# WFS文件系统

WFS是基于Fuse框架制作的一个简易的文件系统，源代码行数仅数百行，目前的功能非常简陋，但已实现的部分源代码注释非常详尽，目的在于同学们根据这个文件系统的进一步扩展其功能，起到抛砖引玉的作用。

## 一、Fuse简介及开发环境的安装配置

FUSE (https://github.com/libfuse/libfuse) 是一个 Linux 内核扩展，它允许用户在用户空间实现一个文件系统。本课程设计要求大家使用 FUSE 实现一个文件系统。

FUSE的安装

首先，你需要在虚拟机上安装 ubuntu 20.04，然后在ubuntu上安装如下软件包：

sudo apt-get install git gcc vim lrzsz openssh-server meson pkg-config make unity-tweak-tool libtool m4 autoconf pkg-config

为了你的工作方便，将vscode也安装到你的ubuntu中。当然你如果喜欢vim+gdb的开发方式，也可以直接使用vim+gdb，不必安装vscode。

然后用git将 FUSE的最新源代码克隆到你的ubuntu本地：

git clone <https://github.com/libfuse/libfuse.git>



进入你的libfuse目录：

cd libfuse-master

mkdir build; cd build

meson ..

ninja

sudo ninja install

详见：<https://github.com/libfuse/libfuse>

### 二、第一个FUSE 例子

先看一下hello world例子。输入命令：

cd libfuse-master/examples

mkdir testmount

ls -al testmount

./hello testmount

ls -al testmount

你应该会看到3个目录项：. , .. 和 hello。为什么？

cat testmount/hello

你应该会看到 hello world。用你喜欢的IDE（比如vscode）看一下 hello.c 文件，或许你会得到答案。最后，将这个hello文件系统卸载：

fusermount -u testmount

FUSE High-level Description

上面运行的 hello 应用就是一个 FUSE 文件系统，演示了 FUSE的主要思想。

## 三、Building and Testing

下载WFS的源代码：

git clone https://gitee.com/my180mobile/wfs.git

进入WFS目录，直接make就可以编译wfs，编译成功后，将diskimg文件拷贝到/tmp目录下，打入命令：

./WFS tt

即可运行WFS，之后进入tt目录。

为了方便你的代码编写、阅读以及调试，建议使用vscode集成开发环境。注意vscode的launch.json改为：

{

// Use IntelliSense to learn about possible attributes.

// Hover to view descriptions of existing attributes.

// For more information, visit: https://go.microsoft.com/fwlink/?linkid=830387

"version": "0.2.0",

"configurations": [{

"name":"debug",

"type":"cppdbg",

"request": "launch",

"program": "${workspaceRoot}/WFS",

"cwd":".",

"args": ["-d","-s","tt"],

}]

}

Task.json改为：

{

"tasks": [

{

"type": "cppbuild",

"label": "C/C++: gcc-11 build active file",

"command": "gcc",

"args": [

"-fdiagnostics-color=always",

"-Wall",

"-D\_FILE\_OFFSET\_BITS=64",

"-g",

"-pthread",

"-lfuse3",

"-lrt",

"-ldl",

"${file}",

"-o",

"${fileDirname}/${fileBasenameNoExtension}"

],

"options": {

"cwd": "${fileDirname}"

},

"problemMatcher": [

"$gcc"

],

"group": {

"kind": "build",

"isDefault": true

},

"detail": "Task generated by Debugger."

}

],

"version": "2.0.0"

}