第 32 章
磁碟陣列

第 32 章 磁碟陣列

磁碟陣列(RAID)

磁碟陣列的基本概念是結合多個小型且便宜的磁碟機成為一個陣列，以達到一個大且昂貴的磁碟機無法做到的效能表現和安全性的目標。 這個磁碟機的陣列將會以一個單一的邏輯儲存單位或磁碟機呈現在電腦中。因此磁碟陣列(Redundant Array Inexpensive Disk)就是使用多顆較便宜的磁碟組成一個容量大，安全性高的整合性磁碟機。

磁碟陣列是用來分散資訊到許多磁碟上的一種方法，使用例如磁碟機平行儲存(disk striping) (RAID Level 0)、磁碟機映射儲存(disk mirroring) (RAID level 1)與具備分佈式同位元檢測資料的磁碟機平行儲存(disk striping with parity)(RAID Level 5) 等技術來達到安全、較短的延遲時間亦或增加讀取或寫入到磁碟的頻寬，並且強化硬碟毀損時的回復能力。

磁碟陣列的概念是資料可以一致地分散到陣列中的每一個磁碟，如要做到這樣，資料必須先劃分為一致大小的區塊。根據所使用的磁碟陣列等級，再寫入每一個區塊到陣列中的硬碟。當資料要被讀取時，過程則相反，這樣將造成多個磁碟機實際上看起來像一個大型磁碟機。

需要保留大量資料在手邊的任何人（如系統管理員）都可藉由使用磁碟陣列而受益。使用磁碟陣列的主要原由包括了速度加快、使用一個單一的虛擬磁碟以增加儲存的容量和減少磁碟發生錯誤時的衝擊。

32-1 硬體磁碟陣列和軟體磁碟陣列

目前有兩種實作磁碟陣列的方法：硬體磁碟陣列與軟體磁碟陣列。

32-1-1 硬體磁碟陣列

硬體為基礎的系統以獨立於主機之外的方式管理磁碟陣列子系統，並且以每一個 RAID 陣列中只有一個單一的磁碟呈現在主機面前。

硬體磁碟陣列裝置的一個例子是連接到一個 SCSI 控制器並且以一個單一的 SCSI 磁碟機代表 RAID 陣列的裝置。 一個外部的磁碟陣列系統移動所有磁碟陣列的處理"能力"到位於外部磁碟子系統的一個控制器，這整個子系統是透過一個一般的 SCSI 控制器來連接到主機，並以一個單一的磁碟呈現給主機。

對作業系統來說，磁碟陣列控制器也是以卡的形式來模擬類似一個 SCSI 控制器，不過它們自己本身處理所有實際的磁碟通訊。我們將一個磁碟插入磁碟陣列 控制器就像是我們對 SCSI 控制器所做的一樣, 不過我們已將它們加入到磁碟陣列控制器的組態設定中，而作業系統則像從未知道發生什麼事一樣。

32-1-2 軟體磁碟陣列

軟體磁碟陣列在核心磁碟（區塊裝置）程式碼上實作這許多種的磁碟陣列等級，它能提供最經濟的解決方案，因為並不需要昂貴的磁碟控制器或熱插拔的主機版。軟體磁碟陣列可以使用在便宜的 IDE 硬碟以及 SCSI 硬碟上，加上今日速度相當快的 CPU，軟體磁碟陣列的效能表現已經超越硬體磁碟陣列了。

Linux 核心中的 MD 驅動程式是一種完全與硬體無關的磁碟陣列解決方案例子。軟體為基礎之陣列的效能表現，是依賴在伺服器的 CPU 效能與負載。

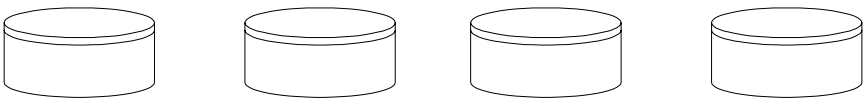
Linux 支援磁碟陣列的選項有執行緒的重建程序、核心為基礎的設定、在 Linux

機器間不需重建的陣列可攜性、使用閒置的系統資源來背景化陣列的重建、支援磁碟的熱插拔和自動偵測 CPU 以使用某些 CPU 的最佳化。

32-2 磁碟陣列等級與線性支援

磁碟陣列支援許多種設定，包括 0, 1,2,3, 4, 5,0+1,1+0 等模式與線性(linear)模式，而經常用到的磁碟陣列有 0,1,4 和 5 這幾個磁碟陣列等級。這些磁碟陣列類型定義如下：

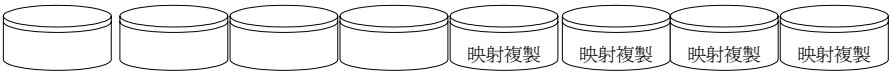
Level 0 — RAID level 0，通常稱為『平行儲存』，它是一種以效能為導向的資料條狀分佈儲存的技術。將要寫入到陣列的資料會先劃分為條狀，再寫入到陣列中的成員磁碟，這個方法以較低的固有開支提供了相當高的 I/O 存取效能，不過並沒有任何的容錯能力(非重複性的儲存)。 level 0 陣列的儲存容量等於硬體磁碟陣列設定中所有成員磁碟的總容量，也等於軟體磁碟陣列設定中成員分割區的總容量。



平行儲存

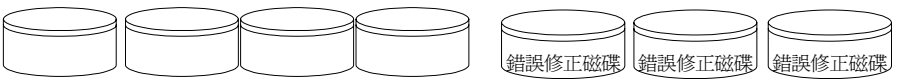
Level 1 — RAID level 1（映射儲存）是所有磁碟陣列模式中使用最長久的一種。Level 1 藉由寫入相同的資料到陣列中的每一個成員磁碟中以提供容錯能力，這種方式將會在每一個磁碟上留下一份映射的複本。由於它的單純與相當高的資料可用性，使得映射儲存一直都很受歡迎。 Level 1 運用兩個以上的磁碟並使用平行式的存取，以提供讀取時相當高的資料傳輸速率，不過通常是獨立式的運作以提供快速的 I/O 處理速率。 Level 1 提供了相當好的資料可靠性，並且改善了執行讀取密集之應用程式的效能，不過相對的所需的成本是相當可觀的。Level 1 陣列的儲存容量等於硬體磁碟陣列設定中映射儲存硬碟中的其中一個磁碟容量，或者是軟體磁碟陣列設定中映射儲存分割區其中一個的容量。 RAID level 1 的成本是很高的，因為我們會寫入相同的資訊到陣列中所有的磁碟上，這將很消耗硬碟

的空間。假如我們已經設定了 RAID level 1，而我們的根目錄分割區 (/) 存放在兩個 60G 的硬碟上，雖然我們總共有 120GB 的空間，不過我們只能存取 120GB 中的 60GB。 另外的 60GB 空間是用來當作第一個 60GB 的映射複製儲存。



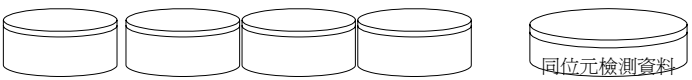
映射儲存

Level 2 — Level 2 是以錯誤修正的記憶型態。這是使用同位元的方式來檢測。錯誤修正系統將修正的記憶系統儲存在兩個或更多的磁碟上。假如有其中之一的磁碟損壞，這殘存的位元組位元會從磁碟被讀取，而且會和這錯誤修正位元一起來重建損壞的資料。Level 2 只需要三個額外的磁碟來安裝錯誤修正系統，而 Level 1 確需要額外的四個磁碟來儲放資料。



