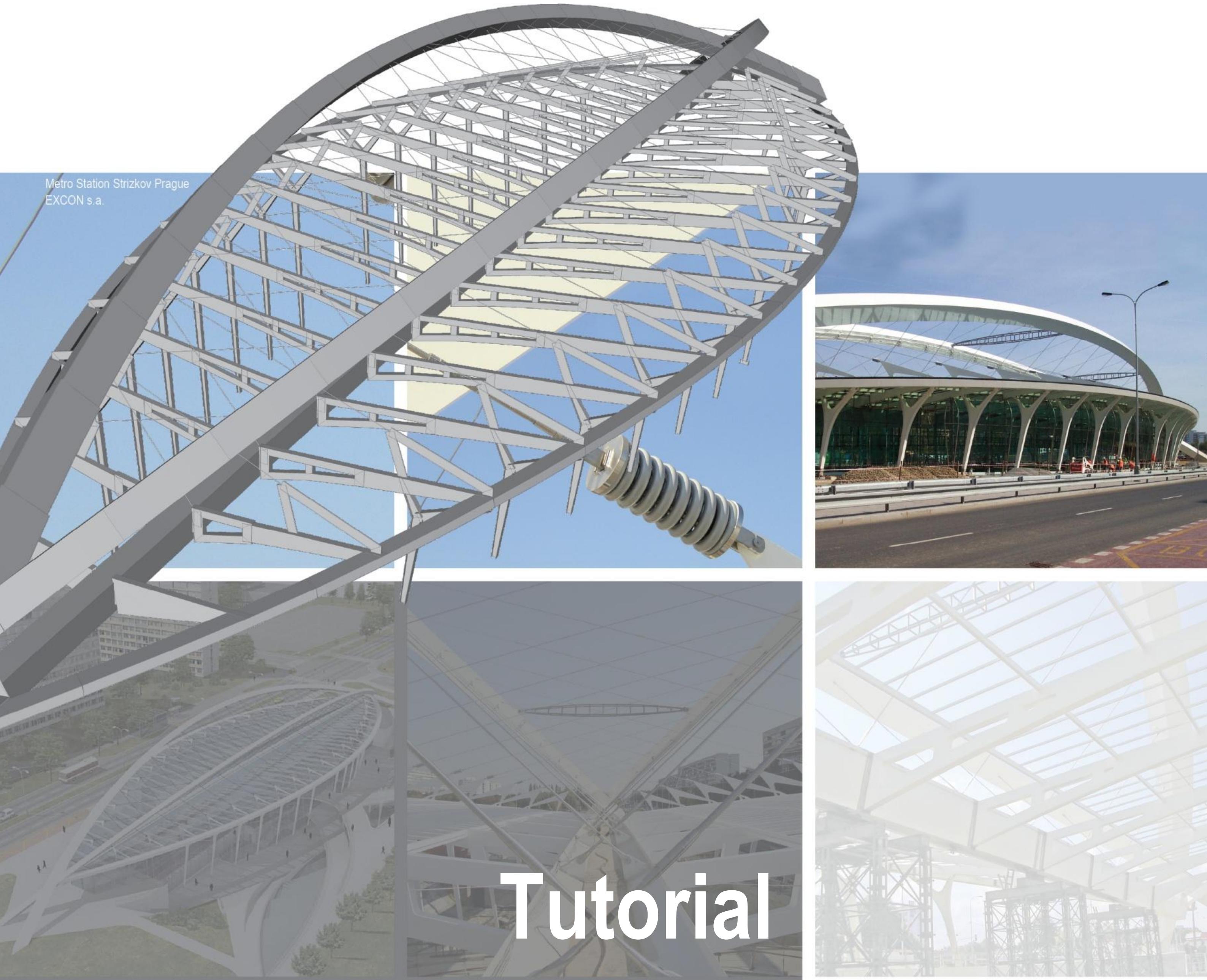


Tutorial

External Application Checks using Excel

Table of contents

Version Information

Welcome to the External Application Checks Tutorial for Excel.

This document provides guided examples and information on the use of the External Application Checks based on Excel.

Version info

Document Title	External Application Checks Tutorial - Excel
Release	2008.1
Revision	06/2008

Introduction

In this tutorial, guided examples and information is given on the use of the External Application Checks using Excel.

In Scia Engineer, a large amount of advanced checks are available for a 1D member: Concrete Reinforcement Design, Steel Code Checks, Aluminium Design, and Connection Checks …

It is off course possible that a user would like a special kind of check, something which is not currently implemented in Scia Engineer.

This is where the External Application Checks module for Excel comes up: using this module, the user can define his/her own type of check and link this to one or more existing Excel files. During the check, the input data from Scia Engineer (like internal forces, members data, dimensions …) are sent to Excel and the results are read back. The Excel file itself can even be shown within the document preview of Scia Engineer!

The basic idea behind the external application checks is the following: the user can define his/her own additional data. This data contains the so called 'mapping' (which properties are send to and read from the external application) but can also contain user defined parameters, check boxes, combo-boxes and so on.

In general, the external check procedure involves the following steps:

Step 1:	Activate the functionality **External Application Checks**
Step 2:	Create **User Defined Additional Data**
Step 3:	Input the **User Defined Additional Data** on members/nodes
Step 4:	Execute the **Custom Check**
Step 5:	Save the **User Defined Additional Data** into a database for future use

In this tutorial, the above procedure for the External Application Checks using Excel is illustrated using four guided examples.

Example 1 In the first example the general principles of the external application checks are explained.

A typical bending check is used as a practical example to illustrate the workings of the module.

Example 2 In the second example, the use of a combo-box is illustrated. In addition, the use of Named cells and output parameters with units are explained.

As a practical case, Flange Induced Buckling as specified in article 8 of EN 1993-1-5 is used.

Example 3	In the third example, the mapping of arrays is explained. In addition, multiple detailed outputs are used. As a practical case, concrete Corbel Design is used.
Example 4	In the final example, the working of slave data is explained. In addition, the use of point data and nodal data is illustrated As a practical case a steel moment resisting connection is used.

Accompanying this document, the reader will find a set of files for each example.

Excel_Example_X.xls	The Excel file which will be used for example X.
Excel_Example_X_Picture.bmp	The picture which will be used for the additional data of example X.
Excel_Example_X_Icon.bmp	The icon which will be used for the custom check of example X.
Excel_Example_X_Initial.esa	The project file for example X without any External Application Check data. The user should use this file when going through this tutorial to follow all the steps described within the document.
Excel_Example_X.esa	The final project file for example X, after completing all the steps explained in this tutorial.

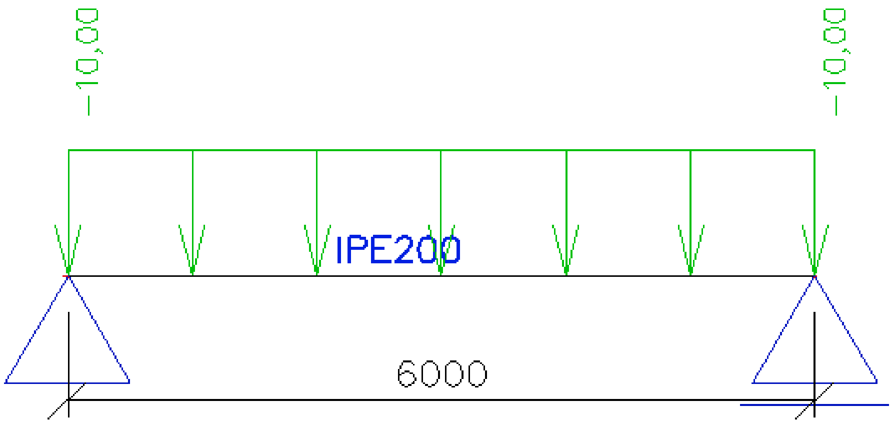
The External Application Checks using Excel supports Excel 97-2003 workbooks (.xls) which do not require any intermediate user interaction.

Example 1: Bending Check

In this first example the general principles of the external application checks are explained.

More specifically a bending check is used as a practical example to illustrate the workings of the module.

In this example, a beam on two supports is modelled. The beam has a cross-section type **IPE 200**, a length of **6m** and is manufactured in **S235** according to **EC-EN**.



One load case is defined, a uniform line load of **10 kN/m**.

The check will be done according to the Excel file "**Excel_Example_1.xls**"

	A	B	C
1	**Bending Check**		
2			
3	**Data from SCIA Engineer**		
4			
5	Moment My	100000	Nm
6			
7	Section modulus Wel	0,0015	m^3
8			
9	Section modulus Wpl	0,002	m^3
10			
11	Yield Strength fy	235000000	N/m^2
12			
13	Safety factor Gamma M	1,1	-
14			
15	Elastic Check ?	0	
16			
17	**Bending Resistance**		
18			
19	MRd	427272,73	Nm
20			
21	**Unity Check**		
22			
23	UC	0,23	-

The bending resistance MRd of the member is calculated using the following formula:

$$MRd = \frac{W \cdot fy}{\gamma M}$$

With:	W	Section modulus
	fy	Yield strength
	γM	Safety factor

The choice of the section modulus as plastic or elastic will be determined from a check box labelled ‘Elastic Check’.

As specified in the introduction, the following steps are required:

Step 1: Activate the functionality **External Application Checks**

Step 2: Create **User Defined Additional Data**

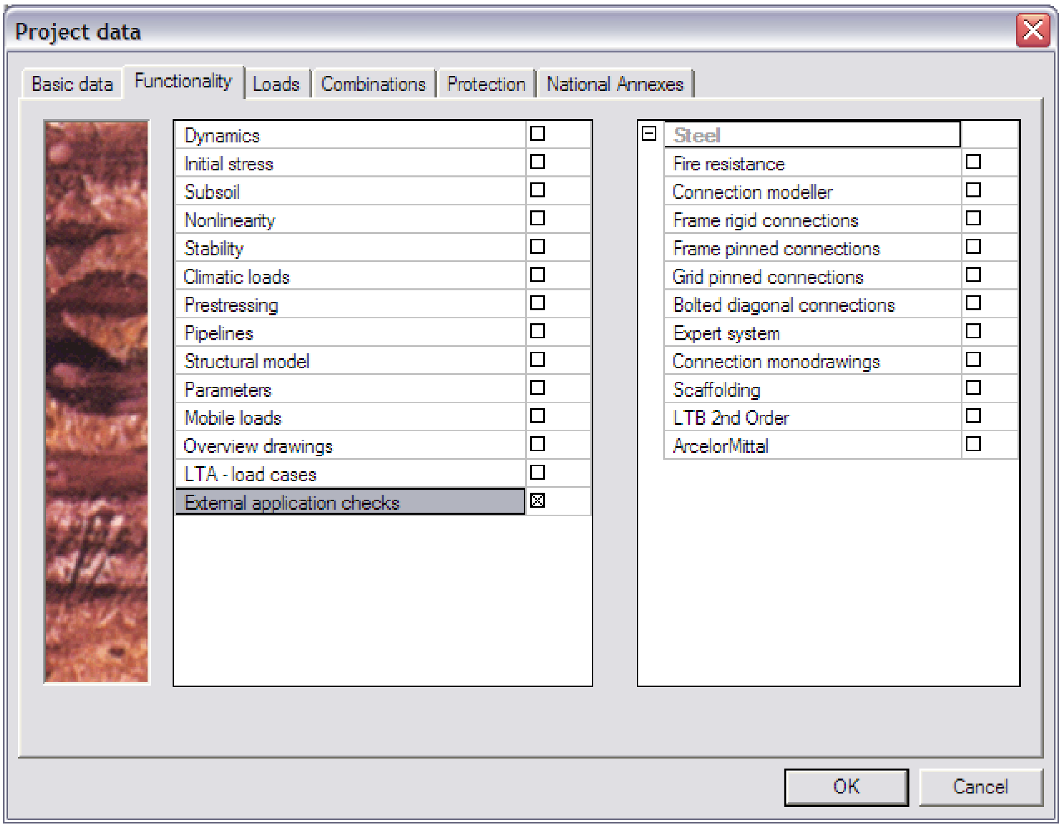
Step 3: Input the **User Defined Additional Data** on members/nodes

Step 4: Execute the **Custom Check**

Step 5: Save the **User Defined Additional Data** into a database for future use

Step 1: Activate the functionality External Application Checks

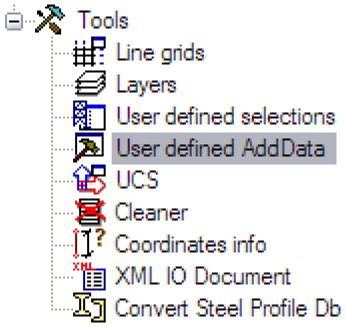
The first step is to activate the functionality **External application checks** on the **Functionality** tab in the **Project Data**.

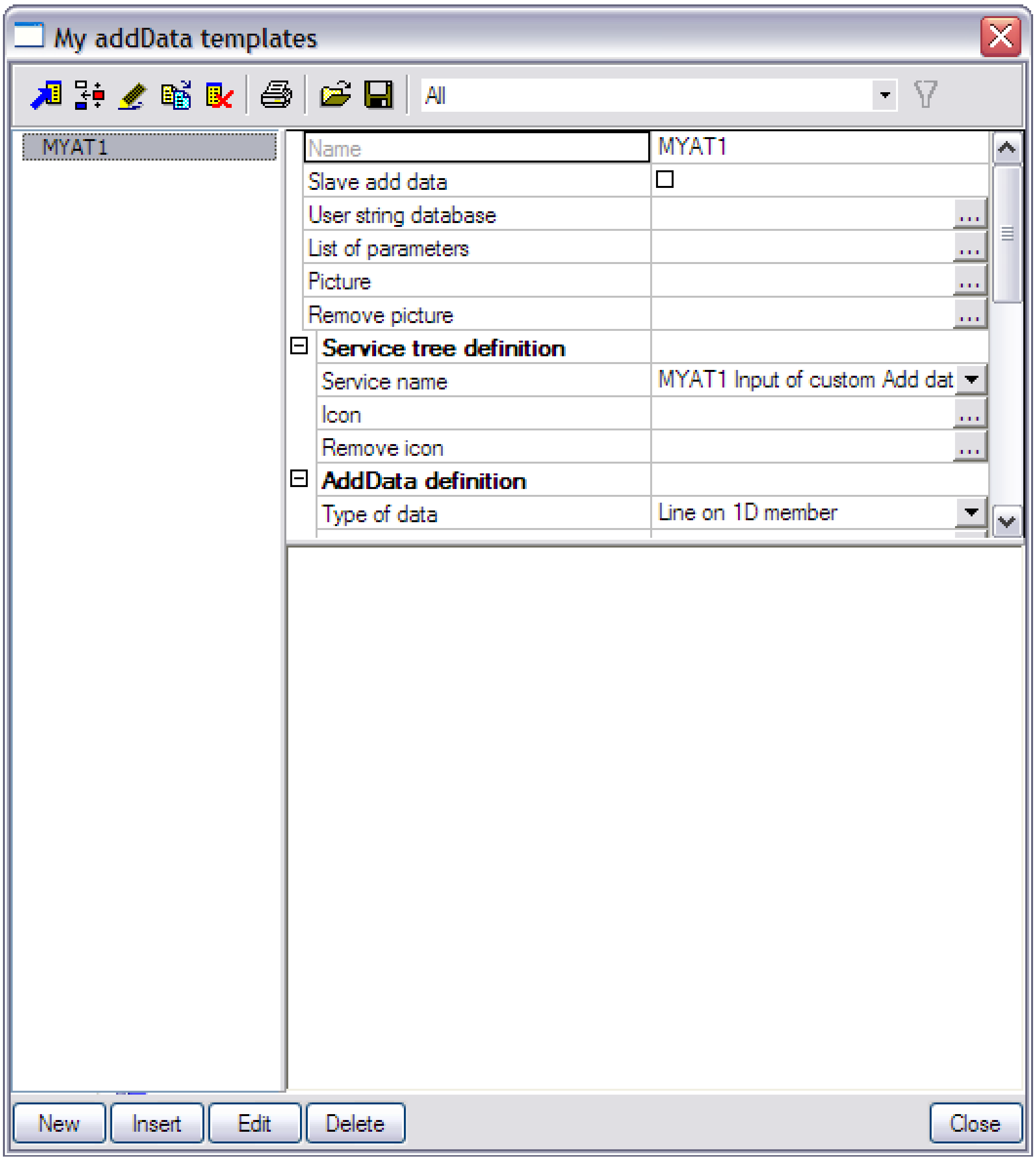


Step 2: Create User Defined Additional Data

In the second step, User Defined Additional Data will be defined.

Through **Tools** > **User defined AddData** the **User Defined Additional Data** Library can be opened.





By default the Library contains a new item labelled ‘MYAT1’. Since in this example a bending check will be defined, this **Name** is changed to ‘Bending’.

The following steps will explain how to define User Additional Data.

Step 2.1 Slave data

The checkbox **Slave add data** can be used to specify that the current additional data is of the type 'slave'.

A distinction is made between 'master' data and 'slave' data.

Master data have all options available: they can be used to send data to Excel and read data back from Excel. The data for this example will be master data: member properties and loading are sent to Excel and a unity check value is read back.

Slave data do not have all options available: they can only be used to send data to Excel, not to read data from Excel. Slave data can then be linked to master data. The check and output options are thus defined in the master data while the input options are defined in both the master and slave data.

A typical example is a beam to column connection: the connection (master data) is inputted on the node between beam and column. In this connection data the check is defined as well as input properties like bolts, welds,...

On the beam and column, slave data will be defined which will send the beam and column properties to Excel.

The use of slave data will be illustrated in 'Example 4'.

For this example only one additional data will be defined and thus the checkbox is not activated.

Name	Bending
Slave add data	☐

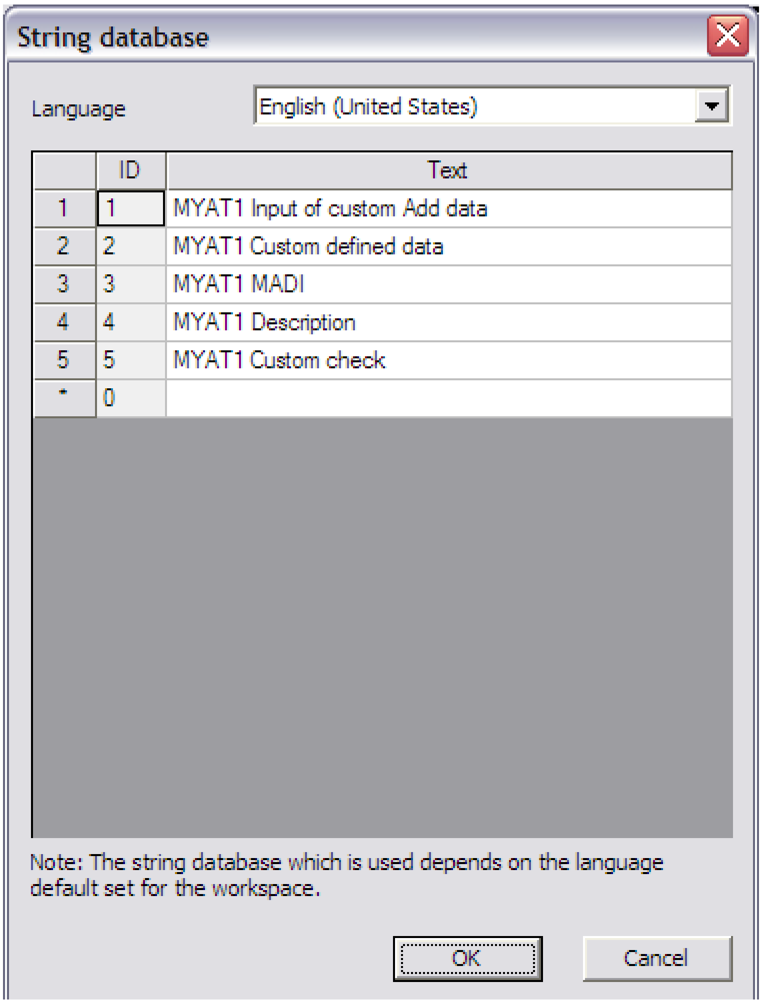
Step 2.2 Define text strings

In Scia Engineer, all text strings of the user interface are saved into a string database. This allows for easy translation of the interface to several languages.

The same logic is used in the User Defined Additional Data: all text strings are inputted in the **User string database** and can then be used when defining the additional data.

For example, when the user creates additional data in the English interface and also provides the German words in the string database, everything will automatically be shown in German in case the user switches to the German interface.

When opening the **User string database** the following contents are shown:

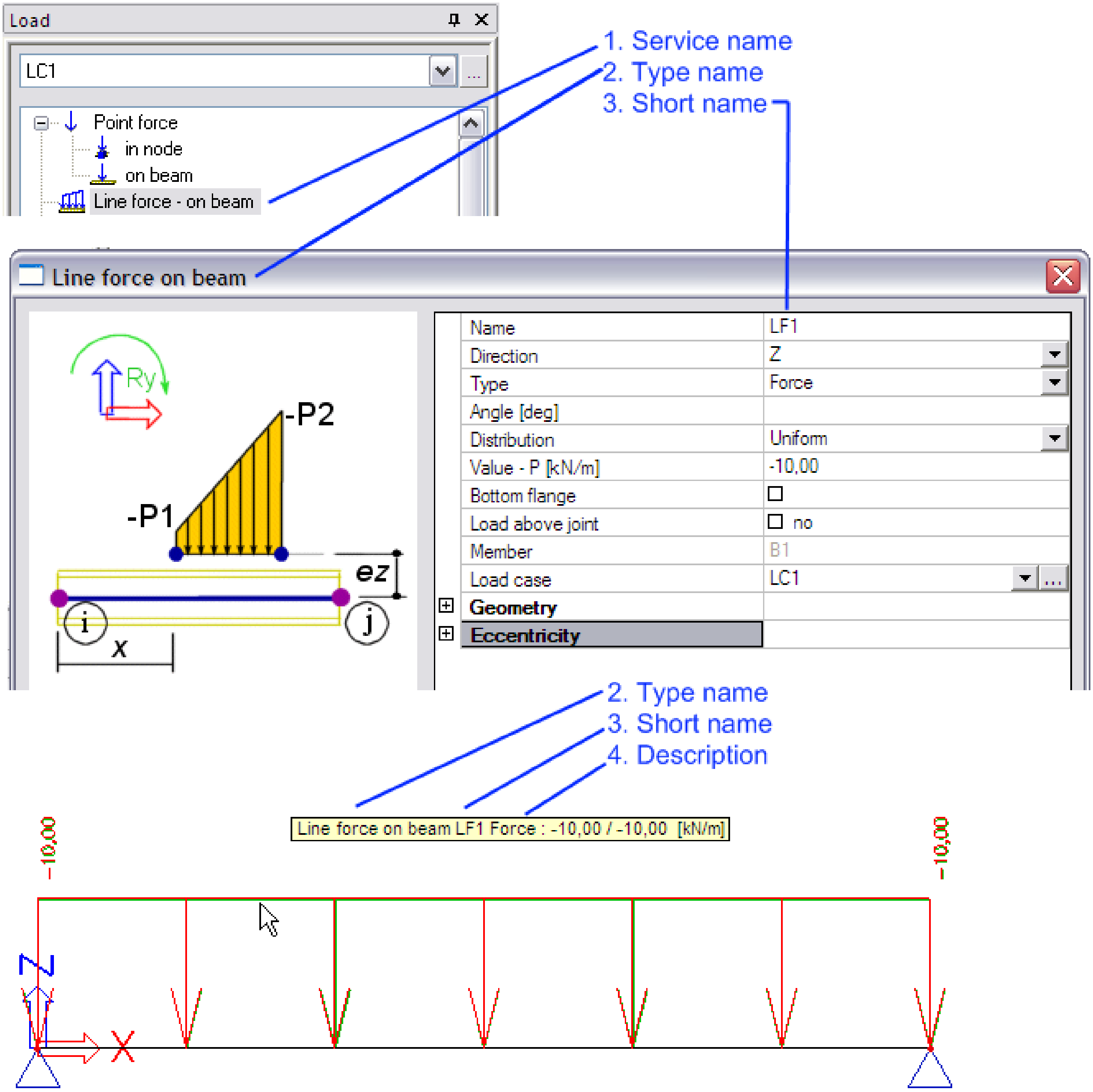


Using the **Language** combo-box the user can switch between different languages. In the grid the different text strings can be inputted.

> ***It is important to use the same ordering of strings when making the input for different languages i.e. the string in English with ID 4 should have its German translation also on ID 4 in the German string database.***

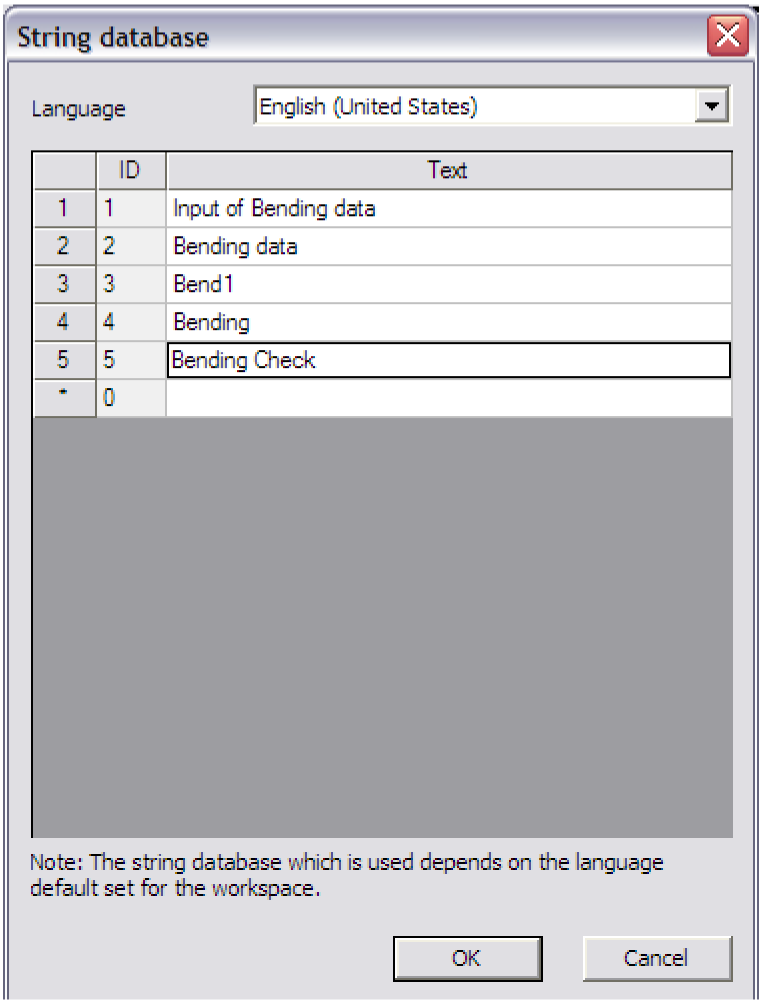
By default, Scia Engineer provides 5 strings which will be used by default for the name of the Service, the Type, the Short name of the additional data, its Description and the name of the Custom check.

The following picture illustrates how these strings are used for a line load:



In this example a Bending Check is being illustrated and therefore the strings are modified as follows:

Type for which the string is used	Default string	String used in this example
Service name	MYAT1 Input of custom add data	Input of Bending data
Type name	MYAT1 Custom defined add data	Bending data
Short name	MYAT1 MADI	Bend1
Description	MYAT1 Description	Bending
Name of check	MYAT1 Custom check	Bending Check



By default the first five strings are used for the types specified above however the user can specify any other string to be used (for example the string with ID 6) as will be illustrated later.

Step 2.3 Define parameters

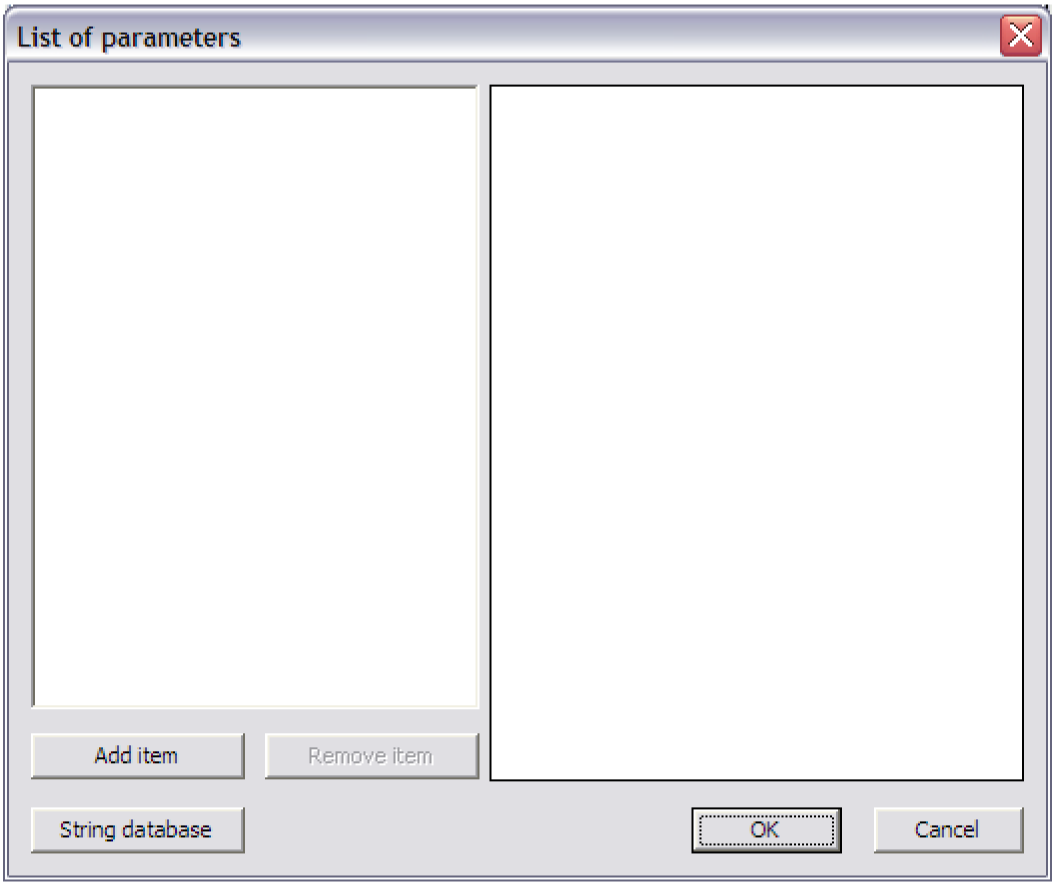
In Scia Engineer a large amount of parameters/properties is available for use with the Excel link (Cross-section properties, member data, internal forces, material properties …). It is off course possible that more parameters are required.

Through **List of parameters** it is possible to create new parameters of the following types:

Parameter type	Description
Number	A numerical parameter For example a length, safety factor, reduction factor…
Text	User defined text
Check box	A checkbox which can either be activated or de-activated. For example Advanced Calculation, Elastic check only,
Combo box	A list from which the user can choose the desired value. For example a list of bolt diameters, a list of thicknesses, a list of weld methods…

As will be illustrated later, for a check box the value 1 will be sent to Excel in case it is activated and 0 in case it is de-activated. For a combo-box the string of the selected line will be sent.

> ***It is not required to define parameters. If the existing data of Scia Engineer is sufficient, no new parameters need to be defined.***



For this example, two parameters will be defined:

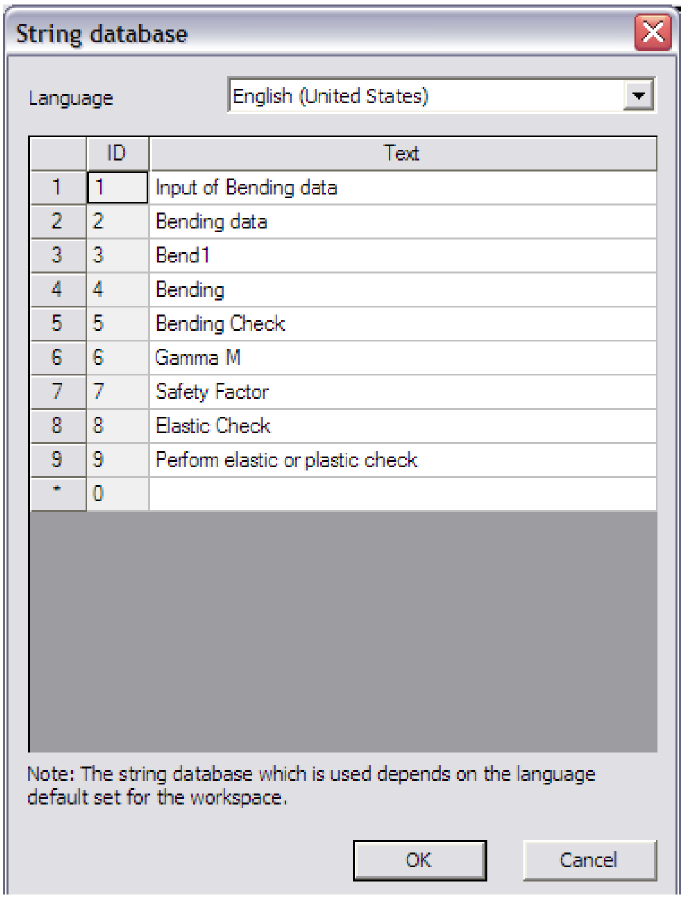
Parameter	Type	Default value
Safety factor Gamma M	Number	1,1
Elastic Check	Check box	de-activated

Through the button **String database** the text string database can be directly accessed. This allows a quick input of the strings required for the parameters.

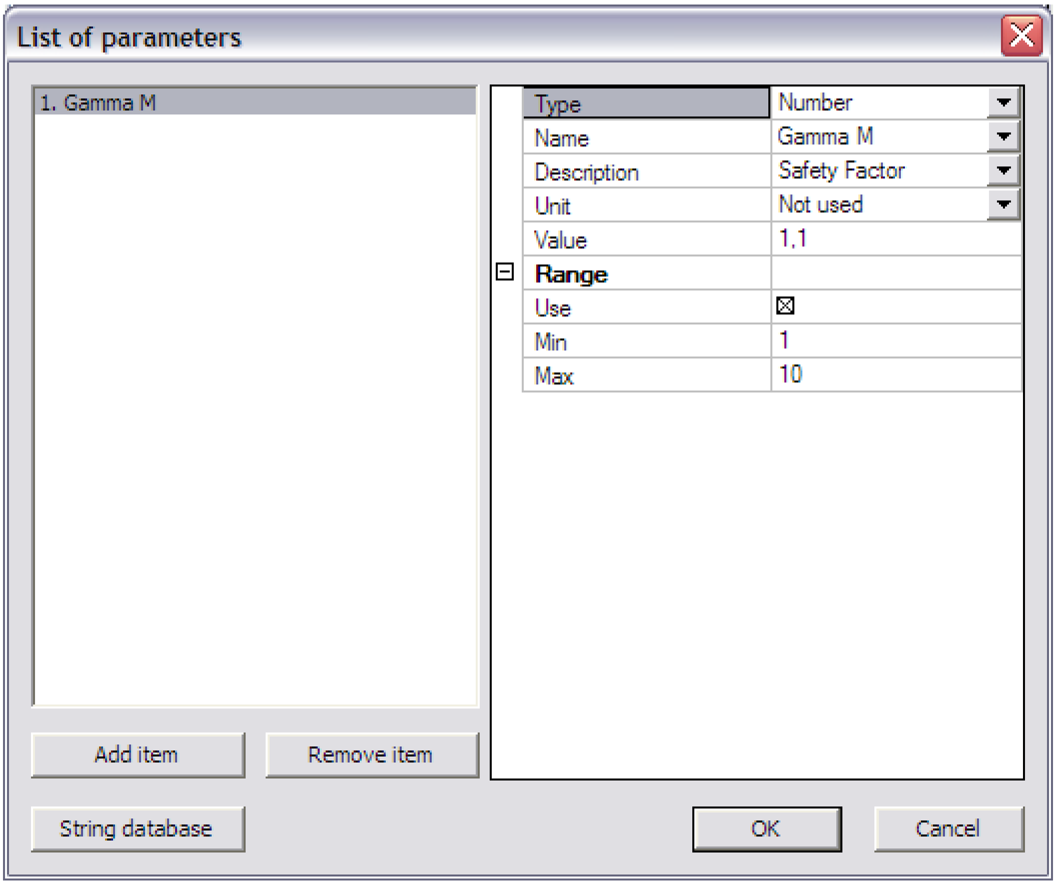
For this example the following strings are added:

Strings used in this example
Gamma M
Safety Factor
Elastic Check
Perform elastic or plastic check

For each parameter a string is thus defined for the name and for the description.



Through the button **Add item** the first parameter, the safety factor Gamma M, is added.



The **Type** field is set to ‘Number’.

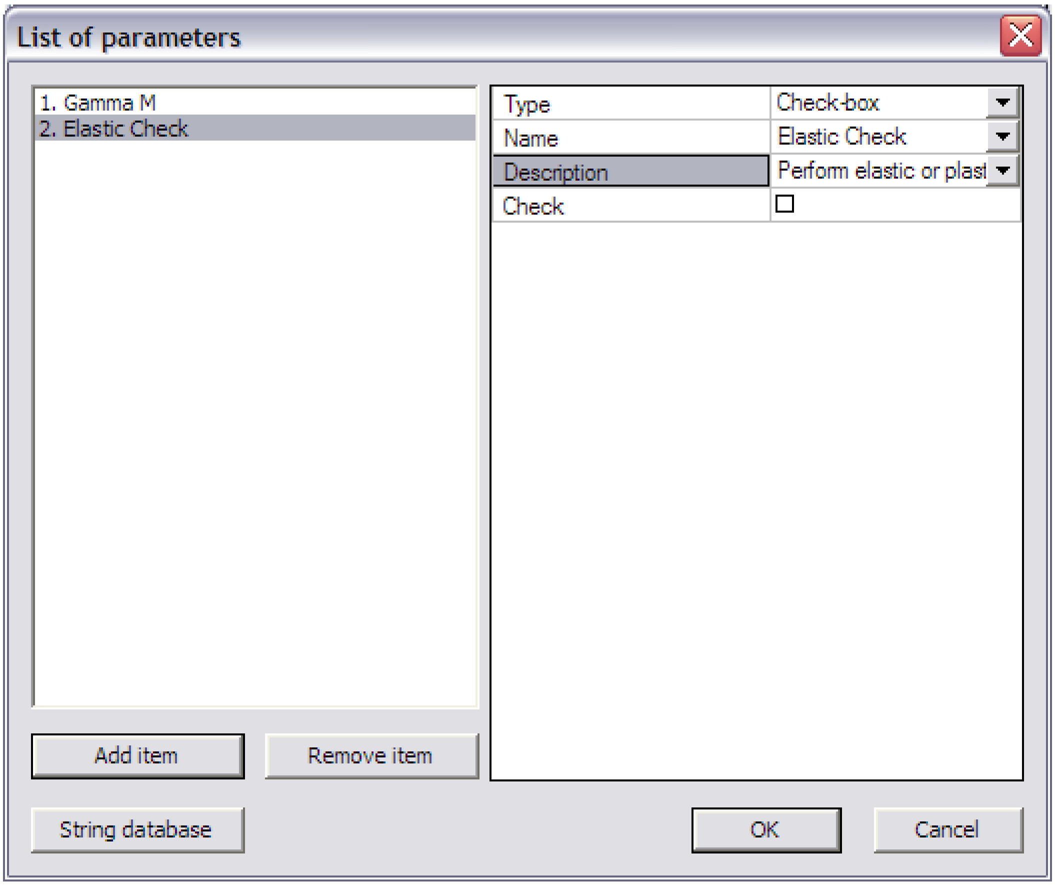
In the **Name** and **Description** fields the respective strings can be chosen from the string database, in this case ‘Gamma M’ and ‘Safety Factor’.

The **Unit** field can be used to specify a unit for the defined parameter. In this case, for the safety factor, no unit is assigned since it concerns a dimensionless parameter.

The **Value** field allows setting the default value for the numerical parameter. In this example the default value for Gamma M is set to **1,1**.

Since it concerns a numerical parameter, the **Range** group allows specifying an input range. This is used to avoid incorrect input values. For this example, the input of the Gamma M value is limited between a minimum of **1** and a maximum of **10**. Only values between these limits are allowed.

In exactly the same way using the button **Add item** the second parameter, the elastic check check-box, is added.



The **Type field** is set to ‘Check-box’.

In the **Name** and **Description** fields the respective strings can be chosen from the string database, in this case ‘Elastic Check’ and ‘Perform elastic or plastic check’.

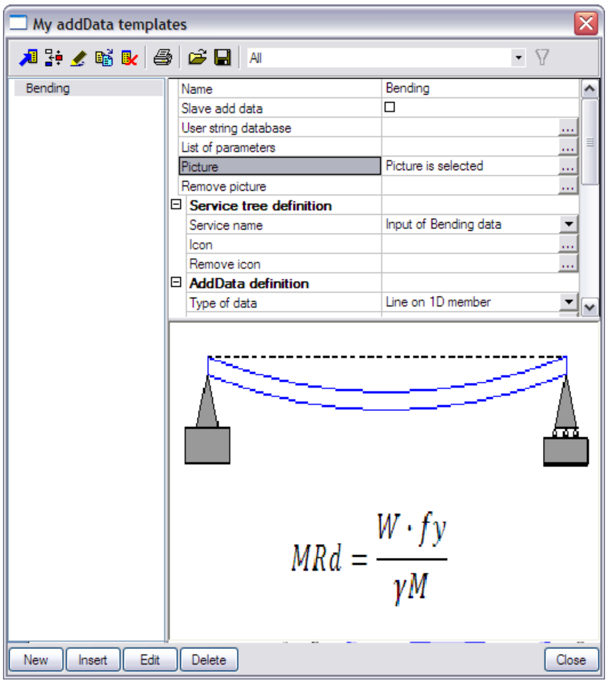
The **Check** field allows setting the default value for the check-box parameter. In this example the check-box will be de-activated by default which indicates a plastic check will be done.

Using the button **Remove item** a parameter can be removed again if required.

Step 2.4 Add a picture to the Additional Data

To clarify the use of the additional data and the defined parameters a picture can be added using the **Picture** button.

In this example the picture **Excel_Example_1_Picture.bmp** will be used.

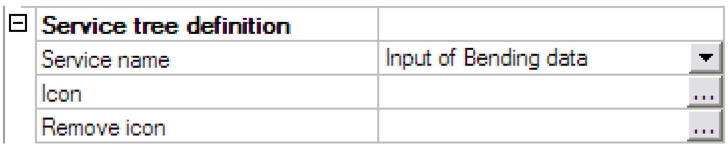


Using the button **Remove picture** the picture can be removed again from the additional data if required.

It is not required to add a picture.

Step 2.5 Define Service Tree

In the next step the Service Tree is defined through the group **Service tree definition**.



As specified in *Step 2.2* the **Service name** is taken automatically from the text string database. If required a different string can be chosen from the database using the combo-box.

To clarify the Service name, an icon can be added using the **Icon** button.

In this example the icon **Excel_Example_1_Icon.bmp** will be used.

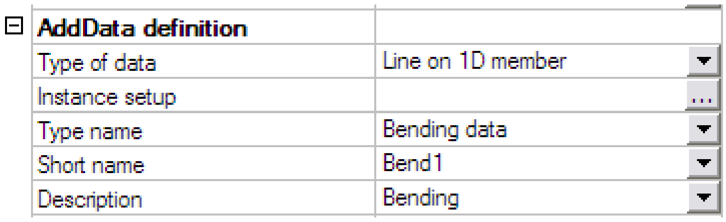
An Icon has to have a bitmap format of 16 x 15 pixels

Using the button **Remove icon** the icon can be removed again from the additional data if required.

It is not required to add an icon.

Step 2.6 Define the Additional Data

Using the data from the previous steps, the additional data can now be defined in the group **AddData definition**.



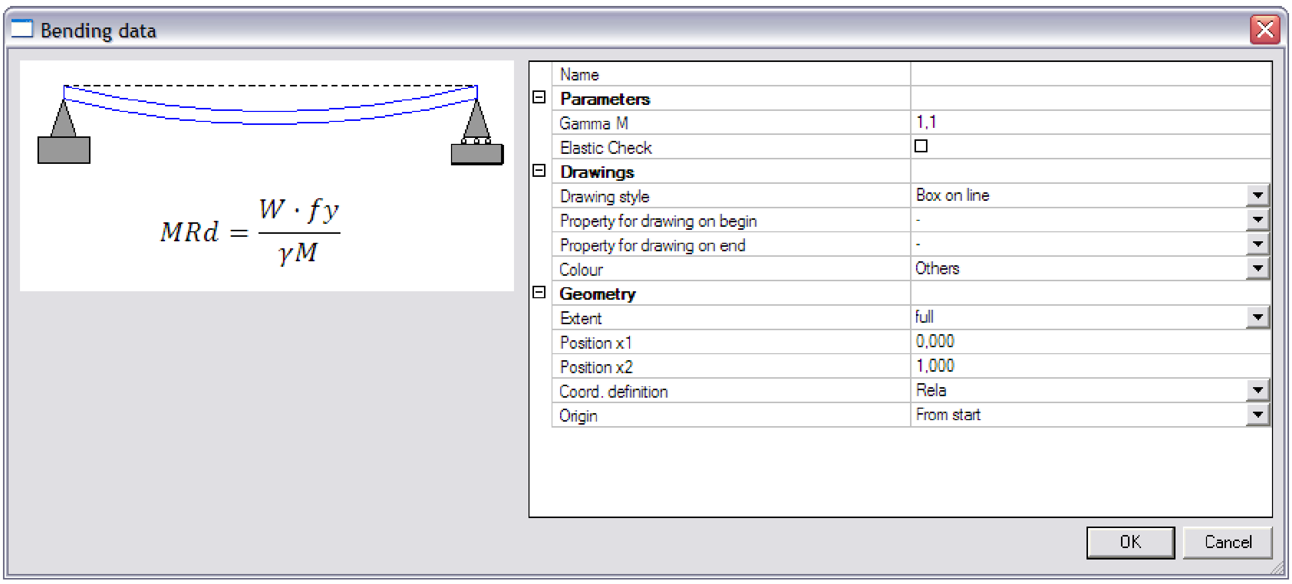
The **Type of data** field allows specifying the type of the additional data:

Type of data	Description
Line on 1D member	Line data is inputted along the length of the member like for example a line load. The check will be executed in each section of the member.
Point on 1D member	Point data is inputted on a specific position along the member like for example a point load. The check will be executed only in the specified section.
In node	Nodal data is inputted on a node like for example a connection. The check will be executed only in the node.

In this example the bending check needs to be executed in each section of the member and thus 'Line on 1D member' is chosen.

The **Type name**, **Short name** and **Description** are taken automatically from the text string database as specified in *Step 2.2*. If required different strings can be chosen from the database using the combo-boxes.

To get an overview of all the data entered in the previous steps the button **Instance Setup** is used.



This dialog shows how the additional data will look like.

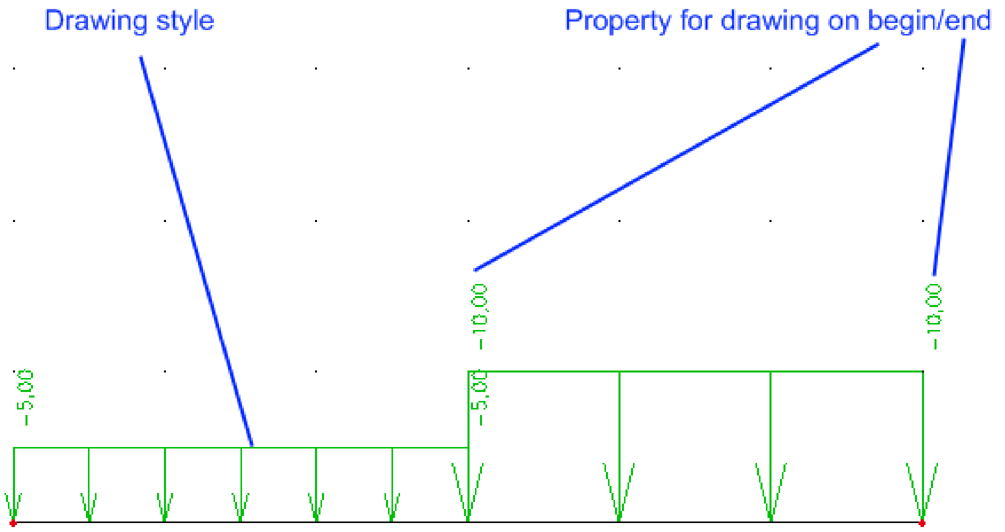
The **Parameters** group shows the parameters which have been defined in *Step 2.3*, in this case the safety factor ‘Gamma M’ and the check-box ‘Elastic Check’.

The **Geometry** group is dependent on the selected Type of Data. For Line data the begin and end positions can be set as for a line load. For Point data a single position can be set as for a point load. For nodal data no position can be set since this data is not inputted on a member.

Since in this example the check needs to be executed in each section, no changes are made to the Geometry settings.

In addition to showing a preview of the eventual dialog, the **Instance Setup** also allows to define the Drawing style of the additional data. These options can be found in the **Drawings** group.

The options **Drawing style** and **Property for drawing on begin/end** are explained for a line load on the following picture:



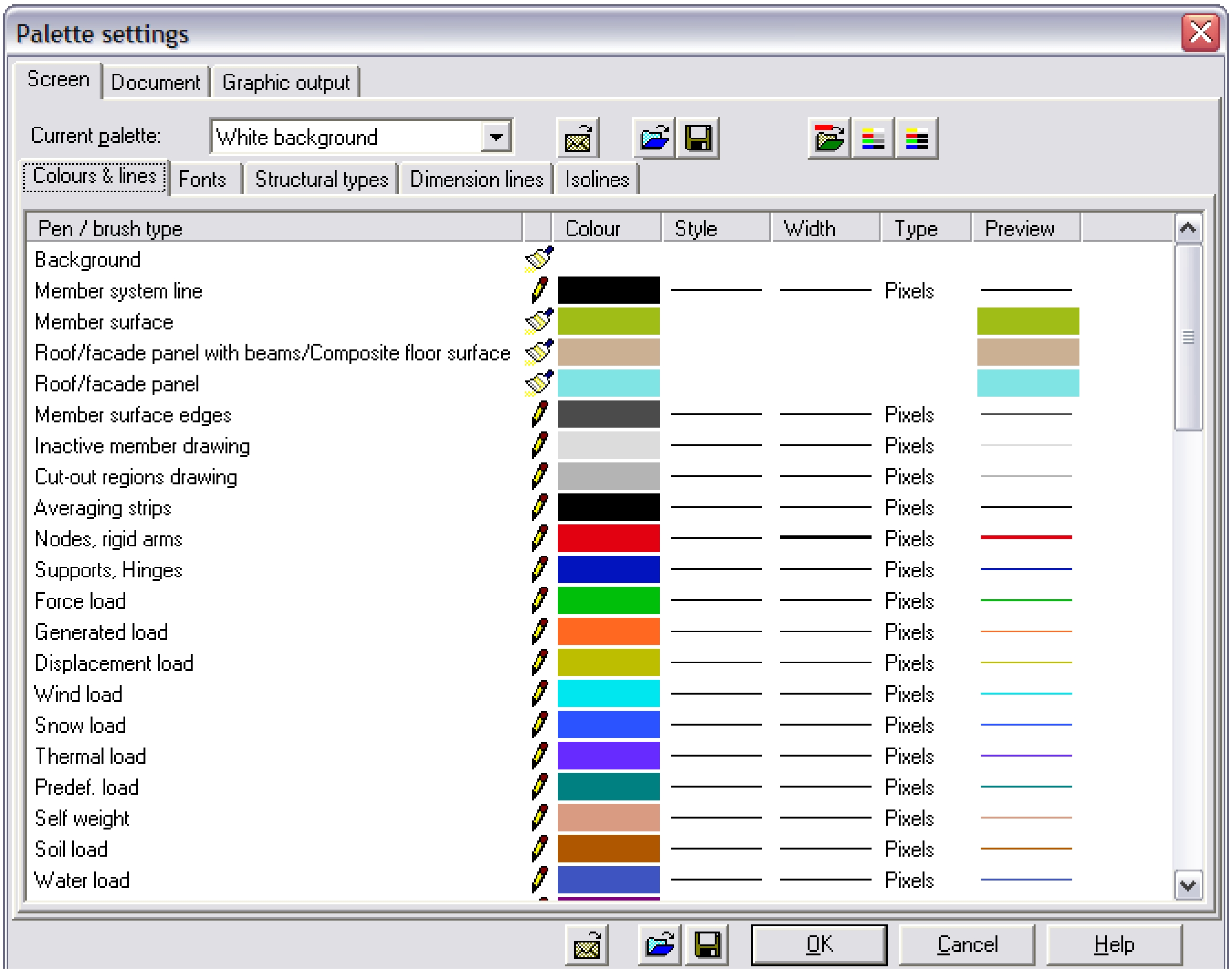
For a line load the drawing style will be a set of arrows. The property for drawing on begin/end will be the value of the line load. As illustrated on the picture, this property serves as a scale for the drawing style: the line load with value -5 will be drawn smaller compared to the line load with value -10.

For this example the **Drawing style** is set to ‘Box on Line’. The **Property for drawing on begin/end** is set to ‘Gamma M’. The scale of the drawing style will thus be dependent on the value of the safety factor.

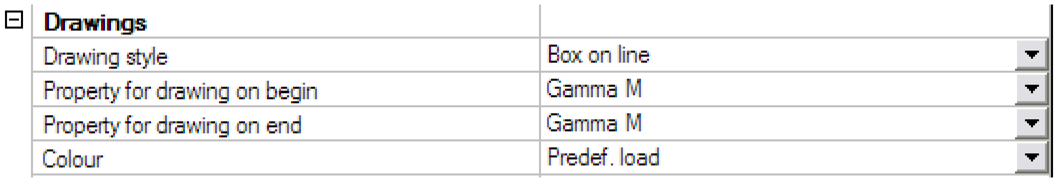
Only numerical parameters can be used as property for drawing on begin/end.

