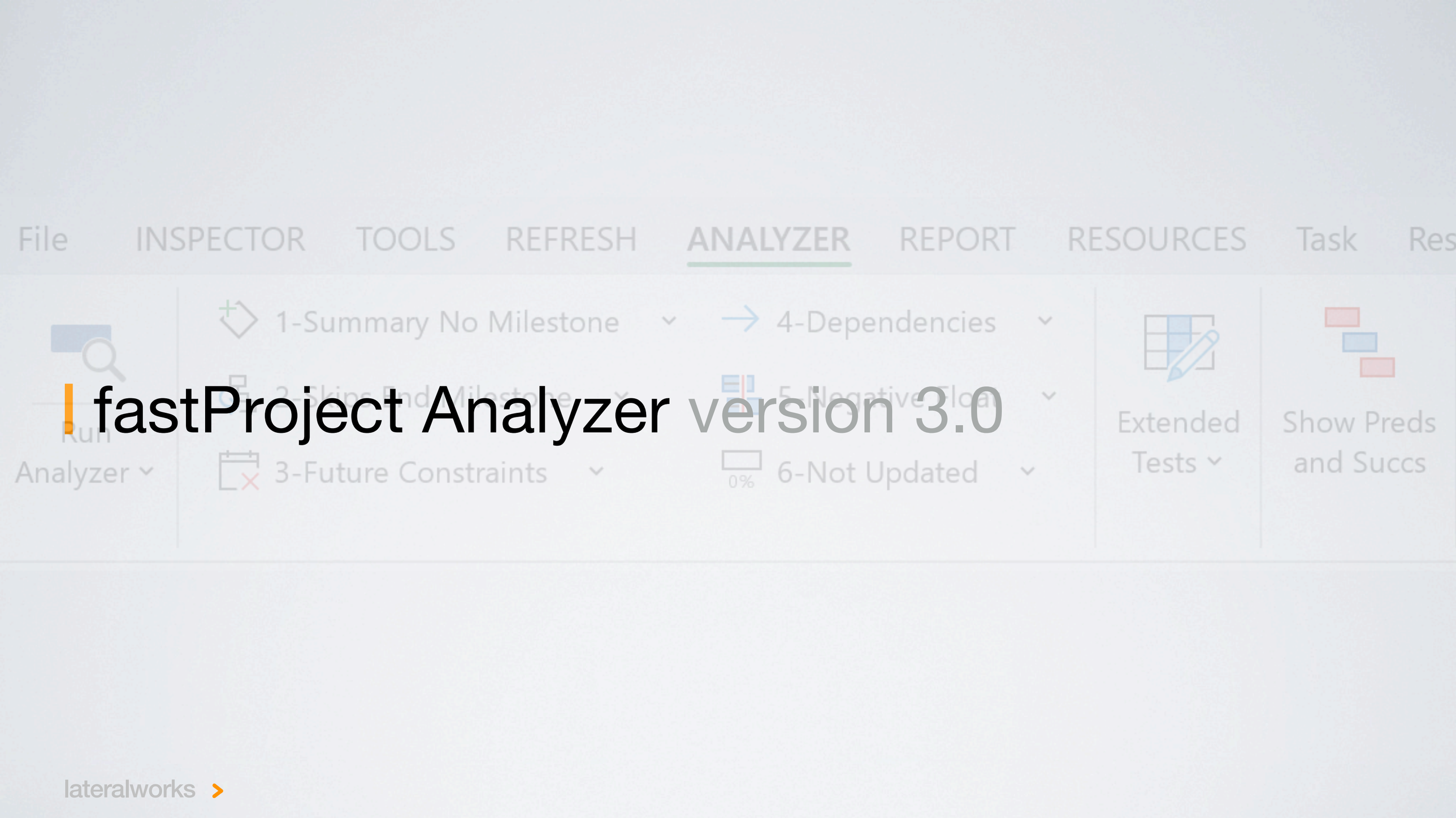


fastProject Analyzer version 3.0

Analyzer Released September 2022

- Core tests that are essential for proper critical path get a pass or fail rating. They must all pass to achieve an overall schedule Pass.
- Additional tests added (under Extended tests)
  - Time permitting, those tests identified as a problem should be fixed
- Optimized to run faster
- Increased the number of problems that can be automatically fixed via the dropdown to each test
- Some problems are now automatically fixed when the Analyzer is run:
  - Manual/Split tasks
  - Updated in the Future
  - Durations in anything but days gets converted to days
  - Duplicate dependencies deleted (with the option of turning this off through fastProject Options)

Why run the Analyzer?

- **To ensure a correct Critical Path (per CPM best known practice) in order to**
  - Predict a project's end-date based on team's current outlook
  - Understand the gap between the Target date and the Project Milestone date and what is driving that gap
    - Knowing this ahead of time gives you more time to close the gap
- **To preserve the integrity of the Critical Path**
  - Any "mistake" in the schedule can corrupt the "true" critical path
  - The wrong Critical Path means:
    - Wrong end dates are reported
    - The team is focused on the wrong thing
    - Inability to accelerate
    - The the integrity of the planning process is compromised

Why make a Critical Path schedule?

…to find the critical path driving the end date, why is this important?

…to know what to pull-in today in order to accelerate, why accelerate?

…because

- you have to accelerate to bank time
- you have to accelerate before you slip
- you have to accelerate just to be on time

But what if the critical path is wrong?

…you can’t bank time

…you can’t pull-in before you slip

…you can’t pull-in after you slip

…you don’t know where you are

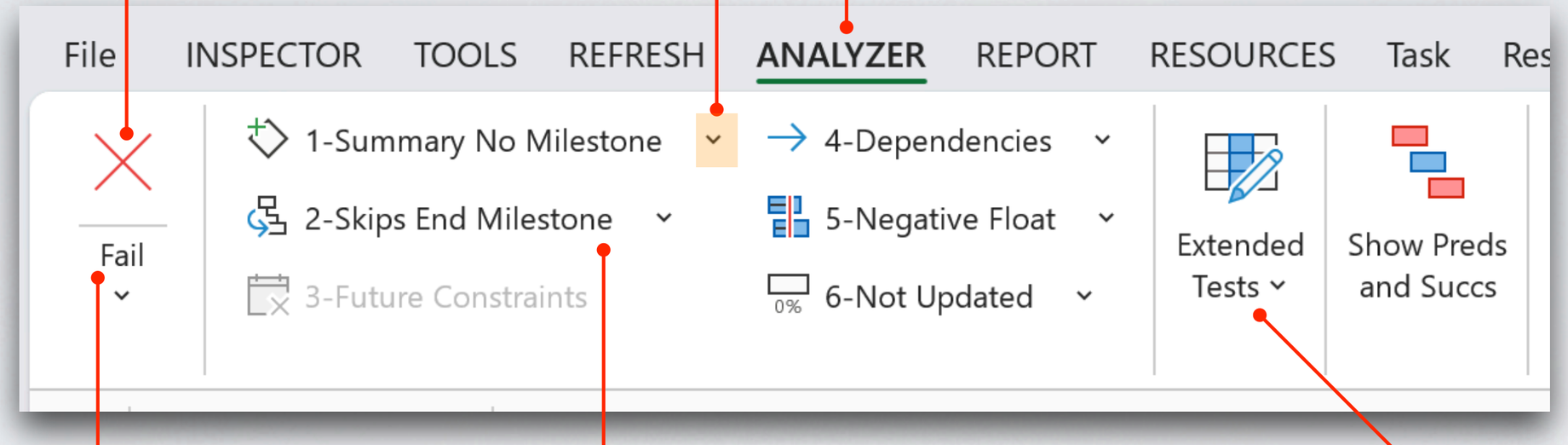
…you can’t use it to predict when you will finish

- The purpose of the Analyzer is to get an accurate schedule with a contiguous critical path from today to the end of the project
- The majority of the projects we see don’t have a good critical path and therefore cannot be used to accelerate a project or predict when it will be finished with any degree of accuracy

Run at least once a week and/or every time you make substantial changes to your schedule (i.e. prior to weekly Refresh Meeting)

Click dropdown for more options (i.e. auto fixes)

6 Core Tests enable accurate and technically correct schedules, based on Best Practices (these tests drive the Pass or Fail grade)

