

# READY FOR **ANY** vForum2015

9 December 2015 | Taipei, Taiwan

## VMware Integrated Container 與 VMware Integrated OpenStack 您怎麼看?

Mika Chan  
SE, VMware Taiwan

# VMware 原生雲端應用程式解決方案

VMware Integrated Container / Cloud-native application Solution

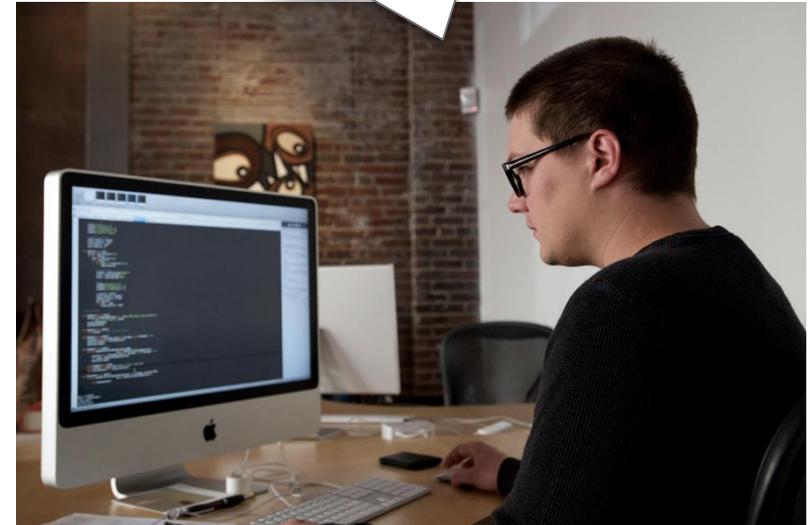
也不是很久的事情, 生活曾經美好...

Apps!



使用者

登入到虛擬機  
器做程式開發



開發人員

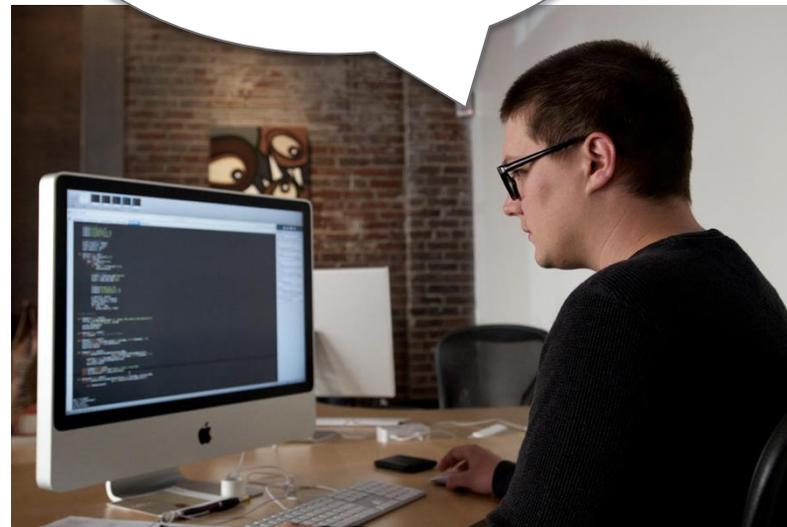
# 隨著行動裝置盛行, 一切都變了...

Apps! Apps! More  
Apps! Apps!更多的  
功能! Apps!  
快給我!



