

Sitecore CMS 6.6

拡張性ガイド

複数の Sitecore インスタンスを構成する管理者のためのガイド

目次

第 1 章	イントロダクション	4
第 2 章	標準的な複数インスタンス構成	6
2.1	標準的な複数インスタンス環境を構成する	7
2.2	データベース サーバーを構成する	8
2.3	コンテンツ管理環境を構成する	9
2.4	コンテンツ デリバリー環境を構成する	10
第 3 章	高度な構成	11
3.1	リモート パブリッシュ ターゲットを構成する	12
3.2	CM と CD 環境の分離	13
3.2.1	概要	13
3.2.2	セキュリティ同期に対応する	14
3.3	CM または CD 環境内のインスタンスの分離	16
3.4	ファイルの自動同期を構成する	18
3.4.1	Windows の DFS (分散ファイル システム)	19
3.4.2	Web 配置	19
3.5	セッション状態の管理	20
3.6	リンク データベース、ID テーブル、検索インデックスの同期	21
3.6.1	リンク データベース	21
3.6.2	ID テーブル	21
3.6.3	Lucene 検索インデックス	22
3.7	Sitecore DMS の拡張	23
3.7.1	インスタンス毎の DMS 設定	23
	デフォルト アナリティクス定義データベースの設定	23
	スケジュールの設定	23
3.8	SQL Server データベースのミラーリングおよびフェールオーバー	25
3.8.1	基本概念	25
3.8.2	SQL Server でのデータベース ミラーリングの設定	26
3.8.3	Sitecore でのデータベース ミラーリングの設定	27
	Web 設定で SQL Server Retryer を有効にする	27
	属性文字列の設定ファイルでミラーリングを有効にする	27
第 4 章	構成タスク	29
4.1	拡張性の設定の有効化の方法	30
4.2	ファイル ベースのメディアを無効化または有効化する方法	31
4.3	パブリッシュ インスタンスの名前を設定する方法	32
4.4	HTML キャッシュ クリアを構成する方法	33
4.5	マシン キーを構成する方法	35

4.6	Master データベースへの参照を削除する方法	36
	include ファイルを使用して Master データベースへの参照を削除する	36
	Master データベースへの参照を手動で削除する	36
4.7	クライアント インターフェースへのアクセスを制限する方法	38
4.8	リモート パブリッシュ ターゲットを構成する方法	39
	パブリッシュ ターゲット データベースを作成する	39
	パブリッシュ ターゲット データベースを登録する	39
4.9	SQL Server レプリケーションを構成する方法	40
4.10	セキュリティ キャッシュのクリアに対応する方法	41
4.11	DFS を構成する方法	43
4.12	Web 配置の構成方法	52
	複数の Web 配置ターゲットの構成方法	53

第 1 章

イントロダクション

この文書では Sitecore CMS のソリューションを、1 つまたは複数の環境 (コンテンツ管理: CM 環境、コンテンツ デリバリー: CD 環境など) において複数のインスタンスを構成することによって拡張を図る方法について説明します。また各種のインフラストラクチャまたはセキュリティ上の要件を満たすために代替の構成を行う方法についても説明します。

この文書の読者はコンテンツ エディターの知識を持ち、Sitecore のアプリケーションの体系の理解があることを前提としています。これには Sitecore のインストール、データベース、セキュリティの理解などが含まれます。また Microsoft Windows、IIS (インターネット インフォメーション サービス)、Microsoft SQL Server または Oracle の知識を持つことも前提としています。この文書で行う操作には Sitecore および Windows の管理者権限を必要とするものも含まれています。

この『拡張性ガイド』を読む前に、『Sitecore CMS 6.3 のイントロダクション』マニュアルを十分に読んでください。Sitecore CMS を複数インスタンス構成でインストールする前に、この『拡張性ガイド』の第 2 章を十分に読むことを推奨します。

このマニュアルの情報は、Sitecore CMS 6.5 および 6.6 に適用されます。このマニュアルの情報は、Sitecore Azure には適用されません。Sitecore Azure の詳細については、『Getting Started with Sitecore Azure』のマニュアルを参照してください。

この文書には次の章があります。

- **第 1 章 — イントロダクション**

この文書の紹介です。

- **第 2 章 — 標準的な複数インスタンス構成**

この章では Sitecore を複数インスタンス環境で構成する方法を説明します。

- **第 3 章 — 高度な構成**

この章ではいくつかの高度なタスクを実行する方法について説明します。これにはリモート パブリッシュ ターゲットを構成して環境を分離する方法などが含まれます。

- **第 4 章 — 構成タスク**

この章ではシステム管理者が実行する必要がある、その他の重要なタスクについて説明します。

第 2 章

標準的な複数インスタンス構成

この章では標準的な複数インスタンス環境を構成する方法を説明します。このソリューションではコンテンツ管理 (CM) サーバーとコンテンツ デリバリー (CD) サーバーのどちらも複数インスタンスを持ち、それらは物理的に同一の場所でホストされます。

1 つ以上のインスタンスで構成されるすべての Sitecore ソリューションに対し、ここで説明する構成手順を適用することが重要です。これらの手順は、Sitecore インスタンスを本番稼働に備えるために必要十分であり、キャッシュの自動同期や分散パブリッシュといった機能を使用できるようにします。

この章には次のセクションがあります。

- 標準的な複数インスタンス環境を構成する
- データベース サーバーを構成する
- コンテンツ管理環境を構成する
- コンテンツ デリバリー環境を構成する

2.1 標準的な複数インスタンス環境を構成する

この章ではコンテンツ管理 (CM) サーバーとコンテンツ デリバリー (CD) サーバーのどちらも複数インスタンスを持ち、物理的には同一場所でホストされている Sitecore にとって適切な構成について説明します。これは最もシンプルな構成であり、大部分の Sitecore のインストールにとって適切な構成です。

標準の単一インスタンスの Sitecore の構成では次の 3 つのデータベースが含まれます：

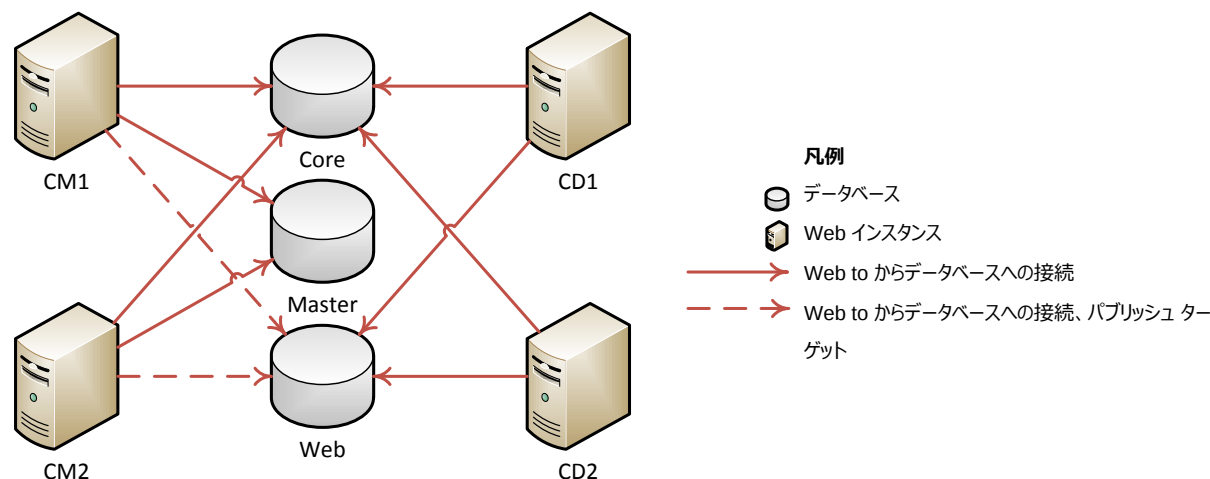
- Master データベース：すべてのアイテムのすべての言語のすべてのバージョンを含みます。これにはパブリッシュされる Web サイトのソースとなる、まだパブリッシュされていない作業中の内容も含まれます。
- Web データベース：デフォルトのパブリッシュ ターゲットです。各アイテムのパブリッシュされた言語のパブリッシュされたバージョンのみを含みます。
- Core データベース：Sitecore のユーザー インターフェイスの構成情報が含まれます。

標準的な複数インスタンスによる Sitecore のセットアップではこれらの 3 つのデータベースを使用し、すべての CM と CD インスタンス間で Core と Web データベースを共有し、すべての CM インスタンス間で Master データベースを共有します。

標準的な複数インスタンスによる Sitecore のセットアップでは CM インスタンスの 1 つをパブリッシュ CM インスタンスとする必要があります。

標準的な複数インスタンスによる Sitecore のセットアップでは、すべてのメディア アイテム データを Sitecore データベースに保存する必要があります。そのため、ファイル ベースのメディアを無効にする必要があります。

次のダイアグラムに、標準的な複数インスタンスによる Sitecore のセットアップにおける Web インスタンスと Sitecore データベースの接続を示します。



メモ

Sitecore DMS を使用する場合には、複数インスタンスのセットアップはさらに Analytics データベースを含みます。Analytics データベースは、CM 環境と CD 環境のすべてのインスタンスで共有します。Sitecore DMS の拡張性に関する詳細は、「Sitecore DMS の拡張」のセクションを参照してください。

2.2 データベース サーバーを構成する

Sitecore では、すべての Web サーバーと同じネットワーク上に位置する単一のデータベース サーバー、または複数のデータベース サーバーに Core、Master、Web の各データベースをインストールすることを推奨します。

Sitecore DMS を使用する場合には、Analytics データベースを専用のデータベース サーバーにインストールすることを推奨します。Sitecore DMS の拡張性に関する詳細は、「Sitecore DMS の拡張」のセクションを参照してください。

Sitecore データベースをインストールするためには、Sitecore のインストール プログラムを使用するか、または Sitecore パッケージを使用して手動でインストールすることが可能です。どちらも SDN から入手することができます。

Sitecore での SQL または Oracle データベースのインストールに関する詳細は、『Sitecore CMS インストール ガイド』を参照してください。

2.3 コンテンツ管理環境を構成する

CM 環境を構成するためには、1 つまたは複数の Web サーバー上に 1 つまたは複数の CM インスタンスをセットアップする必要があります。

Sitecore では環境内のすべての CMS インスタンスに専用の Web サーバーを使用することを推奨します。

CM 環境の Sitecore CMS のインスタンスを構成する手順:

1. サーバー上に Sitecore CMS Web アプリケーションをインストールします。

Sitecore CMS のインストールに関する詳細は、『Sitecore CMS インストール ガイド』を参照してください。

2. Web アプリケーションを構成してデータベース サーバーから Core、Master、Web の各データベースに接続します。

3. 拡張性の設定を有効化します。

拡張性の設定の有効化に関する詳細は、「拡張性の設定の有効化の方法」のセクションを参照してください。

4. ファイル ベースのメディアをすべて無効にします。

ファイル ベースのメディアの無効化に関する詳細は、「ファイル ベースのメディアを無効化または有効化する方法」のセクションを参照してください。

ファイル ベース メディアを無効化したくない場合に、ファイル ベース メディアで作業を行う他の方法について詳しくは、「ファイルの自動同期を構成する」のセクションを参照してください。

5. パブリッシュ インスタンスの名前を設定します。

パブリッシュ インスタンスの名前の設定に関する詳細は、「パブリッシュ インスタンスの名前を設定する方法」のセクションを参照してください。

6. HTML キャッシュ クリアを構成します。

HTML キャッシュ クリアの構成に関する詳細は、「HTML キャッシュ クリアを構成する方法」のセクションを参照してください。

7. マシンのキーのセクションを設定します。

マシンのキーのセクションの設定に関する詳細は、「マシン キーを構成する方法」のセクションを参照してください。

2.4 コンテンツ デリバリー環境を構成する

CD 環境を構成するためには、1 つまたは複数の Web サーバー上に 1 つまたは複数の CD インスタンスをセットアップする必要があります。

Sitecore ではすべての CMS インスタンスに専用の Web サーバーを使用することを推奨します。

CD 環境の Sitecore CMS のインスタンスを設定する手順:

1. サーバー上に Sitecore CMS Web アプリケーションをインストールします。

Sitecore CMS のインストールに関する詳細は、『Sitecore CMS インストール ガイド』を参照してください。

2. Web アプリケーションを構成してデータベース サーバーから Core と Web データベースに接続します。Master データベースには接続しません。

3. Master データベースへのすべての参照を削除します。

Master データベースへの参照の削除に関する詳細は、「Master データベースへの参照を削除する方法」のセクションを参照してください。

4. 拡張性の設定を有効化します。

拡張性の設定の有効化に関する詳細は、「拡張性の設定の有効化の方法」のセクションを参照してください。

5. ファイル ベースのメディアをすべて無効にします。

ファイル ベースのメディアの無効化に関する詳細は、「ファイル ベースのメディアを無効化または有効化する方法」のセクションを参照してください。

ファイル ベース メディアを無効化したくない場合に、ファイル ベース メディアで作業を行う他の方法について詳しくは、「ファイルの自動同期を構成する」のセクションを参照してください。

6. HTML キャッシュ クリアを構成します。

HTML キャッシュ クリアの構成に関する詳細は、「HTML キャッシュ クリアを構成する方法」のセクションを参照してください。

7. マシンのキーのセクションを設定します。

マシンのキーのセクションの設定に関する詳細は、「マシン キーを構成する方法」のセクションを参照してください。

8. Sitecore CMS のクライアント インターフェイスへのアクセスを制限します。

Sitecore CMS クライアント インターフェイスへのアクセスの制限に関する詳細は、「クライアント インターフェイスへのアクセスを制限する方法」のセクションを参照してください。

第 3 章

高度な構成

この章では各種のインフラストラクチャまたはセキュリティ上の要件を満たすために行う必要のある、標準的な複数インスタンスの Sitecore CMS のセットアップと異なる点について説明します。各種の追加要件をもつソリューションの構成のために必要な操作を解説します。

この章には次のセクションがあります。

- リモート パブリッシュ ターゲットを構成する
- CM と CD 環境の分離
- CM または CD 環境内のインスタンスの分離
- ファイルの自動同期を構成する
- セッション状態の管理
- リンク データベース、ID テーブル、検索インデックスの同期
- Sitecore DMS の拡張
- SQL Server データベースのミラーリングおよびフェールオーバー

3.1 リモート パブリッシュ ターゲットを構成する

標準的な Sitecore CMS の複数インスタンス構成は 3 つのデータベースを含みます。Core、Master、Web の各データベースです。Web データベースは CM と CD 環境で Web サイトのレンダリングに使用する、デフォルトのパブリッシュターゲットです。

通常よく行われるもう 1 つの方法は、リモート パブリッシュ ターゲットを構成して、それを CD インスタンスで Web サイトのレンダリングに使用することです。これによって、CD 環境に影響を及ぼすことなく、CM 環境のデフォルトの Web ターゲットを使って CM 環境の Web サイトのコンテンツをパブリッシュおよびプレビューすることが可能となります。

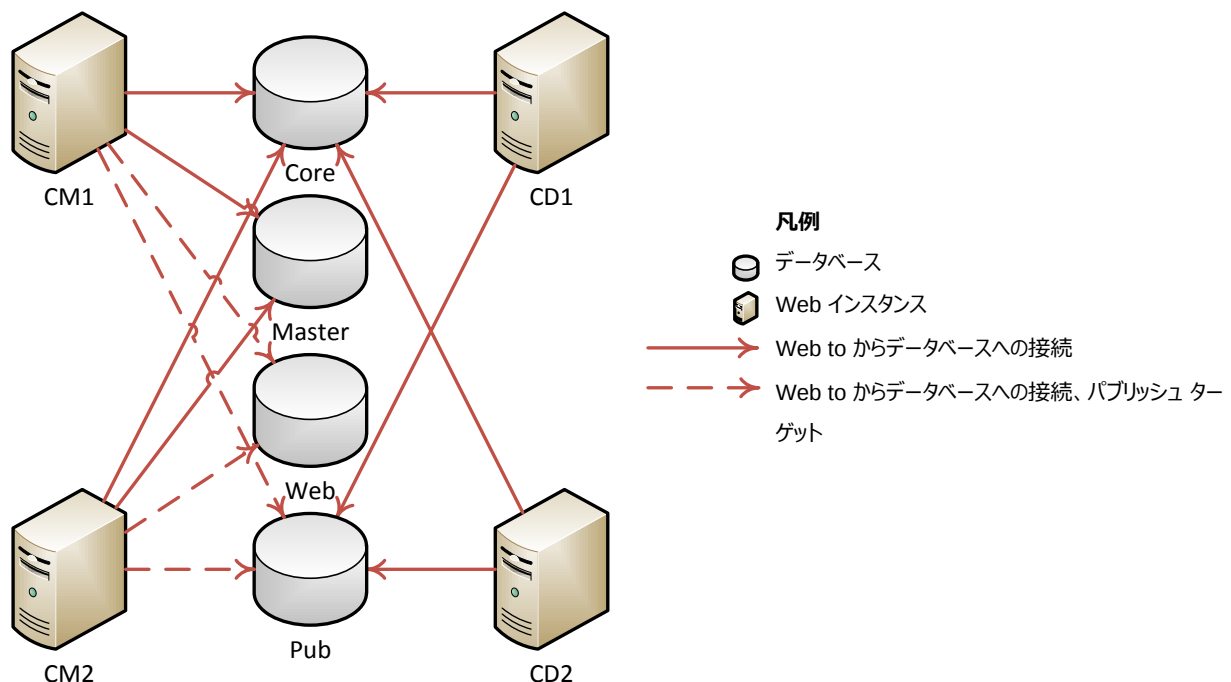
このような複数インスタンス構成の Sitecore CMS では 4 つのデータベースを含みます。Core、Master、Web、Pub (CD Web) です。

CM インスタンスは 4 つのデータベースすべてに接続し、2 つのパブリッシュ ターゲットを持ち、それらは Web と Pub にリンクします。

CD インスタンスは Core と Pub データベースに接続します。

リモート パブリッシュ ターゲットの設定に関する詳細は、「リモート パブリッシュ ターゲットを構成する方法」のセクションを参照してください。

次のダイアグラムに、リモート パブリッシュ ターゲットを使用した標準的な複数インスタンスによる Sitecore のセットアップにおける Web インスタンスと Sitecore データベースの接続を示します。



3.2 CM と CD 環境の分離

セキュリティまたはインフラストラクチャ上の要件により、CM と CD 環境を分離する必要がある場合があります。またこのセクションの情報は、CM と CD 環境を地理的に分散する場合の構成にも適用することができます。

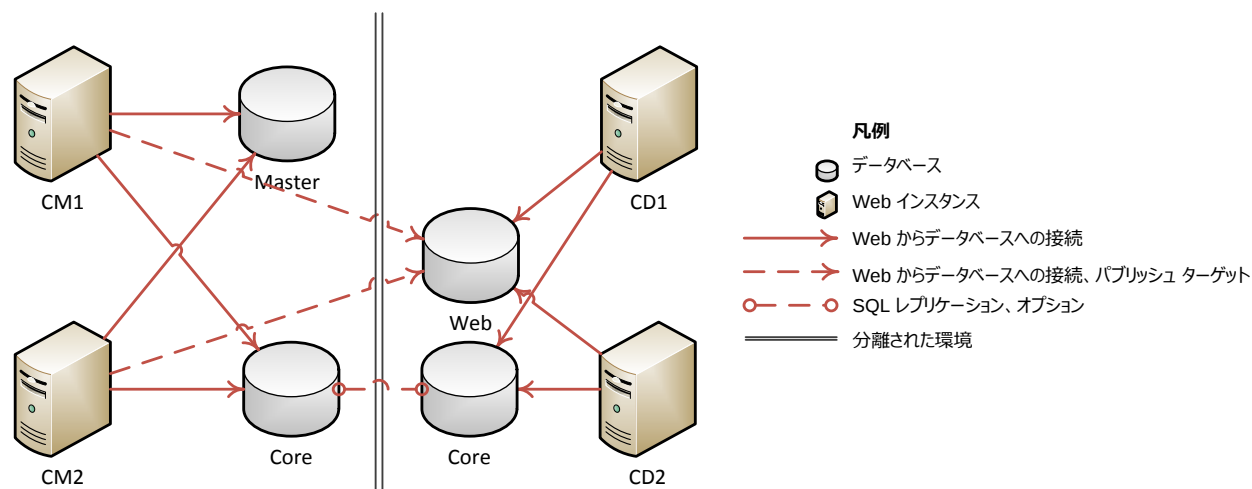
3.2.1 概要

分離された CM と CD の構成は通常、下記を含みます：

- 2 つの個別のデータベース サーバー。1 つが CM 環境用、もう 1 つが CD 環境用となる。
- CM データベース サーバー上に Core と Master データベース。
- CD データベース サーバー上に Core と Web データベース。
- CM データベースと CD データベース サーバー上の Web データベースに接続された 1 つまたは複数の CM Web インスタンス。
- CD データベースに接続された 1 つまたは複数の CD Web インスタンス。

各 CM Web インスタンスと各 CD Web インスタンスを「第 2 章」の指示に従って構成します。

次のダイアグラムに、CM と CD の環境を分離した Sitecore のセットアップにおける Web インスタンスと Sitecore データベースの接続を示します。



メモ

「リモート パブリッシュ ターゲットを構成する方法」のセクションの手順に従ってこのセットアップを構成し、Web データベースを CM 環境に置き、Pub データベースを CD 環境に置くことが可能です。

3.2.2 セキュリティ同期に対応する

CM と CD の分離環境では、セキュリティ データとキャッシュは、CM 環境のすべてのインスタンスにおいて、また CD 環境のすべてのインスタンスにおいて、自動的に同期されます。これは、環境内のすべてのインスタンスは同じデータベースを共有し、同じセキュリティ データとシステム イベント キューを共有しているためです。しかし、分離された CM と CD の環境間ではセキュリティ データとキャッシュは自動的に同期されません。

これには Sitecore アプリケーションのために構成された .NET セキュリティ プロバイダーに保管される、メンバーシップ、ロール、プロファイル情報に関するセキュリティ データのみが含まれます。これには、Sitecore のコンテンツ データベースに保管されたコンテンツ アイテムに対してユーザーに割り当てられるセキュリティ アクセス許可は含まれません。Sitecore のこのセキュリティ情報はコンテンツ データベースが環境間で共有されている場合にはパブリッシュの過程で自動的に同期されます。

システム イベント キューに関する詳細は、次の URL から『Sitecore CMS 6.3 のイントロダクション』を参照してください。

メモ

.NET セキュリティ プロバイダーが Sitecore に実装および設定される方法の詳細については、『下位レベルの Sitecore セキュリティとカスタム プロバイダー』のマニュアルを参照してください。

実装の要件によっては、CM と CD の分離環境におけるセキュリティ同期に対応するための別の方法もあります。

- 実装するソリューションがセキュリティ データとキャッシュの同期を必要としない場合には、このセクションに記載する変更を行う必要はありません。たとえば CD インスタンスがエクストラネット セキュリティを使用しない場合には、同期の必要はありません。
- 実装するソリューションが SQL のデフォルトのメンバーシップ、ロール、プロファイル プロバイダー、またはその他のセキュリティ プロバイダー (Active Directory、CRM、またはカスタムのプロバイダー) のために CM と CD 環境のセキュリティ データとキャッシュのリアルタイムでの同期を必要とする場合には、データベース レプリケーションを構成して CM の Core データベースと CD の Core データベースの同期を取ることを推奨します。

これによって 2 つのデータベースに保存されたセキュリティ アカウント データ、およびセキュリティ キャッシュの同期に必要な関連するセキュリティ イベントを、環境間でレプリケーションすることが可能です。

