

# SUSE Linux Enterprise High Availability Extension

11

[www.novell.com](http://www.novell.com)

2009年8月06日

High Availability 指南



## **High Availability 指南**

版權所有 © 2006-2009 Novell, Inc.

根據自由軟體基金會 (Free Software Foundation) 所發佈的 GNU 自由文件授權 (GNU Free Documentation License) 1.2 版或更新版本，使用者可以複製、散佈與/或修改本文件；「恆常章節」為此著作權聲明與授權。本節中包含授權「GNU 自由文件授權」的一份副本。

SUSE®、openSUSE®、openSUSE® 標幟、Novell®、Novell® 標幟、N® 標幟是 Novell, Inc. 在美國和其他國家/地區的註冊商標。Linux\* 是 Linus Torvalds 的註冊商標。所有其他協力廠商商標，為各所有人所有之財產。商標符號 (®、™ 等)代表 Novell 的商標；星號 (\*) 代表協力廠商的商標。

本手冊中所有資訊在編輯時，都已全力注意各項細節。但這不保證百分之百的正確性。因此，Novell, Inc.、SUSE LINUX Products GMBH、作者或譯者都不需對任何錯誤或造成的結果負責。

# 目錄

關於本指南	vii
<b>I 安裝與設定</b>	<b>1</b>
<b>1 概念綜覽</b>	<b>3</b>
1.1 產品功能	3
1.2 產品利益	5
1.3 叢集組態	8
1.4 結構	11
1.5 最新功能?	14
<b>2 開始使用</b>	<b>17</b>
2.1 硬體要求	17
2.2 軟體需求	18
2.3 共享磁碟系統需求	18
2.4 準備	18
2.5 綜覽：安裝及設定叢集	19
<b>3 使用 YaST 進行安裝及基本設定</b>	<b>21</b>
3.1 安裝 High Availability Extension	21
3.2 初始叢集設定	22
3.3 連線叢集	24

<b>II</b>	<b>組態與管理</b>	<b>27</b>
<b>4</b>	<b>使用 GUI 設定叢集資源</b>	<b>29</b>
4.1	Linux HA Management Client . . . . .	29
4.2	建立叢集資源 . . . . .	31
4.3	建立 STONITH 資源 . . . . .	35
4.4	設定資源限制 . . . . .	36
4.5	指定資源容錯移轉節點 . . . . .	41
4.6	指定資源錯誤回復節點 (資源相粘性) . . . . .	42
4.7	設定資源監控 . . . . .	43
4.8	啟動新的叢集資源 . . . . .	45
4.9	移除叢集資源 . . . . .	45
4.10	設定叢集資源群組 . . . . .	46
4.11	設定複製品資源 . . . . .	51
4.12	移轉叢集資源 . . . . .	52
4.13	如需更多資訊 . . . . .	53
<b>5</b>	<b>從指令行設定叢集資源</b>	<b>55</b>
5.1	指令行工具 . . . . .	55
5.2	組態變更除錯 . . . . .	55
5.3	建立叢集資源 . . . . .	56
5.4	建立 STONITH 資源 . . . . .	60
5.5	設定資源限制 . . . . .	61
5.6	指定資源容錯移轉節點 . . . . .	63
5.7	指定資源錯誤回復節點 (資源相粘性) . . . . .	64
5.8	設定資源監控 . . . . .	64
5.9	啟動新的叢集資源 . . . . .	64
5.10	移除叢集資源 . . . . .	65
5.11	設定叢集資源群組 . . . . .	65
5.12	設定複製品資源 . . . . .	66
5.13	移轉叢集資源 . . . . .	67
5.14	使用陰影組態進行測試 . . . . .	67
5.15	如需更多資訊 . . . . .	68
<b>6</b>	<b>設定簡易測試資源</b>	<b>69</b>
6.1	使用 GUI 設定資源 . . . . .	69
6.2	資源的手動組態設定 . . . . .	71
<b>7</b>	<b>新增或修改資源代辦</b>	<b>73</b>
7.1	STONITH 代辦 . . . . .	73
7.2	撰寫 OCF 資源代辦 . . . . .	74

<b>8</b>	<b>圍籬區隔與 STONITH</b>	<b>75</b>
8.1	圍籬區隔的類別	75
8.2	節點層級圍籬區隔	76
8.3	STONITH 組態	78
8.4	監控圍籬區隔設備	82
8.5	特殊圍籬區隔設備	82
8.6	如需更多資訊	83
<b>9</b>	<b>Linux Virtual Server 的負載平衡</b>	<b>85</b>
9.1	概念綜覽	85
9.2	High Availability	86
9.3	如需更多資訊	87
<b>10</b>	<b>網路設備 Bonding</b>	<b>89</b>
<b>11</b>	<b>將叢集更新至 SUSE Linux Enterprise 11</b>	<b>93</b>
11.1	準備與備份	94
11.2	更新/安裝	95
11.3	資料轉換	95
11.4	如需更多資訊	97
<b>III</b>	<b>儲存與資料複製</b>	<b>99</b>
<b>12</b>	<b>Oracle Cluster File System 2</b>	<b>101</b>
12.1	特點及優勢	101
12.2	管理公用程式與指令	102
12.3	OCFS2 套件	103
12.4	建立 OCFS2 卷冊	103
12.5	裝載 OCFS2 卷冊	107
12.6	其他資訊	109
<b>13</b>	<b>叢集 LVM</b>	<b>111</b>
13.1	cLVM 的組態	111
13.2	明確設定適合的 LVM2 設備	113
13.3	如需更多資訊	114
<b>14</b>	<b>分散式複製區塊設備 (DRBD)</b>	<b>115</b>
14.1	安裝 DRBD 服務	116

14.2	設定 DRBD 服務 . . . . .	116
14.3	測試 DRBD 服務 . . . . .	118
14.4	DRBD 疑難排解 . . . . .	120
14.5	其他資訊 . . . . .	122
<b>IV</b>	<b>疑難排解與參考</b>	<b>123</b>
<b>15</b>	<b>疑難排解</b>	<b>125</b>
15.1	安裝問題 . . . . .	125
15.2	HA 叢集除錯 . . . . .	126
15.3	常見問題集 . . . . .	127
15.4	獲取詳細資訊 . . . . .	129
<b>16</b>	<b>叢集管理工具</b>	<b>131</b>
<b>17</b>	<b>叢集資源</b>	<b>183</b>
17.1	受支援的資源代辦類別 . . . . .	183
17.2	OCF 傳回代碼 . . . . .	184
17.3	資源選項 . . . . .	186
17.4	資源作業 . . . . .	187
17.5	例項屬性 . . . . .	188
<b>18</b>	<b>HA OCF 代辦</b>	<b>191</b>
<b>V</b>	<b>附錄</b>	<b>263</b>
<b>A</b>	<b>GNU 授權</b>	<b>265</b>
A.1	GNU General Public License . . . . .	265
A.2	GNU Free Documentation License . . . . .	268
	<b>術語</b>	<b>273</b>

# 關於本指南

SUSE® Linux Enterprise High Availability Extension 是一個採用開放原始碼叢集技術的整合式套裝軟體，它可讓您實作高可用性的實體和虛擬 Linux 叢集。為進行快速有效的組態設定和管理，High Availability Extension 中提供有圖形使用者介面 (GUI) 和指令行介面 (CLI)。

本指南適用於需要安裝、設定及維護 High Availability (HA) 叢集的管理員。文中對兩種方法 (GUI 與 CLI) 均有詳細說明，以幫助管理員選擇滿足其執行主要任務需求的適當工具。

本指南分為以下幾個部分：

## 安裝與設定

在開始安裝及設定叢集之前，請先熟悉叢集基礎知識與結構，取得主要功能與優點的綜覽，以及自上一版本以來的變更。瞭解須滿足的硬體與軟體要求，及在執行下一步操作之前的準備工作。使用 YaST 執行 HA 叢集的安裝與基本設定。

## 組態與管理

使用 GUI 或 `crm` 指令行介面新增、設定並管理資源。瞭解如何利用負載平衡與圍籬區隔如果您要撰寫自己的資源代辦或修改現存資源代辦，請取得有關如何建立不同類型資源代辦的背景資訊。

## 儲存與資料複製

SUSE Linux Enterprise High Availability Extension 內建有支援叢集檔案系統 (Oracle 叢集檔案系統, OCFS2) 與磁碟區管理員 (叢集化邏輯磁碟區管理員, cLVM)。為了複製資料，High Availability Extension 還提供了 DRBD (分散式複製區塊設備)，使用它您可以將高可用性服務的資料從叢集的使用中節點鏡像複製到待機節點。

## 疑難排解與參考

若要管理自己的叢集，您需要執行一定的疑難排解作業。瞭解最常見的問題及其修復方法。請尋找 High Availability Extension 提供之指令行工具的完整參考，管理自己的叢集。還列出了有關叢集資源及資源代辦之最重要的資料與統計數字。

本手冊的許多章節包含連到其他文件資源的連結。這包括系統和網際網路上所提供的其他文件。

如需適用於產品的文件與最新文件更新的綜覽，請參閱<http://www.novell.com/documentation>。

# 1 意見反應

以下為可供使用的數種意見回應管道：

- 若要報告產品元件的錯誤，或提交增強功能要求，請使用 <https://bugzilla.novell.com/>。如果您不熟悉 Bugzilla，Novell Bugzilla 首頁中的 *Bug Writing FAQs* 或許會有幫助。
- 我們希望得到您對本手冊以及本產品隨附之其他文件的意見和建議。請使用線上文件每頁下方的使用備註功能，並在其中輸入您的意見。

# 2 文件慣例

本手冊使用下列印刷慣例：

- `/etc/passwd`：目錄名稱與檔名
- `placeholder`：以實際的值來取代 `placeholder`
- `PATH`：環境變數 `PATH`
- `ls, --help`：指令、選項和參數
- `user`：使用者或群組
- `Alt, Alt+F1`：供人按下的按鍵或案件組合；顯示的按鍵與鍵盤上一樣為大寫
- 「檔案」，「檔案」 > 「另存新檔」：功能表項目、按鈕
- 本段僅與指定的結構有關。箭頭標示了文字區塊的開頭與結尾。  
本段僅與指定的架構有關。箭頭標示了文字區塊的開頭與結尾。
- *Dancing Penguins* (章節 *Penguins*, ↑其他手冊)：這是對其他手冊某章節的參考。

# I. 安裝與設定



# 概念綜覽

SUSE® Linux Enterprise High Availability Extension 是一個採用開放原始碼叢集技術的整合式套裝軟體，它可讓您實作高可用性的實體和虛擬 Linux 叢集，並可避免單一故障點。該套裝軟體可確保資料、應用程式和服務等重要網路資源的高可用性及其可管理性。因此，可協助您保持業務持續運作，保護資料完整性，並可降低關鍵任務 Linux 工作負載的意外停機時間。

它內建有基本的監控、訊息傳送和叢集資源管理功能，支援個別受管理叢集資源的容錯移轉、錯誤回復和移轉 (負載平衡)。High Availability Extension 做為 SUSE Linux Enterprise Server 11 的附加產品提供。

## 1.1 產品功能

SUSE® Linux Enterprise High Availability Extension 可協助您確保和管理網路資源的可用性。以下清單反白顯示部分主要功能：

### 支援眾多叢集案例

包括主動/主動和主動/被動 (N+1、N+M、N 到 1、N 到 M) 案例以及混合式實體和虛擬叢集 (允許將虛擬伺服器與實體伺服器叢集在一起，以提高服務可用性與資源使用率)。

多節點主動叢集，最多可包含 16 台 Linux 伺服器。可以使用叢集內的任一伺服器重新啟動同一叢集中之失敗伺服器中的資源 (應用程式、服務、IP 位址和檔案系統)。

## 彈性解決方案

High Availability Extension 內建 OpenAIS 訊息傳送與成員層及 Pacemaker 叢集資源管理員。使用 Pacemaker，管理員可以持續監控其資源的狀態、管理相依性，並能根據高度可組態規則 (Rule 和 Police) 自動停止和啟動服務。High Availability Extension 可讓您對叢集進行特定應用程式和硬體架構的調整以符合貴組織的需求。時間相關組態可讓服務在指定時間自動移轉回已修復節點。

## 儲存與資料複製

使用 High Availability Extension，您可以視需要動態指定和重新指定伺服器儲存。它支援光纖通道或 iSCSI 儲存區域網路 (SAN)。還支援共享磁碟系統，但這些系統並非必要系統。SUSE Linux Enterprise High Availability Extension 還隨附了支援叢集檔案系統 (Oracle 叢集檔案系統, OCFS2) 和磁碟區管理員 (叢集化邏輯磁碟區管理員, cLVM)。為了複製資料，High Availability Extension 還提供了 DRBD (分散式複製區塊設備)，使用它您可以將高可用性服務的資料從叢集的使用中節點鏡像複製到待機節點。

## 支援虛擬化環境

SUSE Linux Enterprise High Availability Extension 支援包含實體與虛擬 Linux 伺服器的混合叢集。SUSE Linux Enterprise Server 11 內建有 Xen (一個開放原始碼虛擬化監管程式)。High Availability Extension 中的叢集資源管理員可辨識、監控和管理在使用 Xen 建立之虛擬伺服器中執行的服務，以及在實體伺服器中執行的服務。叢集可以將訪客系統做為服務進行管理。

## 資源代辦

SUSE Linux Enterprise High Availability Extension 含有大量資源代辦用以管理 Apache、IPv4、IPv6 等眾多資源。它還內建有適用於 IBM WebSphere Application Server 等廣受歡迎的協力廠商應用程式的資源代辦。如需產品隨附的開放叢集架構 (OCF) 資源代辦的清單，請參閱第 18 章「HA OCF 代辦」[191 頁]。最新的清單可從 [www.novell.com/products/highavailability](http://www.novell.com/products/highavailability) 線上取得。

## 簡單易用的管理

為了能輕鬆地進行組態設定與管理，High Availability Extension 內建有圖形使用者介面 (如 YaST 與 Linux HA Management Client) 和功能強大的整合式指令行介面。這兩種方法皆提供有單一管理點，可有效地監控與管理叢集。後面的章節中將介紹如何執行此作業。

## 1.2 產品利益

High Availability Extension 可讓您最多將 16 台 Linux 伺服器設定成一個 High Availability 叢集 (HA 叢集)，其中的資源可以動態切換或移動至叢集中的任一伺服器。資源可以設成在遇到伺服器故障時自動進行移轉，也可以選擇手動移動以對硬體進行疑難排解或平衡工作負載。

High Availability Extension 利用商用元件提供高可用性。透過將應用程式和作業整合到一個叢集可降低成本。High Availability Extension 還可讓您集中管理整個叢集並調整資源以滿足不斷變化的工作負載要求 (從而手動使叢集達到「負載平衡」)。允許具有兩個以上節點的叢集還可透過允許幾個節點共享一個「熱備用」來節省成本。

它還具有另一個同等重要的優點，就是可以潛在地縮短計畫外服務的中斷運作時間，以及為了執行軟體和硬體的維護與升級所需的計劃內中斷運作時間。

您希望實作叢集的理由包括：

- 增加可用性
- 改善效能
- 降低作業成本
- 可調適性
- 災害復原
- 資料保護
- 伺服器整合
- 儲存整合

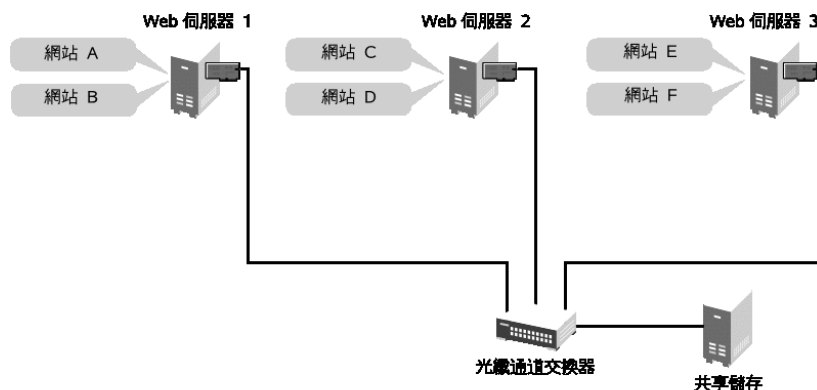
藉由在共享磁碟子系統上建置 RAID，可以達到磁碟容錯共享。

以下案例說明 High Availability Extension 具備的一些優點。

## 範例叢集案例

假設您已經設定一個含有三台伺服器的叢集，而這三台伺服器上都已安裝了 Web 伺服器。叢集中的每個伺服器都代管兩個網站。每個網站上的所有資料、圖形以及網頁內容都儲存在與叢集中的每一個伺服器相連接的共享磁碟子系統上。以下的圖解可以描繪這個設定的可能外觀。

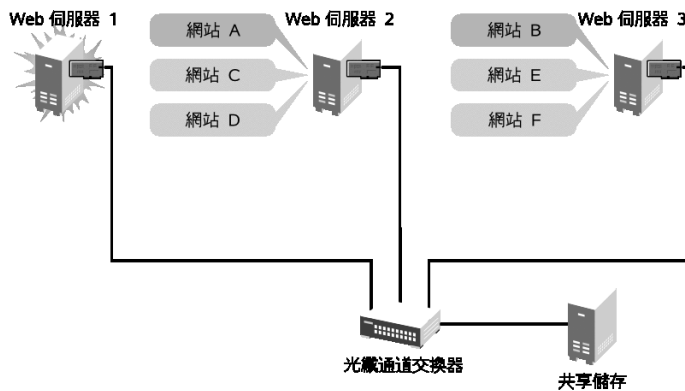
圖形 1.1 由三台伺服器所構成的叢集



正常的叢集作業期間，每一伺服器都會和叢集中的其他伺服器保持通訊，並且定期檢查所有已登錄的資源以偵測是否有故障發生。

假設 Web 伺服器 1 發生硬體或軟體問題，且利用 Web 伺服器 1 進行網際網路存取、收發電子郵件和資訊的使用者失去連線。下圖說明當 Web 伺服器 1 故障時，資源移動的情形。

**圖形 1.2** 一台伺服器故障後，由三台伺服器所構成的叢集



網站 A 將移至 Web 伺服器 2，而網站 B 將移至 Web 伺服器 3。IP 位址和證書也會移至 Web 伺服器 2 和 Web 伺服器 3。

當您進行叢集設定時，可以決定發生故障時每一台 Web 伺服器上代管之網站的移動目的地。在前面的範例中，將網站 A 設定為移至 Web 伺服器 2，而將網站 B 設定為移至 Web 伺服器 3。這樣，之前由 Web 伺服器 1 處理的工作負載仍可用，並且會在所有正常運行的叢集成員之間平均分散。

Web 伺服器 1 失敗時，High Availability Extension 軟體將

- 偵測故障並向 STONITH 確認 Web 伺服器 1 確實已停止運行
- 將先前裝載於 Web 伺服器 1 上的共享資料目錄重新裝載於 Web 伺服器 2 和 Web 伺服器 3。
- 在 Web 伺服器 2 和 Web 伺服器 3 上重新啟動先前於 Web 伺服器 1 上執行的應用程式
- 將 IP 位址傳送至 Web 伺服器 2 和 Web 伺服器 3

在此範例中，容錯移轉程序會快速完成，而使用者也將在數秒內重新恢復存取網站資訊。大部份的情況下，無需再次登入。

現在假設 Web 伺服器 1 所發生的問題已經解決，並且它已恢復到正常的作業狀態。此時，網站 A 和網站 B 可以自動錯誤回復 (移回) 到 Web 伺服器 1，也可以保留在現有伺服器上。這取決於您為它們設定資源的方式。將服務移轉回 Web

伺服器 1 會導致一定的停機時間，因此，High Availability Extension 也可讓您將移轉延遲一段時間，等到不會使服務運行中斷或只會讓服務短時間中斷時再進行移轉。兩種備選方法各有優缺點。

High Availability Extension 還提供了資源移轉功能。您可以根據系統管理需要，將應用程式、網站等移轉至叢集中的其他伺服器。

例如，您可以手動將網站 A 或網站 B 從 Web 伺服器 1 移至叢集中的任一其他伺服器。您也許希望藉此對 Web Server 1 進行升級或執行排程維護的工作，或僅為了增加網站效能或連線能力。

## 1.3 叢集組態

具有 High Availability Extension 的叢集組態可能包含，也可能不包含共享磁碟子系統。共享磁碟子系統可以透過高速光纖通道卡、纜線和交換器進行連接，也可設定為使用 iSCSI。如果有一台伺服器失敗，叢集中的另一台指定伺服器就會自動裝載先前裝載於失敗伺服器上的共享磁碟目錄。這使得網路使用者得以繼續存取共享磁碟子系統上的目錄。

---

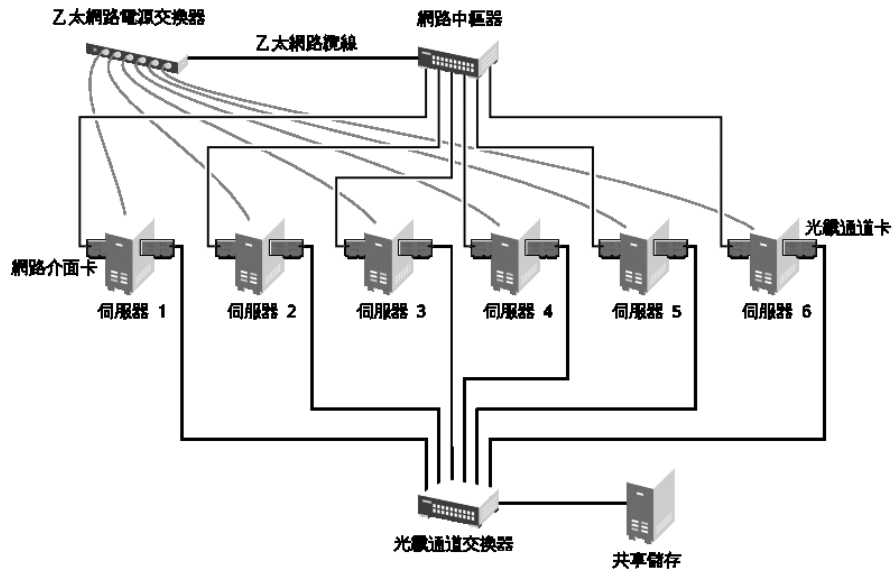
### 重要：具有 cLVM 的共享磁碟子系統

使用具有 cLVM 的共享磁碟子系統時，必須將該子系統連接至叢集中需要從中存取它的所有伺服器。

---

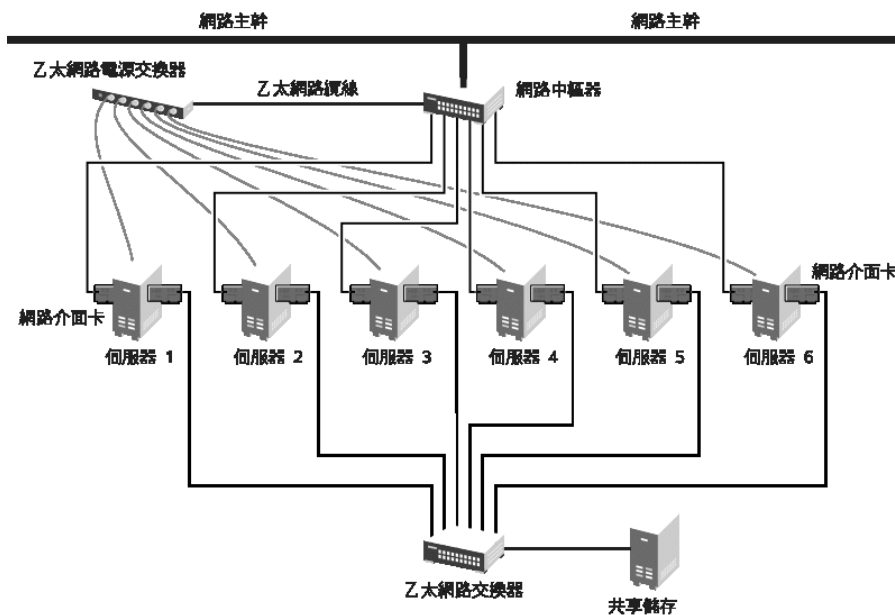
一般資源包括資料、應用程式和服務。下圖顯示一般光纖通道叢集組態的可能外觀。

圖形 1.3 一般光纖通道叢集組態



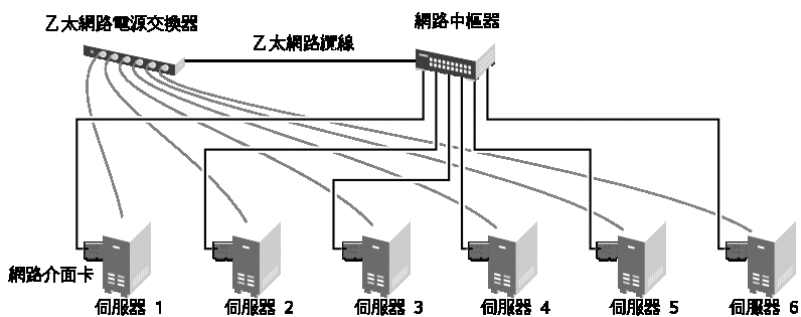
雖然光纖通道提供的效能最佳，但您仍可以將叢集設定為使用 iSCSI。iSCSI 是光纖通道的替代方案，可用於建立低成本的儲存區域網路 (SAN)。下圖顯示一般 iSCSI 叢集組態的可能外觀。

圖形 1.4 一般 iSCSI 叢集組態



雖然大部分叢集都包含共享磁碟子系統，但也可以建立不含共享磁碟子系統的叢集。下圖顯示不含共享磁碟子系統之叢集的可能外觀。

圖形 1.5 不含共享儲存的一般叢集組態



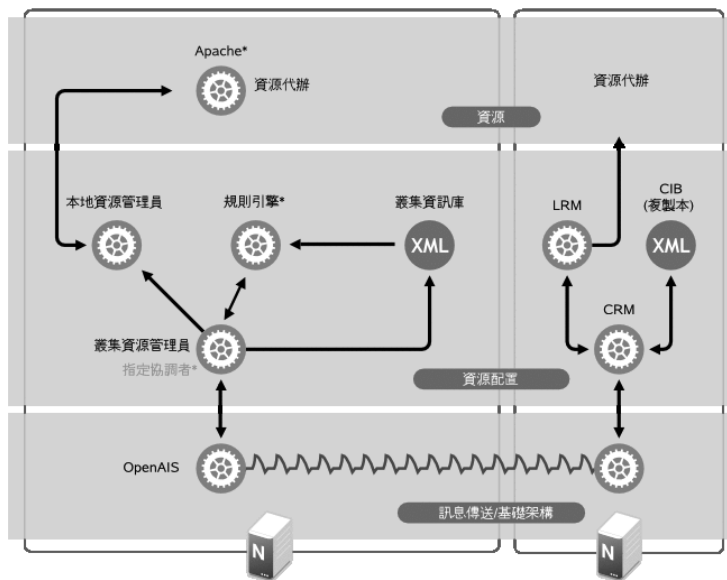
## 1.4 結構

本節提供 High Availability Extension 結構的簡要綜覽。它識別並提供有關結構元件的資訊，並說明這些元件如何相互操作。

### 1.4.1 結構層

High Availability Extension 採用分層結構。圖形 1.6 「結構」 [11頁] 說明不同的層及其相關的元件。

圖形 1.6 結構



## 訊息傳送與基礎架構層

主要層或第一層為訊息傳送/基礎架構層，也稱為 OpenAIS 層。此層包含送出含有「I'm alive」(我存在) 訊號及其他資訊之訊息的元件。High Availability Extension 的程式存放在訊息傳送/基礎架構層。

## 資源配置層

下一層為資源配置層。此層最為複雜，由下列元件組成：

### 叢集資源管理員 (CRM)

在資源配置層中執行的每個動作均透過叢集資源管理員進行傳遞。如果資源配置層的其他元件 (或更高一層中的元件) 需要進行通訊，通訊將透過本地 CRM 完成。

在每個節點上，CRM 維護叢集資訊庫 (CIB) [12頁]，它包含所有叢集選項、節點、資源、它們的關係和目前狀態的定義。將會選出叢集中的一個 CRM 做為指定協調者 (DC)，這意味著它擁有主要 CIB。叢集中的所有其他 CIB 則擁有主要 CIB 的複製本。CIB 上的一般讀寫作業透過主要 CIB 序列化。DC 是叢集中可以決定是否需要執行全叢集變更 (例如圍籬區隔節點或移動資源) 的唯一實體。

### 叢集資訊庫 (CIB)

叢集資訊庫是整個叢集組態和目前狀態的記憶體內部 XML 表示。它包含所有叢集選項、節點、資源、限制及相互關係的定義。CIB 還可同步更新至所有叢集節點。叢集中有一個主要 CIB，由 DC 負責維護。其他所有節點均包含一個 CIB 複製本。

### 規則引擎 (PE)

指定協調者需要進行全叢集變更 (對新的 CIB 做出反應) 時，規則引擎會根據目前狀態和組態計算出叢集的下一個狀態。PE 還可產生轉換圖表，該圖表包含 (資源) 動作和相依性清單，用以取得下一個叢集狀態。PE 將在每個節點上執行，以提高 DC 容錯移轉速度。

### 本地資源管理員 (LRM)

LRM 代表 CRM 呼叫本地資源代辦 (請參閱章節「資源層」 [13頁])。因此，它可以執行啟動/停止/監控作業，並將結果報告給 CRM。它還可以隱藏資源代辦的受支援程序檔標準 (OCF、LSB、Heartbeat 版本 1) 之間的差異。LRM 是其本地節點上之所有資源相關資訊的管理來源。

## 資源層

最高層為資源層。資源層包含一或多個資源代辦 (RA)。資源代辦是一些程式，通常是外圍程序檔，撰寫這些程式是為了啟動、停止和監控特定種類的服務(資源)。資源代辦僅可由 LRM 呼叫。協力廠商可以將他們自己的代辦包含在檔案系統中的已定義位置，從而為他們自己的軟體提供即裝即用的叢集整合功能。

### 1.4.2 程序流程

SUSE Linux Enterprise High Availability Extension 使用 Pacemaker 做為 CRM。CRM 被當做精靈 (crmd) 來實作，即在每個叢集節點上都有一個例項。Pacemaker 集中了所有透過選出一個 crmd 例項做為主要例項做出的叢集決策。如果所選的 crmd 程序 (或它所在的節點) 失敗，則會建立一個新的 crmd 程序。

可反映出叢集組態及叢集中所有資源的目前狀態的 CIB 會保留在每個節點上。CIB 的內容將在整個叢集中自動保持同步。

在叢集中執行的許多動作都會導致全叢集發生變更。這些動作包括新增或移除叢集資源，或變更資源限制等。執行此類動作時，必須瞭解叢集中會發生什麼狀況。

例如，假設您要新增叢集 IP 位址資源。為此，您可以使用指令行工具或 GUI 其中之一來修改 CIB。不必在 DC 上執行動作，您可以使用叢集中的任何節點上的其中一種工具，動作即會轉送至 DC。然後，DC 會將 CIB 變更複製到所有叢集節點。

接著，PE 將根據 CIB 中的資訊計算出叢集的理想狀態，以及達到該狀態的方式，並會將一系列指示饋送至 DC。DC 透過訊息傳送/基礎架構層送出指令，而這些指令將由其他節點上的 crmd 對等接收。每個 crmd 皆使用其 LRM (當做 lrmd 實作) 執行資源修改。lrmd 為非支援叢集，它可直接與資源代辦 (程序檔) 互動。

對等節點都會將其作業結果回報給 DC。一旦 DC 確定已在叢集中成功執行了所有必要的作業，叢集就會轉回閒置狀態，等待接下來的事件。如果有任何作業未按計畫執行，則會用 CIB 中記錄的新資訊再次呼叫 PE。

在某些情況下，可能需要關閉節點以保護共享資料或完成資源復原。為執行此操作，Pacemaker 提供了圍籬區隔子系統 stonithd。STONITH 是「Shoot The Other Node In The Head」的縮寫，通常使用遠端電源交換器實作。在 Pacemaker 中，

將 STONITH 設備模式化為資源 (並在 CIB 中進行設定)，以便可以輕鬆地對它們進行故障監控。不過，stonithd 負責瞭解 STONITH 拓樸，因此，其用戶端只需要圍籬區隔一個節點，stonithd 將完成剩餘的操作。

## 1.5 最新功能?

借助 SUSE Linux Enterprise Server 11，叢集堆疊已從 Heartbeat 變更為 OpenAIS。OpenAIS 實作工業標準 API，即 Service Availability Forum (服務可用性論壇) 所發佈的應用程式介面規範 (Application Interface Specification, AIS)。SUSE Linux Enterprise Server 10 中的叢集資源管理員被保留下來，但功能獲得顯著提升，且已移植到 OpenAIS，現在稱為 Pacemaker。

如需從 SUSE® Linux Enterprise Server 10 SP2 到 SUSE Linux Enterprise High Availability Extension 11 以來，High Availability 元件發生了哪些變更的詳細資料，請參閱後面各節。

### 1.5.1 新特性和新功能

#### 移轉限定值與故障逾時

現在，High Availability Extension 提出了移轉限定值與故障逾時的概念。您可定義大量資源故障，發生這些故障後會將資源移轉至新節點。預設情況下，在管理員手動重設資源的故障計數前，將不再允許節點執行失敗資源。不過，還可以透過設定資源的 `failure-timeout` 選項，讓資源過期。

#### 資源和作業預設值

現在，您可為資源選項和作業設定全域預設值。

#### 支援離線組態變更

通常，在以原子方式更新組態前，需要先預覽一系列變更的效果。現在，您可以建立組態的「陰影」副本，在提交並以原子方式變更使用中的叢集組態之前，可以使用指令行介面編輯該副本。

#### 重複使用作業的規則、選項和設定

只需定義作業的規則、`instance_attributes`、`meta_attributes` 和設定一次，便可在多個地方進行參考。

將 XPath 運算式用於 CIB 中的特定作業

現在，CIB 接受 XPath 式 create、modify 和 delete 作業。如需詳細資訊，請參閱 cibadmin 說明文字。

多維並存和順序限制

若要建立一組並存資源，以前可以定義一個資源群組 (但它並不總能準確反映出設計目地)，或將每個關係定義為個別限制，但隨著資源和組合數目的增長，此方法會導致限制爆炸。現在，您還可以透過定義 resource\_sets 使用並存限制的替代形式。

從非叢集機器連接至 CIB

只要機器上安裝了 Pacemaker，即使機器本身不屬於叢集，也可能會連接該叢集。

在已知時間觸發週期性動作

預設會相對於資源啟動時間排定週期性動作，但並非所有情況都適合如此操作。若要指定作業應相對於的日期/時間，請設定作業的 interval-origin (間隔起始點)。叢集使用此時間點計算正確的 start-delay (啟動延遲時間)，即作業將在「起時點 + (間隔 \* N)」時發生。

## 1.5.2 變更的特性與功能

資源和叢集選項的命名慣例

現在，所有資源和叢集選項皆使用破折號 (-) 取代底線 (\_)。例如，master\_max 中繼選項已重新命名為 master-max。

重新命名 master\_slave 資源

master\_slave 資源已重新命名為 master。主要資源是一種特殊類型的複製資源，可以兩種模式之一進行操作。

屬性的容器標籤

已移除 attributes 容器標籤。

先決條件的作業欄位

pre-req 作業欄位已重新命名為 requires。

作業間隔

所有作業都必須設定有間隔。對於啟動/停止動作，必須將間隔設定為 0 (零)。

並存和順序限制的屬性

為了清楚起見，重新命名了並存和順序限制的屬性。

因發生故障而進行移轉的叢集選項

`resource-failure-stickiness` 叢集選項已由 `migration-threshold` 叢集選項取代。並請參閱移轉限定值與故障逾時 [14頁]。

指令行工具的引數

指令行工具的引數已經過一致化處理。並請參閱資源和叢集選項的命名慣例 [15頁]。

驗證和剖析 XML

叢集組態以 XML 格式撰寫。現在已使用功能更為強大的 RELAX NG 綱要取代了「文件類型定義」(DTD)，用來定義結構和內容的模式。`libxml2` 用做剖析程式。

id 欄位

id 欄位現在為 XML ID，它們有以下限制：

- ID 不能含有冒號。
- ID 不能以數字開頭。
- ID 必須是全域唯一的 (而不僅僅對於標籤唯一)。

參考其他物件

有些欄位 (例如參考資源之限制中的那些欄位) 是 IDREF。這表示它們必須參考現有資源或物件才能使組態生效。因此，移除在其他地方參考之物件的動作將會失敗。

## 1.5.3 已移除的特性與功能

設定資源中繼選項

資源中繼選項將不再能夠設定為頂層屬性。請轉為使用中繼屬性。

設定全域預設值

資源和作業預設值將不再從 `crm_config` 讀取。

# 2

## 開始使用

在下文中，您將瞭解到有關系統要求的資訊以及在安裝 High Availability Extension 之前需要進行的準備工作，並可檢視安裝和設定叢集的基本步驟的簡要綜覽。

### 2.1 硬體要求

以下清單指定以 SUSE® Linux Enterprise High Availability Extension 為基礎之叢集的硬體要求。這些要求代表最低硬體組態。依您打算如何使用叢集而定，您可能還需要其他硬體。

- 1 至 16 部安裝了第 2.2 節「軟體需求」[18頁]中所指定之軟體的 Linux 伺服器。伺服器不需要配備完全一樣的硬體 (記憶體、磁碟空間等)。
- 至少兩個 TCP/IP 通訊媒體。叢集節點使用多路廣播進行通訊，因此，網路設備必須支援多路廣播。通訊媒體應支援 100 Mbit/s 或更高的資料速率。最好結合乙太網路通道。
- 選擇性：連接至叢集 (需要從該叢集存取磁碟子系統) 內所有伺服器的共享磁碟子系統。
- STONITH 機制。STONITH 是「Shoot the other node in the head」的縮略字。STONITH 設備是一種電源交換器，叢集可使用它重設被視為已停止動作或行為錯誤的節點。重設無活動訊號的節點是確保掛起節點或只是呈現為已停止動作的節點不會損毀資料的唯一可靠方法。

若需更多資訊，請參閱第 8 章「圍籬區隔與 STONITH」[75頁]。

## 2.2 軟體需求

確定下列軟體需求都已符合：

- 在將要屬於叢集的所有節點上安裝了 SUSE® Linux Enterprise Server 11 及所有可用的線上更新。
- 在將要屬於叢集的所有節點上安裝了 SUSE Linux Enterprise High Availability Extension 11，包括所有可用的線上更新。

## 2.3 共享磁碟系統需求

如果您希望資料具有高度可用性，建議您使用共享磁碟系統(儲存區域網路，即 SAN)。如果使用共享磁碟子系統，請注意下列事項：

- 已根據製造商的指示正確設定共用磁碟系統，而且可以正常運作。
- 共享磁碟系統中包含的磁碟應設定為使用鏡像複製或 RAID，以增強共享磁碟系統的容錯能力。建議使用硬體型 RAID。系統並不是對所有組態都支援主機型軟體 RAID。
- 如果使用 iSCSI 存取共用磁碟系統，必須確定已正確設定 iSCSI 啟動程式和目標。
- 當您使用 DRBD 來實作在兩部機器之間配送資料的鏡像複製 RAID 系統時，請確認僅存取複製的設備。請與叢集的其餘部分使用相同(繫結)的 NIC，以利用該處所提供的備援。

## 2.4 準備

在安裝之前，請執行下列準備步驟：

- 透過在叢集中的每部伺服器上編輯 `/etc/hosts` 檔案，設定主機名稱解析並使用靜態主機資訊。如需詳細資訊，請參閱《*SUSE Linux Enterprise Server 管理指南*》之「基本網路」一章中的「設定主機名稱和 DNS」一節，該文件可從 <http://www.novell.com/documentation> 取得。

叢集的成員必須能夠藉由名稱相互找到對方，這是一項基本要求。如果名稱不可用，內部叢集通訊將會失敗。

- 透過將叢集節點與叢集外部的時間伺服器同步，設定時間同步。如需詳細資訊，請參閱《*SUSE Linux Enterprise Server 管理指南*》中的「與 NTP 同步時間」一章，該文件可從 <http://www.novell.com/documentation> 取得。

叢集節點將使用時間伺服器做為其時間同步來源。

## 2.5 綜覽：安裝及設定叢集

完成準備工作後，需要執行下列步驟來安裝並設定使用 SUSE® Linux Enterprise High Availability Extension 的叢集：

1. 在 SUSE Linux Enterprise Server 的基礎上將 SUSE® Linux Enterprise Server 11 及 SUSE® Linux Enterprise High Availability Extension 11 做為附加產品進行安裝。如需詳細資訊，請參閱第 3.1 節「安裝 High Availability Extension」[21頁]。
2. 設定 OpenAIS。如需詳細資訊，請參閱第 3.2 節「初始叢集設定」[22頁]。
3. 啟動 OpenAIS 並監控叢集狀態。如需詳細資訊，請參閱第 3.3 節「連線叢集」[24頁]。
4. 透過圖形使用者介面 (GUI) 或指令行新增並設定叢集資源。如需詳細資訊，請參閱第 4 章「使用 GUI 設定叢集資源」[29頁] 或第 5 章「從指令行設定叢集資源」[55頁]。

若要透過圍籬區隔和 STONITH 保護資料，以避免可能的損毀，請確認將 STONITH 設備設定為資源。如需詳細資訊，請參閱第 8 章「圍籬區隔與 STONITH」[75頁]。

如果共享磁碟 (儲存區域網路，即 SAN) 上尚不存在檔案系統，您可能還需要在其上建立檔案系統，並視需要將這些檔案系統設定為叢集資源。

叢集感知檔案系統 (OCFS2) 與非叢集感知檔案系統皆可透過 High Availability Extension 來設定。必要時，您也可以透過 DRBD 使用資料複製。如需詳細資訊，請參閱第 III 部分「儲存與資料複製」[99頁]。



# 3

## 使用 YaST 進行安裝及基本設定

安裝 High Availability 叢集所需軟體的方法有多種：使用 `zypper` 從指令行進行安裝，或使用提供圖形使用者介面的 YaST 進行安裝。在所有將成為叢集組成部分的節點上安裝軟體後，下一步就是對叢集進行初始設定，這樣節點之間就可以相互通訊。可以透過編輯組態檔案來手動完成該操作，或是使用 YaST 叢集模組完成操作。

---

**注：安裝軟體套件**

High Availability 叢集所需的軟體套件不會自動複製到叢集節點。在所有將成為叢集組成部分的節點上安裝 SUSE® Linux Enterprise Server 11 與 SUSE® Linux Enterprise High Availability Extension 11。

