.NET-Funktionen für die ClickOnce-Technologie

Think twice, ClickOnce

Hinter ClickOnce verbirgt sich mehr als „einmal klicken“. dotnetpro erklärt, wie sich mithilfe der Klassen aus dem Namensraum System.Deployment typische Szenarien aus dem ClickOnce-Alltag elegant lösen lassen.

ClickOnce bietet eine solide Grundausstattung für typische Installationsaufgaben. Aber auch wer noch mehr Kontrolle über den Installationsvorgang benötigt, ist nicht verloren. Der Updatevorgang lässt sich mit dem ClickOnce-API, das im Namensraum *System.Deployment* beherbergt ist, feingranular steuern. Manches lässt sich aber nicht verändern. Das Aussehen etwa der Dialoge oder auch die einzelnen Schritte der Installation, was dem Kopieren der Dateien in den lokalen ClickOnce-Cache entspricht, lassen sich nicht anpassen. Hier gibt es aus Sicherheitsgründen eine konsistente Oberfläche, und ganz bewusst sind keine Systemmanipulationen neben dem Kopiervorgang, Startmenüeintrag und dem Eintrag unter *Systemsteuerung/Software* möglich.

System.Deployment

Kernstück der Programmierschnittstelle ist die Klasse *ApplicationDeployment*. Über sie lässt sich feststellen, ob ein Update zur Verfügung steht. Ein Update kann heruntergeladen und eingespielt werden und Applikationsteile lassen sich bedarfsweise nachladen.

Wie in Tabelle 1 dargestellt, werden jeweils synchrone und asynchrone Varianten (deren Namen enden auf *Async)* der Methoden angeboten. Somit ist auch die Realisierung eines eigenen Fortschrittsanzeigefensters möglich.

Wenn Sie die Logik verändern möchten, nach der auf Updates geprüft wird und diese heruntergeladen werden, dann müssen Sie erst in den Projekteinstellungen unter *Veröffentlichen/Update* die ClickOnce-eigene Prüfung abschalten. Es ist auch möglich, beides zu kombinieren. ClickOnce kann beispielsweise wie voreingestellt bei jedem Programmstart nach Updates suchen, und der Benutzer kann zusätzlich über einen Button die Prüfung auf neuere Versionen starten.

Um das zu bewerkstelligen, müssen zunächst drei allgemeine Schritte ausgeführt werden, die immer beim Umgang mit dem ClickOnce-API benötigt werden.

1. Referenz auf *System.Deployment.dll* einfügen, falls dies nicht schon durch den Visual-Studio-Veröffentlichungsassistenten geschehen ist.

2. Den Namensraum *System.Deployment.Application* einbinden:

```
using System.Deployment.Application;
```

3. Eine Referenz auf das aktuelle *ApplicationDeployment*-Objekt anfordern, vergleiche dazu auch Tabelle 2:

```
ApplicationDeployment ad =
ApplicationDeployment.CurrentDeployment;
```

Nun können wir hinter einem Button auf Wunsch des Benutzers eine Updateprüfung durchführen:

```
if (ad.CheckForUpdate() == true)
  {
    // Es gibt ein neues Update!
  }
```

Die Methode *CheckForUpdate* liefert ein einfaches *Ja* oder *Nein* zurück. Detailliertere Informationen liefert *CheckForDetailedUpdate*. Damit erfahren wir beispielsweise auch, welche neue Version

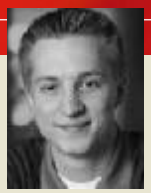
Tabelle 1

Angelpunkt des ClickOnce-APIs: Die Methoden der Klasse ApplicationDeployment.

Methode	Beschreibung
CheckForUpdate	Prüft, ob eine neue Version zur Verfügung steht und liefert True oder False als Ergebnis.
CheckForUpdateAsync	Wie CheckForUpdate, aber asynchron. Für Ergebnis und Fortschritt werden Ereignisse ausgelöst.
CheckForUpdateAsyncCancel	Bricht den asynchronen Prüfungsvorgang ab.
CheckForDetailedUpdate	Wie CheckForUpdate, liefert jedoch ausführliche Informationen über das neue Update (Versionsnummer, Größe etc.)
Update	Lädt das Update herunter und spielt es ein.
UpdateAsync	Wie Update, aber asynchron.
UpdateAsyncCancel	Bricht den asynchronen Updatevorgang ab.
DownloadFileGroup	Lädt eine Dateigruppe herunter. Diese werden in den Projekteigenschaften unter "Application Files" aus Dateien, die im VS-Projekt eingebunden sind, definiert.
DownloadFileGroupAsync	Wie DownloadFileGroup, aber asynchron.
DownloadFileGroupAsyncCancel	Bricht den asynchronen Ladevorgang ab.
IsFileGroupDownloaded	Überprüft, ob eine Dateigruppe bereits heruntergeladen wurde.

Auf einen Blick

Autor



Neno Loje ist Strategischer Berater und Associate bei Thinktecture. Er entwickelt Software, berät Firmen und schult Entwickler rund um die .NET-Technologie mit Schwerpunkt auf Visual Studio Team System. Sie erreichen ihn unter www.dotnet-online.de.

dotnetpro.code
A0708ClickOnceHighTech

Sprachen C#

Technik .NET Framework 2.0, ClickOnce

Voraussetzungen .NET Framework 2.0 SDK