データベース レプリケーションの構成に関する詳細は、「SQL Server レプリケーションを構成する方法」のセクションを参照してください。

- 実装するソリューションが SQL のデフォルトのメンバーシップ、ロール、プロファイル プロバイダーのためには CM と CD 環境間のセキュリティ データのリアルタイムでの同期を必要としないが、その他のセキュリティ プロバイダー (Active Directory、CRM、またはカスタムのプロバイダー) のためにセキュリティ キャッシュの同期を必要とする場合には、Core データベースのレプリケーションによらずとも、セキュリティ キャッシュのクリアを行うことにより対応が可能です。

セキュリティ キャッシュのクリアへの対応に関する詳細は、「セキュリティ キャッシュのクリアに対応する方法」のセクションを参照してください。

メモ

SQL セキュリティ プロバイダーのデータは環境間で共有されているが、システム イベント キューを保存している Core データベースは共有されていない場合にも、後者の方法を使ってセキュリティ キャッシュへの対応を行うことが可能です。このためには Core データベースからセキュリティ関連のテーブルを専用のデータベースに抽出し、SQL のすべてのセキュリティ プロバイダーを再構成する必要があります。これは地理的に分散した環境ではパフォーマンスに悪影響を及ぼす場合があります。この方法については、この文書では記載していません。

Sitecore CMS のセキュリティと他のキャッシュの実装と設定の詳細については、『キャッシュ設定ガイド』のマニュアルを参照してください。

3.3 CM または CD 環境内のインスタンスの分離

CM と CD 環境の分離に加えて、インフラストラクチャ上の要件により、同じ環境内のインスタンスを分離する必要がある場合があります。これは、通常は同一の環境内であるが地理的に分散した Sitecore のインスタンスを構成する場合に当てはまります。

CM または CD 環境内でインスタンスを地理的に 2 箇所またはそれ以上に分散させるためには、「CM と CD 環境の分離」セクションで説明した構成の手順を実施しますが、次の点が異なります。

- 環境内のすべての Sitecore データベースは各場所にコピーを持つようにします。
- 環境内の各場所のすべての Web インスタンスは Sitecore データベースのローカル コピー（および CM 環境では最も近い Web データベース）に接続するようにします。
- CM 環境内のすべての Sitecore データベースは、環境内のすべてのコピーに対してデータベース レプリケーションを構成します。
- CD 環境内のすべての Sitecore データベースには、環境内のすべてのコピーに対してデータベース レプリケーションをオプションで構成します。

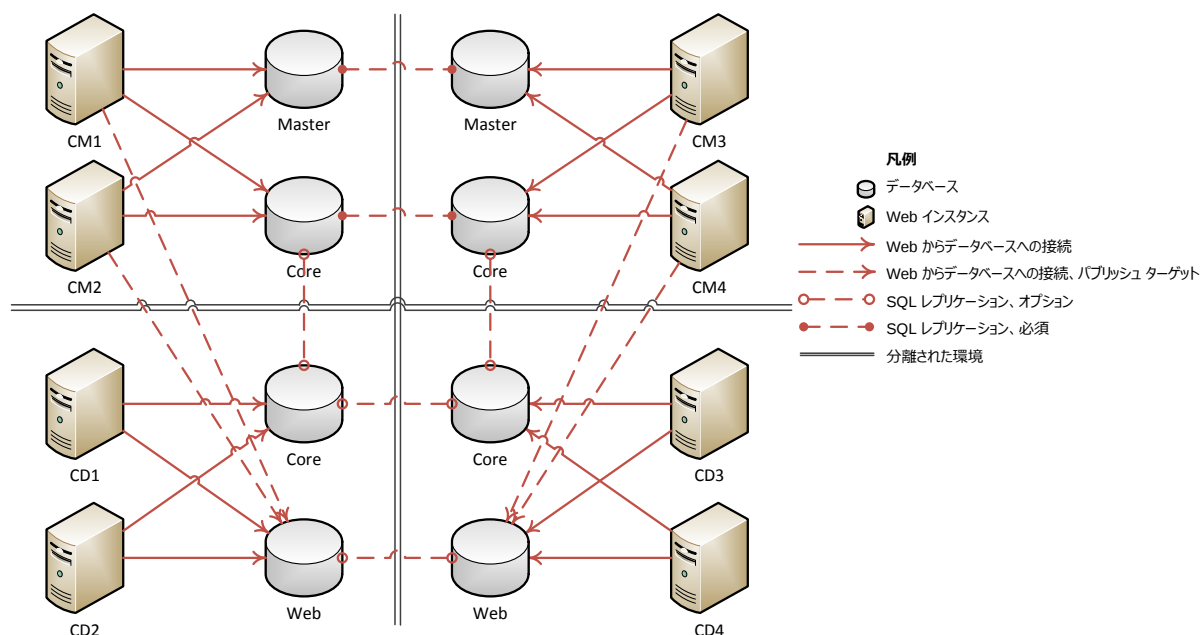
メモ

各 CD Web データベースが、個別のパブリッシュ ターゲットとして構成された場合は、CD 環境内で Web データベース レプリケーションを構成する必要はありません。

リモート パブリッシュ ターゲットを設定する方法の詳細については、「セキュリティ同期に対応する」のセクションを参照してください。

データベース レプリケーションの構成に関する詳細については、「SQL Server レプリケーションを構成する方法」のセクションを参照してください。

次のダイアグラムに、CM と CD の環境内に分離したインスタンスをもつ Sitecore のセットアップにおける Web インスタンスと Sitecore データベースの接続を示します。



メモ

「リモート パブリッシュ ターゲットを構成する方法」のセクションの指示に従って構成を行い、Web データベースを CM 環境に置き、Pub データベースを CD 環境に置くことが可能です。

3.4 ファイルの自動同期を構成する

複数インスタンスの Sitecore ソリューションで、Sitecore メディア アイテムのメディア データを保存するためにファイル システムを使用する場合、または Sitecore ドメイン構成を変更する場合には、ファイルとフォルダーの同期が必要です。

複数の Sitecore インスタンス間でファイル システムの同期を構成するためには、2 つの方法があります。

- Windows の DFS (分散ファイル システム)
- Web 配置

Sitecore は Sitecore メディア アイテムに対応するメディア ファイルとフォルダー (/App_Data/MediaFiles フォルダー) の同期を行うため、またセキュリティ ドメイン構成ファイル (/App_Config/Security フォルダー) の同期を行うために、これらの方法をサポートします。

注意

Sitecore は上述以外のフォルダーのファイルを同期するためには、これらの方法をサポートしません。これは、ASP.NET アプリケーションの性質上の理由に加え、DFS と Web 配置の実装によって、ファイル システム上の同じフォルダーを同期するよう両方のユーティリティを構成した場合に、競合が生じるためです。

ここで指定された以外のフォルダーとファイルの同期は、通常のコンテンツ オーサリング プロセスの一環ではありません。これらのフォルダーとファイルは、Sitecore サイトの新しいビルドと同じ方法で配置することを推奨します。たとえば、Sitecore Packager や、その他の配置ツールを使用して手動で配置するなどです。

Sitecore の複数インスタンス構成では、DFS と Web 配置は異なる目的と異なる制限を持ちます。

- DFS は同一 Windows Server ドメイン内の Web サーバー間のファイルの同期のみを、双方向で行います。
- Web 配置は異なるドメインのサーバー間の変更を同期することが可能ですが、一定方向の同期のみをサポートします。

下記の使用を推奨します:

- DFS を使用して CM 環境内のファイル システムを同期する。
- Web 配置を使用してメディア ファイルの変更を CD 環境内の配置ターゲットにレプリケーションする。
- DFS を使用して CD 環境内で配置された変更を他の CD インスタンスにレプリケーションする。

3.4.1 Windows の DFS (分散ファイル システム)

ファイル メディアを使用するが、インスタンス間でメディア ファイルのためにファイル システムを共有しない場合には、/App_Config/Include/ScalabilitySettings.config または web.config ファイルの Media.FileFolder 設定で指定されたサブディレクトリを環境内の異なるインスタンス間で同期するように DFS を構成することができます。

DFS はサーバーのグループの間でファイル システムを同期します。DFS を使用して、CM または CD 環境内の Sitecore のインスタンス間で、ファイル システムを同期することが可能です。DFS は、ネットワーク ドライブや他の共有リソースがない場合でも、ローカルのディスク ドライブのサブディレクトリで機能します。

セキュリティ ドメインを追加、削除、更新した場合に、セキュリティ ドメイン構成ファイルを含むディレクトリを同期するように、DFS を構成することができます。

DFS を構成する場合は、環境ごとに DFS レプリケーション グループを構成する必要があります。通常は、CM 環境に 1 つ、CD 環境に 1 つの DFS レプリケーション グループがあります。環境にさらに CD または CM インスタンスを追加した場合には、そのインスタンスを適切な DFS レプリケーション グループに追加する必要があります。

DFS の構成に関する詳細は、「DFS を構成する方法」のセクションを参照してください。

3.4.2 Web 配置

Microsoft Web 配置ツールを使ってファイル システムの変更をソース インスタンスから 1 つまたは複数のターゲット インスタンスに転送するように Sitecore を構成することができます。具体的には、ファイル メディアの CM 環境から CD 環境への Web 配置を構成することが可能です。

Sitecore は Web 配置ツールを使って、指定されたソース サブディレクトリおよびファイルから、ターゲット ファイル システムに、すべての追加、変更、および削除を配置します。レプリケーション グループ内のすべてのサーバーで双方向にファイルのレプリケーションを行う DFS と異なり、Web 配置は一方方向のみに作用します。したがって Web 配置は、CM のパブリッシュ インスタンスから CD のターゲット インスタンスに対する変更の配置のみを行います。ターゲットからソースへの配置は行われません。

注意

/App_Config/Include/ScalabilitySettings.config ファイルまたは web.config ファイルの Media.FileFolder 設定で指定されたサブディレクトリ以外のファイルまたはサブディレクトリに対して、Web 配置を構成しないでください。

ネットワークの帯域幅の使用を最小化するため、Web 配置ツールはファイル システムを比較し、差分のみを配置します。

Web 配置の構成に関する詳細は、「Web 配置の構成方法」のセクションを参照してください。

3.5 セッション状態の管理

Sitecore CMS のユーザー インターフェースは ASP.NET のインプロセス セッション管理を必要とします。CM インスタンスでは、web.config ファイルの /configuration/system.web/sessionState 要素の mode 属性の値が InProc である必要があります。

インプロセス セッション管理の要件により、サーバー アフィニティ (スティッキー セッションとも呼ばれます) のための CM ロード バランサーを構成する必要があります。CD 環境の mode 属性に他の値を使用することもできます。

3.6 リンク データベース、ID テーブル、検索インデックスの同期

Sitecore の複数インスタンス構成で次の機能を実装した場合、いくつかの Sitecore CMS の機能を構成する必要があります。

3.6.1 リンク データベース

Sitecore リンク データベース API を使用すると、特定の Sitecore アイテムに参照されている、または特定の Sitecore アイテムを参照している、すべてのアイテムにアクセスすることができます。

リンク データベースのすべての情報はデフォルトでは Core データベースに保存されており、すべての Sitecore コンテンツ データベースのアイテムのデータを集計しています。

リンク データベースの場所は web.config ファイルの Link Database セクションの connectionStringName パラメーターで設定します：

```
<!-- LINK DATABASE -->
<LinkDatabase type="Sitecore.Data.${(database)}.${(database)}LinkDatabase, Sitecore.Kernel">
  <param connectionStringName="core" />
</LinkDatabase>
```

