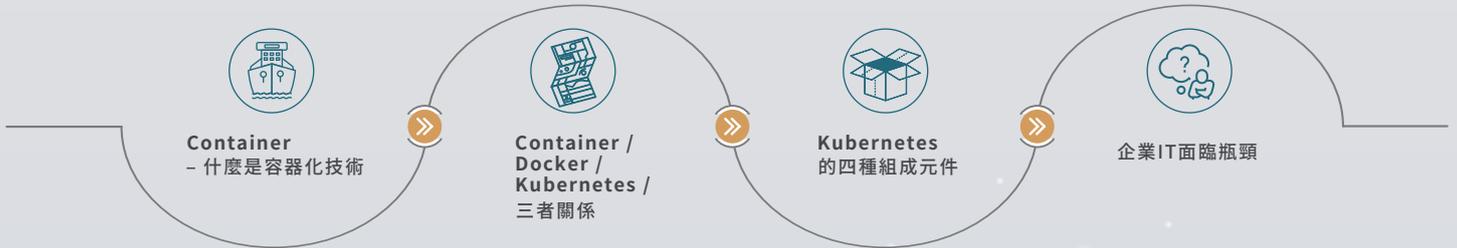


你知道這是什麼嗎？

這是Container最近好像很紅，本篇電子報將針對Container、Docker、Kubernetes說明其功用與關係，以及未來在維運管理型態的趨勢，雖然我不是數學家，但這解決方案聽起來還不錯吧。



本期我們想告訴你的事



迎向大航海時代|Container - 容器化技術介紹

Container在中文被稱為集裝箱或貨櫃，能夠裝載許多物品並便於運送，延伸到資訊科技領域中被稱為『容器』，通常將應用服務 (Application) 與其資源封裝成輕量化的映像檔，使開發、維運團隊可輕易攜帶與搬遷。

在虛擬架構中，為了增加硬體資源的利用率，會在硬體層上建立一層虛擬化平台，並於平台中建立與運行多個虛擬機器 (VM)，彼此分離獨立並共享硬體資源。雖然透過虛擬化的作法可以大幅節省實體機器的數量與有效利用硬體資源，但由於虛擬機器是以作業系統 (OS) 為中心，每個虛擬機器都需要一個Guest OS的配置，因此會造成幾種問題：



佔據系統資源

每個虛擬機器都需要使用底層硬體資源建立 Guest OS，佔據記憶體與硬碟空間，並且還需要透過Hypervisor進行虛擬機器與底層資源的溝通。



啟動時間漫長

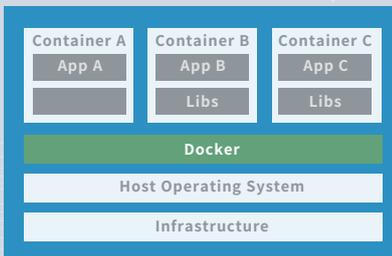
每個虛擬機器都需要安裝Guest OS，使得映像檔需涵蓋OS相關內容而過於肥大。



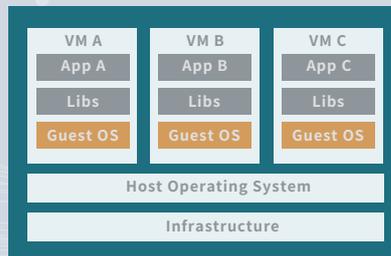
機器搬遷瓶頸

搬遷時需要考慮OS的版本與規格，增加搬遷困難度之外，也無法隨意攜帶。

Container



Virtual Machines



但早在1982年就出現的容器化技術，為何在近十年才開始受到企業關注呢？在於早期容器化做法的門檻較高，且並無統一標準，直到2013年 dotCloud這家公司釋出一套可將容器標準化的平台Docker，並受到各大科技巨頭的支持後，使Docker成為目前主流的容器標準。



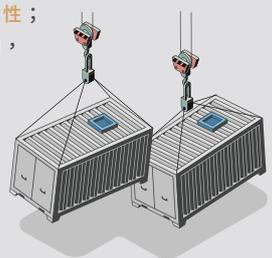
海洋中的藏寶圖 | 淺談Container/Docker/Kubernetes 三者關聯性

在剛開始接觸容器化時，常看到有人講Container、有人講Docker，容易被專有名詞搞混，到底兩者有什麼差別？而Kubernetes又是什麼？

Container

容器化 (Containerization) 是一種連同硬體作業系統與運行服務做虛擬化的技術，讓運作在平台上的容器 (Container) 共享一個Host OS的資源，而容器是容器化技術具體化的一個實例 (Instance)，就像虛擬化技術所實踐出來的實例稱做為虛擬機一樣。

