



ライフサイクル管理

NetApp solutions for SAP

NetApp
February 25, 2026

目次

ライフサイクル管理	1
Ansibleを使用したNetApp SAP Landscape Managementの統合	1
Ansibleを使用したNetApp SAP Landscape Managementの統合	1
SAPシステムのクローニング、コピー、更新のシナリオ	1
システムの更新、コピー、クローニングのユースケース	2
Ansibleを使用して、NetApp SAP LaMaを統合	5
実装例	5
SAP LaMaプロビジョニングワークフロー-クローンシステム	11
SAP LaMaによるワークフローのプロビジョニング解除-システムの破壊	19
SAP LaMaプロビジョニングワークフロー-コピーシステム	22
SAP LaMaプロビジョニングワークフロー-システムの更新	26
プロバイダスクリプトの設定とAnsibleのプレイブック	28
まとめ	41
SnapCenter を使用して SAP HANA システムのコピーおよびクローン処理を自動化	42
SnapCenter を使用して SAP HANA システムのコピーおよびクローン処理を自動化	42
SAPシステムのコピー、更新、クローニングのシナリオ	44
システムの更新とクローニングのユースケース	45
サポートされるインフラとシナリオ	48
SnapCenter を使用したSAPシステムの更新ワークフローの概要	49
SnapCenter を使用したSAPシステムのクローニングワークフローの概要	51
ストレージSnapshotバックアップを使用したSAP HANAシステムの更新処理に関する考慮事項	53
自動化スクリプトの例	58
SAP HANAシステムをSnapCenter で更新	61
SnapCenter によるSAPシステムのクローニング	86
追加情報およびバージョン履歴の参照先	99
Libelle SystemCopyによるSAPシステムのコピー処理の自動化	99
Libelle SystemCopyによるSAPシステムのコピー処理の自動化	99
LSCおよびSnapCenter でSAP HANAシステムが更新されます	102
LSC、AzAcSnap、およびAzure NetApp Files を使用してSAP HANAシステムが更新されます	115
追加情報およびバージョン履歴の参照先	128

ライフサイクル管理

Ansibleを使用したNetApp SAP Landscape Managementの統合

Ansibleを使用したNetApp SAP Landscape Managementの統合

SAP Landscape Management (LaMa) を使用すると、SAPシステム管理者は、SAPシステムのエンドツーエンドのクローニング、コピー、更新などのSAPシステム運用を自動化できます。

著者：Michael Schlosser、Nils Bauer、NetApp

ネットアップは、SAP LaMa対応の各種Ansibleモジュールを搭載しており、SAP LaMaのAutomation Studioを通じて、NetApp SnapshotやFlexCloneなどのテクノロジーにアクセスできます。これらのテクノロジーを使用することで、SAPシステムのクローニング、コピー、更新の処理を簡易化、高速化できます。

この統合は、ネットアップのストレージソリューションをオンプレミスで運用しているお客様や、Amazon Web Services、Microsoft Azure、Google Cloud Platformなどのパブリッククラウドプロバイダでネットアップのストレージサービスを使用しているお客様が利用できます。

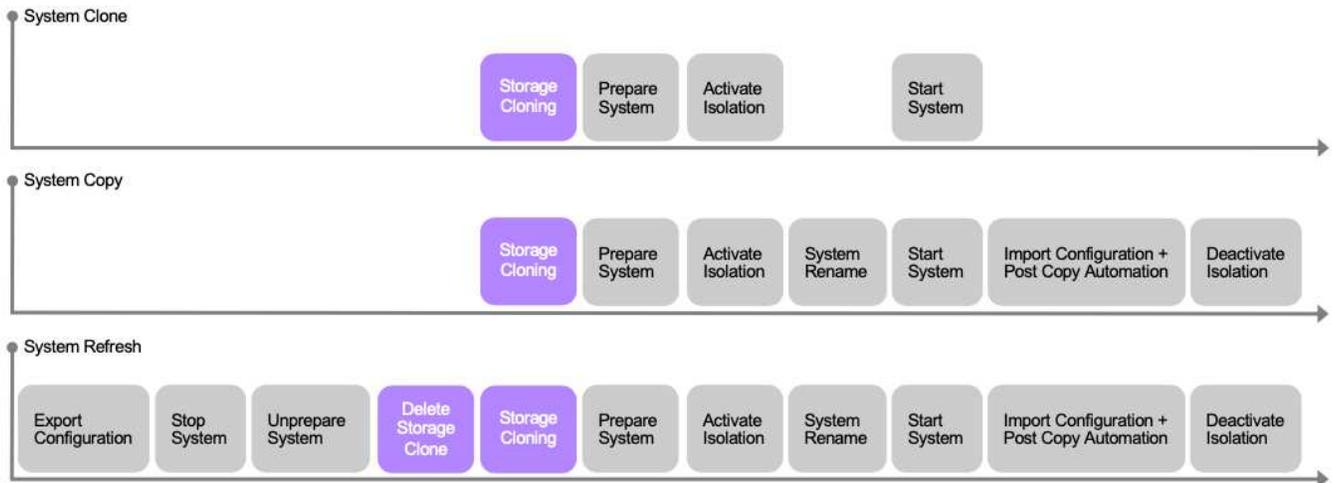
このドキュメントでは、Ansibleによる自動化を使用して、SAPシステムのコピー、クローニング、更新の処理に対応するネットアップのストレージ機能を利用したSAP LaMaの設定について説明します。

SAPシステムのクローニング、コピー、更新のシナリオ

SAPシステムのコピーという用語は、SAPシステムのクローニング、SAPシステムのコピー、SAPシステムの更新の3つのプロセスを表す同義語としてよく使用されます。ワークフローとユースケースはそれぞれ異なるため、異なる処理を区別することが重要です。

- * SAPシステムのクローン。* SAPシステムのクローンは、ソースSAPシステムの同一クローンです。SAPシステムクローンは、一般に論理的な破損に対処したり、ディザスタリカバリのシナリオをテストしたりするのに使用されます。システムのクローニング処理では、ホスト名、インスタンス番号、およびSIDの値は変更されません。そのため、本番環境と通信できないように、ターゲットシステムの適切なネットワークフェンシングを設定することが重要です。
- * SAPシステムのコピー。* SAPシステムのコピーとは、ソースSAPシステムのデータを含む新しいターゲットSAPシステムのセットアップを指します。たとえば、本番システムのデータを含む追加のテストシステムを新しいターゲットシステムとして使用できます。ソース・システムとターゲット・システムでは、ホスト名、インスタンス番号、およびSIDが異なります。
- * SAPシステムの更新。* SAPシステムの更新は、ソースSAPシステムのデータを使用した既存のターゲットSAPシステムの更新です。ターゲットシステムは、通常、品質管理システムなどのSAP転送環境の一部であり、本番システムで更新されます。ソース・システムとターゲット・システムでは、ホスト名、インスタンス番号、およびSIDが異なります。

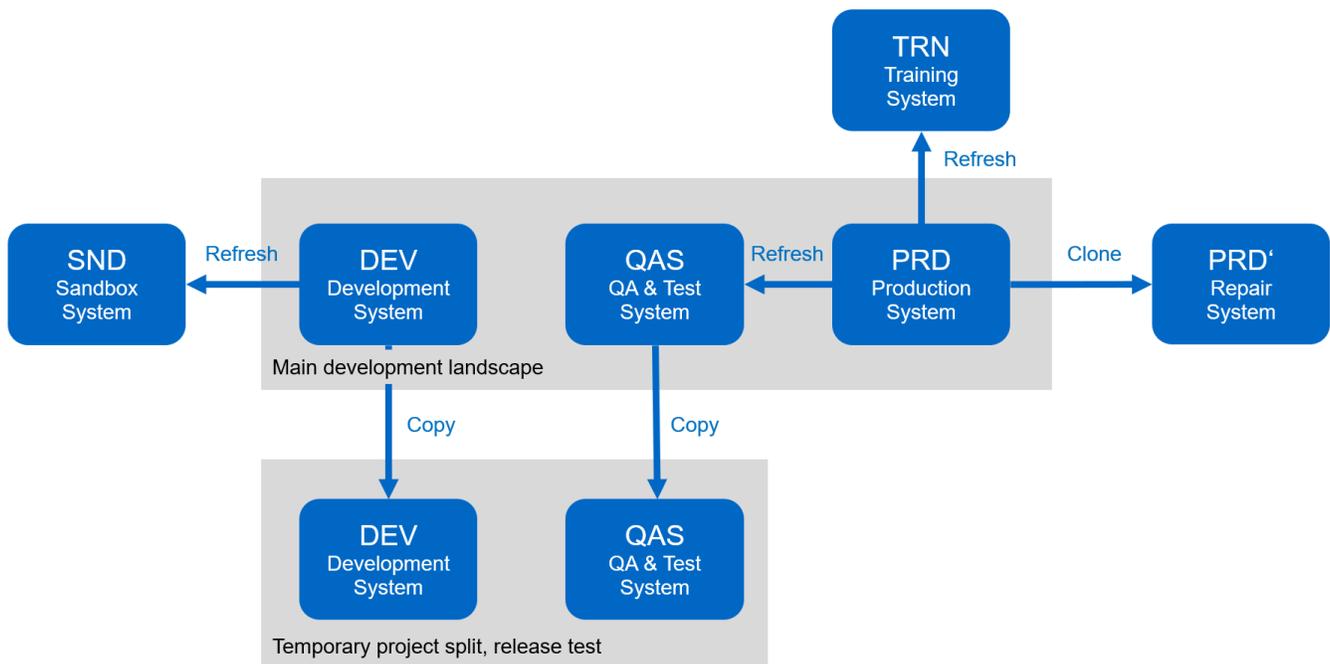
次の図に、システムのクローン、システムのコピー、またはシステムの更新の各処理で実行する必要がある主な手順を示します。紫のボックスは、ネットアップストレージ機能を統合できる手順を示しています。3つの運用はすべて、SAP LaMaを使って完全に自動化できます。



システムの更新、コピー、クローニングのユースケース

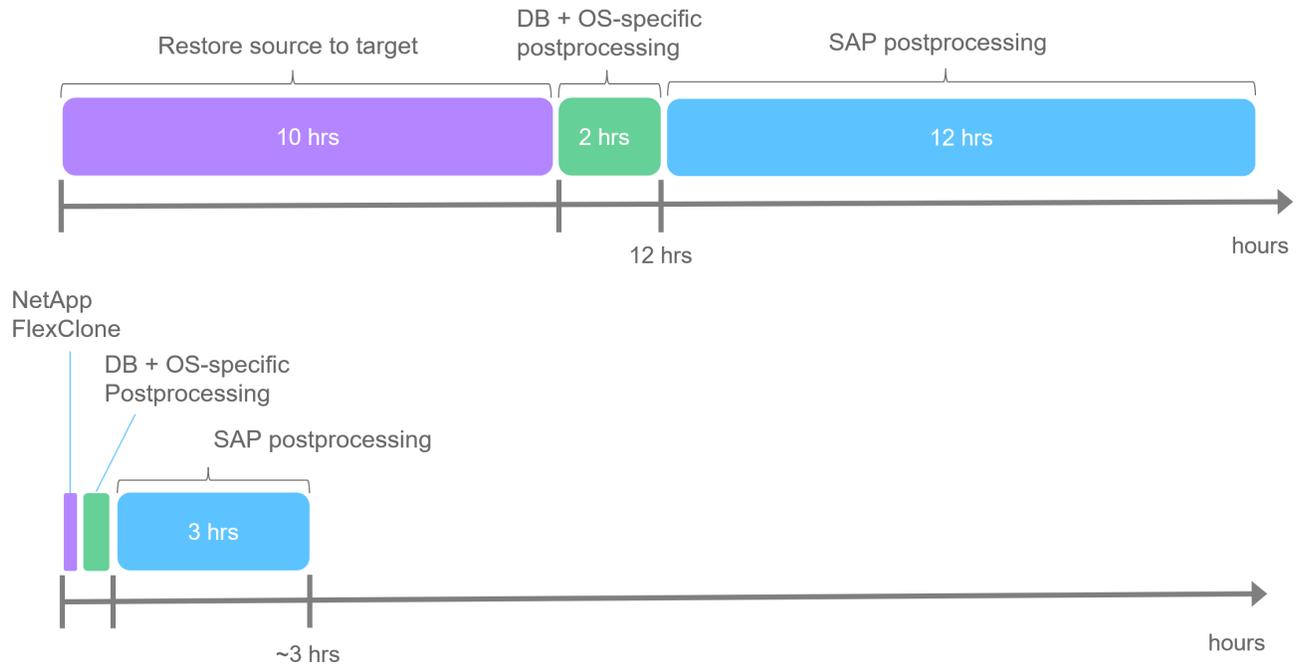
テストやトレーニングの目的で、ソースシステムのデータをターゲットシステムで使用できるようにする必要があるシナリオは複数あります。テストおよびトレーニング用のシステムは、ソースシステムのデータで定期的に更新し、現在のデータセットでテストとトレーニングが実行されていることを確認する必要があります。

このシステム更新処理は、インフラ、データベース、アプリケーションの各レイヤ上で実行される複数のタスクで構成されます。自動化のレベルによっては、数日かかる場合があります。



SAP LaMaとネットアップのクローニングワークフローを使って、インフラレイヤとデータベースレイヤで必

要なタスクを高速化し、自動化できます。SAP LaMaは、バックアップをソースシステムからターゲットシステムにリストアする代わりに、NetApp SnapshotコピーとNetApp FlexCloneテクノロジーを使用して、起動したHANAデータベースまでの必要なタスクを、次の図に示すように数時間ではなく数分で実行します。クローニングプロセスに要する時間はデータベースのサイズに左右されないため、非常に大規模なシステムでも数分で作成できます。また、オペレーティングシステムとデータベースレイヤ、およびSAPの後処理側でタスクを自動化することにより、ランタイムをさらに削減できます。



論理的破損に対処する

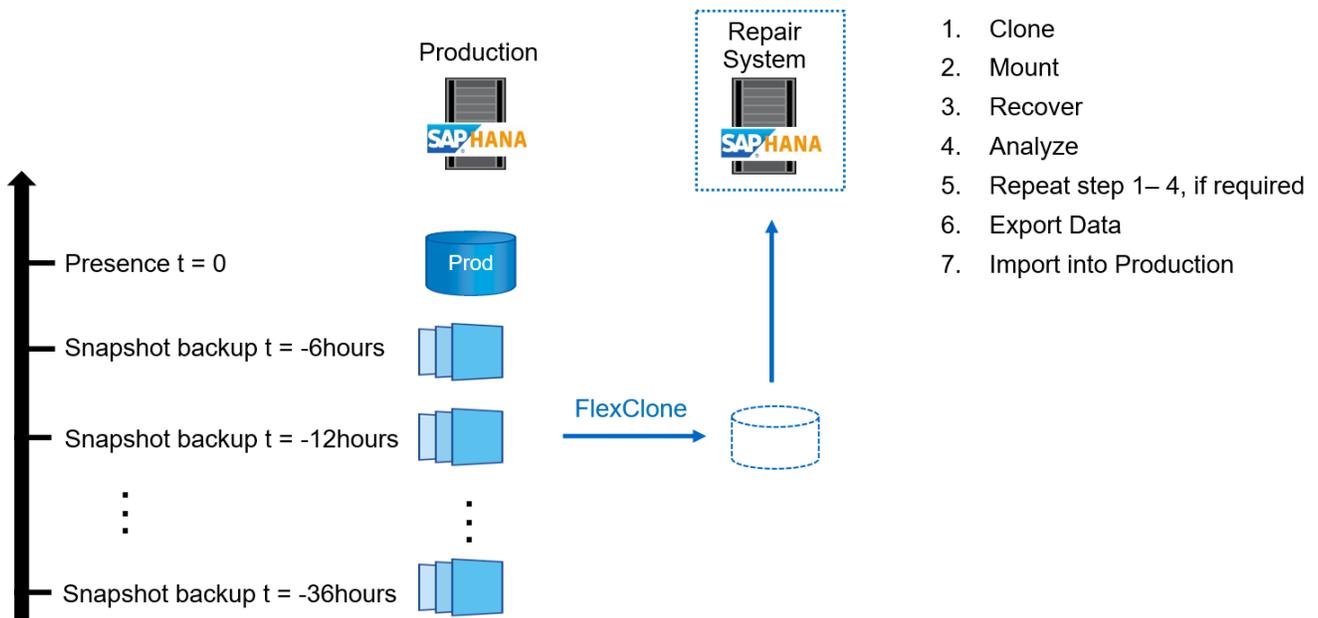
論理的破損は、ソフトウェアエラー、人為的エラー、破壊行為などが原因で発生する可能性があります。残念ながら、論理的破損は、標準的な高可用性ソリューションやディザスタリカバリソリューションでは対処できないことがよくあります。その結果、論理的な破損が発生したレイヤ、アプリケーション、ファイルシステム、またはストレージによっては、ダウンタイムを最小限に抑え、データ損失要件を許容できる範囲で満たすことができない場合があります。

最悪のケースは、SAPアプリケーションが論理的に破損した場合です。SAPアプリケーションは多くの場合、異なるアプリケーションが相互に通信してデータを交換する環境で動作します。このため、論理的な破損が発生したSAPシステムはリストアとリカバリを実行しないことを推奨します。破損が発生する前の時点でシステムをリストアすると、データが失われます。また、SAP ランドスケープは同期されず、さらにポストプロセスが必要になります。

SAPシステムをリストアする代わりに、別の修復システムで問題を分析して、システム内の論理エラーを修正する方法を推奨します。ルート原因分析には、ビジネスプロセスやアプリケーション所有者の関与が必要です。このシナリオでは、論理的破損が発生する前に格納されたデータに基づいて、修復システム（本番システムのクローン）を作成します。リペアシステム内では、必要なデータをエクスポートし、本番システムにインポートできます。このアプローチでは、本番システムを停止する必要はなく、最良のシナリオでは、データの損失だけでなく、ごくわずかなデータの損失も発生します。

リペアシステムを設定する際には、柔軟性とスピードが不可欠です。ネットアップのストレージベースのSnapshotバックアップでは、複数の整合性のあるデータベースイメージを使用し、NetApp FlexCloneテクノロジーを使用して本番システムのクローンを作成できます。ファイルベースのバックアップからリダイレク

トされたリストアを使用して修復システムを設定する場合、FlexCloneボリュームは数時間ではなく数秒で作成できます。

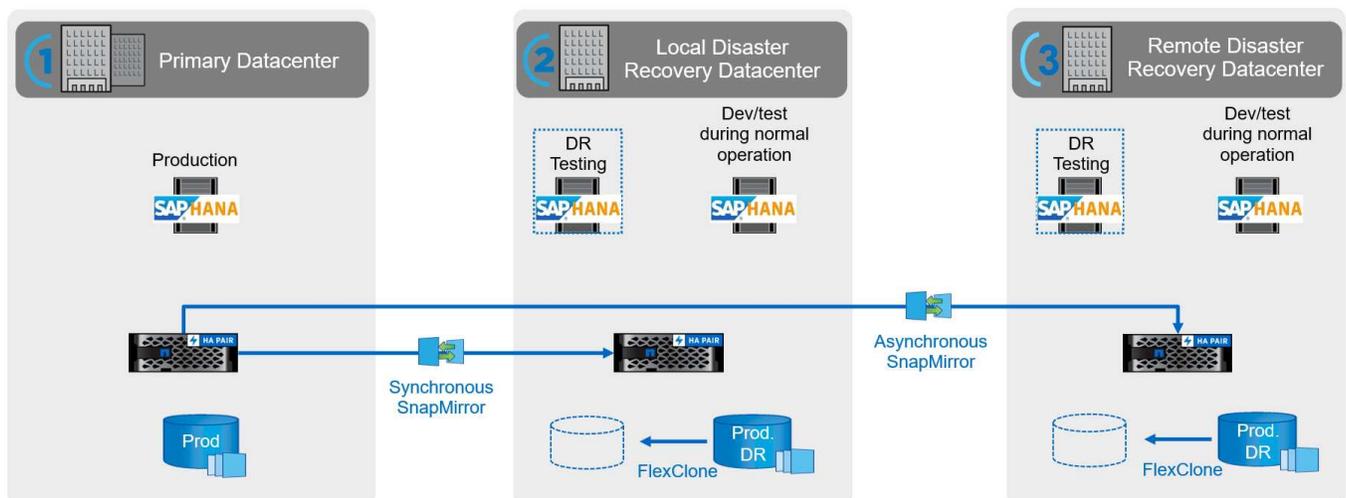


ディザスタリカバリのテスト

効果的なディザスタリカバリ戦略を策定するには、必要なワークフローをテストする必要があります。テストでは、戦略が機能するかどうか、および内部ドキュメントで十分かどうかを検証します。また、管理者は必要な手順をトレーニングできます。

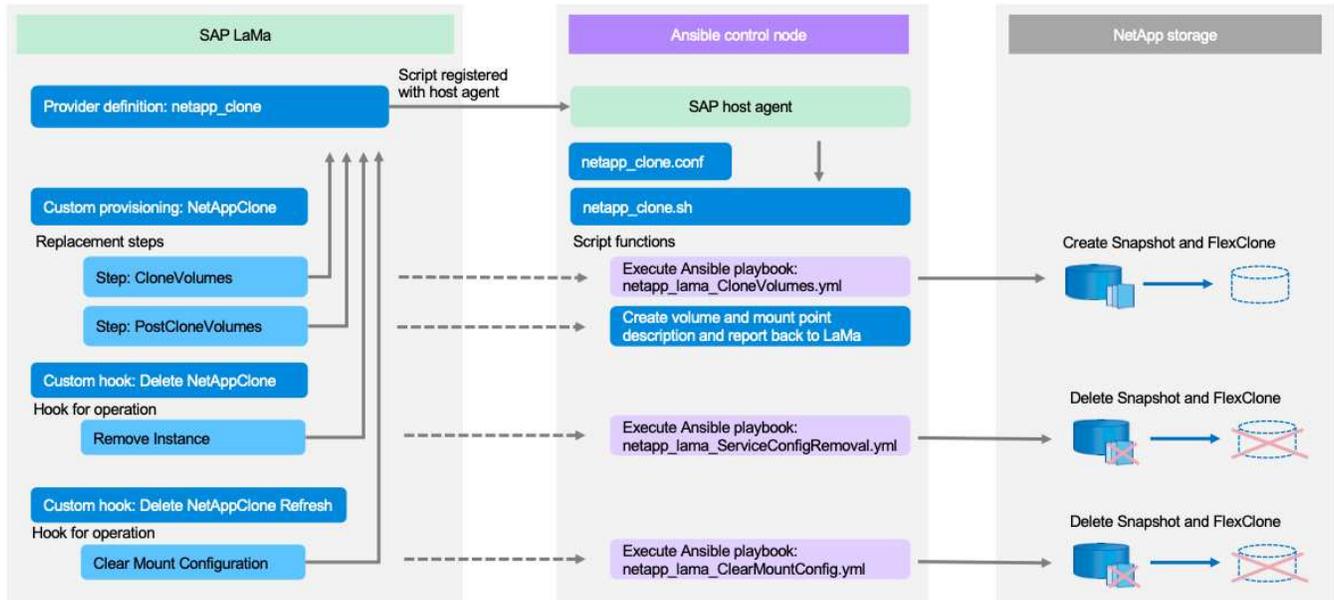
SnapMirrorを使用したストレージレプリケーションでは、RTOとRPOをリスクにさらすことなく、ディザスタリカバリのテストを実行できます。ディザスタリカバリテストは、データレプリケーションを中断することなく実行できます。非同期SnapMirrorと同期SnapMirrorのディザスタリカバリテストでは、ディザスタリカバリターゲットでSnapshotバックアップとFlexCloneボリュームを使用します。

SAP LaMaは、テスト手順全体のオーケストレーションに使用でき、ネットワークの遮断やホストのメンテナンスなどにも対応します。



Ansibleを使用して、NetApp SAP LaMaを統合

この統合アプローチでは、SAP LaMaのカスタムプロビジョニングと運用のフックを、ネットアップストレージ管理のAnsibleプレイブックと組み合わせて使用します。次の図は、LaMa側の設定の概要と、実装例の対応するコンポーネントを示しています。



Ansible制御ノードとして機能する中央ホストは、SAP LaMaからの要求を実行し、Ansibleプレイブックを使用してネットアップストレージの運用を開始するために使用されます。ホストをSAP LaMaへの通信ゲートウェイとして使用できるように、SAPホストエージェントコンポーネントをこのホストにインストールする必要があります。

LaMa Automation Studioでは、AnsibleホストのSAPホストエージェントに登録されたプロバイダが定義されます。ホストエージェント構成ファイルは、要求された処理に応じて、一連のコマンドラインパラメータを使用してSAP LaMaによって呼び出されるシェルスクリプトを指します。

LaMa Automation Studioでは、カスタムプロビジョニングとカスタムフックを定義して、プロビジョニング時にストレージのクローニング処理を実行し、システムがプロビジョニング解除されたときにクリーンアップ処理を実行します。その後、Ansible制御ノード上のシェルスクリプトによって、対応するAnsibleプレイブックが実行されます。これにより、SnapshotとFlexCloneの処理およびプロビジョニング解除ワークフローを使用したクローンの削除がトリガーされます。

NetApp AnsibleモジュールとLaMaプロバイダの定義の詳細については、以下をご覧ください。

- ["NetApp Ansibleモジュール"](#)
- ["SAP LaMaのドキュメント-プロバイダ定義"](#)

実装例

システムとストレージのセットアップでは多数のオプションを使用できるため、システムの個々のセットアップと設定の要件をテンプレートとして使用する必要があります。



このサンプルスクリプトは現状のまま提供されており、ネットアップではサポートしていません。スクリプトの最新バージョンは、mailto: ng-sapcc@netapp.com [ng-sapcc@netapp.com]にEメールでリクエストできます。

検証済みの構成と制限事項

以下の原則はサンプル導入に適用されており、お客様のニーズに合わせて調整する必要がある場合があります。

- マネージドSAPシステムは、NFSを使用してネットアップストレージボリュームにアクセスし、アダプティブデザインの原則に基づいてセットアップしました。
- NetApp Ansibleモジュール（ZAPIとREST API）でサポートされているすべてのONTAP リリースを使用できます。
- 1つのネットアップクラスタとSVMのクレデンシャルが、プロバイダスクリプトの変数としてハードコードされました。
- ストレージクローニングは、ソースSAPシステムで使用されていたのと同じストレージシステムで実行しました。
- ターゲットのSAPシステムのストレージボリュームの名前は、付録のソースと同じです。
- セカンダリストレージ（SV / SM）でのクローニングは実装されていません。
- FlexCloneスプリットを実装できませんでした。
- インスタンス番号は、ソースとターゲットのSAPシステムで同じです。

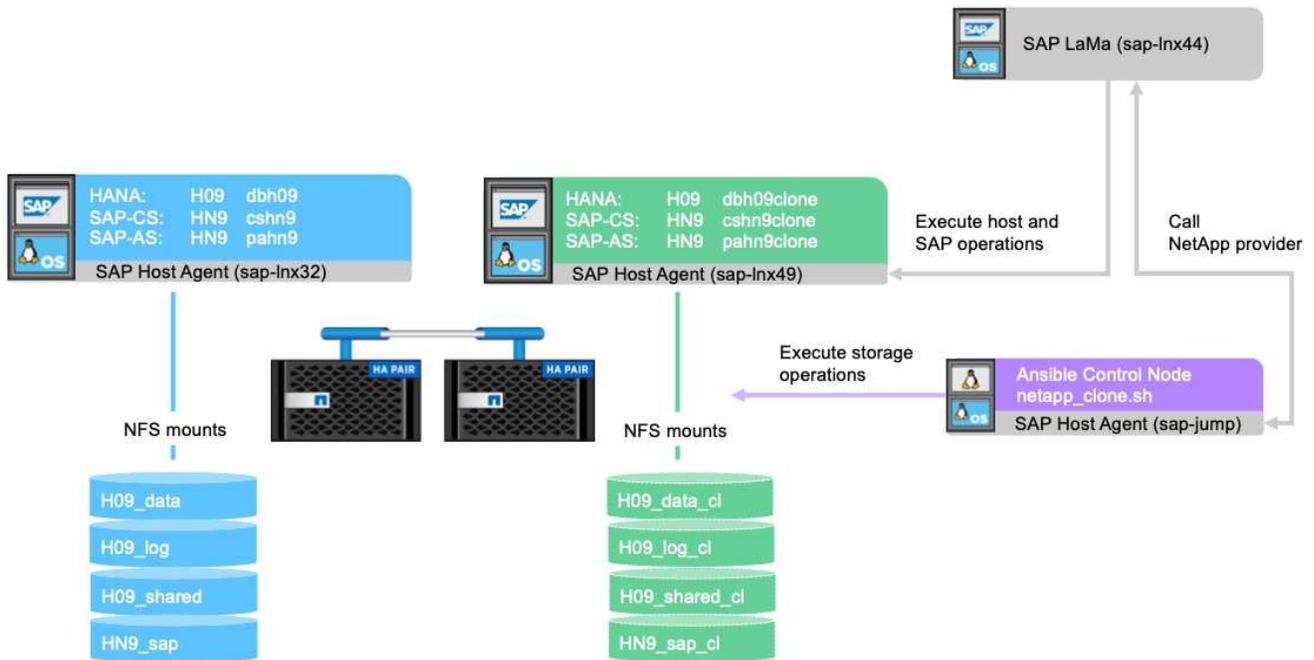
ラボのセットアップ

次の図は、ラボのセットアップを示しています。システムのクローニングに使用されるソースのSAPシステムHN9では、データベースH09、SAP CS、およびSAP ASサービスが、がインストールされた同じホスト（sap-lnx32）で実行されています ["アダプティブデザイン"](#) 有効。に従ってAnsibleコントロールノードが準備されました ["NetApp ONTAP 向けのAnsibleプレイブック"](#) ドキュメント

SAPホストエージェントもこのホストにインストールされています。NetAppプロバイダスクリプトとAnsibleプレイブックは、Ansibleコントロールノードで設定しました（を参照）。"[付録：プロバイダスクリプトの設定](#)"

ホスト sap-lnx49 SAP LaMaクローニング運用のターゲットとして使用し、隔離に対応した機能を設定しました。

システムのコピーおよび更新には、さまざまなSAPシステム（ソースとしてHNA、ターゲットとしてHN2）が使用されました。これは、Post Copy Automation（PCA）が有効になっているためです。



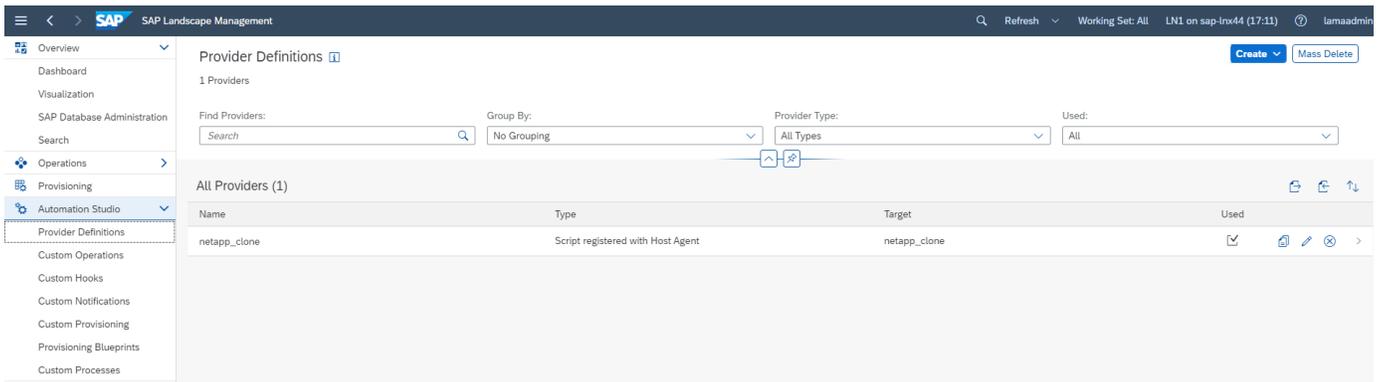
ラボ環境では、次のソフトウェアリリースを使用しました。

- SAP LaMa Enterprise Edition 3.00 SP23_2
- SAP HANA 2.00.052.00.1599235305
- SAP 7.77パッチ27 (S/4 HANA 1909)
- SAPホストエージェント7.22パッチ56
- SAPACEXT 7.22パッチ69
- Linux SLES 15 SP2
- Ansible 2.13.7.
- NetApp ONTAP 9.8P8

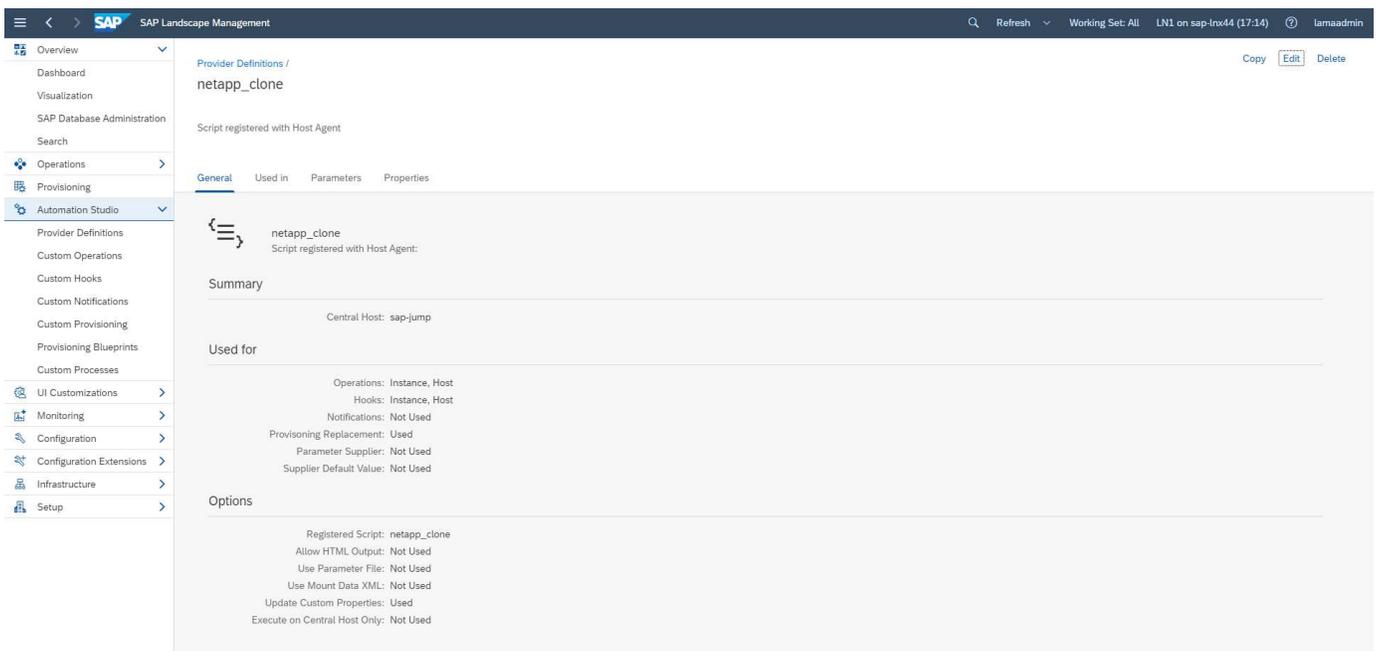
SAP LaMa設定

SAP LaMaプロバイダによる定義

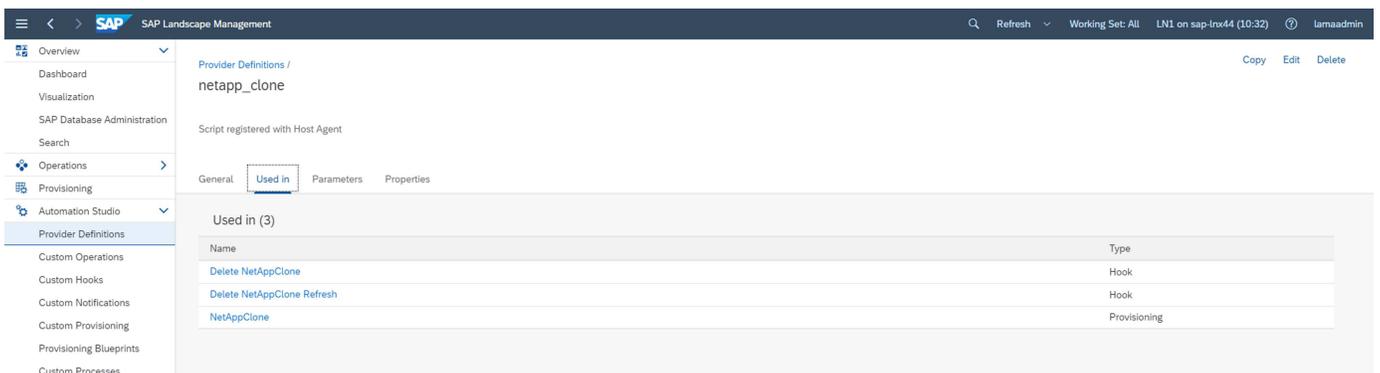
プロバイダの定義は、次のスクリーンショットに示すように、SAP LaMaのAutomation Studio内で実行されます。この例の実装では、前に説明したように、異なるカスタムプロビジョニングステップと操作フックに使用される単一のプロバイダ定義を使用します。



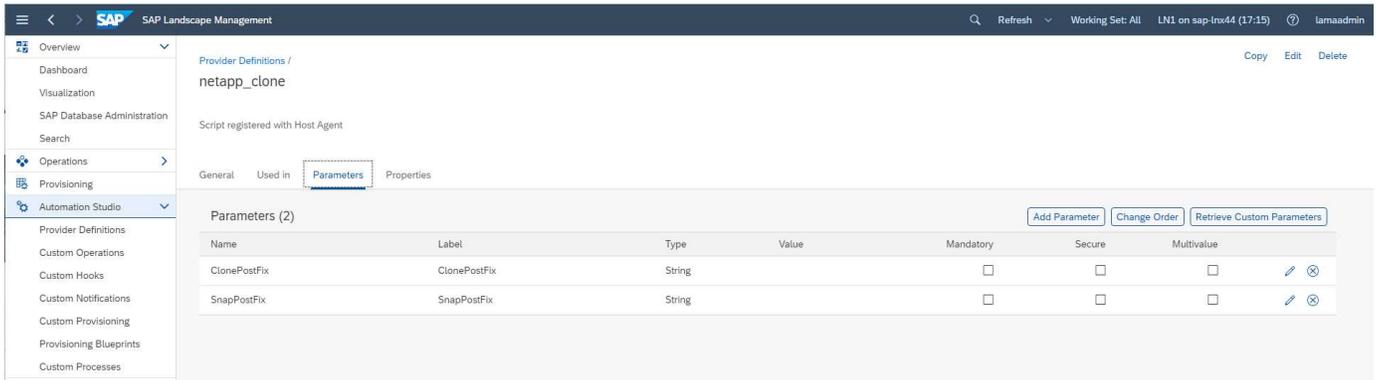
プロバイダ netapp_clone スクリプトとして定義されます netapp_clone.sh SAPホストエージェントに登録されている。SAPホストエージェントは、中央のホストで実行されます `sap-jump` Ansibleコントロールノードとしても機能します。



[で使用 (Used in)]タブには、プロバイダーが使用されるカスタムオペレーションが表示されます。カスタムプロビジョニング NetAppClone およびカスタムフック Delete NetAppClone および Delete NetAppClone Refresh *の構成は、次の章で説明します。

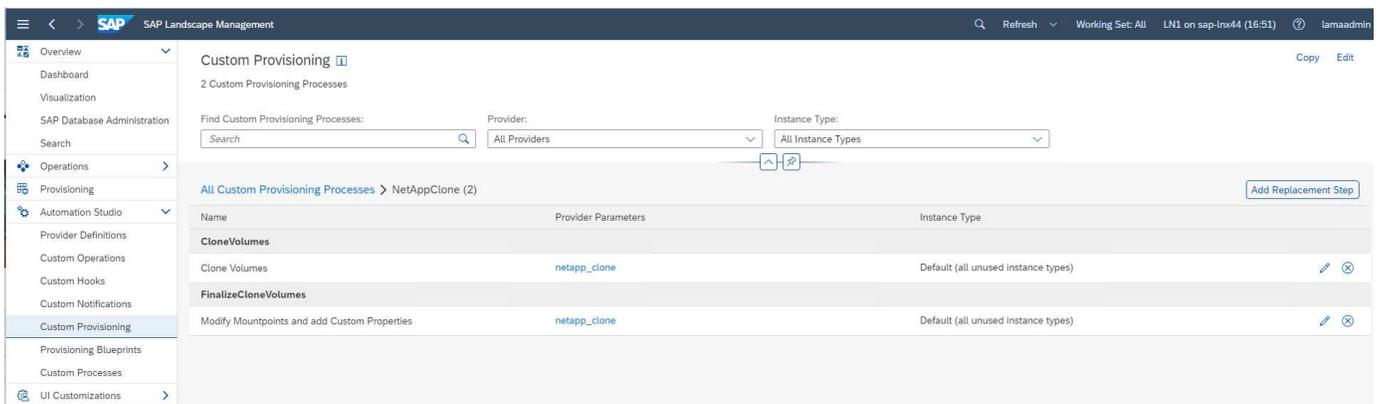


プロビジョニングワークフローの実行中にパラメータ* ClonePostFix および SnapPostFix *が要求され、SnapshotとFlexCloneボリュームの名前に使用されます。



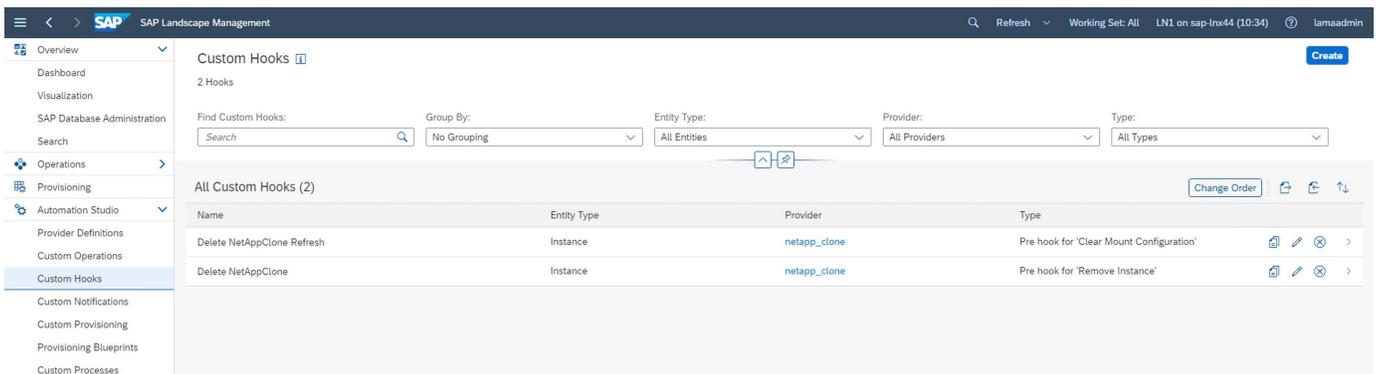
SAP LaMaカスタムプロビジョニング

SAP LaMaのカスタムプロビジョニング設定で、前述したお客様のプロバイダを使用して、プロビジョニングワークフローの手順* Clone Volumes と PostCloneVolumes *を置き換えます。



SAP LaMaカスタムフック

システムの削除ワークフロー時にシステムを削除した場合は、フック* Delete NetAppClone を使用してプロバイダ定義が呼び出されます **netapp_clone**。インスタンスは実行中も保持されるため、システムの更新ワークフロー中は Delete NetApp Clone Refresh *フックが使用されます。



SAP LaMaがマウントポイント設定の情報をプロバイダに提供できるように、カスタムフックに「データマウントXML *を使用」を設定することが重要です。

The screenshot shows the SAP Landscape Management interface for a custom hook named 'Delete NetAppClone'. The 'Additional Information' section is highlighted with a red box, showing the following settings:

- Use Mount Data XML: Yes
- Parallel Execution: No
- Background Step: No
- Process Error Hook: No
- Is System Wide Hook: No
- Retrieve Secure Parameters: No

カスタムプロビジョニングワークフローを使用してシステムを作成したときにのみカスタムフックが使用および実行されるようにするために、次の制約がそのフックに追加されます。

The screenshot shows the 'Constraints' section for the custom hook 'Delete NetAppClone'. The constraint table is as follows:

Name	Operator	Value
Custom clone process name (Static)	=	NetAppClone

カスタムフックの使用方法の詳細については、を参照してください ["SAP LaMaのドキュメント"](#)。

SAPソースシステムでカスタムのプロビジョニングワークフローを有効にします

ソースシステムのカスタムプロビジョニングワークフローを有効にするには、その設定を該当するものにする必要があります。[カスタムプロビジョニングプロセスを使用する*]チェックボックスをオンにして、対応するカスタムプロビジョニング定義を選択する必要があります。

SAP Landscape Management

Working Set: <AB> Search: [] Go LN1 on sap-inx4

Automation Studio Configuration Infrastructure

Pools Systems Hosts Characteristics

Overview of Systems and Instances

Discover Remove Instance and System Reassign Instances Mass Configuration Filtering Export Import

Name	Managed	AC-Enabled	Operational	Pool	Network	Description
HN9: NetWeaver ABAP 7.77, cshn9	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MUCCBC		
• System database: MASTER (configured) : H09, SAP HANA 02, dbh09	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MUCCBC	MUCCBC-SAP-Front	
• Central services: 01, cshn9	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MUCCBC	MUCCBC-SAP-Front	
• AS instance: 00, pahn9	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MUCCBC	MUCCBC-SAP-Front	
▶ HNA: NetWeaver ABAP 7.77, cshna	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	MUCCBC		

Systems: 2 Selected: HN9: NetWeaver ABAP 7.77, cshn9

System Details Log

Edit Show In

General

System Name: HN9: NetWeaver ABAP 7.77, cshn9

SID: HN9

Instance ID: SystemID:HN9.SystemHost:cshn9

Solution Manager settings

Assign Solution Manager System:

Focused Run Settings

Assign Focused Run System:

Disable Workmode Management:

System and AS Provisioning

This system was provided by:

This system can be used for:

Installation

Cloning Application Server (Un-Installation)

Copying Diagnostic Agent (Un-Installation)

Renaming nZDM Java

Standalone PCA Replication Configuration

Use Custom Provisioning Process:

NetAppClone

Use as TDMS Control System:

Is B/W Source System:

Use Replication for Single Tenant Database Refresh:

Intersystem Dependencies

From Instance To Instance

• [0] Outgoing (0)

• [0] Incoming (0)

Entity Relations

Custom Relation Type Target Entity Type Target Entity

[1] Table is empty

E-Mail Notification

Enable Email Notification:

Custom Notification

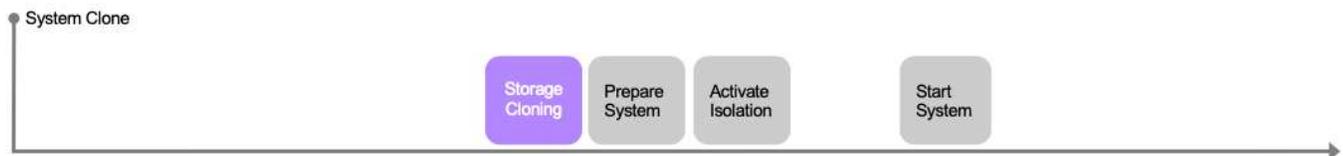
Enable Custom Notification:

ACM Settings

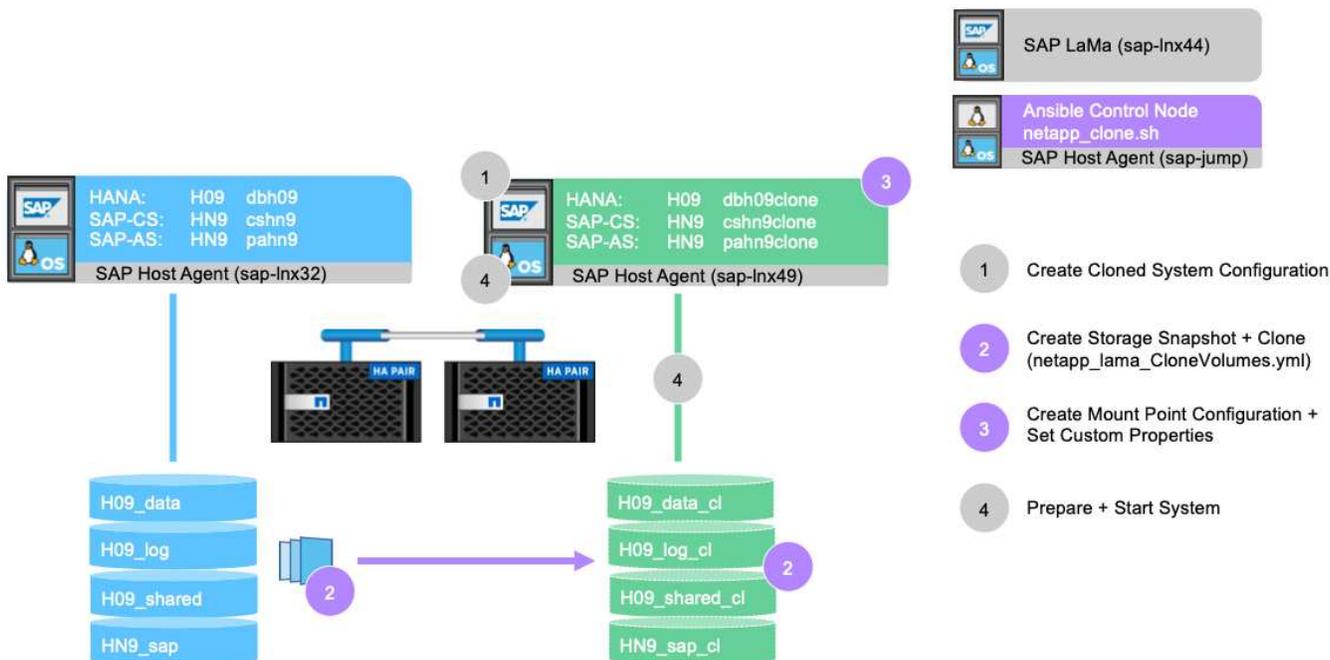
ACM-Managed:

SAP LaMa プロビジョニングワークフロー-クローンシステム

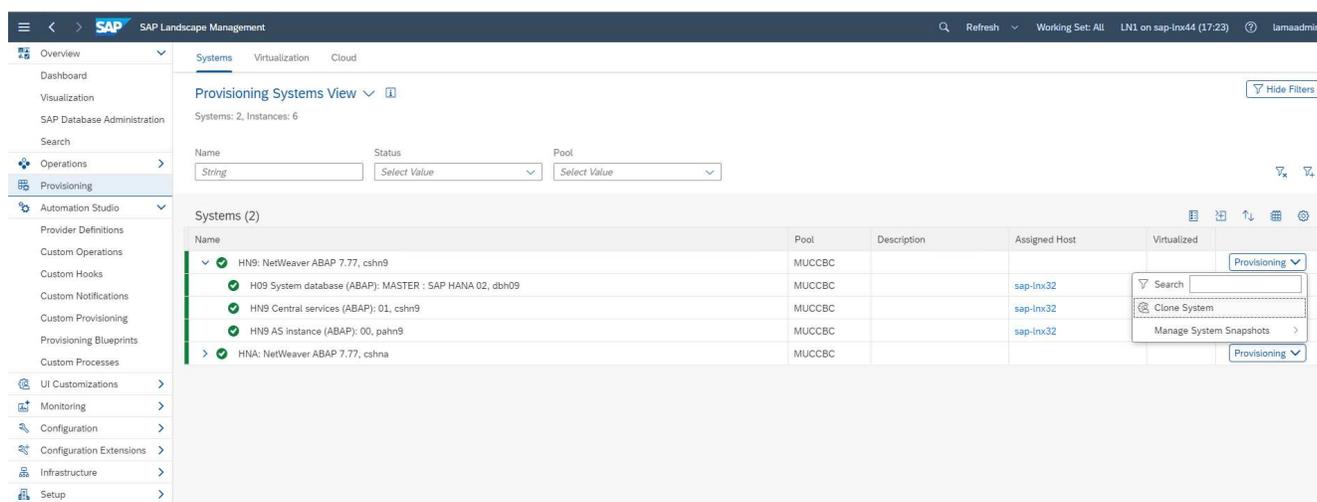
次の図は、システムのクローニングワークフローで実行する主な手順を示しています。



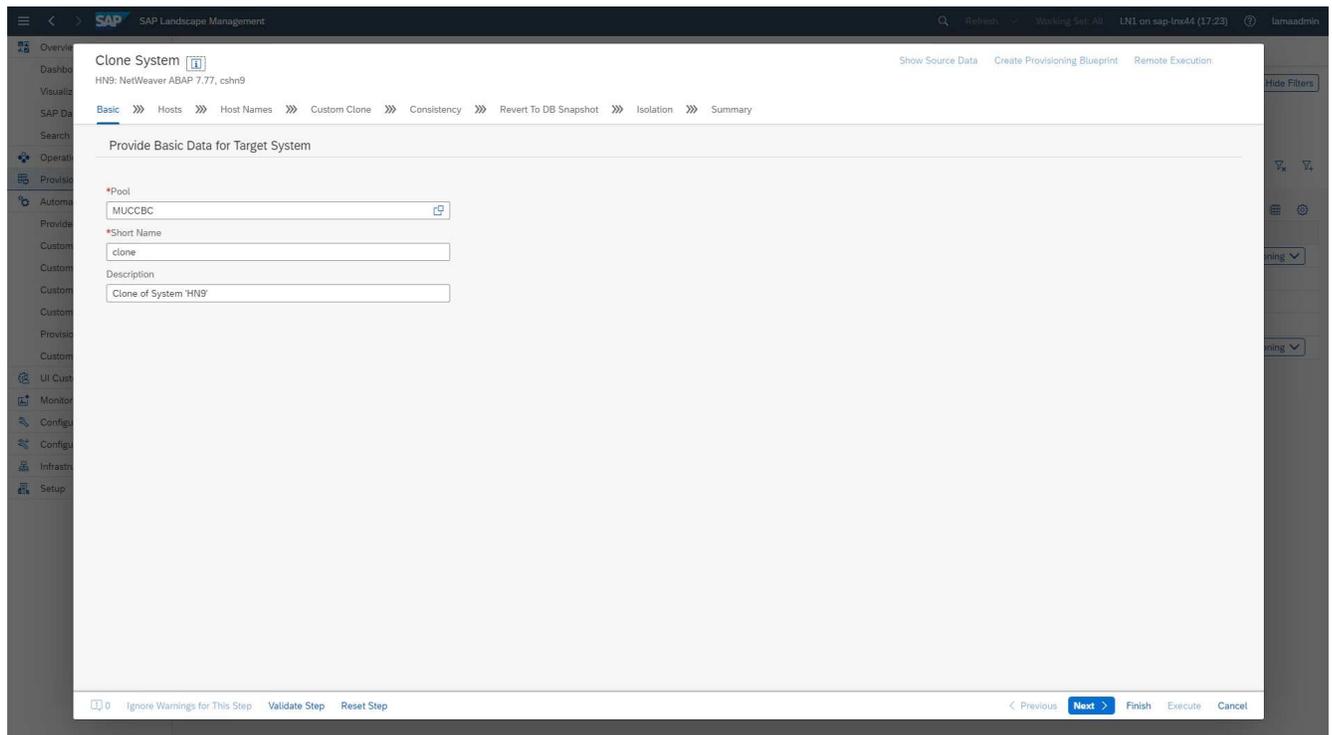
このセクションでは、HANAデータベースH09を使用するソースのSAPシステムHN9に基づく、SAP LaMaシステムの完全なクローニングワークフローを紹介します。次の図は、ワークフロー中に実行される手順の概要を示しています。



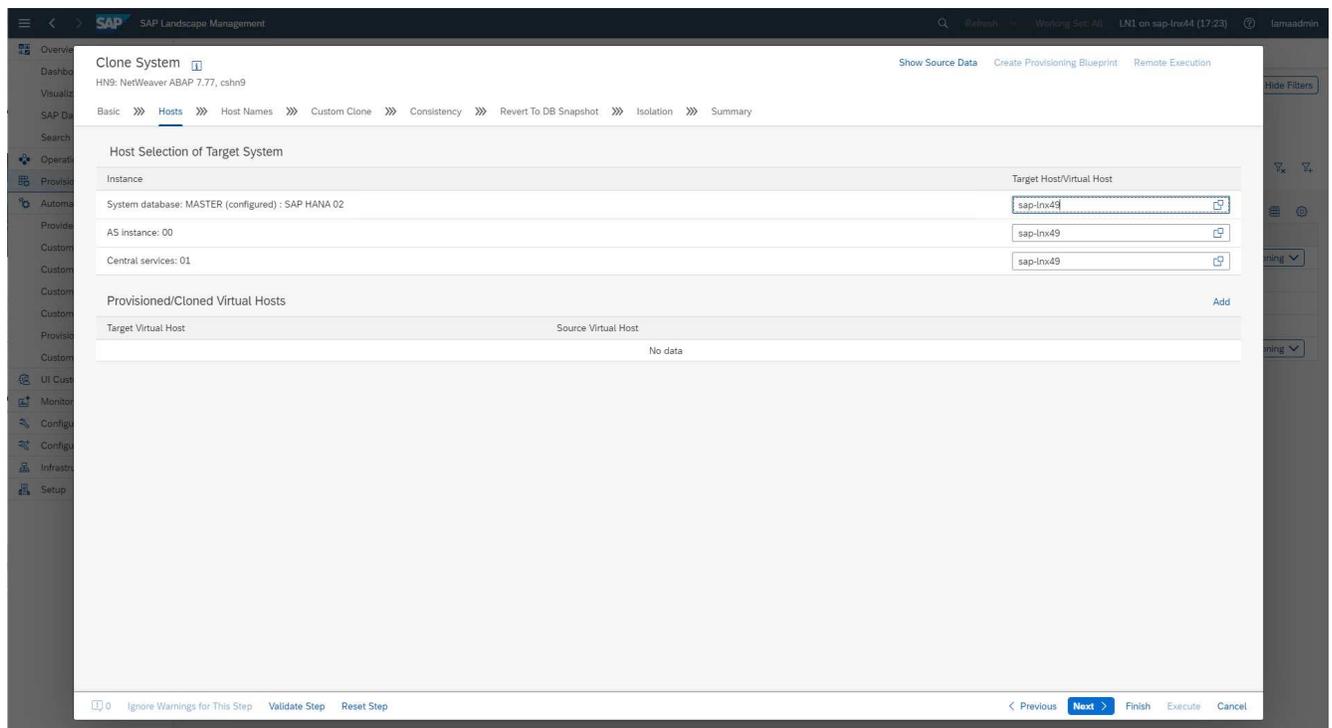
1. クローニングワークフローを開始するには、メニューツリーで* Provisioning を開き、ソースシステム（この例ではHN9）を選択します。次に、Clone System *ウィザードを起動します。



