

INX2FO User Documentation

Part II - Data Binding and Barcodes

RenderX ©, www.renderx.com

Table of contents

Overview	3
Adobe InDesign data merge overview	3
INX2FO stylesheets overview	3
Dynamic barcodes embedding overview	3
How does INX2FO data binding work?	4
Data source	4
Adobe InDesign data source format	4
INX2FO data source format	4
Implemented features	5
Limitations (on June 15th 2007)	6
How to render PDF file using INX2FO stylesheets	7
Prepare source InDesign Document	7
Static part of the document	7
Dynamic part of the document	7
Choose the most suitable way to make document with data binding	8
Use InDesign data merge	8
Use INX2FO stylesheets data binding mechanism	8
How to process sample document with INX2FO data binding mechanism	8
To get XSL-FO file with data:	8
To get PDF file with RenderX XEP:	9
Note about dynamicTest.xml	9
How to process sample document with Adobe InDesign data merge	9
To get merged INX document:	9
How does INX2FO barcodes support work?	11
Three ways of using barcodes	11
Static barcodes	11
Dynamic barcodes connected to a data source	11
Dynamic barcodes connected to a data source with template in external XML	11
How to insert the barcode into the InDesign document	12
Static Barcodes	12
Creating a XML data definition file for static barcodes	12
Dynamic barcodes connected to a data source	12
Setting up the dynamic barcodes with external data source	13
Dynamic barcodes connected to a data source with template in external XML	13
Setting up the dynamic barcodes with template in external XML	13
References and links	15
Glossary	16

Overview

The part II of INX2FO documentation provides instructions for converting InDesign INX document with data-field placeholders and barcodes to XSL-FO using INX2FO stylesheets. It includes information about implemented features, customization rules and limitations. It demonstrates how to create your own InDesign document with barcodes and using data merge technology and render it to PDF file using INX2FO stylesheets. The documentation also contains introduction to InDesign data merge and glossary of basic terms.

Adobe InDesign data merge overview

Using data merge technology you can add to your InDesign document data-field placeholders linked to external data source records fields. This will allow you to generate package of personalized documents at once. More details about linking data via data merge are available in Adobe InDesign documentation (hereinafter referred to as InDesign documentation).

INX2FO stylesheets overview

INX2FO is a collection of stylesheets to convert InDesign INX documents to XSL-FO. These stylesheets were created by RenderX's development team. For more details about INX2FO stylesheets refer to the part I of INX2FO documentation.

Dynamic barcodes embedding overview

The dynamic barcodes embedding mechanism uses a set of stylesheets to generate barcodes representation. These stylesheets were created by RenderX's development team.

How does INX2FO data binding work?

Data source

Adobe InDesign data source format

A data source file consist of fields and records. It contains the information that varies in each iteration of the target document, such as the names and addresses of the recipients of a form letter. A data source file for InDesign documents can be a comma-delimited file (.csv) or a tab-delimited (.txt) file in which each piece of data is separated by a comma or a tab, respectively (for more details about data merge and data source refer to InDesign documentation). See below example of csv file content:

```
Name,Code,Text,Number
Name1,123,First text string,1
Name2,456,Second text string,2
...
NameN,999,Last text string,N
```