It is not required to define a property for drawing on begin/end.

The **Colour** field allows choosing the colour of the additional data. The items in this list correspond to the dialog of **Setup > Colours/Lines**:

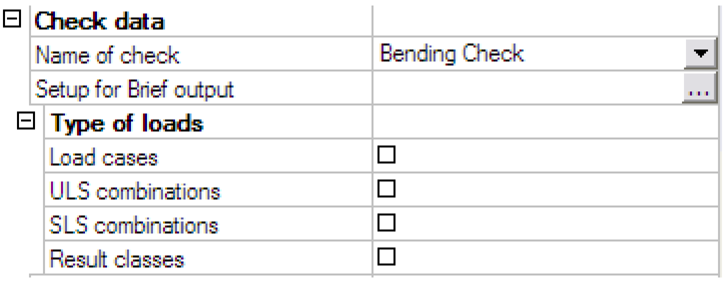


In this example, the colour is chosen as for a Predefined load



Step 2.7 Define the Check

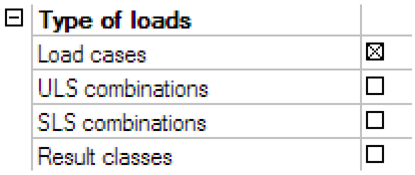
In the group **Check data** the necessary data for the check itself are defined.



As specified in *Step 2.2* the **Name of check** is taken automatically from the text string database. If required a different string can be chosen from the database using the combo-box.

The **Type of loads** group allows to specify which load types will be available for the check. Only the selected items will be available when executing the check.

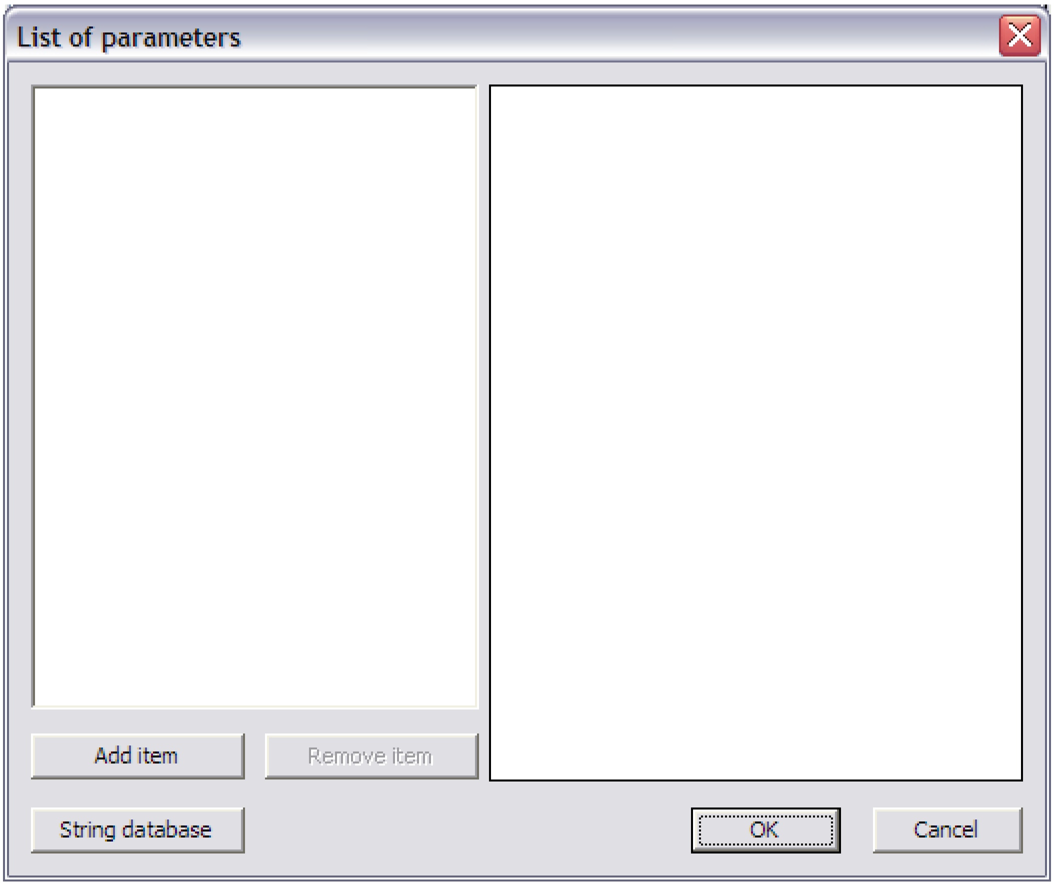
Since in this example only one load case was defined, only the option ‘Load cases’ will be activated.



At least one load type has to be activated.

Important remark: In case more than one load type has been activated, the check will be executed SIMULTANEOUSLY for all load types together! This implies for example that the check is done for both a load case and a combination at the same time. This allows the use of special checks: in the Excel file it can be set that a certain check can be done for the load case while a different check is done for the combination. In general, it is recommended to use only one load type.

The final item for defining the check is the **Setup for Brief output**. In this dialog the output parameters have to be defined i.e. the unity check values which will be read from the external application. The dialog has the same layout as the dialog for the input parameters specified in *Step 2.3*



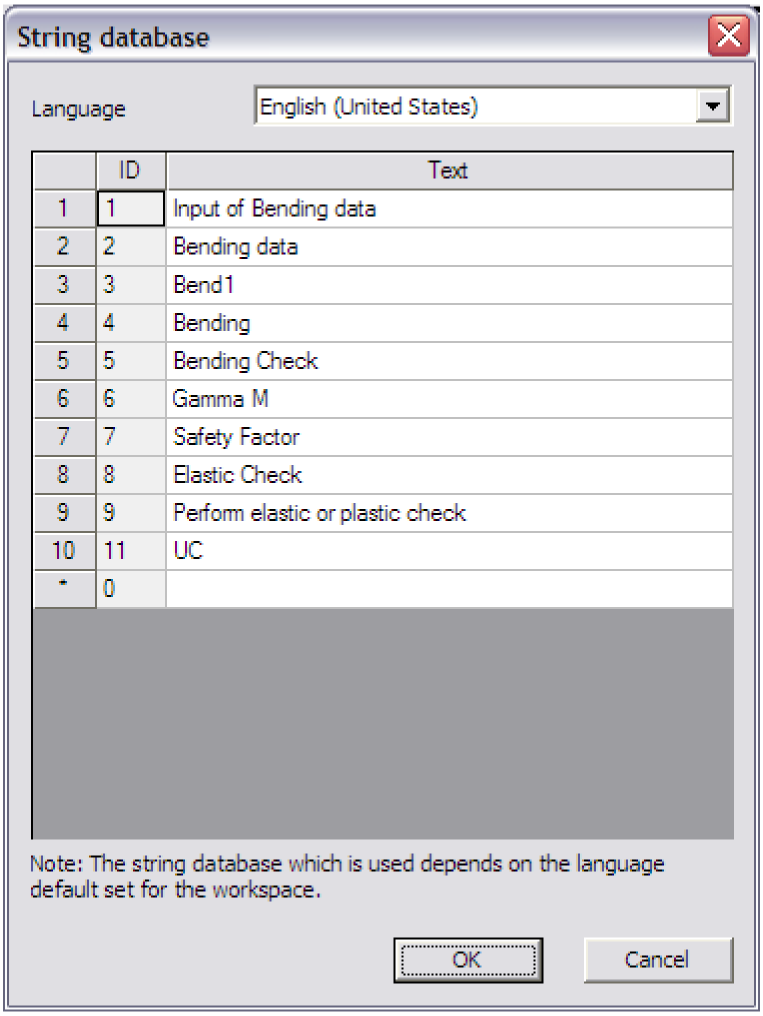
For this example, one parameter will be defined:

Parameter	Unit
Unity Check UC	-

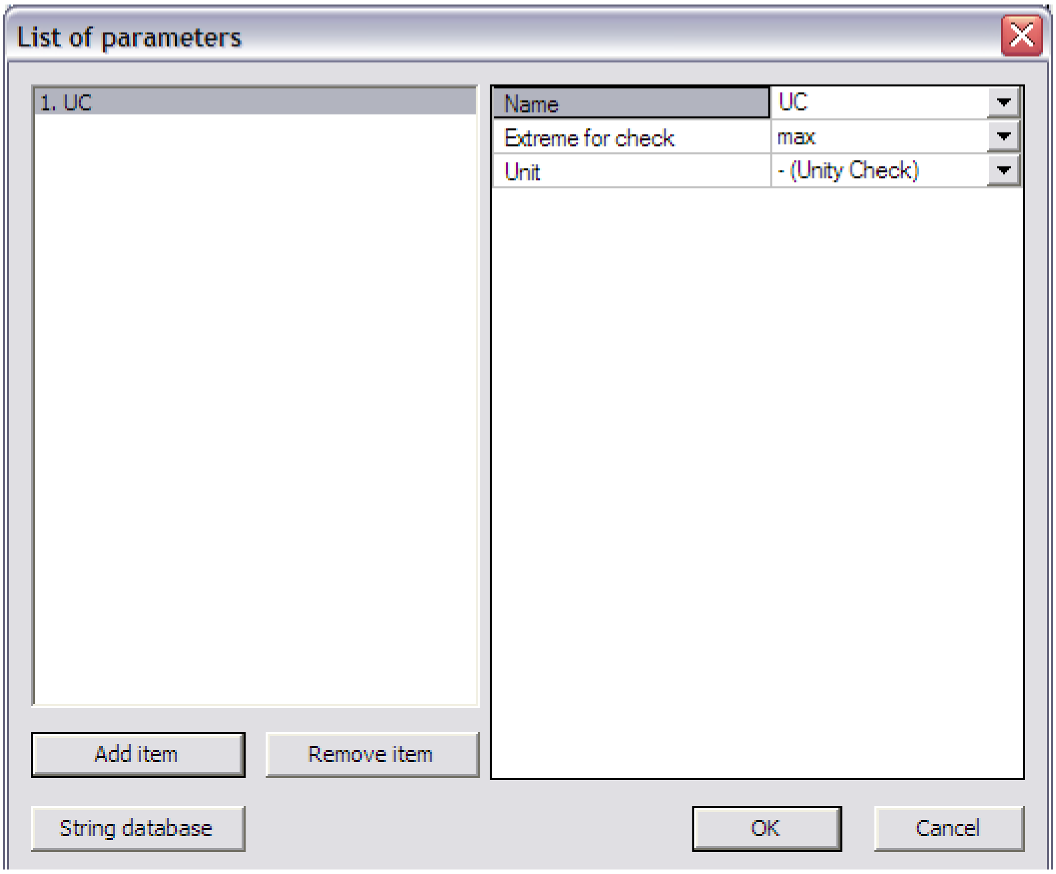
Through the button **String database** the text string database can be directly accessed. This allows a quick input of the strings required for the output parameters.

For this example the following string is added:

String used in this example
UC



Through the button **Add item** the unity check parameter is added.



In the **Name** field the respective string can be chosen from the string database, in this case ‘UC’.

The **Extreme for check** is used to specify if the extreme is a minimum or a maximum. When it concerns a unity check, in most cases the extreme is a maximum.

In this example the extreme is set to ‘max’ since the maximal bending check will be limiting.

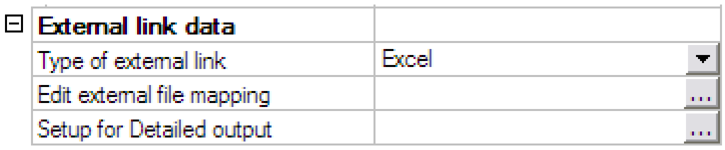
The **Unit** field can be used to specify a unit for the defined output parameter. In this case, since it concerns a unity check the default unit ‘- (Unity check)’ is used.

Using the button **Remove item** a parameter can be removed again if required.

At least one output parameter has to be defined, else it is not possible to execute a check.

Step 2.8 Specify the type of external link

In the **External link data** group the **Type of external link** allows to specify which external application will be used.

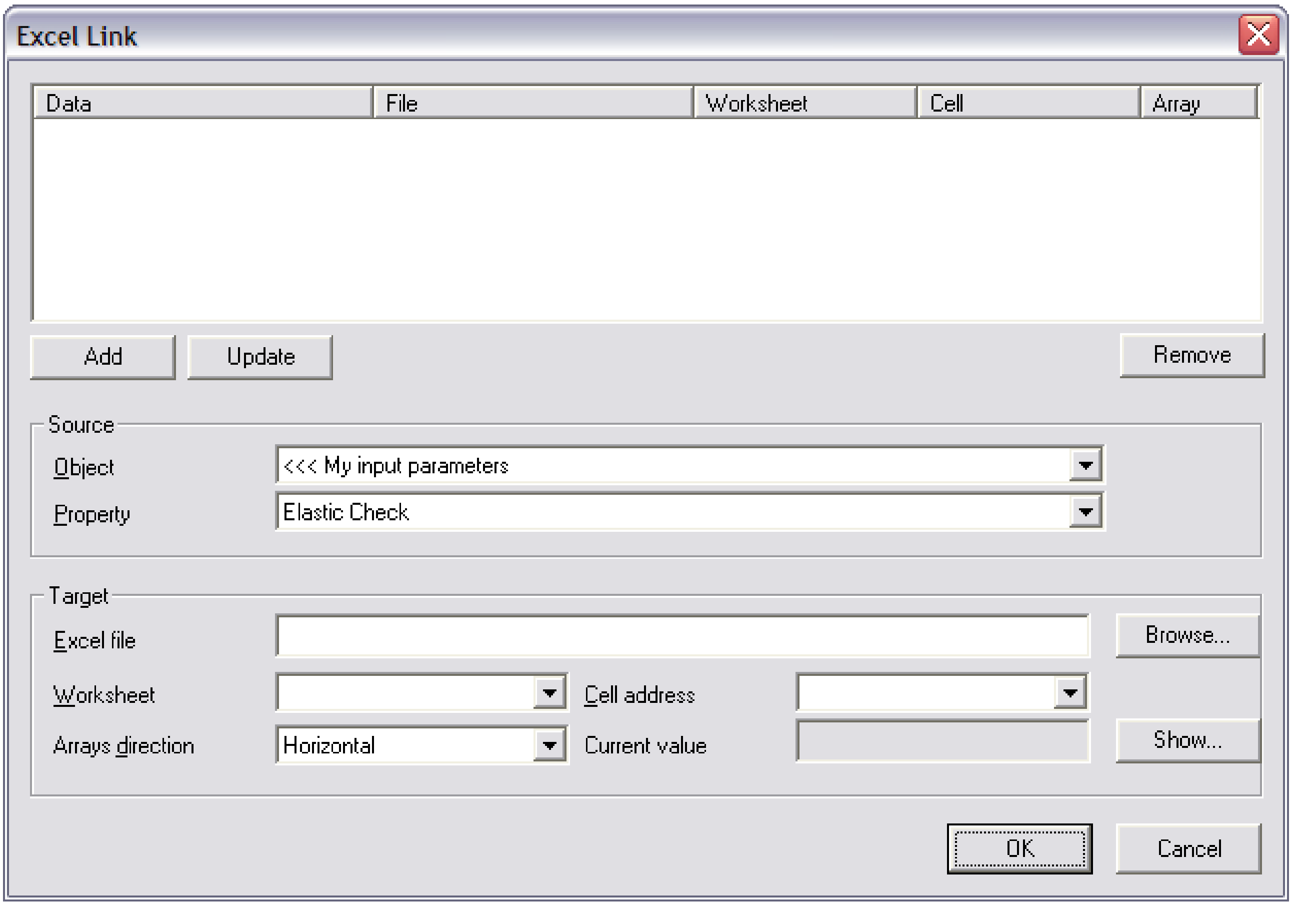


In this example the link is made with Excel and thus ‘Excel’ is chosen.

Step 2.9 Define the mapping with the external application

All preparation has now been done, what remains is the most important step of the process: defining the actual mapping between properties and parameters of Scia Engineer and the data fields (i.e. Excel cells) of the external application.

Through the button **Edit external file mapping** the mapping dialog is opened.

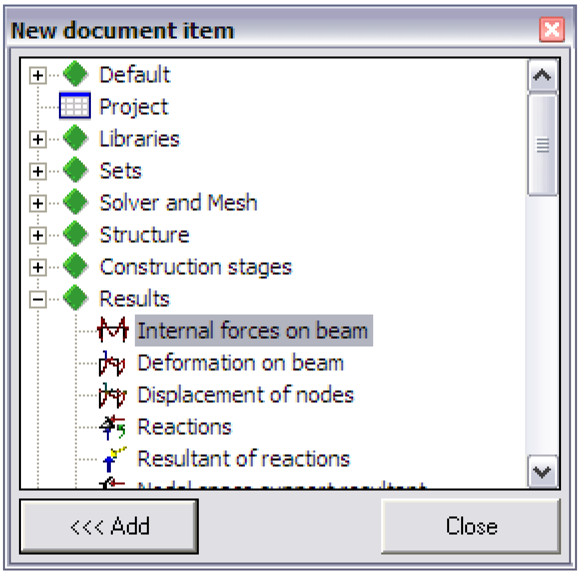


In the **Source** group, the **Object** field shows a list of all available objects in Scia Engineer as well as the parameters which have been defined in the previous steps. These objects can thus be classified into three different types:

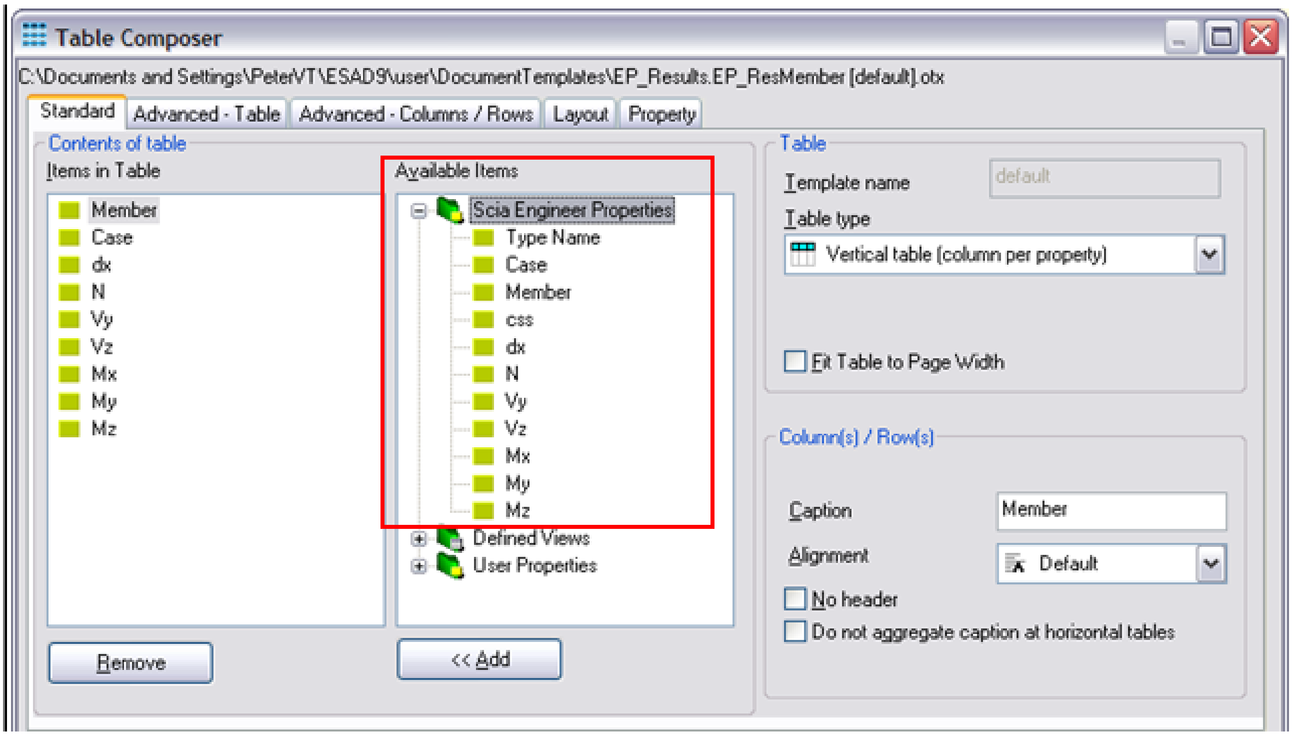
Object	Description
<<< My input parameters	This object contains the input parameters specified in step 2.3.
>>> My output parameters	This object contains the output parameters specified in step 2.7.
All others	These objects contain the existing Scia Engineer properties

In the **Property** field the actual property can be selected. For ease of reference, this list of objects and properties is taken the same as the 'Available Items' found in the table composer when adding properties to a document table.

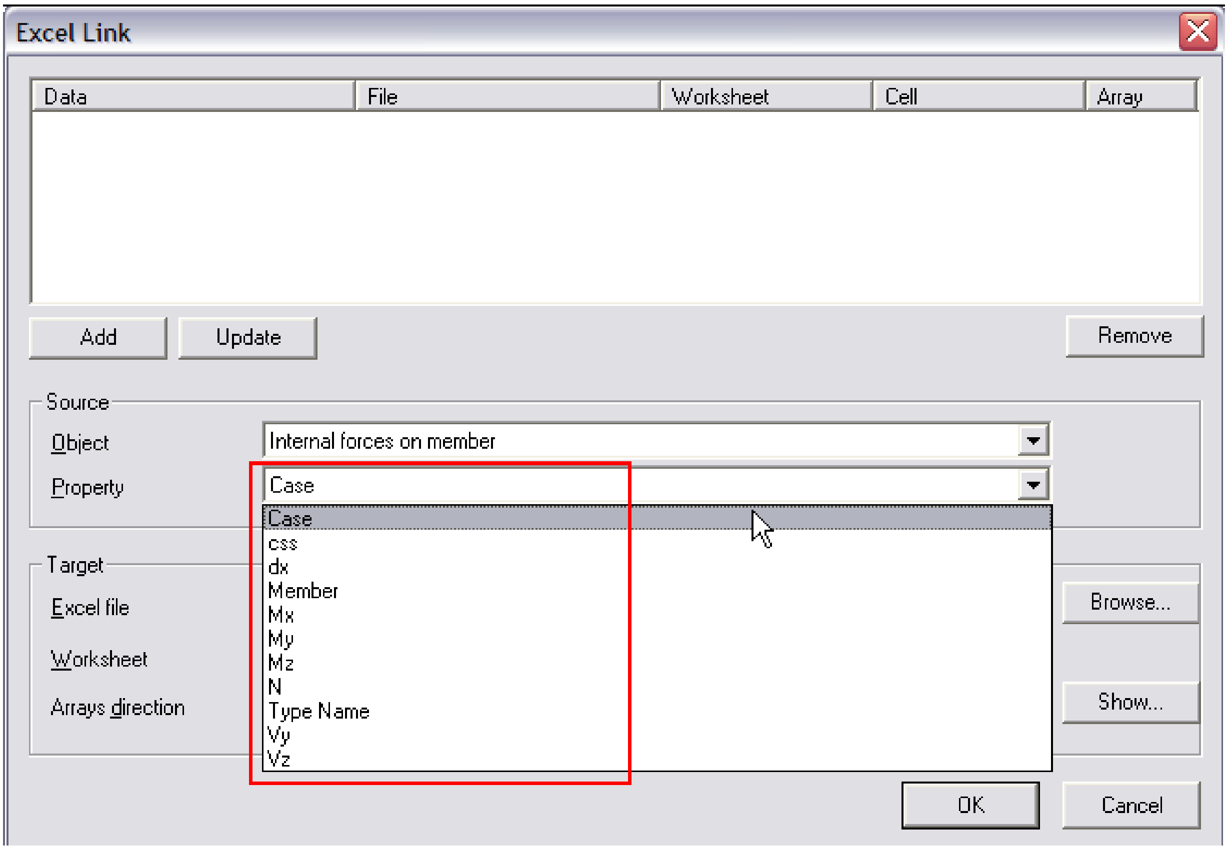
For example, in the document, the table for Internal forces on a member can be found under Results:



When viewing the contents of this table (otx file) in the table composer, the following screen is shown:



The right side table, 'Available items', shows all Scia Engineer properties related to internal forces on a member. It is this same table which is shown in the **Property** field of the mapping dialog in case **Object** is set to 'Internal forces on member':



The reference with the document properties is a very helpful tool: when looking for a specific Scia Engineer property, go through the document tables. These tables will indicate where the property can be found.

The **Target** group is used to specify to which Excel cell the property should be mapped. The group contains the following items:

Item in Target group	Description
Excel file	Using the **Browse** button an Excel file can be selected. This field will show the path and filename of the Excel file.
Worksheet	In this field a worksheet can be set. The list of worksheets is automatically read from the specified Excel file.
Cell address	In this field the cell address can be inputted in two ways: a) Manual input: the cell address can be typed i.e. A1 or B7 … b) Named cells: In case Named cells have been used in the Excel file, these names are shown here automatically.
Current value	When a cell address has been inputted, this field shows the current content of that cell. Using the **Show** button the Excel file can be directly opened to show the location of this cell.
Arrays direction	The array direction is used to specify in which direction an arrayed property should be mapped. For example, when mapping the value of a point load on a member, this value is sent into one cell. In case there are more point loads on the member, this array of values can be mapped horizontally or vertically from the specified cell.

In this example, the Excel file has the following layout:

	A	B	C
1	**Bending Check**		
2			
3	**Data from SCIA Engineer**		
4			
5	Moment My	100000	Nm
6			
7	Section modulus Wel	0,0015	m^3
8			
9	Section modulus Wpl	0,002	m^3
10			
11	Yield Strength fy	235000000	N/m^2
12			
13	Safety factor Gamma M	1,1	-
14			
15	Elastic Check ?	0	
16			
17	**Bending Resistance**		
18			
19	MRd	427272,73	Nm
20			
21	**Unity Check**		
22			
23	UC	0,23	-

The following table shows which properties should be mapped to which cells:

Object	Property	Cell Address
Internal forces on member	My	B5
Cross-Sections	Wely (Property)	B7
Cross-Sections	Wply (Property)	B9
Steel EC3	Yield strength (code independent)	B11
<<< My input parameters	Gamma M	B13
<<< My input parameters	Elastic Check	B15
>>> My output parameters	UC	B23

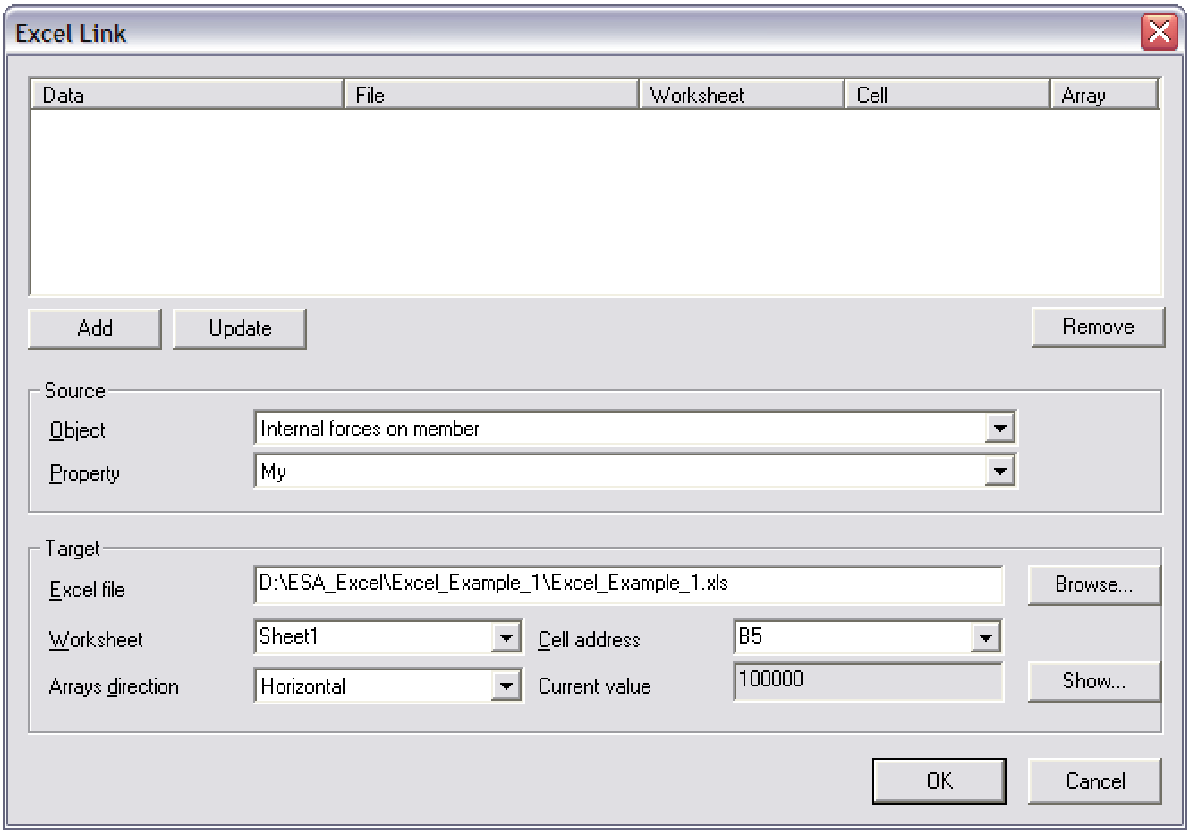
The mapping of the first property, the bending moment My is thus done as follows:

The **Object** field is set to 'Internal forces on member'.
In the **Property** field 'My' can then be chosen.

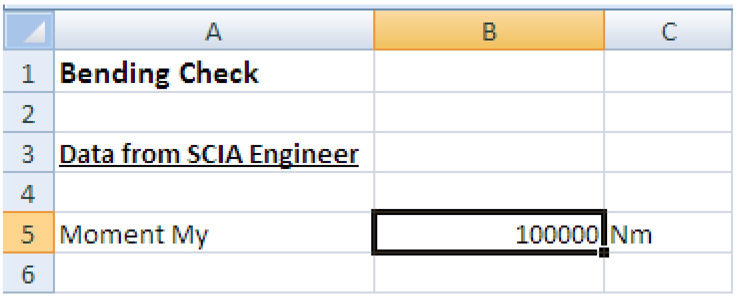
Using the **Browse** button, the file **Excel_Example_1.xls** is searched.

After the file has been specified, the **Worksheet** field contains a list of all sheets. This field is set to 'Sheet1'.

The **Arrays direction** is set to ‘Horizontal’. In this example no array properties are mapped so choosing ‘Horizontal’ or ‘Vertical’ would make no difference.

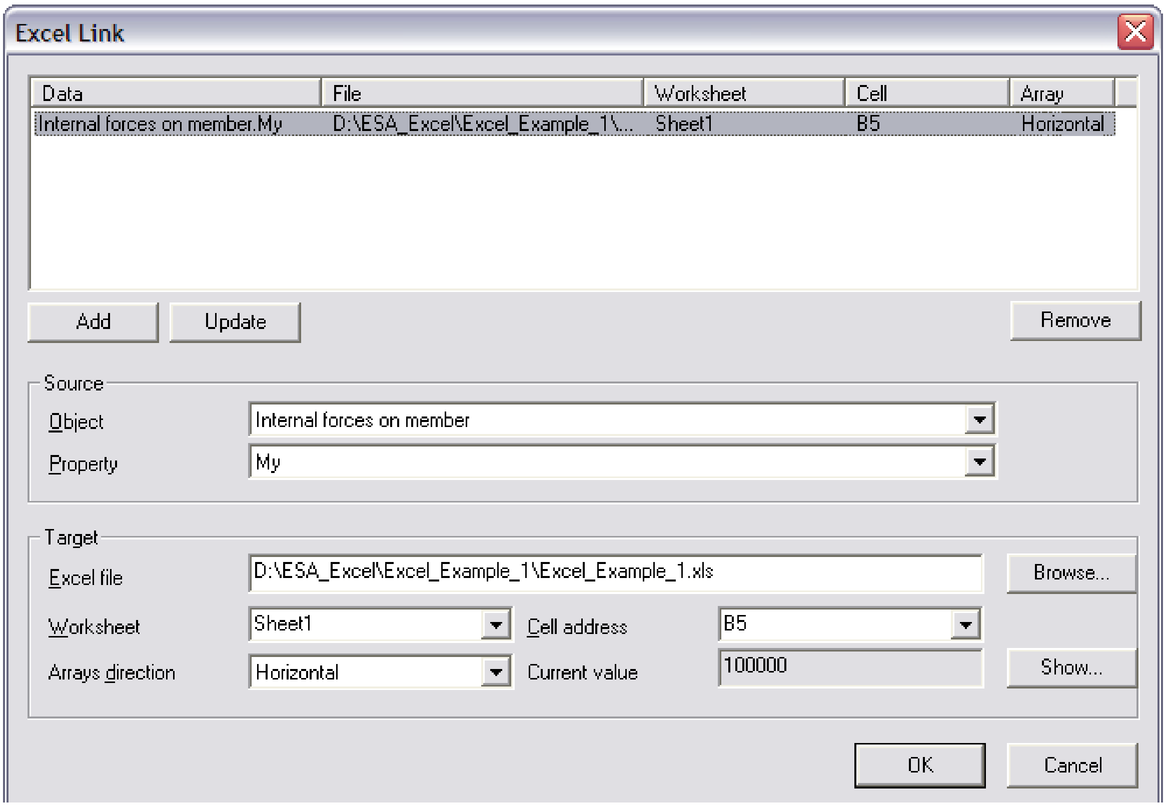


Finally, in the field **Cell address** the cell ‘B5’ is typed. Automatically the **Current value** field will show the current content of the cell, in this case 100000. Using the **Show** button, the Excel file is opened and the specified cell is highlighted. This provides an easy way to check if the correct cell has been set.



Important note: When clicking on the Show button a warning is displayed. The Excel window which is opened may NOT be closed. It should be hidden using the Hide button in the Excel link dialog.
In case the user by accident closes the Excel window instead of using the Hide button, the link with Excel will be lost! In this case, Scia Engineer has to be restarted in order to restore the link!

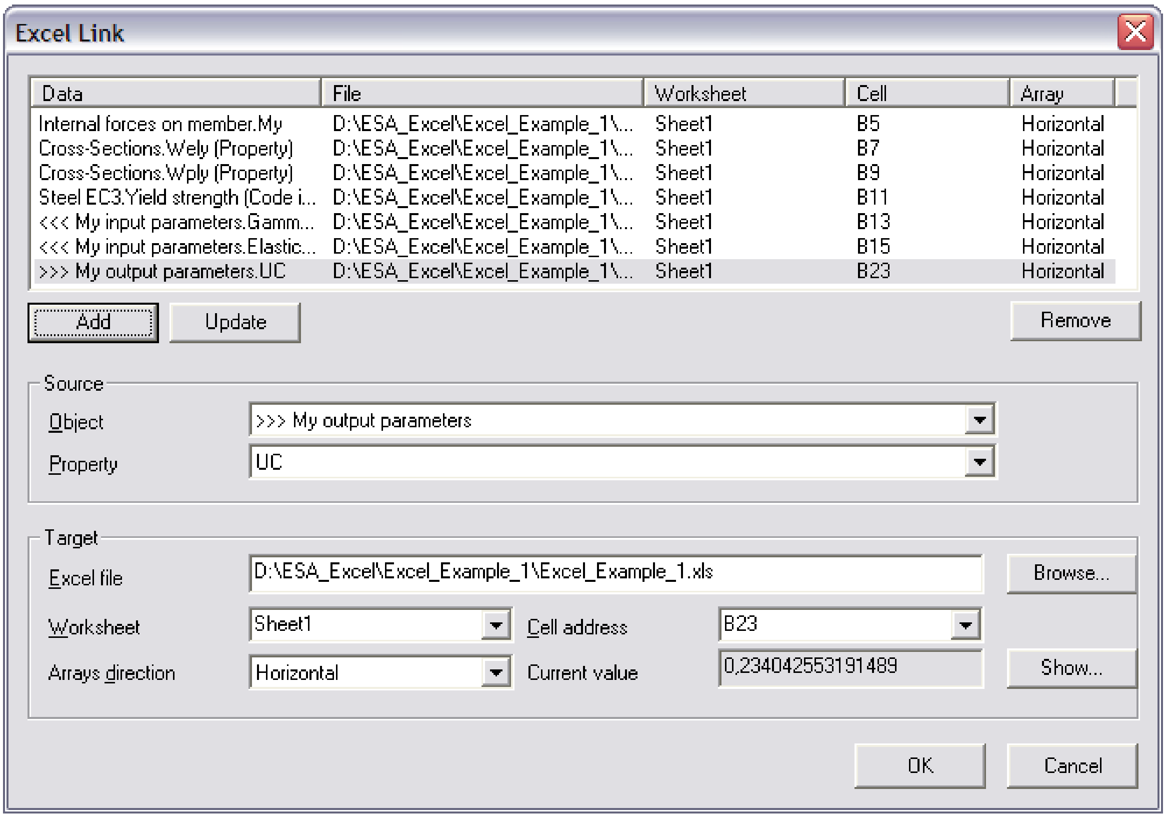
When all input has been done, this mapping is added to the table using the **Add** button.



To remove a mapped parameter from the table, select the desired line in the table and press the button **Remove**.

When selecting a line in the table, the properties are shown at the lower part of the dialog. This way, the settings for the mapping can easily be modified. Using the **Update** button the selected line is updated with the modifications.

In the same way as described above, the other parameters can now be mapped to obtain the following mapping:



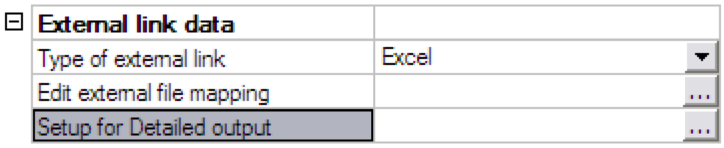
During the check the Scia Engineer and input parameters are sent to Excel while the output parameters are read from Excel. Since in this example the additional data is of type ‘Line on 1D member’, this procedure will be repeated for each section along the member.

> ***The export of the Scia Engineer and input parameters follows the same logic as XML export. This implies that these properties are sent to Excel in basic SI-units! If required, a unit conversion should be accounted for in the Excel file.***

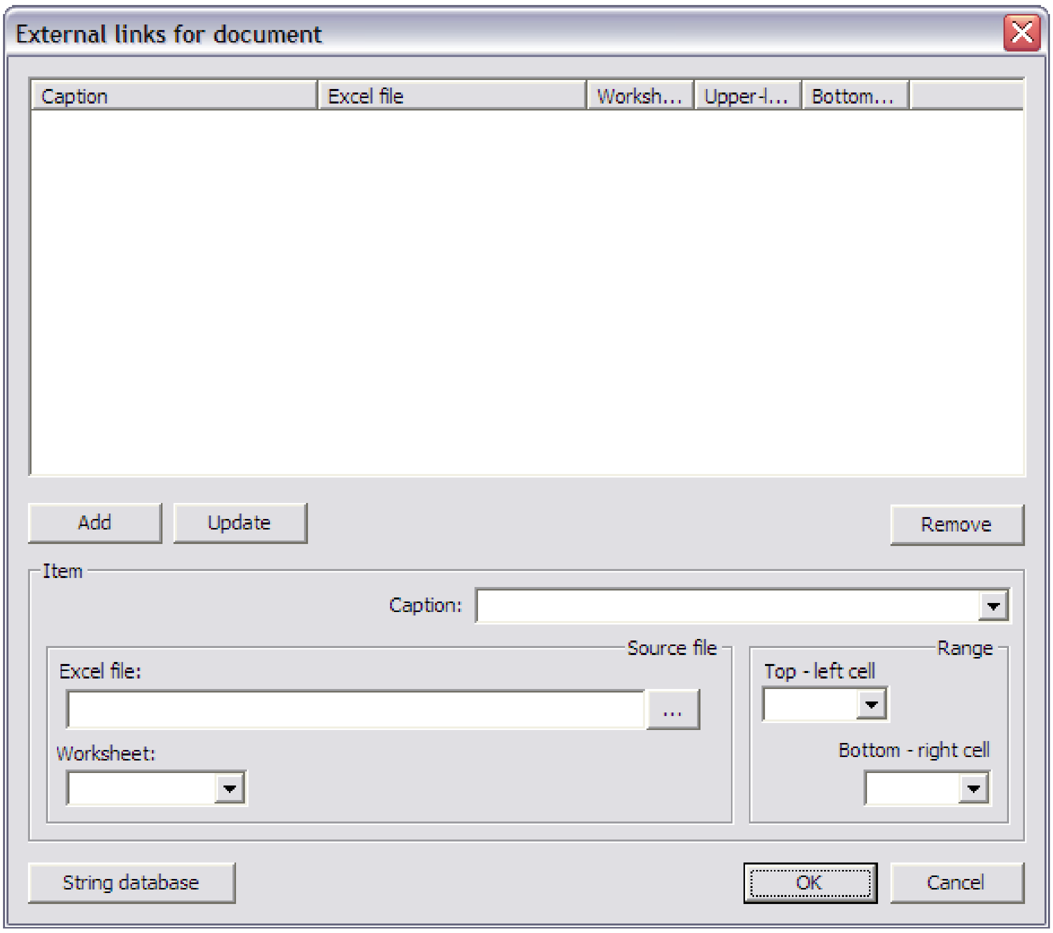
> ***At least one output parameter has to be mapped since else the check cannot be executed.***

Step 2.10 Define the Detailed output

In *Step 2.7* the parameters for the Brief output have been defined. These parameters will be used for the check. In addition, a Detailed output can also be specified to show an in-depth overview of the check.



By clicking on **Setup for Detailed output**, the Detailed output dialog is opened.



A Detailed output concerns an exact snapshot from the Excel file after the mapping has been sent. This way, the full details, check results, intermediate values… from Excel can be shown within Scia Engineer.

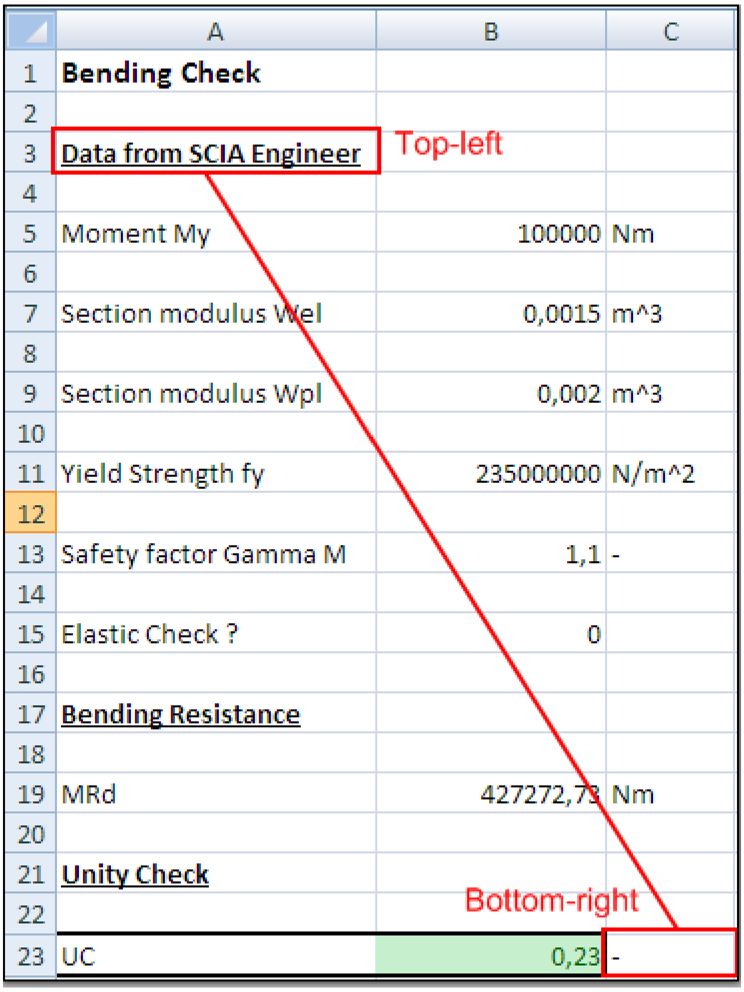
> ***The unity checks shown on screen will always be the output parameters specified for the Brief output. The Detailed output concerns only a snapshot of the Excel file, not an actual unity check value.***

In the same way as explained in *Step 2.9* the **Excel file** and **Worksheet** can be specified.

In the **Range** group, the range can be defined which defines the area for the snapshot. An Excel range is defined by a **Top – left cell** and a **Bottom – right cell**. As specified in *Step 2.9* here also Named cells can be used in case they have been defined in the Excel file.

The **Caption** field allows specifying the header of the table. This field is directly linked to the text string database. Through the button **String database** the text string database can be directly accessed in case a new text string is needed for the caption.

In this example, the range will be defined from the cell A3 to the cell C23.

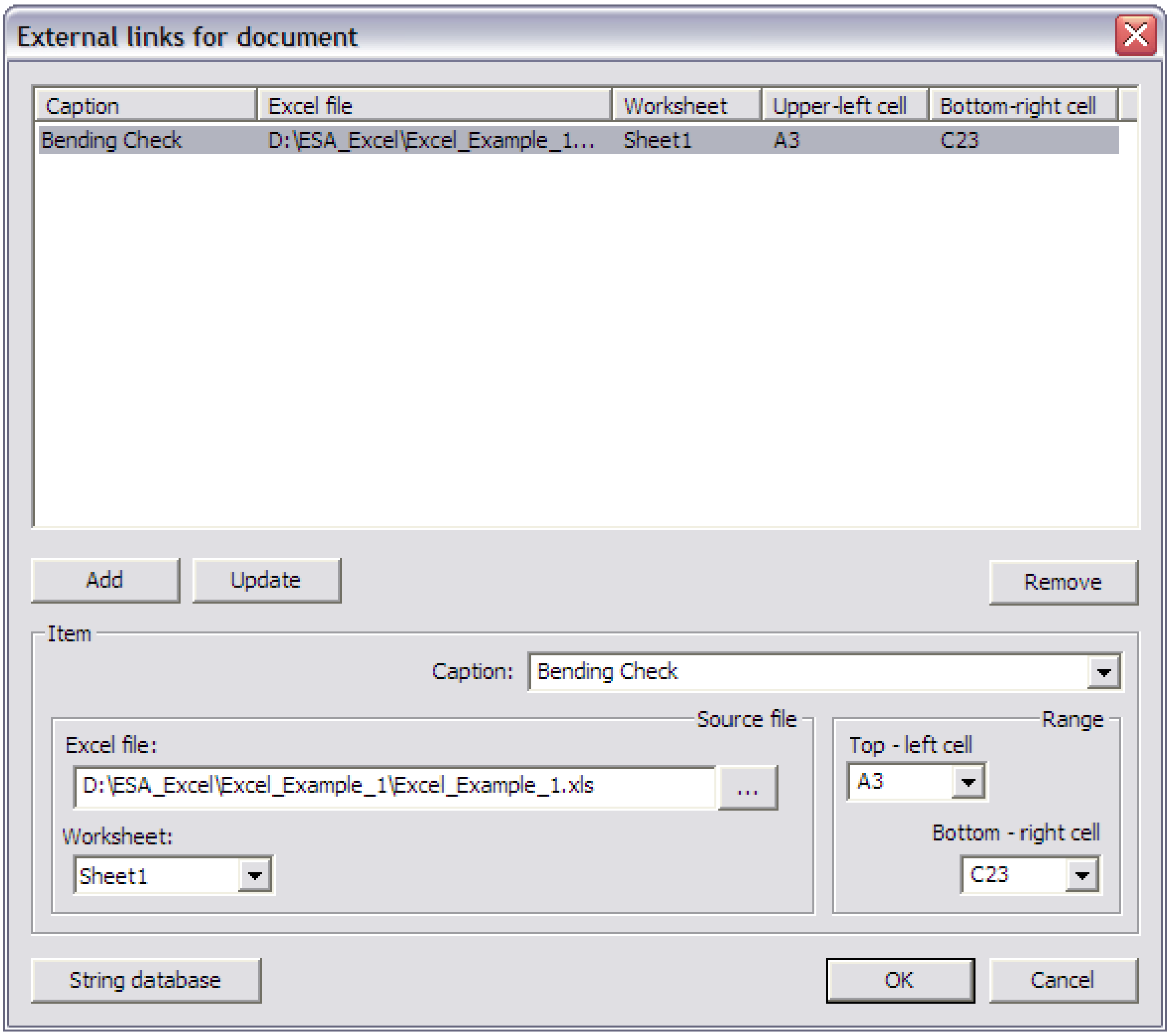


In the **Caption** field the string 'Bending Check' is chosen.

In the **Excel file** field the file **Excel_Example_1.xls** is searched using the browse button.

The **Worksheet** field is set to ‘Sheet1’.
In the **Range** group the **Top - left** cell is set as ‘A3’ and the **Bottom - right** cell as ‘C23’.

When all input has been done, the data is added to the table using the **Add** button.



To remove a Detailed output range from the table, select the desired line in the table and press the button **Remove**.

When selecting a line in the table, the properties are shown at the lower part of the dialog. This way, the settings for the Detailed output can easily be modified. Using the **Update** button the selected line is updated with the modifications.

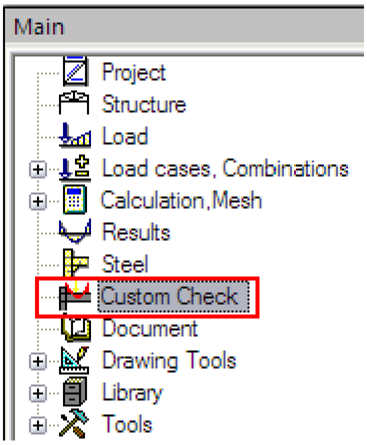
> ***More than one output range can be inputted with different captions, for example Bending Check, Shear Check, …***

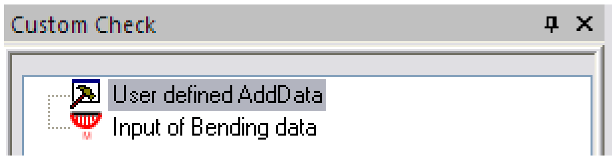
> ***It is not required to define a Detailed output. When no range has been defined, the Detailed output will be empty.***

With this final step, the User Defined Additional Data has been fully inputted and the **User Defined Additional Data** Library can be closed.

Step 3: Input the User Defined Additional Data on members/nodes

After closing the **User Defined Additional Data** Library a new service will be shown in the Scia Engineer tree: **Custom Check**.

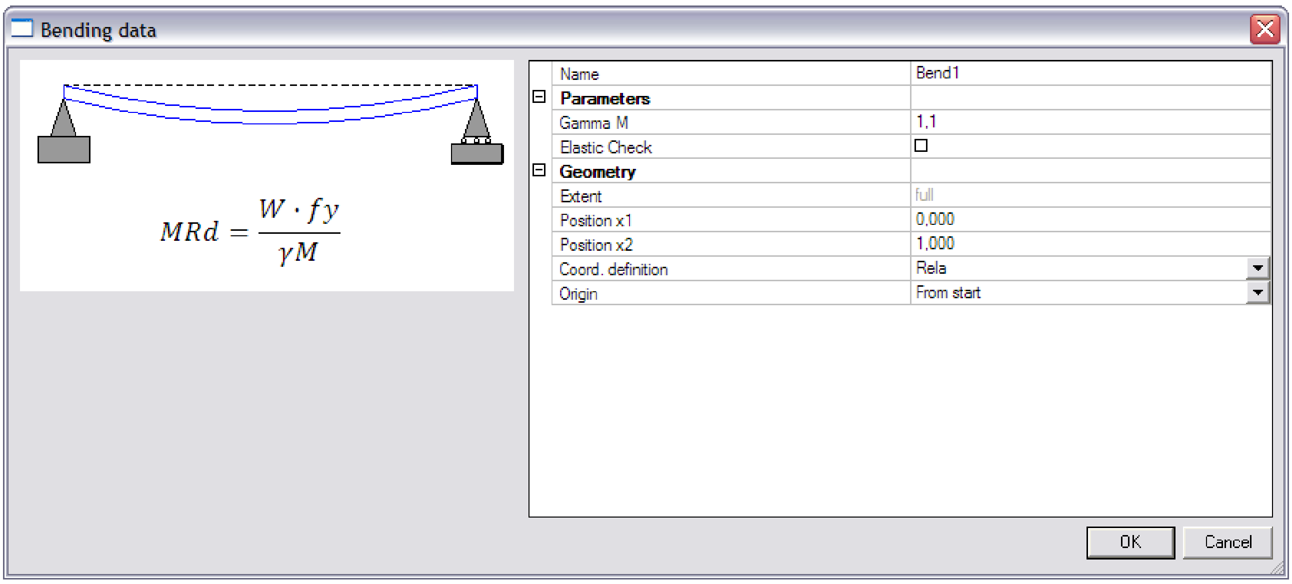




Currently, this service holds two objects: **User defined AddData** provides a direct link to the **User Defined Additional Data** Library. This way, the library can be easily accessed in case modifications are needed.

Second, the additional data which was defined in *Step 2*, **Input of Bending data** is shown. As can be seen, the icon and Service name defined in *Step 2.5* are shown.

This user data can now be inputted on the member. When double clicking on **Input of Bending data** the dialog with the properties of the data is displayed:



The title of the dialog shows the Type name specified in *Step 2.6*.

