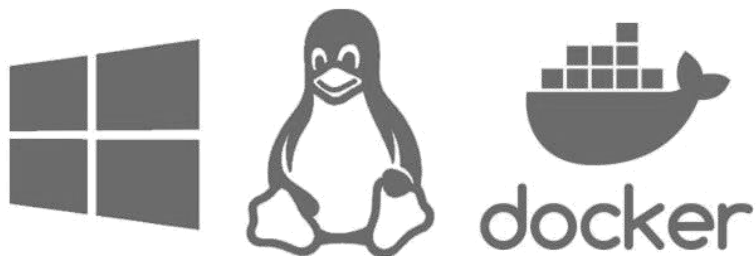


# 在Windows下使用WSL和Docker 配置脑科学分析与开发环境

王鑫迪



# Overview

- 使用Linux进行脑科学/机器学习数据分析的必要性
- WSL: Windows用户迁移使用Linux的最平滑解决方案
- WSL/WSLG/D3D12: 基于独立显卡使用WSL打开Linux下的图形界面
- WSL+Docker: 利用Docker创建自己的开发环境并持久化
- Docker/GPU/OpenGL: 在Docker中内置OpenGL, 实现图形渲染与GPU计算
- Docker/Cuda/Python/Matlab: 使用Docker配置用户自订制的集成开发环境

# 使用Linux进行脑科学/机器学习数据分析的必要性

- 大部分脑影像/脑电分析工具依托Linux开发，如FSL、AFNI、ANTs、FreeSurfer
- Linux下，R\Python等科学计算常用语言安装更为方便快捷
- 长期使用Linux，更利于掌握使用脚本语言进行编程，可以更为熟练地编写数据分析脚本
- Linux和Mac OS都是类UNIX系统，命令脚本类似
- 出国做研究，许多实验室因为未购买windows授权，仅配置Linux，且使用服务器进行数据分析时，也均为Linux系统
- 中美脱钩的大背景下，国内会使用Linux进行操作系统替代

# WSL：Windows用户迁移使用Linux的最平滑解决方案

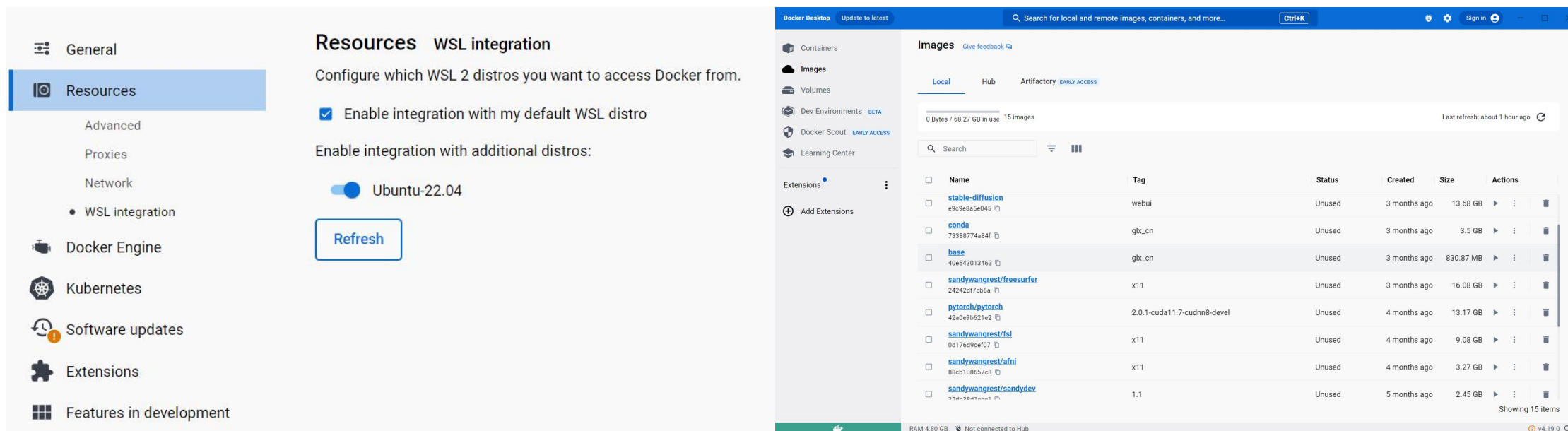
- WSL全称为Windows Subsystem for Linux，目前迭代到第二代（即WSL 2），是windows下源生的Linux虚拟机，效率与直接安装Linux系统类似，更易于使用
- 在“启动或关闭Windows功能”中，打开“适用于Windows的Linux子系统”和“虚拟机平台”，并重启电脑，即可开启WSL
- 打开PowerShell或CMD，运行`wsl --update`，更新wsl至最新版本
- 在Microsoft Store中，下载对应的Linux系统，即可完成安装，推荐安装Ubuntu 22.04 LTS版本

