

Manjaro 指南

冯振华

2021/12/07 v1.0.3 *

摘要

This edition of Manjaro includes the GNOME desktop, which is part of the GNU project. It offers an unconventional but intuitive desktop aiming to help productivity. It comes with a very complete and aesthetically coherent set of applications to fill the most common needs. GNOME edition disc image files start with manjaro-gnome.¹

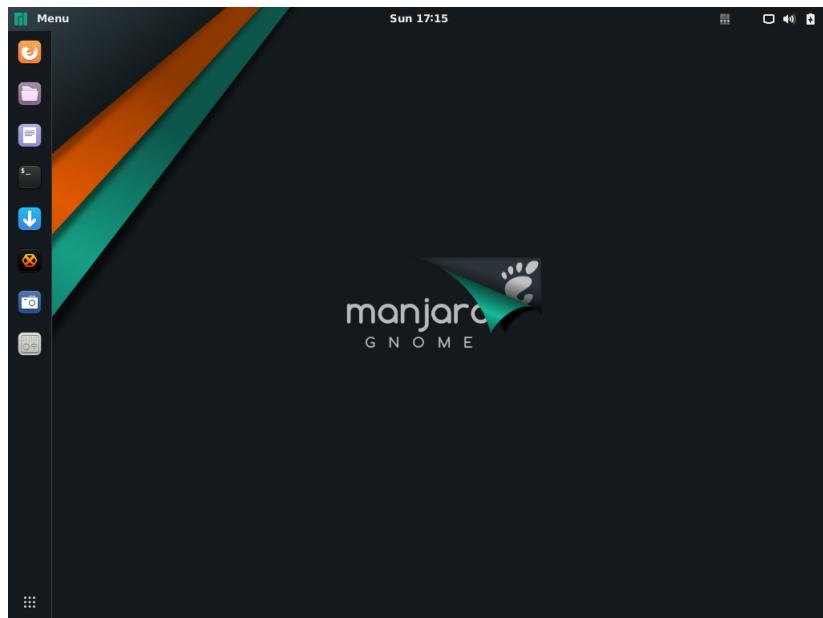


图 1: Gnome-Edition

*fengzhenhua@outlook.com

¹2020 年 11 月 18 日在 Dell-latitude3510 上实际测试了 Manjaro 系统，其软件基本最新，对硬件的支持相当完美，因此决定在 ThinkPadT490 上安装。由于时间紧张，故在 Manjaro 使用手册中直接截取了部分英文，同时结合之前的 Debian(buster) 完成这份配置手册。

目录

1 Download and install	5
1.1 Writing to a USB flash drive in Linux	5
1.1.1 Using the GUI	5
1.1.2 Using the CLI	6
1.1.3 Writing to a USB flash drive in Microsoft Windows	6
1.2 安装 Manjaro	7
1.2.1 Booting in UEFI mode	7
1.2.2 The boot menu	7
1.2.3 Manual installation on a UEFI system	8
1.2.4 Creating partitions from scratch	8
1.2.5 Create a partition table	8
2 软件管理工具 Pacman	13
2.1 Using Pacman	13
2.1.1 切换国内源	13
2.1.2 Synchronising with the Manjaro repositories	14
2.1.3 Updating software	14
2.1.4 Searching for software	14
2.1.5 Installing software	14
2.1.6 Removing software	14
2.1.7 Other Pacman commands	15
2.2 Updating the mirror list	15
3 基础配置	16
3.1 设置 root 权限	16
3.2 配置备份	16
3.3 gnome3 恢复初始设置	16
3.4 必备工具	16
3.5 配置输入法 fcitx	17
3.5.1 fcitx4 的配置	17
3.5.2 fcitx5 配置	17
3.6 配置 gnome3	18
3.6.1 自定义 Terminal 快捷键	18
3.6.2 安装 gnome 拓展	18
3.7 配置 zsh	18
3.8 安装额外字体	18
4 固件升级	18
5 办公工具	19
5.1 邮件客户端	19
5.2 多媒体	19
5.3 电子词典	19
5.4 电子书工具	20

5.5	LaTeX	20
5.5.1	安装 Texlive2021	20
5.5.2	vim 插件 vundle 配置	22
5.5.3	fcitx.vim 配置	23
5.6	TeXmacs	23
5.7	PDFTK	23
5.7.1	PDF 文件加 outline	23
5.7.2	PDF 水印处理	24
5.8	FTP	24
5.8.1	问题解决	27
5.8.2	什么是 xinetd 模式和 initd 模式	27
5.8.3	xinetd 运行模式	27
5.8.4	standalone 模式	27
5.8.5	xinetd 模式和 standalone 模式的区别	27
5.9	搭建个人博客	28
6	Shell 命令	28
6.1	批量创建文件	28
6.2	批量重命名	28
6.2.1	rename 重命名文档	28
6.2.2	借助 for 循环, 利用 mv 重命名	28
6.2.3	linux 获取文件名的原理	29
6.3	linux shell 循环: for、while、until 用法详解	29
6.4	循环中的步进	29
6.4.1	显示 1—10	29
6.4.2	1—10 递加	29
6.4.3	1—10 递减	29
6.4.4	随机生成 16 位包含数字、大小写字母、符号的密码	29
6.4.5	tr -d 和 tr -dc 的用法	29
6.5	for 循环常见语法格式	30
6.5.1	for 循环的一般用法	30
6.5.2	for 循环的特殊用法	30
6.6	while 循环	30
6.6.1	while 循环的特殊用法	31
6.7	until 循环	31
6.8	循环控制语句 continue	31
6.9	循环控制语句 break	31
6.10	循环控制语句 shift	31
6.11	select 循环	31
6.12	信号捕捉 trap	32
7	可选软件	32
7.1	安装 thinkfan	32
7.1.1	thinkfan 源码编译	32
7.1.2	thinkfan 配置	33

7.2 安装 office2007	34
7.3 安装 wps 11.1.0.10702	35
7.3.1 安装 yay 和 debtap	35
7.3.2 debtap 升级	36
7.3.3 安装 wps-office	36

1 Download and install

登录 Manjaro 官网<https://manjaro.org/get-manjaro/> 图2是下载页面截屏，此有两个版本，一个为预装软件的版本，拿来即用，十分方便；一个是 Minimal 版本，后者没有预装过多的软件。作为日常办公用，我选择了后者，用到一个功能则安装一个。

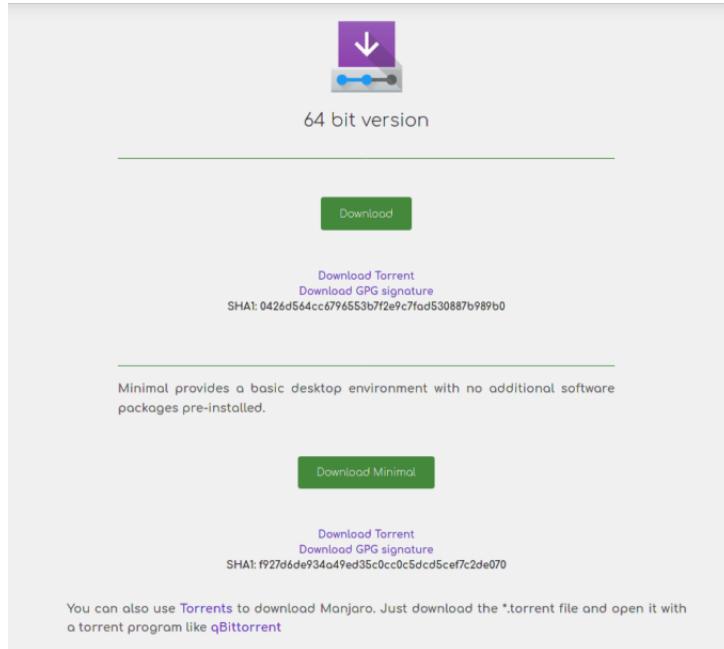


图 2: manjaro-gnome 的两个版本

1.1 Writing to a USB flash drive in Linux

1.1.1 Using the GUI

To take advantage of the faster access times and much quieter nature of a USB flash drive, there are a number of methods of writing the disc image file. One of the easiest is to use SUSE Studio Image- writer3.



图 3: SUSE Studio Imagewriter https://en.opensuse.org/SDB:Live_USB_stick

SUSE Studio Imagewriter should be available for installation from your distribution's Software Center / Software Manager / Package Manager / repositories. Once SUSE Studio Imagewriter has been downloaded and installed, ensure that your USB flash drive is plugged in before starting the application. The process of using SUSE Studio Imagewriter is very straightforward:

1. Click on the centre icon.
2. Navigate to where you downloaded the disc image file, and select it.
3. Ensure that your USB flash drive has been selected from the drop-down menu.
4. Click on the Write button.
5. Once it has finished, reboot your system, making sure to select the USB flash drive as the boot device in the BIOS menu (or its UEFI equivalent).

1.1.2 Using the CLI

If you don't want to install extra software, and you are comfortable using the command line, there's an even easier way of writing a disc image to a USB device.

1. Connect the USB flash drive to your PC, open a terminal and cd to where you downloaded the disc image.
2. Check the device identifier with: sudo fdisk -l.
3. Write the image with: sudo dd if=manjaro-gnome-18.0-stable-x86_64.iso of=/dev/(Device identifier from above)

My USB drive is at /dev/sdc so I would use:

```
sudo dd if=manjaro-gnome-18.0-stable-x86_64.iso of=/dev/sdc
```

To view how the writing process progresses, use the option status=progress: sudo dd if=manjaro-gnome-18.0-stable-x86_64.iso of=/dev/sdc status=progress

4. Once completed reboot your system and boot from the USB drive by selecting it as the boot device in the BIOS menu (or its UEFI equivalent).

1.1.3 Writing to a USB flash drive in Microsoft Windows

For Windows users using USB media, Rufus4 is highly recommended.

Select the USB key to be used in the Device menu. Then, on the line beginning with Boot selection, click on Select to select your downloaded disc image, and then Start. After doing so, select DD Image in the window that appears.