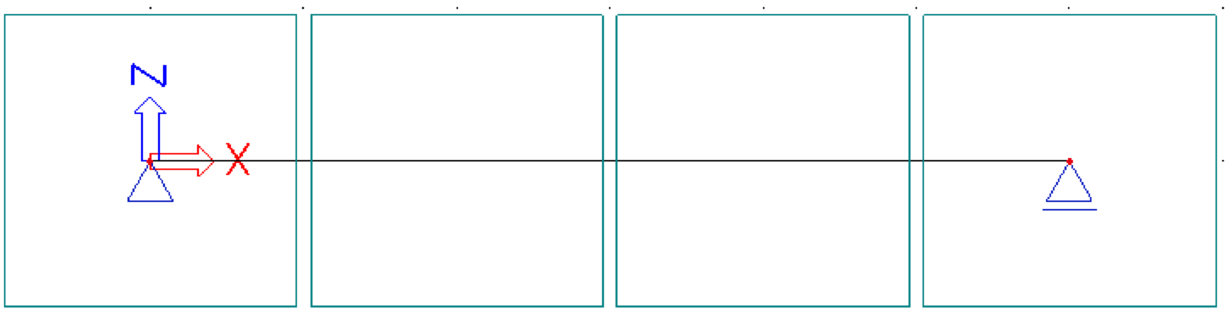
At the left side, the picture specified in *Step 2.4* is shown.

The Name field shows the short name as defined in *Step 2.6*.

The **Parameters** group holds the user defined parameters of *Step 2.3* with their default values. In this case the factor 'Gamma M' and the check-box 'Elastic Check'.

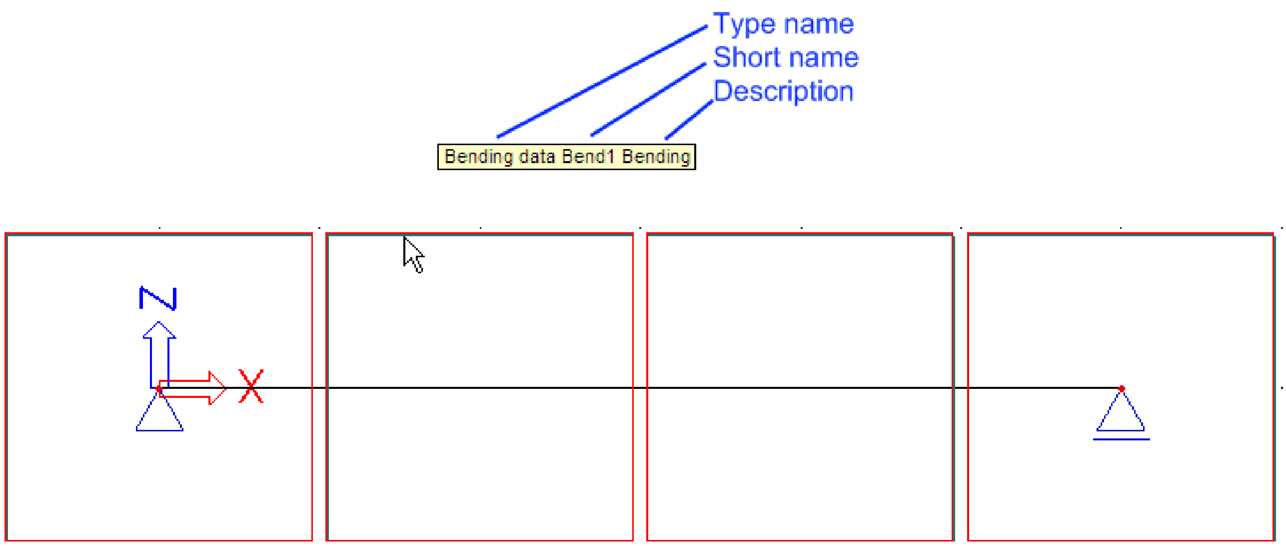
The **Geometry** group shows the default geometry options related to line additional data as specified in *Step 2.6*.

The default values are confirmed with [**OK**] and the data is inputted on member B1.

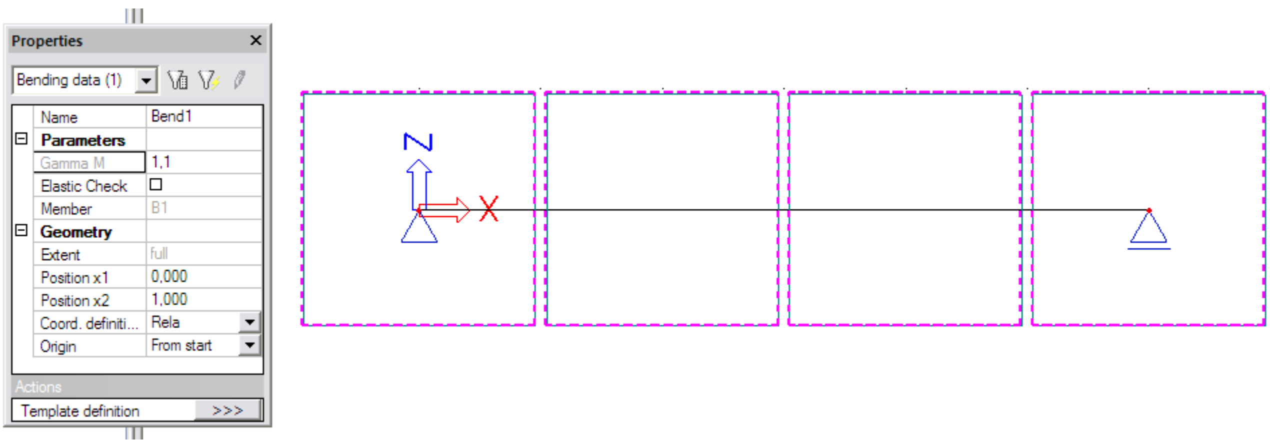


The colour of the additional data corresponds to the colour of a predefined load as specified in *Step 2.6*.

When moving the mouse pointer over the data, the tooltip shows the Type name, Short name and Description as specified in *Steps 2.2* and *2.6*.

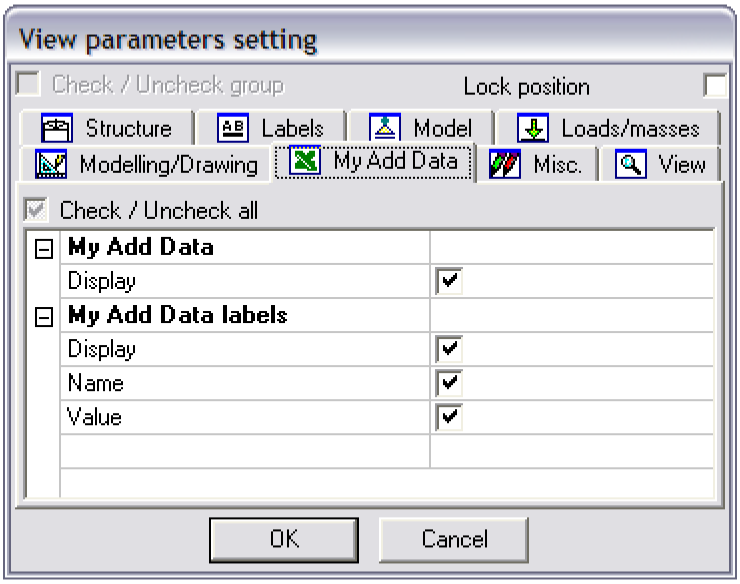


When selecting the additional data, the properties are shown in the property window:

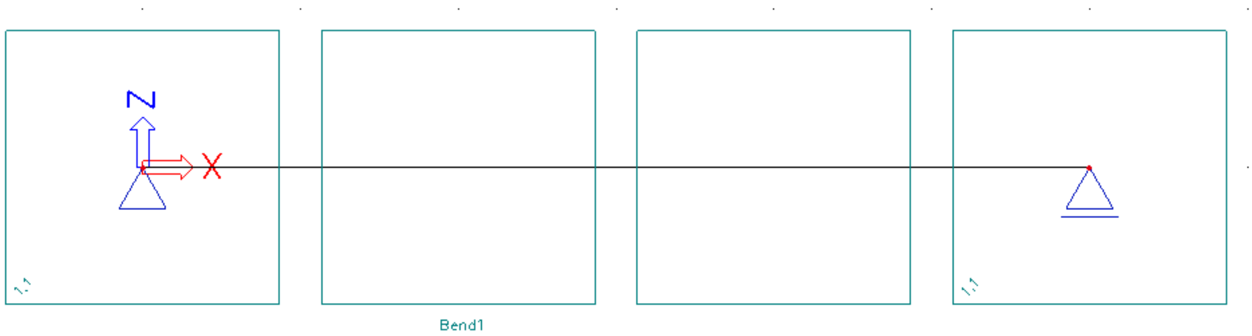


The action button 'Template definition' can be used to get a direct access to the definition of the additional data. This allows for quick modifications without the necessity to access the Library.

The Scia Engineer **View parameter setting** now holds an extra tab **My Add Data**.



In the same way as for other additional data, the labels can be displayed to show the Name and the Value of the additional data. After activating these options the following is displayed:



The label shows the short name of the additional data and at both ends the value of the safety factor Gamma M is displayed since this factor was set as 'Property for drawing on begin/end' as specified in *Step 2.6*.

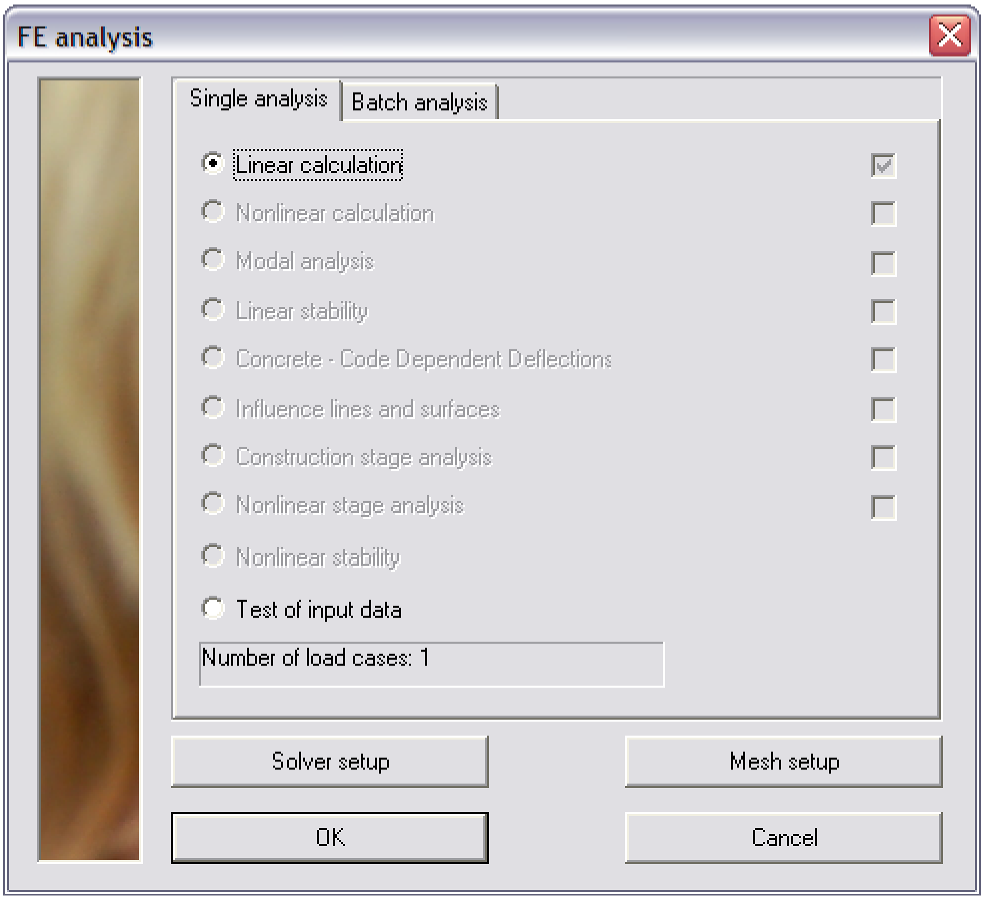
The data has now been inputted and in the next step the check can be executed.

The user defined additional data has to be inputted on members/nodes to indicate on which members/nodes the custom check should be (or can be) executed.

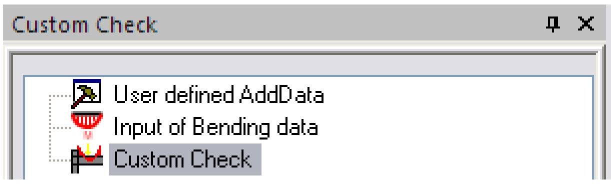
Step 4: Execute the Custom Check

In *Step 2* the additional data has been defined including the definition of the check, the mapping to Excel… In *Step 3* the additional data has been inputted. What is left is the execution of the check.

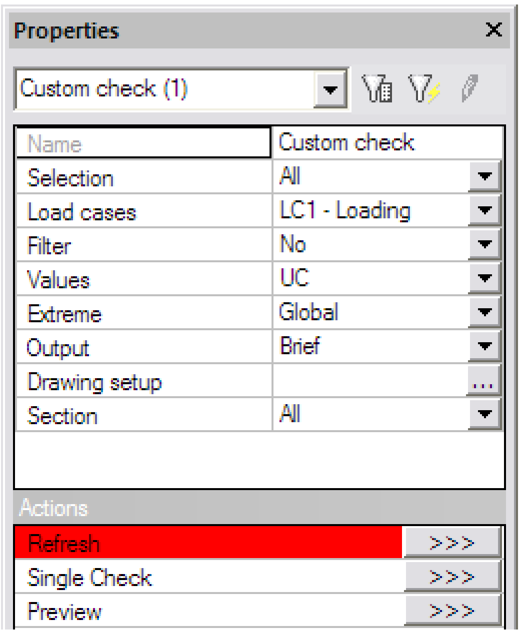
Since in this example the bending moment will be sent to Excel, it is required to launch a linear analysis.



When user defined additional data was inputted and the analysis has been executed, the **Custom Check** service will show a new item: **Custom Check**.



The property window of this check has the same layout as the property window of other standard checks in Scia Engineer (Steel, Timber, Aluminium, Concrete …)

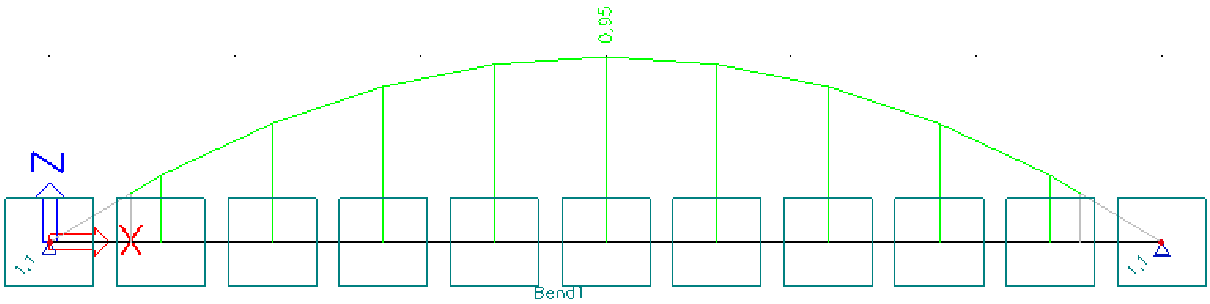


Since many types of User Defined Additional Data can be defined, the **Selection** field is now depended on a selection of additional data (and not on a selection of members/nodes).

In this example the field **Load cases** is shown since, in *Step 2.7*, only the load type Load cases has been activated.

The **Values** field contains the output parameters specified for the Brief output in *Step 2.7*. In this example the parameter ‘UC’ was defined and thus this parameter is shown.

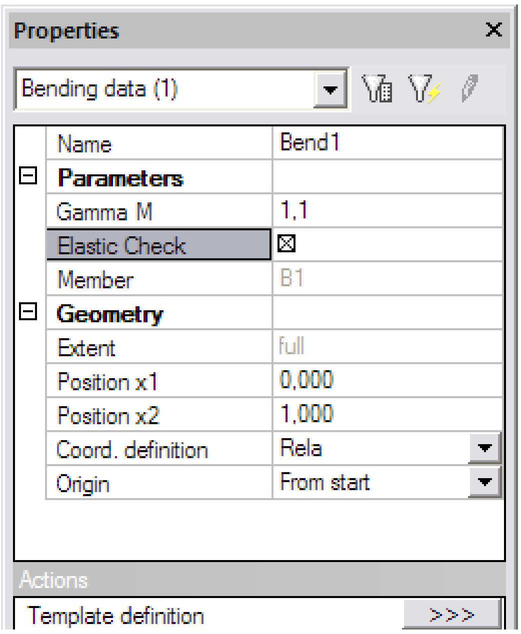
The **Refresh** action button is pressed to execute the check. The following check result is shown on screen:



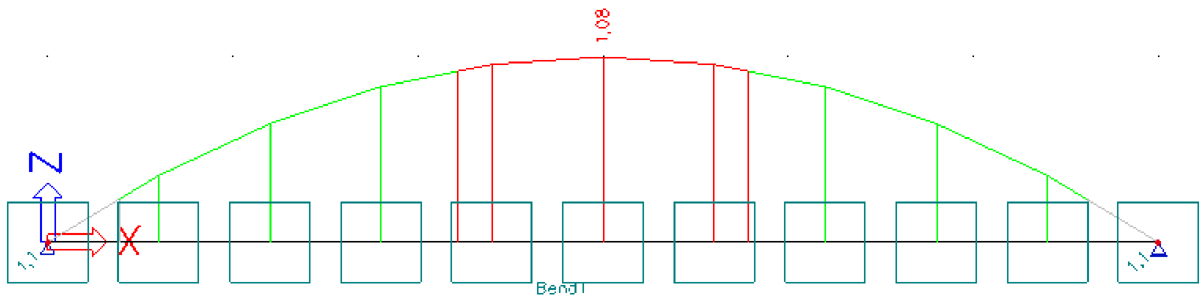
As can be seen, the shape of the unity is as expected, it follows the bending moment diagram My.

> ***Note: The drawing style of the additional data can be scaled using the default scale settings of Scia Engineer.***

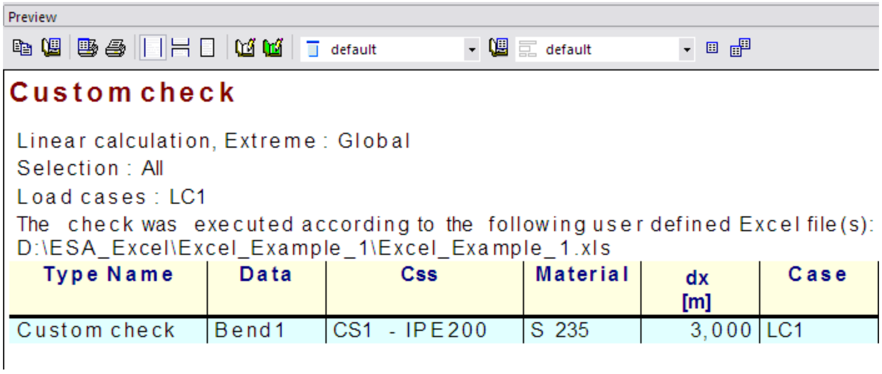
Next the additional data is selected and the ‘Elastic Check’ option is activated.



After pressing the **Refresh** action button again, the check results are updated:



The **Output** field is set to 'Brief'. Using the **Preview** action button the preview for the Brief output can be shown.

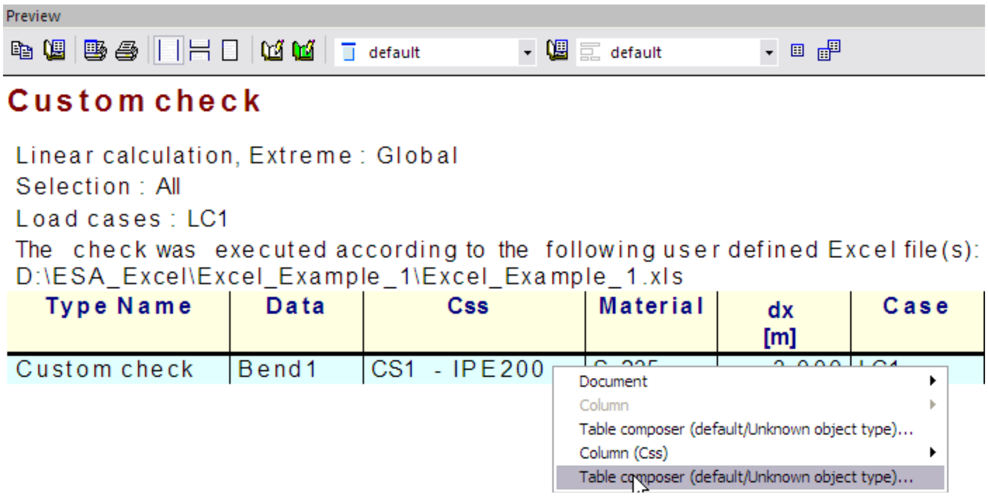


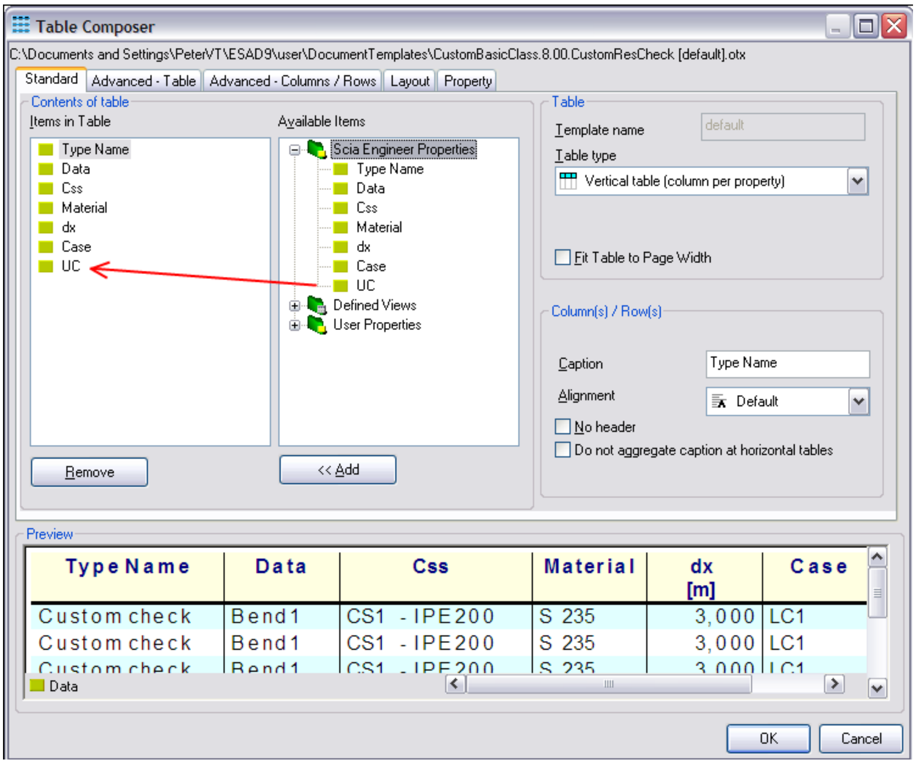
Custom check

Linear calculation, Extreme : Global
Selection : All
Load cases : LC1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_1\Excel_Example_1.xls

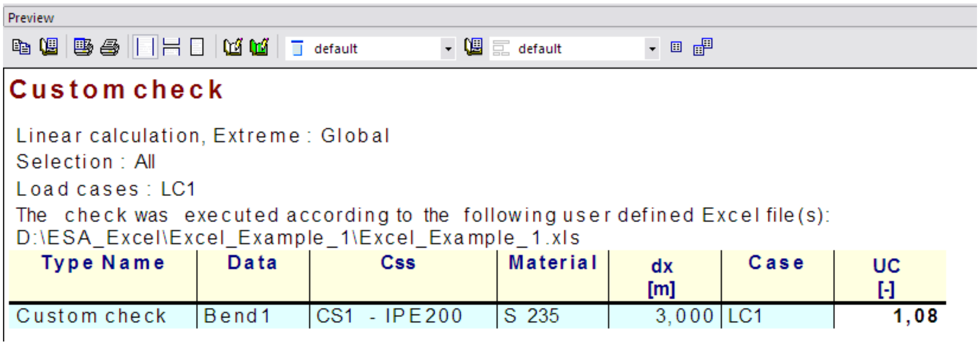
Type Name	Data	Css	Material	dx [m]	Case
Custom check	Bend1	CS1 - IPE200	S 235	3,000	LC1

As can be seen, the unity check value is not listed in the output table. This is because this new, user defined output parameter is not yet in the default otx for the Custom Check. Using default Table composer manipulation this value can be easily added to the output:



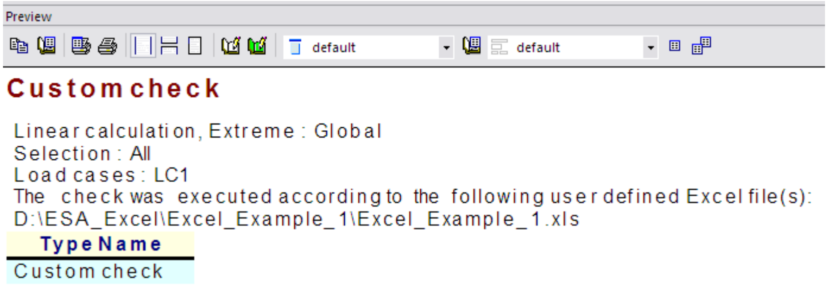


Now the **UC** value is correctly displayed:

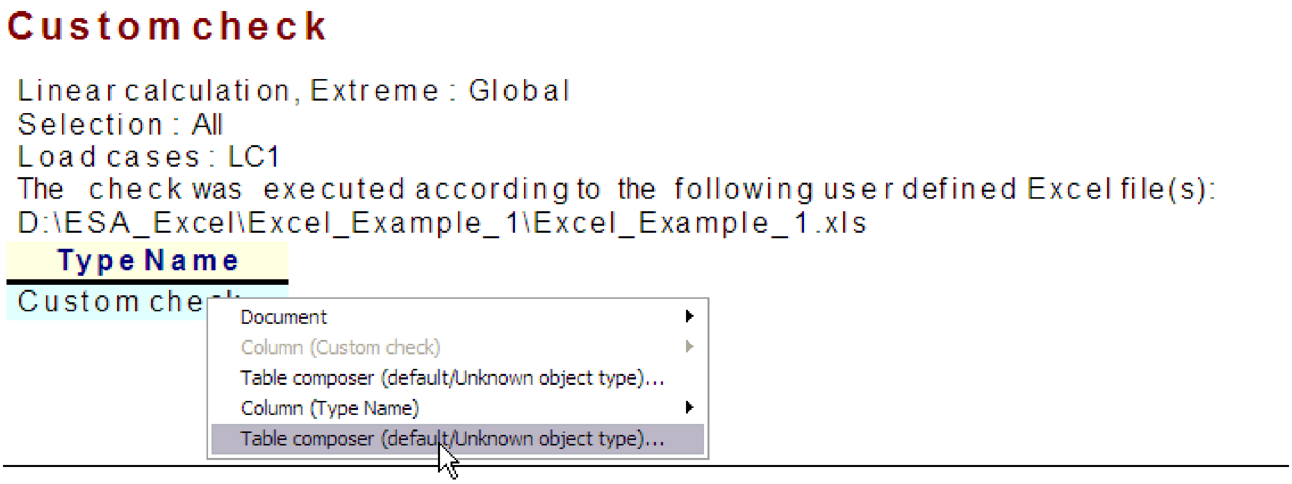


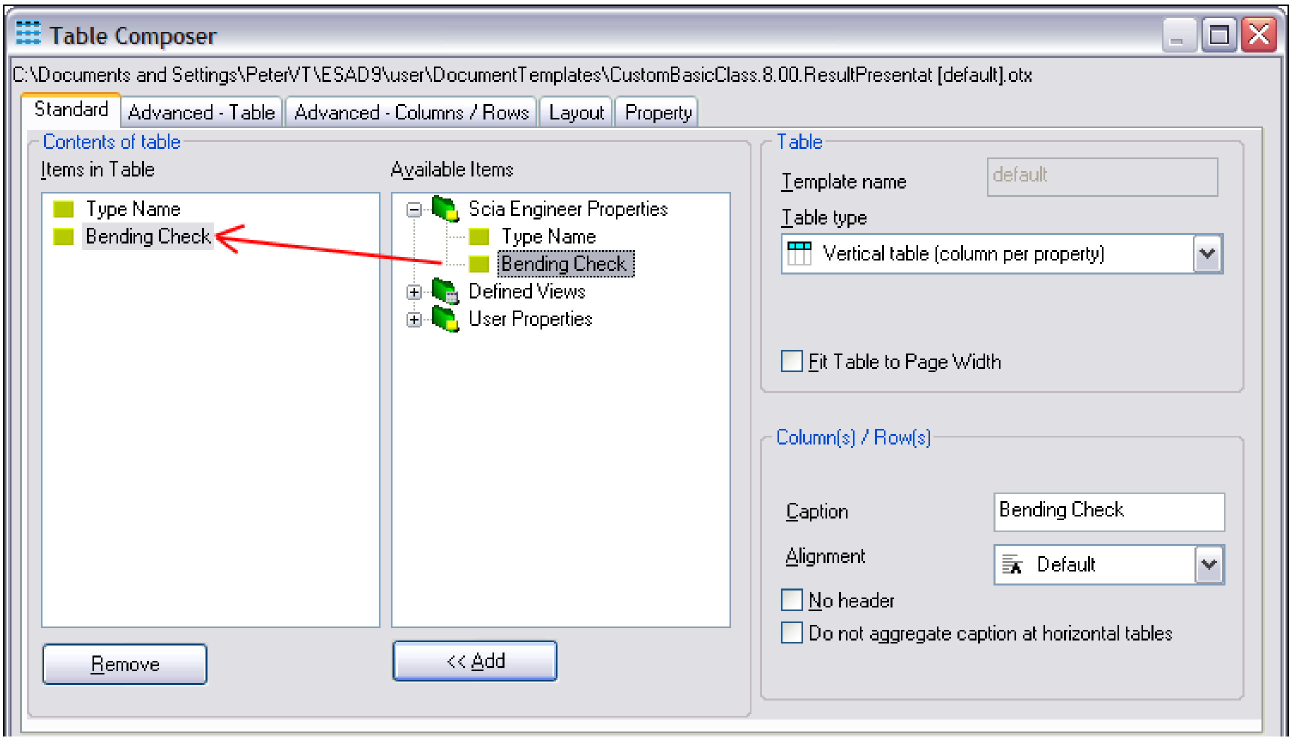
Next the Detailed output is examined. The **Output** field is set to ‘Detailed’ and the **Refresh** action button is pressed.

Here also, the first time the Detailed output is shown, this new, user defined output is not yet added to the default otx for the custom check.



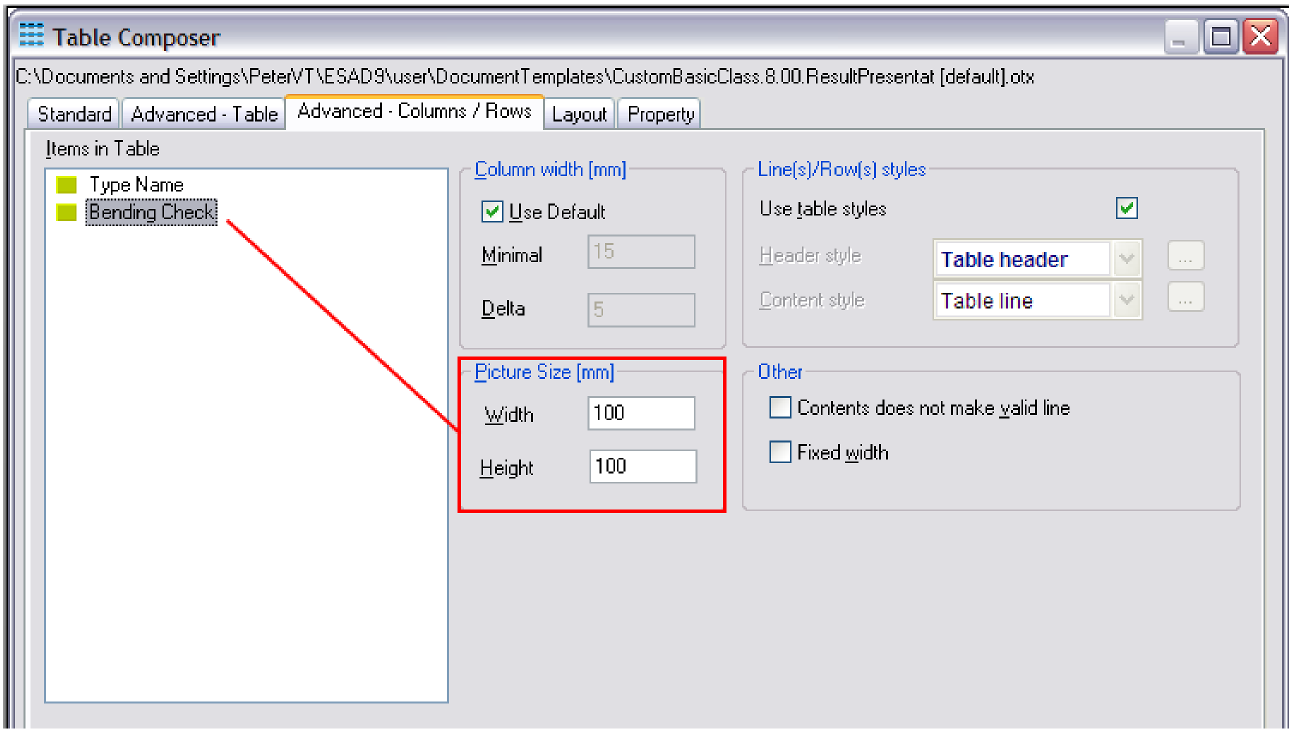
Again using default Table composer manipulation this value can be easily added to the output:



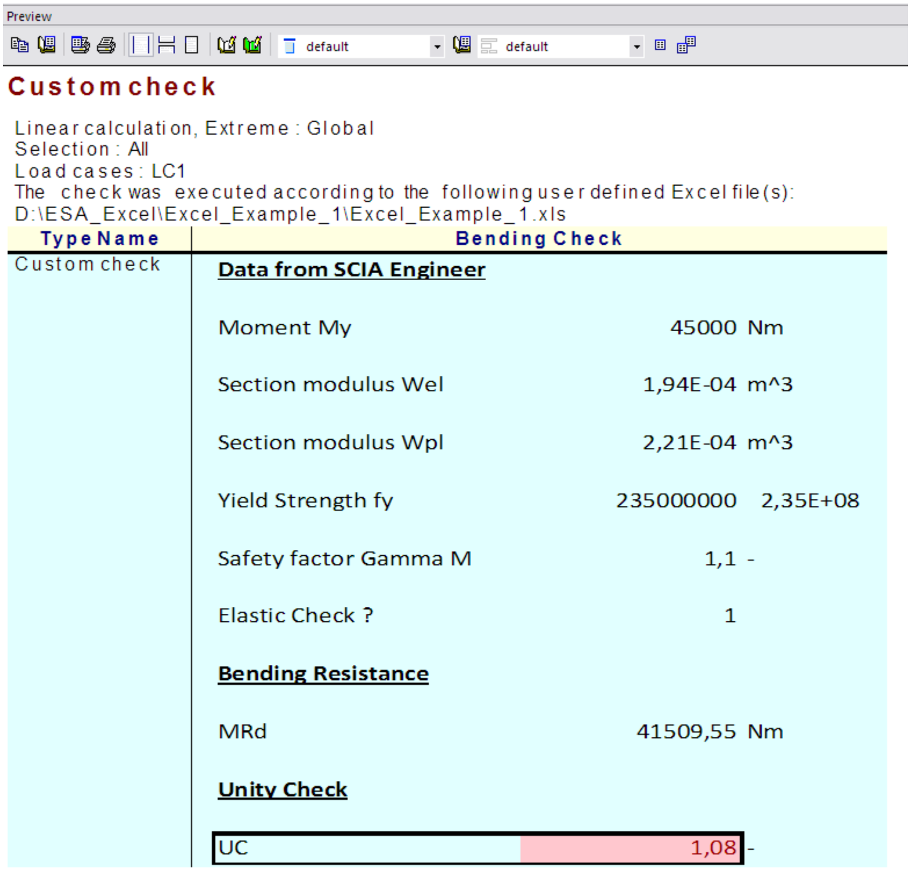


As can be seen, the name of the item in the table composer corresponds to the Caption name specified in *Step 2.10*.

In the Table composer, the size of the Bending Check item is also set to 100mm by 100mm



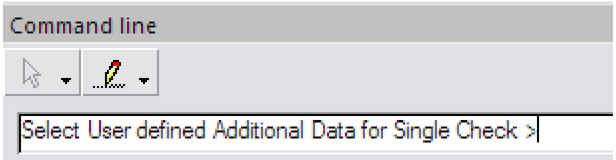
The Detailed output now shows the snapshot of the Excel file as defined in *Step 2.10*.



As can be seen, layout and colours from Excel are shown also in the Scia Engineer output.

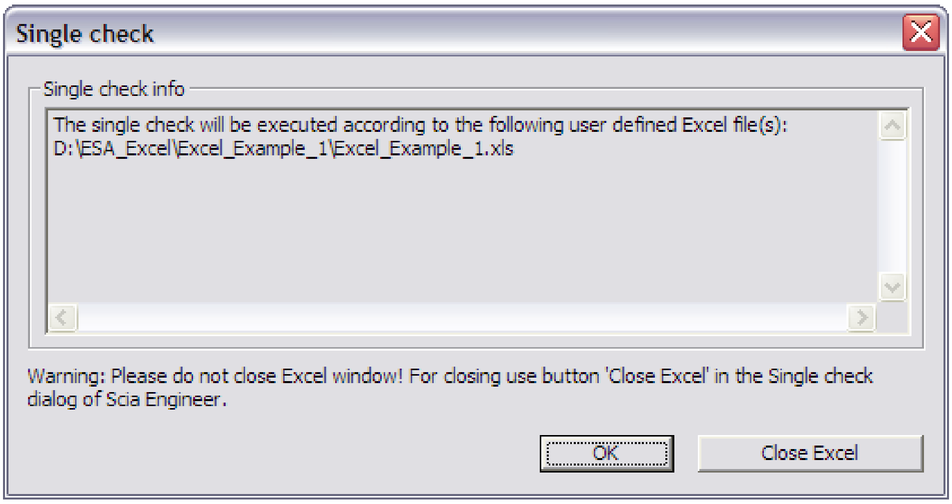
The action button **Single Check** can be used to directly open the Excel file after the mapping data has been sent. This provides an easy way to check if all data has been sent correctly to Excel.

When clicking the **Single Check** action button, the following message is given in the command line:



As specified previously, the Custom Check is executed on user defined additional data and therefore this data has to be selected instead of members or nodes.

After selecting the additional data, the following dialog is shown:



Important note: The Single Check dialog shows a clear warning. The Excel window which is opened may NOT be closed. It should be hidden using the Close Excel button in the Single Check dialog.

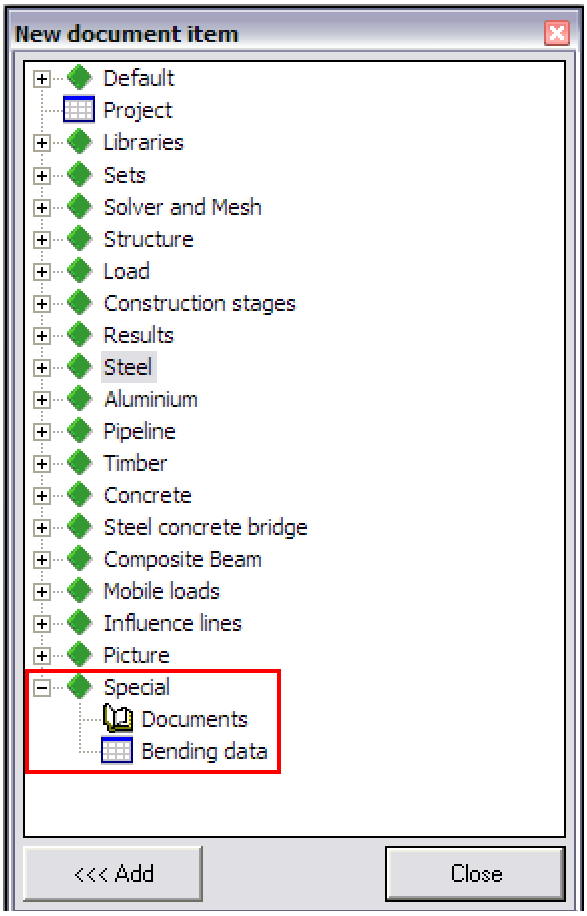
After confirming the dialog with [OK] Excel is opened and the Excel file is shown after all data have been sent to it:

	A	B	C	D
1	**Bending Check**			
2				
3	**Data from SCIA Engineer**			
4				
5	Moment My	45000	Nm	
6				
7	Section modulus Wel	1,94E-04	m^3	
8				
9	Section modulus Wpl	2,21E-04	m^3	
10				
11	Yield Strength fy	235000000	2,35E+08	
12				
13	Safety factor Gamma M	1,1	-	
14				
15	Elastic Check ?	1		
16				
17	**Bending Resistance**			
18				
19	MRd	41509,55	Nm	
20				
21	**Unity Check**			
22				
23	UC	1,08	-	

The check has now been executed and reviewed. To end this step, the document of Scia Engineer is examined.

In the document, the inputted User defined additional data can be inserted into the document in the same way as any other default additional data.

In the **New document item** dialog, the **Special** chapter holds the tables for all user defined additional data.



In this example **Bending data** was defined and thus this data can be added into the document.

Project	Excel_Example_1
Part	-
Description	Example 1 for Tutorial Excel Link
Author	PVT

1. Bending data

Type Name	Name	Member	Extent	Pos x_1	Pos x_2	Coor	Orig	Gamma M	Elastic Check
Bending data	Bend1	B1	full	0,000	1,000	Rela	From start	1,1	✓

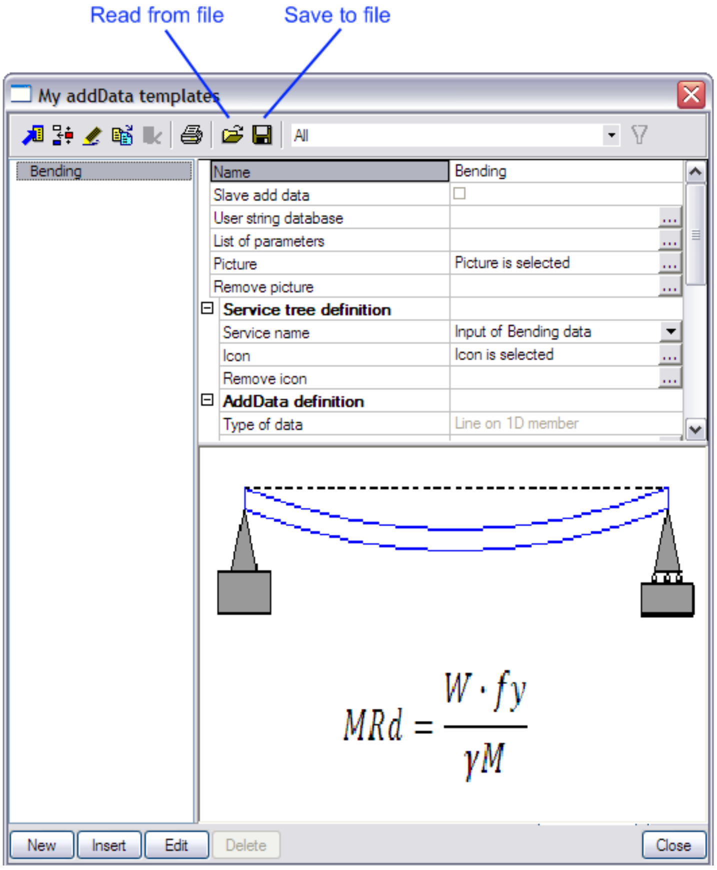
The table shows the different properties of the additional data, including the user defined parameters ‘Gamma M’ and ‘Elastic Check’. In the same way as for any other default additional data of Scia Engineer this table can be edited and modified through the Table Composer.

Step 5: Save the User Defined Additional Data into a database for future use

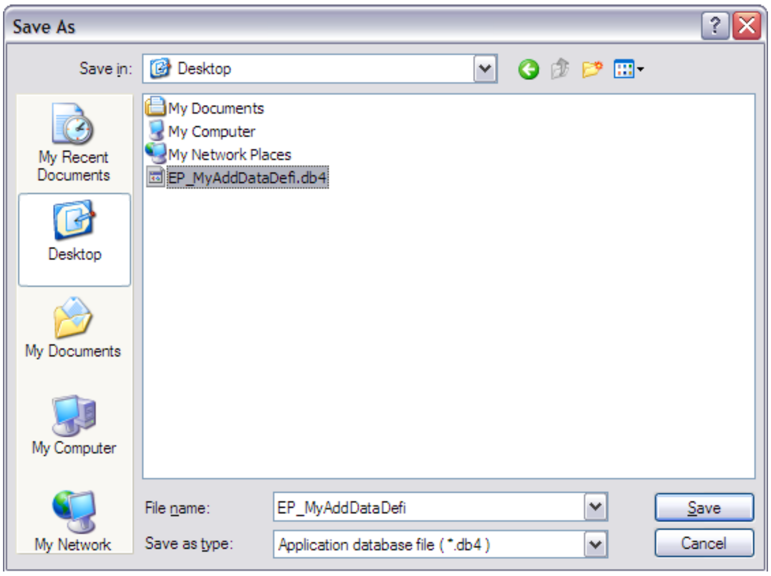
In the previous steps it has been explained how to define additional data, how to perform the input and execute the check. In this final step it is specified how this definition of additional data can be saved for easy use in other projects.

The User defined additional data can be saved into a database using the standard Scia Engineer functionality of libraries.

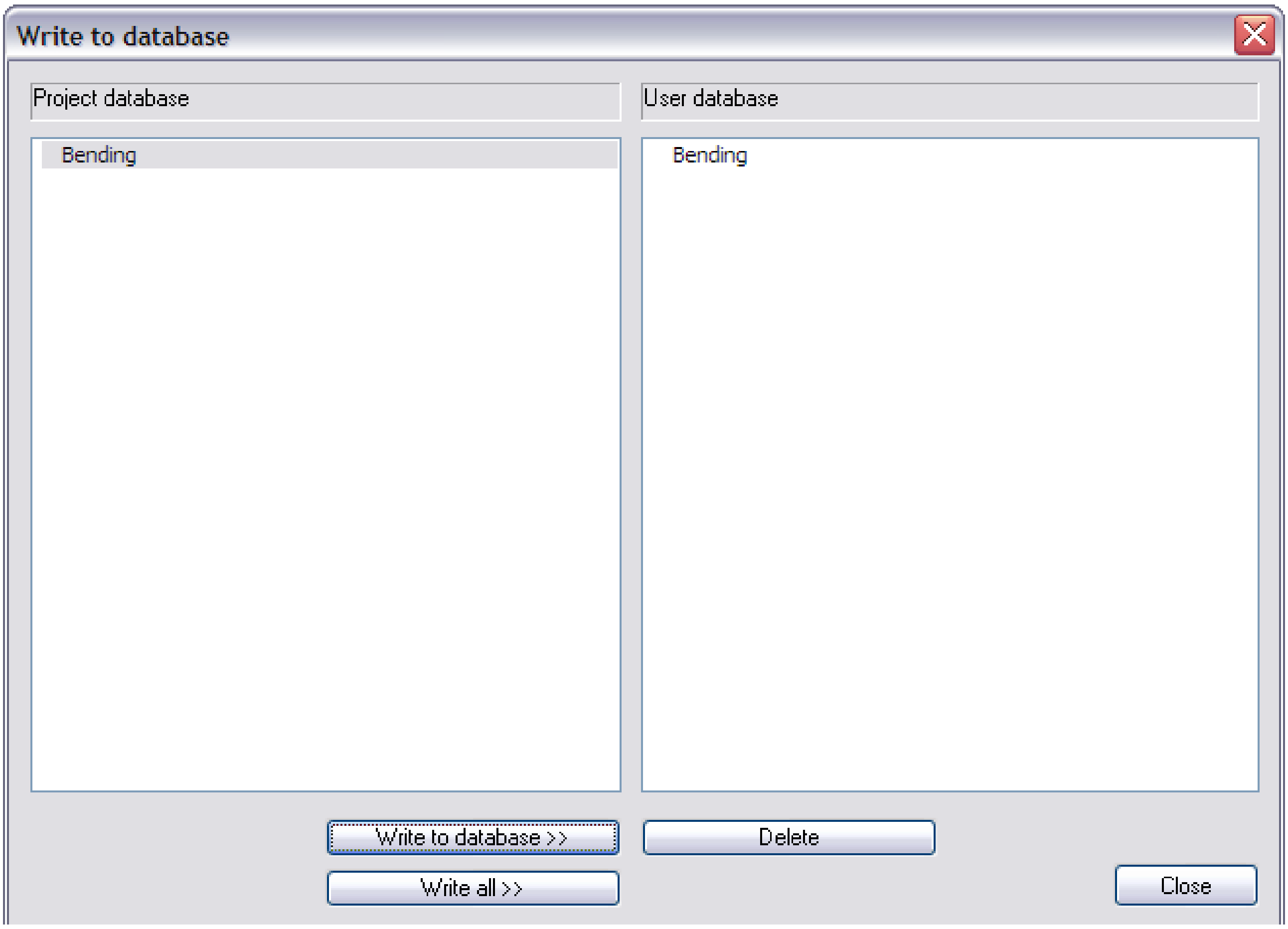
Through **Tools** > **User defined AddData** the **User Defined Additional Data** Library can be opened.



Using the button **Save to file** the desired additional data's can be added into a database file (db4 file).



A new db4 filename can be inputted or an existing file can be specified.



Using **Write to database >>** the selected additional data is added into the database. In this example the 'Bending data' is added to the database file.

In any other project, after activating the **External Application Checks** functionality and opening the **User Defined Additional Data** Library the additional data can be read directly from the database using the button **Read from file**.

This functionality works in the same way as for Materials, Cross-sections, Load cases, …

In this way, all data (definition of the additional data, definition of the check, icon, picture, mapping, …) is already defined. Step 2 thus becomes very easy and one can continue directly with Step 3, the input of the additional data.

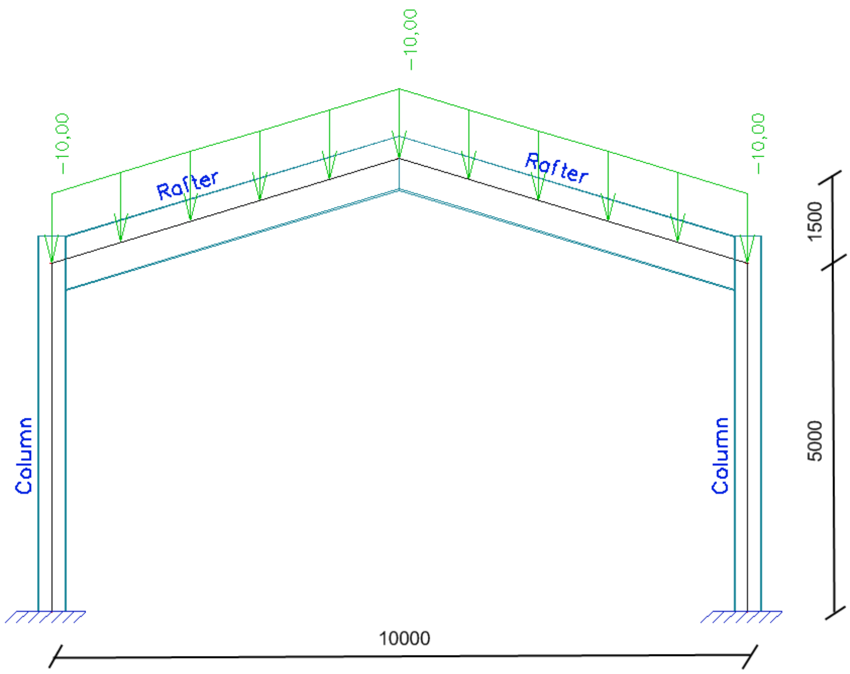
When using additional data from a database in another project, make sure that the paths to the Excel files are still valid! In case an Excel file cannot be found, the check will not be executed.

Example 2: Flange Induced Buckling

In this second example, the use of a combo-box is illustrated. In addition, the use of Named cells and output parameters with units are explained.

As a practical case, Flange Induced Buckling as specified in article 8 of EN 1993-1-5 is used.

In this example, a frame with rigid supports is modelled. The frame has a column distance of **10m**, a column height of **5m** and a column top to ridge height of **1,5m**. All members are manufactured in **S355** according to **EC-EN**.



The columns have a sheet welded Iwn cross section with parameters (400, 12, 300, 12, 200, 16). The rafters have a sheet welded Iwn cross section with parameters (750, 12, 500, 12, 200, 16).

One load case is defined, a uniform line load of **10 kN/m** on the rafters.

Flange Induced Buckling as specified in article 8 of EN 1993-1-5 concerns the following:

(1) To prevent the compression flange buckling in the plane of the web, the following criterion should be met:

$$\frac{h_w}{t_w} \leq k \frac{E}{f_{yf}} \sqrt{\frac{A_w}{A_{fc}}} \tag{8.1}$$

where A_w is the cross section area of the web;

A_{fc} is the effective cross section area of the compression flange;

h_w is the depth of the web;

t_w is the thickness of the web.

The value of the factor k should be taken as follows:

- plastic rotation utilized $k = 0,3$
- plastic moment resistance utilized $k = 0,4$
- elastic moment resistance utilized $k = 0,55$

Cross-section and material properties will be sent to Excel. In addition, the factor k will be determined according to a combo-box setting. The combo-box contents are those shown in the blue rectangle of the previous picture.

