

Adobe InDesign CS2 and XML: A Technical Reference

TABLE OF CONTENTS

- 1 Overview of XML syntax
- 2 Integrating XML into a publishing workflow
- 4 XML tagging features in InDesign
- 11 Planning an XML workflow
- 12 Importing XML files
- 18 Formatting XML text content manually
- 19 Formatting XML text using style mapping
- 19 Formatting text using namespace attributes in import and export
- 23 Using anchored objects for repeating elements
- 24 Exporting XML
- 25 Using InDesign Interchange (INX) files in XML workflows
- 26 Where to go from here
- 27 Appendix: XML syntax

Adobe® InDesign® CS2 empowers design creativity with its support for attractive layouts, sophisticated graphics, and beautiful typography. Creative professionals can also integrate InDesign and Extensible Markup Language (XML) files to set up a multichannel publishing workflow.

Combining the predictable structure of XML files with InDesign lets you automate layout tasks. You can associate InDesign formatting with XML content, so that when you import an XML file, content is automatically laid out on your pages. If you lay out many documents that have similar formatting, you can set up your design for the first document and then automate subsequent publications.

XML also serves as a useful interchange format to move information among different applications. For example, if you create content in InDesign that will be published in other media (such as on the web), you can export the InDesign file as XML to reuse the content.

Together, the powerful layout engine in InDesign and the XML structures let you establish a publishing workflow that reduces production effort without compromising your design standards.

This white paper provides detailed technical information about XML support in InDesign CS2.

Overview of XML syntax

Extensible Markup Language, or XML, is a text-based file format managed by the World Wide Web Consortium (www.w3.org). The Consortium describes XML as follows: “Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the exchange of a wide variety of data on the Web and elsewhere.” A basic XML file is readable, as shown in the following sample:

```
<?xml version="1.0"?>
<Article author="CJM" date="2008-07-17">
  <Dateline>Anytown, Anywhere</Dateline>
  <Title>Traffic is Worse than Ever</Title>
  <Para>Complaints about traffic are on the rise. A recent survey shows
that commuting time is increasing.</Para>
  ...
</Article>
```

Like Hypertext Markup Language (HTML), XML uses angle brackets to indicate markup tags (for example, <Article> or <Para>). HTML has a predefined set of tags; in XML, you describe content more precisely by creating custom tags. The preceding example uses descriptive tags, such as Article, to label the information.

For more details about XML syntax, see the appendix on page 27.

Integrating XML into a publishing workflow

XML is widely used for data exchange, such as e-business transactions. For creative professionals, who mainly work with content, XML is also becoming a critical technology. XML provides some exciting new opportunities for your publishing workflow:

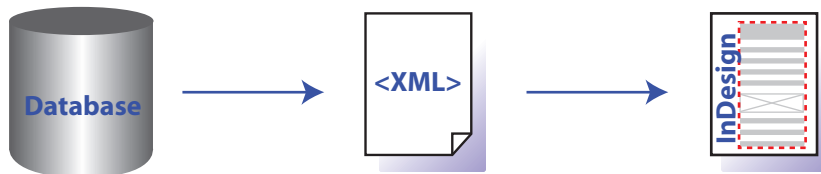
- **Layout automation.** When the content structure is predictable, you can automatically process it to produce a consistent look and feel in your final output. For example, you can import XML into InDesign and lay out documents automatically. You can also export content from a database file to XML, and then use the XML file to produce the final content (for example, to publish a directory). To ensure that content is organized consistently, you can define the required structure using a document type definition (DTD) file.
- **Reusing content.** You can use XML as an intermediate format to exchange information between otherwise incompatible applications. This process lets you reuse information without tedious manual reformatting. XML provides extensive support for metadata, so you can provide additional information about your document and about elements within the document.

The following sections provide more detail about how InDesign supports XML-based publishing.

Layout automation with InDesign

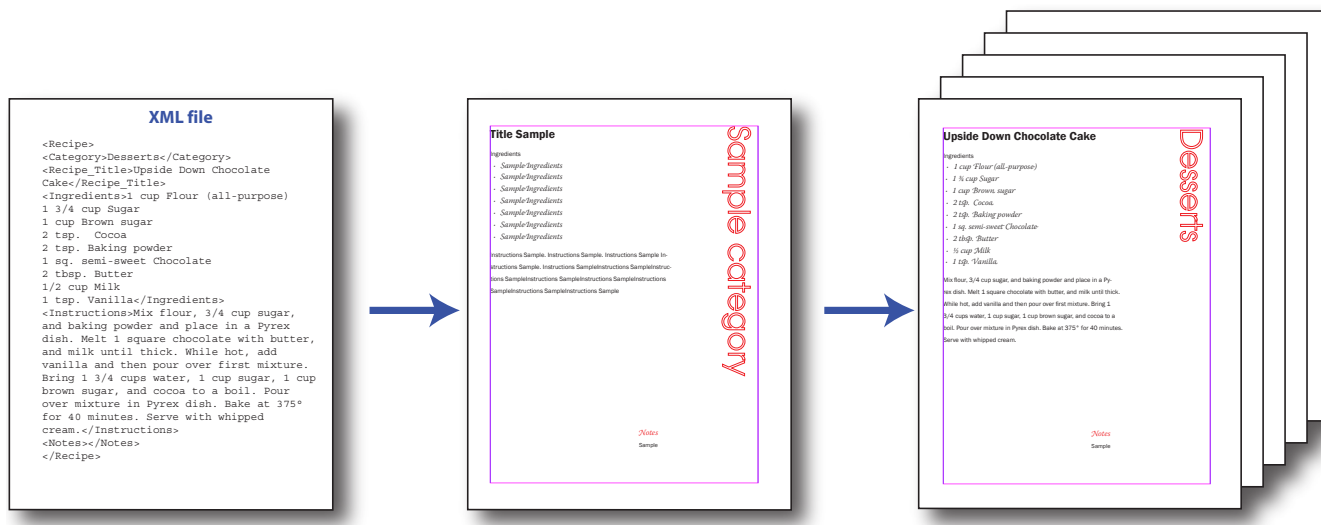
XML lets you set up workflows with layout automation. For example, you can create content in an XML file, and then import the XML file into InDesign. Instead of applying formatting paragraph by paragraph, you define the look and feel and associate XML tags with formatting. To ensure that information in the InDesign file changes automatically whenever the XML file is updated, you can create a link to the XML file in InDesign.

If the information you need to publish is stored in a database, you can export from the database to XML, and then import the XML file into InDesign, where InDesign formats the content automatically.



XML-based database publishing with InDesign

When publishing from a database, you often have a structure pattern that needs to be laid out identically from page to page. InDesign lets you define the page layout once, and then clone that layout for additional information. You can set up your layout using sample content or a typical set of information and then repeat that layout as many times as required for all of the content. For details on cloning layouts and other XML import options, see Importing XML files on page 12.



XML file with repeating structured content

Tagged InDesign layout

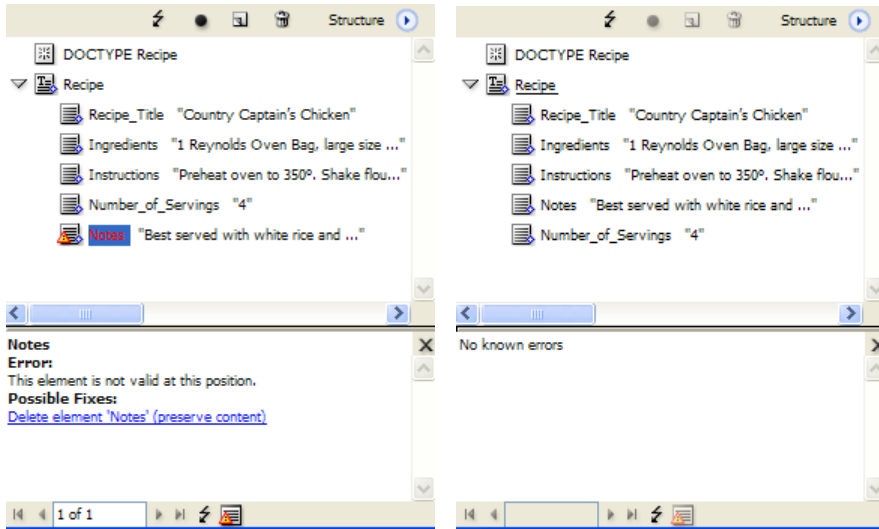
Page design and formatting can be cloned automatically.

PDF versus XML

PDF and XML support the publishing workflow in different, but complementary, ways. PDF is designed to preserve the exact appearance of a document from one medium to another. PDF documents can be printed without reformatting. A PDF file may also contain structured information, enabling it to adapt to different display devices and to interact with XML workflows. XML preserves the structure, but not the appearance, of content, as well as metadata about elements. XML is easy to edit and can be displayed differently for different viewing media, making it an excellent choice when you need to share content that will be edited or reprocessed into another output format.

Enforcing structure and organization

Using XML, you can spell out the rules for your documents and ensure that they are followed. Consider the example of a cookbook. In a cookbook, each recipe needs to be organized consistently—ingredients first, followed by instructions, notes, and perhaps a picture. If you have multiple authors contributing recipes, you want to make sure that all recipes are organized the same way. You can specify the required organization, and then validate a document to ensure that it follows the required structure.

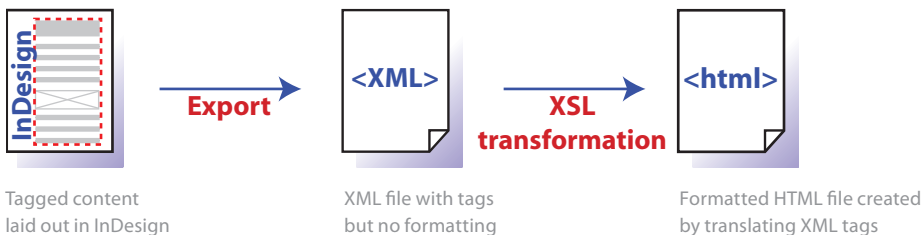


Use a DTD to validate the structure of your imported XML. On the left, the DTD identifies an error for you to fix—the Notes element must occur before the Number_of_Servings element. On the right, the structure is completely valid.

