

12、ROS2元功能包介绍

1、元功能包简介

完成一个系统性的功能，可能涉及到多个功能包，比如实现了机器人导航模块，该模块下有地图、定位、路径规划...等不同的子级功能包。那么调用者安装该模块时，需要逐一的安装每一个功能包吗？

显而易见的，逐一安装功能包的效率低下，在ROS2中，提供了一种方式可以将不同的功能包打包成一个功能包，当安装某个功能模块时，直接调用打包后的功能包即可，该包又称之为元功能包 (metapackage)。

MetaPackage是Linux的一个文件管理系统的概念。是 ROS2 中的一个虚包，里面没有实质性的内容，但是它依赖了其他的软件包，通过这种方法可以把其他包组合起来，我们可以认为它是一本书的目录索引，告诉我们这个包集中有哪些子包，并且该去哪里下载。

例如：sudo apt install ros-humble-desktop 命令安装 ros2 时就使用了元功能包，该元功能包依赖于 ROS2 中的其他一些功能包，安装该包时会一并安装依赖。

元功能包经典案例：navigation2

- navigation2 github仓库：[GitHub - ros-navigation/navigation2: ROS 2 Navigation Framework and System](https://github.com/ros-navigation/navigation2)

2、作用

方便用户的安装，我们只需要这一个包就可以把其他相关的软件包组织到一起安装了。

3、实现

1、新建一个功能包

```
ros2 pkg create pkg_metapackage
```

2、修改 package.xml 文件，添加执行时所依赖的包

```
<?xml version="1.0"?>
<?xml-model href="http://download.ros.org/schema/package_format3.xsd"
schematypens="http://www.w3.org/2001/XMLSchema"?>
<package format="3">
  <name>pkg_metapackage</name>
  <version>0.0.0</version>
  <description>TODO: Package description</description>
  <maintainer email="1461190907@qq.com">root</maintainer>
  <license>TODO: License declaration</license>

  <buildtool_depend>ament_cmake</buildtool_depend>

  <exec_depend>pkg_interfaces</exec_depend>
  <exec_depend>pkg_helloworld_py</exec_depend>
  <exec_depend>pkg_topic</exec_depend>
```

```

<exec_depend>pkg_service</exec_depend>
<exec_depend>pkg_action</exec_depend>
<exec_depend>pkg_param</exec_depend>

<test_depend>ament_lint_auto</test_depend>
<test_depend>ament_lint_common</test_depend>

<export>
  <build_type>ament_cmake</build_type>
</export>
</package>

```

3、文件CMakeLists.txt内容如下

```

cmake_minimum_required(VERSION 3.5)
project(pkg_metapackage)

if(CMAKE_COMPILER_IS_GNUCXX OR CMAKE_CXX_COMPILER_ID MATCHES "Clang")
  add_compile_options(-Wall -Wextra -Wpedantic)
endif()

find_package(ament_cmake REQUIRED)

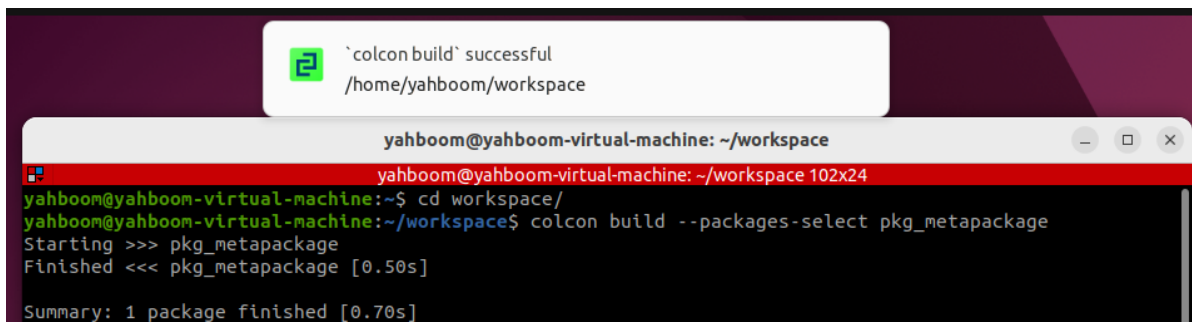
ament_package()

```

4.编译元功能包

- 不会有任何实际可执行文件

```
colcon build --packages-select pkg_metapackage
```



The image shows a terminal window on a virtual machine named 'yahboom@yahboom-virtual-machine'. The user is in the directory '~/workspace'. A notification bubble at the top indicates that the 'colcon build' command was successful. The terminal output shows the command 'colcon build --packages-select pkg_metapackage' being executed, followed by the progress of building the 'pkg_metapackage' package, which finishes in 0.50s. A summary at the bottom states 'Summary: 1 package finished [0.70s]'.

```

yahboom@yahboom-virtual-machine: ~/workspace
yahboom@yahboom-virtual-machine:~$ cd workspace/
yahboom@yahboom-virtual-machine:~/workspace$ colcon build --packages-select pkg_metapackage
Starting >>> pkg_metapackage
Finished <<< pkg_metapackage [0.50s]
Summary: 1 package finished [0.70s]

```