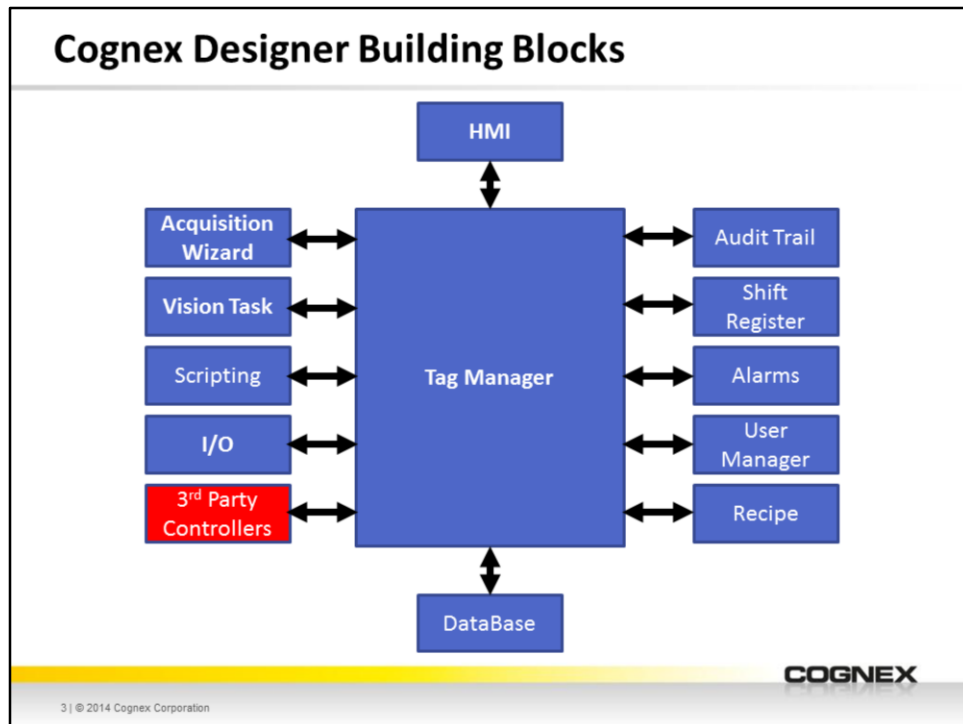# Cognex Designer Advanced – Plugins

COGNEX

Objective

Cognex Designer Advanced – Plugins

- Discover how Cognex Designer can be extended
- Look at a use of a Plugin
- Think of "what next?"

❖ Lab: Use a third party HMI interface to control Accept Threshold of PatMax

COGNEX

Cognex Designer Building Blocks

HMI

Acquisition Wizard

Vision Task

Scripting

I/O

3rd Party Controllers

Tag Manager

Audit Trail

Shift Register

Alarms

User Manager

Recipe

DataBase

COGNEX

3 | © 2014 Cognex Corporation

In this section, we will discuss a little bit about controls. For this first release of Cognex Designer, the engineers are not focusing on users creating controls. We want to make sure you know it is there.

Cognex Designer allows one to extend its functionality by allowing the user to implement various kinds of plugins. Plugins can be used to implement new devices, function blocks, global functions or HMI components.

Free Suite of WPF Controls

Available on the Internet
- http://dashboarding.codeplex.com/

- **Silverlight and WPF Dashboard**

CodePlex Project Hosting for Open Source Software

Silverlight and WPF Dashboards and gauge

download

CURRENT   Release 008
DATE   Sun Jul 12, 2009 at 3:00 AM
STATUS   Stable
DOWNLOADS   15,783
RATING   ★ ★ ★ ★ ½   5 ratings
            Review this release

COGNEX

5 | © 2014 Cognex Corporation

Library of various gauges that use the WPF

SilverLight is just an environment that people use to create the visual for the controls

