



Spark编程实践

安装Spark

安装Spark之前需要

<https://ubuntu.com/download/desktop/thank-you?version=22.04.2&architecture=amd64#download>

CANONICAL

ubuntu®

Enterprise ▾

Developer ▾

Community ▾

Download ▴

We are hiring

Products ▾

Search 🔍

Sign in

Ubuntu Desktop ▸

Download Ubuntu desktop and replace your current operating system whether it's Windows or Mac OS, or, run Ubuntu alongside it.

22.04 LTS

22.10

Ubuntu Server ▸

The most popular server Linux in the cloud and data centre, you can rely on Ubuntu Server and its five years of guaranteed free upgrades.

Get Ubuntu Server

Mac and Windows

ARM

IBM Power

s390x

Ubuntu for IoT ▸

Are you a developer who wants to try snappy Ubuntu Core or classic Ubuntu on an IoT board?

Raspberry Pi

Intel IoT platforms

Intel NUC

KVM

Qualcomm Dragonboard 410c

Intel IEL TANK 870

AMD-Xilinx Evaluation kits & SOMs

RISC-V platforms

Ubuntu Cloud ▸

Use Ubuntu optimised and certified server images on most major clouds.

Get started on Amazon AWS, Microsoft Azure, Google Cloud Platform and more...

Download cloud images for local development and testing

TUTORIALS

If you are already running Ubuntu - you can [upgrade](#) with the Software Updater

Burn a DVD on [Ubuntu](#), [macOS](#), or [Windows](#). Create a bootable USB stick on [Ubuntu](#), [macOS](#), or [Windows](#)

Installation guides for [Ubuntu Desktop](#) and [Ubuntu Server](#)

You can learn how to [try Ubuntu before you install](#)

READ THE DOCS

Read the official docs for [Ubuntu Desktop](#), [Ubuntu Server](#), and [Ubuntu Core](#)

UBUNTU APPLIANCES

An [Ubuntu Appliance](#) is an official system image which blends a single application with Ubuntu Core. Certified to run on Raspberry Pi and PC boards.

OTHER WAYS TO DOWNLOAD

Ubuntu is available via [BitTorrents](#) and via a [minimal network installer](#) that allows you to customise what is installed, such as additional languages. You can also find [older releases](#).

UBUNTU FLAVOURS

Find new ways to experience Ubuntu, each with their own choice of default applications and settings.

[Kubuntu](#)

[Lubuntu](#)

[Ubuntu Budgie](#)

[Ubuntu Kylin](#)

[Ubuntu MATE](#)

