

## Project Formulas

Project has numerous fields in its database but there's always the one that you need that is not provided. You have to do it yourself; create a custom field containing a calculated formula. The results can either show data or graphics.

## Graphical Indicators

Graphical Indicators or “Traffic Lights” are very popular and show task status at a glance. You need to create a custom field first and then insert it into your table.

If you like using traffic lights then don't create the custom field from scratch every time you need it, instead copy the field you've created in this file into a new project file.

|    |    | Task Mode ▾   | RAG ▾   | Task Name ▾         | Duration ▾       | Start ▾             |
|----|---|---|---|---------------------|------------------|---------------------|
| 0  |   |    |   | ▲ <b>sample</b>     | <b>35.5 days</b> | <b>Mon 02/07/18</b> |
| 1  |   |    |   | ▲ <b>Phase 1</b>    | <b>22.5 days</b> | <b>Mon 02/07/18</b> |
| 2  | <input checked="" type="checkbox"/>   |    |    | Task 1              | 2 days           | Mon 02/07/18        |
| 3  |   |    |    | Task 2              | 3 days           | Wed 04/07/18        |
| 4  |   |   |   | Task 3              | 2 days           | Mon 09/07/18        |
| 5  | <input checked="" type="checkbox"/>   |  |  | Task 4              | 1 day            | Wed 11/07/18        |
| 6  |  |  |  | <b>Phase 1 done</b> | 0 days           | Fri 27/07/18        |
| 7  |   |  |   | ▲ <b>Phase 2</b>    | <b>20.5 days</b> | <b>Thu 12/07/18</b> |
| 8  | <input checked="" type="checkbox"/>   |  |  | Task 1              | 4 days           | Thu 12/07/18        |
| 9  |  |  |  | Task 2              | 2 days           | Tue 31/07/18        |
| 10 |  |  |  | Task 3              | 2 days           | Thu 02/08/18        |
| 11 |  |  |  | Task 4              | 3 days           | Mon 06/08/18        |
| 12 |  |  |  | <b>Phase 2 done</b> | 0 days           | Thu 09/08/18        |

In the picture above we are implementing a simple % Complete convention. Tasks are marked as 0% if they have not been started, 50% if they are work-in-progress; started but not finished, and 100% if completed.

So there's three possible states: Zero is Red, WIP is Amber and Completed is Green.

The process is as follows:

1. Make the custom field and name it.
2. Create a formula that works on the % Complete field and returns 1 if it's 100, 2 if it's 50 and 3 if it's 0.
3. Assign the traffic lights: Green for 1, Amber for 2 and Red for 3.
4. Show the Custom field in the table.

## 1. Create the Custom Field



Click the **Project** tab and then **Custom Fields**.

From the **Type** (1) drop down list, choose **Number**. From the **Field** options, select **Task** (2)

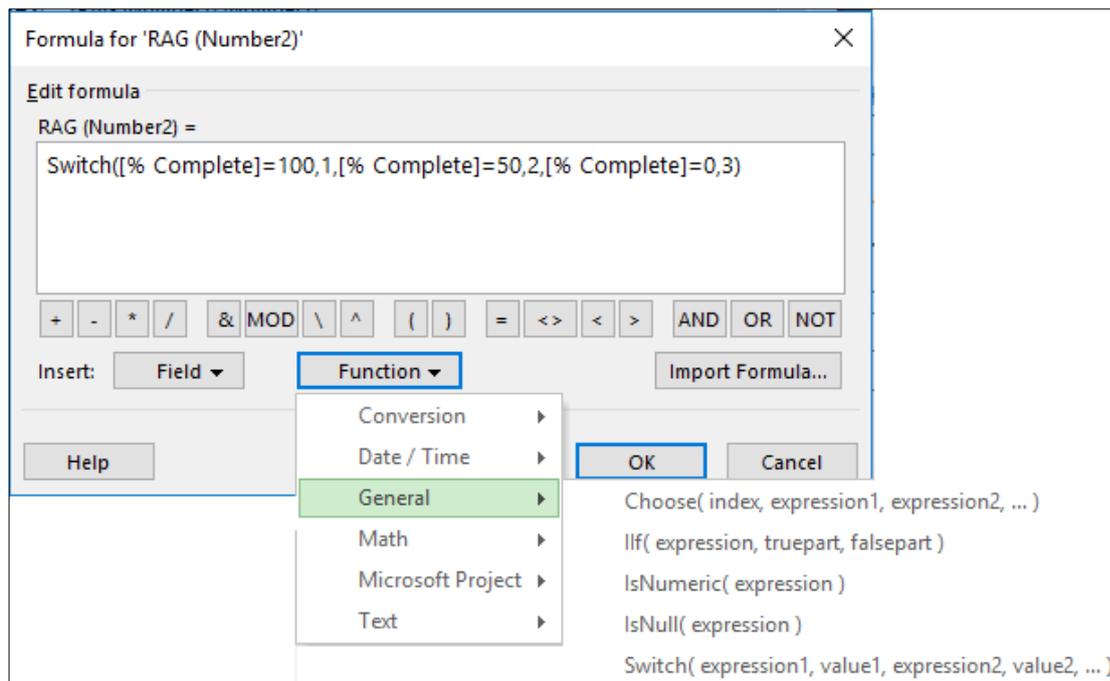
From the **Field** (3) list, select the field you want to define (Number1, Number2, etc.).

