

Novell Identity Manager Driver for SOAP

1.0.2

www.novell.com

實作指南

2005 年 11 月 23 日

N

Novell®

法律聲明

Novell, Inc. 不對本文件的內容或使用做任何陳述或保證，且特別聲明不對任何特定用途的適銷性或適用性提供任何明示或隱喻的保證。此外，Novell, Inc. 保留隨時修改本出版品及其內容的權利，且在進行此類修正或更動時，不需另行通知任何人士或公司。

此外，Novell, Inc. 不對任何軟體作任何陳述或保證，且特別聲明不對任何特定用途的適銷性或適用性提供任何明示或隱喻的保證。此外，Novell, Inc. 保留隨時修改任何或全部 Novell 軟體的權利，且在進行此類更動時，不需通知任何人士或公司。

這份授權書中所提及的任何產品或技術資訊皆受到美國出口管制法 (U.S. Export Control) 及其他國家的交易法約束。您同意遵守所有出口管制法規，並取得出口、再出口或進口交付物品所需之任何必要的授權或類別。您同意不出口或再出口至目前美國出口排除清單上所列公司，或者至美國出口法所指定之禁運或恐怖份子的國家。您同意不將交付產品用在禁止的核子武器、飛彈或化學生物武器等用途上。如需更詳細的 Novell 軟體出口資訊，請參閱 www.novell.com/info/exports/。Novell 無須承擔您無法取得任何必要的出口核准之責任。

Copyright © 2005 Novell, Inc. 版權所有。未經出版者的書面同意，本出版品的任何部份皆不可複製、影印、傳送，或是儲存在可擷取系統上。

Novell, Inc. 擁有在此份文件中所描述產品內含技術的智慧財產權。尤其 (但不限於) 這些智慧財產權可能包含一或多個列於 <http://www.novell.com/company/legal/patents/> 的美國專利，以及一或多個在美國和其他國家的額外專利或申請中的專利。

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

線上文件：若要存取本產品及其他 Novell 產品的線上文件，或取得更新，請參閱 www.novell.com/documentation。

Novell 商標

DirXML 是 Novell, Inc. 在美國與其他國家的註冊商標。

eDirectory 是 Novell, Inc. 在美國與其他國家的商標。

NetWare 是 Novell, Inc. 在美國與其他國家的註冊商標。

Novell 是 Novell, Inc. 在美國與其他國家的註冊商標。

Nsure 是 Novell, Inc. 在美國與其他國家的註冊商標。

SUSE 是 Novell, Inc. 在美國與其他國家的註冊商標。

協力廠商資料

所有的協力廠商商標均為其個別擁有廠商的財產。

目錄

關於本指南	3
1 綜覽	5
1.1 驅動程式概念	5
1.1.1 資料管理	5
1.1.2 驅動程式的運作方式	6
1.1.3 了解操作資料	7
1.2 驅動程式功能	7
2 安裝驅動程式	9
2.1 驅動程式先決條件	9
2.2 新功能	9
2.3 安裝驅動程式	9
2.4 升級	9
3 使用範例驅動程式組態	11
3.1 使用驅動程式組態檔案建立驅動程式物件	11
3.1.1 在 Designer 中輸入驅動程式組態檔案	11
3.1.2 在 iManager 中輸入驅動程式組態檔案	11
3.1.3 組態參數	12
3.2 了解 DSML 組態	15
3.3 了解 SPML 組態	16
3.4 可透過處理「發行者」通道上的「修改」事件取消關聯物件	17
4 設定驅動程式	19
4.1 設定驅動程式設定	19
4.2 設定訂閱者設定	22
4.2.1 設定訂閱者以建立至遠端 Web 服務的 HTTPS 連接	22
4.2.2 設定訂閱者以使用代理	23
4.3 設定發行者設定	23
4.3.1 設定發行者以接收 HTTPS 連接	24
4.4 建立 XSLT 樣式表	25
4.5 操作資料	25
4.5.1 使用操作資料指定要傳回到結果中的 XML	25
4.5.2 使用操作資料置換預設訂閱者選項	25
5 使用驅動程式	27
5.1 啟動驅動程式	27
5.2 移轉並重新同步資料	27
5.3 啟用驅動程式	27
6 驅動程式疑難排解	29
6.1 驅動程式 Shim 錯誤	29

6.2	Java 自訂錯誤	30
A	使用 Java 延伸功能	33
A.1	綜覽	33
A.2	建立和設定 Java 延伸功能	34

關於本指南

本指南將說明如何安裝及設定 Identity Manager Driver 1.0 for SOAP (也稱為 SOAP 驅動程式)。

- ◆ 「綜覽」，第 5 頁
- ◆ 「安裝驅動程式」，第 9 頁
- ◆ 「使用範例驅動程式組態」，第 11 頁
- ◆ 「設定驅動程式」，第 19 頁
- ◆ 「驅動程式疑難排解」，第 29 頁

使用對象

本指南適用於實作 Identity Manager 的 eDirectory™ 管理員、應用程式伺服器開發人員、Web 服務管理員與顧問。此外，您還應該對 DSML/SPML、SOAP 和 HTML 有所了解。

意見反應

我們想知道您對於本手冊與其他本產品隨附之文件的意見與建議。請使用線上文件中每頁底下的「使用者意見」功能，或請造訪 www.novell.com/documentation/feedback.html，然後寫下您的意見。

文件更新

如需本文件的最新版本，請參閱 [驅動程式實作指南 \(http://www.novell.com/documentation/dirxmldrivers/index.html\)](http://www.novell.com/documentation/dirxmldrivers/index.html) 一節中的 Identity Manager Driver for SOAP。

其他文件

如需 Identity Manager 的相關資訊，請參閱 [Identity Manager 文件網站 \(http://www.novell.com/documentation/dirxml20/index.html\)](http://www.novell.com/documentation/dirxml20/index.html)。如需其他 Identity Manager 驅動程式的相關資訊，請參閱 [驅動程式實作指南 \(http://www.novell.com/documentation/dirxmldrivers/index.html\)](http://www.novell.com/documentation/dirxmldrivers/index.html)。

文件慣例

在 Novell® 文件中，大於符號 (>) 是用以分隔步驟中的各個動作，以及前後參照路徑中的數個項目。

商標符號 (®、™ 等) 代表 Novell 的商標。星號 (*) 代表協力廠商的商標。

雖然在寫入單一路徑名稱時，有些平台採用反斜線，其他平台採用正斜線，但在顯示時，路徑名稱一律使用反斜線。Linux* 或 UNIX* 等要求使用正斜線之平台的使用者，應依據軟體的要求使用正斜線。

SOAP (簡易物件存取協定) 是一種基於 XML 的協定，用於在不同的應用程式與作業系統之間進行網際網路通訊。

SOAP 驅動程式使用語言與協定的組合，在 Identity Vault with Identity Manager 與啓用 HTTP 的應用程式 (例如啓用 SOAP 的 Web 服務) 之間啓用身份提供與資料同步。

該驅動程式並非用於特定的 Web 服務。它是一般的 Shim，僅處理 Identity Vault 與 Web 服務之間的 HTTP 資料傳輸。對於此驅動程式而言，Web 服務是定義為以 XML 與 HTTP 做為輸送協定的應用程式。該應用程式還會使用 SOAP 對訊息進行編碼。

本節提供下列 Identity Manager Driver 1.0 for SOAP 的相關資訊：

- ◆ 「驅動程式概念」，第 5 頁
- ◆ 「驅動程式功能」，第 7 頁

1.1 驅動程式概念

本節包含下列資訊：

- ◆ 「資料管理」，第 5 頁
- ◆ 「驅動程式的運作方式」，第 6 頁

1.1.1 資料管理

驅動程式會使用各種網際網路協定和語言，在 Identity Manager 與 Web 服務之間交換資料。

- ◆ 「SOAP」，第 5 頁
- ◆ 「SPML 和 DSML」，第 6 頁
- ◆ 「XML」，第 6 頁
- ◆ 「HTTP」，第 6 頁

SOAP

SOAP (簡易物件存取協定) 是基於 XML 的協定，用於交換 Identity Manager 中的訊息。它定義訊息交換，但不定義訊息內容。此驅動程式支援 SOAP 1.1。