使用者

APIs! DevOps!  
**Containers!**  
~~Control the~~  
~~infrastructure!~~



開發人員

# Dev + Ops



開發人員

IT 維運人員

Developer-as-a-Service

=

API + Containers + Container Scheduler

# 但IT維運人員只需提供虛擬機....



開發人員

我的應用程式需要使用 **Jenkins** 和 **Mesos** 叢集

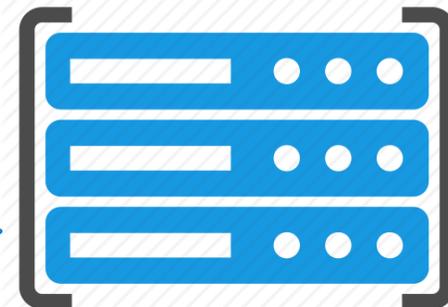


**Jenkins**



**MESOS**

以下是一些虛擬機



資料中心

## 導致效率不彰



需要耗時地  
將虛擬機轉換成  
叢集架構

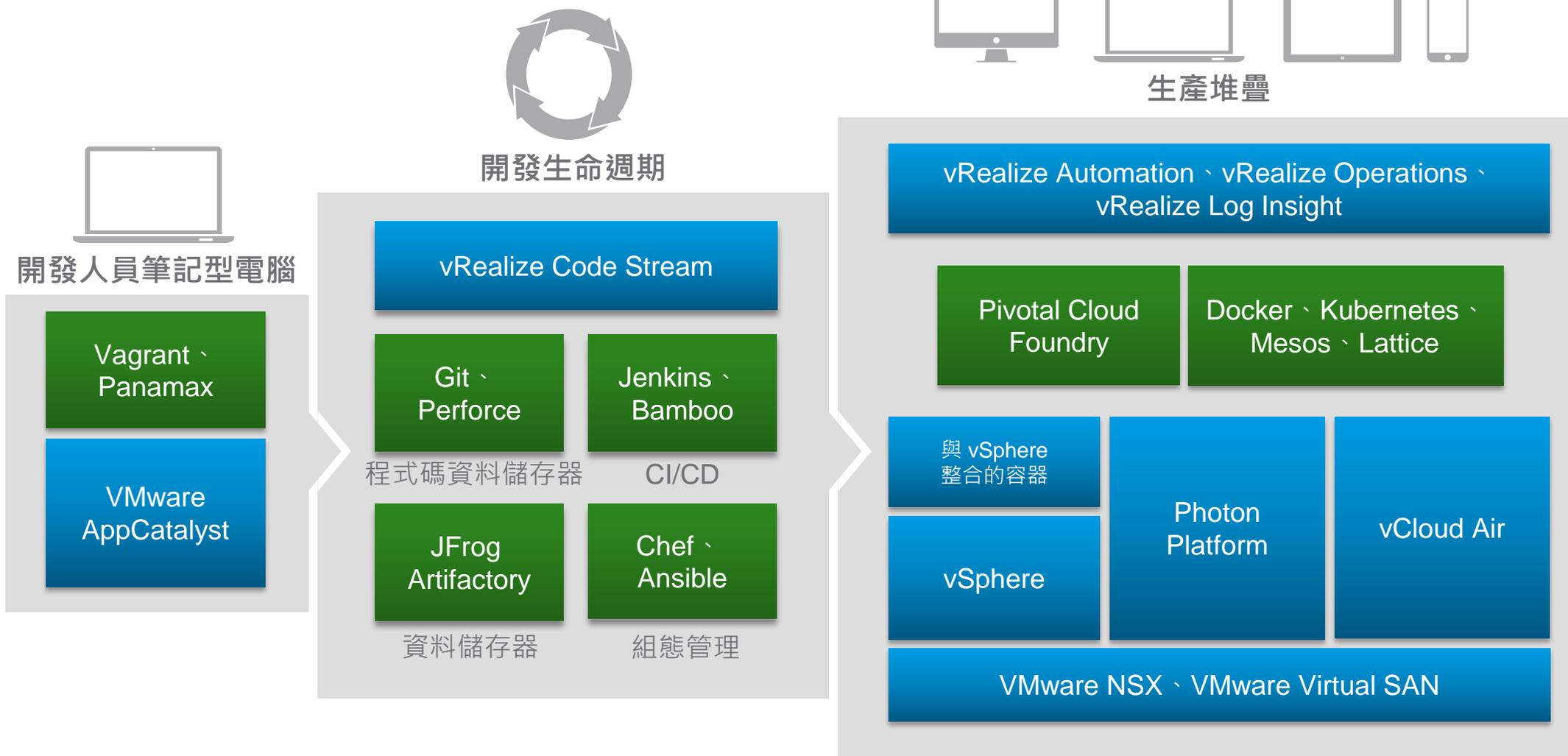


每種部署各有不同。  
難以強制標準化

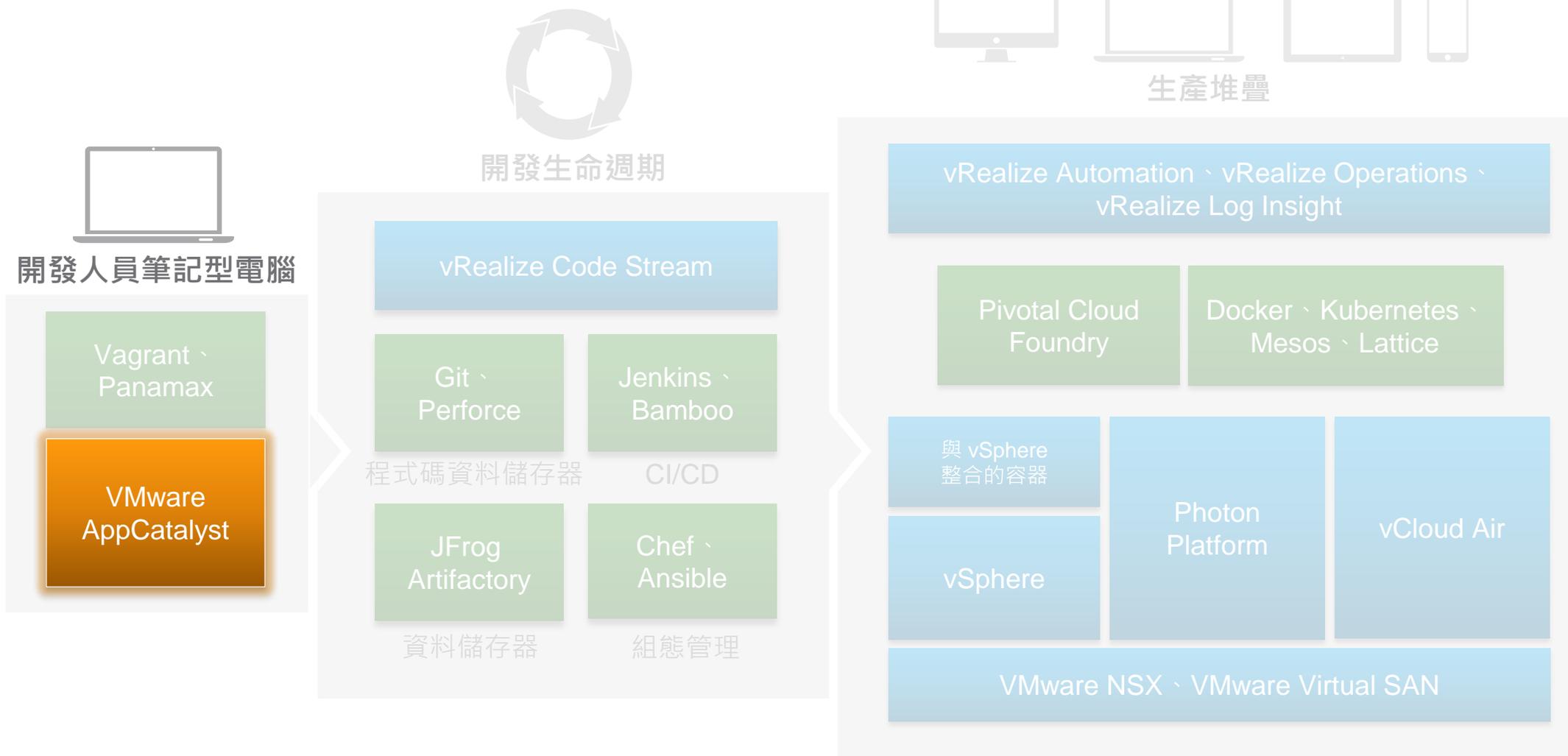


每種應用程式  
均會 建立專屬的  
自訂管理系統

# VMware 原生雲端應用程式堆疊



# VMware 原生雲端應用程式堆疊



# VMware AppCatalyst



## 準備邁向原生雲端之路

AppCatalyst 隨附 Photon OS 和 Vagrant，而且支援立即可用的 Docker 容器。