Click the **Rename** (4) button to assign a more meaningful name to the custom field. In this case, it's named RAG.

A screenshot of the 'Custom Fields' dialog box. The 'Field' section has 'Task' selected (2) and 'Type' set to 'Number' (1). A list of fields includes 'RAG (Number1)' (3), 'Number2', 'Number3', 'Number4', 'Number5', 'Number6', and 'Number7'. The 'Rename...' button (4) is highlighted. The 'Custom attributes' section has 'None' selected. The 'Calculation for task and group summary rows' section has 'None' selected. The 'Calculation for assignment rows' section has 'None' selected. The 'Values to display' section has 'Data' selected (6). The 'OK' button is highlighted.

## 2. Create the Formula

Click the **Formula (5)** button and enter the formula. During this process, click OK to any message boxes asking you to discard field values and replace with the formula.



Use the drop down lists under the **Field** and **Function** buttons to help you enter the formula rather than type everything in from scratch. The **% Complete** field is in the **Number** category in the **Field** list and the **Switch** function is in the **General** category in the **Function** list. The formula is as follows:

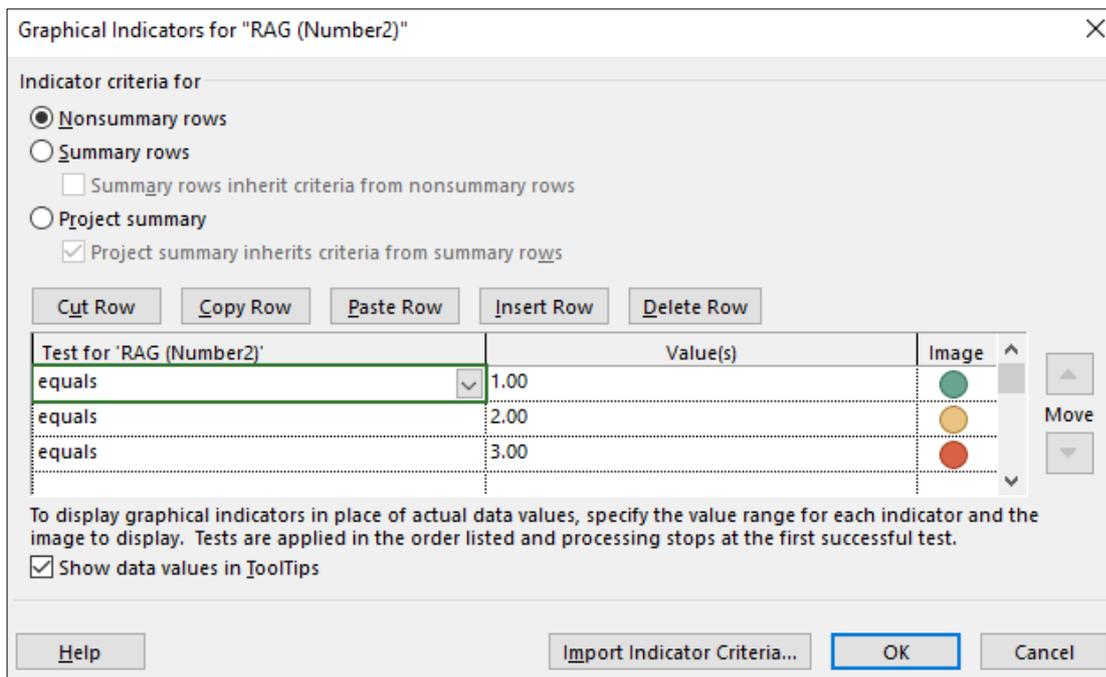
```
Switch([% Compete]=100,1, [% Compete]=50,2, [% Compete]=0,3)
```

