

# 1 NCM 水平スケーラビリティの概念

NCM 水平スケーラビリティでは、複数の NCM コアを 1 つの NCM データベースに接続します。この設定では、NCM コアは論理ユニットとして動作します。すべての NCM ユーザに対して、NCM コアごとに同じ情報が表示されます。NCM は水平スケーラビリティのトポロジに従って NCM コア間にデバイス タスクを分散します。ユーザは、どの NCM コアが特定のデバイスにアクセスするのかわかる必要はありません。

NCM 水平スケーラビリティにより、1 つの NCM コアの範囲を超えて NCM を拡張できます。2 ~ 5 つのアクティブな NCM コアを 1 つの NCM データベースに接続できます。スケーラビリティの詳細については、NCM の「Maximum Supported Managed Environment」を参照してください。

▶ NCM 1.8.01 では、アクティブおよび非アクティブな NCM コアの概念が導入されました。

- アクティブな NCM コアは NCM のフル機能のインスタンスです。
- NCM コアが非アクティブである間は、次の条件が適用されます。
  - その NCM コアはタスクを実行しません。
  - ユーザは NCM にログインできません。
  - ユーザは、ディザスタ リカバリのスイッチオーバーまたはスイッチバック中に NCM コアの状態を設定する目的に限り、Telnet または SSH を通じて NCM のコマンドライン インターフェイスに接続できます。

NCM 水平スケーラビリティは NCM のディザスタ リカバリ アーキテクチャの基盤です。NCM のディザスタ リカバリの実装については、NCM を参照してください。

NCM 1.8.02 以降、NCM 水平スケーラビリティでは、NCM コアが使用できなくなったときに自動フェールオーバーを実行することで、NCM ハイ アベイラビリティを提供します。詳細については、[第 3 章「水平スケーラビリティ環境での NCM コア フェールオーバー」](#)を参照してください。

▶ NCM 1.8 には、NCM 1.7.02 までのバージョンの機能が含まれます。したがって、NCM 1.7.03 で提供される機能は、NCM 1.8.01 では使用できますが、NCM 1.8 では使用できません。

▶ このマニュアルは、新しい情報が入手され次第更新されます。最新の更新を確認するか、またはマニュアルの最新版を使用していることを確認するには、次の URL にアクセスします。

<http://h20230.www2.hp.com/selfsolve/manuals>

詳細については、「[Documentation Updates](#)」(P.3)を参照してください。

この章では、次の事項について説明します。

- 「[水平スケーラビリティのアーキテクチャ](#)」(P.6)
- 「[水平スケーラビリティのトポロジ](#)」(P.6)

## 水平スケーラビリティのアーキテクチャ

NCM 水平スケーラビリティ環境では、NCM データベースは NCM コア間で共有される情報のプライマリ ソースです。待機キューの各タスクのデータベース レコードには、そのタスクを実行する予定の NCM コアが含まれます。

さらに、水平スケーラビリティ環境の NCM コアの各ペア間に軽量チャンネルがあります。このチャンネルは、ソフトウェア イメージ、ドライバ パッケージ、NCM 設定の一部など、特定のファイル システム オブジェクトの NCM コア間の同期に Java Remote Method Invocation (RMI) を使用します。

水平スケーラビリティ環境でアクティブな NCM コアの最大数は 5 です。

ディザスタ リカバリ設定では、アクティブおよび非アクティブな NCM コアの最大数は 9 です。これは、ディザスタ リカバリ シナリオの NCM コアの最大数が、プライマリ ロケーションに 5 つのアクティブな NCM コアおよびディザスタ リカバリ ロケーションに 4 つの非アクティブな NCM コアであることを意味します。ディザスタ リカバリの設定中に、9 つの NCM コアすべてを一度にアクティブにできます。ただし、デバイス管理はディザスタ リカバリ ロケーションの NCM コアで実行されません。

