



ホットスポット修復用のFlexCache ONTAP 9

NetApp
February 12, 2026

目次

ホットスポット修復用のFlexCache	1
ONTAP FlexCacheボリュームを使用したハイパフォーマンス コンピューティング ワークロードにおけるホットスポットの修復	1
主要な概念	1
ONTAP FlexCache ホットスポット修復ソリューションの設計	2
ボトルネックを理解する	2
自動プロビジョニングされたFlexCacheが解決策ではない理由	3
FlexCacheの構造	4
高密度FlexCacheの構造	5
ONTAP FlexCache密度を決定する	5
2x2x2 HDFA 構成	6
4x1x4 HDFA構成	7
ONTAP SVM 間または SVM 内の HDFA オプションを決定する	8
SVM間HDFA導入	8
SVM 内 HDFA 展開	8
HDFAとONTAPデータLIFを設定する	9
2x2x2インターSVM HDFA構成を作成する	10
4x1x4のSVM内HDFAを作成する	11
ONTAP NAS接続を分散するようにクライアントを構成する	12
Linuxクライアントの設定	12
Windowsクライアントの設定	14

ホットスポット修復用のFlexCache

ONTAP FlexCacheボリュームを使用したハイパフォーマンスコンピューティングワークロードにおけるホットスポットの修復

アニメーションレンダリングやEDAなど、多くのハイパフォーマンスコンピューティングワークロードに共通する問題は、ホットスポットです。ホットスポットとは、クラスターまたはネットワークの特定の部分が他の領域と比較して著しく高い負荷を受ける状況であり、その場所に過剰なデータトラフィックが集中することで、パフォーマンスのボトルネックが発生し、全体的な効率が低下します。たとえば、実行中のジョブで1つまたは複数のファイルへの需要が高まると、そのファイルへのリクエスト（ボリュームアフィニティ経由）を処理するために使用されるCPUでボトルネックが発生します。FlexCacheは、このボトルネックの軽減に役立ちますが、適切に設定する必要があります。

このドキュメントでは、FlexCacheを設定してホットスポットを修正する方法について説明します。



2024年7月より、これまでPDF形式で公開されていたテクニカルレポートの内容がONTAP製品ドキュメントに統合されました。このONTAPホットスポット修復テクニカルレポートの内容は、発行日時点で最新のものであり、以前の形式で作成されたものではありません。

主要な概念

ホットスポット修復を計画する際には、これらの基本的な概念を理解することが重要です。

- **高密度FlexCache (HDF)** : キャッシュ容量の要件が許す限り少ないノードにまたがるように凝縮されたFlexCache
- **HDF アレイ (HDFA)** : 同じオリジンのキャッシュである HDF のグループで、クラスター全体に分散されています。
- **SVM 間 HDFA** : サーバー仮想マシン (SVM) ごとに HDFA からの 1 つの HDF
- **SVM内HDFA** : 1つのSVM内のHDFA内のすべてのHDF
- **東西トラフィック** : 間接的なデータアクセスから生成されるクラスター バックエンド トラフィック

次の手順

- "高密度FlexCacheを使用してホットスポットの修正に役立つアーキテクチャの構築方法を理解する"
- "FlexCacheアレイ密度を決定する"
- "HDF の密度を決定し、SVM 間 HDFA と SVM 内 HDFA を使用した NFS で HDF にアクセスするかどうかを決定します。"
- "ONTAP 構成でクラスター内キャッシュを使用するメリットを実現するために、HDFA とデータ LIF を構成する"
- "クライアント構成を使用して ONTAP NAS 接続を分散するようにクライアントを構成する方法を学びます"

ONTAP FlexCache ホットスポット修復ソリューションの設計

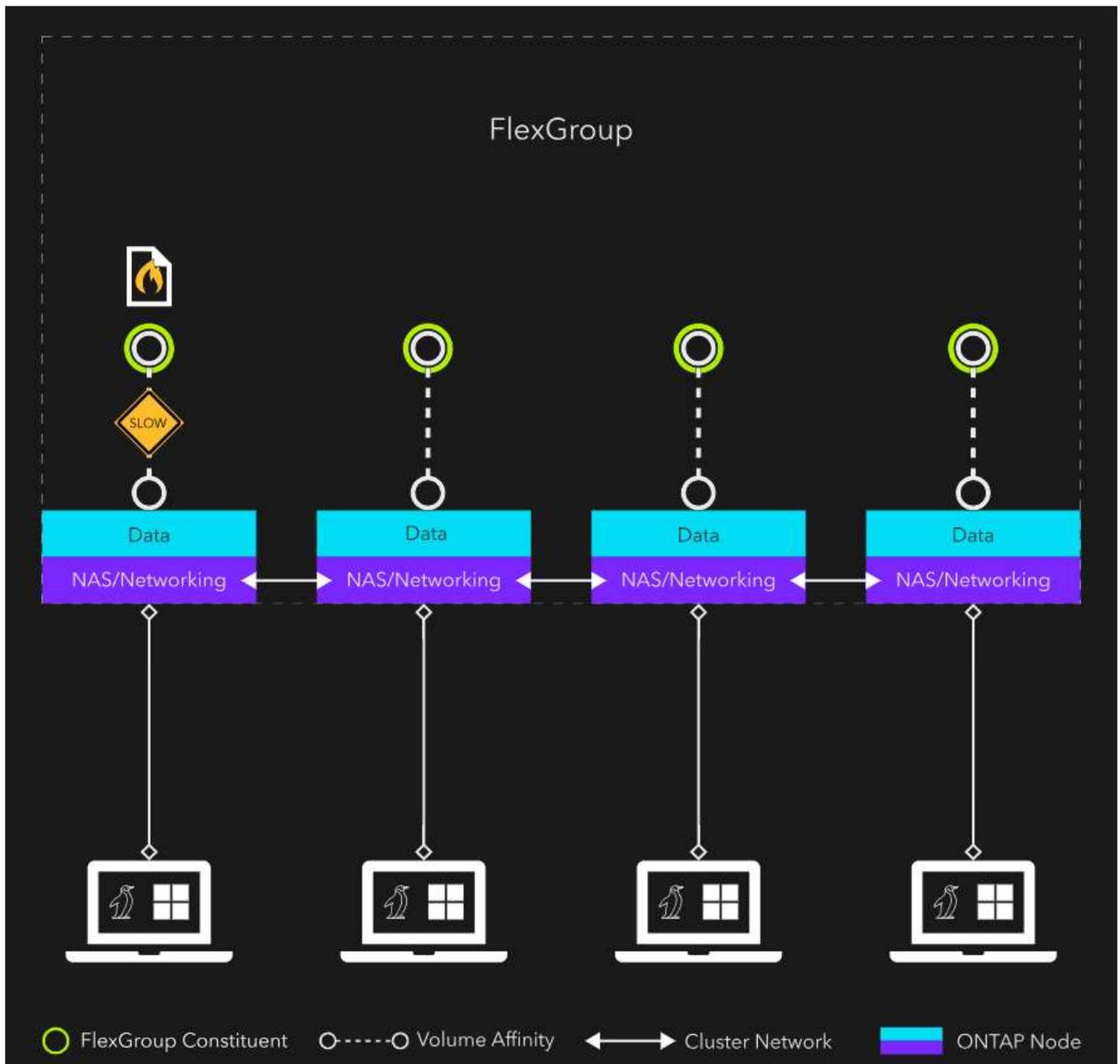
ホットスポットを改善するには、ボトルネックの根本的な原因、自動プロビジョニングされたFlexCacheが不十分な理由、そして効果的なFlexCacheソリューションを設計するために必要な技術的な詳細を探求する必要があります。高密度FlexCacheアレイ（HDFA）を理解して実装することで、高負荷ワークロードにおけるパフォーマンスを最適化し、ボトルネックを解消できます。