You could easily extend this by adding further expressions and values; maybe have 0%, 25%, 50%, 75% or 100%.

```
Switch([% Compete]=0,1,[% Compete]=25,2,[% Compete]=50,3,[% Compete]=75,4,  
[% Compete]=100,5)
```

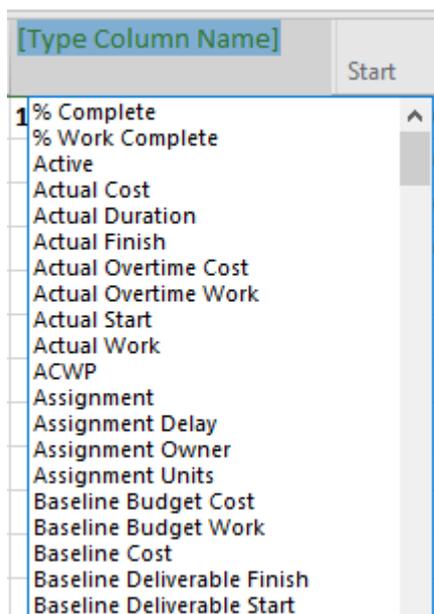
### 3. Assign the Traffic lights

Click the **Graphical Indicators** (6) button, enter a simple test like, equals 1 and then select the required image from the Image list.



### 4. Show the Custom field in the table

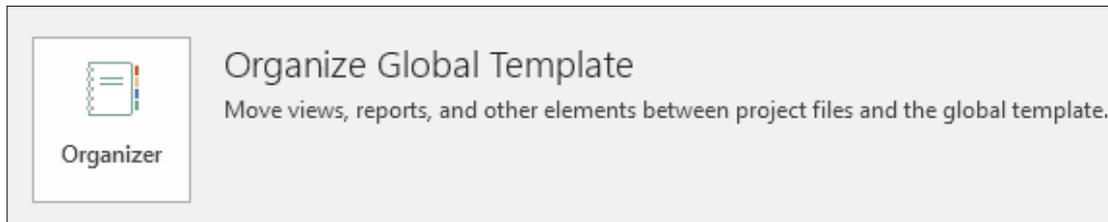
Like inserting any other field you show the RAG images in your current table by right-clicking on any column heading and choosing **Insert Column** from the shortcut menu.



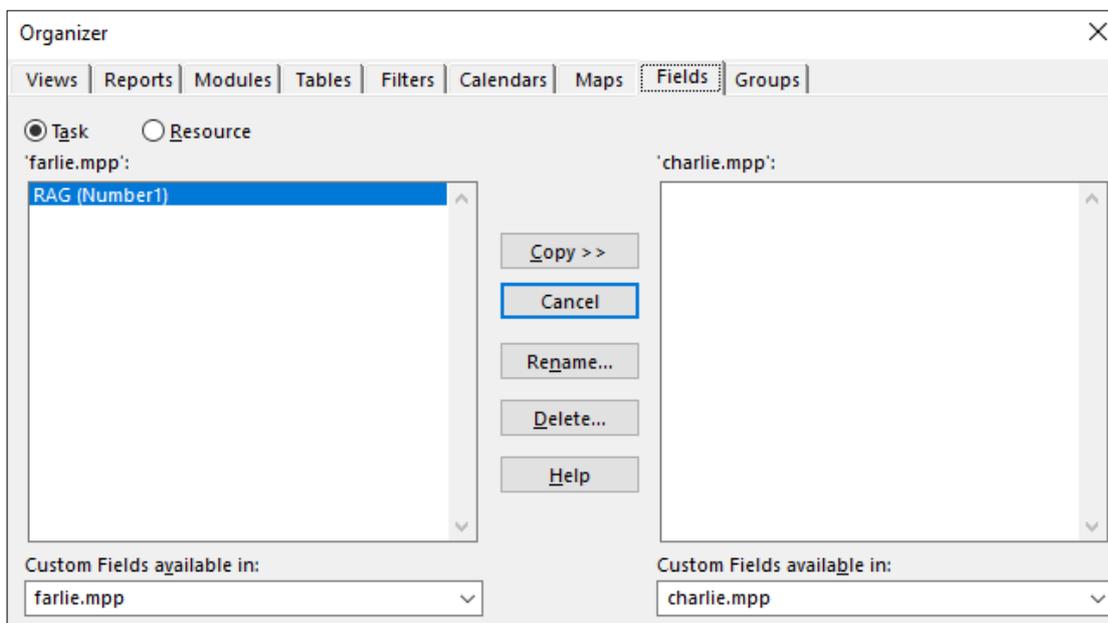
You can type in its original name (like Number2) or the name that you assigned (like RAG)

## Copying a Custom Field

When you've created a custom field in one file you do not want to have to go through the same thing again when you start another project plan. Keep the file where you created the custom field open and then open the file where you want to copy the field. Click the **File** tab and then click **Organize Global Template** in the **Info** section.



