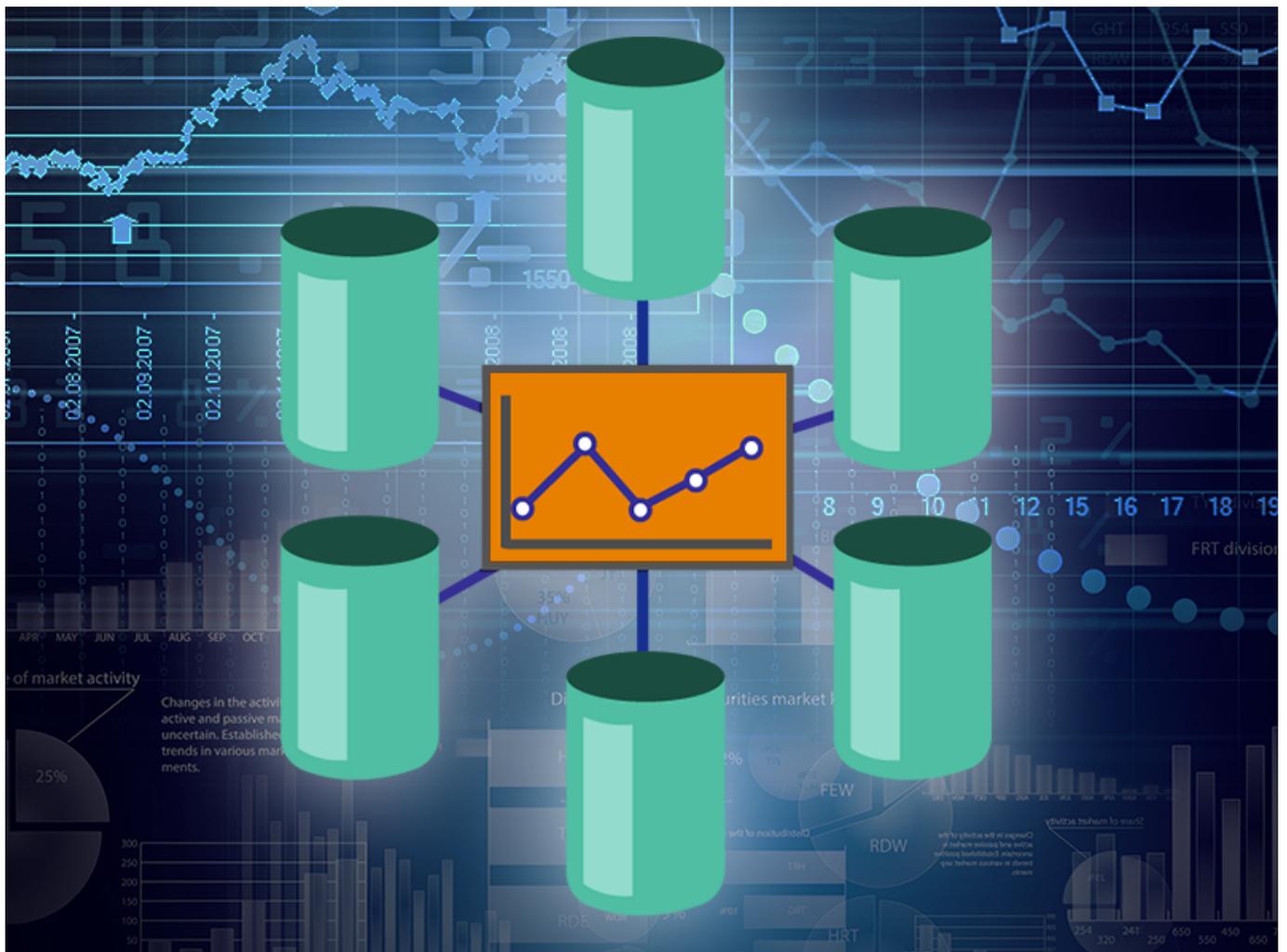


記事

[Toshihiko Minamoto](#) · 2020年12月8日 4m read

DeepSee: データベース、ネームスペース、マッピング(1/5)

インスタンスのデータに基づくビジネスインテリジェンスを実装しようとしています。DeepSeeを使うには、データベース環境をどのようにセットアップするのがベストですか？



このチュートリアルでは、DeepSee の 3 つのアーキテクチャ例を示しながら、上記の質問を解決します。基本的なアーキテクチャモデルを、その制限を重点に説明するところから始めましょう。以降のモデルは、複雑さが中程度のビジネスインテリジェンスアプリケーションに推奨されており、ほとんどのユースケースで十分ではありません。チュートリアルの最後には、高度な実装を管理できるように、アーキテクチャの柔軟性を高める方法を説明します。

このチュートリアルに含まれる例では、新しいデータベースとグローバルマッピングを紹介し、それらをセットアップする理由とタイミングについて説明します。アーキテクチャを構築する際には、より柔軟な例から得られるメリットについて説明します。

