

CA Mediation Manager 與 CA Mediation Manager for Infrastructure Management

管理指南

CA Mediation Manager 2.2.3 版/CA Mediation Manager for
Infrastructure Management 2.0 2.2.3 版



本文件包含內嵌說明系統與文件 (以下稱為「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」為 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，若您為「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」的製造商為 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 © 2013 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為相關公司所有。

CA Technologies 產品參考資料

本文件提及下列 CA Technologies 產品：

- CA Mediation Manager
- CA Mediation Manager for Infrastructure Management 2.0

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：簡介	7
架構.....	8
第 2 章：總覽	9
元件總覽.....	9
主要元件.....	9
MultiController.....	10
LocalController	10
Web	11
子元件.....	16
Engine.....	16
Presenter.....	16
其他元件.....	16
一般執行程式.....	17
傳遞服務.....	17
第 3 章：安裝、解除安裝及升級 CA Mediation Manager	19
系統需求.....	19
安裝與升級.....	20
啟動及停止服務.....	20
UNIX	20
Windows	21
第 4 章：元件配置	23
一般執行程式配置.....	24
一般執行程式的運作方式.....	25
一般執行程式配置選項.....	27
一般執行程式啟動順序.....	28
新增其他一般執行程式 (UNIX)	28
MultiController 配置.....	29
MultiController 配置選項.....	32
手動啟動及停止 MultiController	35
LocalController 配置	36

LocalController 配置選項	38
手動啟動及停止 LocalController	42
Engine 與 Presenter 的配置	42
容錯移轉作業	42
MultiController 失敗	42
MultiController 通訊	43
主要 MultiController 失敗	44
MultiController 代理功能	44
LocalController 失敗	45
子元件失敗	45
高可用性配置	46
配置主要 MultiController	46
配置次要 MultiController	46
配置 LocalController	47
記錄檔配置	47
logging.properties 檔案 - 依元件提供的範例	48
配置記錄檔清理	49

第 5 章：將 EMS 設定檔用於 CA Mediation Manager for Infrastructure Management 2.0 (2.2.3 版) 55

EMS 整合設定檔	55
新增 EMS 整合設定檔	55
手動啟動 EMS 搜索	57
檢視 EMS 搜索結果	58
啟動或停止 EMS 搜索服務	58
新增事件規則	59

第 1 章：簡介

本指南提供關於安裝 CA Mediation Manager 的架構、安裝、先決條件與需求的更多資訊。

附註：此章中的資訊僅適用於 CA Mediation Manager。

本節包含以下主題：

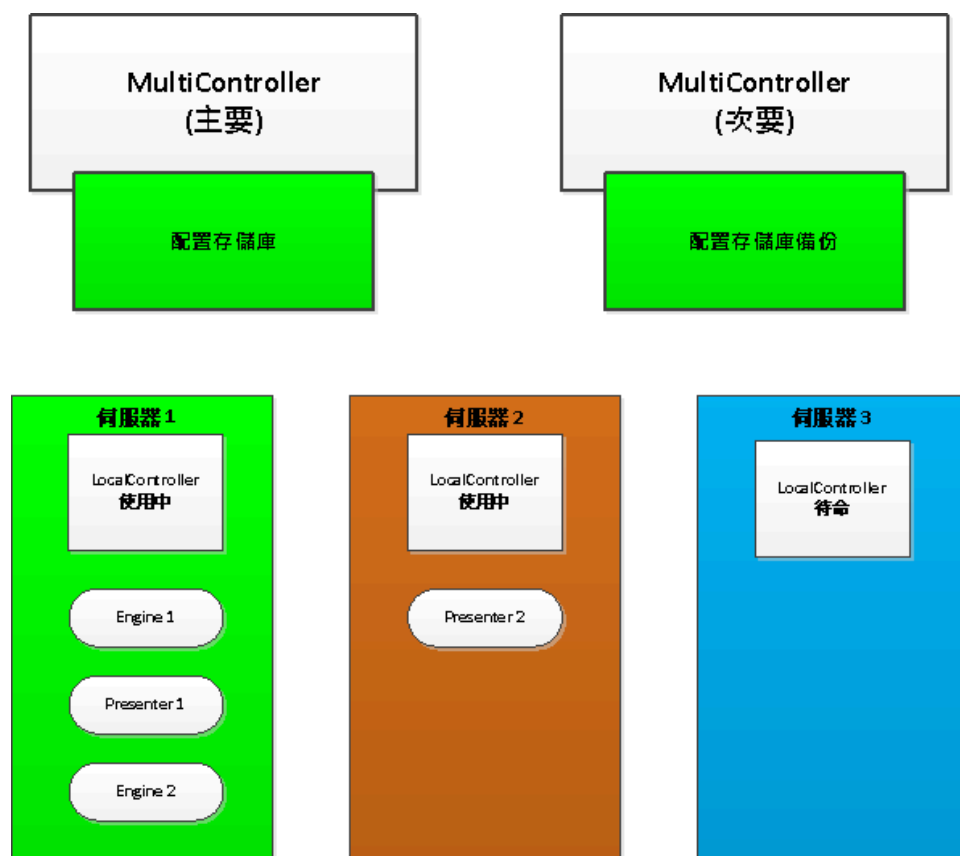
[架構](#) (位於 p. 8)

架構

CA Mediation Manager 由兩項主要元件與兩項子元件組成。主要元件為 MultiController (MC) 與 LocalController (LC)。

子元件為 Engine 與 Presenter。

下圖說明其一般架構：



此架構中也包含其他名為「一般執行程式」與「傳遞服務」的元件。這些元件未於上圖中顯示，但將在本指南的後續內容中說明。

第 2 章：總覽

附註：此章中的資訊僅適用於 CA Mediation Manager。

本節包含以下主題：

[元件總覽](#) (位於 p. 9)

[主要元件](#) (位於 p. 9)

[子元件](#) (位於 p. 16)

[其他元件](#) (位於 p. 16)

元件總覽

CA Mediation Manager 安裝程式會安裝 MultiController、LocalController 與 Web 元件。

CA Mediation Manager 安裝程式也會安裝下列元件：

- 傳遞服務 (LocalController 的一部份)
- 一般執行程式

在您安裝裝置套件時，將會安裝子元件 (Engine 與 Presenter)。

附註：如需特定裝置套件的更多安裝詳細資料，請參閱 CA Mediation Manager 安裝目錄 CAMM_HOME 下 DpConfig 資料夾中的《裝置套件指南》。

主要元件

CA Mediation Manager 有三項主要元件：MultiController、LocalController 與 Web。

MultiController

在一個叢集中最多可部署兩個 **MultiController** (一個為主要，而另一個為次要)。每個叢集至少要部署一個 **MultiController**。 **MultiController** 會在您的叢集環境中執行下列動作：

- 監控遠端伺服器上的 **LocalController** 元件所傳回的活動訊號訊息。
- 作為叢集的集中式授權伺服器。
- 將元件的集中式配置檔儲存在叢集中。

LocalController

在叢集中每個有子元件 (**Engine** 或 **Presenter**) 的實體伺服器上安裝一個 **LocalController**。 **LocalController** 會執行下列動作：

- 為安裝在伺服器上的子元件提供通訊機制。
- 對本機伺服器上的子元件監控活動訊號訊息，並在子元件失敗時自動加以重新啟動。
- 使用傳遞服務，處理來自 **Engine** 的輸出。此服務會將 **XML** 文件以壓縮且加密格式傳遞至本機或遠端 **Presenter**。

Web

Web 元件可讓您使用其 Web 型介面集中管理裝置套件部署。此介面會顯示下列資訊：

- 執行中裝置套件的狀態。
- 安裝裝置套件所在之 LocalController 的狀態。
- 主要與次要 MultiController 的狀態。

CA Mediation Manager 會安裝兩部 Web 伺服器：

- 主要 Web 伺服器
- 次要 Web 伺服器

主要 Web 伺服器會在主要 MultiController 安裝期間安裝，次要 Web 伺服器則會在次要 MultiController 安裝期間安裝。

- 若要存取主要 Web 伺服器，請啟動 CA Mediation Manager Web UI：

`http://<PrimaryMCMachineIP>:<web-port>/tim-web/index.htm`

其中，<web-port> 是在安裝 CA Mediation Manager 期間所配置的連接埠號碼，<PrimaryMCMachineIP> 是主要 *MultiController* 系統的 IP 位址或主機名稱。

如果主要 MultiController 未回應，次要 MultiController 將自動啟動次要 Web 伺服器。

如果主要 MultiController 開始回應，次要 MultiController 將停止次要 Web 伺服器。

- 若要存取次要 Web 伺服器，請啟動 CA Mediation Manager Web UI：

`http://<SecondaryMCMachineIP>:<web-port>/tim-web/index.htm`

其中，<web-port> 是在安裝 CA Mediation Manager 期間所配置的連接埠號碼，<SecondaryMCMachineIP> 是次要 *MultiController* 系統的 IP 位址或主機名稱。

後續各節將說明 Web 元件選項。

預設選項

下列資訊將說明 CA Mediation Manager 的 [管理] 索引標籤中的預設選項：

安裝或移除

從現有的儲存庫安裝或移除裝置套件。安裝期間所使用的預設路徑是 `$CAMM_HOME/MC/repository/device packs`。如果您安裝期間使用了不同的路徑，則可以瀏覽並選取裝置套件路徑。

