

Operating Systems, Virtual Machine and Cloud computing

Dumrong Mairiang, PhD

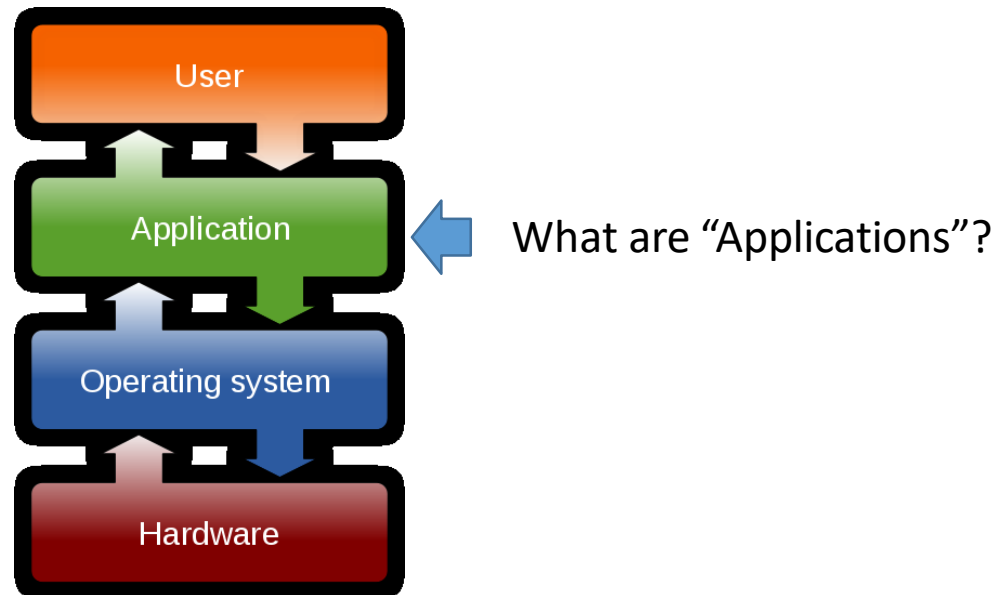
SIRE507: FUNDAMENTAL COMPUTER SCIENCE FOR BIOLOGIST

Operating system (OS)

- What is it?
- Why is it important?

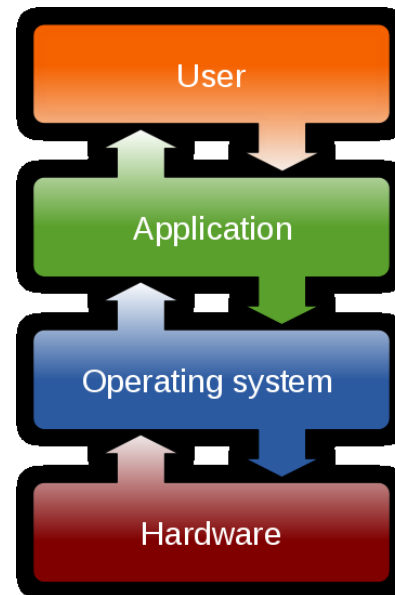
Operating system (OS)

- What is it?
 - The basic software that manages a computer
- Why is it important?



Operating system (OS)

- What is it?
 - The basic software that manages a computer
- Why is it important?



[Golftheman](#)



What are “Applications”?

Self-contained programs that perform a specific function



Operating system (OS)

- Do you know any OS?

Operating system (OS)

- Do you know any OS?



macOS

ubuntu 

The Ubuntu logo, which is an orange circle containing a white stylized gear or flower-like shape.

Operating system (OS)

- Do you know any OS?



Mobile OS

iOS



Operating system (OS)

- Do you know any OS?

Microsoft Windows family



Other



Unix/Unix-like family



Macintosh/Darwin



Linux



Why do you need to know about Oses other than MS Windows or MacOS?

- Many of bioinformatic applications were developed for Unix-like or Linux OS
- If you want to develop a web application, web servers usually run with Linux
- Many free and open-source applications must be run on Unix-like or Linux OS

Operating system (OS)

- Old machine, legacy application and IoT:
 - 32-bit vs 64-bit (32-bit apps on Windows?)
 - Lite version/distribution of OS (e.g. Alpine, Lubuntu)
 - AMD/intel vs ARM/Apple silicon
- Users of your application
 - Windows
 - General users with probably no or little bioinformatic background
 - Graphical user interface is likely to be expected
 - Unix-like/Linux
 - Bioinformaticians
 - Command lines are acceptable

Distinct features of each OS

- OS-specific file extensions
 - Microsoft Windows
 - File.exe
 - File.msi
 - Mac OS
 - File.dmg
 - Linux (Debian/Ubuntu)
 - File.deb

Distinct features of each OS

- File naming rules

- Microsoft Windows

- Reserved characters `< > : " / \ | ? *`

- Mac OS

- Reserved characters `/ :`

- Linux

- Reserved characters `/`

- Mac OS & Linux

- Non-reserved characters but can be problematic for a shell command

`< > : | ? *`

- As bioinformatician: Please AVOID using “space” in the file name

- FileName.txt (Camel case)
 - File_Name.txt (Underscore case)

Distinct features of each OS

- File paths to YourFile.txt in your “home” directory
 - Microsoft Windows
 - C:\Users\YourName\YourFile.txt
 - Mac OS
 - /Users/YourName/YourFile.txt
 - Linux
 - /home/YourName/YourFile.txt
- Linux (root access)
 - /root/YourFile.txt

Distinct features of each OS

- Application for accessing command line
 - Microsoft Windows
 - Command Prompt
 - Mac OS
 - Terminal
 - Linux
 - Terminal

Distinct features of each OS

- Application for Back-up
 - Microsoft Windows
 - System Restore and Restore Point
 - Mac OS
 - Time Machine
 - Linux
 - Ubuntu backups
 - Backups application (e.g. cronopete)

Distinct features of each OS

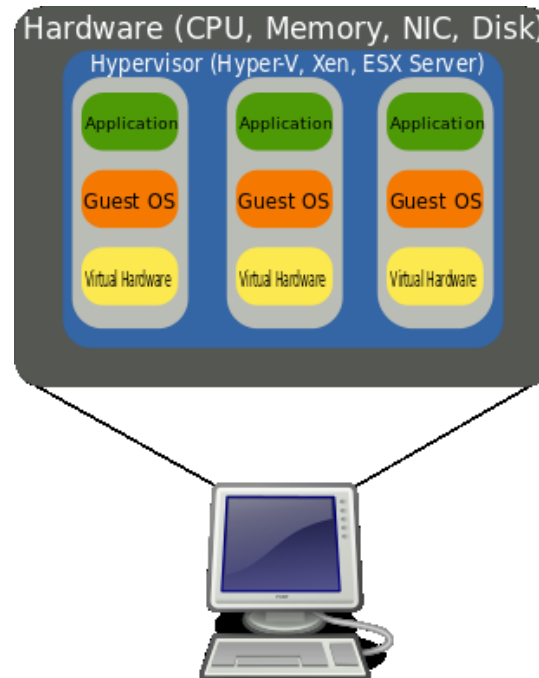
- Application for Hardware Management/List
 - Microsoft Windows
 - Device Manager
 - Mac OS
 - System Reports (“About this Mac”)
 - Linux (Ubuntu)
 - hardinfo, lspci, lsusb

Distinct features of each OS

- Checking or interrupting programs
 - Microsoft Windows
 - Task Manager
 - Mac OS
 - top (basic) and htop (to be installed)
 - Linux (Ubuntu)
 - top (basic) and htop (to be installed)

Virtual Machine (VM)

- What is it?
 - A complete environment for a guest operating system to function as though that operating system were installed on its own computer



Virtual Machine (VM)

- What VM is not:
 - Virtual machine \neq Emulator
 - Emulator converts commands to and from a host machine to an entirely different platform
 - Emulator: DosBOX (CPU), PuTTY (Terminal), ZSNES (Gaming), PCSX2 (Gaming), N64 Emulator (Gaming), Android Virtual Device (Application Development)

Why do you need to know about VMs?

