PS.2324

Evaluating CPM Schedules for Best Practices: A First Principles Approach

J. Gerard Boyle[1]; Andrew Podolny; and Dr. Wail Menesi

Abstract—There are now several guides meant to assist the industry in evaluating critical path method schedules. For example, various analytical programs have been built to check for compliance with the DCMA 14-Point Schedule Metrics which are intended to assist in finding indicators of schedule problems. The GAO Schedule Assessment Guide offers ten best practices associated with a high quality and reliable schedule. Although there are many similarities amongst these offerings, there are also significant differences which have not been resolved, and this lack of consensus has not benefited the practice of scheduling. The first part of this article makes the point that some of the lack of consensus can be resolved by rigorously imposing a clear definition of what a CPM schedule is. Once this is established, certain essential characteristics present themselves as sine qua non for a CPM schedule. Even with clear definition of the minimum requirements, scheduling software presents unique challenges which must be overcome. The next step after establishing that a schedule is CPM compliant is to establish best practice criteria.

Table of Contents

List of Figures

List of Tables

Introduction

In 2003, four leading schedule experts were interviewed in the Engineering News-Record (ENR) to discuss the state of Critical Path Method (CPM) schedules [1]. A major concern they shared at the time was the sub-standard quality of CPM schedules. They observed that what was presented as a CPM schedule frequently "*wasn't one at all*", and criticized the "*widespread abuses of powerful software to produce badly flawed or deliberately deceptive schedules that look good but lack mathematical coherence or common sense*". Exacerbating the problem, in their opinion, were "*inexperienced and poorly trained practitioners"* and software features that were vulnerable to abuse.

The experts concluded that as a result of the prevailing poor practice, there was "*confusion, delayed projects and lawsuits*", and that instead of being an important planning and control instrument, schedules were being used as "*tools for claims*". They urged a return to fundamental principles in CPM.

A decade later, at an annual conference of the AACE, a debate was held on the subject of the quality of CPM scheduling. Reference was made to the ENR article, and the general consensus of the panel was that CPM schedules had not improved over the intervening years. Given the lack of progress, and the seeming intractability of the problem, the question of whether CPM might be replaced by some other method was discussed [2]. A primary conclusion of the debate was that if "*CPM scheduling is to maintain its place as the leading time management technique in the engineering and construction industry*" then the "*quality of schedules must improve*" and "*meaningless schedules*" cannot be the norm.

Indeed, the significance of this problem for the construction industry[2] in terms of project time and cost over-runs and unnecessary or protracted litigation cannot be over-stated[3]. How sustainable is a situation where productivity continues to drop and disputes are taking longer to resolve?

The many reasons for the current malaise include: industry inertia; lack of expertise, scheduling software issues; lack of commitment to a scheduling program, insufficient resources, etc. But it is contended here that one part of the solution is return to first principles. Evaluating a schedule for best practices[4] before determining that the schedule actually meets the definition of a CPM schedule is putting the cart before the horse. This article describes a two part approach to evaluating CPM schedules for Best Practices.

Schedule Standards and Best Practice

In response to the need to improve schedule quality, a number of best practice schedule standards have emerged. A prominent example is the *Schedule Assessment Guide – Best Practices for Project Schedules* (GAO Guide) recently released by the *Government Accountability Office* (GAO) [3]. The GAO Guide provides "*ten best practices to help managers and auditors* *ensure that the program schedule is reliable*", with the goal of improving program performance. Another approach is offered by the *Defence Contract Management Agency which* has included, in its *Earned Value Management System Program Analysis Pamphlet*, a "*14-Point Schedule Metrics for IMS (Project, Open Plan, etc.) Analysis*"[4]. Various organizations and agencies have drawn from these and other practice standards in an effort to arrive at a hybrid solution which cherry picks from each and best suits their particular requirements [6] [7].

The Missing Step in Best Practice Schedule Assessment

It is evident from the above that the industry is not in agreement on what schedule best practice should be. It is suggested that a reason for this is that all of these approaches proceed to best practice evaluation before verifying that the schedule even meets the test of being a CPM schedule. The remedy proposed here is a first principles approach whereby a schedule is required to meet a *minimum* standard of CPM compliance.

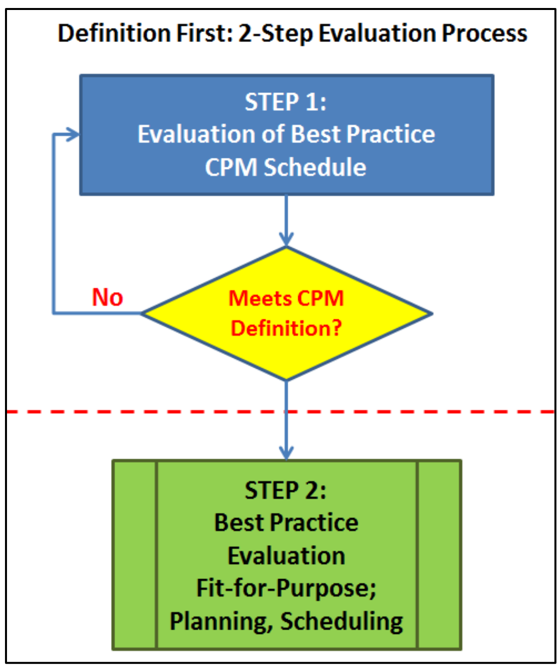


Figure 1 – Two-Step Evaluation Process for CPM Schedules

Figure 1 depicts the suggested two-step approach to evaluate CPM Schedules for Best Practices. First, the schedule is subjected to a test to determine if it meets the definition of a CPM. Only if it passes this test, by definition, can it then be evaluated for best practices. Determination of CPM compliance is largely a mathematical exercise, and therefore admits of much less subjectivity then the subsequent determination of best practices. Only after CPM compliance is confirmed does the schedule review process proceed (in Step 2) to a best practice evaluation.

CPM Definition

As noted above, the essential requirement for a schedule to be a CPM schedule is meet the *definition* of a CPM schedule. The Project Management Institute [8] provides the following definition, which is broken down into its constituent sentences along with commentary explaining the implication for CPM evaluation criteria.

- Def.-S1: "*A method used to estimate the minimum project duration and determine the amount of schedule flexibility on the logical network paths within the schedule model.*" Identifies a two-fold purpose: identify the longest (or driving) path, and; use float values to evaluate how much non-critical float paths can be delayed without becoming critical or delaying the project. This requires that *all* activities are logically connected in the network.
- Def.-S2: "*This schedule network analysis technique calculates the early start, early finish, late start, and late finish dates for all activities without regard for any resource limitations by performing a forward and backward pass analysis through the schedule network*." Float of all activities in the logically connected network is mathematically calculated using the forward and backward passes. Resource limits are not allowed to interfere with the float calculation because the network must be driven exclusively by activity logic.
- Def.-S3: "*The critical path is the sequence of activities that represents the longest path though the project, which determines the shortest possible duration*." The path (or paths) of activities in the logic-driven network which determines the end date because it is the longest path.
- Def.-S4: "*The resulting early and late start and finish dates are not necessarily the project schedule, rather they indicate the time periods within which the activity could be executed, using the parameters entered in the schedule model for activity durations, logical relationships, leads, lags, and other known constraints*. *The Critical Path Method is used to calculate the amount of scheduling flexibility on the logical network paths within the schedule model*." Makes the point clearly that the critical path, even though it gets the most attention, is not the exclusive focus in a CPM.

It follows from the above definition of a CPM schedule that the activity network created in CPM scheduling software must have the following characteristics:

1. **Closed-In Network:** Activities cannot have logical open ends and must have start and finish relationships: That is, all activities must have at least one predecessor and successor. The only exceptions are the first activity, which will not have a predecessor, and the last activity, which will not have a successor. Moreover, every activity, except the start and finish activities must have a start and finish relationship. (For example, if an activity has a Start-to-Start successor relationship with another activity, but no Finish-to-Finish, there is nothing, other than completion, constraining the finish of the activity). If these conditions are not met, the network is not completely "closed-in" and,

as a result, correct float calculations for all activities (and paths) are not available, and it is therefore not possible to evaluate schedule flexibility of all network paths.

