

章節 2. OPC Tunnelling

目錄

- 2.1. 簡介
- 2.2. 為server設定DataHub
- 2.3. 為client設定Cogent DataHub
- 2.4. 測試連線
- 2.5. Tunnelling部分的資料集
- 2.6. 擴充應用
 - 2.6.1. Tunnelling和橋接
 - 2.6.2. Tunnelling和彙總

2.1. 簡介

當談及資料網路、*Tunnelling*時，是指把一個通訊協定封裝進另一個通訊協定，以使其能更容易的被發送和/或安全地橫跨網路。Cogent DataHub提供OPC tunnelling避免DCOM設定的問題。



Tunnelling和mirroring是一樣的，如同[章節 5.4](#)，“Excel的網路功能”所述。Tunnelling也可以用來連接到執行在Linux的Cascade DataHub。請參閱用於Linux和QNX之Cascade DataHub使用手冊的鏡像資料章節以獲得更多細節。

使用Cogent DataHub的OPC tunnelling代表：

- 橫跨一個LAN或Wan同樣能輕鬆連接。
- 不需要DCOM - 沒有逾時或是設定問題。
- 完整和安全的資料存取。
- 簡單地建立。

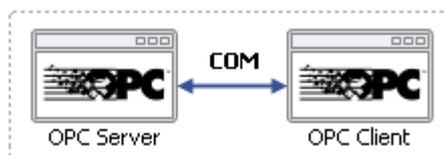
建立Cogent DataHub的OPC tunnelling是個簡單、幾分鐘就能完成的3步驟設定。你只需要做：

1. 在OPC server的機器中設定DataHub。
2. 在OPC client的機器中設定DataHub。
3. 啟動OPC client。

為什麼要替OPC DA使用tunnelling?

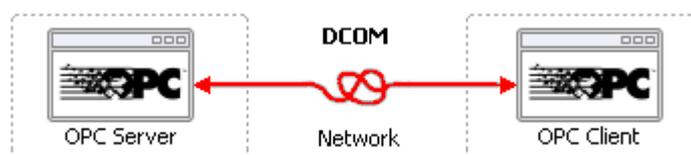
嘗試網路連接OPC DA servers和clients的大多數人們會遇到許多問題。網路功能不是OPC的強項。OPC最初是根據執行在單一電腦上的COM (component object model)。

Easy:
OPC on a
single node
using COM.

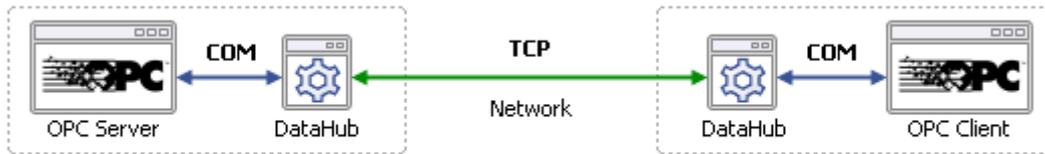


只要OPC server和OPC client是在一台電腦上，建立它們之間的連接非常容易。但是，當OPC server和client是在不同電腦上，且需要網路連接時，就會出現困難。

Difficult:
OPC over
a network
using DCOM.



欲透過網路通訊，OPC使用DCOM(分散式COM)，其中許多系統工程師發現它不適合他們的需求。DCOM在每次的逾時可長達5分鐘，隨著網路連接OPC servers和clients的數量增加，DCOM的困難也會呈指數級增長。DCOM通過LAN連接是非常困難的，大多數人甚至不通過WAN來嘗試它。



Solution: Eliminate DCOM. Use COM at each end and TCP over the network.

Cogent DataHub提供一個COM(OPC)介面給OPC client和server，並使用TCP橫跨網路。這就是所謂的tunnelling。每個已連接的OPC server或client把其他視為本機OPC連接。它們無法察覺到它們的訊息完全是被轉換為TCP。Tunnelling橫跨LAN或WAN也同樣地運作良好，它甚至可以通過防火牆來進行tunnel。

版權所有 © 1995-2011 by Cogent Real-Time Systems, Inc.

