TRK.000

The Half-Step Schedule is Just the Start
Introducing the Zero-Step Schedule

by Ron Winter, PSP

March 1, 2013

Abstract
Mash-ups aren't just for the young – Schedule Forensic Experts and Schedulers are beginning to see the value in combining two different schedule updates into a third that is a logical combination of the two. AACE Recommend Practice 29R-03 already recognizes the Half-Step Schedule. This schedule 'mask-up' will be presented along with an entire taxonomy of schedule 'stages' that eventually lead to a complete update. Understanding the stages helps schedule analysts to isolate and define the cause based upon the symptom.

This paper will also introduce Half-Step's 'Evil Twin'; the Zero-Step Schedule. Instead of 'removing' the Update Schedule's logic changes as the Half-Step process prescribes, why not just add those logic changes to the Target Schedule and create a Zero-Step Schedule? A Zero-Step Schedule would include no new status, just the updated means and methods of the future plan. Instead of a What-If Schedule, why not consider a What-Was Schedule? The results can be startling. This brand-new concept may change the way you perform schedule analysis.

Keywords: Scheduling, Half-Step, Bifurcation, Zero-Step, taxonomy, schedule development, and What-If, What-Was

Table of Contents

Introduction

Mash-ups aren't just for the young – Schedule Forensic Experts and Schedulers are beginning to see the value in combining two different schedule updates into a third that is a logical combination of the two. AACE Recommend Practice 29R-03 already recognizes the Half-Step Schedule. This schedule 'mash-up' will be presented along with an entire taxonomy of schedule 'stages' that eventually lead to a complete update. Understanding the stages helps schedule analysts to isolate and define the cause based upon the symptom.

This paper will also introduce Half-Step's 'Evil Twin'; the Zero-Step Schedule. Instead of 'removing' the Update Schedule's logic changes as the Half-Step process prescribes, why not just add those logic changes to the Target Schedule and create a Zero-Step Schedule? A Zero-Step Schedule would include no new status, just the updated means and methods of the future plan. Instead of a What-If Schedule, why not consider a What-Was Schedule? The results can be startling. This brand-new concept may change the way you perform schedule analysis.

Taxonomy

Before we can begin to explain the results of our process, we first need to take a clearer look at the real 'life-cycle' of a schedule update. There are several distinct phases and properties of an update that are routinely grouped together. We will see that the Half-Step Schedule is one such phase. There are several other formal phases which need to be clarified before we can fully understand the complexities of a full schedule update.

- Target Schedule
- Conformed Target Schedule
- Half-Step Schedule
- Activity Duration Adjusted Schedule
- Logic Adjusted Schedule
- Added/Deleted Node Adjusted Schedule
- Computational Adjusted Schedule

This taxonomy of schedule development can be considered, Schedule Mash-Ups: The steps necessary to advance a complex, jumbled combination of the description of progress, problems and revised plans. This is the gulf between Then and Now.

Target Schedule

The Target Schedule can be a Baseline Schedule or last month's update. The review to be performed is based mostly upon the premise that the Target Schedule was accepted as representing the status of the project as of that status date. The review to be performed involves comparing the changes and differences between the two and to determine what actually happened in the time interval between that represented by the Target Schedule and the next update (called the Update Schedule.)

The Target Schedule must be a CPM schedule presented in native format. This means that it is not a static picture (such as a PDF file) but a fully configurable CPM schedule in the format used for the Update Schedule.

A formal submittal of the Target Schedule is preferred[1]. You do not want to spend a week reviewing a schedule that later turns out to be a What-If schedule or an incomplete working copy. Formal submissions also drive home the issue that accurate progress dates are required.

Conformed Target Schedule

Before performing any analysis, we need to make sure that the Target Schedule is 1) usable for further analysis and 2) does not contradict any later information. RP FSA [2] calls this process, "rectification." Engineers call this process, Conforming.

Many schedule reviews grant conditional acceptance providing that the next update reflects requested corrections. Before proceeding with the analysis, we must first modify a copy of the schedule to incorporate those corrections. We recommend that the corrections made to the Conformed Target Schedule match those corrections that were performed in the new Update Schedule.

In addition, if the Target Schedule is in a different native format from the Update Schedule, then you must decide whether to convert the Target Schedule to the Update's format or to convert the Update Schedule to the format used by the Target Schedule. Either way, you should compare durations and CPM settings for accuracy of the conversion. The 'standard' method of comparing dates will not be effective as the two schedules will be using different data dates and have different status.