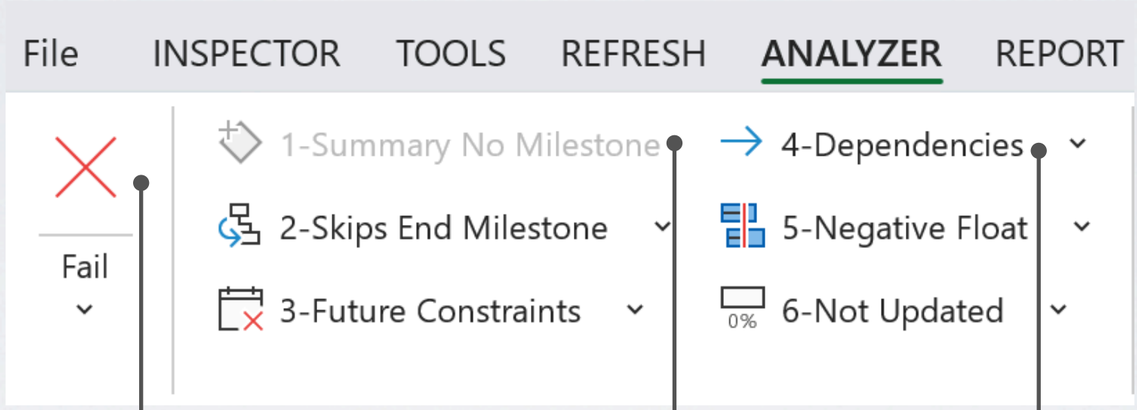


Your Score; **Pass or Fail** goal is to Pass every week

- Failed Test (a passed test is grayed out)
- Fix and then re-run Analyzer until all tests pass
- Fix in the order of the test, starting at #1 (some of the earlier tests will correct later ones)

Extended Tests should also be fixed, yet these do not contribute to the Pass/Fail score

Fail & Pass



Fix each test, then re-run Analyzer until you get a Pass

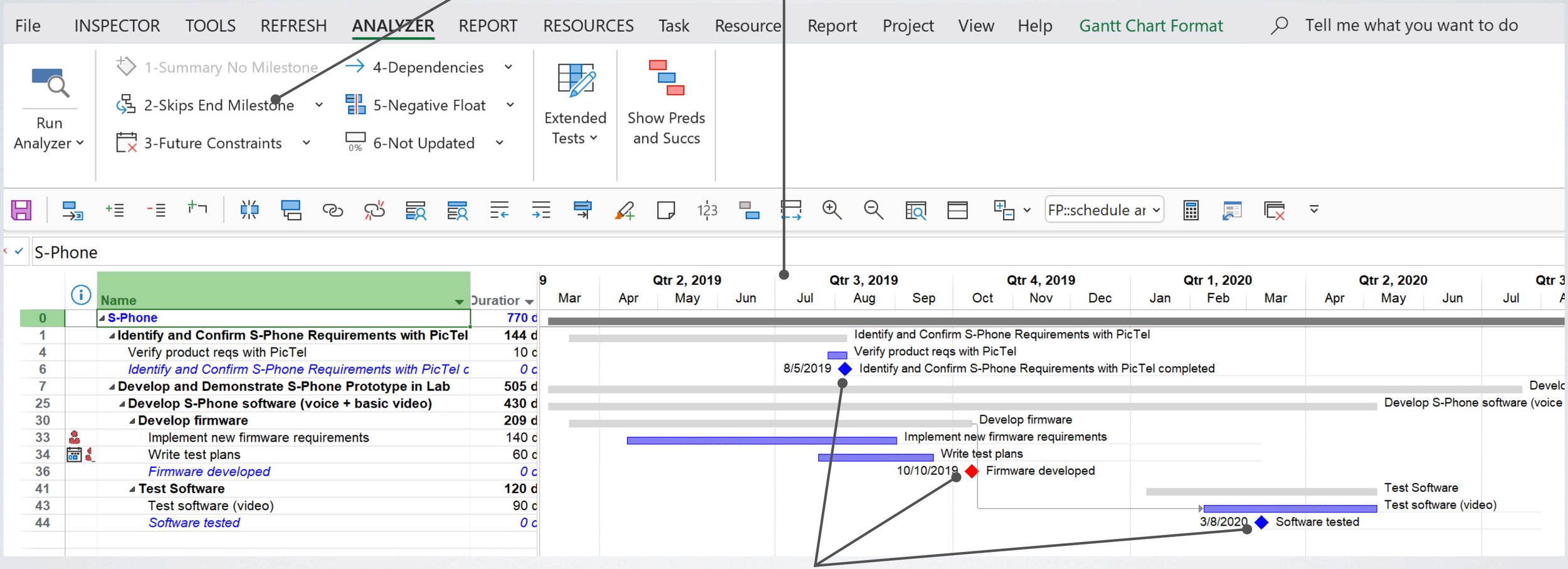
Re-run by clicking on the X

This test Passed

This test Failed

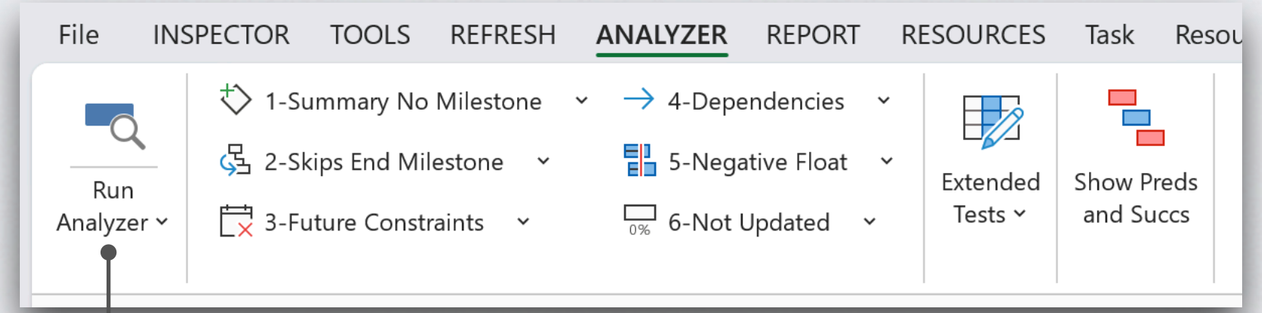
How to filter out the tasks involved in failed test

Clicking on failed test displays the offending tasks



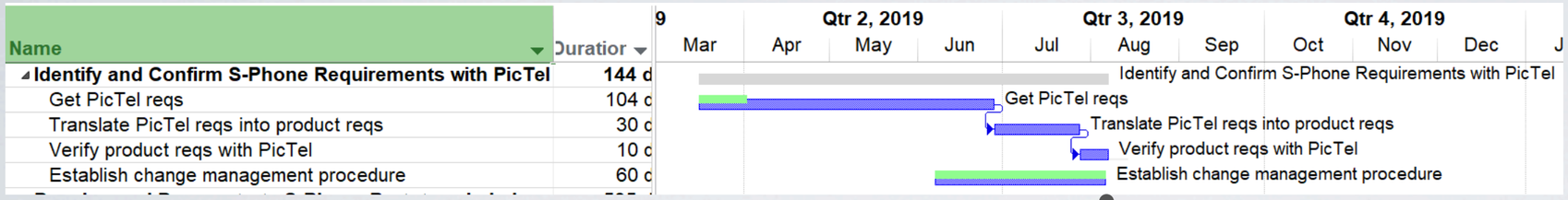
Each of these milestones skip the last milestone in their group hierarchy

Core Tests

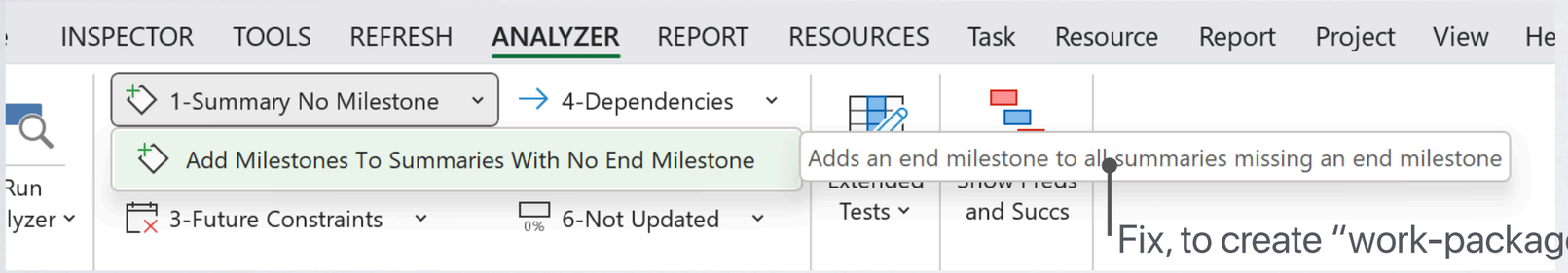


Must re-run the Analyzer, until all tests are grayed out. Non-gray tests indicate the test failed and need to be fixed.

