



Java Desktop System Configuration Manager 發行版本 1.1 安裝指南

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

文件號碼: 819-0963-10
2005 年 2 月

Copyright 2004 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. 版權所有

此產品或文件受著作權的保護，其使用、複製、分送以及反編譯，均受軟體使用權限制。未經 Sun 及其授權者 (如果有的話) 的書面授權，不得以任何形式或任何方法，重新製造本產品或文件的任何部分。至於協力廠商的軟體，包括字型技術，亦受著作權保護，並經過 Sun 供應商授權使用。

產品的某些部分可能源自 Berkeley BSD 系統，由加州大學授權。UNIX 是在美國和其他國家/地區的註冊商標，由 X/Open Company, Ltd. 獨家授權。

Sun、Sun Microsystems、Sun 標誌、docs.sun.com、AnswerBook、AnswerBook2、以及 Solaris 都是 Sun Microsystems, Inc. 在美國和其他國家/地區的商標或註冊商標。所有的 SPARC 商標都是在獲得授權的情況下使用，而且是 SPARC International, Inc. 在美國和其他國家/地區的商標或註冊商標。有 SPARC 商標的產品均基於 Sun Microsystems, Inc. 所開發的架構。

Sun 保有 Xerox 對 Xerox 圖形化使用者介面非獨佔性的授權，這項授權也涵蓋獲得 Sun 授權使用 OPEN LOOK GUI，或者符合 Sun 的書面軟體授權合約的廠商。Sun 認可 Xerox 研發電腦業之視覺化或圖形化使用者介面觀念的先驅貢獻。對於「Xerox 圖形化使用者介面」，Sun 保有來自於 Xerox 的非獨家授權，這項授權的適用也涵蓋取得 Sun 的授權而使用 OPEN LOOK GUI、或者遵循 Sun 的書面授權合約的廠商。

美國政府的權利 – 商業軟體。政府使用者要遵守 Sun Microsystems, Inc. 的標準授權合約以及 FAR 及其補遺的適當條款。

文件以「現狀」提供，所有明示或暗示的條件、陳述或保證，均恕不負責，此亦包括對於適銷性、特定用途的適用性或非侵權行為的任何暗示性保證在內，除非此免責聲明在法律上被認為無效。



050105@10536



目錄

- 1 介紹 9
 - 介紹 9

- 2 LDAP 伺服器 11
 - 概念 11
 - 設定 12
 - 部署工具 12
 - 模式延伸 12
 - 組織對映 12
 - 使用者設定檔對映 14
 - 部署 14
 - 其他注意事項 15

- 3 Sun Web Console 17
 - 系統需求 17
 - 用戶端 17
 - 伺服器 18
 - 安裝 Sun Web Console 18
 - ▼ 安裝 Sun Web Console 18
 - 執行主控台 19
 - 解除安裝 Sun Web Console 20
 - Sun Web Console 連接埠資訊 20

- 4 Java Desktop System Configuration Manager 發行版本 1.1 21
 - 安裝 Configuration Manager 21

▼ 若要安裝 Configuration Manager :	21
執行 Configuration Manager	22
▼ 若要啓動 Configuration Manager , 請 :	22
解除安裝 Configuration Manager	23
5 在 Linux 和 Solaris 上安裝桌面元件	25
Configuration Agent	25
啓動程式資訊	26
連接埠設定	28
變更偵測間隔	28
操作設定	29
套用代理程式設定	31
資料存取/使用者認證	31
GConf 配接卡	32
Java Preferences 配接卡	32
Mozilla 配接卡	33
StarSuite 配接卡	33
6 在 Microsoft Windows 上安裝桌面元件	35
Configuration Agent 安裝	35
歡迎使用	36
軟體許可授權合約	36
安裝類型	37
準備安裝	44
安裝摘要	45
變更 Configuration Agent 設定	46
取消安裝 Configuration Agent	46
A Sun Web Console 套裝軟體	47
已知問題	47
安全性	47
安裝程序檔用法	47
Sun Web Console 套裝軟體	48
Solaris 套裝軟體	48
Linux RPM	48

B	Configuration Manager 套裝軟體	51
	Configuration Manager 套裝軟體	51
	Solaris 套裝軟體	51
	Linux RPM	51
C	將 OpenLDAP 和 Active Directory 與 Configuration Manager 一起使用	53
	將 OpenLDAP 伺服器與 Configuration Manager 一起使用	53
	將 Active Directory 伺服器與 Configuration Manager 一起使用	54

圖清單

圖 5-1	Configuration Agent，配置儲存庫	26
圖 5-2	Configuration Agent，驗證機制	27
圖 5-3	Configuration Agent，連接埠設定	28
圖 5-4	Configuration Agent，資料目錄	29
圖 5-5	Configuration Agent，請求處理和記錄	29
圖 5-6	Configuration Agent，摘要頁面	31
圖 6-1	Configuration Agent，[歡迎使用] 頁面	36
圖 6-2	Configuration Agent，軟體許可授權合約	36
圖 6-3	Configuration Agent，安裝類型頁面	37
圖 6-4	Configuration Agent，配置儲存庫	38
圖 6-5	Configuration Agent，驗證機制	39
圖 6-6	Configuration Agent，連接埠設定	41
圖 6-7	Configuration Agent，資料目錄	42
圖 6-8	Configuration Agent，請求處理和記錄	42
圖 6-9	Configuration Agent，摘要頁面	43
圖 6-10	Configuration Agent，[準備安裝] 頁面	44
圖 6-11	Configuration Agent，[安裝摘要] 頁面	45

第 1 章

介紹

提供有關 Java™ Desktop System Configuration Manager 發行版本 1.1 的簡短介紹。

介紹

Java™ Desktop System Configuration Manager 發行版本 1.1 旨在為執行 Java™ Desktop System 的桌面主機提供中央配置。可以將設定指定給某組織結構的各種元素，使管理員可以簡單地管理使用者群組或主機群組。它的主元件包括：

- LDAP 伺服器，包括要管理之使用者和主機的組織結構，將會儲存配置資料，
- 網路型的管理工具，可讓管理員定義配置資料並將其指定至該組織結構的元素，
- 安裝在用戶端主機上的桌面元件，可代表目前登入的使用者擷取配置資料並將其傳給組成 Java Desktop System 的各種應用程式。

管理工具是在 Sun Web Console 中執行的網路型應用程式。它可讓管理員瀏覽 LDAP 伺服器的組織結構並為其元素指定策略。策略可依據策略範本來顯示和編輯，這些範本定義了管理工具將要處理的設定。

桌面元件以 Java™ Desktop System Configuration Agent 為中心進行組織，可代表使用者從 LDAP 伺服器中擷取配置資料，並將其提供給許多配置系統配接卡使用，這些配置系統配接卡使用策略設定補充本機配置 (應用程式的預設設定和使用者設定)。目前受支援的配置系統為 GConf (可處理 Gnome 應用程式 [如 Gnome 桌面或 Evolution] 的配置)、Mozilla™ Preferences 和 StarRegistry (StarSuite 的配置系統)。

第 2 章

LDAP 伺服器

本章提供了有關將 LDAP 伺服器設置和部署為與 Configuration Manager 一起使用的資訊。

概念

在 Java Desktop System Configuration Manager 架構中，配置資料與實體 (亦即 LDAP 儲存庫中的項目，它們與公司組織結構的元素相對應) 相關聯。

識別的實體包括：

- 組織：通常表示整個階層結構的組織單元 (部門、群組、團隊) 或地理單元 (大洲、國家/地區、站點)。
- 使用者：表示整個階層結構的葉節點，顧名思義，通常指使用者。
- 網域：表示網路組織的邏輯結構單元。
- 主機：也表示整個階層結構的葉節點，但指網路上的機器。
- 角色：表示屬性，通常指功能 (管理員和站點管理) 方面的區別，套用於一組使用者。

組織實體和使用者實體用於定義使用者樹，而網域實體和主機實體則定義主機樹。這兩種樹相互獨立，但在架構中的處理方法類似。

組織實體和網域實體與其他實體的關係由實體在儲存庫中的實際位置所定義。即組織實體和網域實體可包含樹中位於這兩個實體下面的任何項目。角色與使用者或主機的關係由使用者項目和主機項目的屬性所定義。