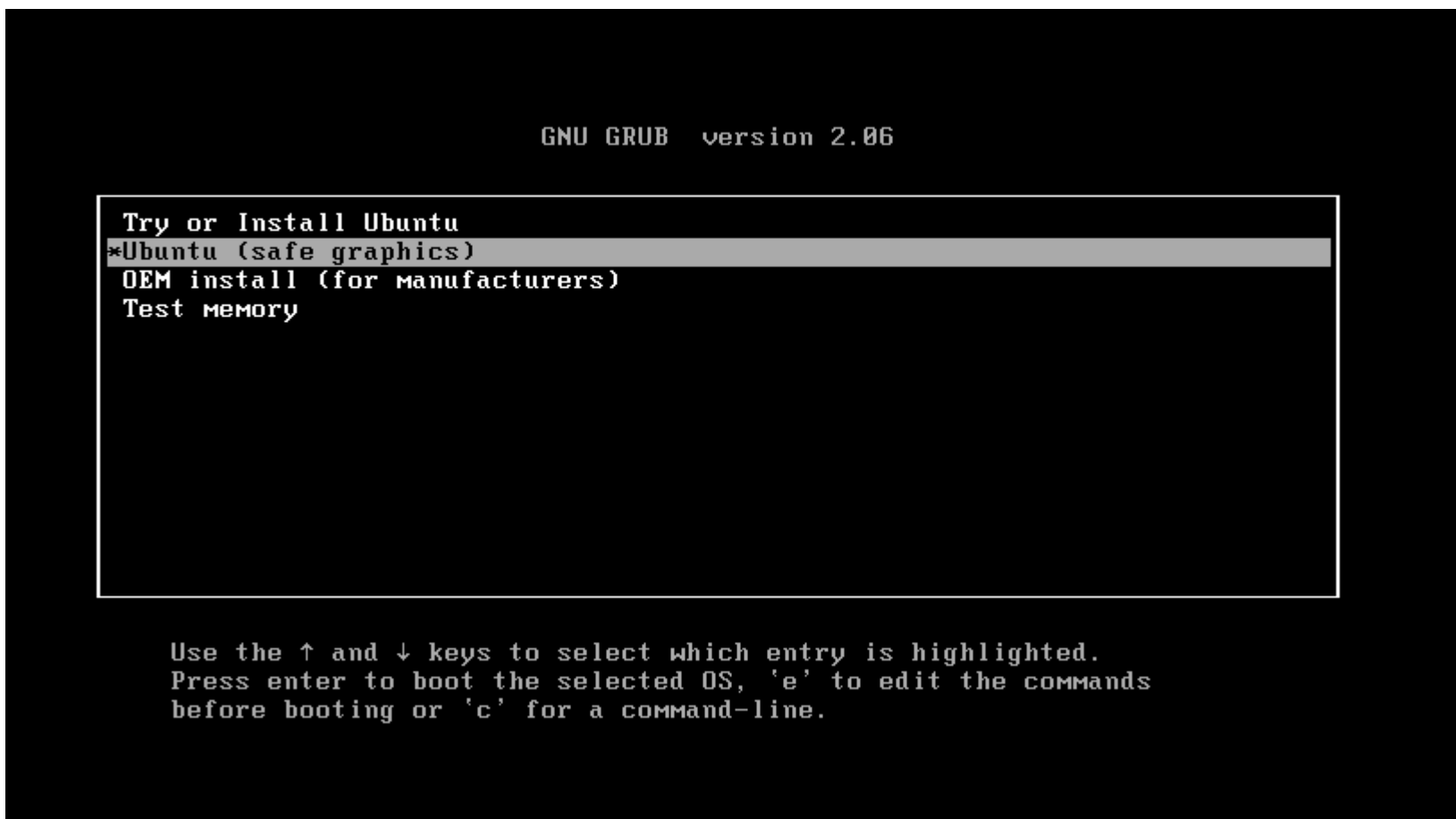
[Ubuntu Studio](#)

[Xubuntu](#)



