InDesign CS Online Training Seminars: Changes from CS to CS2

This document lists the details in the InDesign CS Developer Online Training Seminars that are specific to InDesign CS AND need to be changed for InDesign CS2. Please juxtapose this document with the InDesign CS Developer Online Training Seminar presentations to get a full picture for solutions in InDesign CS2.

-- Adobe InDesign Developer Technologies, May 2005

Global changes

- The name of the InDesign product has changed from InDesign CS to InDesign CS2.
- The name of the SDK has changed from "The Adobe InDesign CS/InCopy CS Combined SDK" to the "Adobe InDesign CS2 Products SDK".
- The ASN Partners URL has changed. The OLD URL is: http://partners.adobe.com/asn/indesign/sdk.jsp
- The URLs to the exploded SDK reference documentation (included within notes of all presentations) all need to be updated, however, at the time of this writing, they were unknown. Please visit http://partners.adobe.com/public/developer/indesign/devcenter.html for details.

Session 1: Getting Started

Slide 7:
- The CodeWarrior version that is supported by the SDK has changed from 8.3 to 9.3.
- The Visual Basic 6 Runtime is no longer needed for using the SDK, since the TypeInfoConverter tool (written with Visual Basic 6.0) is no longer included in the SDK.

Slides 8, 9:
- The installation paths of the SDK installer is different in CS2. The size of the installer and installation footprint are also different in CS2. Please refer to the actual installer for details.

Slide 13:
- The [Making your first plug-in] technote has been renamed in the CS2 SDK. See details below under Session 2: Foundation.
- SnowGoose no longer exists in CS2 SDK, as the task of migrating plug-in projects from CS to CS2 is quite different than the task of migrating from 2.x to CS. Also, there are no project setting changes required for Visual Studio .NET 2003 plug-in projects (.vcproj), but there are significant changes for CodeWarrior projects.
For that purpose, the CS2 SDK contains a tool called Graygoose. Please refer to the [Porting Guide] in the CS2 SDK for details.

Slide 14:

Slide 15:
- CodeWarrior now creates frameworks, not single .pln files. See the [Porting Guide] section Creating a Macintosh plug-in project from scratch for more details.

Slide 17:
- This slide shows a screen shot from InDesign CS. The equivalent screen in InDesign CS2 may look slightly different.

Slide 18:
- The InDesign debug build is downloadable from the ASN Partners web site. Please contact the ASN Developer Program, or the InDesign Developer Evangelist - Mark Niemann-Ross (mnr@adobe.com) for details.

Slide 19:
- The [Porting Guide] section "Building and testing the debug plug-in on Mac" no longer exists. The [Porting Guide] section "Building and testing the debug plug-in on Windows" is on a different page number, but the section name is still the same. The Windows-specific section in the [Porting Guide] mentions the use of a "PluginConfig.txt" file to help load your plug-ins that may be in directories outside of the application's Plug-Ins folder; the same technique does still work on the Macintosh. Make sure you use the Macintosh-style path delimiters (";"), not POSIX-style delimiters ("/").

Slide 24:
- The URL for ASN developer program is now:

Slide 26:
- Snippet Runner's UI has a few changes, including an extra button for opening the parameters edit box. Please refer to the SnippetRunner documentation in the CS2 SDK, which is included in {SDK}/docs/references/index.chm or sdkdocs.tar.gz. (The URL for exploded SDK documentation is unknown at the time of this writing.)
Session 2: Foundation

The [Making Your First InDesign Plug-in] technote in the InDesign CS SDK has been renamed "Working with InDesign Plug-ins" (ww-plugins.pdf) in the CS2 SDK. Some details, such as DollyXs usage steps and section titles have been modified; however, the flow of the document is the same.

Session 3: Object Model

Slide 34:
- API change:
  CS: ILayoutUtils::QueryActiveWorkspace
  CS2: ILayoutUIUtils::QueryActiveWorkspace
- In addition to QueryPreferences (whose parameter list has changed between CS and CS2), you can also use the new CS2 API QuerySessionPreferences.

