



INFORMATION ANYWHERE

iAnywhere
A SYBASE COMPANY

Technical Guide

SQL Anywhere Studioの Visual Studioへの統合



目次

はじめに	3
サーバーエクスプローラのプラグイン	4
データの表示と修正	4
ASA統合コンポーネント	9
ASA統合コンポーネントの追加	10
AsaDataAdapterコンポーネントの利用	11
アプリケーションによるAsaDataAdapterの使用	18
AsaDataAdapterコンポーネントを使うようアプリケーションを構成する	18
ASA Data Adapterのプロパティの確認と変更	20
データセットの生成	22
データのプレビュー	25
法的注意	26

はじめに

ソフトウェアバージョン

本書の内容は、SQL Anywhere Studioのバージョン9.0.2を対象にしています。

SQL Anywhere Studio 9.0.2は、Microsoft社のVisual Studio.NET 2003との統合機能を数多く備えています。これらの機能は、Visual Studioによるアプリケーション開発でSQL Anywhere Studioデータベースを使いやすくするためのものです。

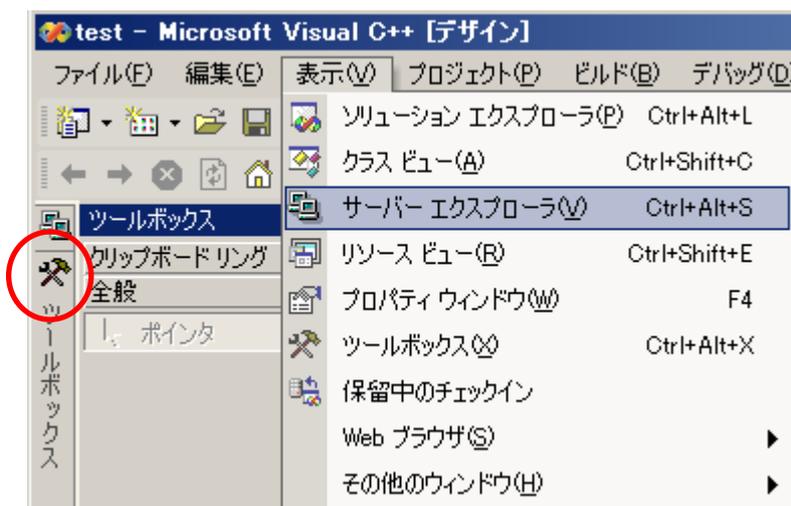
本書では、Visual Studioによるデータベース利用のアプリケーション開発で、SQL Anywhere Studio 9.0.2の統合機能を活用する手順を説明します。

サーバーエクスプローラのプラグイン

Visual Studioのサーバーエクスプローラにより、データベースのスキーマや格納されているデータなどの情報を表示できます。またデータベース・テーブル中のデータを変更することもできます。

データの表示と修正

1. Visual Studioで [表示] → [サーバーエクスプローラ] を選択してください。



サーバーエクスプローラがツールボックスに代わって画面左側に表示されます。

NOTE

ツールボックスに戻る場合は、[ツールボックス] アイコン（赤で表示）をクリックしてください。サーバーエクスプローラをVisual Studioの画面横に出した場合は、底部に表示される [ツールボックス] タブをクリックしてください。



2. [データ接続] を右クリックして [接続の追加] を選択してください。



[データ リンク プロパティ] ダイアログが表示されます。

3. [プロバイダ] タブで [Sybase Adaptive Server Anywhere Provider 9.0] を選択し、[次へ] をクリックしてください。



4. [接続] タブで [データソース] フィールドにデータベースのODBCデータ名を入力してください。この例ではデータソースとして [ASA 9.0 Sample] と入力します。

5. [特定のユーザー名とパスワードを使用する] ラジオボタンをチェックし、[ユーザー名] フィールドに「dba」、[パスワード] フィールドに「sql」とそれぞれ入力してください。
6. 必要に応じて [詳細設定] タブと [すべて] タブに情報を入力してください。以下に入力が終了した状態の [接続] タブを示します。

データリンク プロパティ

プロバイダ 接続 詳細設定 すべて

このデータに接続するために、次の項目に情報を入力します:

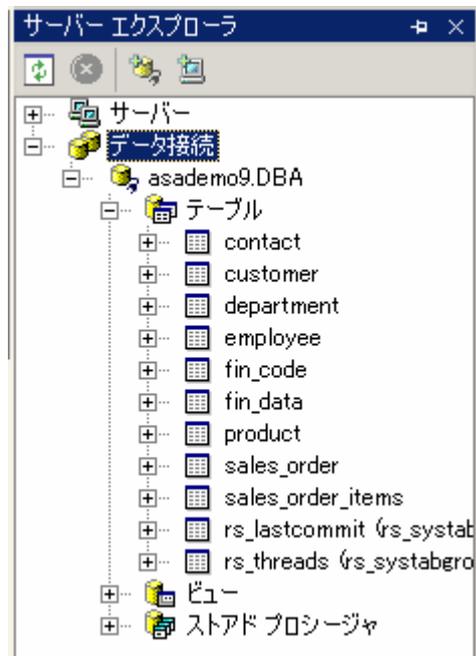
1. データソースとデータの場所を入力します:
データソース(D): ASA 9.0 Sample
場所(L):
2. サーバーへのログオンに必要な情報を入力します:
 Windows NT の統合セキュリティを使用する(W)
 特定のユーザー名とパスワードを使用する(U):
ユーザー名(N): dba
パスワード(P): ***
 パスワードを空にする(B) パスワードを保存する(S)
3. 使用する初期カタログを入力します(C):

接続のテスト(T)

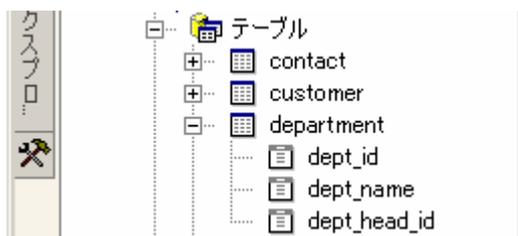
OK キャンセル ヘルプ

7. [OK] をクリックして接続を追加してください。サーバーエクスプローラに新しいエントリ「asademo9.DBA」が表示されます。これが新たにユーザー名「dba」で接続したデータベースとなります。