2. **Logic-Driven Activities:** CPM scheduling software permits "constraints" to be applied to individual activities and milestones. Hard constraints prevent activities from starting or finishing later than planned. Mandatory constraints are even more rigid because activities are locked into a time place and float is eliminated. Both over-ride the schedule logic, which is antithetical to the CPM definition because the CPM relies on the calculation of early and late dates based on unimpeded schedule logic. In theory, Soft Constraints allow delays past the constraint date, and so do not over-ride logic, however, as will be discussed later, such constraints may result in distortions to the critical path. Automatic resource levelling may also over-ride activity logic. Finally, actual dates (on or after the Data Date) also over-ride schedule logic and therefore transgress the CPM definition.
3. **A continuous, logic-driven critical (or longest) path(s), which determines the shortest possible duration must be generated:** A primary requirement of a CPM schedule is to correctly calculate and identify the activities on the longest continuous path of *activities*. If a delay occurs to either the start of finish date of an activity on the critical path, a delay will result[5] to the overall project duration. In managing a project, the critical and near-critical[6] paths must be reliable in terms of indicating flexibility to alter timing of activities. If the critical path is not continuous, or if an increase to activity start of finish date does not alter the completion date[7], then the schedule is not CPM compliant.
4. **Appropriate logical relationships are used:** Activity logic must give effect to the three characteristics described above. The Precedence Diagraming Method provides for four types of logical relationships: Finish-to-Start; Start-to-Start; Finish-to-Finish; and Start-to-Finish. Of these, the Finish-to-Start is the default relationship which is preferred because it is straightforward, intuitively understandable and is less likely to produce unintended consequences. However, any of the available relationships can be used in a CPM so long as the relationship reflects the planned sequence of the work based on reasonable assumptions[8]. Although in some unusual cases a Start-to-Finish relationship, which means that a succeeding activity cannot finish until its predecessor is started, is appropriate, use of it should be examined to ensure it is correct.[9] Moreover, any relationship pairs used at the same time, other than S-S and F-F, should be suspect because the effect may violate CPM principles. For example, in a case where the duration of an activity with an F-F predecessor and S-S successor increases, the overall logic path actually "shortens"[10] in time because the F-F relationship is honoured [9].
5. **No Activities with Fractional Durations:** The improper use of multiple activity calendars can result in fractional durations which produce erroneous calculations of the completion date.
6. **No Alteration of Critical Path or Longest Path Filters:** Implicit in the definition is the understanding that if scheduling software is used to calculate the network float values, then the resultant critical path report based on the software algorithm(s) is used without alteration by manual intervention.[11]
7. **Appropriate Selection "Critical Path" Filter – (where the Critical Path is not identical to the Longest Path filter):** This applies to Oracle P6, but not to MS Project (which has only

one critical path algorithm). Depending upon the use of schedule features such as activity calendars, it may be that the P6 "Critical Path" filter is not identical to the P6 "Longest Path" filter. In such cases, so that all parties to the contract understand what is actually driving completion of the work according to the contractor's plan, the contractor must identify the actual critical path which is the path that determines the completion date.

Schedule Software Dilemma: What Is Critical?

Before proceeding to a discussion of the minimum criteria required to meet the definition of a CPM schedule, it is necessary to digress into a consideration of how CPM software identifies the critical path. Leading packages such as Oracle P6 and MS Project use different algorithms to identify the critical path and each is impacted differently by the use of constraints and calendars. The features of the scheduling software must be used (or not used) so as to give effect to the definition of the Critical Path which is, as mentioned earlier, the path that determines the length of the project.

Most would assume that so long as the sort of poor practices identified in the ENR Interview [1] (such as open-ends, mandatory/hard constraints, etc.) are avoided, the software simply does the math, and the result is a reliable critical path filter. But it turns out that for a number of reasons, this seemingly reasonable expectation is disappointed.

To begin with, multiple calendars can render Total Float values misleading in terms of identifying criticality. Microsoft project uses only a float-based calculation, and therefore does not address the problem of identifying criticality in the case of multiple calendars. However, Oracle P6 provides the following alternative algorithms to calculate the critical path:

1. "Critical" Filter: Using Total Float, a value, usually 0-days, is selected and only activities with Total Float less than or equal to this value will be shown as critical;
2. "Longest Path": By identifying "driving relationships" the software algorithm creates a string of directly related activities that comprise the longest path from the Data Date to the last activity in the schedule;
3. "Multi Float Path":
   Total Float Calculates critical float path and sub-paths based on relationship total float. In this case, the float path ranked number 1 is comprised of activities that drive the activity with the least total float.
   a. Free Float: Calculates the critical float path and sub- paths based on relationship free float. In this case, the most critical path is the longest path, or the path that drives the Scheduled Finish of the project back to the project start date.

The "Longest Path" filter is widely believed to reliably identify the critical path for schedules with multiple calendars. Indeed this is the assumption of the GAO Guide which states: "… *because the longest path makes no reference to total float, it is the only guaranteed*[12] *method* *of identifying the driving sequence of activities when using multiple calendars*." [3]. However, examples are provided below which demonstrate that the software does not always correctly identify the critical path in such cases.

Furthermore, there are other problems with both the P6 and MS Project critical path algorithms[13] which can result in errors where constraints and multiple calendars are used. Unfortunately, owing to certain functional problems with the most popularly used scheduling software programs (Oracle P6 and MS project are considered here), the resulting critical path filters must be checked to verify that the critical path is correctly constituted with driving activities. Examples of the software problems which create an added challenge in evaluating a CPM schedules are provided below.

1. Multiple Calendars & ("Soft") Constraints Can Distort the Critical Path

The effect of multiple calendars and constraints on the critical path is first considered using the P6 software. If the schedule is free of constraints, uses only one calendar, and is restricted to only finish-to-start logic, then the activity path generated by the "Critical" filter should not differ from the "Longest path" filter, and in fact this is the case in the (Case 1) example below. The Longest ("Driving") path filter shows both "Activity 1" and "Activity 3" as residing on the critical path, and the "Critical" path filter yields an identical path. "Activity 2" has two days of Total Float and two days of Relationship Total Float.

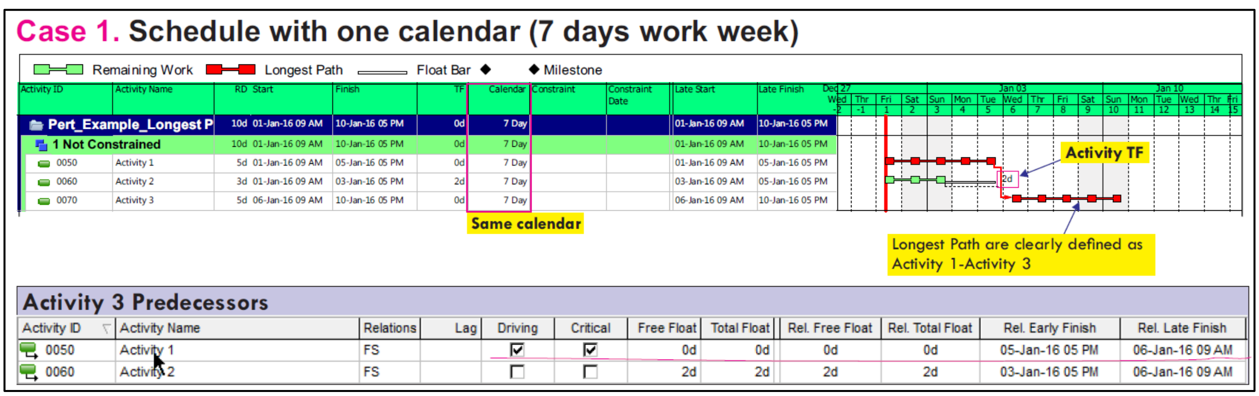


Figure 2 – Case1: P6's Critical Path and Longest Path Are the Same

However, when a "*Start On or After*" constraint is added to "Activity 3", as in Case 2 below, notice the resulting Longest Path. "Activity 1" disappears from the longest path and only Activity 3 is identified as critical: a continuous critical path is not generated. This result may be surprising to some as a "*Start On or After*" is considered a "*Soft*" constraint because it does not constrain the activity from starting later (i.e. it does not over-ride predecessor logic). "*Mandatory*" and "*Hard*" constraints are often excluded from best practice recommendations, but "*Soft*" constraints are generally considered acceptable[14]. The key point here is that even the use of a "*Soft*" constraint, widely considered to be acceptable practice, prevents the generation of a *continuous* critical path which, by definition, the critical path must have.

