

# Microsoft Project Resource Leveling Cheat Sheet

## Scheduling:

- Timeline oriented
- Focus is to create shortest possible schedule duration
- Moves tasks in either direction
- Can split all remaining work from actual
- Sets remaining work resume date

## Leveling:

- Resource oriented
- Focus is the most effective resource utilization
- Only delays tasks from start or resume date
- Can split remaining work after resume date

## Leveling Mechanics (Leveling Options Window)

What constitutes an overallocation? When does leveling run? What date range is leveled?

		Advice
<b>Automatic or Manual</b>	Determines if leveling occurs with <u>EVERY</u> schedule change (Automatic) or only when initiated by the project manager (Manual).	Manual
<b>Look for overallocations on a [timeframe] basis</b>	Tells Project what constitutes an overallocation. For example, is it exceeding 8 hours in a day (Day by Day), 40 hours in a week (Week by Week) or something similar?	Week by Week
<b>Clear leveling values before leveling</b>	Resets fields calculated by leveling to their pre-leveling values.	Checked
<b>Leveling range for [project name]</b>	Defines the date range boundaries for leveling. Options are Entire Project or a defined date range. Date ranges are more useful when debugging a schedule.	Entire Project

## Resolution Options (Leveling Options Window)

Controls how Microsoft Project resolves an overallocation (the techniques it can use)

		Advice
<b>Level Only within available slack</b>	Levels resources as much as possible without changing the <i>project</i> end date.	Not Checked
<b>Leveling can adjust individual assignments on tasks</b>	Allows Project to adjust individual assignment <i>start dates</i> within the same task Task Override Field: <b>Level Assignments</b> .	Not Checked
<b>Leveling can create splits in remaining work</b>	Allows Project to create one or more splits in remaining work <u>AFTER</u> the task Resume date. This may allow the task to complete sooner by scheduling pieces of the remaining work into allocation gaps that exist around other tasks. When this occurs, Project is creating a custom work load pattern and the Work Contour field will be set to Contoured. See also Resolution options below: <b>Split in progress tasks</b> . Task Override Field: <b>Leveling Can Split</b> .	Checked
<b>Level resource with proposed booking type</b>	Includes or excludes proposed resources in the leveling process.	Checked
<b>Level manually scheduled tasks</b>	Allows Project to move one <u>manual</u> task when it conflicts with another <u>manual</u> task.	Checked

## Resolution Options (Project Backstage)

Scheduling options impacting leveling

		Advice
<b>Tasks will always honor their constraint dates</b>	Scheduling option (File   Options   Schedule) that causes leveling to honor constraint dates, even if the result is an overallocation.	Not Checked
<b>Split in progress tasks</b>	Scheduling option (File   Options   Schedule) that allows Scheduling to split a task and schedule all remaining work honoring the defined work breakdown structure (WBS). See also Resolution options above: <b>Leveling can create splits in remaining work</b> .	Checked

## Predefined Leveling Hierarchy & Definitions (Remaining Work)

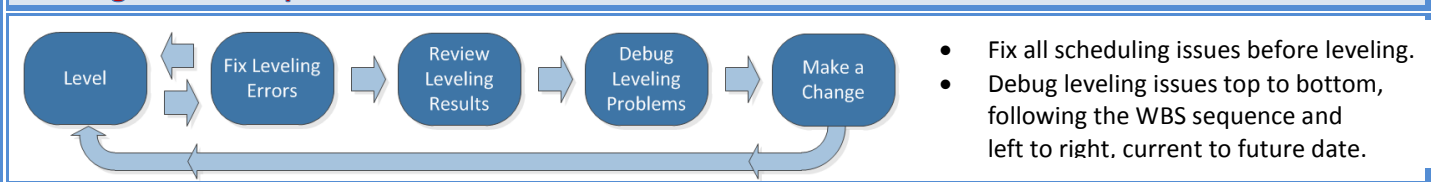
The tie-breaking hierarchy Project will follow to resolve overallocations

