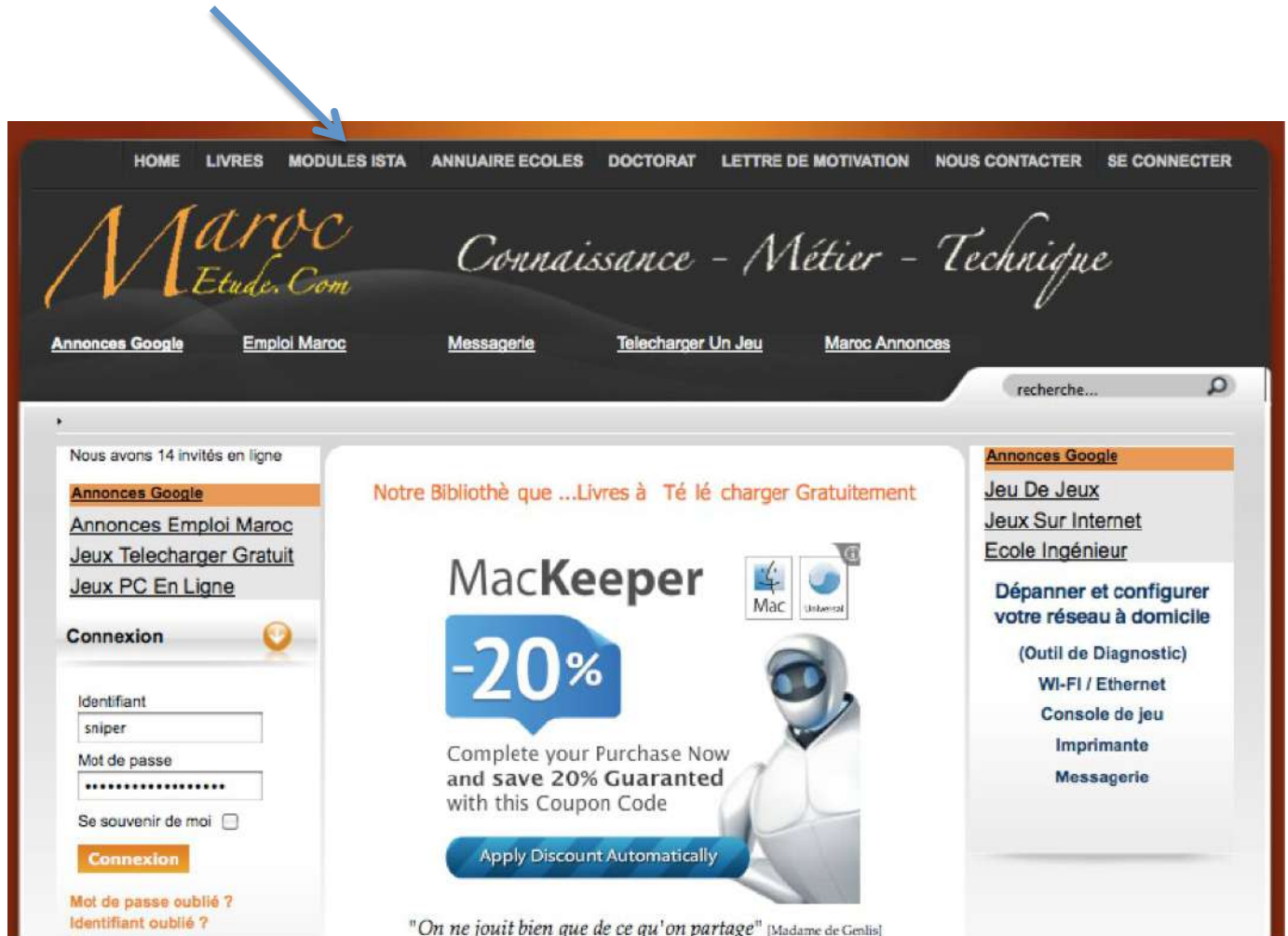


PORTAIL DE LA FORMATION PROFESSIONNELLE AU MAROC

Télécharger tous les modules de toutes les filières de l'OFPPT sur le site dédié à la formation professionnelle au Maroc : www.marocetude.com

Pour cela visiter notre site www.marocetude.com et choisissez la rubrique :

MODULES ISTA



The image shows a screenshot of the website Maroc Etude.Com. At the top, there is a navigation menu with the following items: HOME, LIVRES, **MODULES ISTA**, ANNUAIRE ECOLES, DOCTORAT, LETTRE DE MOTIVATION, NOUS CONTACTER, and SE CONNECTER. Below the menu is the website's logo, "Maroc Etude.Com", and the tagline "Connaissance - Métier - Technique". There are also several utility links: Annonces Google, Emploi Maroc, Messagerie, Telecharger Un Jeu, and Maroc Annonces. A search bar is located in the top right corner. The main content area features a central advertisement for MacKeeper with a -20% discount. On the left side, there is a login section with fields for "Identifiant" (containing "sniper") and "Mot de passe", and a "Connexion" button. On the right side, there is a sidebar with a search bar and a list of links under the heading "Annonces Google": Jeu De Jeux, Jeux Sur Internet, Ecole Ingénieur, Dépanner et configurer votre réseau à domicile (Outil de Diagnostic), WI-FI / Ethernet, Console de jeu, Imprimante, and Messagerie. At the bottom of the page, there is a quote: "On ne jouit bien que de ce qu'on partage" [Madame de Genlis].

Déploiement d'applications et de composants

Le déploiement est le processus selon lequel une application ou un composant fini est distribué en vue de son installation sur d'autres ordinateurs. Dans Visual Studio .NET, le déploiement est basé sur la technologie Microsoft Windows Installer.

Certains outils permettent de créer des programmes d'installation pour des applications et des composants Windows ou Web, ainsi que des fichiers .cab pour les téléchargements.

Les outils de déploiement de Visual Studio .NET permettent de créer des fichiers Windows Installer (.msi) qui peuvent être distribués et installés sur d'autres ordinateurs. Les fichiers du programme d'installation peuvent être distribués sur des supports traditionnels, comme les disquettes ou les CD-ROM, ou peuvent être placés sur un lecteur réseau pour une installation sur un réseau.