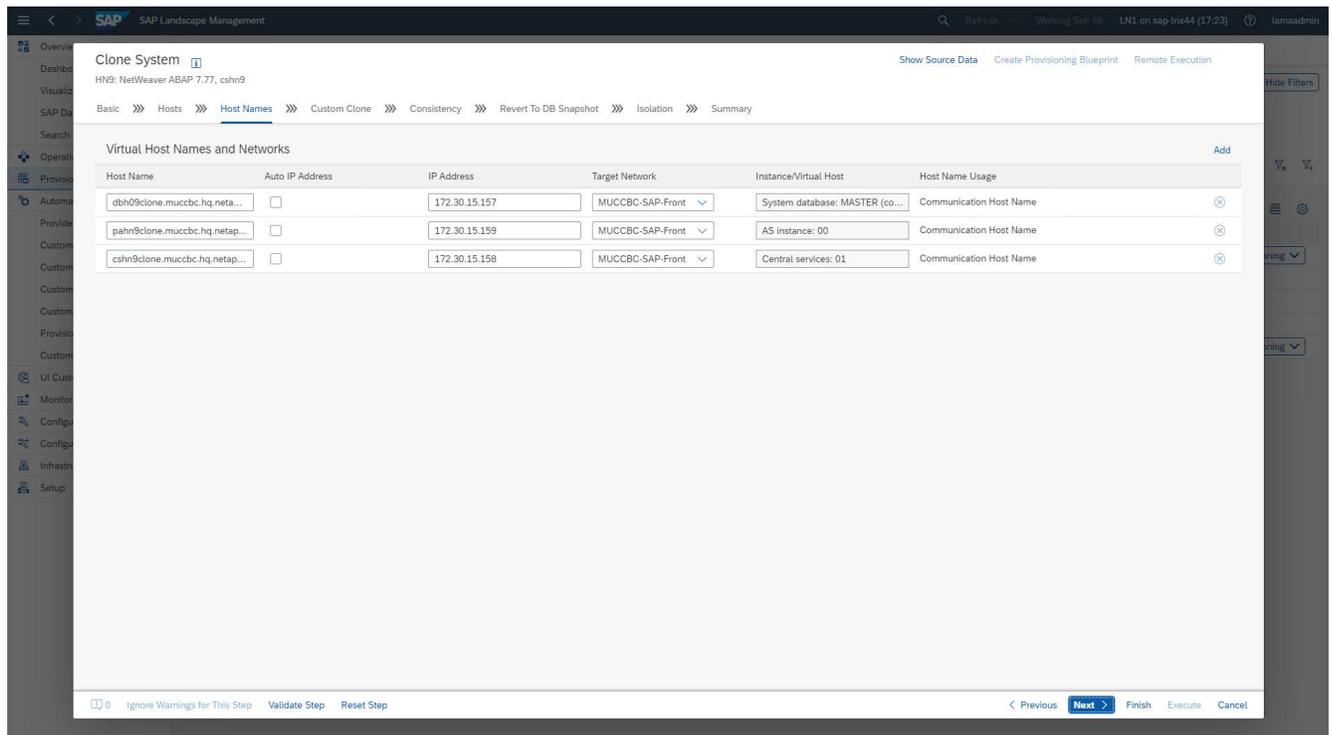
2. 必要な値を入力します。ウィザードの画面1で、クローニングしたシステムのプール名を入力するように求められます。このステップでは、クローニングされたシステムを起動するインスタンス（仮想または物理）を指定します。デフォルトでは、システムはターゲットシステムと同じプールにクローニングされます。



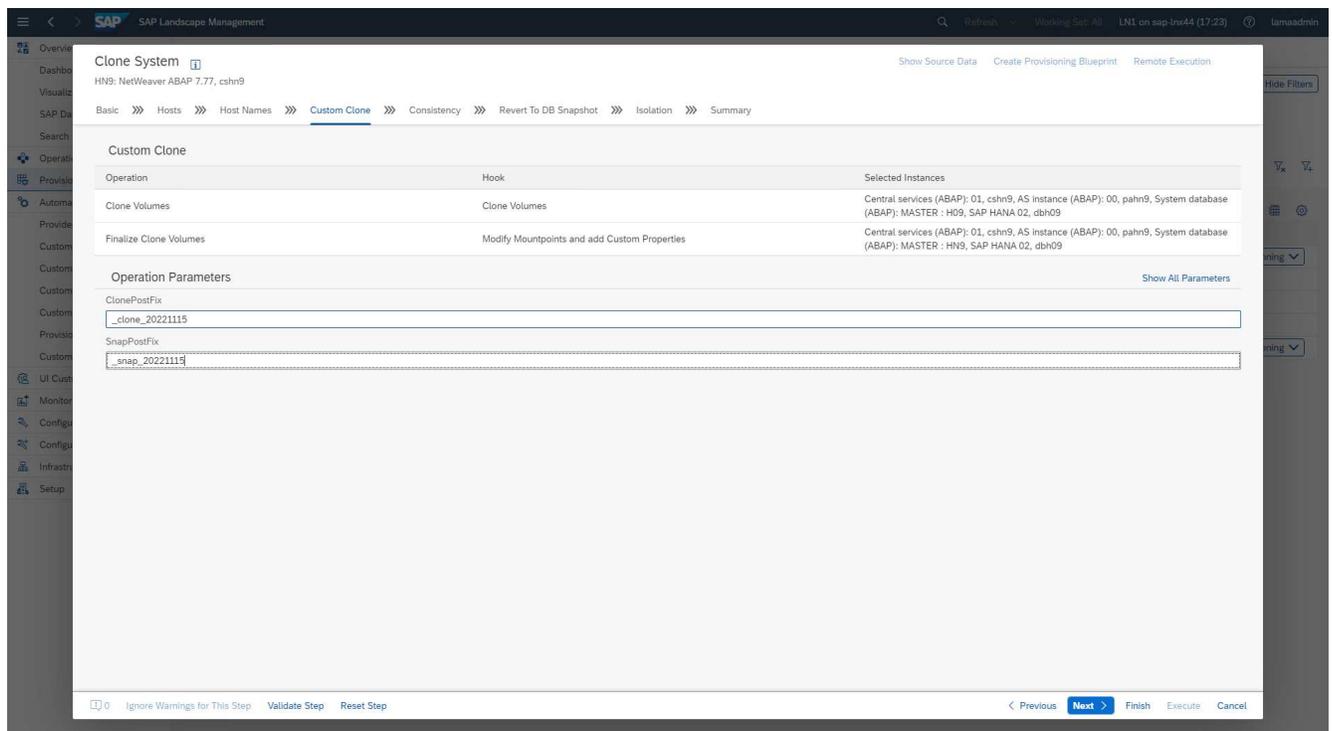
3. ウィザードの画面2で、新しいSAPインスタンスを起動するターゲットホストを指定するよう求められます。このインスタンスのターゲットホストは、前の画面で指定したホストプールから選択できます。各インスタンスまたはサービスは、別々のホストで開始できます。この例では、3つのサービスがすべて同じホスト上で実行されています。



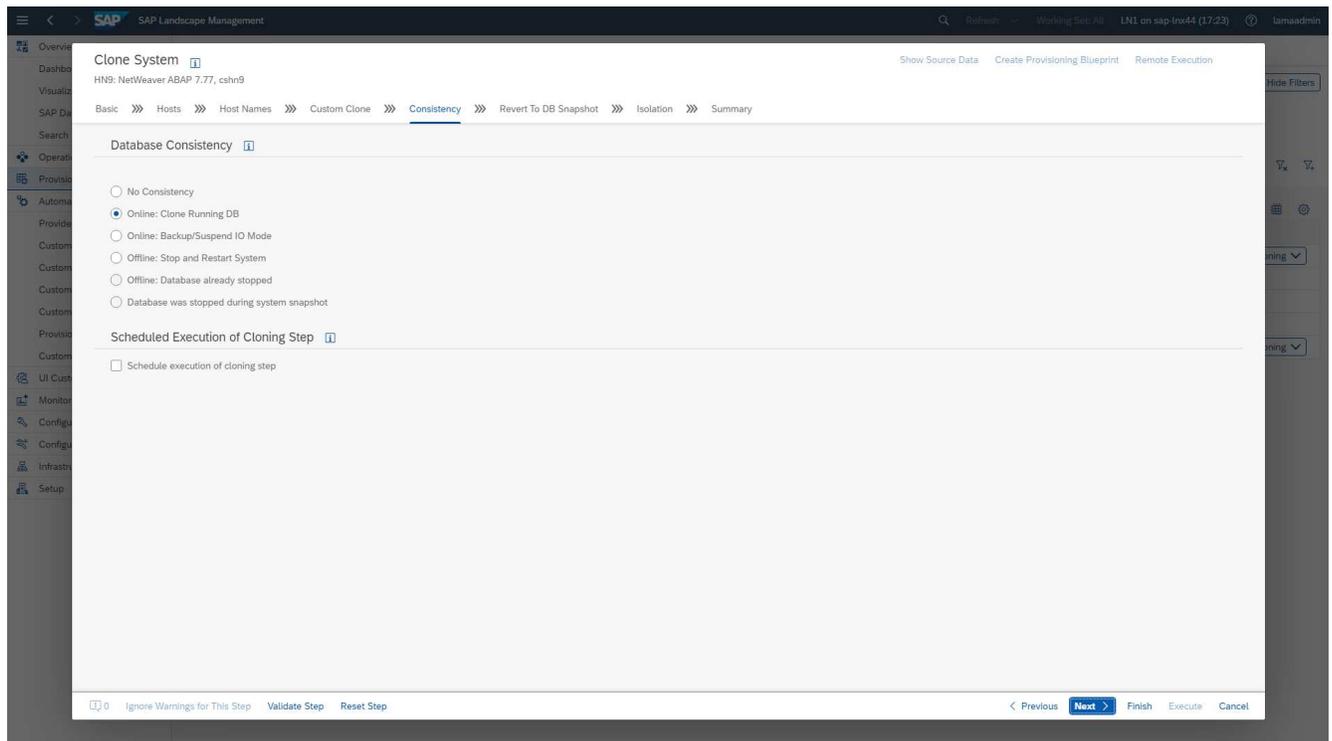
4. 画面3で要求された情報を入力します。この情報は、仮想ホスト名とネットワークを要求します。通常、ホスト名はDNSで管理されるため、IPアドレスもそれに応じて事前に入力されます。



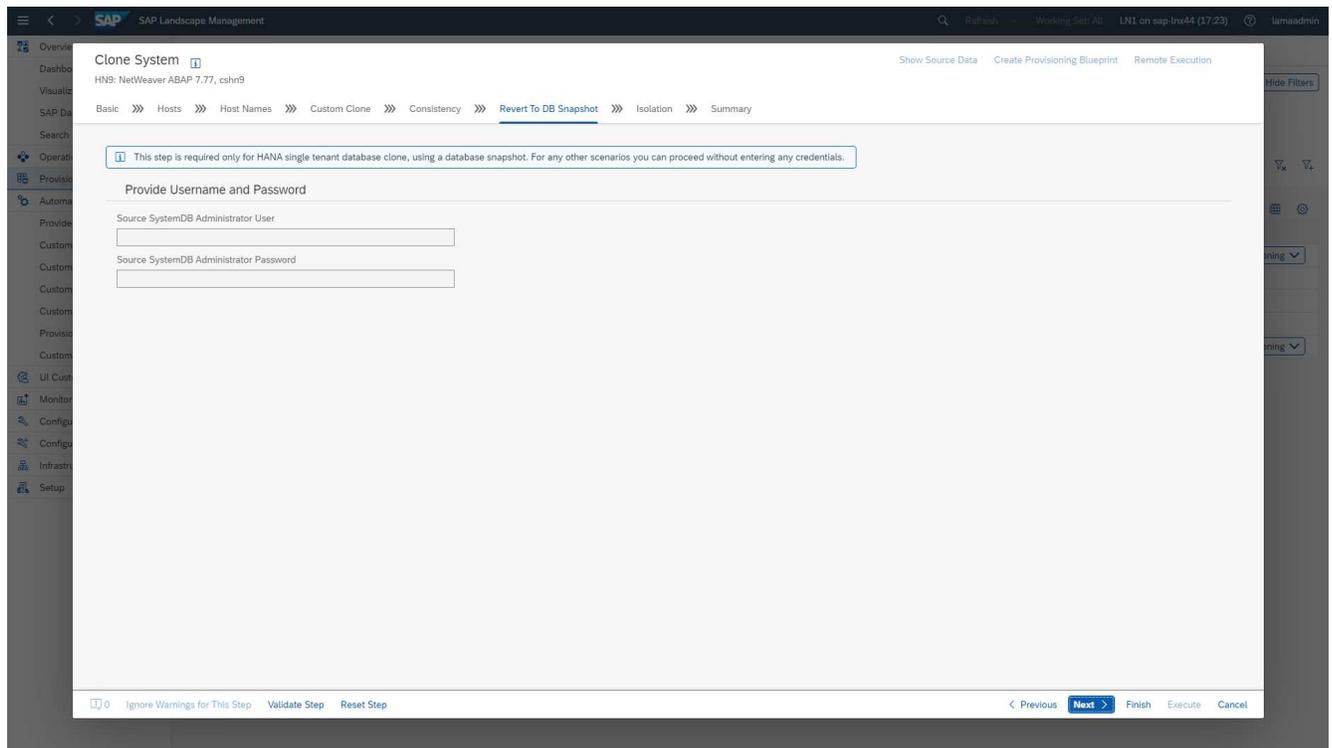
5. 画面4に、カスタムのクローン処理が表示されます。クローン名と* SnapPostfix *名を指定した場合、この名前は、FlexCloneボリュームとSnapshot名に対するストレージのクローン処理でそれぞれ使用されます。これらのフィールドを空のままにしておくと、プロバイダスクリプトの変数セクションで設定されたデフォルト値になります netapp_clone.sh を使用します。



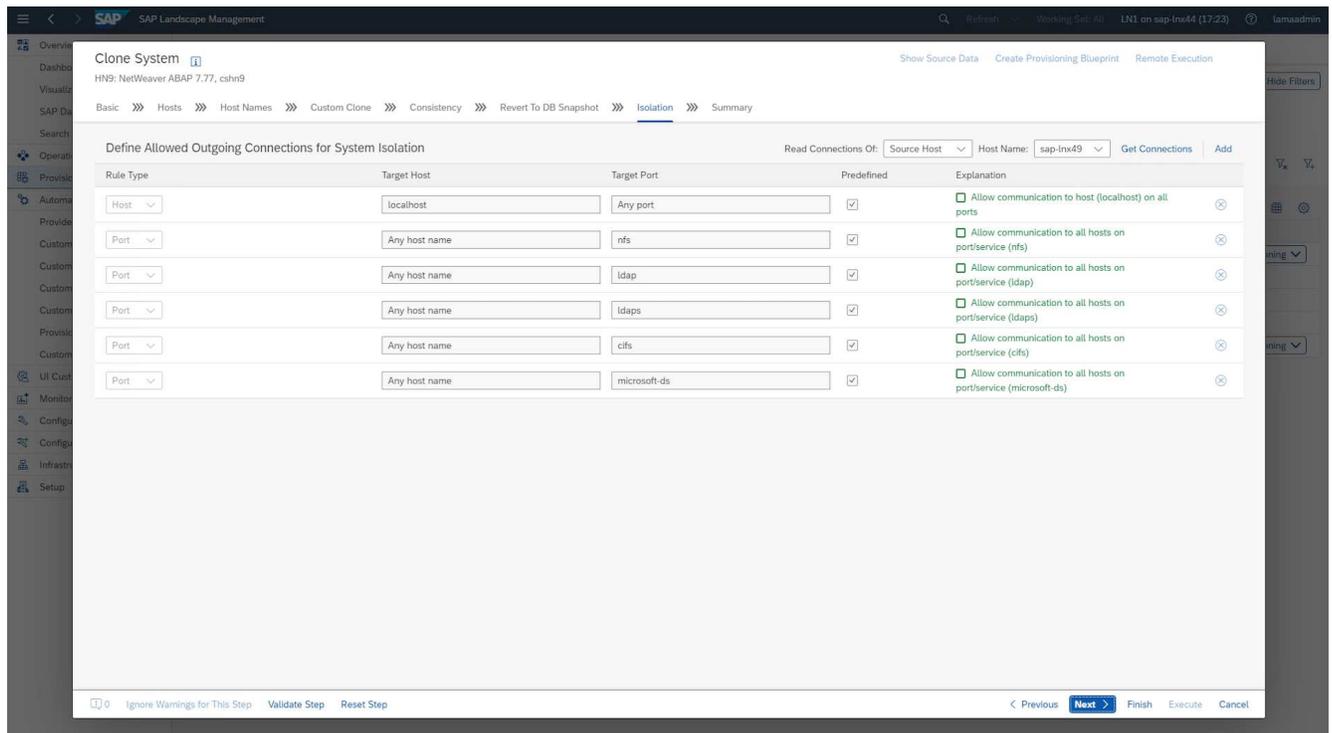
6. 画面5では、データベース整合性オプションが選択されています。この例では、「* Online : Clone running DB *」を選択しました。



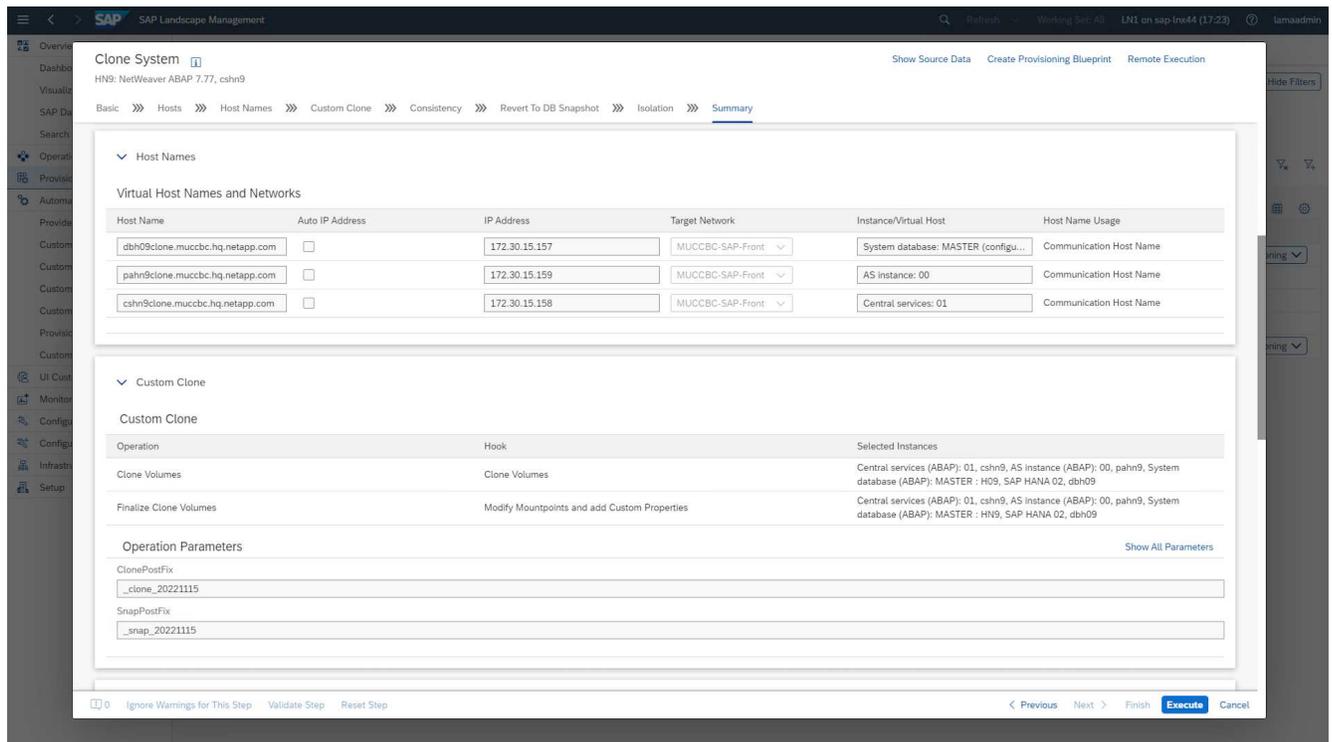
7. 画面6では、テナントのクローンを実行する場合にのみ入力する必要があります。



8. 画面7では、システムの隔離を構成できます。

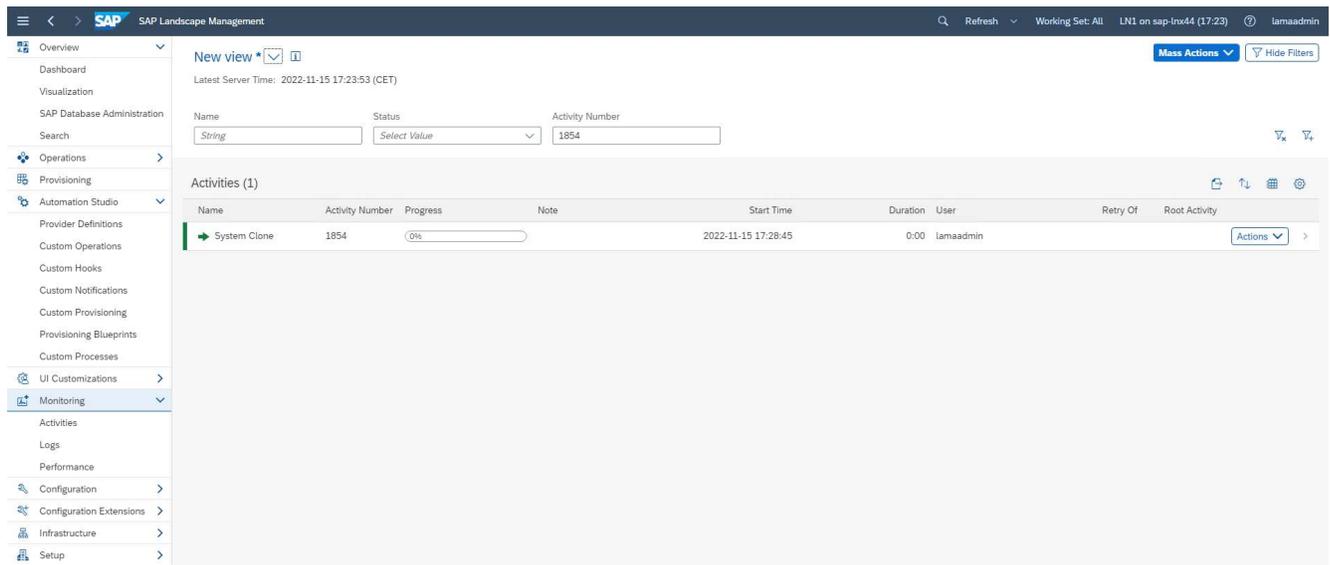


9. 画面8では、ワークフローを開始する前の最終確認のためのすべての設定が概要ページに表示されます。**[Execute]**をクリックして'ワークフローを開始します'

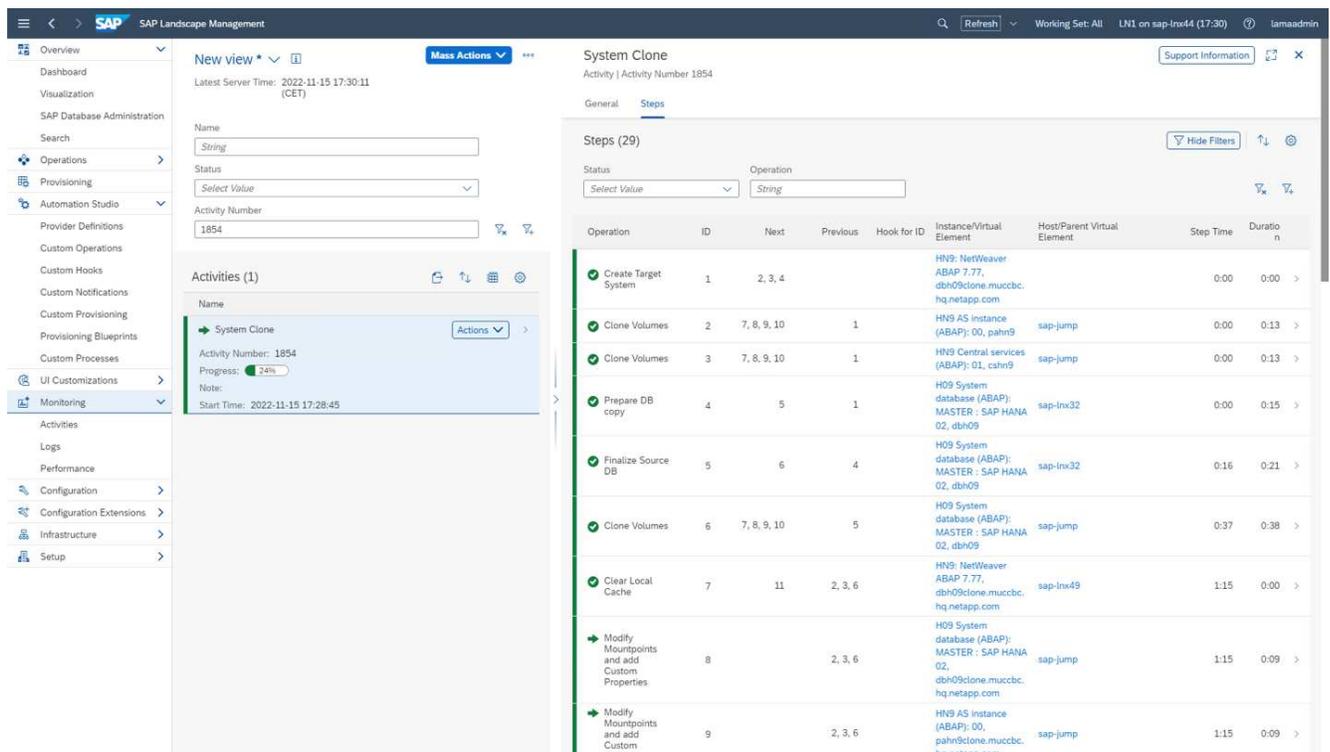


SAP LaMaが、設定に記載されたすべてのアクションを実行します。具体的には、ストレージボリュームのクローンとエクスポートの作成、ターゲットホストへのマウント、分離のためのファイアウォールルールの追加、HANAデータベースとSAPサービスの開始です。

10. クローンワークフローの進行状況は、* Monitoring *メニューで監視できます。



詳細ログには、* Clone Volume および ModifyマウントポイントとAdd Custom Properties *の処理がAnsibleノード、で実行されます sap-jump ホスト：以下の手順は、各サービス、HANAデータベース、SAP中央サービス、およびSAPサービスに対して実行します。



- Clone Volumes *タスクを選択すると、そのステップの詳細ログが表示され、Ansible Playbookの実行状況が表示されます。Ansibleプレイブックも確認できます netapp_lama CloneVolumes.yml は、HANAデータベースのボリューム、データ、ログ、および共有ごとに実行されます。

The screenshot displays the SAP Landscape Management interface during a 'System Clone' activity. The 'Clone Volumes' step is active, and the 'Messages' pane shows the following log entries:

- DEBUG | ID: 59 | Message Code: OSP-0200
Time: 2022-11-15 17:30:01 | Entry Time: 0:38
Operation succeeded
- DEBUG | ID: 58 | Message Code: TMP-1001
Time: 2022-11-15 17:30:01 | Entry Time: 0:38
Temp File /tmp/VCMIsaplmg9MDYH removed
- DEBUG | ID: 57 | Message Code: FWD-0003
Time: 2022-11-15 17:30:01 | Entry Time: 0:38
No valid sapacext not found. Request will be handled by sapacosprep. See log for further details
- DEBUG | ID: 56 | Message Code: HALog
Time: 2022-11-15 17:30:01 | Entry Time: 0:38
Download logfile /usr/sap/hostctrl/work/ASUJ.log from host sap-jump
- DEBUG | ID: 55 | Message Code: LVM
Time: 2022-11-15 17:30:01 | Entry Time: 0:38
Removing temp file /tmp/VCMIsaplmg9MDYHnow
- DEBUG | ID: 39 | Message Code: NetApp Clone for Custom Provis**
Time: 2022-11-15 17:29:40 | Entry Time: 0:17
Running ansible playbook netapp_lama_CloneVolumes.yml on Volume H09_shared
- DEBUG | ID: 31 | Message Code: NetApp Clone for Custom Provis
Time: 2022-11-15 17:29:40 | Entry Time: 0:17
Running ansible playbook netapp_lama_CloneVolumes.yml on Volume H09_log
- DEBUG | ID: 23 | Message Code: NetApp Clone for Custom Provis
Time: 2022-11-15 17:29:40 | Entry Time: 0:17
Running ansible playbook netapp_lama_CloneVolumes.yml on Volume H09_data
- DEBUG | ID: 22 | Message Code: NetApp Clone for Custom Provis
Time: 2022-11-15 17:29:40 | Entry Time: 0:17
saving mount config...
- DEBUG | ID: 21 | Message Code: NetApp Clone for Custom Provis
Time: 2022-11-15 17:29:40 | Entry Time: 0:17
netapp_clone.sh --HookOperationName=CloneVolumes --SAPSYSTEMNAME=H09 --SAPSYSTEM=02 --MOUNT_XML_PATH=/tmp/VCMIsaplmg9MDYH --PARAM_ClonePostFix=_clone_20221115 --PARAM_SnapPostFix=_snap_20221115 --PROP_ClonePostFix --PROP_SnapPostFix --SAP_LVM_SRC_SID=HN9 --SAP_LVM_TARGET_SID=HN9
- DEBUG | ID: 20 | Message Code: NetApp Clone for Custom Provis
Time: 2022-11-15 17:29:40 | Entry Time: 0:17
Running Script netapp_clone.sh Version 0.9

12. マウントポイントの変更とカスタムプロパティの追加の手順*マウントポイントの詳細ビューには、マウントポイントと、実行スクリプトによって渡されたカスタムプロパティに関する情報が表示されます。

The screenshot displays the SAP Landscape Management interface during a 'Modify Mountpoints and add Custom Properties' activity. The 'Messages' pane shows the following log entries:

- DEBUG | ID: 40 | Message Code: LVM
Time: 2022-11-15 17:30:31 | Entry Time: 0:30
Updates Persisted
- DEBUG | ID: 39 | Message Code: LVM
Time: 2022-11-15 17:30:31 | Entry Time: 0:30
Updating SystemID
- RESULT | ID: 24 | Message Code: NetApp Clone for Custom Provis**
Time: 2022-11-15 17:30:20 | Entry Time: 0:18
Got new property SnapPostFix=_snap_20221115
- RESULT | ID: 23 | Message Code: NetApp Clone for Custom Provis**
Time: 2022-11-15 17:30:20 | Entry Time: 0:18
Got new property ClonePostFix=_clone_20221115
- DEBUG | ID: 22 | Message Code: NetApp Clone for Custom Provis
Time: 2022-11-15 17:30:20 | Entry Time: 0:18
netapp_clone.sh --HookOperationName=FinalizeCloneVolumes --SAPSYSTEMNAME=HN9 --SAPSYSTEM=01 --MOUNT_XML_PATH=/tmp/VCMIsaplmg9MDYH --PARAM_ClonePostFix=_clone_20221115 --PARAM_SnapPostFix=_snap_20221115 --PROP_ClonePostFix --PROP_SnapPostFix --SAP_LVM_SRC_SID=HN9 --SAP_LVM_TARGET_SID=HN9
- DEBUG | ID: 21 | Message Code: NetApp Clone for Custom Provis
Time: 2022-11-15 17:30:20 | Entry Time: 0:18
Running Script netapp_clone.sh Version 0.9
- DEBUG | ID: 12 | Message Code: LVM
Time: 2022-11-15 17:30:01 | Entry Time: 0:00
Retrieved the following parameters from hostagent (name: ClonePostFix, is a CustomProperty, name: ClonePostFix, is a CustomParameter, name: SnapPostFix, is a CustomProperty, name: MOUNT_XML_PATH, name: SAPSYSTEMNAME, name: HookOperationName, name: SnapPostFix, is a CustomParameter, name: SAP_LVM_SRC_SID, name: SAP_LVM_TARGET_SID, name: SAPSYSTEM)
- DEBUG | ID: 10 | Message Code: LVM
Time: 2022-11-15 17:30:01 | Entry Time: 0:00
Updating logs
- DEBUG | ID: 9 | Message Code: LVM
Time: 2022-11-15 17:30:01 | Entry Time: 0:00
Generic Transferred Parameters:
CustomOpld: 'e0c689cc-6017-11ed-c90e-0000007e9672'
HookOperationName: 'FinalizeCloneVolumes'
HostName: 'sap-Inv49'
Previous Service ID: 'SystemID.HN9.Number.01.InstanceHost.cshn9'
ServiceId: 'SystemID.HN9.Number.01.InstanceHost.cshn9clone.muccbc.hq.netapp.com'
srcServiceId: 'SystemID.HN9.Number.01.InstanceHost.cshn9'

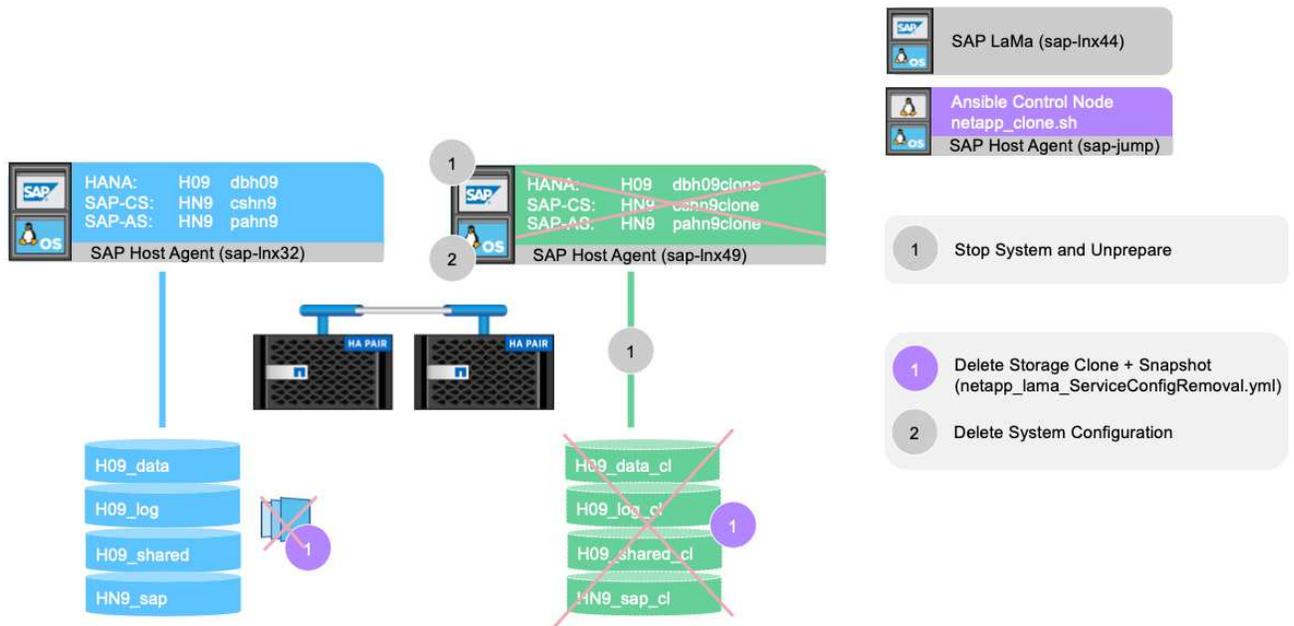
ワークフローが完了すると、クローニングしたSAPシステムの準備が完了し、使用できる状態になります。

SAP LaMaによるワークフローのプロビジョニング解除-システムの破壊

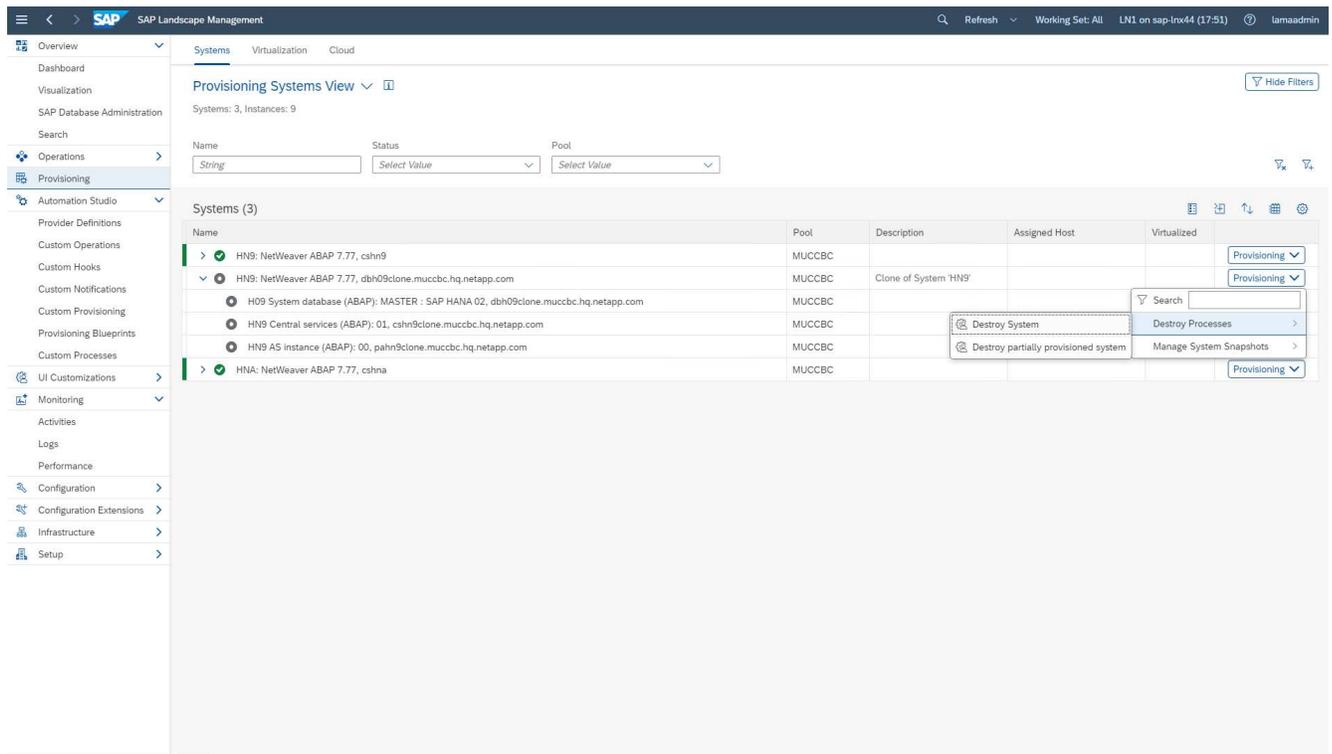
次の図は、system destroyワークフローで実行する主な手順を示しています。



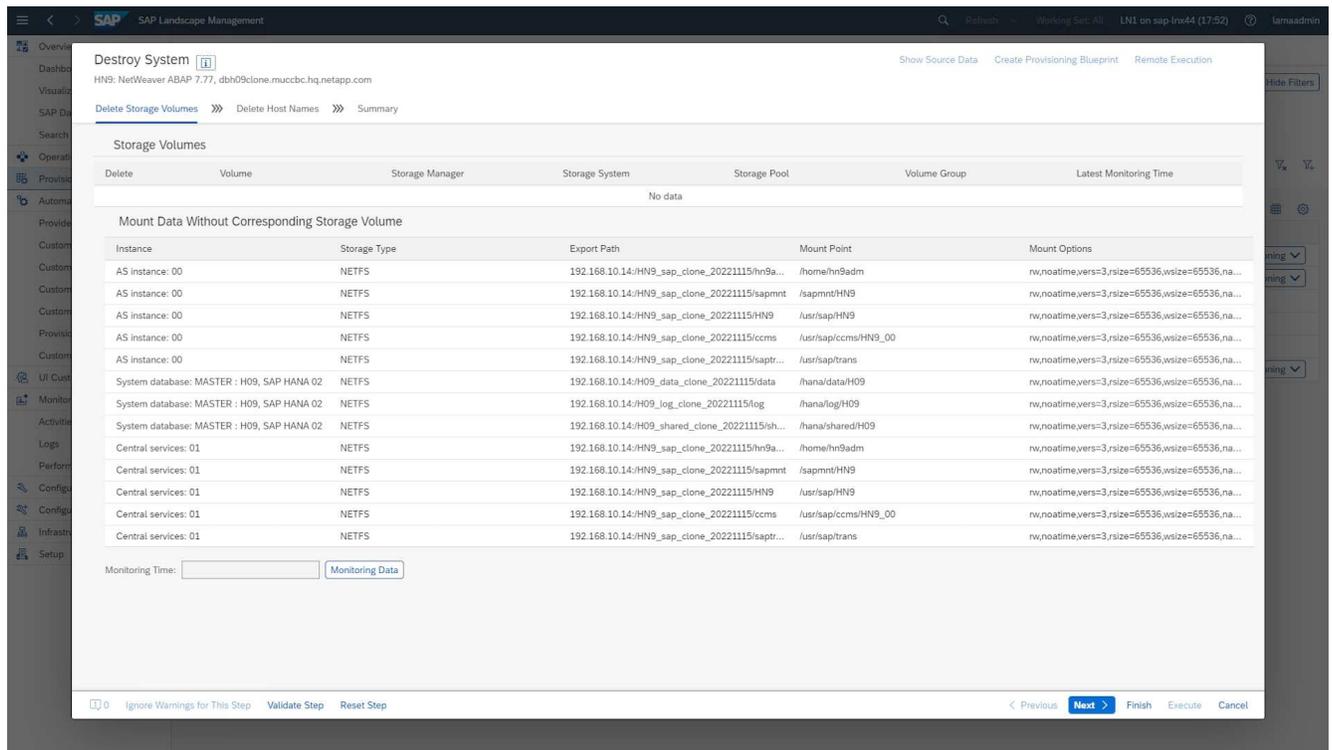
1. クローンシステムの運用を停止するには、運用を停止し、準備を事前に完了しておく必要があります。その後、システムの破壊ワークフローを開始できます。



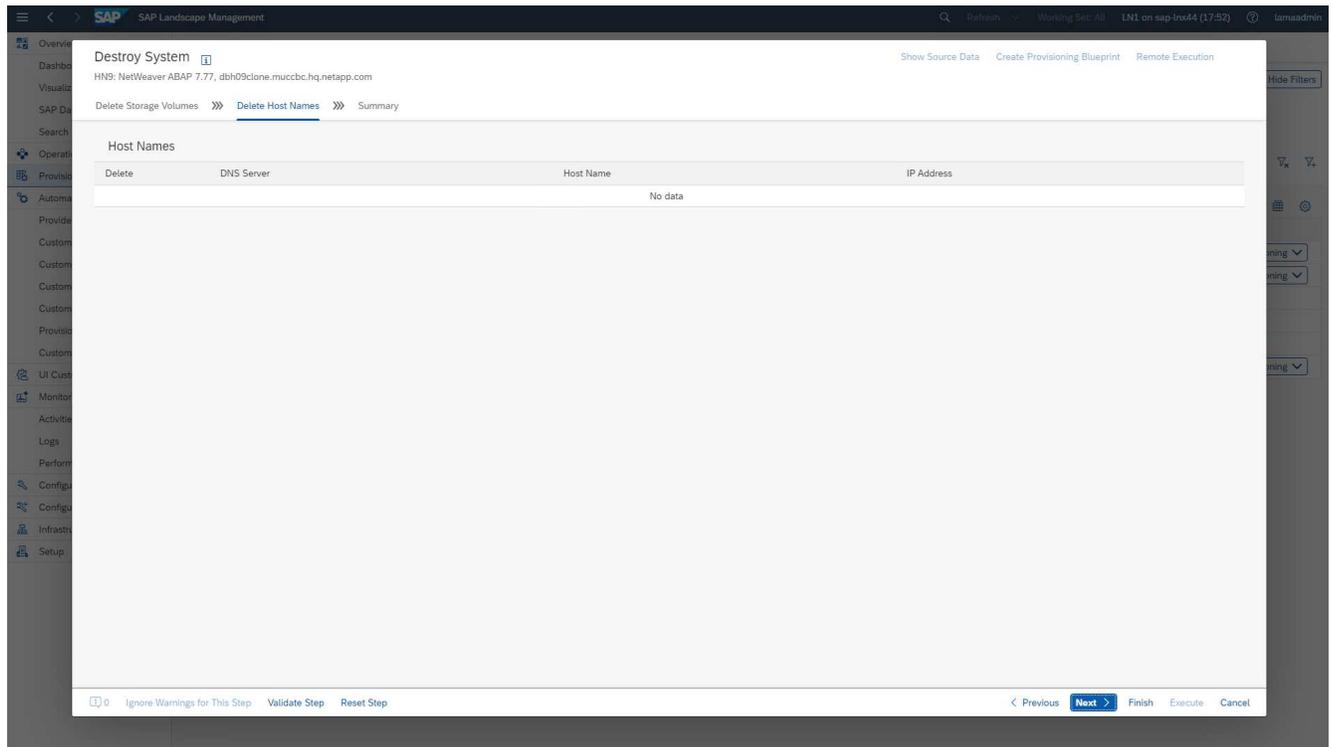
2. この例では、前に作成したシステムに対してsystem destroyワークフローを実行します。システムビュー*画面でシステムを選択し、*破壊プロセス*でシステム破壊ワークフローを開始します。



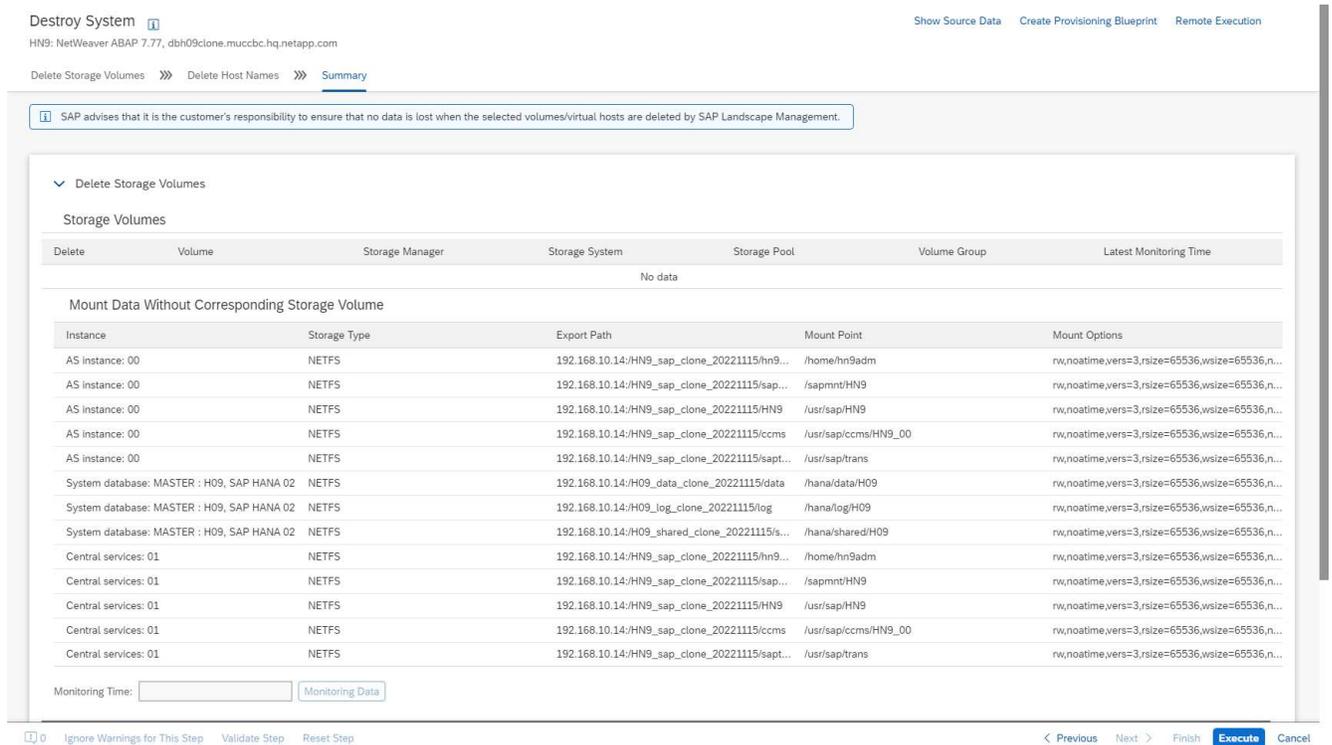
3. プロビジョニングフェーズで維持されたすべてのマウントポイントが表示され、システムの削除ワークフロープロセス中に削除されます。



仮想ホスト名はDNSを通じて管理され、自動的に割り当てられているため、削除されません。

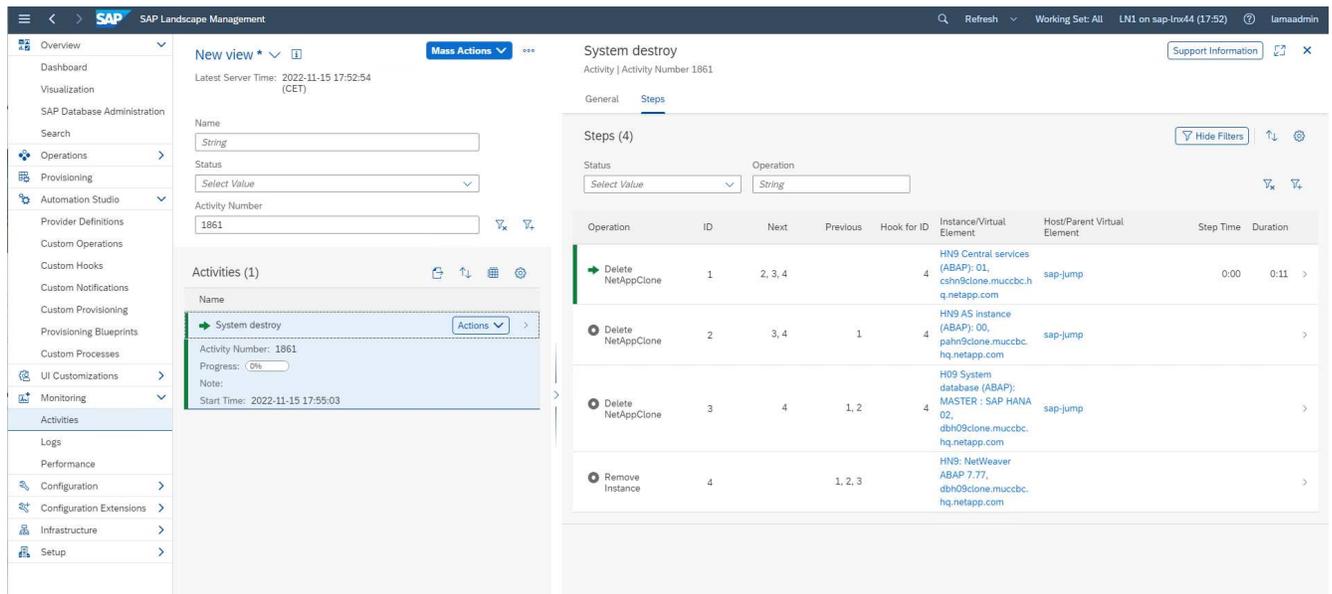


4. 実行ボタンをクリックすると、処理が開始されます。

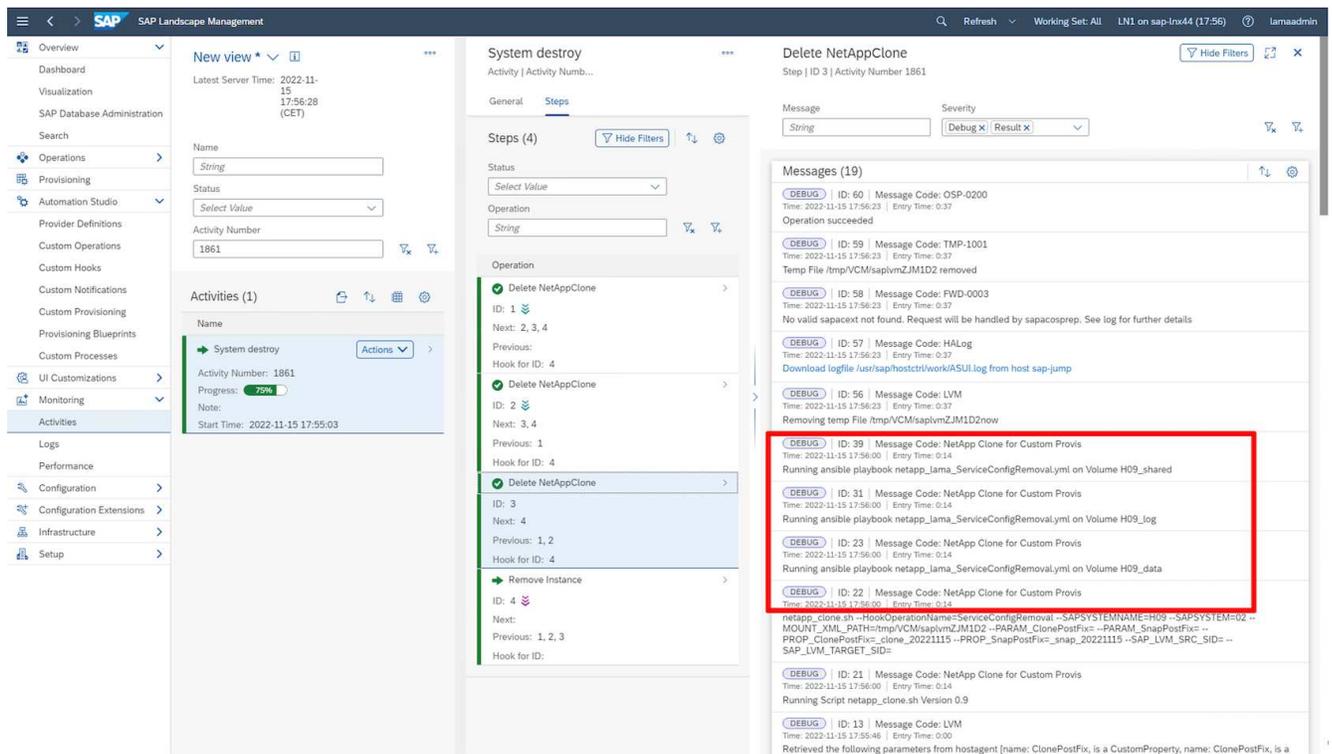


ボリュームクローンを削除し、そのクローンシステムの設定を削除する処理がSAP LaMaから実行されます。

5. クローンワークフローの進行状況は、* Monitoring *メニューで監視できます。



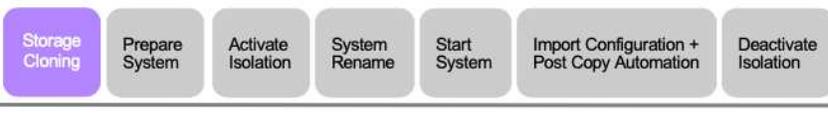
6. 「* Delete NetAppClone *」タスクを選択すると、その手順の詳細ログが表示されます。Ansibleプレビューの実行方法を以下に示します。ご覧のように、Ansibleプレビューもご覧いただけます
 netapp_lama_ServiceConfigRemoval.yml は、HANAデータベースのボリューム、データ、ログ、および共有ごとに実行されます。



