Sun StorEdge™ Availability Suite 3.2 Remote Mirror 軟體配置指南

Sun Microsystems, Inc.
www.sun.com

文件號碼：817-4791-10
2003 年 12 月，修訂版 A

請將關於本文件的意見傳送至：http://www.sun.com/hwdocs/feedback

目錄

前言

*《Sun StorEdge™ Availability Suite 3.2 Remote Mirror 軟體配置指南》*提供了本軟體的快速設定與使用資訊。

使用 UNIX 指令

本文件可能不包括有關基本 UNIX® 指令及程序的資訊，例如關閉系統、啓動系統及配置裝置。請參閱以下文件資料以取得相關資訊：

- 系統隨附的軟體文件資料
- Solaris™ 作業環境的文件資料（位於下列網址）

  `http://docs.sun.com`

Shell 提示符號

Shell	提示符號
C shell	*機器名稱*%
C shell 超級使用者	*機器名稱*#
Bourne shell 與 Korn shell	$
Bourne shell 與 Korn shell 超級使用者	#

印刷排版慣例

字體*	意義	範例
AaBbCc123	指令、檔案和目錄的名稱；電腦螢幕的輸出。	編輯您的 .login 檔案。 使用 ls -a 列出所有檔案。 % You have mail.
AaBbCc123	您鍵入的內容，與電腦螢幕輸出不同。	% **su** Password:
AaBbCc123	書名、新字或專有名詞、或要強調的文字。以實際的名稱或數值取代指令行變數。	請參考 *《使用者指南》*中的第六章。 這些是*類別*選項。 您*必須*是超級使用者才能執行此項操作。 若要刪除檔案，請鍵入 rm *檔案名稱*。

* 您瀏覽器的設定可能與上述設定不同。

相關文件資料

適用範圍	書名	文件號碼
線上說明手冊	sndradm iiadm dsstat kstat svadm	無
最新版次資訊	*《Sun StorEdge Availability Suite 3.2 軟體版次注意事項》*	817-4776
	《Sun Cluster 3.0/3.1 和 Sun StorEdge Availability Suite 3.2 軟體版次注意事項補充資料》	817-4786
安裝和使用者	*《Sun StorEdge Availability Suite 3.2 軟體安裝指南》*	817-4766
系統管理	*《Sun StorEdge Availability Suite 3.2 Point-in-Time Copy 軟體管理與操作指南》*	817-4761
	《Sun StorEdge Availability Suite 3.2 Remote Mirror 軟體管理與操作指南》	817-4771

存取 Sun 文件資料

若要檢視、列印或購買各種精選的 Sun 文件資料及其本土化版本，請至：

`http://www.sun.com/documentation`

聯絡 Sun 技術支援

若本文件無法解決您對本產品相關技術上的疑惑，請至下列網址尋求協助：

`http://www.sun.com/service/contacting`

Sun 歡迎您的指教

Sun 一直致力於改善相關的文件資料，因此歡迎您提出批評和建議。您可至下列網站留下您的意見：

`http://www.sun.com/hwdocs/feedback`

請在您的意見中註明本文件的書名和文件號碼：

《Sun StorEdge Availability Suite 3.2 Remote Mirror 軟體配置指南》，
文件號碼：817-4791-10

配置 Remote Mirror 軟體

Sun StorEdge™ Availability Suite 3.2 Remote Mirror 軟體爲用於 Solaris™ 8 和 9（Update 3 和更新版次）作業系統的容體等級*複製*軟體。Remote Mirror 軟體會以即時方式複製實體上分離的*主要*與*次要*站點之間的磁碟容體寫入作業。Remote Mirror 軟體可與所有 Sun™ 網路配接卡及支援 TCP/IP 的網路連結一起使用。

由於本軟體是以容體爲基礎的 (volume-based)，因此它是儲存體獨立的，並可支援原始容體或各種 Volume Manager 軟體 － 無論是 Sun 或是協力廠商的產品。此外，本產品也支援各種應用程式，或具有寫入資料的 Solaris 系統之單一主機的資料庫。但不支援那些已配置成允許多部主機在 Solaris 系統上寫入資料至共享容體的資料庫、應用程式或檔案系統（例如：Oracle 9iRAC、Oracle Parallel Server）。

作爲災後復原和企業永續方案的一部分，Remote Mirror 軟體能讓您在遠端站點保有重要資料的最新副本。Remote Mirror 軟體能讓您演練並測試企業永續方案。如需可行度高的方案，可以將 Sun StorEdge Availability Suite 軟體配置成在 Sun Cluster 3.x 環境中進行錯誤移轉。

當您的應用程式正在存取資料容體、連續將資料複製到遠端站點或記錄變更時（可在稍後進行快速重新同步化），Remote Mirror 軟體爲運作中的狀態。

Remote Mirror 軟體能讓您以手動的方式從主要站點到次要站點（所謂的*正向同步化*），或從次要站點到主要站點（所謂的*反向同步化*）初始啓動重新同步化。

Remote Mirror 軟體中的複製與配置是以容體集爲基礎而進行的。一個 Remote Mirror 容體集包含一個主要容體、一個次要容體、一個位於主要與次要站點的點陣圖容體（用以追蹤及記錄快速重新同步化的變更），以及一個用於*非同步複製*模式的選擇性*非同步佇列*容體。推薦您使用相同大小的主要與次要容體。您可以使用 `dsbitmap` 工具來判定點陣圖容體的大小需求。如需更多關於配置 Remote Mirror 容體集或 `dsbitmap` 工具的資訊，請參閱 *《Sun StorEdge Availability Suite 3.2 Remote Mirror 軟體管理與操作指南》*。