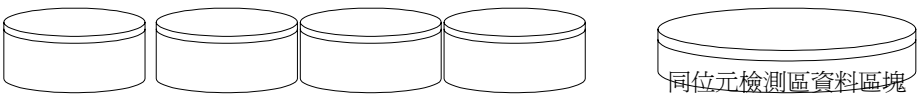
錯誤修正磁碟的記憶系統

Level 3 — Level 3 磁碟陣列三和磁碟陣列二相同，但它只需要一個磁碟來存放同位元檢測資料，因此它的經濟效益比磁碟陣列二還要好。當有個磁區壞掉時，我們磁碟陣列三知道是哪一個磁區毀損，我們可以使用同位元檢測指出它是 0 還是 1。假如這個受檢測的同位元和儲存的同位元資料相同，則這個 bit 為 0(資料沒有毀損)，否則這個 bit 為 1(資料被毀損)。因為 Level 3 的磁碟陣列只用一個磁碟來存放同位元檢測資料，所以它的成本比 Level 2 還便宜。分布式同位元檢測資料是根據陣列中其餘的成員磁碟之內容所計算出來的，如果陣列中的一個磁碟發生錯誤時，這些資訊可以使用來重建資料。 在發生錯誤之磁碟被取出，然後重新輸入資料於其中之前，這些重建的資料可以被使用來滿足對該磁碟 I/O 存取的要求。



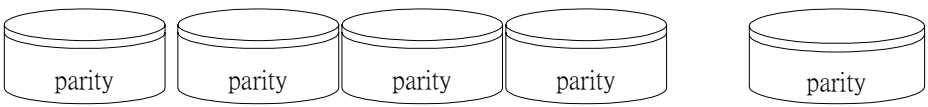
同位元檢測資料

Level 4 — Level 4 使用分布式同位元檢測資料(parity)，並將之存放在單一的磁碟機中以保護資料。Level 4 使用區塊來放置同位元檢測資料，這個方式較適用在交易式的 I/O 存取，而不適用於大型的檔案傳輸。 因為這個既定的 parity 磁碟機代表了一個固有的瓶頸限制，因此在不使用寫回快取 (write-back caching)技術的情況下，很少用到 level 4 模式，一般磁碟機的控制器都有支援快取。雖然 RAID level 4 是某些 RAID 磁碟分割機制的一個選擇，不過在 Red Hat Linux 磁碟陣列 安裝中並沒有如此的選擇。硬體 磁碟陣列 level 4 的儲存容量等於所有成員磁碟機的容量減掉一個成員磁碟機的容量。 軟體磁碟陣列 level 4 的儲存容量等於成員分割區的容量減掉一個分割區的大小（假如它們是同樣大小的話）。



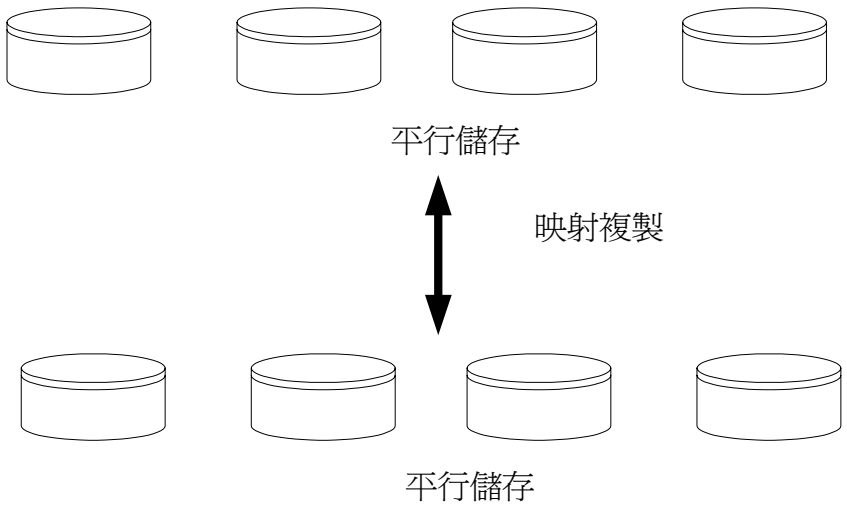
同位元檢測資料區塊

Level 5 — 磁碟陣列五是最普遍被使用的磁碟陣列模式，藉由分散同位元檢測資料區塊 parity 的資訊到陣列中某些或所有的成員磁碟機中，RAID level 5 減少了在 level 4 中存在的寫入瓶頸，在這裏我們使用 parity 來代表同位元檢測資料區塊。 僅有的效能表現瓶頸在於 parity 計算的過程，不過如果使用現今相當快的 CPU 加上軟體 磁碟陣列 設定，這通常不是一個很大的問題。 與 level 4 相同的是，結果會是不對稱的效能表現，也就是讀取的速度比起寫入速度快了很多。 Level 5 通常使用寫回快取來減少這種不對稱性。 硬體 RAID level 5 的儲存容量等於所有成員磁碟機的容量減掉一個成員磁碟機的容量。 軟體 RAID level 5 的儲存容量等於成員分割區的容量減掉一個分割區的大小（假如它們是同樣大小的話）。 RAID level 4 與 RAID level 5 佔有相同的磁碟空間，不過 level 5 擁有較多的優點，因此 level 4 並不被支援。



線性磁碟陣列模式 — 線性磁碟陣列模式是一種簡易地群組磁碟機來建立一個大型的虛擬磁碟。 在線性磁碟陣列模式設定中，空間區塊是從一個成員磁碟依序分配下來，當第一個磁碟完全填滿時，再分配到第二個磁碟，依此類推。 這種群組化並沒有提供任何的效能增益，因為在成員磁碟機之間不可能會有任何分開的 I/O 操作。 線性的磁碟陣列模式也沒有提供重複性，而且說實在的它也降低了可靠性 — 假如任何一個成員磁碟發生錯誤，整個陣列便無法存取使用。 總磁碟容量是所有成員磁碟機的容量。

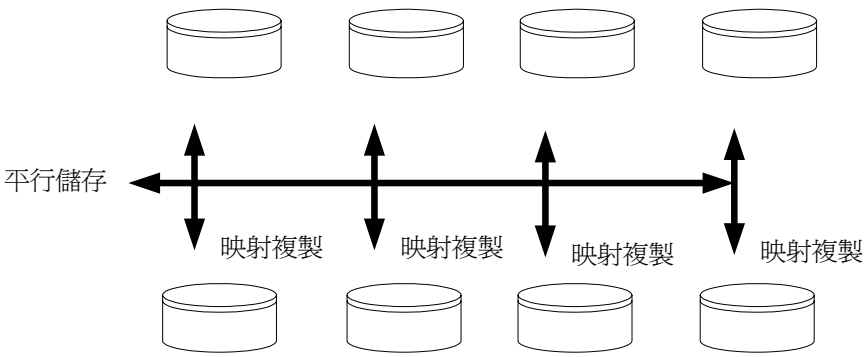
Level 0+1— RAID level 0+1 是 level 0 和 level 1 組成，RAID level 0 提供高的存取效能，而 RAID level 1 提供高的可信賴度。一般而言 Level 0+1 提供比 RAID 5 還要好的效能。所以高存取效能和可信賴度是我們磁碟機很重要的需求。在 RAID level 0+1 中，它會將磁碟切成條狀來平行儲存，再來才是將資料映相複製到其它的磁碟中。