升級

升級裝置套件的版本。

進階選項

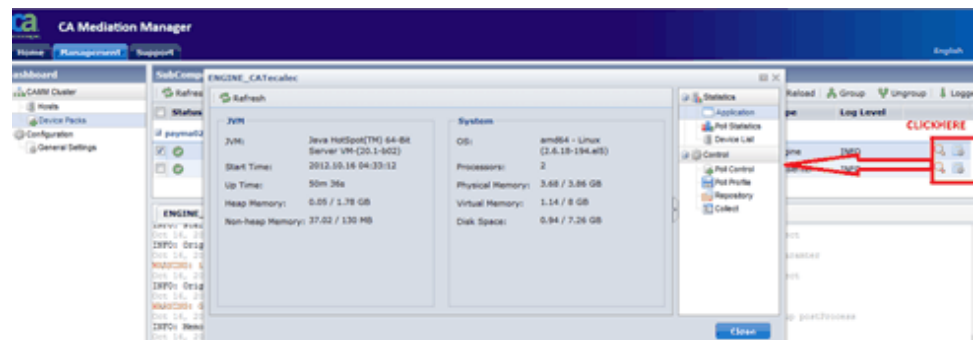
下列資訊說明進階選項。

統計資料

提供下列用以收集及瞭解統計資料的選項：

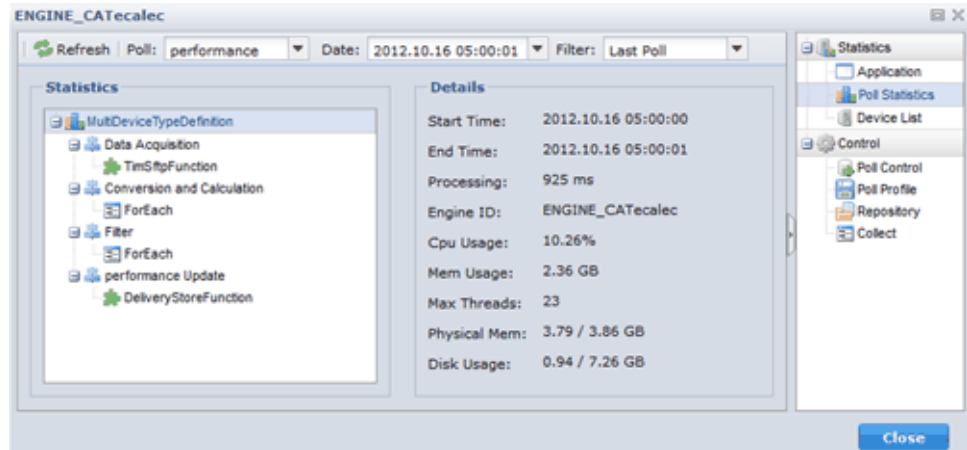
Application

提供 Java Virtual Machine (JVM) 和 Engine 或 Presenter 執行所在之系統的效能度量。



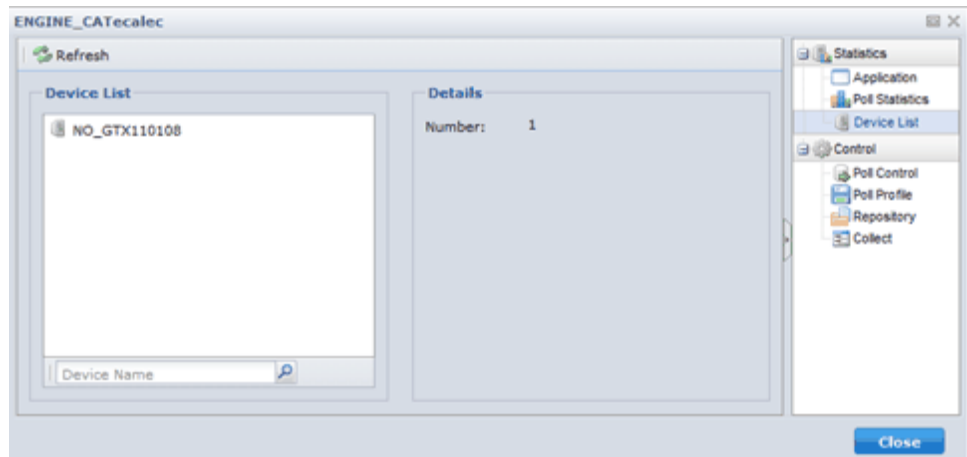
輪詢統計資料

提供清查輪詢與效能輪詢的狀態。請重新啟動引擎，以反映變更。



裝置清單

提供由清查輪詢透過位於 \$CAMM_HOME/repository/work 資料夾中的清查裝置清單檔案而搜索到的裝置清單。



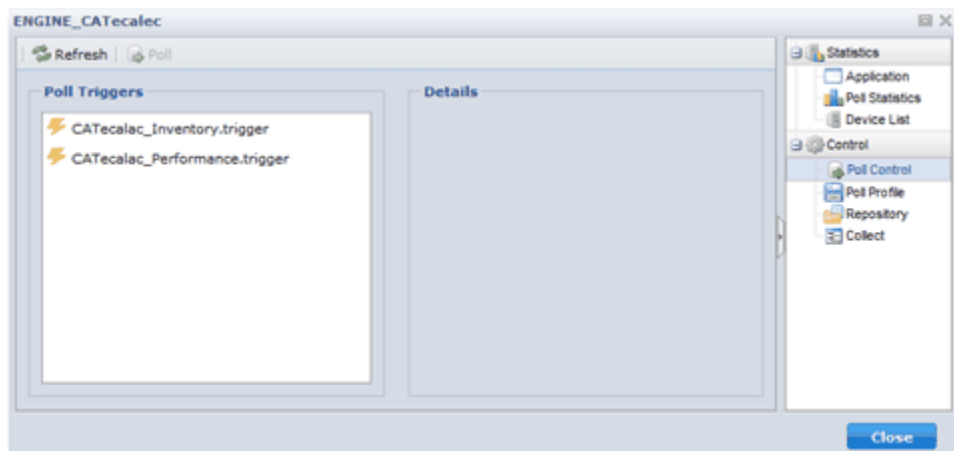
控制

提供下列介面，讓您：

- 觸發手動輪詢
- 檢視及編輯裝置套件所使用的外部全域變數值
- 檢視及編輯裝置套件檔案
- 收集記錄檔

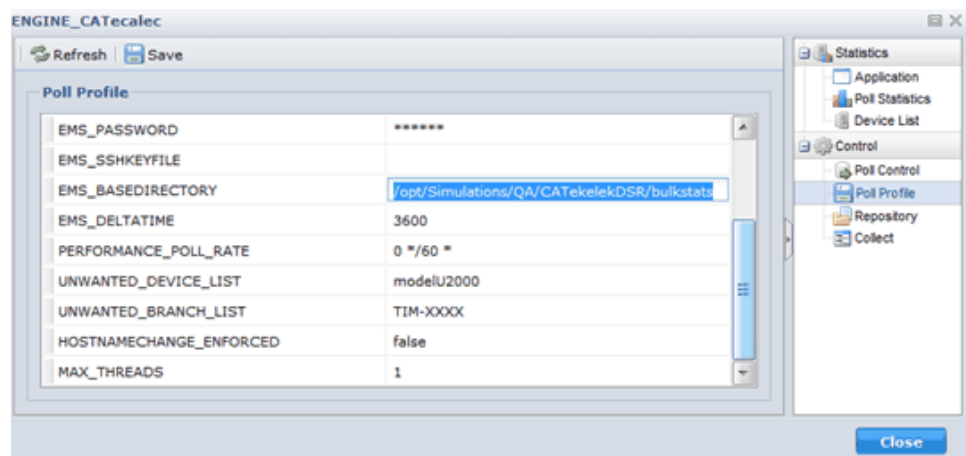
輪詢控制

觸發清查輪詢與效能輪詢。



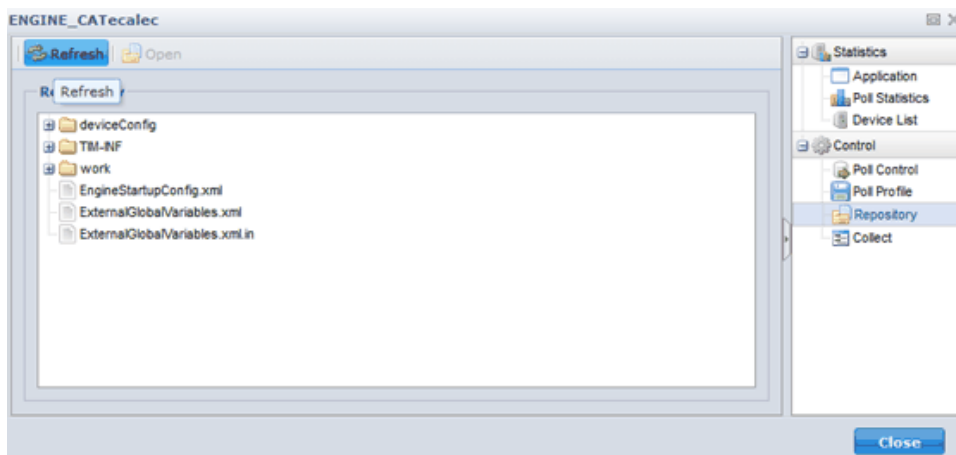
輪詢設定檔

納入您所修改的外部全域變數清單。重新啟動 Engine 或 Presenter，以反映您的變更。



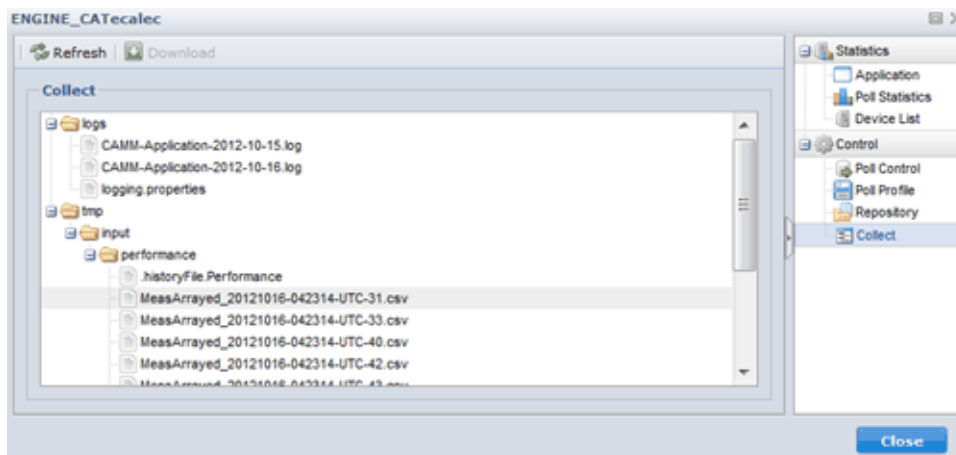
儲存庫

讓您檢視及修改 MultiController 中的裝置套件元件檔案。重新啟動 Engine 或 Presenter，以反映您的變更。



收集

收集 Engine 與 Presenter 記錄。



一般設定

使用下列選項可指定一般設定：

連線

指定 CA Mediation Manager 元件之間的重試次數與連線間隔的預設設定。

記錄

指定 CA Mediation Manager 元件的記錄選項，例如重新整理率、緩衝區大小、行數上限。

子元件

CA Mediation Manager 中有兩個子元件：Engine 與 Presenter。

Engine

Engine 是 CA Mediation Manager 中主要的執行緒輪詢引擎。您可以將 Engine 部署為使用中或待命模式。Engine 會執行下列動作：