容器化著重在應用程式的開發、攜帶，封裝應用程式的相依資源與運行環境，強調輕量、快速與彈性；虛擬化則側重於硬體設備的整併與資源運用，強調資源利用率與高可用性。



Docker

Docker是由dotCloud所發佈的一個容器標準化的開源平台，並以Docker為名成立新公司推廣，如同虛擬化需要有虛擬層一樣，在Hypervisor上運作虛擬機器，容器化也需要有一個平台可以運行容器與資源分配，稱為Docker Engine。

Docker可以將應用程式與其相依資源封裝成Docker映像檔，使其可被重複利用或移植於其他環境，同時提供Dockerfile記錄建立容器的每一步驟，只要有Docker Engine，就可以將Docker映像檔或Dockerfile轉換成容器。



Kubernetes

Kubernetes是一個可以自動化部署與管理容器的工具，是由Google設計並開源系統。就像你可以透過VMware vCenter去部署與管理虛擬機器一樣，可以藉由Kubernetes達到大量容器的部署與調動，目前市面上包含SUSE的Rancher、Red Hat的OpenShift，以及VMware的Tanzu都是依循Kubernetes所開發出來的容器管理產品，而除了Kubernetes之外，也有其他的管理工具，包含Docker自家推出的Docker Swarm，以及Apache的Mesos。



K8S的水手工具箱 | Kubernetes的四種組成元件

Kubernetes需要知道的四個基本的原件：
Cluster、Master Node、Worker Node、Pod

Cluster 任務的小船

叢集 (Cluster) 是大家較為熟悉的概念，一個叢集裡可以包含很多的節點 (Nodes)，例如你可以將三台伺服器組成一個叢集，在叢集裡可以做到資源共享、負載平衡與高可用性等等。

Master Node 任務指揮中心

是負責發號施令與管理其他Worker Node的角色。

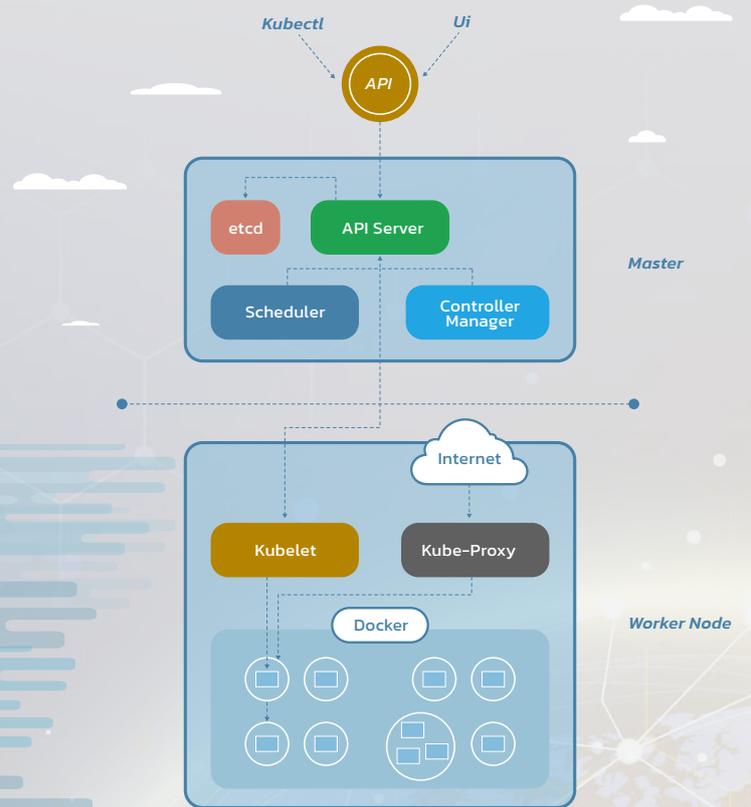
Worker Node 任務作戰單位

Worker Node就像底下的執行單位一樣，負責處理各種大小事。

Pod 任務最小運作單位

被包含在Worker Node之中，一個Pod對應一個應用服務，而Pod裡面包含一個或多個容器，這些容器共享資源與網路，並彼此透過localhost的Port進行溝通，由於Pod的資源有限，所以通常建議一個Pod只存在一個容器。

每一個Pod都會有一個yaml設定檔，類似於身分證的概念，設定檔裡包含要部署在這個Pod的容器資訊。





汪洋大海中 求助無門的IT人員



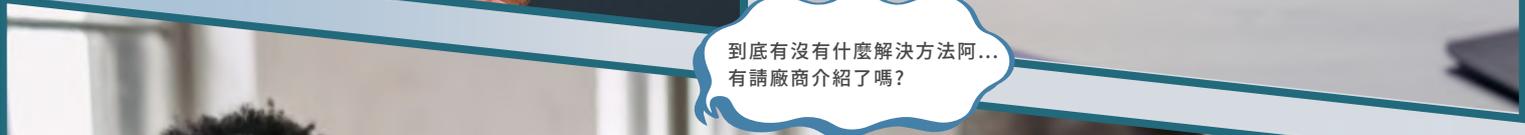
公司之後要用 Docker做開發，去了解一下！

好的，老闆...

