

$\text{\textcolor{violet}{T}}\text{\textcolor{blue}{E}}\text{\textcolor{blue}{X}}$ $\text{\textcolor{blue}{L}}\text{\textcolor{blue}{A}}\text{\textcolor{blue}{T}}\text{\textcolor{blue}{E}}\text{\textcolor{blue}{X}}\ 2\epsilon$ $\text{\textcolor{blue}{M}}\text{\textcolor{blue}{E}}\text{\textcolor{blue}{T}}\text{\textcolor{blue}{A}}\text{\textcolor{blue}{F}}\text{\textcolor{blue}{O}}\text{\textcolor{blue}{N}}\text{\textcolor{blue}{T}}$
 $\text{\textcolor{black}{B}}\text{\textcolor{black}{i}}\text{\textcolor{black}{b}}\text{\textcolor{black}{T}}\text{\textcolor{black}{E}}\text{\textcolor{black}{X}}$

如何利用 $\text{\textcolor{red}{T}}\text{\textcolor{red}{E}}\text{\textcolor{red}{X}}$ 高效编写论文文档

余恒

北京师范大学天文系

2012 年 11 月 6 日

什么是 T_EX

- 1040 年左右北宋毕昇发明活字印刷术，1440 年德国古登堡发明金属活字印刷机。但无论是雕版印刷还是活字印刷都需要在排版上耗费大量的人力。得益于阿波罗登月计划，计算机在上世纪六十年代开始进入工业界。计算机排版系统的出现给印刷出版业带来了全新的变革。
- 电子排版系统可以方便的实现内容编辑、字体更改、版面设计等功能。无需实体印刷就可以校对最终的印刷效果。极大的降低了劳动强度，缩短了出版周期。
- T_EX 就是这样一套电子排版系统，它对数学公式的处理效果公认是业界最好的。其它常见的电子排版软件包括 Adobe Indesign、Microsoft Publisher、方正书版等。

T_EX 由来 -缘起

1962 年，斯坦福大学计算机系博士生高德纳（Donald E. Knuth）开始写一本关于编译器的书。但初稿篇幅远超出出版社的预想，于是他们启动了一个多卷本的编写计划。

- 1968 年，《计算机程序设计艺术》第一卷《基本算法》出版。
- 1969 年，第二卷《半数字化算法》。
- 1973 年，第三卷《排序与搜索》、第一卷第二版。
- 1974 年，荣获图灵奖，是历史上最年轻的获奖者。
- 1975 年，当选为美国国家科学院院士。
- 1976 年出版第二卷第二版时采用了计算机排版技术，效果很不理想。失望的 Knuth 暂停了出版计划，决心设计一个简洁优美的排版软件。

T_EX 由来 - 执念

- 1977 年, 开始构造 T_EX 系统 (在希腊语中这个词的意思是“科技”和“艺术”), 并为 T_EX 设计了一个字符字体生成软件: *METAFONT*
- 1982 年, TeX 系统正式公布, 高德纳也如愿出版了《计算机程序设计艺术》的第二卷第二版。
- 1984 年, 出版 TeX 权威参考书《The TeXbook》。
- 1992 年, 54 岁申请退休, 以便专心写书。
- 1993 年, 宣布不再对 TeX 和 METAFONT 进行更新。
- 2006 年, 被诊断出患前列腺癌, 手术成功。
- 2011 年, 第四卷第一部分……

T_EX 由来 - 善果

Knuth 使用无理数 π 的近似值作为 T_EX 的版本号, e 的近似值作为 METAFONT 的版本号。每升级一次, 其版本号就增加一位数字。到目前为止, T_EX 系统的版本序号是 3.141592, ME-TAFONT 版本序号为 2.71828。

在 1986 年出版的《T_EX : The Program》的序言中他这样写道:

"I believe that the final bug in TeX was discovered and removed on November 27, 1985. But if, somehow, an error still lurks in the code, I shall gladly pay a finder's fee of \$20.48 to the first person who discovers it. (This is twice the previous amount, and I plan to double it again in a year; you see, I really am confident!)"

