

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

# LINUX 命令、vim 使用与 GNU 软件编译

北京市计算中心 云平台事业部 姜骏

E-mail: [jiangjun@bcc.ac.cn](mailto:jiangjun@bcc.ac.cn)

北京科技大学 理化楼-308

2023.07.03

# Outline

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

## 1 Linux 操作系统概述

## 2 Linux 命令简介

## 3 文本编辑器 vi/vim 简介

## 4 Linux 下的软件安装与 Make

# Linux 操作系统简介

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

Linux 是 1991 年由芬兰的 Linus Benedict Torvalds 设计的、可运行在微机上的 UNIX 系统<sup>1</sup>。

Linux 基于 POSIX 和 UNIX 的多用户、多任务、支持多线程和多 CPU 的操作系统。它能运行主要的 UNIX 工具软件、应用程序和网络协议。它支持 32 位和 64 位硬件。Linux 继承了 UNIX 以网络为核心的设计思想，是性能稳定的多用户网络操作系统。

Linux 操作系统的特点：

- 完全免费
- 支持多用户、多任务工作方式
- 界面友好
- 支持多种硬件平台



<sup>1</sup> UNIX 是一种多任务分时操作系统，以“核 kernel”-“壳 (shell)”结构为特色，最初是由 AT&T Bell 实验室开发的，具有安全可靠、使用方便、开放性和可移植性等优点，可在各类机型上广泛使用。Internet 就是在 UNIX 基础上发展起来的。

# 为什么使用 Linux

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

## ■ 开放性和良好的可移植性

- 1 Linux 是完全开源和免费的
- 2 Linux 使用 C 语言对硬件资源的管理，可移植性好

## ■ 丰富的软件开发环境，系统高效

- 1 多种高级语言
- 2 强有力的调试手段
- 3 方便的文本编辑和应用程序

## ■ 自然科学与工程技术研究软件中的大部分是在 Linux 环境下开发与使用的

用户知道自己想要什么，也明白自己在做什么，并会为自己的行为负责

# Linux 的常见发行版本

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

## ■ 基于 RMP 系列



(a) Redhat



(b) Fedora



(c) CentOS

## ■ 基于 Debian 系列



(d) Debian



(e) Ubuntu

# Linux 发行版本的关系

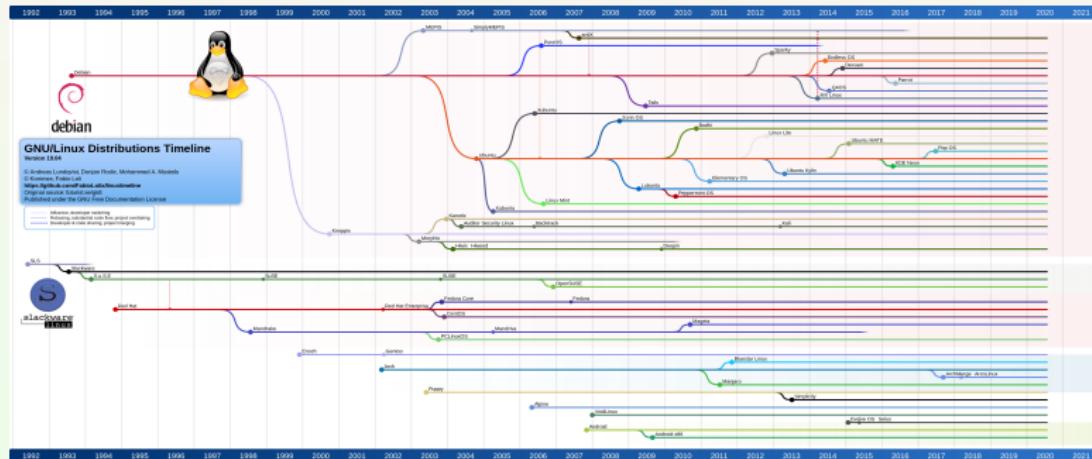
LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make



# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

## ■ mkdir: 创建目录

```
name@name-VirtualBox:~/TEST
name@name-VirtualBox:~/TEST$ mkdir test
name@name-VirtualBox:~/TEST$ ll
total 32
drwxrwxr-x  3 name name 4096 7月  26 15:01 .
drwxr-xr-x 21 name name 4096 7月  26 14:31 ..
drwxrwxr-x  2 name name 4096 7月  26 15:01 test/
-rw-rw-r--  2 name name  25 7月  26 12:04 test1
-rw-rw-r--  1 name name  25 7月  26 12:05 test2
-rw-rw-r--  1 name name   50 7月  26 12:05 test3
-rw-rw-r--  1 name name  25 7月  26 14:30 test4
-rw-rw-r--  2 name name  25 7月  26 12:04 test5
lrwxrwxrwx  1 name name   5 7月  26 14:57 test6 -> test1
name@name-VirtualBox:~/TEST$
```

## ■ cd: 改变工作目录

```
name@name-VirtualBox: ~
name@name-VirtualBox:~$ cd TEST/
name@name-VirtualBox:~/TEST$ cd /home/name/Softs/fftw-3.3.4
name@name-VirtualBox:~/Softs/fftw-3.3.4$ cd ..
name@name-VirtualBox:~/Softs$ cd ~
name@name-VirtualBox:~$ cd ~
name@name-VirtualBox:~$
```

改变绝对路径

返回上层目录

返回home目录

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

## ■ chmod: 改变工作目录的权限

```
name@name-VirtualBox: ~/TEST$ ll
total 20
drwxrwxr-x  2 name name 4096 7月  26 12:05 .
drwxr-xr-x 21 name name 4096 7月  26 12:05 ..
-rw-rw-r--  1 name name  25 7月  26 12:04 test1
-rw-rw-r--  1 name name  25 7月  26 12:05 test2
-rw-rw-r--  1 name name  50 7月  26 12:05 test3
name@name-VirtualBox:~/TEST$ chmod +x test1
name@name-VirtualBox:~/TEST$ ll
total 20
drwxrwxr-x  2 name name 4096 7月  26 12:05 .
drwxr-xr-x 21 name name 4096 7月  26 12:05 ..
-rwxrwxr-x  1 name name  25 7月  26 12:04 test1* ← 全部用户增加执行权
-rw-rw-r--  1 name name  25 7月  26 12:05 test2
-rw-rw-r--  1 name name  50 7月  26 12:05 test3
name@name-VirtualBox:~/TEST$ chmod 664 test1
name@name-VirtualBox:~/TEST$ ll
total 20
drwxrwxr-x  2 name name 4096 7月  26 12:05 .
drwxr-xr-x 21 name name 4096 7月  26 12:05 ..
-rw-rw-r--  1 name name  25 7月  26 12:04 test1* ← 文件属主有读写权
-rw-rw-r--  1 name name  25 7月  26 12:05 test2* ← 同组用户有读写权
-rw-rw-r--  1 name name  50 7月  26 12:05 test3* ← 其他用户有都权
name@name-VirtualBox:~/TEST$
```

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

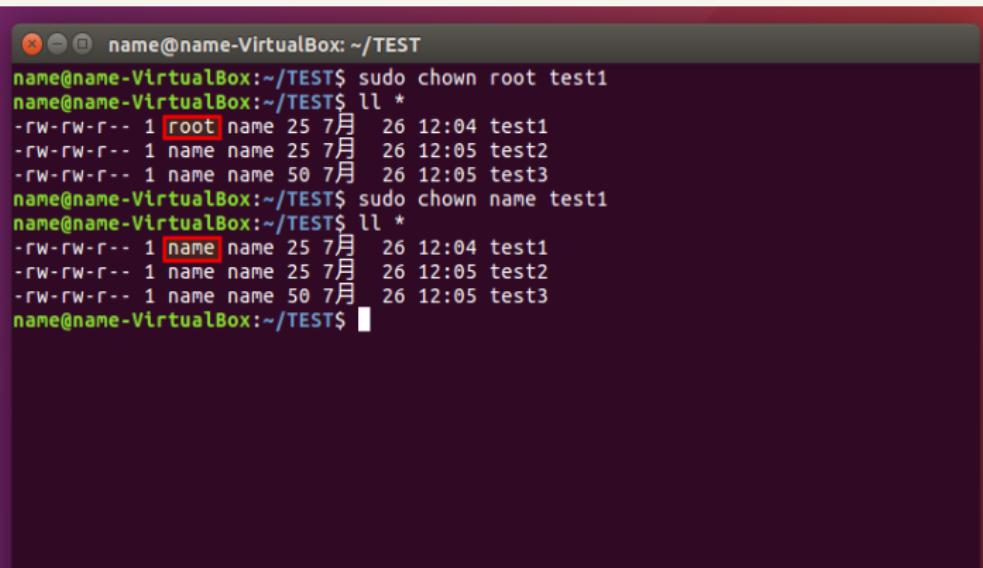
Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ chown: 改变文件的属主



The screenshot shows a terminal window titled "name@name-VirtualBox: ~/TEST". The user runs two commands: "sudo chown root test1" and "ls \*". The output shows three files: test1, test2, and test3. The first file, test1, has its ownership changed from "name" to "root". The second command, "ls \*", lists all files in the current directory.

```
name@name-VirtualBox:~/TEST$ sudo chown root test1
name@name-VirtualBox:~/TEST$ ls *
-rw-rw-r-- 1 root name 25 7月 26 12:04 test1
-rw-rw-r-- 1 name name 25 7月 26 12:05 test2
-rw-rw-r-- 1 name name 50 7月 26 12:05 test3
name@name-VirtualBox:~/TEST$ sudo chown name test1
name@name-VirtualBox:~/TEST$ ls *
-rw-rw-r-- 1 name name 25 7月 26 12:04 test1
-rw-rw-r-- 1 name name 25 7月 26 12:05 test2
-rw-rw-r-- 1 name name 50 7月 26 12:05 test3
name@name-VirtualBox:~/TEST$
```

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

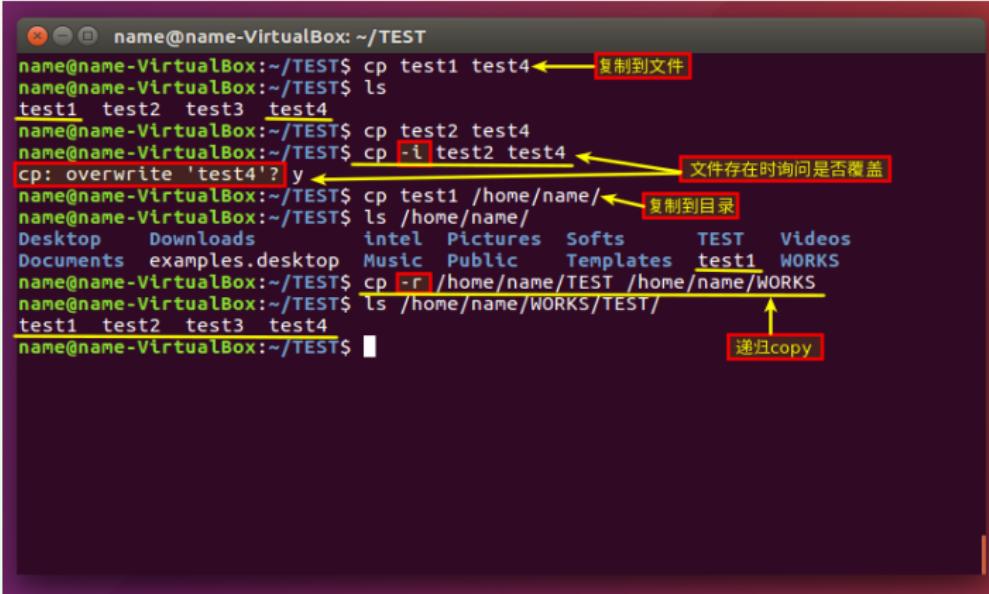
Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ cp: 将文件 copy 到另一个文件或另一目录



The screenshot shows a terminal window with the following session:

```
name@name-VirtualBox:~/TEST$ cp test1 test4 ← 复制到文件
name@name-VirtualBox:~/TEST$ ls
test1 test2 test3 test4
name@name-VirtualBox:~/TEST$ cp test2 test4
name@name-VirtualBox:~/TEST$ cp -i test2 test4 ← 文件存在时询问是否覆盖
cp: overwrite 'test4'? y
name@name-VirtualBox:~/TEST$ cp test1 /home/name/ ← 复制到目录
name@name-VirtualBox:~/TEST$ ls /home/name/
Desktop Downloads intel Pictures Softs TEST Videos
Documents examples.desktop Music Public Templates test1 WORKS
name@name-VirtualBox:~/TEST$ cp -r /home/name/TEST /home/name/WORKS
name@name-VirtualBox:~/TEST$ ls /home/name/WORKS/TEST/
test1 test2 test3 test4
name@name-VirtualBox:~/TEST$
```

Annotations in red boxes with arrows point to specific parts of the command line:

- "复制到文件" points to the first command: `cp test1 test4`.
- "文件存在时询问是否覆盖" points to the confirmation prompt: `cp: overwrite 'test4'? y`.
- "复制到目录" points to the command: `cp test1 /home/name/`.
- "递归copy" points to the command: `cp -r /home/name/TEST /home/name/WORKS`.

