

# SQLiteと ADO.NET Entity Frameworkで お手軽DB利用

エヌ・ティ・ティ・システム開発  
こくぶんまさひろ

## 自己紹介

### ➤ こくぶん まさひろ

- 独立系SI企業の一般社員
- C#を中心に.NET関連技術を趣味で日々追いかけています

### ➤ 不定期ながらWeb logを書いています

- <http://d.hatena.ne.jp/masa-k/>
- <http://blogs.wankuma.com/masak/>

### ➤ わんくま同盟で活動しています

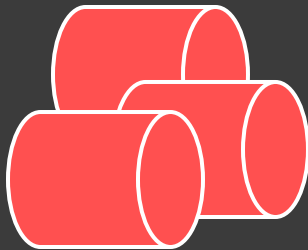
## SQLiteって？

- サーバを使わずにアプリケーションに組み込んで使用する軽量のデータベース
- データは単一のファイルで管理される
- 大規模なシステムには向かない

# ADO.NET Entity Frameworkって？

- .NETの新しいデータアクセステクノロジー

オブジェクト



オブジェクトとデータベースでフォーマットにずれがある  
(O/Rインピーダンスミスマッチ)

ADO.NET Entity Framework



O/Rインピーダンスミスマッチを解消

データベース

# ADO.NET Entity Frameworkって？

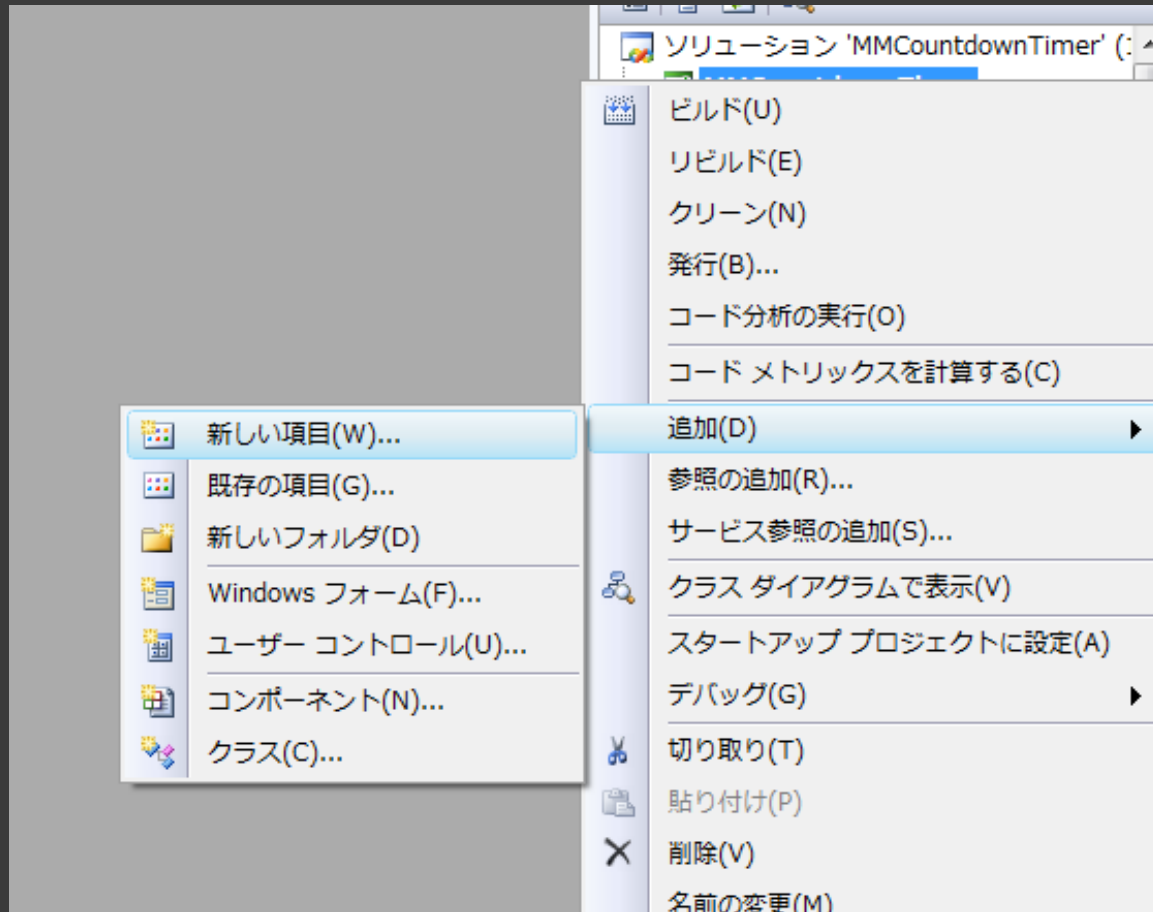
- Visual Studio 2008 Service Pack 1で追加
- SQL Serverが対象
  - ※他のRDBMSはサードパーティ製のデータプロバイダが必要
- 現時点(V1)ではまだまだ発展途上
  - ※複雑な事をやろうとすると、ちょっと大変かも...

