



# WFA的編碼準則

## OnCommand Workflow Automation

NetApp  
October 09, 2025

# 目錄

WFA的編碼準則	1
變數準則	1
PowerShell變數	1
Perl變數	3
縮排準則	4
意見準則	5
PowerShell註解	5
Perl評論	6
記錄準則	7
PowerShell記錄	7
Perl記錄	8
錯誤處理準則	9
PowerShell錯誤處理	9
Perl錯誤處理	11
WFA的一般PowerShell與Perl慣例	12
Windows隨附的Perl模組	13
新增自訂PowerShell和Perl模組的考量	13
WFA Cmdlet與功能	13
PowerShell與Perl WFA模組	14
PowerShell模組	14
Perl模組	14
將PowerShell命令轉換為Perl時的考量	17
命令輸入類型	17
PowerShell聲明	17
Perl聲明	17
命令定義	19
WFA建置區塊準則	19
WFA中的SQL準則	20
WFA功能準則	22
WFA字典項目準則	22
命令準則	23
工作流程準則	25
建立遠端系統類型驗證指令碼的準則	29
建立資料來源類型的準則	30

# WFA的編碼準則

您應該瞭解OnCommand Workflow Automation 有關建立各種建置區塊（例如篩選器、功能、命令和工作流程）的一般資訊（WFA）編碼準則、命名慣例和建議。

## 變數準則

建立命令或資料來源類型時、您必須瞭解OnCommand Workflow Automation 到有關PowerShell和Perl變數的準則（WFA）。

### PowerShell變數

準則	範例
對於指令碼輸入參數： <ul style="list-style-type: none"><li>使用Pascal案例。</li><li>請勿使用底線。</li><li>請勿使用縮寫。</li></ul>	\$' Volume名稱' 「\$AutoDelete選項」 \$'大小'
對於指令碼內部變數： <ul style="list-style-type: none"><li>使用Camel Case。</li><li>請勿使用底線。</li><li>請勿使用縮寫。</li></ul>	「\$newVolume」 \$qtreeName' \$time
對於功能： <ul style="list-style-type: none"><li>使用Pascal案例。</li><li>請勿使用底線。</li><li>請勿使用縮寫。</li></ul>	「GetVolume大小」
變數名稱不區分大小寫。不過、為了提高讀取能力、您不應該對同一個名稱使用不同的大寫。	「\$variable」與「\$variable」相同
變數名稱應為純英文、並應與指令碼的功能有關。	使用「\$name」而非「\$a」
明確宣告每個變數的資料類型。	[str]名稱 [Int]大小
請勿使用特殊字元（！@#&%、.）和空格。	無

準則	範例
請勿使用PowerShell保留關鍵字。	無
先放置必要參數、然後再放置選用參數、即可將輸入參數分組。	<pre> param( [parameter(Mandatory=\$true)] [string]\$Type, [parameter(Mandatory=\$true)] [string]\$Ip, [parameter(Mandatory=\$false)] [string]\$VolumeName ) </pre>
使用「 <i>HelpMess_</i> 」附註加上有意義的說明訊息、來註解所有輸入變數。	<pre> [parameter(Mandatory=\$false,HelpMe ssage="LUN to map")] [string]\$LUNName </pre>
請勿使用「'Filer'」做為變數名稱；請改用「'Array」。	無
如果參數獲得列舉值、請使用「 <i>ValidateSet</i> 」註解。這會自動轉譯為參數的Enum資料類型。	<pre> [parameter(Mandatory=\$false,HelpMe ssage="Volume state")] [ValidateSet("online","offline","r estricted")] [string]\$State </pre>
將別名新增至以「`_Capacity」結尾的參數、以表示參數為容量類型。	<p>「Create Volume」命令使用別名如下：</p> <pre> [parameter(Mandatory=\$false,HelpMe ssage="Volume increment size in MB")] [Alias("AutosizeIncrementSize_Capa city")] [int]\$AutosizeIncrementSize </pre>

準則	範例
在以「_Password」結尾的參數中新增別名、表示該參數為密碼類型。	<pre>param (     [parameter(Mandatory=\$false,     HelpMessage="In order to create an     Active Directory machine account     for the CIFS server or setup CIFS     service for Storage Virtual     Machine, you must supply the     password of a Windows account with     sufficient privileges")]     [Alias("Pwd_Password")]     [string]\$ADAdminPassword )</pre>

## Perl變數

準則	範例
對於指令碼輸入參數：	<p>\$' Volume名稱'</p> <p>「\$AutoDelete選項」</p> <p>\$'大小'</p>
請勿使用指令碼內部變數的縮寫。	<p>\$_new_volume</p> <p>\$qtree名稱</p> <p>\$time</p>
請勿將縮寫用於功能。	"Get_Volume大小"
變數名稱區分大小寫。為了提高讀取能力、您不應使用相同名稱的不同大寫字母。	「\$variable」與「\$variable」不同
變數名稱應為純英文、並應與指令碼的功能有關。	使用「\$name」而非「\$a」
先放置必要參數、然後再放置選用參數、即可將輸入參數分組。	無