- 使用 XML、CSV、Telnet、SSH 等從裝置收集資訊，並將資料處理為 CA Mediation Manager 標準 XML 文件。
- 將 CA Mediation Manager 標準 XML 文件部署至佇列中，由「傳遞服務」進行處理。

Presenter

Presenter 是可執行下列動作的執行緒顯示引擎：

- 接收來自 Engine 的 CA Mediation Manager 標準 XML 文件。
- 將資料設定為需要的輸出格式，例如 CSV、XML、SNMP、DDI。

其他元件

CA Mediation Manager 的另外兩個元件是「一般執行程式」與「傳遞服務」。「一般執行程式」可啟動其他元件。「傳遞服務」可將 XML 檔案輸出傳送至 Presenter。

一般執行程式

叢集中的所有元件會共用一組通用的功能，用於通訊及執行。「一般執行程式」啟動 **Engine** 與 **Presenter** 子元件，並清除暫存檔和記錄檔。

一般執行程式會在系統啟動時啟動，並接聽特定的 TCP 連接埠。在啟動諸如 **MultiController** 之類的元件時，**CA Mediation Manager** 控制公用程式 (**cammCtrl**) 會將 **MultiController XML** 配置檔傳送至一般執行程式。一般執行程式在接收到這項資料時，會使用該配置檔中的資訊來識別及啟動 **MultiController** 元件。

傳遞服務

Engine 在完成其輪詢循環時，會在佇列目錄中產生一或多個 **CA Mediation Manager** 標準 XML 文件。傳遞服務會個別監控佇列目錄，並將資料分配至一或多個本機或者遠端 **Presenter**。

如果無法使用本機或遠端 **Presenter**，「傳遞服務」將等到本機或遠端 **Presenter** 可用時才會處理此佇列。

第 3 章：安裝、解除安裝及升級 CA Mediation Manager

附註：此章中的資訊僅適用於 CA Mediation Manager。

本節包含以下主題：

[系統需求](#) (位於 p. 19)

[安裝與升級](#) (位於 p. 20)

[啟動及停止服務](#) (位於 p. 20)

系統需求

CA Mediation Manager 需要 Java Runtime Environment (JRE) 1.7 版或更新版本。

下表說明各種支援作業系統的最低硬體需求：

作業系統	架構	CPU	記憶體	磁片
Solaris 9 或 10	SPARC (64 位元)	1 x 1.4 GHz	4 GB	18 GB
Linux	x86 (64 位元)	1 x 2 GHz	4 GB	18 GB
Windows 2003	x86 (64 位元)	1 x 2 GHz	4 GB	18 GB
Windows 2008	x86 (64 位元)	1 x 2 GHz	4 GB	18 GB

附註：請維持 JRE 與作業系統架構之間的一致性。例如，在 64 位元作業系統上，您用來安裝及執行 CA Mediation Manager 的 JRE 也必須是 64 位元。CA Technologies 建議使用最新版的 JRE，您可以從 [Java 下載網站](#) 取得。

安裝與升級

若要安裝及升級 CA Mediation Manager 與裝置套件，請參閱《*CA Mediation Manager 安裝指南*》。

啓動及停止服務

下列資訊說明如何在 UNIX 與 Windows 上啓動及停止 CA Mediation Manager 中的服務。

UNIX

您可以執行 `startall`、`stopall` 或 `init.camm` 指令檔，以啓動或者停止 CA Mediation Manager。 `init.camm` 指令檔位於 CAMM 主目錄下的 `Tools` 目錄中。

以 `root` 或 `sudo su` 身分執行下列 `init.camm.install` 指令檔，可在該系統啓動或關閉時自動啓動或者停止 CA Mediation Manager：

```
shell# tools/init.camm.install
```

執行下列 `init.camm.uninstall` 指令檔，可移除能夠自動啓動或停止 CA Mediation Manager 的設定：

```
shell# tools/init.camm.uninstall
```

Windows

在 Windows 上安裝 CA Mediation Manager 期間，「一般執行程式」與 Web 元件會登錄為 Windows 服務。服務名稱為 CAMM-GE-{user}-{port} 與 CAMM-tomcat7-8880。

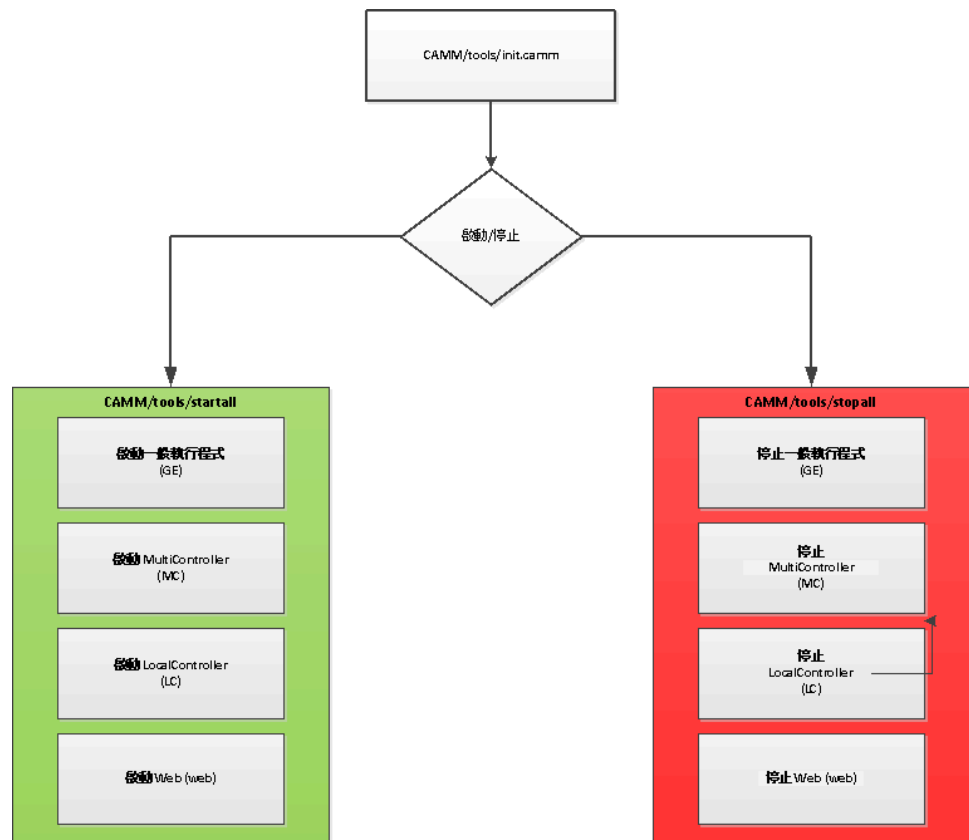
根據預設，這些服務須手動啟動。與 Linux 類似，在 Windows 中，您可以透過執行 init.camm.install.bat 在系統啟動或關閉時自動啟動或停止 CA Mediation Manager：

```
C:/CAMM/tools/init.camm.install.bat
```

執行 init.camm.uninstall.bat，可移除能夠自動啟動或停止 CA Mediation Manager 的設定：

```
C:/CAMM/tools/init.camm.uninstall.bat
```

下圖說明啟動及停止服務的處理流程：



第 4 章：元件配置

以下幾節將說明如何配置 CA Mediation Manager 中的元件。

附註：此章中的資訊僅適用於 CA Mediation Manager。

本節包含以下主題：

[一般執行程式配置](#) (位於 p. 24)

[MultiController 配置](#) (位於 p. 29)

[LocalController 配置](#) (位於 p. 36)

[容錯移轉作業](#) (位於 p. 42)

[MultiController 失敗](#) (位於 p. 42)

[高可用性配置](#) (位於 p. 46)

[記錄檔配置](#) (位於 p. 47)

一般執行程式配置

「一般執行程式」在初始安裝後通常即無需額外配置。每部伺服器只需要一個「一般執行程式」，除非您需要以不同的使用者 ID 執行元件。

「一般執行程式」配置會安裝在 CAMM HOME 目錄下名為 GE_<userid> 的「一般執行程式」目錄中。<userid> 是您可以在安裝期間指定的使用者名稱。

「一般執行程式」目錄中包含名為 LocalConfig-ge.xml 的檔案。

當「一般執行程式」擔負某個元件的角色時，將會使用 LocalConfig-ge.xml 中配置的相同使用者 ID 來執行。

UNIX 的 LocalConfig-ge.xml 檔案範例

```
<?xml version="1.0" encoding="UTF-8"?>
<AppDaemon>
  <Names>
  </Names>
  <Paths>
    <Path name="tim.base">/opt/CA/CAMM</Path>
    <Path name="appHome">${tim.base}/GE_camm</Path>
    <Path name="configBase">${appHome}/tmp</Path>
  </Paths>
  <Binding>
    CA Portal29560</Port>
    <UserId>camm</UserId>
  </Binding>
</AppDaemon>
```

Windows 的 LocalConfig-ge.xml 檔案範例

對 Windows 使用者而言，主要「一般執行程式」配置檔略有不同：

```
<?xml version="1.0" encoding="UTF-8"?>
<AppDaemon>
  <Names>
  </Names>
  <Paths>
    <Path name="tim.base">/opt/CA/CAMM</Path>
    <Path name="appHome">${tim.base}/GE_camm</Path>
    <Path name="configBase">${appHome}/tmp</Path>
  </Paths>
  <Binding>
    CA Portal29560</Port>
    <UserId>camm</UserId>
  </Binding>
  <CompanyItems>
```



```

<Item>
  <Name>MC</Name>
  <Config>${tim.base}/MC/LocalConfig-mc.xml</Config>
  CA Portal29599</Port>
</Item>
<Item>
  <Name>LC</Name>
  <Config>${tim.base}/LC/LocalConfig-lc.xml</Config>
  CA Portal29598</Port>
</Item>
</CompanyItems>
</AppDaemon>