與實體關聯的配置資料儲存在由架構所管理的特殊項目中。這些項目可透過與項目關聯的服務名稱和服務容器來識別。

設定

若要將現有 LDAP 伺服器與 Configuration Manager 一起使用，您需要：

- 延伸伺服器模式，以支援 Configuration Manager 用來儲存配置資料所用的自訂物件類別和屬性。
- 在伺服器中自訂與儲存儲存庫中項目與 Configuration Manager 所支援的實體之間的對映資訊。

部署工具

將現有 LDAP 伺服器與 Configuration Manager 一起使用時需要使用安裝 CD 上的以下部署工具：

- 88apoc-registry.ldif：介紹儲存配置資料所需物件類別和屬性的模式檔案。
- OrganizationMapping：描述 LDAP 項目與 Configuration Manager 實體之間對映的預設特性檔案。
- UserProfileMapping：描述 LDAP 使用者項目屬性與 Configuration Manager 使用者設定檔屬性之間對映的預設特性檔案。
- createServiceTree：將對映檔案儲存在 LDAP 儲存庫中的程序檔。
- deployApoc：延伸 LDAP 伺服器模式並將對映檔案儲存在 LDAP 儲存庫中的程序檔。

模式延伸

配置資料儲存在與資料關聯的項目隨附的項目樹中。在 LDAP 伺服器上儲存這些樹所使用的物件類別和屬性之前，您必須將物件和類別加入 LDAP 伺服器模式。例如，提供的模式延伸檔案使用 LDIF 格式將這些物件和類別增加到 Sun Java™ System Directory Server 中。若要將這些物件和類別加入其他 LDAP 伺服器，您需要使用伺服器可識別的格式。

組織對映

若要定義 LDAP 項目和 Configuration Manager 實體之間的對映，則必須編輯 Organization 對映檔案。必須為各種鍵提供符合 LDAP 系統訊息庫佈局的值。

使用者實體可透過所有實體均使用的物件類別來識別，也可由其值在整個系統訊息庫中唯一的屬性識別。可提供顯示名稱格式以決定使用者在管理應用程式中的顯示方式，或者如果組織中的使用者項目使用容器項目，則亦可定義該類項目。鍵名稱及其預設值包括：

```

# Object class that all user entries use
User/ObjectClass=inetorgperson
# Attribute whose value in user entries is unique within the repository
User/UniqueIdAttribute=uid
# Optional container in organization entries of the user entries,
# remove line if not used
User/Container=ou=People
# Display name format within the management application
User/DisplayNameFormat=sn, givenname

```

角色實體可由其使用的可能物件類別清單以及相應的命名屬性清單來識別。這些清單使用格式 <item1>, <item2>, ..., <itemN> 並且必須對齊。即，各清單必須含有相同的項目數，且第 n 個物件類別必須與第 n 個命名屬性配合使用。兩個鍵定義角色與使用者之間的關係，以及角色與主機之間的關係。*VirtualMemberAttribute* 鍵必須從使用者項目或主機項目指定可查詢其值的屬性。該鍵還必須包含項目所從屬角色的完整 DN。*MemberAttribute* 鍵必須從使用者項目或主機項目為搜尋過濾器指定屬性。該鍵還必須包含使用者或主機所從屬角色的完整 DN。*VirtualMemberAttribute* 鍵可以是一種服務虛擬屬性，而 *MemberAttribute* 鍵則必須是可在過濾器中使用的實際屬性。鍵名稱及其預設值包括：

```

# List of object classes for roles
Role/ObjectClass=nsRoleDefinition
# Aligned list of corresponding naming attributes
Role/NamingAttribute=cn
# Physical attribute (usable in a filter) containing the DNs
# of the roles of a user/host
Role/MemberAttribute=nsRoleDN
# Attribute whose query on a user or host return the DNs of the
# roles it belongs to
Role/VirtualMemberAttribute=nsRole

```

識別組織實體的方法類似於識別角色的方法，即，使用兩個對齊清單 (物件類別清單和相應命名屬性清單)。鍵名稱及其預設值包括：

```

# List of object classes for organizations
Organization/ObjectClass=organization
# Aligned list of corresponding naming attributes
Organization/NamingAttribute=o

```

識別網域實體的方法類似於識別組織實體的方法。鍵名稱及其預設值包括：

```

# List of object classes for domains
Domain/ObjectClass=ipNetwork
# Aligned list of corresponding naming attributes
Domain/NamingAttribute=cn

```

識別主機實體的方法類似於識別使用者實體的方法。鍵名稱及其預設值包括：

```

# Object class that all host entries use
Host/ObjectClass=ipHost
# Attribute whose value in host entries is unique within the repository
Host/UniqueIdAttribute=cn
# Optional container in domain entries of the host entries,
# remove line if not used

```

Host/Container=ou=Hosts

使用者設定檔對映

若要定義 LDAP 使用者項目屬性與 Configuration Manager 使用者實體屬性之間的對映，則必須編輯使用者設定檔對映檔案。每個鍵均對應一個 Configuration Manager 使用者屬性。由於鍵已被組織對映識別，因此可被指定為使用者項目中屬性名稱的值。User/DisplayNameFormat 設定中所用的屬性必須在 *User Profile* 對映中指定。鍵名稱及其預設值包括：

```
# inetOrgPerson.givenName
org.openoffice.UserProfile/Data/givenname = givenname
# person.sn
org.openoffice.UserProfile/Data/sn = sn
# inetOrgPerson.initials
org.openoffice.UserProfile/Data/initials = initials
# organizationalPerson.street
org.openoffice.UserProfile/Data/street = street,postalAddress,streetAddress
# organizationalPerson.l (city)
org.openoffice.UserProfile/Data/l = l
# organizationalPerson.st (state)
org.openoffice.UserProfile/Data/st = st
# organizationalPerson.postalCode
org.openoffice.UserProfile/Data/postalcode = postalcode
# country.c (country)
org.openoffice.UserProfile/Data/c =
# organizationalPerson.o (company)
org.openoffice.UserProfile/Data/o = o,organizationName
# deprecated -- no LDAP corollary
org.openoffice.UserProfile/Data/position =
# organizationalPerson.title
org.openoffice.UserProfile/Data/title = title
# inetOrgPerson.homePhone
org.openoffice.UserProfile/Data/homephone = homephone
# organizationalPerson.telephoneNumber
org.openoffice.UserProfile/Data/telephonenumber = telephonenumber
# organizationalPerson.facsimileTelephoneNumber
org.openoffice.UserProfile/Data/facsimiletelephonenumber =
facsimiletelephonenumber,officeFax
# inetOrgPerson.mail
org.openoffice.UserProfile/Data/mail = mail
```

部署

已將對映檔案自訂為反映 LDAP 儲存庫的狀態之後，即可對其進行部署。如果 LDAP 伺服器模式已包含所需的物件類別和屬性，則可以直接執行程序檔 `createServiceTree`，否則必須執行程序檔 `deployApoc`。

`deployApoc` 程序檔旨在與 Sun Java™ System Directory Server 一起使用。它會將提供的模式延伸檔案複製到適當的目錄並重新啟動 LDAP 伺服器，然後呼叫 `createServiceTree` 程序檔。必須將它做為具有在模式系統訊息庫中複製檔案和重新啟動伺服器之許可權的使用者來執行，並且必須透過以下指令啟動它：

```
./deployApoc <Directory Server Directory>
```

<Directory Server Directory> 參數必須是目錄伺服器安裝的 slapd-*<server name>* 子目錄的路徑。假定安裝使用預設目錄，並且伺服器名為 *myserver.mydomain*，則該目錄將是 */var/Sun/mps/slapd-myserver.mydomain*。

createServiceTree 程序檔無論被直接呼叫，還是透過 deployApoc 程序檔呼叫，均將提示使用者提供 LDAP 伺服器 (主機名稱、連接埠號和基底 DN) 的位置以及擁有管理權限的使用者的定義 (完整 DN 和密碼)。然後程序檔會在 LDAP 伺服器中建立啟動程式服務樹並在其中儲存對映檔案。可將其做為任何使用者執行，並可透過以下指令來啟動它：

```
./createServiceTree
```