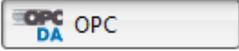
2.2. 為server設定DataHub

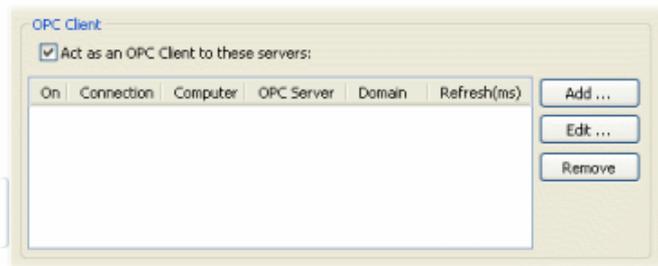
把DataHub設定成爲OPC client

OPC server機器裡的DataHub會執行成爲OPC client。您應該設定如下:

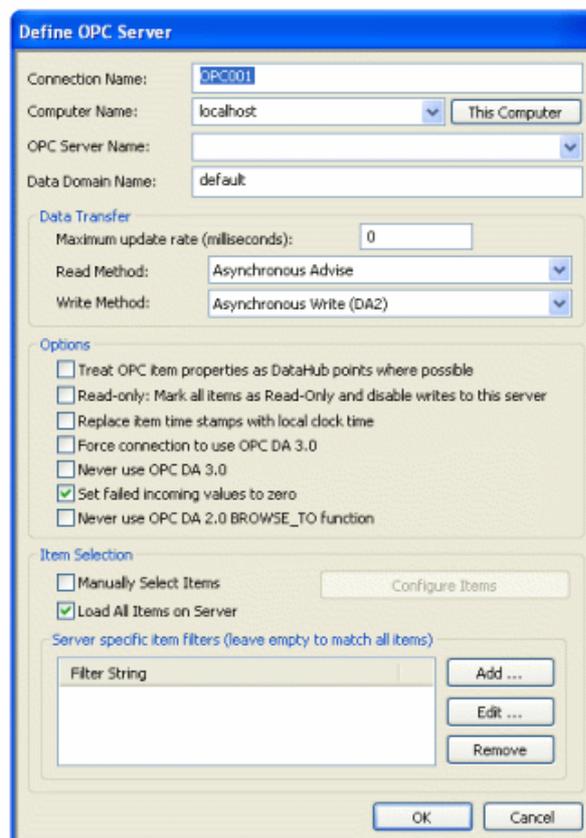


1. 在Cogent DataHub系統圖示點擊右鍵，接著選取**Properties**。

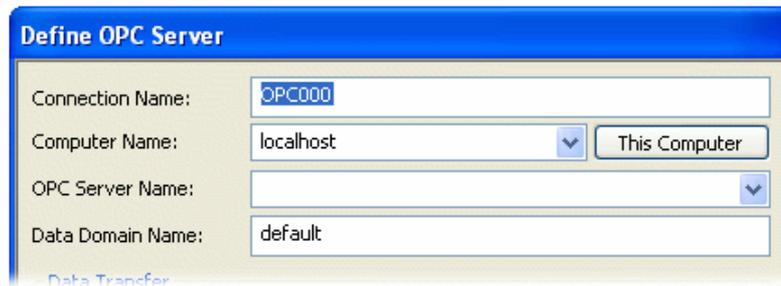
2. 在屬性視窗中，選取**OPC** 。



3. 勾選**Act as an OPC Client**方框。因爲DataHub不只是一個OPC server而已，它也可以是client，所以您必須指定想要連接的OPC server。欲新增一個server，請點擊**Add**按鈕 並在**Define OPC Server**視窗上填入:

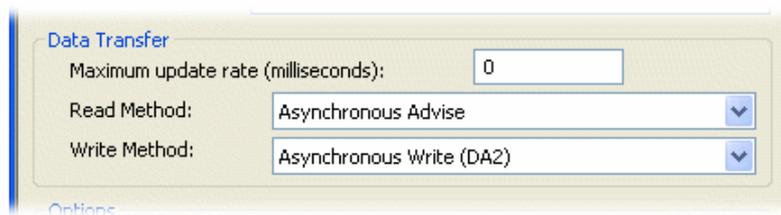


4. 輸入或選取適當的必要資訊。
 - a. 上方的四個欄位定義OPC server:



- **Connection Name:** 輸入名稱來識別該連線，名稱中不可有空白鍵。名稱的選擇不重要，但是必須和其他連線名稱不同。
- **Computer Name:** 輸入您想連接到的執行OPC server電腦的電腦名稱或IP位址，或者是從下拉式清單中選取。
- **OPC Server Name:** 從可供使用的servers清單中選取您想要連接之OPC server的名稱。
- **Data Domain Name:** 輸入DataHub Data Domain名稱，就會顯示它的資料點。