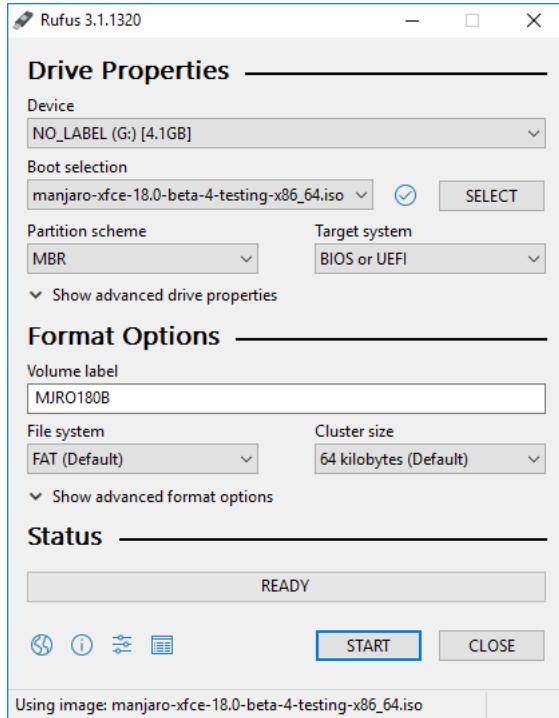


图 4: Rufus with recommended settings

Another recommended piece of software is Etcher⁵. It is incredibly simple to use. All you have to do is select the disc image, the USB drive, and start flashing the image!



图 5: Etcher with recommended settings

1.2 安裝 Manjaro

1.2.1 Booting in UEFI mode

The most important step is to ensure that your machine does not have Secure Boot enabled. Only a very limited number of operating systems will boot with this enabled. You must disable Secure Boot in your UEFI/BIOS before proceeding; if you don't know how to do this please refer to your computer's user manual, as it differs for all computers.

1.2.2 The boot menu

When you boot from your installation media (CD/DVD or USB flash drive) you should be presented with the Manjaro boot screen.⁶ This screen provides several options to help get the best experience from the Live environment.



图 6: Boot menu in UEFI mode

It is possible at this stage to set your preferred language and key- board layout before using Manjaro. This means that you will be able to use and install Manjaro in your native language straight away.

1.2.3 Manual installation on a UEFI system

Note :The windows decorations displayed in the following chapter may differ from the ones currently used. It does not affect the methods shown.

The manual installation option is the one that changes the most if you are using a UEFI system compared to one with BIOS. The fol- lowing steps will walk you through the process of creating a partition scheme suitable for your system in three different scenarios. The first one (Creating partitions from scratch) will show you how to create your partition scheme from within Calamares without using existing partitions. The second one (Using GParted to create partitions be- forehand) will show you how to create the partitions with GParted and then put them to use in the installer. Finally, the third one (Using an already existing EFI partition) will show you how to use a EFI partition previously created and used by another operating system.

1.2.4 Creating partitions from scratch

This scenario will assume that you do not wish to use a single exist- ing partition. We will thus start with a completely empty hard drive. This will also be the case if you are using a brand new drive or are using a virtual machine.

1.2.5 Create a partition table

- Once you select the Manual partitioning option and click on Next, the first thing to make sure is that the partition table is of type GPT. To create a new partition table, click on the New Partition Table button and select GUID Partition Table (GPT). For a refresher on partitions and partition tables, see chapter Some useful definitions.

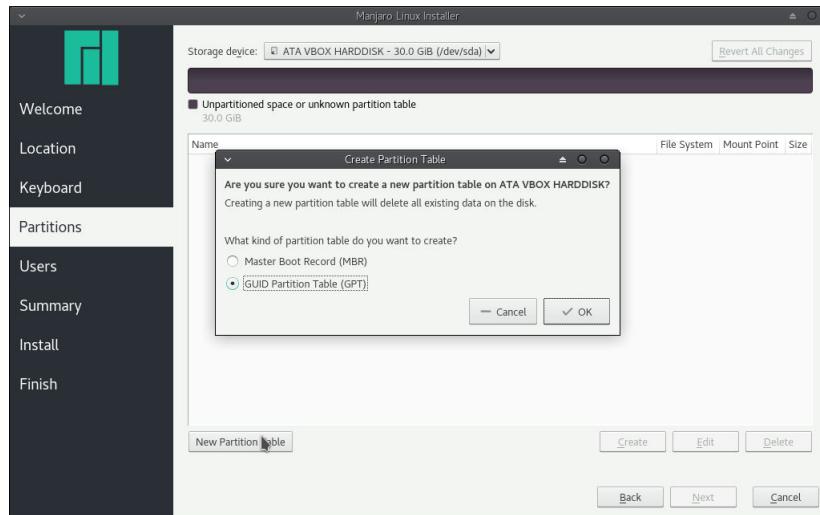


图 7: Create a partition table

2. Then, select the empty space and click on Create to make a new partition. The first one we will create will be the EFI partition, which will house some files used when booting.

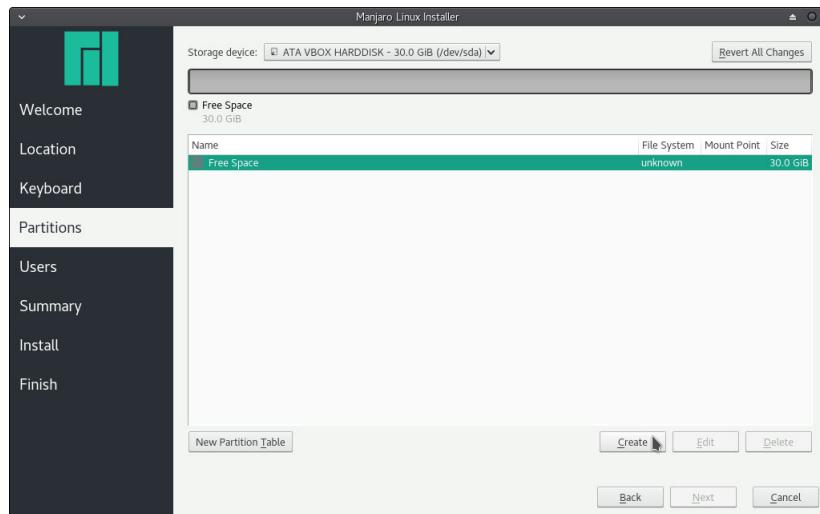
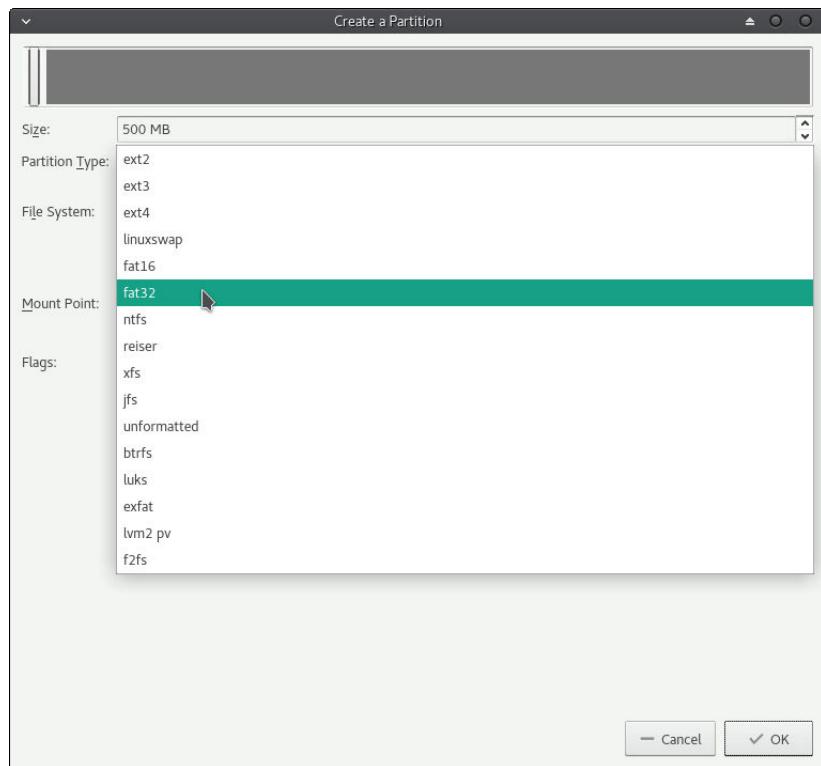
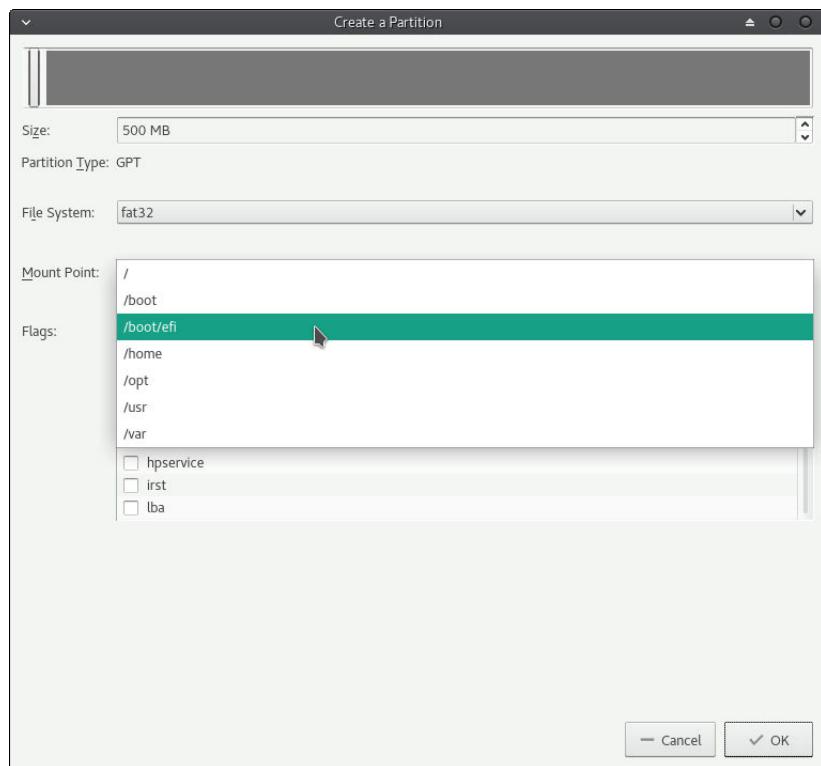