Pour déployer une application, vous créez d'abord un projet de configuration et définissez les propriétés du projet de déploiement qui déterminent quand et où le programme d'installation sera généré. Pour un déploiement via des supports traditionnels, vous copiez ensuite le fichier .msi depuis l'emplacement de génération vers une disquette ou autre support. Pour déployer vers un emplacement réseau, vous créez un projet de configuration Web et ajoutez le groupe de sorties du projet correspondant à l'application au projet de configuration Web dans l'Éditeur du système de fichiers. Après avoir généré le programme d'installation, vous le copiez sur le serveur, où il peut alors être téléchargé sur le réseau.

Introduction à Microsoft Windows Installer

Microsoft Windows Installer est un service d'installation et de configuration fourni avec Windows 2000, Windows Me et Windows XP. Il existe également pour Windows 95, Windows 98 et Windows NT 4.0.

Windows Installer s'appuie sur un modèle déterminé par les données et fournissant, dans un seul package toutes les instructions et données d'installation. Par opposition, les programmes traditionnels d'installation écrits en scripts s'appuyaient sur un modèle procédural qui fournissait des scripts d'instructions aux installations d'applications. Ces programmes mettaient l'accent sur la procédure d'installation, alors que Windows Installer s'intéresse surtout au contenu de l'installation.

Avec Windows Installer, chaque ordinateur conserve une base de données d'informations sur chaque application qu'il installe, y compris les fichiers, les clés de Registre et les composants. Lors de la désinstallation d'une application, la base de données est contrôlée afin de s'assurer qu'aucune autre application ne nécessite un fichier, une clé de Registre ou un composant qui s'apprête à être supprimé. Cette vérification empêche que le retrait d'une application n'entraîne l'interruption d'une autre application.

Windows Installer prend aussi en charge la réparation automatique, c'est-à-dire la capacité pour une application de réinstaller automatiquement les fichiers manquants qui ont pu être supprimés par mégarde par l'utilisateur.

De plus, Windows Installer permet d'annuler une installation. Par exemple, si une application s'appuie sur une base de données spécifique et que la base de données est introuvable lors de l'installation, celle-ci peut être interrompue et l'ordinateur retrouve l'état qui était le sien avant l'installation.

Les outils de déploiement de Visual Studio .NET sont basés sur Windows Installer ; ils mettent à votre disposition de puissantes capacités de déploiement rapide et de maintenance des applications créées avec Visual Studio .NET.

Projets de déploiement

Visual Studio fournit des modèles pour quatre types de projets de déploiement : Installation, Installation Web, Module de fusion et Cab. De plus, un Assistant Configuration montre les différentes étapes de la création d'un projet de déploiement. Ces modèles et cet Assistant sont proposés dans la boîte de dialogue Nouveau projet, sous le nœud **Projets de configuration et de déploiement**.

Vous trouverez ci-dessous quelques conseils pour choisir le type de projet de déploiement adapté à votre cas.

Type de projet	Description
Installation	Crée un programme d'installation pour une application Windows.
Installation Web	Crée un programme d'installation pour une application Web.
Module de fusion	Composants en package susceptibles d'être partagés par plusieurs applications.
Cab	Crée un fichier cab pour un téléchargement sur un navigateur Web.

Attention Une fois le projet créé, il est impossible de le faire basculer entre les types Web et standard. Si vous avez créé un projet de déploiement de type standard, puis souhaitez le déployer sur le Web, vous ne pouvez pas l'utiliser tel quel. Il faut en créer un autre.

Projets de configuration

Les projets de configuration permettent de créer des programmes d'installation destinés à distribuer une application. Le fichier Windows Installer (.msi) obtenu contient l'application, les fichiers dépendants, les informations sur l'application (entrées du Registre, par exemple) et les instructions d'installation. Quand le fichier .msi est distribué et exécuté sur un autre ordinateur, vous pouvez être assuré que tout élément nécessaire à l'installation est inclus ; si, pour une raison quelconque, l'installation échoue (l'ordinateur cible ne dispose pas de la version requise du système d'exploitation, par exemple), celle-ci est annulée et l'ordinateur retrouve l'état qui était le sien avant l'installation.

Il existe deux types de projets de configuration dans Visual Studio : les projets de configuration et les projets de configuration Web. La différence entre ces deux types de projets concerne l'emplacement où le programme d'installation sera déployé : les projets de configuration installent les fichiers dans le système de fichiers de l'ordinateur cible et les projets de configuration Web dans un répertoire virtuel du serveur Web.

De plus, un Assistant Configuration est disponible pour simplifier la création d'un projet de configuration ou d'un projet de configuration Web.

Projets de modules de fusion

Les projets de modules de fusion permettent de créer des composants d'installation réutilisables. De manière très similaire aux bibliothèques de liens dynamiques, qui permettent le partage du code entre applications, les modules de fusion permettent le partage du code d'installation entre plusieurs modules Windows Installer.

Un module de fusion (fichier .msm) est un seul et même package contenant tous les fichiers, toutes les ressources, toutes les entrées de registre et toute la logique d'installation nécessaires à l'installation d'un composant.

Les modules de fusion permettent de capturer toutes les dépendances d'un composant particulier, pour garantir que les versions correctes sont installées.

Pour éviter les problèmes liés au versioning, placez toujours des fichiers et des composants dans des modules de fusion s'ils doivent être partagés par plusieurs applications.

Les modules de fusion constituent une fonctionnalité de Windows Installer offrant une méthode standard de livraison des composants et permettant de s'assurer que c'est la bonne version du composant qui a été installée.

Les modules de fusion ne peuvent pas être installés directement, mais sont fusionnés dans un programme d'installation pour chaque application utilisant le composant. Cela garantit une installation cohérente du composant pour toutes les applications et supprime les problèmes tels que conflits de versions, entrées du Registre manquantes et fichiers incorrectement installés.

Chaque module de fusion contient des informations de version uniques et utilisées par la base de données de Windows Installer pour déterminer les applications qui utilisent le composant et empêcher la suppression prématurée de celui-ci. Pour cette raison, un nouveau module de fusion doit être créé pour chaque version incrémentielle du composant ; le module ne doit jamais être mis à jour après avoir été inclus dans un programme d'installation.