NCM 水平スケーラビリティは、NCM コアとデータベースが WAN リンクによって分離されている WAN で使用するために設計されていません。最適なパフォーマンスを得るために、各 NCM コアを NCM データベース サーバと同じデータセンターに配置することを推奨します。

NCM 水平スケーラビリティは、Oracle または Microsoft SQL Server データベースでのみサポートされます。サポートされているバージョンについては、NCM を参照してください。

## 水平スケーラビリティのトポロジ

NCM 水平スケーラビリティは次のいずれかのトポロジに対して設定できます。

- 「分散ラウンドロビン」(P.7)
- 「コア バインディング」(P.11)

これらのトポロジの特徴は NCM が NCM コアにデバイス タスクを割り当てる方法です。デバイス タスクとは、デバイスに対して実行する NCM タスクのことです。

サポートされるトポロジごとに、水平スケーラビリティ環境の NCM コアを特殊化するためのバリエーションが 1 つ以上用意されています。

コア バインディングはデフォルトの NCM 水平スケーラビリティ トポロジです。ただし、どちらのトポロジも環境で使用できる場合に推奨される NCM 水平スケーラビリティ トポロジは、分散ラウンドロビンです。

水平スケーラビリティ環境には、ファイアウォールの背後にあるデバイスとの通信に使用する 1 つ以上の NCM サテライトを含めることができます。詳細については、「[「水平スケーラビリティ環境での NCM サテライトの設定」\(P.31\)](#)」を参照してください。

表 1 に、サポートされる水平スケーラビリティのトポロジの概要レベルでの比較を示します。

表 1 水平スケーラビリティのトポロジの比較

カテゴリ	分散ラウンドロビン	コア バインディング
ネットワーク アクセシビリティ	すべての NCM コアは、すべての管理対象デバイスと通信できる必要があります。	一部またはすべての NCM コアは、すべての管理対象デバイスと通信できません。
ロード シェアリング	各 NCM コアは、ほぼ同数のデバイスを管理します。NCM が特定のデバイスを管理する NCM コアを決定します。	パーティションがデバイスを管理する NCM コアを決定します。NCM 管理者は各パーティションを NCM コアに割り当て、NCM コア間にデバイス タスクの負荷を分散します。
フェールオーバー	均等なロード シェアリングを使用して自動的に行われます。NCM は、使用できない NCM コアによって管理されていたデバイスを管理する役割を残りの NCM コアに分散します。	不均等なロード シェアリングを使用して自動的に行われます。使用できない NCM コアからのすべてのパーティションが 1 つの NCM コアに割り当てられます。デバイスのアクセシビリティ制限によって、正常なフェールオーバーが妨げられる可能性があります。
フェールバック	NCM は、回復した NCM コアに管理対象デバイスを管理する役割を自動的に付与します。	回復した NCM コアにパーティションを手動で割り当てます。

## 分散ラウンドロビン

分散ラウンドロビン トポロジでは、図 1 (P.9) に示すように、各 NCM コアは、管理対象ネットワーク内のすべてのデバイスと通信できます。この設定は、パーティションの使用に依存しません。ただし、パーティションは他の理由で使用されている可能性があります。設定済みのアクセス制限内で、ユーザは任意の NCM コアにログインし、NCM インベントリ内の任意のデバイスに対して実行するタスクを作成できます。NCM は NCM コア間にタスクを分散します。また、ユーザは、任意の NCM コアの任意のタスクの結果を表示できます。

複数のデバイスに対して同じアクションを実行するタスクの場合、NCM は、対象の各デバイスに 1 つの子デバイス タスクを作成する軽量グループ タスクを実行します。分散ラウンドロビン トポロジでは、グループ タスクは、それが作成された NCM コアで実行されますが、各子デバイス タスクは、任意の NCM コアで実行できます。