图 8: Create the EFI partition

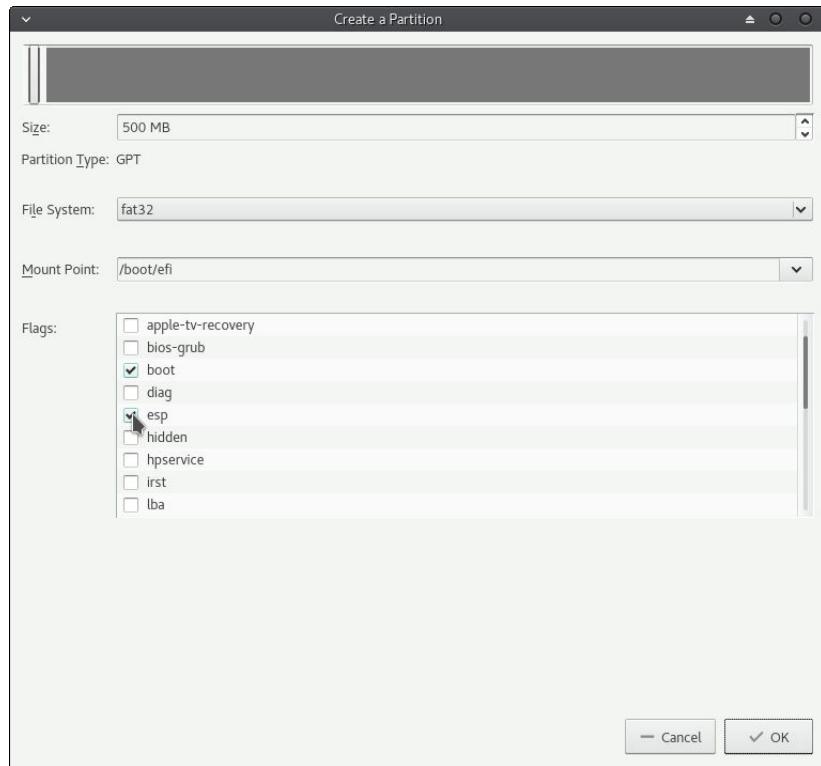
3. In the window that appears, select a size of a few hundred MB. Here I chose 500 MB. Then, select fat32 as the file system. For a refresher on file systems, see chapter Some useful definitions.



4. Next, choose /boot/efi as the mount point. The contents of the partition will be accessible from this directory once Man-jaro is installed. Be careful not to accidentally choose /boot!



5. Finally, select the boot and esp flags from the list. Once this is done, click the OK button.



6. The next partition we will create is the swap. This partition will be used if you ever run out of RAM to complete operations. The suggested size for this partition is a few GB. Here I chose 4 GB. Choose linuxswap as the file system.

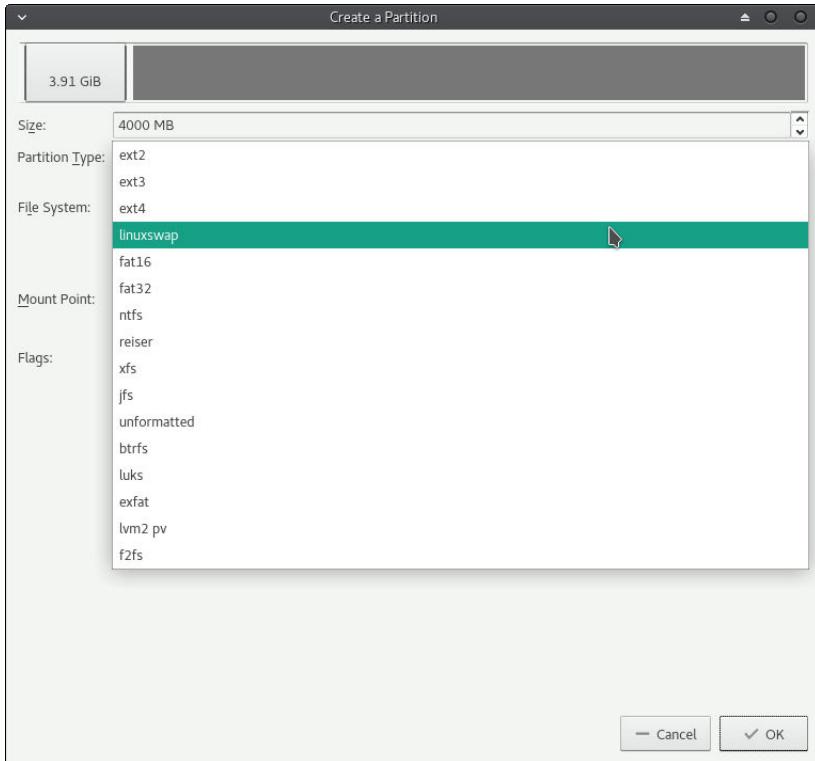


图 9: Create the swap partition

7. You can also click and drag the partition to move it elsewhere in the unallocated space.

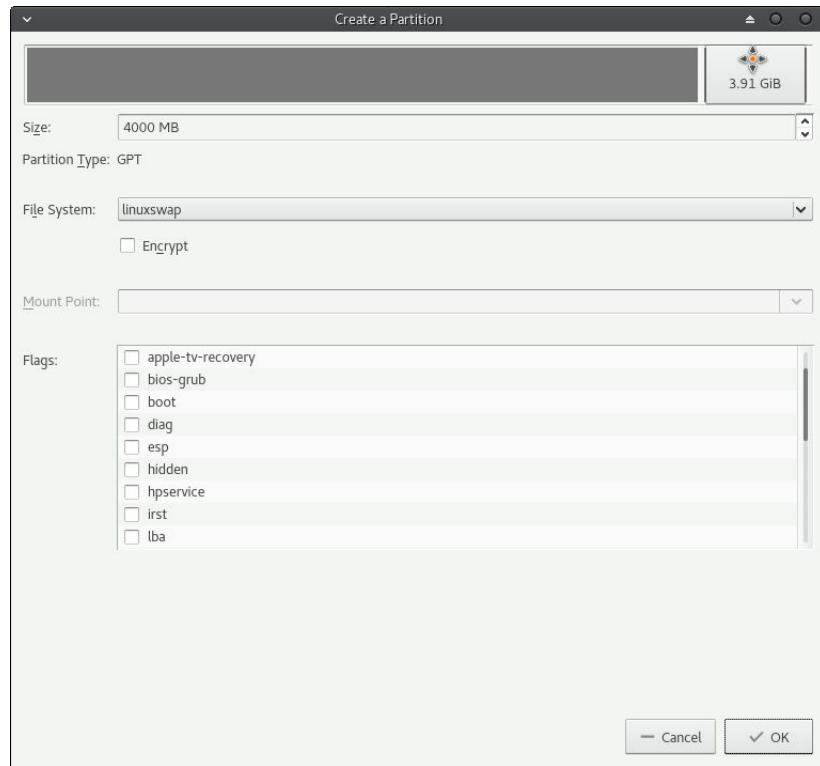


图 10: Create the swap partition

8. Next are the partitions used by Manjaro itself. There are a variety of possible scenarios. Here I chose the simplest case where I only created a single partition mounted under the / directory. Another scenario with the /home directory being housed on another partition is presented in chapter Manual installation on a BIOS system.

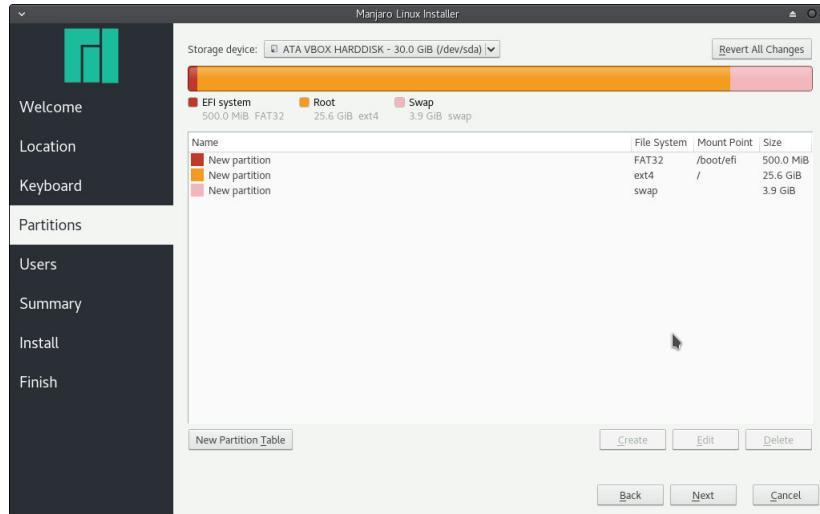


图 11: Create other partitions

9. You can then proceed as usual up to the summary, which is the last step before the changes are applied to your drive.

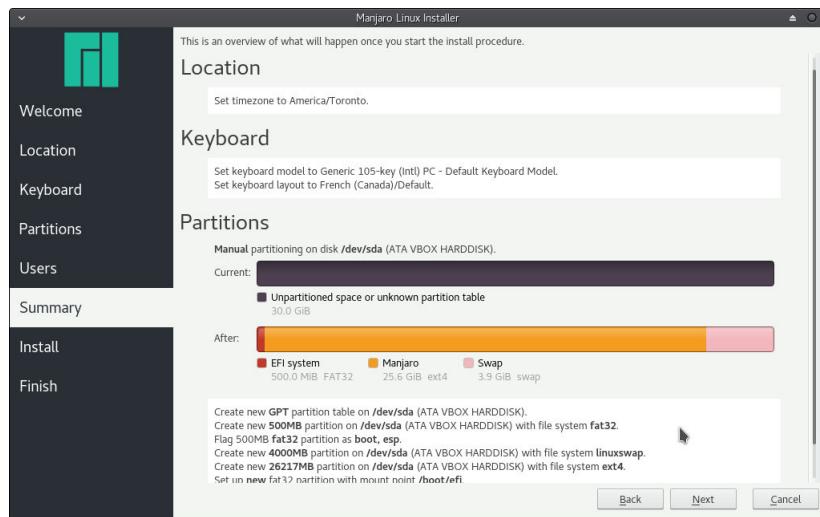


图 12: Look back and confirm

完成上述分区设置后，点击 Next，系统进入安装过程，其间需要设置一些时区等的设置。这里要注意：在安装过程中我遇到了卡在 93% 进度条不动的情况，这是因为联网下载相关软件的问题，同时连接国外网络很慢，于是断网安装可以完美解决问题。

2 软件管理工具 Pacman

2.1 Using Pacman

Pacman is the Arch Linux package manager. This underpins Pamac and Octopi but doesn't have a graphical interface. Instead, it is used by typing commands on the terminal. The commands allow you to install, upgrade, configure, and remove software.

