How can I use UDI for application prompting during Post-Delivery Configuration in a Dell CFI project?

Category MDT 2012 ; Procedural documentation for customers ; General Information (Tribal Knowledge) ; Tips & Tricks

Note: This article is directed toward customers who already have an MDT task sequence prepared for deployment through the Dell factory. If you need guidance on getting to that point the following resource is your Bible: [**Whitepaper – Configure Microsoft Deployment Toolkit 2013 for Dell factory image load by Configuration Services / Custom Factory Integration**](http://en.community.dell.com/dell-groups/dtcmedia/m/mediagallery/20442113/download).

In an MDT or SCCM project intended to boot in the Dell factory there is an important delimiter. The Post-Delivery Configuration group, which you will be directed to add during the integration of the ‘factory hooks’ outlined in the user guide, represents the point at which execution of the task sequence continues upon unboxing at the customer site.

Prior to Post-Delivery Configuration there are two important restrictions in place. First, it must be assumed that there is no visibility to the customer network. Second, and more relevant to this article, is that no user interaction can occur. After all, there will be no user present in the Dell factory to answer any questions. Because of this, UDI (User-Driven Installation) cannot be used prior to Post-Delivery Configuration in any boot-in-factory task sequence.

##### Note: This article applies specifically to a Dell CFI boot-in-factory project. If you are looking for general information on UDI usage then this is not applicable.

# General considerations with UDI

It is common these days for customers who are leveraging MDT to have some UDI elements integrated into their task sequences. A full discussion of what is entailed in leveraging every possible UDI scenario during Post-Delivery Configuration is outside of the scope of this article, but since we are discussing UDI I will point you in the right direction.

In most cases, UDI is setting task sequence variables based on the responses it receives to its dialogs. If the actions which consume a particular variable have already taken place, then setting the variable is pointless. This is why UDI is usually presented at the outset of a task sequence. While it is certainly possible to move UDI out to the Post-Delivery Configuration phase if a boot-in-factory project, careful consideration must be given to the chronology of events in the task sequence. In this case, we may very well be setting variables after the point when they are needed, and custom task sequence steps must be added to act upon the variables, and careful testing must be performed to verify that all intended tasks are being accomplished.

A broader discussion of all the possible implications is outside of the scope of this article. Instead, I will focus on one very common inquiry which we receive: During Post-Delivery Configuration, how can present the end user (or bench technician as the case may be) with a list of applications and allow them to choose which ones to install?

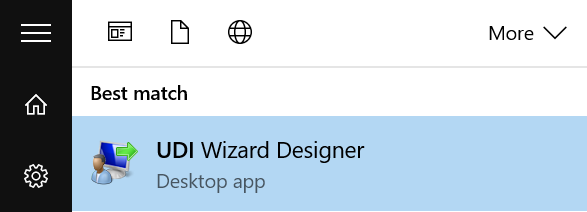
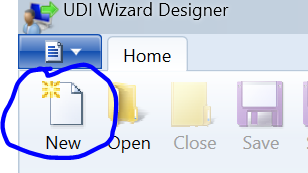
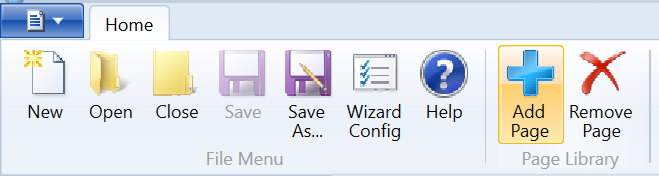
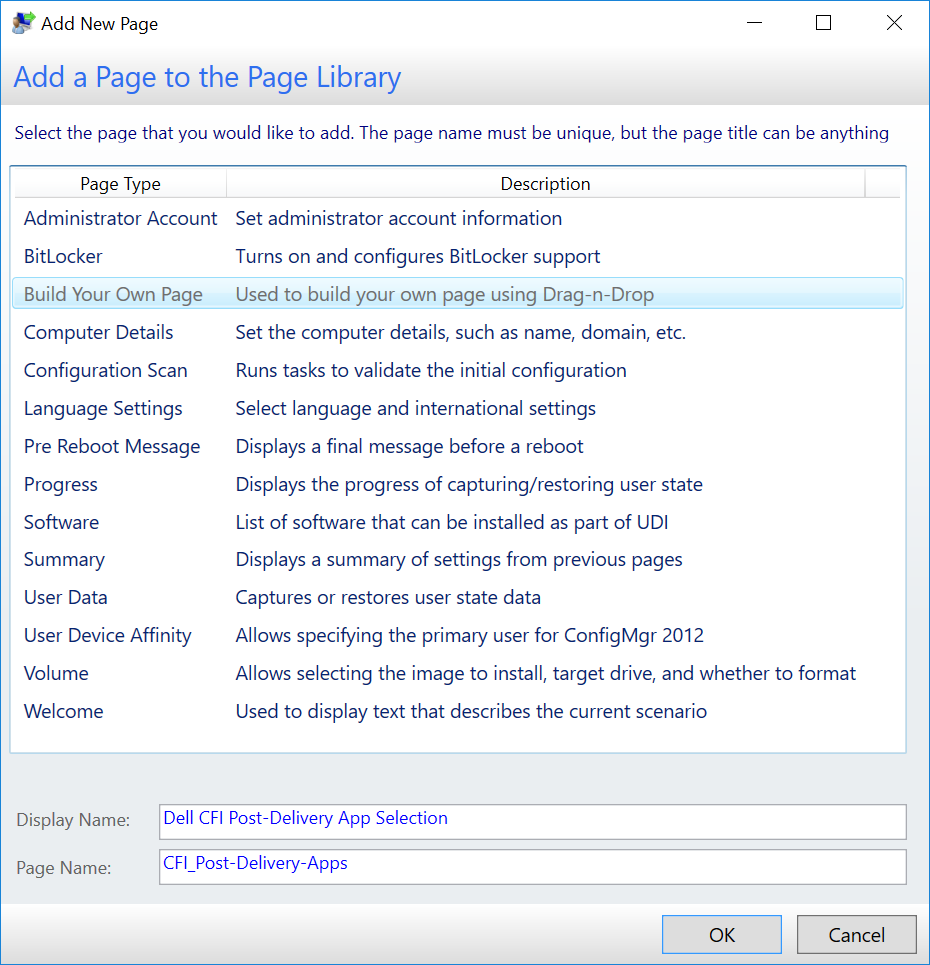
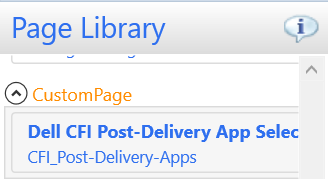
# Using UDI to prompt for applications to install during Post-Delivery Configuration