□ 安装Spark

- 安装Spark之前需要安装Linux系统、Java环境

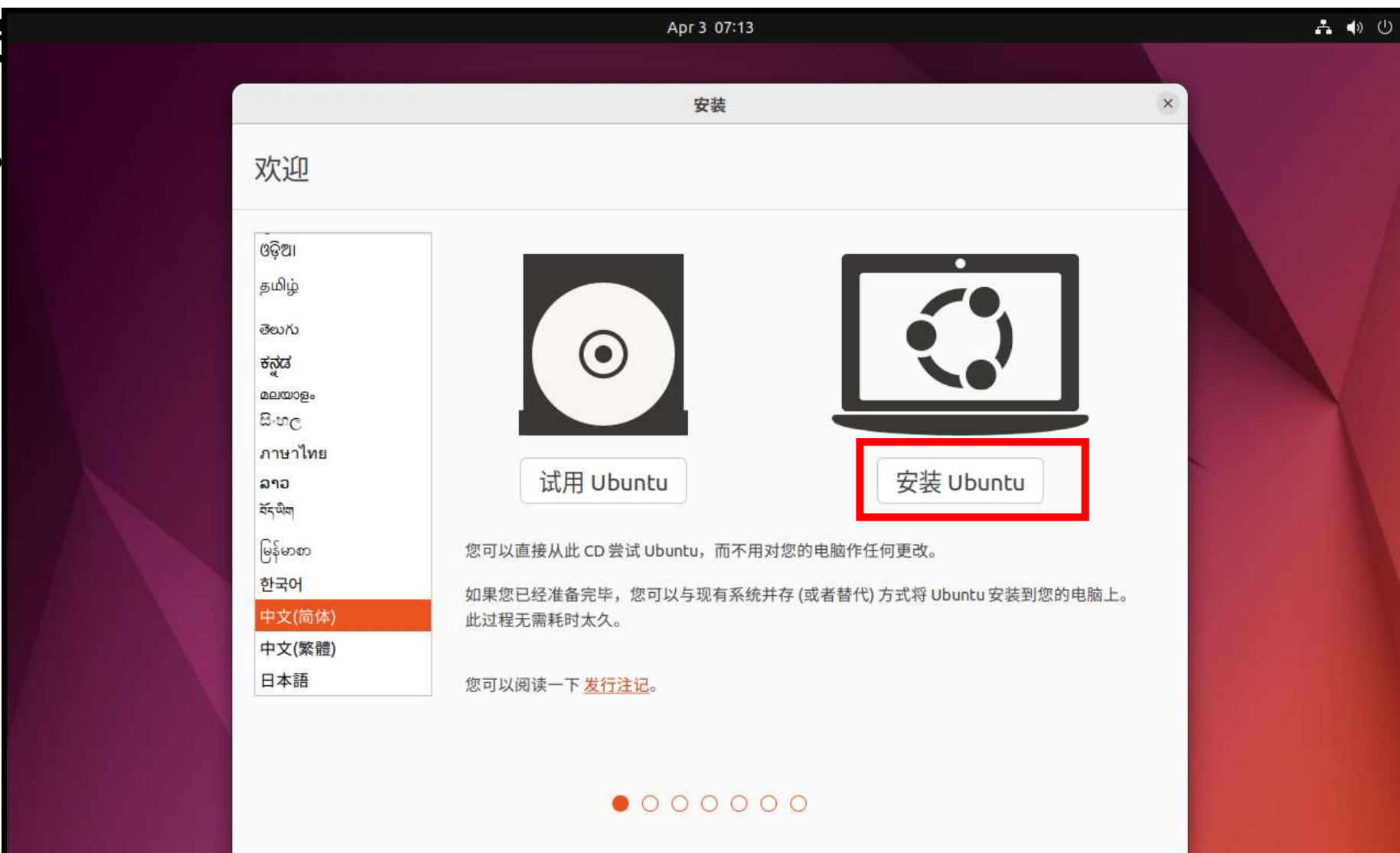




Spark编程实践

□ 安

➤ 安装S

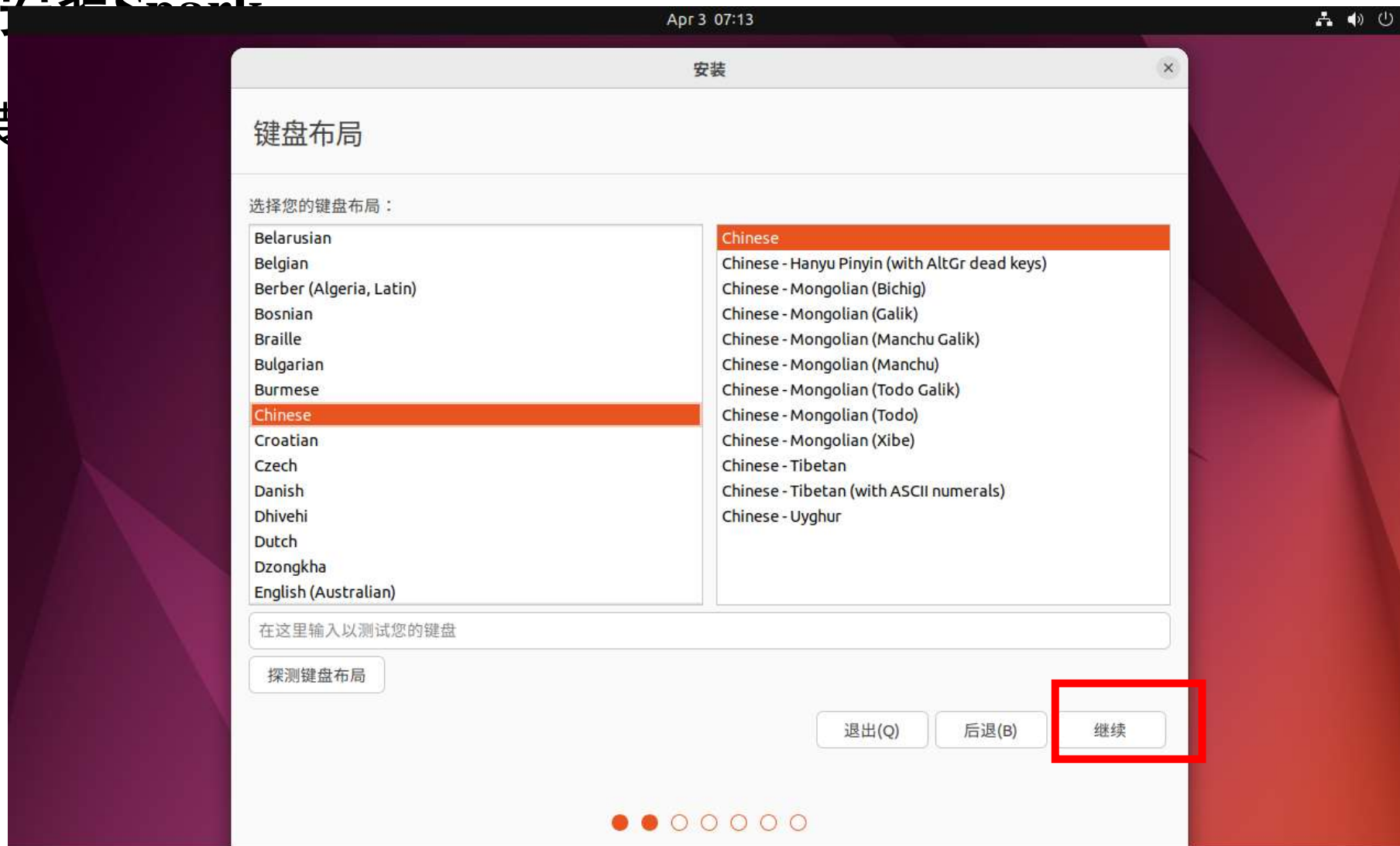




Spark编程实践

安装Spark

安装



安装

更新和其他软件

您希望先安装哪些应用？

☒ 正常安装

网络浏览器、工具、办公软件、游戏和媒体播放器。

☐ 最小安装

网络浏览器和基本工具

其他选项

☒ 安装 Ubuntu 时下载更新

这能节约安装后的时间。

☐ 为图形或无线硬件，以及其它媒体格式安装第三方软件

此软件及其文档遵循许可条款。其中一些是专有的。

退出(Q)

后退(B)

继续





Spark编程实践

安装



安装

安装类型

这台计算机似乎没有安装操作系统。您准备怎么做？

☒ 清除整个磁盘并安装 Ubuntu

注意：这会删除所有系统里面的全部程序、文档、照片、音乐和其他文件。

高级特性...

尚未选择任何项目

☐ 其他

如果您继续，以下所列出的修改内容将会写入到磁盘中。或者，您也可以手动来进行其它修改。

以下设备的分区表已改变：

SCSI33 (0,0,0) (sda)

以下分区将被格式化：

SCSI33 (0,0,0) (sda) 设备上的第 2 分区将设为 系统分区

SCSI33 (0,0,0) (sda) 设备上的第 3 分区将设为 ext4

后退

继续

后退(B)

现在安装(I)

安装

您在什么地方？



Shanghai

后退(B)

继续



安装

您是谁？

您的姓名：

yunjisaun2



您的计算机名：

yunjisaun2-virtual-mac