操作理論

複製可同步或非同步進行。在同步模式中，應用程式寫入作業要等到將寫入作業交給主要與次要主機後才會進行確認。在非同步模式中，應用程式寫入作業會在將寫入作業交給本端儲存體及寫入至非同步佇列時即進行確認。此佇列會使得寫入至次要站點的寫入作業非同步。

同步複製

同步作業的資料流向如下：

1. 在點陣圖容體中設定記錄日誌位元。
2. 平行初始化本端寫入作業與網路寫入作業。
3. 當這兩項寫入作業完成時，則會清除記錄日誌位元 (*lazy clear*)。
4. 應用程式確認寫入作業。

*同步複製*的優點爲主要與次要站點兩者皆能隨時處於同步狀態。此類型的複製只有當連結的延遲低，且連結可符合應用程式的頻寬需求時才實用。這些限制通常會使同步方案限於校園或都會地區內。

在此情況下，一個寫入作業的平均服務時間則爲：

點陣圖寫入 + MAX （本端資料寫入、網路來回傳輸時間 + 遠端資料寫入）

在校園和都會地區中，網路的來回傳輸時間是無關緊要的，且服務時間大約是在 Remote Mirror 軟體尚未安裝時所觀測到的兩倍。

假設一次寫入需要 5 毫秒，則：

5 毫秒 + MAX （5 毫秒、1 毫秒 + 5 毫秒） = 11 毫秒

注意 – 在輕量載入的系統上，5 毫秒這個值是合理的假設。在更符合實際載入情形的系統上，佇列積存 (backlog) 會使此值增高。

然而若網路來回傳輸時間約爲 50 毫秒（在遠距複製時大多都是這樣），網路延遲會使得同步方案變得不實用。詳情請參見下列範例：

5 毫秒 + MAX （5 毫秒、50 毫秒 + 5 毫秒） = 60 毫秒

非同步複製

非同步複製會將遠端寫入作業與應用程式寫入作業加以區隔。在此模式中，會在網路寫入作業新增至非同步佇列時進行確認。這表示要等到所有寫入作業都已傳送至次要站點時，次要站點才能離開與主要站點的同步作業。在此模式中，資料的流向如下：

1. 設定記錄日誌位元。
2. 本端寫入 — 非同步佇列寫入作業以平行的方式進行。
3. 應用程式確認寫入作業。
4. 清理器 (flusher) 執行緒讀取非同步佇列項目及平台網路寫入。
5. 清除記錄日誌位元 (lazy clear)。

服務時間爲下列所需的時間：

點陣圖寫入 + MAX（本端寫入、非同步佇列項目資料）

在寫入作業使用 5 毫秒的服務時間，估計用於非同步寫入作業的服務時間爲：

5 毫秒 + MAX（5 毫秒、5 毫秒）= 10 毫秒

若在一段時間後寫入速率超過容體或*一致性群組*的網路排出 (drain) 速率，則非同步佇列會開始堆積。適當的大小是很重要的，因此本文件在稍後會討論估計適當容體大小的方法。

下列爲控制 Remote Mirror 軟體在發生非同步磁碟佇列堆積時如何反應的兩種模式：

- *阻攔模式*

  在阻攔模式中（此爲預設的設定），Remote Mirror 軟體會阻攔並等候非同步磁碟佇列排出到某種程度後，才會將寫入新增至非同步佇列。這會影響應用程式的寫入作業，但仍會維持連結上的寫入次序。

- *非阻攔模式*

  在非阻攔模式中（以記憶體爲基礎的佇列中無法使用），當磁碟非同步佇列堆積時，Remote Mirror 軟體不會阻攔，但會進入*日誌*模式並記錄寫入。在後續的*更新同步化*中，這些會從位元 0 往前讀取，而且不會保留寫入次序。若使用此模式、若非同步磁碟佇列堆積且寫入次序消失，相關容體或一致性群組則會變得不一致。強烈建議您於啓動更新同步化之前在次要站點製作一份 Point-in-Time 副本；例如，使用自動同步化常駐程式。

一致性群組

在同步模式中，會確保橫跨許多容體的應用程式之寫入次序，因爲在需要排定次序時，應用程式會等到完成後再發放另一個 I/O 作業。而 Remote Mirror 軟體則要等到寫入作業到達主要與次要站點時才會發出完成訊號。

在非同步模式中（根據預設），各個容體的佇列會被一個或多個獨立執行緒排出。由於此作業是與應用程式分離的，因此不會保留寫入至多個容體的寫入作業之寫入次序。

若應用程式需要排定寫入次序，Remote Mirror 軟體可提供一致性群組的功能。每個一致性群組都有單一的網路佇列，且雖然可允許平行執行多個寫入作業，但仍會藉由使用序號保留寫入次序。

規劃遠端複製

當您在規劃遠端複製時，請考量您的企業需求、應用程式寫入工作量、以及您網路的特性。

企業需求

當您決定複製企業資料時，請考量最大的延遲時間：對於次要站點上的資料，您能容許怎樣的過期程度？這是決定複製模式與快照排程的因素。此外務必要知道，您正在複製的應用程式是否需要寫入作業在寫入至次要容體時以正確的次序複製。

