

Linux Container

國立臺灣師範大學物理學系 陳俊明

chunming@@ntnu.edu.tw

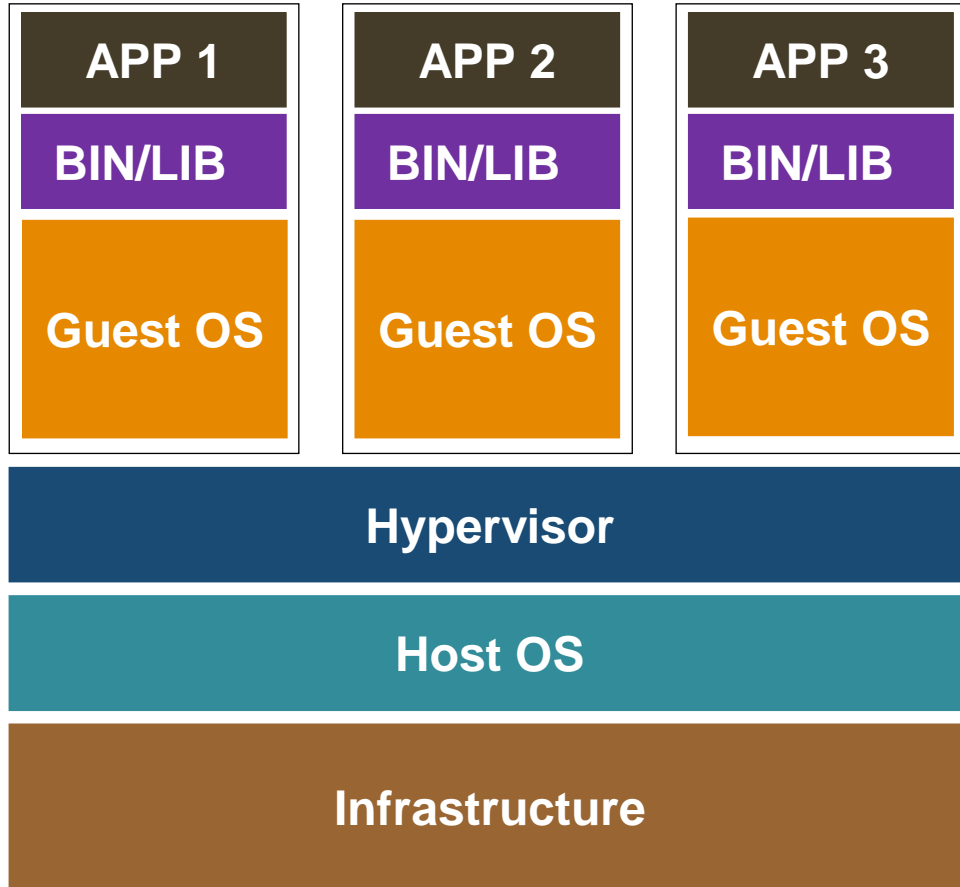
什麼是容器 (Containers) ?

Container 是作業系統輕量級虛擬化技術，用來封裝應用程式和應用程式所依存的函示庫於獨立環境。容器提供標準化輕量級封裝及部署程式的方式，所以可以擺脫基礎設施 (Infrastructure) 的差異在不同地方進行執行。

