Enterprise Linux 實戰講座

7-11 全年無休的 Linux －談 RHEL High-Availability Solution

由於網際網路及電子商務的盛行改變了企業 Business Model，現在企業的顧客已不是侷限在 Local 的客戶。全世界都有可能是您的客戶，加上人們生活型態改變，所以全年 365 天 24 小時隨時都可能有商機產生。

當資訊系統發生無可預期的停機事件時，除了企業損失生產力、客戶與收入。還可能遭遇到進一步受罰、訴訟、聲譽不良的影響。此外，擁有一個永不打烊(High Availability)的運作模式，更是一個不容忽視的競爭優勢和服務關鍵。所以現在許多企業除了要求伺服系統能在上班時間提供可靠與連續運轉的服務，更希望能對客戶提供 24 X 7 全年無休的服務。

簡介

在商業的 Unix 市場中，高可用性 (High Availability) 是銷售 Unix 伺服器解決方案的關鍵。事實上每個 Unix 供應商都有他們自己的高可用性軟體解決方案，例如 IBM 的高可用性叢集軟體解決方案，就是 AIX 上的 HACMP (High Availability Cluster Multi-Processing ） 。其他主要的 Unix 供應商像 HP，Sun，DEC 和其他的供應商有許多類似的軟體解決方案可用。

High Availability 是現今銷售 Unix 給許多企業的關鍵。特別對於需要 web-based 和其他必須一整年，每週七天，每天 24 小時可用的伺服器。至於新竄起的網格運算市場而言更是如此。但是在 Linux 一直沒有很成熟的 HA 解決方案，即便是 RedHat 在 Advanced Server 2.1 上提出的 HA 解決方案和其他的 Unix 廠商的 HA 解決方案也有一段不小的差距。

不過，隨著 RedHat Enterprise 3.0 的推出，Red Hat 在其上推出一個重量級企業應用軟體「**RedHat Cluster Suite**」，使得情況有所改觀。Cluster Suite 包含兩個技術：Cluster Manager 和 Linux Virtual Server。Cluster Manager 是 HA 的最佳解決方案，只要兩台伺服器和共用的外接儲存設備，透過 Cluster Manager 來控制伺服器所執行的服務，就可輕鬆達成 HA 的目的。不過 Red Hat Cluster Suite 不包含在 RHEL 3.0 中，它必須額外購買。而且只支援 RHEL AS 和 ES 版。(圖 1)

