



ワークフローの作成

OnCommand Workflow Automation

NetApp
October 09, 2025

This PDF was generated from <https://docs.netapp.com/ja-jp/workflow-automation/workflows/concept-tasks-involved-in-creating-workflows.html> on October 09, 2025. Always check docs.netapp.com for the latest.

目次

ワークフローの作成	1
ワークフローの作成に関連するタスク	1
ワークフローを定義する方法	2
ユーザ入力 of 定義方法	3
ユーザー入力タイプのオプション	3
コマンドパラメータのマッピング方法	6
すべてのコマンドカテゴリ	7
オブジェクトを作成するコマンド	7
オブジェクトを更新するコマンド	8
オブジェクトを削除するコマンド	8
オプションの親オブジェクトおよび子オブジェクトを処理するコマンド	8
オブジェクト間の関連付けを更新するコマンド	8
定数の定義方法	9
行の繰り返しの仕組み	9
行の繰り返し変数	9
承認点を含む行の繰り返し	10
定義済みのワークフローで行の例を繰り返します	11
リソース選択の仕組み	11
定義済みワークフローでのリソース選択の例	12
予約の仕組み	12
事前定義されたワークフローでの予約の例	13
増分命名とは何ですか	13
条件付き実行とは	15
定義済みワークフローでの条件付き実行の例	16
戻りパラメータの仕組み	16
戻りパラメータとして使用できるパラメータ	16
定義済みワークフローの戻りパラメータの例	17
どの承認ポイントがあるか	17
定義済みワークフローの承認ポイントの例	18
カスタム REST エンドポイントの実行方法	18
障害発生時の続行方法	19
ワークフロー要件チェックリストの例	19
要件チェックリストの例	19
ワークフローを作成します	23
完了後	27
ワークフローヘルプコンテンツを作成します	27

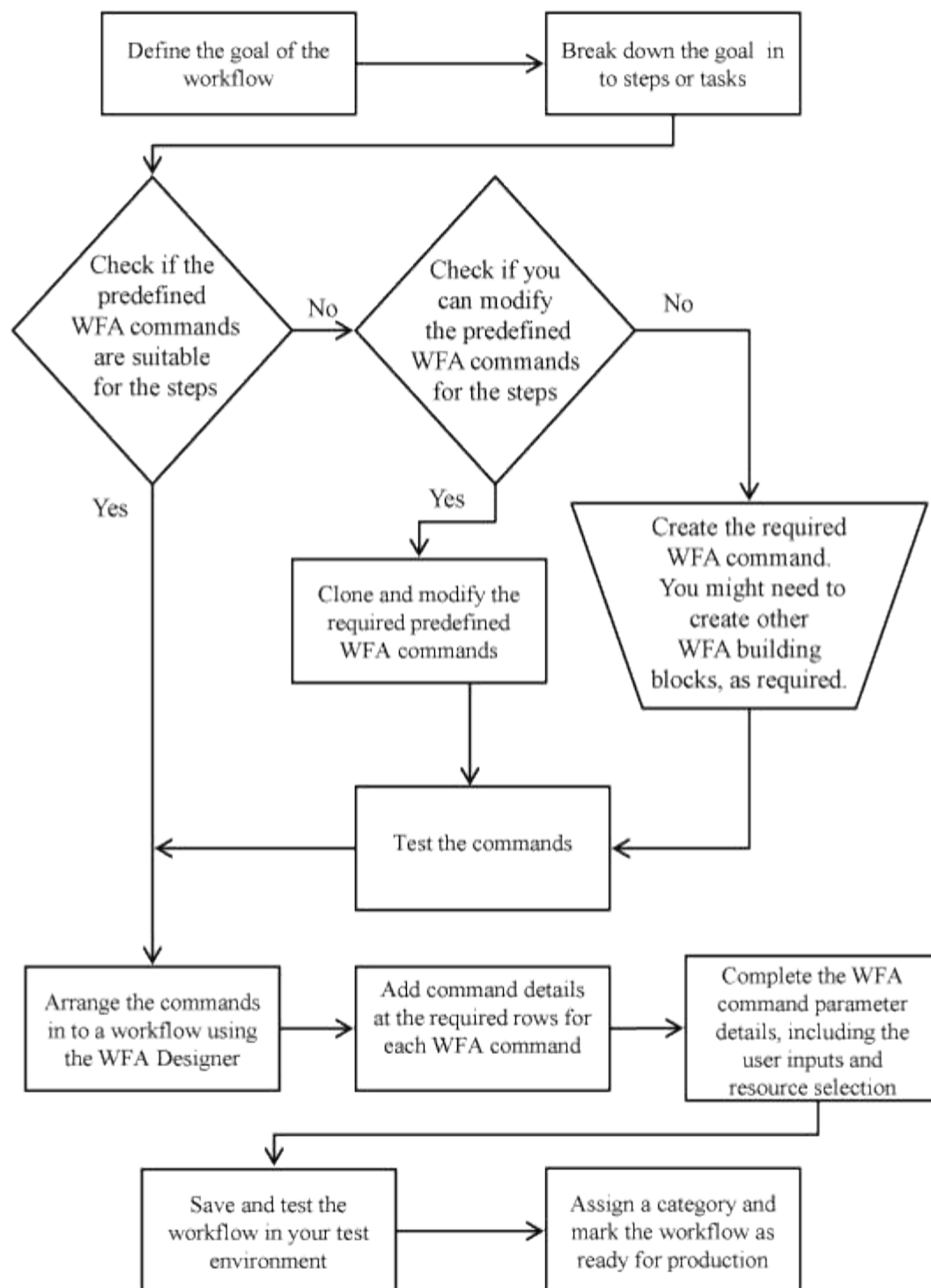
ワークフローの作成

定義済みのワークフローが要件に合わない場合は、必要なワークフローを作成できます。ワークフローを作成する前に、WFA 設計者が利用できる機能を理解し、ワークフローチェックリストを作成する必要があります。

ワークフローの作成に関連するタスク

OnCommand Workflow Automation（WFA）でストレージ自動化ワークフローを作成するには、ワークフローで実行する手順を定義し、コマンド、ファインダ、フィルタ、ディクショナリエントリなどの WFA ビルディングブロックを使用してワークフローを作成します。

次のフローチャートは、ワークフローの作成プロセスを示しています。



ワークフローを定義する方法

ワークフローの目標を、ワークフローで実行する必要がある手順に分割する必要があります。その後、ワークフローを完了するための手順を並べ替えることができます。

ワークフローとは、エンドツーエンドのプロセスを実行するために必要な一連のステップを含むアルゴリズムです。プロセスの範囲は、ワークフローの目標によって異なる場合があります。ワークフローの目的は、ストレージの運用のみ、またはネットワーク、仮想化、IT システム、その他のアプリケーションを 1 つのプロセスの一部として処理するなど、より複雑なプロセスを処理することにあります。OnCommand Workflow

Automation (WFA) ワークフローはストレージアーキテクトが設計し、ストレージオペレータが実行します。

ワークフローを定義するには、ワークフローの目標を一連の手順に分けます。たとえば、NFS ボリュームを作成するには、次の手順を実行します。

1. ボリュームオブジェクトの作成
2. 新しいエクスポートポリシーを作成し、そのポリシーをボリュームに関連付けます

ワークフローの各ステップには、WFA コマンドまたはワークフローを使用できます。WFA には、ストレージの一般的なユースケースに基づく、事前定義されたコマンドとワークフローが用意されています。特定の手順に使用できる定義済みのコマンドまたはワークフローが見つからない場合は、次のいずれかを実行できます。