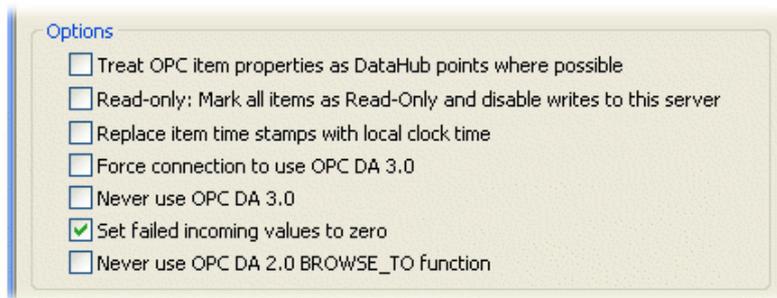
b. 您可以指定資料以何種方式傳送。



- **Maximum update rate (milliseconds):** 輸入您想要的最大資料更新率。這對讓傳入資料減慢速度是非常有用的。它的預設值為0，這會使值儘可能地快速更新。這個值也是用於非同步和同步讀取的輪詢時間(見下文)。
- **Read Method:** 選擇如何從OPC server讀取資料的方式:
 - **Asynchronous Advise** 無論何時，只要點值變更，OPC server就會立即發送一個設定點的資料給DataHub。這是最有效的選項，也具有最少的延遲。
 - **Asynchronous Read** DataHub在時間間隔內輪詢OPC server所有已設定的資料點(由**Maximum update rate**設定)。這個選項的效率比Asynchronous Advise低，並具有較高的延遲。
 - **Synchronous Cache Read** DataHub在時間間隔內輪詢OPC server所有的設定點(由**Maximum update rate**設定)，而且此線程會等待回應。這個選項的效率比Asynchronous Advise和Asynchronous Read低，並且比這兩者具有更高的延遲。
 - **Synchronous Device Read** DataHub在時間間隔內輪詢PLC或其他連接到OPC server所有設定點的硬體裝置(由**Maximum update rate**設定)，而且此線程會等待回應。這是所有選項中最沒有效率的方法，並具有最高的延遲。
- **Write Method:** 選取如何寫入資料到OPC server的方式:
 - **Asynchronous Write** 提供更高的效能。Cogent DataHub將點值的變更寫入OPC server，無需等待任何回應。
 - **Synchronous Write** 從OPC server引出一個更迅速的回應，但會導致整體效能降低。Cogent DataHub把點值的變更寫入OPC server而無需等待回應。當OPC server完全不支援非同步，或是它無法處理大量寫入時，這個選項就很有用。

根據您正在設定的OPC server，您也許可以選擇使用OPC DA 2.0或3.0。請參閱屬性視窗章節裡OPC小節的[資料傳輸](#)說明以了解更多資訊。

c. 這裡有幾個選項:



- **Treat OPC item properties as DataHub points** 選項 讓您 登錄和使用非標準OPC item屬性作為DataHub的點。通常您不需要這個選項，除非您計畫使用DataHub來區分OPC items裡非標準屬性的值變更。



Cogent DataHub只會在OPC server開放這些屬性做為OPC items時才會監視這些屬性。如果在使用核取方塊沒有顯示該屬性，就表示server不會開放非標準的屬性作為items。



某些OPC servers登錄其OPC items和屬性時很耗時，與這其中一個server使用該選項會讓DataHub的啟動時間變得非常緩慢。

- **Read only: Mark all items as Read-Only** 選項 讓您指定OPC server為唯讀，不管個別items是如何被指定。來自這種OPC server之DataHub裡的items，只能讓所有的DataHub clients作唯讀。
- **Replace item time stamps with local clock time** 選項 讓您把此server的items時間戳記設定為本機的時間。
- **Force connection to use OPC DA 3.0** 選項 這個選項能讓使用者從Write Method的下拉式方塊中選取DA 3.0寫入方法。它也指示Cogent DataHub使用DA 3.0瀏覽來嘗試瀏覽server。這個設定會根據server的登錄項目來覆寫任何Cogent DataHub可能會定義之有關server的任何自動資訊。
- **Never use OPC DA 3.0** 選項 這個設定會從Write Method下拉式方塊中移除DA 3.0寫入方法，並且會指示Cogent DataHub只使用DA 2.0瀏覽。這個設定會根據server的登錄項目來覆寫任何Cogent DataHub可能會定義之有關server的任何自動資訊。

欲了解更多有關OPC DA 2.0和3.0的資訊，請參閱屬性視窗章節裡OPC小節的[資料傳輸](#)說明以了解更多資訊。

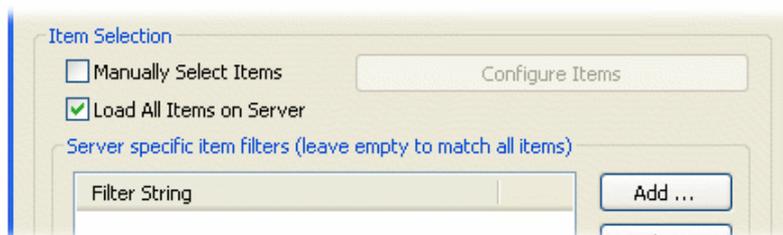
- **Set failed incoming values to zero** 選項 OPC規格要求OPC server無論何時在它發送一個失敗碼以對一個item變更或讀取請求作出回應時，必須發送一個EMPTY(zero)值。然而，某些OPC servers在某些情況下會發送包含失敗碼的有效值。欲忽略任何來自OPC server該類型的值並假定為EMPTY，保留此方框為勾選狀態(預設)。反之，如果您想要使用由OPC server提供的值，請取消勾選該方框。