應用程式寫入工作量

在決定主要與次要站點之間需要的網路連線類型時，您務必要瞭解平均和尖峰寫入工作量。若要決定配置，請收集下列資訊：

- 資料寫入作業的平均速率與大小

  平均速率爲應用程式處於典型工作量時的資料寫入作業量。應用程式讀取作業對預備及規劃遠端複製而言是不重要的。

- 尖峰速率與資料寫入作業的大小

  尖峰速率爲在一段測量持續時間內應用程式所寫入的最大量資料。

- 尖峰寫入速率的持續時間與頻率

  持續時間是指尖峰寫入速率能維持多久，而頻率則是指此情況有多常發生。

若無法得知這些應用程式特性，您可以使用 `iostat` 或 `sar` 等工具測量應用程式執行時的寫入流量。

網路特性

當您得知應用程式寫入工作量時，請判定網路連結的需求。要考慮的最重要網路內容爲網路頻寬及主要與次要站點之間的網路延遲。若網路連結在安裝 Sun StorEdge Availability Suite 軟體前即已存在，您可以使用 `ping` 工具來協助判定站點之間的連結特性。

若要使用同步複製，網路延遲必須低到足以使應用程式回應時間不受每個寫入作業的網路來回傳輸時間之動態影響。此外，網路頻寬必須足以處理在應用程式的尖峰寫入期間所產生的寫入流量。若網路無法隨時處理寫入流量，應用程式回應時間將會受到影響。

若要使用非同步複製，網路連結的頻寬必須能夠處理在應用程式的平均寫入期間所產生的寫入流量。在應用程式尖峰寫入階段期間，超過限度的寫入作業會先被寫入至本端的非同步佇列，然後稍後當網路流量允許時再寫入至次要站點。只要非同步佇列的大小得宜，應用程式回應時間在寫入流量激增超過網路限制時就可被縮到最短。

請參閱本文件第 5 頁的「配置非同步佇列」一節。選取的 Remote Mirror 非同步選項模式（阻攔或非阻攔）會決定軟體如何應對佇列堆積。

配置非同步佇列

若您是使用非同步複製，請規劃本節所述的配置設定。這些設定是以 Remote Mirror 容體集或一致性群組爲基礎而設置的。

磁碟或記憶體佇列

在本軟體的 3.2 版中，Remote Mirror 軟體已新增對以磁碟爲基礎的非同步佇列之支援。爲了方便從先前版本進行升級，以記憶體爲基礎的佇列仍會受到支援，但新增之以磁碟爲基礎的佇列則會提供建立更大、更有效率的佇列之功能。更大的佇列可允許更大量的激增寫入作業，而不會影響應用程式回應時間。此外，與以記憶體爲基礎的佇列相比，以磁碟爲基礎的佇列對系統資源較不會造成影響。

非同步佇列的大小必須足以處理在應用程式尖峰寫入期間的激增寫入流量。大的佇列可以處理延長的激增寫入作業，但也會允許次要站點更遠離與主要站點的同步化之可能性。您可以依照尖峰寫入速率、尖峰寫入持續時間、寫入大小、以及網路連結特性來判定佇列大小應當如何。請參閱第 9 頁的「將以磁碟爲基礎的非同步佇列設成正確的大小」。

您所選取的佇列選項（阻攔或非阻攔）會決定軟體如何應對堆積的磁碟佇列。使用 `dsstat` 工具來判定非同步佇列的統計資料，包括高水印 (high-water mark, `hwm`)，它會顯示曾使用的最大量佇列。若要將非同步佇列新增至 Remote Mirror 容體集或一致性群組，請使用 `sndradm` 指令及 `-q` 選項：`sndradm -q a`

佇列大小

使用 `dsstat`(1SCM) 指令監視非同步佇列以檢查*高水印* (`hwm`)。若 `hwm`（由於應用程式寫入超過佇列所能處理的資料而造成）經常達到佇列總大小的 80% 至 85%，請增加佇列大小。此原則可應用於以磁碟爲基礎的佇列及以記憶體爲基礎的佇列。然而，重新調整各種佇列類型的大小則需使用不同的程序。

以記憶體爲基礎的佇列

- 佇列中預設的最大寫入作業量（可調整）爲 4096。您可以使用 `sndradm -W` 指令變更此值。
- 預設最大量的 512 位元組資料區塊（預設佇列大小）（可調整）爲 16384，其約爲 8 MB 的資料。您可以使用 `sndradm -F` 指令變更此值。

以磁碟爲基礎的佇列

磁碟佇列的有效大小爲磁碟佇列容體的大小。磁碟佇列的大小只能透過將其容體更換爲不同大小的容體來重新調整。例如，對於 16384 個區塊的佇列大小，請確認 `hwm` 未超過 13000 個至 14000 個區塊。如果它超過了這個量，請使用下列程序重新調整佇列大小。

注意 – 最大的磁碟佇列大小爲一個少於 1 TB 的區塊（或 2147483647 個區塊）。請勿使用大於最大之大小的容體。

▼ 重新調整佇列的大小

1. **將容體置於日誌模式（使用 `sndradm -l` 指令）。**
2. **重新調整佇列的大小。**
   - 以記憶體爲基礎的佇列：使用 `sndradm -F` 指令。
   - 以磁碟爲基礎的佇列：使用 `sndradm -q` 指令將目前的磁碟佇列容體更換爲更大的容體。