Sitecore の単一サーバー構成では、リンク データベースは自動的に更新されます。

Sitecore の複数インスタンス構成では、リンク データベースは CM 環境から参照されるデータベースのみが自動的に更新されます。

CD 環境でリンク データベースが自動的に更新されるようにするための方法：

- CD と CM インスタンスは、パブリッシュ ターゲット データベースを指定するために環境全体で同じ名前を使用する必要があります (通常は Web)。
- 次の条件のうちの 1 つが満たされる必要があります：
 - Core データベースが CM と CD インスタンス間で共有またはレプリケーションされる。
 - CM と CD パブリッシュ ターゲット データベースが共有するデータベース (通常は Web) にリンク データベースのデータを保存するように構成する。

3.6.2 ID テーブル

Sitecore ID への任意のキーの永続的なマッピングを Sitecore ID テーブルに保存することができます。

これは、デフォルトでは Sitecore CMS WebDAV 機能が CM 環境でメディア アイテムのロックを保持するためにのみに使用し、CD 環境と同期する必要はありません。

ID テーブルの場所は web.config ファイルの IDTable セクションの connectionStringName パラメーターで設定します:

```
<IDTable type="Sitecore.Data.$(database).$(database)IDTable, Sitecore.Kernel" singleInstance="true">
  <param connectionStringName="master" />
  <param desc="cacheSize">500KB</param>
</IDTable>
```

ID テーブルのコンテンツを CM と CD 環境間で同期するには、ID テーブルの構成セクションの connectionStringName パラメーターを変更して、環境間で共有またはレプリケーションしている Core データベースを指定するようにします。または、共有の Web データベースを指定することも可能です。

3.6.3 Lucene 検索インデックス

Sitecore はデフォルトで、Master と Core データベースのアイテムのための Lucene 検索インデックスを CM 環境で、Core データベースのアイテムのための Lucene 検索インデックスを CD 環境で、自動的に管理します。Sitecore は Lucene インデックスのローカル コピーを各インスタンスのファイル システム上に保持し、インスタンス間でのインデックス共有をサポートしません。

Web データベースの Lucene インデックスを構成する場合は、CM 環境と CD 環境のすべての Sitecore インスタンスの Web データベースで History Engine を有効化し、インデックスが自動的に更新され同期されるようにします。

Sitecore データベースで History Engine を有効化する方法:

web.config ファイルで、/configuration/sitecore/databases/database 要素の id がデータベース名に等しい箇所に次のセクションを追加します:

```
<Engines.HistoryEngine.Storage>
  <obj type="Sitecore.Data.$(database).$(database)HistoryStorage, Sitecore.Kernel">
    <param connectionStringName="$(id)" />
    <EntryLifeTime>30.00:00:00</EntryLifeTime>
  </obj>
</Engines.HistoryEngine.Storage>
<Engines.HistoryEngine.SaveDotNetCallStack>>false</Engines.HistoryEngine.SaveDotNetCallStack>
```

Sitecore のアイテムが変更されると、変更の発生した Sitecore インスタンスで Lucene インデックスが直ちに更新されます。

複数サーバー環境のリモート サーバーでは、Lucene インデックスはアイテムの変更後、直ちには更新されません。Lucene インデックスは web.config ファイルの Indexing.UpdateInterval 設定で指定された間隔と、Indexing.UpdateJobThrottle 設定で指定された更新間隔の最短待ち時間に従って、自動的に更新されます。

Lucene 検索インデックスの統合と構成に関する詳細は、Sitecore の検索とインデックスに関する文書を SDN から参照してください。

3.7 Sitecore DMS の拡張

Sitecore DMS および CMS のサーバー要件、インストール、構成に関する情報および推奨事項は、次の文書から参照できます：

- 『SQL Server 上の DMS 用のインストール ガイド』
- Engagement Analytics 設定リファレンス』
- 『DMS パフォーマンス チューニング ガイド』
- 『CMS パフォーマンス チューニング ガイド』

Sitecore DMS を Sitecore の複数インスタンス環境で構成する場合には、下記を推奨します：

- パフォーマンス上の理由から、Analytics データベースを専用のデータベース サーバーにインストールします。
- DMS の統計を収集してレポートするために個別のデータベースを使用する場合は、これらを専用のサーバーにインストールする必要があります。
- すべての Sitecore CMS インスタンスで、同じ Analytics データベースを共有します。

3.7.1 インスタンス毎の DMS 設定

デフォルト アナリティクス定義データベースの設定

アナリティクス定義データベースは、Web サイト用に設定した DMS 設定のすべてが含まれるデータベースです。

アナリティクス定義データのストアに使用されるデフォルト データベースは、Master データベースです。ただし、本番環境では、これは、通常、Master データベースではありません。したがって、設定ファイルが正しいデータベース（通常、Web データベース）を参照していることを確認する必要があります。

デフォルト アナリティクス定義データベースを設定する方法：

各 CD インスタンスで、/App_Config/Include/Sitecore.Analytics.config ファイルの /configuration/sitecore/settings ノード内に、次の設定を追加します：

```
<setting name="Analytics.DefaultDefinitionDatabase" value="web" />
```

スケジュールの設定

DMS は、レポートを電子メール配信することができます。環境をスケール変更するときに、送信するインスタンスを設定することができます。通常、レポートを送信する Sitecore CM インスタンスが 1 つあります。

特定の Sitecore インスタンスによってレポートが電子メールで送信されないようにするには、Analytics.config ファイルで以下のストリングをコメント化します。

```
<scheduling>
  <agent type="Sitecore.Analytics.Tasks.EmailReportsTask, Sitecore.Analytics" method="Run" interval="1:00:00">
    <DatabaseName>master</DatabaseName>
  </agent>
  <agent type="Sitecore.Analytics.Tasks.SubscriptionTask, Sitecore.Analytics" method="Run" interval="00:15:00">
    <Minutes>15</Minutes>
  </agent>
</scheduling>
```

Sitecore.Analytics.Tasks.EmailReportsTask エージェントはレポートを送信します。こちらについての詳細は、「Sitecore Analytics Configuration Reference」を参照してください。このエージェントを CM インスタンスでは使用しない場合は、DatabaseName に指定してください。

Sitecore.Analytics.Tasks.SubscriptionTask エージェントは、レポートを登録ユーザーに送信します。こちらについての詳細は、「Marketing Operations Cookbook」を参照してください。

3.8 SQL Server データベースのミラーリングおよびフェールオーバー

Sitecore データベースの可用性を向上させるには、SQL Server データベースのミラーリングとフェールオーバーを実装します。また災害時にも、自動フェールオーバーを利用して、データベースのスタンバイ コピーをすばやくオンラインに戻すことができます。

3.8.1 基本概念

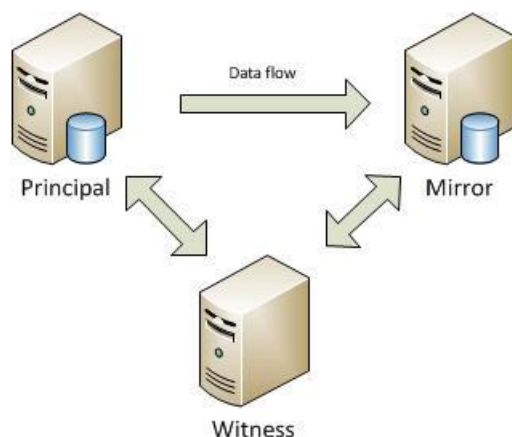
SQL Server データベース ミラーリングの要件を以下に示します。

- SQL Server 2008 以降

データベース ミラーリングと共に自動フェールオーバーを設定するには、少なくとも 3 つのデータベース サーバー インスタンスが必要です。

- プリンシパル – プライマリー データベース サーバー (完全復旧モデルに設定されている必要があります)。
- ミラー – プリンシパル データベースの完全なバックアップ コピー。
- ミラーリング監視 – 他の 2 つのデータベース サーバー インスタンスを監視 (このサーバー インスタンスにはデータベースはインストールされていません)。

プリンシパル インスタンス、ミラー インスタンス、ミラーリング監視インスタンスには SQL Server Standard エディションまたは Enterprise エディションを使用してください。プリンシパル インスタンスとミラー インスタンスでは、同じバージョンの SQL Server (およびホットフィックス) がインストールされている必要があります。



データベース ミラーリングによって、トランザクション ログ レコードが 1 つのサーバーから別のサーバー (プリンシパル インスタンスとミラー インスタンスの間) に転送され、スタンバイ サーバーにすばやくフェールオーバーすることができます。

Sitecore に変更を加えると、その変更はまずプリンシパル データベースのトランザクション ログに記録されます。それと同時にミラーのトランザクション ログも更新されます。つまり、変更が同時にプリンシパル データベースとミラー データベースに書き込まれるため、ミラーは常にプリンシパル データベースと同期が保たれていることになります。

ミラーリング監視サーバー インスタンスにより、問題の検出のために他のデータベース インスタンスが定期的に監視されます。プリンシパル サーバーが利用不可能になったという問題が検出された場合は、フェールオーバーが有効になり、プリンシパル ロールがミラーリング サーバーに切り替えられます。

SQL Server データベースのミラーリングにより、最小限のデータ ロスですばやくフェールオーバーが行われます。特殊なハードウェアは必要なく、すばやく設定することができ、管理も簡単です。

基本概念の詳細や SQL Server データベースのミラーリングの設定手順については、MSDN の記事「データベース ミラーリング (SQL Server)」を参照してください。

3.8.2 SQL Server でのデータベース ミラーリングの設定

このセクションでは、SQL Server データベースのミラーリングを実装するときの考慮事項と必要な手順の概要を説明します。詳細や実行する手順の全リストについては、Microsoft オンライン ブックまたは MSDN の Web サイトを参照してください。

データベース ミラーリングに関する注意の必要な重要事項:

- プリンシパル データベースは「完全」復旧モデルである必要があります。デフォルトでは、Sitecore データベースでは復旧モデルに [単純] が使用されています。ミラーリングするには、復旧モデルを [完全] に変更する必要があります。
- ミラー データベースは、NORECOVERY のプリンシパル データベースの復元から始めて、続いてプリンシパル トランザクション ログ バックアップを復元する必要があります。
- プリンシパル データベースおよびミラー データベースでは、同じバージョンの SQL Server を実行している必要があります。
- ミラー サーバーにミラー データベース用の十分なディスク領域があるか確認してください。
- ミラー データベースは復旧状態になっているため、直接アクセスすることはできません。ミラーのデータベース スナップショットを作成すると、その時点のミラー データベースを間接的に読み取ることができます。

手順の要約:

1. データベースのバックアップを作成し、対象のミラー サーバーに復旧なしで復元します。
2. すべてのデータベース サーバー インスタンス間の信頼関係を作成します。
3. エンドポイントを設定します。

エンドポイントとは、SQL Server データベースへのエントリ ポイントです。ファイアウォールを使用しているのと同じくらい、エンドポイントを使用します。エンドポイントによって、トラフィックが認証されインターネット上でデータベースに接続することが可能になります。SQL Server では、ウィザードを使用して各データベース インスタンスのエンドポイントを設定することができます。

4. パートナーとミラーリング監視のサーバー インスタンスを指定します。

パートナーを指定すると、直ちにミラーリングが開始されます。

各手順の実行方法の詳細については、MSDN の「データベース ミラーリングの設定 (SQL Server)」を参照してください。

3.8.3 Sitecore でのデータベース ミラーリングの設定

Sitecore で SQL Server のデータベース ミラーリングを設定する場合、SQL Server インスタンスでほとんどの設定手順を実行します。ただし、Sitecore と SQL Server が相互でまたはミラー データベースと正常に通信が行われていることを確認するには、Sitecore でいくつかの手順を実行する必要があります。

Web 設定で SQL Server Retryer を有効にする

Web 設定ファイルの RETRYER セクションは、ミラー データベースへの要求を処理します。Sitecore をインストールすると、デフォルトでこのセクションが web.config ファイルに含まれます。

```
<!-- RETRYER
    When enabled, the Retryer retries database requests a specified number of times (disabled by default).

    For example, this is useful if you configure an instance to support hot failover for database calls.
-->
<retryer disabled="true" type="Sitecore.Data.DataProviders.Retryer, Sitecore.Kernel">
  <param desc="Number of tries">6</param>
  <param desc="Interval between tries">00:00:00.500</param>
  <param desc="Log each exception (should be used for debug only)">true</param>
</retryer>
```