然後提示使用者輸入：

- [主機名稱] (預設值：**localhost**)：LDAP 伺服器的主機名稱，
- [連接埠號] (預設值：**389**)：LDAP 伺服器的連接埠號，
- [基底 DN]：LDAP 系統訊息庫的基底 DN，
- [使用者 DN] (預設值：**cn=Directory Manager**)：具有足夠權限以在基底 DN 下建立新項目之使用者的完整 DN，
- [密碼]：該使用者的密碼，

已建立 DN 為：

```
ou=ApocRegistry,ou=default,ou=OrganizationConfig,ou=1.0,  
ou=ApocService,ou=services, <baseDN>
```

的項目，並使用兩個對映檔案的內容來填充。

如前所述，deployApoc 程序檔執行的作業假定了一個安裝目錄、佈局和模式延伸程序與 Sun Java System Directory Server 的對應項很相似的 LDAP 伺服器。在能夠執行 createServiceTree 程序檔之前，其他目錄將需要對模式進行手動延伸。如需有關使用 OpenLDAP 和 ActiveDirectory 的進一步資訊，請參閱附錄 C。

建立的樹 (與將要儲存實體關聯的配置資料的樹匹配) 會與在 Sun Java 系統識別伺服器中用於服務管理的樹在結構上對齊。

其他注意事項

Configuration Manager 架構需要能夠建立連線至 LDAP 伺服器，且該連線要具有讀取和搜尋許可權，以便識別哪個完整 DN 與來自桌面的特定使用者或主機識別碼關聯。為此，必須配置系統訊息庫，以允許匿名連接，或者必須建立具有讀取和搜尋存取權的特殊使用者。

管理應用程式會在對映至實體的項目下建立服務樹，以儲存這些實體的配置資料。因此，用於管理的使用者項目需要具有在其管理的項目之下建立子項目的權限。

使用名稱爲 `Anonymous` 和 `GSSAPI` 的兩種方法，可從桌面用戶端認證架構的使用者。由於桌面用戶端在嘗試從 LDAP 伺服器擷取資料時將不提供任何憑證，因此 `Anonymous` 方法需要在整個儲存庫中啟用讀取和搜尋的匿名存取。若要使用 `GSSAPI` 方法 (使用 Kerberos 來進行認證)，必須按照「*Sun Java™ System Directory Server 5 2004Q2 Administration Guide*」中的「Managing Authentication and Encryption」一章所述配置 LDAP 伺服器。

第 3 章

Sun™ Web Console

Sun Web Console 的設計旨在產生用於 Sun Microsystems 的共用、網路型系統管理解決方案。使用者可以從中存取系統管理應用程式，而所有應用程式均提供一致的使用者介面。

主控台基於 Web 模型出於多種原因。但主要原因是讓系統管理員能夠使用 Web 瀏覽器來存取其系統管理應用程式。

Sun Web Console 提供：

- 共用認證與授權
- 共用記錄
- 單一進入點，即透過基於 HTTPS 的同一連接埠存取所有系統管理應用程式
- 共用的外觀和感覺

主控台的主要優勢是管理員僅登入一次即可使用主控台內部的任何應用程式。

系統需求

Sun Web Console 支援多用戶端、伺服器作業系統以及數種瀏覽器。

用戶端

- Netscape™ 6.2x 和 7.x (對於 Solaris™ 8 或更高版本)
- Netscape 4.7x、6.2x 和 7.x (對於 Windows 98、98 SE、ME、2000 和 XP)
- Internet Explorer 5.x 和 6.x (對於 Windows 98、98 SE、ME、2000 和 XP)
- Mozilla (對於 Linux 和 Solaris)

伺服器

- Solaris 8 或更高版本
- Red Hat 8 或更高版本、Red Hat Enterprise Linux 2.1
- SuSE Linux 2.1 或更高版本
- J2SE™ 版本 1.4.1_03 或更高版本
如果在您的伺服器上偵測到 J2SE 1.4.1 或更早版本，安裝程式會提示您使用 Java Desktop System Management Tools CD 中的 J2SE 版本來升級安裝。
- Tomcat : 4.0.3 或更高版本
Tomcat 包括在 Java Desktop System Management Tools CD 中

安裝 Sun Web Console

安裝 Sun Web Console 之前，請先閱讀本指南附錄 A 中的套裝軟體摘要與已知問題小節。

Java Desktop System Management Tools CD 中有 Solaris SPARC (版本 8 或更高) 與 Linux 作業系統的 Sun Web Console 二進位安裝檔案。

▼ 安裝 Sun Web Console

- 步驟
1. 在 Java Desktop System Management Tools CD 上，變更至與您要安裝主控台之作業系統相對應的 Sun Web Console 目錄 (「swc」)。
 2. 鍵入 `./setup`
依預設，Sun Web Console 將不建立安裝日誌檔。若要建立一個名為「logfile」的安裝日誌，請鍵入 `./setup | tee logfile`。

注意 – 當您執行安裝程式時，Web Console 的大部分安裝與配置會自動執行。如需有關 Sun Web Console 的安裝應用程式的更多詳細資訊，請參閱附錄 A。

3. 如果您要本地化 Sun Web Console，則需為每種語言安裝兩個附加套裝軟體。請使用下表確定每種語言使用的套裝軟體名稱，然後執行下列步驟之一：
 - 在 Solaris 上，鍵入 `pkgadd -d path/pkgname.pkg pkgname`，其中 `pkgname` 是您要增加的語言套裝軟體的名稱。
 - 在 Linux 上，鍵入 `rpm -i path/pkgname<...>.rpm`，其中 `pkgname` 是您要增加的語言套裝軟體的名稱。

套裝軟體名稱	說明
SUNWcmcon、SUNWcmctg	簡體中文 Sun™ Web Console 2.0
SUNWdmcon、SUNWdmctg	德語 Sun™ Web Console 2.0
SUNWemcon、SUNWemctg	西班牙語 Sun™ Web Console 2.0
SUNWfmcon、SUNWfmctg	法語 Sun™ Web Console 2.0
SUNWhmcon、SUNWhmctg	繁體中文 Sun™ Web Console 2.0
SUNWimcon、SUNWimctg	義大利語 Sun™ Web Console 2.0
SUNWjmcon、SUNWjmctg	日文 Sun™ Web Console 2.0
SUNWkmcon、SUNWkmctg	韓文 Sun™ Web Console 2.0
SUNWsmcon、SUNWsmctg	瑞典語 Sun™ Web Console 2.0

執行主控台

如果您要註冊新的應用程式，通常僅需要停止並重新啟動 Sun Web Console 伺服器。



注意 – 在您首次啟動 Sun Web Console 之前，請確定 Configuration Manager 安裝已完成。必須在主控台中至少部署一個應用程式，Sun Web Console 才能成功運行。

- 若要啟動 Sun Web Console，請鍵入 **smcwebserver start**。
- 若要停止 Sun Web Console，請鍵入 **smcwebserver stop**。
- 若要存取 Sun Web Console，請在瀏覽器中輸入如下 URL：`https://<hostname>.<domainname>:6789`

依預設，Sun Web Console 支援基於 Unix 的驗證和基於角色的存取控制 (RBAC)。然而，您也可以配置其他驗證機制，如 LDAP 驗證。

注意 – 預設工作時段逾時為 15 分鐘。您可以使用 **smreg** 指令配置逾時長度。例如，若要將逾時長度設定為 5 分鐘，請鍵入 **smreg add -p -c session.timeout.value=5**。

