

第1季

第2章：C++编程范式



主讲人：张虎
(小虎哥哥爱学习)

- 先导课
- 第1季：快速梳理知识要点与学习方法 ✓
- 第2季：详细推导数学公式与代码解析
- 第3季：代码实操以及真实机器人调试
- 答疑课

----- (永久免费 • 系列课程 • 长期更新) -----

本书内容安排

一、编程基础篇

第1章：ROS入门必备知识

第2章：C++编程范式

第3章：OpenCV图像处理

二、硬件基础篇

第4章：机器人传感器

第5章：机器人主机

第6章：机器人底盘

三、SLAM篇

第7章：SLAM中的数学基础

第8章：激光SLAM系统

第9章：视觉SLAM系统

第10章：其他SLAM系统

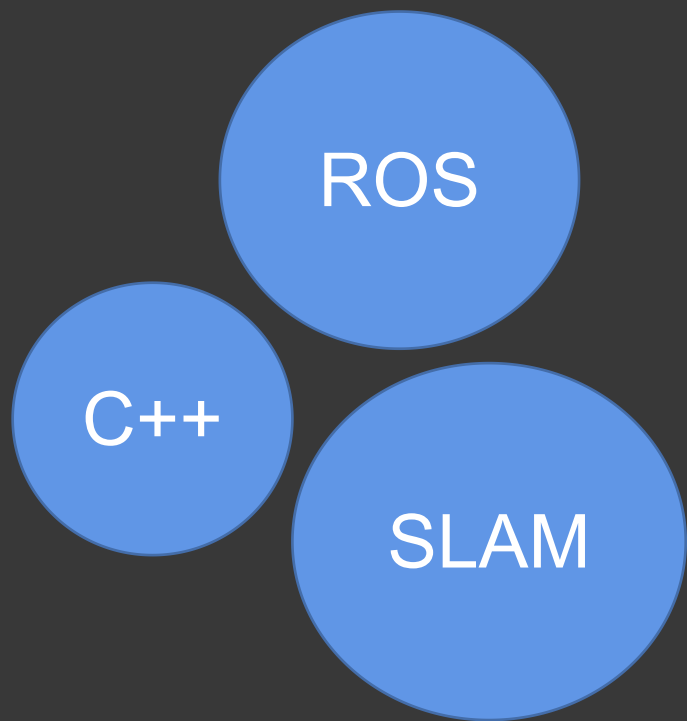
四、自主导航篇

第11章：自主导航中的数学基础

第12章：典型自主导航系统

第13章：机器人SLAM导航综合实战

C++、ROS 和 SLAM 是什么关系？



C++ ↔ 字词句

ROS ↔ 语言技巧

SLAM ↔ 表情达意

《侠客行》 李白

赵客缟胡，吴钩霜雪明。
 银鞍照白马，飒沓如流星。
 十步杀一人，千里不留行。
 事了拂衣去，深藏身与名。
 闲过信陵饮，脱剑膝前横。
 将炙啖朱亥，持觞劝侯嬴。
 三杯吐然诺，五岳倒为轻。
 眼花耳热后，意气素霓生。
 救赵挥金锤，邯郸先震惊。
 千秋二壮士，烜赫大梁城。
 纵死侠骨香，不惭世上英。
 谁能书阁下，白首太玄经。

SLAM大都是用C++编写 为什么很少见到用Python编写呢？

实时性方面，C++要优于Python。

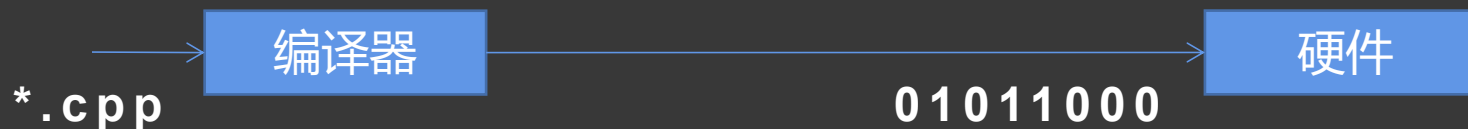
Python是对程序员友好的编程语言。

C++是对计算机友好的编程语言。

SLAM大都是用C++编写

为什么很少见到用Python编写呢?