ボトルネックを理解する

次の画像は、典型的な単一ファイルのホットスポット シナリオを示しています。ボリュームは、ノードごとに1つのコンスティチュエントを持つFlexGroupであり、ファイルはノード1に存在します。

NASクライアントのネットワーク接続をすべてクラスター内の異なるノードに分散させた場合でも、ホットファイルが存在するボリュームアフィニティを処理するCPUでボトルネックが発生します。また、ファイルが存在するノード以外のノードに接続しているクライアントからの呼び出しによって、クラスター ネットワークトラフィック（East-Westトラフィック）が発生します。East-Westトラフィックのオーバーヘッドは通常は小さいですが、ハイパフォーマンス コンピューティングのワークロードでは、わずかな影響も無視できません。

図1：FlexGroup単一ファイルホットスポットシナリオ

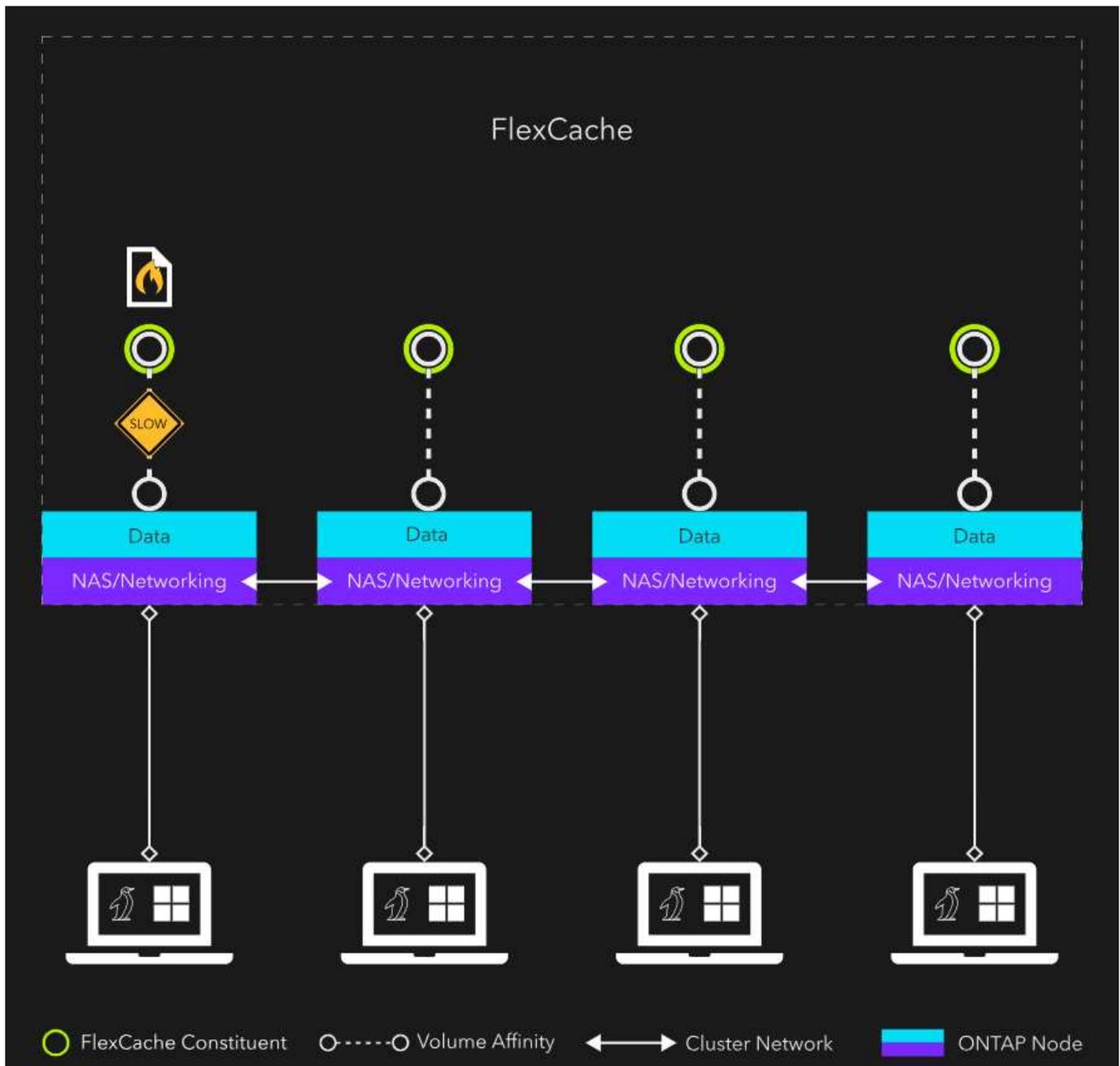


自動プロビジョニングされたFlexCacheが解決策ではない理由

ホットスポットを解決するには、CPUボトルネックを排除し、可能であれば東西トラフィックも排除します。FlexCacheは、適切に設定すれば役立ちます。

以下の例では、FlexCacheはSystem Manager、NetApp Console、またはデフォルトのCLI引数を使用して自動プロビジョニングされています。図1と図2は一見同じように見えます。どちらも4ノードの単一構成NASコンテナです。唯一の違いは、図1のNASコンテナがFlexGroupであり、図2のNASコンテナがFlexCacheであることです。どちらの図も同じボトルネックを示しています：ホットファイルへのアクセスを提供するボリュームアフィニティに使用されるノード1のCPU、およびレイテンシに影響を与えるEast-Westトラフィックです。自動プロビジョニングされたFlexCacheでは、このボトルネックは解消されていません。

