Release Notes VSPAT 1.2.19.0

This major release is a rollup release of several interim internal releases: 1.2.16.0, 12.17.0, 1.2.18.0 and 1.2.19.0.

It contains major breaking changes that require migration for existing toolkits extensions, and existing toolkit projects.

**IMPORTANT**: If migrating from a previous version of VSPAT (i.e. a version prior to 1.2.19.0), please ensure you read and follow the [‘Migration Notes’](#_Migration_Notes:) section at the end of this document.

# What’s New?

* This version of VSPAT, and Hands-On Labs are now publically available from the Visual Studio Gallery.
* This version of VSPAT now has a far superior upgrade story for pattern toolkit authors and users, significantly minimizing the migration steps required between new versions of VSPAT.
  + Note however that there are many complex migration steps required to migrate up to this version. Subsequent version updates will be far simpler as a result.
* Fixed the error with opening a ‘Pattern Toolkit’ Solution from a desktop shortcut.
  + This long standing error has in the past prevented loading solutions from shortcuts on the desktop, requiring the solution to be open and closed once Visual Studio starts.
* Fixed many issues in generate code to integrate better with static code analysis and code coverage tools.
* Significant improvements in performance of Solution Builder, particularly with large numbers of elements.
* Ability to define the ordering of elements and menus in solution builder.
* Many bug fixes.

## New Features:

* As an Author, I want to be able to define ordering of element instances in solution builder.
  + This new feature now allows you to predefine grouping and ordering of element instances for display in the Solution Builder window.
* As an Author/Tailor I want to have a command on the Pattern Model DSL that auto-layout's my diagram in an assumed hierarchical layout.
  + You can now right-click anywhere on the ‘Pattern Model Designer’, and have your shapes ordered hierarchically automatically.
* As an Author, I want to have the ability to suppress the 'Add' menu for certain elements in my pattern.
  + For any Collection and Element in the ‘Pattern Model Designer’ you can now specify if you want to permit whether the user has an ‘Add’ context menu displayed to create new instances.
* As a toolkit author, want to download and install VSPAT from Visual Studio Extension Manager
  + The VSPAT ‘Pattern Toolkit Builder’, and ‘Pattern Toolkit Builder Hands-On Labs’, are now publically available for download on the Visual Studio Gallery.
* As a user I would like to have a prompt confirming a delete in solution builder, in case I select Delete from the menu or hit the delete button by accident.
  + Whenever a user deletes an instance of a solution element in Solution Builder, (with either DEL key, or ‘Delete’ context menu) they will be prompted to confirm the deletion.
* As an author I want better help on implementing drag and drop for files from windows explorer.
  + There are now several topics in the guidance to help implement drag and drop.
* As an Author, I want better support for displaying elements of my pattern model
  + There have been many improvements to how data in the Pattern Model, and data in Solution Builder is displayed and manipulated, including: simpler data types, better descriptions and tooltips, less stringent rules for solution element names, and various other minor UI display enhancements.
* As an Author, I want better support for menus
  + There have been many improvements to how authors can control the display of their custom menus, including: being able to define the sort order relative to other menus, supporting accelerator keys, and displaying the correct captions on the ‘Add’ menus.
* As an Author, I want to be able to configure the BuildAction and CopyToOutput properties on generated files (options on the GenerateCodeCommand)
  + Authors can now define the BuildAction and CopyToOutput settings for generated files.
* As an Author I want to be able to tag a generated file for filtering with other automation
  + Authors can now assign an arbitrary ‘Tag’ to the artifact link of an unfolded or generated file, for processing and filtering with automation.
* As an Author, I want better support for code generation and SyncNames on generated files.
  + There have been many improvements to how generated files are regenerated and artifacts links fixed, when files have either been manually renamed or removed, that improve the user’s experience and cause far less disruption.
* As an Author, I want to upgrade to minor releases of VSPAT without having to migrate my toolkit source.
  + There have been many improvements in many areas to significantly alleviate (and in some cases eliminate) the need to perform manual migration steps when upgrading to newer version of VSPAT. Although, this cannot be realized until a toolkit is built on this version of VSPAT (1.2.19.0).
* As an Author, I want to have a toolkit that has reasonable performance on large models
  + The performance of Solution Builder has been improved in several key areas when displaying hundreds of solution element instances.
* As an Author, I want to have a better typed interface generated for my toolkit.
  + There are a number of new methods and many improvements made to the runtime API that improve converting between the generic and typed interface layers. Several bugs have been fixed, and the API is now improved to help make mocking and unit testing of automation easier. Generated code can now be omitted from static code analysis, and code coverage tools.
* As an Author, I want to have a command that can open an element in the solution, specified by path
  + The ActivateSolutionElementCommand is now provided to open an arbitrary file in ‘Solution Explorer’ based upon a contextualized TargetPath and a TargetFileName.