- 手順に最も近い事前定義されたコマンドまたはワークフローを選択し、要件に応じて事前定義されたコマンドまたはワークフローをクローニングして変更します。
- 新しいコマンドまたはワークフローを作成します。

その場合は、新しいワークフローにコマンドやワークフローを配置して、目的を達成するためのワークフローを作成できます。

ワークフローの実行が開始されると、WFA は実行を計画し、ワークフローとコマンドの入力を使用してワークフローを実行できることを確認します。ワークフローを計画すると、すべてのリソース選択とユーザ入力为解决され、実行計画が作成されます。計画が完了すると、WFA は実行計画を実行します。実行計画は、一連の WFA コマンドと該当するパラメータで構成されます。

ユーザ入力の定義方法

OnCommand Workflow Automation (WFA) ユーザ入力は、ワークフローの実行中に使用できるデータ入力オプションです。ワークフローの柔軟性と使いやすさを高めるために、ワークフローにユーザ入力パラメータを定義する必要があります。

ユーザー入力は入力フィールドとして表示され、ワークフローのプレビューまたは実行時に関連データを入力できます。ワークフローでコマンドの詳細を指定するときに、ドル記号 (\$) でラベルまたは変数を事前に修正することによって、ユーザ入力フィールドを作成できます。たとえば、\$VolumeName と指定すると、Volume Name ユーザ入力フィールドが作成されます。WFA の [ワークフロー <workflow name>] ウィンドウの [ユーザ入力] タブに、作成したユーザ入力ラベルが自動的に入力されます。タイプ、表示名、デフォルト値、検証値などのユーザー入力属性を変更することにより、ユーザー入力のタイプを定義し、入力フィールドをカスタマイズすることもできます。

ユーザー入力タイプのオプション

- * 文字列 *

有効な値には正規表現 (A* など) を使用できます。

「0d」や「0f」などの文字列は、「0d」が double 型の 0 として評価される」のような数字として評価されます。

- * 番号 *

選択できる数値範囲を定義できます。たとえば、1 ～ 15 のように指定できます。

- * Enum *

列挙型を使用して、ユーザー入力フィールドに入力するときに選択できる列挙値を作成できます。必要に応じて、作成した列挙値をロックして、ユーザ入力に対して作成した値のみが選択されるようにすることができます。

- * クエリ *

クエリタイプは、WFA キャッシュの値からユーザ入力を選択するときに選択できます。たとえば、次のクエリを使用すると、ユーザ入力フィールドに WFA キャッシュの IP アドレスと名前の値が自動的に入力されます。 **select ip, name from storagegear.array.** クエリが取得した値をロックして ' クエリが再試行した結果のみが選択されるようにすることもできます

- * 照会（複数選択） *

クエリ（複数選択）タイプはクエリタイプに似ており、ワークフローの実行中に複数の値を選択できます。たとえば、ユーザは、共有とエクスポートとともに複数のボリュームまたはボリュームを選択できます。複数の行を選択したり、選択を 1 行に制限したりできます。行を選択すると、選択した行のすべての列から値が選択されます。

ユーザー入力のクエリ（複数選択）タイプを使用する場合は、次の関数を使用できます。

- getSize の順にクリックします
- getValueAt
- getValueAt2D
- getValueFrom2DByRowKey

- * ブール値 *

ブール型を使用して、ユーザー入力ダイアログボックスにチェックボックスを表示できます。ブール型は、「true」と「false」を持つユーザ入力に使用する必要があります。

- * 表 *

ユーザ入力のテーブルタイプを使用して、ワークフローの実行中に複数の値を入力するために使用できるテーブルの列ヘッダーを指定できます。たとえば、ノード名とポート名のリストを指定するためのテーブルなどです。列ヘッダーに次のいずれかのユーザ入力タイプを指定して、実行時に入力された値を検証することもできます。

- 文字列
- 番号
- 列挙（Enum
- ブール値
- クエリー 'String' は ' カラム・ヘッダーのデフォルトのユーザー入力タイプです別のユーザー入力タイプを指定するには、[タイプ]列をダブルクリックする必要があります。

Designer で SnapMirror ポリシーとルールを作成ワークフローを開いて、ユーザ入力タイプが「SnapMirrorPolicyRule」ユーザ入力でどのように使用されるかを確認できます。

テーブルタイプのユーザ入力を使用する場合は、次の関数を使用できます。

- getSize の順にクリックします
- getValueAt
- getValueAt2D
- getValueFrom2DByRowKey : デザイナーで * Create を開き、 Infinite Volume を備えた Storage Virtual Machine * ワークフローを設定すると、テーブルタイプの使用方法を確認できます。
 - * パスワード *

パスワードの入力用のパスワードタイプをユーザ入力に使用できます。ユーザが入力したパスワードは暗号化され、 WFA アプリケーションとログファイルに一連のアスタリスク文字で表示されます。次の関数を使用してパスワードを復号化できます。このパスワードはコマンドで使用できます。

- Perl コマンドの場合 : WFAUtil::getWfaInputPassword (\$password)
- PowerShell コマンドの場合 : Get-WfaInputPassword-EncryptedPassword\$ password

ここで、 \$password は、 WFA からコマンドに渡される暗号化されたパスワードです。

- * 辞書 *

選択したディクショナリエントリのテーブルデータを追加できます。辞書エントリ属性は、返される属性を選択します。ワークフローの実行中に、単一の値または複数の値を選択できます。たとえば、1 つまたは複数のボリュームを選択できます。デフォルトでは、単一の値が選択されています。フィルタ処理のルールを選択することもできます。ルールは、ディクショナリエントリ属性、演算子、および値で構成されます。属性には、その参照の属性も含めることができます。

たとえば、文字列「 aggr 」で始まる名前のすべてのアグリゲートを一覧表示し、使用可能なサイズが 5GB を超えるアグリゲートのルールを指定できます。グループの最初の規則は ' 属性名 ' で ' 演算子は 'starts-name'、値は aggr です同じグループの 2 番目の規則は ' 属性 available_size_mb' で ' 演算子は '>' で ' 値は '5000.' です

次の表に、ユーザ入力タイプに適用できるオプションを示します。

オプション	説明
検証中です	ユーザ入力タイプを検証して、有効な値のみがユーザから入力されるようにすることができます。 • ユーザー入力の文字列および数値タイプは、ワークフローの実行時に入力した値で検証できます。 • 文字列タイプは正規表現で検証することもできます。 • 数値タイプは数値浮動小数点フィールドであり、指定した数値範囲を使用して検証できます。

オプション	説明
ロック値	クエリーおよび列挙型の値をロックして、ユーザがドロップダウン値を上書きしないようにしたり、表示された値のみを選択できるようにしたりすることができます。
必須としてマークしています	ワークフローの実行を続行するには、ユーザ入力を必須としてマークして、特定のユーザ入力を入力する必要があります。
グループ化	関連するユーザ入力をグループ化し、ユーザ入力グループの名前を指定できます。グループは、ユーザ入力ダイアログボックスで展開および縮小できます。デフォルトで展開するグループを選択できます。
条件の適用	条件付きユーザ入力機能を使用すると、別のユーザ入力に対して入力された値に基づいてユーザ入力の値を設定できます。たとえば 'NAS プロトコルを構成するワークフロー' では 'Read/Write host lists' ユーザ入力を有効にするために 'プロトコルに必要なユーザ入力を nfs' として指定できます