Session 4: Databases and Commands

Slide 12 and 17:
- There are new Façade interfaces in the CS2 SDK. See:
  - {SDK}/source/public/interfaces/architecture/IFootnoteFacade.h
  - {SDK}/source/public/interfaces/architecture/IUserActionFacade.h
  - {SDK}/source/public/interfaces/graphics/IImageFacade.h
  - {SDK}/source/public/interfaces/graphics/IXPAttributeFacade.h
  - {SDK}/source/public/interfaces/layout/IFrameContentFacade.h
  - {SDK}/source/public/interfaces/layout/IGeometryFacade.h
  - {SDK}/source/public/interfaces/layout/IObjectStylesFacade.h
  - {SDK}/source/public/interfaces/layout/ITransformFacade.h
  - {SDK}/source/public/interfaces/text/ITextWrapFacade.h

Slide 14:
- Information about Automatic Undo does not apply in CS nor CS2.

Slide 26:
- The Diagnostics plug-in is also scriptable. Please refer to the [Making Your Plug-in Scriptable] technote in the CS2 SDK for tips on how to get information about plug-in scripting interfaces. Also refer to the notes for session 16: Scripting.
**Session 5: Document Structure**

- The INDD files that are attached to the slides-PDF for this session are in InDesign CS format, however, since they do not use any special plug-ins, they should convert without issues to the CS2 format.

Slide 42:
- In the notes of this slide, a method called HitTest is listed. This method never existed on this interface, and is therefore an error.

**Session 6: Messaging**

Slide 8:
- The [Making Your First InDesign Plug-in] tehnote has been renamed in the CS2 SDK. See **Session 2: Foundation**, above.

Slide 11:
- In addition to the list on the slide, the ActiveSelectionObserver class is ALSO used in the following CS2 SDK plug-ins:
  - HiddenTextEditor
  - PersistentListUI (this is a UI-only plug-in, formerly part of PersistentList)
  - TableAttributes

Slide 23:
- In addition to the list on the slide, the following CS2 SDK plug-ins now include a service provider implementation.
  - BasicShape
  - BasicTextAdornment
  - CHDataMerger
  - CHLinguistic
  - CustomActionFilter
  - CustomDataLink and CustomDataLinkUI
  - Hyphenator
  - PrintMemoryStream
  - PrintSelection
  - SuppressUI
  - XDocBookWorkflowUI

**Session 7: Selection**

Slide 8, 9, 10, 11, 13, 15, 16, 17:
The CS SDK [Selection] technote has been folded into the CS2 SDK [Programming Guide] as the "Selection Fundamentals" chapter. The section names should mostly still be the same; please refer to the section names listed on the slides.

Slide 12, 16:

There are a lot more samples in the SDK that use suite interfaces and active selection observers. The list of samples that include an active selection observer implementation in the CS2 SDK (that did not in the CS SDK) are listed above (see Session 6: Messaging, slide 11). For CS2 SDK sample code that use a specific suite interface, use the API/sample reference documentation to search.

Session 8: User Interface

Slide 7:

InDesign CS2 is now available in 16 locales. Chinese Simplified, Chinese Traditional and Korean have been added to the product family. In addition, the MLocaleIds.h/WLocaleIds.h header files in the SDK include all of the ME/CE version locales and a few more.

Slide 10, etc. (Search for "User Interface")

[User Interface] technote has been folded into the CS2 SDK [Programming Guide] as multiple chapters and the [Working with User Interface] document. Please see those documents for more details.

Slide 17:

The context menu strings "RtMouseLoupe" (loupe tool) and "RtMouseSpellSuggestions" (dynamic spell checker) are supported in CS2.

Slide 37:

For a discussion on the need to separate the model and UI components in your plug-in, please refer to the [InDesign Server Plug-in Techniques] technote in the CS2 SDK. Model plug-ins normally should contain script providers, unless the scriptability is related to the UI itself.

Slide 38:

In addition to the list on this slide, the CS2 SDK contains these model-UI plug-in combinations. In some cases, scriptability and InDesign Server support have also been added in.

- CHDataMerger and CHDataMergerUI
- CustomDatalink and CustomDatalinkUI
- InCopyExport and InCopyExportUI
- HiddenText and HiddenTextEditor
- PersistentList and PersistentListUI
Session 9: Layout

Slide 9:
- The Macintosh file type has been updated in CS2. For InDesign CS documents, it was 'IDd3'. For InDesign CS2 documents, it is 'IDd4'.

Slide 11, 23, etc. (Search for "Document Structure" or "[PG] Section")

Slide 12:
- The ILayoutUtils::GetFrontDocument in InDesign CS has been changed to ILayoutUIUtils::GetFrontDocument in InDesign CS2. Also note that it cannot be called from an InDesign Server plug-in since there is no concept of a "front" document in InDesign Server.