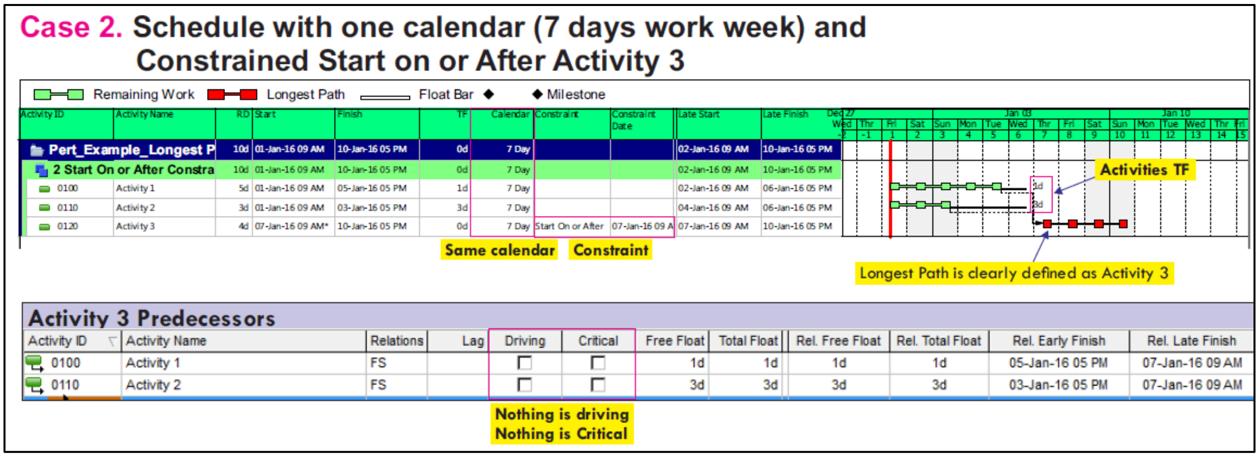


Figure 3 – Case2: The Effect of Constraints on the Longest Path

Finally, Case 3 illustrates the error that can result when only one calendar is added to our simple example. Adding a non-working calendar to "Activity 3" has the effect of identifying both "Activity A" and "B" as on the Longest Path. In fact, both activities are shown to be both "Critical" and "Driving" even though there is both Relationship Float and Total Float available. "Activity 2" is of course not on the critical path, and yet there it resides. How many schedules have been accepted into a construction program with this type of error? Here again, the scheduler has committed no mal practice, but the software has yielded erroneous results.

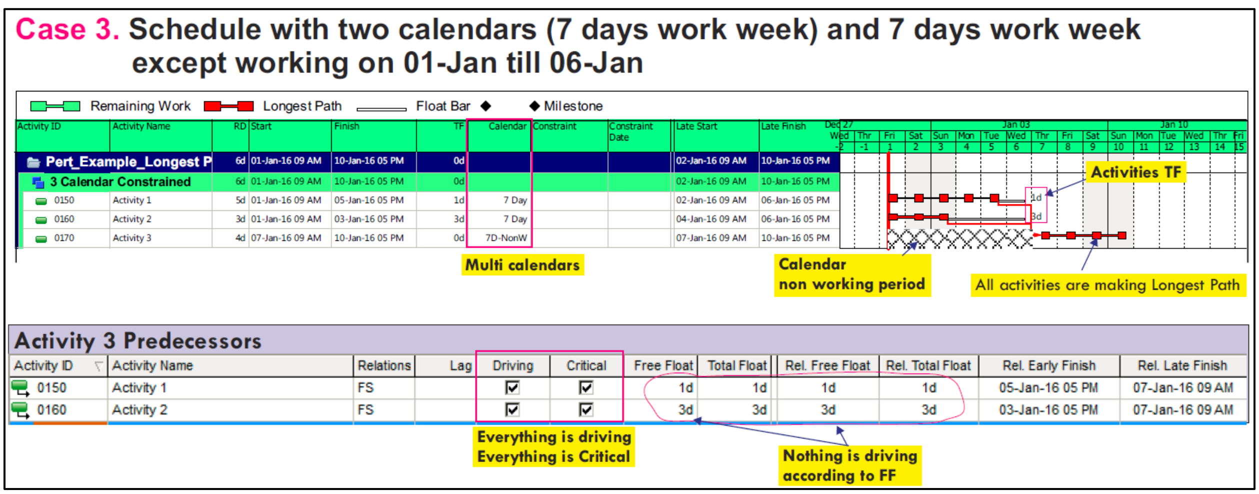


Figure 4 – Case3: The Effect of Multiple Calendars on the Longest Path

MS Project performs no better under the same conditions. As noted earlier, MS Project does not offer an equivalent of P6's Longest Path calculation to discern what is driving in cases of multiple calendars. It relies exclusively on a float (Total Slack) algorithm. As would be expected then, the multi calendar case creates a non-continuous critical path, as does the addition of a soft constraint.

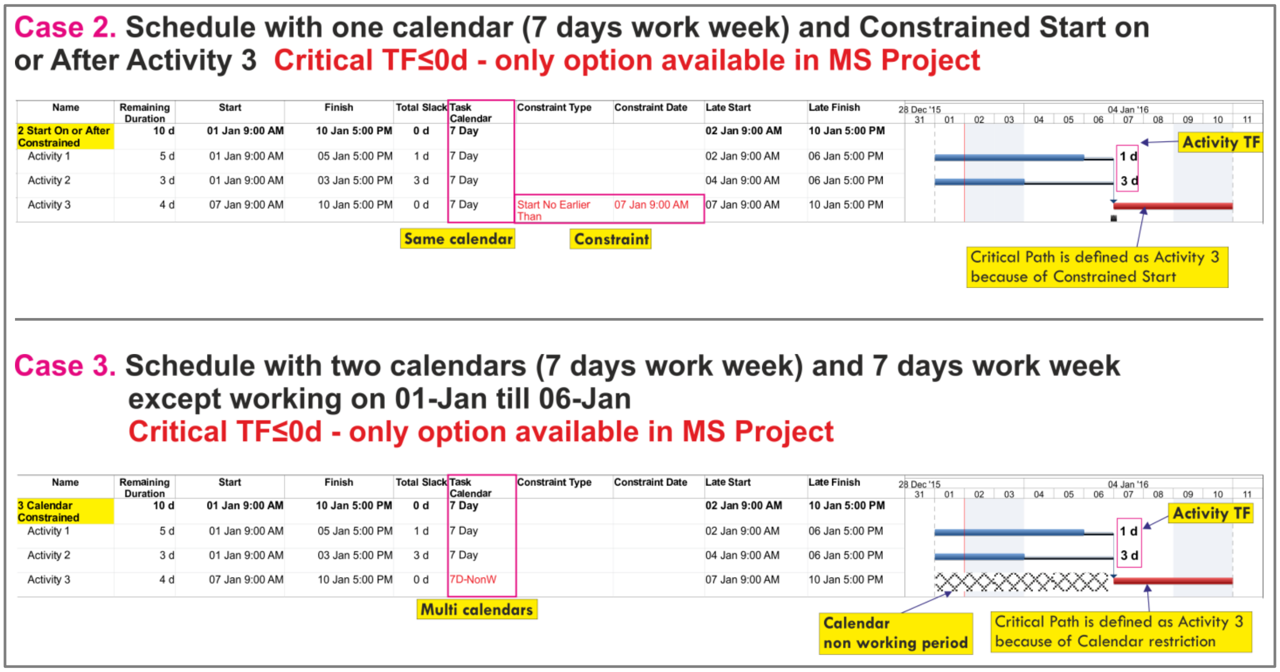


Figure 5 – Non-Continuous Critical Path in Microsoft Project Due to Calendars/Constraints

2. "Level of Effort" Distorts Longest Path

The schedule screenshots below reveals that, in P6, the addition of a Level of Effort activity can have the effect of erroneously adding a non-critical activity to the Longest Path.
In the schedule below, "Activity D" is not on the Longest Path, and this is correct since it is not driving the completion date. A Level-of-Effort activity has been added but not yet logically connected to the activities that are currently on the critical path.

