

# 從 Linux 管理員進階 為雲端架構師

恆逸教育訓練中心 DCKA 課程學習筆記

## Mission Statement

本筆記旨在協助學員掌握容器技術與 K8S<sup>™</sup>  
編排平台的核心概念。透過理論與實作，  
從零開始建置、管理並部署高可用性的  
應用程式。

## Technical Spec Sheet

### 課程規格 / COURSE SPEC

課程名稱 : Docker Containers 與  
Kubernetes 系統管理

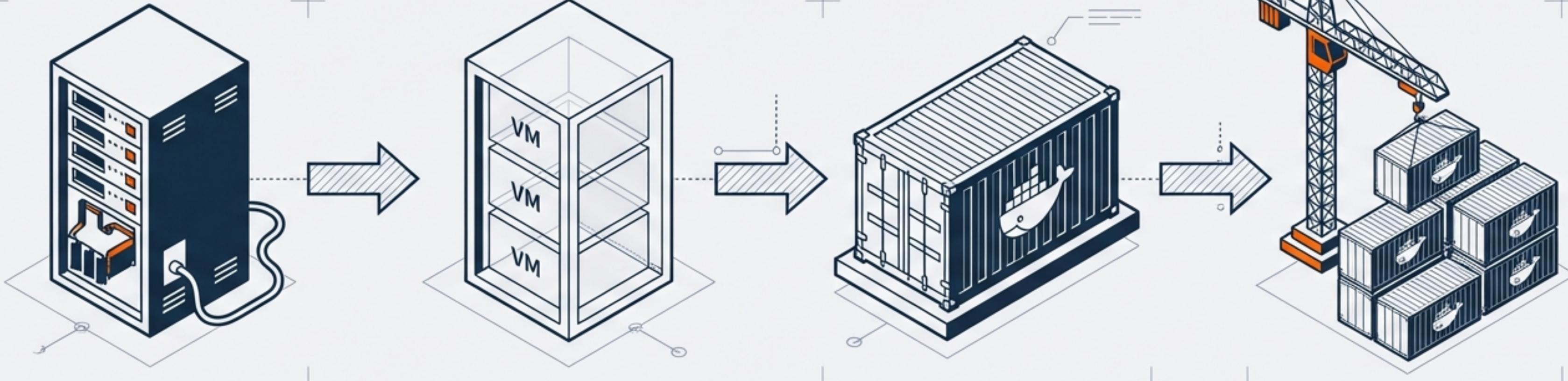
課程代號 : DCKA

總時數 : 21 小時 (3 天)

目標受眾 : 網路工程師、雲端運算工程師



# 基礎架構的演進：為什麼我們需要 Kubernetes？



## 傳統部署

實體機器，資源利用率低

## 虛擬化部署

VM 隔離，但作業系統負擔重

## 容器化部署

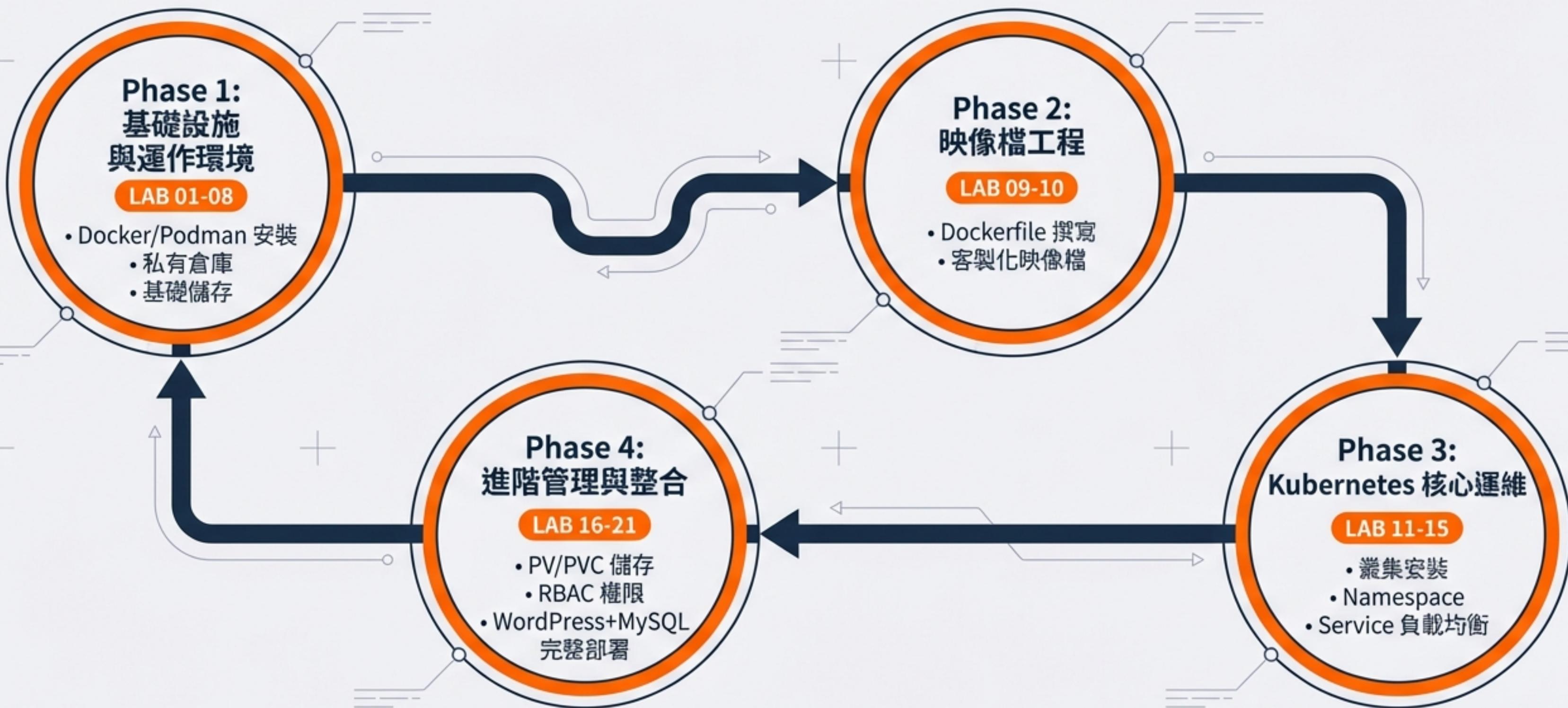
輕量級、共用 OS 核心  
(Docker)