Slide 16:
- Many methods on the ILayoutUtils interface in InDesign CS have been moved to the new ILayoutUIUtils interface in InDesign CS2. Also, the QueryLayoutData and QueryFrontLayoutData methods on ILayoutUIUtils cannot be called from an InDesign Server plug-in since there is no concept of a "front" document in InDesign Server.

Slide 42, 43, 44:
- The InDesign CS code snippet SnpProcessCmds has been changed to SnpProcessDocumentLayerCmds in the InDesign CS2 SDK.

Slide 44, 45:
- Samples for kMoveLayerCmdBoss, kLockLayerCmdBoss, kShowLayerCmdBoss, kCreateLayerFromLayerCmdBoss, and kDeleteLayerCmdBoss no longer exist in the CS2 SDK. (In the CS SDK, these commands were used in MultiUserHelper.cpp. Please refer to the CS SDK for a sample.)

Slide 57:
- The CS API ILayoutUtils::GetVisibleSpreads is now ILayoutUIUtils::GetVisibleSpreads in CS2. (NOTE: In general if you cannot find a method on ILayoutUtils, look in ILayoutUIUtils. Also be aware that ILayoutUIUtils is not usable in InDesign Server.)

Slide 75:
- Samples for kShowGuideLayerCmdBoss and kLockGuideLayerCmdBoss no longer exist in the CS2 SDK. (In the CS SDK, these commands were used in MultiUserHelper.cpp. Please refer to the CS SDK for a sample.)

Session 10: Graphics

Slide 22:
• GraphicAttributes.xls may not have all of the graphic attributes in CS2 listed. Please use the code snippet to dump this table. You will need to modify the code snippet to run under the CS2 version of SnippetRunner (see SnpTemplate.cpp for instructions) and possibly to accommodate new graphic attribute data interfaces.

Slide 23, 24:
• The CS code snippet SnpInspectGraphicAttributes has been merged with the CS code snippet SnpApplyGraphicAttributes into the CS2 code snippet, SnpManipulateGraphicAttributes.

Slide 25:
• JumpStoryHelper::removeGraphicAttribute no longer exists in CS2 SDK. Please refer to the JumpStory sample plug-in in the CS SDK.

Slide 26:
• In the Notes, Ole Kvern's first name is missing.

Slide 36, 37:
• The CS code snippet SnpCreateGradientSwatch has been renamed as SnpManipulateSwatch in the CS2 SDK.

Slide 40:
• InDesign CS2 introduced the concept of object styles. Please note that this presentation does not refer to object styles. Refer to the section "Export object styles from document to snippet" and "Import object styles from snippet" in the "Snippet Fundamentals" chapter of the CS2 SDK Programming Guide.

• The slide about color management API (slide 49) could be moved near this slide. Here are some related topics on the InDesign SDK forum:
  ➢ http://www.adobeforums.com/cgi-bin/webx?128@175.Zbcldd60RbT.4@.3bb841cf
  ➢ http://www.adobeforums.com/cgi-bin/webx?128@175.Zbcldd60RbT.1@.3bb86f34
  ➢ http://www.adobeforums.com/cgi-bin/webx?13@175.Zbcldd60RbT.0@.3bb86a98/1

Slide 52:
• The CS SDK technote [Transparency Effects] and [Colour Fundamentals] have been amalgamated into the "Graphics Fundamentals" chapter of the CS2 SDK Programming Guide.

Session 11: Text Model

• General note: The chapter, section and figure numbers of CS SDK Programming Guide on this slide are no longer relevant in the CS2 SDK Programming Guide. Look for the chapters by title name.

Slide 13:
• Technote "Unicode 3.x and String-Related APIs in InDesign CS/InCopy CS" no longer exists in the CS2 SDK. Please see the API reference documentation for more details on each API.

Slide 25:
• The InDesign SDK Forum has some relevant hints/topics related to the content on this slide.
  ➢ http://www.adobeforums.com/cgi-bin/webx?128@175.BbV1d5QTRbM.4200057@.3bb5e1e5
Slide 28:
- These strands are also available in the CS2 object model.
  - kRedlineStrandBoss
  - kSpellingStrandBoss
  - kTextDataStrandBoss (internal use)
  - kTextIndexIDLStrandBoss (internal use)
  - kXMLStrandBoss (internal use)

Slide 34:
- The CS2 SDK Programming Guide no longer has a text attribute chart. To find a list of text attribute bosses, please use the Diagnostics plug-in to generate an object model dump, import the generated text file into Excel, and filter on the interface IID_IATTRREPORT. You can also refer to the API reference docs for IAttrReport.

