCJK_check_FullRight:
1987       \cs_set_eq:NN \xeCJK_FullRight_symbol:N
1988         \xeCJK_check_FullRight_symbol:Nw
1989     }
1990   } ,
1991   CheckFullRight / false .code:n =
1992   {
1993     \cs_if_eq:NNT \xeCJK_FullRight_and_Boundary: \xeCJK_check_FullRight:
1994     {
1995       \cs_set_eq:NN \xeCJK_FullRight_and_Boundary:
1996         \__xeCJK_save_FullRight_check:
1997       \cs_set_eq:NN \xeCJK_FullRight_symbol:N
1998         \__xeCJK_save_FullRight_symbol:N
1999     }
2000   } ,
2001   CheckFullRight .default:n = { true }
2002 }

```

```

experiment/halfright-prebreakpenalty 2003 \keys_define:nn { xeCJK / options / experiment }
experiment/punct-measure-fix 2004 {
2005   halfright-prebreakpenalty .bool_set:N = \l__xeCJK_halfright_nobreak_bool ,
2006   halfright-prebreakpenalty .default:n = true ,
2007   halfright-prebreakpenalty .initial:n = false ,
2008   punct-measure-fix .bool_gset:N = \g__xeCJK_punct_measure_fix_bool ,
2009   punct-measure-fix .default:n = true ,
2010   punct-measure-fix .initial:n = false
2011 }

```

```

\xeCJK_FullRight_symbol:N 2012 \cs_new_protected:Npn \xeCJK_FullRight_symbol:N
2013 {
2014   \xeCJK_fallback_punct_symbol:NN
2015   \CJKpunctsymbol
2016 }

```

```

\xeCJK_check_FullRight: 2017 \cs_new_protected:Npn \xeCJK_check_FullRight:
2018 {
2019   \xeCJK_get_punct_bounds:No \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
2020   \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
2021   \group_align_safe_begin:
2022   \token_case_meaning:NoTF \l_peek_token
2023   { \l__xeCJK_no_break_cs_case_tl }
2024   {
2025     \group_align_safe_end:
2026     \xeCJK_no_break:
2027     \group_insert_after:N \xeCJK_no_break:
2028   }
2029   { \group_align_safe_end: }
2030   \exp_after:wN \xeCJK_punct_node:N \g__xeCJK_last_punct_tl
2031   \xeCJK_class_group_end:
2032   \__xeCJK_punct_glue:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
2033   \__xeCJK_punct_boundary_guard:
2034 }
2035 \prg_generate_conditional_variant:Nnn \token_case_meaning:Nn { No } { TF , F }

```

```

\XeCJK_check_FullRight_symbol:Nw 2036 \cs_new_protected:Npn \XeCJK_check_FullRight_symbol:Nw #1
2037 { \peek_remove_spaces:n { \__XeCJK_save_FullRight_symbol:N #1 } }

\XeCJK_cs_case_keys_define:nNnNn 2038 \cs_new_protected:Npn \XeCJK_cs_case_keys_define:nNnNn #1#2#3#4#5
2039 {
2040   \tl_new:N #2
2041   \seq_new:N #3
2042   \keys_define:nn { XeCJK / options }
2043   {
2044     #1 .code:n =
2045     {
2046       \seq_set_split:Nnn #3 { } {##1}
2047       \__XeCJK_update_cs_case_tl:NnNn #2#3 {#4} {#5}
2048     } ,
2049     #1+ .code:n =
2050     {
2051       \tl_map_inline:nn {##1}
2052       { \seq_if_in:NnF #3 {###1} { \seq_put_right:Nn #3 {###1} } }
2053       \__XeCJK_update_cs_case_tl:NnNn #2#3 {#4} {#5}
2054     } ,
2055     #1- .code:n =
2056     {
2057       \tl_map_inline:nn {##1} { \seq_remove_all:Nn #3 {###1} }
2058       \__XeCJK_update_cs_case_tl:NnNn #2#3 {#4} {#5}
2059     }
2060   }
2061 }
2062 \cs_new_protected:Npn \__XeCJK_update_cs_case_tl:NnNn #1#2#3#4
2063 {
2064   \tl_clear:N #1
2065   \seq_map_inline:Nn #2 { \tl_put_right:Nn #1 { {##1} {#3} } }
2066   #4
2067 }

```

NoBreakCS 设置不能在全角右标点之后断行的控制序列。

```

2068 \XeCJK_cs_case_keys_define:nNnNn { NoBreakCS }
2069 \l__XeCJK_no_break_cs_case_tl \l__XeCJK_no_break_cs_seq { } { }

```

\XeCJKnobreak 为保险起见, 我们在这里用了一个循环。

```

2070 \NewDocumentCommand \XeCJKnobreak { }
2071 {
2072   \bool_set_true:N \l__XeCJK_tmp_bool
2073   \int_while_do:nNnn \tex_lastnodetype:D = \c__XeCJK_glue_node
2074   {
2075     \bool_if:NTF \l__XeCJK_tmp_bool
2076     {
2077       \bool_set_false:N \l__XeCJK_tmp_bool
2078       \skip_set_eq:NN \l__XeCJK_last_skip \tex_lastskip:D
2079     }
2080     { \skip_add:Nn \l__XeCJK_last_skip \tex_lastskip:D }
2081     \tex_unskip:D
2082   }
2083   \XeCJK_if_last_node:TF
2084   {
2085     \dim_set_eq:NN \l__XeCJK_tmp_dim \l__XeCJK_last_kern_dim
2086     \XeCJK_if_last_node:TF
2087     {
2088       \__XeCJK_if_last_glue:T
2089       {
2090         \exp_args:NnNo \tex_unskip:D \XeCJK_no_break:
2091         \skip_horizontal:n { \skip_use:N \tex_lastskip:D }
2092       }
2093       \__XeCJK_make_node:N \l__XeCJK_last_kern_dim
2094     }
2095     { }
2096     \__XeCJK_make_node:N \l__XeCJK_tmp_dim

```

```

2097     }
2098     { }
2099     \xeCJK_no_break:
2100     \bool_if:NF \l__xeCJK_tmp_bool
2101     { \skip_horizontal:N \l__xeCJK_last_skip }
2102   }

```

5.8 段末孤字处理

CheckSingle 孤字处理功能选项。

```

2103 \keys_define:nn { xeCJK / options }
2104   {
2105     CheckSingle .choice: ,
2106     CheckSingle / true .code:n =
2107       {
2108         \cs_if_eq:NNF \xeCJK_CJK_and_CJK:N \xeCJK_check_single:Nw
2109         {
2110           \cs_set_eq:NN \__xeCJK_check_single_save:N \xeCJK_CJK_and_CJK:N
2111           \cs_set_eq:NN \xeCJK_CJK_and_CJK:N \xeCJK_check_single:Nw
2112         }
2113       } ,
2114     CheckSingle / false .code:n =
2115       {
2116         \cs_if_eq:NNT \xeCJK_CJK_and_CJK:N \xeCJK_check_single:Nw
2117         { \cs_set_eq:NN \xeCJK_CJK_and_CJK:N \__xeCJK_check_single_save:N }
2118       } ,
2119     CheckSingle .default:n = { true } ,
2120     CJKchecksingle .meta:n = { CheckSingle = true }
2121   }

```

WidowPenalty 设置段末汉字的 penalty, 默认值是10 000。

```

2122 \keys_define:nn { xeCJK / options }
2123   {
2124     WidowPenalty .int_set:N = \l__xeCJK_widow_penalty_int ,
2125     WidowPenalty .default:n = { 10 000 }
2126   }

```

`\xeCJK_widow_penalty:` 预防段末孤字而插入的 penalty, 值为 `\l__xeCJK_widow_penalty_int`。

```

2127 \cs_new_protected:Npn \xeCJK_widow_penalty:
2128   { \tex_penalty:D \l__xeCJK_widow_penalty_int }

```

```

\__xeCJK_check_single_end:N \xeCJK_check_single:Nw #1
2129 \cs_new_protected:Npn \xeCJK_check_single:Nw #1
2130   {
2131     \group_align_safe_begin:
2132     \peek_catcode:NTF \c_catcode_letter_token
2133     { \xeCJK_check_single:NNw #1 }
2134     {
2135       \token_if_other:NTF \l_peek_token
2136       { \xeCJK_check_single:NNw }
2137       { \__xeCJK_check_single_end:N }
2138     } #1
2139   }
2140 }
2141 \cs_new_protected:Npn \__xeCJK_check_single_end:N
2142   {
2143     \group_align_safe_end:
2144     \__xeCJK_check_single_save:N
2145   }

```

`\xeCJK_check_single:NNw` 使用 `\group_align_safe_begin:` 和 `\group_align_safe_end:` 是为了防止在表格里面报错。
`__xeCJK_check_single_aux:nNw`

```

2146 \cs_new_protected:Npn \xeCJK_check_single:NNw #1#2

```

```

2147 {
2148   \xeCJK_peek_catcode_ignore_spaces:NTF \c_catcode_letter_token
2149   {
2150     \bool_if:NTF \l__xeCJK_peek_ignore_spaces_bool
2151     {
2152       \bool_if:NTF \l__xeCJK_reserve_space_bool
2153       { \__xeCJK_check_single_end:N #1 #2 ~ }
2154       { \__xeCJK_check_single_space:NN #1#2 }
2155     }
2156     { \__xeCJK_check_single_end:N #1 #2 }
2157   }
2158   {
2159     \token_if_other:NTF \l_peek_token
2160     {
2161       \bool_if:NTF \l__xeCJK_peek_ignore_spaces_bool
2162       { \__xeCJK_check_single_space:NN }
2163       { \__xeCJK_check_single_end:N }
2164     }
2165     {
2166       \bool_if:NTF \l__xeCJK_peek_ignore_spaces_bool
2167       { \__xeCJK_check_single_aux:nNNw { ~ } }
2168       { \__xeCJK_check_single_aux:nNNw { } }
2169     }
2170     #1 #2
2171   }
2172 }
2173 \cs_new_protected:Npn \__xeCJK_check_single_aux:nNNw #1#2#3
2174 {
2175   \token_if_cs:NTF \l_peek_token
2176   { \xeCJK_check_single_cs:NNn }
2177   { \xeCJK_check_single_end:NNnw }
2178   #2 #3 {#1}
2179 }

```

```

\xeCJK_check_single_end:NNnw 2180 \cs_new_protected:Npn \__xeCJK_check_single_end_aux:NNn #1#2#3
\__xeCJK_check_single_end_aux:NNn 2181 { \__xeCJK_check_single_end:N #1 #2 #3 }
\__xeCJK_check_single_end_equation:NNnw 2182 \cs_new_eq:NN \xeCJK_check_single_end:NNnw \__xeCJK_check_single_end_aux:NNn
2183 \cs_new_protected:Npn \__xeCJK_check_single_end_equation:NNnw
2184 {
2185   \token_if_math_toggle:NTF \l_peek_token
2186   { \xeCJK_check_single_equation:NNnw }
2187   { \__xeCJK_check_single_end_aux:NNn }
2188 }

```

```

PlainEquation 2189 \keys_define:nn { xeCJK / options }
2190 {
2191   PlainEquation .choice: ,
2192   PlainEquation / true .code:n =
2193   {
2194     \cs_set_eq:NN \xeCJK_check_single_end:NNnw
2195     \__xeCJK_check_single_end_equation:NNnw
2196   } ,
2197   PlainEquation / false .code:n =
2198   {
2199     \cs_set_eq:NN \xeCJK_check_single_end:NNnw
2200     \__xeCJK_check_single_end_aux:NNn
2201   } ,
2202   PlainEquation .default:n = { true } ,
2203 }

```

```

\__xeCJK_check_single_space:NN 2204 \cs_new_protected:Npn \__xeCJK_check_single_space:NN #1#2
2205 {
2206   \xeCJK_if_CJK_class:NTF #2
2207   {
2208     \xeCJK_if_CJK_class:NTF \l_peek_token
2209     { \__xeCJK_check_single_end:N #1 #2 }
2210     { \__xeCJK_check_single_end:N #1 #2 ~ }
2211   }

```

```

2212     { \__xeCJK_check_single_end:N #1 #2 ~ }
2213   }
\XeCJK_check_single_equation:NNnW 2214 \cs_new_protected:Npn \xeCJK_check_single_equation:NNnW #1#2#3#4
2215   {
2216     \peek_catcode:NTF \c_math_toggle_token
2217     {
2218       \xeCJK_widow_penalty: \__xeCJK_check_single_end:N #1
2219       \xeCJK_make_node:n { CJK-widow } #2 #4
2220     }
2221     { \__xeCJK_check_single_end:N #1 #2#3#4 }
2222   }

```

在使用 CheckSingle 选项时, 在 `tablists` 宏包定义的 `tabenum` 环境中会出现下面的错误:

```

! Forbidden control sequence found while scanning use of \use_ii:nn.
<inserted text>
\par
1.10 \item

```

原因在于 `tabenum` 实际上是一个 \TeX 对齐环境(`\halign`), `\par` 在其中被重定义为 `\cr`。而在下面 `\token_case_meaning:NnF` 的分支里有对 `\par` 的 `\ifx` 判断。解决办法是将判断用 `\group_align_safe_begin:` 和 `\group_align_safe_end:` 包起来。或者改用原语 `\tex_par:D` 作为判断条件。

```

2223 \cs_new_protected:Npn \xeCJK_check_single_cs:NNn #1#2#3
2224   {
2225     \token_case_meaning:Nof \l_peek_token
2226     { \l__xeCJK_check_single_cs_case_tl }
2227     { \use_iii:nnn }
2228     { \xeCJK_check_single_env:nnNn }
2229     {
2230       \xeCJK_widow_penalty:
2231       \__xeCJK_check_single_end:N #1
2232       \xeCJK_make_node:n { CJK-widow } #2#3
2233     }
2234     { \__xeCJK_check_single_end:N #1 #2#3 }
2235   }
2236 \tl_new:N \l__xeCJK_check_single_cs_case_tl
\XeCJK_check_single_env:nnNn 2237 \cs_new_protected:Npn \xeCJK_check_single_env:nnNn #1#2#3#4
2238   {
2239     \str_case_e:noTF {#4}
2240     { \l__xeCJK_inline_env_case_tl }
2241     {#2}
2242     {#1}
2243     #3 {#4}
2244   }
2245 \prg_generate_conditional_variant:Nnn \str_case_e:nn { no } { TF }

```

```

NewLineCS 2246 \xeCJK_cs_case_keys_define:nNNnn { NewLineCS }
2247   \l__xeCJK_new_line_cs_case_tl \l__xeCJK_new_line_cs_seq
2248   { \use_ii:nnn }
2249   {
2250     \tl_concat:NNN \l__xeCJK_check_single_cs_case_tl
2251     \l__xeCJK_new_line_cs_case_tl \l__xeCJK_env_cs_case_tl
2252   }

```

```

EnvCS 2253 \xeCJK_cs_case_keys_define:nNNnn { EnvCS }
2254   \l__xeCJK_env_cs_case_tl \l__xeCJK_env_cs_seq
2255   { \use:n }
2256   {
2257     \tl_concat:NNN \l__xeCJK_check_single_cs_case_tl
2258     \l__xeCJK_new_line_cs_case_tl \l__xeCJK_env_cs_case_tl
2259   }

```

```

InlineEnv 2260 \keys_define:nn { xeCJK / options }
2261   {
2262     InlineEnv      .code:n =
2263     {
2264       \seq_set_from_clist:Nn \l__xeCJK_inline_env_seq {#1}
2265       \__xeCJK_update_inline_env_case_tl:
2266     } ,
2267     InlineEnv+     .code:n =
2268     {
2269       \clist_map_inline:nn {#1}
2270       {
2271         \seq_if_in:NnF \l__xeCJK_inline_env_seq {##1}
2272         { \seq_put_right:Nn \l__xeCJK_inline_env_seq {##1} }
2273       }
2274       \__xeCJK_update_inline_env_case_tl:
2275     } ,
2276     InlineEnv-     .code:n =
2277     {
2278       \clist_map_inline:nn {#1}
2279       { \seq_remove_all:Nn \l__xeCJK_inline_env_seq {##1} }
2280       \__xeCJK_update_inline_env_case_tl:
2281     }
2282   }
2283 \seq_new:N \l__xeCJK_inline_env_seq

\_xeCJK_update_inline_env_case_tl: 2284 \cs_new_protected:Npn \__xeCJK_update_inline_env_case_tl:
2285   {
2286     \tl_clear:N \l__xeCJK_inline_env_case_tl
2287     \seq_map_inline:Nn \l__xeCJK_inline_env_seq
2288     { \tl_put_right:Nn \l__xeCJK_inline_env_case_tl { {##1} { } } }
2289   }
2290 \tl_new:N \l__xeCJK_inline_env_case_tl

```

5.9 增加 CJK 子分区

```
\g__xeCJK_CJK_sub_class_seq 2291 \seq_new:N \g__xeCJK_CJK_sub_class_seq
```

\xeCJKDeclareSubCJKBlock 声明 CJK 子区范围, #1 为自定义名称, #2 为子区的 Unicode 范围。

```

2292 \NewDocumentCommand \xeCJKDeclareSubCJKBlock
2293   { s > { \TrimSpaces } m m }
2294   {
2295     \xeCJK_declare_sub_char_class:nen { CJK } {#2} {#3}
2296     \IfBooleanT {#1} { \xeCJKResetPunctClass }
2297   }
2298 \@onlypreamble \xeCJKDeclareSubCJKBlock

```

\xeCJKCancelSubCJKBlock 取消和恢复对 CJK 子区的声明。

\xeCJKRestoreSubCJKBlock

```

2299 \bool_new:N \l__xeCJK_sub_cancel_bool
2300 \NewDocumentCommand \xeCJKCancelSubCJKBlock { s m }
2301   {
2302     \bool_if:NF \l__xeCJK_sub_cancel_bool
2303     {
2304       \bool_set_true:N \l__xeCJK_sub_cancel_bool
2305       \__xeCJK_sub_restore_or_cancel:e {#2}
2306       \IfBooleanT {#1} { \xeCJKResetPunctClass }
2307     }
2308   }
2309 \NewDocumentCommand \xeCJKRestoreSubCJKBlock { s m }
2310   {
2311     \bool_if:NT \l__xeCJK_sub_cancel_bool
2312     {
2313       \bool_set_false:N \l__xeCJK_sub_cancel_bool
2314       \__xeCJK_sub_restore_or_cancel:e {#2}
2315       \IfBooleanT {#1} { \xeCJKResetPunctClass }
2316     }
2317   }

```

```

\__xeCJK_sub_restore_or_cancel:n 2318 \cs_new_protected:Npn \__xeCJK_sub_restore_or_cancel:n #1
2319 {
2320   \clist_map_inline:nn {#1}
2321   {
2322     \int_if_exist:cTF { \__xeCJK_class_csname:n { CJK/##1 } }
2323     {
2324       \xeCJK_declare_char_class:nn
2325       { CJK \bool_if:NF \l__xeCJK_sub_cancel_bool { /##1 } }
2326       { \use:c { g__xeCJK_CJK/##1_range_clist } }
2327     }
2328     { \__xeCJK_error:ne { SubBlock-undefined } {##1} }
2329   }
2330 }
2331 \cs_generate_variant:Nn \__xeCJK_sub_restore_or_cancel:n { e }
2332 \__xeCJK_msg_new:nn { SubBlock-undefined }
2333 {
2334   The~CJK~sub~block~`#1'~is~undefined.\\\
2335   Try~to~use~\token_to_str:N \xeCJKDeclareSubCJKBlock \
2336   to~declare~it.
2337 }

\xeCJK_declare_sub_char_class:nnn 2338 \cs_new_protected:Npn \xeCJK_declare_sub_char_class:nnn #1/#2#3
2339 {
2340   \int_if_exist:cF { \__xeCJK_class_csname:n { #1/#2 } }
2341   {
2342     \xeCJK_new_class:n { #1/#2 }
2343     \__xeCJK_set_sub_class_toks:nn {#1} {#2}
2344     \xeCJK_new_sub_key:n {#2}
2345   }
2346   \xeCJK_declare_char_class:nn { #1/#2 } {#3}
2347 }
2348 \cs_generate_variant:Nn \xeCJK_declare_sub_char_class:nnn { ne }

\__xeCJK_set_sub_class_toks:nn 2349 \cs_new_protected:Npn \__xeCJK_set_sub_class_toks:nn #1#2
2350 {
2351   \seq_map_inline:Nn \g__xeCJK_base_class_seq
2352   {
2353     \xeCJK_copy_inter_class_toks:nnnn { #1/#2 } {##1} {#1} {##1}
2354     \xeCJK_copy_inter_class_toks:nnnn {##1} { #1/#2 } {##1} {#1}
2355     \str_if_eq:nnTF {##1} { CJK }
2356     {
2357       \xeCJK_pre_inter_class_toks:nnn {##1} { #1/#2 }
2358       { \__xeCJK_switch_font:nn {#1} {#2} }
2359     }
2360     {
2361       \xeCJK_replace_inter_class_toks:nnnn {##1} { #1/#2 }
2362       { \xeCJK_fallback_symbol:NN }
2363       {
2364         \__xeCJK_switch_font:nn {#1} {#2}
2365         \xeCJK_fallback_symbol:NN
2366       }
2367     }
2368   }
2369   \xeCJK_copy_inter_class_toks:nnnn { #1/#2 } { #1/#2 } {#1} {#1}
2370   \seq_map_inline:Nn \g__xeCJK_CJK_sub_class_seq
2371   {
2372     \xeCJK_copy_inter_class_toks:nnnn { #1/#2 } { #1/##1 } {#1} {#1}
2373     \xeCJK_copy_inter_class_toks:nnnn { #1/##1 } { #1/#2 } {#1} {#1}
2374     \xeCJK_pre_inter_class_toks:nnn { #1/#2 } { #1/##1 }
2375     { \__xeCJK_switch_font:nn {#2} {##1} }
2376     \xeCJK_pre_inter_class_toks:nnn { #1/##1 } { #1/#2 }
2377     { \__xeCJK_switch_font:nn {##1} {#2} }
2378   }
2379   \seq_gput_right:Nn \g__xeCJK_CJK_sub_class_seq {#2}
2380   \__xeCJK_save_CJK_class:n { #1/#2 }
2381   \clist_map_inline:nn { CJK , FullLeft , FullRight , HangulJamo }
2382   {
2383     \xeCJK_pre_inter_class_toks:nnn { #1/#2 } {##1}

```

```

2384     { \__xeCJK_switch_font:nn {#2} {#1} }
2385   }
2386 }

```

5.10 标点处理

`\XeTeXglyphbounds` 可以得到一个字符的左右边距, 用于标点压缩。如果它不可用, 则在文档中只能使用 `plain` 这一标点格式原样输出标点。

```

2387 \cs_if_exist:NF \tex_XeTeXglyphbounds:D
2388 {
2389   \__xeCJK_msg_new:nn { XeTeX-too-old }
2390   {
2391     \token_to_str:N \tex_XeTeXglyphbounds:D \ is~not~defined.\\
2392     CJK~punctuation~kerning~will~not~be~available.\\
2393     You~have~to~update~XeTeX~to~the~version~0.9995.0~or~later.
2394   }
2395   \__xeCJK_error:n { XeTeX-too-old }
2396   \AtEndOfPackage
2397   {
2398     \keys_define:nn { xeCJK / options }
2399     {
2400       PunctStyle .code:n =
2401         { \__xeCJK_error:ne { punct-style-unknown } {#1} }
2402     }
2403     \seq_gclear:N \g__xeCJK_punct_style_seq
2404     \__xeCJK_set_punct_style:n { plain }
2405   }
2406 }

```

`\xeCJKsetwidth` 手动设置参数中的标点符号的宽度。

```

2407 \NewDocumentCommand \xeCJKsetwidth { s m m }
2408 {
2409   \IfBooleanTF {#1}
2410   {
2411     \tl_map_inline:en {#2}
2412     {
2413       \tl_if_exist:cF { g__xeCJK_punct_width/##1/tl }
2414       { \tl_new:c { g__xeCJK_punct_width/##1/tl } }
2415       \tl_gset:cn { g__xeCJK_punct_bound_width/##1/tl } {#3}
2416     }
2417   }
2418   {
2419     \tl_map_inline:en {#2}
2420     {
2421       \tl_if_exist:cF { g__xeCJK_punct_width/##1/tl }
2422       { \tl_new:c { g__xeCJK_punct_width/##1/tl } }
2423       \tl_gset:cn { g__xeCJK_punct_width/##1/tl } {#3}
2424     }
2425   }
2426 }
2427 \@onlypreamble \xeCJKsetwidth
2428 \cs_generate_variant:Nn \tl_map_inline:nn { e }

```

`\xeCJKsetkern` 手动设置相邻标点的距离。

```

2429 \NewDocumentCommand \xeCJKsetkern { m m m }
2430 { \tl_gset:cn { g__xeCJK_punct/kern/#1/#2/tl } {#3} }
2431 \@onlypreamble \xeCJKsetkern

\c__xeCJK_left_tl 2432 \tl_const:Nn \c__xeCJK_left_tl { left }
\c__xeCJK_right_tl 2433 \tl_const:Nn \c__xeCJK_right_tl { right }

```

AllowBreakBetweenPuncts 相关选项声明。

```

KaiMingPunct
LongPunct
NoBreakLongPunct
MiddlePunct
PunctWidth
PunctBoundWidth
RubberPunctSkip
2434 \keys_define:nn { xeCJK / options }
2435 {
2436   AllowBreakBetweenPuncts .choice: ,
2437   AllowBreakBetweenPuncts / true .code:n =
2438     {
2439       \bool_set_true:N \l__xeCJK_punct_breakable_bool
2440       \cs_set_eq:NN \xeCJK_punct_kern:NN \__xeCJK_punct_breakable_kern:NN
2441       \cs_set_eq:NN \__xeCJK_punct_bound_kern:N
2442         \__xeCJK_punct_bound_breakable_kern:N
2443     } ,
2444   AllowBreakBetweenPuncts / false .code:n =
2445     {
2446       \bool_set_false:N \l__xeCJK_punct_breakable_bool
2447       \cs_set_eq:NN \xeCJK_punct_kern:NN \__xeCJK_punct_kern:NN
2448       \cs_set_eq:NN \__xeCJK_punct_bound_kern:N
2449         \__xeCJK_nobreak_hskip:N
2450     } ,
2451   AllowBreakBetweenPuncts .default:n = { true } ,
2452   KaiMingPunct .code:n = { \__xeCJK_set_special_punct:nn { mixed_width } {#1} } ,
2453   KaiMingPunct+ .code:n = { \__xeCJK_add_special_punct:nn { mixed_width } {#1} } ,
2454   KaiMingPunct- .code:n = { \__xeCJK_sub_special_punct:nn { mixed_width } {#1} } ,
2455   LongPunct .code:n = { \__xeCJK_set_special_punct:nn { long } {#1} } ,
2456   LongPunct+ .code:n = { \__xeCJK_add_special_punct:nn { long } {#1} } ,
2457   LongPunct- .code:n = { \__xeCJK_sub_special_punct:nn { long } {#1} } ,
2458   NoBreakLongPunct .code:n = { \__xeCJK_set_special_punct:nn { nobreak_long } {#1} } ,
2459   NoBreakLongPunct+ .code:n = { \__xeCJK_add_special_punct:nn { nobreak_long } {#1} } ,
2460   NoBreakLongPunct- .code:n = { \__xeCJK_sub_special_punct:nn { nobreak_long } {#1} } ,
2461   MiddlePunct .code:n = { \__xeCJK_set_special_punct:nn { middle } {#1} } ,
2462   MiddlePunct+ .code:n = { \__xeCJK_add_special_punct:nn { middle } {#1} } ,
2463   MiddlePunct- .code:n = { \__xeCJK_sub_special_punct:nn { middle } {#1} } ,
2464   PunctWidth .tl_gset:N = \g__xeCJK_punct_width_tl ,
2465   PunctBoundWidth .tl_gset:N = \g__xeCJK_punct_bound_width_tl ,
2466   PunctWidth .value_required:n = true ,
2467   PunctBoundWidth .value_required:n = true ,
2468   RubberPunctSkip .choice: ,
2469   RubberPunctSkip .default:n = { true } ,
2470   RubberPunctSkip / true .code:n =
2471     { \cs_set_eq:NN \__xeCJK_use_dim_or_skip:nNN \__xeCJK_use_punct_skip:nNN } ,
2472   RubberPunctSkip / plus .code:n =
2473     { \cs_set_eq:NN \__xeCJK_use_dim_or_skip:nNN \__xeCJK_use_punct_skip_plus:nNN } ,
2474   RubberPunctSkip / minus .code:n =
2475     { \cs_set_eq:NN \__xeCJK_use_dim_or_skip:nNN \__xeCJK_use_punct_skip_minus:nNN } ,
2476   RubberPunctSkip / false .code:n =
2477     { \cs_set_eq:NN \__xeCJK_use_dim_or_skip:nNN \__xeCJK_use_punct_dim:nNN }
2478 }
2479 \bool_new:N \l__xeCJK_punct_breakable_bool

```

相关选项定义的辅助函数。

```

2480 \clist_new:N \g__xeCJK_special_punct_clist
2481 \clist_gset:Nn \g__xeCJK_special_punct_clist { mixed_width , long , nobreak_long , middle }
2482 \cs_new:Npn \__xeCJK_special_punct_seq:n #1 { g__xeCJK_special_punct_#1_seq }
2483 \cs_new:Npn \__xeCJK_special_punct_tl:nN #1#2 { g__xeCJK_special_punct_#1_#2_tl }
2484 \clist_map_inline:Nn \g__xeCJK_special_punct_clist
2485   { \seq_new:c { \__xeCJK_special_punct_seq:n {#1} } }
2486 \cs_new_protected:Npn \__xeCJK_set_special_punct:nn #1#2
2487   {
2488     \seq_map_inline:cn { \__xeCJK_special_punct_seq:n {#1} }
2489     { \cs_undefine:c { \__xeCJK_special_punct_tl:nN {#1} {##1} } }
2490     \seq_gclear:c { \__xeCJK_special_punct_seq:n {#1} }
2491     \tl_map_inline:en {#2}
2492     {
2493       \tl_new:c { \__xeCJK_special_punct_tl:nN {#1} {##1} }
2494       \seq_gput_right:cn { \__xeCJK_special_punct_seq:n {#1} } {##1}
2495     }
2496   }

```

```

2497 \cs_new_protected:Npn \__xeCJK_add_special_punct:nn #1#2
2498 {
2499   \tl_map_inline:en {#2}
2500   {
2501     \seq_if_in:cnF { \__xeCJK_special_punct_seq:n {#1} } {##1}
2502     {
2503       \tl_new:c { \__xeCJK_special_punct_tl:nN {#1} {##1} }
2504       \seq_gput_right:cn { \__xeCJK_special_punct_seq:n {#1} } {##1}
2505     }
2506   }
2507 }
2508 \cs_new_protected:Npn \__xeCJK_sub_special_punct:nn #1#2
2509 {
2510   \tl_map_inline:en {#2}
2511   {
2512     \cs_undefine:c { \__xeCJK_special_punct_tl:nN {#1} {##1} }
2513     \seq_gremove_all:cn { \__xeCJK_special_punct_seq:n {#1} } {##1}
2514   }
2515 }

```

判断一个标点符号是否为全角右标点和长标点符号。

```

2516 \prg_new_conditional:Npnn \__xeCJK_punct_if_right:N #1 { p , T , F , TF }
2517 {
2518   \if_int_compare:w \xeCJK_token_value_class:N #1 =
2519     \xeCJK_class_num:n { FullRight }
2520   \prg_return_true: \else: \prg_return_false: \fi:
2521 }
2522 \clist_map_inline:Nn \g__xeCJK_special_punct_clist
2523 {
2524   \exp_args:Nc
2525   \prg_new_conditional:Npnn { \__xeCJK_punct_if_#1:N } ##1 { p , T , F , TF }
2526   {
2527     \if_cs_exist:w \__xeCJK_special_punct_tl:nN {#1} {##1} \cs_end:
2528     \prg_return_true: \else: \prg_return_false: \fi:
2529   }
2530 }

```

一些用于记录的辅助函数。

```

2531 \cs_new:Npn \__xeCJK_punct_csname:n #1
2532 { c__xeCJK_\l_xeCJK_current_punct_font_tl/\l_xeCJK_punct_style_tl/#1/tl }
2533 \cs_new:Npn \__xeCJK_use_punct_dim:nN #1#2
2534 { \use:c { \__xeCJK_punct_csname:n { dim/#1/#2 } } }
2535 \cs_new:Npn \__xeCJK_use_punct_dim:nNN #1#2#3
2536 { \use:c { \__xeCJK_punct_csname:n { dim/#1/#2/#3 } } }
2537 \cs_new:Npn \__xeCJK_use_punct_skip:nNN #1#2#3
2538 { \use:c { \__xeCJK_punct_csname:n { skip/#1/#2/#3 } } }
2539 \cs_new:Npn \__xeCJK_use_punct_skip_plus:nNN #1#2#3
2540 { \use:c { \__xeCJK_punct_csname:n { skip/plus/#1/#2/#3 } } }
2541 \cs_new:Npn \__xeCJK_use_punct_skip_minus:nNN #1#2#3
2542 { \use:c { \__xeCJK_punct_csname:n { skip/minus/#1/#2/#3 } } }
2543 \cs_new_protected:Npn \__xeCJK_save_punct_dim:nNn #1#2
2544 { \__xeCJK_save_punct_width_aux:nNnn { dim } {#1} { #1/#2 } }
2545 \cs_new_protected:Npn \__xeCJK_save_punct_dim:nNNn #1#2#3
2546 { \__xeCJK_save_punct_width_aux:nNnn { dim } {#1} { #1/#2/#3 } }
2547 \cs_new_protected:Npn \__xeCJK_save_punct_skip:nNNn #1#2#3#4
2548 {
2549   \__xeCJK_save_punct_width_aux:nNnn { skip } {#1} { #1/#2/#3 } {#4}
2550   \__xeCJK_save_punct_width_aux:nNnn { skip } {#1} { plus/#1/#2/#3 } {#4}
2551   \__xeCJK_save_punct_width_aux:nNnn { skip } {#1} { minus/#1/#2/#3 } {#4}
2552 }
2553 \cs_new_protected:Npn \__xeCJK_save_punct_skip:nNNnn #1#2#3#4#5#6
2554 {
2555   \exp_last_unbraced:Ne
2556   \__xeCJK_save_punct_skip_aux:nNnnn
2557   {
2558     {#1}

```

```

2559     { #1/#2/#3 }
2560     { \dim_eval:n {#4} }
2561     { \dim_max:nn { \c_zero_dim } {#5} }
2562     { \dim_max:nn { \c_zero_dim } {#6} }
2563   }
2564 }
2565 \cs_new_protected:Npn \__xeCJK_save_punct_skip_aux:nnnn #1#2#3#4#5
2566 {
2567   \__xeCJK_save_punct_width_aux:nnnn { skip } {#1}
2568   {#2} { #3 ~ plus ~ #4 ~ minus ~ #5 ~ }
2569   \__xeCJK_save_punct_width_aux:nnnn { skip } {#1}
2570   { plus/#2 } { #3 ~ plus ~ #4 ~ }
2571   \__xeCJK_save_punct_width_aux:nnnn { skip } {#1}
2572   { minus/#2 } { #3 ~ minus ~ #5 ~ }
2573 }
2574 \cs_new_protected:Npn \__xeCJK_save_punct_width_aux:nnnn #1#2#3#4
2575 {
2576   \__xeCJK_save_punct_width_aux:cen
2577   { \__xeCJK_punct_csname:n { #1/#3 } }
2578   { \use:c { #1_eval:n } {#4} }
2579   {#2}
2580 }
2581 \cs_new_protected:Npn \__xeCJK_save_punct_width_aux:Nnn #1#2#3
2582 {
2583   \tl_const:Nn #1 {#2}
2584   \str_if_eq:nnT {#3} { glue }
2585   { \prop_gput:Nnn \g__xeCJK_punct_skip_prop {#2} { } }
2586 }
2587 \prop_new:N \g__xeCJK_punct_skip_prop
2588 \prop_gput:Non \g__xeCJK_punct_skip_prop { \skip_use:N \c_zero_skip } { }
2589 \cs_generate_variant:Nn \__xeCJK_save_punct_width_aux:Nnn { ce }
2590 \cs_new_eq:NN \__xeCJK_use_dim_or_skip:nNN \__xeCJK_use_punct_skip:nNN

```

定义标点处理模板。

```

2591 \DeclareObjectType { xeCJK / punctuation } { 0 }
2592 \DeclareTemplateInterface { xeCJK / punctuation } { basic } { 0 }
2593 {
2594   enabled-global-setting : boolean = true ,
2595   fixed-punct-width     : length  = \c_max_dim ,
2596   fixed-punct-ratio     : real    = \c_one_fp ,
2597   mixed-punct-width     : length  = \KeyValue { fixed-punct-width } ,
2598   mixed-punct-ratio     : real    = \KeyValue { fixed-punct-ratio } ,
2599   middle-punct-width    : length  = \KeyValue { fixed-punct-width } ,
2600   middle-punct-ratio    : real    = \KeyValue { fixed-punct-ratio } ,
2601   fixed-margin-width    : length  = \c_max_dim ,
2602   fixed-margin-ratio    : real    = \c_one_fp ,
2603   mixed-margin-width    : length  = \KeyValue { fixed-margin-width } ,
2604   mixed-margin-ratio    : real    = \KeyValue { fixed-margin-ratio } ,
2605   middle-margin-width   : length  = \KeyValue { fixed-margin-width } ,
2606   middle-margin-ratio   : real    = \KeyValue { fixed-margin-ratio } ,
2607   bound-punct-width     : length  = \c_max_dim ,
2608   bound-punct-ratio     : real    = \c_nan_fp ,
2609   bound-margin-width    : length  = \c_max_dim ,
2610   bound-margin-ratio    : real    = \c_zero_fp ,
2611   enabled-hanging       : boolean = false ,
2612   add-min-bound-to-margin : boolean = false ,
2613   optimize-margin       : boolean = false ,
2614   margin-minimum        : length  = \c_zero_dim ,
2615   enabled-kerning       : boolean = true ,
2616   min-bound-to-kerning   : boolean = false ,
2617   kerning-total-width   : length  = \c_max_dim ,
2618   kerning-total-ratio   : real    = 0.75 ,
2619   optimize-kerning      : boolean = false ,
2620   same-align-margin     : length  = \c_max_dim ,
2621   same-align-ratio      : real    = \c_nan_fp ,
2622   different-align-margin : length  = \c_max_dim ,

```

```

2623 different-align-ratio : real = \c_nan_fp ,
2624 kerning-margin-width : length = \c_max_dim ,
2625 kerning-margin-ratio : real = \c_one_fp ,
2626 kerning-margin-minimum : length = \c_zero_dim
2627 }

2628 \DeclareTemplateCode { xeCJK / punctuation } { basic } { 0 }
2629 {
2630 enabled-global-setting = \l__xeCJK_enabled_global_setting_bool ,
2631 fixed-punct-width = \l__xeCJK_fixed_punct_width_dim ,
2632 fixed-punct-ratio = \l__xeCJK_fixed_punct_ratio_fp ,
2633 mixed-punct-width = \l__xeCJK_mixed_punct_width_dim ,
2634 mixed-punct-ratio = \l__xeCJK_mixed_punct_ratio_fp ,
2635 middle-punct-width = \l__xeCJK_middle_punct_width_dim ,
2636 middle-punct-ratio = \l__xeCJK_middle_punct_ratio_fp ,
2637 fixed-margin-width = \l__xeCJK_fixed_margin_width_dim ,
2638 fixed-margin-ratio = \l__xeCJK_fixed_margin_ratio_fp ,
2639 mixed-margin-width = \l__xeCJK_mixed_margin_width_dim ,
2640 mixed-margin-ratio = \l__xeCJK_mixed_margin_ratio_fp ,
2641 middle-margin-width = \l__xeCJK_middle_margin_width_dim ,
2642 middle-margin-ratio = \l__xeCJK_middle_margin_ratio_fp ,
2643 bound-punct-width = \l__xeCJK_bound_punct_width_dim ,
2644 bound-punct-ratio = \l__xeCJK_bound_punct_ratio_fp ,
2645 bound-margin-width = \l__xeCJK_bound_margin_width_dim ,
2646 bound-margin-ratio = \l__xeCJK_bound_margin_ratio_fp ,
2647 enabled-hanging = \l__xeCJK_enabled_hanging_bool ,
2648 add-min-bound-to-margin = \l__xeCJK_add_min_bound_to_margin_bool ,
2649 optimize-margin = \l__xeCJK_optimize_margin_bool ,
2650 margin-minimum = \l__xeCJK_margin_minimum_dim ,
2651 enabled-kerning = \l__xeCJK_enabled_kerning_bool ,
2652 min-bound-to-kerning = \l__xeCJK_min_bound_to_kerning_bool ,
2653 kerning-total-width = \l__xeCJK_kerning_total_width_dim ,
2654 kerning-total-ratio = \l__xeCJK_kerning_total_ratio_fp ,
2655 optimize-kerning = \l__xeCJK_optimize_kerning_bool ,
2656 same-align-margin = \l__xeCJK_same_align_margin_dim ,
2657 same-align-ratio = \l__xeCJK_same_align_ratio_fp ,
2658 different-align-margin = \l__xeCJK_different_align_margin_dim ,
2659 different-align-ratio = \l__xeCJK_different_align_ratio_fp ,
2660 kerning-margin-width = \l__xeCJK_kerning_margin_width_dim ,
2661 kerning-margin-ratio = \l__xeCJK_kerning_margin_ratio_fp ,
2662 kerning-margin-minimum = \l__xeCJK_kerning_margin_minimum_dim
2663 }
2664 { \AssignTemplateKeys }

\XeCJK_get_punct_bounds:NN #1 为 \c__xeCJK_left_tl 或 \c__xeCJK_right_tl, #2 为标点符号。

2665 \cs_new_protected:Npn \XeCJK_get_punct_bounds:NN #1#2
2666 {
2667 \tl_if_exist:cF { \__xeCJK_punct_csname:n { dim/glue/#1/#2 } }
2668 { \__xeCJK_get_punct_bounds_aux:NN #1 #2 }
2669 }
2670 \cs_new_protected:Npn \__xeCJK_get_punct_bounds_aux:NN
2671 {
2672 \tl_if_eq:NNTF \l_xeCJK_punct_style_tl \c__xeCJK_punct_style_plain_tl
2673 { \__xeCJK_save_punct_margin_plain:NN }
2674 { \__xeCJK_save_punct_margin:NN }
2675 }
2676 \cs_new_protected:Npn \XeCJK_get_punct_bounds:No
2677 { \exp_last_unbraced:NNo \XeCJK_get_punct_bounds:NN }
2678 \cs_new_protected:Npn \__xeCJK_save_punct_margin_plain:NN #1#2
2679 {
2680 \__xeCJK_save_punct_dim:nNNn { glue } #1 #2 { \c_zero_dim }
2681 \__xeCJK_save_punct_dim:nNNn { offset } #1 #2 { \c_zero_dim }
2682 \__xeCJK_save_punct_dim:nNNn { margin } #1 #2 { \c_zero_dim }
2683 \__xeCJK_save_punct_dim:nNNn { rule } \c__xeCJK_left_tl {#2} { \c_zero_dim }
2684 \__xeCJK_save_punct_dim:nNNn { rule } \c__xeCJK_right_tl {#2} { \c_zero_dim }
2685 \__xeCJK_save_punct_dim:nNNn { bound } \c__xeCJK_left_tl {#2} { \c_zero_dim }
2686 \__xeCJK_save_punct_dim:nNNn { bound } \c__xeCJK_right_tl {#2} { \c_zero_dim }

```

```

2687 \__xeCJK_save_punct_skip:nNNn { glue } #1 #2 { \c_zero_skip }
2688 }
2689 \cs_new_protected:Npn \__xeCJK_save_punct_margin:NN #1#2
2690 {
2691 \group_begin:
2692 \xeCJK_select_punct_font:
2693 \xeCJK_fallback_punct_symbol:NN
2694 \xeCJK_calc_punct_dimen:N #2
2695 \group_end:
2696 \dim_set:Nn \l__xeCJK_bound_dim
2697 { \__xeCJK_use_punct_dim:nNN { bound } #1 #2 }
2698 \tl_if_eq:NNTF #1 \c__xeCJK_right_tl
2699 { \tl_set_eq:NN \l__xeCJK_reverse_tl \c__xeCJK_left_tl }
2700 { \tl_set_eq:NN \l__xeCJK_reverse_tl \c__xeCJK_right_tl }
2701 \dim_set:Nn \l__xeCJK_reverse_bound_dim
2702 { \__xeCJK_use_punct_dim:nNN { bound } \l__xeCJK_reverse_tl #2 }
2703 \UseInstance { xeCJK / punctuation } { \l_xeCJK_punct_style_tl }
2704 \xeCJK_punct_margin_process:NN #1 #2
2705 \xeCJK_punct_offset_process:NN #1 #2
2706 \__xeCJK_punct_if_long:NT #2
2707 { \__xeCJK_long_punct_kerning:N #2 }
2708 }
2709 \tl_new:N \l__xeCJK_reverse_tl
2710 \dim_new:N \l__xeCJK_bound_dim
2711 \dim_new:N \l__xeCJK_reverse_bound_dim

```

`__xeCJK_long_punct_kerning:N` 相同长标点压缩。对于破折号，计算两标点之间的空白，保证它中间不被断开。注意，破折号的边界可能为负值(比如方正新书宋)，此时不必压缩。

```

2712 \cs_new_protected:Npn \__xeCJK_long_punct_kerning:N #1
2713 {
2714 \dim_set:Nn \l__xeCJK_tmp_dim
2715 {
2716 \dim_max:nn
2717 { \l__xeCJK_bound_dim + \l__xeCJK_reverse_bound_dim }
2718 { \c_zero_dim }
2719 }
2720 \__xeCJK_save_punct_dim:nNNn { bound_width } #1 #1 { \l__xeCJK_tmp_dim }
2721 \dim_set:Nn \l__xeCJK_tmp_dim
2722 {
2723 \str_case:nnTF {#1}
2724 { { ~~~~2025 } { } { ~~~~2026 } { } }
2725 { \c_zero_dim }
2726 { -\l__xeCJK_tmp_dim }
2727 }
2728 \__xeCJK_save_punct_dim:nNNn { kern } #1 #1 { \l__xeCJK_tmp_dim }
2729 \__xeCJK_save_punct_skip:nNNn { kern } #1 #1 { \l__xeCJK_tmp_dim }
2730 \dim_add:Nn \l__xeCJK_tmp_dim
2731 { \dim_max:nn { \l__xeCJK_bound_dim } { \c_zero_dim } }
2732 \__xeCJK_save_punct_dim:nNNn { bound_kern } #1 #1 { \l__xeCJK_tmp_dim }
2733 \__xeCJK_save_punct_skip:nNNn { bound_kern } #1 #1 { \l__xeCJK_tmp_dim }
2734 }

```

`\xeCJK_get_punct_kerning:NN` 标点压缩。

```

2735 \cs_new_protected:Npn \xeCJK_get_punct_kerning:NN #1#2
2736 {
2737 \tl_if_exist:cF { \__xeCJK_punct_csname:n { dim/kern/#1/#2 } }
2738 {
2739 \tl_if_eq:NNTF \l_xeCJK_punct_style_tl \c__xeCJK_punct_style_plain_tl
2740 { \__xeCJK_save_punct_kerning_plain:NN }
2741 { \__xeCJK_save_punct_kerning:NN }
2742 #1 #2
2743 }
2744 }
2745 \cs_new_protected:Npn \xeCJK_get_punct_kerning:oN
2746 { \exp_after:wN \xeCJK_get_punct_kerning:NN }

```

```

2747 \cs_new_protected:Npn \__xeCJK_save_punct_kerning_plain:NN #1#2
2748 {
2749   \__xeCJK_save_punct_dim:nNNn { kern } #1 #2 { \c_zero_dim }
2750   \__xeCJK_save_punct_dim:nNNn { bound_kern } #1 #2 { \c_zero_dim }
2751   \__xeCJK_save_punct_dim:nNNn { bound_width } #1 #2 { \c_zero_dim }
2752   \__xeCJK_save_punct_skip:nNNn { kern } #1 #2 { \c_zero_skip }
2753   \__xeCJK_save_punct_skip:nNNn { bound_kern } #1 #2 { \c_zero_skip }
2754 }
2755 \cs_new_protected:Npn \__xeCJK_save_punct_kerning:NN
2756 {
2757   \UseInstance { xeCJK / punctuation } { \l_xeCJK_punct_style_tl }
2758   \xeCJK_punct_kerning_process:NN
2759 }
\xeCJK_punct_margin_process:NN 2760 \cs_new_protected:Npn \xeCJK_punct_margin_process:NN #1#2
2761 {
2762   \dim_set:Nn \l__xeCJK_tmp_dim
2763   {
2764     \bool_if:NTF \l__xeCJK_enabled_global_setting_bool
2765     {
2766       \cs_if_exist_use:cF { g__xeCJK_punct_width/#2/tl }
2767       {
2768         \tl_if_empty:NTF \g__xeCJK_punct_width_tl
2769         { \__xeCJK_calc_punct_width:N #2 }
2770         { \g__xeCJK_punct_width_tl }
2771       }
2772     }
2773     { \__xeCJK_calc_punct_width:N #2 }
2774   }
2775   \dim_set:Nn \l__xeCJK_margin_dim
2776   {
2777     \dim_max:nn
2778     { \l__xeCJK_margin_minimum_dim }
2779     {
2780       \dim_compare:nNnTF \l__xeCJK_tmp_dim < \c_max_dim
2781       {
2782         \__xeCJK_punct_if_middle:NTF #2
2783         {
2784           ( \l__xeCJK_tmp_dim
2785             - ( \__xeCJK_use_punct_dim:nN { dimen } #2 )
2786           ) / 2
2787         }
2788         {
2789           \bool_if:NTF \l__xeCJK_optimize_margin_bool
2790           {
2791             \dim_max:nn
2792             {
2793               \dim_min:nn
2794               { \l__xeCJK_bound_dim }
2795               { \l__xeCJK_reverse_bound_dim }
2796             }
2797           }
2798           { \use:n }
2799           {
2800             \l__xeCJK_tmp_dim
2801             - \l__xeCJK_reverse_bound_dim
2802             - ( \__xeCJK_use_punct_dim:nN { dimen } #2 )
2803           }
2804         }
2805       }
2806     }
2807     \bool_if:NTF \l__xeCJK_optimize_margin_bool
2808     { \dim_min:nn { \l__xeCJK_bound_dim } }
2809     { \use:n }
2810     { \__xeCJK_calc_margin_width:N #2 }
2811   }
2812 }
2813 }

```

```

2814 \l__xeCJK_save_punct_dim:nNn { margin } #1 #2 { \l__xeCJK_margin_dim }
2815 }
2816 \dim_new:N \l__xeCJK_margin_dim

```

```

\__xeCJK_calc_punct_width:N 2817 \cs_new:Npn \__xeCJK_calc_punct_width:N #1
2818 {
2819 \__xeCJK_punct_if_middle:NTF #1
2820 { \__xeCJK_punct_width_or_ratio:nN { middle } }
2821 {
2822 \__xeCJK_punct_if_mixed_width:NTF #1
2823 { \__xeCJK_punct_width_or_ratio:nN { mixed } }
2824 { \__xeCJK_punct_width_or_ratio:nN { fixed } }
2825 }
2826 #1
2827 }

```

```

\__xeCJK_calc_margin_width:N 2828 \cs_new:Npn \__xeCJK_calc_margin_width:N #1
2829 {
2830 \__xeCJK_punct_if_middle:NTF #1
2831 {
2832 \dim_compare:nNnTF \l__xeCJK_middle_margin_width_dim < \c_max_dim
2833 { \l__xeCJK_middle_margin_width_dim }
2834 {
2835 \__xeCJK_dim_ratio:Nn \l__xeCJK_middle_margin_ratio_fp
2836 { ( \l__xeCJK_bound_dim + \l__xeCJK_reverse_bound_dim ) / 2 }
2837 }
2838 }
2839 {
2840 \__xeCJK_punct_if_mixed_width:NTF #1
2841 { \__xeCJK_margin_width_or_ratio:n { mixed } }
2842 { \__xeCJK_margin_width_or_ratio:n { fixed } }
2843 }
2844 }

```

```

\__xeCJK_dim_ratio:Nn 2845 \cs_new:Npn \__xeCJK_dim_ratio:Nn #1#2
2846 { \fp_to_dim:n { #1 \dim_to_fp:n {#2} } }
2847 \cs_generate_variant:Nn \__xeCJK_dim_ratio:Nn { c }

```

```

\xeCJK_punct_offset_process:NN 2848 \cs_new_protected:Npn \xeCJK_punct_offset_process:NN #1#2
2849 {
2850 \dim_set:Nn \l__xeCJK_tmp_dim
2851 {
2852 \bool_if:NTF \l__xeCJK_enabled_global_setting_bool
2853 {
2854 \cs_if_exist_use:cF { g__xeCJK_punct_bound_width/#2/tl }
2855 {
2856 \tl_if_empty:NTF \g__xeCJK_punct_bound_width_tl
2857 { \__xeCJK_punct_width_or_ratio:nN { bound } #2 }
2858 { \g__xeCJK_punct_bound_width_tl }
2859 }
2860 }
2861 { \__xeCJK_punct_width_or_ratio:nN { bound } #2 }
2862 }
2863 \dim_set:Nn \l__xeCJK_tmp_dim
2864 {
2865 \bool_if:NTF \l__xeCJK_enabled_hanging_bool
2866 { \use:n }
2867 { \dim_max:nn { \l__xeCJK_margin_minimum_dim } }
2868 {
2869 \dim_compare:nNnTF \l__xeCJK_tmp_dim < \c_max_dim
2870 {
2871 \__xeCJK_punct_if_middle:NTF #2
2872 {
2873 \l__xeCJK_tmp_dim
2874 - \l__xeCJK_margin_dim
2875 - ( \__xeCJK_use_punct_dim:nN { dimen } #2 )
2876 }
2877 {

```

```

2878         \l__xeCJK_tmp_dim
2879         - \l__xeCJK_reverse_bound_dim
2880         - ( \__xeCJK_use_punct_dim:nN { dimen } #2 )
2881     }
2882 }
2883 {
2884     \bool_if:NTF \l__xeCJK_optimize_margin_bool
2885     { \dim_min:nn { \l__xeCJK_bound_dim } }
2886     { \use:n }
2887     { \__xeCJK_margin_width_or_ratio:n { bound } }
2888 }
2889 }
2890 }
2891 \__xeCJK_save_punct_dim:nNn { offset } #1 #2
2892 { \l__xeCJK_tmp_dim }
2893 \__xeCJK_save_punct_dim:nNn { rule } #1 #2
2894 { \l__xeCJK_tmp_dim - \l__xeCJK_bound_dim }
2895 \__xeCJK_save_punct_dim:nNn { rule } \l__xeCJK_reverse_tl #2
2896 { \l__xeCJK_tmp_dim - \l__xeCJK_reverse_bound_dim }
2897 \__xeCJK_save_punct_dim:nNn { glue } #1 #2
2898 { \l__xeCJK_margin_dim - \l__xeCJK_tmp_dim }
2899 \__xeCJK_save_punct_skip:nNnnn { glue } #1 #2
2900 { \l__xeCJK_margin_dim - \l__xeCJK_tmp_dim }
2901 {
2902     \__xeCJK_punct_if_middle:NTF #2
2903     {
2904         ( \__xeCJK_use_punct_dim:nN { width } #2 -
2905           \__xeCJK_use_punct_dim:nN { dimen } #2 ) / 2
2906         - \l__xeCJK_margin_dim
2907     }
2908     { \l__xeCJK_bound_dim - \l__xeCJK_margin_dim }
2909 }
2910 {
2911     \__xeCJK_punct_if_middle:NTF #2
2912     { \l__xeCJK_margin_dim / 2 }
2913     { \l__xeCJK_margin_dim - \l__xeCJK_reverse_bound_dim }
2914 }
2915 }

```

```

\__xeCJK_punct_width_or_ratio:n 2916 \cs_new:Npn \__xeCJK_punct_width_or_ratio:n #1#2
2917 {
2918     \dim_compare:nNnTF { \use:c { l__xeCJK_#1_punct_width_dim } } < \c_max_dim
2919     { \use:c { l__xeCJK_#1_punct_width_dim } }
2920     {
2921         \fp_if_nan:nTF { \use:c { l__xeCJK_#1_punct_ratio_fp } }
2922         { \c_max_dim }
2923         {
2924             \__xeCJK_dim_ratio:cn
2925             { l__xeCJK_#1_punct_ratio_fp }
2926             { \__xeCJK_use_punct_dim:nN { width } #2 }
2927         }
2928     }
2929 }

```

```

\__xeCJK_margin_width_or_ratio:n 2930 \cs_new:Npn \__xeCJK_margin_width_or_ratio:n #1
2931 {
2932     \dim_compare:nNnTF { \use:c { l__xeCJK_#1_margin_width_dim } } < \c_max_dim
2933     { \use:c { l__xeCJK_#1_margin_width_dim } }
2934     {
2935         \__xeCJK_dim_ratio:cn
2936         { l__xeCJK_#1_margin_ratio_fp }
2937         { \l__xeCJK_bound_dim }
2938     }
2939     \bool_if:NT \l__xeCJK_add_min_bound_to_margin_bool
2940     { + \dim_min:nn \l__xeCJK_bound_dim \l__xeCJK_reverse_bound_dim }
2941 }

```

\xeCJK_punct_kerning_process:NN 当标点之一为长标点时,不必进行压缩。

```

2942 \cs_new_protected:Npn \xeCJK_punct_kerning_process:NN #1#2
2943 {
2944   \dim_set:Nn \l__xeCJK_margin_dim
2945     { \__xeCJK_original_kerning_margin:NN #1 #2 }
2946   \dim_set:Nn \l__xeCJK_minimum_bound_dim
2947     { \__xeCJK_punct_min_bound:NN #1 #2 }
2948   \__xeCJK_punct_if_long:NTF #1
2949     { \bool_set_false:N \l__xeCJK_enabled_kerning_bool }
2950     {
2951       \__xeCJK_punct_if_long:NT #2
2952       { \bool_set_false:N \l__xeCJK_enabled_kerning_bool }
2953     }
2954   \dim_set:Nn \l__xeCJK_kerning_margin_dim
2955     {
2956       \bool_if:NTF \l__xeCJK_enabled_global_setting_bool
2957         {
2958           \cs_if_exist_use:cF { g__xeCJK_punct/kern/#1/#2/tl }
2959             { \__xeCJK_punct_kerning_process_aux:NN #1 #2 }
2960         }
2961         { \__xeCJK_punct_kerning_process_aux:NN #1 #2 }
2962     }
2963   \__xeCJK_save_kerning:nnNN { kern } { bound } #1 #2
2964   \__xeCJK_save_punct_dim:nNNn { bound_width } #1 #2
2965     { \l__xeCJK_kerning_margin_dim - \l__xeCJK_tmp_dim }
2966   \__xeCJK_punct_if_right:NTF #1
2967     {
2968       \__xeCJK_punct_if_right:NTF #2
2969         { \__xeCJK_save_kerning:nnnNN { bound_kern } { offset } { bound } }
2970         { \__xeCJK_save_kerning:nnNN { bound_kern } { offset } }
2971     }
2972     {
2973       \__xeCJK_punct_if_right:NTF #2
2974         { \__xeCJK_save_kerning:nnNN { bound_kern } { bound } }
2975         { \__xeCJK_save_kerning:nnnNN { bound_kern } { bound } { offset } }
2976     }
2977     #1 #2
2978 }
2979 \cs_new:Npn \__xeCJK_punct_kerning_process_aux:NN #1#2
2980 {
2981   \bool_if:NTF \l__xeCJK_enabled_kerning_bool
2982     { \__xeCJK_calc_kerning_margin:NN #1 #2 }
2983     { \l__xeCJK_margin_dim }
2984 }
2985 \dim_new:N \l__xeCJK_minimum_bound_dim
2986 \dim_new:N \l__xeCJK_kerning_margin_dim

```

__xeCJK_save_kerning:nnNN 相邻两个标点符号的间距能伸长到原始空白(未压缩时的状态),能收缩到较小边距。

```

2987 \cs_new_protected:Npn \__xeCJK_save_kerning:nnNN #1#2
2988 { \__xeCJK_save_kerning:nnnNN {#1} {#2} {#2} }
2989 \cs_new_protected:Npn \__xeCJK_save_kerning:nnnNN #1#2#3#4#5
2990 {
2991   \dim_set:Nn \l__xeCJK_tmp_dim
2992     {
2993       \l__xeCJK_kerning_margin_dim
2994       - ( \__xeCJK_use_punct_dim:nNN {#2} \c__xeCJK_right_tl #4 )
2995       - ( \__xeCJK_use_punct_dim:nNN {#3} \c__xeCJK_left_tl #5 )
2996     }
2997   \__xeCJK_save_punct_dim:nNNn {#1} #4 #5 { \l__xeCJK_tmp_dim }
2998   \__xeCJK_save_punct_skip:nNNnnn {#1} #4 #5
2999     { \l__xeCJK_tmp_dim }
3000     { \l__xeCJK_margin_dim - \l__xeCJK_kerning_margin_dim }
3001     { \l__xeCJK_kerning_margin_dim - \l__xeCJK_minimum_bound_dim }
3002 }

```

```

\__xeCJK_original_kerning_margin:NN 相邻两个标点符号之间的本来空白宽度。
3003 \cs_new:Npn \__xeCJK_original_kerning_margin:NN #1#2
3004 {
3005   \dim_eval:n
3006   {
3007     \__xeCJK_use_punct_dim:nNN
3008     { \__xeCJK_punct_if_right:NTF #1 { margin } { bound } } \c__xeCJK_right_tl #1
3009     +
3010     \__xeCJK_use_punct_dim:nNN
3011     { \__xeCJK_punct_if_right:NTF #2 { bound } { margin } } \c__xeCJK_left_tl #2
3012   }
3013 }

\__xeCJK_calc_kerning_margin:NN 3014 \cs_new:Npn \__xeCJK_calc_kerning_margin:NN #1#2
\__xeCJK_calc_kerning_margin_aux:NN 3015 {
3016   \dim_max:nn
3017   { \l__xeCJK_kerning_margin_minimum_dim }
3018   {
3019     \bool_if:NTF \l__xeCJK_min_bound_to_kerning_bool
3020     { \l__xeCJK_minimum_bound_dim }
3021     {
3022       \bool_if:NTF \l__xeCJK_optimize_kerning_bool
3023       { \dim_max:nn { \l__xeCJK_minimum_bound_dim } }
3024       { \use:n }
3025       { \__xeCJK_calc_kerning_margin_aux:NN #1 #2 }
3026     }
3027   }
3028 }

3029 \cs_new:Npn \__xeCJK_calc_kerning_margin_aux:NN #1#2
3030 {
3031   \dim_compare:nNnTF \l__xeCJK_kerning_total_width_dim < \c_max_dim
3032   { \__xeCJK_calc_kerning_margin:nNN \l__xeCJK_kerning_total_width_dim }
3033   {
3034     \fp_if_nan:nTF { \l__xeCJK_kerning_total_ratio_fp }
3035     {
3036       \xeCJK_if_same_class:NNTF #1 #2
3037       { \__xeCJK_kerning_width_or_ratio:nNN { same } }
3038       { \__xeCJK_kerning_width_or_ratio:nNN { different } }
3039     }
3040     {
3041       \__xeCJK_calc_kerning_margin:nNN
3042       {
3043         \__xeCJK_dim_ratio:Nn \l__xeCJK_kerning_total_ratio_fp
3044         {
3045           \__xeCJK_use_punct_dim:nN { width } #1 +
3046           \__xeCJK_use_punct_dim:nN { width } #2
3047         }
3048       }
3049     }
3050   }
3051   #1 #2
3052 }

\__xeCJK_kerning_width_or_ratio:nNN 3053 \cs_new:Npn \__xeCJK_kerning_width_or_ratio:nNN #1#2#3
3054 {
3055   \dim_compare:nNnTF { \use:c { l__xeCJK_#1_align_margin_dim } } < \c_max_dim
3056   { \use:c { l__xeCJK_#1_align_margin_dim } }
3057   {
3058     \fp_if_nan:nTF { \use:c { l__xeCJK_#1_align_ratio_fp } }
3059     {
3060       \dim_compare:nNnTF \l__xeCJK_kerning_margin_width_dim < \c_max_dim
3061       { \l__xeCJK_kerning_margin_width_dim \use_none:n }
3062       { \__xeCJK_dim_ratio:Nn \l__xeCJK_kerning_margin_ratio_fp }
3063     }
3064     { \__xeCJK_dim_ratio:cn { l__xeCJK_#1_align_ratio_fp } }
3065     { \l__xeCJK_margin_dim }
3066   }
3067 }

```

```

\__xeCJK_punct_min_bound:NN 3068 \cs_new:Npn \__xeCJK_punct_min_bound:NN #1#2
3069 {
3070   \dim_max:nn
3071   {
3072     \dim_min:nn
3073     { \__xeCJK_use_punct_dim:nNN { bound } \c__xeCJK_left_tl #1 }
3074     { \__xeCJK_use_punct_dim:nNN { bound } \c__xeCJK_right_tl #1 }
3075   }
3076   {
3077     \dim_min:nn
3078     { \__xeCJK_use_punct_dim:nNN { bound } \c__xeCJK_left_tl #2 }
3079     { \__xeCJK_use_punct_dim:nNN { bound } \c__xeCJK_right_tl #2 }
3080   }
3081 }

```

__xeCJK_calc_kerning_margin:nNN #2 和 #3 为相邻的两个标点, #1 为要确定的相邻两个标点总共占的宽度。