## 容器編排

自動化管理大規模容器  
(Kubernetes)

Key Takeaway 課程將帶領你跨越從「單機容器」到「叢集編排」的技術門檻。

# 學習路徑圖：從 指令列到自動化叢集

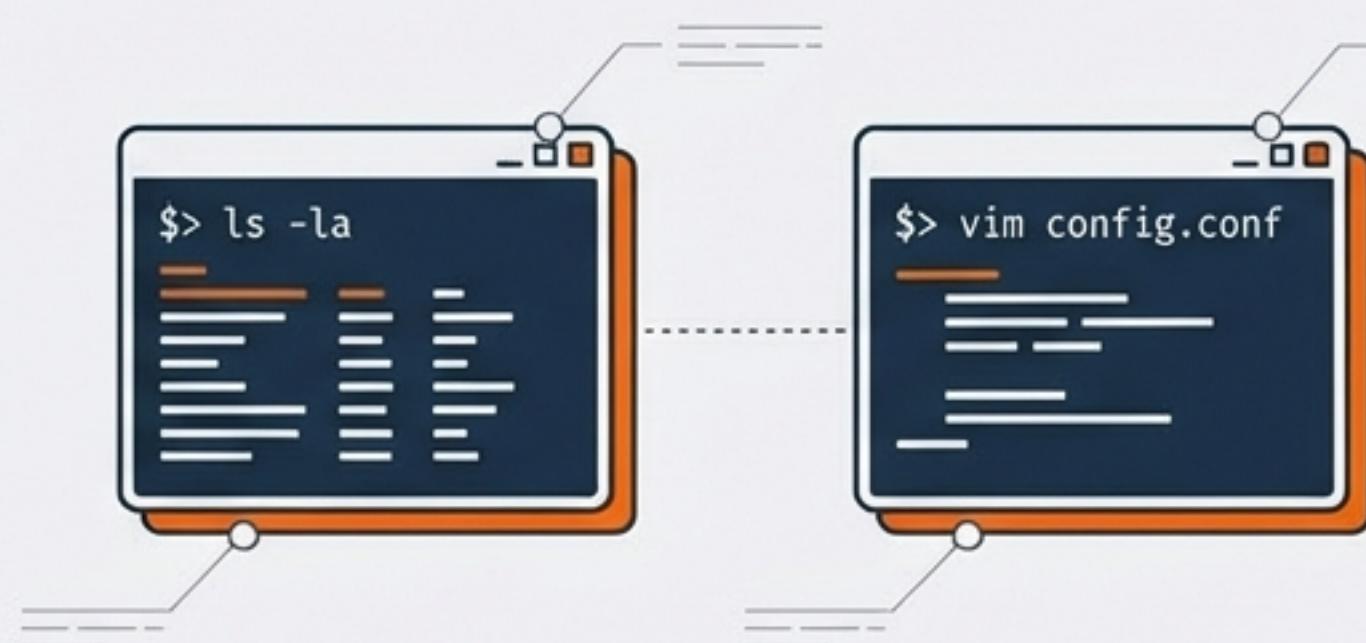


# 起飛前檢查：必備預備知識

在進入容器世界之前，你需要具備扎實的 Linux 系統管理能力。



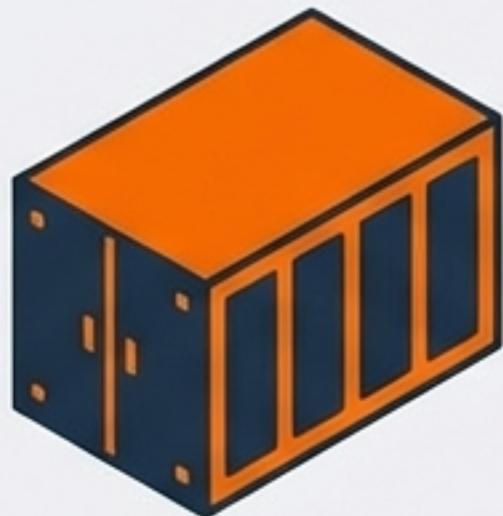
注意：所有的 Lab 實作都將建立在 Linux 環境之上。



# 階段一：掌握容器運作環境

LAB 01, 02, 03, 06

## Dual Engines



### Docker

業界標準

- LAB 02, 06: 安裝 Docker CE 與基本操作



### Podman

Daemonless (無常駐程式)