SOAP 文件分成三個元素：

- ◆ 封套：XML 根節點。
- ◆ 標題：提供網路位置知識，例如異動 ID 與安全性資訊。
- ◆ 本文：方法特定的資訊。

SOAP 採用 HTTP 申請 / 回應訊息模式，在 HTTP 申請中提供 SOAP 申請參數，並在 HTTP 回應中提供 SOAP 回應參數。

SPML 和 DSML

SOAP 驅動程式包含下列兩個協定的範例組態：SPML 1.0 和 DSML 2.0。

- ◆ **SPML 1.0**：「服務供應標記語言」是基於 XML 的提供申請和回應協定。用戶端會向伺服器發出 SPML 申請。該申請描述要在指定服務點上執行的操作。服務點則會執行必要的操作以實作申請的服務。操作完成後，服務點會將 SPML 回應傳回至用戶端，詳細描述與該申請相關的任何結果或錯誤。

驅動程式支援 SPML 1.0。SPML 與 SOAP 1.1 結合，並以 HTTP 和 HTTPS 1.1 做為輸送協定。

- ◆ **DSML 2.0**：「目錄服務標記語言」以 XML 文件的形式顯示目錄結構資訊、目錄查詢和更新，以及這些操作的結果。

DSML 與 SOAP 1.1 結合，並以 HTTP 和 HTTPS 1.1 做為輸送協定。

如需驅動程式包含之範例 SPML 和 DSML 組態的相關資訊，請參閱「[使用範例驅動程式組態](#)」，第 11 頁。

XML

XML (可延伸標記語言) 是「標準通用標記語言 (SGML)」的一般子集，可用於在網際網路上交換結構化資料。

HTTP

HTTP 是一種通訊協定，用於經由網際網路或其他電腦網路申請和傳輸資料。該協定適用於網際網路基礎結構，也可與防火牆搭配使用。

HTTP 是一種無狀態的申請 / 回應系統，因為通常只有在立即申請的情況下才會保持連接狀態。用戶端會建立與伺服器的 TCP 連接，並會向伺服器傳送申請指令。隨後，伺服器會傳回回應。

1.1.2 驅動程式的運作方式

下圖說明 Identity Manager 與 Web 服務之間的資料流程：

特性 1-1 SOAP 驅動程式資料流程



Identity Manager 引擎使用 XDS (一種特殊格式的 XML) 來表示 Identity Vault 中的事件。Identity Manager 將 XDS 傳遞至驅動程式規則，該規則由基本規則、DirXML® Script 及 XSLT 樣式表組成。

在「訂閱者」通道上，驅動程式規則會將 XDS 轉譯為 XML (例如 SOAP)。而在「發行者」通道上，驅動程式規則會將 SOAP 等其他格式的 XML 轉譯為 XDS。

驅動程式 Shim 使用 HTTP 與 Web 服務進行通訊。通常，在驅動程式 Shim 與應用程式之間進行遞交的是序列化 XML。

例如，假設驅動程式使用 DSML 範例組態與僅設定為「訂閱者」的 DSML 伺服器進行通訊，當 Identity Vault 中發生事件時，Identity Manager 會建立 XDS 指令來代表該事件。Identity Manager 會將 XDS 指令傳遞至驅動程式規則。

驅動程式規則會以輸出轉換樣式表轉換 XDS 指令。XSLT 樣式表會將 XDS 轉換為包含 DSML 的 SOAP 封套。隨後會將該 SOAP 封套傳遞至驅動程式 Shim。驅動程式 Shim 會將 SOAP 封套轉換為位元組陣列，建立適當的 HTTP 連接，然後執行 HTTP POST 操作以將資料提交至 Web 服務。

Web 服務或應用程式會處理申請，並將 SOAP 回應傳回至驅動程式 Shim。Shim 會以位元組陣列形式接收回應，並將該回應轉換為 XML 文件，然後傳回至驅動程式規則。輸入轉換樣式表會處理回應，將其轉換為回報給 Identity Manager 引擎的適當 XDS。

1.1.3 了解操作資料

驅動程式 Shim 會根據內嵌於指令中的 XML 元素，對訂閱者指令套用特殊處理，在 Shim 中會顯示為 <operation-data>。使用 <operation-data> 元素有兩個目的。首先，可以用來比對指令與指令所產生的回應，這對於建立兩者之間的關聯非常有用。其次，可以用來置換預設「訂閱者」通道連接屬性。

<operation-data> 元素會從某個「訂閱者」通道規則新增至指令。驅動程式 Shim 會先從指令中移除 <operation-data> 元素，然後再將其傳送至應用程式，並將 <operation-data> 元素還原到產生的回應中。

依預設，當在回應中還原 <operation-data> 元素時，該元素會附加為根節點的子元素。也可以為 <operation-data> 元素提供一或多個 parent-node-*n* 屬性來置換預設，其中，*n* 是從 1 開始的數字，且每當要提供一個父指定器時，該數字便會遞增。驅動程式 Shim 會檢查操作資料節點以尋找 parent-node-*n* 屬性。如果找到屬性，會依次嘗試每個屬性，如果存在具名節點，則該節點會用做回應上之操作資料的父節點。

若要查看 <operation-data> 元素與樣式表搭配使用的方式，請參閱「[操作資料](#)」，第 25 頁。

1.2 驅動程式功能

驅動程式包含下列功能：

- ◆ Identity Vault 與 Web 服務之間的 HTTP 資料傳輸。
- ◆ SPML 與 DSML 的範例組態
- ◆ HTTP 申請標題欄位的自訂

依預設，會為「訂閱者」通道提供具有 ID 和密碼的基本授權申請標題。如需相關資訊，請參閱「[使用驅動程式組態檔案建立驅動程式物件](#)」，第 11 頁。

- ◆ 使用 HTTPS 協定的 SSL 連接
 - ◆ 訂閱者 HTTP 與 HTTPS 代理伺服器
 - ◆ 在執行時期定義並選取規則中的多個訂閱者連接
 - ◆ 可以做為發行者通道上內送連接的 HTTP 或 HTTPS 監聽程式
 - ◆ 可以使用自定的 Java^{*} 程式碼進行擴充
- 如需相關資訊，請參閱[附錄 A 「使用 Java 延伸功能」](#)，第 33 頁。

安裝驅動程式

本節包含下列安裝驅動程式的相關資訊：

- ◆ 「[驅動程式先決條件](#)」，第 9 頁
- ◆ 「[安裝驅動程式](#)」，第 9 頁

2.1 驅動程式先決條件

- 下列其中一個作業系統：
 - ◆ 含最新支援套件 (Support Pack) 的 NetWare® 6 或 6.5
 - ◆ 含最新支援套件的 Novell® Open Enterprise Server
 - ◆ 含最新 Service Patch 的 Windows* NT*、2000 或 2003
 - ◆ Linux Red Hat* AS、ES 2.1 或 AS 3.0
 - ◆ SUSE® LINUX Enterprise Server 8 或 9 (含 SP1)
 - ◆ Solaris* 8 或 9
 - ◆ AIX* 5.2L
- 含最新支援套件的 Novell eDirectory™ 8.7.3 或 Novell eDirectory 8.8
- Novell Identity Manager 3.0
- Novell iManager 2.5 或更新版本

2.2 新功能

- ◆ 置換 SOAP-action 的能力
- ◆ 處理 Cookie

2.3 安裝驅動程式

此驅動程式在您安裝 Novell Identity Manager 3 程式時，就會同時安裝。如需安裝指示，請參閱《[Identity Manager 3.0 安裝指南](#)》中的「[安裝 Identity Manager](#)」和「[升級](#)」這兩章。

輸入驅動程式組態就會建立驅動程式物件。輸入組態之後，您便可以使用 iManager 設定和管理該驅動程式。如需如何設定驅動程式的指示，請參閱第 3 章「[使用範例驅動程式組態](#)」，第 11 頁。

2.4 升級

若要升級到 Identity Manager 3.0，請遵循《[Identity Manager 3.0 安裝指南](#)》中「[升級](#)」章節內的指示。

使用範例驅動程式組態

您可以使用 Identity Manager Driver for SOAP 中隨附的兩個範例組態做為建立「驅動程式」物件的起點。

本節包含下列主題：

- ◆ 「使用驅動程式組態檔案建立驅動程式物件」，第 11 頁
- ◆ 「了解 SPML 組態」，第 16 頁
- ◆ 「了解 DSML 組態」，第 15 頁

3.1 使用驅動程式組態檔案建立驅動程式物件

SOAP 驅動程式中隨附的兩個組態檔案，可以用來建立「驅動程式」物件：

- ◆ SOAP-SPML.xml：服務供應標記語言 (SPML) 組態檔案
- ◆ SOAP-DSML.xml：目錄服務標記語言 (DSML) 組態檔案

如需範例檔案的相關資訊，請參閱「了解 SPML 組態」，第 16 頁和「了解 DSML 組態」，第 15 頁。

3.1.1 在 Designer 中輸入驅動程式組態檔案

Designer 可以讓您輸入 SOAP 驅動程式的驅動程式組態檔案。這些檔案會建立和設定驅動程式正常運作所需的物件及規則。下列指示會說明建立驅動程式和輸入驅動程式組態的方法。