## Bugs Fixed

There have been many bugs fixed in this rollup release, and many were fixed as part of the new features above. Only the titles of the bugs are recorded here for completeness.

* Various bindings in the pattern model schema are not being serialized to Json.
* RegistrySetValue build task is failing on newly installed machine.
* Nodes in Solution Builder cannot be drag & dropped anywhere
* ExtensionTo property is broken for tailored toolkits
* ObjectValidator.ThrowIfInvalid(this) is not being called on Command.Execute().
* Replace ObjectValidator with DataAnnotations
* DOCBUG: Need sample in guidance for Checkout and CheckIn of files from commands.
* DOCBUG HOL: should have a step in guidance to tell authors to enable extensions in VSExp
* DOCBUG: Topic "Common Automation Tasks" needs to be referenced from other topics
* DOCBUG: VisualT4 does not set ContentType and IncludeVSIX back to original values if T4 opened in editor
* ObjectValidator - replace with .NET framework version
* Command Validator rule is implemented wrong
* Runtime interface incomplete - missing implementation
* typed interface has no methods for CreateExtension()
* MVC App Toolkit not realistic enough
* MVC App Toolkit Search Form needs to generate UI for the search criteria and search columns
* MVC Search Form implementation is not realistic
* Cardinality validation rules (resource strings) need revising!
* Feature Extension Runtime VSIX not updated in ExpHive after installing PatternToolkitBuilder, and authoring a new or existing toolkit
* Authoring VSIX not dependent on VSSDK
* UInt64 is not CLSCompliant, so need to apply the CLSCompliant[false] attribute in all generated code of typed interface.
* DOCBUG: Need to update authoring/runtime guidance with new MVC screenshots
* DOCBUG: KnownIssue for VSSp1 and missing VSSDK SP1
* DOCBUG HOL: Modal dialog (AddNewSolutionElement) in one step prevents reading next steps in guidance.
* DOC BUG HOL: "Configure the Command" topic in HOL guidance needs updated screenshot of Command properties
* DOCBUG: SortOrder on MLP needs documenting
* DOCBUG HOL: Add a note when creating project template how to change and control the category.
* DOCBUG: Topic on Command Design & Implementation
* DOCBUG: Guidance topic for doing Drag&Drop using built in base classes for files, and solution items.
* DOCBUG: No guidance for building VSPAT toolkit on build server
* DOCBUG: Guidance for packaging multiple toolkits is incorrect
* VSPAT Toolkit has no EULA
* Fix CreateXXXX in MVCAppToolkit (CreateDefaultInstances.cs)
* Update VSPAT 1.2.19.0 to FBR 1.2
* Update from 1.2.18.0 to 1.2.19.0
* An element's name in solution builder should allow a single character instance name.
* Need a validation rule for both IOnInstantiate and IOnPropertyChangeEvent to make sure the user choses 'Current Element Only' = true. Otherwise performance suffers.
* Cannot compile generated code for typed interface if there are elements with the CodeIdentifier 'View', 'Product', 'Element', 'Collection', 'ExtensionPoint' etc.
* CurrentElementOnly property on Event Launch Point should default to True
* When tailoring a toolkit, the documentation assets are not copied.
* Suggest that Current Element Only property should default to True on launch points
* \*.productstore file takes too long to save. (Validation takes too long to run)
* PLATU VSIX does not contain a license agreement and does not have a digital signature.
* Prompt user to delete artifacts when product/element/collection is deleted.
* A Customized Factory does not have cloned diagram files.
* Opening Pattern Toolkit project from shortcut fails, and Solution Builder is ruined until VS Restart
* Opening a Toolkit Solution throws error
* Descriptions for validation rules need refinement
* Creating a New Pattern Toolkit project quits after toolkit element is created
* Fix issues with Delete logic in solution builder, and don’t allow delete on cardinalities if cannot re-add
* New element/collection/extension points have duplicate names across the model
* Setting active guidance in FBR no longer selects the guidance instance
* proxy.GetExtension in typed interface is always returning null when extension instances are created.
* Extension Point instances always empty