## WSL/WSLg/D3D12：基于独立显卡使用WSL打开Linux下的图形界面

- 最新版本的WSL，支持打开图形界面，即WSLg
- 运行`sudo apt-get update`
- 运行`sudo apt-get install mesa-utils gedit`
- 在`~/.bashrc`中添加“`export MESA_D3D12_DEFAULT_ADAPTER_NAME=NVIDIA`”，  
使用D3D12兼容层，更改默认渲染显卡为独立显卡
- 运行`sudo apt-get install mricron`，安装脑影像Viewer MRICroN

# WSL+Docker：利用Docker创建自己的开发环境并持久化

- 前往Docker官网，下载并安装Windows版本的Docker Desktop
- <https://www.docker.com/products/docker-desktop/>
- 在Docker Desktop中配置WSL Integration



# Docker/GPU/OpenGL：在Docker中内置OpenGL，实现图形渲染与GPU计算

- 在Docker中内置OpenGL支持、中文支持

```
FROM ubuntu:jammy
ARG DEBIAN_FRONTEND=noninteractive
ARG TZ="Asia/Shanghai"

# Install Base Lib
RUN apt-get update && \
    apt-get install -y --no-install-recommends \
        libgl1-mesa-dri=22.0.1-1ubuntu2 \
        libglx-mesa0=22.0.1-1ubuntu2 \
        libgl1-mesa-glx=22.0.1-1ubuntu2 \
        libglapi-mesa=22.0.1-1ubuntu2 \
        libgbm1=22.0.1-1ubuntu2 \
        libegl-mesa0=22.0.1-1ubuntu2 \
        libgl1-mesa-dri=22.0.1-1ubuntu2 \
        libglx-mesa0=22.0.1-1ubuntu2 \
```

```
        mesa-utils \
        qtbase5-dev \
        xauth \
        libnss3 \
        libpci-dev \
        libxcomposite1 \
        libxdamage1 \
        libxrandr2 \
        libxcursor1 \
        libxi6 \
        libxtst6 \
        libasound2 \
        language-pack-zh-hans fonts-droid-fallback ttf-wqy-zenhei \
        ttf-wqy-microhei fonts-arphic-ukai fonts-arphic-uming \
        mtrcron && \
    apt-mark hold libglapi-mesa libgbm1 libegl-mesa0 libgl1-mesa-dri libglx-mesa0 && \
    apt-get clean && rm -rf /var/lib/apt/lists/* /tmp/* /var/tmp/*
```

```
# Support zh_CN
RUN echo "LANG=\"zh_CN.UTF-8\"" >> /etc/environment && \
    echo "LANGUAGE=\"zh_CN:zh:en_US:en\"" >> /etc/environment && \
    echo "en_US.UTF-8 UTF-8" >> /var/lib/locales/supported.d/local && \
    echo "zh_CN.UTF-8 UTF-8" >> /var/lib/locales/supported.d/local && \
    echo "zh_CN.GBK GBK" >> /var/lib/locales/supported.d/local && \
    echo "zh_CN GB2312" >> /var/lib/locales/supported.d/local && \
    locale-gen
```

# Docker/Cuda/Python/Matlab：使用Docker配置用户自订制的集成开发环境

