## Contents

1. Welcome to the Dynamic Content User Guide ........................................................................................................ 4  
   1.1 What’s New in Dynamic Content 12.1 .................................................................................................................. 4  
   1.2 Copyright Notice .................................................................................................................................................. 4

2. Getting Started .......................................................................................................................................................... 6

3. Dynamic Content Basics .......................................................................................................................................... 9  
   3.1 Introducing Dynamic Content ........................................................................................................................................... 9  
      3.1.1 Why Dynamic Content? .................................................................................................................................. 9  
      3.1.2 Benefits of Using Dynamic Content ............................................................................................................. 10  
      3.1.3 A Typical Workflow in Dynamic Content ................................................................................................ 11  
      3.1.4 XML Basics ................................................................................................................................................... 11  
   3.2 Step 1: Create an Adobe Illustrator Template ..................................................................................................... 12  
      3.2.1 Map Locale Variables (only for IPC XML standard) .................................................................................. 13  
   3.3 Step 2: Link the XML file to the Illustrator document ......................................................................................... 14  
      3.3.1 Automatic linking ........................................................................................................................................... 14  
      3.3.2 Manual linking ............................................................................................................................................... 16  
   3.4 Step 3: Work with Changes in XML Content ...................................................................................................... 18  
   3.5 Creating Design Variants (Line Extensions) ........................................................................................................ 20  
      3.5.1 XML Linking Strategies for Design Variants .............................................................................................. 20  
   3.6 Relink text elements to different languages ....................................................................................................... 22

4. Viewing XML Content in Dynamic Content ........................................................................................................ 24  
   4.1 The Dynamic Content Palette ...................................................................................................................................... 24  
      4.1.1 Smart View ..................................................................................................................................................... 24  
      4.1.2 Advanced View ................................................................................................................................................ 25  
      4.1.3 The Fly-out Menu .......................................................................................................................................... 29  
      4.1.4 The Toolbars: Button Reference .................................................................................................................. 33  
   4.2 Viewing XML Information ........................................................................................................................................ 35  
   4.3 Filter Options for the XML Tree .......................................................................................................................... 36  
      4.3.1 Applying a Filter to the XML Tree ................................................................................................................ 37  
      4.3.2 Filter Options in Detail ................................................................................................................................... 38  
   4.4 Manipulating the XML Tree ................................................................................................................................... 40

5. Managing Changes to the XML Content ................................................................................................................. 43  
   5.1 How Dynamic Content Deals with Changes ...................................................................................................... 43  
   5.2 Approving Changes in the External XML File .................................................................................................... 44  
   5.3 Applying Text Edits in Illustrator to the XML File .............................................................................................. 45  
   5.4 Creating New XML Files in Illustrator ............................................................................................................... 45  
      5.4.1 New XML ....................................................................................................................................................... 45  
      5.4.2 Add Illustrator Text to the XML .................................................................................................................. 46
1. Welcome to the Dynamic Content User Guide

Welcome to the Esko Dynamic Content user documentation. This User Guide will provide you with all the knowledge and skills you need to integrate the Dynamic Content plug-in into your daily workflow.

1.1 What’s New in Dynamic Content 12.1

This version of Dynamic Content supports both the IPC standard as the GS1 BMS (Business Message Standard).

The selection between GS1 and IPC standard can be made in the Preferences dialog of the Dynamic Art plugin or the Dynamic Content preferences.

1.2 Copyright Notice

© Copyright 2013 Esko Software BVBA, Gent, Belgium

All rights reserved. This material, information and instructions for use contained herein are the property of Esko Software BVBA. The material, information and instructions are provided on an AS IS basis without warranty of any kind. There are no warranties granted or extended by this document. Furthermore Esko Software BVBA does not warrant, guarantee or make any representations regarding the use, or the results of the use of the software or the information contained herein. Esko Software BVBA shall not be liable for any direct, indirect, consequential or incidental damages arising out of the use or inability to use the software or the information contained herein.

The information contained herein is subject to change without notice. Revisions may be issued from time to time to advise of such changes and/or additions.

No part of this document may be reproduced, stored in a data base or retrieval system, or published, in any form or in any way, electronically, mechanically, by print, photoprint, microfilm or any other means without prior written permission from Esko Software BVBA.

This document supersedes all previous dated versions.

PANTONE®. PantoneLIVE and other Pantone trademarks are the property of Pantone LLC. All other trademarks or registered trademarks are the property of their respective owners. Pantone is a wholly owned subsidiary of X-Rite, Incorporated. © Pantone LLC, 2012. All rights reserved.

This software is based in part on the work of the Independent JPEG Group.

Portions of this software are copyright © 1996-2002 The FreeType Project (www.freetype.org). All rights reserved.

Portions of this software are copyright 2006 Feeling Software, copyright 2005-2006 Autodesk Media Entertainment.

Portions of this software are copyright ©1998-2003 Daniel Veillard. All rights reserved.

Portions of this software are copyright ©1999-2006 The Botan Project. All rights reserved.
Part of the software embedded in this product is gSOAP software. Portions created by gSOAP are Copyright ©2001-2004 Robert A. van Engelen, Genivia inc. All rights reserved.

Portions of this software are copyright ©1998-2008 The OpenSSL Project and ©1995-1998 Eric Young (eay@cryptsoft.com). All rights reserved.

This product includes software developed by the Apache Software Foundation (http://www.apache.org/).

Adobe, the Adobe logo, Acrobat, the Acrobat logo, Adobe Creative Suite, Illustrator, InDesign, PDF, Photoshop, PostScript, XMP and the Powered by XMP logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Microsoft and the Microsoft logo are registered trademarks of Microsoft Corporation in the United States and other countries.

SolidWorks is a registered trademark of SolidWorks Corporation.

Portions of this software are owned by Spatial Corp. 1986 2003. All Rights Reserved.

JDF and the JDF logo are trademarks of the CIP4 Organisation. Copyright 2001 The International Cooperation for the Integration of Processes in Prepress, Press and Postpress (CIP4). All rights reserved.

The Esko software contains the RSA Data Security, Inc. MD5 Message-Digest Algorithm.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems in the U.S. and other countries.

Part of this software uses technology by BestTM Color Technology (EFI). EFI and Bestcolor are registered trademarks of Electronics For Imaging GmbH in the U.S. Patent and Trademark Office.


All other product names are trademarks or registered trademarks of their respective owners.

Correspondence regarding this publication should be forwarded to:

Esko Software BVBA
Kortrijksesteenweg 1095
B – 9051 Gent
info.eur@esko.com
2. Getting Started

To introduce you to the concept of Dynamic Content, follow the five simple steps below to create dynamic links between an Adobe Illustrator document and a very basic, non-IPC simple XML file.

1. In the Sample Files folder on the installation disk image you will find an Illustrator file called: Esko Connect More.ai. Open the sample file with Adobe Illustrator and:
   a) note the two text objects that are present in the design: Link Statement1 here and Link Statement2 here.
   b) choose Windows > Esko > Dynamic Content > Dynamic Content.

The Dynamic Content palette appears.
2. Generate a dynamic link with an XML file:
   a) click the Add XML file... button.
   b) browse for the GettingStarted.xml file included on the Sample Files folder on the installation disk image.

3. Link the text objects with the XML elements:
   a) select the text box containing Link with Statement1.
   b) in the Dynamic Content palette, expand GettingStarted.xml and the body entry, and select the entry Statement1.
c) Click the Make Dynamic Object button.

d) Repeat for Statement 2.

4. Change the text style using the normal Illustrator text tools:

After linking an XML element to a text object, its contents are protected from accidental changes, but you can still select text with the text tool and change its style (e.g. assign another typeface or text color).

5. Change the XML file.

To experience the dynamic behavior of the link that you established between the Illustrator text object and the XML file, go ahead and:

a) open the GettingStarted.xml file in a text editor of your choice (e.g. TextEdit on Mac OS, or Notepad on Windows).

b) change the content of the first text statement. For example, change Connect to Esko into Connect with Esko. If you have not changed your preferences from the default, the plug-in will detect your changes to the linked XML file, and it will ask you to update the artwork.
3. Dynamic Content Basics

In this section, you will learn the basic concepts of the Dynamic Content plug-in, and experience a simple, typical workflow that guides you through your first steps in Dynamic Content.

3.1 Introducing Dynamic Content

Esko’s Dynamic Content for Adobe® Illustrator® brings the power of XML-based automation to Adobe® Illustrator® CS5, CS5.1, CS6 and CS7 on Mac and Windows.

3.1.1 Why Dynamic Content?

Esko Dynamic Content allows packaging designers to add externally maintained content elements such as legal copy or nutritional information to a packaging design. This solves an increasingly common problem in the designer’s day to day work: human error in copied content.

The ever growing number of stock keeping units that are brought onto the market by brand owners has resulted in an explosive growth in the volume of artwork that needs to be created for all the packaging of each unit. At the same time, the designs themselves become more and more complex in their competition for the consumer’s attention.

Many designs share pieces of content. Some content is mandatory, required by regulatory instances such as the FDA, EMEA, or national bodies. Other content simply recurs on different variants of a shared design.

The end result is the same: an increasing number of human interactions, often of the "copy-paste" type, that at the same time increases the risk of errors slipping into the design. Errors in artwork are wasteful, generating rework, delays in product launches and sometimes even requiring product recalls.
3.1.2 Benefits of Using Dynamic Content

The Dynamic Content plug-in helps automate the insertion and updating of externally maintained content into a design. This prevents a number of common problems, and it saves valuable production time by eliminating repetitive manual tasks.

Some typical problems prevented by a Dynamic Content workflow:

- Typing errors;
- Wrong versions of copy text or barcodes are used in the final design;
- Cut-and-paste errors (e.g. when copy text is provided in a Microsoft Word document or PDF, and the designer has to manually copy this into the Illustrator document);
- Text overflows;
- Shared elements on design variants are not synchronised.

Because Dynamic Content allows you to keep this mandatory or shared content separated from the Illustrator document, it also eliminates repetitive and time consuming updates: simply modify the
external XML file to update the text, barcodes, and nutrition tables on the design. Review cycles have never been this easy.

### 3.1.3 A Typical Workflow in Dynamic Content

A typical Dynamic Content-based design workflow includes an exchange of information between brand owners and designers. Often, the content to be used in a design will change multiple times during the design and production process, and it is in these cases that Dynamic Content will prove its worth.

A sample workflow for the use of Dynamic Content could be the following:

<table>
<thead>
<tr>
<th>Stage</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The brand owner’s marketing department draws up the text that needs to appear on the packaging design. The text is entered into a PDF form or a content management system, which can export an XML file.</td>
</tr>
<tr>
<td>2</td>
<td>The brand owner sends the PDF form or XML file containing the text to the designer.</td>
</tr>
<tr>
<td>3</td>
<td>The designer produces a packaging design in Adobe Illustrator and links the text elements from the XML file to text objects in the Illustrator template. In the GS1 workflow, the designer can generate the &quot;Response&quot; XML and send it back to the brand owner. This Response message tells the brand owner what items were used in the artwork and how many times.</td>
</tr>
<tr>
<td>4</td>
<td>Later, the brand owner makes changes to the text, produces a new XML file and sends this to the designer.</td>
</tr>
<tr>
<td>5</td>
<td>The designer replaces the original XML file on his disk with the new version. Dynamic Content automatically detects the changes, and immediately alerts the designer. It clearly shows exactly what has changed, and where.</td>
</tr>
<tr>
<td>6</td>
<td>Based on the alerts displayed in Dynamic Content, the designer can very quickly accept automatically updated text, or make any required formatting changes. There is no need for copying, pasting, or any other manual and error-prone operations, and Dynamic Content alerts the designer to problems such as text overflow.</td>
</tr>
<tr>
<td>7</td>
<td>Thanks to Dynamic Content, stages 4 - 6 can occur repeatedly without losing time or introducing errors.</td>
</tr>
</tbody>
</table>

### 3.1.4 XML Basics

XML stands for EXtensible Markup Language, and it is a markup language similar to HTML. However, where HTML describes the layout of a web page, XML markup describes the contents and the structure of a document.