T_EX 和 Word 比较

- Word 是字处理软件，使用自有的文件格式（doc、docx），并不能保证在不同的系统和平台上有同样的显示效果。而 T_EX 是电子排版软件，使用独立于设备的印刷文件格式（dvi、ps、pdf），确保在所有平台上得到完全相同的输出效果。
- Word 是“所见即所得 (WYSIWYG)”的，修改的字体，颜色，用鼠标画出表格，都可以直接看到效果。屏幕上显示的页面同本机打印的效果一致。这类软件所有对页面的控制和设定都是隐性的、不直接可见。T_EX 输入的是文本文件，需要编译之后才能得到排版文件。因为所有的格式设定都写在文本文件中，有经验的用户能够精确地控制最终效果。

思路上的区别

用 \TeX 写文章，重点在逻辑结构和段落组织

- 这一段是否应该归入上一节？
- 这节内容太多了，还是拆成两部分吧
- 这个数据的出处是哪篇文章？回头记得加上
- 是否应该开始新的一章呢？ ...

用 Word 要花很多时间调整格式：

- 这是第 3 章，标题应该用黑体三号字，正文用楷体小四
- 这一节要换个位置，希望章节、图片、公式的编号不要乱掉
- 参考文献一个个编号改格式太麻烦了，还是少引一些吧
- 这个符号好奇怪，该怎么输入？ ...

公式效果

T_EX 输出的公式:

$$\sum_{p \text{ prime}} f(p) = \int_{t>1} f(t) d\pi(t).$$

Word 2007 的公式效果:

$$\sum_{p \text{ prime}} f(p) = \int_{t>1} f(t) d\pi(t).$$

T_EX 公式输入也比较方便。比如输入上面的公式的语句如下:

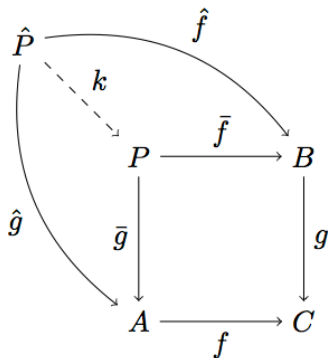
```
\$ \sum_{p \rm \; prime} f(p) = \int_{t>1} f(t) {\rm d} \pi(t). \$
```


复杂公式

很多符号，在 Word 里不容易找到，在 $\text{T}_{\text{E}}\text{X}$ 输入则非常方便：

$$2 \uparrow\uparrow k \stackrel{\text{def}}{=} 2^{2^{\cdot^{\cdot^{\cdot^2}}}} \}^k.$$

你还可以输入更加复杂的公式：



段落中的公式

Word 2007 中的公式对象能够对齐所在行，但会撑大行距：

The overdensity of optical galaxies in clusters is:

$1 + \sigma_o = N_o / \langle N \rangle$, with corresponding Poisson uncertainty

of: $\sigma_{\delta_o}^2 \cong \Delta_o [1 + \sigma_b^2 \Delta_o]$, where $\Delta_o = \frac{(1+\delta_o)^2}{N_o}$ and ...

TEX 对接工整，行距不变：

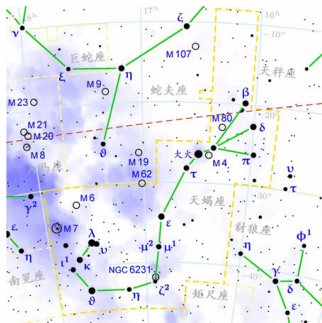
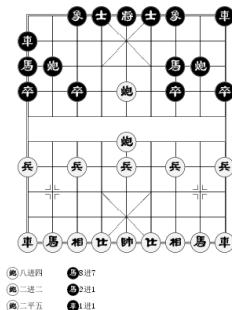
The overdensity of optical galaxies in clusters is:

$1 + \sigma_o = N_o / \langle N \rangle$, with corresponding Poisson

*uncertainty of: $\sigma_{\delta_o}^2 \cong \Delta_o [1 + \sigma_b^2 \Delta_o]$ where $\Delta_o = \frac{(1+\delta_o)^2}{N_o}$
and ...*