INX2FO data source format

XSL-FO stylesheets can only work with XML files. So you have to convert InDesign source document (csv or txt) to XML format before using data binding mechanism of INX2FO stylesheets. You can do it yourself, or order the interactive tool from RenderX to automate such conversion. To get more functionality for working with dynamic data binding please email to sales@renderx.com. INX2FO data source XML file must have the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <columns>
    <cell1>column1name</cell1>
    <cell2>column2name</cell2>
    ...
    <cellN>columnNname</cellN>
  </columns>
  <row>
    <cell1>value of column1</cell1>
    <cell2> value of column2</cell2>
    ...
    <cellN> value of columnN</cellN>
  </row>
  <row>
    <cell1>value of column1</cell1>
    <cell2> value of column2</cell2>
    ...
    <cellN> value of columnN</cellN>
  </row>
  ...
</root>
```

Note: To ensure safety of your data, always store backup copies of your InDesign data source files.

You can find some examples of XML data source files in "examples\INX" folder of INX2FO package.

Implemented features

INX2FO stylesheets support following data binding and barcodes features such as:

- network or local path to external XML data source file;
- text data fields values processing;
- image data fields values processing;
- record number to start processing from;
- record number to end processing with;
- static barcodes;
- dynamic barcodes connected to a data source;
- dynamic barcodes connected to a data source with template in external XML.

Limitations (on June 15th 2007)

- If XML data source file has special XML symbols inside text then you have to change them to their correct XML equivalent manually (for example: "&" to "&")

How to render PDF file using INX2FO stylesheets

You can render PDF file from INX document with data binding using INX2FO stylesheets and any XSL-FO formatter (for more details about rendering INX documents to PDF refer to the part I of INX2FO documentation).

See instructions below on rendering PDF file using INX2FO stylesheets with XEP formatter. For other formatters your steps may be slightly different.

To obtain PDF file you need to perform the following actions:

Prepare source InDesign Document

Static part of the document