準則	範例
在GetOptions函數中、明確宣告輸入參數的每個變數資料類型。	<pre data-bbox="850 192 1220 346">GetOptions (     "Name=s"=&gt;\\$Name,     "Size=i"=&gt;\\$Size )</pre>
請勿使用「'Filer」做為變數名稱；請改用「'Array」。	無
Perl不包含列舉值的「ValidateSet」註釋。對於參數獲得列舉值的情況、請使用明確的「'if」陳述式。	<pre data-bbox="850 551 1454 663">if   (defined\$SpaceGuarantee&amp;&amp; ! (\$SpaceGuarantee eq 'none')) {</pre>
	<pre data-bbox="817 751 1184 783">\$SpaceGuarantee eq 'volume'</pre>
所有Perl WFA命令都必須使用「嚴格」Pragma,以防止對變數、參考和子程序使用不安全的結構。	<pre data-bbox="817 804 1496 903">\$SpaceGuarantee eq 'file')) { die'Illegal SpaceGuarantee argument: \\".\$SpaceGuarantee.\\"; }</pre> <p data-bbox="817 882 861 903">----</p> <pre data-bbox="850 956 1348 1146">use strict; # the above is equivalent to use strictvars; use strictsubs; use strictrefs;</pre>
<p>所有Perl WFA命令都必須使用下列Perl模組：</p> <ul style="list-style-type: none"> <li>• getopt</li> </ul> <p>這用於指定輸入參數。</p> <ul style="list-style-type: none"> <li>• WFAUtil</li> </ul> <p>此功能用於提供命令記錄、報告命令進度、連線至陣列控制器等公用程式功能。</p>	<pre data-bbox="850 1262 1148 1374">use Getopt::Long; use NaServer; use WFAUtil;</pre>

## 縮排準則

撰寫OnCommand Workflow Automation 適用於還原（WFA）的PowerShell或Perl指令碼時、您必須瞭解縮排準則。

準則	範例
索引標籤等於四個空格。	
使用定位點和大括弧來顯示區塊的開頭和結尾。	<p>PowerShell指令碼</p> <pre data-bbox="850 354 1351 536"> if (\$pair.length-ne 2) { throw "Got wrong input data" } </pre>
在作業集或程式碼區塊之間新增空白行。	<p>Perl指令碼</p> <pre data-bbox="850 713 1380 1022"> if defined \$MaxDirectorySize { # convert from MBytes to Bytes my \$MaxDirectorySizeBytes = \$MaxDirectorySize * 1024 * 1024; } </pre> <p>PowerShell指令碼</p> <pre data-bbox="850 1142 1364 1332"> \$options=\$option.trim(); \$pair=\$option.split(" "); Get-WFALogger -Info -messages \$("split options: "+ \$Pair) </pre>

## 意見準則

您必須瞭解指令碼OnCommand Workflow Automation for the指令碼（WFA）中的PowerShell和Perl註解準則。

### PowerShell註解

準則	範例
請使用#字元做為單行註解。	<pre># Single line comment \$options=\$option.trim();</pre>
使用#字元做為行尾註解。	<pre>\$options=\$option.trim(); # End of line comment</pre>
請使用<#和#>字元做為區塊註解。	<pre>&lt;# This is a block comment #&gt; \$options=\$option.trim();</pre>

## Perl評論

準則	範例
使用#字元進行單行註解。	<pre># convert from MBytes to Bytes my \$MaxDirectorySizeBytes = \$MaxDirectorySize * 1024 * 1024;</pre>
使用#字元做為行尾註解。	<pre>my \$MaxDirectorySizeBytes = \$MaxDirect orySize * 1024 * 1024; # convert to Bytes</pre>

準則	範例
<p>在每一行中使用#字元、開頭和結尾處使用空白#、以建立多行註解的註解框線。</p>	<pre># This is a multi-line comment. Perl 5, unlike # Powershell, does not have direct support for # multi-line comments. Please use a '#' in every line # with an empty '#' at the beginning and end to create # a comment border #</pre>
<p>請勿在WFA命令中加入已註解和已失效的程式碼。不過、為了測試目的、您可以使用純舊文件 (pod) 機制來註解程式碼。</p>	<pre>=begin comment     # Set deduplication     if(defined \$Deduplication &amp;&amp; \$Deduplication eq "enabled")     {         \$wfaUtil- &gt;sendLog("Enabling Deduplication");     } =end comment =cut</pre>

## 記錄準則

撰寫OnCommand Workflow Automation 適用於還原 (WFA) 的PowerShell或Perl指令碼時、您必須注意記錄的準則。

### PowerShell記錄

準則	範例
<p>使用Get-WFALogger Cmdlet進行記錄。</p>	<pre>Get-WFALogger -Info -message "Creating volume"</pre>

準則	範例
記錄每個需要與Data ONTAP 內部套件（例如、VMware和PowerCLI）互動的動作。所有記錄訊息都可在工作流程執行狀態歷程記錄的「執行記錄」中找到。	無
記錄傳遞給內部套件的每個相關引數。	無

## Perl記錄

準則	範例
使用WFAUtil sendlog進行記錄。	<pre data-bbox="850 825 1396 1058"> my wfa_util = WFAUtil-&gt;new(); eval {     \$wfa_util-&gt;sendLog('INFO',         "Connecting to the         cluster: \$DestinationCluster"); } </pre>
記錄所有需要與命令以外的任何內容互動的動作、例如Data ONTAP：VMware和WFA。您使用WFAUtil sendLog例程建立的所有記錄訊息、都會儲存在WFA 資料庫中。這些記錄訊息可用於執行的工作流程和命令。	無
記錄傳遞給所呼叫例行工作的每個相關引數。	無
使用適當的記錄層級。資訊、錯誤、警告和-Debug是各種可用的記錄層級。	無

準則	範例
<p>在資訊層級記錄時、請保持精確簡潔。請勿在記錄訊息中指定實作詳細資料、例如類別名稱和功能名稱。請以純英文說明確切步驟或確切錯誤。</p>	<p>下列程式碼片段顯示良好訊息和不良訊息的範例：</p> <pre>\$wfa_util-&gt;sendLog('WARN', "Removing volume: '.'.\$VolumeName); # Good Message</pre> <pre>\$wfa_util-&gt;sendLog('WARN', 'Invoking volume- destroy ZAPI: '.\$VolumeName); # Bad message</pre>