The check is done using the corresponding Excel file “**Excel_Example_2.xls**”

The Excel file contains two worksheets. On the sheet ‘Input’ the input data from Scia Engineer are set:

	A	B	C
1	Top flange width	0,3	m
2			
3	Top flange thickness	0,012	m
4			
5	Bottom flange width	0,2	m
6			
7	Bottom flange thickness	0,016	m
8			
9	Cross-section height	0,4	m
10			
11	Web thickness	0,012	m
12			
13			
14	Yield Strength	355000000	N/m^2
15			
16	E-modulus	2,1E+11	N/m^2
17			
18			
19	Determination of factor k	Elastic moment resistance	
20			
21	Moment My	50000	Nm

The sheet ‘Check’ shows the intermediate results and the unity check:

	A	B	C	D	E	F
1	web height hw	372	mm			
2						
3	web thickness tw	12	mm			
4						
5	Web area Aw	4464	mm^2			
6						
7	Compression flange	Top				
8						
9	Compression flange area Afc	3600	mm^2		0,0036	m^2
10						
11	Factor k	0,55				
12						
13	Yield Strength	355	N/mm^2			
14						
15	E-modulus	210000	N/mm^2			
16						
17						
18	Web slenderness	31				
19						
20	Limit slenderness	362,30				
21						
22	Unity Check	0,086				

Since the check requires the area of the compression flange, the sign of the bending moment is used to determine which flange is in compression at each section along a member.

In the Excel file, all cells to which data has to be mapped and from which data is read have been given a name. This allows for a very easy definition of the mapping since these same names will be available in the mapping dialog of Scia Engineer.

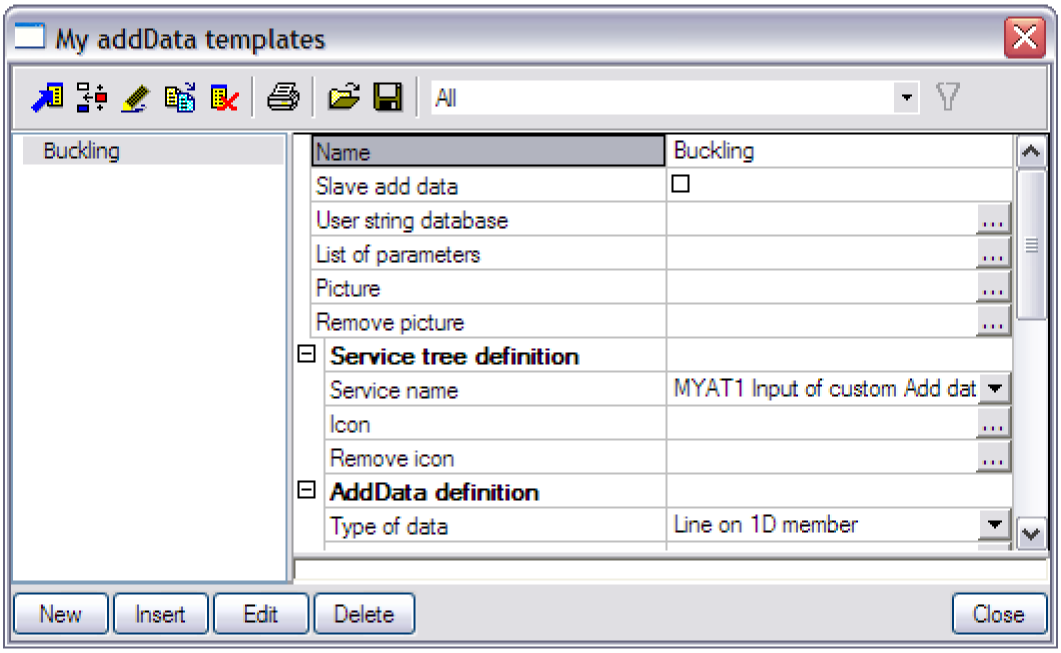
Step 1: Activate the functionality External Application Checks

The first step is to activate the functionality **External application checks** on the **Functionality** tab in the **Project Data**.

Step 2: Create User Defined Additional Data

In the second step, User Defined Additional Data will be defined.

Through **Tools** > **User defined AddData** the **User Defined Additional Data** Library can be opened.



The **Name** of the additional data is changed to ‘Buckling’.

Step 2.1 Slave data

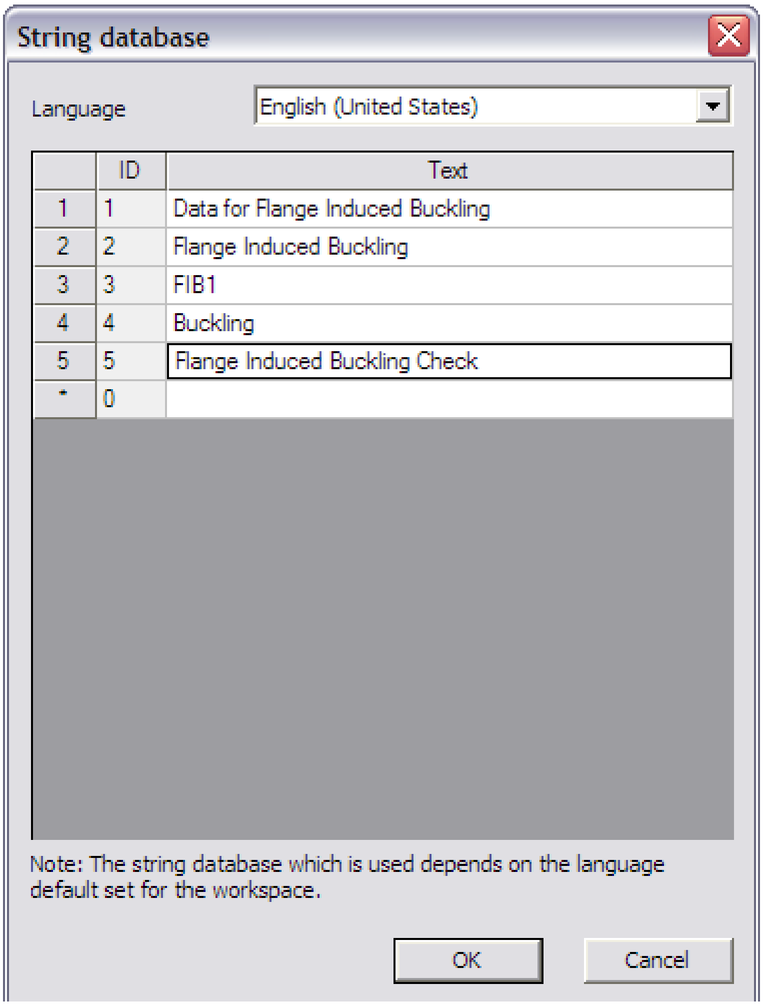
Only one type of additional data will be defined here and as such the check-box **Slave add data** is left unchecked.

Step 2.2 Define text strings

In the **User string database** the required strings are defined for the definition of the additional data.

Since in this example Flange Induced Buckling is being illustrated the strings are modified as follows:

Type for which the string is used	Default string	String used in this example
Service name	MYAT1 Input of custom add data	Data for Flange Induced Buckling
Type name	MYAT1 Custom defined add data	Flange Induced Buckling
Short name	MYAT1 MADI	FIB1
Description	MYAT1 Description	Buckling
Name of check	MYAT1 Custom check	Flange Induced Buckling Check



The necessary strings for the definition of the data have been inputted and in the next step the parameters can be defined.

Step 2.3 Define parameters

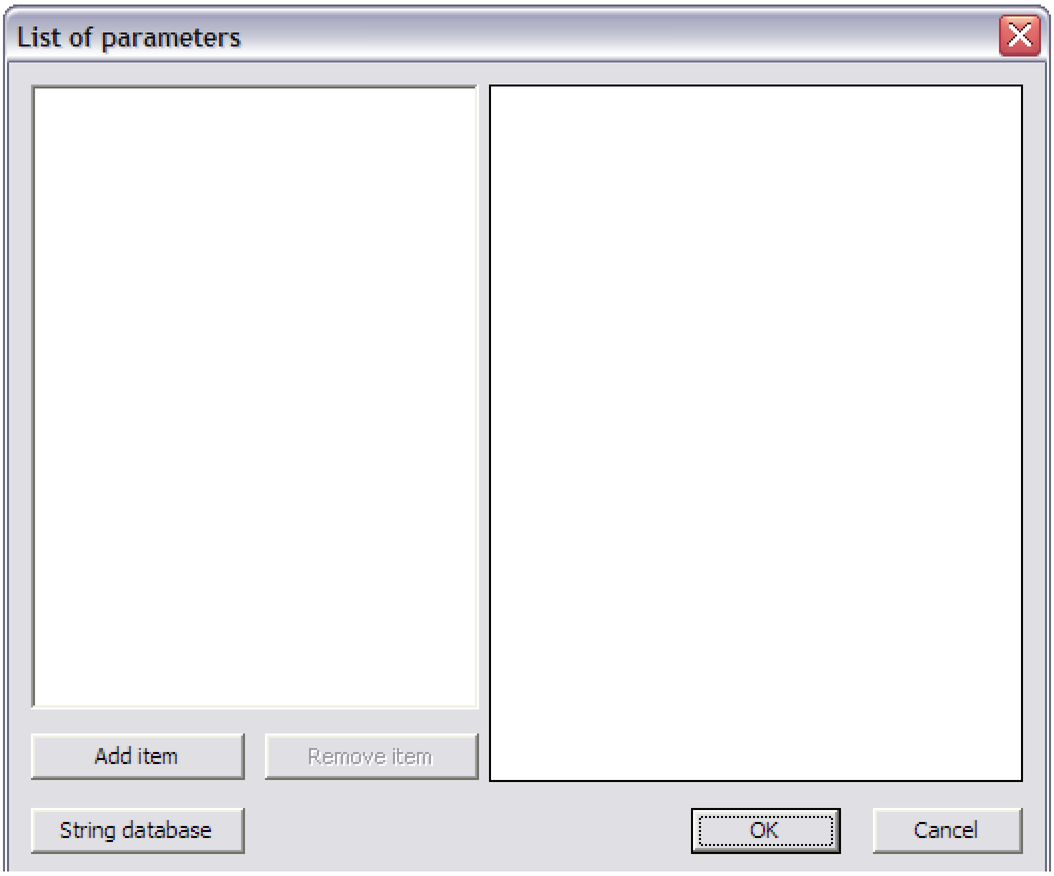
In this example, the mapping will concern of default Scia Engineer data (Cross-section dimensions, material properties and internal forces) except for the determination of the factor k.

In the code, this parameter was defined as follows:

The value of the factor k should be taken as follows:

- plastic rotation utilized $k = 0{,}3$
- plastic moment resistance utilized $k = 0{,}4$
- elastic moment resistance utilized $k = 0{,}55$

To choose between these options, a combo-box parameter will be defined through **List of parameters**.



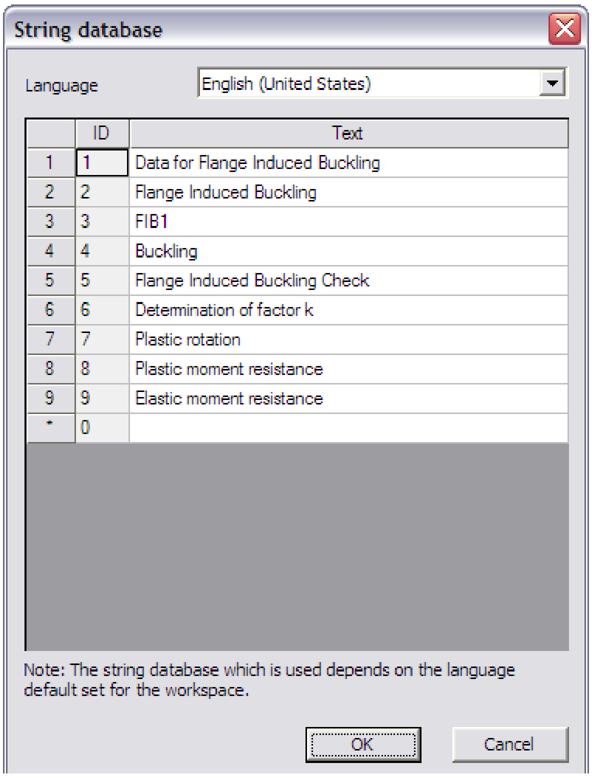
For this example, one parameter will thus be defined:

Parameter	Type	Combo-box lines
Determination of factor k	Combo-box	Plastic rotation Plastic moment resistance Elastic moment resistance

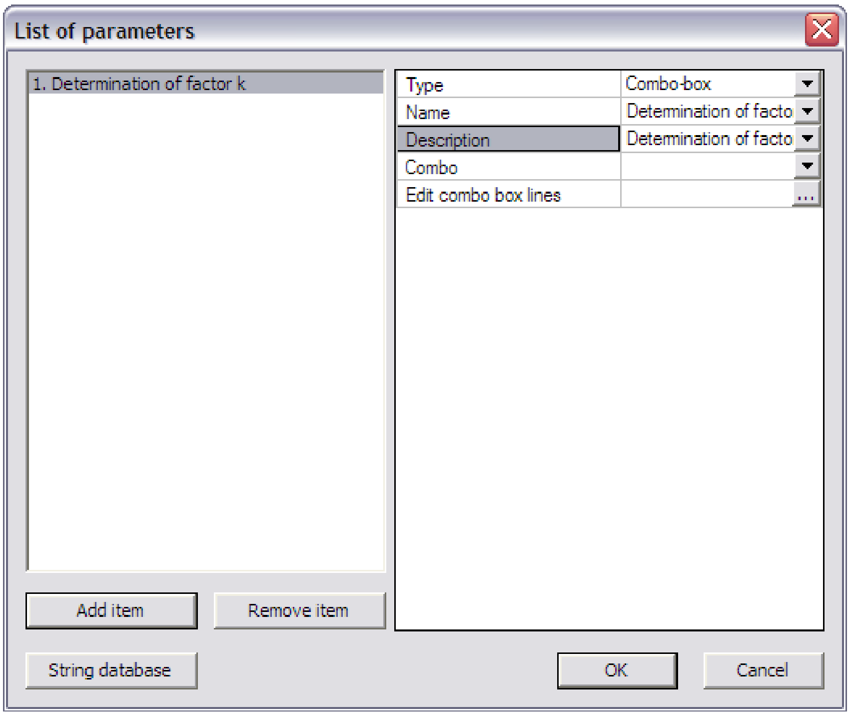
Through the button **String database** the text string database can be directly accessed. This allows a quick input of the strings required for the parameters.

For this example the following strings are added:

Strings used in this example
Determination of factor k
Plastic rotation
Plastic moment resistance
Elastic moment resistance



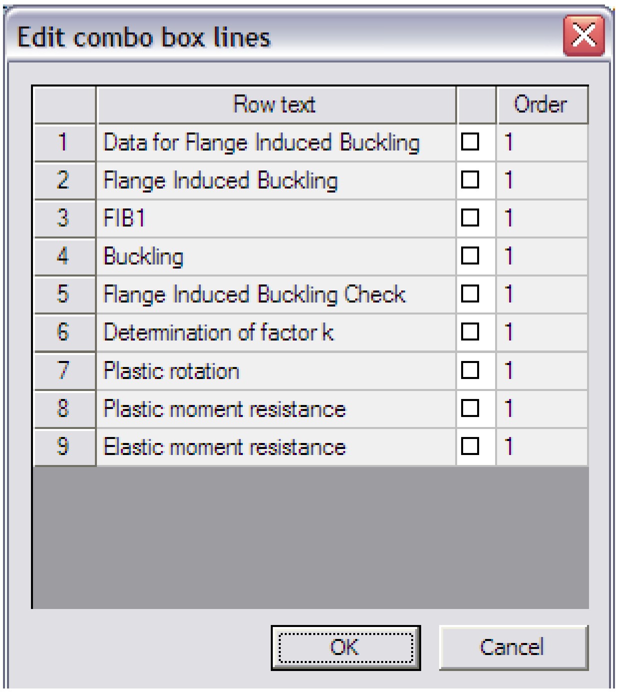
Next, through the button **Add item** the parameter is added.



The **Type** field is set to ‘Combo-box’.

For both the **Name** and **Description** fields the string ‘Determination of factor k’ is set.

Next, the lines in the combo-box are defined through the edit button **Edit combo box lines**.

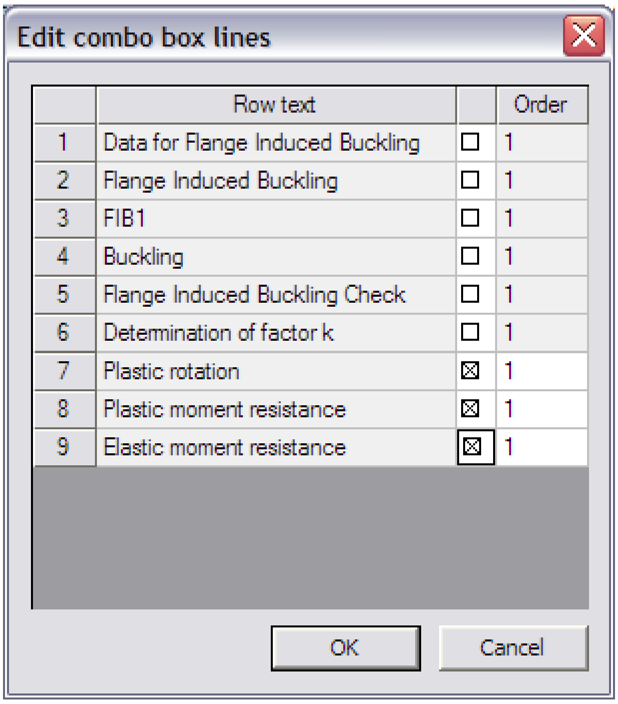


Edit combo box lines

	Row text		Order
1	Data for Flange Induced Buckling	☐	1
2	Flange Induced Buckling	☐	1
3	FIB1	☐	1
4	Buckling	☐	1
5	Flange Induced Buckling Check	☐	1
6	Determination of factor k	☐	1
7	Plastic rotation	☐	1
8	Plastic moment resistance	☐	1
9	Elastic moment resistance	☐	1

OK Cancel

This dialog shows all strings defined in the user string database in the column **Row text**. The checkboxes can be used to specify which strings should be in the combo-box. For this example, the three final strings are thus activated.



Edit combo box lines

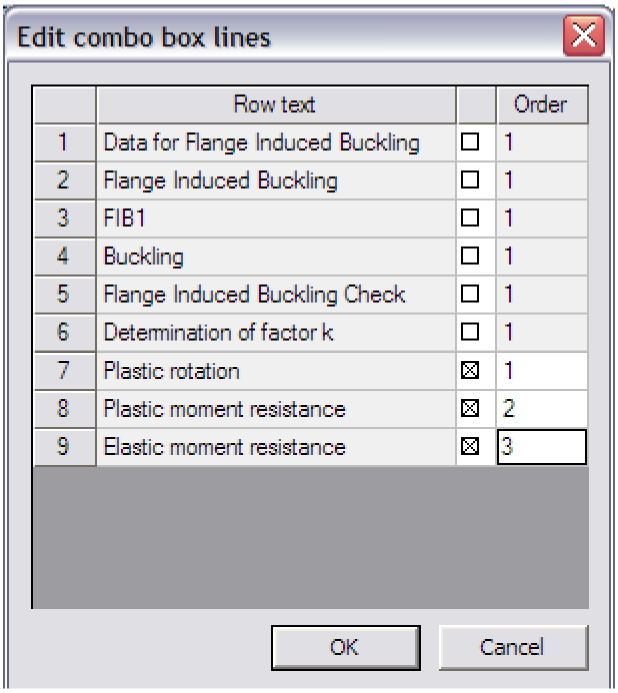
	Row text		Order
1	Data for Flange Induced Buckling	☐	1
2	Flange Induced Buckling	☐	1
3	FIB1	☐	1
4	Buckling	☐	1
5	Flange Induced Buckling Check	☐	1
6	Determination of factor k	☐	1
7	Plastic rotation	☒	1
8	Plastic moment resistance	☒	1
9	Elastic moment resistance	☒	1

OK Cancel

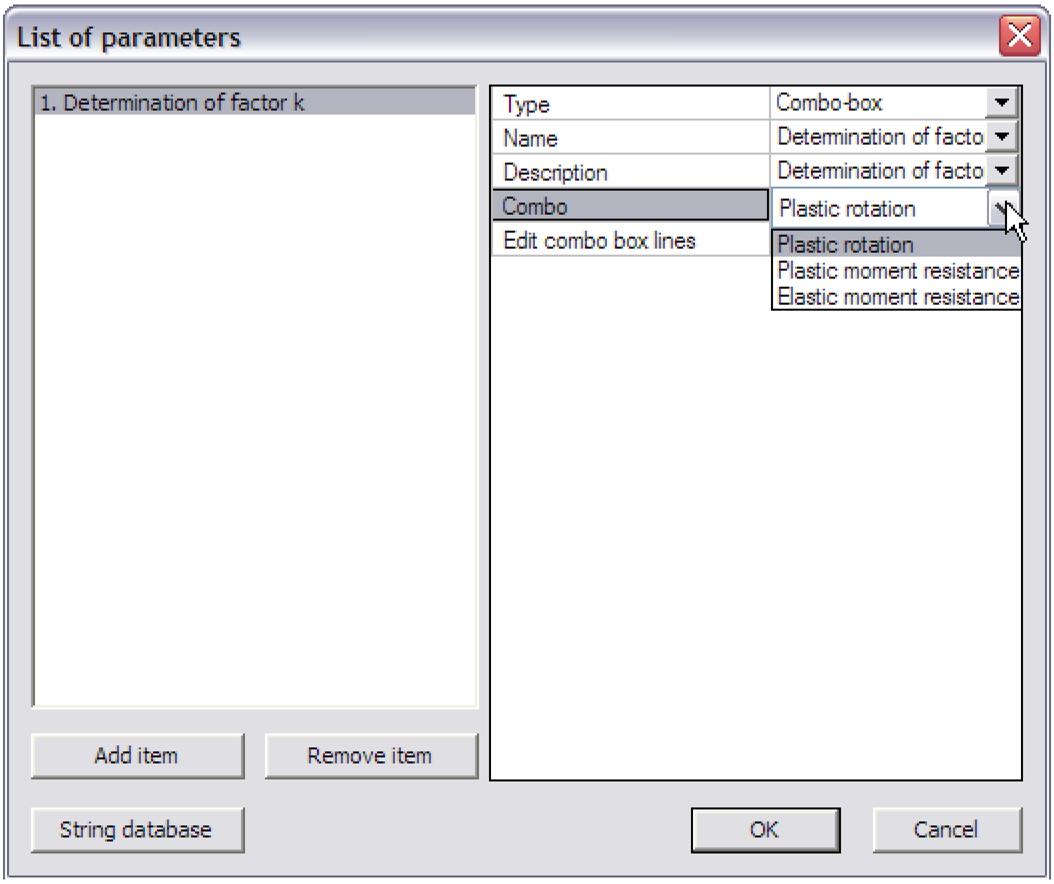
The lines for the combo-box have been defined so the next step is to set the order of the lines in the **Order** column.

> ***The line with Order number 1 will be the first line in the combo-box. Each next line should have its Order incremented by 1.***

For this example, the order in which the strings have been inputted in the string database is kept and thus in the Order column the numbers ‘1’, ‘2’ and ‘3’ are inputted.



When closing this dialog, the **Combo** item in the **List of Parameters** dialog shows how the combo-box will look like.

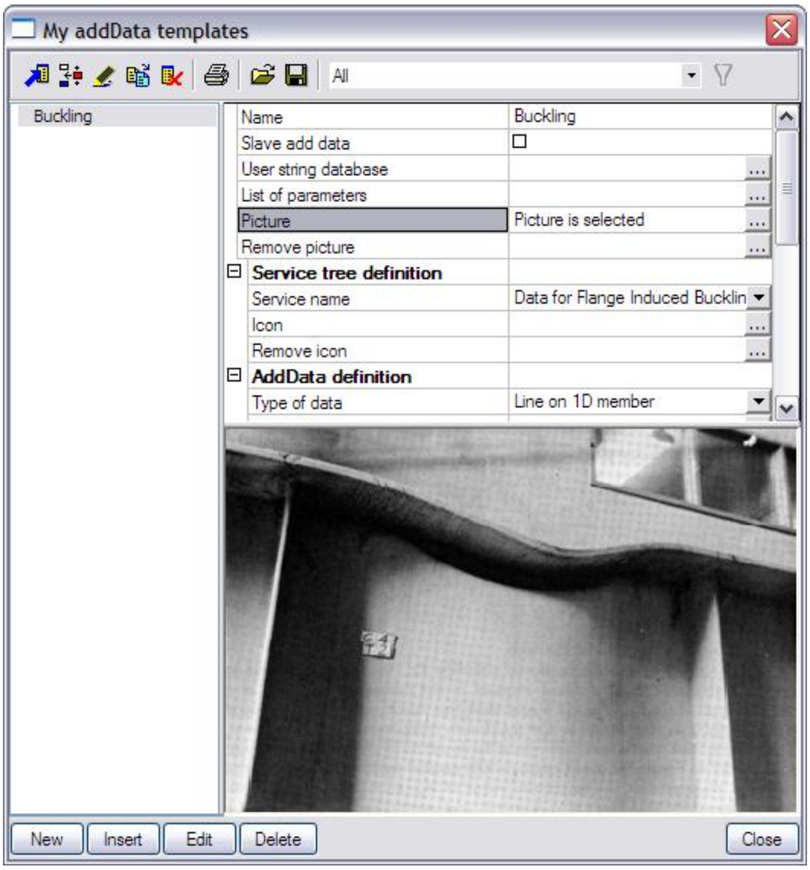


The combo-box parameter has now been defined and the dialog can be closed.

Step 2.4 Add a picture to the Additional Data

To clarify the use of the additional data and the defined parameters a picture can be added using the **Picture** button.

In this example the picture **Excel_Example_2_Picture.bmp** will be used.



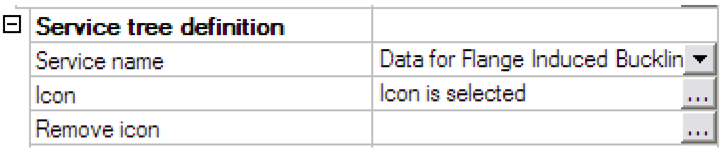
Step 2.5 Define Service Tree

In the next step the Service Tree is defined through the group **Service tree definition**.

The **Service name** is taken automatically from the text string database.

To clarify the Service name, an icon can be added using the **Icon** button.

In this example the icon **Excel_Example_2_Icon.bmp** will be used.



Step 2.6 Define the Additional Data

Using the data from the previous steps, the additional data can now be defined in the group **AddData definition**.

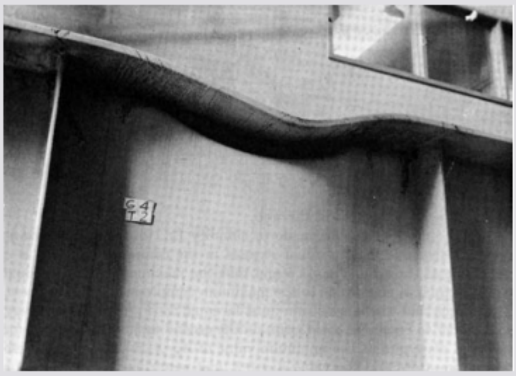
AddData definition	
Type of data	Line on 1D member
Instance setup	...
Type name	Flange Induced Buckling
Short name	FIB1
Description	Buckling

Flange Induced Buckling has to be checked in each section of the member since the check contains the area of the compression flange which can change along the length of the member. Therefore the field **Type of data** is set to 'Line on 1D member'.

The **Type name**, **Short name** and **Description** are taken automatically from the text string database.

To get an overview of all the data entered in the previous steps the button **Instance Setup** is used.

Flange Induced Buckling



Name	
Parameters	
Determination of factor k	Plastic rotation
Drawings	
Drawing style	Box on line
Property for drawing on begin	-
Property for drawing on end	-
Colour	Others
Geometry	
Extent	full
Position x1	0,000
Position x2	1,000
Coord. definition	Rela
Origin	From start

OK Cancel

The **Parameters** group shows the combo-box defined in *Step 2.3*.

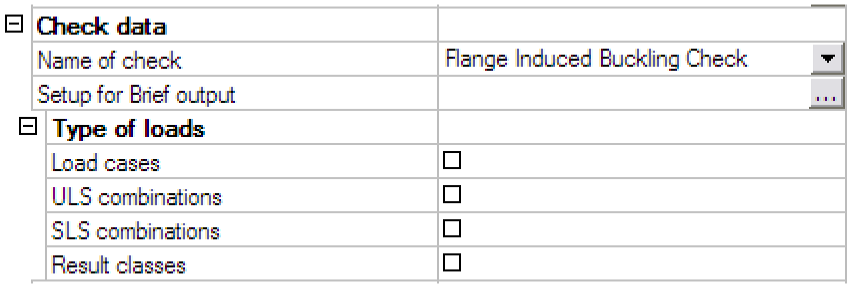
In the **Drawings** group, the **Drawing style** is set to 'Simple'. For the **Colour field** 'Water load' is chosen.

Drawings	
Drawing style	Simple
Property for drawing on begin	-
Property for drawing on end	-
Colour	Water load

Since in this example no numerical parameter was defined, no **Property for drawing on begin/end** is specified.

Step 2.7 Define the Check

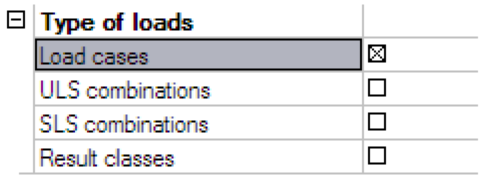
In the group **Check data** the necessary data for the check itself can now be defined.



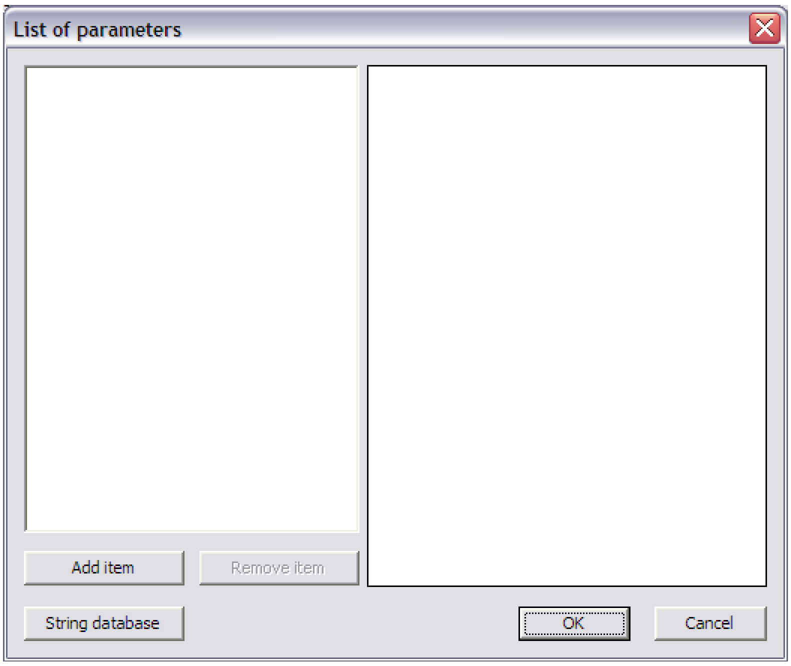
The **Name of check** is taken automatically from the text string database.

The **Type of loads** group allows to specify which load types will be available for the check. Only the selected items will be available when executing the check.

Since in this example only one load case was defined, only the option 'Load cases' will be activated.



The final item for defining the check is the **Setup for Brief output** where the output parameters have to be defined.

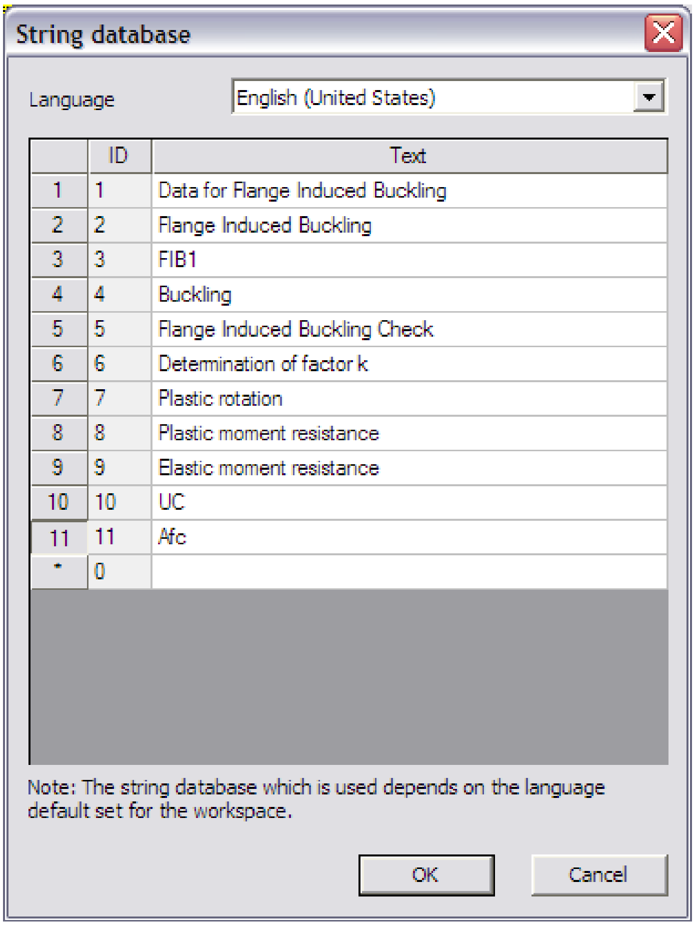


For this example, two parameters will be defined: the unity check value and the area of the compression flange.

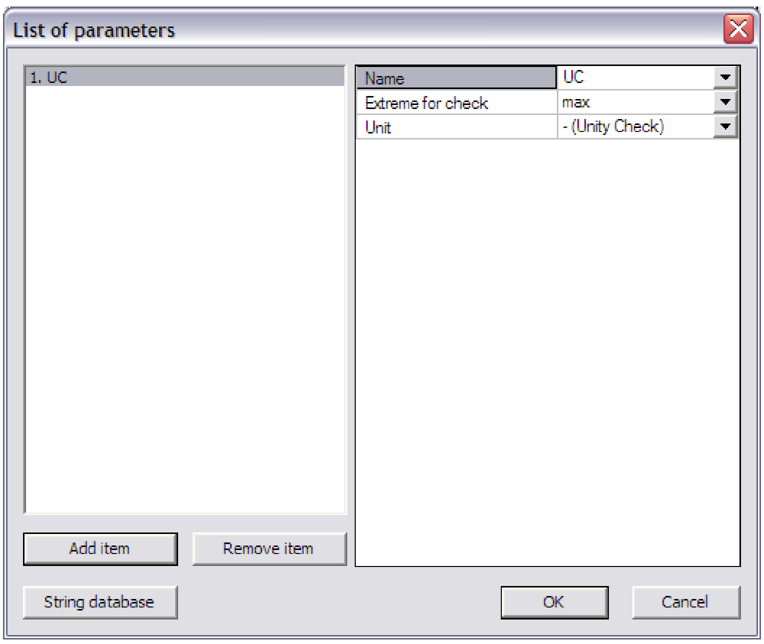
Parameter	Unit
Unity Check UC	-
Area of compression flange Afc	mm²

First of all, through the button **String database** the text string database is accessed to define the required strings. For this example the following strings are added:

Strings used in this example
UC
Afc



When the strings are defined, the first parameter is added through the button **Add item**.

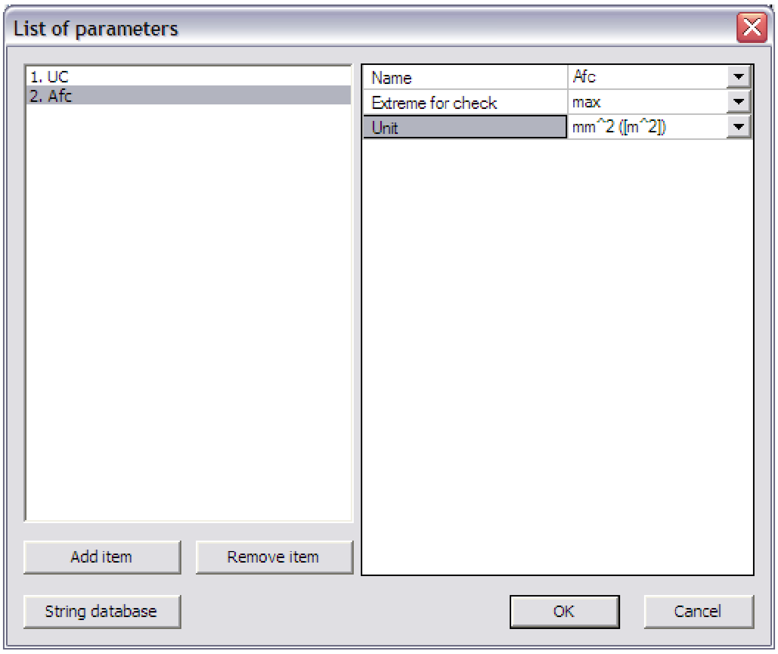


In the **Name** field the ‘UC’ string is chosen from the string database.

The **Extreme for check** is left on ‘max’ since the maximal unity check value is extreme in this case.

Since it concerns a unity check, the **Unit** field is left on ‘– (Unity Check)’.

Again using the button **Add item** the second parameter is added.



In the **Name** field the ‘Afc’ string is chosen from the string database.

The **Extreme for check** is left on ‘max’. For this example it is of no importance if the extreme is minimum or maximum, the purpose of the parameter is to see which flange is in compression.

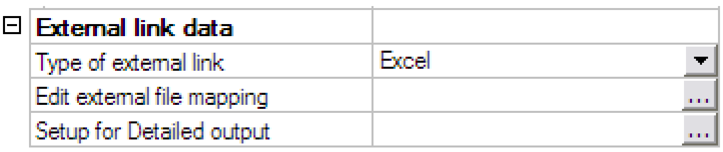
Since this parameter concerns an area, the **Unit** field is set to ‘mm^2 ([m^2])’.

During the mapping, parameters are always sent to Excel in basic SI units. Output parameters are also read from Excel in basic SI units. For this example this implies that the area in Excel has to be in m^2 as indicated in the Unit field. This unit will then be converted to mm^2 in Scia Engineer.

The check and output parameters have now been defined so in the next step the link can be set.

Step 2.8 Specify the type of external link

In the **External link data** group the **Type of external link** allows to specify which external application will be used.



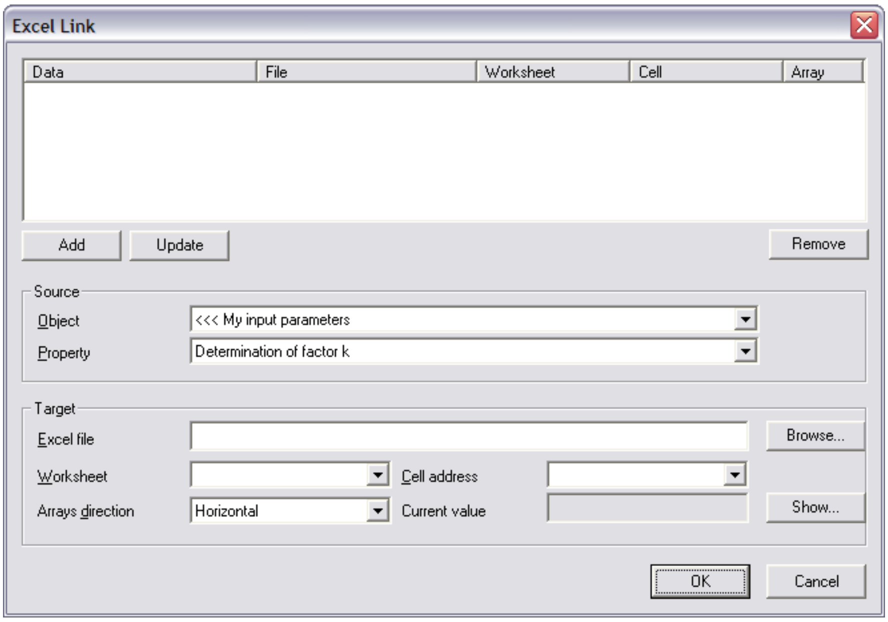
In this example the link is made with Excel and thus ‘Excel’ is chosen.

Step 2.9 Define the mapping with the external application

All preparation has now been done, what remains is the most important step of the process: defining the actual mapping between properties and parameters of Scia Engineer and the data fields (i.e. Excel cells) of the external application.

Through the button **Edit external file mapping** the mapping dialog is opened.

The first time the mapping dialog is opened can take a few seconds. This is because, during the opening, all document tables are refreshed since these properties are available in the mapping dialog. This way, when new items are added to the document in future versions of Scia Engineer, they will automatically be available in the mapping dialog also.



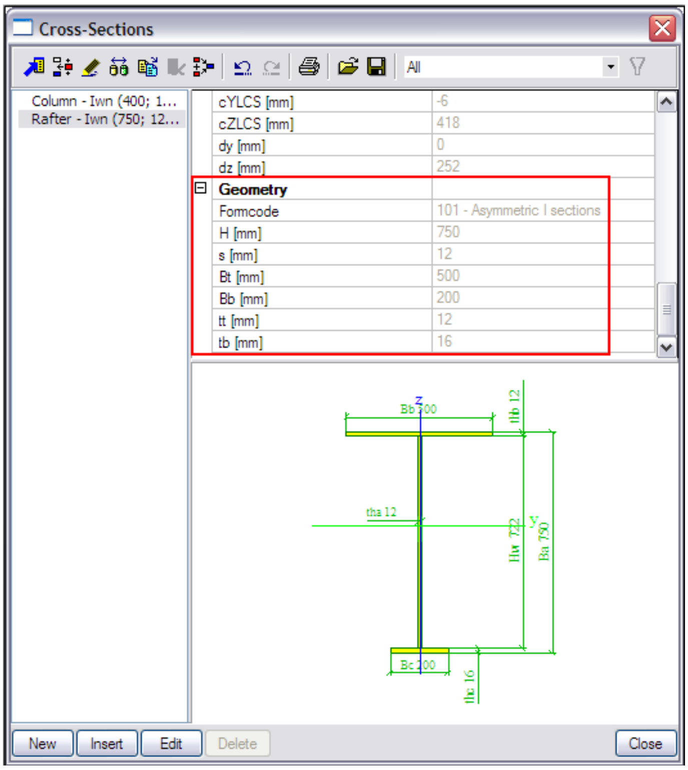
In this example, the Excel file contains two worksheets. On the sheet 'Input' the input data from Scia Engineer are set:

	A	B	C
1	Top flange width	0,3	m
2			
3	Top flange thickness	0,012	m
4			
5	Bottom flange width	0,2	m
6			
7	Bottom flange thickness	0,016	m
8			
9	Cross-section height	0,4	m
10			
11	Web thickness	0,012	m
12			
13			
14	Yield Strength	355000000	N/m^2
15			
16	E-modulus	2,1E+11	N/m^2
17			
18			
19	Determination of factor k	Elastic moment resistance	
20			
21	Moment My	50000	Nm

The sheet ‘Check’ shows the intermediate results and the unity check:

	A	B	C	D	E	F
1	web height hw	372	mm			
2						
3	web thickness tw	12	mm			
4						
5	Web area Aw	4464	mm^2			
6						
7	Compression flange	Top				
8						
9	Compression flange area Afc	3600	mm^2		0,0036	m^2
10						
11	Factor k	0,55				
12						
13	Yield Strength	355	N/mm^2			
14						
15	E-modulus	210000	N/mm^2			
16						
17						
18	Web slenderness	31				
19						
20	Limit slenderness	362,30				
21						
22	Unity Check	0,086				

In this example, cross-section properties have to be sent to Excel. In the Cross-section manager, it can be seen how the dimensions of a sheet welded Iwn section are defined:



More specifically the properties H, s, Bt, Bb, tt and tb will have to be mapped to Excel.

As specified in the introduction of this example, in the Excel file, all cells to which data has to be mapped and from which data is read have been given a name. These named cells can now be used in the **Cell address** field instead of manually typing the cell number.

The following table shows which properties should be mapped to which cells:

Object	Property	Worksheet	Named Cell (Address)
Cross-Sections	Bt (Geometry)	Input	Top_flange_width (B1)
Cross-Sections	tt (Geometry)	Input	Top_flange_thickness (B3)
Cross-Sections	Bb (Geometry)	Input	Bottom_flange_width (B5)
Cross-Sections	tb (Geometry)	Input	Bottom_flange_thickness (B7)
Cross-Sections	H (Geometry)	Input	Cross_section_height (B9)
Cross-Sections	s (Geometry)	Input	Web_thickness (B11)
Steel EC3	Yield strength (code independent)	Input	Yield_Strength (B14)
Steel EC3	E modulus (code independent)	Input	E_modulus (B16)
<<< My input parameters	Determination of factor k	Input	Determination_of_k (B19)
Internal forces on member	My	Input	Moment_My (B21)
>>> My output parameters	UC	Check	UC (B22)
>>> My output parameters	Afc	Check	Afc (E9)

As specified, during the mapping, parameters are always sent to Excel in basic SI units. Output parameters are also read from Excel in basic SI units. Therefore the cell E9 on the Check worksheet in Excel shows the area of the compression flange in SI units.

The mapping of the first property, the width of the top flange Bt is thus done as follows:

The **Object** field is set to ‘Cross-sections’.

In the **Property** field ‘Bt’ can then be chosen.

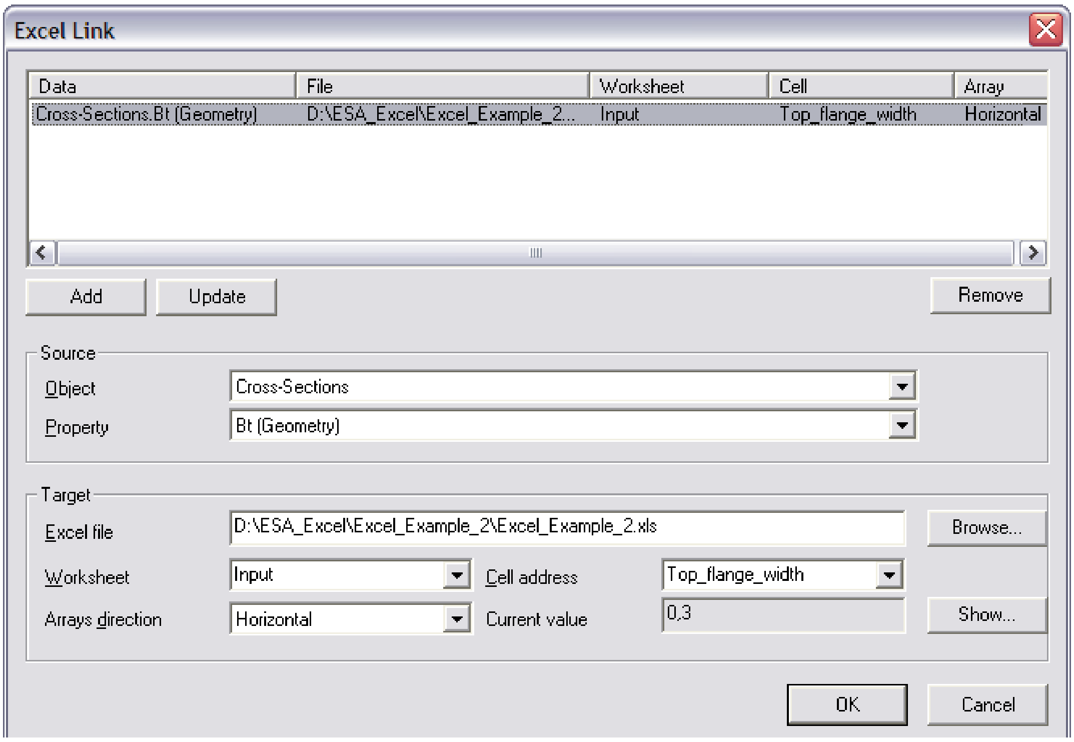
Using the **Browse** button, the file **Excel_Example_2.xls** is searched.

After the file has been specified, the **Worksheet** field contains a list of all sheets. This field is set to ‘Input’.

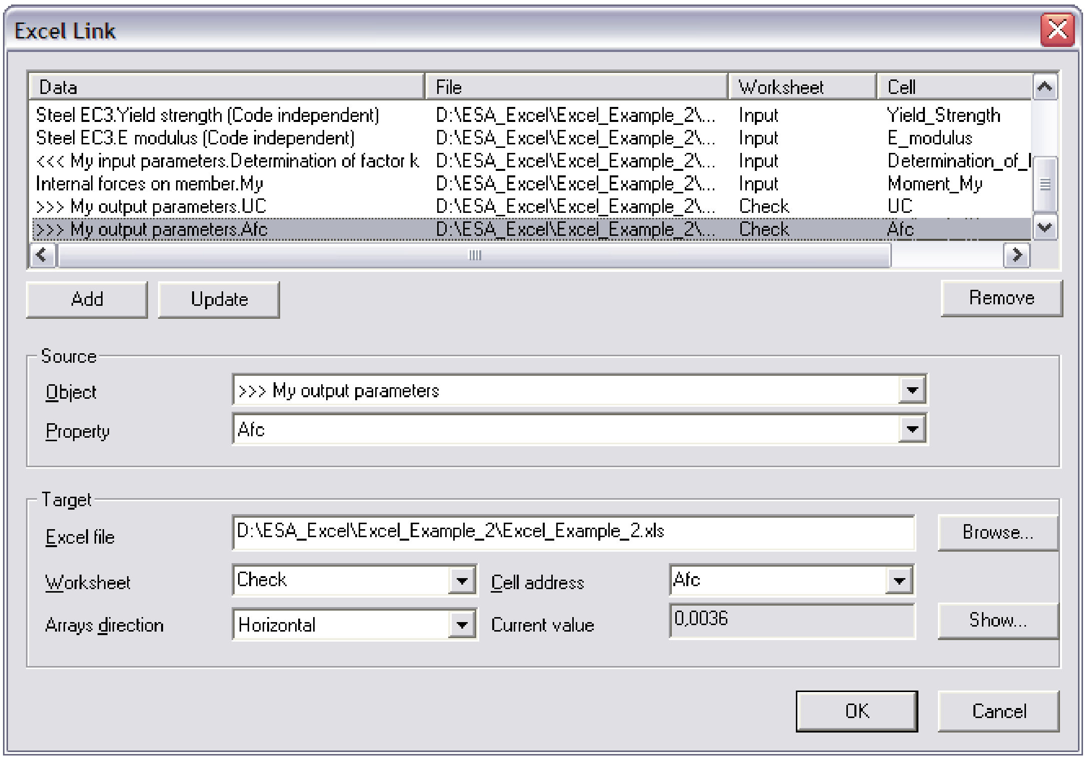
The **Arrays direction** is set to ‘Horizontal’. In this example no array properties are mapped so choosing ‘Horizontal’ or ‘Vertical’ would make no difference.

Finally, in the field **Cell address**, using the combo-box the named cell ‘Top_flange_width’ is chosen. Automatically the **Current value** field will show the current content of the cell, in this case 0,3.

When all input has been done, this mapping is added to the table using the **Add** button.



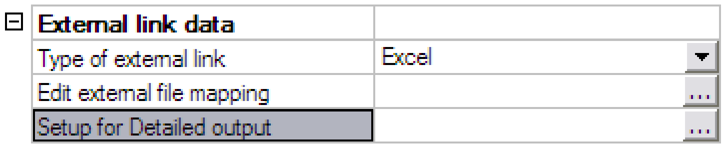
In the same way, all other parameters can be mapped using the above table. For all parameters the **Arrays direction** is set to 'Horizontal'.



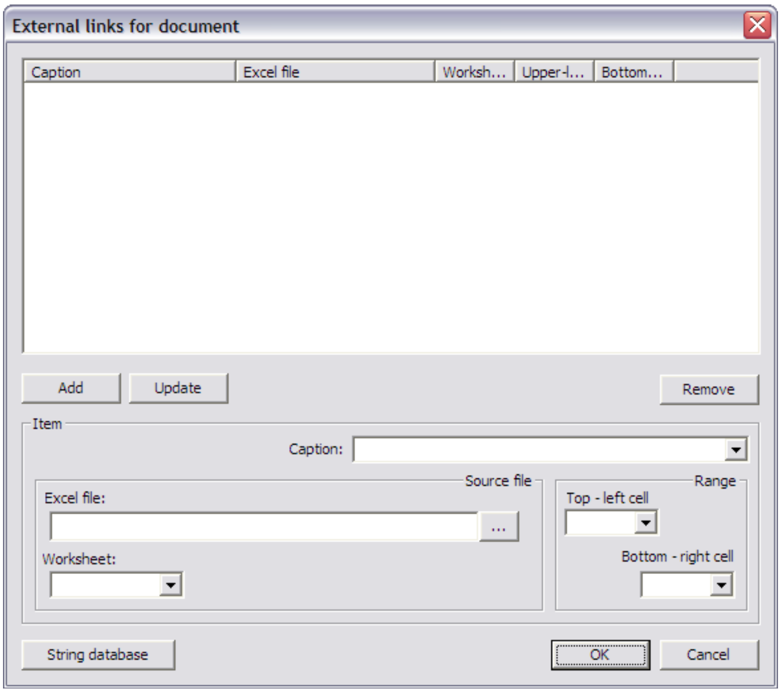
All parameters are now mapped to Excel. The final step left for the definition of the additional data is specifying a Detailed output.