原生编程语言
(C++)



不能跨平台

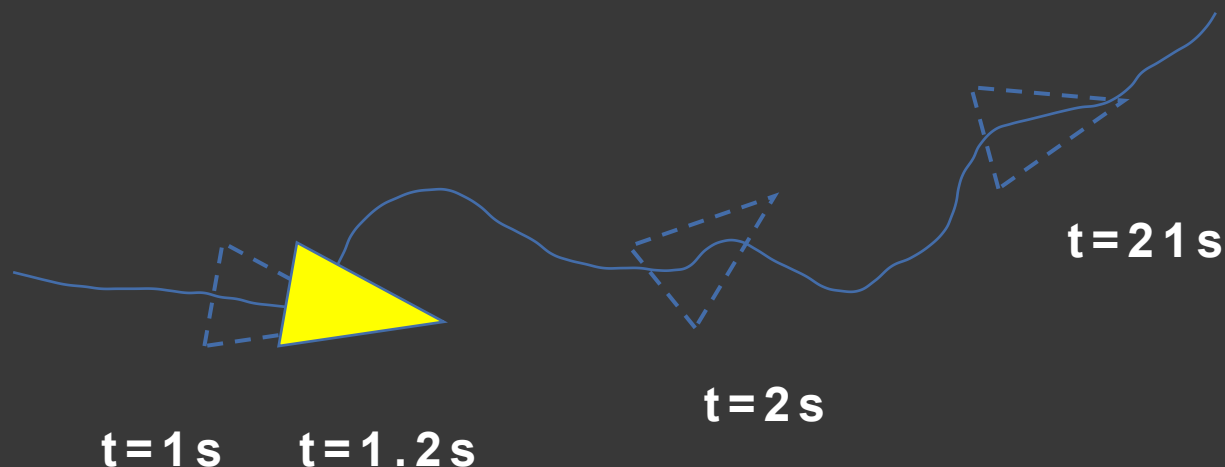
解释型编程语言
(Python)



能跨平台

SLAM大都是用C++编写

为什么很少见到用Python编写呢？



实时性对SLAM的意义

内容概要

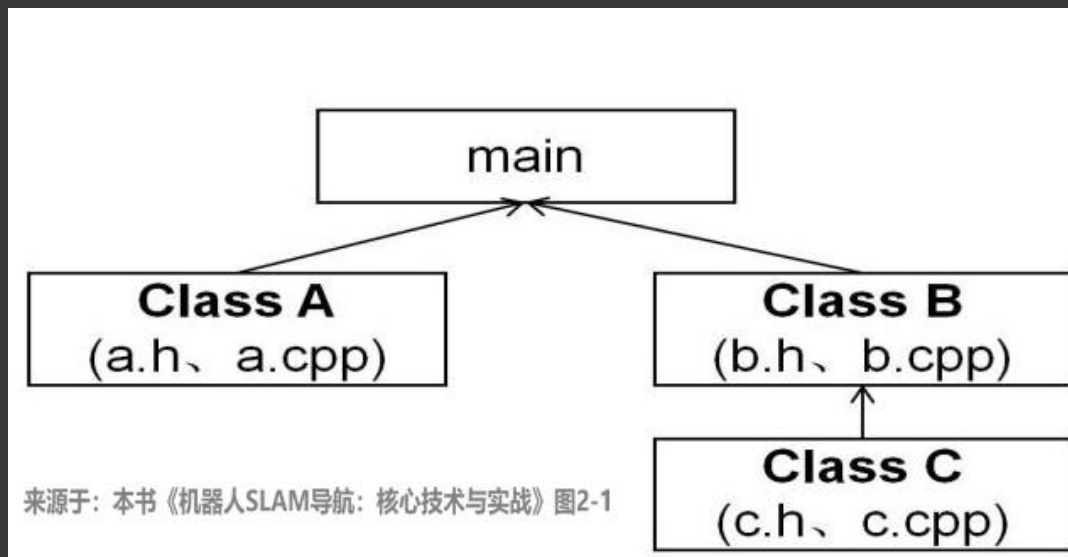
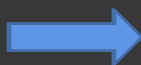
2.1 C++工程的组织结构

