

ComponentOne Studio 共通ヘルプ

2017.01.24 更新

グレースィティ株式会社

Copyright© GrapeCity, Inc. All rights reserved.

目次

ComponentOne Studio 共通ヘルプ	1
共通ヘルプについて	1
使用上のご注意	2
必要システム	3
インストールとアンインストール	3
ライセンス認証	4
アプリケーション開発の手順	4
プロジェクトの作成	5
Windows フォームアプリケーション	6
ASP.NET Web アプリケーション	7
XAML と XAML 名前空間	9
WPF プロジェクトの作成	11
Silverlight プロジェクトの作成	13
Windows ストアプロジェクトの作成	16
ライセンスの組み込み	18
Windows フォーム用コントロール	19
ASP.NET Web フォーム用コントロール	25
ASP.NET MVC 用コントロール	28
WPF 用コントロール	29
配布ガイド	32
ランタイムファイルの配布について	32
コンポーネントの登録について	33
Windows フォームアプリケーションの配布	34
ASP.NET Web アプリケーションの配布	43
WPF アプリケーションの配布	49
Silverlight アプリケーションの配布	57
Windows ストアアプリケーションの配布	59
移行ガイド	59
インストーラの使用	61
ComponentOne Studio Live の使用	61
ComponentOne Studio プロジェクト移行ツール	66

チャートデータ移行ツール.....	67
C1Report 移行ツール.....	68
関連製品との共存	70
互換性について	71
サンプルについて	72

ComponentOne Studio 共通ヘルプ

このたびは、本製品をお求めいただき、誠にありがとうございます。

ComponentOne Studio は、高度なソフトウェア構築を強力に支援するコンポーネントスイートです。Windows 系システム開発に必要なコンポーネントがすべて揃っており、少ない手間と短い時間で、高機能で洗練されたアプリケーションを構築できます。最上位エディションは Windows フォーム、ASP.NET Web フォーム、ASP.NET MVC、WPF、Silverlight の複数プラットフォームに対応。**ComponentOne Studio** を 1 つ用意しておけば、ユーザーからの多様な要求にもすばやく応えられます。

ComponentOne Studio が皆さまのお役に立てることを開発スタッフ一同心より願っております。

また、弊社ではツールライブラリをさらに充実させる計画を進めております。新しい機能や新しいツールに関するご意見やご要望がございましたら、弊社 Web サイトのフィードバックフォーム

(<http://c1.grapecity.com/SuperPages/contact/>) よりお寄せいただければ幸いです。



グレープシティ株式会社

〒981-3205

宮城県仙台市泉区紫山 3-1-4

<http://c1.grapecity.com/>

共通ヘルプについて

このセクションでは、本製品のご使用にあたって、はじめにお知りいただきたい内容について説明します。

- [使用上のご注意](#)
- [必要システム](#)
- [インストールとアンインストール](#)
- [ライセンス認証](#)
- [アプリケーション開発の手順](#)

 **注意：**

- 本ヘルプドキュメント以外にも、付属のリリースノート（ReleaseNote_C1Studio.htm）に各種情報を記載していますので、そちらもあわせてご覧ください。
- リリースノートは、製品をインストールしたフォルダ内にあり、スタートメニューの [ComponentOne] – [ComponentOne Studio for XXX] – [リリースノート] から起動できます。※XXX はエディション名です。

使用上のご注意

本製品を使うに当たっての基本的な注意事項です。

ヘルプドキュメントについて

製品全般、およびテクノロジー共通の内容（プロジェクトの作成方法や配布方法など）は、こちら「[ComponentOne Studio 共通ヘルプ](#)」をご覧ください。

各コンポーネントごとの詳細な使用方法、リファレンスにつきましては、各コンポーネントのヘルプをご覧ください。

- [ComponentOne Studio for WinForms ヘルプ](#)
- [ComponentOne Studio for ASP.NET Web Forms ヘルプ](#)
- [ComponentOne Studio for ASP.NET MVC ヘルプ](#)
- [ComponentOne Studio for Silverlight ヘルプ](#)
- [ComponentOne Studio for WPF ヘルプ](#)
- [ComponentOne Studio for UWP ヘルプ](#)

各コンポーネントの制限事項と注意点について

各コンポーネントごとの「制限事項と注意点」は、弊社 Web サイトの「[ナレッジベース](#)」にございますので、そちらをご覧ください。「対象製品」で「ComponentOne Studio for XXX」を選択し、「文書種別」で「Limit」をチェックして「検索開始」ボタンをクリックすると、制限事項の一覧を表示できます。

更新履歴について

各製品の更新履歴は、弊社 Web サイトの「更新履歴」をご覧ください。


- [ComponentOne Studio for WinForms の更新履歴](#)
- [ComponentOne Studio for ASP.NET Web Forms の更新履歴](#)
- [ComponentOne Studio for ASP.NET MVC の更新履歴](#)
- [ComponentOne Studio for Silverlight の更新履歴](#)
- [ComponentOne Studio for WPF の更新履歴](#)
- [ComponentOne Studio for UWP の更新履歴](#)

複数種類のアセンブリと対象フレームワーク

for WinForms エディションには、複数の .NET Framework でビルドされたアセンブリが収録されています。これらの対象フレームワークについては、付属のリリースノート（ReleaseNote_C1Studio.htm）の「インストール」-「6. 2 種類のコントロール」をご覧ください。リリースノートは、製品をインストールしたフォルダ内にあり、スタートメニューの [ComponentOne]-[ComponentOne Studio for WinForms]-[リリースノート] から起動できます。


必要システム

本製品を使用するために必要なシステムについては、付属のリリースノート（ReleaseNote_C1Studio.htm）に記載されていますので、そちらをご覧ください。リリースノートは、製品をインストールしたフォルダ内にあり、スタートメニューの [ComponentOne]-[ComponentOne Studio for XXX]-[リリースノート] から起動できます。

メモ： XXX はエディション名です。


インストールとアンインストール

本製品のインストール/アンインストール方法については、付属のリリースノート（ReleaseNote_C1Studio.htm）に記載されていますので、そちらをご覧ください。リリースノートは、製品をインストールしたフォルダ内にあり、スタートメニューの [ComponentOne]-[ComponentOne Studio for XXX]-[リリースノート] から起動できます。

メモ： XXX はエディション名です。

ライセンス認証

本製品をインストールしてご使用いただくためには、ライセンス認証を行う必要があります。また、本製品をアンインストールする場合は、ライセンス認証解除を行う必要があります。本製品のライセンス認証／認証解除方法については、付属のリリースノート（ReleaseNote_C1Studio.htm）に記載されていますので、そちらをご覧ください。リリースノートは、製品をインストールしたフォルダ内にあり、スタートメニューの [ComponentOne] - [ComponentOne Studio for XXX] - [リリースノート] から起動できます。

メモ： XXX はエディション名です。

アプリケーション開発の手順


本製品を使用したアプリケーション開発の大まかな流れを説明します。各手順の詳細については、リンク先をご覧ください。

1. インストール後に必要な設定

本製品のインストールと[ライセンス認証](#)の手続きを行った後、各コンポーネントを使用するために必要な設定があります。はじめにこちらの設定を行ってください。

各コンポーネントのヘルプ [○○○ for △△△ ユーザーガイド] - [はじめに] - [コンポーネントをプロジェクトに組み込む方法]

※○○○はコンポーネント名、△△△はテクノロジー名。例：FlexGrid for WinForms ユーザーガイド

注意：本製品に更新版を適用した後は、ツールボックスへの登録をしない必要があります。

2. プロジェクトの作成方法

各テクノロジーごとにプロジェクトの作成方法を紹介しています。

- [プロジェクトの作成](#)

3. コンポーネントの使用方法

コンポーネントごとの詳細な使用方法、リファレンスにつきましては、弊社 Web サイトのヘルプをご覧ください。ヘルプへのリンクは「[使用上のご注意](#)」の「ヘルプドキュメントについて」の項を参照してください。

メモ： 製品の付属サンプルの利用方法については、「[サンプルについて](#)」をご覧ください。

4. プロジェクトの移行

既存のプロジェクトを本製品へ移行する方法について説明します。

本製品の旧バージョンから移行される方や、本製品の更新版を適用された方は、こちらをご覧ください。

- [移行ガイド](#)

⚠注意： 既存のプロジェクトに本製品の更新版を適用する場合は、プロジェクトの移行が必要です。

5. アプリケーションの配布

各作成したアプリケーションの配布方法について説明します。

- [配布ガイド](#)

6. トラブルシューティング

何らかの問題が発生した場合に、解決するための方法やヒントを紹介します。

サポートサービス（弊社 Web サイトの「[ナレッジベース](#)」 「[サポート](#)」ページ）

プロジェクトの作成

本製品を使用したアプリケーションの作成について、プロジェクトの作成方法をその形式ごとに説明します。

プロジェクトを作成する前に、以下の注意事項をご一読ください。

コンポーネントをプロジェクトに組み込む方法

コンポーネントの組み込み

Visual Studio は、ツールボックスにコンポーネントを追加しただけでは、プロジェクトにコンポーネントを追加したことにはなりません。プロジェクトの参照設定へ追加された時点でコンポーネントが組み込まれます。

以下のいずれかの操作を行うとプロジェクトへコンポーネントが組み込まれます。

- フォームにコンポーネントを配置する
- ソリューションエクスプローラ上で参照の追加を行う

プロジェクトに組み込まれているコンポーネントの一覧は、ソリューションエクスプローラで確認できます。また、各コンポーネントが使用している DLL もソリューションエクスプローラに登録される場合があります。詳細については、Visual Studio の製品ヘルプを参照してください。

フォームにコンポーネントを配置する方法

例として、**FlexGrid for WinForms** の **C1FlexGrid** コントロールをツールボックスに追加し、フォームに配置する方法を示します。これにより、コンポーネントがプロジェクトに組み込まれます。

配置手順

1. [ツール] メニューから [ツールボックスアイテムの選択] を選択します。
2. [ツールボックスアイテムの選択] ダイアログの「.NET Framework コンポーネント」タブを選択します。

- 使用するコンポーネントのチェックボックスを ON にして 〈OK〉 ボタンをクリックしてください。ツールボックスに指定したコントロールのアイコンが表示されます。

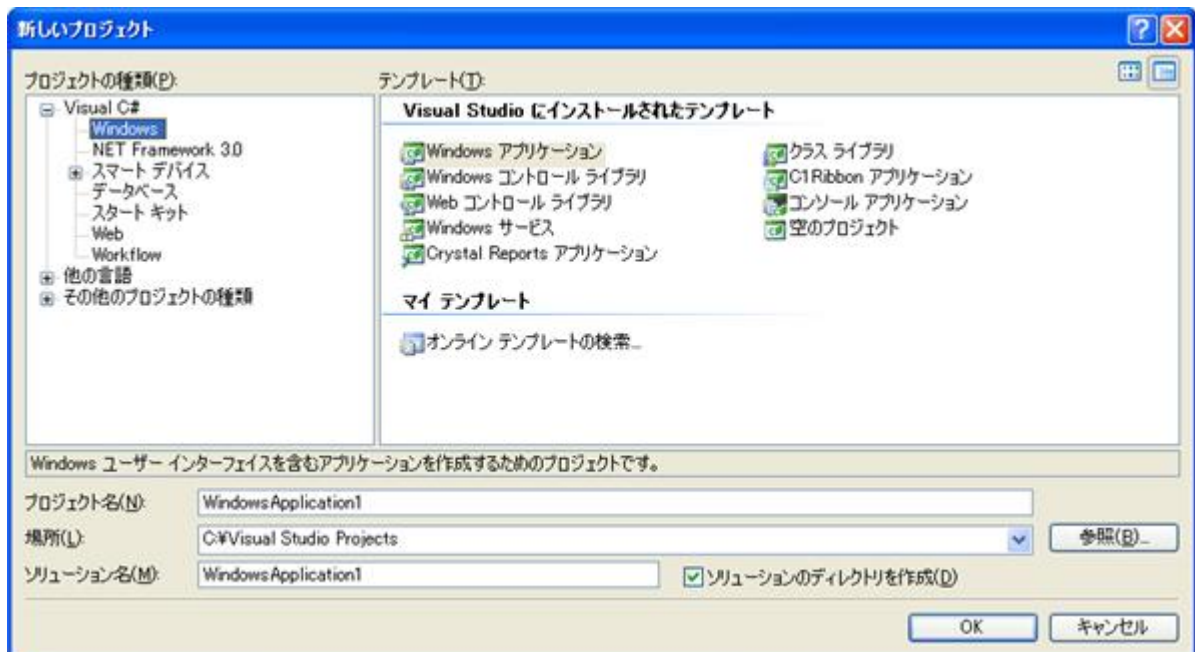


- ツールボックスから指定したコントロールのアイコンを選択してフォームに配置します。ソリューションエクスプローラの参照設定に指定したコントロールの名前空間が追加されます。

Windows フォームアプリケーション

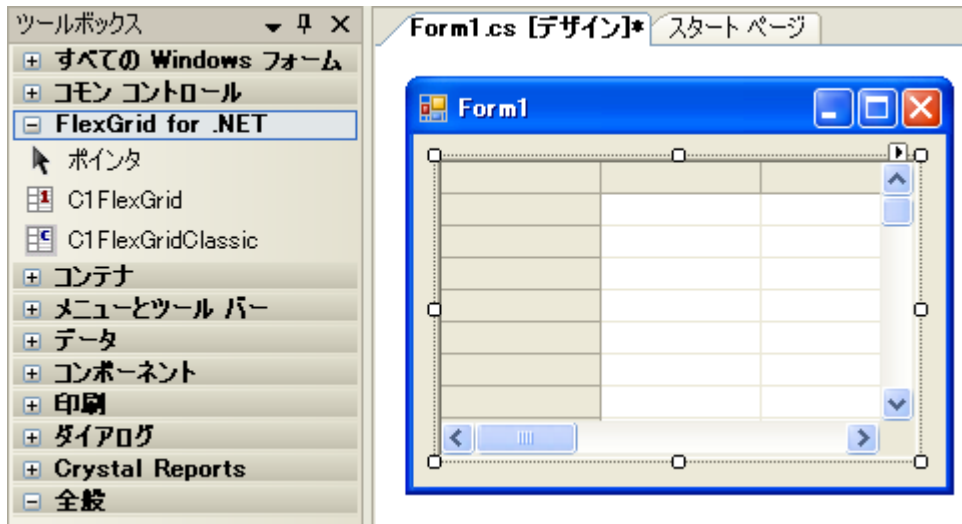
新しい Windows フォームアプリケーションプロジェクトを作成するには、以下の手順を実行します。

- Microsoft Visual Studio の [ファイル] メニューから、[新しいプロジェクト] を選択します。[新しいプロジェクト] ダイアログボックスが開きます。
- [プロジェクトの種類] で、[Visual Basic] または [Visual C#] を選択し、右ペインの [テンプレート] のリストから [Windows フォームアプリケーション] を選択します。



- [プロジェクト名] フィールドにアプリケーションの名前を入力し、〈OK〉 をクリックします。

4. 新しい Windows フォームアプリケーションプロジェクトが作成され、デザイナビューに新規の Form1 が表示されます。
5. ツールボックスから、使用する Windows フォーム用コンポーネント（下図の例では **C1FlexGrid** コントロール）をダブルクリックして、Form1 に追加します。ツールボックスにコンポーネントを追加する方法の詳細については、「[プロジェクトの作成](#)」内の「コンポーネントをプロジェクトに組み込む方法」を参照してください。

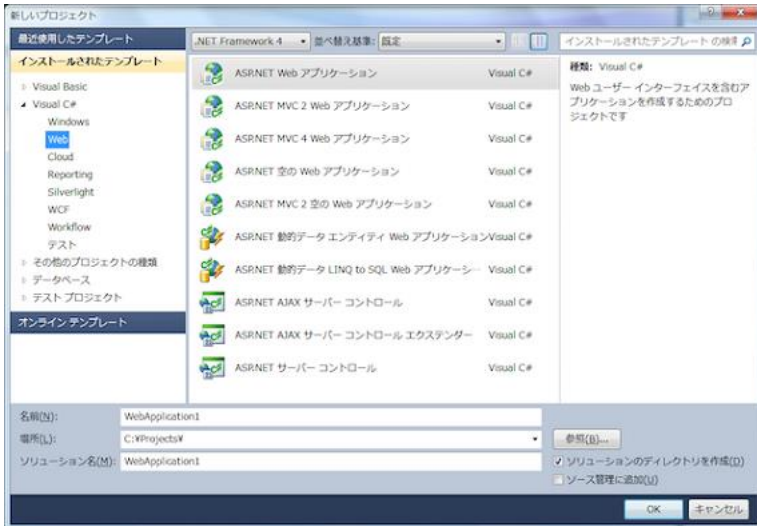


ASP.NET Web アプリケーション

新しい ASP.NET Web アプリケーションプロジェクトを作成するには、次の手順を実行します。

Visual Studio 2010/2012 の場合

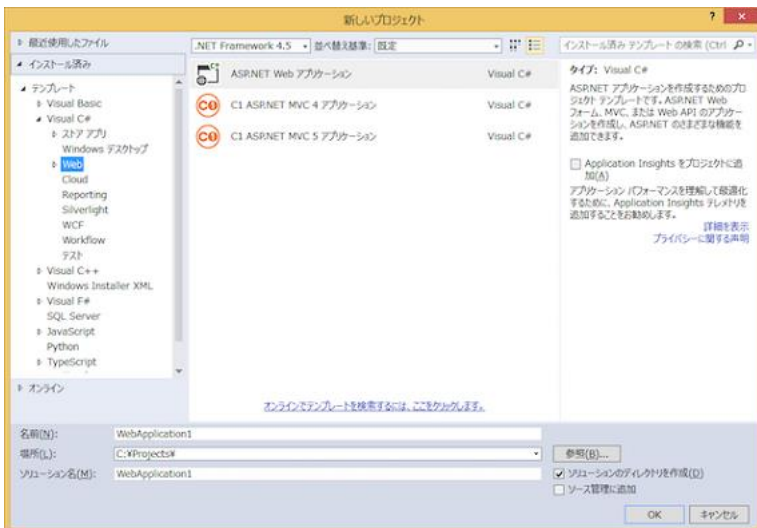
1. Microsoft Visual Studio の [ファイル] メニューから、[新規作成] - [プロジェクト] を選択します。[新しいプロジェクト] ダイアログボックスが開きます。
2. [インストールされたテンプレート] の一覧から、[Visual Basic] - [Web] または [Visual C#] - [Web] を選択します。
3. テンプレート一覧から、[ASP.NET Web アプリケーション] または [ASP.NET MVC Web アプリケーション] を選択します。



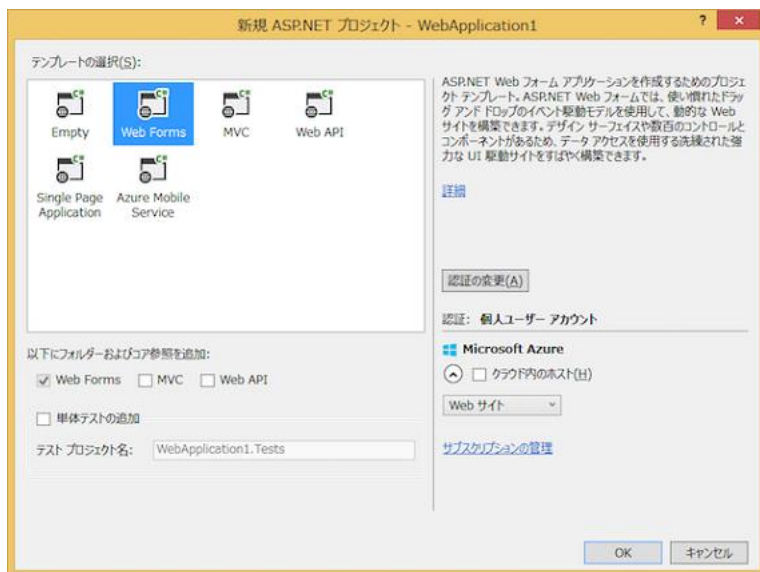
4. [プロジェクト名] フィールドにアプリケーションの名前を入力し、〈OK〉をクリックします。
5. 新しいプロジェクトが作成されます。

Visual Studio 2013 の場合

1. Microsoft Visual Studio の [ファイル] メニューから、[新規作成] - [プロジェクト] を選択します。[新しいプロジェクト] ダイアログボックスが開きます。
2. [インストール済み] の一覧から、[Visual Basic] - [Web] または [Visual C#] - [Web] を選択します。
3. テンプレート一覧から、[ASP.NET Web アプリケーション] を選択して、[プロジェクト名] フィールドにアプリケーションの名前を入力し、〈OK〉をクリックします。[新規 ASP.NET プロジェクト] ダイアログボックスが開きます。



4. [Web Forms] または [MVC] を選択し、〈OK〉をクリックします。



5. 新しいプロジェクトが作成されます。

XAML と XAML 名前空間

XAML は、Windows Presentation Foundation および Silverlight でユーザーインターフェイスマークアップ言語として使用される XML ベースの宣言型言語です。XAML を使用することで、グラフィカルにカスタマイズされた機能豊富なユーザーインターフェイスを作成したり、データ連結を実行することができます。XAML の詳細については、Microsoft 社の Web サイト等を参照してください。

XAML 名前空間

名前空間は、アセンブリ内で定義されるオブジェクトを組織化します。アセンブリは複数の名前空間を持つことができ、それらの名前空間も他の名前空間を持つことができます。名前空間はあいまいさを取り除き、クラスライブラリなどの大きなオブジェクトグループを使用する場合の参照を簡略化します。

Microsoft Blend プロジェクトを作成すると、1つの XAML ファイルが自動的に作成され、いくつかの初期名前空間が指定されます。

名前空間	説明
<code>xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"</code>	これは、Windows Presentation Foundation のデフォルトの名前空間です。
<code>xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"</code>	これは、 x: 接頭辞にマップされる XAML 名前空間です。 x: プレフィックスは、WPF アプリケーションに必要なよく使用される多くの機能を定義する名前空間をすばやく

簡単に参照する方法を提供します。

WPF コンポーネントの名前空間

Chart for WPF の **C1Chart** コントロールを例にすると、**C1Chart** コントロールを Microsoft Expression Blend のウィンドウに追加すると、Blend は、**C1.WPF.C1Chart** アセンブリにマップされる共通言語ランタイム (CLR) 名前空間 (clr-namespace) を自動的に作成します。この名前空間は次のようになります。

```
xmlns:C1_WPF_C1Chart="clr-namespace:C1.WPF.C1Chart;assembly=C1.WPF.C1Chart"
```

clr-namespace 値は **C1.WPF.C1Chart**、assembly 値は **C1.WPF.C1Chart** です。

C1Chart コントロールを Visual Studio 2008 のウィンドウに追加すると、Visual Studio は、**C1.WPF.C1Chart** アセンブリにマップされる共通言語ランタイム (CLR) 名前空間 (clr-namespace) を自動的に作成します。この名前空間は次のようになります。

```
xmlns:my="clr-namespace:C1.WPF.C1Chart;assembly=C1.WPF.C1Chart"
```

また、この名前空間に独自の名前を付けることもできます。たとえば、次のように記述します。

```
xmlns:MyC1Chart="clr-namespace:C1.WPF.C1Chart;assembly=C1.WPF.C1Chart"
```

これで、プロパティ、メソッド、およびイベントを割り当てる際に、独自の名前空間を使用できます。

Silverlight コンポーネントの名前空間

本製品の Silverlight コンポーネントでは、以下の名前空間が定義されています。

名前空間	説明
<code>xmlns:c1="http://schemas.componentone.com/winfx/2006/xaml"</code>	これは、本製品のすべての Silverlight コンポーネントで使用される既定の名前空間です。

Silverlight コントロールをページ上に配置すると、本製品のすべての Silverlight アセンブリの名前空間が自動的に作成されます。ページ上に Silverlight コントロールを配置する際は、c1: プレフィックスを使用して以下のように記述できます。

```
<c1:C1FlexGrid />
```