図2：自動プロビジョニングのFlexCacheシナリオ



FlexCacheの構造

ホットスポット修復のためのFlexCacheを効果的に設計するには、FlexCacheに関する技術的な詳細を理解する必要があります。

FlexCacheは常にスパースなFlexGroupです。FlexGroupは複数のFlexVolsで構成されています。これらのFlexVolsはFlexGroup構成要素と呼ばれます。デフォルトのFlexGroupレイアウトでは、クラスター内のノードごとに1つ以上の構成要素が存在します。構成要素は抽象化レイヤの下で「縫い合わされ」、単一の大きなNASコンテナとしてクライアントに提示されます。FlexGroupにファイルが書き込まれると、取り込みヒューリスティックによって、ファイルがどの構成要素に保存されるかが決定されます。クライアントのNAS接続を含む構成要素の場合もあれば、別のノードの場合もあります。すべてが抽象化レイヤの下で動作し、クライアントからは見えないため、場所は関係ありません。

FlexGroupに関するこの理解をFlexCacheに適用してみましょう。FlexCacheはFlexGroup上に構築されている

ため、デフォルトでは図1に示すように、クラスタ内のすべてのノードに構成要素を持つ単一のFlexCacheが存在します。ほとんどの場合、これは素晴らしいことです。クラスタ内のすべてのリソースを活用していることとなります。

しかし、ホットファイルの修復には、2つのボトルネック（単一ファイルのCPUと東西トラフィック）があるため、この方法は理想的ではありません。ホットファイル用に各ノードに構成要素を持つFlexCacheを作成した場合、そのファイルは構成要素の1つにのみ保存されます。つまり、ホットファイルへのすべてのアクセスを処理するCPUは1つということになります。また、ホットファイルに到達するために必要な東西トラフィック量を制限することも重要です。

このソリューションは、高密度のFlexCachesのアレイです。

高密度FlexCacheの構造

高密度FlexCache（HDF）では、キャッシュされたデータの容量要件で許容される限り、できるだけ少ないノードに構成要素を配置します。目標は、キャッシュを単一のノードに配置することです。容量要件によってそれが不可能な場合は、代わりに少数のノードにのみ構成要素を配置できます。

例えば、24ノードクラスタには、3つの高密度FlexCachesを持つことができます：

- ノード1～8にまたがるもの
- ノード9～16にまたがる2つ目
- 3番目はノード17～24にまたがる

これら3つのHDFは、1つの高密度FlexCacheアレイ（HDFA）を構成します。ファイルが各HDF内に均等に分散されている場合、クライアントから要求されたファイルがフロントエンドNAS接続のローカルに存在する確率は8分の1です。もし、それぞれ2つのノードにまたがるHDFが12個ある場合、ファイルがローカルに存在する確率は50%です。HDFを1つのノードに縮小し、24個作成すれば、ファイルがローカルに存在することが保証されます。

この構成により、東西トラフィックがすべて排除され、最も重要な点として、ホットファイルにアクセスするための24個のCPU/ボリューム アフィニティが提供されます。

次の手順

["FlexCacheアレイ密度を決定する"](#)

関連情報

["FlexGroupおよびTRに関するドキュメント"](#)

ONTAP FlexCache密度を決定する

ホットスポット修復の設計において最初に行うべきことは、FlexCacheの密度を把握することです。以下の例は4ノードのクラスタです。ファイル数は各HDF内のすべての構成要素に均等に分散されていると仮定します。また、フロントエンドNAS接続もすべてのノードに均等に分散されていると仮定します。