コマンドパラメータのマッピング方法

Workflow Automation（WFA）コマンドのパラメータは、特定のルールに基づいて特定の属性およびディクショナリエントリ参照にマッピングされます。WFA コマンドを作成または編集するときは、コマンドパラメータをマッピングするルールを理解しておく必要があります。

コマンドパラメータのマッピングは、ワークフローでコマンドの詳細を定義する方法を定義します。ワークフロー内のコマンドの詳細を指定する場合、コマンドのマッピングされたコマンドパラメータがタブに表示されます。タブの名前は、[パラメータマッピング（Parameters Mapping）] タブの [オブジェクト名（Object Name）] 列で指定したグループ名に基づいて決まります。マッピングされていないパラメータは、ワークフローでコマンドの詳細を指定するときに [その他のパラメータ] タブに表示されます。

コマンドパラメータマッピングのルールは、コマンドカテゴリおよびワークフローエディタでのコマンドの表記方法に基づいています。

コマンドのカテゴリは次のとおりです。

- オブジェクトを作成するコマンド
- オブジェクトを更新するコマンド
- オブジェクトを削除するコマンド
- オプションの親オブジェクトおよび子オブジェクトを処理するコマンド
- オブジェクト間の関連付けを更新するコマンド

各カテゴリのルールは次のとおりです。

すべてのコマンドカテゴリ

コマンドパラメータをマッピングする場合は、ワークフローでのコマンドの使用方法に基づいたナチュラルパスを使用する必要があります。

次の例は、自然パスを定義する方法を示しています。

- ArrayIP パラメータでは、コマンドに応じて、`array.ip` 属性ではなく、ボリュームディクショナリエントリの `aggregate.arrp.ip` 属性を使用する必要があります。

これは、ワークフローでボリュームを作成し、作成されたボリュームを参照して追加の手順を実行する場合に重要です。同様の例を次に示します。

- `qtree` ディクショナリエントリの `volume-aggregate.arrp.ip`
- LUN ディクショナリエントリの `volume-aggregate.arrp.ip`
- コマンドで使用するクラスタの場合は、次のいずれかを使用する必要があります。
 - ボリュームディクショナリエントリの `vserver.cluster.primary_address`
 - `qtree` ディクショナリエントリの `volume_vserver.cluster.primary_address`

オブジェクトを作成するコマンド

このカテゴリのコマンドは、次のいずれかに使用されます。

- 親オブジェクトの検索と新規オブジェクトの定義
- オブジェクトを検索し、存在しない場合はオブジェクトを作成します

このカテゴリのコマンドには、次のパラメータマッピングルールを使用する必要があります。

- 作成されたオブジェクトの関連パラメータをオブジェクトのディクショナリエントリにマップします。
- 作成されたディクショナリエントリのリファレンスを使用して、親オブジェクトをマッピングします。
- 新しいパラメータを追加するときは、関連する属性がディクショナリエントリに存在することを確認します。

このルールの例外シナリオを次に示します。

- 作成されたオブジェクトの中には、対応するディクショナリエントリがなく、親オブジェクトだけが該当する親ディクショナリエントリにマッピングされているものがあります。たとえば、`* VIF の作成 *` コマンドなどです。この場合、アレイはアレイディクショナリエントリにのみマッピングできます。
- パラメータのマッピングは必要ありません

たとえば、`* Create or resize aggregate *` コマンドの `ExecutionTimeout` パラメータは、マッピングされていないパラメータです。

このカテゴリの証明済みコマンドの例を次に示します。

- ボリュームを作成します
- LUN を作成します

オブジェクトを更新するコマンド

このカテゴリのコマンドは、オブジェクトを検索し、属性を更新するために使用されます。

このカテゴリのコマンドには、次のパラメータマッピングルールを使用する必要があります。

- 更新されたオブジェクトをディクショナリエントリにマッピングします。
- オブジェクトに対して更新されたパラメータをマッピングしないでください。

例えば、`* Set Volume State *` コマンドでは、`Volume` パラメーターはマップされていますが、新しい状態はマップされていません。

オブジェクトを削除するコマンド

このカテゴリのコマンドは、オブジェクトを検索して削除するために使用されます。

コマンドによって削除されたオブジェクトをディクショナリエントリにマッピングする必要があります。例えば、`* ボリュームの削除 *` コマンドでは、削除するボリュームは、ボリューム辞書エントリの関連する属性と参照にマップされます。

オプションの親オブジェクトおよび子オブジェクトを処理するコマンド

このカテゴリのコマンドには、次のパラメータマッピングルールを使用する必要があります。

- コマンドの必須パラメータを、オプションのパラメータからの参照としてマッピングしないでください。

このルールは、コマンドが特定の親オブジェクトのオプションの子オブジェクトを扱う場合に適しています。この場合、子オブジェクトと親オブジェクトを明示的にマッピングする必要があります。たとえば、`* 重複排除ジョブの停止 *` コマンドでは、アレイまたは特定アレイのすべてのボリュームと一緒に指定した場合に、特定のボリュームで実行中の重複排除ジョブを停止します。この場合、配列パラメータは、`Volume.1` ではなく、アレイディクショナリエントリに直接マッピングする必要があります。Array は、このコマンドのオプションパラメータであるためです。

- 親と子の関係が論理レベルでディクショナリエントリ間に存在するが、特定のコマンドの実際のインスタンス間に存在しない場合は、それらのオブジェクトを個別にマッピングする必要があります。

たとえば、`* ボリュームの移動 *` コマンドでは、ボリュームが現在の親アグリゲートから新しいデスティネーションアグリゲートに移動されます。したがって、ボリュームパラメータはボリュームディクショナリエントリにマッピングされ、デスティネーションアグリゲートパラメータは、アグリゲートディクショナリエントリには個別にマッピングされますが、`volume.aggregate.name` にはマッピングされません。

オブジェクト間の関連付けを更新するコマンド

このカテゴリのコマンドでは、関連付けとオブジェクトの両方を、関連するディクショナリエントリにマッピングする必要があります。たとえば、`Add Volume to vFiler` コマンドでは、`Volume` パラメータと `vFiler` パラメータが、ボリュームおよび vFiler ディクショナリエントリの関連属性にマッピングされます。

定数の定義方法

1 つのワークフローで利用できる値を定義するために、定数を作成して使用できます。定数はワークフローレベルで定義されます。

ワークフローで使用されている定数とその値は、計画および実行中のワークフローの監視ウィンドウに表示されます。定数には一意の名前を使用する必要があります。

定数を定義する際には、次の命名規則を使用できます。

- 各単語の最初の文字に大文字を使用します。単語間にアンダースコアやスペースは使用しません

すべての用語と略語で大文字を使用する必要があります。たとえば、「ActualVolumeSizeInMB」と入力します。

- すべての文字で大文字

アンダースコアを使用して単語を区切ることができますたとえば 'aggregate_Used_space_threshold' のようにします

ワークフロー定数には、次の値を指定できます。

- 数字
- 文字列
- MVEL 式

式は、ワークフローの計画フェーズと実行フェーズで評価されます。式では、ループで定義されている変数を参照しないでください。

- ユーザ入力
- 変数 (variables)