磁碟陣列Level 0+1

Level 1+0— RAID level 1+0 是 level 1 和 level 0 組成。在 Level 0+1 中，是先將磁碟平行儲存，然後才映相複製。在 Level 1+0 中，是先將磁碟映相複製，

然後才平行儲存。RAID Level 1+0 的優點是當有一台磁碟機壞掉時，而它的映射複製磁碟機會取代壞掉磁碟機所存放的資料。



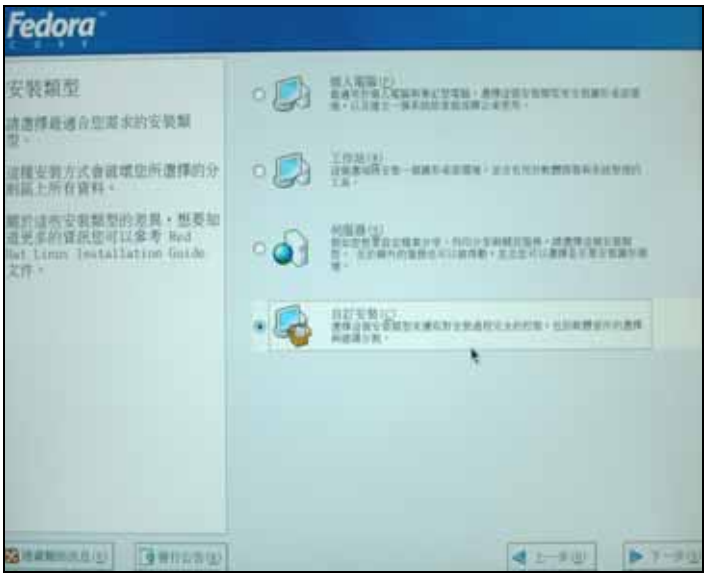
磁碟陣列Level 1+0

32-3 軟體磁碟陣列的設定

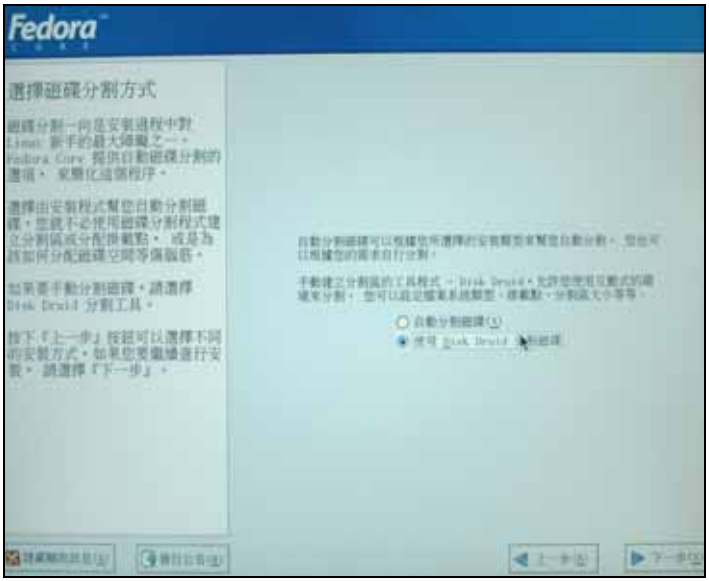
在 Fedora 作業系統的圖形安裝程式或在 kickstart 的安裝階段中，可以配置軟體 磁碟陣列的設定。這節將敘述如何在 Linux 作業系統安裝過程中使用 Disk Druid 介面來配置軟體 磁碟陣列設定。在 Red Hat Linux Fedora 1 中，其設定過程和下列相似。

在我們建立一個磁碟陣列裝置前，我們必須先建立磁碟陣列分割區，請使用下列的指示：

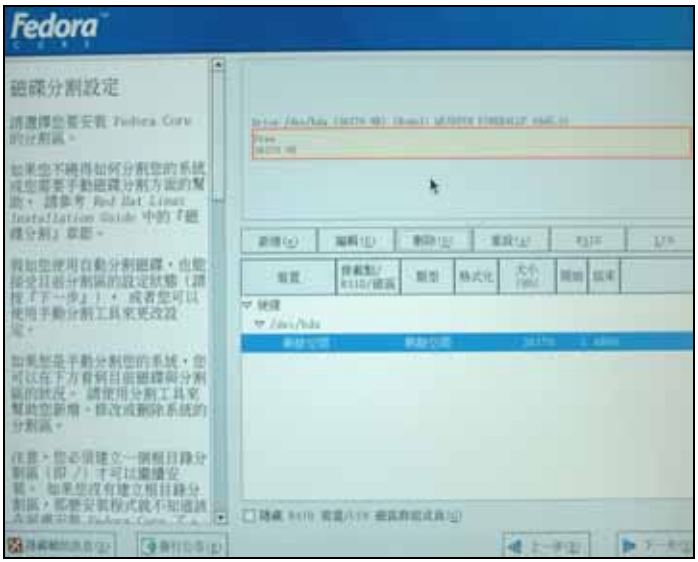
這是選擇要安裝 LINUX 作業系統的型態，我們選取自訂安裝，CUSTOM 來自行選取套件設定。自己動手做可以讓自己更方便且更了解作業系統。



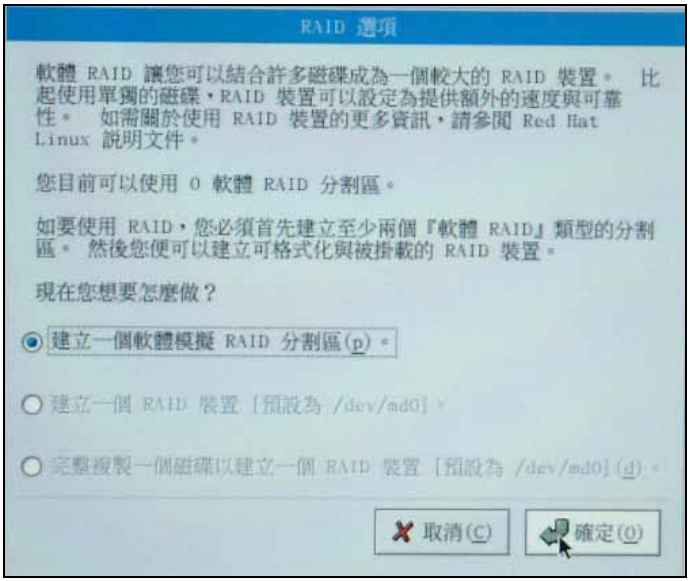
1.我們在『磁碟分割設定』 的畫面，選擇 使用 Disk Druid 分割磁碟。



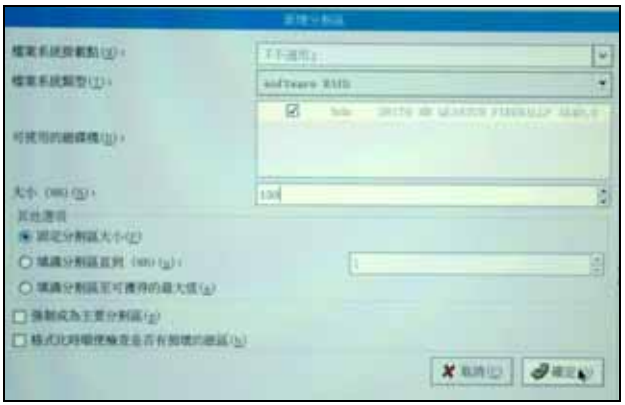
2.在 Disk Druid 中我們可以直接選取 RAID，來建立軟體磁碟陣列。