- SENARIO 1 (Flexibility): The OS of your machine is not compatible with the application you need to run/test:
 - No spare machine for installing a new OS
 - No space or resources to create a dual boot

Why do you need to know about VMs?

- SENARIO 2 (Simulation): You want to simulate network connection to your web application in your own machine
 - No spare machine
 - Do not want to deploy in web server yet

Why do you need to know about VMs?

- SENARIO 3 (Security): Controlled/Quarantined environment for developing, testing or running applications
 - Potentially harmful applications
 - Secured VM

Basic information about VM

- Applications for virtualization:
 - VM ware (Commercial)
 - Oracle VM VirtualBox (Free and open-source)
 - Vagrant (Free and open-source, but no GUI)
 - Other...

Basic information about VM

- Check your machine

- CPU: at least 2 cores
- RAM: Host OS minimal requirement + Guest OS minimal requirement
- Hard disk: Guest OS minimal requirement or External HDD or USB Flash drive

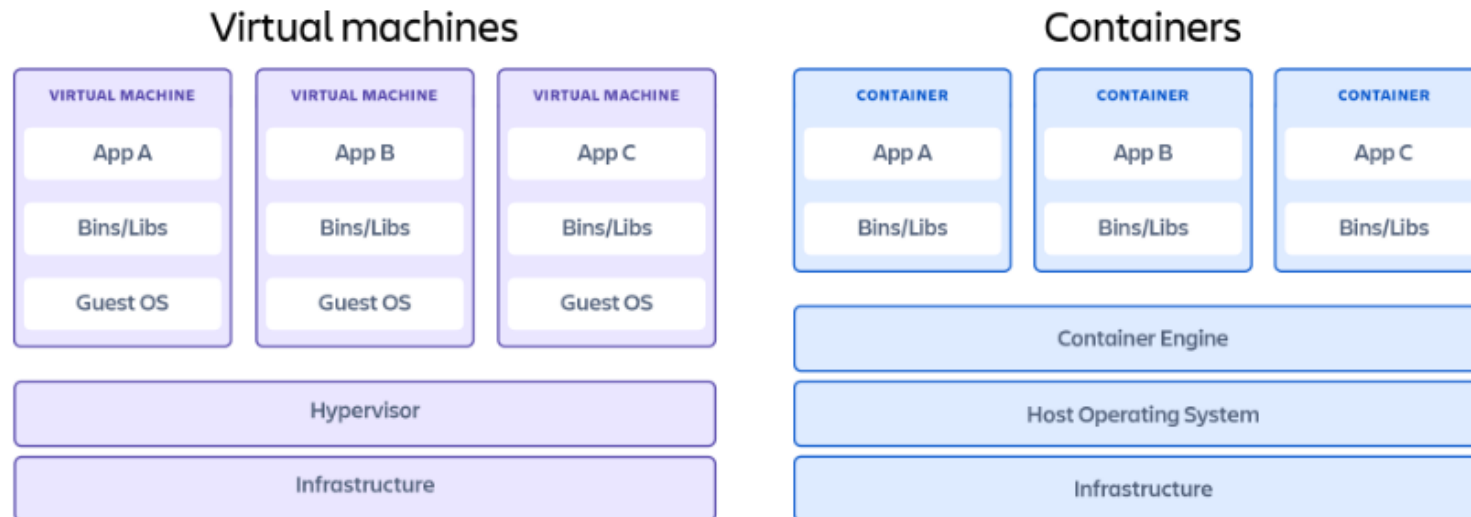
Portable VM

- You can save file.vdi to external HDD or USB flash drive
- Connect to any host machine with Virtualbox to quickly create a copy of VM in file.vdi
- **NOTE:**
 - USB flash drive must be formatted as NTFS
 - FAT32 format (default format in some USB flash drives) cannot accommodate a single file larger than 4GB

VM versus Container



- VM requires resources (e.g. RAM, CPU, HDD space) allocation while a container shares resources with the host.
- VM virtualizes an entire machine while a container virtualizes an “environment” for specific software.



VM versus Container



- Container use cases:
 - Run old software in new machine/OS
 - Software needs to run on 32-bit OS with Python 2
 - Deployment with reproducibility
 - Run your software developed on your machine on HPC
 - Microservices
 1. Container for database
 2. Container for web application
 3. Container for backup

VM versus Container

- **Docker** (<https://docs.docker.com/>)
 - Most popular container (i.e. good documentation)
 - Large repository of container images (DockerHub)
- **Singularity** (<https://docs.sylabs.io/guides/latest/user-guide/>)
 - “Portable” container (Singularity Image Format, SIF)
 - Other containers may be converted to Singularity container (especially Docker)
 - Run on HPC with job scheduler (e.g. Slurm)



Running Docker on Windows

<https://docs.docker.com/desktop/install/windows-install/>



The screenshot shows the Docker Docs website. The top navigation bar is blue with the Docker logo and links to Home, Guides, Manuals, Reference, FAQ, Samples, and Contribute. The breadcrumb trail indicates the current location: Home / Manuals / Docker Desktop / Install Docker Desktop / Install on Windows. The sidebar on the left lists various topics, with 'Install on Windows' highlighted. The main content area is titled 'Install Docker Desktop on Windows' and includes a blue button labeled 'Docker Desktop for Windows'. An orange arrow points to this button. To the right of the button, there are two bullet points: 'Install as guided' and 'Use WSL (check if WSL is installed)'. Below the button, there is a link to 'Release notes' for checksums. A light blue box contains 'Docker Desktop terms' and a note about commercial use. The 'System requirements' section has two tabs: 'WSL 2 backend' (selected) and 'Hyper-V backend and Windows containers'. The 'WSL 2 backend' section lists requirements for WSL version, Windows 11, and Windows 10.

Install Docker Desktop on Windows

This page contains the download URL, information about system requirements, and instructions on how to install Docker Desktop for Windows.

[Docker Desktop for Windows](#)

- Install as guided
- Use WSL (check if WSL is installed)

For checksums, see [Release notes](#)

Docker Desktop terms

Commercial use of Docker Desktop in larger enterprises (more than 250 employees OR more than \$10 million USD in annual revenue) requires a paid subscription.

System requirements

[WSL 2 backend](#) [Hyper-V backend and Windows containers](#)

WSL 2 backend

- WSL version 1.1.3.0 or later.
- Windows 11 64-bit: Home or Pro version 21H2 or higher, or Enterprise or Education version 21H2 or higher.
- Windows 10 64-bit: Home or Pro 21H2 (build 19044) or higher, or Enterprise or Education 21H2 (build 19044) or higher.

Turn on the WSL 2 feature on Windows. For detailed instructions, refer to the [Microsoft documentation](#).

Install Docker Desktop for MacOS

docs.docker.com

Command Types In Linux – vegibit

Install Docker Desktop on Mac | Docker Documentation

docker docs Search the docs Home Guides Manuals Reference FAQ Samples Contribute

Home / Manuals / Docker Desktop / Install Docker Desktop / Install on Mac

Docker Desktop

- Overview
- Install Docker Desktop
 - Install on Mac**
 - Understand permission requirements for Mac
 - Install on Windows
 - Understand permission requirements for Windows
 - Install on Linux
 - Installation per Linux distro
- Sign in
- Explore Docker Desktop
- Hardened Docker Desktop
- Dev Environments (Beta)
- containerd image store (Beta)
- Wasm workloads (Beta)

Install Docker Desktop on Mac

This page contains download URLs, information about system requirements, and instructions on how to install Docker Desktop for Mac.