行の繰り返しの仕組み

ワークフローには、コマンドとコマンドの詳細が行に表示されます。検索条件の結果に基づいて、一定のイテレーション数または繰り返し回数に対して繰り返されるコマンドを行に指定できます。

行のコマンドの詳細を指定して、特定の回数繰り返したり、ワークフローの設計時に繰り返したりできます。ワークフローは ' ワークフローの実行時または実行スケジュール時に行を繰り返す必要のある回数を指定できるように設計することもできますオブジェクトの検索条件を指定し、行のコマンドを設定して、検索条件から返されたオブジェクトの数を繰り返すことができます。特定の条件が満たされたときに行を繰り返すように設定することもできます。

行の繰り返し変数

変数リストでは、行のイテレーション中に操作できる変数を指定できます。変数には、名前、変数の初期化に使用する値、および行の繰り返しのたびに評価される MVEL 式を指定できます。

次の図は、行の繰り返しオプションと行の繰り返し変数の例を示しています。

Row Repetition Details ?

Repeats*

Number of times

Number of Times*

Number of times

For every resource in a group

Index Variable*

Index1

Variables

Name	Initial Value	Expression
size_to_alloc	SIZE_MB	(int)size_to_allocated - getData()

Add

Remove

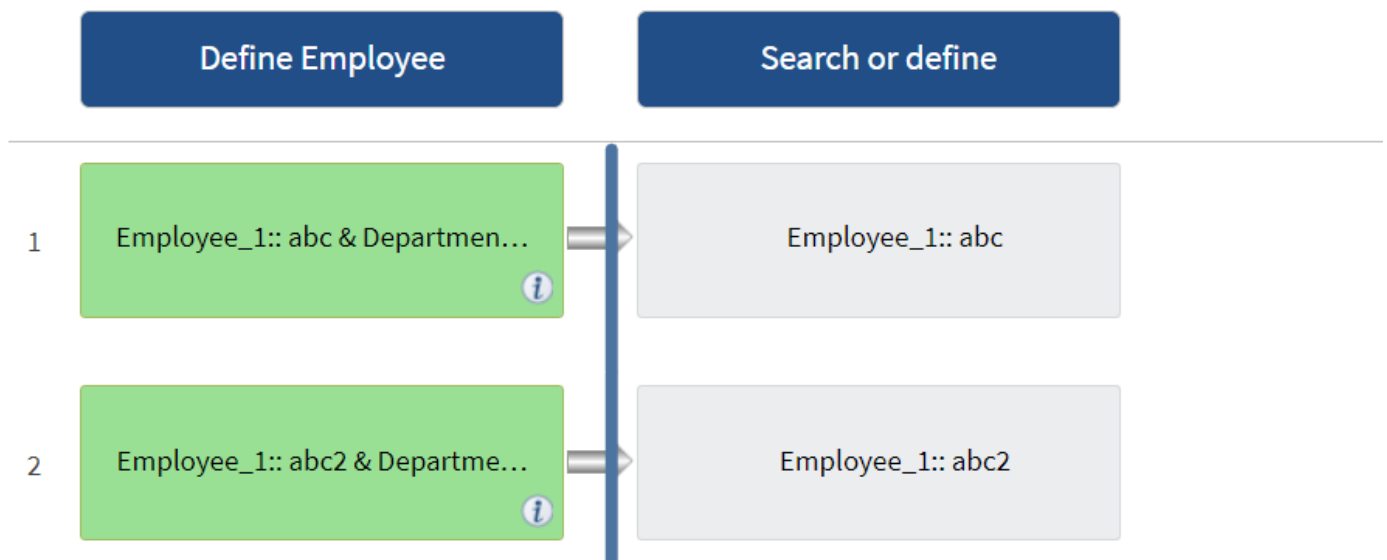
Cancel

OK

承認点を含む行の繰り返し

コマンドおよび含まれる承認ポイントに対して繰り返し行を指定すると、承認ポイントの前にコマンドのすべてのイテレーションが実行されます。承認ポイントを承認すると、次の承認ポイントまで、連続するすべてのコマンドの実行が続行されます。

次の図は、承認ポイントがワークフローに含まれている場合に繰り返し行の繰り返しがどのように実行されるかを示しています。



定義済みのワークフローで行の例を繰り返します

Designer で次の定義済みワークフローを開いて、リピート行の使用方法を理解できます。

- clustered Data ONTAP の NFS ボリュームを作成
- clustered Data ONTAP ストレージに VMware NFS データストアを作成します
- クラスピアリングを確立する
- clustered Data ONTAP ボリュームを削除

リソース選択の仕組み

OnCommand Workflow Automation（WFA）では、検索アルゴリズムを使用して、ワークフローの実行に使用するストレージリソースを選択します。ワークフローを効率的に設計するには、リソースの選択の仕組みを理解しておく必要があります。

WFA では、検索アルゴリズムを使用して、vFiler ユニット、アグリゲート、仮想マシンなどのディクショナリエントリのリソースが選択されます。選択したリソースを使用してワークフローが実行されます。WFA 検索アルゴリズムは WFA ビルディングブロックの一部であり、ファインダとフィルタが含まれています。必要なリソースを特定して選択するために、検索アルゴリズムでは、Active IQ Unified Manager、VMware vCenter Server、データベースなど、さまざまなリポジトリからキャッシュされたデータを検索します。デフォルトでは、すべてのディクショナリエントリで、自然キーに基づいてリソースを検索するためのフィルタが使用できます。

ワークフロー内の各コマンドのリソース選択基準を定義する必要があります。また、Finder を使用して、ワークフローの各行にリソース選択条件を定義することもできます。たとえば、特定の量のストレージスペースを必要とするボリュームを作成する場合、「ボリュームの作成」コマンドで「使用可能な容量でアグリゲートを検索」ファインダを使用すると、指定した量の使用可能なスペースを持つアグリゲートを選択してボリュームを作成できます。

vFiler ユニット、アグリゲート、仮想マシンなど、ディクショナリエントリリソース用のフィルタルールのセットを定義できます。フィルタルールには、1 つ以上のルールグループを含めることができます。ルールは、ディクショナリエントリ属性、演算子、および値で構成されます。属性には、その参照の属性も含めることができます。たとえば、次のようにアグリゲートのルールを指定できます。List all aggregates that have names starting with the string "aggr>" and have more than 5GB available space。グループの最初のルールは属性 "name" で、演算子 "starts-name", および値 "aggr") です。同じグループの 2 番目の規則は '属性 "available_size_MB" で '演算子 ">" と値 "5000" です。一連のフィルタルールとパブリックフィルタを定義できます。Finder を選択した場合は、「フィルタルールを定義」オプションが無効になります。フィルタルールを定義（Define filter rules）チェックボックスを選択した場合、Finder として保存（Save As Finder）オプションは無効になります。