Slide 35:
- The CS2 SDK Programming Guide no longer has a sample code snippet within the documentation for TextAttributeRunIterator. Instead, please refer to the CS2 code snippet SnpInspectTextModel (see InspectStoryPointSizes).

Slide 45:
- “REFERENCE: Master list of text attributes” is no longer in the CS2 SDK Programming Guide. Please see note for Slide 34 above.

Session 12: Text Layout

General note: The chapter, section and figure numbers of CS SDK Programming Guide on this slide are no longer relevant in the CS2 SDK Programming Guide. Look for the chapters by title name.

Slide 18:
- TinMutHelper::GetTextInsets no longer exists in CS2 SDK.

Slide 19:
- In CS2, kChangeNumberOfColumnsCmdBoss uses a new data interface, ITextColumnData. Please see the code snippet SnpManipulateTextFrame (IncrementNumberOfColumns method) in the CS2 SDK.

Slide 20:
- The [Japanese Feature Additions in InDesign CS] technote mentioned on this slide is the process of being folded into the CS2 SDK Programming Guide. In the mean time, please refer to the InDesign CS version of this technote, which is available at:
Please note that in InCopy-Japanese, the grids for the story are set using IGridAttributesPrefs (kDocWorkspaceBoss, kWorkspaceBoss, kDocPresetBoss, kDocPresetDefaultBoss). To change the InCopy-Japanese grid settings, process one of these commands:

- kSetGridAttributesPrefsCmdBoss (for general InCopy-Japanese grid attributes)
- kSaveDocumentPresetDataCmdBoss (for document preset settings)

Session 13: Tables

Slide 12:
- In CS2, you can also call the ITableUtils::GetTableModel method to get the table model at a specific text index.

Slide 19:
- In CS2, creating a new table can be done by first calling ITableUtils::CanInsertTableAt, then calling ITableUtils::InsertTable façade instead of processing the command. See SnpCreateTable::CreateTable. You then call ITableUtils::GetTableModel to get the table model at a specific text index.

Slide 26:
- The master table of table attributes needs to be updated for InDesign CS2. To find a list of table attribute bosses, please use the Diagnostics plug-in to generate an object model dump, import the generated text file into Excel, and filter on the interface IID_ITABLEATTRREPORT. You can also refer to the API reference docs for ITableAttrReport.

Slide 30, 31:
- The ITablesStyleNameTable interface is no longer available in CS2. You need to use IStyleNameTable instead, with the specific IID of IID_ITABLESTYLENAMETABLE. Creation of table styles have not changed, however on slide 31, instead of clearing the style with UIDRef::gNull, that setting will apply the root table style (which is essentially the same thing).

Slide 34:
- (Error in slide, applies to CS as well) ITableComposer is on kTableModelBoss, not kTableFrameBoss. NOTE: It's been this way since 2.x.

Slide 36:
- In CS2, you can also use ITableUtils::GetCellChunks, which does the same thing internally.

Session 14: Service Providers

- General note: The chapter, section and figure numbers of CS SDK Programming Guide on this slide are no
longer relevant in the CS2 SDK Programming Guide. Look for the chapters by title name.

Slide 9:
- The Service Providers table provided in Session 6 will need to be updated. To find a list of service providers, please use the Diagnostics plug-in to generate an object model dump, import the generated text file into Excel, and filter on the interface IID_IK2SERVICEPROVIDER. You can also refer to the API reference docs for IK2ServiceProvider.

Slide 10:
- The reference table in the CS SDK Programming Guide is no longer there in CS2 SDK version. Instead, refer to the spreadsheet from Session 6 instead, or refer to the API reference docs for IK2ServiceProvider.

Slide 18:
- The MultiUser and JumpStory sample plug-ins are not available in the CS2 SDK. Please refer to the CS SDK.

Slide 20:
- A new sample in the CS2 SDK called XMLCatalogHandler also has a startup-shutdown service.

Slide 21:
- There are corrections needed for the notes on this slide, in the "to do nothing" section. See notes on slide 25 of Session 11: Text Model.