Docker Desktop for Mac with Intel chip Docker Desktop for Mac with Apple silicon

For checksums, see [Release notes](#).

Docker Desktop terms

Commercial use of Docker Desktop in larger enterprises (more than 250 employees OR more than \$10 million USD in annual revenue) requires a paid subscription.

System requirements

Mac with Intel chip Mac with Apple silicon

Mac with Intel chip

- macOS must be version 11 or newer. That is Big Sur (11), Monterey (12), or Ventura (13). We recommend

Contents:

Page details

- 5 minute read
- Edit this page
- Request a change

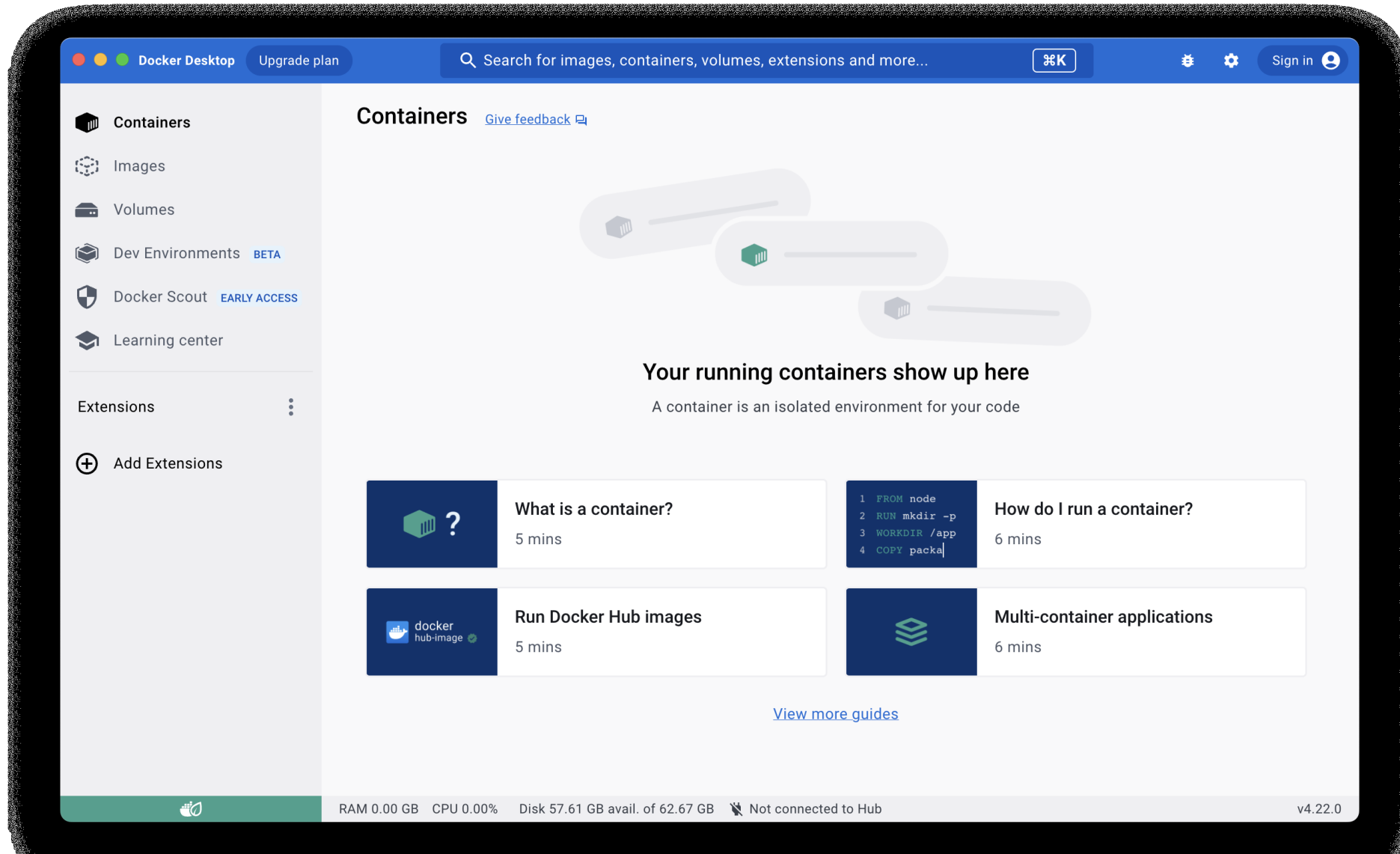
Contents

- System requirements
 - Mac with Intel chip
 - Mac with Apple silicon
- Install and run Docker Desktop on Mac
 - Install interactively
 - Install from the command line
- Where to go next

Follow installation instructions

Give feedback

Launch Docker Desktop



Run Docker in Terminal

A terminal window with a title bar showing 'kwan — root@855edc88cc25: / — zsh — 80x24'. The terminal content shows a Docker command being executed, followed by the output of 'cat /etc/os-release' which displays Ubuntu 22.04.3 LTS (Jammy Jellyfish) information. The session ends with 'exit' and returns to the host prompt.

```
[(base) kwan@MBPkhxnwanrutai ~ % docker run -it --rm ubuntu:latest /bin/bash  
root@855edc88cc25:/# cat /etc/os-release  
PRETTY_NAME="Ubuntu 22.04.3 LTS"  
NAME="Ubuntu"  
VERSION_ID="22.04"  
VERSION="22.04.3 LTS (Jammy Jellyfish)"  
VERSION_CODENAME=jammy  
ID=ubuntu  
ID_LIKE=debian  
HOME_URL="https://www.ubuntu.com/"  
SUPPORT_URL="https://help.ubuntu.com/"  
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"  
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"  
UBUNTU_CODENAME=jammy  
[root@855edc88cc25:/# exit  
exit  
(base) kwan@MBPkhxnwanrutai ~ %
```


Basic Docker commands

- `docker run -it --rm ubuntu:latest /bin/bash`
 - `docker run` → Create and run a new container from an image
 - `-it` → Interactive session
 - `--rm` → Automatically remove the container when it exits
 - `ubuntu:latest` → Image name and tag (i.e. version)
 - `/bin/bash` → Command to run within the container (/bin/bash = command prompt. Could be omitted.)

Basic Docker commands

- `docker run -it --rm -v /my/local/path:/data -w /data ubuntu:latest /bin/bash`
 - `-v /my/local/path:/data` → Mount /my/local/path to /data in the container
 - `-w` → Set /data of the container as the working directory

Basic Docker commands

- `docker ps` → Check all running instances
- `docker ps -a` → Check all running and exited instances
- `docker rm [CONTAINER ID]` → Remove an instance
- `docker images` → Check locally available images
- `docker rmi [IMAGE ID]` → Remove an image

Basic Docker commands

- `docker run -it -v /my/local/path:/data -w /data ubuntu:latest` → `exit` → Exited container (see instance ID with `docker ps -a`)
- `docker start -a -i [instance ID]` → Start and access an exited container (-a = attach to access the container; -i = interactive)

Dockerfile

The screenshot shows the GitHub interface for the `rocker-org/rocker` repository. The repository is public and has 277 forks and 1.4k stars. The current view is the `Dockerfile` for the `rocker/r-base/4.3.1` branch, committed by `eddelbuettel` 2 months ago. The Dockerfile is 70 lines long (59 loc) and 3.07 KB. The code is as follows:

```
1  ## Emacs, make this -*- mode: sh; -*-
2
3  FROM debian:testing
4
5  LABEL org.opencontainers.image.licenses="GPL-2.0-or-later" \
6        org.opencontainers.image.source="https://github.com/rocker-org/rocker" \
7        org.opencontainers.image.vendor="Rocker Project" \
8        org.opencontainers.image.authors="Dirk Eddelbuettel <edd@debian.org>"
9
10 ## Set a default user. Available via runtime flag `--user docker`
11 ## Add user to 'staff' group, granting them write privileges to /usr/local/lib/R/site.library
12 ## User should also have & own a home directory (for rstudio or linked volumes to work properly).
```

Build image from a Dockerfile

- `cd /path/to/Dockerfile`
- `docker build -t image_name:tag .` (Do not forget '.')

