

L^AT_EX 2_ε 的配置选项

© Copyright 1998, 2001, 2003 L^AT_EX 项目团队

保留所有权

赣医一附院神经科 黄旭华 翻译

2003 年 2 月 14 日

目 录

配置 L ^A T _E X	3
系统配置	3
texsys.cfg	3
配置 L ^A T _E X 格式	4
字体配置	4
fonttext.cfg	5
fontmath.cfg	5
preload.cfg	6
连字符配置	6
hyphen.cfg	6
配置字体定义文件	7
配置兼容模式	8
latex209.cfg	9

标准包和类的配置文件	9
sfonts.cfg	9
ltnews.cfg	10
ltxdoc.cfg	10
ltxguide.cfg	11
其他受支持包的配置	11
graphics.cfg	11
color.cfg	12
非标准版本	12
示例	12

配置 L^AT_EX

新的标准 L^AT_EX 的主要目标之一是通过可靠文档处理系统 (reliable document processing system) 为所有用户提供自由，该文档处理系统可以连接到高度可移植文档格式 (portable document format)，因此其配置 (configuration) 受到严格限制。`modguide.tex` 文件中的 *Modifying L^AT_EX* (《修改 L^AT_EX》) 一文对此进行了更详细的解释。这样做的一个重要后果是，依赖于任何扩展包 (extension package) 的任何文档都必须在文档文件中声明此包；这有助于确保文档可在不同的位置 (site) 工作，该位置 L^AT_EX 系统的配置可能不同。

按照惯例 (convention)，局部配置选项 (local configuration options) 放在扩展名为 `.cfg` 的“配置文件 (configuration files)”中。该文件描述了这个 L^AT_EX 发行版中配置的可能性 (possibilities)；它还解释了如何配置字体定义文件 (font definition files) 以利用可用的字体。

最后一节将简要介绍在需要进一步自定义格式化程序时如何进行操作。

系统配置

`texsys.cfg`

这是必须存在的唯一配置文件 (configuration file)。在安装过程中，如果 L^AT_EX 无法找到这个名称的文件，那么将写出并使用一个完全由注释 (comments) 组成的缺省文件 `texsys.cfg`。请注意，在读取该文件之前，L^AT_EX 无法可靠地测试系统上是否存在给定的文件。

`texsys.cfg` 文件的内容允许 L^AT_EX 处理不同 T_EX 系统行为之间的各种差异，主要涉及到文件处理 (file handling)。该文件的默认版本在其注释中包含一系列 T_EX 系统可能需要的可能设置。有关更多信息，请排版 (typeset) `ltdirchk.dtx` 文件。

如果您已经从使用不同操作系统的计算机上复制了 L^AT_EX 安装 (installation)，那么您很可能有一个版本的 `texsys.cfg`，这将使得在您的系统上安装 L^AT_EX 变得困难。如果发生这种情况，那么使用一个空的 `texsys.cfg` 文件重新启动该过程；这将产生一个安装，该安装至少应该允许您对文档进行排版。

但是，有可能 L^AT_EX 仍然只能找到当前目录 (current directory) 中的那些文件，在这种情况下，您必须正确设置 `\input@path` 宏以适合您的系统。

配置 L^AT_EX 格式

有四个配置文件 (configuration files) 可以将个人首选项 (personal preferences) 合并到 L^AT_EX 格式文件 `latex.fmt` 中。这些文件可以配置的首选项范围受到严格限制，因为这有助于确保文档的可移植性 (document portability)。

所有四个文件的工作方式相同：如果找到了 `\file`.`.cfg` 文件，它将由 `iniTEX` 输入，否则输入默认文件 `\file`.`.ltx`；这有时是通过最小 `\file`.`.cfg` 输入 `\file`.`.ltx` 来完成的。因此，提供您自己的版本的任何这些 `.cfg` 文件可以完全覆盖相应的默认标准 `.ltx` 文件中的任何设置。