Here are the steps we will take to control application installation based on user responses:

* Create a custom UDI wizard definition which lists the applications
* Set a custom task sequence variable for each application selected in the UDI wizard
* Use the custom task sequence variable for each application to condition the corresponding application install in the task sequence.

## Create a custom UDI wizard definition which lists the applications

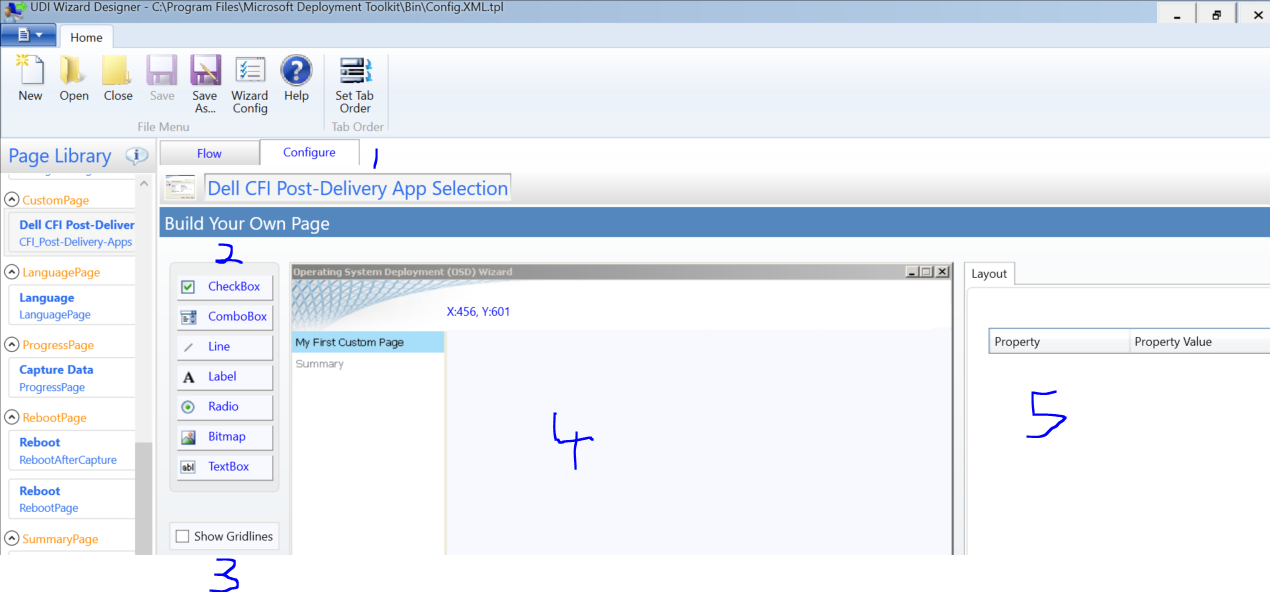
### Creating your page

1. From the Start Menu type ‘UDI’ and choose the UDI Wizard Designer  
   
2. Choose the ‘New’ icon from the ribbon menu  
   
3. Choose the ‘Add Page’ icon on the ribbon menu  
   
4. Choose ‘Build Your Own Page’. Give it a descriptive Display Name and Page Name.  
   
5. In the Page Library in the left pane, double-click your new page to open it in the design pane.  
   

### Designing your page

#### Getting to know the designer

Take a few moments to discover the features available to you.

