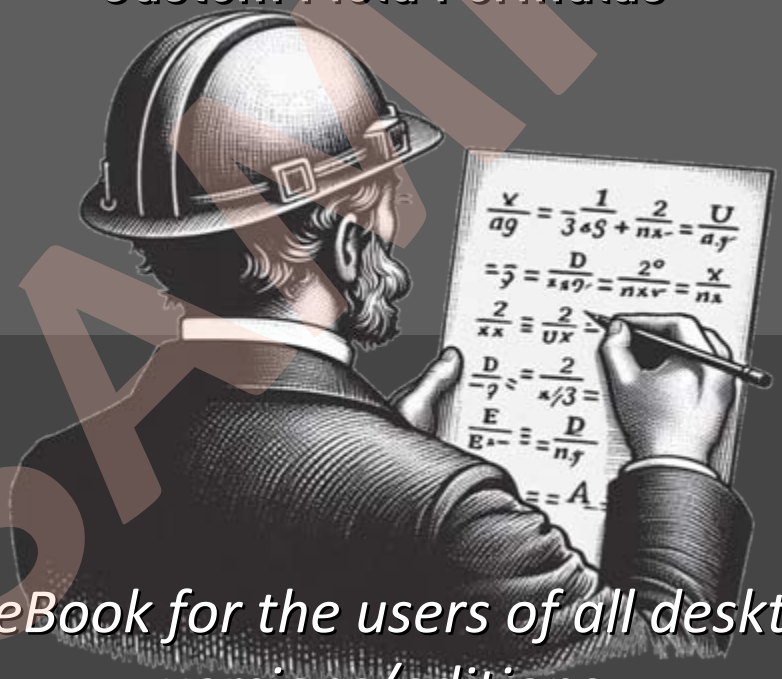
Articles & Tips Collection

on Microsoft® Office Project's Custom Field Formulas



An eBook for the users of all desktop versions/editions

Ismet Kocaman

SAMPLE

Articles & Tips Collection

on Microsoft® Office Project's
Custom Field Formulas

An eBook for the users of all desktop versions/editions

Ismet Kocaman

SAMPLE

© Ismet Kocaman

Notice of Rights

All rights reserved. No part of this eBook may be reproduced, stored in a retrieval system or transmitted in any form or by any means, without the prior written permission of the author.

Notice of Liability

Every effort has been made to ensure the accuracy of the information herein. However, the information contained in this eBook is provided without warranty, either expressed or implied. The author will not be held liable for any damages to be caused either directly or indirectly by the instructions contained in this eBook, or by the application software described herein. The author provides examples for demonstration only, without warranty either expressed or implied.

Trademark Notice

Microsoft is a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries. All other trademarks mentioned herein are the property of their respective owners. The author has no affiliation with Microsoft Corporation. Screen captures were reprinted with authorization from Microsoft Corporation. This document is not a product of Microsoft Corporation.

SAMPLE

About the Author

Ismet Kocaman is a Management Consultant, Project Management Consultant, Technical Project Manager and a Mechanical Engineer with over 20 years of experience in the manufacturing sector.

He is currently providing management consultancy to the companies in the manufacturing sector on system improvement projects and technical projects. He also conducts training seminars for engineers on Project Management and MS Project with focus on the project management process in the manufacturing environment.

He is a Project Management Professional (PMP) and holds several Microsoft® certifications on MS Project.

Visit the website at http://www.ismetkocaman.com for more information on the author's profile and other eBooks

SAMPLE

Introduction

Even though MS Project desktop contains plenty of the data fields in various types and categories presenting the planned, scheduled and actual project data interpreted in many different ways to the users, one might still need some custom information that is not directly available in the existing fields.

MS Project desktop (that is, the standalone desktop versions of MS Project's Professional/Standard editions) is a highly customizable application such that it enables you to build your own formulas in the custom fields in order to perform further operations on the existing project data to produce some new project data needed. This feature was added to the product with the version, MS Project 2000.

Enabling users to process a project's task and resource data in further ways through the expressions entered as formulas to various types of custom fields is one of the distinguishing features of MS Project desktop.

You can visit the sample content from the eBook Mastering Custom Field Formulas in MS Project desktop in the website in order to learn more about the benefit of calculating custom project data by using the custom field formulas and some other useful applications of the custom field formulas.

The eBook Articles & Tips Collection on MS Project's Custom Field Formulas presents a collection of selected articles and tips on the custom field formulas that have been published in various platforms and many MS Project-related websites, and posted in many MS Project-related forums, by the author since 2010.

While the eBook Mastering Custom Field Formulas in MS Project desktop dives into this feature, the articles and tips in this eBook show various ways to make use of the custom field formulas while working on the project data. Any professionals who currently use any desktop versions/editions of MS Project, that supports the feature, as a project management application while managing projects may benefit from the content presented in this eBook.

SAMPLE

Each article/tip in the eBook was tested with the latest updated version of the desktop product at the time of producing the content. But the reader must be aware of the fact that any new version of the desktop product released may include minor and/or major changes to any features, without any public announcement. Therefore, always check the changes in the features across the versions if any feature described in any article/tip functions differently on the version of the desktop product that you are currently running ■

SAMPLE

List of Articles and Tips

Articles

Using Author and Subject Fields to Pass Data to Formulas 9
Understanding Assignment Calculations 12
Quick and Simple Way to See the Resources with Assignments Starting Next Week 28
Listing Number of Task Assignments for Resources 33
How to Find the Number of Days Between Task Finish Dates and the Fiscal Year's End 40
How to Create a Fiscal Year to Date Filter 45
- Developing the formula 49

Handling Default Start/End Times in Project Schedules with Task Calendars 51
Graphical Reports / Burndown Charts 54
- Introduction 55
- The Report Tab: Commands 56
  - Customizing the Ribbon for the Report Tab Commands 58
- Contextual Tabs: Report Tools, Chart Tools, Table Tools, Drawing Tools and Picture Tools 62
- Fields Referenced in Task Burndown Chart 70
  - How Does MS Project Calculate Numbers at Task Level ? 70
    - Remaining Tasks field at the task level 70
    - Remaining Actual Tasks field at the task level 72

SAMPLE

BaselineX Remaining Tasks field at the task level 73
How Does MS Project Calculate Total Numbers at Summary Task Levels ? 75
How Does MS Project Calculate the Remaining Tasks Field's Data at Summary Task Levels ? 75
How Does MS Project Calculate the Remaining Actual Tasks Field's Data at the Summary Task Level ? 80
How Does MS Project Calculate the BaselineX Remaining Tasks Field's Data at the Summary Task Level ? 82
Fields Referenced in Work Burndown Chart 86
Building a Custom Task Burndown Chart 101
Scenario 1/3: Project on Track 109
Scenario 2/3: Project Ahead of Schedule 116
Scenario 3/3: Project Behind Schedule 124
Adding a Trendline to the Chart 132
Forecasting Progress by Using a Trendline 138
Forecasting Progress by Using a Trendline - Using Visual Reports 147
Completing the Project 158
Calculating Additional Number-of-Tasks Data by Using Custom Field Formulas 161
Building a Custom Work Burndown Chart 174
Scenario 1/3: Team's Progress on Schedule 177
Rescheduling the Work 178
Scenario 2/3: Progress Ahead of Schedule 180
Scenario 3/3: Progress Behind Schedule 186
Completing the Project 189
Displaying Task and Resource Information in a Graphical Report 192
Task Custom Field Formulas: 192