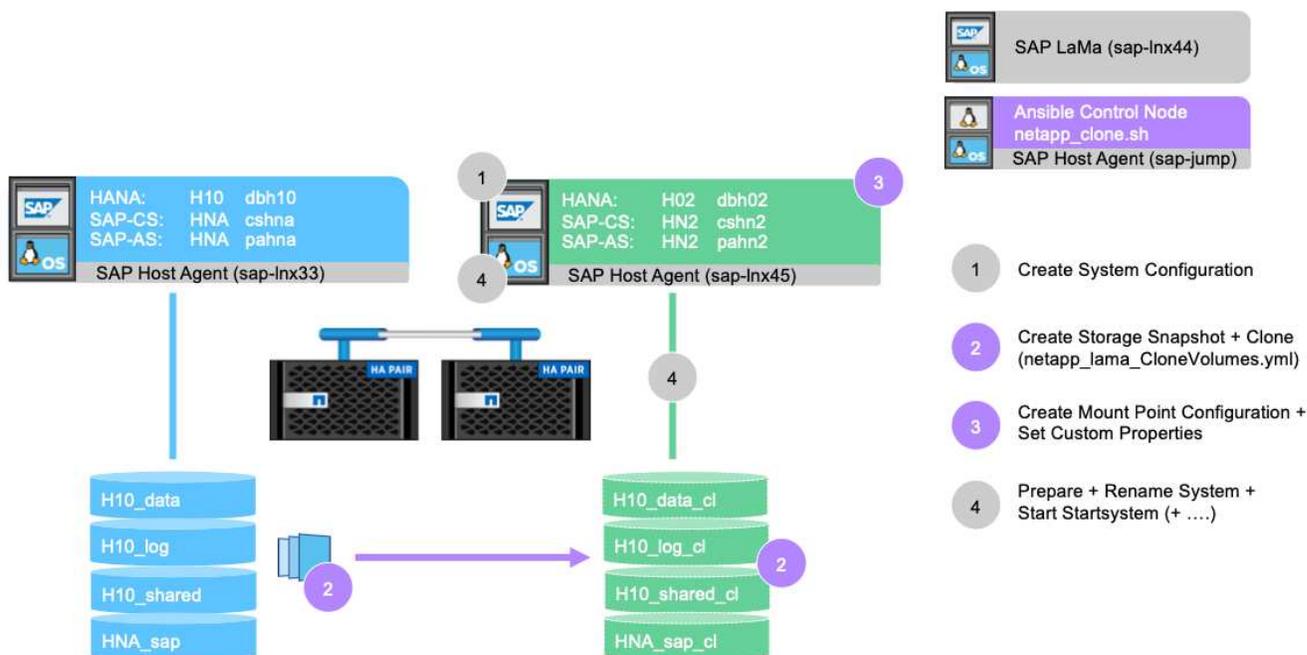
SAP LaMa プロビジョニングワークフロー-コピーシステム

次の図は、システムコピーワークフローで実行する主な手順を示しています。

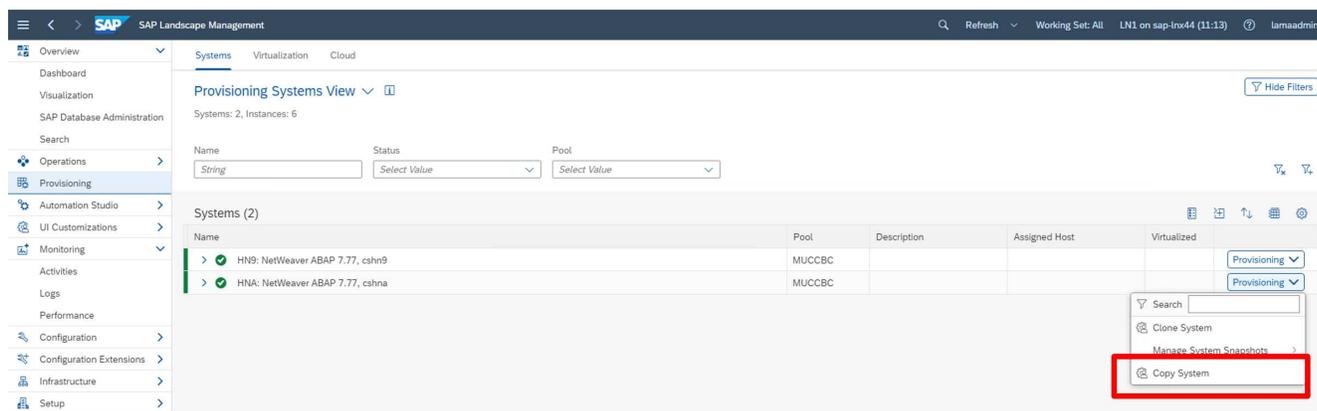
System Copy



この章では、システムクローンのワークフローと入力画面の違いについて簡単に説明します。次の図に示すように、ストレージワークフローは変更されていません。

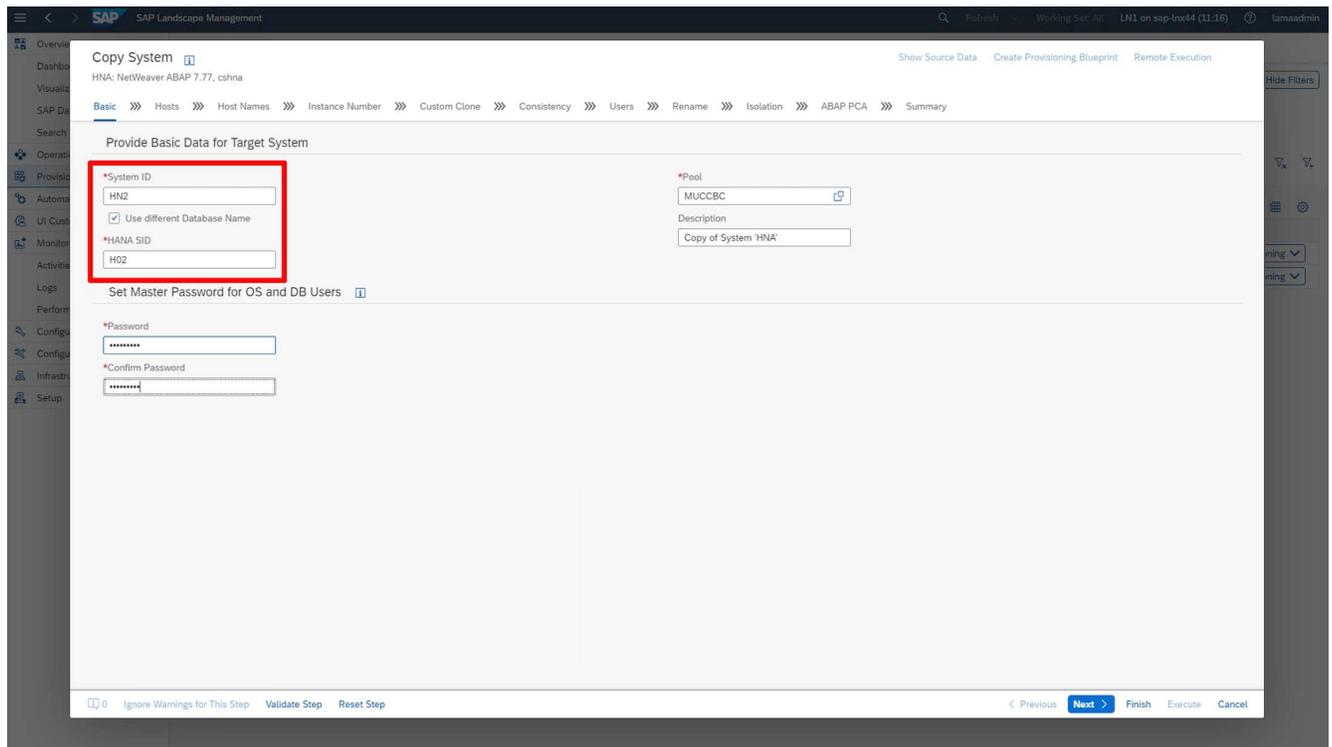


1. システムの準備が整ったら、システムコピーワークフローを開始できます。これは、この設定に関する特定のタスクではなく、詳細な説明も行いません。詳細については、SAP LaMaのドキュメントをご覧ください。

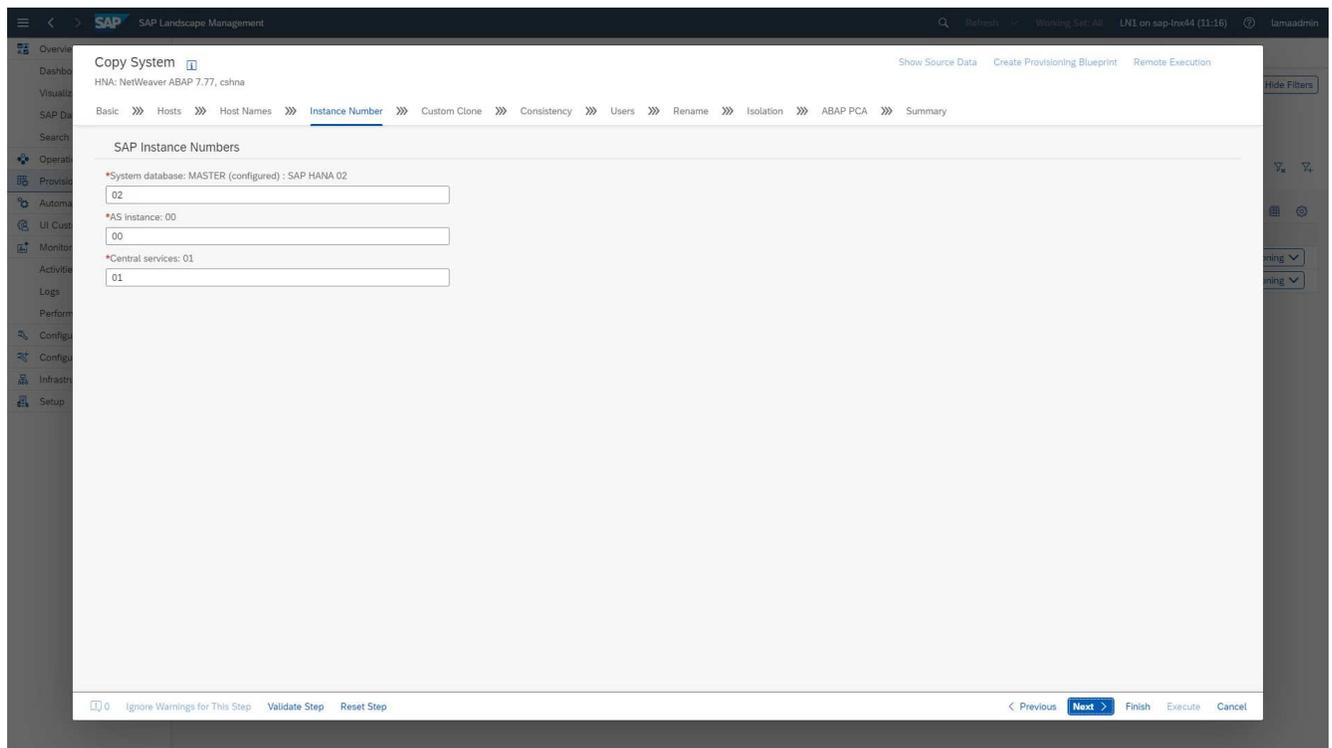


ドロップダウンが強調表示されます。"]

2. コピーワークフローの実行中、最初の画面で指定する必要があるのと同様に、システムの名前が変更されます。

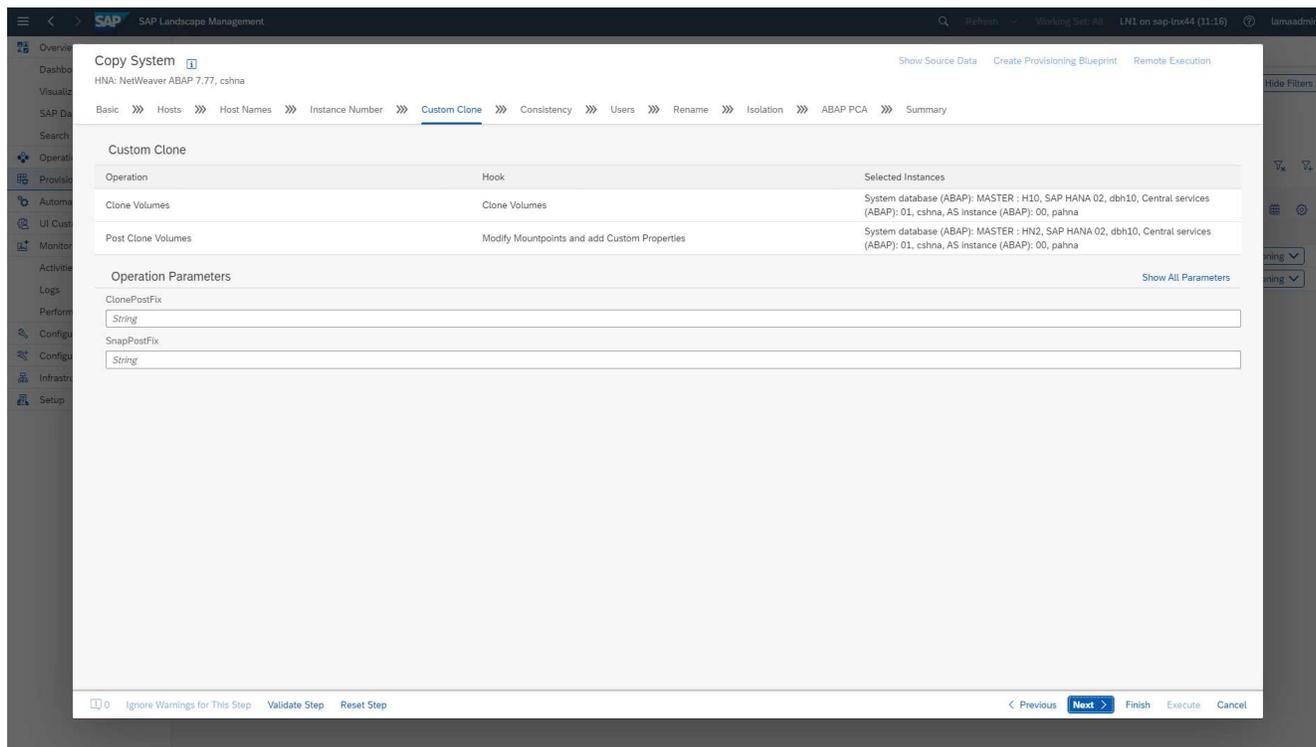


3. ワークフロー中にインスタンス番号を変更することができます。

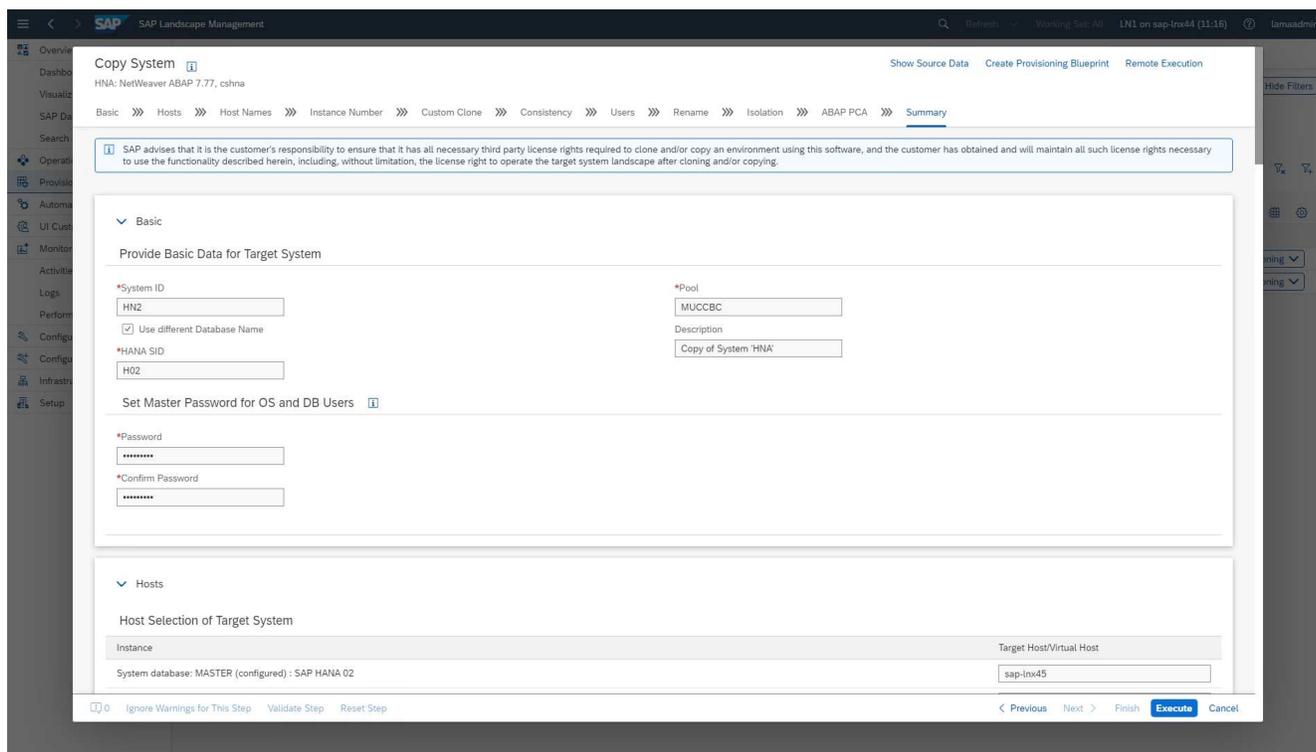


インスタンス番号の変更はテストされていないため、プロバイダスクリプトの変更が必要になる場合があります。

4. 前述のように、* Custom Clone *画面はクローニングワークフローとは異なります。

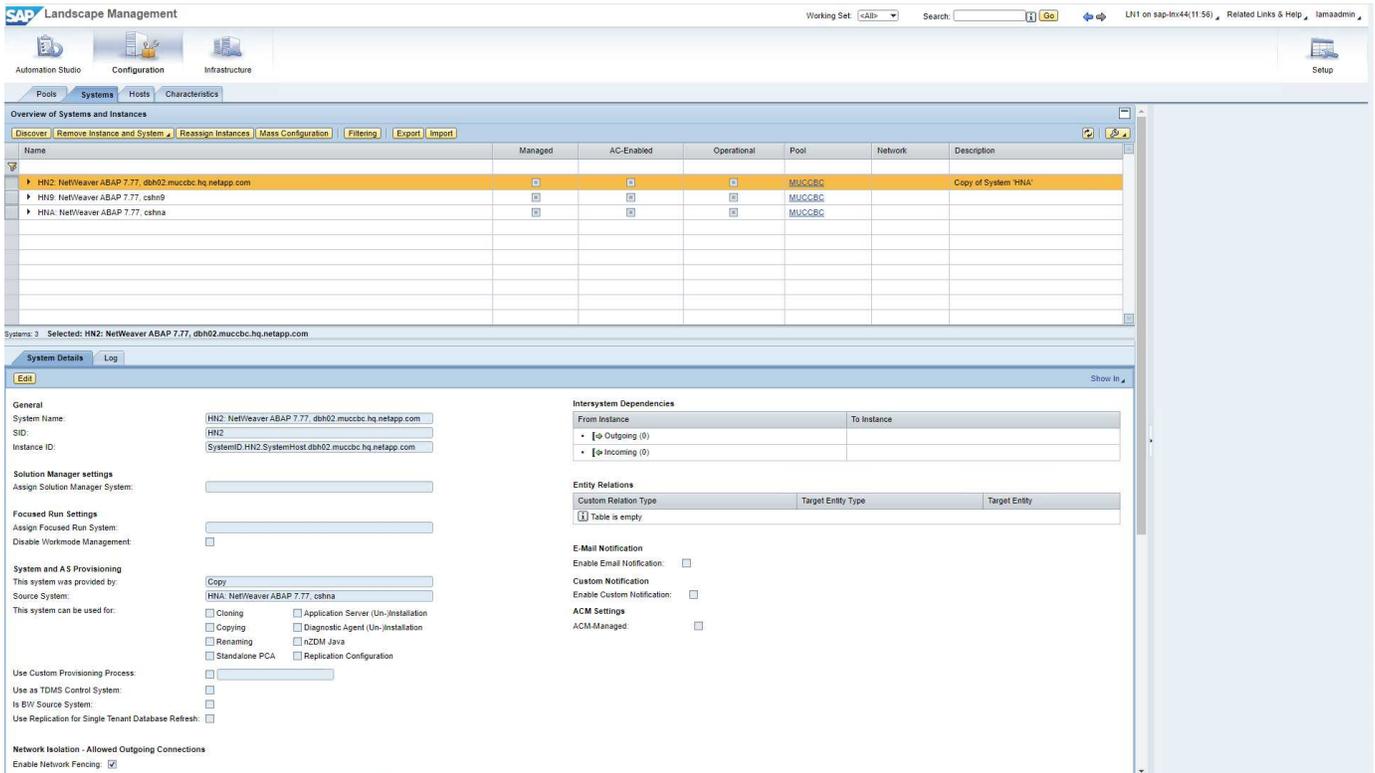


5. すでに説明したように、残りの入力マスクは標準とは異なるので、ここではそれ以上のマスクには入りません。最後の画面に概要が表示され、実行を開始できます。

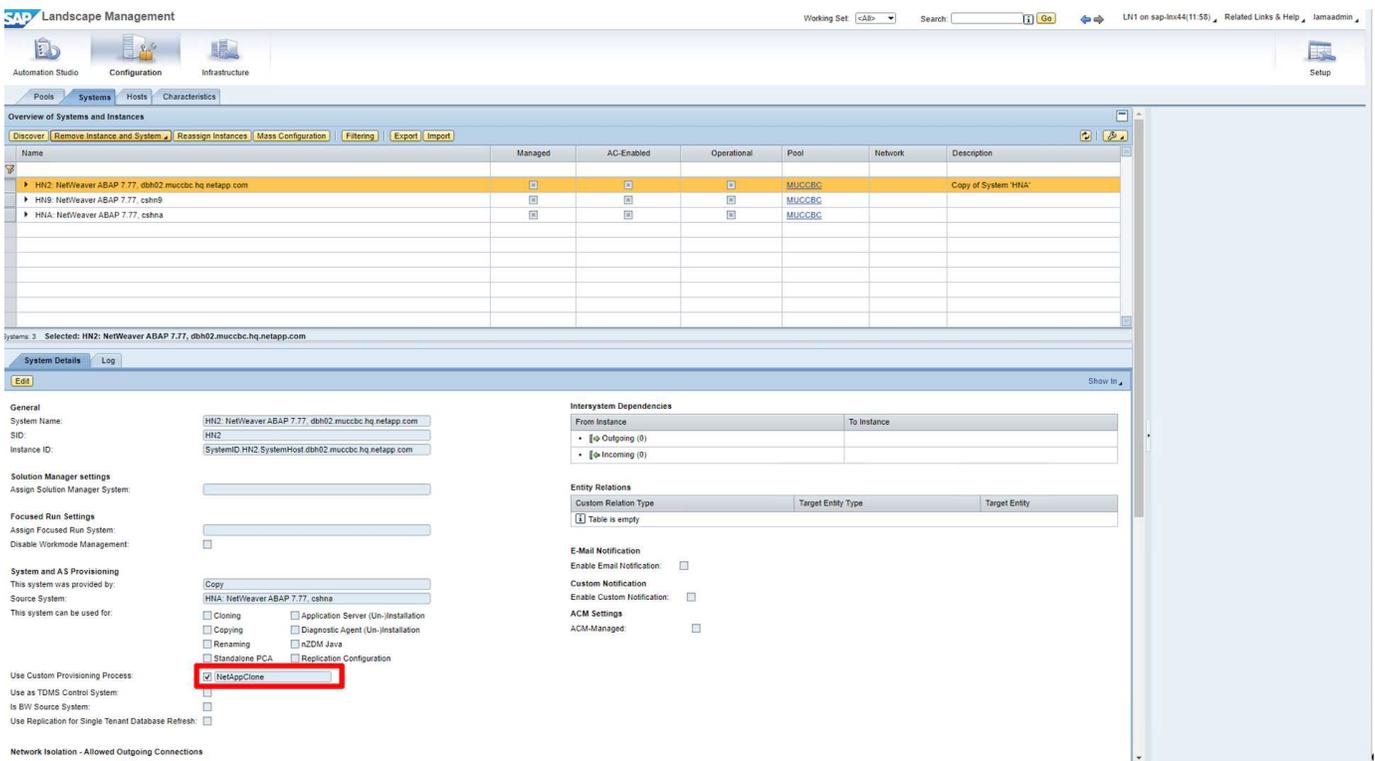


ボタンをクリックします。"]

コピープロセスの完了後、カスタムのクローニングプロセスでターゲットインスタンスが有効になりません。

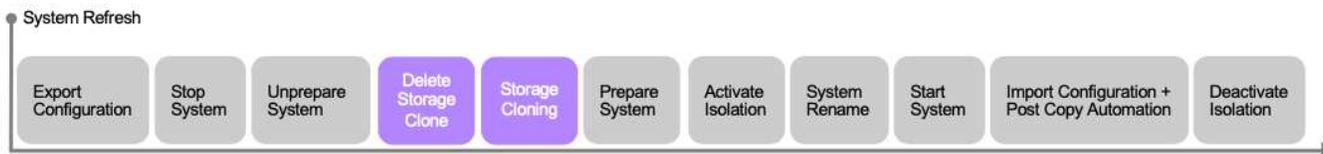


拘束が設定されて実行を妨げるため'システム破壊プロセス中にブックスステップを実行するには'手動で採用する必要があります

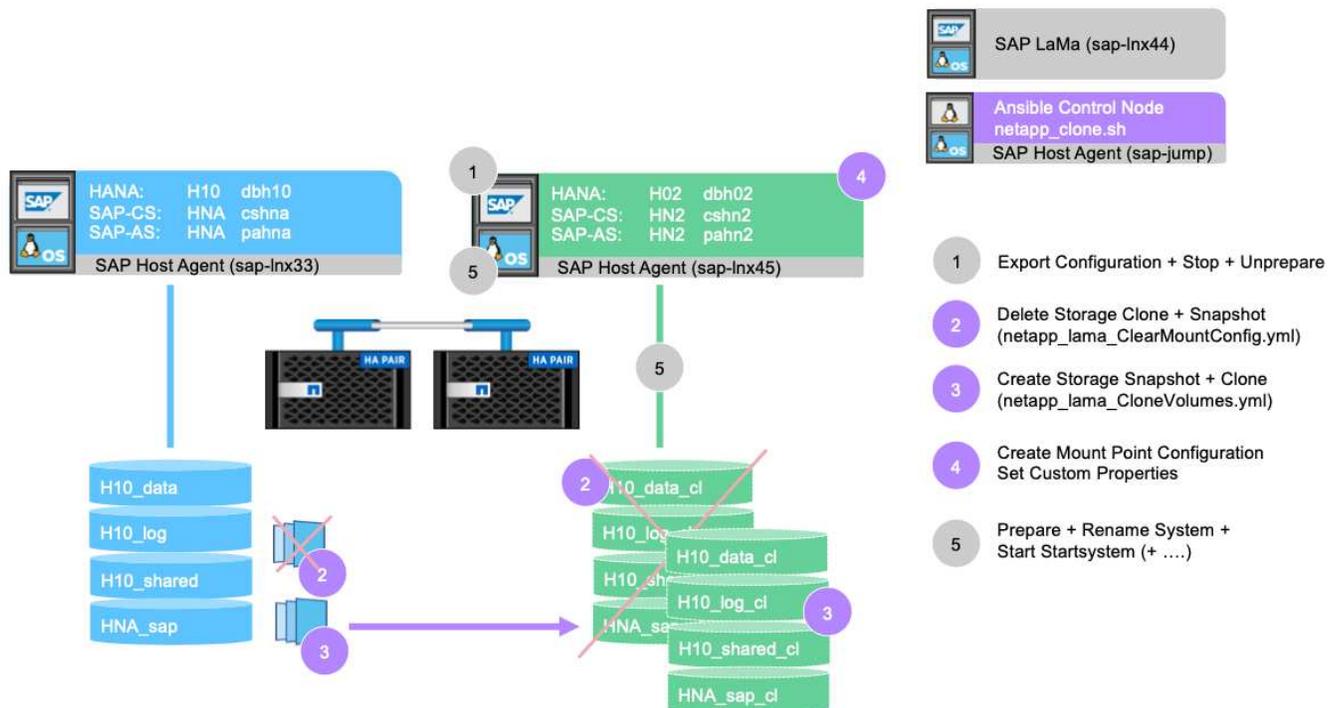


SAP LaMa プロビジョニングワークフロー-システムの更新

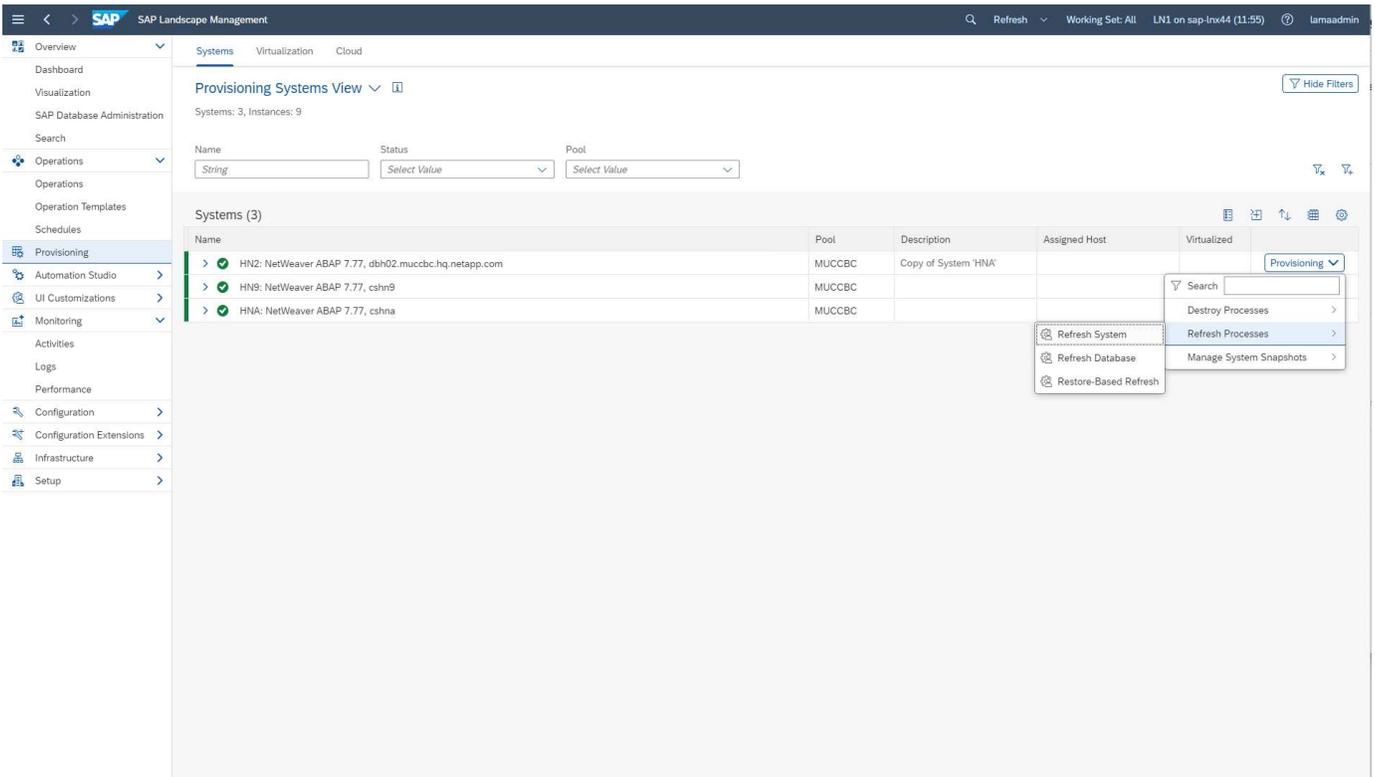
次の図は、システムの更新ワークフローで実行する主な手順を示しています。



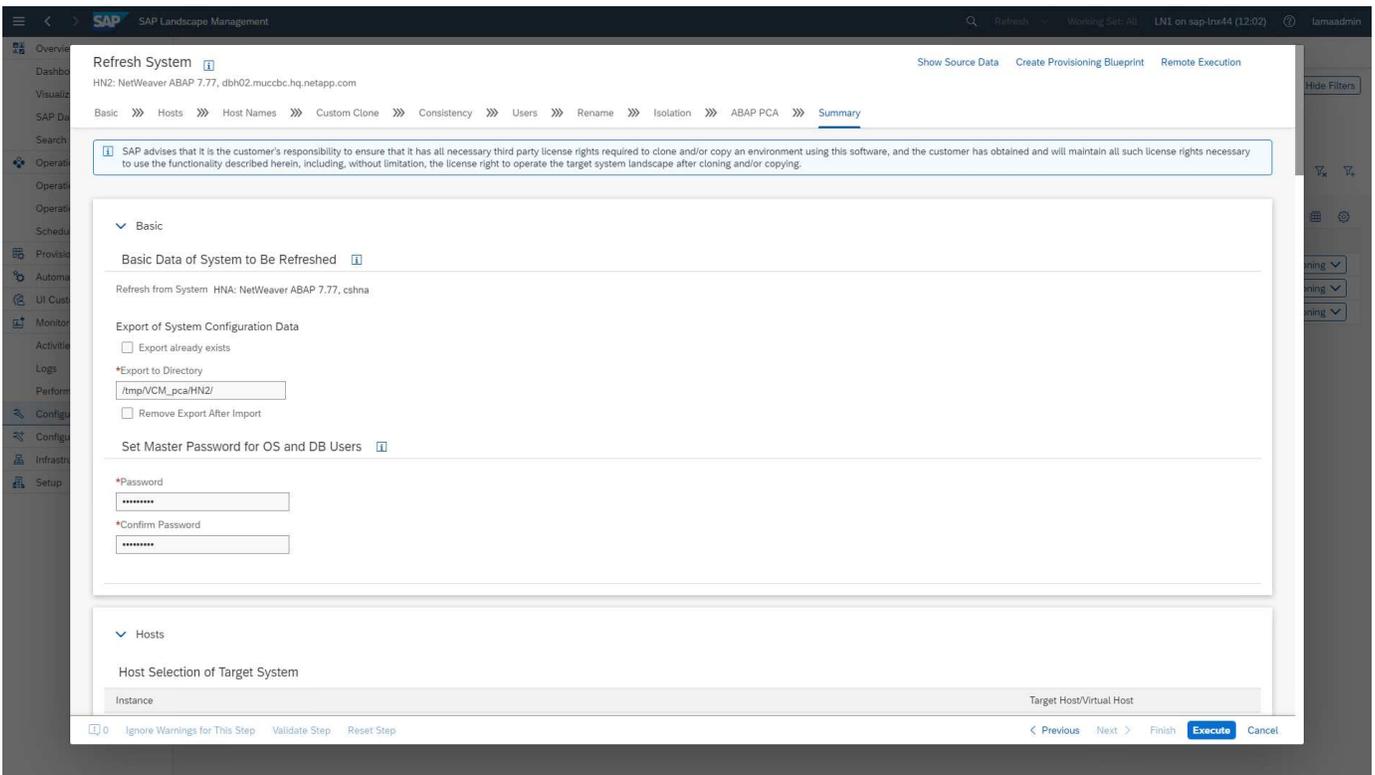
更新ワークフロー中に、ストレージクローンを削除する必要があります。システム破棄ワークフローにも同じAnsibleプレイブックを使用できます。ただし、カスタムフックは別の手順に定義されるため、プレイブックの名前も適宜変更されます。クローンのプロセス手順に違いはありません。



更新ワークフローは、コピーしたシステムのプロビジョニング画面から開始できます。



繰り返しますが、入力画面と標準画面では何も異なりません。ワークフローの実行は概要画面から開始できません。



プロバイダスクリプトの設定とAnsibleのプレイブック

このドキュメントの導入例とワークフローの実行時には、次のプロバイダ構成ファイ

ル、実行スクリプト、およびAnsibleプレイブックを使用します。



このサンプルスクリプトは現状のまま提供されており、ネットアップではサポートしていません。スクリプトの最新バージョンは、mailto:ng-sapcc@netapp.com [ng-sapcc@netapp.com]にEメールでリクエストできます。

プロバイダ構成ファイルNetApp_clone.conf

構成ファイルは、の説明に従って作成されます "SAP LaMaドキュメント-「Configuring SAP Host Agent Registered Scripts"。この構成ファイルは、SAPホストエージェントがインストールされているAnsibleコントロールノードに配置する必要があります。

設定されているos-user sapuser スクリプトおよび呼び出されたAnsibleプレイブックを実行するための適切な権限が必要です。このスクリプトは共通のスクリプトディレクトリに配置できます。SAP LaMaは、スクリプトを呼び出す際に複数のパラメータを提供します。

カスタムパラメータに加えて、PARAM_ClonePostFix、PROP_ClonePostFix、`PARAM_ClonePostFix` および `PROP_ClonePostFix` に示されているように、他の多くは手渡されることができる "SAP LaMaのドキュメント"。

```
root@sap-jump:~# cat /usr/sap/hostctrl/exe/operations.d/netapp_clone.conf
Name: netapp_clone
Username: sapuser
Description: NetApp Clone for Custom Provisioning
Command: /usr/sap/scripts/netapp_clone.sh
--HookOperationName=${HookOperationName} --SAPSYSTEMNAME=${SAPSYSTEMNAME}
--SAPSYSTEM=${SAPSYSTEM} --MOUNT_XML_PATH=${MOUNT_XML_PATH}
--PARAM_ClonePostFix=${PARAM_ClonePostFix} --PARAM_SnapPostFix=${PARAM_SnapPostFix}
--PROP_ClonePostFix=${PROP_ClonePostFix}
--PROP_SnapPostFix=${PROP_SnapPostFix}
--SAP_LVM_SRC_SID=${SAP_LVM_SRC_SID}
--SAP_LVM_TARGET_SID=${SAP_LVM_TARGET_SID}
ResulConverter: hook
Platform: Unix
```

プロバイダスクリプトnetapp_clone.sh

プロバイダスクリプトには保存する必要があります /usr/sap/scripts プロバイダの構成ファイルで設定されているとおりです。

変数 (variables)

次の変数はスクリプトでハードコードされており、適宜変更する必要があります。

- PRIMARY_CLUSTER=<hostname of netapp cluster>
- PRIMARY_SVM=<SVM name where source system volumes are stored>

証明書ファイル PRIMARY_KEYFILE=/usr/sap/scripts/ansible/certs/ontap.key および

PRIMARY_CERTFILE=/usr/sap/scripts/ansible/certs/ontap.pem の説明に従って指定する必要があります "NetApp Ansibleモジュール- ONTAP を準備"。



それぞれのSAPシステムで異なるクラスタやSVMが必要な場合は、SAP LaMaプロバイダ定義のパラメータとして追加できます。

機能：インベントリファイルを作成する

Ansibleプレイブックをより動的に実行するために、を使用します inventory. yml その場でファイルが作成されます。一部の静的値は変数セクションで設定され、一部は実行中に動的に作成されます。

関数：Ansibleプレイブックを実行する

この関数は、Ansibleプレイブックと動的に作成されるを実行するために使用します inventory. yml ファイル。プレイブックの命名規則は netapp_lama_ \${HookOperationName}. yml。 の値 \${HookOperationName} LaMaの運用に依存し、LaMaをコマンドラインパラメータとして引き継ぎます。

セクションMain

ここでは、主な実行計画について説明します。変数 \${HookOperationName} LaMaの代替手順の名前が含まれ、スクリプトが呼び出されるとLaMaによって提供されます。

- システムクローンとシステムコピーのプロビジョニングワークフローを使用した値：
 - CloneVolumes
 - CloneVolumes
- system destroyワークフローを使用した値：
 - ServiceConfigRemovalサービス構成の削除
- システムの更新ワークフローによる値：
 - ClearMountConfigの場合

HookOperationName=CloneVolumes

この手順では、Ansibleプレイブックが実行され、Snapshotコピーとクローニングの処理が開始されます。ボリューム名とマウント設定は、変数で定義されたXMLファイルを使用してSAP LaMaによって引き継がれます \$MOUNT_XML_PATH。このファイルは、あとで手順で使用するために保存されます FinalizeCloneVolumes 新しいマウントポイント設定を作成します。ボリューム名がXMLファイルから抽出され、各ボリュームに対してAnsibleのクローニングプレイブックが実行されます。



この例では、ASインスタンスと中央サービスは同じボリュームを共有します。したがって、ボリュームクローニングはSAPインスタンス番号でのみ実行されます (\$SAPSYSTEM) はではありません 01。これは他の環境とは異なる場合があるため、変更する必要があります。

HookOperationName=PostCloneVolumes

このステップでは、ユーザー定義プロパティを指定します ClonePostFix および SnapPostFix また、ターゲットシステムのマウントポイント設定が保持されます。

カスタムプロパティは、あとででの運用停止時に入力として使用されます ServiceConfigRemoval または ClearMountConfig フェーズ：システムのプロビジョニングワークフローで指定されたカスタムパラメ

ータの設定を保持するように設計されています。

この例で使用されている値はです `ClonePostFix=_clone_20221115` および `SnapPostFix=_snap_20221115`。

ボリューム `HN9_sap`` は、動的に作成される Ansible ファイルに次の値が含まれています。
``datavolumename: HN9_sap、 snapshotpostfix: _snap_20221115`` および ``clonepostfix: _clone_20221115`。

これにより、ボリューム `HN9_SAP` 上の Snapshot 名に変換されます `HN9_sap_snap_20221115` 作成されたボリュームクローンの名前も表示されます `HN9_sap_clone_20221115`。



カスタムプロパティを使用すると、プロビジョニングプロセスで使用されるパラメータを保持できます。

マウントポイントの設定は、で LaMa に引き継がれた XML ファイルから抽出されます `CloneVolume` ステップ。。 `ClonePostFix` がボリューム名に追加され、デフォルトのスクリプト出力で LaMa に返信されます。機能については、を参照してください "[SAP ノート 1889590](#)"。



この例では、ストレージシステム上の `qtree` を、単一のボリュームに異なるデータを配置する一般的な方法として使用します。例: `HN9_sap` のマウントポイントを保持します `/usr/sap/HN9、 /sapmnt/HN9`` および ``/home/hn9adm`。サブディレクトリも同様に機能します。これは他の環境とは異なる場合があるため、変更する必要があります。

HookOperationName = ServiceConfigRemoval

この手順では、ボリュームクローンの削除を実行する Ansible プレイブックを使用します。

ボリューム名は、マウント構成ファイルとカスタムプロパティを通じて SAP LaMa から引き継がれます `ClonePostFix` および `SnapPostFix` は、システムプロビジョニングワークフローで最初に指定されたパラメータの値を渡すために使用します (の注を参照) `HookOperationName = PostCloneVolumes`) 。

ボリューム名が XML ファイルから抽出され、各ボリュームに対して Ansible のクローニングプレイブックが実行されます。



この例では、AS インスタンスと中央サービスは同じボリュームを共有します。そのため、ボリュームの削除は SAP インスタンス番号がの場合にのみ実行されます (`$SAPSYSTEM`) はではありません `01`。これは他の環境とは異なる場合があるため、変更する必要があります。

HookOperationName=ClearMountConfig

この手順では、システムの更新ワークフロー中にボリュームクローンを削除する Ansible プレイブックを実行します。

ボリューム名は、マウント構成ファイルとカスタムプロパティを通じて SAP LaMa から引き継がれます `ClonePostFix` および `SnapPostFix` は、システムプロビジョニングワークフローで最初に指定されたパラメータの値を渡すために使用します。

ボリューム名が XML ファイルから抽出され、各ボリュームに対して Ansible のクローニングプレイブックが実行されます。



この例では、ASインスタンスと中央サービスは同じボリュームを共有します。そのため、ボリュームの削除はSAPインスタンス番号がの場合にのみ実行されます (\$SAPSYSTEM) はではありません 01。これは他の環境とは異なる場合があるため、変更する必要があります。

```
root@sap-jump:~# cat /usr/sap/scripts/netapp_clone.sh
#!/bin/bash
#Section - Variables
#####
VERSION="Version 0.9"
#Path for ansible play-books
ANSIBLE_PATH=/usr/sap/scripts/ansible
#Values for Ansible Inventory File
PRIMARY_CLUSTER=grenada
PRIMARY_SVM=svm-sap01
PRIMARY_KEYFILE=/usr/sap/scripts/ansible/certs/ontap.key
PRIMARY_CERTFILE=/usr/sap/scripts/ansible/certs/ontap.pem
#Default Variable if PARAM ClonePostFix / SnapPostFix is not maintained in
LaMa
DefaultPostFix=_clone_1
#TMP Files - used during execution
YAML_TMP=/tmp/inventory_ansible_clone_tmp_$$.yml
TMPFILE=/tmp/tmpfile.$$
MY_NAME="`basename $0`"
BASE_SCRIPT_DIR="`dirname $0`"
#Sendig Script Version and run options to LaMa Log
echo "[DEBUG]: Running Script $MY_NAME $VERSION"
echo "[DEBUG]: $MY_NAME $@"
#Command declared in the netapp_clone.conf Provider definition
#Command: /usr/sap/scripts/netapp_clone.sh
--HookOperationName=${HookOperationName} --SAPSYSTEMNAME=${SAPSYSTEMNAME}
--SAPSYSTEM=${SAPSYSTEM} --MOUNT_XML_PATH=${MOUNT_XML_PATH}
--PARAM_ClonePostFix=${PARAM-ClonePostFix} --PARAM_SnapPostFix=${PARAM
-SnapPostFix} --PROP_ClonePostFix=${PROP-ClonePostFix}
--PROP_SnapPostFix=${PROP-SnapPostFix}
--SAP_LVM_SRC_SID=${SAP_LVM_SRC_SID}
--SAP_LVM_TARGET_SID=${SAP_LVM_TARGET_SID}
#Reading Input Variables hand over by LaMa
for i in "$@"
do
case $i in
--HookOperationName=*)
HookOperationName="${i#*=}";shift;;
--SAPSYSTEMNAME=*)
SAPSYSTEMNAME="${i#*=}";shift;;
--SAPSYSTEM=*)
```

```

SAPSYSTEM="${i#*=}";shift;;
--MOUNT_XML_PATH=*)
MOUNT_XML_PATH="${i#*=}";shift;;
--PARAM_ClonePostFix=*)
PARAM_ClonePostFix="${i#*=}";shift;;
--PARAM_SnapPostFix=*)
PARAM_SnapPostFix="${i#*=}";shift;;
--PROP_ClonePostFix=*)
PROP_ClonePostFix="${i#*=}";shift;;
--PROP_SnapPostFix=*)
PROP_SnapPostFix="${i#*=}";shift;;
--SAP_LVM_SRC_SID=*)
SAP_LVM_SRC_SID="${i#*=}";shift;;
--SAP_LVM_TARGET_SID=*)
SAP_LVM_TARGET_SID="${i#*=}";shift;;
*)
# unknown option
;;
esac
done
#If Parameters not provided by the User - defaulting to DefaultPostFix
if [ -z $PARAM_ClonePostFix ]; then PARAM_ClonePostFix=$DefaultPostFix;fi
if [ -z $PARAM_SnapPostFix ]; then PARAM_SnapPostFix=$DefaultPostFix;fi
#Section - Functions
#####
#Function Create (Inventory) YML File
#####
create_yml_file()
{
echo "ontapservers:">$YAML_TMP
echo " hosts:">>$YAML_TMP
echo "  ${PRIMARY_CLUSTER}:">>$YAML_TMP
echo "  ansible_host: "'"$PRIMARY_CLUSTER'"'">>$YAML_TMP
echo "  keyfile: "'"$PRIMARY_KEYFILE'"'">>$YAML_TMP
echo "  certfile: "'"$PRIMARY_CERTFILE'"'">>$YAML_TMP
echo "  svmname: "'"$PRIMARY_SVM'"'">>$YAML_TMP
echo "  datavolumename: "'"$datavolumename'"'">>$YAML_TMP
echo "  snapshotpostfix: "'"$snapshotpostfix'"'">>$YAML_TMP
echo "  clonepostfix: "'"$clonepostfix'"'">>$YAML_TMP
}
#Function run ansible-playbook
#####
run_ansible_playbook()
{
echo "[DEBUG]: Running ansible playbook
netapp_lama_${HookOperationName}.yml on Volume $datavolumename"

```

```

ansible-playbook -i $YAML_TMP
$ANSIBLE_PATH/netapp_lama_${HookOperationName}.yaml
}
#Section - Main
#####
#HookOperationName - CloneVolumes
#####
if [ $HookOperationName = CloneVolumes ] ;then
#save mount xml for later usage - used in Section FinalizeCloneVolumes to
generate the mountpoints
echo "[DEBUG]: saving mount config...."
cp $MOUNT_XML_PATH /tmp/mount_config_${SAPSYSTEMNAME}_${SAPSYSTEM}.xml
#Instance 00 + 01 share the same volumes - clone needs to be done once
if [ $SAPSYSTEM != 01 ]; then
#generating Volume List - assuming usage of qtrees - "IP-
Adress:/VolumeName/qtrees"
xmlFile=/tmp/mount_config_${SAPSYSTEMNAME}_${SAPSYSTEM}.xml
if [ -e $TMPFILE ];then rm $TMPFILE;fi
numMounts=`xml_grep --count "/mountconfig/mount" $xmlFile | grep "total: "
| awk '{ print $2 }'`
i=1
while [ $i -le $numMounts ]; do
    xmllint --xpath "/mountconfig/mount[$i]/exportpath/text()" $xmlFile
|awk -F"/" '{print $2}' >>$TMPFILE
i=$((i + 1))
done
DATAVOLUMES=`cat $TMPFILE |sort -u`
#Create yaml file and rund playbook for each volume
for I in $DATAVOLUMES; do
datavolumename="$I"
snapshotpostfix="$PARAM_SnapPostFix"
clonepostfix="$PARAM_ClonePostFix"
create_yaml_file
run_ansible_playbook
done
else
echo "[DEBUG]: Doing nothing .... Volume cloned in different Task"
fi
fi
#HookOperationName - PostCloneVolumes
#####
if [ $HookOperationName = PostCloneVolumes ] ;then
#Reporting Properties back to LaMa Config for Cloned System
echo "[RESULT]:Property:ClonePostFix=$PARAM_ClonePostFix"
echo "[RESULT]:Property:SnapPostFix=$PARAM_SnapPostFix"
#Create MountPoint Config for Cloned Instances and report back to LaMa

```

```

according to SAP Note: https://launchpad.support.sap.com/#/notes/1889590
echo "MountDataBegin"
echo '<?xml version="1.0" encoding="UTF-8"?>'
echo "<mountconfig>"
xmlFile=/tmp/mount_config_${SAPSYSTEMNAME}_${SAPSYSTEM}.xml
numMounts=`xml_grep --count "/mountconfig/mount" $xmlFile | grep "total: "
| awk '{ print $2 }'`
i=1
while [ $i -le $numMounts ]; do
MOUNTPOINT=`xmllint --xpath "/mountconfig/mount[$i]/mountpoint/text()"
$xmlFile`;
    EXPORTPATH=`xmllint --xpath
"/mountconfig/mount[$i]/exportpath/text()" $xmlFile`;
    OPTIONS=`xmllint --xpath "/mountconfig/mount[$i]/options/text()"
$xmlFile`;
#Adopt Exportpath and add Clonepostfix - assuming usage of qtrees - "IP-
Adress:/VolumeName/qtrees"
TMPFIELD1=`echo $EXPORTPATH|awk -F"/" '{print $1}'`
TMPFIELD2=`echo $EXPORTPATH|awk -F"/" '{print $2}'`
TMPFIELD3=`echo $EXPORTPATH|awk -F"/" '{print $3}'`
EXPORTPATH=$TMPFIELD1":/${TMPFIELD2}$PARAM_ClonePostFix"/$TMPFIELD3
echo -e '\t<mount fstype="nfs" storagetype="NETFS">'
echo -e "\t\t<mountpoint>${MOUNTPOINT}</mountpoint>"
echo -e "\t\t<exportpath>${EXPORTPATH}</exportpath>"
echo -e "\t\t<options>${OPTIONS}</options>"
echo -e "\t</mount>"
i=$((i + 1))
done
echo "</mountconfig>"
echo "MountDataEnd"
#Finished MountPoint Config
#Cleanup Temporary Files
rm $xmlFile
fi
#HookOperationName - ServiceConfigRemoval
#####
if [ $HookOperationName = ServiceConfigRemoval ] ;then
#Assure that Properties ClonePostFix and SnapPostfix has been configured
through the provisioning process
if [ -z $PROP_ClonePostFix ]; then echo "[ERROR]: Propertiy ClonePostFix
is not handed over - please investigate";exit 5;fi
if [ -z $PROP_SnapPostFix ]; then echo "[ERROR]: Propertiy SnapPostFix is
not handed over - please investigate";exit 5;fi
#Instance 00 + 01 share the same volumes - clone delete needs to be done
once
if [ $SAPSYSTEM != 01 ]; then

```

```

#generating Volume List - assuming usage of qtrees - "IP-
Adress:/VolumeName/qtrees"
xmlFile=$MOUNT_XML_PATH
if [ -e $TMPFILE ];then rm $TMPFILE;fi
numMounts=`xml_grep --count "/mountconfig/mount" $xmlFile | grep "total: "
| awk '{ print $2 }'`
i=1
while [ $i -le $numMounts ]; do
    xmllint --xpath "/mountconfig/mount[$i]/exportpath/text()" $xmlFile
|awk -F"/" '{print $2}' >>$TMPFILE
i=$((i + 1))
done
DATAVOLUMES=`cat $TMPFILE |sort -u| awk -F $PROP_ClonePostFix '{ print $1
}'`
#Create yml file and rund playbook for each volume
for I in $DATAVOLUMES; do
datavolumename="$I"
snapshotpostfix="$PROP_SnapPostFix"
clonepostfix="$PROP_ClonePostFix"
create_yml_file
run_ansible_playbook
done
else
echo "[DEBUG]: Doing nothing .... Volume deleted in different Task"
fi
#Cleanup Temporary Files
rm $xmlFile
fi
#HookOperationName - ClearMountConfig
#####
if [ $HookOperationName = ClearMountConfig ] ;then
    #Assure that Properties ClonePostFix and SnapPostfix has been
configured through the provisioning process
    if [ -z $PROP_ClonePostFix ]; then echo "[ERROR]: Propertiy
ClonePostFix is not handed over - please investigate";exit 5;fi
    if [ -z $PROP_SnapPostFix ]; then echo "[ERROR]: Propertiy
SnapPostFix is not handed over - please investigate";exit 5;fi
    #Instance 00 + 01 share the same volumes - clone delete needs to
be done once
    if [ $SAPSYSTEM != 01 ]; then
        #generating Volume List - assuming usage of qtrees - "IP-
Adress:/VolumeName/qtrees"
        xmlFile=$MOUNT_XML_PATH
        if [ -e $TMPFILE ];then rm $TMPFILE;fi
        numMounts=`xml_grep --count "/mountconfig/mount" $xmlFile
| grep "total: " | awk '{ print $2 }'`

```

```

        i=1
        while [ $i -le $numMounts ]; do
            xmllint --xpath
"/mountconfig/mount[$i]/exportpath/text()" $xmlFile |awk -F"/" '{print
$2}' >>$TMPFILE
                i=$((i + 1))
        done
        DATAVOLUMES=`cat $TMPFILE |sort -u| awk -F
$PROP_ClonePostFix '{ print $1 }'`
        #Create yml file and rund playbook for each volume
        for I in $DATAVOLUMES; do
            datavolumename="$I"
            snapshotpostfix="$PROP_SnapPostFix"
            clonepostfix="$PROP_ClonePostFix"
            create_yml_file
            run_ansible_playbook
        done
    else
        echo "[DEBUG]: Doing nothing .... Volume deleted in
different Task"
        fi
        #Cleanup Temporary Files
        rm $xmlFile
    fi
#Cleanup
#####
#Cleanup Temporary Files
if [ -e $TMPFILE ];then rm $TMPFILE;fi
if [ -e $YAML_TMP ];then rm $YAML_TMP;fi
exit 0

```

Ansible Playbook : NetApp_LaMa_CloneVolume.yml

LaMaシステムのクローニングワークフローのCloneVolumesの手順で実行されるPlaybookは、を組み合わせたものです create_snapshot.yml および create_clone.yml (を参照) ["NetApp Ansibleモジュール-YAMLファイル"](#))。このプレイブックは、セカンダリからのクローニング処理やクローンプリット処理など、他のユースケースにも簡単に対応できます。

```

root@sap-jump:~# cat /usr/sap/scripts/ansible/netapp_lama_CloneVolumes.yml
---
- hosts: ontapserver
  connection: local
  collections:
    - netapp.ontap
  gather_facts: false
  name: netapp_lama_CloneVolumes
  tasks:
  - name: Create SnapShot
    na_ontap_snapshot:
      state: present
      snapshot: "{{ datavolumename }}{{ snapshotpostfix }}"
      use_rest: always
      volume: "{{ datavolumename }}"
      vsserver: "{{ svmname }}"
      hostname: "{{ inventory_hostname }}"
      cert_filepath: "{{ certfile }}"
      key_filepath: "{{ keyfile }}"
      https: true
      validate_certs: false
  - name: Clone Volume
    na_ontap_volume_clone:
      state: present
      name: "{{ datavolumename }}{{ clonepostfix }}"
      use_rest: always
      vsserver: "{{ svmname }}"
      junction_path: '/{{ datavolumename }}{{ clonepostfix }}'
      parent_volume: "{{ datavolumename }}"
      parent_snapshot: "{{ datavolumename }}{{ snapshotpostfix }}"
      hostname: "{{ inventory_hostname }}"
      cert_filepath: "{{ certfile }}"
      key_filepath: "{{ keyfile }}"
      https: true
      validate_certs: false

```

Ansible Playbook : NetApp_LaMa_ServiceConfigRemoval.yml

実行されるプレイブック ServiceConfigRemoval LaMaシステムの破棄ワークフローのフェーズは、のフェーズです delete_clone.yml および delete_snapshot.yml (を参照) ["NetApp Ansibleモジュール-YAMLファイル"](#))。の実行ステップに合わせて調整する必要があります netapp_lama_CloneVolumes Playbook :

```

root@sap-jump:~# cat
/usr/sap/scripts/ansible/netapp_lama_ServiceConfigRemoval.yml
---
- hosts: ontapservers
  connection: local
  collections:
    - netapp.ontap
  gather_facts: false
  name: netapp_lama_ServiceConfigRemoval
  tasks:
    - name: Delete Clone
      na_ontap_volume:
        state: absent
        name: "{{ datavolumename }}{{ clonepostfix }}"
        use_rest: always
        vservers: "{{ svmname }}"
        wait_for_completion: True
        hostname: "{{ inventory_hostname }}"
        cert_filepath: "{{ certfile }}"
        key_filepath: "{{ keyfile }}"
        https: true
        validate_certs: false
    - name: Delete Snapshot
      na_ontap_snapshot:
        state: absent
        snapshot: "{{ datavolumename }}{{ snapshotpostfix }}"
        use_rest: always
        volume: "{{ datavolumename }}"
        vservers: "{{ svmname }}"
        hostname: "{{ inventory_hostname }}"
        cert_filepath: "{{ certfile }}"
        key_filepath: "{{ keyfile }}"
        https: true
        validate_certs: false
root@sap-jump:~#

```

Ansible Playbook : NetApp_LaMa_ClearMountConfig.yml

プレイブックは、の実行時に指定します netapp_lama_ClearMountConfig LaMaシステムの更新ワークフローのフェーズは、のフェーズです delete_clone.yml および delete_snapshot.yml (を参照) "NetApp Ansibleモジュール- YAMLファイル")。の実行ステップに合わせて調整する必要があります netapp_lama_CloneVolumes Playbook :

```

root@sap-jump:~# cat
/usr/sap/scripts/ansible/netapp_lama_ServiceConfigRemoval.yml
---
- hosts: ontapservers
  connection: local
  collections:
    - netapp.ontap
  gather_facts: false
  name: netapp_lama_ServiceConfigRemoval
  tasks:
    - name: Delete Clone
      na_ontap_volume:
        state: absent
        name: "{{ datavolumename }}{{ clonepostfix }}"
        use_rest: always
        vserver: "{{ svmname }}"
        wait_for_completion: True
        hostname: "{{ inventory_hostname }}"
        cert_filepath: "{{ certfile }}"
        key_filepath: "{{ keyfile }}"
        https: true
        validate_certs: false
    - name: Delete SnapShot
      na_ontap_snapshot:
        state: absent
        snapshot: "{{ datavolumename }}{{ snapshotpostfix }}"
        use_rest: always
        volume: "{{ datavolumename }}"
        vserver: "{{ svmname }}"
        hostname: "{{ inventory_hostname }}"
        cert_filepath: "{{ certfile }}"
        key_filepath: "{{ keyfile }}"
        https: true
        validate_certs: false
root@sap-jump:~#