# SQLite+ADO.NET Entity Framework

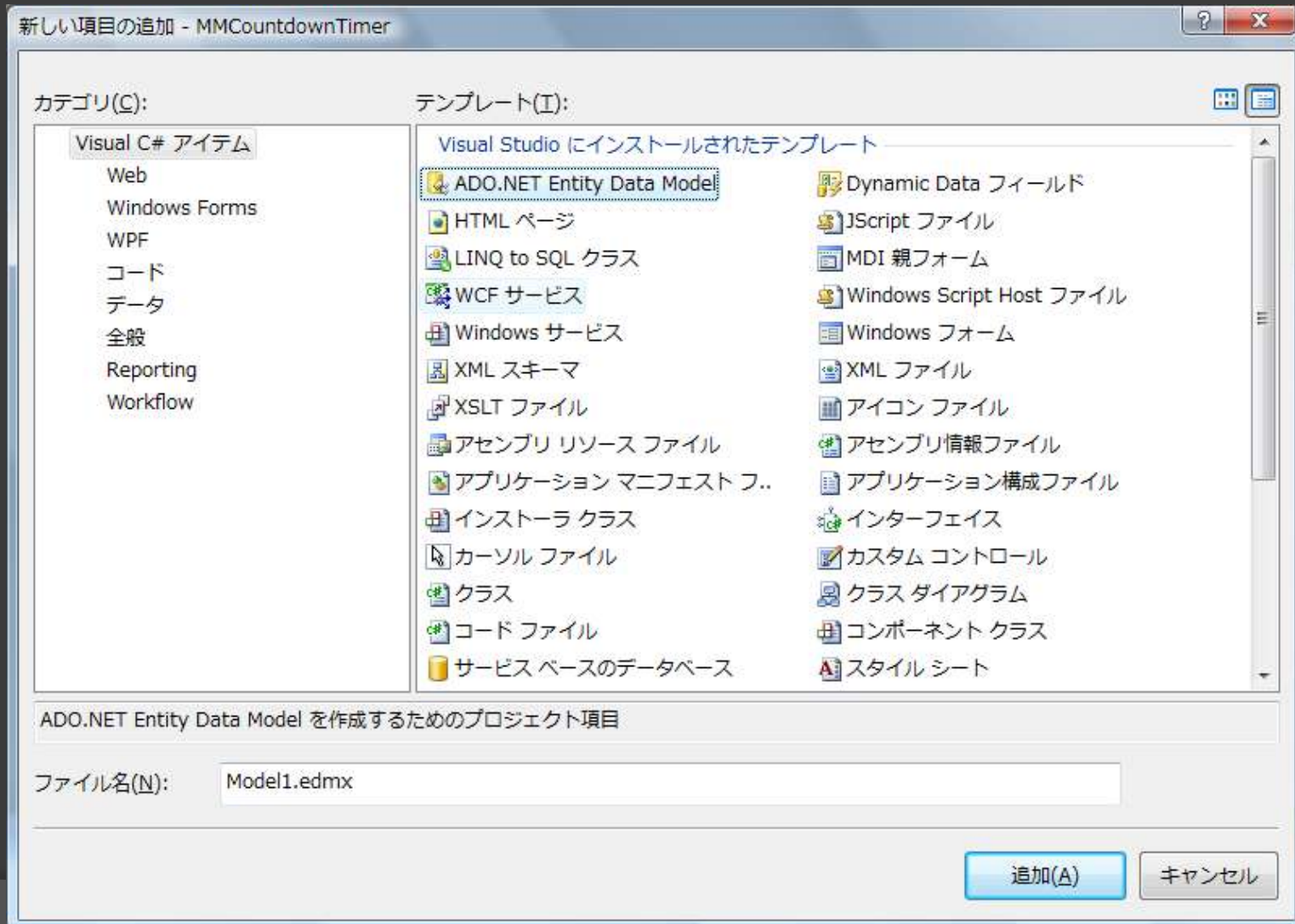
- SQLite用のADO.NETプロバイダを使えばADO.NET Entity Frameworkを利用可能
- SQLite用のADO.NETプロバイダはフリーで公開されています