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

## ■ diff: 逐行比较两个文件，列出不同

```
name@name-VirtualBox:~/TEST$ diff test1 test2
1c1
< This is CONTENT of test1
...
> This is CONTENT of test2
name@name-VirtualBox:~/TEST$ diff -e test1 test2
1c1
This is CONTENT of test2
.
name@name-VirtualBox:~/TEST$
```

## ■ find: 搜索文件 (文件所在位置检索)

```
name@name-VirtualBox:~/TEST$ find /opt/intel -name 'lapack' -print
/opt/intel/composer_xe_2013.0.079/examples/lapack
name@name-VirtualBox:~/TEST$ find /home/name/Softs/fftw-3.3.4/bin -user name -print
/home/name/Softs/fftw-3.3.4/bin 查找名为lapack的文件
/home/name/Softs/fftw-3.3.4/bin/fftw-wisdom-to-conf 找出属主为name的文件
/home/name/Softs/fftw-3.3.4/bin/fftw-wisdom
name@name-VirtualBox:~/TEST$
```

# 与目录和文件有关的命令

LINUX 命令、vim 使用  
与 GNU 软件编译

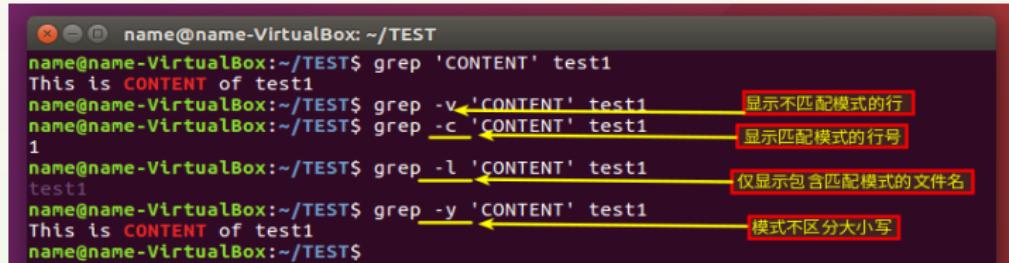
Linux 操作系统  
综述

Linux 命令简介

文本编辑器  
vi/vim 简介

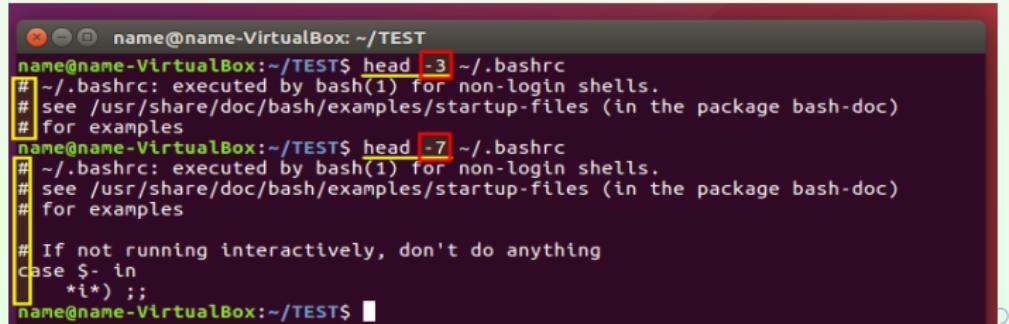
Linux 下的软件安装与  
Make

## ■ grep: 按给定模式搜索文件 (文件所含内容搜索)



```
name@name-VirtualBox:~/TEST$ grep 'CONTENT' test1
This is CONTENT of test1
name@name-VirtualBox:~/TEST$ grep -v 'CONTENT' test1      显示不匹配模式的行
name@name-VirtualBox:~/TEST$ grep -c 'CONTENT' test1      显示匹配模式的行号
1
name@name-VirtualBox:~/TEST$ grep -l 'CONTENT' test1      仅显示包含匹配模式的文件名
test1
name@name-VirtualBox:~/TEST$ grep -y 'CONTENT' test1      模式不区分大小写
This is CONTENT of test1
name@name-VirtualBox:~/TEST$
```

## ■ head: 显示指定文件前若干行



```
name@name-VirtualBox:~/TEST$ head -3 ~/.bashrc
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples
name@name-VirtualBox:~/TEST$ head -7 ~/.bashrc
# ~/.bashrc: executed by bash(1) for non-login shells.
# see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
# for examples

# If not running interactively, don't do anything
case $- in
    *i*) ;;
esac
name@name-VirtualBox:~/TEST$
```

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ ln: 建立文件链接 (link)

```
name@name-VirtualBox:~/TEST$ ln test1 test5
name@name-VirtualBox:~/TEST$ ll
total 28
drwxrwxr-x 2 name name 4096 7月 26 14:57 .
drwxr-xr-x 21 name name 4096 7月 26 14:31 ..
-rw-rw-r-- 2 name name 25 7月 26 12:04 test1
-rw-rw-r-- 1 name name 25 7月 26 12:05 test2
-rw-rw-r-- 1 name name 50 7月 26 12:05 test3
-rw-rw-r-- 1 name name 25 7月 26 14:30 test4
-rw-rw-r-- 2 name name 25 7月 26 12:04 test5
name@name-VirtualBox:~/TEST$ ln -s test1 test6
name@name-VirtualBox:~/TEST$ ll
total 28
drwxrwxr-x 2 name name 4096 7月 26 14:57 .
drwxr-xr-x 21 name name 4096 7月 26 14:31 ..
-rw-rw-r-- 2 name name 25 7月 26 12:04 test1
-rw-rw-r-- 1 name name 25 7月 26 12:05 test2
-rw-rw-r-- 1 name name 50 7月 26 12:05 test3
-rw-rw-r-- 1 name name 25 7月 26 14:30 test4
-rw-rw-r-- 2 name name 25 7月 26 12:04 test5
lrwxrwxrwx 1 name name 5 7月 26 14:57 test6 -> test1
name@name-VirtualBox:~/TEST$
```

**注意 1:** cp -r 会将链接文件 copy 成独立文件，如果需要保护文件符号链接 (软链接) 属性，建议用 tar 命令

**注意 2:** 不在同一文件系统 (如位于不同磁盘分区) 中的文件只能用符号链接，不能用硬链接

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ ls: 列出目录中的内容

```
name@name-VirtualBox:~/TEST$ ls
test1 test2 test3 test4 test5 test6
name@name-VirtualBox:~/TEST$ ls -a          列出所有文件，包括隐藏文件
[.] [..] test1 test2 test3 test4 test5 test6
name@name-VirtualBox:~/TEST$ ls -la         列出文件长格式列表
total 28
drwxrwxr-x  2 name name 4096 7月 26 14:57 .
drwxr-xr-x  21 name name 4096 7月 26 14:31 ..
-rw-rw-r--  2 name name 25 7月 26 12:04 test1
-rw-rw-r--  1 name name 25 7月 26 12:05 test2
-rw-rw-r--  1 name name 50 7月 26 12:05 test3
-rw-rw-r--  1 name name 25 7月 26 14:30 test4
-rw-rw-r--  2 name name 25 7月 26 12:04 test5
lrwxrwxrwx  1 name name 5 7月 26 14:57 test6 -> test1
name@name-VirtualBox:~/TEST$ ls -lta        文件类型及许可机制
total 28
drwxrwxr-x  2 name name 4096 7月 26 14:57 .
drwxr-xr-x  21 name name 4096 7月 26 14:31 ..
-rw-rw-r--  2 name name 25 7月 26 12:04 test1
-rw-rw-r--  1 name name 50 7月 26 12:05 test2
-rw-rw-r--  1 name name 25 7月 26 14:30 test3
-rw-rw-r--  2 name name 25 7月 26 12:04 test4
lrwxrwxrwx  1 name name 5 7月 26 14:57 test6 -> test1
-rw-rw-r--  1 name name 25 7月 26 12:04 test5
-rw-rw-r--  1 name name 25 7月 26 14:30 test4
-rw-rw-r--  1 name name 50 7月 26 12:05 test3
-rw-rw-r--  1 name name 25 7月 26 12:05 test2
-rw-rw-r--  2 name name 25 7月 26 12:04 test1
drwxr-xr-x  21 name name 4096 7月 26 14:31 ..
drwxrwxr-x  2 name name 4096 7月 26 14:57 .
name@name-VirtualBox:~/TEST$ l             用逆序列表
```

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ more: 分屏显示信息

```
name@name-VirtualBox:~/TEST$ ls /home/name/Softs/openmpi-1.8.4/lib/ | more
libmpl.a
libmpl.la
libmpl_mpifh.a
libmpl_mpifh.la
libmpl_mpifh.so
libmpl_mpifh.so.2
libmpl_mpifh.so.2.5.0
libmpl.so
libmpl.so.1
libmpl.so.1.6.0
libmpl_usenpif08.a
libmpl_usenpif08.la
libmpl_usenpif08.so
libmpl_usenpif08.so.0
libmpl_usenpif08.so.0.6.0
libmpl_usenpl_ignore_tkr.a
libmpl_usenpl_ignore_tkr.la
libmpl_usenpl_ignore_tkr.so
libmpl_usenpl_ignore_tkr.so.0
libmpl_usenpl_ignore_tkr.so.0.1.0
libompitrace.a
libompitrace.la
libompitrace.so
libompitrace.so.0
--More--
```

- -C 清屏而非滚屏
- -f 遇长行不折回
- i 空格 显示后续 i 行
- q 或 Q 退出more
- = 显示当前行号
- h 帮助信息

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ mv: 文件或目录的移动或更名

```
name@name-VirtualBox:~/TEST$ mv test2 test7
name@name-VirtualBox:~/TEST$ ll
total 32
drwxrwxr-x 3 name name 4096 7月 26 15:08 .
drwxr-xr-x 21 name name 4096 7月 26 14:31 ../
drwxrwxr-x 2 name name 4096 7月 26 15:01 test/
-rw-rw-r-- 2 name name 25 7月 26 12:04 test1
-rw-rw-r-- 1 name name 50 7月 26 12:05 test3
-rw-rw-r-- 1 name name 25 7月 26 14:30 test4
-rw-rw-r-- 2 name name 25 7月 26 12:04 test5
lrwxrwxrwx 1 name name 5 7月 26 14:57 test6 -> test1
-rw-rw-r-- 1 name name 25 7月 26 12:05 test7
name@name-VirtualBox:~/TEST$ mv test7 test
name@name-VirtualBox:~/TEST$ ll
total 28
drwxrwxr-x 3 name name 4096 7月 26 15:08 .
drwxr-xr-x 21 name name 4096 7月 26 14:31 ../
drwxrwxr-x 2 name name 4096 7月 26 15:08 test/
-rw-rw-r-- 2 name name 25 7月 26 12:04 test1
-rw-rw-r-- 1 name name 50 7月 26 12:05 test3
-rw-rw-r-- 1 name name 25 7月 26 14:30 test4
-rw-rw-r-- 2 name name 25 7月 26 12:04 test5
lrwxrwxrwx 1 name name 5 7月 26 14:57 test6 -> test1
name@name-VirtualBox:~/TEST$ ll test
total 12
drwxrwxr-x 2 name name 4096 7月 26 15:08 .
drwxrwxr-x 3 name name 4096 7月 26 15:08 ../
-rw-rw-r-- 1 name name 25 7月 26 12:05 test7
name@name-VirtualBox:~/TEST$
```

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ pwd: 显示当前目录绝对路径

```
name@name-VirtualBox: ~
name@name-VirtualBox:~/TEST$ pwd
/home/name/TEST
name@name-VirtualBox:~/TEST$ cd -
name@name-VirtualBox:~$ pwd
/home/name
name@name-VirtualBox:~$
```

## ■ rm: 删除文件或目录

```
name@name-VirtualBox: ~/TEST
name@name-VirtualBox:~/TEST$ rm test5
name@name-VirtualBox:~/TEST$ ll
total 24
drwxrwxr-x  3 name name 4096 7月  26 15:10 .
drwxr-xr-x  21 name name 4096 7月  26 14:31 ../
drwxrwxr-x  2 name name 4096 7月  26 15:08 test/
-rw-rw-r--  1 name name   25 7月  26 12:04 test1
-rw-rw-r--  1 name name   50 7月  26 12:05 test3
-rw-rw-r--  1 name name   25 7月  26 14:30 test4
lrwxrwxrwx  1 name name     5 7月  26 14:57 test6 -> test1
name@name-VirtualBox:~/TEST$ rm -i test6
rm: remove symbolic link 'test6'? y
name@name-VirtualBox:~/TEST$ rm test
rm: cannot remove 'test': Is a directory
name@name-VirtualBox:~/TEST$ rm -r test
name@name-VirtualBox:~/TEST$ ll
total 20
drwxrwxr-x  2 name name 4096 7月  26 15:11 .
drwxr-xr-x  21 name name 4096 7月  26 14:31 ../
-rw-rw-r--  1 name name   25 7月  26 12:04 test1
-rw-rw-r--  1 name name   50 7月  26 12:05 test3
-rw-rw-r--  1 name name   25 7月  26 14:30 test4
name@name-VirtualBox:~/TEST$
```

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

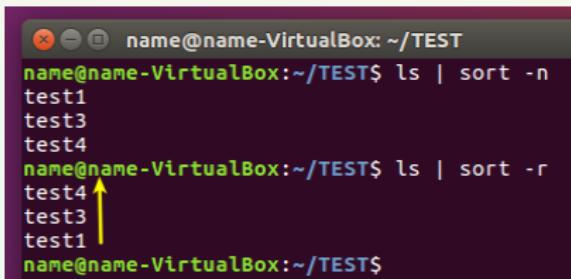
Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ sort: 对指定文件按行排序



The screenshot shows a terminal window with the following session:

```
name@name-VirtualBox:~/TEST$ name@name-VirtualBox:~/TEST$ ls | sort -n
test1
test3
test4
name@name-VirtualBox:~/TEST$ ls | sort -r
test4
test3
test1
name@name-VirtualBox:~/TEST$
```

A yellow arrow points to the second 'ls' command, highlighting the pipe operator '|'. The terminal window has a dark background with light-colored text.

- **-b** 忽略开头的空格和制表符
- **-d** 仅字母、数字、空格按字典排序
- **-f** 不区分字母大小写
- **-n** 按数字升序排列
- **-r** 按当前顺序的逆序排序
- **-u** 忽略重复行
- **-o** 指定输出文件名

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ tail: 显示指定文件的最后部分

```
name@name-VirtualBox:~/TEST$ tail -3 ~/.bashrc
export BERRYPI_PYTHON=/usr/bin/python2.7
alias berrypi="${BERRYPI_PYTHON} ${BERRYPI_PATH}/berrypi"
# --- BERRYPI END ---
name@name-VirtualBox:~/TEST$ tail -5 ~/.bashrc
# --- BERRYPI START ---
export BERRYPI_PATH=$WIENROOT/SRC_BerryPI/BerryPI
export BERRYPI_PYTHON=/usr/bin/python2.7
alias berrypi="${BERRYPI_PYTHON} ${BERRYPI_PATH}/berrypi"
# --- BERRYPI END ---
name@name-VirtualBox:~/TEST$ tail -10 ~/.bashrc
export PATH=$PATH:$WIENROOT:..
ulimit -s unlimited
alias octave="octave -p $OCTAVE_PATH"
#
# added by WIEN2k: END
# --- BERRYPI START ---
export BERRYPI_PATH=$WIENROOT/SRC_BerryPI/BerryPI
export BERRYPI_PYTHON=/usr/bin/python2.7
alias berrypi="${BERRYPI_PYTHON} ${BERRYPI_PATH}/berrypi"
# --- BERRYPI END ---
name@name-VirtualBox:~/TEST$
```

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ tar: 将若干文件存档或读取存档文件

```
name@name-VirtualBox:~/TEST/test
name@name-VirtualBox:~/TEST$ tar -cvzf test.tar.gz *
test1
test3
test4
name@name-VirtualBox:~/TEST$ ll
total 24
drwxrwxr-x 2 name name 4096 7月 26 15:25 .
drwxr-xr-x 21 name name 4096 7月 26 14:31 ..
-rw-rw-r-- 1 name name 173 7月 26 15:20 test1
-rw-rw-r-- 1 name name 50 7月 26 12:05 test3
-rw-rw-r-- 1 name name 25 7月 26 14:30 test4
-rw-rw-r-- 1 name name 403 7月 26 15:25 test.tar.gz
name@name-VirtualBox:~/TEST$ mkdir test
name@name-VirtualBox:~/TEST$ mv test.tar.gz test
name@name-VirtualBox:~/TEST$ cd test/
name@name-VirtualBox:~/TEST/test$ tar -xvzf test.tar.gz
test1
test3
test4
name@name-VirtualBox:~/TEST/test$ ll
total 24
drwxrwxr-x 2 name name 4096 7月 26 15:26 .
drwxrwxr-x 3 name name 4096 7月 26 15:26 ..
-rw-rw-r-- 1 name name 173 7月 26 15:20 test1
-rw-rw-r-- 1 name name 50 7月 26 12:05 test3
-rw-rw-r-- 1 name name 25 7月 26 14:30 test4
-rw-rw-r-- 1 name name 403 7月 26 15:25 test.tar.gz
name@name-VirtualBox:~/TEST/test$
```

# 与目录和文件有关的命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ wc: 统计指定文件的行数、词数和字符数

```
name@name-VirtualBox:~/TEST/test$ wc test4
 1 5 25 test4
name@name-VirtualBox:~/TEST/test$ wc -c test4      显示字符数
25 test4
name@name-VirtualBox:~/TEST/test$ wc -l test4      显示行数
1 test4
name@name-VirtualBox:~/TEST/test$ wc -w test4      显示词数
5 test4
name@name-VirtualBox:~/TEST/test$ ls -l
total 16
-rw-rw-r-- 1 name name 173 7月 26 15:20 test1
-rw-rw-r-- 1 name name 50 7月 26 12:05 test3
-rw-rw-r-- 1 name name 25 7月 26 14:30 test4
-rw-rw-r-- 1 name name 403 7月 26 15:25 test.tar.gz
name@name-VirtualBox:~/TEST/test$ ls -l | wc -l
5
name@name-VirtualBox:~/TEST/test$
```