これらの例は使用できる唯一の構成ではありませんが、スペース要件と使用可能なリソースが許す限り多くのHDFを作成するための設計原則を理解する必要があります。

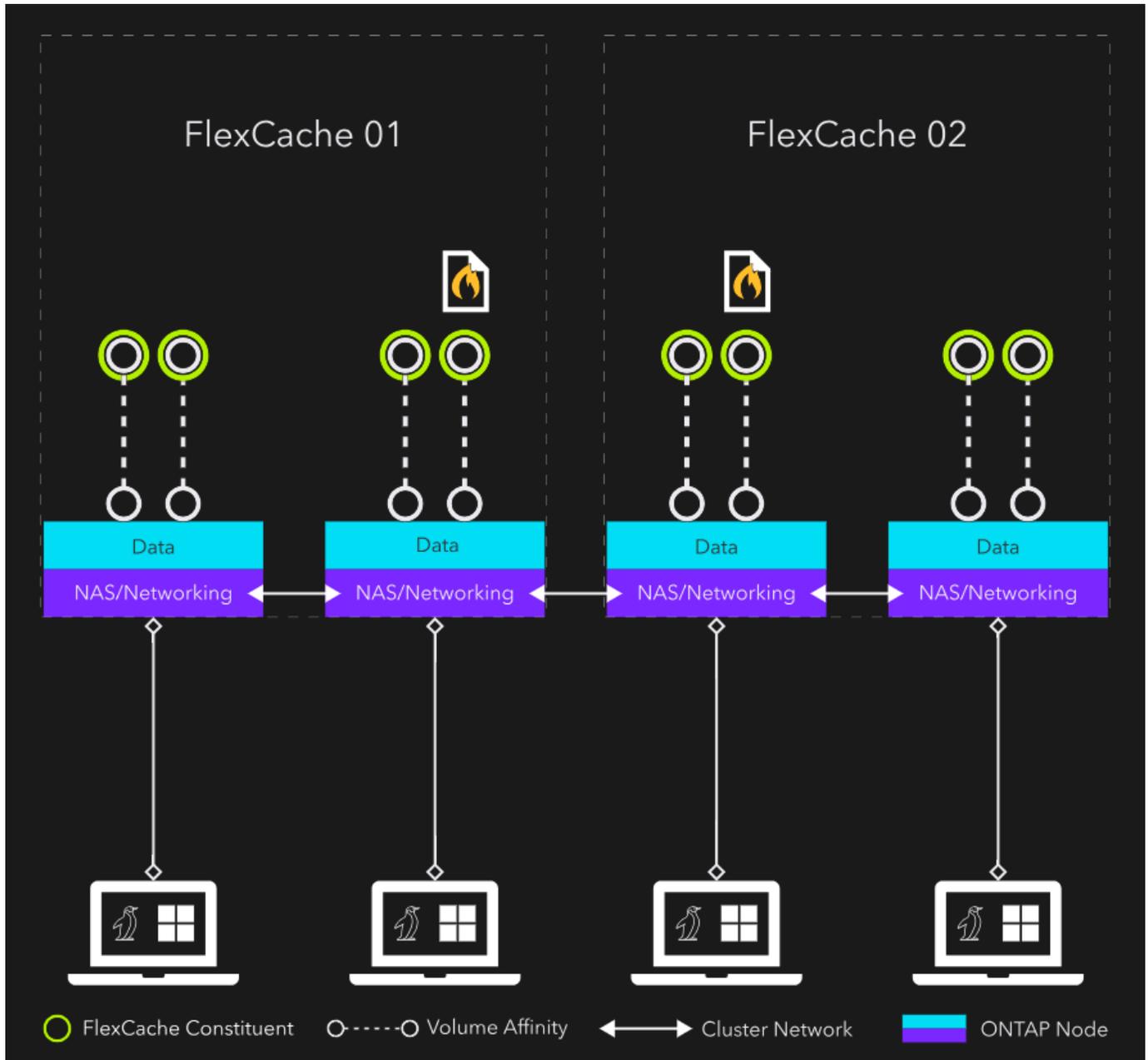


HDFA は次の構文を使用して表されます： $HDFs \text{ per HDFA} \times nodes \text{ per HDF} \times constituents \text{ per node per HDF}$

2x2x2 HDFA 構成

図1これは2x2x2 HDFA構成の例です：2つのHDFがそれぞれ2つのノードにまたがり、各ノードには2つの構成ボリュームが含まれています。この例では、各クライアントがホットファイルに直接アクセスできる可能性は50%です。4つのクライアントのうち2つは東西方向のトラフィックを持っています。重要なのは、HDFが2つあることです。つまり、ホットファイルのキャッシュが2つあるということです。ホットファイルへのアクセスを処理するCPU/ボリュームアフィニティが2つあります。

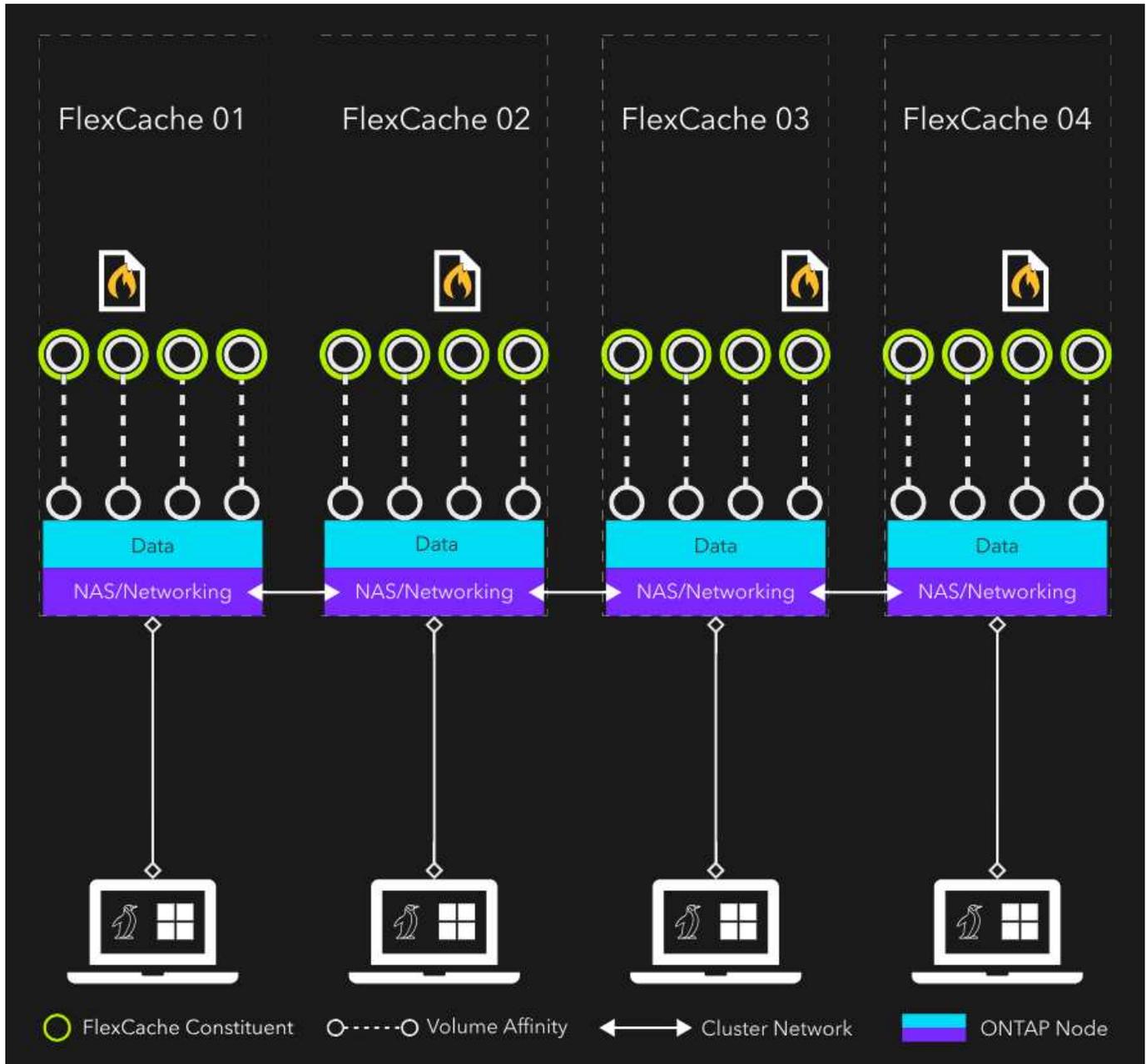
図1：2x2x2 HDFA構成



4x1x4 HDFA構成

図2は最適な構成を表しています。これは4x1x4 HDFA構成の例です：4つのHDFがそれぞれ単一のノードに保持され、各ノードには4つの構成要素が含まれています。この例では、各クライアントはホットファイルのキャッシュに直接アクセスできることが保証されています。4つの異なるノードに4つのキャッシュファイルが存在するため、4つの異なるCPU/ボリュームアフィニティがホットファイルへのアクセスをサービス提供します。さらに、East-Westトラフィックは生成されません。

図2：4x1x4 HDFA構成



次の手順

HDF の密度を決定した後、"SVM間HDFAおよびSVM内HDFA" を使用して NFS で HDF にアクセスする場合は、別の設計上の決定を行う必要があります。

ONTAP SVM 間または SVM 内の HDFA オプションを決定する

HDF の密度を決定したら、NFS を使用して HDF にアクセスするかどうかを決定し、SVM 間 HDFA と SVM 内 HDFA のオプションについて学習します。