Slide 23:
- The C++ code that is shown in the notes is incorrect. You do need to create an implementation, but you really don't need to implement any of the methods but the constructor. In the CS2 SDK, see BPIErrorStringService.cpp, CHDMErrorStringService.cpp, PstLstErrorStringService.cpp or SnipRunErrorStringService.cpp.
- Another requirement change for CS2 is that script providers should now return proper error codes that have corresponding error strings registered with an error string service, instead of just returning kFailure. This is so that the scripting engine can report a proper error string to the script user, thus enhancing the usability of the InDesign scripting interface.

Slide 25:
- Error: kOpenProviderService should be kOpenServiceImpl instead. This also applies to CS.

Slide 30:
- The following export format names are available in CS2:
  - "InCopyStory" and "InCopyInterchange"
    (preferences: IInCopyExportOptions – kWorkspaceBoss)
  - "InDesignSnippetFile"
  - "InDesign 4.0"
- SnpExportSelection was never shipped in the CS SDK or in training content.

Slide 32:
- See notes for Slide 30 above for preference interfaces for additional format names. The preference interfaces for Sangam related export options are not available in the SDK.

Slide 34:
- IUnitOfMeasure is also used by SnpGetTableParcel. The reference to TinMutDialog should be removed.

Slide 35:
- There is a new sample plug-in in the CS2 SDK called CustomActionFilter. Please take a look.

Slide 39:
- Easter Egg implementations in the application have ALL been removed (SVG, Bounce, etc.), but the hook is still there.

**Session 15: Printing**

Slide 25:
- The confusing aspect of the Edit command’s notification has been filed as a bug (1162593). However, the resolution might be deferred until a future version. Another way to tell apart the two commands is to get the ClassID from ICommand* (after typecasting the void* changedBy).

Slide 26:
- You don’t need to set call SetStyleIndexList for import.

Slide 30, 32:
- The issue about the parameters for AddStyle/EditNthStyle methods not being included has been reported as bug 1002088.

Slide 33:
- The issue about the parameters for ITrapStyleUtils methods not being included has been reported as bug 1002089.

Slide 35:
- The no-argument method for QueryTrapStyleListMgr no longer exists in CS2. Use the method one with a parameter.

Slide 48:
- Error: DrawEventData::flags is not a call, it is a member variable.

Slide 56:
- IPDFSecurityPrefs is the primary data interface for the Security panel.

Slide 59:
- InDesign CS2 uses PDFL70 version 7.0.0

Slide 63:
The Package for GoLive technote is no longer available in the CS2 SDK.

Slide 64:
The Printing chapter number has changed in the CS2 SDK Programming Guide. It has also been largely expanded, including most of the content from this training session.

Session 16: Scripting

The [Making Your Plug-in Scriptable] technote has been largely revamped in the CS2 SDK. As a result, the section titles, page numbers, and various other content in the CS SDK technote may no longer apply in the CS2 version.

At the time of this writing, a new CS2 specific training session was being developed on the topic of making your plug-in scripting in CS2. In CS2, the need for implementing script providers has increased with the prolific use of the INX snippet architecture in various parts of the application.

Slide 7:
The path on the Product CD-ROM for the Scripting Guide is probably different for CS2. (Unknown at the time of this writing.)

Slide 9:
The second bullet: "If your plug-in adds custom persistent objects or preference interfaces, adding scripting capabilities will add support during:" should have the following:

- File formats related to the InDesign Interchange (INX) features (*.inds, *.inx, *.incx, Library assets, etc.)
- Package for GoLive CS2

Also the notes should reflect the same set of items.

Slide 14:

Slide 15:
The paths where the TypeLibrary/AETE get generated are different in CS2. Refer to the section "Reviewing Scripting Resources" in the CS2 SDK technote [Making Your Plug-in Scriptable]. Also, in Windows, InDesign CS2 will generate two copies of the type library (one in the All Users folder, and one in the user-specific folder), but only one is updated at startup. Again, refer to the same section in the technote.

Slide 18:
The CS resource ScriptElementInfo still exists, however, to allow for versioning of scripting resources, it is highly recommended that you use the VersionedScriptElementInfo resource instead, in CS2. See the "Versioning of Scripting Resources" section in the CS2 SDK technote [Making Your Plug-in Scriptable].
There are a few new resources in the SDK that help you determine where in the Scripting DOM you can add your own scripting elements. You can refer to:

- The revamped CS2 SDK technote [Making Your Plug-in Scriptable] (See section "Recipes")
- DOM references in the \{sdk\}/docs/references folder (See section "Scripting DOM Reference")
- The scripting DOM dump feature in the Debug build. (See section "Dumping the Scripting DOM")