取消勾選該方框會使Cogent DataHub的行為不符合OPC規範。

- **Never use OPC DA 2.0 BROWSE_TO function** 選項 此設定使得與OPC DA 2 servers通訊時，將不允許BROWSE_TO函式。有時候，使用這個函式會使OPC server出現問題，這會阻止Cogent DataHub連接到OPC server。勾選此方框可能會在這些情況下允許連線被建立。

d. 最後，您可以指定OPC items被指定的方式。您可以手動的選取或是載入全部的item。



手動選取Items



[Click here to watch a video.](#)



勾選**Manually Select Items**方塊並按下**Configure Items**按鈕來 開啟OPC Item Selection視窗，您可以在此視窗裡指定您想要使用的點：



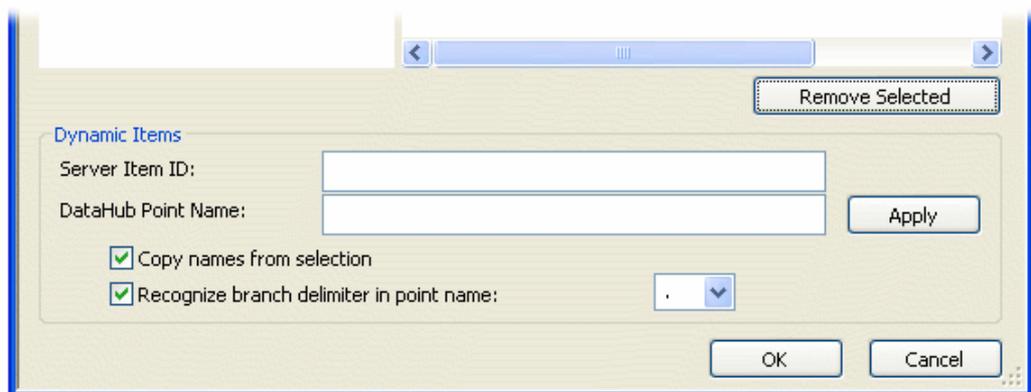
您可以透過左側窗格的樹狀目錄瀏覽，選取想要的點，這些選項會出現在右側窗格。按照這些指南以作出選擇：

- 欲從右側窗格選取一個server item，請點擊它的核取方塊。
- 欲反白連續的server items清單，點擊第一個item，按住**Shift**鍵，接著點擊最後一個item。欲反白分開的server items，當您選取每個item時，也按住**Ctrl**鍵。欲選取反白item的群組，請使用**Spacebar**。



在Windows NT或Windows 2000作業系統下，可能無法如描述般正常運作。

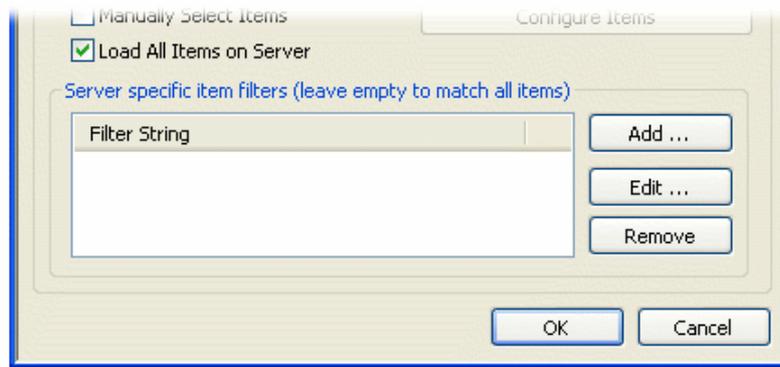
- 選取一個server item並不會自動新增任何子items。每個子item必須個別新增。欲檢視子items，點擊item前的+號。如果一個item裡有一個或更多子item被指定，item名稱就會以粗體呈現。
- 欲刪除所選的items，請在右側窗格將items反白並按**Remove Selected**按鈕。如上述般使用**Shift**和**Ctrl**鍵來反白指定items的群組。



您也許還可以在server上設定動態item。當您輸入**Server Item ID**時，系統會替您填進一個相同的**DataHub Point Name**(您可以隨時變更)。按**Enter**鍵或是**Apply**按鈕來建立item。勾選**Copy names from selection**方框會填入您從Selected Items清單裡(如上)所選取的輸入項目名稱。**Recognize branch delimiter in point name**選項 讓您為您的動態items選取並套用point分隔符號。