フィルタやファインダのほかに、検索コマンドや定義コマンドを使用して、使用可能なリソースを検索することもできます。検索コマンドまたは定義コマンドは、No-op コマンドよりも推奨されます。検索および定義コマンドを使用して、証明済みディクショナリエントリタイプとカスタムディクショナリエントリタイプの両方のリソースを定義できます。検索コマンドまたは定義コマンドではリソースが検索されますが、リソースに対する操作は実行されません。ただし、リソースの検索に Finder を使用する場合は、コマンドのコンテキストで Finder を使用し、コマンドで定義されたアクションがリソースに対して実行されます。検索コマンドまたは定義コマンドによって返されるリソースは、ワークフロー内の他のコマンドの変数として使用されます。

次の図は、リソースの選択にフィルタが使用されていることを示しています。

定義済みワークフローでのリソース選択の例

Designer で次の定義済みワークフローのコマンドの詳細を開き、リソース選択オプションの使用方法を理解できます。

- clustered Data ONTAP の NFS ボリュームを作成
- クラスタピアリングを確立する
- clustered Data ONTAP ボリュームを削除

予約の仕組み

OnCommand Workflow Automation のリソースリザーベーション機能では、ワークフローを正常に実行するために必要なリソースが確保されています。

WFA のコマンドを使用すると、WFA キャッシュデータベースにリソースを追加したあと、通常はキャッシュを取得したあとに、必要なリソースをリザーブしたり予約を解除したりできます。リザーベーション機能を使用すると、WFA の設定で設定したリザーベーションの有効期限まで、リザーブリソースをワークフローに確実に割り当てることができます。

リザーベーション機能を使用すると、リソースの選択時に他のワークフローで予約されているリソースを除外できます。たとえば、あるアグリゲート上に 100GB のスペースをリザーブしたワークフローが 1 週間後に実行されるようにスケジュール設定されている場合、また、* Create Volume * コマンドを使用して別のワークフローを実行している場合、実行中のワークフローは、スケジュールされたワークフローによって予約されたスペースを消費して新しいボリュームを作成することはありません。また、リザーベーション機能を使用すると、ワークフローを並行して実行することができます。

実行用のワークフローをプレビューするとき、WFA プランナーは、キャッシュデータベース内の既存のオブジェクトを含むすべてのリザーブオブジェクトを考慮します。リザーベーションを有効にした場合、スケジュールされたワークフローおよび並行して実行されるワークフローの影響、およびワークフローを計画する際にストレージ要素の有無が考慮されます。

次の図の矢印は、ワークフローで予約が有効になっていることを示しています。

Workflow 'Abort SnapMirror relationship' ⓘ

Details	Define Workflow	User Inputs	Constants	Return Parameters	Help Content	Advanced
Workflow Name *	Abort SnapMirror relationship					
Entity Version *	1.0.0					
Categories	Data Protection					
Workflow Description	The 'Abort SnapMirror' workflow stops ongoing transfers for a					
Ready For Production	☒					
Consider Reserved Elements	☒ ←					
Enable Element Existence Validation	☒					
Minimum Software Versions	Clustered Data ONTAP 8.2.0					

事前定義されたワークフローでの予約の例

設計者で次の定義済みワークフローを開いて、予約の使用方法を理解できます。

- クローン環境
- clustered Data ONTAP ボリュームを作成
- クラスピアリングを確立する
- clustered Data ONTAP ボリュームを削除

増分命名とは何ですか

増分命名とは、パラメータの検索結果に基づいてワークフロー内の属性に名前を付けるためのアルゴリズムです。属性には、増分値またはカスタム式に基づいて名前を付けることができます。命名機能が強化され、要件に基づいた命名規則を実装できます。

ワークフローを設計する際に増分の命名機能を使用すると、ワークフローで作成されたオブジェクトに動的に名前を付けることができます。この機能を使用すると、リソース選択フィーチャーを使用してオブジェクトの検索条件を指定でき、検索条件によって返される値がオブジェクトの属性に使用されます。また、指定された

検索条件でオブジェクトが見つからなかった場合は、属性の値を指定できます。

属性に名前を付けるには、次のいずれかのオプションを使用します。

- 増分値とサフィックスを指定します

検索条件で検出されたオブジェクトの値とともに使用する値を指定し、指定した数だけ増分することができます。たとえば 'filer name_unique number_environment' という命名規則を使用してボリュームを作成する場合 'Finder' を使用して最後のボリュームを名前のプレフィックスで検索し一意の番号を 1 ずつ増やして 'ボリューム名にサフィックス名を追加できます最後に見つかったボリューム名の接頭辞が _VF_023_prod で、3 つのボリュームを作成している場合、作成されるボリュームの名前は _VF_024_prod、VF_025_prod、および VF_026_prod です。

- カスタム式を指定する

検索条件で検出されたオブジェクトの値とともに使用する値を指定し、入力した式に基づいて値を追加できます。たとえば 'last volume name_environment name が 1 でパディングされたボリュームを作成する場合' は 'last_volume.name +' + *nextName("lab1")* という式を入力できます見つかった最後のボリューム名が VF_023 の場合、作成されたボリュームの名前は VF_023_lab2 です。

次の図に、命名規則を指定するカスタム式を指定する方法を示します。

Incremental Naming Wizard for Volume : name ?

X

The Incremental Naming wizard allows you to define the value of **name** based on a search for an existing **Volume**

Search criteria for existing Volume

Volume Name : \$VolumeName, Cluster Name or IP Address : \$...

Enter a value for **name** if no **Volume** matches the above search criteria

PRE_8_2_CLUSTER

if **Volume** was found using above search criteria, set value for **name** by

providing a custom expression ▼

Custom expression

last_volume.name

Cancel

Save

条件付き実行とは

条件付き実行は、指定された条件が満たされたときにコマンドを実行できるワークフローを設計するのに役立ちます。

ワークフロー内のコマンドの実行は動的に実行できます。各コマンドの実行条件、またはワークフロー内のコマンド行を指定できます。たとえば '特定のデータセットが検出された場合にのみ 'Add volume to dataset' コマンドを実行し 'データセットが見つからない場合にワークフローを失敗させないようにすることができます。この場合は、「Add volume to dataset」コマンドを有効にして特定のデータセットを検索し、見つからない場合はワークフローでコマンドを無効にできます。

コマンドの条件付き実行のオプションは '辞書オブジェクトタブ' およびパラメータのコマンド用パラメータダイアログの詳細タブで使用できます

ワークフローを中止したり、ワークフロー内の特定のコマンドを無効にしたりできます。また、次のいずれかのオプションを使用して実行するようにコマンドを設定することもできます。

- 条件なし

- 指定した変数が見つかった場合
- 指定した変数が見つからない場合
- 指定した式が true の場合

また、特定の時間間隔を待機するようにコマンドを設定することもできます。

定義済みワークフローでの条件付き実行の例

Designer で次の定義済みワークフローのコマンド詳細を開き、コマンドの条件付き実行の使用方法を理解できます。

- 基本的な clustered Data ONTAP ボリュームを作成
- clustered Data ONTAP の NFS ボリュームを作成

戻りパラメータの仕組み

戻りパラメータは、ワークフローの計画フェーズのあとに使用できるパラメータです。これらのパラメータから返される値は、ワークフローのデバッグに役立ちます。戻りパラメータの仕組みと、デバッグワークフローへの戻りパラメータとして使用できるパラメータについて理解しておく必要があります。

ワークフローでは、変数属性、式、ユーザ入力値などの一連のパラメータを戻りパラメータとして指定できます。ワークフローの実行中に、指定したパラメータの値が計画フェーズで入力され、ワークフローの実行が開始されます。これらのパラメータの値は、ワークフローのその特定の実行での計算方法で返されます。ワークフローをデバッグする場合は、パラメータから返された値を参照します。