```

新增至「一般執行程式」配置檔的 `<CompanyItems>` 區段，會定義 `MultiController` 與 `LocalController` 的位置與服務連接埠。在已定義 `<CompanyItems>` 的情況下，「一般執行程式」會每分鐘檢查一次這些項目，且若這些項目未執行，「一般執行程式」將會自動加以啟動。此配置可讓 `MultiController` 與 `LocalController` 在無外部操作的情況下啟動。

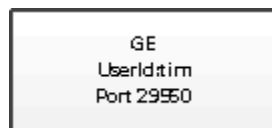
根據預設，`<CompanyItems>` 功能在安裝之後會停用。您可以執行 `caminit.install` 命令加以啟用。

附註：只有主要「一般執行程式」需使用 `<CompanyItems>` 區段。其他「一般執行程式」則不得包含此區段。

一般執行程式的運作方式

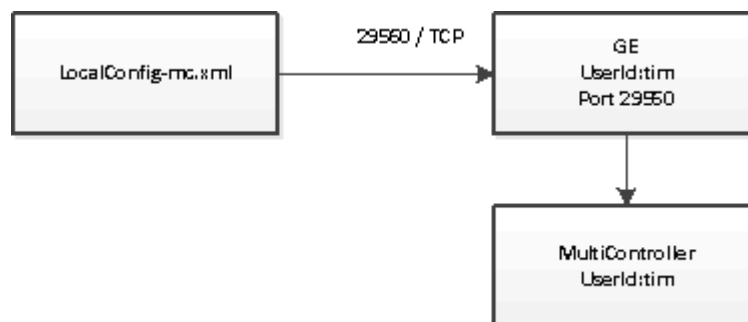
「一般執行程式」是可重複使用的實體，並且是 `CA Mediation Manager` 元件的基礎。當伺服器啟動時，至少必須有一個「一般執行程式」元件並且啟動。

根據預設，此 `LocalConfig-ge.xml` 檔案包含 `CAMM_USER` 的使用者 ID 與 TCP 連接埠資訊。下圖說明 TCP 連接埠 29560 上的「一般執行程式」以及已安裝 `CA Mediation Manager` 的 `CAMM_USER`：



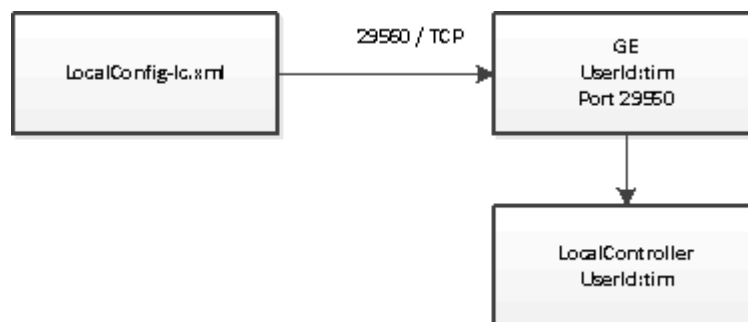
如果您需要在伺服器上執行某個 MultiController，您需要該 MultiController 的 LocalConfig-mc.xml 配置檔，以及「一般執行程式」的 TCP 連接埠。

下圖說明如何傳送 LocalConfig-mc.xml 檔案至 TCP 連接埠 29560 上的「一般執行程式」，以及「一般執行程式」如何為 MultiController 啟動新的程序：



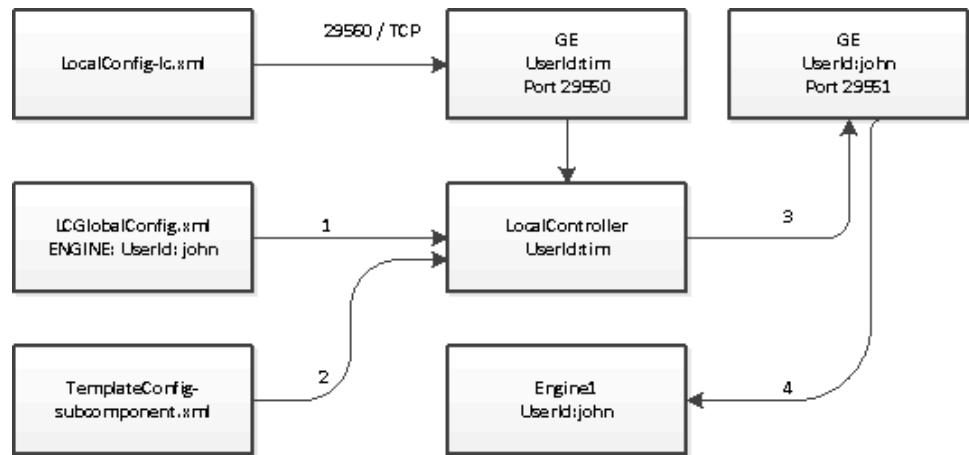
如果需要在伺服器上執行某個 LocalController，您只需要該 LocalController 的 LocalConfig-lc.xml 配置檔，以及「一般執行程式」的 TCP 連接埠。

下圖說明如何傳送 LocalConfig-lc.xml 檔案至 TCP 連接埠 29560 上的「一般執行程式」，以及「一般執行程式」如何為 LocalController 啟動新的程序：



LocalController 啟動時，會從其儲存庫讀取 LCGlobalConfig.xml 檔案。
LCGlobalConfig.xml 檔案包含子元件與連接埠資訊。

如果需要在此伺服器上執行某個 Engine，LocalController 會使用 LCGlobalConfig.xml 檔案決定必須以哪個使用者 ID 執行該 Engine。在 LocalController 目錄中，LocalController 會查閱 ExecutorMap.xml 檔案。此動作可讓 LocalController 找到以必要使用者 ID 執行的「一般執行程式」。接著，TemplateConfig-subcomponent.xml 檔案會與 LCGlobalConfig.xml 檔案的特定配置相結合。結合後的內容將傳送至「一般執行程式」的 TCP 連接埠。



一般執行程式配置選項

tim.base、appHome 與 configBase 皆為必要項目，且是由安裝程式配置。

tim.base

指定 CA Mediation Manager 的基準目錄。

appHome

指定此 CA Mediation Manager 一般執行程式的基準目錄。

configBase

指定暫存目錄，用以儲存「一般執行程式」目前執行之元件的配置。

連接埠

指定「一般執行程式」接聽的 TCP 連接埠。

預設值：TCP 29560

Userld

指定擁有「一般執行程式」程序及其子程序的使用者 ID。

一般執行程式啟動順序

您可以使用 `startall` 或 `startall.bat` 指令檔手動啟動「一般執行程式」程序 (或 Windows 上的服務)。但若執行了 `camm.init.install`，則「一般執行程式」會在系統啟動時自動啟動。

新增其他一般執行程式 (UNIX)

如果您有某個元件需要以不同的使用者 ID 執行，請新增其他「一般執行程式」。

請依循下列步驟：

1. 為需要其他使用者 ID 的 LocalController 新增新的使用者。新使用者必須是擁有 `CAMM_HOME` 目錄之相同群組的成員之一。

附註：為所有待命 LocalController 重複此程序。

2. 以新使用者的身分登入。
3. 執行下列命令，以新增新的「一般執行程式」：

```
shell# /opt/CA/CAMM/tools/camm.ge.install
```

其他「一般執行程式」隨即新增。

附註：LocalController 目錄中名為 `ExecutorMap.xml` 的配置檔會被修改，以提供必要使用者 ID 與 TCP 連接埠之間的對應。

當 Engine 子元件啟動時，LocalController 會參考在 `LCGlobalConfig` 中針對該元件而提供的使用者 ID。如果此使用者 ID 不是預設使用者 ID，則會參考 `ExecutorMap`，且替代「一般執行程式」的 TCP 連接埠會啟動子元件。

在 Engine 需使用以使用者或主機為基礎的 RSA 金鑰驗證來執行 SFTP 或 SCP 資訊時，此參考將很實用。

MultiController 配置

MultiController 是叢集中最重要的部份。它不僅包含集中的授權檔案，也維護叢集中各元件的活動訊號。

此外，叢集中的 LocalController、Engine 與 Presenter 的配置，也位於 MultiController 的儲存庫中。

MultiController 會在 TCP 連接埠號碼 29599 上接聽來自叢集成員的活動訊號作業。

在 GUI 安裝期間，您只能配置安裝基本 MultiController 所需的必要選項。但您可以手動編輯 MC 或 LocalConfig-mc.xml 檔案。

範例 LocalConfig-mc.xml 檔案 (基本配置)

```
<?xml version="1.0" encoding="UTF-8"?>
<LocalConfig>
  <Description>Configuration for Multi Controller</Description>
  <Names>
    <Name name="mainClass">com.torokina.tim.mc.Main</Name>
    <Name name="appName">CAMM-Multi-Controller</Name>
    <Name name="appShortName">MC</Name>
    <Name name="primaryMcAddress">127.0.0.1</Name>
    <Name name="secondaryMcAddress"></Name>
    <Name name="primaryMcPort">29599</Name>
    <Name name="secondaryMcPort">-1</Name>
    <Name name="myMode">primary</Name>
    <Name name="myAddress">127.0.0.1</Name>
    <Name name="mcPort">29599</Name>
    <Name name="otherMcAddress"></Name>
    <Name name="otherMcPort">-1</Name>
    <Name name="heartbeatFrequency">15</Name>
    <Name name="heartbeatTimeout">180</Name>
    <Name name="repositoryFrequency">15</Name>
  </Names>
  <Paths>
    <Path name="license">${tim.base}/license.lic</Path>
  </Paths>
</LocalConfig>
```