- LAB 03: 符合新一代安全標準的替代方案

## Core Skills:

>\_ 建立 (Create)

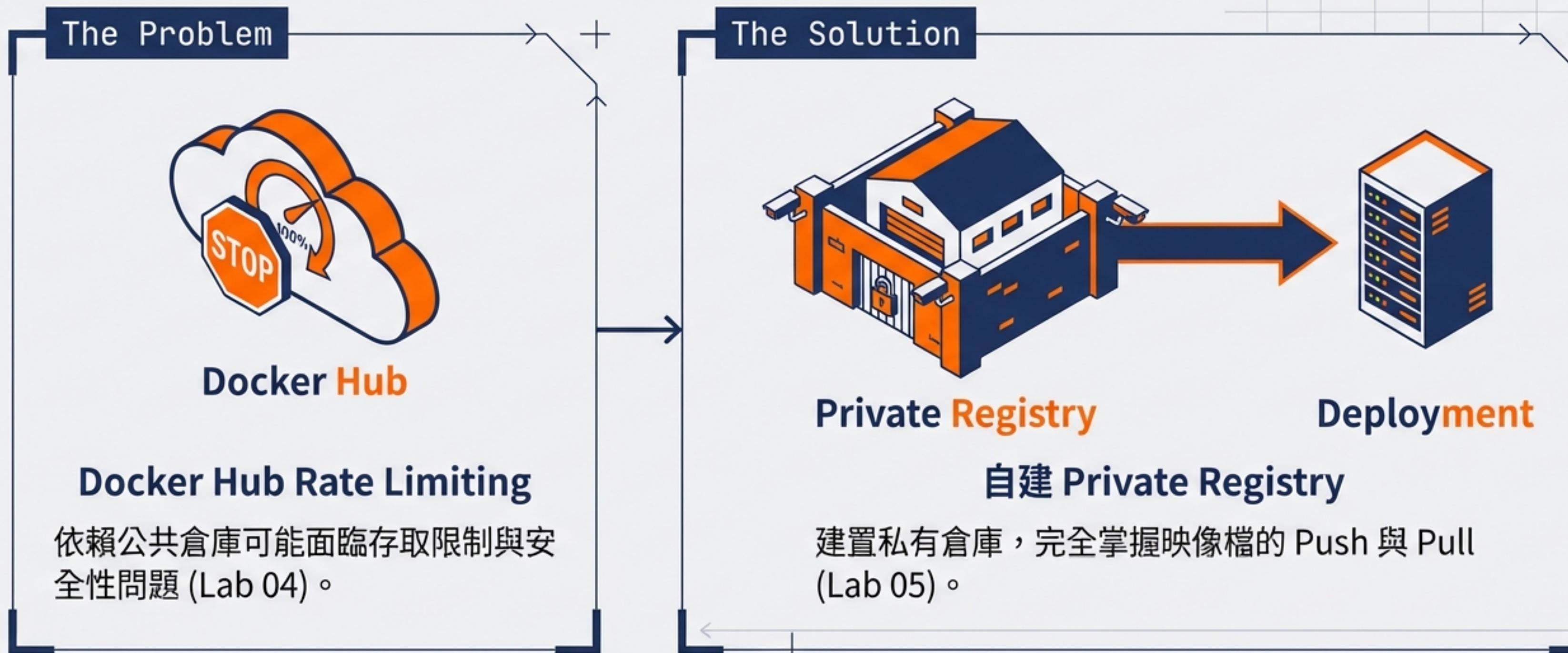
>\_ 啟動 (Start)

>\_ 停止 (Stop)

>\_ 刪除 (Delete)

# 映像檔管理與私有倉庫

LAB 04, 05



# 初次部署：單機版 WordPress 與資料持久化

LAB 07, 08

## Mini-Project Goal

架設部落格 (Lab 08)