## 錯誤處理準則

撰寫OnCommand Workflow Automation 適用於WFA（WFA）的PowerShell或Perl指令碼時、您必須瞭解錯誤處理的準則。

### PowerShell錯誤處理

準則	範例
<p>PowerShell執行時間新增至Cmdlet的一般參數包括錯誤處理參數、例如錯誤動作和警告動作：</p> <ul style="list-style-type: none"> <li>• 錯誤動作參數決定Cmdlet應如何回應命令中的非終止錯誤。</li> <li>• WarningAction參數決定Cmdlet應如何回應命令的警告。</li> <li>• Stop、SilentlyContinue、Inquire和Continue是錯誤行動和警告行動參數的有效值。</li> </ul> <p>如需詳細資訊、您可以在PowerShell CLI中使用「Get-Help關於_CommonParameters」命令。</p>	<p>錯誤動作：下列範例顯示如何將非終止錯誤當作終止錯誤處理：</p> <pre>New-NcIgroup-Name \$IgroupName- Protocol \$Protocol-Type\$OSType- ErrorActionstop</pre> <p>WarningAction</p> <pre>New-VM-Name \$VMName-VM \$SourceVM- DataStore\$DataStoreName- VMHost\$VMHost- WarningActionSilentlyContinue</pre>

準則	範例
<p>如果傳入例外的類型不明、請使用一般的「try/catch」陳述式。</p>	<pre>try {     "In Try/catch block" } catch {     "Got exception" }</pre>
<p>如果已知傳入例外的類型、請使用特定的「try/catch」陳述式。</p>	<pre>try {     "In Try/catch block" } catch[System.Net.WebException], [System.IO. IOException] {     "Got exception" }</pre>
<p>使用「finally」聲明來釋出資源。</p>	<pre>try {     "In Try/catch block" } catch {     "Got exception" } finally {     "Release resources" }</pre>

準則	範例
使用PowerShell自動變數來存取例外狀況的相關資訊。	<pre>try {     Get-WFALogger -Info -message     \$("Creating     Ipspace: " + \$Ipspace)     New-NaNetIpspace-Name \$Ipspace } catch {     Throw "Failed to create Ipspace. Message: " + \$_.Exception.Message; }</pre>

## Perl錯誤處理

準則	範例
<p>Perl不包含試用/攔截區塊的原生語言支援。使用評估區塊來檢查及處理錯誤。盡量將評估區塊保持在最小的位置。</p>	<pre data-bbox="840 200 1460 1256"> eval {     \$wfa_util-&gt;sendLog('INFO',         "Quiescing the relationship :         \$DestinationCluster://\$Destination         Vserver         /\$DestinationVolume"     );     \$server-&gt;snapmirror_quiesce(         'destination-vserver' =&gt;         \$DestinationVserver,         'destination-volume'  =&gt;         \$DestinationVolume     );     \$wfa_util-&gt;sendLog('INFO',         'Quiesce operation         started successfully.'); };  \$wfa_util-&gt;checkEvalFailure(     "Failed to quiesce the SnapMirror     relationship     \$DestinationCluster://\$Destination     Vserver     /\$DestinationVolume",     \$@ ); </pre>

## WFA的一般PowerShell與Perl慣例

您必須瞭解WFA中使用的特定PowerShell和Perl慣例、才能建立與現有指令碼一致的指令碼。

- 使用有助於釐清指令碼內容的變數。
- 撰寫可讀取的程式碼、無需註解即可理解。
- 尽量簡化指令碼和命令。
- PowerShell指令碼：
  - 盡可能使用Cmdlet。
  - 在沒有可用的Cmdlet時叫用.NET程式碼。
- Perl指令碼：

- 請務必以換行字元結束「die」陳述。

如果沒有換行字元、則會列印指令碼行編號、這對於偵錯WFA執行的Perl命令並沒有用。

- 在「'getopt'」模組中、將命令的字串引數設定為必填。

## Windows隨附的Perl模組

有些Perl模組與Windows Active狀態Perl發佈for OnCommand Workflow Automation the WFA (WFA) 搭售。您可以在Perl程式碼中使用這些Perl模組來撰寫命令、只有在這些模組與Windows搭售時才適用。

下表列出Windows for WFA隨附的Perl資料庫模組。

資料庫模組	說明
DBD : mysql	可讓您連線至MySQL資料庫的Perl5資料庫介面驅動程式。
試用：極小	利用評估區塊將常見錯誤減至最低。
XML : : libxml	與提供XML與HTML剖析器的libxml2介面、其中包含了DOM、SAX及XmlReader介面。
DBD : Cassandra	適用於使用QL3查詢語言之Cassandra的Perl5資料庫介面驅動程式。

## 新增自訂PowerShell和Perl模組的考量

在OnCommand Workflow Automation 將自訂PowerShell和Perl模組新增至還原 (WFA) 之前、您必須注意某些考量事項。自訂PowerShell和Perl模組可讓您使用自訂命令來建立工作流程。

- 在執行WFA命令期間、會自動將所有自訂PowerShell模組新增至WFA安裝目錄「/奢侈/模組」。
- 所有新增至「WFA/perl」目錄的自訂Perl模組、都會包含在 @\_Inc\_ 程式庫中。
- 自訂PowerShell和Perl模組不會在WFA備份作業中備份。
- 自訂PowerShell和Perl模組不會在WFA還原作業中還原。

您必須手動備份自訂PowerShell和Perl模組、才能將它們複製到新的WFA安裝。

模組目錄中的資料夾名稱必須與模組名稱相同。

## WFA Cmdlet與功能

支援各種PowerShell指令程式、以及可在WFA命令中使用的PowerShell和Perl功

## 能。OnCommand Workflow Automation

您可以使用下列PowerShell命令來檢視WFA伺服器所提供的所有PowerShell Cmdlet和功能：

- 「Get-Command -模組WFAWrapper」
- 「Get-Command - Module WFA」

您可以在「WFAUtil.pPM」模組中檢視WFA伺服器提供的所有Perl功能。WFA說明模組支援連結的說明區段、WFA PowerShell Cmdlet說明及WFA Perl方法說明、可讓您存取所有PowerShell Cmdlet和功能、以及Perl功能。