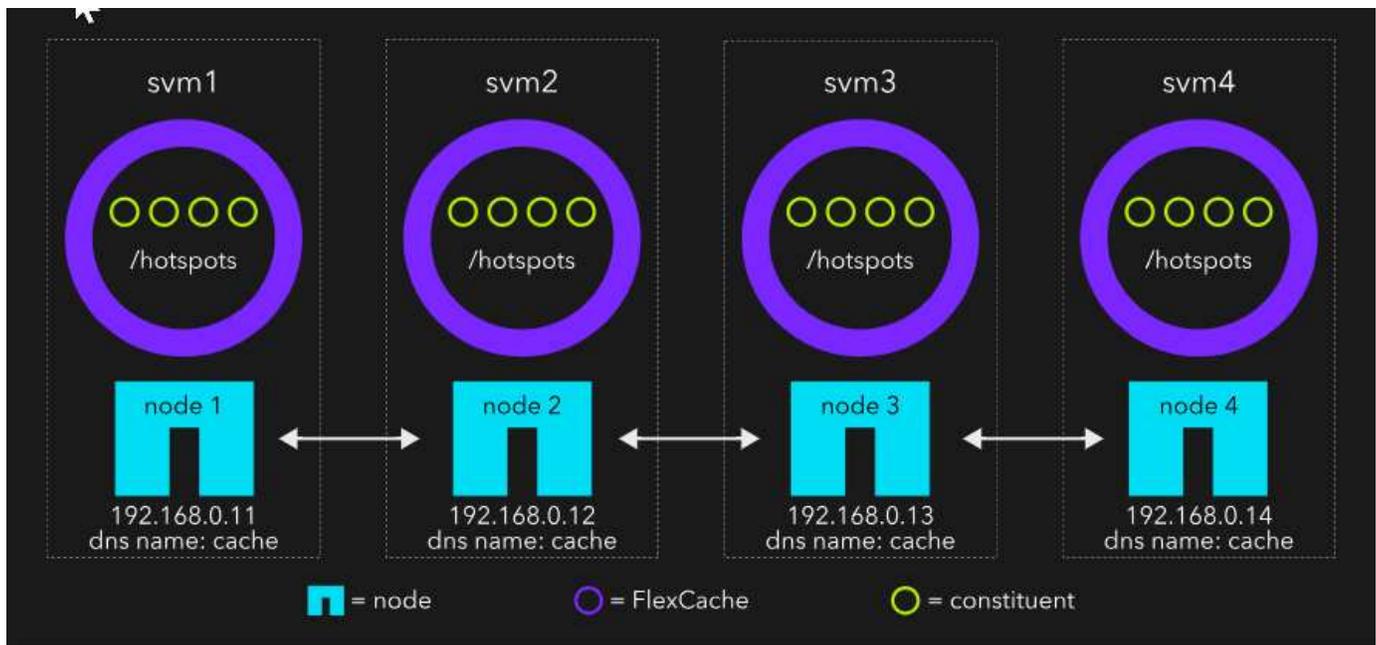
SMBクライアントのみがHDFにアクセスする場合は、すべてのHDFを単一のSVMに作成する必要があります。ロード バランシングにDFSターゲットを使用する方法については、Windowsクライアント構成を参照してください。

SVM間HDFA導入

SVM間HDFAでは、HDFA内の各HDFごとにSVMを作成する必要があります。これにより、HDFA内のすべてのHDFが同じジャンクションパスを持つようになり、クライアント側での設定が容易になります。

この図1例では、各HDFはそれぞれ独自のSVMに存在します。これはSVM間HDFAデプロイメントです。各HDFには/hotspotsのジャンクションパスがあります。また、すべてのIPアドレスにはホスト名キャッシュのDNS Aレコードがあります。この構成では、DNSラウンド ロビンを利用して、異なるHDF間でマウントのロード バランシングを行っています。

図1：4x1x4インター-SVM HDFA構成

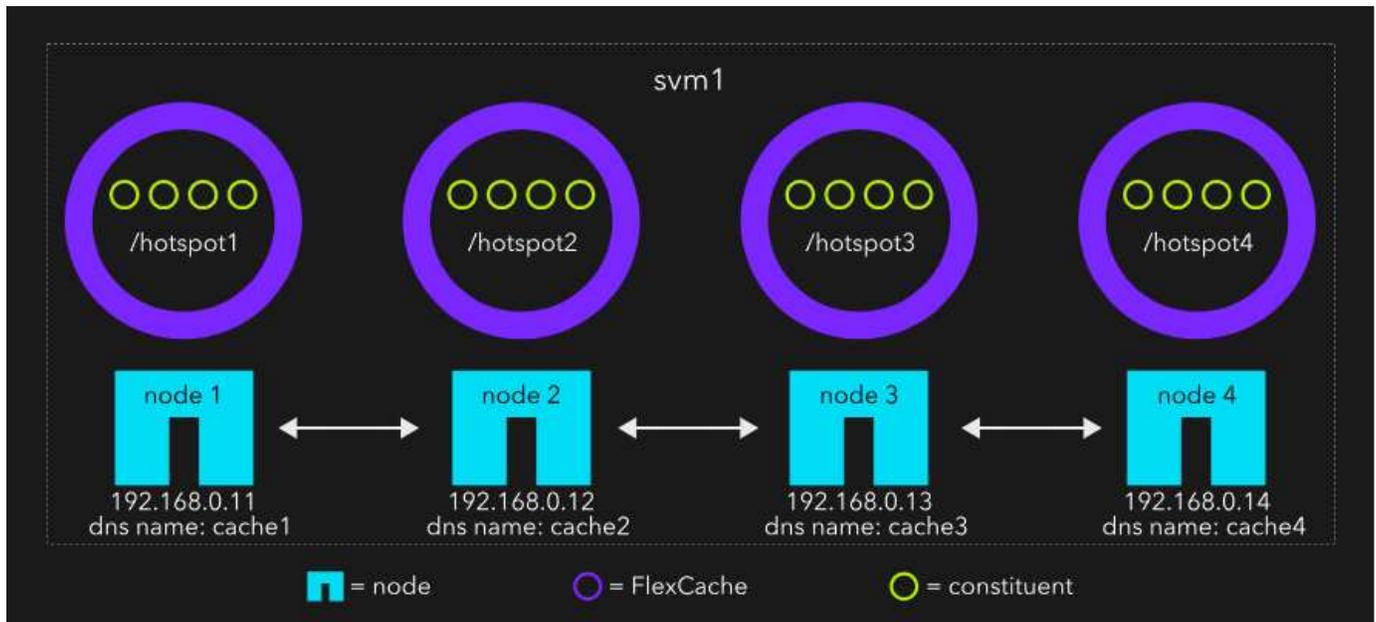


SVM 内 HDFA 展開

SVM内構成では、各HDFに固有のジャンクションパスが必要ですが、すべてのHDFは1つのSVMに存在します。ONTAPではSVMが1つしか必要ないため、この設定は簡単ですが、Linux側では`autofs`とONTAPでのデータLIF配置により、より高度な構成が必要になります。

