

移动机器人开发技术（激光SLAM版）配套教学PPT

序 章

机器人操作系统

机器人硬件平台

机器人核心技术

机器人应用实战

第1课 移动机器人的过去、现在及未来

第2课 初识ROS

第3课 ROS编程初步

第4课 机器人的坐标变换

第5课 机器人仿真环境

第6课 TurtleBot3仿真环境实战

第7课 自主搭建机器人小车

感知

第08课 环境感知基础
第09课 感知数据融合

建图与定位

第10课 机器人的移动控制
第11课 SLAM基础
第12课 SLAM实战

路径规划与导航

第13课 导航基础
第14课 ROS中的导航包
第15课 ROS导航实战

送餐

- 1 送餐机器人结构设计
- 2 送餐机器人环境搭建
- 3 送餐机器人建图
- 4 送餐机器人导航

物流（专题讲座）

- 1 物流机器人结构设计
- 2 物流机器人环境模拟
- 3 物流机器人关键技术
- 4 大规模多机器人调度

图书盘点（专题讲座）

- 1 图书盘点机器人结构
- 2 图书盘点机器人环境
- 3 图书盘点机器人工作模式
- 4 图书盘点中的视觉分析

移动机器人开发技术（激光SLAM版）配套教学PPT

第二课 初识ROS



北京邮电大学

Beijing University of Posts and Telecommunications

移动机器人与智能技术实验室编

宋桂岭 明安龙 2021.9

expsong@qq.com

第2课 初识ROS

北邮移动机器人与智能技术实验室 编

1

Why ROS?

2

安装ROS

3

Hello! Robot

4

ROS体验之旅

5

ROS史话



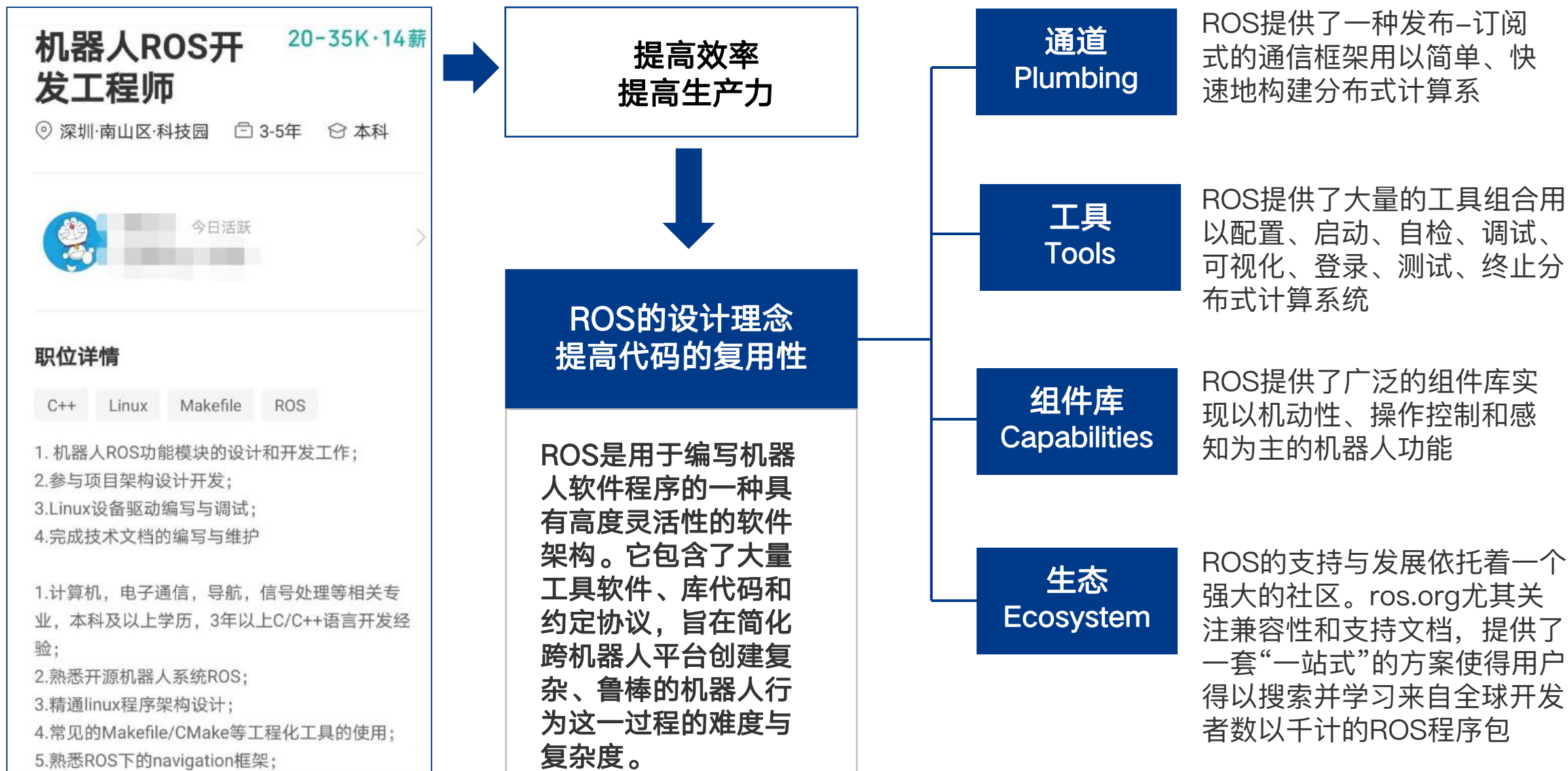
第2课 初识ROS

北邮移动机器人与智能技术实验室 编

1 Why ROS

ROS 是机器人操作系统（Robot Operating System）的英文缩写

1.1 Why ROS



1.2 Why ROS2



ROS Why ROS?
始于2007
提供了可复用的机器人软件开发工具包

2 Why ROS2?
始于2017
更关注实时性、多机协同和安全性

语言支持	C++03 C++11 Python2/3	C++11 C++14 C++17 Python3
平台支持	Linux Mac OS	Linux Windows 10 Mac OS
多节点支持	否 一个进程只能创建一个节点	是 一个进程可以创建多个节点
优点	强大的社区资源 成熟的开发工具包 广泛的开发教程	最小化库依赖 可移植性 持久稳定可靠 实时性
适用对象	ROS初学者和学生群体 科研人员 一般开发群体	专业机器人开发人员 强实时性要求项目

第2课 初识ROS

北邮移动机器人与智能技术实验室 编

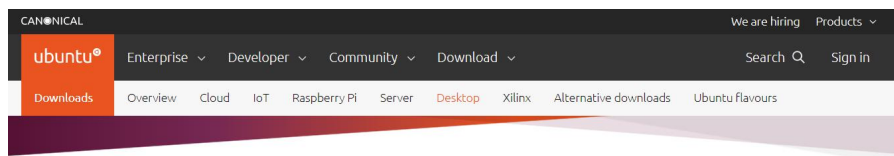
2 安装 ROS

2.1 安装Ubuntu

1

下载最新版本Ubuntu镜像文件

<https://ubuntu.com/download/desktop>



Download Ubuntu Desktop

Ubuntu 20.04.3 LTS

Download the latest [LTS](#) version of Ubuntu, for desktop PCs and laptops. LTS stands for long-term support — which means five years, until April 2025, of free security and maintenance updates, guaranteed.

[Ubuntu 20.04 LTS release notes](#)

Recommended system requirements:

- 2 GHz dual core processor or better
- 4 GB system memory
- 25 GB of free hard drive space
- Internet access is helpful
- Either a DVD drive or a USB port for the installer media

Download

For other versions of Ubuntu Desktop including torrents, the network installer, a list of local mirrors, and past releases see our [alternative downloads](#).