Structure rules are stored in a Document Type Definition (DTD). The structure rules need not be rigidly defined. For example, a DTD could declare that a recipe could have pictures, could have a bullet list, and so on, but in each case, leave open the option for it not having these items. With InDesign, an imported XML file can specify the DTD, or you can load it into a document separately. Once you have associated a DTD with XML content, you can validate the content. Alternatively, you could use an XML generation and editing application to do this validation separately from InDesign. In addition, InDesign does not require you to use a DTD, so you can also work without validating.

Reusing content from InDesign

Perhaps you create a weekly newspaper using InDesign. You have been asked to create a web-based version of the newspaper. You need to continue publishing the printed version, but you also need to extract stories from the newspaper and post them on your website. XML provides an excellent platform for setting up this type of content exchange. You can publish the newspaper as usual, then tag the content, export it to XML, and generate the HTML you need for the website from the XML file using Extensible Stylesheet Language (XSL) transformation.



Specifying metadata

Metadata is information about content. You can include metadata in an XML file and even assign metadata to specific document components. In a newspaper, for example, it would be useful to know the author and editor of each story. You can create metadata that provides that information as part of the XML structure:

```
<Newspaper>
  <story author="CJM" editor="SSO">story goes here</story>
  <story author="MDM" editor="SSO">story goes here</story>
</Newspaper>
```

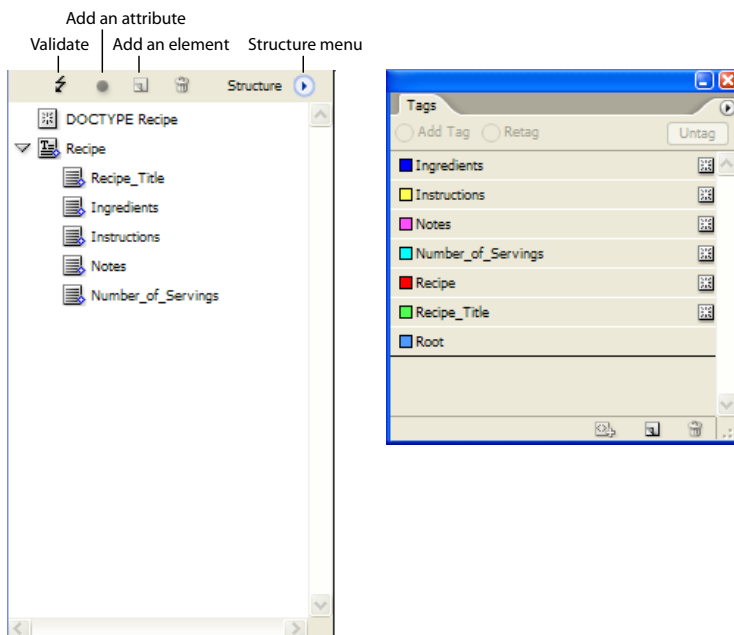
Metadata is embedded at the location of the content it refers to as an attribute of that content, ensuring that it always stays with the described element. Later, you could use the metadata to filter information; for example, to locate all of the stories a particular author has written.

InDesign provides additional support for metadata in any InDesign file with the Extensible Metadata Platform (XMP) standard. The XMP standard, created by Adobe, lets you store document information, such as the author, copyright information, and keywords, in a file. The XMP information is accessible to other users even if they do not have the application in which you created the file. All Creative Suite 2 applications automatically embed certain XMP metadata. For example, InDesign files include XMP metadata about the swatches and fonts used in a document, as well as any linked files. You can view, search, and edit metadata using the new Adobe Bridge (available from any CS2 application). You can also embed metadata in InDesign files by choosing File > File Info and editing or adding metadata. The XMP metadata is saved as XML in a special section of the application file.

XML tagging features in InDesign

InDesign offers several different ways of working with XML. This flexibility lets you set up a workflow that is compatible with your publishing environment.

Use the Structure pane and the Tags palette to work with XML structure in InDesign. The Structure pane shows you the document structure and lets you make changes to it. The Tags palette lists the XML tags that are available in the document. Just click the tag you want to apply it to selected content. (If you don't want to tag items manually, you can use the automated tagging techniques described later in this document.)



The Structure pane and Tags palette give you control over XML content. To display the Structure pane, choose View > Structure > Show Structure. Choose Window > Tags to open the Tags palette.

XML and InDesign documents

An InDesign document consists of pages that contain text, still graphics, and even motion graphics (for embedding in interactive PDF files). Text stories can flow through one or more linked frames. Associated XML files reference the external images or other media types that can be placed in InDesign.

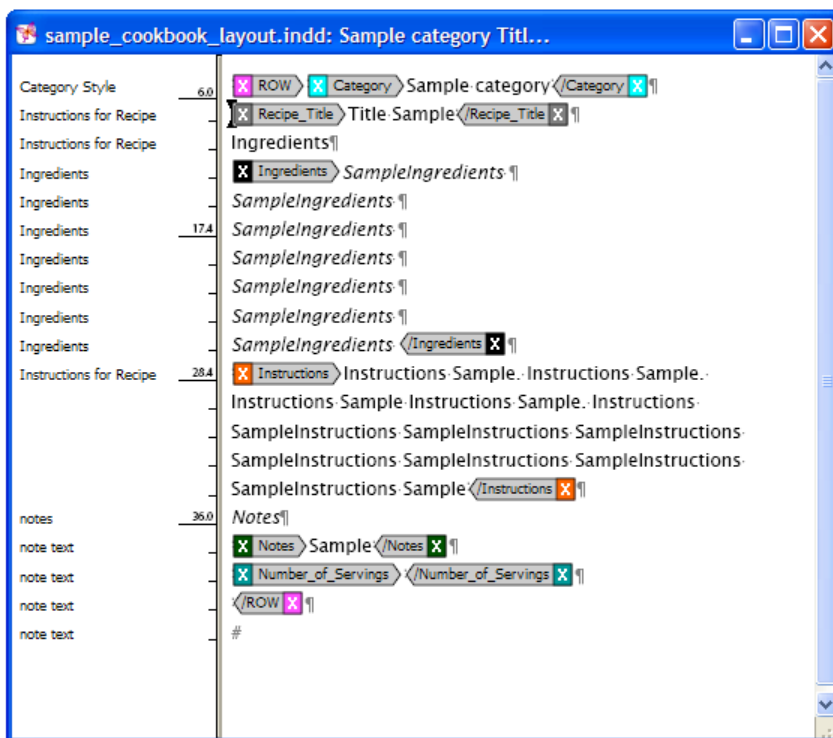
In contrast to this, an XML file consists of elements, attributes, and text. An XML file does not usually provide any font, page positioning, or other formatting information. You must specify how you want XML content to flow into your InDesign document.

An XML document is required to have a single root element. The root element contains all of the elements in the document, and is shown near the top of the Structure pane. In the art in the preceding section, the root element is Recipe.

You can associate XML with different InDesign objects. Most often, you associate XML with stories, paragraphs, text ranges, graphics, and other media. When an XML element is associated with a text frame, InDesign tags any frames that are linked to that text frame.

Viewing XML tags

To help you manage XML tags in an InDesign file, you can set up the file to show tags. In the main document window, tags appear as colored brackets. In the Story Editor, the entire tag name appears. To display XML tags, choose View > Structure > Show Tag Markers.



Tags visible in the Story Editor

You can also set your text frames to indicate the tag they are associated with by choosing View > Structure > Show Tagged Frames. InDesign displays the frames in the document window with a background color, which matches the color for the tag in the Tags palette.

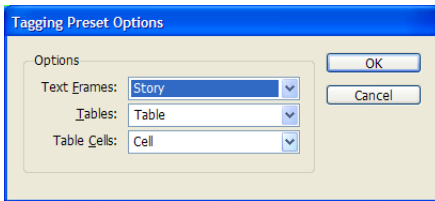
XML tagging techniques: Manual, automatic, and style mapping

You can assign XML tags to any text and graphics content. To assign a tag manually, select the content and then click a tag in the Tags palette. (If the content already has a tag assigned, make sure that Retag is selected at the top of the palette.)

For text frames, tables, and table cells, InDesign lets you specify the tags to be used when InDesign automatically tags those objects. If you tag a range of text or other item in an untagged story, InDesign tags the story automatically with the default text frame tag. Similarly, if you tag some text inside a table cell, InDesign tags the table cell, table, and containing story using the specified tags.

To use automatic tagging for text frames and tables:

- 1 In the Tags palette menu or the Structure pane menu, select Tagging Preset Options. By default, text frames are mapped to the Story tag, tables are mapped to the Table tag, and table cells are mapped to the Cell tag.
- 2 Make any changes to the settings, and then click OK.



Tagging Preset Options dialog box

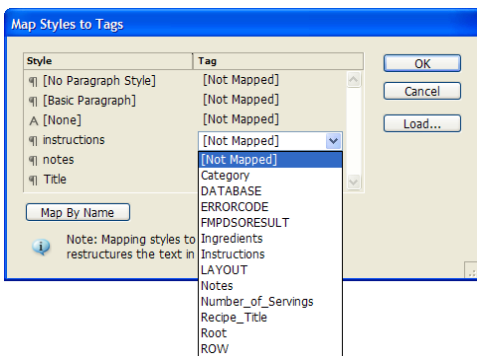
- 3 Select a text frame, table, or table cell, and then click the Autotag icon (🔗) in the Tags palette.

You don't have to select paragraphs one at a time and tag them if you're using paragraph, character, and nested styles. Instead, you can map style names to tags, so InDesign can automatically tag the specified content.

In addition, if you can control style names or tag names, consider matching them so that you can automate the mapping process. For example, if your XML content has a Caption tag, you can create a matching Caption style in your document. The Map By Name button lets you associate those styles automatically.

To assign tags automatically using style mapping:

- 1 In the Tags palette menu (or the Structure pane menu), choose Map Styles to Tags. The Map Styles to Tags dialog box opens.



Map styles for automatic XML tag assignment.

- 2 For each style you want to map, click the Tag list to display a pop-up menu and choose the appropriate XML tag from the list. If your style names match the XML tag names, click Map By Name.

Note: To use mappings you set up in another InDesign file, click Load and select the file.

- 3 Click OK. The new XML tags are applied throughout the document.

Loading a DTD into a document

To use InDesign structure validation features, you must either load a DTD into InDesign or validate against a DTD that is defined and referenced in the imported XML file. To load a DTD, it must be available on the system.

To load a DTD into an InDesign file:

- 1 Click the Structure pane menu, and then click Load DTD.
- 2 Locate the DTD file and select it.