名前空間には、独自の名前を付けることもできます。例えば、次のように記述することができます。

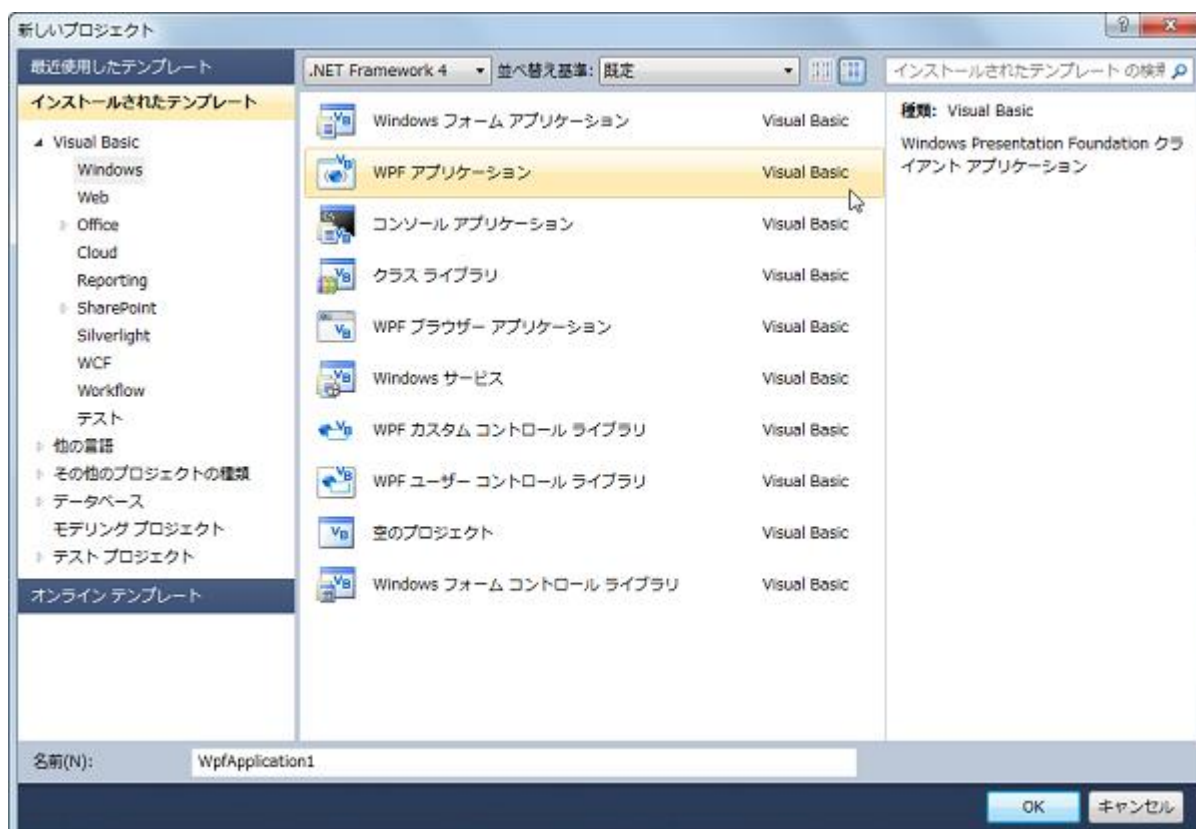
```
<UserControl ...  
xmlns:MyC1="http://schemas.componentone.com/winfx/2006/xaml"  
>  
</>  
  
<MyC1:C1FlexGrid />
```

WPF プロジェクトの作成

WPF アプリケーション

Microsoft Visual Studio で新しい WPF アプリケーションプロジェクトを作成するには、次の手順を実行します。

1. Visual Studio で、[ファイル] → [新規作成] → [プロジェクト] を選択し、[新しいプロジェクト] ダイアログボックスを開きます。
2. 上部のドロップダウンボックスから、.NET Framework のタイプを選択します。
3. [インストールされたテンプレート] ペインで、Visual Basic または Visual C# ノードのいずれかを展開し、[Windows] を選択します。
4. 右ペインのリストから [WPF アプリケーション] を選択します。

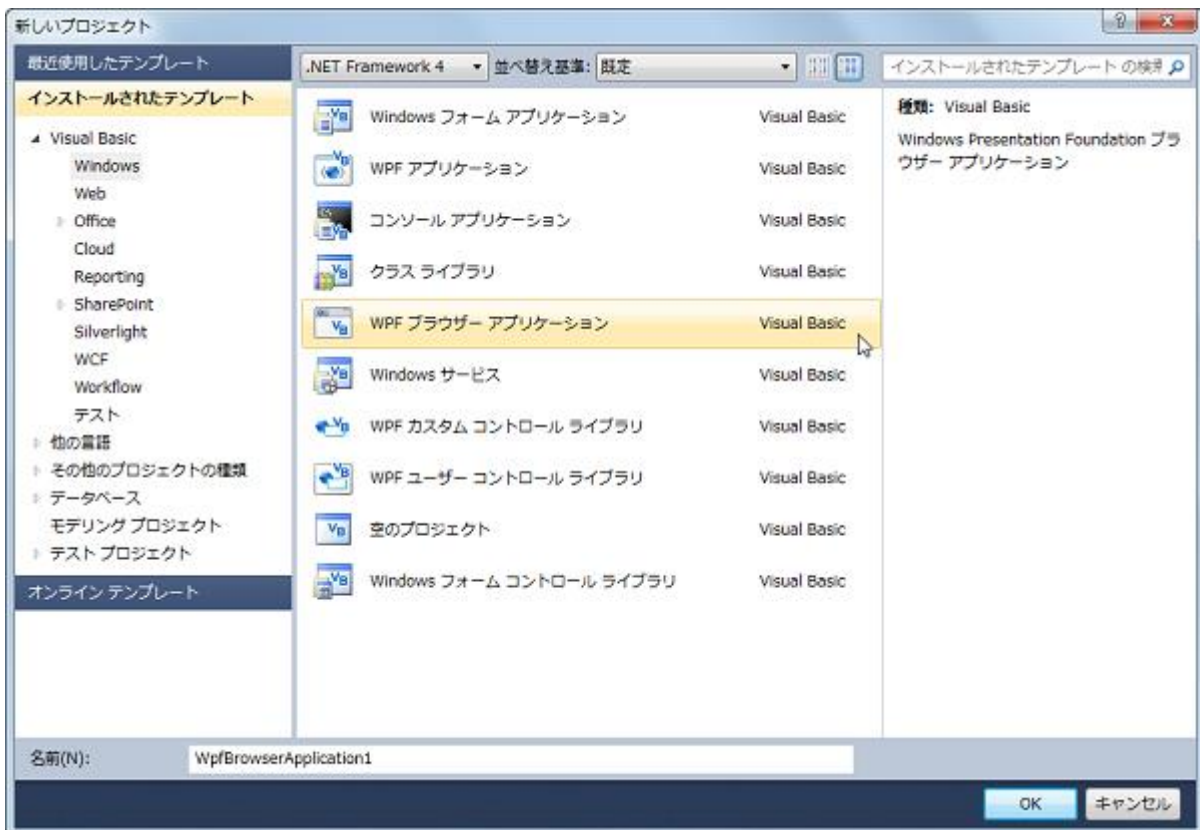


5. [名前] フィールドにアプリケーションの名前を入力し、[OK] をクリックします。新しい WPF アプリケーションプロジェクトが .xaml ファイルと共に作成されます。このファイルは、アプリケーションのユーザーインターフェイスおよびコマンドを定義するために使用されます。

WPF ブラウザーアプリケーション

Visual Studio で新しい XAML ブラウザーアプリケーション (XBAP) を作成するには、次の手順を実行します。

1. Visual Studio で、[ファイル] → [新規作成] → [プロジェクト] を選択し、[新しいプロジェクト] ダイアログボックスを開きます。
2. 上部のドロップダウンボックスから、.NET Framework のタイプを選択します。
3. [インストールされたテンプレート] ペインで、Visual Basic または Visual C# ノードのいずれかを展開し、[Windows] を選択します。
4. 右ペインのリストから [WPF ブラウザーアプリケーション] を選択します。



5. プロジェクトに名前を付け、プロジェクトの場所を指定して、[OK] をクリックします。

新しい WPF ブラウザーアプリケーションプロジェクトが XAML ファイルと共に作成されます。このファイルは、アプリケーションのユーザーインターフェイスおよびコマンドを定義するために使用されます。

Expression Blend でのプロジェクトの作成

Microsoft Expression Blend で新しい WPF アプリケーションプロジェクトを作成するには、次の手順を実行します。

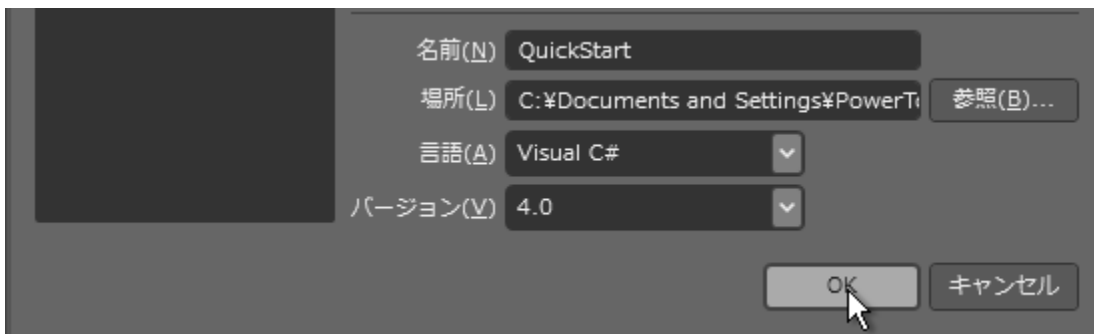
1. Expression Blend で、[ファイル] → [新しいプロジェクト] を選択し、[新しいプロジェクト] ダイアログボックスを開きます。

午後 3:11 2014/01/24

2. [プロジェクトの種類] ペインで、WPF ノードをクリックします。
3. Web サイトに関連付けたプロジェクトを作成するには、右ペインで [WPF アプリケーション] を選択します。



4. プロジェクトに名前を付け、このプロジェクトの場所を指定します。言語 (Visual C# または Visual Basic) を選択し、[OK] をクリックします。



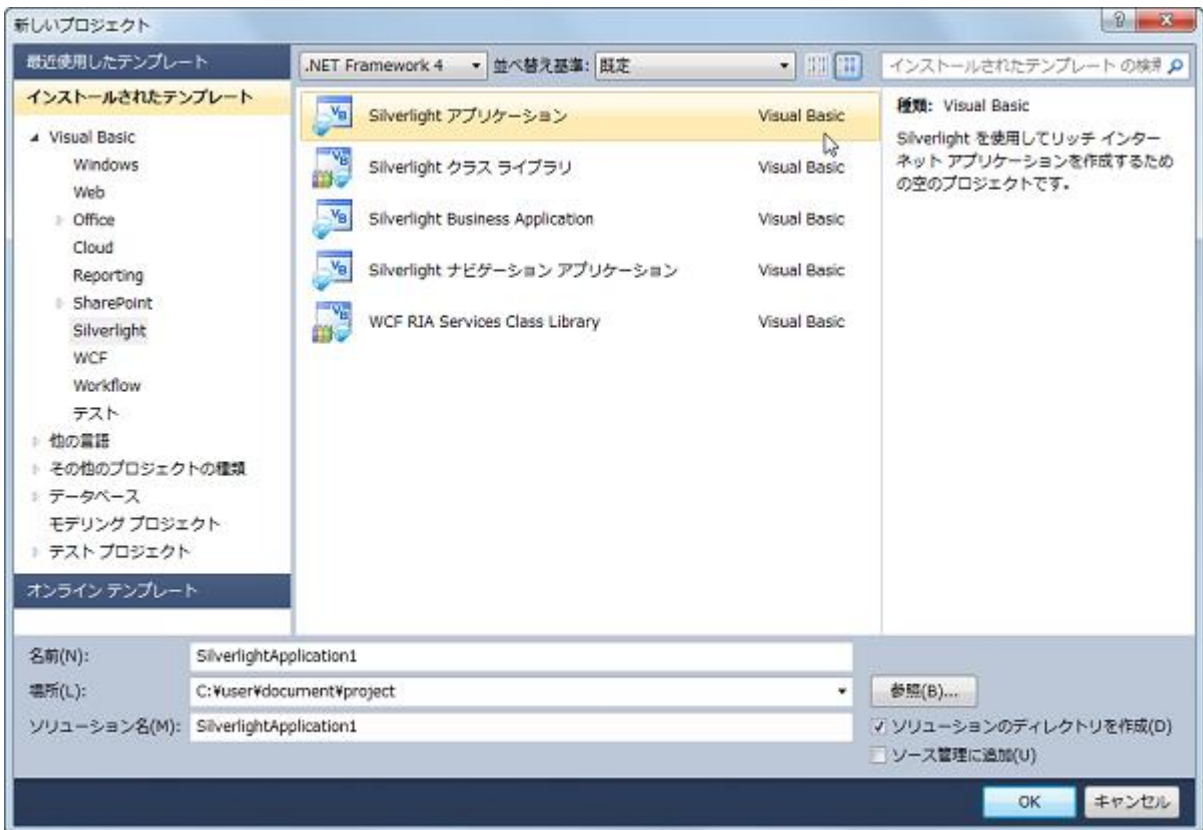
XAML ウィンドウを持つ新しい Blend プロジェクトが作成されます。

Silverlight プロジェクトの作成

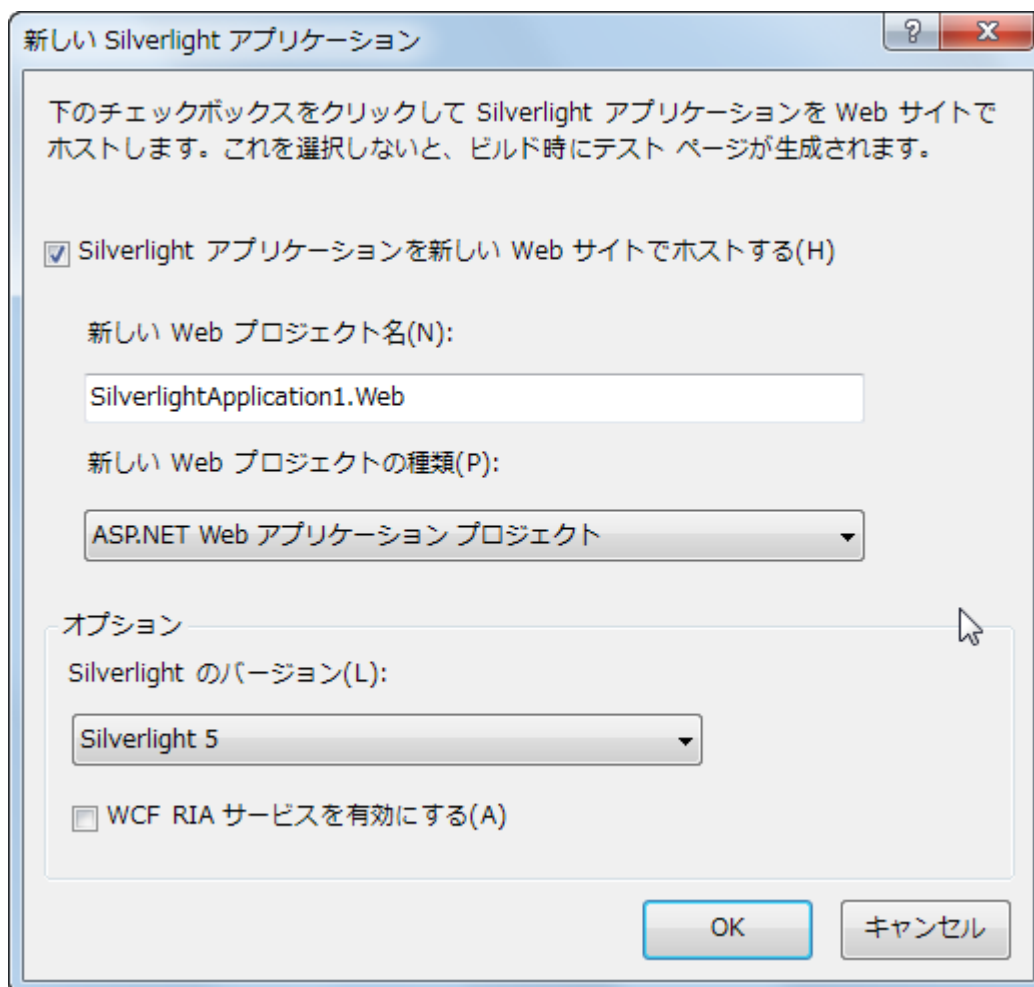
Silverlight アプリケーション

Microsoft Visual Studio で新しい Silverlight プロジェクトを作成するには、次の手順に従います。

1. Visual Studio で、[ファイル] → [新規作成] → [プロジェクト] を選択し、[新しいプロジェクト] ダイアログボックスを開きます。
2. [インストールされたテンプレート] ペインで、Visual Basic または Visual C# ノードのいずれかを展開し、[Silverlight] を選択します。
3. 右ペインで、[Silverlight アプリケーション] を選択します。



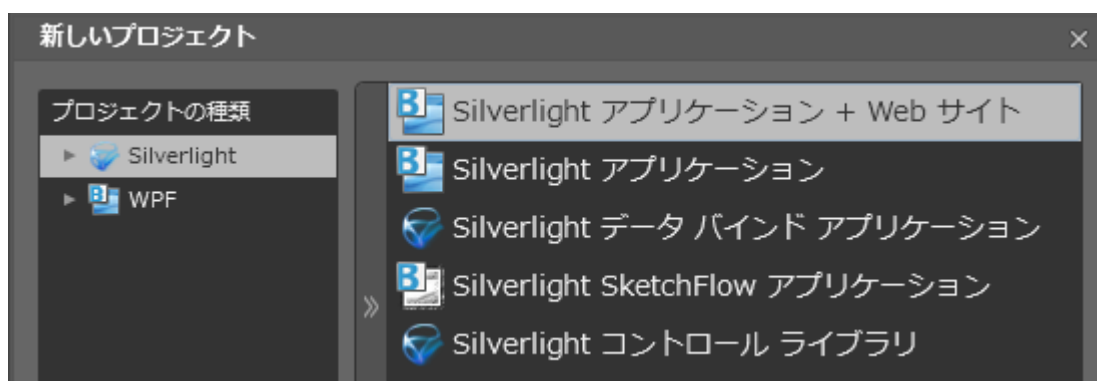
4. プロジェクトに名前を付け、プロジェクトの場所を指定して、[OK] をクリックします。
次に、Visual Studio から、新しいプロジェクトで使用するホスティングのタイプを入力するように指示されます。
5. [新しい Silverlight アプリケーション] ダイアログボックスで、[OK] を選択してデフォルトの名前とオプションを受け入れ、プロジェクトを作成します。



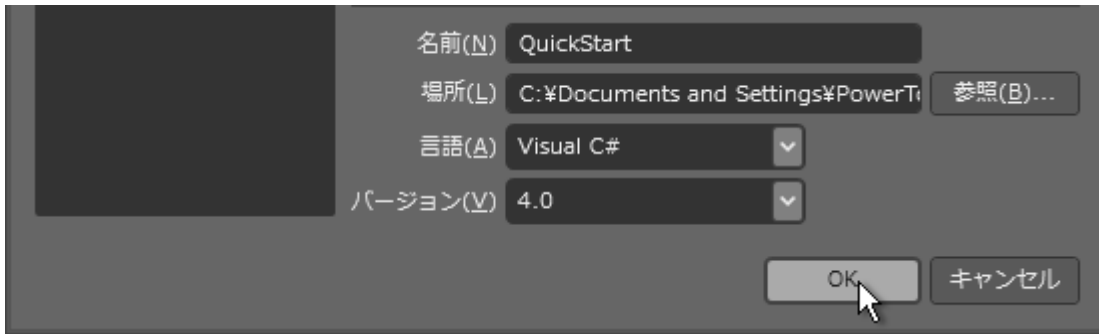
Expression Blend でのプロジェクトの作成

Microsoft Expression Blend で新しい Silverlight プロジェクトを作成するには、次の手順に従います。

1. Expression Blend で、[ファイル] → [新しいプロジェクト] を選択し、[新しいプロジェクト] ダイアログボックスを開きます。
2. [プロジェクトの種類] ペインで、Silverlight ノードをクリックします。
3. Web サイトに関連付けたプロジェクトを作成するには、右ペインの [テンプレート] ペインで、[Silverlight アプリケーション + Web サイト] を選択します。



4. プロジェクトに名前を付け、このプロジェクトの場所を指定します。言語（Visual C# または Visual Basic）を選択し、[OK] をクリックします。



新しいプロジェクトが作成されます。

プロジェクト

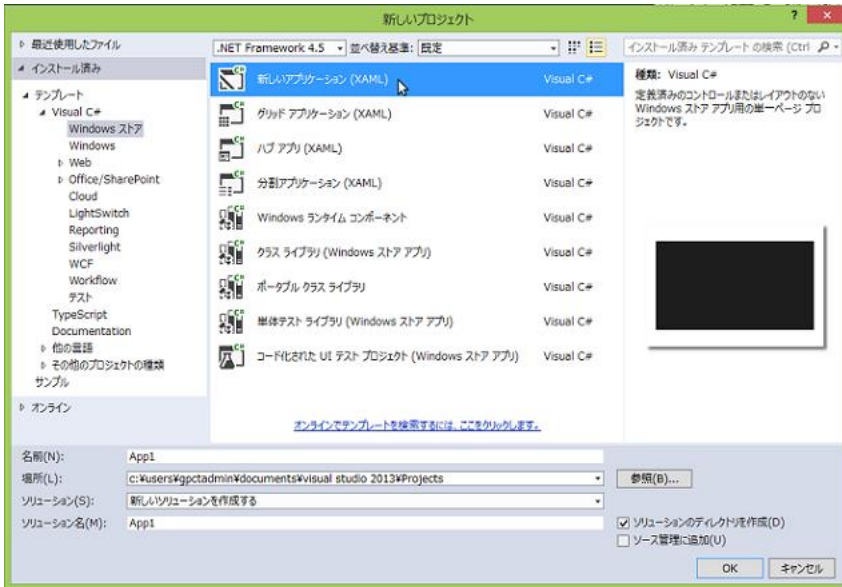
作成したソリューションには、YourProject と YourProject.Web の2つのプロジェクトが含まれます。

- YourProject : この Silverlight アプリケーションそのものです。クライアントにダウンロードされ、Silverlight プラグイン内で実行される XAP ファイルを生成します。
- YourProject.Web : ホストアプリケーションです。サーバー側で実行され、Silverlight アプリケーションをサポートします。

Windows ストアプロジェクトの作成

Microsoft Visual Studio で新しい WinRT スタイルアプリケーションを作成するには、次の手順を実行します。

1. Microsoft Visual Studio の [ファイル] メニューから、[新しいプロジェクト] を選択します。[新しいプロジェクト] ダイアログボックスが開きます。
2. 左ペインの一覧から、[インストール済み] の [テンプレート] ノードの下にある言語を展開します。そのほかの言語は [他の言語] ノードの下に表示されていますのでご確認ください。例えば、ComponentOne プロジェクトの場合 C# または Visual Basic を使用できます。



3. 言語ノードの下に表示するリストから [Windows ストア] を選択します。真ん中ペインに WinRT スタイル のテンプレートのリストが表示されます。
4. 以下に説明するテンプレートリストから一つ選択してください。

テンプレート	サポートされる言語	説明
新しいアプリケーション(XAML)	C#/VB/C++	定義済みのレイアウトのない Windows ストアアプリを提供します。
グリッド アプリケーション (XAML)	C#/VB/C++	ユーザーはショッピングやニュースなどのカテゴリ間を移動できる、カスタマイズ可能な Windows ストアアプリを提供します。
固定レイアウト アプリケーション	JavaScript	ゲームアプリに最適な固定ビューポートを提供する JavaScript テンプレートです。
ナビゲーション アプリケーション	JavaScript	定義済みのナビゲーションコントロールのある Windows ストアアプリを提供します。
分割アプリケーション(XAML)	C#/VB/C++	ユーザーはニュースリーダーなどのような項目リストを表示できる、カスタマイズ可能な Windows ストアアプリを提供します。

