



OnCommand Workflow Automation

デザイナーの機能

OnCommand Workflow Automation

NetApp
October 09, 2025

This PDF was generated from <https://docs.netapp.com/ja-jp/workflow-automation/help/concept-how-repeat-row-works.html> on October 09, 2025. Always check docs.netapp.com for the latest.

目次

OnCommand Workflow Automation デザイナの機能	1
行の繰り返しの仕組み	1
行の繰り返し変数	1
承認点を含む行の繰り返し	2
定義済みのワークフローで行の例を繰り返します	3
どの承認ポイントがあるか	3
定義済みワークフローの承認ポイントの例	4
障害発生時の続行方法	4
リソース選択の仕組み	4
定義済みワークフローでのリソース選択の例	6
予約の仕組み	6
事前定義されたワークフローでの予約の例	7
増分命名とは何ですか	7
条件付き実行とは	9
定義済みワークフローでの条件付き実行の例	10
戻りパラメータの仕組み	10
戻りパラメータとして使用できるパラメータ	10
定義済みワークフローの戻りパラメータの例	11
方式は何ですか	11
リモートシステムの種類	12
エンティティのバージョン管理の仕組み	12
エンティティをインポートする際のバージョン管理の仕組み	14
インポートされた子エンティティのバージョンが親のバージョンに与える影響の例	15

OnCommand Workflow Automation デザイナの機能

OnCommand Workflow Automation には、ストレージワークフローの設計に役立つさまざまな機能が用意されています。

機能の詳細については、次のトピックを参照してください。

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

Index Variable*

Index1

Variables

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

Add
Remove

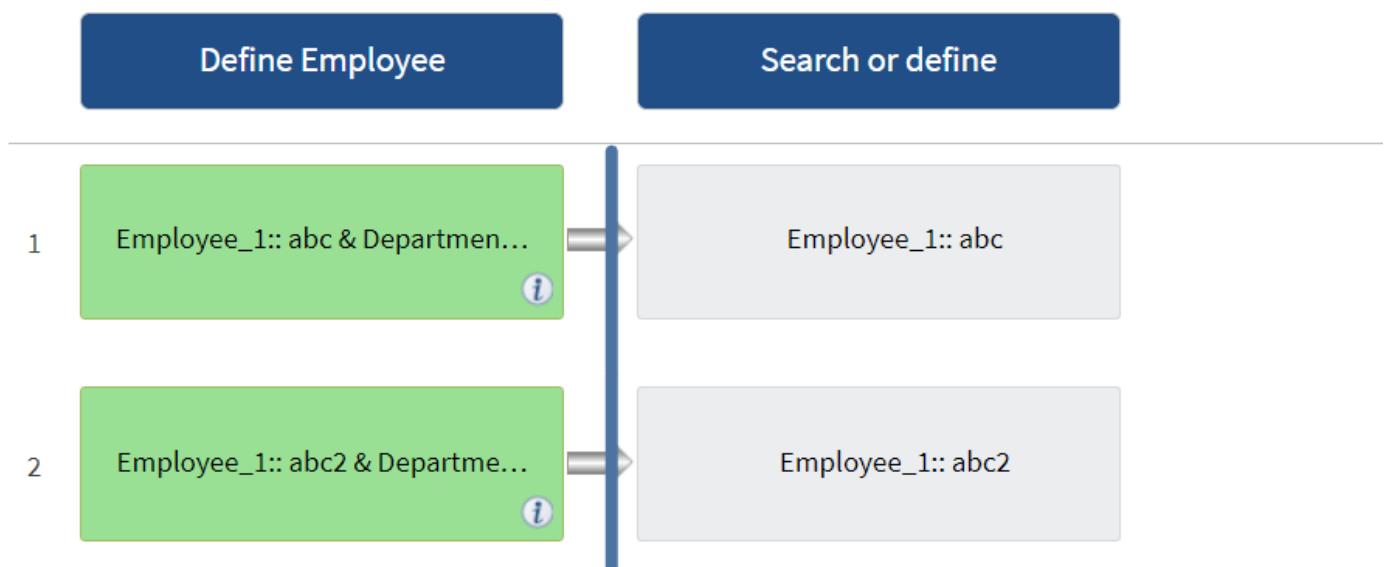
Cancel

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

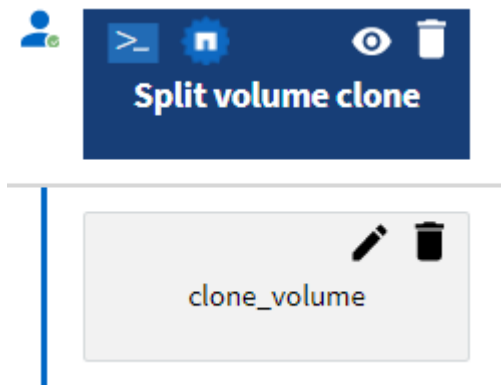
Designer で次の定義済みワークフローを開いて、リPEAT行の使用方法を理解できます。