## 專為開發人員所打造

AppCatalyst 為 REST API 和 CLI 導向，可與容器順暢整合，而且也是微服務式工作流程。

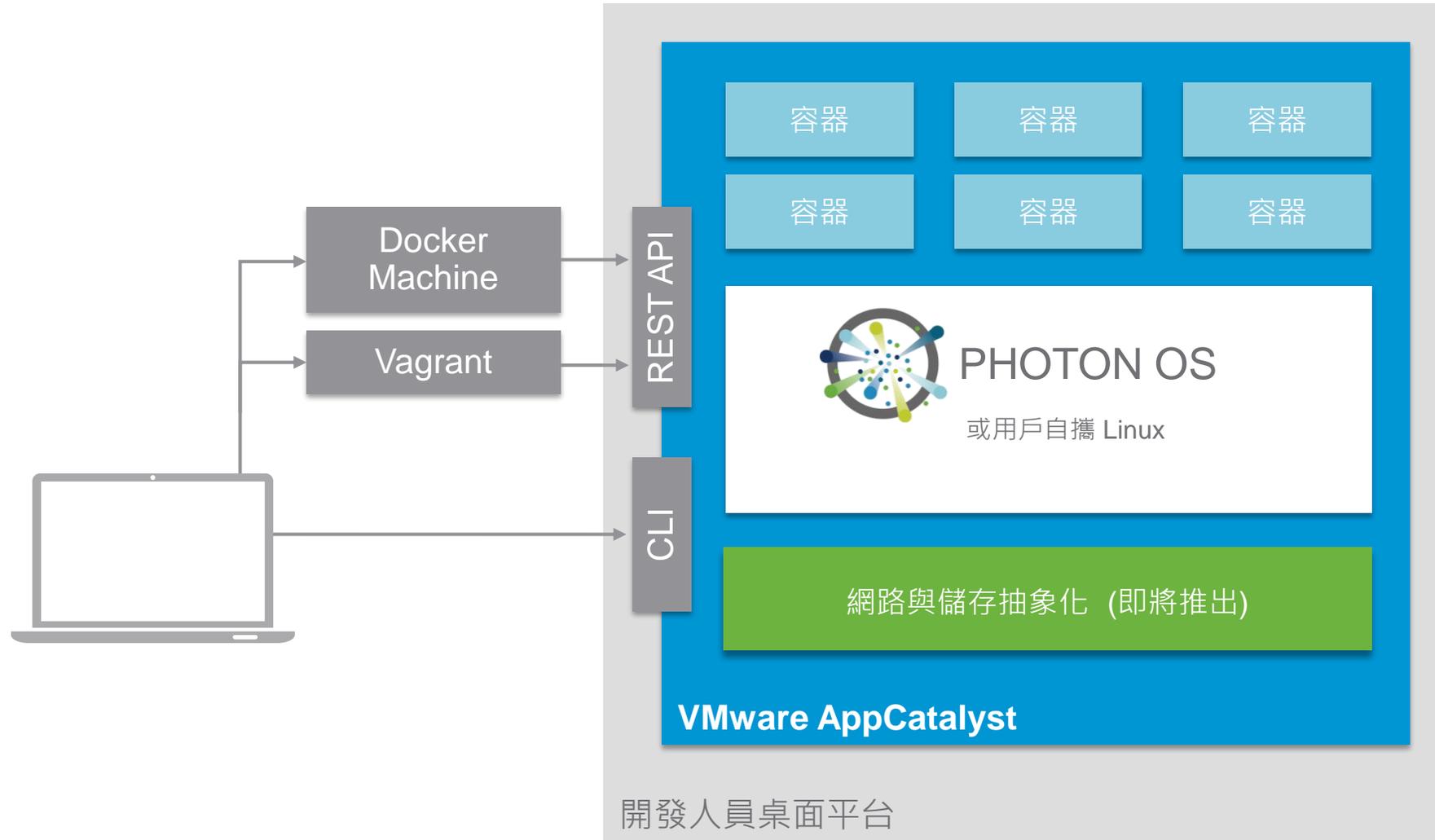


## 可免費使用

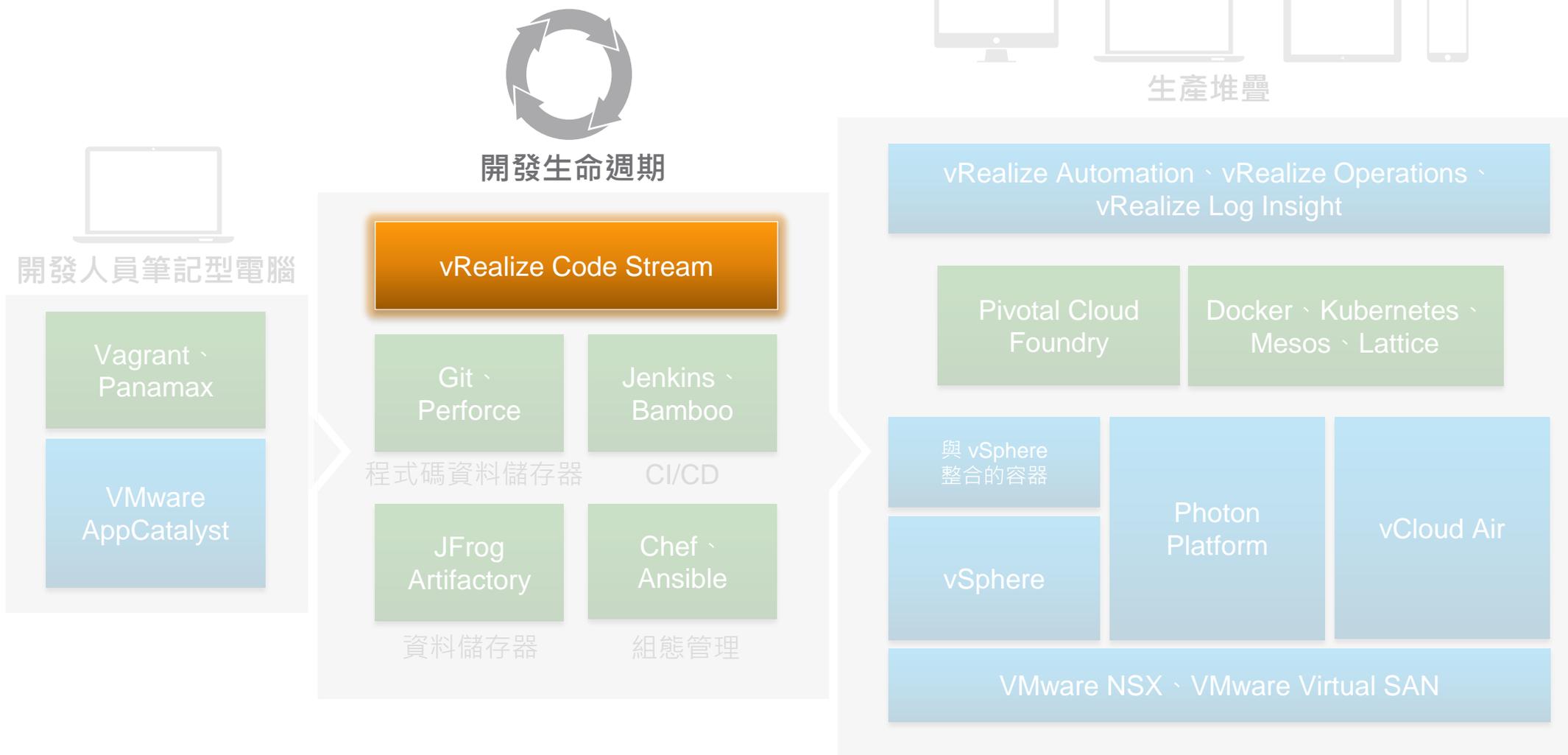
AppCatalyst 可免費提供給使用者，現在就可以下載使用。

立即下載技術預覽！  
<http://getappcatalyst.com>

# VMware AppCatalyst

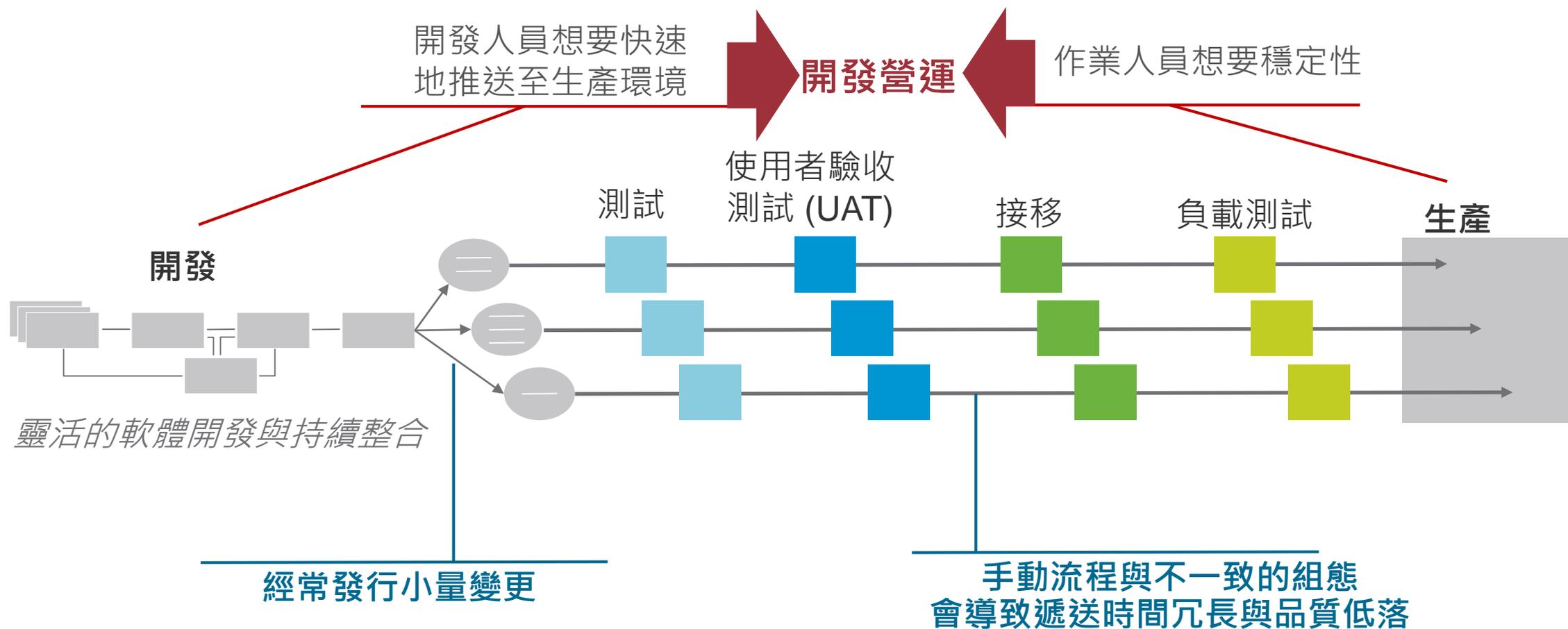


# VMware 原生雲端應用程式堆疊

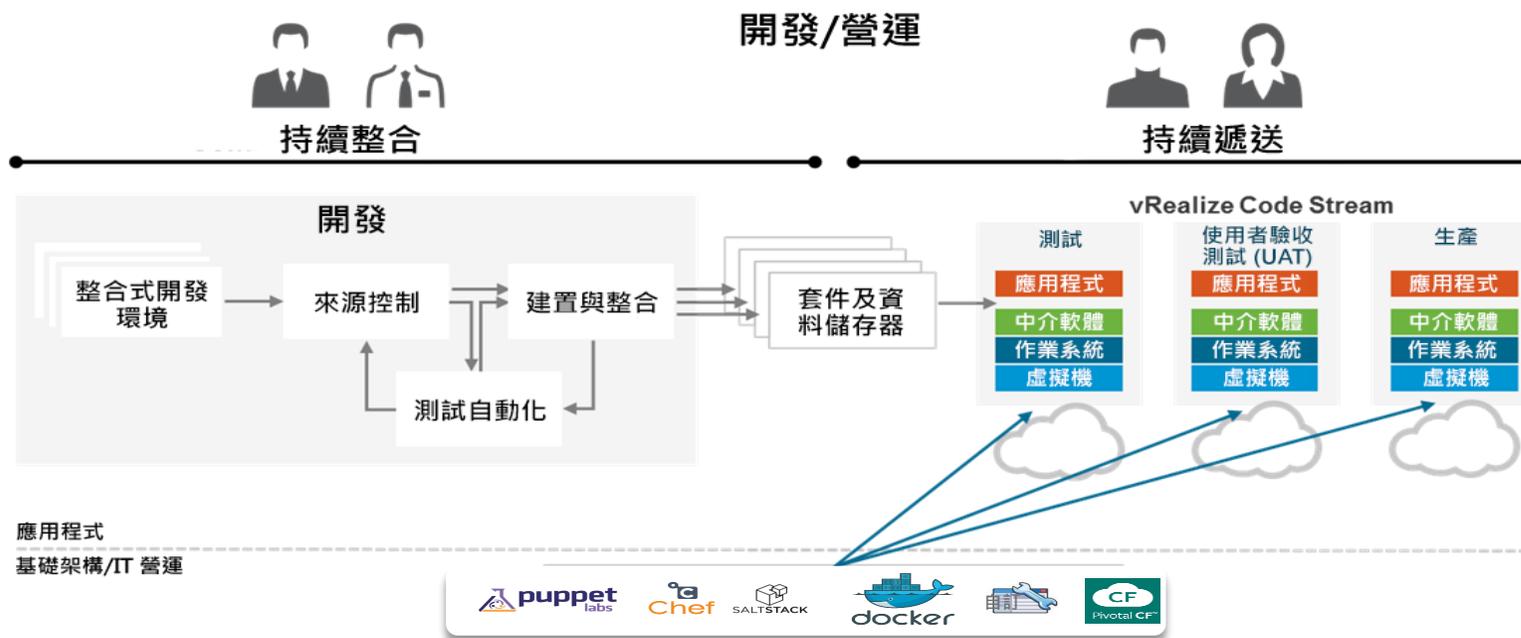


# 持續遞送(CD)所面臨的挑戰

具有靈活開發的發行流程



# vRealize Code Stream – 加速應用程式遞送



確保一致、可重複且可預測的軟體發行流程



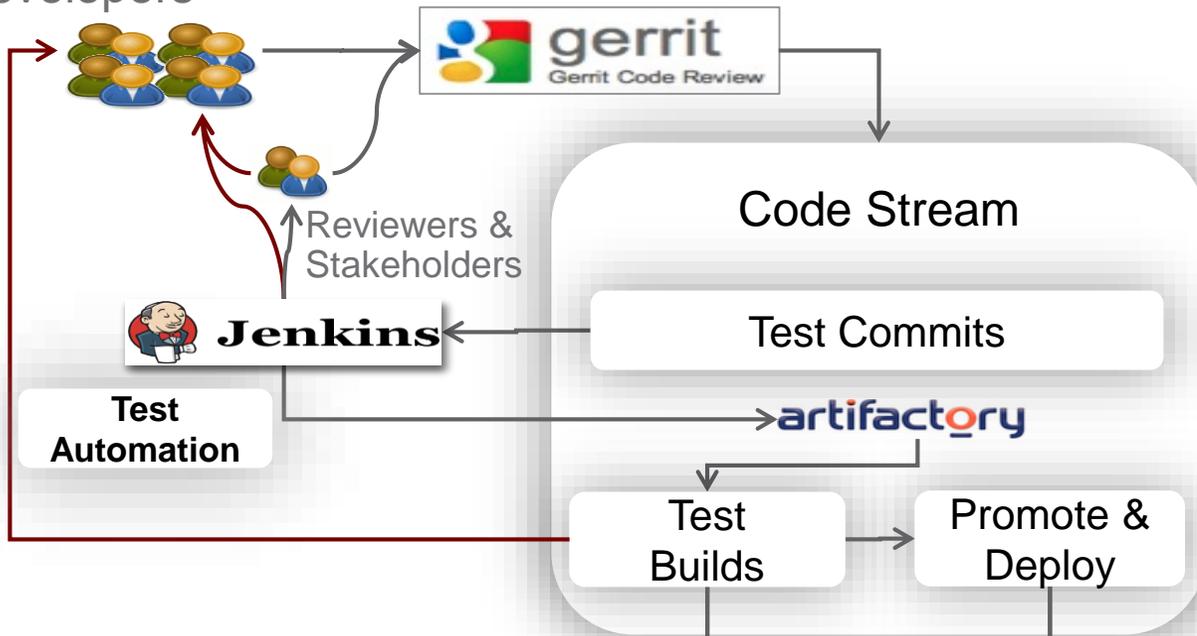
免除與手動作業和遞交相關的成本與錯誤



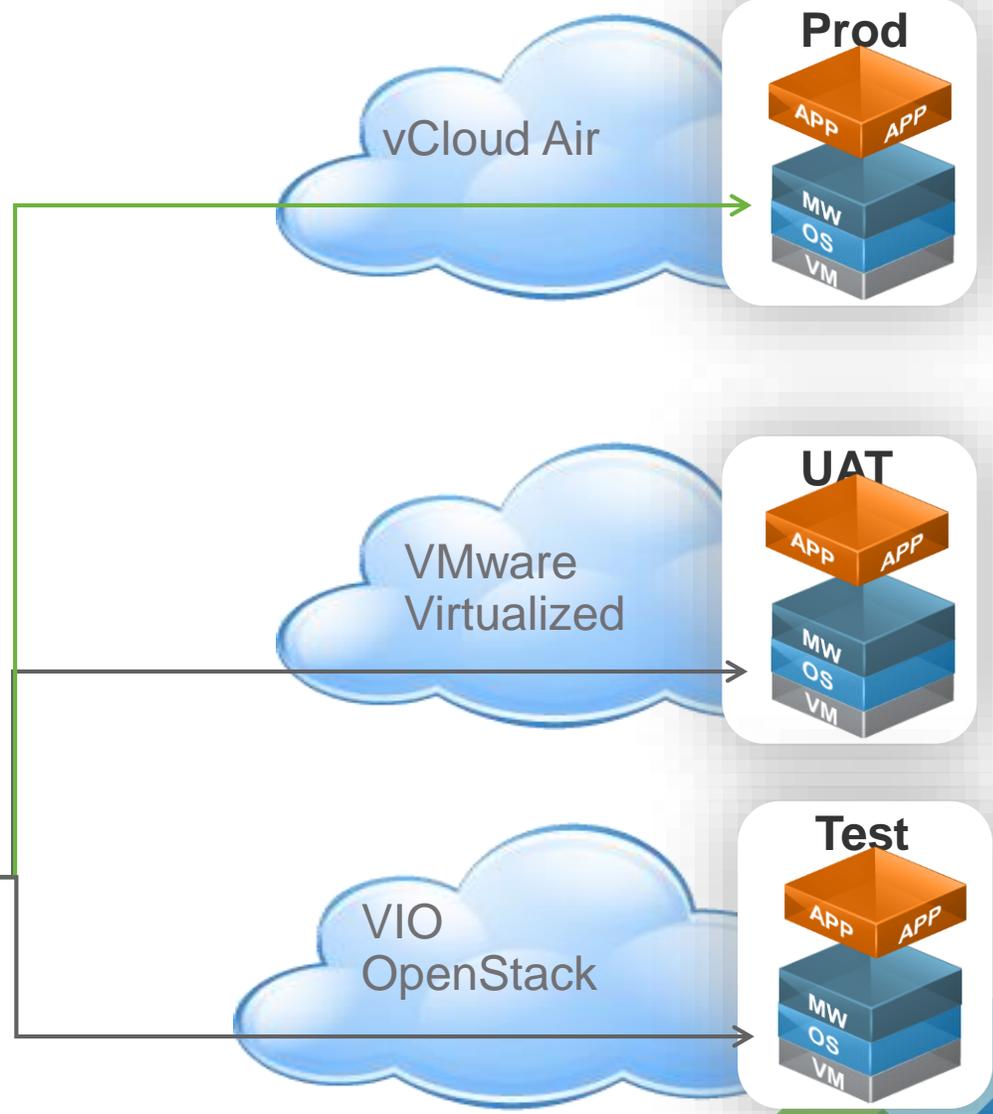
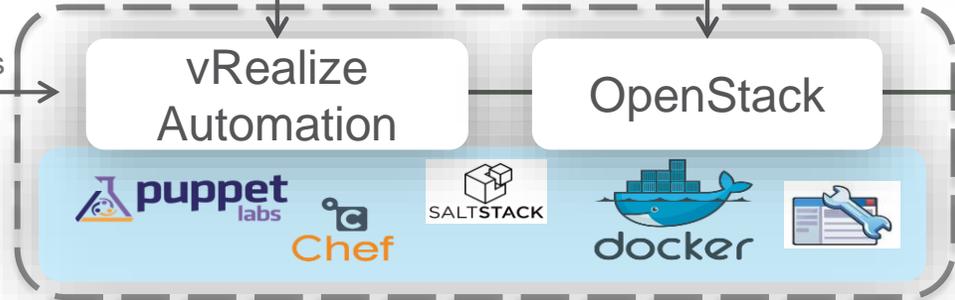
運用您的軟體開發發行鏈中所有工具的值