輸入驅動程式組態檔案的方法有很多種。此程序僅記錄了其中一種方法。

- 1 在 Designer 中開啓專案，並以滑鼠右鍵按一下模擬器中的「驅動程式集」物件，然後選取「新增已連接的應用程式」。
- 2 從下拉式清單中選取「SOAP-DSML.xml」或「SOAP-SPML.xml」，然後按一下「執行」。
- 3 在「執行提示驗證」視窗中按一下「是」。
- 4 利用填入欄位來設定驅動程式。請指定您環境的特定資訊。如需這些設定的相關資訊，請參閱表格 3-1 頁上 13 和表格 3-2 頁上 14。
- 5 指定參數後，請按一下「確定」來輸入驅動程式。
- 6 輸入驅動程式後，請自定並測試該驅動程式。
- 7 驅動程式測試全部完成之後，再將驅動程式部署到 Identity Vault。請參閱《Designer for Identity Manager 3：管理指南》中的「將規則部署至 Identity Vault」。

3.1.2 在 iManager 中輸入驅動程式組態檔案

SOAP 預先設定檔案是一個範例組態檔案。當在 iManager 伺服器上安裝 Identity Manager Web 元件時就會安裝此檔案。您可以將預先設定檔案視為您輸入，並根據環境予以自定或設定的範本。

- 1 在 iManager 中，選取「Identity Manager 公用程式」>「輸入驅動程式」。

- 2 選取驅動程式集，然後按「下一步」。

您要將新的驅動程式置於何處？

- 在現有的驅動程式集中

drvset.novell  

- 在新的驅動程式集中

如果您將此驅動程式置於新的驅動程式集中，則必須指定驅動程式集名稱、網路位置和相關聯的伺服器。

- 3 選取「SOAP DSML」或「SOAP SPML」，然後按「下一步」。

-  SOAP DSML
-  SOAP SPML

- 4 利用填入組態參數來設定驅動程式。如需這些設定的相關資訊，請參閱 [表格 3-1 頁上 13](#) 和 [表格 3-2 頁上 14](#)。
- 5 使用使用者物件定義安全性等值，該物件要具備伺服器上之驅動程式所需具備的權限。此任務最常使用的是「管理員」使用者物件。不過，您可能要建立 DriversUser (舉例來說)，並為該使用者指定安全性等值。不管伺服器上之驅動程式所需具備的權限為何，DriversUser 物件都必須具有相同的安全性權限。
- 6 識別所有代表「管理角色」的物件，並將它們從複製中排除。排除在步驟 2 中指定的安全性等值物件 (例如，DriversUser)。如果您刪除安全性等值物件，則表示已從驅動程式中移除權限。因此，驅動程式無法對 Identity Manager 進行變更。
- 7 按一下「完成」。
- 8 設定驅動程式的其他設定。
如需相關資訊，請參閱 [「設定驅動程式」](#)，第 19 頁。

3.1.3 組態參數

下表說明啓始設定驅動程式組態期間必須提供的參數。

附註：這些參數會顯示在數個螢幕上，但是有些參數僅在回答先前提示需要更多資訊來正確設定規則時顯示。

表格 3-1 SOAP DSML 驅動程式的組態參數

欄位	描述
驅動程式名稱	在 Identity Manager 中指定驅動程式物件的名稱。
設定資料流程	指定您要啓用的驅動程式通道。 eDirectory 至 DSML ：將 Identity Vault 事件傳送至應用程式。 DSML 至 eDirectory ：從應用程式接收事件。 雙向 ：同時啓用 eDirectory™ 和 DSML 通道。
<nds>、<input>、<output> 元素處理	選取下列其中一項： 移除/新增元素 ：驅動程式 Shim 可以移除和新增 nds、input 及 output 這些必要的 XML 元素。在將 XML 文件傳送至應用程式之前會從其中移除這些必要元素，而從應用程式接收到 XML 文件之後會再次加入這些必要元素，然後會將此文件傳送至 Metadirectory 引擎。 此為 SOAP 驅動程式的偏好選項。 略過元素 ：關閉元素處理。不需要將 nds、input 及 output 的必要 XML 元素新增至 XML 文件或從其中移除。
驅動程式為本地/遠端	選取下列其中一項： 本地 ：從擁有驅動程式集的伺服器執行驅動程式 Shim。 遠端 ：利用「遠端載入器」從遠端伺服器執行驅動程式。若要指定此選項，請按「下一步」，然後指定「遠端載入器」組態資訊。如需相關資訊，請參閱《 Novell Identity Manager 3.0 管理指南 》中的「設定已連接系統」。
遠端 DSML 伺服器的 URL： (條件式) 訂閱者通道欄位	指定「遠端 DSML 伺服器的 URL」以及此伺服器所監聽的連接埠號碼。 例如：http://137.66.10.13:18180/soap
附註：只有當您在「設定資料流程」欄位中選取「eDirectory 至 DSML」或「雙向」時，才會顯示這些欄位。	伺服器是用於監聽、處理以及傳回有效 DSML 申請之結果的軟體元件。
	提示：如果您將驅動程式設定為使用 SSL，則 URL 必須以 https 開頭而不是以 http 開頭。
驗證 ID (條件式) 訂閱者通道欄位	如果遠端伺服器需要「驗證 ID」，請在此欄位中指定它。否則，將此欄位保留空白。
驗證密碼 (條件式) 訂閱者通道欄位	如果在上述欄位中指定了「驗證 ID」，請指定遠端伺服器的「驗證密碼」。否則，將此欄位保留空白。
監聽 IP 位址和連接埠 (條件式) 發行者通道欄位	指定裝有 SOAP 驅動程式之伺服器的 IP 位址，以及此驅動程式監聽的連接埠號碼。如果伺服器中只安裝了一張網路卡，則您可以指定 127.0.0.1。選擇伺服器上未使用的連接埠號碼，例如 127.0.0.1:18180。驅動程式會監聽此位址，以獲取申請，並加以處理，然後傳回結果。
附註：只有當您在「設定資料流程」欄位中選取「DSML 至 eDirectory」或「雙向」時，才會顯示這些欄位。	

欄位	描述
驗證 ID (條件式) 發行者通道欄位	指定遠端 DSML 伺服器的「驗證 ID」以驗證內送申請。如果遠端伺服器不傳送「驗證 ID」，則將此欄位保留空白。
驗證密碼 (條件式) 發行者通道欄位	如果在上述欄位中指定了「驗證 ID」，請指定遠端伺服器的「驗證密碼」以驗證內送申請。否則，將此欄位保留空白。
遠端主機名稱和連接埠 (條件式) 遠端載入器欄位	輸入執行遠端載入器伺服器之伺服器的主機名稱或 IP 位址及連接埠。 範例：137.66.10.13:8090
附註：只有當您在「驅動程式為本地/遠端」欄位中選取「遠端」時，才會顯示這些欄位。	連接埠 8090 是「遠端載入器」服務所監聽的預設連接埠。
驅動程式密碼 (條件式) 遠端載入器欄位	遠端載入器使用驅動程式密碼向 Identity Manager 伺服器執行自我驗證。此密碼必須與遠端載入器伺服器上指定的「驅動程式物件密碼」相同。
遠端密碼 (條件式) 遠端載入器欄位	遠端密碼用於控制對遠端載入器的存取。此密碼必須與「遠端載入器」伺服器上指定的「遠端載入器」密碼相同。

表格 3-2 SOAP SPML 驅動程式的組態參數

欄位	描述
驅動程式名稱	在 Identity Manager 中指定驅動程式物件的名稱。
設定資料流程	指定您要啓用的驅動程式通道。 eDirectory 至 SPML ：將 Identity Vault 事件傳送至應用程式。 SPML 至 eDirectory ：從應用程式接收事件。 雙向 ：同時啓用 eDirectory 和 SPML 通道。
<nds>、<input>、<output> 元素處理	選取下列其中一項： 移除/新增元素 ：驅動程式 Shim 可以移除和新增 nds、input 及 output 這些必要的 XML 元素。在將 XML 文件傳送至應用程式之前會從其中移除這些必要元素，而從應用程式接收到 XML 文件之後會再次加入這些必要元素，然後會將此文件傳送至 Metadirectory (Identity Manager) 引擎。 此為 SOAP 驅動程式的偏好選項。 略過元素 ：關閉元素處理。不需要將 nds、input 及 output 的必要 XML 元素新增至 XML 文件或從其中移除。
驅動程式為本地/遠端	選取下列其中一項： 本地 ：從擁有驅動程式集的伺服器執行驅動程式 Shim。 遠端 ：利用「遠端載入器」從遠端伺服器執行驅動程式。若要指定此選項，請按「下一步」，然後指定「遠端載入器」組態資訊。如需相關資訊，請參閱《Novell Identity Manager 3.0 管理指南》中的「設定已連接系統」。