```

Ansibleのinventory.ymlの例

このインベントリファイルは、ワークフローの実行時に動的に作成されます。このファイルは、説明のためにのみここに表示されています。

```
ontapserver:
  hosts:
    grenada:
      ansible_host: "grenada"
      keyfile: "/usr/sap/scripts/ansible/certs/ontap.key"
      certfile: "/usr/sap/scripts/ansible/certs/ontap.pem"
      svmname: "svm-sap01"
      datavolumename: "HN9_sap"
      snapshotpostfix: " _snap_20221115"
      clonepostfix: " _clone_20221115"
```

まとめ

Ansibleなどの最新の自動化フレームワークがSAP LaMaプロビジョニングワークフローに統合されているため、柔軟な解決策を利用して、標準的なインフラや複雑なインフラ要件に対応できます。

追加情報の参照先

このドキュメントに記載されている情報の詳細については、以下のドキュメントや Web サイトを参照してください。

- NetApp名前スペース内のコレクション

["https://docs.ansible.com/ansible/latest/collections/netapp/index.html"](https://docs.ansible.com/ansible/latest/collections/netapp/index.html)

- AnsibleとAnsibleのPlaybookのサンプルに関するドキュメント

["https://github.com/sap-linuxlab/demo.netapp_ontap"](https://github.com/sap-linuxlab/demo.netapp_ontap)

- Ansibleとネットアップの統合

["https://www.ansible.com/integrations/infrastructure/netapp"](https://www.ansible.com/integrations/infrastructure/netapp)

- SAP LaMaとAnsibleの統合に関するブログ

["https://blogs.sap.com/2020/06/08/outgoing-api-calls-from-sap-landscape-management-lama-with-automation-studio/"](https://blogs.sap.com/2020/06/08/outgoing-api-calls-from-sap-landscape-management-lama-with-automation-studio/)

- SAP Landscape Management 3.0, Enterprise Edition Documentation

["https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/4df88a8f418c5059e1000000a42189c.html#loio4df88a8f418c5059e1000000a42189c"](https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/4df88a8f418c5059e1000000a42189c.html#loio4df88a8f418c5059e1000000a42189c)

- SAP LaMaのドキュメント–プロバイダの定義

["https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/bf6b3e43340a4cbcb0c0f3089715c068.html"](https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/bf6b3e43340a4cbcb0c0f3089715c068.html)

- SAP LaMaドキュメント-カスタムフック

["https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/139eca2f925e48738a20dbf0b56674c5.html"](https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/139eca2f925e48738a20dbf0b56674c5.html)

- SAP LaMaドキュメント-「Configuring SAP Host Agent Registered Scripts

["https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/250dfc5eef4047a38bab466c295d3a49.html"](https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/250dfc5eef4047a38bab466c295d3a49.html)

- SAP LaMaドキュメント-カスタム運用とカスタムフックのパラメータ

["https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/0148e495174943de8c1c3ee1b7c9cc65.html"](https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/0148e495174943de8c1c3ee1b7c9cc65.html)

- SAP LaMaドキュメント-アダプティブデザイン

["https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/737a99e86f8743bdb8d1f6cf4b862c79.html"](https://help.sap.com/doc/700f9a7e52c7497cad37f7c46023b7ff/3.0.11.0/en-US/737a99e86f8743bdb8d1f6cf4b862c79.html)

- ネットアップの製品マニュアル

["https://www.netapp.com/support-and-training/documentation/"](https://www.netapp.com/support-and-training/documentation/)

バージョン履歴

バージョン	日付	ドキュメントのバージョン履歴
バージョン 1.0 以降	2023年1月	初版リリース

SnapCenter を使用して SAP HANA システムのコピーおよびクローン処理を自動化

SnapCenter を使用して SAP HANA システムのコピーおよびクローン処理を自動化

今日のダイナミックなビジネス環境では、企業は継続的なイノベーションを提供し、変化する市場に迅速に対応する必要があります。このような競争の激しい状況下で、業務プロセスの柔軟性を高める企業は、市場の需要に効果的に対応できます。

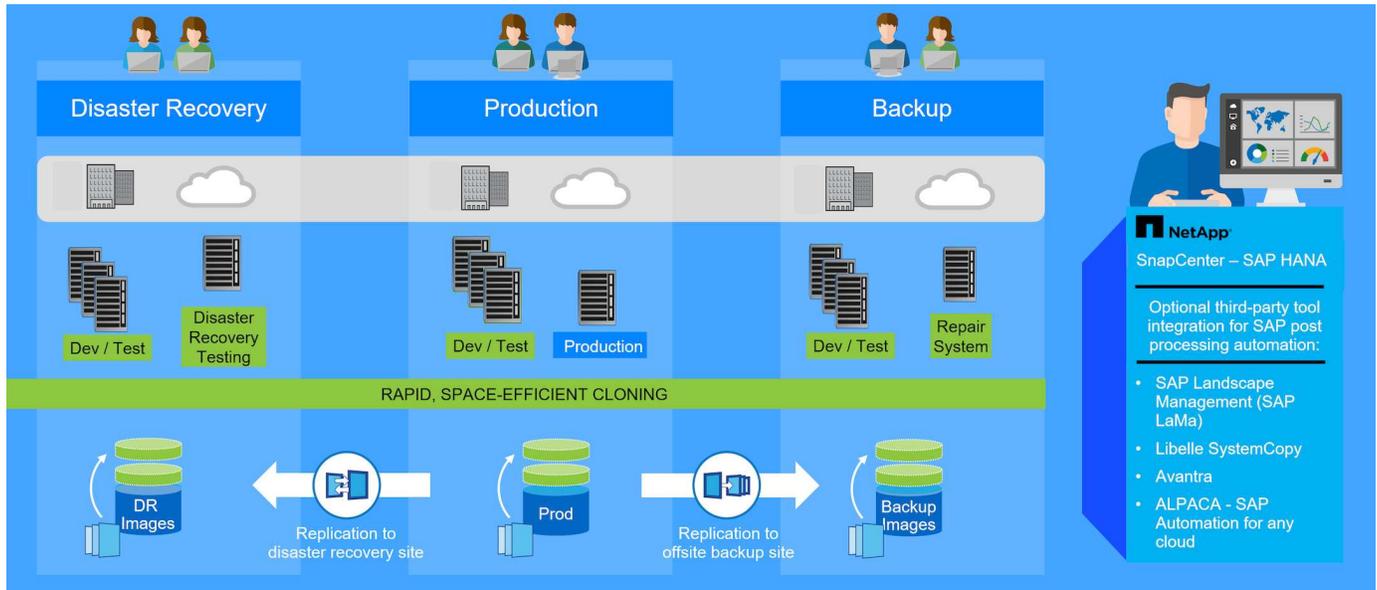
作成者：Nils Bauer、NetApp

はじめに

市場ニーズの変化は、企業のSAP環境にも影響を及ぼし、定期的な統合、変更、更新が必要になります。IT部門は、これらの変更を、より少ないリソースでより短い期間で実装する必要があります。変更を導入する際のリスクを最小限に抑えるには、本番環境の実際のデータを使用するSAPシステムを追加で必要とする、テストとトレーニングを徹底的に実施する必要があります。

従来のSAPライフサイクル管理アプローチでは、このようなシステムは主に手動プロセスに基づいてプロビジョニングされます。このような手動プロセスは、ミス避けられず、時間もかかることが多く、イノベーションの遅れやビジネス要件への対応の遅れにつながります。

次の図に示すように、SAP HANAデータベースとライフサイクル管理ツールにSAPのライフサイクル管理を最適化するネットアップのソリューションが統合されています。効率的なアプリケーション統合データプロテクションに、SAPのテストシステムを柔軟にプロビジョニングすることができます。これらのソリューションは、オンプレミスのSAP HANAにも、Azure NetApp Files (ANF) やAmazon FSx for NetApp ONTAP (FSx for ONTAP) のクラウドで実行されているSAP HANAにも対応しています。



アプリケーションと統合された**Snapshot**バックアップ処理

アプリケーションと整合性のあるSnapshotバックアップをストレージレイヤで作成する機能は、本ドキュメントで説明するシステムコピー処理とシステムクローニング処理の基盤となります。ストレージベースのSnapshotバックアップは、SAP HANA用のNetApp SnapCenter プラグインと、SAP HANAデータベースが提供するインターフェイスを使用して作成します。SnapCenter は、SnapshotバックアップをSAP HANAバックアップカタログに登録して、リストアやリカバリ、クローニング処理に使用できるようにします。

オフサイトのバックアップおよびまたは災害復旧データのレプリケーション

アプリケーションと整合性のあるSnapshotバックアップを、ストレージレイヤから、SnapCenter の制御下にあるオフサイトのバックアップサイトやディザスタリカバリサイトにレプリケートします。レプリケーションは変更されたブロックと新しいブロックに基づいて行われるため、スペースと帯域幅の効率が向上します。

任意の**Snapshot**バックアップを**SAP**システムのコピーやクローン処理に使用

ネットアップのテクノロジーとソフトウェアの統合により、ソースシステムのSnapshotバックアップを利用してSAPシステムのコピーやクローニングを実行することができます。このSnapshotバックアップは、SAP本番用システムと同じストレージ、オフサイトのバックアップ用ストレージ、またはディザスタリカバリサイトのストレージから選択できます。この柔軟性により、必要に応じて開発用システムとテスト用システムを本番用システムと分離し、ディザスタリカバリサイトでのディザスタリカバリのテストなど、さまざまなシナリオに対応することができます。



オフサイトのバックアップストレージやディザスタリカバリストレージからのクローニングは、オンプレミスのNetAppシステムとAmazon FSx for NetApp ONTAPでサポートされます。Azure NetApp Filesを使用する場合、クローンはソースボリュームにのみ作成できます。

SAPテストシステムのプロビジョニングにはさまざまなシナリオとユースケースがあり、自動化のレベルに関してもさまざまな要件が存在する場合があります。ネットアップのSAP向けソフトウェア製品は、SAPのデータベースやライフサイクル管理製品と統合されているため、さまざまなシナリオや自動化のレベルに対応できます。

NetApp SnapCenter とSAP HANA向けプラグインを使用して、アプリケーションと整合性のあるSnapshotバックアップに基づいて必要なストレージボリュームをプロビジョニングし、開始されたSAP HANAデータベースまで、必要なホストおよびデータベースのすべての処理を実行します。SAPシステムのコピー、システムクローン、システムの更新、SAPのポストプロセスなどの追加の手動手順が必要になる場合があります。詳細については、次のセクションで説明します。

SAPテストシステムのプロビジョニングは、サードパーティのツールを使用してNetAppの機能を統合することで、完全に自動化されたエンドツーエンドで実行できます。詳細については、次のサイトを参照してください。

["Ansibleを使用したNetApp SAP Landscape Managementの統合"](#)

["Libelle SystemCopyによるSAPシステムのコピー操作の自動化 \(netapp.com\) "](#)

["AlpacaとNetApp SnapCenterを使用したSAPシステムのコピーとクローンワークフローの自動化"](#)

["AvantraとNetApp SnapCenterを使用したSAPシステムのコピーとクローンワークフローの自動化"](#)

SAPシステムのコピー、更新、クローニングのシナリオ

SAPシステムの更新、SAPシステムのコピー、SAPシステムのクローニングの3つのプロセスの同義語としてよく使用されます。ワークフローとユースケースはそれぞれ異なるため、異なる処理を区別することが重要です。

- * SAPシステムの更新。* SAPシステムの更新は、ソースSAPシステムのデータを使用した既存のターゲットSAPシステムの更新です。ターゲットシステムは、通常、品質管理システムなどのSAP転送環境の一部であり、本番用システムのデータで更新されます。ソース・システムとターゲット・システムでは、ホスト名、インスタンス番号、およびSIDが異なります。
- * SAPシステムのコピー。* SAPシステムのコピーとは、ソースSAPシステムのデータを含む新しいターゲットSAPシステムのセットアップを指します。たとえば、本番用システムのデータを含む追加のテストシステムを新しいターゲットシステムとして使用できます。ソース・システムとターゲット・システムでは、ホスト名、インスタンス番号、およびSIDが異なります。
- * SAPシステムのクローン。* SAPシステムのクローンは、ソースSAPシステムの同一クローンです。SAPシステムクローンは、一般に論理的な破損に対処したり、ディザスタリカバリのシナリオをテストしたりするのに使用されます。システムのクローニング処理では、ホスト名、インスタンス番号、およびSIDの値は変更されません。そのため、本番環境と通信できないように、ターゲットシステムの適切なネットワークフェンシングを設定することが重要です。

次の図は、システム更新、システムコピー、またはシステムクローンの処理中に実行する必要がある主な手順を示しています。青のボックスはSnapCenterで自動化できる手順を示し、グレーのボックスは手動またはサードパーティのツールを使用してSnapCenter以外で実行する必要がある手順を示します。

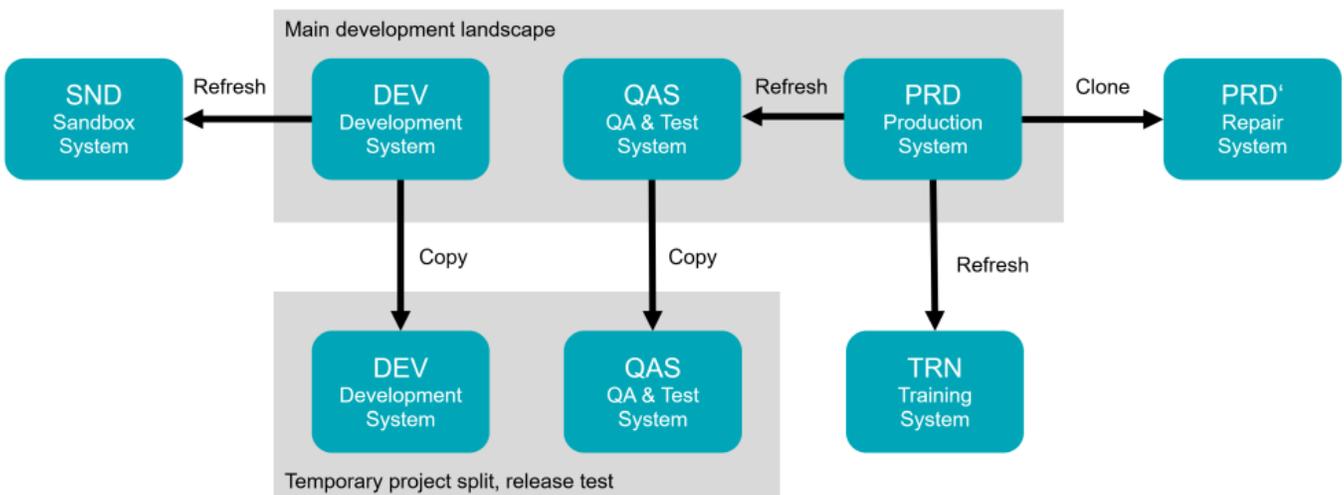


システムの更新とクローニングのユースケース

SAPのライフサイクル管理を最適化するNetAppソリューションは、SAP HANAデータベースとライフサイクル管理ツールに統合されており、効率的なアプリケーション統合データプロテクションと、SAPテストシステムの柔軟なプロビジョニングが組み合わされています。

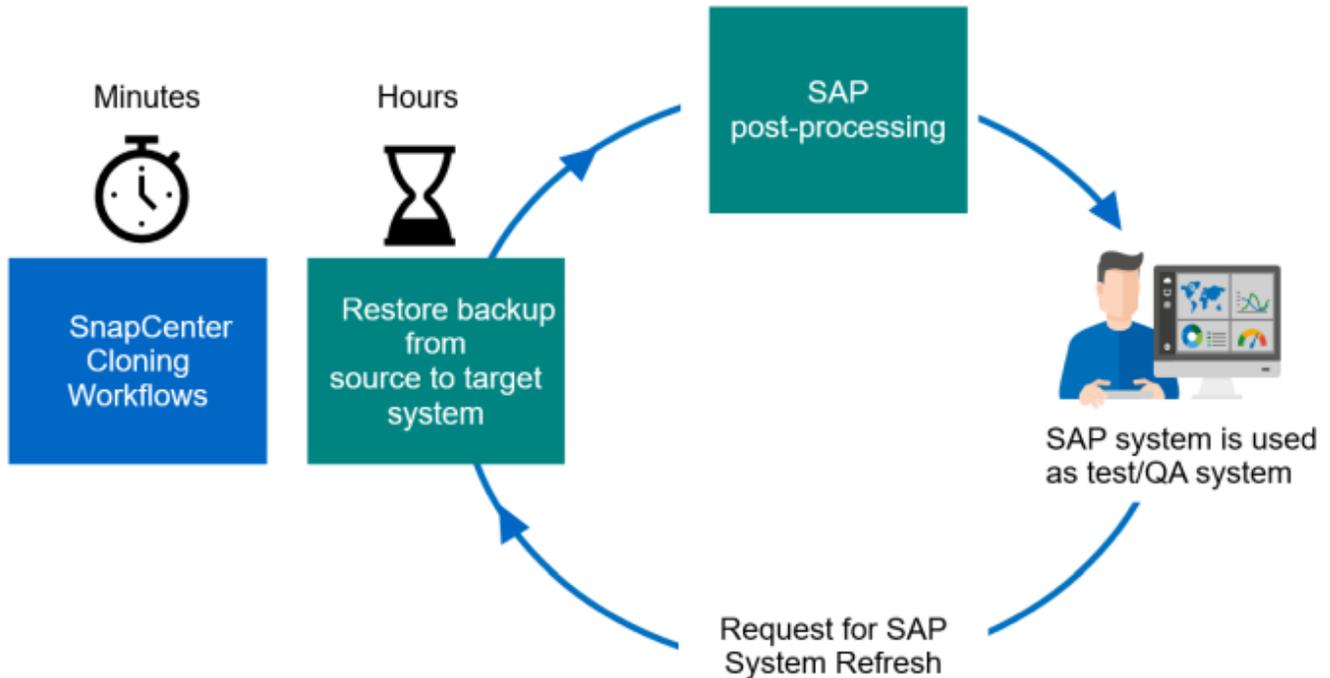
QA、テスト、サンドボックス、トレーニングシステムのデータ更新

テストやトレーニングの目的で、ソースシステムのデータをターゲットシステムで使用できるようにする必要があります。シナリオは複数あります。テストおよびトレーニング用のシステムは、ソースシステムのデータで定期的に更新し、現在のデータセットでテストとトレーニングが実行されていることを確認する必要があります。このシステム更新処理は、インフラ、データベース、アプリケーションの各レイヤ上で実行される複数のタスクで構成されます。自動化のレベルによっては、数日かかる場合があります。



SnapCenterクローニングのワークフローを使用すると、インフラレイヤとデータベースレイヤで必要なタスクを高速化、自動化できます。SnapCenterでは、ソースシステムからターゲットシステムにバックアップをリストアする代わりに、NetApp SnapshotコピーとNetApp FlexCloneテクノロジーを使用しているため、起動し

たSAP HANAデータベースに必要なタスクを数時間ではなく数分で実行できます。クローニングプロセスに要する時間はデータベースのサイズに左右されないため、非常に大規模なシステムでも数分で作成できます。起動時間は、データベースのサイズ、およびデータベースサーバとストレージシステム間の接続によって異なります。



システム更新処理のワークフローについては、を参照してください。"[「SnapCenter でSAP HANAシステムを更新」](#)"

論理的破損に対処する

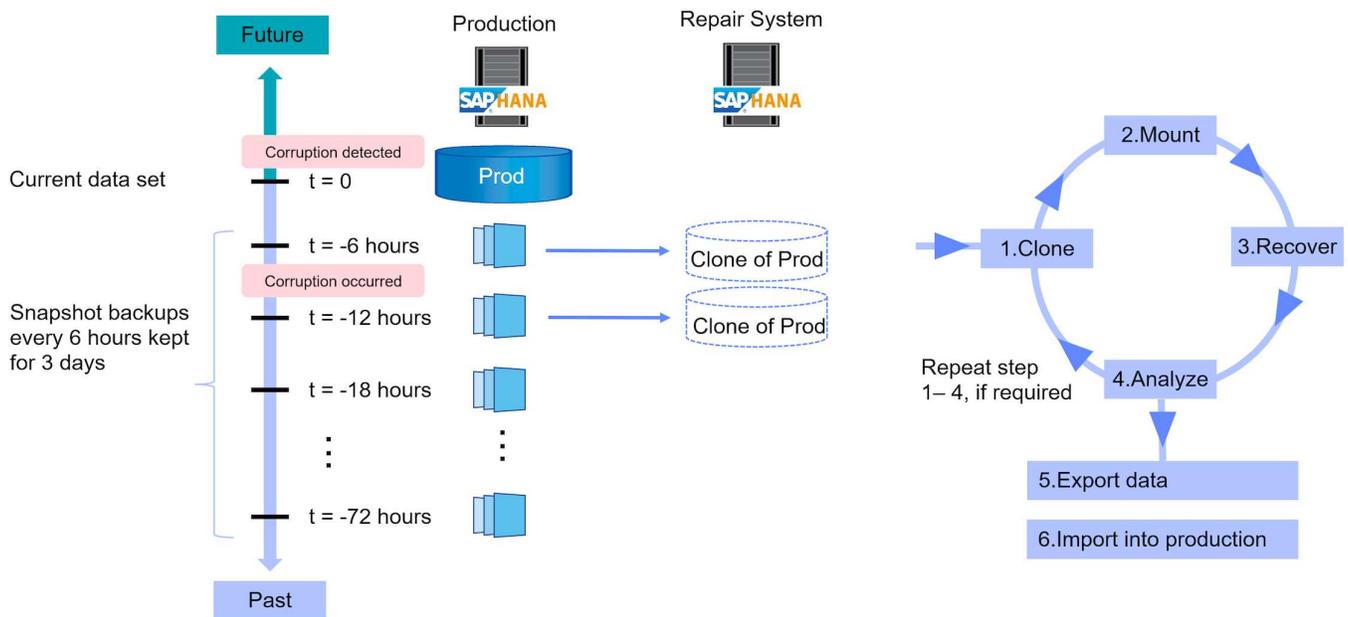
論理的な破損は、ソフトウェア エラー、人為的エラー、または妨害行為によって発生する可能性があります。残念ながら、論理的な破損は、標準的な高可用性および災害復旧ソリューションでは対処できないことがよくあります。その結果、論理破損が発生したレイヤー、アプリケーション、ファイル システム、またはストレージによっては、最大ダウンタイムとデータ損失の要件を満たせない場合があります。

最悪のケースは、SAPアプリケーションが論理的に破損した場合です。SAP アプリケーションは多くの場合、異なるアプリケーションが相互に通信してデータを交換する環境で動作します。このため、論理的な破損が発生した SAP システムはリストアとリカバリを実行しないことを推奨します。破損が発生する前の時点でシステムをリストアすると、データが失われます。また、SAP ランドスケープは同期されず、さらにポストプロセスが必要になります。

SAPシステムをリストアする代わりに、別の修復システムで問題を分析して、システム内の論理エラーを修正する方法を推奨します。ルート原因分析には、ビジネスプロセスやアプリケーション所有者の関与が必要です。このシナリオでは、論理的破損が発生する前に格納されたデータに基づいて、修復システム（本番システムのクローン）を作成します。リペアシステム内では、必要なデータをエクスポートし、本番システムにインポートできます。このアプローチでは、本番システムを停止する必要はなく、最良のシナリオでは、データの損失だけでなく、ごくわずかなデータの損失も発生します。

リペアシステムを設定する際には、柔軟性と俊敏性が重要です。NetAppのストレージベースのSnapshot/バックアップを使用すると、NetApp FlexCloneテクノロジーを使用して、複数の整合性のあるデータベースイメー

ジを使用して本番用システムのクローンを作成できます。ファイルベースのバックアップからリダイレクトされたリストアを使用して修復システムを設定する場合、FlexCloneボリュームは数時間ではなく数秒で作成できます。



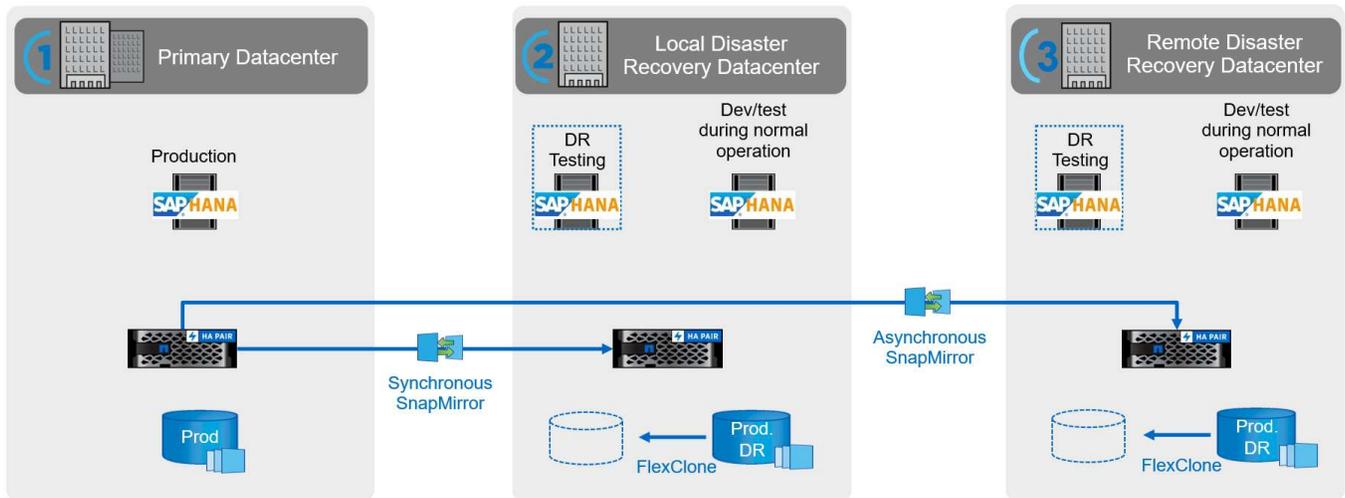
リペアシステムの作成ワークフローについては、のセクションを参照してください。"「SnapCenter を使用したSAPシステムのクローン作成」"

ディザスタリカバリのテスト

効果的なディザスタリカバリ戦略では、必要なワークフローをテストする必要があります。テストでは、戦略が機能するかどうか、および内部ドキュメントで十分かどうかを検証します。また、管理者は必要な手順をトレーニングすることもできます。

SnapMirrorを使用したストレージレプリケーションでは、RTOとRPOをリスクにさらすことなく、ディザスタリカバリのテストを実行できます。ディザスタリカバリテストは、データレプリケーションを中断することなく実行できます。

非同期SnapMirrorと同期SnapMirrorのディザスタリカバリテストでは、ディザスタリカバリターゲットでSnapshotバックアップとFlexCloneボリュームを使用します。



詳細な手順については、テクニカルレポートを参照してください。

"SAP HANA Disaster Recovery with Storage Replication』を参照してください"

"Azure NetApp Filesを使用したSAP HANA向けディザスタリカバリ"

サポートされるインフラとシナリオ

このドキュメントでは、オンプレミスのNetAppシステム、Amazon FSx for NetApp ONTAPシステム、およびAzure NetApp Filesで実行されるSAP HANAシステムの更新とクローニングのシナリオについて説明します。ただし、すべてのストレージプラットフォームですべての機能とシナリオを使用できるわけではありません。次の表は、サポートされる構成をまとめたものです。

このドキュメントでは、オンプレミスのNetAppシステムで実行されるSAP HANAランドスケープを使用し、NFSをストレージプロトコルとして使用しています。ワークフローの手順のほとんどは、プラットフォームごとに同じです。違いがある場合は、このドキュメントで説明します。

	オンプレミスのNetAppシステム	* AWS FSx for NetApp ONTAP *	* Azure NetApp Files *
ストレージプロトコルとファイルシステム	ベアメタル：NFS、XFS を使用したFC VMware： ゲスト内マウントの NFS、VMDKおよびXFS を使用したFC	NFS	NFS
シンクローン (FlexClone)	はい。	はい。	はい。ただしSnapCenter には統合されていま せん。
クローンスプリット処理	はい。	はい。	はい。ただしSnapCenter には統合されていま せん。
プライマリカラノクロー ニク	はい。	はい。	はい。

	オンプレミスのNetAppシステム	* AWS FSx for NetApp ONTAP *	* Azure NetApp Files *
オフサイトバックアップからのクローニング	はい。	はい。	はい、ANFバックアップから新しいボリュームにリストアします
DRサイトでのクローニング	はい。	はい。	はい。ただしSnapCenterには統合されていません。

VMware VMFSおよびNetApp ASAr2システムを使用したHANAのSAPシステムリフレッシュ操作に関する追加情報については、"[SAP システムリフレッシュ](#)"を参照してください。

SnapCenter を使用したSAPシステムの更新ワークフローの概要

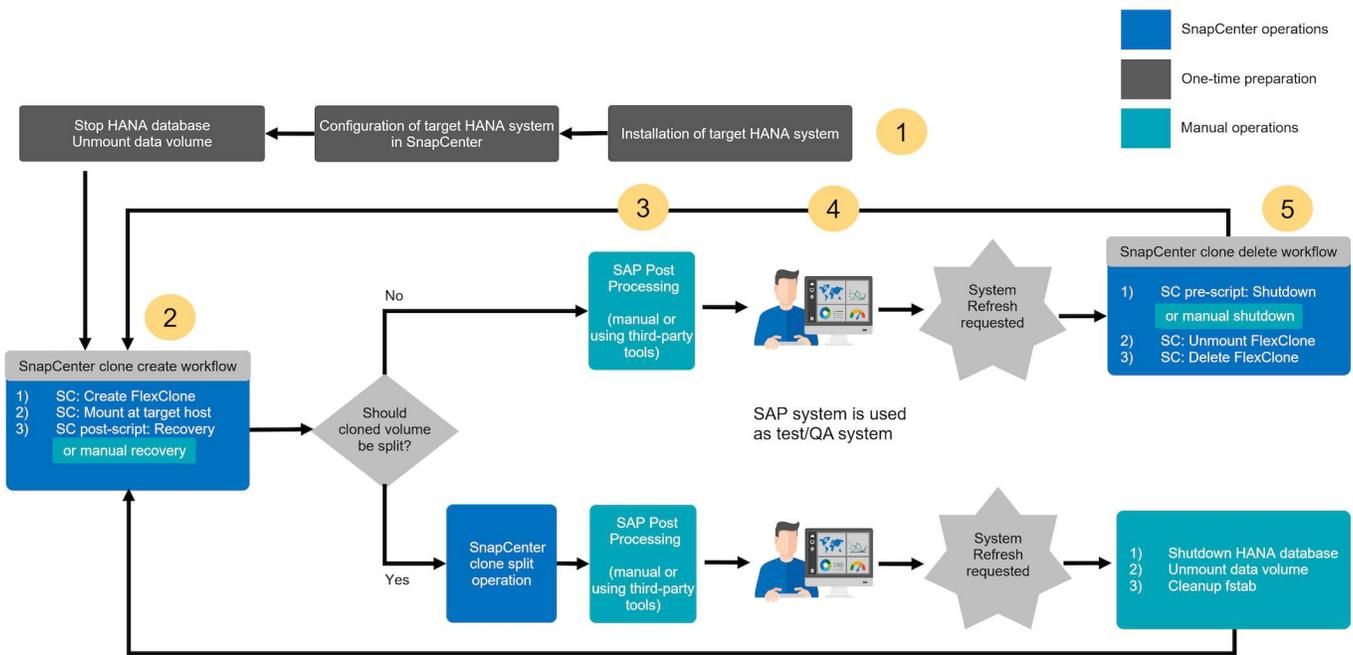
SnapCenterには、既存のSnapshotバックアップからデータセットのクローンを管理するワークフローが用意されています。このクローンデータセットはFlexCloneボリュームで、ソースシステムからHANAデータボリュームを迅速にプロビジョニングしてターゲットシステムに接続するために使用できます。したがって、QA、テスト、サンドボックス、またはトレーニングシステムのシステム更新処理を実行する場合に最適です。

SnapCenter クローニングワークフローは、ストレージレイヤで必要なすべての処理を処理します。また、スクリプトを使用して拡張し、ホスト固有の処理とHANAデータベース固有の処理を実行することもできます。本ドキュメントでは、スクリプトを使用して、HANAデータベースのリカバリ処理とシャットダウン処理を実行します。SnapCenter ワークフローにスクリプトを使用してさらに自動化を進めることで、必要なHANAデータベースの処理はすべて処理されますが、SAPの後処理の手順については説明しません。SAPの後処理は、手動またはサードパーティのツールを使用して実行する必要があります。

SnapCenterを使用したSAPシステムの更新ワークフローは、次の図に示すように、主に5つの手順で構成されます。

1. ターゲットシステムの初回インストールと準備
 - a. 新しいターゲットシステムにSnapCenter HANAプラグインをインストールし、SnapCenterでHANAシステムを設定する必要があります。
 - b. ターゲットシステムを停止し、HANAデータボリュームをアンマウントする必要があります。
2. SnapCenterクローン作成ワークフロー
 - a. SnapCenterがソースシステムの選択したSnapshotのFlexCloneボリュームを作成
 - b. SnapCenterがターゲットシステムにFlexCloneボリュームをマウントする
 - c. ターゲットのHANAデータベースのリカバリは、スクリプトをポストスクリプトとして使用して自動化することも、手動で実行することもできます `sc-system-refresh`。
3. SAPの後処理（手動またはサードパーティのツールを使用）
4. これで、システムをテスト/QAシステムとして使用できるようになります。
5. 新しいシステム更新が要求されると、SnapCenterクローン削除ワークフローを使用してFlexCloneボリュームが削除されます。
 - a. ターゲットのHANAシステムがSnapCenterで保護されている場合は、クローン削除ワークフローを開始する前に保護を削除する必要があります。

- b. HANAシステムは手動で停止するか、SnapCenterのプリスクリプトとしてスクリプトを使用して自動的に停止する必要があります。 `sc-system-refresh`
- c. SnapCenterがHANAデータボリュームをアンマウントする
- d. SnapCenterがFlexCloneボリュームを削除
- e. ステップ2でリフレッシュが再開されます。



ほとんどの場合、ターゲットテスト/QAシステムは少なくとも数週間使用されます。FlexCloneボリュームがソースシステムボリュームのSnapshotをブロックしているため、このSnapshotには、ソースシステムボリュームでのブロック変更率に基づいて追加の容量が必要になります。本番ソースシステムで1日あたりの平均変更率が20%の場合、ブロックされたSnapshotは5日後に100%に達します。そのため、NetAppでは、クローンが本番用のソースシステムに基づいている場合は、FlexCloneボリュームをすぐにスプリットするか、数日後にスプリットすることを推奨しています。クローンスプリット処理ではクローンボリュームの使用はブロックされないため、HANAデータベースの使用中にいつでも実行できます。



FlexCloneボリュームをスプリットすると、SnapCenter はターゲットシステムに作成されたすべてのバックアップを削除します。



SnapCenterとAzure NetApp Filesでは、クローン スプリット処理は使用できません。クローンは作成後に常に分割されます。

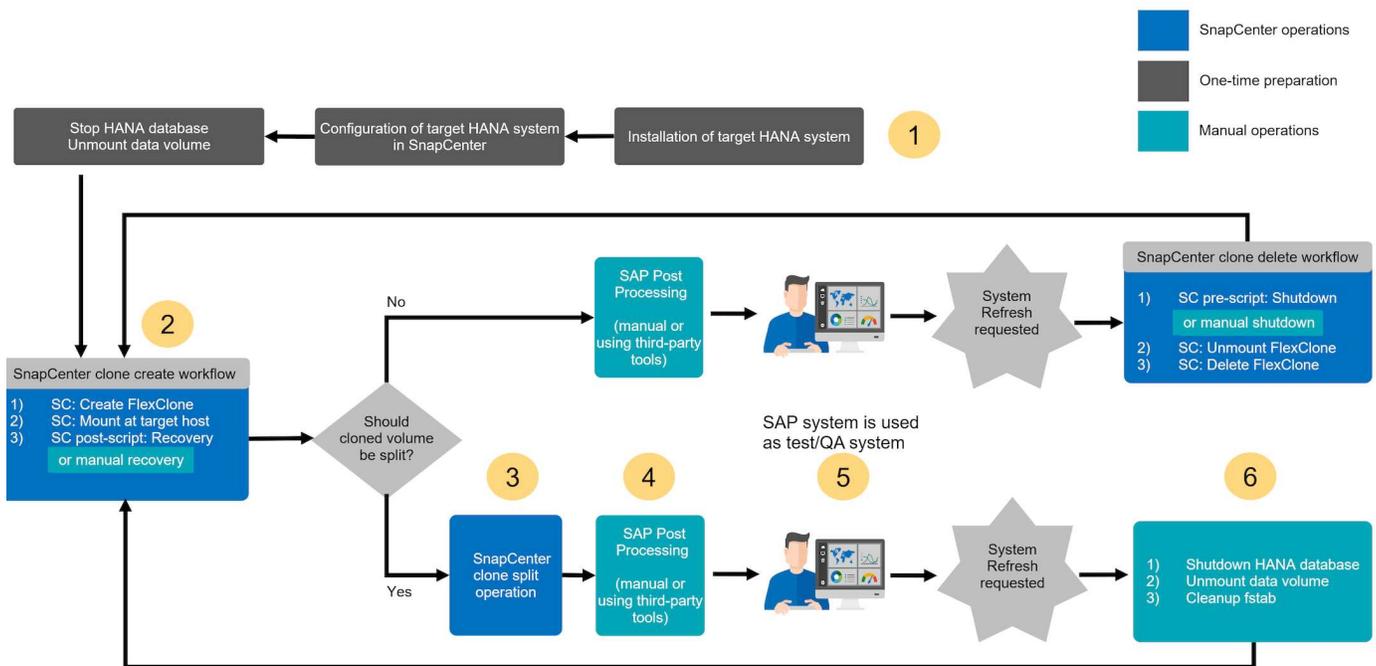
クローンスプリットを含む更新処理は、次の手順で構成されます。

1. ターゲットシステムの初回インストールと準備
 - a. 新しいターゲットシステムにSnapCenter HANAプラグインをインストールし、SnapCenterでHANAシステムを設定する必要があります。
 - b. ターゲットシステムを停止し、HANAデータボリュームをアンマウントする必要があります。
2. SnapCenterクローン作成ワークフロー

- a. SnapCenterがソースシステムの選択したSnapshotのFlexCloneボリュームを作成
 - b. SnapCenterがターゲットシステムにFlexCloneボリュームをマウントする
 - c. ターゲットのHANAデータベースのリカバリは、スクリプトをポストスクリプトとして使用して自動化することも、手動で実行することもできます `sc-system-refresh`。
3. SnapCenterクローンスプリットワークフローを使用してFlexCloneボリュームをスプリットします。
 4. SAPの後処理（手動またはサードパーティのツールを使用）
 5. これで、システムをテスト/QAシステムとして使用できるようになります。
 6. 新しいシステムの更新が要求されると、クリーンアップは次の手順で実行されます。
 - a. ターゲットのHANAシステムがSnapCenterで保護されている場合は、保護を削除する必要があります。
 - b. HANAシステムを手動で停止する必要がある
 - c. HANAデータボリュームをアンマウントし、SnapCenterからfstabエントリを削除する（手動タスク）
 - d. ステップ2でリフレッシュが再開されます。



以前にスプリットされていた古いデータボリュームは、ストレージシステムで手動で削除する必要があります。



このセクションでは「[SnapCenterでSAP HANAシステムを更新](#)」、両方のシステム更新ワークフローについて、詳細なステップバイステップで説明します。

SnapCenter を使用したSAPシステムのクローニングワークフローの概要

前のセクションで説明したように、SnapCenter は、任意の既存のSnapshotバックアップからデータセットのクローンを管理し、これらのデータセットを任意のターゲットシステムに迅速にプロビジョニングできます。論理的な破損に対処するために本番データ

をリペアシステムに柔軟かつ迅速にプロビジョニングすることは、リペアシステムをリセットして別の本番データセットを選択する必要があることが多いため、非常に重要です。FlexCloneテクノロジーを使用すると、修復システムは通常短時間しか使用されないため、高速なプロビジョニングプロセスが可能になり、容量を大幅に削減できます。

次の図は、SnapCenterを使用したSAPシステムのクローニング処理に必要な手順をまとめたものです。

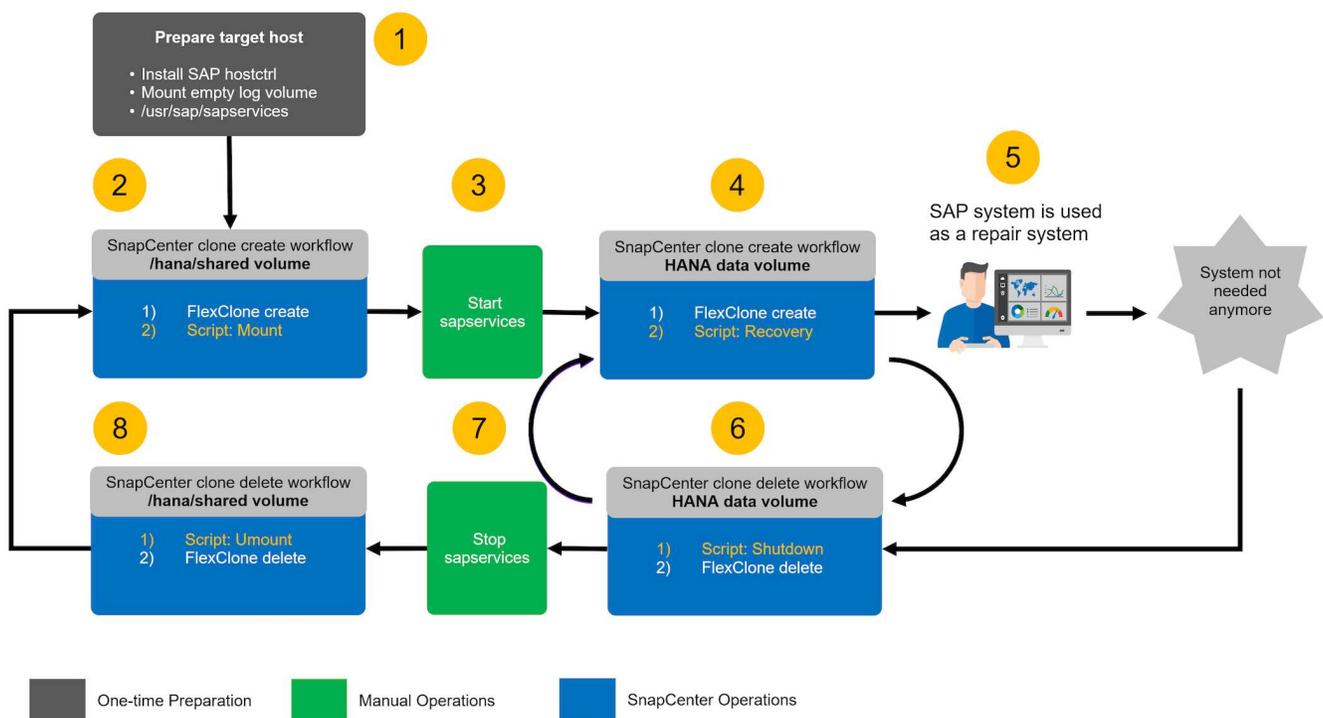
1. ターゲットホストを準備します。
2. SAP HANA共有ボリューム用のSnapCenterクローン作成ワークフロー
3. SAP HANAサービスを開始します。
4. SnapCenterクローンは、SAP HANAデータボリュームのデータベースリカバリを含むワークフローを作成します。
5. SAP HANAシステムを修復システムとして使用できるようになりました。

システムが不要になった場合は、次の手順でクリーンアッププロセスを実行します。

1. データベースのシャットダウンを含む、SAP HANAデータボリュームのSnapCenterクローン削除ワークフロー（自動スクリプト使用時）。
2. SAP HANAサービスを停止します。
3. SAP HANA共有ボリュームのSnapCenterクローン削除ワークフロー



システムを別のSnapshotバックアップにリセットする必要がある場合は、手順6と手順4で十分です。SAP HANA共有ボリュームの更新は必要ありません。



「」セクションでは「[SnapCenterを使用したSAPシステムのクローン作成](#)」、システムクローニングワークフローの詳細な手順を説明します。

ストレージSnapshotバックアップを使用したSAP HANAシステムの更新処理に関する考慮事項

SAPのライフサイクル管理を最適化するNetAppソリューションは、SAP HANAデータベースとライフサイクル管理ツールに統合されており、効率的なアプリケーション統合データプロテクションと、SAPテストシステムの柔軟なプロビジョニングが組み合わされています。

ターゲットシステムのテナント名

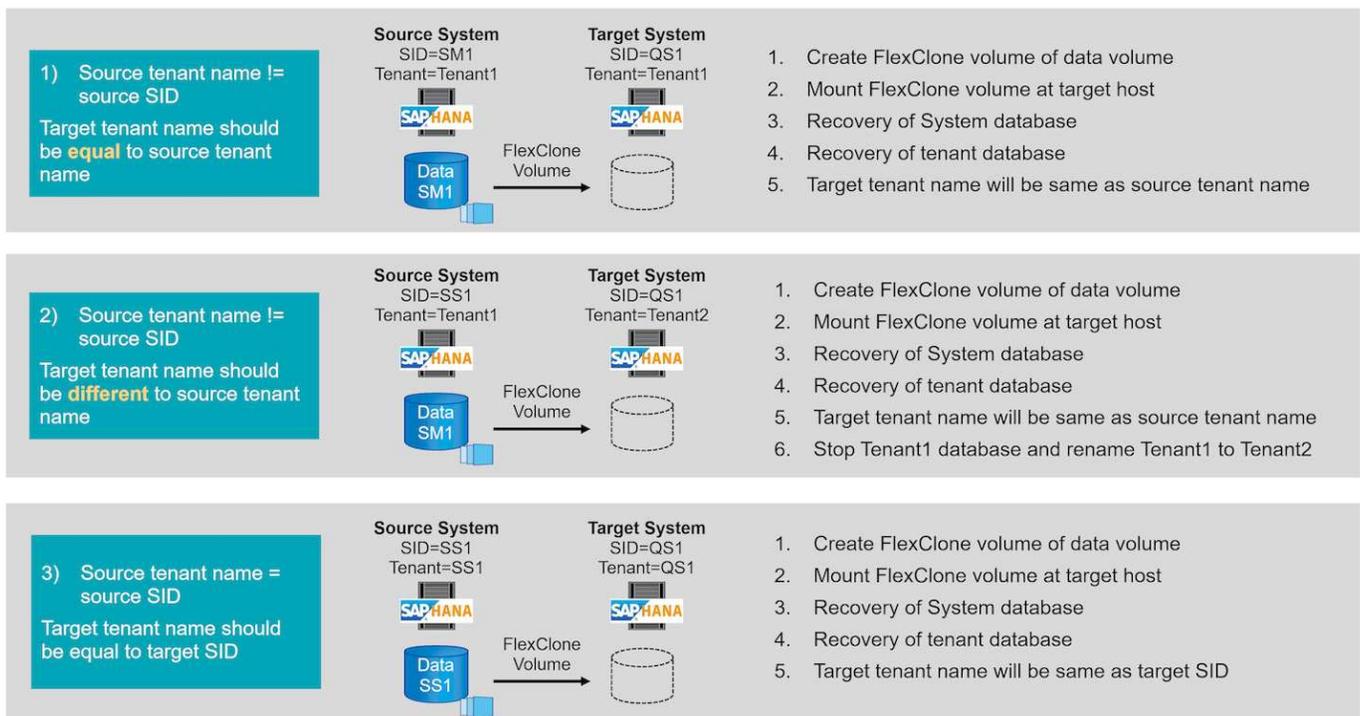
SAP HANAシステムの更新を実行する手順は、次の図に示すように、ソースシステムのテナント構成とターゲットシステムで必要なテナント名によって異なります。

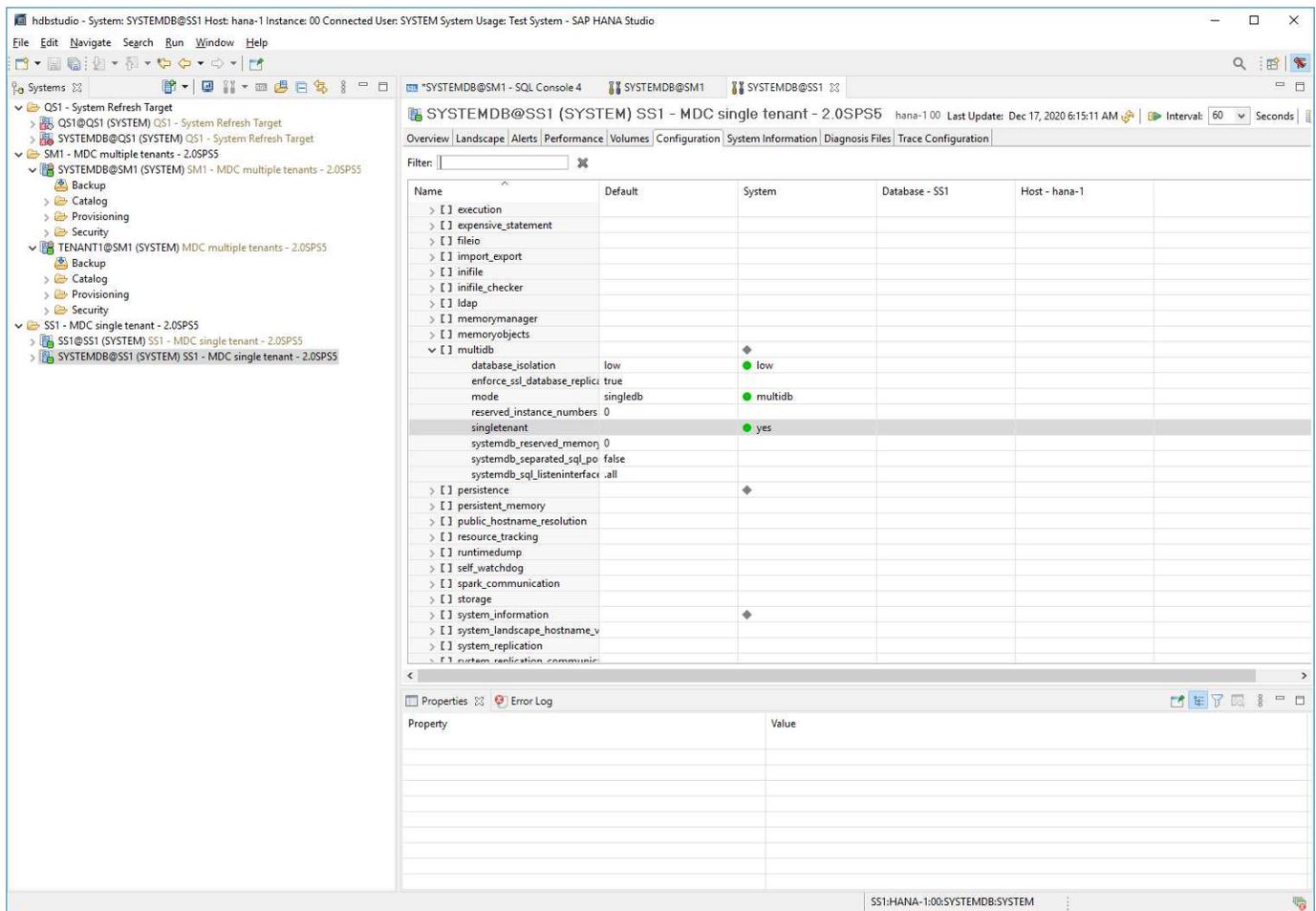
システムデータベースにはテナント名が設定されているため、システムデータベースのリカバリ後にターゲットシステムでもソースシステムのテナント名を使用できます。そのため、ターゲットシステムのテナントは、オプション1に示すように、ソーステナントと同じ名前ではリカバリできません。ターゲットシステムでテナント名を異なる名前にする必要がある場合は、最初にソーステナントと同じ名前でテナントをリカバリし、必要なターゲットテナント名に変更する必要があります。これはオプション2です。

ただし、テナント名がシステムのSIDと同じであるシングルテナントのSAP HANAシステムは例外です。これは、SAP HANAの初回インストール後のデフォルトの設定です。この特定の構成は、SAP HANAデータベースによってフラグ付けされます。この場合、ターゲットシステムでのテナントリカバリは、ターゲットシステムのテナント名を使用して実行できます。この名前は、ターゲットシステムのシステムSIDと同じである必要もあります。このワークフローをオプション3に示します。



この設定フラグは、テナントの作成、名前変更、削除処理がソースシステムで実行されるとすぐにSAP HANAデータベースによって削除されます。したがって、構成をTenant = SIDの状態に戻しても、フラグは使用できなくなり、ワークフロー3でのテナントのリカバリに関する例外は実行できなくなります。この場合、必要なワークフローはオプション2です。





SAP HANAの暗号化を有効にしたシステム更新ワークフロー

SAP HANA 永続暗号化が有効になっていて、暗号化ルート キーに使用されるセキュア ストア（インスタンス SFSS またはローカル セキュア ストア LSS）に応じて、ターゲット システムで SAP HANA データベースを回復する前に追加の手順が必要になります。

SFSS では、ルート キーはファイル システム内に保存されるため、HANA データベースに加えて個別にバックアップする必要があります。HANA データベースの復旧を可能にするには、ルート キーのバックアップをターゲット システムにインポートする必要があります。ルート キーが変更されていない場合は、これを 1 回だけ実行する必要があります。

LSSを使用すると、ルートキーはHANAデータベースとともに1回の操作でバックアップおよびリストアされます。HANAデータベースをリカバリするには、ルートキーのバックアップに使用されたパスワードが必要です。

必要な操作については、以降の章で詳しく説明します。

セキュア ストア SFSS

ソースシステムで、システムデータベースおよびすべてのテナントデータベースの暗号化ルートキーのバックアップを作成する必要があります。リカバリ処理を実行する前に、バックアップファイルをターゲットシステムにコピーし、ルートキーをバックアップからインポートする必要があります。

も参照してください "『SAP HANA Administration Guide』をご覧ください"。

ルートキーのバックアップ

ルートキーに変更があった場合は、常にルートキーのバックアップが必要です。backupコマンドでは、CLIパラメータとしてDBIDを指定する必要があります。DBIDは、次のSQLステートメントを使用して識別できます。



The screenshot shows a SQL console window with the following query and result:

```
SELECT DATABASE_NAME,  
CASE WHEN (DBID = " AND  
DATABASE_NAME = 'SYSTEMDB')  
THEN 1  
WHEN (DBID = " AND  
DATABASE_NAME <> 'SYSTEMDB')  
THEN 3  
ELSE TO_INT(DBID)  
END DATABASE_ID  
FROM (SELECT DISTINCT DATABASE_NAME, SUBSTR_AFTER (SUBPATH, '.') AS DBID FROM SYS_DATABASES.M_VOLUMES)
```

	DATABASE_NAME	DATABASE_ID
1	SYSTEMDB	1
2	SS1	3

SQLステートメントおよびその他のドキュメントについては、『SAP HANA Admin Guide』を参照してください。"ルートキーのバックアップ| SAPヘルプポータル" 以下の手順は、シングルテナントSS1を使用するHANAシステムに必要な処理を示しており、ソースシステムで実行されます。

1. システムおよびテナント (SS1) データベースのバックアップパスワードを設定します (まだ設定していない場合)。

```
hdbsql SYSTEMDB=> ALTER SYSTEM SET ENCRYPTION ROOT KEYS BACKUP PASSWORD  
Netapp123;  
0 rows affected (overall time 3658.128 msec; server time 3657.967 msec)  
hdbsql SYSTEMDB=>  
hdbsql SS1=> ALTER SYSTEM SET ENCRYPTION ROOT KEYS BACKUP PASSWORD  
Netapp123;  
0 rows affected (overall time 2424.236 msec; server time 2424.010 msec)  
hdbsql SS1=>
```

1. システムおよびテナント (SS1) データベースのルートキーのバックアップを作成します。

```
ssladm@hana-1:/usr/sap/SS1/home> /usr/sap/SS1/HDB00/exe/hdbnsutil
-backupRootKeys root-key-backup-SS1-SYSTEMDB.rkb --dbid=1 --type='ALL'
Exporting root key backup for database SYSTEMDB (DBID: 1) to
/usr/sap/SS1/home/root-key-backup-SS1-SYSTEMDB.rkb
done.
ssladm@hana-1:/usr/sap/SS1/home> /usr/sap/SS1/HDB00/exe/hdbnsutil
-backupRootKeys root-key-backup-SS1-SS1.rkb --dbid=3 --type='ALL'
Exporting root key backup for database SS1 (DBID: 3) to
/usr/sap/SS1/home/root-key-backup-SS1-SS1.rkb
done.
```

1. ルートキーのバックアップの検証 (オプション)

```
ssladm@hana-1:/usr/sap/SS1/home> ls -al root*
-rw-r----- 1 ssladm sapsys 1440 Apr 24 07:00 root-key-backup-SS1-SS1.rkb
-rw-r----- 1 ssladm sapsys 1440 Apr 24 06:54 root-key-backup-SS1-
SYSTEMDB.rkb
ssladm@hana-1:/usr/sap/SS1/home>

ssladm@hana-1:/usr/sap/SS1/home> /usr/sap/SS1/HDB00/exe/hdbnsutil
-validateRootKeysBackup root-key-backup-SS1-SS1.rkb
Please Enter the password:
Successfully validated SSFS backup file /usr/sap/SS1/home/root-key-backup-
SS1-SS1.rkb
done.

ssladm@hana-1:/usr/sap/SS1/home> /usr/sap/SS1/HDB00/exe/hdbnsutil
-validateRootKeysBackup root-key-backup-SS1-SYSTEMDB.rkb
Please Enter the password:
Successfully validated SSFS backup file /usr/sap/SS1/home/root-key-backup-
SS1-SYSTEMDB.rkb
done.
```

ターゲットシステムでのルートキーのインポート

最初のシステム更新処理では、最初にルートキーをインポートする必要があります。ソースシステムでルートキーが変更されていない場合は、追加のインポートは必要ありません。importコマンドでは、CLIパラメータとしてDBIDを指定する必要があります。DBIDは、ルートキーのバックアップと同じ方法で識別できます。