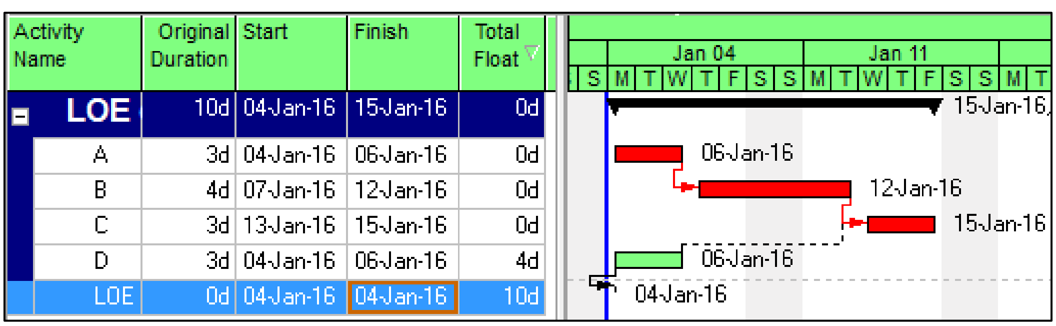
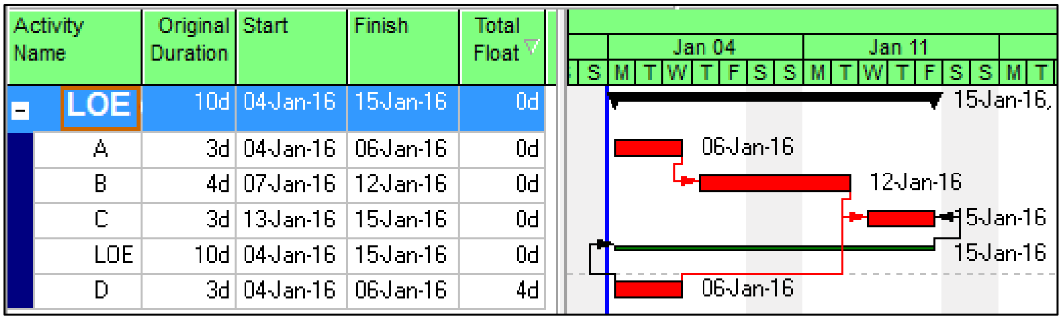


Figure 6 – Example Schedule with a Level of Effort Activity

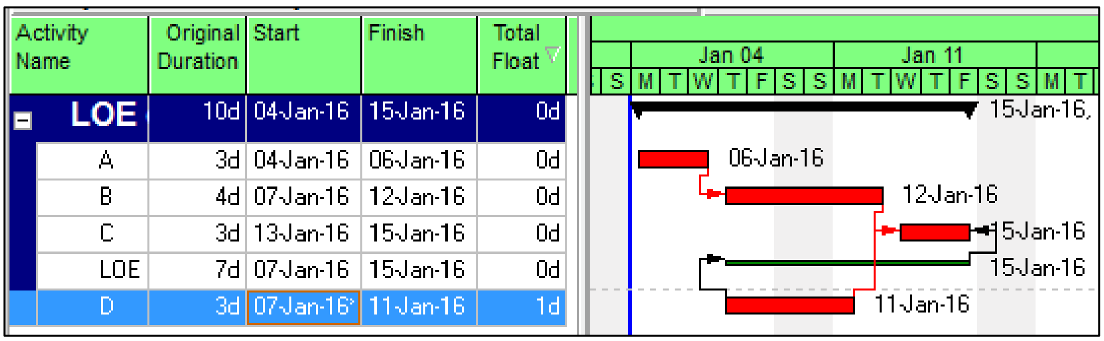
However, as shown below, creating an S-S relationship between the LOE activity and "Activity D", and an F-F relationship between the LOE activity and "Activity C" results in the addition of "Activity D" to the Longest Path.



Activity Name	Original Duration	Start	Finish	Total Float
LOE	10d	04-Jan-16	15-Jan-16	0d
A	3d	04-Jan-16	06-Jan-16	0d
B	4d	07-Jan-16	12-Jan-16	0d
C	3d	13-Jan-16	15-Jan-16	0d
LOE	10d	04-Jan-16	15-Jan-16	0d
D	3d	04-Jan-16	06-Jan-16	4d

Figure 7– The Effect of Level of Effort Activities on Longest Path

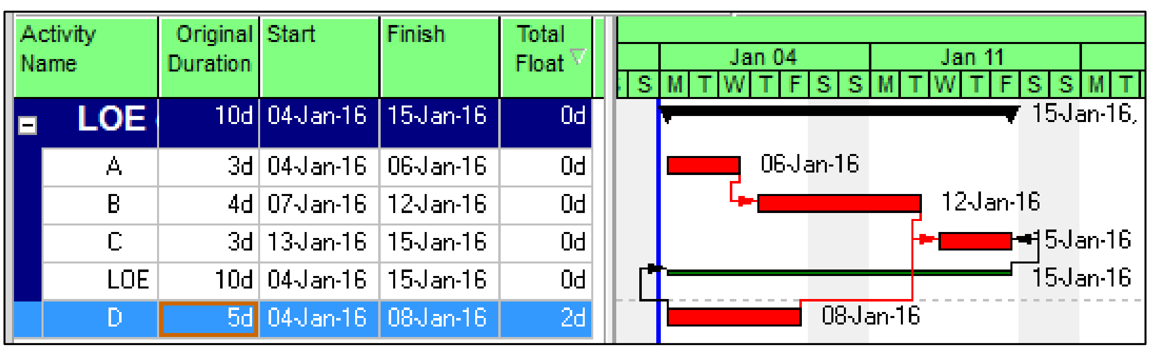
The addition of "Activity D" to the critical path is erroneous as the simple tests below illustrate. The screenshot below shows that delaying the start of "Activity D" by 2 days does not impact the completion date.



Activity Name	Original Duration	Start	Finish	Total Float
LOE	10d	04-Jan-16	15-Jan-16	0d
A	3d	04-Jan-16	06-Jan-16	0d
B	4d	07-Jan-16	12-Jan-16	0d
C	3d	13-Jan-16	15-Jan-16	0d
LOE	7d	07-Jan-16	15-Jan-16	0d
D	3d	07-Jan-16*	11-Jan-16	1d

Figure 8– Test1: Delaying the Start of an Activity on Longest Path

The screenshot below shows that increasing the duration of "Activity D" by 2 days does not impact the completion date.