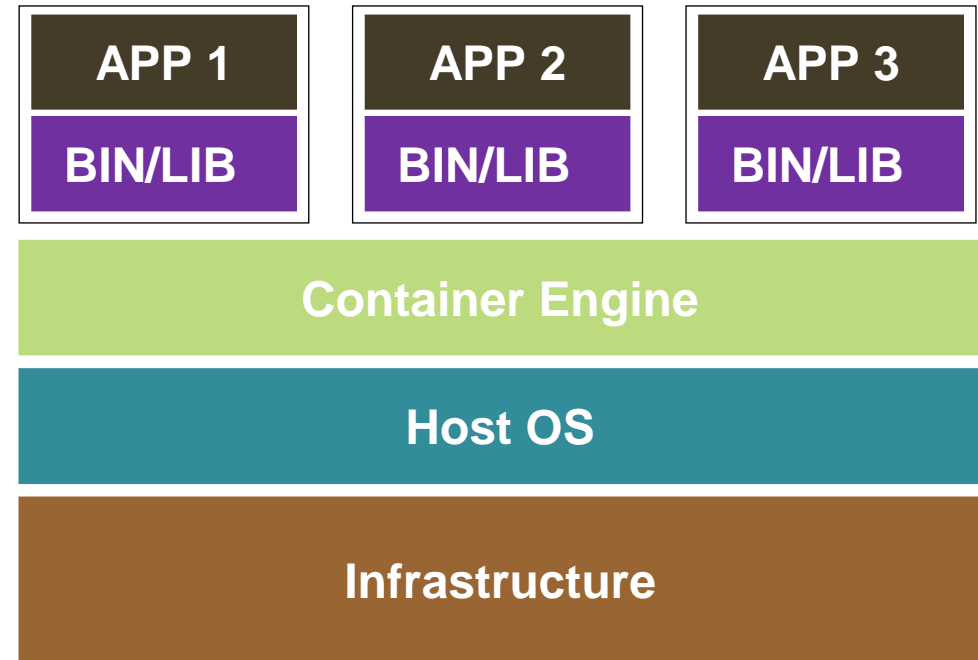
早在 1982 年，Unix 內建的 chroot 機制就是一種 Container 的技術，但直到 2013 年，dotCloud 這家公司釋出了一套將 Container 標準化的套裝平台 Docker，才真正開始了 Container 的發展。

目前市面上所有主流的Linux發行版本(如RedHat、CentOS、Ubuntu、SLES、OpenSUSE)、虛擬化的龍頭VMware、主流的雲端服務平台(如Amazon AWS、Microsoft Azure、Google Cloud)以及開放原始碼的雲端架構OpenStack 都支援Docker。

Container VS VM



Virtual Machine



Container

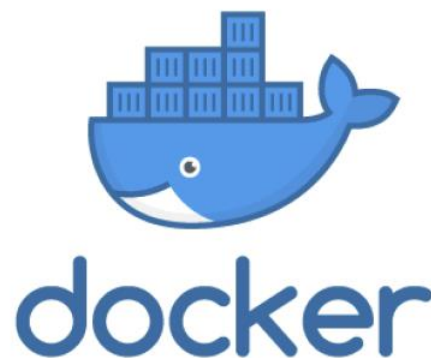
使用 Container 原因

- 快速部署：可更快速的更換版本，且不影響使用者操作
- 適合開發：方便更換不同函式庫或環境版本，可以方便開發者在容器內因應不同環境進行開發
- 使用環境差異大：不同的使用者需要不同環境或作業系統
- 方便移動：封裝好的映像檔 (image) 可以移動套不同平台環境上使用
- 資源充分使用：提高使用效率且容器執行效能直逼裸機效能
- 節省資金：搭配資源管理把忙碌的節點容器進行轉移，可提高機器運轉率

Docker

目前的Container標準，所有主流的Container技術都依據Docker而來。

「就像是貨櫃允許利用船隻、火車或貨車運輸貨物，而不論內含哪種貨物，軟體容器是軟體部署的標準單位，可包含不同的程式碼和相依性。以此方式容器化軟體可讓開發人員和 IT 專業人員只需要一點修改或不需要任何修改，就能跨環境進行部署。」節錄於[微軟容器和 Docker 簡介](#)