<http://sqlite.phxsoftware.com/>

# SQLite+ADO.NET Entity Framework

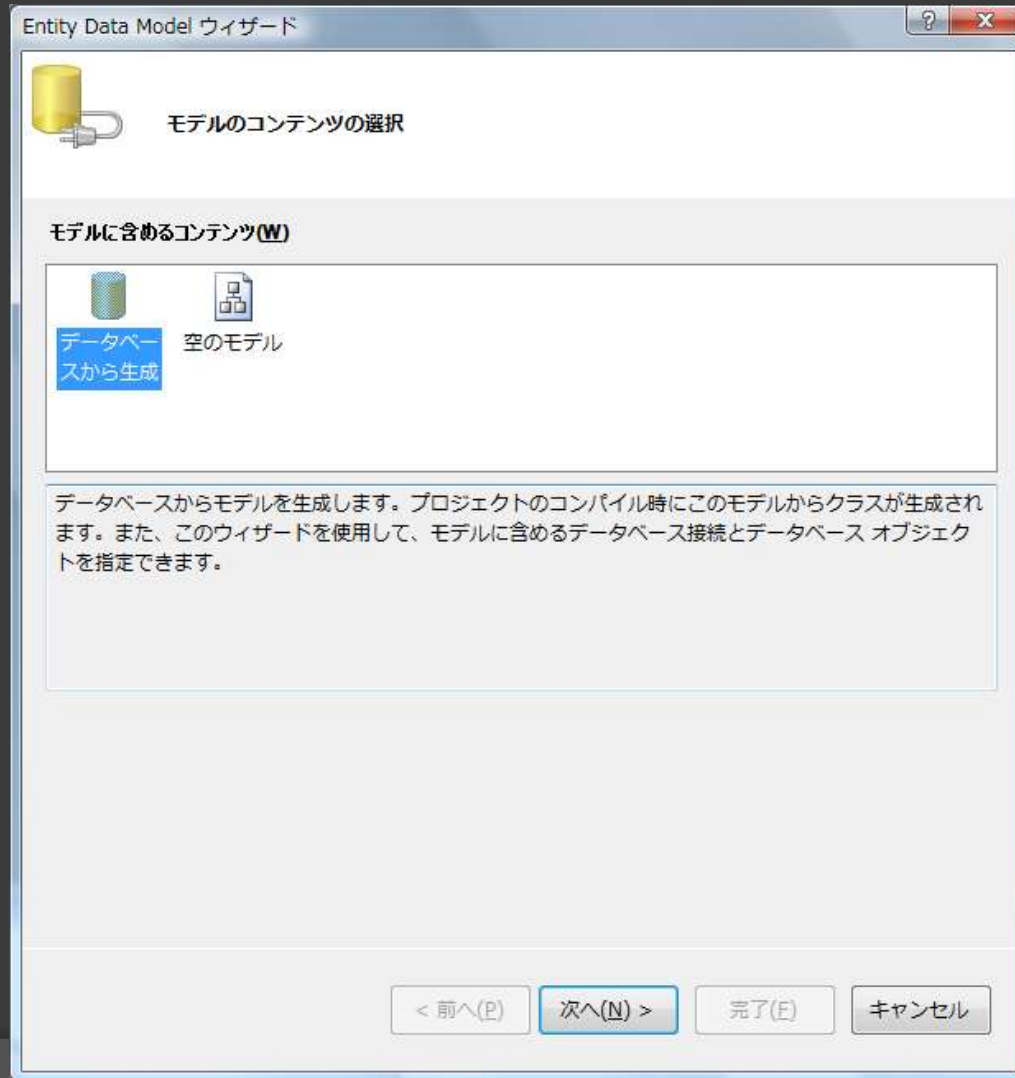


# SQLite+ADO.NET Entity Framework






# SQLite+ADO.NET Entity Framework



# SQLite+ADO.NET Entity Framework

Entity Data Model ウィザード

 データ接続の選択

アプリケーションがデータベースへの接続に使用するデータ接続(W)

SQLite [MMCountdownTimer] 新しい接続(C)...

この接続文字列にはデータベース接続に必要な重要情報 (例: パスワード) が含まれています。接続文字列内に重要情報を保存するとセキュリティ上の問題が発生する可能性があります。接続文字列にこの重要情報を含めますか?