1. Summary no milestone

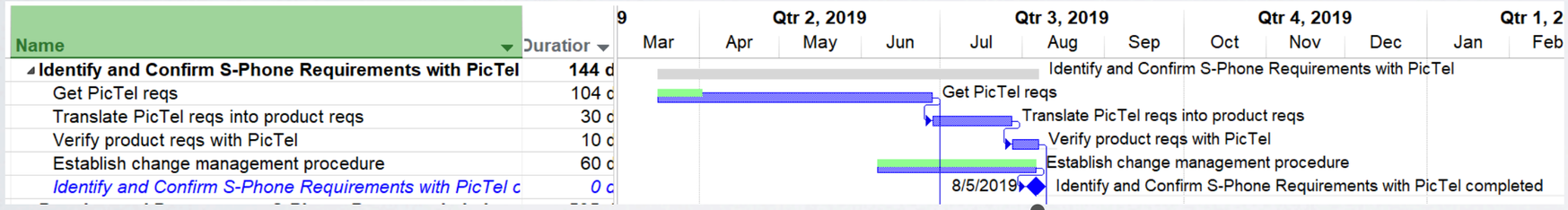


Missing milestone



Fix, to create “work-package”

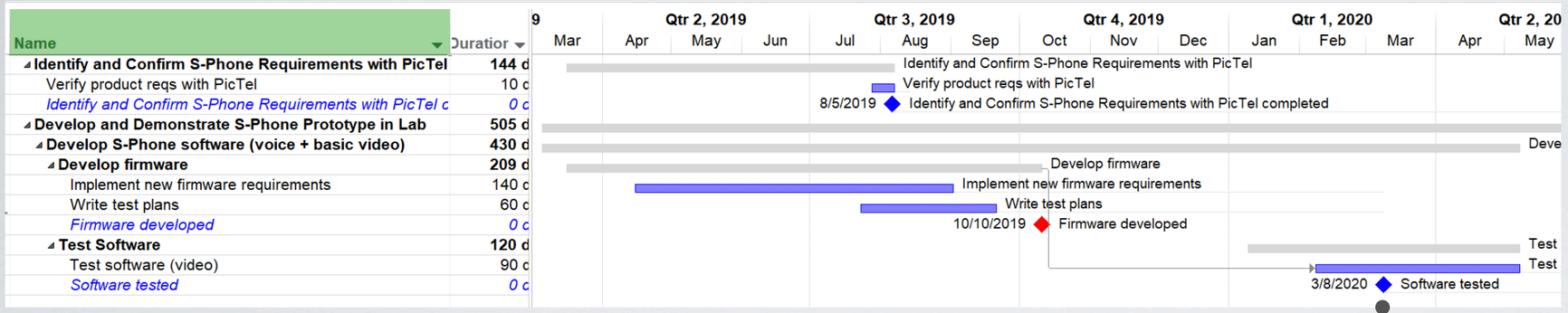
- This can reveal structural problems in the schedule whilst also indicating the subtasks under a summary were not determined "with the end in mind." By first defining the end state of a work package using doneness criteria, subtasks can then be better defined.
- Also used during Updating to determine if all the preceding tasks are complete and for linking between work packages.



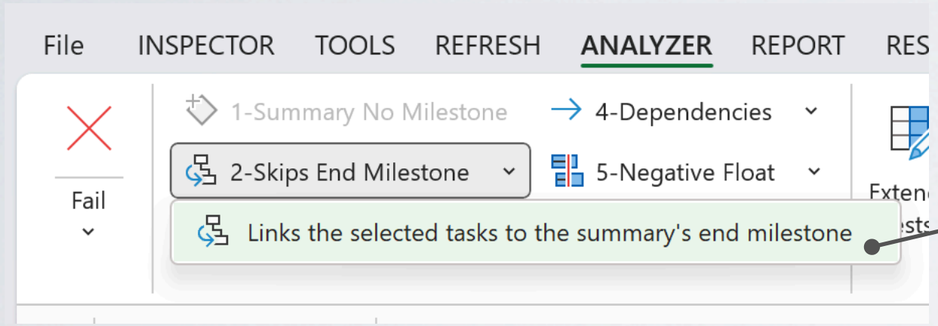
A **Work Package** is a summary, subtasks which conclude with a milestone

- Adds a new milestone at the correct indentation level and adds the correct predecessors and successors
- Will auto name the milestone (past tense) based on the name of the summary

2. Skips end milestone

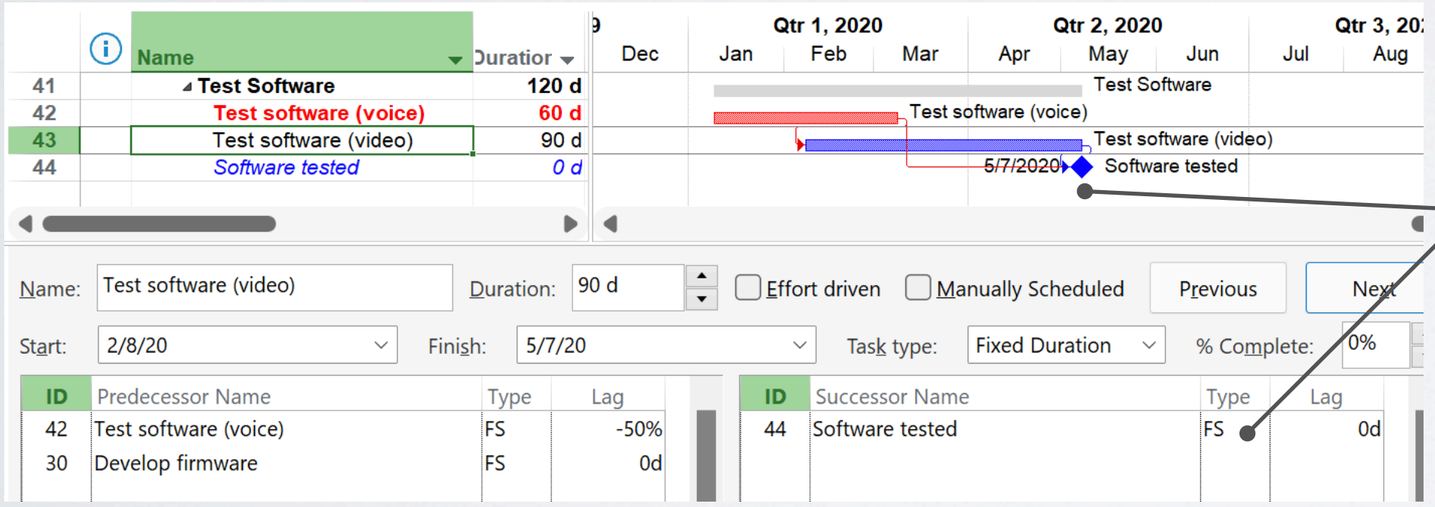


Visual clue is last milestone in work package should not finish earlier than its preceding task, also note long float paths of unconnected milestones



Auto fix:

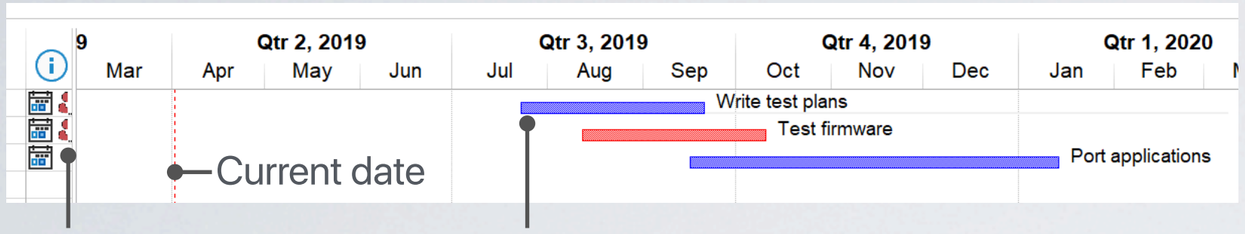
1. select the task (43)
2. select dropdown fix
3. automatically adds FS 43 to 44



Manual fix: add dependency - in this example, task 43 to 44 FS in order to make the work package complete

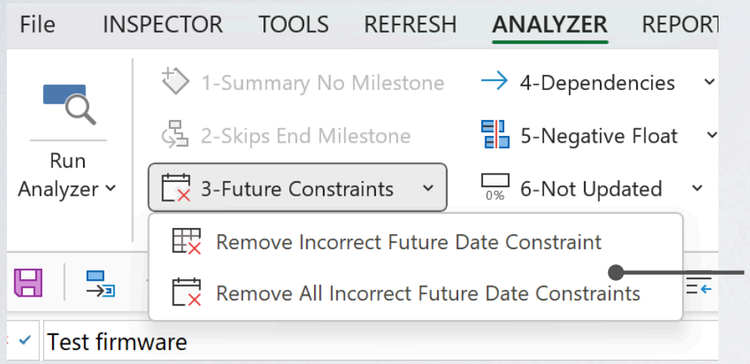
- Can indicate structural problems with the schedule and the need to relocate task(s) to another work package.
- Check the doneness criteria to determine if the task should be linked to the end milestone or whether it should be moved.
- This test can distort the critical path because the milestone at the end of the work package is not finishing when it should.