範例 LocalConfig-mc.xml 檔案 (隱藏的記錄與清除配置)

```
<Logging>
  <LogLevel>INFO</LogLevel>
  <LogDirectory>${logbase}</LogDirectory>
  <ObjectLogging>
    <ObjectToLog>
      <ObjectName>com.torokina.tim.config</ObjectName>
```

```

        <ObjectLogLevel>TRACE</ObjectLogLevel>
    </ObjectToLog>
</ObjectLogging>
</Logging>
<CleanUps>
    <CleanUp>
        <CleanUpName>clean-temporary-directory</CleanUpName>
        <CleanUpAction>delete</CleanUpAction>
        <CleanUpTarget>${tmp}</CleanUpTarget>
        <Parameter>
            <ParameterName>expire</ParameterName>
            <ParameterValue>3d</ParameterValue>
        </Parameter>
    </CleanUp>
    <CleanUp>
        <CleanUpName>archive-log-directory</CleanUpName>
        <CleanUpAction>archive</CleanUpAction>
        <CleanUpTarget>${logbase}</CleanUpTarget>
        <Parameter>
            <ParameterName>expire</ParameterName>
            <ParameterValue>3d</ParameterValue>
        </Parameter>
    </CleanUp>
    <CleanUp>
        <CleanUpName>clean-log-directory</CleanUpName>
        <CleanUpAction>delete</CleanUpAction>
        <CleanUpTarget>${logbase}</CleanUpTarget>
        <Parameter>
            <ParameterName>expire</ParameterName>
            <ParameterValue>7d</ParameterValue>
        </Parameter>
    </CleanUp>
</CleanUps>

```

在 LocalConfig-mc.xml 檔案中以正確的 XML 結構指定以下任何欄位，以便覆寫預設的內容。例如，下列配置會將預設的記錄層級變更為「最精細」。

將預設記錄層級變更為最精細

```

<?xml version="1.0" encoding="UTF-8"?>
<LocalConfig>
    ... ..
    <Logging>
        <LogLevel>FINEST</LogLevel>
        <LogDirectory>${logbase}</LogDirectory>
    </Logging>
    ... ..
</LocalConfig>

```

範例 LocalConfig-mc.xml (MultiController 執行時期) 檔案

```
<?xml version="1.0" encoding="UTF-8"?>
<Runtime>
  <Names>
    <Name name="mainClass">com.torokina.tim.mc.Main</Name>
    <Name name="appName">CMM-Multi-Controller</Name>
    <Name name="appShortName">MC</Name>
    <Name name="primaryMcAddress">127.0.0.1</Name>
    <Name name="secondaryMcAddress"/>
    <Name name="primaryMcPort">29599</Name>
    <Name name="secondaryMcPort">-1</Name>
    <Name name="myMode">primary</Name>
    <Name name="myAddress">127.0.0.1</Name>
    <Name name="mcPort">29599</Name>
    <Name name="otherMcAddress"/>
    <Name name="otherMcPort">-1</Name>
    <Name name="heartbeatFrequency">15</Name>
    <Name name="heartbeatTimeout">180</Name>
    <Name name="repositoryFrequency">15</Name>
    <Name name="lcPort">29598</Name>
    <Name name="manageable">469</Name>
  </Names>
  <Paths>
    <Path name="license">${tim.base}/license.lic</Path>
    <Path name="apphome">${tim.base}/${appShortName}</Path>
    <Path name="runtimeConfig">${apphome}/runtime.xml</Path>
    <Path name="tmp">${apphome}/tmp</Path>
    <Path name="logbase">${apphome}/logs</Path>
    <Path name="basedir">${tim.base}</Path>
  </Paths>
</Runtime>
```

runtime.xml 檔案會與隱藏的配置合併，然後啟動 MultiController 元件。

MultiController 配置選項

下列資訊說明可用來配置 MultiController 的選項。

路徑

可讓您指定下列項目的路徑資訊：

basedir

指定 CA Mediation Manager 的基準目錄。

apphome

指定 CA Mediation Manager MultiController 應用程式主目錄。

tmp

指定 CA Mediation Manager MultiController 暫存檔案目錄。

logbase

指定 CA Mediation Manager MultiController 記錄目錄。

runtimeConfig

指定「一般執行程式」所提供的 CA Mediation Manager MultiController 執行階段 XML 配置。

Java

可讓您指定使用 Java 的選項。

CommandPath

指定「一般執行程式」所呼叫以啟動 MultiController 之 Java 執行程式的完整路徑。

ClassPath/JarBase

可讓您建立內含一或多個可新增至 ClassPath 之項目的清單。

Options

指定剖析為 Java 的命令列選項。

環境

指定執行 MultiController 元件的環境變數。

MainClass/Class

指定要執行的主要 Java 類別。

MainClass/Args

指定剖析為 Java 類別的引數。

執行階段

可讓您指定執行階段選項。

Binding/BindAddress

指定 MultiController 元件所繫結的 IP 位址。

值：對所有 MultiController 元件使用 IP 位址 0.0.0.0。若有兩個或更多 IP 位址，請使用逗號分隔清單。

MultiController 會繫結至 TCP 連接埠 29599。

Binding/MyAddress

指定 MultiController 用來識別本身的 IP 位址。

附註：IP 位址必須是這部主機上的有效 IP 位址。

MultiControllerConfig/Mode

指定 MultiController 的作業模式。

值：指定 [主要] 或 [次要]。

MultiControllerConfig/MCAddresses/Other

指定叢集中其他 MultiController 的 IP 位址。

MultiControllerConfig/Heartbeat/ParameterName == frequency

指定傳送至其他 MultiController 之活動訊號訊息的頻率。

MultiControllerConfig/Heartbeat/ParameterName == timeout

指定此 MultiController 等待 LocalController 元件傳回活動訊號的時間長度。如果 MultiController 在 180 秒後仍未接收到來自 LocalController 的活動訊號，MultiController 將會觸發容錯移轉至第一個可用的 LocalController。

記錄

可讓您指定記錄選項。

LogLevel

指定輸出記錄檔中的記錄層級。

值：指定 DEBUG、TRACE、INFO、WARNING 或 ERROR。

LogDirectory

指定輸出記錄檔中的記錄層級。

值：指定 DEBUG、TRACE、INFO、WARNING 或 ERROR。

ObjectLogging/ObjectToLog/ObjectName

指定要啓用記錄的 Java 類別名稱。

ObjectLogging/ObjectToLog/ObjectLogLevel

指定 Java 類別的記錄層級。

Cleanup

可讓您指定清理選項。

CleanUpName

指定清理的描述性名稱。

CleanUpAction

指定清理動作。

值：指定 [刪除] 或 [封存]。

CleanupTarget

指定要清理的目錄。

值：可使用 \${camm.variable} 之類的 CA Mediation Manager 變數進行指定。

Parameter/ParameterName – Parameter/ParameterValue

使用 <n><unit> 的格式，指定要到期時間的參數及其到期時間。例如：

10d = 10 天

10h = 10 小時

10m = 10 分鐘

手動啓動及停止 MultiController

init.camm 指令檔會自動啓動 MultiController 程序。您可以使用 cammCtrl 公用程式，手動個別停止或啓動 MultiController 元件。

請依循下列步驟：

1. 以 CAMM_USER 的身分登入，然後移至 CAMM 主目錄。
2. 執行下列命令，以啓動 MultiController：

```
/opt/CA/CAMM# tools/startall -c mc
```

3. 執行下列命令，以停止 MultiController：

```
/opt/CA/CAMM # tools/stopall -c mc
```

LocalController 配置

LocalController 是叢集中的每部伺服器上都會安裝的重要服務。

LocalController 會執行下列主要功能：

- 讓執行於本機伺服器上的子元件與遠端 MultiController 之間的通訊更加順暢。
- 監控 Engine 與 Presenter 的效能與可用性，並重新啟動任何失敗的元件。
- 接聽來自叢集成員的活動訊號作業。根據預設，LocalController 會在連接埠 29598 或 TCP 上接聽。
- 將活動訊號資訊傳送至 MultiController。
- 啟動、停止及重新啟動本機 Engine 與 Presenter。

在 GUI 安裝期間，您只能配置安裝基本 LocalController 所需的必要選項。但您可以手動編輯 LC 或 LocalConfig-lc.xml 檔案。

範例 LocalConfig-lc.xml 檔案 (基本配置)

```
<LocalConfig>
  <Names>
    <Name name="mainClass">com.torokina.tim.lc.Main</Name>
    <Name name="primaryMcAddress">127.0.0.1</Name>
    <Name name="secondaryMcAddress"></Name>
    <Name name="primaryMcPort">29599</Name>
    <Name name="secondaryMcPort">-1</Name>
    <Name name="myAddress">127.0.0.1</Name>
    <Name name="appName">CMM-Local-Controller</Name>
    <Name name="appShortName">LC</Name>
    <Name name="lcPort">29598</Name>
    <Name name="heartbeatFrequency">15</Name>
    <Name name="heartbeatTimeout">180</Name>
  </Names>
  <Paths>
    <Path name="dsLocalConfig">${basedir}/DS/LocalConfig-ds.xml</Path>
  </Paths>
</LocalConfig>
```

範例 LocalConfig-lc.xml 檔案 (隱藏的記錄與清除配置)

```
<Logging>
  <LogLevel>INFO</LogLevel>
  <LogDirectory>${logbase}</LogDirectory>
  <ObjectLogging>
    <ObjectToLog>
```

```

        <ObjectName>com.torokina.tim.config</ObjectName>
        <ObjectLogLevel>TRACE</ObjectLogLevel>
    </ObjectToLog>
</ObjectLogging>
</Logging>
<CleanUps>
    <CleanUp>
        <CleanUpName>clean-temporary-directory</CleanUpName>
        <CleanUpAction>delete</CleanUpAction>
        <CleanUpTarget>${tmp}</CleanUpTarget>
        <Parameter>
            <ParameterName>expire</ParameterName>
            <ParameterValue>3d</ParameterValue>
        </Parameter>
    </CleanUp>
    <CleanUp>
        <CleanUpName>archive-log-directory</CleanUpName>
        <CleanUpAction>archive</CleanUpAction>
        <CleanUpTarget>${logbase}</CleanUpTarget>
        <Parameter>
            <ParameterName>expire</ParameterName>
            <ParameterValue>3d</ParameterValue>
        </Parameter>
    </CleanUp>
    <CleanUp>
        <CleanUpName>clean-log-directory</CleanUpName>
        <CleanUpAction>delete</CleanUpAction>
        <CleanUpTarget>${logbase}</CleanUpTarget>
        <Parameter>
            <ParameterName>expire</ParameterName>
            <ParameterValue>7d</ParameterValue>
        </Parameter>
    </CleanUp>
</CleanUps>