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

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

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

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確保。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	<input checked="" type="checkbox"/>					
Consider Reserved Elements	<input checked="" type="checkbox"/> ←					
Enable Element Existence Validation	<input checked="" type="checkbox"/>					
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接頭辞で検索し、一意の番号を 1 ずつ増やすことができます。また、ボリューム名にサフィックス名を追加することもできます。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する場合、作成されるボリュームの名前は `_VF_024_prod VF_025_prod_ and _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内のアグリゲートを選択する場合 '戻りパラメータ' として 'aggregate' を指定すると 'ワークフローの計画中にどのアグリゲートが選択されたか' 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

方式は何ですか

スキームはシステムのデータモデルを表します。データモデルは、ディクショナリエントリのコレクションです。スキームを定義してから、データソースタイプを定義できます。データソースは、データの取得方法とスキームの設定方法を定義します。たとえば、VC スキームは、仮想マシン、ホスト、データストアなどの仮想環境に関するデータを取得します。

スキーマには、特定の問題を解決するようにカスタマイズされたワークフローを通じて、データを直接入力することもできます。

ディクショナリエントリは、ディクショナリエントリが作成されるときに、既存のスキームに関連付けられます。ディクショナリエントリはキャッシュクエリーにも関連付けられ、キャッシュクエリーには SQL クエリーが含まれます。

スキームでは、スクリプトベースのデータソースタイプまたは SQL データソースタイプのいずれかを使用してデータを取得できます。スクリプトはデータソースタイプの作成時に定義され、SQL クエリはキャッシュクエリで定義されます。

WFA には次のスキームが含まれています。

- * 7-Mode (ストレージ) *

Active IQ Unified Manager を介して Data ONTAP 7-Mode からデータを取得するスキーム。

- * clustered Data ONTAP (cm_storage) *

clustered Data ONTAP から Active IQ Unified Manager 経由でデータを取得するスキーム。

- * 7-Mode のパフォーマンス (パフォーマンス) *

Performance Advisor から Data ONTAP 7-Mode のパフォーマンスデータを取得するスキーム。

- * clustered Data ONTAP のパフォーマンス (cm_performance) *

Performance Advisor から clustered Data ONTAP のパフォーマンスデータを取得するスキーム。

- * VMware vCenter (VC) *

VMware vCenter からデータを取得するスキーム。

- * プレイグラウンド (プレイグラウンド) *

データを直接取り込むことができるスキーム。

リモートシステムの種類

OnCommand Workflow Automation (WFA) はリモートシステムタイプと通信します。WFA が通信できるリモートシステムのタイプはリモートシステムです。WFA ではリモートシステムタイプを設定できます。たとえば、Data ONTAP システムをリモートシステムタイプとして設定できます。

リモートシステムタイプには、次の属性があります。

- 名前
- 説明
- バージョン
- プロトコル
- ポート
- タイムアウト

リモートシステムのクレデンシャルを検証するには、リモートシステムタイプごとに Perl スクリプトを使用します。WFA で設定されているリモートシステムのクレデンシャルを保存できます。新しいカスタムリモートシステムタイプを追加または編集できます。既存のリモートシステムタイプをクローニングすることもできます。リモートシステムタイプは、関連付けられているシステムがない場合にのみ削除できます。

エンティティのバージョン管理の仕組み

コマンドやワークフローなどの OnCommand Workflow Automation エンティティはバージョン管理されています。バージョン番号を使用すると、WFA エンティティに対する変更を簡単に管理できます。

各 WFA エンティティには、「major.minor.revision」形式のバージョン番号が含まれています。たとえば、1.1.20 です。バージョン番号の各部分に最大 3 桁を含めることができます。

WFA エンティティのバージョン番号を変更する前に、次のルールを確認しておく必要があります。

- バージョン番号を現在のバージョンから以前のバージョンに変更することはできません。
- バージョンの各部分は、0~999の数値である必要があります。
- 新しいWFAエンティティは、デフォルトでは1.0にバージョン管理されます。
- エンティティのバージョン番号は、クローン作成時、または*名前を付けて保存*を使用してエンティティのコピーを保存するときに保持されます。
- WFA インストールには、エンティティの複数のバージョンを存在させることはできません。