Docker Hub



官方網頁：<https://hub.docker.com/>

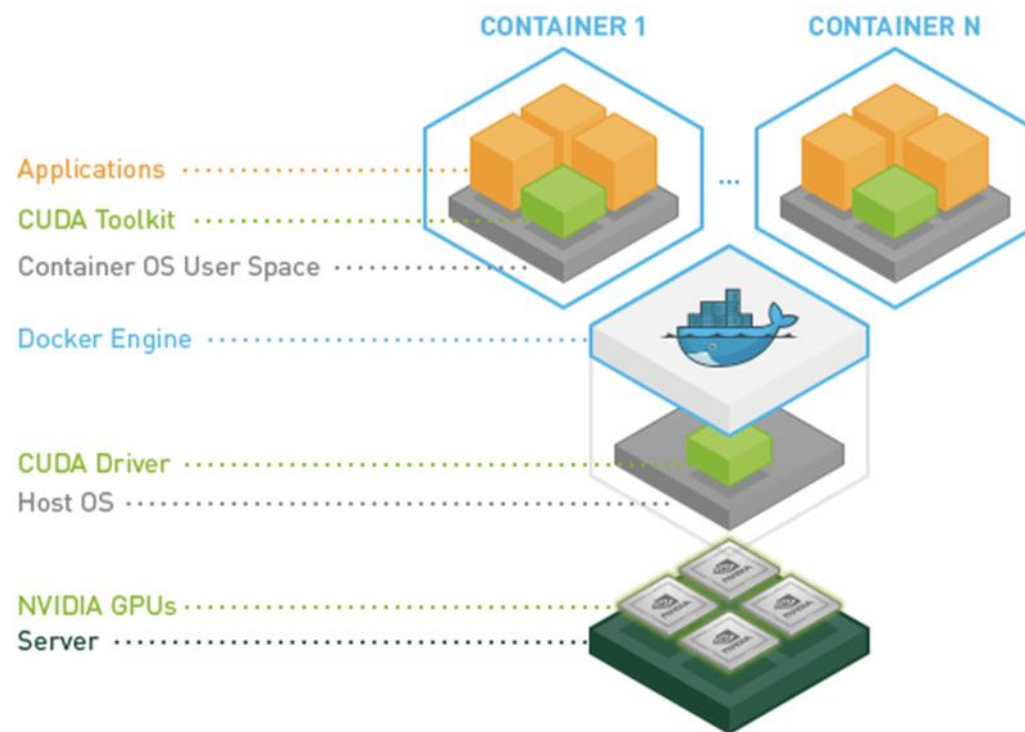
A screenshot of the Docker Hub search results page. The page has a blue header with the Docker Hub logo, a search bar containing "Search for great content (e.g., mysql)", and navigation links for "Explore", "Pricing", "Sign In", and "Register". Below the header, the search results are displayed. On the left, there are filter sections for "Products" (Images, Extensions, Plugins), "Trusted Content" (Docker Official Image, Verified Publisher, Sponsored OSS), "Operating Systems" (Linux, Windows), and "Architectures" (ARM, ARM 64, IBM POWER, IBM Z, PowerPC 64 LE, x86, x86-64). The main content area shows "1 - 25 of 9,460,420 available results." and a "Suggested" dropdown. The search results list several Docker Official Images: 1. ubuntu: Updated 8 days ago, 1B+ Downloads, 10K+ Stars. Description: "Ubuntu is a Debian-based Linux operating system based on free software." Architectures: Linux, IBM Z, 386, x86-64, ARM, ARM 64, PowerPC 64 LE, riscv64. 2. alpine: Updated 19 hours ago, 1B+ Downloads, 9.1K Stars. Description: "A minimal Docker image based on Alpine Linux with a complete package index and only 5 MB in size!" Architectures: Linux, IBM Z, riscv64, x86-64, ARM, ARM 64, 386, PowerPC 64 LE. 3. nginx: Updated 5 hours ago, 1B+ Downloads, 10K+ Stars. Description: "Official build of Nginx." Architectures: Linux, IBM Z, x86-64, ARM, ARM 64, 386, mips64le, PowerPC 64 LE. 4. busybox: Updated 9 days ago, 1B+ Downloads, 2.7K Stars. Description: "Busybox base image." Architectures: Linux, riscv64, IBM Z, x86-64, ARM, ARM 64, 386, mips64le, PowerPC 64 LE. 5. redis: Updated 6 hours ago, 1B+ Downloads, 10K+ Stars. Description: "Redis key-value store." Architectures: Linux, IBM Z, x86-64, ARM, ARM 64, 386, mips64le, PowerPC 64 LE.