Slide 19:
- Reminder ScriptID/Name pairs now need to be registered. See note on Slide 14.
- The last paragraph in the notes needs to be removed, and replaced with the following links shareware GUID/UUID generators for OS X.
  - http://www.adobeforums.com/cgi-bin/webx?13@674.IotRdnLNbw2.706956@.3bb4bb8c/0
  - UUID Generator: http://www.versiontracker.com/dyn/moreinfo/macosx/11848

Slide 20/21:
- The "Types of Objects" bullets from both the slide and the notes have been elaborated upon in much greater detail in the revamped CS2 SDK technote [Making Your Plug-in Scriptable] (See section "Recipes"). Please refer to the technote instead.

Slides 23-30:
- Instead of referring to this Step-by-step section, you can now refer to the "Recipes" section in the revamped CS2 SDK technote [Making Your Plug-in Scriptable]. The CS2 SDK also contains more scriptable plug-ins:
  - BasicPersistInterface
  - BasicShape
  - BasicTextAdornment
  - CHDataMerger
  - PersistentList
  - PrintSelection (further enhanced since CS)
  - SnippetRunner (first introduced in the training session, further enhanced and bugs fixed 😊)

Slide 32:
- CanHandleFile now takes an IDFile in CS2, not a SysFile (as it did in CS).
- RunFile takes more parameters, including a variable to return the result of the script (which can be passed back from the script via a "returnValue" variable in a single-file VBScript, the result of an AppleScript, or the result of a JavaScript), and the error string associated with the script if any. RunScript is another method you can use if you want to run a script in memory (not from a file).

Slide 33:
- SnpInspectScriptingDOM is not yet updated for CS2. Nonetheless, using IDOMElement to iterate the content is perhaps a bit more efficient than using IHierarchy.
- Instead of looking at the scripting DOM contents of a specific document, you can generate the complete
runtime Scripting DOM in InDesign or InCopy, by dumping it to a file by using the Debug build, or by using
the new code snippet SnpCreateScriptingDOMReference. See the “Dumping the Scripting DOM” section in
the revamped CS2 SDK technote [Making Your Plug-in Scriptable] for details.

Slide 34:

• In CS2, IScript inherits from IScriptLabel, where the Get/SetTag methods are declared.
  ➢ Get: You can still call IScript::GetTag.
  ➢ Set: Call IScriptUtils::SetScriptingTag or SetScriptingTags.
    To get the ScriptList, query the IScript interface on the items you want to modify labels, and add them
to the list (K2Vector)
    To get the RequestContext, do this;
    InterfacePtr<IScriptManager>
    scriptMgr(Utils<IScriptUtils>()->QueryScriptManager(kScriptTagMgrBoss));
    RequestContext context = scriptMgr->GetRequestContext(),
    The ScriptLabelKeyValueList is a K2Vector<ScriptLabelKeyValuePair>. See IScriptLabel.h.
  ➢ (NOTE: you can still process kSetScriptingTagCmdBoss, but the data interface is now
    IScriptTagCmdData)
  ➢ Notification details are still the same.

Session 17: Cross Media

• IXmLUtils has a few new methods in CS2 to allow for getting the backing store and specific XMLReferences
  from content UIDs.

Slide 17, 18:

• SnpPerformXMLTags is no longer available in the CS2 SDK. See SnpXMLHelper or SnpManipulateXMLTags
  instead.

Slide 24, 25, 26:

• SnpPerformXMLElements is no longer available in the CS2 SDK. See SnpManipulateXMLSelection or
  SnpManipulateXMLElements instead. For samples using IIDXMLElement, examine any of the Snp***XML***
  code snippets.

Slide 30, 31, 32, 33:

• SnpPerformXMLTagAndStyle is no longer available. Refer to the documentation for IXmLUtils,
  IXMLElementCommands and IXMLElementToStyleMap.

Slide 34-38:

• There are many enhancements to the XML Import APIs in CS2. Refer to the CS2 SDK for the latest XML
  technotes.

Slides 43, 44:
● IID_IXMLTAGLIST is the protocol for tag related commands (via the ISubject on the boss with IXMLTagList)

Slide 47:
● XDocBookWorkflow has undergone a few improvements. Some of the contents on this slide may be out of date. Refer to the XDocBookWorkflow sample code.

Slide 49:
● The CS SDK technote [XML Productivity Features] is now the “XML Fundamentals” chapter in the CS2 SDK [Programming Guide].