2.1.1 切换国内源

只需一条命令：

```
sudo pacman-mirrors -c China
sudo pacman-mirrors -i -c China -m rank
```

第一条命令非常方便，但是第二条命令可以让源按下载速度排序，然后自己确认使用哪一个。我个人觉得使用第一条命令换成国内的镜像即可。原有的被保存在

`/var/lib/pacman-mirrors/custom-mirrors.json`

现在的被保存在

`/etc/pacman.d/mirrorlist`

另外 pacman 的下载缓存在

`/var/cache/pacman`

对于始终无法信任的情况，选择 Optional TrustAll 即可，所有 res 都用这个策略。

2.1.2 Synchronising with the Manjaro repositories

As new packages are added to the repositories you will need to regularly synchronise the package lists. This is normally taken care of automatically on a regular basis by the software managers, but to perform this manually type the following into the terminal:

```
sudo pacman -Sy
```

This will only download the package lists if there has been a change. Occasionally you may want to force the package lists to be downloaded. To do this, type:

```
sudo pacman -Syy
```

2.1.3 Updating software

Pacman will allow you to perform an update of software already installed with the command:

```
sudo pacman -Su
```

It's a good idea to check whether the package lists are up-to-date at the same time. To do so, type:

```
sudo pacman -Syyu
```

You can also force a package list synchronisation before performing an update with:

```
sudo pacman -Syyuu
```

2.1.4 Searching for software

Finding a package by name is really straightforward. For example, to search for a text editor called Leafpad, type:

```
sudo pacman -Ss leafpad
```

2.1.5 Installing software

Installing a package is equally straightforward. For example, to install Leafpad, type:

```
sudo pacman -S leafpad
```

2.1.6 Removing software

To remove a software package you need to know its name, but the command is straightforward. To remove Leafpad, type:

```
sudo pacman -R leafpad
```

It's also possible to remove a package and any dependencies it required when it was installed. Assuming those other packages are not being used by another piece of software, they will become orphan packages. These serve no function other than taking up space! To remove a software package with its dependencies, type:

```
sudo pacman -Rs leafpad
```

A package may also create configuration files. Normally these are left in place so your configuration is not lost, and can be reused if you install the package again. However, if you want to remove these configuration files too, the command is:

```
sudo pacman -Rns leafpad
```

If at a later date you want to remove all orphan packages and configuration files for packages that you removed some time ago, the following command will do it:

```
sudo pacman -Rns $(pacman -Qtdq)
```

清除已下载的安装包

```
# pacman -Scc
```

查看日志文件

```
# du -t 100M /var
```

or

```
# journalctl --disk-usage
```

删除指定大小的日志文件

```
# journalctl --vacuum-size=50M
```

Be warned, though: this is an advanced command!

2.1.7 Other Pacman commands

It's a good idea to become familiar with the Pacman commands. It can be a very useful tool in case the other software managers refuse to complete an installation, for example if an installation process is interrupted. The Pacman manpage is very informational and can be viewed with the command:

```
man pacman
```

To exit, press q.

2.2 Updating the mirror list

Manjaro Linux packages are hosted on a number of servers across the world; these servers mirror the official Manjaro software repository. When you first install Manjaro it will try to work out which server is closest to you so that software downloads complete as quickly as possible.

Sometimes, though, the list of mirrors can get old. New mirrors can become available, and some mirrors are removed. Updates to Manjaro will regularly trigger an update of the mirror list, but sometimes it's useful to do this manually.

To update the mirror list use the following command:

```
sudo pacman-mirrors -f o
```

This will test the speed of all currently available mirrors and set your machine to use the best for you. Once this command has completed, you should force a download of the package lists with the command:

```
sudo pacman -Syuu
```

If you would like more control over which mirrors to use, run the command:

```
sudo pacman-mirrors -i
```

This will allow you to select which mirror, or mirrors, to try when checking for software updates and downloading new packages. It's normally best to choose mirrors that are geographically close to you, so if you're in Canada then servers from Canada and the US are generally a good bet. Of course, you're free to choose whichever you want!

To reset back to an automatic choice, run this command:

```
sudo pacman-mirrors -f
```

Remember to force a download of the package lists to use the new mirrors!

3 基础配置

3.1 设置 root 权限

在 2021 年 01 月 03 日升级 manjaro 后发现 fcitx5 不与很好的工作，同时系统也不能快速的进入和退出，我意识到可能升级系统导致了问题。于是决定下载最新的安装镜像，然后记录 U 盘后重新安装了一遍，但是新安装成的系统的安装过程中做了简化，它没有设置 root 密码一项，所以在安装成功后，需要首先取得 root 权限才能进行系统的维护。这里记录命令如下

```
$ sudo passwd root
```

然后输入两次密码就可以了。

3.2 配置备份

如果是新电脑或从 windows 环境下全面转入 linux 工作的用户不必进行这一步设置。之前安装过 linux 的用户，由于其/home/yourname/ 中存在一些你常用软件的配置文件，所以在安装新系统时，需要重置原来的设置。删除主目录的配置文件，由于 vim 的重要性，此处注意先备份一下 vim 配置文件.vimrc

```
# rm -rf /home/thinkpad/.[a-z]*
```

3.3 gnome3 恢复初始设置

使用 gnome3 时间久了有可能会因为设置导致这样那样的问题，所以在这里我百度到了重置 gnome3 的方法。

```
$ dconf dump / > my_settings 备份配置  
$ dconf reset -f / 重置桌面  
$ dconf load / < my_settings 还原配置
```

3.4 必备工具

Manjaro 已经足够好用，但是在测试完成之后，用在工作当中有些软件还是和预装的有所差别的，这里我选择了最小化的安装镜像，所以这里需要安装一些我日常工作必备的工具。

```
# pacman -S vim unarchiver pdftk cups-pdf poppler-data
```

1. vim :linux 下的编辑器之王。

2. unarchiver : manjaro 中的解压缩软件，提供命令 unar。
3. pdftk: pdf 文件的强大处理工具。
4. cups-pdf : pdf 虚拟打印机。
5. poppler-data : evince ,okular ,epdfview 的 pdf 功能均由 xpdf 的分支 poppler 提供，如果不安装则有部分的 pdf 文件在用 evince 打开时不显示汉字。

3.5 配置输入法 fcitx

目前 fcitx 一共有两个版本可用，一个是老版的 fcitx4，一个是新版的 fcitx5，分别记录其安装方法和配置如下。

3.5.1 fcitx4 的配置

此为 fcitx4 的基础配置，安装即可，没有多余的配置。

```
# pacman -S fcitx fcitx-table fcitx-table-wbpy
```

1. fcitx : 输入法基础框架主程序
2. fcitx-table: 输入法面板
3. fcitx-table-wbpy : 五笔输入法

3.5.2 fcitx5 配置

```
# pacman -S fcitx5 fcitx5-qt fcitx5-gtk fcitx5-configtool fcitx5-chinese-addons
fcitx5-material-color
```

1. fcitx5: 输入法基础框架主程序
2. fcitx5-qt :QT5 程序的支持
3. fcitx5-gtk:GTK 程序的支持
4. fcitx-chinese-addons : 简体中文输入的支持，云拼音
5. fcitx5-material-color: 一款使用 Material Design 色的 fcitx5 皮肤，旨在模仿 windows10 自带输入法的 UI

对于 xorg 显示服务器，编辑文件/etc/profile 添加或修改为

```
export GTK_IM_MODULE=fcitx5
export QT_IM_MODULE=fcitx5
export XMODIFIERS="@im=fcitx5"
fcitx5 &
```

对于 wayland 显示服务器，编辑文件 /.pam_environment 添加

```
GTK_IM_MODULE DEFAULT=fcitx5
QT_IM_MODULE DEFAULT=fcitx5
XMODIFIERS DEFAULT=@im=fcitx5
```

由于 xorg 和 wayland 读取环境变量时所需要的文件和格式不同，所以在设置 fcitx5 时要特别注意以上设置。

3.6 配置 gnome3

3.6.1 自定义 Terminal 快捷键

设置-> 键盘-> 自定义快捷键-> 添加

名称:Terminal

命令:gnome-terminal

设置:Ctrl+Alt+T

3.6.2 安装 gnome 拓展

为了方便 gnome3 的使用，安装以下拓展

```
Desktop Icons ,Dash to Dock, Hide Top Bar, Suspend Button
```

1. Desktop Icons : 桌面显示图标拓展
2. Dash to Dock : Dock 拓展
3. Hide Top Bar : 系统工具栏自动隐藏拓展
4. Suspend Button : 挂起按钮

在 Manjaro 中上述插件只有 Hide Top Bar 没有安装，这个需要自己安装。

3.7 配置 zsh

当更新系统后我发现新安装的系统和滚动升级后的系统，zsh 主题并未跟着更新，2021 年 3 月 7 日得到解决方法。就是重新安装一下配置文件，即执行

```
# pacman -S manjaro-zsh-config
# cp /etc/skel/.zshrc ~/.zshrc
```

重新进入 shell 后，得到新的 zsh 主题。

3.8 安装额外字体

安装 windows10 下拷贝来的 fonts 字体，及 founder ,adobe 等到系统的指定目录。

```
# cp ./Fonts/Adobe/* /usr/share/fonts/Adobe/
# cp ./Fonts/windows10/ /usr/share/fonts/windows10/
# cp ./Fonts/founder_GBK/ /usr/share/fonts/Founder/
# fc-cache -vf
```