---

### 3.1 安裝 High Availability Extension

High Availability 安裝模式中包含了使用 High Availability Extension 設定與管理叢集所需的套件。僅當 SUSE® Linux Enterprise High Availability Extension 做為附加產品安裝後才能使用此模式。如需有關如何安裝附加產品的資訊，請參閱 SUSE Linux Enterprise 11 *部署指南* 中的「安裝附加產品」一章。

- 1 啟動 YaST，然後選取「軟體」>「軟體管理」，即會開啟 YaST 套件管理員。
- 2 從「過濾器」清單中選取「模式」，然後啟用模式清單中的「*High Availability*」模式。

- 3 按一下「接受」開始安裝這些套件。

## 3.2 初始叢集設定

安裝 HA 套件後，您可以使用 YaST 設定初始叢集設定。此設定包括節點間的通訊通道，以及使用加密通訊和將 OpenAIS 當做服務啟動等安全方面的設定。

對於通訊通道，需要定義結合網路位址 (bindnetaddr)、多路廣播位址 (mcastaddr) 以及多路廣播埠 (mcastport)。bindnetaddr 為要結合到的網路位址。為方便在整個叢集共享組態檔案，OpenAIS 使用網路介面網路遮罩來僅遮罩用於路由網路的地址位元。mcastaddr 可以是 IPv4 或 IPv6 多路廣播位址。mcastport 是為 mcastaddr 指定的 UDP 埠。

叢集中的節點可透過使用相同的多路廣播位址及相同的埠號互相瞭解。對於不同的叢集，請使用不同的多路廣播位址。

### 過程 3.1 設定叢集

- 1 啟動 YaST，然後選取「其他」>「叢集」或在指令行上執行 `yast2 cluster`，以啟動「初始叢集設定」對話方塊。
- 2 在「通訊通道」類別中，設定用於在叢集節點間通訊的通道。此資訊將寫入到 `/etc/ais/openais.conf` 組態檔案中。

The screenshot shows the 'Cluster - Communication Channels' dialog box in YaST. The left sidebar shows 'Communication Channels' selected under 'Security' and 'Service'. The main area contains the following fields and options:

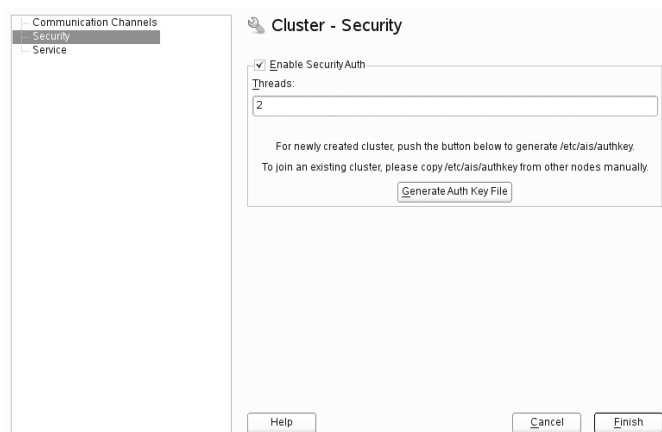
- Channel:**
  - Bind Network Address: 10.10.100.0 (dropdown)
  - Multicast Address: 226.94.1.1 (text input)
  - Multicast Port: 5405 (text input)
- Redundant Channel:**
  - Bind Network Address: (dropdown)
  - Multicast Address: (text input)
  - Multicast Port: (text input)
- Node ID:**
  - Auto Generate Node ID**
  - Node ID: 0 (text input)
- ip mode:** none (dropdown)

Buttons: Help, Cancel, Finish

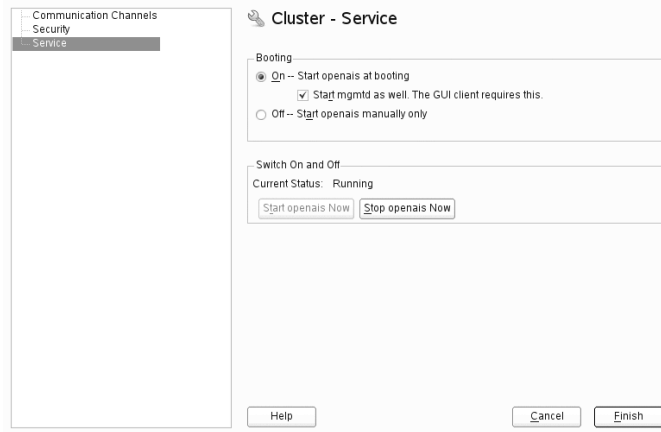
定義「結合網路位址」、「多路廣播位址」及「多路廣播埠」，以供所有叢集節點使用。

- 3 為每個叢集節點指定唯一的「節點 ID」。建議您從 1 開始指定。
- 4 在「安全性」類別中，定義叢集的驗證設定。如果啟用了「啟用安全性驗證」，則 HMAC/SHA1 驗證將用於叢集節點間的通訊。

此驗證方法需要共享秘密，用於保護和驗證訊息。您指定的驗證金鑰(密碼)將用於叢集中的所有節點。對於新建立的叢集，請按一下「產生驗證金鑰檔案」，建立將寫入到 /etc/ais/authkey 的驗證金鑰。



- 5 在「服務」類別中，選擇每次開機這台叢集伺服器時，是否啟動其上的 OpenAIS。



如果選取「關閉」，則每次開機這台叢集伺服器時，必須手動啟動 OpenAIS。若要手動啟動 OpenAIS，請使用 `rcopenais start` 指令。

若要立即啟動 OpenAIS，請按一下「立即啟動 OpenAIS」。

- 視需要設定所有選項之後，按一下「完成」。YaST 隨後還將自動調整防火牆設定，並開啟用於多路廣播的 UDP 埠。
- 完成初始組態設定後，您需要將該組態傳送到叢集中的其他節點。最簡便的方法是將 `/etc/ais/openais.conf` 檔案複製到叢集中的其他節點。由於每個節點需要一個唯一的節點 ID，因此請確保在複製檔案後相應調整節點 ID。
- 若要使用加密通訊，請同時將 `/etc/ais/authkey` 複製到叢集中的其他節點。

## 3.3 連線叢集

完成基本組態設定後，您可以將堆疊連線並檢查其狀態。

- 在每個叢集節點上執行以下指令，以啟動 OpenAIS：

```
rcopenais start
```
- 在其中一個節點上，使用以下指令檢查叢集的狀態：

```
crm_mon
```

如果所有節點都已上線，則輸出應如下所示：

```
=====
Last updated: Thu Feb  5 18:30:33 2009
Current DC: d42 (d42)
Version: 1.0.1-node: b7ffe2729e3003ac8ff740bebc003cf237dfa854
3 Nodes configured.
0 Resources configured.
=====

Node: d230 (d230): online
Node: d42 (d42): online
Node: e246 (e246): online
```

完成基本組態設定並將節點連線之後，您便可立即開始使用 `crm` 指令行工具或圖形使用者介面設定叢集資源。如需更多資訊，請參閱第 4 章「*使用 GUI 設定叢集資源*」[29頁] 或第 5 章「*從指令行設定叢集資源*」[55頁]。



## II. 組態與管理



# 4

## 使用 GUI 設定叢集資源

HA 叢集的主要目的是管理使用者服務。使用者服務的典型範例就是 Apache Web 伺服器或資料庫。從使用者的角度來看，命令這些服務執某些特定作業，它們就會按要求執行。不過，對於叢集而言，它們只是可以啟動或停止的資源——服務的性質與叢集無關。

做為叢集管理員，您需要為您叢集中的伺服器上執行的所有資源或應用程式建立叢集資源。叢集資源可包括網站、電子郵件伺服器、資料庫、檔案系統、虛擬機器，以及其他您希望使用者隨時都可以存取的伺服器型應用程式或服務。

若要建立叢集資源，請使用圖形使用者介面 (Linux HA Management Client) 或 `crm` 指令行公用程式。有關指令行這種方式，請參閱第 5 章「從指令行設定叢集資源」[55頁]。

本章對 Linux HA Management Client 做了介紹，並且涵蓋了設定叢集時所需瞭解的幾個主題：建立資源、設定限制、指定容錯移轉節點與錯誤回復節點、設定資源監控、啟動或移除資源、設定資源群組或複製資源，以及手動移轉資源。

`pacemaker-pygui` 套件中包含用於設定叢集資源的圖形使用者介面。

### 4.1 Linux HA Management Client

當您啟動 Linux HA Management Client 時，需要連接至一個叢集。

---

**注：hacluster 使用者的密碼**

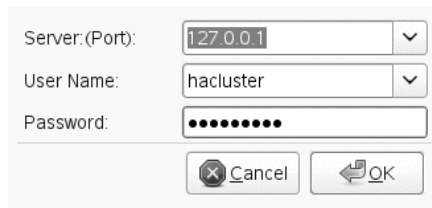
安裝會建立名為 hacluster 的 Linux 使用者。在使用 Linux HA Management Client 之前，必須設定 hacluster 使用者的密碼。若要執行此操作，請以 root 的身份在指令行中輸入 `passwd hacluster`，並輸入 hacluster 使用者的密碼。

在需要使用 Linux HA Management Client 連接的所有節點上執行此操作。

---

若要啟動 Linux HA Management Client，請在指令行中輸入 `crm_gui`。若要連接至叢集，請選取「連線」>「登入」。依預設，「伺服器」欄位會顯示 localhost 的 IP 位址，以及做為「使用者名稱」的 hacluster。輸入使用者密碼以繼續。

**圖形 4.1** 連接至叢集

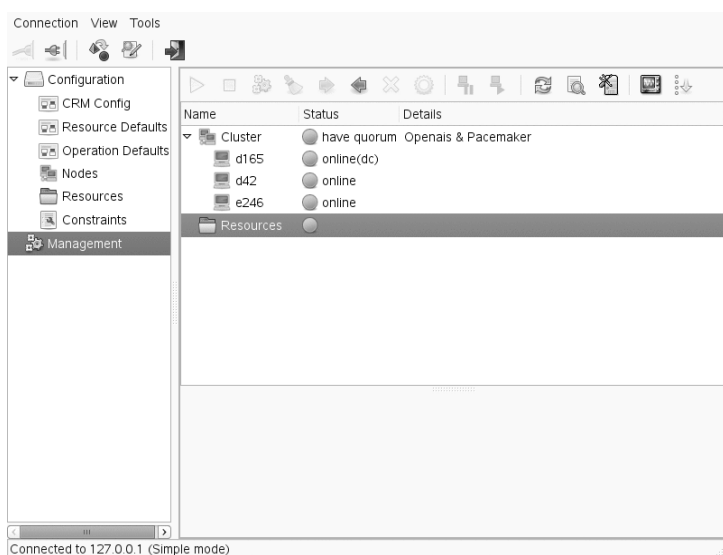


The image shows a graphical user interface for connecting to a cluster. It consists of three input fields stacked vertically. The first field is labeled 'Server:(Port)' and contains the text '127.0.0.1'. The second field is labeled 'User Name' and contains the text 'hacluster'. The third field is labeled 'Password' and contains a series of dots. Below these fields are two buttons: 'Cancel' and 'OK'.

若要從遠端執行 Linux HA Management Client，請輸入做為「伺服器」之叢集節點的 IP 位址。對於「使用者名稱」，您也可以使用屬於 haclient 群組的任何其他使用者來連接叢集。

連接之後，主視窗即會開啟：

圖形 4.2 Linux HA Management Client - 主視窗



Linux HA Management Client 可讓您新增及修改資源、限制、組態等。它還提供了管理叢集元件的功能，如啟動、停止或移轉資源、清理資源或將節點設定為待機模式。此外，您只需選取任意「組態」子項目並選取「顯示」>「XML 模式」，即可輕鬆檢視、編輯、輸入和輸出 CIB 的 XML 結構。

下文中提供了一些如何使用 Linux HA Management Client 建立和管理叢集資源的範例。

## 4.2 建立叢集資源

您可以建立以下類型的資源：

### 原始

原始資源，是最基本的資源類型。

### 群組

群組包含一組資源，它們需放置在一起，依順序啟動並以相反順序停止。若需更多資訊，請參閱第 4.10 節「設定叢集資源群組」[46頁]。

## 複製品

複製品是可在多個主機上處於使用中狀態的資源。任何資源都可複製，只要相應的資源代辦支援複製功能。若需更多資訊，請參閱第 4.11 節「設定複製品資源」[51頁]。

## 主複製本

主複製本是複製品資源的一種特殊類型，它可以擁有多個模式。主複製本只能包含一個群組或一個一般資源。

### 過程 4.1 新增原始資源

- 1 啟動 Linux HA Management Client，並依第 4.1 節「Linux HA Management Client」[29頁]中所述登入叢集。
- 2 在左側窗格中，選取「資源」並按一下「新增」>「原始」。
- 3 在下一個對話方塊中，設定資源的以下參數：
  - 3a 輸入資源的唯一 ID。
  - 3b 在「類別」清單中，選取您要針對該資源使用的資源代辦類別：「*heartbeat*」、「*lsb*」、「*ocf*」或「*stonith*」。如需詳細資訊，請參閱第 17.1 節「受支援的資源代辦類別」[183頁]。
  - 3c 若選取了「*ocf*」類別，請另外指定 OCF 資源代辦的「提供者」。OCF 規格允許多個廠商提供相同的資源代辦。
  - 3d 在「類型」清單中，選取要使用的資源代辦(例如「*IPaddr*」或「*Filesystem*」)。此資源代辦的簡要描述會顯示在下方。

「類型」清單中顯示的選項取決於您所選的「類別」(對於 OCF 資源，還取決於「提供者」)。
  - 3e 在「選項」下方，設定「資源的初始狀態」。
  - 3f 若要讓叢集監控資源是否正常，請啟用「新增監控作業」。

**Add Primitive - Basic Settings**

**Required**

ID:

Class:

Provider:

Type:

**Description**

Manages virtual IPv4 addresses.

This script manages IP alias IP addresses  
It can add an IP alias, or remove one.

**Options**

Initial state of resource:

Add monitor operation

4 按「下一步」。下一個視窗會顯示您已為該資源定義的參數摘要。其中會列出該資源所需的所有「例項屬性」。您需要對它們進行編輯，以將其設定為適當的值。依您的部署與設定而定，您可能還需要新增其他屬性。如需如何執行此操作的詳細資料，請參閱新增或修改中繼屬性與例項屬性 [34頁]。

5 按需要設定了所有參數後，請按一下「套用」以完成該資源的組態設定。組態對話方塊會關閉，同時主視窗會顯示新增的資源。

您可以隨時新增或修改原始資源的以下參數：

#### 中繼屬性

中繼屬性是可為資源新增的選項。它們會告訴 CRM 如何處理特定資源。如需可用中繼屬性、中繼屬性值及預設值的綜覽，請參閱第 17.3 節「資源選項」 [186頁]。

#### 例項屬性

例項屬性是特定資源類別的參數，可決定資源的行為方式以及資源所控制的服務例項。若需更多資訊，請參閱第 17.5 節「例項屬性」 [188頁]。

## 作業

為資源新增的監控作業。它們會指示叢集來確定資源是否仍處於正常狀態。您可為所有類別的資源代辦新增監控作業。您還可以設定特定參數，例如為 `start` 或 `stop` 作業設定 `Timeout`。若需更多資訊，請參閱第 4.7 節「設定資源監控」[43頁]。

### 過程 4.2 新增或修改中繼屬性與例項屬性

- 1 在 Linux HA Management Client 主視窗中，按一下左側窗格中的「資源」，以檢視已為叢集設定的資源。
- 2 在右側窗格中，選取要修改的資源，然後按一下「編輯」(或在該資源上連接兩下)。下一個視窗會顯示基本的資源參數，以及已為該資源定義的「中繼屬性」、「例項屬性」或「作業」。

The screenshot shows a configuration window for a resource. At the top, there is a 'Show' dropdown set to 'List Mode'. Below this, the 'Required' section contains fields for ID (my\_ipaddress), Class (ocf), Provider (heartbeat), and Type (IPaddr). An 'Optional' section is collapsed. The 'Description' section states: 'Manages virtual IPv4 addresses. This script manages IP alias IP addresses. It can add an IP alias, or remove one.' Below the description is a table with three tabs: 'Meta Attributes', 'Instance Attributes', and 'Operations'. The 'Instance Attributes' tab is active, showing a table with one row: 'ip' with value '192.168.1.1'. To the right of the table are 'Up' and 'Down' buttons. Below the table, the ID is 'nvpair-4424c744-a46b-4af3-8623-101b58926936', Name is 'ip', and Value is '192.168.1.1'. At the bottom are buttons for '+ Add', 'Edit', '- Remove', 'Cancel', 'Reset', and 'OK'.

- 3 若要新增新的中繼屬性或例項屬性，請選取相應的索引標籤並按一下「新增」。
- 4 選取要新增之屬性的「名稱」。一個簡要「描述」即會顯示。
- 5 視需要指定屬性「值」。若不指定，系統將使用該屬性的預設值。

- 6 按一下「確定」以確認所做的變更。索引標籤上隨即會顯示新增的或修改的屬性。
- 7 按需要設定了所有參數後，請按一下「確定」以完成該資源的組態設定。組態對話方塊會關閉，同時主視窗會顯示修改的資源。

---

**提示：XML 原始程式碼**

Linux HA Management Client 可讓您檢視透過您為特定資源或所有資源定義的參數產生的 XML。在資源組態對話方塊的右上角或主視窗的「資源」檢視窗中，選取「顯示」>「XML 模式」。

顯示 XML 程式碼的編輯器可讓您「輸入」或「輸出」XML 元素，或手動編輯 XML 程式碼。

---

## 4.3 建立 STONITH 資源

若要設定圍籬區隔，您需要設定一或多個 STONITH 資源。

### 過程 4.3 新增 STONITH 資源

- 1 啟動 Linux HA Management Client，並依第 4.1 節「Linux HA Management Client」[29頁] 中所述登入叢集。
- 2 在左側窗格中，選取「資源」並按一下「新增」>「原始」。
- 3 在下一個對話方塊中，設定資源的以下參數：
  - 3a 輸入資源的唯一 ID。
  - 3b 在「類別」清單中，選取資源代辦類別「*stonith*」。
  - 3c 從「類型」清單中，選取用於控制 STONITH 設備的 STONITH 外掛程式。此外掛程式的簡要描述會顯示在下方。
  - 3d 在「選項」下方，設定「資源的初始狀態」。
  - 3e 若要讓叢集監控圍籬區隔設備，請啟用「新增監控作業」。若需更多資訊，請參閱第 8.4 節「監控圍籬區隔設備」[82頁]。

- 4 按「下一步」。下一個視窗會顯示您已為該資源定義的參數摘要。其中會列出所選 STONITH 外掛程式需要的所有「例項屬性」。您需要對它們進行編輯，以將其設定為適當的值。依您的部署與設定而定，您可能還需要新增其他屬性或監控作業。如需如何執行此操作的詳細資料，請參閱新增或修改中繼屬性與例項屬性 [34頁] 與第 4.7 節「設定資源監控」[43頁]。
- 5 按需要設定了所有參數後，請按一下「套用」以完成該資源的組態設定。組態對話方塊會關閉，同時主視窗會顯示新增的資源。

若要完成圍籬區隔組態，請新增限制或使用複製品，或同時使用這兩種方式。如需詳細資訊，請參閱第 8 章「圍籬區隔與 STONITH」[75頁]。

## 4.4 設定資源限制

設定所有資源只是工作的一部分。即使叢集瞭解所有必需的資源，可能仍然無法正確地對其進行處理。資源限制可讓您指定資源可在哪些叢集節點上執行，載入資源的順序以及特定資源所依賴的其他資源。

系統中的限制分為三種類型：

### 「資源位置」

位置限制，定義資源可在、不可在或偏好在哪些節點上執行。

### 「資源並存」

並存限制，告訴叢集哪些資源可以或不可以同時在節點上執行。

### 「資源順序」

順序限制，定義動作執行的順序。

定義限制時，還需要處理分數。所有類型的分數對於叢集的工作方式而言都是不可或缺的。實際上，從移轉資源到決定要將降級叢集中哪個資源停止的所有作業，都是透過某種方式對分數進行操作來實現。系統會對每個資源都計算分數，針對某個資源分數為負數的所有節點都不能執行該資源。計算完針對資源的分數之後，叢集會選擇分數最高的節點。INFINITY 目前定義為 1,000,000。與其相關的加法或減法計算遵循以下 3 項基本規則：

- 任何值 + INFINITY = INFINITY
- 任何值 - INFINITY = -INFINITY

- $\text{INFINITY} - \text{INFINITY} = -\text{INFINITY}$

定義資源限制時，您還要指定每個限制的分數。分數表示您要指定給此資源限制的值。系統會先套用分數較高的限制，然後再套用分數較低的限制。透過為指定資源建立不同分數的其他位置限制，即可指定資源將容錯移轉至之目標節點的順序。

#### 過程 4.4 新增或修改位置限制

- 1 啟動 Linux HA Management Client，並依第 4.1 節「Linux HA Management Client」[29頁] 中所述登入叢集。
- 2 在 Linux HA Management Client 主視窗中，按一下左側窗格中的「限制」，檢視已為叢集設定的限制。
- 3 在左側窗格中，選取「限制」，然後按一下「新增」。
- 4 選取「資源位置」，然後按一下「確定」。
- 5 輸入限制的唯一「ID」。在您修改現有的限制時，其 ID 已定義好，會顯示在組態對話方塊中。
- 6 選取要對其定義限制的「資源」。清單會顯示已為叢集設定之所有資源的 ID。
- 7 為限制設定「分數」。正值表示資源可以在您於下方指定的「節點」上執行。負值表示資源不能在此節點上執行。 $+/- \text{INFINITY}$  值則表示將「可以」變更為「必須」。
- 8 為限制選取「節點」。

Required

Show: List Mode

ID: my\_location\_constraint

Resource: my\_stonith

Score: INFINITY

Node: d42

+ Add Edit - Remove

Cancel Reset OK

- 9 若將「節點」與「分數」欄位留為空白，您也可以透過按一下「新增」>「規則」來新增規則。若要新增生命期間，只需按一下「新增」>「生命期間」。
- 10 按需要設定了所有參數後，請按一下「確定」以完成該限制的組態設定。組態對話方塊會關閉，同時主視窗會顯示新增的或修改的限制。

#### 過程 4.5 新增或修改並存限制

- 1 啟動 Linux HA Management Client，並依第 4.1 節「Linux HA Management Client」[29頁] 中所述登入叢集。
- 2 在 Linux HA Management Client 主視窗中，按一下左側窗格中的「限制」，檢視已為叢集設定的限制。
- 3 在左側窗格中，選取「限制」，然後按一下「新增」。
- 4 選取「資源並存」，然後按一下「確定」。
- 5 輸入限制的唯一「ID」。在您修改現有的限制時，其 ID 已定義好，會顯示在組態對話方塊中。
- 6 選取做為並存來源的「資源」。清單會顯示已為叢集設定之所有資源的 ID。

若無法符合限制的要求，叢集可能會決定不允許執行該資源。

- 7 若將「資源」及「與資源」欄位留為空白，您也可以透過按一下「新增」>「資源集」來新增資源集。若要新增生命期間，只需按一下「新增」>「生命期間」。
- 8 在「與資源」中，定義並存目標。叢集會先決定放置此資源的位置，然後決定放置「資源」欄位中之資源的位置。
- 9 定義「分數」以決定兩個資源間的位置關係。正值表示資源應該在同一個節點上執行。負值表示資源不應該在同一個節點上執行。+/- INFINITY 值則表示將「應該」變更為「必須」。該分數會結合其他因素來決定放置節點的位置。
- 10 如有需要，請指定其他參數，例如「資源角色」。  
  
根據所選的參數與選項，系統會顯示一條簡要「描述」，說明要設定之並存限制的效果。
- 11 按需要設定了所有參數後，請按一下「確定」以完成該限制的組態設定。組態對話方塊會關閉，同時主視窗會顯示新增的或修改的限制。

#### 過程 4.6 新增或修改順序限制

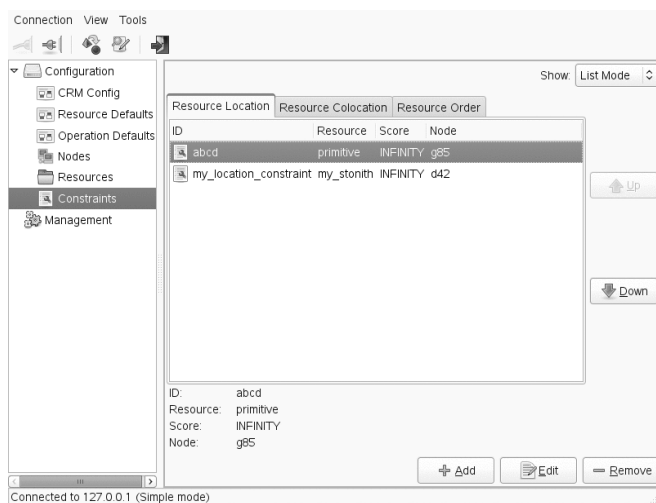
- 1 啟動 Linux HA Management Client，並依第 4.1 節「Linux HA Management Client」[29頁] 中所述登入叢集。
- 2 在 Linux HA Management Client 主視窗中，按一下左側窗格中的「限制」，檢視已為叢集設定的限制。
- 3 在左側窗格中，選取「限制」，然後按一下「新增」。
- 4 選取「資源順序」，然後按一下「確定」。
- 5 輸入限制的唯一「ID」。在您修改現有的限制時，其 ID 已定義好，會顯示在組態對話方塊中。
- 6 使用「首先」，定義在「接著」資源允許啟動之前必須先啟動的資源。
- 7 使用「接著」，定義在「首先」資源啟動之後將要啟動的資源。
- 8 視需要定義其他參數，例如「分數」(若大於零，則限制為強制性；否則僅做為建議)，或「對稱」(若為 true，則以相反順序停止資源)。

根據所選的參數與選項，系統會顯示一條簡要「描述」，說明要設定之順序限制的效果。

- 9 按需要設定了所有參數後，請按一下「確定」以完成該限制的組態設定。組態對話方塊會關閉，同時主視窗會顯示新增的或修改的限制。

您可在 Linux HA Management Client 的「限制」檢視窗中存取和修改已設定的所有限制。

**圖形 4.3** *Linux HA Management Client - 限制*



如需設定限制的詳細資訊，以及有關順序與並存基本概念的詳細背景資訊，請參閱<http://clusterlabs.org/wiki/Documentation> 中提供的以下文件：

- 《*Configuration 1.0 Explained*》(Configuration 1.0 說明) 中的「*Resource Constraints*」(資源限制) 一章
- *Collocation Explained* (並存說明)
- *Ordering Explained* (順序說明)

## 4.5 指定資源容錯移轉節點

若資源失敗，系統會自動將其重新啟動。若在目前節點上無法將其重新啟動，或資源已在目前節點上失敗 N 次，則資源會嘗試容錯移轉至其他節點。您可以定義一個數值，讓資源在失敗該次數 (`migration-threshold`) 之後移轉至新節點。若叢集中有兩個以上的節點，則特定資源容錯移轉所至的節點由 `High Availability` 軟體來選擇。

若您要選擇資源將容錯移轉所至的節點，則必須執行以下動作：

- 1 依新增或修改位置限制 [37頁] 中所述為該資源設定位置限制。
- 2 依新增或修改中繼屬性與例項屬性 [34頁] 中所述將 `migration-threshold` 中繼屬性新增至該資源，並為該 `migration-threshold` 輸入「值」。該值應該為小於 `INFINITY` 的正數。
- 3 若要讓資源的 `failcount` 自動過期，請依新增或修改中繼屬性與例項屬性 [34頁] 中所述將 `failure-timeout` 中繼屬性新增至該資源，並為該 `failure-timeout` 輸入「值」。
- 4 若要指定具有資源優先設定的其他容錯移轉節點，請建立其他位置限制。

例如，假設您已為資源 `r1` 設定位置限制，讓其偏向於在 `node1` 上執行。若資源在該節點上失敗，系統會檢查 `migration-threshold`，並將其與 `failcount` 進行比較。若 `failcount >= migration-threshold`，則將資源移轉至優先設定次佳的節點。

依預設，一旦達到限定值，則不再允許該節點執行失敗的資源，除非管理員手動重設資源的 `failcount` (在修復失敗原因之後)。

不過，透過設定資源的 `failure-timeout` 選項可以使失敗計數過期。因此，設定 `migration-threshold=2` 及 `failure-timeout=60s` 會導致資源在兩次失敗後將移轉至新節點，並且可能允許其在一分鐘後移回原節點 (具體視相粘性與限制分數而定)。

移轉限定值概念有兩種例外情況，發生於資源無法啟動或無法停止時：啟動失敗將 `failcount` 設定為 `INFINITY`，因此始終導致資源立即移轉。停止失敗會導致圍籬區隔 (當 `stonith-enabled` 設定為預設值 `true` 時)。若未定義任何

STONITH 資源 (或將 `stonith-enabled` 設定為 `false`)，則資源一律不會移轉。

若要使用 Linux HA Management Client 清理資源的 `failcount`，請在左側窗格中選取「管理」，並在右側窗格中選取相應的資源，然後按一下工具列中的「清理資源」。如此會在指定節點上針對指定資源執行指令 `crm_resource -C 與 crm_failcount -D`。如需詳細資訊，另請參閱 `crm_resource(8)` [157頁] 和 `crm_failcount(8)` [148頁]。

## 4.6 指定資源錯誤回復節點 (資源相粘性)

當原始節點恢復連線且位於叢集中時，資源可以錯誤回復至該節點。若不想讓資源錯誤回復至資源在容錯移轉之前所處的節點，或要為資源指定另一個要錯誤回復至的節點，您必須變更其資源相粘性的值。您可以在建立資源時或建立之後指定資源相粘性。

指定資源粘性的值時，請考慮以下事項：

值為 0：

此為預設值。資源處於系統中的最佳位置。這表示當有「更佳」的或負載更低的節點可用時，則移動資源。此選項幾乎等同於自動錯誤回復，除了資源可能會移至原先節點 (資源之前於其上處於使用中狀態) 之外的節點這種情況。

值大於 0：

資源偏向於保留在其目前的位置，但當有更合適的節點時，則可能會移動。值越高表示越偏向於將資源保留在目前的位置。

值小於 0：

資源偏向於從其目前位置移開。絕對值越高表示越偏向於移動資源。

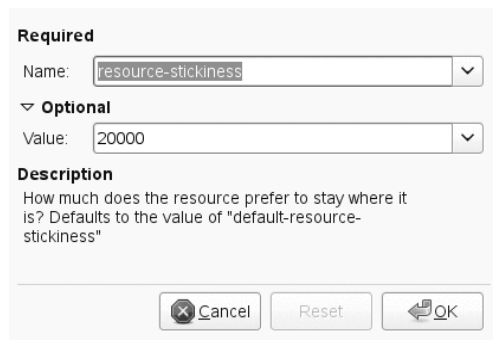
值為 INFINITY：

資源始終保留在其目前的位置，除非由於此節點不再符合執行該資源的條件 (節點關機、節點待機、達到 `migration-threshold`，或組態變更) 而強制關閉此選項。此選項幾乎等同於完全停用自動錯誤回復。

值為  $-\text{INFINITY}$ ：  
資源始終從其目前位置移開。

#### 過程 4.7 指定資源相粘性

- 1 依新增或修改中繼屬性與例項屬性 [34頁] 中所述將 `resource-stickiness` 中繼屬性新增至資源。



The screenshot shows a configuration dialog box with the following fields and controls:

- Required** section: A dropdown menu with the value `resource-stickiness`.
- Optional** section: A dropdown menu with the value `20000`.
- Description** section: Text explaining the parameter: "How much does the resource prefer to stay where it is? Defaults to the value of 'default-resource-stickiness'".
- Buttons at the bottom: **Cancel**, **Reset**, and **OK**.

- 2 對於 `resource-stickiness` 的「值」，請指定介於  $-\text{INFINITY}$  與  $\text{INFINITY}$  之間的值。

## 4.7 設定資源監控

High Availability Extension 不僅可以偵測節點失敗，而且還可以偵測節點上個別資源失敗的時間。若要確定資源是否正在執行，您必須針對該資源設定資源監控。資源監控包括指定逾時與/或啟動延遲值以及間隔。該間隔會告知 CRM 應檢查資源狀態的頻率。

#### 過程 4.8 新增或修改監控作業

- 1 啟動 Linux HA Management Client，並依第 4.1 節「Linux HA Management Client」 [29頁] 中所述登入叢集。
- 2 在 Linux HA Management Client 主視窗中，按一下左側窗格中的「資源」，以檢視已為叢集設定的資源。

- 3 在右側窗格中，選取要修改的資源，然後按一下「編輯」。下一個視窗會顯示基本的資源參數，以及已為該資源定義的中繼屬性、例項屬性及作業。
- 4 若要新增新的監控作業，請選取相應的索引標籤，然後按一下「新增」。  
若要修改現有作業，請選取相應的項目，然後按一下「編輯」。
- 5 輸入監控作業的唯一「ID」。在您修改現有的監控作業時，其 ID 已定義好，會顯示在組態對話方塊中。
- 6 在「名稱」中，選取要執行的動作，例如「監控」、「啟動」，或「停止」。
- 7 在「間隔」欄位中輸入值(以秒計)。
- 8 在「逾時」欄位中輸入值(以秒計)。作業在經過指定的逾時期間之後會被視為 failed。PE 將會決定需要採取的措施，或執行您在監控作業的「失敗時」欄位中指定的動作。
- 9 視需要設定選擇性參數，例如「失敗時」(若此動作失敗時應採取何措施?)或「必要」(執行此動作之前需要符合哪些條件?)。

Dialog box configuration for a monitoring job:

- Show: List Mode
- ID: primitive-op-monitor-15
- Name: monitor
- Interval: 15
- Timeout: 15
- Optional section:
  - Description: (empty)
  - Start Delay: 15
  - Interval Origin: (empty)
  - Enabled: (dropdown)
  - Record Pending: (dropdown)
  - Role: (dropdown)
  - Requires: (dropdown)
  - On Fail: (dropdown)
- Buttons: Add, Edit, Remove, Cancel, Reset, OK

- 10 按需要設定了所有參數後，請按一下「確定」以完成該資源的組態設定。組態對話方塊會關閉，同時主視窗會顯示修改的資源。

若不設定資源監控，則不會向您通知資源成功啟動後發生的失敗，並且叢集會始終將資源顯示為處於正常狀態。

若資源監控偵測到失敗，系統將會執行以下動作：

- 根據 `/etc/ais/openais.conf` 的 `logging` 區段中指定的組態產生記錄檔案訊息 (預設會寫入 `syslog`，通常為 `/var/log/messages`)。
- 在 Linux HA Management Client、`crm_mon` 工具及 CIB 狀態區段中會反應出失敗。若要在 Linux HA Management Client 中檢視失敗，請在左側窗格中按一下「管理」，然後在右側窗格中選取要檢視其詳細資料的資源。
- 叢集會啟動重要的復原動作，其中可能包括停止資源以修復失敗狀態，以及在本地或在其他節點上重新啟動資源。也可能根本不重新啟動資源，具體視組態及叢集狀態而定。

## 4.8 啟動新的叢集資源

---

**注：啟動資源**

使用 High Availability Extension 設定資源時，不應手動啟動或停止相同的資源 (在叢集外部)。High Availability Extension 軟體負責所有服務的啟動或停止動作。

---

建立資源期間，若將其初始狀態設定為 `stopped` (`target-role` 中繼屬性的值設定為 `stopped`)，則建立資源後其將不會自動啟動。若要使用 Linux HA Management Client 來啟動新的叢集資源，請在左側窗格中選取「管理」。在右側窗格中，在資源上按一下滑鼠右鍵，然後選取「啟動」(或從工具列中將資源啟動)。

## 4.9 移除叢集資源

若要使用 Linux HA Management Client 移除叢集資源，請切換至左側窗格中的「資源」檢視窗，然後選取相應的資源並按一下「移除」。

---

### 注：移除參考資源

若有限制正在參考叢集資源的 ID，則不能移除叢集資源。若無法刪除某個資源，請檢查參考該資源 ID 的位置，先從限制中移除該資源。

---

## 4.10 設定叢集資源群組

某些叢集資源依賴於其他元件或資源，要求每個元件或資源以特定順序啟動，並在同一個伺服器上執行。若要簡化此組態，您可以使用群組。

群組具有以下內容：

### 啟動和停止資源

系統是以資源顯示的順序來啟動資源，並以其顯示的相反順序停止資源。

### 相依性

若群組中的某個資源在任何地方都無法執行，則群組中位於該資源之後的所有資源均不允許執行。

### 群組內容

群組僅會包含原始叢集資源的集合。若要參閱群組資源的子代，請使用子代 ID，而不是群組的 ID。

### 限制

儘管您可以在限制中參考群組的子代，但一般最好使用群組的名稱。

### 相粘性

相粘性在群組中屬於加法類內容。群組中每個使用中成員的粘性值都會影響群組的總值。因此，若 `resource-stickiness` 預設值為 100，並且群組有 7 個成員，其中 5 個處於使用中狀態，則整個群組偏向於其目前位置的分數為 500。

### 資源監控

若要對群組啟用資源監控，您必須為要監控的群組中的每個資源分別設定監控。

---

**注：空群組**

群組至少須包含一個資源，否則組態視為無效。

---

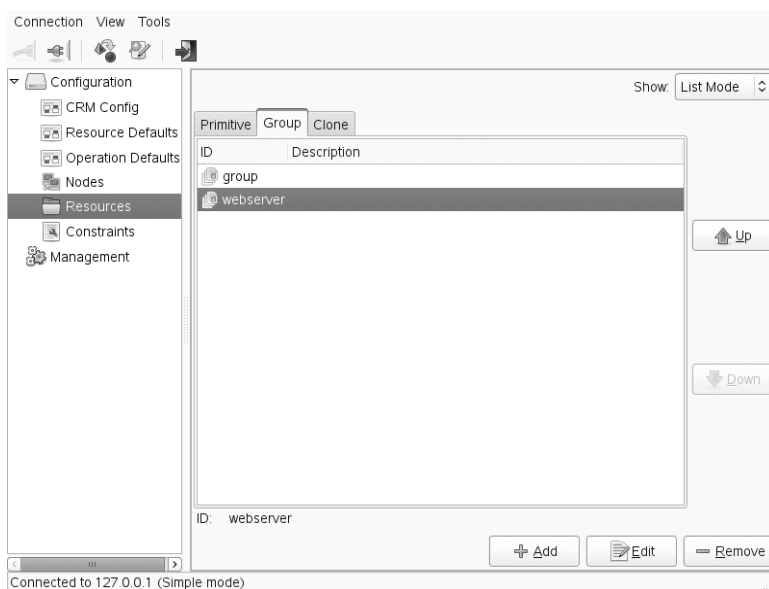
**過程 4.9 新增資源群組**

- 1 啟動 Linux HA Management Client，並依第 4.1 節「Linux HA Management Client」[29頁]中所述登入叢集。
- 2 在左側窗格中，選取「資源」並按一下「新增」>「群組」。
- 3 輸入群組的唯一「ID」。
- 4 在「選項」下方，設定「資源的初始狀態」，然後按「下一步」。
- 5 在下一步中，您可以將原始資源新增為群組的子資源。建立這些資源的方式與新增原始資源 [32頁]中所述方式類似。
- 6 按需要設定了所有參數後，請按一下「套用」以完成該原始資源的組態設定。
- 7 在下一個視窗中，您可以再次選擇「原始」並按一下「確定」，繼續為群組新增子資源。

如果不想為群組新增更多原始資源，請按一下「取消」。下一個視窗會顯示已為該群組定義的參數摘要。其中會列出群組的「中繼屬性」與「原始資源」。「原始」索引標籤中資源的位置表示該資源在叢集中的啟動順序。

- 8 由於群組中資源的順序很重要，因此請使用「向上」與「向下」按鈕，對群組中的「原始資源」進行排序或重新排序。
- 9 按需要設定了所有參數後，請按一下「確定」以完成該群組的組態設定。組態對話方塊會關閉，同時主視窗會顯示新建的或修改的群組。

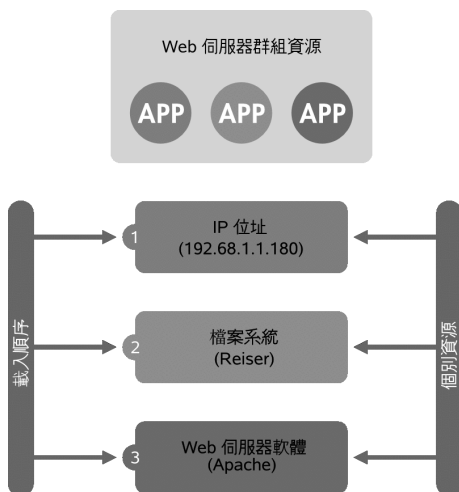
圖形 4.4 Linux HA Management Client - 群組



#### 範例 4.1 Web 伺服器的資源群組

資源群組的其中一個範例就是需要 IP 位址與檔案系統的 Web 伺服器。在此案例中，每個元件都是組合到叢集資源群組中的獨立的叢集資源。資源群組會在一或多個伺服器上執行，若軟體或硬體出現異常，資源群組會容錯移轉至叢集中的其他伺服器，這一點與個別叢集資源相同。

圖形 4.5 群組資源

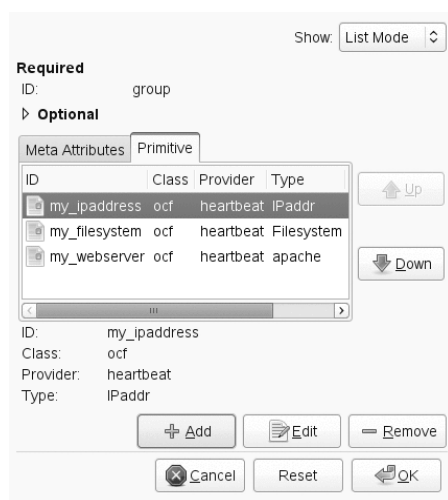


在新增資源群組 [47頁] 中，您會瞭解如何建立資源群組。假設您已依照上文中的說明建立資源群組。將資源新增至現有群組 [49頁] 會向您說明如何修改群組以符合範例 4.1 「Web 伺服器的資源群組」 [48頁]。

#### 過程 4.10 將資源新增至現有群組

- 1 啟動 Linux HA Management Client，並依第 4.1 節「Linux HA Management Client」 [29頁] 中所述登入叢集。
- 2 在左側窗格中，切換至「資源」檢視窗，然後在右側窗格中選取要修改的群組，並按一下「編輯」。下一個視窗會顯示基本的群組參數，以及已為該資源定義的中繼屬性與原始資源。
- 3 按一下「原始資源」索引標籤，然後按一下「新增」。
- 4 在下一個對話方塊中，設定以下參數以將 IP 位址新增為群組的子資源：
  - 4a 輸入唯一的「ID」，例如 `my_ipaddress`。
  - 4b 在「類別」清單中，選取「`ocf`」做為資源代辦類別。
  - 4c 對於 OCF 資源代辦的「提供者」，選取「`heartbeat`」。