在Server上載入所有的Items

除了手動載入items，您也可以選擇在Define OPC Server對話框登錄所有的點，或從OPC server對點的群組進行篩選。



在**Server specific item filters**裡，您可以選擇建立一個過濾條件來選取部份資料集。如果您在這欄留白，DataHub會查詢OPC server的所有items並登錄。這些過濾條件會全部套用在邏輯'或'基準上，舉例來說，如果一個點滿足任何過濾條件，就會被登錄到DataHub。

- 點擊**Add...** 按鈕來新增一個篩選條件。**Edit a filter string**視窗顯示如下：



輸入一個字串或樣式來符合OPC server裡一個或多個item名稱。每個server有它自己樣式比對的語法，所以您可能要實驗一下以得到您需要的點。通常，符號*會符合任意數量的字元，而符號?通常只符合一個字元。在這種情況下，一個輸入項目?a*會帶來所有名稱的第二個字母裡有a的items。

- 點擊**Edit...** 按鈕來開啓**Edit a filter string**視窗並編輯一個現存的篩選條件。您也可以藉由連接兩下清單裡的篩選字串來開啓**Edit a filter string**視窗並編輯。
 - 點擊**Remove** 按鈕以從清單中移除一個特定的過濾條件。
5. 在屬性視窗點擊**Apply** 按鈕，DataHub應該會開始執行成爲OPC server的client。您可以使用**Data Browser**或是**Connection Viewer**來驗證，也可以隨時變更這些設定。當您在屬性視窗中點擊**Apply** 按鈕時，Cogent DataHub就會重新連線，並且套用所做的變更。

把DataHub設定爲tunnelling master

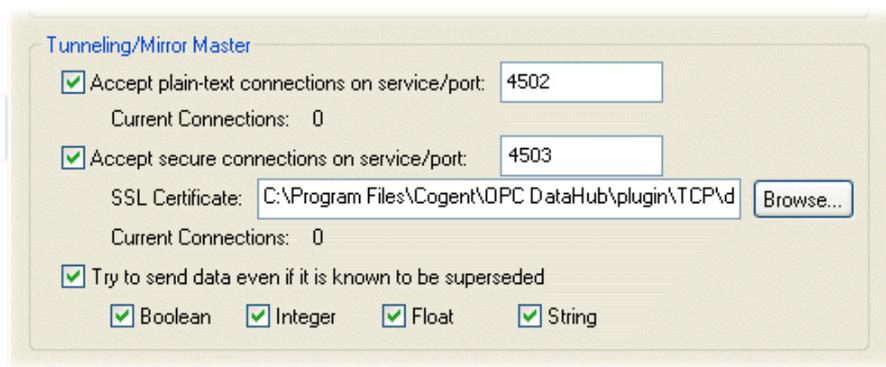
在最初或是連線中斷後，tunnelling master DataHub會接收來自tunnelling slave的初始要求來建立tunnelling連接。基於這個原因，我們建議任何兩個tunnelling DataHub，其master要在OPC server機器裡。只要連線被建立，是無法分辨這兩個DataHubs的，兩個DataHubs都會發送和接收資料變更。



1. 在Cogent DataHub系統圖示點擊右鍵，接著選取**Properties**。

2. 在屬性視窗中，選取**Tunnel/Mirror**





3. 在**Tunnelling Master**部分，您可以設定純文字或是安全的tunnelling。確保至少勾選其中一個。如果您想要變更任何其他預設值，請參閱**章節 18.4**，**“Tunnel/Mirror”**以了解更多資訊。

 欲最佳化傳輸量，取消勾選**Try to send data even if it is known to be superseded**選項。這會讓 DataHub在client已通知原始變更之前為已經變更的值卸除過時的值。最新的值會一直被傳送。

4. 點擊**OK**來關閉Properties視窗。

OPC server tunnelling連接的機器端已經準備好了，現在您可以移至OPC client機器。

2.3. 為client設定Cogent DataHub

現在，您需要在該機器上安裝Cogent DataHub以便tunnel橫跨到OPC server機器裡的DataHub。

設定Cogent DataHub為tunnelling slave

Tunnelling slave DataHub完全就像tunnelling master DataHub一樣，除了slave會在最初時建立tunnelling連線，以及在網路中斷後重新建立連線。基於這個原因，我們建議在OPC client端的DataHub執行成為tunnelling slave，而在OPC server端的DataHub執行成為tunnelling master。



Click here to watch a video.