# 状态信息查询命令

LINUX 命令、vim 使用与 GNU 软件编译

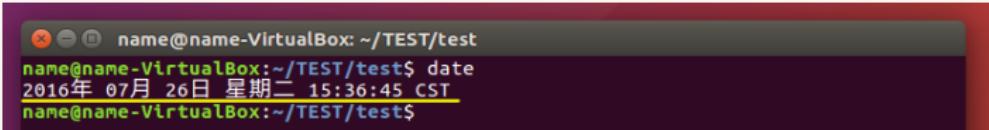
Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

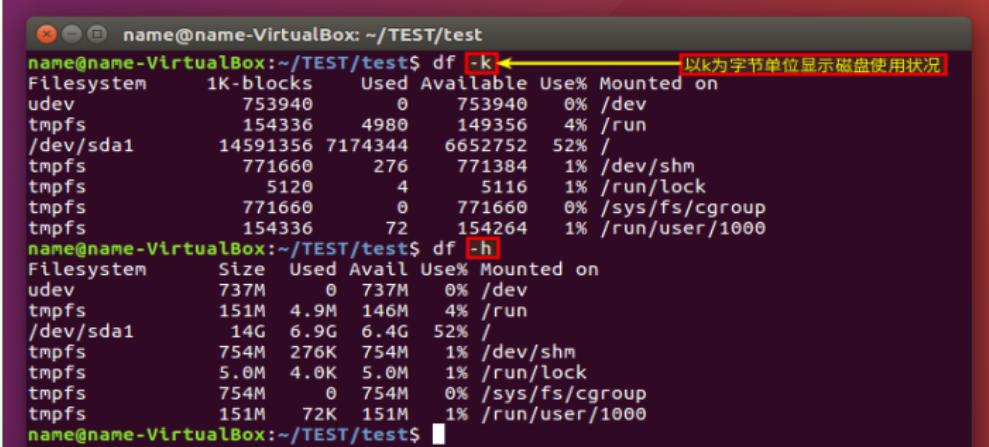
Linux 下的软件安装与  
Make

## ■ date: 显示日期与时间



```
name@name-VirtualBox: ~/TEST/test
name@name-VirtualBox:~/TEST/test$ date
2016年 07月 26日 星期二 15:36:45 CST
name@name-VirtualBox:~/TEST/test$
```

## ■ df: 查询磁盘空间使用情况



```
name@name-VirtualBox:~/TEST/test$ df -k
Filesystem      1K-blocks    Used Available Use% Mounted on
udev             753940      0   753940  0% /dev
tmpfs            154336   4980   149356  4% /run
/dev/sda1        14591356 7174344   6652752  52% /
tmpfs            771660    276   771384  1% /dev/shm
tmpfs             5120      4   5116  1% /run/lock
tmpfs            771660      0   771660  0% /sys/fs/cgroup
tmpfs            154336     72   154264  1% /run/user/1000
name@name-VirtualBox:~/TEST/test$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev             737M    0   737M  0% /dev
tmpfs            151M  4.9M  146M  4% /run
/dev/sda1        14G  6.9G  6.4G  52% /
tmpfs            754M 276K  754M  1% /dev/shm
tmpfs             5.0M  4.0K  5.0M  1% /run/lock
tmpfs            754M    0   754M  0% /sys/fs/cgroup
tmpfs            151M  72K  151M  1% /run/user/1000
name@name-VirtualBox:~/TEST/test$
```

# 状态信息查询命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ du: 统计目录或文件所占磁盘空间的大小

```
name@name-VirtualBox:~/TEST/test$ du -h /home/name/ | more
8.0K  /home/name/Softs/fftw-3.3.4/lib/pkgconfig
1.6M  /home/name/Softs/fftw-3.3.4/lib
16K   /home/name/Softs/fftw-3.3.4/share/man/man1 ← 显示每个目录和文件的大小
20K   /home/name/Softs/fftw-3.3.4/share/man
384K  /home/name/Softs/fftw-3.3.4/share/info
408K  /home/name/Softs/fftw-3.3.4/share
1.2M  /home/name/Softs/fftw-3.3.4/bin
212K  /home/name/Softs/fftw-3.3.4/include
3.3M  /home/name/Softs/fftw-3.3.4
16K   /home/name/Softs/openmpi-1.8.4/etc
180K  /home/name/Softs/openmpi-1.8.4/lib/openmpi
24K   /home/name/Softs/openmpi-1.8.4/lib/pkgconfig
19M   /home/name/Softs/openmpi-1.8.4/lib
8.0K  /home/name/Softs/openmpi-1.8.4/share/openmpi/amca-param-sets
8.0K  /home/name/Softs/openmpi-1.8.4/share/openmpi/doc
452K  /home/name/Softs/openmpi-1.8.4/share/openmpi
40K   /home/name/Softs/openmpi-1.8.4/share/man/man7
264K  /home/name/Softs/openmpi-1.8.4/share/man/man1
1.9M  /home/name/Softs/openmpi-1.8.4/share/man/man3
2.2M  /home/name/Softs/openmpi-1.8.4/share/man
2.6M  /home/name/Softs/openmpi-1.8.4/share
344K  /home/name/Softs/openmpi-1.8.4/bin
12K   /home/name/Softs/openmpi-1.8.4/include/mpp
468K  /home/name/Softs/openmpi-1.8.4/include
22M  /home/name/Softs/openmpi-1.8.4
25M  /home/name/Softs
4.0K  /home/name/Public
20K  /home/name/intel/lsm/rm
24K  /home/name/intel/lsm
--More--
```

```
name@name-VirtualBox:~/TEST/test$ du -s /home/name/ ← 统计name目录所占空间，不显示子目录结果
69960  /home/name/
name@name-VirtualBox:~/TEST/test$ du -sh /home/name/
68M   /home/name/
name@name-VirtualBox:~/TEST/test$
```

# 状态信息查询命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ ps: 检查进程状态

```
name@name-VirtualBox:~/TEST/test$ ps
 PID TTY          TIME CMD
 1928 pts/4    00:00:01 bash
 3198 pts/4    00:00:00 ps
name@name-VirtualBox:~/TEST/test$ ps aux | grep bash
 name   1928  0.0  0.3  8836  5236 pts/4    Ss   12:03  0:01 bash
 name   3200  0.0  0.0  6856  892 pts/4    S+   15:52  0:00 grep --color=auto bash
name@name-VirtualBox:~/TEST/test$
```

## ■ time: 统计程序或命令运行的时间

```
name@name-VirtualBox:~/TEST/test$ time ls
test1 test3 test4 test.tar.gz

real    0m0.003s
user    0m0.000s
sys     0m0.000s
name@name-VirtualBox:~/TEST/test$
```

## ■ w: 检查状态

```
name@name-VirtualBox:~/TEST/test$ w
15:58:18 up 3:59, 1 user, load average: 0.00, 0.06, 0.11
USER TTY          FROM          LOGIN@ IDLE JCPU   PCPU WHAT
name  tty7 :0        12:00   3:59m  1:22   0.22s /sbin/upstart --user
name@name-VirtualBox:~/TEST/test$
```

# 状态信息查询命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

- **whereis**: 确定命令位置
- **which**: 确定命令位置
- **who**: 列出正在使用系统的用户
- **whoami**: 显示正在使用本终端的用户名

```
name@name-VirtualBox:~/TEST/test$ whereis mpif90
mpif90: /home/name/Softs/openmpi-1.8.4/bin/mpif90
name@name-VirtualBox:~/TEST/test$ which mpif90
/home/name/Softs/openmpi-1.8.4/bin/mpif90
name@name-VirtualBox:~/TEST/test$ who
name      tty7          2016-07-26 12:00 (:0)
name@name-VirtualBox:~/TEST/test$ whoami
name
name@name-VirtualBox:~/TEST/test$
```

# 网络查询命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

- **arp**: 网络和 IP 地址查询
- **ifconfig**: 网络和 MAC 地址查询

```
name@name-VirtualBox:~/TEST/test$ arp
Address          Hwtype Hwaddress          Flags Mask      Iface
bogon           ether   52:54:00:12:35:03      C
name@name-VirtualBox:~/TEST/test$ ifconfig
enp0s3    Link encap:Ethernet Hwaddr 08:00:27:b0:1c:c5
          inet addr:10.0.2.15 Bcast:10.0.2.255 Mask:255.255.255.0
                  inet6 addr: fe80::7103:57ad:c39c:a5c6/64 Scope:Link
                      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                      RX packets:14 errors:0 dropped:0 overruns:0 frame:0
                      TX packets:91 errors:0 dropped:0 overruns:0 carrier:0
                      collisions:0 txqueuelen:1000
                      RX bytes:2430 (2.4 KB)  TX bytes:9963 (9.9 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
                      UP LOOPBACK RUNNING MTU:65536 Metric:1
                      RX packets:44 errors:0 dropped:0 overruns:0 frame:0
                      TX packets:44 errors:0 dropped:0 overruns:0 carrier:0
                      collisions:0 txqueuelen:1
                      RX bytes:3043 (3.0 KB)  TX bytes:3043 (3.0 KB)

name@name-VirtualBox:~/TEST/test$
```

# 远程登录命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ ssh: 远程登录

```
name@name-VirtualBox:~/TEST$ ssh cywang@192.168.30.15
The authenticity of host '192.168.30.15 (192.168.30.15)' can't be established.
RSA key fingerprint is SHA256:qaI1xBWSFqqpm5bYJFesseiBxt7L0TUX89TRAZHTBYk.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.30.15' (RSA) to the list of known hosts.
cywang@192.168.30.15's password:
Permission denied, please try again.
cywang@192.168.30.15's password:
Permission denied, please try again.
cywang@192.168.30.15's password:

name@name-VirtualBox:~/TEST$ ssh jiangjun@192.168.30.15
jiangjun@192.168.30.15's password:
Last login: Thu Jul 28 11:49:19 2016 from 192.168.113.107

        Welcome to Beijing Computing Center!
        Any questions please contact your sales or supports.
        Site:      http://www.bcc.ac.cn
        -----
        Please download the latest mannuals from
        /share/software/doc
        -----
        command "myaccount" can be used to check user's ExpireDate.
        command "myquota" can be used to check user's Amount and Quota.
        -----
        Please login tyh-build1 to compile.
        -----

[jiangjun@tyh-login ~]$
```

# 远程无密码登录

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

```

Terminal File Edit View Search Terminal Help
name@name-VirtualBox:~/.ssh/
name@name-VirtualBox:~/.ssh$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/name/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/name/.ssh/id_rsa.
Your public key has been saved in /home/name/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:f13ps090kb3j0xUzCzCxyxE18dm7Nddhy5SQycdV0k name@name-VirtualBox
The key's randomart image is:
+---[RSA 2048]---+
| .+++=+ |
| .+#+. |
| . +*#* |
| . . 0.E* |
| . S o . 0o |
| . = o . 0= |
| .o= o . . |
| .o+ . . |
|=+-----|
+---[SHA256]----+
name@name-VirtualBox:~/.ssh$ scp id_rsa.pub jiangjun@192.168.30.15:~
jiangjun@192.168.30.15's password:
100% 462 0.4KB/s 00:00
name@name-VirtualBox:~/.ssh$ ssh jiangjun@192.168.30.15
jiangjun@192.168.30.15's password:
Last login: Thu Jul 28 12:02:35 2016 from 192.168.113.107

Welcome to Beijing Computing Center!
Any questions please contact your sales or supports.
Site: http://www.bcc.ac.cn
-----
Please download the latest manuals from
/share/software/doc

command "myaccount" can be used to check user's ExpireDate.
command "myquota" can be used to check user's Amount and Quota.
-----
Please login tyh-bullid to compile.
-----

[...]
[jiangjun@tyh-login ~]$ cat id_rsa.pub > ~/.ssh/authorized_keys
chmod 644 ~/.ssh/authorized_keys
chmod 700 ~/.ssh
[jiangjun@tyh-login ~]$ exit
logout
Connection to 192.168.30.15 closed.
name@name-VirtualBox:~/.ssh$ ssh jiangjun@192.168.30.15
Last login: Thu Jul 28 12:08:25 2016 from 192.168.113.107

Welcome to Beijing Computing Center!
Any questions please contact your sales or supports.
Site: http://www.bcc.ac.cn
-----
Please download the latest manuals from
/share/software/doc

command "myaccount" can be used to check user's ExpireDate.
command "myquota" can be used to check user's Amount and Quota.
-----
Please login tyh-bullid to compile.
-----

[...]

```

# 远程传输命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ scp: 远程传输

```
name@name-VirtualBox:~/TEST$ scp jiangjun@192.168.30.15:~/WORKS/VASP_Cal/Si_MD_1/OUTCAR /home/name/WORKS/TEST/
jiangjun@192.168.30.15's password: download
OUTCAR                                         100%   160KB 159.9KB/s  00:00
name@name-VirtualBox:~/TEST$ scp /home/name/WORKS/TEST/test1 jiangjun@192.168.30.15:~/WORKS
jiangjun@192.168.30.15's password: upload
test1                                           100%     25    0.0KB/s  00:00
name@name-VirtualBox:~/TEST$
```

# 程序运行命令

LINUX 命令、vim 使用与 GNU 软件编译

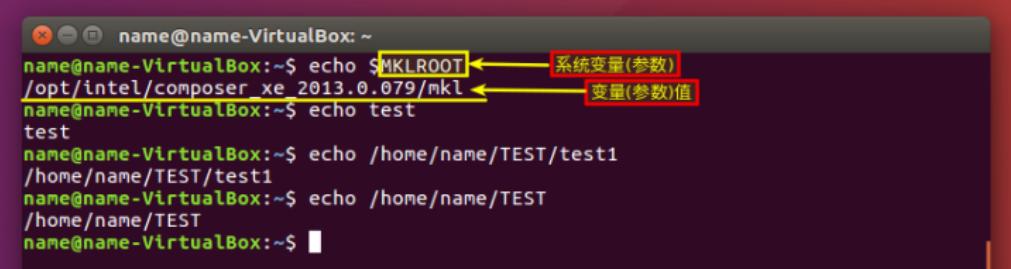
Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ echo: 参数回应至标准输出



```
name@name-VirtualBox: ~
name@name-VirtualBox:~$ echo $MKLROOT
/opt/intel/composer_xe_2013.0.079/mkl
name@name-VirtualBox:~$ echo test
test
name@name-VirtualBox:~$ echo /home/name/TEST/test1
/home/name/TEST/test1
name@name-VirtualBox:~$ echo /home/name/TEST
/home/name/TEST
name@name-VirtualBox:~$
```

Annotations in the screenshot:

- echo \$MKLROOT: 系统变量(参数)
- /opt/intel/composer\_xe\_2013.0.079/mkl: 变量(参数)值

## ■ kill: 终止指定进程



```
name@name-VirtualBox: ~
name@name-VirtualBox:~$ ps
 PID TTY      TIME CMD
 4325 pts/4    00:00:00 bash
 4345 pts/4    00:00:00 ps
name@name-VirtualBox:~$ kill -9 4325
```

Annotations in the screenshot:

- ps: 强制进程终止
- 9: 进程PID号

# 程序运行命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ tee: 将标准输出同时复制到文件



The screenshot shows a terminal window with the following session:

```
name@name-VirtualBox: ~
name@name-VirtualBox:~$ ps | tee TEST/test6
  PID TTY      TIME CMD
11857 pts/4    00:00:00 bash
11889 pts/4    00:00:00 ps
11890 pts/4    00:00:00 tee
name@name-VirtualBox:~$ cat TEST/test6
  PID TTY      TIME CMD
11857 pts/4    00:00:00 bash
11889 pts/4    00:00:00 ps
11890 pts/4    00:00:00 tee
name@name-VirtualBox:~$
```

A red box highlights the command `ps | tee TEST/test6`. A red arrow points from the text "屏显结果写入test6" to the word `tee` in the command line.