8. 「asademo9.DBA」のとなりの「+」をクリックしてエントリーを展開し、続いて以下に示すように「テーブル」エントリーを展開してください。



9. サーバーエクスプローラではデータベース中のすべてのテーブルを表示できます。任意のテーブル中のスキーマを見たい場合は、その名前となりの「+」をクリックしてください。たとえば [department] テーブルを展開してそのスキーマを見るには、そのとなりの「+」をクリックしてください。



[department] テーブルには [dept_id]、[dept_name]、[dept_head_id] の三つの列があることがわかります。テーブル中のデータを見るにはテーブル名をダブルクリックしてください。データが Visual Studio のメインエリアに表示されます。セル中のデータを変更するにはセルをクリックしてください。セル中のデータの変更は、自動的にデータベースに登録されます。

10. 新しい列のデータをテーブルに加えるには、以下に示すように、その左側に「*」（アスタリスク）を持つ行のセルにデータを入力してください。入力されたデータは、データベースにただちに追加されます。

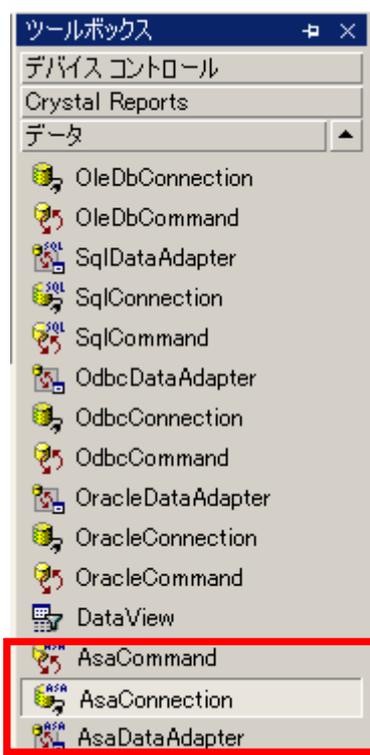
	dept_id	dept_name	dept_head_id
▶	100	R & D	501
	200	Sales	902
	300	Finance	1293
	400	Marketing	1576
	500	Shipping	703
*			

ASA統合コンポーネント

ASA統合コンポーネントを使うことで、アプリケーション設計者はVisual Studio開発環境およびそのコードからデータベースと対話できるようになります。

ASA統合コンポーネントには、AsaDataAdapter、AsaConnection、AsaCommandがあります。これらのコンポーネントのうち、本書のテーマと特に関係するのはAsaDataAdapterです。AsaDataAdapterにより、アプリケーション設計者はデータベースから結果セットを容易に取り出し、それをDataSetオブジェクトに格納することができます。DataSetオブジェクト中のデータは編集可能となっており、AsaDataAdapterは編集後のデータベースを自動的にアップデートします。

ASA統合コンポーネントを表示するには、ツールボックスの「データ」セクションをクリックします。



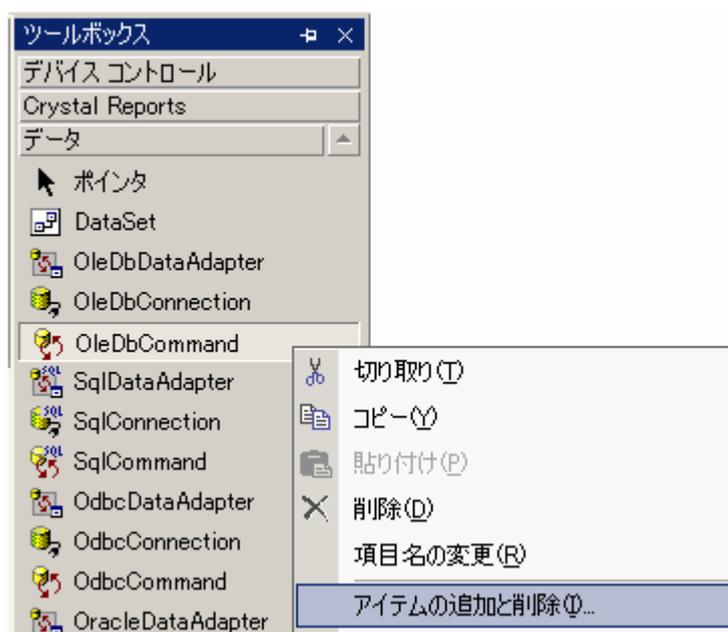
上記の図中で赤で囲んだものがASA統合コンポーネントです。これらのコンポーネントが表示されない場合は、以下の[ASA統合コンポーネントの追加](#)手順に従い、それらを手動で追加してください。

ASA統合コンポーネントの追加

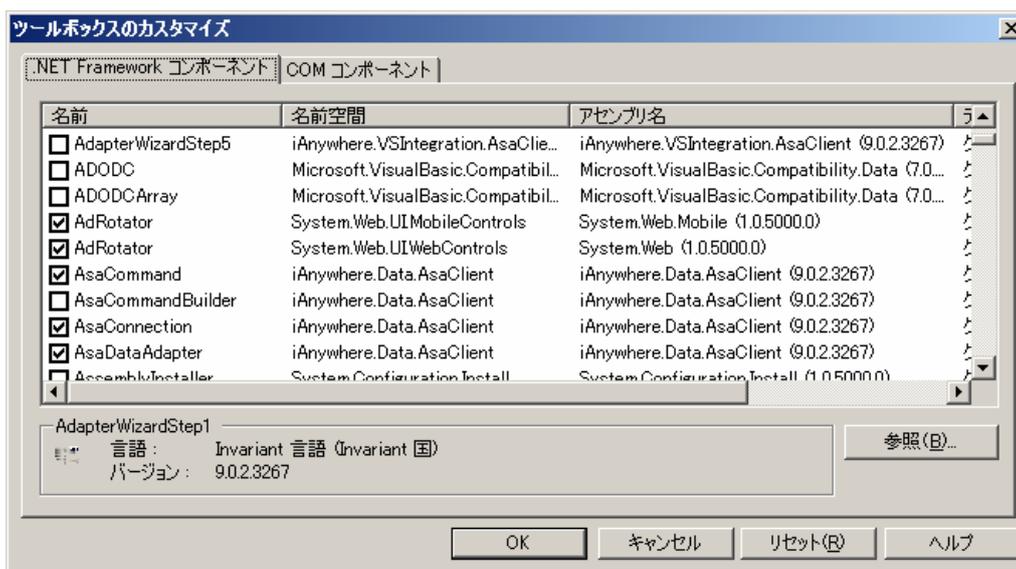
NOTE

ツールボックスの [データ] セクションにすでにASAコンポーネントが表示されている場合は、この手順は必要ありません。

1. ASA統合コンポーネントを追加するには、ツールボックスのいずれかの場所を右クリックし、以下に示すように [アイテムの追加と削除] を選択してください。



2. [ツールボックスのカスタマイズ] ダイアログが表示されます。AsaCommand、AsaConnection、AsaDataAdapterのとなりのチェックボックスにチェックを入れ、[OK] をクリックしてください。

