><PARSEPORT

# ParsePort XBRL Manual

## Converting Excel files to XBRL and back

A complete guide for using our Excel Add-In and Excel templates to convert your financial data into XBRL. This manual covers the conversion process, error handling and auditing of XBRL files.

**#XBRL** made simple!

# Content

# The ParsePort XBRL Engine

All of our XBRL products are in some way connected to our XBRL Engine via API's. This includes the Excel Add-In. Through the functions of the Add-In, information is sent to and received from the engine. This data can be an Excel file, but it can also be CSV, JSON, TXT, XML and even XBRL. The information can also be received in all kinds of formats.

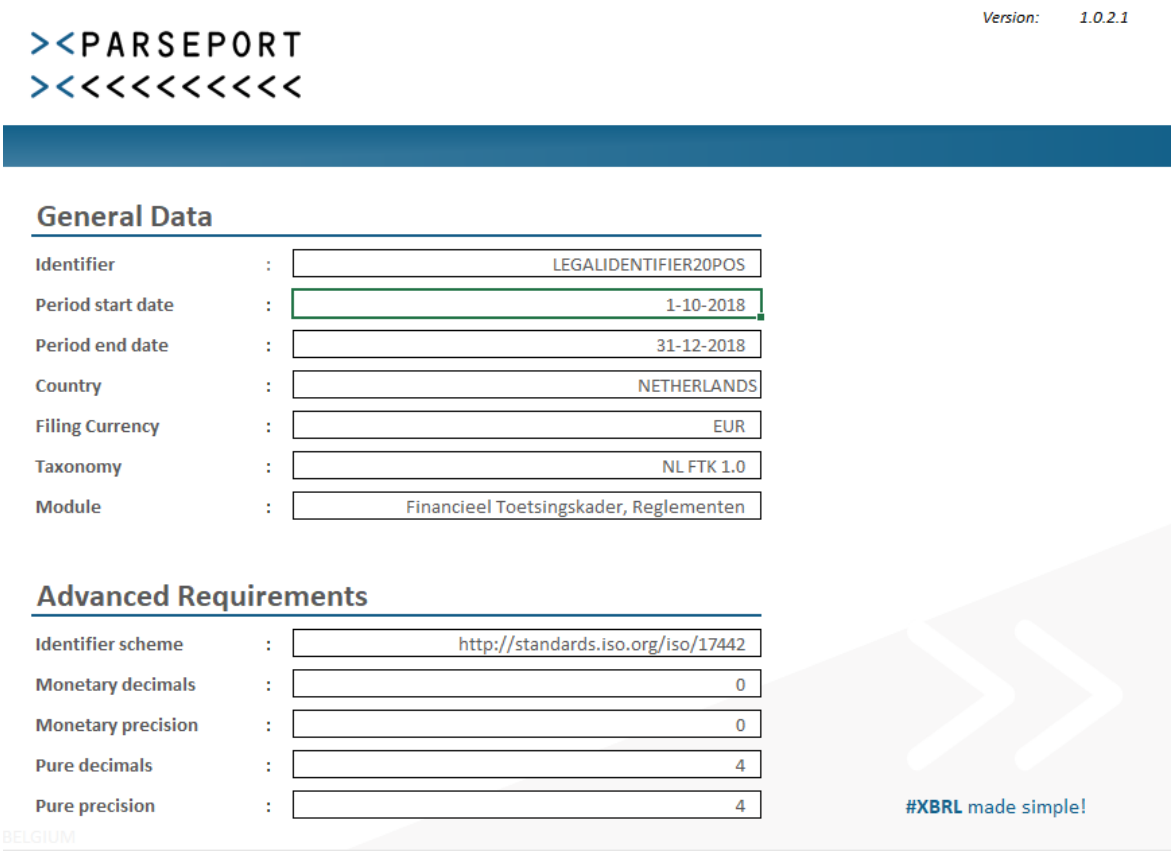
The Excel Add-In connects to the XBRL Engine to convert data, audit files and import data into the Excel file. The Add-In only works with ParsePort modified Excel templates. These can be recognized by the "General Data" sheet and the red labels which can be found next to the tables.

# Preparing an Excel file for conversion

To get a valid XBRL file from Excel data, the data first needs to be filled in in the ParsePort modified template. This requires the user to fill in the General Data sheet and the tables found in the various sheets.