Create the new document in InDesign as described in InDesign documentation.
Add all objects that do not depend on data source.

Dynamic part of the document

Select specific CSV file as data source to the document as described in InDesign documentation (Menu Window -> Automation -> Data Merge; in opened window choose "Select Data Source" (see Fig.1) and select correct csv or txt data source file.

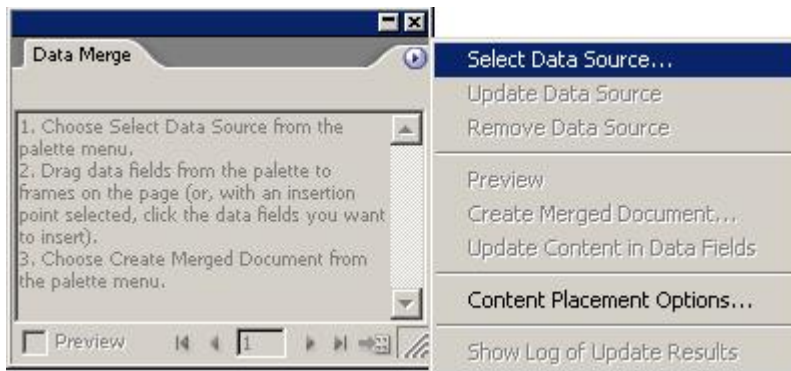


Fig.1 Select data source

Data Merge window will contain all available data source fields. Use drag and drop to take necessary data source fields to corresponding text objects on current document. These fields will be displayed as <<field name>> (see Fig.2).

After document is processed with INX2FO stylesheets, the fields will be replaced with values of corresponding records from data source.

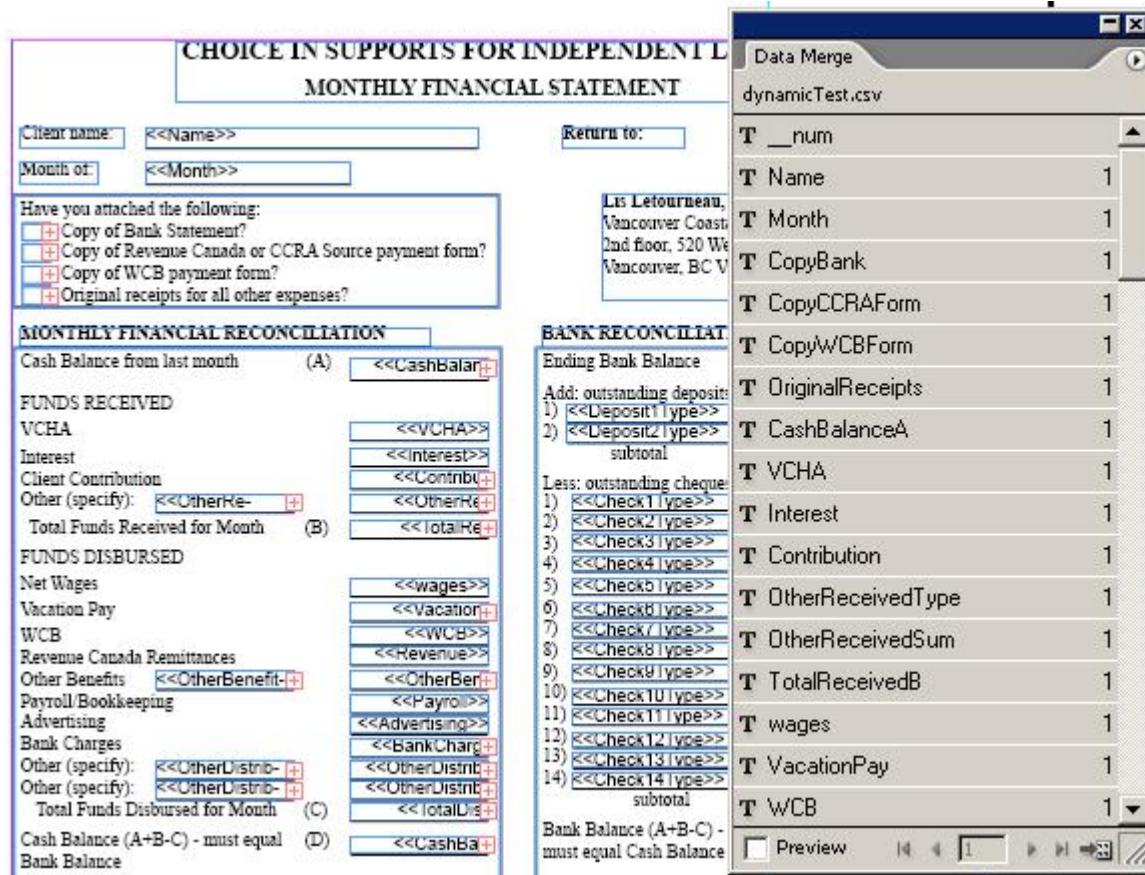


Fig.2 Document with data source bound

Choose the most suitable way to make document with data binding

It is important to know that there are several ways to make data binding with InDesign. After you have bound CSV file data with your InDesign document, you can choose data binding mechanism, from one of the following:

Use InDesign data merge

Make the document with data merged using InDesign, and then convert it to XSL-FO using INX2FO stylesheets. This variant is simpler but needs much resources - all generated pages (e.g., 1000, 10000 or more) need to be placed into the InDesign application and then to INX file. This needs a lot of memory and CPU time.

Use INX2FO stylesheets data binding mechanism

Make only "template" in InDesign, and generate full document with data using only INX2FO stylesheets. This variant is more powerful, fast and scalable. But to do it, you should convert your InDesign data source file to XML-file (see "INX2FO data source format" section of this document for details). Then convert Adobe Interchange (INX) documents into XSL-FO using INX2FO stylesheets using data binding mechanism to make sure that it is easy way to get perfect documents.

How to process sample document with INX2FO data binding mechanism

To process sample document (for example DataSourcePresentation.inx) with external data, located in database file (dynamicTest.xml), run the following commands in "examples\INX" directory to transform document.

To get XSL-FO file with data:

```
JAVA -Xmx512M -jar [path_to_file_SAXON.JAR] -o "DataSourcePresentation.fo"
"DataSourcePresentation.inx" [Path_to_file_inx2fo.xsl]
"data.source=[path_to_file_dynamicTest.xml]" "data.source.record.from=[start
record number]" "data.source.record.to=[end record number]"
```

Last two parameters ("data.source.record.from" and "data.source.record.to") specify range of data source records that will be processed. Those parameters are not required and can be omitted if you want to process all data source records.

For example if you have extracted archive into directory C:\INX2FO, the command string for processing all records for DataSourcePresentation.inx sample would be as described below:

```
JAVA -Xmx512M -jar "C:\Program Files\renderx\xep\lib\saxon.jar" -o
"DataSourcePresentation.fo" "DataSourcePresentation.inx"
"C:\INX2FO\stylesheets\inx2fo.xsl"
"data.source=C:\INX2FO\examples\INX\dynamicTest.xml"
```

For processing data source records from the 2nd to the last, previous example command string would be as described below:

```
JAVA -Xmx512M -jar "C:\Program Files\renderx\xep\lib\saxon.jar" -o
"DataSourcePresentation.fo" "DataSourcePresentation.inx"
"C:\INX2FO\stylesheets\inx2fo.xsl"
"data.source=C:\INX2FO\examples\INX\dynamicTest.xml"
"data.source.record.from=2" "data.source.record.to=LAST"
```

To get PDF file with RenderX XEP:

If you have [RenderX XEP](http://www.renderx.com/tools/xep.html) [http://www.renderx.com/tools/xep.html] installed, you can use it for converting InDesign INX files to PDF format using INX2FO stylesheets.

Here is the command to get PDF file from XSL-FO file:

```
"C:\Program Files\RenderX\XEP\xep.bat" DataSourcePresentation.fo
```

See XEP Reference for details.

Note about dynamicTest.xml

DynamicTest.xml file has been automatically created with application written by RenderX development team. This application is not included into the distribution package. To get more functionality for working with dynamic data binding please email to sales@renderx.com. Otherwise you still can generate XSL-FO files using INX2FO stylesheets from INX files with dynamic data. But in this case you will need to merge data into InDesign target document. See InDesign documentation to learn about merging data in InDesign.

How to process sample document with Adobe InDesign data merge

To process sample document (for example W-9.inx) with external data, located in database file (W-9.csv), do the following steps to transform document.

To get merged INX document:

1. You have to choose "Menu Window -> Automation ->Data Merge".
2. In opened window choose "Create Merged document" (see Fig.3).
3. Then you need to fill the options dialog (see Fig.4). You may not change any field if you do not need it. Press OK.
4. Then you will get a new document (for W-9.inx it will be named W-9-1).
5. You can export the new document to INX format, and then convert it to XSL-FO or PDF file according to instructions in the README.TXT file from INX2FO distribution package.

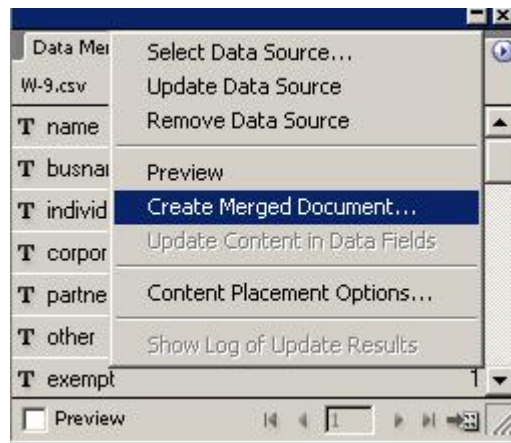


Fig.3 Create Merged document

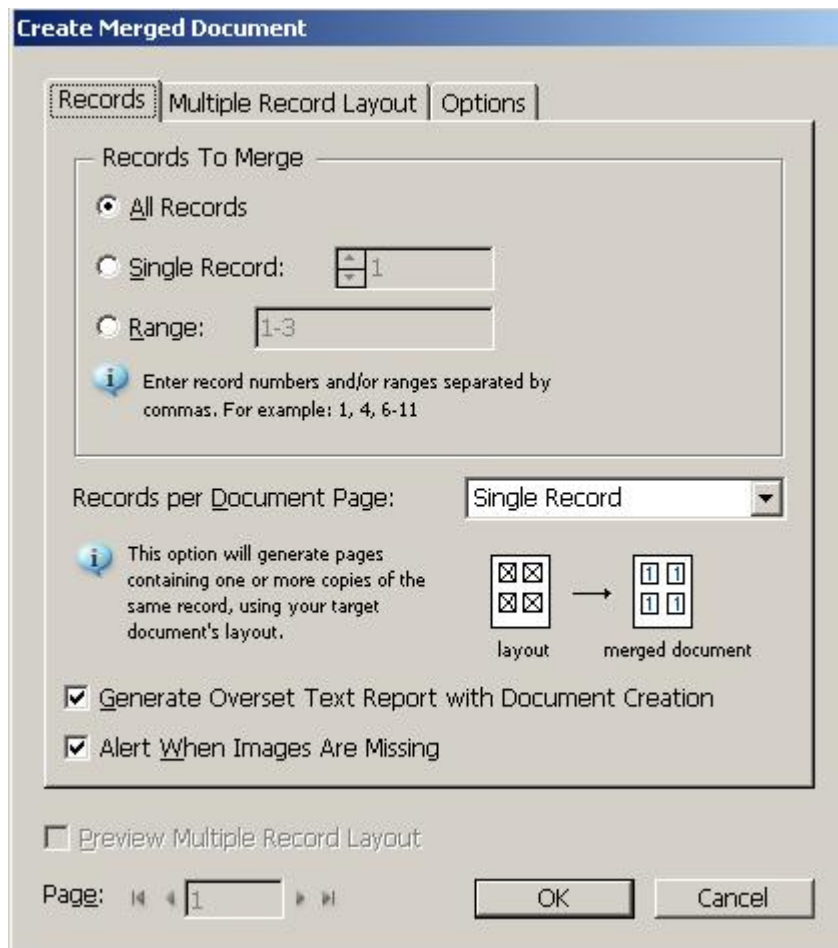


Fig.4 Create Merged document options

How does INX2FO barcodes support work?

Three ways of using barcodes

There are three possible variants of barcodes usage.

Static barcodes

This type of barcodes has all the barcode data embedded into the INX document statically. So the appearance of the barcode is independent of the data source and so all the required data stored inside of the INX file. The data structure to embed is taken from the XML file, with the predefined structure.

Dynamic barcodes connected to a data source

This approach uses data source for all the parameters required to render the barcode. So the data source data controls the type of barcode to render, the value of the barcode, its height and width and so on, in each specific record.

Dynamic barcodes connected to a data source with template in external XML

Another variant of usage the dynamic barcodes with some special ability. If all the barcodes should have static look and differ only by value - this is the case to choose. It uses a data source only for values extraction, and an additional XML file with the template(s) for all the barcodes within a document. Such approach allows centralized maintenance of the barcodes parameters for all the barcodes in the database on the document design time.