* At the top of the design pane you will see two tabs: The ‘Flow’ tab and the ‘Configure’ tab. Make sure the ‘Configure’ tab is selected (1).
* At the left side of the design window you will see the controls available to you (2).
* Below the control toolbox is a checkbox for ‘Show Gridlines’. Toggling this on while placing controls will make alignment easier (3).
* In the main part of the window you will see a blank form on which you can place elements from the control toolbox as you design the form (4).
* To the far right you will see a ‘Layout’ pane which lists properties and their corresponding values. This will be blank for now until you start populating the form (5).  
  

#### Choosing your control type

We will primarily be using two kinds of controls for the application list:

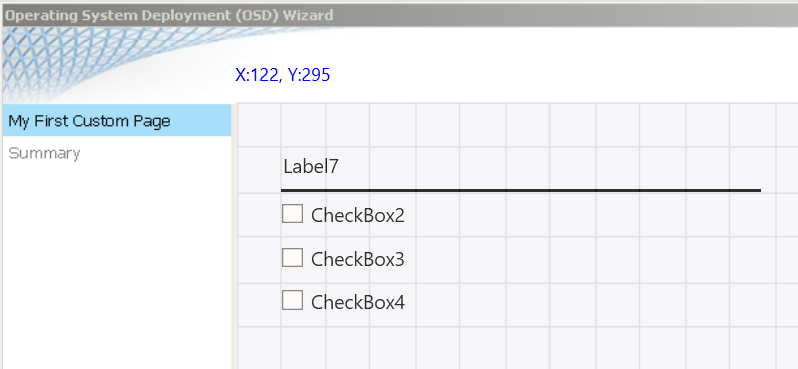
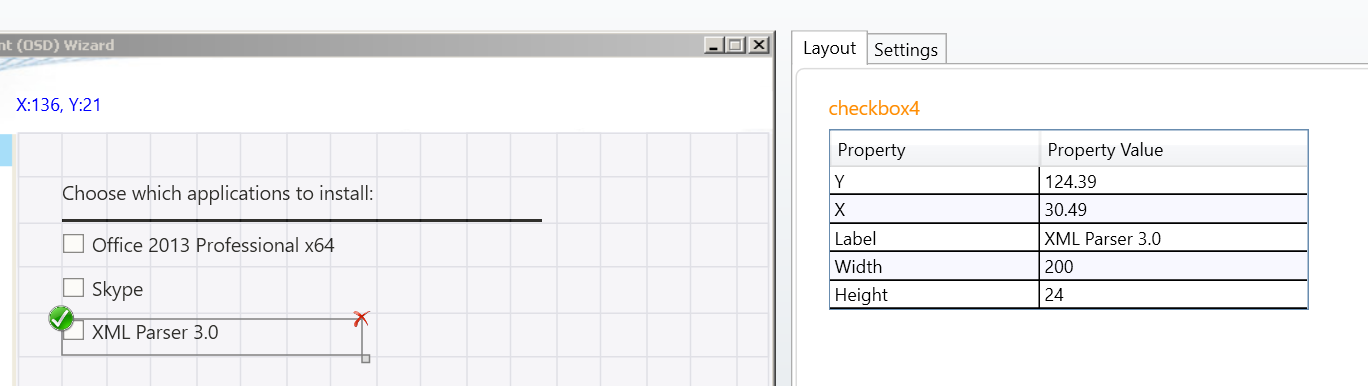
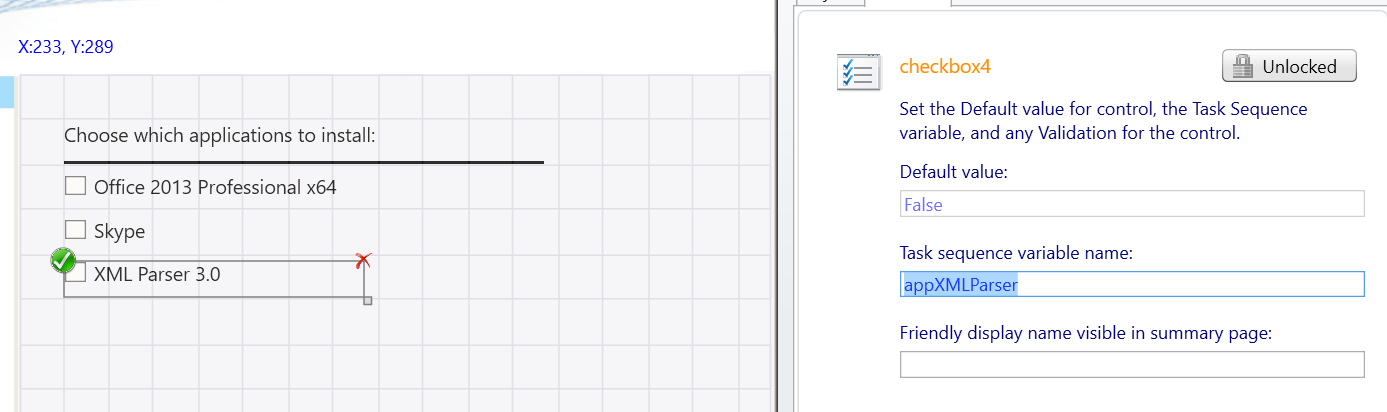
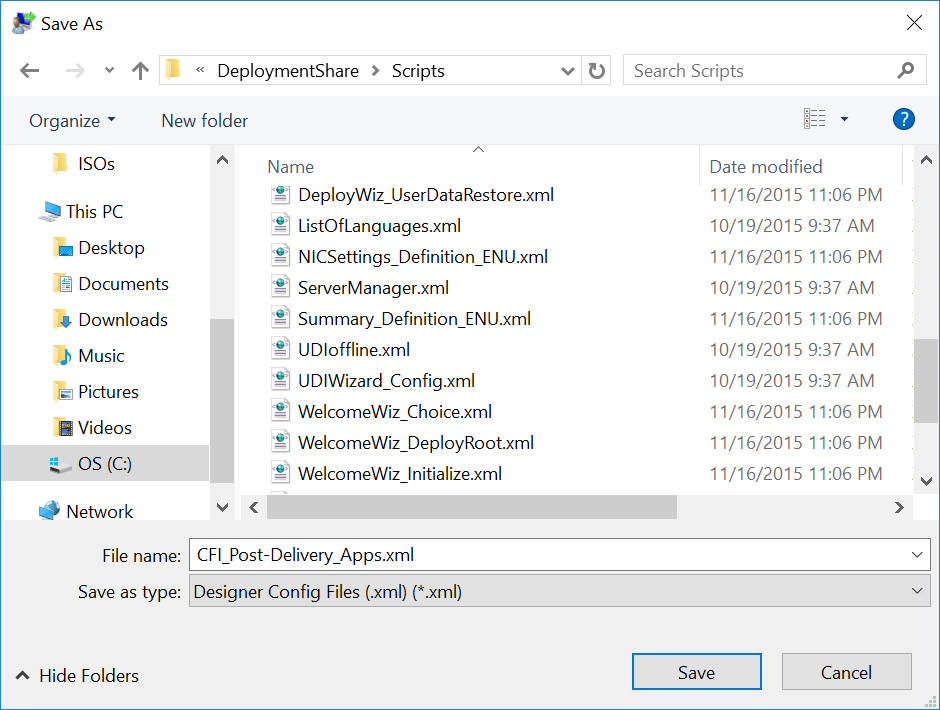
* CheckBox controls are not mutually exclusive and will therefore be used where a list of applications which can be chosen simultaneously.
* ‘Radio’ controls are mutually exclusive and will be used for either/or choices. An example of this would be a choice to install either the Finance application bundle or the HR application bundle, or a choice to install either Office 2013 or Office 365.