与其他计算机联络时使用的名称。

选择一个用户名：

yunjisaun2



选择一个密码：

●●●●●●



密码强度：合理

确认您的密码：

●●●●●●

☒ 自动登录☐ 登录时需要密码☐ 使用 Active Directory

您将在下一步中输入域和其他详细信息。

后退(B)

继续



安装

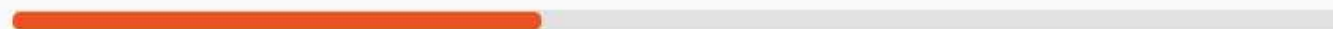
欢迎使用 Ubuntu

最新版本的 Ubuntu 快速且具有丰富新特性，
用起来比以往更方便。这里有一些值得注意的
的新玩意.....



> 正在复制文件...

Skip





□ 安装Spark

➤ 安装Spark之前需

<https://mirrors.tuna.tsinghua.edu.cn/help/ubuntu/>

Ubuntu 软件仓库

线路选择

☒ 是否使用 HTTPS

☐ 是否使用 sudo

本镜像仅包含 32/64 位 x86 架构处理器的软件包，在 ARM(arm64, armhf)、PowerPC(ppc64el)、RISC-V(riscv64) 和 S390x 等架构的设备上（对应官方源为 ports.ubuntu.com）请使用 [ubuntu-ports 镜像](#)。

对于 Ubuntu 不再支持的版本，请参考 [Ubuntu 旧版本帮助](#)。

在 Ubuntu 24.04 之前，Ubuntu 的软件源配置文件使用传统的 One-Line-Style，路径为 `/etc/apt/sources.list`；从 Ubuntu 24.04 开始，Ubuntu 的软件源配置文件变更为 DEB822 格式，路径为 `/etc/apt/sources.list.d/ubuntu.sources`。

