

# Time Impact Analysis or Windows Analysis? A Construction PM's Guide to Proving Delay

*By Damon Dewey*

Delay claims are not decided by who argues best. They are decided by what the project record can prove, and the record is built, or not built, long before anyone is arguing. I have spent thirty years building and reviewing construction schedules, and the forensic work I do now when projects end in disputes keeps teaching the same lesson: the analysis method matters, but the record decides whether any method can be used at all.

This article is for the project manager who is not a scheduler and does not want to become one, but who will someday have to answer the question "how many days did that delay cost, and whose fault was it?" with something better than an opinion.

## Two methods you will actually encounter

Forensic schedule analysis has several recognized methods; AACE International's Recommended Practice 29R-03 catalogs them. On contemporary projects with monthly schedule updates, two of them do nearly all the work.

Time impact analysis (TIA) is event-driven and forward-looking. The analyst takes the schedule update that was current when a delay event began, builds a small sub-network of activities representing the event (a "fragnet"), inserts it, and recalculates. The difference between the projected completion date before and after the insertion is the time impact of that event, measured at the moment it occurred. Many contracts require it by name for change orders and time extension requests.

Windows analysis is period-driven and backward-looking. The analyst divides the project into time windows, usually matching the monthly update periods, and for each window identifies the critical path as it actually stood, measures how much the completion date moved during that window, and assigns the movement to the events that caused it. It handles overlapping delays from several parties better than TIA, because it evaluates causation period by period instead of one event at a time.

## When each one applies

Use TIA when the delay is a discrete event with a definable start and end, when the contract calls for it, and when you have a credible schedule update from just before the event. It is the natural method for a differing site condition, a late owner decision, or a design change, and it works in real time: a TIA submitted with the change order request is far stronger than one reconstructed two years later.

Use windows analysis when delays overlapped, when more than one party contributed, or when the dispute covers a long stretch of the project rather than one event. Transit, hospital and large infrastructure programs almost always end up here, because by the time anyone is counting, a dozen things have gone wrong at once. Windows analysis is also what a neutral analyst tends to reach for, because it shows what happened rather than what a model predicts would have happened.

The two are not rivals. On a well-documented project, a TIA done at the time of each event and a windows analysis done at the end should tell the same story. When they do not, the record is usually the reason.

### **What each method needs from the record**

TIA needs one thing above all: the native schedule file for the update immediately preceding the event, with its logic, calendars and progress intact. It also needs the documents that fix the event's start and end: the RFI and its response, the change order, daily reports, correspondence. A PDF of a Gantt chart is not a schedule; it cannot be recalculated, so it cannot support a TIA.

Windows analysis needs every update from baseline to completion, as native files, plus the contemporaneous records for each window. A missing update does not just weaken the analysis; it creates a window nobody can see into, and the other side will fill that window with their own story.

Both methods need a baseline that was accepted, or at least used, and that reflected the plan the contractor actually executed. An unrealistic baseline is a problem for every method, because every method measures against it.

### **The mistakes that get claims thrown out**

The record was never kept. The single most common failure. Updates were made but not archived; the native files were overwritten each month; daily reports exist for some months and not others. No method survives this.

The schedule was updated for payment, not for truth. Progress was entered to support the pay application while logic, durations and sequence were left untouched. When the dispute arrives, the updates show a critical path nobody was managing to.

Notice was late or missing. Most contracts require written notice of a delay event within a set number of days. An analysis of an event that was never noticed in writing is an analysis of a claim that may already be barred.

The fragnet is fiction. A TIA fragnet that models the delay as the claimant wishes it had been, rather than as the daily reports show it was, fails the first time someone compares the two.

The windows are drawn to win. Window boundaries chosen to maximize one party's apparent delay, rather than matching the update periods, will not survive review by a competent opposing analyst.

Concurrency is ignored. If the owner's delay and the contractor's delay overlapped, the analysis has to say so and deal with it. Pretending the other side's delay did not exist is the fastest way to lose credibility with a tribunal.

### **What a PM should do during the project**

None of this requires a scheduler on staff. It requires a few habits, kept every month.

Archive every schedule update as a native file, named by data date, somewhere it cannot be overwritten. Keep the baseline the same way.

Insist that updates reflect reality: actual dates, remaining durations and any logic changes, with the changes listed in the update narrative so they can be explained later.

When a delay event begins, give written notice on time and open a file for the event: the date it started, what it affected, the documents that show it. Ask for a time impact analysis with the change order while the schedule update that preceded the event is still fresh.

Keep daily reports complete and consistent, especially crew counts, work locations and the reasons work did not proceed. They are the evidence that fixes dates when memories differ.

Record the owner's and the designer's response times to submittals, RFIs and change requests. Delay often lives in those gaps, and the log is the only place it shows.

Do these things and either method is available when you need it, and the first question any analyst asks, "what do you have?", gets an answer that ends arguments instead of starting them.

## **One-page decision checklist**

### **Which method?**

- Discrete event, definable start and end, contract requires it, credible pre-event update exists: time impact analysis.
- Overlapping delays, several parties, long disputed period, or an after-the-fact review: windows analysis.
- Few or no native updates survive: expect a weaker as-planned versus as-built comparison, and expect it to be challenged.

### **What the record must contain**

- Baseline schedule, native file, accepted or used.
- Every monthly update, native file, named by data date, archived where it cannot be overwritten.
- Update narratives listing logic, duration and calendar changes.
- Daily reports: crews, locations, reasons work stopped.
- Submittal, RFI and change-request logs with response dates.
- Written delay notices and the documents that fix each event's start and end.

### **Before signing an expert's engagement**

- What updates and records exist for the period in question?
- Does the contract specify a method?
- Was the delay one event or many, from one party or several?
- An analyst who picks the method before seeing the record is working backwards.

**About the author:** Damon Dewey is the founder of CPM Pros, a construction scheduling consultancy in the Seattle area. He has thirty years of construction scheduling and project controls experience in Primavera P6 on data center, hospital, airport and transit programs, and prepares forensic schedule analyses and expert reports when projects end in disputes.