如需有關用於 Sun Web Console 的指令的更多資訊，請參閱 **smcwebserver** 和 **smreg** 線上說明手冊。

解除安裝 Sun Web Console

若要解除安裝 Sun Web Console，請執行 `/usr/lib/webconsole/setup -u`。

注意 – 如果您正位於 `/usr/lib/webconsole` 目錄或任何相關子目錄中，請勿執行此指令，否則 `pkgrm` 將會失敗。

Sun Web Console 連接埠資訊

Configuration Manager 使用 Sun Web Console 的連接埠：

- 8005 用於關閉服務，而
- 6789 用於存取 https。

可在 `/etc/opt/webconsole/server.xml` 中變更這兩個連接埠。變更連接埠後，請使用 `/usr/sbin/smcwebserver restart` 重新啓動 Sun Web Console。

第 4 章

Java™ Desktop System Configuration Manager 發行版本 1.1

Configuration Manager 提供在 Sun Web Console 上執行的一種管理工具。這個基於 Web 的使用者介面可讓管理員遍歷組織的階層結構，以定義用於桌面應用程式的策略。可以為階層結構中的每個項目 (如組織、角色、使用者、網域和主機) 定義這些策略。Configuration Manager 使用數個配置範本顯示專用於不同桌面應用程式 (如 Gnome、Mozilla、StarSuite 和 Evolution) 的設定。

安裝 Configuration Manager

安裝 Configuration Manager 之前，您需要安裝有效的 Sun Web Console。

▼ 若要安裝 Configuration Manager：

- 步驟
1. 變更至 Java Desktop System Management Tools CD 上相應的 Configuration Manager 目錄 (「apoc」)。
 2. 鍵入 `./setup`。
 3. 輸入 LDAP 伺服器的主機名稱。
預設名稱為 `localhost`。
 4. 輸入 LDAP 伺服器的連接埠號 (預設為：389)。
 5. 輸入 LDAP 系統訊息庫的基底 DN。
 6. 輸入用於識別使用者實體之物件類別的名稱。預設物件類別為 `inetorgperson`。
如需更多詳細資訊，請參閱「LDAP 伺服器」一章中的第 12 頁的「組織對映」。

7. 輸入在整個 LDAP 系統訊息庫中唯一的屬性名稱。預設屬性為 `uid`。
如需更多詳細資訊，請參閱「LDAP 伺服器」一章中的第 12 頁的「組織對映」。
8. 輸入具有在 LDAP 伺服器上執行查詢的必要存取權之使用者的完整 DN。
使用具有讀取和搜尋存取權的任何完整 DN。對於匿名存取，請保留此欄位為空白。
9. 輸入您為其指定了 LDAP 存取權限之使用者的密碼。
如果您已設定匿名存取 LDAP 伺服器，請忽略此步驟。
安裝期間，一個附加登入模組將被增加至 Sun Web Console，可讓您透過 LDAP 驗證使用者。
在安裝的最後階段，Sun Web Console 將自動重新啟動，以便您可以存取 Configuration Manager。

注意 – 您可以使用 `/usr/share/webconsole/apoc/configure` 程序檔隨時修改先前的 Configuration Manager 設定。例如，您可以使用該程序檔變更至不同的 LDAP 伺服器，而無需重新安裝 Configuration Manager。

執行 Configuration Manager

▼ 若要啟動 Configuration Manager，請：

- 步驟
1. 在瀏覽器中鍵入以下 URL：`https://<hostname>.<domainname>:6789`
 2. 在提示處，鍵入使用者名稱 (`uid`) 和現有 LDAP 使用者的密碼。
Sun Web Console 將會開啓。
 3. 在主控台視窗中，按一下 Sun Java Desktop System Configuration Manager 發行版本 1.1。

更多資訊 直接存取 Configuration Manager

若要略過 Sun Web Console 的啟動頁面而直接進入 Configuration Manager，請在瀏覽器中輸入以下 URL：

`https://<hostname>.<domainname>:6789/apoc`

解除安裝 Configuration Manager

若要從 Sun Web Console 解除安裝 Configuration Manager，請變更至 Java Desktop System Management Tools CD 中相應的 Configuration Manager 目錄，然後執行 `./setup -u`。

注意 – 解除安裝 Configuration Manager 時，LDAP 登入模組將從 Sun Web Console 中移除。

第 5 章

在 Linux 和 Solaris™ 上安裝桌面元件

本章提供了有關 Solaris 和 Linux 作業系統的特定資訊。

若要從 Configuration Manager 中存取配置資料，桌面用戶端需要 Java™ Desktop System Configuration Agent。Configuration Agent 可與遠端配置資料儲存庫和配接卡進行通訊，也可將資料整合於特定的配置系統中。目前支援的配置系統為 GConf、Java Preferences、Mozilla Preferences 和 StarSuite Registry。

Configuration Agent

Configuration Agent 是下表所列出的其他一些套裝軟體的一部分。

Solaris 套裝軟體名稱	Linux RPM 名稱	說明
SUNWapbas	apoc-base	配置共用程式庫
SUNWapmsc	apoc-misc	Configuration Agent 其他檔案
SUNWapoc	apoc	Configuration Agent
SUNWapdc	apoc-config	Configuration Agent 精靈

安裝這些套裝軟體時，該 API 所需的檔案都會安裝。您可以手動安裝這些套裝軟體或經由 Java Desktop System 進行安裝。安裝後，您必須在系統中配置並啟用 Configuration Agent。

若要存取遠端配置資料，Configuration Agent 需要最基本的啟動程式資訊，例如 LDAP 伺服器的主機名稱和連接埠。本資訊維護在一組特性檔案中，例如 `polycmgr.properties`、`apocd.properties` 和 `os.properties`。這些檔案以本機方式儲存在 `/etc/apoc` 目錄中。您可以手動編輯這些特性檔案，或者使用 Configuration Agent 的配置精靈。

配置精靈提供一個圖形化使用者介面，可指導您瞭解 Configuration Agent 的必要設定。精靈的每一個頁面都會有相應的說明螢幕。您可以使用 `/usr/bin/apoc-config` 程序檔，以超級使用者身份 (root) 啟動精靈。在 [喜好設定]/[系統工具]/[網路設定] 下，或 Nautilus 檔案管理員中的 [系統設定]://[網路設定] 下，也有相應的桌面功能表項目。

注意 – 也可以不啟動圖形介面而啟動該精靈。例如，在主控台模式下執行 `/usr/bin/apoc-config -nodisplay` 以啟動精靈。

啟動程式資訊

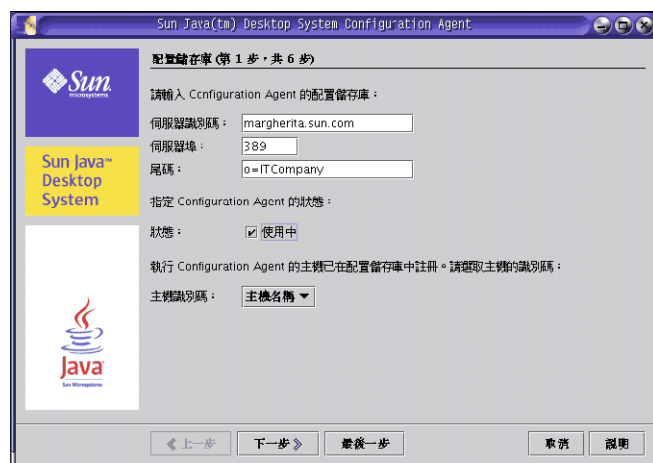


圖 5-1 Configuration Agent，配置儲存庫

注意 – 在適當的位置，關聯的特性檔案關鍵字在圓括號中進行指示。

- [伺服器識別碼] (Server)：LDAP 伺服器的主機名稱。
- [伺服器連接埠] (Port)：LDAP 伺服器的連接埠號。
- [尾碼] (基底 DN)：LDAP 儲存庫的基底 DN。
- [狀態]：Configuration Agent 的狀態。該核取方塊可用於啟動或關閉 Configuration Agent。若要使用配置儲存庫，Configuration Agent 必須處於使用中。自動啟動包括使用 `inetd` 進行必要的註冊。

注意 – 若要手動啟用或停用 Configuration Agent，請以**超級使用者**身份登入，並分別鍵入指令 `/usr/lib/apocd enable` 或 `/usr/lib/apocd disable`。