Note: When you load a DTD file, it is embedded in the InDesign file (not linked). If you make updates to the DTD file, you must reload the DTD into InDesign.

- 3 Click Open.

InDesign checks the DTD as it is loaded. If any problems are found, InDesign displays an error message. To correct errors, open the DTD in a text editor, make the appropriate changes, and then try loading it again.

Validating documents against a DTD

Once you load a DTD, you can validate the InDesign file.

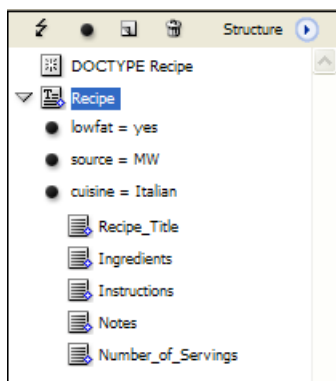
To validate the document:

- 1 Click the Validate button (🔍) in the Structure pane.
- 2 InDesign reports any structure errors at the bottom of the Structure pane and provides suggestions on how to fix the problem. Click a suggestion to try that correction. When InDesign identifies invalid structure, it displays a warning icon on the element icon and highlights it in red in the Structure pane.

When the document is valid, InDesign tells you that no validation errors were found.

Managing attributes

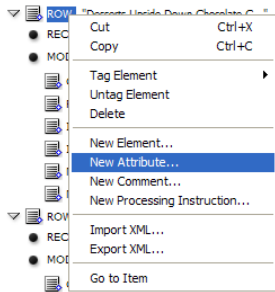
Attributes provide additional information about an element. In a cookbook, you could use attributes to provide metadata, such as the cuisine type, the recipe source, and whether a recipe is low in fat. InDesign displays attributes in the Structure pane only. Attributes cannot be placed into the document or used as content.



The Recipe element has three attributes.

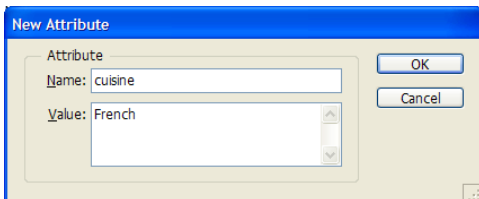
To add a new attribute:

- 1 In the Structure pane, right-click (Ctrl-click for Mac OS) the element that you want to add an attribute to and then choose New Attribute from the pop-up menu.



Pop-up menu resulting from the Structure pane.

- 2 Specify the name and value of the attribute, and then click OK.



New Attribute dialog box.

The new attribute is shown underneath its element in the Structure pane.

Note: Some attributes, such as table and image attributes, are created automatically when you export content to XML. You do not need to define these attributes in the Structure pane. For more information, see “Table attributes” on page 20 and “Images” on page 9.

To edit an existing attribute, double-click the attribute in the Structure pane, make any needed changes, and then click OK.

Note: DTDs may contain specifications for attributes. When you validate a document, InDesign notifies you about any required attributes that are missing, as well as attribute values that are not allowed by the DTD.

Inserting comments

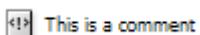
XML comments are similar to HTML comments. They let you embed information in the XML file. An XML comment looks like the following:

```
<!-- This is a comment. -->
```

To insert an XML comment:

- 1 In the Structure pane menu, choose New Comment. Or right-click (Ctrl-click for Mac OS) an element and select New Comment.
- 2 In the Value field, type the text you want for the comment. Omit the beginning and ending indicators (<!-- and -->, respectively). That is, if you type `This is a comment` the result in the XML output is:

```
<!-- This is a comment -->
```



The Structure pane shows comments.

Inserting processing instructions

Processing instructions let you embed special commands in an XML file. These commands are usually application-specific; for example, you can use processing instructions to embed page breaks or conditional processing information.

```
<?command command_details ?>
```

To create a processing instruction:

- 1 In the Structure pane, click where you want to insert the processing instruction.
- 2 Click the Structure pane menu, and then choose New Processing Instruction.
- 3 Enter values in the Target and Data fields. For example, to create a processing instruction that reads: `<?xxx yyy>`, type `xxx` in the Target field and `yyy` in the Data field.
- 4 Click OK. The new processing instruction appears in the Structure pane.

XML and tables

InDesign tables must conform to a specific structure. When you import an XML file, tables must follow the InDesign table model. When you export to XML, InDesign automatically converts tables to this structure.


InDesign tables consist of a single `Table` tag, which contains `Cell` tags. You can use different for the tags, but all of your tables must match that basic structure, as shown in the following example:

```
<Table>
  <Cell>first cell content</Cell>
  <Cell>second cell content</Cell>
  <Cell>third cell content</Cell>
  <Cell>fourth cell content</Cell>
  <Cell>fifth cell content</Cell>
  <Cell>sixth cell content</Cell>
</Table>
```

Use attributes to specify the number of columns, cell widths, and other information. When you export a table from InDesign to XML, these formatting attributes are created automatically in the XML file. Because the information is specific to InDesign and provides formatting information, the table attributes use the *aid* namespace (see page 19). The namespace lets you easily identify an attribute as being part of the table formatting. For details, see “Table attributes” on page 20.

Images

XML generally handles images with a linked reference. When you tag an image frame, InDesign assigns an *href* attribute that points to the referenced (linked) image file. The *href* attribute shows the file path and filename and appears in the Structure pane as follows:



The screenshot shows a tree view in the Structure pane. A folder icon is expanded to show an 'Image' item. Below the 'Image' item, a radio button is selected next to the text 'href = file://path/images/anchored_int.tif'.

The href attribute indicates the file path and file name.

InDesign can then export these *href* attributes in an XML file (as shown below), and can also process them correctly on import.

```
<Image href="file://path/images/testfile.jpg"></Image>
```

To ensure that InDesign exports images correctly, the images must be linked, not embedded. They must also be properly tagged. Only tagged content is exported, while content that is not tagged is ignored on export.

Note: *The recommended workflow is to work with linked, not embedded, graphics, so that InDesign can better handle collecting the images in a subfolder during export.*

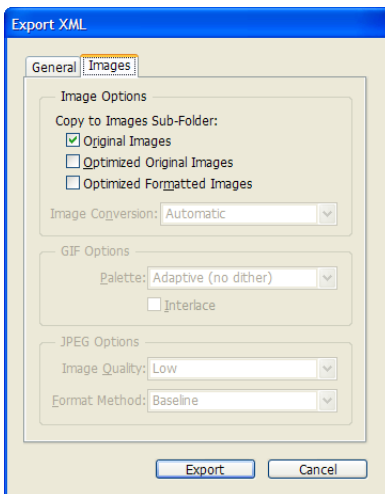
To set up images for export to XML:

- 1 Choose Window > Links to open the Links palette and verify that all of your graphics are up to date.
- 2 Use the Tags palette to assign an XML tag, such as Image, to each image. If the tag is not available in your Tags palette, create it.

When you export images to XML, the images are automatically collected in a subfolder to make it easier to move the XML file and its linked graphics. The file references in the exported XML file are then updated to point to the images in the subfolder. On export, you control the final file format of the images and whether certain images are exported.

To configure image options:

- 1 Choose File > Export.
- 2 Specify a file name for the exported file and choose XML as the format.
- 3 Click Save.
- 4 In the Export XML dialog box, click the Images tab.



Control the format of your exported images, and whether InDesign includes the original image as well.

- 5 Set the image options. For detailed instructions, see InDesign Help.
- 6 Click Export.

Understanding entity processing

When you import an XML file into InDesign, it may contain entities for special characters. For example, the ampersand uses the **&#amp;** entity. When you import an XML file, the entity is translated into the appropriate special character.

Text entities refer to text defined in the XML file or in the DTD. For example, an XML file might contain a reference to a shared copyright statement:

```
<WhitePaper>
<!ENTITY copyright "Copyright (c) 2008 Adobe Systems Incorporated
All Rights Reserved." >
white paper content...
&copyright;
</WhitePaper>
```

The entity definition is in the **<!ENTITY ...>** section

The entity reference starts with an ampersand (&).

When you import XML, text entities are expanded. You would not see the ©right; placeholder in your InDesign document. Instead, you see the copyright entity text.

XML files can also contain entity references that point to other files. Entity references look similar to the copyright example above, but the entity definition specifies a file instead of text.

```
<!ENTITY copyright SYSTEM "copyright.txt">
```

These entities are also expanded on import. InDesign does not preserve the link to the original entity file.

Note: Unparsed entity references are not processed by InDesign. For images, use an href attribute that contains the file name instead. If necessary, transform the XML file with XSLT to create the required structure.

Planning an XML workflow

When you import an XML file, the file structure appears in the Structure pane, but the content may not initially appear in your document. You must explicitly assign XML tags to placeholder text frames to automate flowing XML content into layouts on import. Assigning XML tags to text frames after importing an XML file will not cause the content to appear in those frames.

InDesign provides several different ways of working with XML content. The best approach depends on your workflow:

- For a one-time project, create a blank InDesign document, import the XML file, and then format the content as needed. This approach is suited for small, unique projects.
- To lay out the same type of XML content repeatedly, import the XML file into an InDesign document that contains paragraph and character styles. Then, associate the XML tags with the styles in the InDesign document. This approach is faster than applying styles manually, but requires more planning. If you have several documents that require the same layout, this workflow will save you time.
- If the XML content contains repeating elements that need to be formatted identically, format a sample element set and then clone that formatting for the remainder of the XML file. This approach is very efficient for highly structured content, such as directories or variable data publishing.
- Instead of associating XML tags with styles using the InDesign mapping feature, embed style names directly into the XML file using attributes. When you import the XML file into an InDesign layout that includes those defined styles, InDesign associates the XML tags with the specified styles automatically. If you control the content of the XML file, this approach provides additional automation.

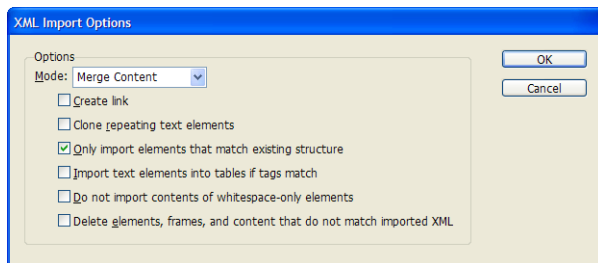
Importing XML files

InDesign offers a number of XML import options that enable you to more finely control how XML content is placed into your documents. When you import an XML file into InDesign, the Structure pane displays the XML content. You can then drag and drop the content into frames or directly onto the page, depending on how you've set up your layout. Alternatively, if you have prepared the InDesign document with tagged text and graphics frames that match the structure of the imported XML, InDesign displays the XML content in the Structure pane and flows it into the tagged layout automatically.