下载文件

ubuntu-20.04.3-desktop-amd64.iso

2

安装方式选择

选项一：硬盘安装

[参考：Create a bootable USB stick on Windows](#)

选项二：虚拟机安装

设备	摘要
内存	8 GB
处理器	8
硬盘 (SCSI)	30 GB
CD/DVD (SATA)	自动检测
网络适配器	NAT
USB 控制器	存在
声卡	自动检测
打印机	存在
显示器	自动检测

① [下载VMware Workstation Pro](#)

② 安装VMware Workstation Pro

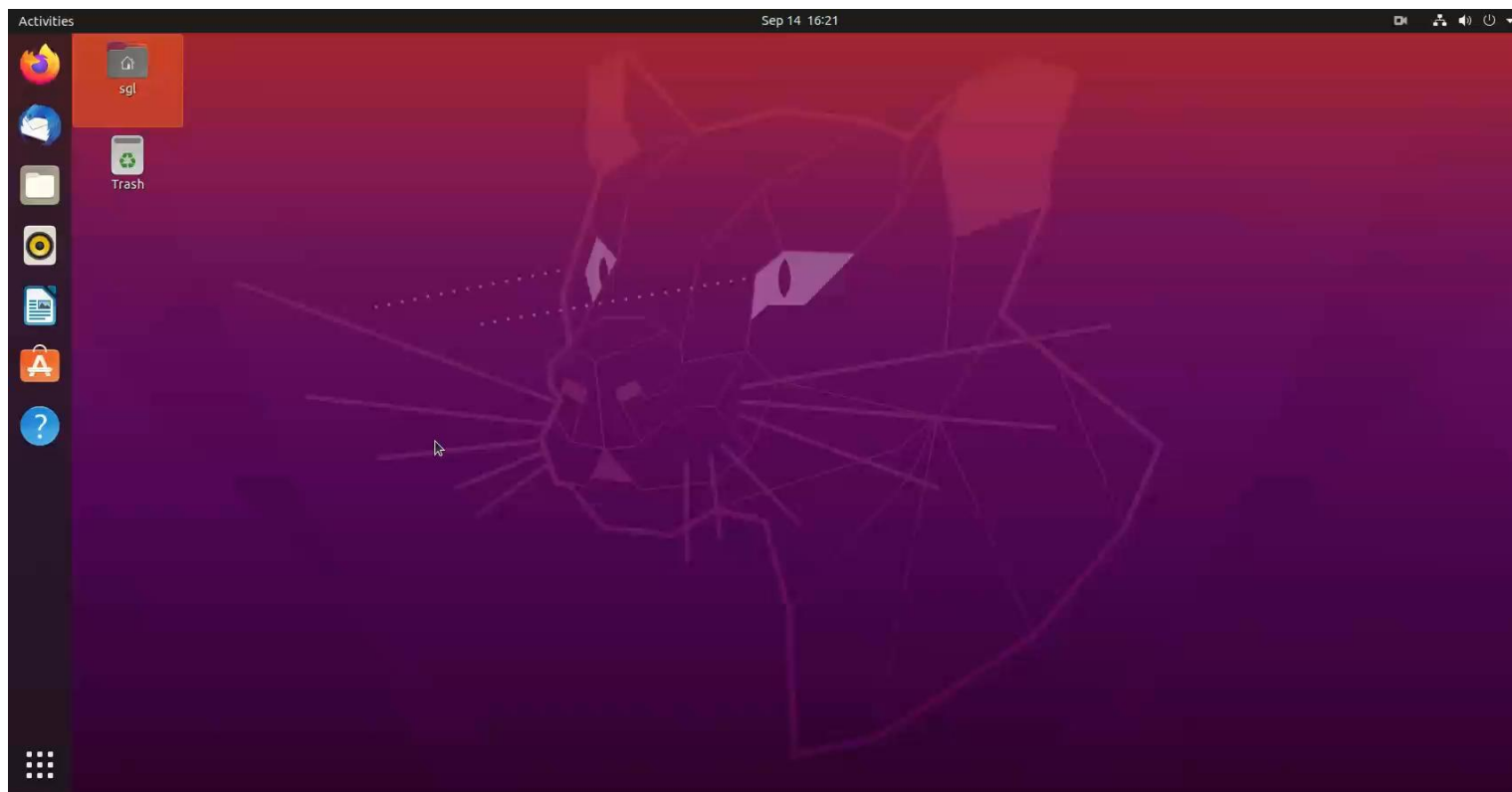
③ 打开VMWare, 添加镜像, 建议内存8G, 硬盘40-60G

2.1 安装Ubuntu

3

快速设置

① 软件更新源



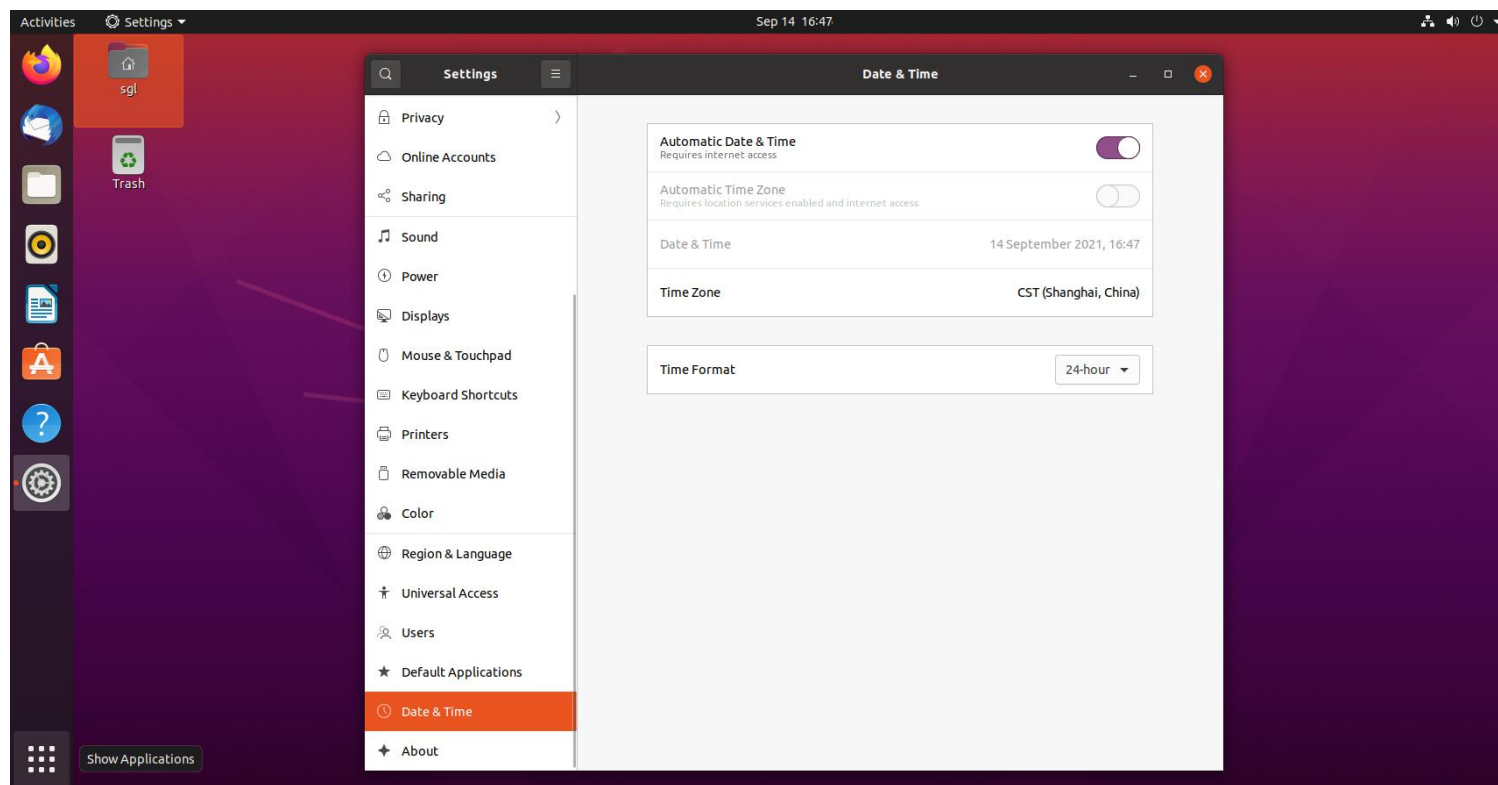
- ① 打开控制台
- ② 选择Software Updater
- ③ 点击Settings
- ④ 选择Ubuntu Software选项卡
- ⑤ 从Download Froms 选项中选择Other
- ⑥ 点击Choose Best Server
- ⑦ 根据系统推荐操作