3. 使用 sndradm -u 指令執行更新同步化。

▼ 顯示目前的佇列大小、長度及 hwm

1. 鍵入下列內容以顯示佇列大小：

   - 以記憶體爲基礎的：

```
# sndradm -P
/dev/vx/rdsk/data_t3_dg/vol0 -> priv-2-
230:/dev/vx/rdsk/data_t3_dg/vol0
autosync: off, max q writes: 4096, max q fbas: 16384, async
threads:8, mode: async, state: replicating
```

   佇列大小的區塊數是由 max q fbas 所給予的（在此範例中爲 16384 個區塊）。此佇列中所允許的項目最大數量是由 max q writes 所給予的（在此範例中爲 4096）。在此範例中，這表示佇列中的項目之平均大小爲 2K。

   - 以磁碟爲基礎的：

```
# sndradm -P
/dev/vx/rdsk/data_t3_dg/vol0 -> priv-
230:/dev/vx/rdsk/data_t3_dg/vol0
autosync: off, max q writes: 4096, max q fbas: 16384, async
threads: 1, mode: async, blocking diskqueue:
/dev/vx/rdsk/data_t3_dg/dq_single, state: replicating
```

   會顯示磁碟佇列容體 (/dev/vx/rdsk/data_t3_dg/dq_single)。佇列大小可由檢查容體大小來判定。

2. 鍵入下列內容以顯示目前的佇列長度及其 hwm：

```
# dsstat -m sndr -d q
name              q role    qi     qk  qhwi   qhwk
data_a5k_dg/vol0  D  net     4     13     5    118
```

   其中：

   - qi 代表目前佇列中的項目數量。
   - qk 代表目前佇列中的資料總大小（以 KB 計）。
   - qhwi 代表曾在某個時間處於佇列中的最大量項目。
   - qhwk 代表曾在某個時間處於佇列中的最大量資料（以 KB 計）。

3. 若要顯示串流摘要與磁碟佇列資訊，請鍵入：

```
# dsstat -m sndr -r bn -d sq 2
```

4. 若要顯示更多資訊，請執行 dsstat(1SCM) 及其他顯示選項。

範例 dsstat *輸出 — 大小正確的佇列*

注意 – 此範例只顯示本節需要用到的指令輸出部分，實際上 dsstat 指令會顯示更多資訊。

下列 dsstat(1SCM) 核心統計資料輸出會顯示有關非同步佇列的資訊。在這些範例中，佇列的大小是正確的，而且目前並未開始堆積。此範例會顯示下列的設定和統計資料：

以磁碟爲基礎的範例

```
# dsstat -m sndr -r n -d sq -s \ priv-2-230:/dev/vx/rdsk/data_t3_dg/vol67
name              q role    qi     qk  qhwi   qhwk    kps   tps  svt
data_t3_dg/vol67  D  net    48    384   240   1944     10     1   54
```

其中：

- qi 項目表示共有 48 個寫入處理已置入此佇列。
- qk 項目表示已有 384 KB 置入此佇列。
- qhwi 項目表示已佇列項目的 hwm 爲 240 個項目；目前尚未到達。
- qhwk 項目表示已佇列資料（以 KB 計）的 hwm 爲 1944；目前尚未到達。

假設磁碟佇列容體大小爲 1 GB（或 2097152 個磁碟區塊），1944 個區塊的 hwm 爲處於全部的 80% 以下之良好狀態。磁碟佇列針對寫入工作量的大小是正確的。

範例 dsstat *輸出 — 大小不正確的磁碟佇列*

下列 dsstat(1SCM) 核心統計資料輸出會顯示有關非同步佇列的資訊，而其大小不正確。

以記憶體爲基礎的範例

```
# sndradm -P
/dev/vx/rdsk/data_a5k_dg/vol0 -> priv-230:/dev/vx/rdsk/data_a5k_dg/vol0
autosync: off, max q writes:4096, max q fbas: 16384, async threads: 2, mode:
async, state: replicating

# dsstat -m sndr -d sq
name               q role    qi     qk  qhwi   qhwk    kps   tps  svt
data_a5k_dg/vol0   M  net  3609   8060  3613   8184     87    34   57
k/bitmap_dg/vol0     bmp     -      -     -      -       0     0    0
```

此範例會顯示預設的佇列設定，但應用程式寫入超過佇列所能處理的資料。8184 KB 的 qhwk 值與 16384 個區塊 (8192 KB) 的 max q fbas 之間的差異表示應用程式正逐漸接近可允許的最大 512 位元組區塊之限制。有可能之後幾個 I/O 作業不會置入佇列中。

在此情況下，增加佇列大小會是一種可行的解決方法。然而，請考慮提升網路連結品質（如使用較大頻寬的介面）以符合長期效益。或者，請考慮製作 Point-in-Time 容體副本並複製備份容體。請參閱 *《Sun StorEdge Availability Suite 3.2 Point-in-Time Copy 軟體管理與操作指南》*。

摘要

- 若堆積率低於或等於排出率，預設的佇列大小即已足夠。
- 若排出率低於堆積率，增加佇列大小可爲暫行的解決方法。然而，若寫入作業持續進行了一段延長的時間，佇列最後仍會堆積。