図2の例では、すべてのHDFが同じSVM内にあります。これはSVM内HDFAデプロイメントであり、ジャンクションパスは一意である必要があります。ロード バランシングを適切に機能させるには、各IPに一意的なDNS名を作成し、ホスト名が解決するデータLIFをHDFが存在するノードにのみ配置する必要があります。また、"Linuxクライアントの設定"で説明されているように、`autofs`で複数のエントリを設定する必要があります。

図2：4x1x4 SVM内HDFA構成



次の手順

HDFA をどのように展開するかが決まったので、"HDFAを展開し、分散形式でアクセスできるようにクライアントを構成する"。

HDFAとONTAPデータLIFを設定する

このホットスポット修復ソリューションのメリットを享受するには、HDFAとデータLIFを適切に設定する必要があります。このソリューションでは、オリジンとHDFAを同一クラスタに配置し、クラスタ内キャッシュを使用します。

次に、2つのHDFAサンプル構成を示します：

- 2x2x2 インター SVM HDFA
- 4x1x4 イントラ SVM HDFA

タスク概要

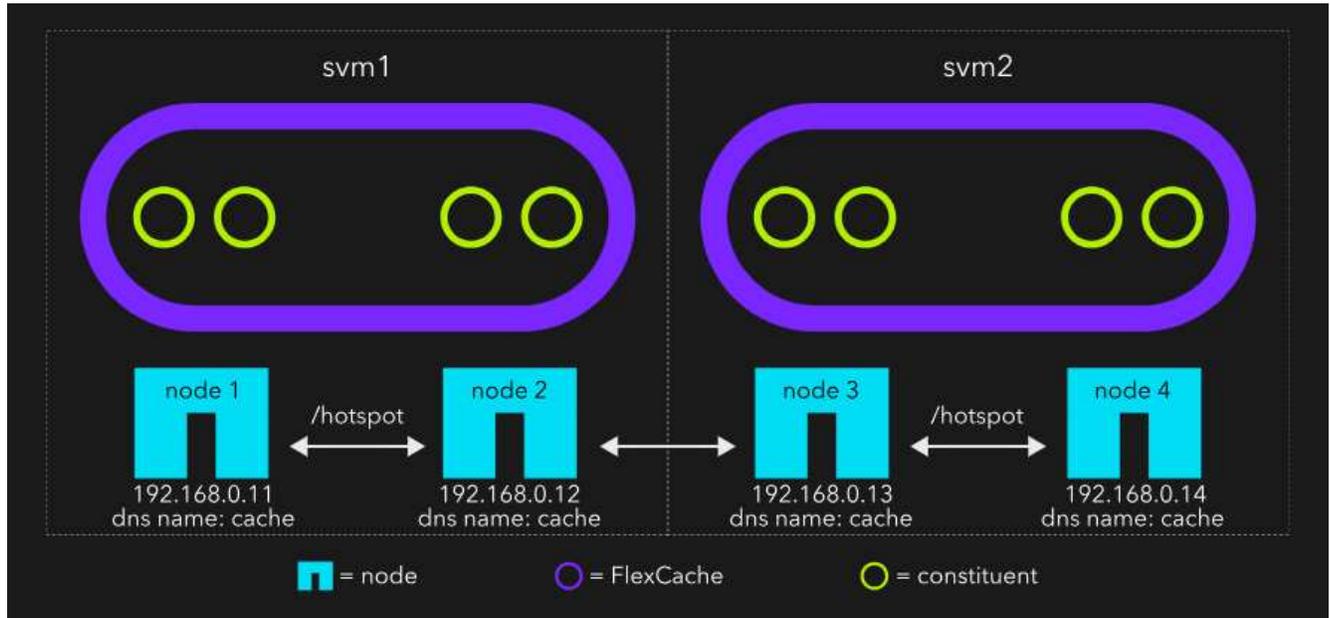
ONTAP CLIを使用して、この高度な設定を実行します。`flexcache create`コマンドで使用する必要がある設定が2つあり、設定されていないことを確認する必要がある設定が1つあります：

- `-aggr-list`：HDF を制限するノードまたはノードのサブセットに存在するアグリゲート、またはアグリゲートのリストを指定します。
- `-aggr-list-multiplier`：`aggr-list` オプションにリストされているアグリゲートごとに作成されるコンスチチュエントの数を決定します。2つのアグリゲートがリストされていて、この値を`2`に設定した場合、4つのコンスチチュエントが作成されます。NetAppでは、アグリゲートごとに最大8つのコンスチチュエントを推奨していますが、16でも十分です。
- `-auto-provision-as`：タブキーで移動した場合、CLIは自動入力を試みて値を`flexgroup`に設定します。これが設定されていないことを確認してください。設定されている場合は削除してください。

2x2x2インター-SVM HDFA構成を作成する

1. 図 1 に示すように 2x2x2 インター SVM HDFA を構成するには、準備シートに記入します。

図1：2x2x2 Inter-SVM HDFAレイアウト



SVM	HDF あたりのノード 数	アグリゲート	ノードあたりの 構成要素	ジャンクション パス	データ LIF IP
svm1	node1、node2	aggr1、aggr2	2	/ホットスポット	192.168.0.11,19 2.168.0.12
svm2	node3、node4	aggr3、aggr4	2	/ホットスポット	192.168.0.13,19 2.168.0.14

2. HDFを作成します。以下のコマンドを準備シートの各行ごとに1回ずつ、計2回実行します。2回目の実行では、必ず `vserver` と `aggr-list` の値を調整してください。

```
cache::> flexcache create -vserver svm1 -volume hotspot -aggr-list
aggr1,aggr2 -aggr-list-multiplier 2 -origin-volume <origin_vol> -origin
-vserver <origin_svm> -size <size> -junction-path /hotspot
```

3. データ LIF を作成します。コマンドを 4 回実行し、準備シートに記載されているノード上の SVM ごとに 2 つのデータ LIF を作成します。各反復で値を適切に調整してください。

```
cache::> net int create -vserver svm1 -home-port e0a -home-node node1
-address 192.168.0.11 -netmask-length 24
```