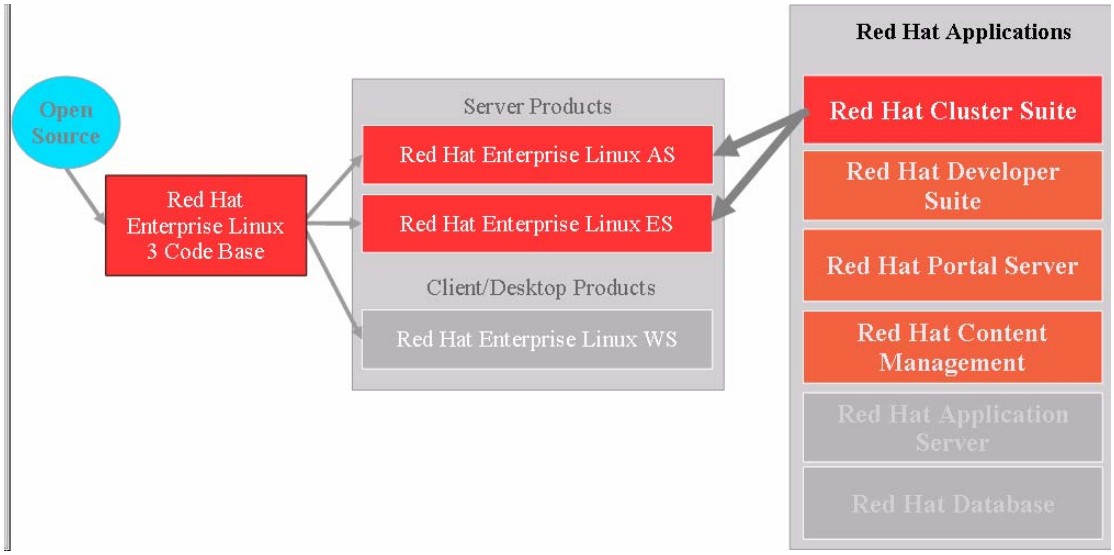

圖 1：RedHat 產品架構圖

筆者一直認為 Linux 要成為企業重要關鍵的伺服器，甚至取代高階 Unix 的伺服器，穩定可靠的 HA 的解決方案是不可或缺的。但 RedHat 一直沒有推出筆者認為足以媲美 AIX HACMP 的產品。總算隨著 RedHat Enterprise 3.0 的上市，也推出了 Red Hat Cluster Suite。它比之前 AS 2.1 穩定性更高，而且也較容易設定。本篇文章先為讀者介紹 Cluster、HA 等相關技術及觀念，下一篇文章筆者將利用 RedHat Cluster Suite 實作 HA 解決方案。

Clustering 分類

筆者有時遇到客戶或學生詢問 Cluster 叢集系統及 High Availability 相關問題，總覺得 Cluster、HA、High performance 這幾個名詞常讓人混淆。就筆者的看法，「**所謂 Cluster 就是由一台以上的機器為了某種特定需求所組成的架構**」，根據不同的需求我們可將 Cluster 分為以下三種，並對應 RedHat 由何種軟體提供相關功能。

- **High availability clusters**
  增加伺服器和以網路為基礎的應用程式的高可用性及備援性。
  由 Cluster Suite 中的 **Cluster Manager** 技術提供
- **Load Balancing clusters**
  將服務需求分派給多台伺服器，可視系統負載隨時彈性增加伺服器
  由 Cluster Suite 中的 **Linux Virtual Server** (Piranha) 技術提供
- **High performance clusters (HPC)**
  提供同步運算及平行處理的能力

Cluster Suite 不提供 (另外有 lam、pvm 套件，規劃由 WS 擔綱)

本篇文章最主要介紹有關「**High availability cluster**」相關技術，其它兩種 Cluster 不在此次探討範圍內。RedHat Cluster Manager 是以 open source Kimberlite cluster project 為基礎來發展而來，讀者可參考下列網址得到相關的資訊 http://oss.missioncriticallinux.com/kimberlite/。

一個完善的 HA 解決方案必須具備下列特性：

- Reliability (可靠性)
- Availability (可用性)
- Scalability (擴充性)

High availability clusters 功能

HA Cluster 通常必須提供下列功能：

- 具備 Hardware redundancy for fault tolerance 功能

「Redundancy」筆者看到有些文章譯成「冗餘、冗、過多、多餘、贅..」真的是很怪，其實 Redundancy 的意義就是「備援」。至於 fault tolerance 是指「容錯」，所謂「容錯」代表當系統能夠回應非預期性的故障時，可在容許狀況及範圍之下仍然可以繼續運作，無須做任何的的切換或轉移。在 HA Cluser 中「Hardware redundancy for fault tolerance」相關解決方法可參考表 1。

表 1：HA Hardware redundancy for fault tolerance 解決方法

問題	解決方法
磁碟故障	採用硬體 RAID
RAID 控制卡故障	雙重的 RAID 控制卡以提供存取 RAID 資料
Heartbeat 失效	乙太網路的 channel bonding 以及 fault tolerance
電源來源失效	備援不斷電系統(UPS)

問題	解決方法
所有失效狀況下的資料毀損	電源切換器或硬體為基礎的 watchdog

- 「**SPOF**」(Single Points Of Failure)