You can exclude certain elements by name, which allows you to filter XML information and display only what you need in the InDesign file. You have the option of linking or embedding the XML content.

To control how XML content is imported:

- 1 Choose File > Import XML.
- 2 Navigate to and select the XML file you want to import.
- 3 If necessary, select Show XML Import Options, and then click Open.



Control the behavior of XML import with these options.

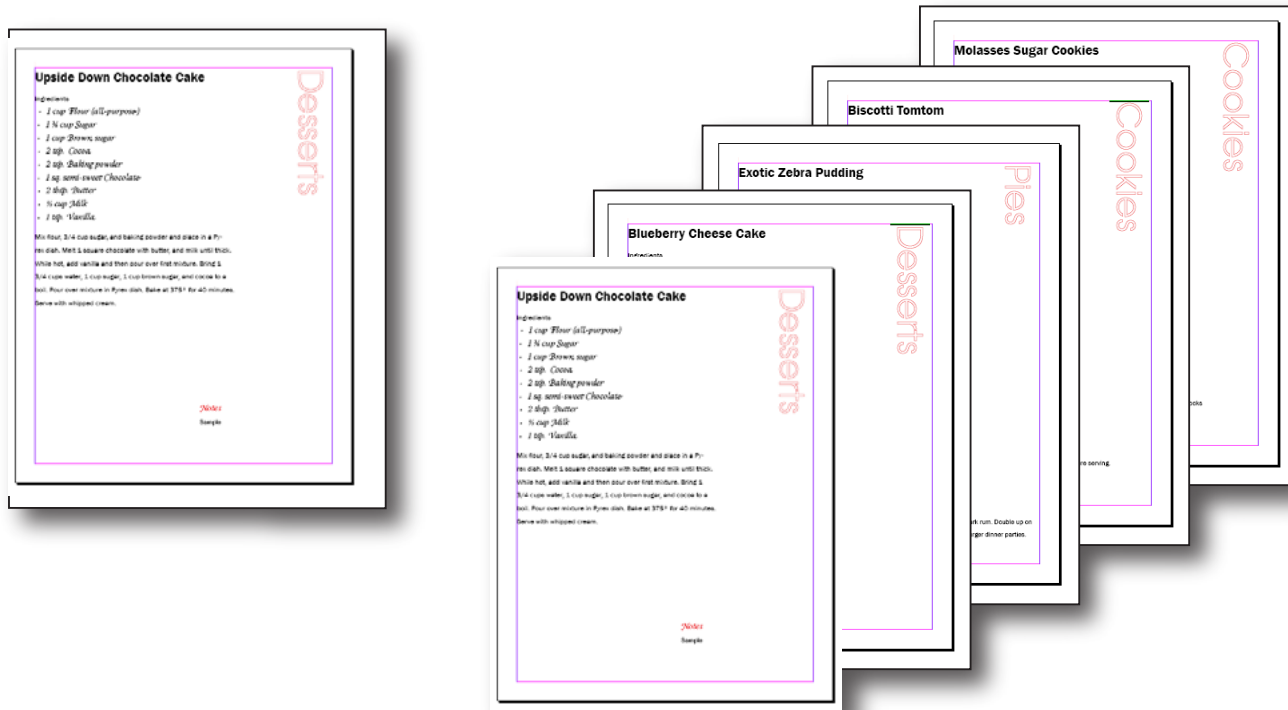
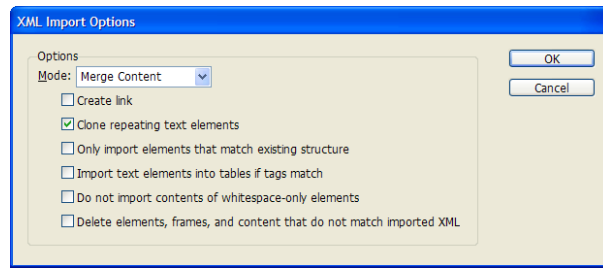
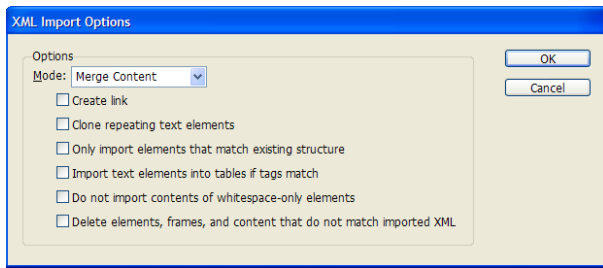
- 4 Select the import options you want.

- *Create link*. Sets up a pointer to the source XML file, which is useful if the XML file is updated regularly and you want to ensure that your InDesign file remains up-to-date.

Links to an XML file behave like other text and graphics links—if the linked file is updated outside of InDesign, InDesign refreshes it. There is, however, one complication: Because you control how information is displayed by associating the XML tags with InDesign formatting, changing the structure of the XML file may have unexpected results. For example, if a new XML tag appears in the linked file, you may need to modify the InDesign file to accommodate that tag.

- *Clone repeating text elements*. Select this option to replicate the formatting in the current file for several elements.

InDesign lets you clone a layout for repeating text elements. You can create a single formatting instance (in this example, for a recipe), and then reuse that layout to create pages for all of the other content.



Automatically generate repetitive layouts and populate it with your content using the efficient cloning import option.

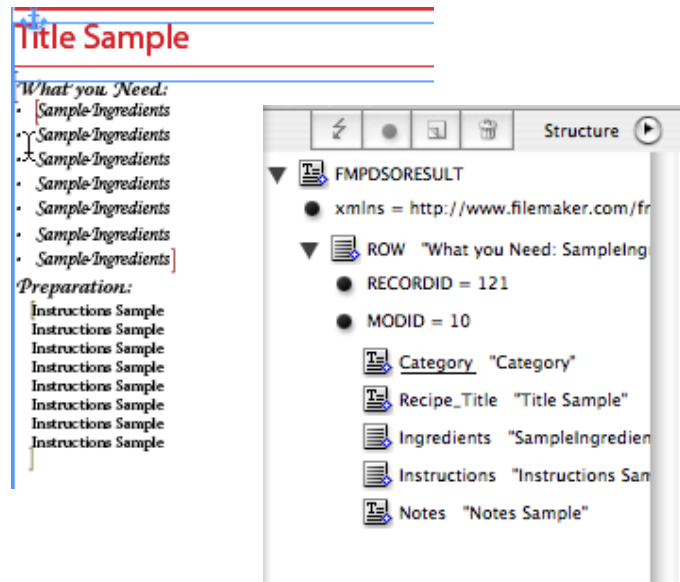
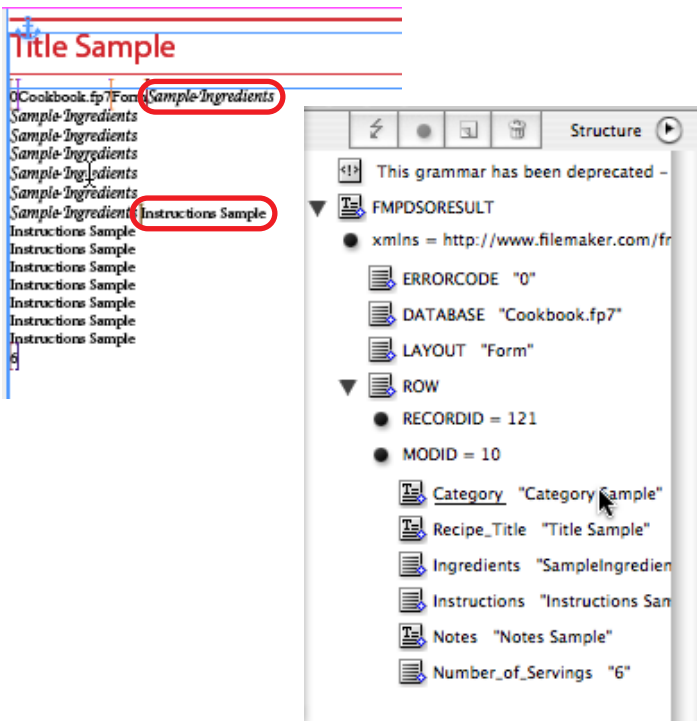
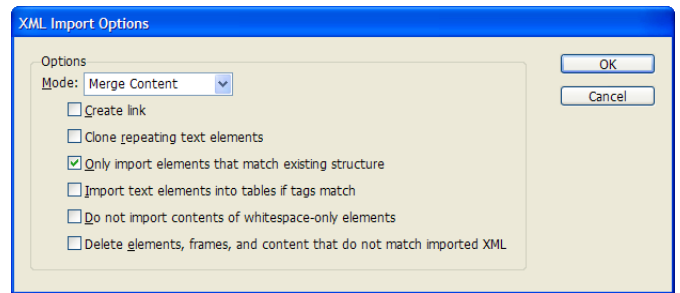
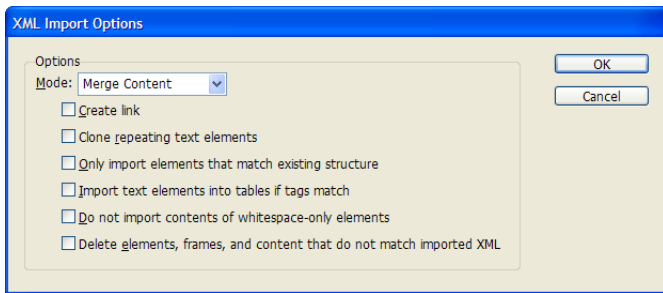
- *Only import elements that match existing structure.* If the XML file contains elements that you do *not* want to show in your layout, and your tagged InDesign file does not use elements of the same name, select this option. Any elements that do not match a tag used in the InDesign document will not be imported into InDesign.

```

- <!--
- → This grammar has been deprecated - use FMPXMLRESULT instead
-->
- <FMPDSORESULT>
- → <ERRORCODE>0</ERRORCODE>
- → <DATABASE>Cookbook.fp7</DATABASE>
- → <LAYOUT>Form</LAYOUT>
- <ROW MODID="10" RECORDID="121">
  <Category>Category Sample</Category>
  <Recipe_Title>Title Sample</Recipe_Title>
  - <Ingredients>
    SampleIngredients SampleIngredients SampleIngredients SampleIngredi
  </Ingredients>
  - <Instructions>
    Instructions Sample Instructions Sample Instructions Sample Instructions
  </Instructions>
  <Notes>Notes Sample</Notes>
- → <Number_of_Servings>6</Number_of_Servings>
- </ROW>
- </FMPDSORESULT>

```

The XML source file contains several items that are not needed in the InDesign document (as indicated by the red arrows calling out lines in the XML source above).