The 'history' of the project as shown in the Target Schedule should match the 'history' shown in the Update. All past, completed events in the Target Schedule including activity descriptions, durations, and CPM logic must match that same history in the Update Schedule. While changes in planned versus actual can be anticipated, there should be no changes between actual data in the Target Schedule and actual data in the Update Schedule.

In particular, all actual dates in the Target Schedule must match those in the Update Schedule. Even actual dates can affect some CPM calculations. Logic between completed activities can affect the scheduling of planned activities. Even if it can be proved that no appreciable effect was produced, you still do not want to deal with two different versions of the same thing. The basic rule of record-keeping is that uncertainty in the schedule translates into uncertainty in the analysis.[7]

Activities with actual dates registered that are later than the current schedule's Data Date (i.e. Dates in the future) are also problematic and must be corrected. Actual Dates in the future interrupts and falsifies CPM date calculations. Oracle/Primavera P6™ and Microsoft Project™ both allow this condition to occur.

Since Microsoft Project computes all activities as if they were in the future, actual dates in the future are not an especially odious condition for it. With P6, both Retained Logic and Progress Override CPM calculation modes handle actual dates in the future incorrectly. They only look upon a completed activity in the future as a zero-duration activity and schedule all succeeding work to occur after this incorrect duration has transpired.

This is perhaps why P6 incorporates a third computing option called, Actual Dates. The use of the Actual Dates CPM setting carries its own set of problems. We advise you to just correct the actual dates in the future condition by re-statusing the activity as incomplete and adjusting the Remaining Duration.

Out-of-sequence status may signal a possible status mistake or logic error. Sometimes, planned work is just performed piecemeal. Other times, out-of-sequence progress may reveal a statusing oversight that needs to be corrected. More problematic is when the logic does not reflect the logical constraints in the field. The general consensus is that the logic should be modified to reflect the As-Built logic used.

Lastly, every date constraint assigned to the project, an activity, or resource must be reviewed. Just because a constraint existed in a past update and is unchanged does not mean that its existence is still justified. Constraints are difficult to spot in listings and can be 'sleepers'; inactive over a long period only to pop-up and control the timing of project completion late into the project. Constraints can be used as a method to sequester float.

Once the Target Schedule is conformed, an analysis must be made to see if any significant shift in the critical path or near-critical path was observed. Common wisdom says that the Conformed Target Schedule represents the best depiction of the project's status as of that status date. We will see later that he Zero Step Schedule may be a better choice.

Sadly, a Conformed Target Schedule is not truly complete until the final As-Built Schedule is submitted and the dates checked against project logs and other records. Any errors that can be caught early will prevent more work and errors generated later.

Forward-Looking Models

The purpose for identifying the following taxonomy is to 'simplify' the changes in a schedule between the conformed Target and the Update Schedule. Conceptually, we want to break-up a typical schedule update into various stages. Each new stage will introduce another layer of complexity. Each layer is added until you reach an entire, full schedule update.

With each new schedule layer, we will analyze the results to the critical path and project completion. We want to identify the first modification to the schedule that changes either the critical path or project completion as this modification is the 'cause' for the overall schedule

change. The responsibility for this change can be also attributed to the responsibility for the overall schedule change.

In addition, a later stage in the schedule update process may erase or modify the results found in an earlier stage. We should map these changes and results to better be able to tell the story of what really happened over the course of a schedule update.

The first stage in a schedule update is the addition of progress observed over the update period in question. Note: Some Claims Analysts use RP 29R-03 MIP 3.3, "As-Is" method that advocates not modifying the Target Schedule. This technique is not a good candidate for creating Half-Step Schedules or any of the forward-looking schedules that we will discuss here.

Half-Step Schedule

The formal term for this type of Schedule Mash-up is the Bifurcated Contemporaneous Period Analysis, as defined in AACE Recommended Practice 29R-03 "Forensic Schedule Analysis" [2]. The goal of this process is to create a copy of the Update Schedule with the Update's later data date that reflects only the status change reported. All other changes in the Update Schedule are not included in the Half-Step Schedule.

The Half-Step Schedule isolates the current status from the other possible modifications made to the schedule during the schedule update process. This 'strips-out' all possible scheduling mediations that might be made to recover lost time. The changes in calculated dates and total float between the Conformed Target Schedule and the Half-Step Schedule reflect the changes caused by the contractor's progress in the period in question.