Activity Name	Original Duration	Start	Finish	Total Float
LOE	10d	04-Jan-16	15-Jan-16	0d
A	3d	04-Jan-16	06-Jan-16	0d
B	4d	07-Jan-16	12-Jan-16	0d
C	3d	13-Jan-16	15-Jan-16	0d
LOE	10d	04-Jan-16	15-Jan-16	0d
D	5d	04-Jan-16	08-Jan-16	2d

Figure 9– Test2: Increasing the Duration of an Activity on Longest Path

3. Start-to-Start Creates In-Progress distortion

The example below provides an example of how even a properly performed progress update can result in errors in the critical path. Prior to the progress update, "Activity B", which is connected to "Activity C" in an S-S relationship, is properly identified as critical (despite the fact that only the start date of "Activity C" is critical and the finish date contains float). However, even after "Activity B" has started; it continues to be shown as critical, despite the fact that its duration could be increased by several days without any effect on completion. Total Float is calculated to be zero, even though the software option to compute total float based on "Finish Float" is selected.

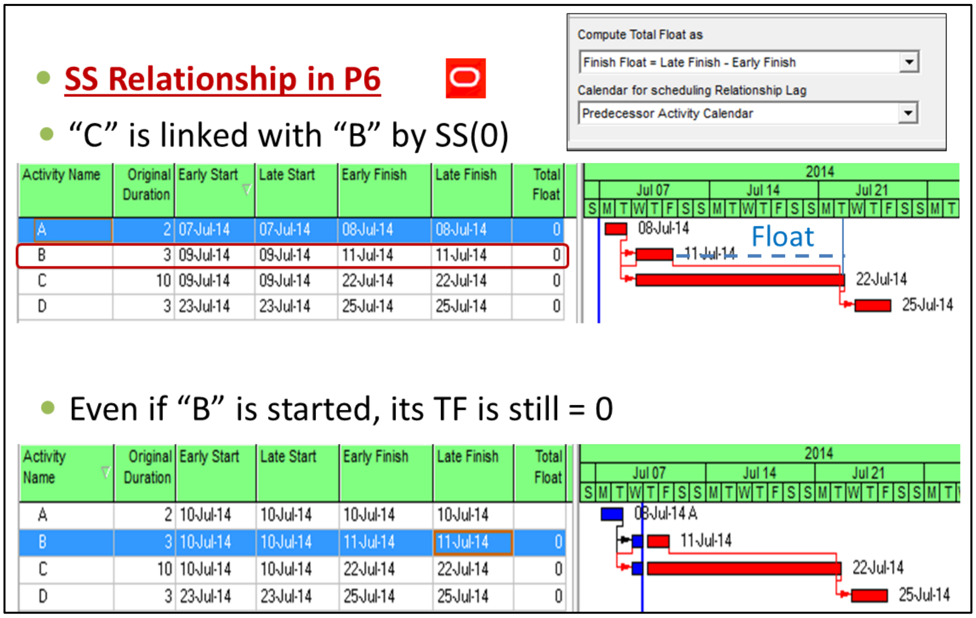


Figure 10 – Unreliability of P6's Longest Path When S-S Relationship is Used

4. Logical Relationship Limitations in MS Project

As illustrated in the example below, it is not possible in MS Project to have more than one logical relationship connecting two activities, and this is a serious limitation in terms of calculating reliable CPM float calculations enabling understanding of true schedule flexibility. In this case, "Activity 2" and "Activity 3" are connected by an F-F relationship, but there is no logical connection between the two activities to close in the start of "Activity 3". Since there is no constraint to the start of "Activity 3", its duration can be increased without any impact on completion or on Total Slack value. If the two activities should properly have logical start and finish relationship to reflect the actually intended sequence, this is not possible in MS Project (though it is possible in P6).

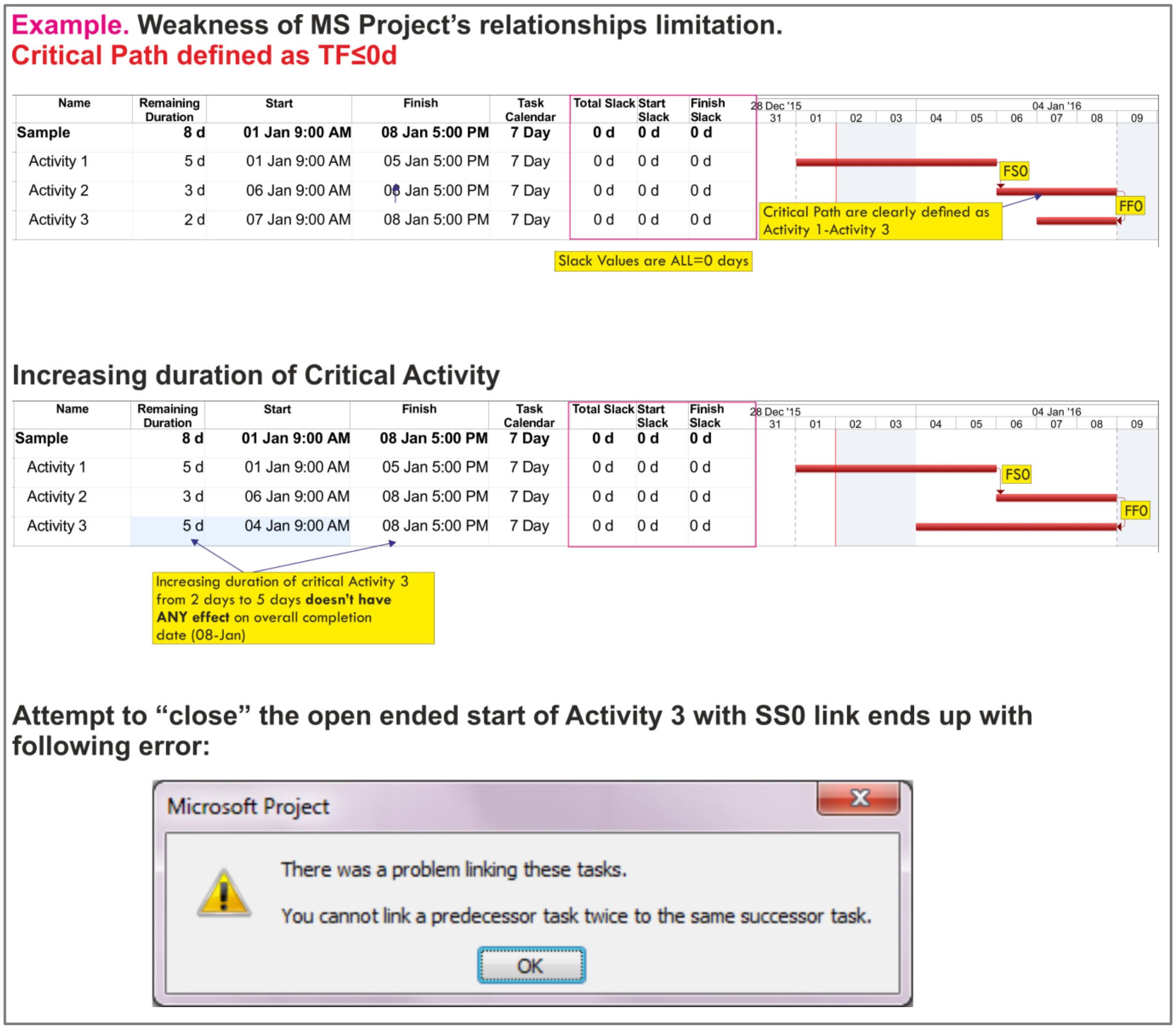


Figure 11 – Weakness of Microsoft Project Due to Relationships Limitation

To summarize the foregoing issues, it turns out that even when schedulers do not "abuse" the scheduling software, the software can behave badly. The scheduling software creates a dilemma for schedule reviewers, practitioners, and forensic analysts, in that in some cases, best practices have been followed, but the schedule does not correctly calculate the critical path.

