

COREP TEMPLATES TO XBRL MAPPER

1. Introduction

▪ Concept

- It's a **low cost tool** which converts **COREP Templates written in Excel** into an **XBRL instance**
 - It's a low cost tool because it has been developed using Visual Basic for Applications in Excel. That implies a quick developing and no training cost. In fact, it can be considered a large Excel macro.
 - To use this tool, the Excel file with the corep templates is recommended to be generated with the coreptest tool, that is a data entry facility that you have available in the corep.info webpage, in order to take the parameters correctly about the operation number, the dimensions, and so on.
 - The data source is in Excel format, so it's an appropriate environment to create a corep templates export possibility.

▪ Benefits

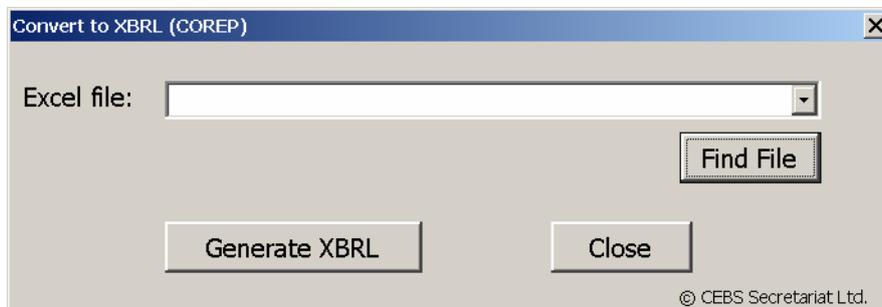
- Easy to use
 - It's a very user friendly tool. You only have to open the file and press a button.
- Efficient, reliable and useful for testing instances
 - The operation is very fast, so it's good for intensive testing.
- Easy to understand its internal operation
 - The Excel macros use Visual Basic for Applications, so the code can be followed perfectly by people who program Excel macros. It's commented in English.
- Autonomous: the taxonomy is included in the Excel file
 - The dimension elements are included in the Excel file, so you don't need to have the taxonomy files available.

- It doesn't have links to external information or databases.
 - No Internet connection required
 - Once the tool has been downloaded, you don't need to be connected to the Internet.
 - Open code
 - You don't need a specific programming environment to see or edit the code. You simply need the Microsoft Excel Application.
- **Construction:**
- Developed in Excel Visual Basic for Applications

2. How to use it

Once opened the tool, the steps that have to be followed are the next:

1. **Open the corep file:** if the file that contains the corep templates is not already open, you can use the **Find File** button or open it directly from the Excel menu.
2. **Open the form** by pressing the “Open form” button.



3. **Select the corep file:** once opened, the file must be selected from the scroll-down box.

4. Press Generate XBRL

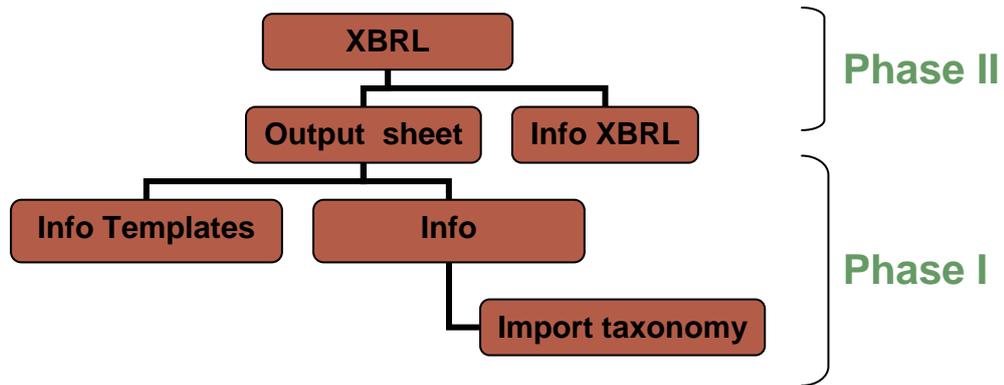
The xbrl instance will be generated **in the same folder as the corep file was located** and **with the same name**, changing the extension to xml.

You can also see the results in the Output sheet. In this sheet you can see the dimension values, units, contexts and other information about each item. It can be useful because you don't need to open the instance to see the results.

3. How it works

We are going to see now how the tool works internally.

The internal operation is divided into two phases:



I) The generation of the Output sheet with all data required

The first phase is the generation of the Output sheet. It's the most difficult step because the corep templates are very irregular. So, the tool must know several parameters about each template. When the tool detects an item, I mean a cell with a valid number or string, the cell where the item is located must have its dimensions perfectly defined.

