CA Spectrum®

分散式 SpectroSERVER 管理員指南

9.4 版

本文件包含內嵌說明系統與文件 (以下稱爲「文件」)僅供您參考之用，且 CA 得隨時予以變更或撤銷。

未經 CA 事先書面同意，任何人不得對本「文件」之任何部份或全部內容進行影印、傳閱、再製、公開、修改或複製。 此「文件」爲 CA 之機密與專屬資訊，您不得予以洩漏或 用於任何其他用途，除非 (i) 您與 CA 已另立協議管理與本「文件」相關之 CA 軟體之使用；或 (ii) 與 CA 另立保密協議同意使用之用途。

即便上述，若您爲「文件」中所列軟體產品之授權使用者，則可列印或提供合理份數之「文件」複本，供您以及您的員工內部用於與該軟體相關之用途，但每份再製複本均須附上所有 CA 的版權聲明與說明。

列印或提供「文件」複本之權利僅限於軟體的相關授權有效期間。 如果該授權因任何原因而終止，您有責任向 CA 以書面證明該「文件」的所有複本與部份複本均已經交還 CA 或銷毀。

在相關法律許可的情況下，CA 係依「現狀」提供本文件且不做任何形式之保證，其包括但不限於任何針對商品適銷性、適用於特定目的或不侵權的暗示保證。 在任何情況下，CA 對於您或任何第三方由於使用本文件而引起的直接、間接損失或傷害，其包括但不限於利潤損失、投資損失、業務中斷、商譽損失或資料遺失，即使 CA 已被明確告知此類損失或損害的可能性，CA 均毋須負責。

「文件」中提及之任何軟體產品的使用均須遵守相關授權協議之規定，本聲明中任何條款均不得將其修改之。

此「文件」的製造商爲 CA。

僅授與「有限權利」。美國政府對其之使用、複製或公開皆受 FAR 條款 12.212，52.227-14 與 52.227-19(c)(1) - (2) 與 DFARS 條款 252.227-7014(b)(3) 中所設之相關條款或其後續條約之限制。

Copyright © 2014 CA. All rights reserved. 本文提及的所有商標、商品名稱、服務標章和公司標誌均爲相關公司所有。

CA Technologies 產品參考資料

本指南涉及 CA Spectrum®。

連絡技術支援

如需線上技術協助及完整的地址清單、主要服務時間以及電話號碼，請洽「技術支援」，網址為：http://www.ca.com/worldwide。

目錄

第 1 章： 介紹分散式 SpectroSERVER 9

關於分散式 SpectroSERVER 9
範圍 10
範圍對應 10
模型化類別 11
使用者模型 11
SpectroSERVER (.vnmrc) 資源 12
一般 SpectroSERVER (.vnmrc) 資源 13
事件封存檔 (.vnmrc) 資源 20
工作執行緒 (.vnmrc) 資源 21
容錯警報服務 (.vnmrc) 資源 22
如何封裝 CA Spectrum 公用程式並將其移至另一部電腦 24
封裝 CA Spectrum 公用程式 25
將 CA Spectrum 公用程式移至另一部電腦 26

第 2 章： 設定分散式 SpectroSERVER 環境 31

名稱解析需求 31
CA Spectrum 和多重介面 31
SpectroSERVER 間的通訊 32
連接埠衝突解決方法 32
DSS 環境需求 34
位置伺服器 35
位置伺服器如何互動 36
主要位置伺服器連線 37
指定新的主要位置伺服器 37
範圍對應完整性保護 38
範圍控制代碼 39
指派範圍控制代碼 39
變更範圍控制代碼 40
程序精靈 (processd) 40
Windows 和 Solaris 環境中的 Processd 差異 41
變更 Processd 中的 Windows 密碼 42
安裝工單檔案 43
停止再重新啟動 Processd 48

Userconf 程序在安裝期間如何運作 49
使用者登入時 Userconf 如何運作 50
網路分割 50
分散式環境中的重複模型 51
無法連絡主目錄範圍錯誤 51
主機資源配置檔案 (.hostrc) 52
DSS 環境中的時區 52
SpectroSERVER 關機 52
在 Solaris 環境中關閉 SpectroSERVER 53
在 Windows 環境中關閉 SpectroSERVER 53
設定範圍對應項目逾時 53
現有 DSS 設定中範圍對應的問題 54
清除舊範圍的相關問題 56

第 3 章： 分散式 SpectroSERVER 環境中的跨防火牆通訊 57

跨防火牆通訊 57
SpectroSERVER 和 OneClick Web 伺服器的跨防火牆通訊 58
OneClick 預設連接埠和防火牆 59
HTTP 接聽連接埠 59
CORBA 接聽連接埠 59
遠端 SpectroSERVER 和防火牆 59
主要和次要 SpectroSERVER 的跨防火牆通訊 61
NAT 防火牆環境的 CA Spectrum 配置檔案 61
預設連接埠配置 62
變更 SpectroSERVER 連接埠號碼 62
變更 Archive Manager 連接埠號碼和通訊端號碼 63
變更位置伺服器連接埠號碼和通訊端號碼 63
變更 Visibroker 命名服務連接埠號碼 64
遠端複製程序精靈 (rcpd) 連接埠號碼配置 64
CLI 精靈 (vnmshd) 連接埠號碼配置 64

第 4 章： 容錯 67

關於 SpectroSERVER 容錯 67
容錯環境的 SpectroSERVER 優先順序 68
SpectroSERVER 資料同步 68
支援容錯 Archive Manager 69
Archive Manager 資料同步 70
次要 SpectroSERVER 未重新啟動時產生警報 70

次要 SpectroSERVER 整備層級 71
SpectroSERVER 警報同步 72
從主要往次要 SpectroSERVER 同步 73
從次要往主要 SpectroSERVER 同步 74
建立容錯 76
驗證容錯配置 78
測試容錯 79
容錯復原 80
變更主要和次要 SpectroSERVER 的主機名稱 80
監控主要和次要 SpectroSERVER 間的轉換 82
如何監控次要 SpectroSERVER 狀態 83

第 5 章： 使用 Trap Director 85

Trap Director 85
設陷與記憶體使用量 86
設陷資料流量整合 86
Trap Director 如何更新位址快取 87
容錯設定中的 Trap Director 88
設陷風暴設定 88
啓用和停用 Trap Director 88
定義快取記錄保留期限 89

第 1 章： 介紹分散式 SpectroSERVER

本節包含以下主題：

關於分散式 SpectroSERVER (位於 p. 9)
範圍 (位於 p. 10)
範圍對應 (位於 p. 10)
模型化類別 (位於 p. 11)
使用者模型 (位於 p. 11)
SpectroSERVER (.vnmrc) 資源 (位於 p. 12)
如何封裝 CA Spectrum 公用程式並將其移至另一部電腦 (位於 p. 24)

關於分散式 SpectroSERVER

分散式 SpectroSERVER (DSS) 是強大的模型化功能，可以將管理作業分散至大規模基礎架構。 基礎架構可根據地理位置分類，也可橫跨單一實體位置中的數個伺服器。 DSS 可以改善 CA Spectrum 在管理運算基礎架構時的效能。 效能改善來自於分散管理流量的網路負載，以及委派工作給遠端工作站。

使用 DSS，您可以建立基礎架構的統一表示法，其中包含多個範圍，每個都有本機 SpectroSERVER。在 DSS 環境中，SpectroSERVER 用戶端 (例如 OneClick 主控台) 可以同時存取多個 SpectroSERVER 的資訊。

DSS 包含下列功能：

分散式能見度

OneClick 主控台會顯示所有已部署 SpectroSERVER 的資訊。 會為每個 SpectroSERVER 顯示摘要警報計數。 網路管理員可以迅速找到出有問題的區域。

本地化的輪詢流量

輪詢網路的管理工作站在地理位置上更接近其使用 DSS 管理的裝置。 此設定可降低廣域連結的流量，避免本機網路壅塞。 多個小型 SpectroSERVER 產生的流量，少於從遙遠距離輪詢裝置的單一 SpectroSERVER。

擴充性

擴充網路時，使用低端工作站作爲額外的專用 SpectroSERVER 伺服器，通常會比較便宜。 在大型非-分散式環境中，這樣的解決方案會比將一個工作站升級以便容納一個更爲合適。

容錯

DSS 支援 SpectroSERVER 間的容錯。 可以提供次要 SpectroSERVER 作爲容錯備份或主要 SpectroSERVER 的待命伺服器。 如果執行主要 SpectroSERVER 的工作站失敗，網路管理仍可繼續進行。

只需將特定範圍的資料庫重新載入至次要 SpectroSERVER，您就能預防失敗。 當發生失敗而停用主要 SpectroSERVER 時，次要 SpectroSERVER 會自動接管。 所有 CA Spectrum 應用程式都會自動使用次要 SpectroSERVER。

更多資訊：

建立容錯 (位於 p. 76)

範圍

單一 SpectroSERVER 管理的網路網域稱爲「*範圍*」。 範圍包括特定 SpectroSERVER 所屬的模型、關聯、屬性值、警報、事件和統計資料。 網路中的每個範圍都是獨特的。 唯一範圍控制代碼 (ID) 可識別每個範圍。

在 OneClick 中，範圍圖示代表每個範圍。 範圍圖示代表 SpectroSERVER 資料庫。 透過按兩下區域和功能表選項，範圍圖示可讓您存取遠端網路模型。 圖示也提供這些遠端資料庫中模型化之裝置的警報資訊彙總。

附註：「*本機*」和「*遠端*」等詞是用來界定特定 SpectroSERVER 觀點下的範圍。

範圍對應

CA Spectrum 會自動維護包含特定 DSS 環境之所有範圍的「對應」。 SpectroSERVER 拓撲中發生的變更會啓動內部搜索機制，更新該環境中所有其他 SpectroSERVER 的範圍對應。 例如，搜索會偵測到網路中有新增或移除 SpectroSERVER。

範圍模型是一種容器模型。 範圍模型可讓您透過 CA Spectrum 使用者介面，連線至範圍對應中的其他 SpectroSERVER。 您可在三個檢視階層的任何層級建立範圍模型：拓撲、位置或組織。

模型化類別

模型化類別是安裝在特定 SpectroSERVER 上的一組範本 (例如模型類型或關係)。 這些範本將用於建立模型。 透過這些範本建立的模型組成該 SpectroSERVER 的範圍。

附註：OneClick 會在啓動時，向預設範圍查詢可用模型類型的清單。

當您使用 DSS 模型化多個範圍時，每個範圍必須包含相同的模型化類別。 一個範圍之模型化類別中的所有模型類型，也必須存在於其連線之其他每個範圍的模型化類別中。 如果您在一個 SpectroSERVER 上安裝新的管理模組，則也必須在範圍對應的其他每個 SpectroSERVER 上安裝同一個管理模組。

使用者模型

使用者模型包含個別使用者權限及其他使用者資料的詳細資訊。 此項資訊儲存在範圍中。 第一次安裝 CA Spectrum 時，由預設使用者模型代表您在 CA Spectrum 安裝期間於 [安裝擁有者] 欄位中指定的使用者。

附註：如需詳細資訊，請參閱《*管理指南*》。

在 DSS 環境中，所有範圍內都必須有相同的使用者模型，以讓使用者連線至範圍對應中的遠端 SpectroSERVER，並讓範圍彼此通訊。 新增或修改使用者模型，會造成與範圍相關聯的 SpectroSERVER 查詢範圍對應中的所有其他 SpectroSERVER。 查詢會檢查其他範圍中的使用者模型。 如果在其他範圍中找到使用者模型，這些範圍的使用者模型就會自動更新以反映初始使用者模型中進行的任何變更。

附註：刪除使用者模型僅會刪除該範圍中的使用者模型。 您必須手動刪除範圍對應中其他 SpectroSERVER 上的重複使用者模型。

SpectroSERVER (.vnmrc) 資源

SpectroSERVER 資源定義在 <*$SPECROOT*>/SPECTRUM/SS/.vnmrc 檔案中。 這些資源檔案的許多預設值內建在程式碼中，且保持空白。 不過，這些資源中有指定值的時候，就不會使用程式碼編碼的預設值。

.vnmrc (虛擬網路機器執行階段配置) 檔案中的資源會定義 SpectroSERVER 的路徑名稱和預設設定。 CA Spectrum 系統軟體包含 SpectroSERVER 資源的執行階段配置檔以及預設設定。

您可以修改資源，進行下列變更：

- 定義一般 SpectroSERVER 資源和目錄路徑
- 調整事件封存
- 指定名稱服務變數
- 調整執行緒配置
- 控制容錯警報同步

這些資源項目都以 "resource = resource_value" 的格式列出。 對於 .vnmrc 檔案中的許多資源，資源值是空白的。

如果特定資源沒有顯示值，CA Spectrum 就使用預設值。 .vnmrc 檔案中的資源會在 SpectroSERVER 啓動時生效。 如果您在 SpectroSERVER 執行時變更此檔案，這些變更會在 SpectroSERVER 重新啓動後生效。

範例：定義 .vnmrc 資源

此範例設定 [事件日誌] 資料庫中的記錄數上限：

```
max_event_records=5000
```

max_event_records

此爲資源名稱。

5000

此爲資源值。

一般 SpectroSERVER (.vnmrc) 資源

一般 SpectroSERVER (.vnmrc) 資源控制許多 SpectroSERVER 函數。下列清單說明一般 SpectroSERVER (.vnmrc) 資源：

comm_port

指定用戶端使用者介面用來與 SpectroSERVER 通訊的 TCP 連接埠。 連接埠號碼於安裝期間定義。

這個命令具有下列格式：

`comm_port=0xBEEF`

0xBEEF

此爲預設的 TCP 連線連接埠通訊端。 只要是大於指派給 /netinet/in.h 檔案中 IPPORT_USERRESERVED 參數之連接埠號碼的任何有效 TCP 連接埠，都可以作爲此通訊端。 但連接埠必須小於 65535 (0xFFFF)。

expiration_date

指定 CA Spectrum 授權到期的日期。 您在安裝期間輸入的 CA Spectrum 金鑰即包含到期日期。

這個命令具有下列格式：

`expiration_date=mm/dd/yyyy`

密碼

會自動填入於安裝期間輸入的 CA Spectrum 金鑰。 安裝後請勿變更此資源。

這個命令具有下列格式：

`password=password`

密碼由 CA 提供。

snmp_comm_port

指定可透過 SpectroSERVER 傳送 SNMP 要求的連接埠。

這個命令具有下列格式：

`snmp_comm_port=port#`

port#

設定爲不帶正負號的 16 位元整數，範圍爲 0x400 (1,024) 到 0xFFFF (65,535)。 有些 SNMP 代理程式實作 (例如 IBM Mainframe MVS 系統) 會將該連接埠視爲帶正負號的數字。 在這種情況下，請將此資源設爲介於 0x400 (1,024) 和 0x7FFF (32,768) 之間的值。

預設值：0xFFFF

snmp_trap_port_enabled

將 SpectroSERVER 繫結至 SNMP 設陷連接埠並接聽設陷。 設定爲 False 時，SpectroSERVER 不會繫結至 SNMP 設陷連接埠。

這個命令具有下列格式：

`snmp_trap_port_enabled=TRUE`

預設值：True

vnm_file_path

指定包含 SpectroSERVER 外部檔案 (例如資料庫檔案) 的根子目錄。

這個命令具有下列格式：

`vnm_file_path=<directory>/spectrum/SS/CsVendor`

/spectrum/SS/CsVendor

此爲根子目錄的檔案路徑

tcp_buffer_size

可讓您更改緩衝區大小以便符合您的 LAN 流量。 較大的緩衝區大小可以提高輸送量。 此資源接受數值，代表 TCP 緩衝區的位元組數目。 這個命令具有下列格式：

`tcp_buffer_size=<blank>`

預設值：空白

限制：8192 至 65536 位元組

附註：低於 8 KB 的值會進位爲 8 KB。 高於 64 KB 的值則設爲 64 KB。

resource_file_path

此為 VNM 資源檔案 (例如 Ether Map) 的檔案路徑。

這個命令具有下列格式：

`resource_file_path=./CsResource`

./CsResource

(選擇性) 此為 VNM 資源檔案的檔案路徑。 此參數可留空。

wait_active

決定伺服器是要在所有模型載入後隨即接受連線，或是要等到所有模型都在使用中後才接受連線。 若設為 *Yes*，則會出現 [控制台] 訊息，顯示 SpectroSERVER 啓動期間啓用之模型的執行中百分比。

這個命令具有下列格式：

`wait_active=no`

預設值：no。

max_bind_retry_count

指定接聽通訊端繫結重試次數的上限。

這個命令具有下列格式：

`max_bind_retry_count=50`

預設值：50。

bind_retry_interval

指定每次重試接聽通訊端繫結之間的延遲秒數。

這個命令具有下列格式：

`bind_retry_interval=30`

預設值：30 秒。

min_client_version

此為 SpectroSERVER 可接受用於用戶端連線的最小參數區塊版本。

這個命令具有下列格式：

`min_client_version=0`

預設值：0 個用戶端連線下限。

max_connections

此為 SpectroSERVER 可接受的連線數目上限。 連線可以來自用戶端或其他伺服器，例如另一個 SpectroSERVER、Archive Manager 或位置伺服器 (主要和本機)。

這個命令具有下列格式：

```
max_connections=50 gensv000316 docs002958
```

預設值：50 個連線

超過用戶端連線數目上限時，連線會失敗。 提高 SpectroSERVER 連線數目上限時，有時候必須在您的工作站上提高開啓檔案描述元的數目，以便改進效能。

handshake_timeout

設定連線交握期間，交換初始 ID 資訊的秒數。

這個命令具有下列格式：

```
handshake_timeout=40
```

預設值：40 秒

vnm_message_timeout

vnm_message_timeout 資源設定 VNM 之間所傳送訊息的逾時秒數。

這個命令具有下列格式：

```
vnm_message_timeout=180
```

預設值：180 秒

vnm_close_timeout

vnm_close_timeout 資源設定關閉 VNM 之間連線的逾時秒數。

這個命令具有下列格式：

```
vnm_close_timeout=180
```

預設值：180 秒

connect_time_limit

設定 VNM 連線的逾時毫秒數。

這個命令具有下列格式：

```
connect_time_limit=5000
```

預設值：5000 毫秒

rcpd_comm_port

指定遠端複製程序精靈 (rcpd) 的接聽連接埠。 此資源將用於容錯資料庫備份。

這個命令具有下列格式：

`rcpd_comm_port=0xCAFE`

預設值：0xCAFE

procd_comm_port

指定 VNM 用來連線至程序精靈的連接埠。

這個命令具有下列格式：

`procd_comm_port=0xFEEB`

預設值：0xFEEB

snmp_trap_port

指定 VNM 用來接收設陷的連接埠。

這個命令具有下列格式：

`snmp_trap_port=162`

預設值：162

enable_traps_for_pingables

指定 CA Spectrum 是否能接收可 Ping 模型的 SNMP 設陷。 .vnmrc 檔案預設不含此資源。 請使用下列語法，將此資源手動新增至檔案：

`enable_traps_for_pingables = TRUE`

預設值：TRUE

device_limit

指定當套用限制時，可模型化的裝置數目上限。 此值衍生自安裝期間輸入的 CA Spectrum 金鑰。 此值無法編輯。 不是簡單主機、可 Ping 裝置或節點之 Proxy 代理程式的受管理節點，就是模型化裝置。 如果未對您的 CA Spectrum 安裝套用裝置限制，則此資源不會有關聯值。 (另請參閱 max_device。)