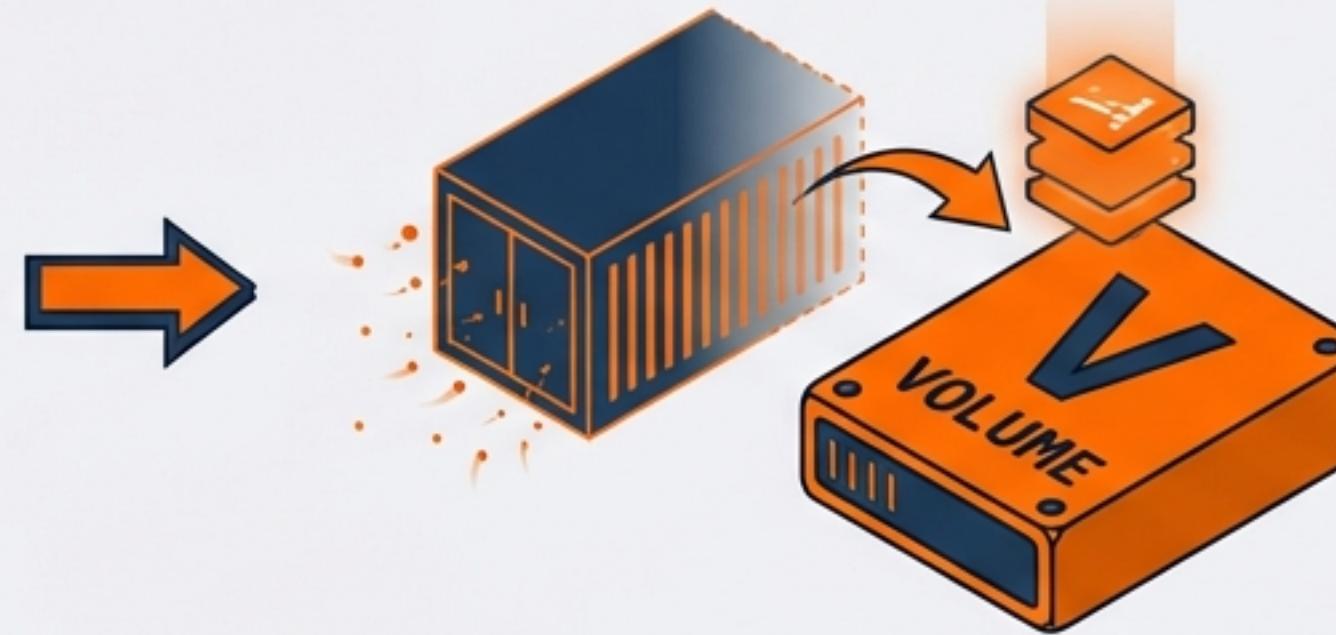
## Critical Concept: Persistent Storage (Lab 07)

State A (No Storage)



容器預設是暫時的 (Ephemeral)

State B (With Volume)



使用 Volume 確保資料安全

限制：尚未具備  
自動修復與水平  
擴展能力 (待  
K8s 階段解決)