Les projets de module de fusion servent à packager des fichiers ou des composants à partager entre plusieurs applications. Ils créent un fichier de module de fusion (.msm) qui inclut la totalité des fichiers, ressources et entrées du Registre, ainsi que la logique de configuration, de votre composant. Le fichier .msm résultant peut ensuite être fusionné dans d'autres projets de déploiement, ce qui assure la cohérence de l'installation du composant dans plusieurs applications.

Comparaison entre programme d'installation et module de fusion

Dans Visual Studio, il est possible de créer deux types de programmes d'installation : programmes d'installation Windows (.msi) et modules de fusion (.msm). Les programmes d'installation Windows servent à installer des applications et des fichiers sur un autre ordinateur ; les modules de fusion servent à installer des composants que plusieurs applications se partagent.

La principale distinction entre les deux types est le public cible. Tout ce que l'utilisateur final de l'application utilisera directement (fichiers exécutables, documents, pages HTML) doit être mis dans un package dans un programme d'installation Windows. Tout ce qu'un développeur sera seul à utiliser (DLL, contrôles, ressources) doit

se trouver dans un module de fusion, que le développeur peut placer dans un package dans un programme d'installation Windows, pour distribution à l'utilisateur final.

En règle générale, tout composant que plusieurs applications partagent ou sont susceptibles de partager doit se trouver dans son propre module de fusion. Il est possible de mettre plusieurs composants dans un seul et même module de fusion, mais il vaut mieux en créer un par composant, pour éviter de distribuer des fichiers inutiles.

Un module de fusion pourrait être comparé à une photo instantanée de la version particulière d'un composant. Il vaut mieux créer un module de fusion par version du composant, pour éviter les conflits dus aux différences de version.

Comme les programmes d'installation peuvent inclure plusieurs applications, il est possible d'installer toute une suite d'applications en une seule et même étape. Dans ce cas, le programme d'installation doit inclure des modules de fusion pour tous les composants utilisés par une quelconque des applications incluses ; si un module de fusion est utilisé par plusieurs applications, il suffit de l'ajouter une fois.

Projets de fichier CAB

Les projets CAB permettent de créer un fichier .cab regroupant les contrôles ActiveX à télécharger depuis un serveur Web vers un navigateur Web.

À l'inverse des autres types de projets de déploiement, aucun éditeur n'est fourni pour travailler avec les projets CAB. Les fichiers et sorties de projet peuvent être ajoutés à un projet CAB dans l'Explorateur de solutions, et les propriétés peuvent être définies dans la fenêtre Propriétés ou dans les pages de propriétés Projet.

Les propriétés des projets CAB permettent de spécifier un niveau de compression, d'implémenter la signature Authenticode, de définir le nom complet et les informations de version et de spécifier l'emplacement des fichiers dépendants sur le Web.

Remarque Les dépendances ne sont pas calculées sur les projets CAB ; vous devez déterminer toutes les dépendances et les référencer.

Création ou ajout de projets de déploiement

Les projets de déploiement permettent de spécifier où et comment votre solution sera déployée à la fois pendant et après le développement. Une solution peut contenir plusieurs projets de déploiement qui peuvent varier selon la configuration.

Pour créer un projet de déploiement

1. Dans le menu **Fichier**, pointez sur **Ajouter un projet**, puis cliquez sur **Nouveau projet**.
2. Dans la boîte de dialogue **Ajouter un nouveau projet**, sélectionnez le dossier **Projets de configuration et de déploiement**.
3. Sélectionnez le type de projet de déploiement à créer. Pour plus d'informations, consultez [Projets de déploiement](#).

Pour ajouter un projet de déploiement existant à une solution

1. Dans le menu **Fichier**, pointez sur **Ajouter un projet**, puis cliquez sur **Projet existant**.
2. Dans la boîte de dialogue **Ajouter un projet existant**, recherchez l'emplacement du projet de déploiement, puis cliquez sur **Ouvrir**.

Conseil Pour ne voir que des projets de déploiement, dans la zone **Type de fichiers**, sélectionnez Projets de configuration et de déploiement.

Procédure pas à pas : déploiement d'une application Windows

La procédure pas à pas suivante décrit le processus de création du programme d'installation d'une application Windows qui lance le Bloc-notes. Dans cette procédure pas à pas, vous allez créer l'application Windows, puis créer un programme d'installation qui définit des raccourcis et des associations de fichiers, ajoute une entrée au Registre, affiche des boîtes de dialogue personnalisées et vérifie la version de Microsoft Internet Explorer pendant l'installation.

Remarque Vous n'êtes pas tenu de suivre l'intégralité de ces étapes lorsque vous créez vos propres programmes d'installation. Cette procédure pas à pas est destinée à vous présenter certaines fonctionnalités facultatives de déploiement que vous pourriez trouver utiles. Pour créer un programme d'installation de base, effectuez les quatre premières étapes de la procédure qui suit.

Pour créer une application Windows

1. Dans le menu **Fichier**, pointez sur **Nouveau**, puis choisissez **Projet**.
2. Dans la boîte de dialogue **Nouveau projet**, sélectionnez **Projets Visual Basic** dans le volet **Types de projets**, puis choisissez **Application Windows** dans le volet **Modèles**. Dans la zone **Nom**, tapez **Mon Bloc-notes**.
3. Cliquez sur **OK** pour fermer la boîte de dialogue.
Le projet est ajouté à l'Explorateur de solutions et le Concepteur Windows Forms s'ouvre.
4. Sélectionnez l'onglet **Windows Forms** de la boîte à outils, puis faites glisser un contrôle **Button** jusqu'au formulaire.
5. Double-cliquez sur le contrôle **Button** pour ajouter un gestionnaire d'événements pour le bouton. Dans le gestionnaire d'événements, ajoutez le code suivant :

```
Shell("Notepad.exe", AppWinStyle.NormalFocus)
```


Vous lancez ainsi Notepad.exe et lui donnez le focus.
6. Dans le menu **Générer**, choisissez **Générer Mon Bloc-notes** pour générer l'application.