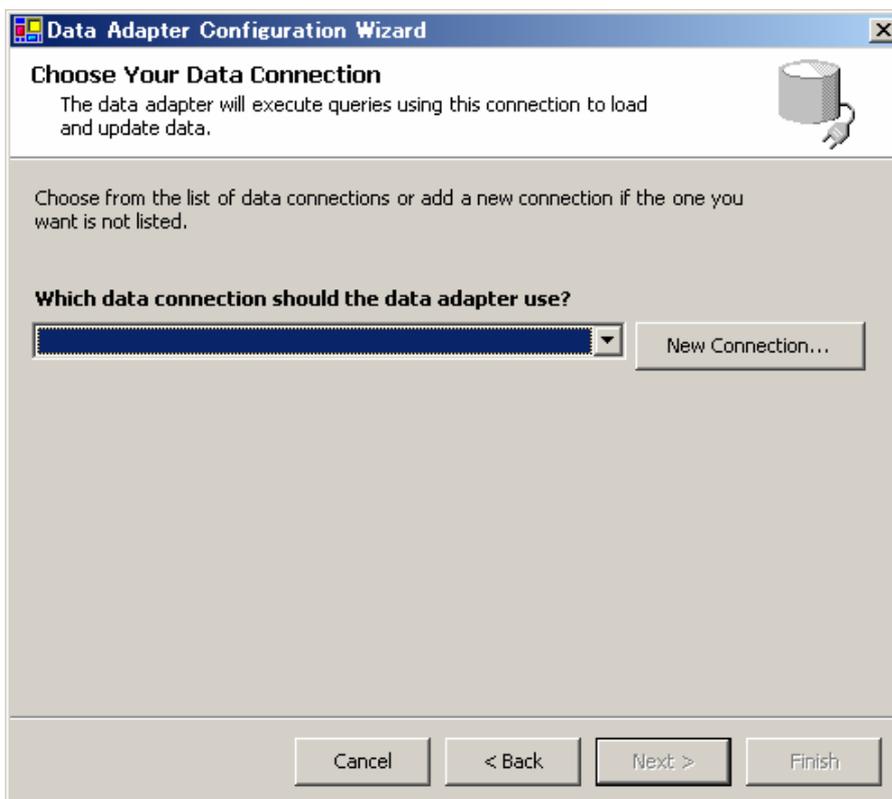


AsaDataAdapterコンポーネントの利用

1. ツールボックスから [AsaDataAdapter] コンポーネントを選択し、フォーム上にドラッグ&ドロップしてください。[asaDataAdapter1] オブジェクトがキャンパスの下部に現れ、同時に [データ・アダプタ設定] ウィザードが表示されます。

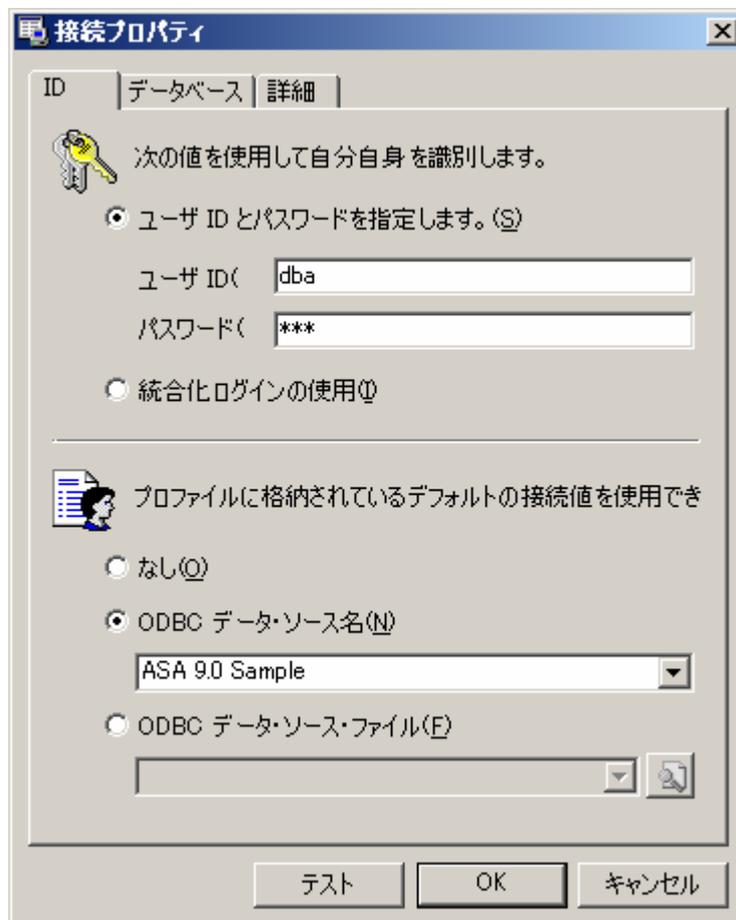


2. [次へ]をクリックしてください。AsaDataAdapterが使用する接続を選択する画面が表示されます。



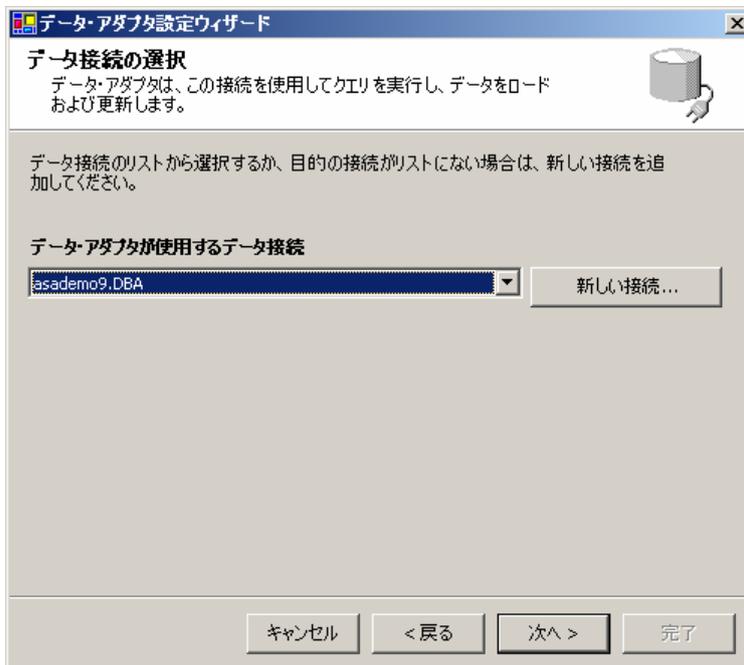
3. [新しい接続...]をクリックして新しい接続を開始してください。Sybase Centralでデータベースに接続する際と同様のダイアログが表示されます。

