Phytium飞腾

Automake 安装手册

（V1.0）

飞腾信息技术有限公司
www.phytium.com.cn
2024 年 02 月 21 日

商标声明

Phytium 和其他飞腾商标均为飞腾信息技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

特别提示

本文档仅作为使用指导，飞腾对本文档内容不做任何明示或暗示的声明或保证。本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

由于产品版本升级或其他原因，本文档内容会不定期进行更新，如有变更，恕不另行通知。

目录

1 产品概述

Automake 是一个帮助自动生成 Makefile 文件的程序，允许开发类似于 Apache、MySQL 和通用 GNU 软件的软件。

详细介绍请参考：https://baike.baidu.com/item/aotumake/7356085?fr=ge_ala

2 环境要求

2.1 硬件要求

硬件要求如表 2.1 所示。

表 2.1 硬件要求

项目	说明
CPU	飞腾腾云 S2500
网络	无要求
存储	无要求
内存	无要求

2.2 操作系统要求

操作系统要求如表 2.2 所示。

表 2.2 操作系统要求

项目	说明
OpenEuler	22.03

3 Automake 安装

3.1 安装步骤

步骤一 下载 Automake 安装包 automake-1.16.5，链接如下：

```
git@gitee.com:src-openeuler/automake.git
```

步骤二 解压到当前目录

```
tar xf automake-1.16.5.tar.xz
```

如图：

```
[root@localhost automake-master]# tar xf automake-1.16.5.tar.xz
```

步骤三 安装软件依赖

执行以下命令：

```
yum-builddep automake
```

安装的依赖，分别如图：

步骤四 直接 yum 软件包

可以直接执行命令

```
yum install automake
```

4 功能测试

4.1 编写测试用 test.c

步骤 1：创建一个新目录

创建一个新的目录，用于测试目录。您可以在命令行中执行以下命令：

```
mkdir automake_test
cd automake_test
```

步骤 2：创建源代码文件

在新目录中，创建一个简单的 c 或 c++源代码文件，例如‘main.c’或‘main.cpp’，并添加一些代码。以下是一个实例 C 代码：

```
// main.c

#include <stdio.h>
```

```
int main() {
    printf("Hello, Automake!\n");
    return 0;
}
```

步骤 3：创建‘configure.ac’文件

在项目目录中创建一个名为‘configure.ac’文件，并将以下内容添加到文件中：

```
# 使用 AC_INIT 宏来定义项目的名称、版本和联系方式
AC_INIT([myproject], [1.0], [youremail@example.com])

# 使用 AM_INIT_AUTOMAKE 来初始化 Automake，通常使用"-Wall -Werror foreign"
AM_INIT_AUTOMAKE([-Wall -Werror foreign])

# 检查编译器是否支持 C 语言
AC_PROG_CC

# 使用 AC_CONFIG_HEADERS 生成 config.h 头文件，用于存放配置信息
AC_CONFIG_HEADERS([config.h])

# 如果有其他需要的检查或设置，可以在这里添加

# 最后使用 AC_OUTPUT 指定生成的输出文件，通常是 Makefile
AC_CONFIG_FILES([Makefile])

# 结束 configure.ac 文件
AC_OUTPU
```

步骤 4：创建一个‘makefile.am'

打开文件根据项目需求定义构建规则，构建上述提到的简单 c 项目：

```
# Makefile.am

# 声明要构建的可执行文件
bin_PROGRAMS = automake_test

# 可执行文件的源文件
automake_test_SOURCES = main.c
```

步骤 5：运行’autoreconf‘命令

在项目目录中运行以下命令来生成配置脚本：

```
autoreconf --install
```

这将使用 Automake 和 Autoconf 生产必要的文件和脚本

```
[root@localhost automake_test]# autoreconf --install
configure.ac:8: installing './compile'
configure.ac:5: installing './install-sh'
configure.ac:5: installing './missing'
Makefile.am: installing './depcomp'
```

步骤 6：创建 Makefile

运行’./configure‘命令，他将使用’configure.ac‘文件中的配置信息生产 Makefile:

```
./configure
```

步骤 7：编译项目

使用生成的 makefile 来编译项目。

```
make
```

步骤 8：运行项目

如果您的项目是一个可执行文件，您可以运行他，以确保它在您的系统上正常工作：

```
./automake_test
```

正常的话会看到输出’Hello，Automake！‘。

通过完成这些步骤，您已经创建了一个简单的测试项目，并使用 Automake 和 Autoconf 工具链成功配置、编译和运行它。这个过程可以帮助您验证 Automake 在您的系统上是否正常工作。如果一切顺利，您可以添加更多功能和文件来构建实际的项目。

```
[root@localhost automake_test]# ./configure
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a race-free mkdir -p... /usr/bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking whether make supports nested variables... yes
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether the compiler supports GNU C... yes
checking whether gcc accepts -g... yes
checking for gcc option to enable C11 features... none needed
checking whether gcc understands -c and -o together... yes
checking whether make supports the include directive... yes (GNU style)
checking dependency style of gcc... gcc3
checking that generated files are newer than configure... done
configure: creating ./config.status
config.status: creating Makefile
config.status: creating config.h
config.status: executing depfiles commands
[root@localhost automake_test]# make
make  all-am
make[1]: Entering directory '/root/automake_test'
gcc -DHAVE_CONFIG_H -I.     -g -O2 -MT main.o -MD -MP -MF .deps/main.Tpo -c -o main.o main.c
mv -f .deps/main.Tpo .deps/main.Po
gcc  -g -O2   -o automake_test main.o
make[1]: Leaving directory '/root/automake_test'
[root@localhost automake_test]# ./automake_test
Hello, Automake!
```

A 更新记录

发布日期	说明
2024-02-21	初稿