5. プロジェクトに名前を付け、プロジェクトの場所を指定して、 [OK] をクリックします。新しいプロジェクトが開きます。

これで、Windows ストアアプリ用の新しいアプリケーションが作成できました。また、MainPage.xaml ファイルは自動的に開かないことがあり、 [ソリューションエクスプローラ] から選択する必要がありますのでご注意ください。

ライセンスの組み込み

コントロールによって、以下の 2 通りの方法のうちのいずれかを適用します。

(1) Licenses.licx ファイルを使用する

コントロールを製品版として使用した場合、設計時にコントロールをフォームに貼り付けると Licenses.licx ファイルが自動的に作成され、プロジェクトに組み込まれます。しかし、実行時にコントロールを生成する場合には、Licenses.licx ファイルは作成されません。この場合は、該当するすべてのコントロールの情報を含んだ Licenses.licx ファイルを作成し、プロジェクトに組み込む必要があります。プロジェクトに Licenses.licx ファイルが組み込まれていない場合は、実行時にコントロールが生成されたときに、ライセンスのエラーを示すダイアログが表示されます。

Licenses.licx ファイルを作成するには、Visual Basic のプロジェクトの場合は「My Project」フォルダ、C# のプロジェクトの場合は「Properties」フォルダの下に、ファイル名を「Licenses.licx」とした空のテキストファイルを作成します。

Licenses.licx ファイルが使用されるのは、次のコントロールです。

- Windows フォーム用コントロール
- ASP.NET Web フォーム用コントロール
- ASP.NET MVC 用コントロール
- WPF 用コントロール

なお、コントロールをトライアル版として使用する場合、Licenses.licx ファイルは不要です。

(2) ランタイムライセンス情報を組み込む

この方法に該当するコントロールでは、トライアル版および製品版のいずれの場合でも、ランタイムライセンスを生成して組み込む必要があります。

ランタイムライセンスが使用されるのは、次のコントロールです。

- ASP.NET MVC Core コントロール
- UWP 用コントロール

ランタイムライセンスは **C1 Web サイト** で生成することができます。生成および組み込み手順の詳細については、[ASP.NET MVC Core](#) を参照してください。

※ライセンスの仕組み > ASP.NET Core タブをクリックしてください。

Windows フォーム用コントロール

Licenses.licx ファイルに記載するライセンス情報は、次の通りです。

.NET Framework 2.0 用

使用するコントロール名	Licenses.licx に記載が必要なライセンス情報
BarCode for WinForms (legacy)	C1.Win.C1BarCode.C1BarCode, C1.Win.C1BarCode.2 C1.Win.C1BarCode.C1QRCode, C1.Win.C1BarCode.2
DockingTab for WinForms	C1.Win.C1Command.C1DockingManager, C1.Win.C1Command.2 C1.Win.C1Command.C1DockingTab, C1.Win.C1Command.2
DynamicHelp for WinForms	C1.Win.C1DynamicHelp.C1DynamicHelp, C1.Win.C1DynamicHelp.2
Editor for WinForms	C1.Win.C1Editor.C1Editor, C1.Win.C1Editor.2 C1.Win.C1Editor.ToolStrips.C1EditorToolStripMain, C1.Win.C1Editor.2 C1.Win.C1Editor.ToolStrips.C1EditorToolStripObjects, C1.Win.C1Editor.2 C1.Win.C1Editor.ToolStrips.C1EditorToolStripReview, C1.Win.C1Editor.2 C1.Win.C1Editor.ToolStrips.C1EditorToolStripStyle, C1.Win.C1Editor.2 C1.Win.C1Editor.ToolStrips.C1EditorToolStripTable, C1.Win.C1Editor.2
Excel for .NET	C1.C1Excel.C1XLBook, C1.C1Excel.2
FlexGrid for WinForms	C1.Win.C1FlexGrid.C1FlexGrid, C1.Win.C1FlexGrid.2 C1.Win.C1FlexGrid.Classic.C1FlexGridClassic, C1.Win.C1FlexGrid.Classic.2
GanttView for WinForms	C1.Win.C1GanttView.C1GanttView, C1.Win.C1GanttView.2
Gauges for WinForms	C1.Win.C1Gauge.C1Gauge, C1.Win.C1Gauge.2
Input for WinForms	C1.Win.C1Input.C1Button, C1.Win.C1Input.2 C1.Win.C1Input.C1CheckBox, C1.Win.C1Input.2 C1.Win.C1Input.C1ColorPicker, C1.Win.C1Input.2 C1.Win.C1Input.C1ComboBox, C1.Win.C1Input.2 C1.Win.C1Input.C1DateEdit, C1.Win.C1Input.2 C1.Win.C1Input.C1DbNavigator, C1.Win.C1Input.2 C1.Win.C1Input.C1DropDownControl, C1.Win.C1Input.2 C1.Win.C1Input.C1FontPicker, C1.Win.C1Input.2 C1.Win.C1Input.C1Label, C1.Win.C1Input.2

	<p>C1.Win.C1Input.C1NumericEdit, C1.Win.C1Input.2</p> <p>C1.Win.C1Input.C1PictureBox, C1.Win.C1Input.2</p> <p>C1.Win.C1Input.C1RangeSlider, C1.Win.C1Input.2</p> <p>C1.Win.C1Input.C1SplitButton, C1.Win.C1Input.2</p> <p>C1.Win.C1Input.C1TextBox, C1.Win.C1Input.2</p>
InputPanel for WinForms	C1.Win.C1InputPanel.C1InputPanel, C1.Win.C1InputPanel.2
List for WinForms	<p>C1.Win.C1List.C1Combo, C1.Win.C1List.2</p> <p>C1.Win.C1List.C1List, C1.Win.C1List.2</p>
Menus and Toolbars for WinForms	<p>C1.Win.C1Command.C1CommandDock, C1.Win.C1Command.2</p> <p>C1.Win.C1Command.C1CommandHolder, C1.Win.C1Command.2</p> <p>C1.Win.C1Command.C1ContextMenu, C1.Win.C1Command.2</p> <p>C1.Win.C1Command.C1MainMenu, C1.Win.C1Command.2</p> <p>C1.Win.C1Command.C1ToolBar, C1.Win.C1Command.2</p>
NavBar for WinForms	C1.Win.C1Command.C1NavBar, C1.Win.C1Command.2
OLAP for WinForms	<p>C1.Win.Olap.C1OlapChart, C1.Win.Olap.2</p> <p>C1.Win.Olap.C1OlapGrid, C1.Win.Olap.2</p> <p>C1.Win.Olap.C1OlapPage, C1.Win.Olap.2</p> <p>C1.Win.Olap.C1OlapPanel, C1.Win.Olap.2</p>
OutBar for WinForms	C1.Win.C1Command.C1OutBar, C1.Win.C1Command.2
PDF for .NET	C1.C1Pdf.C1PdfDocument, C1.C1Pdf.2
RadialMenu for WinForms	C1.Win.C1Command.C1RadialMenu, C1.Win.C1Command.2
Reports for WinForms	<p>C1.C1Preview.C1MultiDocument, C1.C1Report.2</p> <p>C1.C1Preview.C1PrintDocument, C1.C1Report.2</p> <p>C1.C1Rdl.Rdl2008.C1RdlReport, C1.C1Report.2</p> <p>C1.C1Report.C1Report, C1.C1Report.2</p> <p>C1.Win.C1Preview.C1PreviewPane, C1.Win.C1Report.2</p> <p>C1.Win.C1Preview.C1PrintPreviewControl, C1.Win.C1Report.2</p> <p>C1.Win.C1ReportDesigner.C1ReportDesigner,</p> <p>C1.Win.C1ReportDesigner.2</p>
Ribbon for WinForms	<p>C1.Win.C1Ribbon.C1Ribbon, C1.Win.C1Ribbon.2</p> <p>C1.Win.C1Ribbon.C1StatusBar, C1.Win.C1Ribbon.2</p>
Scheduler for WinForms	<p>C1.Win.C1Schedule.C1Schedule, C1.Win.C1Schedule.2</p> <p>C1.Win.C1Schedule.C1Calendar, C1.Win.C1Schedule.2</p>

Sizer for WinForms	C1.Win.C1Sizer.C1Sizer, C1.Win.C1Sizer.2 C1.Win.C1Sizer.C1SizerLight, C1.Win.C1Sizer.2
SpellChecker for WinForms	C1.Win.C1SpellChecker.C1SpellChecker, C1.Win.C1SpellChecker.2
SplitContainer for WinForms	C1.Win.C1SplitContainer.C1SplitContainer, C1.Win.C1SplitContainer.2
SuperToolTip for WinForms	C1.Win.C1SuperToolTip.C1SuperErrorProvider, C1.Win.C1SuperToolTip.2 C1.Win.C1SuperToolTip.C1SuperLabel, C1.Win.C1SuperToolTip.2 C1.Win.C1SuperToolTip.C1SuperToolTip, C1.Win.C1SuperToolTip.2
Themes for WinForms	C1.Win.C1Themes.C1ThemeController, C1.Win.C1Themes.2
TileControl for WinForms	C1.Win.C1Tile.C1TileControl, C1.Win.C1TileControl.2
TopicBar for WinForms	C1.Win.C1Command.C1TopicBar, C1.Win.C1Command.2
TouchToolkit for WinForms	C1.Win.TouchToolkit.C1ApplicationZoom, C1.Win.C1TouchToolkit.2 C1.Win.TouchToolkit.C1Magnify, C1.Win.C1TouchToolkit.2 C1.Win.TouchToolkit.C1MultiScaleImage, C1.Win.C1TouchToolkit.2 C1.Win.TouchToolkit.C1TouchEventProvider, C1.Win.C1TouchToolkit.2 C1.Win.TouchToolkit.C1Zoom, C1.Win.C1TouchToolkit.2 C1.Win.TouchToolkit.C1ZoomCommandProvider, C1.Win.C1TouchToolkit.2 C1.Win.TouchToolkit.C1ZoomPanel, C1.Win.C1TouchToolkit.2
True DBGrid for WinForms	C1.Win.C1TrueDBGrid.C1TrueDBGrid, C1.Win.C1TrueDBGrid.2
TrueChart for WinForms (2D)	C1.Win.C1Chart.C1Chart, C1.Win.C1Chart.2
TrueChart for WinForms (3D)	C1.Win.C1Chart3D.C1Chart3D, C1.Win.C1Chart3D.2
Windows 7 Control Pack for WinForms	C1.Win.C1Win7Pack.C1TaskbarButton, C1.Win.C1Win7Pack.2 C1.Win.C1Win7Pack.C1TaskDialog, C1.Win.C1Win7Pack.2
Word for WinForms	C1.C1Word.C1WordDocument, C1.C1Word.2
Zip for .NET	なし

.NET Framework 4 用

使用するコントロール名	Licenses.licx に記載が必要なライセンス情報
BarCode for WinForms	C1.Win.BarCode.C1BarCode, C1.Win.BarCode.4
Bitmap for WinForms	なし

DataSource for Entity Framework	C1.Win.Data.Entities.C1DataSource, C1.Win.Data.Entity.4
DockingTab for WinForms	C1.Win.C1Command.C1DockingTab, C1.Win.C1Command.4 C1.Win.C1Command.C1DockingManager, C1.Win.C1Command.4
DynamicHelp for WinForms	C1.Win.C1DynamicHelp.C1DynamicHelp, C1.Win.C1DynamicHelp.4
Editor for WinForms	C1.Win.C1Editor.C1Editor, C1.Win.C1Editor.4 C1.Win.C1Editor.ToolStrips.C1EditorToolStripMain, C1.Win.C1Editor.4 C1.Win.C1Editor.ToolStrips.C1EditorToolStripObjects, C1.Win.C1Editor.4 C1.Win.C1Editor.ToolStrips.C1EditorToolStripReview, C1.Win.C1Editor.4 C1.Win.C1Editor.ToolStrips.C1EditorToolStripStyle, C1.Win.C1Editor.4 C1.Win.C1Editor.ToolStrips.C1EditorToolStripTable, C1.Win.C1Editor.4
Excel for .NET	C1.C1Excel.C1XLBook, C1.C1Excel.4
FinancialChart for WinForms	C1.Win.Chart.Finance.FinancialChart, C1.Win.FinancialChart.4
FlexChart for WinForms	C1.Win.Chart.FlexChart, C1.Win.FlexChart.4 C1.Win.Chart.Interaction.C1RangeSlider, C1.Win.FlexChart.4
FlexGrid for WinForms	C1.Win.C1FlexGrid.C1FlexGrid, C1.Win.C1FlexGrid.4 C1.Win.C1FlexGrid.C1FlexGridSearchPanel, C1.Win.C1FlexGrid.SearchPanel.4 C1.Win.C1FlexGrid.Classic.C1FlexGridClassic, C1.Win.C1FlexGrid.Classic.4
FlexPie for WinForms	C1.Win.Chart.FlexPie, C1.Win.FlexChart.4 C1.Win.Chart.FlexRadar, C1.Win.FlexChart.4 C1.Win.Chart.Sunburst, C1.Win.FlexChart.4
FlexPivot for WinForms	C1.Win.FlexPivot.C1FlexPivotChart, C1.Win.FlexPivot.4 C1.Win.FlexPivot.C1FlexPivotGrid, C1.Win.FlexPivot.4 C1.Win.FlexPivot.C1FlexPivotPage, C1.Win.FlexPivot.4 C1.Win.FlexPivot.C1FlexPivotPanel, C1.Win.FlexPivot.4
FlexReport for WinForms	C1.Win.FlexReport.C1FlexReport, C1.Win.FlexReport.4
FlexViewer for WinForms	C1.Win.FlexViewer.C1FlexViewer, C1.Win.FlexViewer.4 C1.Win.FlexViewer.C1FlexViewerDialog, C1.Win.FlexViewer.4 C1.Win.FlexViewer.C1FlexViewerPane, C1.Win.FlexViewer.4
GanttView for WinForms	C1.Win.C1GanttView.C1GanttView, C1.Win.C1GanttView.4

Gauges for WinForms	C1.Win.C1Gauge.C1Gauge, C1.Win.C1Gauge.4
Input for WinForms	C1.Win.C1Input.C1Button, C1.Win.C1Input.4 C1.Win.C1Input.C1CheckBox, C1.Win.C1Input.4 C1.Win.C1Input.C1ColorPicker, C1.Win.C1Input.4 C1.Win.C1Input.C1ComboBox, C1.Win.C1Input.4 C1.Win.C1Input.C1DateEdit, C1.Win.C1Input.4 C1.Win.C1Input.C1DbNavigator, C1.Win.C1Input.4 C1.Win.C1Input.C1DropDownControl, C1.Win.C1Input.4 C1.Win.C1Input.C1FontPicker, C1.Win.C1Input.4 C1.Win.C1Input.C1Label, C1.Win.C1Input.4 C1.Win.C1Input.C1NumericEdit, C1.Win.C1Input.4 C1.Win.C1Input.C1PictureBox, C1.Win.C1Input.4 C1.Win.C1Input.C1RangeSlider, C1.Win.C1Input.4 C1.Win.C1Input.C1SplitButton, C1.Win.C1Input.4 C1.Win.C1Input.C1TextBox, C1.Win.C1Input.4
InputPanel for WinForms	C1.Win.C1InputPanel.C1InputPanel, C1.Win.C1InputPanel.4
List for WinForms	C1.Win.C1List.C1Combo, C1.Win.C1List.4 C1.Win.C1List.C1List, C1.Win.C1List.4
Maps for WinForms	C1.Win.Map.C1Map, C1.Win.Map.4
Menus and Toolbars for WinForms	C1.Win.C1Command.C1CommandDock, C1.Win.C1Command.4 C1.Win.C1Command.C1CommandHolder, C1.Win.C1Command.4 C1.Win.C1Command.C1ContextMenu, C1.Win.C1Command.4 C1.Win.C1Command.C1MainMenu, C1.Win.C1Command.4 C1.Win.C1Command.C1ToolBar, C1.Win.C1Command.4
NavBar for WinForms	C1.Win.C1Command.C1NavBar, C1.Win.C1Command.4
OLAP for WinForms	C1.Win.Olap.C1OlapChart, C1.Win.Olap.4 C1.Win.Olap.C1OlapGrid, C1.Win.Olap.4 C1.Win.Olap.C1OlapPage, C1.Win.Olap.4 C1.Win.Olap.C1OlapPanel, C1.Win.Olap.4
OutBar for WinForms	C1.Win.C1Command.C1OutBar, C1.Win.C1Command.4
PDF for .NET	C1.C1Pdf.C1PdfDocument, C1.C1Pdf.4
PDF DocumentSource for WinForms	C1.Win.C1Document.C1PdfDocumentSource, C1.Win.C1Document.4

RadialMenu for WinForms	C1.Win.C1Command.C1RadialMenu, C1.Win.C1Command.4
Reports for WinForms	C1.C1Preview.C1MultiDocument, C1.C1Report.4 C1.C1Preview.C1PrintDocument, C1.C1Report.4 C1.C1Rdl.Rdl2008.C1RdlReport, C1.C1Report.4 C1.C1Report.C1Report, C1.C1Report.4 C1.Win.C1Preview.C1PreviewPane, C1.Win.C1Report.4 C1.Win.C1Preview.C1PrintPreviewControl, C1.Win.C1Report.4 C1.Win.C1ReportDesigner.C1ReportDesigner, C1.Win.C1ReportDesigner.4
Ribbon for WinForms	C1.Win.C1Ribbon.C1Ribbon, C1.Win.C1Ribbon.4 C1.Win.C1Ribbon.C1StatusBar, C1.Win.C1Ribbon.4
RibbonPreview for WinForms	C1.Win.C1RibbonPreview.C1RibbonPreview, C1.Win.C1RibbonPreview.4 C1.Win.C1RibbonPreview.C1RibbonPreviewDialog, C1.Win.C1RibbonPreview.4
Scheduler for WinForms	C1.Win.C1Schedule.C1Schedule, C1.Win.C1Schedule.4 C1.Win.C1Schedule.C1Calendar, C1.Win.C1Schedule.4
Sizer for WinForms	C1.Win.C1Sizer.C1Sizer, C1.Win.C1Sizer.4 C1.Win.C1Sizer.C1SizerLight, C1.Win.C1Sizer.4
SpellChecker for WinForms	C1.Win.C1SpellChecker.C1SpellChecker, C1.Win.C1SpellChecker.4
SplitContainer for WinForms	C1.Win.C1SplitContainer.C1SplitContainer, C1.Win.C1SplitContainer.4
SSRSDataSource for WinForms	C1.Win.C1Document.C1SSRSDataSource, C1.Win.C1Document.4
SuperToolTip for WinForms	C1.Win.C1SuperToolTip.C1SuperErrorProvider, C1.Win.C1SuperToolTip.4 C1.Win.C1SuperToolTip.C1SuperLabel, C1.Win.C1SuperToolTip.4 C1.Win.C1SuperToolTip.C1SuperToolTip, C1.Win.C1SuperToolTip.4
Themes for WinForms	C1.Win.C1Themes.C1ThemeController, C1.Win.C1Themes.4
TileControl for WinForms	C1.Win.C1Tile.C1TileControl, C1.Win.C1TileControl.4
TopicBar for WinForms	C1.Win.C1Command.C1TopicBar, C1.Win.C1Command.4
TouchToolkit for WinForms	C1.Win.TouchToolkit.C1ApplicationZoom, C1.Win.C1TouchToolkit.4 C1.Win.TouchToolkit.C1Magnify, C1.Win.C1TouchToolkit.4 C1.Win.TouchToolkit.C1MultiScaleImage, C1.Win.C1TouchToolkit.4

	C1.Win.TouchToolkit.C1TouchEventProvider, C1.Win.C1TouchToolkit.4 C1.Win.TouchToolkit.C1Zoom, C1.Win.C1TouchToolkit.4 C1.Win.TouchToolkit.C1ZoomCommandProvider, C1.Win.C1TouchToolkit.4 C1.Win.TouchToolkit.C1ZoomPanel, C1.Win.C1TouchToolkit.4
TreeView for WinForms	C1.Win.TreeView.C1TreeView, C1.Win.TreeView.4
True DBGrid for WinForms	C1.Win.C1TrueDBGrid.C1TrueDBGrid, C1.Win.C1TrueDBGrid.4
TrueChart for WinForms (2D)	C1.Win.C1Chart.C1Chart, C1.Win.C1Chart.4
TrueChart for WinForms (3D)	C1.Win.C1Chart3D.C1Chart3D, C1.Win.C1Chart3D.4
Windows 7 Control Pack for WinForms	C1.Win.C1Win7Pack.C1TaskbarButton, C1.Win.C1Win7Pack.4 C1.Win.C1Win7Pack.C1TaskDialog, C1.Win.C1Win7Pack.4
Word for WinForms	C1.C1Word.C1WordDocument, C1.C1Word.4
Zip for .NET	なし

ASP.NET Web フォーム用コントロール

Licenses.licx ファイルに記載するライセンス情報は、次の通りです。

使用するコントロール名	Licenses.licx に記載が必要なライセンス情報
Accordion for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Accordion.C1Accordion, C1.Web.Wijmo.Controls.4
AppView for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1AppView.C1AppView, C1.Web.Wijmo.Controls.4
AutoComplete for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1AutoComplete.C1AutoComplete, C1.Web.Wijmo.Controls.4
BarChart for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Chart.C1BarChart, C1.Web.Wijmo.Controls.4
BarCode for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1BarCode.C1BarCode, C1.Web.Wijmo.Controls.4
BinaryImage for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1BinaryImage.C1BinaryImage, C1.Web.Wijmo.Controls.4
BubbleChart for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Chart.C1BubbleChart, C1.Web.Wijmo.Controls.4
Calendar for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Calendar.C1Calendar, C1.Web.Wijmo.Controls.4

CandlestickChart for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Chart.C1CandlestickChart, C1.Web.Wijmo.Controls.4
Carousel for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Carousel.C1Carousel, C1.Web.Wijmo.Controls.4
ChartNavigator for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Chart.C1ChartNavigator, C1.Web.Wijmo.Controls.4
ComboBox for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1ComboBox.C1ComboBox, C1.Web.Wijmo.Controls.4
CompositeChart for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Chart.C1CompositeChart, C1.Web.Wijmo.Controls.4
Dialog for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Dialog.C1Dialog, C1.Web.Wijmo.Controls.4
Editor for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Editor.C1Editor, C1.Web.Wijmo.Controls.4
EventsCalendar for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1EventsCalendar.C1EventsCalendar, C1.Web.Wijmo.Controls.4
Excel for .NET	C1.C1Excel.C1XLBook, C1.C1Excel.4
Expander for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Expander.C1Expander, C1.Web.Wijmo.Controls.4
Extender Controls for ASP.NET Web Forms	C1.Web.Wijmo.Extenders.C1Calendar.C1CalendarExtender, C1.Web.Wijmo.Extenders.4
FileExplorer for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1FileExplorer.C1FileExplorer, C1.Web.Wijmo.Controls.4
FlipCard for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1FlipCard.C1FlipCard, C1.Web.Wijmo.Controls.4
Gallery for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Gallery.C1Gallery, C1.Web.Wijmo.Controls.4
GridView for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1GridView.C1GridView, C1.Web.Wijmo.Controls.4
Input for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Input.C1InputCurrency, C1.Web.Wijmo.Controls.4 C1.Web.Wijmo.Controls.C1Input.C1InputDate, C1.Web.Wijmo.Controls.4 C1.Web.Wijmo.Controls.C1Input.C1InputMask, C1.Web.Wijmo.Controls.4 C1.Web.Wijmo.Controls.C1Input.C1InputNumeric, C1.Web.Wijmo.Controls.4