データベース ミラーリングを開始するには、RETRYER セクションを有効にする必要があります（デフォルトでは無効になっています）。

データベース ミラーリングを有効にするには、「disabled」の設定を「false」に変更する必要があります。

```
<retryer disabled="false" type="Sitecore.Data.DataProviders.Retryer, Sitecore.Kernel">
```

変更を web.config ファイルに保存します。

メモ

パフォーマンスをより良くするために、RETRYER セクションで他のパラメーターの編集が必要なことがあります。ただし、設定を変更する場合は十分に注意してください。最高のパフォーマンスを得るには、すべてのデフォルト パラメーターを変更しないことをお勧めします。

属性文字列の設定ファイルでミラーリングを有効にする

ミラーリングを有効にして Sitecore ソリューションでフェールオーバーを行うには、ミラーとして使用するデータベースへの参照を追加して属性文字列の設定ファイルを更新する必要があります。更新を行うには、「Failover Partner」パラメーターを使用してください。

Failover Partner=<Name of failover Mirror instance>

Master データベースの属性文字列の例：

```
<add name="master" connectionString="user id=sa;password=*****;Data Source=10.38.10.224;Failover Partner=10.38.10.213;Database=Sitecore_Master"/>
```

ミラーリングする各データベースの属性文字列にこのパラメーターを追加します。標準的な Sitecore ソリューションには、「Master」、「Core」、「Web」の 3 種類のデータベースがあります。各データベースにフェールオーバーのパラメーターを追加してください。

ミラーリングのフェールオーバーにかかる時間は 5 秒～20 秒が目安です。.NET SQL プロバイダーの持つリトライ機能により可用性の高い運用が可能となります。遅くとも .NET プロバイダーがハンドリングできる 20 秒以内にフェールオーバーが完了することが求められます。フェールオーバーに時間を要してフェールオーバー パートナーのデータベースが準備できていない場合やフェールオーバー パートナー自体にアクセスできない場合は、.NET SQL プロバイダーは旧プリンシパル（ダウンしたサーバー）と旧ミラー（フェールオーバーによりプリンシパルになる予定のサーバー）のどちらのデータベースにもアクセスが行えないためデータベース アクセス エラーとなります。

サイトコア 6.6 では .NET SQL プロバイダーからデータベース アクセス エラーが返ってきた場合にデータベース アクセスをリトライする RETRYER 機能を実装しており、RETRYER を有効に設定することで .NET SQL プロバイダーの待機時間を超えてエラーとなった場合でもデータベース アクセスを再試行するため可用性が向上します。

しかしながら、絶えず RETRYER による再試行が動作する環境は適切ではありません。5 ～ 20 秒以内にフェールオーバーが完了するように SQL サーバー環境の設定を見直す必要があります。

第 4 章

構成タスク

この章では、複数インスタンス環境を構成するために行う必要のあるタスクについて説明します。

この章には次のセクションがあります。

- 拡張性の設定の有効化の方法
- ファイル ベースのメディアを無効化または有効化する方法
- パブリッシュ インスタンスの名前を設定する方法
- HTML キャッシュ クリアを構成する方法
- マシン キーを構成する方法
- Master データベースへの参照を削除する方法
- クライアント インターフェースへのアクセスを制限する方法
- リモート パブリッシュ ターゲットを構成する方法
- SQL Server レプリケーションを構成する方法
- セキュリティ キャッシュのクリアに対応する方法
- DFS を構成する方法
- Web 配置の構成方法

4.1 拡張性の設定の有効化の方法

Sitecore CMS アプリケーションの拡張性の設定を有効化する手順:

- `/App_Config/Include/ScalabilitySettings.config.example` ファイルの名前を `/App_Config/Include/ScalabilitySettings.config` に変更します (`.example` 拡張子を削除します)。

メモ

`/App_Config/Include/ScalabilitySettings.config` ファイル、またはその他の `/App_Config/Include/*.config` ファイルの値は、`web.config` の値を上書きします。

また、`web.config` ファイルを編集し、`ScalabilitySettings.config` ファイルに含まれる構成変更のリストを適用することによって、拡張性の設定を有効化することも可能です。しかし、エラーを誘発する危険のため、この方法は推奨しません。

`ScalabilitySettings.config` ファイルは、`EnableEventQueues` 設定の値を `true` に設定します。これは、Sitecore がリモート イベントを記録および処理し、複数のインスタンス間で自動キャッシュ同期を実行し、分散パブリッシュを実行することができるよう構成するうえで必要なステップです。

4.2 ファイル ベースのメディアを無効化または有効化する方法

Sitecore CMS アプリケーションのファイル ベースのメディアを無効化または有効化する手順:

- /App_Config/Include/ScalabilitySettings.config ファイルの Media.DisableFileMedia 設定を対応する値に設定します。

メモ

この設定の値を変更することにより、ユーザーが新しいメディアをアップロードする場合にファイル システム ストレージとデータベース BLOB ストレージを選択できるかどうか、またはデータベース BLOB にメディアをアップロードできるかどうかを設定できます。

4.3 パブリッシュ インスタンスの名前を設定する方法

単一インスタンスの CM 環境では変更を行う必要はありません。

CM 環境が 2 つまたはそれ以上のインスタンスを含む場合には:

- パブリッシュ インスタンスとして動作する CM インスタンスを選択します。

CM パブリッシュ インスタンスは編集インスタンスを兼ねることも可能です。または、パブリッシュのみを行う専用のインスタンスとして使用することも可能です。パフォーマンス上の理由や、配置の方針、その他の考慮により、CM のパブリッシュ インスタンスを編集インスタンスから分離するためには、後者の方法を使用します。

- `/App_Config/Include/ScalabilitySettings.config` ファイルの `Publishing.PublishingInstance` 設定の値をパブリッシュ インスタンスの名前に設定します。

Sitecore CMS のアプリケーション インスタンスの名前は `web.config` ファイルの `InstanceName` 設定の値で定義されています。または、`/App_Config/Include` サブディレクトリの `.config` ファイル (たとえば、`/App_Config/Include/ScalabilitySettings.config` など) で定義することも可能です。

`InstanceName` 設定はデフォルトでは空で、その場合にはその実効値は NetBIOS のコンピューター名 (マシン名) と IIS Web サイト名をハイフン文字 (" - ") で区切って連結したものとなります。たとえば、コンピューター名が CM-SERVER-A で Web サイト名が ProjectName である場合、`InstanceName` 設定の値は CM-SERVER-A-ProjectName となります。

Sitecore が IIS Web サイトの名前を判定できない場合には、`InstanceName` のデフォルト値はコンピューター名です。

Sitecore がコンピューター名を判定できない場合には、エラーが発生します。

自動的に生成される名前を使用せずに `InstanceName` 設定を更新してインスタンス名を制御することが可能です。これはインスタンス名を短くする場合や、Sitecore が NetBIOS のマシン名や Web サイト名を判定できない場合に有効です。

Sitecore のログ ファイルを参照して下記のようなエントリーを検索することにより、現在構成されている Sitecore アプリケーションのインスタンス名を確認することが可能です。

5224 12:00:00 INFO Instance Name: CM-SERVER-A-ProjectName

4.4 HTML キャッシュ クリアを構成する方法

デフォルトでは、Sitecore は、パブリッシュ操作の終了直後に、"website" という名前の管理対象サイトの HTML キャッシュを自動的にクリアするよう構成されています。

Sitecore の HTML キャッシュの詳細については、『プレゼンテーション コンポーネント リファレンス ガイド』の「Output Caching」のセクションを参照してください。

複数の管理対象 Web サイトの構成に関する詳細は、「複数サイトの設定」に関する文書を SDN から参照してください。

デフォルト構成以外にも、以下の 3 つすべての条件が当てはまる場合は、HTML キャッシュのクリアを手動で構成する必要があります。

- 1 つ以上のレンダリングまたはサブレイアウトの出力を、キャッシュの対象として構成した。
- ソリューションに、それらのレンダリング/サブレイアウトが使用されるカスタム サイト定義が 1 つ以上ある (つまり、"website" という名前の管理対象サイト以外のサイト)。
- web.config のカスタム サイト定義で HTML キャッシュが有効になっており、HTML キャッシュ サイズが正の値に設定されている。

管理対象の Sitecore サイトの HTML キャッシュのサイズを有効化、および構成する方法の詳細については、『キャッシュ設定ガイド』のマニュアルを参照してください。

つまり、レンダリング/サブレイアウトの出力をキャッシュしない場合、またはレンダリング/サブレイアウトの出力をキャッシュするが、実装されているソリューションでカスタム サイトが定義されていない場合には、このセクションの残りの部分を飛ばすことができます。

カスタム サイト向けに、HTML の自動キャッシュ クリアを設定するには、2 通りの方法があります。スケジュール設定されたエージェントを使用して HTML キャッシュを定期的にクリアするよう構成する方法と、パブリッシュ操作が終了した直後にクリアするよう構成する方法です。

パブリッシュ操作後に HTML キャッシュ クリアを実行するよう構成する手順:

- web.config ファイルで、/events/event[@name="publish:end"]/handler/sites ノードにもう 1 つの site ノードを追加します。ここで、追加されるノードの名前は、HTML キャッシュをクリアするサイトと同じ名前です。
- web.config ファイルで、/events/event[@name="publish:end:remote"]/handler/sites ノードにもう 1 つの site ノードを追加します。ここで、追加されるノードの名前は、HTML キャッシュをクリアするサイトと同じ名前です。

たとえば、カスタム サイトの名前が `mysite` である場合、関連する構成セクションは次のようになります。

```
<event name="publish:end">
  <handler type="Sitecore.Publishing.HtmlCacheClearer, Sitecore.Kernel" method="ClearCache">
    <sites hint="list">
      <site>website</site>
      <site>mysite</site>
    </sites>
  </handler>
</event>
<event name="publish:end:remote">
  <handler type="Sitecore.Publishing.HtmlCacheClearer, Sitecore.Kernel" method="ClearCache">
    <sites hint="list">
      <site>website</site>
      <site>mysite</site>
    </sites>
  </handler>
</event>
```

スケジュール設定されたエージェントを使用し、HTML キャッシュ クリアを構成する手順：

- `web.config` ファイルで、`/scheduling/agent[@name="Sitecore.Tasks.HtmlCacheClearAgent"]` ノードに対し、`interval` 属性の値を必要な値に設定します。

たとえば、HTML キャッシュを 5 分ごとにクリアする必要がある場合は、関連する構成セクションは次のようになります。

```
<agent type="Sitecore.Tasks.HtmlCacheClearAgent" method="Run" interval="00:05:00" />
```

4.5 マシン キーを構成する方法

単一インスタンス環境では、構成を変更する必要はありません。

環境が 2 つまたはそれ以上のインスタンスを含む場合には：

- web.config ファイルで、/configuration/system.web/machineKey 要素の validationKey および decryptionKey 属性を、自動で生成されない値に設定します。環境内のすべてのインスタンスで値が同一であること、また IsolateApps 修飾子がどちらの値にも含まれないことを確認します。

web.config ファイルの <machineKey> セクションに関する詳細については、[http://msdn.microsoft.com/en-us/library/w8h3skw9\(VS.90\).aspx](http://msdn.microsoft.com/en-us/library/w8h3skw9(VS.90).aspx) を参照してください。

4.6 Master データベースへの参照を削除する方法

Master データベースへの参照を削除するには、2 つの方法があります。

- include ファイルを使用する。
- 参照を手動で削除する。

include ファイルを使用して Master データベースへの参照を削除する

Sitecore CMS 6.4 以降を使用している場合には、include ファイルを使用して Sitecore の CD インスタンスでの Master データベースへの参照を削除することができます。

- <http://sdn.sitecore.net/Reference/Sitecore%206/Scaling%20Guide.aspx> からダウンロードできる SwitchMasterToWeb.config ファイルを /App_Config/Include フォルダーにコピーします。