Project Description
This project is a collection of dashboard gauges implemented in Silverlight 2.0 and Windows Presentation Foundation. There are upwards of nineteen dials, gauges and displays in the library
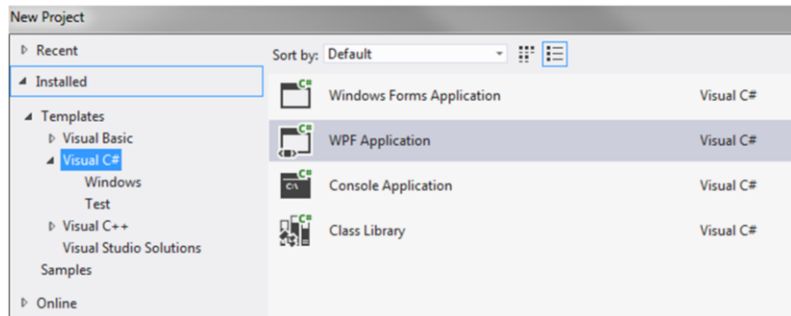
These are the files that are included in the ZIP file downloaded from the website.

By running the interactive demo, we can see all the different controls that we can add to our application. There various gauges, sliders, and LEDs. We will use the knob in the upper right corner.
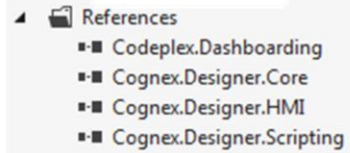
Use Visual Studio 2012 or later to create a WPF Application in C#. Any programming language could work. We are working in C# as all scripting in the Cognex Designer must be in C#

> **Note:** Make sure you are running Visual Studio as Administrator so that the new control .dlls will be put the project directory of Cognex Designer.

Here are the DLLs that need to be added to the VS project to make things work.

Coding for Knob360.xaml

Bring in Namespaces

```
using Cognex.Designer.Core;
using Cognex.Designer.HMI.Controls;
```

Define Control

```
namespace Cognex.Designer.ExampleControls
{

    [DisplayName("Knob")]
    [Category("WPF Controls")]
    public partial class Knob360 : Element
    {

        public Knob360()
        {
            InitializeComponent();

            knob.ValueChanged += knob_ValueChanged;
        }
```

Import in the namespaces for the Cognex Designer. The basic ones needed are Cognex.Designer.Core and Cognex.Designer.HMI.Controls

Then define the name of the control as it will appear in the HMI Toolbox ("Knob") and the category that will hold it ("WPF Controls").
The component then needs to be initialized.

Adding to Property Box

Parameters in Behavior

```
[DisplayName("Minimum")]
[Category("Behavior")]
[Editable]
[Bindable]
public double Minimum
{
    get { return knob.Minimum; }
    set { knob.Minimum = value; }
}
```

⊿ Behavior

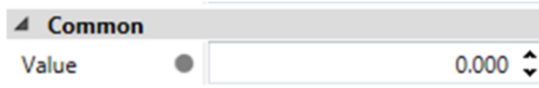|		
Click Enabled	●	☑
Enabled	●	☑
Is Bidirectional	●	☐
Maximum	●	100.000 ▲▼
Minimum	●	0.000 ▲▼

COGNEX

Add the parameters that you want the user to have access to in the Property box when the control is placed on the page. We are adding **Minimum**, **Maximum**, and **Is Bidirectional** to the property box under the **Behavior** category.

Adding to Property Box – cont.

Parameters in Value

```
[DisplayName("Value")]
[Category("Common")]
[Editable]
[Bindable]
public double Value
{
    get { return knob.Value; }
    set { knob.Value = value; }
}
```

◢ **Common**

Value	●	0.000 ⌃⌄

COGNEX

We must add a Value field to the control to be able to set or get the value.