將以磁碟爲基礎的非同步佇列設成正確的大小

請考慮下列的範例。在此範例中，iostat 是以小時爲間隔而執行的，以略述將被複製的 I/O 載入。在此範例中，我們假設的爲 DS3 (45Mb/S) 連結。此外，也假設此應用程式使用單一的一致性群組，因此含有單一佇列。

對於該應用程式，在經過 24 小時的統計資料收集並假設這是平日狀況的前提下，您可以判定平均寫入速率、非同步佇列的適當大小、遠端站點的資料在一天過後可能會變得多過時、以及選用的網路頻寬是否適合該應用程式。

時間	kwr/s	wr/s	網路流量	佇列增長	佇列大小
	A	B	C	A/1000 - C)*3600	
6am	0	0	4MB/S		
7am	1000	400	4MB/S		
8am	2000	1000	4MB/S		
9am	2000	1000	4MB/S		
10am	4000	1800	4MB/S		
11am	5000	2400	4MB/S	3.6GB	3.6GB
12pm	1000	400	4MB/S	-10GB	
1pm	1200	600	4MB/S		
2pm	1000	500	4MB/S		
3pm	1200	400	4MB/S		
4pm	2000	600	4MB/S		
5pm	1000		4MB/S		
6pm	800		4MB/S		
7pm	800		4MB/S		
8pm	3200	1000	4MB/S		
9pm	8000	2500	4MB/S	14GB	14GB
10pm	8000	2500	4MB/S	14GB	28GB
11pm	1000	400	4MB/S	-10	18
12pm	0		4MB/S	-14	4
1am	0		4MB/S	-14	
2am	0		4MB/S		
3am	0		4MB/S		
4am	0		4MB/S		
5am	0		4MB/S		
平均頻寬	1.8MB/S				

在填完表格並計算佇列增長及大小後，您即可明顯看出 30 GB 的佇列就已經足夠了。雖然佇列會增加，且次要站點會因而逐漸脫離同步化，但在夜間執行的這批作業會確保佇列到隔天的營業時間即已變空，而使這兩個站點同步。

這個演練也會驗證網路頻寬適合應用程式所產生的寫入工作量。

配置非同步佇列清理器執行緒

Sun StorEdge Availability Suite 3.2 軟體提供設定清理非同步佇列的執行緒數量之功能。變更此數量可容許網路上的每個容體或一致性群組同時存有多重 I/O。次要節點上的 Remote Mirror 軟體會使用序號處理 I/O 的寫入次序。

在決定怎樣的佇列清理器執行緒數量會對您的複製配置最有效益時，您必須考量許多變數。相關的變數包括容體集或一致性群組的數量、可用的系統資源、網路特性、以及是否有檔案系統。若您的容體集或一致性群組數量小，較大數量的清理器執行緒可能會比較具有效益。推薦您執行一些基本測試或以稍微不同的值與此變數原型加以比對，以判定對配置最具效益的設定。

配置知識、網路特性及對 Remote Mirror 軟體的操作可引導您選擇適當的網路執行緒數量。Remote Mirror 軟體利用 Solaris RPC 作爲傳輸機制：這些 RPC 是處於同步化的狀態。對於每個網路執行緒，個別執行緒可達到的最大流量爲 I/O 大小 / 來回傳輸時間。請考慮優於 2K I/O 的工作量，和 60 毫秒的來回傳輸時間。每個網路執行緒都能：

2K/0.060S = 33K/S

假設有單一容體、或在單一的一致性群組中有許多容體，您可能會發現預設的 2 個網路執行緒會將網路複製限制爲 66K/S。因此比較建議您將此數量調高。若複製網路預備爲 4MB/S，則理論上就 2K 工作量而言，網路執行緒最佳的數量則爲：

(4096K/S) / (2K/0.060 IO/S) = 123 個執行緒

這是假設線性的延展性。實際上，已觀察到增加超過 64 個網路執行緒並不會產生效益。考慮在沒有一致性群組的情況下、30 個容體在 4MB/S 連結上進行複製、以及 8K I/O。每個容體有 2 個網路執行緒的預設會產生 60 個網路執行緒，且若工作量平均地散佈到這些容體上，理論上的頻寬則爲：

60 * (8K / 0.060 IO/S) = 8MB/S

這已超過網路頻寬。不需要進行調整。

非同步佇列清理器執行緒數量的預設設定爲 2。若要變更此設定，您需要使用 `sndradm` CLI 及 `-A` 選項。`-A` 選項的說明爲：`sndradm -A` 會指定當容體集正在非同步模式中進行複製（預設 2）時，可建立來處理非同步佇列的執行緒之最大數量。

若要判定目前已配置成伺服非同步佇列的清理器執行緒數量，您可以使用 sndradm -P 指令。例如，您會發現下面的容體集配置了 2 個非同步清理器執行緒。

```
# sndradm -P
/dev/md/rdsk/d52 -> lh1:/dev/md/sdsdg/rdsk/d102
autosync: off, max q writes: 4096, max q fbas: 16384, async threads: 2, mode:
async, group: butch, blocking diskqueue: /dev/md/rdsk/d100, state: replicating
```

以下爲有關如何使用 sndradm -A 選項來將非同步佇列清理器執行緒變更爲 3 個的範例：

```
# sndradm -A 3 lh1:/dev/md/sdsdg/rdsk/d102
```