メモ

ファイル名は重要です。Sitecore CMS が include ファイルを処理する方法上、アルファベット順で SwitchMasterToWeb.config ファイルが Sitecore.WebDAV.config ファイルの後にリストされます。

/App_Config/ConnectionStrings.config ファイルまたは /App_Config/ConnectionStringsOracle.config ファイルで master 接続文字列を削除またはコメント化します。

Master データベースへの参照を手動で削除する

前のセクションで説明した方法の代替手段として、または Sitecore CMS 6.3 を使用している場合には、Master データベースへの参照を手動で削除することも可能です。

Master データベースへの参照を手動で削除する手順:

web.config ファイルで:

- /sites/site[@name="shell"] ノードの content 属性を web に変更します。
- /sites/site[@name="modules_shell"] ノードの content 属性を web に変更します。
- /sites/site[@name='testing'] ノードが存在する場合は、削除するか、コメント化します。
- /IDTable/param[@connectionStringName="master"] ノードの connectionStringName 属性を web に変更します。
- /databases/database[@id='master'] ノードを削除またはコメント化します。
- /search/configuration/indexes/index/locations/master ノードを削除またはコメント化します。

- database パラメーターが master に設定された Sitecore.Tasks.DatabaseAgent を削除またはコメント化します。

/App_Config/Include/Sitecore.WebDAV.config ファイルで:

- /scheduling/agent[@type="Sitecore.Tasks.CleanupFDAObsoleteMediaData"] /databases/database/[@name='master'] ノードを削除またはコメント化します。

/App_Config/ConnectionStrings.config または /App_Config/ConnectionStringsOracle.config ファイルで:

- master 接続文字列を削除またはコメント化します。

4.7 クライアント インターフェイスへのアクセスを制限する方法

Sitecore のクライアント インターフェイスへの許可されないアクセスを防止するため、CD 環境のすべてのインスタンスでそれらへのアクセスを制限する必要があります。

クライアント インターフェイスへのアクセスを制限するためには、IP ベースのセキュリティ制限を実装するか、または Sitecore の Web アプリケーションの /sitecore フォルダの下にあるすべてのファイルとフォルダー（ただし /sitecore/service フォルダを除く）への IIS の匿名アクセスを無効にすることを推奨します。

/sitecore/service フォルダは、アプリケーションで発生する各種の状況（404 Page Not Found、500 Server Error、など）を Web クライアントにレポートする場合に、Sitecore アプリケーションが使用するいくつかのサービス ASPX ページを含むため、IIS の制限から除外します。

ファイルを /sitecore/service フォルダから /sitecore フォルダの外に移動することも可能です。その場合には web.config ファイルの次の設定を更新する必要があります：

```

E      r      r      o      r      P      a      g      e
N      o      A      c      c      e      s      s      U      r      l
N      o      L      i      c      e      n      s      e      U      r      l
L      a      y      o      u      t      N      o      t      F      o      u      n      d      U      r      l
I      t      e      m      N      o      t      F      o      u      n      d      U      r      l
LinkItemNotFoundUrl

```

IIS 7 以降における IP ベースのセキュリティ制限の構成に関する詳細は、<http://www.iis.net/ConfigReference/system.webServer/security/ipSecurity> を参照してください。

IIS 6 以前における IP ベースのセキュリティ制限の構成に関する詳細は、<http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/128d26dd-decb-42f9-8efb-30724d1a2f29.mspx?mfr=true> を参照してください。

4.8 リモート パブリッシュ ターゲットを構成する方法

リモート パブリッシュ ターゲットを構成するためには:

- パブリッシュ ターゲット データベースを作成します。
- パブリッシュ ターゲット データベースを登録します。

パブリッシュ ターゲット データベースを作成する

データベース サーバーにパブリッシュ ターゲット データベースを作成する手順:

- Sitecore の .zip 配布の中にあるデフォルトの web パブリッシュ ターゲットのアーカイブからデータベースを作成します。

または

- 既存のパブリッシュ ターゲット データベースを複製します。

パブリッシュ ターゲット データベースを登録する

各 CM インスタンスにパブリッシュ ターゲットデータベースを登録する手順:

1. web.config ファイルで id が web である /configuration/sitecore/databases/database 要素を複製し、新しい <database> 要素の id 属性を、パブリッシュ ターゲット データベースの接続文字列の名前、たとえば pub などに変更します。
2. /App_Config/ConnectionStrings.config または /App_Config/ConnectionStringsOracle.config ファイルで、web という名前の /connectionStrings/add 要素を複製し、name 属性を web から接続文字列名、たとえば pub などに変更し、そして connectionString property を変更し、パブリッシュ ターゲット データベースにアクセスするようにします。
3. CM インスタンスの 1 つで、**コンテンツ エディター**を使って /Sitecore/System/Publishing targets アイテムを選択します。
4. **コンテンツ エディター**で、データベースにちなんで名付けられ、System/Publishing target データ テンプレートに基づくパブリッシュ ターゲット定義アイテムを挿入します。
5. **コンテンツ エディター**の新しいパブリッシュ ターゲット定義アイテムで、Data セクションの Target database フィールドで、パブリッシュ ターゲット データベースの名前、たとえば pub を入力します。

4.9 SQL Server レプリケーションを構成する方法

Sitecore データベースに SQL Server レプリケーションを構成する方法の詳細は、次の URL から『[SQL Server レプリケーション ガイド](#)』を参照してください。

4.10 セキュリティ キャッシュのクリアに対応する方法

セキュリティ キャッシュに対応する 2 つの方法があります：

- セキュリティ キャッシュの有効期限切れ
- セキュリティ イベントの転送

「拡張性の設定の有効化の方法」セクションの指示を実装した後では、セキュリティ キャッシュの有効期限切れは自動的に有効化されています。セキュリティ イベントの転送による方法と比較して、この方法の唯一のデメリットは、セキュリティ キャッシュが期限切れのまま残る期間が長くなることです。

セキュリティ イベントの転送を使用すると、セキュリティ キャッシュは関連するセキュリティ イベントが発生した直後に自動的に更新されます。しかしこの方法では、関連するセキュリティ イベントが発生する都度、1 つの環境 (CM または CD) が別の環境 (CD または CM) のリモート データベースに書き込む必要があります。この方法は、セキュリティ キャッシュの即時の更新が非常に重要である環境のみに推奨されます。

セキュリティ キャッシュの有効期限切れを構成する手順：

- セキュリティ データの更新は実行しないが、それを読む必要のある環境 (通常は CD) の各インスタンスにおいて、/App_Config/Include/ScalabilitySettings.config または web.config ファイルの Caching.SecurityCacheExpiration 設定の値を適切な値、たとえば 00:20:00 などに設定します。

セキュリティ イベントの転送を構成する手順：

1. セキュリティ データの更新を実行する環境 (通常は CM) の各インスタンスにおいて、/App_Config/Include サブディレクトリで、ForwardingSecurityEvents.config.example というファイルの名前を ForwardingSecurityEvents.config に変更 (.example 拡張子を削除) します。
2. /App_Config/Include/ForwardingSecurityEvents.config ファイルの /sitecore/pipelines/initialize/processor/domains 要素に、同期を必要とするドメインにちなんだ名前の要素を追加します。ドメインの名前を新しい要素のテキストとして使用します。

たとえば、セキュリティ イベントを domainname という名前のドメインに転送するためには、下記の例で示すように <domainname> 要素を追加します。

```
...
<domains hint="list">
  <extranet>extranet</extranet>
  <domainname>domainname</domainname>
</domains>
...
```

3. /App_Config/Include/ForwardingSecurityEvents.config ファイルの /sitecore/pipelines/initialize/processor/targets 要素に、CD 環境からリモート データベースにちなんだ名前の要素を追加します。データベースの名前を新しい要素のテキストとして使います。

通常的环境では、セキュリティ イベントを pub という名前のデータベースに転送するためには、下記の例で示すように <web> 要素を <pub> に変更します。

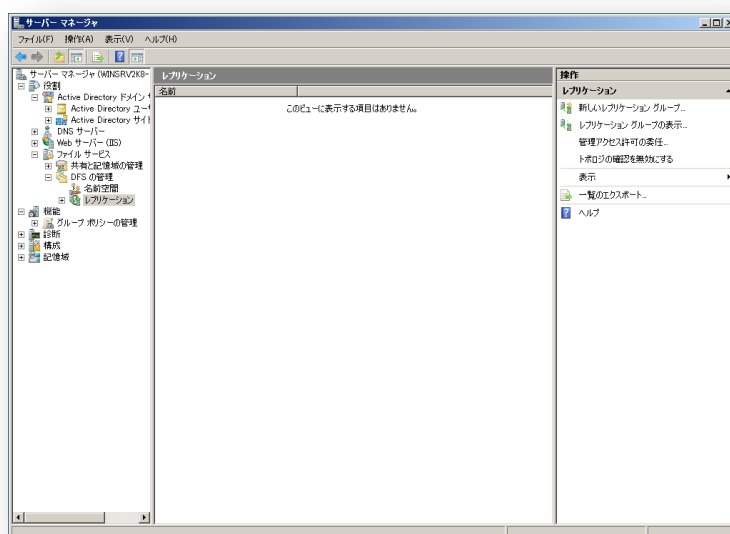
```
...  
<targets hint="list">  
  <pub>pub</pub>  
</targets>
```

4. /App_Config/Include/ForwardingSecurityEvents.config ファイルで /sitecore/databases 要素を削除します。

4.11 DFS を構成する方法

DFS を構成する手順:

1. グループのサーバーの 1 つを、プライマリ サーバーと指定します。レプリケーションは最初にプライマリ サーバーから始まり、それに続くグループ内の任意のサーバーにおけるファイル システム イベントは他にレプリケートされます。
2. プライマリ インスタンスをホストしているサーバー上の Microsoft Windows で、**サーバー マネージャー**を開始します。
3. **サーバー マネージャー**で **[役割]**を展開し、次に **[ファイル サービス]**を、そして **[DFS の管理]**を展開し、**[レプリケーション]**を選択します。

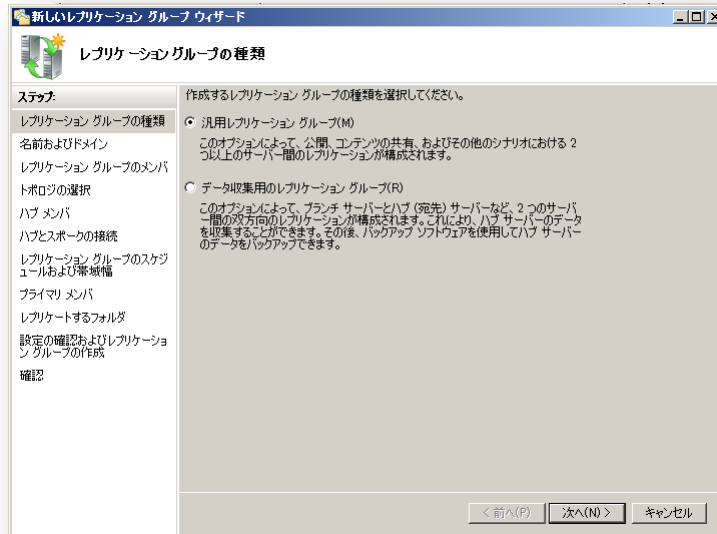


4. **[操作]** パネルで、**[新しいレプリケーション グループ...]** をクリックすると、**[新しいレプリケーション グループ]** ウィザードが表示されます。