# 階段二：映像檔工程與最佳實踐

LAB 09, 10

## Manual vs. Automated

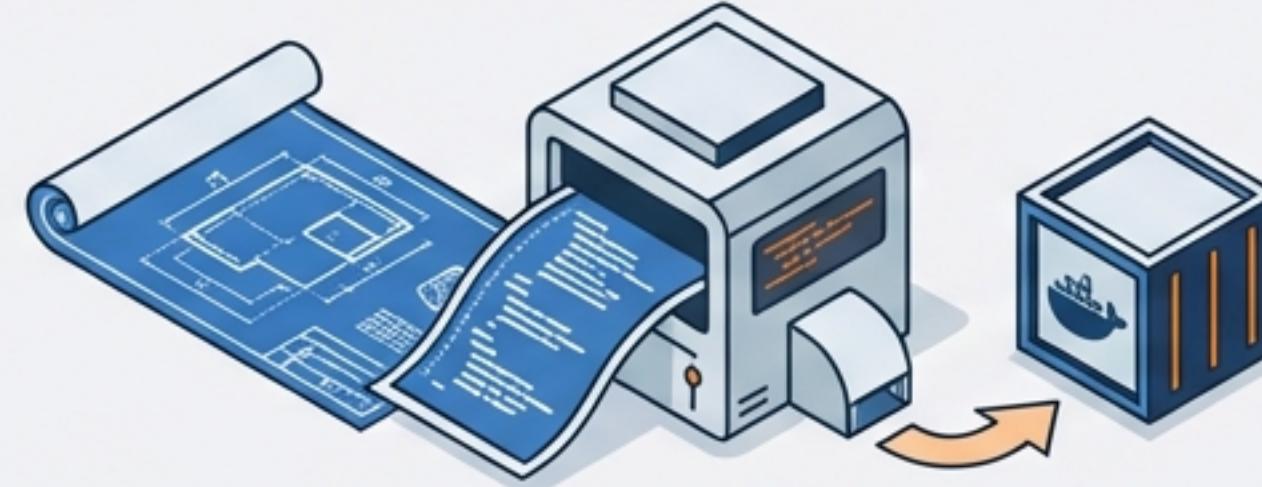
LAB 09: docker commit



將運行中的容器狀態保存為 Image

⚠ 過程不透明，難以維護

LAB 10: Dockerfile



Infrastructure as Code  
使用指令腳本定義建置過程

✓ 可版控、可重複建置、透明化

**Takeaway:** 掌握 Dockerfile 指令與最佳實踐是客製化應用的基礎。

# 階段三：Kubernetes 架構與安裝

LAB 11, 12, 13

## Installation Types



Standalone  
Kubernetes (Lab 11)  
單機開發測試環境



Cluster Installation  
(Lab 12)  
多節點正式環境部署

## The Control Stick

**kubectl**

user@k8s:~\$ kubectl get nodes

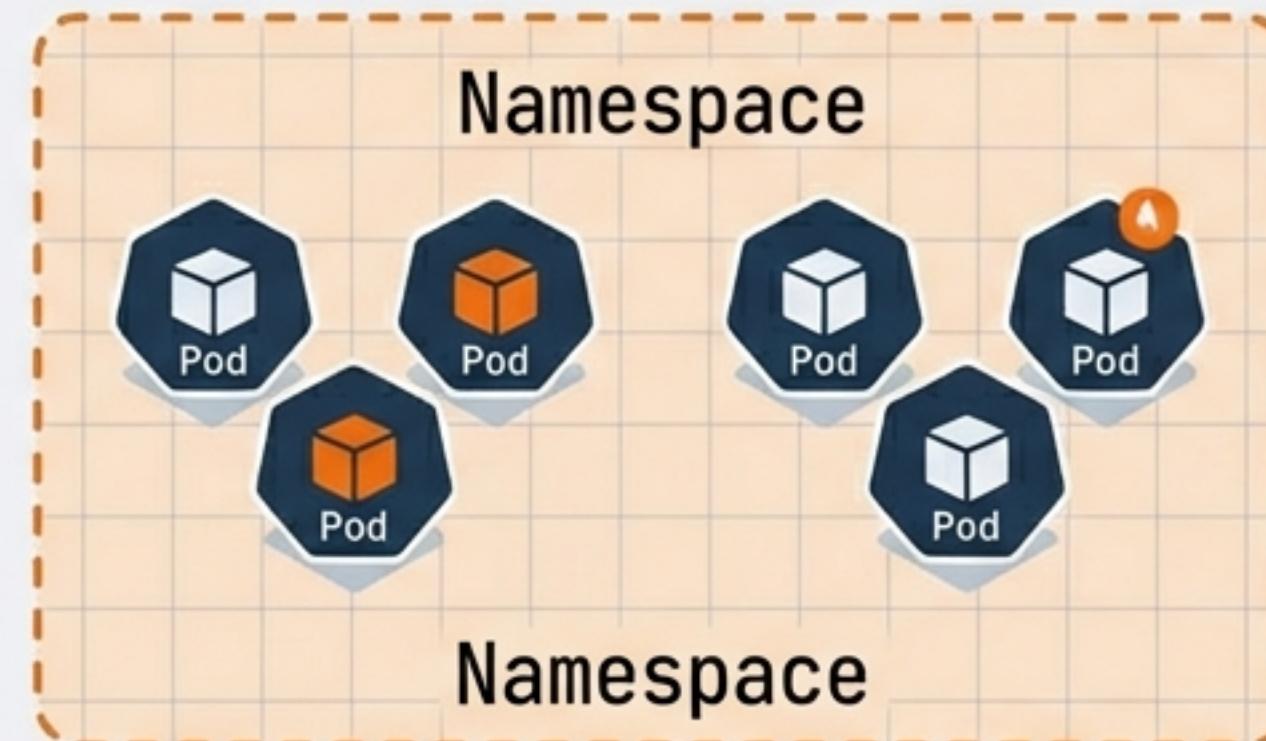
Lab 13: 學習與 API Server 互動，  
控制叢集資源。

# 服務編排：命名空間與流量管理

LAB 14, 15

## Cluster City

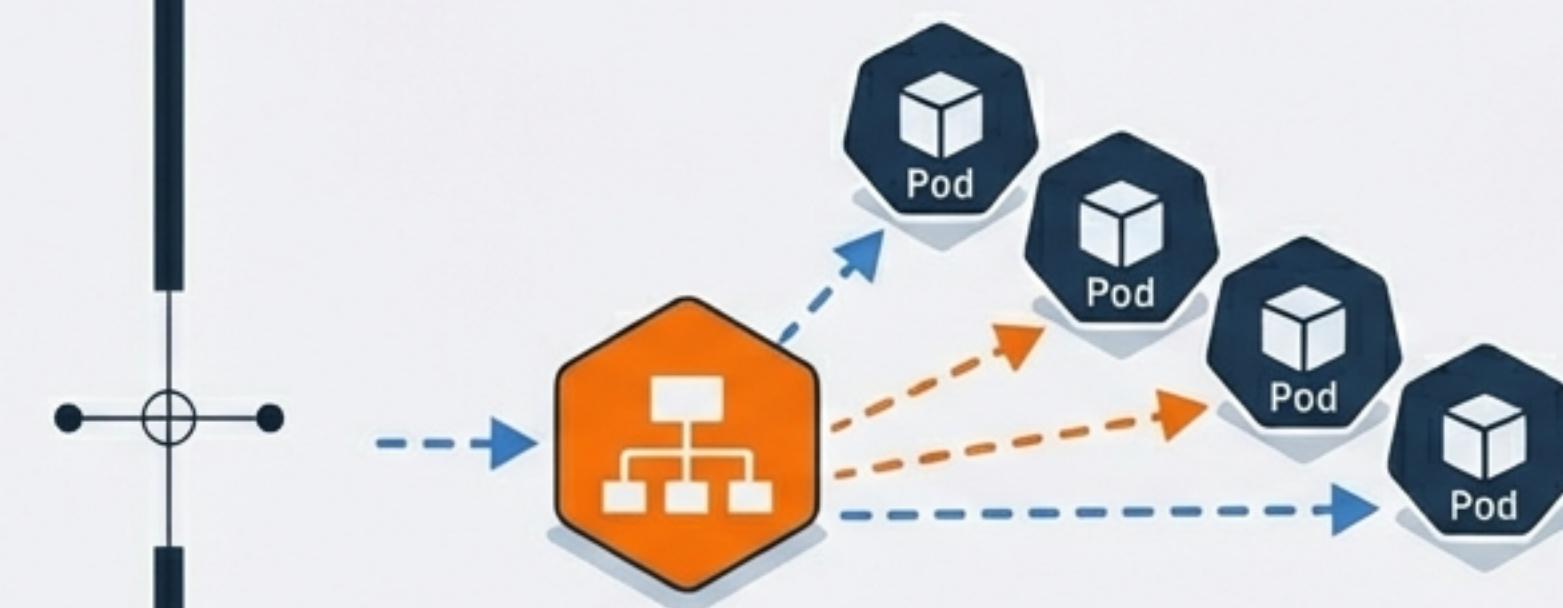
### Organization (Lab 14)