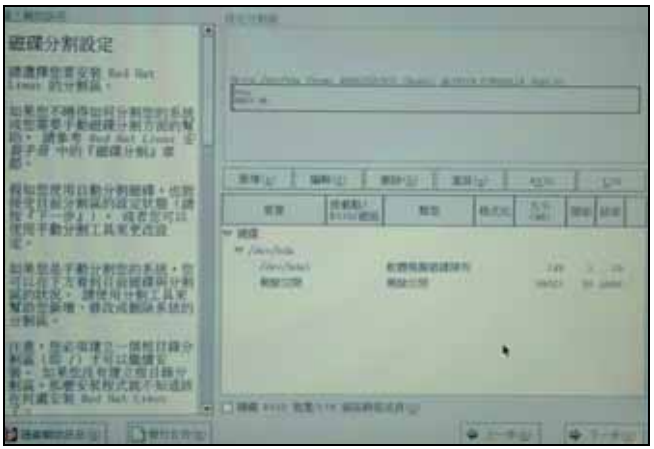
3.我們選擇建立一個軟體模擬 RAID 分割區。



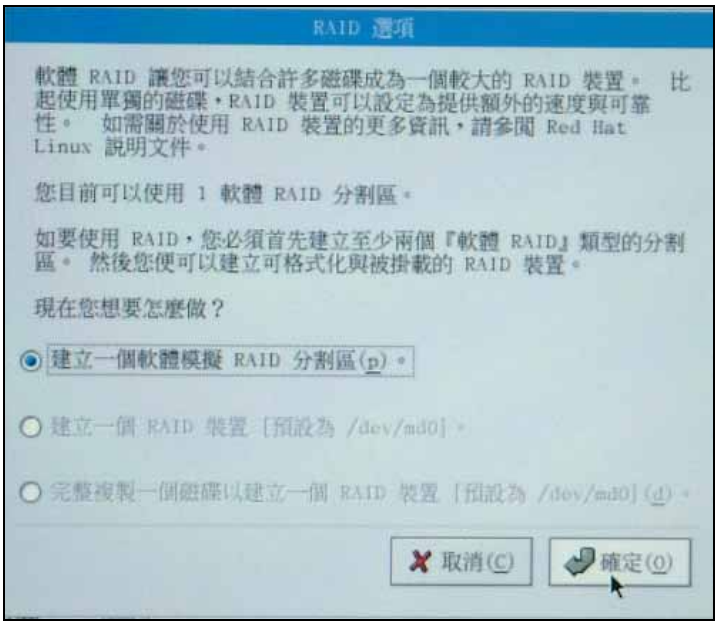
5.在 『可使用的磁碟機』 中選擇要用來建立磁碟陣列的磁碟機。 假如我們有多個磁碟機，在此我們會選取可使用的某一磁碟機，而我們則必須取消選取不放置 RAID 陣列於其上的其它磁碟機。然後輸入該分割區的大小。其它選項中，選擇 『固定大小』 以設定分割區為指定的大小；選擇 『填滿分割區直到 (MB)』，再以 MB 輸入大小以給予分割區大小的一個範圍，或選擇 『填滿分割區至可獲得的最大值』 以使它填滿硬碟上所有可用的空間。 假如我們設定一個以上的分割區為非固定大小的，它們將會分享硬碟上可使用的剩餘空間。假如我們想要該分割區成為一個主分割區，請選擇 『強制成為主要分割區』。假如我們想要安裝程式在格式化前檢查硬碟上的損壞磁區，請選擇 『格式化時順便檢查是否有損壞的磁區』。



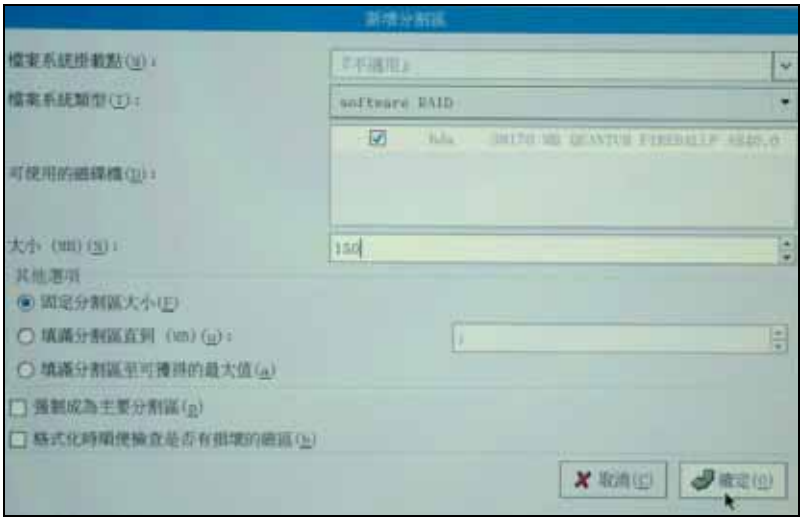
這時我們就建立了軟體模擬磁碟陣列，其裝置為 hda1。我們再選取 RAID，要再建立另外兩個軟體模擬磁碟陣列。