- 以下に示すように、[ユーザID] フィールドと [パスワード] フィールドに「dba」と「sql」をそれぞれ入力し、さらに [ODBCデータソース名] として「ASA 9.0 Sample」と入力してください。

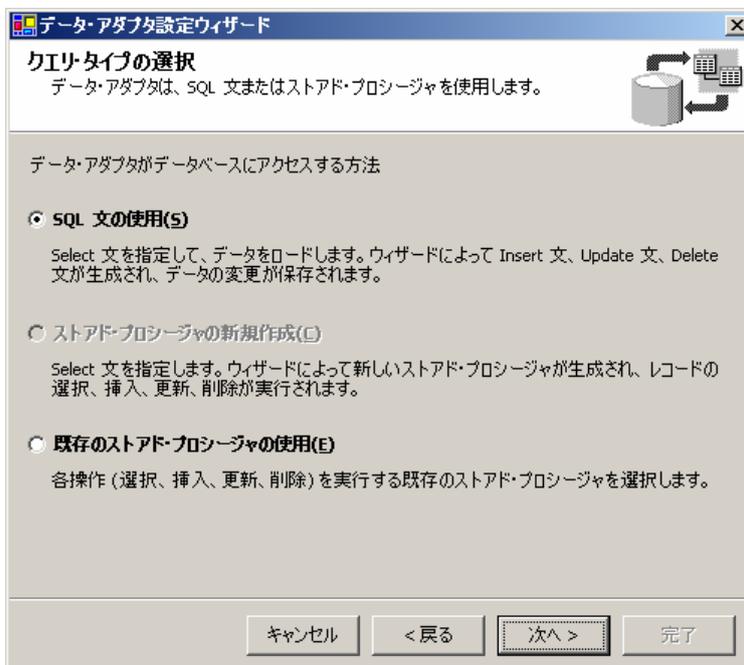


- [テスト] をクリックして接続を試験してください。すべての入力が正しければ確認ダイアログが表示されます。

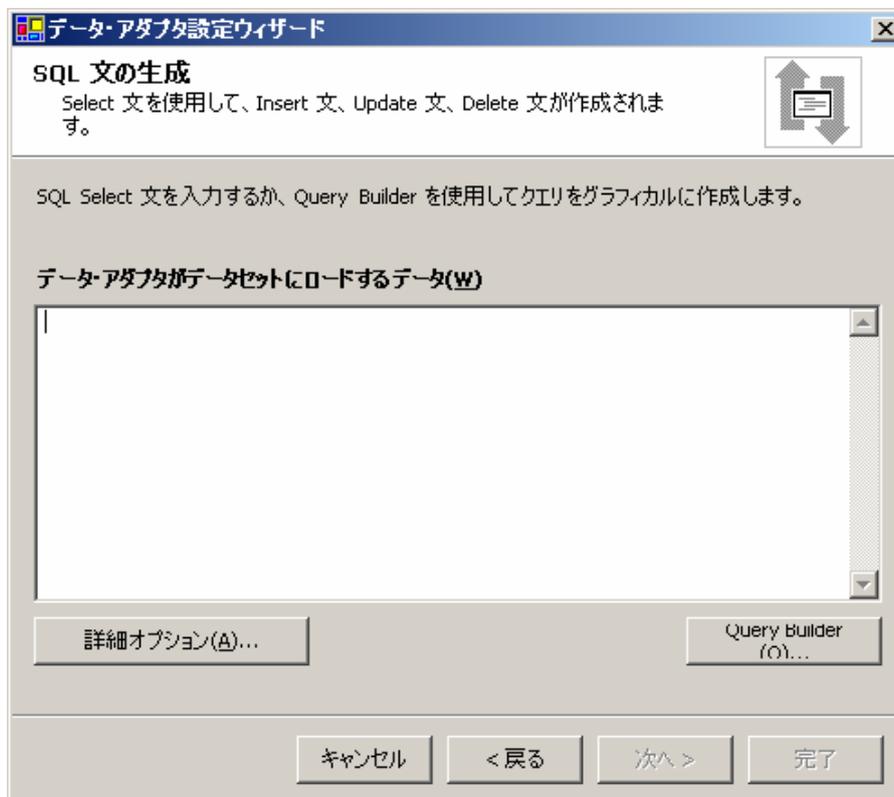
6. [OK] をクリックして接続を確認し、[データ・アダプタ設定] ウィザードに戻ってください。選択された新しい接続が以下のように表示されます。[次へ] をクリックしてください。