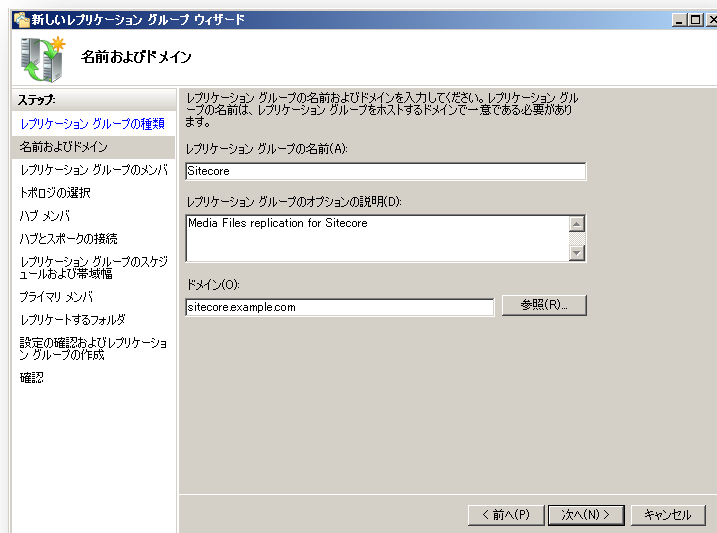
[新しいレプリケーション グループ] ウィザードを使ってレプリケーション グループを構成します。

レプリケーション グループを構成する手順:

1. [新しいレプリケーション グループ] ウィザードの [レプリケーション グループの種類] ウィンドウで [汎用レプリケーション グループ] を選択します。

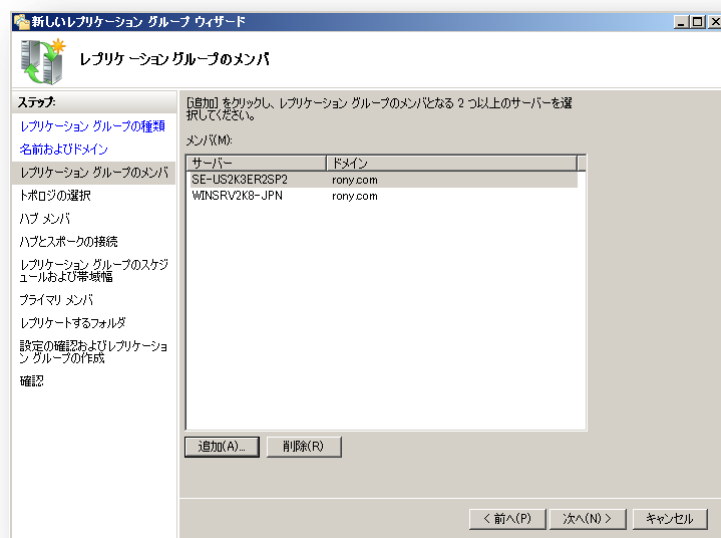


2. [次へ] をクリックすると [名前およびドメイン] ウィンドウが表示されます。



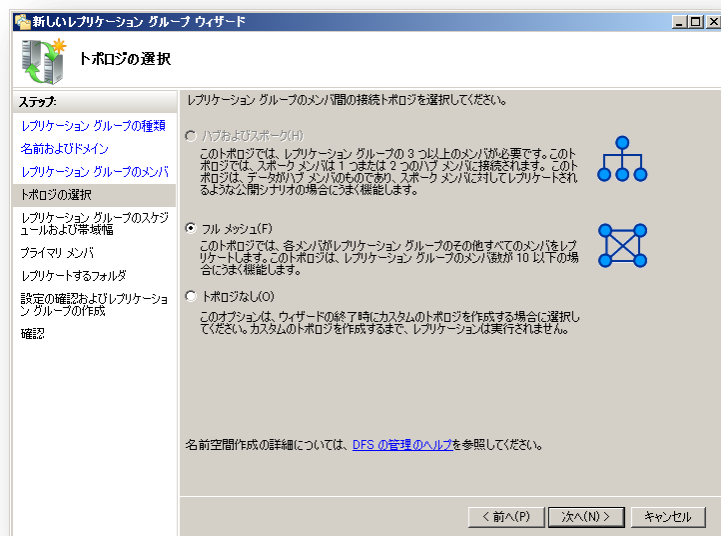
3. [名前およびドメイン] ウィンドウの [レプリケーション グループの名前] フィールドに名前を入力します。
4. [レプリケーション グループのオプションの説明] フィールドに環境とその目的の説明を入力します。たとえば、「CM 本番環境ファイル メディア」などです。
5. [ドメイン] フィールドで [参照] をクリックして適切なドメインを選択します。

6. [次へ] をクリックすると [レプリケーション グループのメンバ] ウィンドウが表示されます。



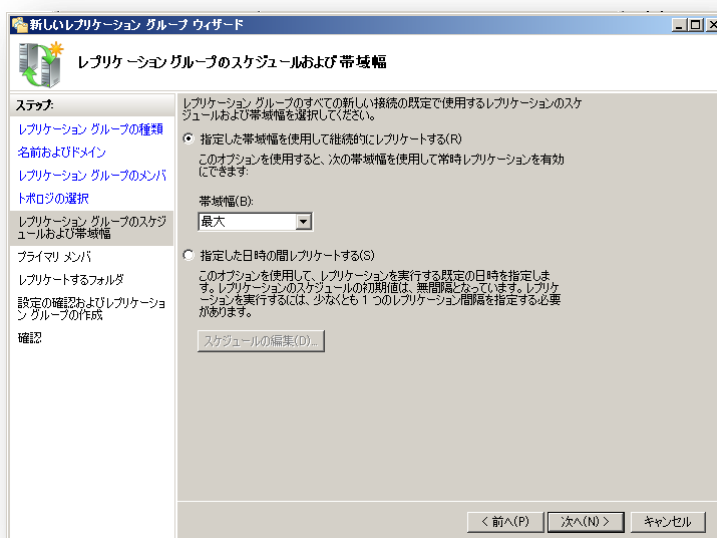
7. [レプリケーション グループのメンバ] ウィンドウで [追加] をクリックして、レプリケーションに参加するサーバーを追加します。

8. [次へ] をクリックすると [トポロジの選択] ウィンドウが表示されます。



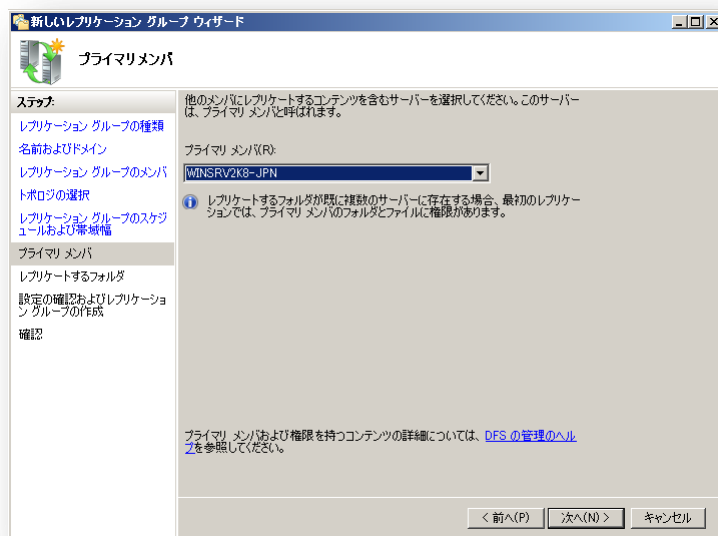
9. [トポロジの選択] ウィンドウで [フル メッシュ] を選択します。

10. **[次へ]** をクリックすると **[レプリケーション グループのスケジュールおよび帯域幅]** ウィンドウが表示されます。



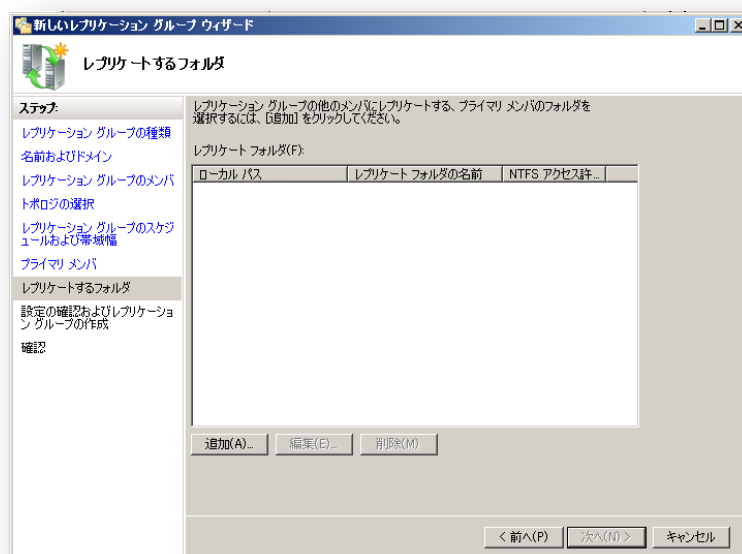
11. **[レプリケーション グループのスケジュールおよび帯域幅]** ウィンドウで、**[指定した帯域幅を使用して継続的にレプリケートする]** を選択し、**[帯域幅]** フィールドで **[最大]** を選択します。

12. **[次へ]** をクリックすると **[プライマリ メンバ]** ウィンドウが表示されます。



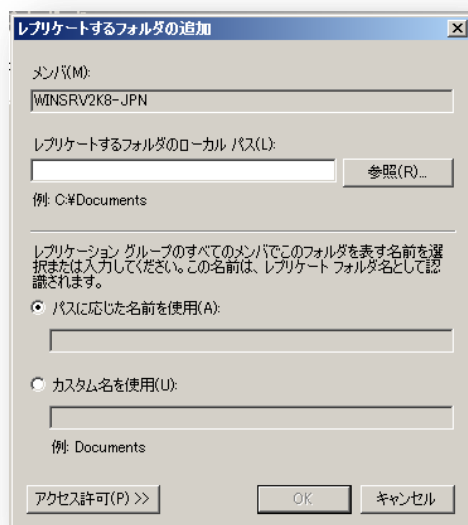
13. **[プライマリ メンバ]** ウィンドウの **[プライマリ メンバ]** フィールドでプライマリ メンバ サーバーを選択します。

14. **[次へ]** をクリックすると **[レプリケートするフォルダ]** ウィンドウが表示されます。



このウィンドウを使って、レプリケートするサブディレクトリを選択します。

15. **[レプリケートするフォルダ]** ウィンドウで **[追加]** をクリックすると **[レプリケートするフォルダの追加]** のダイアログボックスが表示されます。



16. **[レプリケートするフォルダの追加]** ダイアログ ボックスの **[レプリケートするフォルダのローカル パス]** フィールドで **[参照]** をクリックしてレプリケートするサブディレクトリを選択し、**[OK]** をクリックします。

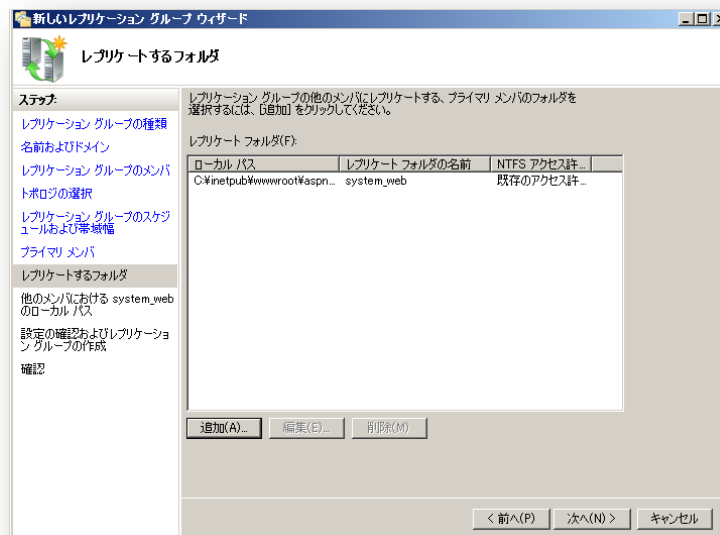
ファイル メディアのレプリケーションを構成するためには、/App_Config/Include/ScalabilitySettings.config または web.config ファイルの Media.FileFolder 設定で指定した親ディレクトリを選択します。たとえば、Media.FileFolder 設定が /App_Data/Replicated/MediaFiles に設定されている場合は、/App_Data/Replicated フォルダ — C:\inetpub\wwwroot\%sitecore%\projectName\%website%\App_Data\Replicated を選択する必要があります。