T_EX 还能做什么？

T_EX 是一个非常多才多艺的程序。它不但可以编辑论文，书籍，试卷，幻灯片，学术杂志，还可以编辑乐谱、星图、电路图、象棋棋谱、围棋棋谱……



更多的例子可参见: <http://www.tug.org/texshowcase/>

T_EX 的优点

- **高质量的输出** T_EX 遵循传统的排版规则，以排版的质量为最重要的目标。所有细节都可以被精确地定义。
- **极其稳定** 十几年前的 T_EX 文件用现在的系统编译得到的结果与当年一样。而且 T_EX 系统极少会崩溃。
- **简单方便** T_EX 文档是文本文件。即使没有 T_EX 系统，也可以看懂绝大部分的内容，并用任何文本编辑器进行修改。
- **可编程** T_EX 是一种宏命令编程语言。你可以用很少的命令来完成非常复杂的工作。
- **良好的通用性** T_EX 支持几乎所有的操作系统和计算平台。
- **自由免费** T_EX 是免费软件，它的源程序可以自由获取。

T_EX 的缺点

- 不是所见即所得的编辑模式，不够直观。
- 命令很多，不容易记忆。要多用，多查帮助手册。
- 没有逻辑结构和固定样式的内容不适合用 T_EX 编辑，比如报纸，画报，广告等。
- 编译出现错误时不容易解决，需要经验。
- 扩展定制 T_EX 宏包需要有一定的编程基础。

- **TeX** 高德纳开发的原始版本，也称作“plain TeX”。所有其它的引擎、宏包、套件都是在它基础之上开发的。
- **LaTeX** TeX 主要针对高级排版和程序设计人员，普通用户使用仍有难度。1984 年，计算机学家 Lamport 开发了一套更易用的命令集。将版面设计与文稿内容分开处理。现在的版本是 LaTeX2e。
- **PDFTeX** TeX 比 pdf 格式的诞生早了 10 余年, TeX 系统的输出文件格式为.dvi (device independent). 而 PDFTeX 可以输出 dvi 和 pdf 两种格式。
- **ConTeXt** ConTeXt 是独立于 LaTeX 的 TeX 宏包。配置使用上更加灵活。但二者不兼容，命令格式不能混用。

- **XeTeX** TeX 不仅早于现代所有主要的字体技术 (比如 TrueType, OpenType, PostScript Type 1 等), 而且也早于现有的操作系统。因此, 原始的 TeX 只能使用自己定义的字体格式。XeTeX 通过增强 TeX 的字体控制功能使其可以使用操作系统已有的各种字体。
- **LuaTeX** TeX 使用自己的语言系统, 许多当代用户很难掌握, LuaTeX 通过引入一个现代的脚本语言, Lua, 来解决这个问题。
- **Omega** 这是一个几乎是完全重新写过的, 支持 Unicode 的 T_EX 程序。可以处理非拉丁语系的文本. 也支持从右到左, 从上到下的排版方式。

常见发行版

- **TeXLive** 跨平台系统，可以直接在光盘上运行，也可以安装在本地磁盘上。
- **MiKTeX** Windows 下最流行的 Latex 套件，定制灵活，功能齐全。
 - **CTeX** 中文 TeX 套件，提供了完整的中文解决方案。
 - **ProTeX** MiKTeX 的英文增强版本，2009 年停止更新。
 - **MiCTeX** MiKTeX 的中文定制版本，2009 年停止更新。
- **W32TeX** 对 XeTeX、LuaTeX 支持较好，Win 下相应的编译工具均来自这个项目。
- 所见即所得的 LaTeX 编辑器：
Scientific Workplace、LyX、BaKoma、etc.

参考图书

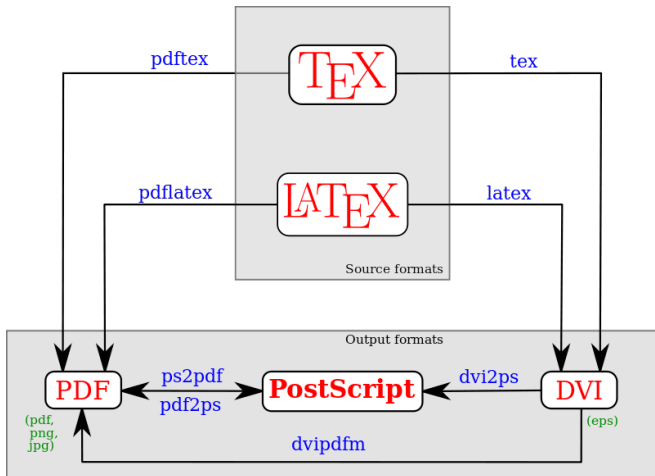
- 一份不太简短的 LATEX2 介绍 v4.2
- 《雷太赫排版系统简介》包太雷 2010
- 《The Latex Companion》2nd Frank Mittelbach 2004