# vRealize Code Stream – 工作流程

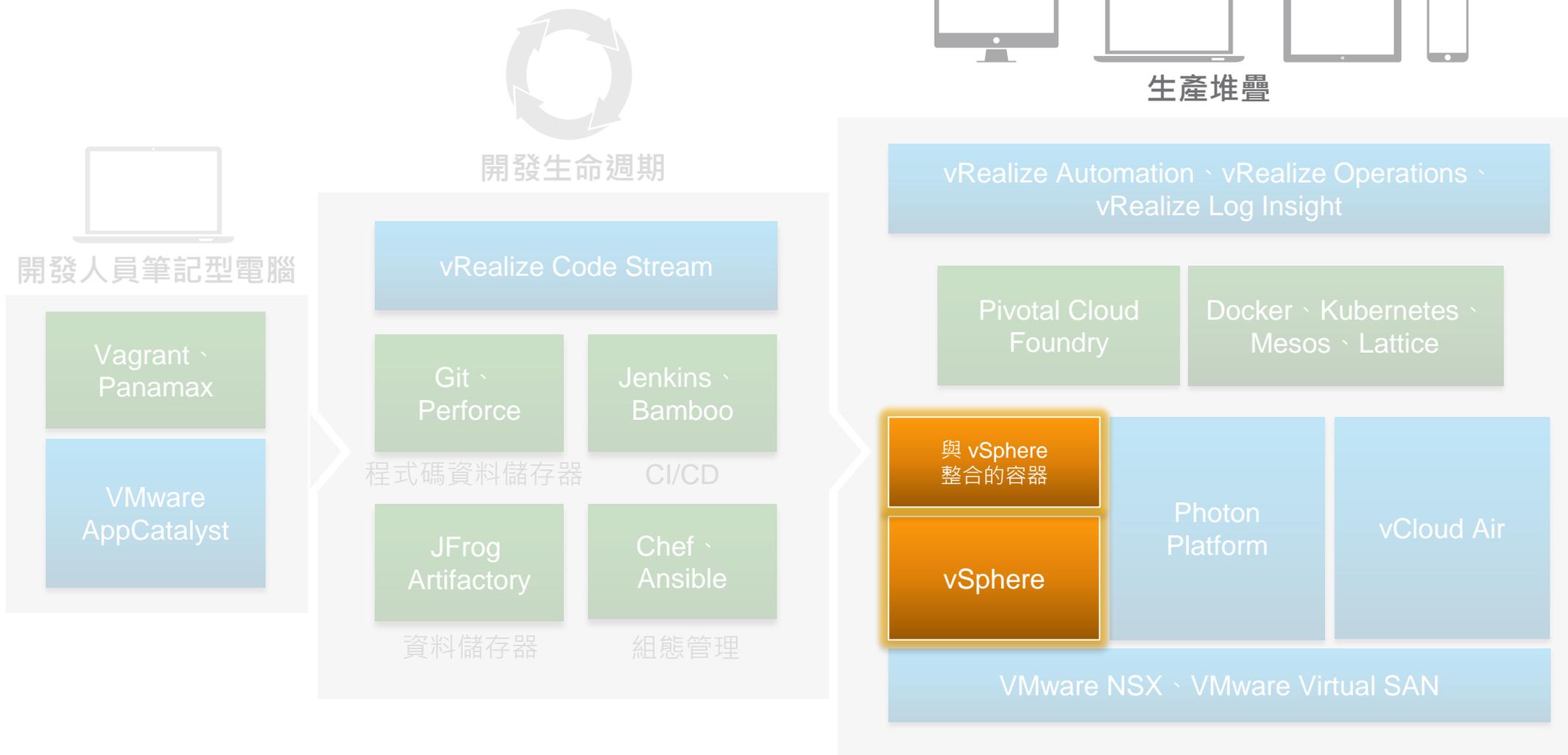
Apps & Ops  
Developers



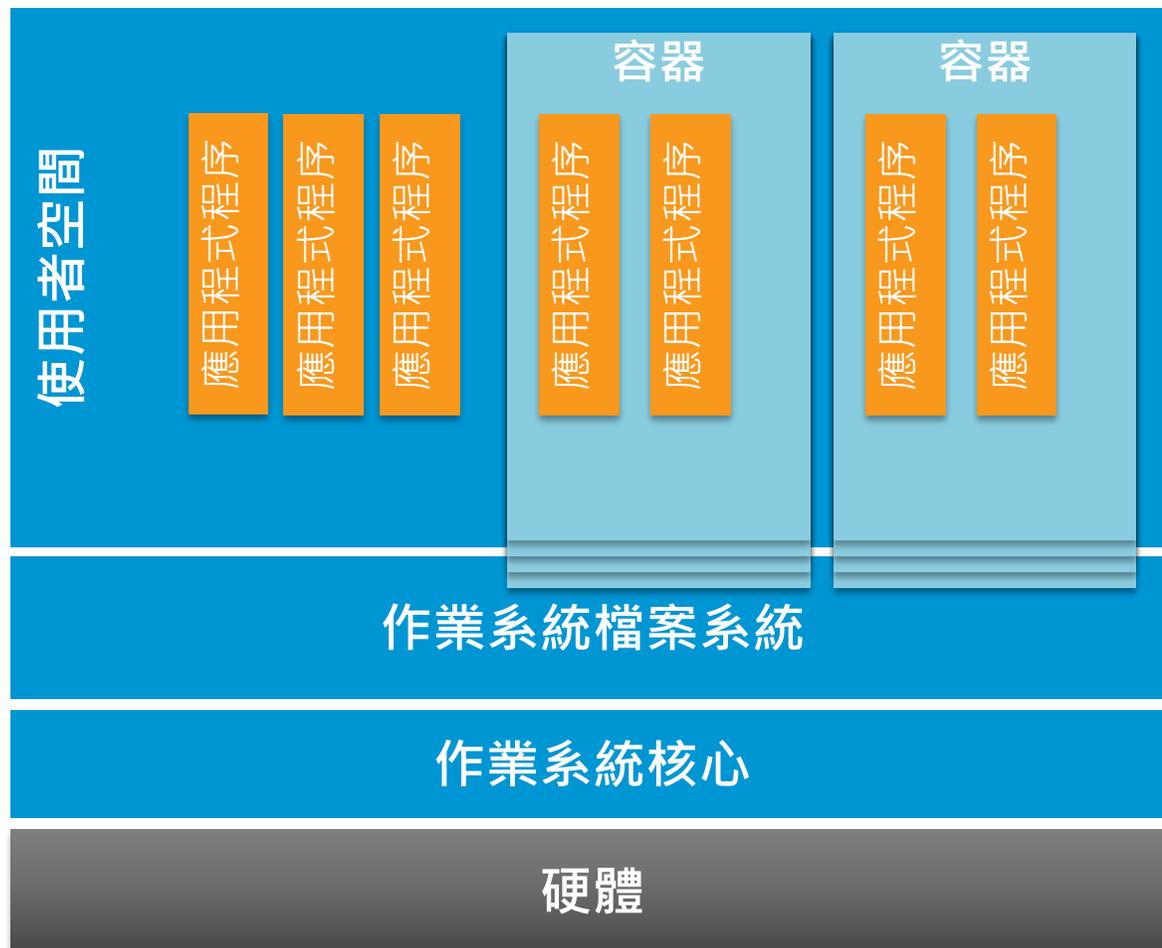
Ops  
Admins



# VMware 原生雲端應用程式堆疊



# Linux 容器是甚麼？



## 作業系統層級的隔離

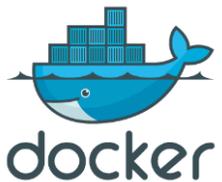
- 個別核心子系統層級的隔離 (例如檔案系統、程序表等等)
- 使用者層級的程序 (LXC、libcontainer) 協調這些子系統來建立容器

## 已存在多年

Solaris Zones、FreeBSD Jails、OpenVZ

## 為什麼？

- 程序隔離
- 可複製的環境
- 大規模啟用管理功能



# 為程式碼的「運送容器」

開發人員



原因為...

- 無障礙的部署與最高的可移植性

在開發人員筆記型電腦上：

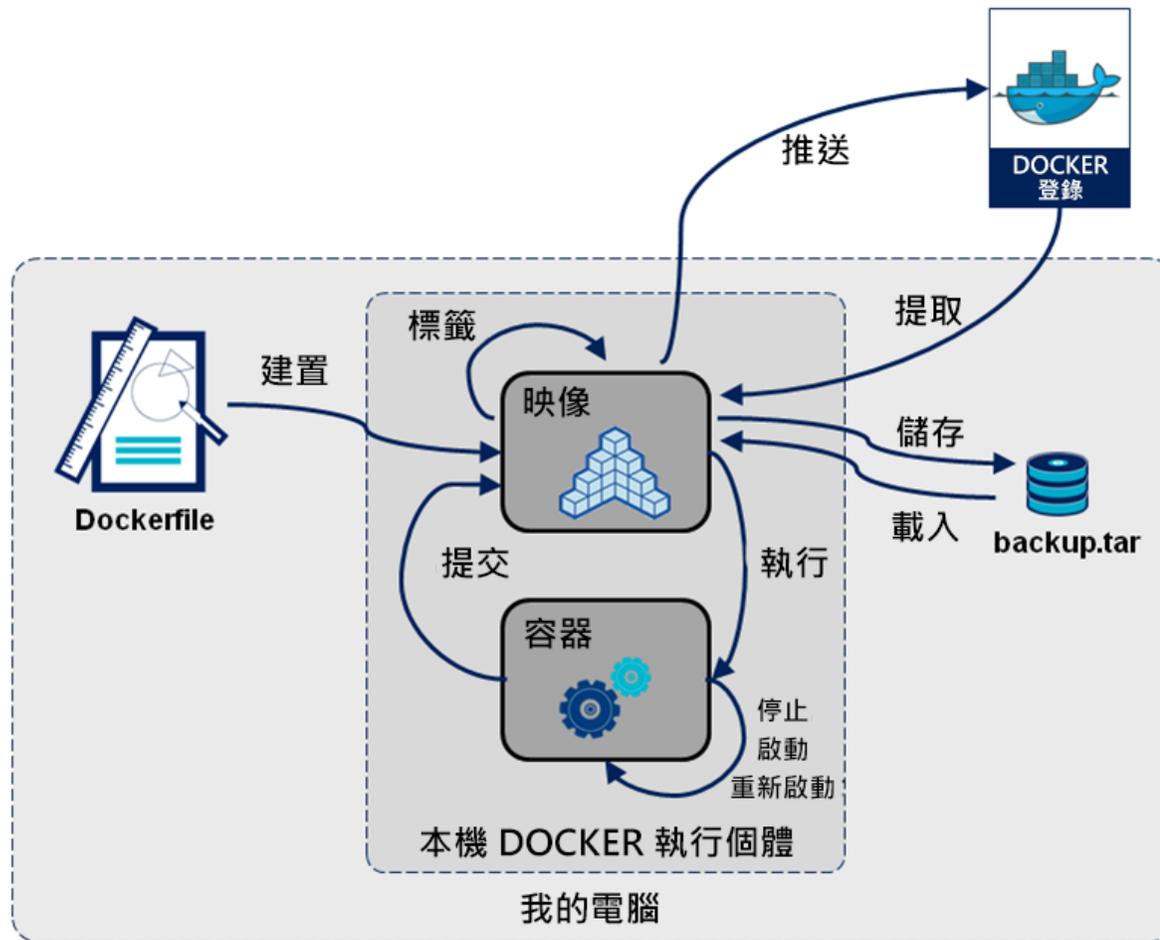
```
~# docker build my_app  
~# docker push my_app  
~#
```