The additional items appear in the InDesign document.

Items that do not match InDesign structure are removed.

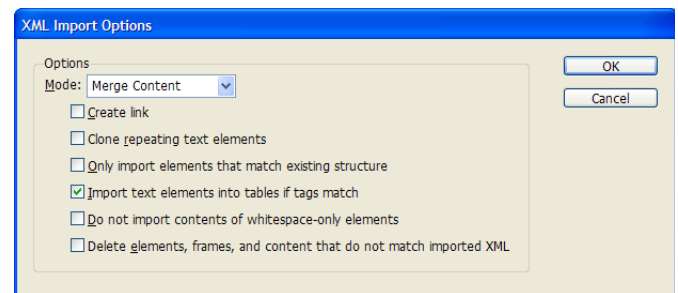
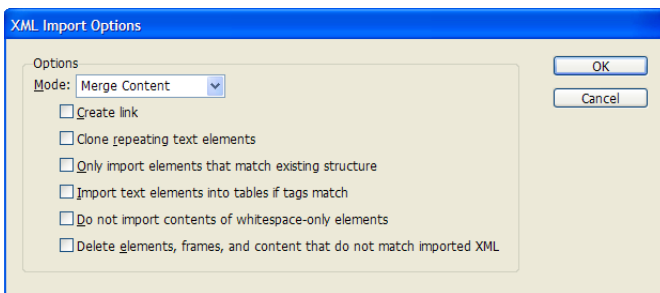
- *Import text elements into tables if tags match.* If your XML file contains a table whose tagging matches the table tagging in the InDesign document, select this option. InDesign inserts the content of the XML table into the InDesign table, even if the XML table is missing table-specific attributes. For details on table attributes, see page 20.

```

- <Root>
- <Story>
- <Table>
  <Cell>A Table of Watches</Cell>
  <Cell>Watch Model</Cell>
  <Cell>Description</Cell>
  <Cell>Price</Cell>
  <Cell>Info</Cell>
  <Cell>SE-1000</Cell>
  <Cell>Underwater wrist watch with built-in GPS.</Cell>
  <Cell>$29.95 </Cell>
- <Cell>
  Lore te volobor incil dolor alis et lutem ad dit voluptat.
</Cell>
  <Cell>SE-1200</Cell>
- <Cell>
  Underwater wrist watch with built-in GPS and Windstar navigation system
</Cell>
  <Cell>$39.95 </Cell>
- <Cell>
  Lore te volobor incil dolor alis et lutem ad dit voluptat.
</Cell>
  <Cell>SE-2999</Cell>
  <Cell>Underwater wrist watch with built-in GPS.</Cell>
  <Cell>$129.95 </Cell>
- <Cell>
  Lore te volobor incil dolor alis et lutem ad dit voluptat.
</Cell>
  <Cell>DE-4000</Cell>
- <Cell>
  Gold wrist watch with built-in GPS and Windstar navigation system
</Cell>
  <Cell>$339.95 </Cell>
- <Cell>
  Lore te volobor incil dolor alis et lutem ad dit voluptat.
</Cell>
</Table>
</Story>
</Root>

```

The XML file contains a basic table.

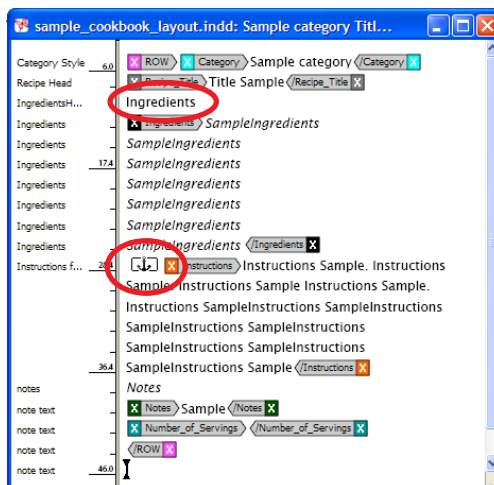
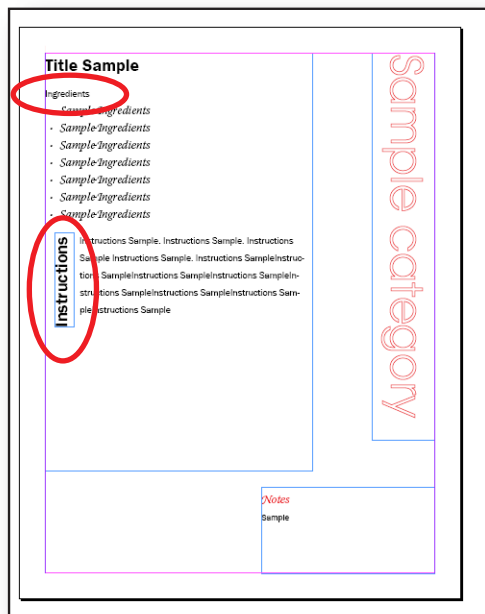


A Table of Watches
 Watch Model Description Price Info
 SE-1000 Underwater wrist watch with built-in GPS. \$29.95 Lore te volobor incil dolor alis et lutem ad dit voluptat.
 SE-1200 Underwater wrist watch with built-in GPS and Windstar navigation system \$39.95 Lore te volobor incil dolor alis et lutem ad dit voluptat.
 SE-2999 Underwater wrist watch with built-in GPS. \$129.95 Lore te volobor incil dolor alis et lutem ad dit voluptat.
 DE-4000 Gold wrist watch with built-in GPS and Windstar navigation system \$339.95 Lore te volobor incil dolor alis et lutem ad dit voluptat.

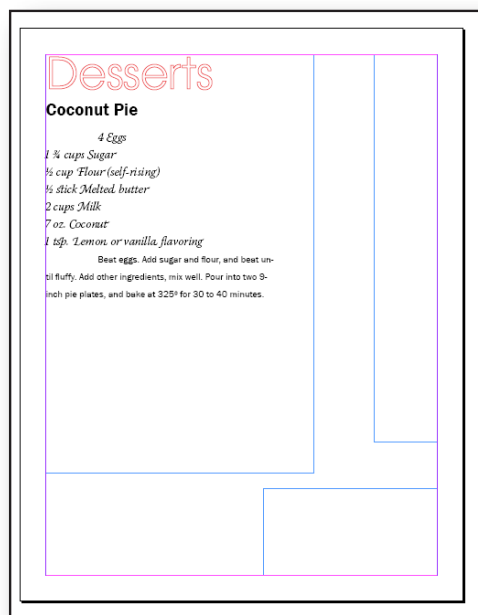
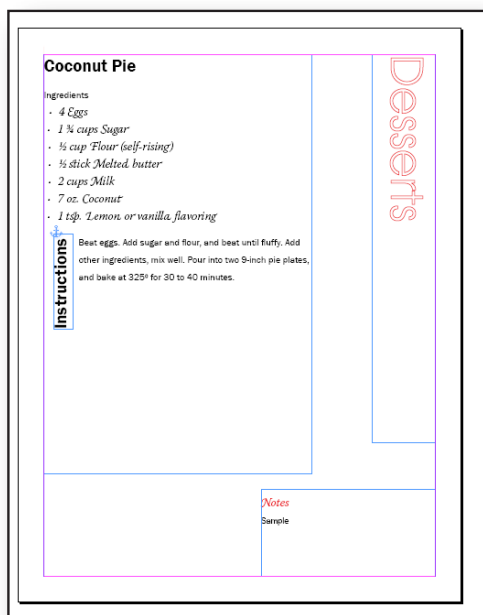
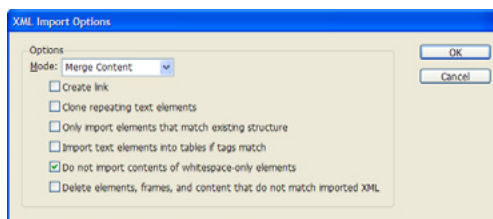
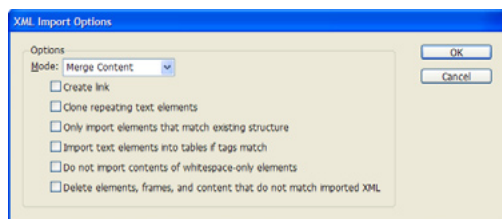
Watch Model	Description	Price	Info
SE-1000	Underwater wrist watch with built-in GPS.	\$29.95	Lore te volobor incil dolor alis et lutem ad dit voluptat.
SE-1200	Underwater wrist watch with built-in GPS and Windstar navigation system	\$39.95	Lore te volobor incil dolor alis et lutem ad dit voluptat.
SE-2999	Underwater wrist watch with built-in GPS.	\$129.95	Lore te volobor incil dolor alis et lutem ad dit voluptat.
DE-4000	Gold wrist watch with built-in GPS and Windstar navigation system	\$339.95	Lore te volobor incil dolor alis et lutem ad dit voluptat.

On import, InDesign can automatically generate a basic table from the XML content, which you can then style. Or you can import the content into a preexisting styled table (shown here on the right).

- Do not import contents of whitespace-only elements. If you want InDesign to ignore empty elements in the XML file, select this option. This option allows you to insert text, carriage returns, and anchored objects—and even to preserve these items when you update a linked XML file—providing they are inserted into elements that did not have any content in the imported XML file.