3. Future constraints

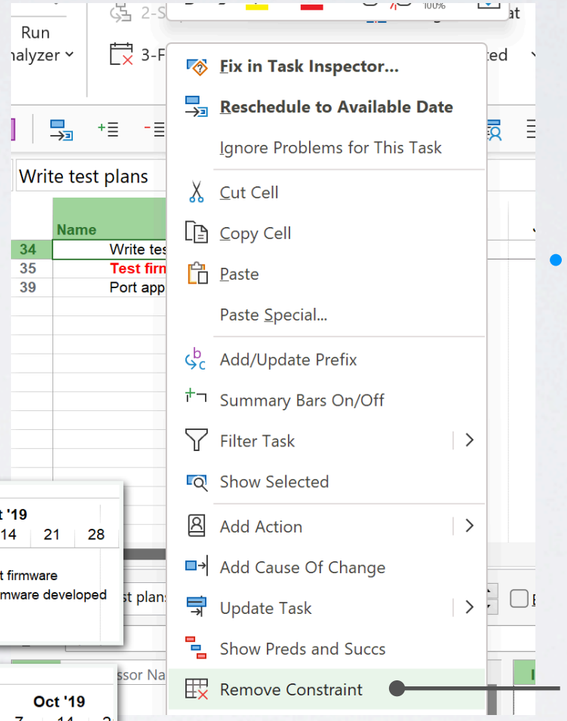


Indicates a date constraint

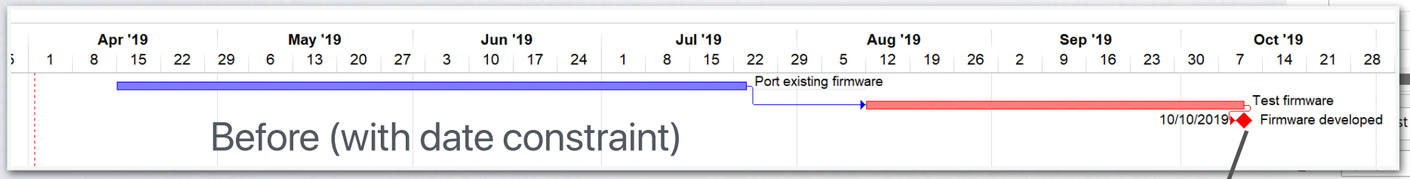
Visual clue, no predecessor, yet it starts in the future. Today's date is the red dotted line (i.e. the current date).

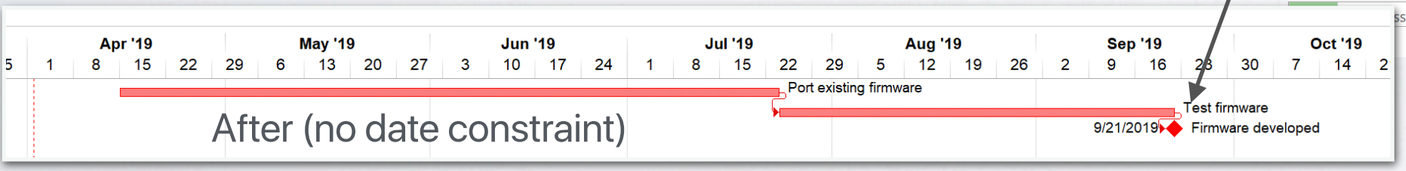


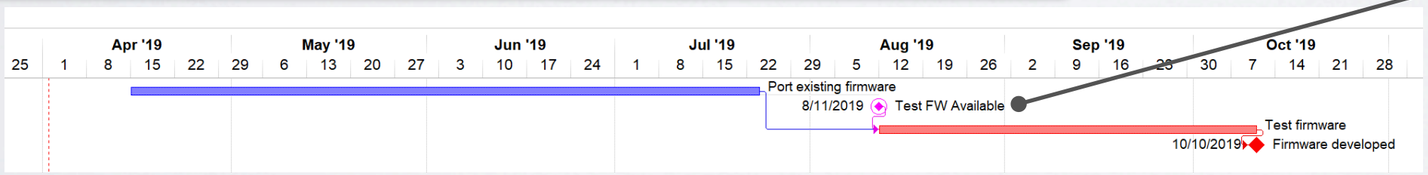
This function will fix the problem automatically for you, removes date constraints in future



Right-mouse click on the task



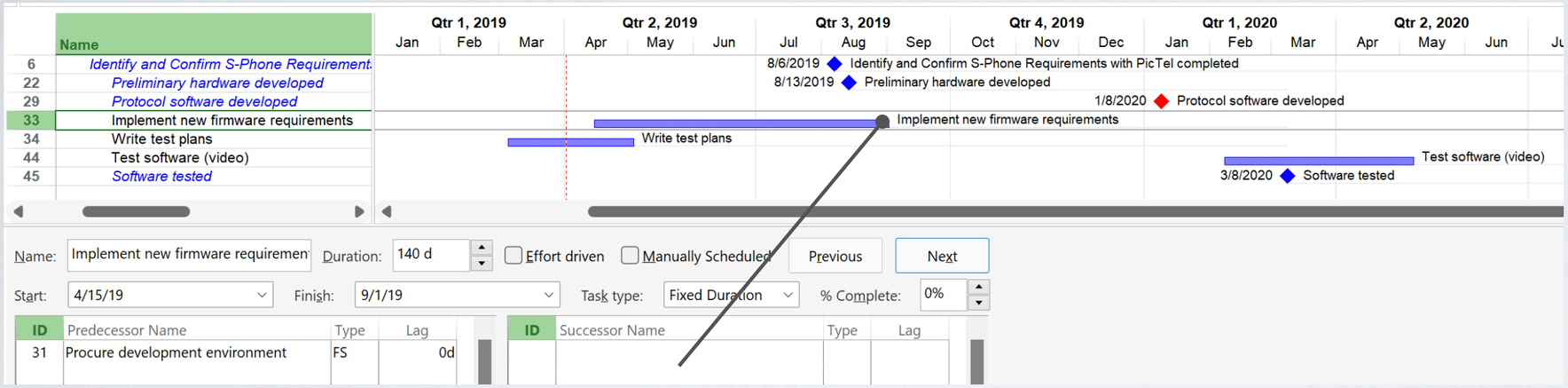




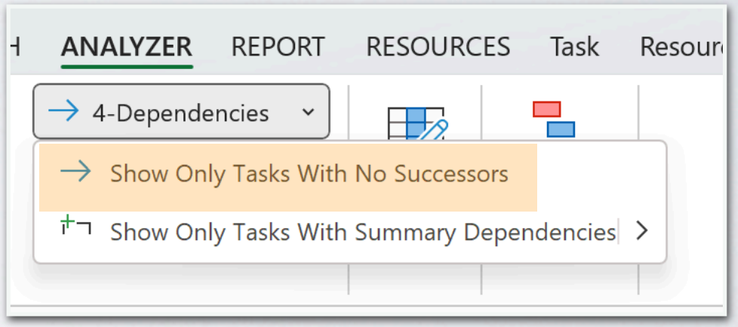
- Date constraints in the future prevent a schedule from moving back and forth. When this happens, they can't be used to assess the health of a project or provide an early warning.
- Date constraints can also distort the critical path.