Adjusting the Application Properties

Cognex.Designer.ExampleControls Properties...   Alt+F7

Reference Path:
- Add C:\Program Files\Cognex\Designer\2.5\

Output Path:
- C:\Program Files\Cognex\Designer\2.5\

Plugin.config
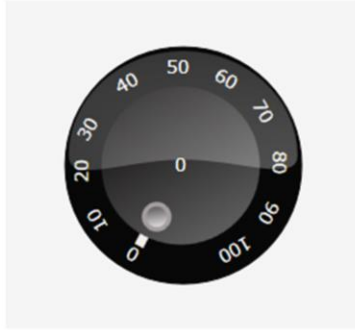- Cognex Designer needs to know how to handle the plugin

COGNEX

13 | © 2014 Cognex Corporation

The Properties of the application need to reference a few things
- Go to **Project** -> {application} **Properties** at the bottom of the pull down list
- Select **Referenced Paths** from the list on the left
  - Put in the path to the folder and select **Add Folder**
- Select **Build** and scroll down to the **Output path:**
  - Put the same path used above
  - This path has been set to a particular directory in the Cognex Common Plugin Library. This is an area that Cognex Designer scans during load time to locate additional plugins.
- Plugin.config:
  - This file is part of the Common Plugin Library behavior of Cognex Designer. It assigns attributes to this plugin such as a name and version number. It is required to be in the same directory as the plugin assemblies.
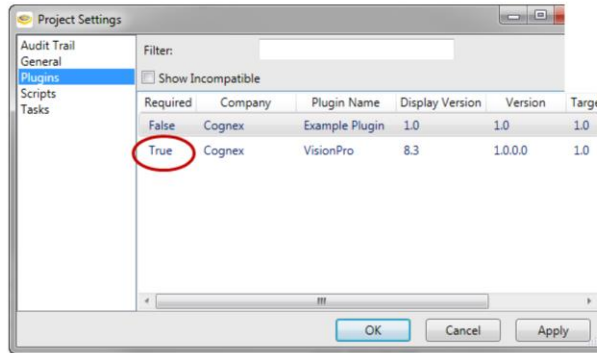
Restarting the Cognex Designer will now display the WPF Controls in the Toolbox for your HMI. Opening the WPF Control section will give access to the control that can now be dropped onto a page of the HMI.
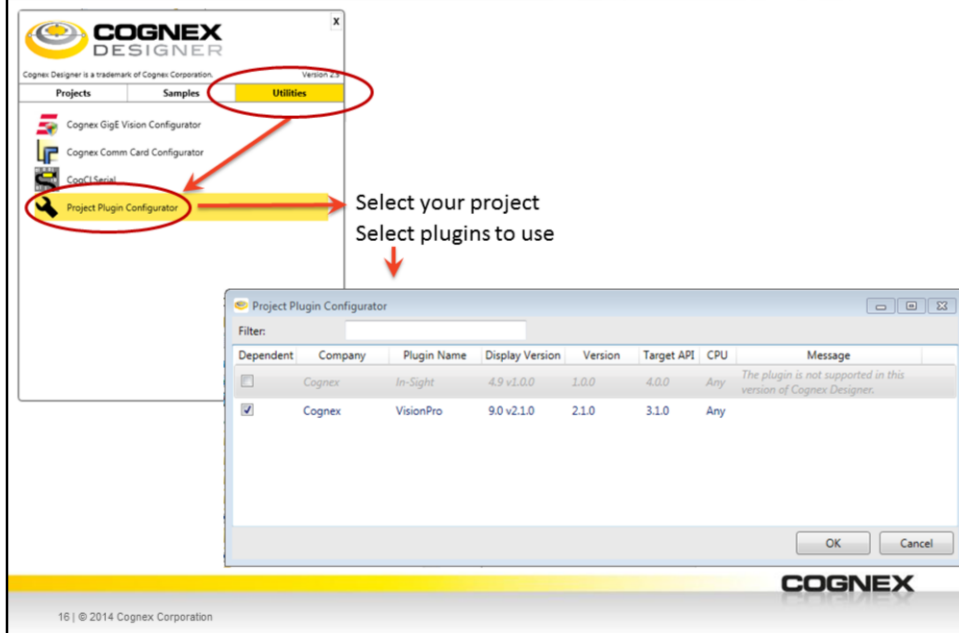
Using Plugins