资源社区

- 校内 FTP: <ftp://202.112.85.101/> 相关目录为
/Soft/Pub2/Science/TeX&LaTeX/TeXsoftware/Systems/
- 中文 CTeX 主站: <http://www.ctex.org/>
- 水木社区 TeX 版: <http://www.newsmth.net/>
- StackOverflow 的 TeX 分站: <http://tex.stackexchange.com/>

T_EX 文档的处理流程

把编辑好的 L^AT_EX 文档使用 *latex* 命令编译。编译流程如下：



源文件结构

L^AT_EX 所处理的源文件需要遵从一定的结构：

```
\documentclass{...} % 指定所写文档的类别。  
\usepackage{...} % 调入宏包，为系统增加功能。  
\begin{document} % 文档正文由此开始  
\title{这里是标题}  
\author{作者}  
\date{日期}  
\maketitle % 生成标题  
这里输入正文和各种 Latex 命令  
\chapter{章}  
\section{节}  
\subsection{小节}  
\end{document} % 文档到此结束，后面的部分都会被忽略。
```

文档布局 I

当 \LaTeX 处理源文件时，首先需要知道的是作者所要创建的文档类型。该信息可以通过命令 `documentclass` 来提供给 \TeX 。

```
\documentclass[options]{class}
```

常见的 class 类有：

`article` 排版科技期刊、短报告、程序文档、邀请函等。

`report` 排版多章节的长报告、短篇的书籍、博士论文等。

`book` 排版书籍。

`slides` 排版幻灯片。其中使用了较大的 sans serif 字体。

`beamer` 最流行的 \TeX 幻灯片宏包。本讲座就是用 `beamer` 生成。

文档布局 II

常见的 option 选项有：

12pt 设置文档所使用的字体的大小。如果没有声明任何选项，缺省将使用 10pt 字体。

a4paper 定义纸张大小，缺省设置为 letterpaper。此外，还可以使用：a5paper, b5paper, executivepaper 和 legalpaper。

twoside 指定 LATEX 排版的文档为双面或单面格式 (oneside)。
article 和 report 缺省使用单面格式，而 book 则缺省使用双面格式。

fleqn 设置该选项将使数学公式左对齐，而不是中间对齐。

leqno 设置该选项将使数学公式的编号置于左侧。

表格样例

left	x_1	x_2	x_3	x_4
A	a1	a2	a3	a4
B	b1	b2	b34	

```
\begin{table}[h]
  \begin{tabular}{|c|c|c|c|c|} \hline
    left &  $x_1$  &  $x_2$  &  $x_3$  &  $x_4$  \\ \hline
    A & a1 & a2 & a3 & a4 \\ \hline
    B & b1 & b2 & \multicolumn{2}{|c|}{b34} \\ \hline
  \end{tabular}
\end{table}
```

插入图片



插入图片的代码如下：

```
\includegraphics[height=0.6\textheight]{knuth.jpg}
```

其它参数还包括旋转 (angle), 裁切 (trim), 缩放 (scale) 等等。

特殊字符

■ 拉丁字符

α	β	Γ	Δ	ϵ	ζ	η
<code>\alpha</code>	<code>\beta</code>	<code>\Gamma</code>	<code>\Delta</code>	<code>\epsilon</code>	<code>\zeta</code>	<code>\eta</code>

■ 数学符号

\neq	\approx	∞	\int	∂	\sum	\forall
<code>\neq</code>	<code>\approx</code>	<code>\infty</code>	<code>\int</code>	<code>\partial</code>	<code>\sum</code>	<code>\forall</code>

■ 其它字符

\vec{x}	\check{x}	\hbar	\odot	\aleph	\S	\dagger
<code>\vec{x}</code>	<code>\check{x}</code>	<code>\hbar</code>	<code>\odot</code>	<code>\aleph</code>	<code>\S</code>	<code>\dag</code>