  HA 的基本準則是硬體都必需使用具有 Redundancy 的硬體設備，例如 RAID、Dual Power。最主要原因是為了避免造成「**SPOF**」(Single Points Of Failure) 的情形發生。什麼是「**SPOF**」? 所謂「**SPOF**」是指當某個零件故障會造成整個系統無法正當運作，那麼這個零件就是整個系統中的 Single Points Of Failure。

  例如在 Client/Server 環境中，一個伺服器系統中的單一網路卡即為這個伺服器的 SPOF。同樣的，連接到外部儲存系統的單一 SCSI 配接卡是 SPOF。 如果一群伺服器中的一整個伺服器故障，並且這個故障的伺服器不能被輕易且快速的由其他伺服器置換，那麼對這個伺服器群組或叢集而言，這個伺服器就是一個 SPOF 。

  這個解決方法十分簡單：配接卡只要藉著在一個伺服器中設置兩張卡，並確認主要配接卡失效時備援配接卡會成為 active ，就可以達成備援 (redundancy)。

- 提供 "*failover*"機制

如果一台伺服器停機或故障, 另一台伺服器可以接手(**takeover**)啟動應用程式

- 以及以下情況發生時仍可以讓應用程式正常運作
  - 硬體故障
  - 系統維護及升級

一個典型的 High availability clusters 的架構應如圖 2，這兩台伺服器彼此如何溝通，還有常見 Share Disk Storage 又有那些?接下來筆者便為各位介紹這些相關技術。