* INVESTIGATE: x64 Word Automation for building guidance
* When building a toolkit project, ensure FBR Runtime and VSPAT Runtime VSIXes are copied to Exp instance
* Runtime installer fails on Windows XP because runtime guidance file paths are too long
* Display name of View in SolutionBuilder should use DisplayName property
* Property validation rule should have different template than element validation rule
* Replace ObjectValidator with DataAnnotations.Validator
* Toolkit Documentation and requirements documents add no value and should be removed from toolkit
* DragDropSettings is missing validations for conditions and other settings
* Authoring project template contains version numbers for WizardExtensions
* Drag and dropped element filenames are lower cased
* InstanceName in AddNewElement dialog validation rule needs to be updated to accommodate InstanceName validation rules
* Validation Rule for InstanceName does not need to be so stringent. Can contain any printable characters, and can be 1 char long
* The 'Add New Element' dialog should tailor its dialog title to state the element type being created.
* Add support for cancelling command execution and element initialization by throwing System.OperationCancelledException
* Exception in the template wizard associated with a template, causes VSPAT to end up in an unusable state
* Garbage collected WPF UI elements cause exceptions on property change manager
* A value provider should be able to resolve a path for a specific item kind
* There is no way to cancel deletion of an element (without hanging VS)
* ProductStore does not expose Saved event, which is exposed at the ProductManager level
* Variable property Reset logic does not take into account the value provider that may be configured for DefaultValue
* ProductManager hooks too late to store events (after the Opened event is raised)
* Runtime store elements should expose Deleting/Deleted events for the interface layer to hook directly, just like INotifyPropertyChanged
* Dynamic binding context not available to bound components
* ValueEditor in wizards do not pass the type descriptor context to type converters and editors
* Extra page in printed guidance topics from IE
* Solution Builder fast scrolling hangs
* Update Json.NET library to fix VerificationException
* SolutionUriProvider traverses entire solution looking for item Uris when it already has the product id/instance
* When author configures TargetPath with ~ on CodeGenCommand, then potential for runtime errors when file is deleted and regenerated
* When SyncName=true on T4 Generate Command it should rename the generated file on next generation.
* Bound propeties should display more helpful text when blank or assigned
* MenuAutomation does not display underscores as accelerator keys
* Double-click on Property.Type sets the wrong Datatype.
* When selecting a value type such as GUID or Float, can't nullify the default value if needing to specify a DefaultValue.ValueProvider or Property.ValueProvider
* Need a 'Reset' on the VP of a default value, and the default value as a whole
* Authoring Item template for a project template and item template are broken
* Add rules forcedly create the interface layer, which may already exist from previous automation that created it
* When accessing the interface layer via enumeration of child elements, cached interface layer is not used
* When creating elements via the typed interface, child element interface layer is not cached
* View DefinitionName is shown between brackets after product instance name, rather than its DisplayName
* View sort order should be alphabetically from DisplayName
* View DefinitionName is show in the context menu, rather than the Display Name
* Views context menu should honor the IsVisible property specified by the author for the view
* Change namespace of all DSL's, i.e. patterndefinition file, patternstate, workflow etc.
* Change Writing file on 'SolutionExists' no 'NoSolution'
* Commands cannot gracefully cancel execution of an aggregator command
* After showing dialog for duplicate element name, Visual Studio cannot be closed
* Traversing the tree with the keyboard causes double focus per tree node
* Toolkit interface layer implementation members are not virtual - making it hard to test/mock them.