セキュリティ ドメイン構成ファイルを同期するためには、web.config ファイルの domains という名前の /sitecore/configuration/configStores/add 要素の arg0 属性で指定されたセキュリティ ドメイン構成ファイルを含むサブディレクトリを選択します。たとえば、/App_Config/Security (C:\inetpub\wwwroot\%Sitecore%\App_Config\Security) などです。

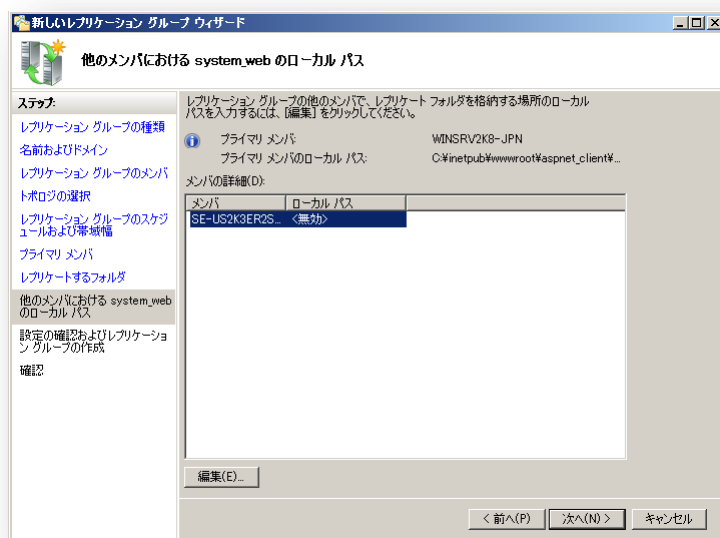
注意

このセクションで指定された以外のサブディレクトリを DFS に構成しないでください。

[レプリケートするフォルダ] ウィンドウは下記の例のようになります:

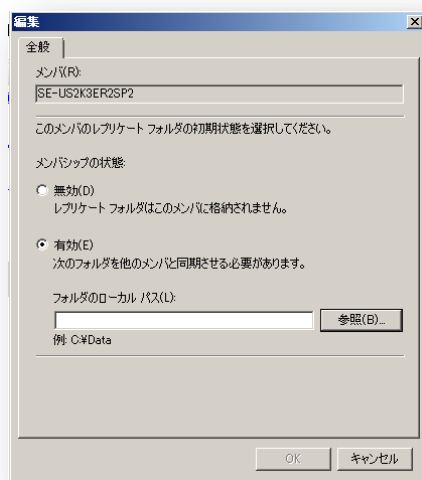


17. [次へ] をクリックすると [他のメンバにおける...ローカル パス] ウィンドウが表示されます。



レプリケーション グループの各サーバー上の各レプリケート対象のサブディレクトリのターゲット パスを構成します。

18. [他のメンバにおける...ローカル パス] ウィンドウで、各サーバーを選択して [編集] をクリックします。[編集] ダイアログ ボックスが表示されます。

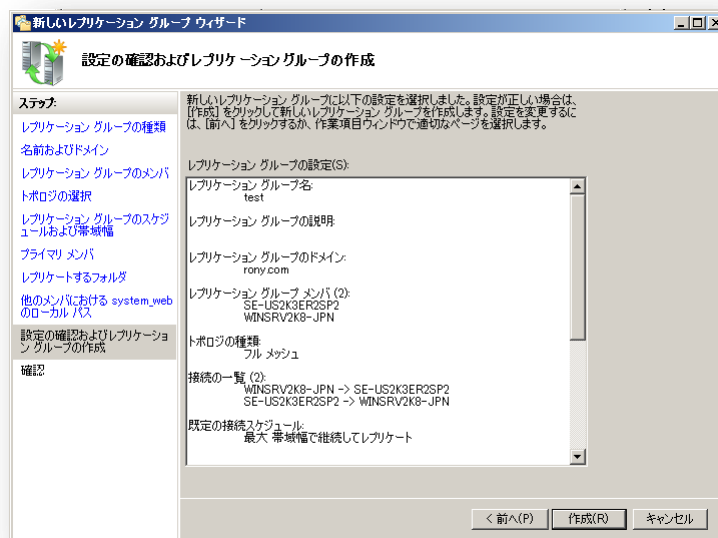


19. [編集] ダイアログで [有効] を選択し、[フォルダのローカル パス] フィールドで [参照] をクリックし、適切なローカル パスを選択し、[OK] をクリックします。

ファイル メディアのレプリケーションを構成するためには、ターゲット インスタンスの /App_Config/Include/ScalabilitySettings.config または web.config ファイルの Media.FileFolder 設定で指定した親ディレクトリを選択します。たとえば /App_Data/Replicated/MediaFiles (C:\inetpub\sitecore\wwwroot\Sitecore\App_Data\Replicated\MediaFiles) などです。

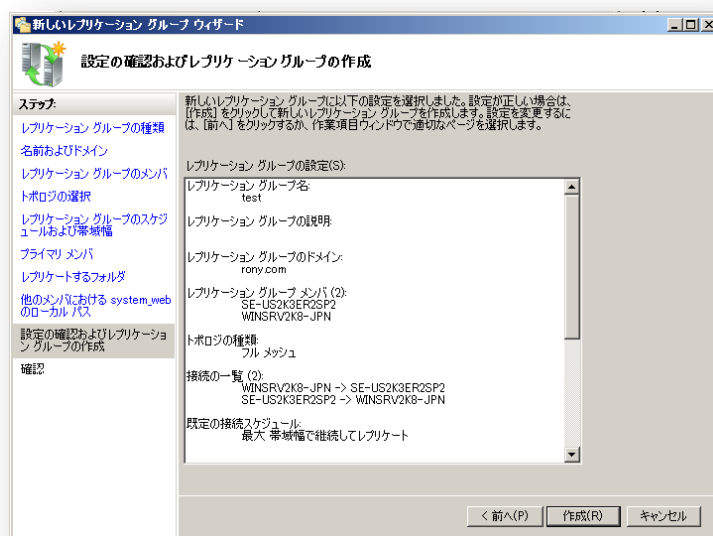
セキュリティドメイン構成ファイルを同期するためには、CD の Web 配置ターゲット インスタンスの web.config ファイルの domains という名前の /sitecore/configuration/configStores/add 要素の arg0 属性で指定されたセキュリティドメイン構成ファイルを含むサブディレクトリを選択します。たとえば、/App_Config/Security (C:\inetpub\sitecore\projectName\website\App_Config\Security) などです。

20. [他のメンバにおける... ローカル パス] のウィンドウで [次へ] をクリックすると [設定の確認およびレプリケート グループの作成] ウィンドウが表示されます。

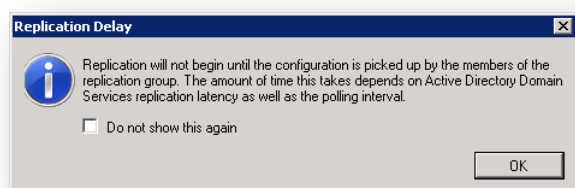


21. 指定した設定がすべて正しいことを確認して [作成] をクリックします。

22. [確認] のウィンドウで、[閉じる] をクリックします。



Windows のドメインのグループ ポリシーの適用の遅延によっては、レプリケーションは直ちに開始されず、次のようなメッセージが表示される場合があります。



4.12 Web 配置の構成方法

ファイル メディアの CM から CD への Web 配置を構成するためには、パブリッシュ インスタンス上またはすべての CM インスタンス上で:

1. /App_Config/Include サブディレクトリで、WebDeploy.config.example ファイルの名前を WebDeploy.config に変更 (.example 拡張子を削除) します。
2. /App_Config/Include/WebDeploy.config ファイルで、/configuration/sitecore/events/event/handler 要素のパラメーターを次のように構成します:

要素	用途
<TargetDatabase>	特定されたパブリッシュ ターゲット データベースにパブリッシュしたとき、イベント ハンドラーが Web 配置ツールを起動する。TargetDatabase は /App_Config/ConnectionStrings.config ファイルの /configuration/add 要素の name 属性に合致する。
<TargetServer>	ファイルを受け取るターゲット サーバーの IP アドレスまたは DNS 名、またはターゲット サーバー上の Web 配置エージェント サービスの URL。詳細については、「 Web 配置ツールのインストール 」の記事を参照してください。
<UserName>	リモート サーバー上の必要なファイル システムへの変更アクセス権をもつアカウントの Windows ドメインとユーザー名。たとえば、DOMAIN¥username (ローカル Windows アカウントにはドメイン名は省略可能)
	メモ Web Deploy 2.0 にはバグがあるため、ユーザーは初期登録された管理者か、ドメイン管理者セキュリティ グループの一員である必要があります。この問題に対する Sitecore 参照番号は 351405 です。この要件に関する詳細については、 http://learn.iis.net/page.aspx/1023/web-deploy-error-codes/ を参照してください。
<Password>	前述の <UserName> パラメーターで指定されたユーザーのパスワード。
<LocalRoot>	配置するファイルを含むサブディレクトリへの完全なパス。省略された場合、<LocalRoot> 要素の実効値はソース Web サイトのドキュメント ルート。
<RemoteRoot>	リモート サーバー上で配置されたファイルを受け取るサブディレクトリへの完全なパス。省略された場合、<RemoteRoot> 要素の実効値はターゲット Web サイトのドキュメント ルート。

要素	用途
<items/*>	配置する、1 つまたは複数のローカル サブディレクトリまたはファイル。それぞれが <LocalRoot> 要素で指定されたサブディレクトリに相対的な、<items> 要素内の個別の要素によって構成されている。サブディレクトリとファイルはターゲット サーバー上の対応する場所に、<RemoteRoot> 要素で指定された値に相対的に配置される。<items> 要素内の要素の名前は任意。

Sitecore は publish:begin イベント ハンドラーを使って Web 配置ツールを起動します。CD インスタンスはパブリッシュしないので、このイベント ハンドラーは CD インスタンスでは意味を持ちません。ロードバランシング CM 環境では、CM パブリッシュ インスタンスのみが Web 配置ツールを起動します。Web 配置ツールを起動するイベント ハンドラーは、他の CM インスタンスには意味を持ちません。このイベント ハンドラーは CD 環境では効果がなく、CM 環境でもパブリッシュ インスタンス以外には効果がありません。

デフォルトでは、Web 配置ツールを起動する publish:begin イベント ハンドラーは非同期で、1 つまたは複数のバックグラウンド スレッドを使用します。このため、コンテンツに参照されるファイル メディアの配置が完了する前に、コンテンツがパブリッシュ ターゲット データベースに現れる場合があります。ハンドラーが同期的に稼働し、ファイル メディアの配置がコンテンツのパブリッシュ前に完了するようにするためには、/App_Config/Include/WebDeploy.config ファイルの /configuration/sitecore/events/event/handler 要素の中に、値が true である <Synchronous> という名前の要素を作成します。たとえば:

```
<configuration>
  <sitecore>
    <events>
      <event name="publish:begin">
        <handler type="Sitecore.Publishing.WebDeploy.PublishHandler,
          Sitecore.Publishing.WebDeploy" method="OnPublish">
          <synchronous>true</synchronous>
        <tasks hint="list:AddTask">
          ...
```

複数の Web 配置ターゲットの構成方法

Web 配置ツールを使って、単一 CM インスタンスから CD 環境の複数のターゲット インスタンスにファイルを配置することができます。

複数の配置ターゲットを構成する手順:

1. /App_Config/Include/WebDeploy.config ファイルで、既存の /configuration/sitecore/events/event/handler/tasks/* 要素を複製します。
2. /App_Config/Include/WebDeploy.config の、新規の要素で、前のセクションの表で説明したパラメーターを更新します。