這個命令具有下列格式：

```
device_limit=<# of device models>
```

# of device models

此爲衍生自 IPAddress (0x000102bc) 模型類型和裝置 (0x0001004b) 模型類型的任何模型。

此外，也會計入衍生自 DIRIG_NODE 應用程式和下列寬頻模型的模型：MOTCBLMODEM、SAEXPLORER 及 DOCSISCM。

此規則的例外狀況是：

GEN_HOST、PINGABLE、PATROL_PET 和 WINDOWS_PC。

max_device

套用至 device_limit 資源生效中的 CA Spectrum 產品。 依預設，當模型數目達到指定之裝置限制的 80% 時，CA Spectrum 會產生黃色警報。 當模型數目達到指定之裝置限制的 100% 時，CA Spectrum 會產生紅色警報。 此資源可讓您設定低於實際裝置限制的值，以便提供早期警告。

這個命令具有下列格式：

```
max_devices=<# of device models>
```

# of device models

是指 IP 位址衍生自裝置模型類型，「以及」其指定數目低於 device_limit 資源值的模型。 預設值爲空白。

disable_redundancy_when_using_loopback

以回送模型化裝置時，停用備援 (0x11d2c)。 此參數僅適用於 VNM 模型的 use_loopback (0x12bb) 屬性設爲 True 時。

這個命令具有下列格式：

```
disable_redundancy_when_using_loopback=False
```

依預設，此資源不會顯示在 .vnmrc 檔案中，因此會自動評估爲 False。

附註：如需詳細資訊，請參閱《*模型化和管理 IT 基礎架構管理指南*》。

persistent_alarms_active

當 SpectroSERVER 關機時，阻止 CA Spectrum 保留警報的詳細資訊。

這個命令具有下列格式：

```
persistent_alarms_active=<False>
```

依預設，此資源不會顯示在 .vnmrc 檔案中。 CA Spectrum 會自動在 SpectroSERVER 關機和重新啟動時保留警報相關資訊 (例如疑難排解員指派或狀態)。 關機前出現的警報會保留下來。 這些警報被視爲「持續性」警報。 新增 persistent_alarms_active=False 至 .vnmrc 檔案，會阻止 CA Spectrum 在 SpectroSERVER 關機時保留警報-的詳細資訊。

附註： 強烈建議您不要新增 persistent_alarms_active 資源，除非您整合了會保留警報相關資訊的協力廠商應用程式。

新增此資源具有以下效果：

- 會遺失警報狀態新增項目 (僅記錄爲過去事件)
- 現有的 SpectroSERVER 關機時警報會在啓動時重新產生，並加上新的時間戳記。
- 重新產生的警報會轉寄至警報通知工具。

unsupported_attr_poll_interval

指定 CA Spectrum 針對傳回 "noSuchName" 錯誤的外部屬性進行輪詢所等候的時間量。 如果未在 .vnmrc 檔案中指定此參數，將使用預設值 (12 個小時)。

這個命令具有下列格式：

```
unsupported_attr_poll_interval=43200
```

預設值：43200 秒 (12 個小時)

更多資訊：

遠端複製程序精靈 (rcpd) 連接埠號碼配置 (位於 p. 64)

事件封存檔 (.vnmrc) 資源

事件封存檔 (.vnmrc) 資源控制 SpectroSERVER 傳送事件至 Archive Manager 以供記錄的程序。 下列清單說明事件封存檔 (.vnmrc) 資源：

max_event_records

指定 [事件日誌] 資料庫中可儲存的記錄數上限。

這個命令具有下列格式：

`max_event_records=# of Records`

# of Record

此為 [事件日誌] 記錄的數目。最小值等於 event_record_increment 值。系統儲存容量會限制最大值。

預設值：20,000

event_record_increment

指定當記錄數超過 max_event_records 值時，要從 [事件日誌] 資料庫刪除的記錄數。

這個命令具有下列格式：

`event_record_increment=# of Records`

# of Records

此為要從資料庫刪除之 [事件日誌] 記錄的數目。 最小值為 100。

event_batch_max_size

設定每批次事件數目上限。 如果批次滿了，事件批次就會立即傳送至 Archive Manager。

這個命令具有下列格式：

`event_batch_max_size=1000`

預設值：1000 (事件數目上限)

event_batch_timeout

決定將事件批次傳送至 Archive Manager 的使用秒數。

這個命令具有下列格式：

`event_batch_timeout=1`

預設值：*1* 秒

log_user_events

控制是否要爲每個使用者啓動的模型屬性值寫入產生事件。 True 值會讓 VNM 產生事件。

這個命令具有下列格式：

log_user_events=*False*

預設值：False

use_log_queue

當事件產生時，將事件放入個別佇列。 這些事件會以不同的執行緒傳送給 Archive Manager。 若設爲 "TRUE"，事件會以其產生時的相同執行緒傳送給 Archive Manager。 這種情況會延遲警報的建立。

這個命令具有下列格式：

use_log_queue=<*blank*>

預設值：<*blank*>

工作執行緒 (.vnmrc) 資源

SpectroSERVER 是多執行緒程序。 在正常操作下，每個子系統會配置多個工作執行緒。

SpectroSERVER 會維護一個工作執行緒集區，當中的工作執行緒可以不斷重複使用。 在活動增加的期間，子系統可以使用集區中高達其個別限制的執行緒。 共用的工作執行緒集區用盡後，會建立新的工作執行緒。 此集區會成長以容納增加的活動。

子系統不再需要的工作執行緒會返回共用集區，以供稍後使用。 從集區取出的執行緒將用於服務後續需求，如果集區是空的，則會配置新執行緒。 一段時間未使用的執行緒會從集區中移除，且其資源會返回系統。 此程序稱爲過時。

下列清單說明工作執行緒 (.vnmrc) 資源：

max_total_work_threads

指定可爲所有 SpectroSERVER 子系統配置的工作執行緒數目上限。 每個工作執行緒需消耗處理資源，且需要極大的記憶體區塊。 請根據系統容量 (記憶體大小和速度) 設定其值。

當此資源的值過高時，SpectroSERVER 會定期用盡記憶體。 當此資源的值過低時，SpectroSERVER 作業會變得遲緩。

如果此資源的值才剛設定或變更，CA Spectrum 會在 SpectroSERVER 重新啓動時讀取新值。 讀取該值後，就會用於更新 VNM 模型的 WorkThreadsMaxAvail 屬性值。 更新 WorkThreadsMaxAvail 的值後，會從 .vnmrc 檔案移除 max_total_work_threads 的值。

此命令的格式如下：

`max_total_work_threads=# of threads`

VNM WorkThreadsMaxAvail 屬性的預設值爲 500。

work_thread_age

子系統不再需要的工作執行緒會返回工作執行緒集區。 此資源指定工作執行緒在未被使用的情形下，可放在集區中多久時間 (單位爲秒)。

請將此資源設爲與子系統活動相容的值。 建立工作執行緒需要相當大的處理負荷。 若將此資源的值設得過低，會因爲要因應工作執行緒需求而太常建立新執行緒，造成系統資源的負擔。 將此值設得過高，則表示資源會維持不必要的配置。

此命令的格式如下：

`work_thread_age =seconds`

預設值：60 秒

容錯警報服務 (.vnmrc) 資源

在容錯環境中，警報必須在主要和次要 SpectroSERVER 之間予以同步。 伺服器需連線以便交換警報資訊。如果初始連線嘗試失敗，稍後會繼續嘗試。「容錯警報服務」使用下列設定來控制警報同步：

ftasv_enabled

啓用警報同步。請將此命令同時加入主要和次要伺服器的 .vnmrc 檔案。

這個命令具有下列格式：

`ftasv_enabled=true`

預設值：True

重要！ 您不能將一部伺服器的 ftasv_enabled 設爲 True，卻將另一部伺服器的 ftasv_enabled 設爲 False。 兩部伺服器需採用相同值。

ftasv_max_conn_retry_count

指定在連線嘗試失敗後，主要伺服器嘗試連線至次要伺服器 (以便進行同步) 的次數。 此參數對於主要 SpectroSERVER 的 .vnmrc 檔案為必要參數。

附註：主要伺服器嘗試連線至次要伺服器時，會採用此參數。 次要伺服器嘗試連線至主要伺服器時，不會採用此參數。

這個命令具有下列格式：

```
ftasv_max_conn_retry_count=# of retries
```

預設值：四次重試 (總共嘗試五次與次要伺服器進行同步)。

ftasv_conn_retry_interval

指定主要伺服器與次要伺服器之間同步嘗試的間隔秒數。 請在主要 SpectroSERVER 的 .vnmrc 檔案中使用此參數。

附註：主要伺服器嘗試連線至次要伺服器時，會採用此參數。 次要伺服器嘗試連線至主要伺服器時，不會採用此參數。

這個命令具有下列格式：

```
ftasv_conn_retry_interval=# of seconds
```

預設值：30 秒

ftasv_debug

啓用警報同步活動的偵錯輸出。 請將此命令同時加入主要和次要伺服器的 .vnmrc 檔案。 偵錯輸出會寫入每部伺服器的 VNM.OUT 檔案。 每個訊息的開頭都是「容錯警報服務」(Fault Tolerant Alarm Service)。

這個命令具有下列格式：

```
ftasv_debug=true
```

預設值：False

重要！ 如果當主要伺服器嘗試連線至未執行的次要 SpectroSERVER 時，主要伺服器會耗盡其同步嘗試次數。因此，啓動主要 SpectroSERVER 會發生延遲。 ftasv_max_conn_retry_count 和 ftasv_conn_retry_interval 使用預設設定時，延遲是兩分鐘 (四次重試，間隔 30 秒)。 這種延遲是無法避免的，因為它發生在模型啓動之前。 解決辦法是啓動主要伺服器時，驗證次要 SpectroSERVER 正在執行中。 您也可以減少重試次數或間隔大小，以降低可能的延遲。

更多資訊：

容錯 (位於 p. 67)
SpectroSERVER 警報同步 (位於 p. 72)

如何封裝 CA Spectrum 公用程式並將其移至另一部電腦

packtool.pl 指令檔可以封裝 CA Spectrum 命令列介面 (CLI)、CA Spectrum SS Logger、AlarmNotifier、modelinggateway 工具和 sbgwimport 工具，供您傳輸至另一部電腦。 在移動檔案時，指令檔會保留這些公用程式及其支援檔案的相對目錄結構。

附註：兩部電腦必須執行相同的作業系統。

執行 packtool.pl 指令檔前，請先考量下列要求：

- 您必須是 Solaris 和 Linux 平台的根使用者，或是 Windows 平台上具有系統管理權限的使用者，才能執行 packtool.pl 指令檔。
- 傳輸 CA Spectrum 公用程式至僅安裝了 OneClick 伺服器的電腦上時，適用下列要求：
  - 每部電腦上的 OneClick 伺服器版本和修補程式等級必須符合。 每次安裝修補程式後，請再次執行 packtool.pl 指令檔，然後再次傳輸公用程式。
  - 傳輸公用程式之電腦上的安裝使用者，必須符合封裝檔案之電腦上的安裝使用者。
  - 如果目標電腦上有安裝偵錯修補程式或 PTF，請在傳輸檔案後重新安裝偵錯修補程式或 PTF。
- 停止封裝公用程式之電腦上的所有 CA Spectrum 程序。
- 擷取 tools_bundle 檔案前，停止傳輸公用程式之電腦上的所有 CA Spectrum 程序。
- 若要擷取 tools_bundle 檔案，您必須是 Solaris 和 Linux 平台的根使用者，或是 Windows 上具有系統管理權限的使用者。

重要！ 若要執行 CA Spectrum 公用程式，您必須以有效 CA Spectrum 使用者的身分登入目標電腦，否則會發生錯誤。 例如，如果您以 "root" 登入 Linux 工作站，但 CA Spectrum 中沒有相符的使用者模型，則會收到錯誤，指出 "NO_USER"。

若要封裝 CA Spectrum 公用程式並將其傳輸至另一部電腦，請執行下列步驟：

1. 使用 packtool.pl 指令檔封裝 SPECTRUM 公用程式。 (位於 p. 25)
2. 在另一部電腦上擷取 tools_bundle 檔案 (位於 p. 26)。

封裝 CA Spectrum 公用程式

在移動 CLI、SSLogger、AlarmNotifier、modelinggateway 工具和 sbgwimport 工具至另一部電腦前，請先執行指令檔打包這些檔案。

請依循下列步驟:

1. 確認環境變數 *<$SPECROOT>* 是設為您封裝公用程式之電腦上的 CA Spectrum 安裝目錄路徑。
2. 將 CYGWIN32 環境變數設為安裝 cygwin 所在的目錄。
3. 執行 packtool.pl 指令檔，此指令檔會封裝公用程式及其支援檔案。 您可在 *<$SPECROOT>*/SS-Tools 目錄中找到 packtool.pl 指令檔。

   若要從 Bash shell 或其他 UNIX shell 執行指令檔，請輸入下列命令：

   ```
   <$SPECROOT>/SS-Tools/packtool.pl [-no_notifier | -no_event_alarms]
   [-f file_name]
   ```

   <$SPECROOT>

   此為 CA Spectrum 安裝於您 SpectroSERVER 上的目錄結構。

   -no_notifier

   指定您不要封裝 AlarmNotifier。

   -no_event_alarms

   指定您不要封裝 AlarmNotifier EvFormat 或 PCause 檔案。

   -f *file_name*

   指定可執行檔的名稱。

   會建立名為 linux_tools_bundle (Linux)、solaris_tools_bundle (Solaris) 或 nt_tools_bundle.exe (Windows) 的可執行檔，其中包含 CA Spectrum 公用程式及其支援檔案。

將 CA Spectrum 公用程式移至另一部電腦

您可將 CA Spectrum 公用程式移至僅執行 OneClick 伺服器的電腦。

請依循下列步驟:

1. 封裝 CA Spectrum 公用程式 (位於 p. 25)。
2. 在您要擷取 CA Spectrum 公用程式的電腦上，將目錄變更為 <*$SPECROOT*>。
3. 使用二進位模式，將名為 linux_tools_bundle (Linux)、solaris_tools_bundle (Solaris) 或 nt_tools_bundle.exe (Windows) 的可執行檔以 FTP 傳送至目前目錄。
4. 從 DOS、Bash 或其他 UNIX shell 擷取 tools_bundle 檔案。

   CA Spectrum 公用程式及其支援檔案會解除封裝至適當的目錄結構。
5. 如果您傳輸的是 AlarmNotifier 和事件及 pcause 檔案，請驗證 <*$SPECROOT*>/SG-Support/CsResource/preferences/*.prf 中的路徑有正確的事件及 pcause 檔案位置。 也請驗證 ui= 和 lhandle= 選項。
6. 將 <*$SPECPATH*> 環境變數設定為您擷取 tools_bundle 檔案的目錄路徑：
   - 在 Windows 上，以下列變數格式使用值建立系統環境變數 <*$SPECPATH*>：

     `driveletter:\PATH_TO_SPECTRUM`
   - 在 Solaris 上，新增下面這行至 /opt/SPECTRUM/spectrum60.env 檔案：

     `SPECPATH=PATH_TO_SPECTRUM`
   - 在 Linux 上，新增下面這行至 /opt/SPECTRUM/spectrum80.env 檔案：

     `SPECPATH=PATH_TO_SPECTRUM`
7. 在 Solaris 和 Linux 平台上，於 /opt/SPECTRUM 目錄的 spectrum*.env 檔案中，將 BES_LIC_DIR 定義為 PATH_TO_SPECTRUM/bin/VBNS/license。
8. 在 Linux 平台上，將 /usr/bin/perl 複製至 <*$SPECROOT*>/bin。
9. 驗證 .hostrc 檔案包含本機主機名稱和傳輸公用程式的來源電腦名稱。

10. 驗證主要位置伺服器上的 .hostrc 檔案包含用於擷取 tools_bundle 檔案之電腦的主機名稱。
11. 驗證 .LocalRegFile 包含正確的主要位置伺服器。
12. 在 shell 中設定 CLISESSID，以使用 CLI 公用程式：

    Windows：

    ```
    set CLISESSID=<NUMBER>
    ```

    Solaris 和 Linux：

    ```
    export CLISESSID=<NUMBER>
    ```

    NUMBER

    此為 shell 的唯一號碼。
13. 確認 Notifier/.alarmrc 具有正確的 SetScript、ClearScript 和 UpdateScript 路徑。 這些指令檔位於您擷取 tools_bundle 檔案的 Notifier 目錄中。
14. 重新啟動 processd。

    您現在可以在此電腦上執行公用程式了。

您也可以將 CA Spectrum 公用程式移至未執行 CA Spectrum 的電腦。

請依循下列步驟:

1. 封裝 CA Spectrum 公用程式。
2. 在您要擷取 CA Spectrum 公用程式的電腦上，建立目錄以解除封裝 CA Spectrum 公用程式及其支援檔案。 例如，建立下列目錄：

   Windows：

   ```
   c:\win32app\spectrum
   ```

   Solaris 和 Linux：

   ```
   /usr/spectrum
   ```

   附註：將作用中目錄 (chdir) 變更為這些目錄。
3. 使用二進位模式，透過 FTP 將可執行檔傳送至目前目錄。 檔案具有下列其中一個名稱：linux_tools_bundle (Linux)、solaris_tools_bundle (Solaris) 或 nt_tools_bundle.exe (Windows)。
4. 從 DOS、Bash 或其他 UNIX shell 執行 tools_bundle 檔案。