4 固件升级

使用 **fwupd** 更新电脑固件。先安装此命令

```
# pacman -S fwupd
```

查询更新

```
# fwupdmgr get-updates
```

如果有更新可用，则直接下载安装

```
# fwupdmgr update
```

注意：升级固件是一项非常谨慎的事情，升级过程中一定确保电脑连接电源，期间不能断电，以免造成损坏。

5 办公工具

5.1 邮件客户端

manjaro 非 minimal 版中默认安装的邮件客户端为 geary，但是我的新浪邮箱不再支持这个客户端。于是重新安装了 thunderbird。

```
# pacman -S thunderbird thunderbird-i18n-zh-cn
```

1. thunderbird : 邮件客户端主程序
2. thunderbird-i18n-zh-cn: 中文界面程序包

5.2 多媒体

系统默认安装的是 totem 播放器，这里以更加简洁的 mpv 取代，同时按装 mplayer 支持 goldendict 发音。

```
# pacman -Rns totem
# pacman -S mpv mplayer
```

5.3 电子词典

系统没有内置，此安装功能强大的 goldendict

```
# pacman -S goldendict
```

同时设置如下

1. 关闭全文搜索，以加快查询速度

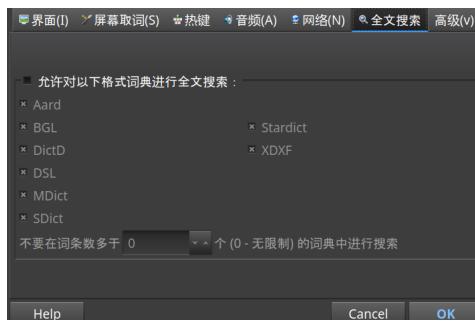


图 13: 关闭全文搜索

2. 使用外部 mplayer 发音

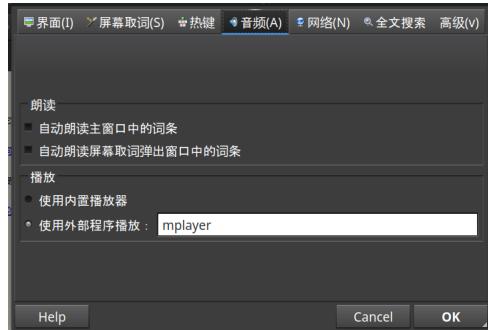


图 14: 外部发音

3. 设置展开可选部分



图 15: 展开可选部分

5.4 电子书工具

系统默认安装了 evince PDF 阅读器，然而我还需要一个 epub 等阅读器，此处安装 clibre

```
# pacman -S clibre
```

5.5 LaTeX

5.5.1 安装 Texlive2021

1. 下载 下载地址:

官方镜像: texlive2021.iso

USTC 镜像: texlive2021.iso

上述下载地址为中国科学技术大学的地址，此外还有若干，请君从如下地址自行选择

<https://ctan.org/mirrors>

2. 挂载 ISO 镜像

```
# mount -o loop texlive2021.iso /mnt/
# cd /mnt
# ./install-tl
```

出现选项后，输入 I 直接安装（也可以更改选项）。不出意外的话，5 分钟应该就 OK 了，然后退出 root 用户。

3. 环境变量

2021 年升级 manjaro 后,发现显示服务器默认为 wayland, 它的响应速度很好。但是对于文件 `/.bashrc` 是不会读取的, 这在 fcitx5 的调用过程中反应出来, 但是使用 texlive 编译文档, 有两种方式, 如果足够熟练则不需要说明文档, 那完全可以用 pacman 安装即可, 但是一般宏包也是会升级的, 如果不看说明文档则有些包在调用时就会出问题。所以还是通过官方 ISO 镜像来安装 texlive 比较好。这样不仅可以使用 tlmgr 和 texdoc , 而且也可以阅读到最新的说明文档, 综合考虑比 pacman 的方式好一些, 应该为上策。

根据 texlive-en.pdf 文件 (texdoc texlive 可以调出) 介绍, 如果在安装过程中选择了建立超链接, 则不必考虑环境变量, 所以在安装时要建立超链接, 这样子可方便一些。texlive-en.pdf 中原文如下:

If you elected to create symlinks in standard directories (described in section 3.2.4), then there is no need to edit environment variables. Otherwise, on Unix systems, the directory of the binaries for your platform must be added to the search path. (On Windows, the installer takes care of this.)

如果没有建立超链接, 则在当前用户的 `/.bashrc` 中加入如下语句:

```
# TeX Live 2021
export MANPATH=${MANPATH}:/usr/local/texlive/2021/texmf-dist/doc/man
export INFOPATH=${INFOPATH}:/usr/local/texlive/2021/texmf-dist/doc/info
export PATH=${PATH}:/usr/local/texlive/2021/bin/x86_64-linux
```

4. 卸载 ISO 镜像

```
# umount /mnt/
```

5. 更新 TeXLive 可以使用如下命令更新 TeXLive 宏包:

```
# 更新 TeXLive 包管理器 tlmgr
# tlmgr update --self
# 更新 TeXLive 的全部包
# tlmgr update --all
```

默认情况下,会自动搜索合适的镜像来更新,也可以使用 `--repository` 选项指定了要使用哪一个 CTAN 镜像。

比如 USTC 镜像:

```
#tlmgr update --self --repository
http://mirrors.ustc.edu.cn/CTAN/systems/texlive/tlnet/
#tlmgr update --all --repository
http://mirrors.ustc.edu.cn/CTAN/systems/texlive/tlnet/
```

比如阿里云镜像:

```
#tlmgr update --self --repository
http://mirrors.aliyun.com/CTAN/systems/texlive/tlnet/
#tlmgr update --all --repository
http://mirrors.aliyun.com/CTAN/systems/texlive/tlnet/
```

为了方便我们也可以直接设置 texlive 源为国内源，命令如下

```
# 清华大学为例
tlmgr option repository https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/tlnet
# 华中科技大学安装源
tlmgr repository set http://mirror.hust.edu.cn/CTAN/systems/texlive/tlnet
```

6. 安装额外的字体 TeXLive 2021 在使用 xeLaTeX 处理中文时，有自己的默认字体。大多数 Linux 发行版下，都使用自带的 Fandol 字体。如果想要使用 Windows 字体，可以将字体文件复制到 /.fonts 目录下，并在 tex 源码中指定字体选项即可。

Xe_TE_X 和 Lu_aT_EX 可以直接使用系统字体。然而 texlive 自带的字体不在系统的字体目录中，所以当使用 Xe_LA_TE_X 编译文件时，如果使用了 texlive 里面带的字体，就会出错，这时需要进行如下配置，让系统可以识别 texlive 所带的字体就可以了。

```
# cp /usr/local/texlive/2021/texmf-var/fonts/conf/texlive-fontconfig.conf
/etc/fonts/conf.d/09-texlive.conf
# fc-cache -fsv
```

5.5.2 vim 插件 vundle 配置

这里借助 vundle 管理 vim 插件，用它安装 vim-latex 插件，来写 PDF 文档。

1. 首先安装 git

```
# pacman -S git
```

2. 下载 Vundle。复制 Vundle 的 Github 仓库地址：

```
$ git clone https://github.com/VundleVim/Vundle.vim.git
~/.vim/bundle/Vundle.vim
```

为方便起见，给出我自己设置好的 bundle 附件，点击另存为到指定路径即可。

附件： [bundle.zip](#)

3. 配置 Vundle 创建 /.vimrc 文件，以通知 Vim 使用新的插件管理器。安装，配置和移除插件需要的这个文件。这里配置文件在 /.vim/bundle/Vundle.vim/README.md 另外也有简体中文本的说明。我使用的英文版的。复制到 /.vimrc 并删除多余的内容，我修改后的.vimrc 文件为：

附件： [.vimrc](#)

在使用 vim-latex 编辑文件时需编译和查看两个步骤。这里需要修改 vim-latex 的配置文件。其位置为

```
~/.vim/bundle/vim-latex/ftplugin/latex-suite/texrc
```

修改此文件以实现编译 \\ 和 \lv 使用 evince 查看生成的 pdf 文件。具体修改内容为

```
if has('macunix')
TexLet g:Tex_DefaultTargetFormat = 'pdf'
```

```

else
  TexLet g:Tex_DefaultTargetFormat = 'pdf'
endif
TexLet g:Tex_Compilerule_pdf = 'xelatex -synctex=1
-interaction=nonstopmode -file-line-error-style $*'
TexLet g:Tex_ViewRule_pdf='evince'

```

注意：在附件 bundle.zip 中已经修改完毕。

5.5.3 fcitx.vim 配置

在使用 vim-latex 写作时，由于经常输入中文，所以在 vim 切换到控制模式后需要将输入法切换到英文再操作，这时可以使用 fcitx.vim 实现自动切换。代码来源为²

<https://github.com/lilydjwg/fcitx.vim/blob/master/so/fcitx.vim>

附件：[fcitx.vim](#)

5.6 TeXmacs

在 linux 中使用 TeXmacs 时，输入中文只是显示字符编码，不显示文字。原因在于 TeXmacs 中没有默认设置好中文字体，所以解决方案为在 TeXmacs 默认字体文件夹中建立一个软链接指向系统字体。然后执行 TeXmacs 的字体刷新即可。如下

```

~:ln /usr/share/fonts/windows10 /home/feng/.TeXmacs/fonts/truetype/windows10
~:texmacs --delete-font-cache

```

再重新启动 TeXmacs 即可输入中文。

5.7 PDFTK

5.7.1 PDF 文件加 outline

从 2019 年 3 月 22 日到 24 日，我一直在研究 PDF 文件。原因在于我手下有大量的 PDF 文件，但是大多数都是扫描版的，它们没有目录，这在阅读的时候很不方便。经过研究后，发现可以使用 pdftk+latex 来处理。

第一步：使用 pdftk 导出文件结构，由于标签中含有中文，所以使用 dump_data_utf8 选项，否则使用 dump_data。

```
$pdftk your.pdf dump_data_utf8 output your.info
```

第二步：如果是英文，则直接编辑文件 your.info 即可。但是目录中含有中文，则研究发现了，在文件 your.info 中，是通过直接引用字符编码来处理中文的，这给直接编辑导致了麻烦。但是可以用 latex 编辑目录，生成 pdf 文件。然后导出 latex.info 文件，用 vim 同时编辑两个 info 文件，your.info 和 latex.info，取其中的 Bookmark 部分，然后再校准页码，导入 your.pdf 就可以了。

```
$pdftk your.pdf update_info your.info output out.pdf
```