調整網路

Remote Mirror 軟體本身會直接注入系統的 I/O 路徑，監視所有流量以判定其可被 Remote Mirror 容體視爲目標。將會追蹤 Remote Mirror 容體的目標 I/O 指令，並管理這些寫入作業的複製。由於 Remote Mirror 軟體是直接處於系統的 I/O 路徑中，因此可能會對系統產生某些效能影響。網路複製所需的額外 TCP/IP 程序也會消耗主機 CPU 資源。

請在主要與次要 Remote Mirror 主機上執行本節所述的程序。

TCP 緩衝區大小

TCP 緩衝區 大小爲在等候確認前，傳輸控制通訊協定所允許傳輸的位元組數量。若要取得最大流量，請務必採用目前使用的連結之最佳 TCP 傳送與接收虛擬連接埠緩衝區大小。若緩衝區太小，TCP 擁塞視窗將永遠無法完全開啓。若接收端緩衝區太大，TCP 流量控制將會中斷，且傳送端會超過接收端而導致 TCP 視窗關閉。若傳送主機比接收主機更快，則有可能發生此事件。只要您有多餘的記憶體，在傳送端的過大視窗就不會造成問題。

注意 – 將緩衝區大小增加到高過共用網路的值可能會影響網路效能。如需關於調整大小的資訊，請參閱 《Solaris System Administrator》文件資料集。

表 1 顯示 100BASE-T 網路的最大可能流量。

表 1　網路流量與緩衝區大小

延遲	緩衝區大小 = 24 KB	緩衝區大小 = 256 KB
10 毫秒	每秒 18.75 MB	每秒 100 MB
20 毫秒	每秒 9.38 MB	每秒 100 MB
50 毫秒	每秒 3.75 MB	每秒 40 MB
100 毫秒	每秒 1.88 MB	每秒 20 MB
200 毫秒	每秒 0.94 MB	每秒 10 MB

檢視與調整 TCP 緩衝區大小

您可以使用 `/usr/bin/netstat`(1M) 及 `/usr/sbin/ndd`(1M) 指令來檢視與調整 TCP 緩衝區大小。在調整時需要考慮的 TCP 參數包括：

- `tcp_max_buf`
- `tcp_cwnd_max`
- `tcp_xmit_hiwat`
- `tcp_recv_hiwat`

當您變更了其中一個參數，請使用 `shutdown` 指令重新啓動 Remote Mirror 軟體，以讓軟體使用新的緩衝區大小。然而，在您關閉及重新啓動伺服器時，TCP 緩衝區會返回預設大小。若要保留變更，請依照本節稍後所述來設定啓動 script 中的值。

調整網路以檢視 TCP 緩衝區與值

▼ 檢視所有 TCP 緩衝區

- 鍵入以下內容：

```
# /usr/sbin/ndd /dev/tcp ? | more
```

▼ 依緩衝區名稱檢視設定

- 此指令會顯示 1073741824 這個值。

```
# /usr/sbin/ndd /dev/tcp tcp_max_buf
1073741824
```

▼ 檢視虛擬連接埠的緩衝區大小

- 您可以使用 /usr/bin/netstat(1M) 指令來檢視特定網路虛擬連接埠的緩衝區大小。
  例如，若要檢視連接埠 121 的大小（預設的 Remote Mirror 連接埠）：

```
# netstat -na |grep "121 "
*.121 *.* 0 0 262144 0 LISTEN
192.168.112.2.1009 192.168.111.2.121 263536 0 263536 0 ESTABLISHED
192.168.112.2.121 192.168.111.2.1008 263536 0 263536 0 ESTABLISHED

# netstat -na |grep rdc
*.rdc *.* 0 0 262144 0 LISTEN
ip229.1009 ip230.rdc 263536 0 263536 0 ESTABLISHED
ip229.rdc ip230.ufsd 263536 0 263536 0 ESTABLISHED
```

本範例中顯示的 263536 這個值爲 256 KB 的緩衝區大小。主要與次要主機上的設定必須是相同的。

▼ 設定與檢驗啓動 Script 中的緩衝區大小

注意 – 在主要與次要主機上建立此 script。

1. 使用下列的值在文字編輯器中建立 script 檔：

```
#!/bin/sh
ndd -set /dev/tcp tcp_max_buf 16777216
ndd -set /dev/tcp tcp_cwnd_max 16777216

# increase DEFAULT tcp window size
ndd -set /dev/tcp tcp_xmit_hiwat 262144
ndd -set /dev/tcp tcp_recv_hiwat 262144
```

2. 將此檔案儲存爲 /etc/rc2.d/S68ndd 並離開檔案。
3. 設定 /etc/rc2.d/S68ndd 檔案的權限與所有權。

```
# /usr/bin/chmod 744 /etc/rc2.d/S68ndd
# /usr/bin/chown root /etc/rc2.d/S68ndd
```

4. 關機並重新啓動伺服器。

```
# /usr/sbin/shutdown -y g0 -i6
```

5. 檢驗之前所顯示的大小。

Remote Mirror 對 TCP/IP 連接埠的使用

主要與次要節點上的 Remote Mirror 軟體會在 `/etc/services` 中所指定的一個公認連接埠（連接埠 121）進行監聽。Remote Mirror 會寫入在虛擬連接埠（在主要站點上爲任意指定的位址；在次要站點上爲公認的位址）上從主要至次要站點的流量。狀態監視 "heartbeat"（活動訊號）會在不同的連線上進行（在次要主機上爲任意指定的位址；在主要主機上爲公認的位址）。Remote Mirror 通訊協定會在這些連線上使用 SUN RPC。