2.2 C++代码的编译方法

2.3 C++编程风格指南

2.1 C++工程的组织结构

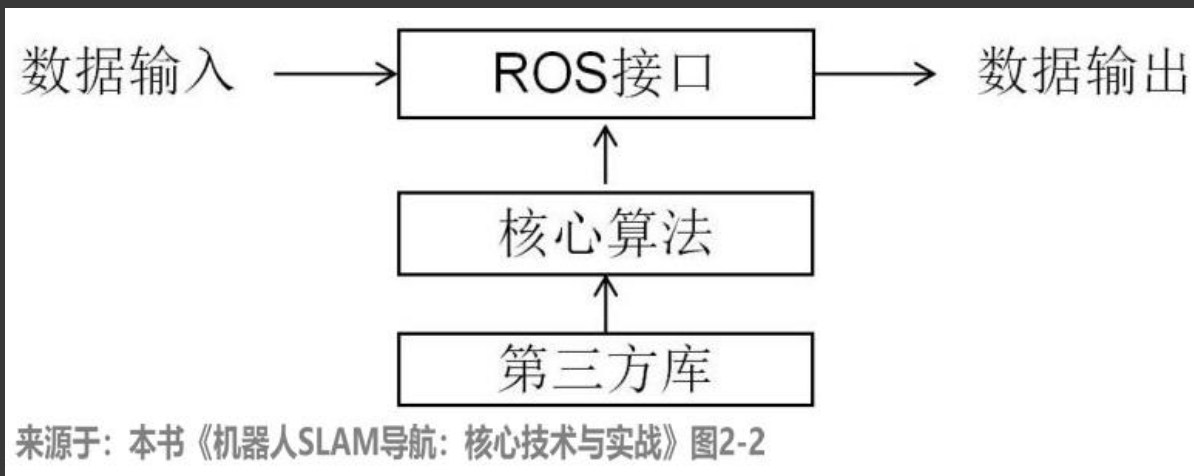
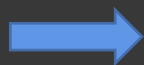
- C++工程的一般组织结构
- C++工程在机器人中的组织结构