```

3082 \cs_new:Npn \__xeCJK_calc_kerning_margin:nNN #1#2#3
3083 {
3084   \dim_eval:n
3085   {
3086     (#1)
3087     - ( \__xeCJK_use_punct_dim:nNN
3088         { \__xeCJK_punct_if_right:NTF #2 { bound } { margin } }
3089         \c__xeCJK_left_tl #2 )
3090     - ( \__xeCJK_use_punct_dim:nNN
3091         { \__xeCJK_punct_if_right:NTF #3 { margin } { bound } }
3092         \c__xeCJK_right_tl #3 )
3093     - ( \__xeCJK_use_punct_dim:nN { dimen } #2 )
3094     - ( \__xeCJK_use_punct_dim:nN { dimen } #3 )
3095   }
3096 }

```

\xeCJK_calc_punct_dimen:N 计算标点的左右实际边距和实际尺寸。

```

3097 \cs_new_protected:Npn \xeCJK_calc_punct_dimen:N #1
3098 {
3099   \__xeCJK_save_punct_dim:nNNn { bound } \c__xeCJK_left_tl #1
3100   { \xeCJK_glyph_bounds:NN 1 #1 }
3101   \__xeCJK_save_punct_dim:nNNn { bound } \c__xeCJK_right_tl #1
3102   { \xeCJK_glyph_bounds:NN 3 #1 }
3103   \__xeCJK_save_punct_dim:nNn { width } #1
3104   { \tex_fontcharwd:D \tex_font:D `#1 }
3105   \__xeCJK_save_punct_dim:nNn { dimen } #1
3106   {
3107     ( \__xeCJK_use_punct_dim:nN { width } #1 ) -
3108     ( \__xeCJK_use_punct_dim:nNN { bound } \c__xeCJK_left_tl #1 ) -
3109     ( \__xeCJK_use_punct_dim:nNN { bound } \c__xeCJK_right_tl #1 )
3110   }
3111 }

```

\xeCJK_glyph_bounds:NN 用 \XeTeXglyphbounds 取得标点符号的上下左右空白。

```

3112 \cs_new:Npn \xeCJK_glyph_bounds:NN #1#2
3113 { \tex_XeTeXglyphbounds:D #1 ~ \tex_XeTeXcharglyph:D `#2 \exp_stop_f: }

```

```

PunctStyle 3114 \keys_define:nn { xeCJK / options }
3115 { PunctStyle .code:n = \exp_args:Ne \__xeCJK_set_punct_style:n {#1} }
3116 \cs_new_protected:Npn \__xeCJK_set_punct_style:n #1
3117 {
3118   \IfInstanceExistTF { xeCJK / punctuation } {#1}
3119   { \tl_set:Nn \l_xeCJK_punct_style_tl {#1} }
3120   {
3121     \prop_get:NnNF \c__xeCJK_punct_style_alias_prop
3122     {#1} \l_xeCJK_punct_style_tl
3123     { \__xeCJK_error:ne { punct-style-unknown } {#1} }
3124   }
3125 }

```

```

3126 \prop_const_from_keyval:Nn \c__xeCJK_punct_style_alias_prop
3127 {
3128   halfwidth      = banjiao ,
3129   fullwidth      = quanjiao ,
3130   mixedwidth     = kaiming ,
3131   marginkerning  = hangmobanjiang ,
3132   plain          = plain
3133 }
3134 \tl_new:N \l_xeCJK_punct_style_tl
3135 \tl_const:Nn \c__xeCJK_punct_style_plain_tl { plain }
3136 \__xeCJK_msg_new:nn { punct-style-unknown }
3137 {
3138   Punctuation~style~"#1"~is~unknown. \\\
3139   The~available~styles~are~listed~as~follow.\\
3140   "plain,~\seq_use:Nnnn \g__xeCJK_punct_style_seq
3141   { ~and~ } { ,~ } { ,~and~ }".\\
3142 }

```

`__xeCJK_trim_spaces:n` `xparse` 处理函数, 先完全展开参数再删除两边空格。

```

3143 \cs_new_protected:Npn \__xeCJK_trim_spaces:n #1
3144 {
3145   \tl_set:Ne \ProcessedArgument
3146   { \exp_args:Ne \tl_trim_spaces:n {#1} }
3147 }

```

`\xeCJKDeclarePunctStyle` 定义新的标点处理风格, 已经存在的同名风格将被覆盖。

```

3148 \NewDocumentCommand \xeCJKDeclarePunctStyle
3149 { > { \__xeCJK_trim_spaces:n } m m }
3150 {
3151   \IfInstanceExistTF { xeCJK / punctuation } {#1}
3152   { \__xeCJK_warning:ne { punct-style-already-defined } {#1} }
3153   { \seq_gput_right:Nn \g__xeCJK_punct_style_seq {#1} }
3154   \DeclareInstance { xeCJK / punctuation } {#1} { basic } {#2}
3155 }
3156 \seq_new:N \g__xeCJK_punct_style_seq
3157 \__xeCJK_msg_new:nn { punct-style-already-defined }
3158 {
3159   Punctuation~style~"#1"~is~already~defined!. \\\
3160   The~existing~style~of~"#1"~will~be~overwritten.\\
3161 }
3162 \@onlypreamble \xeCJKDeclarePunctStyle

```

`\xeCJKEditPunctStyle` 对已有的标点处理风格进行修改。

```

3163 \NewDocumentCommand \xeCJKEditPunctStyle
3164 { > { \__xeCJK_trim_spaces:n } m m }
3165 {
3166   \IfInstanceExistTF { xeCJK / punctuation } {#1}
3167   { \EditInstance { xeCJK / punctuation } {#1} {#2} }
3168   { \__xeCJK_error:ne { punct-style-unknown } {#1} }
3169 }
3170 \@onlypreamble \xeCJKEditPunctStyle

```

默认设置即为全角格式。

```

3171 \xeCJKDeclarePunctStyle { quanjiao } { }
3172 \xeCJKDeclarePunctStyle { hangmobanjiang } { enabled-kerning = false }
3173 \xeCJKDeclarePunctStyle { banjiao }
3174 {
3175   fixed-punct-ratio = 0.5 ,
3176   optimize-margin   = true ,
3177   kerning-total-ratio = 0.5 ,
3178   optimize-kerning  = true
3179 }

```

```

3180 \xeCJKDeclarePunctStyle { kaiming }
3181 {
3182   fixed-punct-ratio = 0.5 ,
3183   mixed-punct-ratio = 0.8 ,
3184   optimize-margin   = true ,
3185   kerning-total-ratio = 0.5 ,
3186   optimize-kerning  = true
3187 }

3188 \xeCJKDeclarePunctStyle { CCT }
3189 {
3190   fixed-punct-ratio = 0.7 ,
3191   optimize-margin   = true ,
3192   kerning-total-ratio = 0.6 ,
3193   optimize-kerning  = true
3194 }

```

5.11 后备字体

AutoFallback 后备字体的宏包选项声明。

```

3195 \keys_define:nn { xeCJK / options }
3196 {
3197   AutoFallback .choice: ,
3198   AutoFallback / true .code:n =
3199   {
3200     \cs_set_eq:NN \xeCJK_fallback_symbol:NN
3201     \__xeCJK_fallback_symbol:NN
3202     \cs_set_eq:NN \xeCJK_fallback_punct_symbol:NN
3203     \__xeCJK_fallback_punct_symbol:NN
3204     \cs_set_eq:NN \xeCJK_clear_fallback_font:
3205     \__xeCJK_clear_fallback_font:
3206   } ,
3207   AutoFallback / false .code:n =
3208   {
3209     \xeCJK_cs_clear:N \xeCJK_fallback_symbol:NN
3210     \xeCJK_cs_clear:N \xeCJK_fallback_punct_symbol:NN
3211     \xeCJK_cs_clear:N \xeCJK_clear_fallback_font:
3212   } ,
3213   AutoFallback .default:n = { true } ,
3214   fallback .meta:n = { AutoFallback = true }
3215 }

```

`\xeCJK_fallback_symbol:NN` 测试当前字体中是否存在当前字符,如存在则直接输出,否则启用后备字体。

```

\__xeCJK_fallback_punct_symbol:NN
3216 \cs_new_protected:Npn \__xeCJK_fallback_symbol:NN #1#2
3217 {
3218   \xeCJK_reset_fallback_font:
3219   \xeCJK_glyph_if_exist:NF #2
3220   { \__xeCJK_fallback_symbol_aux:NN }
3221   #1#2
3222 }
3223 \cs_new_protected:Npn \__xeCJK_fallback_punct_symbol:NN #1#2
3224 {
3225   \xeCJK_reset_fallback_font:
3226   \xeCJK_glyph_if_exist:NF #2
3227   { \__xeCJK_fallback_punct_symbol_aux:NN }
3228   #1#2
3229 }
3230 \cs_new_eq:NN \xeCJK_fallback_symbol:NN \prg_do_nothing:
3231 \cs_new_eq:NN \xeCJK_fallback_punct_symbol:NN \prg_do_nothing:
3232 \cs_new_protected:Npn \__xeCJK_fallback_symbol_aux:NN
3233 {
3234   \__xeCJK_fallback_symbol_aux:nnNN
3235   { \CJK@family }
3236   { \l_xeCJK_family_tl }
3237 }

```

```

3238 \cs_new_protected:Npn \__xeCJK_fallback_punct_symbol_aux:NN
3239   {
3240     \__xeCJK_fallback_symbol_aux:nnNN
3241     { \CJK@punctfamily }
3242     { \l_xeCJK_punct_family_tl }
3243   }
3244 \cs_new_protected:Npn \__xeCJK_fallback_symbol_aux:nnNN
3245   {
3246     \cs_set_protected:Npe \xeCJK_reset_fallback_font:
3247     {
3248       \tex_the:D \tex_font:D
3249       \xeCJK_clear_fallback_font:
3250     }
3251     \exp_args:Nee \__xeCJK_fallback_loop:nnNN
3252   }
3253 \cs_new_protected:Npn \__xeCJK_clear_fallback_font:
3254   { \cs_set_eq:NN \xeCJK_reset_fallback_font: \prg_do_nothing: }
3255 \cs_new_eq:NN \xeCJK_reset_fallback_font: \prg_do_nothing:
3256 \cs_new_eq:NN \xeCJK_clear_fallback_font: \prg_do_nothing:

```

__xeCJK_fallback_loop:nnNN 循环测试后备字体是否包含字符 #1。若后备字体中存在该字符或者再没有后备字体，则结束循环。当前字体族没有备用字体时，使用 \CJKfamilydefault 的设置。

```

3257 \cs_new_protected:Npn \__xeCJK_fallback_loop:nnNN
3258   {
3259     \cs_set_eq:NN \__xeCJK_fallback_loop:TF \use_i:nn
3260     \__xeCJK_fallback_loop:nnnNN { FallBack }
3261   }
3262 \cs_new_protected:Npn \__xeCJK_fallback_loop:nnnNN #1#2#3
3263   {
3264     \xeCJK_select_fallback_font:nnn {#1} {#2} {#3}
3265     \__xeCJK_fallback_loop:TF
3266     { \__xeCJK_fallback_loop_aux:nnnNN }
3267     { \__xeCJK_fallback_missing_glyph:nnnNN }
3268     {#1} {#2} {#3}
3269   }
3270 \cs_new_protected:Npn \__xeCJK_fallback_loop_aux:nnnNN #1#2#3#4#5
3271   {
3272     \xeCJK_glyph_if_exist:NF #5
3273     { \__xeCJK_fallback_loop:nnnNN { #1/FallBack } {#2} {#3} }
3274     #4#5
3275   }
3276 \cs_new_protected:Npn \__xeCJK_fallback_missing_glyph:nnnNN #1#2#3#4#5
3277   {
3278     \__xeCJK_warning:neee { missing-glyph } {#1} {#2} {#5}
3279     #4#5
3280   }
3281 \cs_new_protected:Npn \xeCJK_select_fallback_font:nnn #1#2
3282   {
3283     \__xeCJK_select_fallback_font:cnnn
3284     { \__xeCJK_font_csname:n { #2/#1 } } {#1} {#2}
3285   }
3286 \cs_new_protected:Npn \__xeCJK_select_fallback_font:Nnnn #1
3287   {
3288     \cs_if_exist:NF #1
3289     { \__xeCJK_fallback_font_initial:NNnnn }
3290     #1 \use_none:nnn
3291   }
3292 \cs_generate_variant:Nn \__xeCJK_select_fallback_font:Nnnn { c }
3293 \cs_new_protected:Npn \__xeCJK_fallback_font_initial:NNnnn #1#2#3#4#5
3294   {
3295     \xeCJK_family_if_exist:nTF { #5/#3 }
3296     { \__xeCJK_font_initial:Nn #1 { #5/#3 } }
3297     { \__xeCJK_fallback_font_initial_auxi:Nnnn #1 {#5} {#3} {#4} }
3298     #1
3299   }
3300 \cs_new_protected:Npn \__xeCJK_fallback_font_initial_auxi:Nnnn #1

```

```

3301 {
3302   \exp_args:NNe \__xeCJK_fallback_font_initial_auxii:Nnnnn
3303   #1 { \CJKfamilydefault }
3304 }
3305 \cs_new_protected:Npn \__xeCJK_fallback_font_initial_auxii:Nnnnn #1#2#3
3306 {
3307   \str_if_eq:nnTF {#2} {#3}
3308   { \__xeCJK_fallback_loop_end:Nnnn }
3309   { \__xeCJK_fallback_font_initial_auxiii:Nnnn }
3310   #1 {#2}
3311 }
3312 \cs_new_protected:Npn \__xeCJK_fallback_font_initial_auxiii:Nnnn #1#2
3313 {
3314   \xeCJK_family_if_exist:nTF {#2}
3315   { \__xeCJK_fallback_font_initial_auxiv:Nnnn }
3316   { \__xeCJK_fallback_loop_end:Nnnn }
3317   #1 {#2}
3318 }
3319 \cs_new_protected:Npn \__xeCJK_fallback_font_initial_auxiv:Nnnn #1#2#3#4
3320 {
3321   \__xeCJK_font_initial:Nn #1 {#2}
3322   \exp_args:Nc \__xeCJK_fallback_font_initial_auxiii:Nnnn
3323   { \__xeCJK_font_csname:n { #4/#3/FallBack } }
3324   { #2/FallBack } { #3/FallBack } {#4}
3325 }
3326 \cs_new_eq:NN \__xeCJK_fallback_loop:TF \use_i:nn
3327 \cs_new_protected:Npn \__xeCJK_fallback_loop_end:Nnnn #1#2#3#4
3328 { \cs_gset_eq:NN #1 \__xeCJK_fallback_loop_end: }
3329 \cs_new_protected:Npn \__xeCJK_fallback_loop_end:
3330 { \cs_set_eq:NN \__xeCJK_fallback_loop:TF \use_ii:nn }
3331 \__xeCJK_msg_new:nn { missing-glyph }
3332 {
3333   CJKfamily~\__xeCJK_msg_family_map:n {#2}'~(#1)~
3334   does~not~contain~glyph~`#3'~
3335   ( U + \int_to_Hex:n { `#3 } )~\msg_line_context:.
3336 }
\setCJKfallbackfamilyfont 3337 \NewDocumentCommand \setCJKfallbackfamilyfont { m o m }
3338 {
3339   \__xeCJK_pass_args:nnnn
3340   { \xeCJK_set_family_fallback:nnn {#1} } {#2} {#3}
3341   { }
3342 }
\xeCJK_set_family_fallback:nnn 3343 \cs_new_protected:Npn \xeCJK_set_family_fallback:nnn #1#2#3
3344 {
3345   \group_begin:
3346   \tl_set:Nn \l__xeCJK_fallback_family_tl {#1}
3347   \prop_get:NoNF \g__xeCJK_family_font_name_prop
3348   \l__xeCJK_fallback_family_tl \l__xeCJK_font_name_tl
3349   { \tl_clear:N \l__xeCJK_font_name_tl }
3350   \clist_set:Nn \l__xeCJK_public_options_clist {#2}
3351   \clist_map_function:nN {#3} \__xeCJK_set_family_fallback:n
3352   \group_end:
3353 }
3354 \cs_new_protected:Npn \__xeCJK_set_family_fallback:n #1
3355 {
3356   \tl_put_right:Nn \l__xeCJK_fallback_family_tl { /FallBack }
3357   \__xeCJK_get_sub_features:on \l__xeCJK_fallback_family_tl {#1}
3358   \clist_concat:NNN \l__xeCJK_sub_font_options_clist
3359   \l__xeCJK_public_options_clist
3360   \l__xeCJK_sub_font_options_clist
3361   \exp_args:Nooo \xeCJK_set_family:nnn
3362   \l__xeCJK_fallback_family_tl
3363   \l__xeCJK_sub_font_options_clist
3364   \l__xeCJK_sub_font_name_tl
3365 }
3366 \tl_new:N \l__xeCJK_fallback_family_tl

```

```
3367 \clist_new:N \l__xeCJK_public_options_clist
```

5.12 CJK 字体族声明方式

```
3368 \bool_new:N \g__xeCJK_auto_fake_bold_bool
3369 \bool_new:N \g__xeCJK_auto_fake_slant_bool
3370 \fp_new:N \g__xeCJK_embolden_factor_fp
3371 \fp_new:N \g__xeCJK_slant_factor_fp
```

AutoFakeBold
AutoFakeSlant
EmboldenFactor
SlantFactor

伪粗体和伪斜体的宏包选项声明。

```
3372 \keys_define:nn { xeCJK / options }
3373 {
3374   AutoFakeBold .choices:nn = { true , false }
3375   { \use:c { bool_gset_ #1 :N } \g__xeCJK_auto_fake_bold_bool } ,
3376   AutoFakeBold / unknown .code:n =
3377   {
3378     \bool_gset_true:N \g__xeCJK_auto_fake_bold_bool
3379     \fp_gset:Nn \g__xeCJK_embolden_factor_fp { \l_keys_value_tl }
3380   } ,
3381   AutoFakeBold .default:n = { true } ,
3382   AutoFakeSlant .choices:nn = { true , false }
3383   { \use:c { bool_gset_ #1 :N } \g__xeCJK_auto_fake_slant_bool } ,
3384   AutoFakeSlant / unknown .code:n =
3385   {
3386     \bool_gset_true:N \g__xeCJK_auto_fake_slant_bool
3387     \fp_gset:Nn \g__xeCJK_slant_factor_fp { \l_keys_value_tl }
3388   } ,
3389   AutoFakeSlant .default:n = { true } ,
3390   EmboldenFactor .fp_gset:N = \g__xeCJK_embolden_factor_fp ,
3391   SlantFactor .fp_gset:N = \g__xeCJK_slant_factor_fp ,
3392   BoldFont .meta:n = { AutoFakeBold = true } ,
3393   boldfont .meta:n = { AutoFakeBold = true } ,
3394   SlantFont .meta:n = { AutoFakeSlant = true } ,
3395   slantfont .meta:n = { AutoFakeSlant = true }
3396 }
```

`\xeCJK_new_sub_key:n`
`\g__xeCJK_sub_key_seq`

用于定义 CJK 子区字体和备用字体的选项。

```
3397 \seq_new:N \g__xeCJK_sub_key_seq
3398 \cs_new_protected:Npn \xeCJK_new_sub_key:n #1
3399 {
3400   \seq_gput_right:Nn \g__xeCJK_sub_key_seq {#1}
3401   \keys_define:nn { xeCJK / features }
3402   {
3403     #1 .code:n =
3404     {
3405       \tl_if_blank:nTF {##1}
3406       {
3407         \prop_clear:N \l__xeCJK_sub_key_prop
3408         \tl_set:Ne \l__xeCJK_sub_family_name_tl
3409         { \l__xeCJK_family_name_tl /#1 }
3410         \clist_remove_all:Nn \l__xeCJK_font_options_clist {#1}
3411       }
3412       {
3413         \tl_clear:N \l__xeCJK_sub_family_name_tl
3414         \str_if_eq:nnTF {##1} { * }
3415         { \prop_put:Nnn \l__xeCJK_sub_key_prop {#1} { \q_no_value } }
3416         { \__xeCJK_get_sub_features:nn {#1} {##1} }
3417       }
3418     } ,
3419     #1 .default:n = { }
3420   }
3421 }
```

```

\__xeCJK_get_sub_features:nn 3422 \cs_new_protected:Npn \__xeCJK_get_sub_features:nn #1#2
\__xeCJK_get_sub_features:w 3423 {
3424   \tl_set:Nc \l__xeCJK_tmp_tl { \xeCJK_tl_remove_outer_braces:n {#2} }
3425   \clist_clear:N \l__xeCJK_sub_font_options_clist
3426   \exp_after:wN \__xeCJK_get_sub_features:w \l__xeCJK_tmp_tl
3427   \q_mark [ \q_nil ] \q_mark \q_stop
3428   \tl_if_empty:NTF \l__xeCJK_sub_font_name_tl
3429     { \tl_set_eq:NN \l__xeCJK_sub_font_name_tl \l__xeCJK_font_name_tl }
3430     { \tl_replace_all:Nno \l__xeCJK_sub_font_name_tl { * } \l__xeCJK_font_name_tl }
3431   \prop_put:Nne \l__xeCJK_sub_key_prop {#1}
3432     {
3433       { \exp_not:o \l__xeCJK_sub_font_options_clist }
3434       { \exp_not:o \l__xeCJK_sub_font_name_tl }
3435     }
3436   }
3437 \cs_new_protected:Npn \__xeCJK_get_sub_features:w #1 [#2] #3 \q_mark #4 \q_stop
3438 {
3439   \quark_if_nil:nTF {#2}
3440     { \tl_set_eq:NN \l__xeCJK_sub_font_name_tl \l__xeCJK_tmp_tl }
3441     {
3442       \tl_set:Nc \l__xeCJK_sub_font_name_tl
3443         { \xeCJK_tl_remove_outer_braces:n {#3} }
3444       \tl_if_empty:NTF \l__xeCJK_sub_font_name_tl
3445         { \tl_set_eq:NN \l__xeCJK_sub_font_name_tl \l__xeCJK_tmp_tl }
3446         { \clist_set:Nn \l__xeCJK_sub_font_options_clist {#2} }
3447     }
3448 }
3449 \tl_new:N \l__xeCJK_sub_family_name_tl
3450 \tl_new:N \l__xeCJK_sub_font_name_tl
3451 \clist_new:N \l__xeCJK_sub_font_options_clist
3452 \cs_generate_variant:Nn \__xeCJK_get_sub_features:nn { o }
3453 \cs_generate_variant:Nn \tl_replace_all:Nnn { Nno }

```

FallBack 3454 \xeCJK_new_sub_key:n { FallBack }

BoldFont 调用字体的属性声明, 同 `fontspec` 宏包。
ItalicFont

```

3455 \keys_define:nn { xeCJK / features }
3456 {
3457   BoldFont .tl_set:N = \l__xeCJK_font_name_bf_tl ,
3458   ItalicFont .tl_set:N = \l__xeCJK_font_name_it_tl
3459 }

AutoFakeBold 3460 \keys_define:nn { xeCJK / features }
AutoFakeSlant 3461 {
3462   AutoFakeBold .choice: ,
3463   AutoFakeBold / true .code:n =
3464     {
3465       \bool_set_true:N \l__xeCJK_auto_fake_bold_bool
3466       \fp_set_eq:NN \l__xeCJK_embolden_factor_fp \g__xeCJK_embolden_factor_fp
3467     } ,
3468   AutoFakeBold / false .code:n =
3469     { \bool_set_false:N \l__xeCJK_auto_fake_bold_bool } ,
3470   AutoFakeBold / unknown .code:n =
3471     {
3472       \bool_set_true:N \l__xeCJK_auto_fake_bold_bool
3473       \fp_set:Nn \l__xeCJK_embolden_factor_fp { \l_keys_value_tl }
3474     } ,
3475   AutoFakeBold .default:n = { true } ,
3476   AutoFakeSlant .choice: ,
3477   AutoFakeSlant / true .code:n =
3478     {
3479       \bool_set_true:N \l__xeCJK_auto_fake_slant_bool
3480       \fp_set_eq:NN \l__xeCJK_slant_factor_fp \g__xeCJK_slant_factor_fp
3481     } ,
3482   AutoFakeSlant / false .code:n =
3483     { \bool_set_false:N \l__xeCJK_auto_fake_slant_bool } ,

```

```

3484     AutoFakeSlant / unknown .code:n =
3485     {
3486         \bool_set_true:N \l__xeCJK_auto_fake_slant_bool
3487         \fp_set:Nn \l__xeCJK_slant_factor_fp { \l_keys_value_tl }
3488     } ,
3489     AutoFakeSlant .default:n = { true }
3490 }

\__xeCJK_set_family_initial: 3491 \cs_new_protected:Npn \__xeCJK_set_family_initial:
3492 {
3493     \int_gincr:N \g__xeCJK_family_int
3494     \prop_clear:N \l__xeCJK_sub_key_prop
3495     \tl_clear:N \l__xeCJK_font_name_bf_tl
3496     \tl_clear:N \l__xeCJK_font_name_it_tl
3497     \tl_clear:N \l__xeCJK_sub_family_name_tl
3498     \clist_clear:N \l__xeCJK_fontspec_options_clist
3499     \bool_set_eq:NN \l__xeCJK_auto_fake_bold_bool \g__xeCJK_auto_fake_bold_bool
3500     \bool_set_eq:NN \l__xeCJK_auto_fake_slant_bool \g__xeCJK_auto_fake_slant_bool
3501     \fp_set_eq:NN \l__xeCJK_embolden_factor_fp \g__xeCJK_embolden_factor_fp
3502     \fp_set_eq:NN \l__xeCJK_slant_factor_fp \g__xeCJK_slant_factor_fp
3503 }
3504 \int_new:N \g__xeCJK_family_int
3505 \prop_new:N \l__xeCJK_sub_key_prop
3506 \clist_new:N \l__xeCJK_fontspec_options_clist
3507 \bool_new:N \l__xeCJK_auto_fake_bold_bool
3508 \bool_new:N \l__xeCJK_auto_fake_slant_bool
3509 \fp_new:N \l__xeCJK_embolden_factor_fp
3510 \fp_new:N \l__xeCJK_slant_factor_fp

```

\xeCJK_set_family:nnn 设置一个CJK新字体族,与 \newfontfamily 类似,增加 FallBack 选项。

```

3511 \cs_new_protected:Npn \xeCJK_set_family:nnn #1#2#3
3512 {
3513     \group_begin:
3514     \__xeCJK_set_family_initial:
3515     \tl_set:Nn \l__xeCJK_family_name_tl {#1}
3516     \clist_set:Nn \l__xeCJK_font_options_clist {#2}
3517     \tl_set:Nn \l__xeCJK_font_name_tl {#3}
3518     \clist_concat:NNN \l__xeCJK_font_options_clist
3519     \g__xeCJK_default_features_clist \l__xeCJK_font_options_clist
3520     \keys_set_known:noN { xeCJK / features }
3521     \l__xeCJK_font_options_clist \l__xeCJK_fontspec_options_clist
3522     \__xeCJK_binding_sub_family:
3523     \__xeCJK_parse_font_shape:
3524     \__xeCJK_check_family:o \l__xeCJK_family_name_tl
3525     \__xeCJK_gset_family_cs:n { \l__xeCJK_family_name_tl }
3526     \__xeCJK_save_family_info:
3527     \__xeCJK_set_sub_block_family:
3528     \group_end:
3529 }
3530 \tl_new:N \l__xeCJK_family_name_tl
3531 \tl_new:N \l__xeCJK_font_name_tl
3532 \clist_new:N \l__xeCJK_font_options_clist
3533 \cs_generate_variant:Nn \xeCJK_set_family:nnn { e , o }

```

```

\__xeCJK_binding_sub_family: 3534 \cs_new_protected:Npn \__xeCJK_binding_sub_family:
3535 {
3536     \tl_if_empty:NF \l__xeCJK_sub_family_name_tl
3537     { \tl_set_eq:NN \l__xeCJK_family_name_tl \l__xeCJK_sub_family_name_tl }
3538 }

```

```

\__xeCJK_gset_family_cs:n 3539 \cs_new_protected:Npn \__xeCJK_gset_family_cs:n #1
3540 {
3541     \cs_gset_protected:cpe { \__xeCJK_family_csname:n {#1} }
3542     {
3543         \group_begin:
3544         \exp_not:n { \cs_set_eq:NN \xeCJK@fontfamily \use_none:n }
3545         \exp_not:n { \fontspec_gset_family:Nnn \g__xeCJK_fontspec_family_tl }

```

```

3546         { \exp_not:o \l__xeCJK_fontspec_options_clist }
3547         { \exp_not:o \l__xeCJK_font_name_tl }
3548     \__xeCJK_gset_family_nfss_cs:no
3549     {#1} { \exp_not:N \g__xeCJK_fontspec_family_tl }
3550     \group_end:
3551     \tl_set_eq:NN \exp_not:N \l__xeCJK_fontspec_family_tl
3552         \exp_not:N \g__xeCJK_fontspec_family_tl
3553     }
3554 }
3555 \tl_new:N \g__xeCJK_fontspec_family_tl
3556 \tl_new:N \l__xeCJK_fontspec_family_tl

\__xeCJK_check_family:n 3557 \cs_new_protected:Npn \__xeCJK_check_family:n #1
3558 {
3559     \prop_gpop:NnNT \g__xeCJK_family_font_name_prop {#1} \l__xeCJK_tmp_tl
3560     {
3561         \prop_gpop:NnNT \g__xeCJK_family_name_prop {#1} \l__xeCJK_tmp_tl
3562         {
3563             \cs_undefine:c { \__xeCJK_family_csname:n {#1} }
3564             \cs_undefine:c { \__xeCJK_family_nfss_csname:n {#1} }
3565         }
3566         \__xeCJK_warning:nee { CJKfamily-redef } {#1} { \l__xeCJK_tmp_tl }
3567     }
3568 }
3569 \cs_generate_variant:Nn \__xeCJK_check_family:n { o }
3570 \__xeCJK_msg_new:nn { CJKfamily-redef }
3571 { Redefining~CJKfamily~\__xeCJK_msg_family_map:n {#1}'~(#2). }

\__xeCJK_parse_font_shape: 3572 \cs_new_protected:Npn \__xeCJK_parse_font_shape:
3573 {
3574     \tl_if_blank:oTF \l__xeCJK_font_name_bf_tl
3575     {
3576         \bool_if:NT \l__xeCJK_auto_fake_bold_bool
3577         {
3578             \clist_put_right:Ne \l__xeCJK_fontspec_options_clist
3579                 { AutoFakeBold = { \fp_use:N \l__xeCJK_embolden_factor_fp } }
3580         }
3581     }
3582     {
3583         \clist_put_right:Ne \l__xeCJK_fontspec_options_clist
3584             { BoldFont = { \exp_not:o \l__xeCJK_font_name_bf_tl } }
3585     }
3586     \tl_if_blank:oTF \l__xeCJK_font_name_it_tl
3587     {
3588         \bool_if:NT \l__xeCJK_auto_fake_slant_bool
3589         {
3590             \clist_put_right:Ne \l__xeCJK_fontspec_options_clist
3591                 { AutoFakeSlant = { \fp_use:N \l__xeCJK_slant_factor_fp } }
3592         }
3593     }
3594     {
3595         \clist_put_right:Ne \l__xeCJK_fontspec_options_clist
3596             { ItalicFont = { \exp_not:o \l__xeCJK_font_name_it_tl } }
3597     }
3598 }

\g__xeCJK_family_name_prop 3599 \prop_new:N \g__xeCJK_family_name_prop
\g__xeCJK_family_font_name_prop 3600 \prop_new:N \g__xeCJK_family_font_name_prop
\g__xeCJK_family_font_options_prop 3601 \prop_new:N \g__xeCJK_family_font_options_prop

\__xeCJK_save_family_info: 3602 \cs_new_protected:Npn \__xeCJK_save_family_info:
3603 {
3604     \exp_args:Nooo \__xeCJK_save_family_info:nnn
3605         \l__xeCJK_family_name_tl
3606         \l__xeCJK_font_name_tl
3607         \l__xeCJK_font_options_clist
3608 }
3609 \cs_new_protected:Npn \__xeCJK_save_family_info:nnn #1#2#3

```

```

3610 {
3611   \prop_gput:Nnn \g__xeCJK_family_font_name_prop {#1} {#2}
3612   \prop_gput:Nnn \g__xeCJK_family_font_options_prop {#1} {#3}
3613 }

```

```

\__xeCJK_set_sub_block_family: 3614 \cs_new_protected:Npn \__xeCJK_set_sub_block_family:
3615 {
3616   \prop_if_empty:NF \l__xeCJK_sub_key_prop
3617   {
3618     \prop_map_function:NN
3619     \l__xeCJK_sub_key_prop
3620     \__xeCJK_set_sub_block_family:nn
3621   }
3622 }
3623 \cs_new_protected:Npn \__xeCJK_set_sub_block_family:nn #1#2
3624 {
3625   \tl_set:Ne \l__xeCJK_sub_family_name_tl { \l__xeCJK_family_name_tl/#1 }
3626   \quark_if_no_value:nTF {#2}
3627   { \__xeCJK_copy_sub_family:n {#1} }
3628   { \xeCJK_set_family:onn \l__xeCJK_sub_family_name_tl #2 }
3629 }
3630 \cs_new_protected:Npn \__xeCJK_copy_sub_family:n #1
3631 {
3632   \__xeCJK_check_family:o \l__xeCJK_sub_family_name_tl
3633   \prop_get:NoNT \g__xeCJK_family_font_name_prop
3634   \l__xeCJK_family_name_tl \l__xeCJK_sub_font_name_tl
3635   {
3636     \prop_gput:Noo \g__xeCJK_family_font_name_prop
3637     \l__xeCJK_sub_family_name_tl \l__xeCJK_sub_font_name_tl
3638   }
3639   \prop_get:NoNT \g__xeCJK_family_font_options_prop
3640   \l__xeCJK_family_name_tl \l__xeCJK_sub_font_options_clist
3641   {
3642     \clist_remove_all:Nn \l__xeCJK_sub_font_options_clist { #1 = * }
3643     \prop_gput:Noo \g__xeCJK_family_font_options_prop
3644     \l__xeCJK_sub_family_name_tl \l__xeCJK_sub_font_options_clist
3645   }
3646   \cs_gset_protected:cpe
3647   { \__xeCJK_family_csname:n { \l__xeCJK_sub_family_name_tl } }
3648   {
3649     \xeCJK_family_if_exist:eT { \l__xeCJK_family_name_tl }
3650     {
3651       \__xeCJK_gset_family_nfss_cs:no
3652       { \l__xeCJK_sub_family_name_tl }
3653       { \exp_not:N \l__xeCJK_fontspec_family_tl }
3654     }
3655   }
3656 }

```

```

\__xeCJK_copy_family:nn 3657 \cs_new_protected:Npn \__xeCJK_copy_family:nn #1#2
\__xeCJK_copy_family:ee 3658 {
3659   \xeCJK_family_if_exist:nT {#2}
3660   {
3661     \prop_gput:Nno \g__xeCJK_family_name_prop
3662     {#1} \l__xeCJK_fontspec_family_tl
3663     \tl_map_inline:nn
3664     {
3665       \g__xeCJK_family_font_name_prop
3666       \g__xeCJK_family_font_options_prop
3667     }
3668     {
3669       \prop_get:NnNT ##1 {#2} \l__xeCJK_tmp_tl
3670       { \prop_gput:Nno ##1 {#1} \l__xeCJK_tmp_tl }
3671     }
3672     \cs_gset_eq:cc
3673     { \__xeCJK_family_nfss_csname:n {#1} }
3674     { \__xeCJK_family_nfss_csname:n {#2} }
3675   }

```

```

3676 }
3677 \cs_generate_variant:Nn \__xeCJK_copy_family:nn { ee }

```

5.13 字体切换

`\xeCJK_select_font:` 缓存当前字体的原始格式, 以加速编译。
`\l_xeCJK_current_font_tl`

```

3678 \cs_new:Npn \__xeCJK_font_csname:n #1
3679 { xeCJK/#1/\f@series/\f@shape/\f@size }
3680 \tl_new:N \l_xeCJK_current_font_tl
3681 \tl_set:No \l_xeCJK_current_font_tl
3682 { \__xeCJK_font_csname:n { \CJK@family } }
3683 \cs_new_protected:Npn \xeCJK_select_font:
3684 {
3685   \__xeCJK_select_font:cn
3686   { \l_xeCJK_current_font_tl }
3687   { \l_xeCJK_family_tl }
3688 }
3689 \cs_new_protected:Npn \__xeCJK_select_font:Nn #1#2
3690 {
3691   \xeCJK_clear_fallback_font:
3692   \cs_if_exist:NF #1 { \__xeCJK_font_initial:Nn #1 {#2} }
3693   #1
3694 }
3695 \cs_generate_variant:Nn \__xeCJK_select_font:Nn { c }
3696 \tl_new:N \l__xeCJK_current_coor_tl
3697 \cs_new_eq:NN \xeCJK@setfont \xeCJK_select_font:

```

`__xeCJK_font_initial:Nn` 注意要将 `\selectfont` 放在分组中调用, 防止 `\f@series` 等字体参数被修改, 导致 `\l_xeCJK_current_font_tl` 标记前后不一致, 引发错误(见 #486)。

```

3698 \cs_new_protected:Npn \__xeCJK_font_initial:Nn #1#2
3699 {
3700   \group_begin:
3701   \__xeCJK_font_initial_hook:
3702   \__xeCJK_family_use:n {#2}
3703   \xeCJK_font_gset_to_current:N #1
3704   \group_end:
3705 }
3706 \cs_new_protected:Npn \__xeCJK_font_initial_hook:
3707 { \tl_use:N \g__xeCJK_font_initial_hook_tl }
3708 \cs_new_protected:Npn \__xeCJK_gadd_font_initial_hook:n
3709 { \tl_gput_right:Nn \g__xeCJK_font_initial_hook_tl }
3710 \tl_new:N \g__xeCJK_font_initial_hook_tl

```

`\xeCJK_select_punct_font:` 切换标点符号字体。

```

\l_xeCJK_current_punct_font_tl
3711 \cs_new_eq:NN \xeCJK_select_punct_font: \xeCJK_select_font:
3712 \cs_new_protected:Npn \__xeCJK_select_punct_font_aux:
3713 {
3714   \__xeCJK_select_font:cn
3715   { \l_xeCJK_current_punct_font_tl }
3716   { \l_xeCJK_punct_family_tl }
3717 }
3718 \tl_new:N \CJK@punctfamily
3719 \tl_new:N \l_xeCJK_punct_family_tl
3720 \tl_new:N \l_xeCJK_current_punct_font_tl
3721 \tl_set:Nn \CJK@punctfamily { \CJK@family }
3722 \tl_set:Nn \l_xeCJK_punct_family_tl { \l_xeCJK_family_tl }
3723 \tl_set:No \l_xeCJK_current_punct_font_tl
3724 { \__xeCJK_font_csname:n { \CJK@punctfamily } }
3725 \cs_new_eq:NN \__xeCJK_select_font: \prg_do_nothing:
3726 \cs_new_eq:NN \__xeCJK_select_punct_font: \prg_do_nothing:

```

`_xeCJK_switch_font:nn` 两个 CJK 分区之间的字体切换。

```

3727 \cs_new_protected:Npn \_xeCJK_switch_font:nn #1#2
3728 {
3729   \str_if_eq:nnF {#1} {#2}
3730   {
3731     \_xeCJK_info:nee { CJK-block } {#1} {#2}
3732     \str_if_eq:nnTF {#2} { CJK }
3733     { \xeCJK_select_font: }
3734     { \xeCJK_select_font:n {#2} }
3735   }
3736 }
3737 \_xeCJK_msg_new:nn { CJK-block } { Switch~from~block~`#1'~to~`#2'. }

```

`\xeCJK_select_font:n` 若当前 CJK 字体族没有定义子分区 #1 的字体,则使用 `\CJKfamilydefault` 的对应分区字体;

`\xeCJK_block_family:nn` 若 `\CJKfamilydefault` 也没有定义该分区字体,则使用当前 CJK 字体族的主分区字体。

```

3738 \cs_new_protected:Npn \xeCJK_select_font:n #1
3739 {
3740   \_xeCJK_select_font:cnn
3741   { \_xeCJK_font_csname:n { \CJK@family/#1 } }
3742   { \l_xeCJK_family_tl }
3743   {#1}
3744 }
3745 \cs_new_protected:Npn \_xeCJK_select_font:Nnn #1#2#3
3746 {
3747   \xeCJK_clear_fallback_font:
3748   \cs_if_exist:NF #1
3749   { \_xeCJK_block_font_initial:Nnn #1 {#2} {#3} }
3750   #1
3751 }
3752 \cs_generate_variant:Nn \_xeCJK_select_font:Nnn { c }
3753 \cs_new_protected:Npn \_xeCJK_block_font_initial:Nnn #1#2#3
3754 {
3755   \xeCJK_block_family:nn {#2} {#3}
3756   \_xeCJK_font_initial:Nn #1 { #2/#3 }
3757 }
3758 \cs_new_protected:Npn \xeCJK_block_family:nn #1#2
3759 {
3760   \xeCJK_family_if_exist:eF { #1/#2 }
3761   {
3762     \_xeCJK_copy_family:ee { #1/#2 }
3763     {
3764       \cs_if_exist:cTF
3765       { \_xeCJK_family_csname:n { \CJKfamilydefault/#2 } }
3766       { \CJKfamilydefault/#2 } {#1}
3767     }
3768   }
3769 }

```

```

\_xeCJK_family_csname:n 3770 \cs_new:Npn \_xeCJK_family_csname:n #1
\_xeCJK_family_nfss_csname:n 3771 { xeCJK/family/#1 }
\_xeCJK_family_use:n 3772 \cs_new:Npn \_xeCJK_family_nfss_csname:n #1
\_xeCJK_gset_family_nfss_cs:nn 3773 { xeCJK/family/nfss/#1 }
3774 \cs_new_protected:Npn \_xeCJK_family_use:n #1
3775 { \use:c { \_xeCJK_family_nfss_csname:n {#1} } }
3776 \cs_new_protected:Npn \_xeCJK_gset_family_nfss_cs:nn #1#2
3777 {
3778   \prop_gput:Nnn \g_xeCJK_family_name_prop {#1} {#2}
3779   \cs_gset_protected:cpe
3780   { \_xeCJK_family_nfss_csname:n {#1} }
3781   { \_xeCJK_nfss_family:nn { \c_xeCJK_encoding_tl } {#2} }
3782 }
3783 \cs_generate_variant:Nn \_xeCJK_gset_family_nfss_cs:nn { no }

```

`_xeCJK_nfss_family:n` 用于处理 L^AT_EX₂_ε 2020/02/02 中 `\bfseries@rm` 等与 `\bfdefault` 不一致可能导致的问题。

```

3784 \cs_if_exist:NTF \fontseriesforce

```

```

3785 {
3786   \cs_new_protected:Npn \__xeCJK_nfss_family:nn #1#2
3787   {
3788     \fontencoding {#1}
3789     \str_if_eq:eeF { \f@series } { \bfdefault }
3790     {
3791       \str_case:e:nn { \f@family }
3792       {
3793         { \rmdefault } { \__xeCJK_nfss_series:n { rm } }
3794         { \sfdefault } { \__xeCJK_nfss_series:n { sf } }
3795         { \ttdefault } { \__xeCJK_nfss_series:n { tt } }
3796       }
3797     }
3798     \fontfamily {#2}
3799     \selectfont
3800   }
3801   \cs_new_protected:Npn \__xeCJK_nfss_series:n #1
3802   {
3803     \str_if_eq:eeT { \f@series } { \use:c { bfseries@#1 } }
3804     { \fontseriesforce { \bfdefault } }
3805   }
3806 }
3807 {
3808   \cs_new_protected:Npn \__xeCJK_nfss_family:nn #1#2
3809   {
3810     \fontencoding {#1}
3811     \tl_set:Nn \f@family {#2}
3812     \selectfont
3813   }
3814 }

```

```

\xeCJK_family_if_exist:nTF 3815 \prg_new_protected_conditional:Npnn \xeCJK_family_if_exist:n #1 { T , F , TF }
3816 {
3817   \prop_get:NnNTF \g__xeCJK_family_name_prop
3818   {#1} \l__xeCJK_fontspec_family_tl
3819   { \prg_return_true: }
3820   {
3821     \exp_args:Ne \cs_if_exist_use:cTF
3822     { \__xeCJK_family_csname:n {#1} }
3823     { \prg_return_true: }
3824     { \prg_return_false: }
3825   }
3826 }
3827 \prg_generate_conditional_variant:Nnn \xeCJK_family_if_exist:n { e } { T , F , TF }

```

\CJKfamily 用于切换 CJK 字体族。

```

3828 \NewDocumentCommand \CJKfamily { t+ t- m }
3829 {
3830   \xeCJK_family:NNe #1 #2 {#3}
3831   \tex_ignorespaces:D
3832 }
3833 \cs_new_protected:Npn \xeCJK_family:NNn #1#2#3
3834 {
3835   \tl_if_blank:nTF {#3}
3836   {
3837     \bool_if:NF #1 { \bool_if:NF #2 { \use_none:nn } }
3838     \xeCJK_family_if_exist_use:e { \l__xeCJK_family_tl }
3839   }
3840   {
3841     \bool_if:NTF #2
3842     { \xeCJK_family_if_exist_use:n {#3} }
3843     {
3844       \xeCJK_family_if_exist:nTF {#3}
3845       {
3846         \tl_set:Nn \l__xeCJK_family_tl {#3}
3847         \tl_set_eq:NN \CJK@family \l__xeCJK_fontspec_family_tl
3848         \bool_if:NT #1 { \__xeCJK_family_use:n {#3} }

```

```

3849     }
3850     { \_xeCJK_family_unknown_warning:n {#3} }
3851   }
3852 }
3853 }
3854 \cs_generate_variant:Nn \xeCJK_family:NNn { NNe }
3855 \cs_new_protected:Npn \xeCJK_switch_family:n #1
3856 {
3857   \xeCJK_family_if_exist:nTF {#1}
3858   {
3859     \tl_set:Nn \l_xeCJK_family_tl {#1}
3860     \tl_set_eq:NN \CJK@family \l_xeCJK_fontspec_family_tl
3861   }
3862   { \_xeCJK_family_unknown_warning:n {#1} }
3863 }
3864 \cs_generate_variant:Nn \xeCJK_switch_family:n { e , o }

```

PunctFamily 设置汉字标点符号的字体。

```

3865 \keys_define:nn { xeCJK / options }
3866 {
3867   PunctFamily .choice: ,
3868   PunctFamily .value_required:n = { true } ,
3869   PunctFamily / false .code:n =
3870   {
3871     \tl_set:Nn \CJK@punctfamily { \CJK@family }
3872     \tl_set:Nn \l_xeCJK_punct_family_tl { \l_xeCJK_family_tl }
3873     \xeCJK_cs_clear:N \_xeCJK_select_font:
3874     \xeCJK_cs_clear:N \_xeCJK_select_punct_font:
3875     \cs_set_eq:NN \xeCJK_select_punct_font: \xeCJK_select_font:
3876   } ,
3877   PunctFamily / unknown .code:n =
3878   { \xeCJK_punct_family:e {#1} } ,
3879 }
3880 \cs_new_protected:Npn \xeCJK_punct_family:n #1
3881 {
3882   \xeCJK_family_if_exist:nTF {#1}
3883   {
3884     \tl_set:Nn \l_xeCJK_punct_family_tl {#1}
3885     \tl_set_eq:NN \CJK@punctfamily \l_xeCJK_fontspec_family_tl
3886     \cs_set_eq:NN \_xeCJK_select_font: \xeCJK_select_font:
3887     \cs_set_eq:NN \_xeCJK_select_punct_font: \_xeCJK_select_punct_font_aux:
3888     \cs_set_eq:NN \xeCJK_select_punct_font: \_xeCJK_select_punct_font:
3889   }
3890   { \_xeCJK_family_unknown_warning:n {#1} }
3891 }
3892 \cs_generate_variant:Nn \xeCJK_punct_family:n { e }

```

`\l_xeCJK_family_tl` 用于保存文档当前正在使用的 CJK 字体族。

```
3893 \tl_new:N \l_xeCJK_family_tl
```

`\CJK@family` 用于保存实际的字体族名称。

```
3894 \tl_new:N \CJK@family
```

```

\_xeCJK_gobble_CJKfamily: 3895 \cs_new_protected:Npn \_xeCJK_gobble_CJKfamily:
3896 { \cs_set_eq:NN \CJKfamily \_xeCJK_gobble_CJKfamily:wn }
3897 \NewExpandableDocumentCommand \_xeCJK_gobble_CJKfamily:wn { t+ t- m } { }

```

```

\_xeCJK_family_if_exist_use:n 3898 \cs_new_protected:Npn \xeCJK_family_if_exist_use:n #1
3899 {
3900   \xeCJK_family_if_exist:nTF {#1}
3901   { \_xeCJK_family_use:n {#1} }
3902   { \_xeCJK_family_unknown_warning:n {#1} }
3903 }
3904 \cs_generate_variant:Nn \xeCJK_family_if_exist_use:n { e }

```

```

\__xeCJK_family_unknown_warning:n 3905 \cs_new_protected:Npn \__xeCJK_family_unknown_warning:n #1
3906 {
3907   \prop_if_empty:NF \g__xeCJK_family_font_name_prop
3908   {
3909     \seq_if_in:NnF \g__xeCJK_unknown_family_seq {#1}
3910     {
3911       \seq_gput_right:Nn \g__xeCJK_unknown_family_seq {#1}
3912       \__xeCJK_warning:ne { CJKfamily-Unknown } {#1}
3913     }
3914   }
3915 }
3916 \seq_new:N \g__xeCJK_unknown_family_seq
3917 \__xeCJK_msg_new:nn { CJKfamily-Unknown }
3918 {
3919   Unknown~CJK~family~\__xeCJK_msg_family_map:n {#1}'~is~being~ignored.\\\
3920   Try~to~use~\__xeCJK_msg_def_family_map:n {#1}'~to~define~it.
3921 }
3922 \cs_new:Npn \__xeCJK_msg_def_family_map:n #1
3923 {
3924   \str_case_e:nnF {#1}
3925   {
3926     \CJKrmdefault { \token_to_str:N \setCJKmainfont }
3927     \CJKsfdefault { \token_to_str:N \setCJKsansfont }
3928     \CJKttdefault { \token_to_str:N \setCJKmonofont }
3929   }
3930   { \token_to_str:N \setCJKfamilyfont \{ #1 \} }
3931   [<...>]\{<...>\}
3932 }
3933 \cs_new:Npn \__xeCJK_msg_family_map:n #1
3934 {
3935   \str_case_e:nnF {#1}
3936   {
3937     \CJKrmdefault { \token_to_str:N \CJKrmdefault }
3938     \CJKsfdefault { \token_to_str:N \CJKsfdefault }
3939     \CJKttdefault { \token_to_str:N \CJKttdefault }
3940   }
3941   {#1}
3942 }

```

__xeCJK_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, 类似 `fontspec` 的实现。自带展开功能, 额外参数 #4 用于后处理。

```

3943 \cs_new_protected:Npn \__xeCJK_pass_args:nnnn #1#2#3#4
3944 {
3945   \tl_if_novalue:nTF {#2}
3946   { \__xeCJK_post_arg:w {#1} {#3} {#4} }
3947   {
3948     \use:e { #1 {#2} {#3} }
3949     #4
3950   }
3951 }
3952 \NewDocumentCommand \__xeCJK_post_arg:w { m m m O {} }
3953 {
3954   \use:e { #1 {#4} {#2} }
3955   #3
3956 }

```

`\setCJKmainfont` 设置文档的 CJK 普通字体、无衬线和等宽字体。

`\setCJKsansfont`

`\setCJKmonofont`

```

3957 \NewDocumentCommand \setCJKmainfont { o m }
3958 {
3959   \__xeCJK_pass_args:nnnn
3960   { \xeCJK_set_family:nnn { \CJKrmdefault } } {#1} {#2}
3961   { \__xeCJK_preamble_family:n { \CJKrmdefault } }
3962 }
3963 \cs_new_eq:NN \setCJKromanfont \setCJKmainfont
3964 \NewDocumentCommand \setCJKsansfont { o m }
3965 {

```

```

3966   \__xeCJK_pass_args:nnnn
3967   { \xeCJK_set_family:nnn { \CJKsfdefault } } {#1} {#2}
3968   { \__xeCJK_preamble_family:n { \CJKsfdefault } }
3969   }
3970 \NewDocumentCommand \setCJKmonofont { o m }
3971 {
3972   \__xeCJK_pass_args:nnnn
3973   { \xeCJK_set_family:nnn { \CJKttdefault } } {#1} {#2}
3974   { \__xeCJK_preamble_family:n { \CJKttdefault } }
3975   }

3976 \@onlypreamble \setCJKmainfont
3977 \@onlypreamble \setCJKmathfont
3978 \@onlypreamble \setCJKsansfont
3979 \@onlypreamble \setCJKmonofont
3980 \@onlypreamble \setCJKromanfont

```

`__xeCJK_preamble_family:n` 用在 `\setCJKmainfont` 等主要命令之后, 确保导言区有 CJK 字体可用。

```

3981 \cs_new_protected:Npn \__xeCJK_preamble_family:n #1
3982 { \str_if_eq:eeT {#1} { \CJKfamilydefault } { \normalfont } }

```

`\setCJKfamilyfont` 分别用于预声明 CJK 字体族和声明并马上调用 CJK 字体族。

`\newCJKfontfamily`
`\CJKfontspec`

```

3983 \NewDocumentCommand \setCJKfamilyfont { m o m }
3984 {
3985   \__xeCJK_pass_args:nnnn
3986   { \xeCJK_set_family:nnn {#1} } {#2} {#3}
3987   { }
3988   }
3989 \__xeCJK_msg_new:nn { command-already-defined }
3990 { Control~sequence~#1~is~already~defined. }
3991 \NewDocumentCommand \newCJKfontfamily { o m o m }
3992 {
3993   \tl_set:Nc \l__xeCJK_tmp_tl
3994   { \tl_if_novalue:nTF {#1} { \cs_to_str:N #2 } {#1} }
3995   \cs_if_exist:NTF #2
3996   { \__xeCJK_error:ne { command-already-defined } { \token_to_str:N #2 } }
3997   {
3998     \cs_set_protected:Npe #2
3999     { \xeCJK_switch_family:n { \l__xeCJK_tmp_tl } }
4000   }
4001   \__xeCJK_pass_args:nnnn
4002   { \xeCJK_set_family:nnn { \l__xeCJK_tmp_tl } } {#3} {#4}
4003   { }
4004   }
4005 \NewDocumentCommand \CJKfontspec { o m }
4006 {
4007   \__xeCJK_pass_args:nnnn
4008   { \xeCJK_fontspec:nn } {#1} {#2}
4009   { \tex_ignorespaces:D }
4010   }

```

```

\xeCJK_fontspec:nn 4011 \cs_new_protected:Npn \xeCJK_fontspec:nn #1#2
4012 {
4013   \prop_get:NnNTF \g__xeCJK_fontspec_prop
4014   { CJKfontspec/#1/#2/id } \l__xeCJK_family_tl
4015   { \xeCJK_switch_family:o { \l__xeCJK_family_tl } }
4016   {
4017     \__xeCJK_fontspec:enn
4018     { CJKfontspec ( \int_eval:n { \g__xeCJK_family_int + 1 } ) }
4019     {#1} {#2}
4020   }
4021   }
4022 \cs_new_protected:Npn \__xeCJK_fontspec:nnn #1#2#3
4023 {
4024   \prop_gput:Nnn \g__xeCJK_fontspec_prop { CJKfontspec/#2/#3/id } {#1}
4025   \xeCJK_set_family:nnn {#1} {#2} {#3}

```

```

4026 \xeCJK_switch_family:n {#1}
4027 }
4028 \cs_generate_variant:Nn \xeCJK_fontspec:nn { oo }
4029 \cs_generate_variant:Nn \_xeCJK_fontspec:nnn { e }
4030 \prop_new:N \g__xeCJK_fontspec_prop

```

`\defaultCJKfontfeatures` 和 `\addCJKfontfeatures` 分别用于设置 CJK 字体的默认属性和增加当前 CJK 字体的属性。

```

4031 \clist_new:N \g__xeCJK_default_features_clist
4032 \NewDocumentCommand \defaultCJKfontfeatures { m }
4033 { \clist_gset:Nn \g__xeCJK_default_features_clist {#1} }
4034 \@onlypreamble \defaultCJKfontfeatures
4035 \NewDocumentCommand \addCJKfontfeatures { s O { } m }
4036 {
4037 \xeCJK_add_font_features:Nee #1 {#2} {#3}
4038 \tex_ignorespaces:D
4039 }
4040 \cs_new_eq:NN \addCJKfontfeature \addCJKfontfeatures
\xeCJK_add_font_features:Nnn 4041 \cs_new_protected:Npn \xeCJK_add_font_features:Nnn #1#2#3
4042 {
4043 \prop_get:NoNTF \g__xeCJK_family_font_name_prop
4044 \l_xeCJK_family_tl \l__xeCJK_font_name_tl
4045 {
4046 \clist_set:Nn \l__xeCJK_add_font_features_clist {#3}
4047 \seq_map_inline:Nn \g__xeCJK_sub_key_seq
4048 { \clist_remove_all:Nn \l__xeCJK_add_font_features_clist {##1} }
4049 \seq_clear:N \l__xeCJK_sub_key_seq
4050 \clist_clear:N \l__xeCJK_add_block_features_clist
4051 \clist_map_function:nN {#2} \_xeCJK_add_sub_features:n
4052 \bool_lazy_and:nnT
4053 {#1}
4054 { \seq_if_empty_p:N \l__xeCJK_sub_key_seq }
4055 {
4056 \seq_map_function:NN
4057 \g__xeCJK_sub_key_seq \_xeCJK_add_sub_class_features:n
4058 }
4059 \prop_get:NoNT \g__xeCJK_family_font_options_prop
4060 \l_xeCJK_family_tl \l__xeCJK_font_options_clist
4061 {
4062 \bool_lazy_or:nnT
4063 { \seq_if_empty_p:N \l__xeCJK_sub_key_seq }
4064 {#1}
4065 {
4066 \clist_concat:NNN \l__xeCJK_font_options_clist
4067 \l__xeCJK_font_options_clist \l__xeCJK_add_font_features_clist
4068 }
4069 \clist_concat:NNN \l__xeCJK_font_options_clist
4070 \l__xeCJK_font_options_clist \l__xeCJK_add_block_features_clist
4071 }
4072 \xeCJK_fontspec:oo \l__xeCJK_font_options_clist \l__xeCJK_font_name_tl
4073 }
4074 { \_xeCJK_warning:n { addCJKfontfeature-ignored } }
4075 }
4076 \cs_new_protected:Npn \_xeCJK_add_sub_features:n #1
4077 {
4078 \seq_if_in:NnTF \g__xeCJK_sub_key_seq {#1}
4079 {
4080 \seq_put_right:Nn \l__xeCJK_sub_key_seq {#1}
4081 \_xeCJK_add_sub_class_features:n {#1}
4082 }
4083 { \_xeCJK_warning:ne { SubBlock-undefined } {#1} }
4084 }
4085 \clist_new:N \l__xeCJK_add_font_features_clist
4086 \clist_new:N \l__xeCJK_add_block_features_clist
4087 \cs_generate_variant:Nn \xeCJK_add_font_features:Nnn { Nee , Nne }
4088 \_xeCJK_msg_new:nn { addCJKfontfeature-ignored }
4089 {

```

```

4090 \token_to_str:N \addCJKfontfeature (s)~ignored.\\
4091 It~cannot~be~used~with~a~font~that~wasn't~selected~by~xeCJK.
4092 }

\_xeCJK_add_sub_class_features:n 4093 \cs_new_protected:Npn \__xeCJK_add_sub_class_features:n #1
4094 {
4095 \prop_get:NoNTF \g__xeCJK_family_font_name_prop
4096 { \l_xeCJK_family_tl/#1 } \l__xeCJK_sub_font_name_tl
4097 {
4098 \prop_get:NoN \g__xeCJK_family_font_options_prop
4099 { \l_xeCJK_family_tl/#1 } \l__xeCJK_sub_font_options_clist
4100 }
4101 {
4102 \prop_get:NeNTF \g__xeCJK_family_font_name_prop
4103 { \CJKfamilydefault/#1 } \l__xeCJK_sub_font_name_tl
4104 {
4105 \prop_get:NeN \g__xeCJK_family_font_options_prop
4106 { \CJKfamilydefault/#1 } \l__xeCJK_sub_font_options_clist
4107 }
4108 {
4109 \prop_get:NoN \g__xeCJK_family_font_options_prop
4110 \l_xeCJK_family_tl \l__xeCJK_sub_font_options_clist
4111 \tl_set_eq:NN \l__xeCJK_sub_font_name_tl \l__xeCJK_font_name_tl
4112 }
4113 }
4114 \clist_concat:NNN \l__xeCJK_sub_font_options_clist
4115 \l__xeCJK_sub_font_options_clist \l__xeCJK_add_font_features_clist
4116 \clist_put_right:Ne \l__xeCJK_add_block_features_clist
4117 {
4118 #1 =
4119 {
4120 [ \exp_not:o \l__xeCJK_sub_font_options_clist ]
4121 { \exp_not:o \l__xeCJK_sub_font_name_tl }
4122 }
4123 }
4124 }
4125 \cs_generate_variant:Nn \prop_get:NnN { Ne }
4126 \prg_generate_conditional_variant:Nnn \prop_get:NnN { Ne } { T , TF }

LoadFandol 4127 \keys_define:nn { xeCJK / options }
4128 { LoadFandol .bool_gset:N = \g__xeCJK_fandol_bool }
4129 \cs_new_protected:Npn \__xeCJK_load_fandol:
4130 {
4131 \xeCJK_set_family:enn { \CJKrmdefault }
4132 { Extension = .otf , BoldFont = FandolSong-Bold , ItalicFont = FandolKai-Regular }
4133 { FandolSong-Regular }
4134 \xeCJK_set_family:enn { \CJKsfdefault }
4135 { Extension = .otf , BoldFont = FandolHei-Bold } { FandolHei-Regular }
4136 \xeCJK_set_family:enn { \CJKttdefault }
4137 { Extension = .otf } { FandolFang-Regular }
4138 }

```

在导言区结束的时候,若没有声明CJK字体,则给出一个警告。如果\CJKfamilydefault没有被更改,则在此时根据西文字体的情况更新\CJKfamilydefault。如果\CJKfamilydefault对应的字体族没有定义,则使用\CJKrmdefault作为默认字体族。若\CJKrmdefault也没有定义,则使用在导言区设置的第一个CJK字体作为默认字体族。最后设置数学字体。

```

4139 \__xeCJK_at_end_preamble:n
4140 {
4141 \tl_if_eq:NNT \CJKfamilydefault \l__xeCJK_family_default_init_tl
4142 {
4143 \group_begin:
4144 \cs_set_eq:NN \__xeCJK_family_default_wrap:n \exp_not:n
4145 \tl_gset:Ne \CJKfamilydefault
4146 {
4147 \str_case:onF { \familydefault }
4148 {

```

```

4149         { \rmdefault } { \exp_not:N \CJKrmdefault }
4150         { \sfdefault } { \exp_not:N \CJKsfdefault }
4151         { \ttdefault } { \exp_not:N \CJKttdefault }
4152     }
4153     { \CJKfamilydefault }
4154 }
4155 \group_end:
4156 }
4157 \prop_if_empty:NTF \g__xeCJK_family_font_name_prop
4158 {
4159     \bool_if:NTF \g__xeCJK_fandol_bool
4160     {
4161         \__xeCJK_warning:n { fandol }
4162         \__xeCJK_load_fandol:
4163         \xeCJK_ensure_default_family:
4164     }
4165     { \__xeCJK_warning:ne { no-CJKfamily } { \CJKfamilydefault } }
4166 }
4167 { \xeCJK_ensure_default_family: }
4168 }
4169 \cs_new_protected:Npn \xeCJK_ensure_default_family:
4170 {
4171     \xeCJK_family_if_exist:eF { \CJKfamilydefault }
4172     {
4173         \tl_set_eq:NN \l__xeCJK_tmp_tl \CJKfamilydefault
4174         \str_if_eq:eeTF { \CJKfamilydefault } { \CJKrmdefault }
4175         { \use:n }
4176         {
4177             \xeCJK_family_if_exist:eTF { \CJKrmdefault }
4178             { \tl_gset:Nn \CJKfamilydefault { \CJKrmdefault } }
4179         }
4180     }
4181     \prop_map_inline:Nn \g__xeCJK_family_font_name_prop
4182     {
4183         \prop_map_break:n
4184         { \tl_gset_rescan:Nnn \CJKfamilydefault { } { ##1 } }
4185     }
4186 }
4187 \__xeCJK_warning:nee { CJKfamilydefault-undefined }
4188 { \l__xeCJK_tmp_tl } { \CJKfamilydefault }
4189 }
4190 \xeCJK_switch_family:e { \CJKfamilydefault }
4191 \bool_if:NT \g__xeCJK_math_bool { \xeCJK_set_mathfont: }
4192 }
4193 \__xeCJK_msg_new:nn { no-CJKfamily }
4194 {
4195     It~seems~that~you~have~not~declared~a~CJKfamily.\\
4196     If~you~want~to~use~xeCJK~in~the~right~way,~you~should~use\\\\\\
4197     `\\__xeCJK_msg_def_family_map:n {#1}'\\\\\\
4198     in~the~preamble~to~declare~the~default~CJKfamily.\\
4199 }
4200 \__xeCJK_msg_new:nn { CJKfamilydefault-undefined }
4201 {
4202     Undefined~CJK~default~family~`\\__xeCJK_msg_family_map:n {#1}'~
4203     has~been~replaced~by~`\\__xeCJK_msg_family_map:n {#2}'~.\\\\\\
4204     Try~to~use~`\\__xeCJK_msg_def_family_map:n {#1}'~to~define~it.
4205 }
4206 \__xeCJK_msg_new:nn { fandol }
4207 {
4208     Fandol~is~being~set~as~the~default~font~for~CJK~text.\\
4209     Please~make~sure~it~has~been~properly~installed.
4210 }

```

5.14 数学字体设置

CJKmath 是否启用 CJK 数学字体的宏包选项。