2.1 安装Ubuntu

3

快速设置

② 时区



① 打开控制台

② 选择Settings工具

③ 左侧选项卡选择Date&Time

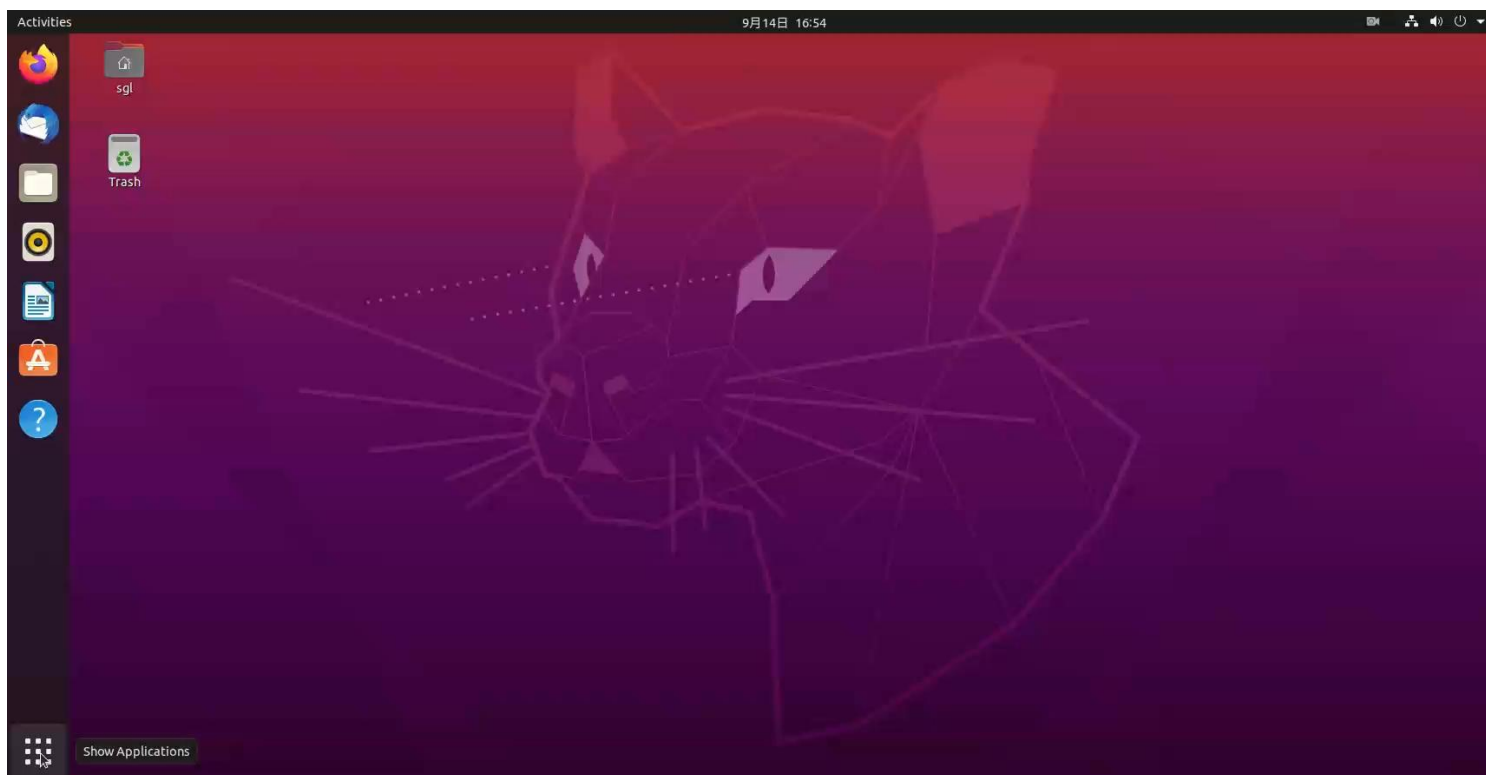
④ 选择时区为上海，系统时间将正确设置

2.1 安装Ubuntu

3

快速设置

③ 语言支持



① 打开控制台

② 选择Settings工具

③ 左侧选项卡选择
Region&Language

④ 添加汉语支持（注意：本步骤
在视频中未体现，添加方式为点
最下面的【Mange Installed
Languages】按钮）

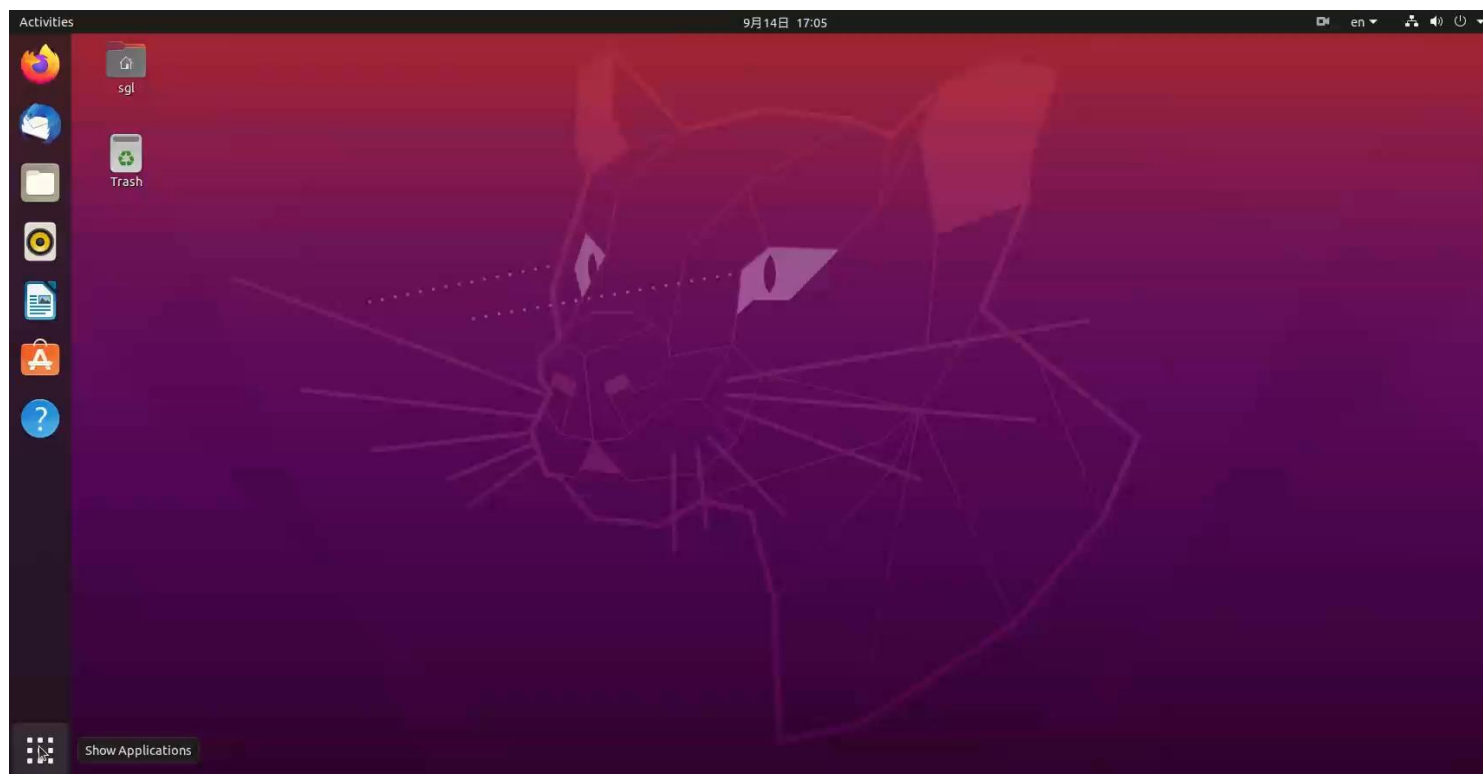
⑤ 添加汉语拼音输入法

2.1 安装Ubuntu

3

快速设置

④ 安装Terminator



① 打开控制台

② 选择Terminal工具

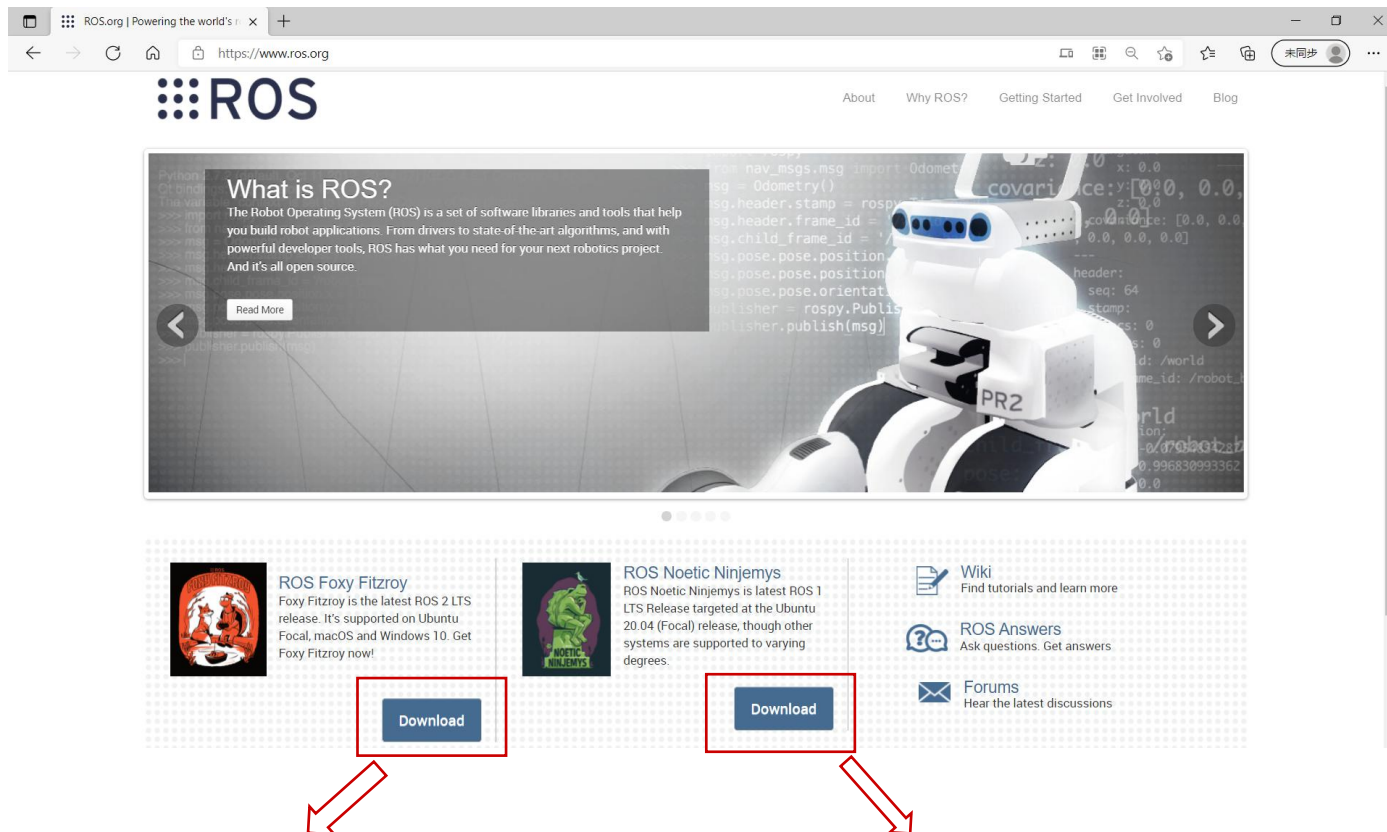
③ `sudo apt-get install terminator`

④ 开打terminator

[Terminator快捷键](#)

