

关于 L^AT_EX 的提问的智慧

Ch'en Meng

2013 年 10 月 8 日

1 首先要说的——为了让你能将这篇文档看完

写这篇文档，我没有带着丝毫“看轻”或者“鄙视”提问者的心态；我，以及很多潜在的乐意帮助你解决问题的 T_EXer/L^AT_EXer 们，都能够理解你因为一段代码纠结住的急切心情，因为我们都是从新手一路走来，也遇到过和你一样的情况。

写这篇文档的目的，是为了帮助使用 L^AT_EX 的人，能够更加高效地提出自己的问题，并获得最大可能的解答。这有两个好处，首先自然是解决了提问者的问题，其次是一个“好问题”能够帮助遇到同样问题的后来人更简便地明白问题的原委并解决问题。

所以，我希望阅读到此的人，能够以友好的态度继续阅读下去。

文章的第2节介绍了怎样的问题才是清晰的，第3节介绍了对于 L^AT_EX 的提问至少需要提供哪些信息，第4节能帮助你将问题提得更好，最后的第5节介绍了注明的文档——提问的智慧。

2 如何才算是一个清晰的问题——提问题的方法论

不论提问者抱着何种样的态度提问（这个在后续会讲到），将问题表述清楚是最基本的要求。

在讨论这一小节的内容之前，允许我带着你回想一下我们自己解决问题时候所做的事情：

1. 明确知道自己的需求；
2. 了解问题的表现形式；
3. 探索问题发生的原因；

4. 寻找解决问题的方法；
5. 实践并验证它的效果。

这应当是一个很自然的结构，所以很多人在平时的生活中能够“一气呵成”地完成这些步骤而忽略了他们是有先后顺序的——前者是后者的前提。

提问者通常会在其中的第三步（或者第四步）遇到困惑，从而不知如何解决。（也许）在搜索网络和已有文档未果之后，ta 就会向他人寻求帮助。而基于上面那个有关“一气呵成”的假设，ta 往往会在“自己已经了解问题表现形式”的基础上去描述问题。这种行为的后果就是：提问者认为自己的提问很清晰了，但是潜在的回答者看不明白问题。

事实上提问并得到解决的过程，就是让他人代替你进行上述五个步骤的过程。因此你必须让潜在的回答者明了你的需求，以及明了你遇到的问题表现形式。

3 对于 L^AT_EX 来说应该怎么办——提问应当提交哪些内容

你的代码没有出错，但是没有达到预期效果 对于这种情况，你应当

- 将理想中的效果描述清楚；
- 将当前的效果截图或将输出文件（.pdf 之类的文档）上传；
- 将能够直接编译就出现图示效果的代码上传。

你的代码出错了 对于这种情况，你应当在之前三条的基础上，再补充以下内容

- 提供编译产生的错误提示（通常以叹号开头，下面接着以字母 1 开头的分为两行的代码）。

4 如何调整你的提问——让潜在的回答者有回答的欲望

精简 一个出现几十行甚至上百行的代码或者日志文件内容的问题，是极其让人厌烦的。

对于代码，你应当提供一份“最小工作示例”或称“最小问题代码”：直接编译（无需添加任何东西）就能重现你所遇到的问题；没有与问题无关

的其他输出。比如有提问者提供了以下代码，就是合格的“最小工作示例”：

```
\documentclass{minimal}
\usepackage{amsmath}
\listfiles
\begin{document}
\[
y = \begin{cases}
a& b;\backslash
c& d.
\end{cases}
\]
\end{document}
```

在这里出现了两个对于你可能并非熟悉的东西。`minimal` 文档类，顾名思义，是一个能工作的做小文档类——事实上它只提供了一个 \LaTeX 内核。这样做的好处是可以避免文档类的设置对问题判断带来的影响，从而更加精确地定位问题。此外 `\listfiles` 命令可以列出编译文档所用到的所有文件以及版本，协助判断问题。比如上面的代码就会因这条命令在日志文件内写入下面的内容。

```
*File List*
minimal.cls      2001/05/25 Standard LaTeX minimal class
amsmath.sty      2013/01/14 v2.14 AMS math features
amstext.sty      2000/06/29 v2.01
amsgen.sty       1999/11/30 v2.0
amsbsy.sty       1999/11/29 v1.2d
amsopn.sty       1999/12/14 v2.01 operator names
*****
```

当然，这是一份十分专业的“最小工作示例”。通常提问者并不需要提供如此专业的“最小工作示例”就能获得满意的答案（例如此例中，只需要使用 `article` 文档类即可，`\listfiles` 也非是必须）。

总结一下，最小工作示例要求：

- 别人拿到直接就能编译并重现你所遇到的问题，而不需要向其中添加任何东西；
- 不要有冗余信息，出现几十行甚至上百行的代码是“反人类”的。

不论如何，一定不要以截图的形式给出代码；相反对于 *log* 文件关键部位的截图是能够接受的。

文档 大多数的问题，文档都能为你提供标准的解答。因此在你提问之前，你最好先去阅读一下相关的文档，至少确保你阅读过一份完整的入门文档¹。如果你能够在你提问的同时表明自己已经阅读过文档，但是依旧留有困惑，潜在的回答者会更加愿意为你解答。

中文 \TeX 社区三个著名的文档：新手请先读我， \CTeX FAQ, China \TeX Math FAQ.

搜索 除非是特别的需求，通常你遇到的问题都有人遇到过了（这也是问题能够被轻易解决的基础），因此和你相同或类似的提问可能已经出现过了。你最好是先在互联网上搜索相关问题，然后再进行提问；或至少如前段所述，表明你有经过搜索这一过程。

中文 \TeX 社区： \CTeX , China \TeX , \LaTeX Studio.

排版 \LaTeX 本身是一个排版工具，对此略有心得的人恐怕对排版都多少有点“洁癖”，至少对于乱七八糟的排版他们可能很难提起兴趣。

在论坛中提问，你应该确保你的排版足够简洁，不会因为排版让你的提问看起来混乱不堪。此外各论坛可能还有他们的礼节，请仔细阅读板块置顶。

在腾讯 QQ 群中提问，你至少应该确保你的字体不要太大——群是公共场所，大字体刷屏容易惹人厌烦。

5 提问的智慧——一份著名的关于提问的文档

这是一篇最先由 Eric Steven Raymond 撰写的文档，说明了提问者应当如何提问，使得回答者能够更容易理解并能使提问者学到更多东西。

这里是它的英文版和中译版。

¹例如你确定你能够编译出类似本文的简单文档、你清楚地知道命令和环境的差别、你清楚地知道“导言区”是什么东西。