

Lotusphere 2011 IBM Collaboration Solutions Development Lab

Lab # 2

Part 4: Deploying plugins using the Add-on
Installer Toolkit



Introduction

If you have completed the previous parts of this Lab, you will have now developed a Live Text Recognizer, and a Java plugin to perform the 'useful' work behind the live text. As there are a number of ways to deploy Java plugins to your clients, we are showing you 3 different methods of getting the solution installed on your users' Notes clients.

This part of the lab utilizes the Notes Add-on Installer toolkit that comes with the Notes install image.

Objective

This lab will show you the way to package your Java Update Site in a self contained install image to deploy to a Notes client.

At the end of this lab, you will have created the installer, executed it, and have a working Java plugin behind your Live Text Recognizer.

Procedure

Note: If you have completed the previous parts of this lab, you can utilize the Update Site you created at the end of Part 2. If you have not completed the previous sections of this lab, please use the Update Site found in [c:\plugin_lab2.zip](#). Simply unzip that archive in a new folder, and use the com.ibm.lotusphere.update folder that was unzipped.

Step 1: Customizing the installer

- Open Windows Explorer, and navigate to the C:\notes_852_addon_installer_toolkit folder. This toolkit is normally found in the Note client install image.
- Edit the 'AddonConfig.ini' file found inside. Update these fields:

```
MSIFileName=Lab2_Installer.msi
ProductName=Lab2 Plugin 1.0.0
ProductNameNoVersion=Lab2 Plugin
ProductVersion=1.0.0
NOLICENSE=true
LicenseDir=
LauncherName=Lab2_Setup.exe
EnableLangDlg=N
UpgradeCode=
```

Note: Clear the LicenceDir and UpgradeCode comments, so the values are empty. Please leave all of the other fields in the file (other than the ones listed here) as their default value.



Make sure to save the file <CTRL+S>

Step 2: Customizing the install manifest

- Navigate to the 'deploy' subdirectory in the toolkit folder.

Note: the default Notes Add-on Installer Toolkit does not currently contain a stub 'deploy' directory. The one included here was copied from the Expeditor version of the toolkit, which can be found [<find web link to where XPD add-on toolkit can be found>](#)

- In this directory are 3 files, install.addon.xml, plugin_customization.ini, and uninstall.addon.xml. You will need to modify the install and uninstall XML files. The plugin_customization.ini file is the standard method Java plugins use to get configuration information. In this case, since we have no preferences to pass to our plugin, we can leave it alone. Modify the install.addon.xml in this manner (**bold** text indicates the changes you need to make):

```
<domain-object name="com.ibm.rcp.installmanifest">
  <object-data>
    <install version="1.0.0">
      <installfeature id="Lab02Plugin" description="Lotusphere 2011
Lab 02 Plugin" mergeaction="add" required="true" version="1.0.0">
        <requirements>
          <feature action="install" download-size="19"
id="com.ibm.lotusphere.feature" match="perfect" shared="true" size="13"
url="" version="1.0.0"/>
        </requirements>
      </installfeature>
    </install>
  </object-data>
</domain-object>
</ibm-portal-composite>
```

Save that file, and then modify the uninstall.addon.xml in this manner:

```
<domain-object name="com.ibm.rcp.installmanifest">
  <object-data>
    <install version="1.0.0">
      <installfeature id="Lab02Plugin" description="Lotusphere 2011 Lab
02 Plugin" mergeaction="add" required="true" version="1.0.0">
        <requirements>
          <feature action="uninstall" download-size="19"
id="com.ibm.lotusphere.feature" match="perfect" shared="true" size="13"
url="" version="1.0.0"/>
        </requirements>
      </installfeature>
    </install>
  </object-data>
```



```
</domain-object>  
</ibm-portal-composite>
```

- Once those are modified, you also need to copy the plugin's update site into the toolkit folder. To do so, copy the com.ibm.lotusphere.update folder from your plugin directory into the toolkit directory, and rename it "updateSite"

Step 3: Build the installer

- Move the "CustomizeAddon.exe" and "AddonConfig.ini" files up one directory (into c:\), and launch a CMD prompt from the shortcut on the Desktop. We do this because when the tool runs, it makes a copy of the toolkit directory and MSI files to include in the completed installer, and there is no need to bundle the tool itself.
- In the CMD prompt, 'cd c:\', and execute the "CustomizeAddon.exe" The 'CustomizeAddon.exe' utility uses these three arguments: Source Directory, Output Directory, and Configuration File, where:
 - Source Directory** = the directory where the toolkit and plugin reside
 - Output Directory** = where you want the assembled installer to go
 - Configuration File** = the location of the modified AddonConfig.ini (which is in the toolkit directory). This will default to whatever directory your "CustomizeAddon.exe" binary is executed from, so if they are in the same directory, this parameter can be omitted
- Execute:

```
.\CustomizeAddon.exe .\notes_852_addon_installer_toolkit c:\Lab2_Installer
```

Note: the Output Directory parameter must be a full path, not a relative path
- Once that completes, you should have a directory [c:\Lab2_Installer](#), that looks remarkably similar to the toolkit folder. The main difference will be the presence of a "Lab2_Installer.msi" executable.