这样就可以得到带有 outline 的文件 out.pdf 了。

²在 Manjaro 中我没有使用这个配置，但是依然记录下，以备不时之需。

5.7.2 PDF 水印处理

在使用 latex 编辑文件时可以通过宏包 draftwatermark 来加上水印。同时可以使用命令 SetWatermarkText 来修改水印文字。

```
\usepackage{draftwatermark}
\SetWatermarkText{冯老师手稿}
```

还有一些文件，是通过添加简单的图片水印来实现的。在网络上发现了一个方法来去除水印，但是并不是很普遍。第一步使用 pdftk 解压缩 pdf 文件，其中 pdf 文件结构可以很清楚的识别。但是原来文件中的文字及图片等都是二进制流，所以使用 vim 打开时，要加上选项-b 来告知 vim 这是二进制文件，不加的话就会出错。如果发现在文件中有某个图片在每一页中都会出现，则删除此项目就可以了。对于某些文字水印也成立。

```
$pdftk your.pdf output uncompress.pdf uncompress
$vim -b uncompress.pdf
$pdftk uncompress.pdf output compress.pdf compress
```

有一种情况比较难处理，比如：朗道十卷.pdf，在我处理此文件时没有发现有大量相同的项目被引用，所以可以判断这个“万千合集站”的水印是以图片的方式和原扫描图片叠加到一块的。使用以上步骤后，发现它是经过 freepic2pdf 处理的，这样的话文件所构成的 pdf 文件就不好识别水印。即便是能够识别，由于水印是混在二进制流中的，所以处理起来也是很麻烦的。事实上，像这种情况，可能也就是 photoshop 或 gimp 等软件一页一页的处理，再合成 pdf 文件才有可能去掉水印，使用 vim -b your.pdf 的方式将是一种可能但不可行的方法。由于以考研大业为重，此处不再研究，等今后有时间了再去考虑去水印的问题。

5.8 FTP

搭建 ftp 服务器的目的是为了能够与我的 ipad 顺利传输资料，但是 mac 系统的封闭性决定了这不适合直接用传输线连接传输资料。所以我只能考虑可行的各种方案，最终决定了使用 ftp 服务器来共享文件。³

1. 安装

```
# pacman -S vsftpd
```

2. 配置

```
# cp /etc/vsftpd.conf /etc/vsftpd.conf.bak
# vim /etc/vsftpd.conf
```

附件：[vsftpd.conf](#)

3. 新建 ftp 根目录，并增加权限

```
# mkdir -m 777 /home/feng/myftp
```

4. 重启，停止，启动服务器

```
# systemctl restart vsftpd
# systemctl start vsftpd
# systemctl stop vsftpd
```

³2019 年 3 月 5 日晚正式测试成功。

5. 添加用户名和密码

```
\# useradd -d /home/feng/myftp ipad
\# passwd ipad
输入新的 UNIX 密码:
重新输入新的 UNIX 密码:
passwd: 已成功更新密码
```

6. 新建文件 vsftpd.user_list，其内容就是刚刚添加的用户名

```
# vim /etc/vsftpd.user_list
~ipad
```

7. 新建文件 vsftpd.chroot_list(针对问题 500)，不需要添加任何用户，如果添加，则这个用户可以访问 ftp 根文件之外的文件

```
$ vim /etc/vsftpd.chroot_list
```

8. 针对可能出现的问题（530），修改文件：如果是 Debian 系统，则不用执行这一步是可以，直接跳到第 9 步。

```
# vim /etc/selinux/config
~SELINUX=disable
```

9. 针对 530 问题，删除文件，适合于 Ubuntu/Debian 系统

```
# rm /etc/pam.d/vsftpd
```

因为 Ubuntu/Debian 启用了 PAM，所以用到 vsftpd 时需要用到/etc/pam.d/vsftpd 这个文件（默认源码安装的不会有文件），因此除了匿名用户无法登录。所以解决的办法就是删除这个文件。

10. 由于我是最小化安装的，所以默认没有安装 ftp，则安装 ftp 客户端，这一步不是必须的，为了在本机测试而设置的。

```
# pacman -S ftp
$ ftp 192.168.1.112
Connected to 192.168.1.112
220 (vsFTPd 3.0.3)
Name (192.168.1.112:feng):
```

11. 本机测试

```
~$ ftp 10.80.3.192
Connected to 10.80.3.192.
220 (vsFTPd 3.0.3)
Name (10.80.3.192:ipad):ipad
331 Please specify the passwd.
```

```
Password:  
230 Login successful.  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp> ls  
200 PORT command successful. Consider using PASV.  
150 Here comes the directory listing.  
226 Directory send OK.  
ftp> pwd  
257 ``/home/feng/myftp'' is the current directory  
ftp> bye  
221 Goodbye.
```

12. 附录

FTP 数字代码的意义

- 110 重新启动标记应答。
- 120 服务在多久时间内 ready。
- 125 数据链路埠开启，准备传送。
- 150 文件状态正常，开启数据连接端口。
- 200 命令执行成功。
- 202 命令执行失败。
- 211 系统状态或是系统求助响应。
- 212 文件的状态。
- 214 求助讯息。
- 215 名称系统类型。
- 220 新的联机服务 ready。
- 221 数据连续开启，但无法传输动作。
- 226 关闭数据连接端口，请求的文件操作成功。
- 227 进入 passive mode。
- 230 使用者登入。
- 250 请求的文件操作完成。
- 257 显示目前的路径名称。
- 331 用户名称正确，需要密码。
- 332 登入时需要账号信息。
- 350 请求的操作需要进一步的人命令。
- 421 无法开启数据链路。
- 426 关闭联机，终止传输。
- 450 请求的操作未执行。
- 451 命令终止：有本地的错误。
- 452 未执行命令：磁盘空间不足。
- 500 格式错误，无法识别命令。
- 501 参数语法错误。
- 502 命令执行失败。
- 503 命令顺序错误。
- 504 命令所接的参数不正确。
- 530 示登入。
- 532 储存文件需要账户登入。
- 550 未执行请求的操作。
- 551 请求的命令终止，类型未知。
- 552 请求的文件终止，储存位水溢出。

553 未执行请求的命令，名称不正确。

5.8.1 问题解决

2021 年 03 月 07 日星期日在使用 vsftpd 向 ipad 传输数据时，发现不能正常开启 vsftpd，经百度发现原因。解决方案为：在 /etc/vsftpd.conf 中添加

```
seccomp_sandbox=NO
```

5.8.2 什么是 xinetd 模式和 initd 模式

像其它守护程序一样，vsftpd 提供了 standalone 和 inetd（inetd 或 xinetd）两种运行模式。简单解释一下，standalone 一次性启动，运行期间一直驻留在内存中，优点是对接入信号反应快，缺点是损耗了一定的系统资源，因此经常应用于对实时反应要求较高的专业 FTP 服务器。inetd 恰恰相反，由于只在外部连接发送请求时才调用 FTP 进程，因此不适合应用在同时连接数量较多的系统。此外，inetd 模式不占用系统资源。除了反应速度和占用资源两方面的影响外，vsftpd 还提供了一些额外的高级功能，如 inetd 模式支持 *perIP*(单一 IP) 限制，而 standalone 模式则更有利于 PAM 验证功能的应用。

5.8.3 xinetd 运行模式

大多数较新的系统采用的是 xinetd 超级服务守护进程。使用 “vi /etc/xinetd.d/vsftpd” 看一下它的内容，如下：

```
disable = no
socket_type = stream
wait = no
```

这表示设备是激活的，它正在使用标准的 TCP Sockets。如果 “/etc/vsftpd.conf” 中的有选项为 “listen=YES”，注销它最后，重启 xinetd，命令如下：

```
$ /etc/rc.d/init.d/xinetd restart
```

需要注意的是，“/etc/xinetd.d” 目录中仅能开启一个 FTP 服务。

5.8.4 standalone 模式

standalone 模式便于实现 PAM 验证功能。进入这种模式首先要关闭 xinetd 下的 vsftpd，设置 “disable = yes”，或者注销掉 “/etc/inetd.conf” 中相应的行。然后修改 “/etc/vsftpd.conf” 中的选项为 “listen=YES” 如果是 standlone 模式，那么它是作为单独的一个服务启动的，不需要系统协作，不作为系统服务，如果要是成为 xinetd 模式，那么它的服务就要受系统服务的限制，比如创建一个新的服务进程，但是也有缺点，如果 xinetd 服务本身出了问题，那么相关的服务也是会受到影响的。

5.8.5 xinetd 模式和 standalone 模式的区别

以 xinetd 模式运行的服务表示该服务进程并不以守护进程执行，以 FTPD 进程为例吧，以 xinetd 模式运行这个服务，情况是这样的，本身 FTP 服务是会监听 21 号端口的，但是以这种模式运行这个服务的话，21 号端口则由 xinetd 进程来监听（此时 FTPD 服务并没有运行），如果你的网卡接收到有 21 号端口请求，则有 xinetd 进程会去调用 FTPD 程序，将在 21 号端口接收到的数据移交给 FTPD 进程去处理，处理完后 FTPD 进程退出，而 xinetd 进程继续监听 21 号端口，这有点类似 windows 的 svhost 进程；而以 standalone 模式运行的服务则是服务进程，如 ftpd 以守护进程在内存中运行，接收到 21 号端口的请求后由 FTPD 进程 fork 出一个子进程进行处理，而原进程继续监听 21 号端口。

5.9 搭建个人博客

- 需要软件共有三个 node ,npm ,hexo

<++>

6 Shell 命令

6.1 批量创建文件

以下命令可以创建 9 个文件，分别为： file1 , file2 ,...,file9 。其中对 {1..9} 的解释为：其中的两个点号表示循环，所以可以建立，删除对应的多个有规律的文件。

```
$ touch file{1..9}
```

