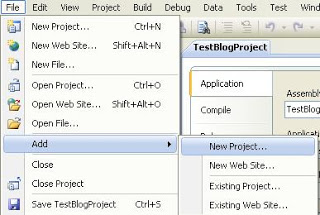
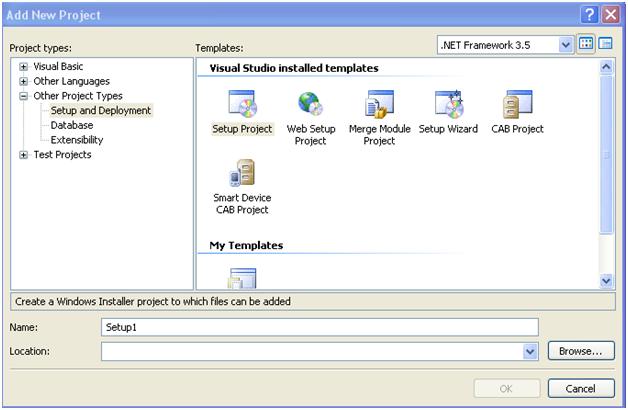
**Creating a Setup and Deployment Project**

**Step 1:** Open the Sales Force Automation project in the Visual Studio IDE (integrated development environment).

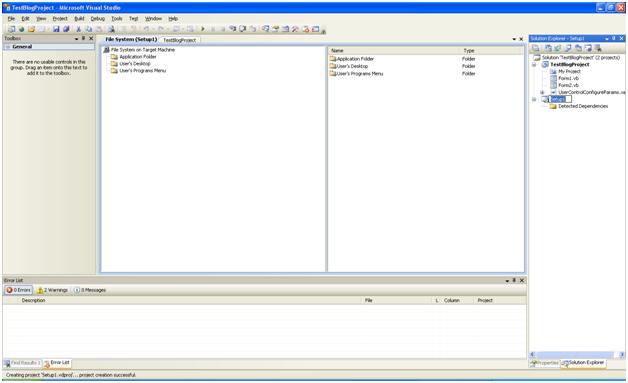
**Step 2:** Click on File, Add, New Project:



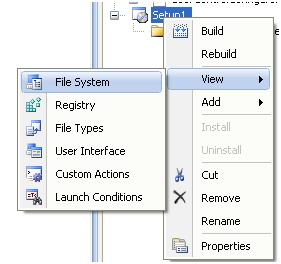
**Step 3:** Click once on ‘Setup Project’. Next, you can rename the project to ‘KC01 Setup’ or whatever you want to name it. You can also choose a different location by clicking on ‘Browse’. If you leave it at is, it will create the setup folder in the same location as the KC01 folder.



**Step 4:** Next you will see a screen that looks like this:



**Step 5:** Right Click on the Setup Project, click View, and then click File System:



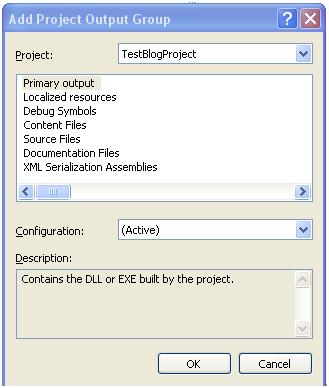
**File System Editor**  
The File System provides you the essential tools to work with the application folder, user's desktop, and the user's programs menu.



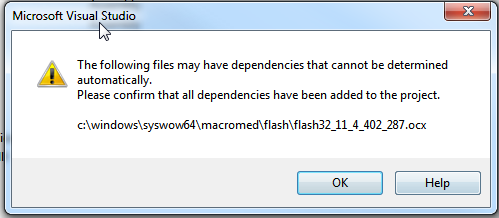
The first thing we want to do here is to include the relevant files and folders we wish to install on the user's machine. This is accomplished by right clicking the Application Folder (on left side of the screen) and selecting 'Add'. The following screenshot is an example of something similar you may see.



Now to add project output files to this folder, Click Primary Output and click ‘OK’

. 

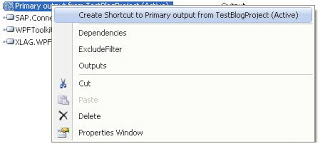
You may see a warning message that looks like this:



This is just a warning that an .ocx file may have its own dependencies that Visual Studio cannot determine. This Flash .ocx does not have any dependent files.

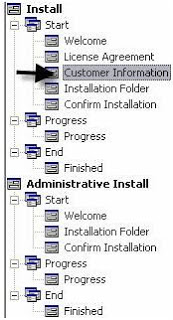
You can take additional steps such as manipulating the user's desktop with items such as a shortcut. To do this, right click on the “User’s Desktop” folder and click on “Create Shortcut to User’s Desktop”. Name the shortcut whatever you want (i.e. “St Croix Sales Force Automation”)

**Create Shortcut**



**Step 6:** The next item you find of immense importance is the User Interface Editor. You can get to it by clicking on the View menu, Editor, User Interface. Or get to it by right clicking on ‘Setup1’ in the Solution Explorer, click View, User Interface.

Inside of this editor you may eliminate or include dialogs that are needed for this install project. The following screenshot is the default User Interface that is created when I initially created this setup project. The only variation is that I have added a Customer Information dialog.



Each of these represents a dialog box in the installation routine. You can right-click on each of these and click on ‘Properties Windows’ to see what properties are attached to each.