Simple docker file → Test building

```
# Simple Ubuntu container with htop and nano  
installed
```

```
FROM ubuntu:latest
```

```
LABEL AUTHOR=dumrong.mai@mahidol.ac.th
```

```
RUN apt-get update && apt-get install -y --no-  
install-recommends \
```

```
    nano \
```

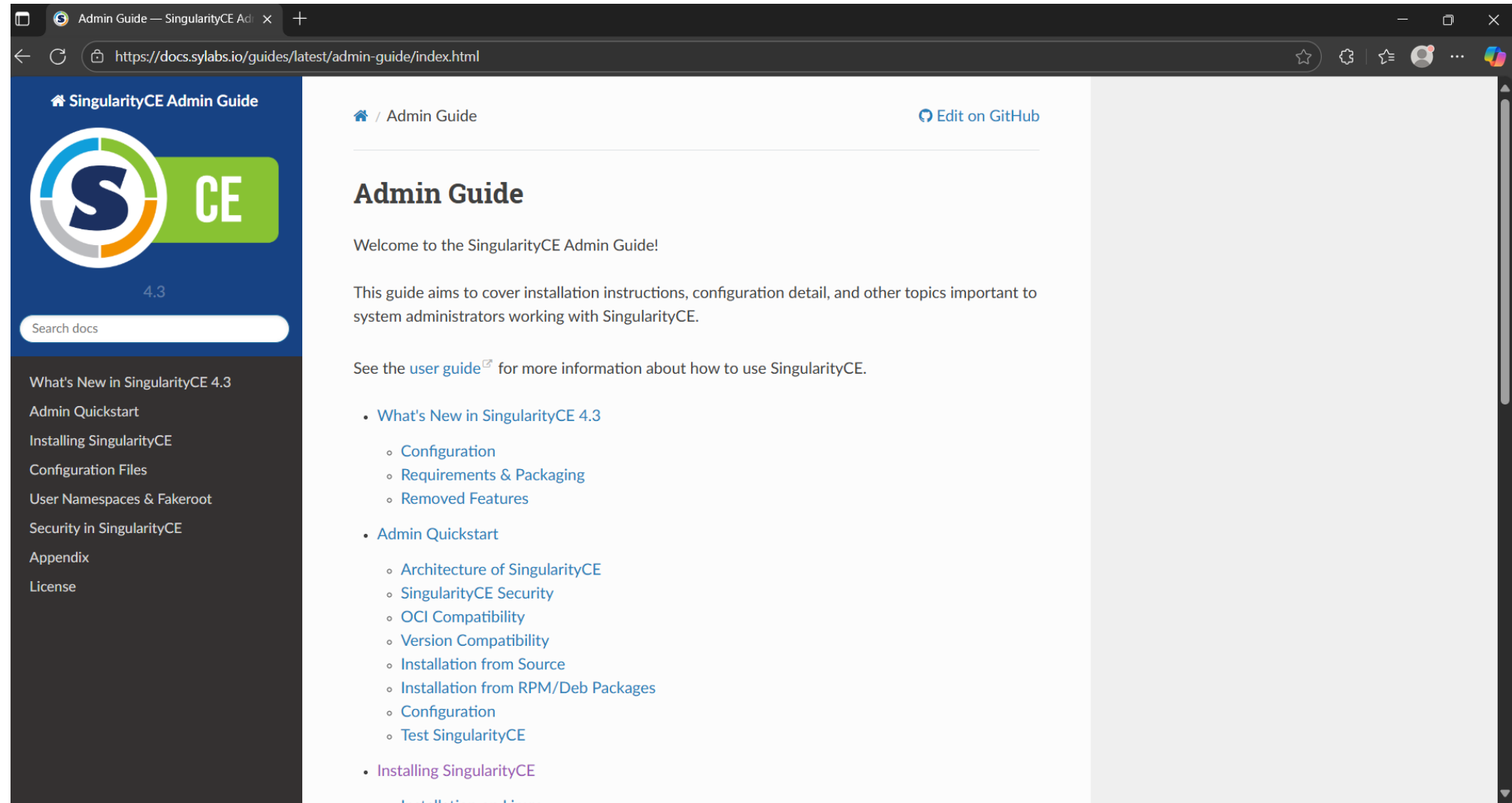
```
    htop
```

VM versus Container

- **Docker** (<https://docs.docker.com/>)
 - Most popular container (i.e. good documentation)
 - Large repository of container images (DockerHub)
- **Singularity** (<https://docs.sylabs.io/guides/latest/user-guide/>)
 - “Portable” container (Singularity Image Format, SIF)
 - Other containers may be converted to Singularity container (especially Docker)
 - Run on HPC with job scheduler (e.g. Slurm)



Singularity installation




The screenshot shows a web browser displaying the SingularityCE Admin Guide. The browser's address bar shows the URL `https://docs.sylabs.io/guides/latest/admin-guide/index.html`. The page has a dark blue sidebar on the left with the SingularityCE logo (a stylized 'S' in a circle) and the text 'SingularityCE Admin Guide' and '4.3'. Below the logo is a search bar labeled 'Search docs'. The sidebar lists the following navigation items: 'What's New in SingularityCE 4.3', 'Admin Quickstart', 'Installing SingularityCE', 'Configuration Files', 'User Namespaces & Fakeroot', 'Security in SingularityCE', 'Appendix', and 'License'. The main content area has a light gray background and contains the following text: 'Welcome to the SingularityCE Admin Guide!', 'This guide aims to cover installation instructions, configuration detail, and other topics important to system administrators working with SingularityCE.', and 'See the [user guide](#) for more information about how to use SingularityCE.' Below this text is a list of links: 'What's New in SingularityCE 4.3' (with sub-links for 'Configuration', 'Requirements & Packaging', and 'Removed Features'), 'Admin Quickstart' (with sub-links for 'Architecture of SingularityCE', 'SingularityCE Security', 'OCI Compatibility', 'Version Compatibility', 'Installation from Source', 'Installation from RPM/Deb Packages', 'Configuration', and 'Test SingularityCE'), and 'Installing SingularityCE'.

Admin Guide — SingularityCE Ad x +

← ↻ 🔒 `https://docs.sylabs.io/guides/latest/admin-guide/index.html` ☆ ⚙️ ☆ 👤 ... 🌈

🏠 SingularityCE Admin Guide

 4.3

Search docs

What's New in SingularityCE 4.3

Admin Quickstart

Installing SingularityCE

Configuration Files

User Namespaces & Fakeroot

Security in SingularityCE

Appendix

License

🏠 / Admin Guide [Edit on GitHub](#)

Admin Guide

Welcome to the SingularityCE Admin Guide!

This guide aims to cover installation instructions, configuration detail, and other topics important to system administrators working with SingularityCE.

See the [user guide](#) for more information about how to use SingularityCE.