2.1 C++工程的组织结构

- C++工程的一般组织结构

- C++工程在机器人中的组织结构



① 如何理解ROS接口的作用

② 从Cartographer代码来理解接口实现


内容概要

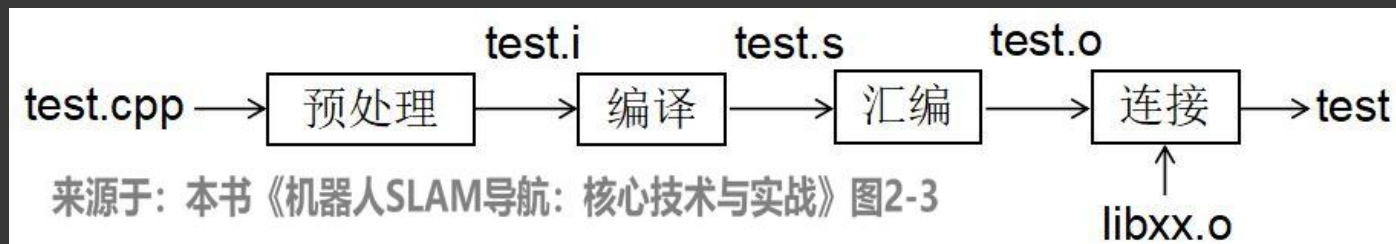
2.1 C++工程的组织结构

2.2 C++代码的编译方法

2.3 C++编程风格指南

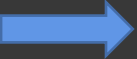
2.2 C++代码的编译方法

- 使用g++编译代码 
- 使用make编译代码
- 使用CMake编译代码



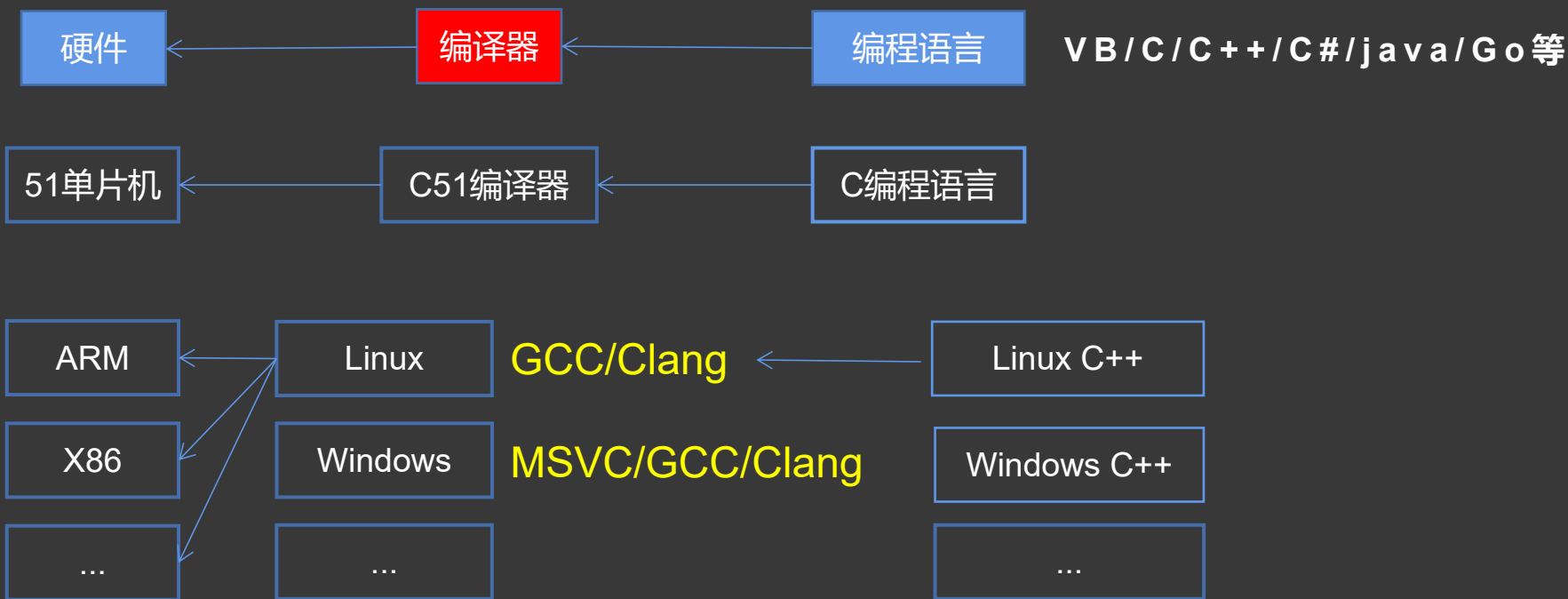
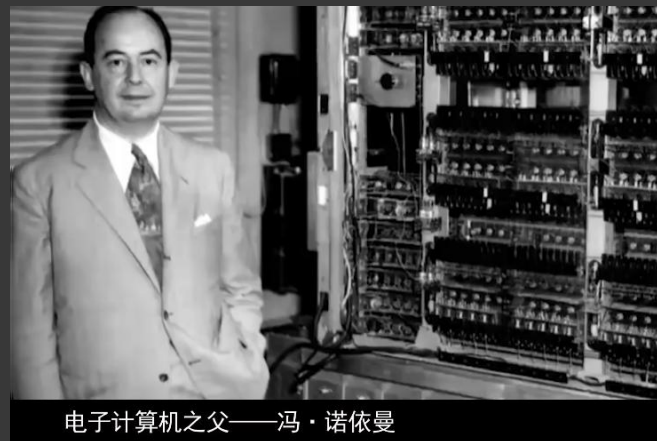
- ① 程序为什么要编译？
- ② gcc/g++是什么？
- ③ 静态链接库libxx.o和动态链接库libxx.so？

2.2 C++代码的编译方法

- 使用g++编译代码 
- 使用make编译代码
- 使用CMake编译代码

① 程序为什么要编译?