Step 2.10 Define the Detailed output

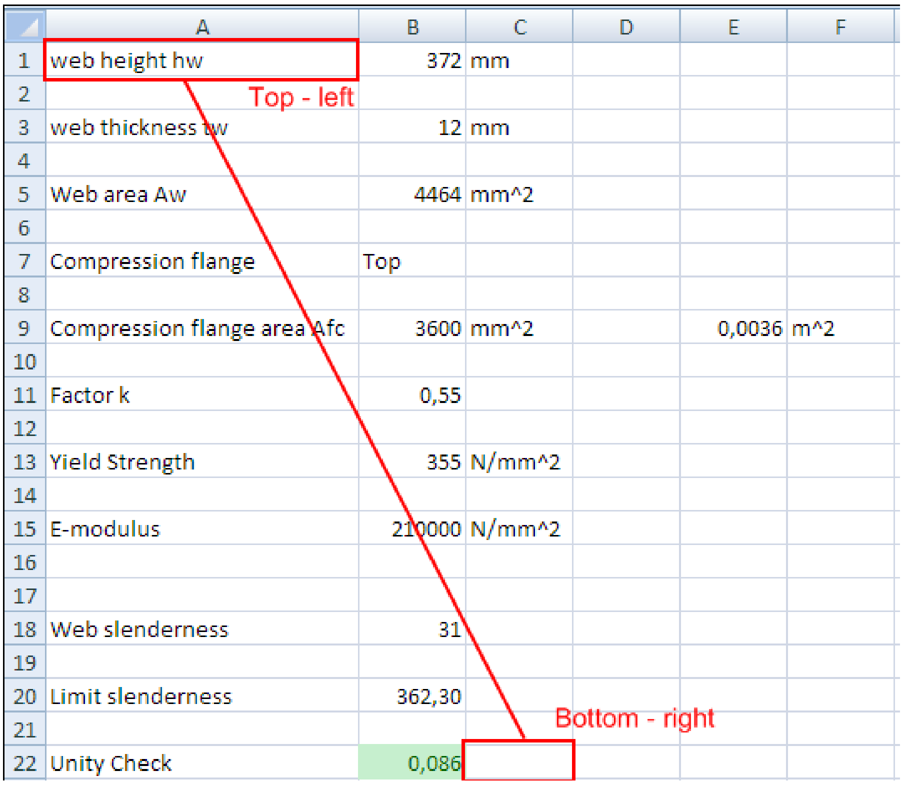
In *Step 2.7* the parameters for the Brief output have been defined. These parameters will be used for the check. In addition, a Detailed output can also be specified to show an in-depth overview of the check.



By clicking on **Setup for Detailed output**, the Detailed output dialog is opened.



In this example, the range will be defined from the cell A1 to the cell C22 on the ‘Check’ worksheet.



	A	B	C	D	E	F
1	web height hw	372	mm			
2						
3	web thickness tw	12	mm			
4						
5	Web area Aw	4464	mm^2			
6						
7	Compression flange	Top				
8						
9	Compression flange area Afc	3600	mm^2		0,0036	m^2
10						
11	Factor k	0,55				
12						
13	Yield Strength	355	N/mm^2			
14						
15	E-modulus	210000	N/mm^2			
16						
17						
18	Web slenderness	31				
19						
20	Limit slenderness	362,30				
21						
22	Unity Check	0,086				

For ease of reference, here also Named cells have been defined in the Excel file.

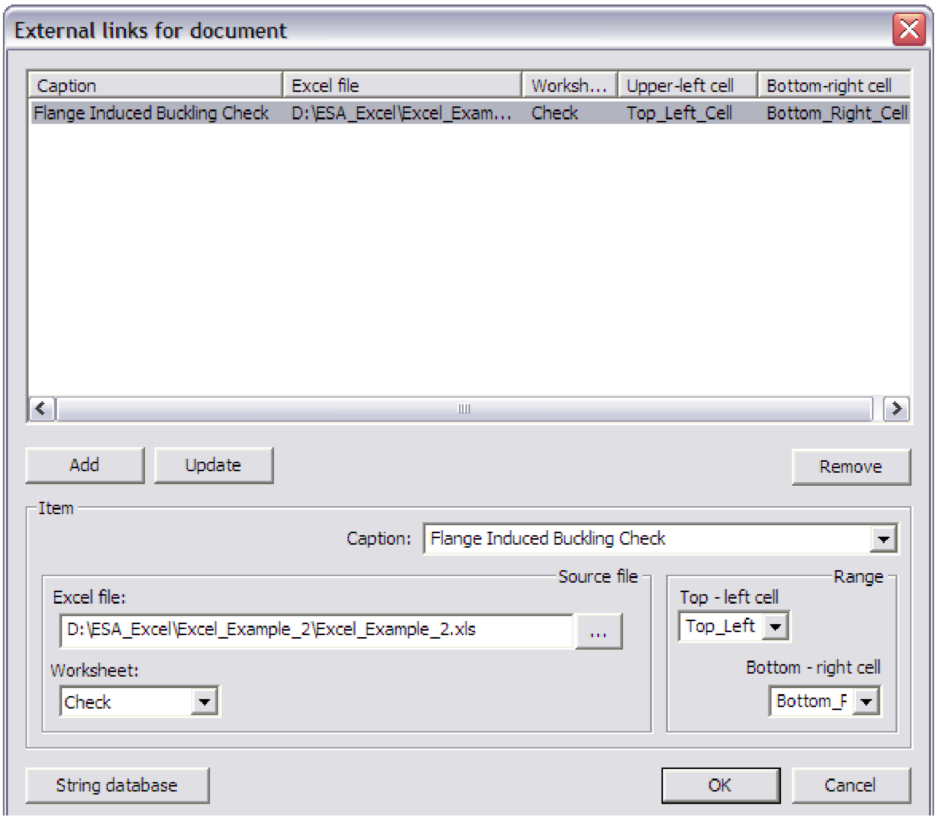
In the **Caption** field the string ‘Flange Induced Buckling Check’ is chosen.

In the **Excel file** field the file **Excel_Example_2.xls** is searched using the browse button.

The **Worksheet** field is set to ‘Check’.

In the **Range** group the **Top - left** cell is set to ‘Top_Left_Cell’ and the **Bottom - right** cell to ‘Bottom_Right_Cell’.

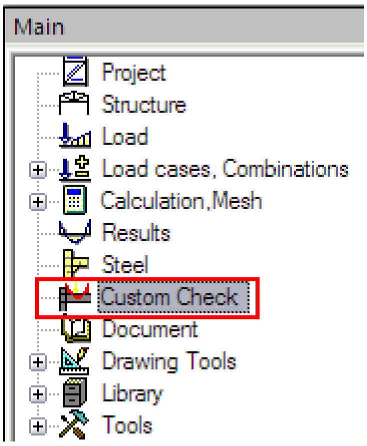
When all input has been done, the data is added to the table using the **Add** button.

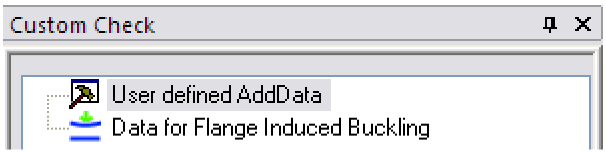


With this final step, the User Defined Additional Data has been fully inputted and the **User Defined Additional Data** Library can be closed.

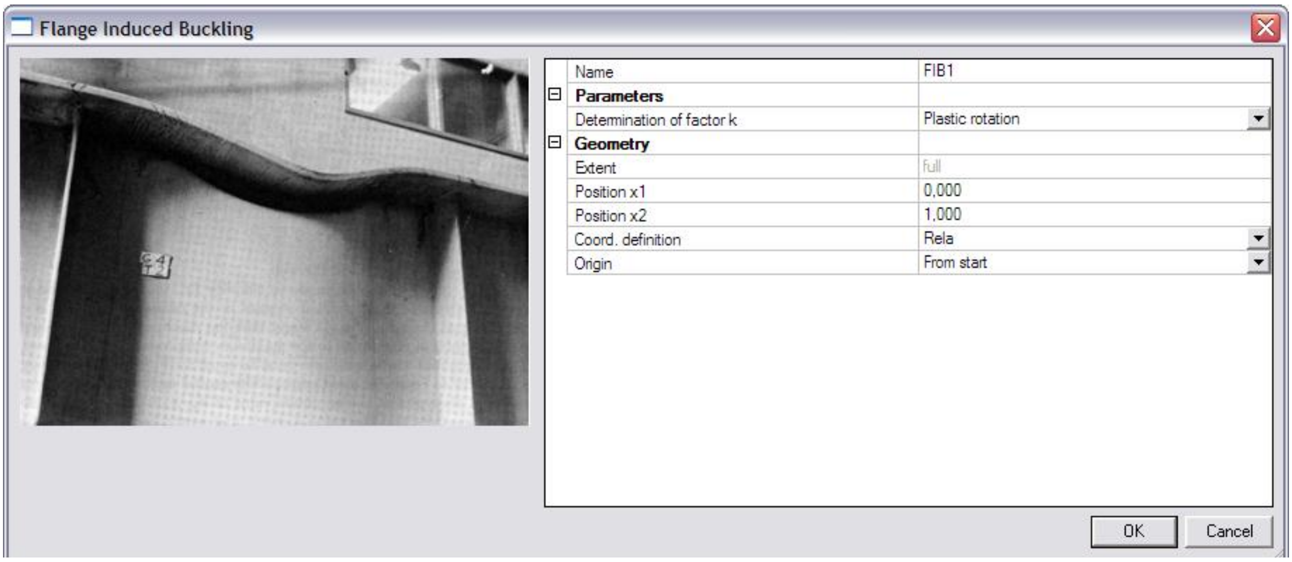
Step 3: Input the User Defined Additional Data on members/nodes

After closing the **User Defined Additional Data** Library a new service will be shown in the Scia Engineer tree: **Custom Check**.



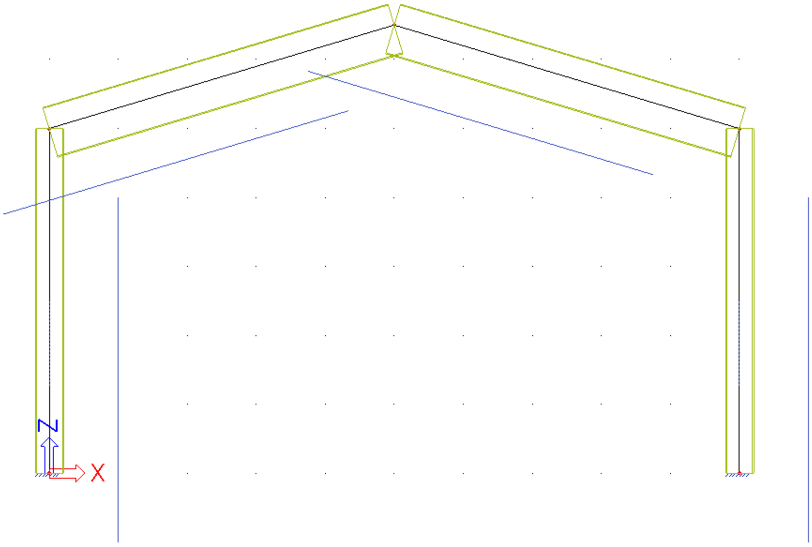


The additional data which was defined in *Step 2*, can now be inputted on the member. When double clicking on **Data for Flange Induced Buckling** the dialog with the properties of the data is displayed:

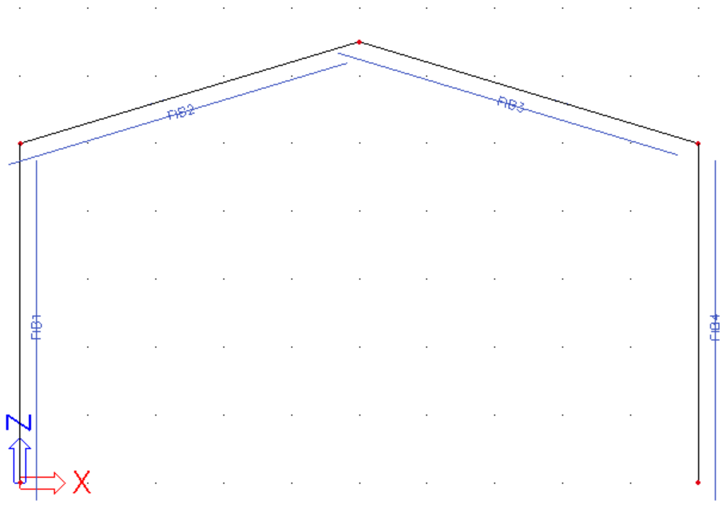


The **Parameters** group holds the user defined parameters of *Step 2.3* with their default values. In this case the combo-box 'Determination of factor k'.

The default values of the dialog are confirmed with [**OK**] and the data is inputted on all members.



Using the default Scia Engineer view parameters, the name of the additional data can be displayed.

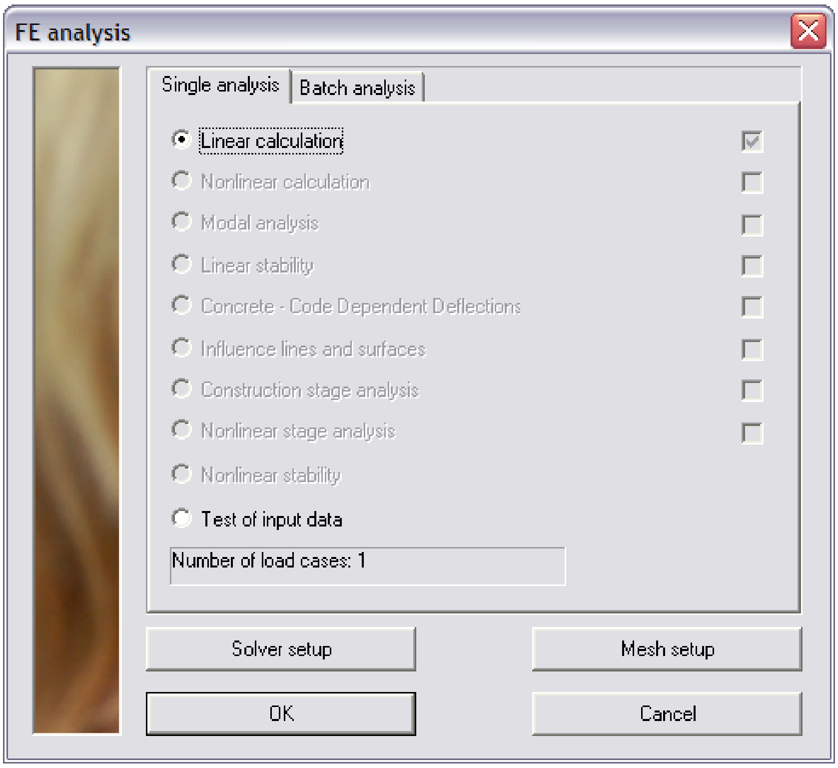


The data has now been inputted and in the next step the check can be executed.

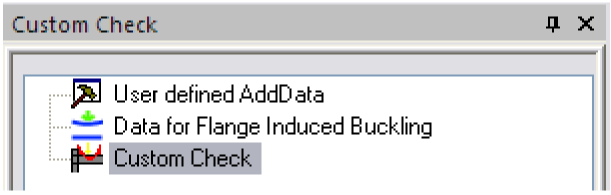
Step 4: Execute the Custom Check

In *Step 2* the additional data has been defined including the definition of the check, the mapping to Excel… In *Step 3* the additional data has been inputted. What is left is the execution of the check.

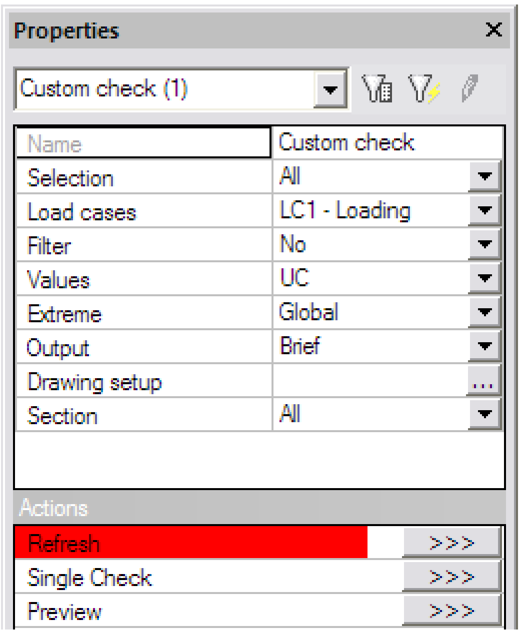
First of all the linear analysis is launched since internal forces will have to be sent to Excel.



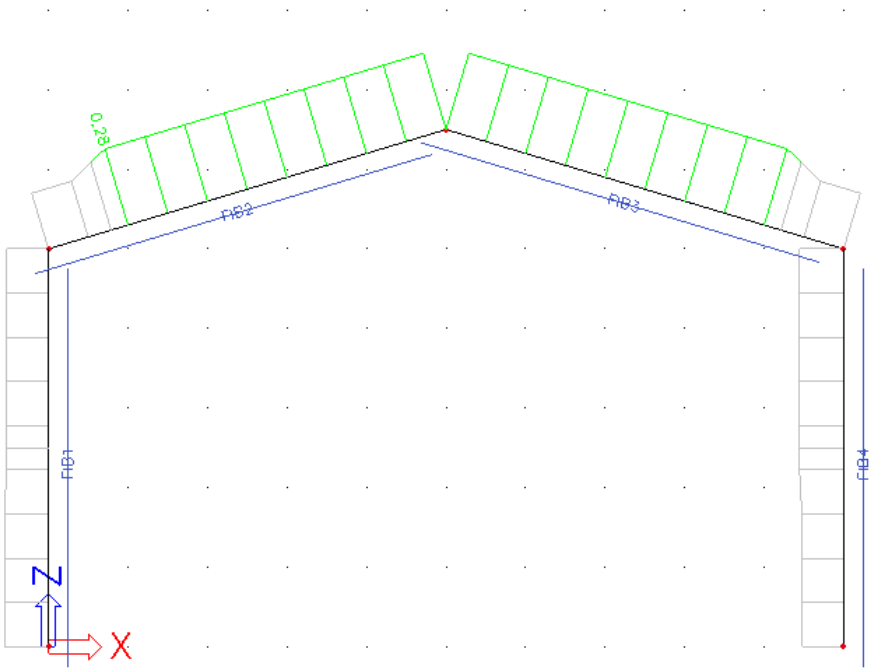
When user defined additional data was inputted and the analysis has been executed, the **Custom Check** service will show a new item: **Custom Check**.



The default check service property window accompanies this check:



The **Refresh** action button is pressed to execute the check. The following check result is shown on screen:

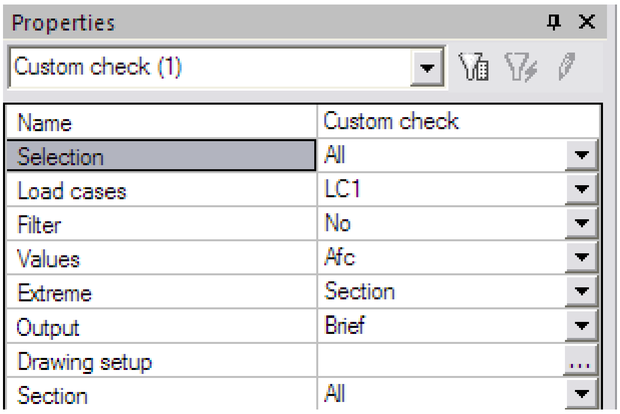


The asymmetric result for the columns is correct since different flanges are in compression on both column sides.

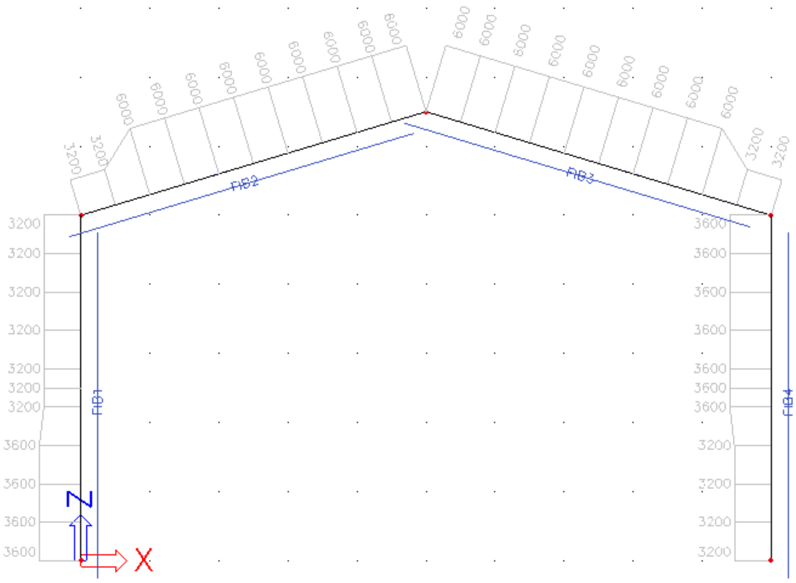
This can be checked by reviewing the compression flange area:

The **Values** field is changed to 'Afc'.

The **Extreme** field is changed to 'Section' to see the results in each section.

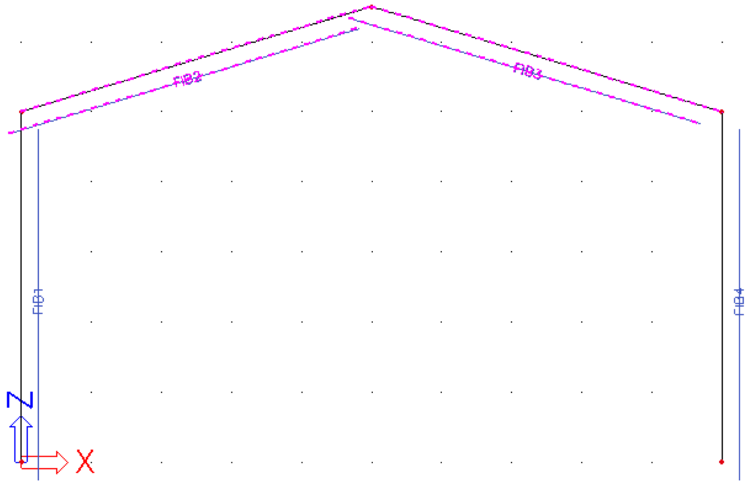


After pressing the **Refresh** action button the following result is shown on screen:

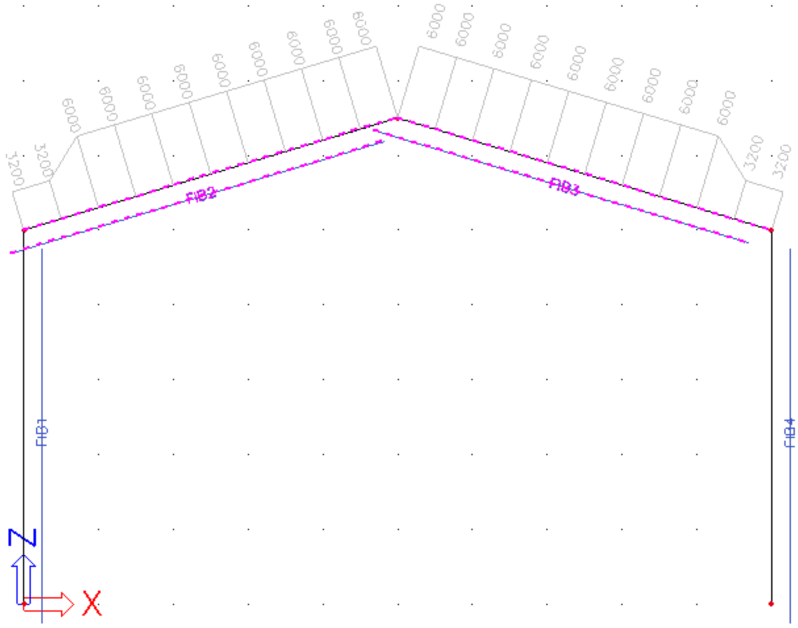


In order to see only the results for the rafters, the **Selection** field is set to ‘Current’. This implies a selection has to be made which in this case implies the additional data of the rafters and not the members themselves!

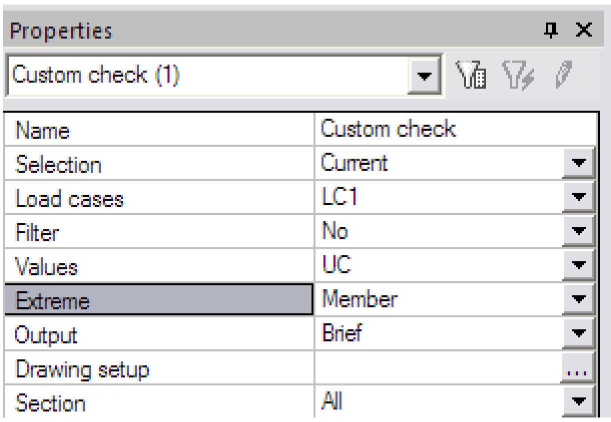
The external application check is based on user defined additional data. The check is performed for the members/nodes on which this user defined additional data has been defined. Therefore this additional data has to be selected and not the member or node as in other checks.



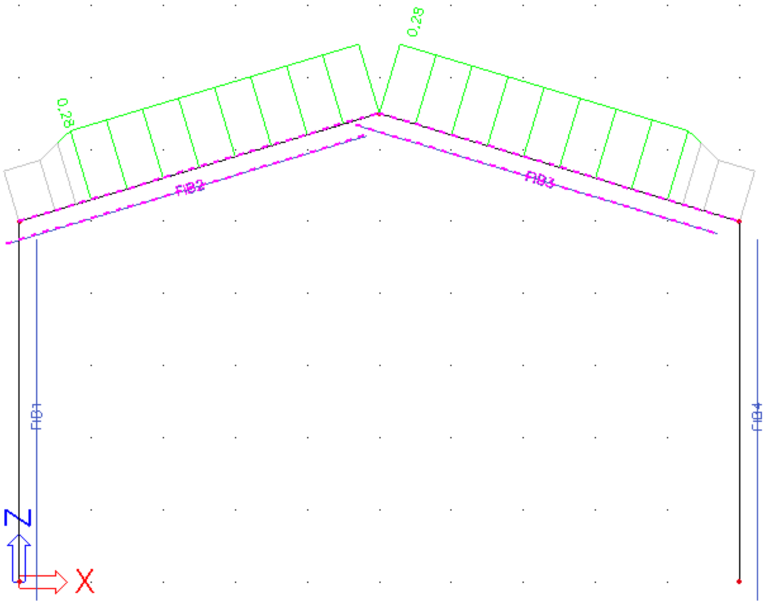
After pressing the **Refresh** action button the following result is shown on screen:



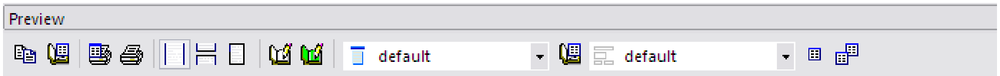
Next, the **Values** field is changed back to ‘UC’ and the **Extreme** field is set to ‘Member’.



After pressing the **Refresh** action button the following result is shown on screen:



When pressing the **Preview** action button, the Brief preview shows the following:

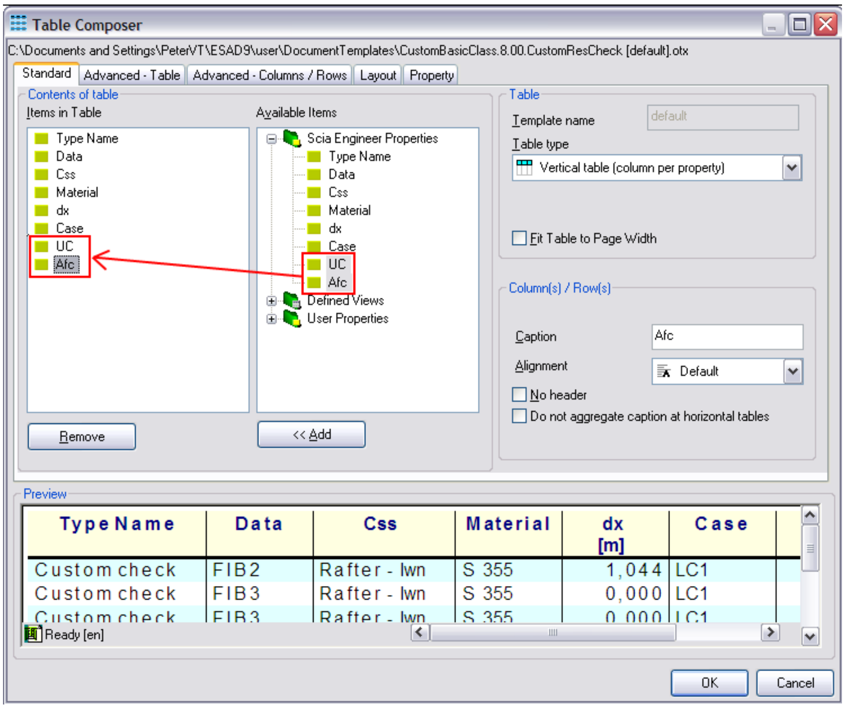


Custom check

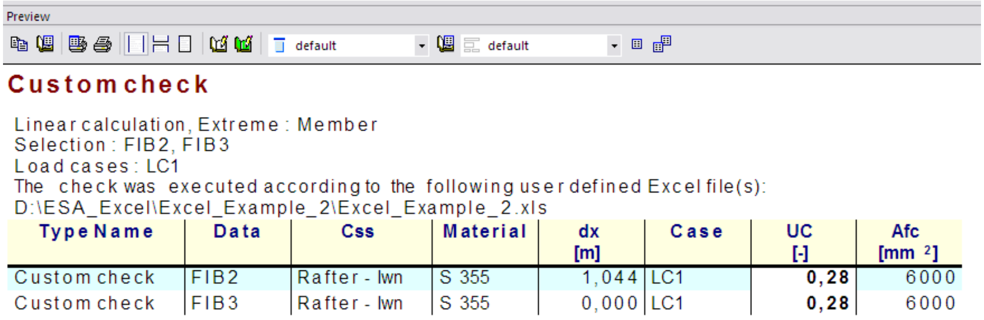
Linear calculation, Extreme : Member
Selection : FIB2, FIB3
Load cases : LC1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_2\Excel_Example_2.xls

Type Name	Data	Css	Material	dx [m]	Case
Custom check	FIB2	Rafter - Iwn	S 355	1,044	LC1
Custom check	FIB3	Rafter - Iwn	S 355	0,000	LC1

Using the tablecomposer, both the UC and Afc parameters can be added to the output:



When pressing the **Preview** action button, the Brief preview now shows the following:



Custom check

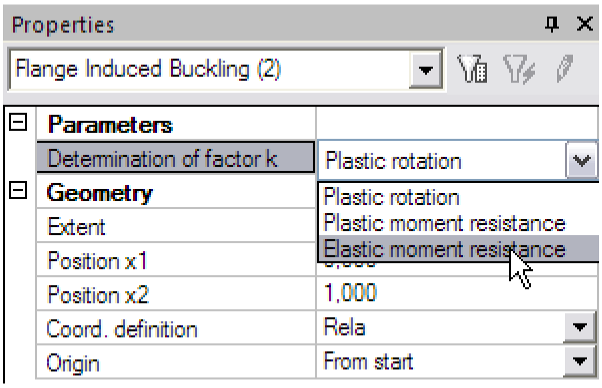
Linear calculation, Extreme : Member
Selection : FIB2, FIB3
Load cases : LC1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_2\Excel_Example_2.xls

Type Name	Data	Css	Material	dx [m]	Case	UC [-]	Afc [mm ²]
Custom check	FIB2	Rafter - Iwn	S 355	1,044	LC1	**0,28**	6000
Custom check	FIB3	Rafter - Iwn	S 355	0,000	LC1	**0,28**	6000

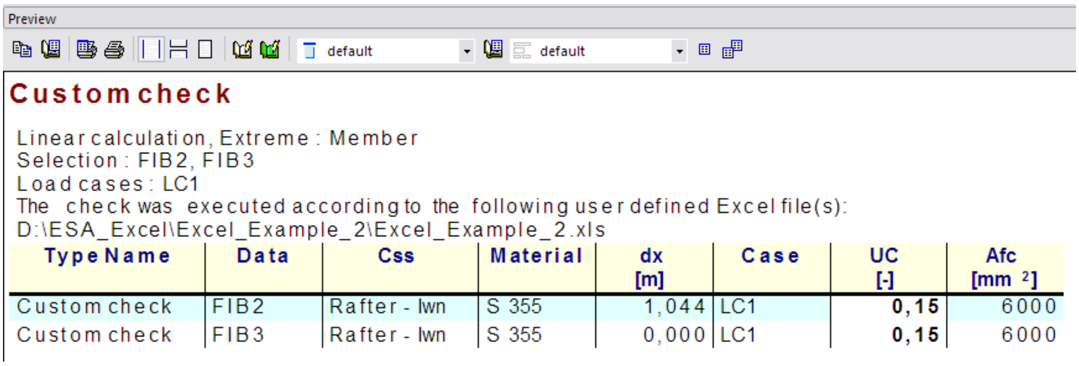
Note the Afc parameter which is shown with the unit defined in *step 2.7*.

The result was obtained by using the default setting for the factor k: 'Plastic rotation'.

Since both additional data on the beams are selected, their properties can be modified in the Property window. In this example, the **Determination of factor k** is changed to 'Elastic moment resistance'.



After pressing the **Refresh** action button, this less severe unity check is shown:

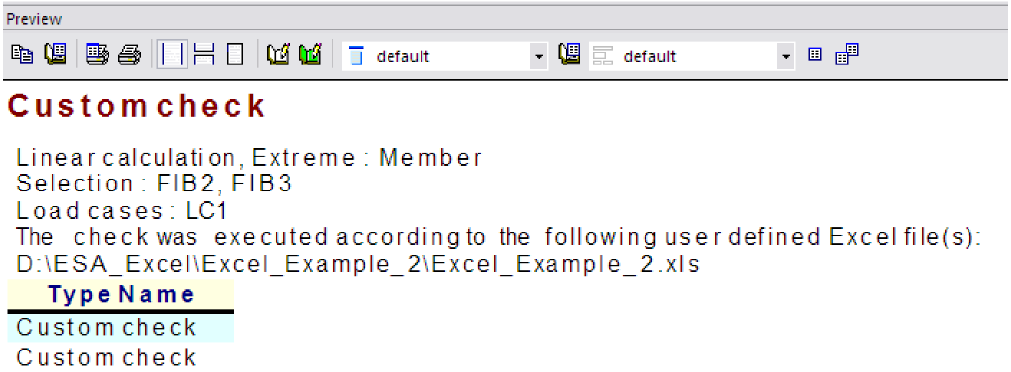


Custom check

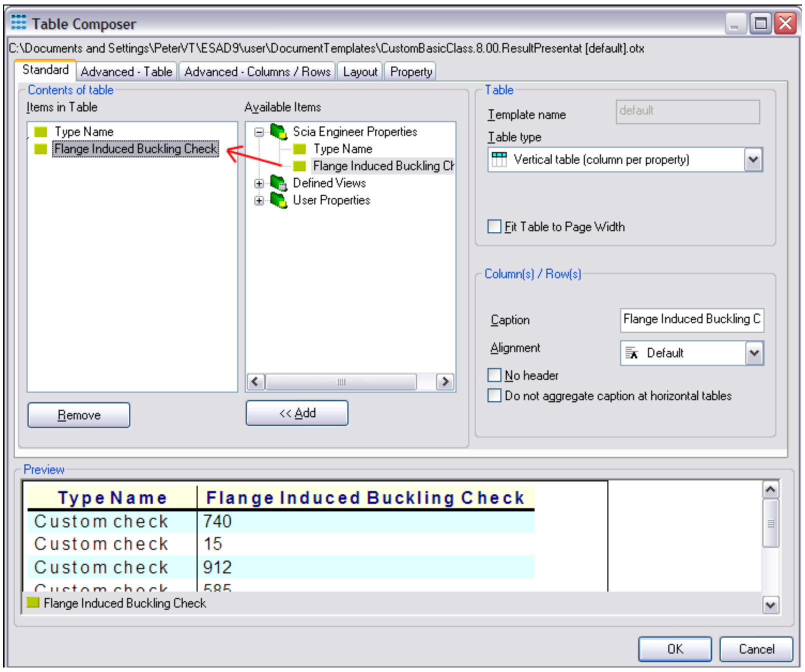
Linear calculation, Extreme : Member
Selection : FIB2, FIB3
Load cases : LC1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_2\Excel_Example_2.xls

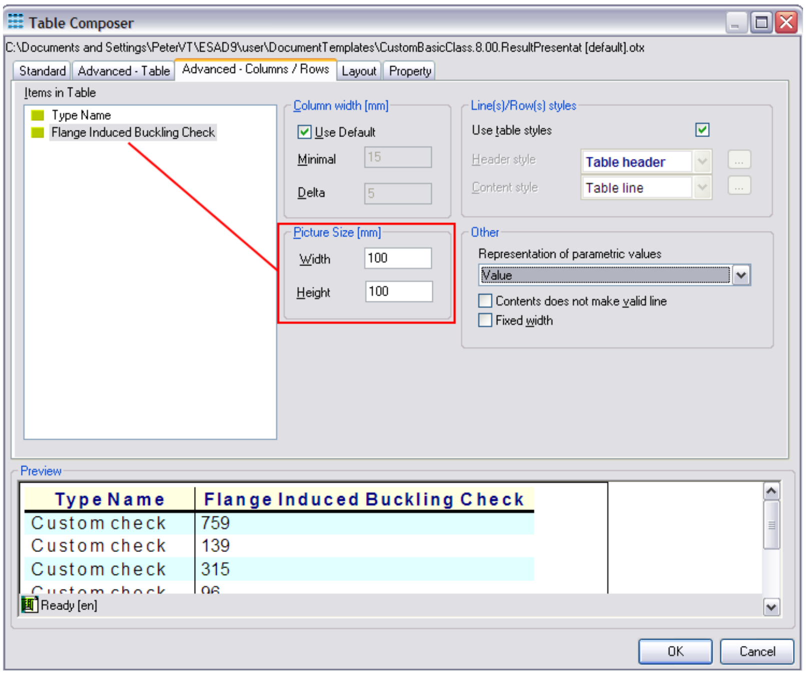
Type Name	Data	Css	Material	dx [m]	Case	UC [-]	Afc [mm ²]
Custom check	FIB2	Rafter - Iwn	S 355	1,044	LC1	**0,15**	6000
Custom check	FIB3	Rafter - Iwn	S 355	0,000	LC1	**0,15**	6000

Finally, the Detailed output is examined. The **Output** field is set to 'Detailed' and the **Refresh** action button is pressed.

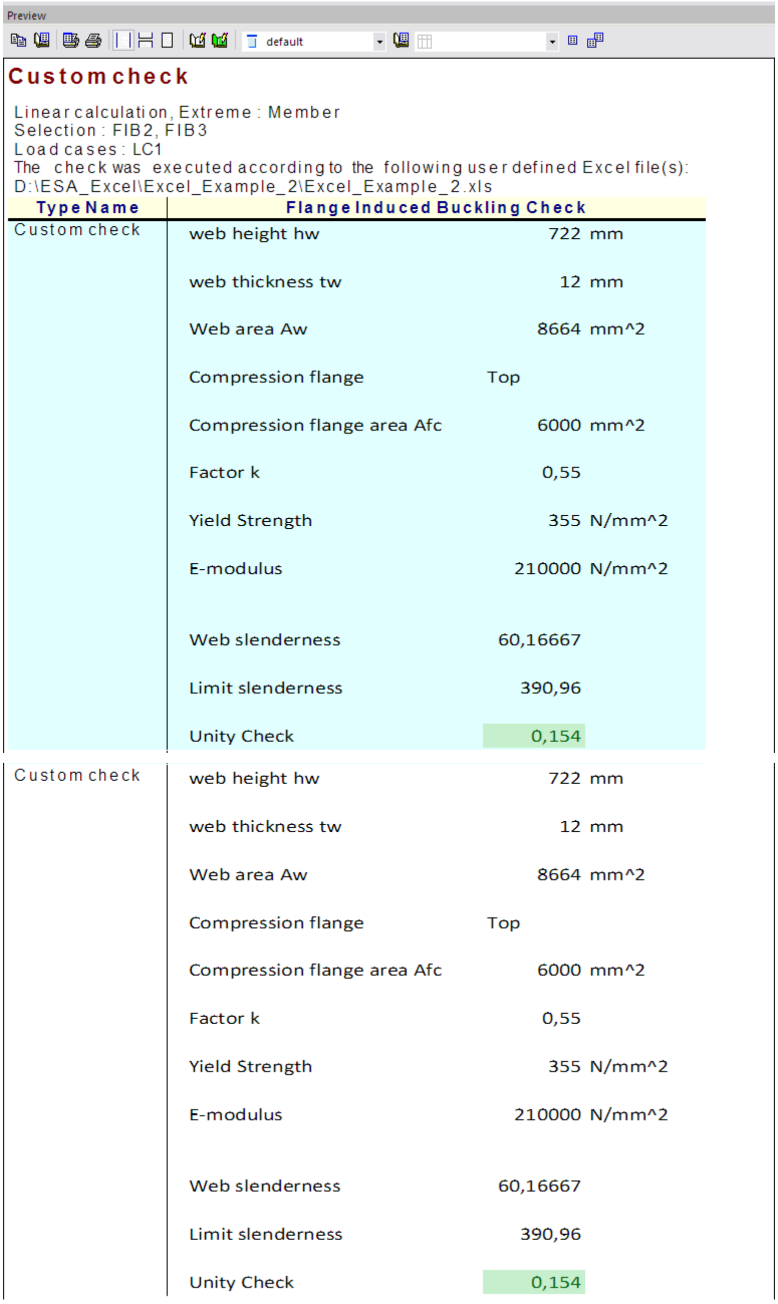


Through the table composer the Flange Induced Buckling Check item can be added and its picture size set to 100mm by 100mm:





This gives the following output after refreshing:


Preview

default

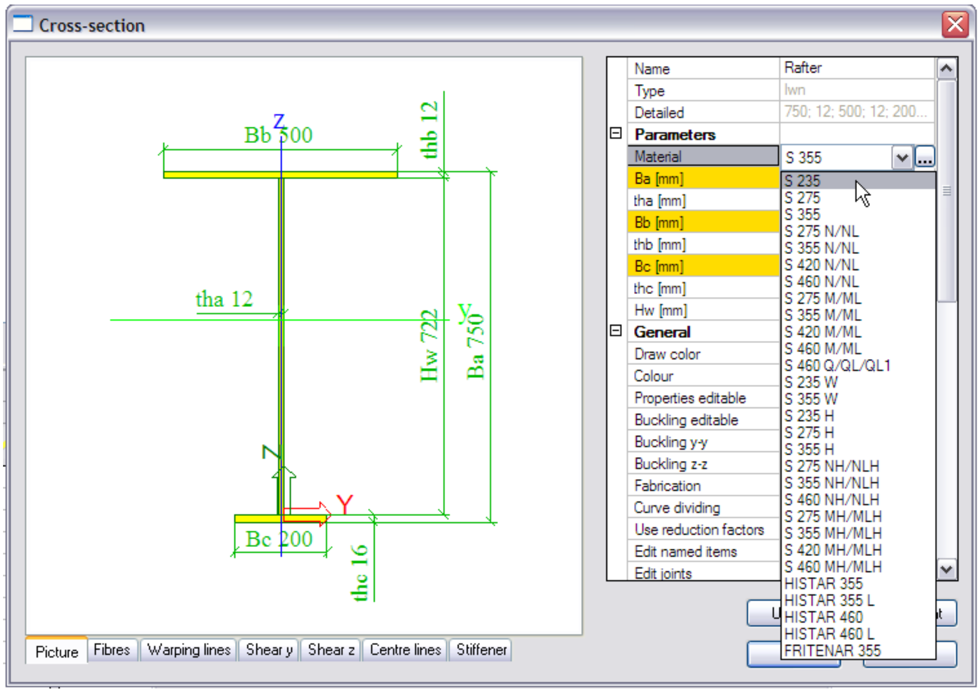
Custom check

Linear calculation, Extreme : Member
Selection : FIB2, FIB3
Load cases : LC1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_2\Excel_Example_2.xls

Type Name	Flange Induced Buckling Check		
Custom check	web height hw	722	mm
	web thickness tw	12	mm
	Web area Aw	8664	mm^2
	Compression flange	Top	
	Compression flange area Afc	6000	mm^2
	Factor k	0,55	
	Yield Strength	355	N/mm^2
	E-modulus	210000	N/mm^2
	Web slenderness	60,16667	
	Limit slenderness	390,96	
	Unity Check	0,154	
Custom check	web height hw	722	mm
	web thickness tw	12	mm
	Web area Aw	8664	mm^2
	Compression flange	Top	
	Compression flange area Afc	6000	mm^2
	Factor k	0,55	
	Yield Strength	355	N/mm^2
	E-modulus	210000	N/mm^2
	Web slenderness	60,16667	
	Limit slenderness	390,96	
	Unity Check	0,154	

Two outputs are given since the extreme per member was asked (with two entities selected). The Excel file could be modified to show also the name of the member and the section position on the output.

To finalize this example, the material for the rafter members is changed from **S355** to **S235**:



After recalculating, the Detailed output for the rafter members is refreshed:

Preview

default default

Custom check

Linear calculation, Extreme : Member
Selection : FIB2, FIB3
Load cases : LC1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_2\Excel_Example_2.xls

Type Name	Flange Induced Buckling Check		
Custom check	web height hw	722	mm
	web thickness tw	12	mm
	Web area Aw	8664	mm^2
	Compression flange	Top	
	Compression flange area Afc	6000	mm^2
	Factor k	0,55	
	Yield Strength	235	N/mm^2
	E-modulus	210000	N/mm^2
	Web slenderness	60,16667	
	Limit slenderness	590,61	
	Unity Check	0,102	

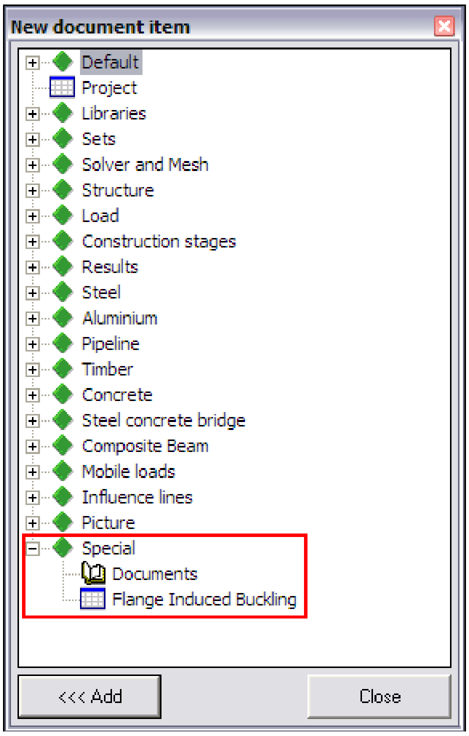
Custom check	web height hw	722	mm
	web thickness tw	12	mm
	Web area Aw	8664	mm^2
	Compression flange	Top	
	Compression flange area Afc	6000	mm^2
	Factor k	0,55	
	Yield Strength	235	N/mm^2
	E-modulus	210000	N/mm^2
	Web slenderness	60,16667	
	Limit slenderness	590,61	
	Unity Check	0,102	

The output clearly shows that the change of material is correctly taken into account.

The check has now been executed and reviewed. To end this step, the document of Scia Engineer is examined.

In the document, the inputted User defined additional data can be inserted into the document in the same way as any other default additional data.

In the **New document item** dialog, the **Special** chapter holds the tables for all user defined additional data.



In this example **Flange Induced Buckling** was defined and thus this data can be added into the document.

Project	Excel_Example_2
Part	-
Description	Example 2 for Tutorial Excel Link
Author	PVT

1. Flange Induced Buckling

Type Name	Name	Member	Extent	Pos x_1	Pos x_2	Coor	Orig	Determination of factor k
Flange Induced Buckling	FIB1	B1	full	0,000	1,000	Rela	From start	Plastic rotation
Flange Induced Buckling	FIB2	B2	full	0,000	1,000	Rela	From start	Elastic moment resistance
Flange Induced Buckling	FIB3	B3	full	0,000	1,000	Rela	From start	Elastic moment resistance
Flange Induced Buckling	FIB4	B4	full	0,000	1,000	Rela	From start	Plastic rotation

The table shows the different properties of the additional data, including the user defined parameter 'Determination of factor k'. In the same way as for any other default additional data of Scia Engineer this table can be edited and modified through the Table Composer.

Through the Active Document feature of Scia Engineer, the user define properties can also be edited inside the document.

Step 5: Save the User Defined Additional Data into a database for future use

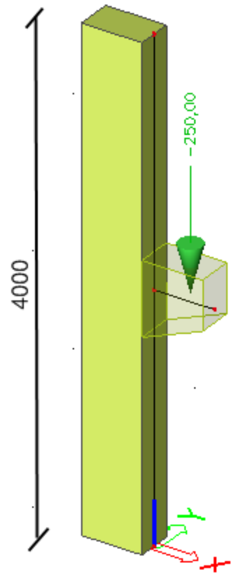
If required, this additional data can be saved into a database for future use as illustrated in Example 1.

Example 3: Corbel Design

In this third example, the mapping of arrays is explained. In addition, multiple detailed outputs are used.

As a practical case, concrete Corbel Design is used.

In this example, a column with corbel is modelled. The column has a height of **4m** and the corbel is attached in the middle. The column has a rectangular section of **500mm** by **300mm**. The corbel has a width of **300mm** and the height varies from **600mm** to **400mm**. The corbel has a length of **0,5m**. The column base is modelled as fully fixed. All members are manufactured in **C30/37** according to **EC-EN**.

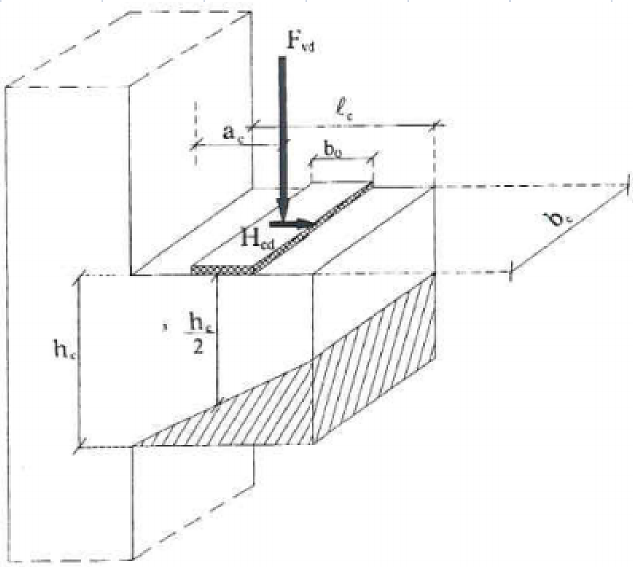


One load case is defined, a point load of **250 kN** acting as design load on the corbel. This load is applied at the mid-length position of the corbel.

The check will be done according to the Excel file "**Excel_Example_3.xls**"

The Excel file contains two worksheets. On the sheet 'Input' the input data are set:

	A	B	C
1	**Input Data from Scia Engineer**		
2			
3	Haunch height hc	0,4	m
4	Haunch width bc	0,25	m
5			
6	Load position ac	0,2	m
7			
8	Load Fvd	150000	N
9			
10	Concrete strength fck	30000000	N/m^2
11			
12	Bearing width b0	0,12	m
13			
14	Gamma M	1,5	-
15			
16	Reinforcement diameter	18	mm
17			
18			
19	**Calculated Input data**		
20			
21	Load Hcd (=0,20 * Fvd)	30000	N



The sheet ‘Check’ shows the check and reinforcement design:

	A	B	C	D
1	**Application of strut and tie model**			
2	0,4 hc <= ac <= hc	OK		
3				
4				
5	**Check of strut**			
6				
7	Ncd	353,55	kN	
8				
9	A0	45000	mm^2	
10				
11	Sigma cd	11,11111111	N/mm^2	
12				
13	Nu*fcd	14	N/mm^2	
14				
15	Check	0,79		
16				
17				
18	**Design of tension reinforcement**			
19				
20	Cover c	25	mm	
21				
22	d	465	mm	
23				
24	As required	578,71	mm^2	
25				
26	As to provide	2	diam.	20
27		628	mm^2	

In addition to the default properties of Scia Engineer, the following user defined parameters will have to be defined: the bearing width b0, the safety factor Gamma M and the reinforcement diameter.

Using the input data, the Excel file generates two types of output.

First the compression strut is being checked and second the required tension reinforcement is determined.

In the Excel file, all cells to which data has to be mapped and from which data is read have been given a name. This allows for a very easy definition of the mapping since these same names will be available in the mapping dialog of Scia Engineer.

The calculation is a simplification based on the book “Reinforced Concrete, Design following NBN B15-002 (1999), Academia Press, 2001.”

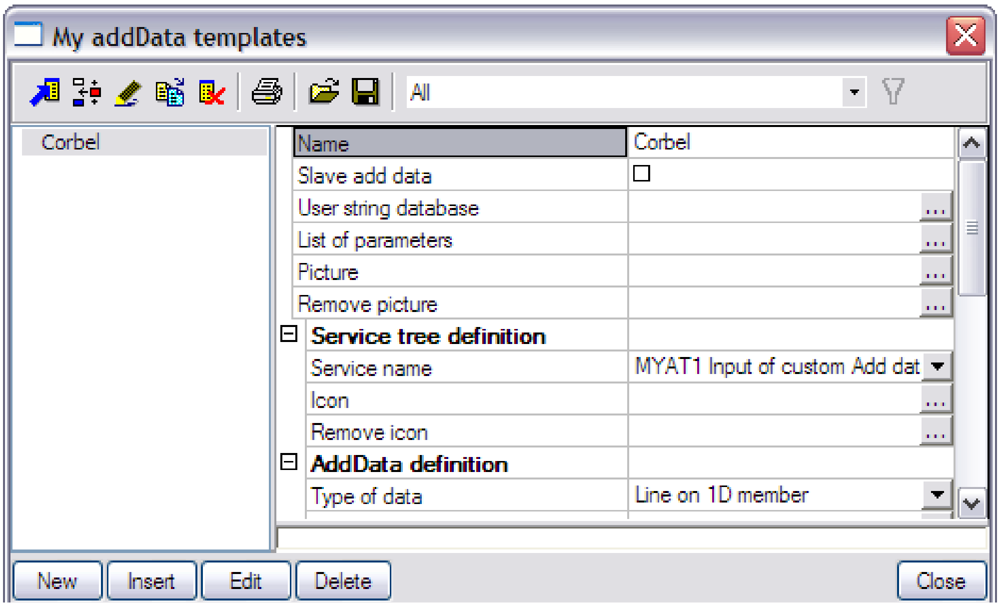
Step 1: Activate the functionality External Application Checks

The first step is to activate the functionality **External application checks** on the **Functionality** tab in the **Project Data**.