数学公式

这是一个行内公式的例子 $\int_0^\infty \frac{\sin x}{x} dx = \pi/2$

`$\int_0^\infty \frac{\sin x}{x} \mathrm{d} x = \pi/2$`

这是个行间公式的例子

$$\int_0^\infty \frac{\sin x}{x} dx = \frac{\pi}{2} \quad (1)$$

这是个行间公式的例子

`\begin{equation}`

`\int_0^\infty \frac{\sin x}{x} \mathrm{d} x = \frac{\pi}{2}`

`\end{equation}`

原生引文格式

代码内容:

```
\begin{thebibliography}{6}  
\bibitem{mascot} H R Simpson and K Jackson, {\em Process  
Synchronisation in MASCOT}, The Computer Journal,  
{\bf 22}, 332, 1979  
.....  
\end{thebibliography}
```

输出结果:

References

- [1] H R Simpson and K Jackson, *Process Synchronisation in MASCOT*, The Computer Journal, **22**, 332, 1979
- [2] D A Troy and S H Zweben, *Measuring the Quality of Structured Designs*, Journal of Systems and Software, **2**, 1981, 113–120

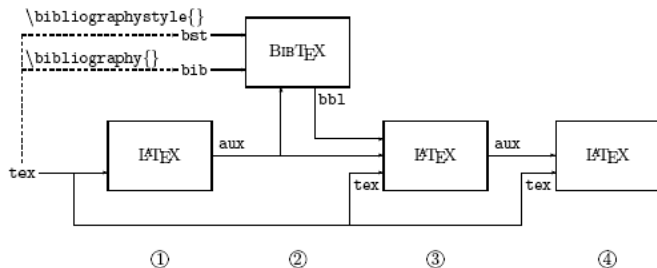
引用方式

最常用的参考文献引用宏包是 natbib,

```
\citet{jon90} % --> Jones et al. (1990)
\citep{jon90} % --> (Jones et al., 1990)
\citet*{jon90} % --> Jones, Baker, and Williams (1990)
\citep*{jon90} % --> (Jones, Baker, and Williams, 1990)
\citep[chap. 2]{jon90} %--> (Jones et al., 1990, chap. 2)
\citep[see][{}]{jon90} %--> (see Jones et al., 1990)
\citep[see][chap. 2]{jon90}
    %--> (see Jones et al., 1990, chap. 2)
\citet{jon90,jam91}
    %--> Jones et al. (1990); James et al. (1991)
\citep{jon90,jam91}
    %--> (Jones et al., 1990; James et al. 1991)
```

显然，手工编辑引用条目 `bibitem` 的办法耗时费力，一旦调整引用顺序或者更换投稿期刊就需要重复大量的编辑工作。于是，我们需要一个自动化的系统可以根据引用出现的顺序和作者的名字来对参考文献排序，并且根据期刊的格式要求按规范输出引文条目。这就是 BibTeX。

对于包含引用文献的 \LaTeX 文档编译流程如下：



Bib 条目范例

Bib 文件包含的条目种类非常齐全，可以是 book, proceedings, phdthesis 等等，每种类型都有相应的信息项，不过我们不用手工填写这些内容，主流文献数据库都提供导出服务。本地的 bib 文件管理和编辑工作可以通过文本编辑器或者 JabRef 软件。

```
@ARTICLE{SHELL02,  
  AUTHOR =      "Michael Shell",  
  TITLE =      "How to Use the {\IEEEtran \LaTeX} Class",  
  JOURNAL =     "Journal of {\LaTeX} Class Files",  
  YEAR =       2002,  
  VOLUME =      12,  
  NUMBER =      4,  
  PAGES =       "100--120"  
}
```

引用样式

参考文献的样式由命令 `\bibliographystyle{}` 定义，可选参数包括

plain Y. Linde, A. Buzo and R.M. Gray. An algorithm for vector quantizer design. IEEE Transactions on Communications, 28(1):84-95, January 1980.

acm LINDE, Y., BUZO, A., AND GRAY, R.M. An algorithm for vector quantizer design. IEEE Transactions on Communications 28, 1 (January 1980), 84-95.

ieeetr Y. Linde, A. Buzo and R.M. Gray, "An algorithm for vector quantizer design," IEEE Transactions on Communications, vol. 28, pp. 84-95, January 1980.