Step 4: Installing the plugin

Now that we have the plugin bundled in an executable installer, we need to tell Notes to allow the installation of an unsigned plugin. Most 3rd party plugins will actually have valid signatures provided by trusted certification authorities, but in this case, we never signed our plugin. In order to have Notes allow it to be installed, you will need to

- Edit the file:

```
c:\notes\framework\rcp\plugin_customization.ini
```
- Change the field:

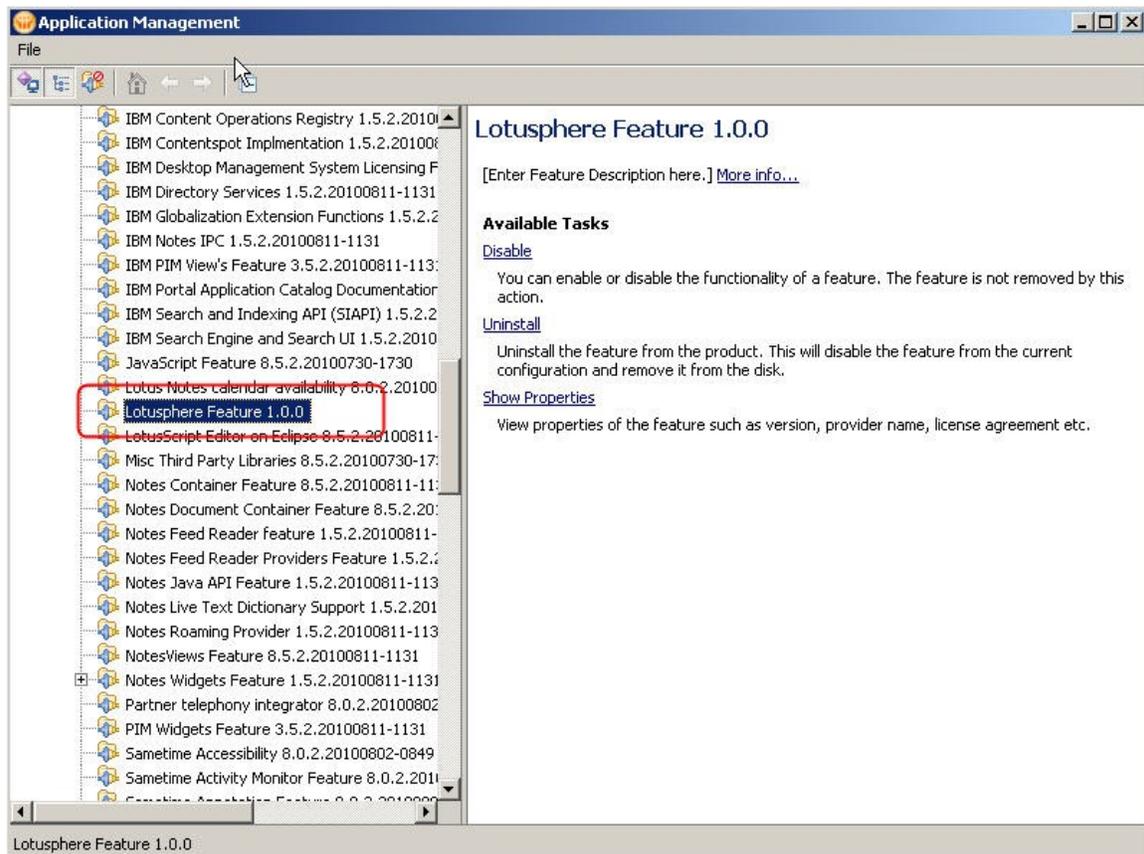
```
com.ibm.rcp.security.update/UNSIGNED_PLUGIN_POLICY=PROMPT
```

to:

com.ibm.rcp.security.update/UNSIGNED_PLUGIN_POLICY=**ALLOW**

Note: This value can be set via a Notes Desktop Settings policy configuration, as well. If a Desktop Settings policy exists, the policy will override whatever is set explicitly in the plugin_customization.ini

- While in the plugin_customization.ini, please also add the line:
com.ibm.notes.branding/enable.update.ui=true
This will allow us to see that the Feature is installed when we launch Notes
- Next, launch the setup file created by the toolkit in step 3: Lab2_Setup.exe
- Click through the prompts on the Install Shield dialogs, and after a few moments the plugin will install. Click 'Finish' when prompted, and Install Shield will exit. Once complete, launch Notes from the desktop
- After launching Notes and entering the password, launch the Application Management dialogs via 'File->Application->Application Management...'
- Expand the 'C:\notes\framework\shared\eclipse' section, and scroll down until you see '**Lotusphere Feature 1.0.0**':



Summary:

This lab has shown you how to create a self-contained installer for your Java plugins and features using the Add-on Installer Toolkit that comes with Expeditor and Notes. This powerful toolkit allows you to create an install executable for any Expeditor based client, such as Symphony, Notes, or Sametime. It is extremely flexible, and allows you to provide your own licences, supports installing, upgrading, or uninstalling, and can automatically detect locales for providing translated plugins for different languages.

References:

For more information, please review the "Creating a customized add-on installer" documentation in the IBM Lotus Domino and Notes Information Center at:

<http://publib.boulder.ibm.com/infocenter/domhelp/v8r0/index.jsp>

and the Lotus Expeditor wiki article "Creating a customized addon installer" at:

http://www.lotus.com/ldd/lewiki.nsf/dx/Creating_a_customized_addon_installer