* Add support for ISupportInitialize interface in interface layers
* .As<T>() returns a transient instance that is different every time, need to cache this value.
* Add partial method to hook the oncreated event of a typed element
* .As<T>() should try .NET (as) before trying ToolkitInterface.As<T>() to aid testing.
* When using .As<>() to convert from generic to typed interface, always succeeds no matter what factory type you give it.
* ElementPropertyValueProvider does not consider interface layer extensions
* IProductElement.As<T> method returns non-null for any interface
* Typed interface does not have methods and properties for common things
* Add AsElement() method to IToolkitInterface
* IProductElement.As<T> method is an extension method which makes mocking it very hard.
* ActivateOrInstantiate command needs moving from authoring to library
* Datatypes are too technical - require a TypeConverter to display them sensibly
* Custom event template needs revising
* Toolkit interface proxies are not exported anymore, causing the imports for them to fail
* Property Changed event has disappeared from the available events
* BindingCompositionService improperly processes shared parts
* Make WizardWindow methods virtual
* ExpressionValueProvider does not take into account toolkit interface extensions
* Setting the PropertyRequiredValidationRule to a custom error message has no effect
* Turn condition evaluation result into Information, not Warning
* Refresh Enabled and Visible status on dynamic automation menus after a QueryStatus refresh
* Wrong error message on query status failed for menu automation
* Typed Interface code is not marked as generated code. Add the attributes and heading
* VERSIONING: Updating versions of code causes migration unnecessary problems
* Resource string includes missing parameter
* Missing a ValueProvider for a DateTime field. CurrentDateTimeValueProvider
* This validation rule is failing now, probably because of the fact the WizardExtension has no version number in it anymore
* Performance: unfolding project templates incrementally takes too long the more pattern instances there are.
* WPFToolkit quickstart does not have guidance configured correctly for solution
* Validation Rules not showing up in the list of types
* Validation Rule properties not allowing configuration of Value. (only ValueProvider)
* Extension Point typed interface does not return extension point instance
* InstantiationTemplateWizard fails to unfold an item template for a product.
* Instantiate event is not 'sent by' the element, when the element is created by a command in the instantiate event of a parent.
* XAML resources not found at runtime issue
* DOCBUG: Need a topic on how to add another automation library to a toolkit.
* DOCBUG: Remove the tutorial from the authoring guidance
* DOCBUG: Need a guidance topic on how to walk your typed model
* DOCBUG: Missing vital content in automation guidance
* DOCBUG: Need a references section on all the built-in automation types
* DOCBUG: HowTo "Which Automation To Use" needs to link another topic
* DOCBUG: Need a section on known issues
* Title of Image Picker incorrect.
* Visual T4 support
* SyncName is off by default, even though =True by default.
* Missing XAML resources error - demystified
* DOCBUG: Reset experimental
* XAML resource errors
* Deleting a file from Solution Explorer closes Solution Builder!!! (REOPENED)
* When template unfold fails for whatever reason, VS crashes due to format exception of the error in the Shield.
* SoftwareFactories.Common is referenced in AutomationLibrary project but is not deployed.
* When adding an element to a collection the Multiplicity should be automatically set to Multiple.
* Require ValueProvider in PLATU Library that provides the value of a property of the current element for easy configuration of commands/conditions/etc.
* It’s not clear what kind of automation I have configured for my element just from the names. Suggest adding the type name also to the compartment text.
* Command Extensions don’t have any validation rules, enforcing correct configuration.
* Event Launch Point Extensions don’t have any validation rules, enforcing correct configuration.
* Template Launch Point Extensions don’t have any validation rules, enforcing correct configuration.