- 4d** 在「類型」清單中，選取「IPaddr」做為資源代辦。
- 4e** 按「下一步」。
- 4f** 在「例項屬性」索引標籤中，選取「IP」項目並按一下「編輯」(或在「IP」項目上連接兩下)。
- 4g** 對於「值」，輸入所需的 IP 位址，例如 192.168.1.1。
- 4h** 按一下「確定」與「套用」。群組組態對話方塊會顯示新增的原始資源。
- 5** 再按一下「新增」以新增下一個子資源 (檔案系統與 Web 伺服器)。
- 6** 為每個子資源設定相應的參數 (類似於步驟步驟 4a [49頁] 至步驟 4h [50頁])，直到您已設定此群組的所有子資源。



由於我們是依照子資源在叢集中的啟動順序對其進行設定，因此「原始資源」索引標籤中的順序已經是正確的。

- 7** 若需要變更群組的資源順序，請使用「向上」與「向下」按鈕對「原始」索引標籤中的資源重新排序。

- 8 若要從群組中移除某個資源，請在「原始資源」索引標籤中選取該資源，然後按一下「移除」。
- 9 按一下「確定」以完成該群組的組態設定。組態對話方塊會關閉，同時主視窗會顯示修改的群組。

## 4.11 設定複製品資源

您可能要讓某些資源同時在叢集的多個節點上執行。若要實現此目的，您必須將資源設定為複製品。可以設定為複製品的資源範例包括 STONITH 以及叢集檔案系統，如 OCFS2。只要資源的資源代辦支援複製功能，您就可複製任何資源。您甚至可以對複製品資源進行不同的設定，具體視代管它們的節點而定。

資源複製品分為三種類型：

### 匿名複製品

這是最簡單的一種複製品。無論在何處執行，它們的行為都相同。因此，每部機器上只能有一個匿名複製品例項處於使用中狀態。

### 全域唯一複製品

這些資源是不同的實體。在一個節點上執行的複製品例項與在另一個節點上執行的另一個例項不同；在相同節點上的任何兩個例項也不相同。

### 狀態複製品

這些資源的使用中例項分為兩種狀態：主動與被動。這兩種狀態有時也稱為主要與次要，或主要與從屬。狀態複製品可以是匿名複製品，也可以是全域唯一複製品。

### 過程 4.11 新增或修改複製品

- 1 啟動 Linux HA Management Client，並依第 4.1 節「Linux HA Management Client」[29頁] 中所述登入叢集。
- 2 在左側窗格中，選取「資源」並按一下「新增」>「複製品」。
- 3 輸入複製品的唯一「ID」。
- 4 在「選項」下方，設定「資源的初始狀態」。
- 5 啟用要為複製品設定的相應選項，然後按「下一步」。

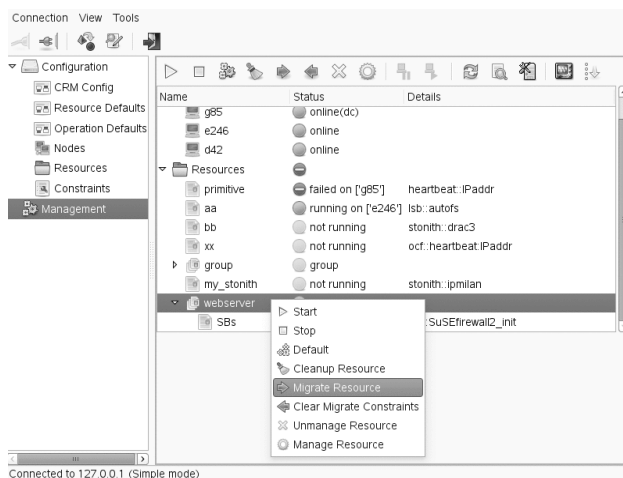
- 6 在下一步中，您可以新增「原始」或「群組」做為複製品的子資源。建立這些資源的方式與新增原始資源 [32頁] 或新增資源群組 [47頁] 中所述方式類似。
- 7 按需要在複製品組態對話方塊中設定了所有參數後，請按一下「套用」以完成該複製品的組態設定。

## 4.12 移轉叢集資源

如第 4.5 節「指定資源容錯移轉節點」 [41頁] 中所述，當軟體或硬體發生失敗時，叢集會自動對資源進行容錯移轉(移轉)—具體情況視您可以定義的某些參數(例如移轉限定值或資源相粘性)。除此之外，您還可以手動將資源移轉至叢集資源中的其他節點。

### 過程 4.12 手動移轉資源

- 1 啟動 Linux HA Management Client，並依第 4.1 節「Linux HA Management Client」 [29頁] 中所述登入叢集。
- 2 切換至左側窗格中的「管理」檢視窗，然後在右側窗格中的相應資源上按一下滑鼠右鍵，並選取「移轉資源」。



- 3 在新視窗中，在「至節點」中選取要將資源移至的節點。如此會建立一個位置限制，其目的節點的分數為 INFINITY。
- 4 若只想暫時移轉資源，請啟用「持續時間」，並輸入資源移轉至新節點後應保留的時間。經過這段持續時間之後，資源可以移回其原始位置，也可以保留在目前的位置 (具體取決於資源相粘性)。
- 5 如果資源無法移轉 (若資源相粘性及限制總分大於目前節點上的 INFINITY)，請啟用「強制」選項。如此會為目前位置建立規則以及一個 -INFINITY 分數，以強制資源移動。

---

#### 注

這樣將會防止資源在此節點上執行，直到使用「清除移轉限制」移除限制，或持續時間過期。

---

- 6 按一下「確定」以確認移轉。

若要允許資源移回原始位置，請切換至「管理讓」，在資源檢視窗上按一下滑鼠右鍵，然後選取「清除移轉限制」。此程序會使用 `crm_resource -U` 指令。資源可以移回其原始位置，也可以保留在目前的位置 (具體取決於資源相粘性)。如需詳細資訊，請參閱 `crm_resource(8)` [157頁] 或《*Configuration 1.0 Explained*》(Configuration 1.0 說明) 中的「*Resource Migration*」(資源移轉) 一章，該文件可從 <http://clusterlabs.org/wiki/Documentation> 取得。

## 4.13 如需更多資訊

<http://clusterlabs.org/>

Pacemaker 首頁，High Availability Extension 隨附的叢集資源管理員。

<http://linux-ha.org>

The High Availability Linux Project 首頁。

<http://clusterlabs.org/wiki/Documentation>

CRM 指令行介面：crm 指令行工具的簡介。

<http://clusterlabs.org/wiki/Documentation>

《*Configuration 1.0 Explained*》(Configuration 1.0 說明): 說明用於設定 Pacemaker 的概念。包含全面詳盡的資訊, 以供參考。

## 從指令行設定叢集資源

如第 4 章 [29頁] 中所述，必須為在叢集中的伺服器上執行的每個資源或應用程式建立叢集資源。叢集資源可包括網站、電子郵件伺服器、資料庫、檔案系統、虛擬機器，以及其他您希望使用者隨時都可以存取的伺服器型應用程式或服務。

您可以使用圖形 HA Management Client 公用程式或 `crm` 指令行公用程式建立資源。本章介紹了幾個 `crm` 公用程式。

### 5.1 指令行工具

安裝後，就會出現幾個工具可用於管理叢集。通常只需要 `crm` 指令。此指令有幾個子指令。執行 `crm help` 可取得所有可用指令的綜覽。該綜覽是包含內嵌式範例的完整說明系統。

`crm` 工具具有管理功能 (子指令為 `resources` 和 `node`)，用於進行組態設定 (`cib`、`configure`)。管理子指令可「立即」生效，但組態最後還需要執行 `commit`。

### 5.2 組態變更除錯

在將變更載入回叢集之前，建議先使用 `ptest` 檢視變更。`ptest` 可顯示要提交的變更將導致之動作的圖表。需要 `graphviz` 套件才能顯示圖表。以下範例是一份記錄，新增了監控作業：

```

# crm
crm(live)# configure
crm(live)configure# show fence-node2
primitive fence-node2 stonith:apcsmart \
    params hostlist="node2"
crm(live)configure# monitor fence-node2 120m:60s
crm(live)configure# show changed
primitive fence-node2 stonith:apcsmart \
    params hostlist="node2" \
    op monitor interval="120m" timeout="60s"
crm(live)configure# pctest
crm(live)configure# commit

```

## 5.3 建立叢集資源

叢集可以使用以下三種類型的 RA (資源代辦)。首先，有舊版 Heartbeat 1 程序檔。High Availability 可以使用 LSB 啟始化程序檔。最後，叢集有它自己的一組 OCF (開放叢集架構) 代辦。此文件重點說明 LSB 程序檔和 OCF 代辦。

若要建立叢集資源，請使用 `crm` 工具。若要將新資源新增至叢集，一般程序如下：

- 1 開啟外圍程序，切換為 `root` 身份。
- 2 執行 `crm` 開啟 `crm` 的內部外圍程序。提示變更為 `crm(live)#`。
- 3 設定原始 IP 位址：

```

crm(live)# configure
crm(live)configure# primitive myIP ocf:heartbeat:IPaddr \
    params ip=127.0.0.99 op monitor intervall=60s

```

以上指令設定名為 `myIP` 的「原始」IP 位址。您需要提供類別 (此處為 `ocf`)、提供者 (`heartbeat`) 和類型 (`IPaddr`)。此外，此原始資源還需要某些參數，例如 IP 位址。必須將位址變更為您的設定。

- 4 顯示並檢閱已進行的變更：

```

crm(live)configure# show

```

若要查看 XML 結構，請使用下列指令：

```

crm(live)configure# show xml

```

## 5 提交變更，使之生效：

```
crm(live)configure# commit
```

### 5.3.1 LSB 啟始化程序檔

所有 LSB 程序檔通常均位於 `/etc/init.d` 目錄中。它們必須執行幾個動作，這些動作至少包含 `start`、`stop`、`restart`、`reload`、`force-reload` 和 `status`，如 [http://www.linux-foundation.org/spec/refspecs/LSB\\_1.3.0/gLSB/gLSB/inisrptact.html](http://www.linux-foundation.org/spec/refspecs/LSB_1.3.0/gLSB/gLSB/inisrptact.html) 中所述。

這些服務的組態尚未標準化。如果要將 LSB 程序檔與 High Availability 搭配使用，請確定您瞭解如何設定各程序檔。您通常可以在 `/usr/share/doc/packages/套件名稱` 中它們各自的套件文件中找到相關文件。

---

#### 注：不要接觸 High Availability 所使用的服務

當服務由 High Availability 使用時，不要藉由其他方式接觸該服務。這表示不應在開機、重新開機時啟動或停止該服務，或進行手動啟動或停止。不過，如果您要檢查服務是否正確設定，請手動將其啟動，但請確保在 High Availability 接管之前將它再次停止。

---

在使用 LSB 資源之前，請確定此資源的組態存在且在所有叢集節點上均相同。該組態不受 High Availability 管理。您必須自行管理。

### 5.3.2 OCF 資源代辦

所有 OCF 代辦皆位於 `/usr/lib/ocf/resource.d/heartbeat/` 中。它們是一些小程序，其功能類似於 LSB 程序檔。不過，始終使用環境變數進行組態設定。所有 OCF 資源代辦都必須至少含有動作 `start`、`stop`、`status`、`monitor` 和 `meta-data`。`meta-data` 動作可取回有關如何設定代辦的資訊。例如，如果您要詳細瞭解 IPaddr 代辦，請使用以下指令：

```
OCF_ROOT=/usr/lib/ocf /usr/lib/ocf/resource.d/heartbeat/IPaddr meta-data
```

輸出為採用簡單 XML 格式的冗長資訊。您可以使用 `ra-api-1.dtd` DTD 驗證輸出。此 XML 格式基本上有三個區段：第一個區段為若干通用描述，第二個區段為所有可用的參數，最後一個區段為此代辦的可用動作。

此輸出用於機器閱讀，可讀性不高。因此，crm工具包含ra指令以取得有關資源代辦的不同資訊：

```
# crm
crm(live)# ra
crm(live)ra#
```

指令 classes 可提供所有類別和提供者的清單：

```
crm(live)ra# classes
stonith
lsb
ocf / lvm2 ocfs2 heartbeat pacemaker
heartbeat
```

若要取得類別 (和提供者) 的所有可用資源代辦的綜覽，請使用 list：

```
crm(live)ra# list ocf
AudibleAlarm      ClusterMon        Delay              Dummy
Filesystem        ICP               IPaddr            IPaddr2
IPsrcaddr         IPv6addr          LVM               LinuxSCSI
MailTo            ManageRAID        ManageVE          Pure-FTPd
Raid1             Route             SAPDatabase       SAPInstance
SendArp           ServerAID         SphinxSearchDaemon Squid
...
```

如需有關資源代辦的詳細資訊，可透過 meta 檢視：

```
crm(live)ra# meta Filesystem ocf heartbeat
Filesystem resource agent (ocf:heartbeat:Filesystem)

Resource script for Filesystem. It manages a Filesystem on a shared storage
medium.

Parameters (* denotes required, [] the default):
...
```

您可以透過按 Q 鍵離開檢視器。組態範例可在第 6 章「設定簡易測試資源」[69頁] 中找到。

## 5.3.3 NFS 伺服器的範例組態

若要設定 NFS 伺服器，需要三種資源：檔案系統資源、drbd 資源和一組 NFS 伺服器及 IP 位址。以下各目顯示如何執行此操作。

## 設定檔案系統資源

filesystem 資源設定為 OCF 原始資源。需要完成根據啟動和停止要求在目錄中裝載和卸載設備這一任務。在本範例中，設備為 /dev/drbd0，要用做裝載點的目錄為 /srv/failover。所使用的檔案系統為 xfs。

在 crm 外圍程序中使用以下指令設定 filesystem 資源：

```
crm(live)# configure
crm(live)configure# primitive filesystem_resource \
    ocf:heartbeat:Filesystem \
    params device=/dev/drbd0 directory=/srv/failover fstype=xfs
```

## 設定 drbd

在開始 drbd High Availability 組態設定之前，請先手動設定 drbd 設備。基本上，此操作為在 /etc/drbd.conf 中設定 drbd 並使其同步。設定 drbd 的確切程序將在 *儲存管理指南* 中加以說明。現在，假設您已設定資源 r0，可於兩個叢集節點上的設備 /dev/drbd0 存取該資源。

drbd 資源為 OCF 主要從屬資源。此資訊可在 drbd RA 中繼資料的描述中找到。不過，更重要的是在中繼資料的 actions 區段中含有 promote 和 demote 動作。它們是主要從屬資源的必要動作，通常不可用於其他資源。

對於 High Availability，主要從屬資源可在不同節點上擁有多個主要資源。甚至可能在同一個節點上既有主要資源又有從屬資源。因此，將使用特定方式設定此資源，使得恰有一個主要資源和一個從屬資源分別在不同的節點上執行。要實現此目的，可使用主要資源的 meta 屬性。主要從屬資源是 High Availability 中的一種特殊類型的複製資源。每個主要資源和每個從屬資源皆計數為一個複製資源。

在 crm 外圍程序中使用以下指令設定主要從屬資源：

```
crm(live)# configure
crm(live)configure# primitive drbd_r0 ocf:heartbeat:drbd params
crm(live)configure# ms drbd_resource drbd_r0 \
    meta clone_max=2 clone_node_max=1 master_max=1 master_node_max=1 notify=true
crm(live)configure# commit
```

## NFS 伺服器與 IP 位址

若要使 NFS 伺服器始終在同一個 IP 位址上可用，請使用其他 IP 位址及機器用於其一般操作的 IP 位址。然後，除系統的 IP 位址之外，還會將此 IP 位址指定給使用中的 NFS 伺服器。

NFS 伺服器與 NFS 伺服器的 IP 位址在同一台機器上應始終處於使用中狀態。在此情況下，啟動順序並不十分重要。甚至可以同時啟動它們。這些是群組資源的一般要求。

在開始 High Availability RA 組態設定之前，請使用 YaST 設定 NFS 伺服器。不要讓系統啟動 NFS 伺服器。只需設定組態檔案。如果要手動執行此操作，請參閱手冊頁 `exports(5)` (`man 5 exports`)。組態檔案為 `/etc/exports`。NFS 伺服器設定為 LSB 資源。

使用 High Availability RA 組態完整設定 IP 位址。不需要在系統中進行其他修改。IP 位址 RA 為 OCF RA。

```
crm(live)# configure
crm(live)configure# primitive nfs_resource lsb:nfsserver
crm(live)configure# primitive ip_resource ocf:heartbeat:IPAddr \
    params ip=10.10.0.1
crm(live)configure# group nfs_group nfs_resource ip_resource
crm(live)configure# commit
crm(live)configure# end
crm(live)# quit
```

## 5.4 建立 STONITH 資源

從 `crm` 角度看，STONITH 設備只是另一個資源。若要建立 STONITH 資源，請執行下列步驟：

- 1 以系統管理員身份執行 `crm` 指令。提示變更為 `crm(live)`。
- 2 使用以下指令取得所有 STONITH 類型清單：

```
crm(live)# ra list stonith
apcmaster                apcsmart                baytech
cyclades                 drac3                   external/drac5
external/hmchttp         external/ibmrsa         external/ibmrsa-telnet
external/ipmi            external/kdumpcheck     external/rackpdu
external/riloe           external/sbd            external/ssh
external/vmware          external/xen0           external/xen0-ha
```

ibmhmc	ipmilan	meatware
null	nw_rpc100s	rcd_serial
rps10	ssh	suicide

- 3 從上述清單中選擇一種 STONITH 類型並檢視可能的選項清單。使用以下指令 (按 Q 鍵可關閉檢視器):

```
crm(live)# ra meta external/ipmi stonith
IPMI STONITH external device (stonith:external/ipmi)

IPMI-based host reset

Parameters (* denotes required, [] the default):
...
```

- 4 使用類別 stonith、您在步驟 3 中選擇的類型，並視需要使用相應參數建立 STONITH 資源，例如:

```
crm(live)# configure
crm(live)configure# primitive my-stonith stonith:external/ipmi \
  meta target-role=Stopped \
  operations my_stonith-operations \
    op monitor start-delay=15 timeout=15 hostlist='' \
      pduip='' community=''
```

## 5.5 設定資源限制

設定所有資源只是工作的一部分。即使叢集瞭解所有必需的資源，可能仍然無法正確地對其進行處理。例如，嘗試在 drbd 從屬節點上裝載檔案系統沒有任何意義 (實際上，此操作將失敗，出現 drbd 錯誤)。若要通知叢集有關這些事項，請定義限制。

在 High Availability 中，有三種不同類型的限制可供使用:

- 位置限制，定義可在哪些節點上執行資源 (在 crm 外圍程序中可使用 location 指令實現)。
- 並存限制，告知叢集哪些資源可以或不可以一併在節點上執行 (colocation)。
- 順序限制，定義動作順序 (order)。

## 5.5.1 位置限制

可以為每個資源多次新增此類限制。將針對提供的資源評估所有 `rsc_location` 限制。下面是一個簡單的範例，可將在名為 `earth` 的節點上執行 ID 為 `fs1-loc` 的資源的可能性提高到 100：

```
crm(live)configure# location fs1-loc fs1 100: earth
```

## 5.5.2 並存限制

`colocation` 指令用於定義應在相同或不同主機上執行的資源。它通常總使用下列順序：

```
crm(live)configure# order rsc1 rsc2
crm(live)configure# colocation rsc2 rsc1
```

只能設定許多 `+INFINITY` 或 `-INFINITY`，定義必須始終或永不在同一個節點上執行的資源。例如，若要始終在同一個主機上執行 ID 分別為 `filesystem_resource` 和 `nfs_group` 的兩個資源，請使用以下限制：

```
crm(live)configure# colocation nfs_on_filesystem inf: nfs_group
filesystem_resource
```

對於主要從屬組態，除本地執行資源之外，還必須瞭解目前節點是否為主要節點。這可使用其他 `to_role` 或 `from_role` 屬性進行檢查。

## 5.5.3 順序限制

有時需要提供服務必須採用的啟動順序。例如，在設備可用於系統之前，不能裝載檔案系統。順序限制可用於在另一個資源符合特定條件(例如啟動、停止或升級為主要資源)的前後啟動或停止服務。在 `crm` 外圍程序中使用以下指令設定順序限制：

```
crm(live)configure# order nfs_after_filesystem mandatory: group_nfs
filesystem_resource
```

## 5.5.4 範例組態的限制

如果沒有其他限制，本章中所用的範例可能無法如預期般作用。所有資源皆必須與 `drbd` 資源的主要資源在同一機器上執行，這是基本要求。另一重要事項

是，在任何其他資源啟動之前，drbd 資源必須是主要資源。若 drbd 不是主要資源，嘗試裝載 drbd 設備必定失敗。必須滿足的限制如下所示：

- 檔案系統必須始終與 drbd 資源的主要資源位於同一個節點上。

```
crm(live)configure# colocation filesystem_on_master inf: \  
    filesystem_resource drbd_resource:Master
```

- NFS 伺服器與 IP 位址必須與檔案系統位於同一個節點上。

```
crm(live)configure# colocation nfs_with_fs inf: \  
    nfs_group filesystem_resource
```

- NFS 伺服器與 IP 位址將在檔案系統完成裝載之後啟動：

```
crm(live)configure# order nfs_second mandatory: \  
    filesystem_resource nfs_group
```

- 必須在將 drbd 資源升級為節點上的主要資源之後，才能在此節點上裝載檔案系統。

```
crm(live)configure# order drbd_first inf: \  
    drbd_resource:promote filesystem_resource
```

## 5.6 指定資源容錯移轉節點

若要判斷資源容錯移轉，請使用 meta 屬性 migration-threshold。例如：

```
crm(live)configure# location r1-node1 r1 100: node1
```

通常，r1 首選在節點 node1 上執行。如果失敗，則會檢查 migration-threshold 並與 failcount 進行比較。若 failcount >= migration-threshold，則將資源移轉至優先設定次佳的節點。

根據 start-failure-is-fatal 選項，啟動失敗會將 failcount 設定為 INFINITY。停止失敗將導致圍籬區隔。如果未定義 STONITH，則根本不會移轉資源。

## 5.7 指定資源錯誤回復節點 (資源相粘性)

僅當管理員重設 `failcount` 或失敗過期時，`rsc` 才會在由於達到失敗次數而產生移轉後進行錯誤回復 (請參閱 `failure-timeout meta` 屬性)。

```
crm resource failcount RSC delete NODE
```

## 5.8 設定資源監控

若要監控資源，有兩種可行方法：使用 `op` 關鍵字定義監控操作或使用 `monitor` 指令。以下範例設定 Apache 資源並每隔 30 分鐘就使用 `op` 關鍵字監控它一次：

```
crm(live)configure# primitive apache apache \  
  params ... \  
  op monitor interval=60s timeout=30s
```

下列指令可達到相同目的：

```
crm(live)configure# primitive apache apache \  
  params ...  
crm(live)configure# monitor apache 60s:30s
```

## 5.9 啟動新的叢集資源

若要啟動新的叢集資源，您需要相應的識別碼。請執行下列步驟：

- 1 以系統管理員身份執行 `crm` 指令。提示變更為 `crm(live)`。
- 2 使用 `status` 指令搜尋各資源。
- 3 使用下列指令啟動資源：

```
crm(live)# resource start ID
```

## 5.10 移除叢集資源

若要移除叢集資源，您需要相應的識別碼。請執行下列步驟：

- 1 以系統管理員身份執行 `crm` 指令。提示變更為 `crm(live)`。

- 2 執行下列指令以取得資源清單：

```
crm(live)# resource status
```

例如，輸出可能如下所示 (`myIP` 為資源的相應識別碼)：

```
myIP      (ocf::IPaddr:heartbeat) ...
```

- 3 刪除具有相應識別碼的資源 (此操作還隱含 `commit` 指令)：

```
crm(live)# configure delete YOUR_ID
```

- 4 提交變更：

```
crm(live)# configure commit
```

## 5.11 設定叢集資源群組

叢集其中一個最常見的元素是需要存放在一起、循序地啟動並以相反順序停止的一組資源。若要簡化此組態，您可以使用群組。以下範例將建立兩個原始資源 (一個 IP 位址和一個電子郵件資源)：

- 1 以系統管理員身份執行 `crm` 指令。提示變更為 `crm(live)`。

- 2 設定原始資源：

```
crm(live)# configure
crm(live)configure# primitive Public-IP ocf:IPaddr:heartbeat \
  params ip=1.2.3.4
crm(live)configure# primitive Email lsb:exim
```

- 3 以正確的順序對原始資源及其各自的識別碼進行分組：

```
crm(live)configure# group shortcut Public-IP Email
```

## 5.12 設定複製品資源

複製最初被認為是啟動 IP 資源的 N 個例項並在整個叢集進行分配以達到負載平衡的一種便利方法。經證明，複製用途多種多樣，包括與 DLM 整合、圍籬區隔子系統和 OCFS2。只要資源代辦支援，就可以複製任何資源。

存在以下幾種複製資源：

### 匿名資源

匿名複製是最簡單的類型。這些資源無論在何處執行，其行為方式均完全相同。因此，每台機器上只能有匿名複製的一個副本處於使用中狀態。

### 多狀態資源

多狀態資源是一種特殊的複製資源。它們允許例項處於兩種作業模式之一。這兩種模式稱為「主要」和「從屬」，但它們的含義可由您隨意指定。唯一的限制是在啟動例項時，必須以從屬狀態出現。

### 5.12.1 建立匿名複製資源

若要建立匿名複製資源，首先要建立原始資源，然後使用 `clone` 指令參考它。請進行下列幾項操作：

- 1 以系統管理員身份執行 `crm` 指令。提示變更為 `crm(live)`。
- 2 設定原始資源，例如：

```
crm(live)# configure
crm(live)configure# primitive Apache lsb:apache
```

- 3 複製原始資源：

```
crm(live)configure# clone apache-clone Apache \
meta globally-unique=false
```

### 5.12.2 建立可設定狀態的/多狀態的複製資源

若要建立可設定狀態的複製資源，首先要建立原始資源，然後建立主要-從屬資源。

- 1 以系統管理員身份執行 `crm` 指令。提示變更為 `crm(live)`。
- 2 設定原始資源。視需要變更間隔：

```
crm(live)# configure
crm(live)configure# primitive myRSC ocf:myCorp:myAppl \
  operations foo \
    op monitor interval=60 \
    op monitor interval=61 role=Master
```

- 3 建立主要從屬資源：

```
crm(live)configure# clone apache-clone Apache \
  meta globally-unique=false
```

## 5.13 移轉叢集資源

雖然資源設定為在遇到硬體或軟體故障時自動容錯移轉 (或移轉) 至叢集中的其他節點，您也可以使用 Linux HA Management Client 或指令行將資源手動移轉至叢集中的另一個節點。

- 1 以系統管理員身份執行 `crm` 指令。提示變更為 `crm(live)`。
- 2 若要將名為 `ipaddress1` 的資源移轉至名為 `node2` 的叢集節點，請輸入以下指令：

```
crm(live)# resource
crm(live)resource# migrate ipaddress1 node2
```

## 5.14 使用陰影組態進行測試

---

**注：**僅限經驗豐富的管理員

雖然概念很簡單，但是仍然建議您在真正需要，並透過 High Availability 取得一定的經驗時，才使用陰影組態。

---

陰影組態用於測試不同的組態案例。如果您已建立幾個陰影組態，則可以逐個進行測試以瞭解變更的效果。

一般程序如下：

1. 使用者啟動 crm 工具。
2. 您切換至 configure 子指令：

```
crm(live)# configure
crm(live)configure#
```

3. 現在即可進行變更。不過，如果您知道這些變更存在風險或者想稍後再套用它們，則可以將它們儲存至新的陰影組態：

```
crm(live)configure# cib new myNewConfig
INFO: myNewConfig shadow CIB created
crm(myNewConfig)configure# commit
```

4. 建立完陰影組態之後，即可進行變更。
5. 若要切換回使用中的叢集組態，請使用以下指令：

```
crm(myNewConfig)configure# cib use
crm(live)configure#
```

## 5.15 如需更多資訊

<http://linux-ha.org>  
High Availability Linux 的首頁

[http://www.clusterlabs.org/mediawiki/images/8/8d/Crm\\_cli.pdf](http://www.clusterlabs.org/mediawiki/images/8/8d/Crm_cli.pdf)  
為您提供 CRM CLI 工具簡介

[http://www.clusterlabs.org/mediawiki/images/f/fb/Configuration\\_Explained.pdf](http://www.clusterlabs.org/mediawiki/images/f/fb/Configuration_Explained.pdf)  
說明 Pacemaker 組態

# 6

## 設定簡易測試資源

在按照第 3 章「使用 YaST 進行安裝及基本設定」[21頁] 中所述安裝和設定叢集，並在瞭解如何使用 GUI 或指令行設定資源之後，本章接下來提供了設定簡易資源 (IP 位址) 的基本範例。本文示範了使用 Linux HA Management Client 或 crm 指令行工具兩種方法來執行此作業。

對於以下範例，我們假設您的叢集至少包含兩個節點。

### 6.1 使用 GUI 設定資源

建立範例叢集資源並將其移轉至其他伺服器，可協助您進行測試以確保叢集正常運作。設定和移轉的簡易資源為 IP 位址。

#### 過程 6.1 建立 IP 位址叢集資源

- 1 啟動 Linux HA Management Client，並依第 4.1 節「Linux HA Management Client」[29頁] 中所述登入叢集。
- 2 在左側窗格中，切換至「資源」檢視窗，然後在右側窗格中選取要修改的群組，並按一下「編輯」。下一個視窗會顯示基本的群組參數，以及已為該資源定義的中繼屬性與原始資源。
- 3 按一下「原始資源」索引標籤，然後按一下「新增」。
- 4 在下一個對話方塊中，設定以下參數以將 IP 位址新增為群組的子資源：
  - 4a 輸入唯一的 ID，例如 myIP。

- 4b 在「類別」清單中，選取「ocf」做為資源代辦類別。
- 4c 對於 OCF 資源代辦的「提供者」，選取「heartbeat」。
- 4d 在「類型」清單中，選取「IPaddr」做為資源代辦。
- 4e 按「下一步」。
- 4f 在「例項屬性」索引標籤中，選取「IP」項目並按一下「編輯」(或在「IP」項目上連按兩下)。
- 4g 對於「值」，請輸入所需的 IP 位址 (例如 10.10.0.1)，然後按一下「確定」。
- 4h 「新增」一個新的例項屬性，並將 nic 指定為「名稱」，將 eth0 指定為「值」，然後按一下「確定」。

名稱與值具體取決於您的硬體組態以及安裝 High Availability Extension 軟體期間為媒體組態所選選項。

- 5 按需要設定了所有參數後，請按一下「確定」以完成該資源的組態設定。組態對話方塊會關閉，同時主視窗會顯示修改的資源。

若要使用 Linux HA Management Client 啟動資源，請選取左側窗格中的「管理」。在右側窗格中，在資源上按一下滑鼠右鍵，然後選取「啟動」(或從工具列中將資源啟動)。

若要將 IP 位址資源移轉至其他節點 (saturn)，請繼續執行以下步驟：

#### 過程 6.2 將資源移轉至其他節點

- 1 切換至左側窗格中的「管理」檢視窗，然後在右側窗格中的 IP 位址資源上按一下滑鼠右鍵，並選取「移轉資源」。
- 2 在新視窗中，從「至節點」下拉式清單中選取 saturn，以將所選資源移至節點 saturn。
- 3 若只想暫時移轉資源，請啟用「持續時間」，並輸入資源移轉至新節點後應保留的時間。

- 4 按一下「確定」以確認移轉。

## 6.2 資源的手動組態設定

資源是電腦提供之任何類型的服務。當由 LSB 程序檔、OCF 程序檔或舊版 Heartbeat 1 資源等 RA (資源代辦) 控制資源時，High Availability 才會識別這些資源。可以使用 `crm` 指令或以 `resources` 區段的 CIB (叢集資訊庫) 中的 XML 設定所有資源。如需可用資源的綜覽，請參閱第 18 章「*HA OCF 代辦*」[191頁]。

若要新增 IP 位址 10.10.0.1 做為目前組態的資源，請使用 `crm` 指令：

### 過程 6.3 建立 IP 位址叢集資源

- 1 以系統管理員身份執行 `crm` 指令。提示變更為 `crm(live)`。
- 2 切換至 `configure` 子指令：

```
crm(live)# configure
```

- 3 建立 IP 位址資源：

```
crm(live)configure# resource  
primitive myIP ocf:heartbeat:IPaddr params ip=10.10.0.1
```

---

#### 注

使用 High Availability 設定資源時，相同的資源不應由 `init` 啟始化。High Availability 負責所有服務的啟動或停止動作。

---

若組態設定成功，`crm_mon` 中會顯示一個新的資源，該資源將在叢集的一個隨機節點上啟動。

若要將資源移轉至其他節點，請執行以下步驟：

### 過程 6.4 將資源移轉至其他節點

- 1 啟動外圍程序，並成為使用者 `root`。

**2** 將資源 myip 移轉至節點 saturn:

```
crm resource migrate myIP saturn
```

## 新增或修改資源代辦

應由叢集管理的所有任務皆必須做為資源來使用。有兩個主要群組應加以區別：資源代辦與STONITH代辦。對於這兩種類別，您都可以新增自己的代辦，以延伸叢集的功能來滿足自己的需要。

### 7.1 STONITH 代辦

叢集有時會偵測到其中一個節點的行為不正常，需要將其移除。這稱為「圍籬區隔」，通常透過STONITH資源來執行。所有STONITH資源皆存放在每個節點的 `/usr/lib/stonith/plugins` 中。

---

**警告：不支援 SSH 和 STONITH**

我們無法得知 SSH 對其他系統問題會做出什麼反應。因此，不支援將 SSH 和 STONITH 代辦用於生產環境。

---

若要取得目前所有可用 STONITH 設備的清單 (從軟體角度)，請使用 `stonith -L` 指令。

遺憾的是，目前還沒有關於撰寫 STONITH 代辦的文件。若要撰寫新的 STONITH 代辦，請參閱 `heartbeat-common` 套件原始碼中提供的範例。

## 7.2 撰寫 OCF 資源代辦

所有 OCF 資源代辦皆存放於 `/usr/lib/ocf/resource.d/` 中；如需詳細資訊，請參閱第 17.1 節「受支援的資源代辦類別」[183頁]。若要避免名稱衝突，請在建立新的資源代辦時建立不同的子目錄。例如，若您擁有資源群組 `kitchen` (內含資源 `coffee_machine`)，請將此資源新增至 `/usr/lib/ocf/resource.d/kitchen/` 目錄。若要存取此資源代辦，請執行 `crm` 指令：

```
configure
primitive coffee_1 ocf:coffee_machine:kitchen ...
```

實作自己的 OCF 資源代辦時，請提供針對此代辦的幾個動作。如需關於撰寫 OCF 資源代辦的更多詳細資料，請造訪 <http://www.linux-ha.org/OCFResourceAgent>。有關 High Availability 2 的幾個概念的特殊資訊，請參閱第 1 章「概念綜覽」[3頁]。

## 圍籬區隔與 STONITH

在 HA (High Availability) 的電腦叢集中，圍籬區隔是一個極其重要的概念。叢集有時會偵測到其中一個節點的行為不正常，需要將其移除。這稱為「圍籬區隔」，通常透過 STONITH 資源來執行。可將圍籬區隔定義為讓 HA 叢集處於已知狀態一種方法。

叢集中的每個資源都附加有狀態，例如：「資源 r1 已在節點 1 上啟動」。在 HA 叢集中，這樣的狀態隱含「資源 r1 已在除節點 1 之外的所有節點上停止」的含義，因為 HA 叢集必須確保每個資源最多只能在一個節點上啟動。每個節點都必須報告資源發生的每一項變更。因此，叢集狀態是資源狀態和節點狀態的集合。

如果無法明確確定某些節點或資源的狀態 (不論出於何種原因)，就會出現圍籬區隔。即使在叢集不瞭解某些節點上發生什麼狀況的情況下，圍籬區隔也可以確保該節點不會執行任何重要資源。

### 8.1 圍籬區隔的類別

有兩類圍籬區隔：資源層級圍籬區隔和節點層級圍籬區隔。後者為本章的主要主題。

#### 資源層級圍籬區隔

如果使用資源層級圍籬區隔，叢集可以確保節點無法存取一或多個資源。一個典型範例為 SAN，在此範例中，圍籬區隔作業會變更 SAN 交換器上的規則，以拒絕從節點進行存取。

資源層級圍籬區隔可利用要保護之資源所依賴的一般資源來實現。只需拒絕在此節點上啟動此資源，所依賴的資源亦不會在相同節點上執行。

#### 節點層級圍籬區隔

節點層級圍籬區隔確保節點根本不會執行任何資源。這通常以一種極其簡單但卻野蠻的方式實現：只需使用電源交換器重設節點。最終可能也必須這麼做，因為節點可能根本沒有回應。

## 8.2 節點層級圍籬區隔

在 SUSE® Linux Enterprise High Availability Extension 中，圍籬區隔實作為 STONITH (Shoot The Other Node in the Head)。它提供節點層級圍籬區隔。High Availability Extension 包含 `stonith` 指令行工具，此為一個可擴充介面，用於從遠端關閉叢集中的節點。如需可用選項的綜覽，請執行 `stonith --help`，如需詳細資訊，請參閱 `stonith` 的 `man` 頁面。

### 8.2.1 STONITH 設備

若要使用節點層級圍籬區隔，首先需要擁有一個圍籬區隔設備。若要取得 High Availability Extension 支援的 STONITH 設備清單，請在任何節點上以 `root` 身分執行以下指令：

```
stonith -L
```

可將 STONITH 設備分成以下類別：

#### 配電裝置 (PDU)

在管理重要網路、伺服器和資料中心設備的電源容量和功能方面，配電裝置扮演著至關重要的角色。它們可以提供已連接設備的遠端負載監控功能，並可進行個別插座電源控制以實現遠端電能回收。

#### 不斷電供電系統 (UPS)

市電電源中斷時，透過獨立來源供電的不斷電供電系統可為連接的設備提供應急電能。

#### 刀鋒電源控制設備

如果您在一組刀鋒上執行叢集，則刀鋒機箱中的電源控制設備是圍籬區隔唯一的候選設備。當然，此設備必須能夠管理單一刀鋒電腦。

### 無人職守設備

無人職守設備 (IBM RSA、HP iLO、Dell DRAC) 越來越受歡迎，將來它們甚至會成為現貨電腦的標準設備。不過，它們略遜於 UPS 設備，因為它們與其主機 (叢集節點) 共享電源。如果節點仍保持斷電狀態，讓設備進行控制亦無用。此時，若 CRM 嘗試圍籬區隔該節點將徒勞無功，且這種狀況將永遠持續，因為所有其他資源作業都要等待該圍籬區隔/STONITH 作業成功完成。

### 測試設備

測試設備專用於測試用途。在硬體上，對測試設備的要求通常比較寬松。一旦叢集進入實際生產環境，它們必須由真正的圍籬區隔設備取代。

是否選擇 STONITH 設備主要取決於您的預算及所使用的硬體類型。

## 8.2.2 STONITH 實作

SUSE® Linux Enterprise High Availability Extension 的 STONITH 實作由兩個元件組成：

### stonithd

stonithd 是一個精靈，可由本地程序或透過網路進行存取。它接受圍籬區隔作業的相應指令：重設、關機和開機。它還可以檢查圍籬區隔設備的狀態。

stonithd 精靈在 CRM HA 叢集中的每個節點上執行。在 DC 節點上執行的 stonithd 例項接收來自 CRM 的圍籬區隔要求。須由此程式及其他 stonithd 程式決定是否執行所需的圍籬區隔作業。

### STONITH 外掛程式

對於每個受支援的圍籬區隔設備，都有一個可控制該設備的 STONITH 外掛程式。STONITH 外掛程式是圍籬區隔設備的介面。所有 STONITH 外掛程式皆存放在每個節點上的 `/usr/lib/stonith/plugins` 中。所有 STONITH 外掛程式對於 stonithd 而言看起來都是一樣的，但在反映圍籬區隔設備性質的其他方面則大不相同。

某些外掛程式支援多個設備。典型範例為 `ipmilan` (或 `external/ipmi`)，它實作 IPMI 通訊協定並可控制支援此通訊協定的任何設備。

## 8.3 STONITH 組態

若要設定圍籬區隔，您需要設定一或多個 STONITH 資源——stonithd 精靈不需要任何組態設定。所有組態皆儲存在 CIB 中。STONITH 資源即類別為 `stonith` 的資源 (請參閱第 17.1 節「受支援的資源代辦類別」[183頁])。STONITH 資源是 STONITH 外掛程式在 CIB 中的表示。除圍籬區隔作業之外，還可對 STONITH 資源執行啟動、停止和監控作業，就如同任何其他資源一樣。在這種情況下，啟動和停止 STONITH 資源就表示啟用和停用 STONITH。因此，啟動和停止只是管理作業，不能轉換為圍籬區隔設備本身的任何作業。不過，監控會轉換為設備狀態。

可以對 STONITH 資源進行設定，就如同任何其他資源一樣。如需有關設定資源的詳細資訊，請參閱第 4.3 節「建立 STONITH 資源」[35頁] 或第 5.4 節「建立 STONITH 資源」[60頁]。

參數 (屬性) 清單取決於各 STONITH 類型。若要檢視特定設備的參數清單，請使用 `stonith` 指令：

```
stonith -t stonith-device-type -n
```

例如，若要檢視 `ibmhmc` 設備類型的參數，請輸入以下指令：

```
stonith -t ibmhmc -n
```

若要取得設備的簡短說明文字，請使用 `-h` 選項：

```
stonith -t stonith-device-type -h
```

### 8.3.1 範例 STONITH 資源組態

下面提供了使用 `crm` 指令行工具的語法撰寫的一些範例組態。若要套用它們，請將範例放入文字檔中 (例如，`sample.txt`)，然後執行以下指令：

```
crm < sample.txt
```

如需有關使用 `crm` 指令行工具設定資源的詳細資訊，請參閱第 5 章「從指令行設定叢集資源」[55頁]。

---

## 警告：測試組態

下面提供的一些範例僅用於演示和測試用途。切勿在現實叢集案例中使用任何測試組態範例。

---

### 範例 8.1 測試組態

```
configure
primitive st-null stonith:null \
params hostlist="node1 node2"
clone fencing st-null
commit
```

### 範例 8.2 測試組態

代用組態：

```
configure
primitive st-node1 stonith:null \
params hostlist="node1"
primitive st-node2 stonith:null \
params hostlist="node2"
location l-st-node1 st-node1 -inf: node1
location l-st-node2 st-node2 -inf: node2
commit
```