CPM Compliance Evaluation

Notwithstanding the complications created by the above-described schedule software problems, a set of criteria which would constitute a minimum standard so as to comply with the definition of a CPM schedule is provided in this section. The figure below depicts the process of CPM compliance determination as one part of the Best Practices evaluation. The "CPM Compliance Test" is broken down into two parts: "Part A" is composed of a combination of logic and scheduling software checks, and; "Part B" which tests the critical path filter to verify it is actually continuous and determines the duration of the project as demanded by the CPM definition. Assuming the schedule passes the test, it moves on to the "Best Practices Review," which will be discussed in the next section. If the schedule fails due to software issues, as opposed to the incorrect use of the software or deficient practice, then it is accommodated and moves on to the Best Practice Review, where modifications are considered to avoid the software-caused problems.

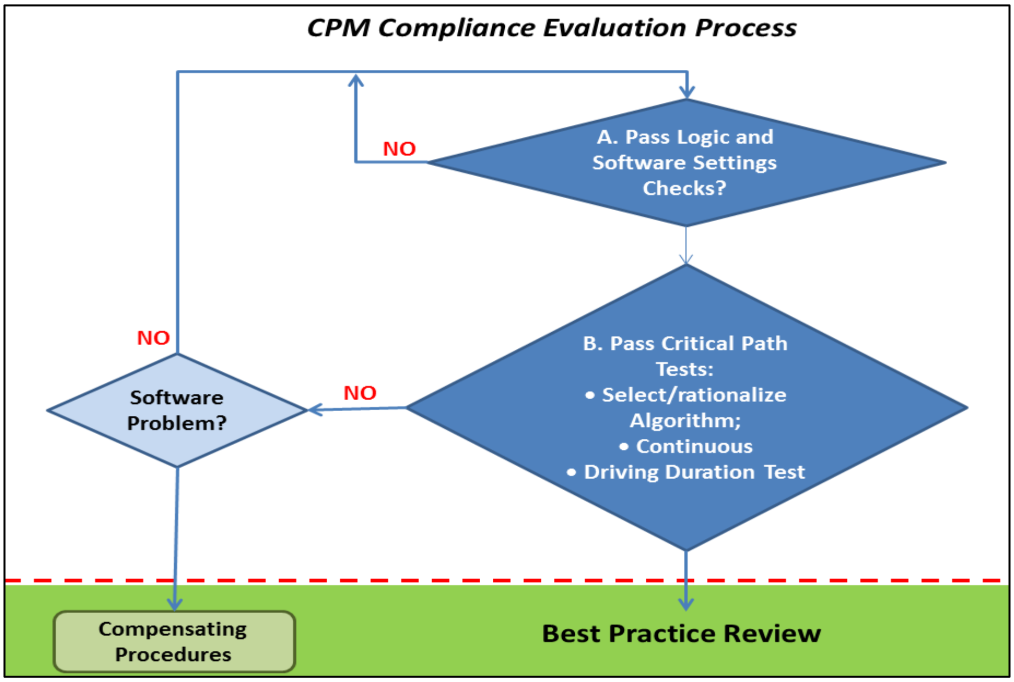


Figure 12 – CPM Compliance Evaluation Process

Each of the tests in the "CPM Compliance Test" table (below) follow from the definition of a CPM schedule provided earlier. Therefore, all of these tests must be satisfied in order for a schedule model created using CPM schedule software to meet the definition of a CPM schedule.

No.	Description:	Comments:
CPM COMPLIANCE TEST		
	A. SCHEDULE LOGIC & SOFTWARE SETTINGS:	
No.	**Description:**	**Comments:**
1	No “open-ends”: every activity has a predecessor and successor (except start and finish activities respectively)	Not permitted because float values would be unreliable.
2	No Activities without a Start relationship (activity has predecessor start not constrained)	Not permitted because float values would be unreliable.
3	No Activities without a Finish relationship (has successor but finish not constrained)	Not permitted because float values would be unreliable.
4	No “Mandatory” Constraints (completely over-ride network logic)	Schedule is not purely logic-driven.
5	No “Hard” Constraints (prevent activities from starting or finishing later than planned, but allows earlier start or finish)	Schedule is not purely logic-driven.
6	No “Soft” Constraints (allows delay to constrained activities)	Not permitted because of distortion to float value and critical path filter. See examples in “Software Dilemma” section.
7	Activity Logic restrictions: No F-S with +ve Lag.	Since network is a continuous path of *activities,* cannot have a condition where a relationship replaces an activity.
8	Logic must reflect intended sequence: For example, No S-F unless justified and appropriate	Finish-to-Start expected to predominate and to be encouraged because results are more reliable, but any relationship may be used if it reflects true constructability requirements.
9	No logic pairs except S-S and F-F.	Other pairs may create unintended consequences.
10	No activity has a F-F with a predecessor and a S-S with a successor. [15]	May cause unintended alteration to activity duration and distortions to critical path.
11	No actual Dates on or after the Data Date	Schedule not purely logic-driven.
12	No software Resource Leveling	Schedule not purely logic-driven.
13	No fractional activity durations	Can result in erroneous calculation of start and finish dates.
14	No manual alteration to the critical path filter generated by the schedule software.	Contractors sometimes *create* a “critical path” (through custom filters and other means) which bears no resemblance to the software algorithmic path.
	B. CRITICAL PATH FILTER TESTS	
1	If CP<>Longest Path, Contractor has identified discrepancy and provided rationale for which algorithm it is using.	CPM algorithms may yield different results depending upon use of calendars, constraints, etc.
2	Level of Effort Test: Delete all LOE activities to verify that the critical path is not impacted and also that no independent activity start or finish dates are altered by deletion of LOE activities.	Per earlier examples, software problems can result in erroneously populating activities on the Longest Path which are not actually driving.
3	Must have a continuous, logic-driven critical path.	From start to finish of the project. Schedules corrupted by use of constraints or improper logic sometimes have interruptions or no longest (driving path) at all. The only possible exception to this is an interruption created by a calendar.

Table 1 – CPM Compliance Test

It may be noticed that all software constraints (even "Soft" ones) are prohibited, whereas use of multiple calendars is not. The justification for the elimination of constraints is that there are practical alternatives to their use that accomplish the desired intention. For example: a "*Start on or After"* constraint might be eliminated by the introduction of a predecessor activity; or a "*Finish on or Before"* constraint applied to a completion milestone might be eliminated in favour of identification by scrutiny of the Finish Variance (versus Baseline) column (as well an activity description which identifies the contract date).

Multiple calendars, on the other hand, are not excluded because they are, as a practical matter, more difficult to eliminate. For example, on a project where the tunnelling operation is being performed 7-days-per-week, and the construction of a separate structure is on a 5-day-per-week schedule, the elimination of a calendar is not readily accommodated without losing the value of the schedule as a model representing the actual plan. Certainly the objective, given the vulnerability of software to critical path distortions caused by calendars, should be to exercise discretion in their use. In the case of a seasonal calendar which defers a block of activities until the spring, for example, it may be preferable to replace the additional calendar with an activity (e.g. - Weather-Impacted Work deferred until spring).

Apart from the prohibition on (positive) Lags which result in discontinuous activity logic (criteria no. '7'), there are no other mandatory limitations on the use of Leads and Lags. However, as will be discussed below, best practice standards may significantly limit their use.

In some important respects, the scheduler will meet different challenges depending upon the schedule software being used. For example, as noted earlier, MS Project does not in all cases offer an entirely satisfactory solution to the condition of an activity not having a start or finish *relationship* (Criteria no. '2' and '3' in the table), but in P6, an activity with an S-S relationship to a successor activity can be "closed-in", where appropriate, by adding an F-F relationship to the activities.

Applying 'CPM Compliance Test' to the DCMA 14-Point Metrics

The DCMA 14-Point metrics are intended to identify "*potential problem areas*" with the ("*contractor's IMS*") schedule [4]. Moreover, the metrics are intended "*to aid in understanding*" and are to be applied with discretion: "*a 'red' metric is not synonymous with failure*". It is fair to say then, that the DCMA does not propose the DCMA 14-Point Metrics as a Best Practice scheduling standard for the industry. However, it is clear that some quarters of the industry have never the less enthusiastically embraced the DCMA metrics as a schedule standard. Capturing this spirit, third party software vendors, asserting that the DCMA 14-Point "*methodology is used by numerous commercial and government organizations as a way to identify schedule deficiencies* …" [5], promise that with "*one click*" it is possible to verify that the schedule satisfies the *"DCMA guidelines"*.