圖 5-2 Configuration Agent，驗證機制

- **[主機識別碼] (HostIdentifier)**：可為「主機名稱」或「IP 位址」。該識別碼必須設定為與用於識別主機之 LDAP 屬性的內容相符。此屬性在對映檔案中定義為 `Host/UniqueIdAttribute`。
- Configuration Agent 的 **[驗證類型]**：可為「匿名」或「簡單」。如果選取「匿名」，**[合格的使用者名稱]** 和 **[密碼]** 欄位會被自動停用。
- **[合格的使用者名稱] (AuthDn)**：具有對儲存庫的讀取和搜尋存取權之使用者的完整 DN。
- **密碼 (Password)**：已註冊 LDAP 使用者的密碼

注意 – 如果已在目錄中啟用匿名存取，則**合格的使用者名稱**和**密碼**設定可以為空。

- 應用程式的 **[驗證類型] (AuthType)**：依據 LDAP 伺服器驗證使用者的方法，可為「匿名」或「GSSAPI」。

注意 – 如需更多資訊，請參閱第 31 頁的「資料存取/使用者認證」。

連接埠設定

Configuration Agent 使用兩個連接埠：

- [代理程式連接埠] (DaemonPort)：由代理程式用來與用戶端應用程式進行通訊 (預設值為 38900)。
- [管理連接埠] (DaemonAdminPort)：由代理程式控制器程式 apocd 在與代理程式通訊時使用 (預設值為 38901)。



圖 5-3 Configuration Agent，連接埠設定

變更偵測間隔

Configuration Agent 使用以下兩種間隔定期檢查配置資料中的所有變更：

- [一般偵測間隔] (ChangeDetectionInterval)：桌面應用程式的 (用戶端的) 配置資料之變更偵測循環的間隔 (以分鐘為單位)。

注意 – 指定 -1 會關閉變更偵測。

- [代理程式設定的間隔] (DaemonChangeDetectionInterval)：代理程式特定配置設定之變更偵測循環的間隔 (以分鐘為單位)。

注意 – 指定 -1 會關閉變更偵測。

您可以使用一般偵測間隔來調準遠端配置資料變更至用戶端應用程式的傳遞。為該設定提供的值，是遠端所做變更在用戶端應用程式中反映出來之前所需的最大時間長度 (以分鐘為單位)。

較小的值將會導致 Configuration Agent 和 LDAP 伺服器活動的增加。因此，調整設定值時應該謹慎。例如，在初始部署階段中，您可以將值設定為一分鐘，以便可以輕鬆地測試遠端配置對用戶端應用程式的影響。完成測試之後，請將此設定恢復為初始值。

操作設定



圖 5-4 Configuration Agent，資料目錄

可以配置以下設定：

- [資料目錄] (DataDir)：用於儲存運行時間資料的目錄。預設為 `/var/opt/apoc`。
- [快取的資料儲存生命] (TimeToLive)：非離線配置資料在本機資料庫中保留的間隔 (以分鐘為單位)。



圖 5-5 Configuration Agent，請求處理和記錄

- [資源回收循環] (GarbageCollectionInterval)：本機配置資料庫中資源回收循環的間隔 (以分鐘為單位)。
- [最大用戶端執行緒數] (MaxClientThreads)：可同時處理的用戶端請求的最大數目。
- [最大用戶端連線數] (MaxClientConnections)：用戶端連線的最大數目。
- [最大請求大小] (MaxRequestSize)：用戶端要求的最大大小。
- [連線逾時] (ConnectTimeout)：表示 LDAP 伺服器回覆連線請求所允許的間隔。預設值為 1 秒。
- [日誌層級] (LogLevel)：代理程式日誌檔中詳細資訊的級別。日誌層級與 Java 記錄程式層級一致。這些層級依嚴重性遞減的次序為：
 - [嚴重]
 - [警告]
 - [資訊]
 - [配置]
 - [詳細]
 - [較詳細]
 - [最詳細]

注意 – 大多數操作設定，除 [資料目錄] 和 [連線逾時] 設定之外，也可以透過儲存於 LDAP 伺服器中的相應策略進行集中維護。如果您要使用此項功能，請勿透過精靈調整相應的設定。而是使用 Configuration Manager 中的 Configuration Agent 策略來集中指定操作設定。

套用代理程式設定

除「資料目錄」和「連線逾時」之外，透過 Configuration Manager 儲存於 LDAP 伺服器上的操作設定將會在代理程式配置的下一階段變更偵測循環開始時自動生效 (請參閱 DaemonChangeDetectionInterval)。



圖 5-6 Configuration Agent，摘要頁面

本機變更的所有其他設定均需要重新載入或重新啟動 Configuration Agent。如果您使用配置精靈，重新載入或重新啟動則會自動執行。

注意 – 若要手動重新啟動 Configuration Agent，請確定未執行相關用戶端應用程式，以超級使用者身份登入，然後鍵入指令 `/usr/lib/apoc/apocd restart`。

資料存取/使用者認證

Configuration Agent 根據桌面使用者的登入 ID 從 LDAP 伺服器中擷取資訊。組織對映檔案的 User/UniqueIdAttribute 設定將登入 ID 對映至 LDAP 伺服器中的使用者實體。Configuration Agent 還擷取有關主機的資訊，例如主機的名稱或 IP 位址。此資訊經由組織對映檔案的 Host/UniqueIdAttribute 設定對映至 LDAP 伺服器中的主機實體。

存取 LDAP 伺服器有兩種方法，即匿名存取或使用 GSSAPI 存取。對於匿名存取，不需要在桌面上執行動作。對於 GSSAPI 方法，則必須在桌面上獲取 Kerberos 憑證。若要整合獲取的 Kerberos 憑證和使用者登入，則必須在 Java Desktop System 主機上安裝和配置 pam_krb5 模組。

您可以使用 `gdm` 來整合 Kerberos 和使用者登入，例如，透過使用下列 `/etc/pam.d/gdm` 檔案：

```
##PAM-1.0
auth    required    pam_unix2.so  nullok #set_secrpc
auth    optional    pam_krb5.so  use_first_pass missing_keytab_ok ccache=SAFE putenv_direct
account required    pam_unix2.so
password required    pam_unix2.so  #strict=false
session required    pam_unix2.so  # trace or none
session required    pam_devperm.so
session optional    pam_console.so
```

如果您按照此方法整合 Kerberos 和使用者登入，則應啟用螢幕保護程式的 Kerberos 支援。例如，透過使用下列 `/etc/pam.d/xscreensaver` 檔案：

```
auth required pamkrb5.so use_first_pass missing_keytab_ok
ccache=SAFE putenv_direct
```

GConf 配接卡

GConf 配接卡是 `SUNWapoc-adapter-gconf` 套裝軟體 (適用於 Solaris) 和 `apoc-adapter-gconf RPM` (適用於 Linux) 的一部分。從相應的套裝軟體或 RPM 安裝配接卡時，`/etc/gconf/2/path` 中的 GConf 資料源路徑會更新，以包含 Configuration Manager 資源。配接卡提供的兩種資料來源包括：

- 「`apoc:readonly:`」：提供對策略中不受保護設定的存取權限。請在使用者設定與本機預設值之間插入此資料來源。
- 「`apoc:readonly:mandatory@`」：提供對策略中受保護設定的存取權限。請在本機強制設定與使用者設定之間插入此資料來源。

Java Preferences 配接卡

Java Preferences 配接卡是 `SUNWapcj` 套裝軟體 (適用於 Solaris) 和 `apoc-adapter-java RPM` (適用於 Linux) 的一部分。從相應的套裝軟體或 RPM 安裝配接卡時，所需檔案會被加入 `/opt/SUNWapcj` 目錄 (位於 Solaris 上) 或 `/opt/apocjava` (位於 Linux 上)。

Mozilla 配接卡

Mozilla 配接卡是 `SUNWmozapoc-adapter` 套裝軟體 (適用於 Solaris) 和 `mozilla-apoc-integration RPM` (適用於 Linux) 的一部分。從相應的套裝軟體或 RPM 安裝配接卡時，所需檔案會被加入 Mozilla 的現有安裝中，並會被自動註冊。