1) <b>Priority 1000</b>	Task Priority = 1000 is a special task designation. It means the task cannot be moved or delayed by the leveling process.																		
2) <b>Manual Tasks</b>	Tasks with Task Mode = Manual are generally left as defined by the PM, but can be moved. See leveling option: <b>Level manually scheduled tasks.</b>																		
3) <b>Started Tasks</b>	Project assumes that remaining work on <u>any</u> Started Task is assumed to be more important because the task has been started and is therefore scheduled first. This includes tasks started out of sequence. See Resolution option: <b>Split in progress tasks.</b>																		
4-8) [ <b>Leveling Order</b> ] <b>Standard</b> <b>Priority,Standard</b> <b>ID Only</b>	<p>The value selected in this option determines additional tiebreakers used to resolve overallocations. The precedence list for each option is shown to the right.</p> <table border="1"> <thead> <tr> <th>Standard¶</th> <th>Priority,Standard¶</th> <th>ID-Only¶</th> </tr> </thead> <tbody> <tr> <td>a)→ Predecessors</td> <td>a)→ Task-Priority</td> <td>a)→ ID-#</td> </tr> <tr> <td>b)→ Slack</td> <td>b)→ Predecessors</td> <td></td> </tr> <tr> <td>c)→ Task-Dates</td> <td>c)→ Slack</td> <td></td> </tr> <tr> <td>d)→ Task-Constraints</td> <td>d)→ Task-Dates</td> <td></td> </tr> <tr> <td>e)→ Task-Priority</td> <td>e)→ Task-Constraints</td> <td></td> </tr> </tbody> </table>	Standard¶	Priority,Standard¶	ID-Only¶	a)→ Predecessors	a)→ Task-Priority	a)→ ID-#	b)→ Slack	b)→ Predecessors		c)→ Task-Dates	c)→ Slack		d)→ Task-Constraints	d)→ Task-Dates		e)→ Task-Priority	e)→ Task-Constraints	
Standard¶	Priority,Standard¶	ID-Only¶																	
a)→ Predecessors	a)→ Task-Priority	a)→ ID-#																	
b)→ Slack	b)→ Predecessors																		
c)→ Task-Dates	c)→ Slack																		
d)→ Task-Constraints	d)→ Task-Dates																		
e)→ Task-Priority	e)→ Task-Constraints																		
9) <b>Task Duration</b>	Project assumes that longer duration tasks are more critical and therefore leveled before shorter duration tasks.																		
10) <b>Task ID #</b>	The overallocation is resolved based on the Task ID #. Lower ID # tasks (top of the task list) are assumed to have higher priority and leveled before higher Task ID #s.																		




## Leveling Order Fields & Definitions (listed in "Standard" order)

<b>Predecessors</b>	The WBS sequence is used to determine overallocation resolution. Tasks earlier in the WBS sequence are leveled first.
<b>Slack</b>	Tasks with lower Slack values level before tasks with higher Slack values.
<b>Task Dates</b>	The task with the earlier Start date levels first.
<b>Task Constraints</b>	Constraints such as "Must start on" or "Start on or after" are honored, if possible. Note: Task Constraints <i>may be</i> ignored if honoring the constraint would create an allocation conflict with a higher precedence task. See Resolution option: <b>Tasks will always honor their constraint dates.</b>
<b>Task Priority</b>	Tasks with higher Priority value are leveled before tasks with lower Priority value. Priority=1000 is a special designation (see above). 1=low, 999=high
<b>Task ID #</b>	The overallocation is resolved based on the Task ID #. Lower ID # tasks (top of the task list) are assumed to have higher priority and leveled before higher Task ID #s.

## Leveling Process Steps



## Leveling Functions Resource Override Field: Can Level=No - Resource is ignored in leveling

 <p>Levels all resources all tasks.</p> <p>Level All</p>	 <p>Levels only the <i>selected</i> tasks.</p> <p>Level Selection</p>	 <p>Levels the selected resources across all their task assignments.</p> <p>Level Resource</p>
---	--	---