如果關注叢集軟體，則此組態範例極其適用。與現實組態的唯一區別在於不發生任何圍籬區隔作業。

### 範例 8.3 測試組態

下面的 external/ssh 組態是更貼近實際但仍然僅用於測試的範例：

```
configure
primitive st-ssh stonith:external/ssh \
params hostlist="node1 node2"
clone fencing st-ssh
commit
```

此組態也可重設節點。該組態與第一個組態非常相似，它具備空 STONITH 設備的功能。在此範例中，將使用複製。它們是 CRM/Pacemaker 功能。基本上，複製是一種捷徑：不需要定義 n 個名稱不同的相同資源，只需一個複製的資源就已足夠。如果 STONITH 設備可從所有節點存取，顯然最常用的複製為使用 STONITH 資源。

#### 範例 8.4 IBM RSA 無人職守設備組態

實際設備組態並沒有明顯差異，雖然某些設備可能需要更多屬性。可按如下所示設定 IBM RSA 無人職守設備：

```
configure
primitive st-ibmrsa-1 stonith:external/ibmrsa-telnet \
params nodename=node1 ipaddr=192.168.0.101 \
userid=USERID passwd=PASSWORD
primitive st-ibmrsa-2 stonith:external/ibmrsa-telnet \
params nodename=node2 ipaddr=192.168.0.102 \
userid=USERID passwd=PASSWORD
location l-st-node1 st-ibmrsa-1 -inf: node1
location l-st-node2 st-ibmrsa-2 -inf: node2
commit
```

在此範例中，由於下列原因，使用了位置限制：STONITH 作業失敗的可能性始終存在。因此，在做為執行者的節點上的 STONITH 作業也不可靠。如果重設節點，它就不能傳送關於圍籬區隔作業結果的通知。想傳送通知的唯一方法是，假設作業即將成功，然後提前傳送通知。但如果作業失敗，就會陷入麻煩。因此，一般來說 stonithd 拒絕停止其主機。

### 範例 8.5 UPS 圍籬區隔設備組態

UPS 類型圍籬區隔設備的組態類似於上述範例，詳細資料留給讀者做為練習之用。所有 UPS 設備皆使用相同技術進行圍籬區隔，但存取設備本身的方法各不相同。舊式 UPS 設備 (曾視為專業 UPS 設備) 通常只有一個序列埠，一般使用專用序列纜線以 1200 鮑率進行連接。許多新型 UPS 設備仍然有一個序列埠，但它們通常還具有一個 USB 介面或一個乙太網路介面。可以使用的連線類型取決於外掛程式所支援的類型。

例如，使用 `stonith -t stonith 設備類型 -n` 指令比較 `apcmaster` 與 `apcsmart` 設備：

```
stonith -t apcmaster -h
```

傳回下列資訊：

```
STONITH Device: apcmaster - APC MasterSwitch (via telnet)
NOTE: The APC MasterSwitch accepts only one (telnet)
connection/session a time. When one session is active,
subsequent attempts to connect to the MasterSwitch will fail.
For more information see http://www.apc.com/
List of valid parameter names for apcmaster STONITH device:
ipaddr
login
password
```

使用

```
stonith -t apcsmart -h
```

您將取得下列輸出：

```
STONITH Device: apcsmart - APC Smart UPS
(via serial port - NOT USB!).
Works with higher-end APC UPSes, like
Back-UPS Pro, Smart-UPS, Matrix-UPS, etc.
(Smart-UPS may have to be >= Smart-UPS 700?).
See http://www.networkupstools.org/protocols/apcsmart.html
for protocol compatibility details.
For more information see http://www.apc.com/
List of valid parameter names for apcsmart STONITH device:
ttydev
hostlist
```

第一個外掛程式支援具有網路埠和 Telnet 通訊協定的 APC UPS。第二個外掛程式透過眾多 APC UPS 產品線皆支援的序列線使用 APC SMART 通訊協定。

## 8.3.2 限制與複製

在第 8.3.1 節「範例 STONITH 資源組態」[78頁] 中，您瞭解到設定 STONITH 資源有幾種方法：使用限制或複製，或同時使用兩者。選擇使用何種建構進行組態設定取決於幾項因素(圍籬區隔設備的性質、設備管理的主機數目、叢集節點數目)，最後(但並非最不重要)還取決於個人優先設定。

簡言之：如果可以放心地將複製與組態搭配使用，且如果它們確實可以減少組態設定工作，則使用複製的 STONITH 資源。

## 8.4 監控圍籬區隔設備

就如同任何其他資源一樣，STONITH 類別代辦也支援用於檢查狀態的監控作業。

---

### 注：監控 STONITH 資源

強烈建議您監控 STONITH 資源。應定期且謹慎地對它們進行監控。

---

圍籬區隔設備是 HA 叢集必不可少的一部分，但需要對它們執行的操作卻是越少越好。眾所周知，電源管理設備在通訊端尤為脆弱。如果線路上的廣播流量過多，某些設備就會停止工作。有些設備無法處理每分鐘十個左右的連線。如果兩個用戶端同時嘗試連接，有些設備就會發生混淆或效能下降。大部分設備不能同時處理多個工作階段。

因此，在大多數情況下，每隔幾個小時檢查一次圍籬區隔設備就已足夠。在這幾個小時內需要執行圍籬區隔作業以及電源交換器失效的可能性通常很低。

如需如何設定監控操作的詳細資訊，請參閱新增或修改中繼屬性與例項屬性 [34頁] (適用於 GUI 方法)，或參閱第 5.8 節「設定資源監控」[64頁] (適用於指令行方法)。

## 8.5 特殊圍籬區隔設備

除處理實際設備的外掛程式之外，某些 STONITH 外掛程式有點與眾不同，需特別留意。

external/kdumpcheck

有時，必須取得核心磁心傾印。可以使用此外掛程式檢查傾印是否正在進行。如果傾印正在進行，則此外掛程式將傳回 `true`，表示已圍籬區隔節點，只要該節點此時無法執行任何資源，就表示此狀況。kdumpcheck 通常與另一個實際的圍籬區隔設備搭配使用。如需詳細資料，請參閱 `/usr/share/doc/packages/heartbeat/stonith/README_kdumpcheck.txt`。

external/sbd

這是一個自我圍籬區隔設備。它會對可插入到共享磁碟中的所謂「毒藥丸」做出反應。在共享儲存連線中斷時，它還會使節點提交自行中止。請參閱 [http://www.linux-ha.org/SBD\\_Fencing](http://www.linux-ha.org/SBD_Fencing)，以取得詳細資料。

meatware

meatware 需要人員協助才能操作。呼叫 meatware 時，它會記錄 `CRIT` 嚴重程度訊息，該訊息會在節點的主控台上顯示。然後，操作員需要確保該節點已關閉，並發出 `meatclient(8)` 指令。這將告訴 meatware，它可以通知叢集可以將節點視為已關閉。如需詳細資訊，請參閱 `/usr/share/doc/packages/heartbeat/stonith/README.meatware`。

null

這是一個虛構的設備，用於各種測試案例。它總是表現為且聲稱自己關閉了一個節點，但從來不會執行任何動作。除非您完全瞭解所執行的操作，否則切勿使用它。

suicide

這是一個僅限軟體的設備，它可以使用 `reboot` 指令重新開機執行它所在的節點。這需要由節點的作業系統執行動作，在某些情況下可能會失敗。因此，應盡可能避免使用此設備 (但可以在單個節點的叢集上使用)。

suicide 和 null 是「do not shoot my host」(不要關閉我的主機) 規則唯一的例外。

## 8.6 如需更多資訊

`/usr/share/doc/packages/heartbeat/stonith/`

在已安裝的系統中，此目錄存放眾多 STONITH 外掛程式和設備的讀我檔案。

<http://linux-ha.org/STONITH>

關於 STONITH 的資訊位於 The High Availability Linux Project 的首頁上。

<http://linux-ha.org/fencing>

關於圍籬區隔的資訊位於 The High Availability Linux Project 的首頁上。

<http://linux-ha.org/ConfiguringStonithPlugins>

關於 STONITH 外掛程式的資訊位於 The High Availability Linux Project 的首頁上。

<http://linux-ha.org/CIB/Idioms>

關於 STONITH 的資訊位於 The High Availability Linux Project 的首頁上。

<http://clusterlabs.org/wiki/Documentation>, *Configuration 1.0 Explained*

說明用於設定 Pacemaker 的概念。包含全面詳盡的資訊，以供參考。

[http://techthoughts.typepad.com/managing\\_computers/2007/10/split-brain-quo.html](http://techthoughts.typepad.com/managing_computers/2007/10/split-brain-quo.html)

說明 HA 叢集中的電腦分裂、法定數目和圍籬區隔之概念的文章。

# Linux Virtual Server 的負載平衡

# 9

Linux Virtual Server (LVS) 的目標是提供一個基本架構，將網路連線導向至共享工作負載的多部伺服器。Linux Virtual Server 是一組伺服器叢集 (一或多個負載平衡器與幾部執行服務的實際伺服器)，對外部用戶端而言則是一個大型的高速伺服器。這種表面上的單個伺服器稱為*虛擬伺服器*。做為進階的負載平衡解決方案，Linux Virtual Server 可用於構建高度可擴充性和可用性的網路服務，例如 Web、快取、郵件、FTP、媒體和 VoIP 等服務。

實際的伺服器與負載平衡器之間可透過高速 LAN 或地理位置分散的 WAN 來連接。負載平衡器可以向不同的伺服器發送要求，使叢集的多個平行服務顯示為單一 IP 位址上的一個虛擬服務，這樣發送請求作業便可使用 IP 負載平衡技術或應用程式層級負載平衡技術。以透明方式在叢集中新增或移除節點可以實現系統的延展性。透過偵測節點或精靈故障，然後重新正確地設定系統可以提供高可用性。

## 9.1 概念綜覽

LVS 包含兩個主要元件：

核心代碼：ip\_vs (或 IPVS)

執行已修補為包含 IPVS 代碼之 Linux 核心的節點稱為 *Director*。在 Director 上執行的 IPVS 代碼是 LVS 的必要特性。

用戶端會連接到用於將封包轉遞至實際伺服器的 Director。Director 是一個四層路由器，包含一組修改過、可使 LVS 運作的路由規則 (例如，連線不在

Director 上產生或終止，Director 不傳送通知)。Director 加上實際的伺服器就組成了虛擬伺服器，其對用戶端而言則是一部機器。

轉遞的方式有多種，不同的方式決定了 Director 將用戶端的封包傳送至實際伺服器的方式。決定要使用哪一部實際的伺服器來處理用戶端要求的新連線，可透過不同的演算法來實作，這些演算法是以模組的方式提供，可根據特定需求進行調整。從用戶端接收到連線請求後，Director 會依據排程為該用戶端指定實際的伺服器。排程器是 IPVS 核心代碼的一部分，用於決定將獲取下一個新連線的實際伺服器。

使用者空間控制器：ipvsadm

ipvsadm 套件提供的 ipvsadm 是一個使用者介面，可用於管理 Linux Virtual Server。例如，您可以設定用來處理服務的規則、處理容錯移轉或設定排程器類型。

可以在指令行中 (或在 rc 檔案中) 使用 ipvsadm 設定以下項目：

- Director 導向的服務/伺服器 (例如，http 導向至所有實際伺服器，而 ftp 僅導向至其中一部實際伺服器)
- 各實際伺服器所占的權重 (在某些伺服器比其他伺服器速度更快的情況下使用)
- 排程演算法

還可以使用 ipvsadm 執行以下任務：

- 新增服務
- 關閉服務
- 刪除服務

## 9.2 High Availability

若要建構高可用性的 Linux Virtual Server 叢集，可以使用本軟體的幾項內建功能。一般情況下，在負載平衡器上會執行服務監控精靈，定期檢查伺服器的狀態。如果在指定時間內，某部伺服器對服務存取要求或 ICMP ECHO\_REQUEST 沒有回應，則服務監控器就會認為該伺服器已停止執行，將其從負載平衡器中的可用伺服器清單中移除。如此，新的要求將不再傳送至此停止的伺服器。當

服務監控器偵測到該停止的伺服器已恢復正常並可重新工作時，會重新將該伺服器新增到可用伺服器清單中。因此，負載平衡器可以自動對服務精靈或伺服器的故障進行遮罩。

此外，管理員也可以使用系統工具新增新的伺服器以提高系統輸送量，或移除伺服器以維護系統，而不需要關閉整個系統服務。

為了防止負載平衡器成為整個系統的單一故障點，須設定一或多個負載平衡器的備份。兩個活動訊號精靈分別在主要負載平衡器和備份負載平衡器上執行。這兩個精靈會透過序列線和/或網路介面定期相互傳送「I'm alive」(我存在)的活動訊號訊息。如果在指定時間內，備份負載平衡器的活動訊號精靈無法監聽到主要負載平衡器的活動訊號訊息，便會接管虛擬 IP 位址以提供負載平衡服務。

如果發生故障的負載平衡器恢復正常，則可能會產生兩種結果：其自動成為備份負載平衡器，另一種結果是使用中的負載平衡器會釋放 VIP 位址，以便讓恢復正常的負載平衡器接管 VIP 位址，重新成為主要負載平衡器。主要負載平衡器瞭解各連線的狀態，也就是說它知道連線應轉遞至哪部伺服器。若備份負載平衡器接管了工作卻不知道該連線資訊，則用戶端必須重新傳送其要求才能存取服務。為了使負載平衡器的容錯移轉對用戶端應用程式透明化，IPVS 中提供了連線同步功能：主要 IPVS 負載平衡器可透過 UDP 多路廣播將連線資訊同步到備份負載平衡器中。如果備份負載平衡器在主要負載平衡器失敗後接管了工作，備份負載平衡器將會瞭解大多數連線的狀態，這樣基本上所有連線便可繼續透過備份負載平衡器存取服務。

## 9.3 如需更多資訊

如需有關 Linux Virtual Server 的詳細資訊，請參閱專案首頁 <http://www.linuxvirtualserver.org/>。



## 網路設備 Bonding

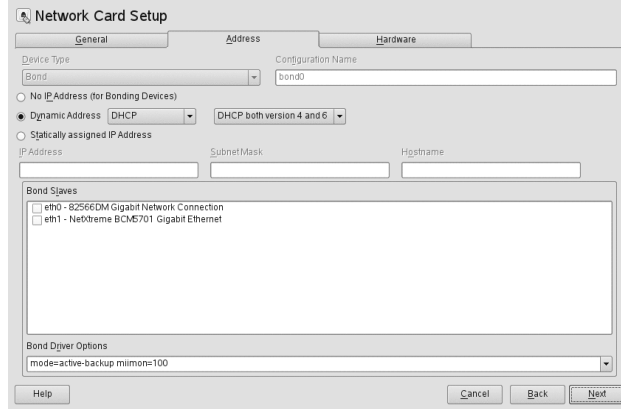
對於許多系統而言，實作的網路連線除了需要符合一般乙太網路設備的標準資料安全性或可用性要求之外，還需要符合其他要求。在這些情況下，數個乙太網路設備可以結集成單一的 bonding 設備。

bonding 設備的組態是透過 bonding 模組選項來設定，而其行為由 bonding 設備的模式決定。該模式預設為 `mode=active-backup`，這表示如果使用中的從屬設備失敗，另一個從屬設備將變成使用中狀態。

使用 OpenAIS 時，bonding 設備不受叢集軟體的管理。因此，必須在可能需要存取 bonding 設備的每個叢集節點上設定該設備。

若要設定 bonding 設備，請執行以下程序：

- 1 執行「YaST」>「網路設備」>「網路設定」。
- 2 使用「新增」，然後將「設備類型」變更為「Bond」。按「下一步」繼續。



3 選取為 bonding 設備指定 IP 位址的方法。有三種方法可供您選擇：

- 無 IP 位址
- 動態位址 (透過 DHCP 或 Zeroconf)
- 靜態指定的 IP 位址

請使用適合您環境的方法。若 OpenAIS 管理虛擬 IP 位址，請選取「靜態指定的 IP 位址」，然後在介面指定一個基本 IP 位址。

- 4 透過啟用相關「Bond 從屬」前面的核取方塊，選取需要加入 bond 的乙太網路設備。
- 5 編輯「Bond 驅動程式選項」。可用模式如下：

balance-rr  
提供負載平衡和容錯。

active-backup  
提供容錯

balance-xor  
提供負載平衡和容錯。

broadcast  
提供容錯

### 802.3ad

提供動態連結聚總 (若連接的交換器支援)。

### balance-tlb

提供外送流量的負載平衡。

### balance-alb

提供內送和外送流量的負載平衡 (若所用的網路設備允許修改使用中網路設備的硬體位址)。

- 6 確認參數 `miimon=100` 已新增至「*Bond* 驅動程式選項」。若沒有此參數，就無法定期檢查資料的完整性。
- 7 按「下一步」，然後按一下「確定」離開 YaST 以建立設備。

「*Linux Ethernet Bonding Driver HOWTO*」(Linux 乙太網路 Bonding 驅動程式 HOWTO) 對所有模式以及許多其他選項做了詳細說明，若您安裝了 `kernel-source` 套件，便可在 `/usr/src/linux/Documentation/networking/bonding.txt` 中找到該資訊。



# 將叢集更新至 SUSE Linux Enterprise 11

# 11

若您現有的叢集是基於 SUSE® Linux Enterprise Server 10 SP2，則可以將其更新，以便透過 High Availability Extension 在 SUSE® Linux Enterprise Server 11 上執行。出於移轉目的，所有叢集節點都必須處於離線狀態，且叢集必須做為一個整體移轉 — 系統不支援混合使用 SUSE Linux Enterprise Server 10/SUSE Linux Enterprise Server 11 叢集。

為了方便起見，SUSE® Linux Enterprise High Availability Extension 中提供了一個 `hb2openais.sh` 程序檔，當您將資料從 Heartbeat 移至 OpenAIS 叢集堆疊期間，可用它來轉換資料。此程序檔可剖析儲存在 `/etc/ha.d/ha.cf` 中的組態，並產生適用於 OpenAIS 叢集堆疊的新組態檔案。此外，它還能調整 CIB 以符合 OpenAIS 慣例、轉換 OCFS2 檔案系統，以及用 cLVM 取代 EVMS。

若要將叢集從 SUSE Linux Enterprise Server 10 SP2 成功移轉至 SUSE Linux Enterprise Server 11，您需要執行以下步驟：

1. 準備 SUSE Linux Enterprise Server 10 SP2 叢集 [94頁]
2. 更新至 SUSE Linux Enterprise 11 [95頁]
3. 測試轉換 [95頁]
4. 轉換資料 [96頁]

成功完成轉換後，可以讓更新後的叢集重新恢復線上狀態。

---

注：更新後回復

在更新至 SUSE Linux Enterprise Server 11 後，就不能重新回復到 SUSE Linux Enterprise Server 10。

---

## 11.1 準備與備份

將叢集更新至產品的最新版本並轉換相應資料之前，您需要準備目前的叢集。

**過程 11.1** 準備 *SUSE Linux Enterprise Server 10 SP2* 叢集

- 1 登入叢集。
- 2 檢閱 Heartbeat 組態檔案 `/etc/ha.d/ha.cf`，並檢查所有通訊媒體是否支援多路廣播。
- 3 確認以下檔案在所有節點上都相同：`/etc/ha.d/ha.cf` 和 `/var/lib/heartbeat/crm/cib.xml`。
- 4 對每個節點執行 `rheartbeat stop`，使所有節點都處於離線狀態。
- 5 在更新至最新版本之前，除了需要依建議備份一般系統之外，還需備份以下檔案，因為在更新至 SUSE Linux Enterprise Server 11 後需要它們來執行轉換程序檔：
  - `/var/lib/heartbeat/crm/cib.xml`
  - `/var/lib/heartbeat/hostcache`
  - `/etc/ha.d/ha.cf`
  - `/etc/logd.cf`

## 11.2 更新/安裝

準備好叢集並備份了檔案之後，可以開始將叢集節點更新至產品的最新版本。您也可以不執行更新，而是在叢集節點上安裝 SUSE Linux Enterprise 11。

### 過程 11.2 更新至 SUSE Linux Enterprise 11

- 1 在所有叢集節點上執行從 SUSE Linux Enterprise Server 10 SP2 至 SUSE Linux Enterprise Server 11 的更新。如需如何更新產品的相關資訊，請參閱《SUSE Linux Enterprise Server 11 部署指南》中的「更新 SUSE Linux Enterprise」一章。

反過來，您也可以在所有叢集節點上安裝 SUSE Linux Enterprise Server 11。

- 2 在所有叢集節點上，於 SUSE Linux Enterprise Server 的基礎上將 SUSE Linux Enterprise High Availability Extension 11 做為附加產品進行安裝。如需詳細資訊，請參閱第 3.1 節「安裝 High Availability Extension」[21頁]。

## 11.3 資料轉換

安裝 SUSE Linux Enterprise Server 11 和 High Availability Extension 之後，便可開始轉換資料。High Availability Extension 隨附的轉換程序檔已經過周密設定，但它並不能在完全自動的模式下完成所有設定。它會警告您它所進行的變更，但仍需要您的互動和決定。您需要瞭解叢集的詳細情況，因為最終是由您來確認這些變更是否有意義。轉換程序檔位於 `/usr/lib/heartbeat` 中 (若您使用的是 64 位元系統，則位於 `/usr/lib64/heartbeat` 中)。

---

### 注：執行測試回合

為了讓您熟悉轉換程序，我們強烈建議您先進行轉換測試 (不做任何變更)。您可以使用相同的測試目錄來執行重複的測試回合，但只需複製一次檔案。

---

### 過程 11.3 測試轉換

- 1 在其中一個節點上，建立測試目錄並將備份檔案複製到測試目錄中：

```
$ mkdir /tmp/hb2openais-testdir
$ cp /etc/ha.d/ha.cf /tmp/hb2openais-testdir
```

```
$ cp /var/lib/heartbeat/hostcache /tmp/hb2openais-testdir
$ cp /etc/logd.cf /tmp/hb2openais-testdir
$ sudo cp /var/lib/heartbeat/crm/cib.xml /tmp/hb2openais-testdir
```

## 2 使用以下指令開始進行測試回合

```
$ /usr/lib/heartbeat/hb2openais.sh -T /tmp/hb2openais-testdir -U
```

若您使用的是 64 位元系統，請使用以下指令執行測試：

```
$ /usr/lib64/heartbeat/hb2openais.sh -T /tmp/hb2openais-testdir -U
```

## 3 讀取並驗證產生的 openais.conf 和 cib-out.xml 檔案：

```
$ cd /tmp/hb2openais-testdir
$ less openais.conf
$ crm_verify -V -x cib-out.xml
```

如需轉換階段的詳細資訊，請參閱所安裝之 High Availability Extension 中的 `/usr/share/doc/packages/pacemaker/README.hb2openais`。

### 過程 11.4 轉換資料

執行測試回合並檢查輸出之後，便可以開始轉換資料。您只需在一個節點上執行轉換。主叢集組態(CIB)會自動複寫到其他節點中。轉換程序檔會自動複製需要複寫的所有其他檔案。

- 1 確認 sshd 在所有具備對 root 開放之存取權限的節點上執行，以便轉換程序檔能成功將檔案複製到其他叢集節點。
- 2 High Availability Extension 隨附了預設的 OpenAIS 組態檔案。若在執行以下步驟時，不希望預設的組態被覆寫，請複製一份 `/etc/ais/openais.conf` 組態檔案。
- 3 以 root 身分啟動轉換程序檔。若正在使用 sudo，請使用 `-u` 選項指定有特權的使用者：

```
$ /usr/lib/heartbeat/hb2openais.sh -u root
```

程序檔會依據 `/etc/ha.d/ha.cf` 中儲存的組態，產生適用於 OpenAIS 叢集堆疊的新組態檔案 `/etc/ais/openais.conf`。程序檔還會分析 CIB 組態，並讓您知道您的叢集組態是否需要因從 Heartbeat 變為 OpenAIS

而變更。所有檔案處理都是在執行轉換的節點上完成，然後再複寫到其他節點。

**4** 按照螢幕上的指示執行操作。

成功完成轉換後，依第 3.3 節「連線叢集」 [24頁] 中所述啟動新的叢集堆疊。

完成升級後，就不能重新回復到 SUSE Linux Enterprise Server 10。

## 11.4 如需更多資訊

如需轉換程序檔和轉換階段的更多詳細資料，請參閱所安裝之 **High Availability Extension** 中的 `/usr/share/doc/packages/pacemaker/README.hb2openais`。



### III. 儲存與資料複製



## Oracle Cluster File System 2

Oracle Cluster File System 2 (OCFS2) 是一般用途的日誌式檔案系統，與 Linux 2.6 和更新版本的核心完全整合。OCFS2 可讓您將應用程式二進位檔案、資料檔案和資料庫儲存於設備上的共享儲存中。業集中所有節點均同時具有檔案系統的讀取與寫入權限。透過複製資源管理的使用者空間控制精靈可提供與 HA 堆疊 (特別是 OpenAIS 和 DLM) 的整合。

### 12.1 特點及優勢

在 SUSE Linux Enterprise Server 10 及更新版本中，可使用 OCFS2 做為以下儲存解決方案的範例：

- 一般應用程式與負荷
- 叢集中的 XEN 影像儲存

XEN 虛擬機器與虛擬伺服器可儲存於由叢集伺服器裝載的 OCFS2 卷冊，以在伺服器間提供快速而便捷的可攜式 XEN 虛擬機器。

- LAMP (Linux、Apache、MySQL 和 PHP | PERL | Python) 堆疊

此外，其亦與 OpenAIS 完全整合。

OCFS2 身為高效能、對稱、平行叢集檔案系統，支援下列功能：

- 業集上的所有節點都可使用應用程式的檔案。使用者只需在叢集上的 OCFS2 卷冊安裝一次即可。

- 所有節點可直接透過標準檔案系統介面同時讀取與寫入儲存區，讓執行於叢集上的應用程式易於管理。

- 「分散式鎖定管理員 (DLM)」會協調檔案存取權。

DLM 控制對於大部分狀況都非常實用，但應用程式的設計與 DLM 競爭檔案存取權協調能力的話，其擴充性可能會受限。

- 所有後端儲存區均可使用儲存區備份功能。您可輕鬆建立共享應用程式檔案的複本，以利於提供有效的災難復原。

OCFS2 亦提供下列功能：

- 中繼資料快取。
- 中繼資料日誌。
- 跨節點資料檔案一致性。
- 支援高達 4 KB 的多區塊大小 (各卷冊可具有不同的區塊大小)，卷冊的最大大小為 16 TB。
- 支援高達 16 個叢集節點。
- 對資料庫檔案提供非同步且直接 I/O 支援，以加強資料庫效能。

## 12.2 管理公用程式與指令

下表說明 OCFS2 公用程式。如需關於這些指令的語法資訊，請參閱其線上文件。

表格 12.1 OCFS2 共用程式

OCFS2 共用程式	描述
debugfs.ocfs2	以偵錯為目的，檢驗 OCFS 檔案系統。
fsck.ocfs2	檢查檔案系統是否有錯誤，並選擇性修復錯誤。

OCFS2 共用程式	描述
mkfs.ocfs2	在裝置上建立 OCFS2 檔案系統，該裝置通常是共享實體或邏輯磁碟上的分割區。
mounted.ocfs2	偵測並列出業集系統上的所有 OCFS2 卷冊。偵測並列出裝載 OCFS2 裝置的系統上之所有節點，或列出所有 OCFS2 裝置。
tunefs.ocfs2	變更 OCFS2 檔案系統參數，包括卷冊標籤、節點插槽數目、所有節點插槽的日至大小，以及卷冊大小。

## 12.3 OCFS2 套件

SUSE Linux Enterprise Server 11 HAE 中會自動安裝 OCFS2 核心模組 (ocfs2)。若要使用 OCFS2，請使用 YaST (或指令行) 在叢集內各節點上安裝 `ocfs2-tools` 和相符的 `ocfs2-kmp-*` 套件。

- 1 以根使用者或同等地位登入，再開啟「YaST 控制中心」。
- 2 選取「軟體」>「附加產品」。
- 3 選取「新增」以讓 SUSE Linux Enterprise High Availability Extension 可用。
- 4 選取「執行軟體管理員」並選擇「過濾」>「模式」。
- 5 選取「High Availability」模式進行安裝。
- 6 按一下「接受」並依照畫面上的指示操作。

## 12.4 建立 OCFS2 卷冊

遵循本節中的程序，設定您的系統使用 OCFS2 並建立 OCFS2 卷冊。

## 12.4.1 先決條件

開始之前，請執行下列步驟：

- 準備 OCFS2 磁碟區要使用的區塊設備。將裝置留為可用空間。  
建議您將應用程式檔案與資料檔案儲存於不同的 OCFS2 卷冊，但唯有您的應用程式卷冊與資料卷冊具有不同的裝載需求，才需強制執行此動作。
- 請確定已安裝 `ocfs2-tools` 套件。若尚未安裝的話，請使用 YaST 或指令行方法進行安裝。如需 YaST 說明，請參閱第 12.3 節「OCFS2 套件」[103頁]。

## 12.4.2 設定 OCFS2 服務

建立 OCFS2 磁碟區之前，必須先設定 OCFS2 服務。

請遵循此節程序設定叢集中的一節點。

- 1 開啟終端機視窗，並以根使用者或同等身分登入。
- 2 新增分散式鎖定管理員組態。
  - 2a 啟動 `crm` 外圍程序，並重新建立新的組態：

```
crm
cib new stack-glue
```

- 2b 建立 DLM 服務，並在叢集中的所有機器上執行該服務：

```
configure
primitive dlm ocf:pacemaker:controld op monitor interval=120s
clone dlm-clone dlm meta globally-unique=false interleave=true
end
```

- 2c 確認對叢集所做的變更，然後提交：

```
cib diff
configure verify
```

**2d** 將組態上載至叢集，並結束外圍程序：

```
cib commit stack-glue
quit
```

**3** 使用「*crm*」新增 O2CB 組態。

**3a** 啟動 *crm* 外圍程序，並重新建立新的組態：

```
crm
cib new oracle-glue
```


**3b** 設定 Pacemaker 以在叢集內各節點上啟動 o2cb 服務。

```
configure
primitive o2cb ocf:ocfs2:o2cb op monitor interval=120s
clone o2cb-clone o2cb meta globally-unique=false interleave=true
```

**3c** 請確定 Pacemaker 僅在同樣已執行 *dlm* 服務之副本的節點上啟動 o2cb 服務：

```
colocation o2cb-with-dlm INFINITY: o2cb-clone dlm-clone
order start-o2cb-after-dlm mandatory: dlm-clone o2cb-clone
end
```

**3d** 將組態上載至叢集，並結束外圍程序：

```
cib commit stack-glue 
quit
```

## 12.4.3 建立 OCFS2 卷冊

建立 OCFS2 檔案系統並將新節點增加至叢集，只能在叢集中的一節點執行。

- 1 開啟終端機視窗，並以根使用者或同等身分登入。
- 2 請使用指令 `crm_mon` 檢查叢集是否上線。

### 3 使用下列方法之一建立並格式化卷冊：

- 使用 `mkfs.ocfs2` 公用程式。如需此指令的語法資訊，請參閱 `mkfs.ocfs2` 線上文件。

若要在最多支援 16 個叢集節點的 `/dev/sdb1` 上建立新的 OCFS2 檔案系統，請使用

```
mkfs.ocfs2 -N 16 /dev/sdb1
```

請參閱下列表格以得知建議設定。

OCFS2 參數	描述與建議
卷冊標籤	<p>卷冊的描述性名稱可讓其裝載於不同節點時易於辨識。</p> <p>使用 <code>tunefs.ocfs2</code> 公用程式依需要修改標籤。</p>
叢集大小	<p>叢集大小是配置給持有資料的檔案之空間最小單位。</p> <p>選項有 4、8、16、32、64、128、256、512 和 1024 KB。格式化卷冊之後就無法修改叢集大小。</p> <p>Oracle 建議資料庫卷冊使用 128 KB 或更大的業集。Oracle 亦建議 Oracle Home 使用 32 或 64 KB 的叢集大小。</p>
節點插槽名稱	<p>可同時裝載卷冊的最大節點數目。對於每個節點，OCFS2 會分別為其建立系統檔案，例如日誌。存取卷冊的節點可以是小 endian 架構 (如 x86、x86-64 和 ia64) 和大 endian 架構 (如 ppc64 和 s390x) 的組合。</p> <p>節點特定的檔案會被視為本機檔案。節點插槽號碼會附加至本機檔案。例如：<code>journal:0000</code> 隸屬於指派至插槽 0 的任一節點。</p> <p>建立時請根據您希望同時裝載卷冊的節點數量，設定節點插槽的最大數目。使用 <code>tunefs.ocfs2</code> 公用程式依需要增加節點插槽數目；該數值無法減少。</p>

---

**OCFS2 參數 描述與建議**

---

區塊大小 檔案系統可定址的空間最小單位。請在建立卷冊時指定區塊大小。

選項有 512 位元 (不建議)、1 KB、2 KB 或 4 KB (建議大部分卷冊使用)。格式化卷冊之後就無法修改區塊大小。

---

## 12.5 裝載 OCFS2 卷冊

- 1 開啟終端機視窗，並以根使用者或同等身分登入。
- 2 請使用指令 `crm_mon` 檢查叢集是否上線。
- 3 使用下列方法之一裝載卷冊。

---

**警告：手動裝載 OCFS2 設備**

---

若手動裝載 `ocfs2` 檔案系統以供測試使用，則需要在透過 `OpenAIS` 開始使用之前，再次卸載該檔案系統。

---

- 在 `ocfs2console` 中，於「可用裝置」清單中選擇一裝置，按一下「裝載」指定目錄裝載點與裝載選項 (選用)，再按一下「確定」。
- 從指令行裝載卷冊，請使用 `mount` 指令。
- 使用叢集管理員裝載檔案系統。`ocf`資源 `Filesystem` 可用於此任務。如需詳細資訊，請參閱使用叢集管理員裝載檔案系統 [108頁]。

成功裝載後，`ocfs2console` 中的裝置清單會顯示裝載點與裝置。

---

**選項****描述**

---

`datavolume`

確認 Oracle 程序以 `o_direct` 旗標開啟檔案。

選項	描述
nointr	無岔斷。確認 IO 未被訊號岔斷。

若要設定檔案系統資源，請使用以下程序：

### 過程 12.1 使用叢集管理員裝載檔案系統

- 1 啟動 `crm` 外圍程序，並重新建立新的組態：

```
crm
cib new filesystem
```

- 2 設定 `Pacemaker` 以在叢集內各節點上裝載檔案系統。

```
configure
primitive fs ocf:heartbeat:Filesystem \
    params device="/dev/sdb1" directory="/mnt/shared" fstype="ocfs2" \
    op monitor interval=120s
clone fs-clone fs meta interleave="true" ordered="true"
```

- 3 請確定 `Pacemaker` 僅在同樣已執行 `o2cb` 資源之複製品的節點上啟動 `o2cb` 複製資源：

```
colocation fs-with-o2cb INFINITY: fs-clone o2cb-clone
order start-fs-after-o2cb mandatory: o2cb-clone fs-clone
end
```

- 4 將組態上載至叢集，並結束外圍程序：

```
cib commit filesystem
quit
```

## 12.6 其他資訊

如需使用 OCFS2 的更多資訊，請參閱「Oracle 的 OCFS2 專案 [<http://oss.oracle.com/projects/ocfs2/>]

」上的《*OCFS2 使用者指南* [<http://oss.oracle.com/projects/ocfs2/documentation/>]》。



## 叢集 LVM

管理叢集上的共享儲存時，當儲存子系統發生變更時必須通知每個節點。廣泛用於管理本地儲存的 Linux Volume Manager 2 (LVM2) 經過延伸，可支援在整個叢集中對磁碟區群組進行透明管理。可使用與本地儲存相同的指令來管理叢集化磁碟區群組。

若要實作 cLVM，必須有可用的共享儲存設定 (例如，由 Fibre Channel、FCoE、SCSI 或 iSCSI SAN 或 DRBD 提供的共享儲存設定)。

在內部，cLVM 使用叢集堆疊的分散式鎖定管理員 (DLM) 元件協調對 LVM2 中繼資料的存取。由於 DLM 會依次與堆疊的其他元件 (例如圍籬區隔) 整合，所以共享儲存的完整性將隨時受到保護。

cLVM 不會協調對共享資料本身的存取；若要執行此操作，必須在受 cLVM 管理之儲存的頂層設定 OCFS2 或其他支援叢集應用程式。

### 13.1 cLVM 的組態

若要設定支援叢集磁碟區群組，必須成功完成以下幾項任務：

- 1 變更要成為支援叢集的 LVM2 的鎖定類型。

編輯 `/etc/lvm/lvm.conf` 檔案並找到下面一行：

```
locking_type = 1
```

將鎖定類型變更為 3，並將組態寫入磁碟。將此組態複製到所有節點。

- 2 將 `clvmd` 資源做為複製品包括在 Pacemaker 組態中，並使之依賴於 DLM 複製資源。`crm` 組態外圍程序中的典型片段如下所示：

```
primitive dlm ocf:pacemaker:controld
primitive clvm ocf:lvm2:clvmd \
    params daemon_timeout="30"
clone dlm-clone dlm \
    meta target-role="Started" interleave="true" ordered="true"
clone clvm-clone clvm \
    meta target-role="Started" interleave="true" ordered="true"
colocation colo-clvm inf: clvm-clone dlm-clone
order order-clvm inf: dlm-clone clvm-clone
...
```

繼續之前，請確定已在叢集中成功啟動這些資源。可以使用 `crm_mon` 或 GUI 來檢查執行中的服務。

- 3 使用下列指令準備 LVM 的實體磁碟區：

```
pvcreate <physical volume path>
```

- 4 建立支援叢集磁碟區群組：

```
vgcreate --clustered y <volume group name> <physical volume path>
```

- 5 視需要建立邏輯磁碟區，例如：

```
lvcreate --name testlv -L 4G <volume group name>
```

- 6 為了確保在整個叢集範圍啟動磁碟區群組，請按如下方式設定 LVM 資源：

```
primitive vg1 ocf:heartbeat:LVM \
    params volgrpname="<volume group name>"
clone vg1-clone vg1 \
    meta interleave="true" ordered="true"
colocation colo-vg1 inf: vg1-clone clvm-clone
order order-vg1 inf: clvm-clone vg1-clone
```

- 7 若要僅在一個節點上獨自啟動磁碟區群組，請使用下面的範例；在此範例中，由於針對非叢集化應用程式採用其他保護措施，`clvm` 可避免在多個節點上啟動 VG 中的所有邏輯磁碟區：

```
primitive vg1 ocf:heartbeat:LVM \  
    params volgrpname="<volume group name>" exclusive="yes"  
colocation colo-vg1 inf: vg1 clvm-clone  
order order-vg1 inf: clvm-clone vg1
```

- 8 現在，VG 中的邏輯磁碟區可做為檔案系統裝載或 RAW 使用量使用。請確定使用它們的服務務必具有正確的相依性，這樣才能在啟動 VG 後對它們進行並存和排序處理。

完成這些組態設定步驟後，即可像在任何獨立工作站上一樣進行 LVM2 組態設定。

## 13.2 明確設定適合的 LVM2 設備

若有多個設備似乎共享同一個實體磁碟區簽名時 (多重路徑設備或 drbd 就有可能發生這種情況)，建議明確設定 LVM2 掃描 PV 的設備。

例如，如果 `vgcreate` 指令使用實體設備而非使用鏡像複製區塊設備，將使 DRBD 感到困惑，從而導致 DRBD 處於電腦分裂狀態。

若要停用 LVM2 的單一設備，請執行以下操作：

- 1 編輯 `/etc/lvm/lvm.conf` 檔案並搜尋以 `filter` 開頭的行。
- 2 該處的模式將被視為正規表示式進行處理。前置「a」表示接受要掃描的設備模式，前置「r」表示拒絕依照該設備模式的設備。
- 3 若要移除名為 `/dev/sdb1` 的設備，請將下列表示式新增至過濾器規則：

```
"r|^/dev/sdb1$|"
```

完整的過濾器行如下所示：

```
filter = [ "r|^/dev/sdb1$|", "r|/dev/.*/by-path/.*/",  
"r|/dev/.*/by-id/.*/", "a./" ]
```

接受 DRBD 和 MPIO 設備但拒絕所有其他設備的過濾器行如下所示：

```
filter = [ "a|/dev/drbd.*|", "a|/dev/.*by-id/dm-uuid-mpath-.*|",  
"r/.*/" ]
```

- 4 寫入組態檔案並將其複製到所有叢集節點。

## 13.3 如需更多資訊

更多資訊可於 Pacemaker 郵寄清單中取得 (位於 <http://www.clusterlabs.org/wiki/Help:Contents>)。

官方 cLVMFAQ 可在 <http://sources.redhat.com/cluster/wiki/FAQ/CLVM> 中找到。

## 分散式複製區塊設備 (DRBD)

DRBD 可讓您跨 IP 網路為位於兩個不同站台的兩個區塊設備建立鏡像複製。與 OpenAIS 搭配使用時，DRBD 支援分散式的 High Availability Linux 叢集。

---

### 重要

鏡像複製間的資料流量不會加密。為了保證資料交換的安全，您應該為連線部署虛擬私人網路 (VPN) 解決方案。

---

將資料從主要設備複製到次要設備的過程中，會使用特定方式保證兩份資料始終一致。使用 ocfs2 等支援叢集檔案系統時，也可以將兩個節點做為主要設備執行。

依預設，DRBD 使用 TCP 埠 7788 在 DRBD 節點間進行通訊。請確定您的防火牆不會阻止此埠上的通訊。

您必須先對 DRBD 設備進行設定，然後才能在其上建立檔案系統。所有與使用者資料相關的操作都必須透過 `/dev/drbd<n>` 設備來獨立執行，而不是在 Raw 設備上完成，因為 DRBD 會將 Raw 設備剩下的 128 MB 容量用於儲存中繼資料。請務必僅在 `/dev/drbd<n>` 設備上而不要在 Raw 設備上建立檔案系統。

例如，如果 Raw 設備的大小為 1024 MB，則 DRBD 設備只能使用 896 MB 來儲存資料，另有 128 MB 隱藏留作儲存中繼資料之用。任何存取大小為 896 MB 至 1024 MB 之間的空間的嘗試都會失敗，因為這樣的空間不適用於使用者資料。

## 14.1 安裝 DRBD 服務

若要為 drbd 安裝所需套件，請在網路叢集中的兩台 SUSE Linux Enterprise Server 機器上安裝 High Availability Extension 附加產品，如第 I 部分「安裝與設定」[1頁] 中所述。安裝 High Availability Extension 的同時也會安裝 drbd 程式檔案。

如果不需要完整的叢集堆疊，而只要使用 drbd，也可以新增 High Availability Extension 附加產品，然後繼續安裝 drbd。此安裝過程中還會安裝所需的核心理模組。

## 14.2 設定 DRBD 服務

---

### 注

下列程序使用伺服器名稱 `node 1` 與 `node 2`，以及叢集資源名稱 `r0`。將 `node 1` 設定為主要節點。請務必修改指示，以使用您自己的節點與檔案名稱。