Below is an example of a basic XML document:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<body>
<p> This is a very cool product! </p>
</body>
```

XML documents are often used to exchange data or textual content.
In the context of the Dynamic Content plug-in, XML is a useful format because:

- it separates content - the text itself - from layout and presentation, which will be defined by Illustrator text objects.
- it describes and contains the structure of the text that must be used on a design.

For more information on the XML format, please refer to the Appendix “An Introduction to XML”.

3.2 Step 1: Create an Adobe Illustrator Template

Before you can start linking XML-sourced content into your work, you need to prepare a design in Adobe Illustrator.

We will refer to the Illustrator design as the "template", which will later hold the placeholder text objects in which you will later insert text from an XML file.

Dynamic Content allows you to use your complete Illustrator toolset as usual:

1. Create a design in Adobe Illustrator.
2. Create Text objects using the Illustrator tools, including:
   - the Type Tool
   - the Area Type Tool
   - the Type on a Path Tool
   - the Vertical Type Tool
   - the Vertical Area Type Tool
   - the Vertical Type on a Path Tool
   - Threaded text
   
   **Note:**
   Dynamic Content also supports Right-To-Left text in Middle-Eastern languages.

3. Create nutrition tables using the Esko Dynamic Tables plug-in, if required for the design.
4. Create barcodes using the Esko Dynamic Barcodes plug-in, if required for the design.
5. Add the text objects, tables, barcodes, Illustrator symbols, and external images to the Dynamic Art palette. If you do, it will be much easier for the person linking the XML content to find linkable objects. Proceed as follows:
   a) Choose Window > Esko > Dynamic Art to display the Dynamic Art palette.
   b) Select all the Dynamic Art objects you created in the previous steps, and click New Dynamic Art on the Dynamic Art palette.
Dynamic Art automatically fills any empty text objects with Lorem ipsum text.

**Note:**

Dynamic Content will apply the last style in the Illustrator text object to any text that is linked from the XML file. For objects that contain different styles or formats, only the last style will be retained.

If this is a multilingual product and you will be using variable locale mapping (initially configured in the **Properties** dialog for a text object on the Dynamic Art palette; see the *Dynamic Art User Guide* for more information), set the locale variables before adding the XML content file.

### 3.2.1 Map Locale Variables (only for IPC XML standard)

If you are using locale variables to define the language of the text objects’ content, define the variables before adding the XML content file.

1. On the Dynamic Content palette’s fly-out menu, choose **Variable Locale Mapping**.
2. In the **Locale Variable Mapping** dialog box, click in the **Language** column next to the first variable you defined and choose the desired language from the list. Do the same thing in the **Country** column and click **OK**. Repeat for the other variables.
When you auto link, Dynamic Content compares the locale of an XML element to the value of the locale variable of a text object, and if they match, it fills the text object with content from the XML element.

### 3.3 Step 2: Link the XML file to the Illustrator document

Once your basic design or template is complete, you have to link the XML file with your source text to the Illustrator template. You can do this automatically or manually.

**Automatic** linking is possible if there is only one corresponding entry in the XML file for each type of dynamic object in the Illustrator document. You must link **manually** if there are more entries for each type of dynamic objects in the XML file than there are placeholders in the Illustrator document.

#### 3.3.1 Automatic linking

To automatically link the dynamic items in the Illustrator document to the elements defined in the XML file, do the following:

1. Choose **Window > Esko > Dynamic Content**. The **Dynamic Content palette** appears.
2. Choose **Add XML...** from the fly-out menu of the Dynamic Content palette. The **Open XML File** dialog box appears.
3. Browse to the XML file you want to link to the design.
4. Select the XML file and click **Open**. The XML file is now linked to the Illustrator document, and the XML file appears in the **Dynamic Content palette**.
5. If the data in the XML file and the number and type of dynamic objects match, Illustrator will ask if you want to link all the dynamic items. Click **OK** to start the auto link process.

6. Click **OK** again to start the auto link.
7. Illustrator will auto link as many items as it can. A status dialog may appear if some items could not be linked. Click OK to dismiss it.

After the auto link completes, review the status in the Dynamic Art palette. Items successfully linked will have an XML label next to them. You will have to manually link those items that did not automatically link.

**Resuming automatic linking**

If you canceled an auto link, you can resume it by doing the following:

1. Select the text items to auto link to the XML file you have already added to the Illustrator document.
2. In the Dynamic Art palette’s menu, click **Auto Link**.

3. Illustrator will ask you to confirm the auto link. Click OK to continue.

4. Illustrator will auto link as many items as it can. A status dialog may appear if some items could not be linked. Click OK to dismiss it.
After the auto link completes, review the status in the Dynamic Art palette. Items successfully linked will have an XML label next to them. You will have to manually link those items that did not automatically link.

### 3.3.2 Manual linking

To manually link the data in the XML file to the dynamic object placeholders, do the following:

1. **Choose** Window > Esko > Dynamic Content.
   - The Dynamic Content palette appears.
2. **Choose Add XML...** from the fly-out menu of the Dynamic Content palette.
   - The Open XML File dialog box appears.
3. **Browse to the XML file you want to link to the design.**
4. **Select the XML file and click Open.**
   - The XML file is now linked to the Illustrator document, and the XML file appears in the Dynamic Content palette.

### Link XML Elements to Illustrator Objects

In this step, you link the XML elements from the source file to the individual Illustrator text objects, barcodes and nutrition tables in the design.

Before you can execute this task, you must have linked an XML file to the Illustrator design. This means that at least one XML file must show up in the Dynamic Content palette.

1. **In the Dynamic Art palette, select the object you want to link to XML content: either click on the far right of the object’s entry in the palette, or click Select and zoom to object.**

   Depending on the type of XML file you linked to the Illustrator document, you can work either in Smart View or Advanced View. For GS1 and IPC XML files, the default is Smart View. Other XML flavors can only use the Advanced View. The steps below assume you are using the Smart View on an IPC XML file.

   Depending on the type of object you selected in the Dynamic Art palette, the Dynamic Content palette now displays only those elements from the XML file that can be linked to your selected object.

2. **In the Dynamic Content palette, click on the Make Dynamic Object icon corresponding to the XML element that contains the text you want to replace in the Illustrator template.**

   Alternatively, you can use the keyboard shortcut Shift+Alt+I on Windows or Shift+Command +I on Mac OS to attach the element to the Illustrator object.

   The object on the Illustrator canvas is now a dynamic object - its text contents are dynamically retrieved from the linked XML file. This is indicated in the Dynamic Content palette by the status icon.
Linking XML Elements to Illustrator Objects in the Advanced View

In this step, you link the XML elements from the source file to the individual Illustrator text objects, barcodes and nutrition tables in the design.

Before you can execute this task, you must have linked an XML file to the Illustrator design. This means that at least one XML file must show up in the Dynamic Content palette.

Depending on the type of XML file you linked to the Illustrator document, you can work either in Smart View or Advanced View. For GS1 and IPC XML files, the default is Smart View. Other XML flavors can only use the Advanced View.

The steps below assume you are using the Advanced View.

1. In the Dynamic Art palette, select the object you want to link to XML content: either click on the far right of the object entry in the palette, or click Select and zoom to object.
2. In the Dynamic Content palette, select the XML element that contains the text you want to replace in the Illustrator template.
3. Select the corresponding object in your Illustrator design.

   Note: If you added the dummy objects to the Dynamic Art palette, the fastest way to work is to select the Dynamic Art object in the list in the Dynamic Art palette, and then click Select and zoom to object.

4. Click Make Dynamic Object in the Dynamic Content palette.

   Alternatively, you can use the keyboard shortcut Shift+Alt+I on Windows or Shift+Command +I on Mac OS to attach the element to the Illustrator object.

The Illustrator objects have been replaced with a Dynamic Object. Also, the status icon appears in the XML tree indicating that the element has been attached to an object in the Illustrator document.
3.4 Step 3: Work with Changes in XML Content

Once the XML file has been linked to the Illustrator document, Dynamic Content will automatically notify you of any changes that are made to the contents of the XML file in external applications. When changes in the XML file are detected, the **XML Changed** dialog box appears, and you will be given two options. You can either:

- Update the Illustrator document now; or
- Cancel the update, in which case you will have to deal with the changed content later. The XML Not Up to Date status icon then appears next to the XML file in the XML Tree until you resolve the version difference.

**Note:** Following the instructions below, you will make manual changes to the XML content. In a real-world production scenario, the changes will of course be induced by the brand owner, for example by sending you a new version of the XML file.

1. Open the XML file you used earlier in a text or XML editor and change the text of the XML element you attached to Illustrator text.
2. Save the XML file. Make sure that you overwrite the file that you linked to the Illustrator design.
3. Open the Illustrator design, or switch back to Illustrator if you still have it open. The **XML Changed** dialog box appears.
4. Click **Update Now**. The Dynamic Objects in the Illustrator document are updated to reflect the changes in the XML file. The linked elements in which a change was detected are marked with a Check Alert⚠️.

5. Select the XML element in the tree that was changed and click **Zoom and Select Art** in the Dynamic Content palette. The corresponding Dynamic Object on the Illustrator artboard is now selected, the view centers on the object and Illustrator zooms into it.

6. Check whether the changes in content have caused any design problems such as text overflow or faulty text styling (highlighted by a special Check Alert⚠️), or other issues and correct them if necessary.

7. Click **Clear Check Alert** to remove the Check Alert from the selected element, indicating that you have checked this update and that the design is once more error-free.
3.5 Creating Design Variants (Line Extensions)

A typical use case for the Dynamic Content plug-in in the packaging industry is creating variants of a design, sometimes called "line extensions".

For example, based on an original packaging design, you can create variants for different language groups, or for different sizes. To create a design variant:

1. Create the base design by linking XML elements to the Illustrator objects using Dynamic Content.
2. Make a copy of the XML file you used in the original design and make any required changes in this copy.
   For example, you could change the text of the element containing the package quantity (e.g. for a box of diapers, change the CopyElement "Quantity" from 36 to 72).
3. Re-link the new XML file, which has the same structure but different contents, to the Illustrator design.
4. Save the Illustrator file as a new design variant file.

Tip: You can copy and paste dynamic objects between different Illustrator files as long as you make sure that the same XML file is linked to them. After copying an object, you can use the Relink XML feature to link the copied content to the variant XML file.

3.5.1 XML Linking Strategies for Design Variants

There are three main linking strategies that you can use for working with design variants or line extensions: one XML file per stock keeping unit (SKU), multiple XML files for each SKU, and a single XML file for multiple SKUs.

Overview
### The strategies in detail

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>One-to-one</td>
<td>Each design variant is linked to an external XML file that contains all the information. This is the least complex way to handle design variants, but it requires the brand owner to provide common data in each of the XML files (repetitions of the same data in different XML files).</td>
</tr>
<tr>
<td>One-to-many</td>
<td>In this workflow, a single XML file would contain all information for all extensions of a line. The advantage here is that the common data is stored only once.</td>
</tr>
<tr>
<td>Many-to-one</td>
<td>At least two XML files are linked to every design variant: one file unique to that variant, and one file that contains all common elements. While this eliminates the duplication of data, this situation can potentially be difficult and confusing for operators to manage.</td>
</tr>
</tbody>
</table>

---

**Diagram:**

- **One-to-One:**
  - System of Record
  - One XML file per SKU
  - Vanilla 12 oz.
  - Chocolate 12 oz.

- **Many-to-One:**
  - System of Record
  - Multiple XML files per SKU
  - Vanilla 12 oz.
  - Chocolate 12 oz.

- **One-to-Many:**
  - System of Record
  - One XML file for all SKUs
  - Vanilla 12 oz.
  - Chocolate 12 oz.
3.6 Relink text elements to different languages

If you have a multilingual XML file, you can relink linked GS1 or IPC Copy elements to different languages contained in the XML content file.

To relink linked GS1 or IPC Copy elements, do the following:

1. Select one or more elements to relink.
2. On the fly-out menu of the Dynamic Content palette, choose **Relink Selected Dynamic Objects**.

3. In the **Relink Selected Dynamic Objects** dialog, select **Target Language** and choose the new language setting from the drop-down list. Only those languages available in the XML content file appear in the list.
4. Click **OK** to relink the elements.

   The elements change to the language you specified.
4. Viewing XML Content in Dynamic Content

In this section you will learn how to interpret all the information displayed in the Dynamic Content palette, including how to manipulate an XML tree structure, what each of the icons and alerts means, and what the functions are of the buttons on the palette.

4.1 The Dynamic Content Palette

The main work area of Dynamic Content is the Dynamic Content palette. The palette contains information about the XML files that have been linked to the current Illustrator document. The Dynamic Content palette can operate in two view modes: Smart View and Advanced View. Much like any other Illustrator palette, Dynamic Content also has a fly-out menu.

4.1.1 Smart View

The Smart View of the Dynamic Content palette provides a simple-to-use view on the linked XML files. It hides much of the underlying XML and its complexity, and is ideal for quickly linking XML elements to placeholder objects.

Note: Smart View only works on GS1 and IPC XML files. When you add an IPC XML file to a document, Smart View is automatically enabled.

Components of the Smart View:
- **A.** The **Find** box, which works much like Spotlight on the Mac - simply type in (part of) a word to filter the list of elements “as you type”. Only the available XML files are searched and only matching elements are displayed. The Find function is case-insensitive. Click the **Reset Search** button to remove the filter.

- **B.** The **Element List**, which displays the XML elements that can be linked to your current selection in the Illustrator document. If you have a text object selected, only **CopyElements** will be listed, but if you select a barcode created with Dynamic Barcodes, only the **BarCode** elements in the GS1 or IPC XML will be listed.

- **C.** The **View** buttons, which allow you to switch between Smart View and Advanced View.

### 4.1.2 Advanced View

The Advanced View of the Dynamic Content palette consists of two panes, surrounded by a top and bottom row of buttons. In the Advanced View, you can make full use of the advanced features of Dynamic Content, and work with any type of XML.

**Two panes**

For ease of reference, the top pane is called the **XML Tree** and displays the structure of the XML file. The **Preview** pane at the bottom can be hidden if desired, and resized to fit more or fewer XML content.

**Components of the palette**

- **A.** The **Find** box, which works much like Spotlight on the Mac - simply type in (part of) a word to filter the list of elements “as you type”. In Advanced View, this searches case-insensitively in every element in all available XML files. All the elements are displayed; use the **Next** and **Previous** buttons to jump between matching elements. Click the **Reset Search** button to remove the filter.
• B. The top toolbar, which offers quick access to the Reflow Text, Synchronize, Add XML and Add Filter commands.
• C. The Filter toolbar, which allows you to filter the content displayed in the XML Tree pane.
• D. The XML Tree, which displays the content found in the linked XML file(s).
• E. The Preview pane, which displays the text content of the selected element in the XML Tree.
• F. The bottom toolbar, which offers quick access to the Clear Check Alert, Select and Zoom to Dynamic Object, Insert Element in Dynamic Object and Make Dynamic Object commands.
• G. The View buttons, which allow you to switch between Smart View and Advanced View.

Understanding the XML Tree

The XML tree pane contains a view on the internal structure of the linked XML file(s). Every node in the file (including elements and attributes) is visualized in the XML tree as an entry that can be expanded or collapsed.

Which nodes are displayed?

The XML tree displays:

• the XML file(s) linked to the Illustrator design, as the root node(s) in the XML tree;
• the XML elements in each XML file, as expandable entries under the root node (XML file);
• any XML attributes for every XML element, preceded by a “bullet” symbol. Attributes can be hidden from the XML Tree to simplify the view by choosing Hide Attributes from the fly-out menu.

Note:
The structure displayed in the XML tree can be simplified by applying one or more filters. See Filter options for the XML tree for more information.

Icons in the XML Tree

The XML tree displays different types of objects: one or more linked XML files, as well as the XML elements, XML attributes, and XML text elements that these files contain. Each of these objects is represented by an entry in the tree structure, and each type of object has its own icon.
Table: XML tree icons

The following table lists all the icons displayed in the XML tree structure view of the Dynamic Content palette.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![XML file]</td>
<td>XML file; not changed in Dynamic Content / changed in Dynamic Content</td>
</tr>
<tr>
<td>![Read-only file]</td>
<td>Read-only XML file; not changed in Dynamic Content / changed in Dynamic Content</td>
</tr>
<tr>
<td>![Node element, text inside]</td>
<td>XML node element containing text inside; placed (linked) / not placed</td>
</tr>
<tr>
<td>![Node element, directly containing text]</td>
<td>XML node element directly containing text; placed (linked) / not placed</td>
</tr>
<tr>
<td>![Node element, no text]</td>
<td>XML node element, which contains no text inside</td>
</tr>
<tr>
<td>![Attribute]</td>
<td>XML attribute</td>
</tr>
<tr>
<td>![Text element]</td>
<td>XML text element (has no icon)</td>
</tr>
</tbody>
</table>

Element Status Icons

The elements in the XML tree can be marked with one or more status icons, that show whether the content has been changed inside Dynamic Content, or externally in the XML file. The icons are displayed on the right side of the elements themselves, and in two columns: primary and secondary icons.

Table: Primary status icons

The following table lists all the status indicator icons displayed in the XML tree structure view:

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Change]</td>
<td>The contents of this XML element have been changed in the source XML file.</td>
</tr>
<tr>
<td><strong>Note:</strong> When at least one XML element in the file is marked with this status icon, the icon also appears next to the XML file name in the Dynamic Content palette.</td>
<td></td>
</tr>
</tbody>
</table>

| ![Warning] | The contents of this XML element have been changed in the source XML file to such an extent that some text styling or additional formatting that was applied by the user (e.g. color, size, font, soft returns, hyphens or spaces) could not be retained. |
| **Note:** When at least one XML element in the file is marked with this status icon, the icon also appears next to the XML file name in the Dynamic Content palette. |

| ![Pencil] | The contents of this element have been changed in the Dynamic Content plug-in. This status icon disappears when you save or close the XML file. |
Dynamic Content

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Note: When at least one XML element in the file is marked with this status icon, the icon also appears next to the XML file name in the Dynamic Content palette.</td>
</tr>
<tr>
<td></td>
<td>The element has a complex status, combining two types of changes:</td>
</tr>
<tr>
<td></td>
<td>• The contents of this XML element have been changed in the source XML file.</td>
</tr>
<tr>
<td></td>
<td>• The contents of this element have been changed in the Dynamic Content plug-in. This status icon disappears when you save or close the XML file.</td>
</tr>
<tr>
<td></td>
<td>The element has a complex status, combining two types of changes:</td>
</tr>
<tr>
<td></td>
<td>• The contents of this XML element have been changed in the source XML file to such an extent that some text styling or additional formatting that was applied by the user (e.g. color, size, font, soft returns, hyphens or spaces) could not be retained.</td>
</tr>
<tr>
<td></td>
<td>• The contents of this element have been changed in the Dynamic Content plug-in. This status icon disappears when you save or close the XML file.</td>
</tr>
</tbody>
</table>

Table: Secondary status icons

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Icon" /></td>
<td>The XML file was changed externally and has not yet been synchronized with the document.</td>
</tr>
<tr>
<td><img src="image2" alt="Icon" /></td>
<td>The XML element itself is not linked, but at least one of its child elements is. Click to select all objects linked to children.</td>
</tr>
<tr>
<td><img src="image3" alt="Icon" /></td>
<td>The XML element is linked to a Dynamic Object. Click to select linked object, Alt+Click to select and zoom.</td>
</tr>
<tr>
<td><img src="image4" alt="Icon" /></td>
<td>The XML element is linked to a Dynamic Object and the attached text object has text overflow. Click to select linked object, Alt+Click to select and zoom.</td>
</tr>
<tr>
<td><img src="image5" alt="Icon" /></td>
<td>The XML element is linked to a Dynamic Object and the object is selected. The background color corresponds to the selection color of the corresponding layer. Alt+Click to zoom.</td>
</tr>
<tr>
<td><img src="image6" alt="Icon" /></td>
<td>The same as above and the object has text overflow. Alt+Click to zoom.</td>
</tr>
<tr>
<td><img src="image7" alt="Icon" /></td>
<td>The XML element is linked / not linked and has at least one child which is linked and selected.</td>
</tr>
<tr>
<td><img src="image8" alt="Icon" /></td>
<td>The same as above and the object has text overflow.</td>
</tr>
</tbody>
</table>

The Preview Pane

At the bottom of the Dynamic Content palette is the Preview pane. The Preview pane displays the content of the element that is selected in the XML Tree, including the text of all its child elements.
Technically speaking, the Preview pane shows the concatenation of all text in the sub tree rooted at the XML element that is currently selected.

The text that is displayed can be used for standard copy and paste purposes, but this pane is not designed for editing.

Its function is mainly to help the user to link the appropriate XML elements to text objects in the Illustrator design.

**Note:**

To hide or display the Preview pane, choose **Hide/Show Preview** in the fly-out menu.

Drag the border between the XML Tree and the Preview pane up or down to resize the Preview pane.

### 4.1.3 The Fly-out Menu

The Dynamic Content palette has a fly-out menu which offers access to a variety of commands. The commands you will use most often are also directly accessible from the palette buttons to speed up your daily work.

**Accessing the fly-out menu**

To open the fly-out menu, click the expand icon in the upper right hand corner of the Dynamic Content palette.

**Fly-out Menu Command Reference**

This topic lists all the commands on Dynamic Content’s fly-out menu, along with a brief description of their functions.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add XML</td>
<td>Adds a new XML file in the Dynamic Content palette.</td>
</tr>
<tr>
<td><strong>Note:</strong> You can add multiple XML files to a single Illustrator document to combine elements from all different XML files in a single design.</td>
<td></td>
</tr>
<tr>
<td>Relink XML...</td>
<td>Replaces an existing XML file link by a new one.</td>
</tr>
<tr>
<td><strong>Note:</strong> This operation cannot be undone.</td>
<td></td>
</tr>
<tr>
<td>Synchronize XML</td>