分散ラウンドロビン トポロジでは、NCM は、管理対象ネットワークのデバイス タスクに対する役割を使用可能な NCM コアに分散し、デバイス タスクの負荷のおおよそのバランスを取ります。この分散は、デバイス タスクの数だけに基づいて行われます。これは、NCM コアの設定またはデバイス設定のサイズを考慮しません。特定のデバイスに対するタスクが NCM コアに割り当てられると、そのデバイスのすべてのタスクは、水平スケーラビリティ環境で実行中の NCM コアの数が変化するまで、同じ NCM コアで実行されます。その時点で、新しいデバイス タスクに関して、NCM は、水平スケーラビリティ環境の NCM コアへのデバイスの分散を再計算します。

分散ラウンドロビン トポロジの利点は次のとおりです。

- NCM コア間でのデバイス タスクの自動ロード シェアリング。
- タスクの自動フェールオーバーおよびフェールバック。

- デフォルトでは、NCM は、常にデバイスの 1 つのタスクだけを実行します。



この動作は、[Wait option] チェックボックスをオフにすることで、Run Command Script タスクに対して上書きできます。

分散ラウンドロビン トポロジの制限は次のとおりです。

- すべてのデバイスはすべての NCM コアにアクセスできるか、NCM サテライトによって管理される必要があります。
- NCM 管理者は特定のデバイスのタスクを実行する NCM コアを制御できません。



NCM 1.8.01 および NCM 1.7.03 よりも前は、分散ラウンドロビン トポロジは「トポロジ 2」と呼ばれていました。

## 分散ラウンドロビンのバリエーション

NCM は標準分散ラウンドロビン トポロジの 1 つのバリエーション（ユーザインタラクション用の分散ラウンドロビン バリエーション）をサポートします。

このバリエーションは、すべてのデバイス タスクと、NCM とユーザ間のすべての対話を切り離します。ユーザインタラクションには、すべての操作可能な NCM セッション、NCM プロキシセッション、NCM API を使用するツールまたはプログラム、統合製品からの接続などがあります。図 2 (P.10) を参照してください。

このバリエーションの主な利点は、ユーザインタラクション用に予約されたすべての NCM コアのリソースは、ユーザ要求に応答することだけに専念することです。さらに、このバリエーションには、標準分散ラウンドロビン トポロジと同じ利点と制限があります。