以下命令可以分别创建文件： file1,file3,file5,file7

```
$ touch file{1,3,5,7}
```

以下命令可以创建 9 个文件夹，分别为： dir1,dir2,⋯ ,dir9 。

```
$ mkdir dir{1..9}
```

以下命令可以分别创建文件夹： dir1,dir3,dir5,dir7

```
$ mkdir dir{1,3,5,7}
```

6.2 批量重命名

6.2.1 rename 重命名文档

以下命令可以将当前文件夹下所有的文件名中的 **file** 替换为 **文档**，同时文件名中的其它部分不变。使用 rename.ul 只能统一命名文件名中的一部分为另一相同的名称。按网上的说法 rename 有两个版本，一个 C 语言版，另一个为 perl 版本，其中 perl 版本支持正则表达式，不巧的是我的 Debian10 中的好象是 C 语言版本，所以它的能力是有限的。

```
$ rename.ul file 文档 /*
```

6.2.2 借助 for 循环，利用 mv 重命名

以下命令为输出预览，由于执行的是大批量的命名工作，谨慎一点还是很有必要的。

```
$ for name in `ls` ; do echo $name ${name%.txt}.tex ; done
```

如果检查无误，则将 **echo** 换为 **mv** 即可

```
$ for name in `ls` ; do mv $name ${name%.txt}.tex ; done
```

在此命名方式中可以去掉名称中的一些附加字符，重点在花括号中的取代部分。

6.2.3 linux 获取文件名的原理

1. linux 获取文件名，截取字符串的方法为: \${}
2. \${var##*/} 该命令的作用是去掉变量 var 从左边算起的最后一个'/' 字符及其左边的内容，返回从左边算起的最后一个'/'（不含该字符）的右边的内容。
3. \${var#*} 该命令的作用是去掉变量 var 从左边算起的第一个'/' 字符及其左边的内容，返回从左边算起第一个'/'（不含该字符）的右边部分的内容。
4. \${var%/*} 该命令的使用是去掉变量 var 从右边算起的第一个'/' 字符及其右边的内容，返回从右边算起的第一个'/'（不含该字符）的左边的内容。
5. \${var%%/*} 该命令的使用是去掉变量 var 从右边算起的最后一个'/' 字符及其右边的内容，返回从右边算起的最后一个'/'（不含该字符）

6.3 linux shell 循环: for、while、until 用法详解

循环语句常见分类有三个：1.for 语句; 2.while 语句; 3.until 语句。循环体包括两部分：1. 循环体进入条件；2. 循环体退出条件。循环次数：1. 事先已知 for 语句; 2. 事先未知 while、until 语句。

6.4 循环中的步进

6.4.1 显示 1—10

```
$ echo {1..10}
1 2 3 4 5 6 7 8 9 10
$ echo $(seq 1 10)
1 2 3 4 5 6 7 8 9 10
```

6.4.2 1—10 递加

```
$echo {1..10..2}
1 3 5 7 9
$echo $(seq 1 2 10)
1 3 5 7 9
```

6.4.3 1—10 递减

```
$echo {10..1..2}
10 8 6 4 2
$echo $(seq 10 -2 1)
10 8 6 4 2
```

6.4.4 随机生成 16 位包含数字、大小写字母、符号的密码

```
$cat /dev/urandom |tr -dc ^\circ-\_!@#$%^&*()_` |head -c 16
```

6.4.5 tr -d 和 tr -dc 的用法

```
$echo 12324345 |tr -d 2 % 不要数字 2
$echo 12324345 |tr -dc 2 % 支取数字 2
```

6.5 for 循环常见语法格式

6.5.1 for 循环的一般用法

```
for 变量名 in 列表; do 循环体 ; done
```

循环执行机制：依次将列表中的元素赋值给变量名，每次赋值后执行一次循环体，直到列表中的元素耗尽循环结束。列表生成方式有：

1. 直接给出列表
2. 整数列表
 - (a) {start..end} 如: {1..10}
 - (b) \$(seq[start[step]] end) 如: ` seq 1 2 100` 表示 1-100 之间的所有奇数
3. 返回列表的命令 \$(COMMAND) 如: ` ls /etc`
4. 使用 glob, 如: *.sh
5. 变量引用: \${@}, \${*}

6.5.2 for 循环的特殊用法

双小括号法，即 ((...)) 格式，也可以用于算术运算。双小括号方法也可心使用 bash shell 实现 C 语言风格的变量操作

```
#i=10
#((i++))
```

用法格式

```
for ((控制变量初始化; 条件判断表达式; 控制变量的修正表达式));do 循环体 ; done
```

6.6 while 循环

while 循环格式为

```
while CONDITION ; do 循环体 ; done
```

CONDITION: 循环条件；进入循环之前，先做一次判断；每一次循环之后会再次做判断；条件为“true”，则执行一次循环；增至条件测试状态为“false”终止循环。因此：CONDITION 一般应该有循环控制变量；而此变量的值会在循环体中不断地被修正。进入条件：CONDITION 为 true；退出条件：CONDITION 为 false

“:” 和 true 的返回值都为真

```
$true
$echo $?
0
$:
$echo $?
0
$false
$echo $?
1
```

6.6.1 while 循环的特殊用法

```
while read line;do 循环体; done < /PATH/FROM/SOMEFILE
```

依次读取 /PATH/FROM/SOMEFILE 文件中的每一行，且将行赋值给变量 line

6.7 until 循环

until 循环格式为

```
until CONDITION ; do 循环体 ; done
```

进入条件：CONDITION 为 false；退出条件：CONDITION 为 true。

6.8 循环控制语句 continue

continue 用于循环体中。

continue[N]: 提前结束第 N 层的本轮循环而直接进入下一轮判断；最内层为第 1 层，格式为

```
while CONDITION1 ; do
```

```
    CMD1
    ...
    if CONDITION2;then
        continue
    fi
    CMDn
```

6.9 循环控制语句 break

用于循环体中

break[N]: 提前结束第 N 层循环，最内层为第 1 层

```
while CONDITION1 ; do
```

```
    CMD1
    ...
    if CONDITION2;then
        break
    fi
    CMDn
    ...
```

6.10 循环控制语句 shift

shift[n] 用于将参量列表 list 左移指定次数，缺省为左移一次。参量列表 list 一旦被移动，最左端的那个参量就从列表中删除。while 循环遍历位置参量列表，常用到 shift

```
./doit.sh a b c d e f g h
./shift.sh a b c d e f g h
```

6.11 select 循环

select 循环主要用于创建菜单，按数字顺序排列的菜单项将显示在标准的位置上，并显示 PS3 提示符，等待用户输入。用户输入菜单列表中的某个数字，执行相当的命令，用户输入被保存在内置变量 REPLY 中。select 是个无限循环，因此要记得用 break 命令退出，或用 exit 命令终止脚本。也可以按 ctrl+c 退出循环。select 经常和 case 联合使用，与 for 循环类似，可以省略 in list，此时使用位置参量。

6.12 信号捕捉 trap

1. trap ‘触发指令’信号：自定义进程收到系统发出的指定信号后，将执行触发指令，而不会执行原操作
2. trap “信号：忽略信号的操作”
3. trap ‘-’ 信号：恢复信号的操作
4. trap -p : 列出自定义信号操作

7 可选软件

这部分记录一些可选择的软件，比如说 thinkfan 可以自主控制风扇的运行，但是这个手动控制我觉得不如自动控制来的好，同时 Manjaro 对于 T490 的支持相当好，所以最终我实现了编译源码安装成功了 thinkfan，但是还是选择放弃它，而使用 Manjaro 自动设置。此处仅记录这几天的辛苦学习的结果，以方便日后的使用。而 office2007 使用 playonlinux 运行，这个 playonlinux 下载一些组件时非常慢，同时我自己的 E47A 已经安装了 windows10 可以处理这些 msoffice 文档，故此也把它列为了可选择安装的软件。

7.1 安装 thinkfan

7.1.1 thinkfan 源码编译

1. 下载 <https://github.com/vmatare/thinkfan>

Code		
master	10 branches	11 tags
@v	vmatare	update version string
a5fae88	on 14 Oct	375 commits
.github/workflows	CI: use plain cmake instead of fancy action	8 months ago
examples	example config: fix typo	4 months ago
include/nvidia/gdk	include nvml.h, fix cmake docstring	5 years ago
rcscripts	thinkfan-wakeup.service: fix dependencies	4 months ago
src	log watchdog pings	last month
CMakeLists.txt	update version string	last month
COPYING	remove duplicate src subdir	9 years ago
README.md	README: add kernel doc links	7 months ago

图 16: 下载 thinkfan 源码

2. 附件：[thinkfan-1.2.1.tar.gz](#)

3. 安装编译工具

```
# pacman -S gcc make pkgconf cmake libyaml-cpp
```

4. 源码编译

- (a) In the thinkfan main directory ,do

```
$ mkdir build & cd build
```

(b) Set your build options from the command line. E.g. to configure a build with debugging support:

```
$ cmake -D CMAKE_BUILD_TYPE:STRING=Debug ..
```

5. To compile simply run:

```
$ make
```

6. If you did not change CMAKE_INSTALL_PREFIX ,thinkfan will be installed under /usr/local by doing:

```
# make install
```

按上述步骤可以编译完成 thinkfan ,还有一些细节没有完全记录,请参考源码包内文件 **README.md**。在编译过程中使用到了 **cmake** 工具,这个可能参考其官方网址 <https://cmake.org> 或百度百科 <https://baike.baidu.com/item/cmake/7138032?fr=aladdin>。同时 thinkfan 的配置中用到了 **yaml** 语言,这可以参考其官方网址 <https://yaml.org> 或者百度百科 <https://baike.baidu.com/item/YAML/1067697?fr=aladdin>。

7.1.2 thinkfan 配置

此 thinkfan 是我在 debian10 下安装使用的,由于 ThinkPadT490 硬件过新,debian10 不能很好的支持它,尤其是显卡问题。这部分是经过多方折腾后的结果,可以使用但不是最佳,这样使用了半年多。然尔在我试用 manjaro 后,发现这个 archlinux 系的系统居然各项驱动都完美安装,且电脑直接能完美运行。但是也有尚未解决的问题,其一为 thinkfan 不能实现方便的一键安装,其二为指纹识别仍然无法识别出来。鉴于这二者影响都不大,同时 manjaro 也是目前我的 ThinkPadT490 工作状态最好的系统,所以决定暂时放弃这两个软件的安装。盼望以后能够实现一键安装吧。

ThinkPad 的风扇是可以设置人工控制的,这需要借助 thinkfan ,所以先安装软件

```
# pacman -S thinkfan lm-sensors
```