いいえ、重要情報を接続文字列に含めません。この情報はアプリケーション コードで設定します。(E)

はい、重要情報を接続文字列に含めます。(I)

エンティティ接続文字列:

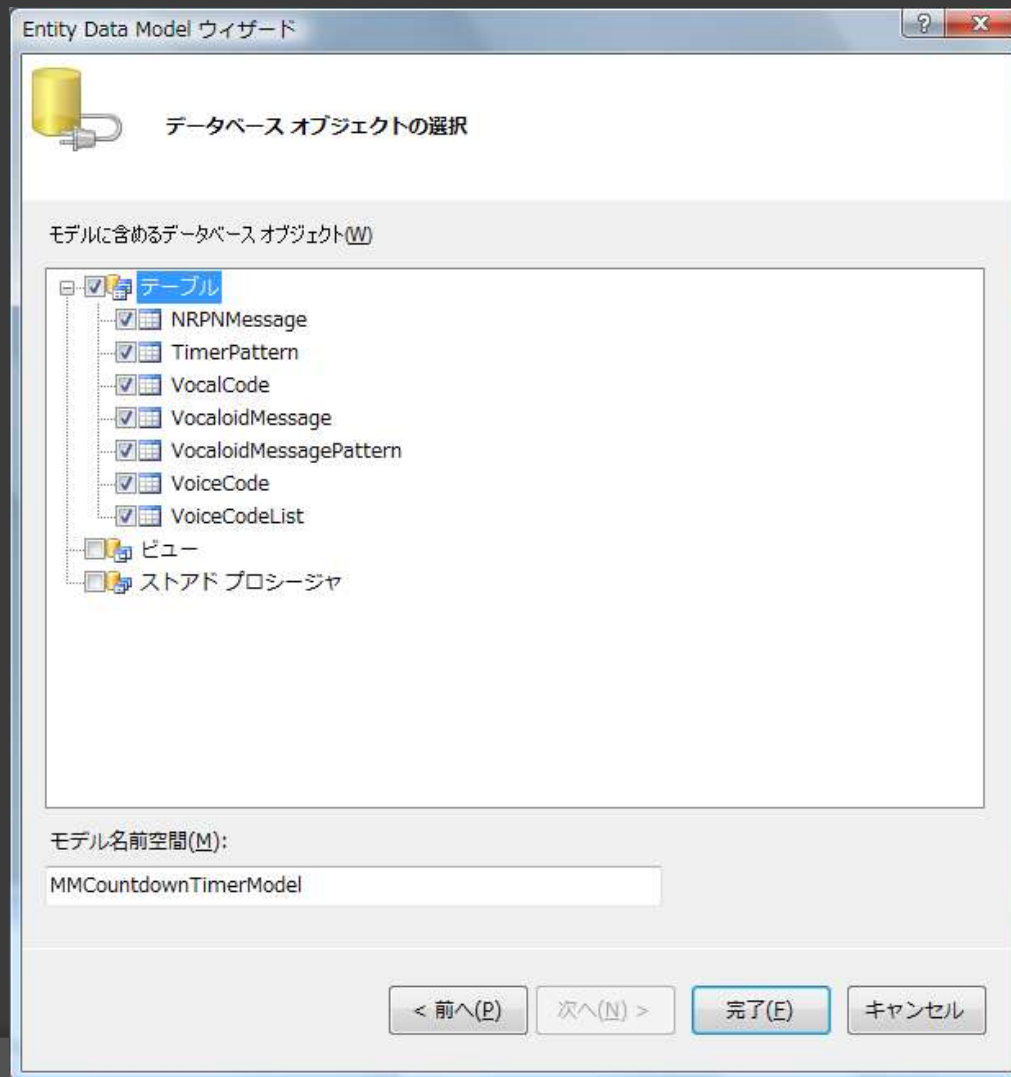
```
metadata=res://*/Model1.csdl|res://*/Model1.ssdl|res://*/Model1.msl;provider=System.Data.SQLite;provider connection string="data source=C:\Users\masa\Documents\csharp\vs2008\MMCountdownTimer\MMCountdownTimer\MMCountdownTimer.db"
```