2.2 安装ROS

①访问ROS官网 <https://www.ros.org/>



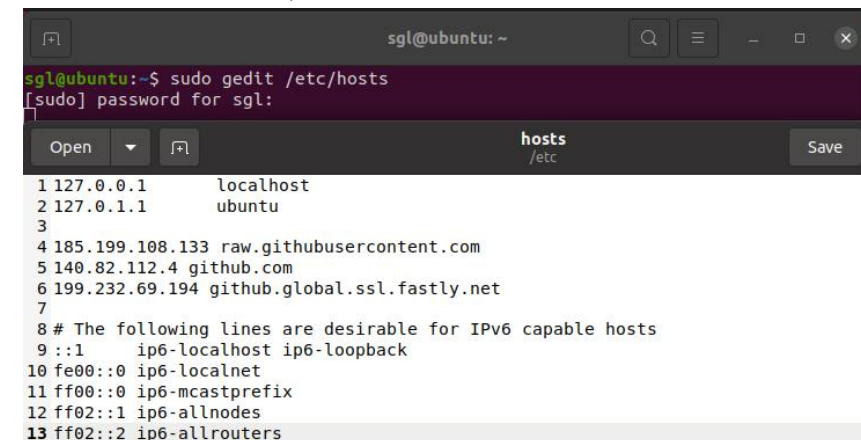
②安装ROS2

③安装ROS1

注意事项一：

安装过程中出现的raw.githubusercontent.com、github.com无法访问等问题，均可通过以下办法解决：

- ① 登录ipaddress.com
- ② 输入raw.githubusercontent.com
- ③ 查询到ip地址，如[185.199.108.133](https://www.ipaddress.com/lookup/raw.githubusercontent.com)
- ④ 打开终端（terminal）
- ⑤ 输入sudo gedit /etc/hosts
- ⑥ 添加地址解析，保存即可



注意事项二：

[安装ROS1 Neotic最后一步出错解决方法](#)

2.2 安装ROS



视频：安装过程—ROS为国内源

①USTC源：

```
$ sudo sh -c '. /etc/lsb-release && echo "deb http://mirrors.ustc.edu.cn/ros/ubuntu/$DISTRIB_CODENAME main" > /etc/apt/sources.list.d/ros-latest.list'
```

②添加私钥：

```
$ wget https://raw.githubusercontent.com/ros/rosdistro/master/ros.key -O - | sudo apt-key add -
```

③更新：

```
$ sudo apt-get update
```

④安装：

```
$ sudo apt-get install ros-noetic-desktop-full
```

⑤测试：

```
$ roscore
```

⑥开发依赖项安装：

```
$ sudo apt install python3-rosdep python3-rosinstall python3-rosinstall-generator python3-wstool build-essential  
$ sudo apt install python3-rosdep  
$ sudo rosdep init  
$ rosdep update
```


2.2 安装ROS

sudo rosdep init出错的解决：

```
sgl@ubuntu:~$ sudo rosdep init
[sudo] password for sgl:

ERROR: Rosdep experienced an error: The read operation timed out
Please go to the rosdep page [1] and file a bug report with the stack trace below.
[1] : http://www.ros.org/wiki/rosdep
```

①创建一个文件：20-default.list，粘贴如下内容：

os-specific listings first

yaml <https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/osx-homebrew.yaml> osx

generic

yaml <https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/base.yaml>

yaml <https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/python.yaml>

yaml <https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/ruby.yaml>

gbpdistro <https://raw.githubusercontent.com/ros/rosdistro/master/releases/fuerte.yaml> fuerte

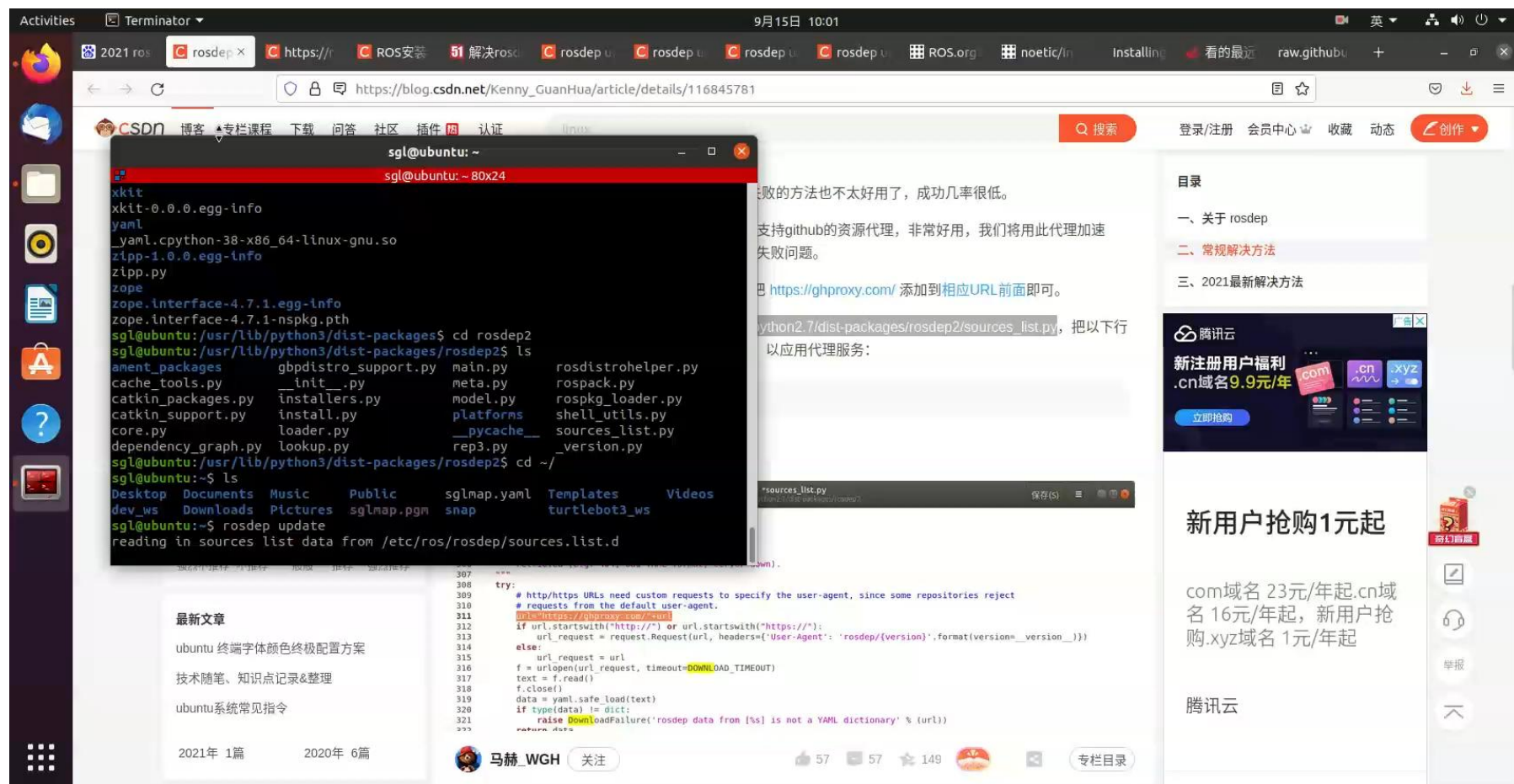
newer distributions (Groovy, Hydro, ...) must not be listed anymore, they are being fetched from the rosdistro index.yaml instead

② 拷贝到/etc/ros/rosdep/sources.list.d文件夹下，如果文件夹不存在，可以如下命令：

sudo mkdir -p /etc/ros/rosdep/sources.list.d

2.2 安装ROS

rosdep update出错的解决:



The screenshot shows a terminal window on an Ubuntu system. The user is in the directory `/usr/lib/python3/dist-packages/rosdep2/` and has run `rosdep update`, which failed with a `YAMLException: data from [url] is not a YAML dictionary` error. The user then navigates to the `sources_list.py` file and modifies the `urls` dictionary to use a proxy (`https://ghproxy.com/`) for the `ubuntu` repository. The user then runs `rosdep update` again, which succeeds.

```
sgl@ubuntu: ~  
sgl@ubuntu: ~ 80x24  
xkit  
xkit-0.0.0.egg-info  
yaml  
_yaml.cpython-38-x86_64-linux-gnu.so  
zipp-1.0.0.egg-info  
zipp.py  
zope  
zope.interface-4.7.1.egg-info  
zope.interface-4.7.1.nspkg.pth  
sgl@ubuntu: /usr/lib/python3/dist-packages$ cd rosdep2  
sgl@ubuntu: /usr/lib/python3/dist-packages/rosdep2$ ls  
ament_packages  gbpdistro_support.py  main.py  rosdistrohelper.py  
cache_tools.py  __init__.py  meta.py  rospack.py  
catkin_packages.py  installers.py  model.py  rospkg_loader.py  
catkin_support.py  install.py  platforms  shell_utils.py  
core.py  loader.py  __pycache__  sources_list.py  
dependency_graph.py  lookup.py  rep3.py  _version.py  
sgl@ubuntu: /usr/lib/python3/dist-packages/rosdep2$ cd ~/  
sgl@ubuntu: ~$ ls  
Desktop  Documents  Music  Public  sglmap.yaml  Templates  Videos  
dev_ws  Downloads  Pictures  sglmap.pgm  snap  turtlebot3_ws  
sgl@ubuntu: ~$ rosdep update  
reading in sources list data from /etc/ros/rosdep/sources.list.d  
...  
try:  
    # http/https URLs need custom requests to specify the user-agent, since some repositories reject  
    # requests from the default user-agent.  
    if url.startswith("http://") or url.startswith("https://"):  
        url_request = request.Request(url, headers={'User-Agent': 'rosdep({version})'.format(version=__version__)})  
    else:  
        url_request = url  
    f = urlopen(url_request, timeout=DOWNLOAD_TIMEOUT)  
    text = f.read()  
    f.close()  
    data = yaml.safe_load(text)  
    if type(data) != dict:  
        raise DownloadFailure('rosdep data from [%s] is not a YAML dictionary' % (url))  
    return data
```

通过修改脚本，添加github代理方式，可以解决网络问题带来的依赖出错问题，详见视频

[链接：安装ROS1 Neotic最后一步出错解决方法](#)

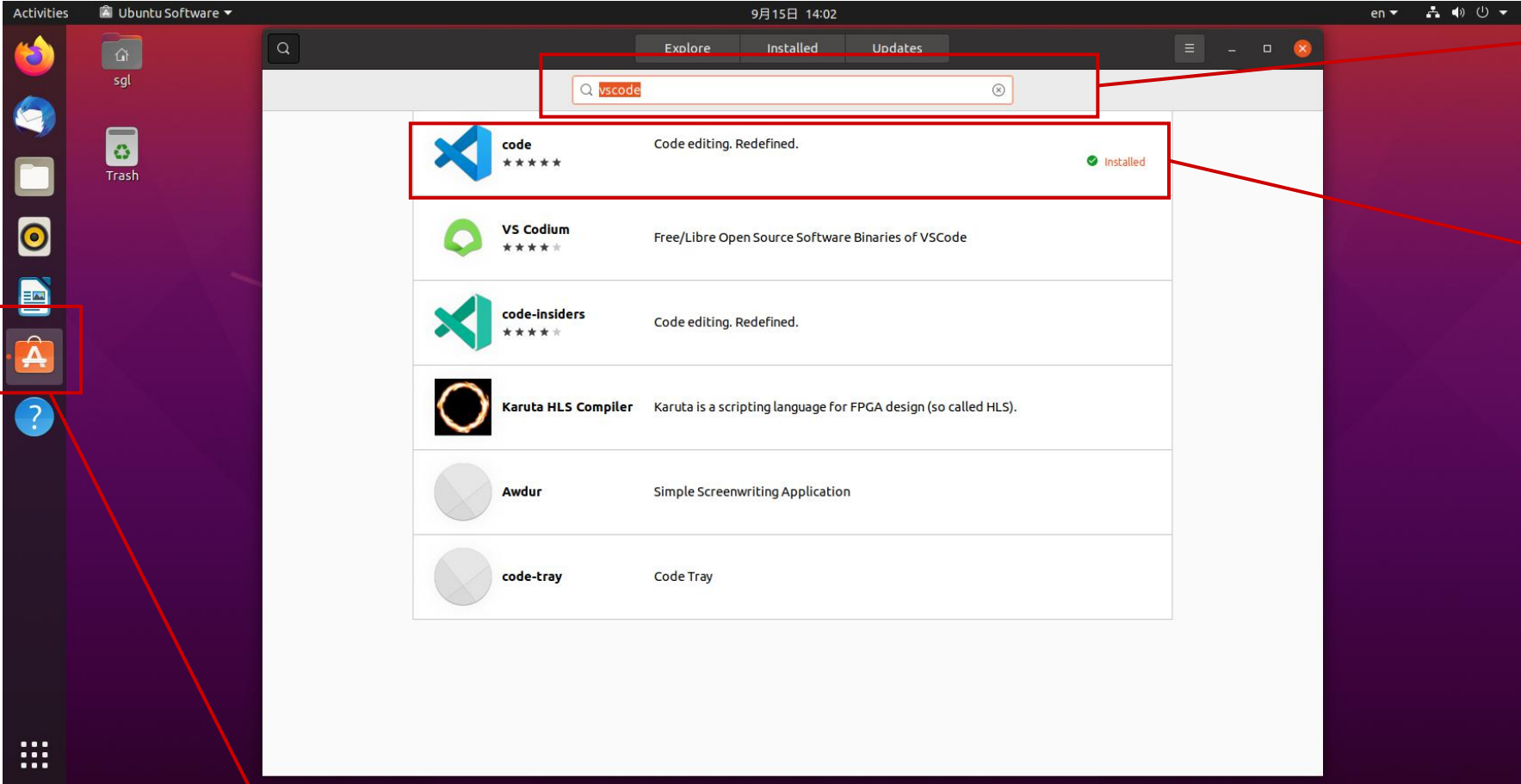
注意安装noetic需要将python2.7改为python3

第2课 初识ROS

北邮移动机器人与智能技术实验室 编

3 Hello! Robot

3.1 开发工具的安装



The screenshot shows the Ubuntu Software Center window. The left sidebar contains icons for various applications, with the 'Software' icon (an orange square with a white 'A') highlighted by a red box. The main window displays a list of applications. At the top, there is a search bar with 'vscode' entered. Below the search bar, the first application listed is 'code' (Visual Studio Code), which is marked as 'Installed' with a green checkmark. A red box highlights the 'code' application entry. Below it, other applications like 'VS Codium', 'code-insiders', 'Karuta HLS Compiler', 'Awdur', and 'code-tray' are listed. Red arrows point from the annotations to the corresponding elements in the screenshot: one to the Software icon, one to the search bar, one to the 'code' application entry, and one to the search bar again.