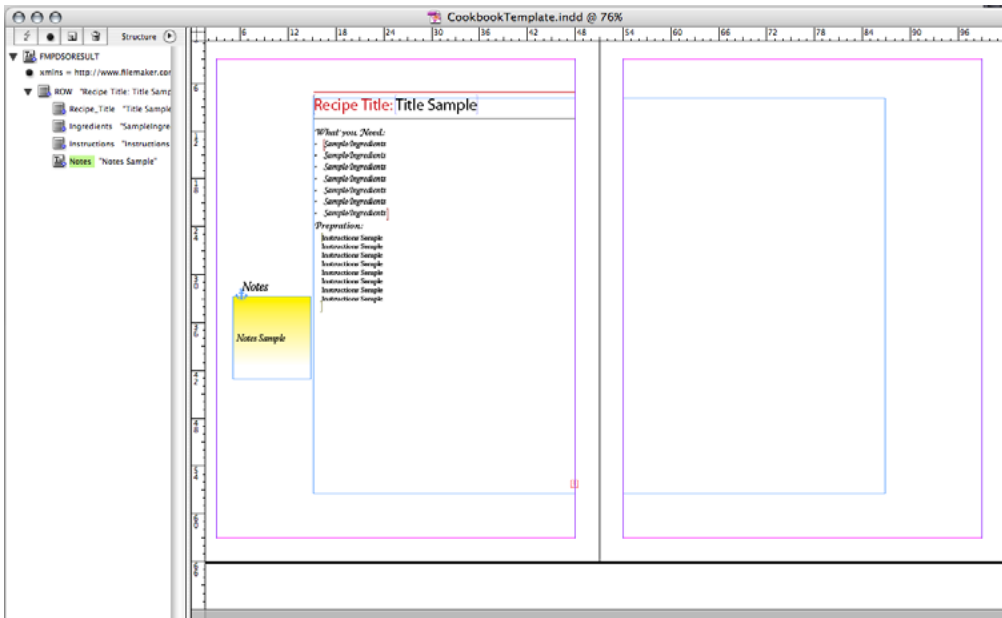
The “Ingredient” and “Instructions” labels are embedded in the template file. In the Story Editor, you can see that the labels are positioned outside the tags that have content.



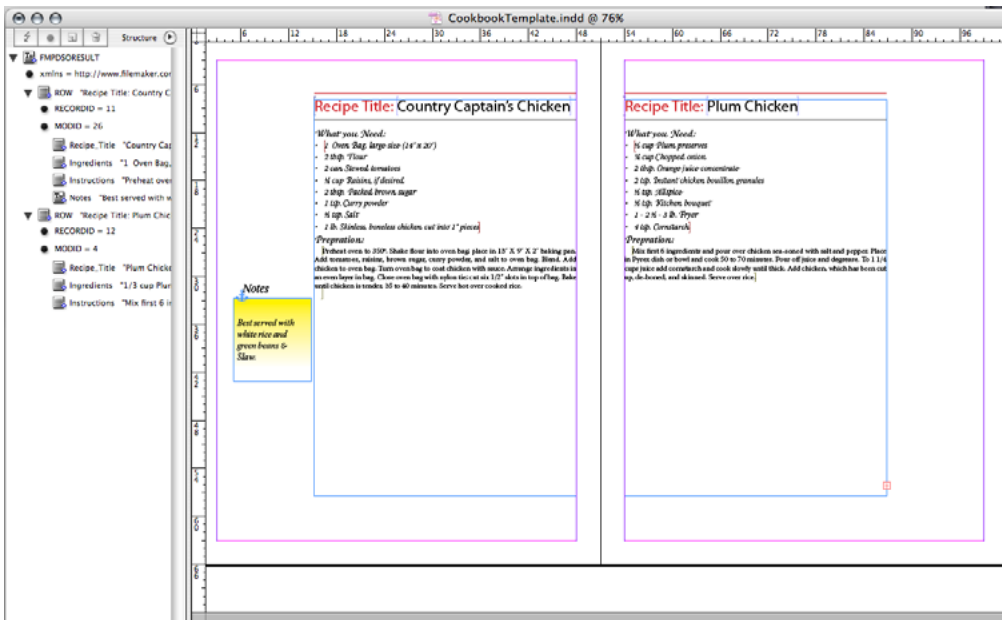
When whitespace-only elements are imported, the labels are deleted and extra tabs and paragraph returns appear in the document.

Content of whitespace-only elements is not imported. “Ingredients” and “Instructions” labels are preserved.

- *Delete elements, frames, and content that do not match imported XML.* If your XML has elements that occur in only some parts of the content, and are shown in the layout, select this option. On import, if there is an InDesign element or frame associated with an element that does not appear in the incoming XML file, the InDesign items will be deleted.

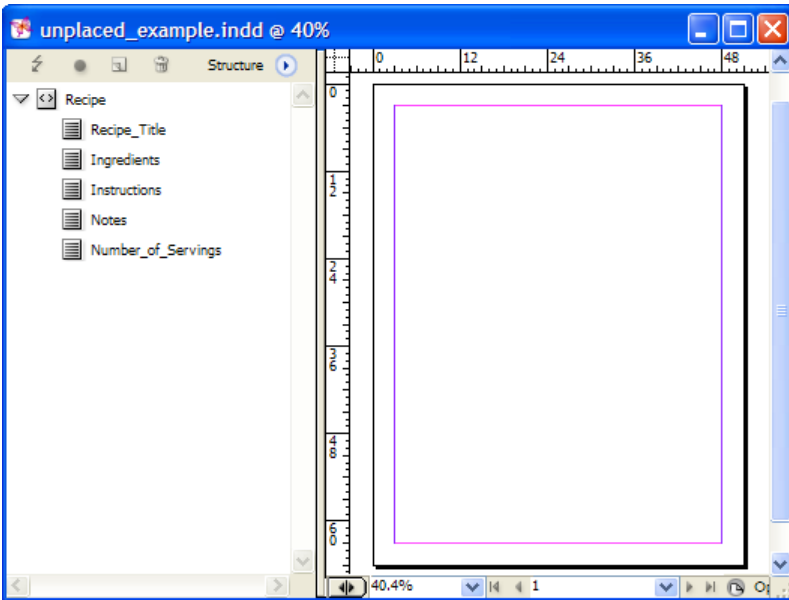


The template contains a placeholder for notes.



When you import XML content, the Notes text frame is deleted from the second recipe page because that recipe does not contain a Notes element.

When the XML file is imported, InDesign displays it in the Structure pane. Depending on whether your layout includes tagged frames that match the imported XML, the content may also appear automatically in the layout.



Imported XML content appears in the layout window only if you've set up tagged frames that match the XML structure.

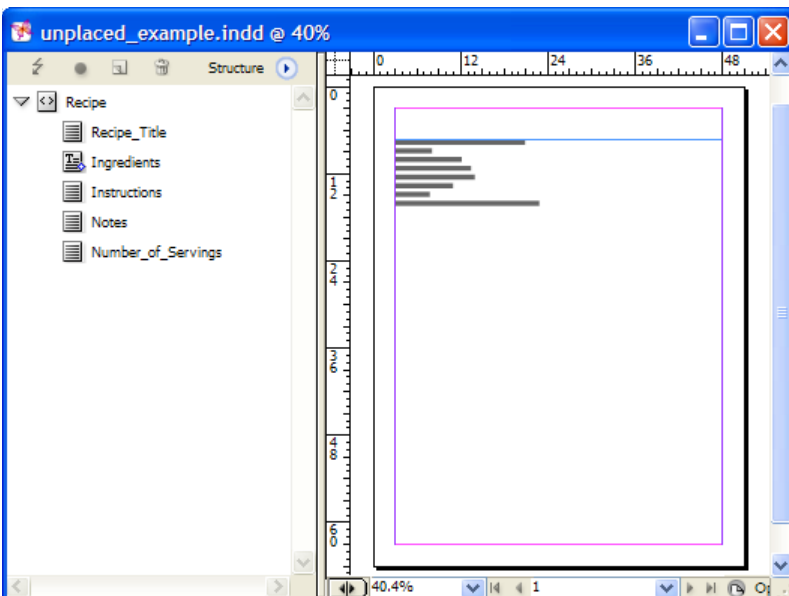
Formatting XML text content manually

After importing an XML file, you can lay out text elements by dragging and dropping them them onto the page, which creates and fills the frames on the fly, or by dragging and dropping them into existing frames. You can then format those text elements manually just as you would any other text content. Alternatively, you can map XML elements to paragraph and character styles to format these elements automatically. For details, see the next section.

Manual formatting is most useful if you rarely receive XML content, or if the structure of the XML files is unpredictable. It requires very little setup or planning, and you can format the content at the paragraph or character level. If you link to the XML file, you can preserve any formatting you've applied if you need to update the linked content.

To lay out and format XML text content manually:

- 1 In the Structure pane, locate the element you want to lay out.
- 2 Drag and drop the element to a frame inside your document. If you drop the content on a blank page, a default text frame is created for it.



Content that is visible on the page has a special icon in the Structure pane. Note that placed content (Ingredient) uses a different icon than unplaced content (all other elements).

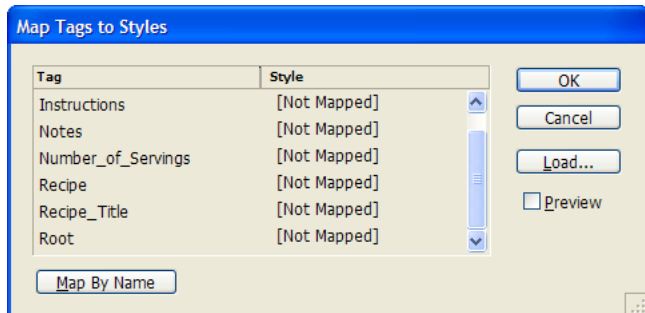
The selected element and all of its descendant elements are displayed in the text frame. You can now apply text formatting to the XML content using the Character, Paragraph, or Control palettes, just as you would to any other text in InDesign. For information about applying text formatting, see InDesign Help.

Formatting XML text using style mapping

Instead of formatting text manually, you can map XML elements to paragraph and character styles. This approach works well when you have predictable element names in the XML content, and have created a template that contains the required paragraph and character styles.

To map XML elements to paragraph and character styles:

- 1 Import the XML file and lay out the elements you want in your document.
- 2 In the Structure pane menu, select Map Tags to Styles.



Map tags to apply formatting automatically

- 3 You can map tags in three different ways:

- If the XML tag names match the InDesign style names, click Map by Name.
- If you have an existing InDesign document with the desired style mappings set up, click Load and select that file.
- If you need to map tags individually, click the corresponding style field (the default value is Not Mapped) and then choose a style from the pop-up list.

- 4 Click OK. The mappings you specify are applied to your document.

Note: Mapping tags to style names, and vice-versa, is exacting. For example, the process is case sensitive, so a tag called Body (with a capital B) will not automatically map to a style called body (with a lowercase b).