StarSuite 配接卡

StarSuite 配接卡包含在標準的 StarSuite 安裝中，可讓您存取策略配置資料，而不需要任何特殊修改。

第 6 章

在 Microsoft Windows 上安裝桌面元件

中央配置功能必需的所有元件均做為 Java Desktop System 的一部分提供與安裝。一般情況下，您可以在其他系統上使用中央配置管理。但是，此時您必須親自執行必需的安裝步驟。為了簡化在 Microsoft Windows 系統下的安裝，為您提供了基於 GUI 的安裝程式，它會指導您完成整個安裝步驟。

Configuration Agent 安裝

注意 – 安裝程式以及 Configuration Agent 都需要安裝有 Java™ Runtime Environment (JRE) 1.4 或更高版本。JRE 不是安裝的一部分。如果您不確定所安裝的 Java 版本，請執行 `java -version`。該指令將告知您系統中是否有正確的 Java 版本可用，以及是否對該版本進行了適當配置。

安裝程式透過批次檔 `setup.bat` 啟動。該檔案位於 Java Desktop System 管理工具 CD 中。您必須以具有足夠管理權限的使用者身份 (**local administrator**) 來啟動該程序檔。否則，無法將 Configuration Agent 做為新的系統服務來安裝與啟動。

鍵入以下指令來啟動安裝程式

```
setup [-help]
      [-saveState 檔案名稱] [-no]
      [-noconsole] [-state 檔案名稱]
      [-nodisplay]
```

安裝程式會識別下列選項：

- help 顯示相應說明文字。
- saveState 將互動式安裝工作時段的輸入寫入指定檔案。預設的檔案名稱爲 statefile.out。
- state 將指定的狀態檔案用作無訊息安裝的輸入。該選項與 -noconsole 一同使用。
- noconsole 啓動安裝程式，而無需任何使用者介面。與 -state 一同使用來啓動無訊息安裝。
- nodisplay 以指令行模式啓動安裝程式，而無需啓動圖形化使用者介面。
- no 運行安裝程式，而無需安裝軟體。與 -saveState 一同使用時，該選項對準備無訊息安裝很有用。

歡迎使用

呼叫 Configuration Agent 安裝精靈後，螢幕上會出現 [歡迎使用] 頁面。[歡迎使用] 頁面爲您提供有關安裝程序的資訊。

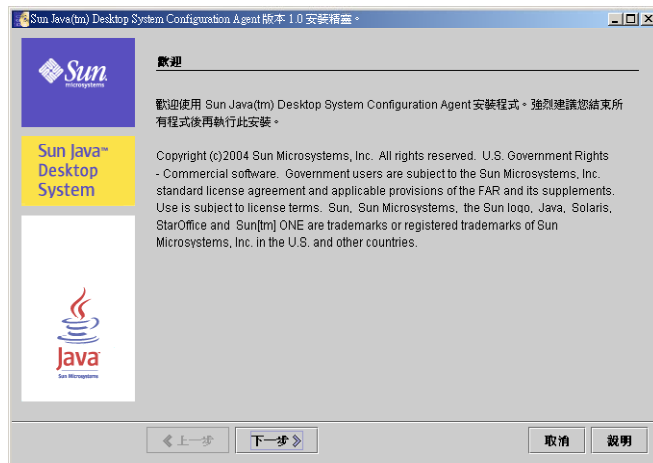


圖 6-1 Configuration Agent，[歡迎使用] 頁面

按一下 [下一頁] 按鈕以前進至精靈的下一頁。

軟體許可授權合約

[軟體許可授權合約] 頁面包含有關 Configuration Agent 的授權資訊。



圖 6-2 Configuration Agent，軟體許可授權合約

請閱讀軟體許可授權合約，然後按一下 [是 (接受授權合約)] 按鈕以接受並繼續執行安裝程序。

安裝類型

安裝程式提供三種安裝類型。這些類型僅是在安裝程序中可以定義的配置設定數目上有所不同。



圖 6-3 Configuration Agent，安裝類型頁面

- **快速**：建議大多數使用者選擇快速類型。安裝程式僅顯示 Configuration Agent 工作安裝必需的設定。其餘所有設定均設定為適當的預設值。
- **自訂**：安裝程式顯示 Configuration Agent 的所有可用設定，從而讓您可以對代理程式的運作方式進行微調。僅建議進階使用者選擇該類型。
- **最小**：安裝程式會安裝 Configuration Agent，但不會提示您提供任何配置。您必須在完成安裝後，透過啟動配置精靈來配置 Configuration Agent。

選取所需的安裝類型後，按一下 [下一頁]。依照所選安裝類型，安裝程式會顯示若干配置設定頁面。以下小節提供對這些設定的描述。

注意 – 如果您選取的是**快速**或**最小**選項，則以下有些配置設定頁面並不適用。

注意 – 若要啟動 Configuration Agent，則需要一些啟動程式資訊 (例如主機名稱和 LDAP 伺服器的連接埠)。此外，您還可以定義若干進階設定 (例如記錄層級或資料目錄)。所有這些設定都可以在安裝程序中指定。本資訊維護在一組特性檔案中，例如 `policymgr.properties`、`apocd.properties` 和 `os.properties`。這些檔案儲存於本機的程式檔案 `\Sun\Apoc\lib` 目錄中。您可以手動編輯這些特性檔案，或者使用 Configuration Agent 的配置精靈。

Configuration Agent 精靈與設定資訊

本小節精確描述了 Configuration Agent 可用的所有配置設定。

配置精靈提供一個圖形化使用者介面，可指導您瞭解 Configuration Agent 的必要設定。精靈的每一個頁面都會有相應的說明螢幕。

啟動程式資訊

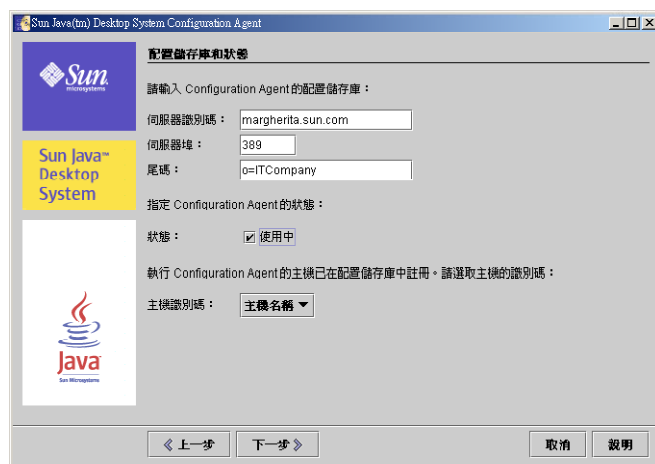


圖 6-4 Configuration Agent，配置儲存庫

注意 – 在適當的位置，關聯的特性檔案關鍵字在圓括號中進行指示。

- **[伺服器識別碼]** (Server)：LDAP 伺服器的主機名稱。
- **[伺服器連接埠]** (Port)：LDAP 伺服器的連接埠號。
- **[尾碼]** (基底 DN)：LDAP 儲存庫的基底 DN。
- **[狀態]**：Configuration Agent 的狀態。該核取方塊可用於啟動或關閉 Configuration Agent。若要使用配置儲存庫，Configuration Agent 必須處於使用中。啟動會自動包含使用服務控制管理程式所進行的必要註冊。

注意 – 若要手動啟用或停用 Configuration Agent，請以管理員身份登入，並分別鍵入指令程式檔案 `\Sun\Apoc\bin\apocd enable` 或程式檔案 `\Sun\Apoc\bin\apocd disable`。