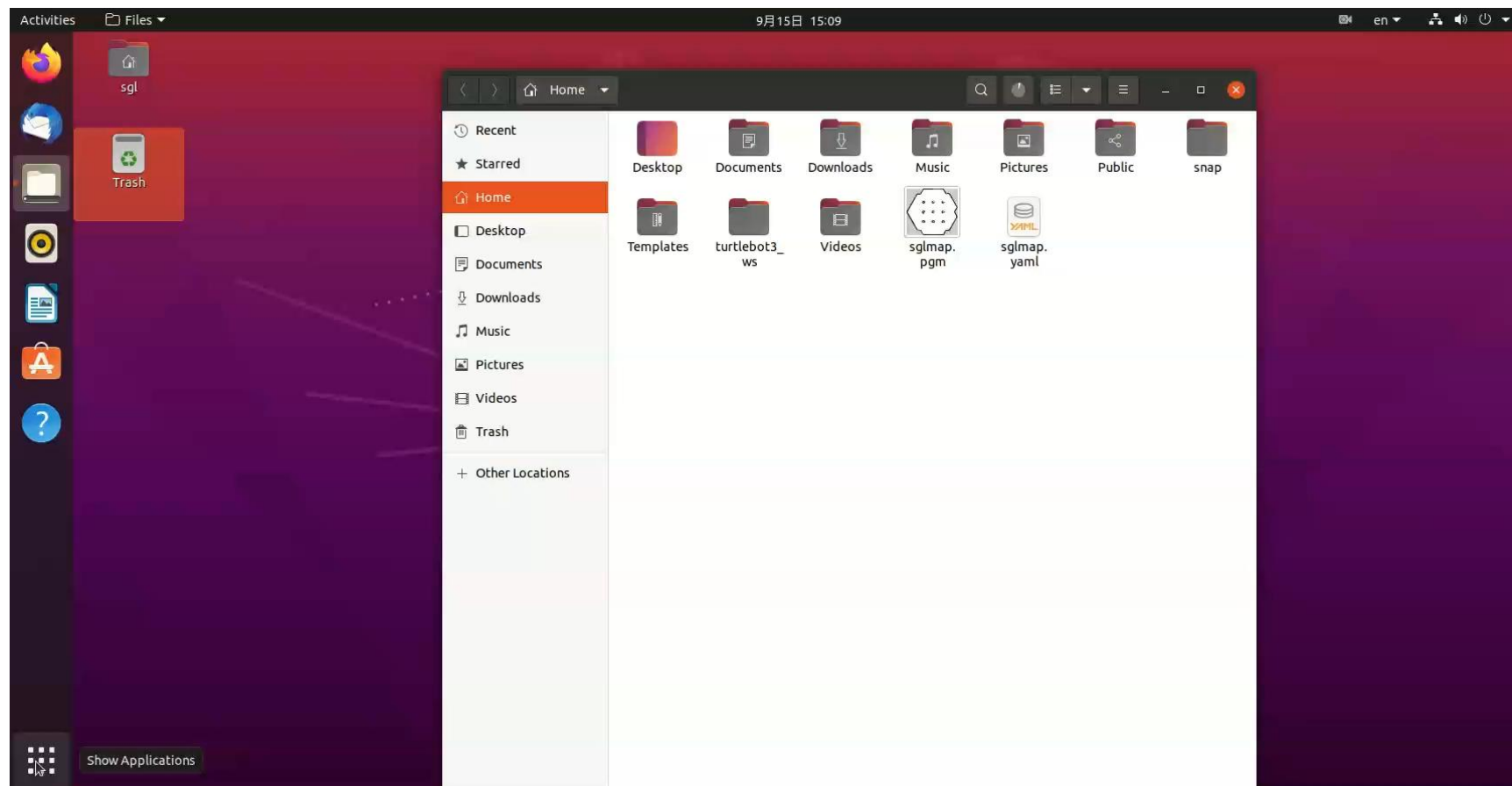
① 打开Ubuntu Software

② 搜索vscode

③ 点击图标安装即可

④ 在工作台中搜索code后，即可打开Visual Studio Code

3.2 ROS1项目开发流程



①创建项目

②编写代码

③配置依赖

④编译

⑤运行

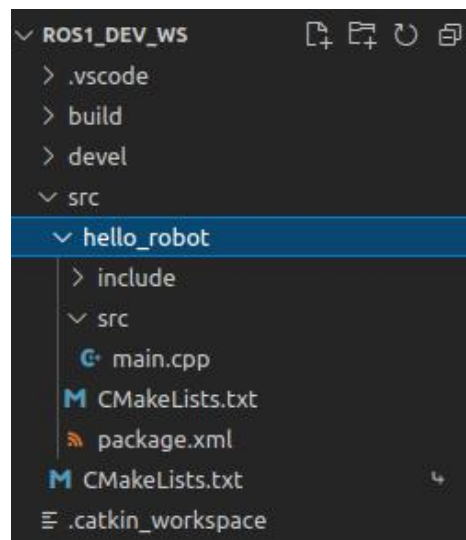
视频：ROS1项目开发流程

3.2 ROS1项目开发流程

1 创建项目

```
$mkdir -p  
~/ros1_dev_ws/src  
$ cd ~/ros1_dev_ws/  
$ catkin_make  
$ cd src  
$ catkin_create_pkg  
hello_robot roscpp
```

文件目录结构



2 编写代码

在src文件夹下添加main.cpp:

```
#include<ros/ros.h>  
int main(int argc, char **argv){  
    ros::init(argc, argv, "hello-!Robot");  
    ROS_INFO( "My First ROS APP!");  
    return 0;  
}
```

3 配置依赖

CMakeList.txt和package.xml

```
add_executable(hello_robot src/main.cpp)  
target_link_libraries(hello_robot  
${catkin_LIBRARIES})
```

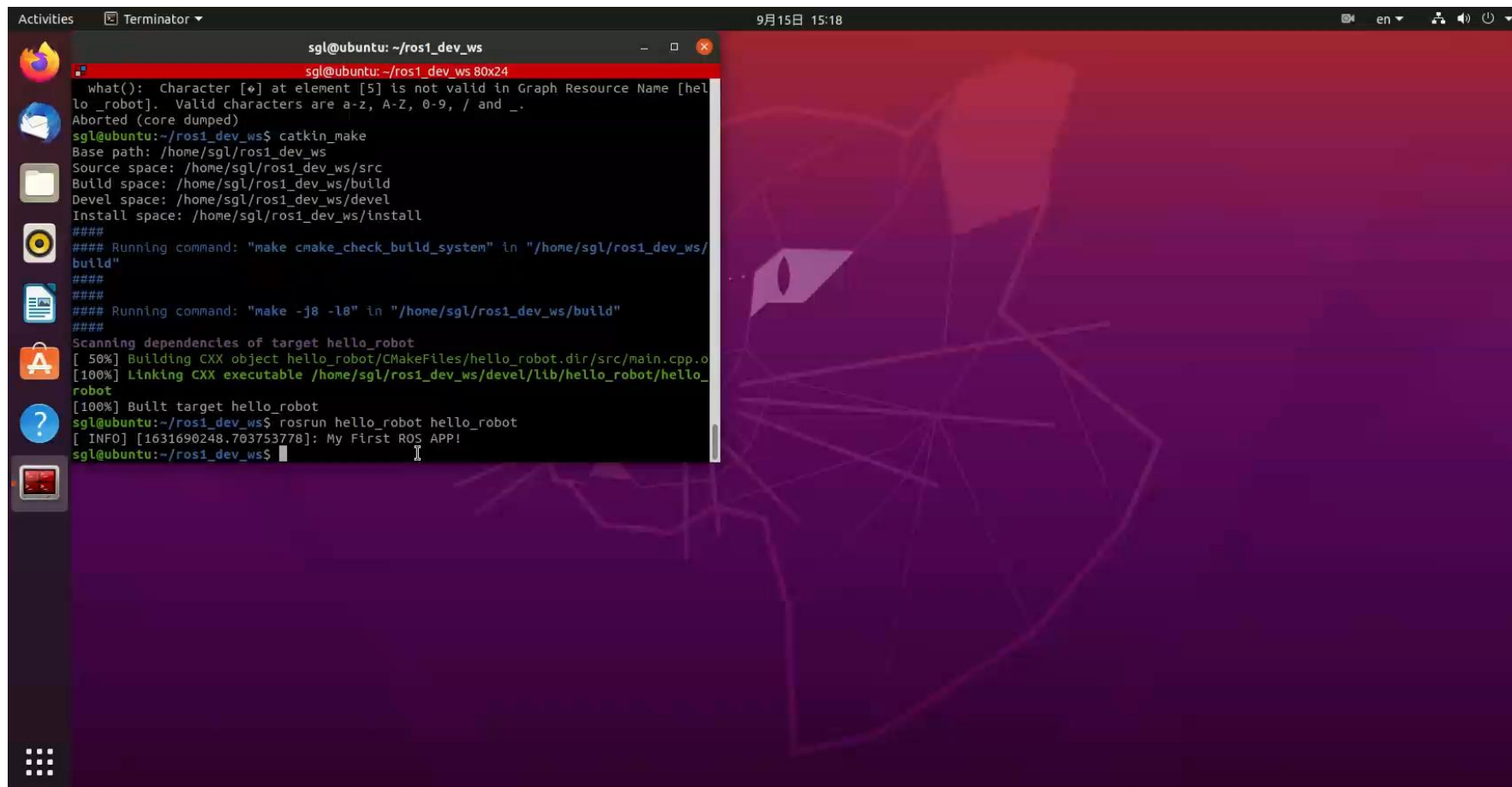
4 编译

```
cd ~/ros1_dev_ws  
catkin_make
```

5 运行

```
$roscore  
$source  
~/ros1_dev_ws/devel/setup.bash  
$ roslaunch hello_robot hello_robot
```

3.3 ROS2项目开发流程



```
sgl@ubuntu: ~/ros1_dev_ws
sgl@ubuntu: ~/ros1_dev_ws 80x24
what(): Character [4] at element [5] is not valid in Graph Resource Name [hello_robot]. Valid characters are a-z, A-Z, 0-9, / and _
Aborted (core dumped)
sgl@ubuntu:~/ros1_dev_ws$ catkin_make
Base path: /home/sgl/ros1_dev_ws
Source space: /home/sgl/ros1_dev_ws/src
Build space: /home/sgl/ros1_dev_ws/build
Devel space: /home/sgl/ros1_dev_ws/devel
Install space: /home/sgl/ros1_dev_ws/install
####
#### Running command: "make cmake_check_build_system" in "/home/sgl/ros1_dev_ws/build"
####
#### Running command: "make -j8 -l8" in "/home/sgl/ros1_dev_ws/build"
####
Scanning dependencies of target hello_robot
[ 50%] Building CXX object hello_robot/CMakeFiles/hello_robot.dir/src/main.cpp.o
[100%] Linking CXX executable /home/sgl/ros1_dev_ws/devel/lib/hello_robot/hello_robot
[100%] Built target hello_robot
sgl@ubuntu:~/ros1_dev_ws$ rosrunc hello_robot hello_robot
[ INFO] [1631690248.703753778]: My First ROS APP!
sgl@ubuntu:~/ros1_dev_ws$
```