欄位	描述
遠端 SPML 提供服務點的 URL： (條件式) 訂閱者通道欄位	指定遠端 SPML 提供服務點 (PSP) 的 URL。 例如： <code>http://137.66.10.13:18180/soap</code>
附註：只有當您在「設定資料流程」欄位中選取「eDirectory 至 SPML」或「雙向」時，才會顯示這些欄位。	PSP 是用於監聽、處理以及傳回有效 SPML 申請之結果的軟體元件。
驗證 ID (條件式) 訂閱者通道欄位	提示：如果您將驅動程式設定為使用 SSL，則 URL 必須以 <code>https</code> 開頭而不是以 <code>http</code> 開頭。 如果遠端 SPML PSP 需要驗證 ID，則請指定該遠端 SPML PSP 的驗證 ID。否則，將此欄位保留空白。
驗證密碼 (條件式) 訂閱者通道欄位	如果您已指定上述驗證 ID，請指定遠端 SPML PSP 的驗證密碼以驗證內送申請。否則，將此欄位保留空白。
監聽 IP 位址和連接埠 (條件式) 發行者通道欄位	指定安裝有驅動程式之伺服器的 IP 位址，以及該驅動程式做為 PSP 監聽的連接埠號碼。如果伺服器中只安裝了一張網路卡，則您可以指定 <code>127.0.0.1</code> 。選擇伺服器上未使用的連接埠號碼。 範例： <code>127.0.0.1:18180</code>
附註：只有當您在「設定資料流程」欄位中選取「SPML 至 eDirectory」或「雙向」時，才會顯示這些欄位。	驅動程式會監聽此位址，以獲取 SPML 申請，並加以處理，然後傳回結果。
驗證 ID (條件式) 發行者通道欄位	指定驗證 ID 以驗證內送 SPML 申請。
驗證密碼 (條件式) 發行者通道欄位	指定驗證密碼以驗證內送 SPML 申請。
遠端主機名稱和連接埠 (條件式) 遠端載入器欄位	輸入執行「遠端載入器」伺服器之伺服器的主機名稱或 IP 位址及連接埠。 範例： <code>137.66.10.13:8090</code>
附註：只有當您在「驅動程式為本地/遠端」欄位中選取「遠端」時，才會顯示這些欄位。	連接埠 8090 是「遠端載入器」服務所監聽的預設連接埠。
驅動程式密碼 (條件式) 遠端載入器欄位	「遠端載入器」使用驅動程式密碼向 Identity Manager 伺服器執行自我驗證。此密碼必須與「遠端載入器」伺服器上指定的「驅動程式物件密碼」相同。
遠端密碼 (條件式) 遠端載入器欄位	遠端密碼用於控制對「遠端載入器」的存取。此密碼必須與「遠端載入器」伺服器上指定的「遠端載入器」密碼相同。

3.2 了解 DSML 組態

範例 DSML 組態使用 DSML 2.0，並與使用 HTTP 或 HTTPS 1.1 做為輸送協定的 SOAP 1.1 相結合。所有資料的轉換和處理都在規則和樣式表中完成。

範例 DSML 輸入檔案會執行下列動作：

- ◆ 顯示用於比對 Identity Vault DSML 實作的簡單組態。
- ◆ 提供規則中 XDS 到 DSML 和 DSML 到 XDS 的轉換。
- ◆ 處理使用者、群組和組織單位。
可以藉由規則和樣式表自訂來處理其他物件。
- ◆ 支援字串、結構化和可辨識名稱 (DN) 屬性類型。
以下是使用其他資料類型處理屬性的兩個範例。「郵寄地址」屬性會顯示結構化屬性的處理方法。「成員」屬性則會顯示 DN 屬性的處理方法。可以藉由規則和樣式表自訂來處理其他屬性資料類型。
- ◆ 處理查詢操作的子集。
可以藉由規則和樣式表自訂來處理特定的查詢操作。
- ◆ 支援密碼設定操作。
可以藉由規則和樣式表自訂來實現密碼同步化。
- ◆ 「訂閱者」通道使用目的 DN 做為關聯金鑰。
- ◆ 「發行者」通道使用應用程式提供的 DN 做為關聯金鑰。

3.3 了解 SPML 組態

範例 SPML 組態使用 SPML 1.0，並與使用 HTTP 或 HTTPS 1.1 做為輸送協定的 SOAP 1.1 相結合。所有資料的轉換和處理都在規則和樣式表中完成。

範例 SPML 輸入檔案會執行下列動作：

- ◆ 提供一般 SPML 功能。
輸入檔案與特定 SPML 應用程式不匹配。
- ◆ 提供規則中的 XDS 到 SPML 和 SPML 到 XDS 轉換。
- ◆ 處理使用者、群組和組織單位
可以藉由規則和樣式表自訂來處理其他物件。
- ◆ 處理每個屬性的單一值。
可以藉由規則和樣式表自訂來處理屬性的多個值。
- ◆ 處理查詢操作的子集。
組態會將所有查詢都當做 SPML scope = "subtree" 來處理，並使用項目和從屬範圍概念。可以藉由規則和樣式表自訂來處理特定的查詢操作。
- ◆ 支援字串、結構化和可辨識名稱 (DN) 屬性類型。
- ◆ 支援密碼設定操作。
可以藉由規則和樣式表自訂來實現密碼同步化。
- ◆ 處理 execution=synchronous 和 processing=sequential 的單一 (非批次) 操作。
可以藉由規則和樣式表自訂支援批次申請。
- ◆ 無法處理 <addResponse><attributes> 或 <modifyResponse><modifications>。
- ◆ 「訂閱者」通道使用應用程式傳回的識別碼值做為關聯金鑰。
- ◆ 「發行者」通道使用 DN 做為關聯金鑰，並會將關聯金鑰做為識別碼值傳回。

3.4 可透過處理「發行者」通道上的「修改」事件取消關聯物件

對 HTTP/SOAP 驅動程式的「發行者」通道有若干限制，即只允許監聽「變更」事件，而無法查詢其他資訊，也不能輪詢 HTTP/SOAP 來源。因此，始終無法在「發行者」通道接收到取消關聯物件（或由不同驅動程式例項建立的物件）的「修改」事件（傳回錯誤）。原因在於，如果驅動程式和 Metadirectory 引擎無法將查詢傳送至 HTTP/SOAP 來源，也就無法完成從取消關聯的「修改」事件到「新增」指令的轉變。因為 SOAP 驅動程式不具備將查詢傳回到來源的機制，它會傳回一個錯誤訊息，說明未實作查詢。

目前對此限制尚無一般的解決方案。所以，如果發生此狀況，DSML 和 SPML 的範例組態都會傳回一個錯誤訊息。不過，如果在特定的驅動程式部署中，必須在物件上套用關聯並且允許此新關聯的物件中存在不一致資訊，那麼只要在「修改」事件中設定「目的 DN」並建立自己的一組關聯事件，就可以在規則中突破上述限制。這樣的話，即使之前未關聯現有物件，也可以對此物件進行修改。

設定驅動程式

在使用其中一個範例檔案建立「驅動程式」物件之後，您還需要設定 Identity Manager Driver for SOAP。本節包含下列有關設定驅動程式的資訊：

- ◆ 「設定驅動程式設定」，第 19 頁
- ◆ 「設定訂閱者設定」，第 22 頁
- ◆ 「設定發行者設定」，第 23 頁
- ◆ 「建立 XSLT 樣式表」，第 25 頁

4.1 設定驅動程式設定

- 1 在 iManager 中，按一下「Identity Manager > Identity Manager 概觀」。
- 2 找出包含 SOAP 驅動程式的驅動程式集，然後按一下驅動程式的圖示。
- 3 從「Identity Manager 驅動程式概觀」頁面，按一下 SOAP 驅動程式物件，驅動程式組態隨即顯示。
- 4 指定驅動程式模組資訊：
 - 4a 在「驅動程式模組」區段中，選取「Java」。
 - 4b 在「名稱」欄位中，指定下列 SOAP 驅動程式的 Java 類別名稱：
`com.novell.nds.dirxml.driver.soap.SOAPDriver`
- 5 指定驅動程式物件的密碼資訊：
 - 5a 捲動至「驅動程式物件密碼」區段，然後按一下「設定密碼」。
 - 5b 在此欄位中，輸入並重新輸入驅動程式物件的密碼。
- 6 指定驗證資訊：
 - 6a 捲動至「驗證」區段。
 - 6b 指定「驗證 ID」。
 - 6c 指定「驗證網路位置」。
 - 6d 指定「遠端載入器連接參數」。
 - 6e 指定「驅動程式快取限制」（千位元組）。
 - 6f 按一下「設定密碼」，即可指定「應用程式密碼」。
 - 6g 輸入並重新輸入「應用程式密碼」。
- 7 指定啟動資訊：
 - 7a 捲動至「啟動」區段。
 - 7b 選取下列其中一項：
 - ◆ 自動開始：eDirectory™ 啟動時會自動啟動驅動程式。
 - ◆ 手動：必須使用 iManager 手動啟動驅動程式。
 - ◆ 關閉：驅動程式不執行。
- 8 指定下列驅動程式設定：