SAMPLE

Resource Custom Field Formulas: 193
Custom Task Group: 195
Custom Resource Group: 196
A Report Listing Overallocated Resources 202
Displaying Project's Start and Finish Dates in Graphical Reports 208
Method 1: Using the existing tables for the project start and finish dates 208
Method 2: Creating new tables for the project start and finish dates 209
Creating a Custom Graphical Report: Work Distribution Report with Hours and Percentages 211
How to Create a Task Table with Indicators Column in a Graphical Report 217
Is it a Whole-Day Duration ? 221
How does the CBool function interpret the numeric value returned from the subtraction operation ? 222
How Does the "Milestones Due This Month" Filter Work ? 223
Filtering Project Data - Filter versus Formula 226
Filtering for Dates 231
Filtering for Durations 236
Filtering Project Data - Handling "the entry is not valid" Error in Custom Task Filters 237
Filtering manually scheduled tasks based on text information only 238
Filtering both manually scheduled and auto-scheduled tasks based on valid dates 239
Using the custom date fields in filters without the "entry is not valid" warning 241
Working with the duration filters 242
Change a Task Bar's Style Based On the Value Selected from a Lookup Table 245

SAMPLE

Some formulas performing rounding on the positive (or negative) dollar amounts 249
Handling the Planning Wizard's Settings 251
Display the Status Date Just Below the Timescale 259
Project Options Dialog Box - What we need to know about the settings 263
Scope of a Setting 263
Settings that affect MS Project's calculations 269
Does the change in a setting apply to the existing project data ? 270
How to Calculate the Date of a Particular Day in a Month 277
Calculating the date of the third Wednesday in a month specified by any date 281
How To Calculate Date of the Next Particular Weekday Based on any Date 283
Calculating the date of the next (or previous) Tuesday if the current date is not a Tuesday 283
An Example: Check Whether Tasks' Start Dates Fall Into a Quarter Starting in a Particular Weekday 285
Calculating the number of the week in which a task starts relative to the start of a quarter 287
Improving the Visibility of the Group-By Summary Bars 290
How To Display the Total Cost for the Selected Task Assignments of a Work Resource 292
Calculating Material Resource Costs Separately 297
Testing the Method 298
Calculating Material Resource Costs Separately 302
Displaying Per-Use Costs Separately 307
Creating a Graphical Report to Display Per-Use Costs for Resources 316
Updating Per-Use Costs 318

SAMPLE

A Formula for Negative Cost Representation 322
Calculating Fixed Costs for Resources: Cost/Assignment and Cost/Calendar Day 326
Calculating fixed costs based on the number of task assignments 326
Calculating fixed costs for the work resources based on the total task assignment duration 334
Bottom-Up Planning with Manual Scheduling 336
Switching from Manual Scheduling to Auto Scheduling 345
Using the Task Inspector 348
Status and Task Mode Indicator Formulas 350
Status Indicator Formula Checking the Status Date 350
An Alternative Task Mode Indicator Formula 352
Search on the WBS field by Using Special Operators 358
Using Special Operators: Like and Between…And 358
Grouping task data based on WBS code 363
Grouping task data based on WBS level 367
Grouping assignment data based on WBS codes 370
How to Handle Durations as Seconds in MS Project 374
Example - Scheduling with Durations as Whole Seconds 379
What is the difference between %Complete and CompleteThrough ? 386
Introduction 386
What is the difference between %Complete and CompleteThrough ? 387
Why Not Use the Stop Field To Draw the Task Progress Bars ? 392
Introduction 392

SAMPLE

Scenario #1: A Task with a Single Work Resource Assignment 392
Scenario #2: A Task with a Multiple Work Resource Assignment 397
How Does MS Project Calculate Progress on Summary Tasks ? 402
Introduction 402
Using the CompleteThrough Date to Display Progress on a Summary Task 403
Using the Summary Progress Date to Display the Progress on a Summary Task 411
How to show progress on summary tasks 421
Displaying Summary Durations in Elapsed Units 422
VBA Methods similar to the Built-in Functions in the Project category 432
DateDifference method 433
DateAdd and DateSubtract methods 439
DurationFormat method 442
DurationValue method 443
DateFormat method 445
Project VBA Macro: Listing Resources' Overallocated Dates and Hours in MS Excel 447
Project VBA Macro: Finding All the 53rd Weeks in MS Project 451

Tips

Tips on MS Project 455
Using XOR in Formulas 455
About the Inactive Tasks in MS Project Standard 456
Keyboard Shortcuts - Using the Function Key F2 While Inputting Information into a Cell 457

SAMPLE

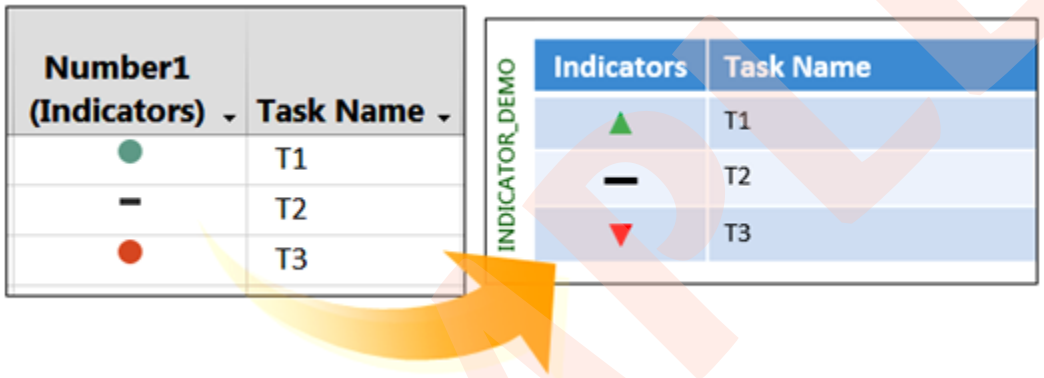
Using the Mouse to Indent or Outdent Task(s) 458
Getting a List of All Custom Field Formulas 459
Just For Fun - Flag Field with Emoticons 464
Defining a working time pattern (recurring) for a task that can only be performed on <the second Thursday of every Month> and <between 08:00 pm and 09:00 pm> 468
How to Display Baseline Milestone Tasks on the Timeline 470
A Simple Formula to Check to See Whether a Resource's Calendar Differs from the Project Calendar 473
How to Use a Custom Image to Represent a Key Milestone in the Gantt graph 475
How to Format Individual Gantt Chart Bars, Based on a Resource Assigned 476
A Simple RAG Indicator without a Formula 477
How to Restore a Custom Table Modified to the Original One Automatically 480
About the Column Titles in the Tables not matching the Field Names 483
About Grouping the Project Data in a Filter Applied Table 484
Formula to Find Tasks with Four or More Predecessors 486
String Addition Operation in a Custom Field Formula 487
How to Create a "Tasks Due Today" Filter that Automatically Gets the Current Date 488
How to Display Dates in Two Different Formats in a Table 490
How to Use the Task Calendars to Schedule the Work of Two Teams Based on 12-hour Shifts 491
How to Remove a Soft Constraint Quickly 492
How to Split a Task by Using the Mouse 493
Some Date-Related Custom Field Formulas for Month, Year, Year to Date 494
Show Dates on the Gantt Bars 495
About the Logic in the Expressions of the Custom Field Formulas 496

SAMPLE

Using the Flag Field <Summary> 497
The Fastest Way to Split a View in Later Versions of MS Project 498
How to Display Work in Two Different Units 500

SAMPLE

How to Create a Task Table with Indicators Column in a Graphical Report



In this example, we will create a custom graphical report containing a task table that displays graphical indicators based on the values that a custom number field holds for the tasks in the active project plan (see the picture above). It takes about 15 to 20 minutes to create such a report, but that will be a one-time job. Then you can use the report for all the project plan files in your desktop. Let us start with creating the custom graphical report; follow the steps below:

- Open a blank project plan. Do not save it.
- Create three tasks just by entering task names. We will populate the custom number field with values entered by typing in, but you can enter any formula to that field, for example, a formula interpreting variances according to certain thresholds in the schedule. Enter the values 1, 0 and -1 to the **Number1** field for the tasks.