始める前に

プライマリサーバと分析サーバ

データの高性能を実現する場合、InterSystems では一般的にミラリングとシャドウイングを使用して、ミラ/シャドウサーバに DeepSee を実装することをお勧めしています。データの元のコードをホストするマシンを「プライマリサーバ」と呼び、データとビジネスインテリジェンスアプリケーションのコードをホストするマシンを「分析(または)ポ・ティンク)サーバ」と呼びます。

プライマリサーバと分析サーバを用意しておくことは非常に重要です。これは主に、いずれのサーバにおいてもパフォーマンスに関する問題を回避するためです。 [推奨アーキテクチャ](#) に関するドキュメントをご覧ください。

データとアプリケーションコード

ソースデータとコードを同じデータベースに格納することは、通常、規模の小さなアプリケーションでのみうまく機能します。より大規模なアプリケーションでは、ソースデータとコードをそれぞれの専用データベースに格納することをお勧めします。専用のデータベースを使用することで、データを分離しながらも、DeepSee が実行するすべてのネームスペースでコードを共有することができます。ソースデータ用のデータベースは、本番サーバからミラリングできるようにしておく必要があります。このデータベースは、読み取り専用または読み取り/書き込みのいずれかです。このデータベースでは、ジャーナリングを有効にしておくことをお勧めします。

ソースとカスタムアプリケーションは、本番サーバと分析サーバの両方にある専用データベースに格納します。これら 2 つのソースコード用データベースは同期している必要がなく、同じバージョンの Caché を実行している必要はありません。コード別の場所で定期的にバックアップされているのであれば、ジャーナリングは通常必要ではありません。

このチュートリアルでは、次の構成を使用しています。分析サーバの APP ネームスペースには、デフォルトのデータベースとして APP-DATA と APP-CODE があります。APP-DATA データベースは、プライマリサーバにあるソースデータ用データベースのデータ(ソースステップのクラスとファクト)にアクセスできます。APP-CODE データベースは、Caché コード(.cls と .INT ファイル)とほかのカスタムコードを格納します。このようにデータとコードを分離するのは典型的なアーキテクチャであり、ユーザは、DeepSee コードとカスタムアプリケーションを効率的にデプロイすることができます。

Namespaces

Create New Namespace

Refresh: off on sec

Current Namespaces and their default databases for globals and routines:

Filter: Page size: Max rows: Results: 6 Page: |< << **1** >> >| of 1

Namespace	Globals	Routines	Temp Storage				
%ALL	CACHETEMP	CACHETEMP	CACHETEMP	Global Mappings	Routine Mappings	Package Mappings	Delete
%SYS	CACHESYS	CACHESYS	CACHETEMP	Global Mappings	Routine Mappings	Package Mappings	-
APP	APP-DATA	APP-CODE	CACHETEMP	Global Mappings	Routine Mappings	Package Mappings	Delete
DOCBOOK	DOCBOOK	DOCBOOK	CACHETEMP	Global Mappings	Routine Mappings	Package Mappings	Delete
SAMPLES	SAMPLES	SAMPLES	CACHETEMP	Global Mappings	Routine Mappings	Package Mappings	Delete
USER	USER	USER	CACHETEMP	Global Mappings	Routine Mappings	Package Mappings	Delete

異なるネームスペースでの DeepSee の実行

DeepSee

を使用し、ビジネスインテリジェンス実装は、異なるネームスペースから実行されるのがよくあります。この記事では単一の APP ネームスペースのセットアップ方法を示しますが、同じ手順を使えば、ビジネスインテリジェンスアプリケーションで実行するすべてのネームスペースをセットアップすることは可能です。

ドキュメント

ドキュメントに含まれる [初回セットアップの実行](#) に関するページの内容を理解しておくことをお勧めします。

このページには、Web アプリケーションのセットアップ、DeepSee グローバルを個別のデータベースに配置する方法、および DeepSee グローバルの代替マッピングのリストが含まれています。

このシリーズの [第 2 部](#) では、[基本的なアーキテクチャモデル](#)の実装について説明します。

[#マッピング](#) [#チュートリアル](#) [#デプロイ](#) [#データベース](#) [#ベストプラクティス](#) [#分析](#) [#初心者](#) [#InterSystems IRIS BI \(DeepSee\)](#)

ソースURL: <https://jp.community.intersystems.com/post/deepsee-%E3%83%87%E3%83%BC%E3%82%BF%E3%83%99%E3%83%BC%E3%82%B9%E3%80%81%E3%83%8D%E3%83%BC%E3%83%A0%E3%82%B9%E3%83%9A%E3%83%BC%E3%82%B9%E3%80%81%E3%83%9E%E3%83%83%E3%83%94%E3%83%B3%E3%82%B0%EF%BC%8815%EF%BC%89>