WFA エンティティのバージョンを更新すると、その親エンティティのバージョンが自動的に更新されます。たとえば、* Create Volume * コマンドのバージョンを更新すると、* Create an NFS Volume * ワークフローが* Create Volume * コマンドの直下の親エンティティであるため、* Create Volume * コマンドのバージョンが更新されます。バージョンの自動更新は、次のように適用されます。

- エンティティのメジャーバージョンを変更すると、その直後の親エンティティのマイナーバージョンが更新されます。
- エンティティのマイナーバージョンを変更すると、その直後の親エンティティのリビジョンバージョンが更新されます。
- エンティティのリビジョンバージョンを変更しても、その直後の親エンティティのバージョンの一部は更新されません。

次の表に、WFA のエンティティとそのすぐ上の親エンティティを示します。

エンティティ (Entity)	即時親エンティティ
キャッシュクエリ	<ul style="list-style-type: none"> • データソースのタイプ
テンプレート	<ul style="list-style-type: none"> • ワークフロー
機能	<ul style="list-style-type: none"> • ワークフロー • テンプレート <div>  <p>関数に特殊文字または大文字と小文字が混在している場合、そのすぐ上の親エンティティのバージョンは更新されない可能性があります。</p> </div>
辞書	<ul style="list-style-type: none"> • テンプレート • フィルタ • キャッシュクエリ • コマンドを実行します • スクリプトメソッドを使用するデータソースのタイプ
コマンドを実行します	<ul style="list-style-type: none"> • ワークフロー

エンティティ（Entity）	即時親エンティティ
フィルタ	<ul style="list-style-type: none"> ・ ファインダ ・ ワークフロー
ファインダ	<ul style="list-style-type: none"> ・ ワークフロー
データソースのタイプ	なし
ワークフロー	なし

WFA では、バージョン番号の一部または完全なバージョン番号を使用してエンティティを検索できます。

親エンティティを削除した場合、子エンティティは保持され、削除のためにそのバージョンは更新されません。

エンティティをインポートする際のバージョン管理の仕組み

Workflow Automation 2.2 より前のバージョンからエンティティをインポートする場合、エンティティのバージョンはデフォルトで 1.0.0 になります。インポートしたエンティティがすでに WFA サーバに存在する場合は、インポートしたエンティティで既存のエンティティが上書きされます。

インポート時に WFA エンティティに変更される可能性がある項目を次に示します。

- ・ エンティティのアップグレード

エンティティは新しいバージョンで置き換えられます。

- ・ エンティティのロールバック

エンティティは以前のバージョンで置き換えられます。



エンティティのロールバックを実行すると、そのすぐ上の親エンティティのバージョンが更新されます

- ・ 新しいエンティティのインポート



.dar ファイルからエンティティを選択的にインポートすることはできません

新しいバージョンのエンティティをインポートすると、その直後の親エンティティのバージョンが更新されます。

インポートされた親エンティティに複数の子エンティティがある場合、子エンティティに対する最高レベルの変更（メジャー、マイナー、またはリビジョン）のみが親エンティティに適用されます。次の例では、このルールの仕組みについて説明します。

- ・ インポートされた親エンティティの場合、マイナー変更のある子エンティティとリビジョン変更のある子エンティティが存在する場合、マイナー変更が親エンティティに適用されます。

親のバージョンのレビジョン部分が増分されます。

- インポートされた親エンティティの場合、メジャー変更を持つ子エンティティが1つ存在し、マイナー変更を持つ子エンティティが別の子エンティティである場合、親エンティティにメジャー変更が適用されます。

親のバージョンのマイナー部分が増分されます。

インポートされた子エンティティのバージョンが親のバージョンに与える影響の例

WFA で次のワークフローを考慮してください。"Create Volume and export using NFS-Custom" 1.0.0"

ワークフローに含まれる既存のコマンドは次のとおりです。

- 「エクスポートポリシーの作成 - カスタム」 1.0.0
- 「ボリュームの作成 - カスタム」 1.0.0

インポートする .dar ファイルに含まれるコマンドは次のとおりです。

- 「エクスポートポリシーの作成 - カスタム」 1.1.0
- 「ボリュームの作成 - カスタム」 2.0.0

この .dar ファイルをインポートすると 'NFS-Custom' ワークフローを使用したボリュームの作成とエクスポートのマイナーバージョンが 1.1.0 に増加します

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。