1. このセットアップでは、ルートキーがソースシステムからNFS共有にコピーされます。

```
hana-1:~ # cp /usr/sap/SS1/home/root-key-backup-SS1-SS1.rkb /mnt/sapcc-
share/SAP-System-Refresh/
hana-1:~ # cp /usr/sap/SS1/home/root-key-backup-SS1-SYSTEMDB.rkb
/mnt/sapcc-share/SAP-System-Refresh/
```

1. これで、hdbnsutilを使用してルートキーをインポートできるようになりました。コマンドでは、システムおよびテナントデータベースのDBIDを指定する必要があります。バックアップパスワードも必要です。

```
qsladm@hana-7:/usr/sap/QS1/HDB11> ./exe/hdbnsutil -recoverRootKeys
/mnt/sapcc-share/SAP-System-Refresh/root-key-backup-SS1-SYSTEMDB.rkb
--dbid=1 --type=ALL
Please Enter the password:
Importing root keys for DBID: 1 from /mnt/sapcc-share/SAP-System-
Refresh/root-key-backup-SS1-SYSTEMDB.rkb
Successfully imported root keys from /mnt/sapcc-share/SAP-System-
Refresh/root-key-backup-SS1-SYSTEMDB.rkb
done.

qsladm@hana-7:/usr/sap/QS1/HDB11> ./exe/hdbnsutil -recoverRootKeys
/mnt/sapcc-share/SAP-System-Refresh/root-key-backup-SS1-SS1.rkb --dbid=3
--type=ALL Please Enter the password:
Importing root keys for DBID: 3 from /mnt/sapcc-share/SAP-System-
Refresh/root-key-backup-SS1-SS1.rkb
Successfully imported root keys from /mnt/sapcc-share/SAP-System-
Refresh/root-key-backup-SS1-SS1.rkb
done.
qsladm@hana-7:/usr/sap/QS1/HDB11>
```

ルートキーのインポート（ターゲットにDBIDが存在しない場合）

前の章で説明したように、DBIDは、システムおよびすべてのテナントデータベースのルートキーをインポートするために必要です。システムデータベースでは常にDBID=0ですが、テナントデータベースでは異なるDBIDを使用できます。

SQL
Result

```

SELECT DATABASE_NAME,
CASE WHEN (DBID = " AND
DATABASE_NAME = 'SYSTEMDB')
THEN 1
WHEN (DBID = " AND
DATABASE_NAME <> 'SYSTEMDB')
THEN 3
ELSE TO_INT(DBID)
END DATABASE_ID
FROM (SELECT DISTINCT DATABASE_NAME, SUBSTR_AFTER (SUBPATH,':') AS DBID FROM SYS_DATABASES.M_VOLUMES)

```

	DATABASE_NAME	DATABASE_ID
1	TENANT1	4
2	SYSTEMDB	1
3	TENANT2	3

上の出力は、DBID=3、DBID=4の2つのテナントを示しています。ターゲットシステムがDBID=4のテナントをまだホストしていない場合、ルートキーのインポートは失敗します。その場合は、まずシステムデータベースをリカバリしてから、DBID=4のテナントのキーをインポートする必要があります。

セキュア ストア LSS



少なくとも HANA バージョン 2.00.088 を使用する必要があります。そうでない場合、説明されている手順は機能しません。SAP ノート 3613110 も参照してください。

暗号化と LSS を使用する HANA システムを復旧するには、ルート キーのバックアップに使用されたパスワードが必要です。

SQL文

```
alter system set encryption root keys backup password <backup-pw-of-source-system>;
```

ターゲット システムの LSS 構成でソース システムのパスワードを設定するために使用できます。システムデータベースとテナント データベースには同じステートメントを使用する必要があります。

事前にパスワードが設定されていない場合は、リカバリ処理中にパスワードを入力できます。

システムデータベース：

```
./exe/Python/bin/python  
/usr/sap/QM1/HDB22/exe/python_support/recoverSys.py --command "RECOVER  
DATA USING SNAPSHOT CLEAR LOG ENCRYPTION ROOT KEYS BACKUP PASSWORD  
'<backup-pw-of-source-system>'"
```

テナントデータベース：

```
hdbsql SYSTEMDB=> RECOVER DATA for QM1 USING SNAPSHOT CLEAR LOG ENCRYPTION  
ROOT KEYS BACKUP PASSWORD '<backup-pw-of-source-system>';
```

自動化スクリプトの例

本ドキュメントでは、SnapCenter クローンの作成処理とクローンの削除処理をさらに自動化するために、2つのスクリプトを使用しています。

- script.shは sc-system-refresh、システム更新とシステムクローニングのワークフローでSAP HANAデータベースのリカバリ処理とシャットダウン処理を実行するために使用します。
- このスクリプトは sc-mount-volume.sh、システムクローンワークフローでSAP HANA共有ボリュームのマウントとアンマウントの処理を実行するために使用します。



このサンプルスクリプトは現状のまま提供されており、ネットアップではサポートしていません。スクリプトは、ng-sapcc@netapp.com宛てにEメールでリクエストできます。

スクリプトsc-system-refresh.sh

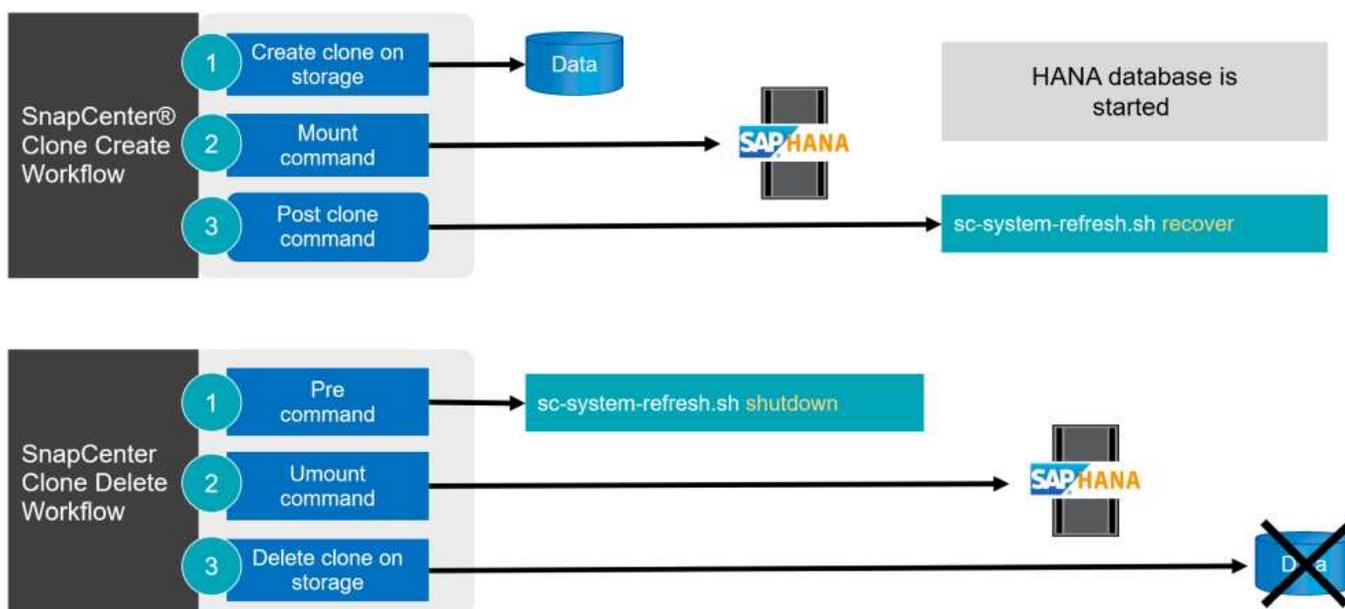
このスクリプト例は `sc-system-refresh.sh`、リカバリ処理とシャットダウン処理を実行するために使用します。このスクリプトは、次の図に示すように、SnapCenterワークフローのclone createおよびclone deleteで特定のコマンドラインオプションを指定して呼び出されます。

このスクリプトは汎用的で、SIDなどの必要なすべてのパラメータをターゲットシステムから読み取ります。このスクリプトは、システム更新処理のターゲットホストで使用できる必要があります。ターゲットシステムのユーザ<SID> admに対してhdbユーザストアキーが設定されている必要があります。このキーには、SAP HANAシステムデータベースへのアクセスを許可し、リカバリ処理の権限を付与する必要があります。キーには<TARGET-SID>キーという名前が必要です。

スクリプトは、実行されるディレクトリと同じディレクトリにログファイルを書き込み ``sc-system-refresh-SID.log`` ます。



現在のバージョンのスクリプトは、シングルホストシステムMDCのシングルテナント構成、またはMDCのマルチテナント構成をサポートしています。SAP HANAマルチホストシステムはサポートされていません。



サポートされているテナントリカバリ処理

セクション「ストレージSnapshotを使用したSAP HANAシステムの更新処理のワークフロー」で説明したように、ターゲットシステムで想定されるテナントリカバリ処理は、ソースシステムのテナント構成によって異なります。このスクリプト `sc-system-refresh.sh` では、次の表に示すように、すべてのテナントリカバリ処理がサポートされます。これらの処理はソースシステムの構成によって可能です。

ターゲットシステムで別のテナント名が必要な場合は、リカバリ処理後にテナントの名前を手動で変更する必要があります。

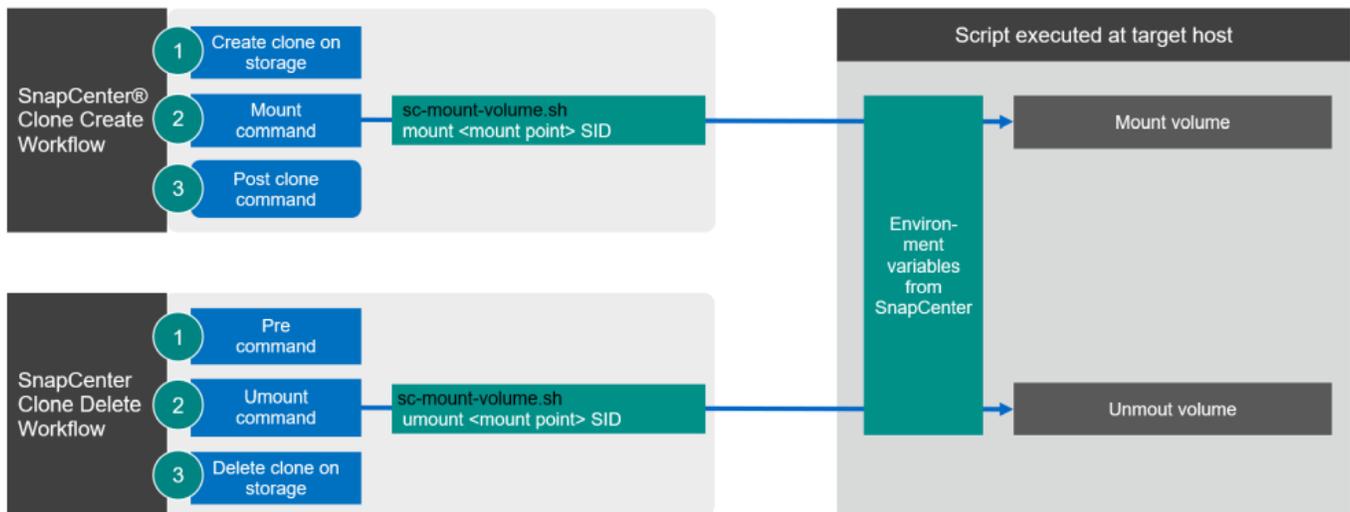
SAP HANAシステム	テナント構成+ソースシステム	変更後のテナント構成+をターゲットシステムに配置
MDCのシングルテナント	ソースのテナント名がソースのSIDと同じです	ターゲットテナント名はターゲットSIDと同じです
MDCのシングルテナント	ソーステナント名がソースSIDと同じではありません	ターゲットテナント名はソーステナント名と同じです
MDCの複数のテナント	任意のテナント名	すべてのテナントがリカバリされ、ソーステナントと同じ名前になります。

スクリプト `sc-mount-volume.sh`

このスクリプト例で `sc-mount-volume.sh` は、任意のボリュームのマウントとアンマウントを実行しています。このスクリプトは、SAP HANAシステムのクローニング処理でSAP HANA共有ボリュームをマウントするために使用されます。このスクリプトは、次の図に示すように、SnapCenterワークフローのclone create およびclone deleteで特定のコマンドラインオプションを指定して呼び出されます。



このスクリプトは、NFSをストレージプロトコルとして使用するSAP HANAシステムをサポートします。



SnapCenter 環境変数

SnapCenter には、ターゲットホストで実行されるスクリプト内で使用可能な一連の環境変数が用意されています。スクリプトはこれらの変数を使用して、関連する構成設定を決定します。

- スクリプト変数 `STORAGE`, `JUNCTION_PATH` はマウント処理に使用されます。
- 「`cloned_volume_mount_path`」環境変数から派生したものです。
- `CLONED_VOLUMES_MOUNT_PATH=${STORAGE}:/${JUNCTION_PATH}`
- 例：
`CLONED_VOLUMES_MOUNT_PATH=192.168.175.117:/SS1_shared_Clone_05112206115489411`

SnapCenter 環境変数を取得するスクリプト

自動化スクリプトを使用せず、手順を手動で実行する必要がある場合は、FlexCloneボリュームのストレージシステムのジャンクションパスを把握しておく必要があります。ジャンクションパスはSnapCenter 内では表示されないため、ストレージシステムで直接ジャンクションパスを検索するか、ターゲットホストにSnapCenter 環境変数を提供する単純なスクリプトを使用する必要があります。このスクリプトは、SnapCenter のクローン作成処理にマウント処理スクリプトとして追加する必要があります。

```
ssladm@hana-1:/mnt/sapcc-share/SAP-System-Refresh> cat get-env.sh
#!/bin/bash
env > /tmp/env-from-sc.txt
ssladm@hana-1:/mnt/sapcc-share/SAP-System-Refresh>
```

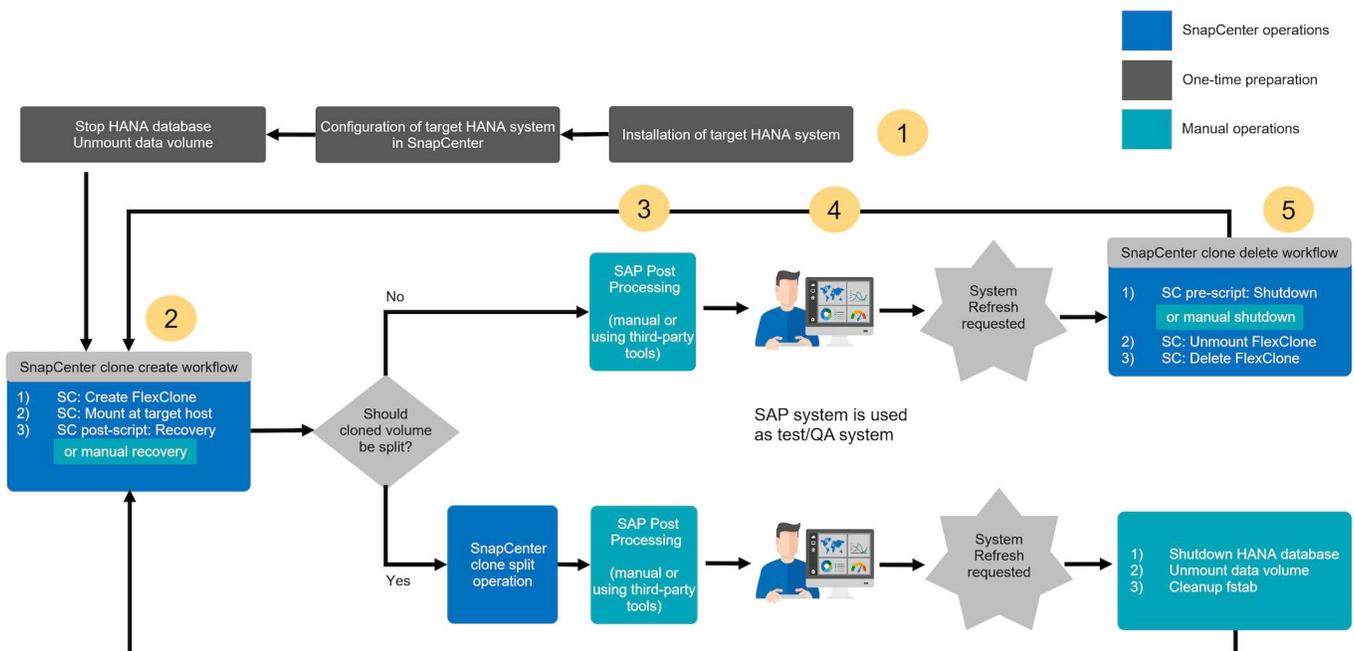
「env-from-ssc.txt」ファイル内で、変数「cloned_volume_mount_path」を検索して、FlexCloneボリュームのストレージシステムのIPアドレスとジャンクションパスを取得します。

例：

```
CLONED_VOLUMES_MOUNT_PATH=192.168.175.117:/SS1_data_mnt00001_Clone_05112206115489
411
```

SAP HANAシステムをSnapCenter で更新

次のセクションでは、SAP HANAデータベースのさまざまなシステム更新処理オプションについて、ステップバイステップの概要を示します。



SAP HANAデータベースの設定に応じて、追加の手順を実行するか、準備が必要です。次の表に概要を示します。

ソースシステム	ソースシステムの構成	SnapCenterとSAP HANAの運用
MDCシングルテナント+ SID =テナント名	標準構成	SnapCenterクローン処理とリカバリスクリプトの実行（オプション）。
	SAP HANAの永続性暗号化	SFSS：最初に、またはソース システムでルート キーが変更された場合は、リカバリを実行する前に、ターゲット システムにルート キーのバックアップをインポートする必要があります。LSS：最初に、またはルート キーのバックアップ パスワードが変更された場合は、LSS 構成でパスワードを設定する必要があります。
	SAP HANAシステムレプリケーションのソース	追加の手順は必要ありません。ターゲットシステムにHSRが設定されていない場合は、スタンドアロンシステムのままになります。
	SAP HANAマルチパーティション	追加の手順は必要ありませんが、ターゲットシステムに同じ命名規則（SIDのみが異なります）を使用してSAP HANAボリュームパーティションのマウントポイントを使用できる必要があります。
MDCの複数のテナント またはMDCシングルテナント+（SID <>テナント名）	標準構成	SnapCenterクローン処理とリカバリスクリプトの実行（オプション）。スクリプトによってすべてのテナントがリカバリされます。ターゲットシステム名にテナント名またはテナント名が存在しない場合は、SAP HANAのリカバリ処理中に必要なディレクトリが自動的に作成されます。テナント名はソースと同じであり、必要に応じてリカバリ後に名前を変更する必要があります。
	SAP HANAの永続性暗号化	SFSS：ソース システムの DBID がターゲットシステムにまだ存在しない場合は、このテナントのルート キー バックアップをインポートする前に、まずシステム データベースを回復する必要があります。LSS：LSS トポロジは、テナント データベースの回復中に HANA トポロジと自動的に同期されます。
	HANAシステムレプリケーションのソース	追加の手順は必要ありません。ターゲットシステムにHSRが設定されていない場合は、スタンドアロンシステムのままになります。
	HANAマルチパーティション	追加の手順は必要ありませんが、ターゲットシステムに同じ命名規則（SIDのみが異なります）を使用してSAP HANAボリュームパーティションのマウントポイントを使用できる必要があります。

このセクションでは、次のシナリオについて説明します。

- クローンスプリット処理を実行せずにSAP HANAシステムが更新されます。
- テナント名がSIDと同じであるプライマリストレージからのクローニング

- オフサイトのバックアップストレージからのクローニング
- 複数テナントのプライマリストレージからのクローニング
- クローンの削除処理
- SAP HANAシステムがクローンスプリット処理で更新されます
- テナント名がSIDと同じであるプライマリストレージからのクローニング
- クローンスプリット処理

前提条件および制限事項

以降のセクションで説明するワークフローには、SAP HANAシステムのアーキテクチャとSnapCenterの設定に関する前提条件と制限事項がいくつかあります。

- ここで説明するワークフローは、SnapCenter 5.0リリース以降でのみ有効です。
- ここで説明するワークフローは、シングルテナントまたは複数テナントのシングルホストSAP HANA MDCシステムに対して有効です。SAP HANAマルチホストシステムは対象外です。
- SnapCenterの自動検出と自動スクリプトの実行を有効にするには、SnapCenter SAP HANAプラグインをターゲットホストに導入する必要があります。
- ワークフローは、物理ホストでNFSまたはFCPを使用するSAP HANAシステム、またはゲスト内NFSマウントを使用する仮想ホストに有効です。

ラボのセットアップ

次の図は、さまざまなシステム更新操作オプションに使用したラボのセットアップを示しています。

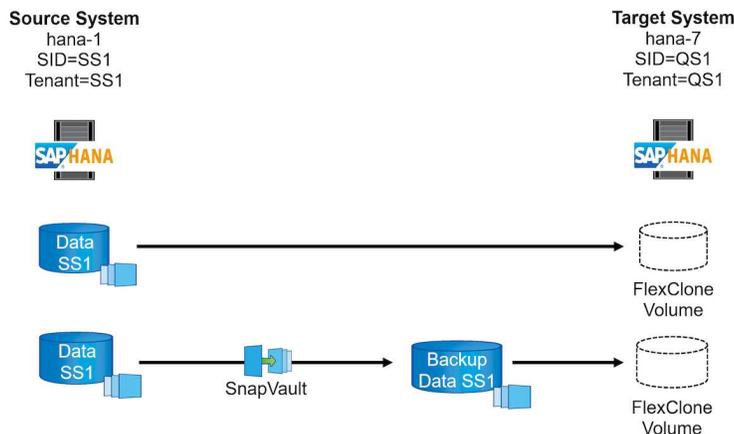
- プライマリストレージまたはオフサイトバックアップストレージからのクローニング。テナント名はSIDと同じです。
 - ソースSAP HANAシステム：SS1とテナントSS1
 - ターゲットのSAP HANAシステム：QS1とテナントQS1
- プライマリストレージからのクローニング（複数テナント）。
 - ソースSAP HANAシステム：SM1とTenant1およびTenant2
 - ターゲットのSAP HANAシステム：QS1とTenant1およびTenant2

使用したソフトウェアバージョンは次のとおりです。

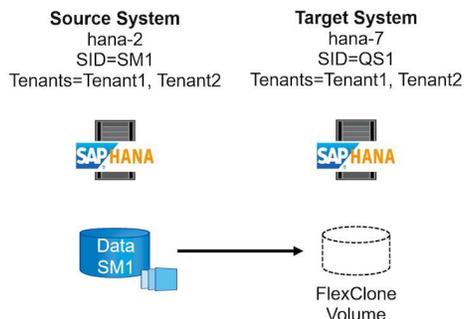
- SnapCenter 5.0
- SAP HANAシステム：HANA 2.0 SPS7 rev.73
- SLES 15
- ONTAP 9.14P1

すべてのSAP HANAシステムは、構成ガイドに基づいて構成する必要があります。"[NFSを使用したNetApp AFF システムでのSAP HANA](#)"。SnapCenterとSAP HANAリソースはベストプラクティスガイドに基づいて構成されました。"[SnapCenterを使用したSAP HANAのバックアップとリカバリ](#)"。

Cloning from primary or offsite backup storage, Tenant name = SID



Cloning from primary storage, Tenant name != SID



最初の1回限りの準備手順

最初のステップとして、ターゲットのSAP HANAシステムがSnapCenter内で設定されている必要があります。

1. SAP HANAターゲットシステムのインストール
2. SnapCenterでのSAP HANAシステムの構成は、"[SnapCenterを使用したSAP HANAのバックアップとリカバリ](#)"
 - a. SnapCenterバックアップ処理用のSAP HANAデータベースユーザの設定このユーザは、ソースシステムとターゲットシステムで同一である必要があります。
 - b. 上記のバックアップユーザを使用した<sid> admのhdbuserstoreキーの設定。リカバリに自動スクリプトを使用する場合は、キー名を<SID>キーにする必要があります。
 - c. SnapCenter SAP HANAプラグインをターゲットホストに導入SAP HANAシステムは、SnapCenterによって自動検出されます。
 - d. SAP HANAリソース保護の設定（オプション）

初期インストールの準備が完了してから、次の手順で最初のSAPシステムの更新処理を実行します。

1. ターゲットのSAP HANAシステムをシャットダウン
2. SAP HANAデータボリュームをアンマウントします。

ターゲットシステムで実行するスクリプトを、SnapCenter allowed commands configファイルに追加する必要があります。

```
hana-7:/opt/NetApp/snapcenter/scc/etc # cat
/opt/NetApp/snapcenter/scc/etc/allowed_commands.config
command: mount
command: umount
command: /mnt/sapcc-share/SAP-System-Refresh/sc-system-refresh.sh
hana-7:/opt/NetApp/snapcenter/scc/etc #
```

テナント名をSIDと同じにしてプライマリストレージからクローニングする

ここでは、SAP HANAシステムの更新ワークフローについて説明します。このワークフローでは、ソースシステムとターゲットシステムのテナント名がSIDと同じになります。ストレージのクローニングはプライマリストレージで実行され、スクリプトによってリカバリが自動化され`sc-system-refresh.sh`ます。



このワークフローは、次の手順で構成されます。

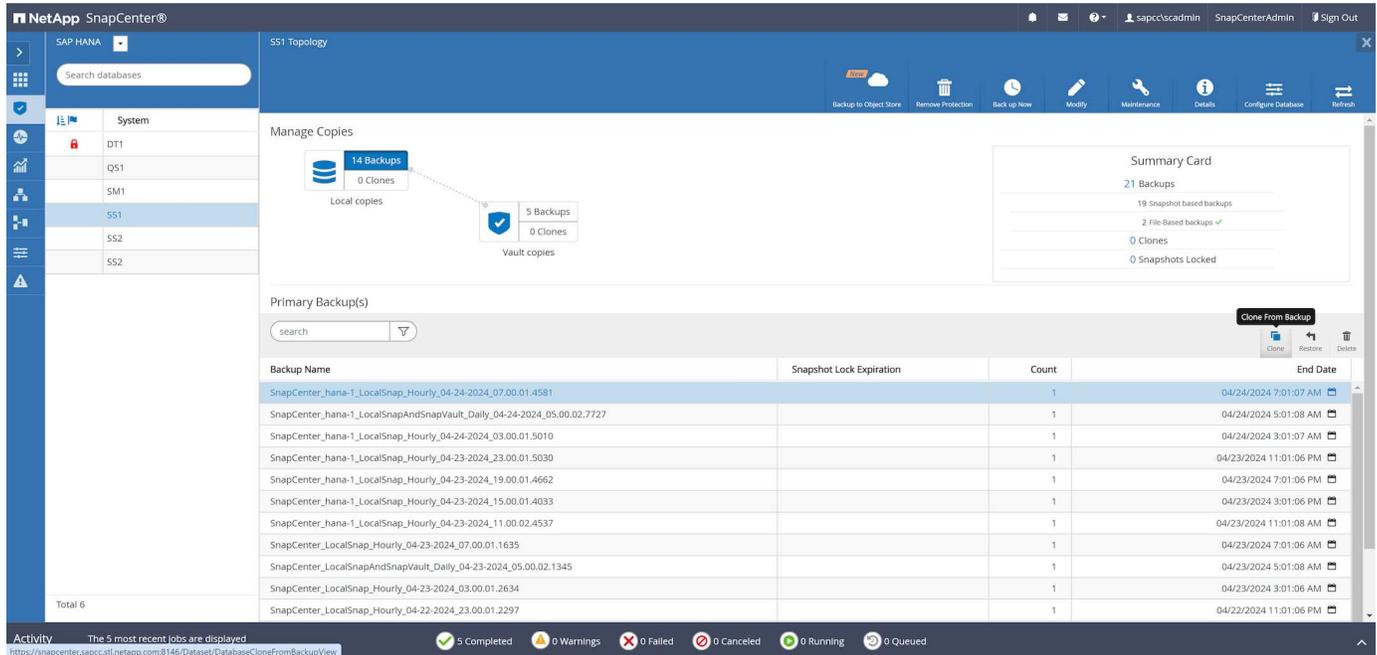
1. ソースシステムでSAP HANA永続性暗号化が有効になっている場合は、暗号化ルートキーを1回インポートする必要があります。ソースシステムでキーが変更されている場合もインポートが必要です。章を参照 "[「ストレージSnapshotバックアップを使用したSAP HANAシステムの更新処理に関する考慮事項」](#)"
2. ターゲットのSAP HANAシステムがSnapCenterで保護されている場合は、まず保護を削除する必要があります。
3. SnapCenter クローンの作成ワークフロー
 - a. ソースSAP HANAシステムSS1から[Snapshot backup]を選択します。
 - b. ターゲットホストを選択し、ターゲットホストのストレージネットワークインターフェイスを指定してください。
 - c. ターゲットシステムのSIDを指定します（この例ではQS1）。
 - d. 必要に応じて、クローン後の処理としてリカバリ用のスクリプトを指定します。
4. SnapCenter クローニング処理。
 - a. ソースSAP HANAシステムの選択したSnapshotバックアップに基づいてFlexCloneボリュームを作成します。
 - b. FlexCloneボリュームをターゲットホストストレージのネットワークインターフェイスまたはigroupにエクスポートします。
 - c. のマウント処理を実行します。FlexCloneボリュームをターゲットホストにマウントします。
 - d. クローニング後処理のリカバリスクリプトを実行します（前に設定した場合）。それ以外の場合には、SnapCenterワークフローが終了したときにリカバリを手動で実行する必要があります。
 - システムデータベースのリカバリ。

- テナント名= QS1でのテナントデータベースのリカバリ

5. 必要に応じて、ターゲットのSAP HANAリソースをSnapCenterで保護します。

以下のスクリーンショットは、必要な手順を示しています。

1. ソースシステムSS1からSnapshotバックアップを選択し、[Clone]をクリックします。



1. ターゲットシステムQS1がインストールされているホストを選択します。ターゲットSIDとして「QS1」と入力します。NFSエクスポートのIPアドレスは、ターゲットホストのストレージネットワークインターフェイスである必要があります。



入力するターゲットSIDによって、SnapCenterによるクローンリソースの管理方法が制御されます。ターゲットSIDのリソースがすでにSnapCenterで設定されており、プラグインホストと一致する場合、SnapCenterはクローンをこのリソースに割り当てます。ターゲットホストでSIDが設定されていない場合、SnapCenterは新しいリソースを作成します。



クローニングのワークフローを開始する前に、ターゲットシステムのリソースとホストをSnapCenterで設定しておくことが重要です。そうしないと、SnapCenterで作成された新しいリソースでは自動検出がサポートされず、説明されているワークフローは機能しません。

×

Clone From Backup

- 1 Location
- 2 Scripts
- 3 Notification
- 4 Summary

Select the host to create the clone

Plug-in host ⓘ

Target Clone SID ⓘ

NFS Export IP Address ⓘ

ファイバチャネル SAN セットアップでは、次の画面で使用するプロトコルを指定する必要があります。



スクリーンショットは、ファイバチャネル接続を使用した別のラボセットアップを示しています。

×

Clone From Backup

- 1 Location
- 2 Settings
- 3 Scripts
- 4 Notification
- 5 Summary

Select the host to create the clone

Plug-in host ⓘ

Target Clone SID ⓘ

×

Clone From Backup

- 1 Location
- 2 Settings
- 3 Scripts
- 4 Notification
- 5 Summary

LUN Map Settings

igroup protocol ⓘ

Azure NetApp Filesと手動のQoS容量プールを使用している場合は、新しいボリュームのスループットを最大化する必要があります。容量プールに十分なヘッドルームがあることを確認してください。そうしないと、クローニングワークフローが失敗します。



スクリーンショットは、Azure NetApp Filesを使用したMicrosoft Azureで実行される別のラボセットアップを示しています。

Clone From Backup
✕

1 Location

2 Scripts

3 Notification

4 Summary

Select the host to create the clone

Plug-in host ⓘ

Target Clone SID ⓘ

NFS Export IP Address ⓘ

Capacity Pool Max. Throughput (MIB/s) ⓘ

- 必要なコマンドラインオプションを指定して、オプションのクローニング後スクリプトを入力します。この例では、クローニング後のスクリプトを使用してSAP HANAデータベースのリカバリを実行します。

Clone From Backup
✕

1 Location

2 Scripts

3 Notification

4 Summary

The following commands will run on the Plug-in Host: hana-7.sapcc.stl.netapp.com

Enter optional commands to run before performing a clone operation ⓘ

Pre clone command

Enter optional commands to run after performing a clone operation ⓘ

Post clone command

ⓘ 前述したように、リカバリスクリプトの使用はオプションです。SnapCenterクローニングのワークフローが終了したあとに、手動でリカバリを実行することもできます。

ⓘ リカバリ処理用スクリプトは、ログのクリア処理を使用してSAP HANAデータベースをSnapshotのポイントインタイムにリカバリし、フォワードリカバリは実行しません。特定の時点までのフォワードリカバリが必要な場合は、リカバリを手動で実行する必要があります。手動フォワードリカバリでは、ソースシステムのログバックアップをターゲットホストで利用できることも必要です。

- SnapCenter の[ジョブの詳細]画面に、処理の進捗状況が表示されます。ジョブの詳細には、データベースリカバリを含めた全体的な実行時間が3分未満であることも示されています。

Job Details ✕

Clone from backup 'SnapCenter_hana-1_LocalSnap_Hourly_04-25-2024_11.00.01.5630'

- ✓ ▾ Clone from backup 'SnapCenter_hana-1_LocalSnap_Hourly_04-25-2024_11.00.01.5630'
- ✓ ▾ hana-7.sapcc.stl.netapp.com
- ✓ ▾ Clone
- ✓ ▶ Application Pre Clone
- ✓ ▶ Storage Clone
- ✓ ▶ Unmount Filesystem
- ✓ ▶ Mount Filesystem
- ✓ ▶ Application Post Clone
- ✓ ▶ Post Clone Create Commands
- ✓ ▶ Register Clone Metadata
- ✓ ▶ Clean-up Snapshot entries on Server
- ✓ ▶ Application Clean-Up
- ✓ ▶ Data Collection
- ✓ ▶ Agent Finalize Workflow

i Task Name: Clone Start Time: 04/25/2024 11:22:40 AM End Time: 04/25/2024 11:25:29 AM

View Logs
Cancel Job
Close

1. スクリプトのログファイル `sc-system-refresh` には、リカバリ処理で実行されたさまざまなステップが表示されます。このスクリプトは、システムデータベースからテナントのリストを読み取り、既存のすべてのテナントのリカバリを実行します。

```

20240425112328###hana-7###sc-system-refresh.sh: Script version: 3.0
hana-7:/mnt/sapcc-share/SAP-System-Refresh # cat sap-system-refresh-QS1.log
20240425112328###hana-7###sc-system-refresh.sh: *****
Starting script: recovery operation *****
20240425112328###hana-7###sc-system-refresh.sh: Recover system database.

```

```

20240425112328###hana-7###sc-system-refresh.sh:
/usr/sap/QS1/HDB11/exe/Python/bin/python
/usr/sap/QS1/HDB11/exe/python_support/recoverSys.py --command "RECOVER
DATA USING SNAPSHOT CLEAR LOG"
20240425112346###hana-7###sc-system-refresh.sh: Wait until SAP HANA
database is started ....
20240425112347###hana-7###sc-system-refresh.sh: Status: YELLOW
20240425112357###hana-7###sc-system-refresh.sh: Status: YELLOW
20240425112407###hana-7###sc-system-refresh.sh: Status: YELLOW
20240425112417###hana-7###sc-system-refresh.sh: Status: YELLOW
20240425112428###hana-7###sc-system-refresh.sh: Status: YELLOW
20240425112438###hana-7###sc-system-refresh.sh: Status: YELLOW
20240425112448###hana-7###sc-system-refresh.sh: Status: GREEN
20240425112448###hana-7###sc-system-refresh.sh: HANA system database
started.
20240425112448###hana-7###sc-system-refresh.sh: Checking connection to
system database.
20240425112448###hana-7###sc-system-refresh.sh:
/usr/sap/QS1/SYS/exe/hdb/hdbsql -U QS1KEY 'select * from sys.m_databases;'
DATABASE_NAME,DESCRIPTION,ACTIVE_STATUS,ACTIVE_STATUS_DETAILS,OS_USER,OS_G
ROUP,RESTART_MODE,FALLBACK_SNAPSHOT_CREATE_TIME
"SYSTEMDB","SystemDB-QS1-11","YES","","","","DEFAULT",?
"QS1","QS1-11","NO","ACTIVE","","","DEFAULT",?
2 rows selected (overall time 16.225 msec; server time 860 usec)
20240425112448###hana-7###sc-system-refresh.sh: Successfully connected to
system database.
20240425112449###hana-7###sc-system-refresh.sh: Tenant databases to
recover: QS1
20240425112449###hana-7###sc-system-refresh.sh: Found inactive
tenants(QS1) and starting recovery
20240425112449###hana-7###sc-system-refresh.sh: Recover tenant database
QS1.
20240425112449###hana-7###sc-system-refresh.sh:
/usr/sap/QS1/SYS/exe/hdb/hdbsql -U QS1KEY RECOVER DATA FOR QS1 USING
SNAPSHOT CLEAR LOG
0 rows affected (overall time 22.138599 sec; server time 22.136268 sec)
20240425112511###hana-7###sc-system-refresh.sh: Checking availability of
Indexserver for tenant QS1.
20240425112511###hana-7###sc-system-refresh.sh: Recovery of tenant
database QS1 succesfully finished.
20240425112511###hana-7###sc-system-refresh.sh: Status: GREEN
20240425112511###hana-7###sc-system-refresh.sh: *****
Finished script: recovery operation *****
hana-7:/mnt/sapcc-share/SAP-System-Refresh

```

1. SnapCenter ジョブが完了すると、ソースシステムのトポロジビューにクローンが表示されます。

The screenshot displays the NetApp SnapCenter interface for SAP HANA. On the left, a sidebar lists systems: DT1, QS1, SM1, SS1, and SS2. The main area is titled 'SS1 Topology' and shows 'Manage Copies' with 14 Backups and 1 Clone in Local copies, and 5 Backups and 0 Clones in Vault copies. A 'Summary Card' on the right provides backup statistics: 21 Backups (19 Snapshot based, 2 File-Based), 1 Clone, and 0 Snapshots Locked. Below this is a table for 'Primary Clone(s)' with columns for Clone SID, Clone Host, Clone Name, Start Date, and End date. One clone is listed with SID QS1 and host hana-7.sapcc.stl.netapp.com. An activity bar at the bottom shows job status: 1 Completed, 2 Warnings, 0 Failed, 0 Canceled, 2 Running, and 0 Queued.

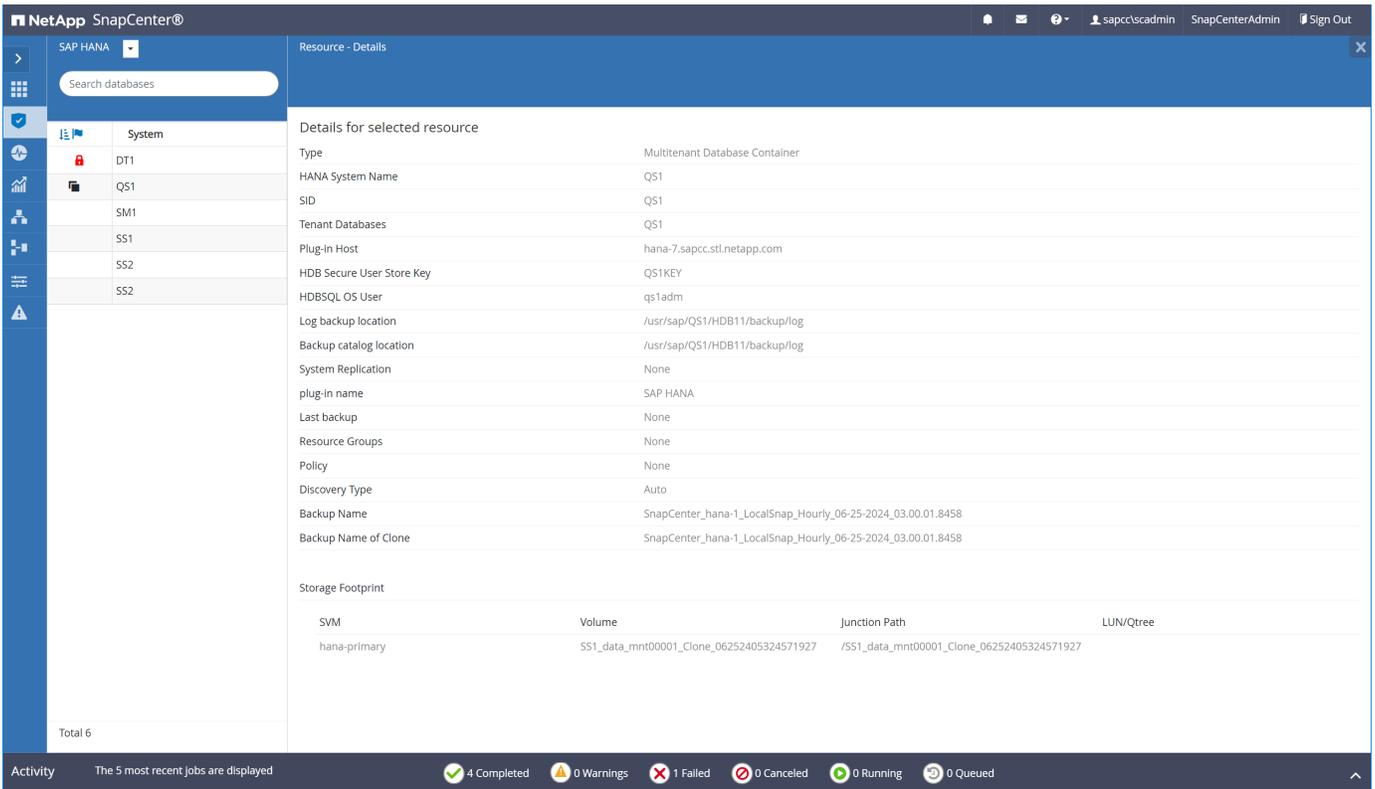
1. SAP HANAデータベースが実行されます。
2. ターゲットのSAP HANAシステムを保護する場合は、ターゲットシステムのリソースをクリックして自動検出を実行する必要があります。

The 'Configure Database' dialog box is shown with the following configuration:

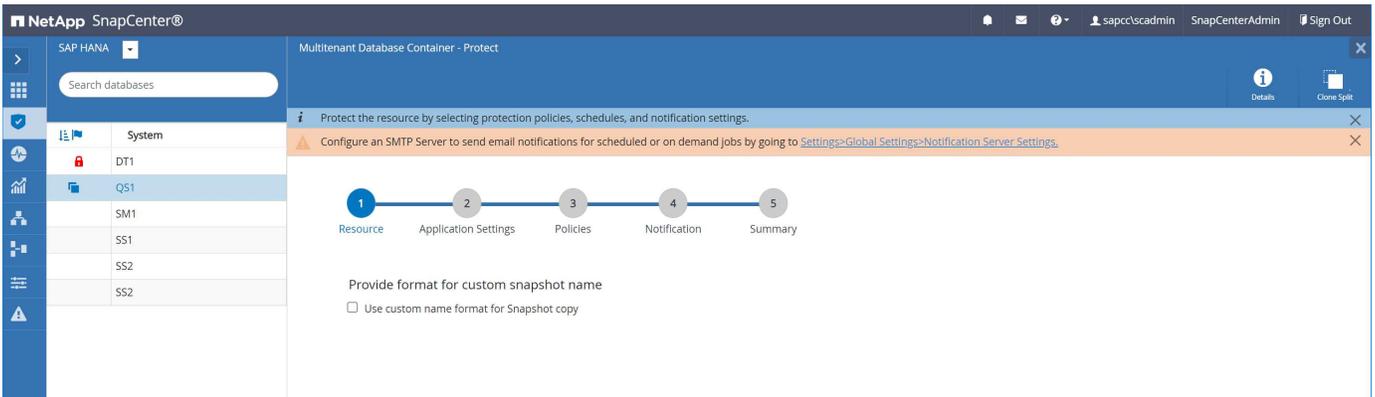
- Plug-in host: hana-7.sapcc.stl.netapp.com
- HDBSQL OS User: qs1adm
- HDB Secure User Store Key: QS1KEY

Buttons for 'Cancel' and 'OK' are visible at the bottom right of the dialog.

自動検出プロセスが完了すると、新しいクローンボリュームがストレージフットプリントセクションに表示されます。



リソースを再度クリックすると、更新したQS1システムのデータ保護を設定できます。



オフサイトのバックアップストレージからのクローニング

ここでは、ソースシステムとターゲットシステムのテナント名がSIDと同じであるSAP HANAシステムの更新ワークフローについて説明します。ストレージのクローニングはオフサイトのバックアップストレージで実行され、スクリプトsc-system-refresh.shを使用してさらに自動化されます。

Source System

hana-1
SID=SS1
Tenant=SS1



SnapVault



Target System

hana-7
SID=QS1
Tenant=QS1



FlexClone
Volume

プライマリとオフサイトのバックアップストレージのクローニングでSAP HANAシステムの更新ワークフローが異なるのは、SnapCenterでSnapshotバックアップを選択することだけです。オフサイトのバックアップストレージのクローニングでは、まずセカンダリバックアップを選択し、次にSnapshotバックアップを選択する必要があります。

Backup Name	Count	IF	End Date
SnapCenter_LocalSnapAndSnapVault_Daily_05-11-2022_05.00.02.9288	1		05/11/2022 5:01:01 AM
SnapCenter_LocalSnapAndSnapVault_Daily_05-10-2022_05.00.02.9444	1		05/10/2022 5:01:01 AM
SnapCenter_LocalSnapAndSnapVault_Daily_05-09-2022_05.00.02.9432	1		05/09/2022 5:01:01 AM
SnapCenter_LocalSnapAndSnapVault_Daily_05-08-2022_05.00.02.9894	1		05/08/2022 5:01:01 AM
SnapCenter_LocalSnapAndSnapVault_Daily_05-07-2022_05.00.02.9253	1		05/07/2022 5:01:01 AM
SnapCenter_LocalSnapAndSnapVault_Daily_05-06-2022_05.00.02.9333	1		05/06/2022 5:01:01 AM
SnapCenter_LocalSnapAndSnapVault_Daily_05-05-2022_05.00.03.8844	1		05/05/2022 5:01:02 AM
SnapCenter_LocalSnapAndSnapVault_Daily_05-04-2022_05.00.03.0342	1		05/04/2022 5:01:01 AM
SnapCenter_LocalSnapAndSnapVault_Daily_05-03-2022_05.00.02.9761	1		05/03/2022 5:01:01 AM

選択したバックアップにセカンダリストレージの場所が複数ある場合は、必要なデスティネーションボリュームを選択する必要があります。

Clone From Backup x

1 Location

Select the host to create the clone

Plug-in host

Target Clone SID

NFS Export IP Address

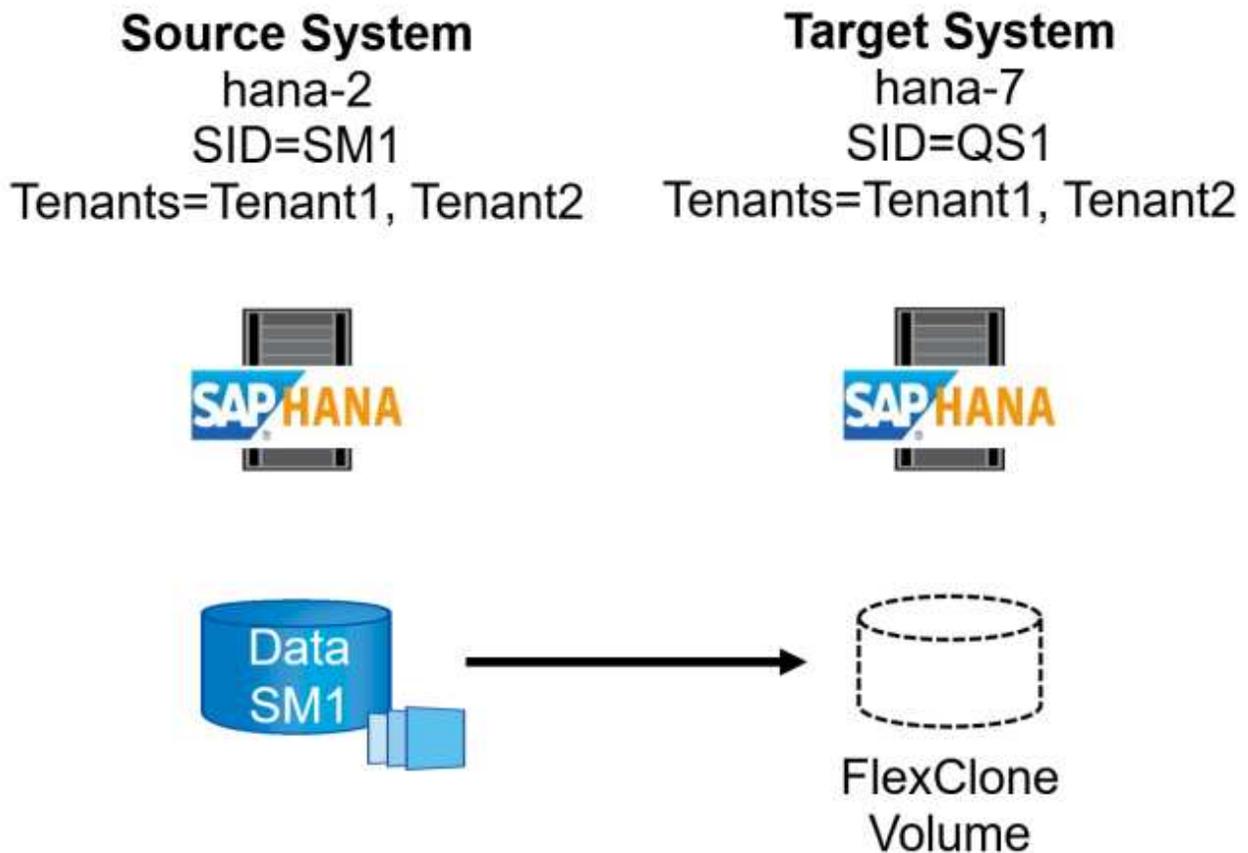
Secondary storage location : Snap Vault / Snap Mirror

Source Volume	Destination Volume
hana-primary.sapcc.stl.netapp.com:SS1_data_mnt0000 1	hana-backup.sapcc.stl.netapp.com:SS1_data

以降の手順は、プライマリストレージからのクローニングのワークフローと同じです。

複数テナントのSAP HANAシステムのクローニング

このセクションでは、複数のテナントを使用したSAP HANAシステムの更新ワークフローについて説明します。ストレージのクローニングはプライマリストレージで実行され、スクリプトを使用してさらに自動化され`sc-system-refresh.sh`ます。



SnapCenterで必要な手順は、「テナント名がSIDと等しいプライマリストレージからのクローニング」セクションで説明した手順と同じです。唯一の違いは、スクリプト内でのテナントリカバリ処理で、すべてのテナントがリカバリされる点です sc-system-refresh.sh。

```
20240430070214###hana-7###sc-system-refresh.sh:
*****
*****
20240430070214###hana-7###sc-system-refresh.sh: Script version: 3.0
20240430070214###hana-7###sc-system-refresh.sh: *****
Starting script: recovery operation *****
20240430070214###hana-7###sc-system-refresh.sh: Recover system database.
20240430070214###hana-7###sc-system-refresh.sh:
/usr/sap/QS1/HDB11/exe/Python/bin/python
/usr/sap/QS1/HDB11/exe/python_support/recoverSys.py --command "RECOVER
DATA USING SNAPSHOT CLEAR LOG"
[140310725887808, 0.008] >> starting recoverSys (at Tue Apr 30 07:02:15
2024)
[140310725887808, 0.008] args: ()
[140310725887808, 0.008] keys: \{'command': 'RECOVER DATA USING SNAPSHOT
CLEAR LOG'}
using logfile /usr/sap/QS1/HDB11/hana-7/trace/backup.log
recoverSys started: =====2024-04-30 07:02:15 =====
testing master: hana-7
hana-7 is master
shutdown database, timeout is 120
stop system
stop system on: hana-7
stopping system: 2024-04-30 07:02:15
stopped system: 2024-04-30 07:02:15
creating file recoverInstance.sql
restart database
restart master nameserver: 2024-04-30 07:02:20
start system: hana-7
sapcontrol parameter: ['-function', 'Start']
sapcontrol returned successfully:
2024-04-30T07:02:32-04:00 P0023828 18f2eab9331 INFO RECOVERY RECOVER DATA
finished successfully
recoverSys finished successfully: 2024-04-30 07:02:33
[140310725887808, 17.548] 0
[140310725887808, 17.548] << ending recoverSys, rc = 0 (RC_TEST_OK), after
17.540 secs
20240430070233###hana-7###sc-system-refresh.sh: Wait until SAP HANA
database is started ....
20240430070233###hana-7###sc-system-refresh.sh: Status: GRAY
20240430070243###hana-7###sc-system-refresh.sh: Status: GRAY
20240430070253###hana-7###sc-system-refresh.sh: Status: GRAY
```

```

20240430070304###hana-7###sc-system-refresh.sh: Status: GRAY
20240430070314###hana-7###sc-system-refresh.sh: Status: GREEN
20240430070314###hana-7###sc-system-refresh.sh: HANA system database
started.
20240430070314###hana-7###sc-system-refresh.sh: Checking connection to
system database.
20240430070314###hana-7###sc-system-refresh.sh:
/usr/sap/QS1/SYS/exe/hdb/hdbsql -U QS1KEY 'select * from sys.m_databases;'
20240430070314###hana-7###sc-system-refresh.sh: Successfully connected to
system database.
20240430070314###hana-7###sc-system-refresh.sh: Tenant databases to
recover: TENANT2
TENANT1
20240430070314###hana-7###sc-system-refresh.sh: Found inactive
tenants(TENANT2
TENANT1) and starting recovery
20240430070314###hana-7###sc-system-refresh.sh: Recover tenant database
TENANT2.
20240430070314###hana-7###sc-system-refresh.sh:
/usr/sap/QS1/SYS/exe/hdb/hdbsql -U QS1KEY RECOVER DATA FOR TENANT2 USING
SNAPSHOT CLEAR LOG
20240430070335###hana-7###sc-system-refresh.sh: Checking availability of
Indexserver for tenant TENANT2.
20240430070335###hana-7###sc-system-refresh.sh: Recovery of tenant
database TENANT2 succesfully finished.
20240430070335###hana-7###sc-system-refresh.sh: Status: GREEN
20240430070335###hana-7###sc-system-refresh.sh: Recover tenant database
TENANT1.
20240430070335###hana-7###sc-system-refresh.sh:
/usr/sap/QS1/SYS/exe/hdb/hdbsql -U QS1KEY RECOVER DATA FOR TENANT1 USING
SNAPSHOT CLEAR LOG
20240430070349###hana-7###sc-system-refresh.sh: Checking availability of
Indexserver for tenant TENANT1.
20240430070350###hana-7###sc-system-refresh.sh: Recovery of tenant
database TENANT1 succesfully finished.
20240430070350###hana-7###sc-system-refresh.sh: Status: GREEN
20240430070350###hana-7###sc-system-refresh.sh: *****
Finished script: recovery operation *****