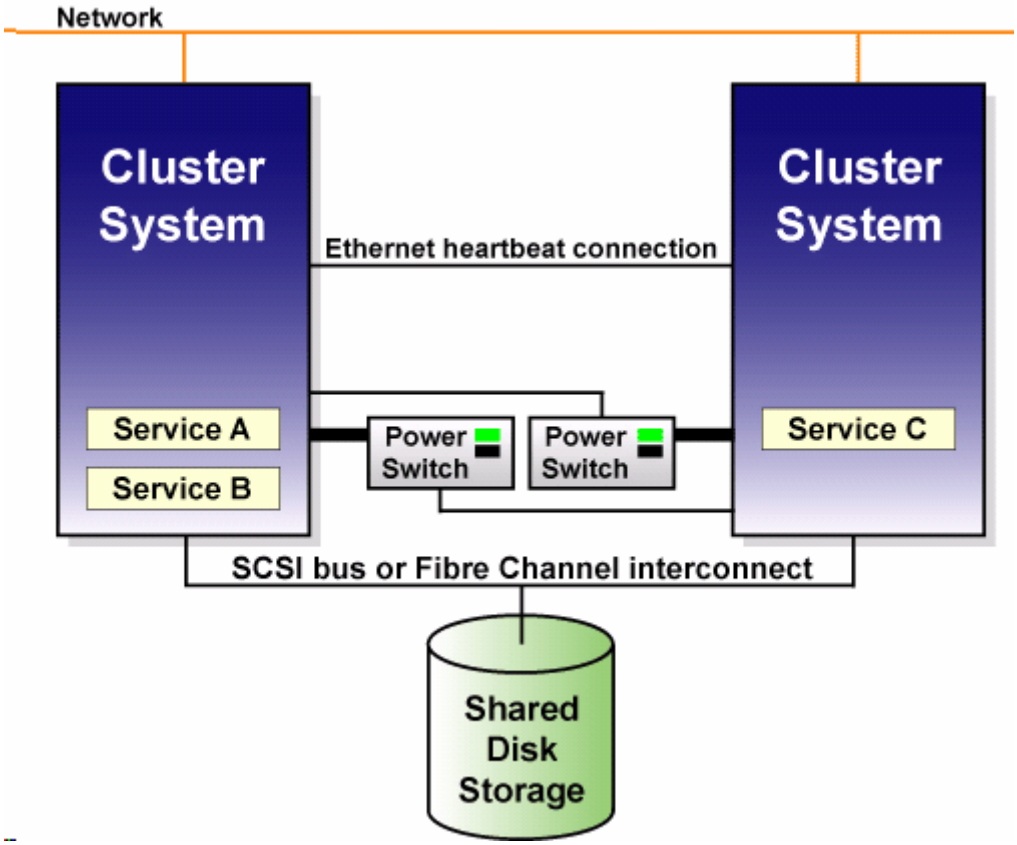

圖 2：High availability clusters 架構圖

High availability clusters 中的通訊機制

Clusters 內部的通訊機制來確保資料的完整性以及當發生 Cluster 成員故障情況時能及正修正 Cluster 的行為，Cluster 使用這些機制來做以下的事情：

- 控制系統何時能成為一部 Cluster 成員
- 決定 Cluster 系統的狀態
- 當問題發生時，控制 Cluster 的行為

RedHat HA Cluster 相關通訊機制如下：

- Ethernet 的 heartbeats

「heartbeats」這個名詞在各家 HA 解決方案中可都是少不了他的身影，「heartbeats」是「心跳」的意義，筆者覺得老外用這個字真的是非常貼切，所謂「heartbeats」的機制就是用來檢查另一台伺服器是否仍正常運作的機制。是不是很像人類利用有無心跳來判斷一個人是否還活著。

Cluster 成員之間是由點對點的 Ethernet 網路線來連線的，每一部成員將會定期地向這些網路線發出 heartbeats (ping) 訊號，Linux-HA 使用這個資訊來幫助找出成員的狀態，並且確保正確地 Linux-HA 操作。

- 共用的(quorum)分割區

在 Linux-HA 中每一部伺服器將會定期地寫入一個時間戳記(time-stamp)與系統狀態到 primary 與 shadow 的共用分割區，也就是位於 Share Disk Storage 中的某塊空間 (raw partition) 。 每一部成員將讀取由其他成員所寫入的系統狀態與時間戳記(time-stamp)，以找出它們 的狀態是否更新。 成員將會試著從 primary 共用分割區讀取資訊。 假如該分割區已毀損，成員將會從 shadow 共用分割區讀取資訊，並且同時修復 primary 分割區。 將透過錯誤檢查 (checksums) 維護資料一致性， 而且分割區間的任何不一致性都會自動地修正。

- 遠端的電源開關監視

Cluster 中的成員將會定期地監視遠端電源開關（假如有的話）連線的使用狀況，成員使用這個資訊來幫助找出 其他 Cluster 成員的狀態，電源開關通訊機制的完全失效並不會自動導致 failover，假如電源開關無法 power-cycle 一部當機的系統，將不會執行任何的 failover，因為 Cluster 的基礎架構無法保證成員目前的狀態。

假如一部成員發現來自另一成員的時間戳記(time-stamp)並沒有即時更新，它將會檢查 heartbeat 的狀態，假如向其他成員發出 heartbeat 訊號的動作仍在執行中，Cluster 管理程式將不會採取任何動作。 假如一部成員在一段時間後都沒有更新它的時間戳記(time-stamp)，而且也不回應 heartbeat pings 的訊號，該部成員將被認定已停止運作。

只要有一部伺服器可以寫入共用的分割區，即時所有其他的通訊機制都失效了，叢集仍將維持運作的狀態。

請注意，在某些兩部成員的設定中，共用的分割區只被用來當作一個備援，網路成員的演算法是對 Cluster 成員 是否正在使用中的主要決定因素。 在這個設定中，不更新時間戳記(time-stamp)的一部成員絕不會 failover 的發生， 除非 clumembd 回報該成員已停止運作。