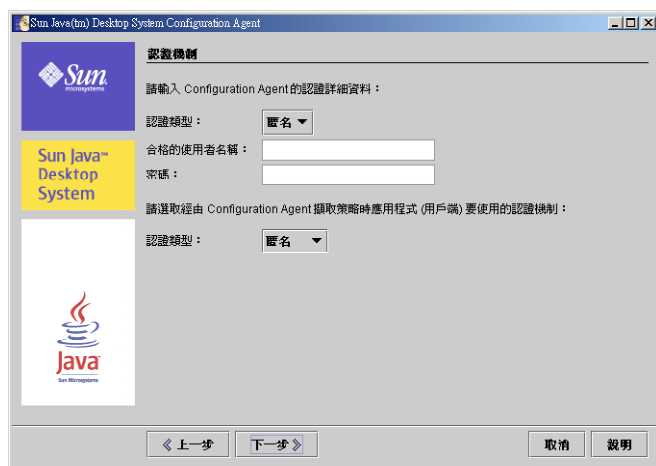


圖 6-5 Configuration Agent，驗證機制

- [主機識別碼] (HostIdentifier)：可為「主機名稱」或「IP 位址」。該識別碼必須設定為與用於識別主機之 LDAP 屬性的內容相符。此屬性在對映檔案中定義為 Host/UniqueIdAttribute。
- Configuration Agent 的 [驗證類型]：可為「匿名」或「簡單」。如果選取「匿名」，[合格的使用者名稱] 和 [密碼] 欄位會被自動停用。
- [合格的使用者名稱] (AuthDn)：具有對儲存庫的讀取和搜尋存取權之使用者的完整 DN。

注意 – 如果在此目錄中啟用了匿名存取，可將此設定保留為空白。

- 密碼 (Password)：已註冊 LDAP 使用者的密碼

注意 – 如果在此目錄中啟用了匿名存取，可將此設定保留為空白。

- 應用程式的 [驗證類型] (AuthType)：依據 LDAP 伺服器驗證使用者的方法，可為「匿名」或「GSSAPI」。

連接埠設定

Configuration Agent 使用兩個連接埠：

- [代理程式連接埠] (DaemonPort)：由代理程式用來與用戶端應用程式進行通訊 (預設為 38900)。

- **[管理連接埠] (DaemonAdminPort)**：由代理程式控制器程式 `apocd.exe` 在與代理程式通訊時使用 (預設值為 **38901**)。



圖 6-6 Configuration Agent，連接埠設定

變更偵測間隔

Configuration Agent 使用以下兩種間隔定期檢查配置資料中的所有變更：

- **[一般偵測間隔] (ChangeDetectionInterval)**：桌面應用程式的 (用戶端的) 配置資料之變更偵測循環的間隔 (以分鐘為單位)。

注意 – 指定 **-1** 會關閉變更偵測。

- **[代理程式設定的間隔] (DaemonChangeDetectionInterval)**：代理程式特定配置設定之變更偵測循環的間隔 (以分鐘為單位)。

注意 – 指定 **-1** 會關閉變更偵測。

您可以使用一般偵測間隔來調準遠端配置資料變更至用戶端應用程式的傳遞。為該設定提供的值，是遠端所做變更在用戶端應用程式中反映出來之前所需的最大時間長度 (以分鐘為單位)。

較小的值將會導致 Configuration Agent 和 LDAP 伺服器活動的增加。因此，調整設定值時應該謹慎。例如，在初始部署階段中，您可以將值設定為一分鐘，以便可以輕鬆地測試遠端配置對用戶端應用程式的影響。完成測試之後，請將此設定恢復為初始值。

操作設定



圖 6-7 Configuration Agent，資料目錄

可以配置以下設定：

- [資料目錄] (DataDir)：用於儲存運行時間資料的目錄。
- [快取的資料儲存生命] (TimeToLive)：非離線配置資料在本機資料庫中保留的間隔 (以分鐘為單位)。



圖 6-8 Configuration Agent，請求處理和記錄

- [資源回收循環] (GarbageCollectionInterval)：本機配置資料庫中資源回收循環的間隔 (以分鐘為單位)。

- [最大用戶端執行緒數] (MaxClientThreads)：可同時處理的用戶端請求的最大數目。
- [最大用戶端連線數] (MaxClientConnections)：用戶端連線的最大數目。
- [最大請求大小] (MaxRequestSize)：用戶端要求的最大大小。
- [連線逾時] (ConnectTimeout)：表示 LDAP 伺服器回覆連線請求所允許的間隔。預設值為 1 秒。
- [日誌層級] (LogLevel)：代理程式日誌檔中詳細資訊的級別。日誌層級與 Java 記錄程式層級一致。這些層級依嚴重性遞減的次序為：
 - [關閉]
 - [嚴重]
 - [警告]
 - [資訊]
 - [配置]
 - [詳細]
 - [較詳細]
 - [最詳細]
 - [全部]

注意 – 大多數操作設定，除 [資料目錄] 和 [連線逾時] 設定之外，也可以透過儲存於 LDAP 伺服器中的相應策略進行集中維護。如果您要使用此項功能，請勿透過精靈調整相應的設定。而是使用 Configuration Manager 中的 Configuration Agent 策略來集中指定操作設定。

代理程式設定摘要

除「資料目錄」和「連線逾時」之外，透過 Configuration Manager 儲存於 LDAP 伺服器上的操作設定將會在代理程式配置的下一次變更偵測循環開始時自動生效 (請參閱 DaemonChangeDetectionInterval)。



圖 6-9 Configuration Agent，摘要頁面

本機變更的所有其他設定均需要重新載入或重新啓動 Configuration Agent。如果您使用配置精靈，重新載入或重新啓動則會自動執行。

注意 – 若要手動重新啓動 Configuration Agent，請確定沒有相關的用戶端應用程式在運行，然後以管理員身份登入，並鍵入指令程式檔案 `\Sun\Apoc\bin\apocd restart`。或者，您也可以啓動「服務控制器」(位於 [管理工具] 下的 [控制面板] 中)。從可用服務清單中選取 **Configuration Agent**，然後按一下 [重新啓動]。

準備安裝

[準備安裝] 頁面顯示要安裝的 Configuration Agent 項目。



圖 6-10 Configuration Agent，[準備安裝] 頁面

此時，並未對系統進行任何變更。當您按一下 [立即安裝] 按鈕後，才開始執行實際安裝。

安裝摘要

精靈的 [安裝摘要] 頁面顯示所安裝項目的狀態。按一下相應項目旁邊的 [詳細資訊] 按鈕，可以獲得有關該項目的更多資訊。



圖 6-11 Configuration Agent，[安裝摘要] 頁面

安裝程式會將 Configuration Agent 二進位檔儲存於預設程式檔案目錄 (通常為 C:\Program Files) 下的 Sun\apoc 中。安裝程式還會自動在 [開始] — [程式集] — [Sun] 功能表中自動建立一個新的 Java Desktop System 資料夾。該資料夾包含兩個捷徑，可讓您進一步配置或取消安裝 Configuration Agent。

變更 Configuration Agent 設定

如前所述，您可以在安裝 Configuration Agent 後隨時修改其設定。您可以透過程式檔案 \Sun\Apoc\bin\apoc-config.bat 程序檔，以管理員身份啟動精靈。或者，您可以選擇 [開始] 功能表中的 [程式集] — [Sun] — [Java Desktop System] 下的 **Configuration Agent** 項目以啟動 Configuration Agent 精靈。

注意 – 也可以不啟動圖形介面而啟動該精靈。例如，執行 程式檔案 \Sun\Apoc\bin\apoc-config -nodisplay 來以主控台模式啟動精靈。

取消安裝 Configuration Agent

若要取消安裝 Configuration Agent，請選擇 [開始] 功能表中的 [程式集] — [Sun] — [Java Desktop System] — **Configuration Agent** — [取消安裝 Configuration Agent]。

附錄 A

Sun Web Console 套裝軟體

已知問題

安全性

在使用者不知情的情況下，如果使用者不刻意執行某些動作，某些工作時段可能仍保持使用中狀態。例如，當使用者關閉瀏覽器視窗時，該使用者不會自動登出 Sun Web Console。相反，在關閉應用程式視窗之前，使用者必須在 Sun Web Console 中特意登出工作時段。