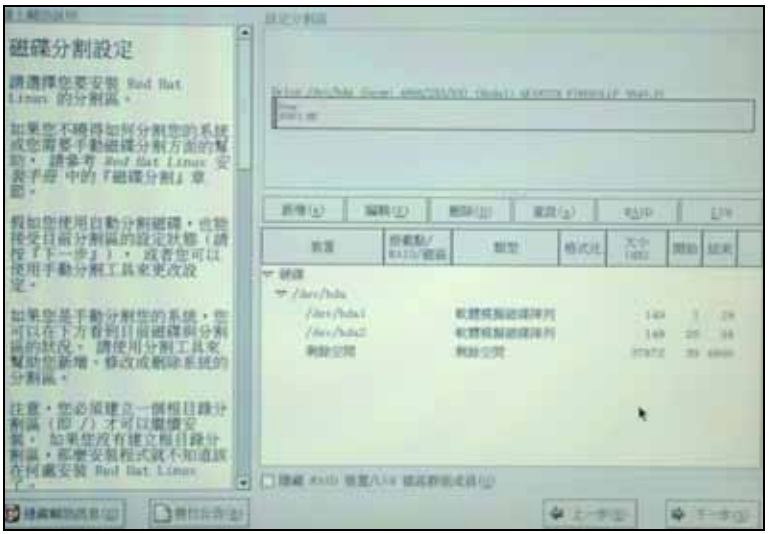
我們選擇建立一個軟體模擬 RAID 分割區。



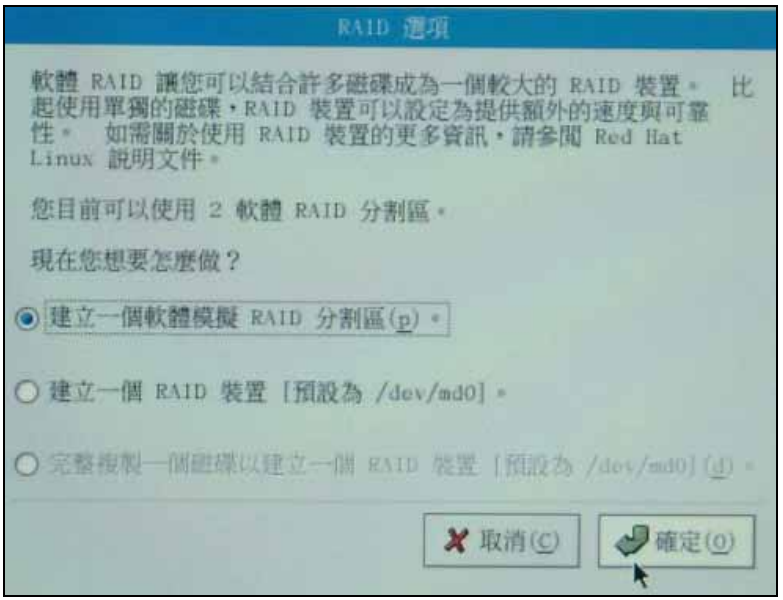
我們選取檔案系統為 software RAID，再設定大小為 150MB。



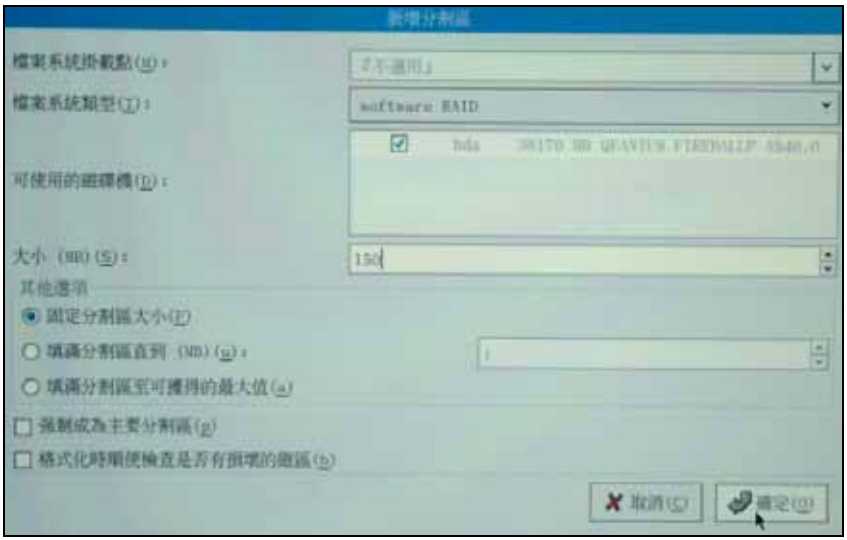
這就是我們新增的 hda2 軟體磁碟陣列。我們現在要建立第三個軟體磁碟陣列。



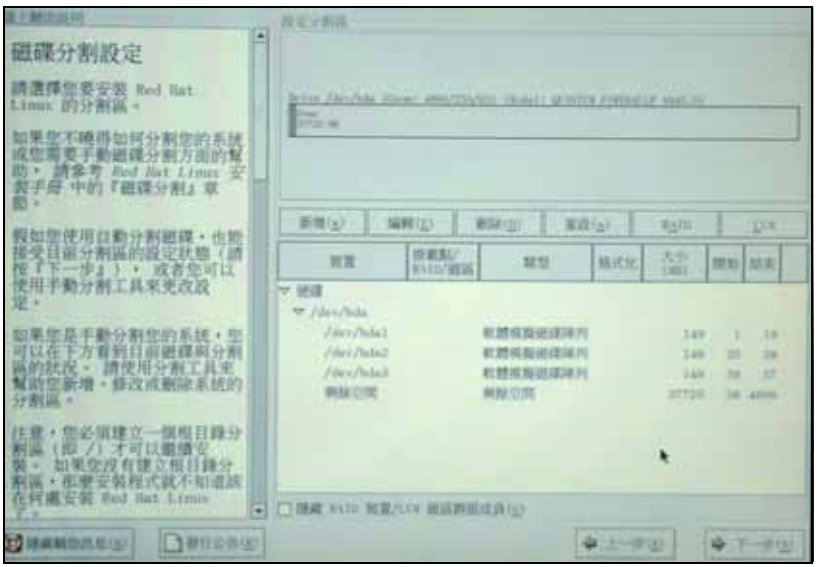
我們選擇建立一個軟體模擬 RAID 分割區。



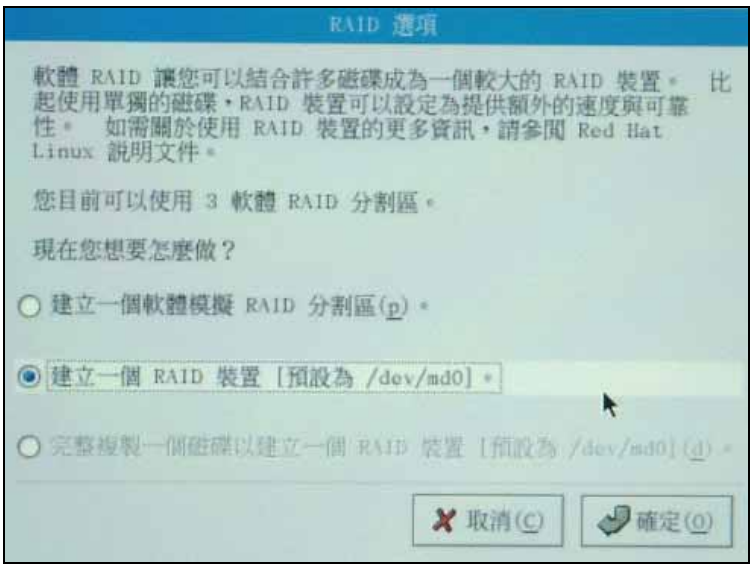
我們選取檔案系統為 software RAID，再設定大小為 150MB。



這就是我們建立的三個軟體磁碟陣列，裝置是/dev/hda1、/dev/hda2 和/dev/hda3 這三個。



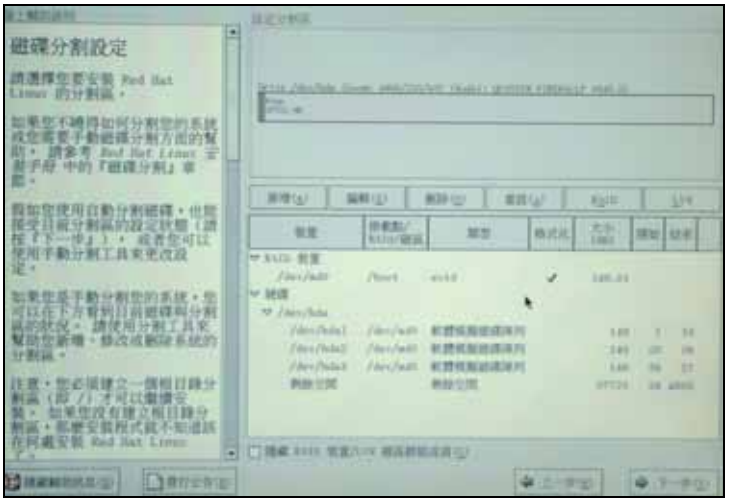
我們現在要建立一個 RAID 裝置。



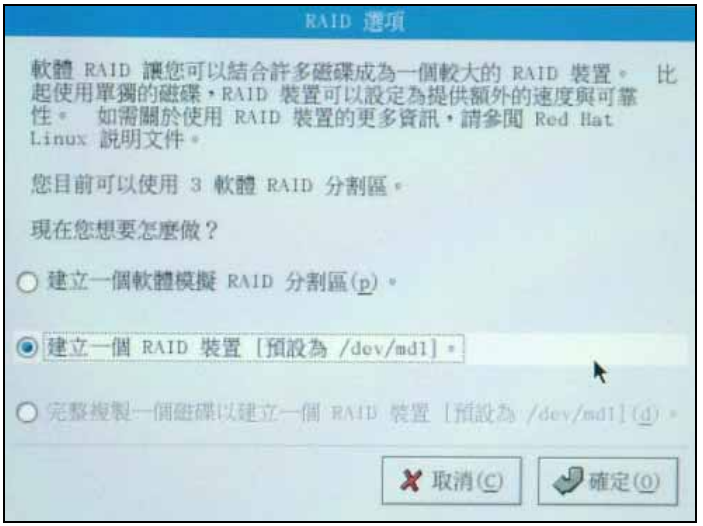
我們檔案系統的掛載點是/boot，檔案類型是ext3，RAID裝置是md0，RAID類型為RAID1，RAID成員是hda1、hda2 和hda3 這三個，備援裝置的數量是 1。RAID類型RAID0 是平行儲存，但沒有容錯的能力。RAID1 藉由寫入相同的資料到陣列中的每一個成員磁碟中以提供容錯能力, 因此hda1、hda2 和hda3 會組成md0 磁碟。RAID5 是最普遍被使用的磁碟陣列模式，藉由分散同位元檢測資料區塊 parity 的資訊到陣列中某些或所有的成員磁碟機中，硬體 RAID level 5 的儲存容量等於所有成員磁碟機的容量減掉一個成員磁碟機的容量。