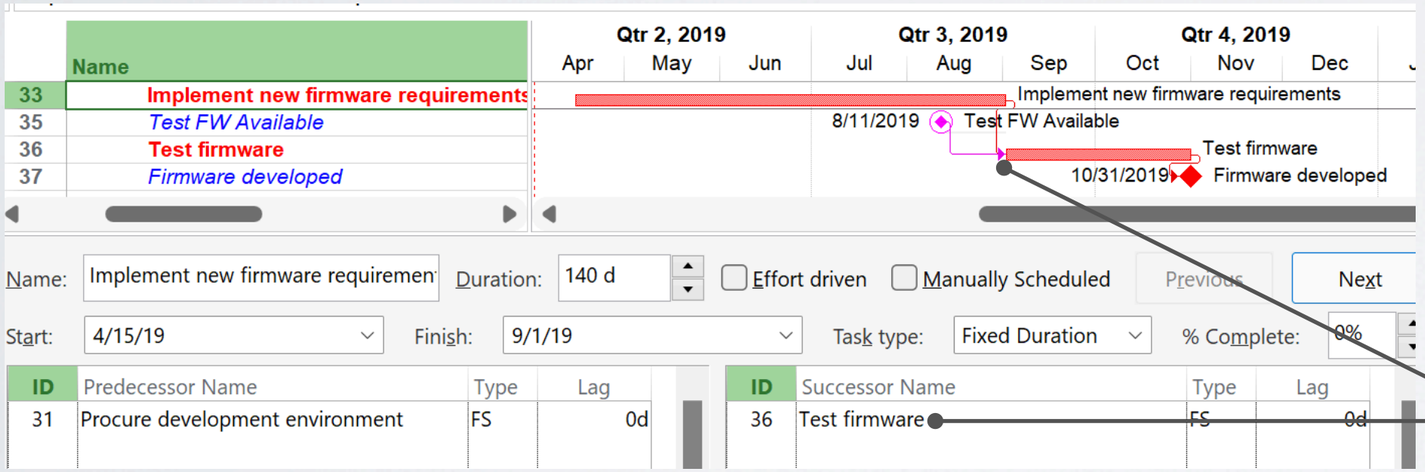
Note: If you have an external input to your project that you don't control (i.e. delivery of equipment from a supplier), use a "Touchpoint" milestone with a date constraint, then connect your tasks to this milestone. The Analyzer will ignore this future constraint if it is a Touchpoint or a Checkpoint.

5. Dependencies



- Missing dependencies prevents the critical path flowing through subtasks of the WBS hierarchy.
- This limits pull-in opportunities and can cause negative float in some conditions.

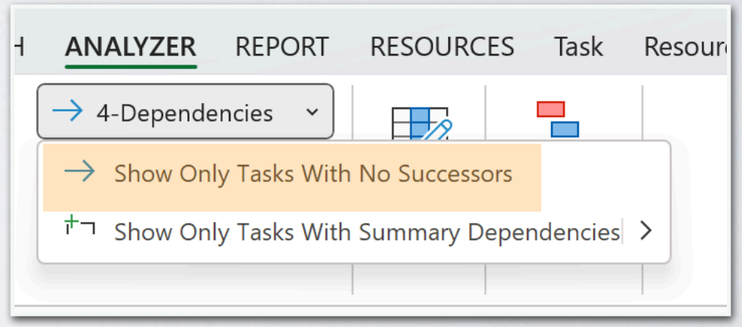


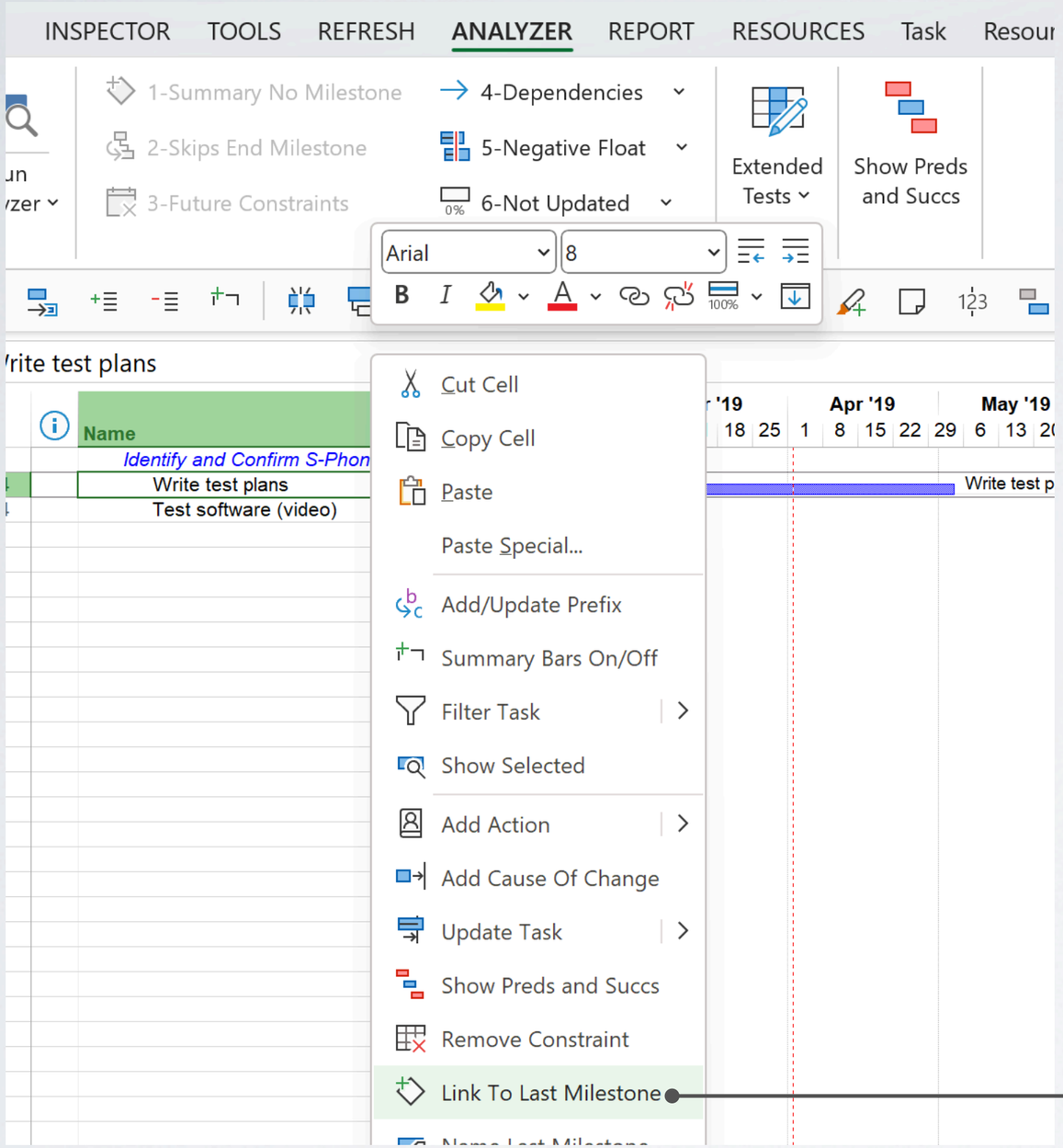


A successor was added, now on Critical Path

5. Dependencies

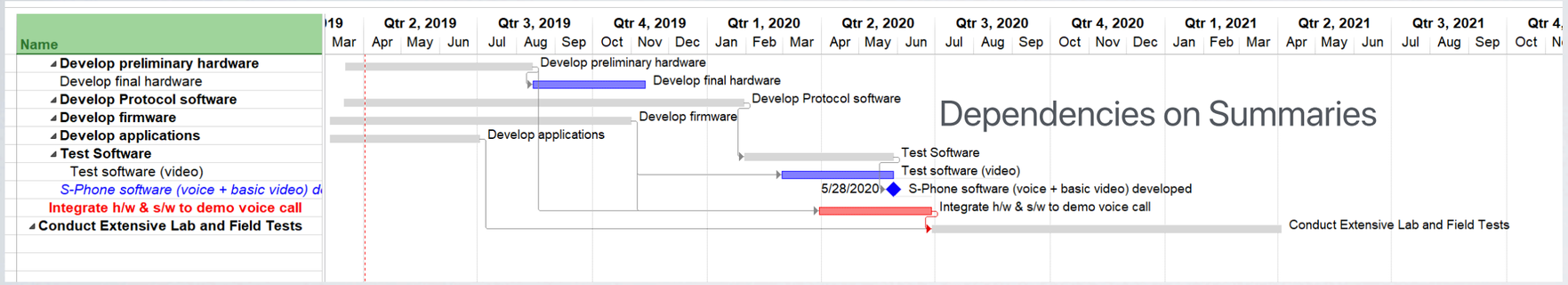
- Missing dependencies prevents the critical path flowing through subtasks of the WBS hierarchy.
- This limits pull-in opportunities and can cause negative float in some conditions.