<td>Synchronizes all the external XML files with the corresponding content used in Dynamic Content. Note that the synchronization works in both directions: import modifications to external XML files and export changes done using &quot;Edit Element&quot; in Dynamic Content back to the external XML files. In case there would be conflicts, the user will be prompted to choose the behavior.</td>
</tr>
<tr>
<td>New XML...</td>
<td>Creates a new XML file according to the GS1 or IPC XML schema. The XML file's format depends on the settings in the Preferences. See <a href="#">Dynamic Content Preferences</a> on page 68</td>
</tr>
<tr>
<td>Save XML As...</td>
<td>Saves the current XML to a new file according to the GS1 or IPC XML schema. The XML file's format depends on the settings in the Preferences. See <a href="#">Dynamic Content Preferences</a> on page 68</td>
</tr>
<tr>
<td>Remove XML</td>
<td>Removes the selected XML file from the Dynamic Content palette. Dynamic Objects that refer to XML elements in a file that is removed will be converted to regular Illustrator text objects.</td>
</tr>
<tr>
<td><strong>Note:</strong> This operation cannot be undone.</td>
<td></td>
</tr>
<tr>
<td>Revert XML</td>
<td>Replaces the contents of the selected XML file in Dynamic Content by the current version on disk, i.e. this clears all changes made inside Dynamic Content.</td>
</tr>
<tr>
<td>XML Info...</td>
<td>Displays the XML Information dialog box, which shows basic information on the selected XML file, and allows you to make the XML file “Read Only”.</td>
</tr>
</tbody>
</table>
| Export Artwork Response| This works only with GS1 or IPC XML files. Exports the selected XML file as a report file that can be used for electronic verification of the linked state of the content by the content creator. The exported XML file has the same structure and content as the original GS1 or IPC XML file and will add, in
In the case of the IPC standard, a "LinkedState" attribute to each CopyElement, Barcode and NutritionFacts element that occurs in the IPC-XML file.

The value of the attribute will be set as follows:

"LinkedToArtwork": Indicates that the CopyElement, Barcode or NutritionFacts element is linked to a text, barcode or table object in the artwork.

"NotLinkedToArtwork": Indicates that the CopyElement, Barcode or NutritionFacts element is not linked to any text, barcode or table object in the artwork.

In the case of GS1 file will export the "BMS Artwork Content Response" XML file. The file is based on the GS1 BMS standard (see [http://www.gs1.org/gsmp/kc/ecom/xml/](http://www.gs1.org/gsmp/kc/ecom/xml/)).

The file indicates e.g.:

- by the `changedByRecipient` tag by the linked element can show whether the content of the element was changed inside the Dynamic Content plug-in
- the `actualNumberOfOccurrences` element informs about the number of really placed instances of the same content (the intended number is indicated by the `expectedNumberOfOccurrences` element of the source file)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Make Dynamic</td>
<td>Replaces the contents in the selected Illustrator object with the contents of the selected XML element, creating a Dynamic Object.</td>
</tr>
<tr>
<td>Release</td>
<td>Converts the selected Dynamic Object(s) into a regular Illustrator text object(s). Note that this permanently removes the link to the XML content.</td>
</tr>
<tr>
<td>Relink Selected Dynamic Objects</td>
<td>Relink Selected Dynamic Objects operates on all selected dynamic objects that are already linked to XML file A and will attempt to re-link all of these objects to selected XML file B. Select both the linked objects to re-link (in the Illustrator document window) and the target XML file to re-link these objects to (in the Dynamic Content palette). For each of the selected objects, the function will attempt to find the corresponding element to re-link to in the target XML file. In case an appropriate new XML element can be found for each of the selected objects, then they will all be re-linked to the new XML file. This function may be useful when creating line-extensions or design variants based on content provided in different XML files. Objects that are linked to XML elements in the content file for the first variant (e.g. strawberry.xml) can then be re-linked to the content file for another variant (e.g. banana.xml).</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Relink Selected Dynamic Objects</td>
<td>Also allows you to change the language of a linked dynamic object if you are working with a multilingual GS1 or IPC XML content file. See <a href="#">Relink text elements to different languages</a> on page 22 for more information.</td>
</tr>
<tr>
<td>New Element…</td>
<td>Appends the text of the selected Illustrator text object(s) to the current XML file as an XML element in the GS1 or IPC format, according to the settings in the Preferences.</td>
</tr>
<tr>
<td>Edit Element…</td>
<td>Opens a basic XML editor in which you can edit the contents of the selected XML element. The editor displays all the text inside the selected XML elements: all child elements in the tree structure are included and editable.</td>
</tr>
<tr>
<td>Locate Element</td>
<td>Locates the XML element associated with the selected Dynamic Object in the Dynamic Content palette. If the XML tree structure was collapsed, it is now expanded in order to reveal the associated XML element.</td>
</tr>
<tr>
<td>Format Dynamic Text</td>
<td>Displays the Dynamic Object Properties dialog box, which allows you to fine-tune how multiple elements in a single Dynamic Object are formatted.</td>
</tr>
<tr>
<td>Insert at Start</td>
<td>Inserts the contents of the selected XML element at the top of the Dynamic Object in the Illustrator design.</td>
</tr>
<tr>
<td>Insert at Cursor Position</td>
<td>Inserts the contents of the selected XML element at the place of the text cursor within the Dynamic Object in the Illustrator design. If the cursor is located inside the word, the content is inserted in the closest space to avoid splitting words.</td>
</tr>
<tr>
<td>Insert at End</td>
<td>Appends the contents of the selected XML element at the end of the Dynamic Object in the Illustrator design.</td>
</tr>
<tr>
<td>Style Mapping…</td>
<td>Opens the Style Mapping dialog box, in which you can assign named character and or paragraph styles to selected XML elements. It also allows you to specify which XML tags should be interpreted as paragraphs (i.e. will be followed by a &quot;hard return&quot;).</td>
</tr>
<tr>
<td>Split Text</td>
<td>Splits the contents of a XML element into multiple text objects. Click inside the initial linked text object at the desired split position and then click Split Text. The result will be two text objects linked to the same XML element but each will contain only part of the element. If the XML changes, Dynamic Content will try to maintain the split at its original position, but if the content is significantly different, you may need to merge the text elements and split them again.</td>
</tr>
<tr>
<td>Merge Text</td>
<td>Merges previously-split dynamic text items back together. Click inside one of them and click Merge Text on the Dynamic Content palette’s fly-out menu.</td>
</tr>
</tbody>
</table>
### Command Description

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable Locale Mapping...</td>
<td>Allows editing the mapping of Locale variables, set in the artwork template previously by the Dynamic Art plug-in. Here is possible to map the Locale variables to the actual locales valid for the current document and remove existing mapping. This is only applicable for IPC XML standard.</td>
</tr>
<tr>
<td>Select Object</td>
<td>Selects the Dynamic Object(s) on the Illustrator artboard that are attached to the selected XML element.</td>
</tr>
<tr>
<td>Highlight Dynamic Objects</td>
<td>Highlights all Dynamic Objects in the Illustrator document.</td>
</tr>
<tr>
<td>Clear &quot;Check&quot; Alert</td>
<td>Removes the Check Alert from the current element. When the contents of the external XML file change, the changes are indicated in Dynamic Content by Check Alerts.</td>
</tr>
<tr>
<td>Clear All &quot;Check&quot; Alerts</td>
<td>Removes the Check Alerts for all XML elements in the XML file.</td>
</tr>
<tr>
<td>Find in XML...</td>
<td>Displays the Find dialog box, in which you can search the text content of the linked XML file.</td>
</tr>
<tr>
<td>Hide/Show Attributes</td>
<td>Displays or hides the XML attributes of the elements in the Dynamic Content palette.</td>
</tr>
<tr>
<td>Hide/Show Preview</td>
<td>Displays or hides the Preview pane in the Dynamic Content palette.</td>
</tr>
</tbody>
</table>

---

### 4.1.4 The Toolbars: Button Reference

The buttons on the Dynamic Content toolbars (in Advanced View) allow you to quickly execute the most common operations in a Dynamic Content workflow. This section details each of the buttons on the two toolbars and their functions.

**Top toolbar**

Table: Buttons on the top toolbar

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
</table>
| ![Add XML File](image.png) | Add XML File.  
Adds a new XML file in the Dynamic Content palette.                                                                                      |
| ![Reflow Text](image.png)  | Reflow Text  
Enables the Reflow Text mode, which allows you to transform (scale, rotate, etc.) the Dynamic Object without deforming the text.                                                                 |
**Button** | **Description**  
--- | ---  
⚠️ | **Caution:**  
Take care when using this mode. It should be turned off as soon as you finish transforming the Dynamic Object with Illustrator's tools. While this mode is enabled, a number of essential Dynamic Content features will not work on this object.

 Readonly | **Synchronize**  
Synchronizes all the external XML files with the corresponding content used in Dynamic Content. Note that the synchronization works in both directions: import modifications to external XML files and export changes done using "Edit Element" in Dynamic Content back to the external XML files. In case there would be conflicts, the user will be prompted to choose the behavior.

🔍 | **Find**  
In Advanced View, enables you to find a piece of text in the XML file. For example, typing “chocolate” will list all copyelements that have “chocolate” in the XML text in all available XML files. In Smart View, it searches only the content currently available in the Smart View window of the palette.

➕ | **Add Filter**  
Adds a new view filter to the Dynamic Content palette.

➖ | **Remove Filter**  
Removes the corresponding filter from the Dynamic Content palette.

### Bottom toolbar

**Table: Buttons on the bottom toolbar**

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
</table>
| ✔️ | **Clear Check Alert.** Depending on the current selection, this button:  
- Removes the Check Alert from the current element. When the contents of the external XML file change, the changes are indicated in Dynamic Content by Check Alerts.  
- Removes the Check Alerts for all XML elements in the XML file. |
| 🔍 | **Select and Zoom to Dynamic Object.** This button:  
- Selects the Dynamic Object(s) on the Illustrator artboard that are attached to the selected XML element.  
- Zooms to the selected Dynamic Objects on the artboard. |
| ✚ | **Insert Element in Dynamic Object.** Inserts the contents of the selected element in the XML tree into the selected Dynamic Object on the artboard. The content of the current element is appended at the end of the Dynamic Object. |
### Button Description

**Tip:**
To insert an element at the top of a Dynamic Object, use the **Insert at Start** command in the fly-out menu or **Alt+Click** this button.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Button Icon" /></td>
<td>Replaces the contents in the selected Illustrator object with the contents of the selected XML element, creating a Dynamic Object.</td>
</tr>
</tbody>
</table>

**Note:**
If an Illustrator text object contains multiple styles (fonts, text sizes, ...) Dynamic Content will only retain the formatting of the last style found in the text object.

---

### 4.2 Viewing XML Information

You can use the **XML Info...** fly-out menu command to view essential characteristics of the selected XML file in the XML tree.

- The **XML Information** dialog box displays the file **Size**, **Location** on disk, **Created** and **Modified** date, and similar basic information about the XML file.
- In addition, it also lists the current **Content Status**, which informs you about the relationship between the text in the XML file and the linked text in your Illustrator design.
- The **XML info** shows whether the XML file is taken as a file conforming to any of the supported standards (GS1 or IPC). This can be important for other productivity enhancers of the Dynamic Content plug-in, e.g. for Automatic Linking.
- You also can set the highlight color by clicking **Select Color**.

![XML Info Dialog Box](image)
• You can use the XML Information dialog box to make the XML file read-only on disk.
• Finally, you can use the Select Color button to change the color used to highlight elements from this particular XML file (both in the XML Tree and on the Illustrator canvas).

2. Do one of the following:
   • choose XML Info... from the fly-out menu of the Dynamic Content palette.
   • double-click the XML file node in the XML tree.

The XML Information dialog box appears.

4.3 Filter Options for the XML Tree

The Add Filter button on the Dynamic Content palette allows you to filter the list of elements that are displayed in the XML tree. You can choose to use a standard, built-in filter, or create your own filter based on an element name, an attribute name, or an attribute value.

Example: combining multiple filters

The following example of a filter shows only those elements that have been updated in the source XML file. In addition, it does this in a simplified view based on the IPC schema.
4.3.1 Applying a Filter to the XML Tree

You can create a filter to display only those elements in the linked XML files(s) that are of interest to you. Dynamic Content's filter option also allows you to simplify what is displayed in the XML tree based on a combination of multiple filters.

Multiple filters work in a cumulative manner, using an “AND” logic.

Note:
When combining filters, keep in mind the cumulative effect. For example, applying a filter based on the Element Name “CopyContent” and then adding another Element Name filter “CopyElement” will display an empty palette (because there can be no elements that have both the names “CopyContent” and “CopyElement” at the same time).

1. Create a character style in Adobe Illustrator and map it to text objects, then select one or more text objects.

2. On the top toolbar of the Dynamic Content palette, click Add Filter. The filter options appear on the palette, just below the top toolbar.

3. Select a filter type from the Filter Type list.

   For detailed descriptions of the filter types, see Filter Options in Detail.

4. Do one of the following:
• If you selected the Element Name or Attribute Name filter type, choose an Element or Attribute Name from the Filter list.
• If you selected the Attribute Value Contains filter type, enter one or more attribute values in the Filter box.

The XML tree is filtered according to your criteria.

5. If necessary, proceed by adding or removing additional filters. You can:
   • click **Add Filter** to add another filter.
   • click **Remove Filter** to remove a filter.
   • switch each filter on or off by (de)selecting the check box preceding it.

### 4.3.2 Filter Options in Detail

A number of filter types can be activated and combined. This topic describes the details of each filter type.

<table>
<thead>
<tr>
<th>The filter type...</th>
<th>Applies the following filter...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simplified View on XML Schema</td>
<td>Filters the XML tree based on the standard GS1 or IPC XML format. This will display only the <code>CopyElement</code> elements that contain the actual text that will be used in your design.</td>
</tr>
<tr>
<td>Element Name</td>
<td>Filters the XML tree to display only elements matching the name you select from the Filter details list.</td>
</tr>
<tr>
<td>Attribute Name</td>
<td>Filters the XML tree to display only elements with attributes matching the attribute name you select from the Filter details list.</td>
</tr>
<tr>
<td>Attribute Value Contains</td>
<td>Filters the XML tree to display only elements that have an attribute whose value matches the text you enter in the Filter details box.</td>
</tr>
<tr>
<td>Multiple keywords can be entered using the following format: &quot;keyword1&quot;,&quot;keyword2&quot;,&quot;keyword3&quot;</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong> Entering &quot;en-us&quot;,&quot;en-uk&quot; will display only those elements that have at least one attribute with a value containing the strings &quot;en-us&quot; or &quot;en-uk&quot;.</td>
<td></td>
</tr>
<tr>
<td>Used Elements</td>
<td>Filters the XML tree to display only elements that have been linked to Illustrator text objects.</td>
</tr>
<tr>
<td>Unused Elements</td>
<td>Filters the XML tree to display only elements that have not yet been linked to Illustrator text objects.</td>
</tr>
<tr>
<td>The filter type...</td>
<td>Applies the following filter...</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Elements with Updated Text</td>
<td>Filters the XML tree to display only elements that have been modified in the source XML file.</td>
</tr>
<tr>
<td>Tip:</td>
<td>This filter can function as a &quot;to-do&quot; list of elements that need to be checked as a result of an update of one or more external XML files.</td>
</tr>
<tr>
<td>Elements with Text Overflow</td>
<td>Filters the XML tree to display only elements that have been modified in the source XML file to such an extent that it has caused text overflow in the Illustrator design.</td>
</tr>
<tr>
<td>Elements with Text</td>
<td>Removes any element that does not contain any text element in its sub-tree. This filter hides any elements in the XML file purely for structural reasons which do not contain any text appearing in the artwork.</td>
</tr>
<tr>
<td>GS1 Instance ID</td>
<td>Filters the XML tree to display only elements with the GS1 instanceSequence matching the value you select from the Filter detail list. The list contains only instanceIDs present in the GS1 XML files(s) actually loaded. Use in combination with ‘Simplified View On GS1-XML Schema’ filter only.</td>
</tr>
<tr>
<td>GS1 Locale</td>
<td>Filters the XML tree to display only elements with the GS1 localeSequence matching the value you select from the Filter detail list. The list contains only locales present in the GS1 XML files(s) actually loaded. The localeSequence numbers are translated using the Artwork Content Locale mapping present in the GS1 XML file(s). Use in combination with ‘Simplified View On GS1-XML Schema’ filter only.</td>
</tr>
<tr>
<td>GS1 Element Type</td>
<td>Filters the XML tree to display only elements of the type you select from the Filter detail list. The possible values are Copy Element for text content, Data Carrier used usually for barcodes, Graphic Element for images and symbols and Structured Copy Element used usually for Nutrition tables. Use in combination with ‘Simplified View On GS1-XML Schema’ filter only.</td>
</tr>
<tr>
<td>GS1 Approved</td>
<td>Filters the XML tree to display only elements with the GS1 Approved status matching the value you selected from the Filter detail list. The values are True and False. Use in combination</td>
</tr>
</tbody>
</table>
4.4 Manipulating the XML Tree

Once you have linked an XML file into the Dynamic Content palette of your Illustrator document, you can use the XML tree in the palette to navigate and even edit the contents of the XML file.

Proceed as follows to view and manipulate the XML tree.

1. In the Dynamic Content palette, click the arrow preceding the topmost XML element.
   This is the root element in your file.
   The XML tree expands one level down the hierarchy, and the elements in your XML file become visible.

2. Click the Add filter button.
   The Filter list appears.

<table>
<thead>
<tr>
<th>The filter type...</th>
<th>Applies the following filter...</th>
</tr>
</thead>
<tbody>
<tr>
<td>GS1 For Placement Only</td>
<td>Filters the XML tree to display only elements with the GS1 forPlacementOnly status matching the value you selected from the Filter detail list. The values are True and False. Use in combination with 'Simplified View On GS1-XML Schema' filter only.</td>
</tr>
<tr>
<td>GS1 Element With Notes</td>
<td>Filters the XML tree to display only elements with any content in the 'Notes' tag. Use in combination with 'Simplified View On GS1-XML Schema' filter only.</td>
</tr>
<tr>
<td>GS1 Element Type Code Contains</td>
<td>Filters the XML tree to display only elements with the GS1 &lt;structuredCopyElementTypeCode&gt; tag matching the value you enter to the filter's detail. Example: If you enter &quot;brand&quot;, the list will contain 'BRAND_NAME' and 'SUB_BRAND_NAME' elements only. Use in combination with 'Simplified View On GS1-XML Schema' filter only.</td>
</tr>
</tbody>
</table>
3. Select an option from the Filter list.
   For example, you can select the Simplified View on XML Schema / IPC-XML filter type to view only the text contents of an IPC XML file.
   The list of XML elements in the palette is filtered to match your criteria.

4. Click the Find button to open the Find dialog box.
5. Enter a search term in the Find dialog box and click Next.
   The first XML element matching your search term is selected in the XML tree. The Find dialog box shows how many more occurrences of the search term were found in the linked XML file(s).
Also, note that the text of the selected element is now displayed in the Preview pane of the Dynamic Content palette.

You have performed some basic operations on the XML tree that will help you speed up your work in Dynamic Content: navigating the tree structure, filtering the list of elements, and searching for text in the XML files are all very common operations.
5. Managing Changes to the XML Content

One of the main reasons for using Dynamic Content is that it allows you to deal with changes in textual content of your designs more efficiently and more quickly. This section examines how you manage changes to the XML content of your designs using Dynamic Content.

The changes in your design can be of two types:

- the external XML file can change, for example if the brand owner sends you a new version of the copy content and you want to update your design accordingly; or
- you can manually edit the copy content in Illustrator, and apply those changes to the external XML file.

In short, changes can be applied in both directions: XML to Illustrator, or Illustrator to XML.

5.1 How Dynamic Content Deals with Changes

As soon as the content of the linked XML file(s) is changed, Dynamic Content will warn you of these changes. If you did not have the Illustrator design open at the time of the changes, you will be notified the next time you open the file. This behavior is governed by the Always Mark and Update Changed Dynamic Objects Without Asking option in the Preferences dialog.

Changes in the XML file

When changes in the XML file are detected, the XML Changed dialog box appears, and you will be given two options. You can either:

- Update the Illustrator document now; or
- Cancel the update, in which case you will have to deal with the changed content later. The XML Not Up to Date status icon then appears next to the XML file in the XML Tree until you resolve the version difference.
If you do click Update Now, the text inside any Dynamic Objects is updated with the new XML content. Every element that experiences these changes is marked with a ! Check Alert in the XML tree.

**Changes inside Dynamic Content**

You can also make changes to the XML content inside Illustrator by using the **Edit Element** command on the Dynamic Content fly-out menu. Any changes you make in Illustrator using Dynamic Content are not automatically applied to the source XML document. Instead, they are marked in the XML tree with the status icon so you have the chance to review these changes before applying them to the linked XML file using the Synchronize command.

---

### 5.2 Approving Changes in the External XML File

To approve changed content, you have to review the Dynamic Objects that have undergone content changes.

1. Open the Illustrator file for which one or more of the linked XML files have been modified externally.
   
   Once the Illustrator document opens, the **XML Changed** dialog box appears.

2. Click **Update Now** to apply the changes to the Illustrator document.
   
   Any Dynamic Objects that are linked to XML elements that have been changed in the XML file are automatically updated to match the new XML content.

   Every XML element that has been modified is marked with a ! Check Alert in the XML tree.

**Note:**

Elements that have been modified to such an extent that one or more Dynamic Objects have lost formatting are marked with a different ! Check Alert. These elements will require your special attention to solve formatting issues in your final design.

3. Review each of the Check Alerts:
   
   a) Select an element with a Check Alert in the XML tree;

   b) Click the **Select and Zoom to Dynamic Object** button on the bottom toolbar of the Dynamic Content palette.

      The Dynamic Objects that have been modified are now visible and selected on the Illustrator artboard.

   c) Make any changes to the Dynamic Objects that are needed as a result of the updated text.

   d) Click the **Clear Check Alert** button on the bottom toolbar of the Dynamic Content palette to remove the Check Alert from this element, then proceed to the next element until there are no more Check Alerts in the XML tree.
5.3 Applying Text Edits in Illustrator to the XML File

If you have edited the text of one or more XML elements using the Edit Element command on the Dynamic Content fly-out menu, you will have to decide whether or not you want to apply those changes to the external XML file.

1. Open the Illustrator file that contains text edits that have not yet been applied to the XML file.
2. Review each of the text edits by selecting the elements that are marked by a Check Alert, and then clicking the Select and Zoom to Dynamic Object button on the bottom toolbar.
3. Make any required changes to the design or to the text.