	C1.Web.Wijmo.Controls.C1Input.C1InputPercent, C1.Web.Wijmo.Controls.4 C1.Web.Wijmo.Controls.C1Input.C1InputText, C1.Web.Wijmo.Controls.4
LightBox for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1LightBox.C1LightBox, C1.Web.Wijmo.Controls.4
LinearGauge for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Gauge.C1LinearGauge, C1.Web.Wijmo.Controls.4
LineChart for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Chart.C1LineChart, C1.Web.Wijmo.Controls.4
ListView for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1ListView.C1ListView, C1.Web.Wijmo.Controls.4
Maps for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Maps.C1Maps, C1.Web.Wijmo.Controls.4
Menu for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Menu.C1Menu, C1.Web.Wijmo.Controls.4
Pager for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Pager.C1Pager, C1.Web.Wijmo.Controls.4
PDF for .NET	C1.C1Pdf.C1PdfDocument, C1.C1Pdf.4
PieChart for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Chart.C1PieChart, C1.Web.Wijmo.Controls.4
ProgressBar for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1ProgressBar.C1ProgressBar, C1.Web.Wijmo.Controls.4
QRCode for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1QRCode.C1QRCode, C1.Web.Wijmo.Controls.4
RadialGauge for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Gauge.C1RadialGauge, C1.Web.Wijmo.Controls.4
Rating for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Rating.C1Rating, C1.Web.Wijmo.Controls.4
ReportViewer for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1ReportViewer.C1ReportViewer, C1.Web.Wijmo.Controls.4
ScatterChart for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Chart.C1ScatterChart, C1.Web.Wijmo.Controls.4
SiteMap for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1SiteMap.C1SiteMap, C1.Web.Wijmo.Controls.4
Slider for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Slider.C1Slider, C1.Web.Wijmo.Controls.4
SparkLine for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Sparkline.C1Sparkline, C1.Web.Wijmo.Controls.4

Splitter for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Splitter.C1Splitter, C1.Web.Wijmo.Controls.4
SuperPanel for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1SuperPanel.C1SuperPanel, C1.Web.Wijmo.Controls.4
Tabs for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Tabs.C1Tabs, C1.Web.Wijmo.Controls.4
ToolTip for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1ToolTip.C1ToolTip, C1.Web.Wijmo.Controls.4
TreeView for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1TreeView.C1TreeView, C1.Web.Wijmo.Controls.4
TrueChart for ASP.NET (2D)	C1.Web.C1WebChart.C1WebChart, C1.Web.C1WebChart.4
TrueChart for ASP.NET (3D)	C1.Web.C1WebChart3D.C1WebChart3D, C1.Web.C1WebChart3D.4
Upload for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Upload.C1Upload, C1.Web.Wijmo.Controls.4
Wizard for ASP.NET Web Forms	C1.Web.Wijmo.Controls.C1Wizard.C1Wizard, C1.Web.Wijmo.Controls.4
Zip for .NET	なし

ASP.NET MVC 用コントロール

Licenses.licx ファイルに記載するライセンス情報は、次の通りです。

使用するコントロール名	Licenses.licx に記載が必要なライセンス情報
MVC コントロール	C1.Web.Mvc.LicenseDetector, C1.Web.Mvc
FinancialChart for ASP.NET MVC	C1.Web.Mvc.LicenseDetector, C1.Web.Mvc C1.Web.Mvc.Finance.LicenseDetector, C1.Web.Mvc.Finance
FlexSheet for ASP.NET MVC	C1.Web.Mvc.LicenseDetector, C1.Web.Mvc C1.Web.Mvc.Sheet.LicenseDetector, C1.Web.Mvc.FlexSheet
FlexViewer for ASP.NET MVC	C1.Web.Mvc.LicenseDetector, C1.Web.Mvc C1.Web.Mvc.Viewer.LicenseDetector, C1.Web.Mvc.FlexViewer C1.Web.Api.Report.LicenseDetector, C1.Web.Api.Report
OLAP for ASP.NET MVC	C1.Web.Mvc.LicenseDetector, C1.Web.Mvc C1.DataEngine.Workspace, C1.DataEngine.4
MVC Classic	なし
Excel for .NET	C1.C1Excel.C1XLBook, C1.C1Excel.4
PDF for .NET	C1.C1Pdf.C1PdfDocument, C1.C1Pdf.4

Web API	C1.Web.Api.Excel.ExcelExporter, C1.Web.Api C1.Web.Api.Image.ImageExporter, C1.Web.Api
Zip for .NET	なし

WPF 用コントロール

Licenses.licx ファイルに記載するライセンス情報は、次の通りです。

使用するコントロール名	Licenses.licx に記載が必要なライセンス情報
Accordion for WPF	C1.WPF.Extended.C1Accordion, C1.WPF.Extended.4
BarCode for WPF	C1.WPF.BarCode.C1BarCode, C1.WPF.BarCode.4
Binding Expressions for WPF	なし
Bitmap for WPF	なし
Book for WPF	C1.WPF.Extended.C1Book, C1.WPF.Extended.4
Calendar for WPF	C1.WPF.Schedule.C1Calendar, C1.WPF.Schedule.4
Carousel for WPF	C1.WPF.Carousel.C1CarouselPanel, C1.WPF.Carousel.4
Chart for WPF	C1.WPF.C1Chart.C1Chart, C1.WPF.C1Chart.4 C1.WPF.C1Chart.C1ChartLegend, C1.WPF.C1Chart.4
Chart3D for WPF	C1.WPF.C1Chart3D.C1Chart3D, C1.WPF.C1Chart3D.4 C1.WPF.C1Chart3D.C1Chart3DLegend, C1.WPF.C1Chart3D.4
ColorPicker for WPF	C1.WPF.Extended.C1ColorPicker, C1.WPF.Extended.4
ComboBox for WPF	C1.WPF.C1ComboBox, C1.WPF.4
DataGrid for WPF	C1.WPF.DataGrid.C1DataGrid, C1.WPF.DataGrid.4
DateTime Editors for WPF	C1.WPF.DateTimeEditors.C1DateTimePicker, C1.WPF.DateTimeEditors.4 C1.WPF.DateTimeEditors.C1DatePicker, C1.WPF.DateTimeEditors.4 C1.WPF.DateTimeEditors.C1TimeEditor, C1.WPF.DateTimeEditors.4
DockControl for WPF	C1.WPF.Docking.C1DockControl, C1.WPF.Docking.4 C1.WPF.Docking.C1DockTabControl, C1.WPF.Docking.4
DragDropManager for WPF	なし
DropDown for WPF	C1.WPF.C1DropDown, C1.WPF.4 C1.WPF.C1DropDownButton, C1.WPF.4
Excel for WPF	なし
Expander for WPF	C1.WPF.Extended.C1Expander, C1.WPF.Extended.4

FinancialChart for WPF	C1.WPF.Chart.Finance.C1FinancialChart, C1.WPF.FinancialChart.4
FlexChart for WPF	C1.WPF.Chart.C1FlexChart, C1.WPF.FlexChart.4
FlexGrid for WPF	C1.WPF.FlexGrid.C1FlexGrid, C1.WPF.FlexGrid.4
FlexPie for WPF	C1.WPF.Chart.C1FlexPie, C1.WPF.FlexChart.4 C1.WPF.Chart.C1FlexRadar, C1.WPF.FlexChart.4 C1.WPF.Chart.C1Sunburst, C1.WPF.FlexChart.4
FlexReport for WPF	なし
FlexSheet for WPF	C1.WPF.FlexGrid.C1FlexSheet, C1.WPF.FlexSheet.4 C1.WPF.FlexGrid.C1FormulaBar, C1.WPF.FlexSheet.4
FlexViewer for WPF	C1.WPF.FlexViewer.C1FlexViewer, C1.WPF.FlexViewer.4 C1.WPF.FlexViewer.C1FlexViewerPane, C1.WPF.FlexViewer.4
GanttView for WPF	C1.WPF.GanttView.C1GanttView, C1.WPF.GanttView.4
Gauges for WPF	C1.WPF.Gauge.C1Knob, C1.WPF.Gauge.4 C1.WPF.Gauge.C1LinearGauge, C1.WPF.Gauge.4 C1.WPF.Gauge.C1RadialGauge, C1.WPF.Gauge.4 C1.WPF.Gauge.C1RegionKnob, C1.WPF.Gauge.4 C1.WPF.Gauge.C1RulerGauge, C1.WPF.Gauge.4 C1.WPF.Gauge.C1SpeedometerGauge, C1.WPF.Gauge.4 C1.WPF.Gauge.C1VolumeGauge, C1.WPF.Gauge.4
HyperPanel for WPF	C1.WPF.C1HyperPanel, C1.WPF.4
Input for WPF	C1.WPF.C1FilePicker, C1.WPF.4 C1.WPF.C1MaskedTextBox, C1.WPF.4 C1.WPF.C1NumericBox, C1.WPF.4 C1.WPF.C1RangeSlider, C1.WPF.4
ListBox for WPF	C1.WPF.C1ListBox, C1.WPF.4 C1.WPF.C1TileListBox, C1.WPF.4
Maps for WPF	C1.WPF.Maps.C1Maps, C1.WPF.Maps.4 C1.WPF.Maps.C1MultiScaleImage, C1.WPF.Maps.4
MediaPlayer for WPF	C1.WPF.MediaPlayer.C1MediaPlayer, C1.WPF.MediaPlayer.4
OLAP for WPF	C1.WPF.Olap.C1OlapChart, C1.WPF.Olap.4 C1.WPF.Olap.C1OlapGrid, C1.WPF.Olap.4 C1.WPF.Olap.C1OlapPage, C1.WPF.Olap.4 C1.WPF.Olap.C1OlapPanel, C1.WPF.Olap.4


OrgChart for WPF	C1.WPF.OrgChart.C1OrgChart, C1.WPF.OrgChart.4
OutlookBar for WPF	C1.WPF.OutlookBar.C1OutlookBar, C1.WPF.OutlookBar.4
PdfViewer for WPF	C1.WPF.PdfViewer.C1PdfViewer, C1.WPF.PdfViewer.4
PropertyGrid for WPF	C1.WPF.Extended.C1PropertyGrid, C1.WPF.Extended.4
RadialMenu for WPF	C1.WPF.C1RadialMenu, C1.WPF.4 C1.WPF.C1RadialPanel, C1.WPF.4
Rating for WPF	C1.WPF.Extended.C1Rating, C1.WPF.Extended.4
Reports for WPF	C1.WPF.C1Report.C1DocumentViewer, C1.WPF.C1Report.4
ReportViewer for WPF	C1.WPF.ReportViewer.C1ReportViewer, C1.WPF.ReportViewer.4
RichTextBox for WPF	C1.WPF.RichTextBox.C1RichTextBox, C1.WPF.RichTextBox.4 C1.WPF.RichTextBox.C1RichTextBoxToolbar, C1.WPF.RichTextBox.Toolbar.4
Scheduler for WPF	C1.WPF.Schedule.C1Scheduler, C1.WPF.Schedule.4
Sparkline for WPF	C1.WPF.Sparkline.C1Sparkline, C1.WPF.Sparkline.4
SpellChecker for WPF	なし
TabControl for WPF	C1.WPF.C1TabControl, C1.WPF.4
Tiles for WPF	C1.WPF.Tile.C1Tile, C1.WPF.Tile.4 C1.WPF.Tile.C1SlideTile, C1.WPF.Tile.4
TileView for WPF	C1.WPF.TileView.C1TileView, C1.WPF.TileView.4
ToolBar for WPF	C1.WPF.Toolbar.C1ToolBar, C1.WPF.Toolbar.4 C1.WPF.Toolbar.C1ToolBarGroup, C1.WPF.Toolbar.4 C1.WPF.Toolbar.C1ToolBarDropDown, C1.WPF.Toolbar.4 C1.WPF.Toolbar.C1ToolBarButton, C1.WPF.Toolbar.C1ToolBar.4 C1.WPF.Toolbar.C1ToolBarSplitButton, C1.WPF.Toolbar.4 C1.WPF.Toolbar.C1ToolBarStrip, C1.WPF.Toolbar.4 C1.WPF.Toolbar.C1ToolBarToggleButton, C1.WPF.Toolbar.4
TreeView for WPF	C1.WPF.C1TreeView, C1.WPF.4
Windows for WPF	C1.WPF.C1Window, C1.WPF.4
Word for WPF	なし
Zip for WPF	なし
WPFBasicLibrary	C1.WPF.C1DockPanel, C1.WPF.4 C1.WPF.C1HyperlinkButton, C1.WPF.4 C1.WPF.C1Menu, C1.WPF.4 C1.WPF.C1MenuList, C1.WPF.4

C1.WPF.C1ProgressBar, C1.WPF.4
C1.WPF.C1ProgressIndicator, C1.WPF.4
C1.WPF.C1SplitButton, C1.WPF.4
C1.WPF.C1ScrollViewer, C1.WPF.4
C1.WPF.C1WrapPanel, C1.WPF.4

配布ガイド

ランタイムファイルの配布について


本製品を使用したアプリケーションを実行するには、実行環境に本製品のランタイムファイルが必要です。そのため、本製品を使用したアプリケーションを配布する場合、ランタイムファイルをアプリケーションと共に配布する必要があります。

 **補足：** 当ドキュメントでは、ランタイムファイルなどの再配布できるファイルを総称して、再配布可能ファイルと呼びます。

Silverlight アプリケーションの場合

Silverlight のアプリケーションは、.xap（ザップと呼びます）という拡張子がついたアプリケーションパッケージ形式となります。これは、実行ファイル（DLL）とアプリケーションマニフェスト（AppManifest.xaml）を zip 形式で圧縮したものです。

Silverlight コンポーネントを使用したアプリケーションを実行するには、.xap ファイル内に Silverlight コンポーネントのランタイムファイルが含まれている必要があります。

 **メモ：** 通常、Visual Studio もしくは Expression Blend 上でプロジェクトのビルドあるいはリビルドを実行すると、実行に必要なアセンブリファイルなどが組み込まれた .xap ファイルが自動で作成されます。ユーザーがアプリケーションマニフェストを直接編集したり、アセンブリファイルを .xap ファイル内に手動で格納する必要はありません。

ランタイムファイルの配布に関する注意

- ランタイムファイルを、サーバーアプリケーションに組み込んで配布する場合、サーバーライセンスが必要です。サーバーライセンスについての詳細は、付属のリリースノート（ReleaseNote_C1Studio.htm）の[ソフトウェア使用許諾契約書]および[ライセンスに関する注意事項]を参照してください。
- ランタイムファイルは、単独で配布することはできません。本製品を用いて開発したアプリケーションとともに配布することのみ許諾されています。

- 本製品に含まれているファイルのうちランタイムファイル以外のファイルは配布できません。
- アプリケーションの運用環境には、製品自体をインストールする必要はありません。

ランタイムファイル

本製品のランタイムファイルは、各ユーザーガイドの「コンポーネントのランタイムファイル」の項を参照してください。

ランタイムファイルの配布方法

ランタイムファイルを配布するには、ランタイムファイルをアプリケーションと同じフォルダに直接コピーします。

ただし、GAC にランタイムファイルがインストールされている場合や、後から GAC へランタイムファイルがインストールされた場合、アプリケーションはローカルフォルダの方ではなく、GAC にインストールされた方のランタイムファイルを優先して参照します。

コンポーネントの登録について

配布先の環境において、プログラムと同じフォルダ内にコンポーネントを配置しそのプログラムだけが独占的にコンポーネントを参照する場合、本製品のアセンブリをグローバルアセンブリキャッシュへ登録する必要はありません。

なお、.NET のマネージコンポーネントを複数のプログラムで共有する場合は、次に示す方法で配布先の環境にコンポーネントを登録する必要があります。

コンポーネントの登録方法

グローバルアセンブリキャッシュへ登録する

Windows Installer を使ってグローバルアセンブリキャッシュにコンポーネントを登録します。登録に際しては、アドミニストレータ権限でログオンしておく必要があります。

コンフィギュレーションファイルに登録する

アプリケーションコンフィギュレーションファイルの<probing>タグにコンポーネントを配置するサブフォルダ名を記述します。

グローバルアセンブリキャッシュおよびアプリケーションコンフィギュレーションファイルの詳細については、Visual Studio ヘルプの「ランタイムがアセンブリを検索する方法」を参照してください。

Windows フォームアプリケーションの配布

ここでは、本製品を使った Windows フォームアプリケーションの配布について3つの項目に分けて説明します。

⚠️注意：

- いずれの方法で配布する場合にも、.NET Framework の再頒布可能パッケージと日本語 LanguagePack が、配布対象環境にあらかじめインストールされていることが必要です。
- .NET Framework の再頒布可能パッケージと日本語 LanguagePack は、Visual Studio のインストールメディアまたは Microsoft 社の Web サイトで再配布可能な実行ファイルとして提供されています。配布方法やライセンスの詳細は、Visual Studio のマニュアルまたは Microsoft 社の Web サイトを参照してください。

XCOPY による配布

XCOPY は、ファイルをフォルダ構造ごとコピーする DOS のコマンドです。しかし、ここでは、単純にファイルをコピーするだけの配布方法を総称して「XCOPY による配布」と呼んでいます。

XCOPY による配布の長所と短所

XCOPY による配布は、他のどの方法よりも単純で簡便です。しかし、その一方で、配布先に .NET Framework が存在しているかどうか、また、存在している場合でもそのバージョンが適切かどうかなどを調べることができません。さらに、必要なファイルを誤って上書きしたときの復旧手段が用意されていません。

これらを考慮すると、XCOPY による配布は、次のような条件が満たされているときに適した方法だと言えます。

- 配布先の環境について、確実に十分な情報があらかじめ用意されている。
- 配布の操作を行う人が、安全にファイルをコピーできる知識と経験を持っている。
- グローバルアセンブリキャッシュやレジストリなどへの登録を必要としない。
- 配布先のマシン台数と予想される更新の頻度が、それほど多くない。

アプリケーションの配布

XCOPY による配布では、本製品の製品インストールフォルダ内の Bin フォルダにある再配布可能なファイルをアプリケーションと同じフォルダにコピーします。これだけで配布が完了します。再配布可能なファイルの一覧は、各製品ヘルプ内の「ファイルの配布について」を参照してください。

実際のコピー操作としては、たとえば次のような方法が考えられます。

- エクスプローラを使ってネットワーク上の対象マシンにコピーする。
- XCOPY コマンドを使った *.bat プログラムを実行し、ネットワーク上の対象マシンにコピーする。

- 配布するファイルを収めた CD-R などを対象マシンに入れ、エクスプローラを使ってコピーする。
- XCOPY コマンドを使った *.bat プログラムと配布するファイルを収めた CD-R などを対象マシンに入れ、*.bat プログラムを実行してコピーする。

インストーラによる配布

本製品のコンポーネントを使用して作成したアプリケーションの配布で、最も一般的な方法はインストーラによる配布です。ここでは、いくつかのインストーラ作成方法を紹介し、その中でもっとも手軽なセットアップウィザードを使う方法について具体的に説明します。

インストーラ作成方法の種類と概要

インストーラには、大きく分けて、Windows Installer を使用するものと、スクリプトを使った従来の手順型とがあります。以下では、Windows Installer を使用する方法について紹介します。

※ Windows Installer の特長

- アプリケーションを削除するときに、他のアプリケーションに悪影響を与えないようにできる。
- 誤って削除したファイルを自動的に再インストールする自動修復機能をサポートしている。
- インストールを中止したときに、コンピュータを元の状態に戻すロールバック機能がある。
- グローバルアセンブリキャッシュに対して、アセンブリを登録、修復、または削除できる。
- 特定のアプリケーション専用のプライベートな場所へアセンブリをインストールできる。

セットアップウィザード

Visual Studio のセットアップウィザードを使うと、基本的なインストーラを短時間に作成することが可能です。設定できる内容は、次の「セットアッププロジェクト」と変わりませんが、基本的な設定だけですむ場合には、より簡単にインストーラを作成できます。セットアップウィザードによるインストーラ作成は、次のような場合に適しています。

- インストーラによるアプリケーション配布を前提にしている。
- 特別なインストール条件やレジストリなどへの登録が必要ない。
- アプリケーションの構成やアセンブリとの依存関係がシンプルである。

起動方法

[新しいプロジェクトの追加] ダイアログボックスで [セットアップ/デプロイメント プロジェクト] - [セットアップウィザード] を選択します。

参考情報

Visual Studio のヘルプの「セットアップウィザード」

セットアッププロジェクト

設定できる内容は、上記の「セットアップウィザード」と同じです。複雑な処理を行なうインストーラを作成する場合は、セットアップウィザードを起動しないで、最初からセットアッププロジェクトを作成しても特に工数が増えるわけではありません。セットアッププロジェクトによるインストーラ作成は、次のような場合に適しています。

- インストーラによるアプリケーション配布を前提にしている。
- 開発者がインストーラ作成に関する十分な知識と経験がある。
- セットアップウィザードが生成する基本構造のほかに多くの項目を設定する必要がある。

起動方法

[新しいプロジェクトの追加] ダイアログボックスで [セットアップ/デプロイメント プロジェクト] - [セットアッププロジェクト] を選択します。

参考情報

Visual Studio .NET のヘルプの「配置の概念」および「チュートリアル : Windows フォームアプリケーションの配置」

その他のインストーラ作成ツール

Visual Studio に付属するインストーラ作成ツールのほかに、いくつかのサードパーティから Windows Installer に対応したインストーラ作成ツールが提供されています。通常、これらのツールは、Visual Studio のインストーラ作成ツールがサポートしていない Windows Installer の機能にも対応しています。このようなツールは、次のような場合に必要になることがあります。

- VS .NET の標準ダイアログではなく、独自の外観を持ったダイアログが必要である。
- 複雑な処理をするインストーラを短時間に開発することが求められている。
- Windows Installer の機能を最大限に活用する必要がある。

ClickOnce による配布

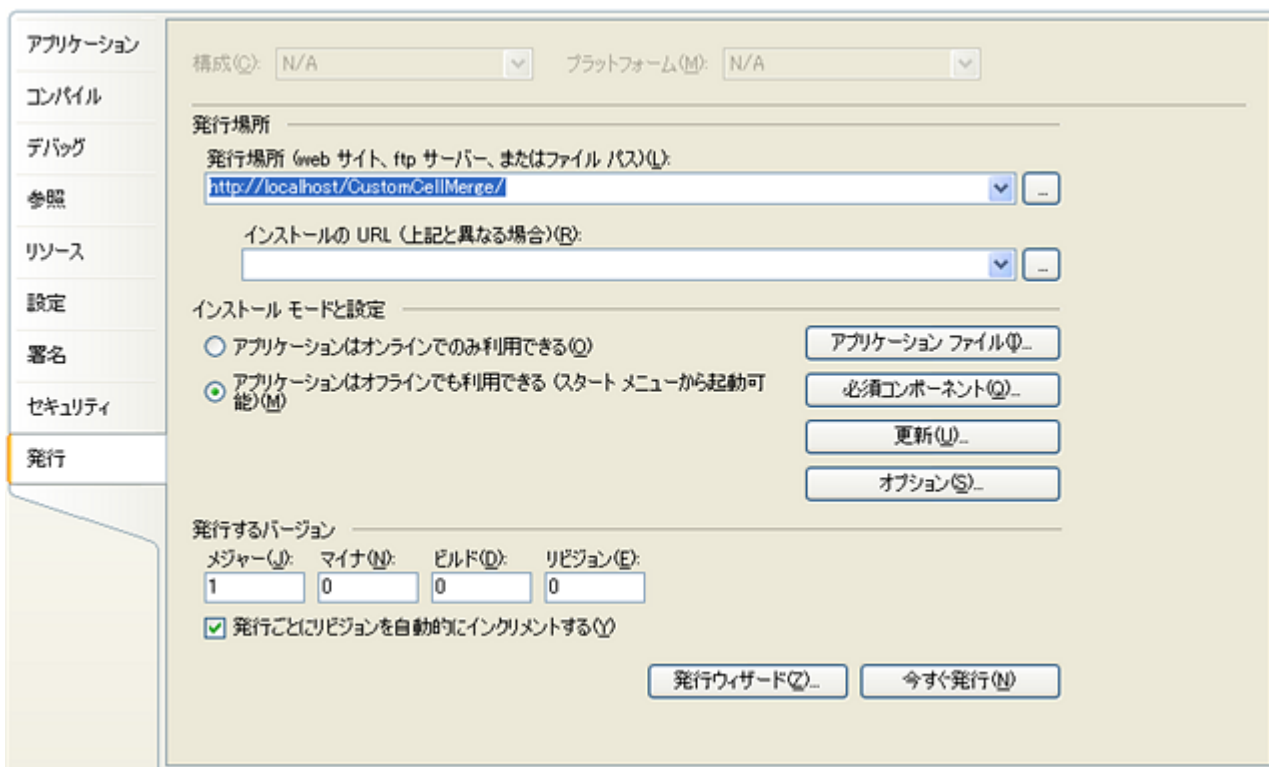
ClickOnce(クリックワンス)は、Visual Studio 2005 以降から提供される新しい配布機能です。Windows フォームアプリケーションを Web ブラウザ経由で配置し、オフラインでの実行を可能にします。ノータッチデプロイメントに対する大きな利点は、事前に .NET Framework 2.0 を配布する必要がないこと、オフラインで実行可能なこと、更新や削除の管理が容易であることなどです。