## PowerShell與Perl WFA模組

您必須知道適用於OnCommand Workflow Automation 還原（WFA）的PowerShell或Perl模組、才能為工作流程撰寫指令碼。

### PowerShell模組

準則	範例
只要工具套件可供使用、就使用Data ONTAP 「支援服務工具套件」來叫用API。	「Add VLAN」命令會使用以下工具組： 「Add-NaNetVlan-Interface \$Interface-vlan\$VlanID」
如果Data ONTAP 無法在《Sfuse Toolkit》中找到Cmdlet、請使用「Invoke -SSH」命令、在Data ONTAP Sfuse上叫用CLI。	「Invoke-NaSsh-Name \$ArrayName-Command "ifconfig-a"" -認證\$Credentials」

### Perl模組

Na伺服器模組用於WFA命令。NavServer模組可叫用Data ONTAP 用於動態管理Data ONTAP 的功能完善的資訊技術API。



準則	範例
<p>只要NetApp Manageability SDK可供使用、即可使用NavServer模組來叫用API。</p>	<p>以下範例顯示如何使用Na伺服器模組來執行恢復SnapMirror作業：</p> <pre data-bbox="816 264 1498 2097"> eval {      \$wfa_util-&gt;sendLog('INFO',         "Connecting to the cluster: \$DestinationCluster"     );     my \$server         = \$wfa_util- &gt;connect(\$DestinationClusterIp, \$DestinationVserver);      my \$sm_info = \$server- &gt;snapmirror_get(         'destination-vserver' =&gt; \$DestinationVserver,         'destination-volume' =&gt; \$DestinationVolume     );      my \$sm_state = \$sm_info- &gt;{ 'attributes' }-&gt;{ 'snapmirror- info' }-&gt;{ 'mirror-state' };     my \$sm_status = \$sm_info- &gt;{ 'attributes' }-&gt;{ 'snapmirror- info' }-&gt;{ 'relationship-status' };      \$wfa_util-&gt;sendLog('INFO',         "SnapMirror relationship is \$sm_state (\$sm_status)");      if (\$sm_status ne 'quiesced')     {         \$wfa_util-&gt;sendLog('INFO',             'The status needs to be quiesced to resume transfer.');     } else {         my \$result = \$server- &gt;snapmirror_resume(             'destination-vserver' =&gt; \$DestinationVserver,             'destination-volume' =&gt; \$DestinationVolume         );         \$wfa_util-&gt;sendLog('INFO',             "Result of resume: \$result");     } } </pre>

準則	範例
<p>如果Data ONTAP 無法使用某個API、請Data ONTAP 使用executeSystemCli公用程式方法來叫用該程式。</p> <p> 不支援執行系統Cli、目前僅適用於Data ONTAP 以7-Mode運作的執行。</p>	無

## 將PowerShell命令轉換為Perl時的考量

將PowerShell命令轉換為Perl時、您必須注意某些重要考量、因為PowerShell和Perl的功能不同。

### 命令輸入類型

藉由使用者可在定義命令時、將陣列和雜湊當作命令的輸入內容。OnCommand Workflow Automation使用Perl定義命令時、無法使用這些輸入類型。如果您想要Perl命令接受陣列和雜湊輸入、可以在設計工具中將輸入定義為字串。然後命令定義可以剖析輸入、並視需要傳遞以建立陣列或雜湊。輸入說明說明預期輸入的格式。

```
my @input_as_array = split(',',$InputString); #Parse the input string of
format val1,val2 into an array

my %input_as_hash = split /[:=]/, $InputString; #Parse the input string of
format key1=val1;key2=val2 into a hash.
```

### PowerShell聲明

下列範例顯示如何將陣列輸入內容傳遞至PowerShell和Perl。這些範例說明輸入CronMonth、指定cron工作排程執行的月份。有效值為1到11的整數。值-1表示排程每月執行一次。任何其他值均代表特定月份、0代表一月、11代表十二月。

```
[parameter(Mandatory=$false, HelpMessage="Months in which the schedule
executes. This is a comma separated list of values from 0 through 11.
Value -1 means all months.")]
[ValidateRange(-1, 11)]
[array]$CronMonths,
```

### Perl聲明

```

GetOptions(
    "Cluster=s"          => \$Cluster,
    "ScheduleName=s"     => \$ScheduleName,
    "Type=s"              => \$Type,
    "CronMonths=s"        => \$CronMonths,
) or die 'Illegal command parameters\n';

sub get_cron_months {
    return get_cron_input_hash('CronMonths', $CronMonths, 'cron-month',
-1,
    11);
}

sub get_cron_input_hash {
    my $input_name    = shift;
    my $input_value   = shift;
    my $zapi_element = shift;
    my $low           = shift;
    my $high          = shift;
    my $exclude       = shift;

    if (!defined $input_value) {
        return undef;
    }

    my @values = split(',', $input_value);

    foreach my $val (@values) {
        if ($val !~ /^[+-]?\d+$/) {
            die
                "Invalid value '$input_value' for $input_name: $val must
be an integer.\n";
        }
        if ($val < $low || $val > $high) {
            die
                "Invalid value '$input_value' for $input_name: $val must
be from $low to $high.\n";
        }
        if (defined $exclude && $val == $exclude) {
            die
                "Invalid value '$input_value' for $input_name: $val is not
valid.\n";
        }
    }
    # do something
}