将系统自带的对应文件做个备份，然后根据格式的选择下面对应的内容替换，即可使用选择的软件源镜像。

传统格式（`/etc/apt/sources.list`）

Ubuntu 版本

☐ 启用源码源

☐ 启用 proposed

☐ 强制安全更新使用镜像

```
# 默认注释了源码镜像以提高 apt update 速度，如有需要可自行取消注释
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ noble main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ noble main restricted universe multiverse
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ noble-updates main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ noble-updates main restricted universe multiverse
deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ noble-backports main restricted universe multiverse
# deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ noble-backports main restricted universe multiverse

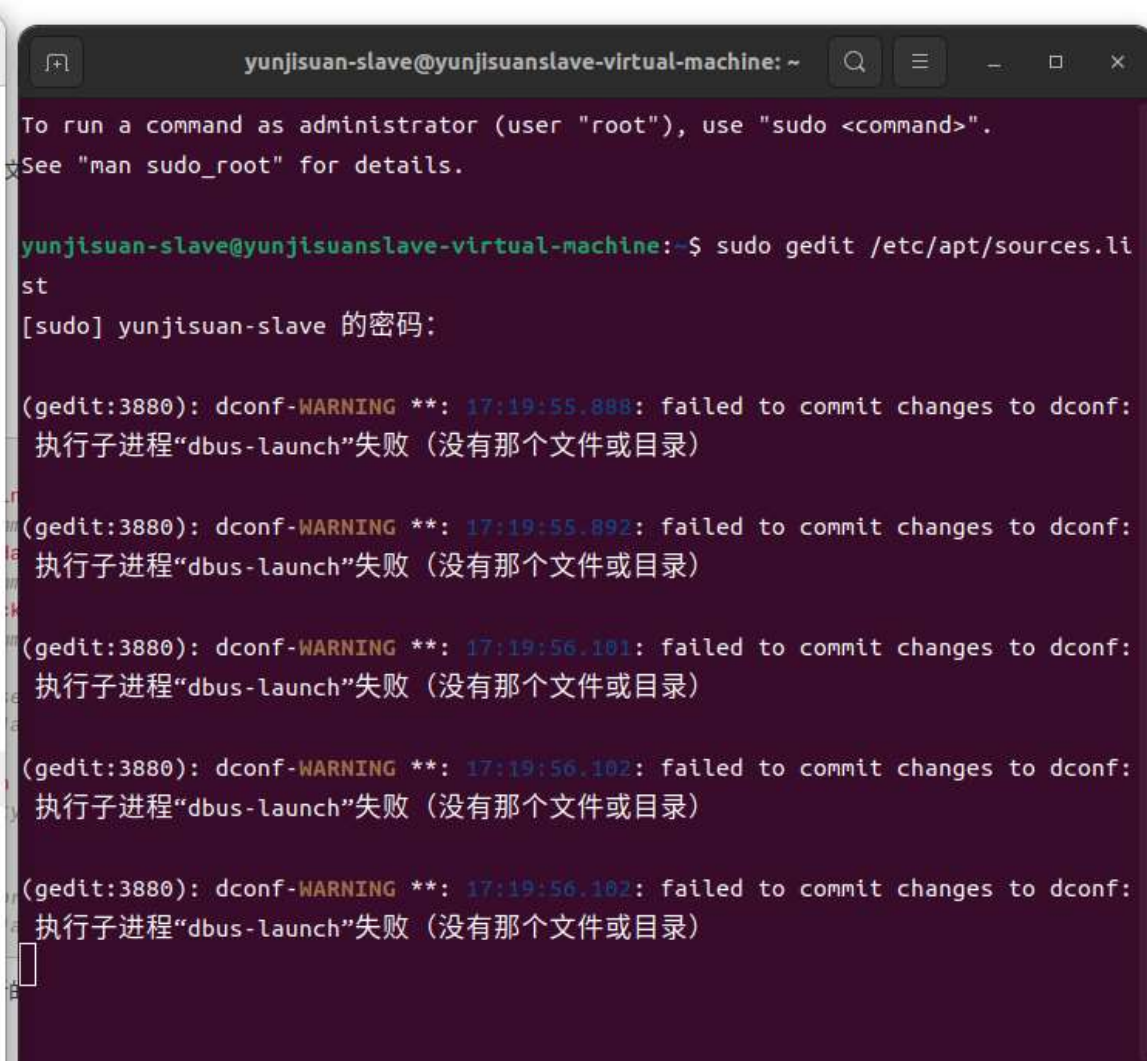
# 以下安全更新软件源包含了官方源与镜像站配置，如有需要可自行修改注释切换
deb http://security.ubuntu.com/ubuntu/ noble-security main restricted universe multiverse
# deb-src http://security.ubuntu.com/ubuntu/ noble-security main restricted universe multiverse

# 预发布软件源，不建议启用
# deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ noble-proposed main restricted universe multiverse
# # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ noble-proposed main restricted universe multiverse
```




