

ADO.Net Entity Framework

[[SQL Server](#)][[SQL Server Compact](#)]

<http://msdn.microsoft.com/ja-jp/library/bb399567.aspx>

概要

- ・データ指向のソフトウェア アプリケーションの開発をサポート
- ・.NET Framework のコンポーネント
- ・.NET Framework 3.5 Service Pack 1 (SP1) で実行可能

■アプリケーション開発では、2つの大きく異なる目的を達成する必要

1. エンティティ、リレーションシップ、およびロジックをモデル化
2. データの格納と取得に使用するデータ エンジンに取り組む

■エンティティ フレームワーク を使用

- ・開発者は、顧客や顧客の住所など、ドメイン固有のオブジェクトおよびプロパティの形式でデータを扱うことができる。
- ・データが格納されている、基になるデータベース テーブルや列を意識する必要がない

■データ モデリングの従来的一般的手法

- ・概念モデル、論理モデル、および物理モデルの3つに分割
- ・概念モデルは通常、アプリケーションの要件を収集して伝えるためのツールとして使用され、プロジェクトの初期段階でのみ考慮および検討され、その後ほとんど使用されない

■エンティティ データ モデル (EDM)

- ・開発者が概念モデル内のエンティティとリレーションシップに対してクエリを実行し、エンティティ フレームワーク がこのような操作をデータ ソース固有のコマンドに変換という形で、概念モデルを活用
- ・特定のデータ ソースへの依存関係をアプリケーションにハードコーディングしなくても済む
- ・概念モデル、ストレージ モデル、および2つのモデル間のマッピングは、エンティティ データ モデル (EDM) と呼ばれる外部仕様で表現

3つのモデル ファイルとマッピング ファイルで定義

ファイル	内容
<u>概念</u> スキーマ定義言語ファイル (.csdl)	<u>概念</u> モデル
ストア スキーマ定義言語ファイル (.ssdl)	ストレージ モデル (論理モデルとも呼ばれる) を定義
マッピング スキーマ言語ファイル (.msl)	ストレージ モデルと <u>概念</u> モデルの間のマッピングを定義

■インピーダンスミスマッチ

単なる O/R マッピング ソリューションではない

- ・既存のソリューションでは、オブジェクト指向のクラスやプロパティをリレーショナルテーブルやリレーショナル列にマップするだけ
- ・エンティティ フレームワーク では、論理モデルのリレーショナル テーブル、列、および外部キー制約を概念モデルのエンティティとリレーションシップにマップすることで、柔軟にオブジェクトを定義して論理モデルを最適化

基本動作

- ・概念モデルのエンティティとリレーションシップとして表されるデータにアプリケーションからアクセスして変更できるようにする
- ・Object Services は EDM を使用して、概念モデルで表されるエンティティ型に対するオブジェクト クエリをデータ ソース固有のクエリに変換
- ・クエリ結果は、Object Services によって管理されるオブジェクトに具体化

■拡張可能なデータ クラス

- ・エンティティ データ モデル ツールでは、概念モデルに基づく拡張可能なデータ クラスが生成される
- ・クラスは、開発者が追加するメンバで拡張できる部分クラス。
- ・特定の概念モデルに対して生成されるクラスは、エンティティをオブジェクトとして具体化したり変更を追跡して保存したりするための Object Services を提供する基本クラスから派生。
- ・開発者は、このようなクラスを使用して、エンティティとリレーションシップをナビゲーション プロパティによって関連付けられたオブジェクトとして扱うことができる。

■ EDM に対してクエリを実行してオブジェクトを返す方法

方法	内容
<u>LINQ to Entities</u>	概念モデルで定義されているエンティティ型に対してクエリを実行するための統合言語クエリ (LINQ) のサポートを提供
<u>Entity SQL</u>	概念モデルのエンティティを直接操作し、継承やリレーションシップなどの EDM 機能をサポートする、ストレージの影響を受けない <u>SQL</u> 。オブジェクト クエリと EntityClient プロバイダを使用して実行されるクエリの両方で使用
<u>クエリ ビルダ メソッド</u>	LINQ スタイルのクエリ メソッドを使用して Entity <u>SQL</u> クエリを構築

■オブジェクトの追加、変更、および削除

- ・概念モデルのエンティティ コンテナを表すObjectContext から派生したクラスを生成し、このオブジェクト コンテキストは、変更の追跡や ID、同時実行、およびリレーションシップの管理などの機能を提供する
- ・このクラスは、データ ソースに挿入、更新、および削除を書き込む SaveChanges メソッドも公開
- ・このような変更は、クエリと同様に、システムによって自動的に生成されるコマンドで行うことも、特定のストアド プロシージャを使用するように指定することも可能

シナリオ

- ・ <http://msdn.microsoft.com/ja-jp/library/bb738689.aspx>

Entity Framework のデータ モデリング

- ・ <http://msdn.microsoft.com/ja-jp/library/bb399183.aspx>

■エンティティ データ モデル (EDM)

- ・ プログラミング言語やリレーショナル データベースの構造による制約を受けない中立的な形式でデータが定義される
- ・ エンティティおよびリレーションシップの詳細を指定したり、それらをデータ構造として実装したりする際に使用

■ 2 つの基本的な型

型	内容
EntityType	アプリケーション ドメインのデータ構造詳細の抽象的な仕様
AssociationType	型と型の論理的な関係