---

- 1 啟動 YaST 並選取組態模組「其他」>「*drbd*」。
- 2 在「啟動組態」>「開機」中，選取「開啟」以在開機時始終啟動 drbd。
- 3 如果需要設定多個複製資源，請選取「全域組態」。在輸入欄位「次要計數」中，選取在不重新啟動電腦的情況下可以設定的不同 drbd 資源的數量。
- 4 實際的資源組態在「資源組態」中設定。按「新增」可建立新資源。必須設定以下參數：

---

「資源名稱」	資源的名稱，通常稱為 <code>r0</code> 。
「名稱」	各節點的主機名稱。
「位址:埠」	各節點的 IP 位址與埠號。

「設備」	存放各節點上的複製資料的設備。使用此設備可建立檔案系統並裝載作業。
「磁碟」	在兩個節點間複製的設備。
「中繼磁碟」	<p>「中繼磁碟」可設定為 <code>internal</code> 值，或指定由索引延伸的明確設備，以存放 <code>drbd</code> 所需的中繼資料。</p> <p>使用 <code>internal</code> 時，複製設備剩下的 128 MB 容量將用於儲存中繼資料。</p> <p>也可為多重 <code>drbd</code> 資源使用一個真實設備。例如，如果第一個資源使用的「中繼磁碟」為 <code>/dev/sda6[0]</code>，您可以將 <code>/dev/sda6[1]</code> 用於第二個資源。不過，此磁碟上必須為每個資源至少預留 128 MB 的空間。</p>

---

上述所有選項在 `/usr/share/doc/packages/drbd/drbd.conf` 檔案與 `drbd.conf(5)` 線上文件中都以範例進行了說明。

- 將 `/etc/drbd.conf` 檔案複製到次要伺服器 (node 2) 上的 `/etc/drbd.conf` 位置。

```
scp /etc/drbd.conf <node 2>:/etc
```

- 在每個節點上輸入以下指令，以在兩個系統上啟始化並啟動 DRBD 服務：

```
drbdadm create-md r0
rcdrbd start
```

- 在 `node1` 上輸入以下指令，將 `node1` 設定為主要節點。

```
drbdsetup /dev/drbd0 primary --overwrite-data-of-peer
```

- 在每個節點上輸入以下指令，檢查 DRBD 服務狀態：

```
rcdrbd status
```

等到兩個節點上的區塊設備完全同步後再繼續操作。重複 `rcdrbd status` 指令瞭解同步化進度。

- 9 兩個節點上的區塊設備完全同步後，以 `reiserfs` 等檔案系統格式化主要節點上的 DRBD 設備。可以使用任何 Linux 檔案系統。例如，輸入

```
mkfs.reiserfs -f /dev/drbd0
```

---

### 重要

務必在指令中使用 `/dev/drbd<n>` 名稱，而不是實際的 `/dev/disk` 設備名稱。

---

## 14.3 測試 DRBD 服務

如果安裝與組態程序按預期執行，您現在就可以執行基本的 DRBD 功能測試。此測試也有助於瞭解軟體的工作原理。

- 1 測試 node 1 上的 DRBD 服務。

- 1a 開啟終端機主控台，以 `root` 使用者或同等身份登入。

- 1b 輸入以下指令，在節點 1 上建立裝載點，如 `/srv/r0mount`

```
mkdir -p /srv/r0mount
```

- 1c 輸入以下指令，裝載 `drbd` 設備

```
mount -o rw /dev/drbd0 /srv/r0mount
```

- 1d 輸入以下指令，以從主要節點建立檔案

```
touch /srv/r0mount/from_node1
```

- 2 測試 node 2 上的 DRBD 服務。

**2a** 開啟終端機主控台，以 `root` 使用者或同等身份登入。

**2b** 在 `node 1` 上輸入以下指令，卸下 `node 1` 上的磁碟：

```
umount /srv/r0mount
```

**2c** 在 `node 1` 上輸入以下指令，將 `node 1` 上的 DRBD 服務降級：

```
drbdadm secondary r0
```

**2d** 在 `node 2` 上輸入以下指令，將 DRBD 服務升級

```
drbdadm primary r0
```

**2e** 在 `node 2` 上輸入以下指令，檢查 `node 2` 是否為主要節點

```
rcdrbd status
```

**2f** 輸入以下指令，在節點 2 上建立裝載點，如 `/srv/r0mount`

```
mkdir /srv/r0mount
```

**2g** 在 `node 2` 上輸入以下指令，裝載 DRBD 設備

```
mount -o rw /dev/drbd0 /srv/r0mount
```

**2h** 輸入以下指令，驗證在節點 1 上建立的檔案是否可以檢視

```
ls /srv/r0mount
```

此時應列出 `/srv/r0mount/from_node1` 檔案。

**3** 如果服務在兩個節點上都可執行，即表示 DRBD 設定已完成。

**4** 再次將 `node 1` 設為主要節點。

**4a** 在 `node 2` 上輸入以下指令，卸下 `node 2` 上的磁碟：

```
umount /srv/r0mount
```

**4b** 在 node 2 上輸入以下指令，將 node 2 上的 DRBD 服務降級：

```
drbdadm secondary r0
```

**4c** 在 node 1 上輸入以下指令，將 DRBD 服務升級為主要服務

```
drbdadm primary r0
```

**4d** 在 node 1 上輸入以下指令，檢查 node 1 是否為主要節點

```
rcdrbd status
```

- 5** 若要讓服務自動啟動並在伺服器出現問題時自動進行容錯移轉，您可以使用 OpenAIS 將 DRBD 設定為高可用性服務。如需安裝和設定適用於 SUSE Linux Enterprise 的 OpenAIS 的資訊，請參閱第 II 部分「組態與管理」[27頁]。

## 14.4 DRBD 疑難排解

drbd 設定涉及眾多不同元件以及產生原因不同的各種問題。以下幾節介紹了一些常見問題，並提供了解決這些問題的提示。

### 14.4.1 組態

如果初始的 drbd 設定未按預期工作，則可能是組態有問題。

若要獲取有關組態的資訊，請執行下列步驟：

- 1** 開啟終端機主控台，然後以 root 使用者身份登入。
- 2** 執行 drbdadm (含 -d 選項)，測試組態檔案。輸入

```
drbdadm -d adjust r0
```

在 `adjust` 選項的試執行 (`dry run`) 期間，`drbdadm` 會將 DRBD 資源的實際組態與 DRBD 組態檔案進行比較，但不會執行呼叫。檢閱輸出，以確定您瞭解所有錯誤的來源及原因。

- 3 如果 `drbd.conf` 檔案中存在錯誤，請先校正這些錯誤再繼續。
- 4 如果分割區與設定均正確，請再次執行 `drbdadm` (不含 `-d` 選項)。輸入

```
drbdadm adjust r0
```

此指令會將組態檔案套用於 DRBD 資源。

## 14.4.2 主機名稱

對於 DRBD 而言，主機名稱是區分大小寫的，因此 `Node0` 與 `node0` 是不同的主機名稱。

如果您擁有多個網路設備，並希望使用專屬的網路設備，主機名稱可能不會解析成所使用的 IP 位址。在這種情況下，請使用參數 `disable-ip-verification` 讓 DRBD 忽略此現象。

## 14.4.3 TCP 埠 7788

如果您的系統無法連接至點，這可能是因為本地防火牆出現了問題。依預設，DRBD 使用 TCP 埠 7788 存取其他節點。請確定在兩個節點上均可存取此連接埠。

## 14.4.4 DRBD 設備在重新開機後損毀

如果 DRBD 不知道存放最新資料的真實設備，就會導致電腦分裂狀態。在這種情況下，各個 DRBD 子系統將會做為次要項目出現，並且不會相互連接。在這種情況下，會將以下訊息寫入到 `/var/log/messages`：

```
Split-Brain detected, dropping connection!
```

若要解決此情況，您必須選取一個應丟棄修改的節點。登入該節點，然後執行以下指令：

```
drbdadm secondary r0
drbdadm -- --discard-my-data connect r0
```

在其他節點上，執行指令：

```
drbdadm connect r0
```

## 14.5 其他資訊

下列開放原始碼資源適用於 DRBD：

- 可在套裝軟體中獲取 DRBD 的下列線上文件：

```
drbd(8)
drbddisk(8)
drbdsetup(8)
drbdadm(8)
drbd.conf(5)
```

- `/usr/share/doc/packages/drbd/drbd.conf` 中可找到含備註的 DRBD 範例組態。
- 專案首頁 <http://www.drbd.org>。
- [http://clusterlabs.org/wiki/DRBD\\_HowTo\\_1.0](http://clusterlabs.org/wiki/DRBD_HowTo_1.0)，由 Linux Pacemaker Cluster Stack Project 提供。

## IV. 疑難排解與參考



## 疑難排解

尤其在剛剛開始使用 Heartbeat 時，可能會出現一些難以理解的棘手問題。但您可借助幾個公用程式更為仔細地瞭解 Heartbeat 內部程序。使用本章可取得如何解決問題的一些方法。

### 15.1 安裝問題

若在安裝套件或連線叢集時遇到困難，請依照以下清單繼續執行。

是否已安裝 HA 套件？

設定和管理叢集所需的套件包含於 High Availability 安裝模式中，由 High Availability Extension 提供。

請檢查 High Availability Extension 是否在各叢集節點上安裝為 SUSE Linux Enterprise Server 11 的附加產品，以及「*High Availability*」模式是否依第 3.1 節「安裝 High Availability Extension」[21 頁] 中所述安裝於各機器上。

所有叢集節點的初始組態是否都相同？

若要在彼此之間進行通訊，則屬於相同叢集的所有節點均需依第 3.2 節「初始叢集設定」[22 頁] 中所述使用相同的 `bindnetaddr`、`mcastaddr` 與 `mcastport`。

請檢查 `/etc/ais/openais.conf` 中為所有叢集節點所設定的通訊通道與選項是否都相同。

若使用加密通訊，請檢查是否所有叢集節點上的 `/etc/ais/authkey` 檔案都可使用。

防火牆允許透過 `mcastport` 進行通訊嗎？

若用於在各叢集節點間通訊的 `mcastport` 被防火牆阻擋，這些節點將無法相互查看。依第 3.1 節「安裝 High Availability Extension」[21 頁] 中所述使用 YaST 設定初始設定時，通常會自動調整防火牆設定。

若要確保 `mcastport` 不被防火牆阻擋，請檢查各節點上 `/etc/sysconfig/SuSEfirewall12` 中的設定。或者，啟動各叢集節點上的 YaST 防火牆模組。按一下「允許的服務」>「進階」之後，將 `mcastport` 新增至允許的「UDP 埠」清單，然後確認變更。

在各叢集節點上啟動了 OpenAIS 嗎？

使用 `/etc/init.d/openais status` 檢查各叢集節點上的 OpenAIS 狀態。若 OpenAIS 沒有執行，則透過執行 `/etc/init.d/openais start` 將其啟動。

## 15.2 HA 叢集除錯

若某些項對於您的叢集不起作用，請先嘗試以下動作。它會顯示資源作業歷程 (選項 `-o`) 和非使用中資源 (`-r`):

```
crm_mon -o -r
```

顯示內容每 10 秒重新整理一次 (您可以透過按 `Ctrl + C` 取消該操作)。範例顯示如下：

## 範例 15.1 停止的資源

```
Refresh in 10s...

=====
Last updated: Mon Jan 19 08:56:14 2009
Current DC: d42 (d42)
3 Nodes configured.
3 Resources configured.
=====

Online: [ d230 d42 ]
OFFLINE: [ clusternode-1 ]

Full list of resources:

Clone Set: o2cb-clone
           Stopped: [ o2cb:0 o2cb:1o2cb:2 ]
Clone Set: dlm-clone
           Stopped [ dlm:0 dlm:1 dlm:2 ]
mySecondIP (ocf::heartbeat:IPaddr):          Stopped

Operations:
* Node d230:
  aa: migration-threshold=1000000
    + (5) probe: rc=0 (ok)
    + (37) stop: rc=0 (ok)
    + (38) start: rc=0 (ok)
    + (39) monitor: interval=15000ms rc=0 (ok)
* Node d42:
  aa: migration-threshold=1000000
    + (3) probe: rc=0 (ok)
    + (12) stop: rc=0 (ok)
```

首先連線節點 (請參閱節 15.3 [127頁])。然後檢查資源與作業。

<http://clusterlabs.org/wiki/Documentation> 下的 *Configuration Explained* PDF 會在 *How Does the Cluster Interpret the OCF Return Codes?* (叢集如何解譯 OCF 傳回代碼?) 一節中說明三種不同的復原類型：

## 15.3 常見問題集

我的叢集狀態為何?

若要檢查您叢集的目前狀態，請使用程式 `crm_mon`。這會顯示目前 DC，以及目前節點已知的所有節點與資源。

我叢集的一些節點相互間看不到。

導致的原因有多種：

- 請先查看組態檔案 `/etc/ais/openais.conf`，然後檢查叢集內各節點的多路傳播位址是否相同 (使用鍵 `mcastaddr` 在 `interface` 區段中搜尋)。
- 檢查防火牆設定。
- 檢查交換器是否支援多路傳播位址
- 還可能是因為您節點間的連線中斷。最常見的原因是防火牆設定不正確。這還可能是導致電腦分裂狀況的原因，其中的叢集被分區。

我要列出目前已知的資源。

使用指令 `crm_resource -L` 瞭解目前資源。

我已設定某個資源，但它總是失敗。

請嘗試手動執行資源代辦。使用 LSB，只需執行程序檔名稱 `start` 與程序檔名稱 `stop`。若要檢查 OCF 程序檔，請先設定所需的環境變數。例如，測試 `IPaddr` OCF 程序檔時，必須設定變數 `ip` 的值，方法是設定使用 `OCF_RESKEY_` 做為變數名稱字首的環境變數。對於此範例，執行以下指令：

```
export OCF_RESKEY_ip=<your_ip_address>
/usr/lib/ocf/resource.d/heartbeat/IPaddr validate-all
/usr/lib/ocf/resource.d/heartbeat/IPaddr start
/usr/lib/ocf/resource.d/heartbeat/IPaddr stop
```

若此指令失敗，很有可能是缺少了某些必要變數，或是參數拼寫錯誤。

我剛收到一條失敗訊息。能取得詳細資訊嗎？

您可以始終將 `-v` 參數新增至指令。若多次執行此操作，將產生冗長的除錯輸出。

如何清理我的資源？

若知道您資源的 ID (可透過使用 `crm_resource -L` 取得)，即可使用 `crm_resource -C -r 資源 ID -H 主機` 移除特定資源。

無法裝載 `ocfs2` 設備。

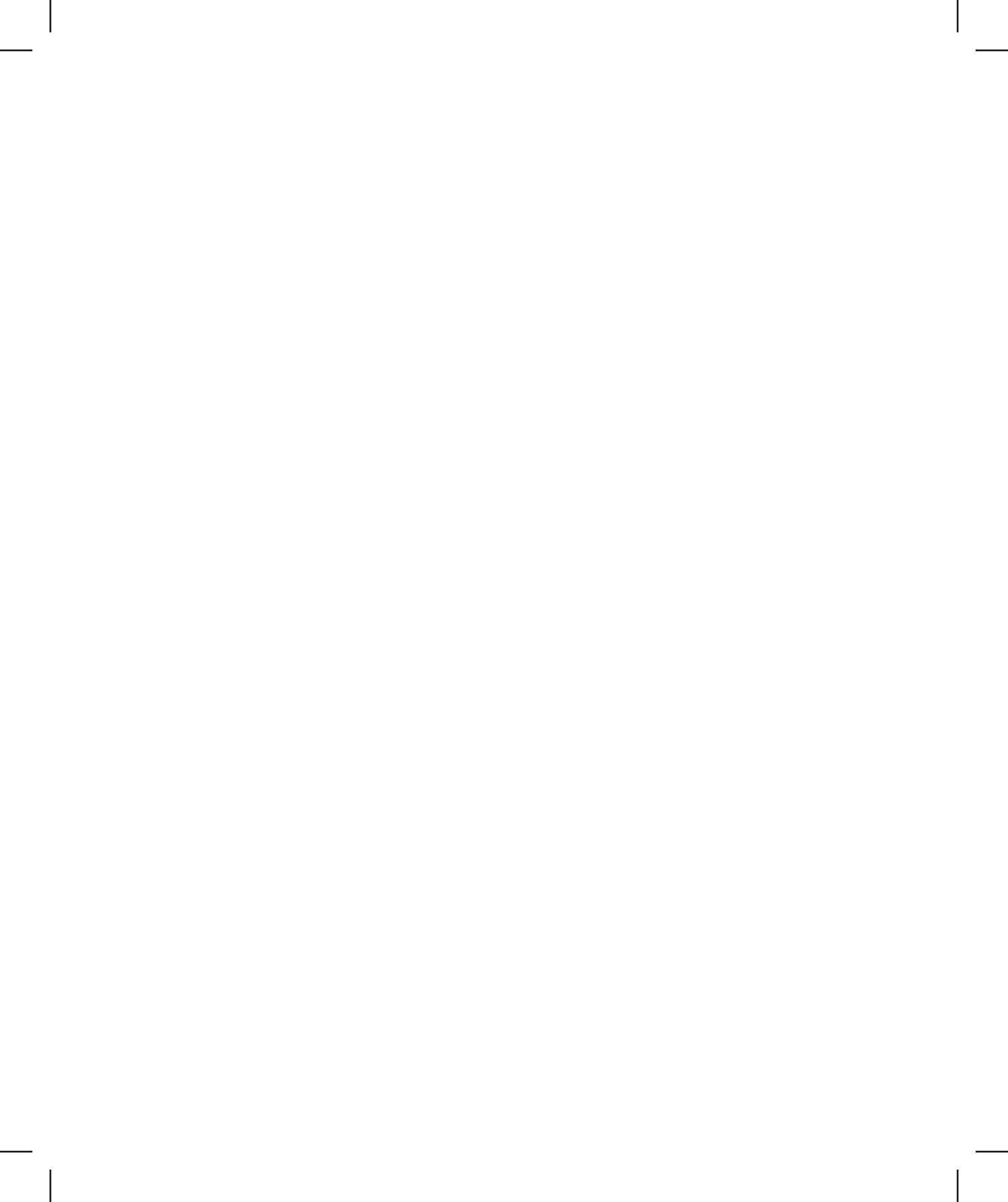
檢查 `/var/log/message` 是否有以下行：

```
Jan 12 09:58:55 clusternode2 lrmd: [3487]: info: RA output:
(o2cb:1:start:stderr) 2009/01/12_09:58:55
  ERROR: Could not load ocfs2_stackglue
Jan 12 16:04:22 clusternode2 modprobe: FATAL: Module ocfs2_stackglue not
found.
```

此範例中，缺少了核心模組 `ocfs2_stackglue.ko`。安裝套件 `ocfs2-kmp-default`、`ocfs2-kmp-pae` 或 `ocfs2-kmp-xen`，具體視所安裝的核心而定。

## 15.4 獲取詳細資訊

如需 Linux 和 Heartbeat 上有關高可用性的其他資訊，包括設定叢集資源及管理  
和自定 Heartbeat 叢集，請參閱 <http://clusterlabs.org/wiki/Documentation>。



# 16

## 叢集管理工具

High Availability Extension 內建有可協助您利用指令行管理叢集的全面工具組。本章介紹管理 CIB 中的叢集組態與叢集資源所需的工具。第 15 章「疑難排解」[125頁] 中介紹了管理資源代辦的其他指令行工具或用於對設定進行除錯和疑難排解的工具。

以下清單提供了多項與叢集管理相關的任務，並簡要介紹了用於完成這些任務的工具：

### 監控叢集狀態

`crm_mon` 指令可讓您監控叢集狀態與組態。其輸出包括節點數、`uname`、`uuid`、狀態、叢集中設定的資源及其各自的目前狀態。`crm_mon` 的輸出可顯示在主控台中或列印成 HTML 檔案。如果提供無狀態區段的叢集組態檔案，`crm_mon` 就會建立節點及資源的綜覽 (如檔案中所指定)。如需此工具的使用與指令語法的詳細介紹，請參閱 `crm_mon(8)` [153頁]。

### 管理 CIB

`cibadmin` 指令為低階管理指令，用於操作 Heartbeat CIB。它可用來傾印、更新及修改所有或部分 CIB，刪除整個 CIB，或執行其他 CIB 管理作業。如需此工具的使用與指令語法的詳細介紹，請參閱 `cibadmin(8)` [133頁]。

### 管理組態變更

`crm_diff` 指令可協助您建立並套用 XML 修補程式。此作業有助於視覺化兩個叢集組態版本之間的變更，或儲存變更以便稍後使用 `cibadmin(8)` [133頁] 套用變更。如需此工具的使用與指令語法的詳細介紹，請參閱 `crm_diff(8)` [145頁]。

## 操作 CIB 屬性

`crm_attribute` 指令可讓您查詢並操作節點屬性和要在 CIB 中使用的叢集組態選項。如需此工具的使用與指令語法的詳細介紹，請參閱 `crm_attribute(8)` [142頁]。

## 驗證叢集組態

`crm_verify` 指令可檢查組態資料庫 (CIB) 的一致性及其他問題。它可檢查檔案是否包含組態或連接正在執行的叢集。它可報告兩類問題。必須先修復錯誤，`Heartbeat` 才能正常工作，管理員負責解決警告問題。`crm_verify` 可協助建立新組態或修改的組態。您可在執行中的叢集中建立 CIB 的本地副本、對其進行編輯，並使用 `crm_verify` 進行驗證，然後使用 `cibadmin` 使新組態生效。如需此工具的使用與指令語法的詳細介紹，請參閱 `crm_verify(8)` [180頁]。

## 管理資源組態

`crm_resource` 指令可在叢集上執行各種與資源相關的動作。它可讓您修改已設定資源的定義，啟動和停止資源，或在節點間刪除和移轉資源。如需此工具的使用與指令語法的詳細介紹，請參閱 `crm_resource(8)` [157頁]。

## 管理資源失敗計數

`crm_failcount` 指令會查詢指定節點上每個資源的失敗次數。此工具還可用來重設失敗計數，允許資源在失敗次數過多的節點上重新執行。如需此工具的使用與指令語法的詳細介紹，請參閱 `crm_failcount(8)` [148頁]。

## 管理節點的待機狀態

`crm_standby` 指令可操作節點的待機屬性。待機模式下的任何節點都無法再代管資源，且其中的所有資源也都必須移出。對於執行核心更新等維護任務，待機模式十分有用。當節點再次成為叢集的完全使用中成員時，請從節點移除待機屬性。如需此工具的使用與指令語法的詳細介紹，請參閱 `crm_standby(8)` [177頁]。

# cibadmin (8)

cibadmin — 提供對叢集組態的直接存取

## 概要

可讓您查詢、修改、取代和刪除整個或部分組態。

```
cibadmin (--query|-Q) -[Vrwlsmfbp] [-i xml-object-id|-o
xml-object-type] [-t t-flag-whatever] [-h hostname]

cibadmin (--create|-C) -[Vrwlsmfbp] [-X xml-string]
[-x xml- filename] [-t t-flag-whatever] [-h hostname]

cibadmin (--replace|-R) -[Vrwlsmfbp] [-i xml-object-id|
-o xml-object-type] [-X xml-string] [-x xml-filename] [-t
t-flag- whatever] [-h hostname]

cibadmin (--update|-U) -[Vrwlsmfbp] [-i xml-object-id|
-o xml-object-type] [-X xml-string] [-x xml-filename] [-t
t-flag- whatever] [-h hostname]

cibadmin (--modify|-M) -[Vrwlsmfbp] [-i xml-object-id|
-o xml-object-type] [-X xml-string] [-x xml-filename] [-t
t-flag- whatever] [-h hostname]

cibadmin (--delete|-D) -[Vrwlsmfbp] [-i xml-object-id|
-o xml-object-type] [-t t-flag-whatever] [-h hostname]

cibadmin (--delete_alt|-d) -[Vrwlsmfbp] -o
xml-object-type [-X xml-string|-x xml-filename]
[-t t-flag-whatever] [-h hostname]

cibadmin --erase (-E)

cibadmin --bump (-B)

cibadmin --ismaster (-m)

cibadmin --master (-w)

cibadmin --slave (-r)

cibadmin --sync (-S)

cibadmin --help (-?)
```

## 描述

`cibadmin` 指令為低階管理指令，用於操作 Heartbeat CIB。它可用於傾印整個或部分 CIB、修改整個或部分 CIB、刪除整個 CIB，或執行其他 CIB 管理作業。

`cibadmin` 對 CIB 的 XML 樹狀結構執行操作時，通常並不瞭解所執行的更新或查詢的意義。這表示對瞭解 XML 樹狀結構中各元素意義的使用者而言，那些習以為常的捷徑無法與 `cibadmin` 搭配使用。該指令要求完全不使用模糊語意，且只能處理輸入與輸出的有效 XML 子樹狀結構 (標籤與元素)。

---

### 注

使用者應一律優先選擇使用 `cibadmin`，而不是手動編輯 `cib.xml` 檔案 — 特別是在叢集為使用中狀態時更應如此。叢集會盡力偵測並防止這種做法，以免資料遺失或損毀。

---

## 選項

- `--obj_type` *物件類型*、`-o` *物件類型*  
指定要對其執行操作的物件類型。有效值包括 `nodes`、`resources`、`constraints`、`crm_status` 及 `status`。
- `--verbose`、`-V`  
開啟除錯模式。另外的 `-v` 選項可增加輸出的詳細度。
- `--help`、`-?`  
從 `cibadmin` 獲得說明訊息。
- `--xpath` *PATHSPEC*、`-A` *PATHSPEC*  
提供有效的 XPath 以取代 `obj_type`。

## 指令

`--bump`、`-B`

增加 CIB 中 epoch 的版本計數器。此值通常在選舉新領袖時由叢集自動增加。若要讓較舊的組態過時(例如儲存在非使用中叢集節點上的組態)，則手動增加此值會很有用。

`--create`、`-C`

使用引數的 XML 內容建立新的 CIB。

`--delete`、`-D`

刪除與提供的準則相符的第一個物件，例如，`<op id="rscl_op1" name="monitor"/>`。標籤名稱及所有屬性皆必須相符才能刪除元素

`--erase`、`-E`

去除整個 CIB 的內容。

`--ismaster`、`-m`

列印一則訊息，指出 CIB 軟體的本地例項是否為主例項。如果本地例項為主例項，則結束時會傳回代碼 0，否則會傳回代碼 35。

`--modify`、`-M`

在 CIB 的 XML 樹狀結構中找到物件並將其更新。

`--query`、`-Q`

查詢 CIB 的一部分。

`--replace`、`-R`

以遞迴方式取代 CIB 中的 XML 物件。

`--sync`、`-S`

強制將所有節點與指定主機上的 CIB 重新同步(若使用 `-h` 選項)，或強制將所有節點與 DC 重新同步(若未使用 `-h` 選項)。

## XML 資料

`--xml-text` 字串、`-X` 字串

指定應對其執行 `crmadmin` 的 XML 標籤或片段。它必須是完整的標籤或 XML 片段。

`--xml-file` 檔名、`-x` 檔名

指定應對其執行 `crmadmin` 的檔案中的 XML。它必須是完整的標籤或 XML 片段。

`--xml_pipe`、`-p`

指定應對其執行 `cibadmin` 的 XML 來自於標準輸入。它必須是完整的標籤或 XML 片段。

## 進階選項

`--host` 主機名稱、`-h` 主機名稱

將指令傳送至指定主機。僅適用於 `query` 及 `sync` 指令。

`--local`、`-l`

讓指令在本地生效 (極少使用，為進階選項)。

`--no-bcast`、`-b`

不廣播指令，即使已變更 CIB 亦如此。

---

### 重要

使用此選項時請謹慎，以免導致叢集分歧。

---

`--sync-call`、`-s`

等待呼叫完成後再返回。

## 範例

若要取得傳遞至 `stdout` 的整個使用中 CIB 的副本 (包括狀態區段等)，請發出以下指令：

```
cibadmin -Q
```

若要將 IPAddr2 資源新增至 *resources* 區段，請先建立含有下列內容的 `foo` 檔案：

```
<primitive id="R_10.10.10.101" class="ocf" type="IPAddr2"
  provider="heartbeat">
  <instance_attributes id="RA_R_10.10.10.101">
    <attributes>
      <nvpair id="R_ip_P_ip" name="ip" value="10.10.10.101"/>
      <nvpair id="R_ip_P_nic" name="nic" value="eth0"/>
    </attributes>
  </instance_attributes>
</primitive>
```

然後發出以下指令：

```
cibadmin --obj_type resources -U -x foo
```

若要變更先前新增之 IPAddr2 資源的 IP 位址，請發出以下指令：

```
cibadmin -M -X '<nvpair id="R_ip_P_ip" name="ip" value="10.10.10.102"/>'
```

---

## 注

此操作不會將資源名稱變更為與新的 IP 位址相符。若要如此，請刪除資源，然後使用新的 ID 標籤重新新增資源。

---

若要停止 (停用) 先前新增的 IP 位址資源而不將其移除，請建立含有下列內容的檔案 `bar`：

```
<primitive id="R_10.10.10.101">
  <instance_attributes id="RA_R_10.10.10.101">
    <attributes>
      <nvpair id="stop_R_10.10.10.101" name="target-role" value="Stopped"/>
    </attributes>
  </instance_attributes>
</primitive>
```

然後發出以下指令：

```
cibadmin --obj_type resources -U -x bar
```

若要重新啟動上一步所停止的 IP 位址資源，請發出以下指令：

```
cibadmin -D -X '<nvpair id="stop_R_10.10.10.101">'
```

若要從 CIB 徹底移除 IP 位址資源，請發出以下指令：

```
cibadmin -D -X '<primitive id="R_10.10.10.101"/>'
```

若要用手動編輯的新 CIB 版本取代 CIB，請使用以下指令：

```
cibadmin -R -x $HOME/cib.xml
```

## 檔案

`/var/lib/heartbeat/crm/cib.xml` — 磁碟上的 CIB (不含 status 區段)。

## 並請參閱

`crm_resource(8)` [157頁]、`crmadmin(8)` [139頁]、`lrmadmin(8)`、`heartbeat(8)`

## 原著者

`cibadmin` 由 Andrew Beekhof 撰寫。

本手冊頁最初由 Alan Robertson 撰寫。

## 警告

請避免在本地磁碟上自動維護的 CIB 複本上工作。叢集中的任何內容一變更，系統就會更新 CIB。因此，使用 CIB 的過時備份傳播您的組態變更可能會導致叢集不一致。

# crmadmin (8)

crmadmin — 控制叢集資源管理員

## 概要

```
crmadmin [-V|-q] [-i|-d|-K|-S|-E] node
crmadmin [-V|-q] -N -B
crmadmin [-V|-q] -D
crmadmin -v
crmadmin -?
```

## 描述

crmadmin 最初是為了控制 CRM 精靈的大部分動作而設計。不過，由於有了其他工具(如 `crm_attribute` 和 `crm_resource`)，它的大部分功能都已過時。剩餘的功能多數都與 `crmd` 程序的測試和狀態有關。

---

### 警告

有些 `crmadmin` 選項適用於測試，若使用不當，將會產生問題。尤其重要的是，不要使用 `--kill` 或 `--election` 選項，除非您清楚瞭解自己在做什麼。

---

## 選項

```
--help、-?
    列印說明文字。

--version、-v
    列印 HA、CRM 和 CIB 功能集的版本詳細資料。

--verbose、-V
    開啟指令除錯資訊。
```

---

## 注

透過提供其他例項來增加詳細層級。

---

`--quiet`、`-q`

不提供任何除錯資訊，並將輸出縮減至最少。

`--bash-export`、`-B`

建立格式為 `export uname=uuid` 的 `bash` 輸出項目。這僅適用於 `crmdadmin` `-N` 節點指令。

---

## 注

`-B` 功能極少使用，在以後的版本中可能會移除。

---

## 指令

`--debug_inc` 節點、`-i` 節點

在指定的節點上增加 CRM 精靈的除錯層級。將 `USR1` 訊號傳送至 `crmd` 程序也可以實現此目的。

`--debug_dec` 節點、`-d` 節點

在指定的節點上減少 CRM 精靈的除錯層級。將 `USR2` 訊號傳送至 `crmd` 程序也可以實現此目的。

`--kill` 節點、`-K` 節點

關閉指定節點上的 CRM 精靈。

---

## 警告

使用此指令時必須格外小心。此動作通常僅應由 `Heartbeat` 發出，並且可能會產生非預期的副作用。

---

`--status` 節點、`-S` 節點

查詢指定節點上 CRM 精靈的狀態。

輸出包含一般狀態指示器和 `crmd` 程序的內部 FSM 狀態。此資訊可以幫助您判斷叢集正在執行何動作。

`--election` 節點、`-E` 節點  
從指定的節點啟始化選舉。

---

### 警告

使用此指令時必須格外小心。此動作通常從內部啟始化，並且可能會產生非預期的副作用。

---

`--dc_lookup`、`-D`  
查詢目前 DC 的 `uname`。

DC 的位置僅在內部對 `crmd` 有意義，除非要決定在哪個節點上檢查記錄，否則管理員極少使用。

`--nodes`、`-N`  
查詢所有成員節點的 `uname`。查詢的結果可能包含處於「離線」模式的節點。

---

### 注

`-i`、`-d`、`-K` 和 `-E` 選項極少使用，在以後的版本中可能會移除。

---

## 並請參閱

`crm_attribute(8)` [142頁]、`crm_resource(8)` [157頁]

## 原著者

`crmadm` 由 Andrew Beekhof 撰寫。

# crm\_attribute (8)

crm\_attribute — 允許查詢、修改和刪除節點屬性和叢集選項

## 概要

```
crm_attribute [options]
```

## 描述

crm\_attribute 指令可查詢和操作在 CIB 中使用的節點屬性和叢集組態選項。

## 選項

--help、-?  
列印說明訊息。

--verbose、-V  
開啟除錯資訊。

---

### 注

透過提供其他例項來增加詳細層級。

---

--quiet、-Q  
使用 -G 執行屬性查詢時，僅會將值列印至 stdout。請將此選項與 -G 搭配使用。

--get-value、-G  
取回而非設定優先設定。

--delete-attr、-D  
刪除而非設定屬性。

--attr-id 字串、-i 字串  
僅供進階使用者使用。識別 ID 屬性。

- `--attr-value` 字串、`-v` 字串  
要設定的值。與 `-G` 搭配使用時，將忽略此選項。
- `--node` 節點名稱、`-N` 節點名稱  
要變更的節點 `uname`
- `--set-name` 字串、`-s` 字串  
指定屬性集合，即讀取或寫入屬性的所在。
- `--attr-name` 字串、`-n` 字串  
指定要設定或查詢的屬性。
- `--type` 字串、`-t` 類型  
決定應將屬性設定到 **CIB** 的哪個區段，或者所查詢的屬性屬於 **CIB** 的哪個區段。可能值有 `nodes`、`status` 或 `crm_config`。

## 範例

在 **CIB** 中之主機 `myhost` 的 `nodes` 區段中查詢 `location` 屬性的值：

```
crm_attribute -G -t nodes -U myhost -n location
```

在 **CIB** 的 `crm_config` 區段中查詢 `cluster-delay` 屬性的值：

```
crm_attribute -G -t crm_config -n cluster-delay
```

在 **CIB** 的 `crm_config` 區段中查詢 `cluster-delay` 屬性的值。僅列印值：

```
crm_attribute -G -Q -t crm_config -n cluster-delay
```

從 **CIB** 的 `nodes` 區段中刪除主機 `myhost` 的 `location` 屬性：

```
crm_attribute -D -t nodes -U myhost -n location
```

將名為 `location`、值為 `office` 的新屬性新增至 **CIB** 的 `nodes` 區段的 `set` 子區段中 (設定已套用到主機 `myhost`):

```
crm_attribute -t nodes -U myhost -s set -n location -v office
```

變更 `myhost` 主機的 `nodes` 區段中之 `location` 屬性的值：

```
crm_attribute -t nodes -U myhost -n location -v backoffice
```

## 檔案

`/var/lib/heartbeat/crm/cib.xml` — 磁碟上的 CIB (不含 status 區段)。  
強烈建議不要直接編輯此檔案。

## 並請參閱

`cibadmin(8)` [133頁]

## 原著者

`crm_attribute` 由 Andrew Beekhof 撰寫。

# crm\_diff (8)

crm\_diff — 識別對叢集組態的變更並將修補程式套用至組態檔案

## 概要

```
crm_diff [-?|-V] [-o filename] [-O string] [-p filename] [-n filename] [-N string]
```

## 描述

crm\_diff 指令可協助您建立和套用 XML 修補程式。此指令可用於視覺化兩個叢集組態版本之間的變更，或者儲存變更以便在以後使用 cibadmin 進行套用。

## 選項

--help、-?