分散ラウンドロビン トポロジのこのバリエーションは、NCM 1.8.01 および NCM 1.7.03 で初めて導入されました。

## 分散ラウンドロビンの図解

ここでは、3 つの NCM コアの水平スケーラビリティ環境のアーキテクチャ図の例を示します。内容は次のとおりです。

- 図 1：アーキテクチャ例：標準分散ラウンドロビン
- 図 2：アーキテクチャ例：ユーザインタラクション用の分散ラウンドロビン バリエーション

図 1 アーキテクチャ例：標準分散ラウンドロビン

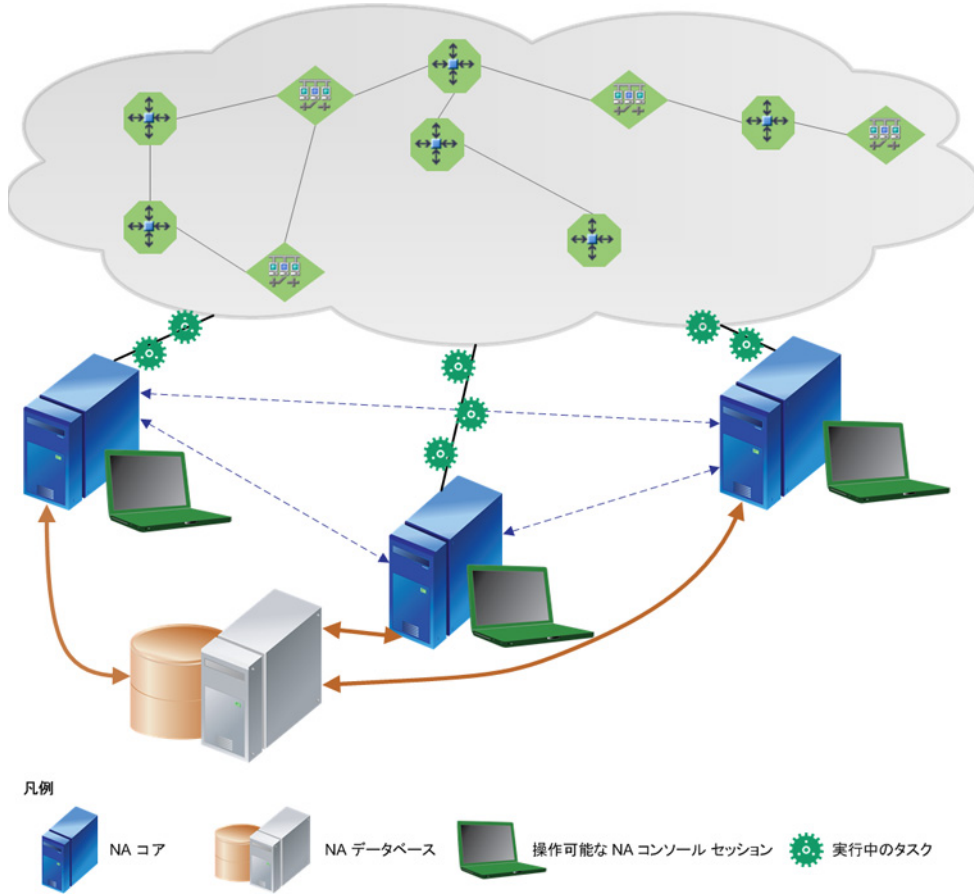
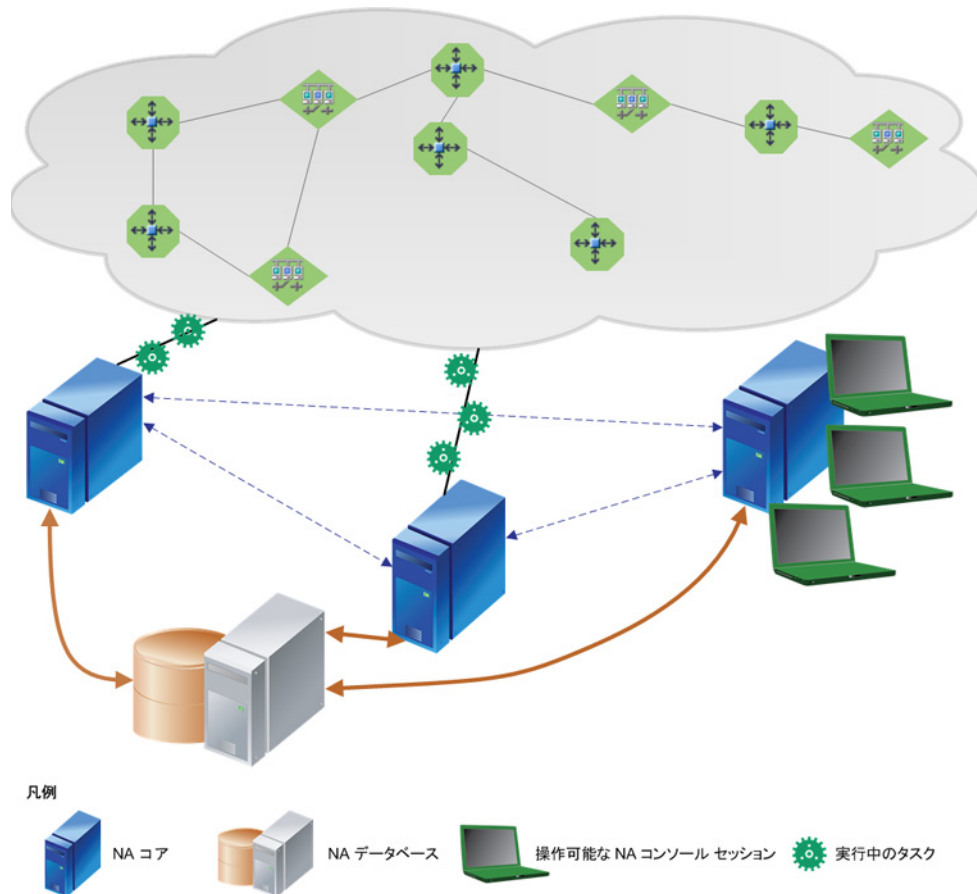