SAMPLE

- Enter the following formulas to the task custom text fields **Text27**, **Text28**, **Text29** and **Text30**, respectively:

  Text27: **Left([Name], 70)**
  Text28: **iif([Number1] > 0, "▲", "")**
  Text29: **iif([Number1] = 0, "—", "")**
  Text30: **iif([Number1] < 0, "▼", "")**

- Insert those fields to the table in order to verify that formulas are working correctly.
- Open a blank report named "INDICATOR_DEMO" (**REPORT** | **New Report** | <select Blank> and enter the report name).
- In the report, create three tables as follows and set [Outline Level] to All Subtasks:

  Table #1: a table with the fields **Text27**, **Text28**

  Rename **Text27** column as "Task Name" and reposition it as the 2nd column, enter a proper width, such as 7 inches, in order to eliminate any text wrapping of the task names trimmed to 70 characters. Text wrapping will destroy the alignment of the rows.

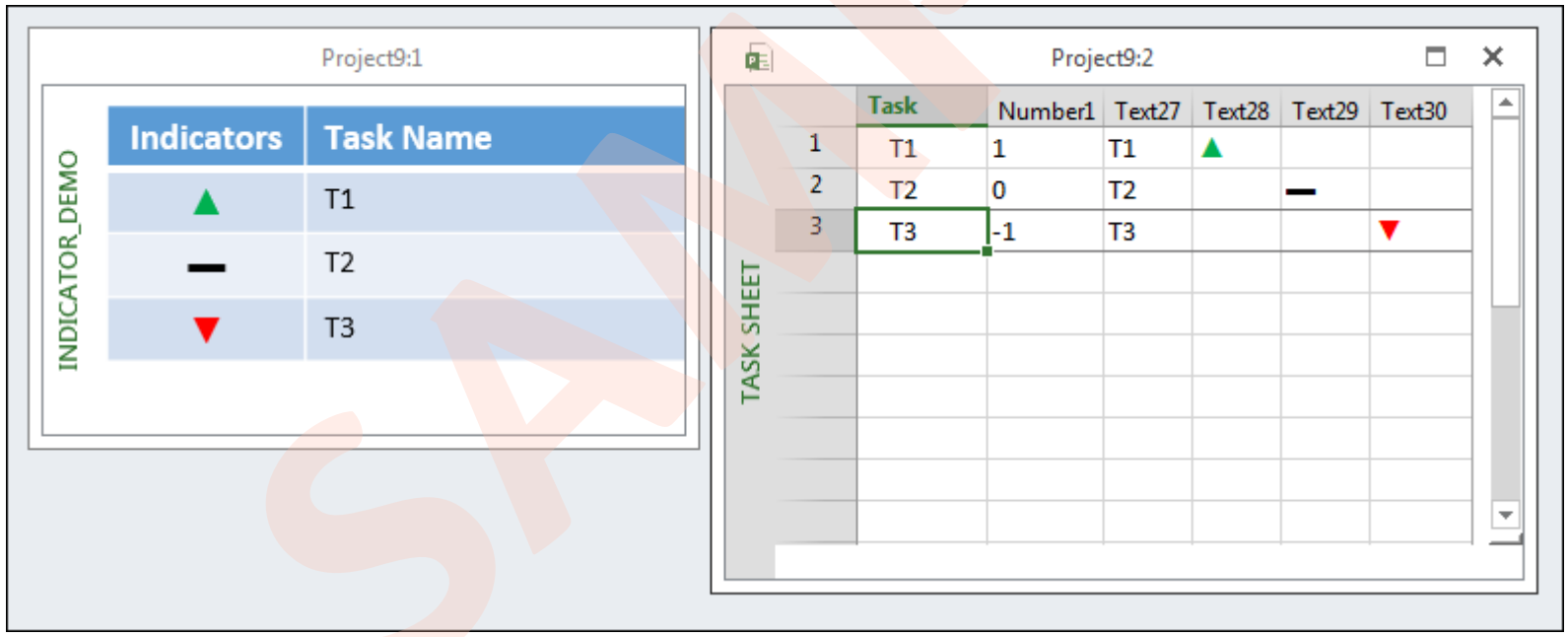
  Rename **Text28** column as "Indicators" (it is now the 1st column), align its content including the header to center, set the font color to **green** for the content, double-click the right border for best fitting.

  Table #2: a table containing the field **Text29**
  Table #3: a table containing the field **Text30**

SAMPLE

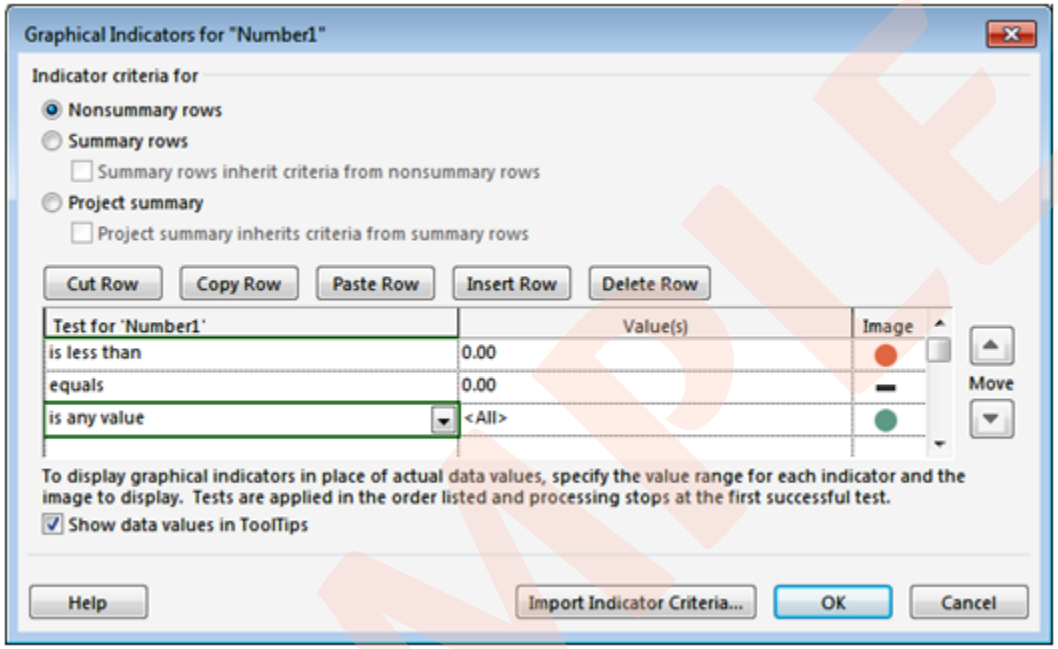
Align content to center in both tables. Set the width of both tables to the width of the Indicators column in Table #1. In both tables; remove the headers by entering space and set style to the preset style "No Style, No Grid". In Table #3, set the font color to **red** in **Text30**'s column.