Pour créer un projet de déploiement

1. Dans le menu **Fichier**, pointez sur **Ajouter un projet**, puis choisissez **Nouveau projet**.
2. Dans la boîte de dialogue **Ajouter un nouveau projet**, sélectionnez **Projets de configuration et de déploiement** dans le volet **Types de projets**, puis choisissez **Projet de configuration** dans le volet **Modèles**. Dans la zone **Nom**, tapez **Programme d'installation de Mon Bloc-notes**.
3. Cliquez sur **OK** pour fermer la boîte de dialogue.
Le projet est ajouté à l'Explorateur de solutions et l'Éditeur du système de fichiers s'ouvre.
4. Sélectionnez le projet **Programme d'installation de Mon Bloc-notes** dans l'Explorateur de solutions. Dans la fenêtre Propriétés, sélectionnez la propriété **ProductName** et tapez **Mon Bloc-notes**.
Remarque La propriété **ProductName** détermine le nom qui s'affichera pour l'application dans les noms de dossier et dans la boîte de dialogue **Ajout/Suppression de programmes**.

Pour ajouter l'application Windows au programme d'installation

1. Sélectionnez le projet **Programme d'installation de Mon Bloc-notes** dans l'Explorateur de solutions. Dans l'Éditeur du système de fichiers, sélectionnez le nœud **Dossier d'application**.
2. Dans le menu **Action**, choisissez **Ajouter, Sortie du projet**.
3. Dans la boîte de dialogue **Ajouter le groupe de sortie du projet**, choisissez **Mon Bloc-notes** dans la liste déroulante **Projet**. Cliquez sur **OK** pour fermer la boîte de dialogue.
4. Sélectionnez le groupe **Sortie principale** dans la liste, puis cliquez sur **OK**.
5. Dans le menu **Générer**, choisissez **Générer Programme d'installation de Mon Bloc-notes**.

Pour déployer l'application (programme d'installation de base)

Vous pouvez ignorer cette étape si vous effectuez la procédure pas à pas dans son intégralité.

- Sélectionnez le projet **Programme d'installation de Mon Bloc-notes** dans l'Explorateur de solutions. Dans le menu **Projet**, choisissez **Installer**.
Cette action exécute le programme d'installation et installe **Mon Bloc-notes** sur votre ordinateur de développement.

Remarque Vous devez avoir des autorisations d'installation sur l'ordinateur pour pouvoir exécuter le programme d'installation.

Fonctionnalités de déploiement facultatives

Les étapes suivantes illustrent des fonctionnalités de déploiement facultatives.

Création d'un raccourci

Cette étape crée un raccourci vers votre application, lequel sera placé sur le bureau d'un ordinateur cible pendant l'installation.

Pour créer des raccourcis pour l'application Windows

1. Sélectionnez le projet **Programme d'installation de Mon Bloc-notes** dans l'Explorateur de solutions.
2. Dans l'Éditeur du système de fichiers, sélectionnez le nœud **Sortie principale de Mon Bloc-notes (Actif)**.
3. Dans le menu **Action**, choisissez **Créer un raccourci vers Sortie principale de Mon Bloc-Notes**.
Cette action ajoute un nœud **Raccourci vers Sortie principale de Mon Bloc-notes (Actif)**.
4. Renommez le raccourci en **Raccourci vers Mon BLOC-NOTES**.
5. Sélectionnez **Raccourci vers Mon BLOC-NOTES** et faites-le glisser vers le dossier **Bureau de l'utilisateur** dans le volet de gauche.

Création d'une association de fichier

Cette étape ajoute une association de fichier pour Mon Bloc-notes afin qu'un double-clic sur les fichiers .vbn lance l'application Mon Bloc-notes.

Pour créer des associations de fichiers pour l'application Windows

1. Sélectionnez le projet **Programme d'installation de Mon Bloc-notes** dans l'Explorateur de solutions. Dans le menu **Affichage**, pointez sur **Éditeur** et choisissez **Types de fichiers**.
2. Sélectionnez le nœud **Types de fichiers sur l'ordinateur cible** dans l'Éditeur des types de fichiers. Dans le menu **Action**, choisissez **Ajouter un type de fichier**.
Un nœud **Nouveau type de document n° 1** est ajouté et ouvert pour l'attribution d'un nouveau nom.
3. Renommez Nouveau type de document n° 1 en **Vbn.doc**.
4. Dans la fenêtre Propriétés, affectez la valeur **vbn** à la propriété **Extension** du type de fichier.
5. Sélectionnez la propriété **Command**, puis cliquez sur le bouton **Sélection** (...). Dans la boîte de dialogue **Sélectionner un élément dans le projet**, naviguez jusqu'à **Dossier d'application**, puis sélectionnez **Sortie principale de Mon Bloc-notes (Actif)**.
6. Cliquez sur **OK** pour fermer la boîte de dialogue.

Ajout d'une entrée de Registre

Cette étape ajoute une clé de Registre et une valeur au Registre. Vous pouvez faire référence à cette clé de Registre dans le code de votre application pour extraire des informations pour chaque utilisateur au moment de l'exécution.

Pour ajouter une entrée de Registre pour l'application Windows

1. Sélectionnez le projet **Programme d'installation de Mon Bloc-notes** dans l'Explorateur de solutions. Dans le menu **Affichage**, pointez sur **Éditeur**, puis choisissez **Registre**.
2. Sélectionnez le nœud **HKEY_CURRENT_USER** et développez-le, puis développez le nœud **Software** et sélectionnez le nœud **[Manufacturer]**.
Remarque Le nœud Manufacturer est placé entre crochets droits pour indiquer qu'il s'agit d'une propriété. Il sera remplacé par la valeur entrée pour la propriété **Manufacturer** du projet de déploiement.
3. Dans le menu **Action**, choisissez **Nouveau**, puis **Clé**.
4. Renommez la clé **UserChoice**.
5. Sélectionnez la clé **UserChoice**.
6. Dans le menu **Action**, pointez sur **Nouveau**, puis cliquez sur **Valeur de chaîne**.
7. Renommez la valeur **TextColor**.
8. Dans la fenêtre Propriétés, sélectionnez la propriété **Value** et entrez **Black**.