## Filling in the General Data sheet

The General Data sheet looks as follows:




This sheet contains some overall file information, such as the reporting entity's identifier, reporting period, currency and more. Below, all aspects will be explained.

| | |
|---|---|
| **Identifier** | This is the identifier of the reporting entity. In most cases, this identifier is the LEI code without using "LEI/" in front of it, but country specific codes can also be used. Ask your authority which code needs to be used. If this is different than the LEI code, the **Identifier Scheme** most like also needs to be changed. |
| **Period Start- and Period end date** | These are the start- and end date of the reporting period where the file references to. |
| **Country** | The country can be filled in to automatically change country specific elements such as **Currency** and **Identifier Scheme.** |
| **Filing Currency** | The default currency of the file. In relevant cases it is possible to choose currencies per table. Here, the three-letter ISO code of the currency needs to be used. |
| **Taxonomy** | Here, the taxonomy label needs to be filled in. This taxonomy label, together with the **Module**, tells our Engine which taxonomy needs to be used during the conversion. Normally this is pre-filled by us. If this is not the case, please contact us and we can tell which label you need to use. |
| **Module** | The module tells our Engine which specific entrypoint needs to be used within the taxonomy. For instance if the report is an annual or a quarterly report. When the template contains sheets with multiple modules, changing this cell also makes sure that only the sheets of that module are visible in the Excel file. |
| **Identifier Scheme** | Normally, the identifier scheme is pre-filled, but this can be changed based on your country or identifier. The standard identifier is http://standards.iso.org/iso/17442 |
| **Monetary/Pure decimals and -precision** | Normally, our Engine uses the specific cell formatting to determine the number of decimals and the validation precision of the file. This can also be overruled by these settings. For instance, it is possible to set the validation precision to -3. In this case, the validator rounds all monetary values to thousands before comparing numbers.<br>Note: some authorities require a specific precision for the validation. In almost all cases, pure elements (e.g. ratio's and percentages) need to be validated on 4 decimals. If you want the cell specific formatting to determine the precision, you can leave these settings empty. |

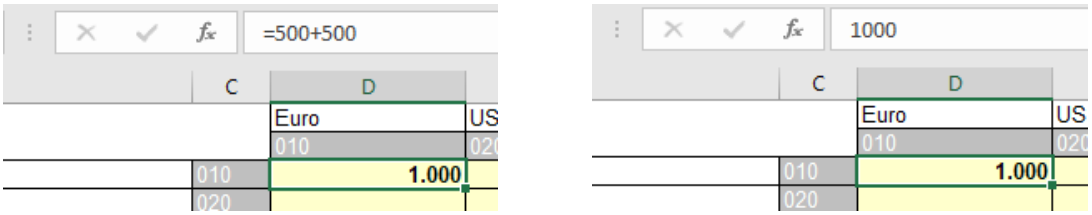
## Setting Filing indicators

Some countries require the reporter to use filing indicators in their XBRL file. Filing indicators show which tables have been reported. A filing indicator can be **true** or **false**.

Our Engine always creates filing indicators (set to true) for the tables which have been filled in in Excel. However, sometimes the authorities want you to report a table, while the table is empty. In this case, the filing indicator is not automatically created. To force our Engine to create a filing indicator, it is possible to use the **"Index" sheet**. (Note: this is not necessary in all taxonomies, since some taxonomies already have a specific table to fill in this information, such as Solvency II.). In this Index sheet, it is possible to state if a specific table or table group has been reported or not. When the Index sheet is filled in, our Engine creates the necessary filing indicator, even when the tables are empty. When the status is set to **"reported"**, the Engine sets the value of the filing indicator to **true**. When the status is set to **"not reported"**, the filing indicator will be set to **false**.

## Filling in the Excel tables

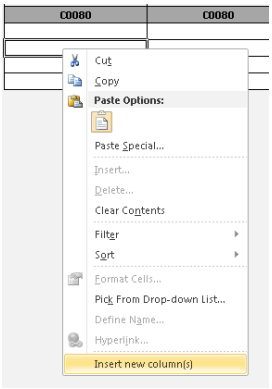
The tables can be filled in by using Excel in its full extend; it is possible to just fill in the values by hand, but it is also possible to use formulas, references to other Excel files or even by importing the data from a data source. When converting, our Engine only looks at the value the way it is presented in Excel. It doesn't look at the formula behind it. If for instance a cell appears empty, but there is a formula in it, it will not add this to the XBRL file. As said before, the Engine also takes the formatting into account in regards with the number of decimals.




*Both options will result in the same value in the XBRL file.*

It is also possible to import data from other Excel files or from XBRL files via the Add-In. This will be explained later on.