Formatting text using namespace attributes in import and export

InDesign supports special formatting attributes, which let you embed a reference to a paragraph or character style in an XML file. When the XML file is imported and laid out, InDesign uses the referenced style to format that element.

Special-purpose attributes in InDesign use an *aid:* prefix, which indicates a namespace.

Note: For an explanation of namespaces, see the W3C Recommendation at www.w3.org/TR/xml-names11/.

Your XML file must declare the *aid* namespace, and then reference the paragraph or character style you want to use with a *pstyle* or *cstyle* attribute, as shown in the following example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE Recipe SYSTEM "recipe.dtd">
<Recipe xmlns:aid="http://ns.adobe.com/AdobeInDesign/4.0/">
...
</Recipe>
```

The Recipe element contains the namespace declaration, starting with *xmlns*. The namespace declaration is required.

Paragraph and character style attributes

For text formatting, you use the *aid:pstyle* and *aid:cstyle* attributes to specify paragraph and character styles, respectively. The following example shows these attributes:

```
...
<Recipe_Title aid:pstyle="title">Country Captain's Chicken
</Recipe_Title>
<Ingredients aid:pstyle="ingredients">1 sOven Bag, large size (14" x 20")
2 tbsp. Flour
2 can Stewed tomatoes
1/4 cup Raisins, <Emphasis aid:cstyle="italics">if desired</Emphasis>
2 tbsp. Packed brown sugar
1 tsp. Curry powder
1/2 tsp. Salt
1 lb. Skinless, boneless chicken cut into 1" pieces</Ingredients>
...
```

The *aid:pstyle* attribute indicates that the `Recipe_Title` element uses the title paragraph style.

For character styles, use the *aid:cstyle* attribute.

When you import this content, InDesign applies the specified paragraph and character styles. You can define the style's attributes either before or after importing the XML content. If your document does not contain the specified styles, InDesign creates and applies default styles using your specified style names when you import the XML, and you can then edit those styles afterwards.

One advantage of this approach is that a single XML element does not have to be formatted with a single style for the entire document. In complex structures, a single element (such as `Title`) may use different paragraph styles when it occurs in different positions in the structure. Use the *aid:pstyle* attributes to specify which style is required in each location, as shown in the following example:

```
<topic>
  <title aid:pstyle="h1">Topic title</title>
  <para>This first-level topic uses an h1 paragraph style for the title.</para>
  ...
  <topic>
    <title aid:pstyle="h2">Subtopic title</title>
    <para>This second-level, nested topic uses an h2 paragraph style for the
    title.</para>
    ...
  </topic>
</topic>
```

Table attributes

The *aid* namespace attributes for tables are described in the following table:

ATTRIBUTE	VALUE	DESCRIPTION
table	table cell	Specifies a table-type element. A value of "table" indicates the container table element; a value of "cell" indicates a cell element.
trows	<i>Numeric</i>	Specifies the number of rows in the table. Used only in the Table element.
tcols	<i>Numeric</i>	Specifies the number of columns in the table. Used only in the Table element.
thead	<i>Empty</i>	If present, the <i>thead</i> attribute indicates that the current cell is part of a table header row.
crow	<i>Numeric</i>	Specifies how many rows the current cell spans. The default is 1.

ATTRIBUTE	VALUE	DESCRIPTION
ccols	<i>Numeric</i>	Specifies how many columns the current cell spans. The default is 1.
ccolwidth	<i>Numeric</i>	Specifies the width, in points, of the current cell.
tfooter	<i>Empty</i>	If present, the <i>tfooter</i> attribute indicates that the current cell is part of a table footer row.

A typical table looks something like the following example.

Table header 1	Table header 2	Table header 3	Table header 4
A	B	C	D
E straddles two columns		F	G
H	I	J straddles two rows	K
L	M		N
Table footer 1	Table footer 2	Table footer 3	Table footer 4

Sample table. The XML code for the table is shown below.

Note: The code example that follows uses line breaks and whitespace to improve readability. Important sample code is shown in bold. In your text editor, the XML may look different.

```
<Table xmlns:aid="http://ns.adobe.com/AdobeInDesign/4.0/" aid:table="table"
aid:trows="6" aid:tcols="4">
  <Cell aid:table="cell" aid:thead="" aid:crows="1" aid:ccols="1"
aid:ccolwidth="130">Table header 1</Cell>
  <Cell aid:table="cell" aid:thead="" aid:crows="1" aid:ccols="1"
aid:ccolwidth="59">Table header 2</Cell>
  <Cell aid:table="cell" aid:thead="" aid:crows="1" aid:ccols="1"
aid:ccolwidth="130">Table header 3</Cell>
  <Cell aid:table="cell" aid:thead="" aid:crows="1" aid:ccols="1"
aid:ccolwidth="130">Table header 4</Cell>

  <Cell aid:table="cell" aid:crows="1" aid:ccols="1" aid:
ccolwidth="130">A</Cell>
  <Cell aid:table="cell" aid:crows="1" aid:ccols="1" aid:ccolwidth="59">B
</Cell>
  <Cell aid:table="cell" aid:crows="1" aid:ccols="1" aid:ccolwidth="130">C
</Cell>
  <Cell aid:table="cell" aid:crows="1" aid:ccols="1" aid:ccolwidth="130">D
</Cell>

  <Cell aid:table="cell" aid:crows="1" aid:ccols="2">E straddles two
columns</Cell>
  <Cell aid:table="cell" aid:crows="1" aid:ccols="1" aid:ccolwidth="130">F
</Cell>
  <Cell aid:table="cell" aid:crows="1" aid:ccols="1" aid:ccolwidth="130">G
</Cell>

  <Cell aid:table="cell" aid:crows="1" aid:ccols="1" aid:ccolwidth="130">H
</Cell>
  <Cell aid:table="cell" aid:crows="1" aid:ccols="1" aid:ccolwidth="59">I
</Cell>
  <Cell aid:table="cell" aid:crows="2" aid:ccols="1" aid:ccolwidth="130">J
straddles two rows</Cell>
  <Cell aid:table="cell" aid:crows="1" aid:ccols="1" aid:ccolwidth="130">K
</Cell>
```

The *aid:trows* and *aid:tcols* attributes specify the number of rows and columns in the table.

The empty *aid:thead* attribute indicates a heading cell. Footing cells use a similar *aid:tfooter* attribute. Regular table cells do not have an attribute to indicate what type they are.

The *aid:ccols* attribute indicates that the current cell spans the specified number of columns. The default, 1, indicates no column spanning. The *aid:crows* attribute indicates row spanning.

```

<Cell aid:table="cell" aid:crows="1" aid:ccols="1" aid:ccolwidth="130">L
</Cell>
<Cell aid:table="cell" aid:crows="1" aid:ccols="1" aid:ccolwidth="59">M
</Cell>
<Cell aid:table="cell" aid:crows="1" aid:ccols="1" aid:ccolwidth="130">N
</Cell>

<Cell aid:table="cell" aid:tfooter="" aid:crows="1" aid:ccols="1"
aid:ccolwidth="130">Table footer 1</Cell>
<Cell aid:table="cell" aid:tfooter="" aid:crows="1" aid:ccols="1"
aid:ccolwidth="59">Table footer 2</Cell>
<Cell aid:table="cell" aid:tfooter="" aid:crows="1" aid:ccols="1"
aid:ccolwidth="130">Table footer 3</Cell>
<Cell aid:table="cell" aid:tfooter="" aid:crows="1" aid:ccols="1"
aid:ccolwidth="130">Table footer 4</Cell>
</Table>

```

If you are importing XML tables and want InDesign to format your table automatically, you need to create XML content that contains the *aid* namespace attributes.

Note: If your XML content uses a different table model, consider using Extensible Stylesheet Language (XSL) transformations to modify the content before import. For details about XSL, see www.w3.org/Style/XSL/

Ruby attributes (for Japanese)

For Japanese text, a ruby attribute is available. InDesign stores ruby (pronunciation guides) in the *aid:ruby* attribute. In the InDesign file, ruby are displayed next to the kanji characters for which they provide pronunciation cues. You can set up ruby one character at a time, using the MonoRuby tag, or in a group with the GroupRuby tag.

A MonoRuby example is shown below:

```

<MonoRuby>
  <aid:ruby xmlns:aid="http://ns.adobe.com/AdobeInDesign/3.0/">
    <aid:rbc>
      <aid:rb>夏</aid:rb>
      <aid:rb>目</aid:rb>
      <aid:rb>漱</aid:rb>
      <aid:rb>石</aid:rb>
    </aid:rbc>
    <aid:rtc>
      <aid:rt>なつ</aid:rt>
      <aid:rt>め</aid:rt>
      <aid:rt>そう</aid:rt>
      <aid:rt>せき</aid:rt>
    </aid:rtc>
  </aid:ruby>
</MonoRuby>

```



The namespace declaration is required.

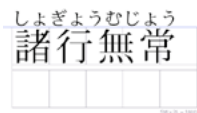
The *aid:rbc* element contains *aid:rb* elements. Individual kanji are stored inside the *aid:rb* elements.

The *aid:rtc* element contains *aid:rt* elements. Ruby text is stored inside the *aid:rt* elements.

