

Texlive 的安装 (xelatex 中文字体和手工宏包加载)

1 Texlive 的安装

以 ubuntu18.04 为例

1. 访问以下网址下载 texlive 的 ISO 文件

<https://mirrors.tuna.tsinghua.edu.cn/#>

wget -c https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/Images/texlive2020-20200406.iso

2. 安装 perl 组件

终端运行以下命令

sudo apt-get install perl-tk

3. 加载该 ISO 文件

sudo mount -o loop texlive2018.iso /mnt 换掉文件路径即可)

注意: 使用该命令会出现错误提示, mount: /dev/loop1 is write-protected, mounting read-only. 不必管它

4. 启动图形化安装界面

cd /mnt

sudo ./install-tl -gui

点击安装即可

安装时间比较长耐心等待, 安装完成后执行以下操作

5. 为了支持中文, 我们需要使用 usepackage{xecjk} 包, 所以需要安装:

sudo apt-get install texlive-lang-chinese

6. 要使用更多软件包和字体进行更完整的安装运行以下命令

sudo apt-get install texlive-latex-base texlive-latex-extra texlive-latex-recommended texlive-fonts-recommended

7. 安装 XeLatex

sudo apt-get install texlive-xetex

8. 安装 texstudio 编辑器 sudo apt-get install texstudio

至此 texlive 的安装就完成了, 可以正常输出中文进行编译.

2 Windows 字体安装

1. 拷贝中文字体进入 /usr/share/fonts/Windows 里面, 没有此文件夹自己创建

常用的 6 个字体是: 宋体 (simsum.ttf)、仿宋 (simfang.ttf)、黑体 (simhei.ttf)、楷体 (simkai.ttf)、隶书 (simli.ttf)、幼圆 (simyou.ttf)

2. 执行如下命令: `$ cd /usr/share/fonts/Windows`
`$ chmod 777 *` # 因为原权限为 600, 改为 644/777
`$ mkfontscale`
`$ mkfontdir`
`$ fc-cache`

3. 测试字体是否安装完毕 `$ fc-list`

如果有上面的字体就是安装成功了

3 手工添加缺失宏包并生效

- 方法一: 手动添加宏包, 在以下任意一个目录下放上.sty 文件 (宏包对应文件)

1. `/usr/local/share/texmf/`
2. `/var/lib/texmf:/usr/share/texmf/`
3. `/usr/share/texlive/texmf-dist/` #默认建议用`/usr/share/texlive/texmf-dist/tex/latex/`

然后执行命令

`sudo texhash` # 建议具体到.sty 所在目录下用 * 标记: `sudo texhash /usr/share/texlive/texmf-dist/tex/latex/*`

默认建议命令为: `sudo texhash /usr/share/texlive/texmf-dist/tex/latex/*`

然后理应就找到了

- 方法二: 进入/home 目录, 在/home 目录下创建如下文件夹 `texmf/tex/latex`, 假设我们添加的宏包叫做 `foo.sty`, 就创建名字为 `foo` 的文件夹:

1. `cd ~`
2. `mkdir -p texmf/tex/latex/foo`
3. 将 `foo.sty` 放入 `foo` 文件夹中

再执行下面的命令

`texhash /texmf/*` # 建议具体到.sty 所在文件
 就没事儿了

命令 `texhash` 与 `mktextlsr` 效果相同, 但区别是:

- `texhash` 针对的是目录
- `mktextlsr` 针对的是具体文件

所以建议两个命令都用!

4 常用的编辑工具

- pdf 编辑: `pdfarranger`
- 图片编辑: `gimp`
- 摄像头驱动: `gucvview`
- 在线视频: `obs-studio`

- Origin: labplot
- 视频编辑: kdenlive
- cooledit: audacity
- 千千静听: qmmp

5 Virtualbox

VirtualBox kernel driver not installed.

The vboxdrv kernel module was either not loaded or /dev/vboxdrv was not created for some reason.
‘/etc/init.d/vboxdrv setup’

as root.

VBox status code: -1908 (VERR_VM_DRIVER_NOT_INSTALLED)

按照提示在终端里输入

```
sudo /etc/init.d/vboxdrv setup
```

仍然不行。给的信息是:

Stopping VirtualBox kernel modules ...done.