- Now reposition the transparent tables (that is, Table #2 and #3) to the same position as the first one (that is, Table #1).
- Next split the view and show the Task Sheet view on the bottom pane; or open new windows (use <Shift+F11>) and arrange them. Then add new tasks, change **Number1**'s values in order to see how the indicator symbols change.
- Copy the custom text fields and the custom graphical report to the Global template and close the project plan without saving. The graphical report is now ready to use.



SAMPLE

This is the **Graphical Indicators** dialog box for the task custom number field which displays indicators in the task table:



SAMPLE

How Does MS Project Calculate Progress on Summary Tasks ?

Introduction

In this article, we will discuss how MS Project calculates and displays the progress on the summary tasks by using the subtask information. We will create some formulas to help us with interpreting how MS Project calculates summary tasks' progress data in the background (we will work with the auto-scheduled tasks and summary tasks, unless otherwise stated).

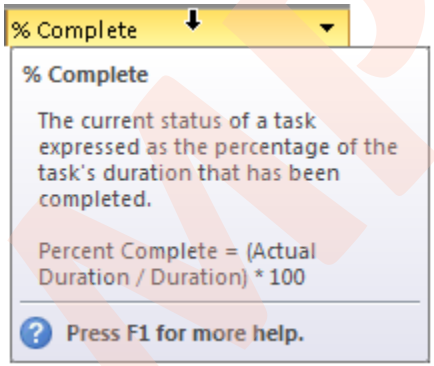
In the articles <Complete vs CompleteThrough> and <Why Not Use the Stop Field To Draw the Task Progress Bars ?>, we have discussed how progress on the tasks are visually represented in MS Project. By default, MS Project does not display a progress bar on a summary task; so the first question would probably be "how can we add a progress bar to a summary task to represent the progress (or the duration-based percentage of completion) graphically on the Gantt graph ?". Let us find out together.

SAMPLE

Using the CompleteThrough Date to Display Progress on a Summary Task

Let us first discuss how MS Project calculates the **% Complete** value for a summary task. MS Project uses the following formula to calculate a task's percent complete value:

Percent Complete = (Actual Duration / Duration) * 100



As soon as we input some progress information for a task, MS Project calculates the percent complete value along with the task's other related project data. MS Project rounds the percent complete value entered (or internally calculated) off to the nearest whole number. Notice that our purpose here is to discuss how the percentage of completion is rolled up to the summary level otherwise MS Project allows us to directly enter progress information into the **% Complete** field (that is, the field entry type is "calculated or entered") for both tasks and summary tasks.

SAMPLE

The above formula does not apply to the summary tasks since a summary task's progress is the rolled up progress of its subtasks. The rollup method is based on calculating a duration weighted average of the percent complete values of all the subtasks by dividing <the sum of the product of the multiplication duration * percent complete > to <the sum of the duration values> for all the subtasks. Consider the example below, where actual duration values are entered for Task 1, Task 2, Task 3, Task 4 and Task 5.

Task Name	Duration	Actual Duration	% Complete
- Summary A	**5 days**	**2.88 days**	**58%**
Task 1	3 days	1.5 days	50%
Task 2	2 days	1 day	50%
- Summary B	**5 days**	**3.5 days**	**70%**
Task 3	3 days	2 days	67%
Task 4	2 days	1.5 days	75%
Task 5	3 days	1.5 days	50%

MS Project calculates and displays subtasks' percent complete values and then rolls up the progress information to the summary level. The summary level percent complete value can be calculated manually, as follows (in this calculation, let us use **Actual Duration** values entered, instead of the multiplication **Duration * % Complete** since the percent complete values that we see in the picture above are the values rounded off to whole numbers):

SAMPLE

Summary A's Percent Complete = (1.5 + 1.5 + 2 + 1 + 1.5) / (3 + 2 + 3 + 2 + 3) = 7.5 / 13 = 58% then

Summary A's Actual Duration = (7.5/13) * 5 = 2.88 days

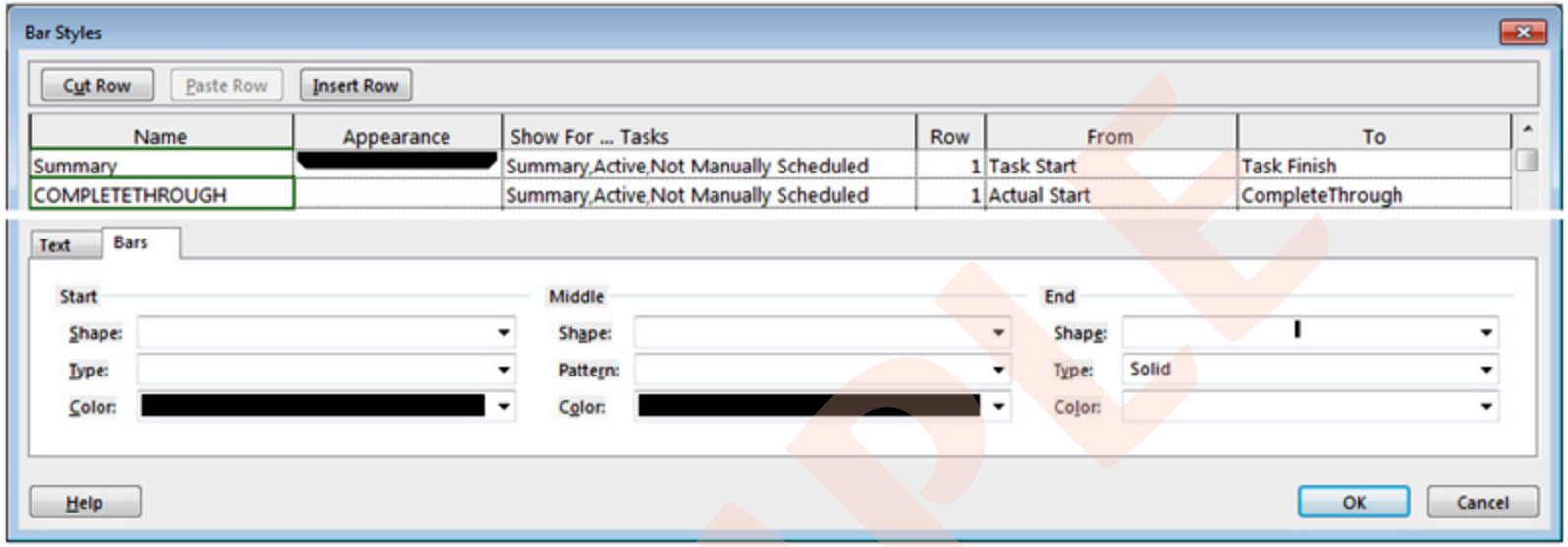
Summary B's Percent Complete = (2 + 1.5) / (3 + 2) = 3.5 / 5 = 70% then

Summary B's Actual Duration = (3.5/5) * 5 = 3.5 days

Note -- Search the product support website for the knowledgebase article "How Percent Complete Is Calculated for Nested Summary Tasks" --