# 其它命令

LINUX 命令、vim 使用与 GNU 软件编译

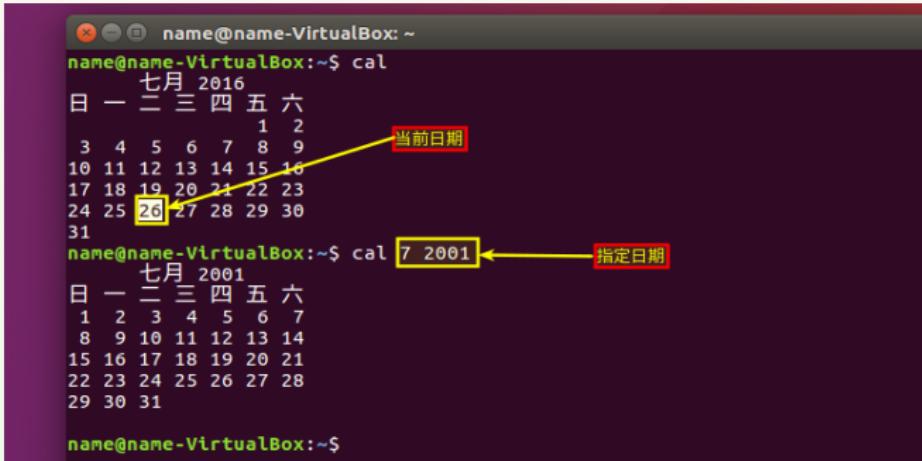
Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## ■ cal: 打印日历



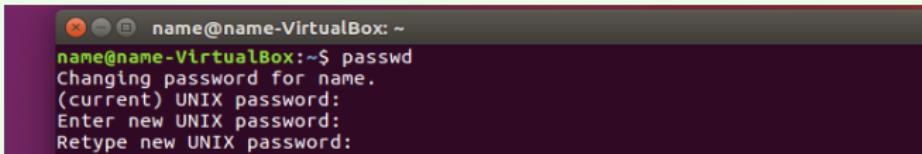
The screenshot shows a terminal window with the following content:

```
name@name-VirtualBox: ~
name@name-VirtualBox:~$ cal
                七月 2016
  日 一 二 三 四 五 六
          1  2
  3  4  5  6  7  8  9
  10 11 12 13 14 15 16
  17 18 19 20 21 22 23
  24 25 26 27 28 29 30
  31
name@name-VirtualBox:~$ cal 7 2001
                七月 2001
  日 一 二 三 四 五 六
          1  2  3  4  5  6  7
  8  9 10 11 12 13 14
  15 16 17 18 19 20 21
  22 23 24 25 26 27 28
  29 30 31

name@name-VirtualBox:~$
```

A yellow arrow points from the text "当前日期" to the date "26" in the first calendar output. Another yellow arrow points from the text "指定日期" to the date "2001" in the second command line.

## ■ passwd: 变更当前账户的密码



The screenshot shows a terminal window with the following content:

```
name@name-VirtualBox: ~
name@name-VirtualBox:~$ passwd
Changing password for name.
(current) UNIX password:
Enter new UNIX password:
Retype new UNIX password:
```



# 其它命令

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

■ **man: 查询参考手册信息**

```
name@name-VirtualBox:~$ man ls
LS(1)                                         User Commands                                         LS(1)

NAME
    ls - list directory contents

SYNOPSIS
    ls [OPTION]... [FILE]...

DESCRIPTION
    List information about the FILEs (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.
    Mandatory arguments to long options are mandatory for short options too.

    -a, --all
        do not ignore entries starting with .

    -A, --almost-all
        do not list implied . and ..

    --author
        with -l, print the author of each file

    -b, --escape
        print C-style escapes for nongraphic characters

    --block-size=SIZE
        scale sizes by SIZE before printing them; e.g., '--block-size=M' prints sizes in units of 1,048,576 bytes; see SIZE format below

    -B, --ignore-backups
        do not list implied entries ending with -

    -c      with -lt: sort by, and show, ctime (time of last modification of file status information); with -l: show ctime and sort by name; otherwise:
            sort by ctime, newest first

    -C      list entries by columns

    --color=[WHEN]
        colorize the output; WHEN can be 'always' (default if omitted), 'auto', or 'never'; more info below

    -d, --directory
        list directories themselves, not their contents

    -D, --dired
        generate output designed for Emacs' dired mode

    -f      do not sort, enable -M, disable -ls --color

Manual page ls(1) line 1 (press h for help or q to quit)
```

# Ubuntu 下的软件安装与系统更新

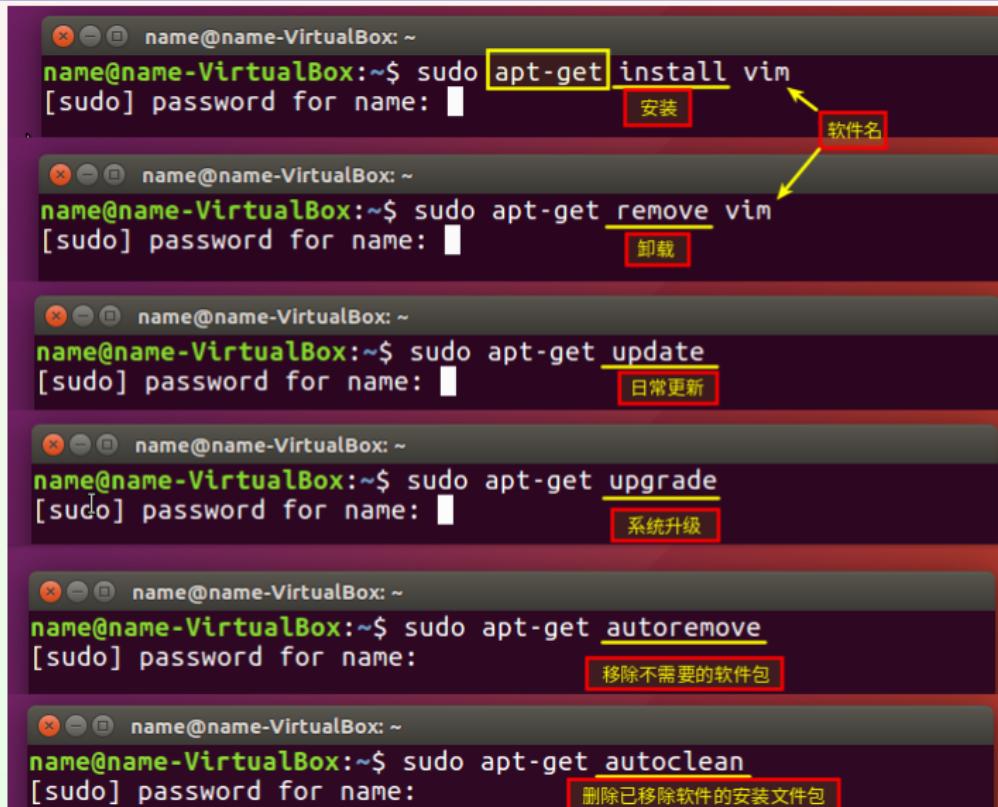
LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make



The screenshot shows a terminal window with six examples of apt-get commands, each annotated with a red box and a label:

- `name@name-VirtualBox:~$ sudo apt-get install vim` 安装 软件名
- `name@name-VirtualBox:~$ sudo apt-get remove vim` 卸载
- `name@name-VirtualBox:~$ sudo apt-get update` 日常更新
- `name@name-VirtualBox:~$ sudo apt-get upgrade` 系统升级
- `name@name-VirtualBox:~$ sudo apt-get autoremove` 移除不需要的软件包
- `name@name-VirtualBox:~$ sudo apt-get autoclean` 删除已移除软件的安装文件包



# vi 简介

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

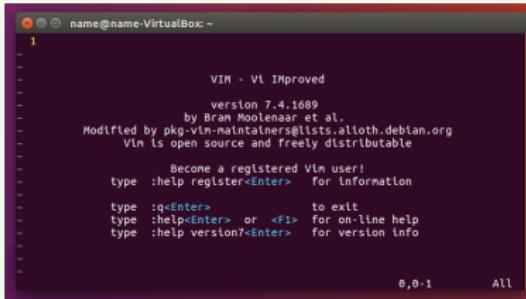
文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

vi 命令是 UNIX 操作系统和类 UNIX 操作系统中最通用的全屏幕纯文本编辑器

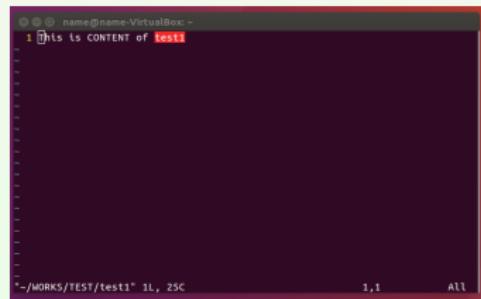
Linux 中的 vi 编辑器叫 vim，是 vi 的增强版 (vi Improved)，与 vi 编辑器完全兼容，而且实现了很多增强功能

Ubuntu 中 vim 的安装: [sudo apt-get install vim](#)



VIM - VI IMproved  
version 7.4.1689  
by Bram Moolenaar et al.  
Modified by pko-vim-maintainers@lists.alioth.debian.org  
Vim is open source and freely distributable  
  
Become a registered Vim user!  
type :help register<Enter> for information  
  
type :q<Enter> to exit  
type :help<Enter> or <F1> for on-line help  
type :help version7<Enter> for version info

(f) A new file



This is CONTENT of test1  
--/WORKS/TEST/test1" 1L, 25C

(g) An existing file

# vi 的三种工作方式

LINUX 命令、vim 使用与 GNU 软件编译

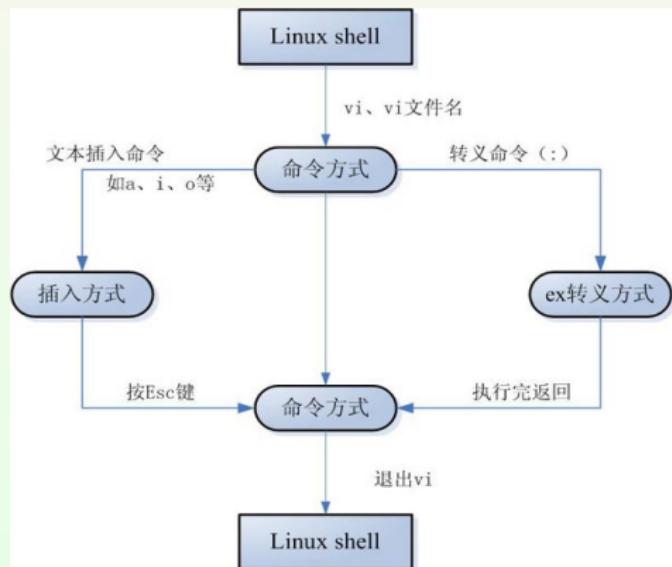
Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

- 命令方式 (默认方式)
- 插入方式
- 末行方式 (ex 转义方式)



# vi 的插入方式

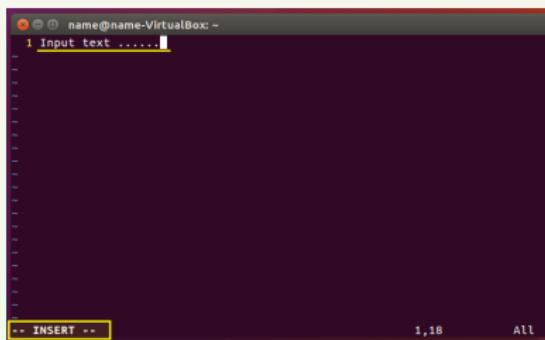
LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make



## 1 默认命令方式下按 “A”、“a”、“O”、“o”、“I”、“i” 键进入插入模式

- **a**: 当前字符后添加文本
- **A**: 在行末添加文本
- **i**: 当前字符前插入文本
- **I**: 在行首插入文本
- **o**: 当前行后面插入一空行
- **O**: 当前行前面插入一空行
- **r**: 替换光标位置的字符
- **R**: 替换光标开始的字符

## 2 按 “**Esc**” 键退出插入模式

# vi 的命令方式

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

- **h**: 光标向左移动一个字符      **i**: 光标向右移动一个字符
- **j**: 光标移动至下一行      **k**: 光标移动到上一行
- **b**: 光标向左移动一个词      **w**: 光标向右移动一个词
- **G**: 光标移动到最后一行      **gg**: 光标移动到第一行
- **\$**: 光标移动本行末      ^: 光标移动到本行第一个字符处
- **x**: 删除光标位置的字符      **dd**: 删除光标所在行
- **D**: 删除光标位置到行末      **d0**: 删除光标所在位置到行首
- **dG**: 删除光标位置到文件尾      **dw**: 删除光标所在位置的单词
- **u**: 撤销上一次操作      .: 重复上一次操作
- **yy**: 复制当前行到编辑缓冲区
- **pp**: 将编辑缓冲区内容 copy 到光标下一行

# vi 的命令方式

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

- **Ctrl+u**: 屏幕向上滚动半屏      **Ctrl+d**: 屏幕向下滚动半屏
- **Ctrl+b**: 屏幕向上滚动一屏      **Ctrl+f**: 屏幕向下滚动一屏
- **Ctrl+g**: 显示当前编辑文本信息
- **ndd**: 删除光标所在行起的**n**行字符
- **nyy**: 复制光标所在行起的**n**行字符
- **:n**: 光标跳转至第**n**行行首
- **:\$**: 光标跳转至末行行首
- **ZZ**: 必要时存盘并退出 vi
- **v**: VISUAL 状态
- **V**: VISUAL LINE 状态
- **V**: VISUAL LINE 状态
- **Ctrl+v**: VISUAL BLOCK 状态

# vi 的末行方式

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

- `/exp`: 向前搜索字符串`exp`    `?exp`: 向回搜索字符串`exp`  
`n`: 重复前一搜索命令 (下一个)  
`N`: 重复前一搜索命令 (前一个)
- `:w`: 执行存盘操作                      `:w file`: 当前编辑内容写入文件`file`
- `:w! file`: 当前编辑内容强制写到文件`file`
- `:q`: 退出 vi                      `:q!`: 强制退出 vi, 放弃编辑修改内容
- `:wq`: 存盘并退出 vi
- `:x`: 必要时存盘并退出 vi
- `:s/old/new`: 将光标所在行的第一个字符串`old`替换为`new`
- `:s/old/new/g`: 将光标所在行的所有的字符串`old`替换为`new`