The screenshot shows a gedit window titled `*sources.list /etc/apt`. The window contains a list of repository sources for Ubuntu Jammy. Line 15 is highlighted in blue. The text in the window is as follows:

```
1 # 默认注释了源码镜像以提高 apt update 速度，如有需要可自行取消注释
2 deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy main restricted universe multiverse
3 # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy main restricted universe multiverse
4 deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-updates main restricted universe
  multiverse
5 # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-updates main restricted universe
  multiverse
6 deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-backports main restricted universe
  multiverse
7 # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-backports main restricted universe
  multiverse
8
9 # deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-security main restricted universe
  multiverse
10 # # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-security main restricted universe
  multiverse
11
12 deb http://security.ubuntu.com/ubuntu/ jammy-security main restricted universe multiverse
13 # deb-src http://security.ubuntu.com/ubuntu/ jammy-security main restricted universe multiverse
14
15 # 预发布软件源，不建议启用
16 # deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-proposed main restricted universe
  multiverse
17 # # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ jammy-proposed main restricted universe
  multiverse
```



The screenshot shows a terminal window with the prompt `yunjisuan-slave@yunjisuan-slave-virtual-machine: ~`. The terminal displays the following sequence of events:

- A message: "To run a command as administrator (user "root"), use "sudo <command>". See "man sudo_root" for details.
- The command `sudo gedit /etc/apt/sources.list` is entered.
- The prompt changes to `[sudo] yunjisuan-slave 的密码:` (password prompt).
- Four warning messages from dconf follow, each indicating a failure to commit changes to dconf due to a failed `dbus-launch` subprocess. The timestamps for these warnings are 17:19:55.888, 17:19:55.892, 17:19:56.101, and 17:19:56.102.

1. `sudo gedit /etc/apt/sources.list`
2. `sudo apt update`
3. `sudo apt upgrade`



□ 安装Spark

- 安装Spark之前需要安装Linux系统、Java环境

使用SCP命令传输文件

宿主机IP	Master IP	Slave IP
192.168.10.1	192.168.10.128	192.168.10.129
	用户名: yunjisuan	

分别向两台虚拟机发送文件

1. `sudo apt install openssh-server`

以下在宿主机执行

1. `scp .\jdk-8u202-linux-x64.tar.gz yunjisuan@192.168.10.128:~`
2. `scp .\spark-3.0.0-bin-hadoop3.2.tgz yunjisuan@192.168.10.128:~`



□ 安装Spark

- 安装Spark之前需要安装Linux系统、Java环境

解压文件，并重命名

宿主机IP	Master IP	Slave IP
192.168.10.1	92.168.10.128	92.168.10.128
	用户名: yunjisuan	

分别在两台虚拟机执行

1. `tar -zxvf jdk-8u2020-linux-x64.tar.gz`
2. `tar -zxvf spark-3.0.0-bin-hadoop3.2.tgz`
3. `mv jdk1.8.0_202/ jdk`
4. `mv spark-3.0.0-bin-hadoop3.2 spark`



□ 安装Spark

- 安装Spark之前需要安装Linux系统、Java环境

配置环境变量

分别在两台虚拟机执行

1. gedit ~/.bashrc

2. 追加以下内容

```
export JAVA_HOME=~/.jdk
```

```
export JRE_HOME=${JAVA_HOME}/jre
```

```
export CLASSPATH=.:${JAVA_HOME}/lib:${JRE_HOME}/lib
```

```
export PATH=.:${JAVA_HOME}/bin:$PATH
```

3. source ~/.bashrc

4. 重启终端

宿主机IP	Master IP	Slave IP
192.168.10.1	92.168.10.128	92.168.10.128
	用户名: yunjisuan	



单台虚拟机执行

- 本章节内容选择使用Scala进行编程实践，了解Scala有助于更好地掌握Spark。

```
$ ~/spark/bin/spark-shell
```

启动Spark Shell成功后在输出信息的末尾可以看到“Scala >”的命令提示符，如下图所示。

```
Spark context Web UI available at http://192.168.135.130:4040
Spark context available as 'sc' (master = local[*], app id = local-1680517920119).
Spark session available as 'spark'.
Welcome to

      ____ _
     / ___ \_/_ _ _ _ _ \_/_ _
    _\_/ _\_/_ _\_/_ _\_/_ _ \_/_ _
   /___/\_/_ _\_/_ _\_/_ _ \_/_ _ version 3.0.0
      /___\

Using Scala version 2.12.10 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_202)
Type in expressions to have them evaluated.
Type :help for more information.

scala>
```



□ 启动Spark Shell

- Spark Shell 提供了简单的方式来学习Spark API
- Spark Shell可以以实时、交互的方式来分析数据
- Spark Shell支持Scala和Python

单台虚拟机执行

查看可视化界面

<http://192.168.10.128:4040/>



Spark Jobs (?)