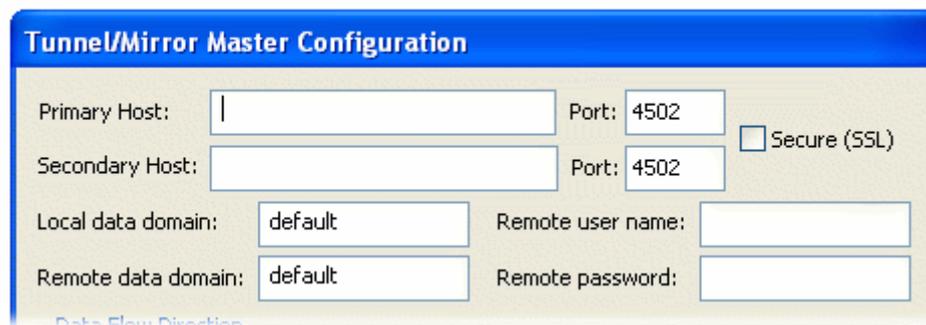


1. 在Cogent DataHub系統匣圖示點擊右鍵，接著選取**Properties**。

2. 在屬性視窗中，選取**Tunnel/Mirror** 。



3. 勾選**Act as a tunnelling/mirror slave to these masters**方框。
4. 點擊**Add Master...**按鈕以指定slave的master。會開啓**Tunnel/Mirror Master Configuration** window 視窗如下：



5. 輸入以下資訊：
 - **Primary Host:** 執行tunnelling master DataHub之電腦的名稱或IP位址。
 - **Port:** 主機的連接埠名稱或service名稱。您應該使用預設的連接埠名稱(4502)，除非您已經變更master DataHub的輸入項目。
 - **Secondary Host:** 讓您能選擇擁有備用主機和service/連接埠號碼。在啓動時或連線中斷後，DataHub會先搜尋主要的主機，接著才會搜尋次要主機，在主要和次要主機之間交替，直到建立連線。如果沒有指定次要主機，就只會在主要主機上嘗試連線。
 - **Local data domain:** 輸入您計畫接收資料的Data Domain。
 - **Remote data domain:** 您想要從中接收資料的master DataHub Data Domain。點名稱會從遠端Data Domain(在master DataHub中)對應到本機Data Domain(在該DataHub中)，反之亦然。



除非您有好的理由需要讓這些不同，否則，爲了簡單化，我們建議兩個DataHub都使用相同的Domain名稱。



以下列出的是在Cogent的server上執行的DataHub，您可以連接它以進行測試。下面是您需要輸入的參數：

- **Primary Host:** developers.cogentrts.com
- **Port:** 4502
- **Local data domain:** test
- **Remote data domain:** test

6. 您現在有鏡像連接方式的多種選擇。

a. **Data Flow Direction:** 讓您決定資料流的方式。slave和master之間的預設值是雙向資料流，但您可以藉由選取個別選項來有效地建立一個唯讀或唯寫的連線。



欲最佳化傳輸量，勾選**Read-only: Receive data from the Master, but do not send**選項。只有當您真的想要一個唯讀連線時才勾選該選項。如果您不要求讀寫雙向的存取，唯讀的tunnel速度會更快。

b. **When the connection is initiated:** 當slave第一次連接到master時，決定點的值如何被指定。這有三種可能性：slave取得來自master的所有值、slave發送其所有值到master、或master和slave同步它們的資料集、逐一地，根據每個點的最新值(預設)。

c. **When the connection is lost:** 決定"未連線"時，要在哪裡顯示資料品質—在master、在slave、或是兩者皆非。



如果您已經設定**When the connection is initiated**爲**Synchronize based on time stamp**(如上文)，那麼該選項必須設爲**Do not modify the data quality here or on the Master**以取得正確的資料同步處理。

d. **Connection Properties** 有下列選項：

- **Replace incoming timestamp...** 該選項讓您在時間戳記裡使用本機時間。如果資料來源無法建立時間戳記，或是您不信任資料來源的時間，那這個選項就非常有用。
- **Transmit point changes in binary** 該選項提供使用者x86 CPUs，這是能加快資料傳輸率的方法。

法。選取該選項可以提升最大傳輸量至高達50%。



欲了解更多資訊，請參閱**章節 17.1**，“二進位模式的Tunnel/Mirror (TCP) 連接”。

- **Target is a Cogent Embedded Toolkit server** 該選項讓此slave連接到內嵌工具組server，而不是連接到另一DataHub。
- **Heartbeat** 該選項使用在此指定的每個milliseconds秒數來發送一個活動訊號訊息到master，藉此驗證已連線。
- **Timeout** 該選項指定活動訊息的逾時期間。如果slave DataHub在逾時之內沒有從master接收到回應，它會中斷該連線。您必須把逾時時間設定為至少是活動訊息時間的兩倍。