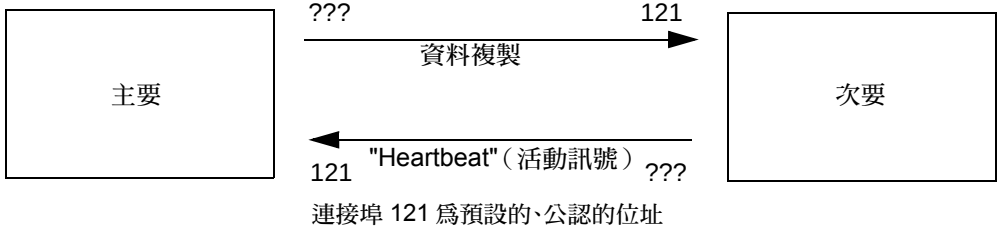

圖 1　Remote Mirror 對 TCP 連接埠位址的使用

預設的 TCP 監聽埠

連接埠 121 是 Remote Mirror `sndrd` 常駐程式所預設使用的 TCP 連接埠。若要變更連接埠號，請使用文字編輯器編輯 `/etc/services` 檔案。如需更多資訊，請參閱 *《Sun StorEdge Availability Suite 3.2 軟體安裝指南》*。

若您變更了連接埠號，您則必須在所有 Remote Mirror 主機的配置集中變更連接埠號（即：主要與次要主機；在一對多、多對一和多躍點配置中的所有主機）。此外，您必須關閉及重新啓動所有受到影響的主機，以使連接埠號變更能夠生效。

使用 Remote Mirror 與防火牆

由於 RPC 需要確認，您必須開啓*防火牆*才能允許來源或終點的封包欄位中有公認的連接埠位址。若此選項可以使用，也請務必配置防火牆以允許 RPC 流量。

在寫入複製流量中，確定前往次要站點的封包在終點欄位中將會有公認連接埠號，這些 RPC 的確認在來源欄位中將會有公認位址。

若是狀態監視，"heartbeat"（活動訊號）將會從次要站點產生並在終點欄位中顯示公認的位址，而確認則會在來源欄位中含有此位址。

Remote Mirror 軟體與 Point-in-Time Copy 軟體

爲了協助確保在一般作業中兩個站點間能有最高等級的資料整合性與系統效能，推薦您將 Remote Mirror 軟體與 Sun StorEdge Availability Suite Point-in-Time Copy 軟體一同搭配使用。

Point-in-Time Copy 可複製到實體的遠端位置，它提供了一致性的容體副本以作爲整體災後復原方案的一部分。這就是一般所謂的整批複製 (batch replication)。關於此方法的程序與優點，請參閱最佳的實施指南：*《Sun StorEdge Availability Suite Software - Improving Data Replication over a Highly Latent Link》*。

您可在從主要站點（作爲主機的主要容體之站點）啓動次要容體的同步化之前，建立 Remote Mirror 次要容體的 Point-in-Time Copy。您可以在開始重新同步化之前，啓動 Point-in-Time Copy 軟體在次要站點建立複製資料的 Point-in-Time Copy，以防雙重失敗。若在重新同步化期間有後續失敗發生，您可以使用 Point-in-Time Copy 作爲備用點，而當後續失敗問題解決之後，重新同步化就可以繼續進行。一旦次要站點已與主要站點完全同步化，您就可以停用 Point-in-Time Copy 軟體容體集，或作爲其他用途，例如：遠端備份、遠端資料分析、或次要站點所需的其他功能。

在啓動、複製或更新作業期間由內部執行的 Point-in-Time Copy 軟體 I/O 會轉變備用容體的內容，而不會新增 I/O 到 I/O 堆疊。當發生這種情形時，I/O 不會在 SV 層中截獲。若備份容體也是 Remote Mirror 容體，Remote Mirror 軟體也不會看見這些 I/O 作業。在這種情況下，I/O 所修改的資料將不會被複製到目標 Remote Mirror 容體。

若要允許此複製產生，您可將 Point-in-Time Copy 軟體配置成提供 Remote Mirror 軟體變更的點陣圖。若 Remote Mirror 軟體處於日誌模式，它會接受點陣圖、執行 Point-in-Time Copy 軟體點陣圖與其本身的容體點陣圖之 OR 比對，並將 Point-in-Time Copy 軟體變更新增至其本身要被複製到遠端節點的變更清單。若 Remote Mirror 軟體處於容體的複製模式，它會拒絕來自 Point-in-Time Copy 軟體的點陣圖。這將會導致啓動、複製或更新作業失敗。一旦重新啓動了 Remote Mirror 日誌，就會重新發出 Point-in-Time Copy 軟體作業。

注意 – 若要使 Point-in-Time Copy 軟體能在 Remote Mirror 容體上順利執行啓動、備份、更新或重設作業，您必須將 Remote Mirror 容體集置於日誌模式中。如果沒有這麼做，Point-in-Time Copy 作業就會失敗，而 Remote Mirror 軟體就會報告作業受到拒絕。