# vi 的末行方式

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

- `:n1,n2 s/old/new/g`: 将第n1行到n2行的全部字符串old替换为new
- `:%s/old/new`: 将所有行的第一个字符串old替换为new
- `:%s/old/new/g`: 将所有行的全部字符串old替换为new
- `:set nu`: 显示文本的行号    `:set nonu`: 不显示文本的行号
- `:set list`: 显示不可见字符    `:set nolist`: 隐藏不可见字符
- `:set all`: 显示全部环境设置
- `:!cmd`: 在 vi 中执行 shell 命令cmd

# vim 的标签窗口

LINUX 命令、vim 使用与 GNU 软件编译

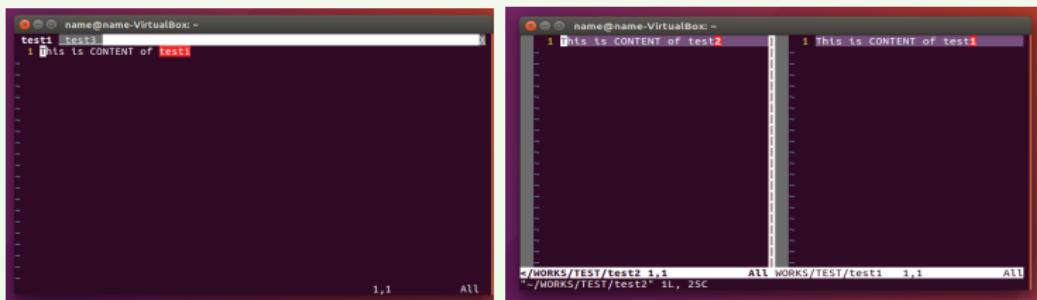
Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

- **:tabnew file**: 在 vim 新标签窗口中打开一个新的编辑文件file按g+t在标签窗口间切换
- **:vertical diffsplit file**: 对比当前文件和新文件file的差别按Ctrl+w在标签窗口间切换



- **:qall**: 退出所有的标签窗口
- **:qall!**: 强制退出所有的标签窗口

# vim 的环境变量配置文件 ~/.vimrc

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

```
name@name-VirtualBox: ~
5
6 " allow backspacing over everything in insert mode
7 set backspace=indent,eol,start
8
9 if has("vms")
10 set nobackup           " do not keep a backup file, use versions instead
11 else
12 set backup            " keep a backup file
13 endif
14 set history=50         " keep 50 lines of command line history
15 set ruler              " show the cursor position all the time
16 set showcmd             " display incomplete commands
17 set incsearch            " do incremental searching
18 set spell
19 set nu
20 set encoding=utf-8 fileencodings=utf-8,gbk,cp936,default,latin1
21
22 filetype plugin indent on
23 set completeopt=longest,menu
24
25 let g:SuperTabRetainCompletionType=2
".vimrc" 92 lines --5%--
```

5,0-1

5%

# vim 的键盘分布图

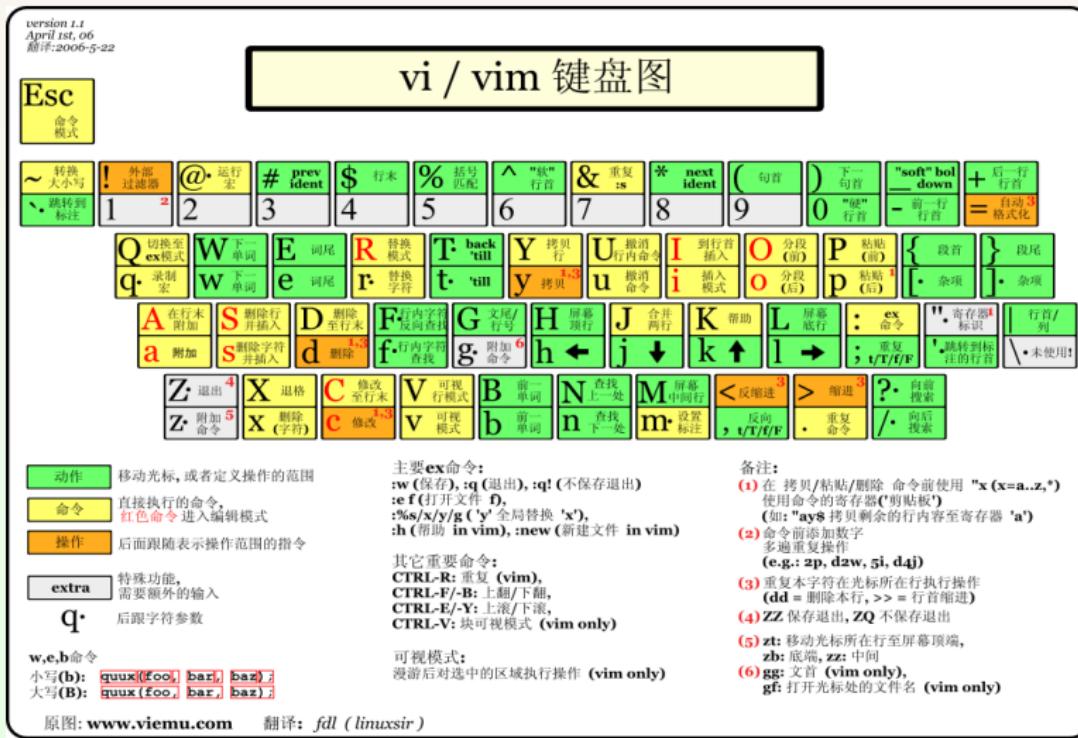
LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make



# 第一次使用 vim

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

**POV:**  
*You opened Vim for the first time*



# 软件编译的一般过程

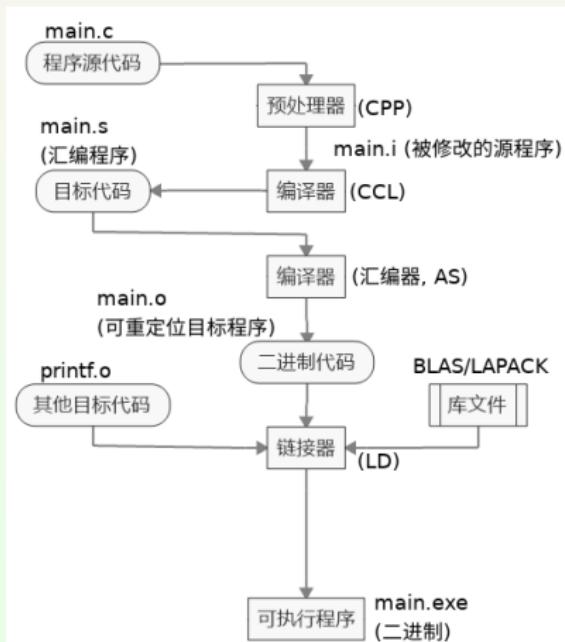
LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make



# 软件编译的一般过程

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

- 预处理 (Pre-Processing): 将根据源代码中以字符 # 开头的指令要求, 用预处理器 cpp 修改原始程序

如 C 程序中 `#include <stdio.h>` 指令告诉预处理器读系统头文件 stdio.h 的内容, 并把它直接插入到程序文本中去, 得到另外一个 C 程序

- 编译 (Compiling): 在编译阶段, 编译器 compiler 会检查代码的规范性

- 1 检查源代码是否存在拼写和语法错误等, 检查无误后, 编译器将把代码翻译成汇编语言
- 2 确定代码实际执行的具体任务和指令

汇编语言为不同高级语言不同编译器提供了通用的语言。比如 C 编译器和 Fortran 编译器产生的输出文件用的都是一样的汇编语言

用户可以使用-S 选项来进行查看, 该选项只完成编译过程而不进行汇编

# 软件编译的一般过程

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

- **汇编 (Assembling):** 汇编阶段是把编译阶段生成的.s 文件 (汇编代码) 转成目标文件  
用户可使用-c 选项就可看到汇编代码将转化为.o 的二进制目标代码
- **链接 (Linking):** 汇编生成二进制代码之后, 就进入了链接阶段  
完成了链接之后, 编译器就可以生成最终可执行文件  
函数库一般分为静态库 (后缀名一般为.a) 和动态库 (后缀名一般为.so) 两种
- **静态库**是指编译链接时, 把库文件的代码全部加入到可执行文件中, 因此生成的文件比较大, 但在运行时就不再需要库文件支持
- **动态库**在编译链接时并没有把库文件的代码加入到可执行文件中, 而在程序执行时由运行时链接文件加载库, 这样可以节省系统的开销

# Linux 的软件编译: Make

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

**make**是一个命令工具，它通过解释 Makefile 文件中的指令，完成对软件的编译、链接和执行

- 在 Linux 环境下使用 GNU 软件，**make**是重要的编译命令和工具，能够帮助用户比较容易地构建一个属于自己的软件工程
- 理解**make**方式安装软件的核心就是学会 Makefile 文件的编写。Makefile 文件是 **make** 正常工作的基础。在 Makefile 文件中描述了整个工程所有文件的编译顺序、编译规则

使用**make**工具编译软件时，有几种文件在执行**make**时会被编译或重新编译

- 1 所有源文件没有被编译过的，将对各源文件进行编译并进行链接，生成最后的可执行程序
- 2 每个上次执行**make**后修改过的源代码文件在本次执行**make**时会被重新编译
- 3 如果头文件在上次执行**make**后被修改，则所有包含此头文件的源文件在本次执**make**时将会被重新编译

# Makefile 文件的编写规则

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

一个简单的 Makefile 描述规则的组成格式：

TARGET … : PREREQUISITES …

COMMAND

…

…

## ■ target: 规则的目标

目标通常是最后需要生成的文件名或者为了实现编译过程中必要的中间过程文件名，可以是.o 文件、也可以是最后的可执行程序的文件名等

目标也可以是一个 make 执行的动作的名称

如目标 clean 就不是一个文件，仅表示执行一个动作的标识，这样的目标也称为“伪目标”

# Makefile 文件的编写规则

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

- **prerequisites**: 规则的依赖

生成规则目标所需要的文件名列表，通常一个目标依赖于一个或者多个文件

- **command**: 规则的命令行

规则所要执行的动作 (可以是任意的 shell 命令或者是可在 shell 下执行的程序)

命令行限定了**make**执行该规则时所需要的动作

一个 Makefile 中的每个规则可以包含多个命令行，每条命令占一行

**注意**: 每一个命令行必须以 [Tab] 字符开始

[Tab] 字符告诉**make**此行是命令行，**make**按命令完成相应的动作

# 通过**make** 运行 Makefile

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

- 对于一般用户来说，**make**命令像命令行参数一样接收目标
- 当**make**命令第一次执行时，它扫描 Makefile 找到目标以及其依赖 (如果这些依赖自身也是目标，继续为这些依赖扫描 Makefile 建立其依赖关系，然后编译)
- 一旦主依赖编译之后，然后编译主目标 (通过 **make**命令传入) 默认情况下，**make**执行的是 Makefile 中的第一个规则，此规则的第一个目标称之为“最终目的”或者“终极目标”(就是一个 Makefile 最终需要更新或者创建的目标)
- 假设用户对某个源文件进行了修改，再次执行 **make**命令，将只编译与该源文件相关的目标文件

采用**make**编译并获得最终的可执行文件，有可能节省大量的时间

# make 执行举例

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

假定程序的源代码为如下结构

```
$ ls  
anotherTest.c Makefile test.c test.h
```

其中 Makefile 的内容如下

```
all: test  
  
test: test.o anotherTest.o  
        gcc -Wall test.o anotherTest.o -o test  
  
test.o: test.c  
        gcc -c -Wall test.c  
  
anotherTest.o: anotherTest.c  
        gcc -c -Wall anotherTest.c  
  
clean:  
        rm -rf *.o test
```

# make 执行举例

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

## 初次执行make命令

```
$ make
gcc -c -Wall test.c
gcc -c -Wall anotherTest.c
gcc -Wall test.o anotherTest.o -o test
```

再次查看该目录下的内容，将会发现里面多了一些 .o 文件和可执行文件test

```
$ ls
anotherTest.c anotherTest.o Makefile test test.c test.h test.o
```

# make 执行举例

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

如果用户对源代码 test.c 作一些改动，再次执行 make 命令

```
$ make
gcc -c -Wall test.c
gcc -Wall test.o anotherTest.o -o test
```

可以看到，只有 test.o 被重新编译，而另一个 anotherTest.o 并未重新编译

最后，用户如果需要清理所有的目标文件和可执行文件 test，可以使用目标 clean：

```
$ make clean
rm -rf *.o test

$ ls
anotherTest.c Makefile test.c test.h
```

可以看到所有的.o 文件和可执行文件 test 都已被删除

# make 的常用可选参数

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

为适应编译过程中的复杂需要，make 工具还提供了一些选项，以满足用户在编译时的不同需求。这里介绍几个常见的可选项

- 默认的make命令不会编译那些自从上次编译之后就没有更改的文件，但如果想覆盖make这种默认行为，可以使用 -B 选项

```
$ make
make: Nothing to be done for `all'.
```

```
$ make -B
gcc -c -Wall test.c
gcc -c -Wall anotherTest.c
gcc -Wall test.o anotherTest.o -o test
```

# make 的常用可选参数

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

- 如果想将了解 make 执行时实际做了什么，可以使用 -d 选项

```
$ make -d | more
GNU Make 3.81
Copyright (C) 2006 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.
There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A
PARTICULAR PURPOSE.

This program built for x86_64-pc-linux-gnu
Reading makefiles...
Reading makefile `Makefile'...
Updating makefiles...
Considering target file `Makefile'.
Looking for an implicit rule for `Makefile`.
Trying pattern rule with stem `Makefile'.
Trying implicit prerequisite `Makefile.o'.
Trying pattern rule with stem `Makefile'.
Trying implicit prerequisite `Makefile.c'.
Trying pattern rule with stem `Makefile'.
Trying implicit prerequisite `Makefile.cc'.
Trying pattern rule with stem `Makefile'.
Trying implicit prerequisite `Makefile.C'.
Trying pattern rule with stem `Makefile'.
Trying implicit prerequisite `Makefile.cpp'.
Trying pattern rule with stem `Makefile'.
--More--
```

# make 的常用可选参数

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

- 如果想让 **make** 在查找 Makefile 时切换到指定目录，可以使用 -C 选项

假设当前目录为

```
$ ls
file file2 frnd frnd1.cpp log1.txt log3.txt log5.txt
file1 file name with spaces frnd1 frnd.cpp log2.txt log4.txt
```

**make** 命令所执行的 Makefile 文件位于 .../make-dir/ 目录下

```
$ make -C ../make-dir/
make: Entering directory `/home/himanshu/practice/make-dir'
make: Nothing to be done for `all'.
make: Leaving directory `/home/himanshu/practice/make-dir'
```

**make** 命令将先切到指定目录，执行完毕后再切换回当前目录 ↵ ↵

# make 的常用可选参数

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

- 如果想将 Makefile 文件重命名 (这里取名为 my\_makefile, 也可是其它名), 为了让 make 仍将它也当成 Makefile, 可以使用 -f 选项

```
make -f my_makefile
```