ワークフロー内の必須の戻りパラメータを指定すると、これらのパラメータの計算値または選択した値を確認できます。たとえば、リソース選択ロジックを使用してワークフローでアグリゲートを選択する場合、戻りパラメータとしてアグリゲートを指定することで、ワークフローの計画時に選択されたアグリゲートを確認できます。

ワークフローをデバッグするための戻りパラメータの値を参照する前に、ワークフローの実行が完了していることを確認する必要があります。戻りパラメータの値は、ワークフローの実行ごとに設定されます。ワークフローをいくつか実行したあとに戻りパラメータを追加した場合、そのパラメータの値は、パラメータの追加後のみ実行できます。

戻りパラメータとして使用できるパラメータ

パラメータを返します	例
スカラーである変数属性	「ボリューム名」変数の属性である volume1.name、`
定数	max_volume_size
ユーザ入力	\$clusterName の略

パラメータを返します	例
変数属性、定数、およびユーザー入力を含む MVEL 式	volume1.name+'-'+\$clusterName
コマンドの実行時に追加する戻りパラメータ	PowerShell コマンドで次の行を使用すると、\$volumeUUID パラメータが戻りパラメータとして追加されます。Add-WfaWorkflowParameter -Name "VolumeUUID" -value "12345"-AddAsReturnParameter \$true

定義済みワークフローの戻りパラメータの例

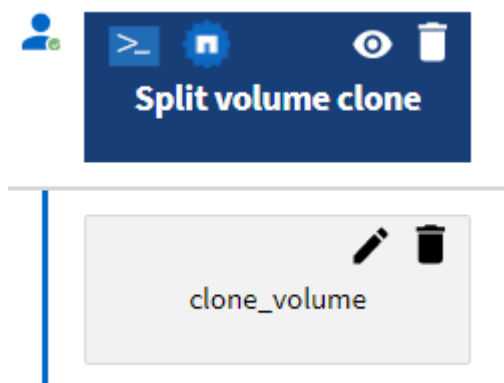
戻りパラメータの指定方法を理解するには、Designer で次の定義済みワークフローを開き、指定した戻りパラメータを確認します。

- vFiler に NFS ボリュームを作成します
- vFiler に qtrees CIFS 共有を作成します
- clustered Data ONTAP ボリュームの CIFS 共有を作成

どの承認ポイントがあるか

承認ポイントは、ワークフローでワークフローの実行を一時停止し、ユーザーの承認に基づいて再開するために使用されるチェックポイントです。

次の図に示す青色の垂直バーは承認ポイントです。



承認ポイントを使用して、ワークフローのセクションを特定の条件が満たされた後にのみ実行する必要がある場合に、ワークフローの段階的な実行を行うことができます。たとえば、次のセクションが承認される必要がある場合や、最初のセクションが正常に実行されたことが確認された場合などです。承認ポイントでは、ワークフローの一時停止と再開の間のプロセスは処理されません。E メール通知と SNMP 通知は、WFA 設定で指定されているように送信されます。ワークフローの一時停止通知を受信すると、ストレージオペレータに特定の操作を実行するよう求められます。たとえば、ストレージオペレータは、承認のために計画の詳細を管理者、承認者、オペレータに送信し、承認を受け取った時点でワークフローを再開できます。

承認が必要になることはありません。一部のシナリオでは、承認が必要になるのは、特定の条件が満たされ、承認ポイントが追加されたときに条件を設定できる場合だけです。たとえば、ボリュームのサイズを拡張する

ワークフローを考えてみましょう。ワークフローの開始時に承認ポイントを追加すると、ボリュームサイズの増加によってボリュームを含むアグリゲートのスペースが 85% 使用された場合に、ストレージオペレータが承認を得ることができます。ワークフローの実行中およびこの条件になるボリュームの選択中に、承認されるまで実行は停止されます。

承認ポイントに設定された条件には、次のいずれかのオプションがあります。

- 条件なし
- 指定した変数が見つかった場合
- 指定した変数が見つからない場合
- 指定した式が true と評価されます

ワークフロー内の承認ポイントの数に制限はありません。ワークフローのコマンドの前に承認ポイントを挿入し、承認ポイントの後にコマンドを設定して、実行前に承認を待つことができます。承認ポイントは、変更時刻、ユーザー、コメントなどの情報を提供します。これにより、ワークフローの実行が一時停止または再開された日時と理由を確認できます。承認ポイントのコメントには、MVEL 式を含めることができます。

定義済みワークフローの承認ポイントの例

Designer で次の定義済みワークフローを開いて、承認ポイントの使用方法を理解できます。

- clustered Data ONTAP ボリュームを削除
- HA ペアのコントローラとシェルフのアップグレード
- ボリュームをマイグレートする

カスタム REST エンドポイントの実行方法

OnCommand Workflow Automation（WFA）には、ワークフローを実行するためのカスタム REST エンドポイントを設定するためのメカニズムが用意されています。カスタム REST のエンドポイントは、アーキテクトがわかりやすく、わかりやすい Uniform Resource Identifier（URI）を設定してワークフローを実行するのに役立ちます。URI は、ワークフローのセマンティクスに基づいて、POST、PUT、DELETE の REST の規則に準拠します。これらの URI を使用すると、クライアント開発者はクライアントコードを簡単に開発できます。

WFA では、API 呼び出しを介してワークフローを実行するためのカスタム URI パスを設定できます。URI パス内の各セグメントは、文字列、または括弧内のワークフローのユーザー入力の有効な名前にすることができます。たとえば、「/Dev/{ProjectName}/clone.」のようになります。ワークフローは 'https://WFAServer:HTTPS_port/rest/Devation/Project1/clone/jobs.' の呼び出しとして呼び出すことができます。

URI パスの検証は次のとおりです。

- REST パスは「/」で始まる必要があります。
- 使用できる文字は、アルファベット、数字、アンダースコアです。
- ユーザー入力名は“{”で囲む必要があります



"{}" で囲まれた値が有効なユーザー入力名であることを確認する必要があります。

- 空のパスセグメントがないようにしてください。たとえば、 /、 / {} / のように指定します。
- HTTP メソッドの設定とカスタム URI パスの設定の両方を行うか、どちらも設定しないでください。

障害発生時の続行方法

障害発生時に続行機能を使用すると、ワークフローのステップを設定して、そのステップが失敗した場合でもワークフローの実行を継続できるようにすることができます。失敗した手順に対処し、「wfa_log」ファイルにアクセスするか、またはをクリックして失敗の原因となった問題を解決できます をクリックします。

このような失敗したステップが 1 つ以上あるワークフローは、実行完了後に部分的に成功した状態になります。[パラメータ for <command_name>] ダイアログボックスの [詳細設定] タブで必要なオプションを選択して、ステップが失敗した場合でもワークフローの実行を続行するようにステップを設定できます。

失敗したときに続行するようにステップが設定されていない場合、そのステップが失敗するとワークフローの実行は中止されます。

失敗時に続行するように設定されているステップが失敗した場合は、次のいずれかのオプションを使用してワークフローを実行するように設定できます。

- ワークフローの実行を中止する（デフォルトオプション）
- 次の手順から実行を続行します
- 次の行から実行を続行します

ワークフロー要件チェックリストの例

ワークフロー要件チェックリストには、計画したワークフローのコマンド、ユーザー入力、リソースなどの詳細な要件が含まれます。このチェックリストを使用してワークフローを計画し、要件とのギャップを特定することができます。