```
4211 \keys_define:nn { xeCJK / options } { CJKmath .bool_gset:N = \g__xeCJK_math_bool }
```

\setCJKmathfont 设置 CJK 数学字体。

```
4212 \NewDocumentCommand \setCJKmathfont { o m }
4213 {
4214   \__xeCJK_pass_args:nnnn
4215   { \xeCJK_set_family:nnn { \c__xeCJK_math_tl } } {#1} {#2}
4216   { }
4217 }
4218 \tl_const:Nn \c__xeCJK_math_tl { CJKmath }
```

\xeCJK_set_mathfont: 当没有设置 CJK 数学字体时,使用 \CJKfamilydefault 作为数学字体。

```
4219 \cs_new_protected:Npn \xeCJK_set_mathfont:
4220 {
4221   \cs_if_exist_use:N \__xeCJK_save_um_char:
4222   \xeCJK_family_if_exist:eTF { \c__xeCJK_math_tl }
4223   { \__xeCJK_set_mathfont_aux: }
4224   {
4225     \xeCJK_family_if_exist:eT { \CJKfamilydefault }
4226     {
4227       \__xeCJK_copy_family:ee { \c__xeCJK_math_tl } { \CJKfamilydefault }
4228       \__xeCJK_set_mathfont_aux:
4229     }
4230   }
4231   \cs_if_exist_use:N \__xeCJK_restore_um_char:
4232 }
4233 \cs_new_protected:Npn \__xeCJK_set_mathfont_aux:
4234 {
4235   \tl_const:Ne \c__xeCJK_math_family_tl
4236   { \l__xeCJK_fontspec_family_tl }
4237   \xeCJK_declare_mathfont:ee
4238   { \c__xeCJK_math_tl }
4239   { \c__xeCJK_math_family_tl }
4240   \int_const:Nn \c_xeCJK_math_fam_int
4241   { \use:c { sym \c__xeCJK_math_tl } }
4242   \clist_gconcat:NNN \g__xeCJK_math_chars_clist
4243   \g__xeCJK_CJK_range_clist \g__xeCJK_FullLeft_range_clist
4244   \clist_gconcat:NNN \g__xeCJK_math_chars_clist
4245   \g__xeCJK_math_chars_clist \g__xeCJK_FullRight_range_clist
4246   \xeCJK_gset_mathcode:Nn \g__xeCJK_math_chars_clist
4247   { \c_xeCJK_math_fam_int }
4248   \xeCJK_set_mathfont_block:
4249 }
4250 \clist_new:N \g__xeCJK_math_chars_clist
4251 \prop_new:N \g__xeCJK_fam_prop
```

\xeCJK_set_mathfont_block: 分区数学字体。

```
4252 \cs_new_protected:Npn \xeCJK_set_mathfont_block:
4253 {
4254   \seq_if_empty:NF \g__xeCJK_CJK_sub_class_seq
4255   {
4256     \seq_map_function:NN
4257     \g__xeCJK_CJK_sub_class_seq
4258     \xeCJK_set_mathfont_block:n
4259   }
4260 }
4261 \cs_new_protected:Npn \xeCJK_set_mathfont_block:n #1
4262 {
4263   \xeCJK_block_family:nn { \c__xeCJK_math_tl } {#1}
4264   \prop_get:NoNTF \g__xeCJK_fam_prop
4265   \l__xeCJK_fontspec_family_tl \l__xeCJK_tmp_tl
```

```

4266     { \int_set:Nn \l__xeCJK_fam_int { \l__xeCJK_tmp_tl } }
4267     {
4268       \xeCJK_declare_mathfont:ee
4269       { \c__xeCJK_math_tl / #1 }
4270       { \l__xeCJK_fontspec_family_tl }
4271       \__xeCJK_set_mathfont_block_aux:cn
4272       { sym \c__xeCJK_math_tl / #1 } {#1}
4273     }
4274     \xeCJK_gset_mathcode:cn { g__xeCJK_CJK/#1_range_clist } { \l__xeCJK_fam_int }
4275   }
4276 \cs_new_protected:Npn \__xeCJK_set_mathfont_block_aux:Nn #1#2
4277 {
4278   \int_set_eq:NN \l__xeCJK_fam_int #1
4279   \prop_gput:Nnn \g__xeCJK_block_fam_prop {#2} {#1}
4280 }
4281 \int_new:N \l__xeCJK_fam_int
4282 \prop_new:N \g__xeCJK_block_fam_prop
4283 \cs_generate_variant:Nn \__xeCJK_set_mathfont_block_aux:Nn { c }

```

`\xeCJK_declare_mathfont:nn` 注意从 L^AT_EX 2_ε 2020/02/02 开始, `\shapedefault` 初始值是 n, 而 `\updefault` 初始值是 up, 两者并不一致。fontspec 包定义字体使用的是 `\shapedefault`。

```

4284 \cs_new_protected:Npn \xeCJK_declare_mathfont:nn #1#2
4285 {
4286   \xeCJK_declare_symbol_font:nnnnn {#1} { \c__xeCJK_encoding_tl }
4287   {#2} { \mddefault } { \shapedefault }
4288   \cs_if_free:cF
4289   { \c__xeCJK_encoding_tl/#2/\bfdefault/\shapedefault }
4290   {
4291     \SetSymbolFont {#1} { bold } { \c__xeCJK_encoding_tl }
4292     {#2} { \bfdefault } { \shapedefault }
4293   }
4294   \prop_gput:Nne \g__xeCJK_fam_prop {#2} { \exp_not:c { sym #1 } }
4295 }
4296 \cs_generate_variant:Nn \prop_put:Nnn { Nne }
4297 \cs_generate_variant:Nn \prop_gput:Nnn { Nne }
4298 \cs_generate_variant:Nn \xeCJK_declare_mathfont:nn { ee }

```

`\xeCJK_declare_symbol_font:nnnnn` 主要功能同 `\DeclareSymbolFont`, 不带编码和重复定义检查。

```

4299 \cs_new_protected:Npn \xeCJK_declare_symbol_font:nnnnn #1
4300 { \__xeCJK_declare_symbol_font:cnnnn { sym #1 } }
4301 \cs_new_protected:Npn \__xeCJK_declare_symbol_font:Nnnnn #1
4302 {
4303   \xeCJK_new_fam:N #1
4304   \xeCJK_new_symbol_font:Nnnnn #1
4305 }
4306 \cs_generate_variant:Nn \__xeCJK_declare_symbol_font:Nnnnn { c }

```

`\xeCJK_new_fam:N` 我们从 255 往下分配 `\fam`, `\count18` 是 L^AT_EX 2_ε 记录最后分配的 `\fam` 编号, 作为我们的分配器的下限。事实上, 还应该相应地减小 `\e@mathgroup@top` 才合理, 但这可能会有不利影响, 我们暂未处理。

```

4307 \cs_new_protected:Npn \xeCJK_new_fam:N #1
4308 {
4309   \int_compare:nNnTF
4310   { \g__xeCJK_fam_allocation_int } > { \g__xeCJK_fam_bottom_int }
4311   {
4312     \int_set_eq:NN \allocationnumber \g__xeCJK_fam_allocation_int
4313     \int_const:Nn #1 { \allocationnumber }
4314     \iow_log:e
4315     {
4316       \token_to_str:N #1 =
4317       \token_to_str:N \mathgroup \int_use:N \allocationnumber
4318     }
4319     \int_gdecr:N \g__xeCJK_fam_allocation_int

```

```

4320     }
4321     { \__xeCJK_error:n { fam-exhausted } }
4322   }
4323 \tex_countdef:D \g__xeCJK_fam_bottom_int = 18 ~
4324 \int_new:N \g__xeCJK_fam_allocation_int
4325 \int_gset:Nn \g__xeCJK_fam_allocation_int { 255 }
4326 \__xeCJK_msg_new:nn { fam-exhausted }
4327 { No~room~for~a~new~fam. }

```

功能同 `\new@symbolfont`，但我们不增加 `\c@mv@normal` 和 `\c@mv@bold` 之类的计数器。

```

\__xeCJK_new_symbol_font:Nnnnn \new@symbolfont
\__xeCJK_new_symbol_font:NN
4328 \cs_new_protected:Npn \xeCJK_new_symbol_font:Nnnnn #1#2#3#4#5
4329 { \__xeCJK_new_symbol_font:Nc #1 { #2/#3/#4/#5 } }
4330 \cs_new_protected:Npn \__xeCJK_new_symbol_font:NN #1#2
4331 {
4332   \tl_put_right:Nn \group@list { \group@elt #1 #2 }
4333   \cs_set:Npn \version@elt ##1
4334     { \tl_put_right:Nn ##1 { \getanddefine@fonts #1 #2 } }
4335   \version@list
4336 }
4337 \cs_generate_variant:Nn \__xeCJK_new_symbol_font:NN { Nc }

```

CJK 字符的数学类别固定为 0(`\mathord`)。

```

\xeCJK_gset_mathcode:Nn \xeCJK_gset_mathcode:Nnn
\xeCJK_gset_mathcode:Nnnn
4338 \cs_new_protected:Npn \xeCJK_gset_mathcode:Nn #1#2
4339 {
4340   \clist_map_inline:Nn #1
4341     {
4342       \__xeCJK_set_char_class_aux:Nnw \xeCJK_gset_mathcode:nnnn { ##1 }
4343       { 0 } {#2}
4344     }
4345 }
4346 \cs_generate_variant:Nn \xeCJK_gset_mathcode:Nn { c }
4347 \cs_new_protected:Npn \xeCJK_gset_mathcode:nnnn #1#2#3#4
4348 {
4349   \__xeCJK_check_num_range:nnNN {#1} {#2} \l__xeCJK_begin_int \l__xeCJK_end_int
4350   \xeCJK_int_until_do:nn { \l__xeCJK_begin_int > \l__xeCJK_end_int }
4351   {
4352     \xeCJK_gset_mathcode:Nnn \l__xeCJK_begin_int {#3} {#4}
4353     \int_incr:N \l__xeCJK_begin_int
4354   }
4355 }
4356 \cs_new_protected:Npn \xeCJK_gset_mathcode:Nnn #1#2#3
4357 { \tex_global:D \tex_Umathcode:D #1 = #2 ~ #3 ~ #1 }

```

5.15 抄录环境中的间距调整

Verb 如果设置为 `env`，则只在 \LaTeX 的抄录环境里使用 `\xeCJKVerbAddon`，而不包括 `\verb`。对当前使用环境的判断基于在标准 \LaTeX 的环境定义里使用 `\begingroup` 和 `\endgroup` 来分组。

```

4358 \int_new:N \l__xeCJK_verb_case_int
4359 \keys_define:nn { xeCJK / options }
4360 {
4361   Verb .choices:nn =
4362     { true , env+ , env , false }
4363     { \int_set_eq:NN \l__xeCJK_verb_case_int \l_keys_choice_int } ,
4364   Verb .default:n = { env }
4365 }
4366 \cs_new_protected:Npn \__xeCJK_verb_font_hook:
4367 {
4368   \if_case:w \l__xeCJK_verb_case_int
4369   \or:
4370     \__xeCJK_nobreak_skip_zero:
4371   \or:
4372     \int_compare:nNnTF \tex_currentgrouptype:D = { 14 }
4373     { \xeCJKVerbAddon }
4374     { \__xeCJK_nobreak_skip: }

```

```

4375 \or:
4376 \int_compare:nNnTF \tex_currentgrouptype:D = { 14 }
4377 { \xeCJKVerbAddon }
4378 { \__xeCJK_nobreak_skip_zero: }
4379 \fi:
4380 }
4381 \__xeCJK_after_preamble:n
4382 {
4383 \cs_set_protected:Npe \verbatim@font
4384 { \exp_not:o { \verbatim@font } \__xeCJK_verb_font_hook: }
4385 }
\__xeCJK_nobreak_skip_zero: 4386 \cs_new_protected:Npn \__xeCJK_nobreak_skip_zero:
\__xeCJK_nobreak_skip: 4387 {
4388 \__xeCJK_reset_shipout_skip:
4389 \cs_set_eq:NN \__xeCJK_shipout_check_for_glue: \xeCJK_check_for_glue:
4390 \cs_set_eq:NN \__xeCJK_shipout_boundary:w \xeCJK_CJK_and_Boundary:w
4391 \tl_put_right:Nn \l__xeCJK_reset_shipout_skip_hook_tl
4392 {
4393 \cs_set_eq:NN \xeCJK_check_for_glue: \__xeCJK_shipout_check_for_glue:
4394 \cs_set_eq:NN \xeCJK_CJK_and_Boundary:w \__xeCJK_shipout_boundary:w
4395 }
4396 \xeCJK_cs_clear:N \CJKglue
4397 \xeCJK_cs_clear:N \CJKecglue
4398 \xeCJK_cs_clear:N \xeCJK_check_for_glue:
4399 \cs_set_eq:NN \xeCJK_CJK_and_Boundary:w \xeCJK_class_group_end:
4400 \cs_set_eq:NN \__xeCJK_punct_hskip:n \__xeCJK_nobreak_hskip:n
4401 \cs_set_eq:NN \__xeCJK_punct_breakable_kern:n \__xeCJK_nobreak_hskip:n
4402 }
4403 \cs_new_protected:Npn \__xeCJK_nobreak_skip:
4404 {
4405 \__xeCJK_reset_shipout_skip:
4406 \xeCJK_glue_to_skip:nN { \CJKglue } \l__xeCJK_ccglue_skip
4407 \skip_if_eq:nnTF { \l__xeCJK_ccglue_skip } { \c_zero_skip }
4408 { \xeCJK_cs_clear:N \CJKglue }
4409 { \cs_set_eq:NN \CJKglue \__xeCJK_nobreak_ccglue: }
4410 \xeCJK_glue_to_skip:nN { \CJKecglue } \l__xeCJK_ecglue_skip
4411 \skip_if_eq:nnTF { \l__xeCJK_ecglue_skip } { \c_zero_skip }
4412 { \xeCJK_cs_clear:N \CJKecglue }
4413 { \cs_set_eq:NN \CJKecglue \__xeCJK_nobreak_ecglue: }
4414 \cs_set_eq:NN \__xeCJK_punct_hskip:n \__xeCJK_nobreak_hskip:n
4415 \cs_set_eq:NN \__xeCJK_punct_breakable_kern:n \__xeCJK_nobreak_hskip:n
4416 }
4417 \cs_new_protected:Npn \__xeCJK_nobreak_ccglue:
4418 { \xeCJK_no_break: \skip_horizontal:N \l__xeCJK_ccglue_skip }
4419 \cs_new_protected:Npn \__xeCJK_nobreak_ecglue:
4420 { \xeCJK_no_break: \skip_horizontal:N \l__xeCJK_ecglue_skip }
\__xeCJK_reset_shipout_skip: 4421 \cs_new_protected:Npn \__xeCJK_reset_shipout_skip:
4422 {
4423 \cs_set_eq:NN \__xeCJK_shipout_CJKglue: \CJKglue
4424 \cs_set_eq:NN \__xeCJK_shipout_CJKecglue: \CJKecglue
4425 \cs_set_eq:NN \__xeCJK_shipout_punct_hskip:n \__xeCJK_punct_hskip:n
4426 \cs_set_eq:NN
4427 \__xeCJK_shipout_punct_breakable_kern:n \__xeCJK_punct_breakable_kern:n
4428 \tl_set:Nc \l__xeCJK_off_verb_addon_tl
4429 {
4430 \bool_if:NTF \l__xeCJK_xecglue_bool
4431 { \keys_set:nn { xeCJK / options } { xCJKecglue = true } }
4432 { \keys_set:nn { xeCJK / options } { xCJKecglue = false } }
4433 \exp_not:n
4434 {
4435 \cs_set_eq:NN \CJKglue \__xeCJK_shipout_CJKglue:
4436 \cs_set_eq:NN \CJKecglue \__xeCJK_shipout_CJKecglue:
4437 \cs_set_eq:NN \__xeCJK_punct_hskip:n \__xeCJK_shipout_punct_hskip:n
4438 \cs_set_eq:NN \__xeCJK_punct_breakable_kern:n
4439 \__xeCJK_shipout_punct_breakable_kern:n
4440 \l__xeCJK_reset_shipout_skip_hook_tl

```

```

4441     }
4442   }
4443   \xeCJK_add_to_shipout:n { \l__xeCJK_off_verb_addon_tl }
4444   \keys_set:nn { xeCJK / options } { xCJKecglue = false }
4445 }
4446 \tl_new:N \l__xeCJK_reset_shipout_skip_hook_tl

```

`\xeCJKOffVerbAddon` `\xeCJKVerbAddon` 进行了比较大的调整,应该只在分组环境里使用。为了方便调整间距以利于对齐,这里只把字符分成了两类,并且在 CJK 类与边界(空格)之间也插入 `\CJKecglue`。以字母“M”的宽度是否等于 `\fontdimen2` 来判断当前字体是否是等宽字体。如果不是等宽字体,则设置间距为零或正文间距。

```

4447 \NewDocumentCommand \xeCJKVerbAddon { }
4448 {
4449   \int_compare:nNnF \tex_currentgrouplevel:D = \c_zero_int
4450   {
4451     \bool_if:NF \l__xeCJK_listings_env_bool
4452     {
4453       \dim_compare:nNnTF
4454         { \tex_fontdimen:D 2 ~ \tex_font:D } =
4455         { \tex_fontcharwd:D \tex_font:D \c__xeCJK_mono_letter_int }
4456         {
4457           \__xeCJK_set_verb_exspace:
4458           \__xeCJK_verb_addon:
4459         }
4460         {
4461           \int_if_odd:nTF { \l__xeCJK_verb_case_int }
4462             { \__xeCJK_nobreak_skip_zero: }
4463             { \__xeCJK_nobreak_skip: }
4464         }
4465     }
4466   }
4467 }
4468 \int_const:Nn \c__xeCJK_mono_letter_int { 77 }
4469 \bool_new:N \l__xeCJK_listings_env_bool
4470 \NewDocumentCommand \xeCJKOffVerbAddon { }
4471 { \tl_use:N \l__xeCJK_off_verb_addon_tl }
4472 \tl_new:N \l__xeCJK_off_verb_addon_tl
4473 \cs_new_protected:Npn \__xeCJK_verb_addon:
4474 {
4475   \bool_if:NF \l__xeCJK_verb_addon_bool
4476   { \__xeCJK_verb_addon_action: }
4477   \skip_if_eq:nnTF { \l__xeCJK_verb_exspace_skip } { \c_zero_skip }
4478   {
4479     \xeCJK_cs_clear:N \CJKglue
4480     \xeCJK_cs_clear:N \CJKecglue
4481   }
4482   {
4483     \skip_set_eq:NN \l__xeCJK_ccglue_skip \l__xeCJK_verb_exspace_skip
4484     \skip_set:Nn \l__xeCJK_ecglue_skip { \l__xeCJK_verb_exspace_skip / 2 }
4485     \cs_set_eq:NN \CJKglue \__xeCJK_nobreak_ccglue:
4486     \cs_set_eq:NN \CJKecglue \__xeCJK_nobreak_ecglue:
4487   }
4488   \cs_set_eq:NN \xeCJK_check_for_glue: \CJKecglue
4489   \cs_set_eq:NN \xeCJK_CJK_and_Boundary:w \__xeCJK_verb_CJK_and_Boundary:w
4490 }
4491 \cs_new_protected:Npn \__xeCJK_verb_addon_action:
4492 {
4493   \bool_set_true:N \l__xeCJK_verb_addon_bool
4494   \__xeCJK_set_char_class_eq:nn { FullLeft } { CJK }
4495   \__xeCJK_set_char_class_eq:nn { FullRight } { CJK }
4496   \__xeCJK_set_char_class_eq:nn { HalfLeft } { Default }
4497   \__xeCJK_set_char_class_eq:nn { HalfRight } { Default }
4498   \__xeCJK_set_char_class_eq:nn { NormalSpace } { Default }
4499   \cs_set_eq:NN \__xeCJK_shipout_CJKglue: \CJKglue
4500   \cs_set_eq:NN \__xeCJK_shipout_CJKecglue: \CJKecglue

```

```

4501 \cs_set_eq:NN \__xeCJK_shipout_check_for_glue: \xeCJK_check_for_glue:
4502 \cs_set_eq:NN \__xeCJK_shipout_boundary:w \xeCJK_CJK_and_Boundary:w
4503 \cs_set_protected:Npe \xeCJKOffVerbAddon
4504 {
4505   \__xeCJK_reset_char_class:n { FullLeft }
4506   \__xeCJK_reset_char_class:n { FullRight }
4507   \__xeCJK_reset_char_class:n { HalfLeft }
4508   \__xeCJK_reset_char_class:n { HalfLeft }
4509   \__xeCJK_reset_char_class:n { NormalSpace }
4510   \bool_if:NTF \l__xeCJK_xecglue_bool
4511     { \keys_set:nn { xeCJK / options } { xCJKecglue = true } }
4512     { \keys_set:nn { xeCJK / options } { xCJKecglue = false } }
4513   \exp_not:n
4514     {
4515       \cs_set_eq:NN \CJKglue \__xeCJK_shipout_CJKglue:
4516       \cs_set_eq:NN \CJKecglue \__xeCJK_shipout_CJKecglue:
4517       \cs_set_eq:NN \xeCJK_check_for_glue: \__xeCJK_shipout_check_for_glue:
4518       \cs_set_eq:NN \xeCJK_CJK_and_Boundary:w \__xeCJK_shipout_boundary:w
4519     }
4520   }
4521   \xeCJK_add_to_shipout:n { \xeCJKOffVerbAddon }
4522   \keys_set:nn { xeCJK / options } { xCJKecglue = false }
4523 }
4524 \cs_new_protected:Npn \__xeCJK_verb_CJK_and_Boundary:w
4525 { \xeCJK_class_group_end: \CJKecglue }
4526 \cs_new_protected:Npn \__xeCJK_reset_char_class:n #1
4527 {
4528   \int_set:Nn \l__xeCJK_tmp_int { \xeCJK_class_num:n {#1} }
4529   \clist_map_inline:cn { c__xeCJK_#1_chars_clist }
4530     { \tex_XeTeXcharclass:D ##1 = \l__xeCJK_tmp_int }
4531 }
4532 \bool_new:N \l__xeCJK_verb_addon_bool
4533 \cs_new_eq:NN \CJKfixedspacing \xeCJKVerbAddon

```

`__xeCJK_set_verb_exspace:` 在抄录环境中, CJK 文字之间的间距为当前西文字体两个空格的宽度与当前字体大小之差, 而与西文和空格的间距为 CJK 文字之间的间距的一半。

```

4534 \cs_new_protected:Npn \__xeCJK_set_verb_exspace:
4535 {
4536   \tl_if_exist:cTF { xeCJK/verb/\CJK@family/\curr@fontshape/\f@size }
4537     {
4538       \skip_set:Nn \l__xeCJK_verb_exspace_skip
4539         { \use:c { xeCJK/verb/\CJK@family/\curr@fontshape/\f@size } }
4540     }
4541     {
4542       \tl_set:Ne \l__xeCJK_current_coor_tl { \CJK@family/\curr@fontshape }
4543       \prop_get:NoNTF \g__xeCJK_scale_family_prop
4544         \l__xeCJK_current_coor_tl \l__xeCJK_family_tl
4545         {
4546           \xeCJK_switch_family:o { \l__xeCJK_family_tl }
4547           \skip_zero:N \l__xeCJK_verb_exspace_skip
4548         }
4549       {
4550         \group_begin: \xeCJK_select_font: \exp_args:NNo \group_end:
4551         \__xeCJK_set_verb_exspace:n
4552           { \dim_use:N \tex_fontcharwd:D \tex_font:D "4E00 ~ }
4553       }
4554     }
4555 }
4556 \skip_new:N \l__xeCJK_verb_exspace_skip

```

`__xeCJK_set_verb_exspace:n` 当两个西文空格的宽度小于一个 CJK 文字的宽度时, 对目前使用的 CJK 字体进行适当缩小。

```

4557 \cs_new_protected:Npn \__xeCJK_set_verb_exspace:n #1
4558 {
4559   \skip_set:Nn \l__xeCJK_verb_exspace_skip
4560     { 2 \tex_fontdimen:D 2 ~ \tex_font:D - #1 }
4561   \dim_compare:nNnTF \l__xeCJK_verb_exspace_skip < \c_zero_dim

```

```

4562 {
4563   \skip_zero:N \l__xeCJK_verb_exspace_skip
4564   \exp_args:Nee \__xeCJK_set_verb_scale:nn
4565   { \dim_to_fp:n { 2 \tex_fontdimen:D 2 ~ \tex_font:D } }
4566   { \dim_to_fp:n { #1 } }
4567 }
4568 {
4569   \tl_const:ce { xeCJK/verb/\CJK@family/\curr@fontshape/\f@size }
4570   { \skip_use:N \l__xeCJK_verb_exspace_skip }
4571 }
4572 }

```

__xeCJK_set_verb_scale:nn 缩小 CJK 字体,并保存相关信息。

```

4573 \cs_new_protected:Npn \__xeCJK_set_verb_scale:nn #1#2
4574 {
4575   \fp_set:Nn \l__xeCJK_scale_factor_fp { #1 / #2 }
4576   \__xeCJK_warning:nee { scale-factor }
4577   { \fp_eval:n { trunc ( \l__xeCJK_scale_factor_fp , 4 ) } }
4578   { \fp_eval:n { ceil ( #2 / #1 , 4 ) } }
4579   \xeCJK_add_font_features:Nne \c_true_bool
4580   { } { Scale = { \fp_use:N \l__xeCJK_scale_factor_fp } }
4581   \prop_gput:Noo \g__xeCJK_scale_family_prop
4582   \l__xeCJK_current_coor_tl \l__xeCJK_family_tl
4583 }
4584 \__xeCJK_msg_new:nn { scale-factor }
4585 {
4586   \token_to_str:N \xeCJKVerbAddon'~may~not~work~properly.\\\
4587   You~may~set~~Scale=#1'~to~CJKfamily~
4588   \__xeCJK_msg_family_map:n { \l__xeCJK_family_tl }',\\
4589   or~set~~Scale=#2'~to~family~
4590   \str_if_eq:eeTF \f@family \ttdefault
4591   { \token_to_str:N \ttdefault } { \f@family }'.
4592 }
4593 \fp_new:N \l__xeCJK_scale_factor_fp
4594 \prop_new:N \g__xeCJK_scale_family_prop

```

\xeCJK_visible_space: 如果文档不使用 EU1 作为默认字体编码,那么默认的打字机字体族很可能是传统的 T_EX 字体,
 \@setupverbvisiblespace 这时可视空格按照 OT1 编码传统一般就是字体中的 \char32。

```

4595 \cs_new_protected:Npn \xeCJK_setup_visible_space:
4596 {
4597   \xeCJK_make_boundary:
4598   \xeCJK_glyph_if_exist:NTF { ~~~~2423 }
4599   { \tl_set:Nn \l__xeCJK_visible_space_tl { ~~~~2423 } }
4600   {
4601     \int_compare:nNnTF { \tex_XeTeXfonttype:D \tex_font:D } = \c_zero_int
4602     {
4603       \tl_set:Ne \l__xeCJK_visible_space_tl
4604       {
4605         \str_if_eq:eeTF { \f@family } { \ttdefault }
4606         { \c_catcode_other_space_tl }
4607         { \exp_not:N \textvisiblespace }
4608       }
4609     }
4610     { \__xeCJK_visible_space_fallback: }
4611   }
4612   \cs_set_eq:NN \@xobeysp \l__xeCJK_visible_space_tl
4613 }
4614 \tl_new:N \l__xeCJK_visible_space_tl
4615 \cs_set_eq:NN \@setupverbvisiblespace \xeCJK_setup_visible_space:

```

__xeCJK_visible_space_fallback: 我们使用 lmtt 字体中的可视空格符号(U+2423)作为当前字体中相应符号的后备,但是 lmtt 的字体大小未必与当前字体匹配。因此,这里需要做一些调整,以保证使用后备可视空格符号时,也能保证对齐。

```

4616 \cs_new_protected:Npn \__xeCJK_visible_space_fallback:

```

```

4617 {
4618   \exp_args:Nc \__xeCJK_visible_space_fallback_auxi:N
4619   { xeCJK/space/\curr@fontshape/\f@size }
4620 }
4621 \cs_new_protected:Npn \__xeCJK_visible_space_fallback_auxi:N #1
4622 {
4623   \cs_if_exist:NF #1
4624   { \__xeCJK_visible_space_fallback_auxii:N #1 }
4625   \tl_set:Nn \l__xeCJK_visible_space_tl {#1}
4626 }

```

`__xeCJK_visible_space_fallback_auxii:N` 当前字体空格的宽度与后备字体 `lmtt` 不一样时,就对 `\textvisiblespace` 的字体尺寸按相应的比例放缩。

```

4627 \cs_new_protected:Npn \__xeCJK_visible_space_fallback_auxii:N #1
4628 {
4629   \group_begin:
4630   \exp_args:No \__xeCJK_set_visible_space_size:n
4631   { \dim_use:N \tex_fontdimen:D 2 ~ \tex_font:D }
4632   \cs_new_protected:Npe #1
4633   { \group_begin: \tex_the:D \tex_font:D ~~~~~2423 \group_end: }
4634   \group_end:
4635 }
4636 \cs_new_protected:Npn \__xeCJK_set_visible_space_size:n #1
4637 {
4638   \fontencoding { \UnicodeEncodingName }
4639   \tl_set:Nn \f@family { lmtt }
4640   \selectfont
4641   \dim_compare:nNnF {#1} = { \tex_fontdimen:D 2 ~ \tex_font:D }
4642   {
4643     \fontsize
4644     {
4645       \dim_eval:n
4646       {
4647         \f@size pt *
4648         \dim_ratio:nn {#1} { \tex_fontdimen:D 2 ~ \tex_font:D }
4649       }
4650     }
4651     { \f@baselineskip }
4652     \selectfont
4653   }
4654 }

```

5.16 xeCJK 其他选项

LocalConfig 声明载入本地配置文件的选项。

```

4655 \keys_define:nn { xeCJK / options }
4656 {
4657   LocalConfig .choice: ,
4658   LocalConfig / false .code:n =
4659   { \bool_gset_false:N \g__xeCJK_config_bool } ,
4660   LocalConfig / true .code:n =
4661   {
4662     \bool_gset_true:N \g__xeCJK_config_bool
4663     \tl_gset:Nn \g__xeCJK_config_name_tl { xeCJK }
4664   } ,
4665   LocalConfig / unknown .code:n =
4666   {
4667     \bool_gset_true:N \g__xeCJK_config_bool
4668     \tl_gset:Ne \g__xeCJK_config_name_tl { xeCJK - \l_keys_value_tl }
4669   } ,
4670   LocalConfig .default:n = { true }
4671 }
4672 \tl_new:N \g__xeCJK_config_name_tl
4673 \bool_new:N \g__xeCJK_config_bool

```

CJKnumber CJKnumber 和 indentfirst 是过时选项。
indentfirst

```

4674 \keys_define:nn { xeCJK / options }
4675 {
4676   CJKnumber      .code:n =
4677     { \__xeCJK_warning:nee { option-deprecated } { \l_keys_key_str } { CJKnumb } } ,
4678   indentfirst    .code:n =
4679     { \__xeCJK_warning:nee { option-deprecated } { \l_keys_key_str } { indentfirst } } ,
4680   normalindentfirst .code:n =
4681     { \__xeCJK_warning:nee { option-deprecated } { \l_keys_key_str } { } }
4682 }
4683 \__xeCJK_msg_new:nn { option-deprecated }
4684 {
4685   The~`#1'~option~is~deprecated.\\
4686   \tl_if_empty:nF {#2}
4687   { You~may~load~the~package~`#2'~after~xeCJK~to~use~its~function.\\ }
4688 }

```

quiet 将调用 xeCJK 时使用的未知的选项传递给 fontspec 宏包。对 fontspec 的 quiet 和 silent
silent 选项进行修改,使其适用于 xeCJK。

```

4689 \keys_define:nn { xeCJK / options }
4690 {
4691   quiet .code:n =
4692     {
4693       \msg_redirect_module:nnn { xeCJK } { warning } { info }
4694       \msg_redirect_module:nnn { xeCJK } { info } { none }
4695       \xeCJK_if_package_loaded:nF { fontspec }
4696       { \PassOptionsToPackage { quiet } { fontspec } }
4697     } ,
4698   silent .code:n =
4699     {
4700       \msg_redirect_module:nnn { xeCJK } { warning } { none }
4701       \msg_redirect_module:nnn { xeCJK } { info } { none }
4702       \xeCJK_if_package_loaded:nF { fontspec }
4703       { \PassOptionsToPackage { silent } { fontspec } }
4704     } ,
4705   unknown .code:n =
4706     {
4707       \xeCJK_if_package_loaded:nTF { fontspec }
4708       { \__xeCJK_error:ne { key-unknown } { \l_keys_key_str } }
4709       { \PassOptionsToPackage { \l_keys_key_str } { fontspec } }
4710     }
4711 }
4712 \__xeCJK_msg_new:nn { key-unknown }
4713 {
4714   Sorry,~but~xeCJK/options~does~not~have~a~key~called~`#1'.\\
4715   The~key~`#1'~is~being~ignored.
4716 }

```

5.17 xeCJK 初始化设置

```

\CJKsymbol 4717 \cs_new_eq:NN \CJKsymbol \use:n
\CJKpunctsymbol 4718 \cs_new_eq:NN \CJKpunctsymbol \use:n

```

xeCJK 宏包的初始化设置。

```

4719 \keys_set:nn { xeCJK / options }
4720 {
4721   CJKglue      = { \skip_horizontal:n { \c_zero_dim plus 0.08 \tex_baselineskip:D } } ,
4722   CJKecglue    = { ~ } ,
4723   xCJKecglue   = false ,
4724   CheckSingle  = false ,
4725   PlainEquation = false ,
4726   CheckFullRight = false ,
4727   CJKspace     = false ,
4728   CJKmath      = false ,
4729   xeCJKactive  = true ,

```

```

4730 LocalConfig      = true  ,
4731 LoadFandol      = true  ,
4732 RubberPunctSkip = true  ,
4733 Verb            = env   ,
4734 EmboldenFactor  = 4     ,
4735 SlantFactor     = 0.167 ,
4736 PunctStyle      = quanjiao ,
4737 NewLineCS      = { \par \[ ] ,
4738 EnvCS          = { \begin \end } ,
4739 WidowPenalty   = { 10 000 } ,
4740 NoBreakCS      = { \footnote \footnotemark \nobreak } ,
4741 KaiMingPunct   = { 3002 ff0e ff1f ff01 } ,
4742 LongPunct      = { 2014 2e3a 2025 2026 } ,
4743 NoBreakLongPunct = { 2025 2026 } ,
4744 MiddlePunct    = { 2013 2014 2e3a 2027 00b7 30fb ff65 } ,
4745 AllowBreakBetweenPuncts = false
4746 }
4747 \defaultCJKfontfeatures { Script = CJK }

```

半字线连接号¹⁶应为半角宽度。

```
4748 \xeCJKsetwidth { 2013 } { 0.5 em }
```

执行宏包选项,并载入 fontspec 宏包。

```

4749 \cs_if_exist:NTF \ProcessKeyOptions
4750 { \ProcessKeyOptions [ xeCJK / options ] }
4751 {
4752   \RequirePackage { l3keys2e }
4753   \ProcessKeysOptions { xeCJK / options }
4754 }
4755 \RequirePackage { fontspec } [ 2020/02/03 ]

```

`\c__xeCJK_encoding_tl` 保存 fontspec 声明字体时使用的字体编码。

```
4756 \tl_const:Ne \c__xeCJK_encoding_tl { \g_fontspec_encoding_tl }
```

对不能通过 `\xeCJKsetup` 设置的选项给出警告。

```

4757 \keys_define:nn { xeCJK / options }
4758 {
4759   LocalConfig .code:n =
4760     { \__xeCJK_warning:ne { option-invalid } { \l_keys_key_str } }
4761 }
4762 \__xeCJK_msg_new:nn { option-invalid }
4763 {
4764   The~'#1'~option~can~only~be~set~in~the~optional~argument~to~the\\
4765   \token_to_str:N \usepackage \ command~when~xeCJK~is~being~loaded.\\
4766   Please~do~not~set~it~via~the~\token_to_str:N \xeCJKsetup \ command.
4767 }

```

```

\CJKrmdefault 4768 \tl_if_exist:NF \CJKrmdefault
\CJKsfdefault 4769 { \tl_new:N \CJKrmdefault \tl_gset:Nn \CJKrmdefault { rm } }
\CJKttdefault 4770 \tl_if_exist:NF \CJKsfdefault
\CJKfamilydefault 4771 { \tl_new:N \CJKsfdefault \tl_gset:Nn \CJKsfdefault { sf } }
4772 \tl_if_exist:NF \CJKttdefault
4773 { \tl_new:N \CJKttdefault \tl_gset:Nn \CJKttdefault { tt } }
4774 \tl_new:N \l__xeCJK_family_default_init_tl
4775 \cs_new_eq:NN \__xeCJK_family_default_wrap:n \use:n
4776 \tl_set:Ne \l__xeCJK_family_default_init_tl
4777 {
4778   \exp_not:N \__xeCJK_family_default_wrap:n
4779   {
4780     \tl_if_exist:NTF \CJKfamilydefault
4781     { \exp_not:o \CJKfamilydefault }
4782     { \exp_not:N \CJKrmdefault }
4783   }
4784 }
4785 \tl_if_exist:NF \CJKfamilydefault

```

¹⁶见《夹用英文的中文文本的标点符号用法(草案)》5.13节。

```
4786 { \tl_new:N \CJKfamilydefault }
4787 \tl_gset_eq:NN \CJKfamilydefault \l__xeCJK_family_default_init_tl
```

`\xeCJKsetup` 在导言区或文档中设置 xeCJK 的接口。

```
4788 \NewDocumentCommand \xeCJKsetup { +m }
4789 {
4790   \keys_set:nn { xeCJK / options } {#1}
4791   \tex_ignorespaces:D
4792 }
```

```
\xeCJKsetemboldenfactor 4793 \NewDocumentCommand \xeCJKsetemboldenfactor { m }
\xeCJKsetslantfactor 4794 { \xeCJKsetup { EmboldenFactor = {#1} } }
4795 \NewDocumentCommand \xeCJKsetslantfactor { m }
4796 { \xeCJKsetup { SlantFactor = {#1} } }

\punctstyle 4797 \NewDocumentCommand \punctstyle { m } { \xeCJKsetup { PunctStyle = {#1} } }
\xeCJKplainchr 4798 \NewDocumentCommand \xeCJKplainchr { } { \xeCJKsetup { PunctStyle = plain } }

\CJKsetecglue 4799 \NewDocumentCommand \CJKsetecglue { m } { \xeCJKsetup { CJKecglue = {#1} } }
4800 \cs_new_eq:NN \xeCJKsetecglue \CJKsetecglue

\CJKspace 4801 \NewDocumentCommand \CJKspace { } { \xeCJKsetup { CJKspace = true } }
\CJKnospace 4802 \NewDocumentCommand \CJKnospace { } { \xeCJKsetup { CJKspace = false } }

\xeCJKallowbreakbetweenpuncts 4803 \NewDocumentCommand \xeCJKallowbreakbetweenpuncts { }
\xeCJKnobreakbetweenpuncts 4804 { \xeCJKsetup { AllowBreakBetweenPuncts = true } }
4805 \NewDocumentCommand \xeCJKnobreakbetweenpuncts { }
4806 { \xeCJKsetup { AllowBreakBetweenPuncts = false } }

\xeCJKenablefallback 4807 \NewDocumentCommand \xeCJKenablefallback { }
\xeCJKdisablefallback 4808 { \xeCJKsetup { AutoFallBack = true } }
4809 \NewDocumentCommand \xeCJKdisablefallback { }
4810 { \xeCJKsetup { AutoFallBack = false } }

\xeCJKsetcharclass 4811 \__xeCJK_msg_new:nn { xeCJKsetcharclass-deprecated }
4812 {
4813   \token_to_str:N \xeCJKsetcharclass\ is~deprecated~and~
4814   does~not~update~CJKmath~char~lists.\\
4815   Use~\token_to_str:N \xeCJKDeclareCharClass\ instead.
4816 }
4817 \NewDocumentCommand \xeCJKsetcharclass { m m m }
4818 {
4819   \__xeCJK_error:n { xeCJKsetcharclass-deprecated }
4820   \xeCJK_set_char_class:nnn {#1} {#2} {#3}
4821   \xeCJKResetPunctClass
4822 }
```

5.18 兼容性修补

`\xeCJK@update@fam` 通过 `\urlstyle` 或者 `\UrlFont` 设置的路径中使用的 CJK 字体生效。使用 `\everymath` 钩子中数学模式中重定义 CJK 数学字体,以确保我们的设置在 `\check@mathfonts` 之后生效,不会被它覆盖。更合理的方式是定义一个新的 `\mathversion` 来切换。

```
4823 \cs_new_protected:Npn \xeCJK@update@fam
4824 {
4825   \addto@hook \everymath
4826   {
4827     \__xeCJK_update_main_fam:
4828     \__xeCJK_update_block_fam:
4829   }
4830 }
4831 \cs_new_protected:Npn \__xeCJK_update_main_fam:
4832 {
4833   \group_begin:
4834   \xeCJK_select_font:
4835   \exp_last_unbraced:NNNo \group_end:
4836   \tex_textfont:D \c_xeCJK_math_fam_int \tex_the:D \tex_font:D
```

```

4837 }
4838 \cs_new_protected:Npn \__xeCJK_update_block_fam:
4839 {
4840   \prop_if_empty:NF \g__xeCJK_block_fam_prop
4841   {
4842     \prop_map_function:NN
4843     \g__xeCJK_block_fam_prop
4844     \__xeCJK_update_block_fam:nn
4845   }
4846 }
4847 \cs_new_protected:Npn \__xeCJK_update_block_fam:nn #1#2
4848 {
4849   \int_set:Nn \l__xeCJK_fam_int {#2}
4850   \group_begin:
4851     \xeCJK_select_font:n {#1}
4852     \exp_last_unbraced:NNNo \group_end:
4853     \tex_textfont:D \l__xeCJK_fam_int \tex_the:D \tex_font:D
4854 }
4855 \__xeCJK_after_end_preamble:n
4856 {
4857   \bool_lazy_and:nnT
4858   { \g__xeCJK_math_bool }
4859   { \cs_if_exist_p:N \Url@MathSetup }
4860   { \tl_put_right:Nn \Url@MathSetup { \xeCJK@update@fam } }
4861 }

```

`\(` 的在 $\text{\LaTeX}_2\epsilon$ 中的定义是

```
\def\({\relax\ifmmode\@badmath\else$\fi}
```

`\math`

`\endmath`

`\ensuremath`

`__xeCJK_math_robust:N`

这个定义最开始的 `\relax` 是为了防止 `\(` 出现在表格单元格的开始位置时, 模式判断不正确 (因为 \TeX 会先看单元格中第一个不可展的非空格记号是否是 `\omit` 或 `\noalign`)。但是它会创建一个边界, 使 `xeCJK` 不能看到 `\relax` 后面出现的 `$`, 从而不能加入间距¹⁷。使用 $\epsilon\text{-TeX}$ 的 `\protected` 来定义它, 可以不需要 `\relax`, 或者将 `\relax` 改成 `\scan_align_safe_stop:`, 都可以避免这些情况。同时 `fixltx2e` 中还使用了 `\MakeRobust\()`, 我们需要小心处理。另外 `ulem` 也定义了一个 `\MakeRobust`, 如果它被放在 `fixltx2e` 之前载入, 那么 `fixltx2e` 的定义就会失效 (因为 `fixltx2e` 使用 `\providecommand*` 来定义 `\MakeRobust`)。但是 `ulem` 的定义并不完全正确, 没有考虑 \TeX 不会略去控制符号后面的空格的情况。

```

4862 \cs_new_protected:Npn \__xeCJK_math_robust:N #1
4863 {
4864   \group_begin: \exp_args:NcNc \group_end:
4865     { \__xeCJK_math_robust_aux:NN } #1 { \cs_to_str:N #1 ~ }
4866 }
4867 \cs_new_protected:Npn \__xeCJK_math_robust_aux:NN #1#2
4868 {
4869   \exp_args:Ne \str_case:nnTF { \cs_replacement_spec:N #1 }
4870   {
4871     { \x@protect #1 \protect #2 } { }
4872     { \protect #2 } { }
4873   }
4874   { \__xeCJK_math_robust:NN #1#2 }
4875   { \__xeCJK_math_robust:NN #1#1 }
4876 }
4877 \cs_new_protected:Npn \__xeCJK_math_robust:NN #1#2
4878 {
4879   \str_if_eq:eeTF { \cs_argument_spec:N #2 } { }
4880   {
4881     \exp_args:No \tl_if_head_eq_meaning:nNTF {#2} \scan_stop:
4882     {
4883       \cs_gset_protected:Npe #1
4884       { \tl_tail:N #2 }
4885     }

```

¹⁷<http://tex.stackexchange.com/q/124773>

```

4886     {
4887         \cs_if_eq:NNTF #1 \ensuremath
4888         {
4889             \cs_gset_protected:Npe #1
4890             { \exp_not:o {#2} }
4891         }
4892         {
4893             \__xeCJK_warning:nee { robust-failure }
4894             { \token_to_str:N #1 } { \token_to_meaning:N #2 }
4895         }
4896     }
4897 }
4898 {
4899     \__xeCJK_warning:nee { robust-failure }
4900     { \token_to_str:N #1 } { \token_to_meaning:N #2 }
4901 }
4902 }
4903 \__xeCJK_msg_new:nnn { robust-failure }
4904 { xeCJK~can~not~make~`#1'~robust. }
4905 {
4906     The~current~meaning~of~`#1'~is:\\
4907     \iow_indent:n {#2}
4908 }
4909 \cs_if_eq:NNTF \(\ \math
4910 {
4911     \__xeCJK_math_robust:N \(\
4912     \cs_set_eq:NN \math \(\
4913 }
4914 {
4915     \__xeCJK_math_robust:N \(\
4916     \__xeCJK_math_robust:N \math
4917 }
4918 \cs_if_eq:NNTF \) \endmath
4919 {
4920     \__xeCJK_math_robust:N \)
4921     \cs_set_eq:NN \endmath \)
4922 }
4923 {
4924     \__xeCJK_math_robust:N \)
4925     \__xeCJK_math_robust:N \endmath
4926 }
4927 \__xeCJK_math_robust:N \ensuremath

```

`\fontfamily` 使用 L^AT_EX 2_ε 2020/10/01 提供的 NFSS 钩子, 使主要 CJK 字体族能随西文主要字体更新。
`\xeCJK@family`

```

4928 \cs_set_eq:NN \xeCJK@family \xeCJK_switch_family:e
4929 \ctex_gadd_ltxhook:nn { rmfamily } { \xeCJK@family { \CJKrmdefault } }
4930 \ctex_gadd_ltxhook:nn { sffamily } { \xeCJK@family { \CJKsfdefault } }
4931 \ctex_gadd_ltxhook:nn { ttfamily } { \xeCJK@family { \CJKttdefault } }
4932 \ctex_gadd_ltxhook:nn { normalfont } { \xeCJK@family { \CJKfamilydefault } }
4933 <@=@>

```

`\xeCJK@fix@penalty` 对 L^AT_EX 2_ε 内核中的 `\fix@penalty` 被用于诸如 `\textit` 之类的文档字体转换命令的定义之中。这里对它进行补丁的目的是修复其中的倾斜校正, 并使得这些文档命令与紧随其后的汉字之间可以正确的插入 `\CJKecglue` 或者忽略其中的空格。例如 这是_␣`\emph{强调}`_␣文本, 第二个空格可以被忽略掉。如果使用 `xCJKecglue` 选项, 第一个空格也可以被省略。事实上, 在 `\sw@slant` 的定义中, `\@@italiccorr` 前面的 `\lastskip` 和 `\lastpenalty` 有四种情况, 这里只对它们都为零的情况进行处理。

```

4934 \cs_new_eq:NN \xeCJK@fix@penalty \fix@penalty
4935 \tl_replace_once:Nnn \xeCJK@fix@penalty { \@@italiccorr } { \xeCJK@italiccorr }
4936 \tl_replace_once:Nnn \sw@slant { \fix@penalty } { \xeCJK@fix@penalty }

```

`\xeCJK@italiccorr` 修复倾斜校正, 并处理汉字后面的空格。

```

4937 \cs_new_protected:Npn \xeCJK@italiccorr
4938   {
4939     \int_compare:nNnTF \tex_XeTeXinterchartokenstate:D > \c_zero_int
4940       { \xeCJK_italic_correction: }
4941       { \@@italiccorr }
4942   }
4943 <@@=xeCJK>

```

`\xeCJK_italic_correction:` 修复倾斜校正, 并处理汉字后面的空格。

```

4944 \cs_new_protected:Npn \xeCJK_italic_correction:
4945   { \_xeCJK_if_last_kern:T { \_xeCJK_italic_correction: } }
4946 \cs_new_protected:Npn \_xeCJK_italic_correction:
4947   {
4948     \dim_case:nnF { \tex_lastkern:D }
4949     {
4950       { \_xeCJK_node:n { default } }
4951       {
4952         \xeCJK_remove_node: \tex_italiccorrection:D
4953         \xeCJK_make_node:n { default }
4954       }
4955       { \_xeCJK_node:n { CJK } }
4956       {
4957         \xeCJK_remove_node: \tex_italiccorrection:D
4958         \xeCJK_make_node:n { CJK }
4959         \_xeCJK_italic_correction_aux:
4960       }
4961       { \_xeCJK_node:n { CJK-space } }
4962       {
4963         \xeCJK_remove_node: \tex_italiccorrection:D
4964         \xeCJK_make_node:n { CJK-space }
4965         \_xeCJK_italic_correction_aux:
4966       }
4967     }
4968     { \tex_italiccorrection:D }
4969   }

```

`\xeCJK_ignore_spaces:w` 里面用到 `peek` 函数来判断后面是不是空格, 而此时它后面还有 4 个 `\fi` 或者 `\else... \fi` 没有被展开, 将影响 `peek` 函数的判断。因此我们需要用 $2^4 - 1 = 15$ 个 `\exp_after:wN` 来展开它们。显然, 这里用 `\exp_last_unbraced:Nf` 会比较方便, 但是它会吃掉 `\textit{...}` 等后面原来存在的空格作为完全展开的结束。要正确使用它还需要另外的处理(使用 `\exp_stop_f:`)。

```

4970 \cs_new_protected:Npn \_xeCJK_italic_correction_aux:
4971   {
4972     \exp_after:wN \exp_after:wN \exp_after:wN
4973     \exp_after:wN \exp_after:wN \exp_after:wN \exp_after:wN
4974     \exp_after:wN \exp_after:wN \exp_after:wN \exp_after:wN
4975     \exp_after:wN \exp_after:wN \exp_after:wN \exp_after:wN
4976     \xeCJK_ignore_spaces:w
4977   }

```

`\g_xeCJK_xetex_allocator_int` L^AT_EX 2_ε 2015/01/01 接管了 `\newXeTeXintercharclass`。

```

4978 \cs_new_eq:NN \g_xeCJK_xetex_allocator_int \xe@alloc@intercharclass

```

`_xeCJK_set_others_toks:n` 简单处理与同样使用 `\XeTeXinterchartoks` 机制的宏包的兼容问题。

```

4979 \_xeCJK_after_end_preamble:n
4980   {
4981     \int_compare:nNnF
4982       { \c__xeCJK_class_begin_int + \seq_count:N \g__xeCJK_new_class_seq } =
4983       { \g__xeCJK_xetex_allocator_int }
4984     {
4985       \int_step_inline:nnn

```

```

4986     { \c__xeCJK_class_begin_int + 1 }
4987     { \g__xeCJK_xetex_allocator_int }
4988     {
4989         \seq_if_in:NnF \g__xeCJK_new_class_seq {#1}
4990         { \__xeCJK_set_others_toks:n {#1} }
4991     }
4992 }
4993 }
4994 \cs_new_protected:Npn \__xeCJK_set_others_toks:n #1
4995 {
4996     \int_set:cn { \__xeCJK_class_csname:n { Others } } {#1}
4997     \seq_map_inline:Nn \g__xeCJK_CJK_class_seq
4998     {
4999         \xeCJK_copy_inter_class_toks:nxxx {##1} { Others } {##1} { NormalSpace }
5000         \xeCJK_copy_inter_class_toks:nxxx { Others } {##1} { NormalSpace } {##1}
5001         \xeCJK_app_inter_class_toks:nne {##1} { Others }
5002         { \xeCJK_get_inter_class_toks:nn { Default } { Others } }
5003         \xeCJK_pre_inter_class_toks:nne { Others } {##1}
5004         { \xeCJK_get_inter_class_toks:nn { Others } { Default } }
5005         \tl_if_blank:eT
5006         { \xeCJK_get_inter_class_toks:nn { Others } { Boundary } }
5007         {
5008             \xeCJK_copy_inter_class_toks:nxxx
5009             { Others } { Boundary } { Default } { Boundary }
5010         }
5011         \tl_if_blank:eT
5012         { \xeCJK_get_inter_class_toks:nn { Boundary } { Others } }
5013         {
5014             \xeCJK_copy_inter_class_toks:nxxx
5015             { Boundary } { Others } { Boundary } { Default }
5016         }
5017     }
5018 }

```

`__xeCJK_inactive_group_begin:` 用于保护下面歧义宽度标点的分组。

`__xeCJK_inactive_group_end:`

```

5019 \cs_new_protected:Npn \__xeCJK_inactive_group_begin:
5020 { \group_begin: \makexeCJKinactive }
5021 \cs_new_eq:NN \__xeCJK_inactive_group_end: \group_end:

```

`__xeCJK_patch_text_command:`

`\c__xeCJK_ambiguous_char_prop`

单独处理宽度有分歧的几个标点: 包括省略号、破折号、间隔号、引号等中西文混用的符号, 保证其命令形式输出的是西文字体。如果 `xunicode` 宏包被载入, 则通过 `xunicode-addon` 处理。

```

5022 \prop_const_from_keyval:Nn \c__xeCJK_ambiguous_char_prop
5023 {
5024     "00B7 = \textperiodcentered \textcentereddot \textdot ,
5025     "2013 = \textendash ,
5026     "2014 = \textemdash ,
5027     "2018 = \textquoteleft \textgrq ,
5028     "2019 = \textquoteright ,
5029     "201C = \textquotedblleft \textgrqq ,
5030     "201D = \textquotedblright ,
5031     "2025 = \texthdofor ,
5032     "2026 = \textellipsis ,
5033     "2027 = \texthyphenationpoint ,
5034     "2E3A = \texttwoemdash
5035 }
5036 \__xeCJK_at_end_preamble:n { \__xeCJK_patch_text_command: }
5037 \cs_new_protected:Npn \__xeCJK_patch_text_command:
5038 {
5039     \xeCJK_if_package_loaded:nTF { xunicode }
5040     { \__xeCJK_patch_xunicode_ambiguous_char: }
5041     {
5042         \exp_args:Ne \__xeCJK_patch_tuenc_ambiguous_char:n
5043         { \UnicodeEncodingName }
5044         \__xeCJK_patch_tuenc_accent:
5045         \__xeCJK_patch_tuenc_composite:
5046     }

```

```

5047 }
5048 \cs_new_protected:Npn \__xeCJK_patch_xunicode_ambiguous_char:
5049 {
5050   \RequirePackage { xunicode-addon }
5051   \prop_map_inline:Nn \c__xeCJK_ambiguous_char_prop
5052     {
5053       \tl_map_inline:nn { ##2 }
5054         {
5055           \xunadd_set_begin_hook:nn { #####1 }
5056             { \__xeCJK_inactive_group_begin: }
5057           \xunadd_set_end_hook:nn { #####1 }
5058             { \__xeCJK_inactive_group_end: }
5059         }
5060     }
5061   \xunadd_append_begin_hook:n { \xeCJK_make_boundary: }
5062 }
5063 \cs_new_protected:Npn \__xeCJK_patch_tuenc_ambiguous_char:n #1
5064 {
5065   \prop_map_inline:Nn \c__xeCJK_ambiguous_char_prop
5066     {
5067       \tl_map_inline:nn { ##2 }
5068         {
5069           \cs_if_exist:NF #####1
5070             { \DeclareTextSymbol #####1 {#1} { ##1 } }
5071           \__xeCJK_patch_ambiguous_char:nN {#1} #####1
5072         }
5073     }
5074 }
5075 \cs_new_protected:Npn \__xeCJK_patch_ambiguous_char:nN #1#2
5076 {
5077   \exp_args:Ne \__xeCJK_patch_ambiguous_char:nn
5078     { #1 \token_to_str:N #2 }
5079     { #1 - #2 }
5080 }
5081 \cs_new_protected:Npe \__xeCJK_patch_ambiguous_char:nNn #1#2#3
5082 {
5083   \exp_not:N \exp_args:Ne
5084   \exp_not:N \__xeCJK_patch_ambiguous_char:nn
5085     {
5086       \c_backslash_str #1
5087       \exp_not:N \token_to_str:N #2 -
5088       \exp_not:N \token_to_str:N #3
5089     }
5090     { #1 - #2#3 }
5091 }
5092 \cs_new_protected:Npn \__xeCJK_patch_ambiguous_char:nn #1#2
5093 {
5094   \cs_if_free:cF {#1}
5095     { \exp_args:Nc \__xeCJK_patch_ambiguous_char:Nn {#1} {#2} }
5096 }
5097 \cs_new_protected:Npn \__xeCJK_patch_ambiguous_char:Nn #1#2
5098 {
5099   \token_if_chardef:NTF #1
5100     {
5101       \prop_gput:Nne \g__xeCJK_ambiguous_slot_prop {#2}
5102         { \int_eval:n {#1} }
5103       \cs_set_protected:Npe #1
5104         { \__xeCJK_ambiguous_char:n { \tex_Uchar:D #1 } }
5105     }
5106     {
5107       \prop_gput:Nne \g__xeCJK_ambiguous_slot_prop {#2}
5108         { \int_eval:n { \exp_after:wN ` #1 } }
5109       \cs_set_protected:Npe #1
5110         { \__xeCJK_ambiguous_char:n { \exp_not:o {#1} } }
5111     }
5112 }
5113 \cs_new_protected:Npn \__xeCJK_ambiguous_char:n #1

```

```

5114 {
5115   \int_compare:nNnTF \tex_XeTeXinterchartokenstate:D > \c_zero_int
5116     { \__xeCJK_inactive_group_begin: #1 \__xeCJK_inactive_group_end: }
5117     {#1}
5118 }
5119 \prop_new:N \g__xeCJK_ambiguous_slot_prop

```

`__xeCJK_patch_tuenc_composite:` `\DeclareUnicodeComposite` 具有检查字符是否存在的功能, 当符号命令紧跟在 CJK 字符类后面时, 需要使字体回到西文状态

```

5120 \cs_new_protected:Npn \xeCJK_text_composite_patch:
5121 {
5122   \str_if_eq:eeT { \f@encoding } { \UnicodeEncodingName }
5123     { \xeCJK_make_boundary: }
5124 }

```

注意 `\xeCJK_text_composite_patch:` 可能会结束分组, 从而导致 `##1` 没有定义时是 `\undefined` 而不是 `\relax`, 所以不能将它与 `\relax` 作比较。

```

5125 \cs_new_protected:Npn \__xeCJK_patch_tuenc_composite:
5126 {
5127   \cs_set_nopar:Npn \@text@composite@x
5128     {
5129       \xeCJK_text_composite_patch:
5130       \cs_if_exist_use:NF
5131     }
5132 }

```

`__xeCJK_patch_tuenc_accent:` `\add@unicode@accent` 定义最后用于截断数字展开的 `\relax` 会造成边界, 可能会影响组合标记。

```

5133 \group_begin:
5134 \char_set_catcode_other:n { "A0 }
5135 \cs_new_protected:Npn \__xeCJK_patch_tuenc_accent:
5136 {
5137   \cs_set_protected_nopar:Npn \add@unicode@accent ##1 ##2
5138     {
5139       \tl_if_blank:nTF { ##2 } { ^~a0 } { ##2 }
5140       \tex_Uchar:D \tex_numexpr:D ##1 \scan_stop:
5141     }
5142 }
5143 \group_end:

```

`__xeCJK_patch_middle_dot:` 常被用作中文间隔号的 U+00B7 与 T1 等旧字体编码下定义的符号命令冲突。在 `encguide.pdf` 的编码符号表中, 如下定义有冲突。

```

\DeclareTextComposite{\r}{T1}{u}{183}
\DeclareTextSymbol{\cyrchvcrs}{T2A}{183}
\DeclareTextSymbol{\cyrchldsc}{T2B}{183}
\DeclareTextSymbol{\cyrabhha}{T2C}{183}
\DeclareTextSymbol\textvibyy{T3}{183}
\DeclareTextComposite{\B}{T4}{t}{183}
\DeclareTextComposite{\`}{T5}{\ecircumflex}{183}
\DeclareTextDoubleComposite{\`}{T5}{\`}{e}{183}
\DeclareTextSymbol{\textperiodcentered}{TS1}{183}
\DeclareTextSymbol{\cyrchldsc}{X2}{183}
\DeclareTextSymbol{\textperiodcentered}{LY1}{183}

```

LGR 编码的符号表有 183 号字符, 但在 `lgrenc.def` 中未找到相应的符号命令。

```

5144 \prop_const_from_keyval:Nn \c__xeCJK_middle_dot_prop
5145 {
5146   T2A = \cyrchvcrs ,
5147   T2B = \cyrchldsc ,
5148   T2C = \cyrabhha ,
5149   X2  = \cyrchldsc ,
5150   TS1 = \textperiodcentered ,
5151   LY1 = \textperiodcentered ,

```

```

5152 T1 = \r u ,
5153 T4 = \B t ,
5154 T5 = \` \ecircumflex
5155 }
5156 \__xeCJK_at_end_preamble:n { \__xeCJK_patch_middle_dot: }
5157 \cs_new_protected:Npn \__xeCJK_patch_middle_dot:
5158 {
5159   \prop_map_inline:Nn \c__xeCJK_middle_dot_prop
5160     { \__xeCJK_patch_middle_dot:nw { ##1 } ##2 \q_stop }
5161   \__xeCJK_patch_ambiguous_char:nNn { T5 } \` { \^ - e }
5162 }
5163 \cs_new_protected:Npn \__xeCJK_patch_middle_dot:nw #1#2#3 \q_stop
5164 {
5165   \tl_if_empty:nTF {#3}
5166     { \__xeCJK_patch_ambiguous_char:nN {#1} #2 }
5167     { \__xeCJK_patch_ambiguous_char:nNn {#1} #2 {#3} }
5168 }

```

pdfont 宏包的符号 `\ding{183}` 也有冲突。

```

5169 \__xeCJK_package_hook:nn { pdfont }
5170 {
5171   \RenewDocumentCommand \Pifont { m }
5172     { \mode_leave_vertical: \makexeCJKinactive \usefont { U } {#1} { m } { n } }
5173 }

```

`\xeCJKchar` `\char` 是 TeX 原语, 用户使用它直接通过字符编码从当前字体取字形, 不经过 NFSS 层。一些宏包(如 `mtpro2`)在数学模式中用 `\char` 取非 CJK 字形时, 会被 xeCJK 的 `interchar` 机制拦截并切换到中文字体, 导致字形变成中文标点(如间隔号, #407)。

参照 LuaTeX-ja 提供 `\ltjalchar/\ltjjachar` 新命令的思路, xeCJK 提供 `\xeCJKchar` 命令, 在输出字符前临时关闭 `XeTeXinterchartokenstate`, 保证输出的字符来自当前字体而不被 `interchar` 拦截。与 LuaTeX-ja 不同的是, XeTeX 的 `interchar` 机制工作在 token 层面, 没有节点级回调, 因此无法自动区分 `\char` 和直接字符输入。`\xeCJKchar` 供用户和第三方宏包显式使用, `\char` 本身不做任何修改。

用法与 `\char` 完全相同: `\xeCJKchar"00B7` 或 `\xeCJKchar183`。用户指南和手动补丁方法参见“已知问题和兼容性”一节。