資源隔離與分組，實現多租戶管理

✓ Rolling Update 實作無停機更新

### Connectivity (Lab 15)



Service : 解決 Pod IP 變動問題  
提供穩定的存取入口與負載均衡  
(Load Balancing)

# 階段四：進階儲存與配置管理

LAB 16, 17

## Enterprise Storage Strategy (Lab 16)



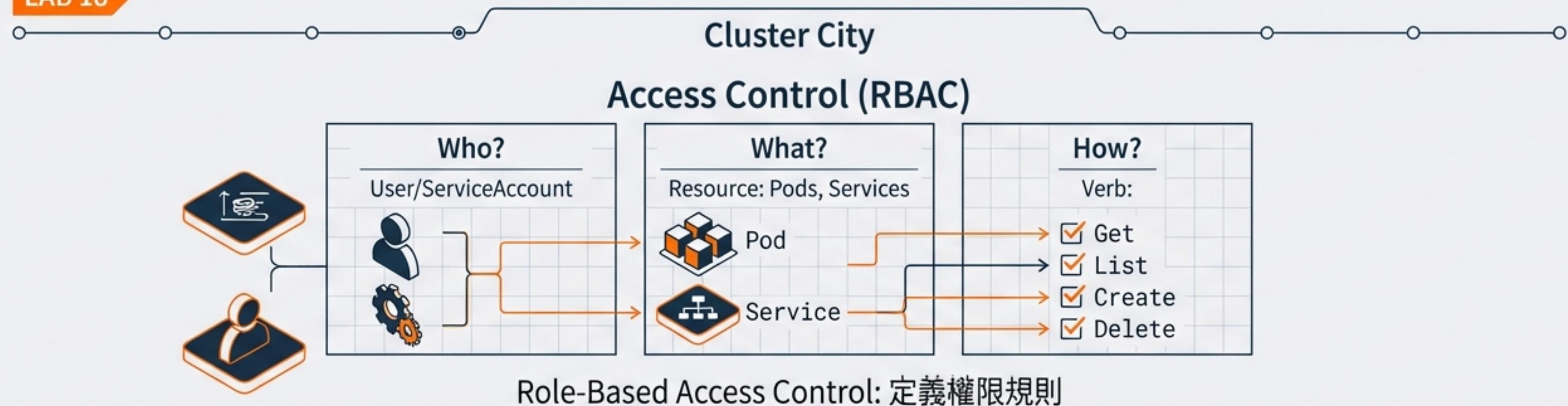
## Security & Config (Lab 17)