7. AsaDataAdapterがデータベースにアクセスする方法の選択を促す画面が表示されます。デフォルト・オプションの [SQL文の使用] を選択し、[次へ] をクリックしてください。



8. データをデータアダプタにロードするためのSQL文を指定する画面が表示されます。[Query Builder] をクリックしてください。

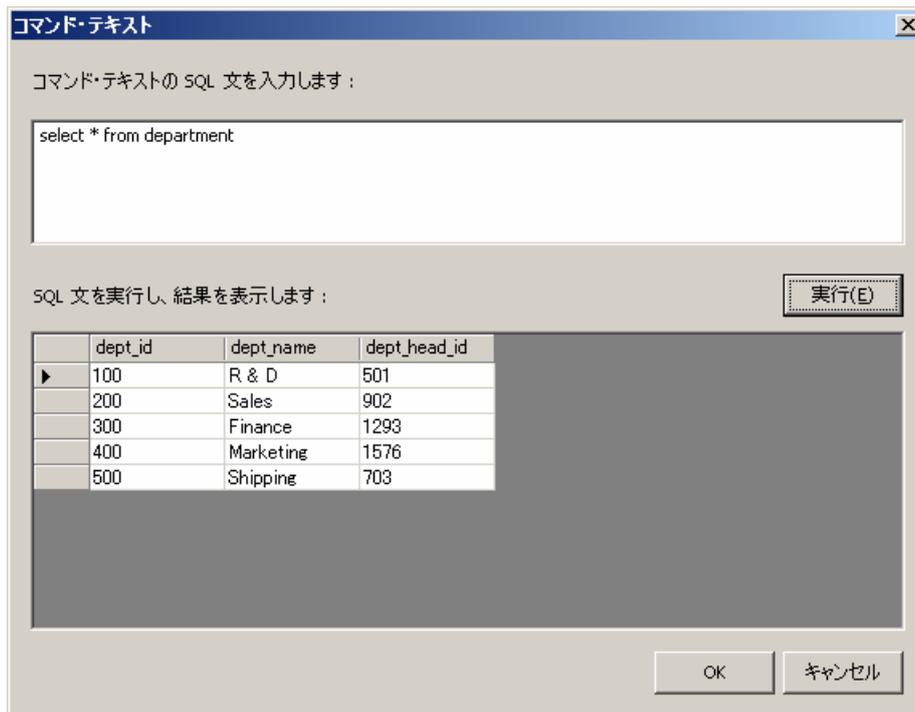


9. SQL文が正しいか確認するための [コマンド・テキスト] ダイアログが起動されます。この例では「department」というテーブルの内容をデータセットにロードすることにします。次の文を入力して、[実行] をクリックしてください。

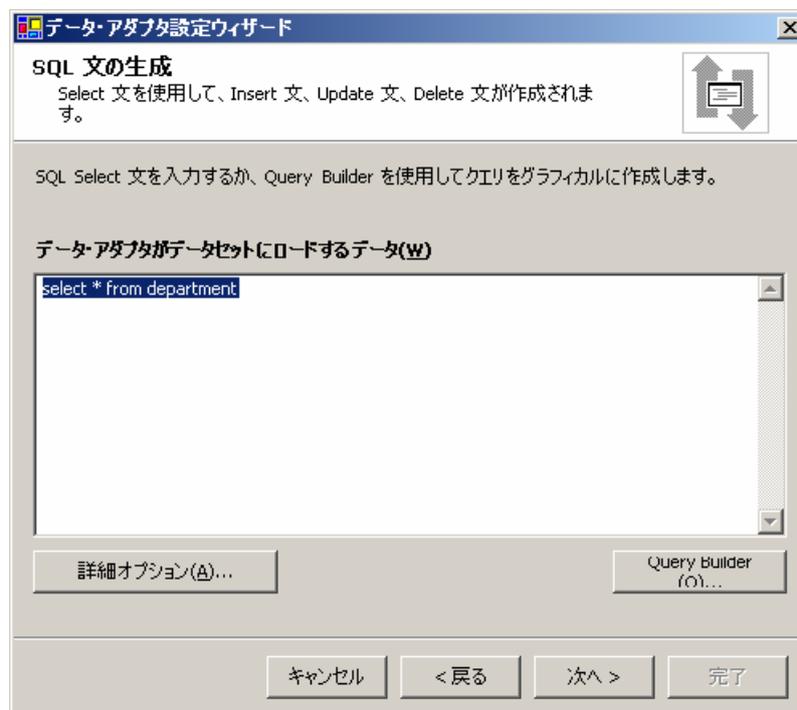
```
select * from department
```

クエリ結果としてdepartmentテーブルのデータが表示されます。

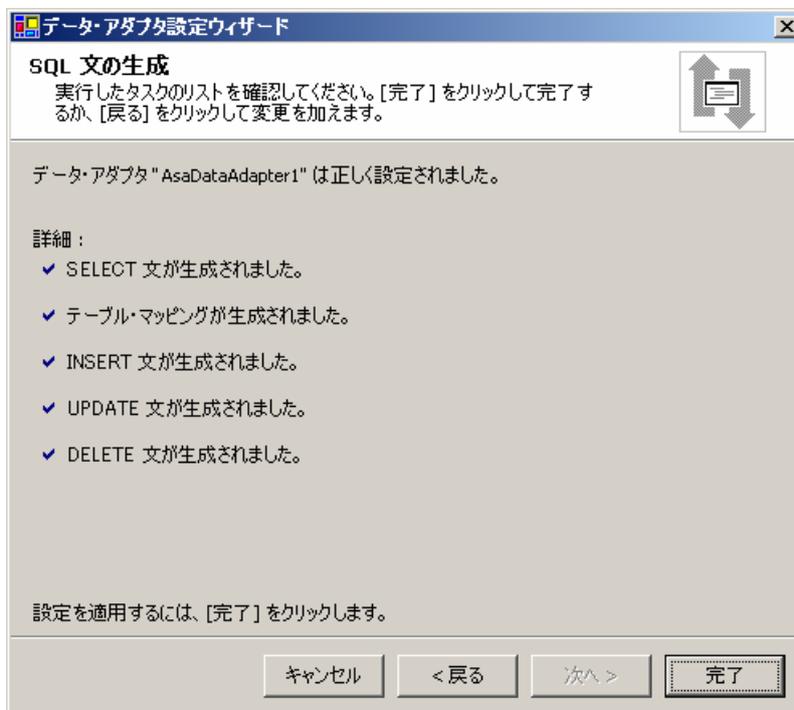
NOTE 他の目的でデータセットをロードする際には異なるSQLクエリを使うことになります。



10. [OK] をクリックしてください。[コマンド・テキスト] ダイアログに記述したクエリがテキストボックスに表示されます。



11. [次へ] をクリックしてください。データアダプタの生成結果を示す画面が表示されます。生成処理が成功した場合、ダイアログは次に示すようなものとなるはずです。



12. [完了] をクリックしてウィザードを閉じてください。

アプリケーションによるAsaDataAdapterの使用

前節までの手順でAsaDataAdapterオブジェクトが生成されたので、それをアプリケーションで使ってみましょう。最も簡単なのはDataGridオブジェクトを利用する方法です。