然後在伺服器上：

```
~# docker pull my_app  
~# docker run my_app  
~#
```

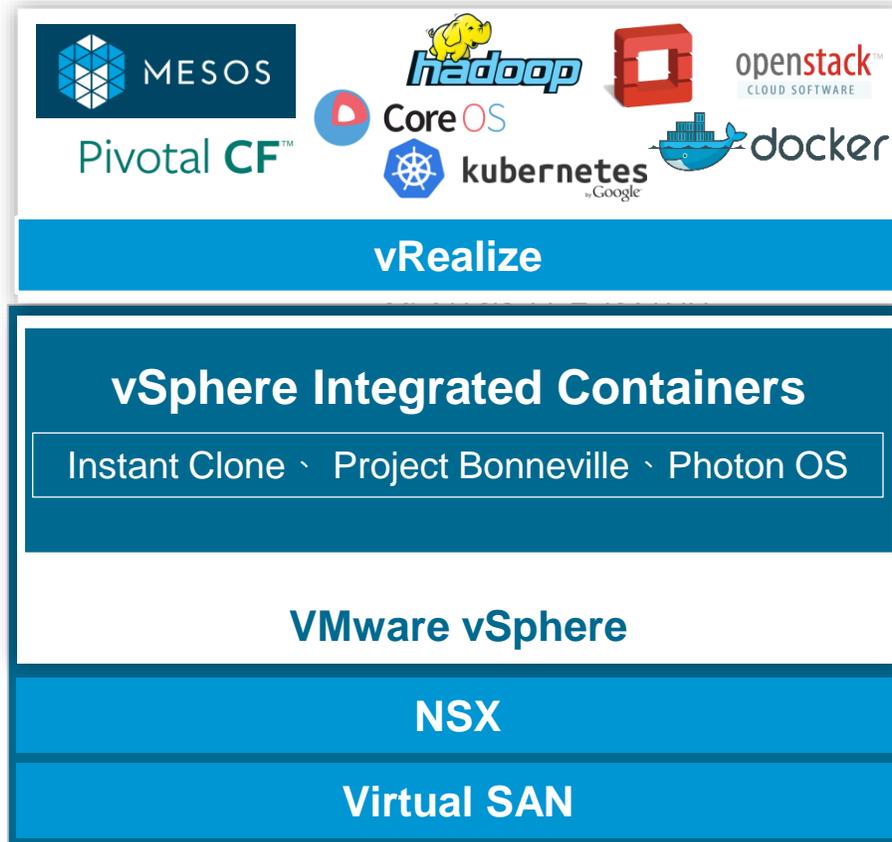
就這麼簡單！！

- 自然而然適合第三平台、微服務
- 這樣讓開發營運工作變得輕鬆許多



# vSphere Integrated Containers 簡介

「原生雲端平台」

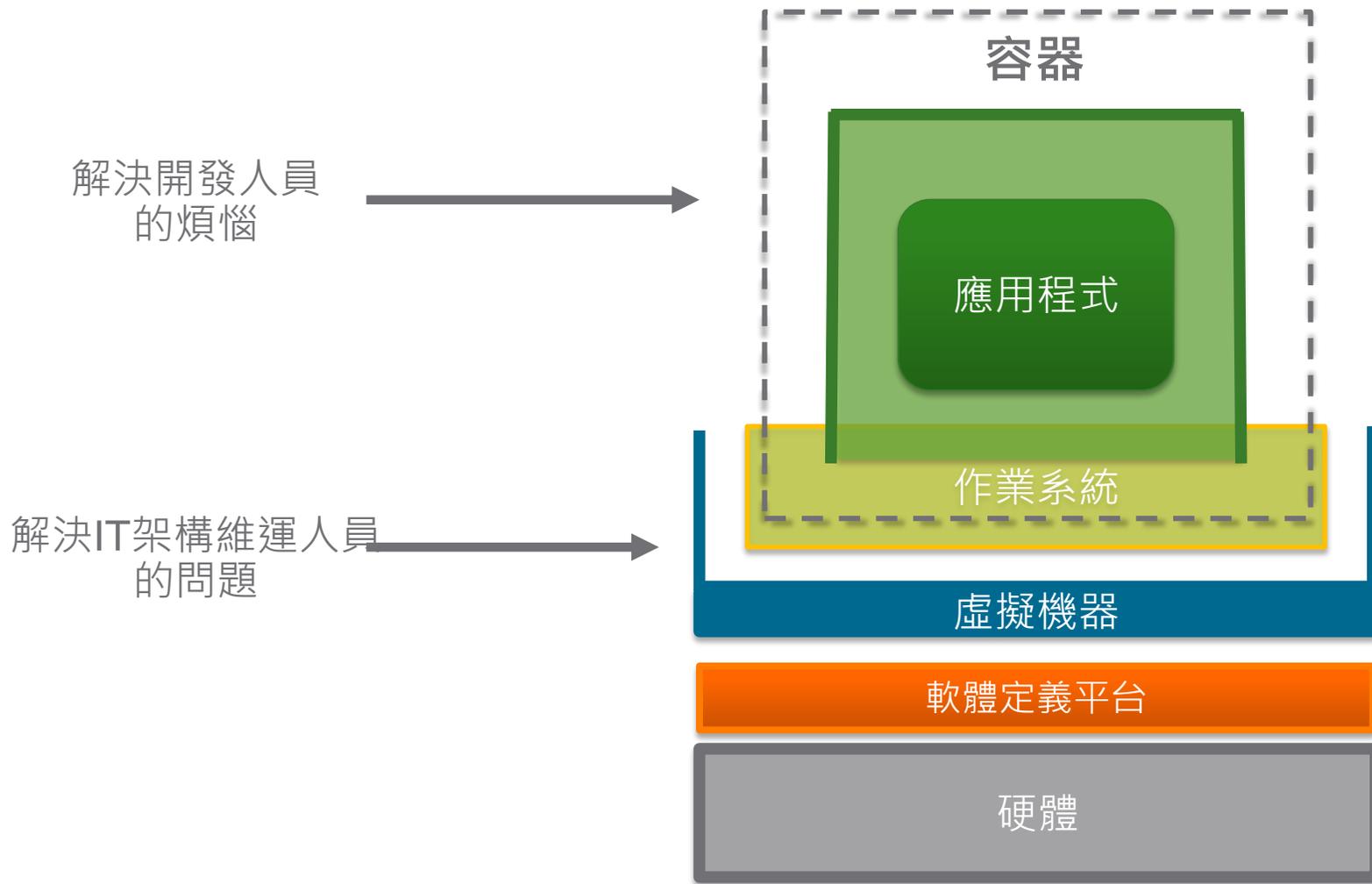


R  
SAN 和 NSX)  
架構

、應用程式服務和硬體平台

運用您現有的投資並實現 邁向原生雲端之路

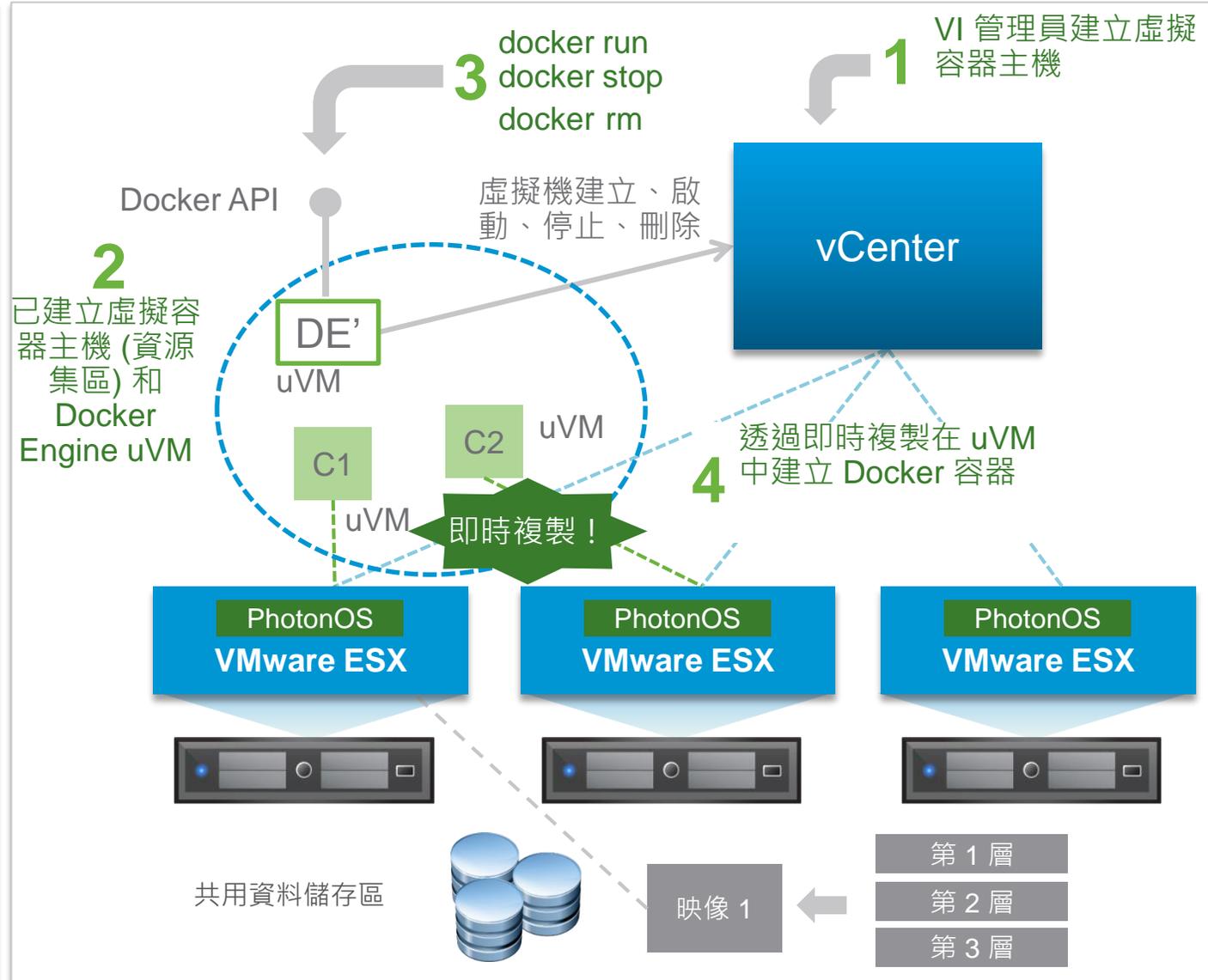
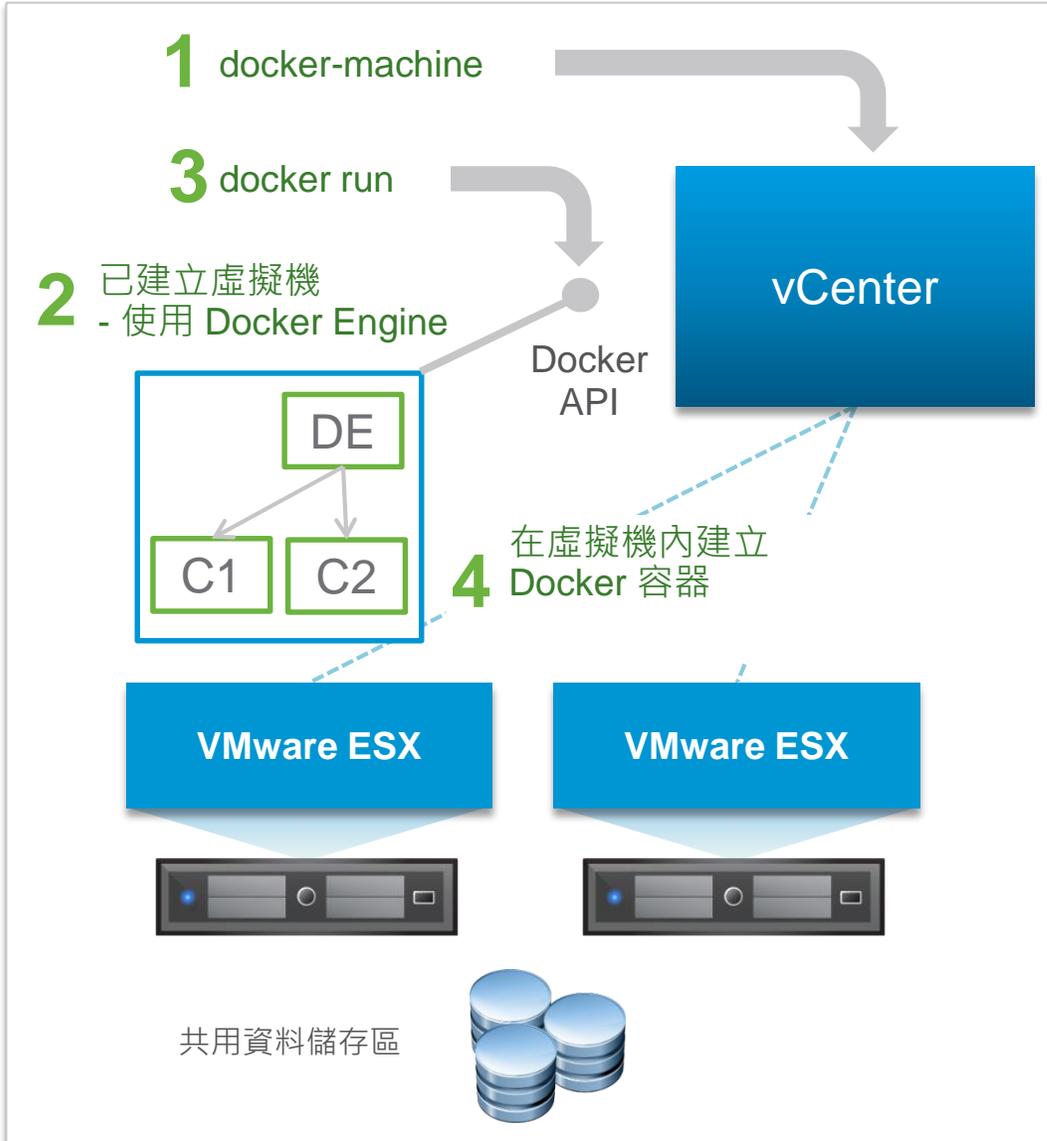
# 鏡子的兩面, 看法不同