   CA Spectrum 公用程式及其支援檔案會解除封裝至適當的目錄結構。

5. 驗證 PATH 變數如下所示：

   Windows：

   確認 <*$SPECROOT*>/lib 位於 PATH 變數中。

   Solaris 和 Linux：

   - 建立 /opt/SPECTRUM 目錄。
   - 建立從 <*$SPECROOT*>/lib 連往 /opt/SPECTRUM/lib 的連結。
   - 建立從 <*$SPECROOT*>/bin 連往 /opt/SPECTRUM/bin 的連結。

6. 將 <*$SPECROOT*> 和 <*$SPECPATH*> 環境變數設定為您擷取 tools_bundle 檔案的目錄路徑：

   - 在 Windows 上，以下列變數格式使用值建立系統環境變數 <*$SPECROOT*> 和 <*$SPECPATH*>：

     ```
     driveletter:/PATH_TO_SPECTRUM
     driveletter:\PATH_TO_SPECTRUM
     ```

   - 在 Solaris 上，建立 /opt/SPECTRUM/spectrum60.env 並加入下列幾行：

     ```
     SPECROOT=PATH_TO_SPECTRUM
     SPECPATH=PATH_TO_SPECTRUM
     ```

   - 在 Linux 上，建立 /opt/SPECTRUM/spectrum80.env 並加入下列幾行：

     ```
     SPECROOT=PATH_TO_SPECTRUM
     SPECPATH=PATH_TO_SPECTRUM
     ```

7. 如果您傳輸的是 AlarmNotifier 和事件及 pcause 檔案，請驗證 <*$SPECROOT*>/SG-Support/CsResource/preferences/*.prf 中的路徑有正確的事件及 pcause 檔案位置。 也請驗證 ui= 和 lhandle= 選項。

8. 在 Solaris 和 Linux 平台上，於 /opt/SPECTRUM 目錄的 spectrum*.env 檔案中，將 BES_LIC_DIR 定義為 PATH_TO_spectrum/bin/VBNS/license。

9. 驗證 .hostrc 檔案包含本機電腦主機名稱和傳輸公用程式的來源電腦名稱。

10. 驗證 .LocalRegFile 包含正確的主要位置伺服器。

11. 驗證 vnmsh/.vnmshrc 包含正確的主要 SpectroSERVER 名稱。

12. 安裝 processd 服務：

    Windows：

    - 如果尚未安裝 SRAdmin，請依下列步驟進行安裝：

      ```
      shell> cd %SPECROOT%\Install-Tools\sdic\nt
      shell> .\sradmin --install
      shell> .\sradmin --start
      ```

    - 安裝 processd 服務：

      ```
      shell> cd %SPECROOT%\lib\SDPM
      shell> .\processd.exe --install --username USERNAME --password PASSWORD
      shell> .\processd.exe --start
      ```

      附註：如果 processd 未啓動，請將電腦重新開機。

    Solaris 和 Linux：

    - 將 <*$SPECROOT*>/lib/SDPM/processd_init.sh 複製至 /etc/init.d/processd。
    - 將 <*$SPECROOT*>/lib/SDPM/processd.pl 複製至 /etc/init.d。
    - 建立從 /etc/init.d/processd 連往 /etc/rc2.d/S99processd 的連結。
    - 使用 /etc/init.d/processd start 啓動 processd。

13. 驗證主要位置伺服器上的 .hostrc 檔案包含用於擷取 tools_bundle 檔案之電腦的主機名稱。

14. 在 shell 中設定 CLISESSID，以使用 CLI 公用程式：

    Windows：

    ```
    set CLISESSID=<NUMBER>
    ```

    Solaris 和 Linux：

    ```
    export CLISESSID=<NUMBER>
    ```

    NUMBER

    此爲 shell 的唯一號碼。

15. 確認 Notifier/.alarmrc 具有正確的 SetScript、ClearScript 和 UpdateScript 路徑。 這些指令檔位於您擷取 tools_bundle 檔案的 Notifier 目錄中。

16. 在 Windows 平台上，如果您要使用 AlarmNotifier 指令檔，請從 http://www.cygwin.com 安裝 Cygwin。 請確定 PATH 變數包含 Cygwin bin 目錄。

    您現在可以在此電腦上執行公用程式了。

更多資訊：

封裝 CA Spectrum 公用程式 (位於 p. 25)

第 2 章： 設定分散式 SpectroSERVER 環境

本節包含以下主題：

名稱解析需求 (位於 p. 31)
CA Spectrum 和多重介面 (位於 p. 31)
SpectroSERVER 間的通訊 (位於 p. 32)
連接埠衝突解決方法 (位於 p. 32)
DSS 環境需求 (位於 p. 34)
位置伺服器 (位於 p. 35)
範圍控制代碼 (位於 p. 39)
程序精靈 (processd) (位於 p. 40)
網路分割 (位於 p. 50)
分散式環境中的重複模型 (位於 p. 51)
主機資源配置檔案 (.hostrc) (位於 p. 52)
DSS 環境中的時區 (位於 p. 52)
SpectroSERVER 關機 (位於 p. 52)
設定範圍對應項目逾時 (位於 p. 53)
現有 DSS 設定中範圍對應的問題 (位於 p. 54)
清除舊範圍的相關問題 (位於 p. 56)

名稱解析需求

SpectroSERVER 系統或 OneClick Web 伺服器系統必須能夠將 SpectroSERVER 的主機名稱解析爲 IP 位址。

建議使用主機檔案來進行名稱解析。 這種作法可確保網路中斷時不會影響 SpectroSERVER 名稱解析。

CA Spectrum 和多重介面

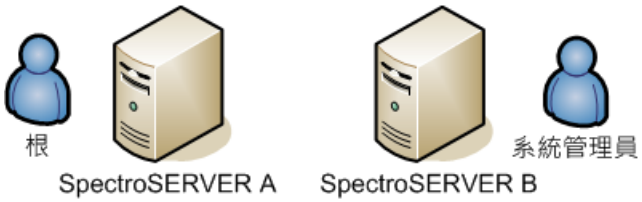
所有 CA Spectrum 伺服器都會繫結和接聽所有可用介面，並公告它們的非完整格式主機名稱。 如此可在配置管理拓撲時提供彈性。 爲了建立連線，CA Spectrum 元件伺服器會根據作業系統決定的順序來嘗試所有介面，直到連線成功爲止。

SpectroSERVER 間的通訊

使用相同的使用者帳戶，在分散式環境中安裝和執行 SpectroSERVER。 不需要設定任何其他的使用者模型配置。 以不同使用者身分安裝和執行的 SpectroSERVER，將無法通訊。

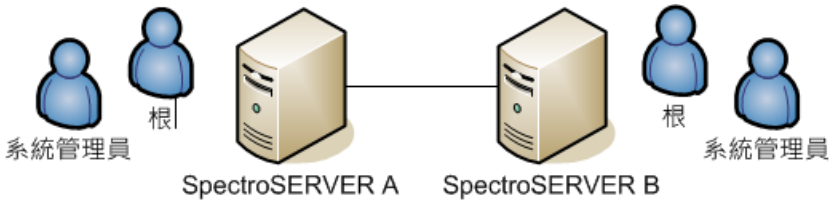
範例：SpectroSERVER A 和 B 無法通訊

在此範例中，SpectroSERVER A 以 "root" 身分執行。 而 SpectroSERVER B 是以 Administrator 身分執行。 因爲要求無法通過伺服器安全性而傳遞，因此無法建立連線：



範例：SpectroSERVER 至 SpectroSERVER 連線的配置

此範例顯示一位使用者以 "root" 身分登入 SpectroSERVER B，一位使用者以 "Administrator" 身分登入 SpectroSERVER A。 此配置可讓兩部 SpectroSERVER 進行通訊：



連接埠衝突解決方法

若要避免連接埠與其他應用程式發生衝突，您可以變更 CA Spectrum 程序和服務使用的預設連接埠號碼。

下表說明 CA Spectrum 程序和服務的預設連接埠。

附註：本表中的連接埠在適用處會註記「不用於遠端連線」。 CA Spectrum 未用於遠端連線的連接埠，通常就沒有防火牆的配置問題。

連接埠 (dec)	連接埠 (hex)	CA Spectrum 元件	通訊協定	注意
80	0x0050	OneClick/SRM	HTTP	無
162	0x00A2	SpectroSERVER	UDP	用來接聽 SNMP 設陷的連接埠。
3306	0x0CEA	SRM 資料的 mysql ArchMgr (Archive Manager)	TCP/ODBC/JDBC	只有在將 SRM 資料庫從 Windows 移轉至 Linux 或 Solaris 時，才用於遠端連線。
3307	0x0CEB	Solaris/Linux 上，BOXI 的 mysql	TCP	不用於遠端連線。
6844	0x1ABC	SDC	TCP	無
6400	0x1900	BOXI CMS	TCP	不用於遠端連線。
7777	0x1E61	CLI vnmshd	TCP	請參閱＜CLI 精靈 (vnmshd) (位於 p. 64)＞。
14001	0x36B1	OneClick 伺服器	TCP/CORBA	請參閱＜OneClick WebServer 主機 (位於 p. 59)＞。
14002	0x36B2	SpectroSERVER	TCP/CORBA	請參閱＜SpectroSERVER (位於 p. 62)＞。
14003	0x36B3	ArchMgr	TCP/CORBA	請參閱＜Archive Manager (位於 p. 63)＞。
14004	0x36B4	LocServ	TCP/CORBA	請參閱＜位置伺服器 (位於 p. 63)＞。
14006	0x36B6	命名服務	TCP/CORBA	無
31415	0x7AB7	透過 SpectroSERVER 進行 Telnet		無
46517	0xB5B5	sradmin	TCP	遠端管理精靈。 **附註：**如需 sradmin 程序的詳細資訊，請參閱《*安裝指南*》。

連接埠 (dec)	連接埠 (hex)	CA Spectrum 元件	通訊協定	注意
47870	0xBAFE	ArchMgr	TCP/SSAPI	請參閱＜Archive Manager (位於 p. 63)＞。
48879	0xBEEF	SpectroSERVER	TCP/SSAPI	請參閱＜SpectroSERVER (位於 p. 62)＞。
51966	0xCAFE	rcpd	TCP	請參閱＜遠端複製程序精靈 (rcpd) (位於 p. 64)＞。
56063	0xDAFF	LocServ	TCP	請參閱＜位置伺服器 (位於 p. 63)＞。
61904	0xF1D0	processd	TCP	此連接埠無法配置。
64222	0xFADE	TL1d	TCP/EPI	TL1 閘道代理程式
65259	0xFEEB	processd	TCP	此連接埠無法配置。

DSS 環境需求

您的分散式環境必須符合下列條件，才能啓用多重 SpectroSERVER 和多重範圍的模型化以代表這些伺服器：

- 每個範圍必須包含資料庫中模型類型及其關係的相同模型化類別。 此種複製可爲 CA Spectrum 智能提供一致的基礎。
- 每個範圍必須包含相同的使用者模型，您可在 OneClick [使用者] 索引標籤中加以檢視。
- 將不同範圍中模型化子網路間的連線數目減至最少。
- 限制在超過一個範圍中模型化之相同裝置的數目。
- 指派唯一範圍控制代碼給每個模型化範圍。 CA Spectrum 接著就能在 DSS 環境中辨別每個範圍。

更多資訊：

指派範圍控制代碼 (位於 p. 39)

位置伺服器

當您安裝 SpectroSERVER 時，也會自動安裝*位置伺服器*。 這個伺服器會識別並找出網路上的其他 CA Spectrum 服務。 CA Spectrum 程序使用位置伺服器來判斷這些服務是否執行中。 Processd 可以啓動和停止位置伺服器。

在分散式環境中，CA Spectrum 使用位置伺服器來維護 SpectroSERVER 範圍對應，以及提供連往用戶端應用程式的連線服務。 在 CA Spectrum 安裝期間，您需在範圍對應 (或 CA Spectrum 網域) 中指定一個位置伺服器作爲*主要位置伺服器 (MLS)*。

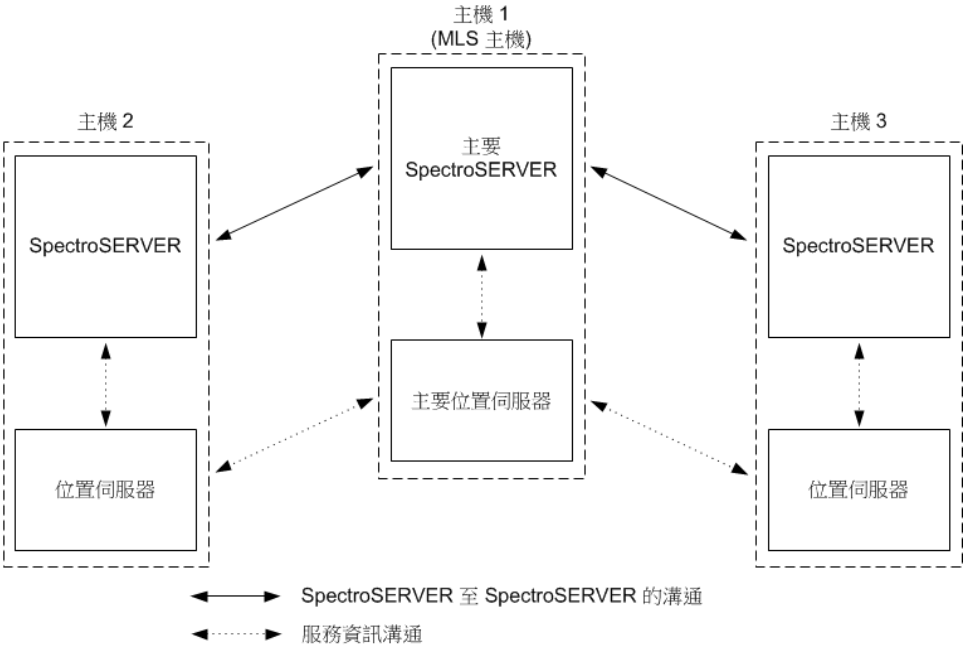
在高度可靠的電腦上安裝主要位置伺服器。 此伺服器會將服務公告傳播給各個位置伺服器，以及在位置伺服器間連絡服務位置。

在下圖中，主機 1 上的位置伺服器已配置爲此環境的主要位置伺服器。 主要位置伺服器與一個 SpectroSERVER 位於相同主機，因此該 SpectroSERVER 被視爲主要 SpectroSERVER。

所有伺服器對伺服器通訊都會透過主要位置伺服器主機進行路由。 兩個非 MLS SpectroSERVER 只能透過主要 SpectroSERVER 進行通訊。

圖中主機 2 上的 SpectroSERVER 具有分散式服務，以及在主機 3 的 SpectroSERVER 上有模型化資源。 所有遠端資源相關通訊都會透過主要位置伺服器主機進行路由。 主機 2 和主機 3 上的伺服器不會直接通訊。

重要！ 本圖反映的是包含一個 MLS 和多個子 SpectroSERVER 的兩層設定。 這種配置是建議的標準配置。 包含多個 MLS 及階層式關係 (多層部署) 的更複雜配置也是可行，但不建議如此配置。 多層部署只會增加複雜性，但無法提供更多功能。



位置伺服器如何互動

每個 CA Spectrum 安裝都包含下列兩個檔案：

- .LocalRegFile 可讓 CA Spectrum 程序找到其位置伺服器。
- LS/.locrc 則是本機安裝之位置伺服器的配置檔案。

這些檔案負責管理位置伺服器階層，以及每個位置伺服器和主要位置伺服器之間的互動。

在分散式 SpectroSERVER 部署中，位置伺服器的互動方式如下：

1. SSAPI 應用程式和伺服器讀取 .LocalRegFile，判斷要使用哪個位置伺服器。
2. 每個本機位置伺服器會讀取 /LS/.locrc 檔案，判斷哪個伺服器是主要位置伺服器。
3. 本機位置伺服器會連線至主要位置伺服器，尋找網路服務。 如果主要位置伺服器無法使用，其他位置伺服器會將服務資訊保留在記憶體中，直到主要位置伺服器可供使用。

主要位置伺服器連線

每個 SpectroSERVER 都必須能夠連線至被指定爲主要位置伺服器之伺服器上執行的 SpectroSERVER。 此項指定位於 <*$SPECROOT*>/LS/.locrc 檔案中。

連線至主要位置伺服器需要下列配置：

- 每個 SpectroSERVER 必須讓其他 SpectroSERVER 主機能夠對其連線。連線需要每部伺服器的 .hostrc 檔案中有適當項目。
- 每個 SpectroSERVER 需要有用於執行其他 SpectroSERVER 之使用者帳戶的 CA Spectrum 使用者模型。

附註： 主要範圍 (或 .locrc 檔案中 MAIN_LOCATION_HOST_NAME 的值) 在 CA Spectrum 中進行模型化。 每個 SpectroSERVER 的 VNM 拓撲都包含此範圍，除非已在其他位置模型化。 如果連往主要範圍的連線中斷，會在此模型上產生警報。

指定新的主要位置伺服器

您可以變更指定爲主要位置伺服器的電腦。

在 SpectroSERVER 上，請透過 [位置伺服器配置] 對話方塊或透過 .locrc 檔案指定新的主要位置伺服器。

在 OneClick Web 伺服器上，請在 OneClick [管理] 頁面的 CA Spectrum [配置] 頁面上設定主要位置伺服器資訊。 這個動作會將主要位置伺服器的名稱更新爲 OneClick Web 伺服器連線的伺服器。

附註：如果您變更 SpectroSERVER 上的主要位置伺服器但未更新 OneClick Web 伺服器，CA Spectrum 將無法正常運作。 如需詳細資訊，請參閱《*管理指南*》。

範例：在 SpectroSERVER 上指定新的主要位置伺服器

分散式網路由 Computer 1 至 4 組成，其中 Computer 1 被指定爲主要位置伺服器。 若要將 Computer 4 設爲主要位置伺服器，請先將 Computer 4 重新配置爲新的主要位置伺服器。 接著再重新配置 Computer 1、2 和 3，指向作爲主要位置伺服器的 Computer 4。

請依循下列步驟:

1. 開啓 SpectroSERVER 控制台。
2. 從 [配置] 功能表中選取 [位置伺服器]。

   [位置伺服器配置] 對話方塊隨即開啓。
3. 變更 [位置伺服器特性] 區域中的 [主要 LS 主機] 欄位，指向 Computer 4。

附註：您也可以透過編輯 <*$SPECROOT*>/LS 目錄中的 .locrc 檔案，變更主要位置伺服器。 .locrc 檔案中的下列項目會識別主要位置伺服器：

```
MAIN_LOCATION_HOST_NAME=Computer 4
```

更多資訊：

建立容錯 (位於 p. 76)

範圍對應完整性保護

CA Spectrum 使用者通常會保持獨立的生產和測試環境。 防止測試環境中的 SpectroSERVER 連線至生產環境 SpectroSERVER 中的 SpectroSERVER 作爲其主要位置伺服器 (或者反過來)，可以保護範圍對應的完整性。

範圍控制代碼

分散式部署中的 SpectroSERVER 可以根據*範圍控制代碼*識別每個範圍。 每個範圍都必須有唯一的範圍控制代碼，這是可被 4 整除的數字，且位於 4 至 16,376 之間。 (在十六進位格式中，控制代碼位於 0x100000 至 0xffe00000 之間，且較低的 20 個位元設爲零)。 編碼的範圍控制代碼會出現在與該範圍關聯之所有檢視的上方。

附註： 建議您對範圍控制代碼使用連續編號格式。 您可以將範圍控制代碼關聯至重要的範圍功能，例如建置號碼或子網路 IP 位址的一部份。 但您輸入的項目會編碼爲範圍控制代碼的 12 個最重要位元。 因此，產生的結果看起來會與適當的範圍功能沒有關聯。

指派範圍控制代碼

您可在安裝 CA Spectrum 時透過 SpectroSERVER [驗證] 對話方塊指派範圍控制代碼，或是使用公用程式 lh_set 來指派。

重要！ 第一次執行 SpectroSERVER 前，請先執行 lh_set 公用程式。 否則，CA Spectrum 會指派預設範圍控制代碼，每次 CA Spectrum 指派時都相同。 如此會在配置多個範圍時，建立重複的範圍控制代碼。 這種範圍將無法供同一個應用程式同時存取。

請依循下列步驟:

1. 導覽至 SS 目錄。
2. 輸入下列命令：

   ```
   ../SS-Tools/lh_set <範圍控制代碼>
   ```

   新的範圍控制代碼可指定爲十進位或十六進位標記法。 如果您使用十進位標記法，lh_set 公用程式會將您的輸入轉換爲十六進位的範圍控制代碼。

附註：如需詳細資訊，請參閱《*安裝指南*》。

更多資訊：

DSS 環境需求 (位於 p. 34)

變更範圍控制代碼

如果 SpectroSERVER 已啓動，變更 SpectroSERVER 資料庫範圍控制代碼的程序就需要更多步驟。 範圍控制代碼隨附於 SpectroSERVER 資料庫之每個模型的模型控制代碼中。 因此，變更範圍控制代碼需要將所有模型控制代碼從舊的範圍控制代碼轉換爲新的範圍控制代碼。

請變更於啓動時自動建立之所有模型的範圍控制代碼，並更新數個資源檔案中的範圍控制代碼。

請依循下列步驟:

1. 使用「模型化閘道」工具，從您想變更其範圍控制代碼的 SpectroSERVER 中匯出模型。
2. 關閉 SpectroSERVER。
3. 使用 SSdbload 公用程式，初始化資料庫。

   附註：如需詳細資訊，請參閱《*資料庫管理指南*》。
4. 使用 lh_set 公用程式，指派新的範圍控制代碼。
5. 啓動 SpectroSERVER。
6. 使用「模型化閘道」工具，將模型匯入至 SpectroSERVER。

附註：如需詳細資訊，請參閱《模型化閘道工具組指南》。

程序精靈 (processd)

CA Spectrum 使用稱爲「程序精靈」(processd) 的程序啓動和追蹤精靈，可讓您控制 DSS 環境中多個伺服器上的程序。 此精靈會在 CA Spectrum 控制台之類的應用程式發出要求時，啓動程序。 您也可以使用安裝工單配置 processd，在系統啓動時自動開始程序。 安裝工單也能自動重新啓動意外停止的重要程序。

CA Spectrum 安裝程式在 Solaris 系統 (必須以 root 身分執行) 和 Windows 系統 (以 LocalSystem 帳戶執行) 上，會配置 processd 自動啓動。

只有在應用程式要求這類動作時，processd 才會啓動並監控程序。 processd 通常在背景運作。

如果程序未啓動或未正常運作，processd 會將錯誤訊息寫入 *<$SPECROOT>*\lib\SDPM\processd_log 檔案。錯誤訊息包含的資訊可用來找出問題。

重新啓動時，processd 會建立一個 processd_log.bak 檔案，保留舊的錯誤訊息並將新的錯誤訊息附加至 processd_log 檔案。

重要！ 您必須徹底熟悉 CA Spectrum、分散式網路和網路配置，才能進行本節所述的任何程序。

更多資訊：

安裝工單檔案 (位於 p. 43)

Windows 和 Solaris 環境中的 Processd 差異

由於平台本身的安全架構差異，processd 精靈在 Windows 和 Solaris 平台中的運作方式略有不同。

- 當您從 Windows 環境的遠端連線要求啓動程序，程序會以 [Windows 服務組態] 對話方塊的使用者身分啓動。 而在 Solaris 上，程序則是以發出要求的使用者身分啓動。
- 至於在使用者登出後 (或在任何人登入前) 需繼續執行的伺服器程序，在 Windows 環境中會使用 SERVERPROCESS 欄位。 爲了避免 Windows 在登出期間將這些程序關閉，這些程序是以 [Windows 服務組態] 對話方塊中指定的使用者身分啓動。 而在 Solaris 上，程序則是以安裝工單中指定的 TICKETUSER 身分啓動。

更多資訊：

安裝工單檔案 (位於 p. 43)

變更 Processd 中的 Windows 密碼

在 Windows 上的安裝期間，會出現提示，請您在 [Windows 服務組態] 對話方塊中輸入使用者名稱和密碼。 processd 精靈會使用此使用者名稱和密碼，以指定的使用者身分啓動 CA Spectrum 程序。 提供安全性資訊，可在沒有使用者登入時讓這些程序執行。

附註：如需詳細資訊，請參閱《*安裝指南*》。

如果您在 [Windows 服務組態] 對話方塊中指定的密碼變更或過期，您可使用新密碼來更新 processd。

重要！ 如果密碼包含特殊字元 (例如驚嘆號 (!))，請使用反斜線 (\) 逸出這些字元。 或者您可以在命令提示字元中變更密碼。

請依循下列步驟:

1. 以本機 Administrators 群組的成員身分登入。

   您必須是 CA Spectrum Users 群組的成員以及系統管理員，才能變更 processd 使用者名稱和密碼。

2. 開啓命令提示字元。
3. 導覽至 /lib/SDPM 目錄。
4. 輸入下列命令：

   ```
   processd --install --username user --password newpassword
   ```

   使用者

   指定使用者名稱。

   newpassword

   指定此使用者名稱的新密碼。

   如果使用者名稱或密碼包含 shell 會錯誤解譯的特殊字元，請用引號將使用者名稱或密碼括起來。 例如，使用者名稱包含網域名稱和使用者，並使用反斜線 (\) 分隔符號。 密碼包含星號 (*)。 這些字元都是特殊字元。 因此，使用者名稱和密碼需以引號括起來，如下列範例所示：

   ```
   processd --install --username “DOMAIN\JSmith” --password “283EJ*”
   ```

附註： 如果 Windows 使用者名稱的密碼變更，但 processd 服務密碼未更新，則 processd 不會啓動。 Windows 事件日誌中會產生下列事件：

- 使用目前密碼的登入嘗試失敗，並有下列錯誤：
  登入失敗：不明的使用者名稱或錯誤密碼。
- CA Spectrum 程序精靈服務無法啓動，因爲發生下列錯誤：
  登入失敗所以服務無法啓動。

安裝工單檔案

您可以配置 processd 在系統啓動時開始程序。 如果程序停止執行，您也可以選取程序以讓其重新啓動。 稱爲「*安裝工單*」的檔案支援這種功能。

安裝工單使用下列兩個目錄中的檔案：

- <*Spectrum 安裝目錄*>/lib/SDPM/partslist
- <*Spectrum 安裝目錄*>/lib/SDPM/runtime

partslist 目錄包含個別的安裝工單檔案。

runtime 目錄包含格式爲 *<PID>*.rt 的編碼檔案，其中 *<PID>* 是執行中程序的程序 ID。

附註：SDPM 代表 CA Spectrum Distributed Process Manager。

您可新增安裝工單至 partslist 目錄。 新的安裝工單必須符合特定的格式化規則。 重新啓動 processd，識別 partslist 目錄中新的或修改過的安裝工單。 processd 啓動時就會讀取這些檔案，並儲存在快取記憶體中以供日後使用。

安裝工單檔案所遵循的命名慣例，可指出其包含之配置資訊的程序。 安裝工單檔案名稱的格式爲 *<PARTNAME>*.idb。 *<PARTNAME>* 變數是內部索引鍵，用以識別 processd 控制的程序。

安裝工單使用下列定義欄位。 格式爲 *<fieldname>;<value>*；其中 *<fieldname>* 是下列顯示的其中一個名稱。 *<value>* 是提供相對應欄位名稱之定義的字串。 不支援引號。

PARTNAME

使用多字元字串 (不含空格) 識別特定的程序/應用程式。 此應用程式的安裝工單具有 *<PARTNAME>*.idb 格式的檔案名稱。 *<PARTNAME>* 是程序名稱。

APPNAME

將應用程式的名稱定義爲多字元字串。

WORKPATH

指定您要在哪裡執行應用程式。 現在提供此欄位的值，*稍後才*能作爲 ARGV 欄位的一部份 (如後所述)。

LOGNAMEPATH

指定應用程式輸出的記錄檔。 此欄位的開頭必須是 *<$SPECROOT>* 或 *<$WORKPATH>*。 不允許使用空格。

ADMINPRIVS

此項保留，僅供用於 Purism 供應的安裝工單。 在您建立的安裝工單中，請將此項標示爲註解。

AUTORESTART

指定程序停止時是否自動重新啓動。 如果安裝工單中不含此欄位，預設會停用自動重新啓動。 接受的值爲 Y、y、N、n。

AUTOBOOTSTART

指定當 processd 啓動時是否啓動程序。 如果安裝工單中不含此欄位，預設會停用此功能。 接受的值爲 Y、y、N、n。

STATEBASED

指定當啓動時，程序是否具有一個以上的狀態。 通常，當程序已備妥可供通訊時，會傳送「已備妥」工單。 此欄位保留，僅供用於 Purism 供應的安裝工單。 如果安裝工單中不含此欄位，預設會停用此功能。 接受的值爲 Y、y、N、n。

NUMPROCS

指定一個平台上可以執行的程序例項數目。 接受數值。

RETRYTIMEOUT

指定 processd 在應用程式失敗後經過幾秒，將嘗試重新啓動應用程式。

TICKETUSER

定義在設定了 AUTOBOOTSTART 和/或 AUTORESTART 欄位時，獲授權可執行程序之使用者的使用者名稱。 只有在安裝工單中包含這些欄位時，才需指定此欄位。 此欄位不適用於 Windows，因爲所有程序都以 processd 安裝使用者的身分執行。

RETRYMAX

指定 processd 在指定的 RETRYTIMEOUT 期間內，嘗試重新啓動應用程式的次數。

STARTPRIORITY

指出相對於其他程序，此程序的相對啓動優先順序。 使用下列值：

- 10 爲獨立程序
- 20 爲需依賴獨立程序的程序
- 30 爲需依賴 SpectroSERVER 的程序

SERVERPROCESS

向 processd 指出在使用者登出後，是否繼續執行程序。 啓用此欄位時，程序一律以獲授權可執行 processd 服務的使用者身分啓動。 此欄位僅適用於 Windows 環境。 下列程序的預設值是 'yes'：

- Archive Manager (ARCHMGR.idb)
- 位置伺服器 (LOCSERV.idb)
- SpectroSERVER (SS.idb)

SERVICE

指定工單是否代表 Windows 服務。 可讓 processd 從 [服務控制管理員] 管理 Windows 服務。

ENV

新增一或多個變數至應用程式環境。 如有多個值，則各列在單獨一行，值和行結尾的分號之間使用巨集 <CSPATHSEP>。

ARGV

定義程序的引數清單，其中包括可執行檔路徑和任意數量的引數 (可使用空格)。

更多資訊：

程序精靈 (processd) (位於 p. 40)
Windows 和 Solaris 環境中的 Processd 差異 (位於 p. 41)

範例：安裝工單

安裝工單套用下列語法慣例：

- 井號 (#) 開頭的行視同註解。
- 所有欄位名稱和每個欄位名稱後面的所有值之後，都必須加上分號 (;)。
- 第二個分號後面接的項目會被忽略。
- 可使用四個巨集：<CSEXE>、<CSBAT>、<CSCMD> 和 <CSPATHSEP>。 視平台而定，可替換適當的定義。 使用 <CSPATHSEP> 取代實際的路徑分隔符號，以避免剖析衝突。

以下範例顯示有效的安裝工單：

```
# Processd Install Ticket for SpectroSERVER Daemon.
PARTNAME;SS;
APPNAME;SpectroSERVER Daemon;
WORKPATH;$SPECROOT/SS;
LOGNAMEPATH;$WORKPATH/VNM.OUT;
ADMINPRIVS;y;
#AUTORESTART;N;
#AUTOBOOTSTART;N;
STATEBASED;y;
NUMPROCS;3; // unlimited
RETRYTIMEOUT;0; // seconds
#TICKETUSER;$USER;
RETRYMAX;0;     // retries
STARTPRIORITY;20;
SERVERPROCESS;Y;
#ENV;<var>=<value>;
ARGV;$SPECROOT/SS/SpectroSERVER<CSEXE>;
```

以下範例顯示另一個有效的安裝工單：

```
# Processd Install Ticket for Visibroker Naming Service
PARTNAME;NAMINGSERVICE;
APPNAME;Visibroker Naming Service;
WORKPATH;$SPECROOT/bin/VBNS;
LOGNAMEPATH;$WORKPATH/NAMINGSERVICE.OUT;
ADMINPRIVS;y;
AUTORESTART;y;
AUTOBOOTSTART;y;
#STATEBASED;N;
NUMPROCS;1; // one per host
RETRYTIMEOUT;600; // 10 minutes
#TICKETUSER;$USER;
RETRYMAX;5;     // 5 retries
STARTPRIORITY;10;
SERVERPROCESS;Y;
#ENV;<var>=<value>;
ENV;CLASSPATH=$SPECROOT/lib/vbjorb.jar<CSPATHSEP>;
ENV;CLASSPATH=$SPECROOT/lib/vbsec.jar<CSPATHSEP>;
ENV;CLASSPATH=$SPECROOT/lib/lm.jar<CSPATHSEP>;
ENV;CLASSPATH=$SPECROOT/lib/sanct6.jar<CSPATHSEP>;
ARGV;
$SPECROOT/bin/JavaApps/bin/nameserv<CSEXE>
-DORBpropStorage=
$SPECROOT/.corbarc
-Dvbroker.orb.admDir=
$SPECROOT/bin/VBNS
-Dborland.enterprise.licenseDir=
$SPECROOT/bin/VBNS/license
-Dborland.enterprise.licenseDefaultDir=
$SPECROOT/bin/VBNS/license
-Djava.endorsed.dirs=
$SPECROOT/lib/endorsed
-Dorg.omg.CORBA.ORBClass=
com.inprise.vbroker.orb.ORB
-Dorg.omg.CORBA.ORBSingletonClass=
com.inprise.vbroker.orb.ORB
com.inprise.vbroker.naming.ExtFactory;
```

啓動新的安裝工單程序

如果您新增了安裝工單檔案，則可使用重新啓動選項來啓動工單中指定的程序。 使用重新啓動選項，您不需停止再重新啓動 processd 和它監控的所有程序。

Solaris 上請遵循下列步驟：

1. 以 root 身分登入。
2. 導覽至 /lib/SDPM 目錄。
3. 輸入下列命令：

   ```
   processd.pl restart
   ```

   程序隨即啓動。

Windows 上請遵循下列步驟：

1. 以 CA Spectrum Users 群組的成員身分登入。
2. 開啓命令提示字元。
3. 導覽至 /lib/SDPM 目錄。
4. 輸入下列命令：

   ```
   perl processd.pl restart
   ```

   程序隨即啓動。

停止再重新啓動 Processd

如果您懷疑 lib/SDPM/runtime 目錄已損毀，請停止再重新啓動 processd。

Solaris 上的 請依循下列步驟:

1. 以 root 身分登入。
2. 導覽至 /lib/SDPM 目錄。
3. 輸入下列命令：

   ```
   processd.pl stop
   ```

4. 驗證所有 CA Spectrum 程序都已關閉 (否則 SpectroSERVER 會發生問題)。

5. 移除 /lib/SDPM/runtime 目錄中的所有項目。
6. 輸入下列命令，重新啓動 processd：

   ```
   processd.pl start
   ```

Windows 上的 請依循下列步驟:

1. 以 CA Spectrum Users 群組的成員身分登入。
2. 開啓命令提示字元。
3. 導覽至 /lib/SDPM 目錄。
4. 輸入下列命令：

   ```
   perl processd.pl stop (或 start)
   ```

附註： 在 Windows 上，processd.pl *<start/stop>* 命令也會停止和啓動執行爲 NT 服務的 CA Spectrum 程序 (例如 MySQL 和 VisiBroker)。

更多資訊：

在 Solaris 環境中關閉 SpectroSERVER (位於 p. 53)

Userconf 程序在安裝期間如何運作

Userconf 程序會在背景執行 processd。 Userconf 可讓您在安裝期間及安裝後，於使用者登入 CA Spectrum 時執行使用者-特定的 CA Spectrum 配置。

Userconf 程序在 CA Spectrum 安裝期間會執行下列工作：

1. 執行 userconf -install %SPECROOT%，進行下列變更：
   - 新增 SPECROOT 和 SPECPATH 至系統環境。
   - 新增 $SPECPATH\lib 至 $PATH 系統變數。
   - 如果安裝目錄已變更，則從路徑中移除舊的 $SPECPATH\lib 項目。
   - 從路徑中移除 %SPECPATH%\lib (若存在)。
   - 修改登錄，以便每次使用者登入時都執行 userconf (無引數)。