NOTE

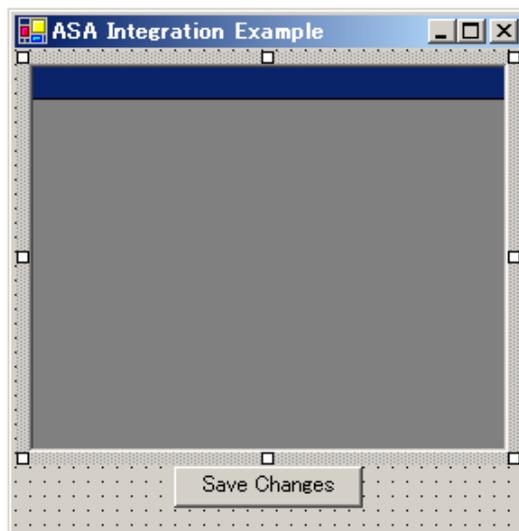
AsaDataAdapterを使う場合は、データベース中の行はロックされません。つまりデータセットからデータベースへ変更を反映するときに矛盾が発生する可能性があります。したがってアプリケーションは発生した矛盾を解決するかあるいは記録するロジックを備えるべきです。この問題については製品に付属の文書にさらに多くの情報が記載されています。

AsaDataAdapterコンポーネントを使うようアプリケーションを構成する

1. プロジェクトを開き、ツールボックス中で [Windows フォーム] をクリックし、 [DataGrid] を選択してください。



2. [DataGrid] をフォーム上にドラッグしてください。また [Button] を同様にしてフォーム上へとドラッグし、その [Text] を [Save Changes] に変更してください。フォームは以下のようになるはずですが。



今はAsaDataAdapterにデータがロードされ、フォーム上にDataGridが置かれた状態ですので、AsaDataAdapterで生成したデータセットをDataGridのコンテンツにリンクできます。

3. フォームのグローバル宣言に以下のコードを記述してください。

[C#の場合]

```
DataSet ds = new DataSet();
```

[Visual Basic.NETの場合]

```
Dim ds As New DataSet
```

4. フォームのLoadメソッドに以下のコードを記述してください。

[C#の場合]

```
asaDataAdapter1.Fill(ds);  
dataGrid1.DataSource = ds;
```

[Visual Basic.NETの場合]

```
AsaDataAdapter1.Fill(ds)  
DataGrid1.DataSource = ds
```

5. AsaDataAdapterがその結果をセーブするには、アップデートされたデータをセーブするためのボタンを生成する必要があります。以下のコードをボタンのClickメソッドに記述してください。

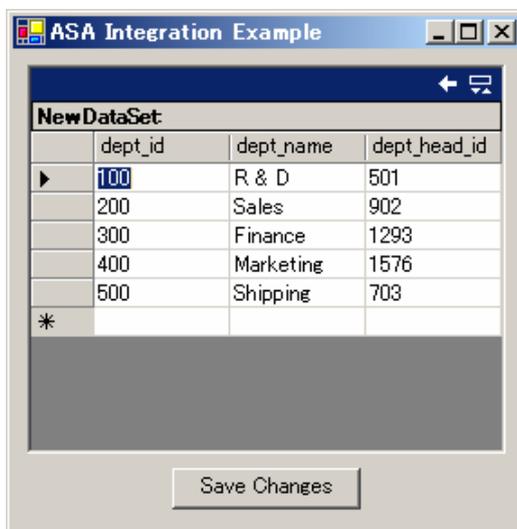
[C#の場合]

```
asaDataAdapter1.Update(ds);
```

[Visual Basic.NETの場合]

```
AsaDataAdapter1.Update(ds)
```

6. コードが記述されたので、アプリケーションを起動してください。フォーム上のデータは編集可能な状態です。[Save] をクリックすることで、変更をデータベースへ自動的に保存できます。以下の図は稼働中のアプリケーションの様子を示すものです。



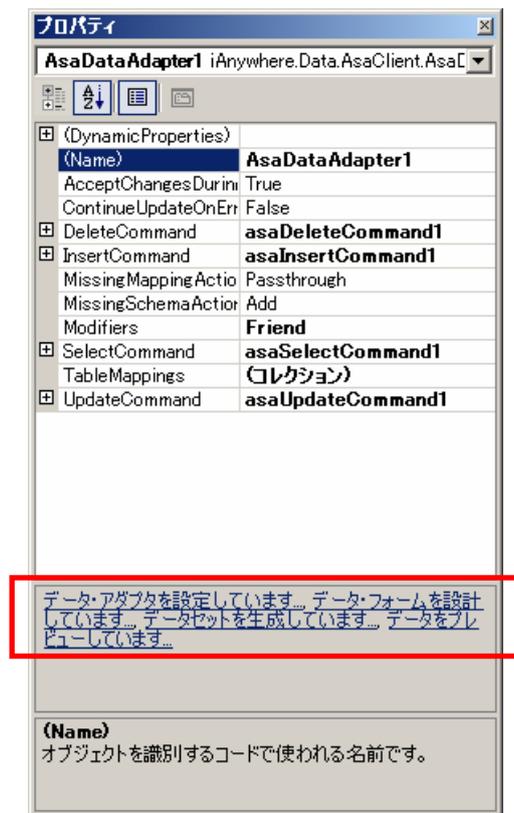
ASA Data Adapterのプロパティの確認と変更

AsaDataAdapterを生成した後、ビルトインのウィザードを使ってカレントの設定を確認したり、アップデートして動作を変更することが可能です。以下の例はアプリケーション上にAsaDataAdapterを生成済みであることを前提としています。まだ生成していない場合は、上述の各手順を実行してAsaDataAdapterを生成してください。

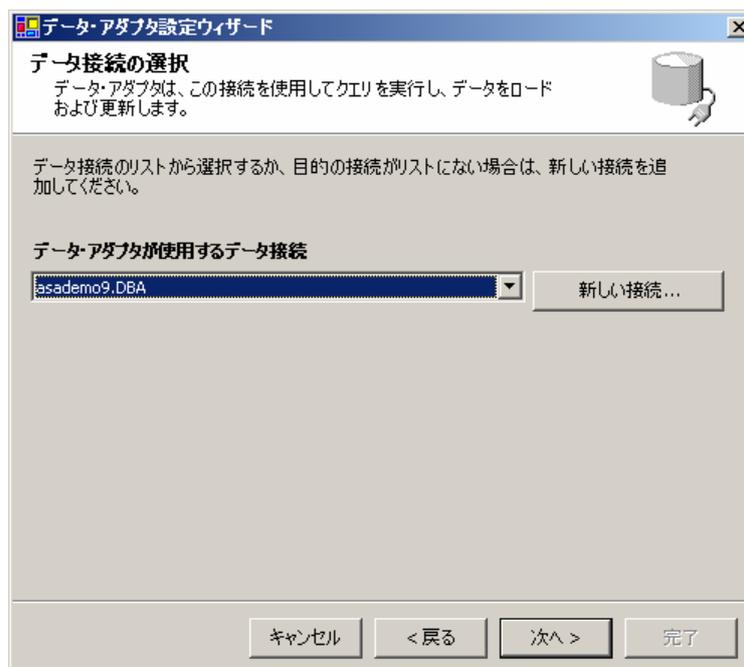
1. [フォームデザイン] ウィンドウの下部で [AsaDataAdapter1] オブジェクトをクリックしてください。