How to insert the barcode into the InDesign document

Static Barcodes

Prior to insert a static barcode into the InDesign document, you need to create a barcode XML data definition file, using one of the predefined templates. The XML data definition file is used to extract data. The templates are available in the folder:

[INX2FO installation folder]\templates\barcodes\

Each template file corresponds to a specific type of barcode. It contains a single (sample) XML node that defines all possible tags for the barcode to be rendered by INX2FO stylesheets.

Creating a XML data definition file for static barcodes

To create a barcode XML data definition file:

1. Copy the template file into your project's folder.
2. Open it in InDesign (using File -> Import XML command).

Note: In InDesign you may copy the barcode XML node, and paste it as many times as the barcodes of that type you need (see Fig.5.).

3. When all the XML nodes are added, you need to set the appropriate attributes values by editing each of them (see Fig.6.).

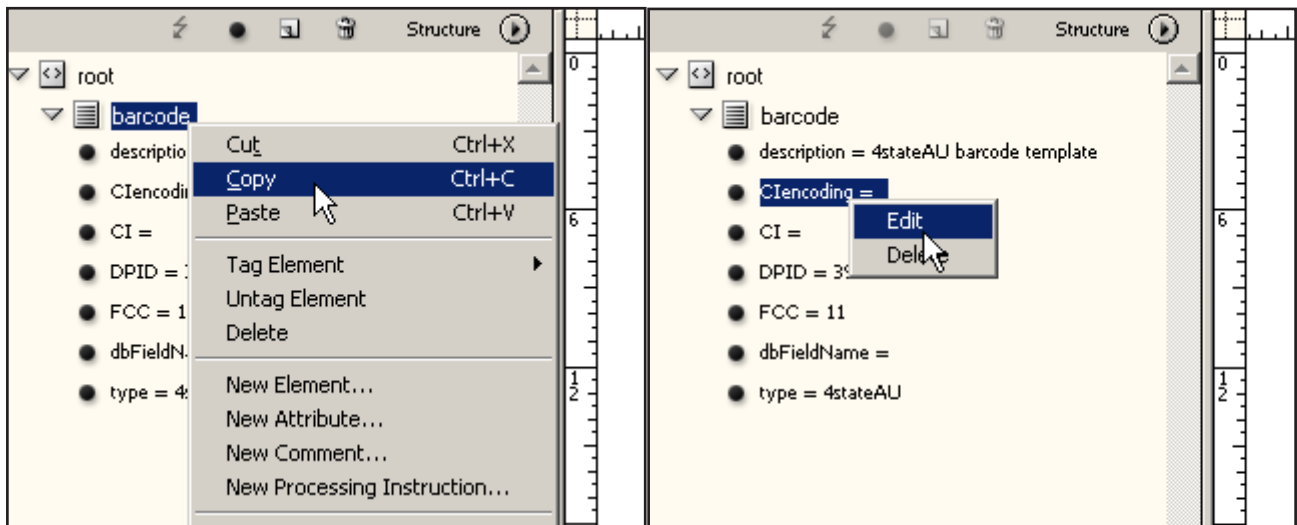


Fig.5. Multiply the XML node template

Fig.6. Attribute value editing

Dynamic barcodes connected to a data source

To create a dynamic barcodes with data-merging capabilities, you need to have a data source in XML format and a CSV representation of that data source (a first XML record in a CSV format) for the InDesign to be able to use it in its data-merging mechanism (at the document design time).

The XML data source should have all the barcodes parameters represented as attributes. The attributes names should be the same as in the corresponding type of barcode template (see "[INX2FO installation folder]\templates\barcodes" folder for templates).

The example of the valid XML data source with barcodes shows the barcode definition cells on the column level and on the row level (see the listing).

If a cell in a row has some attribute missing, the value will be taken from the corresponding column definition cell.

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
  <columns>
    <cell1>RecordID</cell1>
    <cell2>Barcode1Description</cell2>
    <cell13 cellType="barcode" type="Interleaved2of5" module="0.33mm"
      height="15mm" wide-to-narrow="2.4">Barcode1</cell13>
    <cell4>Barcode2Description</cell4>
    <cell15 cellType="barcode" type="Code128">Barcode2</cell15>
  </columns>
  <row>
    <cell1>1</cell1>
    <cell2>This is a barcode #1</cell2>
    <cell13 cellType="barcode" module="0.35mm"
      height="18mm" wide-to-narrow="2.4">12345678</cell13>
    <cell4>This is a barcode #2</cell4>
    <cell15 cellType="barcode">12345678</cell15>
  </row>
</root>
```