2. 執行 userconf -start，啓動 processd。 -start 旗標會啓動 processd，但不驗證使用者是否爲 CA Spectrum Users 群組的成員。

   附註： 安裝程式會將目前使用者加入 CA Spectrum Users 群組。 不過，使用者需先登出再登入，變更才會生效。 加入 -start 旗標，以啓動 processd 但不驗證使用者是否爲 CA Spectrum Users 群組的成員。 安裝會繼續進行，不要求使用者先登出再行登入。

3. 如果使用者是第一次安裝 Exceed，則執行 userconf -restart，停止再重新啓動 processd (發生於 CA Spectrum 安裝後)。 Processd 接著會更新 PATH 環境變更 (此爲 Exceed 安裝的一部份)。

使用者登入時 Userconf 如何運作

安裝會將 userconf 程序新增至登錄 Run 機碼。 當使用者登入時，程序會自動啓動，並檢查使用者是否爲 CA Spectrum Users 群組的成員。 userconf 程序會套用下列規則：

- 如果使用者是 CA Spectrum Users 群組的成員，userconf 會驗證 cygwin 和 Exceed，視需要加以重新配置，然後再啓動 processd。 如果已安裝 Microsoft VC++，userconf 會據以適當配置，以便搭配 CA Spectrum SDK 使用。
- 如果使用者不是 Users 群組的成員，則會出現訊息，指出 CA Spectrum 已安裝，但使用者不是 CA Spectrum Users 群組的成員。

  若要讓使用者能夠使用 CA Spectrum，請將其使用者帳戶加入 CA Spectrum Users 群組中。 接著使用者必須先登出再登入。

  不想使用 CA Spectrum 的使用者，可以選取 [使用者不想執行 Spectrum] 核取方塊。 若使用此選項，userconf 在使用者登入時會繼續執行，但程序會無訊息地結束。

  附註：若要再次查看訊息，請執行下列命令：

  ```
  userconf -install %SPECROOT%
  ```

網路分割

當您模型化的網路已經配置好，網路分割就不重要。 不過，如果您要建立多個範圍，而且想使用「搜索」來建立拓撲階層，那麼請使用支援「搜索」的方法。

您的目標是要將範圍中的重複模型以及範圍間的連線數目降至最低。 爲便於找出錯誤原因，請複製一些模型，例如連接兩個 IP 子網路的路由器。

網路分割提供下列選項：

- 建立屬於子網路的分割，其由 IP 位址界限定義。 IP 位址範圍可提供限制搜索探勘的最方便方法。
- 如果 IP 位址界限無法用於隔離範圍，請使用唯一的社群名稱來區別每個範圍中的裝置。 這個方法比較麻煩。 新增名稱至每個裝置的社群名稱表。
- 根據另一個參數 (例如時區或狀態，包含多個 IP 子網路位址) 建立分割。 在大部份情況下，您必須手動模型化您的網路。 但請保持範圍間連線數盡可能少，並觀察模型化規則。 連線過多會損及 DSS 的優勢。

附註：如需詳細資訊，請參閱《*模型化和管理 IT 基礎架構管理指南*》。

分散式環境中的重複模型

在 DSS 環境中，CA Spectrum 會將一個模型指定爲「主目錄」模型，以便追蹤重複模型 (例如，存在於多個範圍中的使用者模型)。 主目錄模型位於在主機上執行、並有 "root" 主要位置伺服器 (MLS) 的 SpectroSERVER 中。

重要！ 如果您變更 DSS 環境中的 root MLS，所有重複模型的狀態都會更新以符合 MLS 上的「主目錄」模型。 MLS 是 DSS 環境中的最終授權。

CA Spectrum 使用主目錄模型來同步所有重複模型中的資訊。 對重複模型 (非主目錄模型) 進行的修改要求，會轉送至主目錄模型的範圍。 主目錄模型接著再將要求散佈至所有其他重複模型。

無法連絡主目錄範圍錯誤

當無法連絡您正在編輯之模型的主目錄範圍時，編輯作業便會失敗。 OneClick 會報告關係錯誤，例如下列來自 SpectroSERVER 的錯誤：

因爲編輯中的模型存在於多個範圍，且無法連絡其主目錄範圍，因此作業失敗。

這種錯誤類型可能代表無法取得 "root" MLS 上執行之 SpectroSERVER 的主目錄模型。 在分散式環境中新增或刪除重複模型時，會透過主目錄範圍將其路由傳送。 如果該 SpectroSERVER 關機或無法連絡，關係便會失敗並產生錯誤。

下列情況會發生無法連絡主目錄範圍：

- 主 SpectroSERVER 或是重複模型和主 SpectroSERVER 間的中間 SpectroSERVER 未執行。
- 發生網路問題，因此無法存取 SpectroSERVER。
- 發生安全問題，因此無法與 SpectroSERVER 通訊。

主機資源配置檔案 (.hostrc)

.hostrc 檔案可限制用戶端應用程式存取每個本機 SpectroSERVER。 用戶端應用程式無法連線至本機範圍，除非 .hostrc 檔案已配置爲允許此存取。 您可使用文字編輯器來編輯 .hostrc 檔案。

附註：如需詳細資訊，請參閱《*管理指南*》。

依預設，.hostrc 檔案一開始是以本機主機名稱進行安裝，並且限制所有遠端存取。 加入個別主機名稱，可啓用對該伺服器的連線。 加入 "+" 符號則可啓用對所有遠端伺服器的無限制存取。 對 .hostrc 檔案加入機器名稱，可啓用個別遠端伺服器的用戶端存取。

CA Spectrum 根據主機名稱或 IP 位址，實作多層次的主機安全性。 建議您在 .hostrc 檔案中列出主機名稱，包括與主機名稱關聯的所有 IP 位址。 若是使用 IP 位址來連線，那麼依主機名稱啓動的連線嘗試便有可能失敗。

DSS 環境中的時區

在跨越多個時區的 DSS 安裝中，SpectroSERVER 上發生的所有活動都會顯示爲該伺服器的當地時間，包括已排程的事件。 在 OneClick 上建立且套用至模型化裝置的所有排程，都是依據 SpectroSERVER 或裝置範圍的當地時間而開始和結束。

附註：如需詳細資訊，請參閱《*操作員指南*》。

SpectroSERVER 關機

建議您在關閉執行 SpectroSERVER 的伺服器前，先使用 CA Spectrum 控制台關閉 SpectroSERVER。 不過，如果您使用作業系統關機程序來關閉 SpectroSERVER，會增加 processd 停止的所需時間。

在 Solaris 環境中關閉 SpectroSERVER

在 Solaris 環境中，processd 預設會等候 20 秒，待所有子程序都關閉後才會停止。 如果 SpectroSERVER 關機費時過久，processd 就需等候更久的時間才能停止。 可使用 PROCESSD_SHUTDOWN_TIMEOUT 環境變數變更預設值。 新增此變數及其適當值 (單位為毫秒) 至 *<$SPECROOT>*/spectrum60.env 檔案。

例如，若想讓 processd 等候 60 秒以讓所有子程序都關閉，請將下面這行加入 spectrum60.env 檔案：

```
PROCESSD_SHUTDOWN_TIMEOUT=60000
```

然後停止再重新啟動 processd (位於 p. 48)，讓變更生效。

在 Windows 環境中關閉 SpectroSERVER

在 Windows 環境中，processd 會等待所有子程序關閉，然後才停止。 不過，Windows 登錄設定 WaitToKillServiceTimeout 可以設定開始 Windows 關機作業後，Windows 要等待所有服務停止的時間長短。 如果在超過 WaitToKillServiceTimeout 後，processd 之類的服務仍在執行中，Windows 就會終止此服務。 預設值是 20 秒，此值不是每次都夠讓 SpectroSERVER 在系統關機時完全關閉。 您可以增加此逾時值。

請依循下列步驟:

1. 開啟 [登錄編輯程式]。
2. 前往 HKEY_LOCAL_MACHINE/SYSTEM/CurrentControlSet/Control。
3. 按一下 Control 鍵。
4. 按兩下右窗格中的 WaitToKillServiceTimeout 值。
5. 在視窗中，將值變更為最高 600,000 毫秒 (10 分鐘) 的值。
6. 按一下 [確定]。
7. 重新啟動 Windows 伺服器，讓變更生效。

設定範圍對應項目逾時

依預設，範圍對應中的每個範圍項目都會無限期地存在。 使用位置伺服器配置檔案 (.locrc)，您可以指定範圍項目過多久後將逾時，並從範圍對應中自動移除。

附註：在舊版 CA Spectrum 中，範圍項目在一小時後逾時，並會從範圍對應中自動移除。 從 CA Spectrum 9.2.2 開始，範圍項目依預設不再逾時。

重要！ *不*建議使用逾時值來從範圍對應自動移除項目。 請保留預設的逾時值。 如有需要，請使用 MapUpdate 命令，從範圍對應移除項目。 如需詳細資訊，請參閱《*資料庫管理使用者指南*》。

請依循下列步驟:

1. 導覽至 <*$SPECROOT*>/LS 目錄。
2. 在 .locrc 檔案中更新或新增下列項目：

   MET_INTERVAL=*time_out_value*

   time_out_value

   指定範圍項目會在多久後逾時 (單位爲毫秒)，並從範圍對應中自動移除。

   預設值：0 毫秒 (無逾時)

現有 DSS 設定中範圍對應的問題

徵兆：

在複製其他安裝以便設定個別分散式環境時，範圍對應發生問題。 我注意，因主要無法使用而切換至次要 SpectroSERVER 時發生失敗，且使用者安全性也有問題。

解決方案：

若要複製範圍對應，請移除對先前 DSS 設定的所有參考。 範圍對應不能包含對先前 DSS 設定的任何參考。 請使用下列程序，從現有 DSS 設定對應範圍。

請依循下列步驟:

1. 如下所示分離 SpectroSERVER：
   a. 將每個 SpectroSERVER 設爲獨立伺服器。

      位置伺服器現在指向其安裝位置的本機伺服器。

   b. 驗證每個 SpectroSERVER 的範圍對應都不同。

      附註：在現有的 DSS 設定中，範圍對應是包含所有伺服器的單一對應。
2. 儲存 SpectroSERVER 資料庫。

   範圍對應已清除。
3. 變更這些 SpectroSERVER 的 MLS，還原原始 MLS。

   附註：請勿變更原始 DSS 設定。
4. 重新載入不同主機上每個 SpectroSERVER 的資料庫。
5. 在新 DSS 設定中選取 SpectroSERVER 作爲您的 MLS，並將其他 SpectroSERVER 指向它。

   使用者模型中的 dupModeList 會據以更新。
6. 從範圍對應移除次要項目 (如果存在)。
7. 設定次要 SpectroSERVER 項目。
8. 從主要 SpectroSERVER 載入資料庫至新的 DSS。

   範圍會從現有 DSS 設定進行對應。

更多資訊：

設定分散式 SpectroSERVER 環境 (位於 p. 31)

清除舊範圍的相關問題

徵狀：

我們的 CA Spectrum 生產環境包含單一分散式安裝。 環境中有多部 SpectroSERVER 與 OneClick 伺服器，全都使用相同的主要位置伺服器。

我們決定從生產環境中移除兩部 SpectroSERVER 與一部 OneClick 伺服器，以建立個別的開發環境。 我們配置了開發 SpectroSERVER 與 OneClick 伺服器，以參考新的位置伺服器。 我們更新了 .hostrc 檔案，使每個環境中的伺服器都只能存取其各自的位置伺服器。 此外，我們也更新了生產環境中的範圍對應，以移除開發範圍。

但是，當我檢視開發 OneClick 伺服器上的監控範圍清單時，我仍看見所有舊的生產範圍。 由於 .hostrc 檔案已變更，這些範圍在清單中顯示爲「沒有權限」。

解決方法：

根據您所說明的設定，目前您有兩個個別的環境。 但是，您的開發範圍對應快取了過時的生產範圍。 您已從生產環境中移除過時的範圍，但並未從開發環境中加以移除。 因此，您必須手動從開發環境中移除過時的範圍。

有兩種選項可供您移除過時範圍。

解決方法：

您可以使用下列命令，從開發範圍對應中移除過時的範圍：

```
MapUpdate -remove landscape handle
```

解決方法：

您可以重新啟動所有 SpectroSERVER 程序。 如需詳細資訊，請參閱停止再重新啟動 Processd (位於 p. 48)。接著，重新啟動 OneClick 伺服器。如需 OneClick 伺服器管理的詳細資訊，請參閱《管理員指南》。

重新啟動伺服器後，即可解決問題。

第 3 章： 分散式 SpectroSERVER 環境中的跨防火牆通訊

本節包含以下主題：

跨防火牆通訊 (位於 p. 57)
SpectroSERVER 和 OneClick Web 伺服器的跨防火牆通訊 (位於 p. 58)
OneClick 預設連接埠和防火牆 (位於 p. 59)
遠端 SpectroSERVER 和防火牆 (位於 p. 59)
主要和次要 SpectroSERVER 的跨防火牆通訊 (位於 p. 61)
NAT 防火牆環境的 CA Spectrum 配置檔案 (位於 p. 61)
預設連接埠配置 (位於 p. 62)

跨防火牆通訊

跨防火牆通訊適用於許多網路環境。 在分散式環境中，防火牆可能會影響到您的部署。

附註：開始進行分散式 CA Spectrum 安裝前，請配置防火牆爲啓用連接埠 46517 (連接埠 sradmin) 上的流量。

啓用 CA Spectrum 元件間通訊的可用選項，依您部署的防火牆類型而定。 還有其他因素 (例如成本和所需的安全性層級) 也有影響。 這些因素包括虛擬私人網路 (VPN)、節點對節點通道，以及在封包通過防火牆前將其封裝的 Proxy。

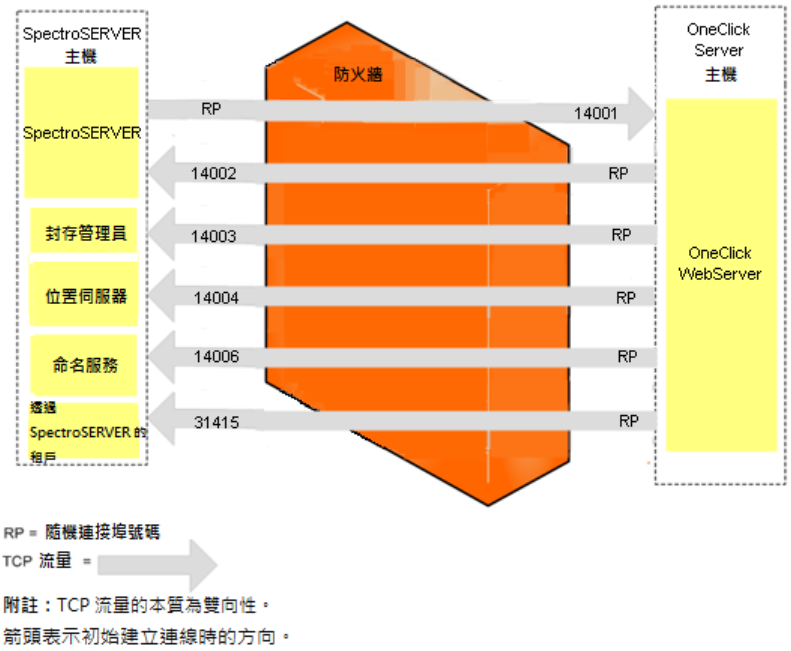
建議您與網路的防火牆管理員合作，找出可讓 CA Spectrum 和 OneClick 流量周遊通過防火牆的策略。

SpectroSERVER 和 OneClick Web 伺服器的跨防火牆通訊

OneClick Web 伺服器需與 SpectroSERVER 主機系統上的程序通訊，以便收集要顯示在 OneClick 用戶端上的資料。 通常由 OneClick Web 伺服器啓動這種通訊。

OneClick Web 伺服器會建立連往特定 SpectroSERVER 主機端 TCP 連接埠的連線。 Web 伺服器使用這些連接埠來傳送要求和接收回應。 不過，OneClick 使用單一接聽連接埠 (預設爲 14001)。 SpectroSERVER 會啓動連往該連接埠的連線。 因此，通常需要修改防火牆配置。 SpectroSERVER 使用雙向 IIOP (網際網路 ORB 間通訊協定) 來與其 CORBA 用戶端通訊。

下圖說明 OneClick Web 伺服器與 SpectroSERVER 通訊所需的 IP 連線。 使用 TCP 的時候，連線都是從隨機連接埠連至特定的固定連接埠。



附註：在容錯配置中，OneClick 伺服器主機及次要 SpectroSERVER 主機之間的相同連接埠必須開啓，如果執行次要 Archive Manager，則包括連接埠 14003。

OneClick 預設連接埠和防火牆

HTTP 接聽連接埠

OneClick 用於 HTTP 通訊的預設連接埠是連接埠 80。 如果您配置 OneClick Web 伺服器爲使用連接埠 80 以外的項目，則也必須設定您的防火牆以允許此流量。

附註：如需詳細資訊，請參閱《*管理指南*》和《*安裝指南*》。

在 Windows XP SP2 平台上，選擇啓用 Windows 防火牆的 OneClick 使用者在執行 OneClick 主控台時可能會發生問題。

附註：如需配置 Windows 防火牆的詳細資訊，請參閱 Microsoft 知識庫文章 842242，網址爲 http://support.microsoft.com。