```

## 命令定義

使用pipe運算子的PowerShell中的單行表示式、可能必須在Perl中擴充成多個陳述區塊、才能達到相同的功能。下表顯示其中一個wait命令的範例。

PowerShell聲明	Perl聲明
<pre># Get the latest job which moves the specified volume to the specified aggregate. \$job = Get-NcJob -Query \$query</pre>	<pre>where {\$_.JobDescription -eq "Split" + \$VolumeCloneName}</pre>
<pre>Select-Object -First 1 ----</pre>	<pre>my \$result = \$server-&gt;job_get_iter(     'query' =&gt; { 'job-type' =&gt;         'VOL_CLONE_SPLIT' },     'desired-attributes' =&gt; {         'job-type' =&gt; '',         'job-description' =&gt; '',         'job-progress' =&gt; '',         'job-state' =&gt; ''     } ); my @jobarray; for my \$job (@{ \$result-&gt;{ 'attributes-list' } }) {     my \$description = \$job-&gt;{ 'job-description' };     if(\$description =~         /\$VolumeCloneName/)     {         push(@jobarray, \$job)     } }</pre>

## WFA建置區塊準則

您必須瞭解使用Workflow Automation建置區塊的準則。

## WFA中的SQL準則

您必須瞭解使用SQL in OnCommand Workflow Automation the WFA (WFA) 來撰寫WFA的SQL查詢準則。

SQL用於WFA的下列位置：

- SQL查詢以填入使用者輸入以供選擇
- 用於建立篩選器以篩選特定字典項目類型的物件的SQL查詢
- Playground資料庫表格中的靜態資料
- 自訂資料來源類型的SQL、必須從外部資料來源擷取資料、例如自訂組態管理資料庫 (CMDB) 。
- SQL查詢保留與驗證指令碼

準則	範例
SQL保留關鍵字必須為大寫字元。	<pre>SELECT     vserver.name FROM     cm_storage.vserver vserver</pre>
表格和欄名稱必須為小寫字元。	表：Aggregate 欄：USed_SPATE_MB
以底線（_）字元分隔字詞。不允許使用空格。	Array效能
表格名稱是以單一名稱定義。表格是一個或多個項目的集合。	「'Functe」、而非「'Functions」
在選取查詢中使用具有有意義名稱的表格別名。	<pre>SELECT     vserver.name FROM     cm_storage.cluster cluster,     cm_storage.vserver vserver WHERE     vserver.cluster_id = cluster.id     AND cluster.name = '\$ClusterName'     AND vserver.type = 'cluster' ORDER BY     vserver.name ASC</pre>

準則	範例
<p>如果您必須在篩選查詢或使用者查詢中參照篩選器輸入參數或使用者輸入參數、請使用語法做為「\${intutVarableName}」。您也可以使用語法來參照保留指令碼和驗證指令碼中的命令定義參數。</p>	<pre> SELECT     volume.name AS Name,     aggregate.name as Aggregate,     volume.size_mb AS 'Total Size (MB)',     voulme.used_size_mb AS 'Used Size (MB)',     volume.space_guarantee AS 'Space Guarantee' FROM     cm_storage.cluster,     cm_storage.aggregate,     cm_storage.vserver,     cm_storage.volume WHERE     cluster.id = vserver.cluster_id     AND aggregate.id = volume.aggregate_id     AND vserver.id = voulme.vserver_id     AND vserver.name = '\${VserverName}'     AND cluster.name = '\${ClusterName}' ORDER BY     volume.name ASC </pre>
<p>針對複雜的查詢使用註解。查詢中支援的一些註解樣式如下：</p> <ul style="list-style-type: none"> <li>「--」直到行尾為止 此註解樣式中第二個連字號之後必須有空格。</li> <li>從="#"字元到行尾</li> <li>從「/」到下列「/」順序</li> </ul>	<pre> /* multi-line comment */ --line comment SELECT     ip as ip, # comment till end of this line     NAME as name FROM --end of line comment     storage.array </pre>

## WFA功能準則

您可以建立函數、將常用且更複雜的邏輯封裝在命名函數中、然後將此函數重複用於命令參數值或篩選OnCommand Workflow Automation 器參數值（WFA）。

準則	範例
請使用Camel Case作為函數名稱。	calulatedVolume大小
變數名稱應為純英文、並與函數的功能有關。	分段分隔符號
請勿使用縮寫。	calculatedVolume大小、_not_ cVolSiz
函數是使用MVFlex運算式語言（MVEL）定義。	無
功能定義應根據官方的Java程式設計語言準則加以指定。	無

## WFA字典項目準則

您必須瞭解OnCommand Workflow Automation 在還原（WFA）中建立字典項目的準則。

準則	範例
字典項目名稱只能包含英數字元和底線。	叢集授權 switch_23
字典項目名稱必須以大寫字元開頭。名稱中的每個字詞都以大寫字元開頭、並以底線（_）分隔每個字詞。	Volume Array授權
字典項目屬性名稱不應包含該字典項目的名稱。	無
在字典項目中的屬性和參照必須為小寫字元。	Aggregate、Size_MB
以底線分隔字詞。不允許使用空格。	resource_pool
字典項目不能包含來自不同配置的參考資料。當某個字典項目需要交叉參照到不同配置中的物件時、請確定所參照物件的所有自然金鑰都存在於該字典項目中。	Array_Performance字典項目要求陣列字典項目的所有自然金鑰都是其中的直接屬性。
為屬性使用適當的資料類型。	無

準則	範例
使用長資料類型作為大小或空間相關屬性。	Size_MB和可用的_Size_MB（儲存空間）。Volume（磁碟區）字典項目
當屬性具有固定值集時、請使用Enum。	在「儲存區」目錄項目中輸入RAID_type
當資料來源提供該屬性或參考的值時、請將屬性或參考的「to be Cached」設為true。若Active IQ Unified Manager 資料來源可以提供值、請新增可快取的屬性。	無
如果提供此屬性或參考值的資料來源可以傳回空值、請將「can be Null」設為真。	無
為每個屬性和參考提供有意義的說明。說明會在設計工作流程時顯示在命令詳細資料中。	無
請勿在字典項目中使用「id」做為屬性名稱。此屬性保留供內部WFA使用。	無