オブジェクトのプロパティが [プロパティ] ウィンドウに表示されます。このプロパティは手動で編集可能です。また [プロパティ] ウィンドウの下部には設定変更用のウィザードを開くリンクがあります。以下の図にリンクを赤枠で示します。



2. [データ・アダプタを設定しています...] リンクをクリックしてください。データアダプタ設定ウィザードが起動します。ウィザードからはデータアダプタをプロジェクトにドロップした時点のすべての初期設定が変更できます。
3. [次へ] をクリックしてください。使用するデータベース接続を指定する画面が表示されます。



すでにサンプルのデータベースへの接続が存在するので、別の接続を選択する必要はありません。
 [次へ] をクリックして先に進んでください。特に別のデータベースを使うことを望む場合は [新しい接続] をクリックして新しい接続を生成し、接続情報を入力してください。

4. データベースへのアクセスにSQL文と既存のストアドプロシージャのどちらを使うかをたずねる画面が表示されます。デフォルトオプションの [SQL文の使用] を選択し [次へ] をクリックしてください。
5. データアダプタがデータベース中から選択するデータを指定するための画面が表示されます。デフォルトとして直前に記述したクエリ [SELECT * FROM department;] が表示されるので、以下に示すように、文を [SELECT * FROM employee;] に変更してください。



6. [クエリ ビルダ] をクリックし、クエリが正しいことを確認してください。正しい場合は [次へ] をクリックし、その後 [完了] をクリックして変更を確定してください。データアダプタはアプリケーションが使うデータをアップデートします。

データセットの生成

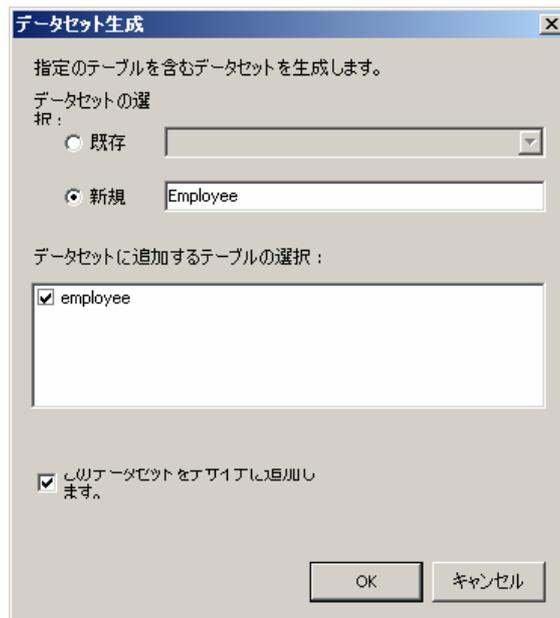
[データセットの生成] リンクから、XMLスキーマ定義ファイル (.xsd) を生成できます。XMLスキーマ定義ファイルは、データベースの構造を示す標準的な方法です。このファイルはDataSetオブジェクトを生成するためにアプリケーションに簡単に配置できます。

以下にデータセットの生成手順を示します。

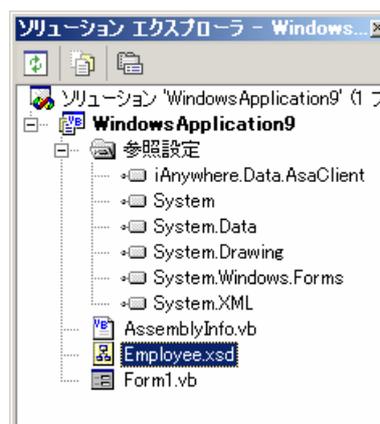
1. AsaDataAdapter1の [プロパティ] ウィンドウの下部にある [データセットを生成しています] リンクをクリックしてください。
2. [データセットの生成] ダイアログが表示されます。既存のデータセットか新規作成のいずれかを指

定できます。この例ではまだデータセットを作成していないので [新規作成] を選択し、データセットの名前として「Employee」を指定します。

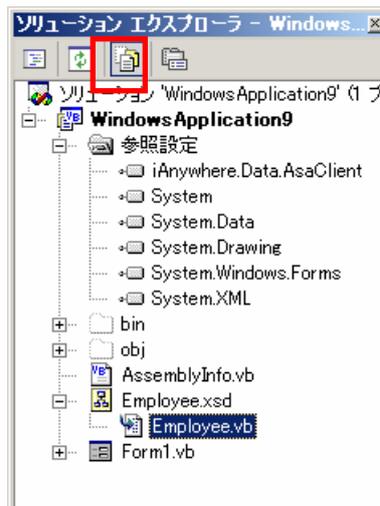
3. [データセットに追加するテーブルの選択] フィールド内には [employee] テーブルしかありませんので、これをチェックしてください。これは現状でAsaDataAdapterが参照する唯一のテーブルです。AsaDataAdapter用にカスタムクエリをあらかじめ設定しておけば、アクセスできるテーブルの一覧は異なるはずです。



4. [OK] をクリックしてください。生成された.xsdファイルがソリューションエクスプローラに以下のように表示されます。



ソリューションエクスプローラで [すべてのファイルを表示] アイコン (下図に赤枠で表示) をクリックすると自動的に生成されたコードを表示することができます。



また [Employee.cs] (Visual Basicユーザーの場合は [Employee.vb]) をダブルクリックしても、生成されたコードが表示されます。

5. DataSetオブジェクトを新たに生成したので、データベーススキーマで事前に設定したデータセットを生成できるようになりました。たとえば [ss_number] 列のタイプを知りたい場合は、以下のコードを記述してください。

[C#の場合]

```
Employee myEmployeeDataSet = new Employee();
MessageBox.Show(myEmployeeDataSet.employee.ss_
    numberColumn.DataType.ToString());
```

[Visual Basic.NETの場合]

```
Dim myEmployeeDataSet As New Employee
    MessageBox.Show(myEmployeeDataSet.employee.ss_
        numberColumn.DataType.ToString)
```