假如我們要建立一個 /boot 的磁碟陣列分割區, 我們必定要選擇 RAID level 1, 而且它必須使用前面的兩個磁碟機的其中一個 (第一個 IDE 或第二個 SCSI)。 假如我們沒有設定一個 /boot 的磁碟陣列分割區，而我們要建立一個 / 的磁碟陣列分割區，此時必須設定為 RAID level 1 而且它必須使用前面的兩個磁碟機的其中一個 (第一個 IDE 或第二個 SCSI)。

這是我們所設定的 md0 裝置。假如我們設定為 RAID 1 或 RAID 5，請指定備援(spare)分割區的數量。當一個軟體磁碟陣列分割區失敗時，將會自動使用備援分割區來當作一個替代品。對於我們想要指定的每一個備援裝置，我們必須要建立一個額外的軟體磁碟陣列分割區（在當作磁碟陣列裝置的分割區之外）。

這是我們新增的 RAID 裝置/dev/md0。三個軟體模擬磁碟陣列組成一個/dev/md0 磁碟陣列裝置。其中一個軟體磁碟陣列是當作備援裝置。

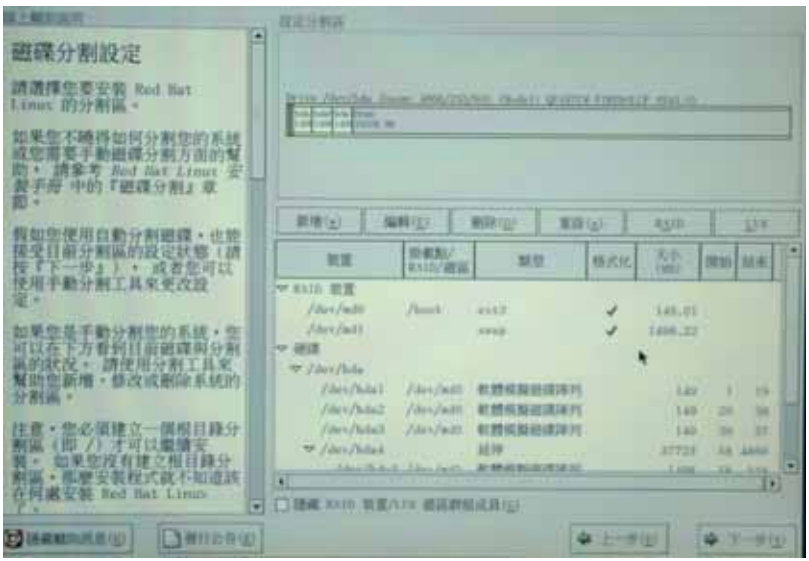


重複以上步驟來建立我們的 swap 磁碟陣列裝置。因此我們建立三個 1500MB，檔案系統為 swap 的軟體模擬磁碟陣列。我們再建立一個 swap 的 RAID 裝置，預設為/dev/md1。

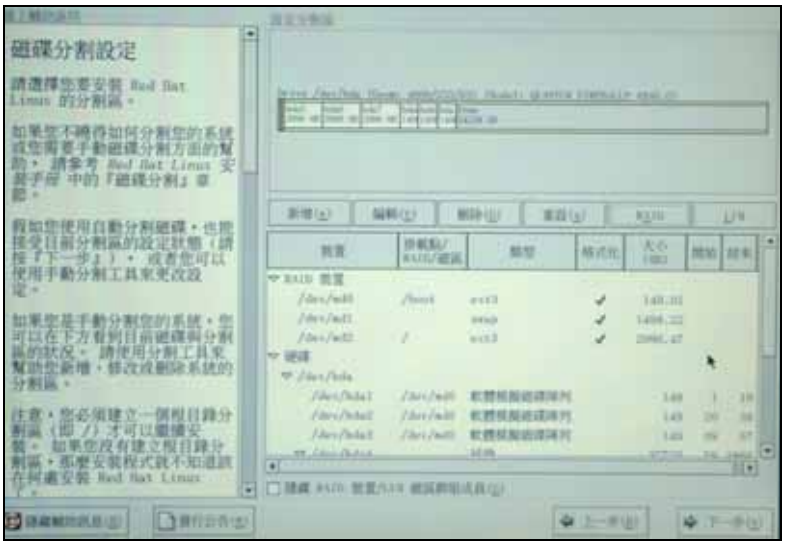


檔案類型是 swap，RAID 裝置是 md1，RAID 類型為 RAID1，RAID 成員是 hda5、hda6 和 hda7 這三個，備援裝置的數量是 1。

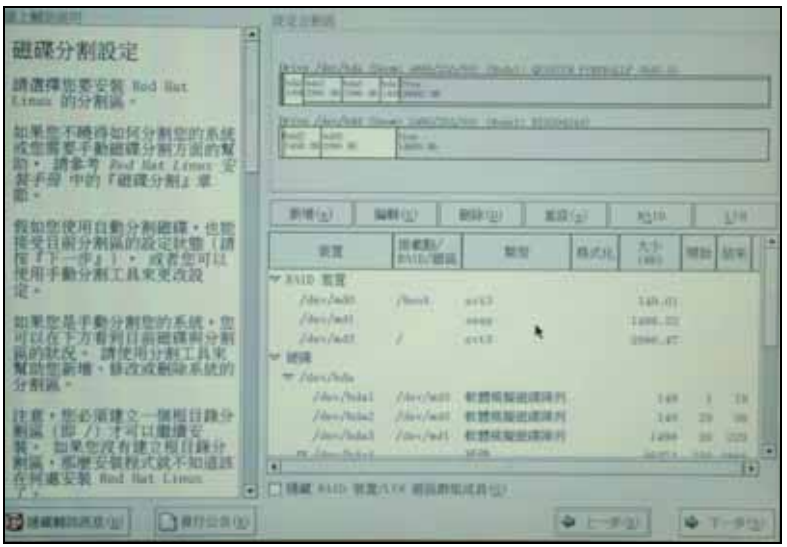
這是我們新增的 RAID 裝置/dev/md1。三個軟體模擬磁碟陣列組成一個/dev/md1 磁碟陣列裝置。其中一個軟體磁碟陣列是當作備援裝置。/dev/md1 裝置的檔案類型為 swap。