①创建项目

②编写代码

③配置依赖

④编译

⑤运行

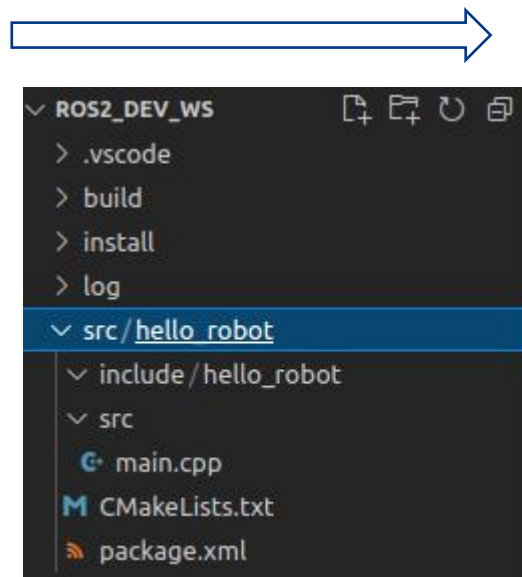
视频：ROS2项目开发流程

3.3 ROS2项目开发流程

1 创建项目

```
$ mkdir -p  
~/ros2_dev_ws/src  
cd ~/ros2_dev_ws/  
$ cd src  
$ ros2 pkg create --build-  
type ament_cmake  
hello_robot
```

文件目录结构



2 编写代码

在src文件夹下添加main.cpp:

```
#include "rclcpp/rclcpp.hpp"  
int main(int argc, char **argv){  
    rclcpp::init(argc, argv);  
    rclcpp::Node node("hello_robot");  
    RCLCPP_INFO(node.get_logger(), "Hello! Robot!");  
    rclcpp::shutdown();  
    return 0; }
```

3 配置依赖

CMakeList.txt

```
find_package(rclcpp REQUIRED)  
add_executable(hello_robot src/main.cpp)  
ament_target_dependencies(hello_robot rclcpp)  
install(TARGETS  
    hello_robot  
    DESTINATION lib/${PROJECT_NAME})
```

package.xml

```
<depend>rclcpp</depend>
```

4 编译

```
cd ~/ros2_dev_ws  
colcon build
```

5 运行

```
roscore  
cd ~/ros2_dev_ws/devel/  
source setup.bash  
roslaunch hello_robot hello_robot
```

第2课 初识ROS

北邮移动机器人与智能技术实验室 编

4 ROS体验之旅

ROS体验之旅--主要以ROS官方教程为主

[ROS1：ROS/Tutorials - ROS Wiki](#)

课后作业：自学完成官方1-9教程

<http://wiki.ros.org/ROS/Tutorials>

1. Core ROS Tutorials

1.1 Beginner Level

1. [Installing and Configuring Your ROS Environment](#)

This tutorial walks you through installing ROS and setting up the ROS environment on your computer.

2. [Navigating the ROS Filesystem](#)

This tutorial introduces ROS filesystem concepts, and covers using the `roscd`, `rosls`, and `rospack` commandline tools.

3. [Creating a ROS Package](#)

This tutorial covers using `roscmake-pkg` or `catkin` to create a new package, and `rospack` to list package dependencies.

4. [Building a ROS Package](#)

This tutorial covers the toolchain to build a package.

5. [Understanding ROS Nodes](#)

This tutorial introduces ROS graph concepts and discusses the use of `roscore`, `roslaunch`, and `rosls` commandline tools.

6. [Understanding ROS Topics](#)

This tutorial introduces ROS topics as well as using the `rostopic` and `rqt_plot` commandline tools.

7. [Understanding ROS Services and Parameters](#)

This tutorial introduces ROS services, and parameters as well as using the `rosservice` and `rosparam` commandline tools.

8. [Using `rqt_console` and `roslaunch`](#)

This tutorial introduces ROS using `rqt_console` and `rqt_logger_level` for debugging and `roslaunch` for starting many nodes at once. If you use ROS `fuerte` or earlier distros where `rqt` isn't fully available, please see this page with [this page](#) that uses old `rx` based tools.

9. [Using `roscd` to edit files in ROS](#)

This tutorial shows how to use `roscd` to make editing easier.

[ROS2: Tutorials — ROS 2 Documentation: Foxy documentation](#)

课后作业：自学完成Beginner: CLI Tools部分

<https://docs.ros.org/en/foxy/Tutorials.html>

Tutorials

The tutorials are a collection of step-by-step instructions meant to steadily build skills in ROS 2.

The best way to approach the tutorials is to walk through them for the first time in order, as they build off of each other and are not meant to be comprehensive documentation.

For quick solutions to more specific questions, see the [Guides](#).

Beginner

Beginner: CLI Tools

- [Configuring your ROS 2 environment](#)
- [Introducing turtlesim and `rqt`](#)
- [Understanding ROS 2 nodes](#)
- [Understanding ROS 2 topics](#)
- [Understanding ROS 2 services](#)
- [Understanding ROS 2 parameters](#)
- [Understanding ROS 2 actions](#)
- [Using `rqt_console`](#)
- [Creating a launch file](#)
- [Recording and playing back data](#)

第2课 初识ROS

北邮移动机器人与智能技术实验室 编

5 ROS史话

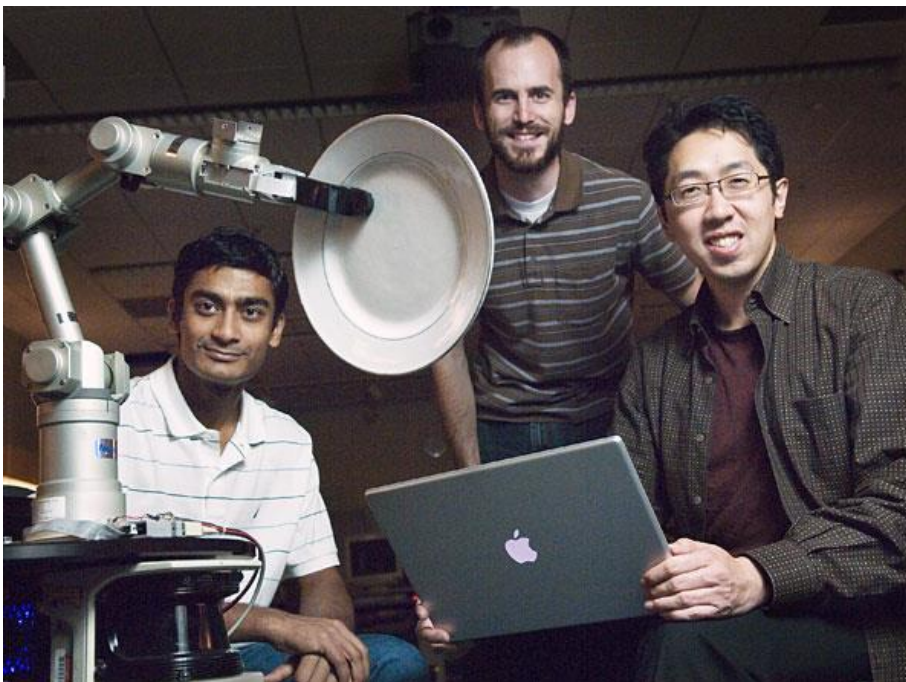
ROS史话

[以史为鉴,可以知兴替：ROS史话36篇](#)、[Pdf版本](#)

Willow Garage 的诞生：2006 年，柳树车库Willow Garage公司诞生于硅谷，距离斯坦福不远。ROS1 编译系统Catkin为柳絮， ROS2 Ament茱萸花序皆为柳树的衍生物， 同年，PR2（Personal Robot，个人机器人2代）项目启动，源于吴恩达学生的PR1项目



柳树街、Facebook、谷歌、斯坦福大学、柳树街 68号地理位置



2006年摩根·奎格利(Morgan Quigley)与他的同学艾舒托什·萨克塞纳(Ashutosh Saxena)和导师吴恩达



PR1项目

ROS史话

[以史为鉴,可以知兴替：ROS史话36篇](#)、[Pdf版本](#)

ROS的诞生：2007年，摩根·奎格利和吴恩达将 STAIR的成果发表在 IEEE国际机器人与自动化会议上，文章的题目是《STAIR: Hardware and Software Architecture》[1]，软件系统的名称是Switchyard6。这个 Switchyard就是 ROS前身

2009年摩根·奎格利、吴恩达和柳树车库机器人公司的工程师们，在当年的 IEEE 国际机器人与自动化会议上发表了《ROS: An Open-Source Robot Operating System》[2]，正式向外界介绍 ROS