Other controls such as labels, lines and bitmaps can be used to improve the visual presentation of the dialog your users will see, but are not related to the actual control of application installation.

#### The basic method – a list of applications

The most common request we get is to just display a list of applications which are available and allow the user to choose which ones they would like to install. To accomplish this, we are going to use CheckBox controls. We will also add a few visual elements such as a label and a line, but I will leave those to you to figure out as they are pretty intuitive.

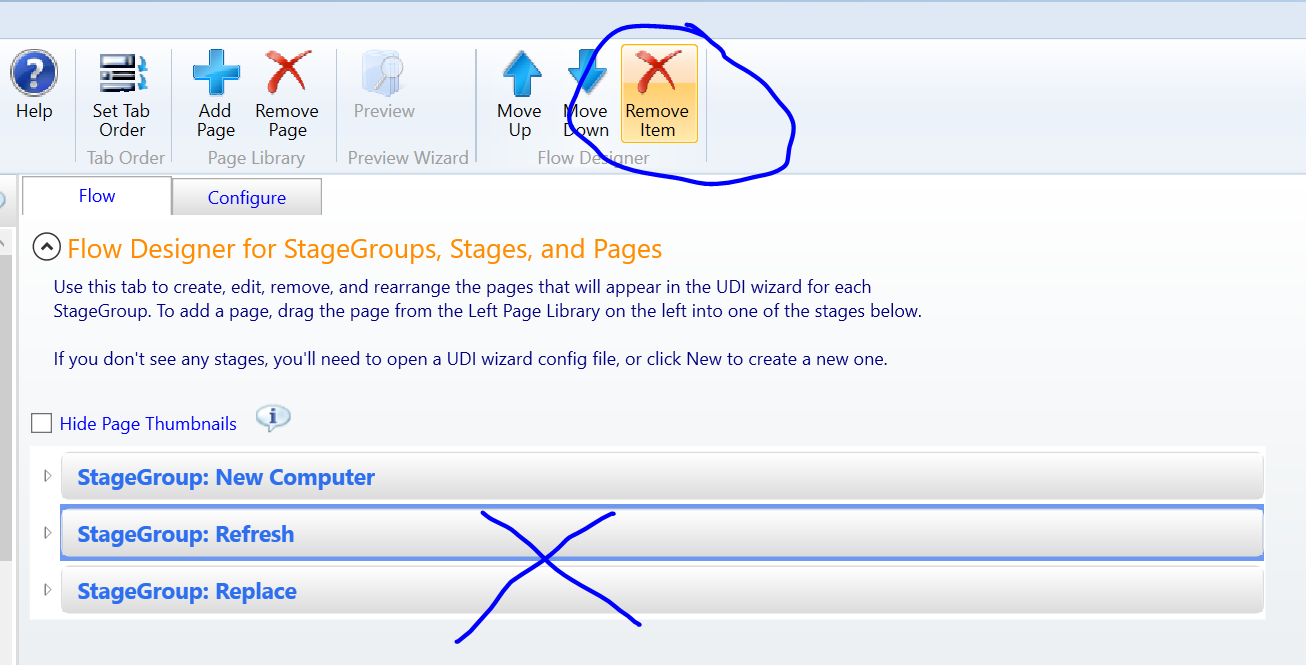
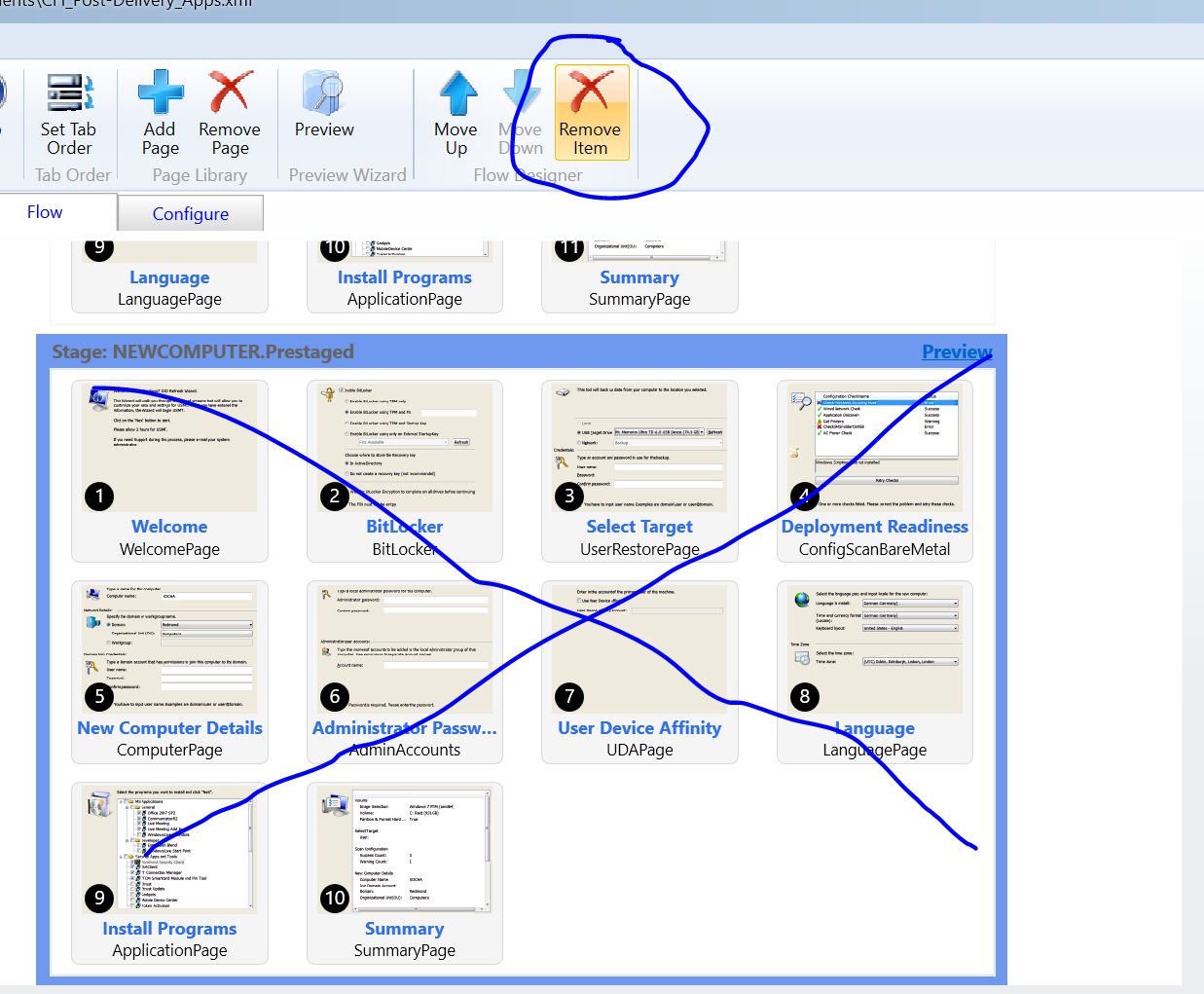
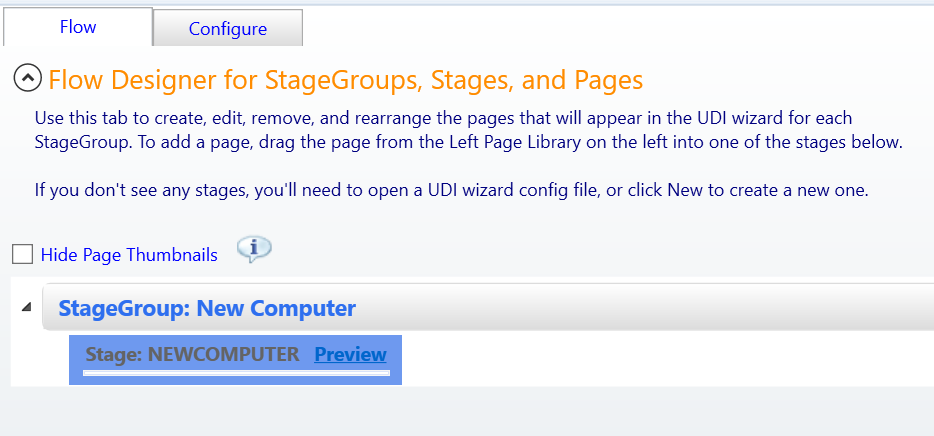
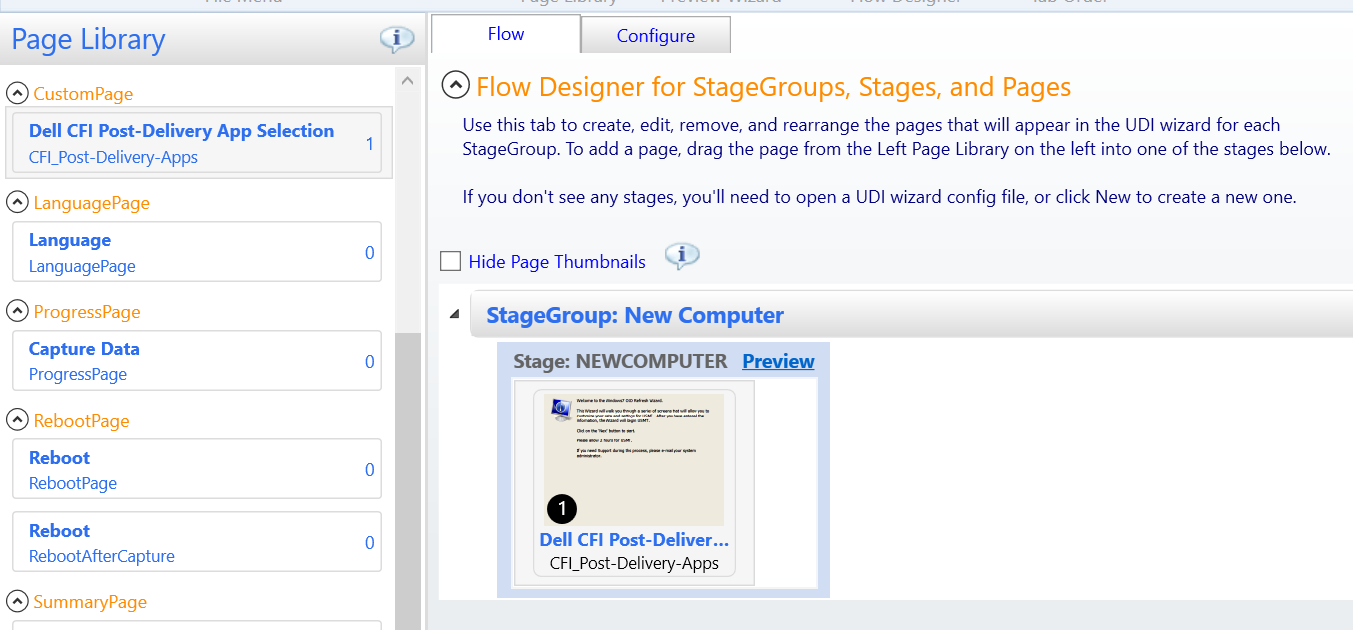
In this example we will allow the unboxer (end user or bench tech) to choose from Office 2013, Skype, and XML Parser. If you are following along (you should be) feel free to substitute some applications from your own MDT instance.