要件チェックリストの例

次のワークフロー要件チェックリストの例は ' クラスタ構成 Data ONTAP ボリュームの作成ワークフローの要件を示していますこのサンプルチェックリストをテンプレートとして使用して、ワークフローの要件をリストできます。

要件	説明
ワークフロー名	clustered Data ONTAP ボリュームを作成
カテゴリ	ストレージのプロビジョニング

要件	説明
説明	ワークフローは、特定の SVM に新しいボリュームを作成します。このワークフローは、あとで使用するためにボリュームをプロビジョニングして委譲するシナリオを想定しています。
ワークフローの仕組みを示す高レベルの概要	• ボリュームを含む SVM はユーザによって指定されます（クラスタ、SVM 名）。 • 指定したサイズに基づいてボリュームが作成されます。 • ボリュームの構成については、テンプレートで説明します。

要件	説明
詳細	• Create CM Volume * コマンドを使用します • Create CM Volume * のコマンドの詳細： ◦ 実行は常にに設定されます ◦ ボリュームの詳細は、ボリュームの属性を入力することで指定します ◦ ボリュームの構成には、* Space ながらパフォーマンスの保証設定 * テンプレートを使用します ◦ ボリュームの名前とサイズはユーザが指定します。 ボリュームは SVM ネームスペース内で「/volname」（ルートネームスペースの下）としてマウントされます。 ◦ スナップリザーブは 5% になるため、* actualVolumeSize * 機能を使用します。 ◦ SVM 参照は、次のリソース選択ロジックで定義されます。 ▪ CM SVM - ユーザが指定した名前とクラスタで SVM を検索します ▪ タイプ別の CM SVM - データ SVM のみ（タイプ = クラスタ） ▪ SVM - 状態別 - （state = running） ◦ 集約基準は、リソース選択ロジックを定義済みのファインダ（スペースしきい値と RAID タイプ別の CM 集約）として定義されます。 ▪ CM で使用可能な容量別のアグリゲート（容量 = プロビジョニングするボリュームのサイズ、ユーザがクラスタを指定） ▪ CM で、SVM に委譲されたアグリゲート ▪ RAID タイプ別の CM アグリゲート（RAID-DP） ▪ CM アグリゲート - aggr0 ではありません ▪ CM アグリゲート - 使用サイズ（しきい値 = 90、spaceToBeProvisioned = size provided、これはギャランティが volume のためです） ▪ CM アグリゲート - オーバーコミット（しきい値 = 300、spaceToBeAllocated = プロビジョニングするボリュームのサイズ） ▪ 空きスペースが最大になるアグリゲートを選択します

名前	を入力します	概要（データ値、検証など）
クラスタ	ロックされたクエリ（表形式）	• SVM をホストするクラスタ • 表形式のクエリでは、クラスタのプライマリアドレスと名前を表示できます • 名前のアルファベット順で並べ替えます
SVM	ロックされたクエリ	• ボリュームがプロビジョニングされている SVM • クエリでは、前の入力で選択したクラスタに属する SVM 名のみが表示されます 管理者またはノードではなく、クラスタタイプの SVM のみを表示（cm_storage.vserver のタイプ列） • アルファベット順に並べ替えます
ボリューム	文字列	• 作成するボリュームの名前を指定します
サイズ（GB）	整数	• プロビジョニングするボリュームのサイズ • データサイズ（スナップリザーブを考慮する必要があります）

• コマンド *

名前	説明	ステータス
CM ボリュームを作成します	SVM にボリュームを作成します	既存

• 戻りパラメータ *

名前	価値
ボリューム名	プロビジョニングされたボリュームの名前
アグリゲート名	選択したアグリゲートの名前
ノード名	ノードの名前

名前	価値
クラスタ名	クラスタの名前

- ギャップおよび問題 *

1.	
2.	
3.	
4.	
5.	

ワークフローを作成します

Workflow Automation（WFA）を使用して、データベースやファイルシステムのストレージのプロビジョニング、移行、運用停止などのタスクのワークフローを作成できます。定義済みの WFA ワークフローが要件に合わない場合は、ワークフローを作成する必要があります。

必要なもの

- WFA のビルディングブロックの概念を理解しておく必要があります。
- ワークフローに必要な繰り返し行、承認ポイント、リソース選択などの機能を理解しておく必要があります。
- ワークフロー要件チェックリストなど、ワークフローに必要な計画を完了しておく必要があります。
- ワークフローに関する情報をストレージオペレータに提供するヘルプコンテンツを作成しておく必要があります。

このタスクについて

各ワークフローの構築は、ワークフローの目標と要件によって異なる場合があります。このタスクでは、特定のワークフローの手順については説明しませんが、ワークフローの作成に関する一般的な手順について説明します。

手順

1. [* ワークフローデザイン > ワークフロー *] をクリックします。
2. をクリックします  をクリックします。
3. [* ワークフロー *] タブで、次の手順を実行します。
 - a. 必要なスキーマを展開し、目的のスキーマをダブルクリックします （コマンド）または （ワークフロー）を [使用可能なステップ（Available steps）] リストから選択します。

この手順は必要に応じて繰り返すことができます。ワークフローエディタでステップをドラッグアンドドロップすると、ステップを並べ替えることができます。

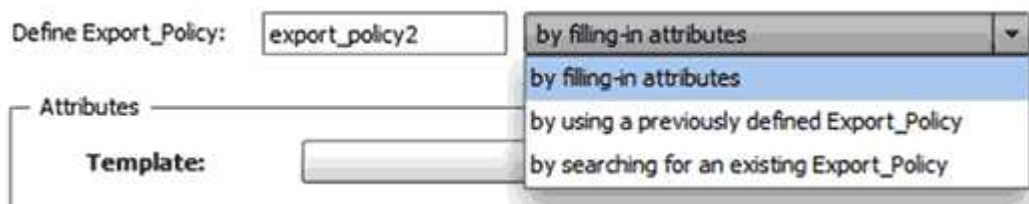
- b. * オプション： * クリック  必要な行数を追加します。これは、ステップの実行の詳細を指定するために使用されます。

各ステップは、指定された行と列で指定されたステップの詳細に基づいて実行されます。ステップは左から右、上から下の順に実行されます。

- c. 追加したステップの下にカーソルを置き、をクリックします  ステップ実行のステップの詳細を必要な行に追加します。

手順	手順
ワークフロー	[* ワークフロー * (* Workflow *)] タブで必要なユーザー入力を入力し、[* 詳細設定 * (* Advanced *)] タブで必要な条件を入力します。
コマンドを実行します	[< コマンド > のパラメータ] タブで、各オブジェクトタブをクリックし、必要なオプションを選択してオブジェクト属性を定義し、[詳細設定] タブと [その他のパラメータ] タブに必要な詳細を入力します。
検索または定義	検索または定義するディクショナリエントリオブジェクトを選択します。

次の図に、オブジェクト属性を定義するための使用可能なオプションを示します。



適切なアクションを選択します。

用途	手順
属性を入力します	次のオプションを使用して、属性の値を入力します。 • 式 • 変数（ variables ） • ユーザ入力 • リソースの選択 • 名前の増分 属性フィールドにカーソルを合わせ、をクリックする必要があります  リソース選択機能または増分命名機能を使用するには、次の手順を実行します。
以前に定義した「 object 」を使用します	オプションリストの前のボックスで ' 以前に定義したオブジェクトを選択します
既存の「 object 」を検索します	i. [* 検索条件を入力してください*]をクリックし、リソース選択機能を使用してオブジェクトを検索します。 ii. 必要なオブジェクトが見つからない場合は、実行に必要なオプションのいずれかを選択します。 ◦ ワークフローを中止する 特定のオブジェクトが見つからない場合は、ワークフローの実行を中止します。 ◦ このコマンドを無効にします This option disables only the current step and executes the workflow. **** 「 object 」の属性を入力し、コマンドを実行します This option enables you to enter the required attributes and execute the workflow.