```

5174 \cs_new_protected:Npn \xeCJK_default_char:w
5175 {
5176   \int_compare:nNnTF \tex_XeTeXinterchartokenstate:D > \c_zero_int
5177     {
5178       \c_group_begin_token
5179       \tex_afterassignment:D \__xeCJK_default_char_aux:
5180       \l__xeCJK_default_char_int =
5181     }
5182     { \tex_char:D }
5183 }
5184 \cs_new_protected:Npn \__xeCJK_default_char_aux:
5185 {
5186   \tex_XeTeXinterchartokenstate:D = \c_zero_int
5187   \tex_char:D \l__xeCJK_default_char_int
5188   \c_group_end_token
5189 }
5190 \int_new:N \l__xeCJK_default_char_int
5191 \cs_new_eq:NN \xeCJKchar \xeCJK_default_char:w

```

`mtpro2` 的 `\overbrace` 和 `\underbrace` 通过 `\char` 访问 `mt2exe` 等字体的特殊位置, 其中部分位置(如 `183 = U+00B7`)与 xeCJK 的 `interchar` 字符分类冲突。由于这两个命令只在数学模式中, 参数不含 CJK 字符, 可以在入口处直接关闭 `XeTeXinterchartokenstate`, 由数学模式的分组自动恢复。

```

5192 \__xeCJK_package_hook:nn { mtpro2 }
5193 {

```

```

5194 \cs_set_eq:NN \__xeCJK_mtpro_overcbrace:w \overcbrace
5195 \cs_gset_protected:Npn \overcbrace #1
5196 {
5197   \makexeCJKinactive
5198   \__xeCJK_mtpro_overcbrace:w {#1}
5199 }
5200 \cs_set_eq:NN \__xeCJK_mtpro_undercbrace:w \undercbrace
5201 \cs_gset_protected:Npn \undercbrace #1
5202 {
5203   \makexeCJKinactive
5204   \__xeCJK_mtpro_undercbrace:w {#1}
5205 }
5206 }

```

`__xeCJK_save_um_char:` 兼容 `unicode-math` 和 `CJKmath` 选项, 避免将一些中西文混用的标点设置为 CJK 字体。

```

\__xeCJK_save_um_char:
\__xeCJK_save_um_char:
5207 \__xeCJK_package_hook:nn { unicode-math }
5208 {
5209   \prop_const_from_keyval:Nn \c__xeCJK_um_ambiguous_char_prop
5210   {
5211     "00B7 = \cdotp ,
5212     "2025 = \enleadertwodots ,
5213     "2026 = \unicodeellipsis
5214   }
5215   \cs_new_protected:Npn \__xeCJK_save_um_char:
5216   {
5217     \cs_set_protected:Npe \__xeCJK_restore_um_char:
5218     {
5219       \prop_map_function:NN
5220       \c__xeCJK_um_ambiguous_char_prop
5221       \__xeCJK_restore_um_char_aux:nn
5222     }
5223   }
5224   \cs_new_eq:NN \__xeCJK_restore_um_char: \prg_do_nothing:
5225   \cs_new:Npn \__xeCJK_restore_um_char_aux:nn #1#2
5226   {
5227     \__xeCJK_gset_mathcodenum:nn
5228     { \int_value:w #1 }
5229     { \int_value:w \tex_Umathcodenum:D #1 }
5230   }
5231   \cs_new_protected:Npn \__xeCJK_gset_mathcodenum:nn #1#2
5232   {
5233     \int_compare:nNnF { \tex_Umathcodenum:D #1 } = {#2}
5234     { \tex_global:D \tex_Umathcodenum:D #1 = #2 ~ }
5235   }
5236 }

```

`__xeCJK_patch_microtype_get_slot:` 兼容 `microtype`。

```

5237 \cs_new_protected:Npn \__xeCJK_patch_microtype_get_slot:
5238 {
5239   \cs_new_eq:NN \xeCJK@original@get@slot \MT@get@slot@
5240   \cs_set_eq:NN \MT@get@slot@ \xeCJK@microtype@get@slot
5241   \cs_set_eq:NN \MT@warn@unknown@once \use_none:n
5242 }
5243 \cs_new_protected_nopar:Npn \xeCJK@microtype@get@slot
5244 {
5245   \int_compare:nNnT \MT@char < \c_zero_int
5246   { \__xeCJK_get_ambiguous_slot: }
5247   \xeCJK@original@get@slot
5248 }
5249 \cs_new_protected:Npn \__xeCJK_get_ambiguous_slot:
5250 {
5251   \prop_get:NeNT \g__xeCJK_ambiguous_slot_prop
5252   { \MT@encoding - \tex_the:D \MT@toks } \l__xeCJK_tmp_tl
5253   { \cs_set_eq:NN \MT@char \l__xeCJK_tmp_tl }
5254 }

```

```

5255 \cs_new_protected:Npn \xeCJK@microtype@restore@pickupfont
5256 { \__xeCJK_gadd_font_initial_hook:n { \MT@ltx@pickupfont } }
5257 \__xeCJK_package_hook:nn { microtype }
5258 {
5259   \cs_if_free:NF \MT@get@slot@
5260   { \__xeCJK_patch_microtype_get_slot: }
5261   \MT@addto@setup { \xeCJK@microtype@restore@pickupfont }
5262 }

```

简单处理与 `hyperref` 宏包的兼容问题。

```

5263 \tl_new:N \g__xeCJK_hyper_saved_node_tl
5264 \cs_new_protected:Npn \__xeCJK_patch_hyperref_annot:
5265 {
5266   \cs_gset_eq:NN \__xeCJK_orig_Hy_BeginAnnot:n \Hy@BeginAnnot
5267   \cs_gset_protected:Npn \Hy@BeginAnnot ##1
5268   {
5269     \xeCJK_if_last_node:TF
5270     { \tl_gset_eq:NN \g__xeCJK_hyper_saved_node_tl \g__xeCJK_last_node_tl }
5271     { \tl_gclear:N \g__xeCJK_hyper_saved_node_tl }
5272     \tl_gclear:N \g__xeCJK_last_node_tl
5273     \__xeCJK_orig_Hy_BeginAnnot:n { ##1 }
5274     \tl_if_eq:NnT \g__xeCJK_hyper_saved_node_tl { CJK }
5275     { \xeCJK_make_node:n { CJK } }
5276     \tl_if_eq:NnT \g__xeCJK_hyper_saved_node_tl { CJK-space }
5277     { \xeCJK_make_node:n { CJK-space } }
5278     \tl_if_eq:NnT \g__xeCJK_hyper_saved_node_tl { CJK-widow }
5279     { \xeCJK_make_node:n { CJK-widow } }
5280   }
5281 }
5282 \__xeCJK_package_hook:nn { hyperref }
5283 {
5284   \pdfstringdefDisableCommands
5285   {
5286     \__xeCJK_gobble_CJKfamily:
5287     \xeCJK_cs_clear:N \__xeCJK_inactive_group_begin:
5288     \xeCJK_cs_clear:N \__xeCJK_inactive_group_end:
5289     \xeCJK_cs_clear:N \makexeCJKinactive
5290     \xeCJK_cs_clear:N \xeCJK_text_composite_patch:
5291   }
5292   \__xeCJK_patch_hyperref_annot:
5293 }

```

`color/xcolor` 的颜色切换通过 `\set@color` 插入 `\special` 节点(whatsit), 会遮蔽 `xeCJK` 的标记 `kern`, 导致后续的 `CJKecglue` 检测失败。此处可在 `\set@color` 和 `\reset@color` 之后重新放置 `xeCJK` 节点标记。`\set@color` 负责颜色推入(push), `\reset@color` 负责颜色弹出(pop)——后者通过 `\aftergroup` 在分组结束时自动调用, 用于恢复原始颜色。当最后一个节点是 `xeCJK` 标记 `kern` 对时直接重放标记; 当最后一个节点是 `hlist` 且 `\g_@@_last_node_tl` 非空时, 仅设置 `\g_@@_reset_color_pending_bool` 以延迟到 `\@@_check_for_glue_skip:` 的 `whatsit` 回退路径中处理, 避免在页面布局结构中插入多余 `kern`。

```

5294 \cs_new_protected:Npn \__xeCJK_patch_set_color:
5295 {
5296   \cs_if_exist:NF \__xeCJK_orig_set_color:
5297   {
5298     \cs_set_eq:NN \__xeCJK_orig_set_color: \set@color
5299     \cs_gset_protected:Npn \set@color
5300     {
5301       \xeCJK_if_last_node:TF
5302       {
5303         \__xeCJK_orig_set_color:
5304         \tl_if_empty:NF \g__xeCJK_last_node_tl
5305         { \exp_args:No \xeCJK_make_node:n { \g__xeCJK_last_node_tl } }
5306       }
5307       { \tl_gclear:N \g__xeCJK_last_node_tl \__xeCJK_orig_set_color: }
5308     }

```

```

5309     }
5310     \cs_if_exist:NF \__xeCJK_orig_reset_color:
5311     {
5312         \cs_set_eq:NN \__xeCJK_orig_reset_color: \reset@color
5313         \cs_gset_protected:Npn \reset@color
5314         {
5315             \xeCJK_if_last_node:TF
5316             {
5317                 \__xeCJK_orig_reset_color:
5318                 \tl_if_empty:NF \g__xeCJK_last_node_tl
5319                 {
5320                     \exp_args:No \xeCJK_make_node:n { \g__xeCJK_last_node_tl }
5321                     \bool_gset_true:N \g__xeCJK_glue_check_pending_bool
5322                 }
5323             }
5324             {
5325                 \__xeCJK_if_last_hlist:TF
5326                 {
5327                     \tl_if_empty:NF \g__xeCJK_last_node_tl
5328                     { \bool_gset_true:N \g__xeCJK_reset_color_pending_bool }
5329                     \__xeCJK_orig_reset_color:
5330                 }
5331                 { \__xeCJK_orig_reset_color: }
5332             }
5333         }
5334     }
5335 }
5336 \__xeCJK_package_hook:nn { color } { \__xeCJK_patch_set_color: }
5337 \__xeCJK_package_hook:nn { xcolor } { \__xeCJK_patch_set_color: }

```

`\l3color`(`expl3` 内置)的颜色机制使用独立的后端代码路径,不经过 `\set@color/\reset@color`。`__color_select:N` 负责颜色推入(调用后端 `select` 并注册 `aftergroup reset`)，`__color_backend_reset:` 负责颜色弹出。此处对这两个函数施加与上面相同的 `kern` 对保护。

```

5338 \cs_new_protected:Npn \__xeCJK_patch_l3color:
5339 {
5340     \cs_if_exist:NF \__xeCJK_orig_color_select:N
5341     {
5342         \cs_set_eq:NN \__xeCJK_orig_color_select:N \__color_select:N
5343         \cs_gset_protected:Npn \__color_select:N ##1
5344         {
5345             \xeCJK_if_last_node:TF
5346             {
5347                 \__xeCJK_orig_color_select:N ##1
5348                 \tl_if_empty:NF \g__xeCJK_last_node_tl
5349                 { \exp_args:No \xeCJK_make_node:n { \g__xeCJK_last_node_tl } }
5350             }
5351             { \tl_gclear:N \g__xeCJK_last_node_tl \__xeCJK_orig_color_select:N ##1 }
5352         }
5353     }
5354     \cs_if_exist:NF \__xeCJK_orig_color_backend_reset:
5355     {
5356         \cs_set_eq:NN \__xeCJK_orig_color_backend_reset: \__color_backend_reset:
5357         \cs_gset_protected:Npn \__color_backend_reset:
5358         {
5359             \xeCJK_if_last_node:TF
5360             {
5361                 \__xeCJK_orig_color_backend_reset:
5362                 \tl_if_empty:NF \g__xeCJK_last_node_tl
5363                 {
5364                     \exp_args:No \xeCJK_make_node:n { \g__xeCJK_last_node_tl }
5365                     \bool_gset_true:N \g__xeCJK_glue_check_pending_bool
5366                 }
5367             }
5368             {
5369                 \__xeCJK_if_last_hlist:TF
5370                 {

```

```

5371         \tl_if_empty:NF \g__xeCJK_last_node_tl
5372         { \bool_gset_true:N \g__xeCJK_reset_color_pending_bool }
5373         \__xeCJK_orig_color_backend_reset:
5374     }
5375     { \__xeCJK_orig_color_backend_reset: }
5376 }
5377 }
5378 }
5379 }
5380 \__xeCJK_patch_l3color:

```

hypdoc 的 `\HD@target` 通过 `\raisebox` 产生 `hbox(0+0)x0` 用于放置 `\special{pdf:dest...}`。该 `hbox` 隔在 xeCJK 标记 kern 对和后续字符之间, 使 `\tex_lastkern:D` 探测失败, 导致 `Boundary→{Default,CJK}` 路径中 `\xeCJK_if_last_node:n / \@_if_last_kern:` 检查返回 `false`, 间距因此丢失。此处保存 `\g__@_last_node_tl`、清除现有标记 kern、调用原始 `\HD@target`、然后根据保存的状态重放节点标记, 使后续触发路径能正确识别前方上下文。

```

5381 \tl_new:N \g__xeCJK_hd_saved_node_tl
5382 \cs_new_protected:Npn \__xeCJK_patch_hd_target:
5383 {
5384     \cs_if_exist:NF \__xeCJK_orig_HD_target:
5385     {
5386         \cs_set_eq:NN \__xeCJK_orig_HD_target: \HD@target
5387         \cs_gset_protected:Npn \HD@target
5388         {
5389             \xeCJK_if_last_node:TF
5390             { \tl_gset_eq:NN \g__xeCJK_hd_saved_node_tl \g__xeCJK_last_node_tl }
5391             { \tl_gclear:N \g__xeCJK_hd_saved_node_tl }
5392             \tl_gclear:N \g__xeCJK_last_node_tl
5393             \__xeCJK_orig_HD_target:
5394             \tl_if_empty:NF \g__xeCJK_hd_saved_node_tl
5395             { \exp_args:No \xeCJK_make_node:n { \g__xeCJK_hd_saved_node_tl } }
5396         }
5397     }
5398 }
5399 \__xeCJK_package_hook:nm { hypdoc } { \__xeCJK_patch_hd_target: }

```

`url` 的 `\url` 命令通过 `\Url@FormatString` 进入数学模式排版 URL 字符串。数学模式 `node` 隔在 xeCJK 标记 kern 对和数学公式之间, 又因 \XeTeX 在数学模式中不触发类别转换 (`interchar class`), 标记 kern 对永远等不到匹配的 `Boundary→{CJK,Default}` 触发, 缓存的 `CJKecglue` 因此在 CJK 与 `\url` 之间丢失。此处在进入数学模式前先 `drain` (排空) 缓存间距: `\url` 排版的内容必然是西文, 前方上下文为 `CJK / CJK-space / CJK-widow / default` 中任意一种时均应补上 `CJKecglue` (西文边界间距)。`\url` 数学模式结束到后续 CJK 文字之间的右侧间距, 由 `\xeCJK_check_for_glue:` 的 `__xeCJK_if_last_math:` 路径自然处理。

```

5400 \cs_new_protected:Npn \__xeCJK_drain_ecglue:
5401 {
5402     \xeCJK_if_last_node:TF
5403     {
5404         \tl_if_empty:NF \g__xeCJK_last_node_tl
5405         {
5406             \xeCJK_remove_node:
5407             \skip_horizontal:N \l__xeCJK_ecglue_skip
5408         }
5409     }
5410     % 无 marker kern: t1 与实际节点状态不一致, 主动清空避免后续误读残留。
5411     { \tl_gclear:N \g__xeCJK_last_node_tl }
5412 }
5413 \cs_new_protected:Npn \__xeCJK_patch_url_format:
5414 {
5415     \cs_if_exist:NF \__xeCJK_orig_Url_FormatString:
5416     {
5417         \cs_set_eq:NN \__xeCJK_orig_Url_FormatString: \Url@FormatString
5418         \cs_gset_protected:Npn \Url@FormatString

```

```

5419     {
5420         \__xeCJK_drain_ecglue:
5421         \__xeCJK_orig_Url_FormatString:
5422     }
5423 }
5424 }
5425 \__xeCJK_package_hook:nn { url } { \__xeCJK_patch_url_format: }

```

L^AT_EX 2_ε 的 `\verb` 命令展开后会执行 `\leavevmode`, 其中 `\null` 等价于 `\hbox{}`, 会产生一个 `\hbox(0+0)x0` 节点。这个 0×0 的 `\hbox` 紧跟在 `xeCJK` 的标记 `kern` 对之后, 会切断后续 `Boundary`→`{Default, CJK}` 类别转换中 `\tex_lastkern:D` 的探测路径, 导致 `CJK` 文字与 `\verb` 之间应有的 `CJKecglue` 丢失。

修复办法是在执行原始 `\verb` 之前先 `drain` 缓存的间距: 因为 `\verb` 排版的内容必然是西文 (catcode 12 字符), 前文若为 `CJK`, 那么这里就应该补上 `CJKecglue`。而 `\verb` 内最后一个字符到后续 `CJK` 文字之间的右侧间距, 会由 `Default`→`CJK` 类别转换中 `\xeCJK_pre_inter_class_toks:nnn` 自动恢复, 不需要额外处理。

数学模式下, 原 `\verb` 走 `\hbox` 分支, 不会展开 `\leavevmode`, 标记 `kern` 不被 0×0 的 `\hbox` 遮蔽, 因此跳过这个 `patch`。

`shortvrb` 的 `\MakeShortVerb` 把分隔符设为 `active` 字符后展开到 `\verb`, 会自动跟随这里的 `patch`。`\verb*` 与 `\verb` 共用入口 (`\@ifstar` 在 `\bgroup` 之后才扫描), 因此也自动覆盖到。

这里选择 `__xeCJK_after_preamble:n` 而非 `__xeCJK_package_hook:nn`, 因为 `\verb` 是 L^AT_EX 2_ε 核心命令, 不来自宏包; 放在 `preamble` 之后注册, 可以避开 `cprotect`、`fancyvrb` 等宏包对 `\verb` 重定义时的时序冲突。

这里不直接复用 `__xeCJK_drain_ecglue:(#880 中 \Url@FormatString 使用的 drain 函数)`: 后者在没有标记时会主动清掉 `\g__xeCJK_last_node_tl`, 这是为 `math` 模式吃掉 `marker` 的场景做的防御。而 `\verb` 不会进入 `math` 模式, 入口前的 `\g__xeCJK_last_node_tl` 是合法状态——在 `ctex` 模式下, `\verb` 内部会切换 `CJK` 字体 (如 `FandolFang`), `\verb` 内字符到 `\verb` 外字符仍会走 `CJK`→`CJK` 类别转换路径, 此时需要保留 `\g__xeCJK_last_node_tl` 的当前值, 才能正确地输出 `\CJKglue`。因此 `\verb` 专用的 `drain` 只在确实存在标记时移除并补 `ecglue`, 没有标记时 `else` 分支故意留空——与 `__xeCJK_drain_ecglue:` 的唯一区别就在于此, 不调用 `\tl_gc_clear:N` 是为了不破坏入口前的合法状态。

此 `drain` 函数同样被 `__xeCJK_patch_codedoc_meta:(#920 中 l3doc \meta 补丁)` 复用: `\meta` 退出后仍然有 `token` 层面的 `interchar` 处理, 需要保留 `\g__xeCJK_last_node_tl` 的状态。后续若有新的调用点, 请同步更新本节注释。

```

\__xeCJK_drain_ecglue_verb: 5426 \cs_new_protected:Npn \__xeCJK_drain_ecglue_verb:
\__xeCJK_patch_verb:      5427 {
5428     \xeCJK_if_last_node:TF
5429     {
5430         \tl_if_empty:NF \g__xeCJK_last_node_tl
5431         {
5432             \xeCJK_remove_node:
5433             \skip_horizontal:N \l__xeCJK_ecglue_skip
5434         }
5435     }
5436     { }
5437 }
5438 \cs_new_protected:Npn \__xeCJK_patch_verb:
5439 {
5440     \cs_if_exist:NF \__xeCJK_orig_verb:
5441     {
5442         \cs_set_eq:NN \__xeCJK_orig_verb: \verb
5443         \cs_gset_protected:Npn \verb
5444         {
5445             \mode_if_math:F { \__xeCJK_drain_ecglue_verb: }
5446             \__xeCJK_orig_verb:

```

```

5447     }
5448   }
5449 }
5450 \__xeCJK_after_preamble:n { \__xeCJK_patch_verb: }

```

`\l3doc` 类用 `__codedoc_meta:n` 排版 `\meta` 命令的参数：内部由 `__codedoc_meta_original:n` 依次输出 `\ensuremath{\langle}` 数学节点、参数内容、以及对称的 `\ensuremath{\rangle}`。当参数内容的首字符是 CJK 时，`Default→CJK` 类别转换中的 `\xeCJK_pre_inter_class_toks:nmn` 会自动加上 `\CJKeclue`，这会在 `<` 与首个 CJK 字符之间产生多余的间距。

这里采用 myhsia 在 `hyperref / hypdoc` 兼容议题中提出的 `hbox` 包装方案：把 `__codedoc_meta:n` 的参数装进 `\hbox:n`。`hbox` 隔离了内部的 XeTeX 类别转换，`Default→CJK` 转换不会跨越 `hbox` 边界触发，从而避免插入多余的 `\CJKeclue`。

但仅用 `hbox` 包装会带来另一个副作用：`\meta` 之前的 CJK 与 `\meta` 之间的边界 `ecglue` 会丢失——`__codedoc_meta_original:n` 入口的 `\ensuremath math` 节点把缓存的 CJK 标记 `kern` 遮在自己之后，后续类别转换探测不到这个标记，原本应有的 `CJKeclue` 也跟着丢失。与 `url(#880)`、`\verb(#910)`，参见 `__xeCJK_patch_verb:` 同型，这里在 `__codedoc_meta:n` 入口调用 `__xeCJK_drain_ecglue_verb:` 移除标记并补上 `\CJKeclue`，恢复左侧的边界 `ecglue`。复用 `\verb` 版本的 `drain`（else 分支不清 `\g__xeCJK_last_node_tl`）是因为 `\meta` 退出后仍然要经过 `token` 层面的类别转换，需要保留 `\g__xeCJK_last_node_tl` 的当前状态。`__xeCJK_drain_ecglue_verb:` 的注释也列出了本处调用点，二者保持双向引用。

`\rangle` 之后到外部字符的 `ecglue` 仍由 `__codedoc_meta:n` 末尾 `\ensuremath` 退出后的 `Default→CJK` 类别转换正常处理。

`__codedoc_meta:n` 同时被 `\Arg`、`\oarg`、`\parg` 等命令共用，此补丁也间接修复后者的相同问题。

这里选择 `__xeCJK_at_end_preamble:n` 而非 `__xeCJK_package_hook:nn`，原因在于 `\l3doc` 是文档类，文档类加载早于 `xeCJK`，需要等到 `preamble` 结束时才能判定 `__codedoc_meta:n` 是否定义。

```

\__xeCJK_patch_codedoc_meta: 5451 \cs_new_protected:Npn \__xeCJK_patch_codedoc_meta:
5452   {
5453     \cs_if_exist:NT \__codedoc_meta:n
5454     {
5455       \cs_if_exist:NF \__xeCJK_orig_codedoc_meta:n
5456       {
5457         \cs_set_eq:NN \__xeCJK_orig_codedoc_meta:n \__codedoc_meta:n
5458         \cs_gset_protected:Npn \__codedoc_meta:n ##1
5459         {
5460           \mode_if_math:F { \__xeCJK_drain_ecglue_verb: }
5461           \__xeCJK_orig_codedoc_meta:n { \hbox:n {##1} }
5462         }
5463       }
5464     }
5465   }
5466 \__xeCJK_at_end_preamble:n { \__xeCJK_patch_codedoc_meta: }

```

`biblatex` 在开启 `pagetracker` 选项（`authoryear` 等 `bbx` 样式默认开启）时，每次排版 `\bibitem` 会在 `\blx@pagetracker@context` 内调用 `\protected@write` 往 `aux` 写入 `\abx@aux@page` 记录。该 `\write` 产生的 `whatsit` 节点紧接在前一条目结尾的 `default` 标记 `kern` 之后，遮蔽了 `\tex_lastkern:D` 探测路径。当下一条目首字符是 CJK 字符时，`Boundary→CJK` 类别转换的 `\xeCJK_check_for_glue:` 会走到 `__xeCJK_recover_glue_whatst:` 的 `default` 分支——它读到全局 `\g__xeCJK_last_node_tl` 仍是 `default`（前一条目的合法残留），就误吐了一个 `CJKeclue`，表现为条目首字符前多出一段空白。

修复策略与 `hyperref \Hy@BeginAnnot (#809/#810)` 同型：在 `whatsit` 插入前主动清空 `\g__xeCJK_last_node_tl`，让下游 `__xeCJK_recover_glue_whatsit`：探到 `whatsit` 时 `default` 分支条件不成立，不再吐出多余 `ecglue`。

前一条目结尾如果确实需要往下一条目续接 CJK 边界语义(例如后续 `\blx@driver` 内部先输出 CJK 字符)，`\blx@driver` 排版首字符时会由 `\xeCJK_make_node:n` 重新设置 `\g__xeCJK_last_node_tl`，不受此清空影响；因此本补丁只切断“前一条目 marker 通过 `whatsit` 遮蔽错误跨到下一条目”的路径，不影响新条目内部的正常边界恢复。

与 `\Hy@BeginAnnot` 补丁的差异：这里没有“选择性重放 CJK/CJK-space 标记”的需求，因为 `\blx@pagetracker` 入口时前文已经是完整段落结构(`\bibsentence`、`\blx@postpunct` 等)，跨条目续接 `xeCJK` marker 的语义不成立，直接清空即可。

补丁点选 `\blx@pagetracker` 而非 `\blx@pagetracker@context:biblatex` 在 `bbx style` 加载时通过 `\ExecuteBibliographyOptions{pagetracker}` 展开 `\blx@opt@pagetracker@page`，其内部 `\let \blx@pagetracker←\blx@pagetracker@context`——注意是 `**\let` 值传递 `**` 而非 `**` 符号引用 `**`。若补丁挂在 `\blx@pagetracker@context`，只能改到源函数；但 `\blx@bibitem` 实际调用的是 `\blx@pagetracker` 的 `\let` 拷贝，早在 `bbx` 加载时就已定型，之后再改源函数已经无效。故补丁点选 `\blx@pagetracker` 本身，并延迟到 `__xeCJK_at_end_preamble:n` 执行——此时 `bbx` 已完全加载，`\blx@pagetracker` 已 `\let` 到最终目标(可能是 `\blx@pagetracker@context`、`\blx@pagetracker@spread` 或 `\relax`)。

```
\__xeCJK_patch_biblatex_pagetracker: 5467 \cs_new_protected:Npn \__xeCJK_patch_biblatex_pagetracker:
5468 {
5469   \cs_if_exist:NT \blx@pagetracker
5470   {
5471     \cs_if_exist:NF \__xeCJK_orig_blx_pagetracker:
5472     {
5473       \cs_set_eq:NN \__xeCJK_orig_blx_pagetracker: \blx@pagetracker
5474       \cs_gset_protected:Npn \blx@pagetracker
5475       {
5476         \tl_gclear:N \g__xeCJK_last_node_tl
5477         \__xeCJK_orig_blx_pagetracker:
5478       }
5479     }
5480   }
5481 }
5482 \__xeCJK_at_end_preamble:n { \__xeCJK_patch_biblatex_pagetracker: }
```

当探测到 `cprotect` 宏包被引入时，则取消 `\cprotect` 宏的 `\outer` 定义。

```
5483 \__xeCJK_package_hook:n { cprotect }
5484 {
5485   \cs_if_free:NF \icprotect
5486   { \exp_after:wN \tex_let:D \cs:w cprotect \cs_end: \icprotect }
5487 }
```

在 `listings` 宏包后自动载入 `xeCJK-listings`。

```
5488 \__xeCJK_package_hook:n { listings }
5489 { \RequirePackage { xeCJK-listings } }
```

由于 `xeCJK` 假装 `CJK` 已经被引入了，这会可能导致旧版本的 `everysel` 包判断错误。需要在它们判断之前取消定义。

```
5490 \__xeCJK_package_hook:n { everysel }
5491 {
5492   \cs_if_exist:NF \@EverySelectfont@Legacy
5493   { \cs_undefine:c { ver@CJK . \c__xeCJK_package_ext_tl } }
5494 }
```

`\CJKaddEncHook` 为使用 `CJKnumb` 宏包而作一些处理。另外 `CJKnumb` 使用的是传统汉字“萬”和“億”，我们在这里把它们修正为简体字。

```
5495 \ctex_at_begin_package:n { CJKnumb }
```

```

5496 {
5497   \tl_new:N \l__xeCJK_CJK_version_tl
5498   \tl_set_eq:Nc \l__xeCJK_CJK_version_tl { ver@CJK . \c__xeCJK_package_ext_tl }
5499   \tl_set:cn { ver@CJK . \c__xeCJK_package_ext_tl } { 9999/99/99 }
5500   \cs_new_protected:Npn \CJKaddEncHook #1#2
5501     {
5502       \str_if_eq:nnT {#1} { \CJK@UnicodeEnc }
5503         {
5504           \group_begin:
5505             \cs_set_eq:NN \Unicode \xeCJK_unicode_char:nn
5506             \cs_set_eq:NN \def \xdef
5507               #2
5508           \group_end:
5509           \str_gset:Nn \CJK@tenthousand { ^^^^4e07 }
5510           \str_gset:Nn \CJK@hundredmillion { ^^^^4ebf }
5511           \tl_if_exist:NF \CJK@UnicodeEnc
5512             { \tl_const:Nn \CJK@UnicodeEnc { UTF8 } }
5513           \cs_if_exist:NF \Unicode
5514             { \cs_new_eq:NN \Unicode \xeCJK_unicode_char:nn }
5515         }
5516     }
5517   \cs_new:Npn \xeCJK_unicode_char:nn #1#2
5518     { \tex_Uchar:D \tex_numexpr:D (#1) * 256 + (#2) \scan_stop: }
5519 }
5520 \ctex_at_end_package:nn { CJKnumb }
5521 { \tl_set_eq:cN { ver@CJK . \c__xeCJK_package_ext_tl } \l__xeCJK_CJK_version_tl }

```

最后引入本地配置文件。

```

5522 \bool_if:NT \g__xeCJK_config_bool
5523 {
5524   \ExplSyntaxOff
5525   \file_input:n { \g__xeCJK_config_name_tl .cfg }
5526   \ExplSyntaxOn
5527 }
5528 </package>

```

5.19 xeCJKfntef

```

5529 <*/fntef>
5530 \PassOptionsToPackage { normalem } { ulem }
5531 \DeclareOption* { \PassOptionsToPackage { \CurrentOption } { ulem } }
5532 \ProcessOptions \scan_stop:
5533 \RequirePackage { xeCJK }
5534 \RequirePackage { ulem }
5535 \addto@hook \UL@hook { \xeCJK_hook_for_ulem: }
\xeCJK_hook_for_ulem: 5536 \cs_new_protected:Npn \xeCJK_hook_for_ulem:
5537 {
5538   \xeCJK_ulem_detect_node:
5539   \l__xeCJK_ulem_text_format_tl
5540   \bool_if:NF \l__xeCJK_ulem_hook_used_bool
5541     {
5542       \bool_set_true:N \l__xeCJK_ulem_hook_used_bool
5543       \__xeCJK_ulem_hook:
5544     }
5545   \xeCJK_ulem_begin_node:
5546 }
5547 \cs_new_protected:Npn \__xeCJK_ulem_hook:
5548 {
5549   \__xeCJK_ulem_initial:
5550   \bool_if:NT \l__xeCJK_ulem_subtract_bool
5551     {
5552       \xeCJK_swap_cs:NN \UL@leaders \xeCJK_ulem_leaders:
5553       \cs_set_eq:NN \__xeCJK_ulem_var_leaders: \xeCJK_ulem_var_leaders:

```

```

5554     \cs_set_eq:NN \xeCJK_ulem_right_skip: \__xeCJK_ulem_right_skip:
5555   }
5556   \bool_if:NT \l__xeCJK_ulem_hidden_bool
5557     { \cs_set_eq:NN \UL@putbox \__xeCJK_ulem_hidden_box: }
5558   \bool_if:NTF \l__xeCJK_ulem_skip_bool
5559     {
5560     \cs_set_eq:NN \__xeCJK_ulem_putbox: \UL@putbox
5561     \cs_set_eq:NN \__xeCJK_ulem_hskip_aux:n \xeCJK_ulem_hskip:n
5562     }
5563     {
5564     \xeCJK_swap_cs:NN \__xeCJK_punct_hskip:n \__xeCJK_ulem_punct_hskip:n
5565     \xeCJK_cs_clear:N \__xeCJK_ulem_skip_punct_begin:
5566     \xeCJK_cs_clear:N \__xeCJK_ulem_skip_punct_end:
5567     }
5568   \xeCJK_glue_to_skip:nN
5569     {
5570     \cs_set_eq:NN \ \tex_space:D
5571     \cs_set_eq:NN \penalty \tex_penalty:D
5572     \cs_set_eq:NN \hskip \skip_horizontal:N
5573     \CJKglue
5574     } \l__xeCJK_ccglue_skip
5575   \xeCJK_glue_to_skip:nN
5576     {
5577     \cs_set_eq:NN \ \tex_space:D
5578     \cs_set_eq:NN \penalty \tex_penalty:D
5579     \cs_set_eq:NN \hskip \skip_horizontal:N
5580     \CJKecglue
5581     } \l__xeCJK_ecglue_skip
5582   \xeCJK_glue_to_skip:nN { \xeCJK_space_glue: } \l__xeCJK_space_skip
5583   \cs_set_protected:Npn \CJKglue
5584     { \__xeCJK_ulem_glue:n \l__xeCJK_ccglue_skip }
5585   \cs_set_protected:Npn \CJKecglue
5586     { \__xeCJK_ulem_glue:n \l__xeCJK_ecglue_skip }
5587   \cs_set_protected:Npn \xeCJK_space_glue:
5588     { \__xeCJK_ulem_glue:n \l__xeCJK_space_skip }
5589   \cs_set_eq:NN \xeCJK_punct_node:N \use_none:n
5590   \cs_set_eq:NN \xeCJK_if_last_punct:TF \use_ii:nn
5591   \keys_set:nn { xeCJK / options }
5592     { CheckFullRight = false , xCJKecglue = false }
5593   }
5594   \skip_new:N \l__xeCJK_space_skip
5595   \bool_new:N \l__xeCJK_ulem_hook_used_bool

```

\UL@word 修改 \UL@word, 目的是取得分组中的 \UL@leadtype, 以便加入 \xeCJK_ulem_right_skip:。
 \xeCJK_ulem_word:nw

```

5596 \cs_new_protected:Npn \xeCJK_ulem_word:nw #1 ~
5597   {
5598     \__xeCJK_ulem_start:w #1 ~
5599     \exp_after:wN \if_meaning:w \exp_after:wN \UL@end #1
5600     \exp_after:wN \__xeCJK_ulem_end:
5601     \else:
5602     \exp_after:wN \__xeCJK_ulem_loop:nw
5603     \fi:
5604   }
5605 \cs_new_protected:Npn \__xeCJK_ulem_end:
5606   {
5607     \c_group_end_token
5608     \c_group_end_token
5609     \tex_unskip:D \tex_unskip:D \tex_unskip:D
5610     \xeCJK_ulem_right_skip:
5611     \xeCJK_ulem_group_end:
5612     \xeCJK_ulem_right_node:
5613     \tl_gset_eq:NN \g__xeCJK_last_node_tl \g__xeCJK_ulem_saved_last_node_tl
5614     \int_set:Nn \tex_spacefactor:D { \UL@spfactor }
5615   }
5616 \cs_new_protected:Npn \__xeCJK_ulem_loop:nw
5617   {
5618     \reverse_if:N \if_mode_math:

```

```

5619     \reverse_if:N \if_dim:w \tex_lastskip:D = \c_zero_dim
5620     \skip_gset_eq:NN \UL@skip \tex_lastskip:D
5621     \tex_unskip:D
5622     \UL@stop \UL@leaders
5623     \fi:
5624     \fi:
5625     \xeCJK_ulem_word:nw \prg_do_nothing:
5626   }
5627 \cs_new_protected:Npn \__xeCJK_ulem_start:w
5628 { \exp_after:wN \UL@start }
5629 \cs_set_eq:NN \UL@word \xeCJK_ulem_word:nw

```

\xeCJK_ulem_left: 在下划线开始之前探测之前的 node, 以便随后插入 \CJKglue 或 \CJKe glue。

\xeCJK_ulem_detect_node:

```

5630 \cs_new_protected:Npn \xeCJK_ulem_left:
5631 {
5632   \xeCJK_ulem_left_node:
5633   \xeCJK_make_group_tag:
5634 }
5635 \cs_new_eq:NN \xeCJK_ulem_left_node: \prg_do_nothing:
5636 \cs_new_protected:Npn \xeCJK_ulem_detect_node:
5637 {
5638   \scan_stop:
5639   \dim_compare:nNnTF \tex_lastkern:D = \c_zero_dim
5640   {
5641     \xeCJK_cs_clear:N \xeCJK_ulem_left_node:
5642     \xeCJK_cs_clear:N \xeCJK_ulem_begin_node:
5643     \cs_set_eq:NN \__xeCJK_ulem_hskip:n \xeCJK_ulem_hskip:n
5644   }
5645   {
5646     \dim_set_eq:NN \l__xeCJK_tmp_dim \tex_lastkern:D
5647     \tex_unkern:D
5648     \dim_compare:nNnTF \tex_lastkern:D = { - \l__xeCJK_tmp_dim }
5649     {
5650       \tex_unkern:D
5651       \cs_set_protected:Npe \xeCJK_ulem_left_node:
5652       {
5653         \tex_kern:D - \dim_use:N \l__xeCJK_tmp_dim \exp_stop_f:
5654         \tex_kern:D \dim_use:N \l__xeCJK_tmp_dim \exp_stop_f:
5655       }
5656       \cs_set_protected:Npn \xeCJK_ulem_begin_node:
5657       { { \xeCJK_make_node:n { ulem-begin } } }
5658       \cs_set_eq:NN \__xeCJK_ulem_hskip:n \__xeCJK_ulem_hskip_first:n
5659     }
5660     {
5661       \tex_kern:D \l__xeCJK_tmp_dim
5662       \xeCJK_cs_clear:N \xeCJK_ulem_left_node:
5663       \xeCJK_cs_clear:N \xeCJK_ulem_begin_node:
5664       \cs_set_eq:NN \__xeCJK_ulem_hskip:n \xeCJK_ulem_hskip:n
5665     }
5666   }
5667 }
5668 \xeCJK_declare_node:n { ulem-begin }
5669 \cs_new_eq:NN \xeCJK_ulem_begin_node: \prg_do_nothing:

```

__xeCJK_ulem_hskip_first:n

\xeCJK_ulem_hskip:n

如果第一次调用的 \CJKglue 或 \CJKe glue 由下划线中的第一个文字和之前的内容产生, 就不用画下划线。

```

5670 \cs_new_protected:Npn \__xeCJK_ulem_hskip_first:n #1
5671 {
5672   \xeCJK_if_last_node:nTF { ulem-begin }
5673   {
5674     \xeCJK_remove_node:
5675     \skip_horizontal:n {#1}
5676   }
5677   { \xeCJK_ulem_hskip:n {#1} }
5678   \cs_set_eq:NN \__xeCJK_ulem_hskip:n \xeCJK_ulem_hskip:n
5679 }

```

```
5680 \cs_new_eq:NN \__xeCJK_ulem_hskip:n \__xeCJK_ulem_hskip_first:n
5681 \cs_new_protected:Npn \xeCJK_ulem_hskip:n #1
5682 { { \skip_set:Nn \UL@skip {#1} \UL@leaders } }
```

`\xeCJK_ulem_right:` 在下划线最后的位置保存 `node`。

```
\xeCJK_ulem_right_node:
5683 \cs_new_protected:Npn \xeCJK_ulem_right:
5684 {
5685   \scan_stop:
5686   \tl_gset_eq:NN \g__xeCJK_ulem_saved_last_node_tl \g__xeCJK_last_node_tl
5687   \dim_compare:nNnTF \tex_lastkern:D = \c_zero_dim
5688     { \xeCJK_cs_gclear:N \xeCJK_ulem_right_node: }
5689     {
5690       \dim_compare:nNnTF \tex_lastkern:D = { 3sp }
5691         { \xeCJK_cs_gclear:N \xeCJK_ulem_right_node: }
5692         {
5693           \exp_args:NNo \tex_unkern:D
5694           \__xeCJK_ulem_right_aux:n { \dim_use:N \tex_lastkern:D }
5695         }
5696       }
5697   }
5698   \tl_new:N \g__xeCJK_ulem_saved_last_node_tl
5699   \cs_new_protected:Npn \__xeCJK_ulem_right_aux:n #1
5700   {
5701     \dim_compare:nNnTF \tex_lastkern:D = { - #1 }
5702     {
5703       \tex_unkern:D
5704       \cs_gset_protected:Npn \xeCJK_ulem_right_node:
5705       {
5706         \tex_kern:D - #1 \exp_stop_f:
5707         \tex_kern:D #1 \exp_stop_f:
5708       }
5709       \tl_gset:Ne \UL@spfactor { \int_use:N \tex_spacefactor:D }
5710     }
5711     {
5712       \tex_kern:D #1 \exp_stop_f:
5713       \xeCJK_cs_gclear:N \xeCJK_ulem_right_node:
5714     }
5715   }
5716   \cs_new_eq:NN \xeCJK_ulem_right_node: \prg_do_nothing:
```

`\xeCJK_ulem_var_leaders:` 第一次画下划线时, 不需要向左平移 `\UL@pixel`, 让左侧有间距。

```
5717 \cs_new_protected:Npn \xeCJK_ulem_leaders:
5718 { \__xeCJK_ulem_var_leaders: }
5719 \cs_new_protected:Npn \xeCJK_ulem_var_leaders:
5720 {
5721   \scan_stop:
5722   \skip_if_eq:nnF { \UL@skip } { \c_zero_skip }
5723   {
5724     \UL@leadtype \skip_horizontal:n { \UL@skip + \UL@pixel }
5725     \skip_horizontal:n { - \UL@pixel }
5726     \cs_gset_eq:NN \__xeCJK_ulem_var_leaders: \xeCJK_ulem_leaders:
5727   }
5728 }
5729 \cs_new_eq:NN \__xeCJK_ulem_var_leaders: \xeCJK_ulem_var_leaders:
```

`\xeCJK_ulem_right_skip:` 在下划线完全画好之后, 我们检测最后的情况。用 `\unskip` 去掉最后一个下划线, 再重新画一个减少 `\UL@pixel` 的。

```
5730 \cs_new_eq:NN \xeCJK_ulem_right_skip: \prg_do_nothing:
5731 \cs_new_protected:Npn \__xeCJK_ulem_right_skip:
5732 {
5733   \int_case:nn { \tex_lastnodetype:D }
5734   {
5735     { \c__xeCJK_hlist_node } { \__xeCJK_ulem_right_skip_hbox: }
5736     { \c__xeCJK_glue_node } { \__xeCJK_ulem_right_skip_glue: }
5737     { \c__xeCJK_penalty_node } { \__xeCJK_ulem_right_skip_penalty: }
5738   }
```

```

5738     }
5739   }
5740 \cs_new_protected:Npn \__xeCJK_ulem_right_skip_hbox:
5741   {
5742     \box_set_to_last:N \l__xeCJK_tmp_box
5743     \__xeCJK_if_last_kern:TF
5744       { \__xeCJK_ulem_right_skip_kern: }
5745       { \__xeCJK_ulem_right_skip_glue: }
5746     \box_use_drop:N \l__xeCJK_tmp_box
5747   }
5748 \cs_new_protected:Npn \__xeCJK_ulem_right_skip_kern:
5749   {
5750     \dim_set:Nn \l__xeCJK_tmp_dim { - \box_wd:N \l__xeCJK_tmp_box }
5751     \dim_compare:nNnT \tex_lastkern:D = \l__xeCJK_tmp_dim
5752       {
5753         \tex_unkern:D
5754         \__xeCJK_ulem_right_skip_glue:
5755         \tex_kern:D \l__xeCJK_tmp_dim
5756       }
5757   }
5758 \cs_new_protected:Npn \__xeCJK_ulem_right_skip_glue:
5759   {
5760     \skip_if_eq:nnT { \tex_lastskip:D } { - \UL@pixel }
5761     {
5762       \tex_unskip:D
5763       \skip_set:Nn \l__xeCJK_tmp_skip { \tex_lastskip:D - \UL@pixel }
5764       \tex_unskip:D
5765       \UL@leadtype \skip_horizontal:N \l__xeCJK_tmp_skip
5766     }
5767   }
5768 \cs_new_protected:Npn \__xeCJK_ulem_right_skip_penalty:
5769   {
5770     \int_set_eq:NN \l__xeCJK_tmp_int \tex_lastpenalty:D
5771     \tex_unpenalty:D
5772     \__xeCJK_if_last_hlist:T
5773       { \__xeCJK_ulem_right_skip_hbox: }
5774     \tex_penalty:D \l__xeCJK_tmp_int
5775   }

```

`__xeCJK_ulem_hidden_box:` 只画线, 不输出盒子。

```

5776 \cs_new_protected:Npn \__xeCJK_ulem_hidden_box:
5777   {
5778     \tl_if_empty:NF \UL@start
5779     {
5780       \box_set_ht:Nn \l__xeCJK_hidden_box { \box_ht:N \UL@box }
5781       \box_set_dp:Nn \l__xeCJK_hidden_box { \box_dp:N \UL@box }
5782       \box_use:N \l__xeCJK_hidden_box
5783       \xeCJK_no_break:
5784       \xeCJK_ulem_hskip:n { \box_wd:N \UL@box }
5785       \box_use:N \l__xeCJK_hidden_box
5786     }
5787   }
5788 \box_new:N \l__xeCJK_hidden_box
5789 \hbox_set:Nn \l__xeCJK_hidden_box { }

```

`__xeCJK_ulem_skip_punct_begin:` 让下划线跳过标点符号的设置。

`__xeCJK_ulem_skip_punct_end:`

```

5790 \cs_new_protected:Npn \__xeCJK_ulem_skip_punct_begin:
5791   {
5792     \cs_set_eq:NN \UL@putbox \__xeCJK_ulem_skip_putbox:
5793     \cs_set_eq:NN \xeCJK_ulem_hskip:n \skip_horizontal:n
5794   }
5795 \cs_new_protected:Npn \__xeCJK_ulem_skip_punct_end:
5796   {
5797     \cs_set_eq:NN \UL@putbox \__xeCJK_ulem_putbox:
5798     \cs_set_eq:NN \xeCJK_ulem_hskip:n \__xeCJK_ulem_hskip_aux:n

```

```

5799 }
5800 \cs_new_eq:NN \__xeCJK_ulem_putbox: \UL@putbox
5801 \cs_new_protected:Npn \__xeCJK_ulem_skip_putbox:
5802 {
5803   \tl_if_empty:NF \UL@start
5804   { \box_use_drop:N \UL@box }
5805 }

```

__xeCJK_ulem_initial: 这里的设置是为了在下划线状态下,下划线可以自动跳过全角标点符号和正确的在它们前/后断行,并且与行首行末对齐。

```

5806 \cs_new_protected:Npn \__xeCJK_ulem_initial:
5807 {
5808   \__xeCJK_ulem_swap_cs:NN
5809   \xeCJK_FullLeft_and_Default: \__xeCJK_ulem_FullLeft_and_Default:
5810   \xeCJK_FullLeft_and_CJK: \__xeCJK_ulem_FullLeft_and_CJK:
5811   \xeCJK_FullLeft_and_Boundary: \__xeCJK_ulem_FullLeft_and_Boundary:
5812   \xeCJK_FullRight_and_Default: \__xeCJK_ulem_FullRight_and_Default:
5813   \xeCJK_FullRight_and_CJK: \__xeCJK_ulem_FullRight_and_CJK:
5814   \xeCJK_FullRight_and_Boundary: \__xeCJK_ulem_FullRight_and_Boundary:
5815   \xeCJK_CJK_and_CJK:N \__xeCJK_ulem_CJK_and_CJK:N
5816   \xeCJK_CJK_and_Boundary:w \__xeCJK_ulem_CJK_and_Boundary:w
5817   \xeCJK@fix@penalty \__xeCJK_ulem_fix_penalty:
5818   \__xeCJK_punct_breakable_kern:n \__xeCJK_ulem_punct_breakable_kern:n
5819   \__xeCJK_Default_and_FullLeft_glue:N \__xeCJK_ulem_Default_and_FullLeft_glue:N
5820   \__xeCJK_Default_and_FullRight_glue:N \__xeCJK_ulem_Default_and_FullRight_glue:N
5821   \__xeCJK_CJK_and_FullLeft_glue:N \__xeCJK_ulem_CJK_and_FullLeft_glue:N
5822   \__xeCJK_CJK_and_FullRight_glue:N \__xeCJK_ulem_CJK_and_FullRight_glue:N
5823   \__xeCJK_Boundary_and_FullLeft_glue:N \__xeCJK_ulem_Boundary_and_FullLeft_glue:N
5824   \q_recursion_tail \q_nil \q_recursion_stop
5825   \seq_map_inline:Nn \g__xeCJK_CJK_sub_class_seq
5826   {
5827     \seq_map_inline:Nn \g__xeCJK_CJK_sub_class_seq
5828     {
5829       \str_if_eq:nnTF {##1} {####1}
5830       {
5831         \xeCJK_inter_class_toks:nnn { CJK } { CJK/##1 }
5832         { \__xeCJK_ulem_between_CJK_blocks:nnN { CJK } {##1} }
5833         \xeCJK_inter_class_toks:nnn { CJK/##1 } { CJK/##1 }
5834         { \__xeCJK_ulem_between_CJK_blocks:nnN { CJK } {##1} }
5835       }
5836       {
5837         \xeCJK_inter_class_toks:nnn { CJK/##1 } { CJK/####1 }
5838         { \__xeCJK_ulem_between_CJK_blocks:nnN {##1} {####1} }
5839       }
5840     }
5841   }
5842 }
5843 \cs_new_protected:Npn \__xeCJK_ulem_swap_cs:NN #1#2
5844 {
5845   \quark_if_recursion_tail_stop:N #1
5846   \xeCJK_swap_cs:NN #1#2
5847   \__xeCJK_ulem_swap_cs:NN
5848 }

```

\xeCJK_if_ulem_patch:TF 在下划线状态下,ulem 宏包在数学模式或者盒子中使用 \UL@hrest 恢复 _ 等的定义,此时不需要使用 \UL@stop 和 \UL@start 来断开下划线而产生断点。

```

5849 \cs_new:Npn \xeCJK_if_ulem_patch:TF
5850 {
5851   \if_meaning:w \ \LA@space
5852   \exp_after:wN \use_ii:nn
5853   \else:
5854     \exp_after:wN \use_i:nn
5855   \fi:
5856 }

```

```

\__xeCJK_ulem_CJK_and_Boundary:w 5857 \cs_new_protected:Npn \__xeCJK_ulem_CJK_and_Boundary:w
5858 {
5859   \xeCJK_if_ulem_patch:TF
5860   {
5861     \xeCJK_peek_catcode_ignore_spaces:NTF \c_math_toggle_token
5862     { \xeCJK_class_group_end: \CJKeckglue }
5863     {
5864       \bool_if:NTF \l__xeCJK_peek_ignore_spaces_bool
5865       { \__xeCJK_ulem_peek_math:w }
5866       { \__xeCJK_ulem_group_end:n { CJK } }
5867     }
5868   }
5869   { \__xeCJK_ulem_CJK_and_Boundary:w }
5870 }
5871 \cs_new_protected:Npn \__xeCJK_ulem_group_end:n #1
5872 {
5873   \xeCJK_class_group_end: \UL@stop
5874   \UL@start { \xeCJK_make_node:n {#1} }
5875   \xeCJK_make_group_tag:
5876   \bool_gset_true:N \g__xeCJK_glue_check_pending_bool
5877 }

```

__xeCJK_ulem_peek_math:w 用于处理下划线中, 汉字与 \$ 之间有空格的情况¹⁸。

```

5878 \cs_new_protected:Npn \__xeCJK_ulem_peek_math:w
5879 {
5880   \cs_set_eq:NN \__xeCJK_ulem_start:w \__xeCJK_ulem_exp_stop:w
5881   \exp_after:wN \peek_after:Nw
5882   \exp_after:wN \__xeCJK_ulem_peek_math_branches:w
5883   \exp:w \exp_end_continue_f:w
5884 }
5885 \cs_new_protected:Npn \__xeCJK_ulem_peek_math_branches:w
5886 {
5887   \token_if_math_toggle:NTF \l_peek_token
5888   { \xeCJK_class_group_end: \CJKeckglue }
5889   { \__xeCJK_ulem_group_end:n { CJK-space } }
5890 }
5891 \cs_new_protected:Npn \__xeCJK_ulem_exp_stop:w
5892 {
5893   \cs_if_eq:NNTF \UL@start \@empty
5894   { \exp_after:wN \exp_stop_f: }
5895   { \exp_after:wN \UL@start }
5896 }

```

```

\__xeCJK_ulem_fix_penalty: 5897 \cs_new_protected:Npn \__xeCJK_ulem_fix_penalty:
5898 {
5899   \xeCJK_if_ulem_patch:TF
5900   { \fix@penalty }
5901   { \__xeCJK_ulem_fix_penalty: }
5902 }

```

__xeCJK_ulem_save_font_state: 在 \xeCJK_class_group_end: 前保存字体状态, 在 __xeCJK_ulem_class_group_begin: 后恢复, 使 \bfseries、\sffamily 等字体声明能跨越下划线内的 CJK 分组边界。

```

\g__xeCJK_ulem_saved_series_tl 5903 \tl_new:N \g__xeCJK_ulem_saved_series_tl
\g__xeCJK_ulem_saved_shape_tl 5904 \tl_new:N \g__xeCJK_ulem_saved_shape_tl
\g__xeCJK_ulem_saved_CJK_family_tl 5905 \tl_new:N \g__xeCJK_ulem_saved_CJK_family_tl
\g__xeCJK_ulem_saved_CJK_fam_tl 5906 \tl_new:N \g__xeCJK_ulem_saved_CJK_fam_tl
\__xeCJK_ulem_save_font_state: 5907 \cs_new_protected:Npn \__xeCJK_ulem_save_font_state:
5908 {
5909   \tl_gset:Ne \g__xeCJK_ulem_saved_series_tl { \f@series }
5910   \tl_gset:Ne \g__xeCJK_ulem_saved_shape_tl { \f@shape }
5911   \tl_gset_eq:NN \g__xeCJK_ulem_saved_CJK_family_tl \l_xeCJK_family_tl
5912   \tl_gset_eq:NN \g__xeCJK_ulem_saved_CJK_fam_tl \CJK@family
5913 }

```

¹⁸<https://github.com/CTeX-org/ctex-kit/issues/614>

```

5914 \cs_new_protected:Npn \__xeCJK_ulem_restore_font_state:
5915 {
5916   \tl_set:Nc \f@series { \g__xeCJK_ulem_saved_series_tl }
5917   \tl_set:Nc \f@shape { \g__xeCJK_ulem_saved_shape_tl }
5918   \tl_set_eq:NN \l_xeCJK_family_tl \g__xeCJK_ulem_saved_CJK_family_tl
5919   \tl_set_eq:NN \CJK@family \g__xeCJK_ulem_saved_CJK_fam_tl
5920 }

```

```

\__xeCJK_ulem_CJK_and_CJK:N 5921 \cs_new_protected:Npn \__xeCJK_ulem_CJK_and_CJK:N
5922 {
5923   \xeCJK_if_ulem_patch:TF
5924   {
5925     \__xeCJK_ulem_save_font_state:
5926     \xeCJK_class_group_end:
5927     \UL@stop \__xeCJK_ulem_ccglue: \UL@start
5928     \__xeCJK_ulem_class_group_begin:
5929     \__xeCJK_ulem_restore_font_state:
5930     \xeCJK_select_font:
5931     \xeCJK_fallback_symbol:NN
5932     \CJKsymbol
5933   }
5934   { \__xeCJK_ulem_CJK_and_CJK:N }
5935 }

```

```

\__xeCJK_ulem_class_group_begin: 5936 \cs_new_protected:Npn \__xeCJK_ulem_class_group_begin:
5937 {
5938   \xeCJK_class_group_begin:
5939   \xeCJK_clear_Boundary_and_CJK_toks:
5940 }

```

```

\__xeCJK_ulem_between_CJK_blocks:nnN 5941 \cs_new_protected:Npn \__xeCJK_ulem_between_CJK_blocks:nnN #1#2
5942 {
5943   \xeCJK_if_ulem_patch:TF
5944   {
5945     \__xeCJK_ulem_save_font_state:
5946     \xeCJK_class_group_end:
5947     \UL@stop \__xeCJK_ulem_ccglue: \UL@start
5948     \xeCJK_class_group_begin:
5949     \xeCJK_clear_Boundary_and_CJK_toks:
5950     \__xeCJK_ulem_restore_font_state:
5951     \__xeCJK_switch_font:nn {#1} {#2}
5952     \xeCJK_fallback_symbol:NN
5953     \CJKsymbol
5954   }
5955   {
5956     \skip_horizontal:N \l__xeCJK_ccglue_skip
5957     \__xeCJK_switch_font:nn {#1} {#2}
5958     \xeCJK_fallback_symbol:NN
5959     \CJKsymbol
5960   }
5961 }

```

```

\__xeCJK_ulem_Default_and_FullLeft_glue:N 5962 \cs_new_protected:Npn \__xeCJK_ulem_Default_and_FullLeft_glue:N #1
5963 {
5964   \xeCJK_if_ulem_patch:TF
5965   {
5966     \UL@stop
5967     \__xeCJK_ulem_skip_punct_begin:
5968     \__xeCJK_punct_glue:NN \c__xeCJK_left_tl #1
5969     \UL@start
5970   }
5971   { \__xeCJK_ulem_Default_and_FullLeft_glue:N #1 }
5972 }

```

```

\__xeCJK_ulem_Boundary_and_FullLeft_glue:N 5973 \cs_new_protected:Npn \__xeCJK_ulem_Boundary_and_FullLeft_glue:N #1
5974 {
5975   \xeCJK_if_ulem_patch:TF
5976   {
5977     \UL@stop

```

```

5978     \__xeCJK_ulem_skip_punct_begin:
5979     \__xeCJK_punct_glue:NN \c__xeCJK_left_tl #1
5980     \UL@start
5981     }
5982     { \__xeCJK_ulem_Boundary_and_FullLeft_glue:N #1 }
5983     }

```

```

\__xeCJK_ulem_CJK_and_FullLeft_glue:N 5984 \cs_new_protected:Npn \__xeCJK_ulem_CJK_and_FullLeft_glue:N #1
5985     {
5986     \xeCJK_if_ulem_patch:TF
5987     {
5988     \xeCJK_class_group_end:
5989     \UL@stop
5990     \__xeCJK_ulem_skip_punct_begin:
5991     \__xeCJK_ulem_punct_ccglue:
5992     \__xeCJK_punct_glue:NN \c__xeCJK_left_tl #1
5993     \UL@start
5994     \__xeCJK_ulem_class_group_begin:
5995     \xeCJK_select_punct_font:
5996     }
5997     { \__xeCJK_ulem_CJK_and_FullLeft_glue:N #1 }
5998     }

```

```

\__xeCJK_ulem_Default_and_FullRight_glue:N 5999 \cs_new_protected:Npn \__xeCJK_ulem_Default_and_FullRight_glue:N #1
6000     {
6001     \xeCJK_if_ulem_patch:TF
6002     {
6003     \UL@stop
6004     \__xeCJK_ulem_skip_punct_begin:
6005     \__xeCJK_punct_if_long:NTF #1
6006     { \xeCJK_allow_break: }
6007     { \xeCJK_no_break: }
6008     \__xeCJK_punct_if_middle:NT #1
6009     {
6010     \__xeCJK_punct_glue:NN \c__xeCJK_right_tl #1
6011     \__xeCJK_punct_rule:NN \c__xeCJK_left_tl #1
6012     }
6013     \UL@start
6014     }
6015     { \__xeCJK_ulem_Default_and_FullRight_glue:N #1 }
6016     }

```

```

\__xeCJK_ulem_CJK_and_FullRight_glue:N 6017 \cs_new_protected:Npn \__xeCJK_ulem_CJK_and_FullRight_glue:N #1
6018     {
6019     \xeCJK_if_ulem_patch:TF
6020     {
6021     \xeCJK_class_group_end:
6022     \UL@stop
6023     \__xeCJK_ulem_skip_punct_begin:
6024     \__xeCJK_punct_if_long:NTF #1
6025     { \xeCJK_allow_break: }
6026     { \xeCJK_no_break: }
6027     \__xeCJK_punct_if_middle:NT #1
6028     {
6029     \__xeCJK_ulem_punct_ccglue:
6030     \__xeCJK_punct_glue:NN \c__xeCJK_right_tl #1
6031     \__xeCJK_punct_rule:NN \c__xeCJK_left_tl #1
6032     }
6033     \UL@start
6034     \__xeCJK_ulem_class_group_begin:
6035     \xeCJK_select_punct_font:
6036     }
6037     { \__xeCJK_ulem_CJK_and_FullRight_glue:N #1 }
6038     }

```

```

\__xeCJK_ulem_FullLeft_and_Default: 6039 \cs_new_protected:Npn \__xeCJK_ulem_FullLeft_and_Default:
6040     {
6041     \xeCJK_if_ulem_patch:TF

```

```

6042     {
6043         \__xeCJK_punct_if_middle:NTF \g__xeCJK_last_punct_tl
6044         {
6045             \xeCJK_get_punct_bounds:No \c__xeCJK_left_tl \g__xeCJK_last_punct_tl
6046             \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
6047             \xeCJK_class_group_end: \UL@stop \xeCJK_no_break:
6048             \__xeCJK_punct_glue:NN \c__xeCJK_left_tl \g__xeCJK_last_punct_tl
6049         }
6050         { \xeCJK_class_group_end: \UL@stop }
6051         \__xeCJK_ulem_skip_punct_end:
6052         \xeCJK_no_break:
6053         \UL@start
6054     }
6055     { \__xeCJK_ulem_FullLeft_and_Default: }
6056 }

```

```

\__xeCJK_ulem_FullLeft_and_Boundary: 6057 \cs_new_protected:Npn \__xeCJK_ulem_FullLeft_and_Boundary:
6058 {
6059     \xeCJK_if_ulem_patch:TF
6060     {
6061         \__xeCJK_punct_if_middle:NTF \g__xeCJK_last_punct_tl
6062         {
6063             \xeCJK_get_punct_bounds:No \c__xeCJK_left_tl \g__xeCJK_last_punct_tl
6064             \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
6065             \xeCJK_class_group_end: \UL@stop \xeCJK_no_break:
6066             \__xeCJK_punct_glue:NN \c__xeCJK_left_tl \g__xeCJK_last_punct_tl
6067         }
6068         { \xeCJK_class_group_end: \UL@stop }
6069         \__xeCJK_ulem_skip_punct_end:
6070         \xeCJK_no_break:
6071         \UL@start
6072         \tex_ignorespaces:D
6073     }
6074     { \__xeCJK_ulem_FullLeft_and_Boundary: }
6075 }

```

```

\__xeCJK_ulem_FullLeft_and_CJK: 6076 \cs_new_protected:Npn \__xeCJK_ulem_FullLeft_and_CJK:
6077 {
6078     \xeCJK_if_ulem_patch:TF
6079     {
6080         \xeCJK_FullLeft_and_Default:
6081         \__xeCJK_ulem_class_group_begin:
6082         \xeCJK_select_font:
6083     }
6084     { \__xeCJK_ulem_FullLeft_and_CJK: }
6085 }

```

```

\__xeCJK_ulem_FullRight_and_Default: 6086 \cs_new_protected:Npn \__xeCJK_ulem_FullRight_and_Default:
6087 {
6088     \xeCJK_if_ulem_patch:TF
6089     {
6090         \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
6091         \xeCJK_class_group_end:
6092         \UL@stop
6093         \__xeCJK_punct_glue:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
6094         \__xeCJK_ulem_skip_punct_end:
6095         \UL@start
6096     }
6097     { \__xeCJK_ulem_FullRight_and_Default: }
6098 }

```

```

\__xeCJK_ulem_FullRight_and_Boundary: 6099 \cs_new_protected:Npn \__xeCJK_ulem_FullRight_and_Boundary:
6100 {
6101     \xeCJK_if_ulem_patch:TF
6102     {
6103         \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
6104         \xeCJK_class_group_end:
6105         \UL@stop

```

```

6106     \__xeCJK_punct_glue:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
6107     \__xeCJK_ulem_skip_punct_end:
6108     \UL@start
6109     \tex_ignorespaces:D
6110   }
6111   { \__xeCJK_ulem_FullRight_and_Boundary: }
6112 }

```