Ajout d'une boîte de dialogue d'installation personnalisée

Cette étape ajoute et configure une boîte de dialogue personnalisée de l'interface utilisateur qui s'affichera au cours de l'installation.

Pour ajouter une boîte de dialogue d'installation personnalisée

1. Sélectionnez le projet **Programme d'installation de Mon Bloc-notes** dans l'Explorateur de solutions. Dans le menu **Affichage**, pointez sur **Éditeur** et choisissez **Interface utilisateur**.
2. Dans l'Éditeur de l'interface utilisateur, sélectionnez le nœud **Début** sous le nœud **Installer**.
3. Dans le menu **Action**, choisissez **Ajouter une boîte de dialogue**.
4. Dans la boîte de dialogue **Ajouter une boîte de dialogue**, choisissez **Cases à cocher (A)**.
5. Cliquez sur **OK** pour fermer la boîte de dialogue.
6. Dans le menu **Action**, choisissez deux fois **Monter** afin de positionner la boîte de dialogue **Cases à cocher (A)** au-dessus de la boîte de dialogue **Dossier d'installation**.
7. Dans la fenêtre Propriétés, affectez à la propriété **BannerText** la valeur **Exemples**.
8. Définissez la propriété **BodyText** à **La case à cocher Installer les exemples vérifie si les fichiers d'exemple sont installés ou non. Si cette case à cocher n'est pas activée, les exemples ne seront pas installés**.
9. Définissez la propriété **CheckBox1Label** à **Souhaitez-vous installer les exemples ?**
10. Affectez la valeur **false** aux propriétés **Checkbox2Visible**, **Checkbox3Visible** et **Checkbox4Visible**. Cela a pour effet de masquer les cases à cocher supplémentaires.

Utilisation d'exemples

Cette étape crée un sous-dossier Exemples qui sera installé sous le dossier Application.

Pour ajouter un dossier Exemples

1. Sélectionnez le projet **Programme d'installation de Mon Bloc-notes** dans l'Explorateur de solutions. Dans le menu **Affichage**, pointez sur **Éditeur** et choisissez **Système de fichiers. Dossier d'application** doit toujours être sélectionné.

2. Dans le menu **Action**, pointez sur **Ajouter**, puis choisissez **Dossier**.
3. Renommez « Nouveau dossier n° 1 » en **Exemples**.

Cette étape crée deux exemples de fichiers texte qui seront installés si l'utilisateur choisit l'option Installer les exemples de la boîte de dialogue personnalisée.

Pour créer des exemples de fichiers pour l'application

1. À l'aide du Bloc-notes ou d'un autre éditeur de texte, créez un fichier texte contenant le texte **Ceci est rules.vbn**. Enregistrez-le sous le nom **Rules.vbn**.

Remarque Pour empêcher le Bloc-notes d'ajouter automatiquement une extension .txt, choisissez **Tous les fichiers** dans la liste déroulante **Types de fichiers**.

2. Créez un autre fichier texte contenant le texte **Ceci est memo.vbn**. Enregistrez-le sous le nom **Memo.vbn**. Cette étape ajoute les exemples de fichiers au dossier Exemples, et définit une condition qui détermine si ces fichiers doivent être installés.

Pour ajouter les exemples au programme d'installation

1. Sélectionnez le projet **Programme d'installation de Mon Bloc-notes** dans l'Explorateur de solutions. Dans le menu **Affichage**, pointez sur **Éditeur**, choisissez **Système de fichiers**, puis sélectionnez le dossier **Exemples**.
2. Dans le menu **Action**, pointez sur **Ajouter**, puis choisissez **Fichier**. Ajoutez les fichiers **Rules.vbn** et **Memo.vbn** au dossier **Exemples**.
3. Sélectionnez le fichier **Rules.vbn** dans l'Éditeur du système de fichiers.
4. Dans la fenêtre Propriétés, affectez la valeur **CHECKBOXA1=1** à la propriété **Condition**. Lorsque le programme d'installation sera exécuté, le fichier **Rules.vbn** ne sera installé que si la case à cocher personnalisée est activée.
5. Dans l'Éditeur du système de fichiers, sélectionnez le fichier **Memo.vbn**.
6. Dans la fenêtre Propriétés, affectez la valeur **CHECKBOXA1=1** à la propriété **Condition**. Lorsque le programme d'installation sera exécuté, le fichier **Memo.vbn** ne sera installé que si la case à cocher personnalisée est activée.

Ajout de conditions de lancement

Cette étape vérifie si Internet Explorer 5.0 ou version ultérieure est installé sur un ordinateur cible, et interrompt l'installation si tel n'est pas le cas.

Pour ajouter une condition de lancement destinée à vérifier la version de Microsoft Internet Explorer

Remarque Cette étape est destinée à illustrer le concept de conditions de lancement ; l'application Mon Bloc-notes n'a pas de véritable dépendance sur Internet Explorer.

1. Sélectionnez le projet **Programme d'installation de Mon Bloc-notes** dans l'Explorateur de solutions. Dans le menu **Affichage**, pointez sur **Éditeur** et choisissez **Conditions de lancement**.
2. Dans l'Éditeur des conditions de lancement, sélectionnez le nœud **Configuration requise pour l'ordinateur cible**.
3. Dans le menu **Action**, choisissez **Ajouter une condition de lancement de fichier**. Un nœud **Recherche du fichier1** est ajouté sous le nœud **Recherche de l'ordinateur cible** et un nœud **Condition1** est ajouté sous le nœud **Conditions de lancement**.
4. Renommez **Recherche du fichier1** en **Recherche de Internet Explorer**.
5. Dans la fenêtre Propriétés, affectez la valeur **Iexplore.exe** à la propriété **FileName**, la valeur **[ProgramFilesFolder]** à la propriété **Folder**, la valeur **2** à la propriété **Depth** et la valeur **5.00** à la propriété **MinVersion**.
6. Sélectionnez le nœud **Condition1**.
7. Définissez la propriété **Message** à **Ce programme requiert Microsoft Internet Explorer 5.0 ou version ultérieure. Installez Internet Explorer, puis réexécutez le programme d'installation du Bloc-notes**.