Docker CE 安裝與測試

```
[root@master ~]# yum install -y yum-utils device-mapper-persistent-data lvm2
[root@master ~]# yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo
[root@master ~]# yum install -y docker-ce
[root@master ~]# systemctl start docker
[root@master ~]# systemctl enable docker
[root@master ~]# docker run hello-world
[root@master ~]# groupadd docker
[root@master ~]# usermod -aG docker $USER

[user1 @master ~]$ docker run -it ubuntu bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
d19f32bd9e41: Pull complete
Digest: sha256:34fea4f31bf187bc915536831fd0afc9d214755bf700b5cdb1336c82516d154e
Status: Downloaded newer image for ubuntu:latest
root@782697d50831:/# cat /etc/os-release
```

NVIDIA Container Toolkit



<https://docs.nvidia.com/datacenter/cloud-native/container-toolkit/overview.html>

NVIDIA Container Toolkit 安裝與測試範例

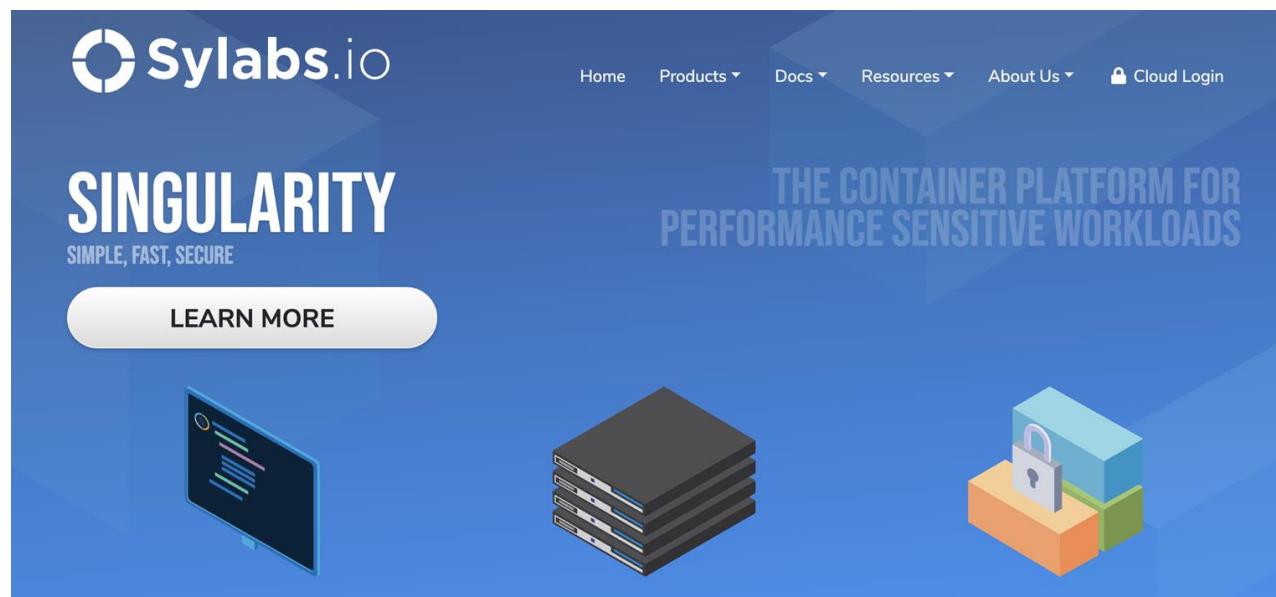
```
[root@master ~]# distribution=$(cat /etc/os-release;echo $ID$VERSION_ID)
[root@master ~]# curl -s -L https://nvidia.github.io/nvidia-docker/$distribution/nvidia-docker.repo | sudo tee /etc/yum.repos.d/nvidia-docker.repo
[root@master ~]# yum install -y nvidia-container-toolkit
[root@master ~]# systemctl restart docker

[user1@master ~]$ docker run --gpus all nvidia/cuda:10.0-base nvidia-smi
[user1@master ~]$ docker run --gpus 2 nvidia/cuda:10.0-base nvidia-smi
[user1@master ~]$ docker run --gpus '"device=1,2"' nvidia/cuda:10.0-base nvidia-smi
```