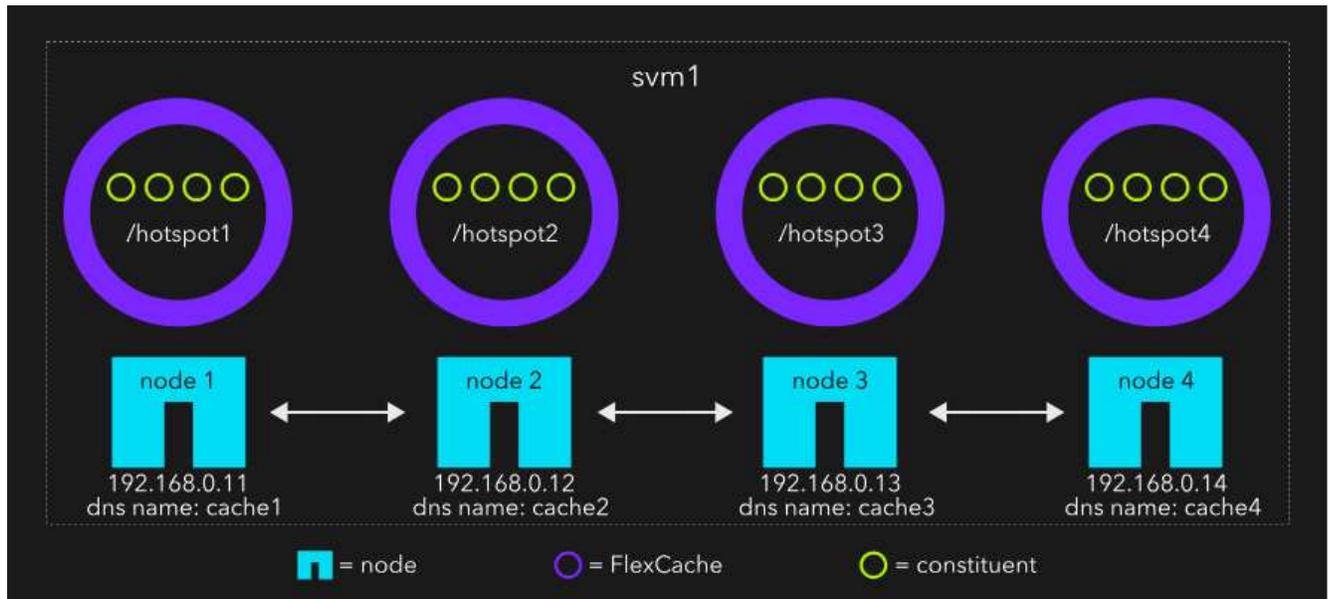
次の手順

次に、HDFAを適切に利用できるようにクライアントを設定する必要があります。["クライアント構成"](#)を参照してください。

4x1x4のSVM内HDFを構成する

1. 図2に示すように4x1x4 インター SVM HDFを構成するには、準備シートに記入します。

図2：4x1x4 SVM内HDFレイアウト



SVM	HDF あたりのノード 数	アグリゲート	ノードあたりの 構成要素	ジャンクション パス	データ LIF IP
svm1	ノード1	aggr1	4	/ホットスポット1	192.168.0.11
svm1	ノード2	aggr2	4	/hotspot2	192.168.0.12
svm1	ノード3	aggr3	4	/hotspot3	192.168.0.13
svm1	ノード4	aggr4	4	/hotspot4	192.168.0.14

2. HDFを作成します。以下のコマンドを準備シートの各行ごとに1回ずつ、計4回実行します。各反復ごとに`aggr-list`と`junction-path`の値を調整してください。

```
cache::> flexcache create -vserver svm1 -volume hotspot1 -aggr-list
aggr1 -aggr-list-multiplier 4 -origin-volume <origin_vol> -origin
-vserver <origin_svm> -size <size> -junction-path /hotspot1
```

3. データLIFを作成します。コマンドを4回実行し、SVMに合計4つのデータLIFを作成します。ノードごとに1つのデータLIFが必要です。各反復で値を適切に調整してください。

```
cache::> net int create -vserver svm1 -home-port e0a -home-node node1
-address 192.168.0.11 -netmask-length 24
```

次の手順

次に、HDFAを適切に利用できるようにクライアントを設定する必要があります。"クライアント構成"を参照してください。

ONTAP NAS接続を分散するようにクライアントを構成する

ホットスポットを解決するには、CPUボトルネックを防ぐ役割を果たすようにクライアントを適切に設定します。

Linuxクライアントの設定

SVM内またはSVM間のHDFAデプロイメントを選択した場合でも、Linuxで`autofs`を使用して、クライアントが異なるHDF間で負荷分散されていることを確認する必要があります。`autofs`の設定は、SVM間とSVM内で異なります。

開始する前に

`autofs`と適切な依存関係をインストールする必要があります。詳細については、Linuxのドキュメントを参照してください。

タスク概要

説明する手順では、次のエントリを含むサンプルの`/etc/auto_master`ファイルを使用します：

```
/flexcache auto_hotspot
```

この設定により、プロセスが`/flexcache`ディレクトリにアクセスしようとするたびに、`autofs`が`/etc`ディレクトリ内の`auto_hotspot`というファイルを検索します。`auto_hotspot`ファイルの内容により、`/flexcache`ディレクトリ内にマウントするNFSサーバとジャンクションパスが指定されます。ここで説明する例は、`auto_hotspot`ファイルのさまざまな設定です。