エンティティ接続設定に名前を付けて App.Config に保存(S):

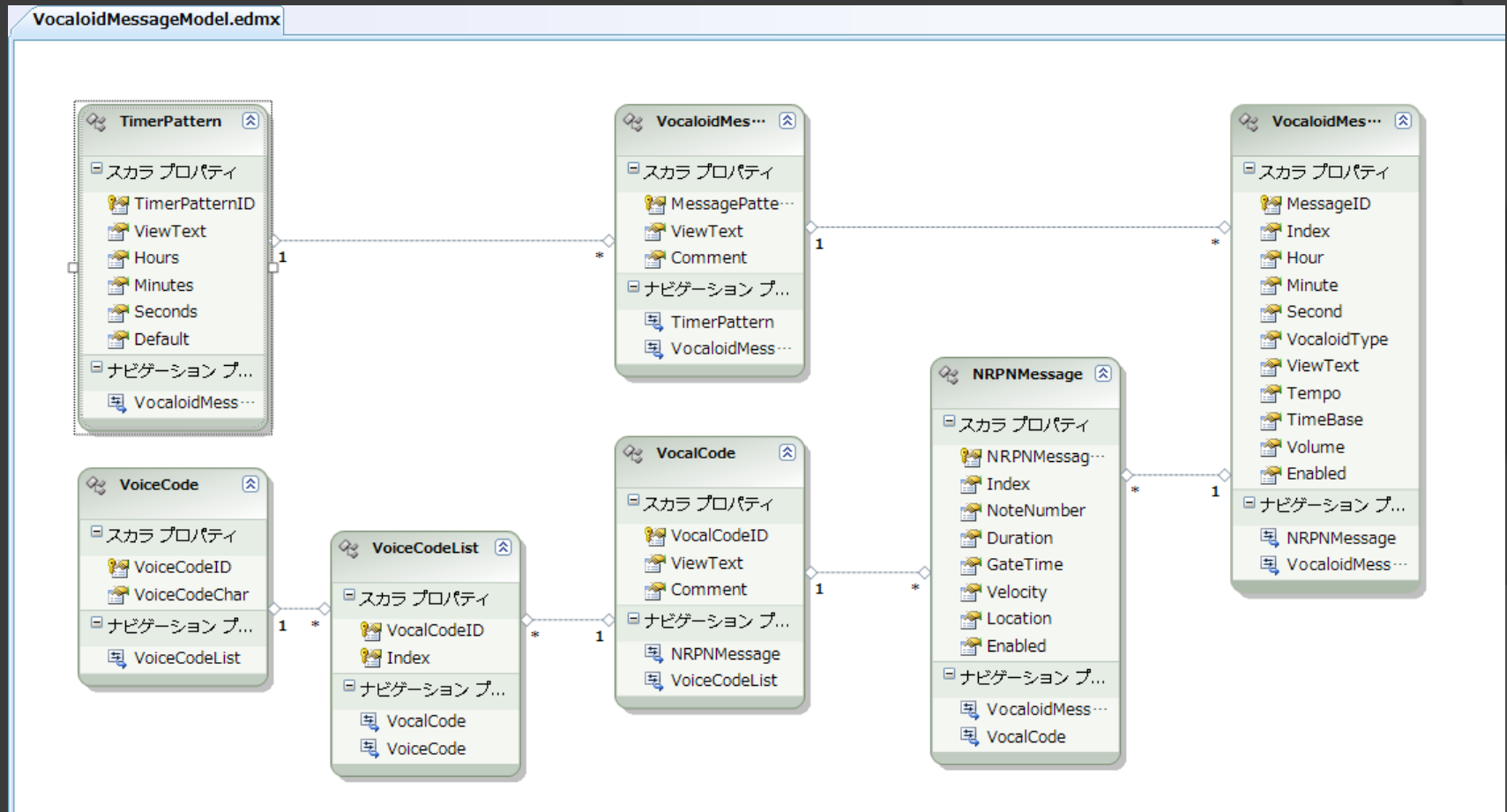
MMCountdownTimerEntities

< 前へ(B) 次へ(N) > 完了(E) キャンセル

# SQLite+ADO.NET Entity Framework



# 使用例 – DB (SQLite)



# 使用例 - O/Rマッピング定義

マッピングの詳細 - VocaloidMessagePattern

列	演算子	値/プロパティ
<ul style="list-style-type: none"> <li>テーブル           <ul style="list-style-type: none"> <li>VocaloidMessagePattern (にマップ)               <ul style="list-style-type: none"> <li>&lt;条件の追加&gt;</li> <li>列マッピング                   <ul style="list-style-type: none"> <li>MessagePatternID : int</li> <li>TimerPatternID : int</li> <li>ViewText : nvarchar</li> <li>Comment : nvarchar</li> <li>&lt;テーブルまたはビューの追加&gt;</li> </ul> </li> </ul> </li> </ul> </li> </ul>	       	       
	↔	MessagePatternID : Int32
	↔	
	↔	ViewText : String
	↔	Comment : String

エラー一覧 | タスク一覧 | マッピングの詳細 | 出力

## 使用例 – LINQ to Entities

```
// Entityのインスタンスを作成
```

```
var entity = new VocaloidMessageEntities();
```

```
// LINQでデータを取得
```

```
var message = from p in entity.VocaloidMessagePattern  
              where p.TimerPattern.TimerPatternID == currentPtnID  
              select p;
```

## 使用例 – Query Builderによる保存

```
// Entityのインスタンスを作成
var entity = new VocaloidMessageEntities();

// Entity内のデータオブジェクトを作成
var message = new VocaloidMessage();
message.MessageID = 100;
message.Hour = 0;
...

// Entityにデータを追加
entity.AddObject("VocaloidMessage", message);

// 更新をストアに反映
entity.SaveChanges();
```

## まとめ

- SQLiteでアプリケーションのデータをDBで管理しよう（新にサーバを用意する必要はない）
- ADO.NET Entity FrameworkでO/Rマッピングを簡単に
- 複雑な処理への対応はV2以降に期待