Step 2: Create User Defined Additional Data

In the second step, User Defined Additional Data will be defined.

Through **Tools** > **User defined AddData** the **User Defined Additional Data** Library can be opened.



The **Name** of the additional data is changed to 'Corbel'.

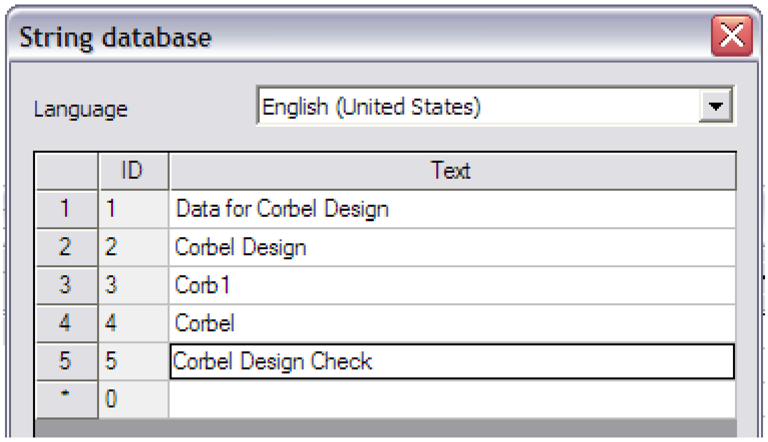
Step 2.1 Slave data

Only one type of additional data will be defined here and as such the check-box **Slave add data** is left unchecked.

Step 2.2 Define text strings

In the **User string database** the required strings are defined for the definition of the additional data. Since in this example Corbel Design is being illustrated the strings are modified as follows:

Type for which the string is used	Default string	String used in this example
Service name	MYAT1 Input of custom add data	Data for Corbel Design
Type name	MYAT1 Custom defined add data	Corbel Design
Short name	MYAT1 MADI	Corb1
Description	MYAT1 Description	Corbel
Name of check	MYAT1 Custom check	Corbel Design Check

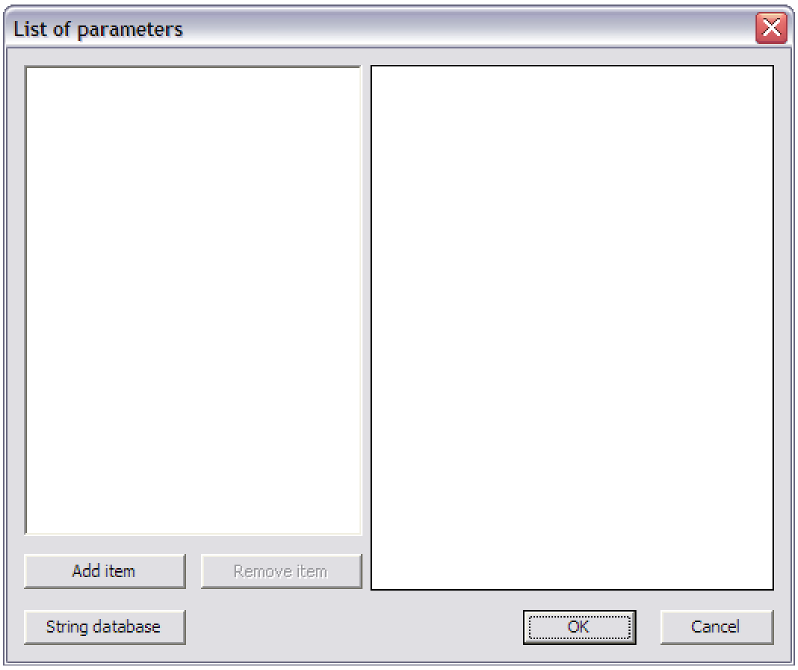


The necessary strings for the definition of the data are inputted and in the next step the parameters can be defined.

Step 2.3 Define parameters

In this example, the mapping will concern of default Scia Engineer data (Cross-section dimensions, material properties and loading properties) as well as user defined parameters.

The user defined parameters can be defined through **List of parameters**.



First, two numerical parameters will be defined, the bearing width and the safety factor:

Parameter	Type	Default value
Bearing width b0	Number	100 mm
Safety factor Gamma M	Number	1,5

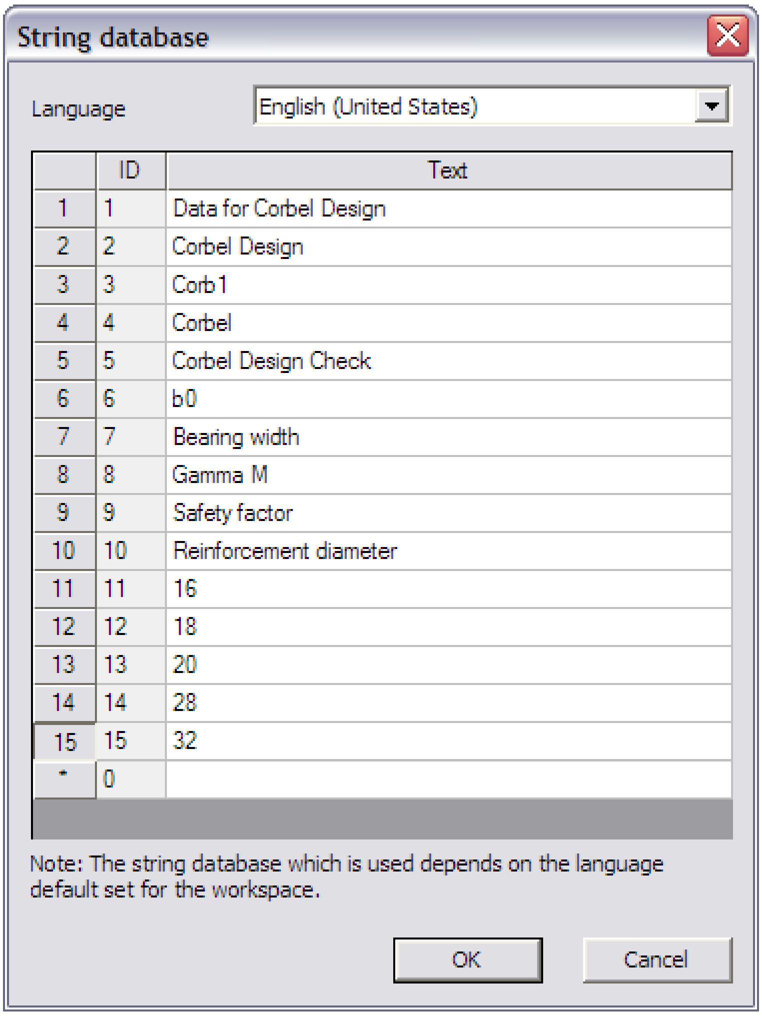
In addition, a combo-box will be defined from which the user can select the reinforcement diameter:

Parameter	Type	Combo-box lines
Reinforcement diameter	Combo-box	16 18 20 28 32

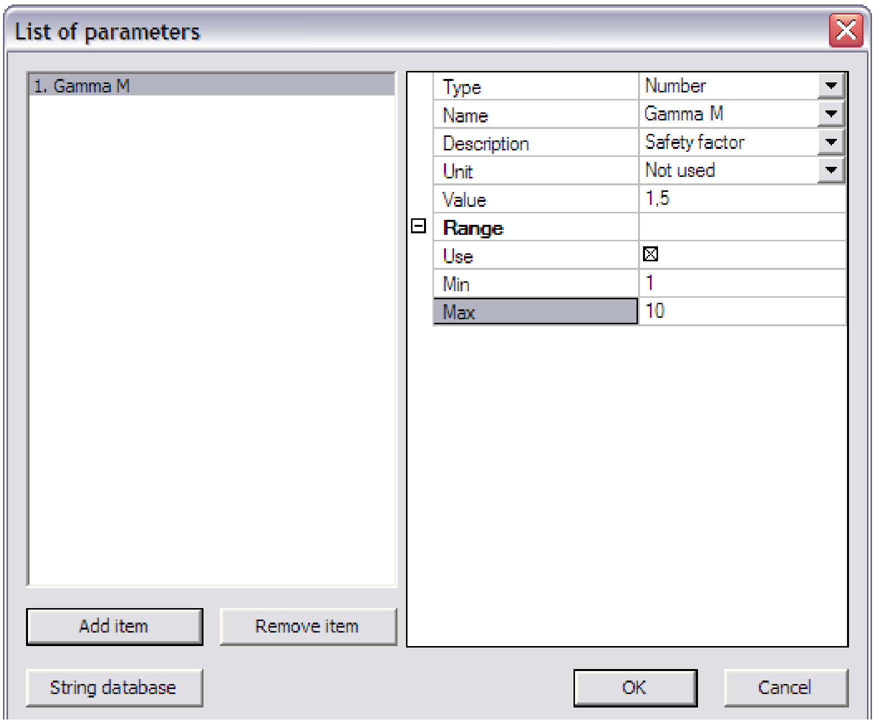
Through the button **String database** the text string database can be directly accessed. This allows a quick input of the strings required for the parameters.

For this example the following strings are added:

Strings used in this example
b0
Bearing width
Gamma M
Safety factor
Reinforcement diameter
16
18
20
28
32



Through the button **Add item** the first parameter, the safety factor Gamma M, is added.



The **Type** field is set to ‘Number’.

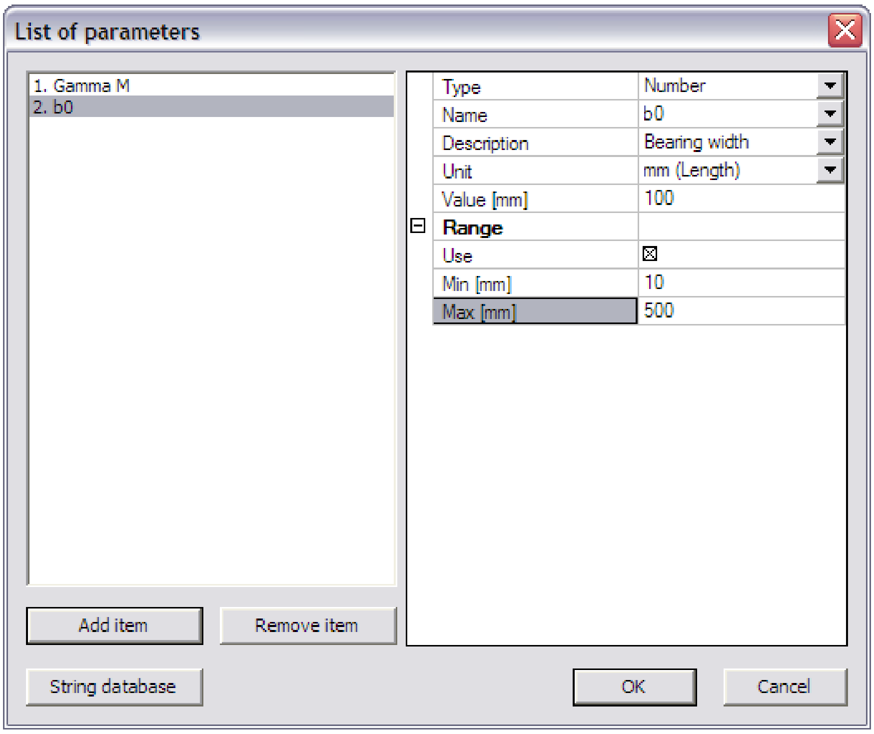
In the **Name** and **Description** fields the respective strings can be chosen from the string database, in this case ‘Gamma M’ and ‘Safety factor’.

For this parameter no **Unit** is used.

The default **Value** of the parameter is set to ‘1,5’.

In addition, a **Range** is set to make sure the input is only allowed between a minimum of ‘1’ and a maximum of ‘10’.

In exactly the same way using the button **Add item** the second parameter, the Bearing width b0, is added.



The **Type** field is set to ‘Number’.

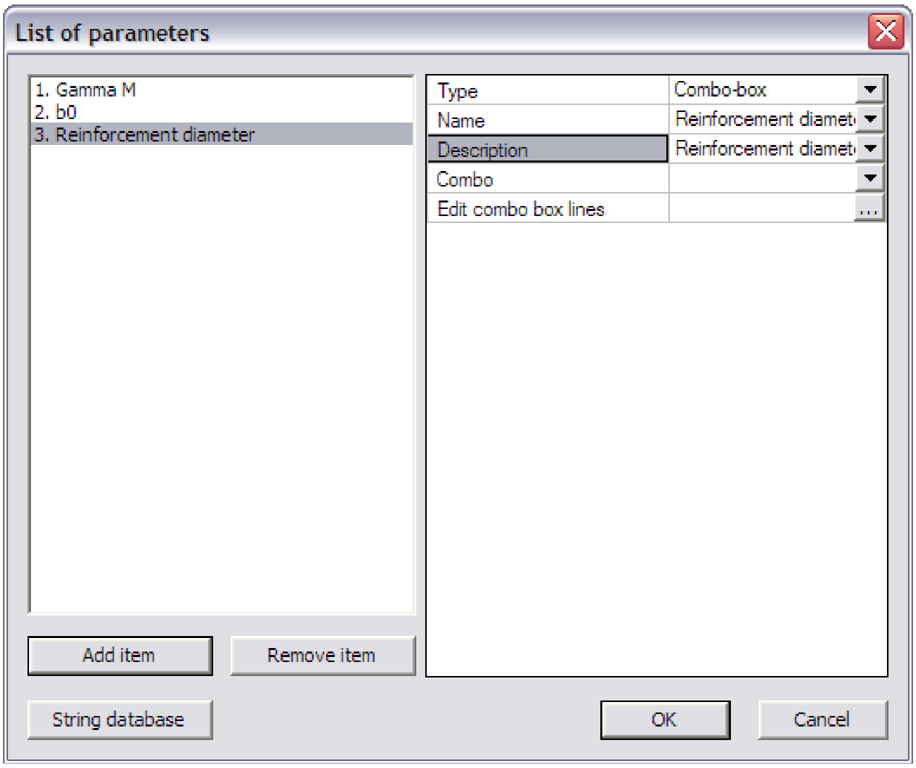
In the **Name** and **Description** fields the respective strings can be chosen from the string database, in this case ‘b0’ and ‘Bearing width’.

For this parameter the **Unit** is set to ‘mm (Length)’.

The default **Value** of the parameter is set to ‘100’ mm.

In addition, a **Range** is set to make sure the input is only allowed between a minimum of ‘10’ mm and a maximum of ‘500’ mm.

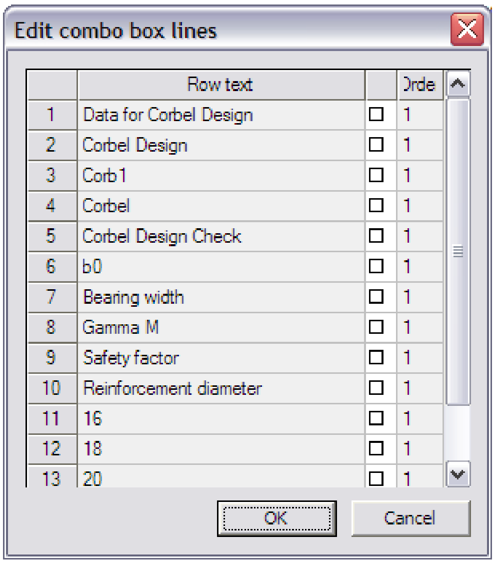
Using the button **Add item** the final parameter, the Reinforcement diameter, is added.



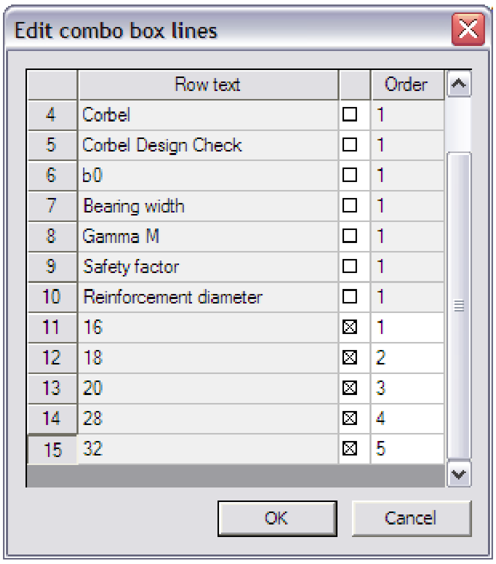
The **Type** field is set to ‘Combo-box’.

For both the **Name** and **Description** fields the string ‘Reinforcement diameter’ is set.

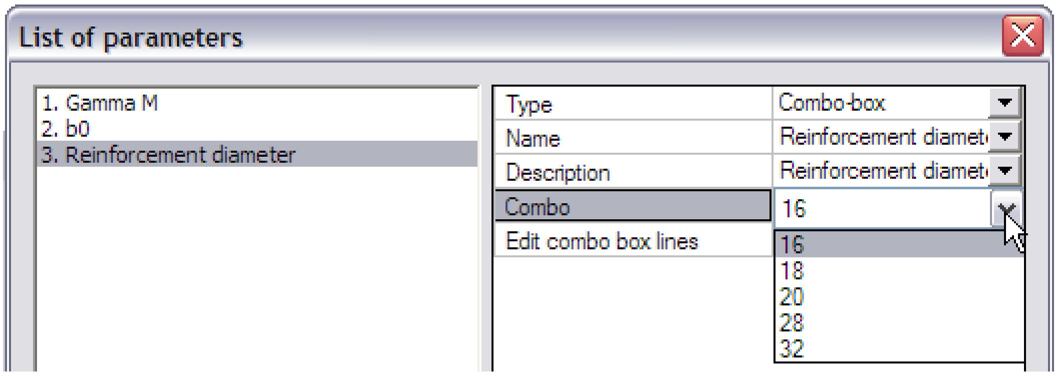
Next, the lines in the combo-box are defined through the edit button **Edit combo box lines**.



The diameters inputted in the string database are selected and in the Order column the numbers ‘1’, ‘2’, ‘3’, ‘4’ and ‘5’ are inputted.



When closing this dialog, the **Combo** item in the **List of Parameters** dialog shows how the combo-box will look like.

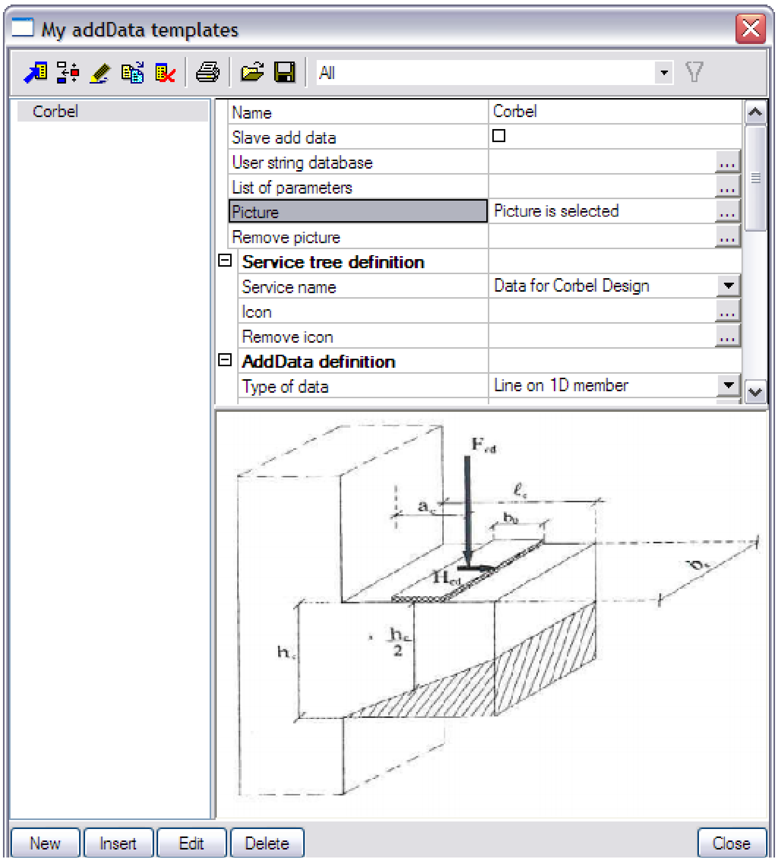


The required parameters are now defined and the dialog can be closed.

Step 2.4 Add a picture to the Additional Data

To clarify the use of the additional data and the defined parameters a picture can be added using the **Picture** button.

In this example the picture **Excel_Example_3_Picture.bmp** will be used.



Step 2.5 Define Service Tree

In the next step the Service Tree is defined through the group **Service tree definition**.

The **Service name** is taken automatically from the text string database.

To clarify the Service name, an icon can be added using the **Icon** button.
In this example the icon **Excel_Example_3_Icon.bmp** will be used.

Service tree definition	
Service name	Data for Corbel Design
Icon	Icon is selected
Remove icon	

Step 2.6 Define the Additional Data

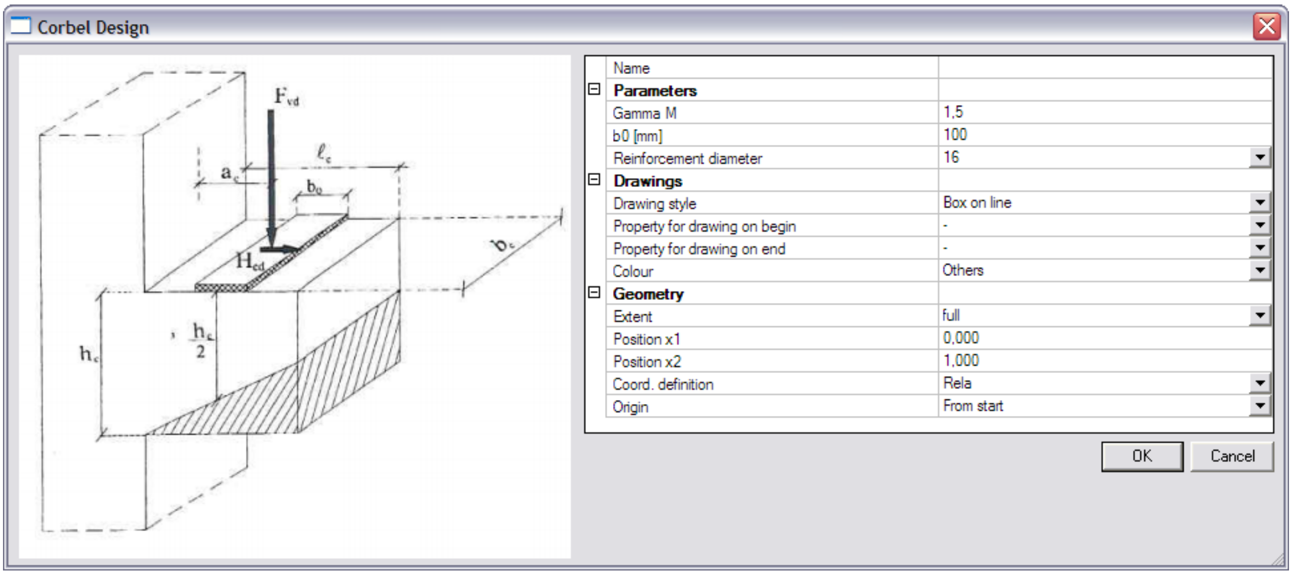
Using the data from the previous steps, the additional data can now be defined in the group **AddData definition**.

AddData definition	
Type of data	Line on 1D member
Instance setup	
Type name	Corbel Design
Short name	Corb1
Description	Corbel

Since the Corbel Design concerns the ‘full’ corbel, the check will be executed on the full length of the corbel. Therefore the field **Type of data** is set to ‘Line on 1D member’.

The **Type name**, **Short name** and **Description** are taken automatically from the text string database.

To get an overview of all the data entered in the previous steps the button **Instance Setup** is used.



The **Parameters** group shows the user defined parameters specified in *Step 2.3*.

In the **Drawings** group, the **Drawing style** is set to 'Box on line'. For the **Colour field** 'Thermal load' is chosen.

Drawings	
Drawing style	Box on line
Property for drawing on begin	b0
Property for drawing on end	b0
Colour	Thermal load

In addition, the Bearing width 'b0' is set as **Property for drawing on begin/end**. The Drawing style will thus change in function of the value of b0.

With this the additional data itself is defined and the dialog can be closed.

Step 2.7 Define the Check

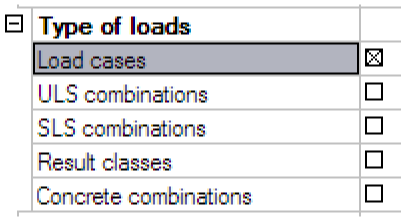
In the group **Check data** the necessary data for the check itself can now be defined.

Check data	
Name of check	Corbel Design Check
Setup for Brief output	...
Type of loads	
Load cases	☐
ULS combinations	☐
SLS combinations	☐
Result classes	☐
Concrete combinations	☐

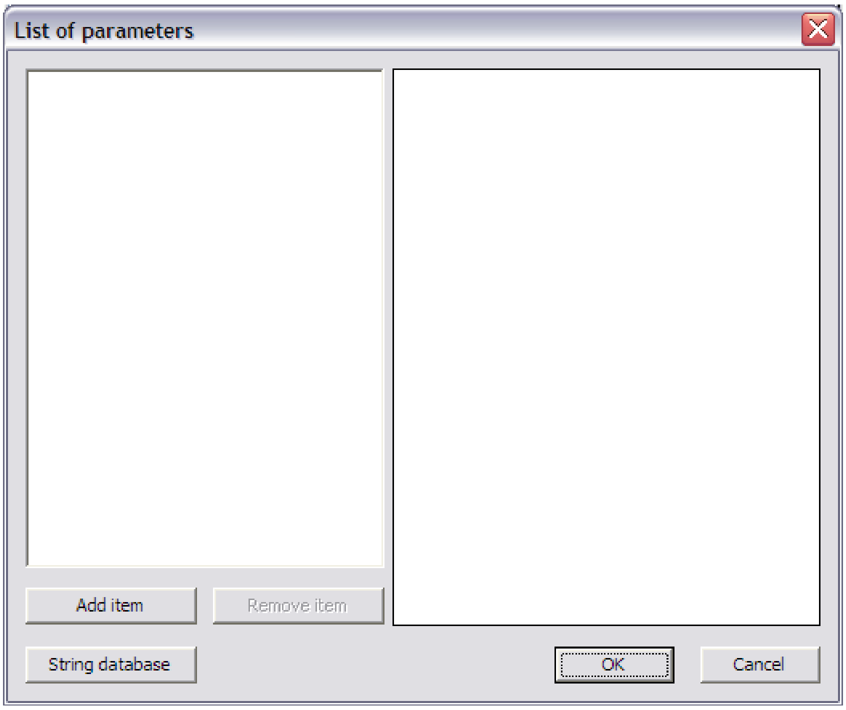
The **Name of check** is taken automatically from the text string database.

The **Type of loads** group allows to specify which load types will be available for the check. Only the selected items will be available when executing the check.

Since in this example only one load case was defined, the design loading for the corbel, only the option ‘Load cases’ will be activated.



The final item for defining the check is the **Setup for Brief output** where the output parameters have to be defined.

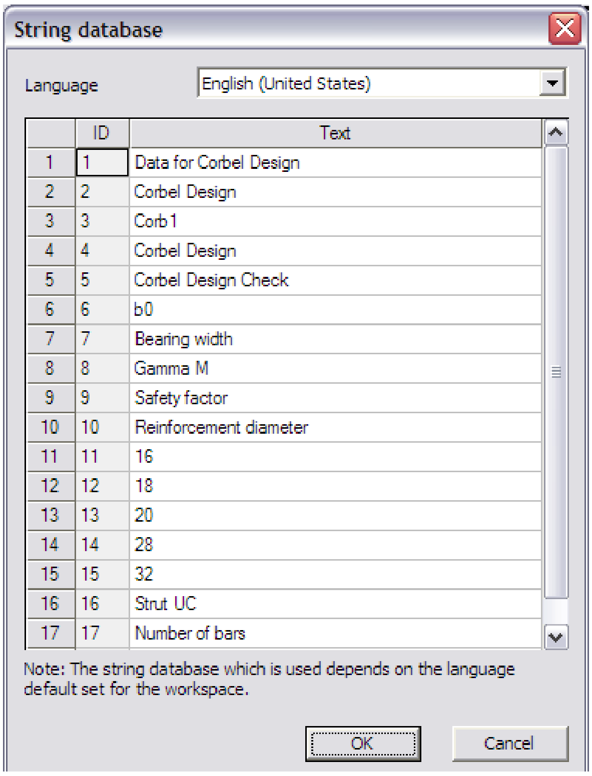


For this example, two output parameters will be defined: the unity check value for the strut and the number of required reinforcement bars.

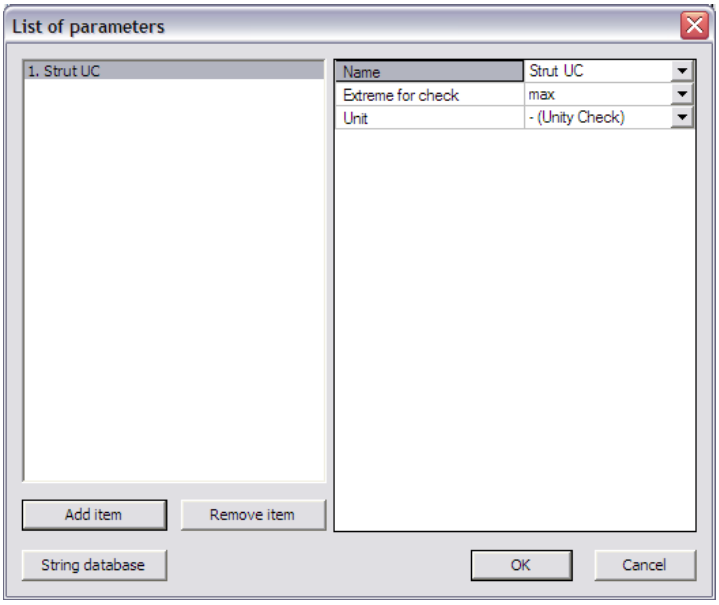
Parameter	Unit
Strut UC	-
Number of bars	Not used

First of all, through the button **String database** the text string database is accessed to define the required strings. For this example the following strings are added:

Strings used in this example
Strut UC
Number of bars



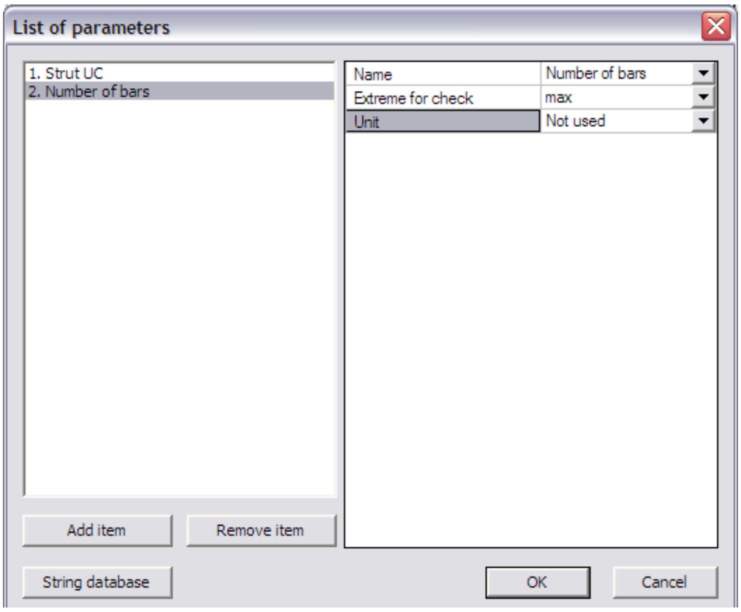
When the strings are defined, the first parameter is added through the button **Add item**.



In the **Name** field the ‘Strut UC’ string is chosen from the string database.
The **Extreme for check** is left on ‘max’ since the maximal unity check value is extreme in this case.

Since it concerns a unity check, the **Unit** field is left on ‘– (Unity Check)’.

Again using the button **Add item** the second parameter is added.



In the **Name** field the 'Number of bars' string is chosen from the string database.

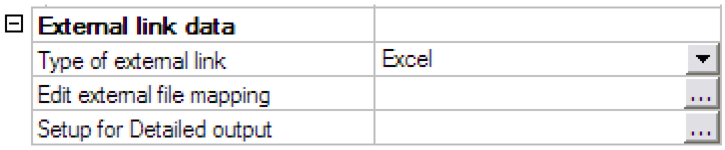
The **Extreme for check** is left on 'max'.

For this parameter no unit is required so the **Unit** field is set to 'Not used'.

The check and output parameters have now been defined so in the next step the link can be set.

Step 2.8 Specify the type of external link

In the **External link data** group the **Type of external link** allows to specify which external application will be used.



In this example the link is made with Excel and thus 'Excel' is chosen.

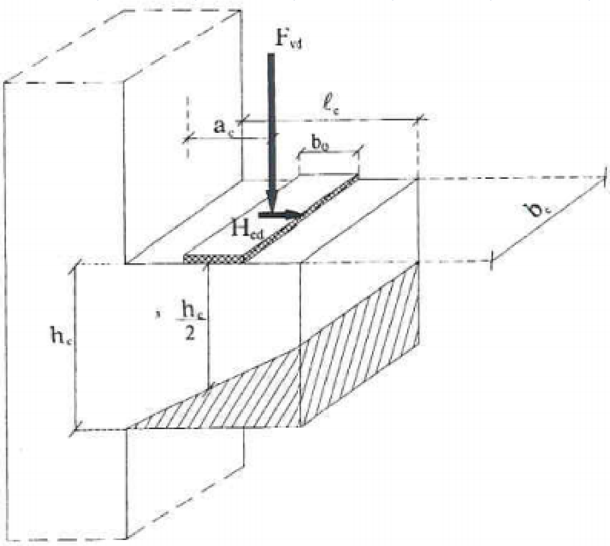
Step 2.9 Define the mapping with the external application

All preparation has now been done, what remains is the most important step of the process: defining the actual mapping between properties and parameters of Scia Engineer and the data fields (i.e. Excel cells) of the external application.

Through the button **Edit external file mapping** the mapping dialog is opened.

In this example, the Excel file contains two worksheets. On the sheet ‘Input’ the input data from Scia Engineer are set:

	A	B	C
1	**Input Data from Scia Engineer**		
2			
3	Haunch height hc	0,4	m
4	Haunch width bc	0,25	m
5			
6	Load position ac	0,2	m
7			
8	Load Fvd	150000	N
9			
10	Concrete strength fck	30000000	N/m^2
11			
12	Bearing width b0	0,12	m
13			
14	Gamma M	1,5	-
15			
16	Reinforcement diameter	18	mm
17			
18			
19	**Calculated Input data**		
20			
21	Load Hcd (=0,20 * Fvd)	30000	N



The sheet ‘Check’ shows the check and reinforcement design:

	A	B	C	D
1	**Application of strut and tie model**			
2	0,4 hc <= ac <= hc	OK		
3				
4				
5	**Check of strut**			
6				
7	Ncd	353,55	kN	
8				
9	A0	45000	mm^2	
10				
11	Sigma cd	11,11111111	N/mm^2	
12				
13	Nu*fcd	14	N/mm^2	
14				
15	Check	0,79		
16				
17				
18	**Design of tension reinforcement**			
19				
20	Cover c	25	mm	
21				
22	d	465	mm	
23				
24	As required	578,71	mm^2	
25				
26	As to provide	2	diam.	20
27		628	mm^2	

In this example, cross-section properties have to be sent to Excel. In the document, it can be seen how the dimensions of a rectangular concrete section are defined:

Param. length name	H
	B
Param. length value [mm]	600
	300

More specifically, the dimensions are located within the **Param. length value** array. This implies that, when this data is mapped to Excel, two cells will be filled since the array contains two items.
Array mapping concern a very convenient way to map multiple values. Only the starting cell and the direction of the array (horizontal or vertical) need to be specified.

As specified in the introduction of this example, in the Excel file, all cells to which data has to be mapped and from which data is read have been given a name. These named cells can now be used in the **Cell address** field instead of manually typing the cell number.

The following table shows which properties should be mapped to which cells:

Object	Property	Worksheet	Named Cell (Address)
Cross-Sections	Param. length value (Parameters)	Input	Cross-Section_Parameters (B3)
Point force on beam	Position x (Geometry)	Input	Load_position_ac (B6)
Point force on beam	Value - F	Input	Load_Fvd (B8)
Concrete EC2	Characteristic compressive cylinder strength [28] (Fck) (EC2)	Input	Concrete_strength_fck (B10)
<<< My input parameters	b0	Input	Bearing_width_b0 (B12)
<<< My input parameters	Gamma M	Input	Gamma_M (B14)
<<< My input parameters	Reinforcement diameter	Input	Reinforcement_diameter (B16)
>>> My output parameters	Strut UC	Check	Strut_UC (B15)
>>> My output parameters	Number of bars	Check	Number_of_bars (B26)

> ***The Object 'Concrete EC2' exists twice, once for EC-ENV and once for EC-EN. The Property 'Characteristic compressive cylinder strength [28] (Fck) (EC2)' can be found in the EC-EN Object.***
>
> ***For the position of the loading, the property Position x is used. It is important to note that this position is dependent on the coordinate definition of the point load: this can either be absolute or relative. For this example, the Position x has been inputted as absolute. The Excel file could be modified with a test to see if the coordinate definition is absolute or relative and modify the Position x value accordingly.***

The mapping of the first property, the cross-section dimensions, is thus done as follows:

The **Object** field is set to 'Cross-sections'.

In the **Property** field 'Param. length value (Parameters)' can then be chosen.

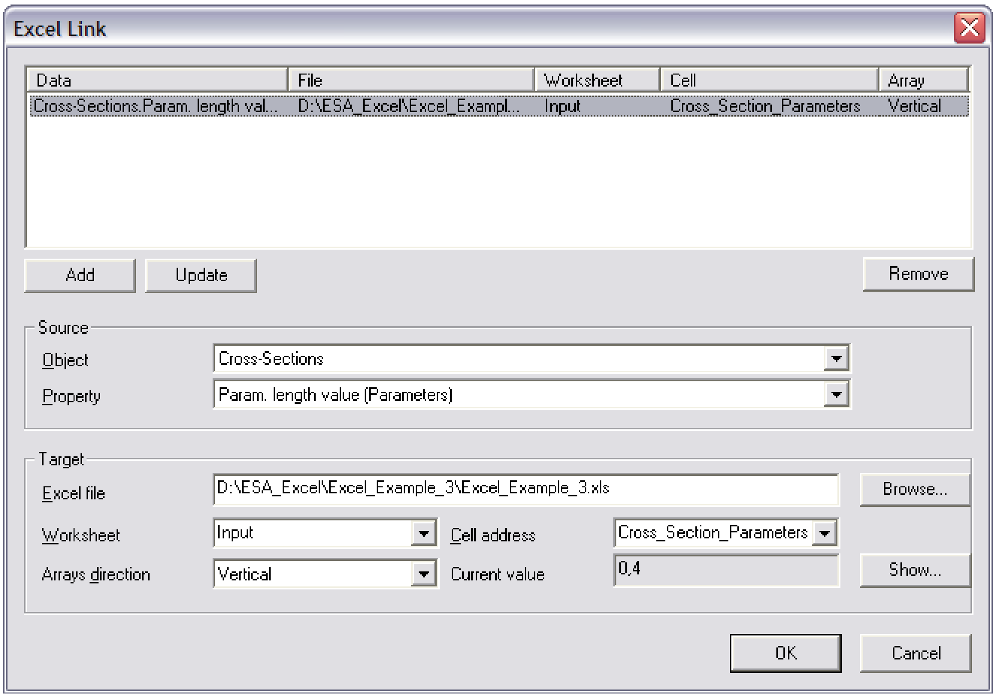
Using the **Browse** button, the file **Excel_Example_3.xls** is searched.

After the file has been specified, the **Worksheet** field contains a list of all sheets. This field is set to ‘Input’.

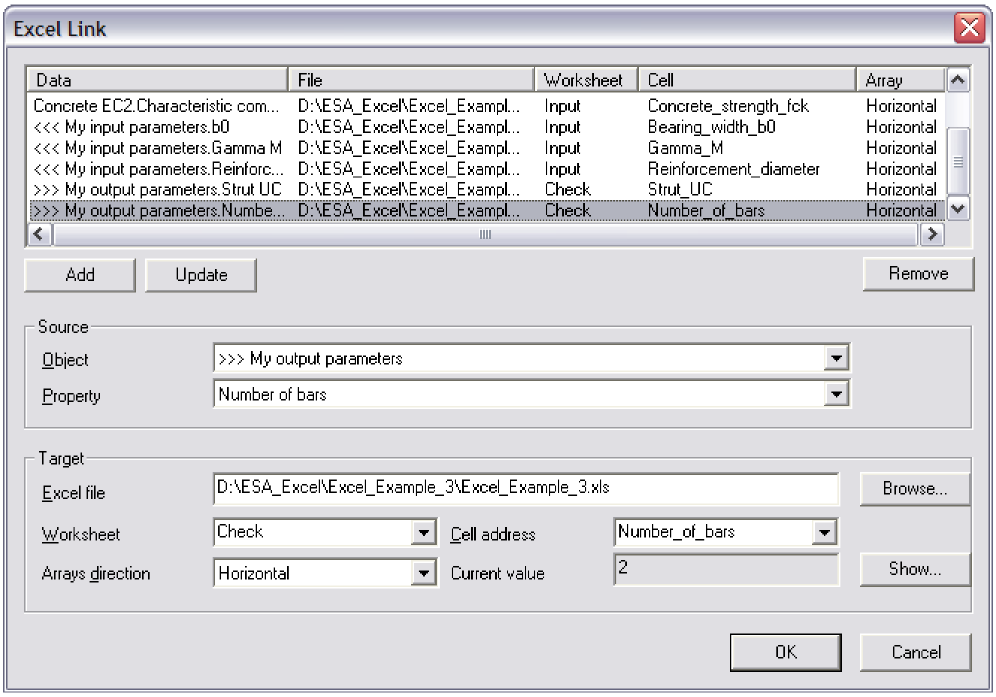
The **Arrays direction** is set to ‘Vertical’ since this array contains two values which need to be positioned in a vertical column.

Finally, in the field **Cell address** the named cell ‘Cross-Section_Parameters’ is chosen using the combo-box. Automatically the **Current value** field will show the current content of the cell, in this case 0,4. During the mapping, the first parameter of the array, in this case the value of the height, will be mapped to cell B3. The second parameter of the array, in this case the value of the width, will be mapped to cell B4.

When all input has been done, this mapping is added to the table using the **Add** button.



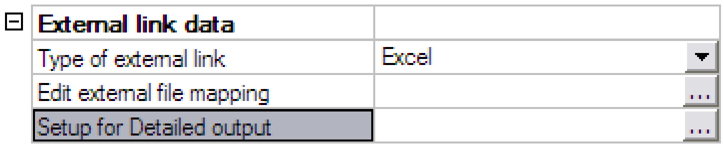
In the same way, all other parameters can be mapped using the above table. For the other parameters, the array direction is left on horizontal.



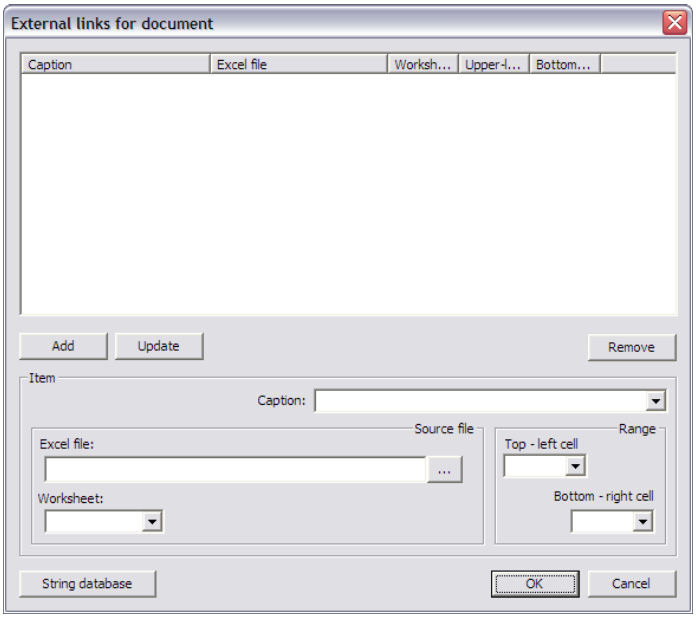
All parameters are now mapped to Excel. The final step left for the definition of the additional data is specifying a Detailed output.

Step 2.10 Define the Detailed output

In *Step 2.7* the parameters for the Brief output have been defined. These parameters will be used for the check. In addition, a Detailed output can also be specified to show an in-depth overview of the check.



By clicking on **Setup for Detailed output**, the Detailed output dialog is opened.

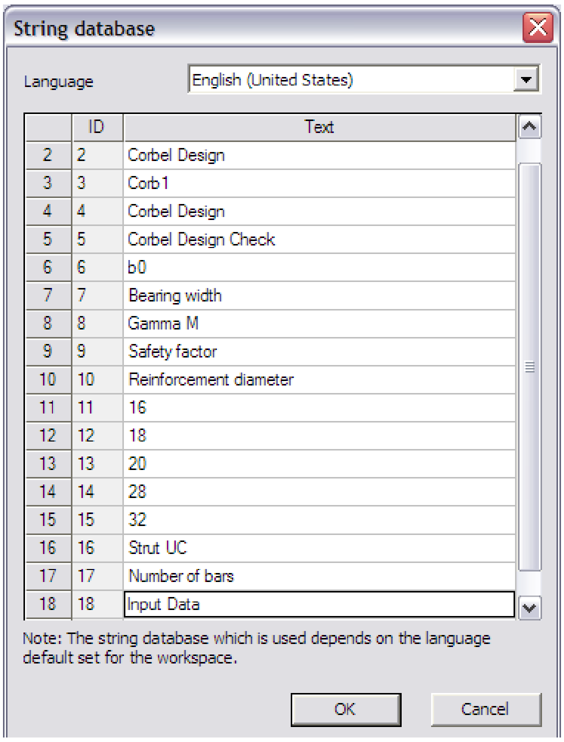


In this example, two ranges will be defined. One which shows the input data and one which shows the results. For ease of reference, here also Named cells have been defined in the Excel file.

Caption	Worksheet	Top – left cell	Bottom – right cell
Input Data	Input	Input_Top_Left	Input_Bottom_Right
Corbel Design	Check	Check_Top_Left	Check _Bottom_Right

Through the button **String database** the text string database can be directly accessed. For this example the following string is added:

Strings used in this example
Input Data



All required strings are no available so the ranges can be defined.

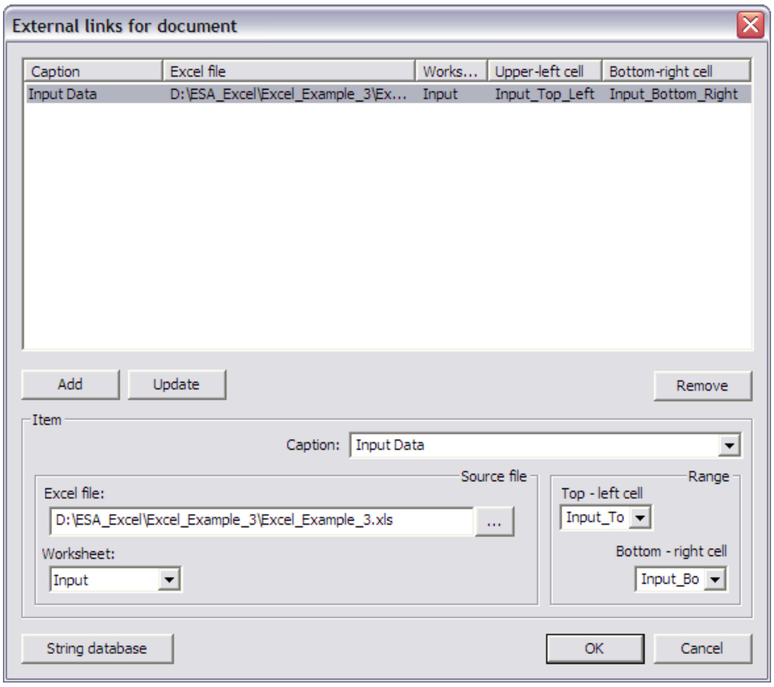
In the **Caption** field the string ‘Input Data’ is chosen.

In the **Excel file** field the file **Excel_Example_3.xls** is searched using the browse button.

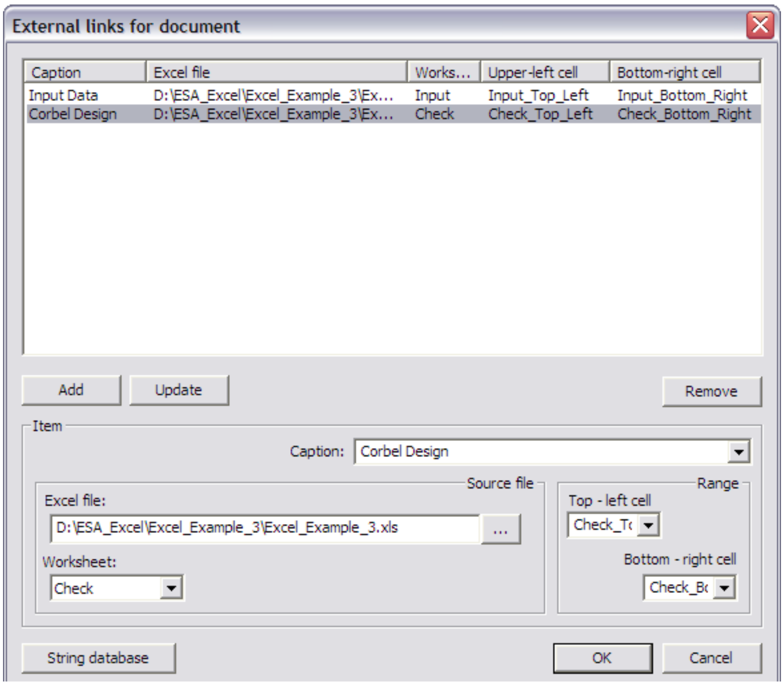
The **Worksheet** field is set to ‘Input’.

In the **Range** group the **Top - left** cell is set as ‘Input_Top_Left’ and the **Bottom - right** cell as ‘Input_Bottom_Right’.

When all input has been done, the data is added to the table using the **Add** button.



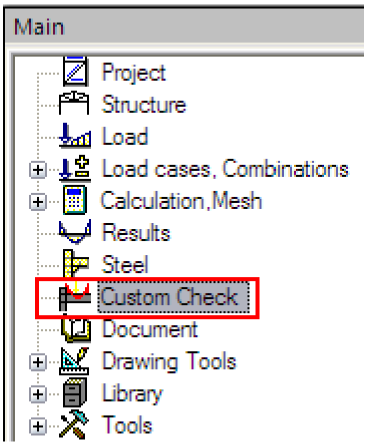
In the same way, using the above table the second range is added.

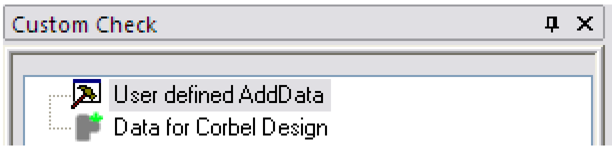


With this final step, the User Defined Additional Data has been fully inputted and the **User Defined Additional Data** Library can be closed.

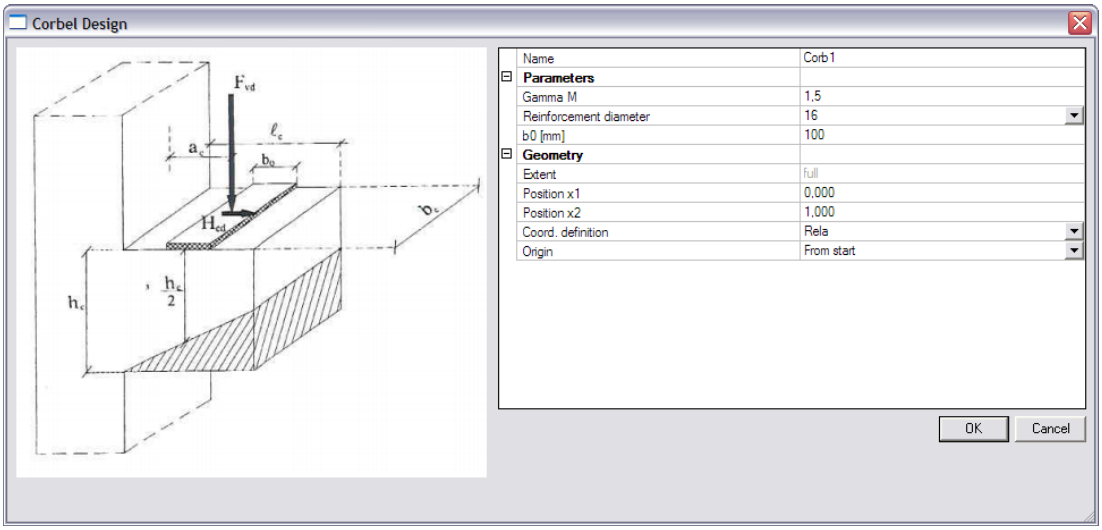
Step 3: Input the User Defined Additional Data on members/nodes

After closing the **User Defined Additional Data** Library a new service will be shown in the Scia Engineer tree: **Custom Check**.