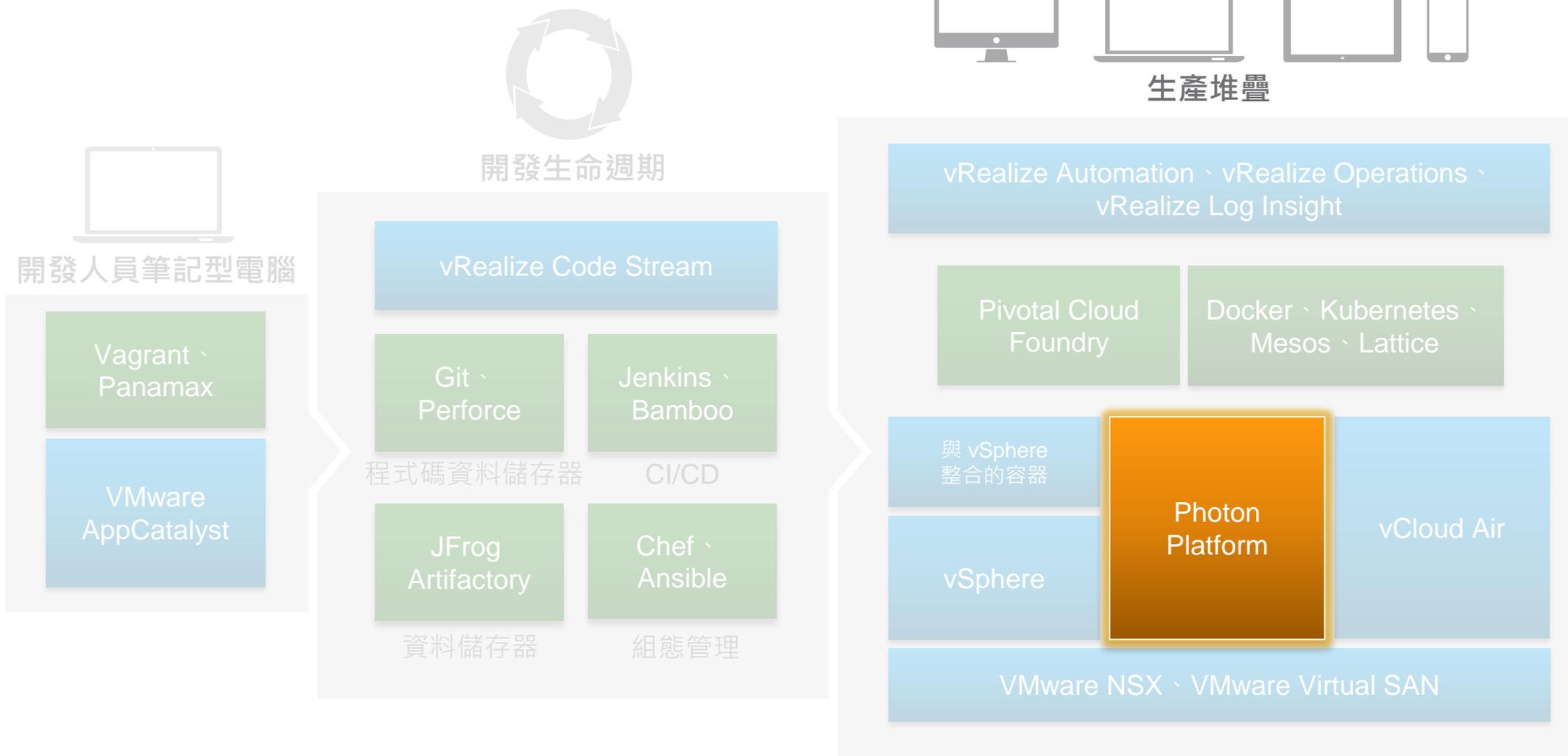
## 對於雙方都有好處

- 隔離與安全
- 具有彈性
- 易於管理
- 責任分工

# vSphere Integrated Containers - 前後比較



# VMware 原生雲端應用程式堆疊



# 另一個方法 – VMware Photon Platform 簡介

「原生雲端平台」



而且專為容器最佳化

- 簡化與最佳化的虛擬化管理程序

Photon Controller

- 優點 – 幾秒內就可啟動
- 擴充 – 支援數十萬個容器

Photon Machine

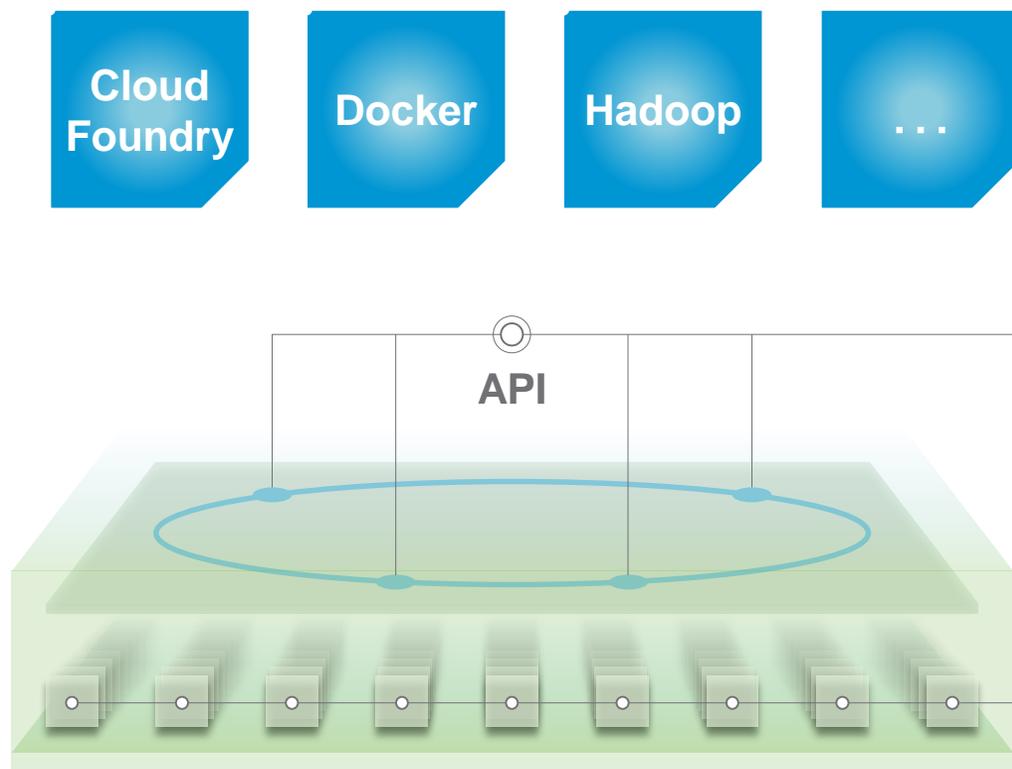
- 移動 – 流暢且動態

VMware Photon Platform

- Photon Controller – 分散式管理、排程、協調、
- Photon Machine – Photon OS、Microvisor

重新設計您的基礎架構來提供原生雲端

# Photon Platform 的涵蓋項目



與新型的開放式原始碼架構及應用程式和資料平台深入整合

## Photon Controller

主機控制器和排程器

分散式、多租戶控制平台  
包含了 Project Lightwave  
單一 API 端點

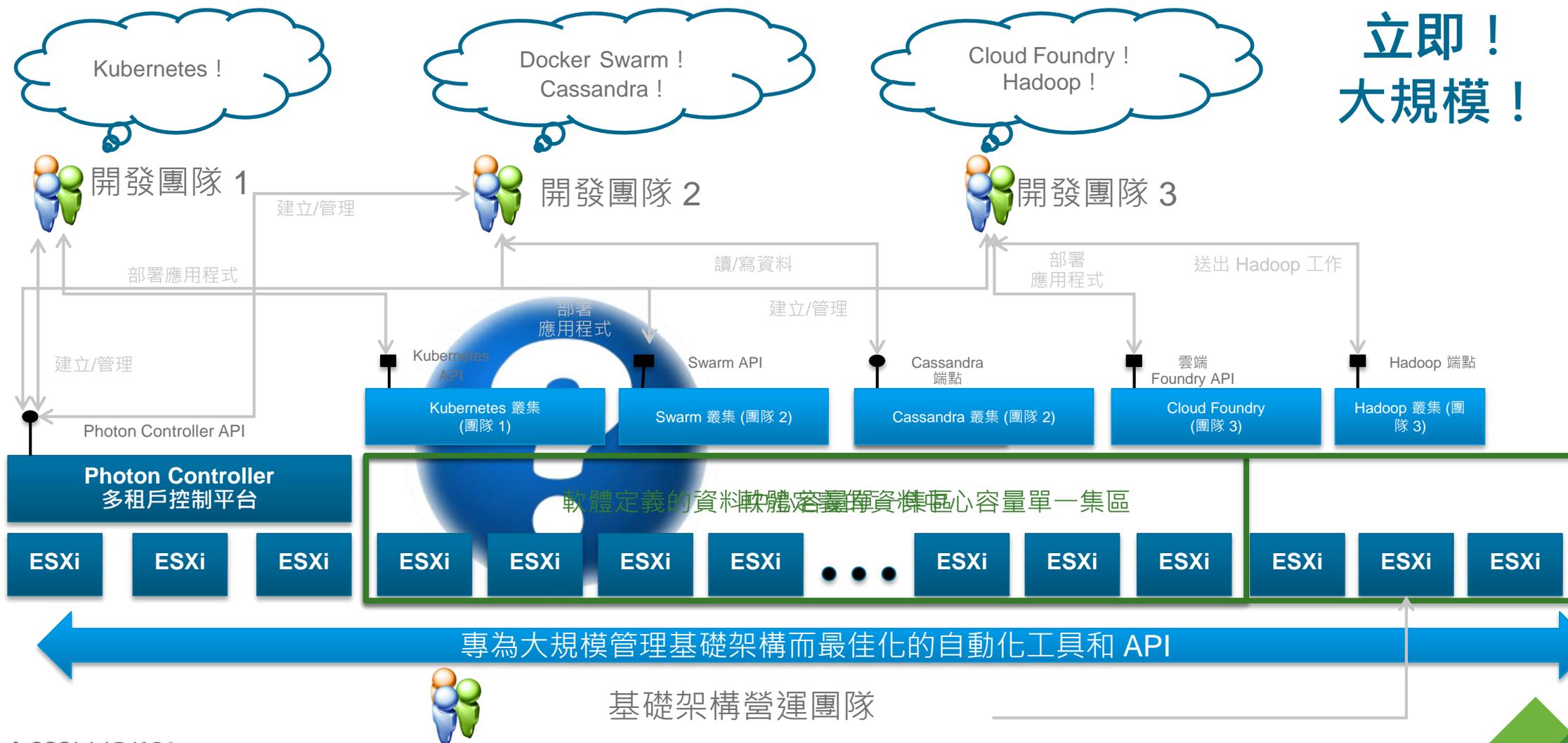
## Photon Machine

運算主機

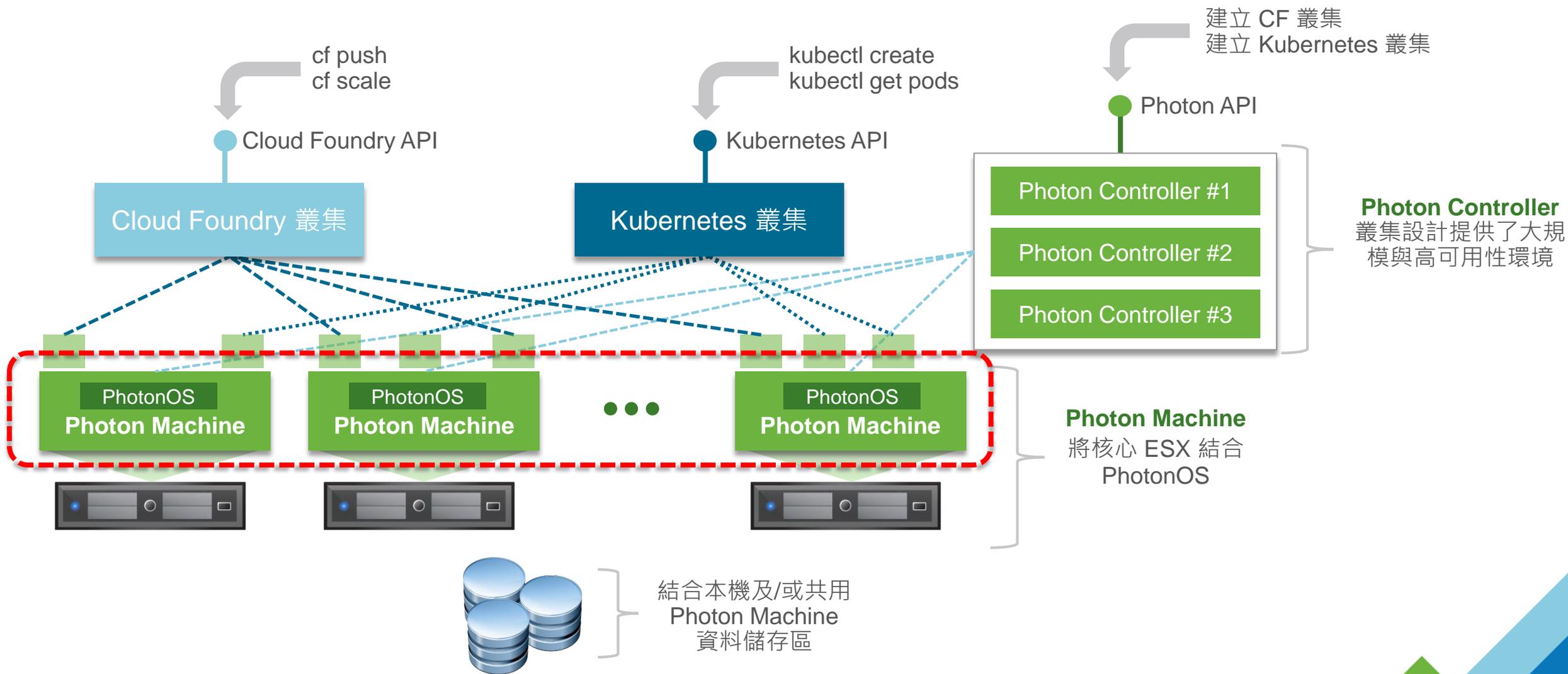
Microvisor 是以 ESX 為基礎，包括 Photon OS  
簡單、可堆疊、可替換的主機

# 什麼是 Photon Controller ?

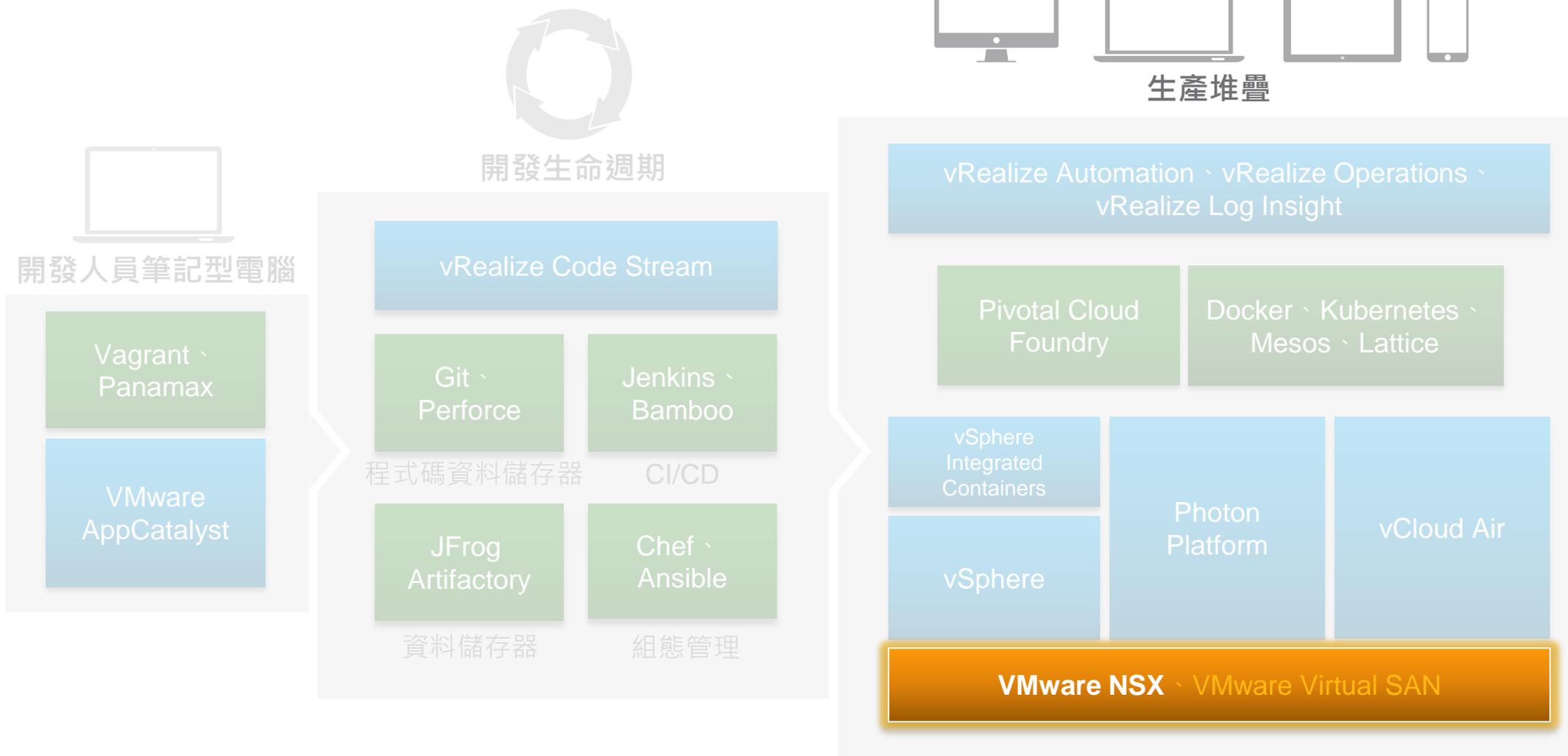
立即！  
大規模！



# Photon Platform 架構



# VMware 原生雲端應用程式堆疊



# 容器網路

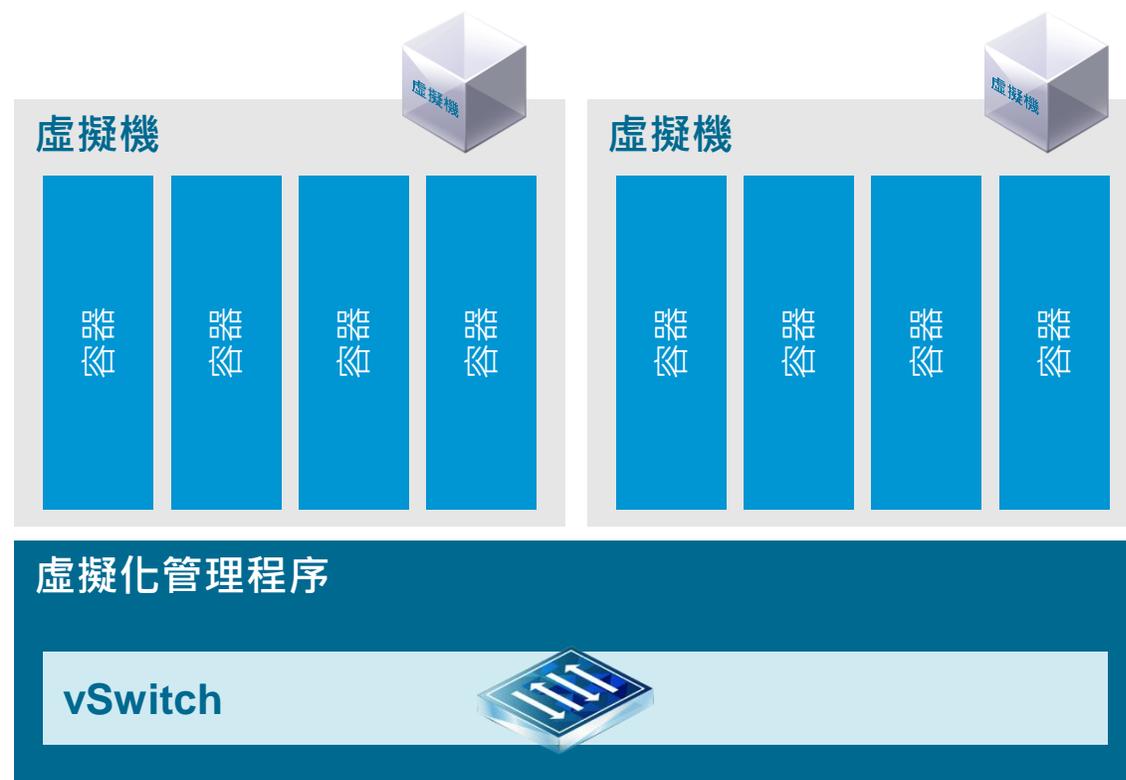
現今的企業模式

## 容器在虛擬機內執行

- 每個安全性網域中的每部伺服器有一個虛擬機
- 容器通常在 NAT 背後
- 無容器層級的網路

## 這樣合理嗎？

- 確實合理...