- -h/-help 显示帮助信息
- -i 在执行时忽略所有的错误
- -j [jobnumber] 指定同时运行命令的个数  
如果 -j 后没有 jobsnum 参数、make 运行命令时能运行多少就运行多少

make 的可选参数还有很多, 有关各选项的具体内容可运行命令  
**man make** 查看

# Makefile 文件详解示例

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

```

CC := gcc
CFLAGS := -Wall -O3
Libs = -lpthread
Target := phone
Source := $(wildcard ./*.c)    ① 列出当前目录下所有.c源文件
Objs := $(patsubst %.c,%o,$(Source))  ② 把字符串Source中以.c结尾的单词替换成以.o结尾的字符串
Modules += allfree create delete display init login menu search main
AllObjs := $(addsuffix .o, $(Modules))  ③ 将.o添加到Modules所代表的所有文件名后面
                                         ④ 生成字符串: allfree/.o create/.o等

1 include scripts/Makefile          根目录Makefile
5 modules_make = $(MAKE) -C $(1);   ⑤ C进入目录, 执行对应的Makefile
modules_clean = $(MAKE) clean -C $(1);

.PHONY: all mm mc clean

2 all: $(Target)                   ⑥ 将目标名作为参数传递给modules_make

4 mm:                            ⑦ 遍历右边所有的命令
@ $(foreach n,$(Modules)) $(call modules_make,$(n))
mc:                            ⑧ 将目标名作为参数传递给modules_clean
@ $(foreach n,$(Modules)) $(call modules_clean,$(n))

3 $(Target) : mm                  ⑨ 打印: phone make done!
$(CC) $(CFLAGS) -o $(Target) $(AllObjs) $(Libs)
@ echo $(Target) make done!

clean : mc
rm -rf $(Target)
@ echo clean done!

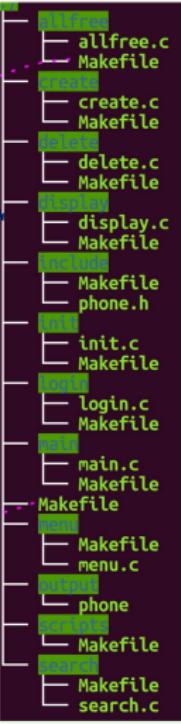
gcc -Wall -O3 -c phone allfree/* o create/* o delete/* o display/* o init/* o login/* o menu/* o search/* o main/* o pthread

```

scripts/Makefile

include .. /scripts/Makefile

all : \$(Objs) ⑥  
clean :  
rm -rf \$(Objs)



# make 的执行过程

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

- 依次读取变量 \${MAKEFILES} 定义的 makefile 文件列表
- 读取工作目录下的 makefile 文件  
根据命名的查找默认顺序 GNUmakefile, makefile, Makefile, 首先找到哪个就读取哪个
- 依次读取工作目录 makefile 文件中使用指示符 include 包含的文件
- 查找重建所有已读取的 makefile 文件的规则  
如果存在一个目标是当前读取的某一个 makefile 文件，则执行此规则重建此 makefile 文件，完成后从第一步开始重新执行
- 初始化变量值并展开那些需要立即展开的变量和函数并根据预设条件确定执行分支
- 根据“终极目标”以及其他目标的依赖关系建立依赖关系链表

# make 的执行过程 (cont.)

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器  
vi/vim 简介

Linux 下的软件安装与  
Make

- 执行除“终极目标”以外的所有目标的规则  
规则中如果依赖文件中任一个文件的时间戳比目标文件新，则使用规则所定义的命令重建目标文件
- 最后执行“终极目标”所在的规则

## 总结

make 命令执行一个规则时的依据：

比较目标文件和所有的依赖文件的时间戳

- 如果目标的时间戳比所有依赖文件的时间戳没有更新（依赖文件在上一次 make 命令执行之后没有被修改），那什么也不做；
- 否则（即依赖文件中的某一个或者全部在上一次 make 命令执行后已经被修改过），规则定义的重建目标的命令将会被执行

这是 make 命令工作的基础，也是其执行规则所定义命令的依据

# 卖油翁：无他 但手熟尔

LINUX 命令、vim 使用与 GNU 软件编译

Linux 操作系统概述

Linux 命令简介

文本编辑器 vi/vim 简介

Linux 下的软件安装与 Make

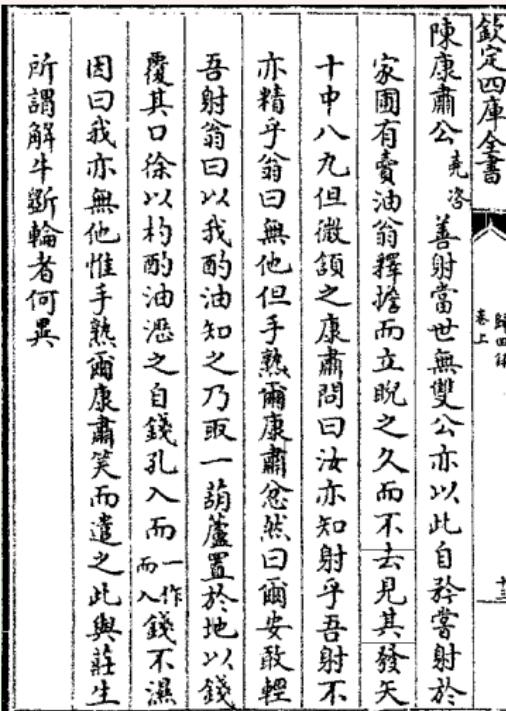


Fig.: 欧阳修 (1007-1072) 《欧阳文忠公文集·归田录》卷上

# 虚拟机

LINUX 命令、vim 使用与 GNU 软件编译

虚拟机与 Ubuntu 的安装

虚拟机 (Virtual Machine, VM) 是指通过软件模拟的具有完整硬件系统功能的、运行在一个完全隔离环境中的完整计算机系统

- 虚拟机是指由软件模拟出来的一台虚拟的计算机：逻辑上是一台计算机
- 可以在同一台物理计算机上运行多个虚拟机：每个虚拟机都有独立的操作系统

流行的的虚拟机软件有 VMware (VMWare ACE)、VirtualBox 和 Virtual PC 等

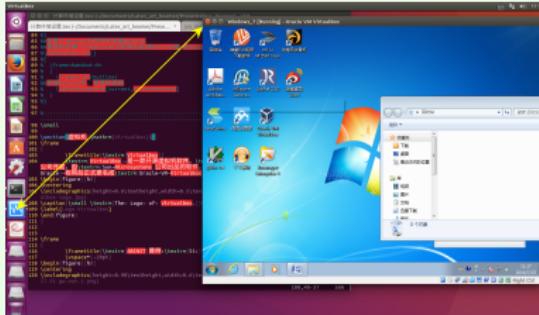


Fig.: The Virtualbox installed in Ubuntu.

# Virtualbox

LINUX 命令、vim 使用与 GNU 软件编译

虚拟机与  
Ubuntu 的安装



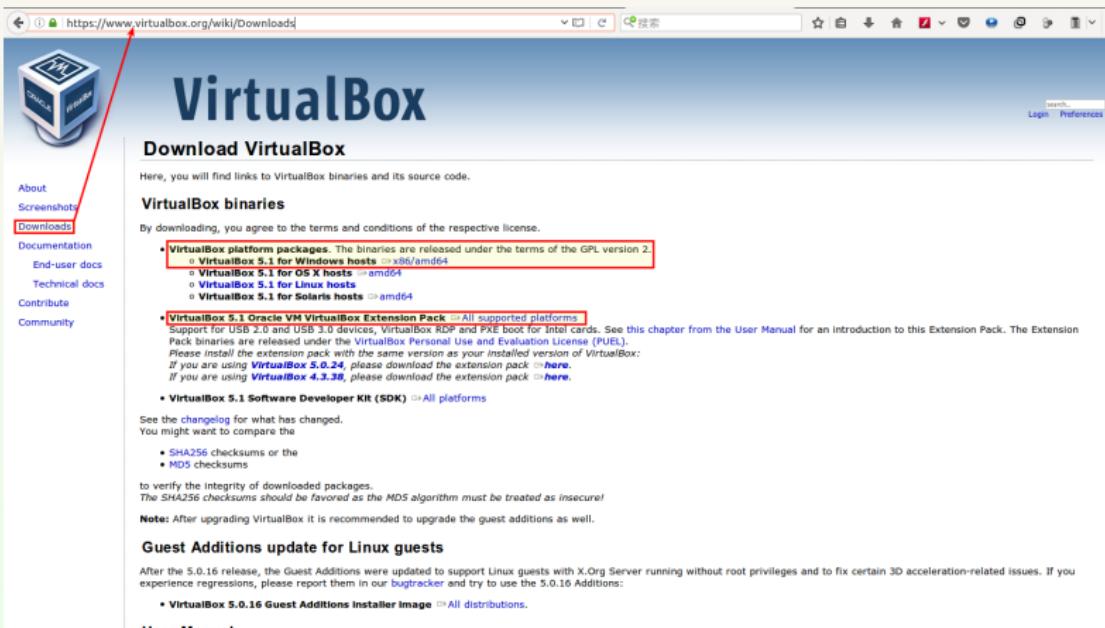
The screenshot shows the official website for Oracle VM VirtualBox. At the top, there's a navigation bar with links for "About", "Screenshots", "Downloads", "Documentation", "End-user docs", "Technical docs", "Contribute", and "Community". Below the navigation is a large blue banner with the text "VirtualBox" and "Welcome to VirtualBox.org!". To the left of the banner is a small icon of a blue cube with "VBOX" on it. The main content area contains text about the software's features and history, mentioning its use for enterprise and home users, its rich feature set, and its open-source nature under the GNU General Public License (GPL) version 2. It also highlights recent releases: "VirtualBox 5.1.2 released!", "VirtualBox 5.0.26 released!", and "VirtualBox 5.1 released!". A "News Flash" section at the bottom right lists "Important February, 2015" and "We're hiring! Looking for a new challenge? We're looking for generic product developers (Russia)". At the bottom, a large blue button invites users to "Download VirtualBox 5.1".

Fig.: The homepage of Virtualbox.

# Virtualbox 的下载

LINUX 命令、vim 使用与 GNU 软件编译

虚拟机与 Ubuntu 的安装



The screenshot shows the 'Download VirtualBox' page from the VirtualBox website. The URL in the address bar is <https://www.virtualbox.org/wiki/Downloads>. A red arrow points from the 'Downloads' link in the left sidebar to the 'VirtualBox platform packages' section of the main content area. This section lists several download links:

- VirtualBox 5.1 for Windows hosts (x86/amd64)
- VirtualBox 5.1 for OS X hosts (amd64)
- VirtualBox 5.1 for Linux hosts
- VirtualBox 5.1 for Solaris hosts (amd64)

Below this, another section is highlighted with a red box:

- VirtualBox 5.1 Oracle VM VirtualBox Extension Pack (All supported platforms)

Support for USB 2.0 and USB 3.0 devices, VirtualBox RDP and PXE boot for Intel cards. See this chapter from the User Manual for an introduction to this Extension Pack. The Extension Pack binaries are released under the VirtualBox Personal Use and Evaluation License (PUEL). Please install the extension pack with the same version as your installed version of VirtualBox:  
If you are using VirtualBox 5.0.24, please download the extension pack [here](#).  
If you are using VirtualBox 4.3.38, please download the extension pack [here](#).

At the bottom of the page, there is a note about guest additions and a link to the guest additions installer image.

Fig.: The url page for Virtualbox download.

# Windows 下安装 Virtualbox

LINUX 命令、vim 使用与 GNU 软件编译

虚拟机与 Ubuntu 的安装

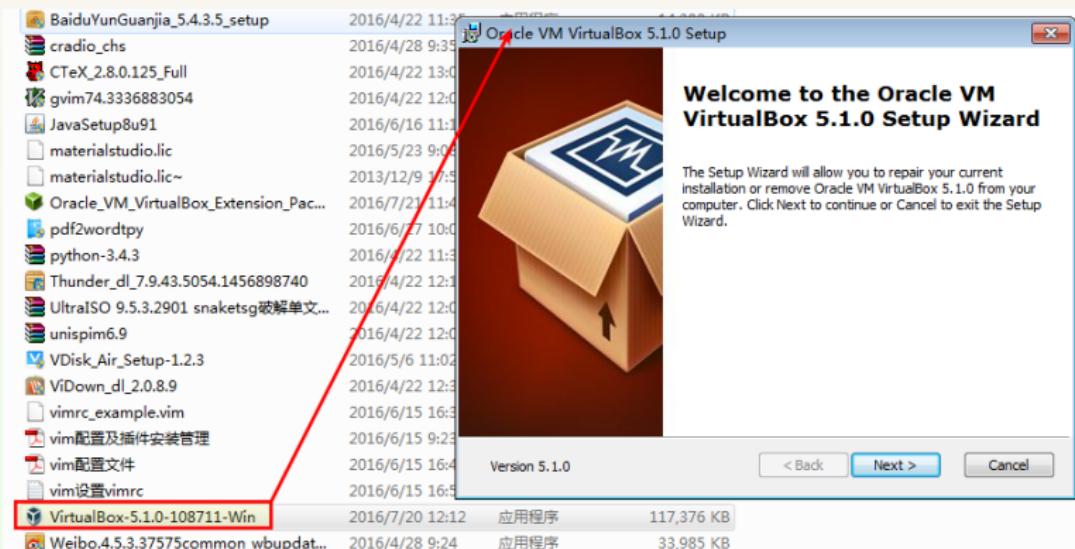


Fig.: The install step-1 of Virtualbox.

# Windows 下安装 Virtualbox

LINUX 命令、vim 使用与 GNU 软件编译

虚拟机与 Ubuntu 的安装

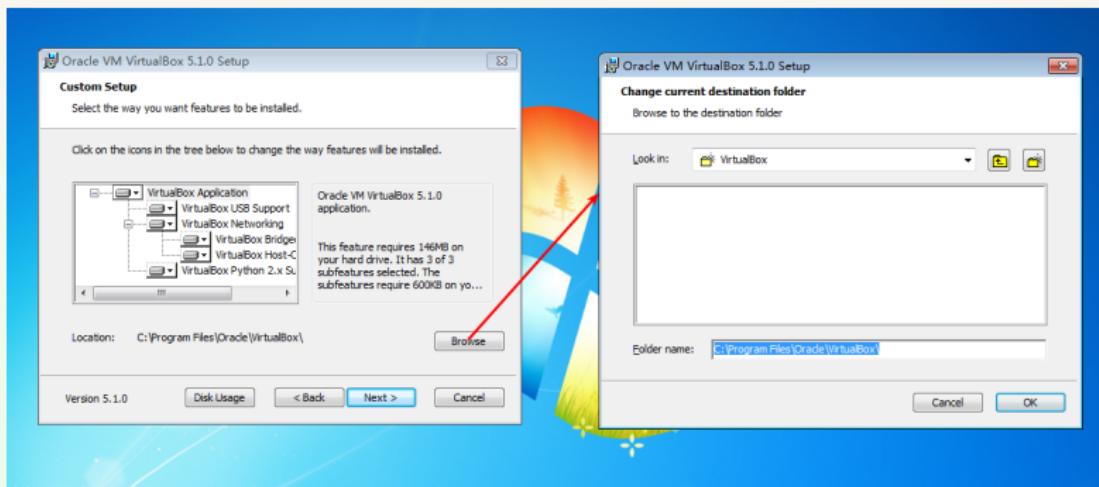


Fig.: The install step-2 of Virtualbox.