If the critical path changed between the Conformed Target Schedule and the Half-Step Schedule, then the change is the direct result of the work performed in the update period. Other changes must be assumed to have occurred due to a change in the project network or plan.

Changes to the critical path displayed between the Half-Step Schedule and the Update schedule reflect a possibly complex set of modifications that are difficult to isolate when lumped together. These modifications can be further broken-down into stages. The following logical stages describe some of the intermediary steps between these two schedules.

Activity Duration-Adjusted Schedule

One step further in the update process beyond the Half-Step Schedule is adjusting the planned Remaining Durations of activities to reflect changes in the workplan[3] or perhaps a better understanding of the work process. Changes made to individual activity durations are generally simpler to extrapolate than those made to the logic.

One might change the Original Duration (and thus, Remaining Durations) for unstarted activities. This will certainly affect the CPM calculations, at least locally. It is possible for a

scheduler to reduce the Original Durations of selected activities in order to 'recover' schedule slippage or even in an attempt to 'steer' the critical path.

The Scheduler may be thinking that the team will 'just work harder' or that they do not need the 'built-in' extra time that may be inherent in the activity's duration estimate. Realistically, such changes in activity original duration should also be accompanied with a stated workplan change; increased crew sizes or longer working days. The difference from the Half-Step Schedule and the Activity Duration-Adjusted Schedule should highlight the impact that changed activity original (and thus remaining) durations have to play in the update.

Logic-Adjusted Schedule

Adjusted logic may just be the result of a shifting of resources. The logic may be 'soft', or preferential to make the schedule more realistic by not scheduling the same crew to work on two different activities at the same time. Such a logic change could simply be a resequencing of existing work.

Small changes such as inserting a small negative lag may signal the real or artificial compression of the work or it may just reflect intended resource overlap found on accelerated jobs. Overlap may be more likely if the same primary resource is assigned to both activities. This would indicate that the lead or prep-work is being performed while the finish work is being completed or waiting for inspection.

Larger logic changes, especially those far removed from work occurring near the Data Date may indicate a change in the work or a change in the project constraints. Such changes demand an explanation in the Narrative submitted with the Update Schedule. Without an explanation, it is very difficult to evaluate the reasonableness of the logic changes.

Creating the logic changes separate from the Update Schedule is actually easier than first comes to mind. All that you need to do is run a comparison program such as the included Claim Digger to compare the Activity Duration-Adjusted Schedule and the Update Schedule and use the Added/Deleted/Modified Relationships reports as a checklist of the changes to be made.

Added/Deleted Node Adjusted Schedule

Deleting an activity involves much more that a change in duration. Whenever an activity is deleted, all relationships to or from that activity are also automatically deleted, with the exception of the use of P6's 'dissolved' activities function.[4] Manual modifications to existing logic are preferable to this automated process as each changed logic tie should be considered on a case-by-case basis.

Modern CPM scheduling software also considers other issues other than duration and logic to determine planned dates. Task resources assigned to the deleted activity may affect the timing of other activities due to resource leveling.

The same issue applies to that of added activities. Added activities without logical predecessor and successor relationships seldom affect other activities other than with possible task resource leveling issues. Added activities with added logic can have far-reaching consequences to the CPM, even if the activity is initially statused as complete. Adding activities and logic that are initially statused as complete can also affect the critical path.

If any of the preceding states causes the project-changing event to occur, then one can concentrate on the reasons behind that one change to determine if the change was reasonable and thus the consequences of that change.

Computation Adjusted Schedule

The final stage in creating an update schedule has to do with the schedule environment. The difference between an Added/Deleted Node-Adjusted Schedule and a full Update Schedule can be immense. The list of differences can include changes to:

- CPM settings
- Administrative settings
- Calendars and calendar settings
- Activity Type setting
- Activity Duration Type setting
- Activity Constraints (should have been covered during Conforming Target Schedule)
- Activity Resource Status
- WBS Earned Value Settings

Finally, one should look for and correct schedule importing errors [5]. There are several ways to import schedules and data to update existing schedules. Such facilities in P6 software have displayed errors in the past and continue to be plagued with errors now. All updates should be checked for accuracy of intent.

Problems with the Half-Step Schedule

Interconnected functions within a schedule can work against the analyst 'in the background.' While trying to remove status, several activity resource functions may override changes made to activity status. First off, activities cannot be statused as "Not Started" if actual dates exist for any of the task resources. This is very plain to observe when manual changes are made. Errors of this nature are much more 'invisible' when automated, batch update methods such as Excel Import or Global Change are used.