CORBA 接聽連接埠

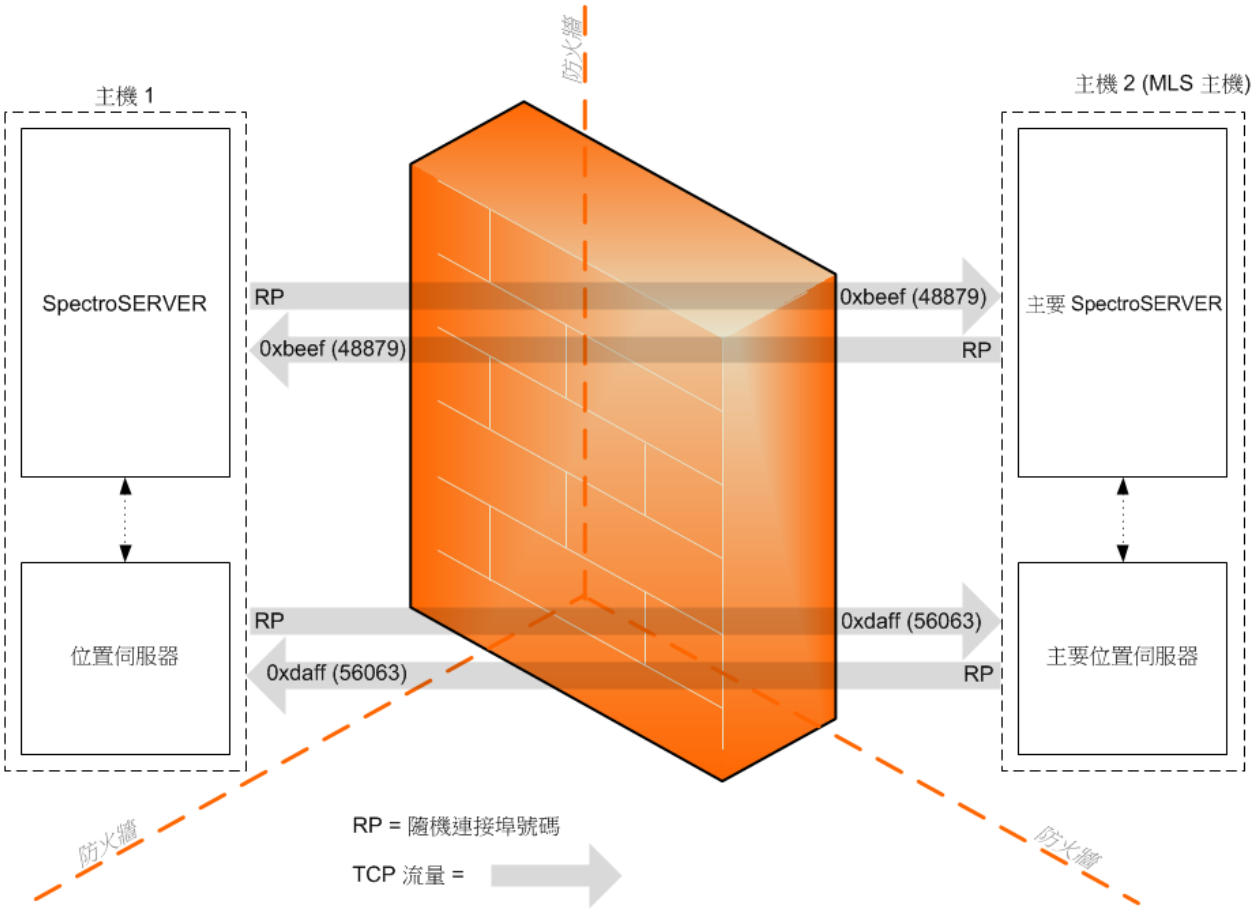
預設的 CORBA OneClick 伺服器接聽連接埠值位於 <*$SPECROOT*>/tomcat/webapps/spectrum/META-INF/context.xml 中。

```
vbroker.se.iiop_tp.scm.iiop_tp.listener.port=<新的連接埠號碼>
```

遠端 SpectroSERVER 和防火牆

下圖說明兩個遠端 SpectroSERVER 透過防火牆通訊時所需的 IP 連線。 使用 TCP 的時候，連線都是從隨機連接埠連至特定的固定連接埠。

重要！ 不是主要位置伺服器 (MLS) 之 SpectroSERVER 間的所有通訊都會透過 MLS-SpectroSERVER 來路由傳送。 因此，每個 SpectroSERVER 只會與 MLS-SpectroSERVER 直接通訊。

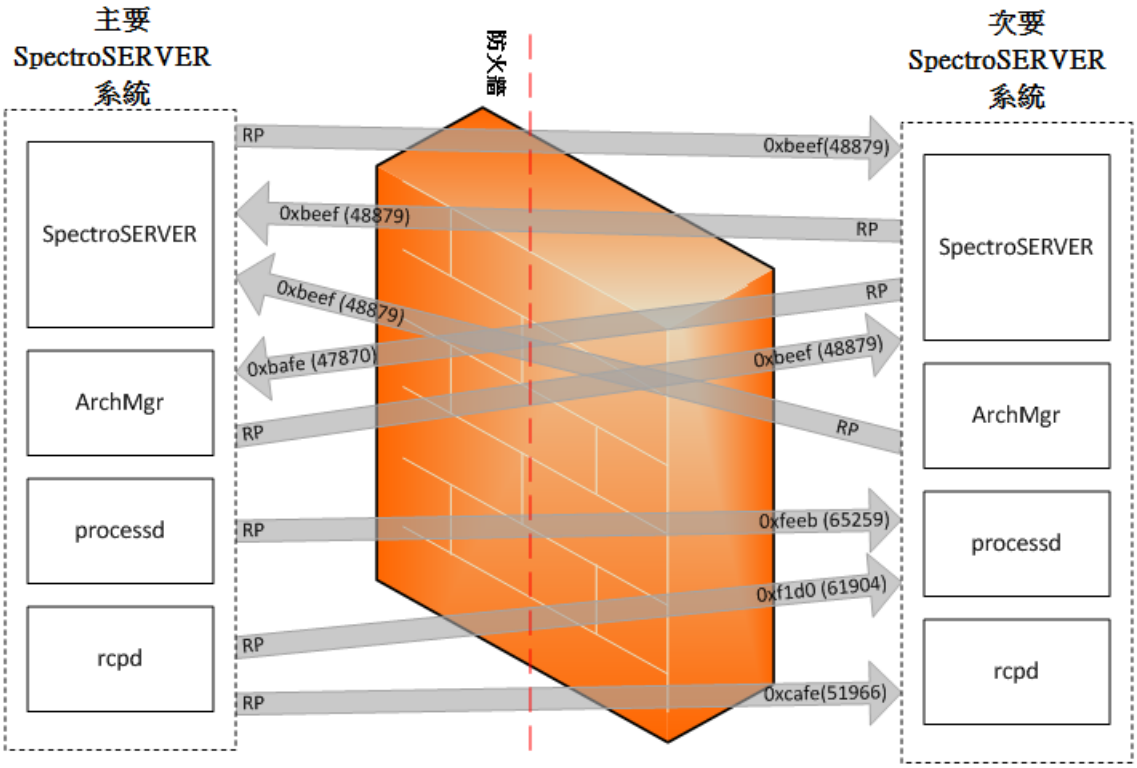


更多資訊：

位置伺服器 (位於 p. 35)

主要和次要 SpectroSERVER 的跨防火牆通訊

下圖說明容錯環境中主要和次要 SpectroSERVER 透過防火牆通訊時所需的 IP 連線。 使用 TCP 的時候，連線都是從隨機連接埠連至特定的固定連接埠。



NAT 防火牆環境的 CA Spectrum 配置檔案

網路位址轉譯 (NAT) 是使用一致的內部 IP 位址和處理 IP 位址轉譯至網際網路的單一節點來建構網路。 CA Spectrum 支援 NAT。 您可以在私人 IP 位址網域中部署 CA Spectrum，並可維護連往 NAT 防火牆外部用戶端的連線。因此，DSS 環境現在可以包含由防火牆分隔的多個網域。

NAT 環境的唯一需求是用戶端需具備依名稱解析伺服器的能力。 在 NAT 的私人端，主機名稱必須解析爲私人端 IP 位址。 在 NAT 的公用端，主機名稱必須解析爲公用端 IP 位址。

預設連接埠配置

您可以變更 CA Spectrum 程序使用的預設連接埠或通訊端號碼，以讓程序在防火牆環境中可以正確運作。

您可以變更下列程序的連接埠號碼和通訊端號碼：

- OneClick WebServer (位於 p. 59)
- SpectroSERVER (位於 p. 62)
- Archive Manager (位於 p. 63)
- 位置伺服器 (位於 p. 63)
- 命名服務 (位於 p. 64)
- 遠端複製程序精靈 (rcpd) (位於 p. 64)
- CLI 精靈 (vnmshd) (位於 p. 64)

附註： 您不能變更的連接埠僅限於遠端管理精靈、sradmin 和經由 SpectroSERVER 的 Telnet 連接埠。

變更 SpectroSERVER 連接埠號碼

您可以變更 SpectroSERVER 用於 CORBA 要求的連接埠號碼。

請使用文字編輯器來編輯 .vnmrc 檔案，以反映新的連接埠號碼。 此檔案位於 *<$SPECROOT>*/SS/。 輸入下列命令：

```
orb_args=-Dvbroker.se.iiop_tp.scm.iiop_tp.listener.port=<新的連接埠號碼>
```

更多資訊：

連接埠衝突解決方法 (位於 p. 32)

變更 Archive Manager 連接埠號碼和通訊端號碼

您可以變更 Archive Manager 用於特定要求的連接埠號碼和通訊端號碼。

請使用文字編輯器來編輯 .configrc 檔案，以反映新的連接埠號碼。 此檔案位於 <*$SPECROOT*>/SS/DDM。 輸入下列命令：

```
orb_args=-Dvbroker.se.iiop_tp.scm.iiop_tp.listener.port=<新的連接埠號碼>
```

若要變更 Archive Manager 用來接聽 VNM 和 SSAPI 用戶端要求的通訊端號碼，請使用文字編輯器來編輯 .configrc 檔案。 此檔案位於 <*$SPECROOT*>/SS/DDM。 變更下列變數：

```
ARCH_MGR_SOCKET_NUMBER=<新的連接埠號碼>
```

更多資訊：

連接埠衝突解決方法 (位於 p. 32)

變更位置伺服器連接埠號碼和通訊端號碼

您可以變更位置伺服器用於特定要求的連接埠號碼和通訊端號碼。

若要變更位置伺服器用於 CORBA 要求的連接埠號碼，請使用文字編輯器來編輯 .locrc 檔案，以反映新的連接埠號碼。 此檔案位於 <*$SPECROOT*>/LS。 輸入下列命令：

```
orb_args=-Dvbroker.se.iiop_tp.scm.iiop_tp.listener.port=<新的連接埠號碼>
```

若要變更位置伺服器用於 VNM 和 SSAPI 要求的通訊端號碼，請使用文字編輯器來編輯 .locrc 檔案。 變更下列變數：

```
LOC_SERVER_SOCKET_NUMBER=<新的連接埠號碼>
```

更多資訊：

連接埠衝突解決方法 (位於 p. 32)

變更 Visibroker 命名服務連接埠號碼

您可以變更 Visibroker 命名服務所用的連接埠號碼。

附註：預設連接埠號碼是 14006。

請依循下列步驟:

1. 以滑鼠右鍵按一下 [我的電腦]，並選取 [內容]。

   [媒體櫃內容] 對話方塊開啓。

2. 按一下 [進階] 索引標籤，並按一下 [環境變數] 按鈕。
3. 選取 NAMING_SERVICE_PORT 並加以編輯，反映新的連接埠號碼：

   NAMING_SERVICE_PORT=<*新的連接埠號碼*>

若要變更 Solaris 上的 Visibroker 命名服務連接埠號碼，請在 spectrum60.env 檔案中設定環境變數。 此檔案位於 /opt/SPECTRUM 目錄。 變更下列變數：

NAMING_SERVICE_PORT=<*新的連接埠號碼*>

遠端複製程序精靈 (rcpd) 連接埠號碼配置

遠端複製程序精靈 (rcpd) 連接埠號碼可透過 rcpd_comm_port .vnmrc 資源進行配置。

更多資訊：

一般 SpectroSERVER (.vnmrc) 資源 (位於 p. 13)
連接埠衝突解決方法 (位於 p. 32)

CLI 精靈 (vnmshd) 連接埠號碼配置

CLI 精靈 (vnmshd) 連接埠號碼可透過 vsh_tcp_port 參數進行配置。

附註：如需 vsh_tcp_port 參數的詳細資訊，請參閱*命令列介面使用者指南*.

更多資訊：

連接埠衝突解決方法 (位於 p. 32)

第 4 章： 容錯

本節包含以下主題：

關於 SpectroSERVER 容錯 (位於 p. 67)
SpectroSERVER 警報同步 (位於 p. 72)
建立容錯 (位於 p. 76)
監控主要和次要 SpectroSERVER 間的轉換 (位於 p. 82)
如何監控次要 SpectroSERVER 狀態 (位於 p. 83)

關於 SpectroSERVER 容錯

容錯需要一個以上的 SpectroSERVER 來管理特定範圍。 每個 SpectroSERVER 上都會載入該範圍的資料庫副本。 不過，隨時都只有一個副本爲使用中。 具有使用中資料庫的 SpectroSERVER 稱爲*主要 SpectroSERVER*。 非使用中資料庫將以待命 SpectroSERVER 的身分執行，也就是次要 SpectroSERVER。 您也可以在第三個 SpectroSERVER 上安裝另一個非使用中資料庫副本。

如果主要 SpectroSERVER 失敗，次要 SpectroSERVER 上的資料庫會變爲使用中，且次要 SpectroSERVER 會開始管理網路。 連線至主要 SpectroSERVER 的應用程式會自動切換至次要 SpectroSERVER。 當主要 SpectroSERVER 恢復服務後，應用程式會自動切換回主要 SpectroSERVER，而次要 SpectroSERVER 再次變爲非使用中狀態。

附註： 從次要 SpectroSERVER 執行時，並非所有應用程式都能充分執行其完整功能。 設定容錯環境的主要原因是要確保持續監控網路，而非建立 CA Spectrum 的完整副本。

更多資訊：

容錯設定中的 Trap Director (位於 p. 88)

容錯環境的 SpectroSERVER 優先順序

管理相同範圍的主要、次要和第三個 SpectroSERVER 必須全部擁有相同的範圍控制代碼和相同的模型化類別。 這些伺服器是以優先順序數值來加以區別。 最低的數字代表主要 SpectroSERVER。 SpectroSERVER 安裝時的預設優先順序值爲 10。 若要將 SpectroSERVER 指定爲次要伺服器，請指派較高的優先順序值給它，例如 20。 同樣的，第三個 SpectroSERVER 具備高於次要的優先順序值，例如 30。

首次設定容錯環境時，可以在使用 SSdbload 公用程式 (位於 p. 76)載入資料庫副本至任何待命 SpectroSERVER 的時候，指派優先順序值。

稍後若想變更優先順序值，可以使用 [已載入範圍] 子檢視。 在 [導覽] 面板中選取本機範圍，然後選取 [元件詳細資料] 面板中的 [資訊] 索引標籤，即可存取此子檢視。

附註：[已載入範圍] 子檢視不同於 [SpectroSERVER 控制] 子檢視。 若要存取 [SpectroSERVER 控制] 子檢視，請在 [導覽] 面板中選取 VNM，然後選取 [元件詳細資料] 面板中的 [資訊] 索引標籤。

更多資訊：

建立容錯 (位於 p. 76)

SpectroSERVER 資料同步

在容錯 CA Spectrum 環境中，任何時候都會有一個資料庫是使用中。 因此，其他資料庫必須定期更新，以反映使用中資料庫的新模型和屬性值變更。 此資料同步是透過 CA Spectrum 的「線上備份」功能來完成。 您可以視需要隨時執行「線上備份」，或以定期排程的間隔執行。 對主要 SpectroSERVER 執行「線上備份」時，會建立目前資料庫的備份副本。 「線上備份」會自動將副本載入至每個指定的次要 SpectroSERVER。

與 DSS 環境相同，容錯環境中的每個 SpectroSERVER 都需安裝相同的模型化類別。「線上備份」會複製目前的模型化類別。 但不會複製所有 .i 檔案或與個別管理模組關聯的其他元素。 因此，如果您在主要 SpectroSERVER 上安裝了任何新的管理模組，也請在次要 SpectroSERVER 上安裝相同的新管理模組。

附註：如需詳細資訊，請參閱《*資料庫管理指南*》。

當次要 SpectroSERVER 輪詢主要 SpectroSERVER 的狀態資訊時，<*$SPECROOT*>/custom/Events 目錄中定義的 EventDisp 和 Alertmap 檔案會傳播至容錯伺服器。

支援容錯 Archive Manager

從 9.4 版 版本開始，您可在容錯 SpectroSERVER 環境中的次要 SpectroSERVER 主機上執行 Archive Manager。 當主要 Archive Manager 關閉時，這個次要 Archive Manager 可讓您檢視 OneClick 中的事件。

在下列兩種情況下，主要或次要 SpectroSERVER 會在本機儲存事件：

- 當主要 Archive Manager 關閉時，主要 SpectroSERVER 便會執行。 在這種情況下，主要 SpectroSERVER 會在事件建立時將其儲存在本機上，直到主要 Archive Manager 恢復運作爲止。
- 當主要 SpectroSERVER 主機本身關閉時。 在這種情況下，次要 SpectroSERVER 會在事件建立時將其儲存在本機上，直到主要 Archive Manager 恢復運作爲止。

在 9.4 版 版本之前，OneClick 伺服器在這兩種情況下無法存取事件。 因此，[OneClick 事件] 檢視中不會出現事件。現在，您可以在次要 SpectroSERVER 主機上啓動次要 Archive Manager，以便能夠檢視主要 Archive Manager 關閉時建立的事件，還可以檢視歷史事件。

當您啓動次要 Archive Manager 時，其可做爲主要 SpectroSERVER 的用戶端，接收並記錄建立的事件。 此行爲並不會影響主要 SpectroSERVER 及主要 Archive Manager 之間的正常連線。 主要 Archive Manager 一旦關閉，OneClick 便會立即容錯移轉至次要 Archive Manager 以提供事件資料。

當主要 SpectroSERVER 主機本身關閉時，次要 SpectroSERVER 會在本機儲存事件，同時也會將事件轉送至次要 Archive Manager。 主要 Archive Manager 恢復運作時，次要 SpectroSERVER 便會將本機儲存的所有事件傳輸給它。

Archive Manager 資料同步

次要 Archive Manager 會以其最大能力同步處理事件，而在主要 Archive Manager 及次要 Archive Manager 之間不會同步處理任何事件。 當次要 Archive Manager 執行並連線至 SpectroSERVER 時，它會收到產生的所有事件的副本。 每當次要 Archive Manager 關閉時，就不會在次要 Archive Manager 上儲存事件。 此功能與主要 Archive Manager 的功能截然不同，後者會由 SpectroSERVER 儲存事件，稍後再傳輸給主要 Archive Manager。