Given the prominence of the DCMA 14-Point Metrics, it instructive to subject them to the "CPM Compliance Test" which, it has been emphasized, is a *minimum* standard intended only to determine if a schedule meets the definition of a CPM schedule. The table below evaluates the DCMA 14-Point Metrics relative to the CPM Compliance test.

DCMA Check No.:	**Metric Description:**	**CPM Compliance by Definition Test:**
4.1 Logic	• Allows up to 5% of activities to be without a predecessor or successor. • Investigate "dangling" activities with Start-to-Start or Finish-to-Finish, but not both.	• Non-compliant: the entire network must be closed in. • Non-compliant: entire network must be closed in. Note that MS Project does not allow multiple links (S-S and F-F) between the same tasks.
4.2 Leads	• States leads (negative lags) should not be used because "critical path and subsequent paths can be adversely affected …"	• Not excluded by definition. Permitted if justified (reflects constructability) and within limits. All Logic other than F-S must be scrutinized.
4.3 Lags	• Allows up to 5%. Concern with manipulation of float.	• Not excluded by definition (except certain cases). See note above (4.2). F-S with +ve lag should be rejected because critical path not continuous with activities.
4.4 Relationship Types	• At least 90% of relationships F-S.	• Not mandated by definition. Relationships must be justified as reflecting plan. Suggest that a *best practice* would be to encourage this metric.
4.5 Hard Constraints	• Number of tasks with "hard" constraints should not exceed 5%.	• Any constraint which over-rides logic is non-compliant by definition. No mention of other constraints which may adverse impact. • Since in MS Project, any constraint may over-rides logic, none would be permitted.[16]
4.6 High Float	• 5% or less have total float greater than 44 working days.	• Not a criteria demanded by definition. Logic evaluation might raise questions during best practice review. Evaluation of reasonableness of float requires constructability analysis.
4.7 Negative Float	• Ideally, there should be no negative float in a schedule.	• Since constraints are excluded, negative float does appear in a CPM compliant schedule. However, this should not be confused with the contract issue of whether a schedule can be shown to be in delay.
4.8 High Duration	• Not more than 5% of activities with baseline duration greater than 44 working	• Not a criteria demanded by definition, but best practice would

DCMA Check No.:	Metric Description:	CPM Compliance by Definition Test:
	days.	recommend not more than 20 w.d. or one reporting period.
4.9 Invalid Dates	• There should not be any activities with forecast start/finish prior to data date or actual start/finish date on or after data date.	• Non-compliant by definition.
4.10 Resources	• All tasks with durations of at least 1 day should have (dollar or hour) resources assigned.	• Not a criteria demanded by definition. Recommended that best practice should require full resource loading.
4.11 Missed Tasks	• Not more than 5% of tasks should have finished later than planned.	• Not a criteria demanded by definition. This is a performance question requiring performance analysis.
4.12 Critical Path Test	• Passes critical path test if the project completion date shows a negative total float number in direct proportion to the amount of intentional slip applied.	• Appears to rely on Total Float values, instead of driving path. Since the critical path by definition is the driving longest path, the effect on it is the primary criteria. In any case, other DCMA metrics permitting hard constraints, open ends, etc., would render the test unreliable under the 14-Point approach.
the 4.13 Critical Path Length Index	• Checks "critical path realism" through ratio of critical path length plus the project total float to critical path length.	• Not a criteria demanded by definition. Appears to be a performance metric, but rationale not clear.
4.14 Baseline Execution Index	• Ratio of completed tasks to should-have-been completed in period up to Data Date should be less than 95%.	• Not a criteria demanded by definition. Performance metric.

Table 2 – Evaluation of the DCMA 14-Point Metrics

As noted in the table, Items "4.1 Logic" and "4.5 Hard Constraints" of the DCMA 14-Point Metrics permit schedules to have up to 5% open ends and hard constraints. But for a schedule to meet the definition of a CPM schedule, it cannot have open ends or constraints that over-ride logic. Therefore, a schedule can pass the DCMA 14-Point Metrics without even meeting the definition of a CPM schedule, let alone best practices!

Best Practices Considerations

The schedule specifications in construction contracts usually require execution[17] schedules to be CPM schedules, and sometimes also require the schedules to meet a standard of "Best" or "Good" Industry Practice in CPM scheduling. It has already been pointed out there is not currently an *agreed* industry standard for CPM schedules that a reviewer might cite (though the specifications could certainly provide one, or choose an existing standard such as the GAO Guide).

A suggested approach to evaluating compliance with these requirements is illustrated in the flow chart below. To evaluate compliance with the requirement to provide a CPM schedule, the specification would, as suggested earlier, require that the schedule first satisfy the CPM Compliance Test (i.e. – Step 1).

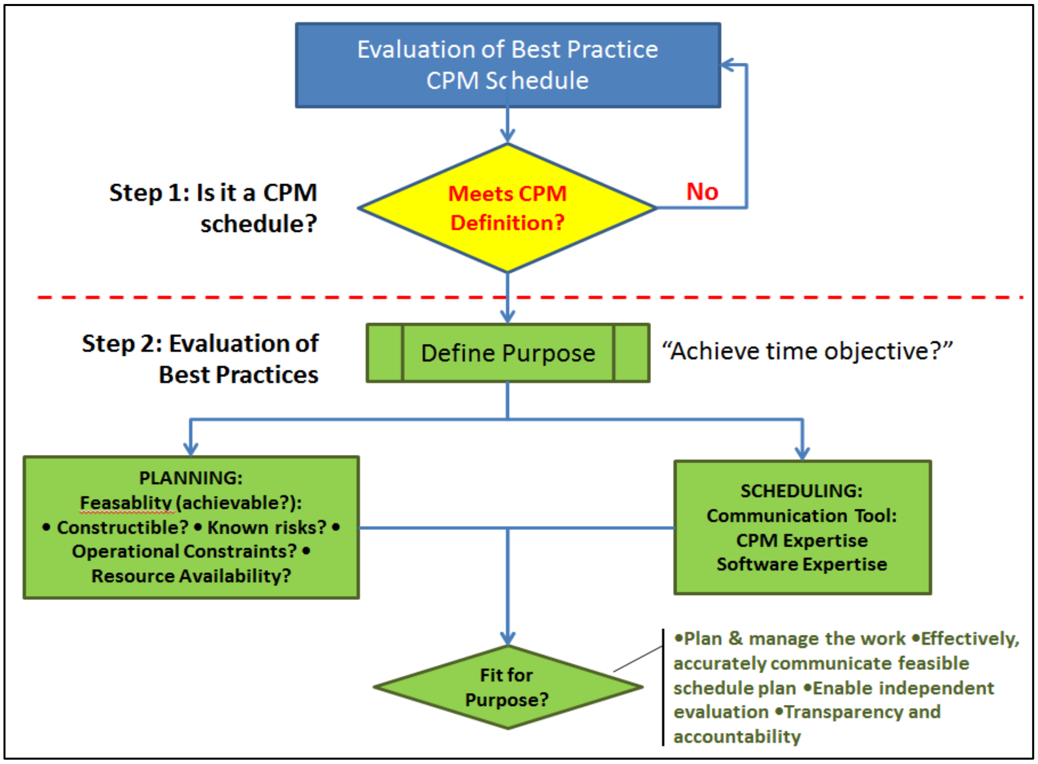


Figure 13–Evaluation of Best Practice CPM Schedule

Once the schedule qualifies as a CPM schedule by definition, it moves to review under Step 2 which is the determination of compliance with *Best Practices*. It is proposed that the Best Practices evaluation must begin with a statement describing the purpose of the schedule. That purpose, it is further suggested, should be to create schedules which tell the truth about time, because such schedules can then be the basis for what is sometimes called a Performance Based (or Earned Value Management) system[18]. Such schedules are meaningful if they communicate a feasible plan to perform the work. Confidence, on the part of all parties to the contract (not just the performing party) in the implemenation of the plan, requires suffiicent information to evaluate on-going performance and ideally, to enable independent monitoring. Once there is buy-in to the schedule forecasts, accountability is compelled by transparency and equally, a commitment on the part of all patries, espeically the owner, to the principle that peak performance is the project objective which all must serve.