User: yunjisuan2023

Total Uptime: 1.2 min

Scheduling Mode: FIFO

▶ Event Timeline



□ 在Spark Shell中运行代码

单台虚拟机执行

1. 在~目录下新建word.txt文件，并写入数据

```
sc.textFile("word.txt").flatMap(_.split(" ")).map((_,1)).reduceByKey(_+_).collect
```

```
scala> sc.textFile("word.txt").flatMap(_.split(" ")).map((_,1)).reduceByKey(_+_).collect  
res0: Array[(String, Int)] = Array((hello,2), (world,1), (spark,2), (hadoop,1))
```



- 编写Spark独立应用程序
- 通过 spark-submit 运行程序

单台虚拟机执行

可以通过spark-submit提交应用程序，该命令的格式如下：

```
./spark/bin/spark-submit  
--class <main-class> //需要运行的程序的主类，应用程序的入口点  
--master <master-url> //Master URL，下面会有具体解释  
--deploy-mode <deploy-mode> //部署模式  
... # other options //其他参数  
<application-jar> //应用程序JAR包  
[application-arguments] //传递给主类的主方法的参数
```



□ 编写Spark独立应用程序

□ 通过 spark-submit 运行程序

单台虚拟机执行

Spark的运行模式取决于传递给SparkContext的Master URL的值。Master URL可以是以下任一种形式：

- local 使用一个Worker线程本地化运行SPARK(完全不并行)
- local[*] 使用逻辑CPU个数数量的线程来本地化运行Spark
- local[K] 使用K个Worker线程本地化运行Spark（理想情况下，K应该根据运行机器的CPU核数设定）
- spark://HOST:PORT 连接到指定的Spark standalone master。默认端口是7077.
- yarn-client 以客户端模式连接YARN集群。集群的位置可以在HADOOP_CONF_DIR 环境变量中找到。
- yarn-cluster 以集群模式连接YARN集群。集群的位置可以在HADOOP_CONF_DIR 环境变量中找到。
- mesos://HOST:PORT 连接到指定的Mesos集群。默认接口是5050。



- 编写Spark独立应用程序
- 通过 spark-submit 运行程序

单台虚拟机执行

```
$ cd spark
$ bin/spark-submit --class org.apache.spark.examples.SparkPi --master local[2] ./examples/jars/spark-examples_2.12-3.0.0.jar 10
```

- 1.--class 表示要执行程序的主类，此处可以更换为咱们自己写的应用程序
- 2.--master local[2] 部署模式，默认为本地模式，数字表示分配的虚拟 CPU 核数量
- 3.spark-examples_2.12-3.0.0.jar 运行的应用类所在的 jar 包，实际使用时，可以设定为咱们自己打的 jar 包
- 4.数字 10 表示程序的入口参数，用于设定当前应用的任务数量

The screenshot shows the Spark Pi application UI. At the top, there's a navigation bar with tabs: Jobs, Stages, Storage, Environment, and Executors. The 'Jobs' tab is selected. Below the navigation bar, the title 'Spark Jobs (?)' is displayed. Underneath, there's a summary section showing 'User: yunjisuan2023', 'Total Uptime: 4 s', 'Scheduling Mode: FIFO', and 'Active Jobs: 1'. A link for 'Event Timeline' is also present. Below this, a section titled 'Active Jobs (1)' shows a table with one job. The table has columns for Job Id, Description, Submitted, Duration, Stages: Succeeded/Total, and Tasks (for all stages): Succeeded/Total. The job details are as follows:

Job Id	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
0	reduce at SparkPi.scala:38 reduce at SparkPi.scala:38 (kill)	2023/04/03 18:48:03	0.9 s	0/1	0/1000 (1 running)

At the bottom of the table, there are pagination controls: 'Page: 1', '1 Pages. Jump to 1', 'Show 100 items in a page.', and a 'Go' button.