Setting up the dynamic barcodes with external data source

To set up dynamic barcodes with external data source follow these steps:

1. Generate a XML data source using your database engine in the form described above.
2. Now you need to generate a CSV representation of the data source with a single record for the InDesign to be able to map the data fields. To do that you can create a CSV file manually with the columns names taken from the XML file "columns" node and the values - from the first row node.
3. After the CSV file is created - you need to select it as a data source for Data Merge. By using command "Select Data Source" on the "Data Merge" panel.
4. When the data source is selected, all available data fields will be displayed in a list.
5. You need to drag and drop each data field on the required text frames.
6. Now you can generate the resulting document with external data source as was described early (see to the "How to process sample document with INX2FO data binding mechanism" section of this document).

Dynamic barcodes connected to a data source with template in external XML

This variant is quite similar to the previous. The whole procedure of the barcode embedding is all the same with only a few differences. There is one more XML file (in addition to the primary data source XML), which contains the secondary barcode parameters. The structure of the XML is the same as in template files, located at "[INX2FO installation folder]\templates\barcodes".

Setting up the dynamic barcodes with template in external XML

To set up dynamic barcodes with external data source follow these steps:

1. Follow 1-4 steps of the previous chapter ("Setting up the dynamic barcodes with external data source").
2. Now you need to run File -> Import XML command and import previously prepared XML template file. This file should contain a XML node for each barcode.
3. In this step you need to create an association between the template and the data merged into the document. To do this, in XML view expand each barcode node and double click the dbFieldName attribute to edit its value. You need to specify a data merging field name as a value of the attribute (see Fig.7.).
4. After the associations to all barcode producing data fields are set up, you may drag each barcode node (from XML view) and drop it to a required text frame where the barcode is expected to render.
5. Now you can generate the resulting document with external data source as was described early (see to the "How to process sample document with INX2FO data binding mechanism" section of this document).