安裝程序檔用法

提要：`setup [-h] | [-n] | [-d <var>, <arch>[, client1, client2, ...]] [-u [-f]]`

-h = 列印用法描述

-n = 安裝結束時不啟動伺服器

-u = 解除安裝 Sun Web Console

-f = 如果 Tomcat 和 Java 1.4 已隨安裝應用程式一起安裝，則解除安裝這些套裝軟體。您只能結合 -u 參數使用此參數。

如需可用安裝程式參數的完整描述，請執行 `setup -h`。

若要解除安裝 Sun Web Console，請執行 `/usr/lib/webconsole/setup -u`。

注意 – 如果您正位於 `/usr/lib/webconsole` 目錄或任何相關子目錄中，請勿執行此指令，否則 `pkgm` 將會失敗。

Sun Web Console 套裝軟體

Solaris 套裝軟體

套裝軟體名稱	說明
SUNWmctag	Sun Web Console UI 標籤檔案庫
SUNWmcon	Sun Web Console
SUNWmcos	針對 Sun Web Console 的一般 Solaris 服務
SUNWmcosx	針對 Sun Web Console 的 Solaris 特定發行版本服務
SUNWmconr	Sun Web Console 根
SUNWjato	Sun One Application Framework 運行時間
SUNWtcatu	Tomcat

Linux RPM

套裝軟體名稱	說明
SUNWmctag	Sun Web Console UI 標籤檔案庫
SUNWmcon	Sun Web Console
SUNWmcos	針對 Sun Web Console 的一般 Linux 服務
SUNWmcosx	針對 Sun Web Console 的 Linux 特定發行版本服務
SUNWmconr	Sun Web Console 根

套裝軟體名稱	說明
SUNWjato	Sun One Application Framework 運行時間
tomcat4	Tomcat

附錄 B

Configuration Manager 套裝軟體

Configuration Manager 套裝軟體

Solaris 套裝軟體

套裝軟體名稱	說明
SUNWapm	Configuration Manager
SUNWapmca	Configuration Agent 範本
SUNWapmev	Evolution 範本
SUNWapg26	Gnome 2.6 範本
SUNWapmmo	Mozilla 範本
SUNWapmso	StarSuite 範本

Linux RPM

套裝軟體名稱	說明
apoc-manager	Configuration Manager
apoc-agent-templates	Configuration Agent 範本

套裝軟體名稱	說明
apoc-evolution-templates	Evolution 範本
26-templates	Gnome 2.6 範本
apoc-mozilla-templates	Mozilla 範本
apoc-starsuite-templates	StarSuite 範本

將 OpenLDAP 和 Active Directory 與 Configuration Manager 一起使用

將 OpenLDAP 伺服器與 Configuration Manager 一起使用

若要使用 OpenLDAP 伺服器做為 Configuration Manager 資料的儲存庫，則必須延伸伺服器模式，以便具有儲存配置資料所用的物件類別與屬性。在 Java Desktop System Management Tools CD 中提供之 Configuration Manager 部署工具的 `openldap` 子目錄中，包含名為 `apoc.schema` 的自訂模式檔案。

必須將該檔案複製到 OpenLDAP 配置目錄 (`/etc/openldap`) 的子目錄 `schema` 中，並透過在位於該目錄的 `slapd.conf` 檔案中包含該檔案將其增加到 OpenLDAP 模式中。透過在該檔案中包含的模式序列結尾處插入一行 `include /etc/openldap/schema/apoc.schema` 來實現此目的。如需有關延伸 OpenLDAP 伺服器模式的更多資訊，請參閱伺服器的使用手冊。

為準備 OpenLDAP 資料庫以儲存配置資料，必須使用 Configuration Manager 隨附的部署工具。由於已在先前的安裝步驟中延伸了模式，因此僅需要運行 `createServiceTree` 程序檔。該程序檔必須從部署工具目錄以任何使用者的身份透過以下指令執行：`./createServiceTree`。該程序檔會提示使用者輸入有關 OpenLDAP 資料庫的資訊，如本文件的部署工具一節中所指示。部署工具的子目錄 `openldap` 中提供了預設對映檔案，該檔案使用 OpenLDAP 中具有的典型物件類別與屬性。該檔案名為 `OrganisationalMapping`，並且可以在啟動 `createServiceTree` 之前，透過用該檔案覆蓋主部署工具目錄中的同名檔案，來對其進行部署。

注意 – Configuration Manager Agent 將透過提供要從其獲得資料之使用者的 DN，但不提供密碼，匿名嘗試並連線至 OpenLDAP 伺服器。此匿名驗證模式在 OpenLDAP 伺服器的某些發行版本中可能被停用，在此情形下，必須透過在 `slapd.conf` 檔案 (位於 OpenLDAP 配置目錄 `/etc/openldap`) 中定義的一般伺服器參數中增加一行 `allow bind anon cred` 來啟動匿名驗證模式。如需有關該參數的更多資訊，請參閱該伺服器的使用手冊。

將 Active Directory 伺服器與 Configuration Manager 一起使用

若要使用 Active Directory 伺服器做為 Configuration Manager 資料的儲存庫，則必須延伸伺服器模式，以便具有儲存配置資訊所用的物件類別與屬性。在 Management Tools CD 提供之 Configuration Manager 部署工具的 `ad` 子目錄中，包含名為 `apoc-ad.ldf` 的模式延伸檔案。請參閱部署工具一節，以取得更多資訊。

必須將 `apoc-ad.ldf` 檔案匯入到 Active Directory 模式中，請使用以下步驟：

1. 啟用模式延伸。請參閱 Active Directory 文件，以取得有關如何執行該作業的更多資訊。
2. 從指令提示符號處執行以下指令：`ldifde -i -c "DC=Sun,DC=COM" <BaseDN> -f apoc-ad-registry.ldf`。

注意 – 請使用 Active Directory 基底 DN 替代 `<BaseDN>`。

為準備 Active Directory 伺服器以儲存配置資料，必須使用部署工具。由於已在先前的安裝步驟中延伸了模式，因此僅需要執行 `createServiceTree` 程序檔。該程序檔必須從部署工具目錄以任何使用者的身份透過以下指令執行：`./createServiceTree`。該程序檔會提示使用者輸入有關 Active Directory 資料庫的資訊。部署工具目錄的 `ad` 子目錄中提供了預設對映檔案，該檔案使用 Active Directory 中具有的典型物件類別與屬性。該檔案名為 `OrganisationalMapping`，在啟動 `createServiceTree` 之前，透過用該檔案覆蓋主部署工具目錄中的同名檔案，可以對其進行部署。

此時便可以將 Active Directory 伺服器與 Configuration Manager 一起使用。安裝 Configuration Manager 時，請提供具有對樹的讀取權之使用者的完整 DN 和密碼。此使用者可以是無法將 Active Directory 用於其他目的的使用者。請參閱 Active Directory 說明文件，以取得有關如何設定此類使用者的更多資訊。此外，Active Directory 的網域名稱對執行 Configuration Manager 的機器必須是已知的。可以透過在該機器的 `/etc/hosts` 檔案中增加使 Active Directory 伺服器的 IP 位址與其網域名稱相對映的行，來實現此目的。

為從 Java Desktop System 主機上擷取配置資料，Active Directory 的網域名稱對該主機也必須是已知的。Java Desktop System 使用者的驗證可以採用兩種方式：匿名認證與使用 GSSAPI 認證。

- 若要使用匿名連線認證，則必須將 Active Directory 伺服器配置為授與每個人讀取權限。請參閱 Active Directory 說明文件，以取得有關如何執行該作業的更多資訊。
- 若要使用 GSSAPI 驗證，則必須修改指定 Kerberos 參數的檔案 `/etc/krb5.conf`，以定義 Active Directory 範圍並指向做為其金鑰分配中心 (KDC) 的 Active Directory 伺服器。做為預設加密類型，該檔案還必須指定 Active Directory 支援的 DES 類型，即 `des-cbc-crc` 與 `des-cbc-md5`。請參閱 Kerberos 說明文件，以取得有關如何執行該作業的更多資訊。存取配置資料之前，必須取得登入 Java Desktop System 之使用者的有效憑證。透過執行 `kinit` 指令並提供 Active Directory 中定義的使用者密碼，可以手動執行此作業。其他模式可能會在登入時自動產生這些憑證。請參閱 Java Desktop System 文件，以取得更多資訊。