通常各个期刊会单独提供引文样式文件以供调用。

北京师范大学模版

北京師範大學

博士学位论文

论文题目: 北京师范大学学位论文 L^AT_EX 模版

作者: 某某某
导师: 某某某 教授
Stephano Prof.
系别年级: 天文系 2010 级
学号: 0812160001
学科专业: 天体物理
完成日期: 2012 年 5 月

北京师范大学研究生院

国内许多大学都提供的 T_EX 论文模版, 比如清华、北大、哈工大、同济等。天文系 08 届硕士生高爽曾根据清华的论文模版做了一个北师大学位论文模版。但在他毕业后就无人维护了。我在 2010 年 9 月基于清华大学论文 L^AT_EX 模版 THUThesis 定制了一个新的版本。
<http://gerry.lamost.org/blog/?p=811>.

- 2010-11-28 BNUThesis 发布 1.0 版
- 2011-3-18 更新为 1.3。增加学术成果部分；调整页边距，取消奇偶页差别，需要在装订时处理；增加目录小节缩进。
- 2011-4-8 更新为 1.4。修正附录中的图表公式编号格式。附录采用“附 1.*”的格式。更新参考文献样式文件。但由某些少见的参考文献类型生成的引用格式可能不够规范，比如报纸、专利等。
- 2012-04-10 更新为 1.5。默认不生成空白页。增加 twoside 选项，使双面打印时正反页面对齐。全面测试兼容性，支持 CTeX2.9 和 TeXLive2011 以上的版本。推荐使用 XeLaTeX 编译。打包 PDFLaTeX 编译所需的 ccmmap.sty 文件。

- `readme` 基本说明文档
- `bnuthesis.cls` 模版样式文件，字体格式在此定义
- `bnuthesis.cfg` 模版样式配置文件，声明等规范文本在此定义
- `bnutils.sty` 模版附加宏包，加入自定义命令和额外宏包
- `bnubib.bst` 参考文献样式文件
- `main.tex` 主控文档，定义文档类型，控制编译范围
- `shuji.tex` 书脊文档
- `data/` 论文具体内容，建议按章分成单独文件
- `ref/` 参考文献目录，如数目较多可按章节分开
- `figures/` 图片目录

主控文档 I

```
\documentclass[doctor]{bnuthesis}
% \documentclass[%
%   bachelor/master/doctor, % mandatory option
%   xetex/pdftex/dvips/dvipdfm, % optional
%   twoside, % 是否双面打印, 默认为单面
%   secret, % 是否涉密
%   arialtoc,arialtitle]{bnuthesis}
% 所有其它宏包都统一放在这里, 可根据实际需要添加或者删除。
\usepackage{bnutils}
% 可以在这里修改配置文件中的定义, 导言区可以使用中文。
% \def\myname{薛瑞尼}

\begin{document}
```

主控文档 II

```
\graphicspath{{figures/}} % 定义图像文件目录
%% 封面部分
\frontmatter
\input{data/cover}
\makecover
\tableofcontents % 目录
% \input{data/denotation} % 符号对照表
%\listoffigures % 插图索引
%\listoftables % 表格索引

\mainmatter %% 正文部分
\include{data/chap01}
\include{data/chap02}
\backmatter %% 其它部分
```

主控文档 III

```
% 参考文献
\bibliographystyle{bnubib}
\bibliography{ref/refs}

% 附录
\begin{appendix}
\input{data/appendix01}
\end{appendix}


\include{data/paper}    % 学术成果
\include{data/ack}     % 致谢
\end{document}
```

编译方式

通常可使用 L^AT_EX 编译示例文档，支持 eps、ps 格式图像文件，具体步骤为：latex main -> bibtex main -> latex main -> latex main -> dvips main.dvi -> ps2pdf main.ps

也可以使用 pdfLaTeX/dvipdfm(x) 生成文档，支持 jpg、png 等通用图像格式，但 eps 文件需转为 pdf 格式。在 CTeX 套装或者 TeXLive 发行版下推荐使用 XeLaTeX 编译，但需要事先安装 Adobe 字体。

注意保存所有文本文件为 UTF-8 编码格式，否则无法通过编译！

适用范围

- 论文包含大量公式、代码、反应方程
- 论文涉及大量文献引用
- 作者熟悉计算机、有一定编程基础

现有问题：

- 无法直接提交学术不端检测系统
- 缺乏足够的技术支持

$$\top \hbar \grave{a} \cap \mathbb{k} \ \gamma \otimes \cup$$

`$\top \hbar \grave{a} \cap \mathbb{k} \ \gamma \otimes \cup$`