## 相關資訊

### 參考學習教材

## 命令準則

您必須瞭解OnCommand Workflow Automation 在還原（WFA）中建立命令的準則。

準則	範例
命令使用易於識別的名稱。	「Create Qtree」
使用空格分隔字詞、每個字詞必須以大寫字元開頭。	「Create Volume」（建立Volume）
提供說明來說明命令的功能、包括選用參數的預期結果。	無
根據預設、標準命令的逾時時間為600秒。建立命令時會設定預設的逾時時間。只有在命令可能需要較長時間才能完成時、才可變更預設值。	「Create Volume」命令
如果是長時間執行的作業、請建立兩個命令：一個叫用長時間執行的作業、另一個則定期報告作業進度。第一個命令應該是「標準執行」命令類型、第二個命令應該是「等待條件」命令類型。	「Create VSM（建立VSM）」和「Wait for VSM（等待VSM）」命令

準則	範例
在「Wait for Condition」命令名稱加上「wait」字首、以便輕鬆識別。	「等待CM Volume搬移」
請針對「wait for condition」命令、使用適當的等待時間間隔。指定的值會控制執行輪詢命令的時間間隔、以檢查長時間執行作業是否完成。	60秒的「等待VSM」命令取樣時間間隔
對於「等待條件」命令、請根據長時間執行作業的預期完成時間、使用適當的逾時時間。如果作業涉及透過網路傳輸資料、預期時間可能會長得多。	VSM基礎傳輸可能需要數天的時間才能完成。因此、指定的逾時時間為6天。

## 字串表示法

命令的字串表示法會在規劃和執行期間、在工作流程設計中顯示命令的詳細資料。命令的字串表示中只能使用命令參數。

準則	範例
避免使用沒有任何值的屬性。沒有值的屬性會顯示為NA。	VolName 10.68.66.212[NA] aggr1/testVol7.
使用下列分隔符號分隔字串表示中的不同項目：[]、/：	ArrayName[ArrayIp]
為字串表示的每個值提供有意義的標籤。	「Volume name=Volume Name_」

## 命令定義語言

可使用下列支援的指令碼語言來撰寫命令：

- PowerShell
- Perl

## 命令參數定義

命令參數會依名稱、說明、類型、參數的預設值、以及參數是否為必填。參數類型可以是「字串」、「布林」、「整數」、「長」、「雙」、列舉、日期時間、容量、陣列、橋式、密碼或mlDocument。雖然大多數類型的值都是直覺式的、但Array和Hashtable的值應採用下表所述的特定格式：

準則	範例
確定Array輸入類型的值是以逗號分隔的值清單。	<pre>[parameter (Mandatory=\$false, HelpMessage="Months in which the schedule executes.")] [array]\$CronMonths</pre> <p>輸入內容傳遞方式如下：0、3、6、9</p>
確認「橋式」輸入類型的值是以分號分隔的索引鍵=值配對清單。	<pre>[parameter (Mandatory=\$false, HelpMessage="Volume names and size (in MB)")] [hashtable]\$VolumeNamesAndSize</pre> <p>輸入會以下列方式傳送：Volume1=100 ; Volume2=250 ; Volume 3=50</p>

## 工作流程準則

您必須瞭解建立或修改預先定義OnCommand Workflow Automation 之for the WFA (WFA) 工作流程的準則。

### 一般準則

準則	範例
將工作流程命名為反映儲存操作員執行的作業。	「Create a CIFS Share」（建立CIFS共用）
對於工作流程名稱、請將第一個字詞的開頭字母和每個物件字詞大寫。縮寫和縮略字的英文字母大寫。	<p>Volume qtree</p> <p>建立叢集Data ONTAP 式的叢集式功能區樹狀CIFS共用</p>
如需工作流程說明、請納入工作流程的所有重要步驟、包括任何先決條件、工作流程結果或執行條件等。	請參閱Data ONTAP 範例工作流程「Create VMware NFS Datastore on the叢集式VMware vCenter Storage」（在叢集式VMware vCenter Storage上建立VMware NFS資料存放區）的說明、其中包含先決條件。
只有當工作流程已準備就緒可供正式作業時、才能將「Ready for Production」設為「true」、並可顯示在入口網站頁面中。	無

準則	範例
<p>根據預設、將「考慮保留元素」設為true。預覽工作流程以供執行時、WFA規劃人員會考量所有保留的物件、以及快取資料庫中的現有物件。如果此選項設為「true」、則在規劃特定工作流程時、會考慮其他排程工作流程或平行執行的工作流程的影響</p>	<ul style="list-style-type: none"> <li>案例1 工作流程1會建立Volume、排定在一週後執行。工作流程2會在要搜尋的磁碟區中建立qtree或LUN、如果工作流程2是在一天左右執行、您應該關閉工作流程2的「考慮保留元素」、以避免考慮在一週內建立的磁碟區。</li> <li>案例2 工作流程1使用「Create Volume」命令。如果排程工作流程2的Aggregate佔用100 GB空間、則工作流程1在規劃期間必須考量工作流程2的需求。</li> </ul>
<p>預設情況下、「Enable Element共存validation」（啟用元素存在驗證）設為「true」</p>	<ul style="list-style-type: none"> <li>案例1 如果您建立的工作流程只有在磁碟區存在時、才會使用「移除磁碟區」命令、依名稱移除磁碟區、然後使用「建立磁碟區」或「Clone Volume」等命令重新建立磁碟區、則工作流程不應使用此旗標。移除Volume的效果無法用於「Create volume」命令、因此工作流程會失敗。</li> <li>案例2 「Create Volume」命令用於特定名稱為「'vol198'」的工作流程。 如果此選項設為true、則WFA規劃人員會在規劃期間檢查指定陣列中是否存在名稱為該名稱的磁碟區。如果Volume存在、則工作流程在規劃期間會失敗。</li> </ul>
<p>在工作流程中選取相同的命令一次以上時、請為命令執行個體提供適當的顯示名稱。</p>	<p>「Create、map and Protect LUNs with SnapVault the示例工作流程」使用「Create Volume」命令兩次。不過、它會使用「Create Primary Volume」（建立主磁碟區）和「Create Secondary Volume」（建立次磁碟區）的顯示名稱、以適當地用於主磁碟區和鏡射目的地磁碟區。</p>