# Windows 下安装 Virtualbox

LINUX 命令、vim 使用与 GNU 软件编译

虚拟机与 Ubuntu 的安装

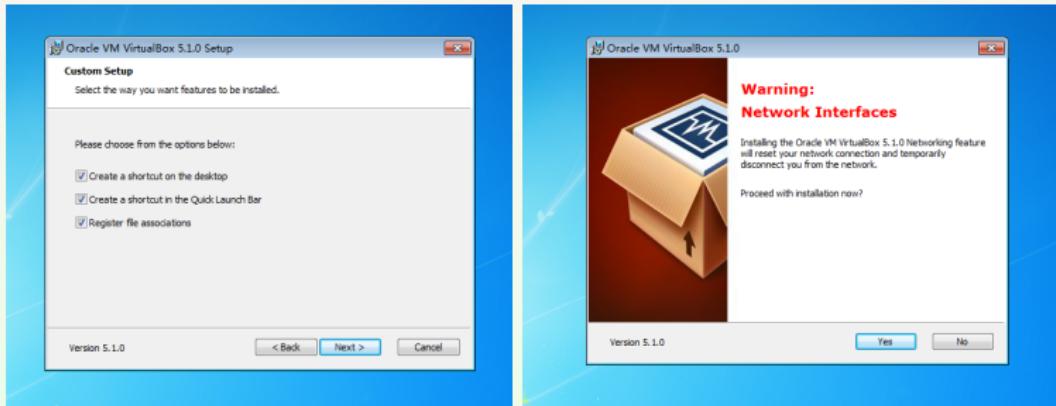


Fig.: The install step-3 of Virtualbox.

# Windows 下安装 Virtualbox

LINUX 命令、vim 使用与 GNU 软件编译

虚拟机与 Ubuntu 的安装

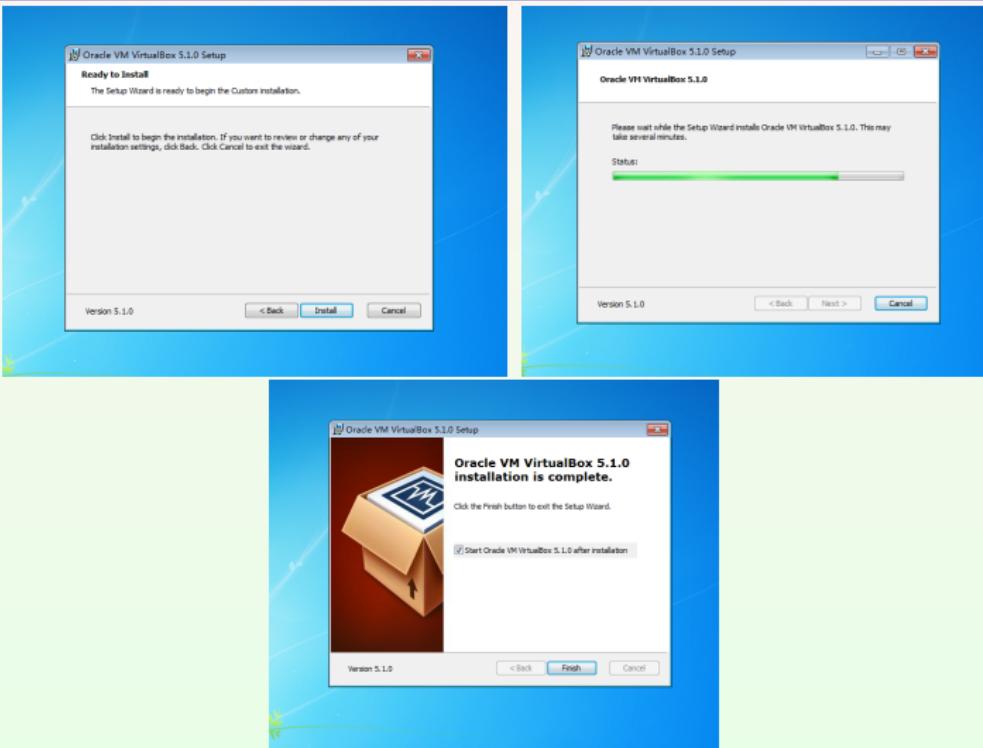


Fig.: The install step-4 of Virtualbox

# Windows 下 Virtualbox 安装的常见问题

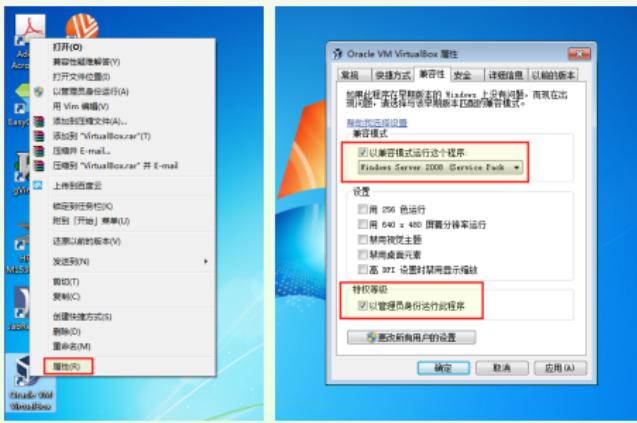
LINUX 命令、vim 使用  
与 GNU 软件编译

虚拟机与  
Ubuntu 的安  
装

## 常见错误



## 解决方案



# Virtualbox 安装扩展功能

LINUX 命令、vim 使用  
与 GNU 软件编译

虚拟机与  
Ubuntu 的安  
装

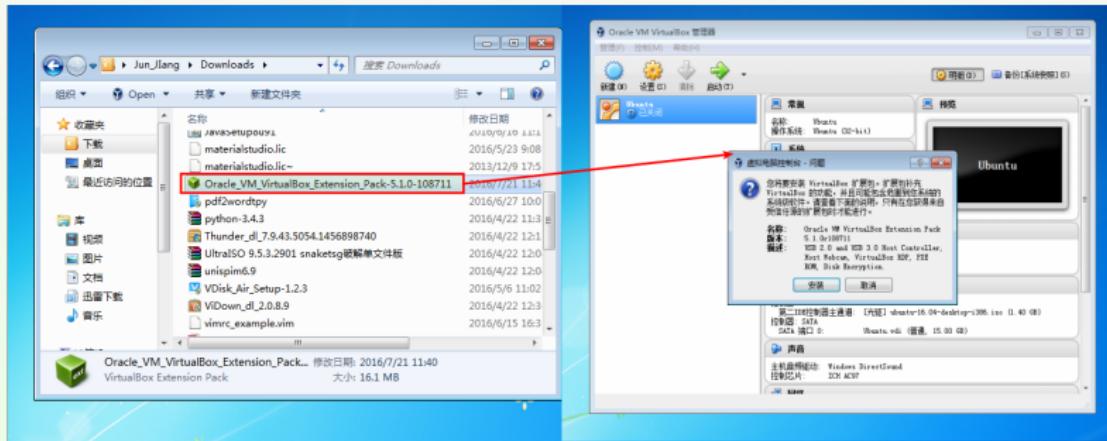


Fig.: The install step-5 of Virtualbox.

# Virtualbox 安装扩展功能

LINUX 命令、vim 使用与 GNU 软件编译

虚拟机与 Ubuntu 的安装

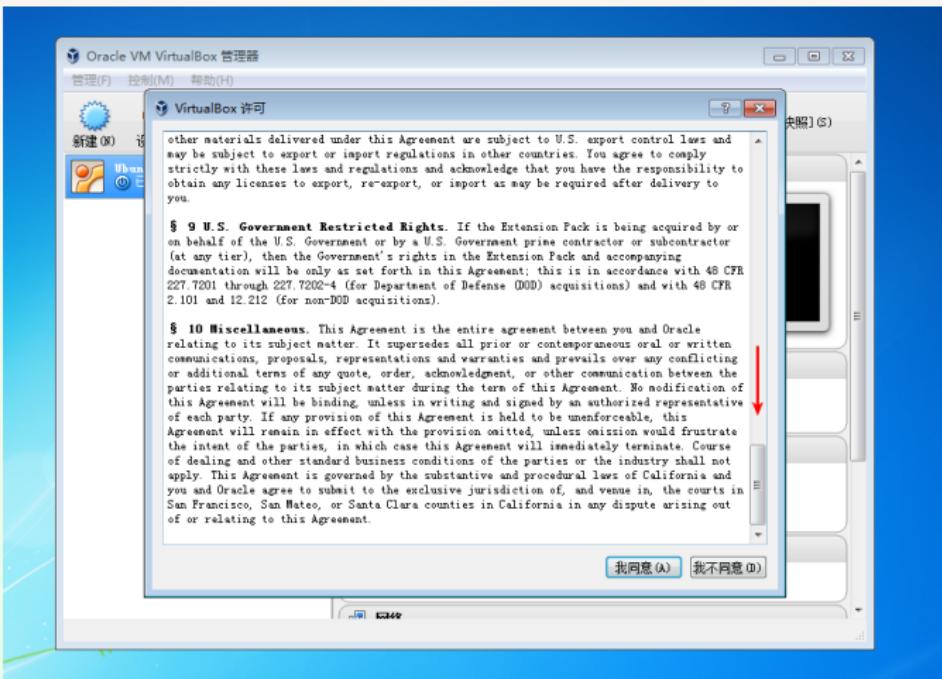


Fig.: The install step-6 of Virtualbox.

# Ubuntu 的安装

LINUX 命令、vim 使用与 GNU 软件编译

虚拟机与 Ubuntu 的安装

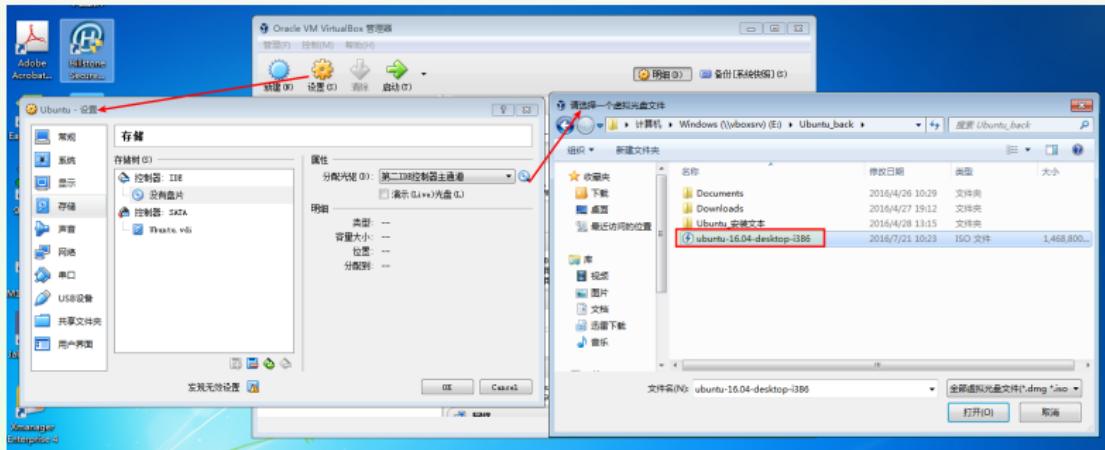


Fig.: The install step-1 of Ubuntu in Virtualbox.

# Ubuntu 的安装

LINUX 命令、vim 使用与 GNU 软件编译

虚拟机与 Ubuntu 的安装

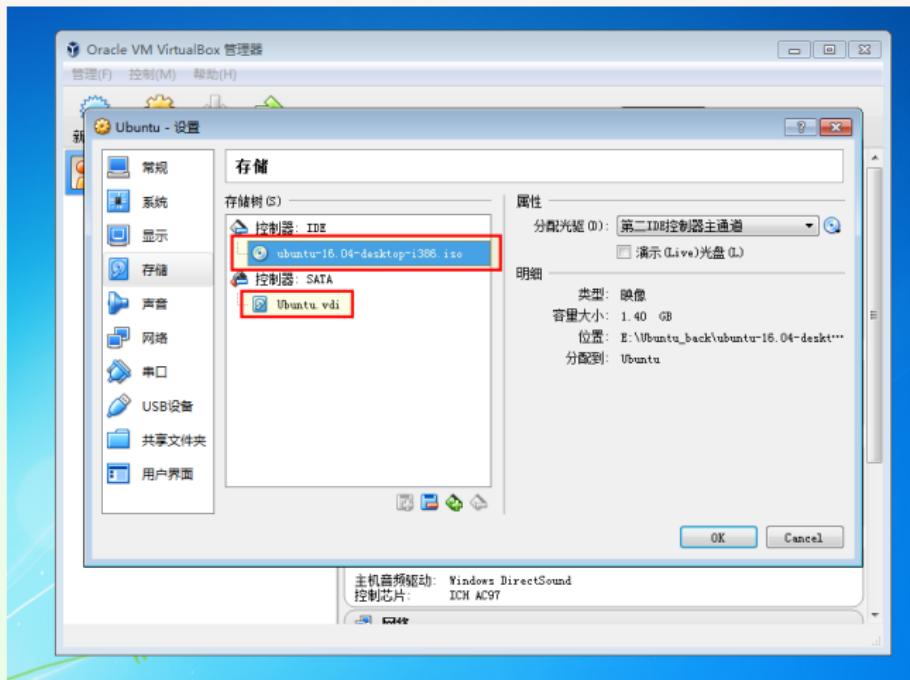


Fig.: The install step-2 of Ubuntu in Virtualbox.