VisionPro is a Plugin

◢ 🗖 **System**
   Alarms
   Recipes
   🖼 Resources
   🛠 **Settings**
   Tag Manager
   👥 Users

Project Settings

Audit Trail
General
Plugins
Scripts
Tasks

Filter:

☐ Show Incompatible

Required	Company	Plugin Name	Display Version	Version	Targe
False	Cognex	Example Plugin	1.0	1.0	1.0
True	Cognex	VisionPro	8.3	1.0.0.0	1.0

[OK] [Cancel] [Apply]

COGNEX

Go to the Explorer and select **Settings** under the **System** category.
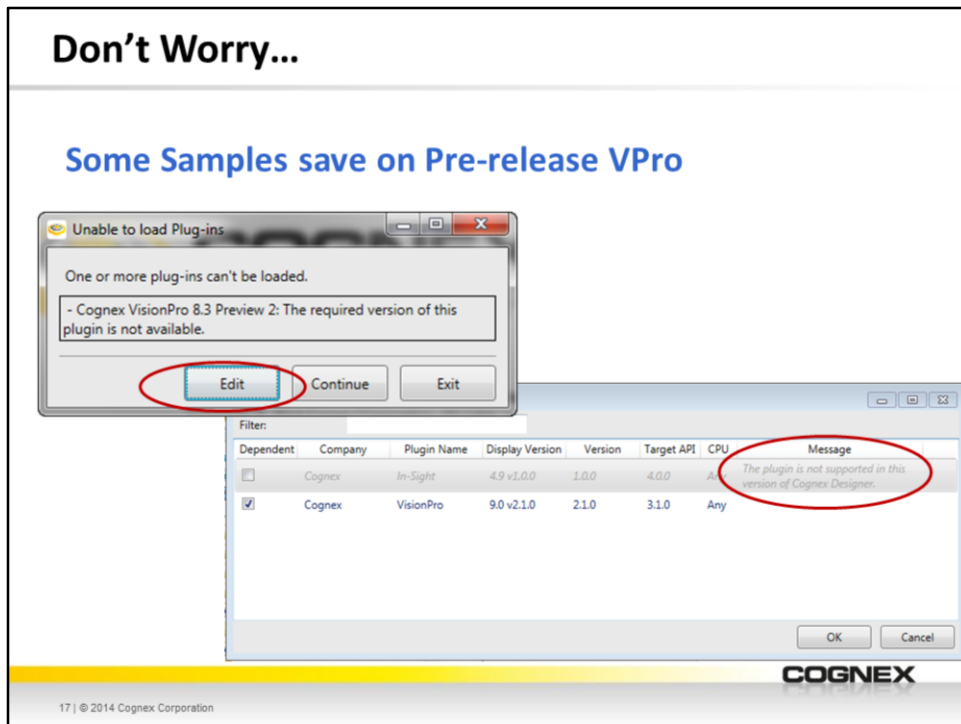Choose **Plugins** and see which plugins the system is / can use.

You can choose which Plugin is to be used by the program by selecting "Edit Project Plugins" then select your project.

The Plugin Selector Utility box will be displayed and you can select or deselect which plugins are to be used.
By default, all installed plugins are available to the application. You may have one developed for a single customer that you do not want to go out to everyone.

When you deploy, the plugin DLLs are brought over with the Deployment directory so the user has all the files needed to deploy.

Some of the samples were saved on the Preview version of the software.
This error will occur whenever a project uses a Plugin that is not currently available on the PC.

In this case, the Preview version of VisionPro was used for the project but now the released version is installed.
We need to deselect the old version and select the current version of VisionPro.
Now the application will work as expected.

Summary

Plugins

- Explored how Cognex Designer can be extended
- Learned about the use of a Plugin
- Thought about "what next?"

COGNEX