列印說明訊息。

--original 檔名、-o 檔名

指定要針對其將找出差異或套用修補程式的原始檔案。

--new 檔名、-n 檔名

指定新檔案的名稱。

--original-string 字串、-O 字串

指定要針對其將找出差異或套用修補程式的原始字串。

--new-string 字串、-N 字串

指定新字串。

--patch 檔名、-p 檔名

將修補程式套用至原始 XML。始終與 -o 搭配使用。

`--cib、-c`

將輸入做為 CIB 進行比較或修補。始終使用 `-o` 指定基礎版本，並使用 `-p` 或 `-n` 分別提供修補程式檔案或第二個版本。

`--stdin、-s`

從 `stdin` 讀取輸入。

## 範例

使用 `crm_diff` 可判斷各 CIB 組態檔案之間的差異並建立修補程式。透過修補程式，您可以輕鬆地重複使用組態的各個部分，而不必對每個單一部分使用 `cibadmin` 指令。

- 1 對要比較的兩個叢集設定執行 `cibadmin`，以獲得兩個不同的組態檔案：

```
cibadmin -Q > cib1.xml
cibadmin -Q > cib2.xml
```

- 2 決定是相互進行比較以找出整個檔案的差異，還是僅比較組態的某一部分。

- 3 若要將檔案之間的差異列印至 `stdout`，請使用以下指令：

```
crm_diff -o cib1.xml -n cib2.xml
```

- 4 若要將檔案之間的差異列印至檔案並建立修補程式，請使用以下指令：

```
crm_diff -o cib1.xml -n cib2.xml > patch.xml
```

- 5 將修補程式套用至原始檔案：

```
crm_diff -o cib1.xml -p patch.xml
```

## 檔案

`/var/lib/heartbeat/crm/cib.xml` — 磁碟上的 CIB (不含 `status` 區段)。  
強烈建議不要直接編輯此檔案。

## 並請參閱

`cibadmin(8)` [133頁]

## 原著者

`crm_diff` 由 Andrew Beekhof 撰寫。

# crm\_failcount (8)

crm\_failcount — 管理用於記錄每個資源失敗次數的計數器

## 概要

```
crm_failcount [-?|-V] -D -u|-U node -r resource
```

```
crm_failcount [-?|-V] -G -u|-U node -r resource
```

```
crm_failcount [-?|-V] -v string -u|-U node -r resource
```

## 描述

如果某個資源有可能會在目前節點上失敗，Heartbeat 會實作複雜的方法進行計算，並強制將該資源容錯移轉至其他節點。資源的屬性 `resource-stickiness` 可以決定其要在特定節點上執行的偏好程度。它的另一個屬性 `migration-threshold` 可以決定當達到何限定值時資源應容錯移轉至其他節點。

系統中為資源新增了 `failcount` 屬性，加強了對資源失敗的監控。`failcount` 的值與 `migration-threshold` 的值相乘的結果用來決定此資源的容錯移轉分數。若此數值超過為此資源設定的優先設定值，則資源會移至其他節點，並且在失敗計數重設之前都不會繼續在原始節點上執行。

`crm_failcount` 指令會查詢指定節點上每個資源的失敗次數。此工具也可用於重設失敗計數，以讓資源在之前失敗多次的節點上繼續執行。

## 選項

`--help`、`-?`  
列印說明訊息。

`--verbose`、`-V`  
開啟除錯資訊。

---

## 注

透過提供其他例項來增加詳細層級。

---

- `--quiet`、`-Q`  
使用 `-G` 執行屬性查詢時，僅會將值列印至 `stdout`。請將此選項與 `-G` 搭配使用。
- `--get-value`、`-G`  
取回而非設定優先設定。
- `--delete-attr`、`-D`  
指定要刪除的屬性。
- `--attr-id` 字串、`-i` 字串  
僅供進階使用者使用。識別 ID 屬性。
- `--attr-value` 字串、`-v` 字串  
指定要使用的值。與 `-G` 一起使用時會忽略此選項。
- `--node` 節點 `uname`、`-U` 節點 `uname`  
指定要變更的節點 `uname`。
- `--resource-id` 資源名稱、`-r` 資源名稱  
指定要對其執行作業之資源的名稱。

## 範例

重設節點 `node1` 上資源 `myrsc` 的失敗計數：

```
crm_failcount -D -U node1 -r my_rsc
```

查詢節點 `node1` 上資源 `myrsc` 的目前失敗計數：

```
crm_failcount -G -U node1 -r my_rsc
```

## 檔案

`/var/lib/heartbeat/crm/cib.xml` — 磁碟上的 CIB (不含 status 區段)。  
強烈建議不要直接編輯此檔案。

## 並請參閱

`crm_attribute(8)` [142頁]、`cibadmin(8)` [133頁] 與 Linux High Availability FAQ 網站  
[[http://www.linux-ha.org/v2/faq/forced\\_failover](http://www.linux-ha.org/v2/faq/forced_failover)]

## 原著者

`crm_failcount` 由 Andrew Beekhof 撰寫。

# crm\_master (8)

crm\_master — 管理主要/從屬資源的優先設定，以將其在指定節點上加以提升

## 概要

```
crm_master [-V|-Q] -D [-l lifetime]
crm_master [-V|-Q] -G [-l lifetime]
crm_master [-V|-Q] -v string [-l string]
```

## 描述

crm\_master 是從資源代辦程序檔內部進行呼叫，用來決定應提升為主要模式的資源例項。請勿從指令行中使用該指令，它只是資源代辦的一個輔助公用程式。資源代辦使用 crm\_master 將特定例項提示為主要模式，或從中移除此優先設定。透過指定生命期間，可以決定此設定是應持續到節點重新開機(將生命期間設定為 forever)，或是不應持續到重新開機(將生命期間設定為 reboot)。

資源代辦需要決定 crm\_master 應該對其操作的資源。這些查詢都必須在資源代辦程序檔內部處理。實際呼叫 crm\_master 時使用的語法類似於 crm\_attribute 指令的語法。

## 選項

--help、-?  
列印說明訊息。

--verbose、-V  
開啟除錯資訊。

---

### 注

透過提供其他例項來增加詳細層級。

---

- `--quiet`、`-Q`  
使用 `-G` 執行屬性查詢時，僅會將值列印至 `stdout`。請將此選項與 `-G` 搭配使用。
- `--get-value`、`-G`  
取回要提升的優先設定，而不是對其進行設定。
- `--delete-attr`、`-D`  
刪除而非設定屬性。
- `--attr-id` 字串、`-i` 字串  
僅供進階使用者使用。識別 ID 屬性。
- `--attr-value` 字串、`-v` 字串  
要設定的值。與 `-G` 搭配使用時，將忽略此選項。
- `--lifetime` 字串、`-l` 字串  
指定優先設定持續的時間。可能的值有 `reboot` 或 `forever`。

## 環境變數

`OCF_RESOURCE_INSTANCE` — 資源例項的名稱

## 檔案

`/var/lib/heartbeat/crm/cib.xml` — 磁碟上的 CIB (不含 `status` 區段)。

## 並請參閱

`cibadmin(8)` [133頁]、`crm_attribute(8)` [142頁]

## 原著者

`crm_master` 由 Andrew Beekhof 撰寫。

# crm\_mon (8)

crm\_mon — 監控叢集狀態

## 概要

```
crm_mon [-V] -d -pfilename -h filename
crm_mon [-V] [-l|-n|-r] -h filename
crm_mon [-V] [-n|-r] -X filename
crm_mon [-V] [-n|-r] -c|-l
crm_mon [-V] -i interval
crm_mon -?
```

## 描述

crm\_mon 指令可讓您監控叢集狀態與組態。其輸出包括節點數、uname、uuid、狀態、叢集中設定的資源及其各自的目前狀態。crm\_mon 的輸出可顯示在主控台中或列印成 HTML 檔案。如果提供無狀態區段的叢集組態檔案，crm\_mon 就會建立節點及資源的綜覽 (如檔案中所指定)。

## 選項

--help、-?  
提供說明。

--verbose、-V  
增加除錯輸出。

--interval 秒、-i 秒  
決定更新頻率。如果未指定 -i，則假定為預設值 15 秒。

--group-by-node、-n  
根據節點分組資源。

`--inactive`、`-r`  
顯示非使用中的資源。

`--simple-status`、`-s`  
將叢集狀態做為簡單的單行輸出 (適用於 Nagios) 顯示一次。

`--one-shot`、`-l`  
在主控台上顯示叢集狀態一次，然後離開 (不要使用 `ncurses`)。

`--as-html` 檔案名、`-h` 檔案名  
將叢集狀態寫入指定的檔案。

`--web-cgi`、`-w`  
其輸出適用於 CGI 的 Web 模式。

`--daemonize`、`-d`  
做為精靈在背景中執行。

`--pid-file` 檔案名、`-p` 檔案名  
指定精靈的 pid 檔案。

## 範例

顯示叢集狀態並每隔 15 秒獲取一次更新的清單：

```
crm_mon
```

顯示叢集狀態並在 `-i` 指定的間隔之後獲取更新的清單。如果未指定 `-i`，則假定採用 15 秒的預設重新整理間隔：

```
crm_mon -i interval[s]
```

在主控台上顯示叢集狀態：

```
crm_mon -c
```

在主控台上顯示叢集狀態一次，然後離開：

```
crm_mon -l
```

顯示叢集狀態並按節點分組資源：

```
crm_mon -n
```

顯示叢集狀態、按節點分組資源，並將非使用中的資源併入清單中：

```
crm_mon -n -r
```

將叢集狀態寫入 HTML 檔案：

```
crm_mon -h filename
```

在背景中將 `crm_mon` 做為精靈執行，指定精靈的 `pid` 檔案以便更輕鬆地控制精靈程序，然後建立 HTML 輸出。此選項可讓您不斷建立可透過其他監控應用程式輕鬆處理的 HTML 輸出：

```
crm_mon -d -p filename -h filename
```

顯示現存叢集組態檔案 (檔案名稱) 中配置的叢集組態、按節點分組資源，並包含非使用中的資源。此指令可用於在將叢集組態捲出到使用中的叢集前對其進行預演。

```
crm_mon -r -n -X filename
```

## 檔案

`/var/lib/heartbeat/crm/cib.xml` — 磁碟上的 CIB (不含 `status` 區段)。  
強烈建議不要直接編輯此檔案。

## 原著者

`crm_mon` 由 Andrew Beekhof 撰寫。

# crm\_node (8)

crm\_node — 列出叢集的成員

## 概要

```
crm_node [-V] [-p|-e|-q]
```

## 描述

列出叢集的成員。

## 選項

- V  
詳細說明
- partition、-p  
列印此分割區的成員
- epoch、-e  
列印此節點加入分割區的紀元
- quorum、-q  
如果分割區具有法定數目則列印 1

# crm\_resource (8)

crm\_resource — 執行與叢集資源相關的任務

## 概要

```
crm_resource [-?|-V|-S] -L|-Q|-W|-D|-C|-P|-p [options]
```

## 描述

crm\_resource 指令可在叢集上執行各種與資源相關的動作。它可修改已設定資源的定義，啟動和停止資源以及以及在節點間刪除和移轉資源。

--help、-?  
列印說明訊息。

--verbose、-V  
開啟除錯資訊。

---

### 注

透過提供其他例項來增加詳細層級。

---

--quiet、-Q  
僅將值列印在 stdout 上 (以搭配 -w 使用)。

## 指令

--list、-L  
列出所有資源。

--query-xml、-x  
查詢資源。

必要：-r

`--locate`、`-W`  
尋找資源。

必要：`-r`

`--migrate`、`-M`  
從目前位置移轉資源。使用 `-N` 指定目的地。

如果未指定 `-N`，則系統會透過建立針對目前位置的規則與 `-INFINITY` 分數來強制移動資源。

---

### 注

如此會阻止資源在此節點上執行，直到使用 `-U` 移除限制。

---

必要：`-r`，選擇性：`-N`、`-f`

`--un-migrate`、`-U`  
移除透過 `-M` 建立的所有限制

必要：`-r`

`--delete`、`-D`  
刪除 CIB 中的資源。

必要：`-r`、`-t`

`--cleanup`、`-C`  
從 LRM 刪除資源。

必要：`-r`。選擇性：`-H`

`--reprobe`、`-P`  
重新檢查 CRM 外啟動的資源。

選擇性：`-H`

`--refresh`、`-R`  
從 LRM 重新整理 CIB。

選擇性：-H

--set-parameter 字串、-p 字串  
為資源設定具名參數。

必要：-r、-v。選擇性：-i、-s 和 --meta

--get-parameter 字串、-g 字串  
獲取資源的具名參數。

必要：-r。選擇性：-i、-s 和 --meta

--delete-parameter 字串、-d 字串  
刪除資源的具名參數。

必要：-r。選擇性：-i 和 --meta

--list-operations 字串、-O 字串  
列出作用中的資源作業。按資源、節點或按兩者進行選擇性地過濾。選擇性：-N、-r

--list-all-operations 字串、-o 字串  
列出所有資源作業。按資源、節點或按兩者進行選擇性地過濾。選擇性：-N、-r

## 選項

--resource 字串、-r 字串  
指定資源 ID。

--resource-type 字串、-t 字串  
指定資源類型 (primitive、clone、group 等)。

--property-value 字串、-v 字串  
指定內容值。

--node 字串、-N 字串  
指定主機名稱。

- `--meta`  
修改資源的組態選項，而非使用傳遞至資源代辦程序檔的選項。用於與 `-p`、`-g` 及 `-d` 配合使用。
- `--lifetime` 字串、`-u` 字串  
移轉限制的存留期。
- `--force`、`-f`  
透過建立針對目前位置的規則與 `-INFINITY` 分數來強制移動資源
- 如果資源的黏著率與限制總分超過 `INFINITY` (目前為 100,000)，則應使用此選項。
- 
- 注**
- 如此會阻止資源在此節點上執行，直到使用 `-U` 移除限制。
- 
- `-s` 字串  
(僅供進階使用) 指定要變更的 `instance_attributes` 物件 ID。
- `-i` 字串  
(僅供進階使用) 指定要變更或刪除的 `nvpair` 物件 ID。

## 範例

列出所有資源：

```
crm_resource -L
```

檢查資源的執行位置 (及其是否在執行)：

```
crm_resource -W -r my_first_ip
```

如果 `my_first_ip` 資源正在執行，則此指令的輸出會顯示執行該資源所在的節點。如果該資源未執行，則輸出如此顯示。

啟動或停止資源：

```
crm_resource -r my_first_ip -p target_role -v started  
crm_resource -r my_first_ip -p target_role -v stopped
```

查詢資源的定義：

```
crm_resource -Q -r my_first_ip
```

將資源從其目前位置移轉出來：

```
crm_resource -M -r my_first_ip
```

將資源移轉至指定位置：

```
crm_resource -M -r my_first_ip -H c001n02
```

允許資源返回其正常位置：

```
crm_resource -U -r my_first_ip
```

---

### 注

`resource_stickiness` 與 `default_resource_stickiness` 的值表示資源不會移回。在此情況下，您應使用 `-M` 將資源移回，然後再執行此指令。

---

刪除 CRM 中的資源：

```
crm_resource -D -r my_first_ip -t primitive
```

刪除 CRM 中的資源群組：

```
crm_resource -D -r my_first_group -t group
```

在 CRM 中停用資源的資源管理：

```
crm_resource -p is-managed -r my_first_ip -t primitive -v off
```

在 CRM 中啟用資源的資源管理：

```
crm_resource -p is-managed -r my_first_ip -t primitive -v on
```

手動清理失敗資源後，重設該資源：

```
crm_resource -C -H c001n02 -r my_first_ip
```

重新檢查於 CRM 外啟動之資源的所有節點。

```
crm_resource -P
```

重新檢查於 CRM 外啟動之資源的一個節點。

```
crm_resource -P -H c001n02
```

## 檔案

`/var/lib/heartbeat/crm/cib.xml` — 磁碟上的 CIB (不含 status 區段)。  
強烈建議不要直接編輯此檔案。

## 並請參閱

`cibadmin(8)` [133頁]、`crmadmin(8)` [139頁]、`lrmadmin(8)`、`heartbeat(8)`

## 原著者

`crm_resource` 由 Andrew Beekhof 撰寫。

# crm\_shadow (8)

crm\_shadow — 在更新使用中的叢集之前，在沙箱中執行組態變更

## 概要

```
crm_shadow [-V] [-p|-e|-q]
```

## 描述

設定讓組態工具(cibadmin、crm\_resource等)在其中離線工作而不是針對使用中叢集工作的環境，並允許預覽及測試變更是否有副作用。

## 選項

- `--verbose`、`-V`  
開啟除錯資訊其他例項以增加詳細度
- `--which`、`-w`  
指定使用中的投影副本
- `--display`、`-p`  
顯示投影副本的內容
- `--diff`、`-d`  
顯示投影副本中的變更
- `--create-empty`、`-e 名稱`  
建立叢集組態為空白的具名投影副本
- `--create`、`-c 名稱`  
建立使用中叢集組態的具名投影副本
- `--reset`、`-r 名稱`  
從使用中的叢集組態重新建立具名投影副本

- `--commit`、`-c` *名稱*  
將具名投影副本的內容上載至叢集
- `--delete`、`-d` *名稱*  
刪除具名投影副本的內容
- `--edit`、`-e` *名稱*  
使用您最愛的編輯器編輯具名投影副本的內容
- `--batch`、`-b`  
不啟動新的外圍程序
- `--force`、`-f`  
不啟動新的外圍程序
- `--switch`、`-s`  
切換至具名投影副本

## 內部指令

若要使用投影組態，必須先建立一個投影組態：

```
crm_shadow --create-empty YOUR_NAME
```

它提供一個與 `crm` 工具中的外圍程序類似的內部外圍程序。使用 `help` 取得所有內部指令的綜覽，或者使用 `help` 子指令取得特定指令的資訊。

**表格 16.1** 內部指令綜覽

指令	語法/描述
<code>alias</code>	<code>alias [-p] [name[=value] ... ]</code>  <code>alias</code> 不搭配引數或與 <code>-p</code> 選項搭配將在標準輸出上以別名名稱 = 值的格式列印別名清單。否則，將為提供了「值」的每個「名稱」定義別名。如果「值」的末尾有空格，會導致系統在展開別名時檢查下一個單字是否為別名替代項。除非未為提供的「名稱」定義任何別名，否則 <code>alias</code> 將傳回真。

指令	語法/描述
bg	bg [JOB_SPEC ...]  將每個工作規格放在背景中，就如它是透過 & 啟動一樣。若工作規格不存在，則會使用目前工作的外圍程序概念。
bind	bind [-lpvsPVS] [-m keymap] [-f filename] [-q name] [-u name] [-r keyseq] [-x keyseq:shell-command] [keyseq:readline-function or readline-command]  將關鍵序列與 Readline 函數或某個巨集關聯，或者設定 Readline 變數。無選項的引數語法與 ~/.inputrc 中的語法效果相同，但其必須做為單一引數傳送： <code>bind "\C-x\C-r": re-read-init-file</code> 。
break	break [N]  從某個 for、while 或 until 迴圈中跳離。若指定了 N，則跳離 N 個層級。
builtin	builtin [shell-builtin [arg ...]]  執行外圍程序 builtin。若要重新命名外圍程序 builtin，使之成為函數，但在函數本身中需要 builtin 的功能，此指令就很有幫助。
caller	caller [EXPR]  傳回目前副程式呼叫的網路位置。若不提供運算式，則會傳回 \$line \$filename。若提供了運算式，則會傳回 \$line \$subroutine \$filename；可以使用此額外資訊提供堆疊追蹤。
case	case WORD in [PATTERN [  PATTERN] [COMMANDS;;] ... esac  依據符合模式的單字，選擇性執行指令。「 」用於分隔多個模式。
cd	cd [-L -P] [dir]  將目前目錄切換至 DIR。

指令	語法/描述
command	<pre>command [-pVv] command [arg ...]</pre> <p>搭配忽略外圍程序函數的引數執行指令。若您擁有一名為「ls」的外圍程序函數，且要呼叫指令「ls」，則可以輸入「command ls」。若提供了 -p 選項，則會使用 PATH 的預設值，以確保找到所有標準公用程式。若提供了 -V 或 -v 選項，則會列印描述指令的字串。-V 選項可產生更詳細的描述。</p>
compgen	<pre>compgen [-abcdefgjkusv] [-o option] [-A action]         [-G globpat] [-W wordlist] [-P prefix]         [-S suffix] [-X filterpat] [-F function]         [-C command] [WORD]</pre> <p>依選項而定顯示可能的補齊內容。專供從產生可能的補齊內容的外圍程序函數中使用。若提供選用的單字引數，將會產生與單字相符的項目。</p>
complete	<pre>complete [-abcdefgjkusv] [-pr] [-o option]         [-A action] [-G globpat] [-W wordlist] [-P prefix]         [-S suffix] [-X filterpat] [-F function] [-C command]         [name ...]</pre> <p>對於每個名稱，指定補齊引數的方式。若提供了 -p 選項，或未提供任何選項，將會列印現有補齊規格，且所採用的列印方式可讓其做為輸入重複使用。-r 選項將移除每個名稱的補齊規格；或者，若未提供名稱，將移除所有補齊規格。</p>
continue	<pre>continue [N]</pre> <p>繼續封閉式 FOR、WHILE 或 UNTIL 迴圈的下一個反覆項目。若指定了 N，則會在第 N 層封閉式迴圈繼續。</p>
declare	<pre>declare [-afFirtx] [-p] [name[=value] ...]</pre> <p>宣告變數及/或為變數提供屬性。若未提供名稱，則改為顯示變數的值。-p 選項將顯示每個名稱的屬性和值。</p>
dirs	<pre>dirs [-clpv] [+N] [-N]</pre>

指令	語法/描述
	顯示目前記住的目錄清單。使用 <code>pushd</code> 指令可以將目錄列出在清單上；使用 <code>popd</code> 指令可以返回到清單的上一層。
<code>disown</code>	<p><code>disown [-h] [-ar] [JOBSPEC ...]</code></p> <p>預設會從使用中的工作表格移除每個工作規格引數。若提供了 <code>-h</code> 選項，則不會從表格移除工作，而是標示工作，以便外圍程序接收到 <code>SIGHUP</code> 時不會將 <code>SIGHUP</code> 傳送給該工作。若不提供工作規格，<code>-a</code> 選項表示從工作表格移除所有工作；<code>-r</code> 選項表示僅移除執行中的工作。</p>
<code>echo</code>	<p><code>echo [-neE] [arg ...]</code></p> <p>輸出引數。若指定了 <code>-n</code>，則會隱藏末尾新行字元。若提供了 <code>-e</code> 選項，將會開啟下列以反斜線逸出之字元的解譯：</p> <ul style="list-style-type: none"> <li><code>\a</code> (警示，鈴聲)</li> <li><code>\b</code> (退格)</li> <li><code>\c</code> (隱藏末尾新行)</li> <li><code>\E</code> (逸出字元)</li> <li><code>\f</code> (換頁)</li> <li><code>\n</code> (新行)</li> <li><code>\r</code> (回行)</li> <li><code>\t</code> (水平定位字元)</li> <li><code>\v</code> (垂直定位字元)</li> <li><code>\\</code> (反斜線)</li> <li><code>\0nnn</code> (ASCII 碼為 <code>NNN</code> (八進位) 的字元。<code>NNN</code> 可以是 0 至 3 個八進位數字)</li> </ul> <p>可以使用 <code>-E</code> 選項明確關閉上述字元的解譯。</p>
<code>enable</code>	<p><code>enable [-pnds] [-a] [-f filename] [name...]</code></p> <p>啟用和停用 <code>builtin</code> 外圍程序指令。這可讓您使用與外圍程序 <code>builtin</code> 同名的磁碟指令，而不必指定完整的路徑名稱。若使用了 <code>-n</code>，則會停用名稱；否則會啟用名稱。例如，若要使用 <code>\$PATH</code></p>

---

指令	語法/描述
----	-------

---

中提供的 `test` 而非外圍程序 `builtin` 版本，請輸入 `enable -n test`。在支援動態載入的系統上，可使用 `-f` 選項從共享物件檔名載入新的 `builtin`。`-d` 選項將刪除之前使用 `-f` 載入的 `builtin`。若未提供任何非選項名稱，或者提供了 `-p` 選項，將會列印 `builtin` 清單。`-a` 選項表示列印每個 `builtin`，並會指出其是否已啟用。`-s` 選項將輸出限制為 POSIX.2 的「特殊」`builtin`。`-n` 選項將顯示所有已停用 `builtin` 的清單。

<code>eval</code>	<code>eval [ARG ...]</code>
-------------------	-----------------------------

將引數讀取為外圍程序的輸入並執行產生的指令。

<code>exec</code>	<code>exec [-cl] [-a name] file [redirection ...]</code>
-------------------	--

執行檔案，並以指定的程式取代此外圍程序。若未指定檔案，則會在此外圍程序中進行重新導向。如果第一個引數為 `-l`，則在傳送到檔案的第零個引數中加一個破折號，就如登入一樣。若提供了 `-c` 選項，將會使用空環境執行檔案。`-a` 選項表示將已執行程序的 `argv[0]` 設定為名稱。若檔案無法執行，且外圍程序為非互動型，則除非已設定外圍程序選項 `execfail`，否則外圍程序將會結束。

<code>exit</code>	<code>exit [N]</code>
-------------------	-----------------------

結束外圍程序並傳回狀態 `N`。若省略 `N`，則結束狀態為上次執行之指令的狀態。

<code>export</code>	<code>export [-nf] [NAME[=value] ...]</code> <code>export -p</code>
---------------------	--

將會標示名稱以自動輸出到後續執行的指令的環境。若提供了 `-f` 選項，則名稱指的是函數。若未提供名稱，或者提供了 `-p`，將會列印在此外圍程序中輸出的所有名稱清單。引數 `-n` 表示從後續名稱移除輸出內容。引數 `--` 將會停用進一步的選項處理。

<code>false</code>	<code>false</code>
--------------------	--------------------

指令	語法/描述
	傳回失敗結果。
fc	<pre>fc [-e ename] [-nlr] [FIRST] [LAST] fc -s [pat=rep] [cmd]</pre> <p>fc用於列出或編輯並重新執行歷程清單中的指令。第一個和最後一個可以是指定範圍的數字，或者第一個可以是字串，表示最近使用的以該字串開頭的指令。</p>
fg	<pre>fg [JOB_SPEC]</pre> <p>將工作規格放在前景中，並使其成為目前的工作。若工作規格不存在，將會使用目前工作的外圍程序概念。</p>
for	<pre>for NAME [in WORDS ... ;] do COMMANDS; done</pre> <p>for 迴圈會針對項目清單中的每個成員執行一系列指令。若 in 單字 ...; 不存在，則會假設 in "\$@"。對於單字中的每個元素，會將名稱設定為該元素，然後執行指令。</p>
function	<pre>function NAME { COMMANDS ; } function NAME () { COMMANDS ; }</pre> <p>建立一個簡單指令，該指令透過執行指令的名稱呼叫。指令行上的引數和名稱會做為 \$0 .. \$n 傳送給該函數。</p>
getopts	<pre>getopts OPTSTRING NAME [arg]</pre> <p>外圍程序使用 Getopts 剖析位置參數。</p>
hash	<pre>hash [-lr] [-p PATHNAME] [-dt] [NAME...]</pre> <p>對於每個名稱，將會判斷並記住指令的完整路徑名稱。若提供了 -p 選項，則會將路徑名稱當成名稱的完整路徑名稱，且不會執行路徑搜尋操作。-r 選項會使外圍程序忘記所有記住的位置。-d 選項會使外圍程序忘記記住的每個名稱的位置。若提供了 -t 選項，將會列印每個名稱所對應的完整路徑名稱。若與 -t 一起提供多個名稱引數，將會在雜湊的完整路徑名稱前面列印名稱。</p>

---

指令	語法/描述
----	-------

---

-l 選項會以可做為輸入重複使用的格式顯示輸出。若未提供任何引數，將會顯示有關所記住之指令的資訊。

history	history [-c] [-d OFFSET] [n] history -ps arg [arg...] history -awrm [filename]
---------	--

顯示歷程清單與行號。列出的帶 \* 的行均已修改。引數 *N* 表示僅列出最後 *N* 行。-c 選項會透過刪除所有項目來清除歷程清單。-d 選項將刪除 偏移處 的歷程項目。-w 選項會將目前歷程寫出到歷程檔案；-r 表示讀取檔案並將內容附加至歷程清單。-a 表示將此工作階段的歷程行附加至歷程檔案。引數 -n 表示讀取歷程檔案中尚未讀取的所有歷程行，並將這些行附加至歷程清單。

jobs	jobs [-lnprs] [JOBSPEC ...] job -x COMMAND [ARGS]
------	--

列出使用中的工作。-l 選項將列出程序 ID 和一般資訊；-p 選項僅列出程序 ID。若提供了 -n，將僅列印自從上次通知以來已變更狀態的程序。工作規格將輸出限制為該工作。-r 和 -s 選項分別將輸出限制為執行中的工作和已停止的工作。若不提供任何選項，將會列印所有使用中的工作的狀態。若提供了 -x，則會在將 引數 中出現的所有工作規格取代為該工作之程序群組領袖的程序 ID 後，執行指令。

kill	kill [-s sigspec   -n signum   -sigspec] pid   JOBSPEC ... kill -l [sigspec]
------	---

將訊號 SIGSPEC 傳送至由 PID (或工作規格) 指定的程序。如果 SIGSPEC 不存在，則會假設 SIGTERM。引數 -l 將列出訊號名稱；若多個引數後面跟著 -l，則假設這些引數為應列出名稱之訊號的編號。Kill 是一個外圍程序 builtin，撰寫它有兩個原因：它允許使用工作 ID 取代程序 ID，並且若您已達到可以建立之程序的限制，您不必啟動一個程序來停止另一個程序。

let	let ARG [ARG ...]
-----	-------------------

指令	語法/描述
	每個 <i>引數</i> 都是需要求值的算術運算式。求值是採用固定寬度的整數來完成，雖然除以 0 會被設陷並標誌為錯誤，但是不會檢查溢位。以下運算子清單已根據運算子的優先順序分為不同的幾個層級。各層級以優先順序遞減的順序列出。
local	local NAME[=VALUE] ...  建立名為 <i>名稱</i> 的本地變數，並為其提供相應的 <i>值</i> 。local 僅可在函數中使用；它將變數 <i>名稱</i> 的可見範圍限制為該函數及其子項。
logout	logout  從登入外圍程序登出。
popd	popd [+N   -N] [-n]  從目錄堆疊移除項目。若不提供任何引數，則會從堆疊移除頂層目錄，並切換至新的頂層目錄。
printf	printf [-v var] format [ARGUMENTS]  printf 將使用 <i>格式</i> 來格式化並列印 <i>引數</i> 。 <i>格式</i> 是一個字元字串，它包含三種類型的物件：簡單複製到標準輸出的純字元、會被轉換並複製到標準輸出的字元逸出序列，以及格式規格，每個格式規格都會導致列印下一個後續引數。除標準 printf(1) 格式之外，%b 表示展開對應引數中的反斜線逸出序列，而 %q 表示以可做為外圍程序輸入重複使用的形式來引用引數。若提供了 -v 選項，則會將輸出放置於外圍程序變數 VAR 的值中，而不是將其傳送至標準輸出。
pushd	pushd [dir   +N   -N] [-n]  將目錄新增至目錄堆疊的頂部，或者旋轉堆疊，使堆疊的新頂部成為目前的工作目錄。若不提供任何引數，將會交換最上面的兩個目錄。
pwd	pwd [-LP]

指令	語法/描述
	<p>列印目前的工作目錄。若提供了 <code>-P</code> 選項，<code>pwd</code> 將會列印實體目錄，不列印任何符號連結；<code>-L</code> 選項將在符號連結後面跟著 <code>pwd</code>。</p>
<code>read</code>	<pre>read [-ers] [-u fd] [-t timeout] [-p prompt] [-a array] [-n nchars] [-d delim] [NAME ...]</pre> <p>所提供的 <i>名稱</i> 將標示為唯讀，且後續指定無法變更這些 <i>名稱</i> 的值。若提供了 <code>-f</code> 選項，則會標示名稱所對應的函數。若未提供任何引數，或者提供了 <code>-p</code>，將會列印所有唯讀名稱的清單。<code>-a</code> 選項表示將每個名稱視為陣列變數。引數 <code>--</code> 將停用進一步的選項處理。</p>
<code>readonly</code>	<pre>readonly [-af] [NAME[=VALUE] ...] readonly -p</pre> <p>所提供的 <i>名稱</i> 將標示為唯讀，且後續指定無法變更這些 <i>名稱</i> 的值。若提供了 <code>-f</code> 選項，則會標示名稱所對應的函數。若未提供任何引數，或者提供了 <code>-p</code>，將會列印所有唯讀名稱的清單。<code>-a</code> 選項表示將每個 <i>名稱</i> 視為陣列變數。引數 <code>--</code> 將停用進一步的選項處理。</p>
<code>return</code>	<pre>return [N]</pre> <p>導致函數結束，並傳回由 <i>N</i> 指定的傳回值。若省略 <i>N</i>，則傳回上次執行之指令的狀態。</p>
<code>select</code>	<pre>select NAME [in WORDS ... ;] do COMMANDS; done</pre> <p>將展開 <i>單字</i>，並產生單字清單。將在標準錯誤上列印展開的單字集，且每個單字前面都有一個數字。若 <i>in</i> 單字不存在，則會假設 <i>in</i> "\$@"。然後顯示 PS3 提示，並從標準輸入讀取一行。若該行由其中一個顯示的單字所對應的數字組成，則會將名稱設定為該單字。若該行為空，將會重新顯示 <i>單字</i> 和提示。若讀取到 EOF，指令即會完成。讀取到任何其他值均會導致將名稱設定為空值。所讀取的行會儲存在變數 <i>回覆</i> 中。每次選取後即會執行指令，直到執行跳離指令為止。</p>

指令	語法/描述
set	<pre>set [--abefhkmnptuvxBCHP] [-o OPTION] [ARG...]</pre> <p>設定內部外圍程序選項。</p>
shift	<pre>shift [n]</pre> <p>將 \$N+1 ... 中的位置參數重新命名為 \$1 ...。若未提供 N，則將其假設為 1。</p>
shopt	<pre>shopt [-pqsu] [-o long-option] OPTNAME [OPTNAME...]</pre> <p>切換控制選擇性行為的變數的值。-s 旗標表示啟用(設定) 每個選項名稱；-u 旗標將取消設定每個選項名稱。-q 旗標將隱藏輸出；離開狀態指出是否已設定或取消設定每個選項名稱。-o 選項將選項名稱限制為定義為與 set -o 搭配使用的那些選項名稱。若不提供任何選項，或者提供了 -p 選項，將會顯示所有可設定選項的清單，並會指出是否已設定每個選項。</p>
source	<pre>source FILENAME [ARGS]</pre> <p>讀取並執行檔名中的指令，然後返回。\$PATH 中的路徑名稱用於尋找含有檔名的目錄。若提供了引數，則在執行檔名時，引數將做為位置參數。</p>
suspend	<pre>suspend [-f]</pre> <p>暫停執行此外圍程序，直到它接收到 SIGCONT 訊號為止。若指定了 -f，則表示如果此外圍程序為登入外圍程序也不發出警告；只是將其暫停。</p>
test	<pre>test [expr]</pre> <p>根據運算式的求值結果結束，並傳回狀態 0 (真) 或 1 (偽)。運算式可為一元運算式或二元運算式。一元運算式通常用於檢查檔案的狀態。系統提供了字串運算子和數字比較運算子。</p>
time	<pre>time [-p] PIPELINE</pre>

---

指令	語法/描述
----	-------

---

執行管線，並在它終止時列印執行管線所花的實際時間、使用者 CPU 時間和系統 CPU 時間的摘要。傳回狀態為「管線」的傳回狀態。`-p` 選項將以略微不同的格式列印所花時間的摘要。此選項使用 `TIMEFORMAT` 變數的值做為輸出格式。

<code>times</code>	<code>times</code>
--------------------	--------------------

列印從外圍程序執行之程序的累計使用者和系統時間。

<code>trap</code>	<code>trap [-lp] [ARG SIGNAL_SPEC ...]</code>
-------------------	---

外圍程序接收到訊號訊號規格時，將讀取並執行指令引數。若引數不存在(提供了單一訊號規格)或`-`，則會將每個指定的訊號重設為其原始值。若引數為空字串，則外圍程序以及它呼叫的指令將會忽略每個訊號規格。若訊號規格為`EXIT(0)`，則會在結束外圍程序時執行指令「引數」。若訊號規格為`DEBUG`，則會在每個簡單指令之後執行引數。若提供了`-p`選項，則會顯示與每個訊號規格相關的設陷指令。若未提供任何引數，或者僅提供了`-p`，設陷將會列印與每個訊號相關的指令清單。每個訊號規格都是 `signal.h` 中的一個訊號名稱或訊號編號。訊號名稱不區分大小寫，且 `SIG` 字首為選用項目。`trap -l` 將列印訊號名稱及其對應編號的清單。請注意，您可以使用 `kill -signal $$` 將訊號傳送至外圍程序。

<code>true</code>	<code>true</code>
-------------------	-------------------

傳回成功結果。

<code>type</code>	<code>type [-afptP] NAME [NAME ...]</code>
-------------------	--

過時，請參閱`declare`。

<code>typeset</code>	<code>typeset [-afFirtx] [-p] name[=value]</code>
----------------------	---

過時，請參閱`declare`。

<code>ulimit</code>	<code>ulimit [-SHacdfilmpqstuvx] [limit</code>
---------------------	--

指令	語法/描述
	<code>ulimit</code> 可讓您對由外圍程序啟動的程序可以使用的資源進行控制 (在允許此類控制的系統上)。
<code>umask</code>	<pre>umask [-p] [-S] [MODE]</pre> <p>將使用者的檔案建立遮罩設定為模式。若省略了模式，或提供了 <code>-S</code>，將會列印遮罩的目前值。若提供了 <code>-S</code> 選項，則會輸出符號；否則會輸出八進位數字。若提供了 <code>-p</code> 且省略了模式，則輸出時採用可以用做輸入的形式。若模式以數字開頭，則會將其解譯為八進位數字，否則將其解譯為 <code>chmod(1)</code> 所接受的符號模式字串。</p>
<code>unalias</code>	<pre>unalias [-a] NAME [NAME ...]</pre> <p>從定義的別名清單移除名稱。若提供了 <code>-a</code> 選項，將會移除所有的別名定義。</p>
<code>unset</code>	<pre>unset [-f] [-v] [NAME ...]</pre> <p>對於每個名稱，移除對應的變數或函數。若提供了 <code>-v</code>，則取消設定僅對變數起作用。若提供了 <code>-f</code> 旗標，則取消設定僅對函數起作用。若未提供任何旗標，則取消設定將先嘗試取消設定變數；如果嘗試失敗，則會嘗試取消設定函數。某些變數無法取消設定，也視為唯讀。</p>
<code>until</code>	<pre>until COMMANDS; do COMMANDS; done</pre> <p>只要 <code>until</code> 指令中的最後一個指令的離開狀態不是零，就展開並執行指令。</p>
<code>wait</code>	<pre>wait [N]</pre> <p>等待指定的程序並報告其終止狀態。若未提供 <code>N</code>，將會等待目前使用中的所有子程序，且傳回碼為零。<code>N</code> 可以是程序 ID 或工作規格；若提供了工作規格，將會等待工作管線中的所有程序。</p>
<code>while</code>	<pre>while COMMANDS; do COMMANDS; done</pre>

---

指令	語法/描述
	只要 <code>while</code> 指令中最後一個指令的離開狀態為零，就展開並執行指令。

---

# crm\_standby (8)

crm\_standby — 操作節點的待機屬性，以判斷資源是否可以在此節點上執行。

## 概要

```
crm_standby [-?|-V] -D -u|-U node -r resource
crm_standby [-?|-V] -G -u|-U node -r resource
crm_standby [-?|-V] -v string -u|-U node -r resource [-l string]
```

## 描述

crm\_standby 指令可操作節點的待機屬性。待機模式下的任何節點都無法再代管資源，且其中的所有資源也都必須移出。對於執行核心更新等維護任務，待機模式十分有用。當節點再次成為叢集的完全使用中成員時，請從節點移除待機屬性。

為 standby 屬性指定存留期，即可決定待機設定在節點重新開機後應繼續存在 (將存留期設定為 forever) 還是應經重新開機進行重設 (將存留期設定為 reboot)。也可以移除 standby 屬性，然後手動將節點從待機模式返回到使用中模式。

## 選項

--help、-?  
列印說明訊息。

--verbose、-V  
開啟除錯資訊。

---

### 注

透過提供其他例項來增加詳細層級。

---

- `--quiet`、`-Q`  
使用 `-G` 執行屬性查詢時，僅會將值列印至 `stdout`。請將此選項與 `-G` 搭配使用。
- `--get-value`、`-G`  
取回而非設定優先設定。
- `--delete-attr`、`-D`  
指定要刪除的屬性。
- `--attr-value` 字串、`-v` 字串  
指定要使用的值。與 `-G` 一起使用時會忽略此選項。
- `--attr-id` 字串、`-i` 字串  
僅供進階使用者使用。識別 ID 屬性。
- `--node` 節點 `uname`、`-u` 節點 `uname`  
指定要變更的節點 `uname`。
- `--lifetime` 字串、`-l` 字串  
判斷此優先設定的持續時間。可能的值有 `reboot` 或 `forever`。

---

### 注

如果存在 `forever` 值，CRM 會一直使用該值取代任何 `reboot` 值。

---

## 範例

讓本地節點進入待機模式：

```
crm_standby -v true
```

讓節點 (`node1`) 進入待機模式：

```
crm_standby -v true -U node1
```

查詢節點的待機狀態：

```
crm_standby -G -U node1
```

從節點移除待機內容：

```
crm_standby -D -U node1
```

讓節點無限期進入待機模式：

```
crm_standby -v true -l forever -U node1
```

讓節點進入待機模式，直到此節點下次重新開機為止：

```
crm_standby -v true -l reboot -U node1
```

## 檔案

`/var/lib/heartbeat/crm/cib.xml` — 磁碟上的 CIB (不含 status 區段)。  
強烈建議不要直接編輯此檔案。

## 並請參閱

`cibadmin(8)` [133頁], `crm_attribute(8)` [142頁]

## 原著者

`crm_standby` 由 Andrew Beekhof 撰寫。

# crm\_verify (8)

crm\_verify — 檢查 CIB 的一致性

## 概要

```
crm_verify [-V] -x file
crm_verify [-V] -X string
crm_verify [-V] -L|-p
crm_verify [-?]
```

## 描述

crm\_verify 可檢查組態資料庫 (CIB) 的一致性及其他問題。該指令可用於檢查包含組態的檔案，或者它是否可以連接到執行中的叢集。它將報告兩類問題：錯誤和警告。必須先修復錯誤才能讓 Heartbeat 正常工作。不過，可由管理員決定是否應修復警告。

crm\_verify 可協助建立新組態或修改的組態。您可在執行中的叢集中建立 CIB 的本地副本、對其進行編輯，並使用 crm\_verify 進行驗證，然後使用 cibadmin 使新組態生效。

## 選項

--help、-h  
列印說明訊息。

--verbose、-V  
開啟除錯資訊。

---

### 注

透過提供其他例項來增加詳細層級。

---

- `--live-check`、`-L`  
連接到執行中的叢集並檢查 CIB。
- `--crm_xml` 字串、`-X` 字串  
檢查提供的字串中的組態。僅會傳遞完整的 CIB。
- `--xml-file` 檔案、`-x` 檔案  
檢查具名檔案中的組態。
- `--xml-pipe`、`-p`  
使用透過 `stdin` 傳送的組態。僅會傳遞完整的 CIB。

## 範例

檢查執行中叢集中組態的一致性並產生詳細的輸出：

```
crm_verify -VL
```

檢查指定檔案中組態的一致性並產生詳細的輸出：

```
crm_verify -Vx file1
```

將組態傳送至 `crm_verify` 並產生詳細的輸出：

```
cat file1.xml | crm_verify -Vp
```

## 檔案

`/var/lib/heartbeat/crm/cib.xml` — 磁碟上的 CIB (不含 `status` 區段)。  
強烈建議不要直接編輯此檔案。

## 並請參閱

`cibadmin(8)` [133頁]

## 原著者

`crm_verify` 由 Andrew Beekhof 撰寫。



## 叢集資源

本章概述了與叢集資源相關的最重要的資料與統計數字：High Availability Extension 支援的資源代辦類別、OCF 資源代辦的錯誤碼及叢集對錯誤碼的反應方式、可用的資源選項、資源作業及例項屬性。設定資源時 (可使用 `crm` 指令行工具或 Linux HA Management Client 進行手動設定)，可參考本綜覽。

### 17.1 受支援的資源代辦類別

對於每個新增的叢集資源，您都需要定義資源代辦所遵循的標準。資源代辦會提取提供的服務並向叢集提供準確的狀態，這樣叢集便可不用知道其所管理的資源。當接收到 `start`、`stop` 或 `monitor` 指令時，叢集依賴於資源代辦來執行恰當的操作。

通常資源代辦的形式為外圍程序檔。High Availability Extension 支援以下類別的資源代辦：

#### 舊版 Heartbeat 1 資源代辦

Heartbeat 版本 1 具有自己的資源代辦樣式。由於很多人已基於其慣例撰寫了自己的代辦，所以這些資源代辦仍受支援。但是，仍建議您在可能的情況下將您的組態移轉至 High Availability OCF RA。如需詳細資訊，請參閱 <http://wiki.linux-ha.org/HeartbeatResourceAgent>。

#### Linux Standards Base (LSB) 程序檔

LSB 資源代辦通常由作業系統/套裝作業系統提供，位於 `/etc/init.d`。若要與叢集一起使用，他們必須符合 LSB 規格。例如，它們必須執行幾個動作，這些動作至少包含 `start`、`stop`、`restart`、`reload`、

force-reload 和 status, 如 [http://www.linuxbase.org/spec/refspecs/LSB\\_3.0.0/LSB-Core-generic/LSB-Core-generic/iniscriptact.html](http://www.linuxbase.org/spec/refspecs/LSB_3.0.0/LSB-Core-generic/LSB-Core-generic/iniscriptact.html) 中所述。

#### 開放叢集架構 (OCF) 資源代辦

OCF RA 代辦最適合與 High Availability 搭配使用, 特別是在您需要主要資源或特殊監控功能的情況下。代辦通常位於 `/usr/lib/ocf/resource.d/heartbeat/`。它們的功能類似於 LSB 程序檔。但是, 組態始終使用環境變數來進行設定, 這樣可讓其輕鬆接受並處理參數。OCF 規格 (與資源代辦相關時) 位於 [http://www.opencf.org/cgi-bin/viewcvs.cgi/specs/ra/resource-agent-api.txt?rev=HEAD\\_4](http://www.opencf.org/cgi-bin/viewcvs.cgi/specs/ra/resource-agent-api.txt?rev=HEAD_4)。OCF 規格對於動作必須返回的離開碼有著嚴格的定義。叢集完全遵循這些規格。如需詳細資訊, 請參閱 <http://wiki.linux-ha.org/OCFResourceAgent>。如需所有可用 OCF RA 的詳細清單, 請參閱第 18 章「*HA OCF 代辦*」[191頁]。

#### STONITH 資源代辦

此類別專用於圍籬區隔相關資源。如需詳細資訊, 請參閱第 8 章「*圍籬區隔與 STONITH*」[75頁]。

提供給 High Availability Extension 的代辦將寫入 OCF 規格。

## 17.2 OCF 傳回代碼

根據 OCF 規格, 對於動作必須返回的離開碼有著嚴格的定義。叢集會始終根據預期結果檢查傳回代碼。如果結果不符合預期值, 則該作業將被視為失敗, 並啟動復原動作。失敗復原有三種類型:

表格 17.1 失敗復原類型

復原類型	描述	叢集採取的動作
軟式	發生暫時錯誤。	重新啟動資源或將其移到新的位置。
硬式	發生非暫時錯誤。該錯誤為目前節點所特有。	將資源移到別處並阻止其在目前節點上被重試。

復原類型	描述	叢集採取的動作
嚴重錯誤	發生對於所有叢集節點均相同的非暫時錯誤。這表示指定了錯誤組態。	停止資源並阻止其在任何叢集節點上啟動。

假設一個動作被視為已失敗，下表概述了不同的 OCF 傳回代碼以及接收到相應的錯誤碼時將啟動的叢集復原類型。

**表格 17.2** OCF 傳回代碼

OCF 傳回代碼	OCF 別名	描述	復原類型
0	OCF_SUCCESS	成功。指令成功完成。這是所有 start、stop、promote 和 demote 指令的預期結果。	軟式
1	OCF_ERR_GENERIC	一般「發生問題」錯誤碼。	軟式
2	OCF_ERR_ARGS	此機器上的資源組態無效 (例如，它參考節點上找不到的位置/工具)。	硬式
3	OCF_ERR_UNIMPLEMENTED	未執行所需動作。	硬式
4	OCF_ERR_PERM	資源代辦沒有足夠的權限完成任務。	硬式
5	OCF_ERR_INSTALLED	此機器上未安裝資源所需的工具。	硬式
6	OCF_ERR_CONFIGURED	資源的組態無效 (例如，所需參數遺失)。	嚴重錯誤