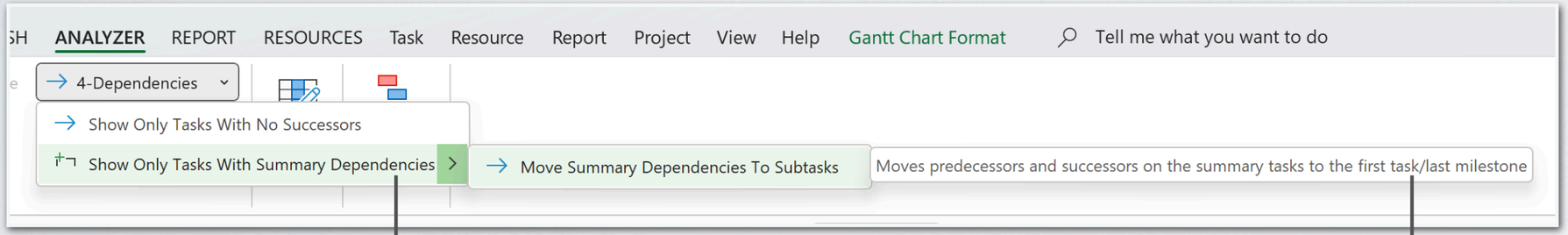


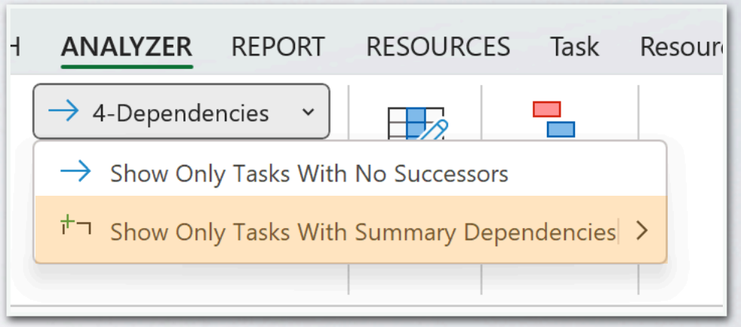
Hint: if a task doesn't link to another subtask within the work package, but belongs within the work package, highlight the task with a missing successor, right mouse click and select "Link to Last Milestone". This will automatically link it to the last milestone in the work package.

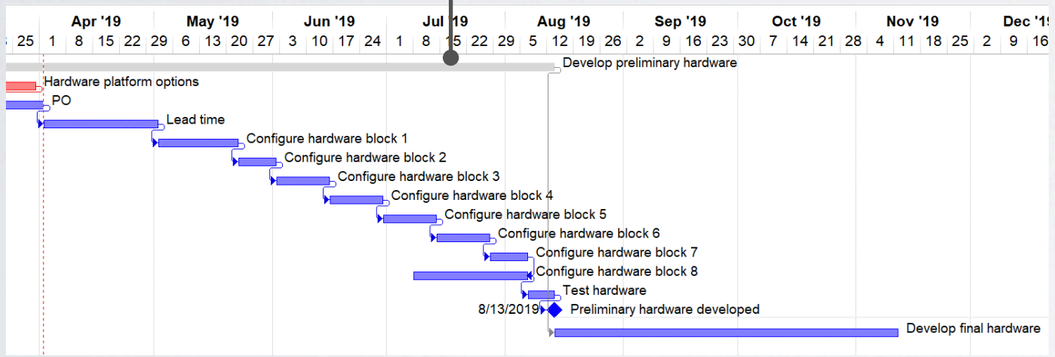
5. Dependencies



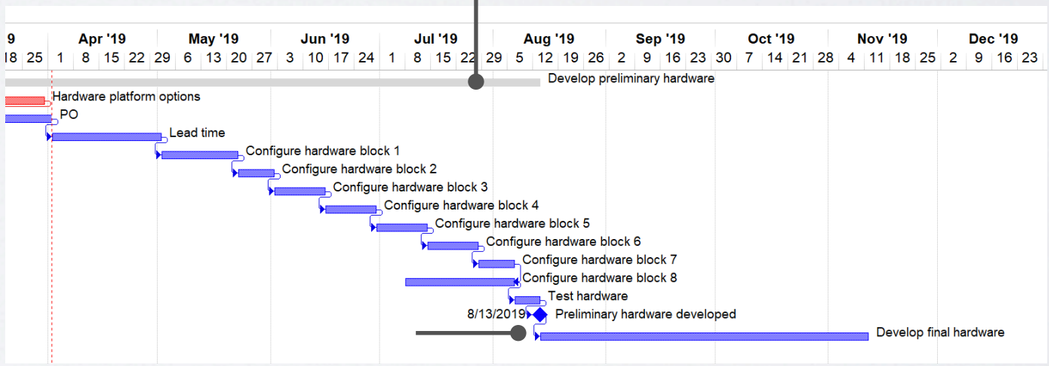
- Dependencies on summary tasks prevents the critical path flowing through the subtasks.
- This prevents any subtask from starting earlier, limits pull-in opportunities and can cause negative float in some conditions.







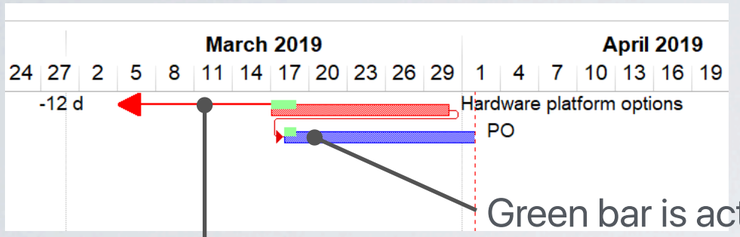
Before Auto-fix



After Auto-fix

5. Negative float

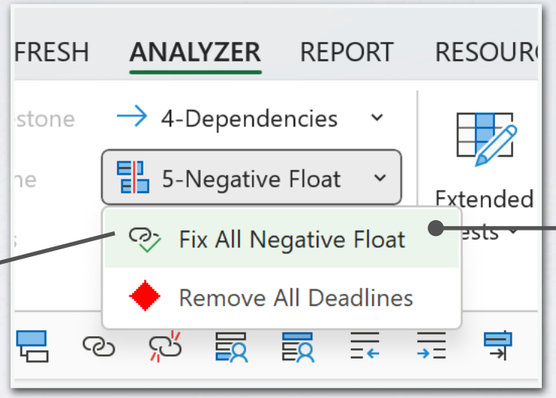
Defined as the Late Schedule bring earlier than the Early Schedule. This must be fixed in order to get an accurate critical path/schedule.



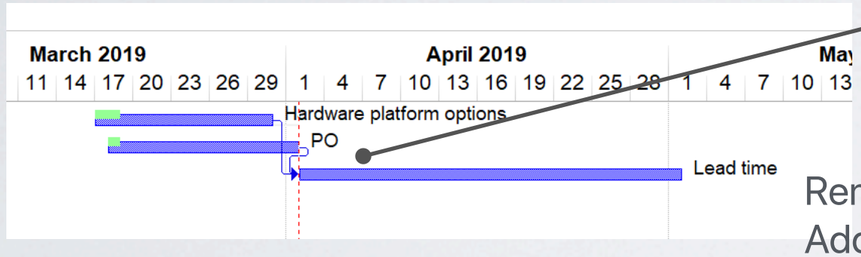
Caused by "PO" starting early, before "Hardware platform options" completed per the FS dependency

Green bar is actual start and actual progress

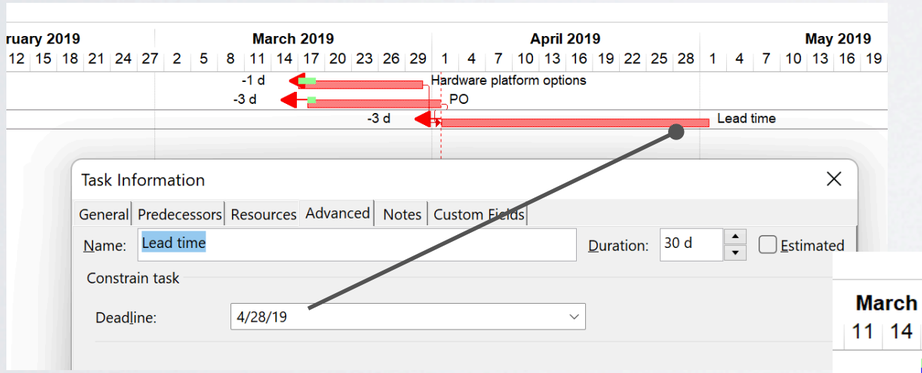
Negative Float indicator



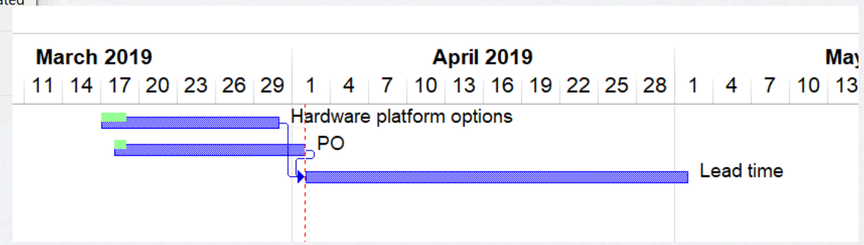
Auto fix



Removed FS from "HW platform options" to "PO"
Added FS between "HW platform options" and "Lead time" permitting predecessors to "Lead time" to run in parallel per the actual start update