其他作業系統安裝參考 <https://nvidia.github.io/nvidia-docker/>

Singularity

- 以 Docker 為基礎進行改良的 Container 引擎，簡化了 Docker 龐大且複雜的指令以及加強了安全性。



<https://sylabs.io/>

Singularity 安裝

- 安裝系統依賴軟體

```
[root@master ~]# yum groupinstall -y 'Development Tools'  
[root@master ~]# yum install -y epel-release  
[root@master ~]# yum install -y openssl-devel libuuid-devel libseccomp-devel squashfs-tools
```

- 安裝golang套件

```
[root@master ~]# yum install -y golang
```

Singularity 安裝

- 下載Singularity

<https://github.com/apptainer/singularity> 檢查發布版本，下載原始碼

```
[root@master ~]# git clone --recurse-submodules https://github.com/sylabs/singularity.git -b "release-3.11"
```

- 安裝Singularity

```
[root@master ~]# cd singularity
[root@master ~]# ./mconfig --prefix=/opt/singularity --sysconffdir=/opt/singularity
[root@master ~]# cd builddir
[root@master ~]# make
[root@master ~]# make install
```

- 設定Singularity環境

```
[root@master ~]# export PATH=/opt/singularity/bin:$PATH
```

Singularity 範例

- Docker 映象檔

<https://cloud.sylabs.io/>



Singularity Container Services

Singularity Container Services makes containerization easy. Use it to build, share, and secure your performance intensive applications

Search Cloud Library for Public Images

```
$ singularity build -r library://demo/demo/alpine docker://alpine
INFO: Starting build...
INFO: Creating SIF file...
2.7MiB / 2.7MiB [=====] 100 % 33.1 MiB/s 0s
INFO: Build complete: library://demo/demo/alpine
```



Get Started Today for Free

Already have an account? [Sign In](#)

[Sign Up](#)

Get started with our curated collection of official base images



Singularity 範例

- 使用7.9版本的 CentOS
 - Singularity 的影像檔

```
[root@master ~]# singularity pull library://library/default/centos:7
```

- Docker 的影像檔

```
[root@master ~]# singularity pull docker://centos:7
```

- 進入互動模式使用

```
[root@master ~]# singularity shell centos_7.sif
```

- 從外部直接執行

```
[root@master ~]# singularity exec centos_7.sif cat /etc/redhat-release
```

Singularity 自訂映像檔範例

- 用現有映像檔或是下載新的映像檔

```
[root@master ~]# singularity build --sandbox centos_7.9 centos_7.sif  
or  
[root@master ~]# singularity build --sandbox centos_7.9 library://library/default/centos:7
```

- 以 **root** 身份進入互動模式安裝套件或更改內容

```
[root@master ~]# singularity shell --writable centos_7.9  
Singularity>yum update  
Singularity>yum install "package-name"  
Singularity>  
Singularity>exit
```

- 把修改好的 **sandbox** 打包成映像檔

```
[root@master ~]# singularity build centos_customized_7.9.sif centos_7.9
```

建立Singularity Image

- 試著建立一個可以跑python3 內含模組NumPy及pandas的Container Base Image使用Ubuntu 22.04

Hint:

1. 更新Ubuntu的官方套件庫 (apt-get update)
2. 搜尋可安裝的套件名稱 (apt-cache search 部份套件名稱)
3. 安裝套件 (apt-install 套件名稱)
4. 更新Python3 pip模組庫 (pip3 install --upgrade pip)
5. 安裝Python3的需要模組 (pip3 install 模組名稱)