這代表次要 Archive Manager 首次啟動時，其 DDM 資料庫不會包含任何事件，而且不會嘗試與主要 Archive Manager 進行同步處理。 一般而言，當次要 Archive Manager 已執行 .configrc 中配置的天數 (MAX_EVENT_DAYS) 時，便會與主要 Archive Manager 資料庫進行同步處理。

次要 SpectroSERVER 未重新啟動時產生警報

主要 SpectroSERVER 與次要 SpectroSERVER 同步其資料庫時，會產生「失去對次要伺服器的連絡」(0x00010c0e) 事件和警報。 次要 SpectroSERVER 已下線，以便從主要 SpectroSERVER 載入新資料庫。

您可以設定規則來處理這個警報，只在次要 SpectroSERVER 未重新啟動時才產生警報。

EventPair 規則可讓您指定如果發生「失去對次要伺服器的連絡」事件且在指定的期間內未接著發生「已建立對次要伺服器的連絡」(0x00010c0f) 事件時，將產生新事件。 接著可以指定此新事件建立事件和警報，指出次要 SpectroSERVER 仍舊停機。

請依循下列步驟:

1. 使用文字編輯器開啟 EventDisp 檔案。

   附註：EventDisp 檔案位於 *<$SPECROOT>*/SS/CsVendor/Cabletron 目錄中。

2. 找到內容為 0x00010c0e E 50 A 2, 0x00010c0e 的行，並將此行變更如下：

   ```
   0x00010c0e R Aprisma.EventPair, 0x00010c0f,
       <numberofsecondstowait><generatedeventcode>
   ```

   <generatedeventcode>

   此為當次要 SpectroSERVER 未在 *<numberofsecondstowait>* 所指定時間內返回線上時，將產生的事件代碼。

3. 新增下面這行至 EventDisp 檔案：

   `<generatedeventcode>E 50 A 2, <generatedalarmcode>`

   <generatedeventcode>

   如果次要 SpectroSERVER 未上線，此為步驟 2 產生的事件代碼。 'E 50' 指出會記錄事件，且嚴重性值為 50。 A 2 指出會建立主要警報。 *<generatedalarmcode>* 是根據此事件將產生的警報代碼。

4. 為此警報建立「可能原因」檔案，指出在同步資料後，未重新建立與次要 SpectroSERVER 的連絡。

附註： 如需詳細資訊，請參閱《*事件配置使用者指南*》。

次要 SpectroSERVER 整備層級

次要 SpectroSERVER 可能位於三個整備層級的其中一個。 整備情形依伺服器配置和狀態而異。 整備層級的定義如下：

熱

次要 SpectroSERVER 執行中且可在主要 SpectroSERVER 失敗時立即接管，因為它已經在進行輪詢。 若要將次要 SpectroSERVER 配置為這種整備層級，請將下面這行加入 .vnmrc 檔案：secondary_polling=yes。 此陳述式會讓待命開始輪詢並在啓動後處理設陷，無論其與主要 SpectroSERVER 的連線狀態為何。

溫

次要 SpectroSERVER 執行中，但伺服器進入完全可用狀態還需要一點時間。 次要 SpectroSERVER 尚未配置為啓動輪詢，*除非*遺失與主要 SpectroSERVER 的連線。例如，其 .vnmrc 檔案中沒有 secondary_polling 項目，或是該項目設為 no。

如果 .vnmrc 檔案中沒有 secondary_polling 項目，或是項目設為 no，次要 SpectroSERVER 位於待命模式時就不會處理設陷。

冷

次要 SpectroSERVER 未執行，當主要 SpectroSERVER 發生失敗時必須將其啓動。 在這種情況下，次要 SpectroSERVER 是否配置為次要輪詢並不重要。

更多資訊：

建立容錯 (位於 p. 76)
驗證容錯配置 (位於 p. 78)

SpectroSERVER 警報同步

主要和次要 SpectroSERVER 使用「全域警報服務」(GAS) 連線來共用警報資訊。 SpectroSERVER 使用警報資訊來同步警報。 此項同步可防止產生重複警報。

容錯警報同步的選項包括啓用服務和偵錯記錄。 這些選項由 .vnmrc 檔案中的設定值控制。 如需詳細資訊，請參閱＜容錯警報服務 (.vnmrc) 資源 (位於 p. 22)＞。

當您部署 VHM 與配置了次要 SpectroSERVER 的機箱裝置時，警報同步處理功能會出現某些非預期的行爲。 當主要 SpectroSERVER 關閉時，在主要 SpectroSERVER 上相關聯的警報，並不會在次要 SpectroSERVER 上有相同的關聯。反之，每個不同的狀況或徵兆都會產生一個警報。當主要 SpectroSERVER 恢復連線時，警報關聯隨即正常運作。 但是在機箱裝置上仍無法執行關聯。

之所以無法保留關聯，是因爲在次要 SpectroSERVER 上產生的警報不是從事件建立的。

警報同步的程序視同步是從主要 SpectroSERVER 往次要 SpectroSERVER 同步，或是從次要往主要同步而異。 以下章節說明每一個案例：

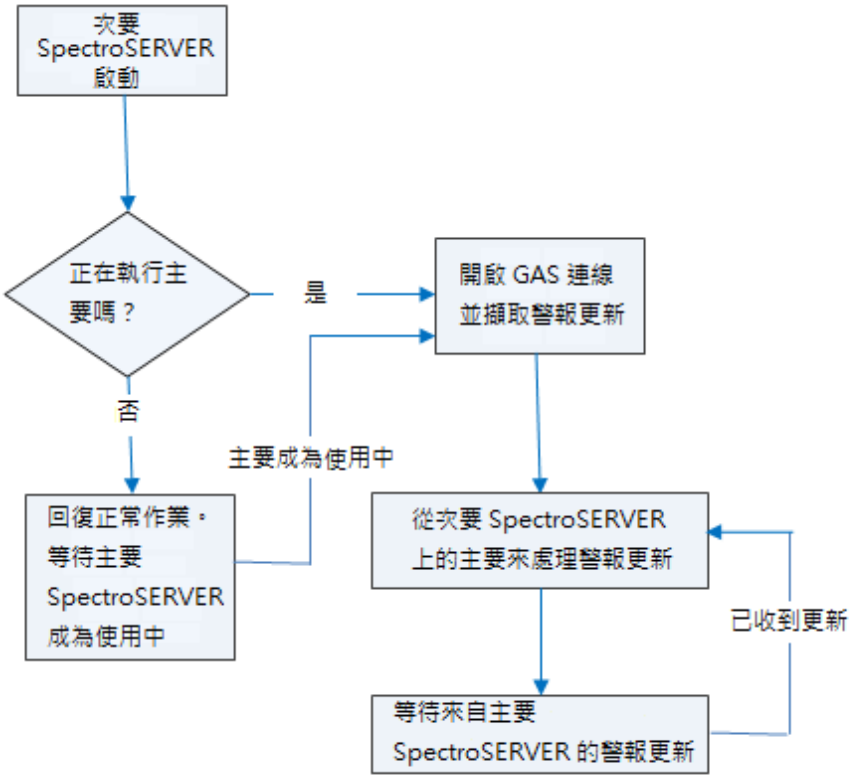
- 從主要往次要 SpectroSERVER 同步 (位於 p. 72)
- 從次要往主要 SpectroSERVER 同步 (位於 p. 74)

從主要往次要 SpectroSERVER 同步

如果 SpectroSERVER 執行爲次要 SpectroSERVER，則會嘗試開啓連往主要 SpectroSERVER 的 GAS 連線。 如果連線成功，次要 SpectroSERVER 會登錄以接收主要 SpectroSERVER 的警報更新。 次要伺服器繼續連線至主要伺服器，並繼續接收主要伺服器上發生的警報更新。 否則，次要 SpectroSERVER 會每分鐘嘗試連線一次，直到主要 SpectroSERVER 成爲使用中爲止。

附註：主要和次要伺服器上都必須啓用「容錯警報服務」，次要 SpectroSERVER 才能嘗試連線至主要 SpectroSERVER 進行警報同步。 若要控制此功能，請在 .vnmrc 檔案中使用 ftasv_enabled 參數。 如需詳細資訊，請參閱< 容錯警報服務 (.vnmrc) 資源 (位於 p. 22) >。

下圖說明次要 SpectroSERVER 如何處理來自主要 SpectroSERVER 的警報更新：



- 新的主要 SpectroSERVER 警報新增至次要 SpectroSERVER，並標示爲 'stale'。 新警報取代現有警報 (如果兩者相等)。 您可以透過驗證下列參數，判斷警報是否相等：
  - 唯一警報 (ID)
  - 模型控制代碼
  - 可能原因
  - 警報鑑別子

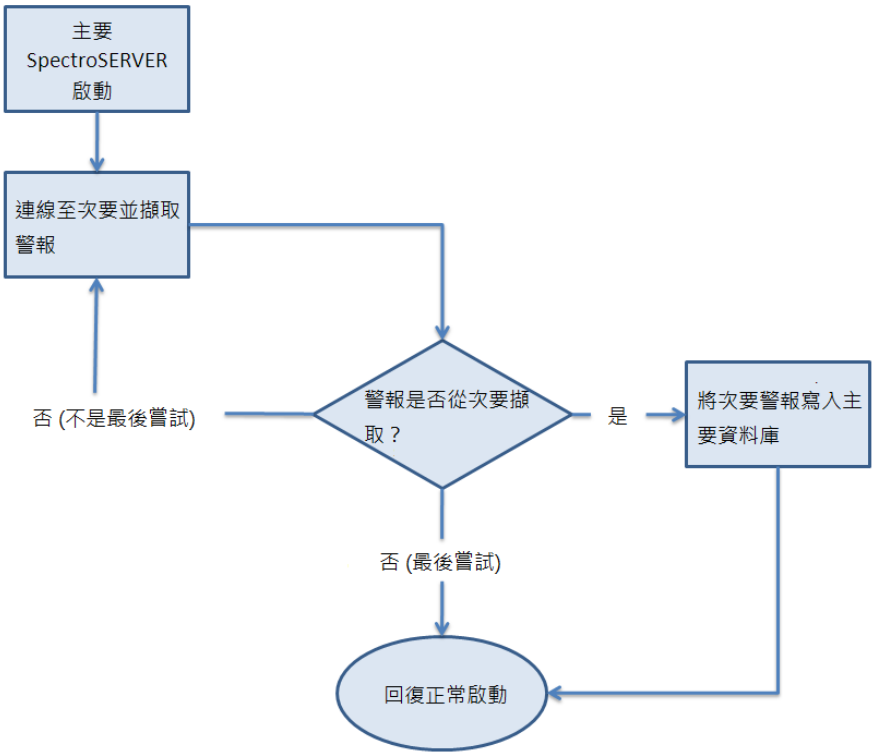
  如果兩個警報有相同 ID，便視爲相等。 相同模型上的警報 (亦即，它們具有相同的模型控制代碼) 以及具有相同可能原因的警報也會視爲相等，除非它們有不同的警報鑑別子。
- 來自主要 SpectroSERVER 的警報屬性更新會套用至次要 SpectroSERVER 上的相同警報。 如果主要 SpectroSERVER 上的警報已清除，則次要 SpectroSERVER 上的警報也會清除。
- 維護模式警報同步的 isManaged 和 isNotHibernating 屬性會更新。

從次要往主要 SpectroSERVER 同步

當主要 SpectroSERVER 在失敗或計劃維護後再次上線，它會連線至次要 SpectroSERVER 並取得所有可用警報。 如果首次連線嘗試失敗，主要伺服器會嘗試連線多次以便同步。 主要 SpectroSERVER 之 .vnmrc 檔案中的 ftasv_max_conn_retry_count 和 ftasv_conn_retry_interval 參數負責控制嘗試次數和頻率。 如需詳細資訊，請參閱< 容錯警報服務 (.vnmrc) 資源 (位於 p. 22) >。

重要！ 如果當主要伺服器嘗試連線至未執行的次要 SpectroSERVER 時，主要伺服器會耗盡其同步嘗試次數。 結果會使啓動主要伺服器發生延遲。 這種延遲是無法避免的，因爲它發生在模型啓動之前。 若想避免這種狀況，請在主要伺服器啓動時確認次要伺服器正在執行。 您也可以減少重試次數或間隔大小，以降低可能的延遲。 或者，您可以使用 .vnmrc 檔案中的 ftasv_enabled 參數來停用「容錯警報服務」。

下圖說明主要 SpectroSERVER 如何開啓連往次要 SpectroSERVER 的 GAS 連線，並擷取其所有的警報。



連線成功後：

- 擷取到警報清單後，主要 SpectroSERVER 會將警報寫入其資料庫。 主要 SpectroSERVER 會繼續啓動程序，並從資料庫讀取這些警報。
- 擷取到次要 SpectroSERVER 的警報後，這些警報會取代主要資料庫中儲存的警報。
- 從資料庫讀取的所有警報都會進行處理，並在正確的模型上判斷提示。
- 當主要 SpectroSERVER 停機時在次要 SpectroSERVER 上產生的每個新警報，都會產生次要「警報設定」事件 (0x10714)。
- 當主要 SpectroSERVER 停機時在次要 SpectroSERVER 上清除的每個警報，則會產生次要警報清除事件 (0x10715)。

附註：藍色和灰色警報不會進行同步。 棕色警報會進行同步，但 WA_Link 模型不會。 WA_Link 模型例外狀況僅套用至棕色警報。

同步失敗後：

- 如果當主要 SpectroSERVER 嘗試連線至次要時警報同步失敗，主要仍會啓動，但會在主要 LocalLscpe 模型中產生事件和重大警報 (0x10c35)。 事件和警報包含每個嘗試失敗的原因。 若要在警報同步失敗後繼續執行主要 SpectroSERVER，請執行「線上備份」以便同步主要和次要 SpectroSERVER 資料庫。

建立容錯

您可以在首次安裝 CA Spectrum 時、尚未建立任何模型前，設定容錯環境。也可以在安裝 CA Spectrum 後再設定容錯環境。

下列程序說明如何設定兩個 SpectroSERVER：主要和次要。 您也可採取相同步驟，設定第三個 SpectroSERVER。 不過，請指派比次要 SpectroSERVER 高的優先順序數字給第三個 SpectroSERVER。

附註： 若要在使用 Southbound Gateway 整合的環境中建立容錯，請參閱《*Southbound Gateway 工具組指南*》。

請依循下列步驟:

1. 在主要 SpectroSERVER 和次要 SpectroSERVER 上安裝相同版本的 CA Spectrum 以及相同的模型化類別。 每部伺服器需要相同的範圍控制代碼。
2. 驗證主要和次要 SpectroSERVER 的 .hostrc 檔案中有項目，可提供 SpectroSERVER 相互存取權限。

   附註：如果您在主要 SpectroSERVER 的 .hostrc 檔案中指定次要 SpectroSERVER 的安全使用者，且次要 SpectroSERVER 在 Windows 環境中執行，則請將使用者 SYSTEM 加入安全使用者清單中。
3. 確認次要 SpectroSERVER 伺服器上 .locrc 檔案中的 MAIN_LOCATION_HOST_NAME 參數，指向主要 SpectroSERVER 上 .locrc 檔案中的同一個系統名稱。 否則同步會失敗。
4. 配置主要和次要 SpectroSERVER，讓執行每個 SpectroSERVER 的使用者都相同。 如果使用者不同，次要 SpectroSERVER 會失敗或在「線上備份」後將無法正常運行。

5. 執行「線上備份」，製作主要 SpectroSERVER 資料庫的副本。 如果 SpectroSERVER 關機，則可使用 SSdbsave 公用程式加上 -cm 引數 (以便儲存模型化類別和所有新模型)。

   附註：如需詳細資訊，請參閱《*資料庫管理指南*》。

6. 確認您建立的儲存檔案可供主控次要 SpectroSERVER 的伺服器使用。 如有需要，將檔案複製至伺服器。

7. 在次要伺服器上，於 SpectroSERVER 關機時，導覽至 CA Spectrum SS 目錄並使用下列命令載入儲存檔案：

   `../SS-Tools/SSdbload -il -add` *`precedence savefile`*

   precedence

   指定大於主要伺服器預設值 10 的數值 (建議使用 20)。

   savefile

   指定先前建立之儲存檔案的名稱。

8. (選擇性) 新增一行 'secondary_polling=yes' 至 .vnmrc 檔案，讓次要 SpectroSERVER 作爲「熱」備份 (位於 p. 71)。

9. 如果主要 SpectroSERVER 尚未執行，則將其啓動。

10. 啓動次要 SpectroSERVER。

11. 若要確認設定，請使用 MapUpdate 命令以及 view 引數來顯示目前的範圍對應。

    附註：如需詳細資訊，請參閱《*資料庫管理指南*》。

    次要 SpectroSERVER 現在可在主要 SpectroSERVER 失敗時自動接管。如果您先前已啓動次要輪詢，則次要 SpectroSERVER 立即可供使用。 否則，會在伺服器偵測到遺失與主要 SpectroSERVER 的連線時開始輪詢。

服務從主要 SpectroSERVER 切換至次要 SpectroSERVER 時，[連線狀態] 圖示 會顯示爲黃色。 若要檢視一個範圍內所有伺服器的連線狀態，請按一下 [連線狀態] 圖示。 在 [連線狀態] 對話方塊中，範圍中每部伺服器的 [連線狀態] 圖示會顯示爲黃色，指出「已切換」狀態。

當主要 SpectroSERVER 恢復上線，次要 SpectroSERVER 就會停止輪詢 (除非您已將 secondary_polling 設爲 'yes')。 所有應用程式都會切換回主要 SpectroSERVER。 不過，您在次要 SpectroSERVER 使用中時對其進行的任何編輯，並*不會*自動複寫至主要 SpectroSERVER。 請在主要 SpectroSERVER 上手動重新建立這些修改。

重新啟動主要 SpectroSERVER 時，所有模型都載入後就會接受連線 (但在所有模型都啟動*前*)。 模型啟動需要花點時間。 因爲次要 SpectroSERVER 在主要 SpectroSERVER 重新啟動後就會停止輪詢，因此您的網路管理涵蓋範圍會出現一個缺口。

若要避免這種情形，請編輯主要 SpectroSERVER 上的 .vnmrc 檔案，將 wait_active 資源設爲 'yes'。 此參數會讓伺服器等候所有模型都啟動後，才開始接受連線。 CA Spectrum 控制台的訊息區域也會動態顯示模型的啟動百分比。 SpectroSERVER 看似需要更久時間才能顯示。 不過，當所有模型都啟動後，SpectroSERVER 就已備妥，可以管理網路。