4. 承認ポイントを挿入する場合は、をクリックします  をクリックし、承認ポイントに必要なコメントを

入力します。

承認ポイントのコメントには、MVEL 式を含めることができます。

5. をクリックします ▼ 行番号の横に表示され、次の処理を実行できます。

- 行を挿入します。
- 行をコピーします。
- 行を繰り返します。

次のいずれかのオプションを使用して、コマンドパラメータの繰り返しを指定できます。

▪ 回数

このオプションを使用すると、指定した繰り返し回数に対してコマンドを繰り返し実行できます。たとえば、「Create Qtree」コマンドを 3 回繰り返して、3 つの qtree を作成するように指定できます。

このオプションは、コマンドの実行数を動的に指定する場合にも使用できます。たとえば、作成する LUN 数に対するユーザ入力変数を作成し、ワークフローの実行時またはスケジュール時にストレージオペレータが指定した数を使用できます。

◦ グループ内のすべてのリソース

このオプションを使用して、オブジェクトの検索条件を指定できます。コマンドは、検索条件からオブジェクトが返される回数だけ繰り返し実行されます。たとえば ' クラスタ内のノードを検索し ' 各ノードに対して Create iSCSI Logical Interface コマンドを繰り返します

- 行を実行するための条件を追加します。
- 行を削除します。

6. [Details] タブで、次の手順を実行します。

- a. [ワークフロー名 *] フィールドと [ワークフロー概要 *] フィールドに必要な情報を指定します。

ワークフロー名と概要 は、ワークフローごとに一意である必要があります。

- b. * オプション：* エンティティバージョンを指定します。
- c. * オプション：予約機能を使用しない場合は、* 予約済みエレメントを考慮 * チェックボックスをオフにします。
- d. * オプション：* 同じ名前のエレメントの検証を有効にしない場合は、* エレメントの存在検証を有効にする * チェックボックスをオフにします。

7. ユーザ入力を編集する場合は、次の手順を実行します。

- a. [ユーザー入力 * (User Inputs *)] タブをクリックします。
- b. 編集するユーザ入力をダブルクリックします。
- c. [変数の編集：< ユーザー入力 > *] ダイアログボックスで、ユーザー入力を編集します。

8. 定数を追加する場合は、次の手順を実行します

- a. [定数] タブをクリックし、[*Add] ボタンを使用してワークフローに必要な定数を追加します。

複数のコマンドのパラメーターを定義するために共通の値を使用している場合は、定数を定義できます。たとえば 'Create 'LUN with SnapVault ワークフローで使用される aggregate_OLIDE_THRESHOLD 定数を参照してください

b. 各定数の名前、概要、および値を入力します。

9. [* 戻りパラメータ *] タブをクリックし、[* 追加] ボタンを使用してワークフローに必要なパラメータを追加します。

ワークフローの計画と実行で、計画中に計算値または選択した値を返す必要がある場合は、戻りパラメータを使用できます。ワークフローのプレビューまたはワークフローの実行が完了した後に、モニタリングウィンドウの [戻りパラメータ] タブで計算値または選択した値を表示できます。

Aggregate：戻りパラメータとしてアグリゲートを指定すると、リソース選択ロジックで選択されたアグリゲートを確認できます。

ワークフローに子ワークフローが含まれていて、子ワークフローの戻りパラメータ名にスペース、ドル記号 (\$) が含まれている場合、または、親ワークフローで子ワークフローの戻りパラメータ値を表示するには、親ワークフローの角かっこ内に戻りパラメータ名を指定する必要があります。

パラメータ名	指定する形式
'ChildWorkflow1.abc\$values	「 ChildWorkflow1 ["abc\$" + " 値 "]」
'ChildWorkflow1.\$values	「 ChildWorkflow1 ["\$" + " 値 "]」
「 ChildWorkflow1.value \$」	「 ChildWorkflow1.value \$」
'ChildWorkflow1.P N	「 ChildWorkflow1 ["P N"] 」
'ChildWorkflow1.return_string("HW")'	'ChildWorkflow1["return_string(\\"HW\\")"]'

10. * オプション：* ヘルプコンテンツ * タブをクリックして、ワークフロー用に作成したヘルプコンテンツ ファイルを追加します。
11. [* プレビュー] をクリックして、ワークフローの計画が正常に完了していることを確認します。
12. [OK] をクリックしてプレビューウィンドウを閉じます。
13. [保存 (Save)] をクリックします。

完了後

テスト環境でワークフローをテストしてから、ワークフローを「*_ワークフロー名_*>*_詳細_*」で本番環境向けの準備完了としてマークします。

ワークフローヘルプコンテンツを作成します

ワークフローを設計する OnCommand Workflow Automation (WFA) の管理者およびアーキテクトは、ワークフローのヘルプコンテンツを作成してワークフローに含めることができます。

必要なもの

HTML を使用して Web ページを作成する方法を理解しておく必要があります。

このタスクについて

このヘルプでは、ワークフローのワークフローに関する情報と、ワークフローを実行するストレージオペレータへのワークフローのユーザ入力について説明します。

手順

1. 次の名前のフォルダを作成します。 workflow-help
2. HTML エディタまたはテキストエディタを使用してヘルプコンテンツを作成し、「 workflow-help 」フォルダに「 index.htm 」ファイルとして保存します。

JavaScript ファイルをヘルプコンテンツの一部として含めることはできません。サポートされるファイル拡張子は次のとおりです。

- .jpg
- .jpeg
- .gif
- .png
- .xml に保存されます
- THMX
- .htm
- .html からのアクセスが可能です
- .css

Windows で作成された 'Thumbs.db' ファイルも含めることができます

3. 「 index.htm 」ファイルと、イメージなどのヘルプコンテンツに関連するその他のファイルが「 workflow-help 」フォルダにあることを確認します。
4. フォルダの「 .zip 」ファイルを作成し、「 .zip 」ファイルのサイズが 2 MB 以下であることを確認します。

「 NFS ボリュームの作成 - help.zip 」を参照してください

5. ヘルプコンテンツを作成したワークフローを編集し、 **Setup>*Help Content*>*Browse*** をクリックして、「 .zip 」ファイルをアップロードします。

著作権に関する情報

Copyright © 2025 NetApp, Inc. All Rights Reserved. Printed in the U.S. このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5252.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および / または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータ ソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用权を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用权については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

商標に関する情報

NetApp、NetAppのロゴ、<http://www.netapp.com/TM>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。