図2 アーキテクチャ例：ユーザインタラクション用の分散ラウンドロビンバリエーション



## コア バインディング

コア バインディング トポロジでは、[図 3 \(P.13\)](#) に示すように、各 NCM コアは固定の重複しない一連のデバイスと通信します。この設定では、各デバイスはパーティションに関連付けられ、各パーティションは 1 つの NCM コアに関連付けられます。設定済みのアクセス制限内で、ユーザは任意の NCM コアにログインし、NCM インベントリ内の任意のデバイスに対して実行するタスクを作成できます。NCM は、対象のデバイスに適した NCM コアに各タスクを割り当てます。また、ユーザは、任意の NCM コアの任意のタスクの結果を表示できます。

複数のデバイスに対して同じアクションを実行するタスクの場合、NCM は、対象の各デバイスに 1 つの子デバイス タスクを作成する軽量グループ タスクを実行します。コア バインディング トポロジでは、グループ タスクは、それが作成された NCM コアで実行されますが、各子デバイス タスクはそのデバイスに適した NCM コアで実行されます。

コア バインディング トポロジの利点は次のとおりです。

- デバイスが複数の NCM コアから到達可能なトポロジの場合、パーティションとタスクの自動フェールオーバー。
- すべての NCM コアからアクセスできないデバイスは特定の NCM コアに割り当てることができます。
- NCM 管理者は特定のデバイスのタスクを実行する NCM コアを制御できます。この制御は、異種環境で重要になる場合があります。次に例を示します。
  - NCM サーバのキャパシティが異なる場合、大きい NCM コアにより多くのデバイスを割り当て、小さい NCM コアにより少ないデバイスを割り当てることができます。
  - 非常に大規模な設定のあるデバイスがネットワークに含まれている場合、NCM 管理者は NCM コア間にこれらのデバイスに対する役割を分散できます。

コア バインディング トポロジの制限は次のとおりです。

- 自動ロード シェアリングがありません。
- NCM 管理者はデバイスのパーティションを定義するために特別な NCM 設定作業を行う必要があります。
- パーティションとタスクのフェールバックは手動で行います。



NCM 1.8.01 および NCM 1.7.03 よりも前は、コア バインディング トポロジは「トポロジ 1」と呼ばれていました。

## コア バインディングのバリエーション

NCM は標準コア バインディング トポロジの次のバリエーションをサポートします。

- ユーザ インタラクション用のコア バインディング バリエーション

このバリエーションは、すべてのデバイス タスクと、NCM とユーザ間のすべての対話を切り離します。ユーザ インタラクションには、すべての操作可能な NCM セッション、NCM プロキシ セッション、NCM API を使用するツールまたはプログラム、統合製品からの接続などがあります。デバイス タスクを実行する各 NCM コアに 1 つまたは複数のパーティションが関連付けられます。ユーザ インタラクション用のみに指定された NCM コアにパーティションは関連付けられません。このバリエーションでは、各 NCM コアと、パーティションによって関連付けられたデバイス間の厳密な関係を維持します。図 4 (P.14) を参照してください。

このバリエーションの主な利点は、ユーザ インタラクション用のみに指定されたすべての NCM コアのリソースは、ユーザ要求に応答することだけに専念することです。さらに、このバリエーションには、標準コア バインディング トポロジと同じ利点と制限があります。