Of course, there are legitmate (from the perspective of the contracting parties) other purposes of schedule which would dictate a different Best Practice standard. For example, an owner might only be interested in schedule insoffar as it verifies that a contractor has committed to the completion date. Some advocates of a performance-based system, the writer included, would counsel against this approach because it has been to a large extent the abiding industry model, expressly required or not, over the long period of unsatisfactory schedule and cost performance that the industry has experienced.

Whatever its orientation, a statement of purpose should provide clarity about the objective of schedule, which is essential to establishing minimum requirements to comply with the Best Practices and which are, by defintion, intended to meet the stated purpose. Space does not permit here a full discussion of schedule Best Practices considerations[19], but the general approach is described below.

Assuming the buyer of construction desires a performance-based project management approach, and intends for the schedule specifications to give this effect then, as shown in the flow chart, evaluating for best practices requires specialized review of both the “Planning” track and the “Scheduling” track.

The end result of the “Planning” review is to determine whether the schedule is a transparent, understandable, and independnetly verifiable, feasible plan, given constructability considerations, known contraints, availability of resources, and identified risks that are related to current scope. It is suggested that in order to evaluate feasibility, the schedule must be fully resource loaded so that productivity and production rate assumptions are understood.

The “Scheduling” evaluation is concerned with how the schedule has been modelled in the CPM software. An important Best Practice concern under this side of the review would be whether resources have been loaded correctly, and so as to best serve understanding, analysis, and monitoring. Once construtability is understood, it must be established that the activity logic reflects the same (likewise with leads and lags).

Conclusion

There is broad recognition in the construction industry that the quality of many CPM schedules is seriously deficient. A number of Best Practice guides have appeared in response to the problem, but none of them offer a minimum standard in order to determine if the schedule actually meets the definition of a CPM schedule. It should be considered axiomatic that before a CPM schedule is evaluated for best practices, it must meet the definition of a CPM schedule. A two-step process is recommended in which the CPM is tested to verify it meets the definition test and, if it passes, a Best Practices review follows. A “CPM Compliance Test” was presented in this article and applied to the DCMA 14-Point Metric evaluation. It was determined that a schedule can readily pass the DCMA evaluation without even being a critical path method schedule. An unfortunate technological challenge may arise, even where CPM best practices are followed, because scheduling software may be unreliable in identifying the critical path where multiple calendars and constraints are used. Notwithstanding the challenges, the creation of a best practice CPM schedule suitable for planning and performance modelling is entirely possible in practice, and is worth all of the effort because such schedules can play a major role in successful project outcomes as well as more expeditious and effective dispute resolution.

References

1. Korman, R., and Daniels, S. H., 2003, *Critics can't find the logic in many of today's CPM schedules users want software with flexibility, but is it true CPM?*, *Engineering News Record, New York*, 250(20), 30-33.
2. Kelly, Patrick M., PE PSP; Nagata, Mark F., PSP; Sanders, Mark C., PE CCP CFCC PSP; and Thorndike, Carmelita L., PSP, 2014. *The Great Debate: Is CPM Past Its Prime?*, Cost Engineering Journal 56, 1 (2014): 4-10, AACE International, Morgantown, WV
3. U.S. Government Accountability Office, 2015, *GAO Schedule Assessment Guide: Best Practices for Project Schedules*, GAO-16-89G December 2015, U.S. Government Accountability Office.
4. Earned Value Management System (EVMS) - Program Analysis Pamphlet (PAP), 2012, http://www.dcma.mil/policy/200-1/PAM-200-1.pdf, Defense Contract Management Agency.
5. DCMA Schedule Analyzer forProject™, 2013, http://www.forproject.com/docs/DCMA_Schedule_Analyzer_forProject.pdf, ForProject Technology Inc.
6. Choi, Jin Ouk and Gonzales, Anthony J., 2015, *Evaluation of Baseline Schedule Metrics for Successful Project Schedule Performance*, Cost Engineering Journal 57, 4 (2015): 17-26, AACE International, Morgantown, WV.
7. Rippe, Antonio; Chamberlain, Bruce; and Elliott, Darren, 2013, *GSDO Schedule Health Metrics,* NASA
8. Project Management Institute, 2013, *A guide to the project management body of knowledge (PMBOK guide)*, Fifth Edition, Newtown Square, Project Management Institute.
9. Schedule Analyzer for the Enterprise User's Manual, 2014, Ron Winter Consulting LLC.

End Notes

[1] The opinions expressed in this article are those of J. Gerard Boyle, who is the primary author of this article. Andrew Podolny and Dr. Wail Menesi created the schedule examples, and also provided expertise in CPM scheduling theory and scheduling software.
[2] The focus of this article is the construction industry.
[3] For example, the Economist concluded 30% waste, and disputes taking longer than ever before.
[4] A best practice is defined as a method or technique that has consistently shown results superior to those achieved with other means, and that is used as a benchmark.

[5] Note that a 1-day delay to an activity start or finish might results in more than a 1-day delay to the project as, for example, happens under certain circumstances where multiple calendars are used.
[6] "Near critical" is not precisely defined in schedule theory or industry literature, but is often taken to include up to 10 working days of float.
[7] There are certain rare exceptions to this which would be acceptable. For example, activities calendars compliant with contract terms, and necessitated by constructability considerations, might create "float" on the driving float path to completion such that the addition of days to the critical path would not add time to the completion date.
[8] The GAO Guide (p. 29) discourages use of S-F link because it is "counter intuitive and … over complicates … logic." In the rare instances where it is appropriate it should be used, because it reflects the actual sequence.
[9] The GAO Guide states: "*The start-to-finish (S–F) link has the bizarre effect of directing a successor activity not to finish until its predecessor activity starts, in effect reversing the expected flow of sequence logic. Its use is widely discouraged because it is counterintuitive and it overcomplicates schedule network logic. Examples of activity sequences used to justify the existence of an S–F relationship can usually be rewritten in simple F–S logic by either subdividing activities or finding more appropriate F–S predecessors within the network."*
[10] The end date of the activity will not change, but the start date will move to an earlier date.
[11] Some very limited alteration to the schedule-generated critical path filter (whether Longest Path or Critical Path in P6) may be excusable, and even preferable, because of the quirks of the various schedule programs, but must be clearly explained and justified in the interest of schedule integrity and transparency.
[12] Unless otherwise stated, all emphasis is added.
[13] Other commercially available CPM software programs have not been evaluated to determine if the same or similar problems exist.
[14] For example, the DCMA 14-Point Assessment states that soft constraints "allow the schedule to be logic driven" and so imposes limits only on what it defines as "Hard" constraints.
[15] Schedule Analyzer for the Enterprise Software identifies this and calls it a "Reverse Logic" condition.
[16] Over-ride feature in MS Project does not completely nullify constraint effect in all cases, and if it were used, why are the constraints included in the schedule in the first place.
[17] An "Execution" schedule is defined as the contractual, detailed construction schedule, reflecting the full scope of work and suitable for planning and monitoring.
[18] A Performance-Based Solution to Avoid Schedule Failures on Construction Projects, The Revay Report, Volume 31, September 2014.
[19] The GAO Schedule Assessment Guide is an excellent detailed reference.

Dr. Wail Menesi
Revay and Associates Limited
wmenesi@revay.com

John Gerard Boyle
Revay and Associates Limited
gboyle@revay.com

Andrii Podolnyi
Revay and Associates Limited
podolny@gmail.com