區段	欄位	描述
驅動程式設定	<nds>、<input>、<output> 元素處理	<p>若要讓驅動程式 Shim 移除和新增必要的 XML 元素 (如 <nds>、<input> 和 <output>)，請指定「移除/新增元素」。</p> <p>在將 XML 文件傳送至應用程式之前會先從其中移除這些必要元素，而從應用程式接收 XML 文件之後又會加入這些必要元素，然後將此文件傳遞給 Metadirectory 引擎。否則，指定「略過元素」以關閉此元素處理。</p>
	自定 Java 延伸功能	<p>如果您已開發自定 Java 類別用以延伸驅動程式 Shim 的功能，請指定「顯示」。否則，指定「隱藏」。</p> <p>如需相關資訊，請參閱附錄 A「使用 Java 延伸功能」，第 33 頁。</p>
訂閱者設定	遠端 DSML 伺服器的 URL	<p>輸入遠端伺服器的 URL 以及此伺服器所監聽的連接埠號碼。</p> <p>通常，URL 必須以 http:// 開頭，但是如果設定 SSL 設定，則應以 https:// 開頭，並且要使用 DNS 主機名稱取代 IP 位址。</p>
	(條件式) 驗證 ID	<p>如果遠端伺服器需要驗證 ID，請在此欄位中輸入 ID。否則，將此欄位保留空白。</p>
	(條件式) 驗證密碼和重新輸入驗證密碼	<p>如果在上述欄位中輸入了「驗證 ID」，請輸入遠端伺服器的驗證密碼。否則，將此欄位保留空白。</p>
	移除現有密碼	<p>按一下此方塊移除現有密碼。然後在「驗證密碼」和「重新輸入驗證密碼」欄位中指定新密碼。</p> <p>必須選取此方塊才能變更密碼。</p>
	Truststore 檔案	<p>指定包含託管證書之 KeyStore 檔案的名稱和路徑，這些證書會在遠端伺服器設定為提供伺服器驗證時使用。例如： c:\security\truststore。如果沒有使用伺服器驗證，則將該欄位保留空白。</p>
	設定相互驗證參數	<p>指定「顯示」以設定相互驗證資訊。指定「隱藏」以取消相互驗證。</p>
	代理主機和連接埠	<p>如果使用代理主機和連接埠，則指定主機位址和主機連接埠。例如：192.10.1.3:18180。</p> <p>如果不使用代理主機和連接埠，則將此欄位保留空白。</p>

區段	欄位	描述
	處理 HTTP 會期 cookie	<p>一些 HTTP 應用程式會設定 cookie，並預期它們在以後發出申請時出現。如果想讓驅動程式追蹤會期 cookie，請選取「處理 Cookie」。</p> <p>Cookie 僅保留到驅動程式停止。</p> <p>如果 HTTP 應用程式不需要 cookie，請選取「忽略 Cookie」。</p>
	自定 HTTP 申請標題欄位	<p>選取「顯示」啟用自定標題欄位，或選取「隱藏」停用此功能。根據您是選取「使用者」還是「忽略」，以下各個欄位為條件式。</p> <ul style="list-style-type: none"> ◆ 授權：如果選取「使用」，請在適當欄位指定金鑰和值。如果在「訂閱者設定」中輸入了驗證 ID 和密碼，則會自動使用此標題。 ◆ 網路位置類型：如果選取「使用」，請在適當欄位指定金鑰和值。 ◆ SOAPAction：如果選取「使用」，請在適當欄位指定金鑰和值。 ◆ 選擇性申請標題：如果選取「使用」，請在適當欄位指定金鑰和值。最多可以指定三個選擇性申請標題。
發行者設定	監聽 IP 位址和連接埠	<p>指定裝有 SOAP 驅動程式之伺服器的 IP 位址，以及此驅動程式監聽的連接埠號碼。</p> <p>如果已輸入範例組態檔案，則此欄位包含您在精靈中指定的 IP 位址和連接埠。</p>
	(條件式) 驗證 ID	<p>指定遠端伺服器的「驗證 ID」以驗證內送申請。如果遠端伺服器不傳送「驗證 ID」，則將此欄位保留空白。</p> <p>如果已輸入範例組態檔案，則此欄位包含您在精靈中指定的 IP 位址和連接埠。</p>
	(條件式) 驗證密碼和重新輸入驗證密碼	<p>如果在上述欄位中輸入了「驗證 ID」，請指定遠端伺服器的驗證密碼以驗證內送申請。否則，將此欄位保留空白。</p>
	移除現有密碼	<p>按一下方塊移除現有密碼，然後在「驗證密碼」和「重新輸入驗證密碼」欄位中指定新密碼。</p> <p>必須選取此方塊才能變更密碼。</p>
	KMO 名稱	<p>指定 eDirectory 中使用的「KMO 名稱」。</p> <p>當伺服器設定為接受 HTTPS 連接時，此名稱便會成為 eDirectory 中的 KMO 名稱。在 RDN 中，「KMO 名稱」是顯示於 "(" (破折號) 之前的名稱。</p> <p>如果使用 KeyStore 檔案 (請參閱下文的 KeyStore 檔案)，或不使用 HTTPS 連接，請將此欄位保留空白。</p>

區段	欄位	描述
	KeyStore 檔案	指定 KeyStore 檔案的 KeyStore 名稱及路徑。當伺服器設定為接受 HTTPS 連接時，便會使用此檔案。 如果使用 KMO 名稱 (請參閱上文的 KMO 名稱)，或不使用 HTTPS 連接，請將此欄位保留空白。
	KeyStore 密碼	當此伺服器設定為接受 HTTPS 連接時，指定與上述指定之 KeyStore 檔案搭配使用的 KeyStore 檔案密碼。 如果使用 KMO 名稱或不使用 HTTPS 連接，請將此欄位保留空白。
	伺服器鍵別名	當伺服器設定為接受 HTTPS 連接時，請指定「伺服器鍵別名」。 如果使用 KMO 名稱或不使用 HTTPS 連接，請將此欄位保留空白。
	伺服器鍵密碼	當此伺服器設定為接受 HTTPS 連接時，此密碼便是鍵別名密碼 (並非 KeyStore 密碼)。如果使用 KMO 名稱 (請參閱上文) 或不使用 HTTPS 連接，請將此欄位保留空白。
	以秒為單位的活動訊號間隔	以秒為單位指定活動訊號間隔。 若要關閉活動訊號，則將此欄位保留空白。如需活動訊號的相關資訊，請參閱《 Novell Identity Manager 3.0 管理指南 》中的「 新增驅動程式活動訊號 」。

- 9 按一下「套用」，然後按一下「確定」。

4.2 設定訂閱者設定

- 1 在 iManager 中，按一下「[Identity Manager > Identity Manager 概觀](#)」。
- 2 找出內含 SOAP 驅動程式的驅動程式集，然後按一下驅動程式的圖示，以顯示「[Identity Manager 驅動程式概觀](#)」頁面。
- 3 在「[Identity Manager 驅動程式概觀](#)」頁面上，再按一下驅動程式的圖示，然後捲動至「[訂閱者設定](#)」。
- 4 根據[步驟 8, 第 19 頁](#)中的描述指定「[訂閱者設定](#)」。
- 5 按一下「套用」，然後按一下「確定」。

4.2.1 設定訂閱者以建立至遠端 Web 服務的 HTTPS 連接

如果存取的遠端 Web 服務允許 HTTPS 連接，則可以設定「[訂閱者](#)」以利用這項功能。您需要一個含有證書的信任儲存，該證書由簽署伺服器證書的證書權限簽發。如需範例，請參閱「[設定發行者以接收 HTTPS 連接](#)」，[第 24 頁](#)。