```

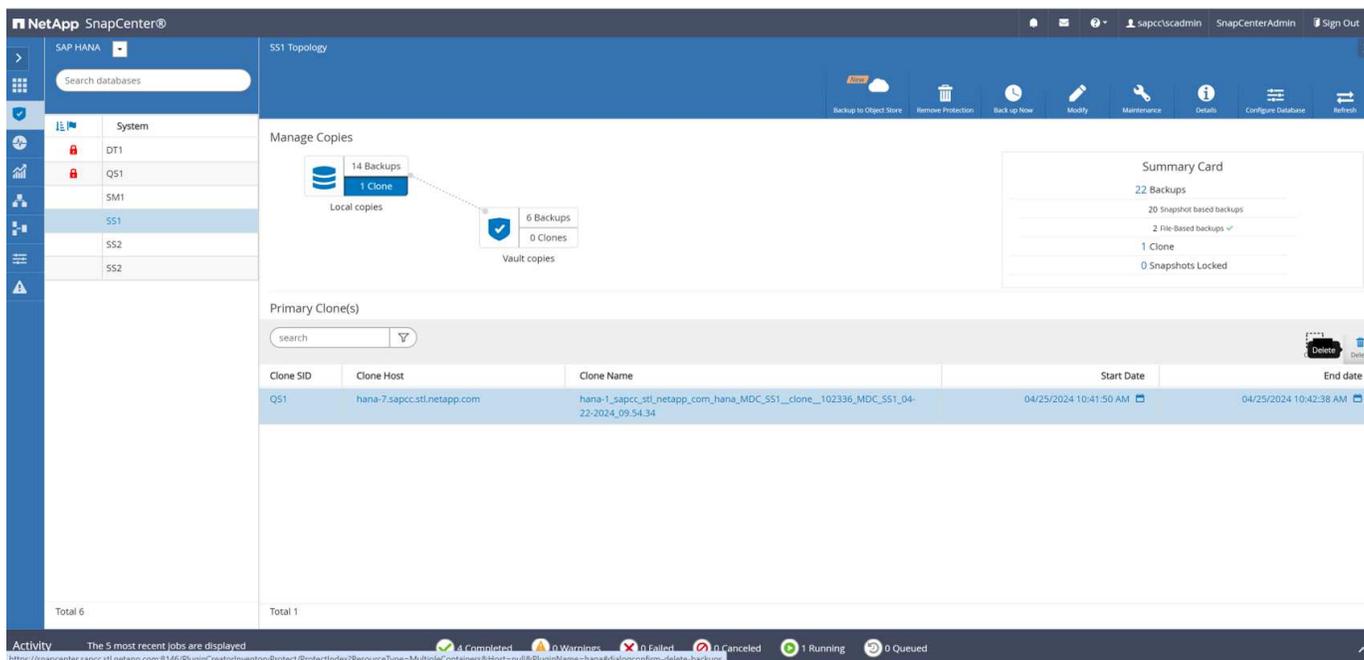
クローンの削除処理

新しいSAP HANAシステムの更新処理を開始するには、SnapCenter のクローンの削除処理を使用してターゲットシステムをクリーンアップします。

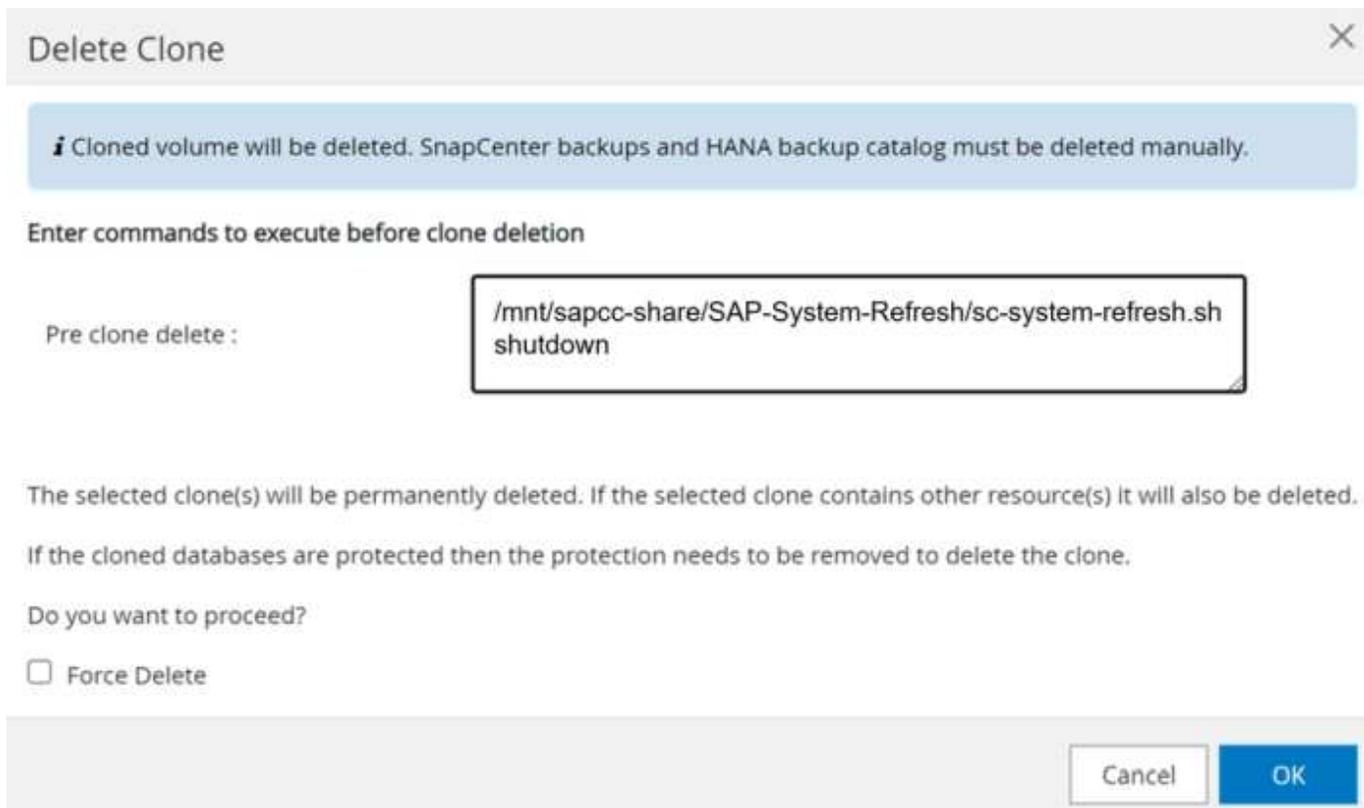
ターゲットのSAP HANAシステムがSnapCenterで保護されている場合は、まず保護を削除する必要があります。ターゲットシステムのトポロジビューで、Remove Protection（保護の削除）をクリックします。

クローン削除ワークフローを次の手順で実行します。

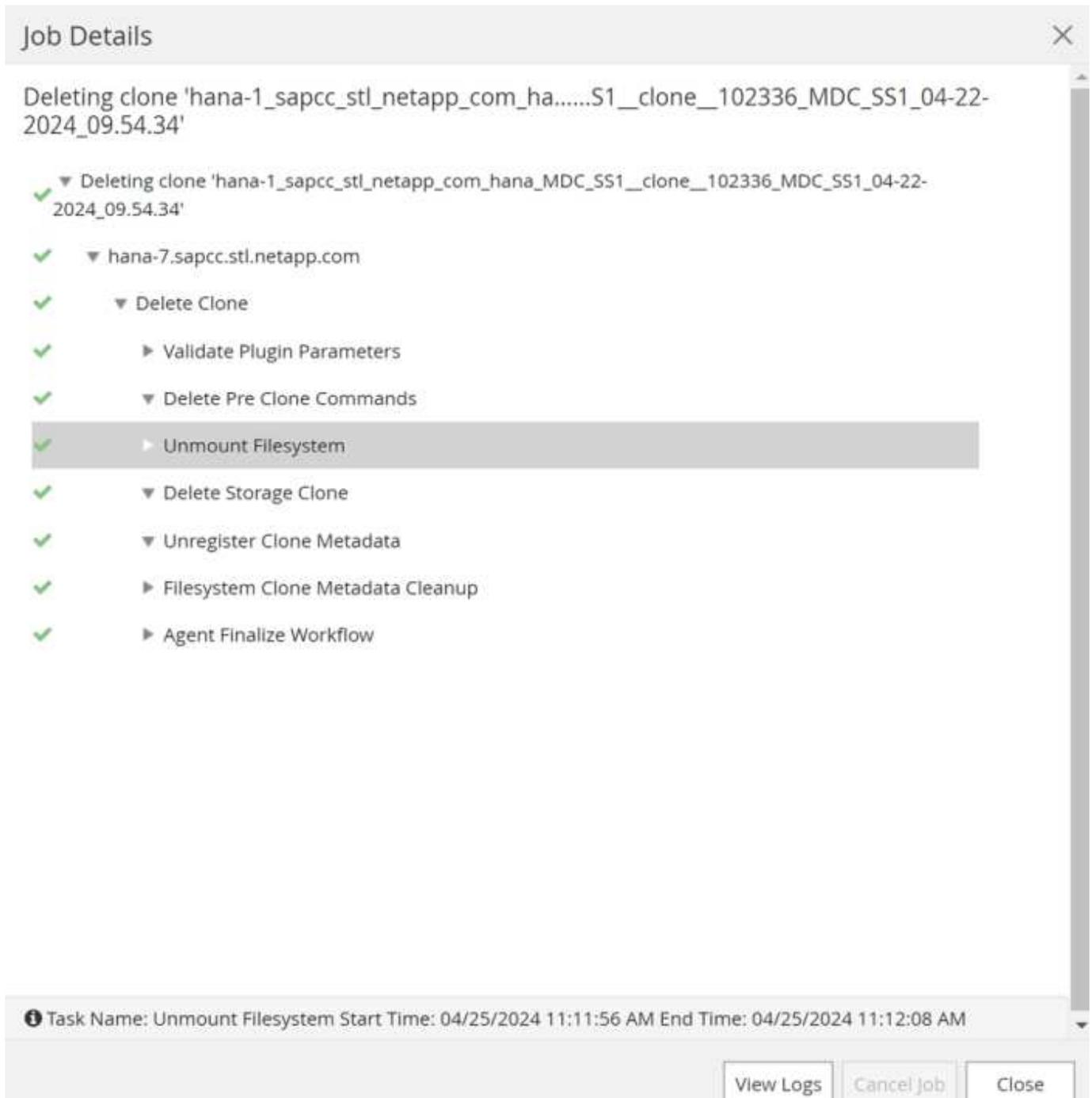
1. ソースシステムのトポロジビューでクローンを選択し、[Delete]をクリックします。



1. 必要なコマンドラインオプションを使用して、クローニング前スクリプトとアンマウント後スクリプトを入力します。



1. SnapCenter のジョブ詳細画面に処理の進捗状況が表示されます。



1. スクリプトのログファイルに `sc-system-refresh` は、シャットダウンとアンマウントの処理手順が表示されます。

```

20240425111042###hana-7###sc-system-refresh.sh:
*****
*****
20240425111042###hana-7###sc-system-refresh.sh: Script version: 3.0
20240425111042###hana-7###sc-system-refresh.sh: *****
Starting script: shutdown operation *****
20240425111042###hana-7###sc-system-refresh.sh: Stopping HANA database.
20240425111042###hana-7###sc-system-refresh.sh: sapcontrol -nr 11
-function StopSystem HDB
25.04.2024 11:10:42
StopSystem
OK
20240425111042###hana-7###sc-system-refresh.sh: Wait until SAP HANA
database is stopped ....
20240425111042###hana-7###sc-system-refresh.sh: Status: GREEN
20240425111052###hana-7###sc-system-refresh.sh: Status: YELLOW
20240425111103###hana-7###sc-system-refresh.sh: Status: YELLOW
20240425111113###hana-7###sc-system-refresh.sh: Status: YELLOW
20240425111123###hana-7###sc-system-refresh.sh: Status: YELLOW
20240425111133###hana-7###sc-system-refresh.sh: Status: YELLOW
20240425111144###hana-7###sc-system-refresh.sh: Status: YELLOW
20240425111154###hana-7###sc-system-refresh.sh: Status: GRAY
20240425111154###hana-7###sc-system-refresh.sh: SAP HANA database is
stopped.
20240425111154###hana-7###sc-system-refresh.sh: *****
Finished script: shutdown operation *****

```

1. SnapCenter のクローン作成処理を使用して、SAP HANAの更新処理を再開できるようになりました。

クローンスプリット処理を使用したSAP HANAシステムの更新

システム更新処理のターゲットシステムを長期間使用する予定がある場合は、システム更新処理の一環としてFlexCloneボリュームをスプリットすることを推奨します。

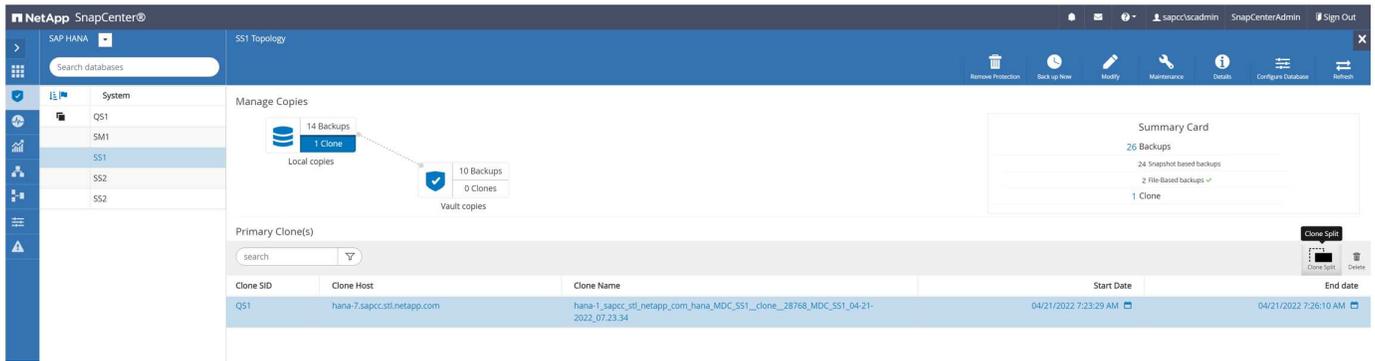


クローンスプリット処理でクローンボリュームの使用がブロックされることはないため、SAP HANAデータベースの使用中にいつでも実行できます。

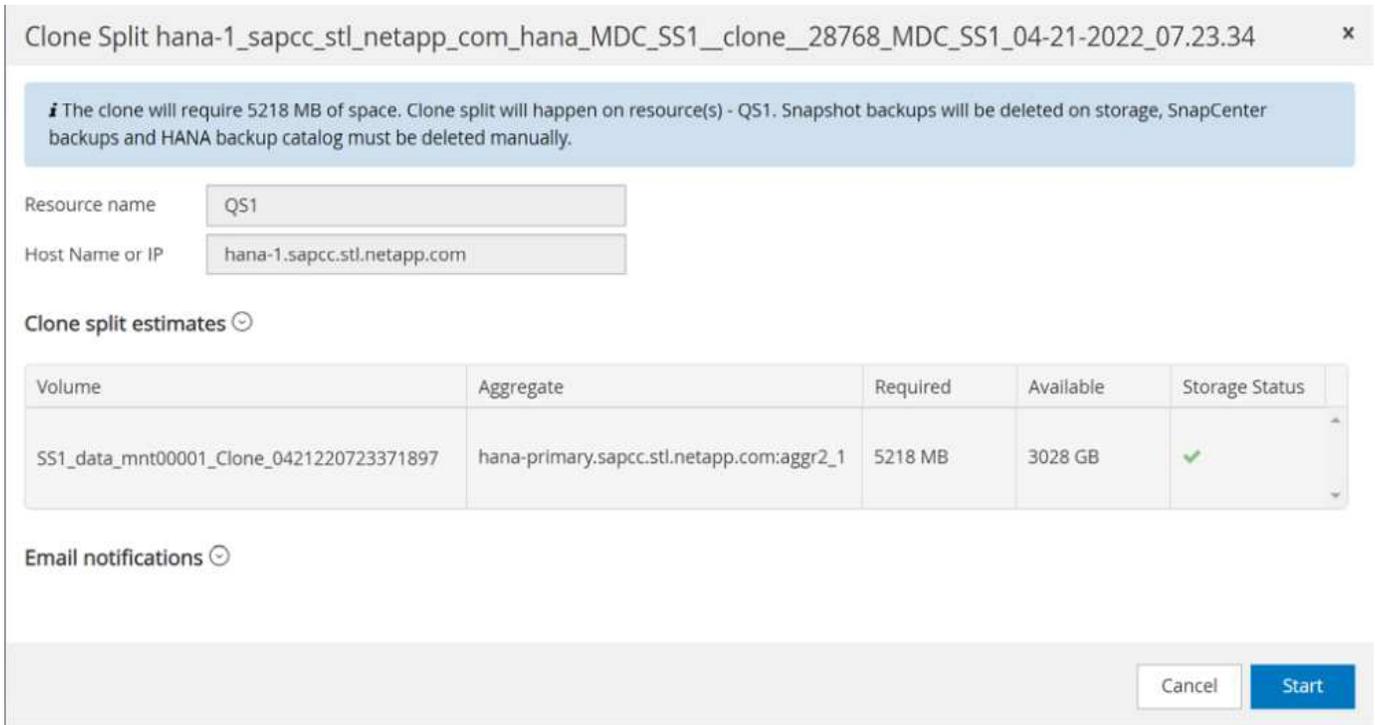


Azure NetApp FilesではAzure NetApp Files、作成後に常にクローンがスプリットされるため、クローンスプリット処理は実行できません。

SnapCenter のクローンスプリットのワークフローは、クローンを選択してクローンスプリットをクリックすることで、ソースシステムのトポロジビューで開始されます。



次の画面には、スプリットボリュームに必要な容量に関する情報がプレビューで表示されます。



SnapCenter ジョブログには、クローンスプリット処理の進捗状況が表示されます。

Job Details



Clone Split Start of Resource 'hana-1_sapcc_stl_ne.....MDC_SS1__clone__28768_MDC_SS1_04-21-2022_07.23.34'

- ✓ ▾ Clone Split Start of Resource 'hana-1_sapcc_stl_netapp_com_hana_MDC_SS1__clone__28768_MDC_SS1_04-21-2022_07.23.34'
- ✓ ▾ SnapCenter.sapcc.stl.netapp.com
 - ✓ ▶ Volume Clone Estimate
 - ✓ ▶ Volume Clone Split Start
 - ✓ ▶ Delete Backups of Clone
 - ✓ ▾ Volume Clone Split Status
 - ✓ ▶ Clone Split Status for volume SS1_data_mnt00001_Clone_0421220723371897 is 'In Progress'
 - ✓ ▶ Clone Split Status for volume SS1_data_mnt00001_Clone_0421220723371897'Completed'
 - ✓ ▶ Register Clone Split
 - ✓ ▶ Data Collection
 - ✓ ▶ Send EMS Messages

i Task Name: Volume Clone Split Status Start Time: 04/21/2022 7:51:16 AM End Time:

View Logs

Cancel Job

Close

SnapCenterのリソースビューで、ターゲットシステムQS1がクローニングされたリソースとしてマークされなくなりました。ソースシステムのトポロジビューに戻ると、クローンは表示されなくなります。スプリットボリュームは、ソースシステムのSnapshotバックアップとは独立しています。

System	System ID (SID)	Tenant Databases	Replication	Plug-in Host	Resource Groups	Policies	Last Backup	Overall Status
Q51	Q51	Q51	None	hana-7.sapcc.stl.netapp.com		LocalSnap	04/21/2022 7:30:50 AM	Backup succeeded
SM1	SM1	TENANT1	None	hana-2.sapcc.stl.netapp.com		LocalSnap	04/21/2022 4:01:01 AM	Backup succeeded
SS1	SS1	SS1	None	hana-1.sapcc.stl.netapp.com		BlockIntegrityCheck LocalSnap LocalSnapAndSnapVault LocalSnap-OnDemand	04/21/2022 7:01:01 AM	Backup succeeded
SS2	SS2	SS2	Enabled (Primary)	hana-3.sapcc.stl.netapp.com	SS2 - HANA System Replication	BlockIntegrityCheck LocalSnapKeep2	04/21/2022 7:57:22 AM	Backup succeeded
SS2	SS2	SS2	Enabled (Secondary)	hana-4.sapcc.stl.netapp.com	SS2 - HANA System Replication	BlockIntegrityCheck LocalSnapKeep2	04/11/2022 2:57:21 AM	Backup succeeded

SS1 Topology

Manage Copies

- Local copies: 14 Backups, 0 Clones
- Vault copies: 10 Backups, 0 Clones

Summary Card

- 26 Backups
- 24 Snapshot based backups
- 2 File-based backups w/
- 0 Clones

Primary Backup(s)

Backup Name	Count	IF	End Date
SnapCenter_LocalSnap_Hourly_04-21-2022_07.00.02.7865	1		04/21/2022 7:01:01 AM
SnapCenter_LocalSnapAndSnapVault_Daily_04-21-2022_05.00.02.8215	1		04/21/2022 5:01:02 AM
SnapCenter_LocalSnap_Hourly_04-21-2022_03.00.01.7085	1		04/21/2022 3:01:00 AM
SnapCenter_LocalSnap_Hourly_04-20-2022_23.00.01.7142	1		04/20/2022 11:01:00 PM
SnapCenter_LocalSnap_Hourly_04-20-2022_19.00.01.9499	1		04/20/2022 7:01:00 PM

クローンプリット処理後の更新ワークフローは、クローンプリットを使用しない処理と少し異なります。クローンプリット処理後は、ターゲットデータボリュームがFlexCloneボリュームでなくなるため、クローン削除処理は必要ありません。

このワークフローは、次の手順で構成されます。

1. ターゲットのSAP HANAシステムがSnapCenterで保護されている場合は、まず保護を削除する必要があります。
2. SAP HANAデータベースをシャットダウンし、データボリュームをアンマウントして、SnapCenterで作成されたfstabエントリを削除する必要があります。これらの手順は手動で実行する必要があります。
3. これで、前のセクションで説明したように、SnapCenterクローン作成ワークフローを実行できるようになりました。
4. 更新処理後も古いターゲットデータボリュームは引き続き存在するため、ONTAP System Managerなどを使用して手動で削除する必要があります。

PowerShellスクリプトによるSnapCenter ワークフロー自動化

前のセクションでは、SnapCenter UIを使用してさまざまなワークフローを実行し、PowerShellスクリプトまたはREST API呼び出しを使用してすべてのワークフローを実行することもできるため、さらなる自動化が可能です。以降のセクションでは、以降のワークフローの基本的なPowerShellスクリプトの例について説明します。

- クローンを作成します
- クローンを削除します



このサンプルスクリプトは現状のまま提供されており、ネットアップではサポートしていません。

すべてのスクリプトはPowerShellコマンドウィンドウで実行する必要があります。スクリプトを実行する前に 'Open-SmConnection' コマンドを使用してSnapCenter サーバへの接続を確立する必要があります

クローンを作成します

以下の簡単なスクリプトは、PowerShellコマンドを使用してSnapCenter クローン作成処理を実行する方法を示しています。SnapCenter の「New-SmClone」コマンドは、ラボ環境に必要なコマンドライン・オプションと、前述した自動化スクリプトを使用して実行します。

```
$BackupName='SnapCenter_hana-1_LocalSnap_Hourly_06-25-2024_03.00.01.8458'
$JobInfo=New-SmClone -AppPluginCode hana -BackupName $BackupName
-Resources @\{"Host"="hana-1.sapcc.stl.netapp.com";"UID"="MDC\SS1"}
-CloneToInstance hana-7.sapcc.stl.netapp.com -postclonecreatecommands
'/mnt/sapcc-share/SAP-System-Refresh/sc-system-refresh.sh recover'
-NFSEExportIPs 192.168.175.75 -CloneUid 'MDC\QS1'
# Get JobID of clone create job
$Job=Get-SmJobSummaryReport | ?\{$_.JobType -eq "Clone" } | ?\{$_.JobName
-Match $BackupName} | ?\{$_.Status -eq "Running"}
$JobId=$Job.SmJobId
Get-SmJobSummaryReport -JobId $JobId
# Wait until job is finished
do \{ $Job=Get-SmJobSummaryReport -JobId $JobId; write-host $Job.Status;
sleep 20 } while ( $Job.Status -Match "Running" )
Write-Host " "
Get-SmJobSummaryReport -JobId $JobId
Write-Host "Clone create job has been finshed."
```

画面出力には、クローン作成PowerShellスクリプトの実行状況が表示されます。

```
PS C:\Windows\system32> C:\NetApp\clone-create.ps1
SmJobId : 110382
JobCreatedDateTime :
JobStartDateTime : 6/26/2024 9:55:34 AM
JobEndDateTime :
JobDuration :
JobName : Clone from backup 'SnapCenter_hana-1_LocalSnap_Hourly_06-25-
2024_03.00.01.8458'
JobDescription :
Status : Running
IsScheduled : False
JobError :
JobType : Clone
PolicyName :
JobResultData :
Running
Completed
SmJobId : 110382
JobCreatedDateTime :
JobStartDateTime : 6/26/2024 9:55:34 AM
JobEndDateTime : 6/26/2024 9:58:50 AM
JobDuration : 00:03:16.6889170
JobName : Clone from backup 'SnapCenter_hana-1_LocalSnap_Hourly_06-25-
2024_03.00.01.8458'
JobDescription :
Status : Completed
IsScheduled : False
JobError :
JobType : Clone
PolicyName :
JobResultData :
Clone create job has been finshed.
```

クローンを削除します

以下の簡単なスクリプトは、PowerShellコマンドを使用してSnapCenter クローンの削除処理を実行する方法

を示しています。SnapCenter のRemove-SmClone コマンドは'実習環境に必要なコマンド・ライン・オプション'前に説明した自動化スクリプトを使用して実行します

```
$CloneInfo=Get-SmClone |?{$_.CloneName -Match "hana-1_sapcc_stl_netapp_com_hana_MDC_SS1" }
$JobInfo=Remove-SmClone -CloneName $CloneInfo.CloneName -PluginCode hana -PreCloneDeleteCommands '/mnt/sapcc-share/SAP-System-Refresh/sc-system-refresh.sh shutdown QS1' -UnmountCommands '/mnt/sapcc-share/SAP-System-Refresh/sc-system-refresh.sh umount QS1' -Confirm: $False
Get-SmJobSummaryReport -JobId $JobInfo.Id
# Wait until job is finished
do \{ $Job=Get-SmJobSummaryReport -JobId $JobInfo.Id; write-host $Job.Status; sleep 20 } while ( $Job.Status -Match "Running" )
Write-Host " "
Get-SmJobSummaryReport -JobId $JobInfo.Id
Write-Host "Clone delete job has been finished."
PS C:\NetApp>
```

画面出力には、PowerShellスクリプトclone-delete.ps1が実行されたことが示されています。

```
PS C:\Windows\system32> C:\NetApp\clone-delete.ps1
SmJobId : 110386
JobCreatedDateTime :
JobStartDateTime : 6/26/2024 10:01:33 AM
JobEndDateTime :
JobDuration :
JobName : Deleting clone 'hana-
1_sapcc_stl_netapp_com_hana_MDC_SS1__clone__110382_MDC_SS1_04-22-
2024_09.54.34'
JobDescription :
Status : Running
IsScheduled : False
JobError :
JobType : DeleteClone
PolicyName :
JobResultData :
Running
Running
Running
Running
Completed
SmJobId : 110386
JobCreatedDateTime :
JobStartDateTime : 6/26/2024 10:01:33 AM
JobEndDateTime : 6/26/2024 10:02:38 AM
JobDuration : 00:01:05.5658860
JobName : Deleting clone 'hana-
1_sapcc_stl_netapp_com_hana_MDC_SS1__clone__110382_MDC_SS1_04-22-
2024_09.54.34'
JobDescription :
Status : Completed
IsScheduled : False
JobError :
JobType : DeleteClone
PolicyName :
JobResultData :
Clone delete job has been finshed.
PS C:\Windows\system32>
```

SnapCenter によるSAPシステムのクローニング

このセクションでは、SAPシステムのクローニング処理の手順ごとの概要を示します。このを使用して、論理的な破損に対処するための修復システムをセットアップできます。

次の図は、SnapCenterを使用したSAPシステムのクローニング処理に必要な手順をまとめたものです。

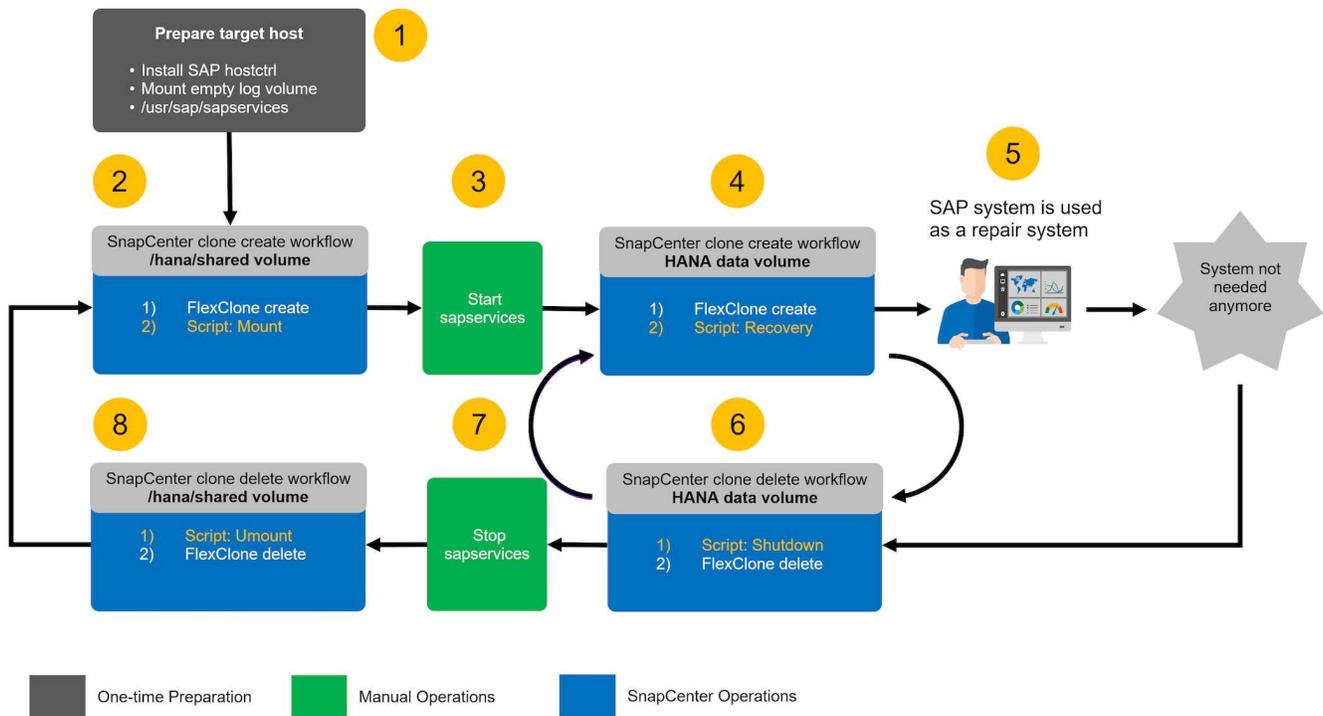
1. ターゲットホストを準備します。
2. SAP HANA共有ボリューム用のSnapCenterクローン作成ワークフロー
3. SAP HANAサービスを開始します。
4. SnapCenterクローンは、SAP HANAデータボリュームのデータベースリカバリを含むワークフローを作成します。
5. SAP HANAシステムを修復システムとして使用できるようになりました。



システムを別のSnapshotバックアップにリセットする必要がある場合は、手順6と手順4で十分です。SAP HANA共有ボリュームは引き続きマウントできます。

システムが不要になった場合は、次の手順でクリーンアッププロセスを実行します。

6. SAP HANAデータボリュームのSnapCenterクローン削除ワークフロー（データベースのシャットダウンを含む）。
7. SAP HANAサービスを停止します。
8. SAP HANA共有ボリュームのSnapCenterクローン削除ワークフロー



前提条件および制限事項

以降のセクションで説明するワークフローには、SAP HANAシステムのアーキテクチャとSnapCenterの設定に関する前提条件と制限事項がいくつかあります。

- ここで説明するワークフローは、シングルホストのSAP HANA MDCシステムに対して有効です。複数のホストシステムはサポートされません。

- 自動スクリプトを実行するには、SnapCenter SAP HANAプラグインをターゲットホストに導入する必要があります。
- NFSのワークフローが検証されました。SAP HANA共有ボリュームのマウントに使用される自動化で`script sc-mount-volume.sh`は、FCPはサポートされません。この手順は、手動で実行するか、スクリプトを拡張して実行する必要があります。
- ここで説明するワークフローは、SnapCenter 5.0リリース以降でのみ有効です。

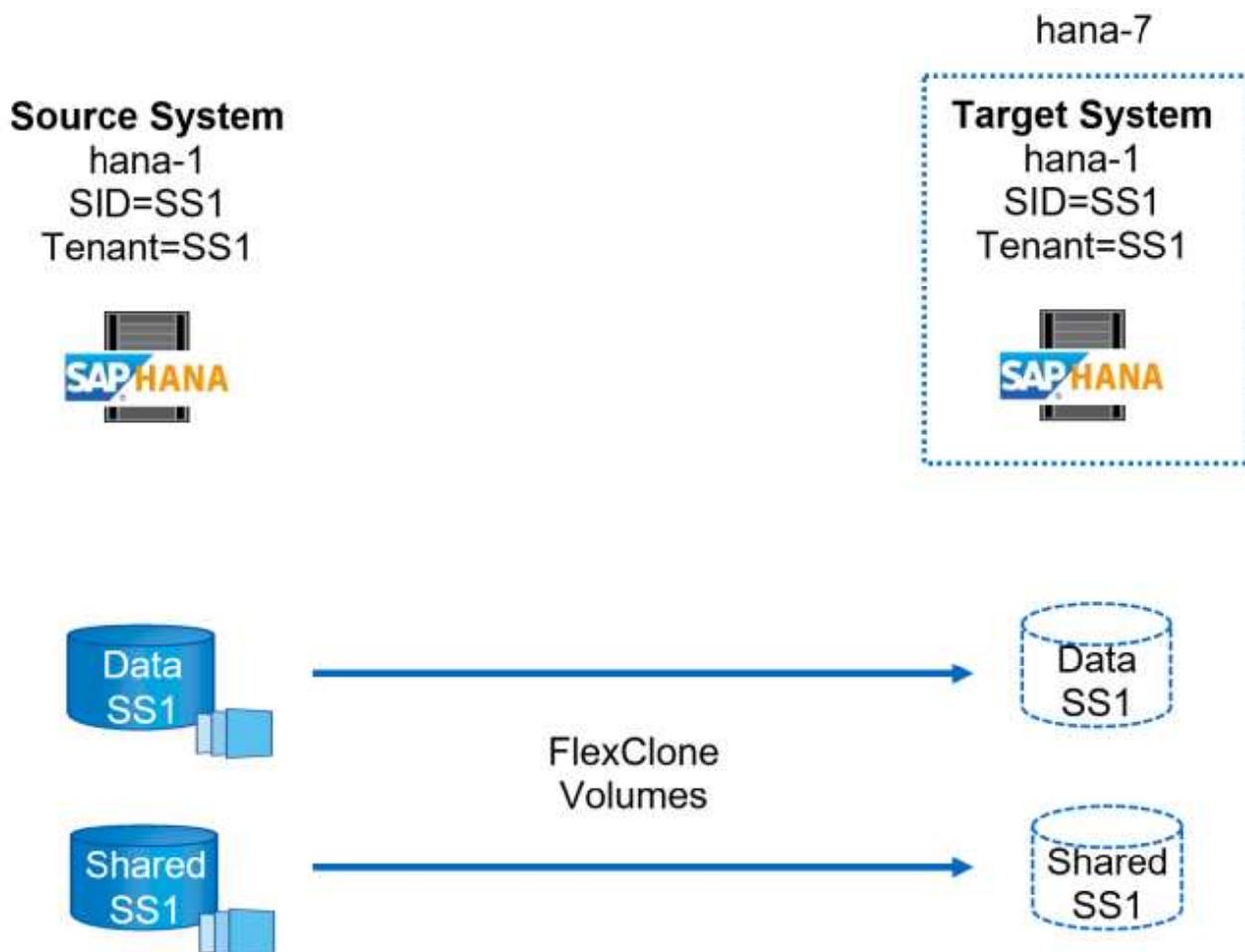
ラボのセットアップ

次の図は、システムのクローン作成に使用するラボのセットアップを示しています。

使用したソフトウェアバージョンは次のとおりです。

- SnapCenter 5.0
- SAP HANAシステム：HANA 2.0 SPS6 rev.61
- SLES 15
- ONTAP 9.7P7.

すべてのSAP HANAシステムは、構成ガイドに基づいて構成する必要があります。"[NFSを使用したNetApp AFF システムでのSAP HANA](#)"。SnapCenterとSAP HANAリソースはベストプラクティスガイドに基づいて構成されました。"[SnapCenterを使用したSAP HANAのバックアップとリカバリ](#)"。



ターゲットホストの準備

ここでは、システムクローンターゲットとして使用するサーバに必要な準備手順について説明します。

通常運用時は、SAP HANA QAやテストシステムなど、他の目的にターゲットホストを使用できます。したがって、前述のほとんどの手順は、システムクローン処理が要求されたときに実行する必要があります。一方、やなどの関連する設定ファイルは /etc/fstab、`/usr/sap/sapservices` 設定ファイルをコピーするだけで準備し、本番環境に置くことができます。

ターゲットホストの準備には、SAP HANA QAまたはテストシステムのシャットダウンも含まれます。

ターゲットサーバーのホスト名とIPアドレス

ターゲット・サーバのホスト名は`ソース・システムのホスト名と同じである必要がありますIP アドレスは異なってもかまいません。



ターゲットサーバが他のシステムと通信できないように、ターゲットサーバの適切なフェンシングを確立する必要があります。適切なフェンシングが設定されていないと、クローニングされた本番用システムは他の本番用システムとデータを交換する可能性があります。



このラボ環境では、ターゲットシステム側から見て、ターゲットシステムのホスト名を内部的にのみ変更しました。ホストの外部からは、ホスト名として「HANA-7」を使用してアクセスできました。ホストにログインすると、ホスト自体がHANAです。

必要なソフトウェアのインストール

SAP ホストエージェントソフトウェアをターゲットサーバにインストールする必要があります。詳細については、SAPヘルプポータルのを参照して "[SAP ホストエージェント](#)" ください。

SnapCenterでのホストの追加処理を使用して、SnapCenter SAP HANAプラグインをターゲットホストに導入する必要があります。

ユーザー'ポート'およびSAPサービスの構成

ターゲットサーバに、SAP HANA データベースに必要なユーザとグループが配置されている必要があります。通常は、ユーザの一元管理が使用されるため、ターゲットサーバで設定手順を行う必要はありません。SAP HANAデータベースに必要なポートがターゲットホストに設定されている必要があります。/etc/servicesファイルをターゲット・サーバにコピーすることにより`ソース・システムから構成をコピーできます

必要な SAP サービスのエントリがターゲットホストにあることが必要です。/usr/sap/sapservices' ファイルをターゲットサーバにコピーすることで`ソースシステムから構成をコピーできます次の出力は、このラボ環境で使用する SAP HANA データベースの必須エントリを示しています。

```
#!/bin/sh
LD_LIBRARY_PATH=/usr/sap/SS1/HDB00/exe:$LD_LIBRARY_PATH;export
LD_LIBRARY_PATH;/usr/sap/SS1/HDB00/exe/sapstartsrv
pf=/usr/sap/SS1/SYS/profile/SS1_HDB00_hana-1 -D -u ssladm
limit.descriptors=1048576
```

ログとログのバックアップボリュームを準備

ログボリュームをソースシステムからクローニングする必要はなく、clear logオプションを使用してリカバリを実行するため、空のログボリュームをターゲットホストで準備しておく必要があります。

ソースシステムには独立したログバックアップボリュームが設定されているため、空のログバックアップボリュームを準備し、ソースシステムと同じマウントポイントにマウントする必要があります。

```
hana-1:/# cat /etc/fstab
192.168.175.117:/SS1_repair_log_mnt00001 /hana/log/SS1/mnt00001 nfs
rw,vers=3,hard,timeo=600,rsiz=1048576,wsiz=1048576,intr,noatime,nolock 0
0
192.168.175.117:/SS1_repair_log_backup /mnt/log-backup nfs
rw,vers=3,hard,timeo=600,rsiz=1048576,wsiz=1048576,intr,noatime,nolock 0
0
```

ログボリュームhdb*内では、ソース・システムと同じ方法でサブディレクトリを作成する必要があります。

```
hana-1:/ # ls -al /hana/log/SS1/mnt00001/
total 16
drwxrwxrwx 5 root root 4096 Dec 1 06:15 .
drwxrwxrwx 1 root root 16 Nov 30 08:56 ..
drwxr-xr-- 2 ssladm sapsys 4096 Dec 1 06:14 hdb00001
drwxr-xr-- 2 ssladm sapsys 4096 Dec 1 06:15 hdb00002.00003
drwxr-xr-- 2 ssladm sapsys 4096 Dec 1 06:15 hdb00003.00003
```

ログバックアップボリュームには、システムとテナントデータベースのサブディレクトリを作成する必要があります。

```
hana-1:/ # ls -al /mnt/log-backup/
total 12
drwxr-xr-- 2 ssladm sapsys 4096 Dec 1 04:48 .
drwxr-xr-- 2 ssladm sapsys 4896 Dec 1 03:42 ..
drwxr-xr-- 2 ssladm sapsys 4096 Dec 1 06:15 DB_SS1
drwxr-xr-- 2 ssladm sapsys 4096 Dec 1 06:14 SYSTEMDB
```

ファイル・システム・マウントの準備

データおよび共有ボリュームのマウントポイントを準備しておく必要があります。

この例では、ディレクトリ /hana/data/SS1/mnt00001、 /hana/shared および usr/sap/SS1 を作成する必要があります。

スクリプト実行の準備

ターゲットシステムで実行するスクリプトを、SnapCenter allowed commands configファイルに追加する必要があります。

```
hana-7:/opt/NetApp/snapcenter/scc/etc # cat
/opt/NetApp/snapcenter/scc/etc/allowed_commands.config
command: mount
command: umount
command: /mnt/sapcc-share/SAP-System-Refresh/sc-system-refresh.sh
command: /mnt/sapcc-share/SAP-System-Refresh/sc-mount-volume.sh
hana-7:/opt/NetApp/snapcenter/scc/etc #
```

HANA共有ボリュームのクローニング

1. ソースシステムのSS1共有ボリュームからSnapshotバックアップを選択し、[Clone]をクリックします。

Backup Name	Count	IF	End Date
SnapCenter_LocalSnap_Hourly_05-13-2022_05.04.01.8012	1		05/13/2022 5:04:12 AM
SnapCenter_LocalSnap_Hourly_05-13-2022_01.04.01.9799	1		05/13/2022 1:04:12 AM
SnapCenter_LocalSnap_Hourly_05-12-2022_21.04.01.8899	1		05/12/2022 9:04:12 PM

1. ターゲット修復システムの準備が完了したホストを選択します。NFSエクスポートのIPアドレスは、ターゲットホストのストレージネットワークインターフェイスである必要があります。ターゲットSIDとして、ソースシステムと同じSIDを保持します。この例では、SS1。

Clone From Backup

1 Location Select the host to create the clone

2 Scripts

3 Notification

4 Summary

Plug-in host: hana-7.sapcc.stl.netapp.com

Target Clone SID: SS1

NFS Export IP Address: 192.168.175.75

1. 必要なコマンドラインオプションを指定して、マウントスクリプトを入力します。



SAP HANAシステムでは、構成ガイドで推奨されているように、およびに
`/usr/sap/SS1` 単一のボリュームをサブディレクトリに分けて使用します
`/hana/shared` "NFSを使用したNetApp AFF システムでのSAP HANA"。スクリプト
`sc-mount-volume.sh` では、マウントパスに特別なコマンドラインオプションを使用し
てこの設定をサポートしています。mount pathコマンドラインオプションをusr-sap-and-
sharedと指定すると、sharedおよびusr-sapのサブディレクトリがボリューム内に適宜マウ
ントされます。

Clone From Backup

- 1 Location
- 2 Scripts**
- 3 Notification
- 4 Summary

Enter optional commands to run before performing a clone operation ⓘ

Pre clone command

Enter optional commands to mount a file system to a host ⓘ

Mount command

Enter optional commands to run after performing a clone operation ⓘ

Post clone command

Configure an SMTP Server to send email notifications for Clone jobs by going to [Settings>Global Settings>Notification Server Settings.](#)

Previous Next

1. SnapCenter の[ジョブの詳細]画面に、処理の進捗状況が表示されます。

Job Details x

Clone from backup 'SnapCenter_LocalSnap_Hourly_05-13-2022_05.04.01.8012'

- ✓ ▾ Clone from backup 'SnapCenter_LocalSnap_Hourly_05-13-2022_05.04.01.8012'
- ✓ ▾ hana-7.sapcc.stl.netapp.com
 - ✓ ▾ Clone
 - ▶ Storage Clone
 - ▶ Register Clone Metadata
 - ▶ Data Collection
 - ▶ Agent Finalize Workflow

i Task Name: Clone Start Time: 05/13/2022 5:14:02 AM End Time: 05/13/2022 5:14:16 AM

[View Logs](#) [Cancel job](#) [Close](#)

1. sc-mount-volume.shスクリプトのログファイルには、マウント処理で実行されたさまざまな手順が表示されます。

```

20201201041441###hana-1###sc-mount-volume.sh: Adding entry in /etc/fstab.
20201201041441###hana-1###sc-mount-volume.sh:
192.168.175.117://SS1_shared_Clone_05132205140448713/usr-sap /usr/sap/SS1
nfs
rw,vers=3,hard,timeo=600,rsz=1048576,wsz=1048576,intr,noatime,nolock 0
0
20201201041441###hana-1###sc-mount-volume.sh: Mounting volume: mount
/usr/sap/SS1.
20201201041441###hana-1###sc-mount-volume.sh:
192.168.175.117://SS1_shared_Clone_05132205140448713/shared /hana/shared
nfs
rw,vers=3,hard,timeo=600,rsz=1048576,wsz=1048576,intr,noatime,nolock 0
0
20201201041441###hana-1###sc-mount-volume.sh: Mounting volume: mount
/hana/shared.
20201201041441###hana-1###sc-mount-volume.sh: usr-sap-and-shared mounted
successfully.
20201201041441###hana-1###sc-mount-volume.sh: Change ownership to ssladm.

```

1. SnapCenterワークフローが完了すると、/usr/sap/ss1と/hana/sharedファイルシステムがターゲットホストにマウントされます。

```

hana-1:~ # df
Filesystem 1K-blocks Used Available Use% Mounted on
192.168.175.117://SS1_repair_log_mnt00001 262144000 320 262143680 1%
/hana/log/SS1/mnt00001
192.168.175.100://sapcc_share 1020055552 53485568 966569984 6% /mnt/sapcc-
share
192.168.175.117://SS1_repair_log_backup 104857600 256 104857344 1%
/mnt/log-backup
192.168.175.117://SS1_shared_Clone_05132205140448713/usr-sap 262144064
10084608 252059456 4% /usr/sap/SS1
192.168.175.117://SS1_shared_Clone_05132205140448713/shared 262144064
10084608 252059456 4% /hana/shared

```

1. SnapCenter では、クローニングされたボリュームの新しいリソースが表示されます。

The screenshot shows the NetApp SnapCenter interface with a table of resources. The table has columns for Name, Associated System ID (SID), Plug-in Host, Resource Groups, Policies, Last backup, and Overall Status. Two rows are visible, representing cloned volumes.

Name	Associated System ID (SID)	Plug-in Host	Resource Groups	Policies	Last backup	Overall Status
SS1-Shared-Volume	SS1	hana-1.sapcc.stl.netapp.com		LocalSnap LocalSnap-OnDemand	05/13/2022 5:04:12 AM Backup succeeded	Backup succeeded
SS1-Shared-Volume	SS1	hana-7.sapcc.stl.netapp.com				Not protected

1. /hana/sharedボリュームが使用可能になったので、SAP HANAサービスを開始できます。

```
hana-1:/mnt/sapcc-share/SAP-System-Refresh # systemctl start sapinit
```

1. これでSAPホスト・エージェントとsapstartsrvのプロセスが開始されました

```
hana-1:/mnt/sapcc-share/SAP-System-Refresh # ps -ef |grep sap
root 12377 1 0 04:34 ? 00:00:00 /usr/sap/hostctrl/exe/saphostexec
pf=/usr/sap/hostctrl/exe/host_profile
sapadm 12403 1 0 04:34 ? 00:00:00 /usr/lib/systemd/systemd --user
sapadm 12404 12403 0 04:34 ? 00:00:00 (sd-pam)
sapadm 12434 1 1 04:34 ? 00:00:00 /usr/sap/hostctrl/exe/sapstartsrv
pf=/usr/sap/hostctrl/exe/host_profile -D
root 12485 12377 0 04:34 ? 00:00:00 /usr/sap/hostctrl/exe/saphostexec
pf=/usr/sap/hostctrl/exe/host_profile
root 12486 12485 0 04:34 ? 00:00:00 /usr/sap/hostctrl/exe/saposcol -l -w60
pf=/usr/sap/hostctrl/exe/host_profile
ssladm 12504 1 0 04:34 ? 00:00:00 /usr/sap/SS1/HDB00/exe/sapstartsrv
pf=/usr/sap/SS1/SYS/profile/SS1_HDB00_hana-1 -D -u ssladm
root 12582 12486 0 04:34 ? 00:00:00 /usr/sap/hostctrl/exe/saposcol -l -w60
pf=/usr/sap/hostctrl/exe/host_profile
root 12585 7613 0 04:34 pts/0 00:00:00 grep --color=auto sap
hana-1:/mnt/sapcc-share/SAP-System-Refresh #
```

追加のSAPアプリケーションサービスのクローニング

その他のSAPアプリケーションサービスのクローニングは、セクション「SAP HANA共有ボリュームのクローニング」で説明したSAP HANA共有ボリュームと同じ方法で行います。もちろん、SAPアプリケーションサーバに必要なストレージボリュームもSnapCenterで保護する必要があります。

必要なサービスエントリを/usr/sap/sapservicesに追加し、ポート、ユーザ、およびファイルシステムのマウントポイント (/usr/sap/SIDなど) を準備しておく必要があります。

データボリュームのクローニングとHANAデータベースのリカバリ

1. ソースシステムSS1からSAP HANA Snapshotバックアップを選択します。

The screenshot shows the NetApp SnapCenter interface. On the left, a sidebar lists systems: Q51, SM1, SS1, SS2, and SS2. The main area displays 'Manage Copies' with a diagram showing 15 Backups, 0 Clones, and 11 Backups, 0 Clones. Below this is a table of 'Primary Backup(s)'. On the right, a 'Summary Card' shows 28 Backups, 26 Snapshot based backups, 2 File-based backups, and 0 Clones.

Backup Name	Count	IF	End Date
SnapCenter_LocalSnapAndSnapVault_Daily_05-13-2022_05.00.03.0030	1		05/13/2022 5:01:01 AM
SnapCenter_LocalSnap_Hourly_05-13-2022_03.00.01.8016	1		05/13/2022 3:01:00 AM
SnapCenter_LocalSnap_Hourly_05-12-2022_23.00.01.8743	1		05/12/2022 11:01:00 PM
SnapCenter_LocalSnap_Hourly_05-12-2022_19.00.01.9803	1		05/12/2022 7:01:00 PM

1. ターゲット修復システムの準備が完了したホストを選択します。NFSエクスポートのIPアドレスは、ターゲットホストのストレージネットワークインターフェイスである必要があります。ターゲットSIDとして、ソースシステムと同じSIDを保持します。この例では、SS1

The 'Clone From Backup' dialog box is shown in Step 1: Location. The title is 'Select the host to create the clone'. The fields are:

- Plug-in host: hana-7.sapcc.stl.netapp.com
- Target Clone SID: SS1
- NFS Export IP Address: 192.168.175.75

1. 必要なコマンドラインオプションを指定して、クローニング後のスクリプトを入力します。



リカバリ処理用スクリプトは、Snapshot処理の時点までSAP HANAデータベースをリカバリし、フォワードリカバリを実行しません。特定の時点までのフォワードリカバリが必要な場合は、リカバリを手動で実行する必要があります。手動フォワードリカバリでは、ソースシステムのログバックアップをターゲットホストで利用できることも必要です。

The 'Clone From Backup' dialog box is shown in Step 2: Scripts. The title is 'Enter optional commands to run before performing a clone operation'. The fields are:

- Pre clone command: (empty text box)
- Post clone command: /mnt/sapcc-share/SAP-System-Refresh/sc-system-refresh.sh recover

SnapCenter のジョブ詳細画面に処理の進捗状況が表示されます。

The screenshot shows a 'Job Details' window with a title bar containing 'Job Details' and a close button 'x'. The main content area displays the job title 'Clone from backup 'SnapCenter_LocalSnap_Hourly_05-13-2022_03.00.01.8016'' and a list of steps, all marked with a green checkmark to indicate success. The steps are: Clone from backup 'SnapCenter_LocalSnap_Hourly_05-13-2022_03.00.01.8016', hana-7.sapcc.stl.netapp.com, Clone, Application Pre Clone, Storage Clone, Application Post Clone, Register Clone Metadata, Application Clean-Up, Data Collection, and Agent Finalize Workflow. At the bottom of the window, there is a status bar with an information icon, the text 'Task Name: Clone Start Time: 05/13/2022 5:24:36 AM End Time: 05/13/2022 5:25:05 AM', and three buttons: 'View Logs', 'Cancel job', and 'Close'.