4. Click the Synchronize button on the top toolbar to apply the text edits to the XML file.

5.4 Creating New XML Files in Illustrator

Dynamic Content allows you to create new XML files with the GS1 or IPC XML schema directly from within Illustrator, based on the text you may already have in your designs. Typically, you will use this feature to convert existing, legacy designs into XML-driven, Dynamic Content designs.

5.4.1 New XML

Use the New XML... command on the Dynamic Content fly-out menu to create a new (empty) XML file.

1. Choose New XML... from the fly-out menu.
   - When using IPC standard, the New XML dialog box appears. Fill in all the fields. The metadata section of your XML file will be filled in based on the information you supplied.
• When using GS1 standard, the GS1 file is created with default values in its header, without asking the user.

You can set the IPC or GS1 standard in the Preferences. See Dynamic Content Preferences on page 68

2. Click OK.
   A new XML file based on the standard IPC or GS1 schema is created.

5.4.2 Add Illustrator Text to the XML

Use the New Element command on the Dynamic Content fly-out menu to add text content from selected Illustrator text objects to the linked IPC XML file. This also converts the selected Illustrator objects into Dynamic Objects.

The Illustrator document must be linked to an IPC-based XML file.

1. Select one or more Illustrator text objects.
2. Choose New Element from the fly-out menu.
   The New Element dialog box appears.
3. When using the IPC XML standard:

   a) Choose a CopyElement type from the Element Type and Element Subtype lists.
   b) If necessary, select the language (locale) of the element content from the Locale lists.
   c) Enter the identification code of the design (e.g. POA1) in the Artwork ID box.
d) If you want to map the text styles in the selected Illustrator text object to XML tags inside the 
CopyElement, enable the Map Text Styles to Tags option. 
Dynamic Content will attempt to preserve the formatting of the original text by inserting 
additional XML tags, which are automatically mapped to the text styles that were applied to 
the Illustrator text object.

4. When using the GS1 XML standard:

a) Choose an **Element Type Category** and an **Element Type**.
b) Define the **Instance Sequence**. This allows to identify different elements of the same Type.
c) Define the **Locale Sequence**. The POA header defines a number for each language. The **Locale 
Sequence** in each element refers to that mapping

5. Click **OK**.

The new content is added to the internal representation of the XML file. When using the 
Synchronise function, the new content is written to the external XML file.
6. Editing Dynamic Objects

This section examines the editing features in Dynamic Content. The following editing operations are available in Dynamic Content:

- You can edit the text of an XML element directly inside Illustrator. Also, you can transform the Illustrator text boxes that are the basis of the Dynamic Objects.
- You can insert a specified set of special characters (such as spaces, hyphens or tabs) in a Dynamic Object to improve the text layout.
- You can insert text from additional XML elements in an existing Dynamic Object, either by appending it at the end or by inserting it at the start of the existing text content.
- You can edit the properties of dynamic barcodes and nutrition tables using the Dynamic Tables and Dynamic Barcodes plug-ins.

6.1 What is a Dynamic Object?

A Dynamic Object is an Illustrator text object that has been linked to an XML element using the Dynamic Content plug-in.

Dynamic Content encapsulates the original text object in the Dynamic Object, and in the process it prevents a number of changes from being made: you can no longer use the standard Illustrator Text tools to modify the contents of a Dynamic Object.

6.2 Which Standard Illustrator Edits Are Allowed?

A Dynamic Object is protected from text edits using the Illustrator Text tools. However, you can still modify the text of a Dynamic Object to a certain extent.

The following edits remain possible:

- You can move, rotate or scale the Dynamic Object using the standard Illustrator transform tools.
- You can change the formatting (e.g. the font family or alignment properties) of the text by applying text styles in Illustrator. To do this, use the controls on the Type palette in Illustrator.
- You can change the color of the text.

**Note:**

When you scale a Dynamic Object, its text content is scaled as well. To allow the text box to “reflow” as per the standard Illustrator behavior, click the Reflow Text button on the top toolbar before making any changes.
6.3 Inserting Additional XML Elements in a Dynamic Object

Once a Dynamic Object has been created, you can add additional content (from other XML elements) to this same object. New content can be added either to the end of the content (Insert at End) or, if needed, before the beginning of the existing content (Insert at Start).

Note:
This feature applies to a single Dynamic Object. You can therefore have a Dynamic Object that is linked to XML Element A, and another Dynamic Object in the same design that is linked to both XML Element A and Element B.

1. Select a Dynamic Object on the Illustrator artboard.
2. In the XML tree, select the XML element that you want to add to the Dynamic Object
3. Do one of the following:
   - To append the text of the new XML element at the end of the Dynamic Object, click the Insert Element in Dynamic Object button on the bottom toolbar of the Dynamic Content palette. Alternately, use Insert at End on the Dynamic Content fly-out toolbar.
   - To insert the text of the new XML element in between two XML elements linked to the single Dynamic Object, place the Type Tool cursor between the two linked elements, and use the Insert at Cursor Position option on the Dynamic Content fly-out toolbar.
   - To insert the text of the new XML element at the start of the Dynamic Object, Alt+Click the Insert Element in Dynamic Object button on the bottom toolbar of the Dynamic Content palette. Alternately, use Insert at Start from the Dynamic Content fly-out menu.
The new XML content is added to the Dynamic Object.

Tip:
You can define how the different XML elements should be combined in the Dynamic Objects by using the Format Dynamic Text command in the fly-out menu.

6.4 Formatting Multiple Elements in a Single Dynamic Object

In the Format Dynamic Text dialog box, you can view information on the XML elements linked to the selected Dynamic Object, and you can define exactly how these XML elements should be combined in the object.

Tip:
You can also use the Format Dynamic Text command to insert a text prefix before the XML element content (e.g. to add "Ingredients: " before the ingredients list), even without adding multiple elements to a single Dynamic Object. You can also add a suffix in the same way.

1. Select a Dynamic Object on the Illustrator artboard.
   The Format Dynamic Text dialog box appears.

3. Make any needed changes to the formatting of the elements in the Dynamic Object:
If you want to... Then...

Insert text or special characters before one or more elements
Select the element(s) in the list and type the desired prefix in the Prefix field.

Insert text or special characters after one or more elements
Select the element(s) in the list and type the desired prefix in the Suffix field.

Treat all elements as part of the same block of text, without paragraphs
Deselect the Preserve Paragraphs option.

Change the order in which the elements appear in the Dynamic Object,
Select the element(s) you want to move, and click the Move Up or Move Down buttons.

Remove blank spaces in the elements,
Enable the Remove Leading, Trailing and Duplicate Spaces option. This removes any undesirable whitespace characters from the XML before the text is displayed in the Dynamic Object. Additional whitespace characters are sometimes added by XML editors in an effort to make the XML more legible for human readers.

4. Click OK.

6.5 Editing the Text of an XML Element

You can edit the text of a Dynamic Object by using the Edit Element command on the fly-out menu. This will change the contents of the linked XML file itself. Of course, Dynamic Content marks these changes so you can review them before they are applied to the external XML file.

The Edit Element dialog box displays the text of the selected XML element. The XML tags from the linked XML file are displayed as square brackets [ ]. This makes it easier to focus on editing the text without worrying about XML structure and syntax.

**Note:**
The square brackets ([ ]) in the Edit Element dialog box represent XML opening and closing tags. To type a square bracket in the text of your design, use the following codes: &left; for “[” or &right; for “]”.

1. Select the XML element for which you want to edit the text content in the XML Tree.
2. Choose Edit Element... in the fly-out menu of the Dynamic Content palette.
   The Edit Element dialog box appears.
3. Make changes to the text where needed.
4. Click OK to apply the changes to your design.

The new text is used in any Dynamic Objects linked to the XML element you edited, and the XML element itself marked with a 🆕 Check Alert icon in the XML Tree.

However, the new text is not yet saved in the linked XML file. To save your edits:
• Click the Synchronize button on the top toolbar. This will save your edits into the currently linked XML file.
• Alternatively, choose Save XML As... from the fly-out menu to save the new content into a new XML file.

6.5.1 The Edit Element Dialog Box in Detail

The Edit Element dialog box displays all the text of the selected XML element and its children. The XML tags from the linked XML file are replaced by square brackets [ ]. This makes it easier to focus on editing the text without worrying about XML syntax.

The option Show elements as story formats the content in the editing area in a more user-friendly manner. If you disable this option, every element starts on a new line in the editing area.

**Note:**

The square brackets ([ ]) in the Edit Element dialog box represent XML opening and closing tags. To type a square bracket in the text of your design, use the following codes: &left; for “[” or &right; for “]”.

The square brackets “[” cannot be deleted, because they represent the structure of the XML tree.
6.6 Making minor layout edits

The text of a Dynamic Object is protected from direct editing using the Illustrator Type tools (Type tool, Area Type Tool, ...). However, you can use the Type tools to make minor layout edits in a very strictly defined number of cases.

Text in a Dynamic Objects cannot be edited using the Type tools because this would go against the basic workflow principle of Dynamic Content: to make sure that your design remains synchronized with the external content and only displays text sourced from the linked XML file(s).

If you do try to edit a Dynamic Object, you will find that you can only insert a limited number of layout characters (i.e. tab, line break, ...) and that all other edits will be blocked.

Figure 1: The Illustrator Type tools can only be used to insert layout characters

- You can insert the following special characters:

<table>
<thead>
<tr>
<th>This character...</th>
<th>Can be inserted...</th>
</tr>
</thead>
</table>
| a whitespace character | • after an existing whitespace character in the XML content;  
• after a manually added hyphen. |
| a hyphen ( - ) character | • inside an existing word in the XML content;  
• after a manually added whitespace character. |
| a tab character | • after any whitespace character (existing or manually added);  
• after a manually added hyphen character, if that is not already followed by a manually added whitespace. |
| a new paragraph (return) and a line break (shift+return) character | • after any whitespace character (existing or manually added);  
• after a manually added hyphen character, if that is not already followed by a manually added whitespace. |

6.7 Releasing a Dynamic Object to regular text

You can easily convert a Dynamic Object back into a regular Illustrator text object. To do this, use the Release command on the fly-out menu of the Dynamic Content palette.
Designers who do not have access to the Dynamic Content plug-in can choose to convert the Dynamic Objects (which retain their special, protected status even when opened on a workstation without Dynamic Content) back into regular text objects using the Illustrator command **Object > Expand...**.

This allows them to work on the design and make changes where needed, but it also breaks the Dynamic Content workflow.

**Note:**
When you release a Dynamic Object using one of the two available methods, the link to the XML element(s) is permanently broken.

- To release a Dynamic Object, select it on the Illustrator artboard and:
  - Choose **Release** from the Dynamic Content fly-out menu; or
  - Choose **Object > Expand...** from the Illustrator menubar.

### 6.8 Exchanging Dynamic Content Documents

If you have to send an Illustrator document that contains linked objects that were created by the Dynamic Content plug-in to someone who does not have the plug-in, you have two options.

- You can release the Dynamic Objects back to normal Illustrator objects before sending the document to the recipient. Select all the linked objects and then choose the Release command on the fly-out menu.
  The document no longer contains any Dynamic Objects and all links to the XML file(s) have been removed.
- You can send the Illustrator document to the recipient without removing the Dynamic Objects.

<table>
<thead>
<tr>
<th>If the recipient...</th>
<th>Then (s)he...</th>
</tr>
</thead>
<tbody>
<tr>
<td>does not need to edit the text in the Dynamic Objects</td>
<td>can work with the Illustrator document without noticing any difference with a regular Illustrator document.</td>
</tr>
<tr>
<td>needs to edit the text of a Dynamic Object</td>
<td>has to use the <strong>Expand</strong> command on the Object menu in Illustrator.</td>
</tr>
</tbody>
</table>

**Note:** Take care when using the **Expand** command in Illustrator. If you apply the command to other objects in your design (e.g. blends, text, ...) it can cause unexpected changes in your design. For this reason, it is often advisable to release the Dynamic Objects before sending it to any recipients who do not have access to the Dynamic Content plug-in.
7. Styling your Dynamic Objects

There are two approaches to styling the Dynamic Objects in your design: manual styling and automated styling (Style Mapping). In this section you will learn how to apply both these methods, how to solve common problems such as text overflow, and more.

7.1 Two Approaches to Text Styling

The two approaches for styling your Dynamic Objects each have their specific use cases. Obviously, the automated method will save you time in certain situations, but it can only be used efficiently if your source XML file contains presentation-related information. This section further explores the differences, and explains exactly how to apply both the manual and automated styling methods.

7.1.1 What Is Manual Styling?

The easiest - but not always the most efficient - approach to styling the text in your Dynamic Objects is to use the Character and Paragraph palettes in Illustrator.

In other words, applying the manual method - sometimes called the “template-based” styling approach - means that you must select every Dynamic Object in the design and apply the appropriate style formats manually.

Alternatively, of course, you can set up your design with properly styled “dummy text” boxes even before you create Dynamic Objects. In this case, the Dynamic Content plug-in will apply the last style found in the Illustrator object to the text in the new Dynamic Object.

When a Dynamic Object contains multiple styles, Dynamic Content will attempt to preserve these styles whenever the content is updated. In some cases, when there are major changes to the content of a Dynamic Content, this will prove impossible. To alert you to this possible loss of formatting, a special Check Alert will appear in the XML Tree for the affected elements.
7.1.2 What Is Style Mapping?

Style Mapping allows you to automate the formatting of your Dynamic Objects to a very high degree. You apply Style Mapping by linking XML elements to named styles in Illustrator.

**Mapping Elements to Styles**

Using the Style Mapping dialog box, you define the relationship between the XML elements in the linked XML file(s) and named Character and Paragraph Styles in your Illustrator design. Once this relationship has been defined, every Dynamic Object that incorporates a specific XML element will automatically be formatted with the corresponding Paragraph and/or Character Styles.

**Style Hierarchy**

Styles in Adobe Illustrator apply to text objects in a specific order. This is of particular importance when defining Style Mappings, as you will be setting up both Paragraph Styles (e.g. for a ‘p’ tag in the XML file) as well as Character Styles (e.g. for a ‘bold’ tag in the XML file).

Style types have a built-in hierarchy, with the last level overruling all the preceding information:

1. First, the named **Paragraph Style** is applied.
2. Any **Character Styles** applied to a specific section in a paragraph overrule the formatting of the Paragraph Style.
3. Finally, local **overrides** overrule the Character Style. An override occurs when you manually change a property (i.e. change the font size) in the Character palette without updating the Character Style itself.

7.1.3 An Example of Style Mapping

In the following example, we will use some HTML content from a web page that we saved in XML format, link it to our design in Illustrator, and then apply Style Mapping to automatically recreate the styling from the original web page.
Step One: Preparing the Files

First, we save the snippet of HTML (actually it is XHTML, a special flavor of HTML that conforms to the XML standard) as an XML file, so we can link to our design using Dynamic Content.

Then, we prepare some named styles in our Illustrator design: the Paragraph Styles ProdListPara and ProdListHeader, and the Character Style ProdName.

As you can see, the Illustrator text box already contains some dummy text that enables us to test these styles.
Step Two: Linking the XML File

In this second stage, we link the XML file to the Illustrator using the Add XML File command. Then we convert the text object into a Dynamic Object by linking it to the `<body>` element.

The result: we lose all of our carefully applied styling. A warning message informs us that not all the style information could be retained during the conversion. Not all is lost, however, as we will restore the styling using Style Mapping.
Step Three: Setting Up Style Mapping

In this third and final step, we use **Style Mapping**... command on the fly-out menu to define the relationships between the predefined Styles on the one hand, and the XML elements on the other.

In this example, we link the `<p>` tag (which is the basic xhtml paragraph) to the `ProdListPara` Paragraph Style in Illustrator. The `<strong>` tag is also defined as a paragraph, this time using the `ProdListHeader` style, creating a nice title for the product list.

Finally, we link the `<a>` element to the `ProdStyle` Character Style, adding some color to the design.

**Tip:**

If you just need to emphasize certain elements within the document by formatting them in bold, italic or underlined text, you can use the **Character Styling options** Bold, Italic or Underline.
The final result

Once we apply the Style Mapping to our design, the Dynamic Object is automatically formatted correctly. As you can see, in text-heavy designs, this can be a huge timesaver.
7.1.4 Applying Style Mapping

To apply Style Mapping to your document, you need to first define the named Character and Paragraph Styles that you will be using, and then set up the relationships between these styles and the XML elements using the Style Mapping command in Dynamic Content.

Before you can set up Style Mapping, your Illustrator document must contain one or more Character and/or Paragraph Styles.

For more information on Character and Paragraph Styles in Illustrator, please consult the Illustrator Help in the Type section.

Creating a Basic Style Mapping Rule

In this task, you will set up a basic Style Mapping rule.

1. Choose Style Mapping... in the fly-out menu.

   The Style Mapping dialog box appears.

2. Select the element in the Element List that you want to associate with an Illustrator named style.

   Click the Show All button if you want the Element List to display all elements in the linked XML files. Otherwise, only the elements currently in use will be listed.

3. Enable the Preview option to immediately see the results of any mappings you make on the artboard.

4. Do one of the following:

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Then...</th>
</tr>
</thead>
<tbody>
<tr>
<td>map the element to a Paragraph Style</td>
<td>• Enable the Paragraph option; and</td>
</tr>
</tbody>
</table>
### If you want to... | Then...
---|---
• Select a Paragraph Style from the Paragraph Styles list.

**Note:**
You can also select a Character Style for the element. In this case, the formatting of the Character Style will be added to the Paragraph Style (overruling it in case of conflicts).

| **map the element to a Character Style** | • Disable the Paragraph option;  
• Select a Character Style from the Character Styles list. |

5. Repeat step 4 for each of the elements you wish to map.
6. Do you want to save the Style Mapping you just defined?
   • If you do, proceed to step 7.
   • If you do not, proceed to step 8.
7. Click the **Save...** button and save the Style Mapping settings on your computer or network.
The Style Mappings are saved in a special XML format.
8. Click **OK** to apply the Style Mapping.

### Creating a Location-Based Rule

In this task, you will set up a Style Mapping rule that only apply to elements at a specific location in the XML tree structure.

1. Choose **Style Mapping...** in the fly-out menu.
   The Style Mapping dialog box appears.
2. Click **Add...** to create a new rule or **Modify...** to refine an existing rule.
   The **Add** or **Modify** dialog box appears.
3. Select the element you want to format with a style from the **Element Tag Name** list.
4. Select the parent element of the element you want to affect from the **Child Element of** list.
   In the example below the Style Mapping rule will apply only to `<span>` elements that are found within a `<p>` element.
5. Select a Style from the Paragraph Style and/or Character Style lists.
6. Click OK to save the Style Mapping rule.

The selected style is applied to all the instances of the elements that match your selection.

Creating an Attribute-Based Rule

In this task, you will set up a refined Style Mapping rule that only applies to elements that have a specific attribute value.

Tip: Use the [Any Element] option in the Element Tag Name list to create a rule that applies to any element in the design that has a specific attribute or attribute value. For example, you could apply a red italic style to all elements that have the attribute value pair status="to_be_reviewed".

1. Choose Style Mapping... in the fly-out menu.
   The Style Mapping dialog box appears.
2. Click Add... to create a new rule or Modify... to refine an existing rule.
   The Add or Modify dialog box appears.
3. Select the element you want to format with a style from the Element Tag Name list.
4. Select the parent element of the element you want to affect from the Child Element of list.
   In the example below the Style Mapping rule will apply only to <span> elements that are found within a <p> element.
5. Click the + (Add Attribute) button underneath the attributes list.

**Note:** You can only click the Add Attribute button if the XML element you selected in the preceding steps has one or more attributes.

A new row is added to the attribute list.

6. Double-click in the Attribute column and select the attribute on which you want to base this rule from the list.

7. Do one of the following:

   - If you want the Style Mapping to affect only elements that have a specific value for the attribute you selected, choose either Is or Contains from the Is/Contains list and enter a value in the Value column.
   - If you want the Style Mapping to affect all elements on which the attribute you selected is found, regardless of the attribute value, proceed with the next step.

In the example below, only `<span>` elements that have a `class` attribute with the value "style1" will be affected by the Style Mapping rule.
8. Select a Style from the Paragraph Style and/or Character Style lists.
9. Click OK to save the Style Mapping rule.

Mapping an Element to a Character Style

In this task you will map an element or attribute to a character style in Illustrator, such as bold, italic or underlined text, using the Style Mapping feature in Dynamic Content. If you map an element or attribute to a character style, Dynamic Content will automatically select the appropriate variant of the font used for the dynamic object. For example, mapping to a “bold” character style will instruct Dynamic Content to switch to the Bold, Semibold or Black variant of the font in use.

Note:
You can enable or disable the use of Semibold and Black variants in the Dynamic Content Preferences dialog.

1. Choose Style Mapping... in the fly-out menu.
   The Style Mapping dialog box appears.
2. Click Add... to create a new rule or Modify... to refine an existing rule.
   The Add or Modify dialog box appears.
3. Create an element- or attribute-based style mapping rule:
   • For an element-based rule, select the Element Tag Name and if necessary, further refine by selecting the parent element from the Child Element of list.
   • For an attribute-based rule, add an attribute by clicking the Add button, select the matching method from the Is/Contains list, and type the Value.
4. Select the type of character styling you want to apply to the matching elements:
   • Enable one or more of the **Bold**, **Italic** and **Underline** options; or
   • Select a predefined **Character Style** from the list.
5. Click **OK** to save the Style Mapping rule.

The presence of a character styling rule is indicated in the Style Mapping dialog with an asterisk.

Example

This example maps an element `<b>` that also has the attribute `style="bold"` to the **Bold** character style.