# Migration Notes

These actions need to be followed in order to migrate existing pattern toolkits to the latest version of VSPAT, from any version from **v.1.2.15.1** to **v.1.2.19.0.**

**WARNING**: The VSPAT extensions are now publically [available online](http://visualstudiogallery.msdn.microsoft.com/332f060b-2352-41c9-b8dc-95d8ad21329b) at the [Visual Studio Gallery](http://visualstudiogallery.msdn.microsoft.com/), and existing toolkit users with existing installed pattern toolkits (built against previous versions of VSPAT) are free to download and install the latest version of VSPAT for creating their own pattern toolkits. Should they either install the latest version of VSPAT ( the ‘Pattern Toolkit Builder’ extension) or install newer pattern toolkits, then their existing pattern toolkits solution (built with a previous version of VSPAT) will no longer work for them. We strongly recommend upgrading the existing pattern toolkits.

## Who Needs to Migrate?

You need to migrate existing pattern toolkits in either of the following scenarios:

1. As a user, you are using a pattern toolkit (built against a previous version VSPAT), and wish to be upgraded to the latest version of the pattern toolkit on your machine.
   * This upgrade will require your to manually:
     1. Obtain a newer version of your pattern toolkit from the orginal author, who has already upgraded it to the latest version of VSPAT (see next scenario for details).
     2. Uninstall your existing pattern toolkit
     3. Uninstall all VSPAT related extensions (i.e. the ‘Pattern Toolkit Runtime’ extension)
     4. Manually install the latest version of the pattern toolkit.
   * Note: This kind of upgrade will not happen automatically for a user.
2. As an author, you are currently developing an existing pattern toolkit (built against a previous version VSPAT), and wish to upgrade it to the latest version of VSPAT.
   * This upgrade will require you to manually:
     1. Uninstall the current version of VSPAT (i.e. the ‘Pattern Toolkit Builder’, ‘Pattern Toolkit Library Suppport’, and ‘Pattern Toolkit Runtime’ extensions)
     2. Install the latest version of ‘Pattern Toolkit Builder’ extension.
     3. Migrate the pattern toolkit to the latest version of VSPAT.
   * Note: this kind of upgrade will not happen automatically for an author.

**Important**: Before starting the migration , it you are strongly recommend to backup your source code, or use source control, to manage any changes you make in migrating your toolkits projects.

### Pattern Toolkit Solutions:

#### Solution file (\*.sln)

* + Open in ‘XML View’:
  + Modify the following:

GlobalSection(ExtensibilityGlobals) = postSolution

Features = 9f6dc301-6f66-4d21-9f9c-b37412b162f6:Creating Pattern Toolkits:1.2.19.0

EndGlobalSection

* + When loading the solution you should be prompted to update the version number of the pattern state file. Accept this offer.

#### Solution.patternstate file (\*.patternstate file)

* + Rename file to: \*.slnbldr
  + Close solution first, and Open in ‘XML View’:
  + Update the root node, version number and default xml namespace

<productState dslVersion="1.2.0.0"xmlns="http://schemas.microsoft.com/visualstudio/patterning/runtime/productstate">

Update all:

    <product Id="1b259e36-d2a5-4c1c-a3ed-3e45f81b4eec" definitionId="c034429e-01f9-48dd-a478-0321fb708dd3" definitionName="PatternToolkit" instanceName="MVCAppToolkit" extensionId="9f6dc301-6f66-4d21-9f9c-b37412b162f6" extensionName="Pattern Toolkit Builder" author="Microsoft" version="1.2.18.0">

Then, open the solution.

### Pattern Toolkit Project:

#### Toolkit Project File (csproj):

* + Unload project, and edit the XML:
  + Replace All:

$(LocalAppData)\Microsoft\VisualStudio\10.0\Extensions\Microsoft\Pattern Toolkit Runtime\1.2.15.1

* + With:

$(PatternToolkitRuntime)

* + Replace All:

$(LocalAppData)\Microsoft\VisualStudio\10.0\Extensions\Microsoft\Feature Extension Runtime\1.0

* + With:

$(FeatureExtensionRuntime)

* + Replace All:

$(LocalAppData)\Microsoft\VisualStudio\10.0\Extensions\Microsoft\Pattern Toolkit Builder\1.2.15.1

* + With:

$(PatternToolkitBuilder)

* + Replace:

  <Import Project="$(PatternToolkitBuilder)\Microsoft.VisualStudio.Patterning.Authoring.Toolkit.targets" />

* + With:

<Import Project="$(localappdata)\Microsoft\MSBuild\Microsoft\Pattern Toolkit Builder\Microsoft.VisualStudio.Patterning.Authoring.PatternToolkitVersion.targets" />

* + Replace All:

PatternToolkitRuntime.vsix

* + With

PatternToolkitManager.vsix

* + Replace:

    <Content Include="$(PatternToolkitRuntime)\GeneratedCode\SchemaModelSchema.xsd">

      <Visible>true</Visible>

      <Link>Properties\Schemas\SchemaModelSchema.xsd</Link>

      <FixedLink>

      </FixedLink>

    </Content>

* + With

    <Content Include="$(PatternToolkitRuntime)\GeneratedCode\PatternModelSchema.xsd">

      <Visible>true</Visible>

      <Link>Properties\Schemas\PatternModelSchema.xsd</Link>

      <FixedLink>

      </FixedLink>

    </Content>

Reload the project

#### Pattern Model File (\*.patterndefinition):

* + Open in ‘XML View’:
  + Update the root node, version number, and default xml namespace

<patternModel dslVersion="1.2.0.0" xmlns="<http://schemas.microsoft.com/visualstudio/patterning/runtime/schemamodel>"

* + Replace:

xmlns:dm1="http://schemas.microsoft.com/dsltools/Library"

* + With:

xmlns:dm1="http://schemas.microsoft.com/visualstudio/patterning/library/automationextensions"

* + Replace:

productLink="productmanager://

* + With:

patternLink="patternmanager://

* + - Replace:

<product

* + - With:

<pattern

* + - Replace All:

<productSchemaMoniker

* + - With:

<patternSchemaMoniker

* + Remove all 'Current Element Only' properties from the XML.
    - Find all <eventSettings elements
    - Remove the filterForCurrentElement="true" attribute
    - Delete the child <conditions element (if any).
  + For any elements that contain CodeGeneration Commands (but not “VS Template Launch Points” or TLP) remove the ‘SyncArtifactName’ event and the ‘SyncArtifactName’ command on each element in the pattern model.
    - To show these vents and commands, select the pattern model canvas and set the ‘Show Hidden Items’ property to “true”.
  + For any pattern model elements that contain “VS Template Launch Points”, reconfigure the 'Target Filename' and ‘Target Path’ Properties.
    - Use your source control system to find out the orginal XML encoded value of these properties, then use an XML Decoder such as <http://coderstoolbox.net/string/> to determine the values you had set previously.
  + For any pattern model elements that contain “DragDrop Launch Points” or “Menu Launch Points”, reconfigure the 'Conditions' Properties.
    - Use your source control system to find out the orginal XML encoded value of these properties, then use an XML Decoder such as <http://coderstoolbox.net/string/> to determine the values you had set previously.
  + For any pattern model elements that contain "Drag Drop Launch Points", reconfigure the 'Conditions' property.
    - Use your source control system to find out the orginal XML encoded value of these properties, then use an XML Decoder such as <http://coderstoolbox.net/string/> to determine the values you had set previously.
  + For any pattern model elements that contain “CodeGenerations” commands, reselect the T4 file for the 'Text Template' Property
  + Update the namespace of (by re-picking) the following Commands, Conditions, & ValueProviders.
    - PropertyStringValueForbiddenCharsValidationRule
    - CreateArtifactLinkCommand
    - CreateElementFromDroppedSolutionItemCommand
    - DropFileCondition
    - DropItemFormatCondition
    - DropSolutionItemCondition
    - GenericDroppedItemValueProvider
  + Select “Validate All” in the designer, and upgrade any old unsupported data types

#### Pattern Model Diagram Files (\*.patterndefinition.diagram):

* + Open in ‘XML View’:
    - Update root node, version number

<patternModelDiagram dslVersion="1.2.0.0" type="Microsoft.VisualStudio.Patterning.Runtime.Schema.PatternModelSchemaDiagram"

* + - Replace:

<productShape

* + - With:

<patternShape

* + - Replace All:

<productLineMoniker

* + - With:

<patternModelSchemaMoniker

* + - Replace All:

<productSchemaMoniker

* + - With:

<patternSchemaMoniker

Open **patternModel.patterndefintion**, it should load without errors. Right-click anywhere on the canvas and select “Auto Arrange Shapes” to replace all missing connectors

#### Tooling Design Model (\*.toolingdesign):

* + Open in ‘XML View’:
    - Update version number, and the default xml namespace

<design dslVersion="1.2.0.0" xmlns="<http://schemas.microsoft.com/visualstudio/patterning/authoring/workflowdesign>"

#### All Project and Item Templates Files (\*.vstemplate):

* + Remove the 'Version' and 'Culture' strings from all <WizardExtension> elements for assemblies beginning with ‘Microsoft.VisualStudio.Patterning’. e.g.