Job Details

Clone from backup 'SnapCenter_LocalSnap_Hourly_05-13-2022_03.00.01.8016'

- ✓ ▾ Clone from backup 'SnapCenter_LocalSnap_Hourly_05-13-2022_03.00.01.8016'
- ✓ ▾ hana-7.sapcc.stl.netapp.com
- ✓ ▾ Clone
 - ▶ Application Pre Clone
 - ▶ Storage Clone
 - ▶ Application Post Clone
 - ▶ Register Clone Metadata
 - ▶ Application Clean-Up
 - ▶ Data Collection
 - ▶ Agent Finalize Workflow

Task Name: Clone Start Time: 05/13/2022 5:24:36 AM End Time: 05/13/2022 5:25:05 AM

View Logs Cancel job Close

スクリプトのログファイル `sc-system-refresh` には、マウント処理とリカバリ処理に対して実行されるさまざまな手順が表示されます。

```
20201201052124###hana-1###sc-system-refresh.sh: Recover system database.
20201201052124###hana-1###sc-system-refresh.sh:
/usr/sap/SS1/HDB00/exe/Python/bin/python
/usr/sap/SS1/HDB00/exe/python_support/recoverSys.py --command "RECOVER
DATA USING SNAPSHOT CLEAR LOG"
20201201052156###hana-1###sc-system-refresh.sh: Wait until SAP HANA
database is started ....
20201201052156###hana-1###sc-system-refresh.sh: Status: GRAY
20201201052206###hana-1###sc-system-refresh.sh: Status: GREEN
20201201052206###hana-1###sc-system-refresh.sh: SAP HANA database is
started.
20201201052206###hana-1###sc-system-refresh.sh: Source system has a single
tenant and tenant name is identical to source SID: SS1
20201201052206###hana-1###sc-system-refresh.sh: Target tenant will have
the same name as target SID: SS1.
20201201052206###hana-1###sc-system-refresh.sh: Recover tenant database
SS1.
20201201052206###hana-1###sc-system-refresh.sh:
/usr/sap/SS1/SYS/exe/hdb/hdbsql -U SS1KEY RECOVER DATA FOR SS1 USING
SNAPSHOT CLEAR LOG
0 rows affected (overall time 34.773885 sec; server time 34.772398 sec)
20201201052241###hana-1###sc-system-refresh.sh: Checking availability of
Indexserver for tenant SS1.
20201201052241###hana-1###sc-system-refresh.sh: Recovery of tenant
database SS1 succesfully finished.
20201201052241###hana-1###sc-system-refresh.sh: Status: GREEN
After the recovery operation, the HANA database is running and the data
volume is mounted at the target host.
hana-1:/mnt/log-backup # df
Filesystem 1K-blocks Used Available Use% Mounted on
192.168.175.117:/SS1_repair_log_mnt00001 262144000 760320 261383680 1%
/hana/log/SS1/mnt00001
192.168.175.100:/sapcc_share 1020055552 53486592 966568960 6% /mnt/sapcc-
share
192.168.175.117:/SS1_repair_log_backup 104857600 512 104857088 1%
/mnt/log-backup
192.168.175.117:/SS1_shared_Clone_05132205140448713/usr-sap 262144064
10090496 252053568 4% /usr/sap/SS1
192.168.175.117:/SS1_shared_Clone_05132205140448713/shared 262144064
10090496 252053568 4% /hana/shared
192.168.175.117:/SS1_data_mnt00001_Clone_0421220520054605 262144064
3732864 258411200 2% /hana/data/SS1/mnt00001
```

これでSAP HANAシステムが利用可能になり、リペアシステムなどとして使用できます。

追加情報およびバージョン履歴の参照先

このドキュメントに記載されている情報の詳細については、以下のドキュメントや Web サイトを参照してください。

- ["SAP Business Application and SAP HANA Database Solutions \(netapp.com\) "](#)
- ["SnapCenter を使用した SAP HANA のバックアップとリカバリ"](#)
- ["ファイバチャネルプロトコルを使用したNetAppオールフラッシュFASシステム上のSAP HANA"](#)
- ["NFSを使用したNetAppオールフラッシュFASシステム上のSAP HANA"](#)
- ["Amazon FSX上のSAP HANA for NetApp ONTAP - SnapCenter を使用したバックアップとリカバリ"](#)
- ["Ansibleを使用したNetApp SAP Landscape Managementの統合"](#)
- ["Libelle SystemCopyによるSAPシステムのコピー操作の自動化 \(netapp.com\) "](#)
- ["AlpacaとNetApp SnapCenterによるSAPシステムのコピー、更新、クローニングのワークフローの自動化"](#)
- ["AvantraとNetApp SnapCenterを使用したSAPシステムのコピー、更新、クローニングのワークフローの自動化"](#)

バージョン	日付	ドキュメントバージョン履歴
バージョン 1.0 以降	2018 年 2 月	初版リリース
バージョン 2.0 以降	2021年2月	SnapCenter 4.3および改良された自動化スクリプトを対象とした完全な書き換え+システム更新処理とシステムクローン処理のワークフローの説明が新しくなりました。
バージョン 3.0 以降	2022年5月	SnapCenter 4.6 P1で変更されたワークフローに対応
バージョン 4.0 以降	2024年7月	このドキュメントはオンプレミスのNetAppシステムを対象としています。 FSx for ONTAPとAzure NetApp Files 新しいSnapCenter 5.0の操作クローンの作成と削除のワークフローでのマウントとアンマウント+ファイバチャネルSANの具体的な手順の追加 Azure NetApp Filesの具体的な手順の追加+対応したシンプルなスクリプト+有効なSAP HANAに必要な手順の追加 sc-system-refresh
バージョン4.1	2026年2月	HANA暗号化とSAPローカルセキュアストアLSSに必要な手順を追加しました

Libelle SystemCopyによるSAPシステムのコピー処理の自動化

Libelle SystemCopyによるSAPシステムのコピー処理の自動化

SAPのライフサイクル管理を最適化するネットアップのソリューションは、SAP AnyDB とSAP HANAデータベースに統合されています。さらに、ネットアップは、アプリケー

ションと統合された効率的なデータ保護と、SAPテストシステムの柔軟なプロビジョニングを組み合わせた、SAPのライフサイクル管理ツールに統合しています。

著者：

Holger Zecha、Tobias Brandl、ネットアップFranz Diegruber、Libelle

今日のダイナミックなビジネス環境では、企業は継続的なイノベーションを提供し、変化する市場に迅速に対応する必要があります。このような競争の激しい状況下で、業務プロセスの柔軟性を高める企業は、市場の需要に効果的に対応できます。

市場ニーズの変化は、企業のSAP環境にも影響を及ぼし、定期的な統合、変更、更新が必要になります。IT部門は、これらの変更を、より少ないリソースでより短い期間で実装する必要があります。変更を導入する際のリスクを最小限に抑えるには、本番環境の実際のデータを使用するSAPシステムを追加で必要とする、テストとトレーニングを徹底的に実施する必要があります。

従来のSAPライフサイクル管理アプローチでは、このようなシステムは主に手動プロセスに基づいてプロビジョニングされます。このような手動プロセスは、ミス避けられず、時間もかかることが多く、イノベーションの遅れやビジネス要件への対応の遅れにつながります。

SAPのライフサイクル管理を最適化するネットアップのソリューションは、SAP AnyDBとSAP HANAデータベースに統合されています。さらに、ネットアップは、アプリケーションと統合された効率的なデータ保護と、SAPテストシステムの柔軟なプロビジョニングを組み合わせた、SAPのライフサイクル管理ツールに統合しています。

このようなネットアップのソリューションは、大規模なデータベースでも膨大なデータを効率的に管理するという問題を解決しますが、SAPシステムの完全なエンドツーエンドのコピーおよび更新処理には、ソースSAPシステムのIDをターゲットシステムに完全に変更するためのコピー前処理とコピー後処理が含まれている必要があります。SAPでは、必要なアクティビティについて説明します ["SAP同種システムのコピーガイド"](#)。ネットアップのパートナー様は、手動プロセスの回数をさらに減らし、SAPシステムのコピープロセスの品質と安定性を高めるために ["Libelle"](#) を開発しました ["Libelle SystemCopy \(LSC\)"](#) ツール。Libelleと共同で、SAPシステムコピー用のネットアップソリューションをLSCに統合して提供しています ["完全なエンドツーエンドの自動システムコピーを、記録的な時間で作成します"](#)。

アプリケーションに統合されたSnapshotコピー処理

アプリケーションと整合性のあるNetApp Snapshotコピーをストレージレイヤに作成する機能は、本ドキュメントで説明するシステムコピー処理とシステムクローン処理の基盤となります。ストレージベースのSnapshotコピーは、ネットアップのSnapCenter Plug-in for SAP HANAまたはSAPのネイティブONTAPシステム上の任意のDBを使用して作成されます ["Microsoft Azure Application Consistent Snapshotツール"](#) (AzAcSnap)、およびMicrosoft Azureで実行されるSAP HANAとOracleデータベースによって提供されるインターフェイス。SAP HANAを使用している場合は、SnapCenterとAzAcSnapがSnapshotコピーをSAP HANAバックアップカタログに登録することで、リストアとリカバリ、およびクローニング処理に使用できます。

オフサイトバックアップやディザスタリカバリのデータレプリケーションを実行できます

アプリケーションと整合性のあるSnapshotコピーは、ストレージレイヤから、オンプレミスのSnapCenterで制御されるオフサイトのバックアップサイトやディザスタリカバリサイトにレプリケートできます。レプリケーションはブロック変更に基づいているため、スペースと帯域幅を効率的に使用できます。クロスリージョンレプリケーション (CRR) 機能を使用してAzureリージョン間でAzure NetApp Files ボリュームを効率的にレプリケートすることで、AzureでAzure NetApp Files 実行されているSAP HANAシステムとOracleシステムにも同じテクノロジーが使用されます。

SAPシステムのコピー処理やクローニング処理には、任意のSnapshotコピーを使用できます

ネットアップのテクノロジーとソフトウェアの統合により、ソースシステムの任意のSnapshotコピーを使用してSAPシステムのコピーやクローニングを実行できます。このSnapshotコピーは、SAP本番用システムと同じストレージ、オフサイトのバックアップに使用するストレージ（AzureのAzure NetApp Files バックアップなど）、またはディザスタリカバリサイトのストレージ（Azure NetApp Files CRRターゲットボリューム）のいずれかから選択できます。この柔軟性により、必要に応じて開発用システムとテスト用システムを本番用システムと分離し、ディザスタリカバリサイトでのディザスタリカバリのテストなど、さまざまなシナリオに対応することができます。

統合による自動化

SAPテストシステムのプロビジョニングにはさまざまなシナリオとユースケースがあり、自動化のレベルに関してもさまざまな要件が存在する場合があります。ネットアップのSAP向けソフトウェア製品は、SAPなどのサードパーティベンダーが提供するデータベースやライフサイクル管理製品（Libelleなど）と統合されており、さまざまなシナリオや自動化のレベルをサポートします。

SAP HANAとSAP AnyDBまたはAzure上のAzAcSnap向けのプラグインを備えたNetApp SnapCenter を使用して、アプリケーションと整合性のあるSnapshotコピーに基づいて必要なストレージボリュームのクローンをプロビジョニングし、開始されたSAPデータベースまで、必要なホストとデータベースのすべての処理を実行できます。SAPシステムのコピー、システムクローン、システムの更新、SAPのポストプロセスなどの追加の手動手順が必要になる場合があります。詳細については、次のセクションで説明します。

完全に自動化されたエンドツーエンドのSAPテストシステムのプロビジョニングまたは更新は、Libelle SystemCopy（LSC）自動化を使用して実行できます。SnapCenter またはAzAcSnapのLSCへの統合については、このドキュメントで詳しく説明します。

Libelle SystemCopy

Libelle SystemCopyは、完全に自動化されたシステムコピーとランドスケープコピーを作成するフレームワークベースのソフトウェア解決策です。ボタンを口頭で操作することで、QAおよびテストシステムを新しい本番データで更新できます。Libelle SystemCopyは、従来のすべてのデータベースとオペレーティングシステムをサポートし、すべてのプラットフォームに独自のコピーメカニズムを提供しますが、同時にバックアップ/リストア手順や、NetApp SnapshotコピーやNetApp FlexCloneボリュームなどのストレージツールを統合します。システムのコピー中に必要なアクティビティは、SAP ABAPスタックの外部から制御されます。そのため、SAPアプリケーションでは転送などの変更は必要ありません。一般に、システムコピー手順を正常に完了するために必要な手順は、次の4つに分類できます。

- *確認フェーズ。*関係するシステム環境を確認してください。
- *プリフェーズ*システムコピーのターゲット・システムを準備します。
- *コピーフェーズ*ソースからターゲット・システムに実際の本番データベースのコピーを提供します
- *ポスト・フェーズ*：コピー後のすべてのタスクで同機種システムのコピー手順を完了し、ターゲット・システムを更新します

コピーフェーズでは、NetApp SnapshotとFlexCloneの機能を使用して、最大のデータベースであっても、必要な時間を数分に短縮できます。

チェックフェーズ、プリフェーズ、およびポストフェーズの場合、LSCには、一般的な更新操作の95%をカバーする450以上の事前設定されたタスクが付属しています。その結果、LSCはSAP標準に従って自動化を推進します。LSCはソフトウェアで定義されるため、システム更新プロセスを簡単に調整および拡張して、お客様のSAP環境固有のニーズに対応できます。

SAPシステムの更新とクローニングのユースケース

ソースシステムのデータをターゲットシステムで使用できるようにする必要があるシナリオは複数あります。

- 品質保証およびテストおよびトレーニングシステムの定期的な更新
- 論理的な破損に対処するために、システム環境の障害修復または修復を行う
- ディザスタリカバリのテストシナリオ

修復システムと災害復旧テスト システムは通常、更新されたテストおよびトレーニング システム用の SAP システム クローン（大規模な後処理操作は不要）を使用して提供されますが、ソース システムとの共存を可能にするには、これらの後処理手順を適用する必要があります。したがって、このドキュメントでは、SAP システム更新シナリオに焦点を当てています。さまざまなユースケースの詳細については、技術レポート"[SnapCenter を使用して SAP HANA システムのコピーおよびクローン処理を自動化](#)"をご覧ください。

このドキュメントの残りの部分は、2つの部分に分かれています。最初のパートでは、オンプレミスのNetApp ONTAP システムで実行されているSAP HANAシステムとSAP AnyDBシステム向けのNetApp SnapCenter とLibelle SystemCopyの統合について説明します。2番目のパートでは、Microsoft Azureで実行されているSAP HANAシステム用のLSCと、提供されているAzure NetApp Files とのAzAcSnapの統合について説明します。ONTAP テクノロジーの基盤は同じですが、Azure NetApp Files は、ネイティブONTAP インストールとは異なるインターフェイスとツールの統合（AzAcSnapなど）を提供します。

LSCおよびSnapCenter でSAP HANAシステムが更新されます

このセクションでは、LSCをNetApp SnapCenter に統合する方法について説明します。LSCとSnapCenter の統合では、SAPでサポートされるすべてのデータベースがサポートされます。ただし、SAP HANAはSAP AnyDBでは使用できない中央通信ホストを提供するため、SAP AnyDBとSAP HANAを区別する必要があります。

SAP AnyDB用のデフォルトのSnapCenter エージェントとデータベースプラグインのインストールは、データベースサーバ用の対応するデータベースプラグインに加えて、SnapCenter エージェントからのローカルインストールです。

このセクションでは、例として、LSCとSnapCenter の統合についてSAP HANAデータベースを使用して説明します。前述したSAP HANAの場合、SnapCenter エージェントとSAP HANAデータベースプラグインのインストール方法は2つあります。

- *標準のSnapCenter エージェントとSAP HANA Plug-inのインストール。*標準のインストールでは、SnapCenter エージェントとSAP HANAプラグインはSAP HANAデータベースサーバにローカルでインストールされます。
- *中央通信ホストを使用するSnapCenter のインストール。*中央通信ホストは、SnapCenter エージェント、SAP HANAプラグイン、および複数のSAP HANAシステムのSAP HANAデータベースのバックアップとリストアに必要なすべてのデータベース関連操作を処理するHANAデータベースクライアントとともにインストールされます。したがって、中央通信ホストに完全なSAP HANAデータベースシステムをインストールする必要はありません。

これらのさまざまなSnapCenterエージェントとSAP HANAデータベースプラグインのインストールオプションの詳細については、技術レポートを参照してください。"[SnapCenter を使用した SAP HANA のバックアップとリカバリ](#)"。

次の項では、標準インストールまたは中央通信ホストを使用したLSCとSnapCenter の統合の違いについて説明します。特に、インストールオプションと使用するデータベースに関係なく、強調表示されない設定手順は

すべて同じです。

ソースデータベースから自動化されたSnapshotコピーベースのバックアップを実行し、新しいターゲットデータベースのクローンを作成するには、LSCとSnapCenterの統合で["SnapCenter を使用して SAP HANA システムのコピーおよびクローン処理を自動化"](#)に記載されている設定オプションとスクリプトを使用します。

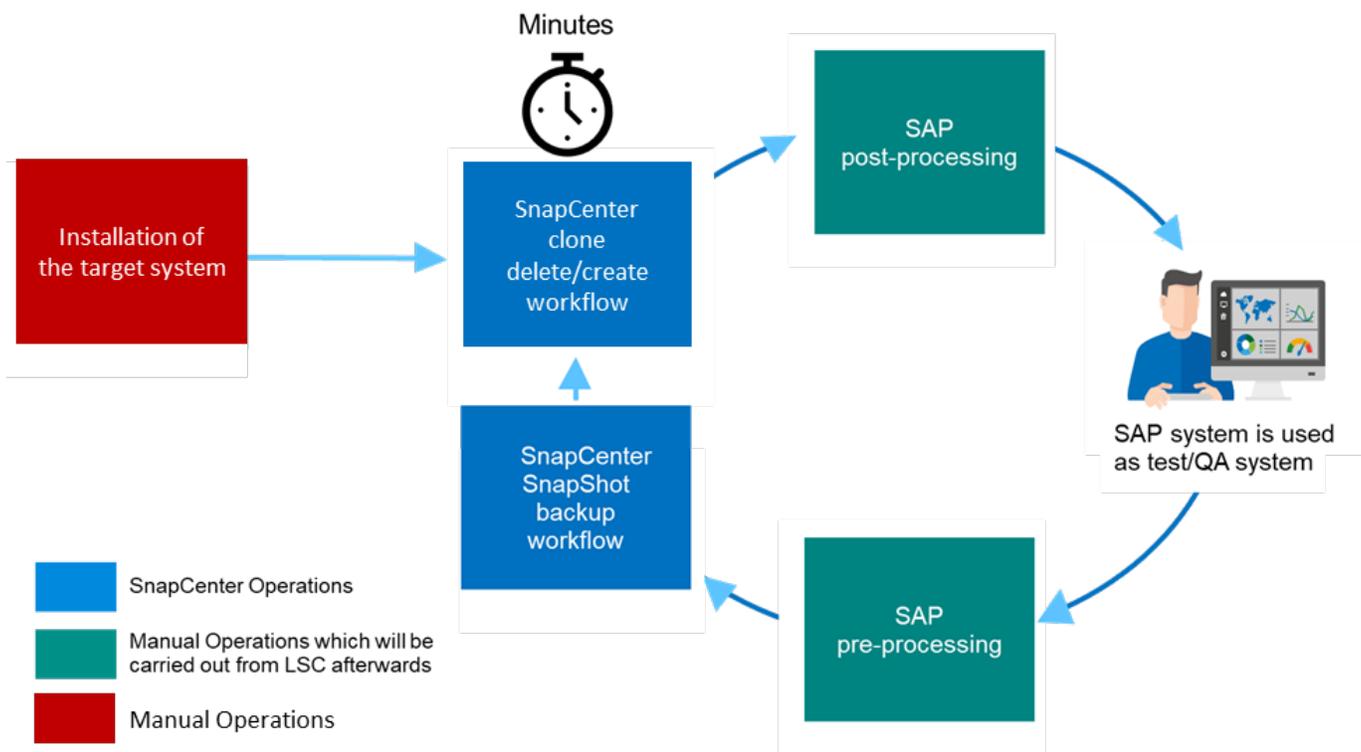
概要

次の図に、LSCを使用しないSnapCenter によるSAPシステムの更新ライフサイクルの一般的なワークフローを示します。

1. ターゲットシステムの初期インストールと準備を1回だけ行います。
2. 手動前処理（ライセンス、ユーザー、プリンターなどのエクスポート）。
3. 必要に応じて、ターゲットシステム上の既存のクローンを削除します。
4. ソースシステムの既存のSnapshotコピーを、SnapCenter によって実行されるターゲットシステムにクローニングすること。
5. SAP後処理の手動操作（ライセンスのインポート、ユーザー、プリンタ、バッチジョブの無効化など）。
6. その後、システムをテストシステムまたはQAシステムとして使用できます。
7. 新しいシステムの更新が要求されると、手順2でワークフローが再開されます。

SAPをご利用のお客様は、次の図に示す手動の手順が緑で表示されているため、時間がかかり、ミスが発生しやすくなっています。LSCとSnapCenter の統合を使用する場合、これらの手動手順は、信頼性の高い反復可能な方法でLSCを使用して実行されます。この方法では、内部および外部の監査に必要なすべてのログが使用されます。

次の図は、SnapCenterベースのSAPシステム更新手順 の概要を示しています。



前提条件および制限事項

次の前提条件を満たしている必要があります。

- SnapCenter をインストールする必要があります。ソースシステムとターゲットシステムは、標準インストールまたは中央通信ホストを使用して、SnapCenter で設定する必要があります。Snapshot コピーはソースシステム上に作成できます。
- 次の図に示すように、SnapCenter でストレージバックエンドが正しく設定されている必要があります。

Storage Connections

<input type="checkbox"/>	Name	IP	Cluster Name	User Name	Controller License
<input type="checkbox"/>	svm-trident		grenada.muccbc.hq.netapp.com		✓
<input type="checkbox"/>	svm-sap02	10.65.58.253	grenada.muccbc.hq.netapp.com		✓
<input type="checkbox"/>	svm-sap01	10.65.58.252	grenada.muccbc.hq.netapp.com		✓

次の2つの図は、SnapCenter エージェントとSAP HANAプラグインが各データベースサーバでローカルにインストールされる標準インストールを示しています。

SnapCenter エージェントと適切なデータベースプラグインをソースデータベースにインストールする必要があります。

<input type="checkbox"/>	Name	Type	System	Plug-in	Version	Overall Status
<input type="checkbox"/>	sap-inx35.muccbc.hq.netapp.com	Linux	Stand-alone	UNIX, SAP HANA	4.3.1	● Running

SnapCenter エージェントと適切なデータベースプラグインをターゲット・データベースにインストールする必要があります。

<input type="checkbox"/>	sap-inx36.muccbc.hq.netapp.com	Linux	Stand-alone	UNIX, SAP HANA	4.3.1	● Running
--------------------------	--------------------------------	-------	-------------	----------------	-------	-----------

次の図は、SnapCenter エージェント、SAP HANAプラグイン、およびSAP HANAデータベースクライアントが一元化されたサーバ（SnapCenter サーバなど）にインストールされ、ランドスケープ内の複数のSAP HANAシステムを管理する、中央通信ホスト環境を示しています。

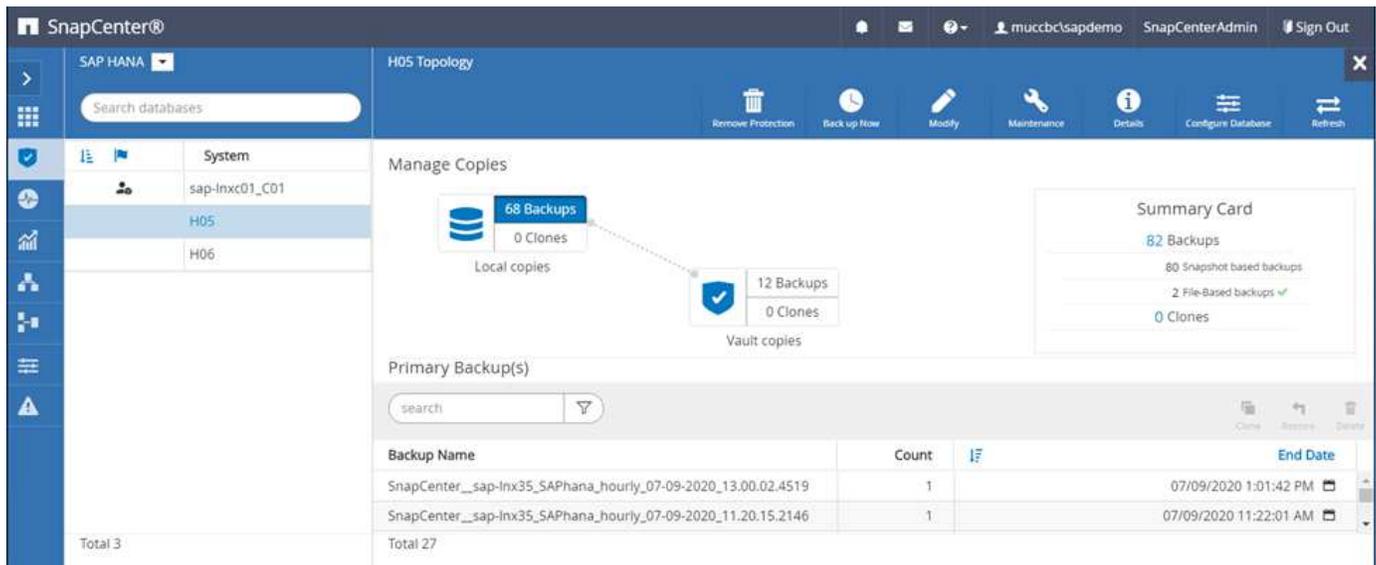
SnapCenter エージェント、SAP HANAデータベースプラグイン、およびHANAデータベースクライアントが、中央の通信ホストにインストールされている必要があります。

Managed Hosts Disks Shares Initiator Groups iSCSI Session

Search by Name

<input type="checkbox"/>	Name	Type	System	Plug-in	Version	Overall Status
<input type="checkbox"/>	dbh03.muccbc.hq.netapp.com	Linux	Stand-alone	UNIX, SAP HANA	4.4	● Upgrade available (optional)
<input type="checkbox"/>	sap-sc-demo-dev.muccbc.hq.netapp.com	Windows	Stand-alone	Microsoft Windows Server, SAP HANA	4.5	● Running
<input type="checkbox"/>	sap-win02.muccbc.hq.netapp.com	Windows	Stand-alone	Microsoft Windows Server	4.5	● Running

Snapshot コピーを正常に作成できるように、SnapCenter でソースデータベースのバックアップが適切に設定されている必要があります。



LSCマスターおよびLSCワーカーがSAP環境にインストールされている必要があります。この展開では、SnapCenter サーバにLSCマスターをインストールし、ターゲットのSAPデータベースサーバにLSCワーカーをインストールします。このサーバは更新する必要があります。詳細については、「」を参照してください[ラボのセットアップ]

ドキュメント：

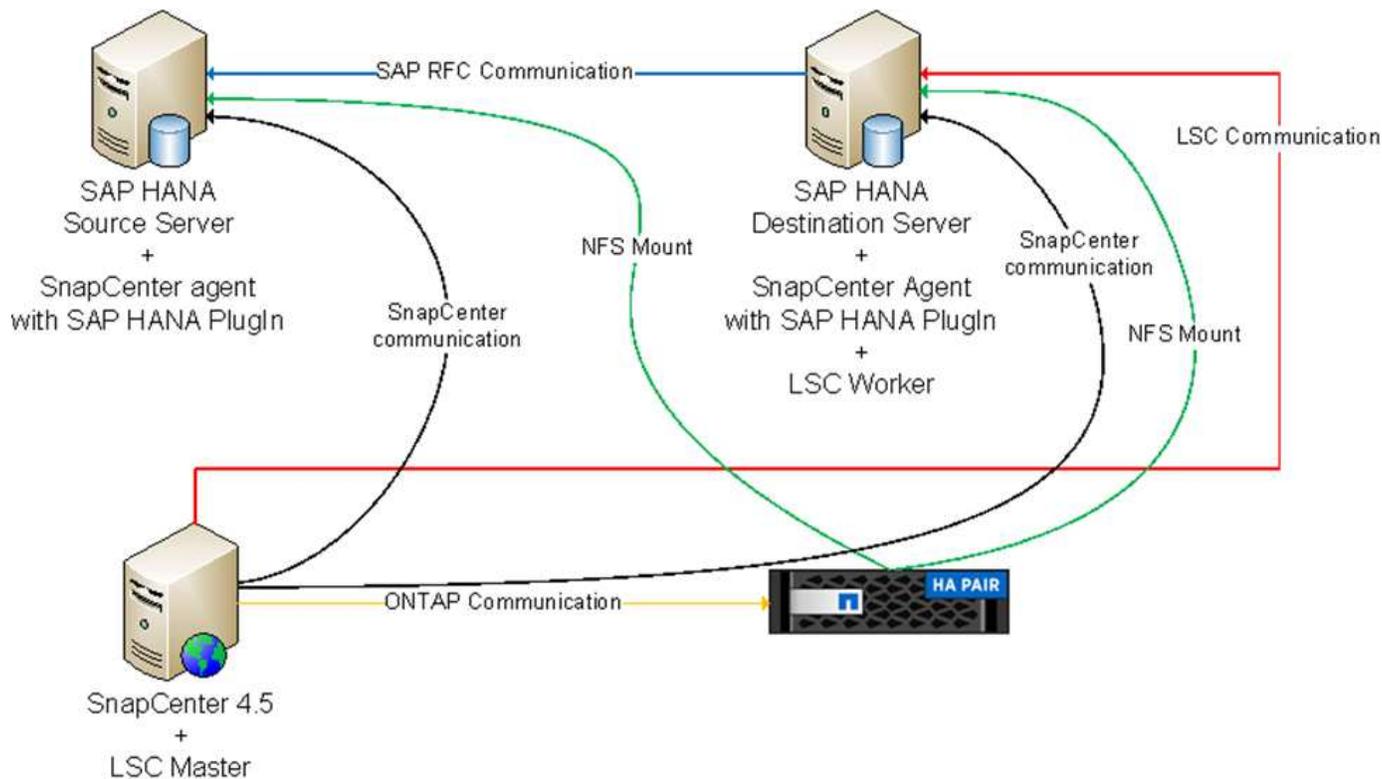
- "SnapCenter ドキュメントセンター"
- "SnapCenter Plug-in for Oracle Database"
- "SnapCenter を使用した SAP HANA のバックアップとリカバリ"
- "SnapCenter を使用して SAP HANA システムのコピーおよびクローン処理を自動化"
- "SnapCenter 4.6コマンドレットリファレンスガイド"

ラボのセットアップ

このセクションでは、デモデータセンターでセットアップされたアーキテクチャの例について説明します。セットアップは、標準的なインストールと、中央の通信ホストを使用したインストールに分かれています。

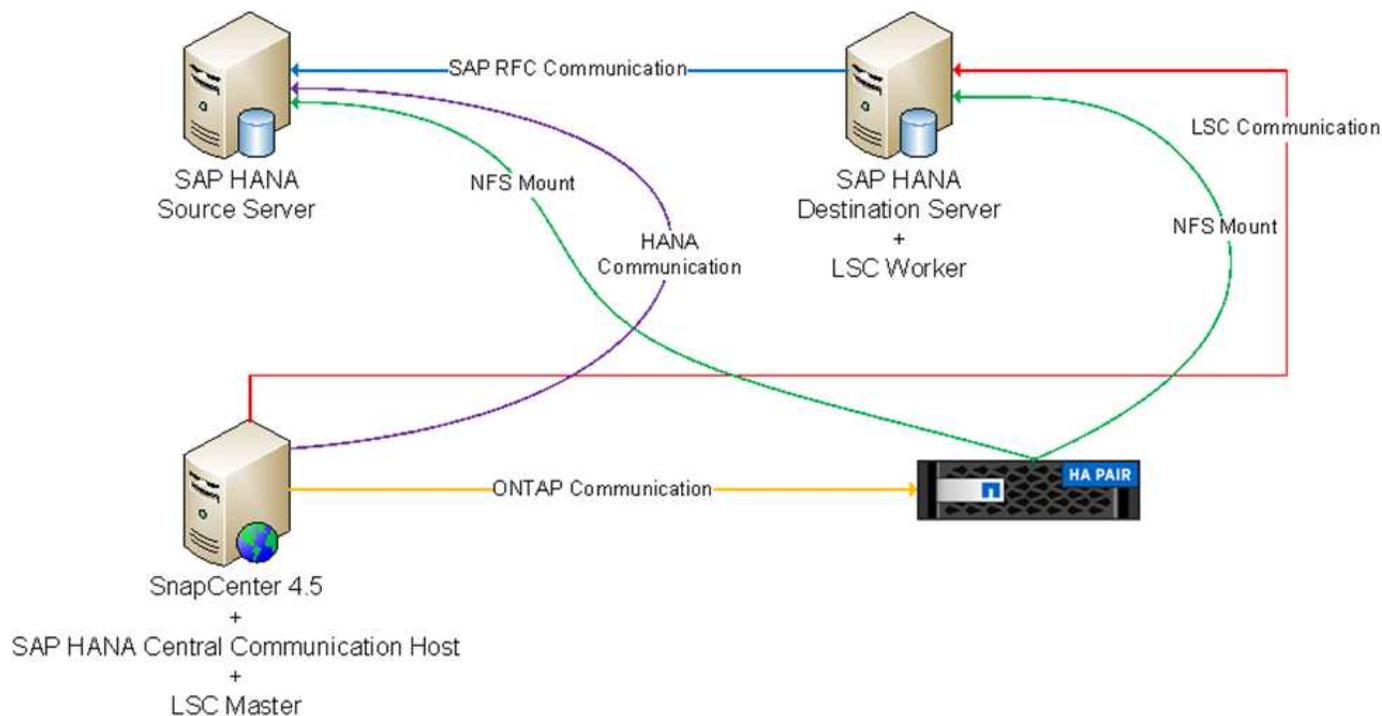
標準インストール

次の図に、SnapCenter エージェントとデータベースプラグインが、ソースおよびターゲットのデータベースサーバ上にローカルにインストールされた標準インストールを示します。このラボ環境では、SAP HANA Plug-inをインストールしました。また、ターゲットサーバにLSCワーカーもインストールされています。簡素化と仮想サーバ数の削減のために、SnapCenter サーバにLSCマスターをインストールしました。次の図は、各種コンポーネント間の通信を示しています。



セントラルコミュニケーションホスト

次の図に、中央通信ホストを使用した設定を示します。この構成では、SnapCenter エージェントと SAP HANA Plug-in および HANA データベースクライアントを専用サーバにインストールしました。このセットアップでは、SnapCenter サーバを使用して中央通信ホストをインストールしました。さらに、LSC ワーカーが再びターゲットサーバにインストールされました。簡素化と仮想サーバ数の削減のため、SnapCenter サーバに LSC マスターもインストールすることにしました。次の図に、異なるコンポーネント間の通信を示します。



Libelle SystemCopyの初期1回限りの準備手順

LSCインストールには、次の3つの主要コンポーネントがあります。

- *LSC master.*という名前が示すように、Libelleベースのシステムコピーの自動ワークフローを制御するマスターコンポーネントです。デモ環境では、LSCマスターがSnapCenter サーバにインストールされています。
- *LSCワーカー。* LSCワーカーは、通常はターゲットSAPシステムで実行されるLibelleソフトウェアの一部であり、自動システムコピーに必要なスクリプトを実行します。デモ環境では、ターゲットのSAP HANAアプリケーションサーバにLSCワーカーがインストールされています。
- *LSC衛星。* LSC衛星は、それ以降のスクリプトを実行する必要があるサードパーティシステム上で実行されるLibelleソフトウェアの一部です。LSCマスターは、LSCサテライトシステムの役割も同時に果たすことができます。

次の図に示すように、最初にLSC内のすべての関連システムを定義しました。

- *172.30.15.35.* SAPソースシステムとSAP HANAソースシステムのIPアドレス。
- *172.30.15.3.*この構成のLSCマスターおよびLSCサテライトシステムのIPアドレス。SnapCenter サーバにLSCマスターをインストールしたため、SnapCenter サーバのインストール時にインストールされたSnapCenter 4.x PowerShellコマンドレットは、このWindowsホストですでに使用できます。そのため、このシステムに対してLSCサテライトロールを有効にし、このホストですべてのSnapCenter PowerShellコマンドレットを実行することにしました。別のシステムを使用する場合は、SnapCenter のマニュアルに従って、このホストにSnapCenter PowerShellコマンドレットをインストールしてください。
- **172.30.15.36** SAPデスティネーションシステム、SAP HANAデスティネーションシステム、およびLSCワーカーのIPアドレス。

IPアドレス、ホスト名、完全修飾ドメイン名の代わりに使用することもできます。

次の図は、マスタ、ワーカー、サテライト、SAPソース、SAPターゲットのLSC構成を示しています。ソースデータベースおよびターゲットデータベース。

System Identifier	Worker	Source SAP	Source Database	Target SAP	Target Database	Satellite System
172.30.15.35		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
172.30.15.3	172.30.15.3:9000	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
172.30.15.36	172.30.15.36:9000	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

メイン統合のためには、設定手順を標準インストールと中央通信ホストを使用したインストールに再度分ける必要があります。

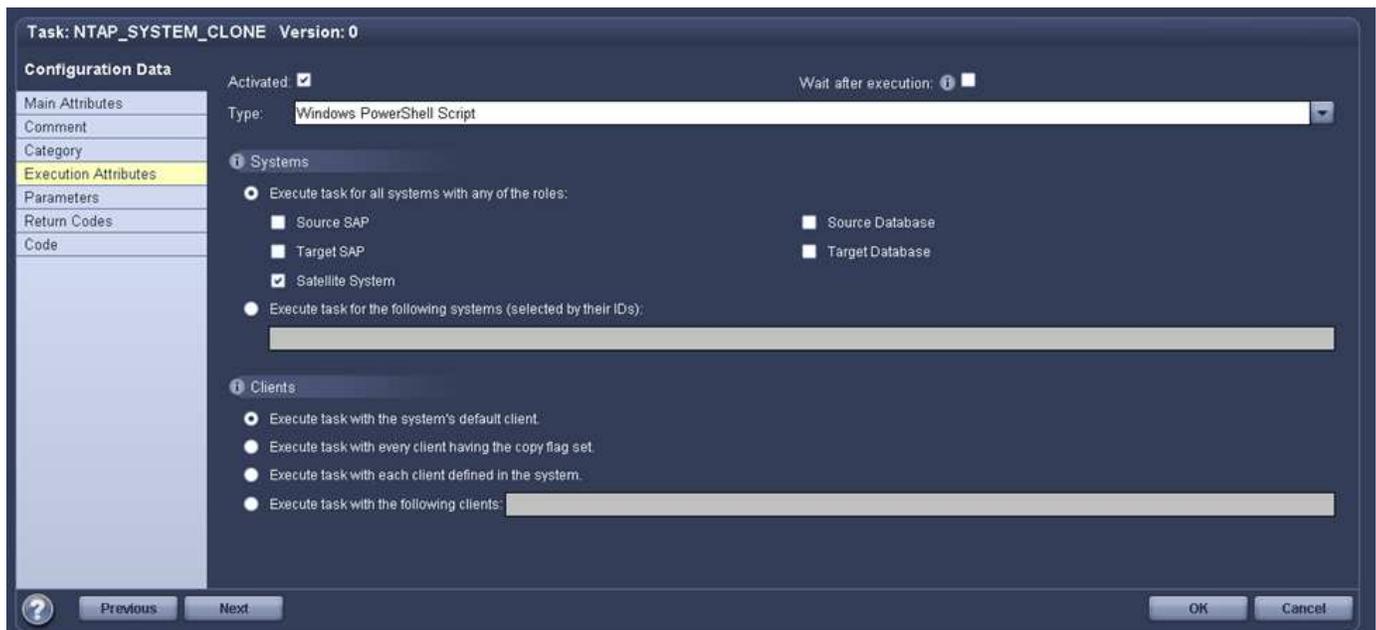
標準インストール

このセクションでは、SnapCenter エージェントと必要なデータベースプラグインがソースシステムとターゲットシステムにインストールされている標準インストールを使用する場合に必要な設定手順について説明します。標準インストールを使用する場合は、クローンボリュームのマウントおよびターゲットシステムのリストアとリカバリに必要なすべてのタスクが、サーバ自体のターゲットデータベースシステムで実行されているSnapCenter エージェントから実行されます。これにより、SnapCenter エージェントの環境変数を使用して、クローン関連の詳細情報にアクセスできるようになります。したがって、LSCコピーフェーズでは、追加のタスクを1つだけ作成する必要があります。このタスクでは、ソース・データベース・システムでSnapshotコピーの処理を実行し、ターゲット・データベース・システムでクローンおよびリストアおよびリカバリの処理を実行します。SnapCenter に関連するすべてのタスクは、LSCタスク「NTAP_SYSTEM_CLONE」に入力されたPowerShellスクリプトを使用してトリガーされます。

次の図は、コピーフェーズのLSCタスクの設定を示しています。

copy	Copy Phase		phase
copy 1	NTAP_SYSTEM_CLONE	NetApp SnapShot and Clone	psh
copy 2	NTAP_SYSTEM_CLONE_CP	NetApp SnapShot and Clone	psh
copy 3	NTAP_MNT_RECOVER_CP	Mount Volume and Recover HANA Database	cmd
copy 4	LPDBBCKP	Backup Source DB in Filesystem	lsh
copy 5	LPDBCPYFLS	Copy DB Backup Files From Source to Target System.	lsh
copy 6	LTDBRESTORE	Restore DB Files	lsh
copy 7	LTDBRESTORE_TENANT	Restore DB Files for Tenant Database	lsh
post	Post Phase		phase

次の図は'NTAP_SYSTEM_CLONEプロセスの構成を示していますPowerShellスクリプトを実行するため、このWindows PowerShellスクリプトはサテライトシステム上で実行されます。この場合、これは、サテライトシステムとしても機能する、インストールされたLSCマスターを持つSnapCenter サーバです。

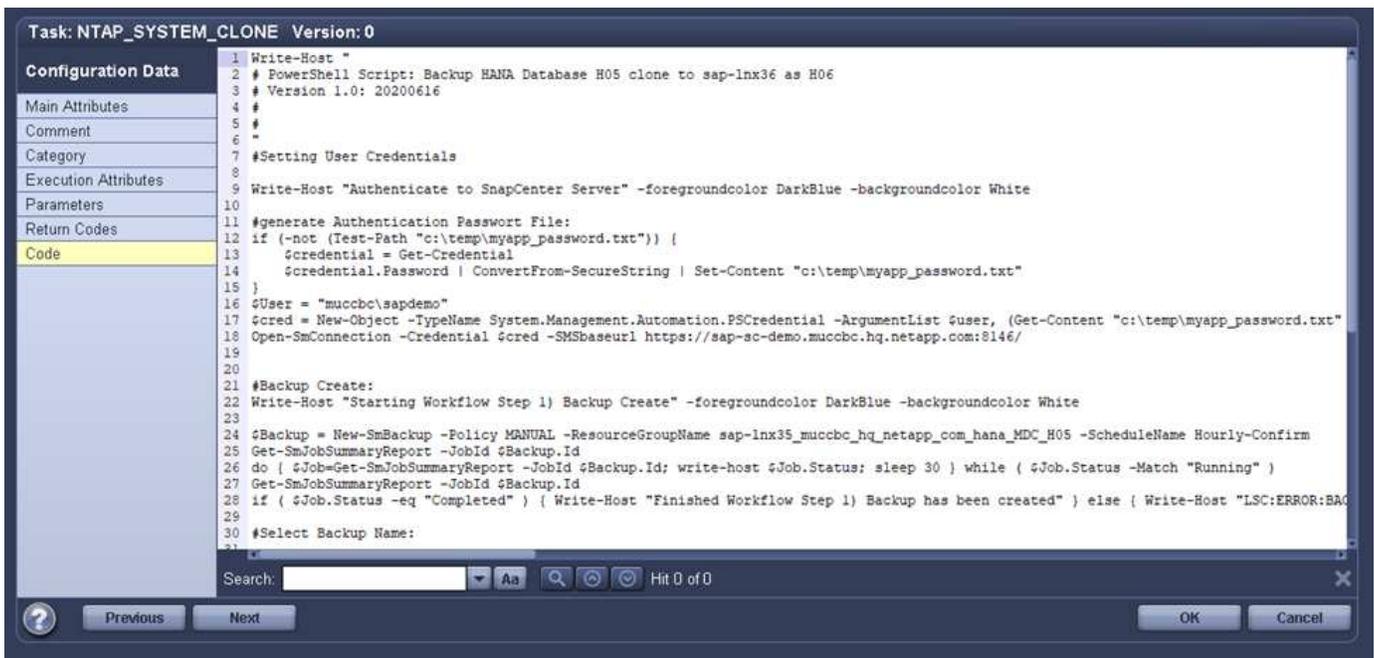


LSCは、Snapshotコピー、クローニング、およびリカバリ処理が成功したかどうかを認識する必要があるため、少なくとも2つの戻りコードタイプを定義する必要があります。次の図に示すように、1つのコードはスクリプトを正常に実行するためのもので、もう1つのコードはスクリプトの実行に失敗するためのものです。

- 実行が成功した場合は、スクリプトから標準出力に「LSC : OK」を書き込む必要があります。
- 実行に失敗した場合は、スクリプトから標準出力に「LSC : error」を書き込む必要があります。



次の図は、ソースデータベースシステムでSnapshotベースのバックアップを実行し、ターゲットデータベースシステムでクローンを実行する、PowerShellスクリプトの一部です。このスクリプトは、完全なものではありません。このスクリプトでは、LSCとSnapCenterの統合がどのように表示されるか、および設定がどの程度簡単かを示します。



スクリプトはLSCマスター（サテライトシステムでもある）上で実行されるため、SnapCenter サーバ上のLSCマスターは、SnapCenterでバックアップおよびクローニング操作を実行するための適切な権限を持つWindowsユーザとして実行する必要があります。ユーザに適切な権限があるかどうかを確認するには、SnapCenter UIでSnapshotコピーとクローンを実行できる必要があります。

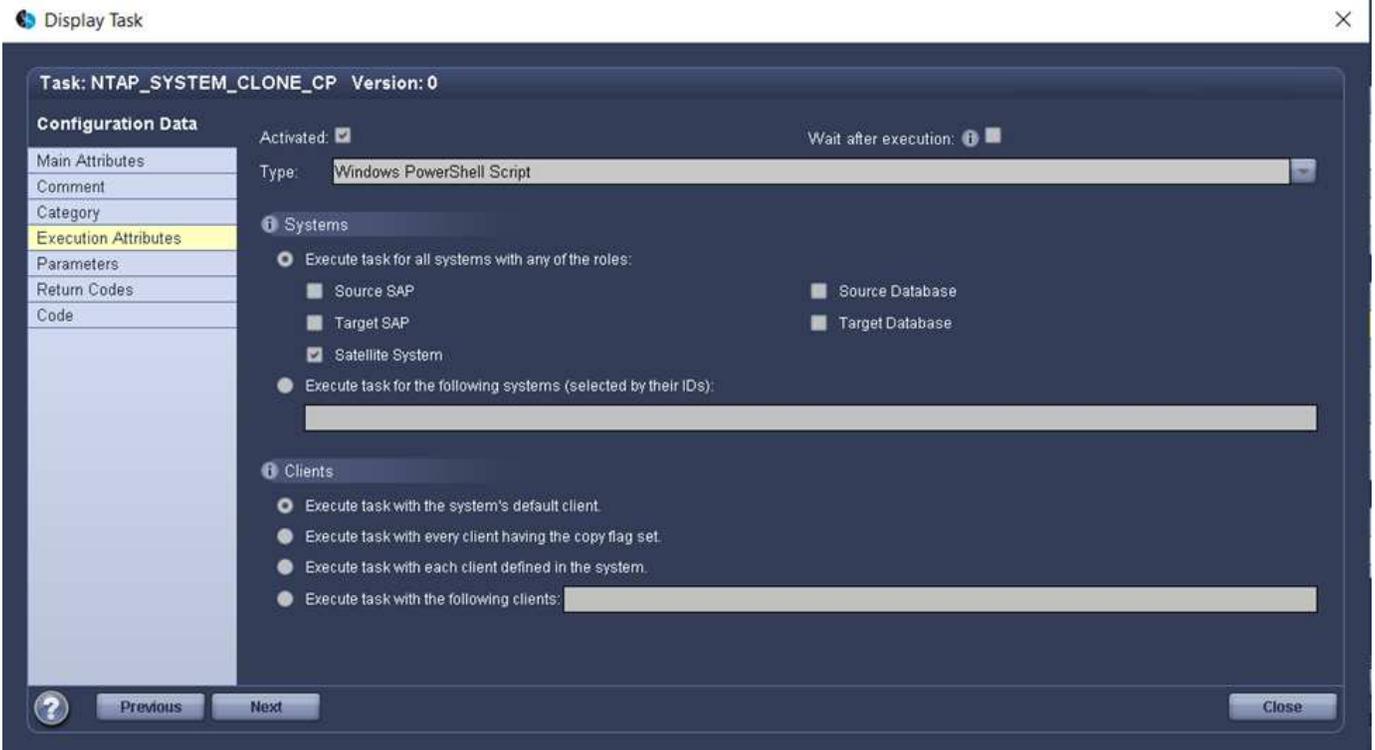
SnapCenter サーバ自体でLSCマスターおよびLSCサテライトを実行する必要はありません。LSCマスターおよびLSCサテライトは、任意のWindowsマシンで実行できます。LSCサテライトでPowerShellスクリプトを実行するための前提条件は、SnapCenter PowerShellコマンドレットがWindowsサーバにインストールされていることです。

中央通信ホストを使用してLSCとSnapCenterの間で統合する場合、コピーフェーズで実行する必要がある調整のみが実行されます。Snapshotコピーとクローンは、中央通信ホスト上のSnapCenter エージェントを使用して作成されます。したがって、新しく作成されたボリュームに関するすべての詳細情報は、ターゲットデータベースサーバではなく、中央通信ホストでのみ使用できます。ただし、これらの詳細は、クローンボリュームをマウントしてリカバリを実行するために、ターゲットデータベースサーバ上に必要です。これは、コピーフェーズで追加のタスクが2つ必要になる理由です。1つのタスクが中央通信ホストで実行され、1つのタスクがターゲットデータベースサーバで実行されます。これら2つのタスクを次の図に示します。

- * NTAP_SYSTEM_CLONE_CP。このタスクでは、中央通信ホストに必要なSnapCenter 機能を実行するPowerShellスクリプトを使用して、Snapshotコピーおよびクローンを作成します。したがって、このタスクはLSCサテライト上で実行されます。この場合、このインスタンスはWindows上で実行されるLSCマスターです。このスクリプトは、クローンおよび新しく作成されたボリュームに関するすべての詳細を収集し、2番目のタスク「NTAP_Mnt_RECOVER_CP」に渡します。このタスクは、ターゲットデータベースサーバで実行されるLSCワーカーで実行されます。
- * NTAP_Mnt_RECOVER_CP。*このタスクは、ターゲットSAPシステムとSAP HANAデータベースを停止し、古いボリュームをアンマウントして、前のタスク「NTAP_SYSTEM_CLONE_CP」から渡されたパラメータに基づいて、新しく作成されたストレージクローンボリュームをマウントします。その後、ターゲットのSAP HANAデータベースがリストアおよびリカバリされます。

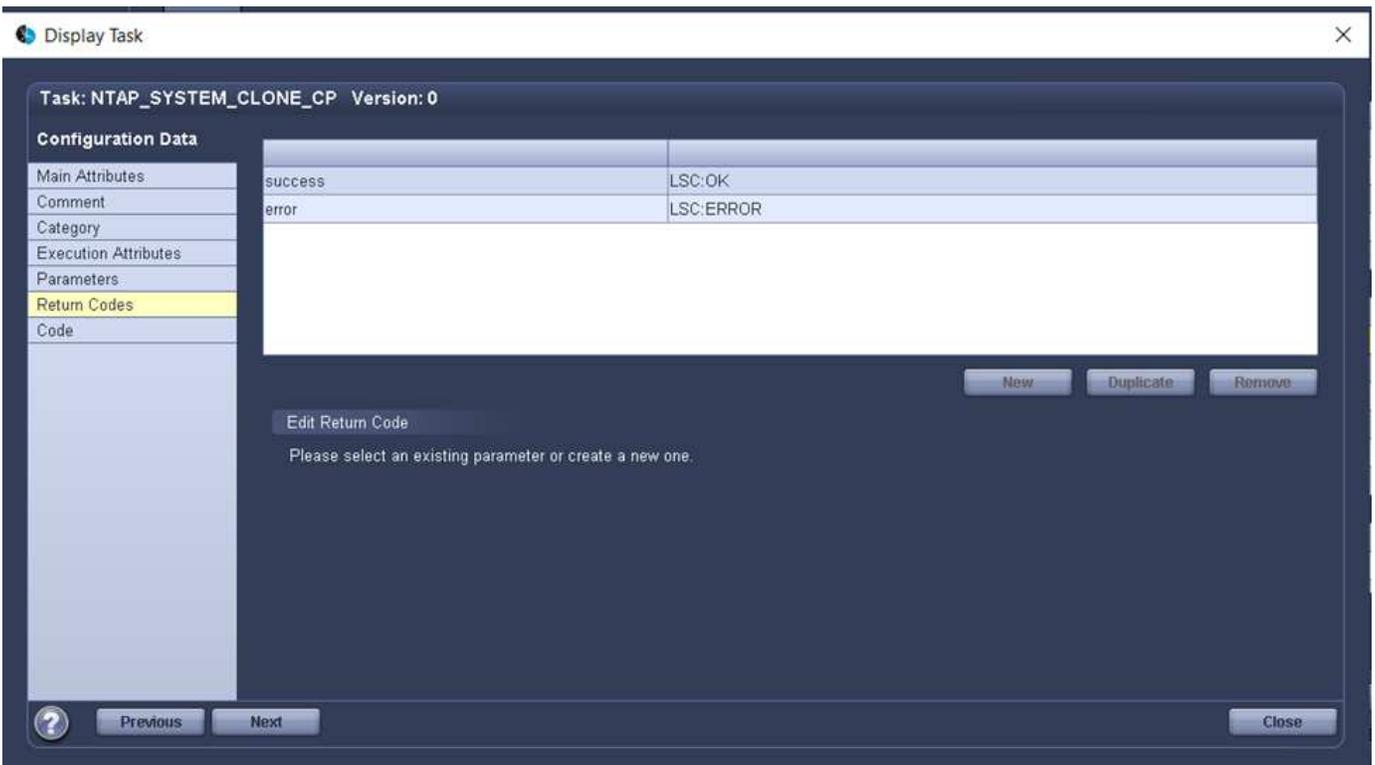
copy	Copy Phase		phase
copy 1	NTAP_SYSTEM_CLONE	NetApp SnapShot and Clone	psh
copy 2	NTAP_SYSTEM_CLONE_CP	NetApp SnapShot and Clone	psh
copy 3	NTAP_MNT_RECOVER_CP	Mount Volume and Recover HANA Database	cmd
copy 4	LPDBBCKP	Backup Source DB in Filesystem	lsh
copy 5	LPDBCOPYFLS	Copy DB Backup Files From Source to Target System.	lsh
copy 6	LTDBRESTORE	Restore DB Files	lsh
copy 7	LTDBRESTORE_TENANT	Restore DB Files for Tenant Database	lsh
post	Post Phase		phase

次の図は'タスク'NTAP_SYSTEM_CLONE_CP'の構成を示していますこれは、サテライトシステムで実行されるWindows PowerShellスクリプトです。この場合、サテライトシステムは、インストールされたLSCマスターを持つSnapCenter サーバになります。

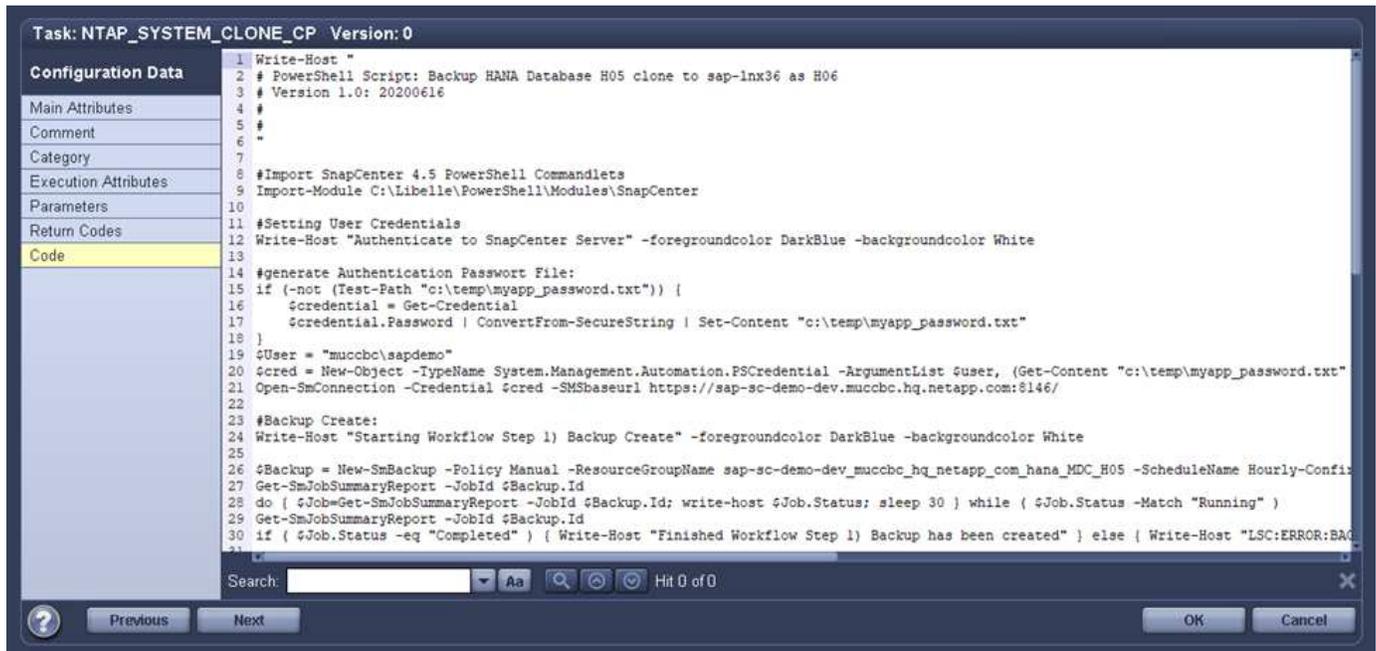


LSCは、Snapshotコピーおよびクローニング処理が成功したかどうかを認識する必要があるため、次の図に示すように、少なくとも2つの戻りコードタイプを定義する必要があります。スクリプトを正常に実行するには1つの戻りコードタイプ、スクリプトの実行に失敗するにはもう1つの戻りコードタイプです。

- 実行が成功した場合は、スクリプトから標準出力に「LSC:OK」を書き込む必要があります。
- 実行に失敗した場合は、スクリプトから標準出力に「LSC:error」を書き込む必要があります。



次の図は、中央通信ホスト上のSnapCenter エージェントを使用してSnapshotコピーとクローンを実行するために実行する必要があるPowerShellスクリプトの一部を示しています。このスクリプトは完了することを意図したものではありません。代わりに、スクリプトを使用して、LSCとSnapCenter の統合がどのように見えるか、および設定がどの程度簡単かを示します。



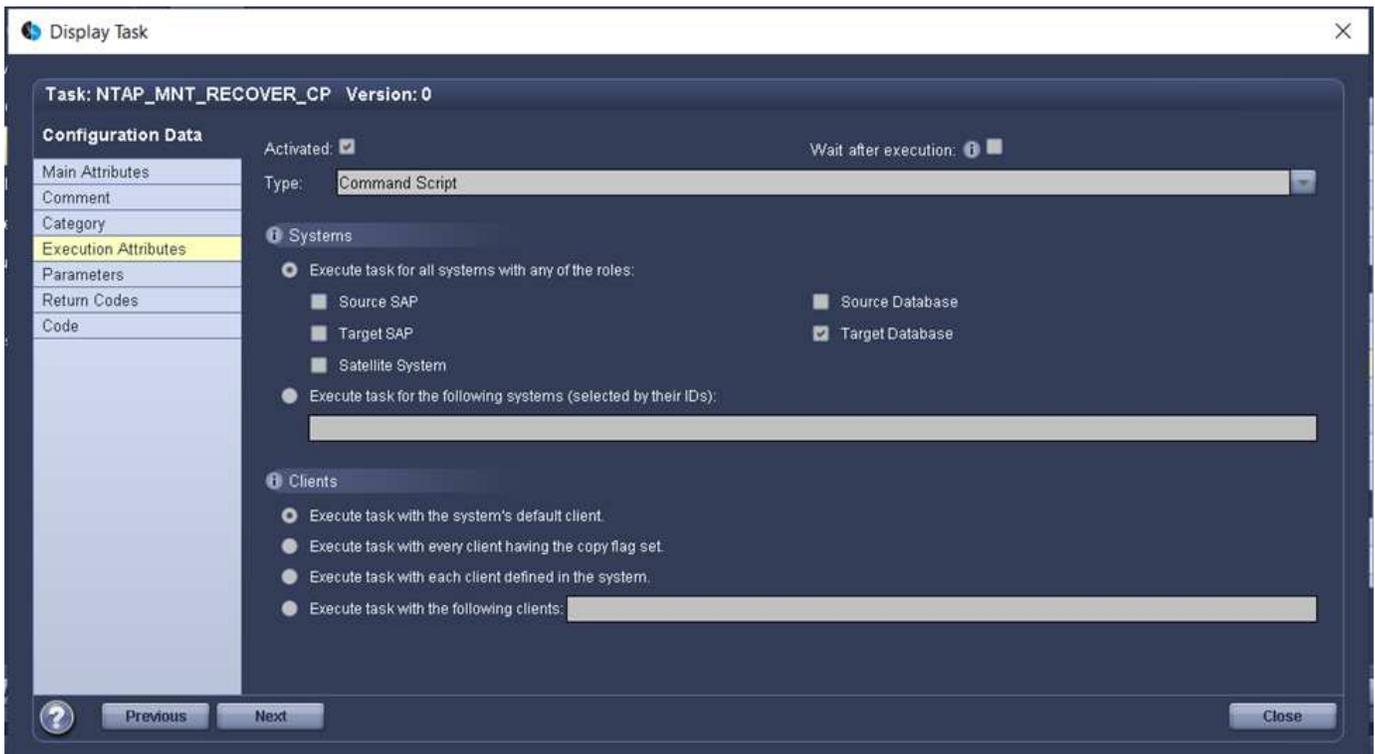
```
1 Write-Host "
2 # PowerShell Script: Backup HANA Database H05 clone to sap-inx36 as H06
3 # Version 1.0: 20200616
4 #
5 #
6 #
7 #
8 #Import SnapCenter 4.5 PowerShell Commandlets
9 Import-Module C:\Libelle\PowerShell\Modules\SnapCenter
10
11 #Setting User Credentials
12 Write-Host "Authenticate to SnapCenter Server" -foregroundcolor DarkBlue -backgroundcolor White
13
14 #generate Authentication Password File:
15 if (-not (Test-Path "c:\temp\myapp_password.txt")) {
16     $credential = Get-Credential
17     $credential.Password | ConvertFrom-SecureString | Set-Content "c:\temp\myapp_password.txt"
18 }
19 $User = "muccbc\sapdemo"
20 $cred = New-Object -TypeName System.Management.Automation.PSCredential -ArgumentList $user, (Get-Content "c:\temp\myapp_password.txt"
21 Open-SmConnection -Credential $cred -SMSbaseurl https://sap-sc-demo-dev.muccbc.hq.netapp.com:8146/
22
23 #Backup Create:
24 Write-Host "Starting Workflow Step 1) Backup Create" -foregroundcolor DarkBlue -backgroundcolor White
25
26 $Backup = New-SmBackup -Policy Manual -ResourceGroupName sap-sc-demo-dev_muccbc_hq_netapp_com_hana_MDC_H05 -ScheduleName Hourly-Confir
27 Get-SmJobSummaryReport -JobId $Backup.Id
28 do { $Job=Get-SmJobSummaryReport -JobId $Backup.Id; write-host $Job.Status; sleep 30 } while ( $Job.Status -Match "Running" )
29 Get-SmJobSummaryReport -JobId $Backup.Id
30 if ( $Job.Status -eq "Completed" ) { Write-Host "Finished Workflow Step 1) Backup has been created" } else { Write-Host "LSC:ERROR:BAK
```

前述したように、クローンボリュームの名前を次のタスク「NTAP_Mnt_RECOVER_CP」に渡して、ターゲットサーバでクローンボリュームをマウントする必要があります。クローン・ボリュームの名前（ジャンクション・パスとも呼ばれます）は変数「\$JunctionalPath」に格納されます。後続のLSCタスクへの引き渡しは、カスタムのLSC変数によって行われます。

```
echo $JunctionalPath > $_task(current, custompath1)_$
```