详细内容见书的附录A (图A-2)

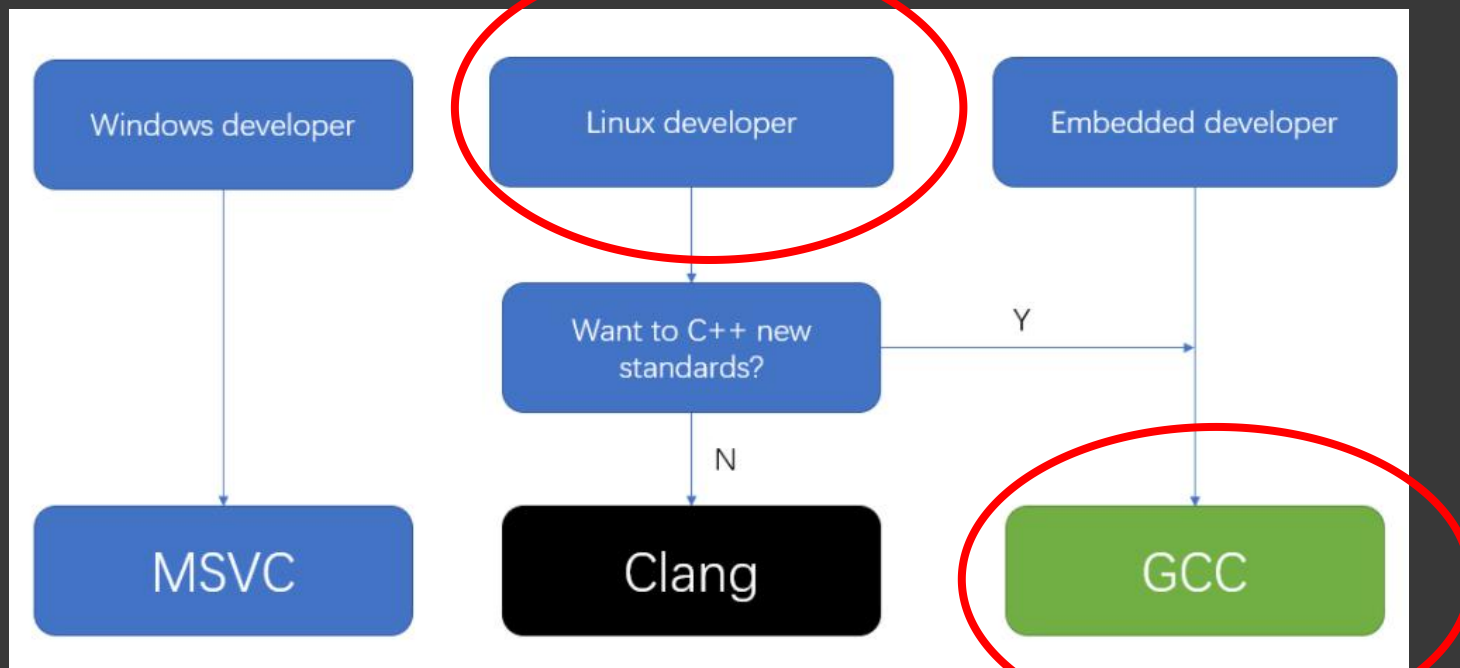
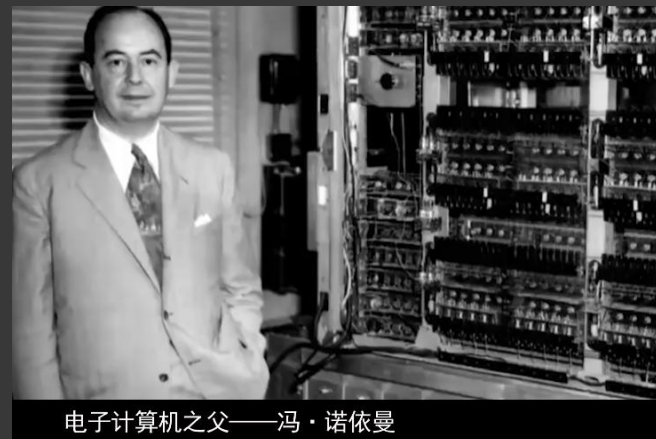


2.2 C++代码的编译方法

- 使用g++编译代码
- 使用make编译代码
- 使用CMake编译代码

① 程序为什么要编译?