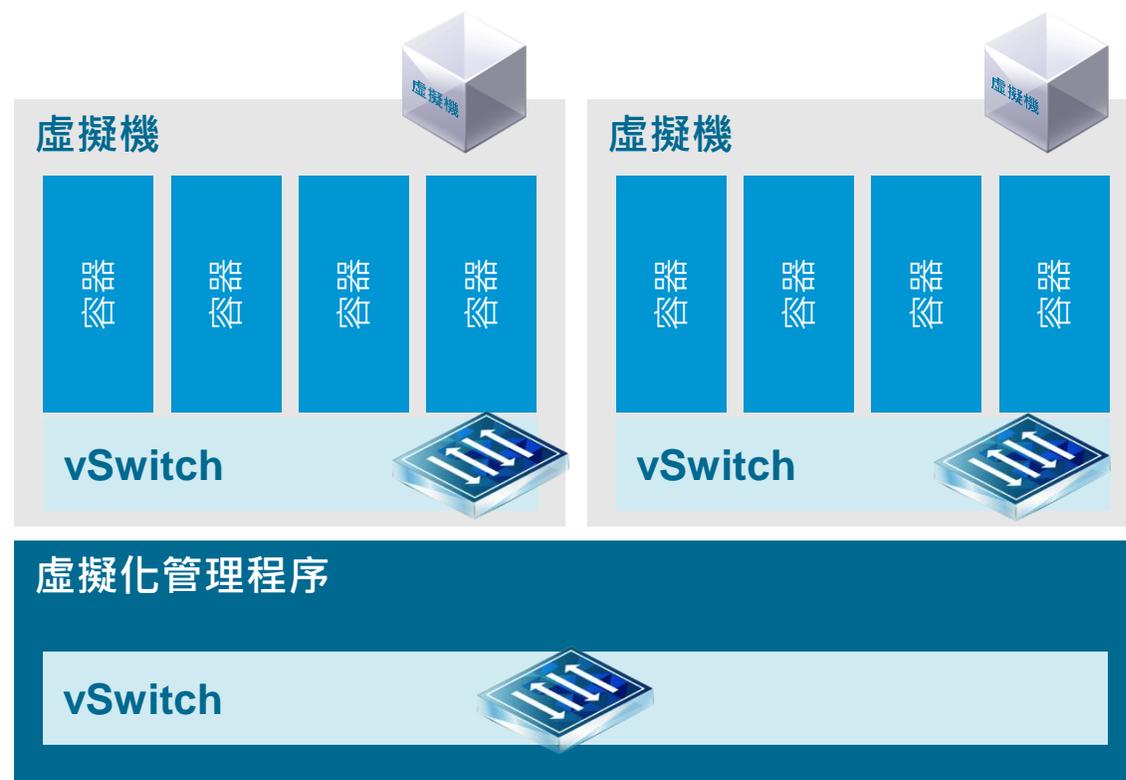


# 容器網路

將來的容器層級透明度

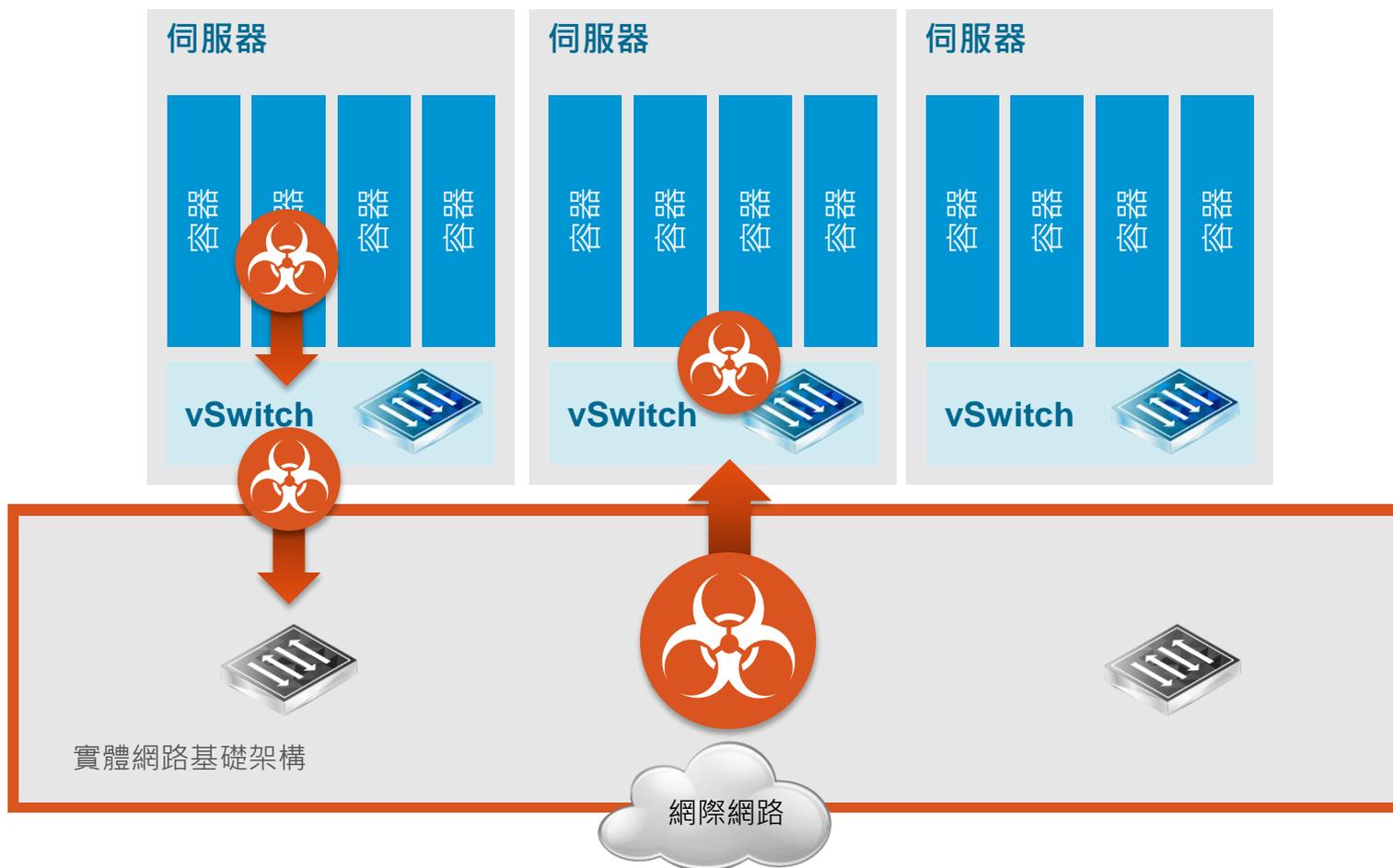
## vSwitch 的兩個層級

- 第一層 vSwitch 位在容器虛擬機內
- 第二層 vSwitch 位在虛擬化管理程序內
- 容器層級的網路



# 容器 – 還需要虛擬化管理程序嗎？

如果沒有虛擬化管理程序，攻擊者可能會擴散

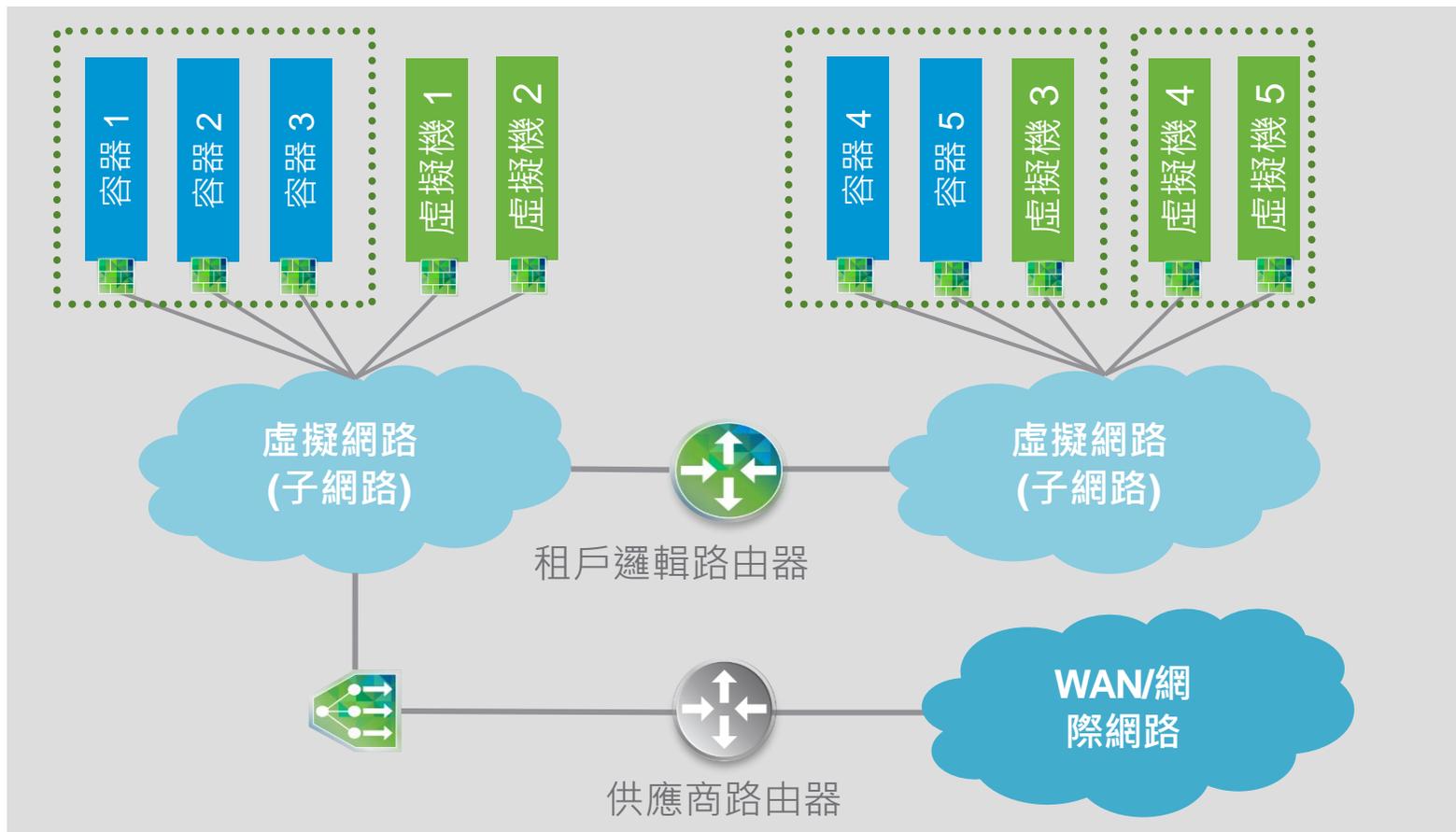


## 如果沒有虛擬化管理程序...

- 攻擊者會危害容器
- 提升權限可獲得容器主機的根存取權限
- 現在可直接存取實體網路
- 可能危害其他實體主機

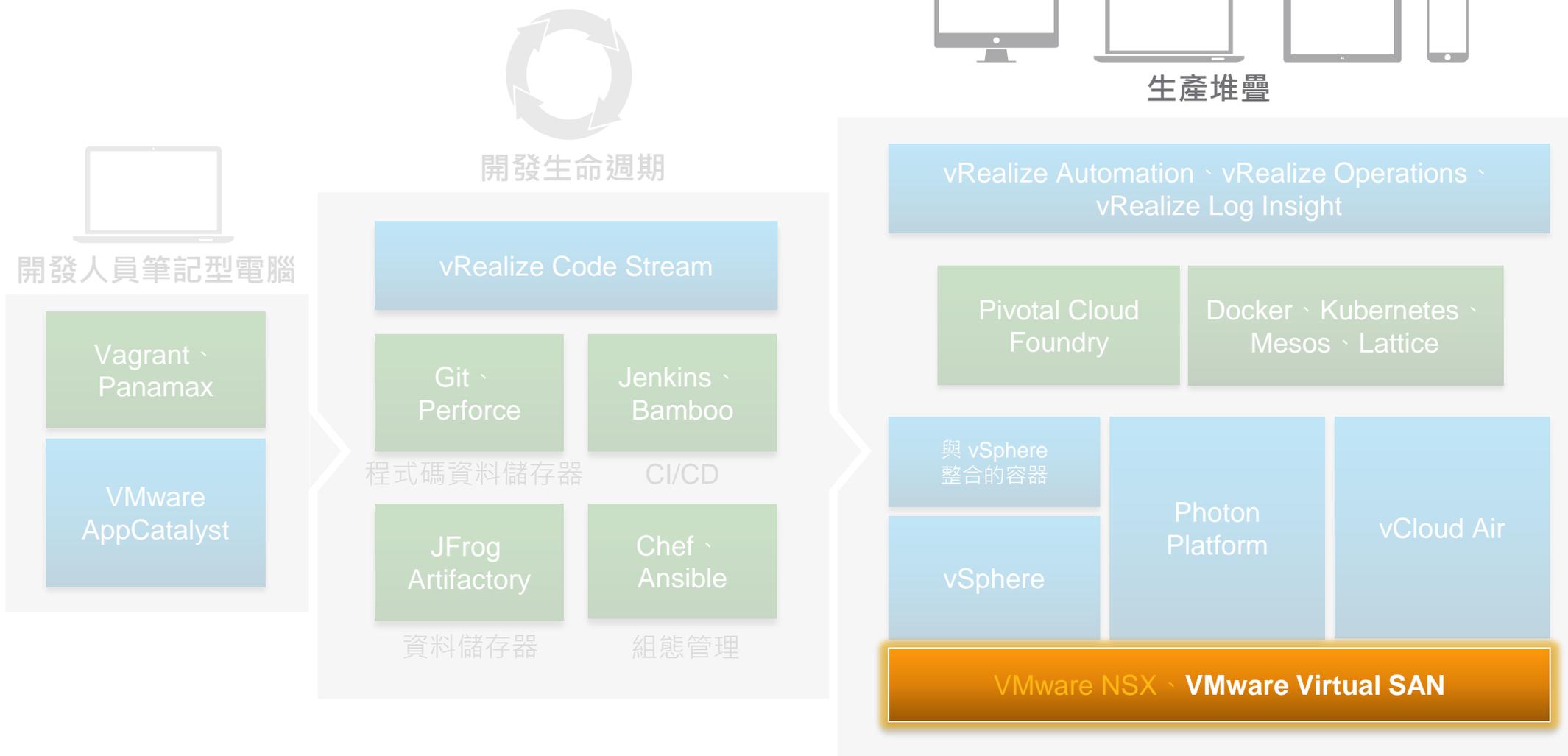
# NSX 和容器

虛擬機和容器適用的 NSX 網路與安全性服務



- 虛擬機
- 容器/Docker
- 防火牆
- 負載平衡器
- ⋯ 安全性群組

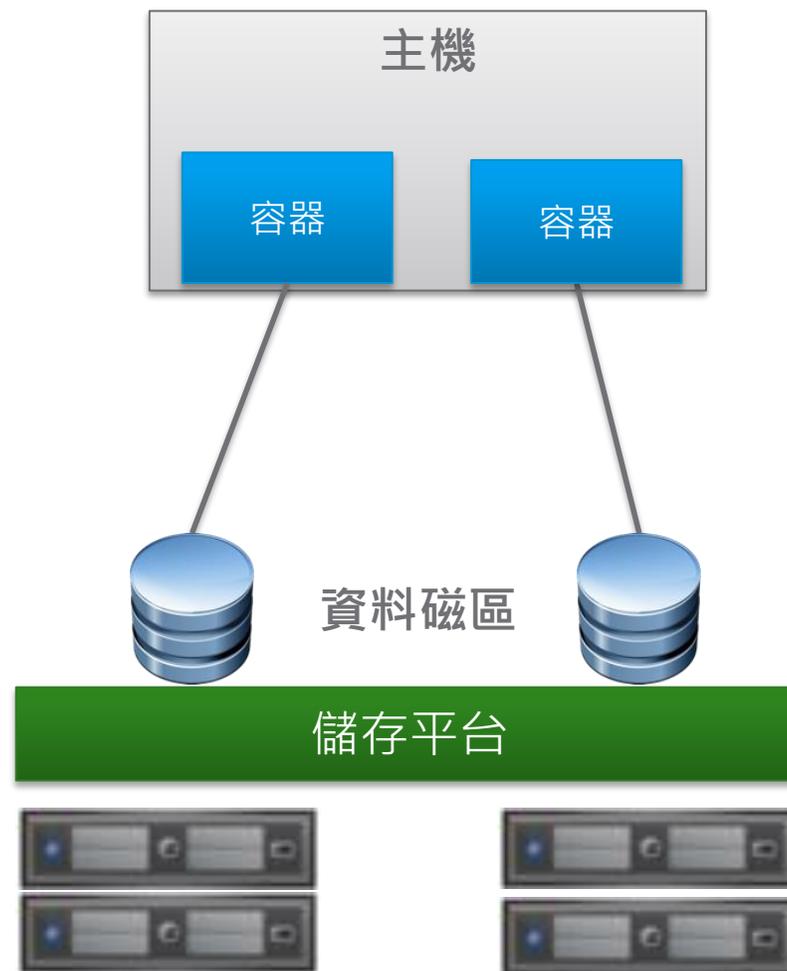
# VMware 原生雲端應用程式堆疊



# 容器資料磁區

- 使用情況
  - 包含本機容器適用的持續性資料
  - 以資料夾形式出現在主機檔案系統內 (例如「/mount/yourdata/」)
  - 可位於主機或外部儲存裝置上
- 優點
  - 管理及保留您的續態資料
  - 利用儲存平台資料服務

但是...當您在不同的主機之間移動應用程式時，要如何保留資料呢？



## 適用於 Flocker 1.0 的 vSphere 驅動程式正式推出！

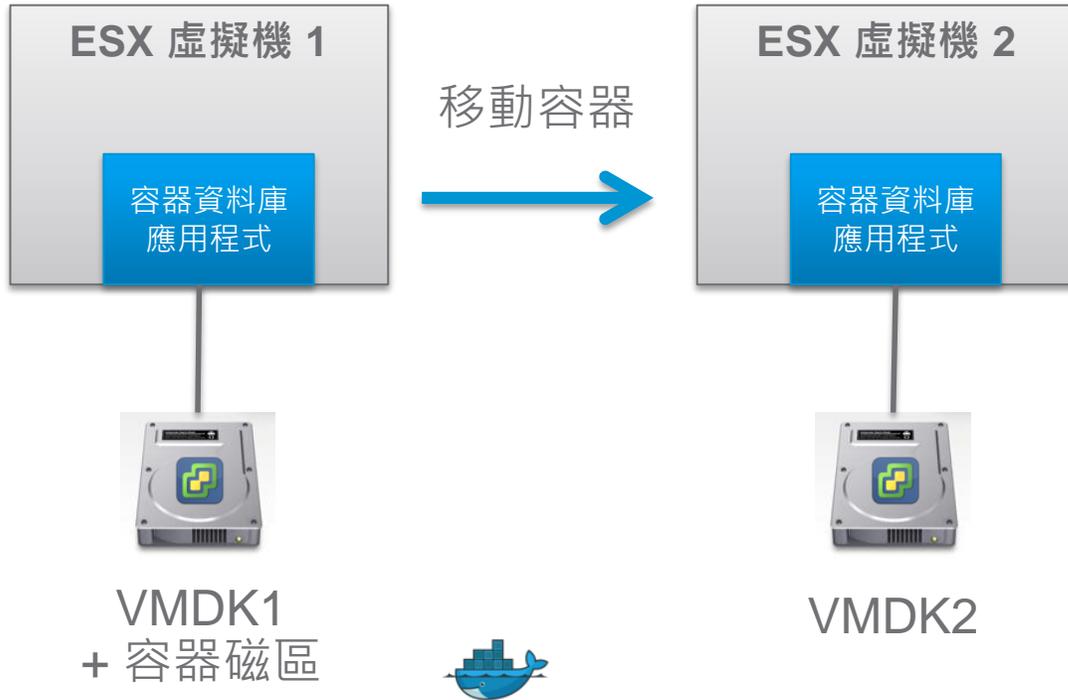
- 使用開放式原始碼 Flocker 軟體在現有的 vSphere 部署上執行容器化續態應用程式
- 在不同的 ESX 虛擬機之間移動容器和連接的資料磁區
- 與所有 vSphere 儲存相容 (Virtual SAN、VVOL、VMFS、NFS)
- 簡單且明確的安裝/設定/部署程序
- 免費！
- 可在 <https://github.com/vmware/vsphere-flocker-driver> 取得

vmware®



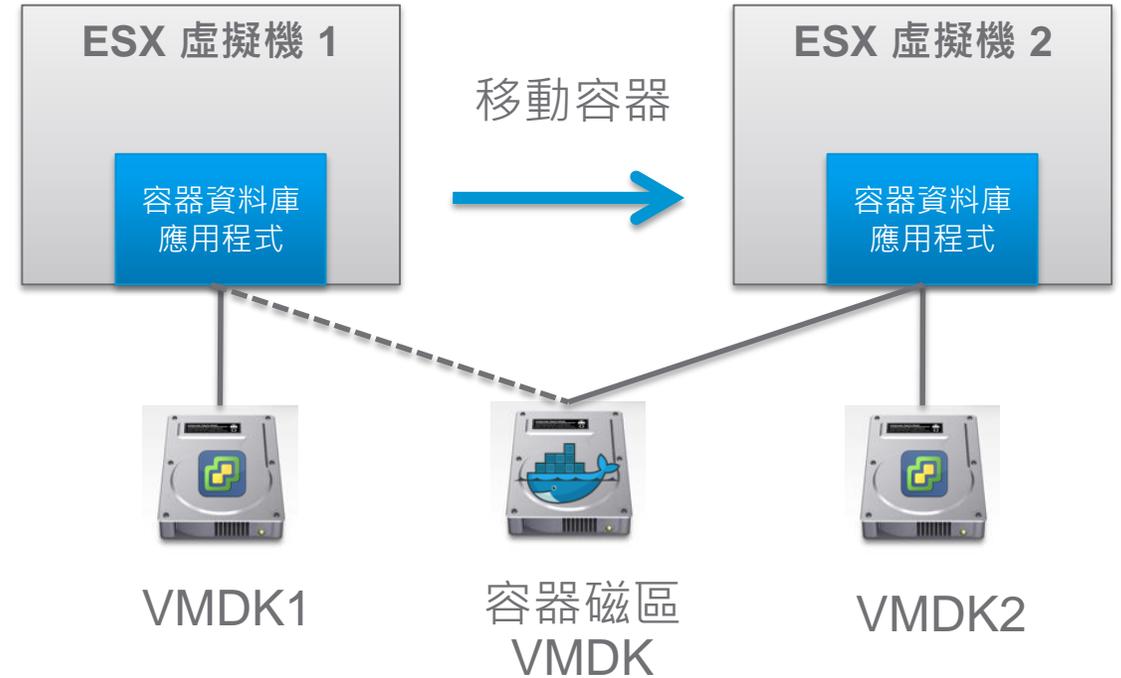
Flocker™  
by ClusterHQ™

## vSphere 上的原生 Docker



容器移動時，資料磁區在  
主機 VMDK 上保留不動。  
資料庫會在新的虛擬機上啟動，  
不含任何資料。

## vSphere + Flocker



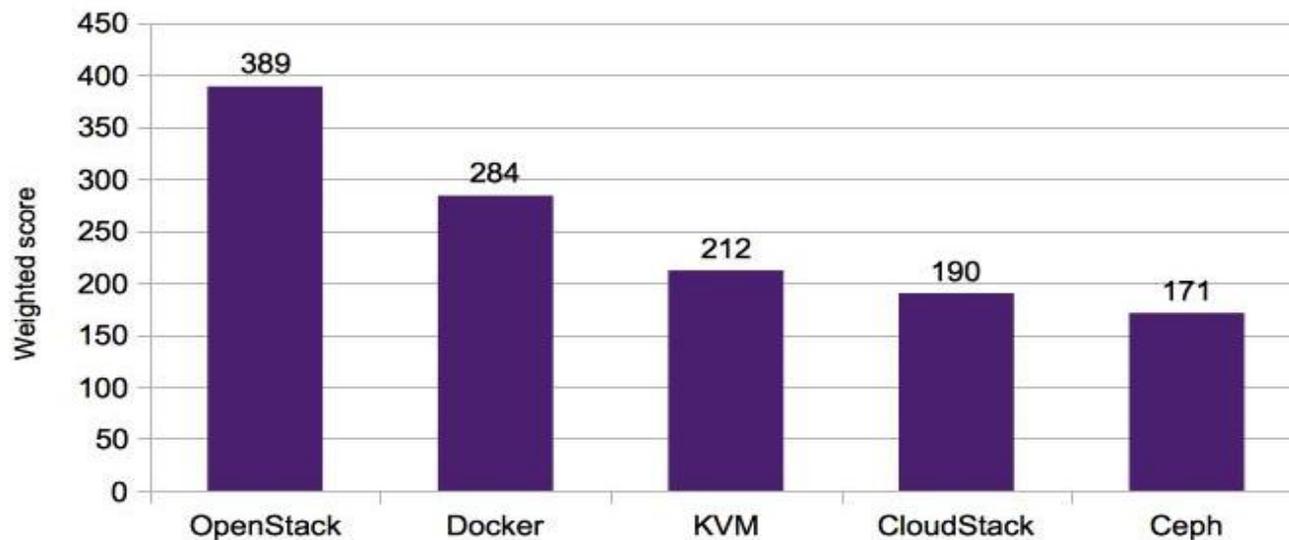
資料磁區已儲存在個別的 VMDK 上。  
容器移動時，VMDK 會一起移動。  
資料庫會保留原有的資料！

# vSphere Integrated OpenStack

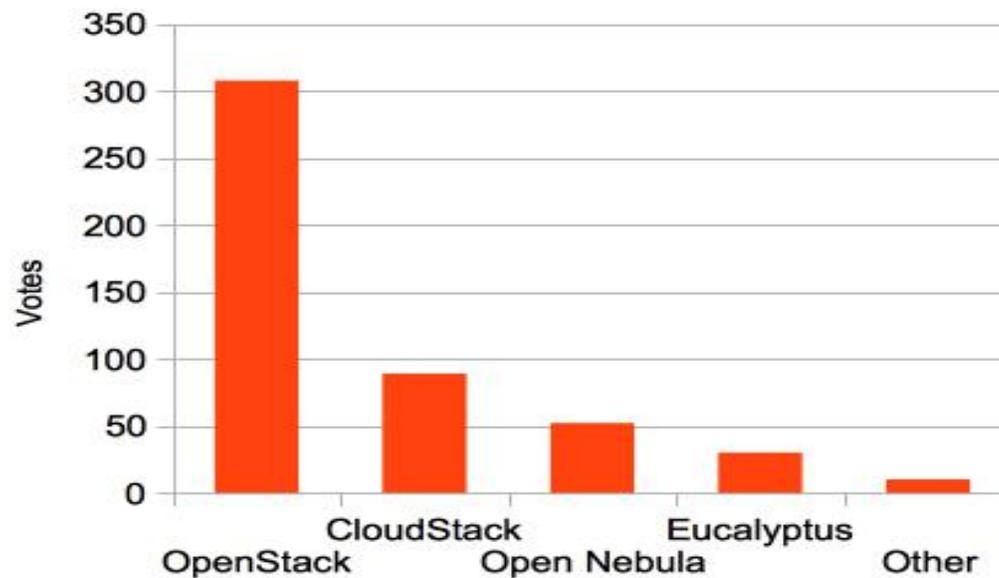
Solution Introduction

# OpenStack——2014年度最受歡迎的開源專案

## 最受歡迎的開源雲項目



## 最受歡迎的開源IaaS項目



# 何謂Openstack?



一個開放雲的架構框架

由非營利的Openstack基金會支持

模組化, 各自獨立的專案所組成  
(圖形平台, 存儲均為各自的專案)