- [What's New in SingularityCE 4.3](#)
 - [Configuration](#)
 - [Requirements & Packaging](#)
 - [Removed Features](#)
- [Admin Quickstart](#)
 - [Architecture of SingularityCE](#)
 - [SingularityCE Security](#)
 - [OCI Compatibility](#)
 - [Version Compatibility](#)
 - [Installation from Source](#)
 - [Installation from RPM/Deb Packages](#)
 - [Configuration](#)
 - [Test SingularityCE](#)
- [Installing SingularityCE](#)

Singularity installation (Windows)

Follow the [WSL2 installation instructions](#) to enable WSL2 with the default Ubuntu 22.04 environment. On Windows 11 and the most recent builds of Windows 10 this is as easy as opening an administrator command prompt or Powershell window and entering:

```
wsl --install
```

Follow the prompts. A restart is required, and when you open the 'Ubuntu' app for the first time you'll be asked to set a username and password for the Linux environment.

You can install SingularityCE from source, or from the Ubuntu packages at the GitHub releases page. To quickly install the 4.0.0 package use the following commands inside the WSL2 Ubuntu window:

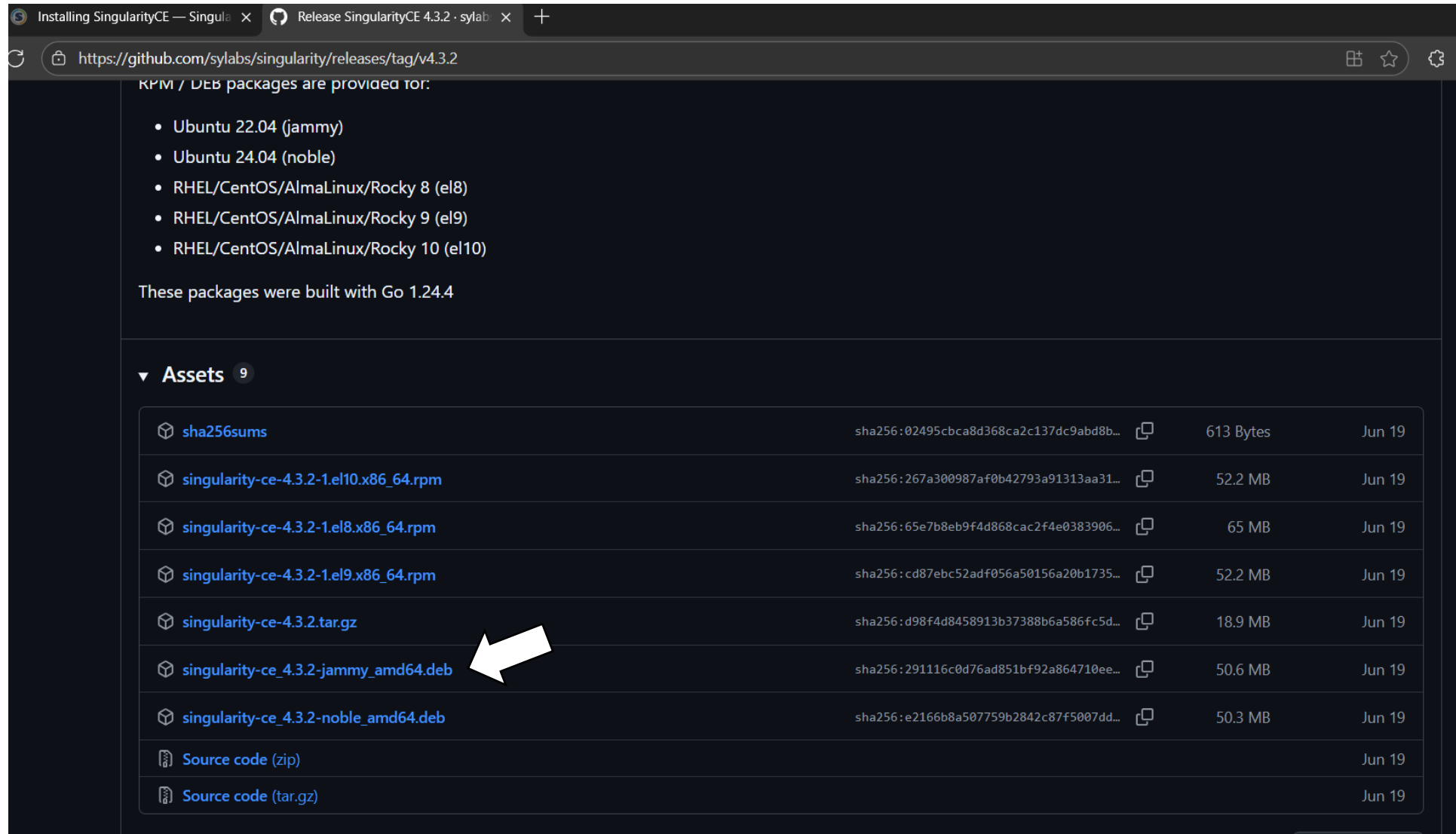
```
$ wget https://github.com/sylabs/singularity/releases/download/v4.0.0/singularity-ce_4.0.0-jammy
$ sudo apt install ./singularity-ce_4.0.0-jammy_amd64.deb
```

The `singularity` command will now be available in your WSL2 environment:

```
$ singularity exec library://ubuntu echo "Hello World!"
INFO:   Downloading library image
28.4MiB / 28.4MiB [=====]
Hello World!
```

- Run via WSL2
- Ubuntu on Windows (v20, Recommend v22)

Singularity installation (Windows)



Installing SingularityCE — Singula x Release SingularityCE 4.3.2 · sylab x +

https://github.com/sylabs/singularity/releases/tag/v4.3.2

RPM / DEB packages are provided for:

- Ubuntu 22.04 (jammy)
- Ubuntu 24.04 (noble)
- RHEL/CentOS/AlmaLinux/Rocky 8 (el8)
- RHEL/CentOS/AlmaLinux/Rocky 9 (el9)
- RHEL/CentOS/AlmaLinux/Rocky 10 (el10)

These packages were built with Go 1.24.4

▼ Assets 9

sha256sums	sha256:02495cbca8d368ca2c137dc9abd8b...	613 Bytes	Jun 19
singularity-ce-4.3.2-1.el10.x86_64.rpm	sha256:267a300987af0b42793a91313aa31...	52.2 MB	Jun 19
singularity-ce-4.3.2-1.el8.x86_64.rpm	sha256:65e7b8eb9f4d868cac2f4e0383906...	65 MB	Jun 19
singularity-ce-4.3.2-1.el9.x86_64.rpm	sha256:cd87ebc52ad056a50156a20b1735...	52.2 MB	Jun 19
singularity-ce-4.3.2.tar.gz	sha256:d98f4d8458913b37388b6a586fc5d...	18.9 MB	Jun 19
singularity-ce_4.3.2-jammy_amd64.deb	sha256:291116c0d76ad851bf92a864710ee...	50.6 MB	Jun 19
singularity-ce_4.3.2-noble_amd64.deb	sha256:e2166b8a507759b2842c87f5007dd...	50.3 MB	Jun 19
Source code (zip)			Jun 19
Source code (tar.gz)			Jun 19

Singularity installation (MacOS)

Mac

To install SingularityCE on macOS, we recommend using the [lima](#) VM platform, available on [Homebrew](#).