```

範例 LocalConfig-lc.xml 檔案 (LocalController 執行時期)

```

<?xml version="1.0" encoding="UTF-8"?>
<Runtime>
    <Names>
        <Name name="mainClass">com.torokina.tim.lc.Main</Name>
        <Name name="primaryMcAddress">127.0.0.1</Name>
        <Name name="secondaryMcAddress"/>
        <Name name="primaryMcPort">29599</Name>
        <Name name="secondaryMcPort">-1</Name>
        <Name name="myAddress">127.0.0.1</Name>
        <Name name="appName">CMM-Local-Controller</Name>
        <Name name="appShortName">LC</Name>
        <Name name="lcPort">29598</Name>
        <Name name="heartbeatFrequency">15</Name>
    </Names>

```

```
<Name name="heartbeatTimeout">180</Name>
<Name name="mcPort">29599</Name>
<Name name="manageable">996</Name>
</Names>
<Paths>
  <Path name="dsLocalConfig">${basedir}/DS/LocalConfig-ds.xml</Path>
  <Path name="apphome">${tim.base}/${appShortName}</Path>
  <Path name="runtimeConfig">${apphome}/runtime.xml</Path>
  <Path name="tmp">${apphome}/tmp</Path>
  <Path name="logbase">${apphome}/logs</Path>
  <Path name="basedir">${tim.base}</Path>
</Paths>
</Runtime>
```

LocalController 配置選項

下列資訊說明可用來配置 LocalController 的選項。

路徑

可讓您指定下列項目的路徑資訊：

basedir

指定 CA Mediation Manager 的基準目錄。

apphome

指定 CA Mediation Manager LocalController 應用程式主目錄。

tmp

指定 CA Mediation Manager LocalController 暫存檔案目錄。

logbase

指定 CA Mediation Manager LocalController 記錄目錄。

runtimeConfig

指定「一般執行程式」所提供的 CA Mediation Manager LocalController 執行階段 XML 配置檔。

Java

可讓您指定使用 Java 的選項。

CommandPath

指定「一般執行程式」所呼叫以啟動 LocalController 之 Java 執行程式的完整路徑。

ClassPath/JarBase

可讓您建立內含一或多個可新增至 ClassPath 之項目的清單。

Options

指定剖析為 Java 的命令列選項。

環境

指定執行 MultiController 元件的環境變數。

MainClass/Class

指定要執行的主要 Java 類別。

MainClass/Args

指定剖析為 Java 類別的引數。

執行階段

可讓您指定執行階段選項。

Binding/BindAddress

指定 LocalController 元件所繫結的 IP 位址。

值：對所有 LocalController 元件使用 IP 位址 0.0.0.0。若有兩個或更多 IP 位址，請使用逗號分隔清單。

Binding/MyAddress

指定 LocalController 用來識別本身的 IP 位址。

附註：此 IP 位址必須是這部主機上的有效 IP 位址。

Binding/BindPort

指定 LocalController 用來識別本身的 IP 位址。

附註：此 IP 位址必須是這部主機上的有效 IP 位址。根據預設，LocalController 會繫結至 TCP 連接埠 29598。

LocalControllerConfig/Mode

指定 LocalController 的作業模式。

值：指定 Active 或 Standby。

LocalControllerConfig/MCAddresses/Primary

指定叢集中主要 MultiController 的 IP 位址。

LocalControllerConfig/MCAddresses/Secondary

指定叢集中次要 MultiController 的 IP 位址。

LocalControllerConfig/Heartbeat/ParameterName == frequency

指定傳送至 MultiController 之活動訊號訊息的頻率。

LocalControllerConfig/Heartbeat/ParameterName == timeout

指定此 LocalController 等待子元件 (Engine 與 Presenter) 傳回活動訊號的時間長度。如果 LocalController 在 180 秒後仍未接收到來自子元件的活動訊號，LocalController 將會重新啟動。

記錄

可讓您指定記錄選項。

LogLevel

指定輸出記錄檔中的記錄層級。

值：指定 DEBUG、TRACE、INFO、WARNING 或 ERROR。

LogDirectory

指定輸出記錄檔中的記錄層級。

值：指定 DEBUG、TRACE、INFO、WARNING 或 ERROR。

ObjectLogging/ObjectToLog/ObjectName

指定要啓用記錄的 Java 類別名稱。

ObjectLogging/ObjectToLog/ObjectLogLevel

指定 Java 類別的記錄層級。

Cleanup

可讓您指定清理選項。

CleanUpName

指定清理的描述性名稱。

CleanUpAction

指定清理動作。

值：指定 [刪除] 或 [封存]。

CleanupTarget

指定要清理的目錄。

值：可使用 \${camm.variable} 之類的 CA Mediation Manager 變數進行指定。

Parameter/ParameterName – Parameter/ParameterValue

使用 <n><unit> 的格式，指定要到期時間的參數及其到期時間。例如：

10d = 10 天

10h = 10 小時

10m = 10 分鐘

手動啟動及停止 LocalController

init.camm 指令檔會自動啟動 LocalController 程序。您可以使用 cammCtrl 公用程式，手動個別停止或啟動 LocalController 元件。

請依循下列步驟：

1. 以 CAMM_USER 的身分登入，然後移至 CAMM 主目錄。
2. 執行下列命令，以啟動 LocalController：

```
/opt/CA/CAMM# tools/startall -c lc
```

3. 執行下列命令，以停止 LocalController：

```
/opt/CA/CAMM # tools/stopall -c lc
```

Engine 與 Presenter 的配置

Engine 的安裝與配置，以及 Presenter 的封存可使用每個裝置套件隨附的「裝置套件」安裝程式來達成。

容錯移轉作業

MultiController 有兩種作業模式：「主要」或「次要」。

MultiController 有三項主要通訊功能：

- 活動訊號
- 名稱服務查閱
- 儲存庫同步處理

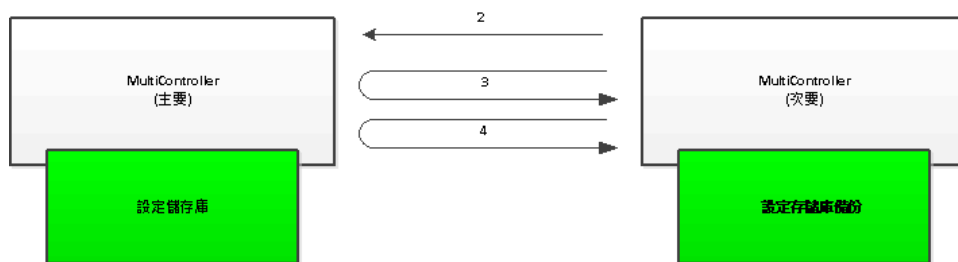
MultiController 失敗

下列資訊說明 CA Mediation Manager 如何管理 MultiController 的失敗。

MultiController 通訊

下圖說明次要與主要 MultiController 之間的通訊方式。

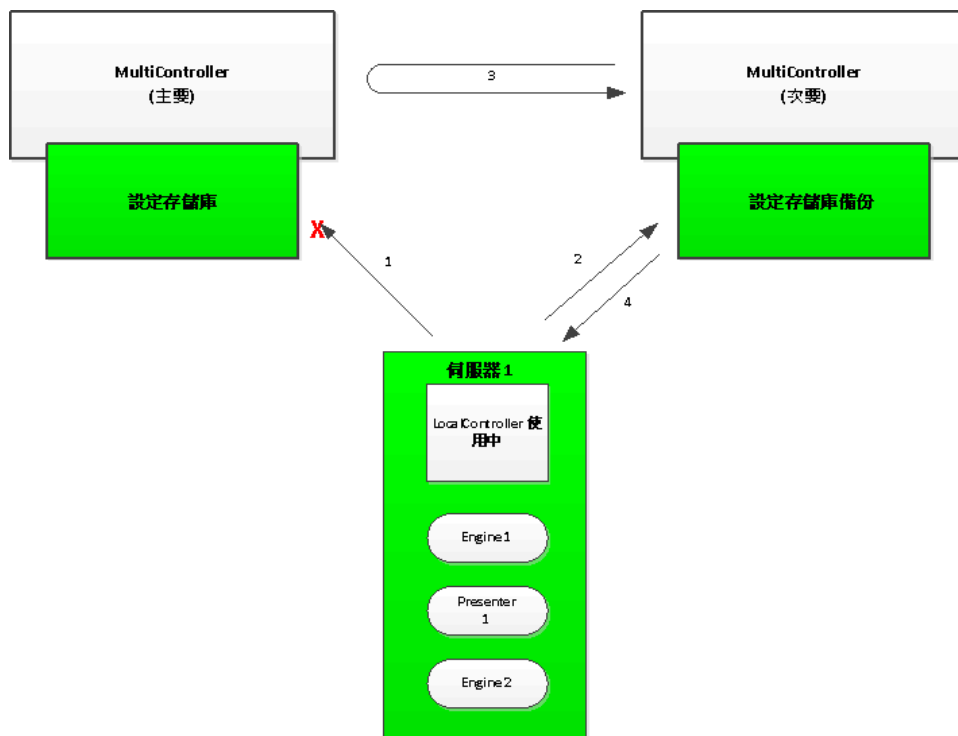
主要 MultiController 上有一個監視程式，會同時注意次要 MultiController 是否已無法使用。當次要 MultiController 無法使用時，將會在記錄檔中建立警示。



1. 啟動次要 MultiController。
2. 次要 MultiController 會將第一個活動訊號傳送到主要 MultiController，以註冊為運作中。
3. 次要 MultiController 會定期將活動訊號傳送到主要 MultiController。
4. 次要 MultiController 會定期同步存儲庫。

主要 MultiController 失敗

下圖說明次要 MultiController 如何處理主要 MultiController 的失敗。在此程序期間，次要 MultiController 會接管活動訊號、名稱服務與配置儲存庫要求，直到主要 MultiController 恢復可用為止。