又來了：
又要加班學新的東西
真是麻煩...



管設備就夠累了...
還要搞DevOps...
真的要累死我...

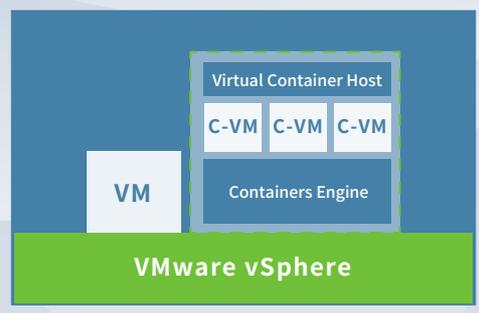


到底有沒有什麼解決方法阿...
有請廠商介紹了嗎？



我這禮拜每天都10點才離開公司...

老闆不補人，
要我們管兩套平台，
大家自己保重...



VMware Tanzu

容器與虛擬機混合部屬的最佳選擇

首部曲 - 容器應用懶人包

二部曲 - 實現虛擬機與容器混合部屬

To be continued...



你這個Tanzu是容器跑在虛擬化上欸，多了一層效能會不好吧？(兇)

而且現在有設備可以支援Tanzu嗎？(鄙視)

雖然可以虛擬機跟容器並存，但是兩種操作邏輯又不同，你要怎麼做？



不用擔心~

Tanzu效能上除了比實體機還快8%之外，還導入了Namespace概念讓維運和開發人員對虛機和容器進行操作，更重要的是包含HPE Alletra、Nimble、Primera和SimpliVity都支援Tanzu哦！

想知道Tanzu怎麼做到的話，記得要繼續看下去！



在上一篇中我們提到了企業在虛擬和容器兩套環境管理上的困難，不論企業對於組織策略發展是遷移至容器平台，亦或是採取兩套環境的共存管理方式，VMware推出的VMware with Tanzu，將虛擬化與容器化整合為共享平台，讓維運與開發團隊共同協作虛擬機和容器，解決企業在移轉過渡期或是共存管理的困境。因此本篇將針對Tanzu，對於維運與開發團隊在協同合作上的三大效益，以及目前HPE支援容器應用界面的設備進行介紹。



破除虛擬機的效能魔咒

Tanzu打破外界對於容器運作在虛擬化的效率擔憂，於vSphere環境中增加了一種ESXi原生的容器運行時環境CRX(Container Runtime for ESXi)，並針對CRX虛機做了專門的優化，在測試報告中交出比物理伺服器還要快上8%的成績單，實現容器在實體機運作上相同的性能水準。



能承載虛擬機與容器的飛天魔毯

在VMware全新升級的控制平台 – Supervisor，維運管理與開發人員皆可用YAML檔實現虛擬機和容器的管理調度和開發。對維運團隊來說，不需要改動現行的運維流程與管理工具，就能實現對於現代應用的支援；而對開發團隊而言，不需像過往依賴於維運團隊來為他們創建所需要的基礎架構，並可使用熟悉的kubectI部署資源物件獲得所需資源，大大提高對於業務的回應速度。



以應用程式為中心的管理邏輯

Tanzu導入了Kubernetes中Namespace概念來管理現代化應用，因為現代應用往往是由幾十個、甚至更多的虛機和容器所組成的混合型應用，與傳統的虛擬化基礎架構在管理上的複雜程度不同，往往不僅費時費力，並且很難保證一致性。因此在Tanzu中每一個應用都有一個對應的Namespace，其中包含了組成該應用的所有虛機和容器，可以針對整個Namespace來指定該應用的運行參數，讓管理人員能夠以Namespace為中心進行管控，協助管理人員針對不同應用進行更為細節的調整。



為Tanzu而生的HPE儲存設備

對於能支援容器化應用，並且持續運行傳統的工作負載，HPE目前有四款儲存設備可與Tanzu共同打造共享平台，包含：

| | | | |
|--|---|--|---|
| <p>HPE PRIMERA</p>  <ul style="list-style-type: none"> • Tier 0 Mission Critical • FC/iSCSI • 100%Availability • Workload Consolidation • CSI | <p>HPE Nimble Storage/dHCI</p>  <ul style="list-style-type: none"> • Tier 1 Business Critical • FC/iSCSI • Agile workloads • Efficiency & Simplicity • CSI | <p>HPE Alletra Storage/dHCI</p>  <ul style="list-style-type: none"> • Tier 0 and Tier 1 • FC/iSCSI/NVMe* • Cloud Managed • Full gamut of workloads • CSI | <p>HPE SimpliVity</p>  <ul style="list-style-type: none"> • SimpliVity:CSI • dHCI:HPE CSI Driver • Edge to Core • FCD/CNS • Scalable/Disaggregated |
|--|---|--|---|

在這幾款HPE儲存設備中皆提供CSI Driver與Tanzu上的容器應用進行界面，通過容器的佈署取得底層儲存設備的空間資源。