So, to create the Output sheet, the tool runs throughout all the sheets in the book with the templates and analyzes the ones that are visible. When it finds a cell with an item, it identifies the dimensions depending on the column and the row where it is located.

To make it, as you can see in the flowchart, we need information about the templates, information about the dimensions and the relationships between them.

The information about the templates is in the sheets Templates and TempDef, included in the tool. In the next chapter you can find explained the meaning of each parameter.

The information about the dimensions is included in two other sheets in the tool, named Dim and PriDim. They have been automatically imported from the taxonomy. In these sheets you can see the element names of the dimensions with their units.

There are other sheets that are also explained in the next chapter.

Once the item has been totally identified, a new line is added in the Output sheet with all the information we need for the next phase.

II) The generation of the XBRL instance from the Output sheet.

This phase is easier than the first one because it only takes the information included in the Output sheet and exports it in XBRL format.

4. Settings

Until now, we have seen the way to generate xbrl instances. Now, we are going to see the tool settings. They are already defined, so you don't have to change them actually, but in certain situations it can be useful to have the possibility of adapting the tool to future changes as, for example:

- Updates of the taxonomy
- Changes in the templates
- Use the tool for national extensions

4.1 Template sheet

In this sheet are listed all the possible templates with their main properties.

The parameters in the Template sheet are the next:

- Template Name: the name of the templates as they appear in the sheet names of the templates Excel corep file.
- Template prefix: the prefix of the template to identify the relevant taxonomy.
- Range: The range of cells where the template is extended. A template can be divided into several ranges because the internal typed dimensions must be alone in the range.

- Internal typed dim: if the range has an internal typed dimension it must be set to ‘T’. The internal typed dimension has the characteristic that the user can change the number of rows. If there is another range after a typed range in the same template, it must be set to ‘AfterT’ in order to adapt the range just in case there had been added rows to the template.
- Internal typed dim end: in typed ranges with another range below, this field indicates the first label of the next range.
- Internal typed dim name: the name of the typed dimension.
- Internal typed dim code: there are two types of typed dimensions, depending on whether the dimension values are specified in a column of the template or they are consecutive numbers. In the latter case, this field must be set to ‘number’. In the former, it must be filled in with the column or columns (separated by a colon sign) where the value of the dimension lies.
- Ext dim prefix: prefix of the external dimension. A dimension is considered as external when it is not included in the template range; in this case, there is one template for each value of the dimension.
- Ext dim name: the name of the external dimension.
- Ext dim cell: it refers to the cell where we can find the value of the dimension.

4.2 TempDef sheet

In the TempDef sheet we can define the dimensions for each row and column of the templates. In this case, the templates are listed in the first row, each template taking up five columns. These five columns are divided into two parts:

- The two first columns make up the first group of columns. In this group, the first column contains the prefix of the dimension corresponding to each column in the template, in the same order as they appear in the template; the second column contains the row number indicating the location of the dimension value in the Dim or PriDim sheets.
- The second group of columns contains the prefix or prefixes of the dimension/s corresponding to each row of the template. Each row of the template can have either one or two dimensions. In the latter, the first prefix should be followed by an equal symbol and the row number indicating the location of the dimension value in the Dim or PriDim sheets. If there is only one dimension, the second column would be empty. The third column of this group contains, as in the case

of the first group, the row number indicating the location of the dimension value (first or second) in the Dim or PriDim sheets.

- If a prefix cell is empty, this means that the item is not a valid one.
- A label 'none' may appear in the first position of the first column to indicate that there aren't any dimensions, except, of course, for the primary dimension. This option is used to define the CA template.
- A label 'typed' may appear in one cell of the third column to indicate that in that row a typed dimension begins. That label tallies all the rows that are included in the typed dimension. If there was another dimension after the typed one, the first row of this dimension should be in the next cell of the column.

4.3 Dim and PriDim sheets

The whole taxonomy elements are imported in these two sheets. They have been imported automatically, with the macro ImportTaxonomy. If you want to reload the taxonomy, for example in case of national extensions, you can run this macro. You only need to select the sheet where you want to import the taxonomies (PriDim for primary dimensions or Dim for the other), and execute the macro. You will be prompted for the folder where the taxonomy is.

4.4 ExtDim sheet

In this sheet you may find the relationships between the names and the labels of the external dimensions.

4.5 TypedDim sheet

In this sheet you may find the names of the typed dimensions with more than one element in the identification of the dimension. It is used for the Group Solvency Details template and for the MKR IM Details template.

Fernando Wagener

Financial Institutions Department

Bank of Spain

fernando.wagener@bde.es