OCF 傳回代碼	OCF 別名	描述	復原類型
7	OCF_NOT_RUNNING	資源未執行。叢集不會嘗試停止對任何動作返回此代碼的資源。  此 OCF 傳回代碼可能需要也可能不需要資源復原 — 具體取決於預期的資源狀態。如果不是預期情況，則進行軟式復原。	無
8	OCF_RUNNING_MASTER	資源正以主要模式執行。	軟式
9	OCF_FAILED_MASTER	資源在主要模式下執行，但已失敗。資源將被降級、停止，然後再次啟動 (還可能升級)。	軟式
其他	無	自定錯誤碼。	軟式

## 17.3 資源選項

對於新增的每個資源，您都可以定義選項。叢集使用選項來決定資源的行為方式 — 它們會告知 CRM 如何處理特定資源。可透過 `crm_resource --meta` 指令或文中提及的 GUI 來設定資源選項。

**表格 17.3** 原始資源選項

選項	描述
<code>priority</code>	如果並非所有資源都可啟動，則叢集將停止優先程度較低的資源，以便讓優先程度較高的資源啟動。

選項	描述
target-role	叢集應嘗試將此資源保持什麼狀態？允許的值：Stopped、Started。
is-managed	叢集是否允許啟動及停止資源？允許的值：true、false。
resource-stickiness	資源希望留在原處的程度如何？預設值為 default-resource-stickiness。
migration-threshold	此資源在節點上的失敗次數達到多少次才會取消該節點代管此資源的資格？預設值：none。
multiple-active	如果叢集發現資源在多個節點上處於使用中狀態，叢集應如何處理？允許的值：block (將資源標示為不受管理)、stop_only、stop_start。
failure-timeout	等待多少秒之後才會當做失敗未發生 (潛在含義為允許資源回到其失敗的節點上)？預設值：never。

## 17.4 資源作業

預設叢集不會確保資源仍正常。若要指示叢集執行此操作，您需要將監控作業新增至資源定義。可為所有類別或資源代辦新增監控作業。

**表格 17.4** 資源作業

操作	描述
ID	動作執行人名稱。必須唯一。
name	執行的動作。通用值：monitor、start、stop。

操作	描述
<code>interval</code>	執行作業的頻率。單位：秒。
<code>timeout</code>	宣告動作失敗之前等待多長時間？
<code>requires</code>	需滿足什麼條件才會發生此動作。允許的值： <code>nothing</code> 、 <code>quorum</code> 、 <code>fencing</code> 。預設值取決於圍籬區隔是否啟用及資源類別是否為 <code>stonith</code> 。對於 <code>STONITH</code> 資源，預設值為 <code>nothing</code> 。
<code>on-fail</code>	此動作失敗時所採取的動作。允許的值： <ul style="list-style-type: none"> <li>• <code>ignore</code>：當做資源未失敗。</li> <li>• <code>block</code>：在資源上不執行任何進一步作業。</li> <li>• <code>stop</code>：停止資源且不在其他任何地方啟動。</li> <li>• <code>restart</code>：停止資源並再次啟動(可能在其他節點上)。</li> <li>• <code>fence</code>：將發生資源失敗的節點關閉 (<code>STONITH</code>)。</li> <li>• <code>standby</code>：將發生資源失敗之節點上的所有資源移出。</li> </ul>
<code>enabled</code>	如果為 <code>false</code> ，則會將作業視做不存在。允許的值： <code>true</code> 、 <code>false</code> 。

## 17.5 例項屬性

所有資源類別的程序檔均可接收參數，參數決定了資源的行為方式及其控制的服務例項。如果您的資源代辦支援參數，則可使用文中提及的 `crm_resource` 指令新增參數。在 `crm` 指令行公用程式中，例項屬性稱為 `params`。您可以 `root` 身份執行以下指令來獲取受 OCF 程序檔支援的例項屬性清單：

```
crm ra meta resource_agent class
```

例如

```
crm ra meta Ipaddr ocf heartbeat
```

顯示



# 18

## HA OCF 代辦

所有 OCF 代辦都要求一些參數在其啟動時已經設定好。以下綜覽顯示如何手動操作這些代辦。此附錄中可用的資料是透過調用相應資源代辦的 `meta-data` 直接取得。`/usr/lib/ocf/resource.d/heartbeat/` 中提供了以下所有代辦。

在設定資源代辦時，請忽略參數名稱的字首 `OCF_RESKEY_`。進行組態設定時可以忽略方括號中的參數。

# ocf:anything\_ra (7)

ocf:anything\_ra — 一切物件

## 概要

```
OCF_RESKEY_binfile=string [OCF_RESKEY_cmdline_options=string]
[OCF_RESKEY_pidfile=string] [OCF_RESKEY_logfile=string]
[OCF_RESKEY_errlogfile=string] [OCF_RESKEY_user=string]
[OCF_RESKEY_monitor_hook=string] anything_ra [start | stop | monitor |
meta-data | validate-all]
```

## 描述

此為用於管理幾乎一切物件的一般 OCF 資源代辦。

## 支援的參數

OCF\_RESKEY\_binfile=要執行之二進位的完整路徑名稱  
要執行之二進位的完整名稱。對其的預期是持續使用同一個 pid 執行，而不是在完成某項任務後就結束。

OCF\_RESKEY\_cmdline\_options=指令行選項  
要傳遞至二進位的指令行選項

OCF\_RESKEY\_pidfile=要將 STDOUT 寫入其中的檔案  
要從中讀取 PID 或將 PID 寫入其中的檔案。

OCF\_RESKEY\_logfile=要將 STDOUT 寫入其中的檔案  
要將 STDOUT 寫入其中的檔案

OCF\_RESKEY\_errlogfile=要將 STDERR 寫入其中的檔案  
要將 STDERR 寫入其中的檔案

OCF\_RESKEY\_user=用於執行指令的使用者身份  
用於執行指令的使用者身份

OCF\_RESKEY\_monitor\_hook=在監控作業中執行的指令  
在監控作業中執行的指令

# ocf:apache (7)

ocf:apache — Apache Web 伺服器

## 概要

```
OCF_RESKEY_configfile=string [OCF_RESKEY_httpd=string]
[OCF_RESKEY_port=integer] [OCF_RESKEY_statusurl=string]
[OCF_RESKEY_testregex=string] [OCF_RESKEY_options=string]
[OCF_RESKEY_envfiles=string] apache [start | stop | status | monitor | meta-data
| validate-all]
```

## 描述

此為 Apache Web 伺服器的資源代辦。此資源代辦可操作 1.x 與 2.x 兩種版本的 Apache 伺服器。啟動作業以迴路結束，在該迴路中會反覆呼叫監控作業，以確認伺服器已啟動且可正常運作。因此，若監控作業在啟動作業逾時之前未成功執行，則 `apache` 資源將以錯誤狀態結束。監控作業預設會載入伺服器狀態頁，此頁面依 `mod_status` 模組與對應的組態檔案 (通常為 `/etc/apache2/mod_status.conf`) 而定。請確認伺服器狀態頁能正常工作，且 \*僅\* 允許從 `localhost` (位址 127.0.0.1) 存取該頁。如需詳細資料，請參閱 `statusurl` 與 `testregex` 屬性，並請參閱 <http://httpd.apache.org/>

## 支援的參數

`OCF_RESKEY_configfile=組態檔案路徑`  
Apache 組態檔案的完整路徑名稱。系統會剖析此檔案以便為其他各資源代辦參數提供預設值。

`OCF_RESKEY_httpd=httpd 二進位路徑`  
`httpd` 二進位的完整路徑名稱 (選擇性)。

`OCF_RESKEY_port=httpd 埠`  
可對其進行查探以透過 `statusurl` 獲取狀態資訊的埠號。其預設值為在組態檔案中找到的埠號，若組態檔案中沒有埠號，則為 80。

OCF\_RESKEY\_statusurl=url 名稱

要監控的 URL (預設為 apache 伺服器狀態頁)。若未指定，則從 apache 組態檔案推斷該值。若設定此 URL，請確認 \*僅\* 可從 localhost (127.0.0.1) 存取該 URL。否則，叢集可能會發出多個節點上正在使用該資源的警告。

OCF\_RESKEY\_testregex=監控正規表示式

用來在 statusurl 的輸出中進行比對的正規表示式，不區分大小寫。

OCF\_RESKEY\_options=指令行選項

啟動 apache 時要套用的額外選項。請參閱man httpd(8)。

OCF\_RESKEY\_envfiles=環境設定檔案

內含其他環境變數的 (一或多個) 檔案，如 /etc/apache2/envvars。

# ocf:AudibleAlarm (7)

ocf:AudibleAlarm — AudibleAlarm 資源代辦

## 概要

```
[OCF_RESKEY_nodelist=string] AudibleAlarm [start | stop | restart | status |  
monitor | meta-data | validate-all]
```

## 描述

AudibleAlarm 的資源程序檔。它會將聲訊鬧鐘設定為在設定的間隔發出嗶聲。

## 支援的參數

OCF\_RESKEY\_nodelist=節點清單  
永遠不應發出警示的節點清單。

# ocf:ClusterMon (7)

ocf:ClusterMon — ClusterMon 資源代辦

## 概要

```
[OCF_RESKEY_user=string] [OCF_RESKEY_update=integer]  
[OCF_RESKEY_extra_options=string] OCF_RESKEY_pidfile=string  
OCF_RESKEY_htmlfile=string ClusterMon [start | stop | monitor | meta-data |  
validate-all]
```

## 描述

此為 ClusterMon 資源代辦。它會將目前的叢集狀態輸出至 html。

## 支援的參數

OCF\_RESKEY\_user=希望用於執行 `crm_mon` 的使用者身份  
希望用於執行 `crm_mon` 的使用者份

OCF\_RESKEY\_update=更新間隔  
更新叢集狀態的頻率

OCF\_RESKEY\_extra\_options=其他選項  
傳遞至 `crm_mon` 的其他選項。例如, `-n -r`

OCF\_RESKEY\_pidfile=PID 檔案  
確保僅有一個例項在執行的 PID 檔案位置

OCF\_RESKEY\_htmlfile=HTML 輸出  
將 HTML 輸出寫入到其中的位置。

# ocf:db2 (7)

ocf:db2 — db2 資源代辦

## 概要

```
[OCF_RESKEY_instance=string] [OCF_RESKEY_admin=string] db2 [start | stop  
| status | monitor | validate-all | meta-data | methods]
```

## 描述

db2的資源程序檔。它會將DB2 Universal Database 例項做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_instance=例項  
資料庫的例項。

OCF\_RESKEY\_admin=管理員  
例項的管理員使用者。

# ocf:Delay (7)

ocf:Delay — 延遲資源代辦

## 概要

```
[OCF_RESKEY_startdelay=integer] [OCF_RESKEY_stopdelay=integer]  
[OCF_RESKEY_mondelay=integer] Delay [start | stop | status | monitor | meta-data  
| validate-all]
```

## 描述

此程序檔是用於引入延遲的測試資源。

## 支援的參數

OCF\_RESKEY\_startdelay=啟動延遲  
啟動作業延遲的時間 (以秒計)。

OCF\_RESKEY\_stopdelay=停止延遲  
停止作業延遲的時間 (以秒計)。若未指定，則預設為「startdelay」。

OCF\_RESKEY\_mondelay=監控延遲  
監控作業延遲的時間 (以秒計)。若未指定，則預設為「startdelay」。

# ocf:drbd (7)

ocf:drbd — 此資源代辦會將分散式複製設備區塊 (DRBD) 物件做為主要/從屬資源進行管理。DRBD 是用於複製儲存的一種機制；如需設定的詳細資料，請參閱相關文件。

## 概要

```
OCF_RESKEY_drbd_resource=string [OCF_RESKEY_drbdconf=string]
[OCF_RESKEY_clone_overrides_hostname=boolean]
[OCF_RESKEY_clone_max=integer] [OCF_RESKEY_clone_node_max=integer]
[OCF_RESKEY_master_max=integer]
[OCF_RESKEY_master_node_max=integer] drbd [start | promote | demote | notify
| stop | monitor | monitor | meta-data | validate-all]
```

## 描述

DRBD 的主要/從屬 OCF 資源代辦

## 支援的參數

OCF\_RESKEY\_drbd\_resource=drbd 資源名稱  
drbd.conf 檔案中之 drbd 資源的名稱。

OCF\_RESKEY\_drbdconf=drbd.conf 的路徑  
drbd.conf 檔案的完整路徑。

OCF\_RESKEY\_clone\_overrides\_hostname=覆寫 drbd 主機名稱  
是否以複製品編號覆寫主機名稱。此參數可用於建立浮動對等組態；系統會要求 drbd 使用 node\_<複製品編號> 而非實際名稱做為主機名稱，然後再將其用於 drbd.conf 中。

OCF\_RESKEY\_clone\_max=複製品數  
此 drbd 資源的複製品數。請勿隨意修改預設值。

OCF\_RESKEY\_clone\_node\_max=節點數  
每個節點的複製品數。請勿隨意修改預設值。

OCF\_RESKEY\_master\_max=主複製品數  
使用中主複製品的最大數目。請勿隨意修改預設值。

OCF\_RESKEY\_master\_node\_max=每個節點的主複製品數  
每個節點上主複製品的最大數目。請勿隨意修改預設值。

# ocf:Dummy (7)

ocf:Dummy — 虛構資源代辦

## 概要

OCF\_RESKEY\_state=string Dummy [start | stop | monitor | reload | migrate\_to | migrate\_from | meta-data | validate-all]

## 描述

此為虛構資源代辦。它只會追蹤自己是否在執行，除此之外完全不執行其他作業。它的使命就是用於測試或做為資源代辦撰寫器的範本。

## 支援的參數

OCF\_RESKEY\_state=狀態檔案  
儲存資源狀態的位置。

# ocf:eDir88 (7)

ocf:eDir88 — eDirectory 資源代辦

## 概要

```
OCF_RESKEY_eDir_config_file=string  
[OCF_RESKEY_eDir_monitor_ldap=boolean]  
[OCF_RESKEY_eDir_monitor_idm=boolean]  
[OCF_RESKEY_eDir_jvm_initial_heap=integer]  
[OCF_RESKEY_eDir_jvm_max_heap=integer]  
[OCF_RESKEY_eDir_jvm_options=string] eDir88 [start | stop | monitor | meta-  
data | validate-all]
```

## 描述

用於管理 eDirectory 例項的資源程序檔，它會將 eDirectory 的單一例項做為 HA 資源進行管理。eDirectory 的「多個例項」功能已新增至 8.8 版。此程序檔不適用於任何 8.8 版之前的 eDirectory 版本。此資源代辦可用於在同一部主機上載入多個 eDirectory 例項。強烈建議您將 eDir 組態檔案 (依據 eDir\_config\_file 參數) 放置於每個節點上的本地儲存中。這是此資源代辦處理共享儲存不可用的情況時所必需的。若 eDir 組態檔案不可用，則此資源代辦將失敗，並且 Heartbeat 將無法管理資源。其副作用包括 STONITH 動作、無法管理的資源等。強烈建議您設定較高的動作逾時值。含有 IDM 的 eDir 啟動時需要 10 多分鐘的時間。若 Heartbeat 在 eDir 尚未正常啟動之前就逾時，則 mayhem\_WILL ENSUE。LDAP 模組會是最後啟動的一個。因此，若啟用了對 IDM 和/或 LDAP 的監控，而安裝中含有 IDM 與 LDAP 時，程序檔需要更長的時間才能啟動，因為啟動指令將等候 IDM 和 LDAP 變為可用狀態。

## 支援的參數

OCF\_RESKEY\_eDir\_config\_file=eDir 組態檔案  
eDirectory 例項的組態檔案路徑。

OCF\_RESKEY\_eDir\_monitor\_ldap=eDir 監控 ldap  
系統應該監控 LDAP 是否在為 eDirectory 例項執行嗎?

OCF\_RESKEY\_eDir\_monitor\_idm=eDir 監控 IDM  
系統應該監控 IDM 是否在為 eDirectory 例項執行嗎?

OCF\_RESKEY\_eDir\_jvm\_initial\_heap=DHOST\_INITIAL\_HEAP 值  
java 環境變數 DHOST\_INITIAL\_HEAP 的值。若未設定，將使用 java 預設值。

OCF\_RESKEY\_eDir\_jvm\_max\_heap=DHOST\_MAX\_HEAP 值  
java 環境變數 DHOST\_MAX\_HEAP 的值。若未設定，將使用 java 預設值。

OCF\_RESKEY\_eDir\_jvm\_options=DHOST\_OPTIONS 值  
java 環境變數 DHOST\_OPTIONS 的值。若未設定，將使用原始值。

# ocf:Filesystem (7)

ocf:Filesystem — 檔案系統資源代辦

## 概要

```
[OCF_RESKEY_device=string] [OCF_RESKEY_directory=string]  
[OCF_RESKEY_fstype=string] [OCF_RESKEY_options=string] Filesystem  
[start | stop | notify | monitor | validate-all | meta-data]
```

## 描述

檔案系統的資源程序檔。它會管理共享儲存媒體上的檔案系統。

## 支援的參數

OCF\_RESKEY\_device=區塊設備

檔案系統的區塊設備名稱、用於裝載的 -U、-L 選項，或 NFS 裝載規格。

OCF\_RESKEY\_directory=裝載點

檔案系統的裝載點。

OCF\_RESKEY\_fstype=檔案系統類型

要裝載之檔案系統的選擇性類型。

OCF\_RESKEY\_options=選項

將指定為 -o 選項的任何其他要裝載的選項。對於結合裝載，請在此處新增「bind」，並將 fstype 設定為「none」。系統將會對「bind,ro」這類選項執行相應作業。

# ocf:ICP (7)

ocf:ICP — ICP 資源代辦

## 概要

```
[OCF_RESKEY_driveid=string][OCF_RESKEY_device=string] ICP [start | stop  
| status | monitor | validate-all | meta-data]
```

## 描述

ICP 的資源程序檔。它會將 ICP Vortex 叢集化的主機磁碟機做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_driveid=ICP 叢集磁碟機 ID  
ICP 叢集磁碟機 ID。

OCF\_RESKEY\_device=設備  
設備名稱。

# ocf:ids (7)

ocf:ids — 稱為 Informix Dynamic Server (IDS) 之 IBM 資料庫伺服器的 OCF 資源代辦

## 概要

```
[OCF_RESKEY_informixdir=string] [OCF_RESKEY_informixserver=string]
[OCF_RESKEY_onconfig=string] [OCF_RESKEY_dbname=string]
[OCF_RESKEY_sqltestquery=string] ids [start | stop | status | monitor | validate-
all | meta-data | methods | usage]
```

## 描述

將 IBM Informix Dynamic Server (IDS) 例項做為 High Availability 資源進行管理的 OCF 資源代辦。

## 支援的參數

OCF\_RESKEY\_informixdir= INFORMIXDIR 環境變數

當您進行 IDS 一般安裝後，環境變數 INFORMIXDIR 所獲得的值。也就是 IDS 的安裝路徑(不含末尾斜線「/」)。若未指定此參數，則程序檔將嘗試從外圍程序環境獲取該值。

OCF\_RESKEY\_informixserver= INFORMIXSERVER 環境變數

當您進行 IDS 一般安裝後，環境變數 INFORMIXSERVER 所獲得的值。也就是要管理之 IDS 伺服器例項的名稱。若未指定此參數，則程序檔將嘗試從外圍程序環境獲取該值。

OCF\_RESKEY\_onconfig= ONCONFIG 環境變數

當您進行 IDS 一般安裝後，環境變數 ONCONFIG 所獲得的值。也就是 INFORMIXSERVER 中指定之 IDS 例項的組態檔案名稱。系統將在「/etc/」處搜尋指定的組態檔案。若未指定此參數，則程序檔將嘗試從外圍程序環境獲取該值。

OCF\_RESKEY\_dbname= 用於監控的資料庫，預設為「sysmaster」  
此參數定義使用哪一個資料庫來監控 IDS 例項。若未指定此參數，程序檔將使用預設的「sysmaster」資料庫。

OCF\_RESKEY\_sqltestquery= 用於監控的 SQL 測試查詢，預設為「SELECT COUNT(\*) FROM systables;」  
在參數「dbname」指定的資料庫中執行，以便監控 IDS 例項並判斷其是否正常運作的 SQL 測試查詢。若未指定此參數，程序檔將使用預設的「SELECT COUNT(\*) FROM systables;」。

# ocf:IPaddr2 (7)

ocf:IPaddr2 — 管理虛擬 IPv4 位址

## 概要

```
OCF_RESKEY_ip=string [OCF_RESKEY_nic=string]
[OCF_RESKEY_cidr_netmask=string] [OCF_RESKEY_broadcast=string]
[OCF_RESKEY_iflabel=string] [OCF_RESKEY_lvs_support=boolean]
[OCF_RESKEY_mac=string] [OCF_RESKEY_clusterip_hash=string]
[OCF_RESKEY_unique_clone_address=boolean]
[OCF_RESKEY_arp_interval=integer] [OCF_RESKEY_arp_count=integer]
[OCF_RESKEY_arp_bg=string] [OCF_RESKEY_arp_mac=string] IPaddr2 [start
| stop | status | monitor | meta-data | validate-all]
```

## 描述

這個 Linux 特定的資源會管理 IP 別名 IP 位址。它可新增或移除 IP 別名。此外，若將其做為複製品資源叫用，則其可實作叢集別名 IP 功能。

## 支援的參數

OCF\_RESKEY\_ip=IPv4 位址

要以點分四組數字表示法設定的 IPv4 位址，例如「192.168.1.1」。

OCF\_RESKEY\_nic=網路介面

可讓 IP 位址處於線上的基礎網路介面。若留為空白，程序檔將嘗試透過路由表確定此介面。請勿以 eth0:1 的格式指定別名介面或任何其他值；請僅指定基礎介面。

OCF\_RESKEY\_cidr\_netmask=CIDR 網路遮罩

CIDR 格式的介面網路遮罩 (例如 24，而不是 255.255.255.0)。若未指定，程序檔也將嘗試透過路由表確定此介面。

- OCF\_RESKEY\_broadcast=廣播位址  
與 IP 關聯的廣播位址。若留為空白，程序檔將透過網路遮罩確定此位址。
- OCF\_RESKEY\_iflabel=介面標籤  
您可在為 IP 位址指定其他標籤。此標籤會附加至您的介面名稱。若是在 NIC 名稱中指定標籤，則此參數即無效。
- OCF\_RESKEY\_lvs\_support=啟用 LVS DR 支援  
啟用 LVS 直接路由組態支援。若 IP 位址被停止，只會將其移至迴路設備，以允許本地節點繼續申請服務，但不會在網路上對其進行通告。
- OCF\_RESKEY\_mac=叢集 IP MAC 位址  
明確設定介面 MAC 位址。目前僅用於叢集 IP 別名。請留為空白以自動選擇。
- OCF\_RESKEY\_clusterip\_hash=叢集 IP 雜湊功能  
指定叢集 IP 功能所用的雜湊演算法。
- OCF\_RESKEY\_unique\_clone\_address=為複製例項建立唯一的位址  
若為 true，則將複製品 ID 新增至提供的 IP 值，以建立將要管理的唯一位址。
- OCF\_RESKEY\_arp\_interval=ARP 封包間隔 (以毫秒計)  
指定不明 ARP 封包之間的間隔 (以毫秒計)。
- OCF\_RESKEY\_arp\_count=ARP 封包計數  
要傳送的不明 ARP 封包數。
- OCF\_RESKEY\_arp\_bg=背景中的 ARP  
是否要傳送背景中的 ARP 封包。
- OCF\_RESKEY\_arp\_mac=ARP MAC  
要將 ARP 封包傳送至的 MAC 位址。請勿變更此值。

# ocf:IPaddr (7)

ocf:IPaddr — 管理虛擬 IPv4 位址

## 概要

```
OCF_RESKEY_ip=string [OCF_RESKEY_nic=string]
[OCF_RESKEY_cidr_netmask=string] [OCF_RESKEY_broadcast=string]
[OCF_RESKEY_iflabel=string] [OCF_RESKEY_lvs_support=boolean]
[OCF_RESKEY_local_stop_script=string]
[OCF_RESKEY_local_start_script=string]
[OCF_RESKEY_ARP_INTERVAL_MS=integer]
[OCF_RESKEY_ARP_REPEAT=integer]
[OCF_RESKEY_ARP_BACKGROUND=boolean]
[OCF_RESKEY_ARP_NETMASK=string] IPaddr [start | stop | monitor | validate-all
| meta-data]
```

## 描述

此程序檔會管理 IP 別名 IP 位址。它可新增或移除 IP 別名。

## 支援的參數

OCF\_RESKEY\_ip=IPv4 位址

要以點分四組數字表示法設定的 IPv4 位址，例如「192.168.1.1」。

OCF\_RESKEY\_nic=網路介面

可讓 IP 位址處於線上的基礎網路介面。若留為空白，程序檔將嘗試透過路由表確定此介面。請勿以 eth0:1 的格式指定別名介面或任何其他值；請僅指定基礎介面。

OCF\_RESKEY\_cidr\_netmask=網路遮罩

CIDR 格式的介面網路遮罩。(例如 24)，或使用點分四組數字表示法 (255.255.255.0)。若未指定，程序檔也將嘗試透過路由表確定此遮罩。

- OCF\_RESKEY\_broadcast=廣播位址  
與 IP 關聯的廣播位址。若留為空白，程序檔將透過網路遮罩確定此位址。
- OCF\_RESKEY\_iflabel=介面標籤  
您可在此為 IP 位址指定其他標籤。
- OCF\_RESKEY\_lvs\_support=啟用 LVS DR 支援  
啟用 LVS 直接路由組態支援。若 IP 位址被停止，只會將其移至迴路設備，  
以允許本地節點繼續申請服務，但不會在網路上對其進行通告。
- OCF\_RESKEY\_local\_stop\_script=釋放 IP 時呼叫的程序檔  
釋放 IP 時呼叫的程序檔。
- OCF\_RESKEY\_local\_start\_script=新增 IP 時呼叫的程序檔  
新增 IP 時呼叫的程序檔。
- OCF\_RESKEY\_arp\_interval\_ms=免費 ARP 之間的時間隔 (以毫秒計)  
ARP 之間的時間隔 (以毫秒計)。
- OCF\_RESKEY\_arp\_repeat=重複計數  
產生新位址時要傳送的免費 ARP 數目。
- OCF\_RESKEY\_arp\_background=在背景中執行  
在背景中執行 (無需任何理由)。
- OCF\_RESKEY\_arp\_netmask=ARP 的網路遮罩  
非標準十六進位格式的 ARP 網路遮罩。

# ocf:IPsrcaddr (7)

ocf:IPsrcaddr — IPsrcaddr 資源代辦

## 概要

```
[OCF_RESKEY_ipaddress=string] IPsrcaddr [start | stop | stop | monitor |  
validate-all | meta-data]
```

## 描述

IPsrcaddr 的資源程序檔。它會管理偏好的來源位址修改。

## 支援的參數

OCF\_RESKEY\_ipaddress=IP 位址  
IP 位址。

# ocf:IPv6addr (7)

ocf:IPv6addr — 管理 IPv6 別名

## 概要

[OCF\_RESKEY\_ipv6addr=string] IPv6addr [start | stop | status | monitor | validate-all | meta-data]

## 描述

此程序檔會管理 IPv6 別名 IPv6 位址。它可新增或移除 IPv6 別名。

## 支援的參數

OCF\_RESKEY\_ipv6addr=IPv6 位址  
此資源代辦將要管理的 IPv6 位址。

# ocf:iscsi (7)

ocf:iscsi — iscsi 資源代辦

## 概要

```
[OCF_RESKEY_portal=string] OCF_RESKEY_target=string  
[OCF_RESKEY_discovery_type=string] [OCF_RESKEY_iscsiadm=string]  
[OCF_RESKEY_udev=string] iscsi [start | stop | status | monitor | validate-all |  
methods | meta-data]
```

## 描述

iSCSI 的 OCF 資源代辦。新增 (start) 或移除 (stop) iSCSI 目標。

## 支援的參數

OCF\_RESKEY\_portal=入口網站  
格式為 {ip\_address|hostname}[:"port] 的 iSCSI 入口網站位址。

OCF\_RESKEY\_target=目標  
iSCSI 目標。

OCF\_RESKEY\_discovery\_type=discovery\_type  
探查類型。目前 open-iscsi 僅支援 sendtargets 類型。

OCF\_RESKEY\_iscsiadm=iscsiadm  
iscsiadm 程式路徑。

OCF\_RESKEY\_udev=udev  
若下一個資源視建立設備的 udev 而定，則系統將等待其完成作業。在負載正常的主機上，此過程完成得應該較快，但也有例外。若您未在使用 udev，請將此參數設定為「no」，否則系統將一直在迴路中重複，直到發生逾時。

# ocf:Ldirectord (7)

ocf:Ldirectord — ldirectord 的包裝程式 OCF 資源代辦

## 概要

```
OCF_RESKEY_configfile=string [OCF_RESKEY_ldirectord=string]  
Ldirectord [start | stop | monitor | meta-data | validate-all]
```

## 描述

這是 ldirectord 的簡單 OCF 資源代辦包裝程式，它會使用 ldirectord 介面建立與 OCF 相容的介面，而您則可以對 ldirectord 進行監控。警告：查詢 ldirectord 狀態是非常耗時的動作。

## 支援的參數

OCF\_RESKEY\_configfile=組態檔案路徑  
ldirectord 組態檔案的完整路徑名稱。

OCF\_RESKEY\_ldirectord=ldirectord 二進位路徑  
ldirectord 的完整路徑名稱。

# ocf:LinuxSCSI (7)

ocf:LinuxSCSI — LinuxSCSI 資源代辦

## 概要

```
[OCF_RESKEY_scsi=string] LinuxSCSI [start | stop | methods | status | monitor |  
meta-data | validate-all]
```

## 描述

此為 LinuxSCSI 的資源代辦。它會從 linux 核心的角度來管理 SCSI 設備的可用性。它可將設備對 Linux 隱藏起來，也可將其重新對 Linux 可見。

## 支援的參數

OCF\_RESKEY\_scsi=SCSI 例項  
要管理的 SCSI 例項。

# ocf:LVM (7)

ocf:LVM — LVM 資源代辦

## 概要

```
[OCF_RESKEY_volgrpname=string] LVM [start | stop | status | monitor | methods |  
meta-data | validate-all]
```

## 描述

LVM 的資源程序檔。它會將 Linux Volume Manager (LVM) 磁碟區做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_volgrpname=磁碟區群組名稱  
磁碟區群組的名稱。

# ocf:MailTo (7)

ocf:MailTo — MailTo 資源代辦

## 概要

[OCF\_RESKEY\_email=**string**] [OCF\_RESKEY\_subject=**string**] MailTo [start | stop | status | monitor | meta-data | validate-all]

## 描述

此為 MailTo 的資源代辦。每當有接管動作發生時，它都會向系統管理員傳送電子郵件。

## 支援的參數

OCF\_RESKEY\_email=電子郵件地址  
系統管理員的電子郵件地址。

OCF\_RESKEY\_subject=主旨  
電子郵件的主旨。

# ocf:ManageRAID (7)

ocf:ManageRAID — 管理 RAID 設備

## 概要

```
[OCF_RESKEY_raidname=string] ManageRAID [start | stop | status | monitor |  
validate-all | meta-data]
```

## 描述

管理 RAID 設備的啟動、停止與監控，這些已在 `/etc/conf.d/HB-ManageRAID` 中預先設定。

## 支援的參數

`OCF_RESKEY_raidname=RAID` 名稱  
要管理之 RAID 的名稱 (區分大小寫)。(已在 `/etc/conf.d/HB-ManageRAID` 中預先設定)。

# ocf:ManageVE (7)

ocf:ManageVE — OpenVZ VE 資源代辦

## 概要

```
[OCF_RESKEY_veid=integer] ManageVE [start | stop | status | monitor | validate-all  
| meta-data]
```

## 描述

與 OCF 相容的此資源代辦會管理 OpenVZ VE，因此需要安裝適當的 OpenVZ，其中要包含最新的 vzctl util。

## 支援的參數

OCF\_RESKEY\_veid=VE 的 OpenVZ ID  
    虛擬環境的 OpenVZ ID (請檢視所有指定 ID 的 vzlist -a 輸出)。

# ocf:mysql (7)

ocf:mysql — MySQL 資源代辦

## 概要

```
[OCF_RESKEY_binary=string] [OCF_RESKEY_config=string]
[OCF_RESKEY_datadir=string] [OCF_RESKEY_user=string]
[OCF_RESKEY_group=string] [OCF_RESKEY_log=string]
[OCF_RESKEY_pid=string] [OCF_RESKEY_socket=string]
[OCF_RESKEY_test_table=string] [OCF_RESKEY_test_user=string]
[OCF_RESKEY_test_passwd=string]
[OCF_RESKEY_enable_creation=integer]
[OCF_RESKEY_additional_parameters=integer] mysql [start | stop | status
| monitor | validate-all | meta-data]
```

## 描述

MySQL 的資源程序檔。它會將 MySQL 資料庫例項做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_binary=MySQL 二進位  
MySQL 二進位的位置。

OCF\_RESKEY\_config=MySQL 組態  
組態檔案。

OCF\_RESKEY\_datadir=MySQL 資料目錄  
包含資料庫的目錄。

OCF\_RESKEY\_user=MySQL 使用者  
執行 MySQL 精靈的使用者。

- OCF\_RESKEY\_group=MySQL 群組  
執行 MySQL 精靈的群組 (針對記錄檔案與目錄許可權限)。
- OCF\_RESKEY\_log=MySQL 記錄檔案  
用於 mysqld 的記錄檔案。
- OCF\_RESKEY\_pid=MySQL pid 檔案  
用於 mysqld 的 pid 檔案。
- OCF\_RESKEY\_socket=MySQL 插槽  
用於 mysqld 的插槽。
- OCF\_RESKEY\_test\_table=MySQL 測試表  
監控陳述式中要測試的表 (以資料庫.表來表示)。
- OCF\_RESKEY\_test\_user=MySQL 測試使用者  
MySQL 測試使用者。
- OCF\_RESKEY\_test\_passwd=MySQL 測試使用者密碼  
MySQL 測試使用者密碼。
- OCF\_RESKEY\_enable\_creation=若資料庫不存在，則建立  
若 MySQL 資料庫不存在，將建立該資料庫。
- OCF\_RESKEY\_additional\_parameters=要傳遞至 mysqld 的其他參數  
在啟動時要傳遞給 mysqld 的其他參數。(例如 --skip-external-locking 或 --skip-grant-tables)。

# ocf:nfsserver (7)

ocf:nfsserver — nfsserver

## 概要

```
[OCF_RESKEY_nfs_init_script=string]
[OCF_RESKEY_nfs_notify_cmd=string]
[OCF_RESKEY_nfs_shared_infodir=string][OCF_RESKEY_nfs_ip=string]
nfsserver [start | stop | monitor | meta-data | validate-all]
```

## 描述

Nfsserver 可協助您將 Linux nfs 伺服器做為 Linux-HA 中的可容錯移轉資源進行管理。它依賴於 Linux 特定的 NFS 實作詳細資料，因此尚未考慮在其他平台上對其進行支援。

## 支援的參數

OCF\_RESKEY\_nfs\_init\_script= nfsserver 的 Init 程序檔  
Linux 發行套件隨附的預設 init 程序檔。nfsserver 資源代辦會將啟動/停止/監控工作分流至 init 程序檔，因為啟動/停止/監控 nfsserver 的程序會因 Linux 發行套件的不同而不同。

OCF\_RESKEY\_nfs\_notify\_cmd= 用於傳送通知的工具。  
用於傳送 NSM 重新開機通知的工具。可以將 nfsserver 的容錯移轉視為重新開機到其他機器。當發生容錯移轉時，nfsserver 資源代辦會使用此指令來通知所有用戶端。

OCF\_RESKEY\_nfs\_shared\_infodir= 儲存 nfs 伺服器相關資訊的目錄。  
nfsserver 資源代辦將 nfs 相關資訊儲存在這個特定的目錄中。此目錄必須可在 nfsserver 之前進行容錯移轉。

OCF\_RESKEY\_nfs\_ip= IP 位址。  
可用於存取 nfs 服務的浮動 IP 位址。

# ocf:oracle (7)

ocf:oracle — oracle 資源代辦

## 概要

```
OCF_RESKEY_sid=string [OCF_RESKEY_home=string]
[OCF_RESKEY_user=string] [OCF_RESKEY_ipcrm=string]
[OCF_RESKEY_clear_backupmode=boolean]
[OCF_RESKEY_shutdown_method=string] oracle [start | stop | status | monitor
| validate-all | methods | meta-data]
```

## 描述

oracle 的資源程序檔，它會將 Oracle 資料庫例項做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_sid=sid  
Oracle SID (aka ORACLE\_SID)。

OCF\_RESKEY\_home=主目錄  
Oracle 主目錄 (也稱為 ORACLE\_HOME)。若未指定，則 SID 及其主目錄應列於 /etc/oratab 中。

OCF\_RESKEY\_user=使用者  
Oracle 擁有者 (也稱為 ORACLE\_OWNER)。若未指定，則會將其設定為檔案 \$ORACLE\_HOME/dbs/\*\${ORACLE\_SID}.ora 的擁有者。若系統未將其設定為該值，只需明確設定該值即可。

OCF\_RESKEY\_ipcrm=ipcrm  
有時，屬於 Oracle 例項的 IPC 物件 (共享記憶體節區與信號) 可能會滯後，這將妨礙例項的啟動。要判斷哪些共享節區屬於哪個例項並不容易，當多個例項以同一個使用者身份執行時尤其如此。為此，系統中使用了「oradeflag」功能及其「ipc」追蹤公用程式。剖析除錯資訊並不是最佳方式，但目前沒

有其他更佳的方法來獲取有關 IPC 的資訊。當追蹤報告的格式或文字發生變更時，剖析可能便會失敗。但有一些預防措施可以避免出錯。另外還有一個 `dumpinstipc` 選項，可讓系統列印屬於該例項的 IPC 物件。可以使用它來判斷系統是否正確剖析了追蹤檔案。它的設定值有三種：- `none`：不與 IPC 相關，結果好壞全憑運氣(請注意：您遲早會有運氣不佳的時候)- `instance`：嘗試判斷屬於例項的 IPC 物件，並僅移除這些物件(預設值；應該安全)- `orauser`：移除屬於執行該例項之使用者的所有 IPC (若您以同一個使用者身份執行多個例項，或以這個使用者身份執行的其他應用程式使用了 IPC，則請勿使用此設定值)。使用預設設定值「`instance`」應該是安全的，但我們不能保證在該情況下例項一定會啟動。若因為(例如)有人惡意停止 Oracle 程序而導致 IPC 物件被丟棄，那麼將沒有任何其他的方法可以判斷應該移除哪些 IPC 物件。在該情況下，有必要進行人為干預，可能還必須停止以同一個使用者身份執行的所有例項。第三個設定值「`orauser`」可保證將 IPC 物件移除，但這必須是以 IPC 物件擁有權為基礎，因此您只能在每個例項都以不同的使用者身份執行時才可使用該設定值。如有任何問題，請向我們報告。歡迎您提供建議或解決方案。

`OCF_RESKEY_clear_backupmode=clear_backupmode`  
清除 ORACLE 的備份模式。

`OCF_RESKEY_shutdown_method=shutdown_method`  
如何停止 Oracle 似乎是一個習慣問題。預設方法(「`checkpoint/abort`」)為：變更系統檢查點；關閉中止；這是關閉例項最為快捷、安全的方法。如果您發現「`shutdown abort`」效果不佳，請將此屬性設定為「`immediate`」，這樣就可以立即關閉 Oracle；如果您仍然認為可以有更好的方法來關閉 Oracle 例項，我們很樂意傾聽您的意見。

# ocf:oralsnr (7)

ocf:oralsnr — oralsnr 資源代辦

## 概要

```
OCF_RESKEY_sid=string [OCF_RESKEY_home=string]
[OCF_RESKEY_user=string] OCF_RESKEY_listener=string oralsnr [start |
stop | status | monitor | validate-all | meta-data | methods]
```

## 描述

Oracle 監聽程式的資源程序檔。它會將 Oracle 監聽程式例項做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_sid=sid  
Oracle SID (aka ORACLE\_SID)。監控作業需要此 SID，即在執行 `tnsping SID` 時需要此 SID。

OCF\_RESKEY\_home=主目錄  
Oracle 主目錄 (也稱為 ORACLE\_HOME)。若未指定，則應在 `/etc/oratab` 中列出該 SID。

OCF\_RESKEY\_user=使用者  
以此使用者身份執行監聽程式。

OCF\_RESKEY\_listener=監聽程式  
要啟動的監聽程式例項 (如 `listener.ora` 中所定義)。預設值為 `LISTENER`。

# ocf:pgsql (7)

ocf:pgsql — postgresql 資源代辦

## 概要

```
[OCF_RESKEY_pgctl=string] [OCF_RESKEY_start_opt=string]
[OCF_RESKEY_ctl_opt=string] [OCF_RESKEY_psql=string]
[OCF_RESKEY_pgdata=string] [OCF_RESKEY_pgdba=string]
[OCF_RESKEY_pghost=string] [OCF_RESKEY_pgport=string]
[OCF_RESKEY_pgdb=string] [OCF_RESKEY_logfile=string]
[OCF_RESKEY_stop_escalate=string] postgresql [start | stop | status | monitor |
meta-data | validate-all | methods]
```

## 描述

PostgreSQL 的資源程序檔。它會將 PostgreSQL 做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_pgctl=pgctl  
pg\_ctl 指令的路徑。

OCF\_RESKEY\_start\_opt=start\_opt  
啟動選項 (pgi\_ctl 中的 -o start\_opt)。例如「-i -p 5432」。

OCF\_RESKEY\_ctl\_opt=ctl\_opt  
其他 pg\_ctl 選項 (-w、-W 等)。預設為 ""

OCF\_RESKEY\_psql=psql  
psql 指令的路徑。

OCF\_RESKEY\_pgdata=pgdata  
PostgreSQL 資料目錄的路徑。

`OCF_RESKEY_pgdba=pgdba`  
擁有 PostgreSQL 的使用者。

`OCF_RESKEY_pghost=pghost`  
PostgreSQL 在其上進行監聽的主機名稱/IP 位址

`OCF_RESKEY_pgport=pgport`  
PostgreSQL 在其上進行監聽的埠

`OCF_RESKEY_pgdb=pgdb`  
將用於監控的資料庫。

`OCF_RESKEY_logfile=logfile`  
PostgreSQL 伺服器記錄輸出檔案的路徑。

`OCF_RESKEY_stop_escalate=停止升級`  
求助於 `-m immediate` 之前的重試次數 (使用 `-m fast`)