Définition des propriétés facultatives

Cette étape définit une propriété permettant d'installer automatiquement les fichiers de l'application d'amorçage de Windows Installer si la version appropriée de Windows Installer ne se trouve pas sur l'ordinateur cible.

Pour définir des propriétés facultatives du projet de déploiement

1. Sélectionnez le projet **Programme d'installation de Mon Bloc-notes** dans l'Explorateur de solutions. Dans le menu **Affichage**, choisissez **Pages de propriétés**.
2. Dans la page de propriétés **Programme d'installation de Mon Bloc-notes**, sélectionnez la liste déroulante **Programme d'amorçage**, puis choisissez **Programme d'amorçage de Windows Installer**.
3. Dans le menu **Générer**, choisissez **Générer Programme d'installation de Mon Bloc-notes**.

Installation sur votre ordinateur de développement

Cette étape exécute le programme d'installation et installe Mon Bloc-notes sur votre ordinateur de développement.

Pour installer Mon Bloc-notes sur votre ordinateur de développement

- Sélectionnez le projet **Programme d'installation de Mon Bloc-notes** dans l'Explorateur de solutions. Dans le menu **Projet**, choisissez **Installer**.
Remarque Vous devez avoir des autorisations d'installation sur l'ordinateur pour pouvoir exécuter le programme d'installation.

Déploiement vers un autre ordinateur

Cette étape exécute le programme d'installation et installe Mon Bloc-notes sur un autre ordinateur.

Pour déployer Mon Bloc-notes vers un autre ordinateur

1. Dans l'Explorateur Windows, naviguez jusqu'à votre répertoire de projet et recherchez le programme d'installation généré. Le chemin d'accès par défaut sera `\Documents and Settings\votrenomdeconnexion\Programme d'installation de Mon Bloc-notes\configuration de projet\Programme d'installation de Mon Bloc-notes.msi`. La *configuration de projet* par défaut est Déboguer.
2. Copiez le fichier Programme d'installation de Mon Bloc-notes.msi ainsi que tous les autres fichiers et sous-répertoires du répertoire sur un autre ordinateur.
Remarque Pour effectuer l'installation sur un ordinateur ne se trouvant pas sur un réseau, copiez les fichiers sur des supports traditionnels, tels que des CD-ROM.
3. Sur l'ordinateur cible, double-cliquez sur le fichier **Setup.exe** pour exécuter le programme d'installation.
Remarque Vous devez avoir des autorisations d'installation sur l'ordinateur pour pouvoir exécuter le programme d'installation.

Test

Cette étape teste le démarrage du programme d'installation à partir d'un raccourci clavier et sa désinstallation.

Pour tester l'installation

1. Vérifiez que le raccourci est installé sur le bureau et qu'il lance correctement l'application Mon Bloc-notes.
2. Désinstallez l'application à l'aide de l'outil **Ajout/Suppression de programmes** du Panneau de configuration.
Conseil Pour effectuer la désinstallation depuis votre ordinateur de développement, dans le menu **Projet**, choisissez **Désinstaller**.

Définition des propriétés facultatives

La procédure pas à pas suivante décrit le processus de création d'une action personnalisée DLL permettant de diriger un utilisateur vers une page Web à la fin d'une installation.

Pour créer l'action personnalisée

1. Dans le menu **Fichier**, pointez sur **Nouveau**, puis choisissez **Projet**.
2. Dans la boîte de dialogue **Nouveau projet**, sélectionnez **Projets Visual Basic** dans le volet **Type de projet**, puis choisissez **Application Windows** dans le volet **Modèles**. Dans la zone **Nom**, tapez **OpenWeb**.
3. Cliquez sur **OK** pour fermer la boîte de dialogue.
Le projet est ajouté à l'Explorateur de solutions.
4. Dans le menu **Projet**, sélectionnez **Ajouter un module**, puis sélectionnez **Module** dans la boîte de dialogue **Ajouter un nouvel élément**. Cliquez sur **OK** pour fermer la boîte de dialogue.
5. Dans l'éditeur de code, ajoutez du code à Module1 pour ouvrir une page Web :

```
' Visual Basic
Public Sub Main()
    Process.Start("http://www.microsoft.com")
End Sub
```
6. Dans l'Explorateur de solutions, sélectionnez le projet **OpenWeb**. Dans le menu **Projet**, choisissez **Propriétés**.
7. Dans la boîte de dialogue **Pages de propriétés de OpenWeb**, sélectionnez la propriété d'objet **Startup** et affectez-lui la valeur **Sub Main**. Cliquez sur **OK** pour fermer la boîte de dialogue.
8. Dans le menu **Générer**, choisissez **Générer OpenWeb**.

Pour ajouter un projet de déploiement

1. Dans le menu **Fichier**, pointez sur **Ajouter un projet**, puis sélectionnez **Nouveau**.
2. Dans la boîte de dialogue **Ajouter un nouveau projet**, sélectionnez **Projets de configuration et de déploiement** dans le volet **Type de projet**, puis choisissez **Projet de configuration** dans le volet **Modèles**. Dans la zone **Nom**, tapez **Programme d'installation d'actions personnalisées**.
3. Cliquez sur **OK** pour fermer la boîte de dialogue.

L'Éditeur du système de fichiers s'affiche.

4. Dans l'Éditeur du système de fichiers, sélectionnez Dossier d'application. Dans le menu **Action**, pointez sur **Ajouter**, puis choisissez **Sortie du projet**.
5. Dans la boîte de dialogue **Ajouter le groupe de sortie du projet**, sélectionnez la sortie principale pour le projet **Mon Bloc-notes**. Cliquez sur **OK** pour fermer la boîte de dialogue.
6. Dans l'Éditeur du système de fichiers, sélectionnez Dossier d'application. Dans le menu **Action**, pointez sur **Ajouter**, puis choisissez **Sortie du projet**.
7. Dans la boîte de dialogue **Ajouter le groupe de sortie du projet**, sélectionnez la sortie principale pour le projet OpenWeb. Cliquez sur **OK** pour fermer la boîte de dialogue.