Python 語言

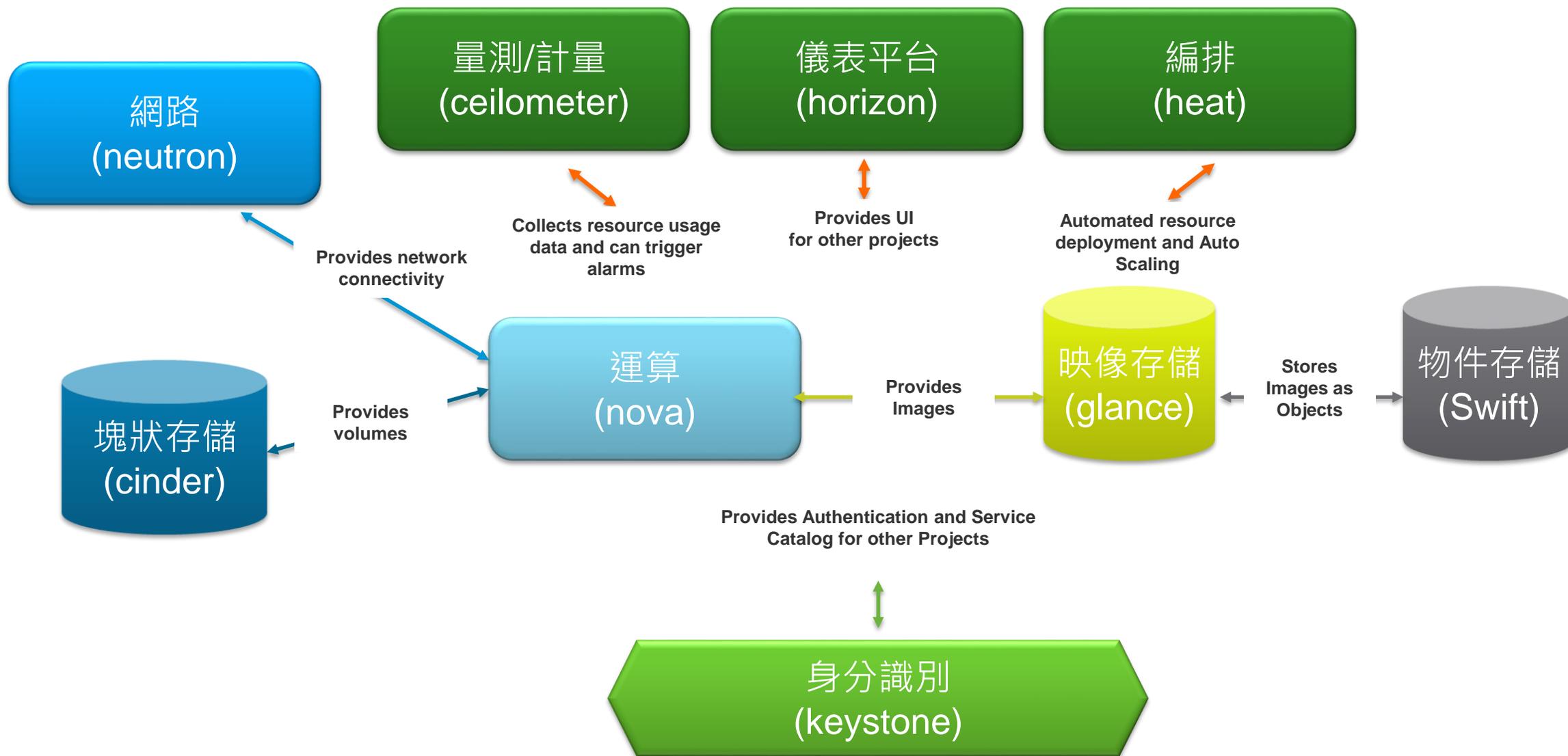
依據生命週期的專案  
*(Incubated, Integrated, Core)*

供應商可以針對不同的專案項目提供驅動程序  
*(whenever applicable)*

六個月為一周期的更新  
*(for all projects)*

字母順序編碼  
(A)ustin  
(B)exar  
....  
(I)cehouse  
(J)uno  
(K)ilo  
(L)iberty

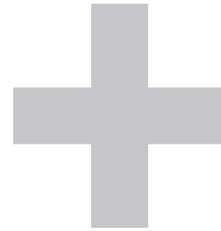
# OpenStack 專案的關係圖



# VMware Integrated OpenStack (VIO)



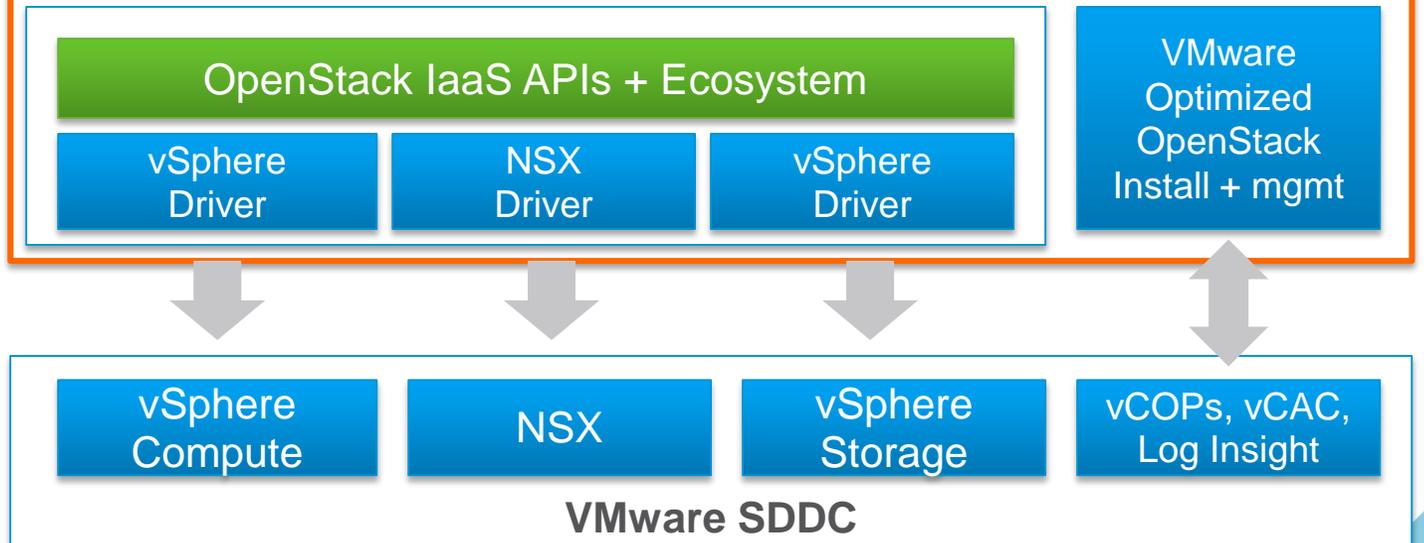
雲應用開發人員喜歡的一個全行業的API和工具生態系統



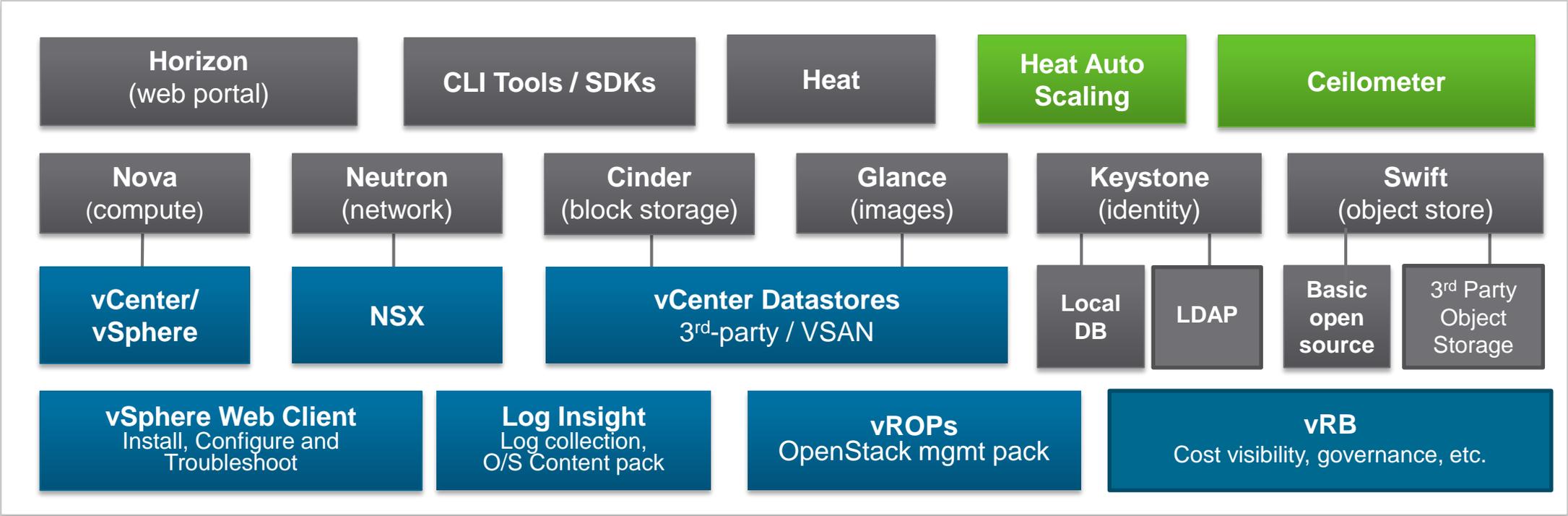
企業IT人員所熟悉如何運營地行業領先資料中心虛擬化技術，並具有可靠、簡單、高效等特點，以及對企業內部現有IT基礎設施具有非常廣泛的相容和整合能力

兩者的融合，將是企業進行新一代IT資訊化建設的福音，“IT即服務、IT促進業務創新”。VMWare讓Openstack更加可靠、簡單、經濟、高效，Openstack將讓VMWare更加與開發人員接近以及加速業務創新

## VMware Integrated OpenStack (VIO)



# OpenStack Projects (Components) Included with VIO 2



Current

Included OpenStack Components:

Integrated VMware Technologies:

# Key New Features of VIO 2.0

Feature	Benefit
<b>VIO 2.0 will be Kilo Based</b>	<ul style="list-style-type: none"><li>• Kilo is the latest stable community release.</li><li>• VIO 2.0 delivers latest OpenStack to customers</li></ul>
<b>Automated Upgrade Support with simple Rollback</b>	<ul style="list-style-type: none"><li>• Upgrade is one of the biggest challenges of OpenStack</li><li>• VIO 2.0 provides automated, reliable and UI driven approach to upgrade from 1.x to 2.0</li><li>• Tenant workloads are not impacted at all. And cloud services are only down for limited time window.</li><li>• Existing services and configurations are left completely untouched so rollback is very simple.</li></ul>
<b>Heat Auto Scaling Support with Ceilometer</b>	<ul style="list-style-type: none"><li>• Customers can design applications that can scale up/down automatically and address unpredictable load situations</li></ul>
<b>vSphere &amp; NSX key features exposed via OpenStack</b>	<ul style="list-style-type: none"><li>• LBaaS is supported in VIO 2.0 with NSXv</li><li>• vSphere Reservations, limits and affinity settings provide better control for VM placements.</li><li>• Non-Native Image formats such as Qcow2 are now supported in VIO</li></ul>
<b>OpenStack Control Plane Backup &amp; Recovery</b>	<ul style="list-style-type: none"><li>• Take backups of the management server and the OpenStack database</li><li>• Restore any component of control plane (ex: controller, load balancer, etc.)</li></ul>
<b>Horizon Branding &amp; Localization Support</b>	<ul style="list-style-type: none"><li>• VIO now supports 6 additional international languages</li><li>• Horizon portal can be customized to company branding.</li></ul>

# Overview of VIO 2.0 Key Features

VIO 1.0 – **Icehouse** Based  
Nova, Neutron, Cinder, Glance, Horizon,  
Keystone, Heat

VIO 2.0 – **Kilo** Based  
+Ceilometer

## OpenStack Deployment & Operations

- Production Ready Architecture
- Automated Deployment & Configuration
- Automated Patching

- Automated Upgrade
- Backup & Recovery Procedures
- Lifecycle management of passwords, certificates etc

## VMware SDDC Differentiation

- Nova with vCenter driver
- Glance & Cinder with VMDK driver
- NSXv Neutron plugin
- NSXv addresses Key Neutron Challenges of scale, HA and stability

- Support for vSphere Reservations, Limits, Affinity, Anti-Affinity
- Qcow2 image format support
- LBaaS
- Heat Auto-scaling with Ceilometer & LBaaS
- Simplified Integration for vROps OpenStack Mgmt Pack

## Security & Usability

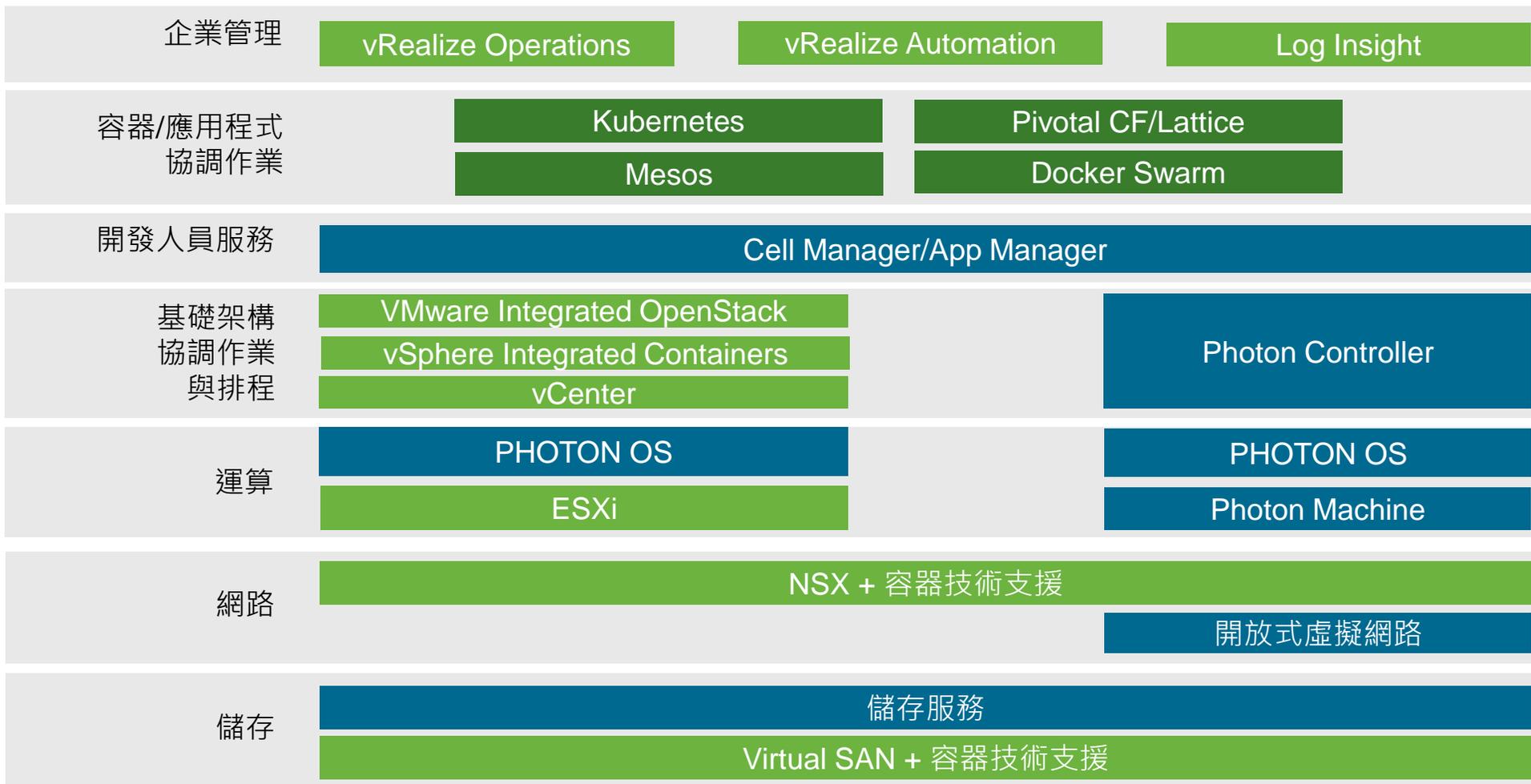
- Authentication w/ Active Directory
- Encrypted Passwords in config files

- Branding of Horizon Portal
- Globalization & localization to additional languages

Mar  
2015

Sep  
2015

# VMware 原生雲端基礎架構堆疊



資料儲存器 + Lightwave

**READY**  
FOR **ANY**  
vForum2015