Share Disk Storage

常見的 Share Disk Storage 技術有下列三種：SCSI、SSA、Fibre

- **SCSI (Setting Up a Single-initiator SCSI Bus)**

如果您的 Share Disk Storage 要採用 SCSI 的裝置，必須選擇有支援多主機通道（Multi-Host），其常見的架構如圖 3。兩個 single-initiator 的 SCSI 匯流排在一個單一控制卡 RAID 陣列的阻絕器，一個 Single-initiator SCSI 匯流排只允許一個成員連接到它，並且提供主機的分離與比一個 multi-initiator 匯流排更佳的效能表現。 Single-initiator 的匯流排確保每 一個成員都能免於由於其他成員的系統負載初始或修復所引起的干擾。

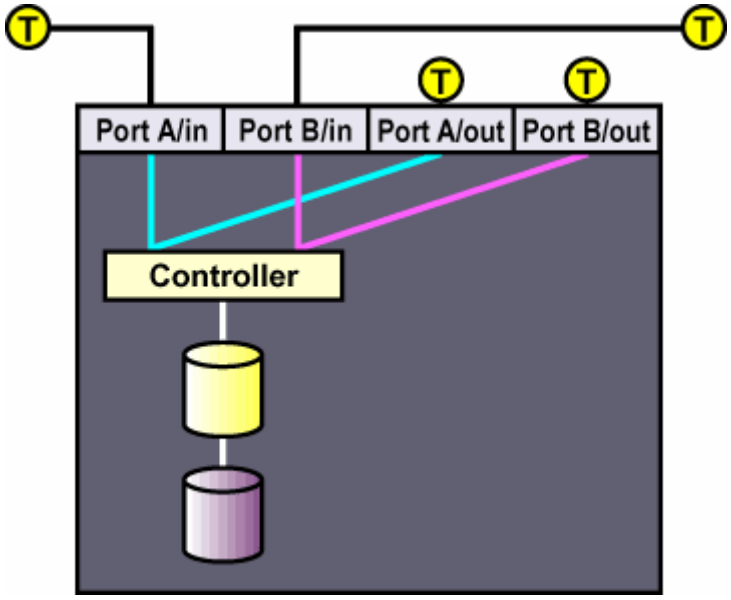

圖 3：連接到 single-initiator 的 SCSI 匯流排的單一控制卡 RAID 陣列示意圖

SSA (Serial Storage Architecture)

串列式儲存架構（ SSA ）是由 X3 授權標準委員會的 X3T10 技術委員會中的 X3T10.1 工作群組所正在發展的高效能串列式電腦與週邊介面。最初由 IBM 發展， 現今 SSA 是由 SSA 工業協會 推廣的開放技術。

SSA 基本上是一個執行於 SCSI-2 軟體協定的串列式技術。意思是 SSA 配接卡的裝置驅動程式應該可以很容易地整合到現有的 Linux SCSI 子系統。底線是，資料是透過以 200 MBit/s 全雙工傳輸的雙絞電纜來傳送，而這比 68 Pin 的平行 Wide SCSI 技術更易於處理。

SSA 和 SCSI 相較之下，有下列優點：

- 它更易於設定和接線 (它不需要終端電阻)。
- 它內建了 HA 特徵。SSA Loop 架構（相對於 SCSI 匯流排）沒有 SPOF（參考圖 4）。如果部份的 Loop 失效，裝置驅動程式將自動並透明地重新自我設定以確保所有的 SSA 裝置可被存取而沒有任何明顯的中斷。
- 它不是使用 SCSI ID 定址，意指設定配接卡毫無困難。
- 相對於 SCSI ， SSA 沒有使用匯流排裁決。而是使用類似網路的設計。資料以 128 位元組的封包被送出和接收，而且迴圈上的所有裝置可以獨立的請求 time slots 。反過來 SCSI 需要匯流排裁決，如果 initiator 沒有及時釋放匯流排可能導致效能死結。
- SSA 允許每兩個裝置間相距 25 公尺。再者，有允許資料穿過 50 微米的光纖電纜傳送達 2.4 公里的距離的光纖延伸器。如果設定適當的話，會使得它甚至合適於站台災難回復。
- 大部分的 SSA 配接卡支援兩個獨立的迴圈，使得連結鏡射的磁碟到不同迴圈以提高可用性成為可能。
- SSA 迴圈是對稱的，雙絞線，自由電位的。沒有 TERMPWR 電位移的問題。