# ocf:pingd (7)

ocf:pingd — pingd 資源代辦

## 概要

```
[OCF_RESKEY_pidfile=string] [OCF_RESKEY_user=string]
[OCF_RESKEY_dampen=integer] [OCF_RESKEY_set=integer]
[OCF_RESKEY_name=integer] [OCF_RESKEY_section=integer]
[OCF_RESKEY_multiplier=integer] [OCF_RESKEY_host_list=integer]
pingd [start | stop | monitor | meta-data | validate-all]
```

## 描述

此為 pingd 資源代辦。它會記錄 (在 CIB 中) 某個節點目前可連接的 ping 節點數。

## 支援的參數

OCF\_RESKEY\_pidfile=PID 檔案  
PID 檔案

OCF\_RESKEY\_user=用於執行 pingd 的使用者身份  
用於執行 pingd 的使用者身份。

OCF\_RESKEY\_dampen=抑制間隔  
等待 (抑制) 進一步變更發生的時間。

OCF\_RESKEY\_set=集名稱  
用於放置值之例項屬性集的名稱，很少需要指定。

OCF\_RESKEY\_name=屬性名稱  
要設定之屬性的名稱。此為限制中要使用的名稱。

OCF\_RESKEY\_section=區段名稱  
放置值的區段。很少需要指定。

OCF\_RESKEY\_multiplier=值的倍數  
要用連接的 ping 節點數與其相乘的數值。

OCF\_RESKEY\_host\_list=主機清單  
要計數之 ping 節點的清單。預設為所有設定的 ping 節點。很少需要指定。

# ocf:portblock (7)

ocf:portblock — portblock 資源代辦

## 概要

```
[OCF_RESKEY_protocol=string] [OCF_RESKEY_portno=integer]  
[OCF_RESKEY_action=string] portblock [start | stop | status | monitor | meta-  
data | validate-all]
```

## 描述

portblock 的資源程序檔，可透過使用 iptable 來暫時封鎖連接埠。

## 支援的參數

OCF\_RESKEY\_protocol=通訊協定  
曾被封鎖/解除封鎖的通訊協定。

OCF\_RESKEY\_portno=埠號碼  
曾被封鎖/解除封鎖的埠號碼。

OCF\_RESKEY\_action=動作  
要對通訊協定::埠號碼執行的動作 (封鎖/解除封鎖)。

# ocf:Pure-FTPd (7)

ocf:Pure-FTPd — 與 OCF 資源代辦相容的 FTP 程序檔。

## 概要

```
OCF_RESKEY_script=string OCF_RESKEY_conffile=string  
OCF_RESKEY_daemon_type=string [OCF_RESKEY_pidfile=string]  
Pure-FTPd [start | stop | monitor | validate-all | meta-data]
```

## 描述

此程序檔會管理主動-被動設定中的 Pure-FTPd。

## 支援的參數

- OCF\_5RESKEY\_script=包含完整路徑的程序檔名稱  
Pure-FTPd 啟動程序檔的完整路徑。例如「/sbin/pure-config.pl」。
- OCF\_RESKEY\_conffile=包含完整路徑的組態檔案名稱  
包含完整路徑的 Pure-FTPd 組態檔案名稱。例如「/etc/pure-ftp/pure-ftp.conf」。
- OCF\_RESKEY\_daemon\_type=包含完整路徑的組態檔案名稱  
Pure-ftp-wrapper 將要呼叫的 Pure-FTPd 精靈。有效選項為：用於 Pure-ftp、  
「mysql」用於 pure-ftp-mysql、「postgresql」用於 pure-ftp-postgresql、  
「ldap」用於 pure-ftp-ldap。
- OCF\_RESKEY\_pidfile=PID 檔案  
PID 檔案

# ocf:Raid1 (7)

ocf:Raid1 — RAID1 資源代辦

## 概要

```
[OCF_RESKEY_raidconf=string] [OCF_RESKEY_raiddev=string]  
[OCF_RESKEY_homehost=string] Raid1 [start | stop | status | monitor | validate-  
all | meta-data]
```

## 描述

RAID1 的資源程序檔。它會管理共享儲存媒體上的軟體 Raid1 設備。

## 支援的參數

OCF\_RESKEY\_raidconf=**RAID** 組態檔案  
RAID 組態檔案。例如 /etc/raidtab or /etc/mdadm.conf。

OCF\_RESKEY\_raiddev=**區塊設備**  
要使用的區塊設備。

OCF\_RESKEY\_homehost=**mdadm** 的主機名稱  
主機指示詞的值；這是一個 mdadm 功能，可保護 RAID 不會被意外啟動。  
建議您建立透過叢集管理的 RAID，並將「homehost」設定為特殊值，以免  
不擁有或不支援這些 RAID 的節點意外將 RAID 自動組合。

# ocf:Route (7)

ocf:Route — 管理網路路由

## 概要

```
OCF_RESKEY_destination=string OCF_RESKEY_device=string  
OCF_RESKEY_gateway=string OCF_RESKEY_source=string Route [start | stop  
| monitor | reload | meta-data | validate-all]
```

## 描述

啟用和停用網路路由。支援主機及網路路由、透過閘道位址的路由，以及使用特定來源位址的路由。如果需要根據節點的角色指定來操作節點的路由表格，該資源代辦將非常有用。考慮以下範例使用案例：- 將一個叢集節點做為 IPsec 通道端點。- 所有其他的節點均使用 IPsec 通道連接特定遠端網路中的主機。接下來是您應如何利用 Route 資源代辦實作此規劃的說明：- 設定 ipsec LSB 資源。- 設定複製的 Route OCF 資源。- 建立順序限制，以確保在 Route 之前啟動 ipsec。- 建立 ipsec 與 Route 資源之間的並存限制，以確保沒有任何複製的 Route 資源例項會在通道端點上自行啟動。

## 支援的參數

OCF\_RESKEY\_destination=目的網路  
要為路由設定的目的網路 (或主機)。以 CIDR 表示法指定網路遮罩字尾 (例如「/24」)。若未指定字尾，系統將會建立主機路由。若要將此資源設定為系統預設路由，請指定「0.0.0.0/0」或「default」。

OCF\_RESKEY\_device=外送網路設備  
要為此路由使用的外送網路設備。

OCF\_RESKEY\_gateway=閘道 IP 位址  
要為此路由使用的閘道 IP 位址。

OCF\_RESKEY\_source=來源 IP 位址  
要為此路由設定的來源 IP 位址。

# ocf:rsyncd (7)

ocf:rsyncd — 與 OCF 資源代辦相容的 rsync 精靈程序檔。

## 概要

```
[OCF_RESKEY_binpath=string] [OCF_RESKEY_confdir=string]
[OCF_RESKEY_bwlimit=string] rsyncd [start | stop | monitor | validate-all | meta-
data]
```

## 描述

此程序檔會管理 rsync 精靈

## 支援的參數

OCF\_RESKEY\_binpath=rsync 二進位的完整路徑  
rsync 二進位路徑。例如「/usr/bin/rsync」

OCF\_RESKEY\_confdir=包含完整路徑的組態檔案名稱  
包含完整路徑的 rsync 精靈組態檔案名稱。例如「/etc/rsyncd.conf」

OCF\_RESKEY\_bwlimit=限制 I/O 頻寬 (KB/s)  
此選項可讓您指定最大傳輸速率 (KB/s)。對大型檔案 (多兆位元組及以上)  
使用 rsync 時，此選項最有效。由於 rsync 傳輸的本質是傳送資料區塊，因  
此如果 rsync 判斷傳輸速率過快，將會在傳送下一個資料區塊之前等待一段  
時間。如此，平均傳輸速率就會與指定限制相同。值為零表示不指定限制。

# ocf:SAPDatabase (7)

ocf:SAPDatabase — SAP 資料庫資源代辦

## 概要

```
OCF_RESKEY_SID=string OCF_RESKEY_DIR_EXECUTABLE=string
OCF_RESKEY_DBTYPE=string OCF_RESKEY_NETSERVICENAME=string
OCF_RESKEY_DBJ2EE_ONLY=boolean OCF_RESKEY_JAVA_HOME=string
OCF_RESKEY_STRICT_MONITORING=boolean
OCF_RESKEY_AUTOMATIC_RECOVER=boolean
OCF_RESKEY_DIR_BOOTSTRAP=string OCF_RESKEY_DIR_SECSTORE=string
OCF_RESKEY_DB_JARS=string OCF_RESKEY_PRE_START_USEREXIT=string
OCF_RESKEY_POST_START_USEREXIT=string
OCF_RESKEY_PRE_STOP_USEREXIT=string
OCF_RESKEY_POST_STOP_USEREXIT=string SAPDatabase [start | stop | status
| monitor | validate-all | meta-data | methods]
```

## 描述

SAP 資料庫的資源程序檔。它會將所有類型的 SAP 資料庫都做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_SID=SAP 系統 ID  
唯一的 SAP 系統識別碼。例如 P01

OCF\_RESKEY\_DIR\_EXECUTABLE=sapstartsrv 與 sapcontrol 的路徑  
儲存 sapstartsrv 與 sapcontrol 的完全合法路徑。

OCF\_RESKEY\_DBTYPE=資料庫廠商  
您使用的資料庫廠商的名稱。設定為：ORA、DB6 或 ADA

- OCF\_RESKEY\_NETSERVICENAME=監聽程式名稱  
Oracle TNS 監聽程式名稱。
- OCF\_RESKEY\_DBJ2EE\_ONLY=僅安裝 JAVA 堆疊  
若您的 SAP 資料庫中未安裝 ABAP 堆疊，請將此參數設定為 TRUE。
- OCF\_RESKEY\_JAVA\_HOME=Java SDK 的路徑  
僅當 DBJ2EE\_ONLY 參數設定為 true 時才需要設定此參數。輸入 SAP WebAS Java 使用的 Java SDK 的路徑。
- OCF\_RESKEY\_STRICT\_MONITORING=啟用應用程式層級監控  
此參數可控制資源代辦監控資料庫的方式。若設定為 true，則會使用 SAP 工具來測試與資料庫的連接。請勿將其與 Oracle 搭配使用，因為如果歸檔程式出現問題，將會導致非預期的容錯移轉。
- OCF\_RESKEY\_AUTOMATIC\_RECOVER=啟用或停用自動啟動復原  
SAPDatabase 資源代辦會嘗試一次自動復原一個失敗的啟動嘗試。可透過執行 RDBMS 的強制中止和/或復原指令來完成該操作。
- OCF\_RESKEY\_DIR\_BOOTSTRAP=j2ee 開機目錄的路徑  
J2EE 例項開機目錄的完全合法路徑。例如 /usr/sap/P01/J00/j2ee/cluster/bootstrap
- OCF\_RESKEY\_DIR\_SECSTORE=j2ee 安全儲存目錄的路徑  
J2EE 安全儲存目錄的完全合法路徑。例如  
/usr/sap/P01/SYS/global/security/lib/tools
- OCF\_RESKEY\_DB\_JARS=jdbc 驅動程式的檔案名稱  
用於資料庫連線測試之 jdbc 驅動程式的完全合法檔名。系統將會從 Java 引擎 6.40 與 7.00 中的 bootstrap.properties 檔案自動讀取此參數。對於 Java 引擎 7.10，此為強制參數。
- OCF\_RESKEY\_PRE\_START\_USEREXIT=啟動前程序檔的路徑  
儲存應在此資源啟動前執行之程序檔或程式的完全合法路徑。
- OCF\_RESKEY\_POST\_START\_USEREXIT=啟動後程序檔的路徑  
儲存應在此資源啟動後執行之程序檔或程式的完全合法路徑。
- OCF\_RESKEY\_PRE\_STOP\_USEREXIT=停止前程序檔的路徑  
儲存應在此資源停止前執行之程序檔或程式的完全合法路徑。

OCF\_RESKEY\_POST\_STOP\_USEREXIT=停止後程序檔的路徑  
儲存應在此資源停止後執行之程序檔或程式的完全合法路徑。

# ocf:SAPInstance (7)

ocf:SAPInstance — SAP 例項資源代辦

## 概要

```
OCF_RESKEY_InstanceName=string OCF_RESKEY_DIR_EXECUTABLE=string  
OCF_RESKEY_DIR_PROFILE=string OCF_RESKEY_START_PROFILE=string  
OCF_RESKEY_START_WAITTIME=string  
OCF_RESKEY_AUTOMATIC_RECOVER=boolean  
OCF_RESKEY_PRE_START_USEREXIT=string  
OCF_RESKEY_POST_START_USEREXIT=string  
OCF_RESKEY_PRE_STOP_USEREXIT=string  
OCF_RESKEY_POST_STOP_USEREXIT=string SAPInstance [start | recover |  
stop | status | monitor | validate-all | meta-data | methods]
```

## 描述

SAP 的資源程序檔。它會將 SAP 例項做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_InstanceName=例項名稱：SID\_INSTANCE\_VIR-HOSTNAME  
完全合法的 SAP 例項名稱。例如 P01\_DVEBMGS00\_sapp01ci

OCF\_RESKEY\_DIR\_EXECUTABLE=sapstartsrv 與 sapcontrol 的路徑  
儲存 sapstartsrv 與 sapcontrol 的完全合法路徑。

OCF\_RESKEY\_DIR\_PROFILE=啟動設定檔的路徑  
儲存 SAP START 設定檔的完全合法路徑。

OCF\_RESKEY\_START\_PROFILE=啟動設定檔名稱  
SAP START 設定檔的名稱。

OCF\_RESKEY\_START\_WAITTIME=在指定時間之後檢查成功的啟動 (不等待 J2EE-Addin)

在經過指定時間 (秒) 之後，資源代辦會執行監控作業。如果監控作業傳回 SUCCESS，則會將啟動視為 SUCCESS。這對於解決與 J2EE-Addin 例項等有關的時間問題很有幫助。

OCF\_RESKEY\_AUTOMATIC\_RECOVER=啟用或停用自動啟動復原

SAPInstance 資源代辦會嘗試一次自動復原一個失敗的啟動嘗試。透過停止執行中的例項程序並執行 cleanipc 可以完成該操作。

OCF\_RESKEY\_PRE\_START\_USEREXIT=啟動前程序檔的路徑

儲存應在此資源啟動前執行之程序檔或程式的完全合法路徑。

OCF\_RESKEY\_POST\_START\_USEREXIT=啟動後程序檔的路徑

儲存應在此資源啟動後執行之程序檔或程式的完全合法路徑。

OCF\_RESKEY\_PRE\_STOP\_USEREXIT=停止前程序檔的路徑

儲存應在此資源停止前執行之程序檔或程式的完全合法路徑。

OCF\_RESKEY\_POST\_STOP\_USEREXIT=停止後程序檔的路徑

儲存應在此資源停止後執行之程序檔或程式的完全合法路徑。

# ocf:scsi2reserve (7)

ocf:scsi2reserve — scsi-2 reservation

## 概要

```
[OCF_RESKEY_scsi_reserve=string] [OCF_RESKEY_sharedisk=string]  
[OCF_RESKEY_start_loop=string] scsi2reserve [start | stop | monitor | meta-  
data | validate-all]
```

## 描述

scsi-2-reserve 資源代辦是 SCSI-2 reservation 的預留位置。scsi-2-reserve 資源的正常例項，指出指定 SCSI 設備的擁有情況。此資源代辦依賴於 Linux 特定 scsires 套件中的 scsi\_reserve。

## 支援的參數

OCF\_RESKEY\_scsi\_reserve=scsi\_reserve 指令  
scsi\_reserve 是 scsires 套件中的一個指令。它可協助您在 SCSI 設備上進行 SCSI-2 reservation。

OCF\_RESKEY\_sharedisk=共享磁碟。  
可以保留的共享磁碟。

OCF\_RESKEY\_start\_loop=放棄前的重試次數。  
系統在放棄前將會做數次嘗試。Start\_loop 表示系統將要嘗試的次數。

# ocf:SendArp (7)

ocf:SendArp — SendArp 資源代辦

## 概要

```
[OCF_RESKEY_ip=string] [OCF_RESKEY_nic=string] SendArp [start | stop |  
monitor | meta-data | validate-all]
```

## 描述

此程序檔會為 IP 位址傳送免費 ARP。

## 支援的參數

OCF\_RESKEY\_ip=IP 位址  
要傳送 arp 套件的 IP 位址。

OCF\_RESKEY\_nic=NIC  
要傳送 arp 套件的 nic。

# ocf:ServeRAID (7)

ocf:ServeRAID — ServeRAID 資源代辦

## 概要

[OCF\_RESKEY\_serveraid=*integer*] [OCF\_RESKEY\_mergegroup=*integer*]  
ServeRAID [start | stop | status | monitor | validate-all | meta-data | methods]

## 描述

ServeRAID 的資源程序檔。它會啟用/停用共享的 ServeRAID 合併群組。

## 支援的參數

OCF\_RESKEY\_serveraid=*serveraid*  
ServeRAID 介面卡的介面卡號碼。

OCF\_RESKEY\_mergegroup=*mergegroup*  
考慮中的邏輯磁碟機。

# ocf:sfex (7)

ocf:sfex — SF-EX 資源代辦

## 概要

```
[OCF_RESKEY_device=string] [OCF_RESKEY_index=integer]
[OCF_RESKEY_collision_timeout=integer]
[OCF_RESKEY_monitor_interval=integer]
[OCF_RESKEY_lock_timeout=integer] sfex [start | stop | monitor | meta-data]
```

## 描述

SF-EX 的資源程序檔。它會以獨占模式管理共享儲存媒體。

## 支援的參數

OCF\_RESKEY\_device=區塊設備  
儲存獨占控制資料的區塊設備路徑。

OCF\_RESKEY\_index=索引  
區塊設備中儲存獨占控制資料的位置。可以指定 1 或大於 1 的值。  
預設值為 1。

OCF\_RESKEY\_collision\_timeout=鎖定擷取的等待時間  
偵測到鎖定擷取衝突時等待的時間。預設值為 1 秒。

OCF\_RESKEY\_monitor\_interval=監控間隔  
監控間隔 (秒)。預設值為 10 秒

OCF\_RESKEY\_lock\_timeout=有效鎖定期間  
有效鎖定期間 (秒)。預設值為 20 秒。

# ocf:SphinxSearchDaemon (7)

ocf:SphinxSearchDaemon — searchd 資源代辦

## 概要

```
OCF_RESKEY_config=string [OCF_RESKEY_searchd=string]
[OCF_RESKEY_search=string] [OCF_RESKEY_testQuery=string]
SphinxSearchDaemon [start | stop | monitor | meta-data | validate-all]
```

## 描述

此為 searchd 資源代辦。它會管理 Sphinx 搜尋精靈。

## 支援的參數

OCF\_RESKEY\_config=組態檔案  
searchd 組態檔案

OCF\_RESKEY\_searchd=searchd 二進位  
searchd 二進位

OCF\_RESKEY\_search=搜尋二進位  
用於監控動作中功能測試的搜尋二進位。

OCF\_RESKEY\_testQuery=測試查詢  
用於監控動作中功能測試的測試查詢。查詢不需要與索引中的任何文件比對。其目的僅僅是為了測試搜尋精靈是否能夠查詢其索引並做出適當回應。

# ocf:Squid (7)

ocf:Squid — Squid 的資源代辦

## 概要

```
[OCF_RESKEY_squid_exe=string] OCF_RESKEY_squid_conf=string  
OCF_RESKEY_squid_pidfile=string OCF_RESKEY_squid_port=integer  
[OCF_RESKEY_squid_stop_timeout=integer]  
[OCF_RESKEY_debug_mode=string] [OCF_RESKEY_debug_log=string] Squid  
[start | stop | status | monitor | meta-data | validate-all]
```

## 描述

Squid 的資源代辦。它會將 Squid 例項做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_squid\_exe=可執行檔

此為必要參數。此參數指定 squid 的可執行檔。

OCF\_RESKEY\_squid\_conf=組態檔案

此為必要參數。此參數為此資源代辦所管理的 squid 例項指定組態檔案。

OCF\_RESKEY\_squid\_pidfile=Pid 檔案

此為必要參數。此參數為此資源代辦所管理的 squid 例項指定程序 id 檔案。

OCF\_RESKEY\_squid\_port=連接埠號碼

此為必要參數。此參數為此資源代辦所管理的 squid 例項指定連接埠號碼。  
如果使用多個連接埠，則只能指定其中的一個。

OCF\_RESKEY\_squid\_stop\_timeout=確認一般停止方法要等待的秒數

此為可省略的參數。發生停止動作時，系統會先使用一般停止方法，然後等待動作的完成確認，持續時間為此參數所指定的秒數。預設值為 10。

OCF\_RESKEY\_debug\_mode=除錯模式

這是一個選擇性的參數。若此參數包含「x」或「v」，則此資源代辦會在除錯模式下執行。若包含「x」，STDOUT 與 STDERR 均會重新導向至「debug\_log」指定的記錄檔案，然後會開啟內建外圍程序選項「x」。「v」選項與此類似。

OCF\_RESKEY\_debug\_log=除錯記錄的目的地

此為可以省略的選擇性參數。此參數指定除錯記錄的目的檔案，並且僅當此資源代辦在除錯模式下執行時才起作用。有關除錯模式的資訊，請參閱「debug\_mode」。如果需要指定一個值卻未指定，則系統會遵循以下規則：將「/var/log/」做為目錄部分，將「syslog\_ng\_conf」所指定的組態檔案的主檔名做為主檔名部分，將「.log」做為字尾。

# ocf:Stateful (7)

ocf:Stateful — 狀態資源代辦範例

## 概要

```
OCF_RESKEY_state=string Stateful [start | stop | monitor | meta-data | validate-all]
```

## 描述

此為可實作兩種狀態的資源代辦範例

## 支援的參數

OCF\_RESKEY\_state=狀態檔案  
儲存資源狀態的位置。

# ocf:SysInfo (7)

ocf:SysInfo — SysInfo 資源代辦

## 概要

```
[OCF_RESKEY_pidfile=string] [OCF_RESKEY_delay=string] SysInfo [start  
| stop | monitor | meta-data | validate-all]
```

## 描述

此為 SysInfo 資源代辦。它會在 CIB 中記錄節點範例 Linux 輸出的各種屬性：  
arch: i686 os: Linux-2.4.26-gentoo-r14 free\_swap: 1999 cpu\_info: Intel(R) Celeron(R)  
CPU 2.40GHz cpu\_speed: 4771.02 cpu\_cores: 1 cpu\_load: 0.00 ram\_total: 513 ram\_free:  
117 root\_free: 2.4 Sample Darwin output: arch: i386 os: Darwin-8.6.2 cpu\_info: Intel  
Core Duo cpu\_speed: 2.16 cpu\_cores: 2 cpu\_load: 0.18 ram\_total: 2016 ram\_free: 787  
root\_free: 13 Units: free\_swap: Mb ram\_\*: Mb root\_free: Gb cpu\_speed (Linux):  
bogomips cpu\_speed (Darwin): Ghz

## 支援的參數

OCF\_RESKEY\_pidfile=PID 檔案  
PID 檔案。

OCF\_RESKEY\_delay=抑制延遲  
讓值保持穩定的間隔。

# ocf:tomcat (7)

ocf:tomcat — tomcat 資源代辦

## 概要

```
OCF_RESKEY_tomcat_name=string OCF_RESKEY_script_log=string
[OCF_RESKEY_tomcat_stop_timeout=integer]
[OCF_RESKEY_tomcat_suspend_trialcount=integer]
[OCF_RESKEY_tomcat_user=string] [OCF_RESKEY_statusurl=string]
[OCF_RESKEY_java_home=string] OCF_RESKEY_catalina_home=string
OCF_RESKEY_catalina_pid=string
[OCF_RESKEY_tomcat_start_opts=string]
[OCF_RESKEY_catalina_opts=string]
[OCF_RESKEY_catalina_rotate_log=string]
[OCF_RESKEY_catalina_rotatetime=integer] tomcat [start | stop | status |
monitor | meta-data | validate-all]
```

## 描述

Tomcat 的資源程序檔。它會將 Tomcat 例項做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_tomcat\_name=資源名稱  
資源名稱。

OCF\_RESKEY\_script\_log=此程序檔記錄的目的地  
此程序檔記錄的目的地。

OCF\_RESKEY\_tomcat\_stop\_timeout=停止時的逾時  
停止時的逾時。

OCF\_RESKEY\_tomcat\_suspend\_trialcount=等待停止時的重試次數  
等待停止時的重試次數。

OCF\_RESKEY\_tomcat\_user=啟動資源的使用者名稱  
啟動資源的使用者名稱。

OCF\_RESKEY\_statusurl=狀態確認的 URL  
狀態確認的 URL。

OCF\_RESKEY\_java\_home=Java 的主目錄  
Java 的主目錄。

OCF\_RESKEY\_catalina\_home=Tomcat 的主目錄  
Tomcat 的主目錄。

OCF\_RESKEY\_catalina\_pid=Tomcat 的 PID 檔案名稱  
Tomcat 的 PID 檔案名稱。

OCF\_RESKEY\_tomcat\_start\_opts=Tomcat 啟動選項  
Tomcat 啟動選項。

OCF\_RESKEY\_catalina\_opts=Catalina 選項  
Catalina 選項。

OCF\_RESKEY\_catalina\_rotate\_log=旋轉 catalina.out 旗標  
旋轉 catalina.out 旗標。

OCF\_RESKEY\_catalina\_rotatetime=旋轉 catalina.out 的時間範圍  
旋轉 catalina.out 的時間範圍。

# ocf:VIPArip (7)

ocf:VIPArip — RIP2 通訊協定支援的虛擬 IP 位址

## 概要

```
OCF_RESKEY_ip=string [OCF_RESKEY_nic=string] VIPArip [start | stop | monitor  
| validate-all | meta-data]
```

## 描述

RIP2 通訊協定支援的虛擬 IP 位址。此程序檔會透過 `quagga/ripd` 管理其他子網路中的 IP 別名。它可新增或移除 IP 別名。

## 支援的參數

`OCF_RESKEY_ip`=其他子網路中的 IP 位址  
其他子網路中的 IPv4 位址，例如「192.168.1.1」。

`OCF_RESKEY_nic`=用於廣播路由資訊的 `nic`  
用於廣播路由資訊的 `nic`。`ripd` 會使用此 `nic` 將路由資訊廣播至其他節點。

# ocf:VirtualDomain (7)

ocf:VirtualDomain — 管理虛擬網域

## 概要

```
OCF_RESKEY_config=string [OCF_RESKEY_hypervisor=string]
[OCF_RESKEY_force_stop=boolean]
[OCF_RESKEY_migration_transport=string]
[OCF_RESKEY_monitor_scripts=string] VirtualDomain [start | stop | status
| monitor | migrate_from | migrate_to | meta-data | validate-all]
```

## 描述

libvirtd 所管理之虛擬網域 (也稱為 domU、虛擬機器、虛擬環境等，依具體情況而定) 的資源代辦。

## 支援的參數

OCF\_RESKEY\_config=虛擬網域組態檔案  
此虛擬網域之 libvirt 組態檔案的絕對路徑。

OCF\_RESKEY\_hypervisor=監管程式 URI  
要連接的監管程式 URI。如需支援的 URI 格式的詳細資料，請參閱 libvirt 文件。預設格式依系統而定。

OCF\_RESKEY\_force\_stop=停止時強制關閉  
停止時強制關閉 (「destroy」) 網域。僅當虛擬網域 (或虛擬化後端) 不支援正常關閉時才啟用此選項。

OCF\_RESKEY\_migration\_transport=遠端監管程式傳輸  
在移轉時用於連接遠端監管程式的傳輸。如需可用傳輸的詳細資料，請參閱 libvirt 文件。若省略此參數，資源將使用 libvirt 的預設傳輸來連接遠端監管程式。

OCF\_RESKEY\_monitor\_scripts=監控程序檔的空格分隔清單

若要另外監控虛擬網域中的服務，請新增此參數以及要監控的程序檔清單。  
注意：使用監控程序檔時，僅當所有監控程序檔成功完成後，啟動與移轉自  
作業才會完成。請務必將這些作業的逾時設定為適合此延遲的值。

# ocf:WAS6 (7)

ocf:WAS6 — WAS6 資源代辦

## 概要

```
[OCF_RESKEY_profile=string] WAS6 [start | stop | status | monitor | validate-all |  
meta-data | methods]
```

## 描述

WAS6 的資源程序檔。它會將 WebSphere Application Server (WAS6) 做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_profile=設定檔名稱  
WAS 設定檔名稱。

# ocf:WAS (7)

ocf:WAS — WAS 資源代辦

## 概要

```
[OCF_RESKEY_config=string] [OCF_RESKEY_port=integer] WAS [start | stop |  
status | monitor | validate-all | meta-data | methods]
```

## 描述

WAS 的資源程序檔。它會將 Websphere Application Server (WAS) 做為 HA 資源進行管理。

## 支援的參數

OCF\_RESKEY\_config=組態檔案  
WAS 組態檔案。

OCF\_RESKEY\_port=連接埠  
WAS (snoop) 連接埠號碼。

# ocf:WinPopup (7)

ocf:WinPopup — WinPopup 資源代辦

## 概要

[OCF\_RESKEY\_hostfile=string] WinPopup [start | stop | status | monitor | validate-all | meta-data]

## 描述

WinPopup 的資源程序檔。每當有接管動作發生時，它都會向系統管理員的工作站傳送 WinPopup 訊息。

## 支援的參數

OCF\_RESKEY\_hostfile=主機檔案  
包含要將 WinPopup 訊息要傳送至的目標主機的檔案。

# ocf:Xen (7)

ocf:Xen — 管理 Xen DomU

## 概要

```
[OCF_RESKEY_xmfile=string] [OCF_RESKEY_name=string]  
[OCF_RESKEY_allow_migrate=boolean]  
[OCF_RESKEY_shutdown_timeout=boolean]  
[OCF_RESKEY_allow_mem_management=boolean]  
[OCF_RESKEY_reserved_Dom0_memory=string]  
[OCF_RESKEY_monitor_scripts=string] Xen [start | stop | migrate_from |  
migrate_to | monitor | meta-data | validate-all]
```

## 描述

Xen 監管程式的資源代辦。它會透過將叢集資源的啟動與停止分別映射至 Xen 的建立與關閉來管理 Xen 虛擬機器例項。請注意：對於名稱，系統將嘗試從組態檔案擷取名稱 (xmfile 屬性)。如果使用簡單的指定陳述式，則情況應該較好。否則，如果涉及某些特殊情況，例如依其他變數而定的動態指定名稱，則系統將嘗試偵測該情況，然後要求您設定名稱屬性。只要有可能會發生不合理的情況，您就應該設定名稱屬性。例如，某個組態檔案位於共享儲存上，因此有可能會遺失的情況下，您就應該如此。如果所有措施均失敗，系統最終會後退到例項 id，以保留反向相容性。您也可透過啟用 meta\_attribute allow\_migrate 來移轉準虛擬化訪客。

## 支援的參數

OCF\_RESKEY\_xmfile=Xen 控制檔案  
此虛擬機器 Xen 控制檔案的絕對路徑。

OCF\_RESKEY\_name=Xen DomU 名稱  
虛擬機器的名稱。

OCF\_RESKEY\_allow\_migrate=使用即時移轉  
此布林參數可讓您對準虛擬化機器使用即時移轉。

OCF\_RESKEY\_shutdown\_timeout=關閉升級逾時  
Xen 代辦會先嘗試使用 `xm shutdown` 來進行順序關閉。如果在逾時之前尚未成功完成該操作，代辦會升級至 `xm destroy`，強制停止節點。若未設定此參數，會將其預設為停止動作逾時值的 2/3。將此值設定為 0 會強制立即執行 `destroy`。

OCF\_RESKEY\_allow\_mem\_management=使用動態記憶體管理  
此參數可啟用對 Dom0 和 DomU 啟動和停止動作所需使用的記憶體進行動態調整的功能。預設為不動態調整記憶體。

OCF\_RESKEY\_reserved\_Dom0\_memory=最小 Dom0 記憶體  
若使用記憶體管理，此參數可定義要為 dom0 保留的最少記憶體容量。預設的最小記憶體容量為 512 MB。

OCF\_RESKEY\_monitor\_scripts=監控程序檔的空格分隔清單  
若要另外監控非特權網域中的服務，請新增此參數以及要監控的程序檔清單。注意：在這種情況下，請確認將監控作業的啟動延遲時間至少設定為 DomU 啟動所有服務所需的時間。

# ocf:Xinetd (7)

ocf:Xinetd — Xinetd 資源代辦

## 概要

```
[OCF_RESKEY_service=string] Xinetd [start | stop | restart | status | monitor |  
validate-all | meta-data]
```

## 描述

Xinetd 的資源程序檔。它會啟動/停止 xinetd 管理的服務。請注意，xinetd 精靈自身必須處於執行狀態：系統不會自行啟動或停止該精靈。重要事項：如果叢集所管理的服務是僅有的已啟用的服務，您應該為 xinetd 指定 `-stayalive` 選項，否則該指令將會在 Heartbeat 停止時結束。此為，您還可以啟用某些內部服務，例如 echo。

## 支援的參數

OCF\_RESKEY\_service=服務名稱  
xinetd 管理的服務名稱。

## **v. 附錄**



# GNU 授權



本附錄包含 GNU 通用公共授權 (General Public License) 和 GNU 自由文件授權 (Free Documentation License)。

## GNU General Public License

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

### Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

## GNU GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

**0.** This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The [ Program ], below, refers to any such program or work, and a [ work based on the Program ] means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term [ modification ].) Each licensee is addressed as [ you ].

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

**1.** You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

**2.** You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a)** You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b)** You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c)** If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

**3.** You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a)** Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b)** Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c)** Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

**4.** You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and [any later version], you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

## *NO WARRANTY*

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

## *END OF TERMS AND CONDITIONS*

### *How to Apply These Terms to Your New Programs*

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the [copyright] line and a pointer to where the full notice is found.

*one line to give the program's name and an idea of what it does. Copyright (C) yyyy name of author*

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details
type `show w`. This is free software, and you are welcome
to redistribute it under certain conditions; type `show c'
for details.
```

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a [ copyright disclaimer ] for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright
interest in the program `Gnomovision'
(which makes passes at compilers) written
by James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License [<http://www.fsf.org/licenses/lgpl.html>] instead of this License.

## GNU Free Documentation License

Version 1.2, November 2002

Copyright (C) 2000,2001,2002 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

### PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of [ copyleft ], which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

## APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The [ Document ], below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as [ you ]. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A [ Modified Version ] of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A [ Secondary Section ] is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The [ Invariant Sections ] are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The [ Cover Texts ] are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A [ Transparent ] copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not [ Transparent ] is called [ Opaque ].

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The [ Title Page ] means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, [ Title Page ] means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section [ Entitled XYZ ] means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as [ Acknowledgements ], [ Dedications ], [ Endorsements ], or [ History ].) To [ Preserve the Title ] of such a section when you modify the Document means that it remains a section [ Entitled XYZ ] according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

## VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

## COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

## MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled 「 History 」, Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled 「 History 」 in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the 「 History 」 section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled 「 Acknowledgements 」 or 「 Dedications 」, Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled 「 Endorsements 」. Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled 「 Endorsements 」 or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled 「 Endorsements 」, provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

## COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled 「 History 」 in the various original documents, forming one section Entitled 「 History 」 ; likewise combine any sections Entitled 「 Acknowledgements 」 , and any sections Entitled 「 Dedications 」 . You must delete all sections Entitled 「 Endorsements 」 .

## COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

## AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an “aggregate” if the copyright resulting from the compilation is not used to limit the legal rights of the compilation’s users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document’s Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

## TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled 「 Acknowledgements 」 , 「 Dedications 」 , or 「 History 」 , the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

## TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

## FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License 「 or any later version 」 applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

## ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME.  
Permission is granted to copy, distribute and/or modify this document  
under the terms of the GNU Free Documentation License, Version 1.2  
only as published by the Free Software Foundation;  
with the Invariant Section being this copyright notice and license.  
A copy of the license is included in the section entitled "GNU  
Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the  
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

# 術語

## 主動/主動、主動/被動

關於服務如何在節點上執行的概念。主動-被動情境是指有一項或多項服務正在主動節點上執行，而被動節點則等待主動節點失敗。另外，主動-主動則表示各節點同時處於主動與被動狀態。

## 叢集(與硬碟，磁片有關時)

高效能叢集是指共同承擔應用程式負載以快速完成操作的一個電腦群組 (真實或虛擬)。High Availability 叢集主要用於最大可能地確保服務的可用性。

## 叢集分割區

當一個或多個節點與叢集其餘節點間的通訊失敗時，就會出現叢集分割區。叢集分割區的節點仍為使用中狀態且能與彼此通訊，但它們無法察覺不能與之通訊的節點。由於無法確認其他分割區的遺失，將出現電腦分裂情境 (另請參閱電腦分裂 [275頁])。

## 共識叢集成員 (CCM)

CCM 決定由哪些節點組成叢集並在叢集中共享此資訊。任何新的新增項及任何節點或法定數目的遺失均透過 CCM 傳送。CCM 模組將在叢集的每個節點上執行。

## 叢集資訊庫 (CIB)

表示整個叢集組態與狀態 (節點成員、資源、限制等)。以 XML 編寫並存放在記憶體中。主要 CIB 將在 DC 上保留並維護，並將複製到其他節點。

## 叢集資源管理員 (CRM)

負責協調所有非本地互動的主要管理實體。叢集的各節點均有其各自的 CRM，但 DC 上所執行的一個節點將被選出負責將決策傳送到其他非本地 CRM 並處理他們的輸入。一個 CRM 可與許多元件進行互動：自己節點及其他節點上的本地資源管理員、非本地 CRM、管理指令、圍籬區隔功能、及成員層。

## 指定協調者 (DC)

「主要」節點。此節點為保留 CIB 主要副本的所在。所有其他節點均從目前 DC 取得其組態和資源配置資訊。成員發生變更後將從叢集的所有節點中選出 DC。

### 分散式複製區塊設備 (drbd)

DRBD 是專用於建立 High Availability 叢集的區塊設備。整個區塊設備透過專屬網路鏡像，並被視做網路 RAID-1。

### 容錯移轉

當一台機器上的資源或節點發生故障且受影響的資源將在其他節點上啟動時，會發生此情況。

### 圍籬區隔

描述阻止非叢集成員存取共享資源的概念。該功能可透過停止(關閉)「行為錯誤的」節點以防止它導致問題，從節點鎖定狀態未定的資源，或多種其他方式實現。此外，圍籬區隔在節點與資源圍籬區隔間是可辨識的。

### Heartbeat 資源代辦

Heartbeat 資源代辦曾在 Heartbeat 版本 1 中廣泛使用，目前已廢棄，但在版本 2 中仍受支援。Heartbeat 資源代辦可執行 start、stop 與 status 作業，其位於 /etc/ha.d/resource.d 或 /etc/init.d。如需 Heartbeat 資源代辦的詳細資訊，請參見 <http://www.linux-ha.org/HeartbeatResourceAgent>。

### 本地資源管理員 (LRM)

本地資源管理員 (LRM) 負責在資源上執行作業。它使用資源代辦程序檔來執行工作。LRM 是「無用的」，自己並不知道任何規則，需要 DC 告訴它該做什麼。

### LSB 資源代辦

LSB 資源代辦是標準 LSB init 程序檔。LSB init 程序檔不限用於高可用性網路位置。任何 LSB 相容的 Linux 系統均可使用 LSB init 程序檔來控制服務。任何 LSB 資源代辦均支援 start、stop、restart、status 與 force-reload 選項，並可選擇性地提供 try-restart 與 reload。LSB 資源代辦位於 /etc/init.d。如需更多有關 LSB 資源代辦及實際規格的詳細資訊，請參閱 <http://www.linux-ha.org/LSBResourceAgent> 與 [http://www.linux-foundation.org/spec/refspecs/LSB\\_3.0.0/LSB-Core-generic/LSB-Core-generic/iniscriptact.html](http://www.linux-foundation.org/spec/refspecs/LSB_3.0.0/LSB-Core-generic/LSB-Core-generic/iniscriptact.html)

### 節點

任何是叢集成員且對使用者不可見的電腦(真實或虛擬)。

### pingd

ping 精靈。它會使用 ICMP ping 持續聯絡叢集之外的一或多台伺服器。

## 規則引擎 (PE)

規則引擎會計算出在 CIB 中實作規則變更所需採取的動作。此資訊隨後將被傳遞到異動引擎，然後便會在叢集設定中實作規則變更。PE 始終在 DC 上執行。

## OCF 資源代辦

OCF 資源代辦與 LSB 資源代辦 (init 程序檔) 類似。任何 OCF 資源代辦均必須支援 start、stop 與 status (有時稱為 monitor) 選項。此外，他們還支援 metadata 選項，該選項會以 XML 返回資源代辦類型的描述。亦可支援其他選項，但並非強制。OCF 資源代辦位於 /usr/lib/ocf/resource.d/ 提供者。如需有關 OCF 資源代辦及規格草案的詳細資訊，請參見 <http://www.linux-ha.org/OCFResourceAgent> 和 <http://www.opencf.org/cgi-bin/viewcvs.cgi/specs/ra/resource-agent-api.txt?rev=HEAD>。

## 法定數目

在叢集中，如果叢集分割區擁有絕大多數節點(或投票)，則其會被定義為具有法定數目(即「到達法定數目」)。法定數目可準確辨識一個分割區。此部分演算法可阻止多個斷線分割區或節點繼續和導致資料與服務毀損(電腦分裂)。法定數目是圍籬區隔的先決條件，因此確保了法定數目是唯一的。

## 資源

Heartbeat 已知的任何類型的服務或應用程式。範例包括 IP 位址、檔案系統或資料庫。

## 資源代辦 (RA)

資源代辦 (RA) 是一個程序檔，做為代理使用以管理資源。資源代辦有三種類型：OCF (開放叢集架構) 資源代辦、LSB 資源代辦 (標準 LSB init 程序檔) 與 Heartbeat 資源代辦 (Heartbeat v1 資源)。

## 單一故障點 (SPOF)

單一故障點 (SPOF) 是叢集的任一元件，若其發生故障，會觸發整個叢集故障。

## 電腦分裂

在該情境下，叢集節點將被分成兩個或多個彼此不知的群組(透過軟體或硬體故障)。若要阻止電腦分裂狀況對整個叢集產生不良影響，STONITH 必須進行救援。也稱為「分割的叢集」情境。

## STONITH

「Shoot the other node in the head」的縮寫，基本上，它會關閉行為錯誤的節點以防止其在叢集中導致問題。

## 轉換引擎 (TE)

轉換引擎(TE)可從PE獲取規則指令，並加以執行。TE始終在DC上執行。TE從該處指示其他節點上的本地資源管理員要採取的動作。