# Ubuntu 的安装

LINUX 命令、vim 使用与 GNU 软件编译

虚拟机与 Ubuntu 的安装

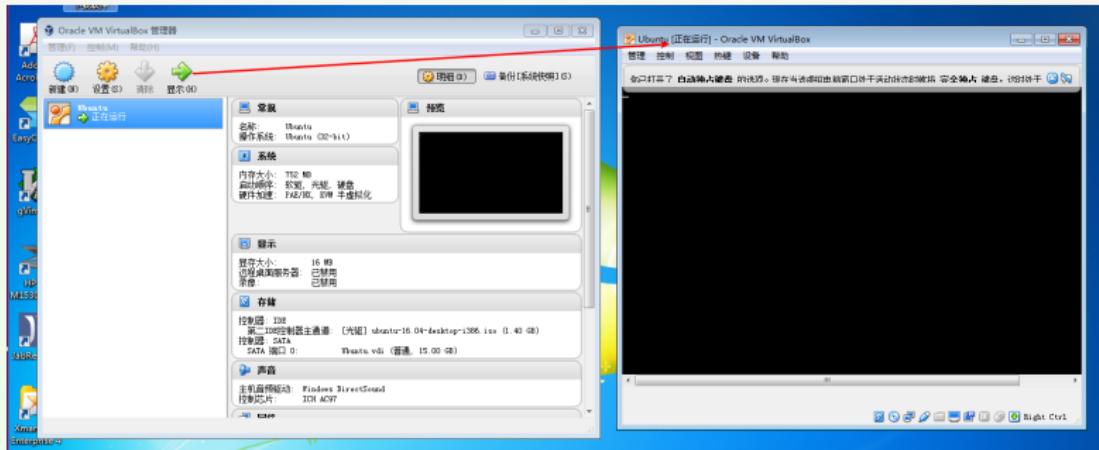
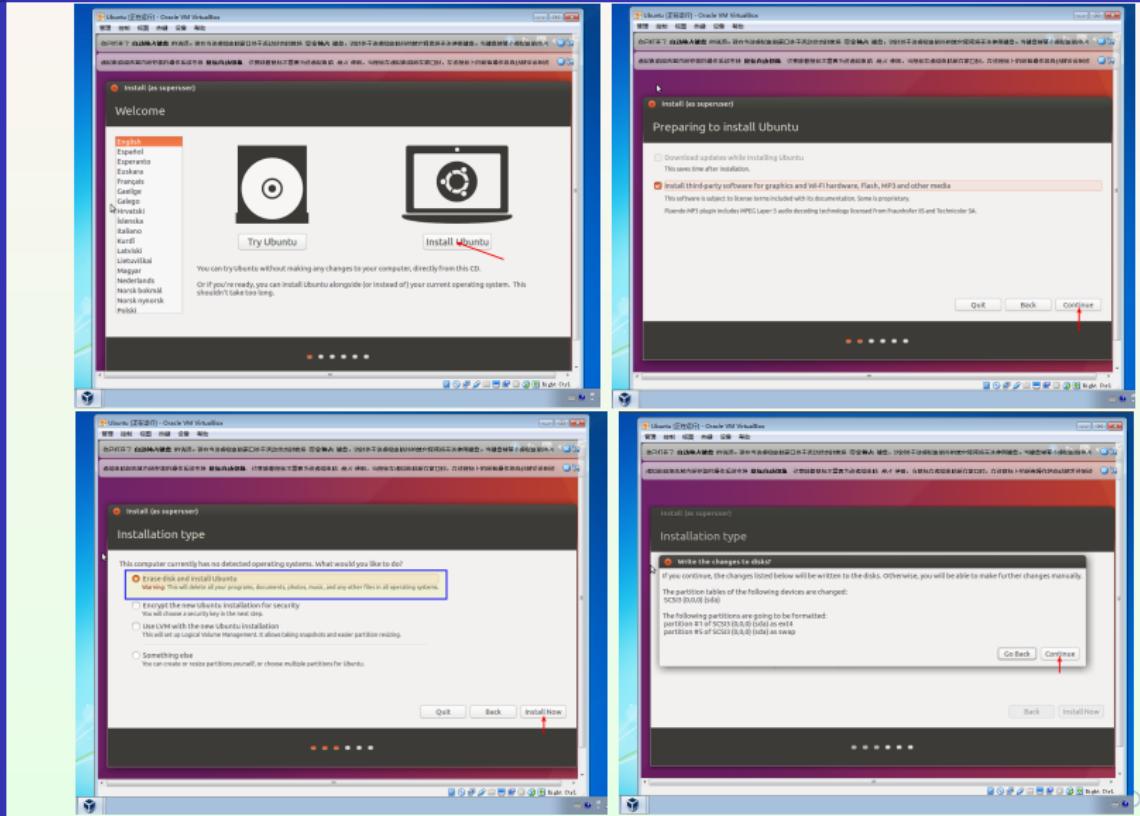


Fig.: The install step-3 of Ubuntu in Virtualbox.

# Ubuntu 的安装

LINUX 命令、vim 使用与 GNU 软件编译

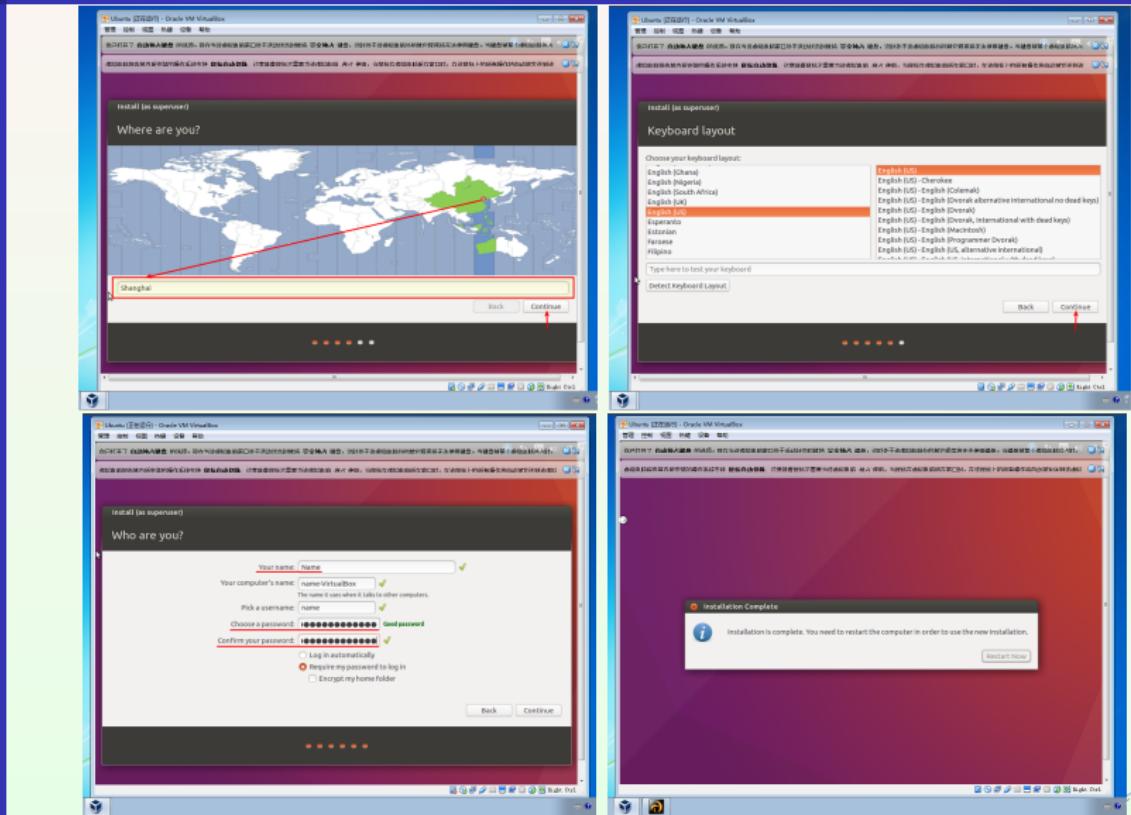
虚拟机与 Ubuntu 的安装



# Ubuntu 的安装

LINUX 命令、vim 使用与 GNU 软件编译

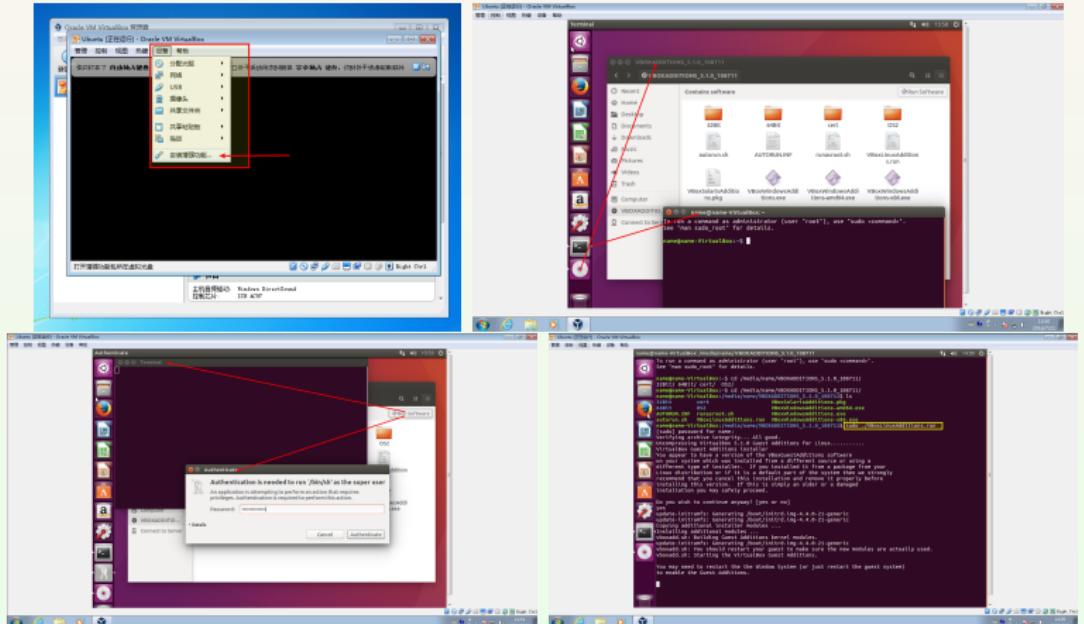
虚拟机与 Ubuntu 的安装



## 安装 Virtualbox 扩展支持



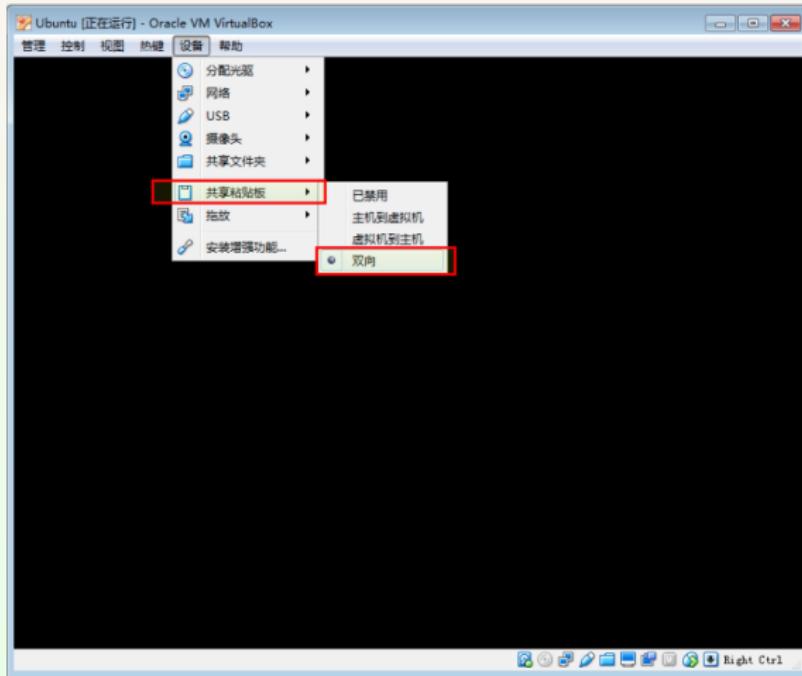
虚拟机与  
Ubuntu 的安  
装



# Virtualbox 设置共享

LINUX 命令、vim 使用与 GNU 软件编译

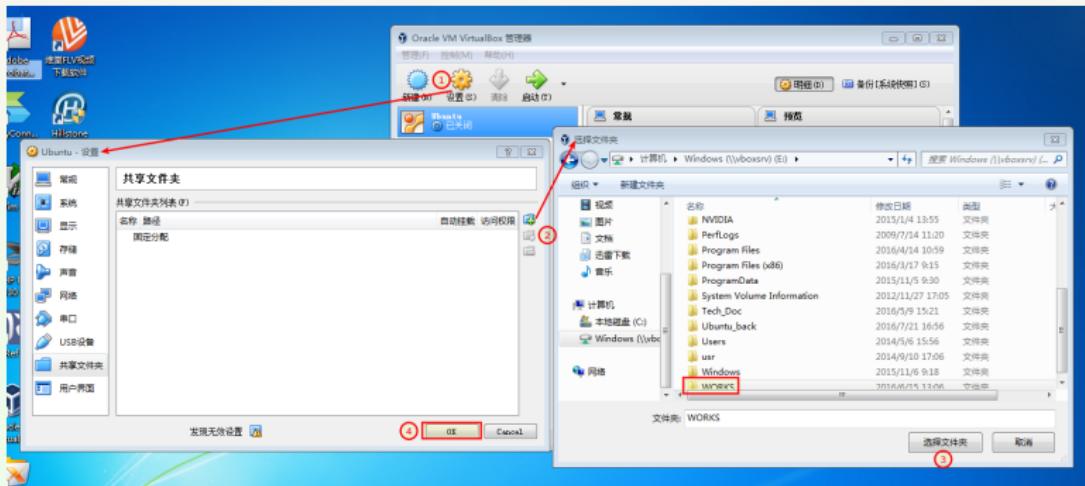
虚拟机与 Ubuntu 的安装



# 文件共享: Virtualbox 的设置

LINUX 命令、vim 使用与 GNU 软件编译

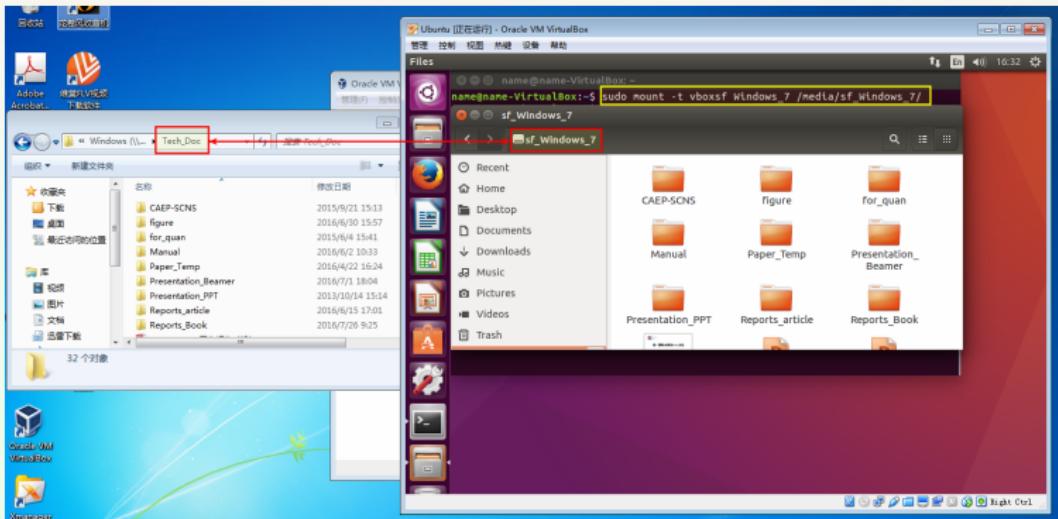
虚拟机与 Ubuntu 的安装



# 文件共享: Ubuntu 的设置

LINUX 命令、vim 使用与 GNU 软件编译

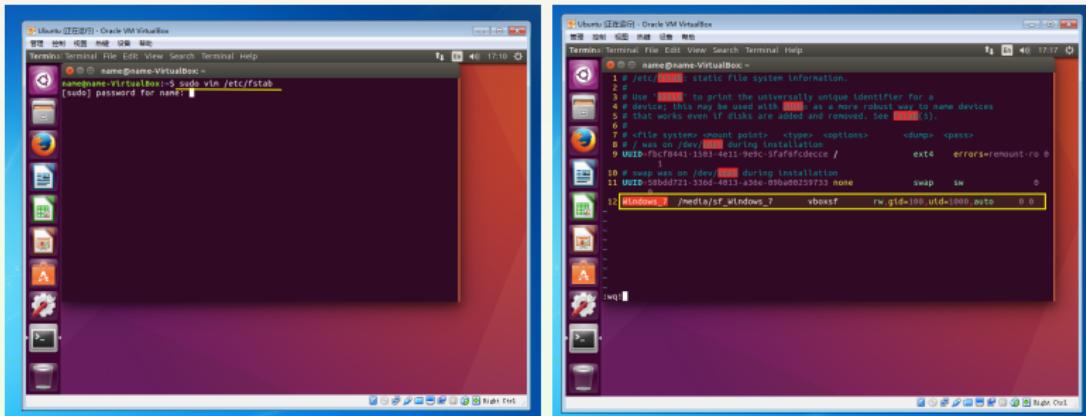
虚拟机与 Ubuntu 的安装



# 文件共享: Ubuntu 设置的自动挂载

LINUX 命令、vim 使用  
与 GNU 软件编译

虚拟机与  
Ubuntu 的安  
装



LINUX 命  
令、vim 使用  
与 GNU 软  
件编译

虚拟机与  
Ubuntu 的安  
装

# 谢谢大家！