安全地管理密碼、金鑰等敏感資訊，  
不直接暴露在程式碼中。

# 叢集安全與監控

LAB 18



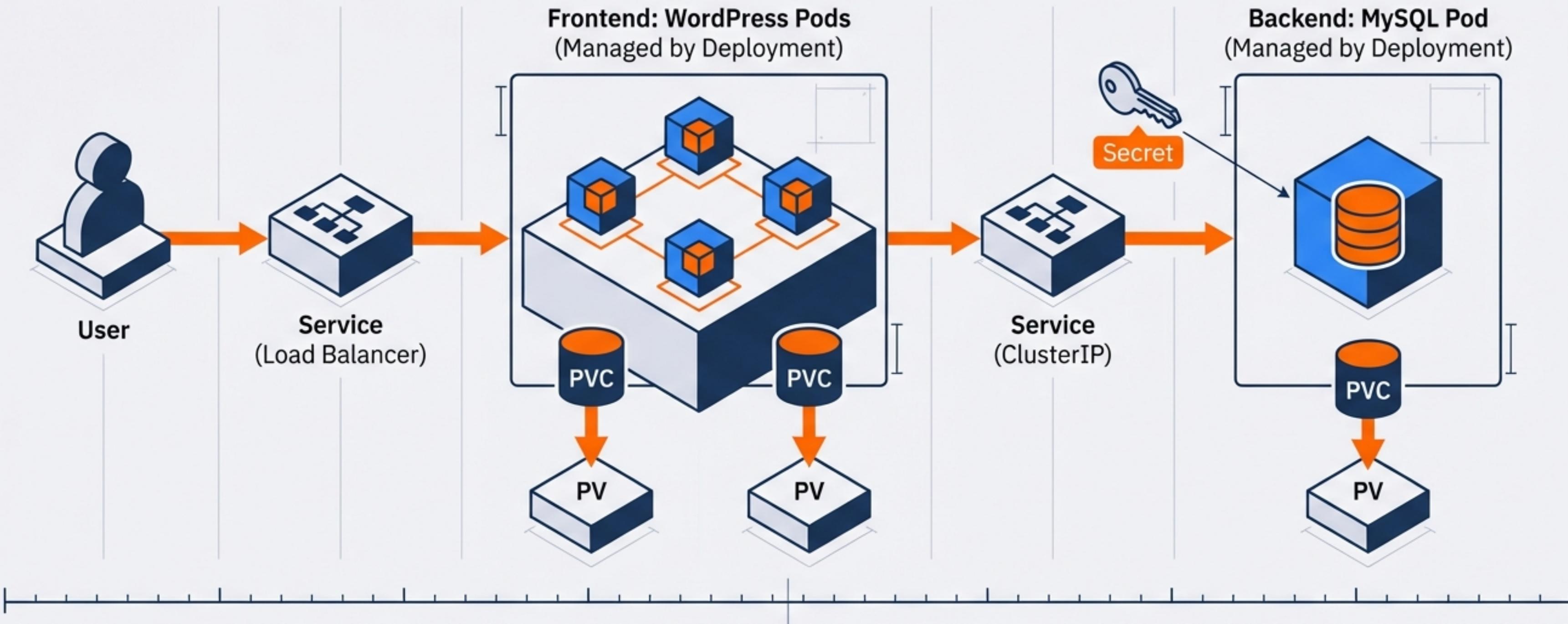
## Observability

• [10:09:30] INFO - Pod 'web-app-1' started.	• [10:09:30] INFO - Pod 'web-app-1' started.	• [10:09:30] INFO - Pod 'web-app-1' started.
⚠ [10:09:45] INFO - Pod 'web-app-1' started.	⚠ [10:09:45] WARN - High CPU usage on Node 'node-2'.	⚠ [10:09:45] WARN - High CPU usage on Node 'node-2'.
✖ [10:09:01] WARN - High CPU usage on Node 'node-2'.	✖ [10:10:01] ERROR - Service 'db-service' connection timeout.	✖ [10:09:01] WARN - High CPU usage on Node 'node-2'.
'test-pod. ⏺ [10:10:15] AUDIT - User 'admin' deleted Pod 'test'.	⏺ [10:10:15] AUDIT - User 'admin' deleted Pod 'test-pod'.	⏺ [10:10:15] AUDIT - User 'admin' deleted Pod 'test-pod'.