In order to show a summary task progress bar on the Gantt Chart view, we can simply insert a summary progress bar definition line right below the line where we see "Summary" in the Name column, in the **Bar Styles** dialog box, as shown below:

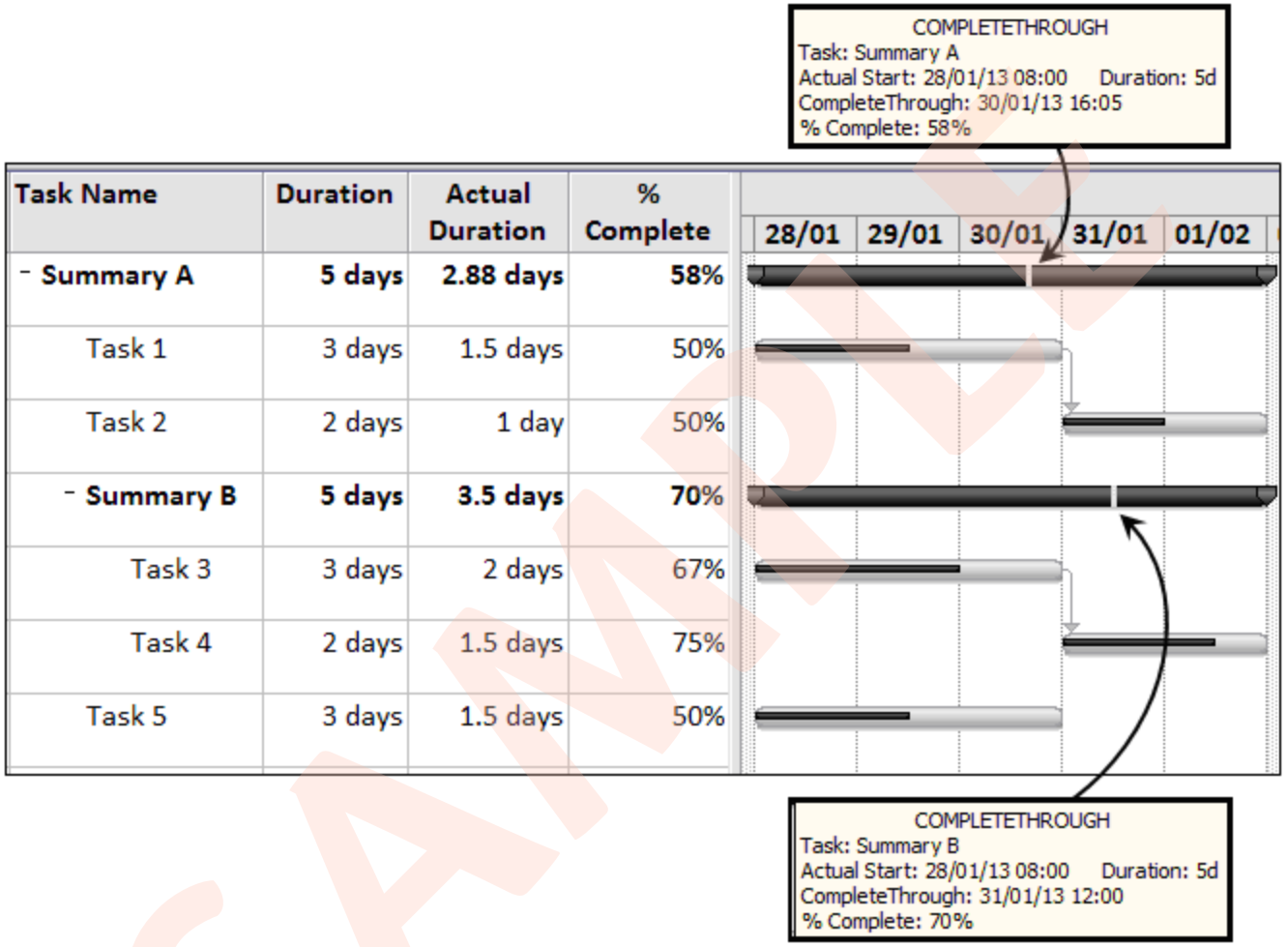
SAMPLE



In order to improve visibility, the image setting for COMPLETETHROUGH in **Bars** tab of the **Bar Styles** dialog box is changed to a white colored vertical bar as the end shape marking the **CompleteThrough** date on the summary task, instead of a horizontal progress bar. Also using a capitalized name such as COMPLETETHROUGH might help us to recognize custom entries in the dialog box while editing later on.

At this point, we are now able to show summary task progress which is based on the percent complete value graphically on the Gantt Chart. We can now see the **CompleteThrough** date in the ScreenTip® box displayed by holding the mouse pointer over a summary task progress bar drawn from the **Actual Start** date to the **CompleteThrough** date on the Gantt Chart, as shown below (the actual start date is January 28th, 2013 08:00 for both Summary A and Summary B in this example) (see the next page):

SAMPLE



(*Note that the schedule has been updated out of sequence for demonstration purposes*)

SAMPLE

We can create formulas to manually calculate the progress data:

- For example, the task custom date field formula **iif([Summary], ProjDateAdd([Actual Start], [Actual Duration], [Project Calendar]), "NA")** returns the dates January 30th, 2013 16:05 and January 31st, 2013 12:00 for Summary A and Summary B, respectively. As it seen in the picture above, these dates and times are the same as the **CompleteThrough** dates and times shown in the ScreenTip®s. While entering the formula, it is required to select the "Use formula" option in the "Calculation for task and group summary rows" section of the **Custom Fields** dialog box in order to see the results at the summary rows. Note that the formula sets the task custom date field to NA for the subtasks since we focus on the summary tasks in this example. The field reference **[Project Calendar]** is redundant in the formula since the default is the project calendar.

- We can calculate the summary percent complete value by using a task custom text field formula, as follows:

  Duration1 field's formula: **[Duration]** Rollup option: Sum
  Duration2 field's formula: **[Actual Duration]** Rollup option: Sum
  Text1 field's formula: **iif([Summary], Format([Duration2]/[Duration1],"0%"),"")** Rollup option: Use formula

  Now the **Text1** field will show the summary percent complete values, 58% and 70%, for Summary A and Summary B, respectively, at the summary rows.

It is important to note that MS Project would warn us about the incomplete zero-duration milestones by creating a one-percent difference in the value of the **%Complete** field when all the subtasks have been completed. We need to complete all the sub-milestones (e.g. enter 100%) to see 100% in the **%Complete** field at the corresponding summary level. The above manual calculations or the formulas do not reflect this behavior.

SAMPLE

Let us now find out how it works with inserted projects. As it is seen in the example below, the **% Complete** field's value at Summary A level is calculated differently when Summary B is inserted as a subproject below it:

- Manually calculating **% Complete** for Summary A, with another summary task Summary B, below it (see the picture below, on the left):

  Summary A's Percent Complete = (0.5 + 1.5 + 0 + 0 + 3) / (2 + 3 + 2 + 2 + 4) = 5 / 13 = 38%

  Summary A's Actual Duration = (5/13) * 5 = 1.92 days

- Manually calculating **% Complete** for Summary A with inserted subproject Summary B below it (see the picture below, on the right):

  Summary A's Percent Complete = (1.14 + 0 + 3) / (4 + 2 + 4) = 4.14 / 10 = 41%

  As it is seen from the results, the inserted project Summary B is treated like a subtask, and the duration data of its subtasks T3, T4 and T5 are not included in calculations.