欲將此減緩網路的設定最佳化，請參閱**章節 17.2**，“慢速網路的Tunnel/Mirror (TCP)連接”。

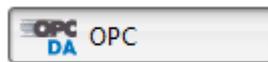
- **Retry** 在嘗試重新連接到已中斷的連線之前，指定一個等候的milliseconds秒數量。
7. 點擊**OK**以關閉**Tunnel/Mirror Master**視窗。屬性視窗裡**Tunnelling Slave**表格的欄位應該已填滿。
 8. 點擊屬性視窗中的**Apply**按鈕。如果正在執行master DataHub，該DataHub應建立tunnelling連線，並且**Status**應會顯示已連線。您也可以使用**Data Browser**來檢視資料，或是用**Connection Viewer**。

設定Cogent DataHub以執行成為OPC client的server

最後，我們建議您確認OPC client機器裡的Cogent DataHub是設定執行成為OPC server。每個Cogent DataHub的預先設定都是如此，但檢查一下也無妨。

1. 在DataHub系統匣圖示點擊右鍵，接著選取**屬性**。

2. 在屬性視窗中，選取**OPC**。



3. 確保已勾選**Act as an OPC Server**方框。



如果您的OPC client要求您手工輸入OPC server名稱，請使用Cogent.OPCDataHub或Cogent.OPCDataHub.1。

Do not adjust OPC registry entries for this program選項告知Cogent DataHub不要改變其登錄設定。當您想要用redundancy server或是某些獨立修改DataHub登錄的其他程式來使用Cogent DataHub時，這個選項就非常有用。若沒有勾選該方框，當它啟動或是當**Act as an OPC Server**變更套用時，DataHub會覆寫任何外部變更。兩個方框共同運作，因為啟動或關閉OPC server行為必定會造成登錄的變更。以下是當您也需要維護登錄設定時，如何變更OPC server行為的方法：

- a. 取消勾選**Do not adjust OPC registry entries for this program**。這會讓**Act as an OPC Server**核取方框變成可選取。
 - b. 依照需求勾選或取消勾選 **Act as an OPC Server**，接著點擊**Apply**。
 - c. 勾選**Do not adjust OPC registry entries for t his program**，接著點擊**Apply**。
4. 點擊屬性視窗下方的**Apply**按鈕以套用變更。您可以使用**Connection Viewer**來檢視連線。

現在可以啟動您的OPC client，連接到Cogent DataHub，並存取您的資料。

版權所有 © 1995-2011 by Cogent Real-Time Systems, Inc.

2.4. 測試連線

您可以用下列方法來測試您的tunnelling:

1. 確保您已正確建立**OPC server機器**和**OPC client機器**。
2. 尚未執行，請在**OPC server機器**上啟動Cogent DataHub。它應該在**OPC server**所在的機器上執行。
3. 在**OPC client機器**裡啟動**OPC client**。它該啟動Cogent DataHub，且只要連線被建立，來自**OPC server**的資料也可在**OPC client**中被看見。
4. 您可以使用**Connection Viewer**來檢視連線。

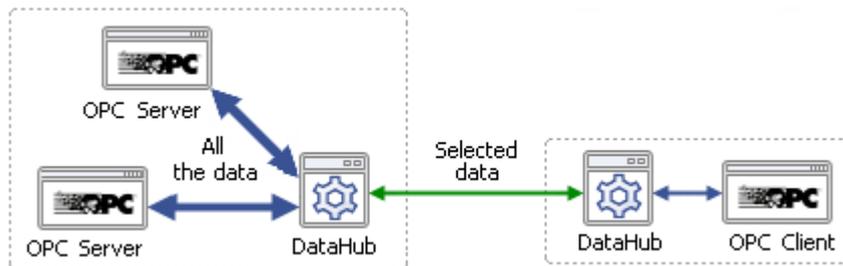
如果您在**OPC client**沒有看到資料，請再次檢查以下內容:

- 在兩個機器上都有安裝Cogent DataHub。
- **OPC server**和**client**的函式。
- 實體網路連線。

版權所有 © 1995-2011 by Cogent Real-Time Systems, Inc.

2.5. Tunnelling部份資料集

通常，您不需要橫跨網路來tunnel來自OPC server的所有資料。它只傳送您所需要的資料，所以能更快，且佔更少的頻寬。Cogent DataHub讓您藉由替已tunnel的資料設定一個單獨的Data Domain，以做上述動作。事實上，您可以從多個servers把部份資料集彙總到單一Data Domain，然後進行tunnel可橫跨網路結合資料集。



把資料從一個或多個servers放進一個單獨的Data Domain，是透過OPC橋接完成的。當您設定橋接時，只需確保在一個新的、獨立的Data Domain中建立目標點。欲了解OPC橋接的更多資訊，請參閱[章節 4, OPC橋接](#)。

2.6. 擴充應用

只使用Tunnelling就能大大地提高OPC的實用性。但是，您可以藉由把tunnelling功能以及OPC橋接功能和/或彙總功能結合在一起，以便從Cogent DataHub得到更多的益處。