This is not the only method of copying fields but it is the best method to know as it is the generic method of sharing resources between files.



Click the **Fields** tab in the **Organizer**. Look at the drop down lists at the very bottom of the box, the one on the left hand side usually reads Global.MPT. Change this so that one file is the one that contains the custom field and the other is the one where you want to copy it. Click the **Copy** button.

If you want to have the field available in all future project files, copy it to the global template file, Global.MPT.

## Useful Project Formulas

### The task is missing either a predecessor or successor

The best practice for network logic is that every task has a predecessor or successor, except for the first task or the last task.

*Type of Field: Flag*

```
IIf(([Predecessors]=""" Or [Successors]="""),True,False)
```

### Task Needs To Be Baselined

A baseline allows you to compare a task's current Start and Finish dates to the originally planned Start and Finish Dates.

*Type of Field: Flag*

```
[Baseline Start] = ProjDateValue("NA")
```

### Task Start Date or Finish Date Needs To Be Updated

Makes it easy to see when a task's Start is earlier than the current date, but still is 0% Complete, or a task's Finish is earlier than the current date, but less than 100% complete.

*Type of Field: Text*

```
Switch(Date(>[Start] And [% Complete]=0,"Should Have Started", Date(<=[Finish] And [% Complete] <100,"Should Have Finished", Date (<=[Start],"", Date(<=[Finish],""))
```

### Non-Summary Tasks That Do Not Have Dependencies

A task that does not have any predecessors or successors will not show how it effects other tasks.

*Type of Field: Flag*

```
IIf(([Predecessors]=""" Or [Successors]=""" ) And [Summary]=False,True,False)
```

### Summary Tasks That Have Dependencies

Obscures the detail of the critical path; you have less visibility as to what is driving the detailed tasks.

*Type of Field: Flag*

```
IIf(([Predecessors]<>"" Or [Successors]<>"" ) And [Summary]=True,True,False)
```

### Summary Tasks That Have Resources Assigned

It is considered a best practice to not assign resources to summary tasks, and this complicates a task's rollup behaviour.

*Type of Field: Flag*

```
IIf([Summary]=True And [Resource Names]<>"" ,True,False)
```

## Tasks That Do Not Have Resources Assigned

Without a resource assigned, you will not know who will be performing a task, or how much Work that task will take.

*Type of Field: Flag*

**IIf([Summary]=False And [Milestone]=False and [Resource Names]="",True,False)**

## Tasks That Have Negative Total Slack

Tasks with negative Total Slack are not meeting the targeted schedule date.

*Type of Field: Flag*

**IIf([Total Slack]<0,True,False)**

## Unfavourable Variance on Critical Path

If a task Finish date is scheduled later than the Baseline Finish date and the task is on the project's critical path, it is negatively impacting the Finish date of the project.

*Type of Field: Flag*

**IIf([Finish Variance]>0 And [Critical]=True,True,False)**

## Difference between 2 dates in elapsed days

*Type of Field: Number*

**DateDiff("d",[Baseline Finish], [Project Finish])**

## Difference between 2 dates in working minutes or days

*Type of Field: Number*

**ProjDateDiff([Baseline Finish], [Project Finish],[Project Calendar])**

The ProjDateDiff function has no date interval parameter (unlike DateDiff) and the result is calculated in minutes. To get the difference in days either divide the result by 480 (the number of minutes in an 8hr day) or use the Minutes Per Day field. Like this:

**ProjDateDiff([Baseline Finish], [Project Finish],[Project Calendar])/[Minutes Per Day]**

## Rounding

The working days calculation shown above will show decimal values for days after the division. Although it's not documented, Project does support the Round function so either use the function to round to zero decimal places or use the Format function to apply a round numbers format. Either formula will show round days:

**Round(ProjDateDiff([Baseline Finish],[Project Finish],[Project Calendar])/[Minutes Per Day],0)**

**Format(ProjDateDiff([Baseline Finish], [Project Finish],[Project Calendar])/[Minutes Per Day],"0")**

### [Link to predecessor task in another project.](#)

To link a task in the current project to a predecessor task in another project you enter an expression into the **Predecessor** field as follows:

File Name \ID

File name is the project's file name including the extension and ID is the index value of the task (number down the left). For example, the following expression references task ID 25 in the Charlie.mpp file:

**Charlie.mpp\25**