### 研杰科技 YOUNGTEC

# Redundancy(复联或二重化机制)



Redundancy 選項 讓您設定 redundant 連接到 Cogent DataHub。

Redundancy 的目的是從兩個数据來源收集数据,並呈現給 client 程式一個單一的輸出数据 集。Cogent DataHub 會判定將呈現給 client 程式哪個資源,並在兩個來源中作切換,而不会影響 client。Client 只會從輸出的数据集讀取数据。

兩個輸入和一個輸出的数据集都是維持在 Cogent DataHub 的獨立数据领域。來源不需要相同的 通訊協定,因此 Redundancy 可以套用到兩种來源,例如其中一個是直接的 OPC 連線,而另一個 是 Tunne1。

Enal	ble redu	ndancy			
edur	ndancy S	Sets			
Dn	Label	Source 1	Source 2	Output	Add
					Edit

點擊以勾取 Enable redundancy 方框来啟用這個功能。Redundant 連線是以数据集的方式建立 和儲存。您可以建立多個 redundant 数据集,使用清單中其對應的 On 核取方框來啟動或停用各 個数据集。欲編輯 redundancy 数据集,在其上連點兩下,或是選取它並按下 Edit 按鈕以開啟 Configure Redundancy 視窗(見下圖)。欲移除一個数据集,請反白它,接著點擊 Remove 按鈕。

欲建立一個 redundancy 数据集, 按下 Add 按鈕, 這會開啟 Configure Redundancy 視窗:

Data Domains	
Label:	
Source Domain 1:	<ul> <li>Preferred source</li> </ul>
Source Domain 2:	<ul> <li>Preferred source</li> </ul>
Output Domain:	



### **Data Domains**

### Label:

Cogent DataHub 用來識別 redundancy 数据集的名稱。Label 文字中不能有空格。選取哪個 Label 不重要,但是這個 Label 必须與其他 Label 不同。

### Source Domain 1:

第一個数据來源的 DataHub 数据领域。如果這是偏好的來源,請勾選 Preferred source 方框。

如果偏好的数据來源已經指定,那麼 DataHub 會盡量使用該來 源,即使也可以使用其他來源。如果兩個数据來源擁有不同的 特性,這就非常有用。例如,偏好的來源可能會提供比其他來 源更高的頻寬。如果沒有選取任何数据來源作為偏好的來源, DataHub 會維護每個正在使用的数据來源,直到它符合失效的 標準(見下文)。

### Source Domain 2:

第二個数据來源的 DataHub 数据领域。如果這是偏好的來源,請勾選 Preferred source 方框。

### **Output Domain:**

DataHub 数据领域的名稱,這會是redundant 連線的輸出,是讓 client 來連接的。如果 输出领域不存在, DataHub 會建立它。



## Input Domain is Invalid When

這個选择的输入決定何時 DataHub 該從一個 redundant 数据來源切換到另一個。

Input Domain is Invalid Wł	hen			
Oata quality is:	is equal to	•	Bad	•
O Data value is:	is equal to	-		
For point(s)				
For any point in the	domain			
For this point				
Treat the point r	ame as a pattern			

### Data quality is:

根據您所選取點的数据品質變更(如下)來讓您切換数据來源的選項。您可以从可用品質 的清單上設定等於或是不等於的條件,如:

Bad	EGU Exceeded	Last Usable	Sensor Calibration
Comm Failure	Good	Local Override	Sensor Failure
Config Error	Initializing	Not Connected	Sub Normal
Device Failure	Last Known	Out of Service	Uncertain

### Data value is:

根據您所選取之数据點的值變更(如下)以給您切換数据來源的選項。

### For point(s)

讓您選取要監看哪些點的品質或是数值(如上)。

#### For any point in the domain

讓您監視领域裡所有的點,並在任何點符合標準時作切換。

### For this point

讓您指定一個個別點的名稱。點的名稱可以套用到單一的點,或是套用到其名稱符合样 式的點群組。



## **Data Flow Detection**

對於規律變化的数据,這個選項在偵測到数据流有過大的停止間隔時,可以讓您切換数据來 源。此选项监看输入领域内任意数据点的任何变化。如果输入领域内任意数据点在指定的时间 内变化,这整个输入领域则被假设为可用。数据变更也必须通过以上有效性检查,才可被断定 为有效。

ta Flow Detection		
Switch sources if data stops for:	0.000	seconds

### Switch sources if data stops for:

在 DataHub 切換到 redundant 数据领域之前,來自此领域的数据流必須停顿的秒數。

### **Options**

Opt	tions
	Never switch. Always use the first source.
	Do not refresh the output domain when switching
	Never copy the data model to the output domain.
	Always copy the data model when switching.
	Ignore values written to the output domain

#### Never switch. Always use the first source.

如果您從不切換,就不會使用到第二個数据來源。因此,這可會讓您有效地建立领域的 複本,藉由將 Source Domain 1 複製到 Output Domain。欲將输出领域作為來源领域的 唯讀複本,請選取 Ignore values written to the output domain. 選項,解释如下。