ここでは、ClickOnce を使用した Windows フォームアプリケーションの配布について、発行、配置、実行、更新、削除の一連のプロセスを説明します。ClickOnce についての詳細は、Visual Studio のドキュメントまたは MSDN、市販の雑誌・解説書を参照してください。

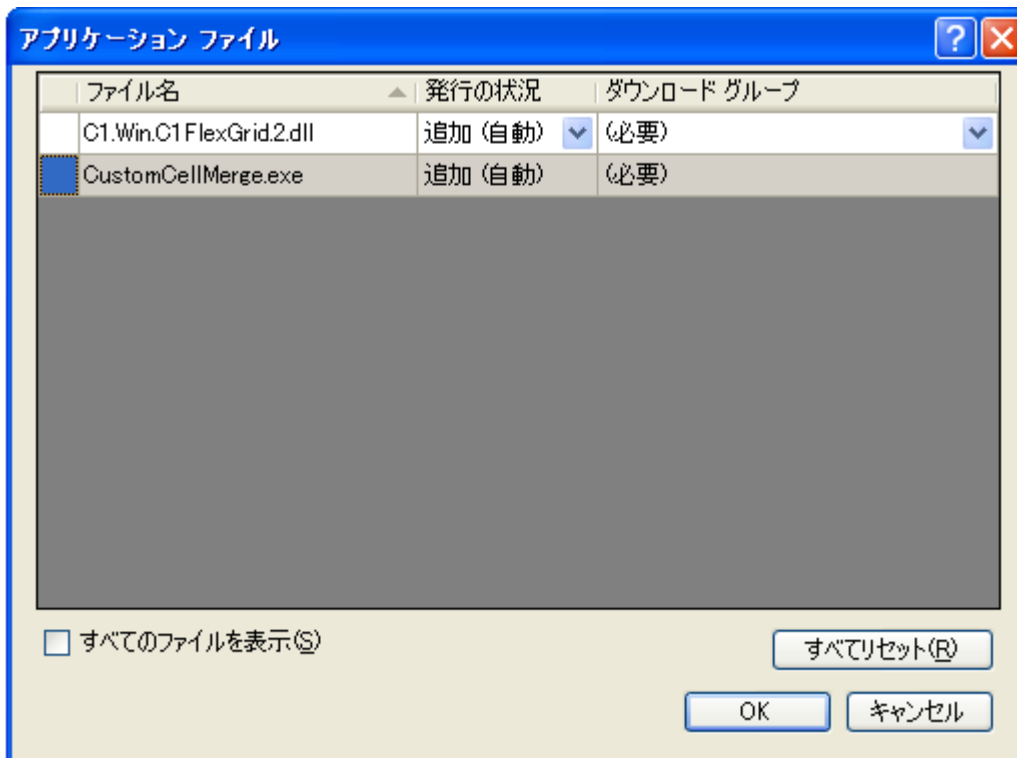
アプリケーションの発行

ClickOnce で Windows フォームアプリケーションを IIS(インターネットインフォメーションサービス)に設置するには、次の手順を実行します。ここでは、**FlexGrid for WinForms** に付属の「CustomCellMerge」サンプルを使用した例を示します。

1. Visual Studio 2005 を起動し、ClickOnce で配布するプロジェクトを開きます。
2. [プロジェクト] メニューから [%プロジェクト名% のプロパティ] を実行します。



3. [アプリケーションファイル] ボタンをクリックし、**FlexGrid for WinForms** のランタイムファイルの設定を表示します。



4. 「すべてのファイルを表示」チェックを ON に変更します。



5. すべての **FlexGrid for WinForms** のランタイムの「発行の状況」を「追加」に変更します。

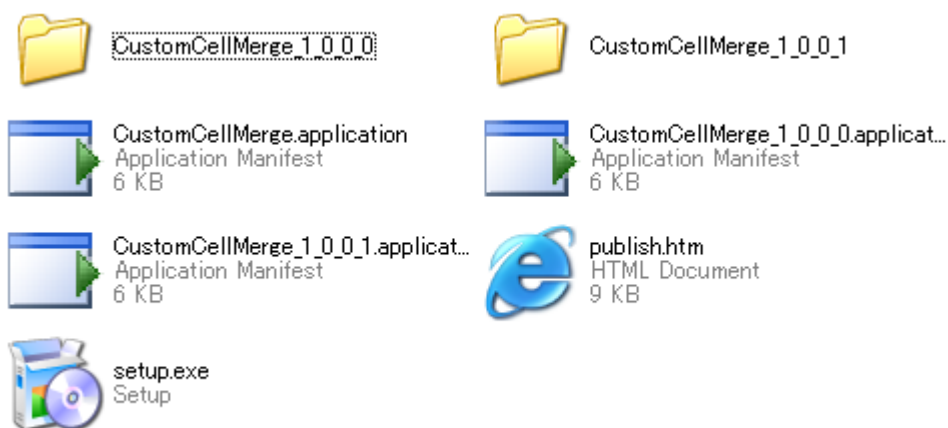


⚠注意： この設定は、実行環境において、**FlexGrid for WinForms** のアセンブリをアプリケーションと同じフォルダに配置させるために必要です。

6. <OK> ボタンをクリックして設定を適用します。
7. [今すぐ発行] をクリックします。

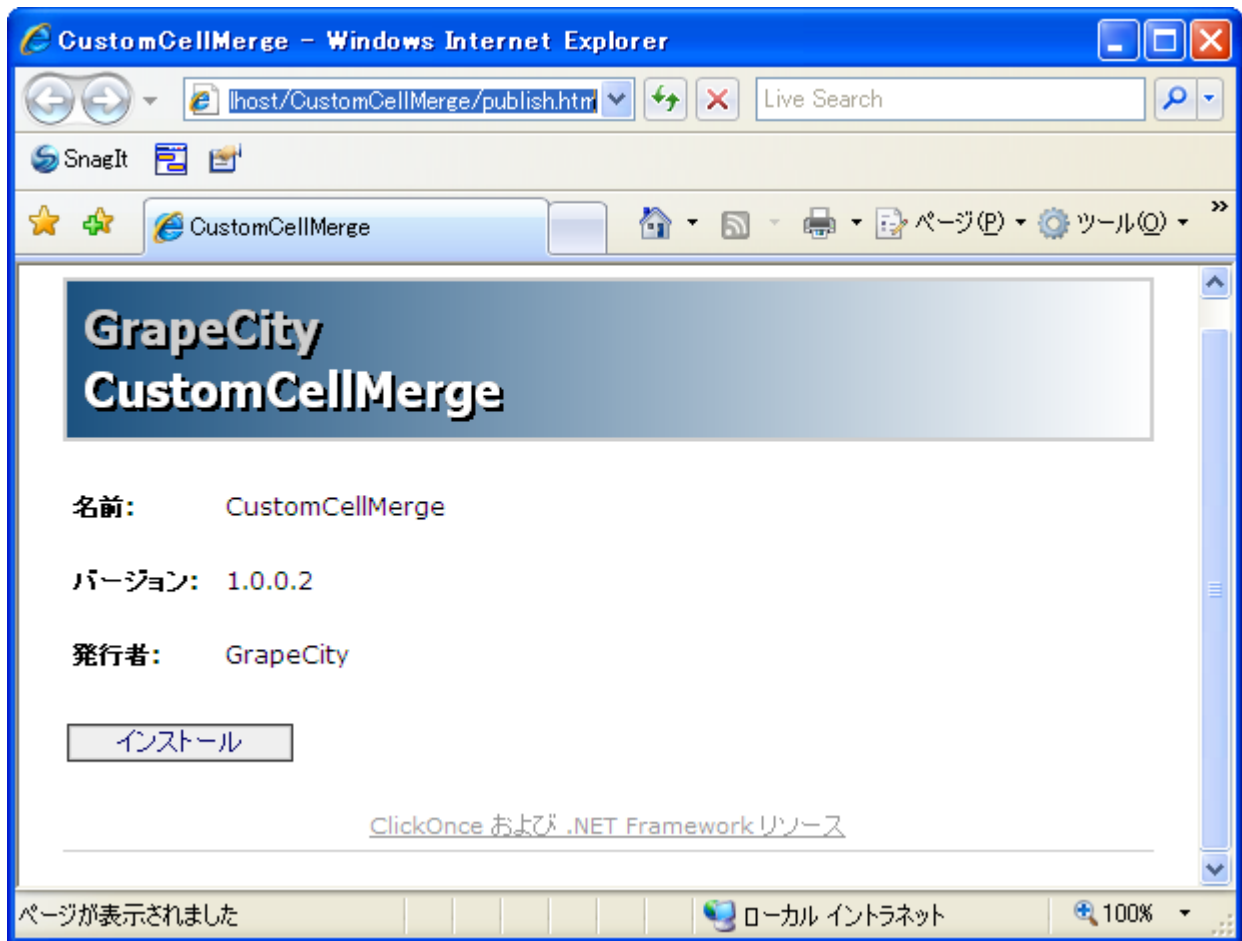
IIS が動作している場合、Visual Studio 2005 によって指定した場所へアプリケーションが発行されます（ここでは <http://localhost/CustomMerge/> となります。）。

IIS のフォルダとプロジェクトの出力フォルダには次のようなファイルがコピーされます。

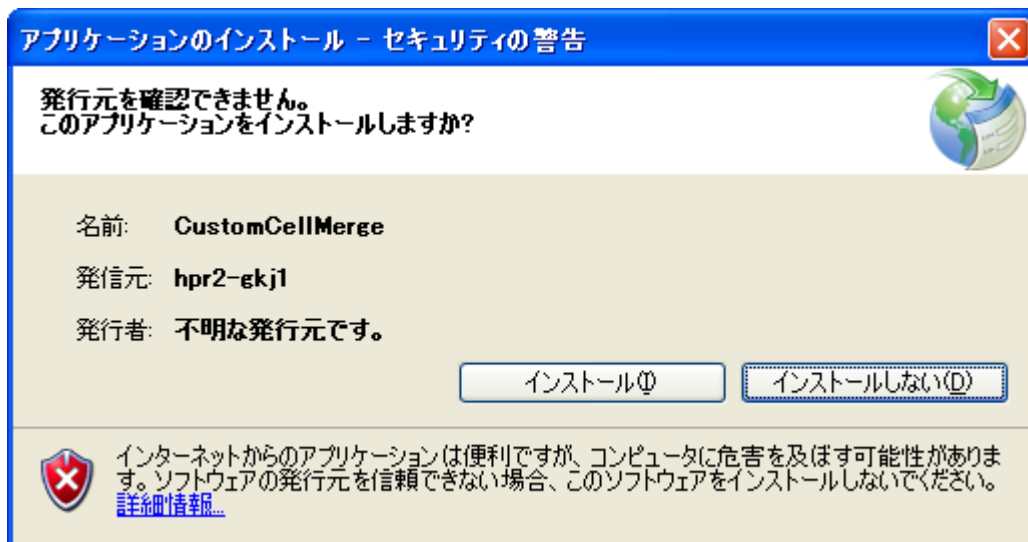


アプリケーションの配置

アプリケーションを発行した URL（例： <http://localhost/CustomCellMerge/publish.htm>）を Internet Explorer で開くと、次のような画面が表示されます。この画面の表示内容は、プロジェクトのプロパティで変更できます。



<インストール>ボタンをクリックすると、アプリケーションの発行元を確認するダイアログが表示されます。



<インストール>ボタンをクリックすると、アプリケーションがクライアント PC にインストールされます。インストールが完了すると、アプリケーションが起動します。

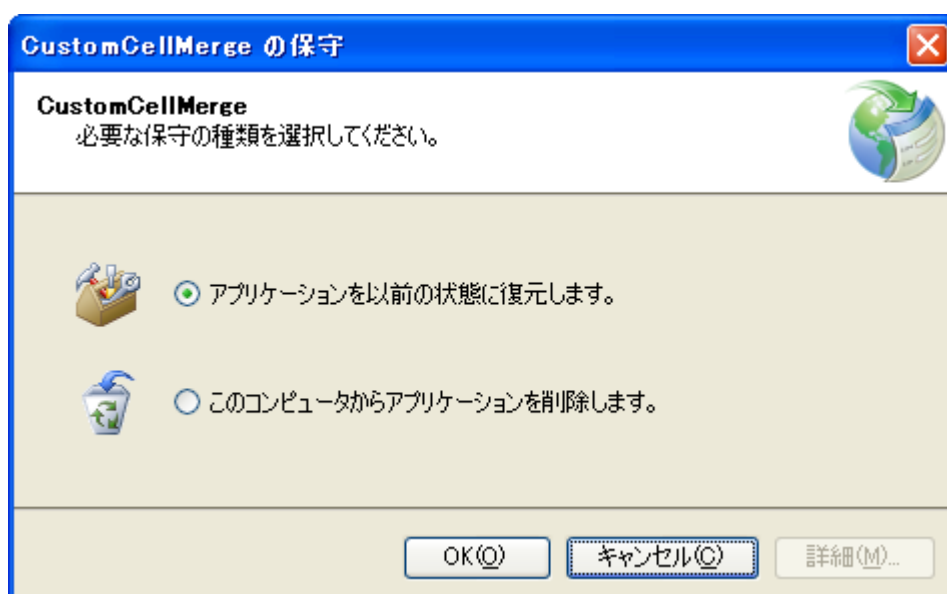
アプリケーションの実行

配置したアプリケーションは、Windows のスタートメニューから実行できます。ClickOnce でスタートメニューに登録するかどうかは、プロジェクトのプロパティで設定します。

アプリケーションの更新

配置したアプリケーションを更新するには、アプリケーションを発行した URL にアクセスし、<インストール>ボタンをクリックします。Web サーバーに新しいバージョンが設置されている場合、自動的に新しいバージョンがインストールされます。

アプリケーションを更新した後以前バージョンに戻すには、「コントロールパネル」の「プログラムの追加と削除」を開き、配置したアプリケーションを選択して<変更と削除>ボタンをクリックします。



「アプリケーションを以前の状態に復元します。」を選択し、<OK>ボタンをクリックするとアプリケーションを以前のバージョンに復元できます。アプリケーションが更新されたバージョンでないとき、「アプリケーションを以前の状態に復元します。」は選択できません。

アプリケーションの削除

配置したアプリケーションを更新するには、「コントロールパネル」の「プログラムの追加と削除」を開き、配置したアプリケーションを選択して<変更と削除>ボタンをクリックします。「このコンピュータからアプリケーションを削除します。」を選択し、<OK>ボタンをクリックするとクライアント PC からアプリケーションを削除できます。

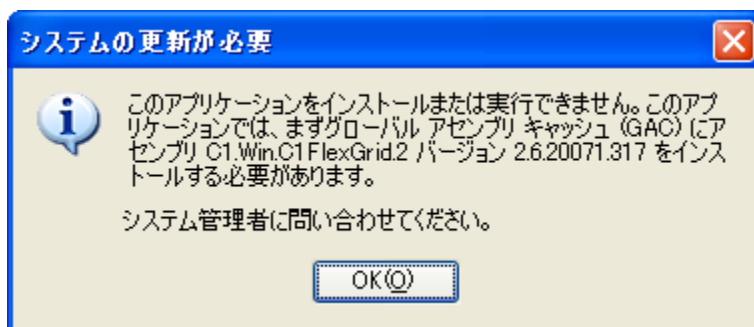


トラブルシューティング

ClickOnce ではほとんどの処理が自動化されているため、現時点では問題が発生する可能性はランタイムの構成とアプリケーションの2点です。

依存ファイル

アプリケーションの依存ファイルの設定が不適切な場合、次のようなエラーメッセージが表示されます。この場合、アプリケーションはインストールに失敗します。これは、ランタイムファイルの設定が「必須コンポーネント(自動)」に設定されていることが原因です。プロジェクトのプロパティを開き、「追加」へ変更してアプリケーションを再発行します。



アプリケーション

ClickOnce は Internet Explorer と IIS の機能や設定に依存しています。他の Web ブラウザや Web サーバーを使用する前に、これらの標準的な組み合わせで検証することで原因を絞り込むことができます。


ASP.NET Web アプリケーションの配布

このセクションでは、本製品を使用した ASP.NET Web アプリケーション（または ASP.NET Web サイト）を配布する際の注意点について説明します。

ASP.NET Web アプリケーションを運用環境に配布するには Web サーバー上に配置する必要があります。Web サーバーにアプリケーションを配置する方法については、MSDN ライブラリの「[ASP.NET Web サイトのビルドと配置](#)」トピックを参照してください。

ここでは、既にアプリケーションが Web サーバーに配置されていることを前提として説明します。

本製品を使用したアプリケーションを正しく動作させるためには、次の手順が必要です。

 **メモ**：本製品に収録されている ASP.NET 用および Windows フォーム/ASP.NET 共用コンポーネントは、ASP.NET アプリケーション上で、Medium 信頼レベルで実行できます。具体的な配置方法については、「[Medium 信頼レベル環境へのアプリケーションの配置](#)」の項目をご確認ください。

アセンブリを配置する


本製品を使用して作成したアプリケーションを配布する場合、各収録コンポーネントのアセンブリファイルをアプリケーションに添付する必要があります。これは、Common Language Runtime の DLL をアプリケーションと一緒に配布するのと同じです。配布先の環境でアプリケーションを正常に動作させるために必要なファイルは、「[コンポーネントの登録について](#)」を参照してください。

Medium 信頼レベル環境へのアプリケーションの配置

ホスティング環境や共有サーバー環境では、ASP.NET Web アプリケーションを Medium 信頼レベル環境に配置しなければならない場合があります。

Medium 信頼レベル環境では、デフォルトでは逆シリアル化やリフレクションなどのアクセスは制限されます。さらに、Medium 信頼レベル環境で厳密な名前を持つアセンブリを利用するためには、アセンブリが AllowPartiallyTrustedCallers 属性を持つ必要があります。

本製品の ASP.NET Web フォーム用コンポーネントは AllowPartiallyTrustedCallers アセンブリ属性を持っていますが、アクセス制限されている逆シリアル化やリフレクションなどの処理を行っているため、Medium 信頼レベルで実行するためには信頼レベルポリシーを変更してアクセス許可を追加する必要があります。

 **メモ**：信頼レベルのポリシーファイルを変更する方法は、MSDN ライブラリの「ASP.NET 2.0 でコード アクセス セキュリティを使用する方法」トピックや「ASP.NET 信頼レベルとポリシー ファイル」トピックで詳しく説明されていますので、こちらも参照してください。

⚠注意 : Web サーバーにアプリケーションを配置する前に、ホスティング環境で信頼レベルポリシーを変更する権限があることを確認してください。

信頼レベルポリシーの変更

アクセス権限を設定するためには、既存の信頼レベルポリシーを変更するか、Medium 信頼レベルをベースにしたカスタム信頼レベルポリシーを作成します。既存の信頼レベルポリシーを変更すると、すべての Web アプリケーションは変更された権限と同じ信頼レベルで実行されます。Medium 信頼レベルポリシーをベースにしたカスタム信頼レベルポリシーを作成すると、個々のアプリケーションに異なる権限を設定することができます。

既存の信頼レベルポリシーの変更

既存の信頼レベルポリシーにアクセス許可を追加するには、以下の手順に従います。

1. Medium 信頼レベルのポリシーファイル「web_mediumtrust.config」を検索します。デフォルトでこのファイルは以下のディレクトリに格納されています。
<%windir%\Microsoft.NET\Framework\{ASP.NET のバージョン}\CONFIG>
2. web_mediumtrust.config ファイルを開きます。
3. 任意のアクセス許可を追加します。詳細については、「[アクセス許可の追加](#)」を参照してください。

カスタム信頼レベルポリシーの作成

Medium 信頼レベルポリシーをベースにしたカスタム信頼レベルポリシーを作成するには、特定のアプリケーションのみにカスタム信頼レベルポリシーを定義する方法と、すべてのアプリケーションから利用できるようにサーバーにカスタム信頼レベルポリシーを定義する方法があります。

アプリケーションにカスタム信頼レベルポリシーを定義する方法

アプリケーションにカスタム信頼レベルポリシーを定義してからアクセス許可を追加するには、以下の手順に従います。

1. Medium 信頼レベルのポリシーファイル「web_mediumtrust.config」を検索します。デフォルトでこのファイルは以下のディレクトリに格納されています。
<%windir%\Microsoft.NET\Framework\{ASP.NET のバージョン}\CONFIG>
2. web_mediumtrust.config をアプリケーションと同じディレクトリ内にコピーして、新しいポリシーファイルを作成します。新しいポリシーファイルを既存の Medium 信頼レベルのポリシーファイルと区別するために、ファイル名を変更します（例 : web_customtrust.config）。

3. アプリケーションの web.config ファイルの <system.web> ノードを以下のように変更して、カスタム信頼レベルポリシーを有効にします。

```
<system.web>
:
<securityPolicy>
<trustLevelname="Custom" policyFile="web_customtrust.config"/>
</securityPolicy>
<trustlevel="Custom" originUrl=""/>
:
</system.web>
```

4. 任意のアクセス許可を追加します。詳細については、「[アクセス許可の追加](#)」を参照してください。

サーバーにカスタム信頼レベルポリシーを定義する方法

サーバーにカスタム信頼レベルポリシーを定義してからアクセス許可を追加するには、以下の手順に従います。

1. Medium 信頼レベルのポリシーファイル「web_mediumtrust.config」を検索します。デフォルトでこのファイルは以下のディレクトリに格納されています。

```
<%windir%¥Microsoft.NET¥Framework¥{ASP.NET のバージョン}¥CONFIG>
```

2. web_mediumtrust.config を 1. と同じディレクトリ内にコピーして、新しいポリシーファイルを作成します。新しいポリシーファイルを既存の Medium 信頼レベルのポリシーファイルと区別するために、ファイル名を変更します（例：web_customtrust.config）。
3. 1. と同じディレクトリにある既定の web.config ファイルの <system.web> ノードを以下のように変更して、サーバーにカスタム信頼レベルポリシーの定義を追加します。

```
<system.web>
:
<securityPolicy>
:
<trustLevelname="Custom" policyFile="web_customtrust.config"/>
:
</securityPolicy>
:
</system.web>
```

4. アプリケーションの web.config ファイルの <system.web> ノードを以下のように変更して、カスタム信頼レベルポリシーを有効にします。

```
<system.web>
:
<trustlevel="Custom" originUrl=""/>
:
</system.web>
```


5. 任意のアクセス許可を追加します。詳細については、「[アクセス許可の追加](#)」を参照してください。

アクセス許可の追加

本製品の ASP.NET Web フォーム用コンポーネントを Medium 信頼レベル環境で実行するためには、信頼レベルのポリシーファイルにいくつかのアクセス許可を追加する必要があります。必要なアクセス許可は、次に示すようにコントロールによって異なります。

コントロール	必要なアクセス許可
C1WebChart	逆シリアル化の許可
C1WebChart3D	リフレクションの許可 アンマネージコード実行の許可 ユーザーインターフェイス操作の許可
C1PDF	逆シリアル化の許可
C1WebReport	リフレクションの許可
C1XLBook	アンマネージコード実行の許可
上記以外のコントロール	逆シリアル化の許可 リフレクションの許可

また、OLE DB データプロバイダを利用する場合、ファイルアクセスを行う場合には、それぞれ OleDbPermission および FileIOPermission のアクセス権限も追加する必要があります。

 **メモ**： 信頼レベルのポリシーファイルを変更する前に、「[信頼レベルポリシーの変更](#)」トピックの手順を実行してください。

逆シリアル化の許可

本製品の ASP.NET Web フォーム用コンポーネントは、Visual Studio により App_Licences.dll に追加されたライセンスデータの逆シリアル化処理を行っています。