使用 Java 的 `keytool` 將此證書輸入到信任儲存。如需 `keytool` 的相關資訊，請參閱 [Keytool - 金鑰與證書管理工具 \(http://java.sun.com/j2se/1.4.2/docs/tooldocs/windows/keytool.html\)](http://java.sun.com/j2se/1.4.2/docs/tooldocs/windows/keytool.html)。

- 1 將證書輸入信任儲存，或在指令提示符中輸入下列指令以新建信任儲存：

```
keytool -import -file name_of_cert_file -trustcacerts -noprompt -  
keystore filename -storepass password
```

例如：

```
keytool -import -file tree_ca_root.b64 -trustcacerts -noprompt -  
keystore dirxml.keystore -storepass novell
```

- 2 設定「訂閱者」以使用您在**步驟 1**中建立的信任儲存。
 - 2a 在 iManager 中，按一下「*Identity Manager > Identity Manager* 概觀」。
 - 2b 找出內含 SOAP 驅動程式的驅動程式集，然後按一下驅動程式的圖示，以顯示「*Identity Manager 驅動程式概觀*」頁面。
 - 2c 在「*Identity Manager 驅動程式概觀*」頁面上，再按一下驅動程式的圖示，然後捲動至「訂閱者設定」。
 - 2d 在「*Truststore* 檔案」設定中，指定您在**步驟 1**中建立之信任儲存的路徑。
- 3 按一下「套用」，然後按一下「確定」。

4.2.2 設定訂閱者以使用代理

您可以設定訂閱者以使用 HTTP 或 HTTPS 代理伺服器。

- 1 在 iManager 中，按一下「*Identity Manager > Identity Manager* 概觀」。
- 2 找出內含 SOAP 驅動程式的驅動程式集，然後按一下驅動程式的圖示，以顯示「*Identity Manager 驅動程式概觀*」頁面。
- 3 在「*Identity Manager 驅動程式概觀*」頁面上，再按一下驅動程式的圖示，然後捲動至「訂閱者設定」。
- 4 在「代理主機」和「連接埠」設定中，使用下列格式指定代理主機和連接埠：
host:port
- 5 按一下「套用」，然後按一下「確定」。

4.3 設定發行者設定

- 1 在 iManager 中，按一下「*Identity Manager > Identity Manager* 概觀」。
- 2 找出內含 SOAP 驅動程式的驅動程式集，然後按一下驅動程式的圖示，以顯示「*Identity Manager 驅動程式概觀*」頁面。
- 3 在「*Identity Manager 驅動程式概觀*」頁面上，再按一下驅動程式的圖示，然後捲動至「發行者設定」。
- 4 根據**步驟 8, 第 19 頁**中的描述指定「訂閱者設定」。
- 5 按一下「套用」，然後按一下「確定」。

4.3.1 設定發行者以接收 HTTPS 連接

- 1 在 iManager 中建立伺服器證書。
 - 1a 按一下「*Novell Certificate Server* > 建立伺服器證書」。
 - 1b 瀏覽到安裝有 SOAP 驅動程式的伺服器物件，並選取此物件。
 - 1c 指定證書綽號。
 - 1d 選取「標準」做為建立方法，然後按「下一步」。
 - 1e 按一下「完成」，然後按一下「關閉」。
- 2 從 eDirectory 的證書權限輸出自行簽署的證書。
 - 2a 按一下「*eDirectory* 管理 > 修改物件」。
 - 2b 選取網路樹的證書權限物件，然後按一下「確定」。
網路樹通常位於「安全性」容器中，名稱類似 *TREENAME CA.Security*。
 - 2c 按一下「證書 > 自行簽署的證書」。
 - 2d 按一下「輸出」。
 - 2e 當詢問您是否想要輸出含證書的私密金鑰時，按一下「否」，然後按「下一步」。
 - 2f 根據要存取 Web 服務的用戶端，為證書選取「二進位 *DER* 格式的檔案」或「*Base64* 格式的檔案」，然後按「下一步」。
如果用戶端使用以 Java 為基礎的 *KeyStore* 或信任儲存，則可以選擇任一格式。
 - 2g 按一下「將輸出的證書儲存至檔案」。
 - 2h 按一下「儲存」，然後瀏覽到電腦上某個已知位置。
 - 2i 按一下「儲存」，然後按一下「關閉」。
- 3 將自行簽署的證書輸入至用戶端的信任儲存。

輸入證書的步驟會因連接至「發行者」通道之 HTTPS 監聽程式的用戶端而有所不同。如果用戶端使用一般的 Java *KeyStore*，您可以執行下列步驟來建立 *KeyStore*：

- 3a 使用任意 Java *JDK** 中包含的 *keytool* 可執行檔。

如需 *keytool* 的相關資訊，請參閱 [Keytool - 金鑰與證書管理工具 \(http://java.sun.com/j2se/1.4.2/docs/tooldocs/windows/keytool.html\)](http://java.sun.com/j2se/1.4.2/docs/tooldocs/windows/keytool.html)。

- 3b 在指令提示符中輸入下列指令：

```
keytool -import -file name_of_cert_file -trustcacerts -noprompt  
-keystore filename -storepass password
```

例如：

```
keytool -import -file tree_ca_root.b64 -trustcacerts -noprompt -keystore dirxml.keystore -  
storepass novell
```

- 4 設定「發行者」以使用您在步驟 1 中建立的伺服器證書。
 - 4a 在 iManager 中，按一下「*Identity Manager* > *Identity Manager* 概觀」。
 - 4b 找出內含 SOAP 驅動程式的驅動程式集，然後按一下驅動程式的圖示，以顯示「*Identity Manager* 驅動程式概觀」頁面。
 - 4c 在「*Identity Manager* 驅動程式概觀」頁面上，再按一下驅動程式的圖示，然後捲動至「發行者設定」。

4d 在「KMO 名稱」設定中，指定您在**步驟 1**中使用的證書綽號。

5 按一下「套用」，然後按一下「確定」。

4.4 建立 XSLT 樣式表

若要讓 SOAP 驅動程式能與 DSML 或 SPML 預設組態以外的任何設定搭配使用，便需要建立 XSLT 樣式表。應用程式特定的協定處理會在「輸入轉換」和「輸出轉換」樣式表中完成。

如需如何撰寫樣式表以處理其他文件類型的詳細資訊，請參閱此驅動程式隨附的範例樣式表。如需樣式表的相關資訊，請參閱《[規則產生器和驅動程式自訂指南](#)》中的「[使用 XSLT 樣式表定義規則](#)」。

4.5 操作資料

驅動程式 Shim 會根據 <operation-data> 元素，將特殊處理套用至「訂閱者」指令。在「訂閱者」通道上，可以將 <operation-data> 元素新增至指令，以達到以下兩個目的。

1. 指定想要包含在指令結果中的 XML 資料。這樣便可以比對指令與其產生的回應，對於建立關聯非常有用。
2. 預設「訂閱者」選項可以根據不同的指令進行置換。

如**第 1 章「綜覽」**，**第 5 頁**中所述，<operation-data> 元素可以從某個「訂閱者」通道規則新增至指令。驅動程式 Shim 會先從指令中移除操作資料，然後再將其傳送到應用程式，並將 <operation-data> 元素（及其所有子元素）還原到產生的回應中。隨後，規則和樣式表可以視需要在結果中存取 operation-data 元素。

4.5.1 使用操作資料指定要傳回到結果中的 XML

SOAP 驅動程式的範例組態使用 <operation-data> 元素追蹤指令的辨識資訊，以辨識結果並正確指定關聯。如需如何使用 <operation-data> 元素的詳細資訊，請查看這些範例。

當在回應中還原 <operation-data> 元素時，該元素會附加為根節點的子元素。您也可以透過為 <operation-data> 元素提供一或多個 parent-node-*n* 屬性進行置換，其中，*n* 是從 1 開始的數字，且每當要提供一個父指定器時，該數字便會遞增。驅動程式 Shim 會尋找 parent-node-*n* 屬性。找到後便檢查屬性，查看是否存在具名節點。如果找到該節點，它便會在回應中做為 <operation-data> 元素的父節點。

4.5.2 使用操作資料置換預設訂閱者選項

有兩種方法可以置換指令的預設「訂閱者」選項。

1. 在組態中建立多個名為連接的「訂閱者」選項集，並使用 <operation-data> 元素指定用於目前指令的連接集。
2. 指定特定選項以置換目前指令，如 url、方法或 soap-action。

建立和使用多個訂閱者選項集（連接）

若要使用 <operation-data> 元素置換預設「訂閱者」連接參數：

- 1 編輯驅動程式組態的「訂閱者」設定區段。

- 2 使用 iManager 的 XML 編輯功能，找到以橫線和數字 1 結尾的「訂閱者」設定 (例如 subURL-1)，然後複製並遞增數字。

例如：subURL-2

- 3 將新的設定值設定為您要用於第二個連接的值。

您可以使用這種方式設定任意個連接，但設定的數值需要按順序遞增。

- 4 將屬性新增到名為 connection 的 <operation-data> 元素，並為其指定要使用的連接數值。

例如：

```
<operation-data connection="2"> ... (@%•L operation-data $³Ø¿) </
operation-data>
```