  Summary A's Actual Duration = (4.14/10) * 5 = 2.07 days

SAMPLE

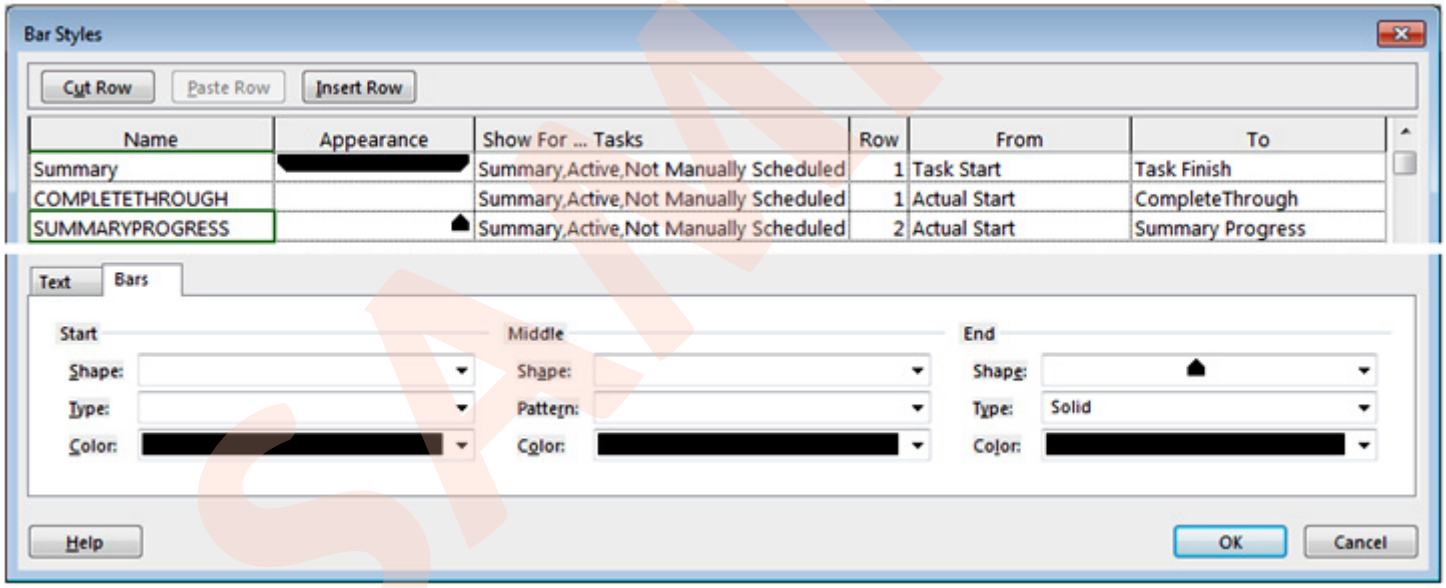
Name	% Complete	Duration	Actual Duration
Summary A	**38%**	**5 days**	**1.92 days**
T1	75%	4 days	3 days
T2	0%	2 days	0 days
Summary B	**29%**	**4 days**	**1.14 days**
T3	0%	2 days	0 days
T4	50%	3 days	1.5 days
T5	25%	2 days	0.5 days

Name	% Complete	Duration	Actual Duration
Summary A	**41%**	**5 days**	**2.07 days**
T1	75%	4 days	3 days
T2	0%	2 days	0 days
Summary B	**29%**	**4 days**	**1.14 days**
T3	0%	2 days	0 days
T4	50%	3 days	1.5 days
T5	25%	2 days	0.5 days

SAMPLE

Using the Summary Progress Date to Display the Progress on a Summary Task

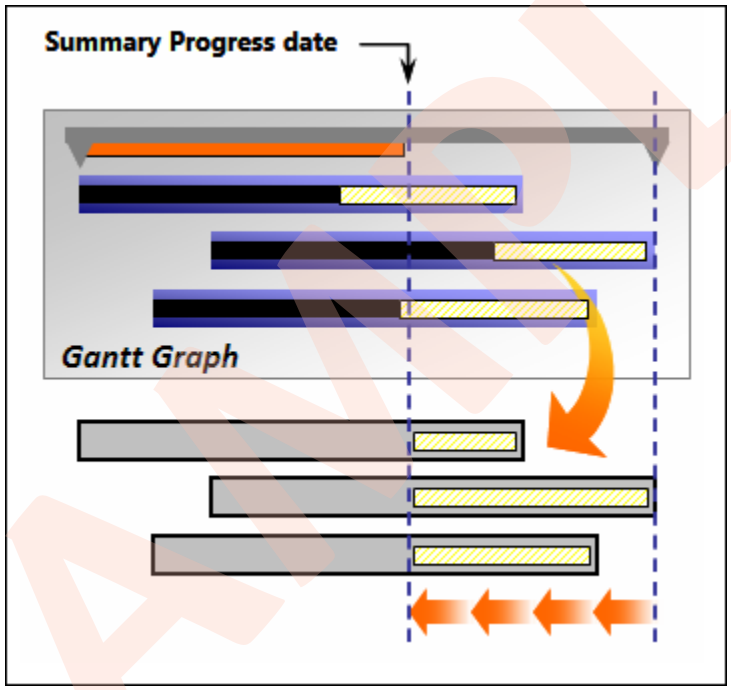
We will now discuss another rollup method in MS Project; it is based on calculating the summary task progress by taking into account where its subtasks have been scheduled. In order to show the summary task progress based on this method, the **Summary Progress** field which is available only in the **Bar Styles** dialog box will be used to draw the summary task progress bar on the Gantt Chart view, instead of the **CompleteThrough** field. In our example, we will keep both progress bars on the Gantt Chart view for comparison purposes by arranging the bar definitions as follows (see the entry SUMMARYPROGRESS using the 2nd rows on the Gantt Chart):



SAMPLE

Note -- Search the field reference section on the product help documentation for the description of the date type task fields, **Summary Progress** and **Complete Through** --

As mentioned above, the **Summary Progress** field considers when the subtasks are scheduled. So how can we simulate this rollup method on a schedule ? The figure below presents a graphical simulation of calculating a summary task's progress based on subtasks' schedule:



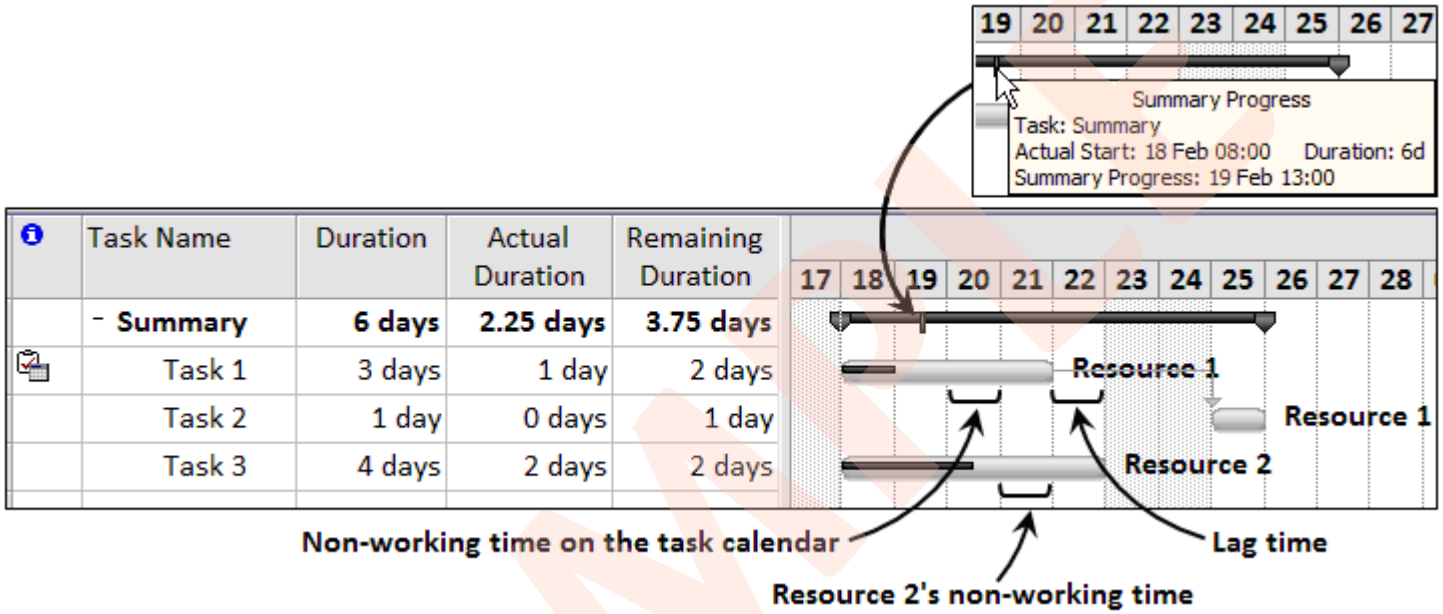
SAMPLE

It works as follows:

- We start at the end of the latest subtask (that is, the **Finish** date and time of the summary task) and count all the subtask minutes scheduled in every minute of the timescale while scanning the schedule from right to left (i.e. backwards), minute by minute, until we get to the point on the timescale where the total counted minutes are equal to the total remaining duration of all the subtasks; this point on the timescale is the date where the remaining portion of the progress for all the subtasks begins and hence it is the **Summary Progress** date where the remaining portion of the summary task progress begins.
- In this manual distribution process, we skip the periods on the timescale where there are no task duration minutes. And also note that we even count the task duration minutes for which there is some progress (actual duration minutes) entered in the active schedule at the present.

SAMPLE

We can further test the method discussed above on a simple schedule by doing a manual calculation; that is, by quickly distributing the remaining duration, day by day as follows:



Total Subtask Remaining Duration : **2 + 1 + 2 = 5d**
Distributing the total subtask remaining duration manually from right to left, as follows:

25th	Task 2: **-1d**
24th, 23rd	Skip non-working time on the project calendar
22nd	Skip the lag time on Task 1, Task 3: **-1d**
21st	Task 1: **-1d**, Skip Resource 2's non-working time on Task 3
20th	Skip non-working time on Task 1's task calendar, Task 3: **-1d**

As shown above, we have distributed **4 days (1d + 1d + 1d + 1d)** so far. Both Task 1 and Task 3 have working times on the 19th, so we need to divide the remaining 1d by 2 in order to get an even distribution as follows:

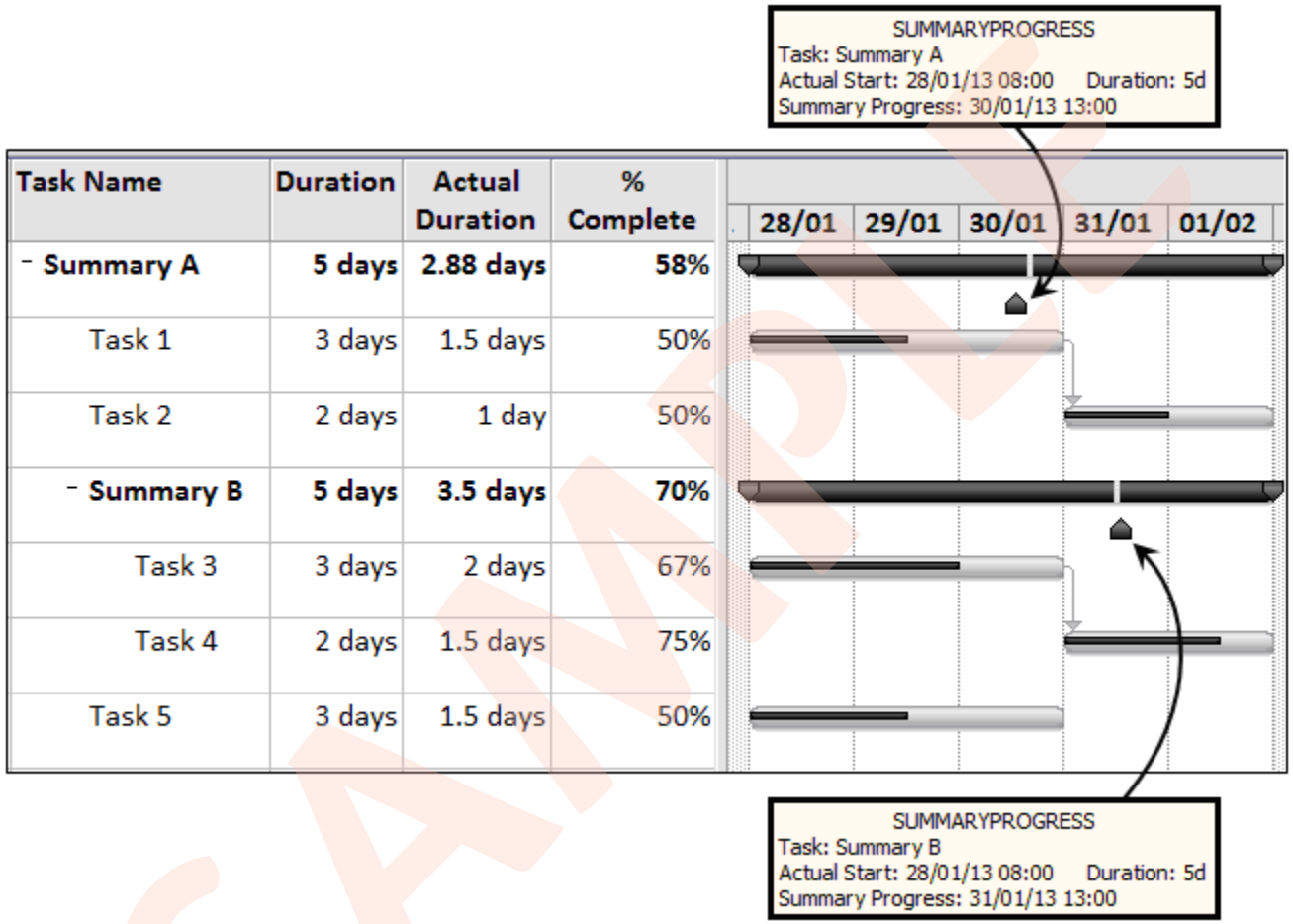
19th : Task 1: **-0.5d**, Task 3: **-0.5d**

The date arithmetic **19th Feb 17:00 – 0.5d** yields the date and time **19th Feb 13:00**; so this is the beginning date and time of the remaining duration at the summary level, that is, the **Summary Progress** date.