使用者輸入

準則	範例
<p>名稱：</p> <ul style="list-style-type: none"> <li>以「\$\$」字元開頭。</li> <li>在每個字的開頭使用大寫字母。</li> <li>所有詞彙和縮寫均使用大寫字母。</li> <li>請勿使用底線。</li> </ul>	<p>\$Array</p> <p>\$' Volume名稱'</p>
<p>顯示名稱：</p> <ul style="list-style-type: none"> <li>在每個字的開頭使用大寫字母。</li> <li>以空格分隔字詞。</li> <li>如果輸入具有特定單位、請直接在顯示名稱中以方括弧指定單位。</li> </ul>	<p>「Volume Name」 (磁碟區名稱)</p> <p>「Volume size (MB) (磁碟區大小 (MB) )」</p>
<p>說明：</p> <ul style="list-style-type: none"> <li>為每個使用者輸入提供有意義的說明。</li> <li>視需要提供範例。</li> </ul> <p>您應該特別是在使用者輸入預期為特定格式時執行此動作。</p> <p>使用者輸入說明會在工作流程執行期間顯示為使用者輸入的工具提示。</p>	<p>要新增至「iGroup」的啟動器。例如啟動器的IQN或WWPN。</p>
<p>類型：如果您要將輸入限制為特定的值集、請選取「Enum」作為類型。</p>	<p>傳輸協定：「iSCSI」、「FCP」、「mited」</p>
<p>類型：選取查詢作為使用者可從WFA快取中可用的值中選取的類型。</p>	<p>\$Array：查詢類型如下：</p> <pre data-bbox="840 1383 1493 1636"> SELECT     ip, name FROM     storage.array </pre>
<p>類型：當使用者輸入應限制為從查詢取得的值、或應僅限於支援的Enum類型時、請將使用者輸入標記為「鎖定」。</p>	<p>\$Array：鎖定查詢類型：只能選取快取中的陣列。\$Protocol：鎖定的Enum類型、有效值為iSCSI、FCP、混合。不支援有效值以外的其他值。</p>
<p>類型：查詢類型在查詢中新增其他欄做為傳回值、以協助儲存設備操作員做出正確的使用者輸入選擇。</p>	<p>\$Aggregate：提供名稱、總大小、可用大小、以便操作員在選取Aggregate之前先知道這些屬性。</p>

準則	範例
<p>類型：查詢類型SQL查詢的使用者輸入內容可參照其前的任何其他使用者輸入。這可用來限制其他使用者輸入的查詢結果、例如陣列的vFiler單元、集合體的磁碟區、儲存虛擬機器（SVM）中的LUN。</p>	<p>在範例工作流程「建立叢集Data ONTAP 式的流通量」中、VserverName的查詢如下：</p> <pre data-bbox="848 295 1403 802"> SELECT     vserver.name FROM     cm_storage.cluster cluster,     cm_storage.vserver vserver WHERE     vserver.cluster_id = cluster.id     AND cluster.name = '\${ClusterName}'     AND vserver.type = 'cluster' ORDER BY     vserver.name ASC </pre> <p>查詢是指\${clusternname}、其中\$clusternname是\$VserverName使用者輸入前的使用者輸入名稱。</p>
<p>類型：對於屬於布林型態的使用者輸入、請使用布林類型、值為「true、false」。這有助於在工作流程設計中直接使用使用者輸入內容來撰寫內部運算式。例如、\$UserIntutName而非\$UserIntutName ='Yes'。</p>	<p>'\$CreCIFSShare'：布林類型、有效值為「true」或「假」</p>
<p>類型：如果是字串和數字類型、當您要使用特定格式驗證值時、請在「值」欄中使用規則運算式。 使用規則運算式輸入IP位址和網路遮罩。</p>	<p>特定位點的使用者輸入可表示為「」（「A-Z」）[A-Z]-0[1-9]」。此使用者輸入內容接受「US-01」、「NB-02」等值、但不接受「NB-00」等值。</p>
<p>類型：針對數字類型、可在「值」欄位中指定範圍型驗證。</p>	<p>對於要建立的LUN數目、「值」欄位中的項目為1-20。</p>
<p>群組：將相關的使用者輸入分組至適當的儲存區、並命名群組。</p>	<p>所有儲存設備相關使用者輸入的「儲存設備詳細資料」。所有與VMware相關的使用者輸入內容均為「詳細資料」。</p>
<p>必填：如果工作流程必須執行任何使用者輸入的值、請將使用者輸入標示為必填。如此可確保使用者輸入畫面以資料方式接受使用者的輸入。</p>	<p>「Create NFS Volume」（建立NFS Volume）工作流程中的「'\$Volume Name'」。</p>
<p>預設值：如果使用者輸入的預設值可用於大部分的工作流程執行、請提供這些值。這有助於讓使用者在執行期間提供較少的輸入（如果預設值符合此目的）。</p>	<p>無</p>