上述のコード中 [myEmployeeDataSet] はデータセット名、[employee] はアクセスしたいテーブル、[ss_numberColumn] は情報を得たい列の名称となります。

5. 同様にDataSetオブジェクトを使ってデータベースのデータを保持することもできます。前述の例で使用したDataGridにデータを埋め込むには、以下のコードを記述してください。

[C#の場合]

```
Employee myEmployeeDataSet = new Employee();
asaDataAdapter1.Fill(myEmployeeDataSet);
dataGridView1.DataSource = myEmployeeDataSet;
```

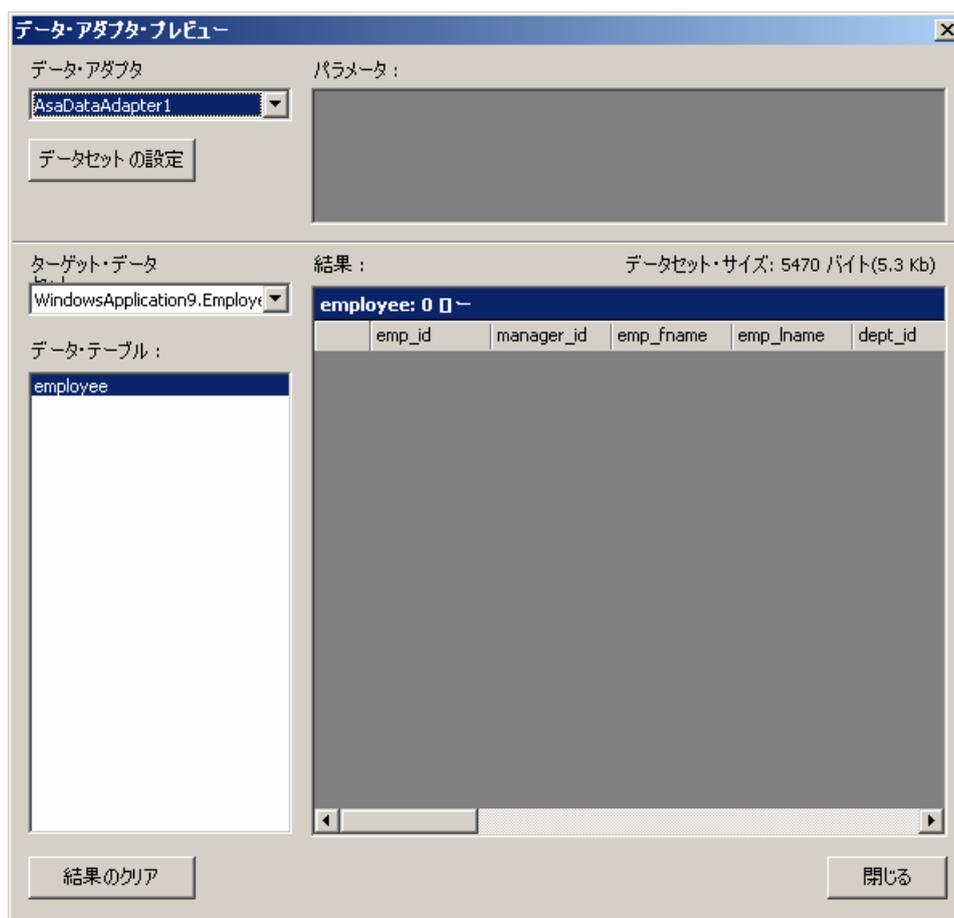
[Visual Basic.NETの場合]

```
Dim myEmployeeDataSet As New Employee
AsaDataAdapter1.Fill(myEmployeeDataSet)
DataGridView1.DataSource = myEmployeeDataSet
```

データのプレビュー

AsaDataAdapter1のプロパティエリアの [データをプレビューしています...] リンクを使えば、AsaDataAdapterが返すデータを見ることができます。リンクをクリックすると [データアダプタ プレビュー] ダイアログが開くので、使いたいデータアダプタと返してほしいテーブルを選択してください。 [データセットの格納] をクリックするとメインウィンドウにデータが出力されます。

データのプレビューは、アプリケーション用に選択したデータが適切かどうかを確認するための優れた方法です。



法的注意

Copyright(C) 2006 iAnywhere Solutions, Inc. All rights reserved.

iAnywhere、iAnywhere Solutions、iAnywhere Solutions(ロゴ)、Adaptive Server、SQL Anywhereは iAnywhere Solutions, Inc.またはSybase, Inc.とその系列会社の米国または日本における登録商標または商標です。その他の商標はすべて各社に帰属します。

Mobile Linkの技術には、Certicom, Inc.より供給を受けたコンポーネントが含まれています。これらのコンポーネントは特許によって保護されています。

本書に記載された情報、助言、推奨、ソフトウェア、文書、データ、サービス、ロゴ、商標、図版、テキスト、写真、およびその他の資料(これらすべてを"資料"と総称する)は、iAnywhere Solutions, Inc.とその供給元に帰属し、著作権や商標の法律および国際条約によって保護されています。また、これらの資料はいずれも、iAnywhere Solutions, Inc.とその供給元の知的所有権の対象となるものであり、iAnywhere Solutions, Inc.とその供給元がこれらの権利のすべてを保有するものとします。

資料のいかなる部分も、iAnywhere Solutionsの知的所有権のライセンスを付与したり、既存のライセンス契約に修正を加えることを認めるものではないものとします。

資料は無保証で提供されるものであり、いかなる保証も行われません。iAnywhere Solutionsは、資料に関するすべての陳述と保証を明示的に拒否します。これには、商業性、特定の目的への整合性、非侵害性の黙示的な保証を無制限に含みます。

iAnywhere Solutionsは、資料自体の、または資料が依拠していると思われる内容、結果、正確性、適時性、完全性に関して、いかなる理由であろうと保証や陳述を行いません。Sybaseは、資料が途切れていないこと、誤りがないこと、いかなる欠陥も修正されていることに関して保証や陳述を行いません。ここでは、「iAnywhere Solutions」とは、iAnywhere Solutions, Inc.またはSybase, Inc.とその部門、子会社、継承者、および親会社と、その従業員、パートナー、社長、代理人、および代表者と、さらに資料を提供した第三者の情報元や提供者を表します。



アイエニウェア・ソリューションズ株式会社

<http://www.ianywhere.jp/>