遠端複製配置

Remote Mirror 軟體能讓您建立一對多、多對一與多躍點容體集。

- 一對多複製能讓您從一個主要容體複製資料至在一部或多部主機上的多個次要容體。一個主要和每個次要站點容體爲一個單一容體集。例如，有了一個主要和三個次要主機容體，您就需要配置三個容體集：主要 A 和次要 B1、主要 A 和次要 B2，以及主要 A 和次要 B3。
- 多對一複製能讓您透過一個以上的網路連線複製橫跨兩部主機以上的容體。本軟體可支援將位在許多不同主機上的容體複製到單一主機的容體。這個專用術語和一對多配置專用術語不同，其中提到的一和多指的是容體。
- 多躍點複製表示將一個容體集的次要主機容體作爲另一容體集的主要主機容體。如果是一個主要主機容體 A 和一個次要主機容體 B，則次要主機容體 B 對次要主機容體 B1 會以主要主機容體 A1 的身份出現。

上述配置的任何一種組合也都會受到 Remote Mirror 軟體的支援。

詞彙

dsstat Sun StorEdge Availability Suite 中的一種工具，可用來從 Remote Mirror 與 Point-in-Time 快照產品顯示核心統計資料。

hwm 請參閱「高水印」。

lazy clear 登入 點陣圖追蹤寫入磁碟之模式，而非每一 I/O 事件的執行日誌。當遠端服務中斷或損壞時，此方法可追蹤尚未遠端複製之磁碟更新。爲每一來源容體識別不再符合其遠端容體集之區塊。軟體利用該日誌透過最佳的更新同步化而不是容體對容體的完整複製來重建 Remote Mirror。

TCP 緩衝區 TCP 緩衝區大小爲在等候確認前，傳輸控制通訊協定所允許傳輸的位元組數量。

一致性群組 一致性群組爲一組共用單一非同步佇列以維持寫入次序的遠端容體。

反向同步化 在復原演練期間使用的作業。日誌會追蹤演練期間用於次要系統之測試更新。當主要系統或容體復原時，便會以主要影像的區塊取代測試更新，並復原相符的遠端容體集。

主要或本端：主機或容體 主機應用程式主要相依系統或容體。例如，這是存取生產資料庫的地方。此資料要用軟體複製到次要主機。

正向重新同步化 請參閱「更新同步化」。

同步化 此過程將一份同樣的來源磁碟複製建立至目標磁碟，爲軟體鏡像的先決條件。

同步複製 由於 I/O 回應時間延遲會造成損壞性的影響，同步複製需限制爲短距離鏡像（數十公里）。

次要或遠端：主機或容體
: 主要主機的遠端對應主機，爲資料複製寫入和讀取之處。遠端複製在傳送過程中，主機並不會介入點伺服器之間。伺服器可能作爲某些容體的主要儲存體與其他容體的次要（遠端）儲存體。

自動同步化
: 若在主要主機上啓動了自動同步化選項，同步化常駐程式 (autosyncd) 會在系統重新啓動或發生連結失敗時試圖重新同步化容體集。

完全同步化
: 完全同步化會執行容體對容體的完整複製，爲同步化作業中最耗時的一種。在大多數的情況下，次要容體會從其來源的主要容體進行同步化。然而，故障的主要磁碟復原可能需要使用 Remote Mirror 作爲來源，進行反向的同步化。

更新同步化
: 更新同步化僅會複製日誌所辨識的磁碟區塊，降低復原 Remote Mirror 容體集的時間。

防火牆
: 一部作爲兩個網路間的介面並調節網路間的流量之電腦，以防內部網路受到來自外部網路的電子攻擊。

阻攔
: （非同步佇列）在阻攔模式中，若非同步佇列逐漸堆積，則所有未來的寫入作業都會延遲到佇列排出空間足以進行寫入作業爲止。阻攔模式（預設的非同步執行選項）會確認封包寫入至次要站點的次序。若設定了阻攔選項而使非同步佇列堆積，應用程式的回應時間可能會受到影響。

非同步佇列
: 用來儲存要複製到遠端站點的寫入之本端磁碟或記憶體區域。在寫入被放置到佇列中後，應用程式會對寫入進行確認。當網路功能允許時，稍後寫入就會被轉寄到遠端站點。

非同步複製
: 非同步複製在進行遠端影像更新前，會向來源主機確認主要 I/O 異動已完成。也就是，當本端寫入作業完成而遠端寫入作業已經排入佇列時，I/O 異動則已被主機所確認完成。延緩次要副本會把遠距傳輸延遲從 I/O 回應時間移除。

非阻攔
: （非同步佇列）在非阻攔模式中，若非同步佇列逐漸堆積，則 Remote Mirror 軟體會進入記錄日誌模式，且會刪除佇列內容。非阻攔模式不會確認封包寫入至次要站點的次序。但它可確保在非同步佇列堆積時，應用程式的回應時間仍不會受到影響。

容體集檔案
: 包含相關特定容體集資訊的文字檔案。此文字檔案和配置位置不同，配置位置包含的是 Remote Mirror 和 Point-in-Time Copy 軟體所使用之所有配置容體集的資訊。

配置位置
: Sun StorEdge Availability Suite 儲存本軟體使用的所有啓動容體之相關配置資訊的位置。

高水印
: 高水印爲使用的最大量非同步佇列。

複製 一旦容體集初始同步化之後，軟體會持續確認主要與次要容體包含相同的資料。複製是由使用者層級的應用寫入作業所驅動的；複製是個持續的過程。