置換單一訂閱者選項

您可以直接使用 <operation-data> 元素的屬性僅置換 url、HTTP 方法或 soap-action 值，而不必使用連接概念置換多個「訂閱者」選項。下表列出可以使用的屬性以及這些屬性要置換的「訂閱者」選項。

表格 4-1 用於置換訂閱者選項的屬性

<operation-data> 屬性	被置換的訂閱者選項	描述
url	subURL-1	這是 Web 服務或 HTTP 應用程式的 URL (或 URI)。如果應用程式有兩個不同的 Web 服務分別用於新增和刪除使用者，置換 URL 可能會很有用。
方法	subHttpMethod-1	這方法預設為 POST，如有需要，也可以設定為 RFC 2616 區段 9 中所定義的其他方法。
soap-action	具有關鍵字 "SOAPAction" 的 HTTP 申請標題欄位	在 DSML 與 SPML 範例中，此值始終為 #batchRequest。不過，某些 Web 服務需要變更此值，這取決於指令。

範例：

```
<operation-data url="http://137.66.10.13:18180/soap">
... (¶p¶Šª >|n¶ A@œ•@®%•L operation-data $³Ø¿) </operation-data>
```

```
<operation-data $È™k="GET"> ... (¶p¶Šª >|n¶ A@œ•@®%•L operation-data
$³Ø¿) </operation-data>
```

```
<operation-data soap-action="addUser"> ... (¶p¶Šª >|n¶ A@œ•@®%•L
operation-data $³Ø¿) </operation-data>[XXX]
```

使用驅動程式

驅動程式安裝完成並輸入範例組態檔案後，還必須完成下列任務：

- ◆ 「[啟動驅動程式](#)」，第 27 頁
- ◆ 「[移轉並重新同步資料](#)」，第 27 頁
- ◆ 「[啓用驅動程式](#)」，第 27 頁

5.1 啟動驅動程式

- 1 在 iManager 中，按一下「[Identity Manager > Identity Manager 概觀](#)」。
- 2 瀏覽至驅動程式所在的驅動程式集，選取此驅動程式集，然後按一下「[搜尋](#)」。
- 3 按一下 SOAP 驅動程式圖示的右上角，然後按一下「[啟動驅動程式](#)」。

若要進一步設定啟動選項，請參閱「[設定驅動程式設定](#)」，第 19 頁。

5.2 移轉並重新同步資料

當資料發生變更時，Identity Manager 會同步化資料。如果您要立即同步化所有資料，可以選擇下列選項：

- ◆ 從 **Identity Vault** 移轉資料：可讓您選取要從 Identity Vault 移轉至應用程式的容器或物件。當您移轉物件時，Identity Manager 引擎會將所有的「相符」、「佈置」、「建立」規則以及「訂閱者」過濾器都套用至該物件。
- ◆ 移轉資料至 **Identity Vault**：假設可以從遠端應用程式（通常為 Web 服務）查詢到符合發行者過濾器中準則的項目。不過，由於 SOAP 驅動程式的一般特性，查詢 Web 服務的方法（如果確實存在一種方法）對於驅動程式 Shim 來說是未知的，所以，此功能通常對於 SOAP 驅動程式無效。
- ◆ 同步化：Identity Manager 引擎會在「訂閱者」類別過濾器中尋找並處理這些類別的所有物件。會合併相關聯的物件。會將取消關聯的物件做為「新增」事件處理。

若要使用上述其中一個選項，請執行下列動作：

- 1 在 iManager 中，按一下「[Identity Manager > Identity Manager 概觀](#)」。
- 2 瀏覽至驅動程式所在的驅動程式集，選取此驅動程式集，然後按一下「[搜尋](#)」。
- 3 按一下驅動程式圖示。
- 4 按一下適當的移轉按鈕。

5.3 啓用驅動程式

您必須在安裝後的 90 天內啓用驅動程式，否則驅動程式將停止工作。

如需啓用資訊，請參閱《[Novell Identity Manager 3.0 管理指南](#)》中的「[啓用 Novell Identity Manager 產品](#)」。

驅動程式疑難排解

本節包含下列有關錯誤訊息的資訊：

- ◆ 「驅動程式 Shim 錯誤」，第 29 頁
- ◆ 「Java 自訂錯誤」，第 30 頁

6.1 驅動程式 Shim 錯誤

下表識別可能發生於核心驅動程式 Shim 的錯誤。含有數字代碼的錯誤訊息視應用程式或 Web 服務的不同，可以包含多條訊息。

錯誤訊息	層級	描述
307 暫時重新導向	重試	「訂閱者」通道嘗試將資料傳送至應用程式或 Web 服務，但是卻收到「307 暫時重新導向」的回應。 「訂閱者」將等待一段時間（通常是 30 秒），然後再重試。
408 申請逾時	重試	「訂閱者」通道嘗試將資料傳送至應用程式或 Web 服務，但是卻收到「408 申請逾時」回應。 「訂閱者」將等待一段時間（通常是 30 秒），然後再重試。
503 服務無法使用	重試	「訂閱者」通道嘗試將資料傳送至應用程式或 Web 服務，但是卻收到「503 服務無法使用」回應。 「訂閱者」將等待一段時間（通常是 30 秒），然後再重試。
504 閘道逾時	重試	「訂閱者」通道嘗試將資料傳送至應用程式或 Web 服務，但是卻收到「504 閘道逾時」回應。 「訂閱者」將等待一段時間（通常是 30 秒），然後再重試。
上表尚有幾種數字錯誤碼未列出。	錯誤	<p>HTTP 伺服器（「訂閱者」通道可能會與之通訊的那些伺服器）會傳回數字值及一條簡短的描述性訊息，用以指出申請的狀態。</p> <p>200-299 範圍內的數字表示成功，因此不會產生錯誤訊息。</p> <p>上面所列的數字（307、408、503 及 504）表示暫時狀況，因此會重試申請。</p> <p>其他數字錯誤碼會產生錯誤訊息，這條錯誤訊息會包含該代碼及 HTTP 伺服器所提供的訊息。在大部分情況下，驅動程式會繼續執行，而導致錯誤的指令則不會重試。</p>

錯誤訊息	層級	描述
與 HTTP 伺服器通訊發生問題。確定伺服器在執行中且正在接受申請。	重試	<p>在通訊或嘗試與 HTTP 伺服器通訊的過程中，「訂閱者」通道收到一個 <code>IOException</code>。</p> <p>您收到此錯誤，可能是因為伺服器不執行、發生多載、因防火牆或其他限制而無法進行存取，或「訂閱者」組態中提供的 URL 不正確。</p> <p>導致此錯誤的指令會在稍後重試。</p>
<p>在預設狀態下，HTTP/SOAP 驅動程式不會傳回任何應用程式綱要。</p> <p>如果您想讓 Shim 報告某個應用程式特定綱要，則可以撰寫自己的 Java 類別來實作 <code>SchemaReporter</code> 介面，然後再設定驅動程式將此類別做為 Java 延伸功能載入。</p>	警告	<p><code>Metadirectory</code> 引擎會呼叫驅動程式的 <code>DriverShim.getSchema()</code> 方法，但是驅動程式尚未使用 <code>SchemaReporter</code> 自訂進行延伸。</p> <p>驅動程式會繼續執行。</p>
<p>已呼叫 <code>Subscriber.execute()</code>，但是未正確設定「訂閱者」。已忽略指令。</p> <p>您應該設定「訂閱者」或清除「訂閱者」的過濾器，這樣就不會收到指令。</p>	警告	<p>未正確初始化驅動程式的「訂閱者」通道。最有可能的原因是驅動程式組態格式不正確。</p> <p>驅動程式會繼續執行，但是每當「訂閱者」通道收到事件時都會顯示此訊息。</p>
pubHostPort 必須使用 host:port 格式	嚴重錯誤	<p>「發行者」通道組態造成的錯誤。</p> <p>檢視「發行者」通道參數，以確認提供的主機及連接埠號碼都有效。</p>
MalformedURLException	嚴重錯誤	「訂閱者」通道參數中提供的 URL 不是有效的 URL 格式。
多個例外	嚴重錯誤	無法正確初始化 HTTP 監聽程式時，此訊息會出現在追蹤中。導致此狀況發生的原因有許多。請檢查您的「發行者」設定，以確定所指定的連接埠未被使用且其他「發行者」設定正確。
HTTPS 主機名稱錯誤：應該...	重試	<p>「訂閱者」通道上無法進行 SSL 信號交換時，會出現此訊息。這表示伺服器證書所提供的接收者與 HTTPS URL 中指定的 IP 位址或主機名稱不符。</p> <p>請在 URL 中使用 DNS 主機名稱，而不要使用 IP 位址。</p>