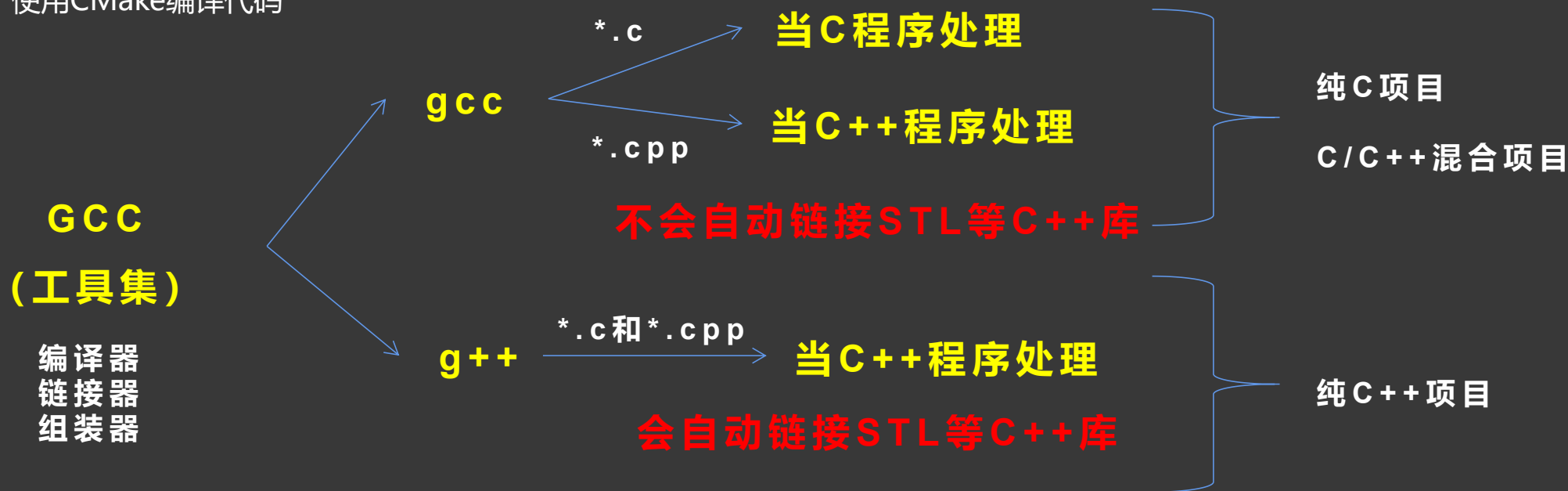
详细内容见书的附录A (图A-2)



2.2 C++代码的编译方法

- 使用g++编译代码
- 使用make编译代码
- 使用CMake编译代码

② gcc/g++是什么?