## Adding rows and columns




In some tables, it is possible to add additional rows and/or columns. In these cases, we have added a function in the right-click menu. When you right click on a cell in the table where you want to add rows or columns, the menu shows an extra function **"insert new column(s)"** or **"insert new row(s)".** When you choose this function, a pop-up appears asking how many rows or columns you want to add. After that, the rows or columns are automatically added to your template.

# Using the Excel Add-In

To help the user with the conversion process, we developed an Excel Add-In. With this Excel Add-In, it is possible to convert, validate and render XBRL files. It is also possible to import data into an Excel template, clear the data from an Excel template and logging changes in the Excel file.

The Excel Add-In consists of an additional tab in the ribbon, called ParsePort Conversion Tools. This ribbon only appears when a ParsePort modified template is active.

## Convert to XBRL

The first button converts the active workbook into an XBRL file. This takes between 10 seconds to 2 minutes, depending on the size of the report. During the conversion, the file is also validated. In case the file is valid, the pop-up on the right will be shown. When using **"Save"**, the XBRL file is saved on the same location as the Excel template. **"Save As"** will let the user choose where to save the file. **"Show"** will only show the html rendering of the XBRL file and the saving process is canceled when pressing **"Cancel"**. Together with the XBRL file, a html file is saved which is a visual representation of the XBRL file. This file can be used to check if the conversion process has converted everything according to expectation.


When the file is not valid, a pop-up is shown saying that there are validation errors. These errors are stored in the **"XBRL Errors" Sheet**. The validation error description is shown according to the syntax of the taxonomy provider. More on this can be read in the **Support** chapter.

It is still possible to continue to save the XBRL file, but it will not be a valid file and it will likely be rejected by the receiving authorities.

## Audit XBRL file

It is also possible to read a previously made XBRL file by converting it into an html form. The Audit XBRL button gives the possibility to view and validate an XBRL file created by any XBRL conversion tool. The viewable form of the HTML looks as follows.

### qes_220_instance

Identifier: 0QP39Z7MWJESXAQYPIDT

Entrypoint: http://eiopa.europa.eu/eu/xbrl/s2md/fws/solvency/solvency2/2017-07-15/mod/qes.xsd

Period: 2017-12-31

Currency: EUR

Hashcode: zrNYIfsVpyENFcarH//SI6ZLPBRh1wJCdkbak6jaVI4=

**Filing Content**

| Filing Indicator | Value |
|---|---|
| S.01.02 | true |
| S.05.01 | true |
| S.06.03 | true |
| S.08.01 | true |
| S.08.02 | true |
| S.12.01 | true |
| S.17.01 | true |
| S.23.01 | true |
| S.28.01 | true |
| S.28.02 | true |
| SE.01.01 | true |
| SE.02.01 | true |
| SE.06.02 | true |
| E.01.01 | true |
| T.99.01 | true |

**E.01.01.16.01**

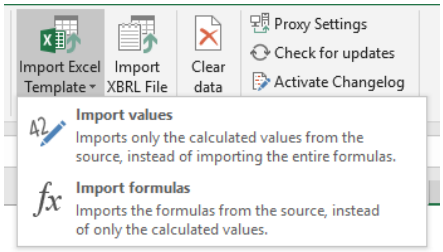
| EC0010<br>E.01.01.zz.01 line identification | EC0020<br>Issuer country | EC0030<br>Currency | EC0040<br>Total Solvency II amount | EC0050<br>Accrued interest | EC0060<br>Par amount |
|---|---|---|---|---|---|
| 1 | ARGENTINA | XFU | € 775305681.99 | € 314846562.76 | € 41755025.42 |

**S.01.02.01.01**

| Row | Item | C0010<br>Values |
|---|---|---|
| R0010 | Undertaking name | cw kg lslx rh |
| R0020 | Undertaking identification code and type of code | ugee eb xm bu ejfvif |
| R0040 | Type of undertaking | Undertakings pursuing both life and non-life insurance activity - article 73 (2) |
| R0050 | Country of authorisation | COLOMBIA |
| R0070 | Language of reporting | Bashkir |
| R0080 | Reporting submission date | 2015-03-02 |
| R0081 | Financial year end | 2002-05-23 |

This shows the general data, such as the identifier, report type, period and currency. It also shows the unique hash code of the file. Below that, the data is shown similar to the Excel tables.