1. LocalController 傳送活動訊號至主要 MultiController，且失敗。
2. LocalController 傳送活動訊號至次要 MultiController。
3. 因為次要 MultiController 相信主要 MultiController 仍然可用，其會將要求代理至主要 MultiController。
4. 主要 MultiController 會透過次要 MultiController 回復活動訊號至 LocalController。

MultiController 代理功能

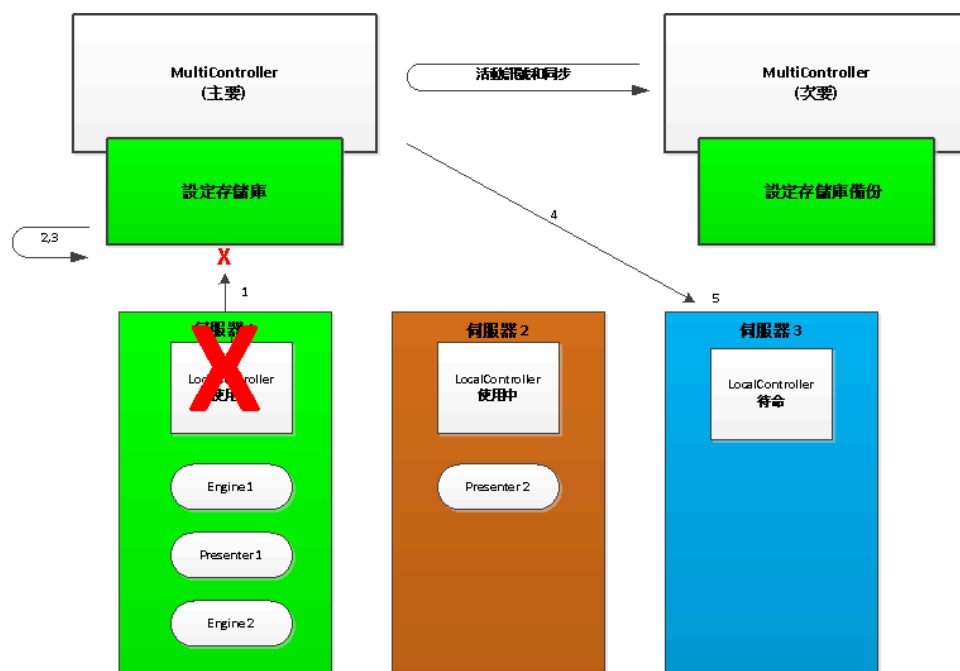
在特定的網路中，LocalController 無法直接與主要 MultiController 聯絡。因此，次要 MultiController 會使用其代理功能，嘗試將活動訊號傳送至主要 MultiController。

但傳送至次要 MultiController 的配置儲存庫要求會由次要 MultiController 進行處理，使得配置儲存庫能夠同步。

LocalController 失敗

所有 LocalController 元件都會將活動訊號資訊傳送至 MultiController。使用中的 MultiController 若偵測到本身未接收到來自 LocalController 的活動訊號，將會記錄遺失的活動訊號。如果這段時間超過逾時期間，則 MultiController 會使用下圖中說明的程序觸發容錯移轉，至第一個可用的待命 LocalController。

容錯回復是手動程序。若要達成容錯移轉，請強制讓伺服器 3 上的 LocalController 失敗。



1. Multi Controller 注意到已超過逾時期間，但尚未收到活動訊號。
2. Multi Controller 會從伺服器 1 將所有 Engine 子元件的設定存儲庫移動至伺服器 3。
3. 若伺服器 1 並未安裝任何子元件，則 Multi Controller 會將伺服器 1 的設定標示為待命。
4. Multi Controller 會傳送訊號給伺服器 3 上的 Local Controller，以重新啟動並重新讀取其設定。
5. 伺服器 3 上的 LocalController 會變為使用中。

子元件失敗

LocalController 元件會監控其執行中的子元件 (Engine 與 Presenter)。如果某子元件失敗，該子元件會立即重新啟動。

高可用性配置

以下幾節將說明如何針對高可用性配置 CA Mediation Manager。

配置主要 MultiController

在次要 MultiController 安裝期間，請針對高可用性配置主要 MultiController。

請依循下列步驟：

1. 在 [MultiController 配置] 面板中，選取 [叢集中是否會有其他 MC] 核取方塊。
2. 提供主要 MultiController 的 IP 位址，然後按 [下一步]。
3. 在 [其他 MultiController 配置] 面板中，提供次要 MultiController 的 IP 位址。
4. 按 [下一步]，並完成安裝。

配置次要 MultiController

在次要 MultiController 安裝期間，請針對高可用性配置次要 MultiController。

請依循下列步驟：

1. 在 [MultiController 配置] 面板中，從 MultiController 下拉式清單中選取 [次要]。
2. 提供次要 MultiController 的 IP 位址。
3. 選取 [叢集中是否會有其他 MC] 核取方塊，然後按 [下一步]。
4. 在 [其他 MultiController 配置] 面板中，提供主要 MultiController 的 IP 位址。
5. 按 [下一步]，並完成安裝。

配置 LocalController

在 LocalController 安裝期間，請針對高可用性配置 LocalController。

請依循下列步驟：

1. 在 [LocalController 配置] 面板中，提供主要 MultiController 與次要 MultiController 的 IP 位址。
2. 按 [下一步]，並完成安裝。
3. 為每個 LocalController 安裝重複這些步驟。

記錄檔配置

以下幾節將說明如何配置 CA Mediation Manager 中的記錄檔。

使用 `logging.properties` 檔配置所有 CA Mediation Manager 元件的記錄內容，這個檔案是在啟動元件時於預設的記錄目錄中建立的。`logging.properties` `file_` 已預先配置為在預設的記錄目錄中產生記錄檔。不過，您可以編輯 `logging.properties` 檔，將記錄檔案導向另一個目錄。修改 `logging.properties` 檔後，請重新啟動元件以載入修改後的記錄內容。所有記錄檔 (應用程式記錄檔與 STD-ERROR/STD-OUTPUT 記錄檔) 均產生於 `logging.properties` 檔案指定的目錄中。

logging.properties 檔案 - 依元件提供的範例

下列範例說明不同的 logging.properties 檔案：

MultiController 的範例 logging.properties 檔案

```
com.torokina.common.logging.apache.FileHandler.directory=/opt/CA/CAMM/MC/logs  
  
handlers=com.torokina.common.logging.apache.FileHandler  
  
com.torokina.common.logging.apache.FileHandler.level=INFO  
  
.level=INFO  
  
com.torokina.common.logging.apache.FileHandler.prefix=CAMM-Multi-Controller-
```

LocalController 的範例 logging.properties 檔案

```
com.torokina.common.logging.apache.FileHandler.directory=/opt/CA/CAMM/LC/logs  
  
handlers=com.torokina.common.logging.apache.FileHandler  
  
com.torokina.common.logging.apache.FileHandler.level=INFO  
  
.level=INFO  
  
com.torokina.common.logging.apache.FileHandler.prefix=CAMM-Local-Controller-
```

傳遞系統的範例 logging.properties 檔案

```
com.torokina.common.logging.apache.FileHandler.directory=/opt/CA/CAMM/DS/logs  
  
handlers=com.torokina.common.logging.apache.FileHandler  
  
com.torokina.common.logging.apache.FileHandler.level=INFO  
  
.level=INFO  
  
com.torokina.common.logging.apache.FileHandler.prefix=CAMM-Delivery-System-
```

ENGINE_CAMM 的範例 logging.properties 檔案

```
com.torokina.common.logging.apache.FileHandler.directory=/opt/CA/CAMM/COMPONENTS/ENGINE_CAMM/logs  
  
handlers=com.torokina.common.logging.apache.FileHandler
```



```
com.torokina.common.logging.apache.FileHandler.level=INFO
.level=INFO

com.torokina.common.logging.apache.FileHandler.prefix=CAMM-ENGINE_
CAMM-
```

一般執行程式的範例 `logging.properties` 檔案

根據預設，「一般執行程式」不會在其記錄目錄中建立 `logging.properties` 檔案。所有記錄都會在 `~GE/logs` 目錄中產生。下列範例說明如何在記錄目錄中建立 `logging.properties` 檔案，以重新導向「一般執行程式」記錄 (僅適用於非 Windows 平台)：

```
#Properties for Logger

#Tue May 07 04:08:45 EDT 2013

com.torokina.common.logging.apache.FileHandler.directory=/opt/CA/C
AMM/GE/logs

handlers=com.torokina.common.logging.apache.FileHandler

com.torokina.common.logging.apache.FileHandler.level=INFO
.level=INFO

com.torokina.common.logging.apache.FileHandler.prefix=CAMM-Generic
-Executor-
```

配置記錄檔清理

根據預設，每個元件的清理動作都會配置成在 `logbase` 目錄上執行。如果記錄檔重新導向至其他目錄，請修改清理配置，以順利執行封存或刪除動作。

請在配置檔的 `LocalConfig` XML 元素中定義清理動作。對於所有的 CAMM 元件，請在對應的配置檔中定義清理動作：

- **MultiController**：LocalConfig-mc.xml
- **LocalController**：LocalConfig-lc.xml
- **傳遞服務**：LocalConfig-ds.xml
- **子元件 (ENGINE 或 PRESENTER)**：TemplateConfig-subcomponent.xml

附註：TemplateConfig-subcomponent.xml 檔案位於 `<cam.base>/LC` 目錄中。

以下範例說明兩個範例清理配置。

範例：傳遞服務元件的清理配置檔 (刪除動作)

```
<LocalConfig>
<Description>Configuration for Delivery Module</Description>
...
...
<CleanUps>
  <!-- SAMPLE DELETE ACTION -->
  <CleanUp>
    <CleanUpName>Delete</CleanUpName>
    <CleanUpAction>delete</CleanUpAction>

    <CleanUpTarget>${apphome}/.local</CleanUpTarget> <!--
Directory Name -->
    <Parameter>
      <ParameterName>expire</ParameterName>
      <ParameterValue>7d</ParameterValue> <!--
1y0m3d1h -->
    </Parameter>
    <Parameter>
      <ParameterName>includeDir</ParameterName>

      <ParameterValue>true</ParameterValue><!-- true/false -->
    </Parameter>
    <Parameter>
      <ParameterName>recursive</ParameterName>

      <ParameterValue>true</ParameterValue><!-- true/false -->
    </Parameter>
```

```
<Parameter>
    <ParameterName>match</ParameterName>

    <ParameterValue>^[\\d]+\\.xml$</ParameterValue>

    <!-- Regular Pattern -->

</Parameter>

</CleanUp>
```