```

<GroupRuby>
  <aid:ruby xmlns:aid="http://ns.adobe.com/AdobeInDesign/3.0/">
    <aid:rb>諸行無常</aid:rb>
    <aid:rt>しよぎようむじよう</aid:rt>
  </aid:ruby>
</GroupRuby>

```



Inside the GroupRuby element, you place all kanji in a single *aid:rb* element and all ruby text in an *aid:rt* element.

To position repeating XML content using anchored objects:

- 1 Choose View > Structure > Show Tag Markers and View > Structure > Show Tagged Frames to make the frames easier to see.
- 2 Choose File > Import to import an XML file.
- 3 Create the initial layout by dragging and dropping XML elements from the Structure pane into text and graphics frames on the page (or use tagged frames to automatically flow the XML content).
- 4 Create an untagged text or graphics frame using the Type, frame, or drawing tools.
- 5 Drag and drop the appropriate XML element onto your new text or graphics frame.
- 6 Select the frame and choose Edit > Cut.
- 7 With the Type tool, click an insertion point in the tagged text where you want to anchor the object. Then choose Edit > Paste.

Note: Make sure your cursor is positioned at a point in the text that maps to the point in the Structure pane where you would like the XML element to be inserted.

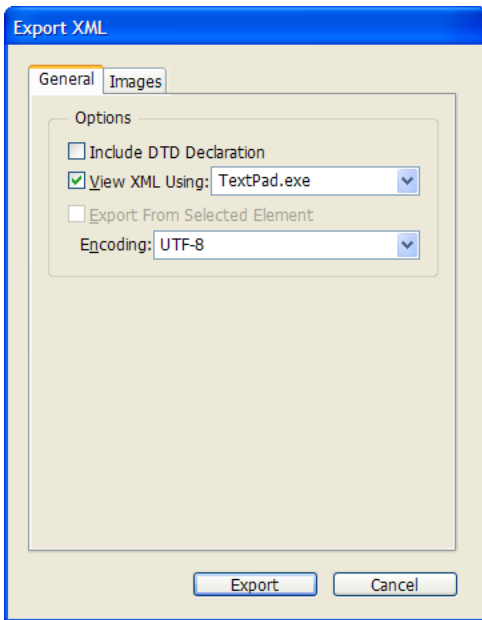
- 8 In the Structure pane, note that InDesign created an extra element when you pasted the frame. Delete the original element and keep the newly created element, which is now properly anchored. To find out which element maps to the newly created element, right-click one of the elements in the Structure pane and choose Go To Element. InDesign will then select that element in the layout. When you have the right one selected, delete it.
- 9 Choose Object > Anchored Object > Options to specify the positioning for the anchored frame.
- 10 Save the resulting InDesign document as a template for automating this layout. Then, when importing XML content into the template, choose Clone Repeating Text Elements in the XML Import Options dialog box.

Exporting XML

You can export XML from any InDesign file. InDesign exports only the information that is tagged in the Structure pane. Content in untagged text frames, embedded images, and untagged images are not exported.

To export XML:

- 1 Make sure that all of the information you want to export is tagged. (If necessary, use the Tags palette to assign tags.)
- 2 (Optional) If you want to export only information below a specific element, select that element in the Structure pane.
- 3 Choose File > Export.
- 4 Specify a file name and location and select XML as the file type. Click Save.
- 6 Set the export options:
 - To include a reference to your DTD in the XML file, select the corresponding option.
 - To display the XML after it is created, select View XML Using and specify the editor of your choice.
 - To export starting with the element you selected in step 2, select Export From Selected Element.
 - Choose a text encoding method. The default is UTF-8.



The export options provide control over XML output.

Note: For information about image export options, see “Images” on page 9.

5 Click Export.

XML files use Unicode encoding. If you try to open the XML file in an editor or viewer that does not support Unicode, you may see error messages or character substitution in your output XML, so be sure to use a Unicode-compliant editor. If you plan to transform the XML files into HTML for display on the web, consider removing special characters by running a script in InDesign before you export to XML. You can also write an XSL transformation that processes the XML file to remove special characters. XML and InDesign support a wider range of characters than HTML does.

For more details, see *Real-world Case Studies: Getting XML from Adobe InDesign Layouts for Repurposing* on the Adobe website at www.adobe.com/products/indesign/pdfs/realworld_XML.pdf.

Using InDesign Interchange (INX) files in XML workflows

The InDesign Interchange format is an external representation of the InDesign document object model. It preserves complete document information, including content, formatting, geometry, and XML tagging, for transparent interchange. This format supports a number of InDesign CS2 features, including backwards compatibility with InDesign CS, InDesign snippets, and the assignments workflow with InCopy CS2. This section focuses on how the InDesign snippets feature can support your XML publishing workflows.

A snippet is an InDesign object or group of objects that have been exported from a document to share with others or re-use in your workflow. These files might be comprised of text, imported graphics, native InDesign objects, styled tables—whatever you can lay out on an InDesign page.

You create a snippet by selecting the objects you want to reuse and then dragging and dropping them onto your desktop, into Adobe Bridge, or into a Library palette, or by choosing File > Export and choosing InDesign Snippet for Format. When you place or drag a snippet back into a layout, InDesign recreates the original objects, their formatting, their relative positioning on the page, and any XML tags that have been applied. (A snippet is an XML wrapper on a series of InDesign scripting calls that direct InDesign how to reproduce the file.)

You can integrate snippets into an XML workflow to achieve more production efficiency. Here is an example of how this might work: Imagine that you update and publish tables of data daily or weekly. For example, you might publish sports scores, top 10 lists, or weather information in tables. The basic design of those tables doesn't vary, though you might freshen the color scheme regularly. One way to streamline this process would be to use snippets.

You simply design and tag the tables, link them to the XML data you publish, and then export each of them as a snippet. The snippet preserves the XML tags and the linked data. When you drag and drop one of the snippets into another InDesign layout, you can then tweak the design, update the link to the XML data, and publish the results. The entire process can happen in a few quick steps.

For more information about working with InDesign snippets, see the InDesign online Help system.

Note: The INX format is not intended to be human readable or to be used as a long-term archiving format. We do not recommend trying to create or edit INX files using a text editor or storing snippets in a database.

Where to go from here

The combination of XML and InDesign offers you the best of both worlds—the structure, predictability, and automation of XML along with the powerful InDesign layout engine that lets your creativity shine. With strong support for XML as an input or output format, InDesign provides you with many possibilities for automating layout and reusing content.

To start, you may want to experiment with a small amounts of content. You might, for example, export some articles to XML for reuse on a website or consider whether some database information could be linked into your publication automatically.

Once you have configured the basic publishing system, you can use scripting to further automate and streamline your XML publishing. Virtually everything in InDesign is scriptable. You can write scripts in JavaScript (for cross-platform compatibility), VBScript, or AppleScript. For details about scripting, see the *InDesign Scripting Guide* and sample scripts on your installation CD. You can also get more information on the Adobe website at www.adobe.com/products/indesign/scripting.html.

The InDesign area on the Adobe website at www.adobe.com/products/indesign/ provides links to additional white papers, customer success stories, training and consulting resources, third-party plug-ins, user groups, and other useful resources.

Appendix: XML syntax

XML files are required to follow certain syntax rules. When a file follows the rules, it is *well-formed*.

The syntax rules include the following:

- All tags must be opened and closed.

```
<p>This paragraph has a closing tag.</p>
```

```
<p>This is not allowed.<p>The second paragraph starts without a closing tag  
for the first.</p>
```

- Empty tags must include a closing slash before the end of the tag, as shown in these examples:

```
<br/>
```

```

```

- Tags must be properly nested.

```
This text is <b><i>bold and italic</i></b>.
```

```
This text uses <b><i>crossed</b></i> tags, which are not allowed.
```

- All attribute values must be quoted.

```
<p align="center">This paragraph has a center attribute</p>
```

```
<table border=0>This table needs quotes around the border value.</table>
```

- Tag names cannot contain spaces or special characters and must start with an alphabetic character.

```
Legal tag names: <Sect1>, <Para>, <Long _ Element _ Name>
```

```
Illegal tag names: <1Sect>, <Long Element Name>, <Asterisk*>
```

Note: For a complete discussion of XML syntax, see the XML Recommendation at www.w3.org/TR/2004/REC-xml-20040204/.

All XML files must follow the basic syntax rules.

Enforcing structure rules

XML gives you the option of specifying a set of rules for content. Files that follow the structure rules are *valid*. You declare structure rules in a document type definition (DTD).

Working with structure rules and valid files gives you the ability to define and enforce document organization. With a DTD or Schema file, you can set up requirements for your content. Conceptually, this process is similar to creating a formatting template, but with structure rules, you can *require* that documents must follow the rules you establish. You could, for example, specify that a story must begin with a title and a byline. For an academic journal, you could require an abstract at the beginning of an article. You can require that bibliographical information must be organized a certain way. Structure rules let you ensure that documents are consistent.

InDesign lets you work with or without a DTD, but you must provide a DTD if you want to validate your documents.

In XML, the DTD is declared at the top of the file, as shown here:

```
<?xml version="1.0" encoding="UTF-8" ?>  
<!DOCTYPE Recipe SYSTEM "recipe.dtd">  
<Recipe>...</Recipe>
```

The DTD is shown here:

```
<!ELEMENT Recipe (Recipe_Title, Ingredients, Instructions, Notes?,  
Number_of_Servings? ) >  
<!ELEMENT Recipe_Title (#PCDATA) >  
<!ELEMENT Ingredients (#PCDATA) >  
<!ELEMENT Instructions (#PCDATA) >  
<!ELEMENT Notes (#PCDATA) >  
<!ELEMENT Number_of_Servings (#PCDATA) >
```

For each element, the DTD specifies what information is permitted.

ELEMENT NAME	ELEMENT DEFINITION	DESCRIPTION
Recipe	Recipe_Title, Ingredients, Instructions, Notes?, Number_of_Servings?	Must contain one Recipe_Title element, followed by one Ingredients element, followed by one Instructions element, followed by zero or one Notes elements, followed by zero or one Number_of_Servings elements. The question mark (?) indicates that the Notes and Number_of_Servings elements are optional.
Recipe_Title Ingredients Instructions Notes Number_of_Servings	#PCDATA	#PCDATA indicates that the element contains text.

Where to get a DTD

If you have a client who requires delivery of XML files, the client may already have a DTD and will provide it to you. You can also create your own DTD, but before you do so, consider whether one of these standard DTDs meets your requirements.

- NewsML (www.newsml.org)
- SportsML (www.sportsml.org)
- Technical documents:
 - DocBook (www.oasis-open.org/committees/tc_home.php?wg_abbrev=docbook)
 - Darwin Information Typing Architecture (DITA) (www.oasis-open.org/committees/tc_home.php?wg_abbrev=dita)
- Chemical Markup Language (<http://cml.sourceforge.net>)
- Other DTD examples (www.service-architecture.com/xml/articles/specific_xml_vocabularies.html)

FOR MORE INFORMATION

For a comprehensive overview of InDesign, please visit www.adobe.com/products/indesign/



Adobe Systems Incorporated
345 Park Avenue, San Jose, CA 95110-2704 USA
www.adobe.com

Adobe, the Adobe logo, Acrobat, and InDesign are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Microsoft, Windows, and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. W3C is a trademark (registered in numerous countries) of the World Wide Web Consortium; marks of W3C are registered and held by its host institutions MIT, ERCIM and Kelo. All other trademarks are the property of their respective owners.

© 2005 Adobe Systems Incorporated. All rights reserved.
11/05