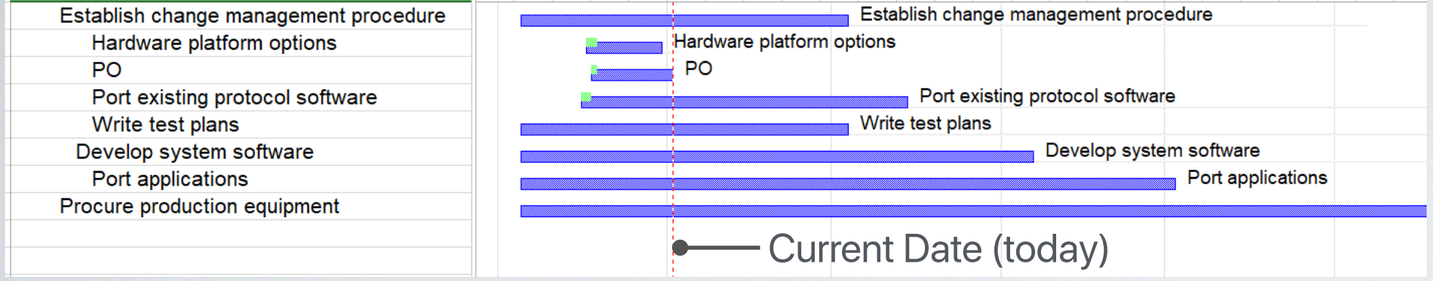


A Deadline was added to "Lead time" which caused negative float.
Deadline behave like hard date constraints.
Auto fix removed Deadline



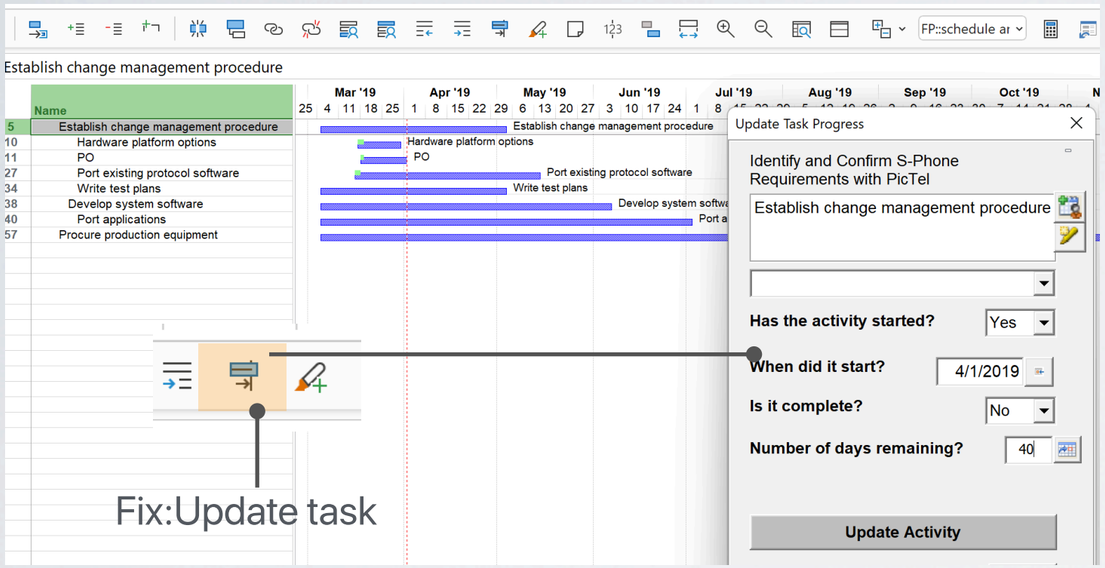
No Deadline

6. Not Updated

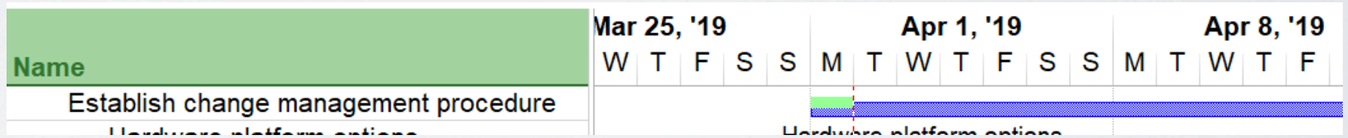


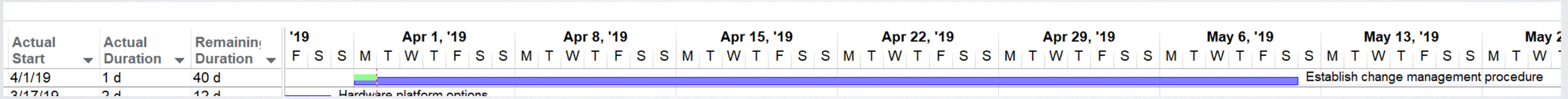
All need to be updated to the current date or not started and moved to the right of the current date

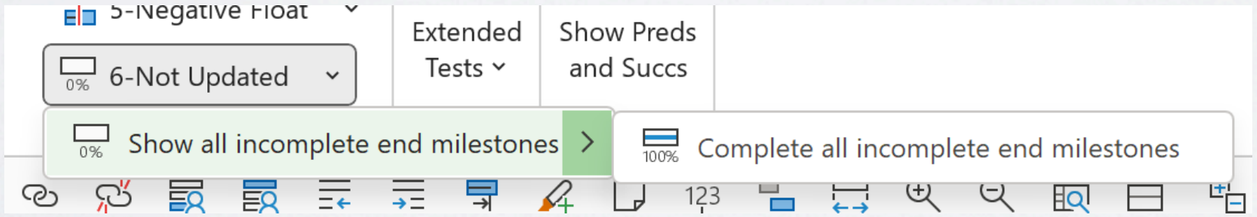
- Incomplete or not started tasks prior to the current date distorts the timeline. When updated, these tasks could be on the critical path.
- The Analyzer looks for tasks that have not been updated for over a week (it will also catch tasks added to the schedule and not updated).



When fixed, it looks like this...



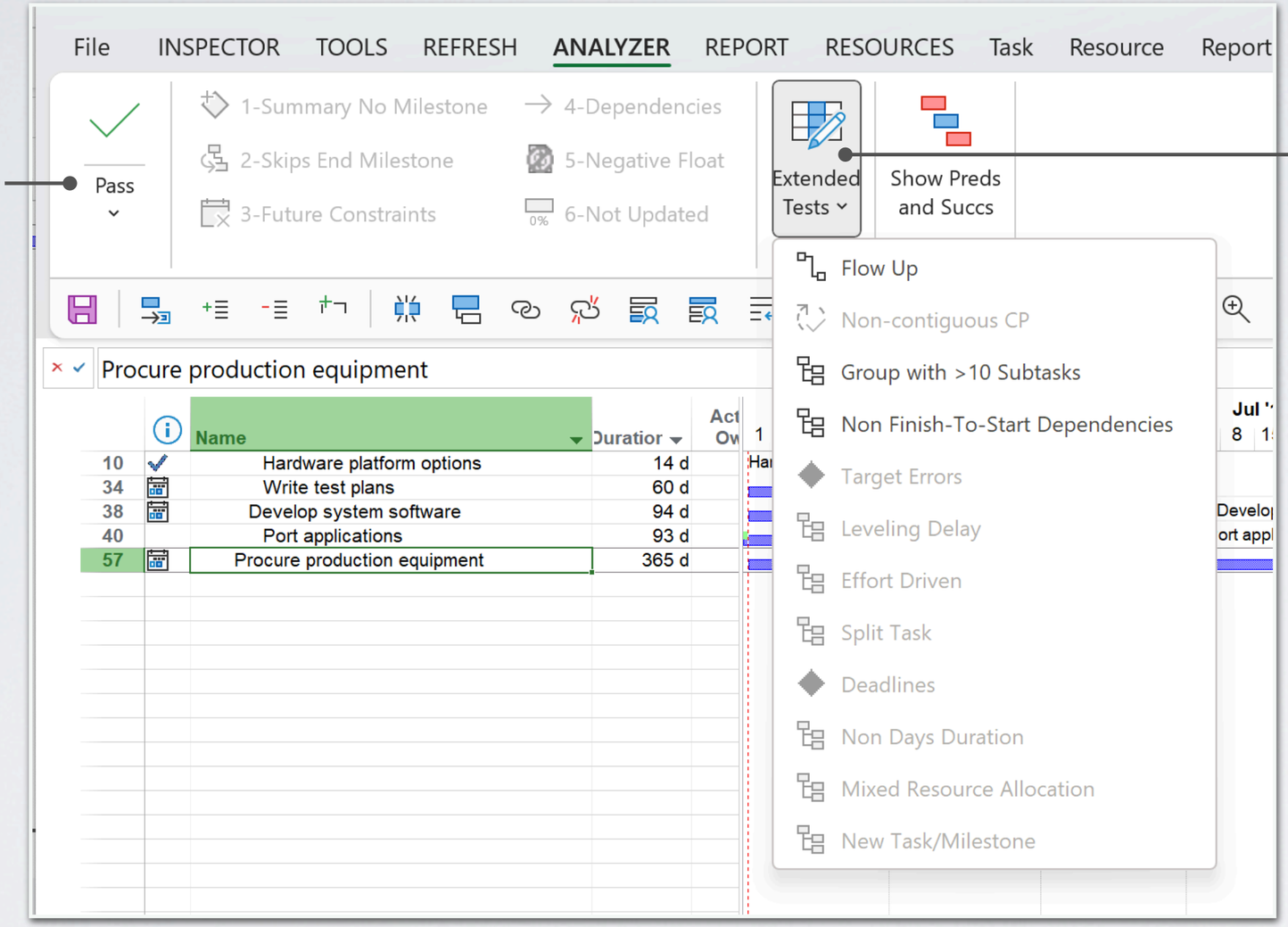




Will also find (and complete) milestones that are incomplete when all of its predecessors have been completed