字体配置

在您考虑通过生成 `fontmath.cfg` 或 `fonttext.cfg` 文件来配置字体声明 (font declarations) 之前，应该先阅读 `fontdef.dtx` 这个文档化的文件 (documented file)。这是生成默认文件 `fonttext.ltx` 和 `fontmath.ltx` 的源文件 (source file)；它包含关于默认文件 (default files) 的内容以及可以进行哪种定制 (customisation) 的信息。特别是，它详细描述了个人定制 (individual customisations) 对文档可移植性 (document portability) 的影响，包括：在不危及与其他位置 (site) 交换文档的能力的情况下，可以进行哪些定制 (即使格式不同)；哪些东西应该保持不变，因为它们会使您的系统与其他系统如此不同，以至于它生成的文档将不可移植 (non-portable)。

警告 请注意，使用这些字体配置文件中的任何一个都会产生以下后果：

- 由于 `fontdef.dtx` 文件的内容将来可能会更改，所以任何编写字体配置文件的人都必须准备好更新它，以便在将来的版本中使用。
- 在您的系统上生成的文档充其量只能在不同的地方 (site) 上处理，在这个意义上说是可移植的——如果使用不同的字体，实际格式 (actual formatting) 将不一样。

- 如果这些问题不能以不使用任何配置文件的格式再现 (reproduced)，则 L^AT_EX 项目团队将无法支持您诊断这些问题。

fonttext.cfg

`fonttext.cfg` 文件可以包含与文本模式 (text modes) 中字体使用相关的声明。

如果存在，它定义了文本模式中通常使用的字体形状 (shapes)、族 (families) 和编码 (encodings)，以及字体属性 (font attribute) 命令 (如 `\textbf` 等) 的行为 (behavior)。

例如，它可以用来生成一种 L^AT_EX 格式，默认情况下，该格式使用 Times 字体排版文档。但是要注意，这种定制 (customisation) 可能会带来不幸的后果；因此，如果您正在考虑这样做，请仔细阅读本节和下面的 `fontdef.dtx` 文件。

请注意以上警告。

fontmath.cfg

`fontmath.cfg` 文件可以包含与数学模式 (math mode) 中字体使用相关的声明。

如果存在，它将定义在数学模式中使用的字体尺寸 (sizes)，以及如何使用这些字体。它还定义了所有数学模式命令 (math mode commands)，这些数学模式命令“很可能会”依赖所选择的数学字体使用的命令 (例如，依赖数学字体中字形位置的命令)

此文件存在的主要原因是，当有标准的数学字体编码 (standard math font encoding) 可用时，可以提供给将来的更新 (future updates)。目前，我们不鼓励将此配置文件 (configuration file) 用于特殊应用程序 (special applications) 以外的其他用途。为数学模式编写合适的配置文件需要专业知识！

请注意以上警告。

preload.cfg

`preload.cfg` 文件的内容可以控制常用字体的预加载 (preloading)。预加载字体 (preloading fonts) 可以加快文档的处理速度, 但由于无法“卸载 (unloaded)”字体, 因此不应预加载过多; 否则, 您可能无法处理需要特殊字体族 (unusual font families) 的文档。

默认文件 `preload.ltx` 由 `preload.dtx` 产生。它只加载一些字体, 如果您通常使用默认的 10pt 尺寸的文档, 那么这些字体是一个不错的选择。如果您通常使用 11pt 或 12pt, 那么如果您为所使用的尺寸预加载相应的字体, 则 \LaTeX 启动的时间可能会显著缩短。同样, 如果您通常使用不同的字体族 (font family), 例如 Times Roman (`ptm`), 那么您可能希望在此族中预加载字体, 而不是默认的计算机现代字体 (Computer Modern fonts)。

连字符配置

hyphen.cfg

为了对文本使用连字 (hyphenate), \TeX 必须具有连字符模式 (hyphenation patterns), 并且由于这些模式只能由 iniTeX 加载, 因此必须在创建格式时选择要加载的模式。

美式英语 (American English) 的连字符模式 (hyphenation patterns) 存储在名为 `hyphen.tex` 的文件中; 在制定格式时, \LaTeX 2.09 总是加载这个文件。

使用 \LaTeX 2_ε 可以配置要加载到格式中的连字符模式。当 iniTeX 处理 `latex.ltx` 时, 它查找一个名为 `hyphen.cfg` 的文件; 该文件可用于控制加载哪些连字符模式。如果找不到文件 `hyphen.cfg`, 那么 iniTeX 将加载 `hyphen.ltx` 文件。