Recompiling VirtualBox kernel modules ...failed!

(Look at /var/log/vbox-install.log to find out what went wrong)

打开/var/log/vbox-install.log 文件

Makefile:73: *** Error: unable to find the sources of your current Linux kernel. Specify KERN_DIR= and run Make again.

更新的时候升级了 Linux kernel, 所以 vboxdrv 需要重新编译。因为是从源里面直接更新的, 没有 Linux kernel 的源文件, 显然编译需要这些源文件, 最后解决方法如下

```
sudo apt-get install linux-headers-$(uname -r)
```

```
sudo apt-get install dkms
```

```
sudo /etc/init.d/vboxdrv setup
```

6 Ubuntu 下的有线网络链接

有线宽带在 Windows 下可以正常连接, 不过在 Ubuntu 下死活连不上, 基本可以排除宽带线路问题和机器的网卡硬件问题, 所以问题应该是出在 Ubuntu 系统设置或驱动上。解决方式:

1. 启动 NetworkManager:

```
sudo vim /etc/NetworkManager/NetworkManager.conf
```

```
# 修改 managed=true
```

2. 重启服务

```
sudo service network-manager restart
```

3. ifconfig 查看网卡

```
# 查看启用的网卡: ifconfig
```

查看所有网卡: `ifconfig -a`

一般宽带使用的为 `enp` 开头的网卡 (如: `enp7s0`), 如果网卡未启用, 则通过 `sudo ifconfig enp7s0 up` 启用网卡 (`down` 为关闭)

4. 通过 `nmcli` 命令查看网卡状态, 有没有被 `network-manager` 所管理:

```
enp7s0: 已连接 to ChinaNet
"Realtek RTL8111/8168/8411 PCI Express Gigabit Ethernet Controller"
ethernet (r8169), CA:08:95:9F:15:30, 硬件, mtu 1500
ip4 默认, ip6 默认
inet4 192.168.1.6/24
route4 0.0.0.0/0
route4 192.168.1.0/24
route4 169.254.0.0/16
inet6 2409:8a20:a9a:de80:8282:eed:18c6:8728/64
inet6 fe80::7083:aea7:4870:dfc/64
route6 2409:8a20:a9a:de80::/64
route6 ::/0
route6 ff00::/8
route6 fe80::/64
route6 fe80::/64

br-546b60ee6b1: 已连接 to br-546b60ee6b1
"br-546b60ee6b1"
bridge, 02:42:29:87:cb:68, 软件, mtu 1500
inet4 172.18.0.1/16
route4 172.18.0.0/16

docker0: 已连接 to docker0
"docker0"
bridge, 02:42:44:bc:10:3c, 软件, mtu 1500
inet4 172.17.0.1/16
route4 172.17.0.0/16

74:60:fa:15:07:ad: 已断开
"HUAWEI P30 Pro-Lyon"
1 连接可用
bt (bluetooth), 74:60:fa:15:07:ad, 硬件

wlp0x0: 已断开
"Intel 200 Series PCH PCI Express Root Port"
1 连接可用
wifi (iwlwifi), 94:e6:f7:a2:51:ac, 硬件, mtu 1500

vmnet1: 未托管
"vmnet1"
ethernet (unknown), 00:50:56:c0:00:01, 硬件, mtu 1500

vmnet8: 未托管
"vmnet8"
ethernet (unknown), 00:50:56:c0:00:08, 硬件, mtu 1500 知乎 @Lyon
lines 1-45
```

如果 `enp7s0` 显示为未托管状态, 则需要修改配置文件

备份原配置文件

`sudo mv /usr/lib/NetworkManager/conf.d/10-globally-managed-devices.conf /usr/lib/NetworkManager/conf.d/10-globally-managed-devices.conf_origin`

新建新配置文件

`sudo touch /usr/lib/NetworkManager/conf.d/10-globally-managed-devices.conf`

修改后重启 `networkmanager` 服务:

`sudo service network-manager restart`

再用 `nmcli` 命令查看, 会发现 `enp7s0` 已经被 `network-manager` 所管理, 在系统 → 设置 → 网络中发现, 多出来了“有线”选项, 表示可以联网了