配置内核模块: 在 /usr/lib/modprobe.d/thinkpad_acpi.conf 中加入

```
echo ''options thinkpad_acpi fan_control=1'' >
/etc/lib/modprobe.d/thinkpad_acpi.conf
```

这样风扇控制就设置成自动启动了。现在加载内核模块

```
modprobe thinkpad_acpi
```

接下来检查传感器

```
# sensors-detect
```

之后的选项都按 y 。执行完毕,再执行

```
find /sys/devices -type f -name ''temp*_input''
```

这样会得到一系列温度传感器的结果，按网上的教程都比较老旧了，这里我自己研究了 thinkfan 的配置文件，其目前使用 yaml 语言写成。由于不同的传感器开机加载的顺序不同，所以按照传统的方法来设置是不同的。研究后发现，使用相对路径的写法不仅可以获得全部温度传感器的结果，还能根据自己的需要关闭一些温度传感器的监控。这个配置文件不是一种写法，此处参考文件 /thinkfan-1.2.1/examples/thinkfan.yaml 附件为：

附件： [example.yaml](#)

此处我的电脑 ThinkPadT490 的具体设置如下（温度区间不一定合理，请君自己修改为认可的值）

附件： [thinkfan.yaml](#)

在 thinkfan.yaml 文件中，最下方的小括号中的设置是风扇的等级设置。第一列是风扇等级，第二列是风扇工作于此等级的最低温度，第三列是最高温度。启用服务以便在每次启动和删除特权时自动启动该服务。设置完毕，检测是否 thinkfan 正确运行，在终端下执行

```
thinkfan -n
```

这时如果正确运行，则会输出一系统温度值及 thinkfan 所处于的风扇等级。

下面设置系统自动加载 thinkfan：执行第一条命令，启用服务以便在每次启动和删除特权时自动启动服务，重新启动电脑，执行第二条命令，检查服务

```
# systemctl enable thinkfan
# systemctl status thinkfan.service

    thinkfan.service - simple and lightweight fan control program
   Loaded: loaded (/lib/systemd/system/thinkfan.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2020-02-27 16:03:45 CST; 1h 28min ago
     Process: 622 ExecStart=/usr/sbin/thinkfan $DAEMON_ARGS (code=exited, status=0/SUCCESS)
    Main PID: 724 (thinkfan)
      Tasks: 1 (limit: 4915)
     Memory: 876.0K
        CPU: 0.000 CPU(s) since start
       CGroup: /system.slice/thinkfan.service
           724 /usr/sbin/thinkfan -q -b 1 -s 15
```

上述结果表明 thinkfan 正常工作了。

7.2 安装 office2007

2021 年 12 月 7 日测试完 wps-office 11.1.0.10702 发现其完全可以应对日常工作了，对于简单的打开编辑工作完全可胜任。但是如果在 linux 下使用 vba 功能，则 wps-office 不能胜任，则此时才建议安装 office2007。安装 playonlinux，由于它是 wine 的前端，所以系统会自动安装 wine 软件，同时由于此软件需要 32 位支持，所以在 linux 上相应的安装上 32 位支持。

```
# pacman -S playonlinux wine32
# pacman -S samba
```

上述第二行的命令是在安装完成 playonlinux 之后，运行安装 office 程序后 wine 崩溃，这个原因在我使用终端运动 wine 时找到了答案，就是第二行命令。由于在 Manjaro 下找不到 winbind，所以百度之后发现 winbind 是 samba 的一个组件，于是安装 samba 后问题解决。同时也可在安装 playonlinux 之前最好还是安装好 wine，这样可以确保安装完成 playonlinux 后可以直接使用。

如果没有安装上述 winbind 的话，office 有可能不能正常工作，但是我实际上没有安装这个软件，系统也正常工作了。由于害怕以后有不可知的问题发生，所以来又装上了这个软件。下载 office2007.iso 然后挂

载镜像再使用 playonlinux 安装即可。在安装时由于 office 自带了一个微软输入法，所以在装 office 时进行自定义安装，在 office 共享功能里，把输入法去掉即可。我在安装时没有去掉，所以在使用 office2007 word 时，发现切换输入法出现问题，百度之后发现可以打开文件-> 选项 -> 高级 -> 输入法控制状态，去掉那个勾就能够正常切换输入法了。

在安装完成 office2007 后发现，即使在我的新电脑下它启动的也不是很快。于是百度了这个问题，得到的答案是 wine 在启动时会请求 gbk-0 字体。一种有效的解决方法是修改

```
/user/share/X11/local/zh_CN.UTF8/XLC_LOCAL
```

文件，将所有的 GBK 字体都注释掉，同时将 GB13000 和 GB2312 也注释掉就可以了。但是我将此文件改名之后发现也没有影响，所以写下此段文字记录一下。

在这一部分记录字体设置问题，安装成功 office2007 后，总感觉字体界面和 windows 下差距太大，所以此处经百度又增加一个设置字体的部分。第一步是将 windows 字体 simsun.ttf simfang.ttf 复制到

```
~/PlayOnLinux's virtual drives/Office2007/drive_c/windows/Fonts
```

第二步，修改

```
~/PlayOnLinux's virtual drives/Office2007/system.reg
```

搜索：LogPixels 找到的行应当是

```
[System\CurrentControlSet\HardwareProfiles\Current\Software\Fonts]
```

将其中的

```
''LogPixels''=dword:00000060
```

改为：

```
''LogPixels''=dword:00000070
```

搜索：FontSubstitutes 找到的行应为

```
[Software\Microsoft\WindowsNT\CurrentVersion\FontSubstitutes]
```

将这一部分的等号右侧的值统一改为 simsun，之后保存。最后，修改

```
~/PlayOnLinux's virtual drives/Office2007/drive_c/windows/win.ini
```

在文件末尾加入：

```
[Desktop]
menufontsize=13
messagefontsize=13
statusfontsize=13
IconTitleSize=13
```

7.3 安装 wps 11.1.0.10702

7.3.1 安装 yay 和 debtap

```
# pacman -S yay
$ yay -S debtap
```

在使用 Manjaro 时，有时候找不到对应的软件包，所以这时候可以借助 yay 转换 debian 的软件包来实现软件安装。

7.3.2 debtap 升级

1. 替换 debtap 同步仓库为国内的源

```
# vim /usr/bin/debtap
```

替换 `http://ftp.debian.org/debian/dists` 为 `https://mirrors.ustc.edu.cn/debian/dists`

替换 `http://archive.ubuntu.com/ubuntu/dists` 为 `https://mirrors.ustc.edu.cn/ubuntu/dists/`

2. 执行升级

```
# debtap -u
```

3. 使用方法

```
# debtap foobar.deb
```

注意：安装时会提示输入包名，以及 license。包名随意，license 就填 GPL 吧，上述操作完成后会在 deb 包同级目录生成 `foobar.tar.xz` 文件。

4. 静默模式

```
# debtap -q foobar.deb
```

选项 `-q` 略过了编辑元数据之外的所有问题。

5. 略过所有的问题（不推荐）

```
# debtap -Q foobar.deb
```

6. 安装转换好的软件包

```
# pacman -U foobar.tar.xz
```

7.3.3 安装 wps-office

在实际操作中使用上述方法转换 wps-office 并没有成功。可以按下述方法安装 wps-office 国际版。原因在于国际版没有必须登录的要求，同时比较精简，但是其下载速度远不如国内版，于是借用下载工具提前下载好 deb 软件包后。执行

```
$ yay -S wps-office
```

此时会弹出二个选项，1 为 wps-office 是国际版，2 为 wps-office-cn 是国内版，要据自己喜好选择就好。对于 wps-office 当进入下载过程时，按 `Ctrl+C` 终止程序，然后将提前下好的 deb 包复制到位置

```
~/.cache/yay/wps-office/
```

再执行

```
$ yay -S wps-office  
$ yay -S wps-office-mui-zh-cn
```

此时系统就可以正常根据 deb 包转换成 manjaro 包，转换完毕申请 root 权限，完成安装。但是在 linux 下 wpspdf 组件是多余的，因为我们有更好的 evince。此处我写了一个脚本用以删除 wpspdf 组件，以 root 权限执行脚本就完成了 wpspdf 组件的去除工作，现在可以开心的使用 wps 办公了。

附件：[RMwpspdf.sh](#)

中文版和国际版是主程序应该是不同的，因为在英文版中是没有一点联网功能的，包括”团队“，”会议“等侧边栏内容，但是一旦安装中文语言包后就会出现这两项。仔细分析后得出结论：主程序会检测语言设置，中英文分别有不同的设置。通过分析配置文件，成功去除了中文语言包后出现的这两个联网项。在最新版 wps-office 11.0.1.10702 中，出现了”账号登录、未保存、协作“，于是升级了优化脚本，适配此版去除了这些联网项。安装成功后如图

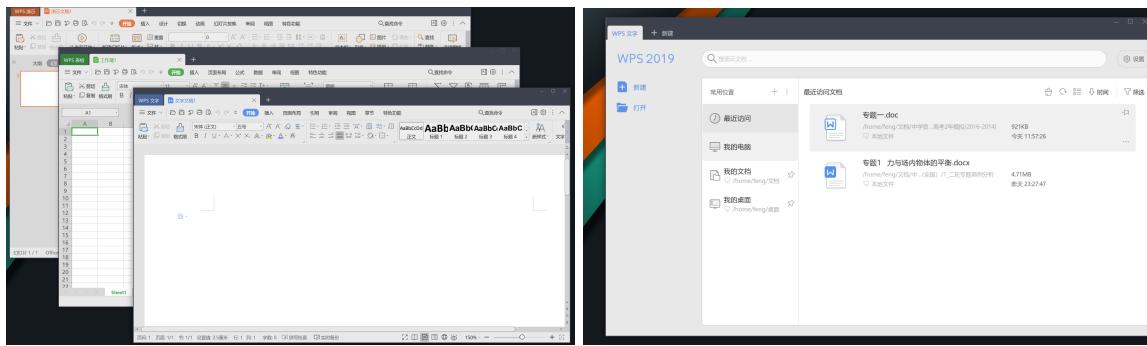


图 17: 去除 PDF 组件和多余联网组件后的样子