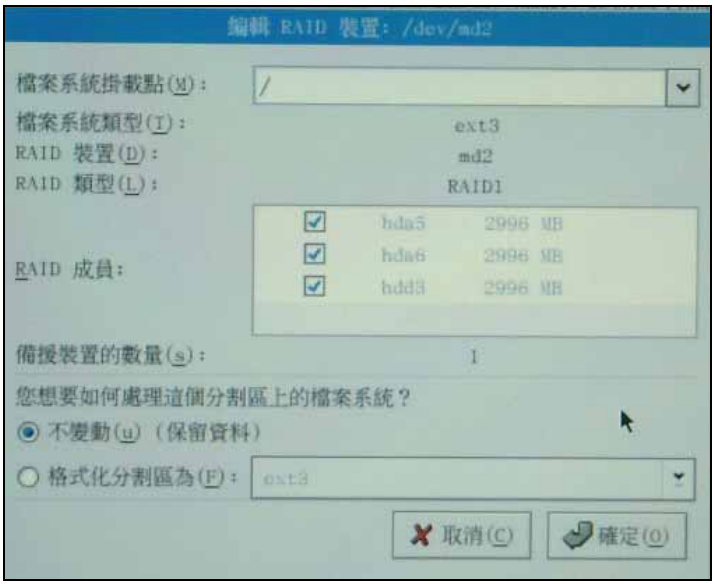
重複以上步驟來建立我們的/dev/md2 磁碟陣列裝置。因此我們建立三個3000MB，檔案系統掛載點為/根目錄的軟體模擬磁碟陣列。



這是兩個或多個磁碟機的設定方法。我們可以將軟體模擬磁碟陣列用在多個不同磁碟機，然後再整合到同一個 RAID 裝置。



hda5 和 hda6 為第一顆磁碟機，hdd3 為第二棵磁碟機，我們將它整合到 RAID 裝置 md2。



32-4 新增磁碟陣列的檔案系統

我們要在/dev/had 第一顆硬碟增加兩個 3000M 的分割區。

```
[root@mandrake /]# /sbin/fdisk /dev/hda

The number of cylinders for this disk is set to 4866.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)
```

我們使用 n 來新增加分割區，並且設定其分割大小為 3000M。其裝置在磁碟機上顯示為/dev/hda10。我們使用 P 指令就可以看到下次要新增的分割區。

```
Command (m for help): n
First cylinder (1917-4866, default 1917):
Using default value 1917
Last cylinder or +size or +sizeM or +sizeK (1917-4866, default 4866): +3000M

Command (m for help): p

Disk /dev/hda: 40.0 GB, 40027029504 bytes
255 heads, 63 sectors/track, 4866 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hda1   *           1          19      152586   fd  Linux raid autodetect
/dev/hda2              20          38      152617+  fd  Linux raid autodetect
/dev/hda3              39         229     1534207+  fd  Linux raid autodetect
/dev/hda4             230        4866    37246702+   5  Extended
/dev/hda5             230         611     3068383+  fd  Linux raid autodetect
/dev/hda6             612         993     3068383+  fd  Linux raid autodetect
/dev/hda7             994        1184     1534176   fd  Linux raid autodetect
/dev/hda8            1185        1550     2939863+  83  Linux
/dev/hda9            1551        1916     2939863+  83  Linux
/dev/hda10           1917        2282     2939863+  83  Linux
```

我們使用 n 來新增加分割區，並且設定其分割大小為 3000M。其裝置在磁碟機上顯示為/dev/hda11。我們使用 w 指令來寫入並且離開。

```
Command (m for help): n
First cylinder (2283-4866, default 2283):
Using default value 2283
Last cylinder or +size or +sizeM or +sizeK (2283-4866, default 4866): +3000M

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
```

我們現在要在第二顆硬碟/dev/hdc 新增一個 3000M 的分割區。

```
[root@mandrake /]# /sbin/fdisk /dev/hdc

The number of cylinders for this disk is set to 59554.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)
```

我們使用 n 來新增分割區，我們使用 p 來設定主要分割區，我們在/dev/hdc 的硬碟裝置上新增 3000M 的分割區，其裝置為/dev/hdc2。

```
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 3
First cylinder (5815-59554, default 5815):
Using default value 5815
Last cylinder or +size or +sizeM or +sizeK (5815-59554, default 59554): +3000M
```

我們使用/sbin/mke2fs –j　/dev/hda10 將裝置/dev/hda10 的裝置格式化成 ext3 的檔案系統。

```
[root@mandrake /]# /sbin/mke2fs -j /dev/hda10
mke2fs 1.32 (09-Nov-2002)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
368000 inodes, 734965 blocks
36748 blocks (5.00%) reserved for the super user
First data block=0
23 block groups
32768 blocks per group, 32768 fragments per group
16000 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912

Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 35 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```

我們使用 mke2fs –j　/dev/hda11 將裝置/dev/hda11 的裝置格式化成 ext3 的檔案系統。

```
[root@mandrake /]# /sbin/mke2fs -j /dev/hda11
mke2fs 1.32 (09-Nov-2002)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
368000 inodes, 734965 blocks
36748 blocks (5.00%) reserved for the super user
First data block=0
23 block groups
32768 blocks per group, 32768 fragments per group
16000 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912

Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 24 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```

我們使用 mke2fs –j　/dev/hdc2 將裝置/dev/hdc2 的裝置格式化成 ext3 的檔案系統。

```
[root@mandrake /]# /sbin/mke2fs -j /dev/hdc2
mke2fs 1.32 (09-Nov-2002)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
366528 inodes, 732564 blocks
36628 blocks (5.00%) reserved for the super user
First data block=0
23 block groups
32768 blocks per group, 32768 fragments per group
15936 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912

Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 32 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```

我們修改磁碟陣列的組態檔/etc/raidtab，我們新增/dev/md5 的磁碟陣列裝置，它是屬於 raid-level1，而且有一個備援，因此/dev/hda10、/dev/hda11 和/dev/hdc2 共同組成 Raid1 的/dev/md5 磁碟陣列裝置。

vi /etc/raidtab

```
raiddev                 /dev/md5
raid-level                     1
nr-raid-disks                  2
chunk-size                     64k
persistent-superblock          1
nr-spare-disks                 1
    device              /dev/hda10
    raid-disk     0
    device              /dev/hda11
    raid-disk     1
    device              /dev/hdc2
    spare-disk     0
```

設定好磁碟陣列組態後。我們使用 mkraid 指令來初始化並且更新磁碟陣列/dev/md5 的裝置。

/sbin/mkraid /dev/md5