逆シリアル化のアクセス許可を追加するには、以下の手順に従います。

1. web_mediumtrust.config ファイルまたは web_mediumtrust.config ファイルをベースとして作成したファイルを開きます。
2. <IPermissionclass="SecurityPermission"> タグ内に SerializationFormatter フラグを追加します。

```
<NamedPermissionSets>
<PermissionSetclass="NamedPermissionSet" version="1" Name="ASP.Net">
<IPermissionclass="SecurityPermission" version="1"
Flags="Assertion, Execution, ControlThread, ControlPrincipal, RemotingConfiguration,
SerializationFormatter"/>
:
</PermissionSet>
</NamedPermissionSets>
```

リフレクションの許可

本製品の ASP.NET Web フォーム用コンポーネントは LicenseManager.Validate() メソッドを使用していますが、このメソッドを実行するためにはリフレクションのアクセス許可が必要です。

リフレクションのアクセス許可を追加するには、以下の手順に従います。

1. web_mediumtrust.config ファイルまたは web_mediumtrust.config ファイルをベースとして作成したファイルを開きます。
<SecurityClasses> タグ内に <SecurityClass> タグを追加します。

```
<SecurityClasses>
<SecurityClassName="ReflectionPermission"
Description="System.Security.Permissions.ReflectionPermission,mscorlib,Version=2.0.0.0,
Culture=neutral,PublicKeyToken=b77a5c561934e089"/>
:
</SecurityClasses>
```

2. <NamedPermissionSets> タグ内に <IPermissionclass="ReflectionPermission"> タグを追加します。

```
<NamedPermissionSets>
<PermissionSetclass="NamedPermissionSet" version="1" Name="ASP.Net">
<IPermissionclass="ReflectionPermission" version="1" Flags="ReflectionEmit,MemberAccess"/>
:
</PermissionSet>
</NamedPermissionSets>
```

アンマネージコード実行の許可

アンマネージコード実行のアクセス許可を追加するには、以下の手順に従います。

1. web_mediumtrust.config ファイルまたは web_mediumtrust.config ファイルをベースとして作成したファイルを開きます。
2. <IPermissionclass="SecurityPermission"> タグ内に UnmanagedCode フラグを追加します。

```
<NamedPermissionSets>
<PermissionSetclass="NamedPermissionSet" version="1" Name="ASP.Net">
<IPermissionclass="SecurityPermission" version="1"
Flags="Assertion, Execution, ControlThread, ControlPrincipal, RemotingConfiguration,
SerializationFormatter, UnmanagedCode"/>
:
</PermissionSet>
</NamedPermissionSets>
```

ユーザーインターフェイス操作の許可

ユーザーインターフェイス操作のアクセス許可を追加するには、以下の手順に従います。

1. web_mediumtrust.config ファイルまたは web_mediumtrust.config ファイルをベースとして作成したファイルを開きます。
2. <NamedPermissionSets> タグ内に <IPermissionclass="UIPermission"> タグを追加します。

```
<NamedPermissionSets>
<PermissionSetclass="NamedPermissionSet" version="1" Name="ASP.Net">
<IPermissionclass="UIPermission" version="1" Window="AllWindows"/>
:
</PermissionSet>
</NamedPermissionSets>
```

OLE DB データプロバイダの許可

Medium 信頼レベル環境ではデフォルトで OleDbPermission アクセスが制限されています。つまり、ADO.NET の OLE DB データプロバイダを使用してデータベースにアクセスできません。

OLE DB データプロバイダのアクセス許可を追加するには、以下の手順に従います。

1. web_mediumtrust.config ファイルまたは web_mediumtrust.config ファイルをベースとして作成したファイルを開きます。
<SecurityClasses> タグ内に <SecurityClass> タグを追加します。

```
<SecurityClasses>
<SecurityClassName="OleDbPermission" Description="System.Data.OleDb.OleDbPermission,
System.Data, Version=2.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089"/>
:
```

```
</SecurityClasses>
```

2. `<NamedPermissionSets>` タグ内に `<IPermissionclass="OleDbPermission">` タグを追加します。

```
<NamedPermissionSets>
<PermissionSetclass="NamedPermissionSet" version="1" Name="ASP.Net">
<IPermissionclass="OleDbPermission" version="1" Unrestricted="true"/>
:
</PermissionSet>
</NamedPermissionSets>
```

ファイルアクセスの許可

Medium 信頼レベル環境ではデフォルトで FileIOPermission アクセスが制限されています。つまり、アプリケーションの仮想ディレクトリ階層にあるファイルにしかアクセスできません。


アプリケーションの仮想ディレクトリの階層外のディレクトリへのファイルアクセス許可を追加するには、以下の手順に従います。

1. web_mediumtrust.config ファイルまたは web_mediumtrust.config ファイルをベースとして作成したファイルを開きます。
2. `<IPermission class="FileIOPermission">` タグ内にディレクトリ名を追加します。

```
<NamedPermissionSets>
<PermissionSetclass="NamedPermissionSet" version="1" Name="ASP.Net">
<IPermission class="FileIOPermission" version="1"
Read="$AppDir$;C:¥Program Files¥Common Files¥C1Studio¥Data"
Write="$AppDir$"
Append="$AppDir$"
PathDiscovery="$AppDir$"
/>
:
</PermissionSet>
</NamedPermissionSets>
```

WPF アプリケーションの配布

ここでは、本製品を使ったアプリケーションの配布について3つの項目に分けて説明します。

 注意 :

- いずれの方法で配布する場合にも、.NET Framework の再頒布可能パッケージと日本語 LanguagePack が、配布対象環境にあらかじめインストールされていることが必要です。

- .NET Framework の再頒布可能パッケージと日本語 LanguagePack は、Visual Studio のインストールメディアまたは Microsoft 社の Web サイトで再配布可能な実行ファイルとして提供されています。配布方法やライセンスの詳細は、Visual Studio のマニュアルまたは Microsoft 社の Web サイトを参照してください。

XCOPY による配布

XCOPY は、ファイルをフォルダ構造ごとコピーする DOS のコマンドです。しかし、ここでは、単純にファイルをコピーするだけの配布方法を総称して「XCOPY による配布」と呼んでいます。

XCOPY による配布の長所と短所

XCOPY による配布は、他のどの方法よりも単純で簡便です。しかし、その一方で、配布先に .NET Framework が存在しているかどうか、また、存在している場合でもそのバージョンが適切かどうかなどを調べることができません。さらに、必要なファイルを誤って上書きしたときの復旧手段が用意されていません。

これらを考慮すると、XCOPY による配布は、次のような条件が満たされているときに適した方法だと言えます。

- 配布先の環境について、確実に十分な情報があらかじめ用意されている。
- 配布の操作を行う人が、安全にファイルをコピーできる知識と経験を持っている。
- グローバルアセンブリキャッシュやレジストリなどへの登録を必要としない。
- 配布先のマシン台数と予想される更新の頻度が、それほど多くない。

アプリケーションの配布

XCOPY による配布では、本製品の製品インストールフォルダ内の Bin フォルダにある再配布可能なファイルをアプリケーションと同じフォルダにコピーします。これだけで配布が完了します。再配布可能なファイルの一覧は、各製品ヘルプ内の「ファイルの配布について」を参照してください。

実際のコピー操作としては、たとえば次のような方法が考えられます。

- エクスプローラを使ってネットワーク上の対象マシンにコピーする。
- XCOPY コマンドを使った *.bat プログラムを実行し、ネットワーク上の対象マシンにコピーする。
- 配布するファイルを収めた CD-R などを対象マシンに入れ、エクスプローラを使ってコピーする。
- XCOPY コマンドを使った *.bat プログラムと配布するファイルを収めた CD-R などを対象マシンに入れ、*.bat プログラムを実行してコピーする。

インストーラによる配布

本製品のコンポーネントを使用して作成したアプリケーションの配布で、最も一般的な方法はインストーラによる配布です。ここでは、いくつかのインストーラ作成方法を紹介し、その中でもっとも手軽なセットアップウィザードを使う方法について具体的に説明します。

インストーラ作成方法の種類と概要

インストーラには、大きく分けて、Windows Installer を使用するものと、スクリプトを使った従来の手順型とがあります。以下では、Windows Installer を使用する方法について紹介します。

※ Windows Installer の特長

- アプリケーションを削除するときに、他のアプリケーションに悪影響を与えないようにできる。
- 誤って削除したファイルを自動的に再インストールする自動修復機能をサポートしている。
- インストールを中止したときに、コンピュータを元の状態に戻すロールバック機能がある。
- グローバルアセンブリキャッシュに対して、アセンブリを登録、修復、または削除できる。
- 特定のアプリケーション専用のプライベートな場所へアセンブリをインストールできる。

セットアップウィザード

Visual Studio のセットアップウィザードを使うと、基本的なインストーラを短時間に作成することが可能です。設定できる内容は、次の「セットアッププロジェクト」と変わりませんが、基本的な設定だけですむ場合には、より簡単にインストーラを作成できます。セットアップウィザードによるインストーラ作成は、次のような場合に適しています。

- インストーラによるアプリケーション配布を前提にしている。
- 特別なインストール条件やレジストリなどへの登録が必要ない。
- アプリケーションの構成やアセンブリとの依存関係がシンプルである。

起動方法

[新しいプロジェクトの追加] ダイアログボックスで [セットアップ/デプロイメント プロジェクト] - [セットアップウィザード] を選択します。

参考情報

Visual Studio のヘルプの「セットアップウィザード」

セットアッププロジェクト

設定できる内容は、上記の「セットアップウィザード」と同じです。複雑な処理を行うインストーラを作成する場合は、セットアップウィザードを起動しないで、最初からセットアッププロジェクトを作成しても特に工数が増えるわけではありません。セットアッププロジェクトによるインストーラ作成は、次のような場合に適しています。

- インストーラによるアプリケーション配布を前提にしている。
- 開発者がインストーラ作成に関する十分な知識と経験がある。
- セットアップウィザードが生成する基本構造のほかに多くの項目を設定する必要がある。

起動方法

[新しいプロジェクトの追加] ダイアログボックスで [セットアップ/デプロイメント プロジェクト] - [セットアッププロジェクト] を選択します。

参考情報

Visual Studio .NET のヘルプの「配置の概念」および「チュートリアル : Windows フォームアプリケーションの配置」

その他のインストーラ作成ツール

Visual Studio に付属するインストーラ作成ツールのほかに、いくつかのサードパーティから Windows Installer に対応したインストーラ作成ツールが提供されています。通常、これらのツールは、Visual Studio のインストーラ作成ツールがサポートしていない Windows Installer の機能にも対応しています。このようなツールは、次のような場合に必要になることがあります。

- VS .NET の標準ダイアログではなく、独自の外観を持ったダイアログが必要である。
- 複雑な処理をするインストーラを短時間に開発することが求められている。
- Windows Installer の機能を最大限に活用する必要がある。

ClickOnce による配布

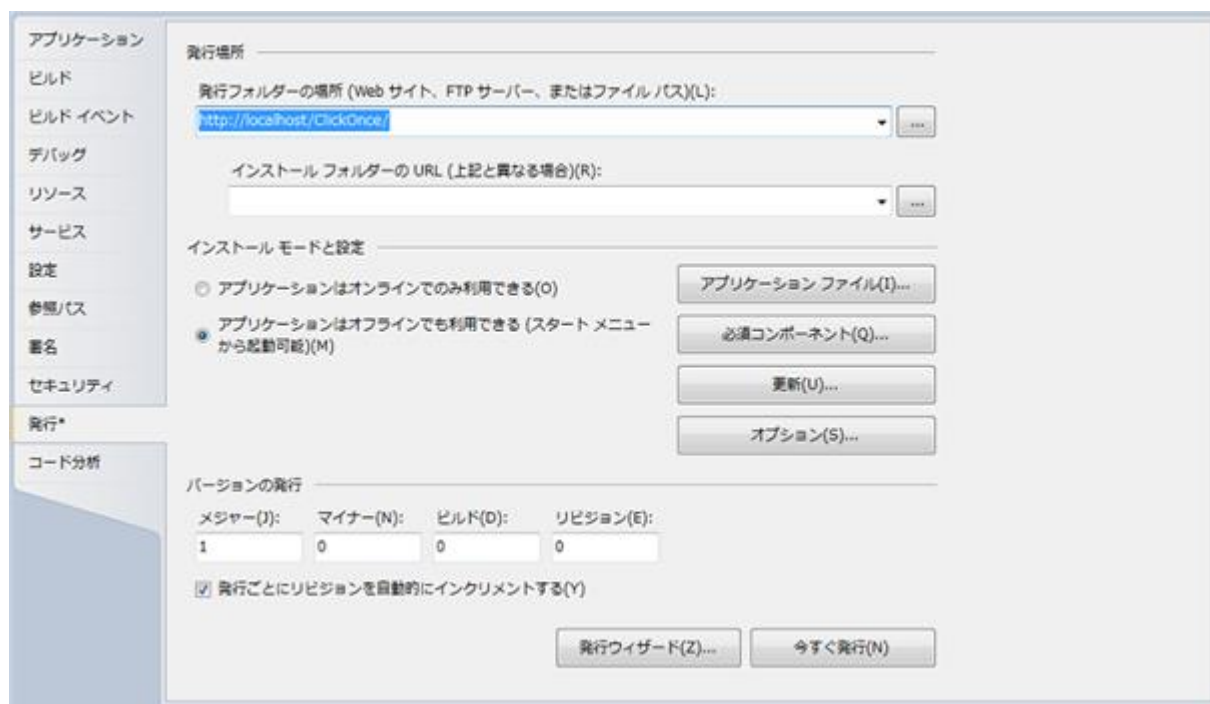
ClickOnce(クリックワンス)は、Visual Studio 2005 以降から提供される新しい配布機能です。Windows フォームアプリケーションを Web ブラウザ経由で配置し、オフラインでの実行を可能にします。ノータッチデプロイメントに対する大きな利点は、事前に .NET Framework 2.0 を配布する必要がないこと、オフラインで実行可能なこと、更新や削除の管理が容易であることなどです。

ここでは、ClickOnce を使用した Windows フォームアプリケーションの配布について、発行、配置、実行、更新、削除の一連のプロセスを説明します。ClickOnce についての詳細は、Visual Studio のドキュメントまたは MSDN、市販の雑誌・解説書を参照してください。

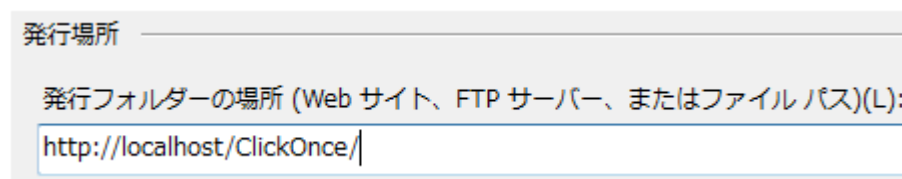
アプリケーションの発行

ClickOnce で Windows フォームアプリケーションを IIS（インターネット インフォメーション サービス）に設置するには、次の手順を実行します。なお、この手順では、ローカル IIS の「ClickOnce」仮想フォルダに対して発行することを想定しています。

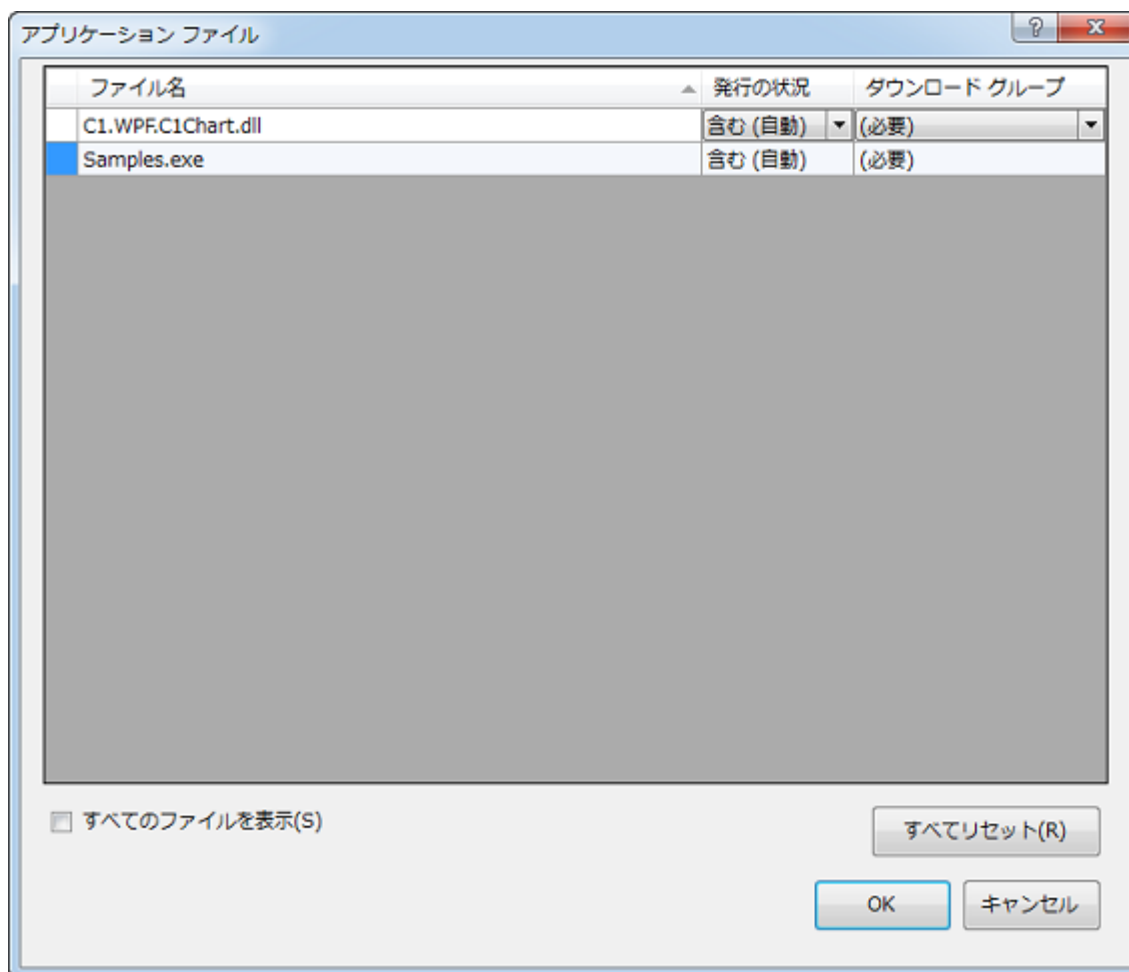
1. Visual Studio を起動し、ClickOnce で配布するプロジェクトを開きます。
2. [プロジェクト] メニューから [%プロジェクト名% のプロパティ] を実行します。次にプロパティ画面内の「発行」タブを選択します。



3. 「発行フォルダの場所」に、アプリケーションを発行する場所（以下の例では http://localhost/ClickOnce/ 仮想フォルダ）を指定します。



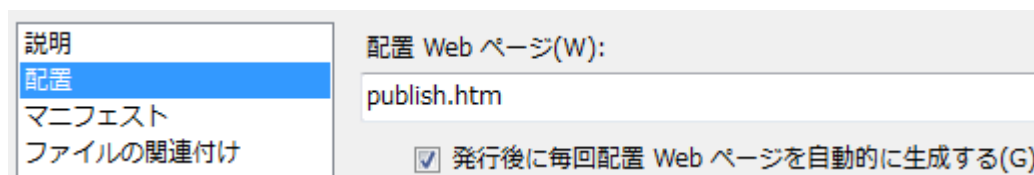
4. [アプリケーションファイル] ボタンをクリックし、アプリケーションで参照しているランタイムファイルがすべて含まれていることを確認します。



アプリケーションで参照していないテーマテンプレート dll や XML ドキュメントファイルが一覧内に含まれている場合は、それらに対する「発行の状況」設定を「除外」に変更します。

確認および変更が完了したら、[アプリケーションファイル]ダイアログを閉じます。

5. [オプション]ボタンをクリックして、[発行オプション]ダイアログを表示します。
6. 左側のペインから[配置]を選択し、「配置 Web ページ」に任意の html ファイル名（以下の例では publish.htm）を設定し、「発行後に毎回配置 Web ページを自動的に生成する」チェックボックスを ON に設定します。

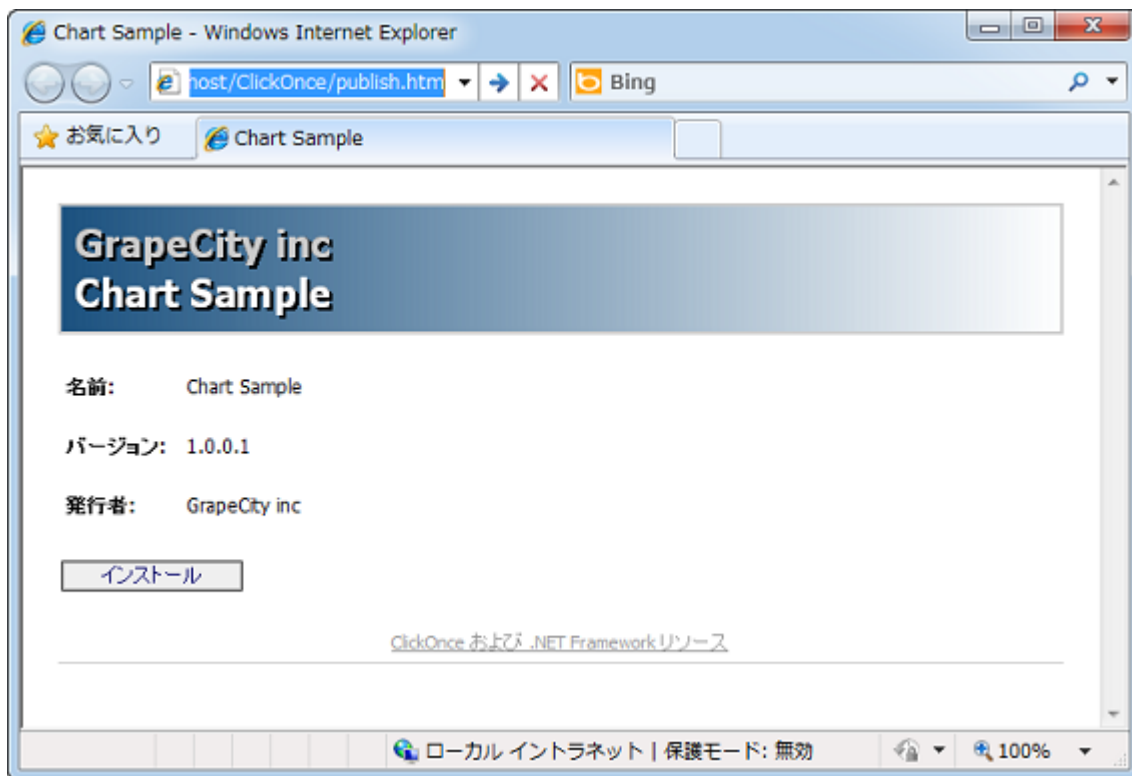


7. [発行オプション]ダイアログを閉じ、[今すぐ発行]ボタンをクリックします。

IIS が動作している場合、Visual Studio によって指定した場所へアプリケーションが発行されます。

アプリケーションの配置

アプリケーションを発行した URL（上記の例では、<http://localhost/ClickOnce/publish.htm>）を Internet Explorer で開くと、次のような画面が表示されます。この画面の表示内容は、プロジェクトのプロパティで変更できます。



[インストール]ボタンをクリックすると、アプリケーションの発行元を確認するダイアログが表示されます。



⚠注意： 上記の例では、アプリケーションが署名されていないため、セキュリティの警告が表示されます。アプリケーションが署名されている場合、表示される画面は上記とは異なります。