More difficult to overcome is the affect of automatically recalculating resource actuals. Regardless of Task Type settings, P6 maintains a functional balance between Actual Units, Original Units, and activity duration. To prevent this readjustment of actual units from automatically changing activity duration, one needs to address the Project's WBS Section, Calculations Tab; "Recalculate Actual Units and Cost when duration % complete changes" setting and un-check this setting assignment. Without this change, P6 will cause activity actual units to repopulate after you have set them to zero, disallowing actual changes.[6]

There are ways that a Half-Step Schedule can 'break-down' and not represent the clear demarcation between pure-status and a logic update. The primary ways involve a Change in the Work and a Change in the Logic.

A change in the work occurs when the work envisioned in the baseline schedule is no longer the intended work. This can arise due to increased knowledge such as if anticipated preparation work is no longer needed or if an un-planned delay occurs preventing further planned work.

An example of the change in work might be where 2 days into a 5-day excavation task, the remaining duration is modified to indicate 8 days remaining due to hard soil conditions. Should the schedule indicate that we are 40% done or 20% complete? Alpha Corporation's bifurcation process [3] specifically forbids an increase in Remaining Duration that is greater than Original Duration. Unfortunately, their procedure does not include this check and does not describe the correct process to handle this situation. Perhaps they are counting on the internal edit checking used by P6 that automatically rejects increases to Remaining Duration in excess of Original Duration.

A change in the logic can represent the As-Built work process. How valid is the analysis of the results of progress using invalid logic that does not represent the way the project was built?

If you input the activity status changes without the corresponding logic change, it is possible to create an out-of-sequence progress situation where none actually exists. At that point, you are leaving it up to the CPM algorithm to determine how to apply the CPM calculation.

If the Half-Step schedule uses the Progress Override setting, then you are telling the computer to automatically ignore the existing logic in out-of-sequence situations. How is this different from manually modifying the logic? In other words, you do not have a true Half-Step schedule if you are using Progress Override and have out-of-sequence progress.

Backward-Looking Modifications

So far, the processes described here have all been 'additive'. We have been adding the changes made (either intentionally or as the result of other changes) consecutively from earlier events to later ones. We have chronicled the 'building-up' of the project.

This process of 'building-up' assumes that we knew exactly where we were and where we were going. Sadly, in a complex process of building a project often times we later discover that things that we thought were important were not so important and other issues were actually the controlling constraints in the project.

Sometimes, this is due to unforeseen or unexpected changes to the project beyond our control. Other times this misunderstanding is due to unintentional lack of due diligence in the planning of the project or even due to intentional misdirection. These types errors will later come to

light but seldom do we look backward and evaluate how the newly-discovered information would have affected the project and its administration in the past.

What is needed is a formal process to allow one to review what the project would have looked like if the changes made later had actually been implemented earlier. To do this, we need to re-create earlier schedules with the 'correct' plan incorporated. This is what Zero-Step Schedules represent.

The Zero-Step Schedule

Zero-Step schedules are a What-If schedule created when we add in every change made in the Update Schedule except for the status updates. Luckily, the process of creating a Zero-Step Schedule can be simplified. Instead of adding everything but the progress changes to the Target Schedule, why not just remove the progress changes from the Update Schedule? Then we change the Data Date to reflect the earlier stage of progress and re-compute the CPM.

Issues Involved in a Successful Conversion

As with our technical warning about schedule CPM settings for the Half-Step Schedule, P6 analysts should un-check the Project's WBS Section, Calculations Tab; "Recalculate Actual Units and Cost when duration % complete changes" setting. This will prevent a change in a resource's status from automatically causing a change in the activity's duration.

Next, one must remove activity resource status before removing the activity's status. P6 will not let you 'un-set' the Actual Start date of an activity if any actual information exists in either activity resource or costs.

You must be sure to remove any External Constraints that did not exist in earlier schedules. External Constraints are automatically inserted by P6 when external relationships exist to activities in other schedules and the schedule is copied or exported without coupling the other schedules with it.

Activity types must be set to be duration-driven and not resource-driven. It is possible to properly change the status of a resource-driven schedule by changing the resource status, but this is a tedious and error-prone procedure. It is much easier to use duration-driven status updates to affect the Zero-Step process.

Expected Finish Dates should be removed from activities that did not have this constraint set in the Target Schedule. Such constraints will only make sure that the schedule does not change, even if this is warranted. In general, all activity constraints should be considered for appropriateness and deleted if it is determined that they are not.