## Import Excel template




When different users create different parts of a report, it is possible for them to use different Excel templates. It can be combined by importing the data from different files into one file. It is possible to either import the values or the formulas. The importing is based on the red labels next to the tables. Therefore it is only possible to import data from ParsePort modified Excel templates. When importing, the user is asked if all data needs to be deleted. When saying yes, all the data will be cleared from all tables. Otherwise, the data is only cleared from the tables which will be imported. **This means that it is not possible to import parts of tables, while other parts have already been filled in.**

Furthermore, data will only be imported from sheets that have been made visible in the source. Invisible sheets will be skipped.

## Import XBRL file

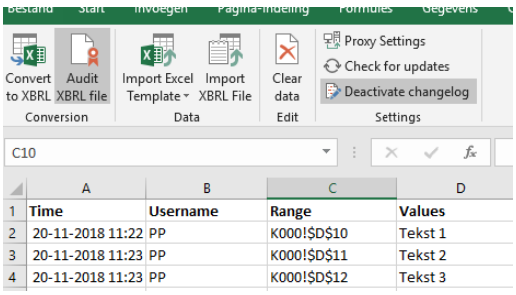
It is also possible to import a previously created XBRL file, or a file from another XBRL conversion tool. This makes it possible to create an adjusted XBRL file, based on a previous version. When choosing an XBRL file to import, make sure that the current template contains the templates of the XBRL file. Otherwise, the application will state that the tables in the XBRL file can't be found in the Excel file.

## Clear data

The Clear Data button clears all values and formulas from the tables in the Excel file. **Note: When clearing the data this way, it is not possible to use ctrl+z to retrieve the values and formulas again.**

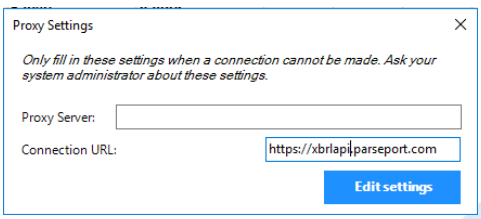
## Changelog

If the Excel template is opened as an administrator, it is possible to activate or deactivate the changelog. Activating the changelog will create an additional worksheet in the file. In this sheet, all changes are shown. To deactivate the changelog, you can press the button again. This is only possible if Excel is used as an administrator.




## Proxy Settings

These settings only need to be changed when the internet connection uses a proxy server. Ask your system administrator which address needs to be filled in. **Note: Always start the server address with http(s)://. Otherwise the address will not be recognized.**




The connection URL normally needs to be https://xbrlapi.parseport.com. This can be changed when using our on-premise services. Please contact ParsePort before you make any changes to the connection URL.

# Support

## Introduction

In general, financial reports are large and complicated. This means that there is a high possibility of errors caused by insufficient knowledge of the XBRL taxonomies. Fortunately, our XBRL Engine validates the XBRL files and returns validation errors, if there are any. However, since we use the error labels set in the taxonomies, not all of them are clear and understandable. For instance, this kind of error could probably easily be solved:

> BV114-1: [ (r0120)] {[S.23.02.01.01, c0010]} = {[S.23.02.01.01, c0040]}.

However, an error like this:

> In context 'xbrli:context id="instant_only"', primary item 's2md_met:ei2342' contains invalid hypercubes in all base sets.

Or this:

> The tag: S.26.03.01.04 Table label is missing its end tag. if (!string.IsNullOrEmpty (conversionError.ErrorType)) { (); }

is more difficult to solve without prior knowledge.

In this document, we will discuss the different types of errors and provide insights on how to solve them.

## Types of errors

We identify four different types of errors: Converter errors, dimension/hypercube errors, invalid value errors and assertion/taxonomy errors. During the conversion from Excel to XBRL, the Excel file and the XBRL file are checked for these types of errors in this order.

### Converter errors

This type of error occurs when there is something wrong with the mapping of the Excel file. The Excel file must be mapped for our converter to know where it can find the data. The file is mapped by using tags; red texts within the file starting with a "/". These tags indicate the start and the end of the different tables in the file. When a tag is accidentally deleted, an error like this occurs:

> The tag: S.26.03.01.04 Table label is missing its end tag. if (!string.IsNullOrEmpty (conversionError.ErrorType)) { (); }

This particular tag indicates that the end tag of table S.26.03.01.04 is missing; this means that the end tag has been deleted. To solve this, try to find the original tag in our original, empty template and copy it to the file.