## 常量、變數及傳回參數

準則	範例
常量：使用通用值定義多個命令的參數時、請定義常量。	<i>Aggregate</i> 、 <i>overimention_dthreshold</i> 、位於「Create、map and Protect LUNs with SnapVault the SURF樣本workflow」（建立、對應及保護LUN）中。
常量：名稱 <ul style="list-style-type: none"><li>• 在每個字的開頭使用大寫字母。</li><li>• 所有詞彙和縮寫均使用大寫字母。</li><li>• 請勿使用底線。</li><li>• 所有固定名稱的字母都使用大寫字母。</li></ul>	<i>Aggregate</i> 使用空間臨界值 <i>ActualVolume SizeInMB</i>
變數：為其中一個命令參數方塊中定義的物件提供名稱。變數是自動產生的名稱、可以變更。	無
變數：名稱使用小寫字元做為變數名稱。	<i>Volume1.</i> <i>CIFs_Share</i>
傳回參數：當工作流程規劃與執行在規劃期間應傳回某些已計算或選取的值時、請使用傳回參數。這些值也會在從Web服務執行工作流程時、以預覽模式提供。	<i>Aggregate</i> ：如果使用資源選擇邏輯選取 <i>Aggregate</i> 、則實際選取的 <i>Aggregate</i> 可定義為傳回參數。

## 建立遠端系統類型驗證指令碼的準則

您必須瞭解建立驗證指令碼的準則、這些指令碼可用來測試OnCommand Workflow Automation 您在WFA (WFA) 中定義的遠端系統類型。

- 您建立的Perl指令碼必須類似於驗證指令碼視窗中提供的範例指令碼。
- 驗證指令碼的輸出必須與範例指令碼類似。

### 驗證指令碼範例

```

# Check connectivity.
# Return 1 on success.
# Return 0 on failure and set $message
sub checkCredentials {
my ($host, $user, $passwd, $protocol, $port, $timeout) = @_;
#
# Please add the code to check connectivity to $host using $protocol here.
#
return 1;
}

```

## 建立資料來源類型的準則

您必須瞭解建立資料來源類型的準則、這些資料來源類型可用來定義OnCommand Workflow Automation for the Sof the WFA (WFA) 的自訂資料來源。

您可以使用下列其中一種方法來定義資料來源類型：

- SQL：您可以使用WFA SQL準則、根據外部資料庫、從資料來源定義選取查詢。
- 指令碼：您可以撰寫PowerShell指令碼、為特定的字典項目配置提供資料。

建立資料來源類型的準則如下：

- 您必須使用PowerShell語言來建立指令碼。
- PowerShell指令碼應提供目前工作目錄中每個字典項目的輸出。
- 資料檔案應命名為「dicitnstite\_enter.csv」、其中的字典項目名稱應為大小寫字元。

從Performance Advisor收集資訊的預先定義資料來源類型、使用指令碼型資料來源類型。輸出檔案的名稱為「array\_Performance · csv」和「Aggregate效能 · csv.」

- 這個「.csv」檔案應以與該字典項目屬性相同的順序來包含內容。

一個字典項目會依下列順序包含屬性：array\_ip、date、Day、hour、cpm\_busy、總計\_ops\_per秒、disk處理量\_per秒

PowerShell指令碼會以相同順序將資料新增至「.csv」檔案。

```

$values = get-Array-CounterValueString ([REF]$data)
Add-Content $arrayFile ([byte[]][char[]] "\n"
$t$arrayIP't$date't$day't$hour't$values'\n")

```

- 您應該使用編碼來確保指令碼的資料輸出正確載入WFA快取。
- 您應該在「.csv」檔案中輸入零值時使用\n。

## 版權資訊

Copyright © 2025 NetApp, Inc. 版權所有。台灣印製。非經版權所有人事先書面同意，不得將本受版權保護文件的任何部分以任何形式或任何方法（圖形、電子或機械）重製，包括影印、錄影、錄音或儲存至電子檢索系統中。

由 NetApp 版權資料衍伸之軟體必須遵守下列授權和免責聲明：

此軟體以 NETAPP 「原樣」提供，不含任何明示或暗示的擔保，包括但不限於有關適售性或特定目的適用性之擔保，特此聲明。於任何情況下，就任何已造成或基於任何理論上責任之直接性、間接性、附隨性、特殊性、懲罰性或衍生性損害（包括但不限於替代商品或服務之採購；使用、資料或利潤上的損失；或企業營運中斷），無論是在使用此軟體時以任何方式所產生的契約、嚴格責任或侵權行為（包括疏忽或其他）等方面，NetApp 概不負責，即使已被告知有前述損害存在之可能性亦然。

NetApp 保留隨時變更本文所述之任何產品的權利，恕不另行通知。NetApp 不承擔因使用本文所述之產品而產生的責任或義務，除非明確經過 NetApp 書面同意。使用或購買此產品並不會在依據任何專利權、商標權或任何其他 NetApp 智慧財產權的情況下轉讓授權。

本手冊所述之產品受到一項（含）以上的美國專利、國外專利或申請中專利所保障。

有限權利說明：政府機關的使用、複製或公開揭露須受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中的「技術資料權利 - 非商業項目」條款 (b)(3) 小段所述之限制。

此處所含屬於商業產品和 / 或商業服務（如 FAR 2.101 所定義）的資料均為 NetApp, Inc. 所有。根據本協議提供的所有 NetApp 技術資料和電腦軟體皆屬於商業性質，並且完全由私人出資開發。美國政府對於該資料具有非專屬、非轉讓、非轉授權、全球性、有限且不可撤銷的使用權限，僅限於美國政府為傳輸此資料所訂合約所允許之範圍，並基於履行該合約之目的方可使用。除非本文另有規定，否則未經 NetApp Inc. 事前書面許可，不得逕行使用、揭露、重製、修改、履行或展示該資料。美國政府授予國防部之許可權利，僅適用於 DFARS 條款 252.227-7015(b)（2014 年 2 月）所述權利。

## 商標資訊

NETAPP、NETAPP 標誌及 <http://www.netapp.com/TM> 所列之標章均為 NetApp, Inc. 的商標。文中所涉及的所有其他公司或產品名稱，均為其各自所有者的商標，不得侵犯。