If you don't already have Homebrew installed, you can install it as follows:

```
$ /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Follow the instructions at the end of the installation process. In particular, make sure to add the relevant lines to your shell configuration:

```
$ (echo; echo 'eval "$(/home/linuxbrew/.linuxbrew/bin/brew shellenv)"') >> $HOME/.profile
$ eval "$(/home/linuxbrew/.linuxbrew/bin/brew shellenv)"
```

Once Homebrew is installed, install lima:

```
$ brew install lima
```

As part of the SingularityCE distribution (starting with version 4), we have provided an example template for using SingularityCE with lima. The example is available under the [examples/lima](#) directory in the SingularityCE source bundle, and can also be downloaded [directly from the code repository](#).

- Run via Lima (VM platform)

Simple SingularityCE definition file (test.def) → Test building

```
# Simple Ubuntu container with htop and nano  
installed
```

```
Bootstrap: docker
```

```
From: ubuntu:latest
```

```
%post
```

```
    apt-get -y update
```

```
    apt-get install -y --no-install-recommends \
```

```
    nano \
```

```
    htop
```

Simple SingularityCE definition file (xxx.def) → Test building

```
$ sudo singularity build test.sif test.def
```

```
$ singularity shell test.sif
```

```
Singularity> htop
```

```
Singularity> ls
```

Convert a local docker image to Singularity .sif file

```
$ sudo docker images
```

```
$ sudo singularity build docker_test.sif docker-  
daemon://docker_test:latest
```

Writable sandbox directories

- Anything modified in the sandbox is kept
 - Docker or SIF files do not save any modification
- Sandbox is a directory, not a file
 - Lose portability <-> Gain modification
- Definition file or Dockerfile is recommended for reproducibility

```
$ sudo singularity build --sandbox lolcow/ library://lolcow  
$ sudo singularity shell --writable lolcow/  
$ sudo singularity build lolcow.sif lolcow/
```


Basic Singularity commands

The shell command allows you to spawn a new shell within your container and interact with it as though it were a virtual machine.

```
$ singularity shell lolcow_latest.sif
```

The exec command allows you to execute a custom command within a container by specifying the image file.

```
$ singularity exec lolcow_latest.sif cowsay moo
```

#SingularityCE containers contain runscripts. These are user-defined scripts that define the actions a container should perform when someone runs it.

```
$ singularity run lolcow_latest.sif
```

Bind paths in SingularityCE (mount in Docker)

- A Singularity container has an access to host's current working directory unless an admin disables the access.
- You may bind a different directory to the container with a following command:

```
$ singularity shell --bind /data:/mnt  
my_container.sif
```

- Use cases: HPC, Cloud VM with a persistent storage volume

Portable SIF

- You can save file.sif to external HDD or USB flash drive
- Connect to any host machine with Singularity installed to quickly run a Singularity container
- **NOTE:**
 - USB flash drive must be formatted as NTFS
 - FAT32 format (default format in some USB flash drives) cannot accommodate a single file larger than 4GB

Cloud computing

- What is it?
- Cloud computing, often referred to as simply “the cloud,” is the delivery of on-demand computing resources — everything from applications to data centers — over the internet on a pay-for-use basis.
 - Elastic resources — Scale up or down quickly and easily to meet demand
 - Metered service so you only pay for what you use
 - Self service — All the IT resources you need with self-service access








IBM **Cloud**

<https://www.ibm.com/cloud/learn/what-is-cloud-computing>

Why do you need to know Cloud Computing?

- Web application deployment
- Requiring intensive but transient computing power
- Backup and archive
- Sharing and hosting (Public cloud)
- Anything else?

Cloud service providers

- Amazon web services The AWS logo consists of the lowercase letters "aws" in a dark blue, sans-serif font, with an orange curved arrow underneath that starts under the 'a' and points towards the 's'.
- Google cloud platform The Google Cloud Platform logo features a hexagonal icon composed of six smaller hexagons in blue, red, yellow, and grey, followed by the text "Google Cloud Platform" in a bold, black, sans-serif font.
- Microsoft Azure The Azure logo features a stylized blue triangle icon to the left of the word "Azure" in a blue, sans-serif font.
- IBM cloud The IBM Cloud logo features a blue icon of a cloud with three short vertical lines above it, followed by the text "IBM Cloud" in a bold, black, sans-serif font.
- Digital Ocean The DigitalOcean logo features a blue circular icon with a stylized 'd' shape inside, followed by the text "DigitalOcean" in a bold, blue, sans-serif font.

Hands on: Cloud Computing

Overview

- Demo for creating a cloud VM
- Practice: remotely access the VM
- Practice: remotely transfer files from/to the VM
- Back-up and clean-up

Creating a VM

- Platform will be based on “your advisor”
 - Digital ocean (Simple Unix/Linux applications, web application)
 - Google Cloud Platform/AWS (Machine learning, NextFlow)
 - IPGG server/MU cluster/Thai-SC (Intensive calculation with CPUs and RAM)

Creating a VM

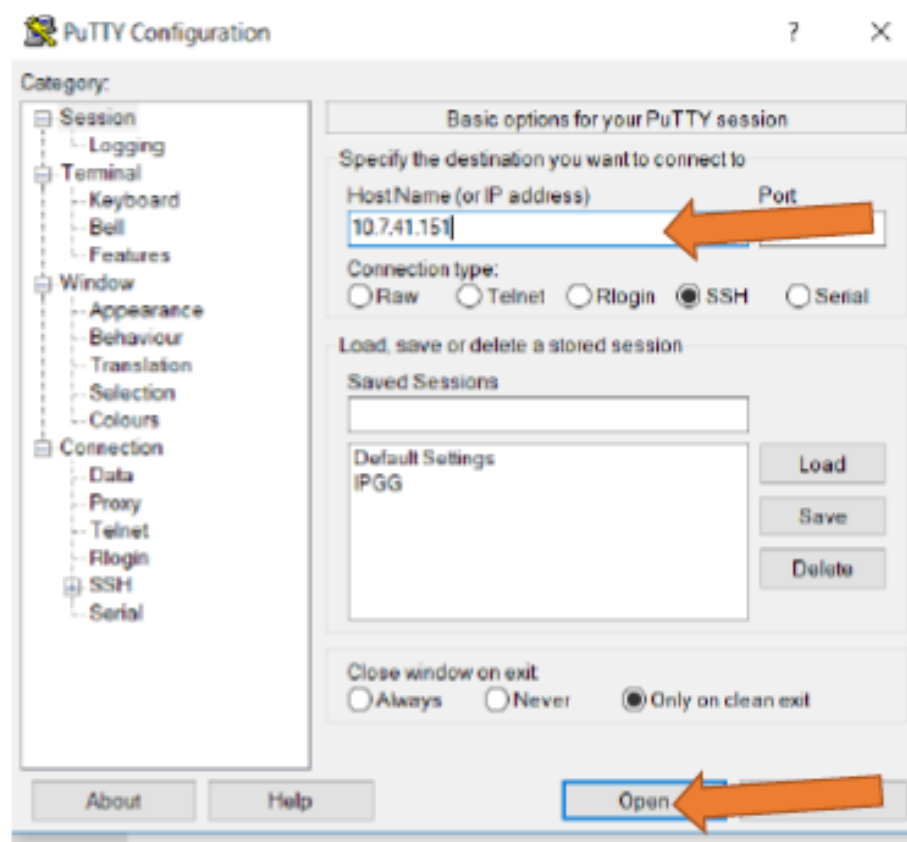
- Calculate/Estimate your resource wisely
 - CPU/RAM/HDD
 - OS
 - Location of your data center
 - Safety features

Creating a VM on



Remotely access the VM (or server)

- Windows
 - Require a software
 - Install “putty”
 - Login as ‘root’



Remotely access the VM (or server)

- Windows
 - Alternatively via Ubuntu on Windows (with WSL2)

Remotely access the VM (or server)

- Mac and Linux
 - Use your 'terminal'
 - `ssh root@<VM IP address>`

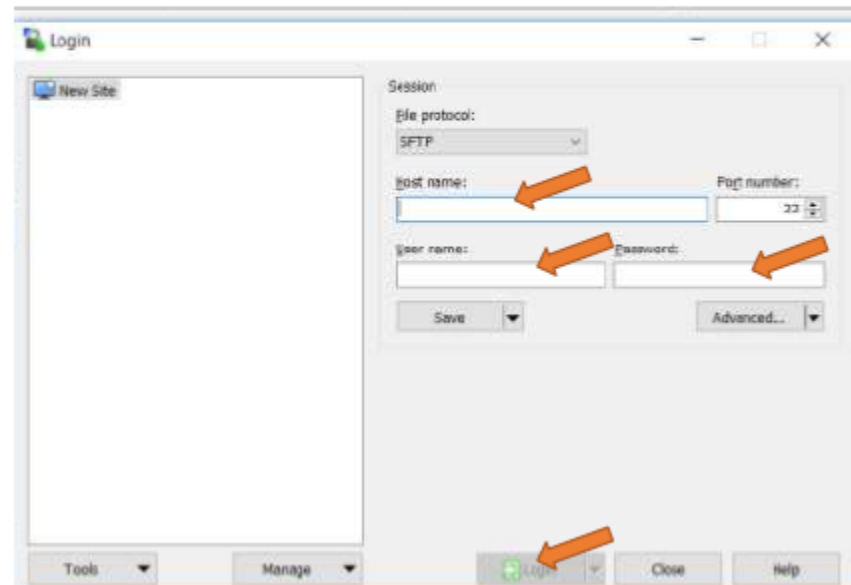
Test basic Unix/Linux commands

- `ls`
- `pwd`
- `touch`
- `cp`
- `mv`
- `rm`
- `mkdir`
- `cd`
- `*nano*`

Remotely transfer files from/to the VM

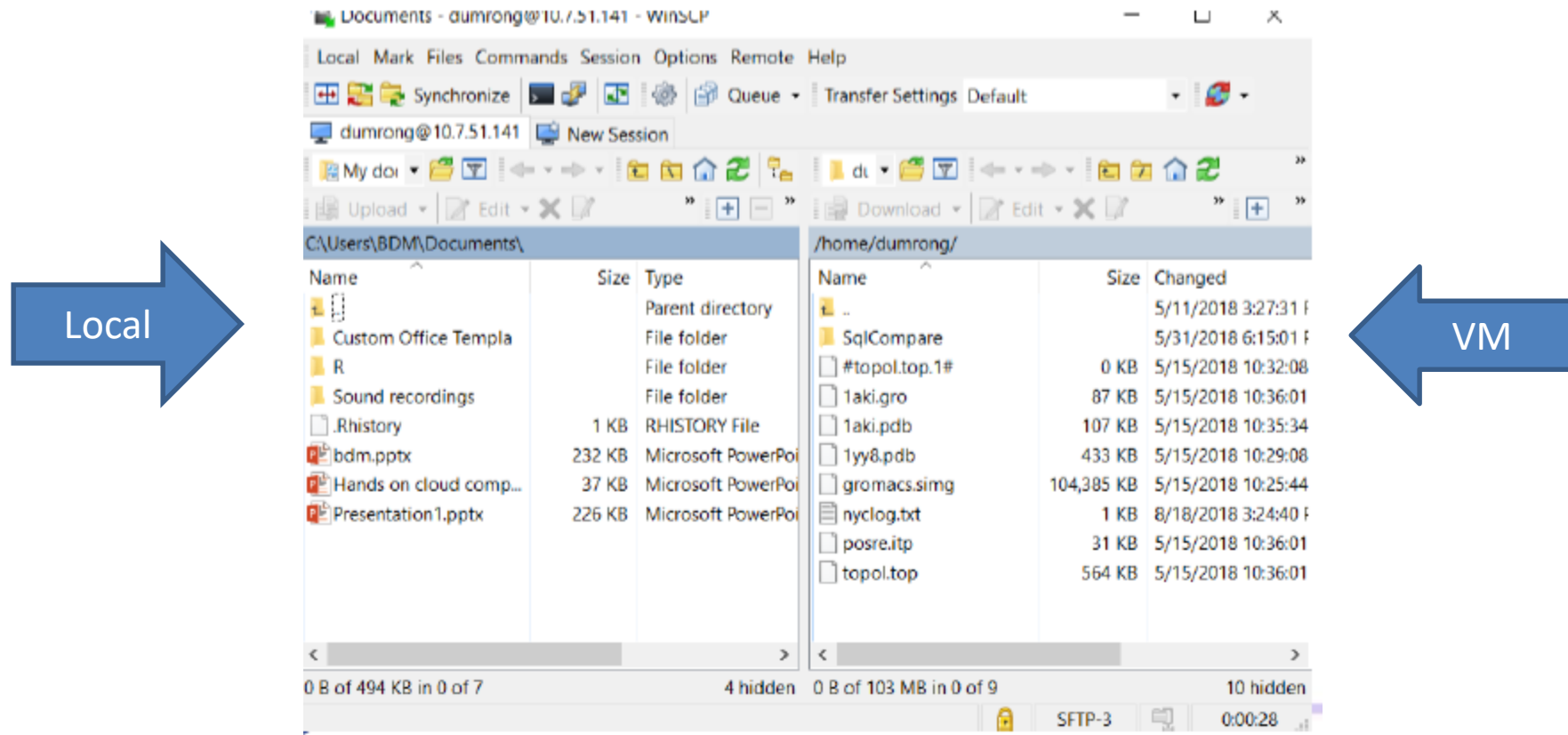
- Windows
 - Require a software
 - Install “WinSCP”

- Windows user: Open WinSCP → enter IP Address
→ Enter username (‘root’) and password → click “Login”



Remotely transfer files from/to the VM

- Windows
 - Require a software
 - Install “WinSCP”



Remotely transfer files from/to the VM

- Mac and Linux
 - Use your 'terminal'
 - Download:

```
scp root@<VM IP address>:/path/to/file /local/path
```

- Upload:

```
scp /local/path/to/file root@<VM IP address>:/path/
```

Remotely transfer files from/to the VM

- Windows
 - Alternatively via Ubuntu on Windows (with WSL2)
 - Command lines

- **Download:**

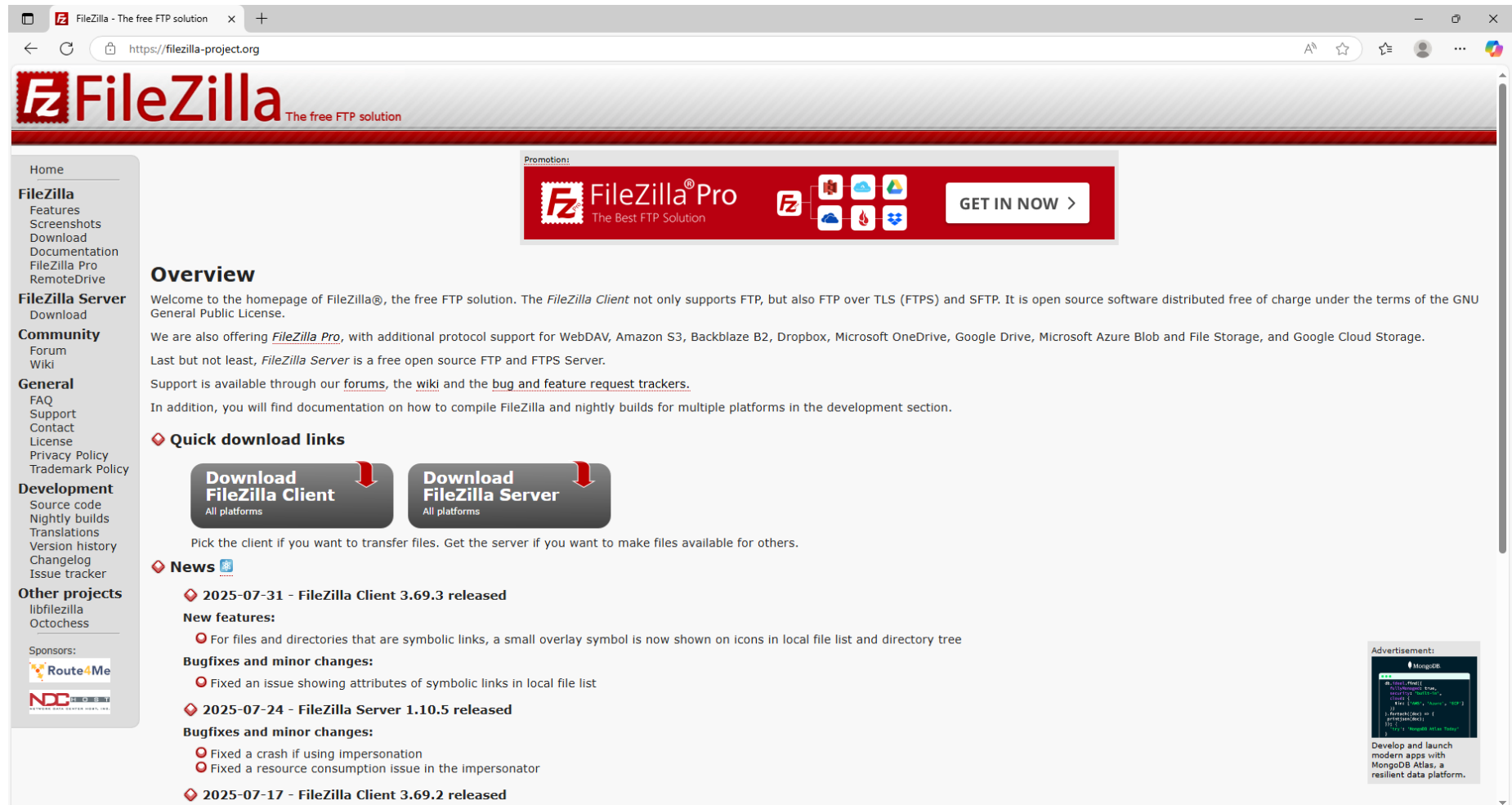
```
scp root@<VM IP address>:/path/to/file /local/path
```

- **Upload:**

```
scp /local/path/to/file root@<VM IP address>:/path/
```

Remotely transfer files from/to the VM

- All platforms



The screenshot shows the FileZilla website homepage. The browser address bar displays 'https://filezilla-project.org'. The page features a prominent red header with the FileZilla logo and the tagline 'The free FTP solution'. A navigation sidebar on the left lists various sections: Home, FileZilla (with sub-links for Features, Screenshots, Download, Documentation, FileZilla Pro, and RemoteDrive), FileZilla Server (with a Download link), Community (with links to Forum and Wiki), General (with links to FAQ, Support, Contact, License, Privacy Policy, and Trademark Policy), Development (with links to Source code, Nightly builds, Translations, Version history, Changelog, and Issue tracker), and Other projects (with links to libfilezilla and Octochess). Below the sidebar, there are 'Sponsors' logos for RouteMe and NDC. The main content area includes a 'Promotion' banner for 'FileZilla Pro' with a 'GET IN NOW >' button. Below this is an 'Overview' section with a welcome message and information about the FileZilla Client and Server. A 'Quick download links' section offers buttons to 'Download FileZilla Client' and 'Download FileZilla Server', both for 'All platforms'. A 'News' section lists recent releases: '2025-07-31 - FileZilla Client 3.69.3 released' with new features and bugfixes, '2025-07-24 - FileZilla Server 1.10.5 released' with bugfixes, and '2025-07-17 - FileZilla Client 3.69.2 released'. An advertisement for MongoDB is visible in the bottom right corner.

FileZilla - The free FTP solution

https://filezilla-project.org

FileZilla The free FTP solution

Home

FileZilla

- Features
- Screenshots
- Download
- Documentation
- FileZilla Pro
- RemoteDrive

FileZilla Server

- Download

Community

- Forum
- Wiki

General

- FAQ
- Support
- Contact
- License
- Privacy Policy
- Trademark Policy

Development

- Source code
- Nightly builds
- Translations
- Version history
- Changelog
- Issue tracker

Other projects

- libfilezilla
- Octochess

Sponsors:

- RouteMe
- NDC

Promotion:

FileZilla Pro The Best FTP Solution

GET IN NOW >

Overview

Welcome to the homepage of FileZilla®, the free FTP solution. The *FileZilla Client* not only supports FTP, but also FTP over TLS (FTPS) and SFTP. It is open source software distributed free of charge under the terms of the GNU General Public License.

We are also offering *FileZilla Pro*, with additional protocol support for WebDAV, Amazon S3, Backblaze B2, Dropbox, Microsoft OneDrive, Google Drive, Microsoft Azure Blob and File Storage, and Google Cloud Storage.

Last but not least, *FileZilla Server* is a free open source FTP and FTPS Server.

Support is available through our [forums](#), the [wiki](#) and the [bug and feature request trackers](#).

In addition, you will find documentation on how to compile FileZilla and nightly builds for multiple platforms in the development section.

Quick download links

Download FileZilla Client All platforms

Download FileZilla Server All platforms

Pick the client if you want to transfer files. Get the server if you want to make files available for others.

News

2025-07-31 - FileZilla Client 3.69.3 released

New features:

- For files and directories that are symbolic links, a small overlay symbol is now shown on icons in local file list and directory tree

Bugfixes and minor changes:

- Fixed an issue showing attributes of symbolic links in local file list

2025-07-24 - FileZilla Server 1.10.5 released

Bugfixes and minor changes:

- Fixed a crash if using impersonation
- Fixed a resource consumption issue in the impersonator

2025-07-17 - FileZilla Client 3.69.2 released

Advertisement:

MongoDB

```
const { MongoClient } = require('mongodb');
const uri = 'mongodb://localhost:27020';
const client = new MongoClient(uri, { useNewUrlParser: true, useUnifiedTopology: true });
client.connect().then(() => {
  console.log('Connected to MongoDB');
}).catch(err => console.error(err));
```

Develop and launch modern apps with MongoDB Atlas, a resilient data platform.

Practice file transfer (to VM)

1. Open notepad/text editor in your computer
2. Write "Thank you." in a new text file
3. Save the file as `thank.txt`
4. Upload this file to your VM (WinSCP or scp)
5. Access your VM (PuTTY or ssh)
6. Type `ls` see if you can find your file
7. Type `cat thank.txt` to read the content in the file

Practice file transfer (from VM)

1. Access your droplet (PuTTY or ssh)
2. Type `echo "you are welcome" > yaw.txt`
3. Connect to your VM with WinSCP (or use scp)
4. Download `yaw.txt` to you computer
5. Use notepad or word to read the file

Back-up and clean-up

- Back-up with 'snapshot':
 - Freeze everything in your VM at the time of the snapshot.
 - The snapshot is used to instantly restore the VM.
 - Use for:
 - Routine back-up
 - Temporary paused VM
 - Cheaper than let the VM run. (Snapshot is **NOT** free!)

Back-up and clean-up

- Clean-up:
 - Running and power-off VMs cost the same!
 - **Everything** must be destroyed to stop the cloud provider from charging you.
 - Snapshots and volumes cost money!
 - Clean-up after:
 - Finish the project
 - Results and scripts are downloaded to local machines
- After a VM is destroyed, data in the VM are gone for good!