`hyphen.ltx` 文件加载 `hyphen.tex` 文件, 如果能找到它; 否则它会停止并出现错误, 因为没有连字符模式的格式不是很有用。然后设置 `\language=0`, 并设置美式英语所需的 `\lefthyphenmin=2` 和 `\righthyphenmin=3`。

因此, 如果希望加载任何其他模式, 则应创建一个文件 `hyphen.cfg`。对于要加载连字符模式的每种语言, 此文件应:

- 设置 `\language=<number>`;
- 加载包含该语言的连字符模式的文件。

如果您使用的模式需要一些定义 (definitions) 或赋值 (assignments), 则应使用组 (group) 将这些更改保留在其局部文件中 (keep such changes local to their file)。

注意 读入的连字符文件 (hyphenation files) 应仅使用 `\hyphenation` 和 `\patterns` 命令设置语言的连字符表 (hyphenation tables)。特别是, 它们不应该对小写/大写表 (lowercase/uppercase tables) (`\lccode` 和 `\uccode`) 进行赋值, 也不应该在读取文件后使用任何全局命令定义 (global command definitions)。不幸的是, 一些较旧的连字符文件确实包含此类设置; 因此, 它们与 L^AT_EX 用于确保输入编码和输出编码独立性的机制不兼容 (*incompatible*)

这个 `hyphen.cfg` 文件之后应该:

- 将 `\language` 设置为其默认值;
- 将 `\lefthyphenmin` 或 `\righthyphenmin` 设置为此默认语言 (default language) 的正确值 (correct values)。

有一些可用的宏包, 例如 “french”, 可以帮助您进行此配置。“babel” 集合 (collection) 包含许多设置多语言 L^AT_EX 格式的示例。`lthyphen.dtx` 中的文档 (`hyphen.ltx` 的源文件) 也包含一些有用的示例。

[我们打算在未来的 L^AT_EX 版本中提供一组用于配置连字符的标准命令。]

配置字体定义文件

如果您的位置 (site) 上有可用的特殊字体 (或某些字体不可用), 那么您可能需要生成自定义版本 (customised versions) 的字体定义文件 (font definition files); 这些文件的扩展名为 `.fd` 并由 L^AT_EX 读取, 以获取有关您系统中已安装的字体文件以及何时加载它们的信息。

虽然我们不鼓励这种自定义 (customisation), 但您可以在 `fntguide.tex` 文件中的文档化的源文件 (documented source file) `cmfonts.fdd` 和 [L^AT_EX 2_ε](#)

font selection【《L^AT_EX 2_ε 的字体选择》】中找到有关这些文件的内容 (content) 及其语法 (syntax) 的信息。[我们希望在未来的某个时候能够提供关于这个主题 (subject) 的更多信息和示例]

请注意，使用自定义字体定义文件会产生以下后果：

- 在您的系统上生成的文档充其量只能在不同的地方 (site) 上处理，在这个意义上说是可移植的——如果使用不同的字体，实际格式 (actual formatting) 将不一样。
- 如果这些问题不能以不使用任何配置文件的格式再现 (reproduced)，则 L^AT_EX 项目团队将无法支持您诊断这些问题。

还请注意，虽然标准字体定义文件 (standard font definition files) 的许可条件允许您制作自定义版本 (customised version) 供自己使用，但它们不允许您以原始文件名 (original file name) 分发 (distribute) 此类自定义字体定义文件 (customised font definition file)！

系统管理员请注意

如果您安装了带有局部配置字体设置 (locally configured font set-up) 的 L^AT_EX 版本，则此系统可能会生成不再“格式兼容 (formatting compatible)”的文档：例如，使用不同的默认字体很可能会产生不同的换行符 (line breaks) 和分页符 (page breaks)。如果您确实在多用户系统 (multi-user system) 上安装了一个配置方式不是“格式兼容”的系统，那么您应该仔细考虑用户需要创建可移植文档 (portable documents) 的需求。而满足用户这样需求的一个好方法是提供 L^AT_EX 的标准形式 (standard form)，而不需要任何“格式不兼容 (formatting incompatible)”的定制 (customisations)。

配置兼容模式

处理以 `\documentstyle` 开头的文档时，L^AT_EX 2_ε 尝试尽可能模拟旧的 L^AT_EX 2.09 系统。

latex209.cfg

每当 L^AT_EX 文档以 `\documentstyle` 而不是 `\documentclass` 类开头时, L^AT_EX 都会假定它是一个 L^AT_EX 2.09 文档, 因此会以“兼容模式 (compatibility mode)”处理它。这将执行以下操作:

- 设置 `\@compatibilitytrue` 标志;
- 输入 `latex209.def` 文件;
- 输入 `latex209.cfg` 文件 (如果存在)。

L^AT_EX 2.09 设置允自定义格式本身。使用 `iniTEX` 制作格式时, 该过程下面的请求 (request) 结束:

Input any local modifications here.[译: 在此处输入任何局部修改 (local modifications)。]

如果您的位置 (site) 当时输入了任何修改, 那么 L^AT_EX 2_ε “兼容性模式 (compatibility mode)”将无法完全模拟您位置上安装的 L^AT_EX 2.09。在这种情况下, 您应该将所有这些“局部修改 (local modifications)”放到一个名为 `latex209.cfg` 的文件中, 并将该文件放在位置 (site) 的默认输入路径 (default input path) 中。这些“局部修改”虽然没有存储在格式中, 但是将在处理任何旧式文档 (old-style document) 之前加载。这将确保您可以继续处理使用本地定制的任何旧文档。

标准包和类的配置文件

发行版 (distribution) 中的大多数包没有任何关联的配置文件 (configuration files)。此处列出了例外情况。

sfonts.cfg

`sfonts.cfg` 文件可以包含与幻灯片类 (slides class) 中字体使用相关的声明 (declarations)。如果这样的声明存在, 则读取它, 而不是读取 `sfonts.def` 文件。

请注意，使用此配置文件 (configuration file) 会产生以下后果：

- 由于幻灯片的字体设置 (font set-up for slides) 尚未修改，以适应现代用法 (modern usage)，因此该文件的内容应在某个时候能被完全更新。故编写此类配置文件的任何人都必须做好更新准备，以便在将来的版本中使用。
- 文档可移植 (portable) 仅仅是因为可以在不同的位置 (site) 处理——如果使用不同的字体，实际的格式 (actual formatting) 将不相同。
- 如果这些问题不能以不使用任何配置文件的格式再现 (reproduced)，则 L^AT_EX 项目团队将无法支持您诊断这些问题。

ltnews.cfg

`ltnews.cfg` 文件可用于定制 `ltnews` 类行为 (behaviour) 的某些方面；这个类用于排版每个 L^AT_EX 发行版附带的简报 (newsletter)。如果该文件存在，则在 `ltnews.cls` 文件的开头读入该文件。

ltxdoc.cfg

`ltxdoc.cfg` 文件可用于定制 `ltxdoc` 类行为 (behaviour) 的某些方面；这个类用于排版 `.dtx` 文件中的文档代码 (documented code)。如果该文件存在，则在 `ltxdoc.cls` 文件的开头读入该文件。

由于此文件是在加载 `article` 类之前读取的，因此可以将选项传递给 `article`。例如，可以将以下行添加到 `ltxdoc.cfg` 来将文档格式化为 A4 纸，而不是默认的美国信纸尺寸 (US letter paper size)。

```
\PassOptionsToClass{a4paper}{article}
```

但是，您应该注意，即使指定了纸张尺寸选项 (paper size options)，`ltxdoc` 类也总是将 `\textwidth` 参数设置为 355pt，以允许 72 列文本显示在 verbatim 代码列表中。如果您真的需要重写它，您可以使用：

```
\AtEndOfClass{\setlength{\textwidth}{...}}
```

在 `ltxdoc` 类的末尾将 `\textwidth` 设置为所需的值。

默认情况下，发行版中的大多数 `.dtx` 文档化代码文件 (documented code files) 都会生成一个“描述 (description)”部分，紧跟其后的是包的完整源列表 (source listing)。如果要禁止显示源列表，可以在 `ltxdoc.cfg` 中添加以下行：

```
\AtBeginDocument{\OnlyDescription}
```

`ltxdoc` 包的文档 (可以从 `ltxdoc.dtx` 文件中进行排版得到) 包含更多使用此配置文件的示例。

ltxguide.cfg

`ltxguide` 类由 \LaTeX 发行版中的“guide”文档 (如此文档) 使用。配置文件 `ltxguide.cfg` 可以与此类一起使用，其方式与前一节中描述的 `ltxdoc` 类的定制 (customisation) 非常相似。