Pour ajouter l'action personnalisée

1. Sélectionnez le projet **Programme d'installation d'actions personnalisées** dans l'Explorateur de solutions. Dans le menu **Affichage**, pointez sur **Éditeur** et choisissez **Actions personnalisées**. L'Éditeur des actions personnalisées s'affiche.
2. Dans l'Éditeur des actions personnalisées, sélectionnez le nœud **Installer**. Dans le menu **Action**, choisissez **Ajouter une action personnalisée**.
3. Dans la boîte de dialogue **Sélectionner un élément dans le projet**, double-cliquez sur **Dossier d'application**.
4. Sélectionnez l'élément **Sortie principale de Mon Bloc Note (Actif)**.
5. Cliquez sur **OK** pour fermer la boîte de dialogue.
6. Dans l'Éditeur des actions personnalisées, sélectionnez le nœud **Installer**. Dans le menu **Action**, choisissez **Ajouter une action personnalisée**.
7. Dans la boîte de dialogue **Sélectionner un élément dans le projet**, double-cliquez sur **Dossier d'application**.
8. Sélectionnez l'élément **Sortie principale de OpenWeb (Actif)**.
9. Cliquez sur **OK** pour fermer la boîte de dialogue.
10. Dans la fenêtre Propriétés, sélectionnez la propriété **InstallerClass** et attribuez-lui la valeur **false**.
11. Dans le menu **Générer**, choisissez **Générer Programme d'installation d'actions personnalisées**.

Suivre les mêmes étapes décrites précédemment pour effectuer l'installation sur votre ordinateur de développement, pour effectuer le déploiement vers un autre ordinateur ou pour désinstaller l'application

Remarque : cette partie ne correspond pas exactement à msdn.Microsoft.Com et a été modifiée pour des besoins pédagogiques

Procédure pas à pas : utilisation d'une action personnalisée pour créer une base de données au cours de l'installation

La procédure pas à pas suivante décrit l'utilisation d'une action personnalisée et de la propriété **CustomActionData** pour créer une base de données pendant l'installation.

Remarque Pour cette procédure, SQL Server doit être installé sur l'ordinateur sur lequel vous déployerez l'application.

Pour créer une classe de programme d'installation

1. Dans le menu **Fichier**, pointez sur **Nouveau**, puis choisissez **Projet**.
2. Dans la boîte de dialogue **Nouveau projet**, sélectionnez **Projets Visual Basic** dans le volet **Type de projet**, puis choisissez **FormulaireWindows** dans le volet **Modèles**. Dans la zone **Nom**, tapez **DBCcustomAction**.
3. Cliquez sur **OK** pour fermer la boîte de dialogue.
4. Dans le menu **Projet**, choisissez **Ajouter un nouvel élément**.
5. Dans la boîte de dialogue **Ajouter un nouvel élément**, choisissez **Classe Installer**. Dans la zone **Nom**, tapez **DBCcustomAction**.
6. Cliquez sur **OK** pour fermer la boîte de dialogue.

Pour ajouter de code à la classe de programme d'installation de création de la base

1. Dans l'Explorateur de solutions, sélectionnez **DBCcustomAction.vb**. Dans le menu **Affichage**, choisissez **Code**.
2. Compléter le code de manière à obtenir :

```

Imports System.ComponentModel
Imports System.Configuration.Install
Imports System.Data.SqlClient
<RunInstaller(True)> Public Class Installer1
    Inherits System.Configuration.Install.Installer
    Sub CréerBase(ByVal Nom_Serveur As String)
        Dim cn As New SqlConnection
        cn.ConnectionString = "Database=master;server=" & Nom_Serveur & ";Integrated Security=SSPI"
        cn.Open()
        Dim t As SqlTransaction
        t = cn.BeginTransaction()
        Dim cmd As New SqlCommand
        cmd.Connection = cn
        cmd.CommandType = CommandType.Text
        Try
            cmd.CommandText = "create database Test"
            cmd.ExecuteNonQuery()
            cn.ChangeDatabase("test")
            cmd.Transaction = t
            cmd.CommandText = "create Table Table1(ChTab1_1 int primary key, ChTab1_2 varchar(100))"
            cmd.ExecuteNonQuery()
            cmd.CommandText = "create Table Table2(ChTab2_1 int primary key, ChTab2_2 varchar(100),
                ChTab2_3 int constraint aa Foreign key references Table1)"
            cmd.ExecuteNonQuery()
            t.Commit()
            cn.Close()
        Catch ex As Exception
            MessageBox.Show("Erreur")
        End Try
    End Sub
End Sub

Public Overrides Sub Install(ByVal stateSaver As System.Collections.IDictionary)
    CréerBase(Me.Context.Parameters.Item("NomServeur"))
End Sub
End Class

```

3. Dans le menu **Générer**, choisissez **Générer DBCustomAction**.

Pour créer un projet de déploiement

1. Dans le menu **Fichier**, choisissez **Ajouter un projet, Nouveau projet**.
2. Dans la boîte de dialogue **Ajouter un projet**, sélectionnez **Projets de configuration et de déploiement** dans le volet **Type de projet**, puis choisissez **Projet de configuration** dans le volet **Modèles**. Dans la zone **Nom**, tapez **Programme d'installation de DBCustomAction**.
3. Cliquez sur **OK** pour fermer la boîte de dialogue.
4. Dans la fenêtre Propriétés, sélectionnez la propriété **ProductName** et tapez **DB Installer**.
5. Dans l'Éditeur du système de fichiers, sélectionnez **Dossier d'application**. Dans le menu **Action**, choisissez **Ajouter, Sortie du projet**.
6. Dans la boîte de dialogue **Ajouter le groupe de sortie du projet**, sélectionnez la sortie principale pour le projet **DBCustomAction**. Cliquez sur **OK** pour fermer la boîte de dialogue.

Pour créer une boîte de dialogue d'installation personnalisée

1. Sélectionnez le projet **Programme d'installation de DBCustomAction** dans l'Explorateur de solutions. Dans le menu **Affichage**, pointez sur **Éditeur** et choisissez **Interface utilisateur**.
2. Dans l'Éditeur de l'interface utilisateur, sélectionnez le nœud **Début** sous **Installer**. Dans le menu **Action**, choisissez **Ajouter une boîte de dialogue**.
3. Dans la boîte de dialogue **Ajouter une boîte de dialogue**, sélectionnez la boîte de dialogue **Zones de texte (A)**, puis cliquez sur **OK** pour fermer la boîte de dialogue.
4. Dans le menu **Action**, choisissez **Monter**. Répétez l'opération jusqu'à ce que la boîte de dialogue **Zones de texte (A)** se trouve au-dessus du nœud **Dossier d'installation**.