<WizardExtension>  
    <Assembly>Microsoft.VisualStudio.Patterning.Library, PublicKeyToken=31bf3856ad364e35</Assembly>

* + Replace all public key tokens. e.g.

<WizardExtension>  
    <Assembly>Microsoft.VisualStudio.Patterning.Library, PublicKeyToken=31bf3856ad364e35</Assembly>

#### Source.extension.tt

* + Update the name, type and file extension of the directive processor:

<#@ **ProductStateStore** processor="ProductStateStoreDirectiveProcessor" requires="fileName='..\YourToolkit.slnbldr" #>

* + Update version number and name of VSIX reference to 'Pattern Toolkit Runtime'

<Reference Id="93373818-600f-414b-8181-3a0cb79fa785" MinVersion="1.2.0.0">

<Name>Pattern Toolkit Manager</Name>

<VsixPath>PatternToolkitManager.vsix</VsixPath>

</Reference>

* + Delete following elements:

<CustomExtension Type="ToolkitDocumentationRef" IsCustomizable="true"><#= toolkitDocumentation #></CustomExtension>

<CustomExtension Type="PatternDocumentationRef" IsCustomizable="true"><#= patternDocumentation #></CustomExtension>

<CustomExtension Type="PatternRequirementsRef" IsCustomizable="true"><#= patternRequirements #></CustomExtension>

* + Replace:

<CustomExtension Type="SchemaModel"

* + With:

<CustomExtension Type="PatternModel"

#### Source.include.t4

* + Delete the following lines:

string toolkitDocumentation = @"Documentation\ToolkitDocumentation.docx";

string patternDocumentation = @"Documentation\PatternDocumentation.docx";

string patternRequirements = @"Documentation\PatternRequirements.xlsx";

#### ‘Documentation’ folder:

* + Delete the documents:
    - Documentation\PatternDocumentation.docx
    - Documentation\PatternRequirements.xlsx
    - Documentation\ToolkitDocumentation.docx
  + Delete the 'Documentation' folder if empty.

### Toolkit Automation Project

* **Toolkit Automation Project File (csproj):**
  + Replace All:

$(LocalAppData)\Microsoft\VisualStudio\10.0\Extensions\Microsoft\Pattern Toolkit Runtime\1.2.15.1

* + With:

$(PatternToolkitRuntime)

* + Replace All:

$(LocalAppData)\Microsoft\VisualStudio\10.0\Extensions\Microsoft\Feature Extension Runtime\1.0

* + With:

$(FeatureExtensionRuntime)

* + Replace All:

$(LocalAppData)\Microsoft\VisualStudio\10.0\Extensions\Microsoft\Pattern Toolkit Builder\1.2.15.1

* + With:

$(PatternToolkitBuilder)

* + Replace (or add if not exist):

  <Import Project="$(PatternToolkitBuilder)\Microsoft.VisualStudio.Patterning.Authoring.Toolkit.targets" />

* + With:

<Import Project="$(localappdata)\Microsoft\MSBuild\Microsoft\Pattern Toolkit Builder\Microsoft.VisualStudio.Patterning.Authoring.PatternToolkitVersion.targets" />

* + Before the above <Import> add this MSBUILD property:

<PropertyGroup>

<CopyVsixDebuggingDependencies>false</CopyVsixDebuggingDependencies>

</PropertyGroup>

### All Custom Automation Code:

* In all custom automation code
* Replace All:

ObjectValidator.ThrowIfInvalid(this)

* With:

this.ValidateObject()

### Author’s Development Machine, and Build Servers:

* On the desktop
* Shut down all instances of Visual Studio, and delete folder:

%AppDataLocal%\Microsoft\MSBuild\Microsoft\Pattern Toolkit Builder