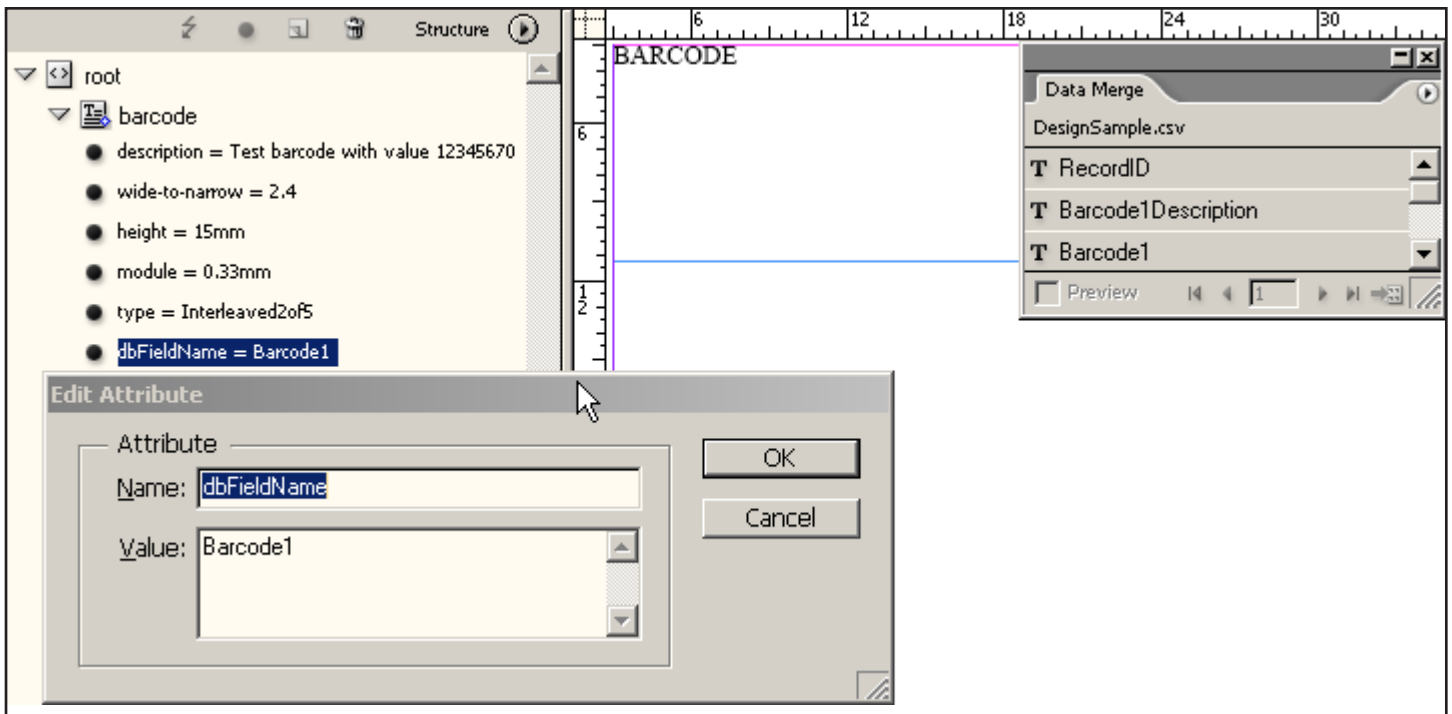


Fig.7. Setting up the association between a data field in a text frame and a template XML node.

References and links

1. [W3C XSL Transformation 1.0 Recommendation](#)
2. [Download trial version of Adobe™ InDesign®](#)
3. [Adobe™ Acrobat® Reader®](#)
4. [RenderX XEP On-line Reference](#)
5. Part I of INX2FO documentation - "INX2FO User Documentation Part I - Stylesheets"
(INX2FOUserDocumentationPart1.pdf)
6. Part II of INX2FO documentation - "INX2FO User Documentation Part II - Data Binding Usage and Barcodes" (INX2FOUserDocumentationPart2.pdf)

Glossary

INX2FO Stylesheets

Collection of XSL stylesheets designed by RenderX to transform INX documents to PDF.

XEP

[RenderX XEP](http://www.renderx.com/tools/xep.html) [http://www.renderx.com/tools/xep.html] is an XML to PDF (XSL-FO) formatter. It takes input in XML, applies an XSL transformation to build XSL Formatting Objects representation, and then formats the Formatting Objects into PDF or PostScript. XEP supports multiple raster and vector graphic formats. Among them, Scalable Vector Graphics (SVG) is an XML-based vector graphics representation, widely used in business applications and for fine typesetting.

XSL-FO

[XSL-FO \(XSL Formatting Objects\)](http://www.w3.org/Style/XSL/) [http://www.w3.org/Style/XSL/] is an XML vocabulary for the formatting of documents. Being part of XSL, the normal way is to produce XSL-FO documents by transforming XML documents using XSLT. Even though the principles behind XSL and CSS (the other Style Sheet Language created by W3C) are quite different, it is planned to align the formatting model between XSL-FO and CSS, so that formatting engines can be based on the same code, both languages can be used to achieve the same results, and formatted results will look identical.

XSL

[XSL \(Extensible Style Language\)](http://www.w3.org/Style/XSL/) [http://www.w3.org/Style/XSL/] is a Style Sheet Language that can be used for displaying XML documents. Using XSL is two-step process, the first step being a transformation of the XML document using XSLT, and the second step being the rendering of the result of the transformation, which is done using XSL-FO. While XSL covers the same application area than CSS, it is much more powerful, because the transformation step (using XSLT) can perform arbitrarily complex transformations of the XML document, while CSS is not able to make any structural changes to the XML document.

XSLT

[XSLT \(XSL Transformations\)](http://www.w3.org/Style/XSL/) [http://www.w3.org/Style/XSL/] is a specialized Programming Language for transforming XML documents. Even though it is part of XSL and as such intended to be used for transforming XML documents into XSL-FO for presentation purposes, it is not limited to this application area. XSLT uses XML syntax (i.e., it is a Programming Language in XML syntax), even though it is based on DSSSL (which uses a Lisp-like syntax). XSLT is particularly interesting in B2B scenarios, where XML documents must be transformed.

INX

[Working with INX file format](#)

[http://partners.adobe.com/public/developer/en/indesign/sdk/working_with_inx_file_format.pdf]. INX is an XML-based format used to serialize and deserialize the InDesign scripting DOM.