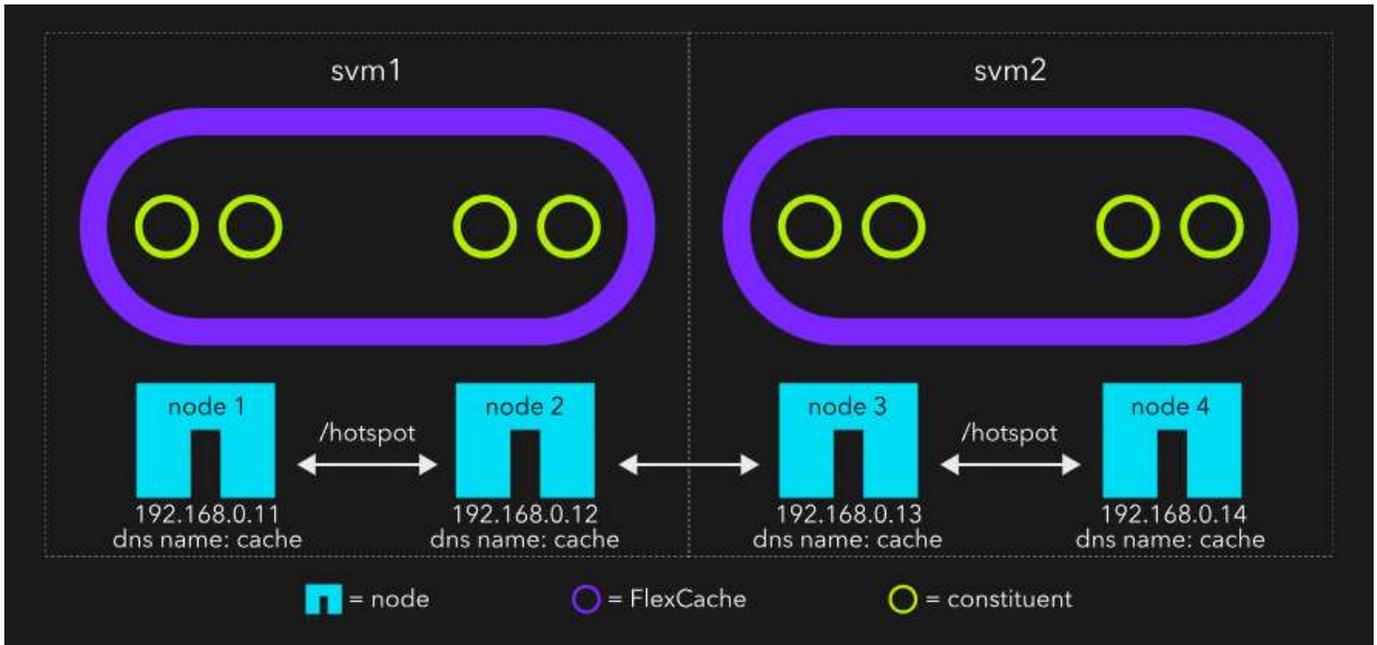
SVM内HDFA autofs設定

以下の例では、[図1](#)の図の`autofs`マップを作成します。各キャッシュは同じジャンクションパスを持ち、ホスト名`cache`には4つのDNS Aレコードがあるため、必要な行は1行だけです：

```
hotspot cache:/hotspot
```

このシンプルな1行で、NFSクライアントはホスト名`cache`のDNSルックアップを実行します。DNSはラウンドロビン方式でIPアドレスを返すように設定されています。これによりフロントエンドNAS接続が均等に分散されます。クライアントはIPアドレスを受信すると、ジャンクションパス`/hotspot`を`/flexcache/hotspot`にマウントします。接続先のSVMはSVM1、SVM2、SVM3、またはSVM4ですが、どのSVMに接続しても構いません。

図1：2x2x2インター-SVM HDFA



SVM内HDFA autofs設定

以下の例では、図2の図の `autofs` マップを作成します。NFSクライアントがHDFジャンクションパス デプロイメントに含まれるIPをマウントにする必要があります。つまり、`/hotspot1`をIPアドレス192.168.0.11以外でマウントしないようにする必要があります。そのためには、`auto_hotspot` マップ内の1つのローカル マウント場所に対して、4つのIP/ジャンクションパスのペアをすべてリストします。

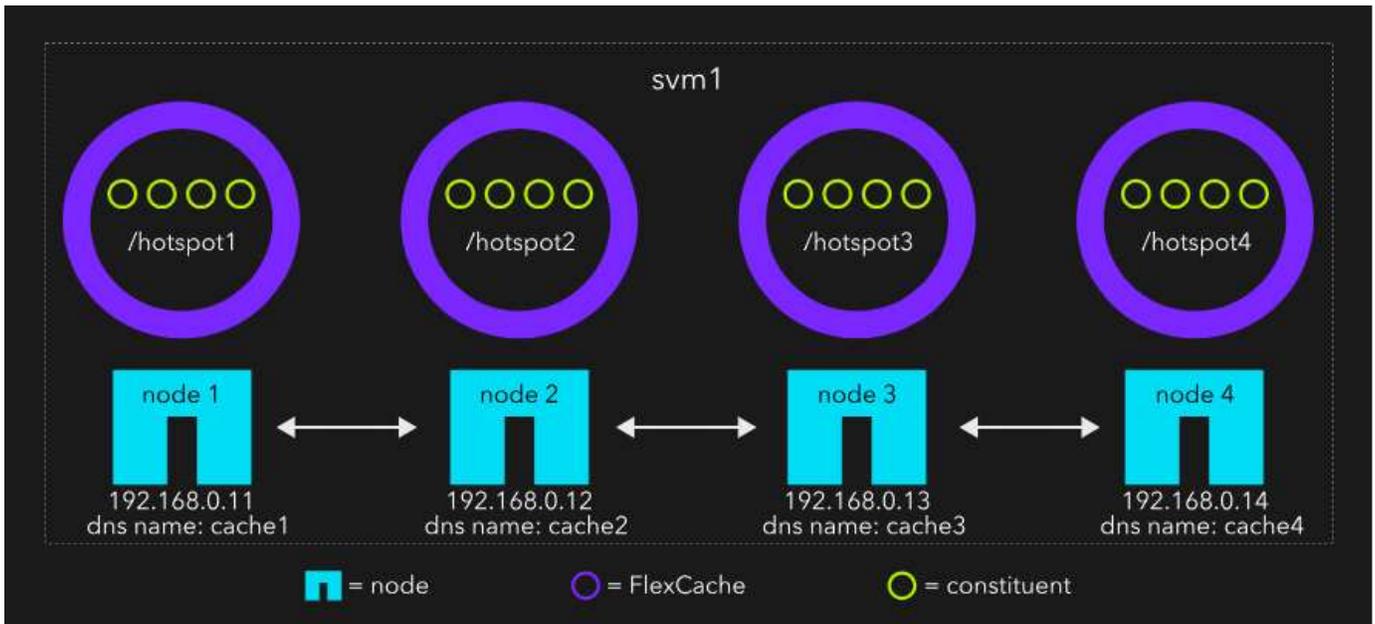
i 次の例のバックスラッシュ (\) はエントリを次の行に継続し、読みやすくします。

```
hotspot    cache1:/hostspot1 \
           cache2:/hostspot2 \
           cache3:/hostspot3 \
           cache4:/hostspot4
```

クライアントが `/flexcache/hotspot` にアクセスしようとする時、`autofs` は4つのホスト名すべてに対して前方参照を実行します。4つのIPアドレスすべてがクライアントと同じサブネット内にあるか、異なるサブネット内にあると仮定すると、`autofs` は各IPアドレスに対してNFS NULL pingを発行します。

このNULL pingでは、ONTAPのNFSサービスによるパケット処理が必要ですが、ディスクアクセスは必要ありません。最初に返されるpingが、`autofs` がマウントに選択するIPアドレスとジャンクションパスになります。

図2：4x1x4 SVM内HDFA



Windowsクライアントの設定

Windowsクライアントでは、SVM内HDFSを使用する必要があります。SVM内の異なるHDFS間で負荷分散を行うには、各HDFSに一意的な共有名を追加する必要があります。その後、"[Microsoftのドキュメント](#)"の手順に従って、同じフォルダに複数のDFSターゲットを実装してください。

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権を有し、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権については、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。