1. After checking Show Gridlines add a label, a line, and three CheckBox controls and align them. To add a control to the form, click it in the control toolbox and hold down the left mouse button while dragging it to the form. To reposition a control, single-click on it, and then click and drag it to reposition. When you have everything aligned to your liking, you can turn the gridlines back off. Don’t worry about the number suffixes, we are going to rename everything anyway.  
   
2. Click on the Label control and notice that the properties and their values are shown on the right. Click in the Property Value field to the right of Label and type whatever you want to appear in the label.  
   
3. Repeat for each of the checkboxes, changing the Label property’s value to the name of one of your pieces of software.  
   
4. Notice that when you select a CheckBox control a new tab Settings appears in the right pane. For each application checkbox you will create a new task sequence variable name to associate with it. Choose something descriptive, but don’t use spaces or special characters, and don’t use [native task sequence variables](https://technet.microsoft.com/en-us/library/bb632442.aspx). This variable will store the value of True if the unboxer checks it, and False if it is left unchecked. In our example, I will use appOffice2013, appSkype and appXMLParser.  
   
5. When you are done, click the Save As… icon in the ribbon menu and navigate to [your deployment share]\Scripts and save it with a descriptive name.  
   

#### Now for some housekeeping…

Until now we have ignored the Flow tab of the wizard designer. If you visit that tab, you will notice that is organized into Stage Groups. Each Stage Group contains one or more stages, and each stage contains the pages which will be displayed during that stage.

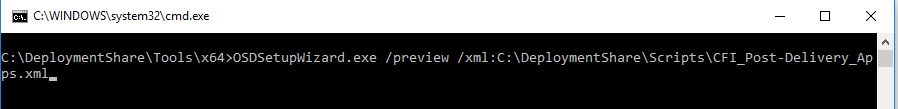
For computers coming out of the factory, the Stage will always be New Computer, so we can get rid of the other stages and groups.

1. Click on the Flow tab and then highlight StageGroup:Refresh, then click the Remove Item icon in the ribbon menu (it is on the far right by default. It looks the same as Remove Page so be careful.) Then click to highlight StageGroup:Replace and Remove Item again.  
   
2. Now expand StageGroup:NewComputer and notice that it contains two stages – Stage: NEWCOMPUTER and Stage: NEWCOMPUTER.Prestaged. Click on Stage: NEWCOMPUTER.Prestaged to highlight the whole stage and Remove Item.  
   
3. Now go into the New Computer stage, which should be the only remaining one. Highlight each stage within the stage and Remove Item. (I know, that’s a lot of clicking. Feel free to take a break and massage your index finger). When you are done you will be left with an empty New Computer stage.  
   
4. In the Page Library on the left, locate your custom page. Click and drag it into the New Computer stage. It’s a small target – that thin white line in the middle of a blue box pictured above – but I know you can do it. You wouldn’t be administering OSD unless you had mad skilz with a mouse. When you are done, it should look like this:  
   
5. Make sure to hit the Save icon in the ribbon menu. You don’t want to go through all that again…

#### A word about Preview…

You may have noticed the Preview icon in the ribbon menu.  Maybe you have already used it to admire your work. Well, unless you are using MDT 2013 Update 2, in which case you may have already been frustrated by it because you see, it is a known bug that it doesn’t work in that version. You can run it manually though:

If you browse into "C:\DeploymentShare\Tools\x64" and run the following command where [XML file] is the path to your XML configuration file it will run the preview.