```
[root@mandrake /]# /sbin/mkraid /dev/md5
handling MD device /dev/md5
analyzing super-block
disk 0: /dev/hda10, 2939863kB, raid superblock at 2939776kB
disk 1: /dev/hda11, 2939863kB, raid superblock at 2939776kB
disk 2: /dev/hdc2, 2930256kB, raid superblock at 2930176kB
```

我們使用 mke2fs –j　/dev/md5 將裝置/dev/md5 的裝置格式化成 ext3 的檔案系統。

/sbin/mke2fs -j /dev/md5

```
[root@mandrake /]# /sbin/mke2fs -j /dev/md5
mke2fs 1.32 (09-Nov-2002)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
366528 inodes, 732544 blocks
36627 blocks (5.00%) reserved for the super user
First data block=0
23 block groups
32768 blocks per group, 32768 fragments per group
15936 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912

Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 39 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```

我們在根目錄新增 raid 目錄，並且使用 mount 指令將它掛載上去。

mkdir raid

mount -t ext3 /dev/md5 /raid

```
[root@mandrake /]# df -k
檔案系統              1K-區段      已用      可用 已用% 掛載點
/dev/md2              3020048    720788   2145848   26% /
/dev/md0               147692      9284    130783    7% /boot
none                   256916         0    256916    0% /dev/shm
/dev/hdc1             2884176     32828   2704840    2% /newdisk
/dev/md5              2884128     32828   2704792    2% /raid
```

我們也可以修改/etc/fstab，並且新增最後一行掛載/dev/md5 的裝置，這樣開機時就會自動掛載/dev/md5 的裝置到/raid 目錄了。

vi /etc/fstab

```
/dev/md2                /                       ext3    defaults          1 1
/dev/md0                /boot                   ext3    defaults          1 2
none                    /dev/pts                devpts  gid=5,mode=620    0 0
none                    /proc                   proc    defaults          0 0
none                    /dev/shm                tmpfs   defaults          0 0
/dev/md1                swap                    swap    defaults          0 0
/dev/cdrom              /mnt/cdrom      udf,iso9660 noauto,owner,kudzu,ro 0 0
/dev/hdc1               /newdisk                ext3    defaults          1 1
/dev/md5                /raid                   ext3    defaults          1 1
```

32-5 新增 Raid5 磁碟陣列

Raid5 磁碟碟陣列需要至少 3 顆硬碟。

我們使用 dmesg 查詢目前的磁碟裝置。

#dmesg|grep hd|more

我們使用 fdisk 來分割硬碟。

#/sbin/fdisk　/dev/hda

#/sbin/fdisk　/dev/hdb

#/sbin/fdisk　/dev/hdc

我們使用 mke2fs 來建立 ext3 的檔案系統。

#/sbin/mke2fs –j /dev/hda

#/sbin/mke2fs –j /dev/hdb

#/sbin/mke2fs –j /dev/hdc

修改磁碟陣列的組態檔/etc/raidtab。raid-level 是指定磁碟陣列的等級。nr-raid-disks 是指示磁碟陣列裝置的數量。

#vi /etc/raidtab

```
raiddev /dev/md1
    raid-level              5
    nr-raid-disks           3
    nr-spare-disks          1
    persistent-superblock   1
    parity-algorithm        left-symmetric

    device                  /dev/sda1
    raid-disk               0
    device                  /dev/sdb1
    raid-disk               1
    device                  /dev/sdc1
    raid-disk               2
    device                  /dev/sdd1
    spare-disk              0
```

指令	說明
raiddev	指示裝置的區段
nr-raid-disks	指示磁碟陣列裝置的數量
raid-level	指定磁碟陣列的等級。
nr-spare-disks	指示備援磁碟機的數量
persistent-superblock 0/1	Persistent-superblock 提供核心安全的偵測磁碟裝置，因此新建立的磁碟裝置都應該要支援。
parity-algorithm	Raid5 的同位元檢測演算法。left-symmetric 在傳統的磁碟轉盤上提供高的效能。
chunk-size	設定分割片段的大小。
device	指定磁碟的裝置。
raid-disk	指定裝置安插到磁碟陣列的磁碟編號。
spare-disk	指定裝置安插到磁碟陣列的備援磁碟編號。從 0 開始編號。

我們使用 mkraid 來初始化或更新磁碟陣列裝置。mkraid 指令可以將各個磁碟區塊整合成一個單一的磁碟陣列裝置。

#/sbin/mkraid　/dev/md1

我們將磁碟陣列裝置/dev/md1 掛載到/home 目錄下。

#mount –t ext3 /dev/md1 /home

課後練習

1. 磁碟陣列的基本概念是結合多個小型且便宜的磁碟機成為一個陣列, 以達到一個大且昂貴的磁碟機無法做到的效能表現或多餘性的目標。 這個磁碟機的陣列將會以一個單一的邏輯儲存單位或磁碟機呈現在電腦中。因此磁碟陣列(Redundant Array Inexpensive Disk)就是使用多顆較便宜的磁碟組成一個容量大, 安全性高的整合性磁碟機。請問哪一種版本的 RAID 可以將多顆小的磁碟組成一顆大容量的磁碟，這樣就可以讓/var 目錄的磁碟空間加大?

(A). RAID0

(B). RAID1

(C). RAID2

(D). RAID3

2. Level 5 — 磁碟陣列 5 是最普遍被使用的磁碟陣列模式，藉由下列哪一個的資訊寫到陣列中某些或所有的成員磁碟機中? RAID level 5 減少了在 level 4 中存在的寫入瓶頸，在這裏我們使用 parity 來代表同位元檢測資料區塊。 僅有的效能表現瓶頸在於 parity 計算的過程，不過如果使用現今相當快的 CPU 加上軟體 磁碟陣列設定，這通常不是一個很大的問題。

(A). 平行儲存

(B). 線性儲存

(C). 分散同位元檢測資料區塊 parity

(D). 映設儲存

3. Linux 核心中的下列何者驅動裝置是一種完全與硬體無關的磁碟陣列解決方案例子? 軟體為基礎之陣列的效能表現，是依賴在伺服器的 CPU 效能與負載。

(A). A./dev/raid

(B). B./dev/md

(C). C./dev/mirror

(D). D./dev/parity

4. 下列何者通常稱為『平行儲存』，它是一種以效能為導向的資料條狀分佈儲存的技術？將要寫入到陣列的資料會先劃分為條狀，再寫入到陣列中的成員磁碟，這個方法以較低的固有開支提供了相當高的 I/O 存取效能，不過並沒有任何的容錯能力(非重複性的儲存)redundancy。

(A). RAID3

(B). RAID2

(C). RAID1

(D). RAID0

5. 修改磁碟陣列的組態檔/etc/raidtab。raid-level 是指定磁碟陣列的等級。nr-raid-disks 是指示磁碟陣列裝置的數量。請問 spare-disks 備援磁碟的編號是從哪一個開始?

(A). A.-1

(B). B.2

(C). C.1

(D). D.0

6. 我們使用下列哪一個裝置來初始化或更新磁碟陣列裝置？該指令可以將各個磁碟區塊整合成一個單一的磁碟陣列裝置。

(A). fdisk

(B). mkdir

(C). mkraid

(D). mount

【答案】

1.→A　2.→C　3.→B　4.→D　5.→D　6.→C

NOTE