```
When applying that style mapping rule, the following code:
```
Automatic Mapping of tab and line break tags

If the XML data use the standard HTML formatting tags `<tab/>` (for tabulator) or `<br/>` (for line break), it’s possible to automatically map these tags to the appropriate formatting in Illustrator.

- Set the **Use `<br/>` element as line break** option to use all `<br/>` tags as a line break
- Set the **Use `<tab/>` element as tabulator** option to use all `<tab/>` tags as a tab in Illustrator.

7.2 Solving Text Overflow Problems

Text overflow occurs when a text object in Illustrator contains more text than can be displayed. Dynamic Objects are especially sensitive to text overflow: their contents can be updated regularly - and all too often without any consideration for the effects the changed content can have for a design.

Luckily, Dynamic Content immediately alerts you to text overflow problems by displaying a special ![Check Alert](icon.png) for every element in which the problem occurs.

1. After updating the contents of an XML file, click **Add Filter**.
2. In the **Filter Type** list, select **Elements with Text Overflow**. Only elements with text overflow are displayed in the XML Tree.
3. Alt+Click on the **Check Alert** icon next to the XML element you want to fix.
   The view in Illustrator zooms to the Dynamic Object(s) linked to the element. If the element is linked to multiple Dynamic Objects, you have to select the object with text overflow manually before proceeding.
4. Do one of the following:
   - Click the ![Reflow Text](icon.png) button and scale the Dynamic Object to allow for the additional text.
   - Use the Illustrator Type tools to reduce the font size of the text in the Dynamic Object.
5. Repeat steps 3 - 4 for each of the elements with text overflow, until the filtered XML Tree does not display any more elements.
8. Dynamic Content Preferences

The Dynamic Content Preferences allow you to determine some XML processing settings.

The Dynamic Content Preferences dialog box

- **XML Standard**
  - The XML Standard option determines the default behavior related to the XML standards. If the GS1 standard is selected (default), the new XML files and XML elements created by the Dynamic Content plug-in and the XML file created by the Export Artwork Response command conform with the GS1 schema. The IPC option changes this behavior to let these files conform with the IPC XML schema.

- **Remove Whitespace**
  - The Remove Leading, Trailing and Duplicate Spaces from Linked XML option determines how Dynamic Content treats whitespace in linked XML elements.
  - If you enable the Remove Leading, Trailing and Duplicate Spaces from Linked XML option, Dynamic Content will remove any whitespace characters preceding or following the text inside an XML element that is linked to a Dynamic Object.
Note:
The space stripping function removes all leading, trailing spaces, except those directly preceding and following a nested XML element. Otherwise, text in nested elements would be rendered “glued” onto the preceding and following text.

Example
Consider the following XML snippet:

```xml
<root>
  <p>This text has leading and trailing spaces.</p>
</root>
```

If the Remove Leading, Trailing and Duplicate Spaces from Linked XML option is enabled, this text in any linked Dynamic Objects will be stripped of the extra whitespaces. However, a single space will be retained before and after the contents of the `<b>` element (“leading and trailing”).

The result:

This text has leading and trailing spaces.

If all spaces surrounding the nested `<b>` element were also stripped, the element would read:

This text has leading and trailing spaces.

8.3 Style Mapping Preferences

The following Style Mapping Preferences can be set in the Dynamic Content Preferences dialog:

<table>
<thead>
<tr>
<th>Preference</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow “semibold” typefaces to be used for automated bolding</td>
<td>Allows Dynamic Content to select the Semibold variant of the current typeface, if it is available, when applying character style mapping. For example, if this option is enabled, text in the Minion Pro Regular typeface will be emboldened using Minion Pro Semibold. If this option is not set, Minion Pro Bold will be used.</td>
</tr>
<tr>
<td>Allow “black” typefaces to be used for automated bolding</td>
<td>Allows Dynamic Content to select the Black variant of the current typeface, if it is available, when applying character style mapping. For example, if this option is set, text in the Minion Pro Bold typeface will be emboldened using Minion Pro Black.</td>
</tr>
</tbody>
</table>
8.4 Use Exact Matching when Updating and Re-linking XML Files

The Use Exact Matching when Updating and Re-linking XML Files option determines how Dynamic Content links a specific XML element to a Dynamic Object in the Illustrator document. By default, this option is disabled.

**Smart or Exact Matching?**

Dynamic Content stores the link between an XML element in the source XML file and the Dynamic Object in the Illustrator design in an "absolute" path. This absolute path contains the complete XML tree structure leading up to the XML element in question. This includes all attribute values of all the elements at a higher level.

By default, Dynamic Content does not use Exact Matching to link an XML element to a Dynamic Object when you update or replace an XML file. Instead, it uses a “smart” matching algorithm to determine the most likely candidate for linking. This algorithm takes into account all of the information in the absolute path, but uses a weighting scheme that allows for small changes in the XML file without breaking all the links to the Dynamic Objects.

If the Use Exact Matching when Updating and Re-linking XML Files option is enabled, Dynamic Content will instead require an exact match to the XML element's path. In this case, if any XML elements in the external file have changed position or received new attribute values, the links to the Dynamic Objects will be lost. Usually, this is not the desired behavior.

**Example**

Consider the following basic XML tree:

```xml
<CopyElement CopyElementType="Usage">
  <body>
    <p lang="de">Automat 40° 60° 90°</p>
    <p lang="ru">############## 40° 60° 90°</p>
  </body>
</CopyElement>
```

If you link the final `<p>` element (`<p style="put_style_here">############## 40° 60° 90°</p>`) to a Dynamic Object, an absolute path to this element is stored. This path includes a reference to the position of the `<p>` element relative to its parent, as well as the value for the `lang` attribute ("ru").

We complete the XML content by adding an additional language:

```xml
<CopyElement CopyElementType="Usage">
```

Note:

If both these options are disabled, Dynamic Content will only search for Bold variants of typefaces when applying character style mapping.

If this option is not set, the Minion Pro Bold will stay untouched.
If we update our Illustrator design:

- with Exact Matching enabled: the last \(<p>\) element’s link to the Dynamic Object will be broken, because the position of this element relative to its parent has changed from being the second child to third child.
- with the default behavior (Exact Matching is disabled): the last \(<p>\) element’s link to the Dynamic Object will be retained. Even though the position of the \(<p>\) element has changed, the fact that its attribute value still matches the original path will result in a weighted, smart match.

### 8.5 Processing Changed XML

The **Always Mark and Update Changed Dynamic Objects Without Asking** option controls how Illustrator responds to differences between dynamic objects and their corresponding XML elements. If this option is checked, Illustrator checks for consistency between dynamic objects and XML elements when it opens a file.

If there are changes, Illustrator changes the dynamic objects and marks them as being changed. If this option is not checked (the default), Illustrator prompts you for action.

### 8.6 Saving Paths to XML Files

The **Always Save Links to IPC XML Files in Document XMP Using Absolute Paths** option controls whether Illustrator saves absolute or relative paths to XML files in the document’s XMP data.

An example of an absolute path would be `file://networkstorage1/current/2496/XMLfile.xml`. An absolute path contains the complete machine and folder structure for the file location.

An example of a relative path would be `file:../../XMLfile.xml`. A relative path contains a pointer to the XML file location based on the location in which the Illustrator document was last saved. If the Illustrator document containing this XMP data were opened on another computer with different drive mapping, the proper XML file would not be found.
9. Appendix: Introduction to XML

This Appendix gives a very brief overview of the basic concepts and terminology of the eXtensible Markup Language (XML) standard.

9.1 What is XML

XML is a text-based markup language that is fast becoming the standard for data interchange.

As with HTML, you identify data using tags (identifiers enclosed in angle brackets, like this: <....>). Collectively, the tags are known as "markup".

But unlike HTML, XML tags identify the data, rather than specifying how to display it. Where an HTML tag says something like "display this data in bold font" ( <b>...</b> ), an XML tag acts like a field name in your program. It puts a label on a piece of data that identifies it (for example: <message>...</message> ).

9.2 Semantic Markup

Since identifying the data gives you some sense of what means (how to interpret it, what you should do with it), XML is sometimes described as a mechanism for specifying the semantics (meaning) of the data.

In the same way that you define the field names for a data structure, you are free to use any XML tags that make sense for a given application. Naturally, though, for multiple applications to use the same XML data, they have to agree on the tag names they intend to use.

Here is an example of some XML data you might use for a messaging application:

```
<message>
  <to>you@yourAddress.com</to>
  <from>me@myAddress.com</from>
  <subject>XML is really cool</subject>
  <text>How many ways is XML cool? Let me count the ways...</text>
</message>
```

The tags in this example identify the message as a whole, the destination and sender addresses, the subject, and the text of the message. As in HTML, the <to> tag has a matching end tag: </to>. The data between the tag and and its matching end tag defines an element of the XML data. Note, too, that the content of the <to> tag is entirely contained within the scope of the <message>../</message> tag. It is this ability for one tag to contain others that gives XML its ability to represent hierarchical data structures.
9.3 Tags and Attributes

Tags can also contain attributes - additional information that is included as part of the tag itself, within the tag’s angle brackets.

The following example shows an email message structure that uses attributes for the "to", "from", and "subject" fields:

```xml
<message to="you@yourAddress.com" from="me@myAddress.com" subject="XML Is Really Cool">
    <text>How many ways is XML cool? Let me count the ways...</text>
</message>
```

Since you could design a data structure like `<message>` equally well using either attributes or tags, it can take a considerable amount of thought to figure out which design is best for your purposes.

9.4 What is IPC XML

IPC XML is a flavor of XML that was designed with the specific purpose of content interchange between brand owners and their suppliers.

It is an open format that borrows concepts from the JDF specification made by the CIP4 organization. The IPC XML schema was designed by the Intelligent Packaging Consortium, an informal workgroup consisting of brand owners, retailers, software vendors, consultants and packaging designers.

Dynamic Content can work with various flavors of XML (XHTML, PIM, SPL, …). The plug-in does not place any restrictions on the XML formats you can use as long as the XML files you add to the palette are well-formed. However, because the IPC XML schema is designed specifically for use in packaging, Dynamic Content has a number of features that make working with IPC-based XML files easier.

9.5 The IPC XML Schema

The root of an IPC XML file is called `<CopyContent>`. It contains 3 sections: `<RevisionHistory>`, `<Project>` information and a `<Copy>` section which contains a number of `<CopyElement>` elements that contain the actual textual contents.

Understanding a sample file

The following XML file clearly shows the three sections in an IPC XML file. All the content is enclosed in the root `<CopyContent>` tag. Following that, you see:

- The `<RevisionHistory>` element, showing that this document is now at version 1.1, after a revision by Mr. Jerry Morris.
The `<Project>` element informs us that the project is called Diapers Test, and is meant for the USA region. The project encompasses 5 Pieces of Art (POAs): there will be five Illustrator documents that use information from this XML file.

The `<Copy>` element lists that text content, including CopyElements for the Brand Name (Pampers), Package Count (ranging from 30 to 50 diapers), and a Generic element that shows how you can add any other textual content to the IPC XML.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<CopyContent xmlns="http://www.esko.com/IPCSchema_1_0" Version="1.0"
AgentName="PG ACT" AgentVersion="1.0">
<RevisionHistory RevisionNumber="1.1">
<Revision Type="Original" Author="Jerry Morris" RevisionNumber="1.0"
TimeStamp="2007-01-31T12:00:00+01:00" />
<Revision Type="Change" Author="Jerry Morris" RevisionNumber="1.1"
TimeStamp="2007-01-31T13:00:00+01:00" />
</RevisionHistory>
<Project ProjectName="Diapers Test" ProjectID="P1" Region="USA">
<POAs>
<POA ID="P1_40" Name="Pampers 40 diapers for US" Locales="en-US es-US"/>
<POA ID="P1_50" Name="Pampers 50 diapers for US" Locales="en-US es-US"/>
<POA ID="P1_50_ES" Name="Pampers 50 diapers for US, Spanish only" Locales="es-US"/>
</POAs>
</Project>
<Copy>
<CopyElement ID="CP001" CopyElementType="Brand Name"
SourceRef="NamesDict" CopySourceRef="21321" POAs="Common">
<body>Pampers</body>
</CopyElement>
<CopyElement ID="CP002" CopyElementType="Package Count" Locale="en-US"
Panels="Front Back" POAs="P1_30">
<body>Contains 30 diapers</body>
</CopyElement>
<CopyElement ID="CP003" CopyElementType="Package Count" Locale="en-US"
Panels="Front Back" POAs="P1_40">
<body>Contains 40 diapers</body>
</CopyElement>
<CopyElement ID="CP004" CopyElementType="Package Count" Locale="en-US"
Panels="Front Back" POAs="P1_50">
<body>Contains 50 diapers</body>
</CopyElement>
<CopyElement ID="CP007" CopyElementType="Package Count" Locale="es-US"
Panels="Front Back" POAs="P1_50_ES">
<body>Contiene 50 pañales</body>
</CopyElement>
<CopyElement ID="CP005" CopyElementType="Generic" Locale="en-US"
POAs="P1_30 P1_40 P1_50">
<body>How are you?</body>
</CopyElement>
<CopyElement ID="CP006" CopyElementType="Generic" Locale="es-US"
POAs="P1_30 P1_40 P1_50 P1_50_ES">
</CopyElement>
</Copy>
```
9.6 What is GS1 XML

GS1 is an international non-profit association with member organisations in over 100 countries. GS1 is dedicated to the design and implementation of global standards and solutions to improve the efficiency and visibility of supply and demand chains globally and across sectors. The GS1 system of standards is the most widely used supply chain standards system in the world.

For more information on GS1 eCom, see [http://www.gs1.org/gsmp/kc/ecom](http://www.gs1.org/gsmp/kc/ecom)

For more information on the BMS XML standard, see [http://www.gs1.org/gsmp/kc/ecom/xml](http://www.gs1.org/gsmp/kc/ecom/xml)
10. Glossary

This Glossary gives you an overview of some basic concepts and terminology used in this manual, and what they mean in the specific context of the Dynamic Content plug-in.

**Dynamic Object**
An Illustrator text object that has been linked to an XML element from an external file by the Dynamic Content plug-in. Dynamic Objects have a number of special characteristics, which makes them behave differently than normal Illustrator text.

**IPC**
The Intelligent Packaging Consortium (IPC) is an informal workgroup consisting of brand owners, retailers, software vendors, consultants and packaging designers. The IPC workgroup designed the IPC XML Schema, a standard document structure for exchanging data between brand owners and their suppliers.

**Node**
A node is an item in an XML document’s tree structure. Node is a generic term - this means nodes can have different types: XML element nodes, XML attribute nodes, text nodes, ...

**XML Attribute**
A qualifier on an XML tag that provides additional information. For example, in the tag `<slide title="My Slide">`, title is an attribute, and My Slide is its value.

**XML Element**
A unit of XML data, delimited by tags. An XML element can enclose other elements. For example, in the XML structure:

```xml
<slideshow>
  <slide>..</slide>
  <slide>..</slide>
</slideshow>
```
the `<slideshow>` element contains two `<slide>` elements.

**XML File**
A file with contents in the form of XML markup. XML markup is similar to HTML in that it also uses tags enclosed by angled brackets (example: `<element>`), but it describes the structure of the data instead of how it should be presented or rendered in a web browser.

**XML Schema**
A method for specifying the structure to which an XML document needs to adhere. For example, the IPC XML schema specifies that documents should contain a root element `<CopyContent>`, followed by the child elements `<RevisionHistory>`, `<Project>`, `<Copy>` in this exact order.