### Do not refresh the output domain when switching.

在切換時,將現有的数据保存在输出领域,且當發生變更時,只更新來自新领域的值。 選取此選項可能會造成输入和输出领域之間不定時間長度的数据不相符。

通常切換的同時,DataHub 會將新來源领域的值複製到输出领域。這種複製動作可能會造成極大量點數更新输出领域的延遲。如果您的情況是這樣,且您知道您的兩個 server 是以有较长时间的方式同步,您不妨選取這個選項。然而,您需要記得,输入和输出领域的值在不確定的時間區間可能不相符。

### Never copy the data model to the output domain.

保留输出领域的数据模型,若沒有数据模型,從输入领域來平贴数据模型。在這兩種情況下,数据點名稱可被維持。如果目標系統的系統資源有限,如嵌入式系統,或是如果您在输出领域裡有現存的数据模型,並且不希望它被输入领域裡的数据模型覆寫,這個選項可能會有幫助。

第4頁

### 研杰科技 YOUNGTEC

### Always copy the data model when switching.

通常來說,如果数据模型變更以及被覆寫時,输出领域會在数据模型變更以及在發生切換時追蹤變更。這個選項會強制输出领域複製数据模型,無論它是否有變更。

### Ignore values written to the output domain.

寫入输出领域的数据通常會傳播回输入领域。這個選項預防這種情況發生。寫入到输出 领域的数据將不會寫到输入领域。與 Never switch. Always use the first source. (如 上)一起使用,這會讓输出领域作為输入领域的唯讀複本。

# Status and Control Data Points (空白表示停用)

### Point for current source number:

將指出正在使用哪個來源的 DataHub 點名稱。

### Point for current state of domain 1:

將指出 Domain 1 狀態的 DataHub 點名稱。

### Point for current state of domain 2:

將指出 Domain 2 狀態的 DataHub 點名稱。

### Point for preferred source number:

將指出哪個数据來源是您偏好來源的 DataHub 點名稱。

點擊 OK 按鈕以提交您的输入。



DataHub 的 Redundancy 功能是可以使用 DataHub script 来支持热待命备援的。一旦我们在数据 层级而不是联机层级施行 Redundancy, DataHub 不会知道当它切换输入领域时,哪一个连接将 被启动和停止。因此您需要一个能监看 Redundancy 狀態,並基于连接如何与数据领域匹配的分 离知识来開啟或關閉连接的 script。

下面是一個範例 script,顯示了如何設置 redundancy 熱備援。它需要您先通過此界面配置 redundancy,然後再次重新輸入一些相同的訊息。例如,這裡是一個 OPC 組態,顯示二个命名 為 OPC000 和 OPC002 的兩個連接喂数据进入领域 input 1 和 input 2:

1. 配置 OPC servers。

OPC	Client					
V	Act as an OPC	Client to these s	ervers:			
00	Status	Computer	Connection	OPC Server	Domain	Pefrech(m
On	Status	Computer	Connection	OPC Server	Domain	Refresh(m
On V	Status Running	Computer localhost	Connection OPC000	OPC Server Softing OPC Toolb	Domain input1	Refresh(m 0

2. 配置 Redundancy 对的卷标,例如 RED001。

ata Domains .abel:	RED001	
Source Domain 1:	input1	▼ V Preferred source
Source Domain 2:	input2	▼ Preferred source
Output Domain:	output	



3. 這 Redundancy 對必須包括四個狀態點配置。您可以依您喜好為它們命名::

Status and Control Data Points (blank f	for disabled)
Point for current source number:	CurrentSource
Point for current state of domain 1:	State 1
Point for current state of domain 2:	State2
Point for preferred source number:	Preference

4. 这个 script 需要去编辑修改,使包含关于 redundancy 对的规划信息,包括 Label、Source Domain 1、Source Domain 2、Output Domain 名称以及所有四個 Status and Control Data Point 名称。

```
8
   method WarmStandby.constructor ()
9
   {
        local redconf = new DomainBridge();
10
       redconf.SetApplication(self);
11
12
        redconf.Attach("RED001", "input1", "input2", "output", 0, 0, nil, 0, nil,
            "CurrentSource", "State1", "State2", "Preference");
13
       redconf.OPCWarmStandby("OPC000", "OPC002");
14
15
   }
10
```

一旦我們附加到 redundancy 對,我們可以告訴 script 哪一个 OPC 連接對應到輸入领域。 這些是由 OPC 連接标签對應到每個輸入领域所提供。

當您執行此 script, 它将监看狀態點, 並根據数据的狀態决定哪个 OPC 連接應該动作, 哪个應該不动作。

### WarmStandby.g 范例 script

redconf.Attach("RED001", "input1", "input2", "output", 0, 0, nil, 0, nil,

第7頁



"CurrentSource", "State1", "State2", "Preference"); redconf.OPCWarmStandby("OPC000", "OPC002");

}

ApplicationSingleton (WarmStandby);