```

\__xeCJK_ulem_FullRight_and_CJK: 6113 \cs_new_protected:Npn \__xeCJK_ulem_FullRight_and_CJK:
6114 {
6115   \xeCJK_if_ulem_patch:TF
6116   {
6117     \__xeCJK_punct_rule:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
6118     \xeCJK_class_group_end:
6119     \UL@stop
6120     \__xeCJK_punct_glue:NN \c__xeCJK_right_tl \g__xeCJK_last_punct_tl
6121     \__xeCJK_ulem_punct_ccglue:
6122     \__xeCJK_ulem_skip_punct_end:
6123     \UL@start
6124     \__xeCJK_ulem_class_group_begin:
6125     \xeCJK_select_font:
6126   }
6127   { \__xeCJK_ulem_FullRight_and_CJK: }
6128 }

```

```

\__xeCJK_ulem_punct_hskip:n 6129 \cs_new_protected:Npn \__xeCJK_ulem_punct_hskip:n
6130 {
6131   \xeCJK_if_ulem_patch:TF
6132   { \xeCJK_ulem_hskip:n }
6133   { \__xeCJK_ulem_punct_hskip:n }
6134 }

```

```

\__xeCJK_ulem_punct_breakable_kern:n 6135 \cs_new_protected:Npn \__xeCJK_ulem_punct_breakable_kern:n #1
6136 {
6137   \xeCJK_if_ulem_patch:TF
6138   {
6139     \xeCJK_class_group_end:
6140     \UL@stop \xeCJK_ulem_hskip:n {#1} \UL@start
6141     \__xeCJK_ulem_class_group_begin:
6142     \xeCJK_select_punct_font:
6143   }
6144   { \__xeCJK_ulem_punct_breakable_kern:n {#1} }
6145 }

```

__xeCJK_ulem_glue:n 在下划线状态下的分别代替 \CJKglue 等。

```

\__xeCJK_ulem_ccglue: 6146 \cs_new_protected:Npn \__xeCJK_ulem_glue:n #1
\__xeCJK_ulem_punct_ccglue: 6147 {
6148   \xeCJK_if_ulem_patch:TF
6149   {
6150     \tl_if_empty:NTF \l__xeCJK_group_tag_tl
6151     { \UL@stop \__xeCJK_ulem_hskip:n {#1} \UL@start }
6152     {
6153       \str_if_eq:eeTF { \l__xeCJK_group_tag_tl } { \c__xeCJK_group_tag_tl }
6154       { \UL@stop \__xeCJK_ulem_hskip:n {#1} \UL@start }
6155       { \skip_horizontal:n {#1} }
6156     }
6157   }
6158   { \skip_horizontal:n {#1} }
6159 }
6160 \cs_new_protected:Npn \xeCJK_make_group_tag:
6161 { \tl_set:Ne \l__xeCJK_group_tag_tl { \c__xeCJK_group_tag_tl } }
6162 \tl_new:N \l__xeCJK_group_tag_tl
6163 \tl_const:Nn \c__xeCJK_group_tag_tl
6164 {
6165   T \int_use:N \tex_currentgroupstype:D
6166   L \int_use:N \tex_currentgrouplevel:D
6167 }

```

```

6168 \cs_new_protected:Npn \__xeCJK_ulem_ccglue:
6169   { { \skip_set_eq:NN \UL@skip \l__xeCJK_ccglue_skip \UL@leaders } }
6170 \cs_new_protected:Npn \__xeCJK_ulem_punct_ccglue:
6171   { \__xeCJK_punct_hskip:n { \l__xeCJK_ccglue_skip } }

\xeCJK_ulem_group_begin: 6172 \cs_new_protected:Npn \xeCJK_ulem_group_begin:
\xeCJK_ulem_group_end:   6173   {
\xeCJK_ulem_on:n        6174     \mode_leave_vertical:
                        6175     \c_group_begin_token
                        6176   }
6177 \cs_new_protected:Npn \xeCJK_ulem_group_end:
6178   { \c_group_end_token }
6179 \cs_new_protected:Npn \xeCJK_ulem_on:n
6180   { \ULon }
6181 \cs_new_eq:NN \__xeCJK_ulem_on:n \UL@on
6182 \cs_set_protected:Npn \UL@on #1
6183   { \__xeCJK_ulem_on:n { \xeCJK_ulem_left: #1 \xeCJK_ulem_right: } }

```

\xeCJKfntefon 扩展 \ULon 的参数。

```

6184 \NewDocumentCommand \xeCJKfntefon { s t- s o }
6185   {
6186     \mode_leave_vertical:
6187     \xeCJK_ulem_boot:NNNn #1#2#3 {#4}
6188     \xeCJK_ulem_on:n
6189   }

```

\CJKunderline

```

6190 \NewDocumentCommand \CJKunderline { s t- s o }
6191   {
6192     \xeCJK_ulem_group_begin:
6193     \xeCJK_fntef_boot:nnNNNn { underline } { uline } #1#2#3 {#4}
6194     \xeCJK_fntef_initial:nnn
6195     { \l__xeCJK_uline_depth_tl }
6196     { \l__xeCJK_uline_sep_tl }
6197     {
6198       \l__xeCJK_uline_format_tl
6199       \tex_vrule:D
6200       height \dim_eval:n { \l__xeCJK_uline_thickness_tl }
6201       depth \c_zero_dim
6202       width .2em
6203     }
6204     \xeCJK_ulem_on:n
6205   }
6206 \NewDocumentCommand \varCJKunderline { }
6207   { \CJKunderline - }

```

\CJKunderwave

```

6208 \NewDocumentCommand \CJKunderwave { s t- s o }
6209   {
6210     \xeCJK_ulem_group_begin:
6211     \xeCJK_fntef_boot:nnNNNn { underwave } { uwave } #1#2#3 {#4}
6212     \xeCJK_fntef_initial:nnn
6213     { \l__xeCJK_uwave_depth_tl }
6214     { \l__xeCJK_uwave_sep_tl }
6215     { \l__xeCJK_uwave_format_tl \l__xeCJK_uwave_symbol_tl }
6216     \xeCJK_ulem_on:n
6217   }

```

\CJKunderdblline

```

6218 \NewDocumentCommand \CJKunderdblline { s t- s o }
6219   {
6220     \xeCJK_ulem_group_begin:
6221     \xeCJK_fntef_boot:nnNNNn { underdblline } { udbline } #1#2#3 {#4}
6222     \xeCJK_fntef_initial:nnn
6223     { \l__xeCJK_udblline_depth_tl }
6224     { \l__xeCJK_udblline_sep_tl }
6225     {
6226       \l__xeCJK_udblline_format_tl
6227       \vbox_top:n
6228       {
6229         \tex_hrline:D

```

```

6230         height \dim_eval:n { \l__xeCJK_udbline_thickness_tl }
6231         depth \c_zero_dim
6232         width .2em
6233         \tex_kern:D \dim_eval:n { \l__xeCJK_udbline_gap_tl }
6234         \tex_hruler:D
6235         height \dim_eval:n { \l__xeCJK_udbline_thickness_tl }
6236         depth \c_zero_dim
6237         width .2em
6238     }
6239 }
6240 \xeCJK_ulem_on:n
6241 }

```

```

\CJKsout 6242 \NewDocumentCommand \CJKsout { s t- s o }
6243 {
6244     \xeCJK_ulem_group_begin:
6245     \xeCJK_fntef_boot:nnNNNn { sout } { sout } #1#2#3 {#4}
6246     \xeCJK_fntef_initial:nn
6247     {
6248         \l__xeCJK_sout_format_tl
6249         \tex_vrule:D
6250         height \dim_eval:n { \l__xeCJK_sout_thickness_tl }
6251         depth \c_zero_dim
6252         width .2em
6253     }
6254     {
6255         \box_move_up:nn
6256         { \l__xeCJK_sout_height_tl - \box_ht:N \l__xeCJK_fntef_box / 2 }
6257         { \box_use:N \l__xeCJK_fntef_box }
6258     }
6259     \xeCJK_ulem_on:n
6260 }

```

```

\CJKxout 6261 \NewDocumentCommand \CJKxout { s t- s o }
6262 {
6263     \xeCJK_ulem_group_begin:
6264     \xeCJK_fntef_boot:nnNNNn { xout } { xout } #1#2#3 {#4}
6265     \xeCJK_fntef_initial:nn
6266     {
6267         \l__xeCJK_xout_format_tl
6268         \tex_kern:D -.1 em $/$
6269         \tex_kern:D -.1 em
6270     }
6271     {
6272         \box_move_up:nn
6273         { \box_dp:N \l__xeCJK_fntef_box / 2 }
6274         { \box_use:N \l__xeCJK_fntef_box }
6275     }
6276     \xeCJK_ulem_on:n
6277 }

```

```

\CJKunderanyline 6278 \NewDocumentCommand \CJKunderanyline { s t- s o m m }
6279 {
6280     \xeCJK_ulem_group_begin:
6281     \xeCJK_ulem_boot:NNNn #1#2#3 {#4}
6282     \xeCJK_fntef_initial:nn
6283     {#6}
6284     {
6285         \box_move_down:nn
6286         {#5}
6287         { \box_use:N \l__xeCJK_fntef_box }
6288     }
6289     \tl_if_empty:NF \l__xeCJK_ulem_boxdepth_tl
6290     { \box_set_dp:Nn \ULC@box { \l__xeCJK_ulem_boxdepth_tl } }
6291     \tl_if_empty:NF \l__xeCJK_ulem_sep_tl
6292     {
6293         \bool_set_true:N \l__xeCJK_fntef_bool
6294         \dim_set:Nn \l__xeCJK_fntef_dim

```

```

6295         { \l__xeCJK_ulem_sep_tl + \box_dp:N \ULC@box }
6296     }
6297     \xeCJK_ulem_on:n
6298 }

```

\xeCJK_fntef_boot:nnNNn 处理参数问题。

```

6299 \cs_new_protected:Npn \xeCJK_fntef_boot:nnNNn #1#2#3#4#5#6
6300 {
6301     \bool_lazy_or:nnT {#3} {#5}
6302     { \bool_set_false:c { l__xeCJK_#2_skip_bool } }
6303     \bool_if:NT #4
6304     { \bool_set_true:c { l__xeCJK_#2_subtract_bool } }
6305     \tl_if_novalue:nF {#6}
6306     { \keys_set:nn { xeCJK / options / #1 } {#6} }
6307     \bool_set_eq:Nc \l__xeCJK_ulem_skip_bool { l__xeCJK_#2_skip_bool }
6308     \bool_set_eq:Nc \l__xeCJK_ulem_hidden_bool { l__xeCJK_#2_hidden_bool }
6309     \bool_set_eq:Nc \l__xeCJK_ulem_subtract_bool { l__xeCJK_#2_subtract_bool }
6310     \tl_set_eq:Nc \l__xeCJK_ulem_text_format_tl { l__xeCJK_#2_text_format_tl }
6311 }
6312 \cs_new_protected:Npn \xeCJK_ulem_boot:NNNn #1#2#3#4
6313 {
6314     \bool_lazy_or:nnT {#1} {#3}
6315     { \bool_set_false:N \l__xeCJK_ulem_skip_bool }
6316     \bool_if:NT #2
6317     { \bool_set_true:N \l__xeCJK_ulem_subtract_bool }
6318     \tl_if_novalue:nF {#4}
6319     { \keys_set:nn { xeCJK / options / ulem } {#4} }
6320 }

```

\xeCJK_fntef_initial:n 不支持下划线的嵌套使用。下划线嵌套使用时，里层的下划线会被放在盒子里，不能折行。

```

6321 \cs_new_protected:Npn \xeCJK_fntef_initial:n
6322 {
6323     \bool_if:NTF \l__xeCJK_nest_bool
6324     { \__xeCJK_warning:n { fntef-nesting } }
6325     {
6326         \bool_set_true:N \l__xeCJK_nest_bool
6327         \__xeCJK_restore_shipout_fntef:
6328     }
6329     \xeCJK_fntef_sbox:n
6330 }
6331 \cs_new_protected:Npn \xeCJK_fntef_initial:nn #1
6332 {
6333     \xeCJK_fntef_initial:n {#1}
6334     \bool_if:NF \l__xeCJK_fntef_bool
6335     { \dim_zero:N \l__xeCJK_fntef_dim }
6336     \markoverwith
6337 }
6338 \cs_new_protected:Npn \xeCJK_fntef_initial:nnn #1#2#3
6339 {
6340     \xeCJK_fntef_initial:n {#3}
6341     \bool_if:NF \l__xeCJK_fntef_bool
6342     {
6343         \bool_set_true:N \l__xeCJK_fntef_bool
6344         \dim_set:Nn \l__xeCJK_fntef_dim {#1}
6345     }
6346     \markoverwith
6347     {
6348         \box_move_down:nn
6349         { \l__xeCJK_fntef_dim + \box_ht:N \l__xeCJK_fntef_box }
6350         { \box_use:N \l__xeCJK_fntef_box }
6351     }
6352     \dim_set:Nn \l__xeCJK_fntef_dim { #2 + \box_dp:N \ULC@box }
6353 }
6354 \box_new:N \l__xeCJK_fntef_box
6355 \cs_new_eq:NN \xeCJKfntefbox \l__xeCJK_fntef_box
6356 \bool_new:N \l__xeCJK_nest_bool

```

```

6357 \bool_new:N \l__xeCJK_fntef_bool
6358 \__xeCJK_msg_new:nn { fntef-nesting }
6359 { Nesting~is~not~supported. }

```

`\l__xeCJK_fntef_dim` 记录下划线或者下划符号的深度, 以便它们嵌套使用时能自动调整好距离。`\ULdepth` 被 `ulem` 初始化为 `\maxdimen`。下划线嵌套时, `ulem` 要使用它作计算, 可能会溢出。为简便起见, `\l__xeCJK_fntef_dim` 与 `\ULdepth` 共用一个寄存器。

```
6360 \cs_new_eq:NN \l__xeCJK_fntef_dim \ULdepth
```

`\xeCJK_fntef_sbox:n` 与 `\hcoffin_set:Nn` 和 L^AT_EX_{2 ϵ} 的 `\sbox` 功能类似, 确保颜色的正确。虽然 `coffin` 可以更方便的操作盒子, 但速度要慢一点。并且, 我们的需求也比较简单, 就不用它了。

```

6361 \cs_new_protected:Npn \xeCJK_fntef_sbox:n #1
6362 {
6363   \tl_set_eq:NN \l__xeCJK_tmp_tl \g__xeCJK_last_node_tl
6364   \hbox_set:Nn \l__xeCJK_fntef_box
6365     { \color_ensure_current: #1 }
6366   \tl_gset_eq:NN \g__xeCJK_last_node_tl \l__xeCJK_tmp_tl
6367 }

```

最合适的是用 `xtemplate` 宏包来实现, 但是比较难于用 `\xeCJKsetup` 来统一设置, 所以这里还是用土办法。

```

6368 \keys_define:nn { xeCJK / options }
6369 {
6370   underdot / symbol      .tl_set:N = \l__xeCJK_udot_symbol_tl ,
6371   underdot / depth      .tl_set:N = \l__xeCJK_udot_depth_tl ,
6372   underdot / sep        .tl_set:N = \l__xeCJK_udot_sep_tl ,
6373   underdot / format     .tl_set:N = \l__xeCJK_udot_format_tl ,
6374   underdot / textformat .tl_set:N = \l__xeCJK_udot_text_format_tl ,
6375   underdot / boxdepth   .tl_set:N = \l__xeCJK_udot_boxdepth_tl ,
6376   symbol / sep          .tl_set:N = \l__xeCJK_symbol_sep_tl ,
6377   symbol / boxdepth     .tl_set:N = \l__xeCJK_symbol_boxdepth_tl ,
6378   symbol / textformat   .tl_set:N = \l__xeCJK_symbol_text_format_tl ,
6379   underline / skip      .bool_set:N = \l__xeCJK_uline_skip_bool ,
6380   underline / hidden    .bool_set:N = \l__xeCJK_uline_hidden_bool ,
6381   underline / subtract  .bool_set:N = \l__xeCJK_uline_subtract_bool ,
6382   underline / thickness .tl_set:N = \l__xeCJK_uline_thickness_tl ,
6383   underline / depth     .tl_set:N = \l__xeCJK_uline_depth_tl ,
6384   underline / sep       .tl_set:N = \l__xeCJK_uline_sep_tl ,
6385   underline / format    .tl_set:N = \l__xeCJK_uline_format_tl ,
6386   underline / textformat .tl_set:N = \l__xeCJK_uline_text_format_tl ,
6387   underdblline / skip   .bool_set:N = \l__xeCJK_udbline_skip_bool ,
6388   underdblline / hidden .bool_set:N = \l__xeCJK_udbline_hidden_bool ,
6389   underdblline / subtract .bool_set:N = \l__xeCJK_udbline_subtract_bool ,
6390   underdblline / thickness .tl_set:N = \l__xeCJK_udbline_thickness_tl ,
6391   underdblline / depth  .tl_set:N = \l__xeCJK_udbline_depth_tl ,
6392   underdblline / sep    .tl_set:N = \l__xeCJK_udbline_sep_tl ,
6393   underdblline / format .tl_set:N = \l__xeCJK_udbline_format_tl ,
6394   underdblline / textformat .tl_set:N = \l__xeCJK_udbline_text_format_tl ,
6395   underdblline / gap    .tl_set:N = \l__xeCJK_udbline_gap_tl ,
6396   underwave / skip      .bool_set:N = \l__xeCJK_uwave_skip_bool ,
6397   underwave / hidden    .bool_set:N = \l__xeCJK_uwave_hidden_bool ,
6398   underwave / subtract  .bool_set:N = \l__xeCJK_uwave_subtract_bool ,
6399   underwave / symbol    .tl_set:N = \l__xeCJK_uwave_symbol_tl ,
6400   underwave / depth     .tl_set:N = \l__xeCJK_uwave_depth_tl ,
6401   underwave / sep       .tl_set:N = \l__xeCJK_uwave_sep_tl ,
6402   underwave / format    .tl_set:N = \l__xeCJK_uwave_format_tl ,
6403   underwave / textformat .tl_set:N = \l__xeCJK_uwave_text_format_tl ,
6404   sout / skip           .bool_set:N = \l__xeCJK_sout_skip_bool ,
6405   sout / hidden        .bool_set:N = \l__xeCJK_sout_hidden_bool ,
6406   sout / subtract      .bool_set:N = \l__xeCJK_sout_subtract_bool ,
6407   sout / thickness     .tl_set:N = \l__xeCJK_sout_thickness_tl ,
6408   sout / height        .tl_set:N = \l__xeCJK_sout_height_tl ,
6409   sout / format        .tl_set:N = \l__xeCJK_sout_format_tl ,
6410   sout / textformat    .tl_set:N = \l__xeCJK_sout_text_format_tl ,

```

```

6411 xout / skip .bool_set:N = \l__xeCJK_xout_skip_bool ,
6412 xout / hidden .bool_set:N = \l__xeCJK_xout_hidden_bool ,
6413 xout / subtract .bool_set:N = \l__xeCJK_xout_subtract_bool ,
6414 xout / format .tl_set:N = \l__xeCJK_xout_format_tl ,
6415 xout / textformat .tl_set:N = \l__xeCJK_xout_text_format_tl ,
6416 ulem / skip .bool_set:N = \l__xeCJK_ulem_skip_bool ,
6417 ulem / hidden .bool_set:N = \l__xeCJK_ulem_hidden_bool ,
6418 ulem / subtract .bool_set:N = \l__xeCJK_ulem_subtract_bool ,
6419 ulem / sep .tl_set:N = \l__xeCJK_ulem_sep_tl ,
6420 ulem / boxdepth .tl_set:N = \l__xeCJK_ulem_boxdepth_tl ,
6421 ulem / textformat .tl_set:N = \l__xeCJK_ulem_text_format_tl
6422 }
6423 \clist_map_inline:nn
6424 { underdot , underline , underdblline , underwave , sout , xout , ulem }
6425 {
6426 \keys_define:nn { xeCJK / options }
6427 { #1 .meta:nn = { xeCJK / options / #1 } { ##1 } }
6428 }
6429 \keys_set:nn { xeCJK / options }
6430 {
6431 underdot / symbol = \normalfont . ,
6432 underdot / depth = 0.20 em ,
6433 underdot / sep = 0.04 em ,
6434 symbol / sep = \c_zero_dim ,
6435 underline / skip = true ,
6436 underline / thickness = \ULthickness ,
6437 underline / depth = 0.20 em ,
6438 underline / sep = 0.07 em ,
6439 underdblline / skip = true ,
6440 underdblline / thickness = \ULthickness ,
6441 underdblline / depth = 0.20 em ,
6442 underdblline / sep = 0.17 em ,
6443 underdblline / gap = 1.1 pt ,
6444 underwave / skip = true ,
6445 underwave / symbol = \sixty \tex_char:D 58 \exp_stop_f: ,
6446 underwave / depth = 0.20 em ,
6447 underwave / sep = 0.00 em ,
6448 sout / skip = true ,
6449 sout / thickness = \ULthickness ,
6450 sout / height = 0.35 em ,
6451 xout / skip = true
6452 }

```

```

\CJKunderanysymbol 6453 \NewDocumentCommand \CJKunderanysymbol { o m m m }
6454 {
6455 \xeCJK_under_symbol:nnnnnn { symbol } { symbol } {#1} {#2} {#3} {#4}
6456 }

```

\CJKunderdot \CJKunderdot 是 \CJKunderanysymbol 的特殊情况。**CJKfntef** 原来使用的是数学符号 `\cdot`，这里改成更合适的 `。`。

```

6457 \NewDocumentCommand \CJKunderdot { o m }
6458 {
6459 \xeCJK_under_symbol:nnnnnn { underdot } { udot }
6460 {#1}
6461 { \l__xeCJK_udot_depth_tl }
6462 { \l__xeCJK_udot_format_tl \l__xeCJK_udot_symbol_tl }
6463 {#2}
6464 }

```

`\xeCJK_under_symbol:nnnnnn` 当处在下划线中时，我们先断开下划线，在分组外设置下划符号。

```

6465 \cs_new_protected:Npn \xeCJK_under_symbol:nnnnnn
6466 {
6467 \xeCJK_if_ulem_patch:TF
6468 { \__xeCJK_under_symbol_auxi:nnnnnn }
6469 { \__xeCJK_under_symbol_auxii:nnnnnn }

```

```

6470 }
6471 \cs_new_protected:Npn \__xeCJK_under_symbol_auxi:nnnnn #1#2#3#4#5#6
6472 {
6473   \xeCJK_ulem_right: \UL@stop
6474   \group_begin:
6475     \xeCJK_under_symbol_initial:nnnnn {#1} {#2} {#3} {#4} {#5}
6476     \use:c { l__xeCJK_#2_text_format_tl }
6477     \UL@start \xeCJK_ulem_right_node:
6478       #6
6479     \xeCJK_ulem_right: \UL@stop
6480   \group_end:
6481   \UL@start \xeCJK_ulem_right_node:
6482 }
6483 \cs_new_protected:Npn \__xeCJK_under_symbol_auxii:nnnnn #1#2#3#4#5#6
6484 {
6485   \mode_leave_vertical:
6486   \group_begin:
6487     \xeCJK_under_symbol_initial:nnnnn {#1} {#2} {#3} {#4} {#5}
6488     \__xeCJK_under_symbol_text_format:c { l__xeCJK_#2_text_format_tl }
6489     #6
6490     \xeCJK_ulem_right:
6491   \group_end:
6492   \xeCJK_ulem_right_node:
6493   \bool_gset_true:N \g__xeCJK_glue_check_pending_bool
6494 }
6495 \cs_new_protected:Npn \xeCJK_under_symbol_initial:nnnnn #1#2#3#4#5
6496 {
6497   \tl_if_novalue:nF {#3}
6498   { \keys_set:nn { xeCJK / options / #1 } {#3} }
6499   \xeCJK_fntef_sbox:n {#5}
6500   \bool_if:NTF \l__xeCJK_fntef_bool
6501   { \xeCJK_make_under_symbol:n { \l__xeCJK_fntef_dim } }
6502   {
6503     \bool_set_true:N \l__xeCJK_fntef_bool
6504     \xeCJK_make_under_symbol:n {#4}
6505   }
6506   \tl_if_empty:cF { l__xeCJK_#2_boxdepth_tl }
6507   {
6508     \box_set_dp:Nn \l__xeCJK_under_symbol_box
6509     { \use:c { l__xeCJK_#2_boxdepth_tl } }
6510   }
6511   \dim_set:Nn \l__xeCJK_fntef_dim
6512   { \use:c { l__xeCJK_#2_sep_tl } + \box_dp:N \l__xeCJK_under_symbol_box }
6513   \xeCJK_swap_cs:NN \CJKsymbol \__xeCJK_under_CJKsymbol:N
6514   \__xeCJK_restore_shipout_CJKsymbol:
6515 }
6516 \cs_new_protected:Npn \__xeCJK_under_symbol_text_format:N #1
6517 {
6518   \tl_if_empty:NF #1
6519   { \xeCJK_ulem_right: #1 \xeCJK_ulem_right_node: }
6520 }
6521 \cs_generate_variant:Nn \__xeCJK_under_symbol_text_format:N { c }
6522 \box_new:N \l__xeCJK_under_symbol_box

```

\xeCJK_make_under_symbol:n 我们量取“一”的宽度作为汉字的宽度。

```

6523 \cs_new_protected:Npn \xeCJK_make_under_symbol:n #1
6524 {
6525   \hbox_set:Nn \l__xeCJK_under_symbol_box
6526   {
6527     \box_move_down:nn { #1 + \box_ht:N \l__xeCJK_fntef_box }
6528     {
6529       \hbox_to_zero:n
6530       {
6531         \xeCJK_select_font:
6532         \tex_kern:D \tex_fontcharwd:D \tex_font:D "4E00 \exp_stop_f:
6533         \tex_hss:D \box_use:N \l__xeCJK_fntef_box \tex_hss:D
6534       }

```

```

6535     }
6536   }
6537 }

```

`_xeCJK_restore_shipout_CJKsymbol:` `\CJKunderdot` 中对 `\CJKsymbol` 的修改会影响到页眉和页脚, 需要小心处理。

```

6538 \cs_new_protected:Npn \_xeCJK_restore_shipout_CJKsymbol:
6539 {
6540   \tl_put_right:Nn \l__xeCJK_fntef_shipout_tl
6541     { \xeCJK_swap_cs:NN \CJKsymbol \_xeCJK_under_CJKsymbol:N }
6542   \_xeCJK_restore_shipout_fntef:
6543   \xeCJK_cs_clear:N \_xeCJK_restore_shipout_CJKsymbol:
6544 }
6545 \cs_new_protected:Npn \_xeCJK_restore_shipout_fntef:
6546 {
6547   \tl_put_right:Nn \l__xeCJK_fntef_shipout_tl
6548     {
6549     \bool_set_false:N \l__xeCJK_fntef_bool
6550     \dim_zero:N \l__xeCJK_fntef_dim
6551     }
6552   \xeCJK_cs_clear:N \_xeCJK_restore_shipout_fntef:
6553 }
6554 \tl_new:N \l__xeCJK_fntef_shipout_tl
6555 \xeCJK_add_to_shipout:n { \l__xeCJK_fntef_shipout_tl }

```

`_xeCJK_under_CJKsymbol:N` 盒子放在汉字的左侧, 比较容易处理状态转移的问题。

```

6556 \cs_new_protected:Npn \_xeCJK_under_CJKsymbol:N
6557 {
6558   \box_use:N \l__xeCJK_under_symbol_box
6559   \xeCJK_no_break: \_xeCJK_under_CJKsymbol:N
6560 }

```

CJKfilltwosides 使用 `minipage` 和 `LATEX` 表格(`tabular`)来定义 `CJKfilltwosides` 环境。可选参数 #1 表示环境的垂直对齐位置, 默认居中; 参数 #2 表示环境的宽度。带星号的环境, 如果 #2 不大于零或者不大于环境最长文本行的宽度, 则取环境的自然宽度。

```

6561 \NewDocumentEnvironment { CJKfilltwosides } { 0 { c } m }
6562 {
6563   \use:e { \exp_not:N \minipage [#1] { \dim_eval:n {#2} } }
6564   \cs_set_eq:NN \CJKglue \xeCJK_fntef_hfilll:
6565 }
6566 {
6567   \endminipage
6568   \ignorespacesafterend
6569 }
6570 \NewDocumentEnvironment { CJKfilltwosides* } { 0 { c } m +b }
6571 {
6572   \mode_leave_vertical:
6573   \cs_set_eq:NN \CJKglue \xeCJK_fntef_hfilll:
6574   \tl_set:Nn \arraystretch { 1 }
6575   \cs_if_free:NF \extrarowheight
6576     { \cs_set_eq:NN \extrarowheight \c_zero_dim }
6577   \use:e { \_xeCJK_fill_two_sides:nnn {#1} { \dim_eval:n {#2} } } {#3}
6578 }
6579 { \ignorespacesafterend }
6580 \cs_new_protected:Npn \_xeCJK_fill_two_sides:nnn #1#2#3
6581 {
6582   \dim_compare:nNnTF {#2} > \c_zero_dim
6583     {
6584       \hbox_set:Nn \l__xeCJK_tmp_box
6585         { \tabular [#1] { @ { } c @ { } } #3 \endtabular }
6586       \dim_compare:nNnTF {#2} > { \box_wd:N \l__xeCJK_tmp_box }
6587         { \tabular [#1] { @ { } p {#2} @ { } } #3 \endtabular }
6588         { \box_use:N \l__xeCJK_tmp_box }
6589     }
6590     { \tabular [#1] { @ { } c @ { } } #3 \endtabular }
6591 }

```

`\xeCJK_fntef_hfilll`: `colortbl` 将表格 `c` 列用于填充的 `\hfil` 改为了更高阶的 `fill`, 影响到了 `CJKfilltwosides*`。因此, 我们也要用高阶的 `filll`。

```
6592 \cs_new_protected:Npn \xeCJK_fntef_hfilll:
6593   { \skip_horizontal:N \c__xeCJK_filll_skip }
6594 \skip_const:Nn \c__xeCJK_filll_skip { \c_zero_dim plus 1 filll }

6595 </fntef>
```

5.20 xeCJK-listings

仿照 `luatexja` 宏包中 `lltjp-listings` 的处理, 支持 `listings` 宏包。

```
6596 <*listings>

6597 \DeclareOption* { \PassOptionsToPackage { \CurrentOption } { xeCJK } }
6598 \ProcessOptions \scan_stop:

6599 \RequirePackage { xeCJK }
6600 \RequirePackage { listings }

6601 \lst@AddToHook { Init } { \__xeCJK_listings_initial_hook: }
6602 \lst@AddToHook { SelectCharTable } { \__xeCJK_listings_toks_hook: }
6603 \lst@AddToHook { OutputBox }
6604   {
6605     \tl_set_eq:NN \l_xeCJK_punct_style_tl \c__xeCJK_punct_style_plain_tl
6606     \l_xeCJK_restore_listings_toks_tl
6607     \__xeCJK_listings_output_CM:
6608   }
6609 \lst@AddToHook { PreSet } { \bool_set_true:N \l__xeCJK_listings_env_bool }
```

`__xeCJK_listings_initial_hook`: 为使代码行号结果正确, 需要在 `\lst@numberstyle` 中恢复 `\XeTeXinterchartoks`。在 `listings` 环境中换页时, 对 `\XeTeXinterchartoks` 的修改会影响到页眉和页脚, 需要在 `\shipout` 盒子中恢复成正常定义。加入 `\tex_noindent:D` 是为了进入水平模式, 防止汉字出现在首行的时候可能会产生额外空行。`\lst@prebreak` 和 `\lst@postbreak` 是在 `\discretionary` 中直接输出的, 应该恢复正常的 `\XeTeXinterchartoks`。

```
6610 \cs_new_protected:Npn \__xeCJK_listings_initial_hook:
6611   {
6612     \tex_noindent:D
6613     \bool_gset_false:N \g__xeCJK_listings_CM_bool
6614     \tl_put_left:Nn \lst@numberstyle { \l__xeCJK_restore_listings_toks_tl }
6615     \xeCJK_add_to_shipout:n { \l__xeCJK_restore_listings_toks_tl }
6616     \lst@ifbreaklines
6617       \cs_set_eq:NN \__xeCJK_listings_CJK_toks_hook: \__xeCJK_listings_breaklines_toks:
6618       \tl_if_empty:NF \lst@prebreak
6619         { \tl_put_left:Nn \lst@prebreak { \l__xeCJK_restore_listings_toks_tl } }
6620       \tl_if_empty:NF \lst@postbreak
6621         { \tl_put_left:Nn \lst@postbreak { \l__xeCJK_restore_listings_toks_tl } }
6622     \fi:
6623     \int_set:Nn \l__xeCJK_listings_max_char_int
6624     { \lst@ifec 255 \else: 127 \fi: }
6625   }
6626 \int_new:N \l__xeCJK_listings_max_char_int
```

`__xeCJK_listings_toks_hook`: 采用不同的 `\XeTeXinterchartoks` 处理方式, 输入的时候是将汉字加入到 `listings` 的输出队列, 实际输出的时候是普通文字。

```
6627 \cs_new_protected:Npn \__xeCJK_listings_toks_hook:
6628   {
6629     \tl_clear:N \l__xeCJK_restore_listings_toks_tl
6630     \seq_map_function:NN
6631       \g__xeCJK_class_seq \__xeCJK_backup_inter_class_toks:n
6632     \seq_map_inline:Nn \g__xeCJK_non_CJK_class_seq
6633     {
6634       \str_if_eq:nnF { ##1 } { Boundary }
6635     }
```

```

6636         \xeCJK_inter_class_toks:nnn { Boundary } { ##1 }
6637         { \__xeCJK_listings_process_Default:nN { ##1 } }
6638     }
6639 }
6640 \xeCJK_inter_class_toks:nnn { Boundary } { CM }
6641 { \__xeCJK_listings_process_CM:nN { 0 } }
6642 \__xeCJK_listings_CJK_toks_hook:
6643 }

```

__xeCJK_backup_inter_class_toks:n 注意,给 \XeTeXinterchartoks 赋空值,会导致 XeTeX 崩溃!

```

6644 \cs_new_protected:Npn \__xeCJK_backup_inter_class_toks:n #1
6645 {
6646   \tl_set:Ne \l__xeCJK_tmp_tl
6647   { \xeCJK_get_inter_class_toks:nn { Boundary } {#1} }
6648   \tl_put_right:Ne \l__xeCJK_restore_listings_toks_tl
6649   {
6650     \xeCJK_inter_class_toks:nnn { Boundary } {#1}
6651     {
6652       \tl_if_empty:NTF \l__xeCJK_tmp_tl
6653       { \exp_not:N \prg_do_nothing: }
6654       { \exp_not:o \l__xeCJK_tmp_tl }
6655     }
6656   }
6657 }
6658 \tl_new:N \l__xeCJK_restore_listings_toks_tl

```

__xeCJK_listings_CJK_toks_hook: 根据 breaklines 选项的使用与否,选择不同的处理方式。

__xeCJK_listings_breaklines_toks:

```

6659 \cs_new_protected:Npn \__xeCJK_listings_CJK_toks_hook:
6660 {
6661   \xeCJK_inter_class_toks:nnn { Boundary } { CJK }
6662   { \__xeCJK_listings_process_CJK:nN { 2 } }
6663   \xeCJK_inter_class_toks:nnn { Boundary } { FullLeft }
6664   { \__xeCJK_listings_process_CJK:nN { 2 } }
6665   \xeCJK_inter_class_toks:nnn { Boundary } { FullRight }
6666   { \__xeCJK_listings_process_CJK:nN { 2 } }
6667   \xeCJK_inter_class_toks:nnn { Boundary } { HangulJamo }
6668   { \__xeCJK_listings_process_CJK:nN { 2 } }
6669   \seq_map_inline:Nn \g__xeCJK_CJK_sub_class_seq
6670   {
6671     \xeCJK_inter_class_toks:nnn { Boundary } { CJK/##1 }
6672     { \__xeCJK_listings_process_CJK:nN { 2 } }
6673   }
6674 }
6675 \cs_new_protected:Npn \__xeCJK_listings_breaklines_toks:
6676 {
6677   \xeCJK_inter_class_toks:nnn { Boundary } { CJK }
6678   { \__xeCJK_listings_process_breaklines_CJK:nN { 2 } }
6679   \xeCJK_inter_class_toks:nnn { Boundary } { HangulJamo }
6680   { \__xeCJK_listings_process_breaklines_CJK:nN { 2 } }
6681   \xeCJK_inter_class_toks:nnn { Boundary } { FullLeft }
6682   { \__xeCJK_listings_process_FullLeft:nN { 2 } }
6683   \xeCJK_inter_class_toks:nnn { Boundary } { FullRight }
6684   { \__xeCJK_listings_process_FullRight:nN { 2 } }
6685   \seq_map_inline:Nn \g__xeCJK_CJK_sub_class_seq
6686   {
6687     \xeCJK_inter_class_toks:nnn { Boundary } { CJK/##1 }
6688     { \__xeCJK_listings_process_breaklines_CJK:nN { 2 } }
6689   }
6690 }

```

__xeCJK_listings_process_Default:nN 对于 \charcode 大于 255 的字符,根据 \catcode 进行处理。

```

6691 \cs_new_protected:Npn \__xeCJK_listings_process_Default:nN #1#2
6692 {
6693   \int_compare:nNnTF
6694   { \xeCJK_token_value_charcode:N #2 } > \l__xeCJK_listings_max_char_int

```

```

6695     {
6696         \token_if_letter:NTF #2
6697         { \lst@ProcessLetter #2 }
6698         { \lst@ProcessOther #2 }
6699     }
6700     { \__xeCJK_listings_output_Default:nN {#1} #2 }
6701 }

```

输出时,要注意把对应的 `\XeTeXinterchartoks` 清空掉,否则会造成死循环。`\scan_stop:` 是造边界,输出 `\group_end:`。

```

6702 \cs_new_protected:Npn \__xeCJK_listings_output_Default:nN #1#2
6703 {
6704     \group_begin:
6705     \xeCJK_clear_inter_class_toks:nn { Boundary } {#1}
6706     \xeCJK_inter_class_toks:nnn {#1} { Boundary } { \group_end: }
6707     #2
6708     \scan_stop:
6709 }

```

`__xeCJK_listings_process_CJK:nN` 对 CJK 字符类的处理。

```

6710 \cs_new_protected:Npn \__xeCJK_listings_process_CJK:nN #1#2
6711 {
6712     \token_if_letter:NTF #2
6713     { \__xeCJK_listings_process_letter:nN {#1} #2 }
6714     { \__xeCJK_listings_process_other:nN {#1} #2 }
6715 }

```

`__xeCJK_listings_append:nN` 普通 CJK 字符的宽度为一般基本宽度的两倍,CM 类不增加宽度。这里有一个问题,对 CJK 字符类中的一些半角字符(例如半角日文假名)没有区分开。`listings` 通过重定义 `\lst@Append` 将代码写入外部文件,因此需要保留。

```

6716 \cs_new_protected:Npn \__xeCJK_listings_append:nN #1#2
6717 {
6718     \int_add:Nn \lst@length { #1 - 1 }
6719     \lst@Append #2
6720 }

```

`__xeCJK_listings_process_letter:nN` 在 letter 类中区分汉字和西文字母。

`__xeCJK_listings_process_other:nN`

```

6721 \cs_new_protected:Npn \__xeCJK_listings_process_letter:nN
6722 {
6723     \lst@whitespacefalse
6724     \bool_if:NTF \l__xeCJK_listings_letter_bool
6725     { \lst@lettertrue }
6726     {
6727         \lst@ifletter \lst@Output \else: \lst@OutputOther \lst@lettertrue \fi:
6728         \bool_set_true:N \l__xeCJK_listings_letter_bool
6729     }
6730     \__xeCJK_listings_append:nN
6731 }
6732 \bool_new:N \l__xeCJK_listings_letter_bool
6733 \cs_new_protected:Npn \__xeCJK_listings_process_other:nN #1#2
6734 {
6735     \lst@whitespacefalse
6736     \bool_if:NTF \l__xeCJK_listings_letter_bool
6737     {
6738         \lst@Output \lst@letterfalse
6739         \bool_set_false:N \l__xeCJK_listings_letter_bool
6740     }
6741     { \lst@ifletter \lst@Output \lst@letterfalse \fi: }
6742     \cs_set_eq:NN \lst@lastother #2
6743     \__xeCJK_listings_append:nN {#1} #2
6744 }

```

当使用 `breaklines` 选项时, 立即输出之前的单个文字, 以便于断行。并将标点与它前/后的 CJK 文字放在同一个盒子中, 以保持禁则。但是不能区分 `letter` 和 `other`。

```

6745 \cs_new_protected:Npn \__xeCJK_listings_process_breaklines_CJK:nN
6746 {
6747   \lst@whitespacefalse
6748   \bool_if:NTF \l__xeCJK_listings_letter_bool
6749   {
6750     \int_compare:nNnF \l__xeCJK_listings_flag_int = 2 { \lst@Output }
6751     \lst@lettertrue
6752   }
6753   {
6754     \lst@ifletter \lst@Output \else: \lst@OutputOther \lst@lettertrue \fi:
6755     \bool_set_true:N \l__xeCJK_listings_letter_bool
6756   }
6757   \int_set_eq:NN \l__xeCJK_listings_flag_int \c_one_int
6758   \__xeCJK_listings_append:nN
6759 }
6760 \cs_new_protected:Npn \__xeCJK_listings_process_FullLeft:nN #1#2
6761 {
6762   \lst@whitespacefalse
6763   \bool_if:NTF \l__xeCJK_listings_letter_bool
6764   {
6765     \int_compare:nNnF \l__xeCJK_listings_flag_int = 2
6766     {
6767       \int_compare:nNnTF \l__xeCJK_listings_flag_int = 3
6768       { \bool_if:NT \l__xeCJK_punct_breakable_bool { \lst@Output } }
6769       { \lst@Output }
6770     }
6771     \lst@lettertrue
6772   }
6773   {
6774     \lst@ifletter \lst@Output \else: \lst@OutputOther \lst@lettertrue \fi:
6775     \bool_set_true:N \l__xeCJK_listings_letter_bool
6776   }
6777   \int_set:Nn \l__xeCJK_listings_flag_int { 2 }
6778   \__xeCJK_listings_append:nN {#1} #2
6779 }
6780 \cs_new_protected:Npn \__xeCJK_listings_process_FullRight:nN #1#2
6781 {
6782   \lst@whitespacefalse
6783   \bool_if:NTF \l__xeCJK_listings_letter_bool
6784   {
6785     \int_compare:nNnT \l__xeCJK_listings_flag_int < 2
6786     { \__xeCJK_punct_if_long:NT #2 { \lst@Output } }
6787     \lst@lettertrue
6788   }
6789   {
6790     \lst@ifletter \lst@Output \else: \lst@OutputOther \lst@lettertrue \fi:
6791     \bool_set_true:N \l__xeCJK_listings_letter_bool
6792   }
6793   \int_set:Nn \l__xeCJK_listings_flag_int { 3 }
6794   \__xeCJK_listings_append:nN {#1} #2
6795 }
6796 \int_new:N \l__xeCJK_listings_flag_int
\lst@AppendLetter 6797 \cs_set_protected:Npn \lst@AppendLetter
\lst@AppendOther 6798 {
6799   \bool_if:NTF \l__xeCJK_listings_letter_bool
6800   {
6801     \lst@Output \lst@lettertrue
6802     \bool_set_false:N \l__xeCJK_listings_letter_bool
6803   }
6804   { \reverse_if:N \lst@ifletter \lst@OutputOther \lst@lettertrue \fi: }
6805   \lst@ifbreaklines \int_zero:N \l__xeCJK_listings_flag_int \fi:
6806   \lst@Append
6807 }
6808 \cs_set_protected:Npn \lst@AppendOther

```

```

6809 {
6810   \bool_if:NTF \l__xeCJK_listings_letter_bool
6811   {
6812     \lst@Output \lst@letterfalse
6813     \bool_set_false:N \l__xeCJK_listings_letter_bool
6814   }
6815   { \lst@ifletter \lst@Output \lst@letterfalse \fi: }
6816   \lst@ifbreaklines \int_zero:N \l__xeCJK_listings_flag_int \fi:
6817   \tex_futurelet:D \lst@lastother \lst@Append
6818 }

```

`_xeCJK_listings_process_CM:nN` CM 类作为 letter 处理, 不用增加 `\lst@length`。

```

6819 \cs_new_protected:Npn \_xeCJK_listings_process_CM:nN
6820 {
6821   \reverse_if:N \lst@ifflexible
6822   \bool_gset_true:N \g__xeCJK_listings_CM_bool
6823   \fi:
6824   \_xeCJK_listings_process_letter:nN
6825 }

```

`_xeCJK_listings_output_CM:` 在使用 `columns=fixed` 选项时, `listings` 会在输出盒子里的每个字符之间加入 `\hss`, 这就破坏了 \XeTeX 将基本字和组合标识正确的组合起来。

```

6826 \cs_new_protected:Npn \_xeCJK_listings_output_CM:
6827 {
6828   \reverse_if:N \lst@ifflexible
6829   \bool_if:NT \g__xeCJK_listings_CM_bool
6830   {
6831     \bool_gset_false:N \g__xeCJK_listings_CM_bool
6832     \xeCJK_cs_clear:N \lst@FillOutputBox
6833     \cs_set_eq:NN \CJKglue \tex_hss:D
6834   }
6835   \fi:
6836 }
6837 \bool_new:N \g__xeCJK_listings_CM_bool

```

`_xeCJK_listings_peek_active_loop:TF` `\lstinline` 通过判断参数中第一个字符是否是 active 类来区分它是否被用在其他宏的参数之中。如果这第一个字符不在 `listings` 预定义的符号表中, 判断就会出问题。我们在这里通过一个循环跳过这些字符。

```

6838 \cs_new_protected:Npn \_xeCJK_listings_peek_active_loop:TF #1#2#3
6839 {
6840   \token_if_active:NTF #3
6841   { #1#3 }
6842   {
6843     \token_if_cs:NTF #3
6844     { #2#3 }
6845     {
6846       \int_compare:nNnTF { `#3 } > { \l__xeCJK_listings_max_char_int }
6847       { \_xeCJK_listings_peek_active_loop:TF { #1#3 } { #2#3 } }
6848       { #2#3 }
6849     }
6850   }
6851 }
6852 \cs_set_eq:NN \lst@ifNextCharActive \_xeCJK_listings_peek_active_loop:TF

```

`_xeCJK_listings_rescan:Nn` 当 `\lstinline` 被使用在参数中时, `listings` 会使用一个循环逐个将 `\lstinline` 参数中的字符设置为活动字符。我们可以通过 `\tl_set_rescan:Nnn` 来完成这里的 `\catcode` 转换, 避免将 `\charcode` 超过 255 的字符都设置为活动字符。

当 `\lstinline` 出现在宏参数中时, \TeX 参数传递将 `catcode 6` 的 `parameter token` (通常是 #) 双写为成对的相邻 `token`。 `\tl_set_rescan:Nnn` 底层的 `\scantokens` 在字符串化阶段会再次将 `catcode 6 token` 双写, 导致输出中数量翻倍。修复方法是在 `rescan` 前逐 `token` 扫描 `\l__xeCJK_tmp_tl`, 将每个 `catcode 6` 的 `token` 替换为保留原字符码的 `active token`

(catcode 13), 这是 listings 期望的字符类型。最初的实现用 `\regex_replace_all:nnN` 把所有 catcode 6 token 硬替换成 U+0023, 对一般 # 没问题, 但当用户用 `\catcode` 把别的字符(如 &) 改为 parameter 时, 原字符码会被丢失, 导致输出错字(#879)。

```

6853 \cs_new_protected:Npn \__xeCJK_listings_rescan:Nn #1#2
6854 {
6855   \__xeCJK_listings_set_escape:
6856   \tl_set:Nn \l__xeCJK_tmp_tl {#2}
6857   \__xeCJK_listings_escape_backslash:
6858   \tl_clear:N \l__xeCJK_tmpb_tl
6859   \tl_map_inline:Nn \l__xeCJK_tmp_tl
6860   {
6861     \token_if_parameter:NTF ##1
6862     {
6863       \tl_put_right:Ne \l__xeCJK_tmpb_tl
6864       {
6865         \exp_after:wN \exp_after:wN \exp_after:wN \exp_not:N
6866         \char_generate:nn { \int_value:w `##1 } { 13 }
6867       }
6868     }
6869     { \tl_put_right:Nn \l__xeCJK_tmpb_tl { ##1 } }
6870   }
6871   \tl_set_eq:NN \l__xeCJK_tmp_tl \l__xeCJK_tmpb_tl
6872   \tl_set_rescan:Nno #1
6873   { \char_set_catcode_active:n { 35 } }
6874   { \l__xeCJK_tmp_tl }
6875 }
6876 \cs_new_protected:Npn \__xeCJK_listings_inside_convert:nw #1 ~ \@empty
6877 {
6878   \__xeCJK_listings_rescan:Nn \l__xeCJK_tmp_tl {#1}
6879   \tl_put_right:No \lst@arg { \l__xeCJK_tmp_tl }
6880 }
6881 \cs_set_eq:NN \lst@InsideConvert@ \__xeCJK_listings_inside_convert:nw
6882 \cs_new_protected:Npn \__xeCJK_listings_inline_group:w
6883 {
6884   \exp_after:wN \__xeCJK_listings_inline_group:n
6885   \exp_after:wN { \if_int_compare:w ` = \c_zero_int \fi:
6886 }
6887 \cs_set_eq:NN \lst@InlineGJ \__xeCJK_listings_inline_group:w
6888 \cs_new_protected:Npn \__xeCJK_listings_inline_group:n #1
6889 {
6890   \__xeCJK_listings_rescan:Nn \lst@arg {#1}
6891   \lst@InlineGJEnd
6892 }

```

`__xeCJK_listings_set_escape:` 由于我们在上面的修改, 需要保留 `\` 用于转义 `\lstinline` 参数中的某些 TeX 特殊字符, 与原来宏包一致。

```

6893 \group_begin:
6894 \cs_set:Npn \__xeCJK_tmp:w #1
6895 {
6896   \group_end:
6897   \cs_new_protected:Npn \__xeCJK_listings_set_escape:
6898   { \xeCJK_swap_cs:NN #1 \__xeCJK_listings_escape:N }
6899   \cs_new_protected:Npn \__xeCJK_listings_escape:N ##1
6900   { \cs_if_eq:NNTF #1 ##1 { \__xeCJK_listings_escape:N } {##1} }
6901 }
6902 \use:n
6903 {
6904   \char_set_catcode_active:N \
6905   \__xeCJK_tmp:w
6906 }
6907 { \ }

```

`__xeCJK_listings_escape_backslash:` `\catcode` 为 12 的 `\` 需要双写转义。

```

6908 \cs_new_protected:Npe \__xeCJK_listings_escape_backslash:

```

```

6909 {
6910   \tl_replace_all:Nnn \exp_not:N \l__xeCJK_tmp_tl
6911     { \c_backslash_str }
6912     { \c_backslash_str \c_backslash_str }
6913 }
6914 </listings>
6915 <@@=xunadd>

```

5.21 xunicode-addon

```
6916 <*xunicode>
```

xunicode 对编码相关的符号命令的定义中用的是诸如 `\char"0022\relax` 的形式。例如 `\textbar` 被展开为 `\char"007C\relax`。并且诸如下述的定义是无效的：

```
\DeclareUTFcomposite[\UTFencname]{x1EBF}{\'}{\^e}
```

我们在这里做的修改是把符号命令定义为实际的字符并且使上述定义生效。另外在使用这些符号命令的时候，先判断当前字体中是否存在对应的字符，如果不存在，则使用这些符号命令的默认设置。

```

6917 \bool_lazy_or:nnF
6918 { \sys_if_engine_xetex_p: }
6919 { \sys_if_engine_luatex_p: }
6920 {
6921   \msg_new:nnnn { xunicode-addon } { xetex-luatex }
6922   { This~package~requires~either~XeTeX~or~LuaTeX~to~function.}
6923   {
6924     You~must~change~your~typesetting~engine~to,~e.g.,\\
6925     "xelatex"~or~"lualatex"~instead~of~plain~"latex"~or~"pdflatex".
6926   }
6927   \msg_critical:nn { xunicode-addon } { xetex-luatex }
6928 }
6929 \RequirePackage { xparse }

```

宏包选项是编码的名字。

```

6930 \clist_new:N \g__xunadd_encname_clist
6931 \tl_if_exist:NT \UTFencname
6932 { \clist_gput_right:Ne \g__xunadd_encname_clist { \UTFencname } }
6933 \DeclareOption*
6934 { \clist_gput_right:No \g__xunadd_encname_clist \CurrentOption }
6935 \ProcessOptions \scan_stop:

```

若 xunicode 已经被调用，则在宏包结束的时候，重新设置 `\UTFencname` 对应的编码命令。否则设置 `\UTFencname`，如果使用的是 Lua^AT_EX，则需要作一些设置，使得 xunicode 可用。

```

6936 \@ifpackageloaded { xunicode } { }
6937 {
6938   \clist_get:NNF \g__xunadd_encname_clist \UTFencname
6939   {
6940     \cs_if_exist:NTF \UnicodeEncodingName
6941     { \tl_set:Ne \UTFencname { \UnicodeEncodingName } }
6942     {
6943       \sys_if_engine_xetex:TF
6944       { \tl_set:Nn \UTFencname { EU1 } }
6945       { \tl_set:Nn \UTFencname { EU2 } }
6946     }
6947     \clist_gset_eq:NN \g__xunadd_encname_clist \UTFencname
6948   }
6949   \sys_if_engine_xetex:TF
6950   { \RequirePackage { xunicode } }
6951   {
6952     \cs_set_eq:NN \__xunadd_tmp:w \XeTeXpicfile
6953     \cs_set_eq:NN \XeTeXpicfile \prg_do_nothing:
6954     \RequirePackage { xunicode }

```

```

6955     \cs_set_eq:NN \XeTeXpicfile \__xunadd_tmp:w
6956   }
6957 }
6958 \AtEndOfPackage { \__xunadd_reload:N \g__xunadd_encname_clist }

```

`\ReloadXunicode` 参数可以是多个编码, 设置这些编码对应的命令。如果编码没有预先声明, 则给出一个错误警告。

```

6959 \RenewDocumentCommand \ReloadXunicode { m }
6960 {
6961   \clist_set:Nc \l__xunadd_encname_clist {#1}
6962   \__xunadd_reload:N \l__xunadd_encname_clist
6963 }
6964 \cs_new_protected:Npn \__xunadd_reload:N #1
6965 {
6966   \cs_set_eq:NN \__xunadd_tmp:w \iftipaonetoken
6967   \cs_set_eq:NN \iftipaonetoken \scan_stop:
6968   \use:e
6969   {
6970     \ExplSyntaxOff
6971     \char_set_catcode_letter:n { 64 }
6972     \exp_not:N \clist_map_function:NN \exp_not:N #1 \__xunadd_reload_aux:n
6973     \bool_if:NTF \l__kernel_expl_bool
6974       { \ExplSyntaxOn }
6975       { \ExplSyntaxOff }
6976     \char_set_catcode:nn { 64 } { \char_value_catcode:n { 64 } }
6977   }
6978   \cs_set_eq:NN \iftipaonetoken \__xunadd_tmp:w
6979 }
6980 \cs_new_protected:Npn \__xunadd_reload_aux:n #1
6981 {
6982   \cs_if_exist:cTF { T@ #1 }
6983   {
6984     \tl_set:Nn \UTFencname {#1}
6985     \clist_if_in:NnF \g__xunadd_encname_clist {#1}
6986     { \clist_gput_right:Nn \g__xunadd_encname_clist {#1} }
6987     \file_input:n { xunicode.sty }
6988     \file_input:n { xunicode-extra.def }
6989   }
6990   { \msg_error:nnn { xunicode-addon } { encoding-unknown } {#1} }
6991 }
6992 \clist_new:N \l__xunadd_encname_clist
6993 \msg_new:nnnn { xunicode-addon } { encoding-unknown }
6994 { Encoding~scheme~"#1"~unknown. }
6995 {
6996   You~may~use \\\
6997   \token_to_str:N \usepackage [ #1 , \encodingdefault ] {\fontenc\} \\\
6998   before~xunicode-addon~or~xunicode.
6999 }

```

`\DeclareUTFmathsymbols` 将文本符号定义为 `\protected` 宏后, 为了与 `hyperref` 的书签功能兼容需要作一点额外处理。

```

7000 \RenewDocumentCommand \DeclareUTFmathsymbols { m }
7001 {
7002   \bool_if:NT \l__xunadd_math_as_UTF_text_bool
7003   {
7004     \seq_map_inline:Nn \l__xunadd_math_as_UTF_text_seq
7005     { \__xunadd_declare_math_as_UTF_text:n {##1} }
7006     \bool_set_false:N \l__xunadd_math_as_UTF_text_bool
7007   }
7008 }
7009 \seq_new:N \l__xunadd_math_as_UTF_text_seq
7010 \seq_set_from_clist:Nn \l__xunadd_math_as_UTF_text_seq
7011 { hbar , Finv , aleph , beth , gimel , dalet , Game }
7012 \bool_new:N \l__xunadd_math_as_UTF_text_bool
7013 \RenewDocumentCommand \UseMathAsText { }
7014 {
7015   \math@s@text@true

```

```

7016   \bool_set_true:N \l__xunadd_math_as_UTF_text_bool
7017 }
7018 \@onlypreamble \UseMathAsText
7019 \cs_new_protected:Npn \__xunadd_declare_math_as_UTF_text:n #1
7020 {
7021   \cs_if_exist:cTF {#1}
7022   {
7023     \cs_new_eq:cc { keepmathUTF #1 } {#1}
7024     \cs_gset_protected:cpe {#1}
7025     {
7026       \exp_not:N \mode_if_math:TF
7027       { \exp_not:c { keepmathUTF #1 } }
7028       { \exp_not:c { text #1 } }
7029     }
7030     \tl_put_right:Ne \l__xunadd_hyperref_hook_tl
7031     { \cs_set_eq:NN \exp_not:c {#1} \exp_not:c { text #1 } }
7032   }
7033   { \cs_new:cpe {#1} { \exp_not:c { text #1 } } }
7034 }
7035 \tl_new:N \l__xunadd_hyperref_hook_tl
7036 \AtBeginDocument
7037 {
7038   \cs_if_free:NF \pdfstringdefDisableCommands
7039   { \pdfstringdefDisableCommands { \l__xunadd_hyperref_hook_tl } }
7040 }

```

判断字符在当前字体中是否存在。

```

\__xunadd_glyph_if_exist_p:n
\__xunadd_glyph_if_exist:nTF
7041 \prg_new_conditional:Npnn \__xunadd_glyph_if_exist:n #1 { p , T , F , TF }
7042 {
7043   \tex_iffontchar:D \tex_font:D \tex_numexpr:D #1 \scan_stop:
7044   \prg_return_true: \else: \prg_return_false: \fi:
7045 }

```

`\UndeclareUTFcharacter` 取消编码 #1 下的符号命令 #3。

```

7046 \RenewDocumentCommand \UndeclareUTFcharacter { 0 { \UTFencname } m m }
7047 {
7048   \__xunadd_if_csname:nTF {#3}
7049   { \UndeclareTextCommand {#3} }
7050   { \exp_args:Nc \UndeclareTextCommand { \tl_to_str:n {#3} } }
7051   {#1}
7052 }

```

`\UndeclareUTFcomposite` 取消编码 #1 下的复合符号命令 #3{#4}。

```

7053 \RenewDocumentCommand \UndeclareUTFcomposite { 0 { \UTFencname } m m m }
7054 {
7055   \__xunadd_if_csname:nTF {#3}
7056   { \__xunadd_undeclare_composite:Nnnn #3 }
7057   { \exp_args:Nc \__xunadd_undeclare_composite:Nnnn { \tl_to_str:n {#3} } }
7058   {#1} {#4} {#2}
7059 }
7060 \cs_new_protected:Npn \__xunadd_undeclare_composite:Nnnn #1#2#3#4
7061 { \cs_undefine:c { \__xunadd_composite_cs:Nnn #1 {#2} {#3} } }

```

```

\__xunadd_composite_cs:Nnn 7062 \cs_new:Npe \__xunadd_composite_cs:Nnn #1#2#3
\__xunadd_composite_cs:nnn 7063 { \c_backslash_str #2 \exp_not:N \token_to_str:N #1 - \exp_not:N \tl_to_str:n {#3} }
7064 \cs_new:Npe \__xunadd_composite_cs:nnn #1#2#3
7065 { \c_backslash_str #2 #1 - \exp_not:N \tl_to_str:n {#3} }

```

`__xunadd_if_csname:nTF` 判断 #1 是否可以作为控制序列的名字。这是因为 `xunicode` 使用了下面的定义。

```

\DeclareUTFcharacter[\UTFencname]{x0149}{\n}
7066 \prg_new_conditional:Npnn \__xunadd_if_csname:n #1 { TF }
7067 {
7068   \tl_if_single_token:nTF {#1}
7069   {
7070     \token_if_cs:NTF #1

```

```

7071     { \prg_return_true: }
7072     {
7073         \token_if_active:NTF #1
7074         { \prg_return_true: }
7075         { \prg_return_false: }
7076     }
7077 }
7078 { \prg_return_false: }
7079 }

```

`\DeclareUTFcharacter` 定义编码 #1 下的符号命令 #3, 其对应符号的 Unicode 是 #2。

```

7080 \RenewDocumentCommand \DeclareUTFcharacter { 0 { \UTFencname } m m }
7081 {
7082     \cs_if_exist_use:cF
7083     { __xunadd_restore_ \tl_to_str:n {#3} : }
7084     {
7085         \__xunadd_if_csname:nTF {#3}
7086         { \__xunadd_declare_character:Nnn #3 }
7087         { \__xunadd_declare_character:cnm { \tl_to_str:n {#3} } }
7088         {#1} {#2}
7089     }
7090 }

```

`__xunadd_restore_cmd:N` 恢复 `\hbar` 和 `\nobreakspace` 为原本定义。

```

7091 \cs_new_protected:cpn
7092 { __xunadd_restore_ \tl_to_str:n { \hbar } : }
7093 { \__xunadd_restore_cmd:N \hbar }
7094 \cs_new_protected:cpn
7095 { __xunadd_restore_ \tl_to_str:n { \nobreakspace } : }
7096 { \__xunadd_restore_cmd:N \nobreakspace }
7097 \cs_new_protected:Npn \__xunadd_restore_cmd:N #1
7098 { \__xunadd_restore_cmd:Ne #1 { ? - \token_to_str:N #1 } }
7099 \cs_new_protected:Npn \__xunadd_restore_cmd:Nn #1#2
7100 {
7101     \cs_if_free:cF {#2}
7102     { \__xunadd_restore_cmd:Nc #1 {#2} }
7103 }
7104 \cs_new_protected:Npn \__xunadd_restore_cmd:NN #1#2
7105 {
7106     \cs_gset_eq:NN #1 #2
7107     \cs_undefine:N #2
7108 }
7109 \cs_generate_variant:Nn \__xunadd_restore_cmd:Nn { Ne }
7110 \cs_generate_variant:Nn \__xunadd_restore_cmd:NN { Nc }

```

`__xunadd_declare_character:Nnn` 通过 `\tex_Uchar:D` 直接由 Unicode #3 得到编码 #2 下的符号命令 #1 对应的实际字符。
`\DeclareUTFSymbol` 的参数格式与 `\DeclareTextSymbol` 完全一致。

```

7111 \cs_new_protected:Npn \__xunadd_declare_character:Nnn #1#2#3
7112 {
7113     \__xunadd_provide_text_command_default:N #1
7114     \exp_after:wN \__xunadd_declare_character:NNen
7115     \tex_Uchar:D \__xunadd_check_slot:n {#3} \exp_stop_f:
7116     #1 { \token_to_str:N #1 } {#2}
7117 }
7118 \cs_generate_variant:Nn \__xunadd_declare_character:Nnn { c }

```

`\DeclareUTFSymbol` `\DeclareUTFCommand` 只能用于定义不带参数的符号命令。

```

\DeclareUTFCommand
7119 \NewDocumentCommand \DeclareUTFSymbol { m 0 { \UTFencname } m }
7120 { \__xunadd_declare_character:Nnn #1 {#2} {#3} }
7121 \NewDocumentCommand \DeclareUTFCommand { m 0 { \UTFencname } m }
7122 { \__xunadd_text_command:Nonn #1 { \token_to_str:N #1 } {#2} {#3} }
7123 \cs_new_protected:Npn \__xunadd_text_command:Nnnn #1#2#3#4
7124 { \DeclareTextCommand #1 {#3} { \__xunadd_text_command:nn {#2} {#4} } }
7125 \cs_generate_variant:Nn \__xunadd_text_command:Nnnn { No }

```

```

7126 \cs_new_protected:Npn \__xunadd_text_command:n #1#2
7127 {
7128   \__xunadd_begin_hook:n #1} {#2}
7129   #2
7130   \__xunadd_end_hook:n #1} {#2}
7131 }

```

`__xunadd_provide_text_command_default:N` 如果控制序列 #1 已经存在,但不是符号命令, `xunicode` 会将它定义为 `\UTFencname` 编码下的符号命令。但是编码被转换之后,再使用这些控制序列, `NFSS` 就会报错。为此需要给出这些符号命令的默认定义,与原来的意义相同。这些命令包括

```

\nobreakspace macro:->\protect \nobreakspace
\copyright      macro:->\protect \copyright
\AA             macro:->\r A
\aa            macro:->\r a
\texttrhookopeno \long macro:->\texttrethookbelow {\textopeno }
\hbar          macro:->{\mathchar '26\mkern -9muh}
\texttaolig    macro:->{a\kern -.25em o}

```

影响比较大的是 `\nobreakspace`、`\copyright` 和 `\hbar`。