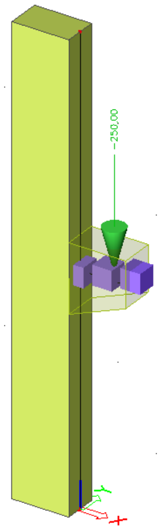


The additional data which was defined in *Step 2*, can now be inputted on the member. When double clicking on **Data for Corbel Design** the dialog with the properties of the data is displayed:



The **Parameters** group holds the user defined parameters of *Step 2.3* with their default values. In this case the safety factor ‘Gamma M’, the Bearing width ‘b0’ and the combo-box ‘Reinforcement diameter’.

The default values of the dialog are confirmed with [**OK**] and the data is inputted on the corbel.



Through the View Parameter settings, the Style / Rendering of the additional data can be specified on the Misc. tab. This works in the same way as for any other type of additional data (loads, supports …).

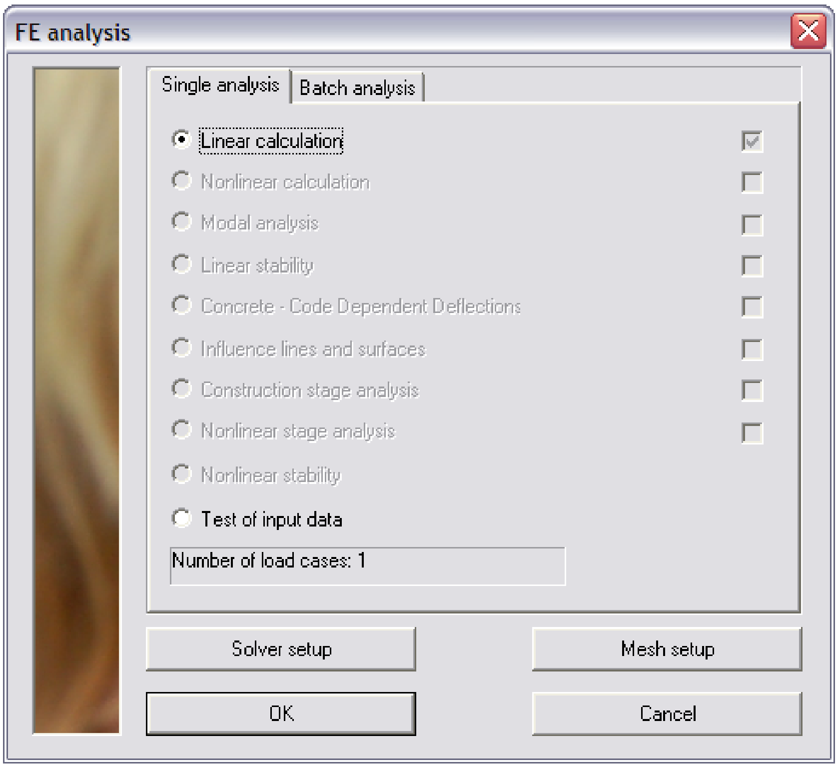
In case the additional data is too large or too small, the scale multiplier ‘User defined AddData’ in the default Scia Engineer Scales manager.

The data has now been inputted and in the next step the check can be executed.

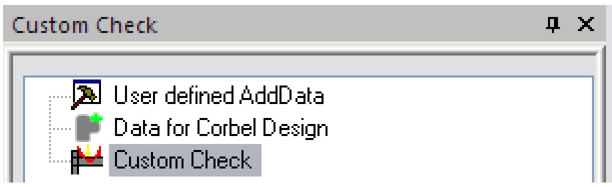
Step 4: Execute the Custom Check

In *Step 2* the additional data has been defined including the definition of the check, the mapping to Excel… In *Step 3* the additional data has been inputted. What is left is the execution of the check.

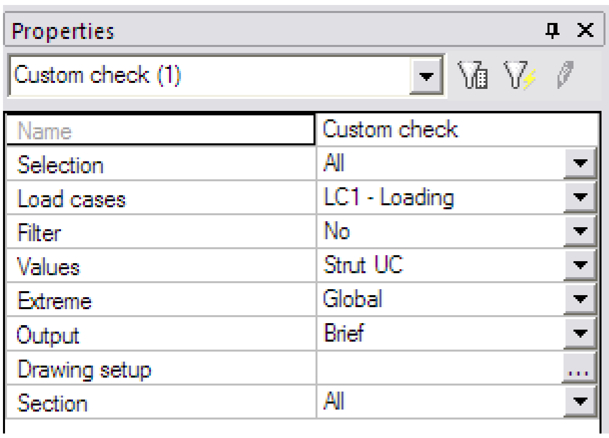
First of all the linear analysis is launched.



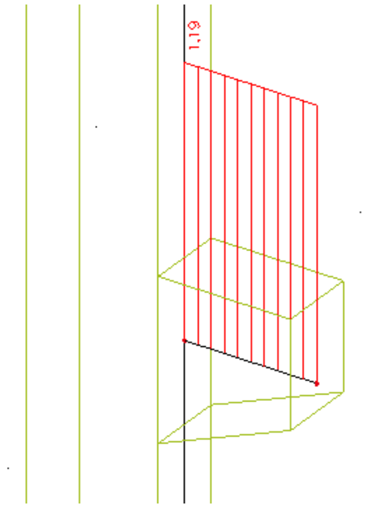
When user defined additional data was inputted and the analysis has been executed, the **Custom Check** service will show a new item: **Custom Check**.



In the property window of the check, the **Values** field is set to 'Strut UC' and the **Refresh** action button is pressed to execute the check.



The following check result is shown on screen:



When pressing the **Preview** action button, the Brief preview shows the following:

Custom check

Linear calculation, Extreme : Global
Selection : All
Load cases : LC1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_3\Excel_Example_3.xls

Type Name	Data	Css	Material	dx [m]	Case
Custom check	Corb1	Corbel - RECT	C30/37	0,000	LC1

Using the table composer, both the Strut UC and number of bars can be added to the output as specified in the previous examples. After refreshing the following preview is shown:

Custom check

Linear calculation, Extreme : Global
Selection : All
Load cases : LC1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_3\Excel_Example_3.xls

Type Name	Data	Css	Material	dx [m]	Case	Strut UC [-]	Number of bars
Custom check	Corb1	Corbel - RECT	C30/37	0,000	LC1	**1,19**	3

Next, the Detailed output is examined. The **Output** field is set to 'Detailed' and the **Refresh** action button is pressed.

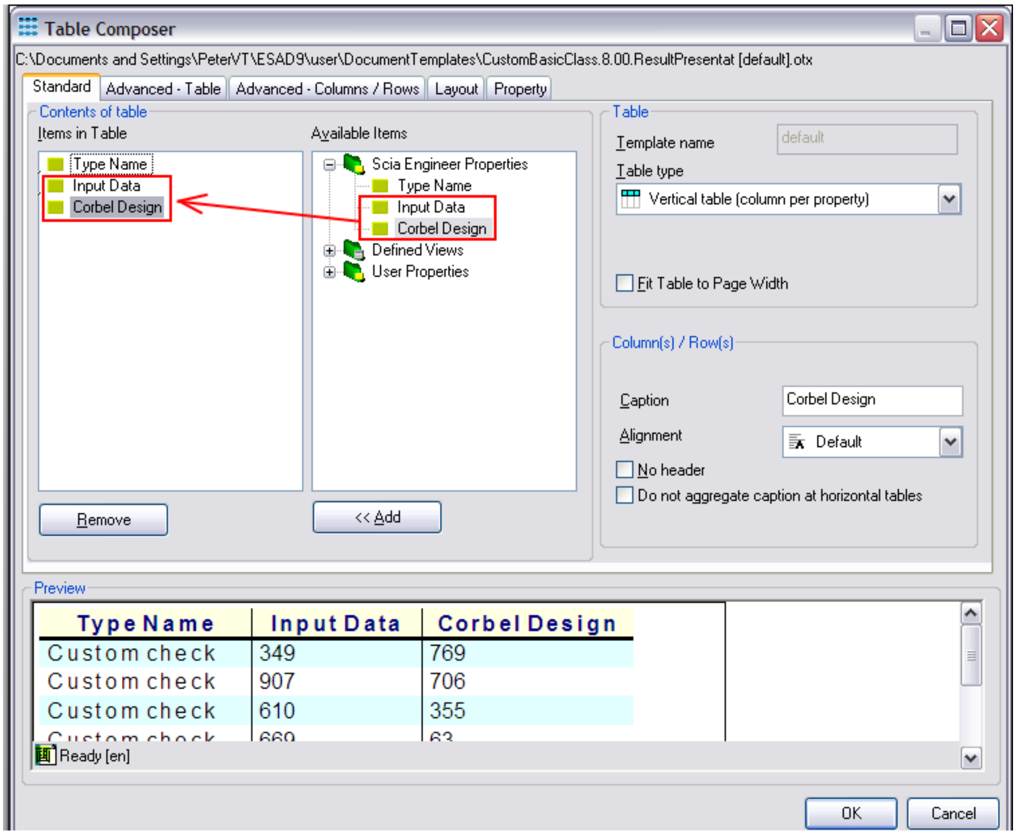
Custom check

Linear calculation, Extreme : Global
Selection : All
Load cases : LC1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_3\Excel_Example_3.xls

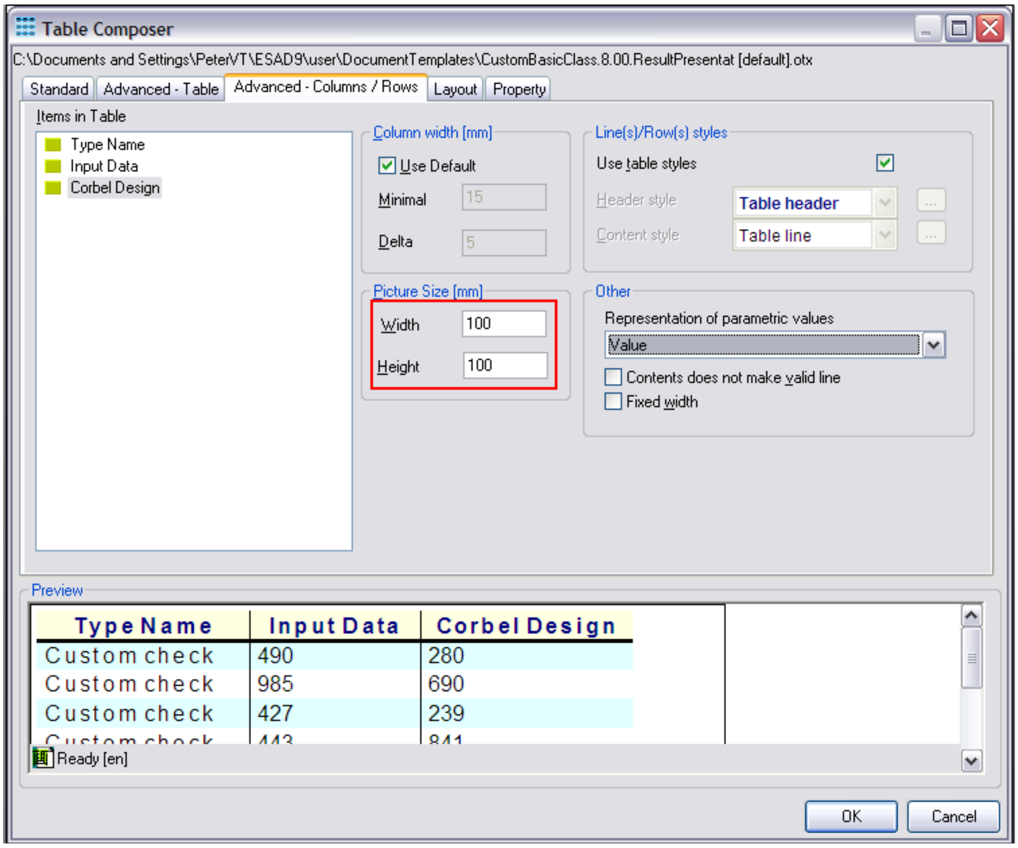
Type Name
Custom check

Here also, the newly created output table needs to be added to the output using the table composer.

Both ranges defined in *step 2.10* can now be added to the output.



In addition, the size of these tables can be set to 100mm by 100mm.



After refreshing, the following output is shown:

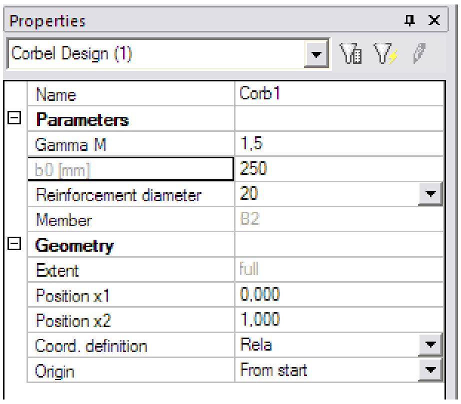
Custom check

Linear calculation, Extreme : Global
Selection : All
Load cases : LC1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_3\Excel_Example_3.xls

Type Name	Input Data		Corbel Design	
Custom check	**Input Data from Scia Engineer**		**Application of strut and tie model**	
			0,4 hc <= ac <= hc	OK
	Haunch height hc	0,6 m		
	Haunch width bc	0,3 m	**Check of strut**	
			Ncd	353,55 kN
	Load position ac	0,25 m		
			A0	30000 mm^2
	Load Fvd	-250000 N		
			Sigma cd	16,66666667 N/mm^2
	Concrete strength fck	30000000 N/m^2	Nu*fcd	14 N/mm^2
			Check	1,19
	Bearing width b0	0,1 m		
	Gamma M	1,5 -	**Design of tension reinforcement**	
			Cover c	21 mm
	Reinforcement diameter	16 mm		
			d	571 mm
			As required	417,10 mm^2
	Calculated Input data			
			As to provide	3 diam. 16 603 mm^2
	Load Hcd (=0,20 * Fvd)	-50000 N		

The output shows the array mapping of the haunch dimensions hc and bc. In addition the conditional formatting used in the Excel file is nicely shown in the Scia Engineer output.

In a next step, the additional data is selected and some changes are made to the input: the bearing width b0 is increased to **250 mm** and the reinforcement diameter is set to **20**.



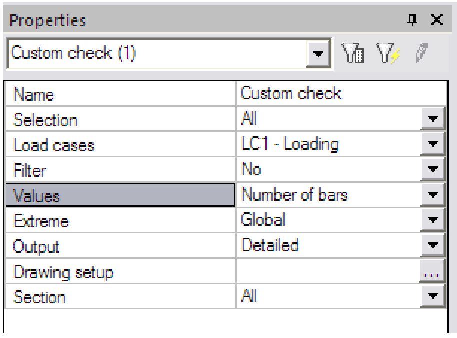
After refreshing these changes are shown in the detailed output:

Custom check

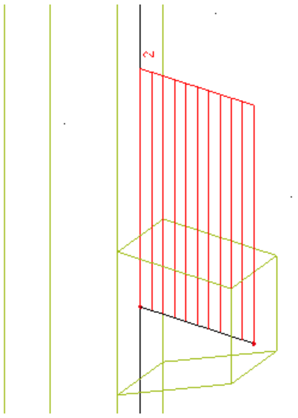
Linear calculation, Extreme : Global
Selection : All
Load cases : LC1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_3\Excel_Example_3.xls

Type Name	Input Data	Corbel Design
Custom check	**Input Data from Scia Engineer**	**Application of strut and tie model** 0,4 hc <= ac <= hc OK
	Haunch height hc 0,6 m Haunch width bc 0,3 m	**Check of strut**
	Load position ac 0,25 m	Ncd 353,55 kN A0 75000 mm^2
	Load Fvd -250000 N	Sigma cd 6,666666667 N/mm^2
	Concrete strength fck 30000000 N/m^2	Nu*fcd 14 N/mm^2
	Bearing width b0 0,25 m	Check 0,48
	Gamma M 1,5 -	**Design of tension reinforcement**
	Reinforcement diameter 20 mm	Cover c 25 mm d 565 mm
	Calculated Input data	As required 496,64 mm^2
	Load Hcd (=0,20 * Fvd) -50000 N	As to provide 2 diam. 20 628 mm^2

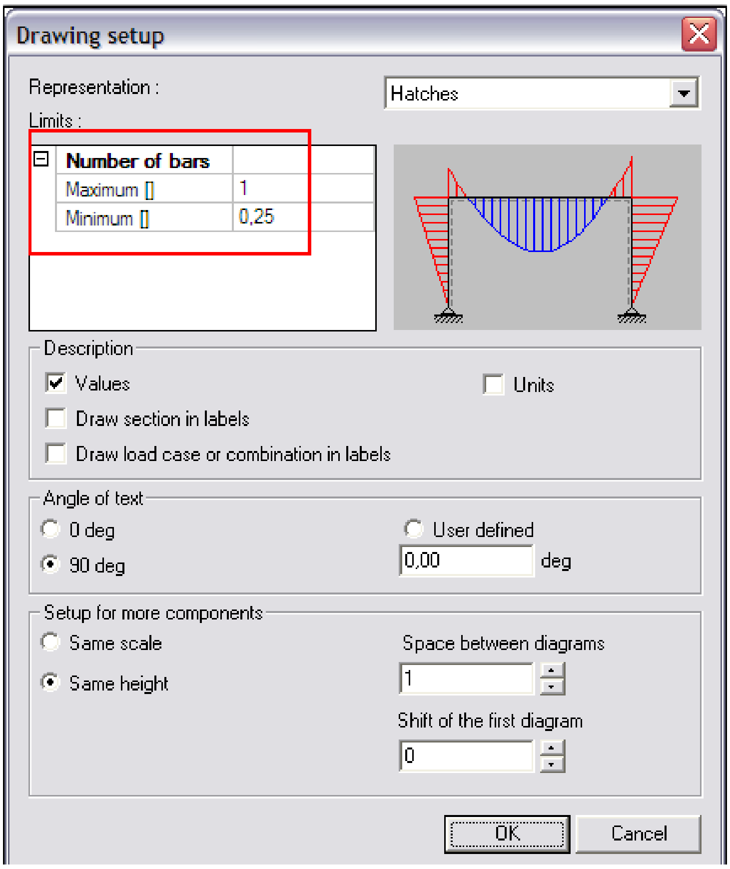
Previously the Strut unity check was displayed on screen. In the same way the number of bars can be shown by changing the **Values** field to ‘Number of bars’.



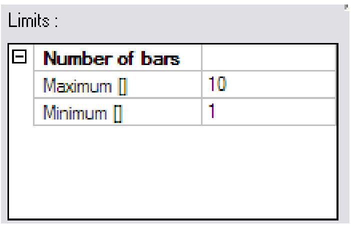
After refreshing, the following result is shown on screen:



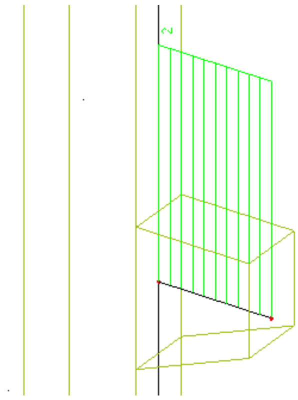
By default this graphical result is shown in red since the result is greater than 1,00. However, using the default Drawing Setup of Scia Engineer, this setting can be changed as in any other check service.



The default minimum and maximum limits for check results are 0,25 and 1,00. For the Number of bars, the minimum can be set to 1 and the maximum to for example 10.



After refreshing, the graphical result is now shown in green since the value is between the minimum and maximum limits.



Step 5: Save the User Defined Additional Data into a database for future use

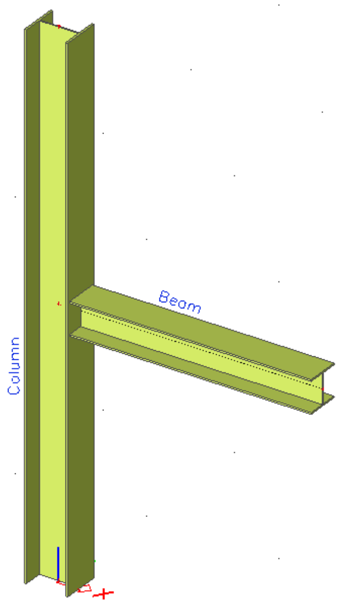
If required, this additional data can be saved into a database for future use as illustrated in Example 1.

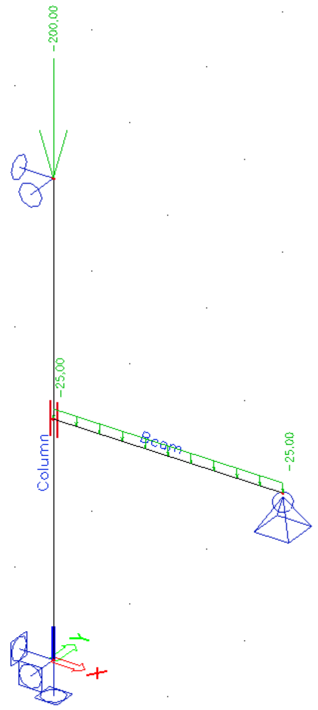
Example 4: Moment Resisting Connection

In this fourth example, the working of slave data is explained. In addition, the use of point data and nodal data is illustrated

As a practical case a steel moment resisting connection is used.

In this example, a typical beam-column connection is modelled. The column has a height of **4m** and the beam is attached in the middle. The beam has a length of **2m**. Both members are manufactured in **S235** according to **EC-EN**.





For this example, the beam has a **HEA 260** cross-section. The column also has a cross-section of type **HEA** but can vary (i.e. any type of HEA can be applied).

Three load cases are defined:

- LC1: Self Weight of the members
- LC2: Dead Load: **50 kN/m** on the beam
- LC3: Live Load: **25 kN/m** on the beam, **200 kN** on the column

The load cases are grouped into an ultimate limit state combination of type **EN-ULS (STR)**.

The check will be done according to the Excel file "**Excel_Example_4.xls**"

The Excel file contains two worksheets. On the sheet 'Input' the input data are set:

	A	B	C
1	**Input Data from Scia Engineer**		
2			
3	Column Cross-section	HEA340	
4			
5	MEd Beam at connection	65000	Nm
6			
7	Bolts	M16	
8			
9			
10	**Calculated Input Data**		
11			
12	Column Type HEA	340	

The sheet 'Check' shows the determination of the moment resistance and the unity check.

	A	B	C	D
1	**DSTV Table for IH3 connection with HEA260 beam**			
2				
3	Bolts	Column HEA	My,Rd [kNm]	Limiting part
4	M16	180	30,24	Column Flange in bending
5		220	45,36	Column Flange in bending
6		240	60,48	Column Flange in bending
7		340	75,6	Bolts in Tension
8	M20	200	47,96	Column Flange in bending
9		280	71,94	Column Web in Shear
10		340	95,92	Column Web in Shear
11		450	119,9	Bolts in Tension
12	M24	240	65,32	Column Web in Compression
13		320	97,98	Column Web in Compression
14		450	130,6	Column Web in Compression
15		700	163,3	Bolts in Tension
16				
17	**Check of Connection**			
18				
19	Connection according to DSTV Anlage 1,43			
20				
21	Connection type: IH 3 with Beam HEA260			
22				
23	Bolts	M16		
24				
25	Column	HEA340		
26				
27	MEd	65	kNm	
28				
29	MRd	75,6	kNm	
30				
31	Limiting part	Bolts in Tension		
32				
33	Unity Check	0,86	-	

The determination of the moment resistance is based on tabulated values from "*DSTV, Anlage 1.43 zum Prüfbescheid II B 3-543-585 vom 14.07.2000"* which gives the MRd value of an extended end plate connection for different combinations of beam – column – bolts.

For this example, the beam has been given a fixed cross-section of HEA 260. The bolts can vary between M16, M20 and M24. The column can have any type of HEA cross-section.

As for a typical connection, data of different entities has to be sent to Excel. This can be done using slave additional data:

- From the column the cross-section has to be mapped. No check or output is required. On the column slave data will thus be defined.
- From the beam the end moment has to be mapped. No check or output is required. On the beam slave data will thus be defined.
- The actual Connection Data will include the bolt type and the connection check, combining the input of the previous two 'slaves'. The connection data will be inputted on the node between the beam and column.

Step 1: Activate the functionality External Application Checks

The first step is to activate the functionality **External application checks** on the **Functionality** tab in the **Project Data**.

Step 2: Create User Defined Additional Data

In the second step, User Defined Additional Data will be defined.

Through **Tools** > **User defined AddData** the **User Defined Additional Data** Library can be opened.

As specified, three types of additional data will be defined:

- Column Data
- Beam Data
- Connection Data

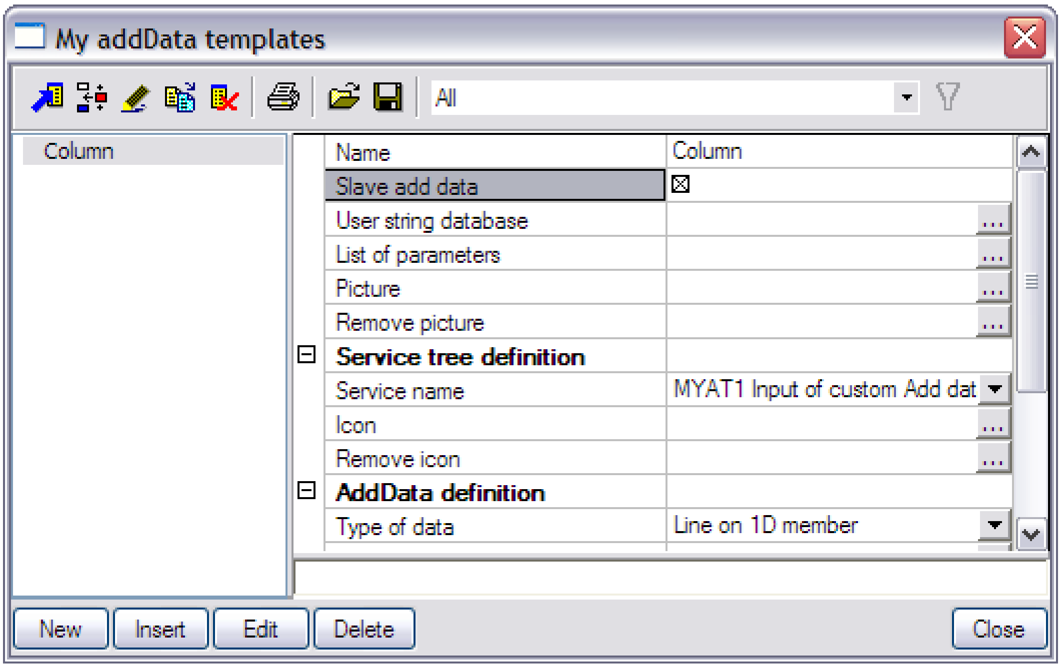
The different input steps are thus repeated for each data.

Step 2.1. (Column) Slave data

First of all the additional data for the column is inputted.

The **Name** of the additional data is therefore changed to 'Column'.

Since it concerns slave additional data the checkbox **Slave add data** is activated.

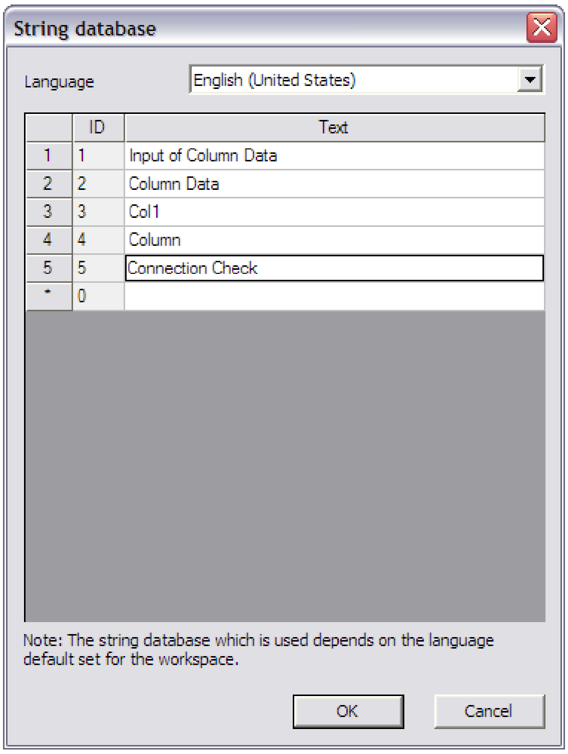


As explained during the first example, slave data can only be used to send data to Excel, not to read data from Excel. Therefore all output options are not available for this kind of data.

Step 2.2. (Column) Define text strings

In the **User string database** the required strings are defined for the definition of the additional data. For the column data the strings are modified as follows:

Type for which the string is used	Default string	String used in this example
Service name	MYAT1 Input of custom add data	Input of Column Data
Type name	MYAT1 Custom defined add data	Column Data
Short name	MYAT1 MADI	Col1
Description	MYAT1 Description	Column
Name of check	MYAT1 Custom check	Connection Check



Since slave data have no output and thus no check, the text string for the check will not be used and therefore it is not required to modify it.

Step 2.3. (Column) Define parameters

In this example, the cross-section of the column has to be mapped to Excel. Therefore no additional parameters are required and thus no parameters are defined.

Step 2.4. (Column) Add a picture to the Additional Data

In this example, no picture is added to the column data since this data just represents slave data and does not require any clarifying picture.

Step 2.5. (Column) Define Service Tree

In the next step the Service Tree is defined through the group **Service tree definition**.

The **Service name** is taken automatically from the text string database.

Service tree definition	
Service name	Input of Column Data
Icon	...
Remove icon	...

In this example, no icon is added to the column data.

Step 2.6. (Column) Define the Additional Data

Using the data from the previous steps, the additional data can now be defined in the group **AddData definition**.

AddData definition	
Type of data	Line on 1D member
Instance setup	...
Type name	Column Data
Short name	Col1
Description	Column

The field **Type of data** is set to 'Line on 1D member'.

> ***In this example, only the cross-section of the column will be sent to Excel so also 'Point on 1D member' could have been used.***

The **Type name**, **Short name** and **Description** are taken automatically from the text string database.

To get an overview of all the data entered in the previous steps the button **Instance Setup** is used.

Column Data

Name	
Parameters	
Drawings	
Drawing style	Box on line
Property for drawing on begin	-
Property for drawing on end	-
Colour	Others
Geometry	
Extent	full
Position x1	0,000
Position x2	1,000
Coord. definition	Rela
Origin	From start

OK Cancel

In the **Drawings** group, the **Drawing style** is set to 'Box on line'. For the **Colour field** 'Thermal load' is chosen.

Drawings	
Drawing style	Box on line
Property for drawing on begin	-
Property for drawing on end	-
Colour	Thermal load

Since no user defined parameters have been inputted, no parameter can be set as **Property for drawing on begin/end**. These fields are thus left empty.

Step 2.7. (Column) Specify the type of external link

In the **External link data** group the **Type of external link** allows to specify which external application will be used.

External link data	
Type of external link	Excel
Edit external file mapping	...

In this example the link is made with Excel and thus ‘Excel’ is chosen.

Step 2.8. (Column) Define the mapping with the external application

The preparation of the slave column data has now been done, what remains is defining the mapping to Excel.

Through the button **Edit external file mapping** the mapping dialog is opened.

In this example, the Excel file contains two worksheets. On the sheet ‘Input’ the input data from Scia Engineer are set:

	A	B	C
1	**Input Data from Scia Engineer**		
2			
3	Column Cross-section	HEA340	
4			
5	MEd Beam at connection	65000	Nm
6			
7	Bolts	M16	
8			
9			
10	**Calculated Input Data**		
11			
12	Column Type HEA	340	

This is the only sheet of importance for the column data since slave data concern only the mapping of input properties.

The following table shows which property should be mapped to which cell:

Object	Property	Worksheet	Cell Address
Cross-Sections	Type	Input	B3

The mapping of this property is thus done as follows:

The **Object** field is set to ‘Cross-sections’.

In the **Property** field ‘Type’ can then be chosen.

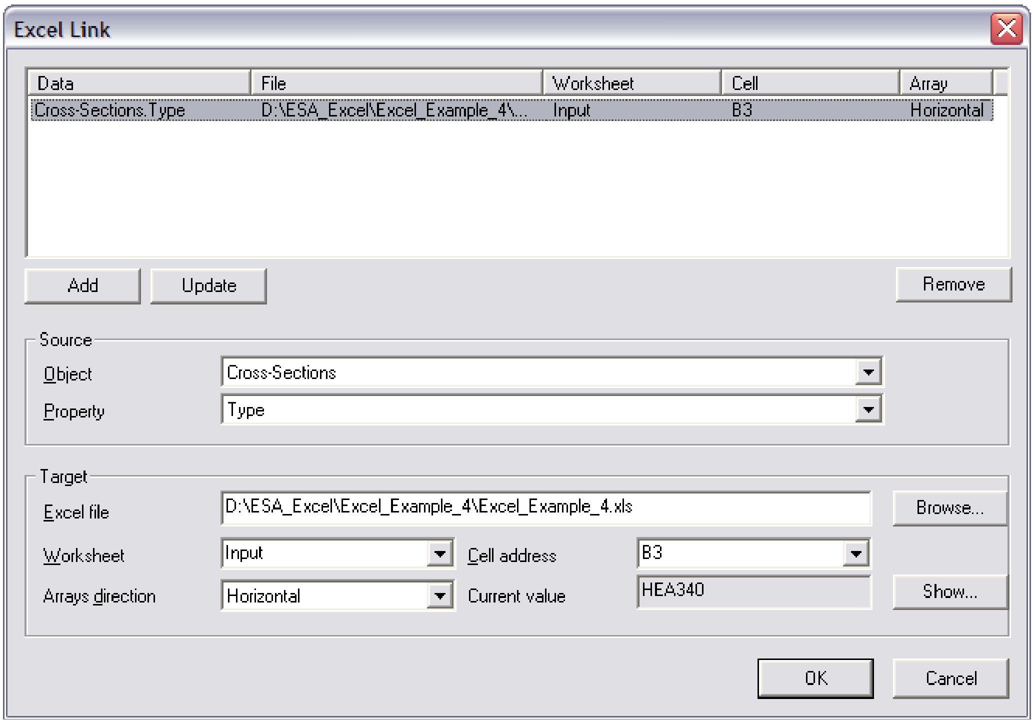
Using the **Browse** button, the file **Excel_Example_4.xls** is searched.

After the file has been specified, the **Worksheet** field contains a list of all sheets. This field is set to ‘Input’.

The **Arrays direction** is set to ‘Horizontal’.

Finally, in the field **Cell address** the cell ‘B3’ is typed. Automatically the **Current value** field will show the current content of the cell, in this case HEA340.

When all input has been done, this mapping is added to the table using the **Add** button.



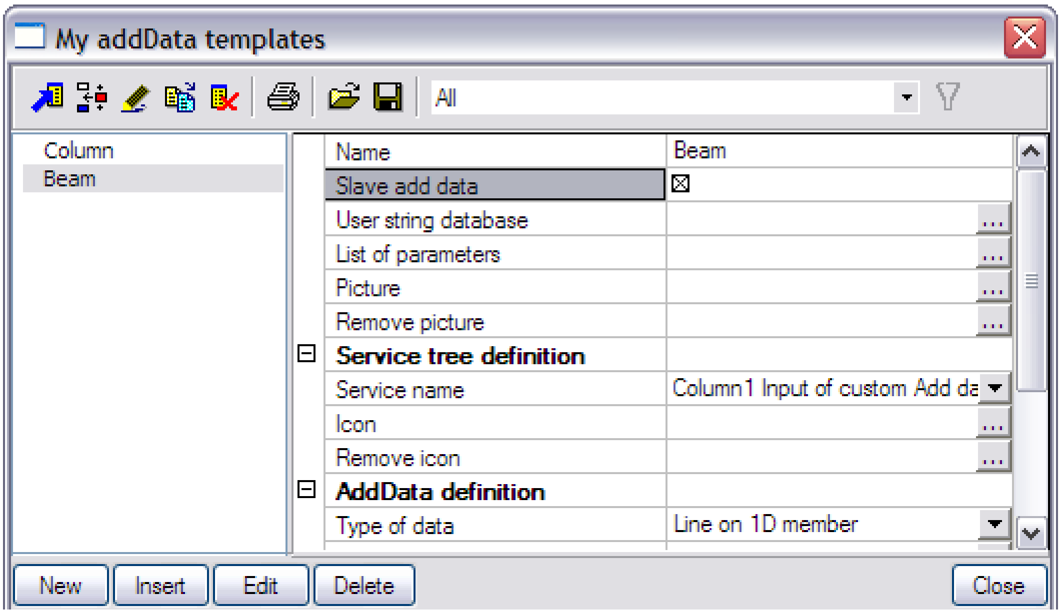
With the input of the mapping, the definition of the column slave data has been completed. In the same way, slave data will now be defined for the beam.

Step 2.1. (Beam) Slave data

Second, the additional data for the beam is inputted using the button **New** in the **User Defined Additional Data** Library.

The **Name** of the new additional data is changed to ‘Beam’.

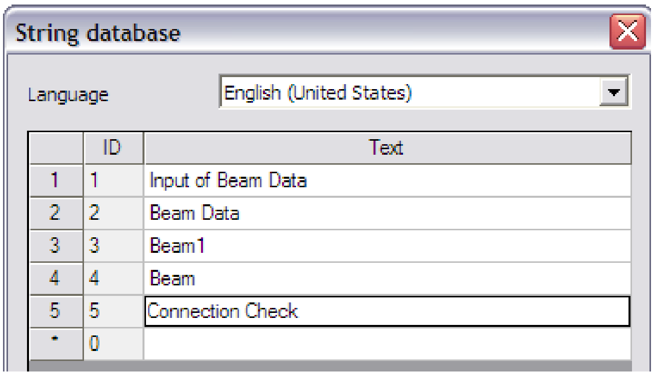
Since it concerns slave additional data the checkbox **Slave add data** is activated.



Step 2.2. (Beam) Define text strings

In the **User string database** the required strings are defined for the definition of the additional data. For the beam data the strings are modified as follows:

Type for which the string is used	Default string	String used in this example
Service name	Column1 Input of custom add data	Input of Beam Data
Type name	Column 1 Custom defined add data	Beam Data
Short name	Column 1 MADI	Beam1
Description	Column 1 Description	Beam
Name of check	Column 1 Custom check	Connection Check



Step 2.3. (Beam) Define parameters

In this example, the bending moment of the beam has to be mapped to Excel. Therefore no additional parameters are required and thus no parameters are defined.

Step 2.4. (Beam) Add a picture to the Additional Data

In this example, no picture is added to the beam data since this data just represents slave data and does not require any clarifying picture.

Step 2.5. (Beam) Define Service Tree

In the next step the Service Tree is defined through the group **Service tree definition**.

The **Service name** is taken automatically from the text string database.

Service tree definition	
Service name	Input of Beam Data
Icon	...
Remove icon	...

In this example, no icon is added to the beam data.

Step 2.6. (Beam) Define the Additional Data

Using the data from the previous steps, the additional data can now be defined in the group **AddData definition**.

AddData definition	
Type of data	Point on 1D member
Instance setup	...
Type name	Beam Data
Short name	Beam1
Description	Beam

The field **Type of data** is set to 'Point on 1D member'. The purpose of this beam slave data is to send the bending moment at the location of the connection. Therefore only one position is required, the end of the beam at which the connection is located.

The **Type name**, **Short name** and **Description** are taken automatically from the text string database.

To get an overview of all the data entered in the previous steps the button **Instance Setup** is used.

Beam Data

Name	
Parameters	
Drawings	
Drawing style	Point box on line
Property for drawing	-
Colour	Thermal load
Geometry	
Extent	full
Position x	0.000
Coord. definition	Rela
Origin	From start
Repeat (n)	1

OK Cancel

In the **Drawings** group, the **Drawing style** is set to 'Point box on line'. For the **Colour field** 'Generated load' is chosen.

Drawings	
Drawing style	Point box on line
Property for drawing	-
Colour	Generated load

Since no user defined parameters have been inputted, no parameter can be set as **Property for drawing on begin/end**. These fields are thus left empty.

Step 2.7. (Beam) Specify the type of external link

In the **External link data** group the **Type of external link** allows to specify which external application will be used.

External link data	
Type of external link	Excel
Edit external file mapping	...

In this example the link is made with Excel and thus ‘Excel’ is chosen.

Step 2.8. (Beam) Define the mapping with the external application

The preparation of the slave beam data has now been done, what remains is defining the mapping to Excel.

Through the button **Edit external file mapping** the mapping dialog is opened.

In this example, the Excel file contains two worksheets. On the sheet ‘Input’ the input data from Scia Engineer are set:

	A	B	C
1	**Input Data from Scia Engineer**		
2			
3	Column Cross-section	HEA340	
4			
5	MEd Beam at connection	65000	Nm
6			
7	Bolts	M16	
8			
9			
10	**Calculated Input Data**		
11			
12	Column Type HEA	340	

This is the only sheet of importance for the beam data since slave data concern only the mapping of input properties.

The following table shows which property should be mapped to which cell:

Object	Property	Worksheet	Cell Address
Internal forces on member	My	Input	B5

The mapping of this property is thus done as follows:

The **Object** field is set to 'Internal forces on member'.

In the **Property** field 'My' can then be chosen.

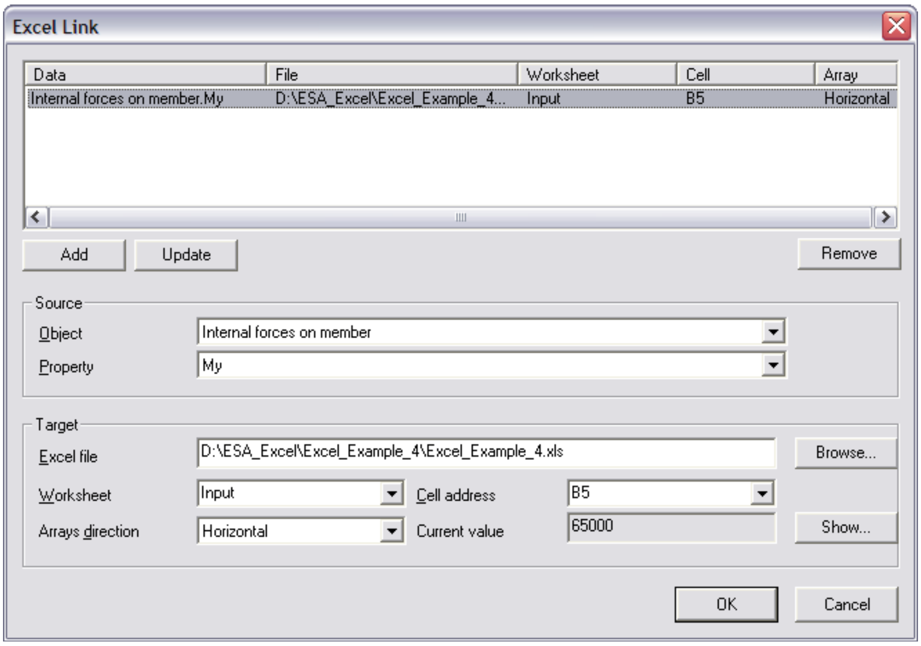
Using the **Browse** button, the file **Excel_Example_4.xls** is searched.

After the file has been specified, the **Worksheet** field contains a list of all sheets. This field is set to 'Input'.

The **Arrays direction** is set to 'Horizontal'.

Finally, in the field **Cell address** the cell 'B5' is typed. Automatically the **Current value** field will show the current content of the cell, in this case 65000.

When all input has been done, this mapping is added to the table using the **Add** button.



With the input of the mapping, the definition of the beam slave data has been completed. Both column and beam slave data are defined and now in the final step the master data, the connection data will be inputted.

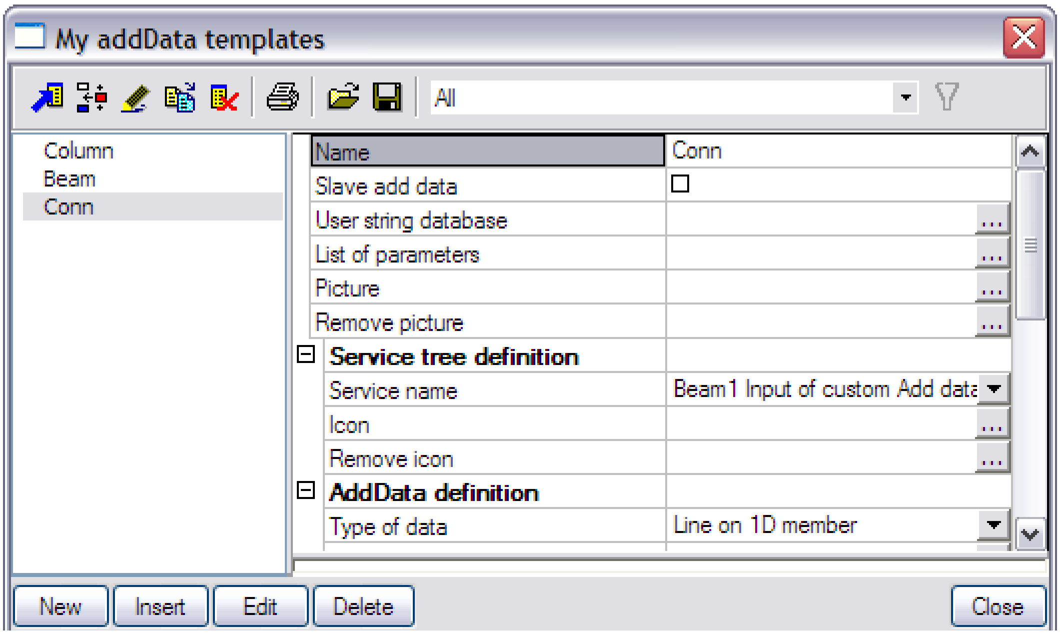
The order of defining additional data is of no importance. In this example, first the slave data was inputted and then the master data. It makes no difference if first the master data would be inputted and then the slave data.

Step 2.1. (Connection) Slave data

Finally, the additional data for the connection itself is inputted using the button **New** in the **User Defined Additional Data** Library.

The **Name** of the new additional data is changed to 'Conn'.

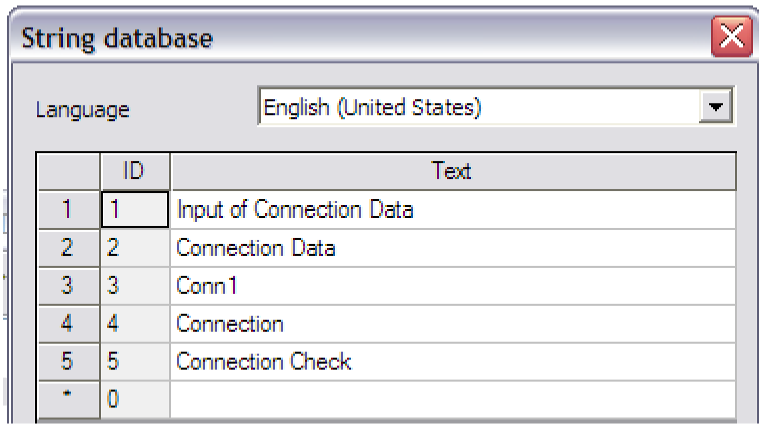
Since it concerns master additional data the checkbox **Slave add data** is not activated.



Step 2.2 (Connection) Define text strings

In the **User string database** the required strings are defined for the definition of the additional data. For the connection the strings are modified as follows:

Type for which the string is used	Default string	String used in this example
Service name	MYAT1 Input of custom add data	Input of Connection Data
Type name	MYAT1 Custom defined add data	Connection Data
Short name	MYAT1 MADI	Conn1
Description	MYAT1 Description	Connection
Name of check	MYAT1 Custom check	Connection Check



The necessary strings for the definition of the data are defined and in the next step the parameters can be defined.

Step 2.3 (Connection) Define parameters

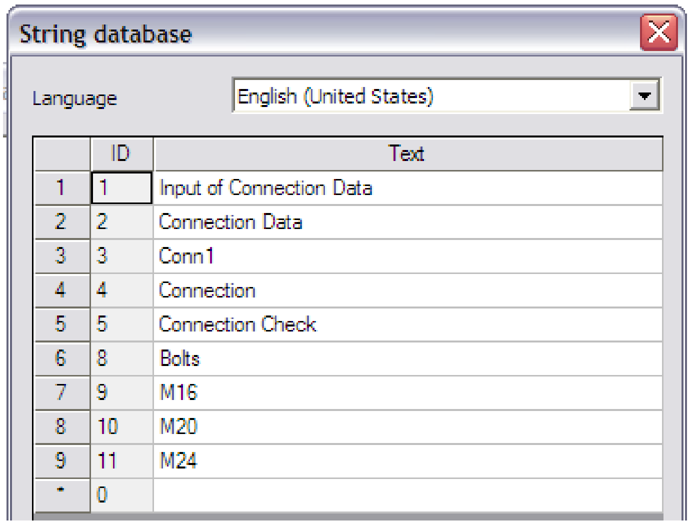
In this example, the connection will have a combo-box parameter from which the user can select the bolt type.

Parameter	Type	Combo-box lines
Bolts	Combo-box	M16 M20 M24

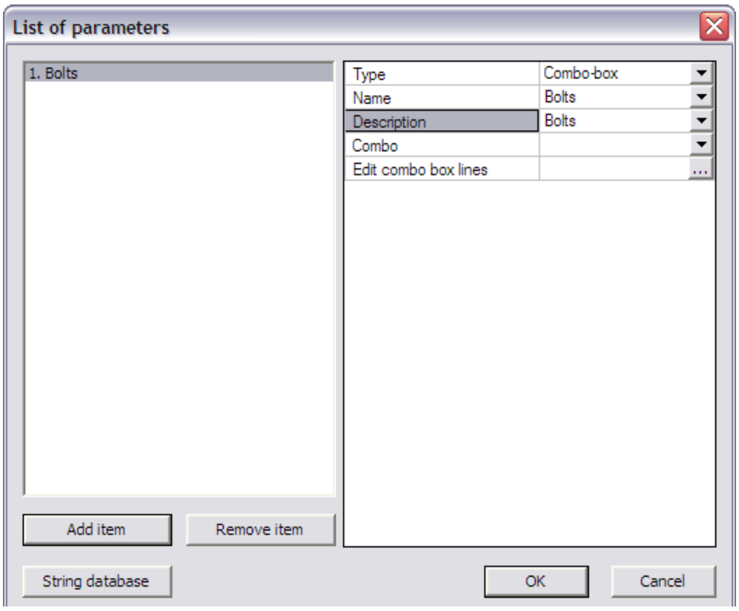
Through the button **String database** the text string database can be directly accessed. This allows a quick input of the strings required for the parameters.

For this example the following strings are added:

Strings used in this example
Bolts
M16
M20
M24



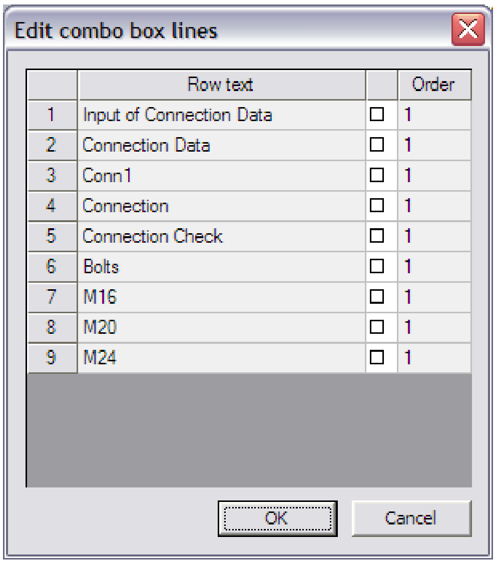
Through the button **Add item** the parameter is added.



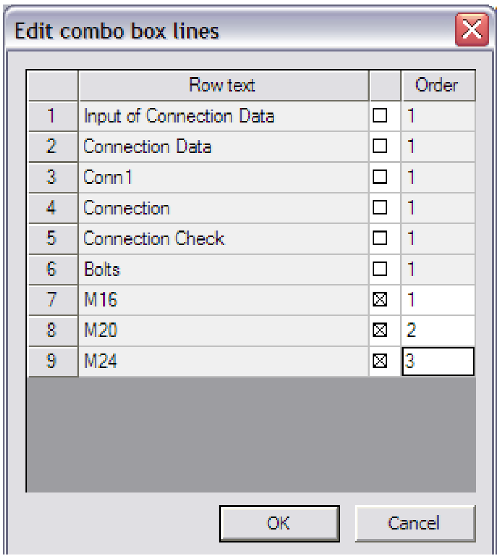
The **Type** field is set to ‘Combo-box’.

For both the **Name** and **Description** fields the string ‘Bolts’ is set.

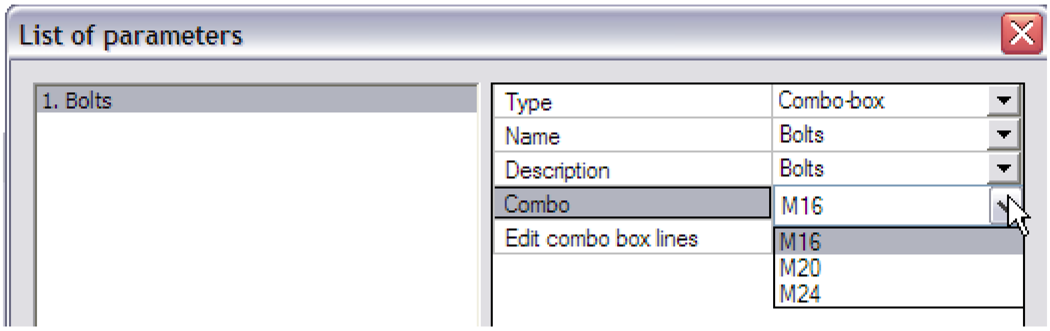
Next, the lines in the combo-box are defined through the edit button **Edit combo box lines**.



The bolt types inputted in the string database are selected and in the Order column the numbers ‘1’, ‘2’and ‘3’ are inputted.



When closing this dialog, the **Combo** item in the **List of Parameters** dialog shows how the combo-box will look like.