Event & Log: 查詢系統事件與容器日誌，進行故障排除與稽核。

# 最終實戰：WordPress + MySQL 完整叢集部署

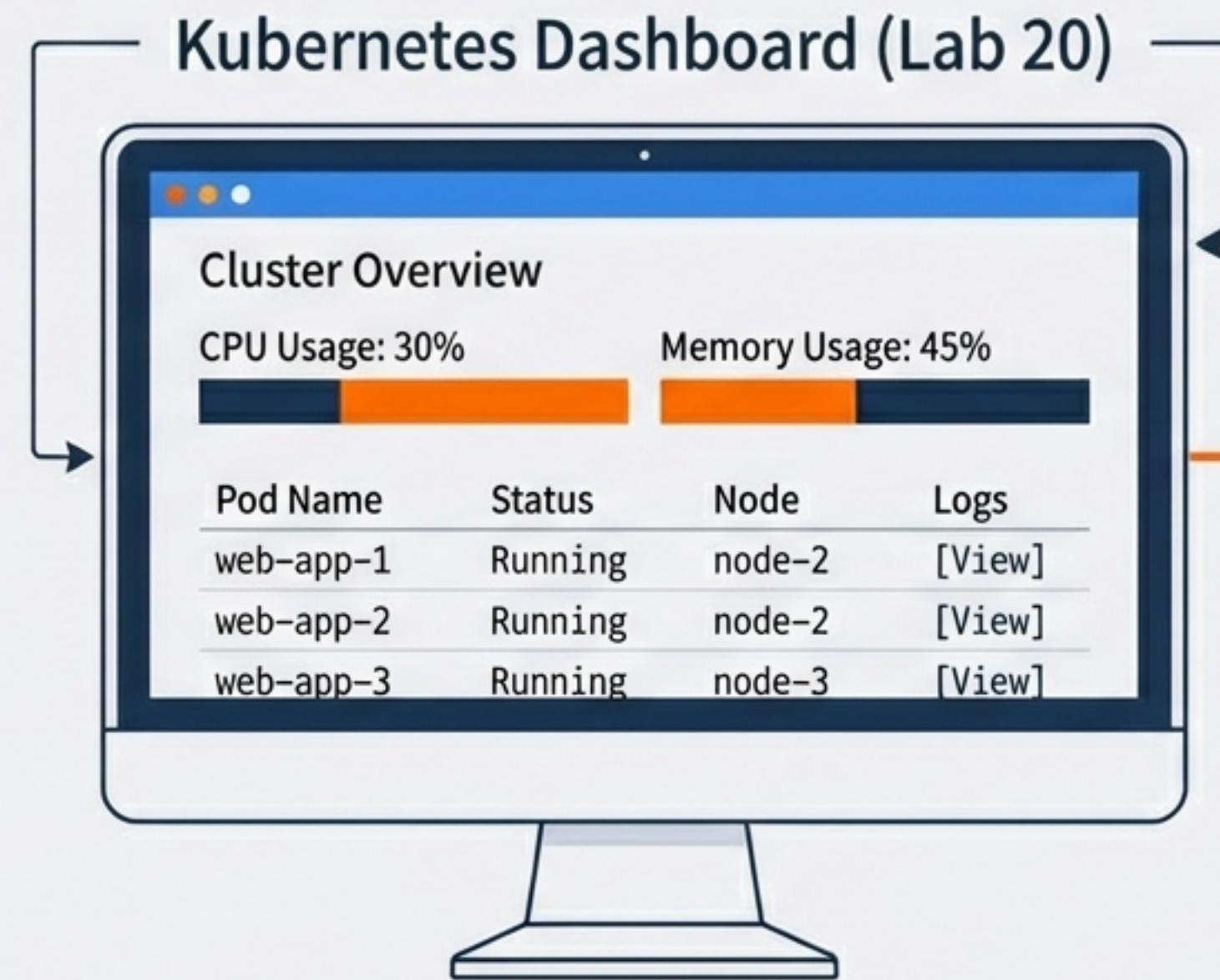
LAB 19



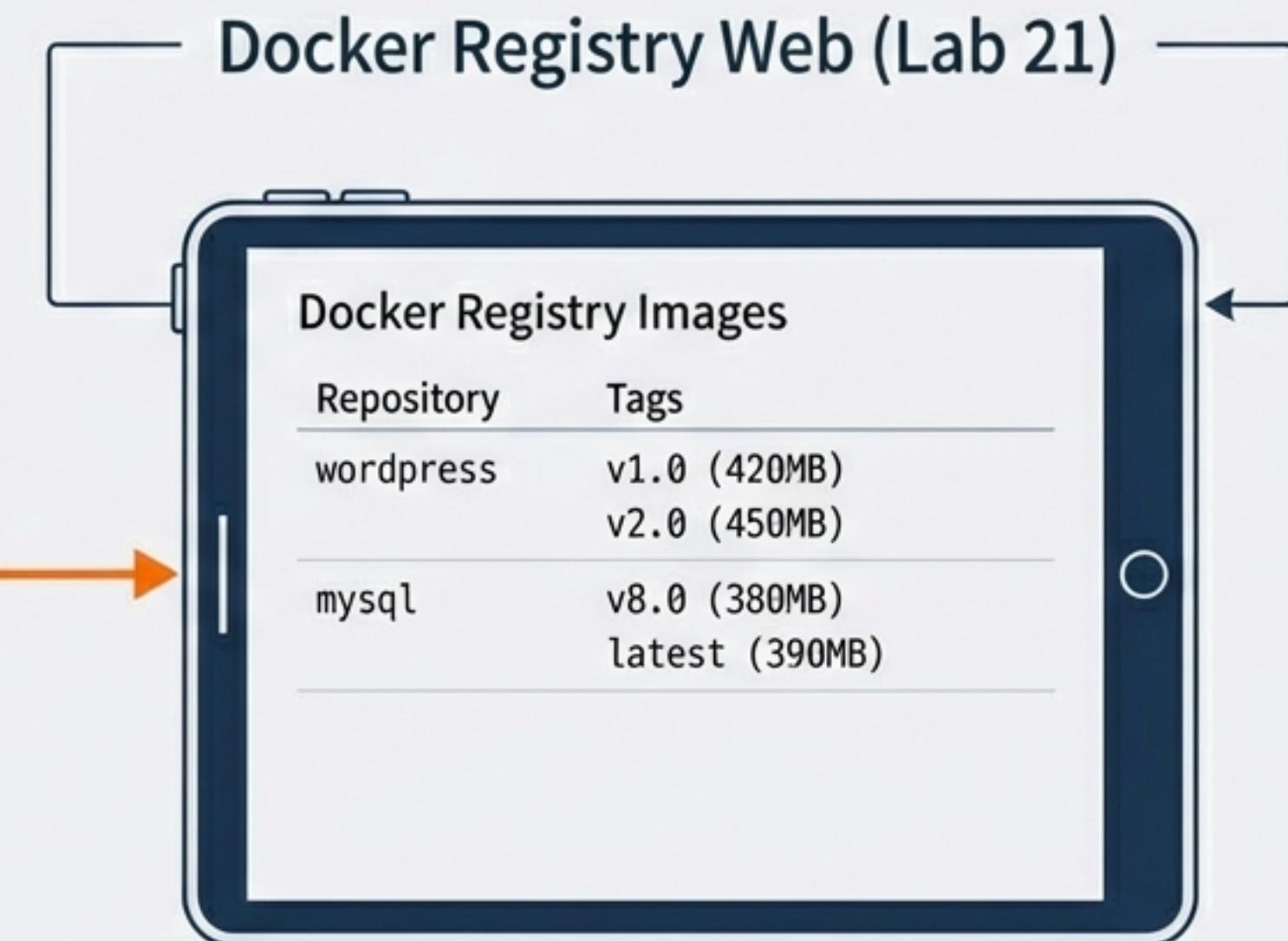
整合 Deployment, Service, PVC, Secret，建置具備自我修復能力的 Web 應用架構。

# 圖形化管理工具

LAB 20, 21



可視化監控資源使用率、查看 Logs、管理 Workloads。



瀏覽與管理已上傳的 Image 版本與 Tag。

# 隨身工具箱與延伸資源

## Cheat Sheets (速查表)

Noto Sans TC

- Docker 指令速查
- Kubernetes 指令速查 (kubectl)



## Troubleshooting (疑難排解)

Noto Sans TC

遇到問題時，請參考 FAQ 與疑難排解指南。



## Official Docs

Noto Sans TC

- Docker / Kubernetes / Podman 官方文件



你的雲端架構師之旅才剛開始