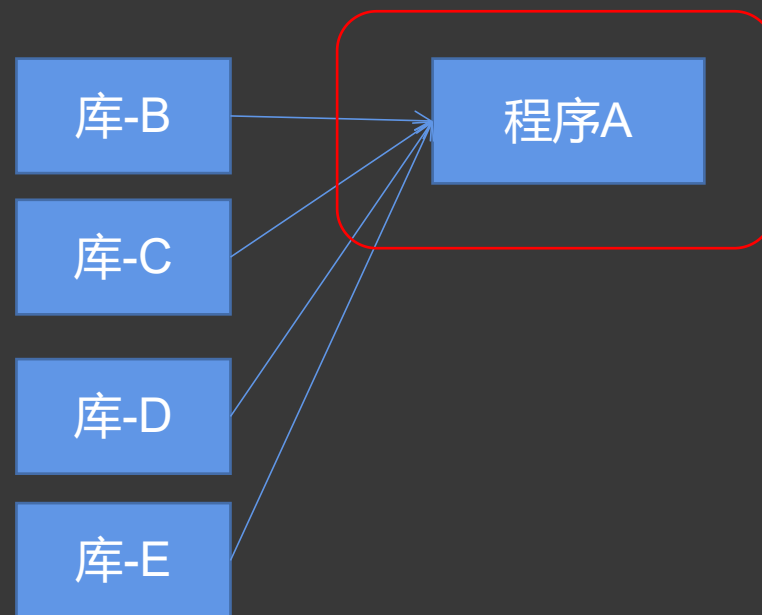
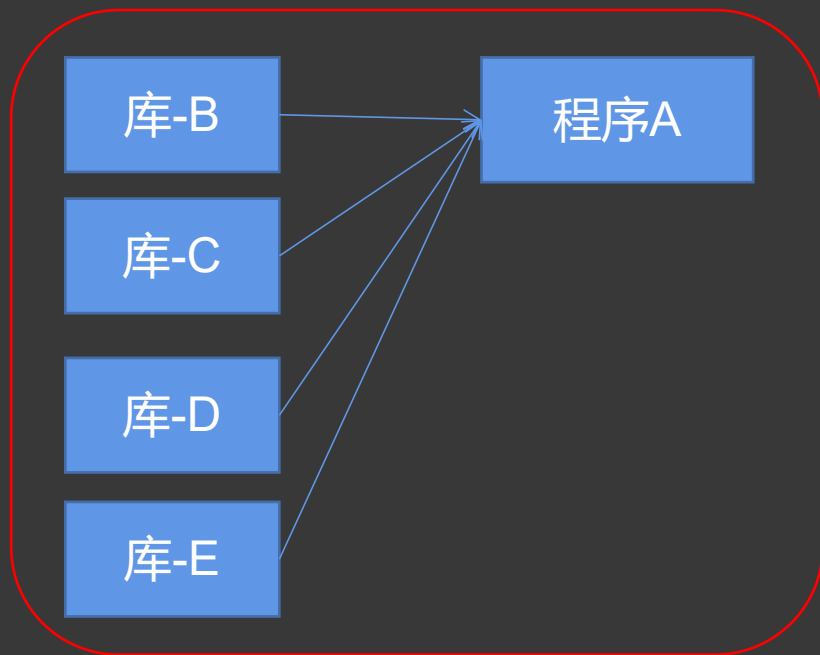
2.2 C++代码的编译方法

■ 使用g++编译代码 

③ 静态链接库libxx.o和动态链接库libxx.so?

■ 使用make编译代码

■ 使用CMake编译代码



2.2 C++代码的编译方法

■ 使用g++编译代码 

■ 使用make编译代码

■ 使用CMake编译代码




```
cd demo/
```

```
g++ foo.cpp main.cpp -o demo
```

```
./demo
```


2.2 C++代码的编译方法

- 使用g++编译代码
- 使用make编译代码 
- 使用CMake编译代码

makefile文件:

```
1 start:  
2  g++ -o foo.o -c foo.cpp  
3  g++ -o main.o -c main.cpp  
4  g++ -o demo foo.o main.o  
5 clean:  
6  rm -rf foo.o main.o
```

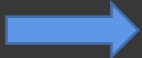
```
cd demo/
```

```
make
```

```
make clean
```

```
./demo
```

2.2 C++代码的编译方法

- 使用g++编译代码
- 使用make编译代码
- 使用CMake编译代码 

CMakeLists.txt文件:

```
1 cmake_minimum_required (VERSION 2.8)
2 project(demo)
3
4 include_directories("${PROJECT_BINARY_DIR}")
5
6 add_library(foo foo.cpp)
7 add_executable (demo main.cpp)
8 target_link_libraries (demo foo)
```

```
cd demo/
```

```
cmake .
```

```
make
```

```
./demo
```

内容概要

2.1 C++工程的组织结构

2.2 C++代码的编译方法

2.3 C++编程风格指南

2.3 C++编程风格指南

- 头文件规范
- 作用域规范
- 类规范
- 命名约定
- ...

① 规避低级错误

② 提高团队开发效率

③ 便于维护升级

Google C++ Style Guide:
<https://google.github.io/styleguide/cppguide.html>

- 例程源码下载：https://github.com/xiihoo/Books_Robot_SLAM_Navigation
- 课件PPT下载：www.xiihoo.com

敬请关注,长期更新...

下集预告