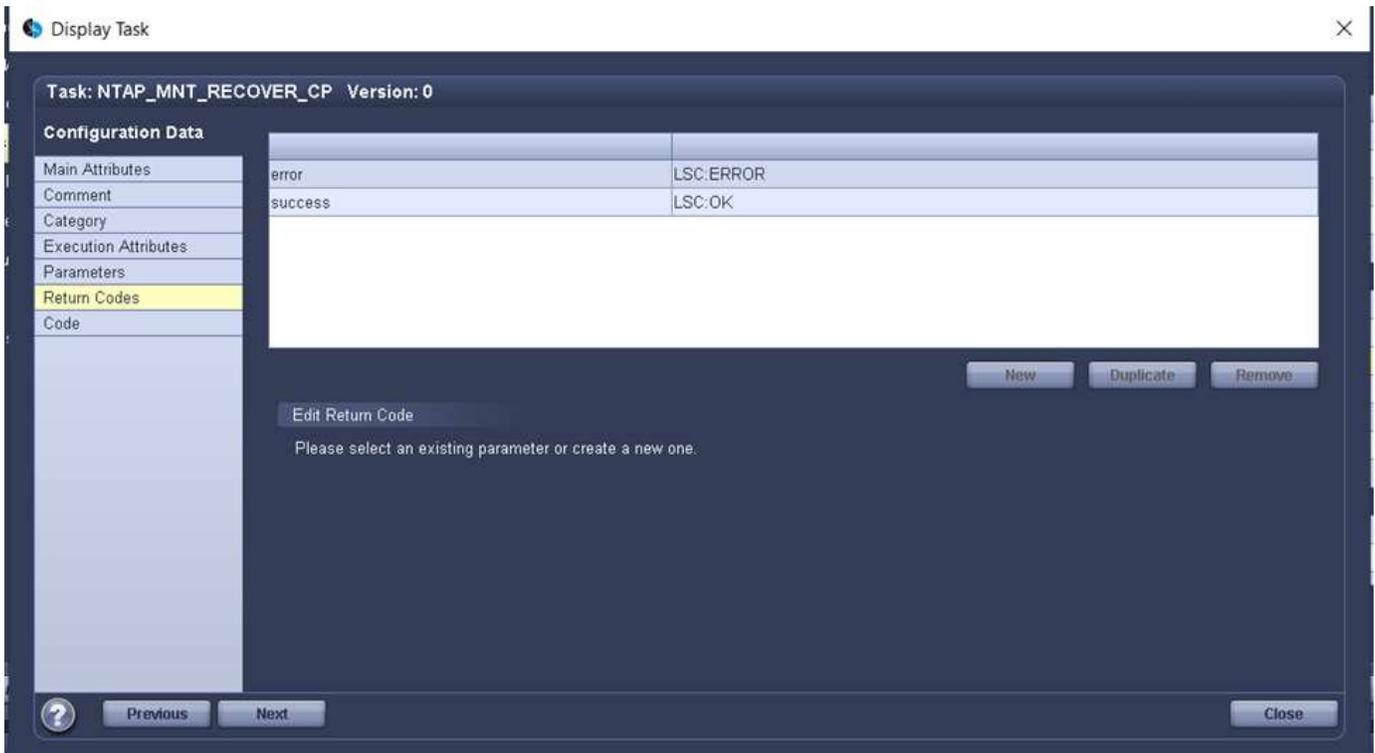
スクリプトはLSCマスター（サテライトシステムでもある）上で実行されるため、SnapCenter サーバ上のLSCマスターは、SnapCenter でバックアップおよびクローニング操作を実行するための適切な権限を持つWindowsユーザとして実行する必要があります。適切な権限があるかどうかを確認するには、ユーザがSnapCenter GUIでSnapshotコピーとクローンを実行できる必要があります。

次の図は「NTAP_Mnt_RECOVER_CP」タスクの構成を示していますLinuxシェルスクリプトを実行するため、これはターゲットデータベースシステムで実行されるコマンドスクリプトです。



LSCは、クローンボリュームのマウントを認識し、ターゲットデータベースのリストアとリカバリが成功したかどうかを確認する必要があるため、少なくとも2つの戻りコードタイプを定義する必要があります。1つはスクリプトを正常に実行するためのコードで、1つはスクリプトの実行に失敗したコードです。次の図に示します。

- 実行が成功した場合は、スクリプトから標準出力に「LSC:OK」を書き込む必要があります。
- 実行に失敗した場合は、スクリプトから標準出力に「LSC:error」を書き込む必要があります。



次の図に、Linux Shellスクリプトの一部を示します。このスクリプトでは、ターゲットデータベースの停止、古いボリュームのアンマウント、クローンボリュームのマウント、ターゲットデータベースのリストアとリカバリを行います。前のタスクでは、ジャンクションパスがLSC変数に書き込まれました。次のコマンドはこのLSC変数を読み取り、値をLinuxシェルスクリプトの「\$JunctionPath」変数に格納します。

```
JunctionPath=$_include($_task(NTAP_SYSTEM_CLONE_CP, custompath1)_$, 1, 1)_$_
```

ターゲットシステム上のLSCワーカーは「<sidadm>」として実行されますが、マウントコマンドはrootユーザとして実行する必要があります。したがってcentral_plugin_host_wrapper_script.shを作成する必要があります。スクリプト「central_plugin_host_wrapper_script.sh」は、「sudo」コマンドを使用して「NTAP_Mnt_recovery_CP」タスクから呼び出されます。スクリプトはsudoコマンドを使用してUID 0で実行され「古いボリュームのアンマウント」クローンボリュームのマウント」ターゲット・データベースのリストアとリカバリなど以降のすべての手順を実行できますsudoを使用してスクリプト実行を有効にするには/etc/sudoersに次の行を追加する必要があります

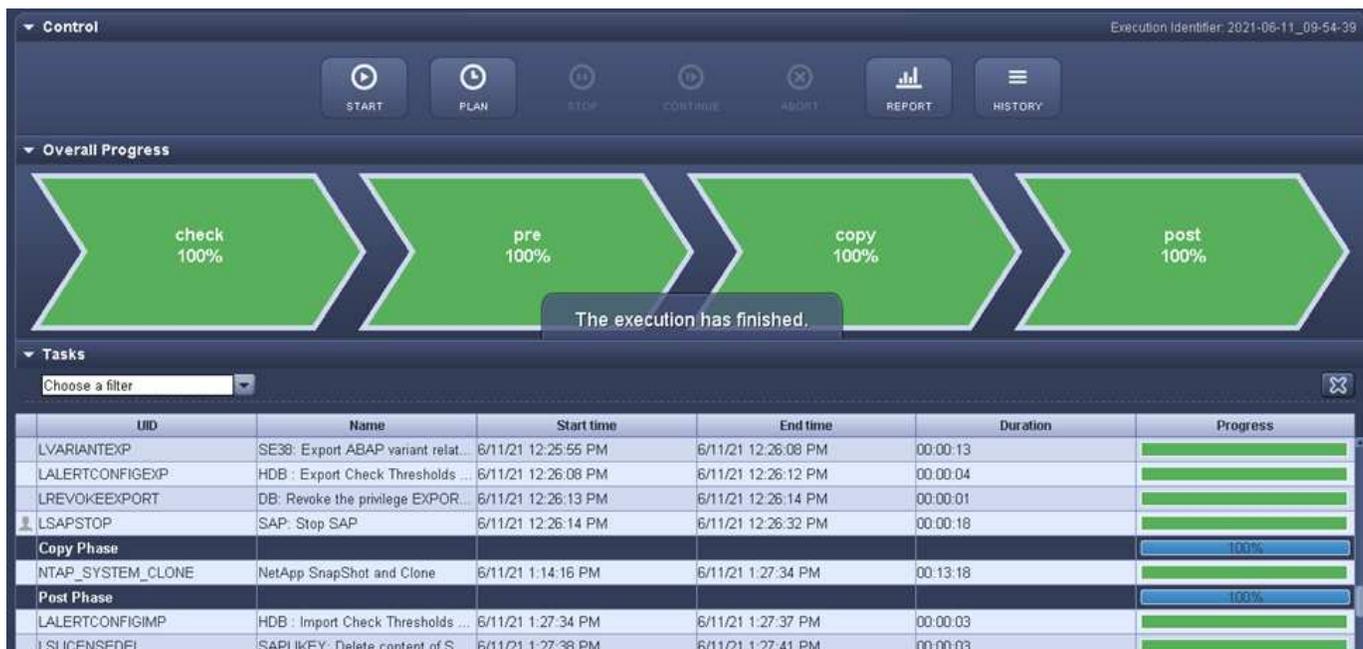
```
hn6adm ALL=(root) NOPASSWD:/usr/local/bin/H06/central_plugin_host_wrapper_script.sh
```



SAP HANAシステムの更新処理

LSCとNetApp SnapCenter の間の必要な統合タスクがすべて実行されたので、完全に自動化されたSAPシステム更新を開始するのはワンクリックタスクです。

次の図は標準インストールにおけるNTAP ``SYSTEM`_CLONE」タスクを示していますこの出力からわかるように、Snapshotコピーとクローンを作成し、クローンボリュームをターゲットデータベースサーバにマウントし、ターゲットデータベースのリストアとリカバリには約14分かかりました。SnapshotとNetApp FlexCloneテクノロジーを使用すれば、ソースデータベースのサイズに関係なく、このタスクの所要時間はほぼ同じです。



次の図に、セントラル通信ホストを使用する場合の「NTAP_SYSTEM_CLONE_CP」と「NTAP_Mnt_RECOVERY_CP」の2つのタスクを示します。この出力からわかるように、Snapshotコピー、クローン、ターゲットデータベースサーバへのクローンボリュームのマウント、ターゲットデータベースのリストアとリカバリには約12分かかりました。これは、標準インストールを使用する場合に、これらの手順を実行するのに必要な時間と同じか、それより短くなります。繰り返しになりますが、SnapshotとNetApp FlexCloneテクノロジーを使用すれば、ソースデータベースのサイズに関係なく、これらのタスクを短時間で一貫して実行できます。



LSC、AzAcSnap、およびAzure NetApp Files を使用してSAP HANAシステムが更新されます

を使用します ["Azure NetApp Files for SAP HANAの略"](#)、Oracle、DB2 on Azureを利用すると、NetApp ONTAP の高度なデータ管理機能とデータ保護機能をMicrosoft Azure

NetApp Files 標準サービスで利用できます。"AzAcSnap" は、SAPシステムの更新処理を高速化して、SAP HANAシステムとOracleシステムのNetApp Snapshotコピーをアプリケーションと整合性のあるものにするための基盤です（DB2は現在AzAcSnapではサポートされていません）。

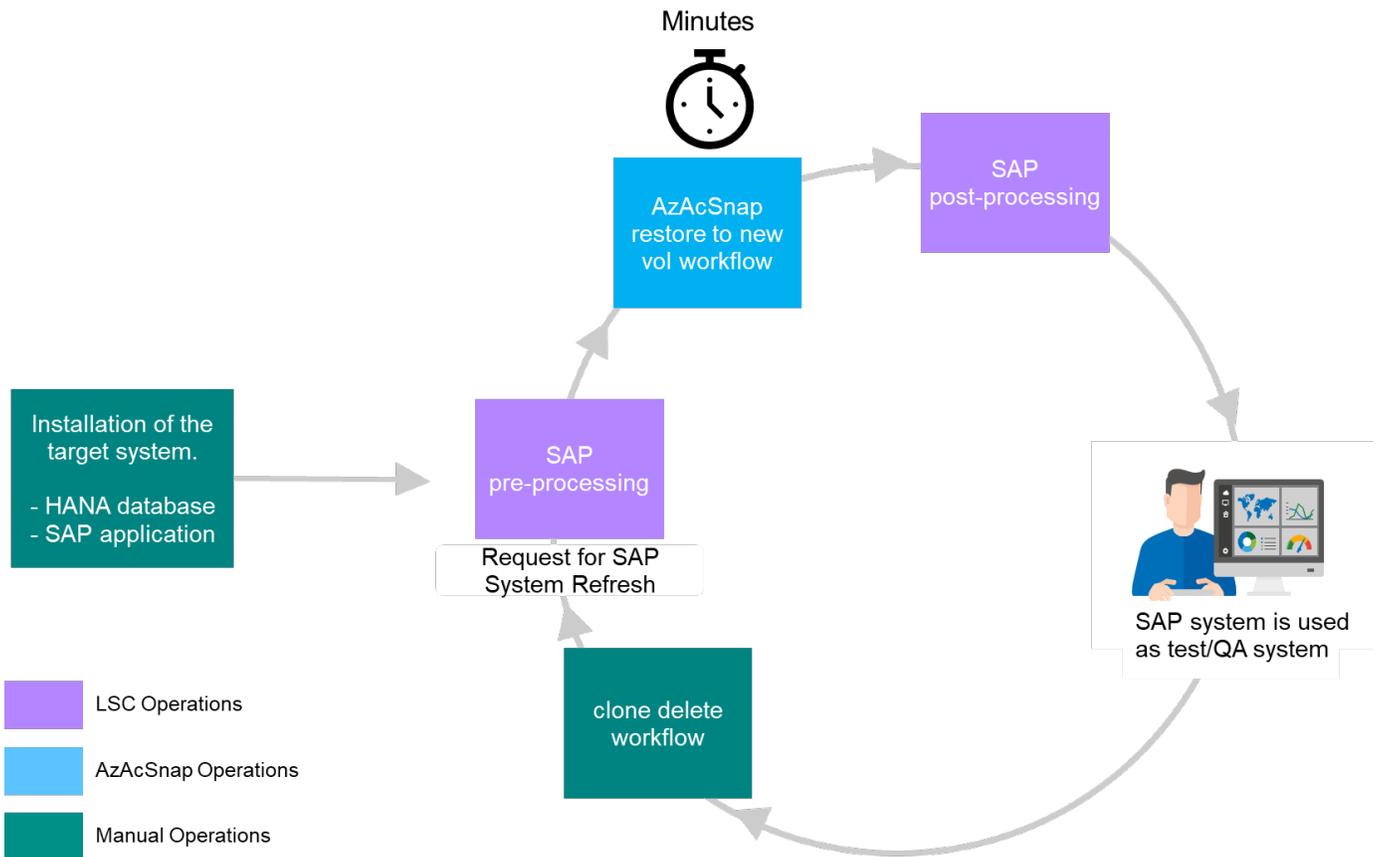
Snapshotコピーのバックアップは、オンデマンドで作成することも、バックアップ戦略の一環として定期的に作成することもでき、効率的に新しいボリュームにクローニングして、ターゲットシステムを迅速に更新することができます。AzAcSnapは、バックアップを作成して新しいボリュームにクローンを作成するために必要なワークフローを提供します。一方、Libelle SystemCopyは、完全なエンドツーエンドのシステム更新に必要な、前処理と後処理の手順を実行します。

この章では、SAP HANAを基盤データベースとして使用したAzAcSnapおよびLibelle SystemCopyを使用したSAPシステムの自動更新について説明します。AzAcSnapはOracleでも利用できるため、AzAcSnap for Oracleを使用して同じ手順を実装することもできます。その他のデータベースは、今後AzAcSnapによってサポートされる可能性があります。この場合、LSCおよびAzAcSnapを使用してこれらのデータベースのシステムコピー操作が有効になります。

次の図は、AzAcSnapおよびLSCを使用したSAPシステム更新ライフサイクルの一般的なワークフローを示しています。

- ターゲットシステムの初期インストールと準備を1回だけ行います。
- LSCによって実行されるSAP前処理操作。
- AzAcSnapで実行されるターゲットシステムへのソースシステムの既存のSnapshotコピーのリストア（またはクローニング）。
- LSCによって実行されるSAP後処理操作。

システムはテストシステムまたはQAシステムとして使用できます。新しいシステムの更新が要求されると、手順2でワークフローが再開されます。残りのクローンボリュームは手動で削除する必要があります。



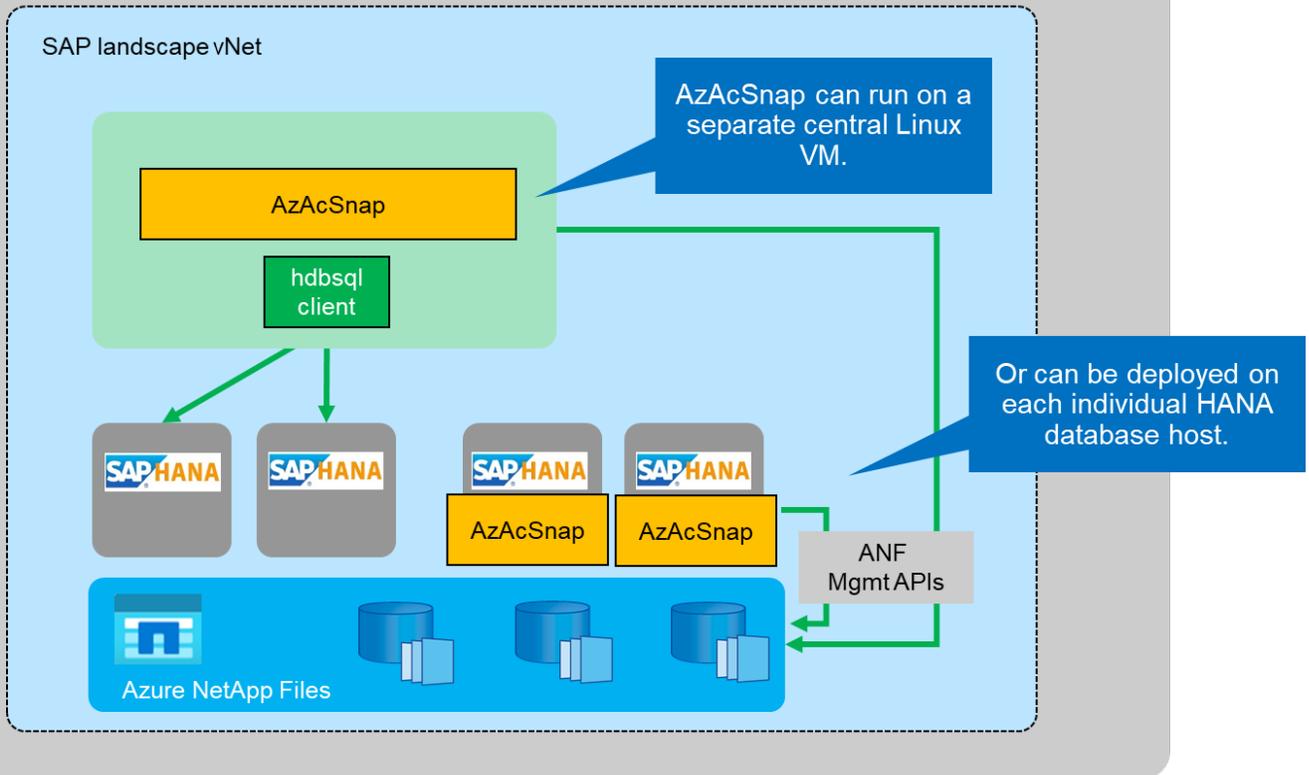
前提条件および制限事項

次の前提条件を満たしている必要があります。

AzAcSnapがインストールされ、ソースデータベース用に設定されている

一般に、AzAcSnapには次の図に示すように、2つの導入オプションがあります。

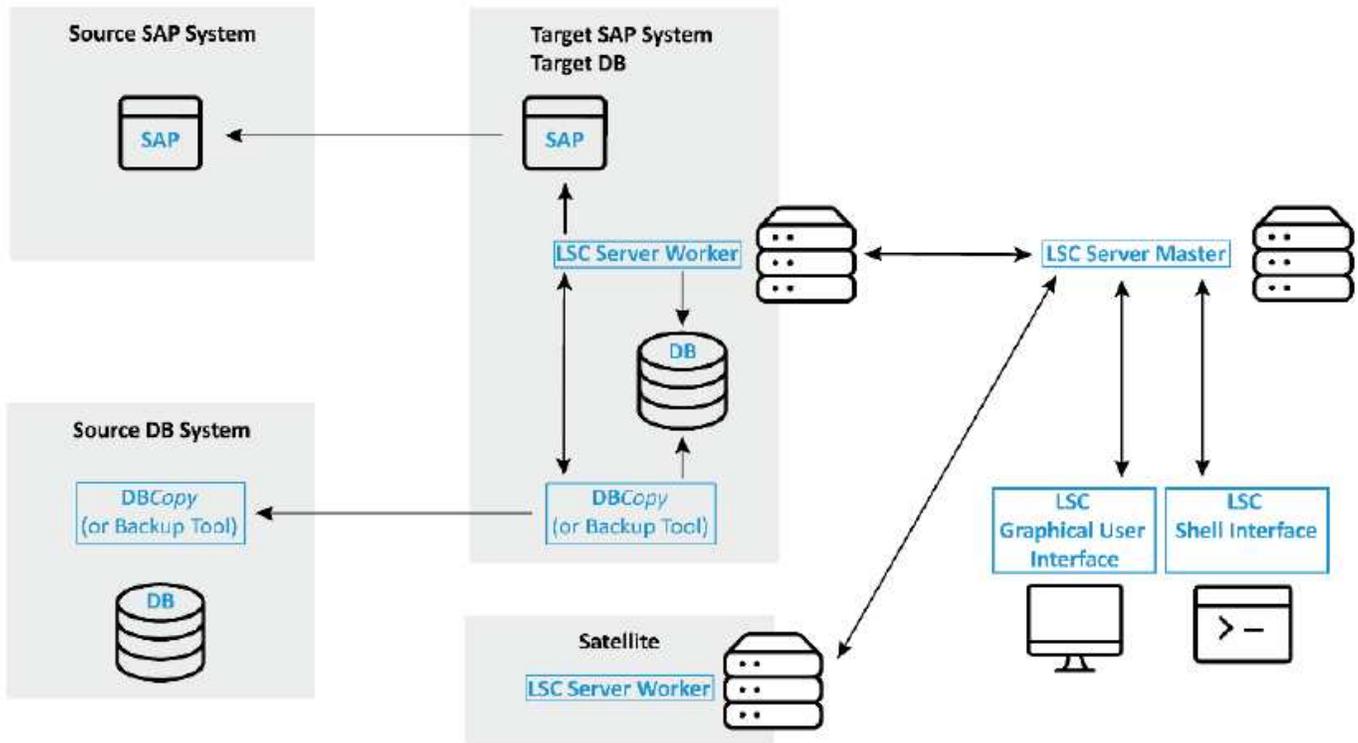
Customer Network and Azure Subscription



AzAcSnapは、すべてのDB構成ファイルが一元的に格納されている中央のLinux VMにインストールして実行できます。AzAcSnapは、すべてのデータベースに（hdbsqlクライアントを介して）すべてのデータベースと、これらすべてのデータベースに設定されたHANAユーザストアキーにアクセスできます。分散型の展開では、AzAcSnapは各データベースホストに個別にインストールされ、通常はローカルデータベースのDB構成のみが格納されます。どちらの展開オプションもLSC統合でサポートされています。ただし、このドキュメントのラボセットアップではハイブリッドアプローチを採用しました。AzAcSnapは、すべてのDB構成ファイルとともに中央のNFS共有にインストールされました。この中央インストール共有は、「/mnt/software/AZACSNAP/snapshot-tool」の下すべてのVMにマウントされました。その後、このツールはDB VM上でローカルに実行され、

Libelle SystemCopyがソースおよびターゲットのSAPシステム用にインストールおよび設定されていること

Libelle SystemCopyの展開は、次のコンポーネントで構成されています。



- *LSC Master.* という名前が示すように、これはLibelleベースのシステムコピーの自動ワークフローを制御するマスターコンポーネントです。
- *LSC Worker.* LSCワーカーは通常、ターゲットSAPシステム上で実行され、自動システムコピーに必要なスクリプトを実行します。
- *LSC Satellite.* LSCサテライトは、追加のスクリプトを実行する必要があるサードパーティシステムで実行されます。LSCマスターは、LSCサテライトシステムの役割も果たします。

Libelle SystemCopy (LSC) GUIが適切なVMにインストールされている必要があります。この実習セットアップでは、LSC GUIは別のWindows VMにインストールされていますが、LSCワーカーとともにDBホストでも実行できます。LSCワーカーは、少なくともターゲットDBのVMにインストールする必要があります。選択したAzAcSnap展開オプションによっては、LSCワーカーの追加インストールが必要な場合があります。AzAcSnapが実行されるVMにLSCワーカーインストールが必要です。

LSCをインストールした後、ソースおよびターゲットデータベースの基本設定をLSCガイドラインに従って実行する必要があります。次の図は、このドキュメントのラボ環境の構成を示しています。ソースとターゲットのSAPシステムおよびデータベースの詳細については、次のセクションを参照してください。

Libelle SystemCopy 9.0.0.0.052

Setup Monitor Administration

Libelle SystemCopy admin

Change State: 26

Configurations +

PN1toQN1

General

Systems

BusinessShadow

DataMasking

Categories

Tasks

Global parameters

Snippets

Execution

Alarm

Permissions

The overview shows the available systems and their roles in the configuration.

System Identifier	Worker	Source SAP	Source Database	Target SAP	Target Database	Satellite System
PN1		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
P01		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
QN1	vm-q1:9000	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
QL1	vm-q1:9000	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

また、SAPシステムに適した標準のタスクリストを設定する必要があります。LSCのインストールおよび設定の詳細については、LSCインストールパッケージの一部であるLSCユーザマニュアルを参照してください。

既知の制限

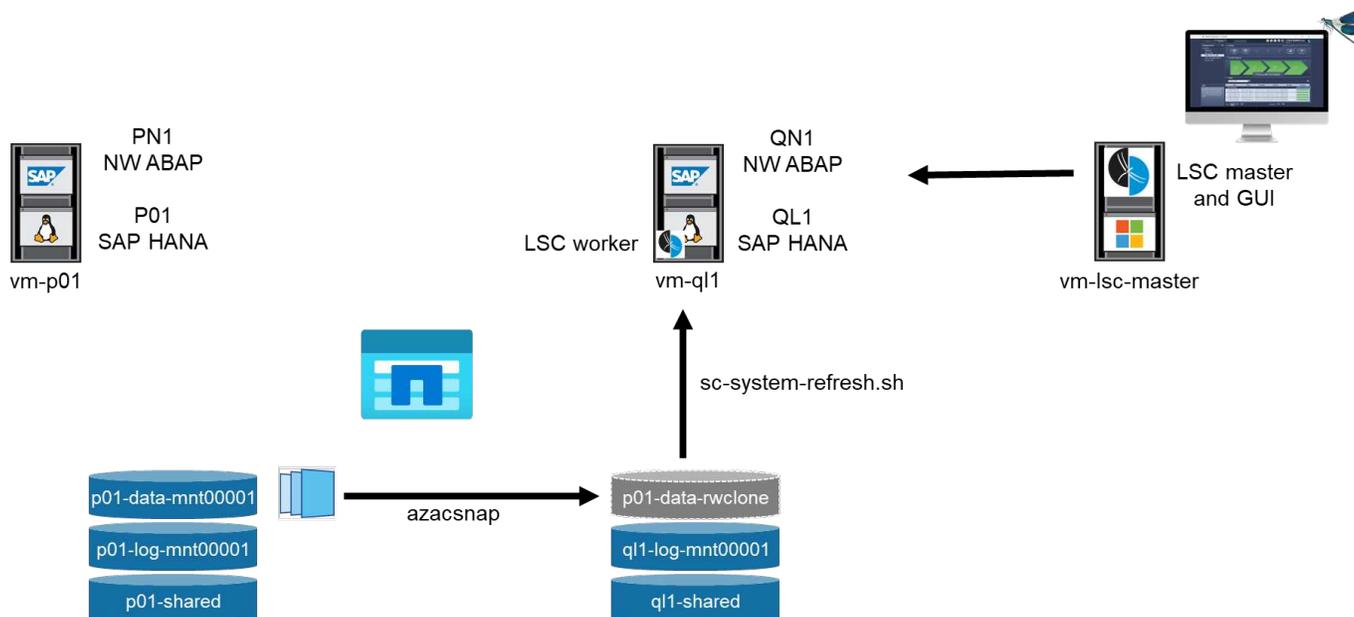
ここで説明するAzAcSnapとLSCの統合は、SAP HANAのシングルホストデータベースでのみ機能します。SAP HANAマルチホスト（またはスケールアウト）配置もサポートできますが、このような配置では、コピーフェーズおよびアンダーレイアウトスクリプトのLSCカスタムタスクをいくつか調整または拡張する必要があります。このような機能強化については、本ドキュメントでは説明していません。

SAPシステムの更新機能が統合される際には、ソースシステムのSnapshotコピーが最新で正常に作成され、ターゲットシステムの更新が実行されます。他の古いSnapshotコピーを使用する場合は、の対応するロジックを指定します **ZAZACSNAPRESORE** カスタムタスクを調整する必要があります。このプロセスについては、本ドキュメントでは説明しません。

ラボのセットアップ

このラボ環境は、ソースのSAPシステムとターゲットのSAPシステムで構成され、どちらもSAP HANAのシングルホストデータベースで実行されます。

次の図は、ラボのセットアップを示しています。



このボリュームには、次のシステム、ソフトウェアバージョン、およびAzure NetApp Files ボリュームが含まれています。

- * P01.* SAP HANA 2.0 SP5データベース。ソースデータベース、シングルホスト、シングルテナント
- * PN1.* SAP NetWeaver ABAP 7.51ソースのSAPシステム：
- * VM-P01.* SLES 15 SP2、AzAcSnapがインストールされている場合。ソースVMでP01とPN1をホストしています。
- * QL1.* SAP HANA 2.0 SP5データベース。ターゲットデータベース、シングルホスト、シングルテナントのシステム更新

- * QN1.* SAP NetWeaver ABAP 7.51システム更新の対象となるSAPシステム：
- * VM-QL1.* LSCワーカーがインストールされたSLES 15 SP2。ターゲットのVMでQL1とQN1をホストしています。
- LSCマスターバージョン9.0.0.052。
- * VM-LSC-MMASTER.* Windows Server 2016。LSCマスターおよびLSC GUIをホストします。
- 専用DBホストにマウントされたP01とQL1のデータ、ログ、共有のAzure NetApp Files ボリューム。
- スクリプト、AzAcSnapのインストール、すべてのVMにマウントされた構成ファイル用のCentral Azure NetApp Files ボリューム。

最初の1回限りの準備手順

最初のSAPシステムの更新を実行する前に、AzAcSnapで実行されるAzure NetApp Files のSnapshotコピーおよびクローニングベースのストレージ処理を統合する必要があります。また、データベースの起動と停止、およびAzure NetApp Files ボリュームのマウントまたはアンマウントを実行する補助スクリプトも実行する必要があります。必要なすべてのタスクは、コピーフェーズの一部としてLSCでカスタムタスクとして実行されません。次の図は、LSCタスクリスト内のカスタムタスクを示しています。

	Phase	UID	Name	Type
pre 76		LALERTCONFIGEXP	HDB : Export Check Threshold...	lsh
pre 77		LREVOKEEXPORT	DB: Revoke the privilege EXPO...	cmd
pre 78		LJAVACONFEXP	JAVA: Backup java config files...	cmd
pre 79		LSTOPSLTJOBS	LTRC: Stop all replication jobs ...	lsh
pre 80		LSAPSTOP	SAP: Stop SAP	intv
pre 81		LSTOPSAPSYSTEM	Stops all SAP instances (appli...	lsh
copy	Copy Phase			phase
copy 1		ZSCCOPYSHUTDOWN	Shutdown HANA DB	cmd
copy 2		ZSCCOPYUMOUNT	Unmount data volumes	cmd
copy 3		ZAZACSNAPRESTORE	Restore snapshot backup of so...	cmd
copy 4		ZSCCOPYMOUNT	Mount data volumes	cmd
copy 5		ZSCCOPYRECOVER	Recover target DB based on sn...	cmd
post	Post Phase			phase
post 1		LCHNGHDBPWD	HDB : Restore the password fo...	cmd
post 2		LHDBLICIMP	HANA DB License Import	lsh
post 3		LALERTCONFIGIMP	HDB : Import Check Threshold...	lsh

5つのコピー・タスクの詳細については以下を参照してくださいこれらのタスクの一部では、サンプルスクリプト「sc-system-refresh.sh」を使用して、必要なSAP HANAデータベースのリカバリ処理と、データボリュームのマウントおよびアンマウントをさらに自動化します。スクリプトは、LSCに対する実行が成功したことを示すために、システム出力で「LSC:SUCCESS」メッセージを使用します。カスタムタスクおよび使用可能なパラメータの詳細については、LSCユーザマニュアルおよびLSC開発者ガイドを参照してください。このラボ環境のすべてのタスクは、ターゲットDB VMで実行されます。



サンプルスクリプトは現状のまま提供されており、ネットアップではサポートしていません。スクリプトは、mailto : ng-sapcc@netapp.com [ng-sapcc@netapp.com ^]にEメールで送信できます。

Sc-system-refresh.sh構成ファイル

前述したように、補助スクリプトを使用して、データベースの起動と停止、Azure NetApp Files ボリュームのマウントとアンマウント、およびSnapshotコピーからのSAP HANAデータベースのリカバリを行います。スクリプト「sc-system-refresh.sh」は中央NFS共有に格納されます。スクリプトでは、ターゲットデータベースごとに構成ファイルが必要です。このファイルは、スクリプト自体と同じフォルダに格納する必要があります。

す。コンフィギュレーションファイルには、「sc-system-refresh-<target DB SID>.cfg」という名前（この実習環境では「sc-system-refresh-ql1.cfg」など）を付ける必要があります。ここで使用する構成ファイルでは、固定/ハードコーディングされたソースDB SIDを使用します。いくつかの変更により、スクリプトと構成ファイルを拡張して、ソースDB SIDを入力パラメータとして取得できます。

特定の環境に応じて、次のパラメータを調整する必要があります。

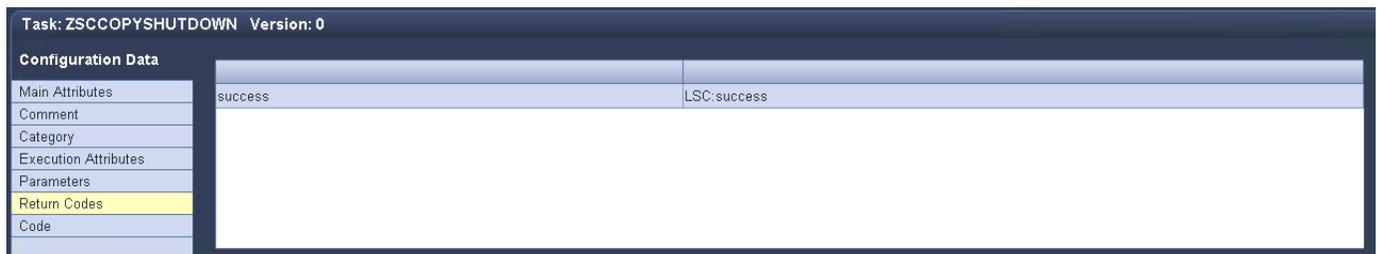
```
# hdbuserstore key, which should be used to connect to the target database
KEY="QL1SYSTEM"
# single container or MDC
export P01_HANA_DATABASE_TYPE=MULTIPLE_CONTAINERS
# source tenant names { TENANT_SID [, TENANT_SID]* }
export P01_TENANT_DATABASE_NAMES=P01
# cloned vol mount path
export CLONED_VOLUMES_MOUNT_PATH=`tail -2
/mnt/software/AZACSNAP/snapshot_tool/logs/azacsnap-restore-azacsnap-
P01.log | grep -oe "[0-9]*\.[0-9]*\.[0-9]*\.[0-9]*:/*.* "`
```

ZSCCOPYSHUTDOWN

このタスクは、ターゲットのSAP HANAデータベースを停止します。このタスクの[コード]セクションには、次のテキストが含まれています。

```
$_include_tool(unix_header.sh)_$
sudo /mnt/software/scripts/sc-system-refresh/sc-system-refresh.sh shutdown
$_system(target_db, id)_$ > $_logfile_$(
```

スクリプト「sc-system-refresh.sh」は'shutdown'コマンドとDB SIDの2つのパラメータを取り'sapcontrol'を使用してSAP HANAデータベースを停止します。システム出力は標準のLSCログファイルにリダイレクトされます。前述のように、「isc: success」メッセージは、正常に実行されたことを示します。



Task: ZSCCOPYSHUTDOWN Version: 0	
Configuration Data	
Main Attributes	success LSC:success
Comment	
Category	
Execution Attributes	
Parameters	
Return Codes	
Code	

ZSCCOPYUMOUNT

このタスクでは、ターゲットのDBオペレーティングシステム（OS）から古いAzure NetApp Files データボリュームをアンマウントします。このタスクのコードセクションには、次のテキストが含まれています。

```

$_include_tool(unix_header.sh)_$
sudo /mnt/software/scripts/sc-system-refresh/sc-system-refresh.sh umount
$_system(target_db, id)_$ > $_logfile_$

```

前のタスクと同じスクリプトが使用されます。渡される2つのパラメータは'umount'コマンドとDB SIDです

ZAZACSNAPRESTORE

このタスクでは、AzAcSnapを実行して、ソースデータベースの最新の成功したSnapshotコピーを、ターゲットデータベースの新しいボリュームにクローニングします。この処理は、従来のバックアップ環境でのバックアップのリダイレクトリストアに相当します。ただし、Snapshotコピーとクローニング機能を使用すれば、最大のデータベースであっても数秒でこのタスクを実行できます。従来のバックアップでは、このタスクに数時間かかることもありましたが、このタスクのコードセクションには、次のテキストが含まれています。

```

$_include_tool(unix_header.sh)_$
sudo /mnt/software/AZACSNAP/snapshot_tool/azacsnap -c restore --restore
snaptovol --hanasid $_system(source_db, id)_$
--configfile=/mnt/software/AZACSNAP/snapshot_tool/azacsnap
-$_system(source_db, id)_$.json > $_logfile_$

```

AzAcSnapの'restore'コマンド・ライン・オプションに関する完全なドキュメントはAzureのドキュメントを参照してください "[Azure Application Consistent Snapshotツールを使用してリストア](#)"。この呼び出しでは、ソースDBのJSON DB構成ファイルが、「azacsnap -<source DB SID>」という命名規則に従って中央のNFS共有にあることが前提となります。JSON形式（このラボ環境では'azacsnap-p0P01 JSON'など）



AzAcSnapコマンドの出力は変更できないため、このタスクにはデフォルトの「LSC:SUCCESS」メッセージを使用できません。そのためAzAcSnap出力の文字列'Example mount instructions'が成功した戻りコードとして使用されます5.0 GAバージョンのAzAcSnapでは、この出力はクローニングプロセスが成功した場合にのみ生成されます。

次の図に、新しいボリュームへのAzAcSnapリストア成功メッセージを示します。

Task: ZAZACSNAPRESTORE Version: 0	
Configuration Data	
Main Attributes	success Example mount instructions
Comment	
Category	
Execution Attributes	
Parameters	
Return Codes	
Code	

ZSCCOPYMOUNT

このタスクでは、ターゲットDBのOSに新しいAzure NetApp Files データボリュームをマウントします。このタスクのコードセクションには、次のテキストが含まれています。

```

$_include_tool(unix_header.sh)_$
sudo /mnt/software/scripts/sc-system-refresh/sc-system-refresh.sh mount
$_system(target_db, id)_$ > $_logfile_$

```

sc-system-refresh.shスクリプトが再び使用され'mount'コマンドとターゲットDB SIDが渡されます

ZSCOPYRECOVER

このタスクでは、リストア（クローン）されたSnapshotコピーに基づいて、システムデータベースとテナントデータベースのSAP HANAデータベースのリカバリを実行します。ここで使用するリカバリ・オプションは、フォワード・リカバリに適用される特定のデータベース・バックアップ（追加ログなしなど）を対象としています。したがって、リカバリ時間は非常に短くなります（最大で数分）。この処理の実行時間は、リカバリプロセス後に自動的に実行されるSAP HANAデータベースの起動によって決まります。起動時間を短縮するために、必要に応じて、次のAzureのドキュメントに従ってAzure NetApp Files データボリュームのスループットを一時的に向上させることができます。"ボリュームクォータの動的な増減"。このタスクのコードセクションには、次のテキストが含まれています。

```

$_include_tool(unix_header.sh)_$
sudo /mnt/software/scripts/sc-system-refresh/sc-system-refresh.sh recover
$_system(target_db, id)_$ > $_logfile_$

```

このスクリプトは'recover'コマンドとターゲットDB SIDとともに再び使用されます

SAP HANAシステムの更新処理

このセクションでは、ラボシステムの更新処理のサンプルとして、このワークフローの主な手順を記載します。

バックアップカタログに記載されたP01ソースデータベースの定期的なSnapshotコピーとオンデマンドSnapshotコピーが作成されている。

The screenshot shows the SAP HANA Backup Catalog interface for database P01. The interface is divided into two main sections: Backup Catalog and Backup Details.

Backup Catalog: A table listing backup operations. The columns are Stat., Started, Duration, Size, Backup Ty..., and Destinati... The table shows a series of hourly snapshots from February 25, 2021, to March 12, 2021, all with a size of 9.75 GB and a duration of approximately 00h 01m 03s to 00h 02m 13s.

Stat.	Started	Duration	Size	Backup Ty...	Destinati...
■	Mar 12, 2021 10:40:54 AM	00h 01m 03s	9.75 GB	Data Back...	Snapshot
■	Mar 12, 2021 8:00:01 AM	00h 01m 04s	9.75 GB	Data Back...	Snapshot
■	Mar 12, 2021 4:00:01 AM	00h 01m 04s	9.75 GB	Data Back...	Snapshot
■	Mar 12, 2021 12:00:02 AM	00h 02m 13s	9.75 GB	Data Back...	Snapshot
■	Mar 11, 2021 8:00:02 PM	00h 01m 05s	9.72 GB	Data Back...	Snapshot
■	Mar 11, 2021 4:00:02 PM	00h 01m 08s	9.72 GB	Data Back...	Snapshot
■	Mar 11, 2021 2:27:21 PM	00h 01m 03s	9.72 GB	Data Back...	Snapshot
■	Mar 11, 2021 12:00:03 PM	00h 01m 10s	9.72 GB	Data Back...	Snapshot
■	Mar 11, 2021 10:38:23 AM	00h 01m 04s	9.72 GB	Data Back...	Snapshot
■	Mar 2, 2021 12:00:04 PM	00h 01m 33s	9.72 GB	Data Back...	Snapshot
■	Mar 2, 2021 9:27:03 AM	00h 04m 13s	9.72 GB	Data Back...	Snapshot
■	Feb 25, 2021 12:00:02 PM	00h 01m 03s	9.72 GB	Data Back...	Snapshot

Backup Details: This section provides information about the selected backup. Key details include:

- ID: 1615545654786
- Status: Successful
- Backup Type: Data Backup
- Destination Type: Snapshot
- Started: Mar 12, 2021 10:40:54 AM (UTC)
- Finished: Mar 12, 2021 10:41:58 AM (UTC)
- Duration: 00h 01m 03s
- Size: 9.75 GB
- Throughput: n.a.
- System ID: Snapshot prefix: hourly; Tools version: 5.0 Preview (20201214.65524)
- Additional Information: <ok>
- Location: /hana/data/P01/mnt00001/

At the bottom, there is a table showing the backup's relationship to the source database:

t	Service	Size	Name	S	EBID
p01	indexserver	9.56 GB	hdb00003.0...	v	hourly_2021-03-12T104054-4046416Z
p01	xsengine	192.11 ...	hdb00002.0...	v	hourly_2021-03-12T104054-4046416Z

更新処理には、3月12日の最新バックアップが使用されています。バックアップの詳細セクションに、このバックアップの外部バックアップID (EBID) が表示されます。次の図に示すように、Azure NetApp Files データボリューム上の、対応するSnapshotコピーバックアップのSnapshotコピー名を指定します。

EastUS > p01-data-mnt00001 (mcScott-EastUS/mcScott-Premium/p01-data-mnt00001)

(mcScott-EastUS/mcScott-Premium/p01-data-mnt00001) | ...

+ Add snapshot Refresh

Search snapshots

Name	Location	Created
hourly_2021-02-25T120001-8350005Z	East US	02/25/2021, 11:59:37 AM
offline-20210226	East US	02/26/2021, 01:09:40 PM
hourly_2021-03-02T092702-8909509Z	East US	03/02/2021, 09:27:20 AM
hourly_2021-03-02T120003-4067821Z	East US	03/02/2021, 11:59:38 AM
hourly_2021-03-11T103823-2185089Z	East US	03/11/2021, 10:37:55 AM
hourly_2021-03-11T120003-0695010Z	East US	03/11/2021, 11:59:23 AM
hourly_2021-03-11T142720-7544262Z	East US	03/11/2021, 02:26:35 PM
hourly_2021-03-11T160002-4458098Z	East US	03/11/2021, 03:59:17 PM
hourly_2021-03-11T200001-9577603Z	East US	03/11/2021, 07:59:17 PM
hourly_2021-03-12T000001-7550954Z	East US	03/11/2021, 11:59:51 PM
hourly_2021-03-12T040001-5101399Z	East US	03/12/2021, 03:59:16 AM
hourly_2021-03-12T080001-5742724Z	East US	03/12/2021, 07:59:34 AM
hourly_2021-03-12T104054-4046416Z	East US	03/12/2021, 10:40:26 AM

1615545654786

Successful

Data Backup

Snapshot

Mar 12, 2021 10:40:54 AM (UTC)

Mar 12, 2021 10:41:58 AM (UTC)

00h 01m 03s

9.75 GB

n.a.

Snapshot prefix: hourly

Tools version: 5.0 Preview (20201214.65524)

Location: <ok>

/hana/data/P01/mnt00001/

Size	Name	EBID
9.56 GB	hdb00003.0...	hourly_2021-03-12T104054-4046416Z
192.11 ...	hdb00002.0...	hourly_2021-03-12T104054-4046416Z

更新操作を開始するには、LSC GUIで正しい設定を選択し、[実行の開始]をクリックします。

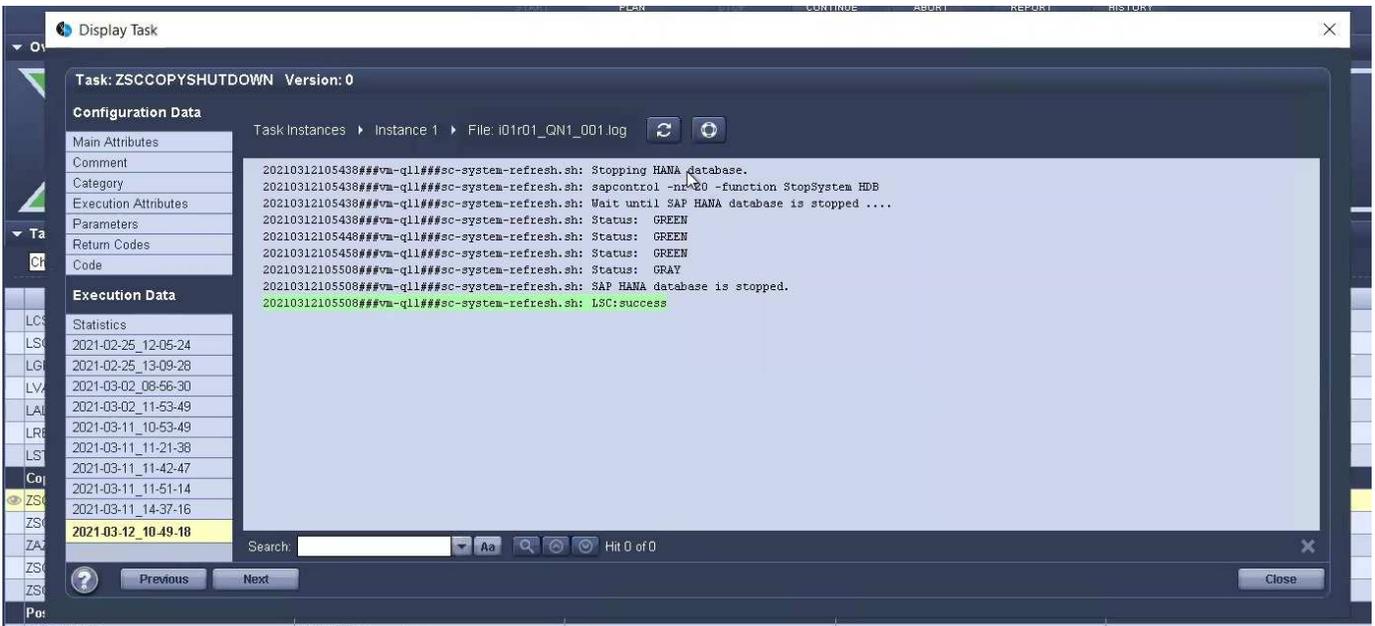
The screenshot shows the Libelle SystemCopy GUI with a progress bar indicating 100% completion for 'check', 'pre', 'copy', and 'post' phases. A 'Start Execution' dialog box is open, showing 'Execution mode' set to 'Accomplishment' and 'Execute Start Checks' checked. The dialog also includes a table with columns for 'End time', 'Duration', and 'Progress'.

End time	Duration	Progress
PM	00:00:04	100%
PM	00:00:03	100%
PM	00:00:03	100%
PM	00:00:04	100%
PM	00:00:03	100%
PM	00:00:02	100%
PM	00:00:02	100%
PM	00:00:01	100%
PM	00:00:05	100%
PM	00:00:03	100%
PM	00:00:03	100%
PM	00:00:01	100%

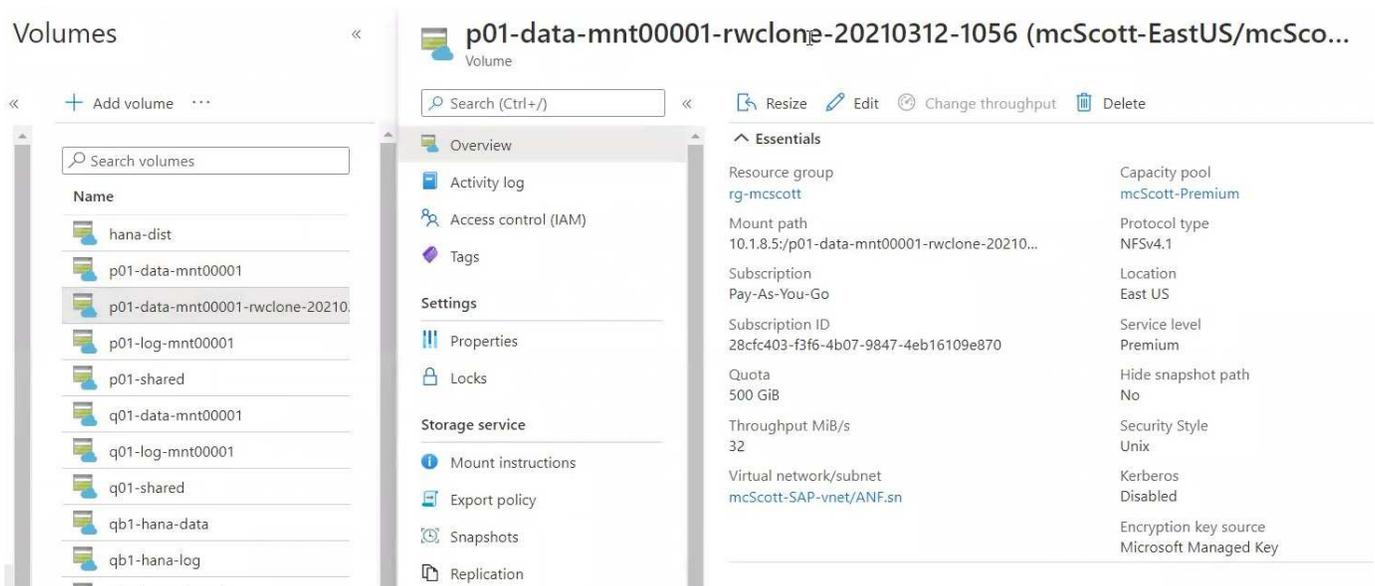
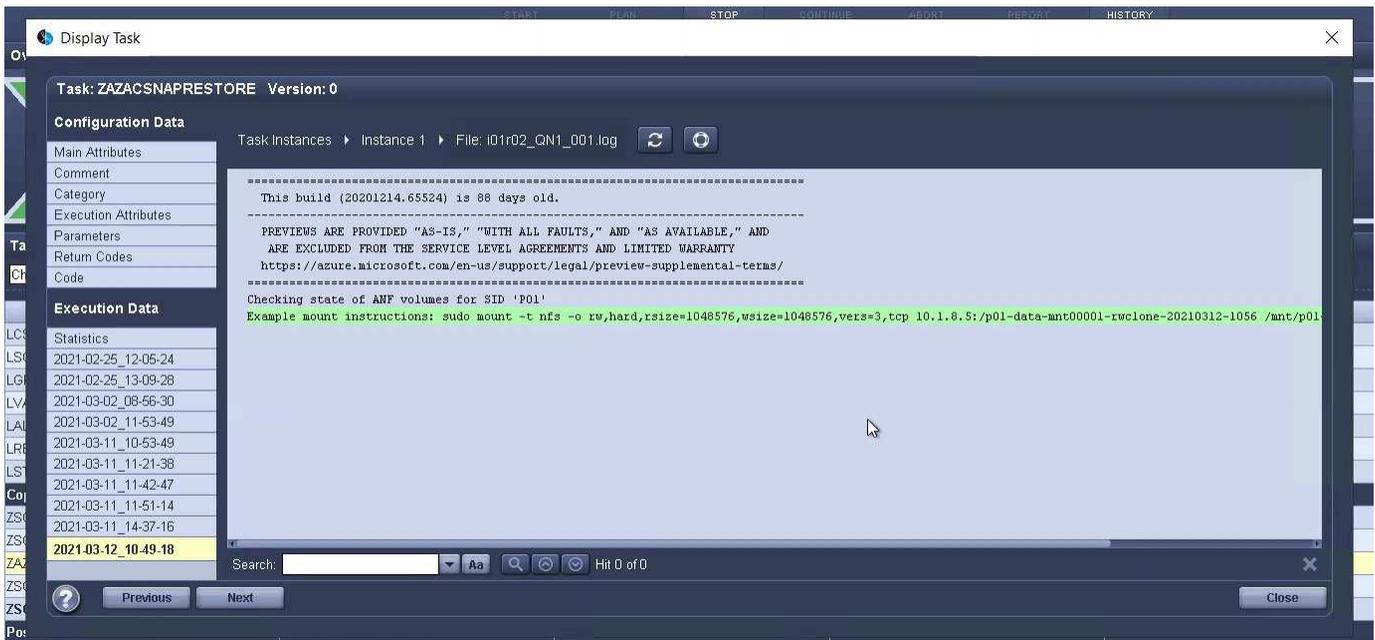
LSCは、チェックフェーズのタスクの実行を開始し、プリフェーズの設定済みタスクを実行します。



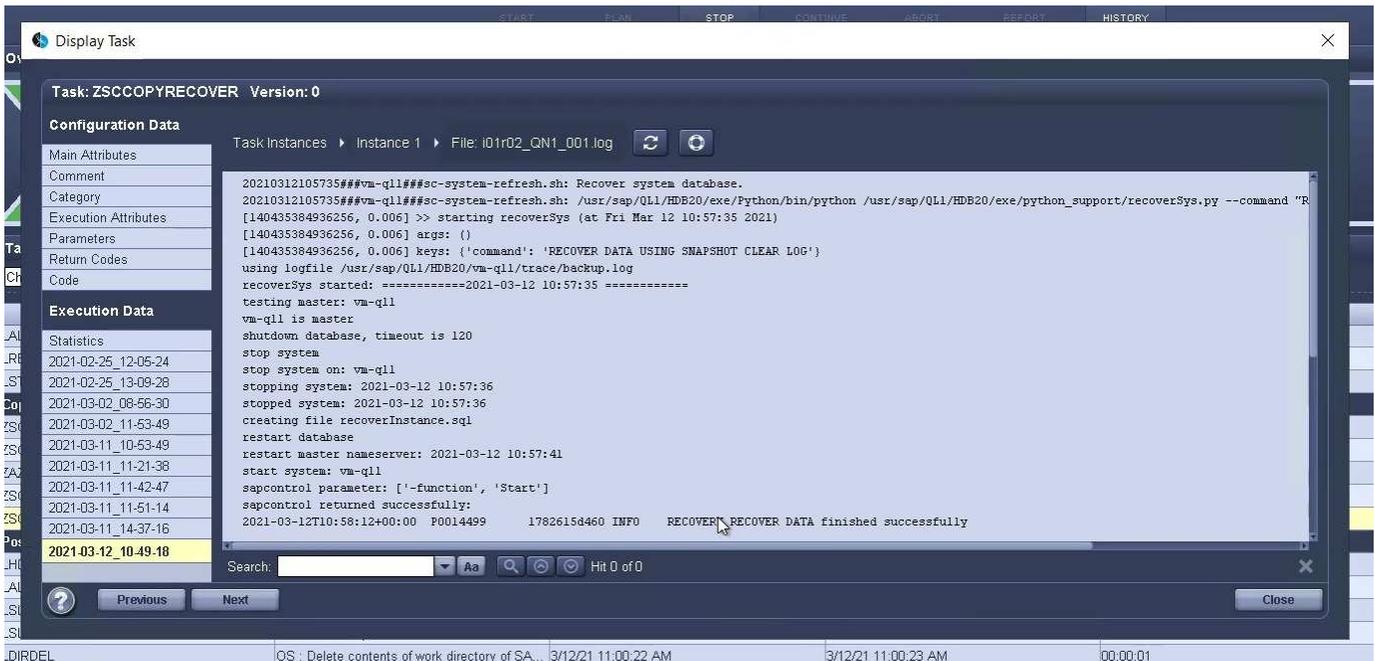
移行前フェーズの最後のステップとして、移行先のSAPシステムが停止します。次のコピーフェーズでは、前のセクションで説明したステップが実行されます。まず、ターゲットのSAP HANAデータベースが停止し、古いAzure NetApp Files ボリュームがOSからアンマウントされます。



次に、ZAZACSNAPRESTOREタスクで、P01システムの既存のSnapshotコピーからクローンとして新しいボリュームを作成します。次の2つの図は、LSC GUIでのタスクのログ、およびAzureポータルでのクローンAzure NetApp Files ボリュームを示しています。



その後、この新しいボリュームがターゲットDBホストとシステムデータベースにマウントされ、テナントデータベースが、包含するSnapshotコピーを使用してリカバリされます。リカバリが完了すると、SAP HANAデータベースが自動的に起動します。このSAP HANAデータベースの起動は、コピーフェーズのほとんどの時間を占めています。残りの手順は、データベースのサイズに関係なく、通常数秒で終了します。次の図は、SAPが提供するPythonリカバリスクリプトを使用してシステムデータベースをリカバリする方法を示しています。



コピーフェーズ後、LSCはPostフェーズで定義されたすべてのステップで続きます。システムの更新プロセスが完了すると、ターゲット・システムは再び稼働し、完全に使用可能になります。このラボシステムでは、SAPシステムの更新に必要な合計実行時間は約25分でした。このうち、コピーフェーズで消費される時間は5分未満です。



追加情報およびバージョン履歴の参照先

このドキュメントに記載されている情報の詳細については、以下のドキュメントや Web サイトを参照してください。

- ネットアップの製品マニュアル

"<https://docs.netapp.com>"

バージョン履歴

バージョン	日付	ドキュメントバージョン履歴
バージョン 1.0 以降	2022年4月	初版リリース

著作権に関する情報

Copyright © 2026 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。