

Requirements Completeness Excel Macro

Release Notes

Scope

This document describes the changes that have been made to the Requirements Completeness macro for Excel.

The requirements completeness macro clarifies the requirements turmoil by taking a set of functional requirements and determining if all of the system outputs can be derived from the system inputs through an unbroken chain of requirements. The requirements start out as English language statements. The user generates a summary of the data used in each requirement. The macro checks the summary for completeness.

Version 2.0 Release 20090504

1. Switch the cursor to an hourglass while the macro is running.
2. Added graphics output. The macro will produce a Requirements Analysis diagram. The diagram is in text form suitable for processing by Graphviz.
3. Added a red fill to the graphics corresponding to the red fill in the Status column when there is an error.
4. Added the Group keyword. The Group keyword produces a subset of the Requirements Analysis diagram. The defined name is used to help name the text file. The argument list contains entries from the ID column. Those requirements and their associated data items are put in the new diagram. All of the coloring from the original diagram remains the same in the new diagram.
5. Added a new column to the left of the ID column. It is ignored by the macro. The intended use is to hold the original wording of a requirement that is being revised. The new, proposed, version goes in the Description column. The original, unedited, version goes in column A. Eventually the new version will be approved and become the official version. At that point the original can be removed.
6. Found that the ps2pdf program from GNU GhostScript can be used to format the Requirements Analysis diagram in a way so that it can print onto multiple pages. Also found that a ps2pdf could be used to produce a searchable PDF file.
7. Added autowrap for long text strings. The Description cell, in particular, can have a lot of text. The macro inserts line breaks to generate a text block with a 5:3 aspect ratio.

8. Added the AKA key word. It is not unusual for the same data item to be referred to with several different names. The AKA requirement type treats the data item in the Defined Name column and the data items in the arguments columns as a single data item. In the Requirements Analysis diagram that single data item is shown as a single ellipse with all of the names used as the label text.
9. Added the Const key word. The Const keyword defines a data name that can be used in an argument list, but cannot be used in the defined name column of a Process, Debug or Output requirement. In the Requirements Analysis diagram, it is like an Input in that it cannot have arrows going into it, but it is not forced to appear in the leftmost part of the diagram. It is useful for default data values that are mentioned in the requirements. Especially when the requirement would not otherwise have an argument. (Process requirements without arguments generate error messages.)
10. Fixed a bug with references to Struct elements.
11. Added the Define key word and the Global attribute to go with it. The Define key word takes a data item in the Defined Name column, and one or more attributes that modify that data item in the arguments columns. Currently the only attribute is the Global key word. The Global key work only affects the Requirements Analysis diagram. Normally, each data item appears only once on the diagram. For data items that are referenced by many requirements, this leads to a very complex diagram with lots of crossing lines. The Global key work causes the data item to appear as a unique bubble each time it is referenced. Graphviz will tend to place each occurrence of the data item near the requirement that references it. Consequently the arrows are shorter and there are fewer crossing lines. However, the same data item is now replicated at many disconnected places on the diagram. Each occurrence is filled with green so that it is easier to match them up.
12. Changed the Debug and Output key words so that they no longer takes arguments. This means that there must be a Process requirement to hold the arguments and a Debug or Output requirement without arguments to set the type.
13. Added the InStruct and OutStruct key words. Inputs and Outputs cannot have arguments. But they can be Structs. Previously, it would take two rows to define an Input or Output that was also a Struct. The InStruct and OutStruct key words combine those two rows into a single row. The intent is to simplify the reimplementaion of the macro in DOORS or Requisite Pro by reducing the number of “non-requirements” that have to be added to support completeness checking.
14. Added code so that Structs could be Global (unless they are Inputs or Outputs).
15. Added FirstRow, LastRow and Break key words. Originally, the macro would stop on the first row with no entry in the Type column. If there was a blank row in the middle of the spreadsheet, the macro would stop processing at that point. Now, the macro starts processing

at FirstRow and stops processing at LastRow or Break. If there is no LastRow or break in the spreadsheet, the macro still stops on the first row with no entry in the Type column.

16. Allow the Global attribute to appear in the arguments columns of Process requirements. The intent is to simplify the reimplementation in DOORS or Requisite Pro by reducing the number of “non-requirements” that have to be added to support completeness checking.
17. Fixed a bug that did funny things with double quotes in the description column.

Version 1.4 Release 20090206

1. Added the Struct key word. This allows a collection of data to be referred to with a single name. A Struct is used when either the Struct defined name is used as an argument, or ALL of the Struct elements are used as arguments. Similarly for being defined.
2. Added the Comment and “_” (underscore) key words as synonyms for Feature. Comment is used to mark some commentary. The underscore is used when the rest of the row provides some sort of formatting (i.e. the underscore is nearly invisible)

Version 1.3 Release 20090125

1. Added the Feature key word. Feature requirements are totally ignored. They can be used as a placeholder, a comment, an otherwise blank row, etc. They can be used to comment out (remove) a requirement to create a subset of an existing system.

Version 1.2 Release 20090123

1. Added “Info”, “Warning” or “Error” as the prefix to the status message.

Version 1.1 Release 20081230

1. The “Ref Only” key word was changed to Debug to better reflect the most common use.
2. Added an error message when there are arguments that have also been used as Debugs or Outputs (i.e. output only data being used as an input to a requirement).
3. Expanded the undefined data name error message to include a list of the undefined data names.
4. Changed the syntax so that a defined name could not be used as both a Process and an Output. The Output name can still be used in any number of Output requirements (i.e. output only data cannot be used as the input (argument) of any other type of requirement).
5. Changed the syntax so that a name defined as an Input could not also be defined as a Process, Debug or Output requirements. (i.e. input only data cannot be used as the defined name (output) of some other requirement).
6. Changed the error message for unused inputs to an information message. Unused inputs are not an error, but they are worth noting.
7. Added an error message for Process requirements without arguments.

Version 1.0 20081207

Initial Release