6.2 Java 自訂錯誤

下表識別自定 Java 延伸功能中可能發生的錯誤。

訊息	層級	描述
SchemaReporter 啓始問題： <i>延伸功能特定的訊息</i>	嚴重錯誤	SchemaReporter Java 自訂在啓始時發生問題。驅動程式關閉。

訊息	層級	描述
延伸功能 (自定碼) 啓始問題： <i>延伸功能特定的訊息</i>	嚴重錯誤	無法啓始化下列其中一項 Java 延伸功能： <ul style="list-style-type: none"> ◆ SubscriberTransport ◆ PublisherTransport ◆ DocumentModifiers ◆ ByteArrayModifiers 驅動程式關閉。
其他各種錯誤	不一定	為 Java 延伸功能提供的介面會將錯誤訊息傳回至追蹤螢幕，偶爾還會傳回至 Identity Manager 引擎。 有時很難將此類錯誤與核心驅動程式 Shim 中產生的其他錯誤加以區分。如果此表中未列出您在使用 Java 延伸功能時遇到的錯誤，請向為您提供延伸功能的人員索取該特定延伸功能的錯誤碼清單。

使用 Java 延伸功能

A

Identity Manager Driver for SOAP 的功能可以使用 Java 進行延伸。使用 Java 介面定義的應用程式介面 (API) 就可以建立自己的自定 Java 類別，讓它能存取經由「訂閱者」通道和「發行者」通道傳送的資料。這些類別可以讀取並解譯資料，並可以選擇性地修改資料。此外還定義了 Java 介面，讓您可以使用自己自定的訂閱者或發行者來取代使用 HTTP 的預設訂閱者或發行者。

本節包含下列有關使用 Java 延伸功能的資訊：

- ◆ 「綜覽」，第 33 頁
- ◆ 「建立和設定 Java 延伸功能」，第 34 頁

A.1 綜覽

如果與 Identity Manager Driver for SOAP 搭配使用的應用程式使用 XML 格式以外的資料，則可以建立 Java 延伸功能，將非 XML 格式的資料轉換成 XML 資料。此外，您可能還會變更各種通訊協定，包括 XML 和 HTTP。例如，可能會取代預設的 HTTP。這些 Java 延伸功能可以用於運作資料，且它們必須用於將非 XML 格式的資料轉換為 XML 資料。如下圖所示，共有十一個點可以延伸功能：

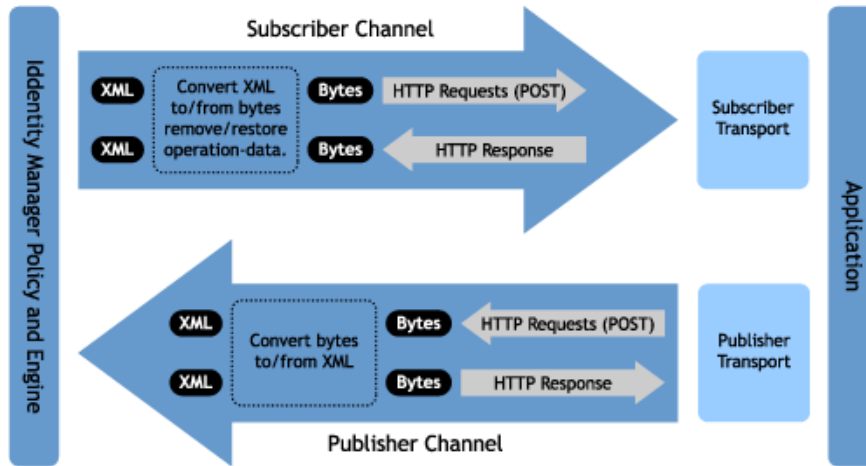
- ◆ 訂閱者通道中有四個
- ◆ 發行者通道中有四個
- ◆ 兩個用於指定傳輸
- ◆ 一個用於報告應用程式綱要

SOAP 驅動程式的設計相當靈活並可延伸。對於想要延伸或修改驅動程式功能的 Java 程式設計人員，還有程式設計介面可供使用。這些介面只能在無法根據規則或樣式表進行所需的轉換時使用。

Javadoc (<http://www.novell.com/documentation/beta/dirxmldrivers/javadoc/api/index.html>) 對這些介面進行了描述。

可以用於延伸或自定驅動程式行為的 Java 介面有五個。它們分別是 DocumentModifiers、ByteArrayModifiers、PublisherTransport、SubscriberTransport 和 SchemaReporter。

特性 **A-1** 如何使用 Java 延伸功能



DocumentModifiers 和 ByteArrayModifiers 的用途類似，因此二者選其一即可。如有必要，它們都可以用於存取及修改經由驅動程式 Shim 傳送的指令和事件。DocumentModifiers 可讓您存取 XML DOM 文件等資料。ByteArrayModifiers 可讓您存取相同資料，但是會將資料序列化為位元組陣列。

PublisherTransport 介面可讓您將驅動程式在「發行者」通道上使用的預設 HTTP 監聽程式取代掉。PublisherTransport 實作可以由事件驅動，也可以按指定的間隔輪詢。

如果想要取代掉驅動程式在「訂閱者」通道上使用的 HTTP 或 HTTPS 連接，則應實作 SubscriberTransport。

如果您能以程式設計方式決定遠端 Web 服務使用的類別和屬性，則可以使用最後一種介面 SchemaReporter。使用這個介面的優勢在於，如果可以動態決定綱要，則建立綱要映射規則就更為容易。

A.2 建立和設定 Java 延伸功能

以 Novell 開發人員下載網站 (<http://developer.novell.com/ndk/downloadaz.htm>) 上找到的範例碼和 SOAP Driver Javadoc 做為指南，撰寫類別的 Java 程式碼。在 A-Z 清單中，搜尋「SOAP 驅動程式」。命名類別時，應使用您的環境和組織所適用的 Java 套件和類別名稱。

例如，如果撰寫您自己的實作 DocumentModifiers 介面的類別，並在 com.novell.idm 套件中將類別命名為 MyDocumentModifiers，則需要執行下列步驟進行編譯、產生 jar，以及部署您的類別：

1 準備環境

請確定電腦上已安裝最新的 Java Development Kit (JDK)。如果需要下載，請造訪 Java 網站 (<http://java.sun.com/>)。

2 在根據套件命名定義的適當目錄結構中蒐集來源碼。

在以上範例中，您應該已有一個包含 novell 目錄的 com 目錄，其中 novell 目錄還包含 idm 目錄。在 idm 目錄中有一個名為 MyDocumentModifiers.java 的來源檔案。

- 3 請確定已擁有編譯類別所需的 jar 檔案。
至少要有 SOAPUtil.jar。如果類別中使用 XML 文件，則還需要 nxsl.jar。
- 4 將所需 jar 檔案的副本置於方便的位置，比如 com 目錄外編譯目錄的根目錄，然後以該位置為目前目錄，存取系統指令提示符或外圍程序提示符。
- 5 輸入以下其中一項來編譯類別：
 - ◆ 對於 Windows：javac -classpath SOAPUtil.jar;nxsl.jar com\novell\idm*.java
 - ◆ 對於 Linux 或 UNIX：javac -classpath SOAPUtil.jar:nxsl.jar com/novell/idm/*.java
- 6 輸入以下其中一項，建立包含類別的 Java 歸檔檔案：
 - ◆ 對於 Windows：jar cvf mydriverextensions.jar com\novell\idm*.class
 - ◆ 對於 Linux 或 UNIX：jar cvf mydriverextensions.jar com/novell/idm/*.class
- 7 將**步驟 6**中建立的 jar 檔案置於 SOAPShim.jar 所在的目錄。
在 Windows 中，一般為 C:\Novell\NDS\lib。
- 8 在 iManager 中，編輯驅動程式設定。
 - 8a 在「自定 Java 延伸功能」旁選取「顯示」。
 - 8b 在「文件處理」旁選取「實作」。
 - 8c 指定 *com.novell.idm.MyDocumentModifiers* 做為「類別」的值，並指定任意字串做為 Init 參數。

init 參數是傳遞到類別之 init 方法的字串，因此會將要在類別啓始化期間使用的資訊放在此處。
- 9 重新啓動驅動程式。

現在，您便可以使用自定類別。