NCM 1.8.01 および NCM 1.7.03 よりも前は、コア バインディング トポロジのこのバリエーションは「トポロジ 4」と呼ばれていました。

- ローカル デバイス タスク用のコア バインディング バリエーション

このバリエーションでは、事前に計画されたデバイス タスク（たとえば、毎日のスナップショット）と、必要になった時点で作成されるローカル デバイス タスク（たとえば、アクセス コントロール リストの即時更新）を切り離します。事前に計画されたデバイス タスクを実行する各 NCM コアに 1 つまたは複数のパーティションが関連付けられます。ローカル デバイス タスクのみを実行する NCM コアにパーティションは関連付けられません。ローカル デバイス タスクの NCM コアは、パーティションの関連付けをバイパスして、この NCM コアで作成されたデバイス タスクが対象とするデバイスに直接接続します。図 5 (P.15) を参照してください。

このバリエーションの主な利点は、事前に計画されたタスクとは別に、ローカル デバイス タスクが実行されることです。この独立により、通常は、ローカル デバイス タスクの結果が迅速化されます。このバリエーションの制限は次のとおりです。

- 複数の NCM コアが同時に 1 つのデバイスにアクセスしようとする可能性があります。
- 事前に計画されたデバイス タスクは、ローカル タスクに予約されていない NCM コアで作成する必要があります。