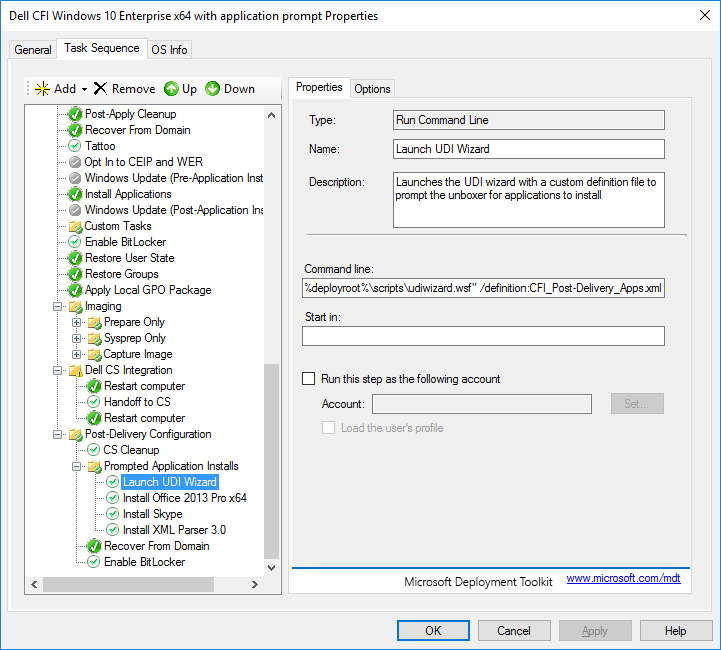
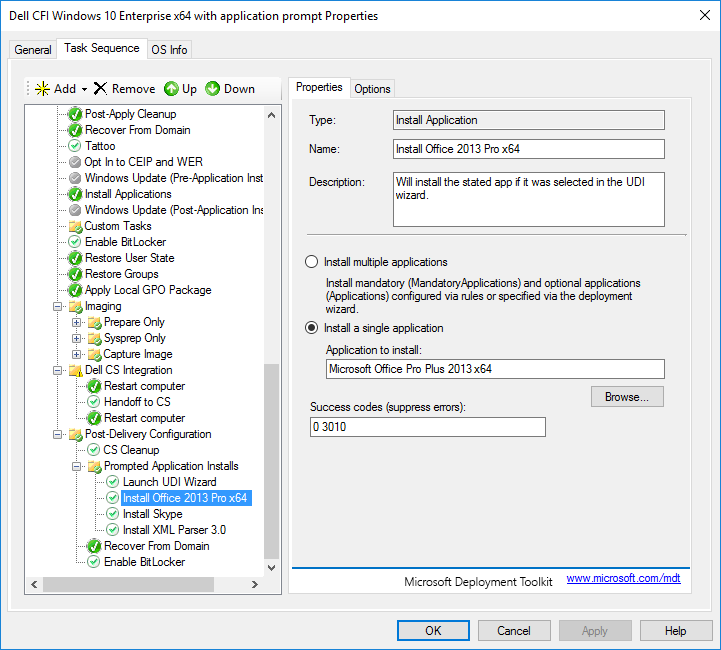
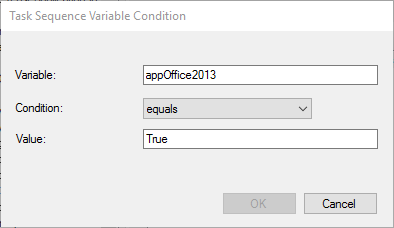
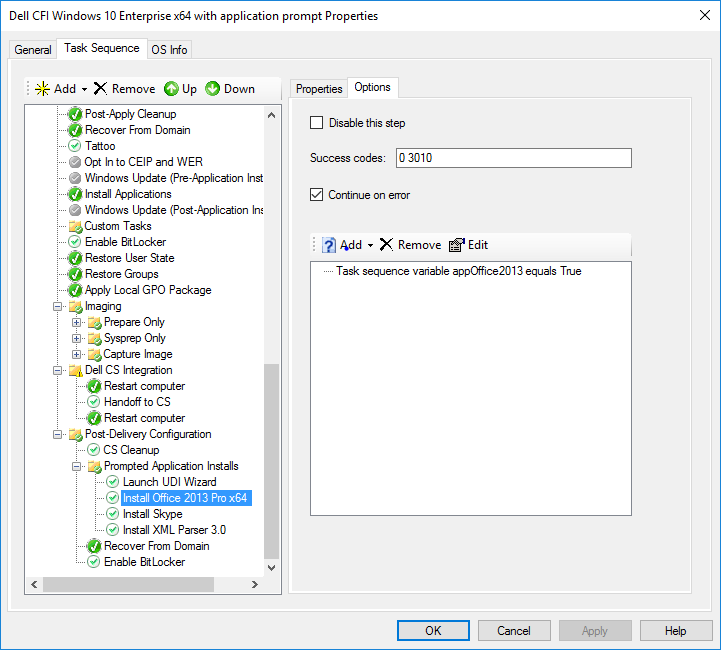
OSDSetupWizard.exe /preview /xml:[XML file]  
  


## Integrating your custom UDI definition into your task sequence

Now that you have created your custom Wizard page you will need to put it to work for you in your task sequence. Here is how we will do it:

* Make a new group under Post-Delivery Configuration to keep things organized.
* Use a Run Command Line step to call the UDI wizard with our custom definition.
* Add Install Application steps for each application conditioned on the corresponding task sequence variable.

### Adding the steps to the task sequence

1. Create a new group under Post-Delivery Configuration immediately following CS Cleanup and name it Prompted Application Installs.
2. Create a new Run Command Line step.   
   Purpose: This is the step which will present the UDI wizard to your unboxers.  
   Name: Launch UDI Wizard  
   Command Line: cscript "%deployroot%\scripts\udiwizard.wsf" /definition:[YourCustomDefinitionFile]  
   For example: cscript "%deployroot%\scripts\udiwizard.wsf" /definition:CFI\_Post-Delivery\_Apps.xml  
   
3. For each application you added to your custom UDI definition you will perform the following steps:
   1. Add an Install Application step.
   2. On the Properties tab:  
      Name and describe it appropriately  
      Choose the radio button for Install a single application  
      Browse to the application to install  
      
   3. On the Options tab:  
      Check Continue on error (Remember, we are in Post-Delivery Configuration and the CFI factory process requires everything under this group to be marked COE)  
      Add a new Task Sequence Variable Condition Variable: [variable you associated with the application in your definition] Condition: equals Value: True  
        
        
      
4. Repeat a-c for each application you added to your custom wizard definition.

That’s it! You are now set to install applications based on user selection during Post-Delivery Configuration.