其他受支持包的配置

“graphics(图形)”捆绑包 (bundle of packages) 需要两个配置文件 (configuration files)，主要用于指定用来处理 \LaTeX 生成的 `.dvi` 文件的驱动程序 (driver)。关于这些文件的更多文档随 `graphics` 捆绑包提供，但为了完整起见，我们在此提及它们。

graphics.cfg

通常，该文件只通过调用 `\ExecuteOptions` 指定默认选项，例如 `\ExecuteOptions{dvips}` 或 `\ExecuteOptions{textures}`。

此文件由 `graphics` 包读取，因此会影响捆绑包中基于 `graphics` 的所有宏包：`graphicx`、`epsfig`、`lscape`。

color.cfg

通常，此文件与 `graphics.cfg` 相同。它指定 `color` 的默认驱动程序选项 (default driver option)。

非标准版本

如果你觉得有必要制作一个不同于标准版本 (standard version) 的 \LaTeX 版本，而使用上述配置方法是不可能的，那么您应该首先阅读 `modguide.tex` 文件中的 *Modifying \LaTeX* 【《修改 \LaTeX 》】，这可能会让您意识到您没有这样的需求

因此，我们确信您将永远不需要创建非标准版本，并且，即使您确实创建了一个非标准版本，我们也希望您不会发布这样的版本。然而，您可以这样做，只要您特别注意以下几点：

- 尊重 `legal.txt` 和个人文件 (individual files) 中关于修改文件和更改名称的条件；
- 更改所有相关的 “`\typeout` 标题 (banners)”：即由您版本中的所有非标准文件和格式生成的标题；
- 确保用于运行您的版本的方法与用于运行标准 \LaTeX 的方法有明显区别：例如，使用与 `latex`(或 `LaTeX` 等) 明显不同的命令名 (command name) 或菜单项 (menu entry)。

示例

尽管我们心存疑虑，但是我们还是被自由编程协会 (League for Programming Freedom, LPF) 的成员提醒要记录如何做到这一点，因此在这里描述一个可能的 \LaTeX 修改 (modification)，以产生一个称为 `fs \TeX` 的系统似乎是合适的。

为此，您应该创建一个名为 `fsftex.tex` 的文件，然后使用 `ini \TeX` 和标准 \LaTeX 格式运行它。

`fsftex.tex` 文件的内容应该如第 14 页所示。您希望对 L^AT_EX 内核进行的特定修改需要添加到指定位置的文件中。您还可以选择要用于系统中的类和包文件的扩展名。

```

% fsftex.tex
%
% iniTeX Source code to make a `fsftex' format.
%
% To make this format on Unix:
%
%   initex &latex fsftex
%
% Then to run the format on file.tex:
%
%   tex &fsftex file
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% *** VERY IMPORTANT!!! ***
% Change the typeout banner so users know that they
%       are NOT running Standard LaTeX.
\everyjob{\typeout{fsfTeX 1.0 based on LaTeX2e \fmtversion}}
\makeatletter

% fsfTeX changes some LaTeX internals:
%   ... put what you like here ...
\def \fsf@xxxx {Some arbitrary \emph{freely modifiable} code goes here}

% fsfTeX class files have extension .fcl (this week):
\def \@clsextension {fcl}

% fsfTeX package files have extension .fsy:
\def \@pkgextension {fsy}

% Change the file handling so that when a fsfTeX package or class
% is not available, the standard LaTeX file will be read.
%
% For example, \documentclass{article} will load article.fcl if such
% a file exists, but article.cls otherwise. This allows arbitrary
% processing on `article' documents without changing the standard
% article.cls file.

\let\fsf@missingfileerror\@missingfileerror

\def\@missingfileerror#1#2{%
  \ifx #2\@clsextension

```

```

\InputIfFileExists {#1.cls}%
  {\wlog {fsfTeX: loading #1.cls rather than #1.#2.}}%
  {\fsf@missingfileerror {#1}{#2}}%
\else
  \ifx #2\@pkgextension
    \InputIfFileExists {#1.sty}%
      {\wlog {fsfTeX: loading #1.sty rather than #1.#2.}}%
      {\fsf@missingfileerror {#1}{#2}}%
  \else
    \fsf@missingfileerror {#1}{#2}%
  \fi
\fi
}

\makeatother
\dump

```