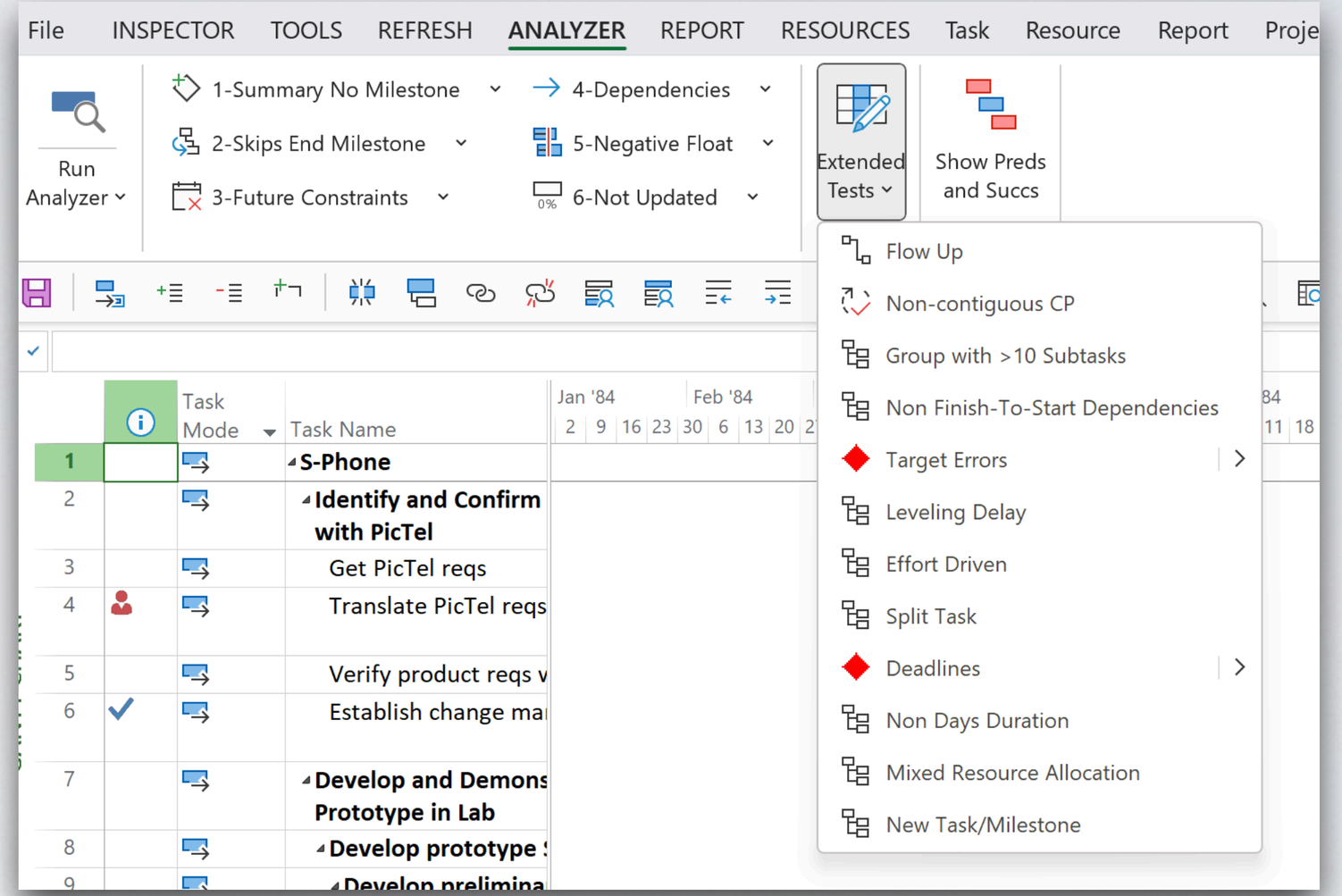
Passing the Core Tests

Core Tests Passed

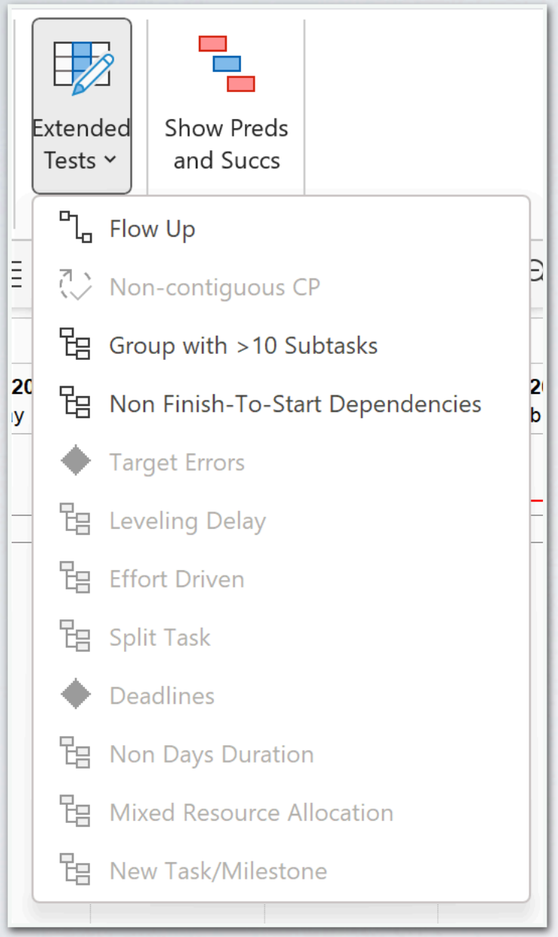


It's still a good idea to fix the Extended tests (those that are not grayed out), but these do not get scored in the Pass/Fail grade

Extended Tests



Extended Tests



- **Flow up**: Indicates a hierarchy problem. Dependencies should flow from top to bottom, not bottom to top. Fix by moving tasks so the schedule flows down or, in the worst case, restructure the schedule.
- **Non-contiguous CP**: Generally caused by 1) A date constraint forcing the CP to start in the future or 2) A Positive FS Lag. Fix 1) by either removing the date constraint or adding a Touchpoint. Fix 2) by removing the positive lag and replace with a task.
- **Group with >10 Subtasks:** More than 10 subtasks are hard to follow. Fix by adding and additional hierarchy to group tasks that are related to each other.
- **Non Finish-to-Start dependencies:** We are looking for Finish-to-Finish (FF & SF) dependencies. FF will pull the successor task to its late schedule, thereby loosing the opportunity to start it sooner. Fix by replacing FF with an FS. The question is not when it will finish but when is the earliest time it can start?
- **Target errors:** Finds problem with targets (multiple problems). Running the auto-fix from the menu will fix most common types of problems.
- **Leveling Delay:** Finds leveling delays, added either manually or through the Resource leveler in MS Project.
- **Effort Driven:** Finds “Effort Driven” tasks. These should be changed to be “not” effort driven (uncheck).
- **Split task:** Finds Split Tasks. Fix by changing the duration to zero then back to it correct duration. Re-update if the task has started.
- **Deadlines:** Finds all tasks that have a deadline. Remove the deadline and replace use a Target instead.
- **Non Days Duration:** Finds all tasks that do not have a duration in “days”. Fix by changing the duration to days. This is automatically corrected when you run the Analyzer.
- **Mixed Resource Allocation:** Finds resources that are allocated to both the summary task and its subtasks and is therefore double-counted. The fix is to either remove the resource from the summary or its subtask.
- **New Task/Milestone:** Finds these tasks/milestones that have been added to the schedule but not yet defined. These are so-called “construction sites”.