

「拝啓、さかもとと申します」

2008年6月28日

わんくま同盟 勉強会初参戦。

※最重要 - 質問はご遠慮願います -



SQL Server Compact概要－1

【バージョン】

3.5 SQL Server Compact

3.1 SQL Server Compact
Edition

3.0 SQL Server Mobile Edition

2.0 SQL Server 2000 Windows
CE Edition

【互換性】

3.5→3.1をアップグレード可能

3.0、3.1は互換性あり

2.0は他エディションとの互換性なし(作り直し)

SQL Server Compact概要－2

【Visual Studio バージョン】

3.5 SQL Server Compact

→Visual Studio2008※3.1はアップ
グレードとなる

3.1 SQL Server Compact
Edition

→Visual Studio2005

→Visual Studio2005

3.0 SQL Server Mobile Edition

→??

2.0 SQL Server 2000 Windows
CE Edition

SQL Server Compact暗号化

・暗号化には2つのタイプ(3. 5からの機能)

1:プラットフォーム依存

OS側に暗号化処理を委ねる

Windows XP→CE などは利用できない

2:エンジン依存

SQL Server Compactの機能による

OSに依存されず、XP⇔CEなどでも利用可能

PCとモバイルの連携運用ではエンジン依存暗号化が良い

デバイスへのインストール方法

1: VisualStudioを利用しての自動インストール

2: 手動インストール

2-1: Copmact Framework2.0のインストール

C:¥Program Files¥Microsoft Visual Studio

8¥ SmartDevices¥SDK¥CompactFramework¥2.0¥v2.0¥WindowsCE

¥<platform>¥<processor>¥ NETCFv2.<platform>.<processor>.cab ファイル

2-2: エンジンのインストール

C:¥Program Files¥Microsoft Visual Studio 8¥SmartDevices¥SDK¥SQL

Server¥Mobile¥v3.0¥<platform>¥<processor>¥

sqlce30.platform.processor.cab

sqlce30.repl.platform.processor.cab

デバイス上でCABファイルをクリックしてインストール開始。

クエリアナライザもインストール可能!!!

NETCFv2.<platform>.<processor>.cab ファイル



デスクトップへのインストール方法

1: インストーラーを利用

1-1: Visual Studioでプロジェクトに%Program Files%\Microsoft SQL Server Compact Edition\v3.5\Desktop フォルダにある System.Data.SqlServerCe.dllの参照を追加

2: ClickOnceを利用

3: プライベートファイルベース配置

sqlceca35.dll

sqlcecompact35.dll

sqlceer35EN.dll

sqlceme35.dll

sqlceoledb35.dll

sqlceqp35.dll

sqlcese35.dll

基本的にこのDLLがあれば利用可能

インストールが必要なの？

【メリット】

- ・WindowsUpdateの適用対象となる→管理が楽
- ・デバイスの場合はかなり楽

【デメリット】

- ・管理者権限が必要
- ・そもそも面倒

機能制限

1: 以下はむり。

- ・ストアドプロシージャ
- ・トリガ
- ・ビュー
- ・SQL CLR

2: サイズは4GBまで(Express Editionと同様)

3: 同時セッションは256まで(100までを推奨) → 複数セッションを利用することがあるかが問題

4: デスクトップとデバイスアプリとの差に注意

例) TransactionScopeはデスクトップはOK デバイスはNG(FWの差がそのままと思ってよい)

5: 接続頻度は少ない方がよい

これに反する場合は他のエディションを利用しよう

SQL Server Compact使いどころ

・SQL Server Compact を採用する基準は？

- ・接続頻度が少ないアプリで(特に更新系)
- ・エンジンごと自由に配布したいとき
- ・配布サイズをひたすら小さくしたいとき
- ・プロセス内でアプリを完結させたいとき

実務で使うには

1: 私はここで使った 2007年春

- ・インターネットへの接続不可→プライベートファイルベース配置でいいや
- ・スタンドアロンシステム
- ・更新頻度が少ない
- ・担当者が簡単にファイルをコピーしてデータ移動できること

2: 私はここであきらめた 2008年春

- ・デバイスメーカーによるサポート不可
- デバイスアプリではまだまだC++とフラットファイルでの運用が主流