範例：傳遞服務元件的清理配置檔 (封存動作)

```
<!-- SAMPLE ARCHIVE ACTION -->
<CleanUp>
    <CleanUpName>Archive</CleanUpName>
    <CleanUpAction>archive</CleanUpAction>
    <CleanUpTarget>${logbase}</CleanUpTarget> <!--
Directory Name -->
    <Parameter>
        <ParameterName>expire</ParameterName>
        <ParameterValue>7d</ParameterValue> <!--
1y0m3d1h -->
    </Parameter>
    <Parameter>
        <ParameterName>includeDir</ParameterName>
        <ParameterValue>true</ParameterValue><!-- true/false -->
    </Parameter>
    <Parameter>
        <ParameterName>recursive</ParameterName>
        <ParameterValue>true</ParameterValue><!-- true/false -->
    </Parameter>
    <Parameter>
        <ParameterName>match</ParameterName>
        <ParameterValue>CMM-.*\..log</ParameterValue>
        <!-- Regular Pattern -->
    </Parameter>
    <Parameter>
```

```

        <ParameterName>achiveHome</ParameterName>

        <ParameterValue>${logbase}</ParameterValue> <!-- folder path
-->

        </Parameter>
        <Parameter>

        <ParameterName>achivePrefix</ParameterName>

        <ParameterValue>Archive-</ParameterValue> <!-- prefix string
-->

        </Parameter>
        <Parameter>

        <ParameterName>achiveSuffix</ParameterName>

        <ParameterValue>.zip</ParameterValue>
<!-- suffix string -->

        </Parameter>

        </Cleanup>

    </CleanUps>

</LocalConfig>

```

您可以在元件配置檔中納入任何清理動作。「一般執行程式」會執行所有的清理動作。「一般執行程式」會將清理檔案儲存在 {camm.base }/GE_<User>/cleanup 目錄中；其中，camm.base 是 CA Mediation Manager 的安裝目錄。在元件配置檔經過修改後，請重新啓動相關的元件，使修改後的配置能夠生效。

第 5 章：將 EMS 設定檔用於 CA Mediation Manager for Infrastructure Management 2.0 (2.2.3 版)

附註：此章中的資訊僅適用於 CA Mediation Manager for Infrastructure Management 2.0。

本節包含以下主題：

[EMS 整合設定檔](#) (位於 p. 55)

[新增事件規則](#) (位於 p. 59)

EMS 整合設定檔

EMS 整合設定檔會指定 EMS 清查搜索在您的 Data Aggregator 環境中的運作方式。

透過 EMS 整合設定檔，您將可指定狀態、資料收集器、裝置套件、EMS IP 與備用 EMS IP。請為您所建立的每個 EMS 整合設定檔指定一個 IP 網域。您指定的 IP 網域，會用於管理多項裝置 (通常會同時管理 1,000 個裝置) 的目標 EMS 伺服器。Data Aggregator 可持續並且一併處理來自 EMS 伺服器的清查資料。(使用 SNMP 或 ICMP 時，輪詢會對個別裝置逐一執行。)

附註：

- 在安裝裝置套件後，[EMS 整合設定檔] 選項會在使用者介面中顯示。
- 請不要為相同的 Data Collector 重複新增相同的 EMS 整合設定檔。

新增 EMS 整合設定檔

您可以建立 EMS 整合設定檔，以指定 EMS 清查搜索在您的 Data Aggregator 環境中的運作方式。

附註：您必須以管理員的身分登入，才能執行此工作。

未先設定租戶的範圍便建立 EMS 整合設定檔，會將該設定檔置於全域空間中供所有租戶存取。使用全域空間中的設定檔執行搜索，任何人都將可看見搜索結果，無論他們是否設定了租戶的範圍。

因此，請在建立 EMS 整合設定檔之前設定租戶的範圍，讓該設定檔僅供特定租戶存取。在您設定租戶的範圍後，頁面右上方會出現租戶指標。接下來，您可以手動同步處理租戶與 CA Infrastructure Management，或等待同步自動執行。在租戶與 Data Aggregator 同步前，您將無法建立 EMS 整合設定檔。

附註：如需設定租戶範圍與同步租戶的更多資訊，請參閱《CA Performance Center 管理指南》。

請依循下列步驟：

1. 依序選取 [管理] 及 [資料來源設定]，然後在 CA Performance Center 使用者介面中按一下 Data Aggregator 資料來源。
2. 從 [監控配置] 功能表中按一下 [EMS 整合設定檔]。

[EMS 整合設定檔] 頁面就會開啓，並且會顯示可用的搜索設定檔清單。

3. 按一下 [新建]。

[新增 EMS 搜索設定檔] 對話方塊隨即開啓。

4. 在欄位中輸入必要資訊。顯示的配置欄位視您選取的裝置套件而定。每個裝置套件都會有您所配置的唯一全域變數。

附註：如需每項產品之唯一全域變數的更多資訊，請參閱 CA 支援網站。

5. 按一下 [儲存]。

EMS 整合設定檔隨即建立。

清查搜索不會在您按一下 [儲存] 且選取 [已啓用] 選項時自動執行。清查搜索只會在符合下列其中一個條件時執行：

- 已到達清查輪詢排程時間。
- EMS 整合設定檔隨即手動建立。

更多資訊：

[檢視 EMS 搜索結果](#) (位於 p. 58)

[手動啓動 EMS 搜索](#) (位於 p. 57)

手動啓動 EMS 搜索

EMS 整合設定檔可在您的網路中搜索裝置及其元件。您可以手動啓動 EMS 整合設定檔，以開始搜索。

附註：或者，您可以等候到達清查輪詢排程時間，以自動開始搜索。

請依循下列步驟：

1. 依序選取 [管理] 及 [資料來源設定]，然後在 CA Performance Center 使用者介面中按一下 **Data Aggregator** 資料來源。

2. 從 [監控配置] 功能表中按一下 [EMS 整合設定檔]。

[EMS 整合設定檔] 頁面就會開啓，並且會顯示可用的搜索設定檔清單。

3. 選取一或多個要執行搜索的 EMS 整合設定檔，然後按一下 [啓動]。

附註：您只能對處於 'READY' 狀態的 EMS 整合設定檔執行搜索。

確認對話方塊隨即開啓。

4. 按一下 [是]。

搜索隨即啓動。選取之搜索設定檔的 [狀態] 欄會顯示為 'Started'。

確認對話方塊隨即開啓。

5. 按一下 [確定]。

此時會搜索裝置及其所有相關聯的介面，並開始執行輪詢。您會回到 [EMS 整合設定檔] 頁面。

如果搜索停滯超過 10 分鐘，**Data Aggregator** 會加以停止。若在 10 分鐘內未搜索到新的裝置，且選取的搜索設定檔的狀態在 10 分鐘內沒有變化，**Data Aggregator** 即會將搜索視為停滯。此時會對搜索執行個體項目產生稽核事件。若未順利搜索到裝置，選取之搜索設定檔的 [狀態] 欄會顯示為 'FAILURE'。若至少搜索到一個裝置，但並未順利搜索到所有裝置，則 [狀態] 欄會顯示為 'PARTIAL_FAILURE'。

找到的裝置與元件最多需要 5 分鐘，才能完成與 CA Performance Center 的同步。當同步處理完成時，找到的裝置和元件會出現在 CA Performance Center 的 [清查] 索引標籤中。

更多資訊：

[檢視 EMS 搜索結果](#) (位於 p. 58)

檢視 EMS 搜索結果

您可以檢視搜索到的可管理 EMS 裝置的總數摘要。

請依循下列步驟：

1. 依序選取 [管理] 及 [資料來源設定]，然後在 CA Performance Center 使用者介面中按一下 Data Aggregator 資料來源。
2. 從 [監控配置] 功能表中按一下 [EMS 整合設定檔]。

[EMS 整合設定檔] 頁面就會開啓，並且會顯示可用的搜索設定檔清單。

3. 選取您要檢視搜索結果的 EMS 整合設定檔執行個體，然後按一下 [歷程記錄]。

[EMS 歷程記錄] 結果的顯示如下所示：

- [裝置] 表格會顯示受監控的裝置，以及每個裝置的建立時間。
- [元素] 表格會顯示受監控的介面，以及每個介面的建立時間。

啓動或停止 EMS 搜索服務

「啓動」服務是用於 EMS 伺服器上的搜索，用以持續檢閱清查。雖然搜索是可排程的作業，但您也可以視需要手動啓動、停止或重新啓動服務。例如，您可以在 EMS 伺服器於關閉後重新開啓時重新啓動搜索，或在升級裝置套件安裝後重新啓動搜索。

停止服務會刪除任何非使用中的資料輪詢，但會等待所有使用中的資料輪詢完成，而不會加以中斷。不會刪除任何 EMS 檔案。此動作也會使任何處於已停止狀態的新輪詢停止執行。此服務會持續處於停止狀態，直到您重新加以啓動爲止。

重新啓動 Data Collector 不會對任何 EMS 整合設定檔的狀態產生影響。

附註：您必須以管理員的身分登入，才能執行此工作。

請依循下列步驟：

1. 依序選取 [管理] 及 [資料來源設定]，然後在 CA Performance Center 使用者介面中按一下 Data Aggregator 資料來源。
2. 從 [監控配置] 功能表中按一下 [EMS 整合設定檔]。

[EMS 整合設定檔] 頁面就會開啓，並且會顯示可用的搜索設定檔清單。

3. 選取設定檔，然後按一下 [啓動] 或 [停止]。

確認對話方塊隨即開啓。

4. 按一下 [是] 以確認動作。

服務會啓動或停止，視您的選擇而定。此服務會持續處於啓動或停止狀態，直到您手動加以變更爲止。

新增事件規則

事件規則可透過 Data Aggregator 監控設定檔對話方塊來新增。

附註：如需更多資訊，請參閱《Data Aggregator 管理指南》或線上說明。