[インストール]ボタンをクリックすると、アプリケーションがクライアント PC にインストールされます。インストールが完了すると、アプリケーションが起動します。

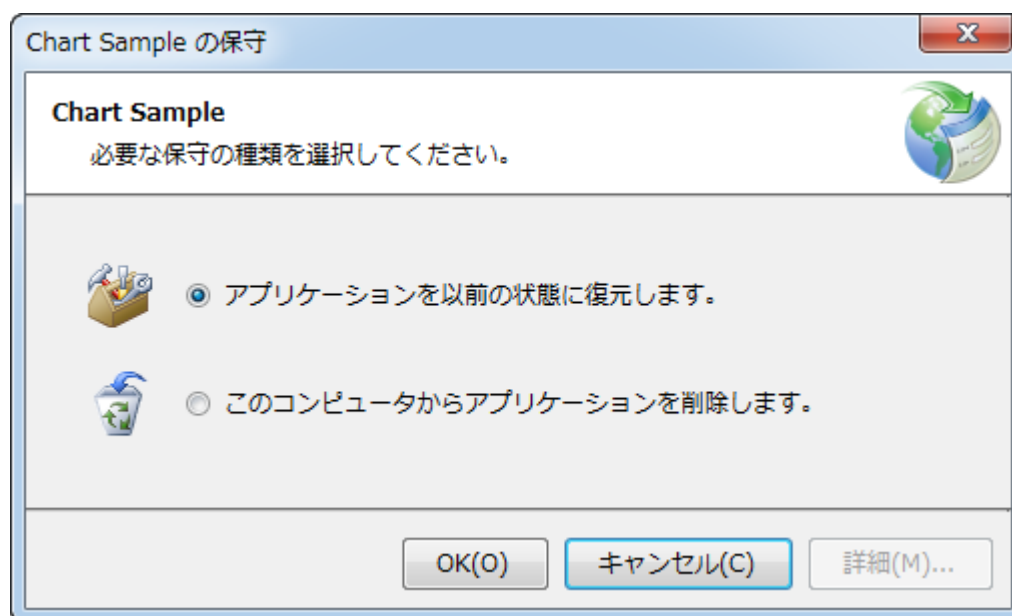
アプリケーションの実行

配置したアプリケーションは、Windows のスタートメニューから実行できます。ClickOnce でスタートメニューに登録するかどうかは、プロジェクトのプロパティで設定します。

アプリケーションの更新

配置したアプリケーションを更新するには、アプリケーションを発行した URL にアクセスし、[インストール]ボタンをクリックします。Web サーバーに新しいバージョンが設置されている場合、自動的に新しいバージョンがインストールされます。

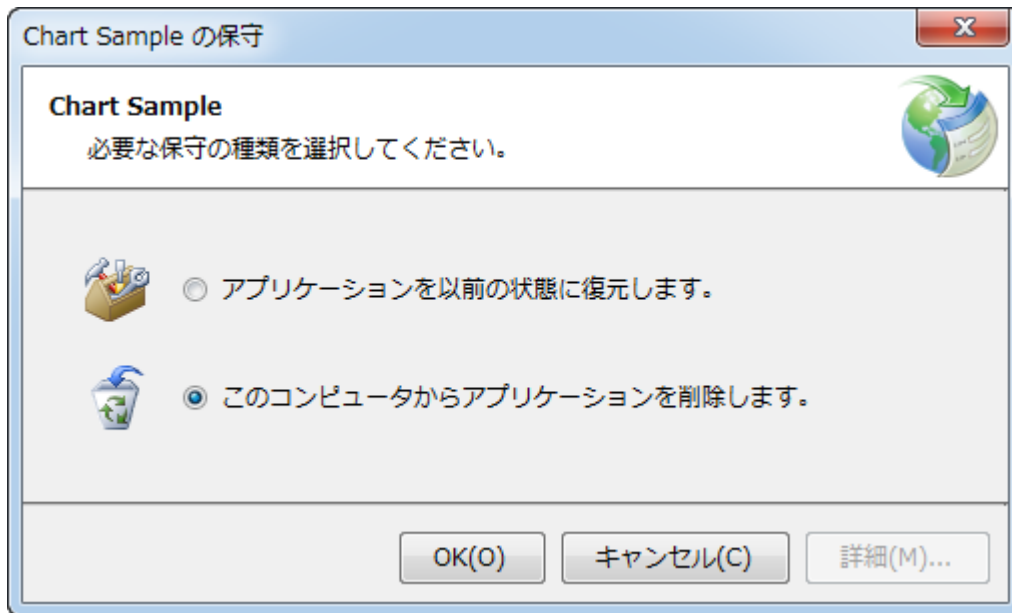
アプリケーションを更新した後に以前のバージョンに戻すには、「コントロールパネル」の「プログラムの追加と削除」を開き、配置したアプリケーションを選択して[変更と削除]ボタンをクリックします。



保守画面にて「アプリケーションを以前の状態に復元します。」を選択し、[OK] ボタンをクリックするとアプリケーションを以前のバージョンに復元できます。アプリケーションが更新されたバージョンでないとき、「アプリケーションを以前の状態に復元します。」は選択できません。

アプリケーションの削除

配置したアプリケーションを更新するには、「コントロールパネル」の「プログラムの追加と削除」を開き、配置したアプリケーションを選択して [変更と削除] ボタンをクリックします。



保守画面にて「このコンピュータからアプリケーションを削除します。」を選択し、[OK] ボタンをクリックするとクライアント PC からアプリケーションを削除できます。

トラブルシューティング

ClickOnce は Internet Explorer と IIS の機能や設定に依存しています。他の Web ブラウザや Web サーバーを使用する前に、これらの標準的な組み合わせで検証することで原因を絞り込むことができます。

Silverlight アプリケーションの配布

Silverlight アプリケーションを配布する際に必要な操作は、作成した Silverlight アプリケーションパッケージ (.xap ファイル) を Web サーバーにコピーし、テスト ページ (***/TestPage.html など) が Silverlight アプリケーションパッケージを呼び出す場合と同じ方法で Web ページから .xap ファイルを呼び出すことです。

メモ : アプリケーションの配置について、詳しくは MSDN の「[配置とローカライゼーション](#)」を参照してください。

Silverlight アプリケーションパッケージの作成

Silverlight アプリケーションパッケージは、Silverlight アプリケーションプロジェクトのビルド時に生成されます。アプリケーション パッケージは、.xap という拡張子を持つ、圧縮された zip ファイルです。アプリケーションを起動するために必要なすべてのファイルが、このパッケージに格納されています。

本製品の Silverlight コンポーネントを使用したアプリケーションの場合、プロジェクトのビルド時に、アプリケーションの実行に必要なアセンブリが自動的に抽出され .xap ファイルに格納されるので、ユーザーはどのファイル

を .xap ファイルに含めるか調べたり、あるいは手動で .xap ファイルにアセンブリを登録したりする必要がありません。

Silverlight アプリケーションの配置

Web サーバー上にアプリケーションを配置する場合は、xcopy を使用したり、Visual Studio 2010 の Web 発行機能などを使用したりできます。

xcopy を使用して Silverlight アプリケーションを展開するには、以下の手順にしたがいます。

1. Visual Studio 上で Silverlight のビルドあるいはリビルドを実行します。このとき、以下のファイルが bin¥Debug¥ もしくは bin¥Release¥ フォルダに作成されます（「***」はプロジェクトの名前を指します）。

ファイル	説明
AppManifest.xaml	AppManifest.xml ファイルの Silverlight バージョン。
***TestPage.html	Silverlight アプリケーション (.xap ファイルに含まれています) をブラウザに読み込む Web ページ。
***.dll	プロジェクトのコンテンツファイルでデザインしたアプリケーションのコンパイル済みアセンブリと、Silverlight コンポーネントのアセンブリ。
***.pdb	アプリケーションのデバッグ情報。
***.xap	Silverlight アプリケーションのすべてのコンテンツとリソースを含む圧縮済みアーカイブ。ユーザーはブラウザでこのファイルをダウンロードし、Silverlight プラグインを使用して実行します。
その他のファイル	アプリケーションで参照する各種リソースファイル。

2. .xap ファイルを、Web サイト上のフォルダにコピーします。例えば、インターネットインフォメーションサービス (IIS) がインストールされている場合は、既定のサイトの仮想ディレクトリに .xap ファイルをコピーできます。このディレクトリは、通常、C:¥Inetpub¥wwwroot にマップされます。
3. 展開をテストするには、テスト ページ (***)TestPage.html) をコピーしたり、テスト ページから Web フォルダ内の既存の Web ページへコードをコピーしたりできます。
4. Silverlight アプリケーションを呼び出す Web ページを参照して、アプリケーションがどのように表示されるかをテストします。

MIME の設定

Web サーバー上に .xap ファイルを配置した場合は、クライアントからそのファイルに適切にアクセスできるように、以下のような内容で MIME の設定を行う必要があります。

項目	設定値
拡張子	.xap

MIME の種類 application/x-silverlight-app

📌メモ： MIME の設定方法については、ご利用の Web サーバーのリファレンス等を参照してください。

Windows ストアアプリケーションの配布

Windows ストアアプリケーションは、次の 2 通りの方法で配布することができます。

- [Windows ストアに掲載する](#)
- [サイドローディングで配布する](#)

配布方法の詳細につきましては、MSDN ライブラリや TechNet ライブラリを参照してください。

移行ガイド

製品（アセンブリ）の更新とメジャーリリース/マイナーリリース

ComponentOne Studio は、メジャーリリース時にインストーラーを更新します。[メジャーリリース](#)後に不具合修正や機能追加が行われた場合は、次メジャーバージョンリリース前にアセンブリのアップデートが行われることがあります。これを[マイナーリリース](#)と言います。マイナーリリースでアップデートされたアセンブリは ComponentOne Studio Live（製品付属のアセンブリ更新ツール）でのみ入手することができます。

製品（アセンブリ）の更新手順

1. 新しい製品（アセンブリ）の入手

以下いずれかの方法で新しいモジュールを入手します。

○ [インストーラの使用](#)

インストーラによるアセンブリの更新（メジャーリリースのアセンブリが対象）

○ [ComponentOne Studio Live の使用](#)

専用ツールによるアセンブリの更新（メジャーリリースおよびマイナーリリースのアセンブリが対象）

2. 移行処理

(1) 旧バージョンからの移行およびサービスパックの適用は、製品付属の移行ツールによって自動的に行うことができます。

- [ComponentOne Studio プロジェクト移行ツール](#)


必要に応じて以下の移行ツールも実行してください。

- [チャートデータ移行ツール](#)
- [C1Report 移行ツール](#)

(2) ツールボックスへ追加したコンポーネントの再登録（任意）


Visual Studio のツールボックスに追加されているコンポーネントがあれば、最新バージョンのインストール後にツールボックスから対象コンポーネントを削除して、ツールボックスへの再登録をする必要があります。

（ComponentOne Studio WinForms および ComponentOne Studio WPF の場合）

 注意：

- 上記手順を実行しても正常にツールボックスにインストールしたコンポーネントが追加されない場合や「コンポーネント～～を生成できませんでした」というエラーが発生する場合は以下の手順を実行してキャッシュを削除します。
- 以下の方法ではツールボックスの全ての項目がリセットされます。

1. Visual Studio のツールボックスで右クリックして、[ツールボックスのリセット] を選択します。
2. Visual Studio を終了します。
3. 後述するフォルダに「*.cache」というファイルが存在すれば、これを削除します。
4. Visual Studio を起動して、ツールボックスにコンポーネントを再登録します。

 「*.cache」の場所：C:\%Users%\<ユーザー名>%AppData%\Roaming%\Microsoft%\VisualStudio%9.0%

- ファイル名は、WinForms では「DotNetComponents.cache」、WPF では「WPFComponents.cache」となります。
- 上記パスは Visual Studio 2008 の場合です。

Visual Studio 2010/2012/2013/2015/2017 では最後の数字がそれぞれ 10.0/11.0/12.0/14.0/15.0 になります。

その他の移行関連情報

- [関連製品との共存](#)
- [互換性について](#)

インストーラの使用

各メジャーリリース時にインストーラが更新されるため、インストーラを使用した製品の更新は各メジャーリリース時のアセンブリとなります。

インストーラのダウンロードと製品インストールの方法は [Web サイト](#) のトピックを参照してください。インストーラをダウンロードし、更新インストールを行うことで最新版へアップデートすることができます。

ComponentOne Studio Live の使用

ComponentOne Studio にはアセンブリ更新ツール「ComponentOne Studio Live（以下、C1Live）」が収録されています。このツールを使用することでアセンブリを最新状態にアップデートすることができます。

 メモ : C1Live は ComponentOne Studio 2013J 以降のバージョンに収録されています。

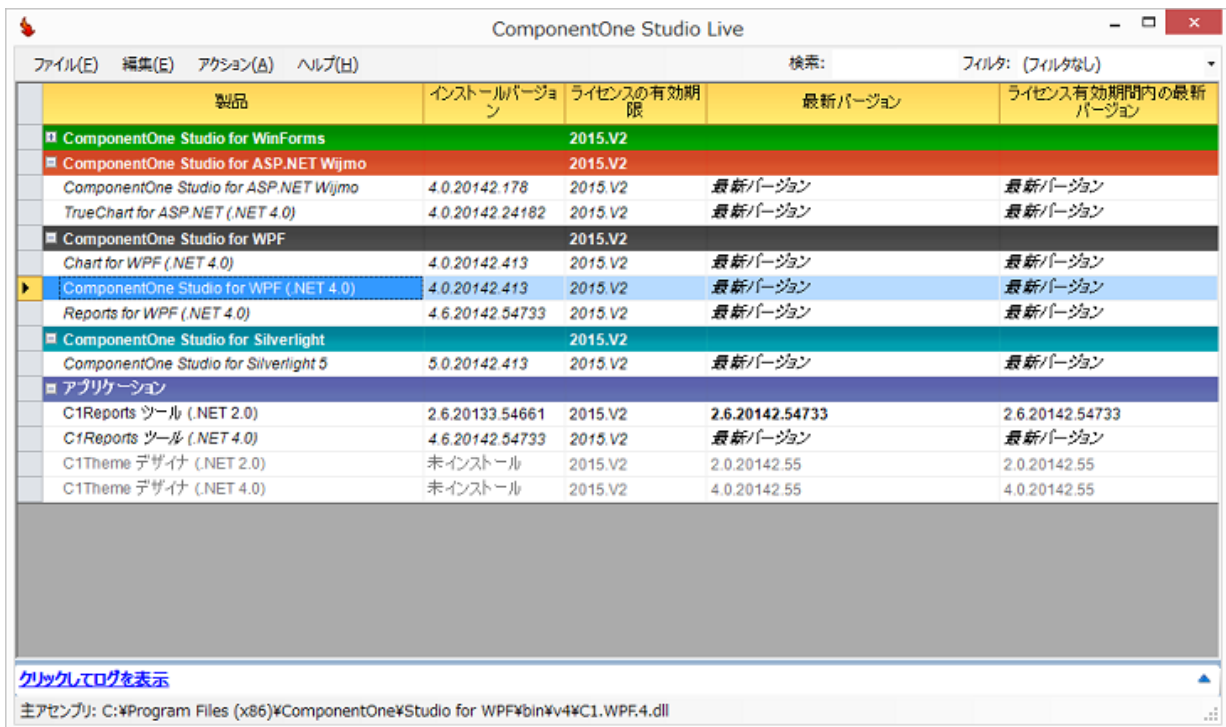
C1Live できること

- 利用中のアセンブリバージョンを確認する
- アセンブリを最新バージョンに更新する（マイナーリリースを含む）
- 更新履歴を確認する
- 任意のアセンブリバージョン（アーカイブ）をダウンロードする

C1Live の使い方：一覧の見方

製品	製品（エディション）ごとに各コントロール名が表示されています。+/-をクリックすることで展開/縮小を行えます。
インストールバージョン	現在インストールされているアセンブリのバージョンが表示されています。
ライセンスの有効期限	ライセンス認証時に使用したシリアルナンバーに対応したライセンスの有効期限が表示されています。
最新バージョン	この項目ではライセンス有効期限に関係なく、公開されている最新のバージョンを確認することができます。※ライセン

	<p>ス有効期限外のバージョンに更新した場合、そのコントロールはトライアル版として動作します。</p> <p>“最新バージョン”と表示されている場合はインストールされているアセンブリが最新の状態になっていることを示しています（インストールされているアセンブリと公開されている最新のバージョンが一致）。アセンブリバージョンが表示されている場合はインストールされているアセンブリよりも新しいバージョンのアセンブリが公開されていることを示します。</p>
<p>ライセンス有効期間の最新バージョン</p>	<p>この項目ではライセンス有効期間内での最新のバージョンを確認することができます。</p> <p>“最新バージョン”と表示されている場合はインストールされているアセンブリがライセンス有効期間内で使用できるアセンブリの中で最新の状態になっていることを示しています。アセンブリバージョンが表示されている場合はインストールされているアセンブリよりも新しいバージョンのアセンブリ（ライセンス有効期間内で使用できる）が公開されていることを示します。</p>

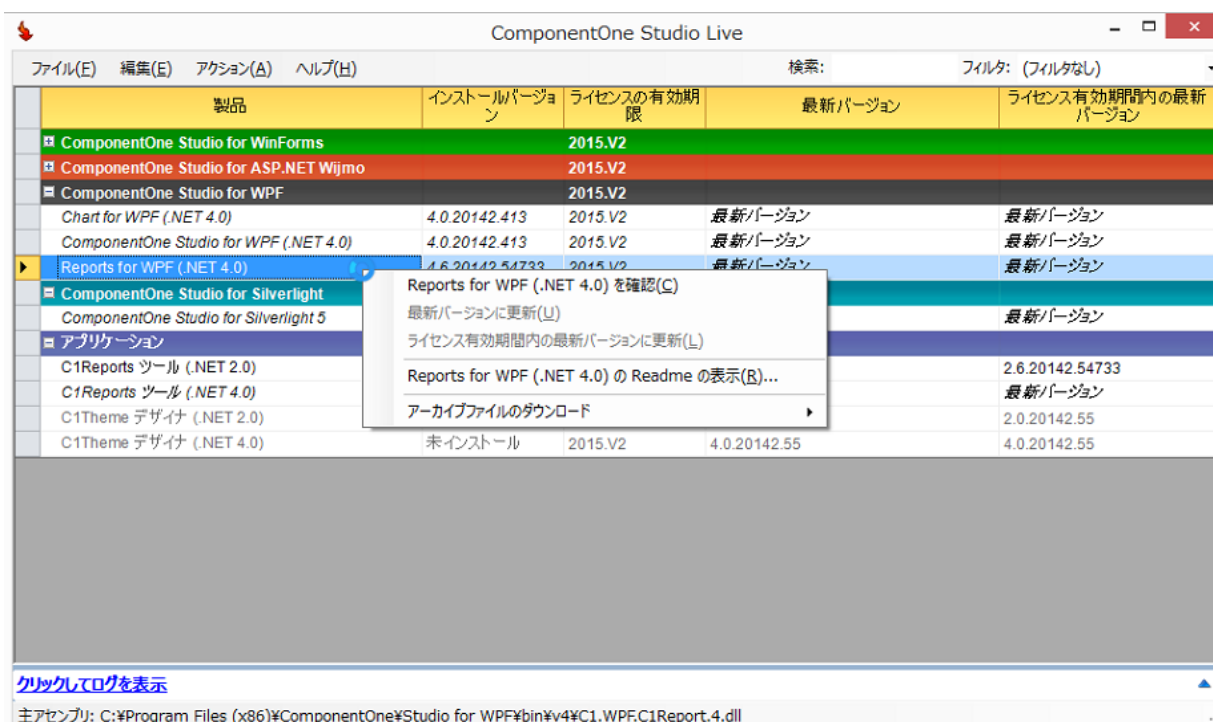


※行ヘッダの矢印は選択行を示します。

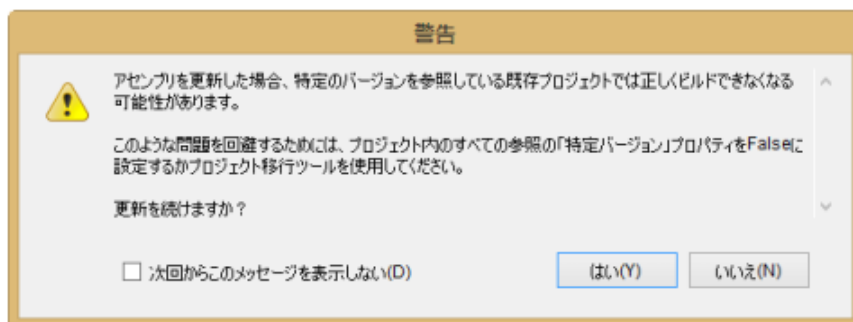
※C1Live ではインストールしていない製品（エディション）も一覧に表示します。

C1Live の使い方 : アセンブリのアップデート方法

1. スタートメニューより[ComponentOne Studio]-[ツール]-[C1Live]をクリックし、C1Live を起動します
2. 一覧で、更新したい製品（エディション）が選択されていることを確認します
3. C1Live のメニューより[アクション]-[ComponentOne Studio for xxx を最新バージョンに更新]をクリックします
(xxx はエディション名)



4. メッセージを確認し、[OK]を押します



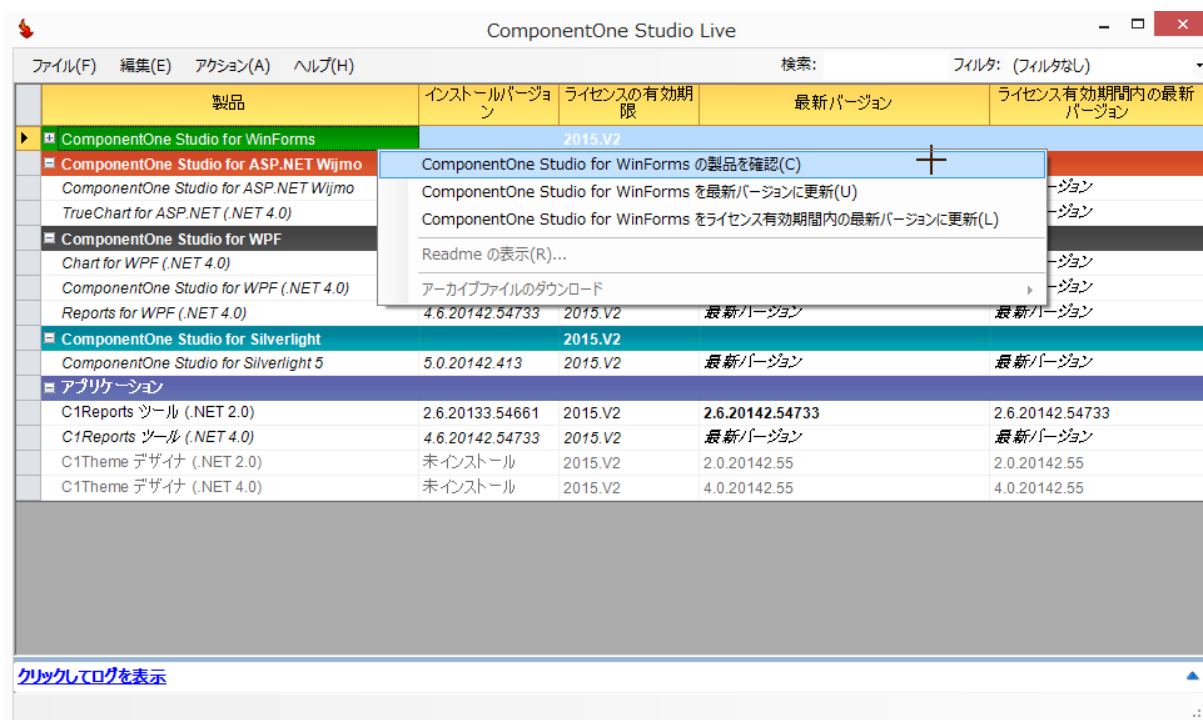
5. 対象製品の全アセンブリが最新バージョンに更新されます

※開発中のプロジェクトに適用する場合は、プロジェクト移行ツールを使用して移行後、プロジェクトのリビルドを行う必要があります。

C1Live の使い方 : その他

C1Lve のメニュー[アクション]から表示される各コマンドの内容をご説明します。C1Lve に関する詳細については、[オンラインヘルプ](#)をご覧ください。

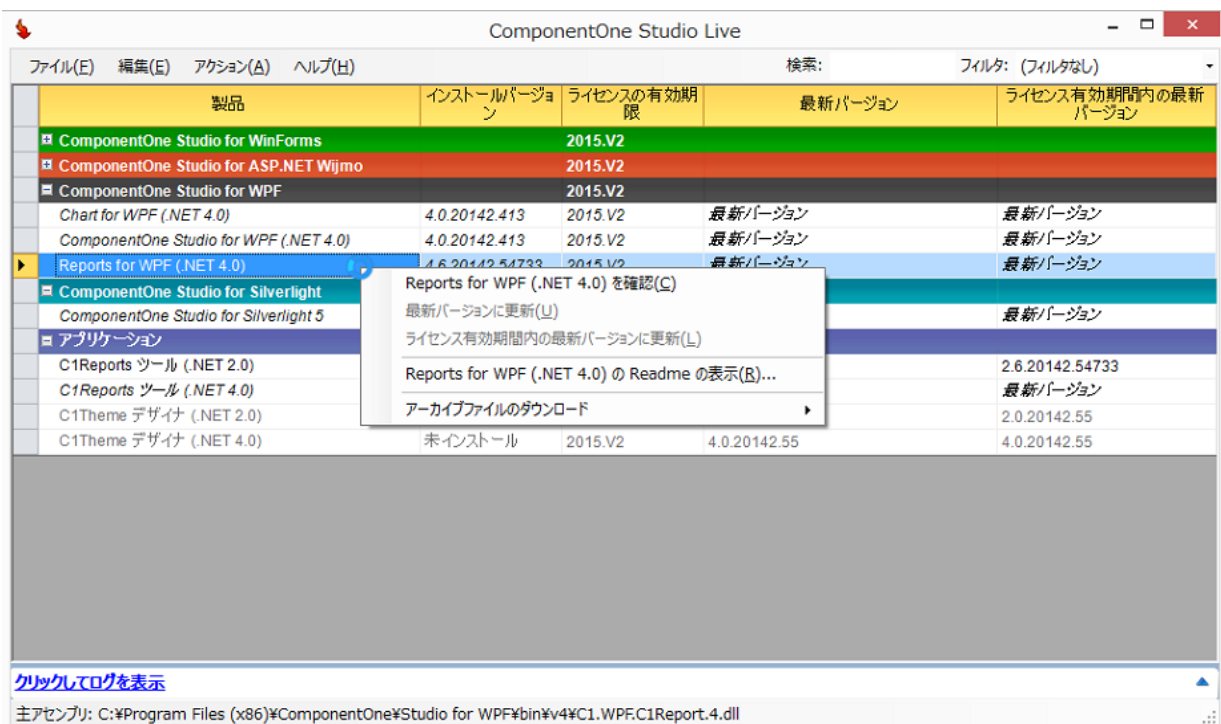
- 一覧で C1Studio を選択している場合



ComponentOne Studio for xxx の製品を確認	選択している製品の一覧を更新します。
ComponentOne Studio for xxx を最新バージョンに更新	<p>選択している製品の全アセンブリを、最新バージョンにアップデートします。</p> <p>※ライセンス有効期限外のバージョンに更新した場合、そのコントロール</p>