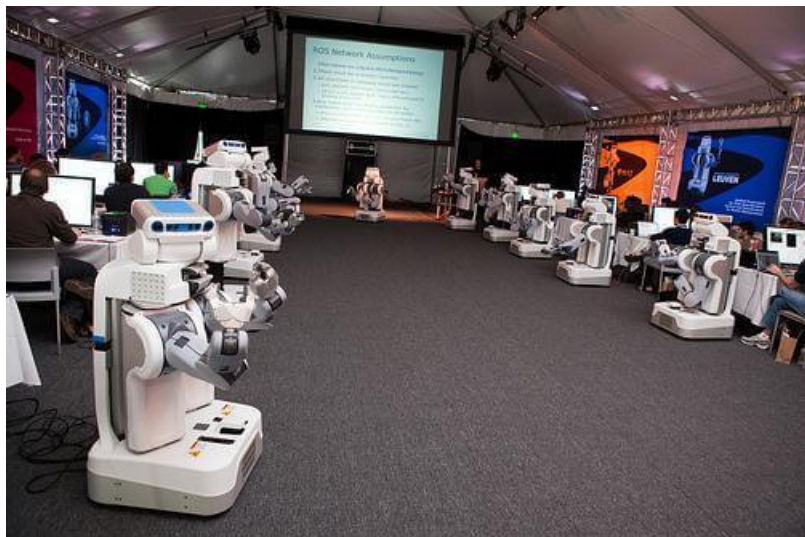
[\[1\]Morgan Quigley, Eric Berger, Andrew Y. Ng. STAIR: Hardware and Software Architecture, IEEE ICRA 2007.](#)

[\[2\] Quigley M , Gerkey B P , Conley K , et al. ROS: An open-source Robot Operating System. IEEE ICRA 2009.](#)

ROS史话

[以史为鉴,可以知兴替：ROS史话36篇](#)、[Pdf版本](#)

PR2的发布：2010 年，随着 PR2 正式对外发布，柳树车库机器人也正式推出 ROS 正式开发版1.0，ROS1命名按照B、C、D、E、F的顺序，一般选择一个同字母开头的单词，第一代ROS命名为ROS Box Turtle，当前版本为ROS Noetic Ninjemys，为第13个发布版本



机器人舞会现场



PR2机器人整齐划一，挥动着 ROS大旗

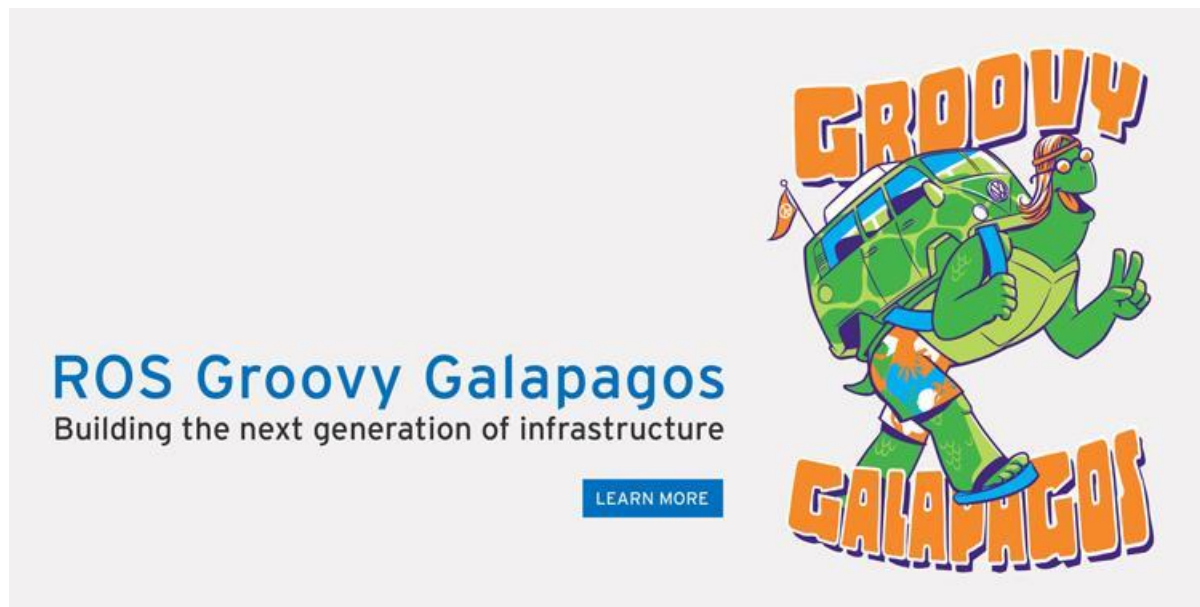


首批 PR2机器人捐给
11所合作研究机构

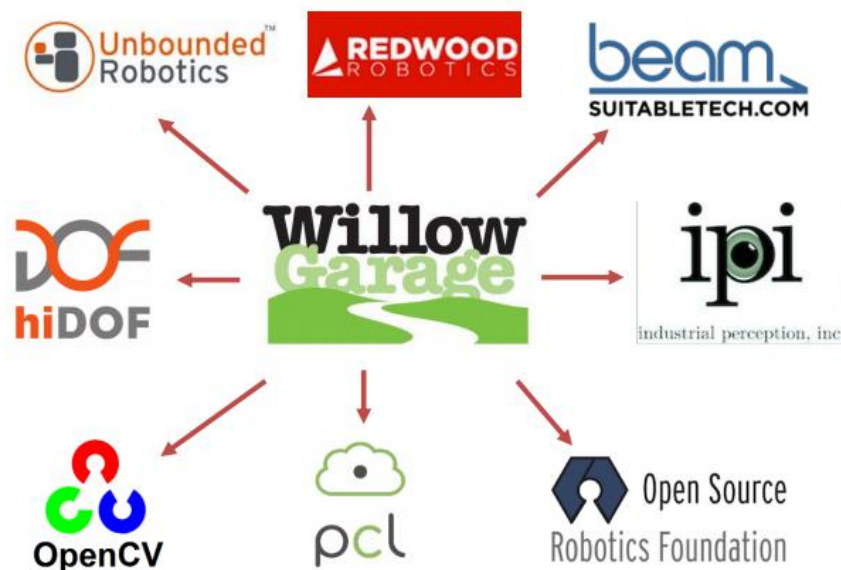
ROS史话

[以史为鉴,可以知兴替：ROS史话36篇](#)、[Pdf版本](#)

Willow Garage的停运：2012年，柳树车库停止资助ROS，最后一个版本为G版本，ROS转为新成立的开源机器人基金会OSRF 推动，新成立的“开源机器人基金会”的创始人兼 CEO 正是在车库负责 ROS的开源软件开发部门的布莱恩·格基（Brian Gerkey）



柳树车库的主页上，ROS永远停留在 G版本上

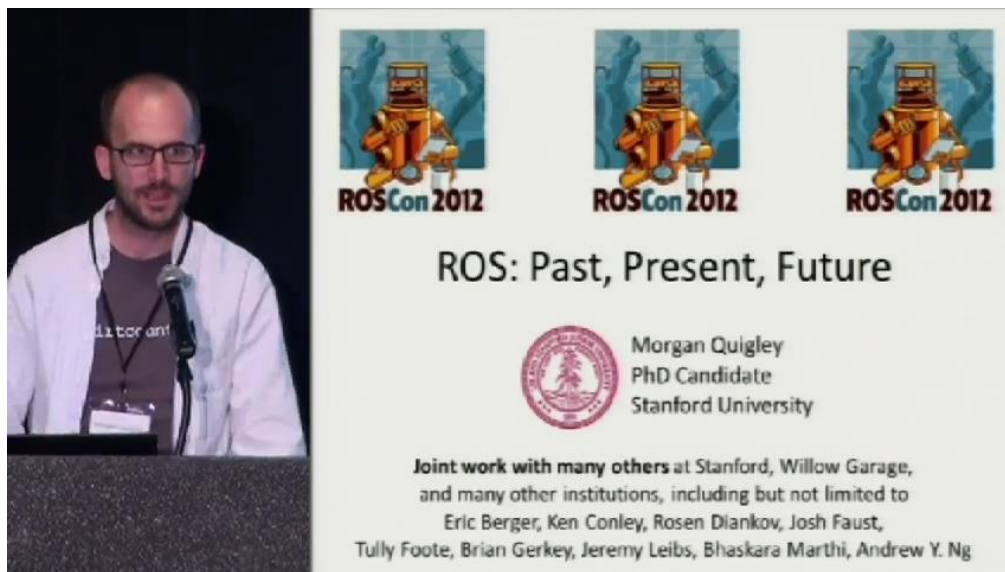


柳树车库系推动了大量的开源项目

ROS史话

[以史为鉴,可以知兴替：ROS史话36篇](#)、[Pdf版本](#)

[ROS开发者大会（https://roscon.ros.org/）](https://roscon.ros.org/)：2012年5月，“开源机器人基金会”组织了第一届ROS开发者大会（简称ROSCon），发布了URDF（统一的机器人描述格式）和MoveIt! 组件，ROS生态开始搭建



2012年第一届ROSCon上
奎格利做了大会唯一的主题报告
《ROS的昨天，今天，明天》



历届ROS开发者大会

移动机器人开发技术（激光SLAM版）配套教学PPT

谢 谢 观 看



北京邮电大学

Beijing University of Posts and Telecommunications

移动机器人与智能技术实验室编

宋桂岭 明安龙 2021.9

expsong@qq.com