```

7132 \cs_new_protected:Npn \__xunadd_provide_text_command_default:N #1
7133 {
7134   \cs_if_exist:cF { ? \token_to_str:N #1 }
7135   {
7136     \cs_if_free:cF { ? - \token_to_str:N #1 }
7137     {
7138       \exp_args:NNv \ProvideTextCommandDefault #1
7139       { ? - \token_to_str:N #1 }
7140     }
7141   }
7142 }

```

`__xunadd_declare_character:NNnn` 使用编码 #4 下的符号命令 #2 的时候先判断它对应的实际字符 #1 在当前字体中是否存在。如果不存在则转换到 `\DeclareTextSymbolDefault` 中设置的编码或者使用 `\DeclareTextCommandDefault` 中设置的命令。

```

7143 \cs_new_protected:Npn \__xunadd_declare_character:NNnn #1#2#3#4
7144 { \DeclareTextCommand #2 {#4} { \__xunadd_text_character:n #3} {#1} } }
7145 \cs_new_protected:Npn \__xunadd_text_character:n #1#2
7146 {
7147   \__xunadd_begin_hook:n #1} {#2}
7148   \__xunadd_glyph_if_exist:nTF { `#2 }
7149   {#2} { \cs_if_exist_use:cF { ? #1 } {#2} }
7150   \__xunadd_end_hook:n #1} {#2}
7151 }
7152 \cs_generate_variant:Nn \__xunadd_declare_character:NNnn { NNe }

```

`__xunadd_check_slot:n` `xunicode` 中使用的 Unicode 格式是诸如 `x0022` 的形式,这就需要一些转换。

```

7153 \cs_new:Npn \__xunadd_check_slot:n #1
7154 {
7155   \int_eval:n
7156   {
7157     \tl_if_head_eq_charcode:nNTF {#1} x
7158     { " \use_none:n #1 } {#1}
7159   }
7160 }

```

`\DeclareUTFcomposite` 设置编码 #1 下的符号命令 #3 与它的参数 #4 的复合对应的符号的 Unicode 是 #2。

```

7161 \RenewDocumentCommand \DeclareUTFcomposite { 0 { \UTFencname } m m m }
7162 {
7163   \__xunadd_if_csname:nTF {#3}
7164   { \__xunadd_declare_composite:Nnn #3 }
7165   { \__xunadd_declare_composite:cnn { \tl_to_str:n {#3} } }
7166   {#1} {#4} {#2}
7167 }

```

`_xunadd_declare_composite:Nnnn` 这里使用 `\tex_afterassignment:D` 是因为 `xunicode` 有如下的定义。

```
\DeclareUTFcomposite[\UTFencname]{x02E8\char"02E5}{\tonebar}{25}
\DeclareUTFcomposite[\UTFencname]{x02E5\char"02E8}{\tonebar}{52}
```

对复合符号命令的定义用的是 `\chardef`, 这有利于下面字符是否存在的判断。

```
7168 \cs_new_protected:Npn \_xunadd_declare_composite:Nnnn #1#2#3#4
7169   {
7170     \tex_afterassignment:D \use_none_delimit_by_q_stop:w
7171     \_xunadd_chardef:cn { \_xunadd_composite_cs:Nnn #1 {#2} {#3} }
7172     { \_xunadd_check_slot:n {#4} }
7173     \q_stop
7174   }
7175 \cs_new_protected:Npn \_xunadd_chardef:Nn #1#2
7176   { \tex_chardef:D #1 = \tex_numexpr:D #2 \scan_stop: }
7177 \cs_generate_variant:Nn \_xunadd_chardef:Nn { c }
7178 \cs_generate_variant:Nn \_xunadd_declare_composite:Nnnn { c }
```

`\DeclareUTFCompositeCommand` 设置编码 #2 下的符号命令 #1 与它的参数 #3 的复合对应结果是 #4。不能直接用 `\DeclareTextCompositeCommand` 来定义, 它与我们的机制冲突。

```
7179 \NewDocumentCommand \DeclareUTFCompositeCommand { m O { \UTFencname } m m }
7180   { \cs_set_protected:cpn { \_xunadd_composite_cs:Nnn #1 {#2} {#3} } {#4} }
```

`\DeclareUTFCompositeSymbol` 设置编码 #2 下的符号命令 #1 与它的参数 #3 的复合对应结果是 #4。不能直接用 `\DeclareTextComposite` 来定义, 它与我们的机制冲突。

```
7181 \NewDocumentCommand \DeclareUTFCompositeSymbol { m O { \UTFencname } m m }
7182   {
7183     \_xunadd_chardef:cn { \_xunadd_composite_cs:Nnn #1 {#2} {#3} }
7184     { \_xunadd_check_slot:n {#4} }
7185   }
```

`\DeclareUTFComposite` 将 #1 设置为编码 #2 下的带一个参数的复合符号命令。

```
7186 \NewDocumentCommand \DeclareUTFComposite { m O { \UTFencname } }
7187   { \use:e { \_xunadd_declare_composite:Nnn \exp_not:N #1 { \token_to_str:N #1 } {#2} } }
```

`\DeclareUTFEncodedAccent` #1 是重音命令, #2 是编码, #3 是组合重音符号的 Unicode, #4 是基本重音符号的 Unicode。当 #1 的参数为空时, 输出 #4, 否则是 #1 的参数与 #3 的组合。

```
7188 \NewDocumentCommand \DeclareUTFEncodedAccent { m O { \UTFencname } m m }
7189   { \_xunadd_declare_encoded:NNnnn \_xunadd_combine_accent:nnNNn #1 {#2} {#3} {#4} }
```

`\DeclareUTFEncodedAccents` #1 是重音命令, #2 是编码, #3 和 #4 都是组合重音符号的 Unicode。输出 #1 与 #3、#4 的组合。

```
7190 \NewDocumentCommand \DeclareUTFEncodedAccents { m O { \UTFencname } m m }
7191   { \_xunadd_declare_encoded:NNnnn \_xunadd_combine_accents:nnNNn #1 {#2} {#3} {#4} }
```

`\DeclareUTFEncodedSymbol` #1 是带参数的符号命令, #2 是编码, #3 是组合符号的 Unicode, #4 是基本符号的 Unicode。当 #1 的参数为空时, 输出 #4, 否则是 #1 的参数与 #3 的组合。

```
7192 \NewDocumentCommand \DeclareUTFEncodedSymbol { m O { \UTFencname } m m }
7193   { \_xunadd_declare_encoded:NNnnn \_xunadd_combine_symbol:nnNNn #1 {#2} {#3} {#4} }
```

`\DeclareUTFEncodedCircle` #1 是带参数的圆圈符号命令, #2 是编码, #3 是组合圆圈符号的 Unicode, #4 是圆圈符号的 Unicode。当 #1 的参数为空时, 输出 #4, 否则是 #1 的参数与 #4 的组合。

```
7194 \NewDocumentCommand \DeclareUTFEncodedCircle { m O { \UTFencname } m m }
7195   { \_xunadd_declare_encoded:NNnnn \_xunadd_combine_circle:nnNNn #1 {#2} {#3} {#4} }
```

`\DeclareEncodedCompositeCharacter` 7196 `\RenewDocumentCommand \DeclareEncodedCompositeCharacter { m m m m }`
7197 `{ \DeclareUTFEncodedSymbol #2 [#1] { "#3 } { "0#4 } }`

`\DeclareEncodedCompositeAccents` 7198 `\RenewDocumentCommand \DeclareEncodedCompositeAccents { m m m m }`
7199 `{ \DeclareUTFEncodedAccents #2 [#1] { "#4 } { "#3 } }`

```

\DeclareUTFDoubleEncodedAccent 7200 \NewDocumentCommand \DeclareUTFDoubleEncodedAccent { m O { \UTFencname } m m }
7201 { \__xunadd_declare_encoded:NNnnn \__xunadd_combine_double_accent:nnNNn #1 {#2} {#3} {#4} }

\DeclareUTFDoubleEncodedSymbol 7202 \NewDocumentCommand \DeclareUTFDoubleEncodedSymbol { m O { \UTFencname } m m }
7203 { \__xunadd_declare_encoded:NNnnn \__xunadd_combine_double_symbol:nnNNn #1 {#2} {#3} {#4} }

```

__xunadd_declare_composite:Nnn 通过 lowercase 技巧, 直接由重音符号的 Unicode 得到实际字符。

```

7204 \cs_new_protected:Npn \__xunadd_declare_composite:Nnn #1#2#3
7205 { \DeclareTextCommand #1 {#3} { \__xunadd_text_composite:nnn {#2} {#3} } }

\__xunadd_text_composite:nnn 7206 \cs_new_protected:Npn \__xunadd_text_composite:nnn #1#2#3
7207 {
7208   \__xunadd_begin_hook:nn {#1} {#3}
7209   \cs_if_exist:cTF { \__xunadd_composite_cs:nnn {#1} {#2} {#3} }
7210   {
7211     \__xunadd_text_composite:cnn
7212     { \__xunadd_composite_cs:nnn {#1} {#2} {#3} } {#1} {#3}
7213   }
7214   { \cs_if_exist_use:cTF { ? #1 } { {#3} } {#3} }
7215   \__xunadd_end_hook:nn {#1} {#3}
7216 }
7217 \cs_new_protected:Npn \__xunadd_text_composite:Nnn #1#2#3
7218 {
7219   \token_if_chardef:NTF #1
7220   {
7221     \__xunadd_glyph_if_exist:nTF {#1}
7222     {#1} { \cs_if_exist_use:cTF { ? #2 } { {#3} } {#3} }
7223   }
7224   {#1}
7225 }
7226 \cs_generate_variant:Nn \__xunadd_text_composite:Nnn { c }

```

__xunadd_declare_encoded:NNnnn 通过 \tex_Uchar:D 直接由重音符号的 Unicode 得到实际字符。

```

7227 \cs_new_protected:Npn \__xunadd_declare_encoded:NNnnn #1#2#3#4#5
7228 {
7229   \exp_after:wN \__xunadd_declare_encoded:NNNnee
7230   \tex_Uchar:D \__xunadd_check_slot:n {#4} \exp_after:wN \exp_stop_f:
7231   \tex_Uchar:D \__xunadd_check_slot:n {#5} \exp_stop_f:
7232   #1 #2 { \token_to_str:N #2 } {#3}
7233 }
7234 \cs_new_protected:Npn \__xunadd_declare_encoded:NNNNnn #1#2#3#4#5#6
7235 { \DeclareTextCommand #4 {#6} { #3 {#5} {#6} {#1} {#2} } }
7236 \cs_generate_variant:Nn \__xunadd_declare_encoded:NNnnn { c }
7237 \cs_generate_variant:Nn \__xunadd_declare_encoded:NNNNnn { NNNnee }

```

__xunadd_text_combine:NnnNNn 若重音命令 #2 与它的参数 #6 的复合已经由 \DeclareUTFcomposite 设置, 并且在当前字体中存在该字符, 则直接使用。否则使用组合命令。

```

7238 \cs_new_protected:Npn \__xunadd_text_combine:NnnNNn #1#2#3#4#5#6
7239 {
7240   \__xunadd_begin_hook:nn {#2} {#6}
7241   \cs_if_exist:cTF { \__xunadd_composite_cs:nnn {#2} {#3} {#6} }
7242   {
7243     \__xunadd_text_combine:cNnNNn
7244     { \__xunadd_composite_cs:nnn {#2} {#3} {#6} } #1 {#2} {#4} {#5} {#6}
7245   }
7246   { #1 {#6} {#2} {#4} {#5} }
7247   \__xunadd_end_hook:nn {#2} {#6}
7248 }
7249 \cs_new_protected:Npn \__xunadd_text_combine:NNnNNn #1#2#3#4#5#6
7250 {
7251   \token_if_chardef:NTF #1
7252   { \__xunadd_glyph_if_exist:nTF {#1} {#1} { #2 {#6} {#3} {#4} {#5} } }
7253   {#1}
7254 }
7255 \cs_generate_variant:Nn \__xunadd_text_combine:NNnNNn { c }

```

```

7256 \cs_new_protected:Npn \__xunadd_combine_symbol:nnNNn
7257   { \__xunadd_text_combine:NnnNNn \__xunadd_add_symbol:nnNN }
7258 \cs_new_protected:Npn \__xunadd_add_symbol:nnNN #1#2#3#4
7259   {
7260     \tl_if_blank:nTF {#1}
7261     {
7262       \__xunadd_glyph_if_exist:nTF {`#4 }
7263       {#4}
7264       { \cs_if_exist_use:cTF { ? #2 } { {#1} } {#4} }
7265     }
7266     {
7267       \__xunadd_glyph_if_exist:nTF {`#3 }
7268       { #1#3 }
7269       { \cs_if_exist_use:cTF { ? #2 } { {#1} } { #1#3 } }
7270     }
7271   }

```

若组合重音符号的 #3 和基本重音符号 #4 在当前字体中都不存在，则转换到 `\DeclareTextAccentDefault` 设置的编码或者使用 `\DeclareTextCommandDefault` 中设置的命令。0.9999 版以前的 XeTeX 需要设置 `\XeTeXinputnormalization` 为 1，才能使用字体中由基础字符和组合符号对应的实际字符；而 0.9999 版以后的 XeTeX 默认就启用这个功能，`\XeTeXinputnormalization` 似乎是无效的，怀疑是使用 HarfBuzz 库替代 ICU 进行字体排版的缘故¹⁹。

```

7272 \cs_new_protected:Npn \__xunadd_combine_accent:nnNNn
7273   { \__xunadd_text_combine:NnnNNn \__xunadd_add_accent:nnNN }
7274 \cs_new_protected:Npn \__xunadd_add_accent:nnNN #1#2#3#4
7275   {
7276     \tl_if_blank:nTF {#1}
7277     {
7278       \__xunadd_glyph_if_exist:nTF {`#4 }
7279       {#4}
7280       { \cs_if_exist_use:cTF { ? #2 } { {#1} } {#4} }
7281     }
7282     {
7283       \__xunadd_glyph_if_exist:nTF {`#3 }
7284       { #1#3 }
7285       {
7286         \__xunadd_glyph_if_exist:nTF {`#4 }
7287         { \add@accent {`#4 } {#1} }
7288         { \cs_if_exist_use:cTF { ? #2 } { {#1} } { #1#3 } }
7289       }
7290     }
7291   }

```

```

7292 \cs_new_protected:Npn \__xunadd_combine_accents:nnNNn
7293   { \__xunadd_text_combine:NnnNNn \__xunadd_add_accents:nnNN }
7294 \cs_new_protected:Npn \__xunadd_add_accents:nnNN #1#2#3#4
7295   {
7296     \tl_if_blank:nTF {#1}
7297     { \cs_if_exist_use:cTF { ? #2 } { {#1} } {#1} }
7298     {
7299       \__xunadd_glyph_if_exist:nTF {`#3 }
7300       { \__xunadd_glyph_if_exist:nTF {`#4 } }
7301       { \use_ii:nn }
7302       { #1#3#4 }
7303       { \cs_if_exist_use:cTF { ? #2 } { {#1} } { #1#3#4 } }
7304     }
7305   }

```

对圆圈中的数字或者字母适当缩小，以适合圆圈的大小。只有字体中存在 U+25EF 时，才使用这里的设置，否则还还是 L^AT_EX 中的设置。

```

7306 \cs_new_protected:Npn \__xunadd_combine_circle:nnNNn

```

¹⁹<http://tug.org/pipermail/xetex/2013-July/024579.html>

```

7307 { \__xunadd_text_combine:NnnNNn \__xunadd_add_circle:nnNN }
7308 \cs_new_protected:Npn \__xunadd_add_circle:nnNN #1#2#3#4
7309 {
7310   \tl_if_blank:nTF {#1}
7311   {
7312     \__xunadd_glyph_if_exist:nTF { `#4 }
7313     {#4}
7314     { \cs_if_exist_use:cTF { ? #2 } { {#1} } {#4} }
7315   }
7316   {
7317     \__xunadd_glyph_if_exist:nTF { `#4 }
7318     { \__xunadd_add_circle:nN {#1} #4 }
7319     { \cs_if_exist_use:cTF { ? #2 } { {#1} } {#1} }
7320   }
7321 }
7322 \cs_new_protected:Npn \__xunadd_add_circle:nN #1#2
7323 {
7324   \hcoffin_set:Nn \l__xunadd_tmp_coffin {#1}
7325   \hcoffin_set:Nn \l__xunadd_circle_coffin {#2}
7326   \fp_set:Nn \l__xunadd_circle_scale_fp
7327   {
7328     \dim_to_decimal_in_unit:nn
7329     {
7330       \fp_use:N \l__xunadd_circle_ratio_fp
7331       \coffin_wd:N \l__xunadd_circle_coffin
7332     }
7333     { \coffin_wd:N \l__xunadd_tmp_coffin }
7334   }
7335   \coffin_scale:Nnn \l__xunadd_tmp_coffin
7336   { \l__xunadd_circle_scale_fp } { \l__xunadd_circle_scale_fp }
7337   \coffin_attach:NnnNnnnn
7338   \l__xunadd_circle_coffin { hc } { vc }
7339   \l__xunadd_tmp_coffin { hc } { vc } { \c_zero_dim } { \c_zero_dim }
7340   \coffin_typeset:Nnnnn \l__xunadd_circle_coffin
7341   { H } { l } { \c_zero_dim } { \c_zero_dim }
7342 }
7343 \fp_new:N \l__xunadd_circle_scale_fp
7344 \coffin_new:N \l__xunadd_tmp_coffin
7345 \coffin_new:N \l__xunadd_circle_coffin

```

`\settextcircledratio` 设置圆圈中文字的宽度与圆圈宽度的比例, 预设为 0.7。

```

7346 \NewDocumentCommand \settextcircledratio { m }
7347 { \fp_set:Nn \l__xunadd_circle_ratio_fp {#1} }
7348 \fp_new:N \l__xunadd_circle_ratio_fp
7349 \settextcircledratio { 0.7 }

```

`__xunadd_combine_double_accent:nnNNn` 使 `\t` 等组合重音符号放在参数的第一个字母的右边。

```

7350 \cs_new_protected:Npn \__xunadd_combine_double_accent:nnNNn
7351 { \__xunadd_text_combine:NnnNNn \__xunadd_add_double_accent:nnNN }
7352 \cs_new_protected:Npn \__xunadd_add_double_accent:nnNN #1#2#3#4
7353 {
7354   \tl_if_blank:nTF {#1}
7355   {
7356     \__xunadd_glyph_if_exist:nTF { `#4 }
7357     {#4}
7358     { \cs_if_exist_use:cTF { ? #2 } { {#1} } {#4} }
7359   }
7360   {
7361     \__xunadd_glyph_if_exist:nTF { `#3 }
7362     { \__xunadd_add_double_symbol:nN {#1} #3 }
7363     {
7364       \__xunadd_glyph_if_exist:nTF { `#4 }
7365       { \add@accent { `#4 } {#1} }
7366       { \cs_if_exist_use:cTF { ? #2 } { {#1} } { #1#3 } }
7367     }
7368   }

```

7369 }

使 \sliding 等组合重音符号放在参数的第一个字母的右边。

```

7370 \cs_new_protected:Npn \__xunadd_combine_double_symbol:nnNNn
7371 { \__xunadd_text_combine:NnnNNn \__xunadd_add_double_symbol:nnNN }
7372 \cs_new_protected:Npn \__xunadd_add_double_symbol:nnNN #1#2#3#4
7373 {
7374   \tl_if_blank:nTF {#1}
7375   {
7376     \__xunadd_glyph_if_exist:nTF {`#4 }
7377     {#4}
7378     { \cs_if_exist_use:cTF { ? #2 } { {#1} } {#4} }
7379   }
7380   {
7381     \__xunadd_glyph_if_exist:nTF {`#3 }
7382     { \__xunadd_add_double_symbol:nN {#1} #3 }
7383     { \cs_if_exist_use:cTF { ? #2 } { {#1} } { #1#3 } }
7384   }
7385 }
```

如果参数的第一个记号是字母类、其他符号类或者由 \chardef 定义, 则将组合符号放在它的右边, 否则不作处理。

```

7386 \cs_new_protected:Npn \__xunadd_add_double_symbol:nN #1#2
7387 {
7388   \tl_if_head_is_N_type:nTF {#1}
7389   {
7390     \exp_after:wN \exp_after:wN \exp_after:wN
7391     \__xunadd_add_double_symbol_aux:NnN \exp_after:wN \exp_after:wN
7392     \tl_head:w #1 \q_stop \exp_after:wN { \use_none:n #1 } #2
7393   }
7394   { #1#2 }
7395 }
7396 \cs_new_protected:Npn \__xunadd_add_double_symbol_aux:NnN #1#2#3
7397 {
7398   \bool_lazy_any:nTF
7399   {
7400     { \token_if_letter_p:N #1 }
7401     { \token_if_other_p:N #1 }
7402     { \token_if_chardef_p:N #1 }
7403   }
7404   { #1#3#2 }
7405   { #1#2#3 }
7406 }
```

设置在符号命令前后使用的钩子, 可选参数用于指定单个符号命名。可以用 #1 引用带参数的组合符号命令的参数或者符号命令对应的符号。

```

7407 \NewDocumentCommand \AtBeginUTFCommand { s O { } +m }
7408 {
7409   \tl_if_blank:nTF {#2}
7410   {
7411     \IfBooleanTF {#1}
7412     { \xunadd_set_begin_hook:n }
7413     { \xunadd_append_begin_hook:n }
7414   }
7415   { \xunadd_set_begin_hook:nn {#2} }
7416   {#3}
7417 }
7418 \NewDocumentCommand \AtEndUTFCommand { s O { } +m }
7419 {
7420   \tl_if_blank:nTF {#2}
7421   {
7422     \IfBooleanTF {#1}
7423     { \xunadd_set_end_hook:n }
7424     { \xunadd_append_end_hook:n }
7425   }
}
```

```

7426     { \xunadd_set_end_hook:nn {#2} }
7427     {#3}
7428   }

\xunadd_set_begin_hook:n 7429 \cs_new_protected:Npn \xunadd_set_begin_hook:n
\xunadd_set_end_hook:n 7430 { \tl_set:Nn \l__xunadd_begin_hook_tl }
7431 \cs_new_protected:Npn \xunadd_append_begin_hook:n
7432 { \tl_put_right:Nn \l__xunadd_begin_hook_tl }
7433 \cs_new_protected:Npn \xunadd_set_end_hook:n
7434 { \tl_set:Nn \l__xunadd_end_hook_tl }
7435 \cs_new_protected:Npn \xunadd_append_end_hook:n
7436 { \tl_put_right:Nn \l__xunadd_end_hook_tl }
7437 \cs_new_protected:Npn \xunadd_set_begin_hook:nn
7438 { \__xunadd_set_cmd_hook:nnn { begin } }
7439 \cs_new_protected:Npn \xunadd_set_end_hook:nn
7440 { \__xunadd_set_cmd_hook:nnn { end } }
7441 \cs_new_protected:Npn \__xunadd_set_cmd_hook:nnn #1#2#3
7442 {
7443   \cs_set_protected:cpn
7444   {
7445     \tl_if_single:nTF {#2}
7446     { \use:c { __xunadd_#1_csname:n } { \token_to_str:N #2 } }
7447     { \__xunadd_set_cmd_hook_aux:Nwn #2 \q_stop {#1} }
7448   } ##1
7449   {#3}
7450 }
7451 \cs_new:Npn \__xunadd_set_cmd_hook_aux:Nwn #1#2 \q_stop #3
7452 { \use:c { __xunadd_#3_csname:n } { \token_to_str:N #1 - \tl_to_str:n {#2} } }
7453 \cs_new:Npn \__xunadd_begin_csname:n #1 { __xunadd_begin_#1_hook:n }
7454 \cs_new:Npn \__xunadd_end_csname:n #1 { __xunadd_end_#1_hook:n }
7455 \tl_new:N \l__xunadd_begin_hook_tl
7456 \tl_new:N \l__xunadd_end_hook_tl

__xunadd_begin_hook:nn 7457 \cs_new_protected:Npn \__xunadd_begin_hook:nn #1#2
__xunadd_end_hook:nn 7458 {
7459   \tl_use:N \l__xunadd_begin_hook_tl
7460   \cs_if_exist_use:cF { \__xunadd_begin_csname:n { #1 - \tl_to_str:n {#2} } }
7461   { \cs_if_exist_use:cF { \__xunadd_begin_csname:n {#1} } { \use_none:n } }
7462   {#2}
7463 }
7464 \cs_new_protected:Npn \__xunadd_end_hook:nn #1#2
7465 {
7466   \cs_if_exist_use:cF { \__xunadd_end_csname:n { #1 - \tl_to_str:n {#2} } }
7467   { \cs_if_exist_use:cF { \__xunadd_end_csname:n {#1} } { \use_none:n } }
7468   {#2}
7469   \tl_use:N \l__xunadd_end_hook_tl
7470 }

\DeclareUTFFTIPACCommand 7471 \NewDocumentCommand \DeclareUTFFTIPACCommand { 0 { \UTFencname } m }
7472 { \use:e { \__xunadd_text_tipa_command:Nnn \exp_not:N #2 { \token_to_str:N #2 } {#1} } }
7473 \cs_new_protected:Npn \__xunadd_text_tipa_command:Nnn #1#2#3
7474 {
7475   \cs_set_eq:cc { UTF/#3#2 } { #3#2 }
7476   \DeclareTextCommand #1 {#3} { \__xunadd_text_tipa_command:nnn {#3} {#2} }
7477 }
7478 \cs_new_protected:Npn \__xunadd_text_tipa_command:nnn #1#2#3
7479 {
7480   \exp_args:Ncc \__xunadd_check_for_tipa:NNn
7481   { \use_none:n #2 } { UTF/#1#2 } {#3}
7482 }
7483 \cs_new_protected:Npn \__xunadd_check_for_tipa:NNn #1#2#3
7484 {
7485   \tl_if_head_eq_meaning:nNTF {#3} \textipa
7486   {
7487     \exp_after:wN \tipacatchonechar \exp_after:wN
7488     { \exp_after:wN #1 \use_none:n #3 }
7489   }
7490   { #2 {#3} }

```

```
7491 }
```

`\xunadd_get_slot:n` #1 是编码, #2 是诸如 `\textendash` 或 `\v C` 等形式的文本命令, 取得他们对应的字符编码。

```
7492 \cs_new_protected:Npn \xunadd_get_slot:n #1#2
7493 { \__xunadd_get_slot:wn #2 \q_nil \q_stop {#1} }
7494 \cs_new_protected:Npn \__xunadd_get_slot:wn #1#2#3 \q_stop #4
7495 {
7496   \int_set:Nn \l_xunadd_slot_int { -1 }
7497   \bool_set_false:N \l_xunadd_rest_bool
7498   \group_begin: \exp_args:Nccc \group_end:
7499   { \__xunadd_get_slot:NNnn }
7500   { #4 \token_to_str:N #1 }
7501   { \__xunadd_composite_cs:Nnn #1 {#4} {#2} }
7502   {#2}
7503   {#3}
7504 }
7505 \int_new:N \l_xunadd_slot_int
7506 \bool_new:N \l_xunadd_rest_bool
7507 \cs_new_protected:Npn \__xunadd_get_slot:NNnn #1#2#3#4
7508 {
7509   \cs_if_free:NF #1
7510   {
7511     \cs_if_exist:NTF #2
7512     { \__xunadd_get_composite_slot:Nn #2 {#4} }
7513     { \__xunadd_get_character_slot:Nn #1 { #3 #4 } }
7514   }
7515 }
7516 \cs_new_protected:Npn \__xunadd_get_composite_slot:Nn #1#2
7517 {
7518   \token_if_chardef:NT #1
7519   {
7520     \int_set:Nn \l_xunadd_slot_int {#1}
7521     \quark_if_nil:nF {#2}
7522     { \bool_set_true:N \l_xunadd_rest_bool }
7523   }
7524 }
7525 \cs_new_protected:Npn \__xunadd_get_character_slot:Nn #1
7526 {
7527   \exp_after:wN \__xunadd_get_character_slot_aux:wn #1
7528   \__xunadd_text_character:nN \q_nil \q_nil \q_stop
7529 }
7530 \cs_new_protected:Npn \__xunadd_get_character_slot_aux:wn
7531 #1 \__xunadd_text_character:nN #2#3#4 \q_stop #5
7532 {
7533   \quark_if_nil:nF {#2}
7534   {
7535     \int_set:Nn \l_xunadd_slot_int { `#3 }
7536     \quark_if_nil:nF {#5}
7537     { \bool_set_true:N \l_xunadd_rest_bool }
7538   }
7539 }
```

`\xunadd@microtype@is@charx` `microtype` 宏包中使用的函数, 我们通过对 `\MT@is@charx` 打补丁来实现功能。

```
7540 \cs_new_protected_nopar:Npn \xunadd@microtype@is@charx #1 \relax
7541 {
7542   \use:e
7543   { \xunadd_get_slot:n { \MT@encoding } { \tex_the:D \MT@toks } }
7544   \int_compare:nNnTF \l_xunadd_slot_int < \c_zero_int
7545   { \xunadd@original@is@charx #1 \relax }
7546   {
7547     \cs_set_nopar:Npe \MT@char@ { \int_use:N \l_xunadd_slot_int }
7548     \bool_if:NT \l_xunadd_rest_bool { \MT@norestfalse }
7549   }
7550 }
7551 \cs_new_protected_nopar:Npn \xunadd@microtype@hook
```

```

7552 {
7553   \cs_if_free:NF \MT@is@charx
7554   {
7555     \cs_new_eq:NN \xunadd@original@is@charx \MT@is@charx
7556     \cs_set_eq:NN \MT@is@charx \xunadd@microtype@is@charx
7557     \cs_set_eq:NN \MT@warn@unknown@once \use_none:n
7558   }
7559 }
7560 \@ifpackageloaded { microtype }
7561 { \use:n } { \AtBeginDocument }
7562 { \xunadd@microtype@hook }

7563 </xunicode>
7564 <*xunextra>

```

我们补充定义 HYPHENATION POINT 和 TWO-EM DASH, 他们默认被归入 CJK 标点符号。

```

7565 \DeclareUTFSymbol\textthyphenationpoint{"2027}
7566 \DeclareUTFSymbol\texttwoemdash{"2E3A}

```

以下内容选自 xunicode, 并做了适当修改。

```

7567 \DeclareUTFComposite\textsuperscript
7568 \DeclareUTFComposite\textsubscript
7569 \DeclareUTFEncodedAccent\textsbleftarrow{"20EE}{02FF}
7570 \DeclareUTFEncodedAccent\`{"0300}{02CB}
7571 \DeclareUTFEncodedAccent\capitalgrave{"0300}{02CB}
7572 \DeclareUTFEncodedAccent\`{"0301}{02CA}
7573 \DeclareUTFEncodedAccent\capitalacute{"0301}{02CA}
7574 \DeclareUTFEncodedAccent\`{"0302}{02C6}
7575 \DeclareUTFEncodedAccent\capitalcircumflex{"0302}{02C6}
7576 \DeclareUTFEncodedAccent\~{"0303}{02DC}
7577 \DeclareUTFEncodedAccent\capitaltilde{"0303}{02DC}
7578 \DeclareUTFEncodedAccent\={"0304}{02C9}
7579 \DeclareUTFEncodedAccent\capitalmacron{"0304}{02C9}
7580 \DeclareUTFEncodedAccent\textoverline{"0305}{0203E}
7581 \DeclareUTFEncodedAccent\u{"0306}{02D8}
7582 \DeclareUTFEncodedAccent\capitalbreve{"0306}{02D8}
7583 \DeclareUTFEncodedAccent\.`{"0307}{02D9}
7584 \DeclareUTFEncodedAccent\capitaldotaccent{"0307}{02D9}
7585 \DeclareUTFEncodedAccent\`{"0308}{00A8}
7586 \DeclareUTFEncodedAccent\capitaldieresis{"0308}{00A8}
7587 \DeclareUTFEncodedAccent\m{"0309}{0309}
7588 \DeclareUTFEncodedAccent\texthookabove{"0309}{0309}
7589 \DeclareUTFEncodedAccent\r{"030A}{02DA}
7590 \DeclareUTFEncodedAccent\capitalring{"030A}{02DA}
7591 \DeclareUTFEncodedAccent\H{"030B}{02DD}
7592 \DeclareUTFEncodedAccent\capitalhungarumlaut{"030B}{02DD}
7593 \DeclareUTFEncodedAccent\v{"030C}{02C7}
7594 \DeclareUTFEncodedAccent\capitalcaron{"030C}{02C7}
7595 \DeclareUTFEncodedAccent\textvbaraccent{"030D}{02C8}
7596 \DeclareUTFEncodedAccent\textdoublevbaraccent{"030E}{0022}
7597 \DeclareUTFEncodedAccent\U{"030E}{0022}
7598 \DeclareUTFEncodedAccent\textdoublegrave{"030F}{02F5}
7599 \DeclareUTFEncodedAccent\G{"030F}{02F5}
7600 \DeclareUTFEncodedAccent\textdotbreve{"0310}{0310}
7601 \DeclareUTFEncodedAccent\textroundcap{"0311}{0311}
7602 \DeclareUTFEncodedAccent\newtie{"0311}{0311}
7603 \DeclareUTFEncodedAccent\capitalnewtie{"0311}{0311}
7604 \DeclareUTFEncodedAccent\textturncommaabove{"0312}{02BB}
7605 \DeclareUTFEncodedAccent\textcommaabove{"0313}{02BC}
7606 \DeclareUTFEncodedAccent\textrevcommaabove{"0314}{02BD}
7607 \DeclareUTFEncodedAccent\overbridge{"0346}{0346}
7608 \DeclareUTFEncodedAccent\crtilde{"034A}{034A}
7609 \DeclareUTFEncodedAccent\dottedtilde{"034B}{034B}
7610 \DeclareUTFEncodedAccent\doubletilde{"034C}{034C}
7611 \DeclareUTFEncodedAccent\textrightarrowhead{"0350}{02C3}

```

```

7612 \DeclareUTFEncodedAccent\textlefthalfring{"0351"}{"02D3}
7613 \DeclareUTFEncodedAccent\textriighthalfring{"0357"}{"02D2}
7614 \DeclareUTFDoubleEncodedSymbol\textdoublebrevebelow{"035C"}{"035C}
7615 \DeclareUTFDoubleEncodedAccent\textdoublebreve{"035D"}{"035D}
7616 \DeclareUTFDoubleEncodedAccent\textdoublemacron{"035E"}{"035E}
7617 \DeclareUTFDoubleEncodedSymbol\textdoublemacronbelow{"035F"}{"035F}
7618 \DeclareUTFDoubleEncodedAccent\textdoubletilde{"0360"}{"0360}
7619 \DeclareUTFDoubleEncodedAccent\t{"0361"}{"0361}
7620 \DeclareUTFDoubleEncodedAccent\capitaltie{"0361"}{"0361}
7621 \DeclareUTFDoubleEncodedAccent\texttoptiebar{"0361"}{"0361}
7622 \DeclareUTFDoubleEncodedSymbol\sliding{"0362"}{"0362}
7623 \DeclareUTFTIPACCommand\t
7624 \DeclareUTFTIPACCommand\capitaltie
7625 \DeclareUTFTIPACCommand\texttoptiebar
7626 \DeclareUTFTIPACCommand\sliding
7627 \DeclareUTFEncodedAccent\texthighrise{"1DC4"}{"1DC4}
7628 \DeclareUTFEncodedAccent\textlowrise{"1DC5"}{"1DC5}
7629 \DeclareUTFEncodedAccent\textrisefall{"1DC8"}{"1DC8}
7630 \DeclareUTFEncodedAccent\textfallrise{"1DC9"}{"1DC9}
7631 \DeclareUTFEncodedAccent\texttaolig{"1DD5"}{"1DD5}
7632 \DeclareUTFCompositeSymbol\textundertie{H}{"1E2A}
7633 \DeclareUTFCompositeSymbol\textundertie{h}{"1E2B}
7634 \DeclareUTFEncodedAccents\textcircumgrave{"0302"}{"0301}
7635 \DeclareUTFSymbol\textFinv{"2132}
7636 \DeclareUTFSymbol\textaleph{"2135}
7637 \DeclareUTFSymbol\textbeth{"2136}
7638 \DeclareUTFSymbol\textgimel{"2137}
7639 \DeclareUTFSymbol\textdaleth{"2138}
7640 \DeclareUTFSymbol\textGame{"2141}
7641 \DeclareUTFCompositeCommand\tonebar{25}{\tonebar{2}\tonebar{5}}
7642 \DeclareUTFCompositeCommand\tonebar{52}{\tonebar{5}\tonebar{2}}
7643 \DeclareUTFSymbol\textbigcircle{"25EF}
7644 \DeclareUTFEncodedCircle\textcircled{"20DD"}{"25EF}
7645 \DeclareUTFCompositeSymbol\textcircled{0}{"24EA}
7646 \DeclareUTFCompositeSymbol\textcircled{1}{"2460}
7647 \DeclareUTFCompositeSymbol\textcircled{2}{"2461}
7648 \DeclareUTFCompositeSymbol\textcircled{3}{"2462}
7649 \DeclareUTFCompositeSymbol\textcircled{4}{"2463}
7650 \DeclareUTFCompositeSymbol\textcircled{5}{"2464}
7651 \DeclareUTFCompositeSymbol\textcircled{6}{"2465}
7652 \DeclareUTFCompositeSymbol\textcircled{7}{"2466}
7653 \DeclareUTFCompositeSymbol\textcircled{8}{"2467}
7654 \DeclareUTFCompositeSymbol\textcircled{9}{"2468}
7655 \DeclareUTFCompositeSymbol\textcircled{10}{"2469}
7656 \DeclareUTFCompositeSymbol\textcircled{11}{"246A}
7657 \DeclareUTFCompositeSymbol\textcircled{12}{"246B}
7658 \DeclareUTFCompositeSymbol\textcircled{13}{"246C}
7659 \DeclareUTFCompositeSymbol\textcircled{14}{"246D}
7660 \DeclareUTFCompositeSymbol\textcircled{15}{"246E}
7661 \DeclareUTFCompositeSymbol\textcircled{16}{"246F}
7662 \DeclareUTFCompositeSymbol\textcircled{17}{"2470}
7663 \DeclareUTFCompositeSymbol\textcircled{18}{"2471}
7664 \DeclareUTFCompositeSymbol\textcircled{19}{"2472}
7665 \DeclareUTFCompositeSymbol\textcircled{20}{"2473}
7666 \DeclareUTFCompositeSymbol\textcircled{21}{"3251}
7667 \DeclareUTFCompositeSymbol\textcircled{22}{"3252}
7668 \DeclareUTFCompositeSymbol\textcircled{23}{"3253}
7669 \DeclareUTFCompositeSymbol\textcircled{24}{"3254}
7670 \DeclareUTFCompositeSymbol\textcircled{25}{"3255}
7671 \DeclareUTFCompositeSymbol\textcircled{26}{"3256}
7672 \DeclareUTFCompositeSymbol\textcircled{27}{"3257}
7673 \DeclareUTFCompositeSymbol\textcircled{28}{"3258}
7674 \DeclareUTFCompositeSymbol\textcircled{29}{"3259}
7675 \DeclareUTFCompositeSymbol\textcircled{30}{"325A}
7676 \DeclareUTFCompositeSymbol\textcircled{31}{"325B}
7677 \DeclareUTFCompositeSymbol\textcircled{32}{"325C}
7678 \DeclareUTFCompositeSymbol\textcircled{33}{"325D}

```

```
7679 \DeclareUTFCompositeSymbol\textcircled{34}{\325E}
7680 \DeclareUTFCompositeSymbol\textcircled{35}{\325F}
7681 \DeclareUTFCompositeSymbol\textcircled{36}{\32B1}
7682 \DeclareUTFCompositeSymbol\textcircled{37}{\32B2}
7683 \DeclareUTFCompositeSymbol\textcircled{38}{\32B3}
7684 \DeclareUTFCompositeSymbol\textcircled{39}{\32B4}
7685 \DeclareUTFCompositeSymbol\textcircled{40}{\32B5}
7686 \DeclareUTFCompositeSymbol\textcircled{41}{\32B6}
7687 \DeclareUTFCompositeSymbol\textcircled{42}{\32B7}
7688 \DeclareUTFCompositeSymbol\textcircled{43}{\32B8}
7689 \DeclareUTFCompositeSymbol\textcircled{44}{\32B9}
7690 \DeclareUTFCompositeSymbol\textcircled{45}{\32BA}
7691 \DeclareUTFCompositeSymbol\textcircled{46}{\32BB}
7692 \DeclareUTFCompositeSymbol\textcircled{47}{\32BC}
7693 \DeclareUTFCompositeSymbol\textcircled{48}{\32BD}
7694 \DeclareUTFCompositeSymbol\textcircled{49}{\32BE}
7695 \DeclareUTFCompositeSymbol\textcircled{50}{\32BF}
7696 \DeclareUTFCompositeSymbol\textcircled{A}{\24B6}
7697 \DeclareUTFCompositeSymbol\textcircled{B}{\24B7}
7698 \DeclareUTFCompositeSymbol\textcircled{C}{\24B8}
7699 \DeclareUTFCompositeSymbol\textcircled{D}{\24B9}
7700 \DeclareUTFCompositeSymbol\textcircled{E}{\24BA}
7701 \DeclareUTFCompositeSymbol\textcircled{F}{\24BB}
7702 \DeclareUTFCompositeSymbol\textcircled{G}{\24BC}
7703 \DeclareUTFCompositeSymbol\textcircled{H}{\24BD}
7704 \DeclareUTFCompositeSymbol\textcircled{I}{\24BE}
7705 \DeclareUTFCompositeSymbol\textcircled{J}{\24BF}
7706 \DeclareUTFCompositeSymbol\textcircled{K}{\24C0}
7707 \DeclareUTFCompositeSymbol\textcircled{L}{\24C1}
7708 \DeclareUTFCompositeSymbol\textcircled{M}{\24C2}
7709 \DeclareUTFCompositeSymbol\textcircled{N}{\24C3}
7710 \DeclareUTFCompositeSymbol\textcircled{O}{\24C4}
7711 \DeclareUTFCompositeSymbol\textcircled{P}{\24C5}
7712 \DeclareUTFCompositeSymbol\textcircled{Q}{\24C6}
7713 \DeclareUTFCompositeSymbol\textcircled{R}{\24C7}
7714 \DeclareUTFCompositeSymbol\textcircled{S}{\24C8}
7715 \DeclareUTFCompositeSymbol\textcircled{T}{\24C9}
7716 \DeclareUTFCompositeSymbol\textcircled{U}{\24CA}
7717 \DeclareUTFCompositeSymbol\textcircled{V}{\24CB}
7718 \DeclareUTFCompositeSymbol\textcircled{W}{\24CC}
7719 \DeclareUTFCompositeSymbol\textcircled{X}{\24CD}
7720 \DeclareUTFCompositeSymbol\textcircled{Y}{\24CE}
7721 \DeclareUTFCompositeSymbol\textcircled{Z}{\24CF}
7722 \DeclareUTFCompositeSymbol\textcircled{a}{\24D0}
7723 \DeclareUTFCompositeSymbol\textcircled{b}{\24D1}
7724 \DeclareUTFCompositeSymbol\textcircled{c}{\24D2}
7725 \DeclareUTFCompositeSymbol\textcircled{d}{\24D3}
7726 \DeclareUTFCompositeSymbol\textcircled{e}{\24D4}
7727 \DeclareUTFCompositeSymbol\textcircled{f}{\24D5}
7728 \DeclareUTFCompositeSymbol\textcircled{g}{\24D6}
7729 \DeclareUTFCompositeSymbol\textcircled{h}{\24D7}
7730 \DeclareUTFCompositeSymbol\textcircled{i}{\24D8}
7731 \DeclareUTFCompositeSymbol\textcircled{j}{\24D9}
7732 \DeclareUTFCompositeSymbol\textcircled{k}{\24DA}
7733 \DeclareUTFCompositeSymbol\textcircled{l}{\24DB}
7734 \DeclareUTFCompositeSymbol\textcircled{m}{\24DC}
7735 \DeclareUTFCompositeSymbol\textcircled{n}{\24DD}
7736 \DeclareUTFCompositeSymbol\textcircled{o}{\24DE}
7737 \DeclareUTFCompositeSymbol\textcircled{p}{\24DF}
7738 \DeclareUTFCompositeSymbol\textcircled{q}{\24E0}
7739 \DeclareUTFCompositeSymbol\textcircled{r}{\24E1}
7740 \DeclareUTFCompositeSymbol\textcircled{s}{\24E2}
7741 \DeclareUTFCompositeSymbol\textcircled{t}{\24E3}
7742 \DeclareUTFCompositeSymbol\textcircled{u}{\24E4}
7743 \DeclareUTFCompositeSymbol\textcircled{v}{\24E5}
7744 \DeclareUTFCompositeSymbol\textcircled{w}{\24E6}
7745 \DeclareUTFCompositeSymbol\textcircled{x}{\24E7}
```

```

7746 \DeclareUTFCompositeSymbol\textcircled{y}{\24E8}
7747 \DeclareUTFCompositeSymbol\textcircled{z}{\24E9}
7748 \DeclareUTFCompositeSymbol\textsuperscript{h}{\02B0}
7749 \DeclareUTFCompositeSymbol\textsuperscript{\textth}{\02B1}
7750 \DeclareUTFCompositeSymbol\textsuperscript{j}{\02B2}
7751 \DeclareUTFCompositeSymbol\textsuperscript{r}{\02B3}
7752 \DeclareUTFCompositeSymbol\textsuperscript{\textturnr}{\02B4}
7753 \DeclareUTFCompositeSymbol\textsuperscript{\textturnrrtail}{\02B5}
7754 \DeclareUTFCompositeSymbol\textsuperscript{\textinvscr}{\02B6}
7755 \DeclareUTFCompositeSymbol\textsuperscript{w}{\02B7}
7756 \DeclareUTFCompositeSymbol\textsuperscript{y}{\02B8}
7757 \DeclareUTFCompositeSymbol\textsuperscript{\textbabygamma}{\02E0}
7758 \DeclareUTFCompositeSymbol\textsuperscript{\textgammalatinsmall}{\02E0}
7759 \DeclareUTFCompositeSymbol\textsuperscript{l}{\02E1}
7760 \DeclareUTFCompositeSymbol\textsuperscript{s}{\02E2}
7761 \DeclareUTFCompositeSymbol\textsuperscript{x}{\02E3}
7762 \DeclareUTFCompositeSymbol\textsuperscript{\textrevglotstop}{\02E4}
7763 \DeclareUTFCompositeSymbol\textsuperscript{\textrevepsilon}{\1D4C}
7764 \DeclareUTFCompositeSymbol\textsuperscript{\cyrn}{\1D78}
7765 \DeclareUTFCompositeSymbol\textsuperscript{\textbarsci}{\1DA7}
7766 \DeclareUTFCompositeSymbol\textsuperscript{V}{\2C7D}
7767 \DeclareUTFCompositeSymbol\textsuperscript{\textHbar}{\A7F8}
7768 \DeclareUTFCompositeSymbol\textsuperscript{\textHslash}{\A7F8}
7769 \DeclareUTFCompositeSymbol\textsuperscript{oe}{\A7F9}
7770 \DeclareUTFCompositeSymbol\textsubscript{h}{\2095}
7771 \DeclareUTFCompositeSymbol\textsubscript{k}{\2096}
7772 \DeclareUTFCompositeSymbol\textsubscript{l}{\2097}
7773 \DeclareUTFCompositeSymbol\textsubscript{m}{\2098}
7774 \DeclareUTFCompositeSymbol\textsubscript{n}{\2099}
7775 \DeclareUTFCompositeSymbol\textsubscript{p}{\209A}
7776 \DeclareUTFCompositeSymbol\textsubscript{s}{\209B}
7777 \DeclareUTFCompositeSymbol\textsubscript{t}{\209C}

```

以下定义取自 `hyperref` 的 `puenc.def`。

```

7778 \DeclareUTFEncodedAccent\textinvbreve{\0311}{\0311}
7779 \DeclareUTFEncodedSymbol\textsubbreve{\032E}{\203F}
7780 \DeclareUTFSymbol\textHT{\0009}
7781 \DeclareUTFSymbol\textLF{\000A}
7782 \DeclareUTFSymbol\textCR{\000D}
7783 \DeclareUTFSymbol\textnumbersign{\0023}
7784 \DeclareUTFSymbol\textparenleft{\0028}
7785 \DeclareUTFSymbol\textparenright{\0029}
7786 \DeclareUTFSymbol\textMVPlus{\002B}
7787 \DeclareUTFSymbol\textMVComma{\002C}
7788 \DeclareUTFSymbol\textMVMinus{\002D}
7789 \DeclareUTFSymbol\textMVPeriod{\002E}
7790 \DeclareUTFSymbol\textMVDivision{\002F}
7791 \DeclareUTFSymbol\textMVZero{\0030}
7792 \DeclareUTFSymbol\textMVOne{\0031}
7793 \DeclareUTFSymbol\textMVTwo{\0032}
7794 \DeclareUTFSymbol\textMVThree{\0033}
7795 \DeclareUTFSymbol\textMVFour{\0034}
7796 \DeclareUTFSymbol\textMVFive{\0035}
7797 \DeclareUTFSymbol\textMVSix{\0036}
7798 \DeclareUTFSymbol\textMVSeven{\0037}
7799 \DeclareUTFSymbol\textMVEight{\0038}
7800 \DeclareUTFSymbol\textMVNine{\0039}
7801 \DeclareUTFSymbol\textMVAt{\0040}
7802 \DeclareUTFCompositeCommand\.\{i\}{i}
7803 \DeclareUTFCompositeCommand\.\{i\}{i}
7804 \DeclareUTFSymbol\textlnot{\00AC}
7805 \DeclareUTFSymbol\textplusminus{\00B1}
7806 \DeclareUTFSymbol\textcedilla{\00B8}
7807 \DeclareUTFSymbol\textmultiply{\00D7}
7808 \DeclareUTFSymbol\textThorn{\00DE}
7809 \DeclareUTFSymbol\textdivide{\00F7}
7810 \DeclareUTFSymbol\textHslash{\0126}

```

```

7811 \DeclareUTFCompositeSymbol\k{\i}{"012F}
7812 \DeclareUTFCompositeSymbol\.\{L}{"013F}
7813 \DeclareUTFCompositeSymbol\.\{l}{"0140}
7814 \DeclareUTFSymbol\textnapostrophe{"0149}
7815 \DeclareUTFSymbol\textTslash{"0166}
7816 \DeclareUTFSymbol\texttslash{"0167}
7817 \DeclareUTFSymbol\textlongs{"017F}
7818 \DeclareUTFSymbol\texthausA{"0181}
7819 \DeclareUTFSymbol\texthausAD{"018A}
7820 \DeclareUTFSymbol\textrevE{"018E}
7821 \DeclareUTFSymbol\texthausAK{"0198}
7822 \DeclareUTFSymbol\textPUnrleg{"019E}
7823 \DeclareUTFSymbol\textinve{"01DD}
7824 \DeclareUTFSymbol\textGslash{"01E4}
7825 \DeclareUTFSymbol\textgslash{"01E5}
7826 \DeclareUTFCompositeSymbol\textinvbreve{E}{"0206}
7827 \DeclareUTFCompositeSymbol\textinvbreve{e}{"0207}
7828 \DeclareUTFCompositeSymbol\textinvbreve{I}{"020A}
7829 \DeclareUTFCompositeSymbol\textinvbreve{i}{"020B}
7830 \DeclareUTFCompositeSymbol\textinvbreve{\i}{"020B}
7831 \DeclareUTFCompositeSymbol\textinvbreve{O}{"020E}
7832 \DeclareUTFCompositeSymbol\textinvbreve{o}{"020F}
7833 \DeclareUTFCompositeSymbol\textinvbreve{U}{"0216}
7834 \DeclareUTFCompositeSymbol\textinvbreve{u}{"0217}
7835 \DeclareUTFSymbol\j{"0237}
7836 \DeclareUTFSymbol\textPUdblig{"0238}
7837 \DeclareUTFSymbol\textPUqplig{"0239}
7838 \DeclareUTFSymbol\textslashc{"023C}
7839 \DeclareUTFSymbol\textniepsilon{"025B}
7840 \DeclareUTFSymbol\textipagamma{"0263}
7841 \DeclareUTFSymbol\textniiota{"0269}
7842 \DeclareUTFSymbol\textniphil{"0278}
7843 \DeclareUTFSymbol\textniupsilon{"028A}
7844 \DeclareUTFSymbol\textring{"02DA}
7845 \DeclareUTFSymbol\texttilde{"02DC}
7846 \DeclareUTFSymbol\texthungarumlaut{"02DD}
7847 \DeclareUTFSymbol\textringlow{"02F3}
7848 \DeclareUTFSymbol\texttildebelow{"02F7}
7849 \DeclareUTFCommand\textnewtie{\textinvbreve\ }
7850 \DeclareUTFCommand\textdotbelow{\d\ }
7851 \DeclareUTFSymbol\textmacronbelow{"02CD}
7852 \DeclareUTFCommand\texttie{\t\ }
7853 \DeclareUTFSymbol\textnumeralsigngreek{"0374}
7854 \DeclareUTFSymbol\textnumeralsignlowergreek{"0375}
7855 \DeclareUTFCompositeSymbol\'\{\textAlpha}{"0386}
7856 \DeclareUTFCompositeSymbol\'\{\textEpsilon}{"0388}
7857 \DeclareUTFCompositeSymbol\'\{\textEta}{"0389}
7858 \DeclareUTFCompositeSymbol\'\{\textIota}{"038A}
7859 \DeclareUTFCompositeSymbol\'\{\textOmicron}{"038C}
7860 \DeclareUTFCompositeSymbol\'\{\textUpsilon}{"038E}
7861 \DeclareUTFCompositeSymbol\'\{\textOmega}{"038F}
7862 \DeclareUTFCompositeSymbol\'\{\textIotadieresis}{"0390}
7863 \DeclareUTFSymbol\textIotadieresis{"03AA}
7864 \DeclareUTFCompositeSymbol\'\{\textIota}{"03AA}
7865 \DeclareUTFCompositeSymbol\'\{\textUpsilon}{"03AB}
7866 \DeclareUTFCompositeSymbol\'\{\textalpha}{"03AC}
7867 \DeclareUTFCompositeSymbol\'\{\textepsilon}{"03AD}
7868 \DeclareUTFCompositeSymbol\'\{\texteta}{"03AE}
7869 \DeclareUTFCompositeSymbol\'\{\textiota}{"03AF}
7870 \DeclareUTFCompositeSymbol\'\{\textupsilonacute}{"03B0}
7871 \DeclareUTFSymbol\textmugreek{"03BC}
7872 \DeclareUTFSymbol\textvarsigma{"03C2}
7873 \DeclareUTFCompositeSymbol\'\{\textiota}{"03CA}
7874 \DeclareUTFCompositeSymbol\'\{\textupsilon}{"03CB}
7875 \DeclareUTFCompositeSymbol\'\{\textomicron}{"03CC}
7876 \DeclareUTFSymbol\textupsilonacute{"03CD}
7877 \DeclareUTFCompositeSymbol\'\{\textupsilon}{"03CD}

```

```
7878 \DeclareUTFCompositeSymbol\'\{\textomega}\{"03CE}
7879 \DeclareUTFSymbol\textStigmagreek{"03DA}
7880 \DeclareUTFSymbol\textstigmagreek{"03DB}
7881 \DeclareUTFSymbol\textDigammagreek{"03DC}
7882 \DeclareUTFSymbol\textdigammagreek{"03DD}
7883 \DeclareUTFSymbol\textKoppagreek{"03DE}
7884 \DeclareUTFSymbol\textkoppagreek{"03DF}
7885 \DeclareUTFSymbol\textSampigreek{"03E0}
7886 \DeclareUTFSymbol\textsampigreek{"03E1}
7887 \DeclareUTFSymbol\textbackepsilon{"03F6}
7888 \DeclareUTFCompositeSymbol\`\{\CYRE}\{"0400}
7889 \DeclareUTFSymbol\CYRYO{"0401}
7890 \DeclareUTFCompositeSymbol\`\{\CYRE}\{"0401}
7891 \DeclareUTFSymbol\CYRDJE{"0402}
7892 \DeclareUTFCompositeSymbol\'\{\CYRG}\{"0403}
7893 \DeclareUTFSymbol\CYRIE{"0404}
7894 \DeclareUTFSymbol\CYRDZE{"0405}
7895 \DeclareUTFSymbol\CYRII{"0406}
7896 \DeclareUTFSymbol\CYRYI{"0407}
7897 \DeclareUTFCompositeSymbol\`\{\CYRII}\{"0407}
7898 \DeclareUTFSymbol\CYRJE{"0408}
7899 \DeclareUTFSymbol\CYRLJE{"0409}
7900 \DeclareUTFSymbol\CYRNJE{"040A}
7901 \DeclareUTFSymbol\CYRTSHE{"040B}
7902 \DeclareUTFCompositeSymbol\'\{\CYRK}\{"040C}
7903 \DeclareUTFCompositeSymbol\`\{\CYRI}\{"040D}
7904 \DeclareUTFSymbol\CYRUSHRT{"040E}
7905 \DeclareUTFCompositeSymbol\U{\CYRU}\{"040E}
7906 \DeclareUTFSymbol\CYRDZHE{"040F}
7907 \DeclareUTFSymbol\CYRA{"0410}
7908 \DeclareUTFSymbol\CYRB{"0411}
7909 \DeclareUTFSymbol\CYRV{"0412}
7910 \DeclareUTFSymbol\CYRG{"0413}
7911 \DeclareUTFSymbol\CYRD{"0414}
7912 \DeclareUTFSymbol\CYRE{"0415}
7913 \DeclareUTFSymbol\CYRZH{"0416}
7914 \DeclareUTFSymbol\CYRZ{"0417}
7915 \DeclareUTFSymbol\CYRI{"0418}
7916 \DeclareUTFSymbol\CYRISHRT{"0419}
7917 \DeclareUTFCompositeSymbol\U{\CYRI}\{"0419}
7918 \DeclareUTFSymbol\CYRK{"041A}
7919 \DeclareUTFSymbol\CYRL{"041B}
7920 \DeclareUTFSymbol\CYRM{"041C}
7921 \DeclareUTFSymbol\CYRN{"041D}
7922 \DeclareUTFSymbol\CYRO{"041E}
7923 \DeclareUTFSymbol\CYRP{"041F}
7924 \DeclareUTFSymbol\CYRR{"0420}
7925 \DeclareUTFSymbol\CYRS{"0421}
7926 \DeclareUTFSymbol\CYRT{"0422}
7927 \DeclareUTFSymbol\CYRU{"0423}
7928 \DeclareUTFSymbol\CYRF{"0424}
7929 \DeclareUTFSymbol\CYRH{"0425}
7930 \DeclareUTFSymbol\CYRC{"0426}
7931 \DeclareUTFSymbol\CYRCH{"0427}
7932 \DeclareUTFSymbol\CYRSH{"0428}
7933 \DeclareUTFSymbol\CYRSHCH{"0429}
7934 \DeclareUTFSymbol\CYRHRDSN{"042A}
7935 \DeclareUTFSymbol\CYRERY{"042B}
7936 \DeclareUTFSymbol\CYRSFTSN{"042C}
7937 \DeclareUTFSymbol\CYREREV{"042D}
7938 \DeclareUTFSymbol\CYRYU{"042E}
7939 \DeclareUTFSymbol\CYRYA{"042F}
7940 \DeclareUTFSymbol\cyra{"0430}
7941 \DeclareUTFSymbol\cyrb{"0431}
7942 \DeclareUTFSymbol\cyrv{"0432}
7943 \DeclareUTFSymbol\cyrg{"0433}
7944 \DeclareUTFSymbol\cyrd{"0434}
```

```
7945 \DeclareUTFSymbol\cyre{"0435}
7946 \DeclareUTFSymbol\cyrzh{"0436}
7947 \DeclareUTFSymbol\cyrz{"0437}
7948 \DeclareUTFSymbol\cyri{"0438}
7949 \DeclareUTFSymbol\cyrishrt{"0439}
7950 \DeclareUTFCompositeSymbol\U{\cyri}{0439}
7951 \DeclareUTFSymbol\cyrk{"043A}
7952 \DeclareUTFSymbol\cyr1{"043B}
7953 \DeclareUTFSymbol\cyrm{"043C}
7954 \DeclareUTFSymbol\cyrn{"043D}
7955 \DeclareUTFSymbol\cyro{"043E}
7956 \DeclareUTFSymbol\cyrp{"043F}
7957 \DeclareUTFSymbol\cyrr{"0440}
7958 \DeclareUTFSymbol\cyrs{"0441}
7959 \DeclareUTFSymbol\cyrt{"0442}
7960 \DeclareUTFSymbol\cyr1{"0443}
7961 \DeclareUTFSymbol\cyrf{"0444}
7962 \DeclareUTFSymbol\cyrh{"0445}
7963 \DeclareUTFSymbol\cyrc{"0446}
7964 \DeclareUTFSymbol\cyrch{"0447}
7965 \DeclareUTFSymbol\cyrsh{"0448}
7966 \DeclareUTFSymbol\cyrshch{"0449}
7967 \DeclareUTFSymbol\cyrhrdsn{"044A}
7968 \DeclareUTFSymbol\cyrrery{"044B}
7969 \DeclareUTFSymbol\cyrsftsn{"044C}
7970 \DeclareUTFSymbol\cyrrerev{"044D}
7971 \DeclareUTFSymbol\cyr1{"044E}
7972 \DeclareUTFSymbol\cyr1{"044F}
7973 \DeclareUTFCompositeSymbol\`{\cyre}{0450}
7974 \DeclareUTFSymbol\cyr1{"0451}
7975 \DeclareUTFCompositeSymbol\`{\cyre}{0451}
7976 \DeclareUTFSymbol\cyrdje{"0452}
7977 \DeclareUTFCompositeSymbol\'\{\cyrg}{0453}
7978 \DeclareUTFSymbol\cyrie{"0454}
7979 \DeclareUTFSymbol\cyrdze{"0455}
7980 \DeclareUTFSymbol\cyrii{"0456}
7981 \DeclareUTFSymbol\cyryi{"0457}
7982 \DeclareUTFCompositeSymbol\`{\cyrii}{0457}
7983 \DeclareUTFSymbol\cyrje{"0458}
7984 \DeclareUTFSymbol\cyrlje{"0459}
7985 \DeclareUTFSymbol\cyrnje{"045A}
7986 \DeclareUTFSymbol\cyrtshe{"045B}
7987 \DeclareUTFCompositeSymbol\'\{\cyrk}{045C}
7988 \DeclareUTFCompositeSymbol\`{\cyri}{045D}
7989 \DeclareUTFSymbol\cyrushrt{"045E}
7990 \DeclareUTFCompositeSymbol\U{\curu}{045E}
7991 \DeclareUTFSymbol\cyrdzhe{"045F}
7992 \DeclareUTFSymbol\CYROMEGA{"0460}
7993 \DeclareUTFSymbol\cyromega{"0461}
7994 \DeclareUTFSymbol\CYRYAT{"0462}
7995 \DeclareUTFSymbol\cyryat{"0463}
7996 \DeclareUTFSymbol\CYRIOTE{"0464}
7997 \DeclareUTFSymbol\cyriote{"0465}
7998 \DeclareUTFSymbol\CYRLYUS{"0466}
7999 \DeclareUTFSymbol\cyrl1{"0467}
8000 \DeclareUTFSymbol\CYRIOTLYUS{"0468}
8001 \DeclareUTFSymbol\cyriotlyus{"0469}
8002 \DeclareUTFSymbol\CYRBYUS{"046A}
8003 \DeclareUTFSymbol\cyrb1{"046B}
8004 \DeclareUTFSymbol\CYRIOTBYUS{"046C}
8005 \DeclareUTFSymbol\cyriotbyus{"046D}
8006 \DeclareUTFSymbol\CYRKSI{"046E}
8007 \DeclareUTFSymbol\cyrk1{"046F}
8008 \DeclareUTFSymbol\CYRPSI{"0470}
8009 \DeclareUTFSymbol\cyrp1{"0471}
8010 \DeclareUTFSymbol\CYRFITA{"0472}
8011 \DeclareUTFSymbol\cyrf1{"0473}
```