□ Standalone 模式运行 Spark

多台虚拟机执行

进入解spark路径的 conf 目录,
修改 slaves.template 文件名为
slaves

Master节点执行

1. `cd ~/spark/conf`
2. 修改slaves.template文件名为slaves
3. 在slaves文件中追加slave IP 192.168.10.129
4. 修改 spark-env.sh.template 文件名为 spark-env.sh
5. 修改 spark-env.sh 文件, 添加 JAVA_HOME 环境变量和集群对应的 master 节点
`export JAVA_HOME=~/.jdk`
`SPARK_MASTER_HOST=192.168.10.128`
`SPARK_MASTER_PORT=7077`
6. 在Slave节点上重复上述操作

宿主机IP	Master IP	Slave IP
192.168.10.1	192.168.10.128	192.168.10.129
	用户名: yunjisuan	



□ Standalone 模式运行 Spark

多台虚拟机执行

Master节点执行

1. 启动Spark集群

~/spark/sbin/start-all.sh

宿主机IP	Master IP	Slave IP
192.168.10.1	192.168.10.128	192.168.10.129
	用户名: yunjisuan	

```
yunjisuan2023@yunjisuan2023-virtual-machine:~$ ~/spark/sbin/start-all.sh
starting org.apache.spark.deploy.master.Master, logging to /home/yunjisuan2023/s
park/logs/spark-yunjisuan2023-org.apache.spark.deploy.master.Master-1-yunjisuan2
023-virtual-machine.out
yunjisuan2023@localhost's password: word:
192.168.135.132: Permission denied, please try again.
yunjisuan2023@192.168.135.132's password:
localhost: starting org.apache.spark.deploy.worker.Worker, logging to /home/yunj
isuan2023/spark/logs/spark-yunjisuan2023-org.apache.spark.deploy.worker.Worker-1
-yunjisuan2023-virtual-machine.out
```



Spark编程实践

□ Standalone 模式运行 Spark


多台虚拟机执行

查看UI界面

<http://192.168.10.128:8080/>

宿主机IP	Master IP	Slave IP
192.168.10.1	192.168.10.128	192.168.10.129
	用户名: yunjisuan	

← → ↻ ⚠ 不安全 | 192.168.135.130:8080 🔍 ⚙ ⭐ ⚡ ⏏ 👤 更新 ⋮

 **Spark Master at spark://192.168.135.130:7077**

URL: spark://192.168.135.130:7077
Alive Workers: 1
Cores in use: 2 Total, 0 Used
Memory in use: 2.8 GiB Total, 0.0 B Used
Resources in use:
Applications: 0 Running, 0 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

▼ Workers (1)

Worker Id	Address	State	Cores	Memory	Resources
worker-20230403190304-192.168.135.130-35811	192.168.135.130:35811	ALIVE	2 (0 Used)	2.8 GiB (0.0 B Used)	

▼ Running Applications (0)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration
----------------	------	-------	---------------------	------------------------	----------------	------	-------	----------

▼ Completed Applications (0)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration
----------------	------	-------	---------------------	------------------------	----------------	------	-------	----------




□ Standalone 模式运行 Spark

多台虚拟机执行

```
1. cd ~/spark
2. bin/spark-submit --class
   org.apache.spark.examples.Sp
   arkPi --master
   spark://192.168.10.
   128:7077 ./examples/jars/spa
   rk-examples_2.12-3.0.0.jar 10
```

宿主机IP	Master IP	Slave IP
192.168.10.1	192.168.10.128	192.168.10.129
	用户名: yunjisuan	

 3.0.0

Spark Master at spark://192.168.135.130:7077

URL: spark://192.168.135.130:7077

Alive Workers: 1

Cores in use: 2 Total, 2 Used

Memory in use: 2.8 GiB Total, 1024.0 MiB Used

Resources in use:

Applications: 1 Running, 1 Completed

Drivers: 0 Running, 0 Completed

Status: ALIVE

Workers (1)

Worker Id	Address	State	Cores	Memory	Resources
worker-20230403190304-192.168.135.130-35811	192.168.135.130:35811	ALIVE	2 (2 Used)	2.8 GiB (1024.0 MiB Used)	

Running Applications (1)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration
app-20230403190819-0001	(kill) Spark Pi	2	1024.0 MiB		2023/04/03 19:08:19	yunjisuan2023	RUNNING	1 s

Completed Applications (1)

Application ID	Name	Cores	Memory per Executor	Resources Per Executor	Submitted Time	User	State	Duration
app-20230403190701-0000	Spark Pi	2	1024.0 MiB		2023/04/03 19:07:01	yunjisuan2023	FINISHED	7 s