### Hypercube errors

Hypercube errors are technical errors. These errors occur when an element contains an impossible set of dimensions and/or dimension values. When filling in the template correctly, you should not receive an error like this. A hypercube error looks like this:

> In context 'xbrli:context id="instant_only"', primary item 's2md_met:ei2342' contains invalid hypercubes in all base sets.

A couple of things can cause a hypercube error:

- The element is missing a typed dimension: Some of the sheets have a so called "z-axis". This axis contains one or several dimensions. Sometimes a value needs to be filled in in the z-axis (see figure 1).
- A cell has been filled in, which wasn't allowed (see figure 2). These can be shown in multiple ways. For instance with a cross, or with a dark grey background.
- A table has been filled in while the report type doesn't allow it.

Finding out which element causes the error can be difficult, since the error doesn't necessarily show a sheet, row- and column number. In some taxonomies, such as the Solvency II taxonomy, it is possible to download additional documentation to find where these items are located in the file. In these cases, it is best to contact us.

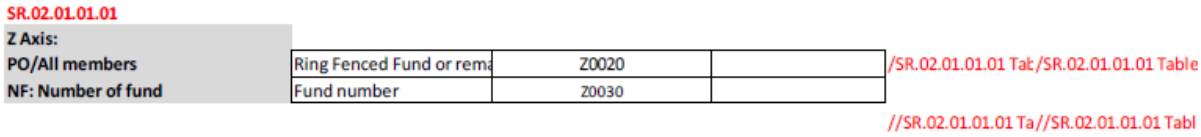



*Figure 1: Z-axis dimension needs to be filled*

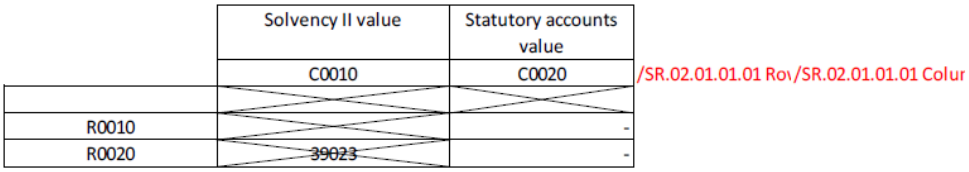



*Figure 2: It is not allowed to fill in a number in row R0020, column C0010*

## Invalid value errors

Some of the elements in the Solvency II tables are so called enumerations. This means that only a certain amount of values are valid to fill in for these elements. Usually a dropdown list is provided in the template and it is possible to use any of the values in the dropdown. When filling in another value, an Excel validation error pops up. However, there are two exceptions when it is possible to fill in an invalid value:

- It is possible to fill in a value while disregarding the necessary capital letters: The Excel validation doesn't look for differences in capital- and small letters. The taxonomy however does see a difference between capital- and small letters. Therefore, when filling in these kinds of elements, look closely if you need to use capital letters.
- It is also possible to copy/paste values into these dropdown cells without getting a validation error. When you decide to copy/paste values from another file, watch closely for the dropdown elements and make sure that the copied value is part of the dropdown list.

An invalid value error looks like this:

> Value 'Not reported' is not allowed for element 's2md_met:ei2292'.

Sometimes, the error doesn't show the filled in value, but a taxonomy code:

> Enumeration value 's2c_MC:x196' in fact 's2md_met:ei2422' is not in the domain of enumeration values for concept 's2md_met:ei2422'.

The enumeration value corresponds with a label, which has been filled in in the Excel file. These labels can also be found in additional documentation. Please contact us in this case, so we can help you further.

### Taxonomy errors

Finally, the file will be checked for business- and assertion errors. These errors are set into the taxonomy. A complete list of these errors can be found in the validation lists, issued by the taxonomy creators. These validations check for instance if certain values equal each other or if certain values add up. Most taxonomies use a similar syntax to explain the error and provide syntax documentation. One of the more extensive syntax explanations can be found on the EIOPA website for Solvency II files:

https://dev.eiopa.europa.eu/Taxonomy/Full/2.3.0/common/EIOPA_Validations_Syntax_2.3.0.pdf

In some cases, it looks like everything in your Excel file is filled in the right way when comparing it to the validation error and you feel like it doesn't make sense that you get the validation error. In this case, it is always best to look at the viewable html file, which is based on the XBRL file. This html file shows what's really in the XBRL file. It could for instance be that a certain table in Excel is missing the red labels and therefore isn't converted. It is also possible that there are duplicate cells in Excel, where one of them has been filled and the other one hasn't. This becomes clear in the viewable html file. If it still looks like the values in the html file would not cause the validation error, you can always contact us.