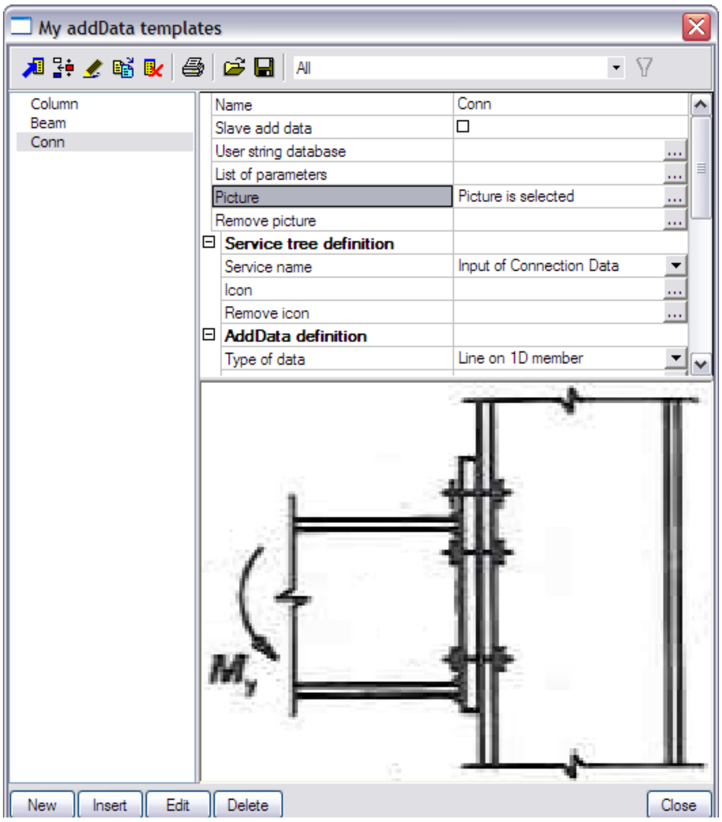


The combo-box has now been inputted and the dialog can be closed.

Step 2.4 (Connection) Add a picture to the Additional Data

To clarify the use of the additional data and the defined parameters a picture can be added using the **Picture** button.

In this example the picture **Excel_Example_4_Picture.bmp** will be used.



Step 2.5 (Connection) Define Service Tree

In the next step the Service Tree is defined through the group **Service tree definition**.
The **Service name** is taken automatically from the text string database.

To clarify the Service name, an icon can be added using the **Icon** button.
In this example the icon **Excel_Example_4_Icon.bmp** will be used.

⊟ Service tree definition	
Service name	Input of Connection Data
Icon	Icon is selected
Remove icon	

Step 2.6 (Connection) Define the Additional Data

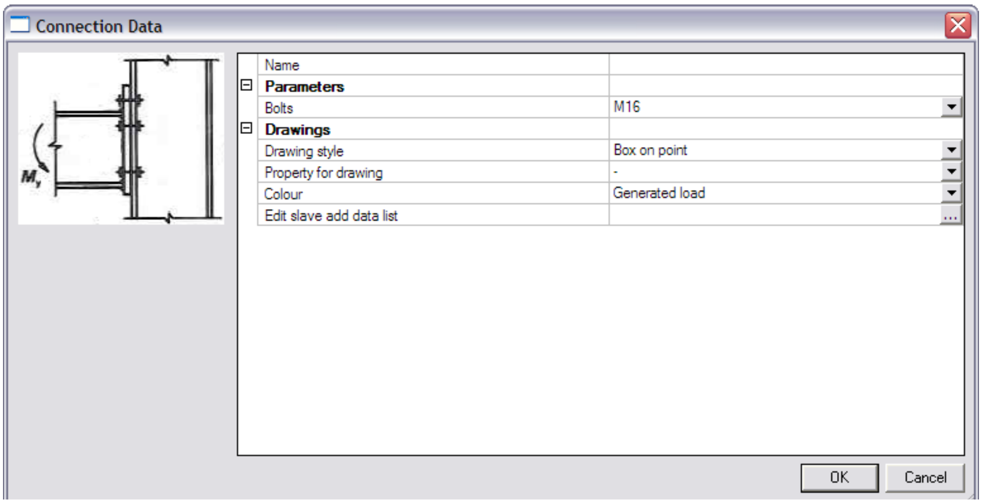
Using the data from the previous steps, the additional data can now be defined in the group **AddData definition**.

⊟ AddData definition	
Type of data	In node
Instance setup	
Type name	Connection Data
Short name	Conn1
Description	Connection

The connection is defined in the node between the beam and column. Therefore the field **Type of data** is set to 'In node'.

The **Type name**, **Short name** and **Description** are taken automatically from the text string database.

To get an overview of all the data entered in the previous steps the button **Instance Setup** is used.



In the **Drawings** group, the **Drawing style** is set to ‘Sphere on point. For the **Colour field** ‘Nodes, rigid arms’ is chosen.

Drawings	
Drawing style	Sphere on point
Property for drawing	-
Colour	Nodes, rigid arms

No numerical user defined parameter has been inputted and therefore the field **Property for drawing** is left empty.

Since the connection data concerns ‘master’ data and the User Defined Additional Data contains also ‘slave’ data (the beam data and column data) a new option is visible: **Edit slave add data list**. This option will be explained further in *Step 3*.

Step 2.7 (Connection) Define the Check

In the group **Check data** the necessary data for the check itself can now be defined.

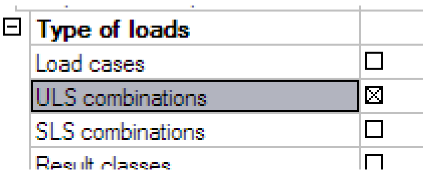
Note that the defining of check data is only available for the connection data since it was not marked as slave data.

Check data	
Name of check	Connection Check
Setup for Brief output	...
Type of loads	
Load cases	☐
ULS combinations	☐
SLS combinations	☐
Result classes	☐

The **Name of check** is taken automatically from the text string database.

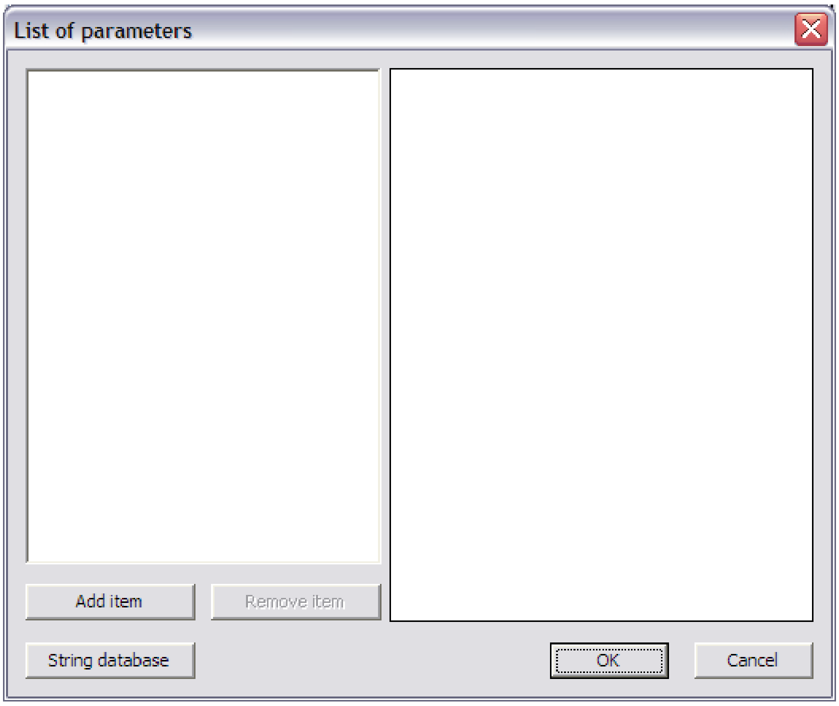
The **Type of loads** group allows to specify which load types will be available for the check. Only the selected items will be available when executing the check.

In this example, both load cases and an ULS combination are available however the check is only required to be executed for the combination. Therefore only the check-box ‘ULS combinations’ is checked.



Important remark: In case more than one load type has been activated, the check will be executed SIMULTANEOUSLY for all load types together! This implies for example that the check is done for both a load case and a combination at the same time. This allows the use of special checks: in the Excel file it can be set that a certain check can be done for the load case while a different check is done for the combination. In general, it is recommended to use only one load type.

The final item for defining the check is the **Setup for Brief output** where the output parameters have to be defined.

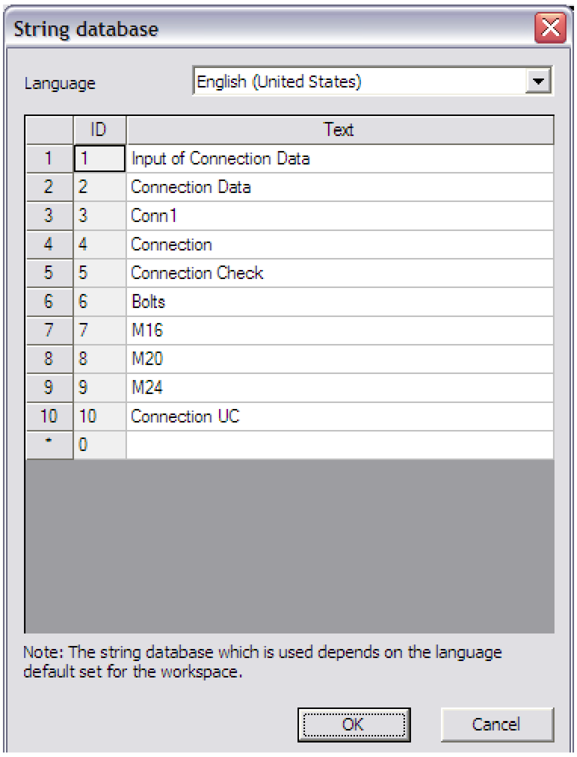


For this example, one output parameters will be defined: the unity check UC of the connection.

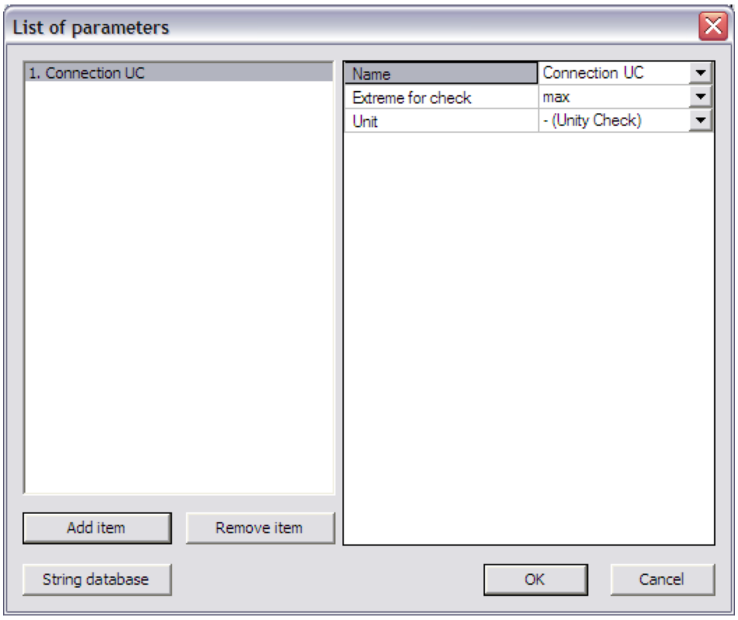
Parameter	Type	Unit
Connection UC	Number	-

First of all, through the button **String database** the text string database is accessed to define the required strings. For this example the following string is added:

String used in this example
Connection UC



When the string is defined, the parameter is added through the button **Add item**.



In the **Name** field the ‘Connection UC’ string is chosen from the string database.
The **Extreme for check** is left on ‘max’ since the maximal unity check value is extreme in this case.

Since it concerns a unity check, the **Unit** field is left on ‘– (Unity Check)’.

The check and output parameters have now been defined so in the next step the link can be set.

Step 2.8 (Connection) Specify the type of external link

In the **External link data** group the **Type of external link** allows to specify which external application will be used.

External link data	
Type of external link	Excel
Edit external file mapping	...
Setup for Detailed output	...

In this example the link is made with Excel and thus ‘Excel’ is chosen.

Step 2.9 (Connection) Define the mapping with the external application

All preparation has now been done, what remains is the most important step of the process: defining the actual mapping between properties and parameters of Scia Engineer and the data fields (i.e. Excel cells) of the external application.

Through the button **Edit external file mapping** the mapping dialog is opened.

In this example, the Excel file contains two worksheets. On the sheet ‘Input’ the input data from Scia Engineer are set:

	A	B	C
1	**Input Data from Scia Engineer**		
2			
3	Column Cross-section	HEA340	
4			
5	MEd Beam at connection	65000	Nm
6			
7	Bolts	M16	
8			
9			
10	**Calculated Input Data**		
11			
12	Column Type HEA	340	

The sheet ‘Check’ shows the determination of the moment resistance and the unity check.

	A	B	C	D
1	**DSTV Table for IH3 connection with HEA260 beam**			
2				
3	Bolts	Column HEA	My,Rd [kNm]	Limiting part
4	M16	180	30,24	Column Flange in bending
5		220	45,36	Column Flange in bending
6		240	60,48	Column Flange in bending
7		340	75,6	Bolts in Tension
8	M20	200	47,96	Column Flange in bending
9		280	71,94	Column Web in Shear
10		340	95,92	Column Web in Shear
11		450	119,9	Bolts in Tension
12	M24	240	65,32	Column Web in Compression
13		320	97,98	Column Web in Compression
14		450	130,6	Column Web in Compression
15		700	163,3	Bolts in Tension
16				
17	**Check of Connection**			
18				
19	Connection according to DSTV Anlage 1,43			
20				
21	Connection type: IH 3 with Beam HEA260			
22				
23	Bolts	M16		
24				
25	Column	HEA340		
26				
27	MEd	65	kNm	
28				
29	MRd	75,6	kNm	
30				
31	Limiting part	Bolts in Tension		
32				
33	Unity Check	0,86	-	

In the column and beam data, the column cross-section and the beam bending moment have already been mapped to Excel. What is left is the mapping of the bolts and the unity check value.

The following table shows which properties should be mapped to which cells:

Object	Property	Worksheet	Cell Address
<<< My input parameters	Bolts	Input	B7
>>> My output parameters	Connection UC	Check	B33

The mapping of the first property is thus done as follows:

The **Object** field is set to '<<< My input parameters'.

In the **Property** field 'Bolts' can then be chosen.

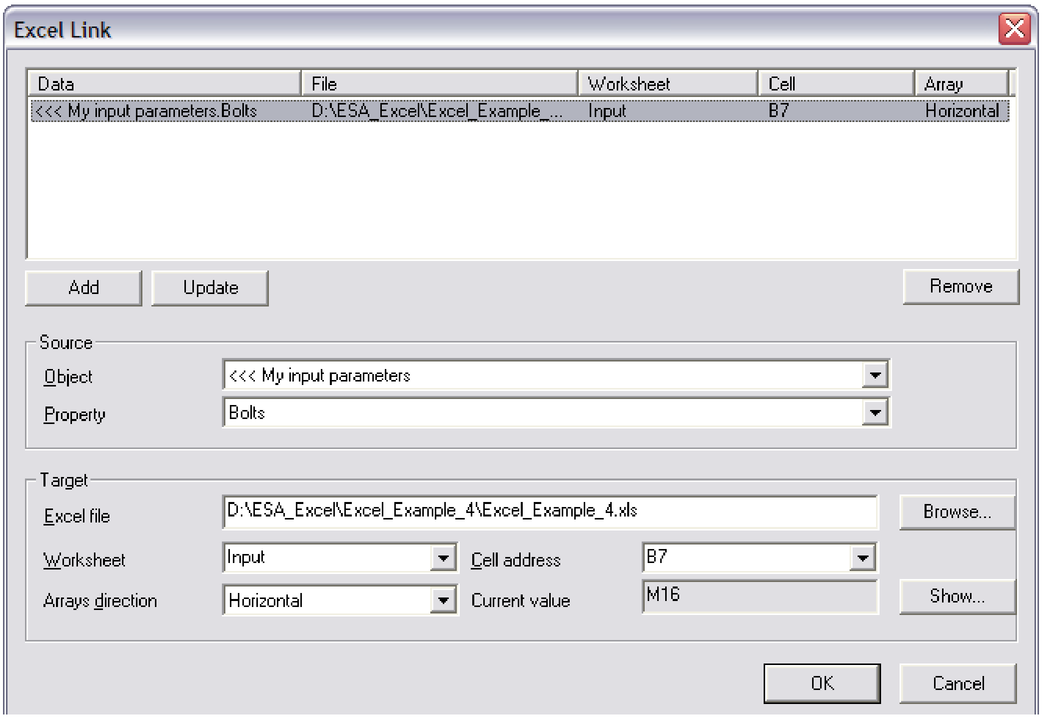
Using the **Browse** button, the file **Excel_Example_4.xls** is searched.

After the file has been specified, the **Worksheet** field contains a list of all sheets. This field is set to 'Input'.

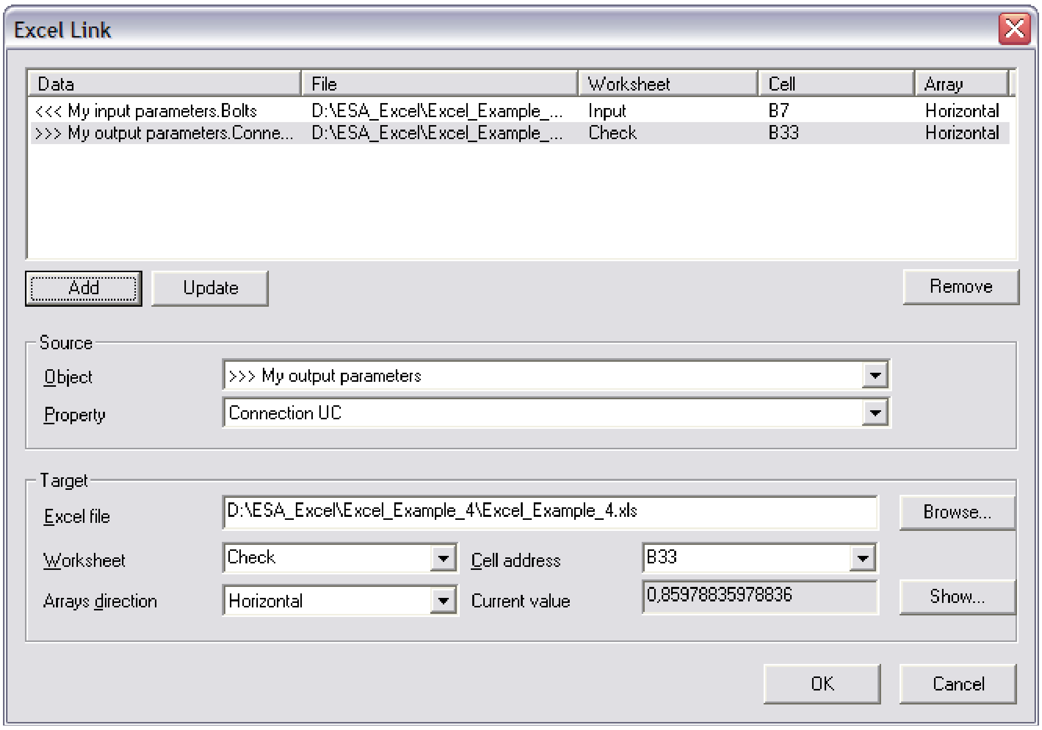
The **Arrays direction** is set to 'Horizontal'.

Finally, in the field **Cell address** the cell 'B7' is typed. Automatically the Current value field will show the current content of the cell, in this case M16.

When all input has been done, this mapping is added to the table using the **Add** button.



In the same way, the second parameter can be mapped using the above table.



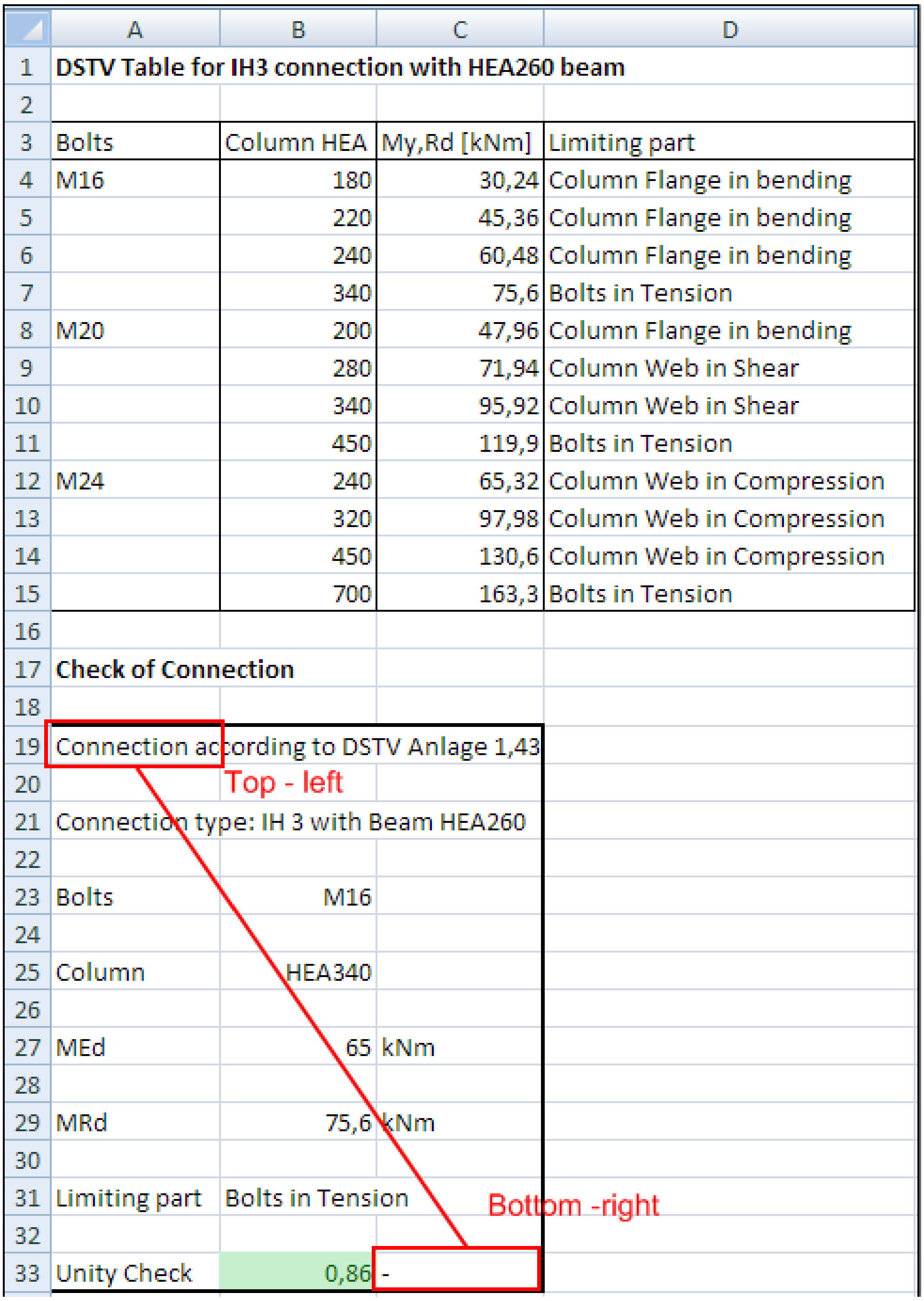
All parameters are now mapped to Excel. The final step left for the definition of the additional data is specifying a Detailed output.

Step 2.10 (Connection) Define the Detailed output

By clicking on **Setup for Detailed output**, the Detailed output dialog is opened.

In this example, one range will be defined to show the output of the connection check.

Caption	Worksheet	Top – left cell	Bottom – right cell
Connection Check	Check	A19	C33



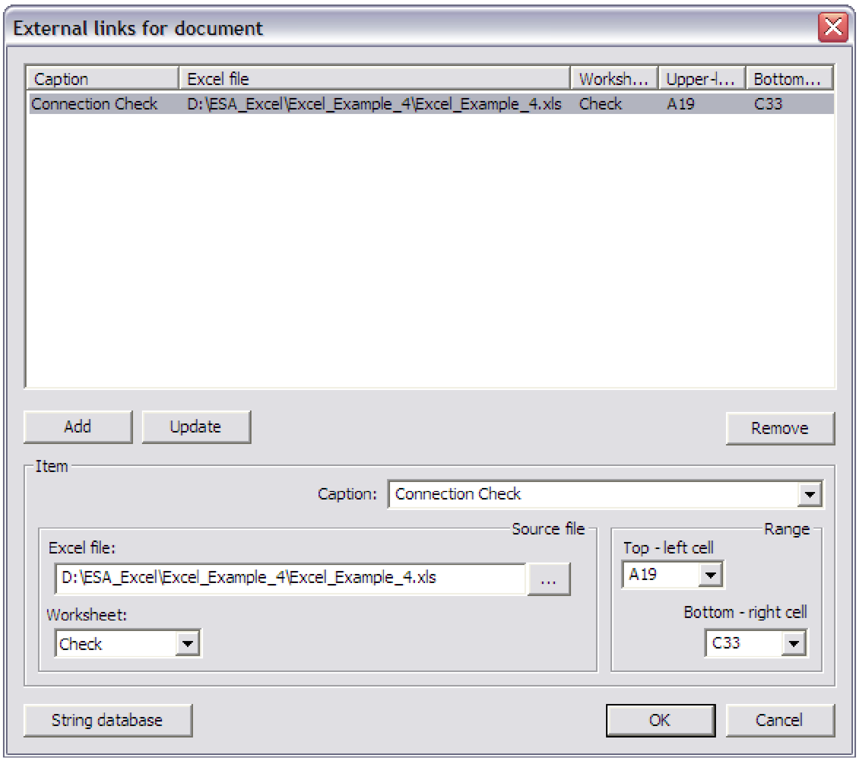
	A	B	C	D
1	DSTV Table for IH3 connection with HEA260 beam			
2				
3	Bolts	Column HEA	My,Rd [kNm]	Limiting part
4	M16	180	30,24	Column Flange in bending
5		220	45,36	Column Flange in bending
6		240	60,48	Column Flange in bending
7		340	75,6	Bolts in Tension
8	M20	200	47,96	Column Flange in bending
9		280	71,94	Column Web in Shear
10		340	95,92	Column Web in Shear
11		450	119,9	Bolts in Tension
12	M24	240	65,32	Column Web in Compression
13		320	97,98	Column Web in Compression
14		450	130,6	Column Web in Compression
15		700	163,3	Bolts in Tension
16				
17	Check of Connection			
18				
19	Connection according to DSTV Anlage 1,43			
20				
21	Connection type: IH 3 with Beam HEA260			
22				
23	Bolts	M16		
24				
25	Column	HEA340		
26				
27	MEd	65	kNm	
28				
29	MRd	75,6	kNm	
30				
31	Limiting part	Bolts in Tension		
32				
33	Unity Check	0,86	-	

In the **Caption** field the string 'Connection Check' is chosen.
In the **Excel file** field the file **Excel_Example_4.xls** is searched using the browse button.

The **Worksheet** field is set to ‘Check’.

In the **Range** group the **Top - left** cell is set as ‘A19’ and the **Bottom - right** cell as ‘C33’.

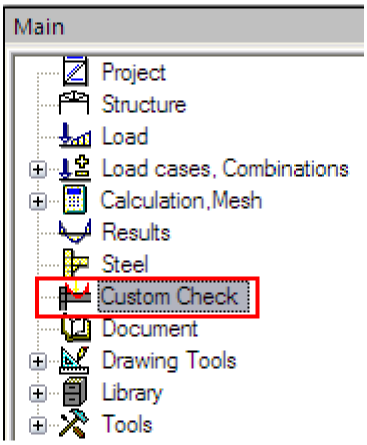
When all input has been done, the data is added to the table using the **Add** button.

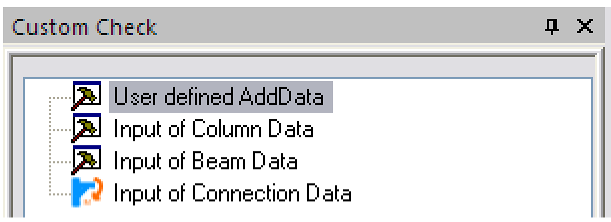


With this final step, the User Defined Additional Data has been fully inputted and the **User Defined Additional Data** Library can be closed.

Step 3: Input the User Defined Additional Data on members/nodes

After closing the **User Defined Additional Data** Library a new service will be shown in the Scia Engineer tree: **Custom Check**.

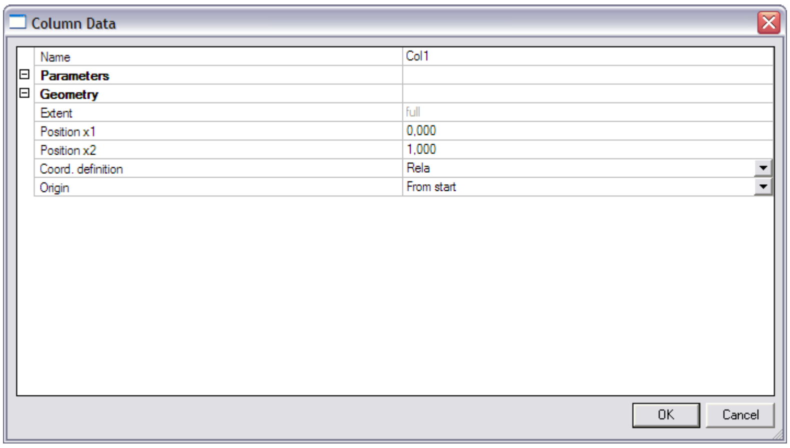




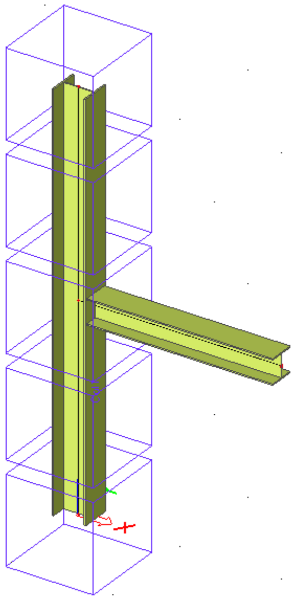
In a first step, the additional data which was defined in *Step 2*, can now be inputted on the members/nodes. Afterwards, the links will be made between the slave data and the master data.

Step 3.1 Input of additional data

First of all the column data will be inputted on the column. When double clicking on **Input of Column Data** the dialog with the properties of the data is displayed:

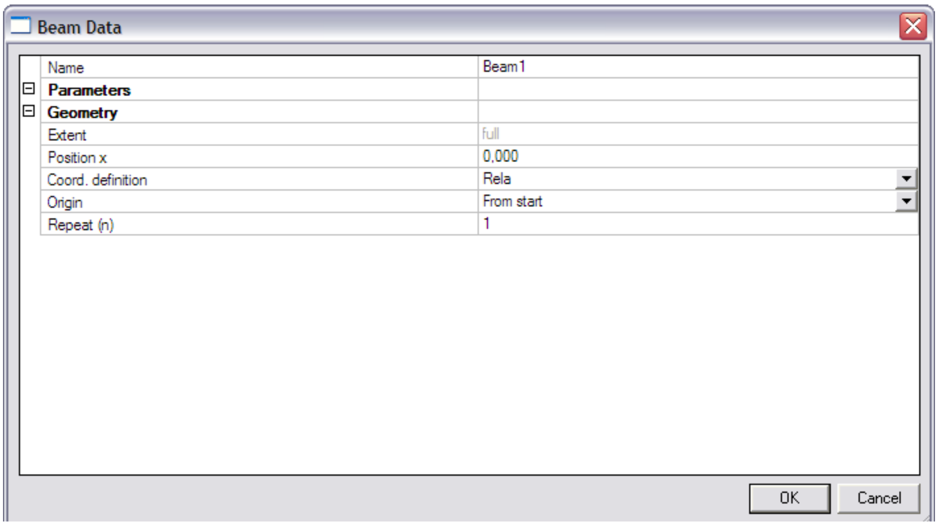


The default values of the dialog are confirmed with [**OK**] and the data is inputted on the column.



Using the default Scia Engineer scale buttons the drawing style of the additional data can be shown bigger or smaller.

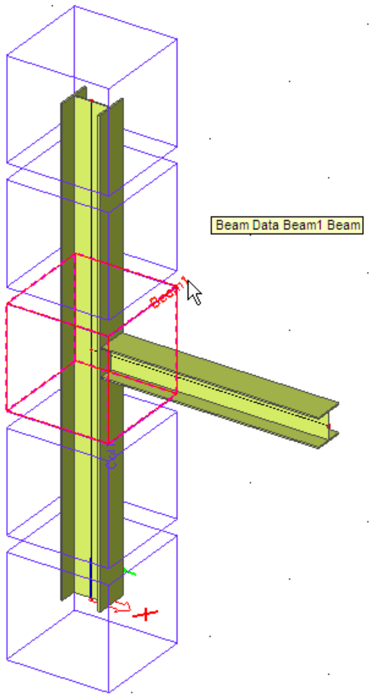
Next the beam data will be inputted on the beam. When double clicking on **Input of Beam Data** the dialog with the properties of the data is displayed:



Since the beam data have been defined as ‘Point on 1D member’ data, the position can be set in the **Position x** field.

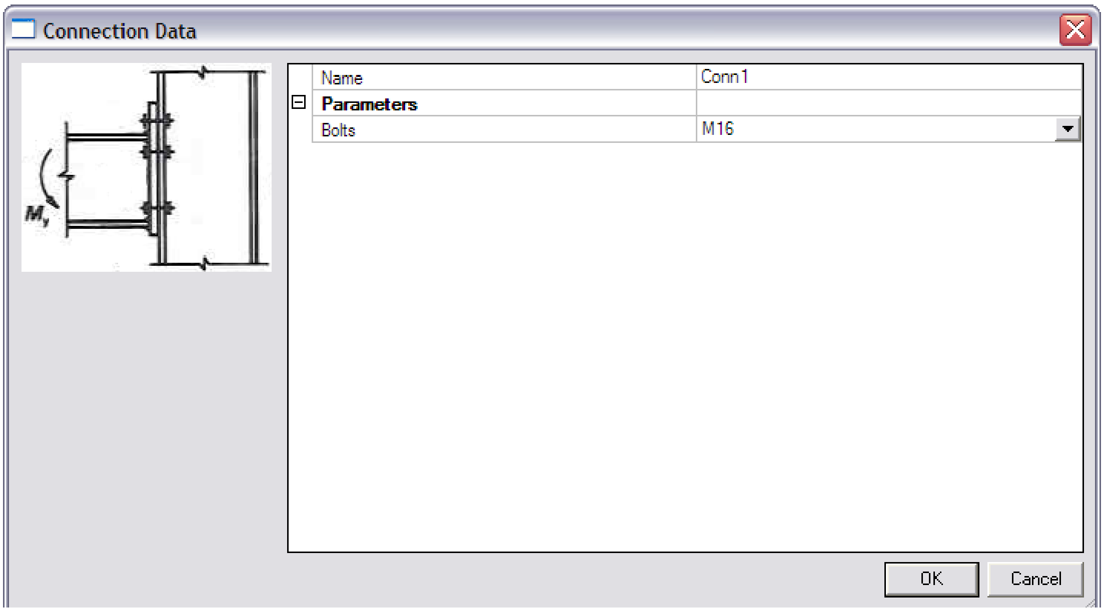
For this example, the bending moment is needed at the position between the beam and column. This position is located at the beginning of the beam and thus the **Position x** field is left to 0.

The default values of the dialog are thus confirmed with [**OK**] and the data is inputted on the beam.

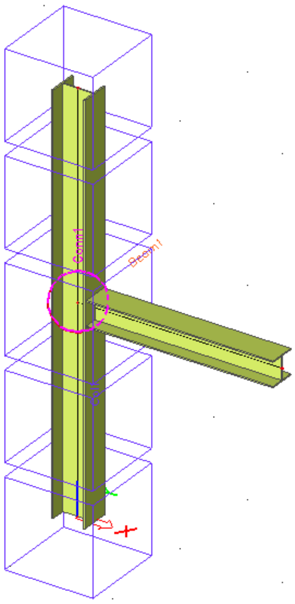


Using the default Scia Engineer view parameters the names of the additional data can be displayed.

Finally the connection data will be inputted in the node. When double clicking on **Input of Connection Data** the dialog with the properties of the data is displayed:



The default values of the dialog are confirmed with [**OK**] and the data is inputted on the node between the beam and column.



The different data has now been inputted and in the next step they can be linked together.

Step 3.2 Linking slave data to master data

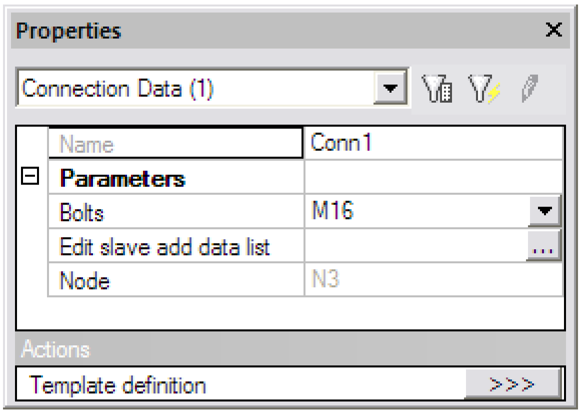
In the User Defined Additional Data library, the column and beam data have been defined as ‘slave data’.

In the previous step, ‘instances’ of this data have been inputted on different members. The final step is to link the correct instances together.

For example, if there would be two beams in this project with additional beam data on both, then it has to be specified which of the two beams should be taken into account for the connection i.e. is the connection between the column and beam 1 or between the column and beam 2.

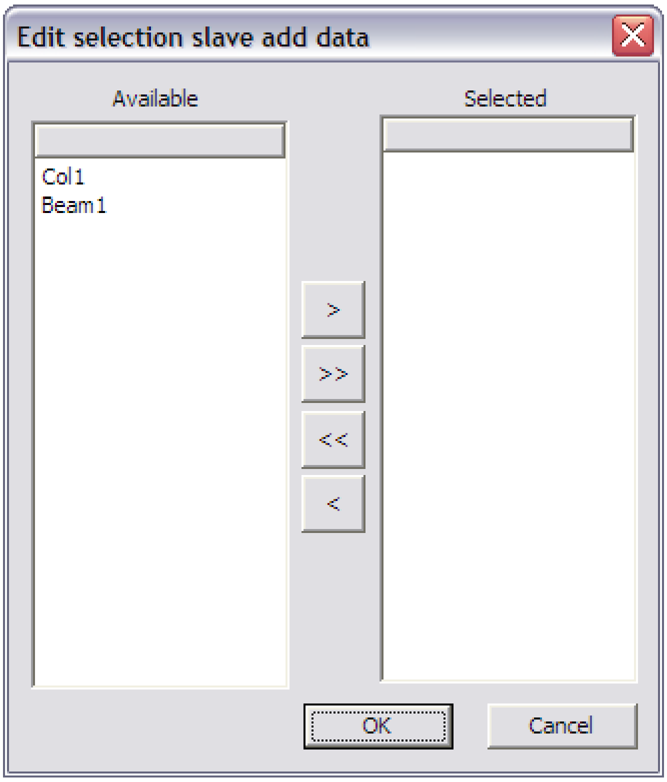
To specify which slave data instance is linked to which master data instance, the master data has to be selected.

When selecting the Connection data inputted in the node between the beam and the column, the property window shows the following:



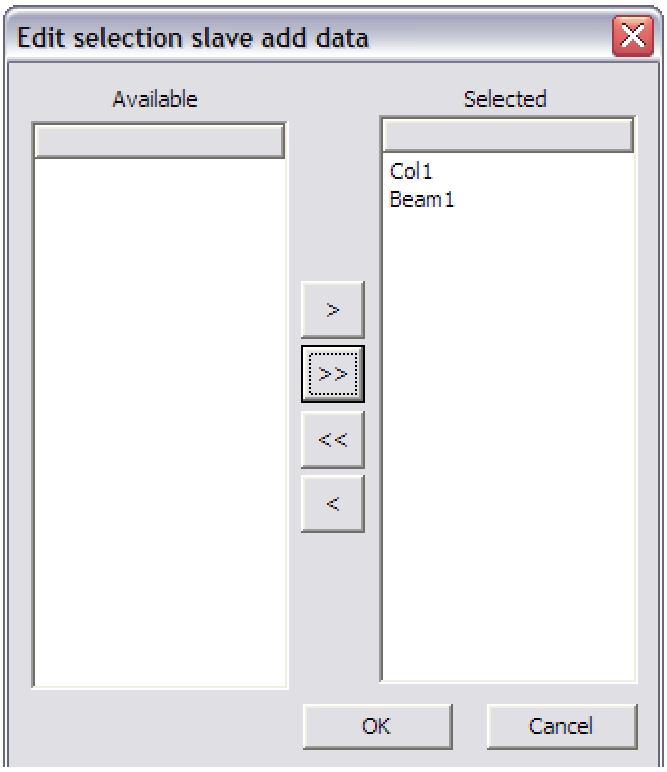
The button **Edit slave add data list** can now be used to specify which slave data instances are linked to this connection data instance.

When clicking the edit button, the following dialog is displayed:



The left column shows all **Available** slave data instances in the project. In this example the data inputted on the column 'Col1' and the data inputted on the beam 'Beam1'.

Using the arrow buttons, these instances can be added to the **Selected** column.



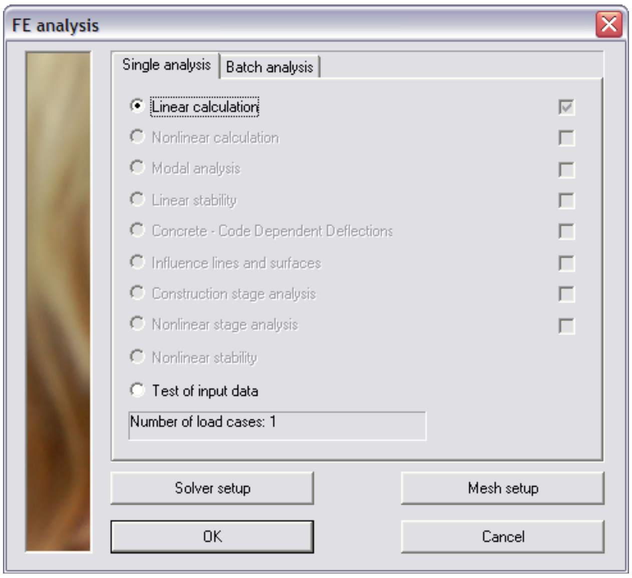
The dialog can then be closed by pressing [**OK**].

Through these steps, the required slave data have been correctly linked to the master data. During the execution of the check, the mapping defined in the master data as well as in the linked slave data will be sent to Excel. In addition, the output mapping from the master data will be read back from Excel.

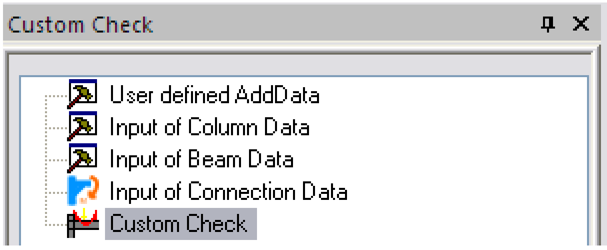
Step 4: Execute the Custom Check

In *Step 2* the additional data has been defined including the definition of the check, the mapping to Excel… In *Step 3* the additional data has been inputted and the slave data instances have been linked to the master data instance. What is left is the execution of the check.

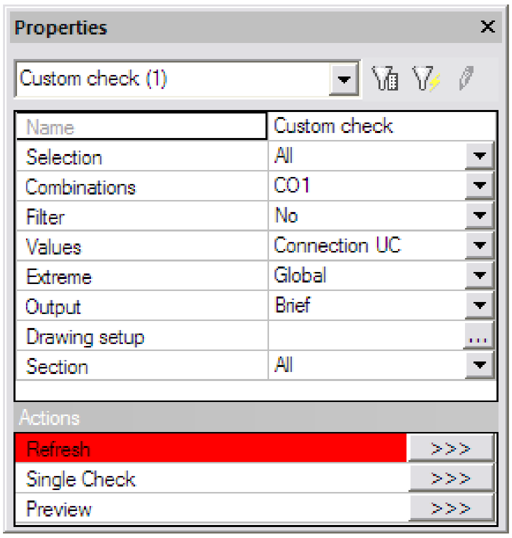
First of all the linear analysis is launched.



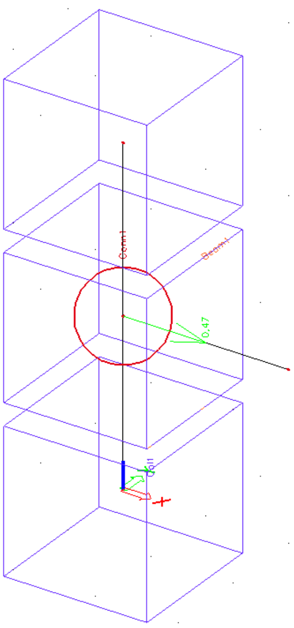
When user defined additional data was inputted and the analysis has been executed, the **Custom Check** service will show a new item: **Custom Check**.



In the property window of the check, the **Values** field contains only the output parameters of master data since the slave data do not have output parameters.



After pressing the **Refresh** action button the following check result is shown on screen:



As specified in the previous examples, the Brief preview can be shown and the check result can be added to it through the table composer. This results in the following output:

Custom check

Linear calculation, Extreme : Global
Selection : All
Combinations : CO1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_4\Excel_Example_4.xls

Type Name	Data	Css	Material	dx [m]	Case	Connection UC [-]
Connection Check	Conn1	not used	not used	0,000	CO1/1	0,47

In the same way the Detailed output can be chosen which shows the following results:

Custom check

Linear calculation, Extreme : Global
Selection : All
Combinations : CO1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_4\Excel_Example_4.xls

Type Name	Connection Check
Connection Check	Connection according to DSTV Anlage 1,43 Connection type: IH 3 with Beam HEA260 Bolts M16 Column HEA340 MEd 35,84 kNm MRd 75,60 kNm Limiting part Bolts in Tension Unity Check 0,47 -

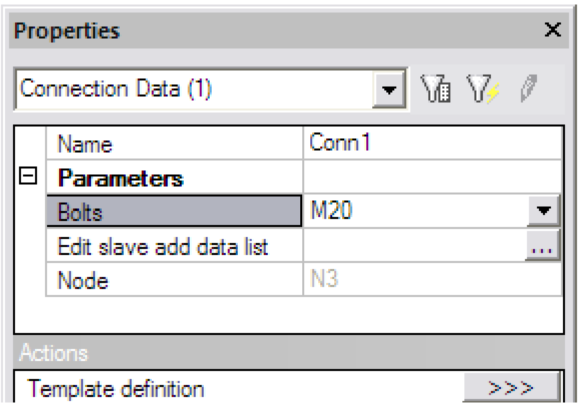
The output shows that parameters from the three user defined additional data types are combined:

- From the Column slave data the cross-section has been sent to Excel.
- From the Beam slave data the bending moment has been sent to Excel.
- From the Connection master data the Bolts have been sent to Excel.

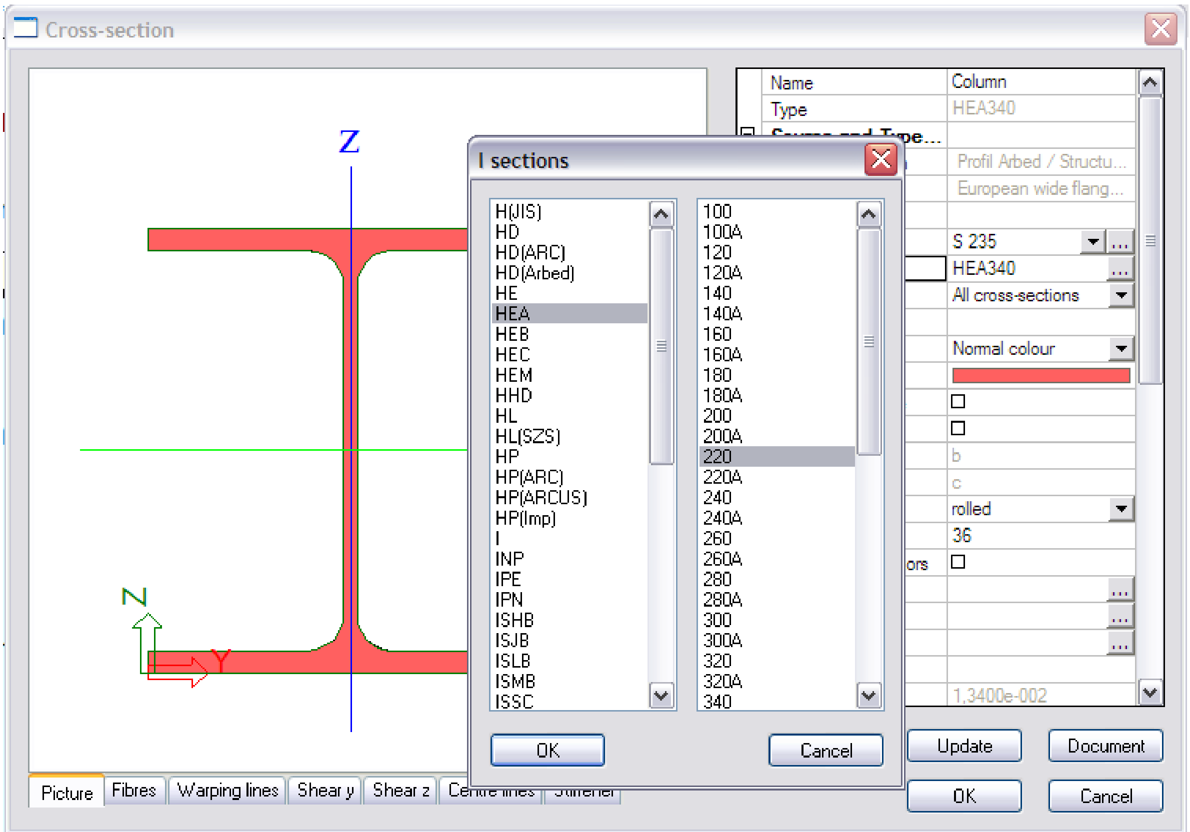
Eventually the resulting unity check defined in the Connection master data is read back from Excel.

In a next step, some changes are made to the input.

First of all the Connection data is selected and the **Bolts** are set to ‘M20’



Next, the cross-section of the column is changed to **HEA 220**.



After recalculating the project, the Detailed output now shows the following results:

Custom check

Linear calculation, Extreme : Global
Selection : All
Combinations : CO1
The check was executed according to the following user defined Excel file(s):
D:\ESA_Excel\Excel_Example_4\Excel_Example_4.xls

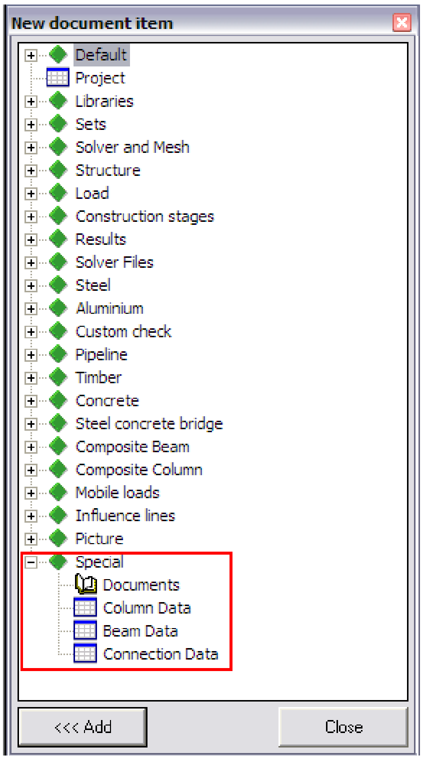
Type Name	Connection Check
Connection Check	Connection according to DSTV Anlage 1,43 Connection type: IH 3 with Beam HEA260 Bolts M20 Column HEA220 MEd 20,49 kNm MRd 47,96 kNm Limiting part Column Flange in bending Unity Check 0,43 -

It can be seen that the changes are correctly taken into account.

The check has now been executed and reviewed. To end this step, the document of Scia Engineer is examined.

In the document, the inputted User defined additional data can be inserted into the document in the same way as any other default additional data.

In the **New document item** dialog, the **Special** chapter holds the tables for all user defined additional data.



In this example, three different types of additional data have been defined and can be added to the output.

NEMETSCHEK Scia	**Project**	Excel_Example_4
	Part	-
	Description	Example 4 for Tutorial Excel Link
	Author	PVT

1. Column Data

Type Name	Name	Member	Extent	Pos x_1	Pos x_2	Coor	Orig
Column Data	Col1	B1	full	0,000	1,000	Rela	From start

2. Beam Data

Type Name	Name	Member	Extent	Pos x	Coor	Orig	Rep (n)
Beam Data	Beam1	B2	full	0,000	Rela	From start	1

3. Connection Data

Type Name	Name	Node	Bolts	Slave data
Connection Data	Conn1	N3	M20	Col1
				Beam1

As can be seen, for master data, also the linked slave data are shown. In this example, the slave data 'Col1' and 'Beam1' have been linked to master data 'Conn1'. This table provides a quick overview of linked master and slave data.

To finalize this example, one final remark is given concerning master and slave data:

> ***When removing a slave data instance which has been linked to a master data instance, the master data instance will ALSO be removed! The reasoning behind this is the following: in case slave data has been linked to master data, the slave data is required for a correct execution of the check. When the slave data instance is removed, the check for the master data instance becomes incomplete/invalid and thus the master data instance is automatically removed. This way, no accidental wrong check results will be obtained after removing slave data.***

Step 5: Save the User Defined Additional Data into a database for future use

If required, this additional data can be saved into a database for future use as illustrated in Example 1.