```
8012 \DeclareUTFSymbol\CYRIZH{"0474}
8013 \DeclareUTFSymbol\cyrizh{"0475}
8014 \DeclareUTFCompositeSymbol\C{\CYRIZH}{0476}
8015 \DeclareUTFCompositeSymbol\C{\cyrizh}{0477}
8016 \DeclareUTFSymbol\CYRUK{"0478}
8017 \DeclareUTFSymbol\cyruk{"0479}
8018 \DeclareUTFSymbol\CYROMEGARND{"047A}
8019 \DeclareUTFSymbol\cyromegarnd{"047B}
8020 \DeclareUTFSymbol\CYROMEGATITLO{"047C}
8021 \DeclareUTFSymbol\cyromegatitlo{"047D}
8022 \DeclareUTFSymbol\CYROT{"047E}
8023 \DeclareUTFSymbol\cyrot{"047F}
8024 \DeclareUTFSymbol\CYRKOPPA{"0480}
8025 \DeclareUTFSymbol\cyrkoppa{"0481}
8026 \DeclareUTFSymbol\cyrthousands{"0482}
8027 \DeclareUTFSymbol\CYRISHRTDSC{"048A}
8028 \DeclareUTFSymbol\cyrishrtdsc{"048B}
8029 \DeclareUTFSymbol\CYRSEMISFTSN{"048C}
8030 \DeclareUTFSymbol\cyrsemisftsn{"048D}
8031 \DeclareUTFSymbol\CYRRTICK{"048E}
8032 \DeclareUTFSymbol\cyrrtick{"048F}
8033 \DeclareUTFSymbol\CYRGUP{"0490}
8034 \DeclareUTFSymbol\cyrgup{"0491}
8035 \DeclareUTFSymbol\CYRGHCRS{"0492}
8036 \DeclareUTFSymbol\cyrghcrs{"0493}
8037 \DeclareUTFSymbol\CYRGHK{"0494}
8038 \DeclareUTFSymbol\cyrghk{"0495}
8039 \DeclareUTFSymbol\CYRZHDSC{"0496}
8040 \DeclareUTFSymbol\cyrzhdsc{"0497}
8041 \DeclareUTFSymbol\CYRZDSC{"0498}
8042 \DeclareUTFCompositeSymbol\c{\CYRZ}{0498}
8043 \DeclareUTFSymbol\cyrzdsc{"0499}
8044 \DeclareUTFCompositeSymbol\c{\cyrz}{0499}
8045 \DeclareUTFSymbol\CYRKDSC{"049A}
8046 \DeclareUTFSymbol\cyrkdsc{"049B}
8047 \DeclareUTFSymbol\CYRKVCRS{"049C}
8048 \DeclareUTFSymbol\cyrkvcrs{"049D}
8049 \DeclareUTFSymbol\CYRKHCRS{"049E}
8050 \DeclareUTFSymbol\cyrkhcrs{"049F}
8051 \DeclareUTFSymbol\CYRKBEAK{"04A0}
8052 \DeclareUTFSymbol\cyrkbeak{"04A1}
8053 \DeclareUTFSymbol\CYRNDSC{"04A2}
8054 \DeclareUTFSymbol\cyrndsc{"04A3}
8055 \DeclareUTFSymbol\CYRNG{"04A4}
8056 \DeclareUTFSymbol\cyrng{"04A5}
8057 \DeclareUTFSymbol\CYRPHK{"04A6}
8058 \DeclareUTFSymbol\cyrphk{"04A7}
8059 \DeclareUTFSymbol\CYRABHHA{"04A8}
8060 \DeclareUTFSymbol\cyrabhha{"04A9}
8061 \DeclareUTFSymbol\CYRSDSC{"04AA}
8062 \DeclareUTFCompositeSymbol\k{\CYRS}{04AA}
8063 \DeclareUTFSymbol\cyrsdsc{"04AB}
8064 \DeclareUTFCompositeSymbol\k{\cyrs}{04AB}
8065 \DeclareUTFSymbol\CYRTDSC{"04AC}
8066 \DeclareUTFSymbol\cyrtedsc{"04AD}
8067 \DeclareUTFSymbol\CYRY{"04AE}
8068 \DeclareUTFSymbol\cyry{"04AF}
8069 \DeclareUTFSymbol\CYRYHCRS{"04B0}
8070 \DeclareUTFSymbol\cyryhcrs{"04B1}
8071 \DeclareUTFSymbol\CYRHDSC{"04B2}
8072 \DeclareUTFSymbol\cyrhdsdsc{"04B3}
8073 \DeclareUTFSymbol\CYRTETSE{"04B4}
8074 \DeclareUTFSymbol\cyrtetse{"04B5}
8075 \DeclareUTFSymbol\CYRCHRDSC{"04B6}
8076 \DeclareUTFSymbol\cyrchrdsc{"04B7}
8077 \DeclareUTFSymbol\CYRCHVCRS{"04B8}
8078 \DeclareUTFSymbol\cyrchvcrs{"04B9}
```

```
8079 \DeclareUTFSymbol\CYRSHHA{"04BA}
8080 \DeclareUTFSymbol\cyrshha{"04BB}
8081 \DeclareUTFSymbol\CYRABHCH{"04BC}
8082 \DeclareUTFSymbol\cyrabhch{"04BD}
8083 \DeclareUTFSymbol\CYRABHCHDSC{"04BE}
8084 \DeclareUTFCompositeSymbol\k{\CYRABHCH}{04BE}
8085 \DeclareUTFSymbol\cyrabhchdsc{"04BF}
8086 \DeclareUTFCompositeSymbol\k{\cyrabhch}{04BF}
8087 \DeclareUTFSymbol\CYRpalochka{"04C0}
8088 \DeclareUTFCompositeSymbol\U{\CYRZH}{04C1}
8089 \DeclareUTFCompositeSymbol\U{\cyrzh}{04C2}
8090 \DeclareUTFSymbol\CYRKHK{"04C3}
8091 \DeclareUTFSymbol\cyrkhk{"04C4}
8092 \DeclareUTFSymbol\CYRLDSC{"04C5}
8093 \DeclareUTFSymbol\cyrldsc{"04C6}
8094 \DeclareUTFSymbol\CYRNHK{"04C7}
8095 \DeclareUTFSymbol\cyrnhk{"04C8}
8096 \DeclareUTFSymbol\CYRCHLDSC{"04CB}
8097 \DeclareUTFSymbol\cyrchldsc{"04CC}
8098 \DeclareUTFSymbol\CYRMDSC{"04CD}
8099 \DeclareUTFSymbol\cyrmdsc{"04CE}
8100 \DeclareUTFCompositeSymbol\U{\CYRA}{04D0}
8101 \DeclareUTFCompositeSymbol\U{\cyr a}{04D1}
8102 \DeclareUTFCompositeSymbol\{"{\CYRA}{04D2}
8103 \DeclareUTFCompositeSymbol\{"{\cyr a}{04D3}
8104 \DeclareUTFSymbol\CYRAE{"04D4}
8105 \DeclareUTFSymbol\cyr a e{"04D5}
8106 \DeclareUTFCompositeSymbol\U{\CYRE}{04D6}
8107 \DeclareUTFCompositeSymbol\U{\cyr e}{04D7}
8108 \DeclareUTFSymbol\CYRSCHWA{"04D8}
8109 \DeclareUTFSymbol\cyr schwa{"04D9}
8110 \DeclareUTFCompositeSymbol\{"{\CYRSCHWA}{04DA}
8111 \DeclareUTFCompositeSymbol\{"{\cyr schwa}{04DB}
8112 \DeclareUTFCompositeSymbol\{"{\CYRZH}{04DC}
8113 \DeclareUTFCompositeSymbol\{"{\cyr zh}{04DD}
8114 \DeclareUTFCompositeSymbol\{"{\CYRZ}{04DE}
8115 \DeclareUTFCompositeSymbol\{"{\cyr z}{04DF}
8116 \DeclareUTFSymbol\CYRABHDZE{"04E0}
8117 \DeclareUTFSymbol\cyr abhdze{"04E1}
8118 \DeclareUTFCompositeSymbol\={\CYRI}{04E2}
8119 \DeclareUTFCompositeSymbol\={\cyr i}{04E3}
8120 \DeclareUTFCompositeSymbol\{"{\CYRI}{04E4}
8121 \DeclareUTFCompositeSymbol\{"{\cyr i}{04E5}
8122 \DeclareUTFCompositeSymbol\{"{\CYRO}{04E6}
8123 \DeclareUTFCompositeSymbol\{"{\cyr o}{04E7}
8124 \DeclareUTFSymbol\CYROTLD{"04E8}
8125 \DeclareUTFSymbol\cyr otld{"04E9}
8126 \DeclareUTFCompositeSymbol\{"{\CYROTLD}{04EA}
8127 \DeclareUTFCompositeSymbol\{"{\cyr otld}{04EB}
8128 \DeclareUTFCompositeSymbol\{"{\CYREREV}{04EC}
8129 \DeclareUTFCompositeSymbol\{"{\cyrerev}{04ED}
8130 \DeclareUTFCompositeSymbol\={\CYRU}{04EE}
8131 \DeclareUTFCompositeSymbol\={\cyr u}{04EF}
8132 \DeclareUTFCompositeSymbol\{"{\CYRU}{04F0}
8133 \DeclareUTFCompositeSymbol\{"{\cyr u}{04F1}
8134 \DeclareUTFCompositeSymbol\H{\CYRU}{04F2}
8135 \DeclareUTFCompositeSymbol\H{\cyr u}{04F3}
8136 \DeclareUTFCompositeSymbol\{"{\CYRCH}{04F4}
8137 \DeclareUTFCompositeSymbol\{"{\cyr ch}{04F5}
8138 \DeclareUTFSymbol\CYRGDSC{"04F6}
8139 \DeclareUTFSymbol\cyr gdsc{"04F7}
8140 \DeclareUTFCompositeSymbol\{"{\CYRERY}{04F8}
8141 \DeclareUTFCompositeSymbol\{"{\cyrery}{04F9}
8142 \DeclareUTFSymbol\CYRHHK{"04FC}
8143 \DeclareUTFSymbol\cyr hkh{"04FD}
8144 \DeclareUTFSymbol\sofpasuq{"05C3}
8145 \DeclareUTFSymbol\heba1ef{"05D0}
```

```

8146 \DeclareUTFSymbol\hebbet{"05D1}
8147 \DeclareUTFSymbol\hebgimel{"05D2}
8148 \DeclareUTFSymbol\hebdalet{"05D3}
8149 \DeclareUTFSymbol\hebhe{"05D4}
8150 \DeclareUTFSymbol\hebvav{"05D5}
8151 \DeclareUTFSymbol\hebzayin{"05D6}
8152 \DeclareUTFSymbol\hebhet{"05D7}
8153 \DeclareUTFSymbol\hebtet{"05D8}
8154 \DeclareUTFSymbol\hebyod{"05D9}
8155 \DeclareUTFSymbol\hebfinalkaf{"05DA}
8156 \DeclareUTFSymbol\hebkafe{"05DB}
8157 \DeclareUTFSymbol\heblamed{"05DC}
8158 \DeclareUTFSymbol\hebfinalmem{"05DD}
8159 \DeclareUTFSymbol\hebmeme{"05DE}
8160 \DeclareUTFSymbol\hebfinalnun{"05DF}
8161 \DeclareUTFSymbol\hebnun{"05E0}
8162 \DeclareUTFSymbol\hebsamekh{"05E1}
8163 \DeclareUTFSymbol\hebayin{"05E2}
8164 \DeclareUTFSymbol\hebfinalpe{"05E3}
8165 \DeclareUTFSymbol\hebpe{"05E4}
8166 \DeclareUTFSymbol\hebfinaltsadi{"05E5}
8167 \DeclareUTFSymbol\hebtsadi{"05E6}
8168 \DeclareUTFSymbol\he bqof{"05E7}
8169 \DeclareUTFSymbol\hebre sh{"05E8}
8170 \DeclareUTFSymbol\hebshin{"05E9}
8171 \DeclareUTFSymbol\hebtav{"05EA}
8172 \DeclareUTFSymbol\doublevav{"05F0}
8173 \DeclareUTFSymbol\vavyod{"05F1}
8174 \DeclareUTFSymbol\doubleyod{"05F2}
8175 \DeclareUTFSymbol\textscd{"1D05}
8176 \DeclareUTFSymbol\textPU sck{"1D0B}
8177 \DeclareUTFSymbol\textPU scm{"1D0D}
8178 \DeclareUTFSymbol\textPU scp{"1D18}
8179 \DeclareUTFSymbol\textPU revscr{"1D19}
8180 \DeclareUTFSymbol\texti inferior{"1D62}
8181 \DeclareUTFSymbol\textr inferior{"1D63}
8182 \DeclareUTFSymbol\textu inferior{"1D64}
8183 \DeclareUTFSymbol\textv inferior{"1D65}
8184 \DeclareUTFSymbol\textbetainferior{"1D66}
8185 \DeclareUTFSymbol\textgammainferior{"1D67}
8186 \DeclareUTFSymbol\textrho inferior{"1D68}
8187 \DeclareUTFSymbol\textphi inferior{"1D69}
8188 \DeclareUTFSymbol\textchi inferior{"1D6A}
8189 \DeclareUTFSymbol\textbarsci{"1D7B}
8190 \DeclareUTFSymbol\textbarp{"1D7D}
8191 \DeclareUTFSymbol\textbarscu{"1D7E}
8192 \DeclareUTFSymbol\textPU rho ka{"1D8F}
8193 \DeclareUTFSymbol\textPU rho ke{"1D92}
8194 \DeclareUTFSymbol\textPU rho ke epsilon{"1D93}
8195 \DeclareUTFSymbol\textPU rho ke openo{"1D97}
8196 \DeclareUTFCompositeSymbol\textsubbreve{H}{1E2A}
8197 \DeclareUTFCompositeSymbol\textsubbreve{h}{1E2B}
8198 \DeclareUTFCompositeSymbol\.\{\textlong s}{1E9B}
8199 \DeclareUTFSymbol\textcompwordmark{"200C}
8200 \DeclareUTFSymbol\textth dotfor{"2025}
8201 \DeclareUTFSymbol\textprime{"2032}
8202 \DeclareUTFSymbol\textsecond{"2033}
8203 \DeclareUTFSymbol\textthird{"2034}
8204 \DeclareUTFSymbol\textbackprime{"2035}
8205 \DeclareUTFSymbol\textlefttherefore{"2056}
8206 \DeclareUTFSymbol\textfourth{"2057}
8207 \DeclareUTFSymbol\textdiamonddots{"2058}
8208 \DeclareUTFSymbol\textzerosuperior{"2070}
8209 \DeclareUTFSymbol\textisuperior{"2071}
8210 \DeclareUTFSymbol\textfoursuperior{"2074}
8211 \DeclareUTFSymbol\textfivesuperior{"2075}
8212 \DeclareUTFSymbol\textsixsuperior{"2076}

```

```
8213 \DeclareUTFSymbol\textsevensuperior{"2077}
8214 \DeclareUTFSymbol\texteightsuperior{"2078}
8215 \DeclareUTFSymbol\textninesuperior{"2079}
8216 \DeclareUTFSymbol\textplussuperior{"207A}
8217 \DeclareUTFSymbol\textminussuperior{"207B}
8218 \DeclareUTFSymbol\textequalsuperior{"207C}
8219 \DeclareUTFSymbol\textparenleftsuperior{"207D}
8220 \DeclareUTFSymbol\textparenrightsuperior{"207E}
8221 \DeclareUTFSymbol\textnsuperior{"207F}
8222 \DeclareUTFSymbol\textzeroinferior{"2080}
8223 \DeclareUTFSymbol\textoneinferior{"2081}
8224 \DeclareUTFSymbol\texttwoinferior{"2082}
8225 \DeclareUTFSymbol\textthreeinferior{"2083}
8226 \DeclareUTFSymbol\textfourinferior{"2084}
8227 \DeclareUTFSymbol\textfiveinferior{"2085}
8228 \DeclareUTFSymbol\textsixinferior{"2086}
8229 \DeclareUTFSymbol\textseveninferior{"2087}
8230 \DeclareUTFSymbol\texteightinferior{"2088}
8231 \DeclareUTFSymbol\textnineinferior{"2089}
8232 \DeclareUTFSymbol\textplusinferior{"208A}
8233 \DeclareUTFSymbol\textminusinferior{"208B}
8234 \DeclareUTFSymbol\textequalsinferior{"208C}
8235 \DeclareUTFSymbol\textparenleftinferior{"208D}
8236 \DeclareUTFSymbol\textparenrightinferior{"208E}
8237 \DeclareUTFSymbol\textainferior{"2090}
8238 \DeclareUTFSymbol\texteinferior{"2091}
8239 \DeclareUTFSymbol\textoinferior{"2092}
8240 \DeclareUTFSymbol\textxinferior{"2093}
8241 \DeclareUTFSymbol\textschwainferior{"2094}
8242 \DeclareUTFSymbol\texthinferior{"2095}
8243 \DeclareUTFSymbol\textkinferior{"2096}
8244 \DeclareUTFSymbol\textlinferior{"2097}
8245 \DeclareUTFSymbol\textminferior{"2098}
8246 \DeclareUTFSymbol\textninferior{"2099}
8247 \DeclareUTFSymbol\textpinferior{"209A}
8248 \DeclareUTFSymbol\textsinferior{"209B}
8249 \DeclareUTFSymbol\texttinferior{"209C}
8250 \DeclareUTFSymbol\textpeseta{"20A7}
8251 \DeclareUTFSymbol\textDeleatur{"20B0}
8252 \DeclareUTFSymbol\textguarani{"20B2}
8253 \DeclareUTFSymbol\textslash{"210F}
8254 \DeclareUTFSymbol\textIm{"2111}
8255 \DeclareUTFSymbol\textell{"2113}
8256 \DeclareUTFSymbol\textwp{"2118}
8257 \DeclareUTFSymbol\textRe{"211C}
8258 \DeclareUTFSymbol\textriota{"2129}
8259 \DeclareUTFSymbol\textangstrom{"212B}
8260 \DeclareUTFSymbol\textfax{"213B}
8261 \DeclareUTFSymbol\textinvamp{"214B}
8262 \DeclareUTFSymbol\textoneseventh{"2150}
8263 \DeclareUTFSymbol\textoneninth{"2151}
8264 \DeclareUTFSymbol\textonetenth{"2152}
8265 \DeclareUTFSymbol\textonethird{"2153}
8266 \DeclareUTFSymbol\texttwothirds{"2154}
8267 \DeclareUTFSymbol\textonefifth{"2155}
8268 \DeclareUTFSymbol\texttwoififths{"2156}
8269 \DeclareUTFSymbol\textthreeififths{"2157}
8270 \DeclareUTFSymbol\textfourififths{"2158}
8271 \DeclareUTFSymbol\textonesixth{"2159}
8272 \DeclareUTFSymbol\textfivesixths{"215A}
8273 \DeclareUTFSymbol\textoneeighth{"215B}
8274 \DeclareUTFSymbol\textthreeeighths{"215C}
8275 \DeclareUTFSymbol\textfiveeighths{"215D}
8276 \DeclareUTFSymbol\textseveeneighths{"215E}
8277 \DeclareUTFSymbol\textrevc{"2184}
8278 \DeclareUTFSymbol\textzerot thirds{"2189}
8279 \DeclareUTFSymbol\textnleftarrow{"219A}
```

```

8280 \DeclareUTFSymbol\textnrightarrow{"219B}
8281 \DeclareUTFSymbol\texttwoheadleftarrow{"219E}
8282 \DeclareUTFCommand\texttwoheadleftarrow{\textlstrickthru\texttwoheadleftarrow}
8283 \DeclareUTFSymbol\texttwoheaduparrow{"219F}
8284 \DeclareUTFSymbol\texttwoheadrightarrow{"21A0}
8285 \DeclareUTFCommand\texttwoheadrightarrow{\textlstrickthru\texttwoheadrightarrow}
8286 \DeclareUTFSymbol\texttwoheaddownarrow{"21A1}
8287 \DeclareUTFSymbol\textleftarrowtail{"21A2}
8288 \DeclareUTFSymbol\textrightarrowtail{"21A3}
8289 \DeclareUTFSymbol\textmapsto{"21A6}
8290 \DeclareUTFSymbol\texthookleftarrow{"21A9}
8291 \DeclareUTFSymbol\texthookrightarrow{"21AA}
8292 \DeclareUTFSymbol\textlooparrowleft{"21AB}
8293 \DeclareUTFSymbol\textlooparrowright{"21AC}
8294 \DeclareUTFSymbol\textnleftrightarrow{"21AE}
8295 \DeclareUTFSymbol\textlightning{"21AF}
8296 \DeclareUTFSymbol\textdlash{"21B5}
8297 \DeclareUTFSymbol\textcurvearrowleft{"21B6}
8298 \DeclareUTFSymbol\textcurvearrowright{"21B7}
8299 \DeclareUTFSymbol\textleftharpoonup{"21BC}
8300 \DeclareUTFSymbol\textleftharpoondown{"21BD}
8301 \DeclareUTFSymbol\textupharpoonright{"21BE}
8302 \DeclareUTFSymbol\textupharpoonleft{"21BF}
8303 \DeclareUTFSymbol\texttrightharpoonup{"21C0}
8304 \DeclareUTFSymbol\texttrightharpoondown{"21C1}
8305 \DeclareUTFSymbol\textdownharpoonright{"21C2}
8306 \DeclareUTFSymbol\textdownharpoonleft{"21C3}
8307 \DeclareUTFSymbol\textrightleftarrows{"21C4}
8308 \DeclareUTFSymbol\textupdownarrows{"21C5}
8309 \DeclareUTFSymbol\textleftrightrightarrows{"21C6}
8310 \DeclareUTFSymbol\textleftleftarrows{"21C7}
8311 \DeclareUTFSymbol\textupuparrows{"21C8}
8312 \DeclareUTFSymbol\texttrightrightarrows{"21C9}
8313 \DeclareUTFSymbol\textdowndownarrows{"21CA}
8314 \DeclareUTFSymbol\textleftrightharpoons{"21CB}
8315 \DeclareUTFSymbol\textrightleftharpoons{"21CC}
8316 \DeclareUTFSymbol\textnLeftarrow{"21CD}
8317 \DeclareUTFSymbol\textnLeftrightarrow{"21CE}
8318 \DeclareUTFSymbol\textnRightarrow{"21CF}
8319 \DeclareUTFSymbol\textLeftarrow{"21D0}
8320 \DeclareUTFSymbol\textUparrow{"21D1}
8321 \DeclareUTFSymbol\textRightarrow{"21D2}
8322 \DeclareUTFSymbol\textDownarrow{"21D3}
8323 \DeclareUTFSymbol\textLeftrightarrow{"21D4}
8324 \DeclareUTFSymbol\textUpdownarrow{"21D5}
8325 \DeclareUTFSymbol\textNwarrow{"21D6}
8326 \DeclareUTFSymbol\textNearrow{"21D7}
8327 \DeclareUTFSymbol\textSearrow{"21D8}
8328 \DeclareUTFSymbol\textSwarrow{"21D9}
8329 \DeclareUTFSymbol\textLleftarrow{"21DA}
8330 \DeclareUTFSymbol\textRrightarrow{"21DB}
8331 \DeclareUTFSymbol\textleftsquigarrow{"21DC}
8332 \DeclareUTFSymbol\textrightsquigarrow{"21DD}
8333 \DeclareUTFSymbol\textdashleftarrow{"21E0}
8334 \DeclareUTFSymbol\textdasheduparrow{"21E1}
8335 \DeclareUTFSymbol\textdashrightarrow{"21E2}
8336 \DeclareUTFSymbol\textdasheddownarrow{"21E3}
8337 \DeclareUTFSymbol\textpointer{"21E8}
8338 \DeclareUTFSymbol\textdownuparrows{"21F5}
8339 \DeclareUTFSymbol\textleftarrowtriangle{"21FD}
8340 \DeclareUTFSymbol\textrightarrowtriangle{"21FE}
8341 \DeclareUTFSymbol\textleftrightarrowtriangle{"21FF}
8342 \DeclareUTFSymbol\textforall{"2200}
8343 \DeclareUTFSymbol\textcomplement{"2201}
8344 \DeclareUTFSymbol\textpartial{"2202}
8345 \DeclareUTFSymbol\textexists{"2203}
8346 \DeclareUTFSymbol\textnexists{"2204}

```

```

8347 \DeclareUTFSymbol\textemptyset{"2205}
8348 \DeclareUTFSymbol\texttriangle{"2206}
8349 \DeclareUTFSymbol\textnabla{"2207}
8350 \DeclareUTFSymbol\textin{"2208}
8351 \DeclareUTFSymbol\textnotin{"2209}
8352 \DeclareUTFSymbol\textsmallin{"220A}
8353 \DeclareUTFSymbol\textni{"220B}
8354 \DeclareUTFSymbol\textnotowner{"220C}
8355 \DeclareUTFSymbol\textsmallowns{"220D}
8356 \DeclareUTFSymbol\textprod{"220F}
8357 \DeclareUTFSymbol\textamalg{"2210}
8358 \DeclareUTFSymbol\textsum{"2211}
8359 \DeclareUTFSymbol\textmp{"2213}
8360 \DeclareUTFSymbol\textdotplus{"2214}
8361 \DeclareUTFSymbol\textDivides{"2215}
8362 \DeclareUTFSymbol\textsetminus{"2216}
8363 \DeclareUTFSymbol\textast{"2217}
8364 \DeclareUTFSymbol\textcirc{"2218}
8365 \DeclareUTFSymbol\textbulletoperator{"2219}
8366 \DeclareUTFSymbol\textpropto{"221D}
8367 \DeclareUTFSymbol\textinfty{"221E}
8368 \DeclareUTFSymbol\textangle{"2220}
8369 \DeclareUTFSymbol\textmeasuredangle{"2221}
8370 \DeclareUTFSymbol\textsphericalangle{"2222}
8371 \DeclareUTFSymbol\textmid{"2223}
8372 \DeclareUTFSymbol\textnmid{"2224}
8373 \DeclareUTFSymbol\textparallel{"2225}
8374 \DeclareUTFSymbol\textnparallel{"2226}
8375 \DeclareUTFSymbol\textwedge{"2227}
8376 \DeclareUTFCommand\textowedge{\textcircled\textwedge}
8377 \DeclareUTFSymbol\textvee{"2228}
8378 \DeclareUTFCommand\textovee{\textcircled\textvee}
8379 \DeclareUTFSymbol\textcap{"2229}
8380 \DeclareUTFSymbol\textcup{"222A}
8381 \DeclareUTFSymbol\textint{"222B}
8382 \DeclareUTFSymbol\textiint{"222C}
8383 \DeclareUTFSymbol\textiiint{"222D}
8384 \DeclareUTFSymbol\textoint{"222E}
8385 \DeclareUTFSymbol\textoiint{"222F}
8386 \DeclareUTFSymbol\textointclockwise{"2232}
8387 \DeclareUTFSymbol\textointctrlockwise{"2233}
8388 \DeclareUTFSymbol\texttherefore{"2234}
8389 \DeclareUTFSymbol\textbecause{"2235}
8390 \DeclareUTFSymbol\textvdotdot{"2236}
8391 \DeclareUTFSymbol\textsquaredots{"2237}
8392 \DeclareUTFSymbol\textdotminus{"2238}
8393 \DeclareUTFSymbol\texteqcolon{"2239}
8394 \DeclareUTFSymbol\textsim{"223C}
8395 \DeclareUTFSymbol\textbacksim{"223D}
8396 \DeclareUTFCommand\textnbacksim{\textlstrikethru\textbacksim}
8397 \DeclareUTFSymbol\textwr{"2240}
8398 \DeclareUTFSymbol\textnsim{"2241}
8399 \DeclareUTFSymbol\texteqsim{"2242}
8400 \DeclareUTFCommand\textneqsim{\textlstrikethru\texteqsim}
8401 \DeclareUTFSymbol\textsimeq{"2243}
8402 \DeclareUTFSymbol\textnsimeq{"2244}
8403 \DeclareUTFSymbol\textcong{"2245}
8404 \DeclareUTFSymbol\textncong{"2247}
8405 \DeclareUTFSymbol\textapprox{"2248}
8406 \DeclareUTFSymbol\textnapprox{"2249}
8407 \DeclareUTFSymbol\textapproxeq{"224A}
8408 \DeclareUTFCommand\textnapproxeq{\textlstrikethru\textapproxeq}
8409 \DeclareUTFSymbol\texttriplesim{"224B}
8410 \DeclareUTFCommand\textntriplesim{\textlstrikethru\texttriplesim}
8411 \DeclareUTFSymbol\textbackcong{"224C}
8412 \DeclareUTFCommand\textnbackcong{\textlstrikethru\textbackcong}
8413 \DeclareUTFSymbol\textasymp{"224D}

```

```

8414 \DeclareUTFCommand\textnasymp{\textlstrikeru\textasymp}
8415 \DeclareUTFSymbol\textBumpeq{"224E}
8416 \DeclareUTFCommand\textnBumpeq{\textlstrikeru\textBumpeq}
8417 \DeclareUTFSymbol\textbumpeq{"224F}
8418 \DeclareUTFCommand\textnbumpeq{\textlstrikeru\textbumpeq}
8419 \DeclareUTFSymbol\textdoteq{"2250}
8420 \DeclareUTFCommand\textndoteq{\textlstrikeru\textdoteq}
8421 \DeclareUTFSymbol\textdoteqdot{"2251}
8422 \DeclareUTFCommand\textnDoteq{\textlstrikeru\textdoteqdot}
8423 \DeclareUTFSymbol\textfallingdoteq{"2252}
8424 \DeclareUTFCommand\textnfallingdoteq{\textlstrikeru\textfallingdoteq}
8425 \DeclareUTFSymbol\textrisingdoteq{"2253}
8426 \DeclareUTFCommand\textnrisingdoteq{\textlstrikeru\textrisingdoteq}
8427 \DeclareUTFSymbol\textcolonequals{"2254}
8428 \DeclareUTFSymbol\textequalscolon{"2255}
8429 \DeclareUTFSymbol\texteqcirc{"2256}
8430 \DeclareUTFCommand\textneqcirc{\textlstrikeru\texteqcirc}
8431 \DeclareUTFSymbol\textcirceq{"2257}
8432 \DeclareUTFCommand\textncirceq{\textlstrikeru\textcirceq}
8433 \DeclareUTFSymbol\texthateq{"2259}
8434 \DeclareUTFCommand\textnhateq{\textlstrikeru\texthateq}
8435 \DeclareUTFSymbol\texttriangleeq{"225C}
8436 \DeclareUTFSymbol\textneq{"2260}
8437 \DeclareUTFSymbol\textne{"2260}
8438 \DeclareUTFSymbol\textequiv{"2261}
8439 \DeclareUTFSymbol\textnequiv{"2262}
8440 \DeclareUTFSymbol\textleq{"2264}
8441 \DeclareUTFSymbol\textle{"2264}
8442 \DeclareUTFSymbol\textgeq{"2265}
8443 \DeclareUTFSymbol\textge{"2265}
8444 \DeclareUTFSymbol\textleqq{"2266}
8445 \DeclareUTFCommand\textnleqq{\textlstrikeru\textleqq}
8446 \DeclareUTFSymbol\textgeqq{"2267}
8447 \DeclareUTFCommand\textngeqq{\textlstrikeru\textgeqq}
8448 \DeclareUTFSymbol\textlneqq{"2268}
8449 \DeclareUTFSymbol\textgneqq{"2269}
8450 \DeclareUTFSymbol\textll{"226A}
8451 \DeclareUTFCommand\textnll{\textlstrikeru\textll}
8452 \DeclareUTFSymbol\textgg{"226B}
8453 \DeclareUTFCommand\textngg{\textlstrikeru\textgg}
8454 \DeclareUTFSymbol\textbetween{"226C}
8455 \DeclareUTFSymbol\textnless{"226E}
8456 \DeclareUTFSymbol\textngtr{"226F}
8457 \DeclareUTFSymbol\textnleq{"2270}
8458 \DeclareUTFSymbol\textngeq{"2271}
8459 \DeclareUTFSymbol\textlesssim{"2272}
8460 \DeclareUTFSymbol\textgtrsim{"2273}
8461 \DeclareUTFSymbol\textnlesssim{"2274}
8462 \DeclareUTFSymbol\textngtrsim{"2275}
8463 \DeclareUTFSymbol\textlessgtr{"2276}
8464 \DeclareUTFSymbol\textgtrless{"2277}
8465 \DeclareUTFSymbol\textngtrless{"2278}
8466 \DeclareUTFSymbol\textnlessgtr{"2279}
8467 \DeclareUTFSymbol\textprec{"227A}
8468 \DeclareUTFSymbol\textsucc{"227B}
8469 \DeclareUTFSymbol\textpreccurlyeq{"227C}
8470 \DeclareUTFSymbol\textsucccurlyeq{"227D}
8471 \DeclareUTFSymbol\textprecsim{"227E}
8472 \DeclareUTFCommand\textnprecsim{\textlstrikeru\textprecsim}
8473 \DeclareUTFSymbol\textsuccsim{"227F}
8474 \DeclareUTFCommand\textnsuccsim{\textlstrikeru\textsuccsim}
8475 \DeclareUTFSymbol\textnprec{"2280}
8476 \DeclareUTFSymbol\textnsucc{"2281}
8477 \DeclareUTFSymbol\textsubset{"2282}
8478 \DeclareUTFSymbol\textsupset{"2283}
8479 \DeclareUTFSymbol\textnsubset{"2284}
8480 \DeclareUTFSymbol\textnsupset{"2285}

```

```

8481 \DeclareUTFSymbol\textsubseteq{"2286}
8482 \DeclareUTFSymbol\textsupseteq{"2287}
8483 \DeclareUTFSymbol\textnsubseq{"2288}
8484 \DeclareUTFSymbol\textnsupseq{"2289}
8485 \DeclareUTFSymbol\textsubsetneq{"228A}
8486 \DeclareUTFSymbol\textsupsetneq{"228B}
8487 \DeclareUTFSymbol\textcupdot{"228D}
8488 \DeclareUTFSymbol\textcupplus{"228E}
8489 \DeclareUTFSymbol\textsqsubset{"228F}
8490 \DeclareUTFCommand\textnqssubset{\textlstrikethru\textsqsubset}
8491 \DeclareUTFSymbol\textsqsupset{"2290}
8492 \DeclareUTFCommand\textnqsupset{\textlstrikethru\textsqsupset}
8493 \DeclareUTFSymbol\textsqsubseq{"2291}
8494 \DeclareUTFCommand\textnqssubseq{\textlstrikethru\textsqsubseq}
8495 \DeclareUTFSymbol\textsqsupseq{"2292}
8496 \DeclareUTFCommand\textnqsupseq{\textlstrikethru\textsqsupseq}
8497 \DeclareUTFSymbol\textsqcap{"2293}
8498 \DeclareUTFSymbol\textsqcup{"2294}
8499 \DeclareUTFSymbol\textoplus{"2295}
8500 \DeclareUTFSymbol\textominus{"2296}
8501 \DeclareUTFSymbol\textotimes{"2297}
8502 \DeclareUTFSymbol\textoslash{"2298}
8503 \DeclareUTFSymbol\textodot{"2299}
8504 \DeclareUTFSymbol\textcircledcirc{"229A}
8505 \DeclareUTFSymbol\textcircledast{"229B}
8506 \DeclareUTFSymbol\textcircleddash{"229D}
8507 \DeclareUTFSymbol\textboxplus{"229E}
8508 \DeclareUTFSymbol\textboxminus{"229F}
8509 \DeclareUTFSymbol\textboxtimes{"22A0}
8510 \DeclareUTFSymbol\textboxdot{"22A1}
8511 \DeclareUTFSymbol\textvdash{"22A2}
8512 \DeclareUTFSymbol\textdashv{"22A3}
8513 \DeclareUTFCommand\textndashv{\textlstrikethru\textdashv}
8514 \DeclareUTFSymbol\texttop{"22A4}
8515 \DeclareUTFCommand\textndownvdash{\textlstrikethru\texttop}
8516 \DeclareUTFSymbol\textbot{"22A5}
8517 \DeclareUTFCommand\textnupvdash{\textlstrikethru\textbot}
8518 \DeclareUTFSymbol\textvDash{"22A8}
8519 \DeclareUTFSymbol\textVdash{"22A9}
8520 \DeclareUTFSymbol\textVvdash{"22AA}
8521 \DeclareUTFCommand\textnVdash{\textlstrikethru\textVdash}
8522 \DeclareUTFSymbol\textVDash{"22AB}
8523 \DeclareUTFSymbol\textnvDash{"22AC}
8524 \DeclareUTFSymbol\textnVdash{"22AD}
8525 \DeclareUTFSymbol\textnVdash{"22AE}
8526 \DeclareUTFSymbol\textnVDash{"22AF}
8527 \DeclareUTFSymbol\textlhd{"22B2}
8528 \DeclareUTFSymbol\textrhd{"22B3}
8529 \DeclareUTFSymbol\textunlhd{"22B4}
8530 \DeclareUTFSymbol\textunrhd{"22B5}
8531 \DeclareUTFSymbol\textmultimapdotbothA{"22B6}
8532 \DeclareUTFSymbol\textmultimapdotbothB{"22B7}
8533 \DeclareUTFSymbol\textmultimap{"22B8}
8534 \DeclareUTFSymbol\textveebar{"22BB}
8535 \DeclareUTFSymbol\textbarwedge{"22BC}
8536 \DeclareUTFSymbol\textstar{"22C6}
8537 \DeclareUTFSymbol\textdivideontimes{"22C7}
8538 \DeclareUTFSymbol\textbowtie{"22C8}
8539 \DeclareUTFSymbol\textltimes{"22C9}
8540 \DeclareUTFSymbol\textrtimes{"22CA}
8541 \DeclareUTFSymbol\textleftthreetimes{"22CB}
8542 \DeclareUTFSymbol\textrightthreetimes{"22CC}
8543 \DeclareUTFSymbol\textbacksimeq{"22CD}
8544 \DeclareUTFCommand\textnbacksimeq{\textlstrikethru\textbacksimeq}
8545 \DeclareUTFSymbol\textcurlyvee{"22CE}
8546 \DeclareUTFSymbol\textcurlywedge{"22CF}
8547 \DeclareUTFSymbol\textSubset{"22D0}

```

```

8548 \DeclareUTFCommand\textnSubset{\textlstrikethru\textSubset}
8549 \DeclareUTFSymbol\textSupset{"22D1}
8550 \DeclareUTFCommand\textnSupset{\textlstrikethru\textSupset}
8551 \DeclareUTFSymbol\textCap{"22D2}
8552 \DeclareUTFSymbol\textCup{"22D3}
8553 \DeclareUTFSymbol\textpitchfork{"22D4}
8554 \DeclareUTFSymbol\textlessdot{"22D6}
8555 \DeclareUTFSymbol\textgtrdot{"22D7}
8556 \DeclareUTFSymbol\textl11{"22D8}
8557 \DeclareUTFSymbol\textggg{"22D9}
8558 \DeclareUTFSymbol\textlesseqgtr{"22DA}
8559 \DeclareUTFSymbol\textgtreqless{"22DB}
8560 \DeclareUTFSymbol\textcurlyeqprec{"22DE}
8561 \DeclareUTFCommand\textncurlyeqprec{\textlstrikethru\textcurlyeqprec}
8562 \DeclareUTFSymbol\textcurlyeqsucc{"22DF}
8563 \DeclareUTFCommand\textncurlyeqsucc{\textlstrikethru\textcurlyeqsucc}
8564 \DeclareUTFSymbol\textnpreccurlyeq{"22E0}
8565 \DeclareUTFSymbol\textnsucccurlyeq{"22E1}
8566 \DeclareUTFSymbol\textnqsubsetq{"22E2}
8567 \DeclareUTFSymbol\textnqsupsetq{"22E3}
8568 \DeclareUTFSymbol\textsqsubsetneq{"22E4}
8569 \DeclareUTFSymbol\textsqsupsetneq{"22E5}
8570 \DeclareUTFSymbol\textlnsim{"22E6}
8571 \DeclareUTFSymbol\textgnsim{"22E7}
8572 \DeclareUTFSymbol\textprecnsim{"22E8}
8573 \DeclareUTFSymbol\textsuccnsim{"22E9}
8574 \DeclareUTFSymbol\textntriangleleft{"22EA}
8575 \DeclareUTFSymbol\textntriangleright{"22EB}
8576 \DeclareUTFSymbol\textntrianglelefteq{"22EC}
8577 \DeclareUTFSymbol\textntrianglerighteq{"22ED}
8578 \DeclareUTFSymbol\textvdots{"22EE}
8579 \DeclareUTFSymbol\textcdots{"22EF}
8580 \DeclareUTFSymbol\textudots{"22F0}
8581 \DeclareUTFSymbol\textddots{"22F1}
8582 \DeclareUTFSymbol\textbarin{"22F6}
8583 \DeclareUTFSymbol\textdiameter{"2300}
8584 \DeclareUTFSymbol\textbackneg{"2310}
8585 \DeclareUTFSymbol\textwasylozenge{"2311}
8586 \DeclareUTFSymbol\textinvbackneg{"2319}
8587 \DeclareUTFSymbol\textclock{"231A}
8588 \DeclareUTFSymbol\textulcorner{"231C}
8589 \DeclareUTFSymbol\texturcorner{"231D}
8590 \DeclareUTFSymbol\textllcorner{"231E}
8591 \DeclareUTFSymbol\textlrcorner{"231F}
8592 \DeclareUTFSymbol\textfrown{"2322}
8593 \DeclareUTFSymbol\textsmile{"2323}
8594 \DeclareUTFSymbol\textKeyboard{"2328}
8595 \DeclareUTFSymbol\textlangle{"2329}
8596 \DeclareUTFSymbol\extrangle{"232A}
8597 \DeclareUTFSymbol\textAPLin{"2339}
8598 \DeclareUTFSymbol\textTumbler{"233C}
8599 \DeclareUTFSymbol\textstmaryrdbaro{"233D}
8600 \DeclareUTFSymbol\textnotslash{"233F}
8601 \DeclareUTFSymbol\textnotbackslash{"2340}
8602 \DeclareUTFSymbol\textboxbackslash{"2342}
8603 \DeclareUTFSymbol\textAPLleftarrowbox{"2347}
8604 \DeclareUTFSymbol\textAPLrightarrowbox{"2348}
8605 \DeclareUTFSymbol\textAPLuparrowbox{"2350}
8606 \DeclareUTFSymbol\textAPLdownarrowbox{"2357}
8607 \DeclareUTFSymbol\textAPLinput{"235E}
8608 \DeclareUTFSymbol\textRequest{"2370}
8609 \DeclareUTFSymbol\textBeam{"2393}
8610 \DeclareUTFSymbol\texthexagon{"2394}
8611 \DeclareUTFSymbol\textAPLbox{"2395}
8612 \DeclareUTFSymbol\textForwardToIndex{"23ED}
8613 \DeclareUTFSymbol\textRewindToIndex{"23EE}
8614 \DeclareUTFSymbol\textbbslash{"244A}

```

```
8615 \DeclareUTFSymbol\textCircledA{"24B6}
8616 \DeclareUTFSymbol\textCleaningF{"24BB}
8617 \DeclareUTFCommand\textCleaningFF{\b\textCleaningF}
8618 \DeclareUTFSymbol\textCleaningP{"24C5}
8619 \DeclareUTFCommand\textCleaningPP{\b\textCleaningP}
8620 \DeclareUTFSymbol\textCuttingLine{"2504}
8621 \DeclareUTFSymbol\textUParrow{"25B2}
8622 \DeclareUTFSymbol\textbigtriangleup{"25B3}
8623 \DeclareUTFSymbol\textForward{"25B6}
8624 \DeclareUTFSymbol\texttriangleright{"25B7}
8625 \DeclareUTFSymbol\textRHD{"25BA}
8626 \DeclareUTFSymbol\textDOWNarrow{"25BC}
8627 \DeclareUTFSymbol\textbigtriangledown{"25BD}
8628 \DeclareUTFSymbol\textRewind{"25C0}
8629 \DeclareUTFSymbol\texttriangleleft{"25C1}
8630 \DeclareUTFSymbol\textLHD{"25C4}
8631 \DeclareUTFSymbol\textdiamond{"25C7}
8632 \DeclareUTFSymbol\textlozenge{"25CA}
8633 \DeclareUTFSymbol\textLEFTCIRCLE{"25D6}
8634 \DeclareUTFSymbol\textRIGHTCIRCLE{"25D7}
8635 \DeclareUTFSymbol\textboxbar{"25EB}
8636 \DeclareUTFSymbol\textCloud{"2601}
8637 \DeclareUTFSymbol\textFiveStar{"2605}
8638 \DeclareUTFSymbol\textFiveStarOpen{"2606}
8639 \DeclareUTFSymbol\textPhone{"260E}
8640 \DeclareUTFSymbol\textboxempty{"2610}
8641 \DeclareUTFSymbol\textCheckedbox{"2611}
8642 \DeclareUTFSymbol\textCrossedbox{"2612}
8643 \DeclareUTFSymbol\textCoffeecup{"2615}
8644 \DeclareUTFSymbol\textHandCuffLeft{"261A}
8645 \DeclareUTFSymbol\textHandCuffRight{"261B}
8646 \DeclareUTFSymbol\textHandLeft{"261C}
8647 \DeclareUTFSymbol\textHandRight{"261E}
8648 \DeclareUTFSymbol\textRadioactivity{"2622}
8649 \DeclareUTFSymbol\textBiohazard{"2623}
8650 \DeclareUTFSymbol\textAnkh{"2625}
8651 \DeclareUTFSymbol\textYinYang{"262F}
8652 \DeclareUTFSymbol\textfrownie{"2639}
8653 \DeclareUTFSymbol\textsmiley{"263A}
8654 \DeclareUTFSymbol\textblacksmiley{"263B}
8655 \DeclareUTFSymbol\textsun{"263C}
8656 \DeclareUTFSymbol\textleftmoon{"263D}
8657 \DeclareUTFSymbol\textrightmoon{"263E}
8658 \DeclareUTFSymbol\textmercury{"263F}
8659 \DeclareUTFSymbol\textPUfemale{"2640}
8660 \DeclareUTFSymbol\textearth{"2641}
8661 \DeclareUTFSymbol\textmale{"2642}
8662 \DeclareUTFSymbol\textjupiter{"2643}
8663 \DeclareUTFSymbol\textsaturn{"2644}
8664 \DeclareUTFSymbol\texturanus{"2645}
8665 \DeclareUTFSymbol\textneptune{"2646}
8666 \DeclareUTFSymbol\textpluto{"2647}
8667 \DeclareUTFSymbol\textaries{"2648}
8668 \DeclareUTFSymbol\texttaurus{"2649}
8669 \DeclareUTFSymbol\textgemini{"264A}
8670 \DeclareUTFSymbol\textcancer{"264B}
8671 \DeclareUTFSymbol\textleo{"264C}
8672 \DeclareUTFSymbol\textvirgo{"264D}
8673 \DeclareUTFSymbol\textlibra{"264E}
8674 \DeclareUTFSymbol\textscorpio{"264F}
8675 \DeclareUTFSymbol\textsagittarius{"2650}
8676 \DeclareUTFSymbol\textcapricornus{"2651}
8677 \DeclareUTFSymbol\textaquarius{"2652}
8678 \DeclareUTFSymbol\textpisces{"2653}
8679 \DeclareUTFSymbol\textspadesuitblack{"2660}
8680 \DeclareUTFSymbol\textheartsuitwhite{"2661}
8681 \DeclareUTFSymbol\textdiamondsuitwhite{"2662}
```

```
8682 \DeclareUTFSymbol\textclubsuitblack{"2663}
8683 \DeclareUTFSymbol\textspadesuitwhite{"2664}
8684 \DeclareUTFSymbol\textheartsuitblack{"2665}
8685 \DeclareUTFSymbol\textdiamondsuitblack{"2666}
8686 \DeclareUTFSymbol\textclubsuitwhite{"2667}
8687 \DeclareUTFSymbol\textquarternote{"2669}
8688 \DeclareUTFSymbol\texttwonotes{"266B}
8689 \DeclareUTFSymbol\textsixteenthnote{"266C}
8690 \DeclareUTFSymbol\textflat{"266D}
8691 \DeclareUTFSymbol\textnatural{"266E}
8692 \DeclareUTFSymbol\textsharp{"266F}
8693 \DeclareUTFSymbol\textrecycle{"2672}
8694 \DeclareUTFSymbol\textWheelchair{"267F}
8695 \DeclareUTFSymbol\textFlag{"2691}
8696 \DeclareUTFSymbol\textMineSign{"2692}
8697 \DeclareUTFSymbol\textdsmilitary{"2694}
8698 \DeclareUTFSymbol\textdsmedical{"2695}
8699 \DeclareUTFSymbol\textdsjuridical{"2696}
8700 \DeclareUTFSymbol\textdschemical{"2697}
8701 \DeclareUTFSymbol\textdsbiological{"2698}
8702 \DeclareUTFSymbol\textdscommercial{"269A}
8703 \DeclareUTFSymbol\textmanstar{"269D}
8704 \DeclareUTFSymbol\textdanger{"26A0}
8705 \DeclareUTFSymbol\textFemaleFemale{"26A2}
8706 \DeclareUTFSymbol\textMaleMale{"26A3}
8707 \DeclareUTFSymbol\textFemaleMale{"26A4}
8708 \DeclareUTFSymbol\textHermaphrodite{"26A5}
8709 \DeclareUTFSymbol\textNeutral{"26AA}
8710 \DeclareUTFSymbol\textPUunconfemale{"26B2}
8711 \DeclareUTFSymbol\texthexstar{"26B9}
8712 \DeclareUTFSymbol\textSoccerBall{"26BD}
8713 \DeclareUTFSymbol\textSunCloud{"26C5}
8714 \DeclareUTFSymbol\textRain{"26C6}
8715 \DeclareUTFSymbol\textnoway{"26D4}
8716 \DeclareUTFSymbol\textMountain{"26F0}
8717 \DeclareUTFSymbol\textTent{"26FA}
8718 \DeclareUTFSymbol\textScissorRightBrokenBottom{"2701}
8719 \DeclareUTFSymbol\textScissorRight{"2702}
8720 \DeclareUTFSymbol\textScissorRightBrokenTop{"2703}
8721 \DeclareUTFSymbol\textScissorHollowRight{"2704}
8722 \DeclareUTFSymbol\textPhoneHandset{"2706}
8723 \DeclareUTFSymbol\textTape{"2707}
8724 \DeclareUTFSymbol\textPlane{"2708}
8725 \DeclareUTFSymbol\textEnvelope{"2709}
8726 \DeclareUTFSymbol\textPeace{"270C}
8727 \DeclareUTFSymbol\textWritingHand{"270D}
8728 \DeclareUTFSymbol\textPencilRightDown{"270E}
8729 \DeclareUTFSymbol\textPencilRight{"270F}
8730 \DeclareUTFSymbol\textPencilRightUp{"2710}
8731 \DeclareUTFSymbol\textNibRight{"2711}
8732 \DeclareUTFSymbol\textNibSolidRight{"2712}
8733 \DeclareUTFSymbol\textCheckmark{"2713}
8734 \DeclareUTFSymbol\textCheckmarkBold{"2714}
8735 \DeclareUTFSymbol\textXSolid{"2715}
8736 \DeclareUTFSymbol\textXSolidBold{"2716}
8737 \DeclareUTFSymbol\textXSolidBrush{"2717}
8738 \DeclareUTFSymbol\textPlusOutline{"2719}
8739 \DeclareUTFSymbol\textPlus{"271A}
8740 \DeclareUTFSymbol\textPlusThinCenterOpen{"271B}
8741 \DeclareUTFSymbol\textPlusCenterOpen{"271C}
8742 \DeclareUTFSymbol\textCross{"271D}
8743 \DeclareUTFSymbol\textCrossOpenShadow{"271E}
8744 \DeclareUTFSymbol\textCrossOutline{"271F}
8745 \DeclareUTFSymbol\textCrossMaltese{"2720}
8746 \DeclareUTFSymbol\textDavidStar{"2721}
8747 \DeclareUTFSymbol\textFourAsterisk{"2722}
8748 \DeclareUTFSymbol\textJackStar{"2723}
```

```
8749 \DeclareUTFSymbol\textJackStarBold{"2724}
8750 \DeclareUTFSymbol\textClowerTips{"2725}
8751 \DeclareUTFSymbol\textFourStar{"2726}
8752 \DeclareUTFSymbol\textFourStarOpen{"2727}
8753 \DeclareUTFSymbol\textFiveStarOpenCircled{"272A}
8754 \DeclareUTFSymbol\textFiveStarCenterOpen{"272B}
8755 \DeclareUTFSymbol\textFiveStarOpenDotted{"272C}
8756 \DeclareUTFSymbol\textFiveStarOutline{"272D}
8757 \DeclareUTFSymbol\textFiveStarOutlineHeavy{"272E}
8758 \DeclareUTFSymbol\textFiveStarConvex{"272F}
8759 \DeclareUTFSymbol\textFiveStarShadow{"2730}
8760 \DeclareUTFSymbol\textAsteriskBold{"2731}
8761 \DeclareUTFSymbol\textAsteriskCenterOpen{"2732}
8762 \DeclareUTFSymbol\textEightStarTaper{"2734}
8763 \DeclareUTFSymbol\textEightStarConvex{"2735}
8764 \DeclareUTFSymbol\textSixStar{"2736}
8765 \DeclareUTFSymbol\textEightStar{"2737}
8766 \DeclareUTFSymbol\textEightStarBold{"2738}
8767 \DeclareUTFSymbol\textTwelveStar{"2739}
8768 \DeclareUTFSymbol\textSixteenStarLight{"273A}
8769 \DeclareUTFSymbol\textSixFlowerPetalRemoved{"273B}
8770 \DeclareUTFSymbol\textSixFlowerOpenCenter{"273C}
8771 \DeclareUTFSymbol\textAsterisk{"273D}
8772 \DeclareUTFSymbol\textSixFlowerAlternate{"273E}
8773 \DeclareUTFSymbol\textFiveFlowerPetal{"273F}
8774 \DeclareUTFSymbol\textFiveFlowerOpen{"2740}
8775 \DeclareUTFSymbol\textEightFlowerPetal{"2741}
8776 \DeclareUTFSymbol\textSunshineOpenCircled{"2742}
8777 \DeclareUTFSymbol\textSixFlowerAltPetal{"2743}
8778 \DeclareUTFSymbol\textSnowflakeChevron{"2744}
8779 \DeclareUTFSymbol\textSnowflake{"2745}
8780 \DeclareUTFSymbol\textSnowflakeChevronBold{"2746}
8781 \DeclareUTFSymbol\textSparkle{"2747}
8782 \DeclareUTFSymbol\textSparkleBold{"2748}
8783 \DeclareUTFSymbol\textAsteriskRoundedEnds{"2749}
8784 \DeclareUTFSymbol\textEightFlowerPetalRemoved{"274A}
8785 \DeclareUTFSymbol\textEightAsterisk{"274B}
8786 \DeclareUTFSymbol\textCircleShadow{"274D}
8787 \DeclareUTFSymbol\textSquareShadowBottomRight{"274F}
8788 \DeclareUTFSymbol\textSquareTopRight{"2750}
8789 \DeclareUTFSymbol\textSquareCastShadowBottomRight{"2751}
8790 \DeclareUTFSymbol\textSquareCastShadowTopRight{"2752}
8791 \DeclareUTFSymbol\textDiamondSolid{"2756}
8792 \DeclareUTFSymbol\textRectangleThin{"2758}
8793 \DeclareUTFSymbol\textRectangle{"2759}
8794 \DeclareUTFSymbol\textRectangleBold{"275A}
8795 \DeclareUTFSymbol\textperp{"27C2}
8796 \DeclareUTFCommand\textnotperp{\textlstrikethru\textperp}
8797 \DeclareUTFSymbol\textveedot{"27C7}
8798 \DeclareUTFSymbol\textwedgedot{"27D1}
8799 \DeclareUTFSymbol\textleftspoon{"27DC}
8800 \DeclareUTFSymbol\textlbrackdbl{"27E6}
8801 \DeclareUTFSymbol\textrbrackdbl{"27E7}
8802 \DeclareUTFSymbol\textcirclearrowleft{"27F2}
8803 \DeclareUTFSymbol\textcirclearrowright{"27F3}
8804 \DeclareUTFSymbol\textlongleftarrow{"27F5}
8805 \DeclareUTFSymbol\textlongrightarrow{"27F6}
8806 \DeclareUTFSymbol\textlongleftrightarrow{"27F7}
8807 \DeclareUTFSymbol\textLongleftarrow{"27F8}
8808 \DeclareUTFSymbol\textLongrightarrow{"27F9}
8809 \DeclareUTFSymbol\textLongleftrightarrow{"27FA}
8810 \DeclareUTFSymbol\textlongmapsto{"27FC}
8811 \DeclareUTFSymbol\textLongmapsfrom{"27FD}
8812 \DeclareUTFSymbol\textLongmapsto{"27FE}
8813 \DeclareUTFSymbol\textnwsearrow{"2921}
8814 \DeclareUTFSymbol\textneswarrow{"2922}
8815 \DeclareUTFSymbol\textlhooknwarrow{"2923}
```

```

8816 \DeclareUTFSymbol\texttrhooknearrow{"2924}
8817 \DeclareUTFSymbol\textlhooksearrow{"2925}
8818 \DeclareUTFSymbol\texttrhookswarrow{"2926}
8819 \DeclareUTFSymbol\textleadsto{"2933}
8820 \DeclareUTFSymbol\texttrcurvearrowne{"2934}
8821 \DeclareUTFSymbol\textlcurvearrowse{"2935}
8822 \DeclareUTFSymbol\textlcurvearrowsw{"2936}
8823 \DeclareUTFSymbol\texttrcurvearrowse{"2937}
8824 \DeclareUTFSymbol\textlcurvearrowdown{"2938}
8825 \DeclareUTFSymbol\texttrcurvearrowdown{"2939}
8826 \DeclareUTFSymbol\texttrcurvearrowleft{"293A}
8827 \DeclareUTFSymbol\texttrcurvearrowright{"293B}
8828 \DeclareUTFSymbol\textleftrightharpoon{"294A}
8829 \DeclareUTFSymbol\textrightleftharpoon{"294B}
8830 \DeclareUTFSymbol\textupdownharpoonrightleft{"294C}
8831 \DeclareUTFSymbol\textupdownharpoonleftright{"294D}
8832 \DeclareUTFSymbol\textleftleftharpoons{"2962}
8833 \DeclareUTFSymbol\textupupharpoons{"2963}
8834 \DeclareUTFSymbol\texttrightrightarpoons{"2964}
8835 \DeclareUTFSymbol\textdowndownharpoons{"2965}
8836 \DeclareUTFSymbol\textleftbarharpoon{"296A}
8837 \DeclareUTFSymbol\textbarleftharpoon{"296B}
8838 \DeclareUTFSymbol\textrightbarharpoon{"296C}
8839 \DeclareUTFSymbol\textbarrightharpoon{"296D}
8840 \DeclareUTFSymbol\textupdownharpoons{"296E}
8841 \DeclareUTFSymbol\textdownupharpoons{"296F}
8842 \DeclareUTFSymbol\textllparenthesis{"2987}
8843 \DeclareUTFSymbol\textrrparenthesis{"2988}
8844 \DeclareUTFSymbol\textinvdiameter{"29B0}
8845 \DeclareUTFSymbol\textobar{"29B6}
8846 \DeclareUTFSymbol\textobslash{"29B8}
8847 \DeclareUTFSymbol\textobot{"29BA}
8848 \DeclareUTFSymbol\textNoChemicalCleaning{"29BB}
8849 \DeclareUTFSymbol\textolessthan{"29C0}
8850 \DeclareUTFSymbol\textogreaterthan{"29C1}
8851 \DeclareUTFSymbol\textboxslash{"29C4}
8852 \DeclareUTFSymbol\textboxslash{"29C5}
8853 \DeclareUTFSymbol\textboxast{"29C6}
8854 \DeclareUTFSymbol\textboxcircle{"29C7}
8855 \DeclareUTFSymbol\textboxbox{"29C8}
8856 \DeclareUTFSymbol\textValve{"29D3}
8857 \DeclareUTFSymbol\textmultimapboth{"29DF}
8858 \DeclareUTFSymbol\textshuffle{"29E2}
8859 \DeclareUTFSymbol\textupplus{"2A04}
8860 \DeclareUTFSymbol\textbigdoublewedge{"2A07}
8861 \DeclareUTFSymbol\textbigdoublevee{"2A08}
8862 \DeclareUTFSymbol\textJoin{"2A1D}
8863 \DeclareUTFSymbol\textfatsemi{"2A1F}
8864 \DeclareUTFSymbol\textcircplus{"2A22}
8865 \DeclareUTFSymbol\textminusdot{"2A2A}
8866 \DeclareUTFSymbol\textdottimes{"2A30}
8867 \DeclareUTFSymbol\textdtimes{"2A32}
8868 \DeclareUTFSymbol\textodiv{"2A38}
8869 \DeclareUTFSymbol\textinvneg{"2A3C}
8870 \DeclareUTFSymbol\textsqdoublecap{"2A4E}
8871 \DeclareUTFSymbol\textcapdot{"2A40}
8872 \DeclareUTFSymbol\textsqdoublecup{"2A4F}
8873 \DeclareUTFSymbol\textdoublewedge{"2A55}
8874 \DeclareUTFSymbol\textdoublevee{"2A56}
8875 \DeclareUTFSymbol\textdoublebarwedge{"2A5E}
8876 \DeclareUTFSymbol\textveedoublebar{"2A63}
8877 \DeclareUTFSymbol\texteqdot{"2A66}
8878 \DeclareUTFCommand\textneqdot{\textlstrikethru\texteqdot}
8879 \DeclareUTFSymbol\textcoloncolonequals{"2A74}
8880 \DeclareUTFSymbol\textleqslant{"2A7D}
8881 \DeclareUTFCommand\textnleqslant{\textlstrikethru\textleqslant}
8882 \DeclareUTFSymbol\textgeqslant{"2A7E}

```

```

8883 \DeclareUTFCommand\textngeqslant{\textlstrikethru\textgeqslant}
8884 \DeclareUTFSymbol\textlessapprox{"2A85}
8885 \DeclareUTFCommand\textnlessapprox{\textlstrikethru\textlessapprox}
8886 \DeclareUTFSymbol\textgtrapprox{"2A86}
8887 \DeclareUTFCommand\textngtrapprox{\textlstrikethru\textgtrapprox}
8888 \DeclareUTFSymbol\textlneq{"2A87}
8889 \DeclareUTFSymbol\textgneq{"2A88}
8890 \DeclareUTFSymbol\textlnapprox{"2A89}
8891 \DeclareUTFSymbol\textgnapprox{"2A8A}
8892 \DeclareUTFSymbol\textlesseqqtr{"2A8B}
8893 \DeclareUTFSymbol\textgtreqqlless{"2A8C}
8894 \DeclareUTFSymbol\texteqslantless{"2A95}
8895 \DeclareUTFSymbol\texteqslantgtr{"2A96}
8896 \DeclareUTFSymbol\textleftslice{"2AA6}
8897 \DeclareUTFSymbol\textrightslice{"2AA7}
8898 \DeclareUTFSymbol\textpreceq{"2AAF}
8899 \DeclareUTFCommand\textnpreceq{\textlstrikethru\textpreceq}
8900 \DeclareUTFSymbol\textsucceq{"2AB0}
8901 \DeclareUTFCommand\textnsucceq{\textlstrikethru\textsucceq}
8902 \DeclareUTFSymbol\textprecneq{"2AB1}
8903 \DeclareUTFSymbol\textsucneq{"2AB2}
8904 \DeclareUTFSymbol\textpreceqq{"2AB3}
8905 \DeclareUTFCommand\textnpreceqq{\textlstrikethru\textpreceqq}
8906 \DeclareUTFSymbol\textsucceqq{"2AB4}
8907 \DeclareUTFCommand\textnsucceqq{\textlstrikethru\textsucceqq}
8908 \DeclareUTFSymbol\textprecneqq{"2AB5}
8909 \DeclareUTFSymbol\textsucneqq{"2AB6}
8910 \DeclareUTFSymbol\textprecapprox{"2AB7}
8911 \DeclareUTFCommand\textnprecapprox{\textlstrikethru\textprecapprox}
8912 \DeclareUTFSymbol\textsuccapprox{"2AB8}
8913 \DeclareUTFCommand\textnsuccapprox{\textlstrikethru\textsuccapprox}
8914 \DeclareUTFSymbol\textprecnapprox{"2AB9}
8915 \DeclareUTFSymbol\textsucnapprox{"2ABA}
8916 \DeclareUTFSymbol\textsubsetteqq{"2AC5}
8917 \DeclareUTFCommand\textnsubsetteqq{\textlstrikethru\textsubsetteqq}
8918 \DeclareUTFSymbol\textsupsetteqq{"2AC6}
8919 \DeclareUTFCommand\textnsupsetteqq{\textlstrikethru\textsupsetteqq}
8920 \DeclareUTFSymbol\textdashV{"2AE3}
8921 \DeclareUTFCommand\textndashV{\textlstrikethru\textdashV}
8922 \DeclareUTFSymbol\textDashv{"2AE4}
8923 \DeclareUTFCommand\textnDashv{\textlstrikethru\textDashv}
8924 \DeclareUTFSymbol\textDashV{"2AE5}
8925 \DeclareUTFCommand\textnDashV{\textlstrikethru\textDashV}
8926 \DeclareUTFSymbol\textdownmodels{"2AEA}
8927 \DeclareUTFCommand\textndownmodels{\textlstrikethru\textdownmodels}
8928 \DeclareUTFSymbol\textupmodels{"2AEB}
8929 \DeclareUTFCommand\textnupmodels{\textlstrikethru\textupmodels}
8930 \DeclareUTFSymbol\textupspoon{"2AEF}
8931 \DeclareUTFSymbol\textinterleave{"2AF4}
8932 \DeclareUTFSymbol\textsslash{"2AFD}
8933 \DeclareUTFSymbol\textpentagon{"2B20}
8934 \DeclareUTFSymbol\textvarhexagon{"2B21}
8935 \DeclareUTFSymbol\textjinferior{"2C7C}
8936 \DeclareUTFSymbol\textslashdiv{"2E13}
8937 \DeclareUTFSymbol\textinterrobangdown{"2E18}
8938 \DeclareUTFSymbol\textfivedots{"2E2D}
8939 \DeclareUTFSymbol\textPUheng{"A727}
8940 \DeclareUTFSymbol\textPULhookfour{"A72C}
8941 \DeclareUTFSymbol\textPUscf{"A730}
8942 \DeclareUTFSymbol\textPUaolig{"A735}
8943 \DeclareUTFSymbol\textoo{"A74F}
8944 \DeclareUTFSymbol\textcircumlow{"A788}
8945 \DeclareUTFSymbol\textfi{"FB01}
8946 \DeclareUTFSymbol\textfl{"FB02}
8947 \DeclareUTFSymbol\textGaPa{"1D13B}
8948 \DeclareUTFSymbol\textHaPa{"1D13C}
8949 \DeclareUTFSymbol\textViPa{"1D13D}

```