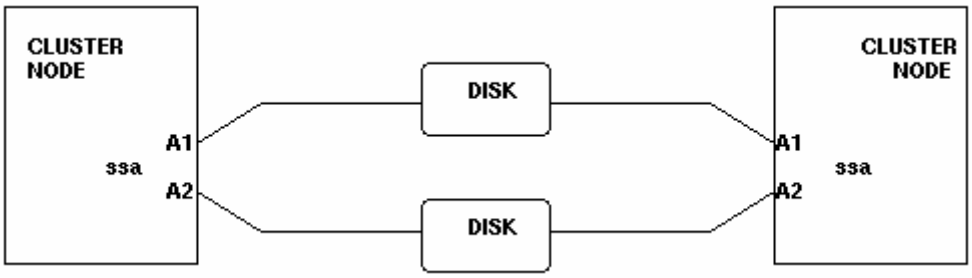

圖 4： SSA 示意圖

SSA 是一個比 SCSI 優良的技術，不過很可惜地，SSA 磁碟只能從 IBM 購買，取得成本太高，這些磁碟價格遠高一般的 SCSI 磁碟價格。而且至今在 Linux 上仍沒有夠成熟的 SSA Driver。

Fibre Channel Interconnect

Fibre 是筆者最喜歡的解決方案，也是筆者認為是比較有擴展性且適合大型企業

的架構。較簡單經濟的架構便如圖 5 所示，兩個主機連接埠的一個單一控制卡 RAID 陣列，沒有使用 Fibre Hub 或 Switch，主機連接埠配接卡直接連線至 RAID 控制卡。當您使用這種類型的 single-initiator 光纖通道連線，您的 RAID 控制器必須擁有分開的主機連接埠給每一部 Cluster 成員。