さらに、このバリエーションには、標準コア バインディング トポロジと同じ利点と制限があります。



NCM 1.8.01 および NCM 1.7.03 よりも前は、コア バインディング トポロジのこのバリエーションは「トポロジ 3」と呼ばれていました。

## コア バインディングの図解

ここでは、3 つの NCM コアの水平スケーラビリティ環境のアーキテクチャ図の例を示します。内容は次のとおりです。

- 図 3：標準コア バインディング アーキテクチャの例
- 図 4：アーキテクチャ例：ユーザ インタラクション用のコア バインディング バリエーション



- 図 5 : アーキテクチャ例 : NCM ローカルデバイス タスク用のコア バインディング バリエーション

図 3 標準コア バインディング アーキテクチャの例

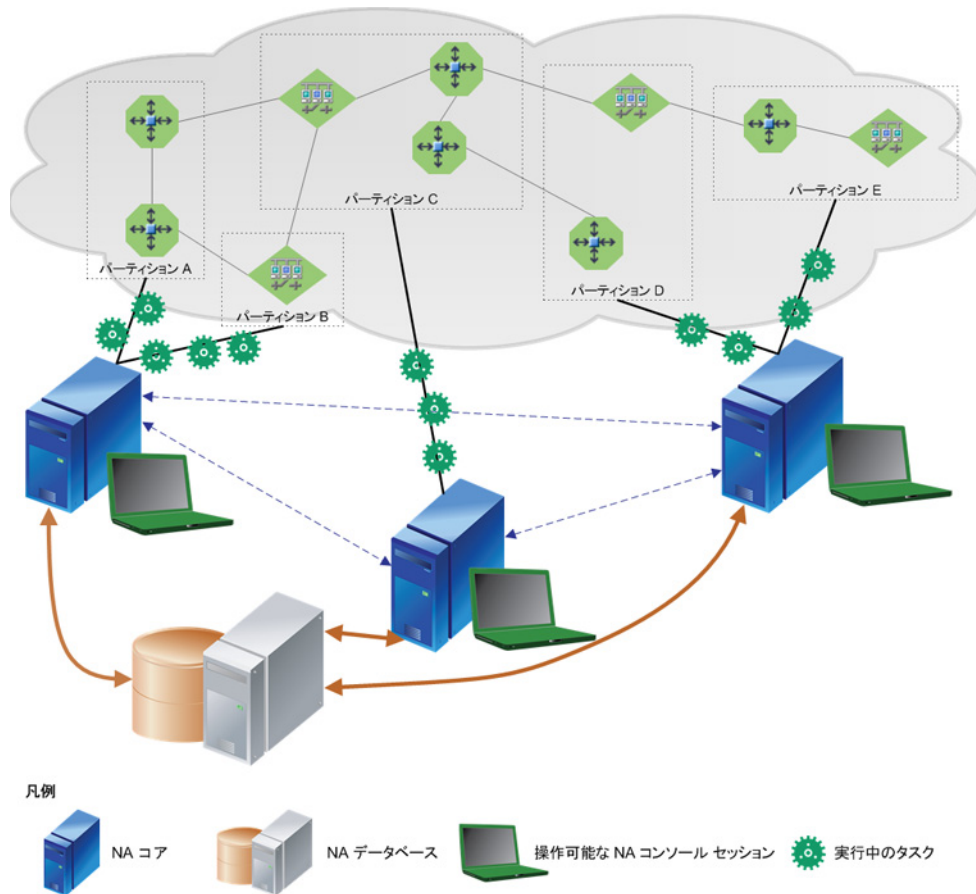


図 4 アーキテクチャ例：ユーザ インタラクション用のコア バインディング バリエーション

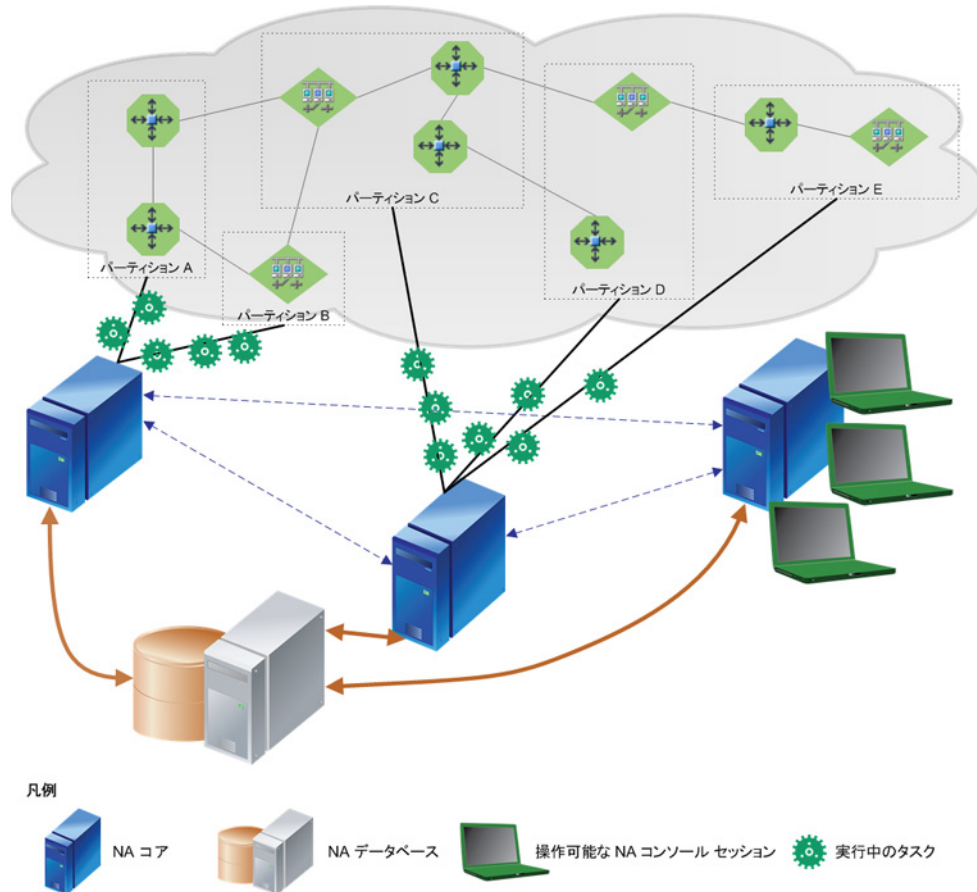


図 5 アーキテクチャ例：NCM ローカルデバイス タスク用のコア バインディング バリエーション