```

8950 \DeclareUTFSymbol\textAcPa{"1D13E}
8951 \DeclareUTFSymbol\textSePa{"1D13F}
8952 \DeclareUTFSymbol\textZwPa{"1D140}
8953 \DeclareUTFSymbol\textfullnote{"1D15D}
8954 \DeclareUTFSymbol\texthalfnote{"1D15E}
8955 \DeclareUTFSymbol\textVier{"1D15F}
8956 \DeclareUTFSymbol\textAcht{"1D160}
8957 \DeclareUTFSymbol\textSech{"1D161}
8958 \DeclareUTFSymbol\textZwdr{"1D162}
8959 \DeclareUTFSymbol\textMundus{"1F30D}
8960 \DeclareUTFSymbol\textMoon{"1F319}
8961 \DeclareUTFSymbol\textManFace{"1F468}
8962 \DeclareUTFSymbol\textWomanFace{"1F469}
8963 \DeclareUTFSymbol\textFax{"1F4E0}
8964 \DeclareUTFSymbol\textFire{"1F525}
8965 \DeclareUTFSymbol\textBicycle{"1F6B2}
8966 \DeclareUTFSymbol\textGentsroom{"1F6B9}
8967 \DeclareUTFSymbol\textLadiesroom{"1F6BA}
8968 \DeclareUTFCommand\textcopyleft{\textcircled\textrevc}
8969 \DeclareUTFCommand\textccsa{\textcircled\textcirclearrowleft}
8970 \DeclareUTFSymbol\textglqq{"201E}
8971 \DeclareUTFSymbol\textgrqq{"201C}
8972 \DeclareUTFSymbol\textglq{"201A}
8973 \DeclareUTFSymbol\textgrq{"2018}
8974 \DeclareUTFSymbol\textflqq{"00AB}
8975 \DeclareUTFSymbol\textfrqq{"00BB}
8976 \DeclareUTFSymbol\textflq{"2039}
8977 \DeclareUTFSymbol\textfrq{"203A}
8978 \DeclareUTFSymbol\textneg{"00AC}
8979 \DeclareUTFSymbol\textcdot{"00B7}

8980 </xunextra>
8981 <@=@=xeCJK>

```

5.22 xeCJK.cfg

```
8982 <*config>
```

预设的配置文件 `xeCJK.cfg` 为一个空文件。可以在里面增加设置，然后保存到本地目录下面。

```
8983
```

```
8984 </config>
```

版本历史

v3.1.0	(2012/11/13 – 2012/11/21)	<code>\xeCJK@family</code> : 不将参数完全展开。	108
General:	放弃对 <code>\outer</code> 宏的特殊处理。	<code>_xeCJK_check_single_space:NN</code> : 使用	
	放弃使用放缩字体大小的方式, 而只采用调整间距的方式与西文等宽字体对齐。并且只适用于与抄录环境下。	<code>\xeCJK_if_CJK_class:NTF</code> 来代替 <code>\int_case:nnn</code> 判断是否是 CJK 字符类。	62
	改用 <code>indentfirst</code> 宏包处理缩进的问题。	<code>_xeCJK_family_unknown_warning:n</code> : 在没有定义任何 CJK 字体的情况下, 不再重复给出字体没有定义的警告。	91
	取消 <code>\protect</code> 的外部宏限制。	v3.2.0	(2013/04/14 – 2013/05/22)
	删除多余的 <code>default-itcorr</code> 结点。	General:	增加 IVS 字符类用于处理异体字选择符。
	使用 <code>xtemplate</code> 宏包的机制来组织标点符号的处理。		增加 Verb 选项。
LocalConfig:	增加 LocalConfig 选项用于载入本地配置文件。		<code>\setCJKmonofont</code> : 定义中加入 <code>\normalfont</code> 。
<code>\xeCJK@fix@penalty</code> :	采用通过不修改原语 <code>\</code> 的方式对修复倾斜校正。		<code>_xeCJK_Boundary_and_FullLeft_glue:N</code> : 当全角左标点前面是 <code>hlist</code> 、 <code>none</code> 、 <code>glue</code> 和 <code>penalty</code> 等节点时, 压缩其左空白。
<code>_xeCJK_fallback_loop:nmN</code> :	调整备用字体的循环方式。		<code>\l_xeCJK_family_tl</code> : 不将其初始化为 <code>\CJKfamilydefault</code> 。
<code>\xeCJK_glyph_if_exist:N</code> :	改进 <code>fontspec</code> 宏包中定义的 <code>\font_glyph_if_exist:NnTF</code> 。		<code>\xeCJK_FullLeft_and_Default::</code> 修正 <code>xeCJK</code> 使西文在部分情况下无法断词的问题。
<code>\xeCJK_hook_for_ulem::</code>	简化对 <code>ulem</code> 宏包的兼容补丁。		<code>\c_xeCJK_space_skip_tl</code> : 字间空格考虑到 <code>\spacefactor</code> 和 <code>\xspaceskip</code> 的情况。
<code>\c_xeCJK_space_skip_tl</code> :	字间空格考虑 <code>\spaceskip</code> 不为零的情况。	v3.2.1	(2013/05/29)
<code>_xeCJK_switch_font:nm</code> :	改进定义, 加快切换速度。	General:	调整 Verb 选项: 在命令 <code>\verb</code> 里使用时, 不破坏标点禁则, 增加值 <code>env+</code> 。
<code>\xeCJKVerbAddon</code> :	新增 <code>\xeCJKVerbAddon</code> 用于抄录环境中的间距调整。	v3.2.2	(2013/05/30 – 2013/06/04)
v3.1.1	(2012/12/02 – 2012/12/13)	General:	修正某些重音不能正确显示的问题。
General:	不再依赖 <code>xpatch</code> 宏包。		增加小宏包 <code>xeCJK-listings</code> , 用于支持 <code>listings</code> 宏包。
	对于与 <code>xltxtra</code> 的冲突给出错误警告。		<code>_xeCJK_ulem_CJK_and_FullRight_glue:N</code> : 修正下划线不能跳过全角右标点的问题。
	增加 <code>NewLineCS</code> 和 <code>EnvCS</code> 选项。	v3.2.3	(2013/06/04 – 2013/06/11)
	增加小宏包 <code>xeCJKintef</code> , 用于处理下划线的问题。	General:	不再改变 CJK 字符类的 <code>\catcode</code> 。
CheckFullRight:	处理全角右标点之后的断行问题。		根据 <code>X_gTeX</code> 的脚本重新整理全角标点符号。
InlineEnv:	改变行内环境的设置方式, 从而使用 <code>\str_case_x:nnn</code> 代替原来的 <code>\clist_if_in:NnTF</code> 来判断是否是行内环境。		解决 <code>CheckSingle</code> 选项与 <code>tablists</code> 宏包的冲突。
PlainEquation:	增加 <code>PlainEquation</code> 选项。		提供四个 <code>TECkit</code> 映射文件用于句号转换和简繁互换。
<code>\xeCJK@family</code> :	修改主要 CJK 字体族的自动更新方式。		完善对 <code>listings</code> 宏包的支持。
<code>_xeCJK_check_single_aux:nmNw</code> :	改进定义, 减少使用 <code>peek</code> 函数的次数。		<code>_xeCJK_listings_initial_hook::</code> 解决 <code>listings</code> 环境中代码行号输出不正确的问题, 并解决在其中跨页时对页眉和页脚的影响。
<code>_xeCJK_check_single_space:NN</code> :	<code>CheckSingle</code> 支持段末“汉字 + 汉字 + 空格 + 汉字/标点”的形式。		<code>_xeCJK_listings_process_Default:nN</code> : 在 <code>listings</code> 环境中对 <code>\charcode</code> 大于 255 的字符根据其 <code>\catcode</code> 区分 <code>letter</code> 和 <code>other</code> 。
<code>\xeCJK_hook_for_ulem::</code>	完全处理下划线里的标点符号的有关问题。		<code>_xeCJK_restore_shipout_CJKsymbol::</code> 解决 <code>\CJKunderdot</code> 跨页使用时影响到页眉页脚的问题。
<code>\xeCJK_peek_catcode_ignore_spaces:NTF</code> :	新增有省略空格标识的 <code>peek</code> 函数。		<code>_xeCJK_ulem_FullLeft_and_CJK::</code> 修正全角左标点下划线与 <code>\CJKunderdot</code> 连用时结果不正常的问题。
<code>\xeCJK_save_class:nm</code> :	使用 <code>\xeCJK_save_class:nm</code> 保存 <code>X_gTeX</code> 预定义的字符类别。		<code>\xeCJKVerbAddon</code> : 新增 <code>\xeCJKOffVerbAddon</code> 用于局部取消 <code>\xeCJKOffVerbAddon</code> 的影响; 并解决跨页使用时影响到页眉页脚的问题。
<code>\xeCJK_set_char_class:nm</code> :	在文档中设置字符类别时不重复设置 <code>\catcode</code> 。	v3.2.4	(2013/06/23 – 2013/07/06)
<code>_xeCJK_set_char_class_eq:nm</code> :	交换参数的顺序。	General:	不再使用 <code>CJKnumber</code> 选项, 可以在 <code>xeCJK</code> 之后直接使用 <code>CJKnumb</code> 宏包得到中文数字。
<code>_xeCJK_set_verb_exspace::</code>	调整间距的计算方法。		改进获取分区字体属性的办法。
<code>\xeCJKnobreak</code> :	增加 <code>\nobreak</code> 的 <code>xeCJK</code> 版本。		解决使用 <code>CheckSingle</code> 时, 某些 <code>\CJKglue</code> 不能被正确加入的问题。
v3.1.2	(2012/12/27 – 2013/01/01)		尽量移除用作判断标志的 <code>\kern</code> 。
General:	解决在下划线状态下使用 <code>\makebox</code> 时的错误。		内部调整分区字体的设置方法。
	修正非 <code>\UTFencname</code> 编码下面 <code>xunicode</code> 重定义的 <code>\nobreakspace</code> 会失效的问题。		
	修正重定义 <code>\CJKfamilydefault</code> 无效的问题, 恢复容错能力。		

使 listings 的 breaklines 选项对 CJK 字符类可用, 并保持标点符号的禁则。	141	修正 unicode-letters.tex 中谚文符号 \catcode 不准的问题。	34
使用 AllowBreakBetweenPuncts 时, 相应标点符号仍能与边界对齐。	51	\Url@MathSetup: 使通过 \UrlFont 等命令设置的 CJK 字体生效。	106
修正 xeCJKfntef 与 natbib 等的冲突。	121	_xeCJK_check_single_aux:nNNw: 与 \CJKspace 兼容。	61
遵循 L ^A T _E X3 变量需要预先声明的原则。	1	_xeCJK_punct_glue:NN: 标点符号左/右空白的伸展值不超过原始边界, 收缩值不小于另一侧边界。	51
\addCJKfontfeatures: 可以单独增加当前各个分区字体的属性。	93	\xeCJK_set_mathfont:: 将 CJK 字符的数学归类由 7 改为 0, 解决汉字路径的问题。	96
CJKfilltwosides: 改用 minipage 和 L ^A T _E X 表格 (tabular) 来实现。	138	v3.2.8 (2013/11/16 – 2013/12/05)	
_xeCJK_Boundary_and_FullLeft_glue:N: 细化边界与全角左标点之间是否压缩空白的判断。	53	General: 启用 xunicode 中的带圈数字和字母设置。	157
_xeCJK_fallback_loop:nmN: 使 \CJKfamilydefault 的 FallBack 设置全局可用。	80	\DeclareUTFmathsymbols: 修正 \UseMathAsText 的功能, 恢复 \hbar 和增加以 text 打头的文本符号命令。	146
_xeCJK_set_verb_exspace:: 当计算得出的间距为负时, 缩小 CJK 字体。	101	_xeCJK_nobreak_skip:: 禁止在 \verb 中断行。	99
\xeCJK_tl_remove_outer_braces:n: 去掉外层分组括号时, 移除空格, 避免死循环。	22	\xeCJKVerbAddon: 增加是否是等宽字体的判断。	100
\xeCJK_token_value_charcode:N: 考虑 charcode 超出 BMP 的情况。	25	v3.2.9 (2013/12/07 – 2013/12/08)	
v3.2.5 (2013/07/10 – 2013/07/25)		General: 文档部分增加 xunicode 定义的符号表。	145
General: 恢复 \nobreakspace 的原始定义。	108	增加 xunicode-extra.def 中, 用于加入 puenc.def 中的符号定义。	160
解决 fixltx2e 和 amsthm 的冲突。	108	\DeclareEncodedCompositeAccents: 修正 xunicode 中的错误定义。	150
修正 CJK 和 NormalSpace 字符类之间因为边界造成的间距不正确的问题。	39	\c__xeCJK_middle_dot_prop: 完整处理 encguide.pdf 的编码符号表中, 与旧编码的 U+00B7 冲突。	112
增加小宏包 xunicode-addon, 为 xunicode 提供判断字符是否存在的功能。	145	v3.2.10 (2014/02/20 – 2014/03/01)	
\@setupverbvisiblespace: 可视空格考虑传统 T _E X 字体的情况。	102	\CJKaddEncHook: 使用 CJKnumb 时, 让 \Unicode 有定义。	120
Verb: 微调定义。	98	\DeclareUTFDoubleEncodedAccent: 改进 \t 等的定义方式。	151
_xeCJK_Boundary_and_FullLeft_glue:N: 细化全角左标点是否位于段首的判断。	53	\DeclareUTFDoubleEncodedSymbol: 改进 \sliding 等的定义方式。	151
增加对 enumitem 宏包修改的 \item 的判断。	53	\DeclareUTFTIPACommand: 检查 \t 和 \sliding 的参数是否以 \textipa 开头。	155
_xeCJK_math_robust:N: 解决汉字后紧跟 \(\dots\) 形式的行内数学公式时, 不能加入间距的问题。	107	LoadFandol: 当没有设置字体时, 使用 Fandol 字体系列。	94
\xeCJKVerbAddon: 禁止自动换行, 与西文一致。	100	v3.2.11 (2014/03/14 – 2014/04/10)	
v3.2.6 (2013/07/29 – 2013/08/15)		General: 删除 \xeCJKcaption。	120
General: AutoFakeBold 和 AutoFakeSlant 选项直接使用 fontspec 的设置, 修正不能调用相应实际字体的问题。	83	左右角括号 U+2329 和 U+232A 是西文标点符号。	27
case 类函数的用法与 L ^A T _E X3 同步。	1	\CJK@family: 引入 \CJK@family 保存实际的字体族名。	90
为 \mathrm 减少一个可能的数学字体族。	107	indentfirst: 放弃 indentfirst 和 CJKnumber 选项。	104
\AtEndUTFCommand: 可以指定特定符号命令使用的钩子。	154	\xeCJK_add_to_shipout:n: 不再使用内部名字。	22
\xeCJK_CJK_and_Boundary:w: 更好的处理边界是 \relax 的情况。	46	v3.2.12 (2014/05/12)	
_xeCJK_math_robust:N: 考虑 ulem 对 \MakeRobust 的不当定义。	107	General: 更新 \int_to_Hex:n。	80
考虑 \math 和 \ensuremath。	107	新增 RubberPunctSkip 选项。	66
\xeCJK_set_mathfont:: 设置粗体时先检查对应字体是否存在。	96	v3.2.13 (2014/06/02 – 2014/06/20)	
v3.2.7 (2013/08/22 – 2013/11/09)		General: 自动调整 \CJKfamilydefault 时, 只将 \familydefault 展开一次。	94
General: 标点宽度设置禁用比例选项的值改为 nan。	74	\xeCJK_set_mathfont:: 修复参数类型错误。	96
处理 AllowBreakBetweenPuncts 与 xeCJKfntef 的兼容问题。	51	v3.2.14 (2014/10/31 – 2014/11/03)	
实现自定义行首/尾标点符号宽度功能。	66	General: xeCJKfntef 不再依赖 CJKfntef。	121
使用 everypage 往 \shipout 盒子里加钩子。	21	解决下划线前后没有 \CJKglue 或 \CJKecglue 的问题。	121
		完善 \varCJKunderline 的实现。	121
		v3.2.15 (2014/11/07 – 2014/11/10)	
		General: xeCJKfntef 增加 hidden 选项。	121
		把 REVERSE SOLIDUS(U+005C)、HYPHEN-MINUS (U+002D)和 EN DASH(U+2013)归入 NormalSpace 类。	27
		增加 HangulJamo 字符类。	27

\CJKunderanyline: 完善选项。	133	v3.4.1	(2016/05/21 – 2016/08/18)
_xeCJK_listings_initial_hook:: 修正 breaklines 无效的问题。	139	General: 补充 Unicode 9.0 的西夏文。	30
\xeCJKfntefon: 完善选项。	132	新的下划线选项 textformat。	121
v3.2.16	(2014/11/20 – 2014/12/16)	修复 CJKspace 功能失效。	45
General: 不再依赖 everypage 宏包。	21	v3.4.2	(2016/10/19)
修复 \hbar。	106	General: 避免在破折号之间折行。	37
整理 xCJKecglue 的部分代码。	44	\xeCJK_clear_Boundary_and_CJK_toks:: 提高效率, 避免重复循环。	36
v3.3.0	(2014/12/26)	v3.4.3	(2016/10/27 – 2016/11/18)
General: 不把 NS 类中的一些有禁则的日文归入 FullRight 类。	28	\CJKfontspec: 允许字体属性可选项在后的新语法。	92
不把小写日文假名归入 FullRight 类。	29	\setCJKfallbackfamilyfont: 允许字体属性可选项在后的新语法。	81
\c_xeCJK_PR_chars_clist: 不把 U+20A9 归入 CJK 的 PR 类。	28	\setCJKmathfont: 允许字体属性可选项在后的新语法。	96
v3.3.1	(2015/01/22 – 2015/05/08)	\setCJKmonofont: 允许字体属性可选项在后的新语法。	91
General: IVS 字符类更名为 CM。	27	_xeCJK_long_punct_kerning:N: 考虑破折号边界为负值的情况。	71
删去 fixltx2e 和 amsthm 的冲突补丁。	108	v3.4.4	(2016/11/30)
新选项 WidowPenalty。	61	General: 不压缩长标点与其他标点的间距。	74
\CJKaddEncHook: 应用 0.99992 版的新原语 \Ucharcat。	120	v3.4.5	(2017/01/02)
LoadFandol: 为方便 MacTeX 用户, Fandol 字体改用文件名。	94	General: 更新 \LaTeX 3 的过时用法。	107
\xeCJK_check_single_cs:Nn: 补充可能遗漏的空格。	63	v3.4.6	(2017/02/23)
\c_xeCJK_CM_chars_clist: 补充音调符号。	32	\xeCJK@family: 将族名参数完全展开, 以解决与 fontspec 2017/01/24 v2.5d 的兼容问题。	108
_xeCJK_listings_initial_hook:: 解决 prebreak 和 postbreak 功能失效的问题。	139	v3.4.7	(2017/03/20)
_xeCJK_listings_process_Default:nN: 对 listings 的字符扩展不影响到其符号表中的七位或八位字符。	140	General: 简化 CheckSingle 的实现, 不再展开宏。	61
_xeCJK_math_robust:N: 兼容 \LaTeX 2 _ε 2015。	107	v3.4.8	(2017/05/15)
\xeCJK_token_value_charcode:N: 0.99992 版修复了 \meaning 的 Bug。	25	General: 转义 \lstineline 参数中的 λ_{12} 。	144
\g_xeCJK_xetex_allocator_int: 兼容 \LaTeX 2 _ε 2015。	109	v3.5.0	(2017/07/19 – 2017/07/22)
v3.3.2	(2015/05/15)	General: 补充 Ext-F。	30
General: 随 Unicode 7.0 更新简繁体汉字映射。	1	常数 \c_minus_one 已过时。	1
\g_xeCJK_xetex_allocator_int:		使用 lazy 函数对 Boolean 表达式进行最小化运算 (\LaTeX 3 2017/07/19)。	1
\xe@alloc@intercharclass 总是有定义的。	109	v3.5.1	(2017/11/16)
v3.3.3	(2015/05/30 – 2016/02/01)	General: 修正 fallback 字体后无法忽略空格的错误。	80
General: 把 EN DASH (U+2013) 作为半字线连接号归入 FullRight 类。	28	v3.6.0	(2018/01/13 – 2018/01/24)
补充 Ext-E。	30	General: Default 类与 MiddlePunct 之间不应该有 \CJKglue。	57
不再把 U+2015 和 U+2500 归入 FullRight 类。	28	把 TWO-EM DASH (U+2E3A) 归入 FullRight 类和设为 LongPunct 与 MiddlePunct。	28
更新 \LaTeX 3 代码。	1	将全角浪线 U+FF5E 等连接号归入 FullRight 类和设为 MiddlePunct。	28
兼容 \LaTeX 2 _ε 2016/02/01 的字符类设置。	27	解决标点中间被隔开的禁则与压缩问题。	55
解决与 microtype 宏包的兼容问题。	120	同步 \LaTeX 3 2017/12/16。	1
使用新的 Unicode 编码名称 TU。	145	新增 PunctFamily 选项支持对汉字标点单独切换字体。	90
CJKfilltwosides: 确保进入水平模式。	138	修正标点同为 LongPunct 与 MiddlePunct 时的实现错误。	57
v3.3.4	(2016/02/07)	总允许长标点与其他标点之间折行。	51
General: 兼容 \XeTeX 0.99994 的边界字符类。	27	v3.6.1	(2018/02/25 – 2018/02/27)
v3.4.0	(2016/05/01 – 2016/05/13)	General: 减少 bool 运算。	1
General: RubberPunctSkip 选项有新的值 plus 和 minus。	66	\xeCJK_if_last_punct:TF: 细化判断。	55
CJKmath 功能也支持分区字体。	96	v3.7.0	(2018/03/12 – 2018/03/18)
标点符号的压缩量能伸长到原始空白, 能收缩到较小边距。	75	General: 补充定义 \texthyphenationpoint 和 \texttwoemdash。	157
改进 xCJKecglue 的实现。	38	不再默认引入 xunicode 宏包。	105
\xeCJK_set_mathfont:: CJKmath 的字符范围遵从 \xeCJKDeclareCharClass 的设置。	96	对 \nobreakspace 的恢复放到 xunicode-addon 中处理。	108

- 修正长标点被隔开时的压缩处理错误。..... 71
- v3.7.1** (2018/04/30)
`\AtEndUTFCommand`: 修复代码重构而引入的新错误。... 154
- v3.7.2** (2018/05/02 – 2019/04/07)
 General: 改用 `xparse` 的新参数类型 `b` 定义
`CJKfilltwosides*` 环境, 不再依赖 `environ` 包。..... 138
 简化 `CJKspace` 的实现, 并修复错误。..... 45
 解决与 `microtype` 宏包的兼容问题。..... 156
 删除定义新字体族时过滤重复选项的功能。..... 84
 同步 L^AT_EX3 2019/03/05。..... 1
`\xeCJK_FullLeft_and_Default::` 再次修正 `FullLeft` 类
 字符与西文连用断词失败的问题。..... 52
`_xeCJK_patch_tuenc_composite::` 修复补丁错误。... 112
- v3.7.3** (2019/04/15)
 General: 补充日文假名扩展。..... 30
 修复 `penalty` 数值错误。..... 55
- v3.7.4** (2019/05/31)
 General: 简化行首/尾标点符号宽度的实现。..... 66
- v3.8.0** (2020/02/09 – 2020/02/10)
 General: 兼容 L^AT_EX 2_ε 2020/02/02 对 `NFSS` 的修改。..... 1
 清理过时的兼容性补丁代码。..... 1
 删除 `CJKfntef` 补丁。..... 120
 删除 `realscripts` 补丁。..... 108
 删除 `\hbar` 补丁。..... 106
 删除 `\mathrm` 补丁。..... 107
 应用 `\fp_if_nan:nTF`。..... 74
 应用 `\peek_remove_spaces:n`。..... 24
`\@setupverbvisiblespace`: 更新可视空格补丁。..... 102
- v3.8.1** (2020/02/14)
 General: 修复 `\l_xeCJK_current_font_tl` 标记错误。... 87
 应用 `\shapedefault`。..... 97
- v3.8.2** (2020/02/17)
 General: 避免导言区字体警告。..... 92
 修复分区字体错误。..... 88
- v3.8.3** (2020/03/15 – 2020/04/27)
 General: `hidden` 选项保留原内容的高度和深度。..... 125
 补充 `U+02EA` 和 `U+02EB`。..... 30
 兼容 `unicode-math` 和 `CJKmath` 选项。..... 114
 取消 `xeCJKfntef` 的初始彩色设置。..... 136
 删除 `_nopar`。..... 1
 同步 Unicode 13.0。..... 30
 修复 `xCJKecglue` 选项。..... 55
 依赖 `ctexhook` 宏包。..... 21
 重构 `PunctStyle` 选项, 完全展开参数。..... 77
- v3.8.4** (2020/05/31 – 2020/06/04)
 General: 重构后备字体的实现, 修正标点符号无后备字体
 的问题。..... 79
`\xeCJK_font_gset_to_current:N`: 不缓存 `\nullfont`。... 22
- v3.8.5** (2020/06/25 – 2020/06/26)
 General: 进一步兼容 `microtype`。..... 114
`_xeCJK_bound_type_1_glue:Nn`: 增加盒子高度判断。... 53
- v3.8.6** (2020/10/17 – 2020/10/18)
 General: 兼容 L^AT_EX 2020/10/01 的 `NFSS` 钩子机制。... 108
 正确还原标点符号后的 `penalty` 状态。..... 55
- v3.8.7** (2021/06/04 – 2021/06/09)
 General: 更好地兼容 `CJKnumb`。..... 120
- 将 `CJKfntef` 包替换为 `xeCJKfntef` 包。..... 20
 应用 `\disable@package@load` 和
`\declare@file@substitution`。..... 20
- v3.8.8** (2021/09/15 – 2021/09/16)
 General: 补充女书。..... 30
 同步 Unicode 14.0。..... 30
- v3.8.9** (2022/05/26)
 General: 修正居中标点悬挂错误。..... 73
`_xeCJK_bound_type_1_glue:Nn`: 增加位于段首的支架
 盒子判断。..... 53
- v3.9.0** (2022/06/06 – 2022/07/08)
 General: 不直接依赖 `xparse` 和 `l3keys2e`。..... 20
 修复西文的 `character protrusion` 功能。..... 44
- v3.9.1** (2022/07/28 – 2022/08/02)
 General: 简化部分内部实现。..... 42
 修复下划线中数学公式的错误处理。..... 126
- v3.10.0** (2022/08/31 – 2026/06/23)
 General: `NoBreakLongPunct` 在右侧时禁止折行。..... 51
`\reset@color` 补丁增加 `hlist` 回退路径, 当 `color pop` 后
 最后节点为 `hlist` 时设置
`\g_xeCJK_reset_color_pending_bool`, 延迟到
`_xeCJK_check_for_glue_skip`: 中处理 `\colorbox`
 等命令右侧间距 (#831)。..... 115
 补丁 `l3color` 后端的 `_color_select:N` 和
`_color_backend_reset::`, 使 `l3color` 接口的颜色切换
 也能正确保持 `xeCJK` 间距 (#832)。..... 116
 补丁 `\reset@color` 以在颜色弹出 `whatsit` 之后重新放
 置 `xeCJK` 节点标记, 修复 `\textcolor` 命令右侧多余
`inter-word glue` 的问题 (#831)。..... 115
 撤回对 `\char` 的重定义, 改为提供 `\xeCJKchar` 新命令,
 避免破坏依赖
`let`
 保存 `\tn {char}` 原语的宏包 (\#800)。 `hdclindex5886113`
 改正拼写错误 `_xeCJK_glue_node:n`。..... 43
 将全角浪线等连接号从 `MiddlePunct` 改为 `LongPunct`,
 消除不必要的标点压缩间距。..... 28
 使用 `\l_keys_key_str` 和 `\l_keys_choice_str` 替代
 已废弃的 `_tl` 版本 (#806)。..... 115
 提升 L^AT_EX3 最低版本要求至 2025/10/09。..... 20
 提升版本号至 v3.10.0。..... 1
 通过在 `\Hy@BeginAnnot` 中保存并选择性恢复 `xeCJK`
 节点标记, 同时清除旧标记, 解决目录中链接注释起始处
 的虚假 `ecglue` (#810), 并为 `\ref` 提供前侧 `ecglue`
 (#809)。..... 115
 同步 Unicode 15.0。..... 30
 同步 Unicode 15.1。..... 30
 同步 Unicode 17.0。..... 30
 为 `color/xcolor` 添加兼容补丁, 在颜色切换 `whatsit` 后重
 放 `xeCJK` 节点标记, 修复 `\textcolor` 后 `CJKecglue` 丢
 失的问题 (#315, #803)。..... 115
 为 `hypdoc` 的 `\HD@target` 添加补丁, 在它产生的 `hbox`
 之后重放 `xeCJK` 节点标记, 修复 `l3doc` 中 `\cs\meta` 等
 命令后 `CJKecglue` 丢失或保留为原始空格的问题
 (#873)。..... 117
 为 `mtpro2` 提供兼容补丁, 使大花括号内部的 `\char` 不
 被 `interchar` 拦截 (#407)。..... 113

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

\\ 5, 6, 7, 13, 14, 15, 16, 30, 365, 366
 \ <font-switch> 7

A

\addCJKfontfeatures 8
 AllowBreakBetweenPuncts 5
 \AtBeginDocument 69
 \AtEndOfPackage 425, 702, 715
 AutoFakeBold 4, 6
 AutoFakeSlant 4, 6
 AutoFallBack 4

B

bool commands:
 \bool_gset_false:N 883, 900, 906, 922
 \bool_gset_true:N 1233, 1373
 \bool_if:NTF 101, 319, 881,
 899, 904, 921, 967, 1215, 1221, 1260, 1264, 1318, 1325, 1371
 \bool_lazy_or:nnTF 549
 \bool_new:N
 ... 52, 110, 269, 700, 1125, 1126, 1189, 1209, 1315, 1364
 \bool_set_false:N 244, 1163, 1201
 \bool_set_true:N 103, 251, 696, 1152, 1174, 1195
 Boundary 368
 box commands:
 \box_new:N 50
 \box_wd:N 223
 boxdepth 15

C

char commands:
 \char_generate:nn 428
 \char_set_catcode_ignore:n 331, 332, 333, 334, 335
 \char_set_catcode_letter:n 567
 CheckFullRight 5
 CheckSingle 3
 CJK 368
 \CJKecglue 744, 802, 1217, 1253, 1261
 CJKecglue 3, 1142
 \CJKfamily 7
 \CJKfamilydefault 8
 CJKfilltwosides 17
 \CJKfontspec 7
 \CJKglue 887, 891,
 928, 936, 950, 954, 970, 1009, 1104, 1110, 1137, 1203, 1289
 CJKglue 3, 1133
 CJKmath 3
 \CJKrmdefault 8
 \CJKsfdefault 8
 \CJKsout 14

CJKspace 3, 1190
 \CJKsymbol 735, 842, 1291, 1297, 1303
 \CJKttdefault 8
 \CJKkunderanyline 16
 \CJKkunderanysymbol 16
 \CJKkunderdblline 14
 \CJKkunderdot 15
 \CJKkunderline 14
 \CJKkunderwave 14
 \CJKxout 14

clist commands:

\clist_const:Nn 392, 394, 396, 397, 404, 406,
 411, 419, 421, 423, 433, 435, 436, 445, 446, 455, 505, 512
 \clist_gclear:N ... 594, 595, 596, 597, 605, 606, 607, 608
 \clist_gconcat:NNN 532, 546
 \clist_map_function:NN 430
 \clist_map_function:nn 691
 \clist_map_inline:Nn 534, 584
 \clist_map_inline:nn .. 33, 38, 726, 739, 786, 1307, 1378
 \clist_new:N 54, 345, 357
 \clist_set:Nn 527

CM 387

cs commands:

\cs_end: 307
 \cs_generate_variant:Nn 546, 621, 634, 640
 \cs_gset_eq:NN 130, 141
 \cs_gset_protected:Npe 669
 \cs_if_eq:NNTF 141
 \cs_if_exist:NTF 44, 94
 \cs_if_exist_use:NTF 1037
 \cs_new:Npn 111, 116,
 200, 270, 272, 279, 285, 296, 310, 361, 518, 622, 1064, 1066
 \cs_new_eq:NN
 266, 267, 268, 362, 701, 766, 819, 972, 1127, 1187
 \cs_new_protected:Npe 1128
 \cs_new_protected:Npn 55, 56, 57, 58, 59, 60, 61, 62, 63,
 64, 66, 72, 74, 76, 78, 80, 82, 84, 91, 107, 127, 129, 131,
 138, 140, 209, 212, 228, 233, 237, 248, 318, 338, 351,
 525, 530, 547, 571, 581, 616, 627, 629, 635, 641, 654,
 665, 667, 683, 693, 746, 748, 754, 767, 806, 808, 814,
 844, 862, 879, 897, 914, 919, 925, 945, 959, 965, 973,
 988, 1008, 1042, 1049, 1059, 1068, 1073, 1083, 1099,
 1211, 1241, 1243, 1249, 1256, 1280, 1287, 1337, 1355, 1366
 \cs_set:Npn 275, 427, 1011, 1017
 \cs_set_eq:NN
 ... 128, 133, 134, 135, 239, 1153, 1154, 1155, 1156,
 1157, 1164, 1167, 1168, 1177, 1178, 1179, 1180, 1181, 1247
 \cs_set_protected:Npe 240, 242
 \cs_set_protected:Npn ... 1137, 1146, 1175, 1196, 1202
 \cs_undefine:N 136, 431

ctex commands:

`\ctex_after_end_preamble:n` 71
`\ctex_at_end_package:nn` 85
`\ctex_at_end_preamble:n` 70
`\ctex_disable_package:n` 20, 40, 43
`\ctex_if_format_at_least:nTF` 41
`\ctex_replace_package:nn` 42

D

Default 368
`\defaultCJKfontfeatures` 7
depth 15

dim commands:

`\dim_case:nn` 947, 1266
`\dim_case:nnTF` 884
`\dim_compare:nNnTF` 867
`\dim_const:Nn` 1046
`\dim_eval:n` 202
`\dim_gset:Nn` 1046, 1374
`\dim_if_exist:NTF` 1045
`\dim_new:N` 51, 1365
`\dim_use:N` 223
`\c_zero_dim` 868

E

else commands:

`\else:` 145, 255, 260, 308, 316, 1023, 1040
EmboldenFactor 4
EnvCS 4
EnvCS+ 4
EnvCS- 4

exp commands:

`\exp_after:wN` 139,
234, 252, 253, 258, 259, 261, 262, 273, 301, 1343, 1350, 1359
`\exp_args:Nc` 343, 1061
`\exp_args:Ncc` 1013
`\exp_args:NNNo` 218, 222, 225
`\exp_last_unbraced:Ne` 113
`\exp_not:N` 46, 241, 243, 257, 674, 1310, 1312
`\exp_not:n` 241, 243, 408,
409, 448, 449, 450, 451, 452, 453, 632, 638, 652, 662, 671
`\exp_stop_f:` 144, 315

experiment/halfright-prebreakpenalty 6
experiment/punct-measure-fix 6

F

FallBack 9

fi commands:

`\fi:` 145, 234, 263, 264, 308, 316, 1023, 1040

format 15

fp commands:

`\fp_compare:nNnTF` 371
`\fp_eval:n` 203

FullLeft 368

FullRight 368

G

gap 16

group commands:

`\group_align_safe_begin:` 245, 1220, 1273
`\group_align_safe_end:` 241, 243, 1251, 1276, 1282
`\group_begin:` 214, 274, 1010
`\c_group_begin_token` 695
`\group_end:` 225, 302, 1033
`\c_group_end_token` 701

H

HalfLeft 387

HalfRight 387

HangulJamo 387

hbox commands:

`\hbox_set:Nw` 215
`\hbox_set_end:` 218, 222

height 16

hidden 15

I

if commands:

`\if_catcode:w` 256
`\if_cs_exist:w` 305
`\if_dim:w` 1036
`\if_int_compare:w` 231, 314, 1022
`\if_meaning:w` 250

`\IfBooleanT` 523

InlineEnv 4

InlineEnv+ 4

InlineEnv- 4

int commands:

`\int_compare:nNnTF` 149, 162, 182, 374
`\int_const:Nn` 68, 356, 379, 385, 1019
`\int_div_truncate:nn` 205
`\int_eval:n` 280, 287, 297, 311
`\int_gincr:N` 1044, 1051
`\int_gset:Nn` 210, 1057
`\int_gset_eq:NN` 790, 823
`\int_if_exist:NTF` 340, 353
`\int_incr:N` 568, 578
`\int_max:nn` 558
`\int_min:nn` 557
`\int_new:N` 49, 208, 235, 236, 1056
`\int_set:Nn` 553, 557, 558, 563, 564, 574, 583
`\int_set_eq:NN` 554
`\int_until_do:nNnn` 24
`\int_use:N` 370
`\int_value:w` 206
`\int_zero:N` 698
`\c_one_int` 328, 374
`\c_zero_int` 65, 330, 1369

K

KaiMingPunct 5

KaiMingPunct+ 5

KaiMingPunct- 5

keys commands:			
\keys_define:nn	320, 1133, 1142, 1190		
L			
LoadFandol	6		
LocalConfig	3		
LongPunct	5		
LongPunct+	5		
LongPunct-	5		
M			
\makeCJKactive	323, 327		
\makeCJKinactive	324, 327		
Mapping	7		
MiddlePunct	5		
MiddlePunct+	5		
MiddlePunct-	5		
mode commands:			
\mode_if_inner:TF	1368		
msg commands:			
\msg_critical:nn	10		
\msg_critical:nnn	20		
\msg_error:nn	57		
\msg_error:nnn	36, 58		
\msg_info:nnnn	63		
\msg_new:nnn	3, 11, 28, 55		
\msg_new:nnnn	56		
\msg_warning:nn	59		
\msg_warning:nnn	60		
\msg_warning:nnnn	61		
\msg_warning:nnnnn	62		
N			
\newCJKfontfamily	7		
\NewDocumentCommand	44, 99, 327, 329, 520, 544, 587, 592, 603		
NewLineCS	4		
NewLineCS+	4		
NewLineCS-	4		
\newXeTeXintercharclass	343		
NoBreakCS	5		
NoBreakCS+	5		
NoBreakCS-	5		
NoBreakLongPunct	5		
NoBreakLongPunct+	5		
NoBreakLongPunct-	5		
NormalSpace	387		
\normalspacedchars	11, 587		
P			
peek commands:			
\peek_after:Nw	246, 252		
\peek_meaning_remove:NTF	791, 824		
\l_peek_token	250, 257, 797, 830, 1223, 1228, 1231, 1274		
PlainEquation	4		
prg commands:			
\prg_do_nothing:	128, 130, 267, 268, 628, 674, 972, 1127		
\prg_new_conditional:Npnn	22, 142, 303, 312, 1020, 1034		
\prg_return_false:	25, 145, 308, 316, 1023, 1040		
\prg_return_true:	25, 145, 308, 316, 1023, 1040		
PunctBoundWidth	5		
PunctFamily	4		
PunctStyle	4		
PunctWidth	5		
Q			
quark commands:			
\q_stop	273, 275, 279, 285, 296, 301		
R			
\RequirePackage	21, 45, 46		
reverse commands:			
\reverse_if:N	231		
RubberPunctSkip	5		
S			
scan commands:			
\scan_stop:	22, 25, 215, 239, 319, 1228, 1246, 1247		
\s_stop	114, 116		
sep	15		
seq commands:			
\seq_gput_right:Nn	346, 347, 358, 685		
\seq_gset_eq:NN	678		
\seq_gset_from_clist:Nn	680		
\seq_map_inline:Nn	704, 717, 1305		
\seq_new:N	336, 337, 677, 679, 682		
\setCJKfallbackfamilyfont	8, 9		
\setCJKfamilyfont	7		
\setCJKmainfont	6		
\setCJKmathfont	8		
\setCJKmonofont	7		
\setCJKsansfont	7		
skip	15		
skip commands:			
\skip_const:Nn	1053		
\skip_gset:Nn	1053		
\skip_horizontal:N	759, 762, 772, 779, 780, 782, 856, 916, 939, 956, 969, 983, 998, 1001, 1002, 1006, 1088, 1091, 1094, 1113		
\skip_if_eq:nnTF	151, 160, 168, 185, 776, 990		
\skip_if_exist:NTF	1052		
\skip_if_finite:nTF	865		
\skip_new:N	53, 785, 1141, 1188		
\skip_set:Nn	219, 223, 226		
\skip_set_eq:NN	769, 864, 977		
\skip_use:N	219, 226		
\c_zero_skip	151, 160, 168, 185		
SlantFactor	4		
\SplitArgument	545		
str commands:			
\str_const:Nn	369		
\str_if_eq:nnTF	17, 536, 706, 710, 719		
subtract	15		
symbol	15		

sys commands:

\sys_if_engine_xetex:TF 10

T

TeX and L^AT_EX 2_ε commands:

\@beginDvi 21, 90, 94, 97
 \@empty 21
 \@ifpackagelater 19
 \@pkgextension 27
 \[..... 4
 \addCJKfontfeatures 8
 \AtBeginDvi 21
 \AtBeginShipout 17, 21
 \baselineskip 3
 \begin 4, 17
 \beginGroup 35
 \catcode 17, 24
 \char 19
 \CJKecglue 38
 \CJKfamily 7, 17
 \CJKfamilydefault 8, 9, 11
 \CJKfontspec 7
 \CJKglue 17
 \CJKrmdefault 8
 \CJKsout 15, 16, 18
 \CJKsymbol 17
 \CJKunderanyline 16
 \CJKunderanysymbol 16
 \CJKunderdblline 15, 16, 18
 \CJKunderdot 15, 16, 18
 \CJKunderline 14, 15, 18
 \CJKunderwave 15, 18
 \CJKxout 18
 \color 15
 \colorbox 18, 40
 \cprotect 18
 \defaultCJKfontfeatures 7
 \document 21
 \dotfill 38
 \emph 18
 \end 4, 17
 \endGroup 35
 \familydefault 8
 \footnote 5, 17
 \footnotemark 5
 \hbox 16, 40, 48
 \hrulefill 38
 \hskip 3
 \hyperref 18, 19
 \icprotect 18
 \kern 6
 \leaders 38
 \maxdimen 12, 13
 \mbox 18, 40
 \meaning 25
 \newCJKfontfamily 4, 7

\nobreak 5
 \normalfont 8
 \normalspacedchars 11
 \outer 18
 \par 4, 6
 \parfillskip 6
 \path 3
 \ref 18, 19
 \relax 18, 46
 \reset@color 40
 \rmfamily 6, 8
 \setCJKfallbackfamilyfont 8, 9
 \setCJKfamilyfont 4, 7
 \setCJKmainfont 2, 6
 \setCJKmathfont 8
 \setCJKmonofont 7
 \setCJKsansfont 7
 \sffamily 7, 8
 \shipout 21, 22
 \spacefactor 23
 \textbf 18
 \textcolor 18
 \textit 18
 \textnormal 8
 \textrm 6, 8
 \textsf 7, 8
 \texttt 7, 8
 \ttfamily 7, 8
 \uline 16
 \ULon 16
 \ULthickness 15
 \unskip 6
 \verb 6
 \verbatim@font 6
 \xeCJK@document@hook 69, 72
 \xeCJK@document@left@hook 70, 74
 \xeCJK@document@right@hook 71, 76
 \xeCJK@first@beginDvi 90, 91
 \xeCJKCancelSubCJKBlock 11
 \xeCJKchar 19
 \xeCJKDeclareCharClass 11
 \xeCJKDeclarePunctStyle 4, 12
 \xeCJKDeclareSubCJKBlock 11
 \xeCJKEditPunctStyle 12
 \xeCJKfntefbox 16
 \xeCJKfntefon 16
 \xeCJKnobreak 5, 17
 \xeCJKOffVerbAddon 17
 \xeCJKRestoreSubCJKBlock 11
 \xeCJKsetkern 12, 14
 \xeCJKsetup 2, 3, 12, 15, 16
 \xeCJKsetwidth 5, 12
 \xeCJKShipoutHook 17
 \xeCJKVerbAddon 6, 17
 \XeTeXdashbreakstate 37
 \XeTeXinterchartoks 18, 23

tex commands:

<code>\tex_font:D</code>	139, 144, 153, 154, 155, 165, 172, 173, 178, 179, 190
<code>\tex_fontdimen:D</code>	153, 154, 155, 165, 172, 173, 178, 179, 190
<code>\tex_glueshrink:D</code>	196, 868
<code>\tex_gluestretch:D</code>	195
<code>\tex_hskip:D</code>	1130, 1131
<code>\tex_iffontchar:D</code>	144
<code>\tex_ignorespaces:D</code>	1353, 1362
<code>\tex_italiccorrection:D</code>	791, 793, 824, 826
<code>\tex_kern:D</code>	1070, 1071
<code>\tex_lastkern:D</code>	884, 947, 1038, 1266
<code>\tex_lastnodetype:D</code>	1022
<code>\tex_lastskip:D</code>	219, 769, 864, 977, 1039, 1374
<code>\tex_nullfont:D</code>	141
<code>\tex_penalty:D</code>	65, 67, 968, 1369
<code>\tex_romannumeral:D</code>	254
<code>\tex_spacefactor:D</code>	790, 823
<code>\tex_spaceskip:D</code>	151, 157, 160, 183, 189, 195, 196
<code>\tex_the:D</code>	139, 624
<code>\tex_unkern:D</code>	1076
<code>\tex_unskip:D</code>	770, 870, 978, 1079
<code>\tex_XeTeXcharclass:D</code>	271, 374, 577, 585, 590
<code>\tex_XeTeXdashbreakstate:D</code>	698
<code>\tex_XeTeXinterchartokenstate:D</code>	328, 330
<code>\tex_XeTeXinterchartoks:D</code>	618, 624, 672
<code>\tex_XeTeXrevision:D</code>	370
<code>\tex_XeTeXversion:D</code>	370
<code>\tex_xspaceskip:D</code>	168, 176, 185, 193
textformat	15
thickness	15

tl commands:

<code>\c_space_tl</code>	1187
<code>\tl_const:Nn</code>	27, 96, 147, 686
<code>\tl_gc_clear:N</code>	928, 931, 935, 1081
<code>\tl_gput_right:Nn</code>	79, 81, 83, 95
<code>\tl_gset:Nn</code>	1062
<code>\tl_if_blank:NnTF</code>	553
<code>\tl_if_blank_p:n</code>	550, 551
<code>\tl_if_empty:NnTF</code>	645, 649, 658
<code>\tl_if_empty:NnTF</code>	277, 283, 289, 297
<code>\tl_if_eq:NnTF</code>	970
<code>\tl_if_eq:NnTF</code>	927, 930, 933, 1087, 1090, 1093, 1103, 1106, 1109, 1112
<code>\tl_if_exist:NnTF</code>	24
<code>\tl_if_head_is_N_type:NnTF</code>	120
<code>\tl_if_no_value:NnTF</code>	557, 558
<code>\tl_if_single:NnTF</code>	118
<code>\tl_map_inline:nm</code>	589
<code>\tl_new:N</code>	47, 48, 86, 87, 88, 109, 1058
<code>\tl_put_right:Nn</code>	90, 108
<code>\tl_replace_all:Nmn</code>	660
<code>\tl_set:Nn</code>	643, 647, 656
<code>\tl_trim_spaces:n</code>	114, 121, 124
<code>\tl_use:N</code>	73, 75, 77, 104

token commands:

<code>\c_math_toggle_token</code>	1213, 1258
<code>\c_space_token</code>	250
<code>\token_if_eq_meaning:NnTF</code>	1228, 1246
<code>\token_if_group_end:NnTF</code>	1231
<code>\token_if_letter:NnTF</code>	561
<code>\token_if_macro:NnTF</code>	1223, 1274
<code>\token_if_space:NnTF</code>	797, 830
<code>\token_to_meaning:N</code>	273, 301
<code>\TrimSpaces</code>	520

U

use commands:

<code>\use:N</code>	519, 687, 1039, 1065, 1067
<code>\use_i:nm</code>	757
<code>\use_none:n</code>	230, 233, 234
<code>\use_none:nm</code>	176, 193

V

Verb	6
------------	---

W

WidowPenalty	3
--------------------	---

X

xCJKecglue	3, 1142
------------------	---------

xeCJK commands:

<code>\xeCJK_add_to_shipout:n</code>	107, 107
<code>\xeCJK_allow_break:</code>	64, 64
<code>\xeCJK_app_inter_class_toks:nm</code>	635, 635, 640, 743, 1316
<code>\xeCJK_Boundary_and_Default:</code>	742, 746, 746
<code>\xeCJK_Boundary_and_FullLeft:N</code>	1330
<code>\xeCJK_Boundary_and_FullRight:N</code>	1332
<code>\xeCJK_Boundary_and_NormalSp:</code>	805, 806, 806
<code>\xeCJK_check_for_ecglue:</code> ..	747, 766, 1156, 1167, 1180
<code>\xeCJK_check_for_ecglue_normalsp:</code>	807, 819, 1158, 1169, 1182
<code>\xeCJK_check_for_glue:</code>	837, 844, 844
<code>\xeCJK_check_for_xglue:</code> ..	963, 972, 1155, 1166, 1179
<code>\xeCJK_CJK_and_Boundary:w</code>	1210, 1211, 1211
<code>\xeCJK_CJK_and_CJK:N</code>	1286, 1287, 1287
<code>\xeCJK_class_group_begin:</code>	693, 693, 730, 838
<code>\xeCJK_class_group_end:</code>	693, 701, 737, 1216, 1217, 1252, 1324, 1342, 1349, 1358
<code>\xeCJK_class_num:n</code>	518, 518, 539, 542, 583, 590, 618, 619, 624, 625, 673, 674
<code>\xeCJK_clear_Boundary_and_CJK_toks:</code>	665, 665, 669, 671, 733, 839
<code>\xeCJK_clear_inter_class_toks:nm</code> ..	627, 627, 650, 732
<code>\xeCJK_copy_inter_class_toks:nmmn</code>	641, 641, 707, 709, 711, 721, 722
<code>\xeCJK_cs_clear:N</code>	127, 127, 1165, 1166
<code>\xeCJK_cs_gc_clear:N</code>	127, 129
<code>\xeCJK_declare_char_class:nm</code>	525, 528, 530, 598, 599, 600, 601, 609, 610, 611, 612
<code>\xeCJK_declare_char_class:nm</code>	522, 525, 525

\xeCJK_declare_glue_node:n	1049, 1124	\xeCJK_space_or_xecglue:	817, 1153, 1164, 1177, 1216, 1261, 1283
\xeCJK_declare_node:n	1042, 1042, 1119, 1120, 1121, 1122, 1123	\c_xeCJK_space_skip_tl	147, 778, 992
\xeCJK_fallback_symbol:NN	734, 841, 1290, 1296, 1302	\xeCJK_swap_cs:NN	131, 131
\xeCJK_font_gset_to_current:N	138, 138	\xeCJK_tl_remove_outer_braces:n	111, 111, 122
\xeCJK_FullLeft_and_Boundary:	1334, 1337, 1337	\xeCJK_token_value_charcode:N	271, 272, 272
\xeCJK_FullLeft_and_CJK:	1295	\xeCJK_token_value_class:N	270, 270, 306, 314, 315
\xeCJK_FullRight_and_Boundary:	1336, 1355, 1355	\xeCJK_widow_penalty:	891, 936, 954, 1110
\xeCJK_FullRight_and_CJK:	1301	xeCJK internal commands:	
\xeCJK_get_inter_class_toks:nn	622, 622, 632, 638, 644, 648, 657	__xeCJK_add_special_punct:nn	429
\xeCJK_glue_to_skip:nN	212, 212, 1138, 1147, 1176, 1253	__xeCJK_after_end_preamble:n	69, 82
\xeCJK_glyph_if_exist:N	142	\g__xeCJK_after_end_preamble_hook_tl	77, 83, 88
\xeCJK_glyph_if_exist:NTF	142	__xeCJK_after_preamble:n	69, 80, 89
\xeCJK_glyph_if_exist_p:N	142	\g__xeCJK_after_preamble_hook_tl	73, 81, 87
\xeCJK_if_CJK_class:N	303	\l__xeCJK_aligni_tl	970
\xeCJK_if_CJK_class:NTF	303	__xeCJK_at_end_preamble:n	69, 78
\xeCJK_if_CJK_class_p:N	303	\g__xeCJK_at_end_preamble_hook_tl	75, 79, 86
\xeCJK_if_last_node:n	1034	\g__xeCJK_base_class_seq	677
\xeCJK_if_last_node:nTF	756, 758, 761, 771, 774, 816, 979, 982, 994, 997, 1034	\l__xeCJK_begin_int	235, 563, 565, 567, 568, 573, 575, 577, 578
\xeCJK_if_last_node_p:n	1034	__xeCJK_boundary_group_end:n	1225, 1234, 1236, 1242, 1245, 1249
\xeCJK_if_last_punct:TF	961	__xeCJK_boundary_reserve_space:	1224, 1241
\xeCJK_if_package_loaded:n	22	__xeCJK_ccglue_or_space:	889, 931, 952, 980, 995, 1008, 1107, 1196, 1202
\xeCJK_if_package_loaded:nTF	22, 35	\l__xeCJK_ccglue_skip	1138, 1141
\xeCJK_if_package_loaded_p:n	22	__xeCJK_check_for_ecglue:	752, 754, 766, 1167
\xeCJK_if_same_class:NN	312	__xeCJK_check_for_ecglue_normalsp:	812, 814, 820, 1170
\xeCJK_if_same_class:NNTF	312	__xeCJK_check_for_glue_auxi:	847, 945
\xeCJK_if_same_class_p:NN	312	__xeCJK_check_for_glue_auxii:	857, 917, 923, 940, 959
\xeCJK_ignore_spaces:w	1256, 1256	__xeCJK_check_for_glue_auxiii:	962, 965
\xeCJK_int_until_do:nn	228, 228, 565, 575	__xeCJK_check_for_glue_skip:	850, 862
\xeCJK_inter_class_toks:nnn	616, 616, 621, 628, 631, 637, 652, 661, 728, 737, 741, 788, 804, 821, 835, 1210, 1285, 1293, 1299, 1309, 1311, 1321, 1329, 1331, 1333, 1335	__xeCJK_check_for_glue_skip_consume_and_fallback:	875, 877, 919
\xeCJK_make_boundary:	318, 318	__xeCJK_check_for_glue_skip_hlist_aux:	902, 908, 925
\xeCJK_make_node:n	794, 799, 827, 832, 1042, 1059, 1254, 1269	__xeCJK_check_for_glue_skip_kern:	872, 879
\xeCJK_make_space_node:	798, 831, 1127, 1127, 1154, 1165, 1178, 1271	__xeCJK_check_for_glue_skip_nonkern:	873, 897
\xeCJK_new_class:n	338, 338, 382, 383, 384, 387, 388, 389, 390, 391	__xeCJK_check_for_glue_skip_restore:	893, 895, 909, 911, 914
\xeCJK_no_break:	64, 66, 1319, 1326, 1344	__xeCJK_check_for_xecglue:	748, 1156, 1180
\xeCJK_peek_catcode_ignore_spaces:NTF	237, 237, 1213, 1258	__xeCJK_check_for_xecglue_normalsp:	808, 1159, 1183
\xeCJK_pre_inter_class_toks:nnn	629, 629, 634, 802	__xeCJK_check_for_xglue:	973, 1155, 1179
\xeCJK_punct_node:N	1343, 1350, 1359	__xeCJK_check_for_xglue_aux:	984, 988
\xeCJK_remove_node:	759, 762, 772, 779, 817, 856, 887, 889, 891, 950, 952, 954, 956, 980, 983, 995, 998, 1073, 1088, 1091, 1094, 1104, 1107, 1110, 1113, 1269, 1271	__xeCJK_check_num_range:nnNN	547, 547, 573
\xeCJK_replace_inter_class_toks:nnnn	654, 654	\c__xeCJK_CJ_chars_clist	436
\xeCJK_reset_space_factor:	208, 209, 211, 697	__xeCJK_CJK_and_Boundary_relax:N	1229, 1243
\xeCJK_save_class:nn	351, 351, 368, 372, 373, 376, 377, 378	\c__xeCJK_CJK_chars_clist	455, 609
\xeCJK_select_font:	731, 840	\g__xeCJK_CJK_class_seq	677
\xeCJK_set_char_class:nnn	538, 542, 571, 571	__xeCJK_CJK_class_tl:n	306, 310, 687
\xeCJK_space_glue:	1164, 1187, 1197	\l__xeCJK_CJK_group_bool	319, 696, 700
		\g__xeCJK_CJK_range_clist	605
		\c__xeCJK_CL_chars_clist	411, 448
		\c__xeCJK_class_begin_int	379, 385

_xeCJK_class_csname:n
 340, 344, 348, 353, 356, 361, 361, 362, 519, 687
 \g_xeCJK_class_seq 336, 346, 358, 678, 704, 717
 \c_xeCJK_CM_chars_clist 505, 611
 \g_xeCJK_CM_range_clist 607
 \l_xeCJK_ecglue_skip
 759, 762, 772, 779, 856, 956, 983,
 998, 1001, 1088, 1091, 1094, 1113, 1147, 1176, 1188, 1253
 \l_xeCJK_end_int 236, 564, 565, 573, 575
 _xeCJK_error:n 55, 57
 _xeCJK_error:nn 55, 58, 341, 354
 \c_xeCJK_EX_chars_clist 433, 450
 _xeCJK_font_gset_to_current_aux:NN 139, 140
 \c_xeCJK_FullLeft_chars_clist 406, 600
 \g_xeCJK_FullLeft_range_clist 596
 \c_xeCJK_FullRight_chars_clist 446, 601
 \g_xeCJK_FullRight_range_clist 597
 _xeCJK_get_charcode:w 273, 279, 285, 296
 \g_xeCJK_glue_check_pending_bool
 881, 883, 899, 900, 921, 922, 1125, 1233
 _xeCJK_glue_node:n 1066, 1130, 1131
 \c_xeCJK_HalfLeft_chars_clist 392, 598
 \g_xeCJK_HalfLeft_range_clist 594
 \c_xeCJK_HalfRight_chars_clist 392, 599
 \l_xeCJK_halfright_nobreak_bool .. 1315, 1318, 1325
 \g_xeCJK_HalfRight_range_clist 595
 \c_xeCJK_HangulJamo_chars_clist 512, 612
 \g_xeCJK_HangulJamo_range_clist 608
 \c_xeCJK_hyphens_chars_clist 419, 430, 453
 _xeCJK_if_last_glue:TF
 216, 750, 810, 849, 975, 1010, 1078
 _xeCJK_if_last_hlist:TF 901, 1010
 _xeCJK_if_last_kern:TF 846, 871, 1010, 1075
 _xeCJK_if_last_math:TF 855, 1000, 1010
 _xeCJK_if_last_none:TF 1010
 _xeCJK_if_last_penalty:TF 1010
 _xeCJK_if_last_whatsit:TF 852, 907, 1085, 1101
 _xeCJK_info:nnn 55, 63
 _xeCJK_int_until_do:wn 228, 230, 233, 234
 \c_xeCJK_IS_chars_clist 435, 451
 \c_xeCJK_iteration_marks_chars_clist 421
 \g_xeCJK_last_node_tl 927, 928, 930, 931, 933, 935,
 1058, 1062, 1081, 1087, 1090, 1093, 1103, 1106, 1109, 1112
 \l_xeCJK_last_penalty_bool 967
 \l_xeCJK_last_penalty_int 968
 \g_xeCJK_last_punct_tl
 1323, 1327, 1339, 1341, 1343, 1345, 1350, 1357, 1359, 1360
 \l_xeCJK_last_skip 769, 777, 780,
 782, 785, 864, 865, 868, 916, 939, 969, 977, 991, 1002, 1006
 \c_xeCJK_left_tl 970, 1345
 _xeCJK_make_node:N 1061, 1068
 _xeCJK_make_space_node: 1128, 1154, 1178
 _xeCJK_msg_new:nn 55, 55, 363
 _xeCJK_msg_new:nnn 56
 \g_xeCJK_new_class_seq 336, 347
 \c_xeCJK_nobreak_penalty_int 67, 68
 _xeCJK_nobreak_zero_glue: 1351
 _xeCJK_node:n
 886, 888, 890, 949, 951, 953, 955, 1064, 1268, 1270
 \g_xeCJK_node_int .. 1044, 1047, 1051, 1054, 1056, 1057
 \g_xeCJK_non_CJK_class_seq 677, 1305
 \c_xeCJK_NormalSpace_chars_clist 392, 610
 \g_xeCJK_NormalSpace_range_clist 606
 \c_xeCJK_NS_chars_clist 419, 449
 \c_xeCJK_OP_chars_clist 397, 408
 \c_xeCJK_package_ext_tl 24, 27
 _xeCJK_package_hook:nn 69, 84
 \g_xeCJK_par_guard_bool 1364, 1373
 \g_xeCJK_par_guard_dim 1365, 1374
 _xeCJK_peek_catcode_false:w 242, 262, 268
 _xeCJK_peek_catcode_ignore_spaces_branches:w
 246, 248, 253
 _xeCJK_peek_catcode_true:w 240, 259, 267
 \l_xeCJK_peek_ignore_spaces_bool
 24, 244, 251, 269, 1215, 1221, 1260, 1264
 \l_xeCJK_peek_search_token 239, 257, 266
 \c_xeCJK_PO_chars_clist 445, 452
 \c_xeCJK_PR_chars_clist 404, 409
 _xeCJK_punct_boundary_guard: 1346, 1361, 1364, 1366
 _xeCJK_punct_glue:NN 1327, 1345, 1360
 _xeCJK_punct_if_middle:NTF 1339
 \g_xeCJK_punct_measure_fix_bool 1371
 _xeCJK_punct_rule:NN 1323, 1341, 1357
 _xeCJK_recover_ecglue_whatsit: 1083
 _xeCJK_recover_glue_whatsit: 853, 1099
 _xeCJK_replace_space: 751, 767, 767, 811
 _xeCJK_reserve_space_aux: 1275, 1280
 \l_xeCJK_reserve_space_bool 1195, 1201, 1209
 \g_xeCJK_reset_color_pending_bool .. 904, 906, 1126
 \c_xeCJK_right_tl 1323, 1327, 1341, 1357, 1360
 _xeCJK_save_CJK_class:n 683, 692
 _xeCJK_set_char_class_aux:Nnw 525, 538, 544
 _xeCJK_set_char_class_eq:nn 581, 581
 \l_xeCJK_shipout_hook_bool 101, 103, 110
 \l_xeCJK_shipout_hook_tl 104, 108, 109
 \g_xeCJK_space_factor_int
 149, 162, 182, 203, 206, 208, 790, 823
 _xeCJK_space_skip_scale:nnn . 164, 170, 183, 187, 200
 _xeCJK_swap_cs_aux:w 133, 135, 136
 _xeCJK_tl_remove_outer_braces:w 114, 116
 _xeCJK_tmp:nn
 1011, 1026, 1027, 1028, 1029, 1030, 1031, 1032
 _xeCJK_tmp:w 275, 301, 427, 430, 431
 _xeCJK_tmp_aux:NNn 1013, 1017
 \l_xeCJK_tmp_bool 52
 \l_xeCJK_tmp_box 50, 215, 223
 \l_xeCJK_tmp_clist 54, 527, 528
 \l_xeCJK_tmp_dim 51
 \l_xeCJK_tmp_int 49, 362, 574, 577, 583, 585
 \l_xeCJK_tmp_skip 53
 \l_xeCJK_tmp_tl
 47, 643, 645, 647, 649, 652, 656, 658, 660, 662

\l_xeCJK_tmpb_tl	48	\xeCJKEditPunctStyle	12
_xeCJK_update_clear_toks:n	667, 689	\xeCJKfntefon	16
_xeCJK_warning:n	59	\xeCJKnobreak	17
_xeCJK_warning:nn	55, 60	\xeCJKOffVerbAddon	17
_xeCJK_warning:nnn	61	\xeCJKResetCharClass	11, 603, 615
_xeCJK_warning:nnnn	62	\xeCJKResetPunctClass	11, 523, 592, 613
\l_xeCJK_xecglue_bool	1152, 1163, 1174, 1189	\xeCJKRestoreSubCJKBlock	11
\c_xeCJK_xetex_version_str	369, 371	\xeCJKsetkern	12
xeCJKactive	3, 320	\xeCJKsetup	2
\xeCJKCancelSubCJKBlock	11	\xeCJKsetwidth	12
\xeCJKDeclareCharClass	11, 520	\xeCJKShipoutHook	17, 89
\xeCJKDeclarePunctStyle	12	\xeCJKVerbAddon	17
\xeCJKDeclareSubCJKBlock	11		