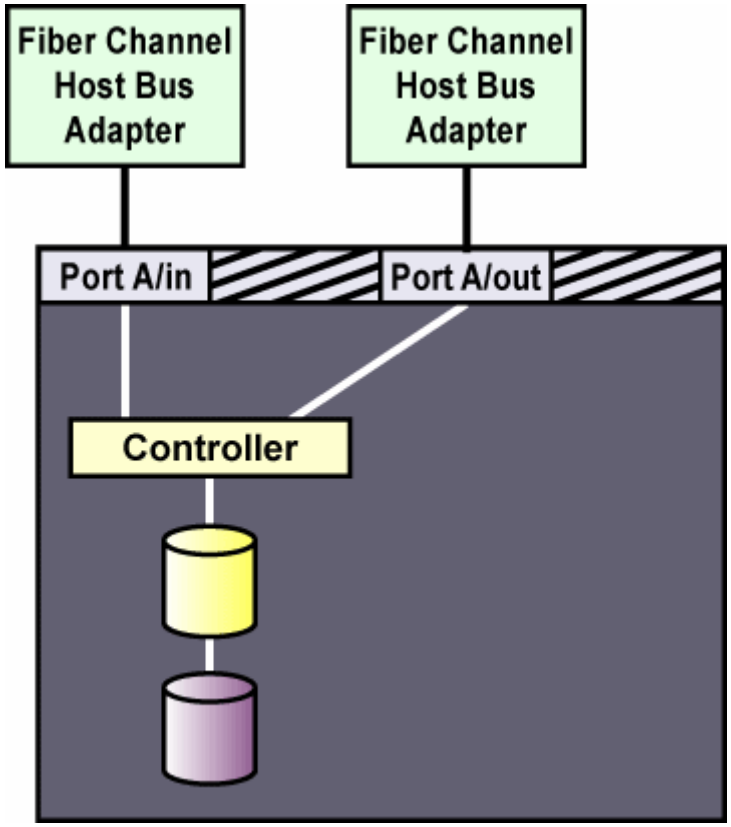

圖 5：連接到 single-initiator 的 SCSI 匯流排的單一控制卡 RAID 陣列示意圖

IP Address Takeover

最後筆者要介紹 IP Address Takeover 技術，通常客戶端機器和應用程式無法在運作時，從幾個 IP Address 中選擇一個 IP Address 來存取伺服器。所以主要的伺服器無法提供服務時，接管的節點必須在重新啟動應用程式之前取原來提供服務的 IP (well-known IP Address)。這個程序稱之為 IP Address Takeover （ IPAT ）。它的基本運作原理如下：

HA Clusters 的成員在有兩個網路介面，一個是 Service IP Interface、一個是 Standby IP Interface。如果它是具有較高優先權的節點或是主要節點 (Primary Node)，一旦這個節點取得資源群組，Service IP Interface 將會變成這個對外提供的服務 IP Address。
例如在 Linux HA 啟動前，Primary Node 與 Secondary Node 上的 IP Address

配置如下表：

表 2：Linux HA 啟動前的 IP 配置表

Interface	Node 1 (primary)	Node 2 (secondary)
Service	192.168.10.10	192.168.10.20
Standby	192.168.11.10	192.168.11.20

表 2 是 Linux-HA 被啟動之前的情形。 兩個服務介面都位於它們個別配置的啟動地址上。備援介面位於它們的備援地址上。假設整個 Clusters 對外服務的 IP Adress (也就是 Client 連至 Clusters 的 IP address)為"**192.168.10.11"**。當 Linux-HA 被啟動於兩個節點時，由於 Node 1 是主要節點，它將取得對外服務地址。

表 3：Linux HA 啟動後的 IP 配置表

Interface	Node 1 (primary)	Node 2 (secondary)
Service	**192.168.10.11**	192.168.10.20
Standby	192.168.11.10	192.168.11.20

假設主要節點的服務(Service) 網卡故障，則備援(Standby)配接卡將接管服務地址 (**192.168.10.11)**，如表 4。

表 4：主要節點的服務網卡故障後的 IP 配置表

Interface	Node 1 (primary)	Node 2 (secondary)
Service		192.168.10.20
Standby	**192.168.10.11**	192.168.11.20

假設主要節點 Node 1 故障，Node 2 將取得包括於資源群組中的服務地址，如表 5。

表 5：主要節點的故障後的 IP 配置表

Interface	Node 1 (primary)	Node 2 (secondary)
Service		192.168.10.20
Standby		**192.168.10.11**

由以上這個簡單的例子中，我們也可以移動 **192.168.10.11** 到節點 2 的服務配接卡，但是移動到備援配接卡較為適合。

首先，備援配接卡就不再需要了，也就是 Node 2 的備援配接卡也有 Client 連線，並提服務，而不是閒置在那兒做備援。

其次，節點 2 也可能正在執行一個資源群組（相互接管）。如果服務 IP Address 總是移動到存活節點的相同的（即備援）配接卡，那麼移動"對外服務 IP Address" 的邏輯就得較為簡單。

讀者看到 HA 利用四張網卡來提供一個對外服務的 IP，一定會覺得麻煩又眼花瞭亂。雖然我們也可以每個節點只使用一張配接卡於。但是有兩個原因不建議這樣做。

- 首先，如果一個節點的配接卡失效，無法判斷是只有配接卡或整個實體網路失效。
- 其次，如果我們加入邏輯來判斷是否是配接卡或是網路失效，我們可能必須立即執行節點的 failover，而這要花費比本地端配接卡 failover 還要更長的時間。

後記

筆者一直希望 RedHat 能推出一套可靠好用 Linux HA 解決方案，看到 Cluster Suit 的推出真是令人雀躍。這期文章筆者先介紹 HA 解決方案中常見的技術及觀念，下期我們再來捲起袖管打造全年無休的 RedHat High Availability Cluster。