It is obvious that the manual calculation would be very difficult on a large schedule. On the other hand, we do not need to worry about the details on how it is being calculated or what algorithm MS Project uses since all the calculations are automatically performed in the background and the resulting value is stored in the **Summary Progress** field. MS Project displays the **Summary Progress** date only in the ScreenTip® of the summary progress bar as this is a date and time information which is not available to access anywhere in a project plan. It is calculated internally by MS Project and its value is made available in the **Bar Styles** dialog box for us to use in order to show the summary progress bars on the graph.

SAMPLE

Let us continue with the previous example:



SAMPLE

Note the summary progress bar's end markers (⏶) used instead of a solid progress bar on the figure. MS Project calculates the summary progress for Summary B, based on the schedule of the subtasks, Task 3 and Task 4. On the other hand, for Summary A, it is calculated based on the schedules of all the subtasks from Task 1 to Task 5. The ScreenTip® boxes displayed over the summary progress bar's end markers (⏶) do not contain the percent complete values but the summary progress dates as shown below:

Summary A's **Summary Progress** date and time: January 30th, 2013 13:00
Summary B's **Summary Progress** date and time: January 31st, 2013 13:00

We can manually calculate the percent complete value corresponding to the **Summary Progress** date as follows:

- Enter the **Summary Progress** dates and times displayed in ScreenTip® boxes into the task custom date field **Date1** by typing in.
- Enter the following formula to the **Text5** field: **iif([Summary], Format(ProjDateDiff([Actual Start], [Date1], [Project Calendar]) / [Duration],"0%"),"")**
- Select the "Use formula" option in the "Calculation for task and group summary rows" section of the **Custom Fields** dialog box while defining **Text5**.

The field reference **[Project Calendar]** is redundant (optional) in the formula since it is the default. And the results will be :

	Text5 field:	**Date1** field:
Summary A	50%	30/01/2013 13:00
Summary B	70%	31/01/2013 13:00

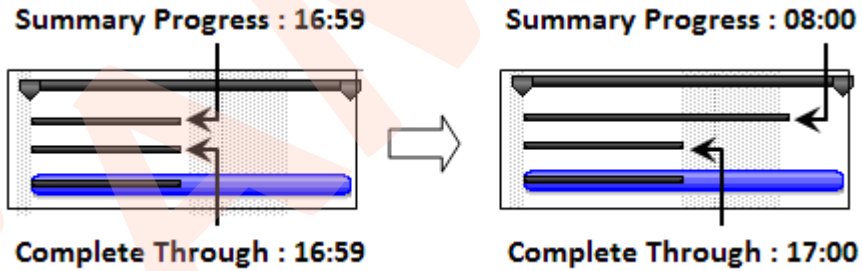
SAMPLE

Let us compare the results from both methods:

	CompleteThrough (%Complete)	Summary Progress
Summary A	30/01/2013 16:05 (58%)	30/01/2013 13:00 (50%)
Summary B	31/01/2013 12:00 (70%)	31/01/2013 13:00 (70%)

As we would expect, the **SummaryProgress** dates are different than the **CompleteThrough** dates.

The **Summary Progress** field gives us the date for the beginning of the remaining portion of the progress. For example, any progress date ending at the time 17:00 of Friday will be calculated as the time 08:00 of the next Monday morning in the **Summary Progress** field, in a schedule using the Standard base calendar as the project calendar. On the other hand, the **CompleteThrough** date is the end date of the actual portion of the rolled up progress of all subtasks, which is calculated based on the method explained earlier in the previous articles. See the single task schedule with no resources below, where we have first entered a stop date at 16:59 and then a stop date at 17:00 in order to see the difference on how MS Project calculates the **CompleteThrough** and **Summary Progress** dates for a summary task:



SAMPLE

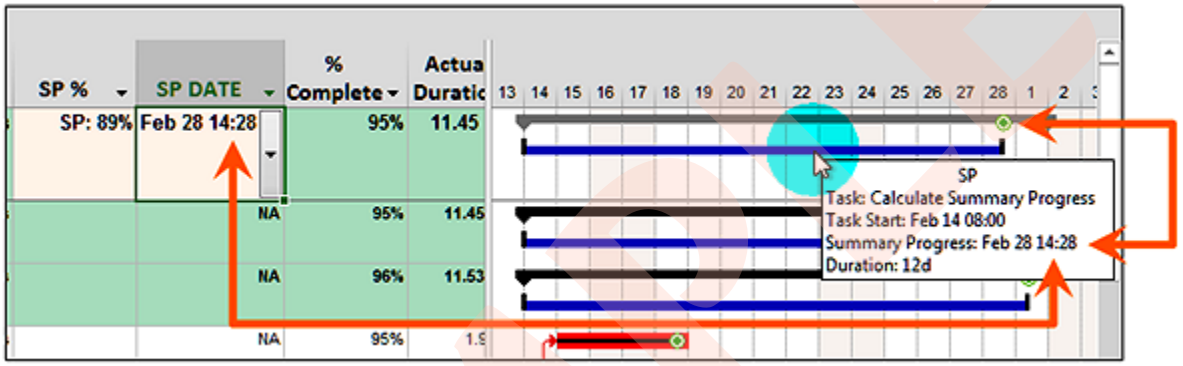
Unchecking the box "Round bars to whole days" in **Layout** dialog box would make the difference more apparent. Note that the **Stop** date of a summary task is the earliest one among the stop dates of all subtasks; and the **Resume** date of a summary task is the earliest one among the resume dates of all subtasks.

In MS Project, as explained above, we can display a summary progress bar using the **Summary Progress** date in the Gantt Chart view and then hover the mouse pointer over the bar to display a ScreenTip® containing its date value, but we cannot directly access the **Summary Progress** field's value or the percent complete value to which it corresponds. MS Project also uses the **Summary Progress** date to place progress point markers on the summary tasks while drawing the progress lines.

> **Important Note** -- The field description for the task field **Summary Progress** does not provide details on how MS Project calculates the **Summary Progress** field's value. So beware of the fact that the calculation method discussed above is based on -- the results of experimenting with MS Project --, but not on the documented product information. Therefore, the content may be updated based on the latest technical information released on the product. The examples here have been tested with the fully updated latest desktop version of the product at the time of publishing this article. The results may be different on the previous versions or the future versions of the product --

SAMPLE

Note -- As a further attempt to test the method described above, - even if it does not serve a useful purpose - a VBA macro may be developed to distribute the subtasks' total remaining duration backwards in a small schedule. The picture below shows the result of such a test --



SAMPLE

About the Author

Visit the author's website at www.ismetkocaman.com

Notice of Rights

All rights reserved. No part of this eBook may be reproduced, stored in a retrieval system or transmitted in any form or by any means, without the prior written permission of the author.

Notice of Liability

Every effort has been made to ensure the accuracy of the information herein. However, the information contained in this eBook is provided without warranty, either expressed or implied. The author will not be held liable for any damages to be caused either directly or indirectly by the instructions contained in this eBook, or by the application software described herein. The author provides formula examples for demonstration only, without warranty either expressed or implied.

Trademark Notice

Microsoft is a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries. All other trademarks mentioned herein are the property of their respective owners. The author has no affiliation with Microsoft Corporation. Screen captures were reprinted with authorization from Microsoft Corporation. This document is not a product of Microsoft Corporation.

Copyright © Ismet Kocaman

eBook's website: www.ismetkocaman.com/Formulas/eBook6.html

SAMPLE