File Explorer

End user facing:

Developing

We have already done most of the development work in the PreviewHandlerCommon common project. To add a preview for the file type of .xyz:

- Add a new .NET project in the preview pane folder.
- Add a reference to the `PreviewHandlerCommon` common project.
- Create your preview handler class and extend the FileBasedPreviewHandler class. See an example below:

```csharp
using System;
using System.Runtime.InteropServices;
using Common;

namespace XYZPreviewHandler
{
    /// <summary>
    /// Implementation of preview handler for .xyz files.
    /// GUID = CLSID / CLASS ID.
    /// </summary>
    [Guid("xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxx")]
    [ClassInterface(ClassInterfaceType.None)]
    [ComVisible(true)]
    public class XYZPreviewHandler : FileBasedPreviewHandler
    {
        private XYZPreviewHandlerControl xyzPreviewHandlerControl;

        /// Call your rendering method here.
        public override void DoPreview()
        {
            this.xyzPreviewHandlerControl.DoPreview(this.FilePath);
        }

        protected override IPreviewHandlerControl CreatePreviewHandlerControl()
        {
            this.xyzPreviewHandlerControl = new xyzPreviewHandlerControl();
            return this.xyzPreviewHandlerControl;
        }
    }
}
```

Create a separate Preview Handler Control class and extend the `FormHandlerControl` Class.

```csharp
using Common;
```

```
namespace XYZPreviewHandler
{
    public class XYZPreviewHandlerControl : FormHandlerControl
    {
        public XYZPreviewHandlerControl()
        {
            // ... do your initializations here.
        }

        public override void DoPreview<T>(T dataSource)
        {
            // ... add your preview rendering code here.
        }
    }
}
```

Integrate the Preview Handler into PowerToys Settings:

Navigate to the [powerpreview](#) project and edit the `powerpreview.h` file. Add the following Settings Object instance to `m_previewHandlers` settings objects array in the constructor initialization:

```
// XYZ Preview Handler Settings Object.
FileExplorerPreviewSettings(
    false,
    L"<--YOUR_TOGGLE_CONTROL_ID-->",
    L"<--A description of your preview handler-->",
    L"xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxx", // your preview handler CLSID.
    L"<--A display name for your preview handler-->")
```

Installation

MSI (Recommended)