5. Dans la fenêtre Propriétés, sélectionnez la propriété **BannerText** et tapez **Spécifier le nom du serveur**.
6. Sélectionnez la propriété **BodyText** et tapez **Cette boîte de dialogue vous permet de spécifier le nom du serveur**.
7. Sélectionnez la propriété **Edit1Label** et tapez **Nom du serveur :**.
8. Sélectionnez la propriété **Edit1Property** et tapez **TXT1**.
9. Sélectionnez les propriétés **Edit2Visible**, **Edit3Visible** et **Edit4Visible**, et affectez-leur la valeur **false**.

Pour créer une action personnalisée

1. Sélectionnez le projet **Programme d'installation de DBCustomAction** dans l'Explorateur de solutions. Dans le menu **Affichage**, pointez sur **Éditeur** et choisissez **Actions personnalisées**.
2. Dans l'Éditeur des actions personnalisées, sélectionnez le nœud **Installer**. Dans le menu **Action**, choisissez **Ajouter une action personnalisée**.
3. Dans la boîte de dialogue **Sélectionner un élément dans le projet**, double-cliquez sur **Dossier d'application**.
4. Sélectionnez l'élément **Sortie principale de DBCustomAction (Actif)**, puis cliquez sur **OK** pour fermer la boîte de dialogue.
5. Dans la fenêtre Propriétés, sélectionnez la propriété **CustomActionData** et tapez **/NomServeur=[TXT1]**.
6. Dans le menu **Générer**, choisissez **Générer Programme d'installation de DBCustomAction**.

Procédure pas à pas : création et utilisation d'un module de fusion

Cette procédure pas à pas montre comment empaqueter un composant dans un module de fusion et comment utiliser ce module de fusion dans un projet de configuration.

Pour créer un composant

1. Dans le menu **Fichier**, pointez sur **Nouveau**, puis choisissez **Projet**.
2. Dans la boîte de dialogue **Nouveau projet**, sélectionnez **Projets Visual C#** dans le volet **Type de projet**, puis choisissez **Bibliothèque de classes** dans le volet **Modèles**. Dans la zone **Nom**, tapez **MonComposant**.
3. Cliquez sur **OK** pour fermer la boîte de dialogue.
4. Dans le menu **Générer**, choisissez **Générer MonComposant** pour générer la bibliothèque de classes.

Pour créer un projet de module de fusion

1. Dans le menu **Fichier**, pointez sur **Ajouter un projet**, puis choisissez **Nouveau projet**.
2. Dans la boîte de dialogue **Ajouter un nouveau projet**, sélectionnez **Projets de configuration et de déploiement** dans le volet **Type de projet**, puis choisissez **Projet de module de fusion** dans le volet **Modèles**. Dans la zone **Nom**, tapez **Exemple**.
3. Cliquez sur **OK** pour fermer la boîte de dialogue.
4. Dans l'Éditeur du système de fichiers, sélectionnez **Dossier Remplacement pour les modules**. Dans le menu **Action**, pointez sur **Ajouter**, puis choisissez **Sortie du projet**.
5. Dans la boîte de dialogue **Ajouter le groupe de sortie du projet**, sélectionnez **Sortie principale** pour le projet **MonComposant**. Cliquez sur **OK** pour fermer la boîte de dialogue.
6. Dans le menu **Générer**, choisissez **Générer Exemple** pour générer le module de fusion.

Pour ajouter un projet de configuration qui utilise le module de fusion

1. Dans le menu **Fichier**, choisissez **Ajouter un projet, Nouveau projet**.
2. Dans la boîte de dialogue **Ajouter un projet**, sélectionnez **Projets de configuration et de déploiement** dans le volet **Type de projet**, puis choisissez **Projet de configuration** dans le volet **Modèles**. Dans la zone **Nom**, tapez **Programme d'installation du module de fusion**.
3. Cliquez sur **OK** pour fermer la boîte de dialogue.
4. Dans l'Éditeur du système de fichiers, sélectionnez **Dossier d'application**. Dans le menu **Action**, pointez sur **Ajouter**, puis choisissez **Sortie du projet**.
5. Dans la boîte de dialogue **Ajouter le groupe de sortie du projet**, sélectionnez le projet **Exemple**, puis **Module de fusion**.
6. Cliquez sur **OK** pour fermer la boîte de dialogue.

Notez que le module de fusion n'est pas ajouté au dossier. Il est ajouté au projet Programme d'installation du module de fusion dans l'Explorateur de solutions. La raison à cela est que les modules de fusion ne peuvent pas être installés directement ; ils doivent être fusionnés dans un programme d'installation.

7. Sélectionnez le module de fusion (**Module de fusion de Exemple (Actif)**) dans l'Explorateur de solutions. Dans la fenêtre Propriétés, sélectionnez la propriété **KeyOutput** et développez le nœud pour afficher le nœud

(**MergeModuleProperties**), puis développez ce nœud et sélectionnez la propriété **Dossier Remplacement pour les modules**.

Remarque Les propriétés affichées sous la propriété **KeyOutput** sont des propriétés dynamiques qui n'apparaissent qu'après la génération du module de fusion.

8. Sélectionnez la liste déroulante associée à la propriété **Dossier Remplacement pour les modules**, puis cliquez sur **Parcourir** pour afficher la boîte de dialogue **Sélectionner un dossier**.
9. Dans la boîte de dialogue **Sélectionner un dossier**, choisissez **Dossier d'application**. Cliquez sur **OK** pour fermer la boîte de dialogue.
10. Dans le menu **Générer**, choisissez **Générer Programme d'installation du module de fusion**.

Suivre les mêmes étapes décrites précédemment pour effectuer l'installation sur votre ordinateur de développement, pour effectuer le déploiement vers un autre ordinateur ou pour désinstaller l'application