您也可以將次要 SpectroSERVER 上的 wait_active 資源設爲 'yes'。 在主要 SpectroSERVER 的計劃停機期間內，您接著可以在 CA Spectrum 控制台中驗證次要 SpectroSERVER 已準備好接管。

附註：如需詳細資訊，請參閱《*資料庫管理指南*》。

更多資訊：

關於分散式 SpectroSERVER (位於 p. 9)
容錯環境的 SpectroSERVER 優先順序 (位於 p. 68)
指定新的主要位置伺服器 (位於 p. 37)

驗證容錯配置

在分散式 SpectroSERVER 部署中設定好容錯後，請確認 OneClick 伺服器可以存取主要和次要 SpectroSERVER。 如果無法同時連線至這兩部伺服器，OneClick 伺服器就無法容錯移轉至次要 SpectroSERVER。

請依循下列步驟:

1. 存取 [OneClick 管理] > [範圍] 網頁。
2. 檢查 [次要狀態] 欄。 驗證 OneClick 已建立與次要 SpectroSERVER 的連絡。

   狀態也會指出「容錯」是否已備妥可進行容錯移轉。

   容錯配置已通過驗證。

測試容錯

在初始安裝期間，次要 SpectroSERVER 可能無法存取主要 SpectroSERVER 能夠存取的所有裝置。 這種情形會導致次要 SpectroSERVER 產生假警報。 若要避免假警報，請測試容錯，確認次要 SpectroSERVER 可以管理您的網路裝置。

附註： 每次將新裝置新增至主要 SpectroSERVER 時，都要測試容錯。

請依循下列步驟:

1. 當主要和次要 SpectroSERVER 都已開機並執行中，將主要 SpectroSERVER 下線。

   [連線狀態] 圖示 呈現黃色，指出「已切換」狀態。

   紅色連接器則指出 OneClick 伺服器無法連絡次要 SpectroSERVER。

2. 請等候 15 - 20 分鐘，讓次要 SpectroSERVER 執行。
3. 驗證下列狀態：
   - [連線狀態] 圖示未顯示爲紅色。
   - 所有裝置模型和可 Ping 模型都維持 SNMP 或 ICMP 連絡。

     如果失去連絡，請驗證次要 SpectroSERVER 可以存取您的裝置。 洽詢網路管理員，請他解決問題 (若適用)。
   - CA Spectrum 會管理所有具已建立連絡狀態的裝置。 檢查任何裝置模型的裝置連絡或管理連絡遺失警報，以驗證狀態。
4. 重新啟動主要 SpectroSERVER。

   [連線狀態] 圖示顯示綠色，代表連絡狀態正常。

容錯復原

可能的失敗情況有以下兩種：

- 主要 SpectroSERVER 停止。次要 SpectroSERVER 接著會轉送事件和統計資料資訊給主要 Archive Manager (主要 Archive Manager 在主控主要 SpectroSERVER 的伺服器上執行)。 當主要 SpectroSERVER 重新啟動時，不會遺失任何事件或統計資料。
- 主要 SpectroSERVER 和主要 Archive Manager 執行所在的電腦會完全停止操作。接下來次要 SpectroSERVER 會在其資料庫中快取事件和統計資料，直到主要 SpectroSERVER 電腦恢復上線爲止。 如果次要 Archive Manager 在執行，OneClick 中就會有歷史和即時資訊，但此時仍會快取這些資訊，以便傳輸給主要 Archive Manager。

重新啟動主要 Archive Manager 和主要 SpectroSERVER (如果其伺服器無法使用或主要 SpectroSERVER 停止操作)。

附註：啟動 SpectroSERVER 前，不再需要先啟動 Archive Manager，從次要 SpectroSERVER 快取的事件可隨時傳輸，即使是在主要 SpectroSERVER 已開始記錄新事件之後。

請依循下列步驟:

1. 在主要 SpectroSERVER 主機上啟動 SPECTRUM 控制面板。
2. 若要啟動 SpectroSERVER，按一下 SPECTRUM 控制面板上的 [啟動 SpectroSERVER]。

   當主要 Archive Manager 再次可供操作，次要 SpectroSERVER 便會連線至主要 Archive Manager 並將它快取的事件資料傳輸過去。

變更主要和次要 SpectroSERVER 的主機名稱

容錯環境中的 SpectroSERVER 使用與其主機名稱關聯的優先順序值，來辨別它和其他伺服器的關係。 因此，若要保留容錯關係，請使用 SSdbsave 和 SSdbload 來變更主要 SpectroSERVER 的主機名稱。

請依循下列步驟:

1. 使用 SSdbsave 和 -cm 選項來儲存資料庫。
2. 變更主機名稱。

3. 使用您在第一個步驟建立的儲存檔案，重新載入資料庫。 執行 SSdbload (使用 -il 選項和 -replace 選項)：

   ```
   SSdbload -il -replace precedence savefile
   ```

   此命令會讓資料庫建立新主機名稱與優先順序值 (10) 的關聯性，此優先順序值可指定主要 SpectroSERVER。

   下次執行「線上備份」並因此同步資料庫時，就會將主機名稱的變更告知所有溫或熱待命 SpectroSERVER。

   但在這期間，主機名稱的變更會讓待命 SpectroSERVER 無法偵測主要 SpectroSERVER 是否正在執行中。 結果，配置爲溫待命的任何 SpectroSERVER 就會開始輪詢。

4. 使用 SSdbload 加上 -il 和 -replace 選項，載入儲存檔案至溫待命伺服器，並指定較高的優先順序值 (例如 20)，將其指定爲待命。

現在，您可以變更次要 SpectroSERVER 的主機名稱。

請依循下列步驟:

1. 使用 SSdbsave 和 -cm 選項來儲存資料庫。
2. 變更主機名稱。
3. 使用您在第一個步驟建立的儲存檔案，重新載入資料庫。 執行 SSdbload (使用 -il 選項和 -replace 選項)：

   ```
   SSdbload -il -replace precedence savefile
   ```

   此命令會讓資料庫建立新主機名稱與優先順序值 (20) 的關聯性，此優先順序值可指定次要 SpectroSERVER。

   重新啟動次要 SpectroSERVER 時，伺服器會將新主機名稱及優先順序提供給主要 SpectroSERVER。

附註：如需詳細資訊，請參閱《*資料庫管理指南*》。

監控主要和次要 SpectroSERVER 間的轉換

您可使用監看來監控容錯環境的狀態。 建立監看，以便在主要或次要 SpectroSERVER 準備好可以接管時提醒您。

請依循下列步驟:

1. 在 VNM 模型上建立監看，監控 PercentInitialized (0x11da6) 屬性。

   當此屬性的值等於百分之 100，就表示 SpectroSERVER 已經初始化，且已準備好可以接管。

2. 設定監看，在下列運算式結果等於 TRUE 時，產生事件或警報，或執行指令檔：

   ```
   PercentInitialized == 100
   ```

3. 將監看設定爲使用中的輪詢監看。

4. 同步次要 SpectroSERVER 和主要 SpectroSERVER，散佈此監看。

5. 在監看運算式中指定 Model_Name (0x1006e) 屬性的值。 此屬性在次要 SpectroSERVER 已準備好可以接管時，才會通知您。

   例如，如果下列監看運算式結果爲 TRUE，次要 SpectroSERVER *<sec_server>* 即已準備好可以接管。

   ```
   (PercentInitalized == 100) & (Model_Name= <sec_server>)
   ```

   附註：如需詳細資訊，請參閱《*監看使用者指南*》。

6. 新增下面這行至次要 SpectroSERVER 上的 .vnmrc 檔案，限制潛在的假事件或警報：

   ```
   is_secondary = TRUE
   ```

   此設定會讓次要 SpectroSERVER 捨棄事件，除非 CA Spectrum 判斷次要 SpectroSERVER 已接管成爲主要 SpectroSERVER。

如何監控次要 SpectroSERVER 狀態

您可以建立監看，以便在次要 SpectroSERVER 伺服器以主要 SpectroSERVER 身分執行時提醒您。

請依循下列步驟:

1. 在 VNM 模型上建立監看，以便用來監控次要 SpectroSERVER 之 PausePolling 屬性 (0x11b63) 的值。

   當次要 SpectroSERVER 上的這個屬性設為 FALSE，表示次要 SpectroSERVER 正在輪詢且以主要 SpectroSERVER 的身分執行。

   例如，如果下列監看運算式結果為 TRUE，即表示次要 SpectroSERVER *<sec_server>* 以主要 SpectroSERVER 的身分執行。

   ```
   !PausePolling & (Model_Name == <sec_server>)
   ```

2. 將監看設定為使用中的輪詢監看。
3. 同步次要 SpectroSERVER 和主要 SpectroSERVER，散佈此監看。

   附註：如需詳細資訊，請參閱《*監看使用者指南*》。

4. 新增下面這行至次要 SpectroSERVER 上的 .vnmrc 檔案，限制潛在的假事件或警報：

   ```
   is_secondary = TRUE
   ```

   此設定會讓次要 SpectroSERVER 捨棄事件，除非 CA Spectrum 判斷次要 SpectroSERVER 已接管成為主要 SpectroSERVER。

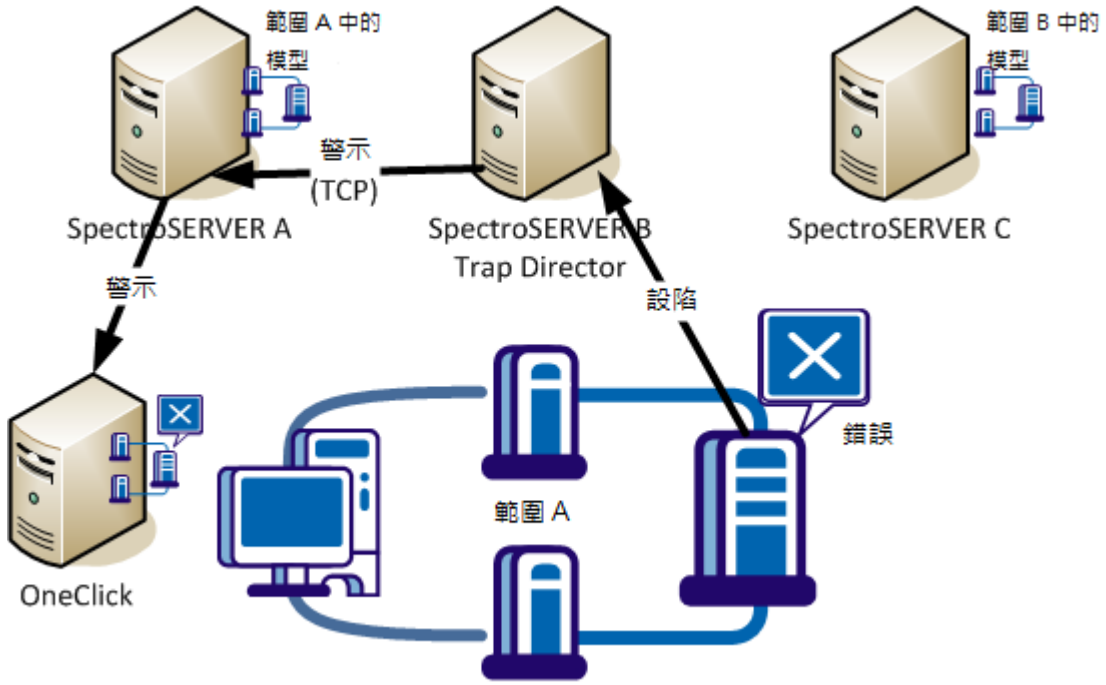
第 5 章： 使用 Trap Director

本節包含以下主題：

Trap Director (位於 p. 85)
設陷與記憶體使用量 (位於 p. 86)
設陷資料流量整合 (位於 p. 86)

Trap Director

Trap Director 是可在分散式 SpectroSERVER 環境中啓用的 SpectroSERVER 功能。「設陷指示器」會在「設陷指示器」主機伺服器上接收設陷，並加以處理。 接著，「設陷指示器」會使用設陷中的資訊建立警示，再透過 TCP 連線將警示轉送至 SpectroSERVER。 因此，主要的「設陷指示器」機制是由 SpectroSERVER 傳送至遠端主機上的模型以進行進一步處理的 CA Spectrum 內部警示。 下圖說明「設陷指示器」的架構：



若要使用「設陷指示器」，請將一個 SpectroSERVER 指定爲「設陷指示器」伺服器。 在該伺服器上啓用設陷指示器 (位於 p. 88)。 接著，請將該伺服器配置爲在遠端範圍中模型化的裝置所傳入之設陷的網路管理站 (NMS) 收件者。

Trap Director 支援加密的 SNMPv3 設陷。 CA Spectrum 會嘗試使用您的 SNMPv3 設定檔來解密傳入的加密 SNMPv3 設陷。

可用的 SpectroSERVER SNMPv3 設定檔將視同不明 SNMPv3 設陷的憑證範本。 收到加密的 SNMPv3 設陷時，會使用 SNMPv3 設定檔來解密設陷。

附註： 如果 Trap Director 範圍具有 SNMPv3 設定檔且憑證有效，執行爲設陷轉寄站的 SpectroSERVER 便可以處理 SNMPv3 設陷。

設陷與記憶體使用量

在啓用「設陷指示器」時，請監控 SpectroSERVER 程序的記憶體使用量。 如果有大量設陷 (每秒超過 100 個) 轉送至多個遠端範圍，就可能發生 SpectroSERVER 無法轉送所有傳入設陷的狀況。 這些設陷會排入佇列中以進行進一步處理。 如果此狀況持續下去，佇列所使用的記憶體可能會耗盡要供 SpectroSERVER 程序使用的記憶體資源。 如此，SpectroSERVER 將會因爲可用記憶體不足而意外關閉。

除了程序記憶體使用量以外，請密切監控 VNM 模型中的下列屬性：

`alert_remote_fwd_queue_length` (屬性 ID：`0x130c3`)

一般而言，如果佇列大小持續增長而超過 1000 個元素，不建議您在環境中啓用「設陷指示器」。

設陷資料流量整合

當網路增長而促使您分配模型給其他範圍時，您可以保留原始範圍主機，作爲這些模型的設陷目的地。 如此就不用在關聯的裝置上重新配置新的設陷目的地。 相反地，您可以整合多個設陷目的地並指定單一 SpectroSERVER 來接收和路由設陷。 如果負載共享是個重要因素，您可在多個伺服器上啓用 Trap Director。 每個伺服器處理一組配置爲將設陷傳送給該伺服器之裝置的設陷。

Trap Director 會維護最新的模型位址快取。 此快取可讓 Trap Director 將設陷來源與模型相比對。 Trap Director 使用比對來轉送設陷給遠端範圍上的目的地模型。 透過維持快取資訊在最新狀態，Trap Director 可以確保 CA Spectrum 能產生模型的事件，而不用考慮其位置。

附註： 將新範圍加入分散式環境時，會影響 Trap Director 效能。 要處理許多新模型的大量設陷時，也會影響效能。 這種情況下會增加設陷通知的延遲時間。

Trap Director 如何更新位址快取

Trap Director 會在分散式環境中維護一個 IP 位址和模型位置的快取。 位址快取可作爲索引，用來判斷應將警示轉送至何處。 模型則可經常新增至範圍、從範圍移除或在範圍間移動。 因此 Trap Director 也會經常更新快取，保持其內容在最新狀態。 符合您可指定之保留期限 (過期) 臨界值的記錄，會予以移除。 Trap Director 也會針對快取中不可用的模型 IP 位址執行跨範圍搜尋，如果它收到來自這些 IP 位址的設陷的話。

下列步驟說明 Trap Director 如何判斷設陷的目的地模型：

1. Trap Director 伺服器收到設陷。
2. 「設陷指示器」將設陷中包含的 IP 位址與快取記錄中的 IP 位址進行比較。
3. 如果「設陷指示器」找到相符項目，便會轉送警示給遠端範圍上的模型。

   否則，「設陷指示器」會執行下列作業，以判斷目的地模型並轉送警示：

   - Trap Director 搜尋已知範圍中是否有相符的 IP 位址。
   - 如果在遠端範圍上找到與 IP 位址關聯的目的地模型，Trap Director 會使用模型資訊來更新快取。 它會轉送警示給模型。
4. 如果跨範圍搜尋找不到相符位址因而捨棄設陷，CA Spectrum 會在「設陷指示器」伺服器的 VNM 模型上產生事件。 事件會指出找不到目的地模型。

   舉例來說，如果模型已刪除，便會找不到相符項目。

容錯設定中的 Trap Director

如果您想實作容錯 Trap Director 設定，則需將裝置配置爲同時轉送設陷給主要和次要 SpectroSERVER。 次要伺服器在偵測到主要伺服器失敗時，才會路由傳送設陷。 次要 SpectroSERVER 會在主要備份或同步處理期間收到 Trap Director 設定。

更多資訊：

關於 SpectroSERVER 容錯 (位於 p. 67)

設陷風暴設定

爲了處理設陷風暴，Trap Director 使用模型化裝置上配置的設陷風暴設定。「設陷指示器」在偵測到設陷風暴時，會停止轉送警示給遠端範圍上的模型。Trap Director 也會判斷提示這些模型的設陷風暴警報。

設陷風暴可能發源於 CA Spectrum 中未模型化的裝置。 對於這些設陷風暴，Trap Director 使用 Trap Director 伺服器上爲 VNM 模型配置的設陷風暴處理設定。

啓用和停用 Trap Director

您可以啓用和停用 SpectroSERVER 上的 Trap Director。

附註：請使用 OneClick 屬性編輯器或 CA Spectrum 命令列介面來設定伺服器 VNM 模型的屬性值。

請依循下列步驟:

1. 展開 VNM 模型 [資訊] 索引標籤中的 [設陷管理] 子檢視。
2. 按一下 [啓用設陷指示器] 欄位中的 [設定]，然後選取 [已啓用]。

   Trap Director 便已啓用。
3. (選用) 若要停用 [設陷指示器]，請按一下 [啓用設陷指示器] 欄位中的 [設定]，然後選取 [已停用]。

   [設陷指示器] 隨即停用。

定義快取記錄保留期限

您可以控制 Trap Director 設陷快取過期的頻率。您可以定義快取記錄保留期限。

若要定義快取記錄保留期限，請定義下列屬性的保留期限：

`trap_cache_age_out_minutes (0x12ad5)`

預設值：180 (分鐘)