	はトライアル版として動作します。
ComponentOne Studio for xxx をライセンス有効期間内の最新バージョンに更新	選択している製品の全アセンブリを、ライセンス有効期間内の最新バージョンにアップデートします。

- 一覧でコンポーネントを選択している場合




xxx を確認	選択しているコンポーネントの一覧を更新します。
xxx の Readme を表示	選択しているコンポーネントの Readme (更新履歴) を表示します。
アーカイブファイルのダウンロード	選択しているコンポーネントのアセンブリをダウンロードします。

ComponentOne Studio プロジェクト移行ツール

「ComponentOne Studio プロジェクト移行ツール」は、ComponentOne Studio に収録されているコンポーネントのアセンブリを使用したプロジェクトに対し、そのアセンブリを最新版へアップグレードするために必要な削除・更新処理を支援するツールです。

アセンブリバージョンに依存せずにアセンブリの参照を行うようになるため、アプリケーションのリビルドを実行するだけで、常に最新のアセンブリが適用されたアプリケーションを作成することが可能になります。

 **メモ**：「ComponentOne Studio プロジェクト移行ツール」は、製品をインストールしたフォルダ内にあり、スタートメニューの[ComponentOne]-[ツール]-[プロジェクト移行ツール]からも起動できます。

使用方法

本ツールは、ウィザード形式のアプリケーションです。ツール本体を起動したら、画面の指示に従って操作してください。

処理対象となるプロジェクトは、単独のプロジェクトを指定できるほか、フォルダを指定することで、そのフォルダおよびすべてのサブフォルダに含まれるプロジェクトを一括で置換することができます。またオプションにて、不要ファイルの削除、明示的な参照設定（HintPath）の保持や、ファイルのバックアップの保存について指定できます。

制限事項と注意点：

- 本ツールによるプロジェクトの移行は完全ではなく、手動による変更を必要とする場合があります。
- 本ツールはプロジェクトファイルを基準に処理を行います。そのためプロジェクトファイルがない場合は、処理することができません。
- 不完全な動作によるデータの損失に備え、操作対象のファイルをすべて本ツール以外の方法であらかじめバックアップしてください。
- 移行を開始する前にプロジェクトを終了してください。
- 本ツールでは、Visual Studio .NET 2002/2003 で作成されたプロジェクト、および Expression Blend にて作成されたプロジェクトの移行はサポートしていません。
- 以下のコントロールは、複数エディションで同じ名前のアセンブリを提供しています。そのため、同一環境に複数エディションがインストールされている場合、移行ツール実行後に Visual Studio の参照設定が不定になることがあります。この場合は参照設定を手動で修正してください。
 - C1DataSource
 - C1Excel
 - C1Report

- C1Pdf
- C1Zip
- オプション設定の「明示的な参照設定 (HintPath) を残す」のチェックは、既定では Off になっており、移行処理後は製品インストール先の Bin フォルダ内のアセンブリが参照されます。複数のエディションがインストールされている環境等、Visual Studio からコントロールを明示的に参照設定している場合はこのオプションを On に設定してください。

移行手順

1. [Visual Studio 変換ウィザード] により、既存の Visual Studio プロジェクトを、最新の Visual Studio プロジェクトに変換します。※必要な場合
2. 移行ツールを実行し、対象のプロジェクトをアップデートします。
3. プロジェクトを開きます。
4. [ビルド(B)]メニューにある[ソリューション (もしくはプロジェクト) のリビルド(R)]を実行します。

⚠注意：「リビルド」ではなく「ビルド」を実行した場合、正しく移行されないことがありますので、必ず「リビルド」を実行してください。

チャートデータ移行ツール

チャートデータ移行ツールは、ActiveX 製品の True Win/WebChart 8.0J で作成したチャートファイル (*.oc2 / *.oc3) を .NET 製品の TrueChart for WinForms/ASP.NET で読み込み可能なチャートファイル (*.chart2dxml / *.chart3dxml) へ移行するツールです。

⚠注意：チャートデータ移行ツールを使用するには、True WinChart 8.0J または True WebChart 8.0J がインストールされている必要があります。

📌メモ：「チャートデータ移行ツール」は、製品をインストールしたフォルダ内にあり、スタートメニューの [ComponentOne]-[ツール]-[チャートデータ移行ツール]からも起動できます。

チャートファイルを移行するには、以下の手順を実行します。

1. スタートメニューから「チャートデータ移行ツール」を実行します。
2. <次へ>をクリックし、移行を開始します。
3. 移行元の ActiveX チャートファイル (*.oc2 / *.oc3) を選択し、<次へ>をクリックします。
4. .NET チャートファイル (*.chart2dxml / *.chart3dxml) の出力先を指定し、<次へ>をクリックします。「移行の開始」画面で<次へ>をクリックし、移行処理を実行します。

5. 移行が完了すると、ActiveX、.NET それぞれのチャートが表示され、.NET チャートファイル (*.chart2dxml / *.chart3dxml) が作成されます。表示されたチャートを右クリックし、チャートデータを参照・変更することも可能です。
6. <終了> をクリックし、移行作業を終了します。

C1Report 移行ツール

Reports for WinForms では、**Preview for .NET** と **True WinReports for .NET** が1つのコンポーネントに統合されました。**Reports for WinForms** には、**C1Report** コンポーネントと **C1Preview** コンポーネントによって提供されたすべての機能が含まれていますが、アセンブリ名と名前空間が以前と異なるため、**Preview for .NET** および **True WinReports for .NET** を使用して作成されたプロジェクトは、新しいアセンブリと名前空間を参照するように変更する必要があります。

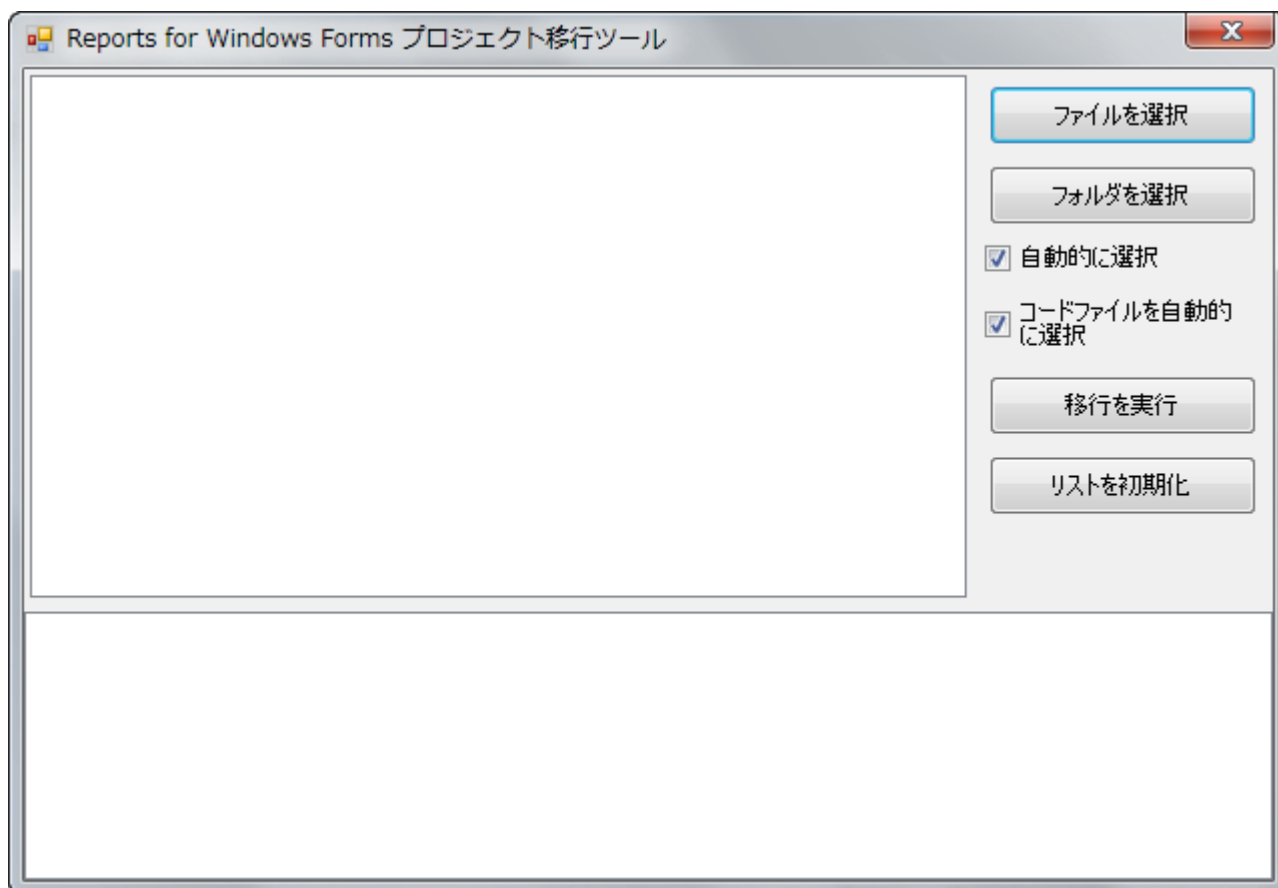
Reports for WinForms の移行ツール「Convert2Report.exe」を使用すると、以下の処理が実行されます。

- C# または VB プロジェクトでは、**C1.C1Preview.2** と **C1.Win.C1Preview.2** から **C1.C1Report.2** と **C1.Win.C1Report.2** に参照が変更されます。
- licenses.licx ファイルでは、**C1Report**、**C1PrintDocument**、プレビューコントロールの情報が新しいアセンブリの情報に更新されます。
- C# または VB ソースコードファイルでは、**C1.Win.C1Report** 名前空間のすべての記載が **C1.C1Report** に置き換えられます。

移行ツールでは、個別のファイルやフォルダ配下のファイルすべてを変換することができます。

プロジェクトを変換するには、以下の手順を実行します。

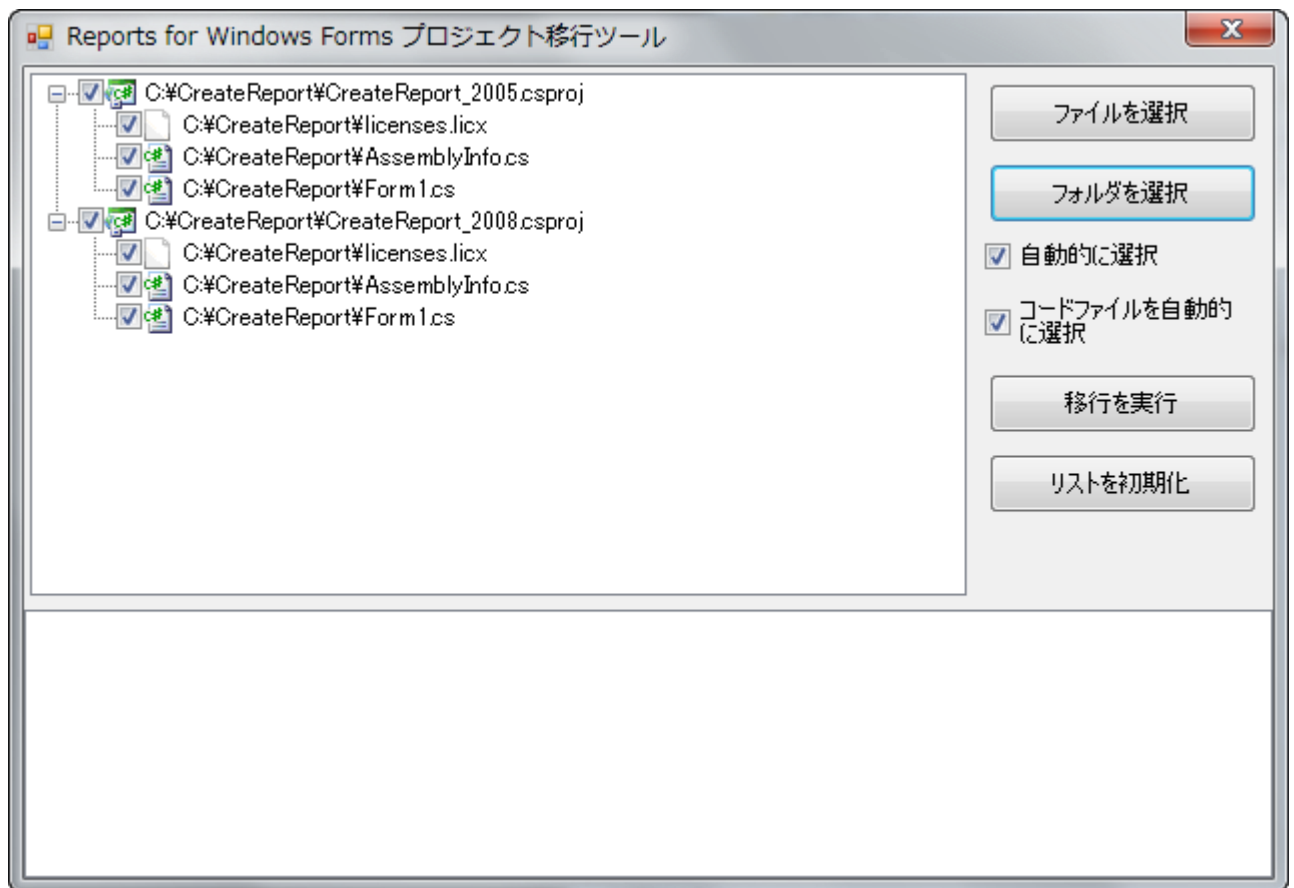
1. スタートメニューから「**C1Report 移行ツール**」を実行します。
アプリケーションは次のように表示されます。



2. [licenses.licx の自動選択] と [コードファイルの自動選択] チェックボックスを ON のままにして、プロジェクトファイルをアップグレードするために選択したときに含まれるすべてのファイルを自動選択します。
3. 〈変換するプロジェクト/ファイルの選択〉または〈変換するフォルダの選択〉ボタンのいずれかをクリックします。[開く] または [フォルダの参照] ダイアログボックスが開きます。
4. [開く] または [フォルダの参照] ダイアログボックスで、変換するファイルまたはフォルダのいずれかを見つけて、〈OK〉を選択します。

追加のフォルダまたはファイルを選択する必要がある場合、〈変換するプロジェクト/ファイルの選択〉または〈変換するフォルダの選択〉ボタンを再度クリックし、追加のファイルを選択できます。選択をクリアし、再度ファイルを選択する必要がある場合は、〈消去〉ボタンを押します。

ダイアログボックスは次のように表示されます。



5. 左ペインにある変換対象ファイルを確認します。ファイルのチェックを ON または OFF にして、変換対象ファイルを変更できます。
6. 〈**変換**〉をクリックすると、**Preview for .NET** および **True WinReports for .NET** の各種情報が **Reports for WinForms** に変換されます。
 正常に変換されたファイルは、左ペインに緑色のチェックマークを付けて表示されます。
 変換されたコードファイルとライセンスファイルが、元のファイルと変換されたファイルの場所と共に下部にあるペインに表示されます。デフォルトでは、元のファイルは **FileName.original** と同じディレクトリに保存され、変換されたファイルが元のファイル名で保存されます（元のファイルが上書きされます）。
7. すべてのファイルが正常に変更された後で、移行ツールを終了します。

関連製品との共存

- 旧製品との共存

開発環境では、2012J 以前の全てのバージョンおよびエディションの ComponentOne Studio および単体製品と共存させることができません。旧製品をアンインストールしてから製品をインストールしてください。なお、運用環境では、対象バージョンのアセンブリを正しく配置することで共存させることが可能です。

⚠ 注意：2013J 以降の異なるエディションの ComponentOne Studio は、開発環境・運用環境共に共存が可能です。

- 英語版との共存

英語版の ComponentOne Studio と共存させることができません。英語版の ComponentOne Studio がインストールされた開発環境に製品をインストールした場合またはその逆を行った場合は、上書きインストールされるため、製品が正常に動作しなくなる可能性があります。

互換性について

コンポーネントごとに互換性の有無が異なります。

本製品に収録されているコンポーネントの旧バージョンとの互換性は以下のとおりです。

⚠ 注意：互換性のあるコンポーネントの場合でも、機能改善や不具合の修正上の都合により、一部のメンバが新たな機能に変更されている場合があります。この場合、ソースコードの修正が必要になりますので、あらかじめご了承ください。変更されたメンバの詳細については、コンポーネントごとの更新履歴（「[使用上のご注意](#)」の「更新履歴について」の項参照）を確認してください。

Windows フォーム用コンポーネント

すべてのコンポーネントで旧バージョンとの上位互換が保たれています。そのため、旧バージョンを使用して作成したプロジェクトを簡単に移行できます。移行についての詳細は、Web サイトの「[移行情報](#)」をご覧ください。

ASP.NET 用コンポーネント

- 以下のコントロールの場合、旧バージョンとの上位互換が保たれています。そのため、旧バージョンを使用して作成したプロジェクトを簡単に移行できます。
 - TrueChart for ASP.NET
 - Excel for .NET
 - PDF for .NET
 - Zip for .NET

上記以外のコンポーネント：

ComponentOne Studio Enterprise 2012J SP1 以降、および ComponentOne Studio for ASP.NET Wijmo 2012J 以降に収録された ASP.NET コンポーネントとの上位互換が保たれています。

これらより前に収録されていた ASP.NET コンポーネントとの互換性はありません。

Silverlight 用コンポーネント

すべてのコンポーネントで旧バージョンとの上位互換が保たれています。そのため、旧バージョンを使用して作成したプロジェクトを簡単に移行できます。

WPF 用コンポーネント


すべてのコンポーネントで旧バージョンとの上位互換が保たれています。そのため、旧バージョンを使用して作成したプロジェクトを簡単に移行できます。なお、WinForms から WPF への移行については、Web サイトの「[移行情報](#)」をご覧ください。

WinRT XAML 用コンポーネント

すべてのコンポーネントで旧バージョンとの上位互換が保たれています。そのため、旧バージョンを使用して作成したプロジェクトを簡単に移行できます。

サンプルについて

ComponentOne Studio には、コントロールの各種機能の理解に役立つサンプルプログラムが多数含まれています。サンプルプログラムは、Sample Explorer を使用して簡単に検索・起動することができます。

 **メモ：** サンプルプログラムは VB、C# それぞれの言語用を収録しており、利用する開発言語に合わせて参照していただけます。ただし一部のサンプルでは、VB のみ、または C# のみを収録しています。

Sample Explorer の起動方法

Sample Explorer を起動するには、スタートメニューの[ComponentOne]–[ComponentOne Studio for XXX]–[サンプル] (XXX はエディション名) から、目的のコンポーネントのアイコンをダブルクリックします。

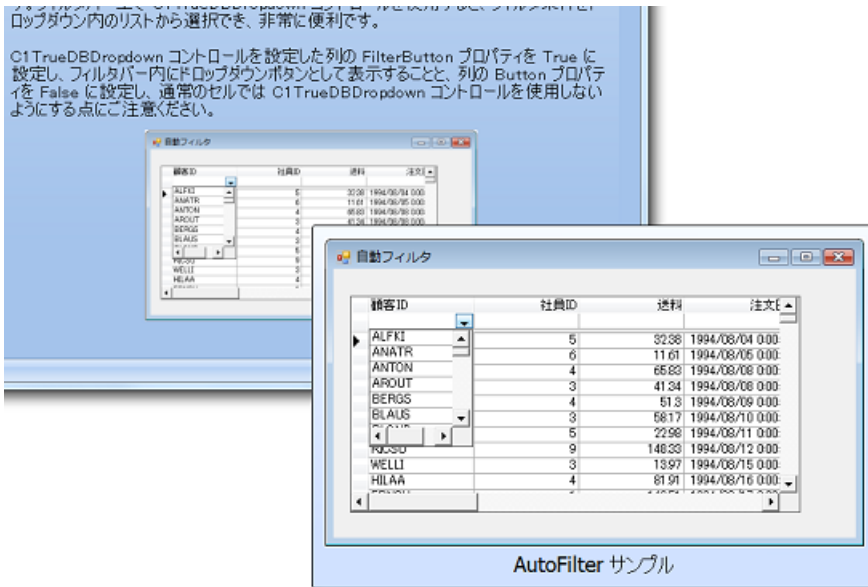
Sample Explorer の使い方

起動した Sample Explorer を使ってサンプルプロジェクトを実行するには、次の手順を実行します。

1. Sample Explorer を起動すると、左ペインにサンプル一覧、右ペインにサンプルの解説が表示されます。



- 必要に応じて左ペインをスクロールし、目的のサンプルを選択します。右上の検索窓に任意のテキストを入力し、左ペインにリストアップされたサンプル一覧から目的のサンプルを選択することもできます。
- 選択したサンプルの解説が右ペインに表示されます。解説に画像が含まれる場合、画像をクリックすると拡大表示します。



- 「C# サンプルの起動」または「VB サンプルの起動」ボタンをクリックすると、目的のサンプルプロジェクトが起動します。

AutoFilter

C# サンプルの起動

VB サンプルの起動

C1TrueDBDropdown コントロールをフィルタバーに設定する方法を紹介します。

【操作方法】

フィルタバーの“顧客ID”列内に表示されるボタンをクリックすると、ドロップダウンが表示されます。ドロップダウンのリストを選択して、リストの値でフィルタリングします。

⚠注意： SampleExplorer を使用せずに、サンプルを直接起動することも可能です。Visual Studio 2013 以降をご使用の場合は、WinForms と ASP.NET は Visual Studio 2012 用のプロジェクトを、WPF と Silverlight は Visual Studio 2010 用のプロジェクトを開いてください。