To add a new Previewer update the `Product.wxs` file in `PowerToysSetup` similar to existing Previewer to register the Preview Handler. More details about registration of Preview Handlers can be [found here.](#)

```
<Component Id="Module_PowerPreview" Guid="FF1700D5-1B07-4E07-9A62-4D206645EEA9"
Win64="yes">
        <!-- Files to include dll's for new Previewer and it's dependencies -->
        <File Source="$(var.BinX64Dir)\modules\XYZPreviewer.dll" />
        <File Source="$(var.BinX64Dir)\modules\Dependency.dll" />
    </Component>
    <Component Id="Module_PowerPreview_PerUserRegistry" Guid="CD90ADC0-7CD5-4A62-
B0AF-23545C1E6DD3" Win64="yes">
        <!-- Added a separate component for Per-User registry changes -->
        <!-- Registry Key for Class Registration of new Preview Handler -->
        <RegistryKey Root="HKCU" Key="Software\Classes\CLSID\{ddee2b8a-6807-48a6-
bb20-2338174ff779}">
            <RegistryValue Type="string" Value="XYZPreviewHandler.XYZPreviewHandler"
/>
```

```xml
          <RegistryValue Type="string" Name="DisplayName" Value="XYZ Preview
Handler" />
          <RegistryValue Type="string" Name="AppID" Value="{CF142243-F059-45AF-8842-
DBBE9783DB14}" />
          <RegistryValue Type="string" Key="Implemented Categories\{62C8FE65-4EBB-
45e7-B440-6E39B2CDBF29}" Value=""/>
          <RegistryValue Type="string" Key="InprocServer32" Value="mscoree.dll" />
          <RegistryValue Type="string" Key="InprocServer32" Name="Assembly"
Value="SvgPreviewHandler, Version=$(var.Version).0, Culture=neutral" />
          <RegistryValue Type="string" Key="InprocServer32" Name="Class"
Value="XYZPreviewHandler.XYZPreviewHandler" />
          <RegistryValue Type="string" Key="InprocServer32" Name="RuntimeVersion"
Value="v4.0.30319" />
          <RegistryValue Type="string" Key="InprocServer32" Name="ThreadingModel"
Value="Both" />
          <RegistryValue Type="string" Key="InprocServer32" Name="CodeBase"
Value="file:///[ModulesInstallFolder]XYZPreviewHandler.dll" />
          <RegistryValue Type="string" Key="InprocServer32\$(var.Version).0"
Name="Assembly" Value="XYZPreviewHandler, Version=$(var.Version).0, Culture=neutral"
/>
          <RegistryValue Type="string" Key="InprocServer32\$(var.Version).0"
Name="Class" Value="XYZPreviewHandler.XYZPreviewHandler" />
          <RegistryValue Type="string" Key="InprocServer32\$(var.Version).0"
Name="RuntimeVersion" Value="v4.0.30319" />
          <RegistryValue Type="string" Key="InprocServer32\$(var.Version).0"
Name="CodeBase" Value="file:///[ModulesInstallFolder]XYZPreviewer.dll" />
        </RegistryKey>
        <!-- Add new previewer to preview handlers list -->
        <RegistryKey Root="HKCU"
Key="Software\Microsoft\Windows\CurrentVersion\PreviewHandlers">
          <RegistryValue Type="string" Name="{Clsid-Guid}" Value="Name of the
Previewer" />
        </RegistryKey>
        <!-- Add file type association for the new Previewer -->
        <RegistryKey Root="HKCU" Key="Software\Classes\.xyz\shellex">
          <RegistryValue Type="string" Key="{8895b1c6-b41f-4c1c-a562-0d564250836f}"
Value="{Clsid-Guid}" />
        </RegistryKey>
      </Component>
```

Directly registering/unregistering DLL's

[Important] This method of registering Preview Handler DLL's is not recommended. It could lead to registry corruption.

Registering Preview Handler

1. Restart Visual studio as administrator.
2. Sign `XYZPreviewHandler` and it's dependencies. To sign an assembly in VS, follow steps given [here](#).
3. Build `XYZPreviewHandler` project.
4. Open developer command prompt from `Tools > Command Line > Developer Command Prompt` .

5. Run following command for each nuget and project dependency to add them to Global Assembly Cache(GAC).

```
gacutil -i <path to dependency>
```

6. Run following commands to register preview handler.

```
cd C:\Windows\Microsoft.NET\Framework64\4.0.x.x
gacutil -i <path to XYZPreviewHandler.dll>
RegAsm.exe /codebase <path to XYZPreviewHandler.dll>
```

7. Restart Windows Explorer process.

Unregistering Preview Handler

1. Run following commands in elevated developer command prompt to unregister preview handler.

```
cd C:\Windows\Microsoft.NET\Framework64\4.0.x.x
RegAsm.exe /unregister <path to XYZPreviewHandler.dll>
gacutil -u XYZPreviewHandler
```

Debugging

Since in-process preview handlers run under a surrogate hosting process (prevhost.exe by default), to debug a preview handler, you need to attach the debugger to the host process.

1. Click on a file with registered extension to start host process.
2. Attach debugger in Visual studio from `Debug->Attach to Process` and select `prevhost.exe` with type `Managed(version), x64` .

Managing Preview Handlers

After successful integration, your preview handler should appear in the PowerToys settings UI under the `File Explorer Preview` Tab. In here you should be able to enable and disable all the preview handles.

Settings UI - File Explorer Preview Tab