Finally, added or deleted work scope that occurred between the two status dates must be dealt with on a case-by-case basis. All assumptions and corresponding changes to the schedule should be clearly documented. Analysis using a Zero-Step Schedule is most applicable when few, if any unusual changes were observed to the scope of work during this period.

Interpreting the Zero-Step Schedule

There are "Dos and Don'ts" involving the proper use of Zero-Step Schedules. To be able to in interpret the results, Zero-Step Schedules should not be credited if one or more significant, unexpected events occurred during the update period.

It is generally not appropriate to use the results of a Zero-Step Schedule analysis if an event has occurred during the update period that could not have been reasonably anticipated. Changes to the workplan caused by on-going events are certainly part of the expected course of a project. The fact that a certain piece of work was given lower priority at one time and a higher one later based upon a response to an unforeseen problem is a normal event and does not disqualify Zero-Step use.

A Zero-Step Schedule can be realistically used to re-interpret the past in situations where conditions have not changed significantly and the failure to anticipate normal construction issues is not a sufficient reason to use incorrect data for compensation or corrections. This situation would also encompass situations where misleading information was intentionally included or left out of the schedule to improve one's negotiating position. In these cases, a Zero-Step Schedule might give new insights as to the real situation at the time, had sufficient information been made available.

In these cases, it may be appropriate to create a Zero-Step Schedule to determine where the real critical path was and what really impacted project completion. Late-arriving information does not have to be 'lost' information. One does not have to wait for a post-project claim analyst to determine what really happened.

In fact, a Zero-Step Schedule might be 'propagated' backwards in time through many such updates. The As-Built logic and schedule structure might be applied to all previous schedules to determine the true project critical path. If the official status-keeper was modifying logic to reflect the As-Built condition, is it not reasonable to use that same information to verify older findings throughout the project's history?

Assuming that a Zero-Step Analysis was appropriate, where would the critical path have gone last month if we knew then what we know now about the planned schedule logic? Could a Zero-Step schedule analysis invalidate Time Impact Analyses and their awards? Is the Zero-Step Schedule a better way to identify and prove where the As-Built Critical Path actually went? These questions have not been answered, just asked.

Conclusion

A Zero-Step Schedule is a hybrid CPM schedule that displays the logic and planned duration changes used in future schedules but not the progress. This creates an opportunity for the analyst to re-evaluate the past using knowledge of the future; to really see 'What Was'. This 'Zero-Step Schedule" amplifies the reviewer's ability to comprehend the importance of the actual plan when looking at the actual status. Instead of evaluating what 'could have

happened', the analyst can consider what actually did occur in the framework of the way a project was actually built.

To fully understand the context of a Zero-Step Schedule, one also has to appreciate the various stages or divisions of a schedule update. These formal stages are mapped-out and defined in this paper. These stages are usually described as proceeding from the Target schedule, gathering complexity until the end result is an Update Schedule. This paper has shown that the reverse process can also be applied.

References

1. O'Brien & Plotnick, *CPM in Construction Management, Sixth Edition*, McGraw-Hill, 2005

2 Section 3.4 K 4, *Bifurcation*, **Forensic Schedule Analysis**, AACE Recommended Practice No. 29R-03

3. **Workplan**: University of Colorado Bolder, http://www.colorado.edu/engineering/civil/CVEN4434/resources/workplan.html [copy supplied]

4. Footnote: P6 Activities can be "Dissolved" instead of just deleted. In this case, the deleted activity's predecessor and successors are automatically linked with a Finish-to-Start relationship. Primavera P6 Project Management Help, Version 7.0.

5. Patrick M. Kelly, PSP: CDR.966, *Creating Mid-Period Schedules in Window Analysis Using Primavera Version 6*, **2012 AACE INTERNATIONAL TRANSACTIONS**

6. *PROCEDURE for PRIMAVERA PROJECT MANAGEMENT (P6): Bifurcation: Creating a Progress-Only Half Step Update*, **Company Guideline** by Chris Carson, PSP, CCM, PMP, Alpha Corporation [Unpublished, copy supplied] (not dated)

7. Ron Winter, PSP: *119 Specific Checks to Perform When Reviewing a Schedule*, 2003, **PRIMAVERA 20th ANNUAL CONFERENCE**

Ron Winter, PSP
Schedule Analyzer Software
Ron@RonWinterConsulting.com