2.6.1. Tunnelling和Bridging

OPC橋接是指從一OPC server連結資料到另一個OPC server，通常是在單一機器上。雖然如此，您也可以使用tunnelling連接來透過網路橋接兩個OPC server。

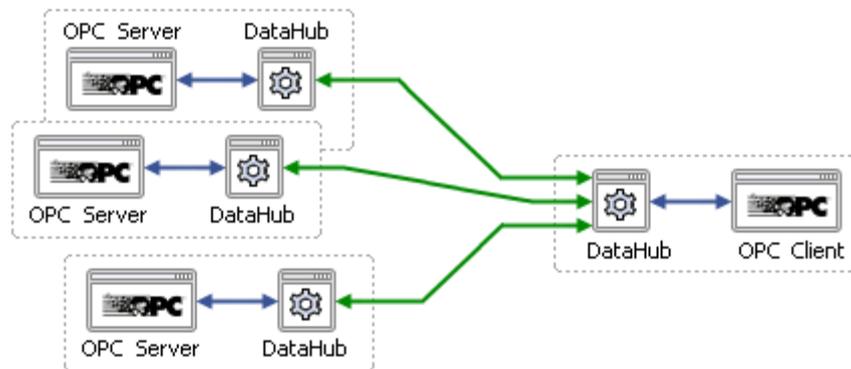


該狀況包含在兩台機器裡建立DataHubs以便為個別的OPC servers執行成為**OPC clients**。然後，DataHubs介面連接其他每個透過TCP tunnelling的連線。在機器上設定DataHub，執行期間最長的為**tunnelling master**，其他DataHub為**tunnelling slave**。我們建議只在其中一個DataHubs裡設定所有橋接。

2.6.2. Tunnelling和Aggregation

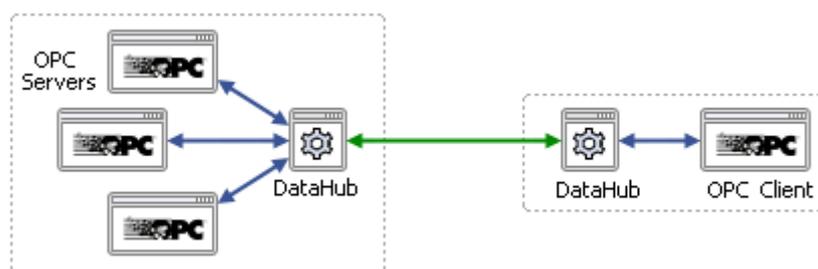
OPC彙總表示結合來自多個OPC servers的資料到一個server。這裡有幾個讓tunnelling可以和彙總結合的方式：

1. **從遠端servers彙總** 使用tunnelling來從不同機器中的多個OPC servers取得資料並提供給一client。



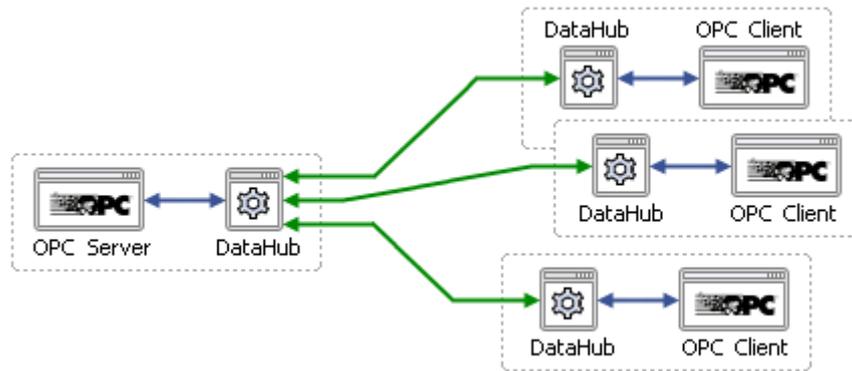
該狀況包含建立三個**OPC server**機器和一個**OPC client**機器以便進行tunnelling。**彙總**在OPC server機器中進行。

2. **遠端連接到多個servers** 進行tunnelling來把一機器中的servers組連接到一個遠端client。



該狀況需要各為這三個不同的OPC servers設定DataHub，並建立**OPC client**機器以進行tunnelling。**彙總**在OPC server機器中進行。

遠端連接許多clients 使用tunnelling來從一server取得資料並提供給許多遠端clients。



該狀況包括建立一個**OPC server機器**和三個**OPC client機器**以進行tunnelling。彙總在OPC server機器中進行。

4. 還有許多其他組合，以上只列出幾個以方便您開始。

欲了解更多有關使用Cogent DataHub來彙總clients或servers的資訊，請參閱**章節 3，OPC彙總**。