**THIS “PREREQUISITES” SECTION IS AN OPTIONAL STEP AS ALL DEPENDENCIES SHOULD AUTOMATICALLY HAVE BEEN INCLUDED IN THE PROJECT IN STEP 5 HOWEVER THE INFORMATION BELOW WILL HELP YOU IF YOU WOULD LIKE TO ADD/REMOVE DEPENDENCIES OR CHOOSE A DIFFERENT INSTALL LOCATION FOR THEM.**

**A PREREQUISITE THAT WAS NOT INCLUDED AUTOMATICALLY IS** [Crystal Reports Basic For Visual Studio 2008 x86 Redistributable Package (32-bit)](http://resources.businessobjects.com/support/communityCS/FilesAndUpdates/CRRedist2008_x86.zip).

**PLEASE REFER TO THE “SALES FORCE AUTOMATION VERSION 8 INSTALLATION TASK LIST.DOC” DOCUMENT IN THE SOLUTION EXPLORER FOR ADDITIONAL INFORMATION**

**Prerequisites Installation in Windows Installer**  
Most applications have prerequisites: Components such as the .NET Framework runtime must be available on a target computer in order for the application to run. The deployment tools in Visual Studio include the capability to automatically detect the existence of components during installation and install a predetermined set of prerequisites — a process known as ***bootstrapping*.**  
  
This will be achieved as follows:  
  
**To choose which prerequisites to install**

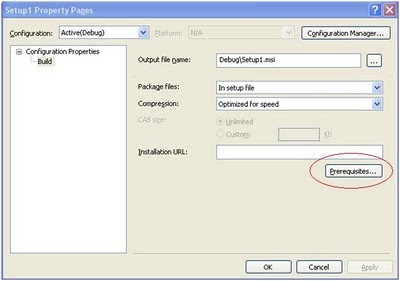
1. In **Solution Explorer**, select the deployment project and Right Click
2. Click **Properties**.
3. In the **Property Pages** dialog box, expand the **Configuration Properties** node, and then select the **Build** property page.
4. Click the **Prerequisites** button.
5. In the **Prerequisites** dialog box, make sure that the **Create setup program to install prerequisite components** box is checked.
6. In the **Choose which prerequisites to install** list, check the prerequisites that you wish to install, and then click **OK**.

**To specify the download location for prerequisites**

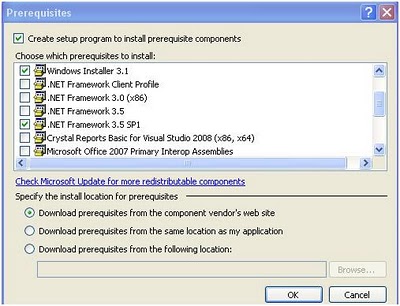
* In **Solution Explorer**, select the deployment project, right click.
* Click **Properties**.
* In the **Property Pages** dialog box, expand the **Configuration Properties** node, and then select the **Build** property page.
* Click the **Prerequisites** button.

In the **Prerequisites** dialog box, choose a location:

* + If you want to deploy the installers for the prerequisites to a vendor, click **Download prerequisites from the component vendor's web site**.
  + If you want to deploy the installers for the prerequisites to the same location as your application installer, click **Download prerequisites from the same location as my application**.
  + If you want to deploy the installers for the prerequisites to a different location, click **Download prerequisites from the following location** and enter a local path, URL, or file-share location.



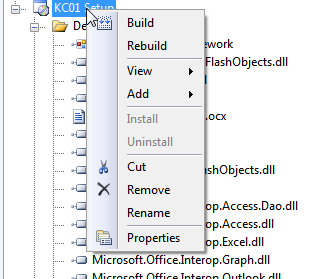
* Click **OK** to continue.



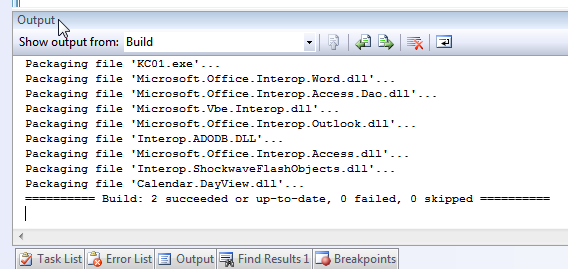
**LAST STEP:** To actually create the setup files, click on the “Solution Configurations” drop down box (directly below the menus) and make sure that it says “Release” and not “Debug”:



Next, right click on the setup project in the Solution Explorer and click on Build.



You will now see the Output window at the bottom of the screen and if you have successfully compiled the setup project you will see that in the window, as follows:



The setup files are now in the folder that you chose in step 3 in the Release folder. You can copy that folder to a CD/DVD if you would like a copy of it on a portable disc.

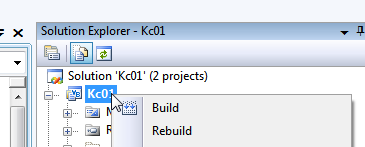
**To create just an .exe do this:**

**NOTE:** If you made changes to the source code but did not add anything that would require a new reference file (i.e. .dll, ocx, etc) than you can just create an executable.

To do this click on the “Solution Configurations” drop down box (directly below the menus) and make sure that it says “Release” and not “Debug”:



Next, right click on the “KC01” project in the Solutions Explorer and click on build.



You will see the same output window you saw in the last step and it will tell you it was successful unless there are errors in the program. Errors are listed in the “Error List” tab at the bottom of the screen. The .exe will be in the <project folder>\obj\release folder. You can rename it “KC01test.exe”, copy and paste it to TS5 if you want to test the changes. Otherwise, just copy and replace the “KC01.exe” file on TS5.