Scheduling is a DRAG

Better project management does not mean making schedules longer but rather making projects shorter.

By William R. Duncan and Stephen A. Devaux

Most organizations have realized that project management is important. Most organizations even understand ***why*** it is important — since projects are how they implement strategy, good project management is necessary to survive; better project management is necessary to thrive and prosper. And this is true in both the public and private sectors.

One of the key indicators of the existence of "better project management" is schedule performance. A competent organization will be delivering the vast majority of their projects on or ahead of the project's committed date.

Yet few organizations come close to this goal. One of the key reasons is that their scheduling processes are totally inadequate — they are focused on minimizing the likelihood of finishing late rather than on maximizing the likelihood of finishing early. Better project management does *not* mean making your schedules longer. Better project management means making your projects shorter.

Time Is of the Essence

Why should you want your projects to take less time? We can think of a few reasons:

- If you are involved in any kind of internal project (information technology, marketing communications, process improvement, etc.), a shorter schedule will reduce the cost of each project while simultaneously increasing the number of projects that you can complete without additional resources.
- If you are involved in any kind of new product development (consumer goods, business-to-business, pharmaceuticals, electronics, automotive, etc.), a shorter schedule will get you to market sooner which can have a huge impact on profits. This is especially true in pharmaceuticals, where an extra year of patent protection can be worth billions (yes, billions) of dollars.
- If you are involved in any kind of competitive bidding (architecture, consulting, construction, engineering, environmental clean-up, etc.), a shorter schedule will improve your chances of winning the bid.

If we've omitted your application area from the above list, please accept our apologies. But if you are involved in the management of projects, we expect that you are willing to accept it as axiomatic that shorter durations are good, longer durations are bad.

The Critical Path Isn't "Critical"

One of the downsides of email is the amount of additional typing that we all must do. Perhaps as a result of this, many people seem to be adopting linguistic short cuts. Something like "LOL" is pretty hard to misunderstand once your kids have clued you in, but other short cuts can be confusing. For example, in a recent NewGrange list server discussion of "project management complexity," several posters kept short cutting to "complexity" which led others to think that the discussion was about technical complexity. In the same way that York is not the same as New York, a linguistic short cut can cause confusion.

The concept of the critical path suffers from a similar malady. One popular software package abandoned its policy of automatically highlighting the critical path because too many users were complaining "that stuff's not critical to me!" By isolating the word "critical" from the phrase "critical path," these users lost sight of the fact that some project activities are critical from a scheduling perspective even if they aren't critical from a technical perspective.

There are few absolutes in project management, but here are two: every project — *every project!* — has a critical path. Every project — *every project!* — that finished late finished late because one or more activities on the as-performed critical path started later or took longer than expected. It doesn't matter why the activity was late or ran long. If the project was late, it was late because of delays on the critical path.

Based on our mutual experience, the vast majority of corporate and government projects do not use Critical Path Analysis (CPA) when planning their projects. While most of the so-called project management software packages provide the needed functionality, many users turn the product into a glorified graphics package by using the mouse to draw bars across the screen without any real analysis of the dependencies or resource constraints that are responsible for determining the critical path.

Isn't Critical Path Analysis Old News?

Yes and no. It is certainly old. The grand-daddy of Critical Path Analysis, the Critical Path Method (CPM) dates back to 1957 which makes it a true *eminence gris* in the world of project management. But in the same way that bigger planes and more powerful engines haven't really changed the fundamental steps involved in traveling by air, the fundamental steps of competent project scheduling also remain unchanged.

Part of the reason that so many organizations no longer use CPA is that they tried it without really understanding it, misapplied it as a result, and then abandoned it because they weren't getting any benefit. We find it notable that CPA is alive and well and considered to be a requirement for any non-trivial project in most organizations that have a solid record of on-time project performance. Part of the problem is training: many courses on project

management claim to cover CPA, but few do anything more than show the participants how to calculate an activity-based critical path. Crashing is often confused with replanning, fast-tracking with concurrent engineering. The impact of resource constraints on the critical path is often ignored. This situation is somewhat analogous to claiming to be an experienced air-traveler on the basis of having ridden in a CRJ-100 from Atlanta to Savannah.

Competent Scheduling

Competent scheduling is an iterative process. Putting a bunch of information into MS Project isn't scheduling; it's data entry. Asking the software to display the critical path isn't scheduling; it's reporting. At a minimum, competent scheduling involves analyzing internal and external dependencies, resource requirements, risks and risk responses, and any other known constraints.

The first step would generally be to calculate a critical path schedule using a "clean" network — no resource constraints, no discretionary dependencies. While we know that this initial schedule is rarely if ever feasible, we still need it. Why? Because the difference between the cost/benefit of this unfettered schedule and the cost/benefit of a constrained schedule is the cost of those constraints. And without knowing that cost, the benefit of removing the constraint (by adding resources, avoiding extra work, or using more expensive subcontractors) can never be assessed.

Here's a simple example. Let's say that you are working on a project to implement a Customer Relationship Management (CRM) system. The system was justified on the basis that it would deliver an incremental $1,000,000 a month in revenue six months after installation. An ideal schedule of 12 months vs. a constrained schedule of 18 months says that those constraints will cost you $6,000,000 of revenue!

Perhaps it would be useful to invest a little more effort into optimizing your schedule. Not optimizing means that market windows are missed, commuters sit longer in traffic, and workers get outsourced in the absence of supporting systems. In some cases the costs are higher: soldiers suffer battlefield wounds because of late deliveries and children die when new drugs are released months or years late.

Hoist with its Own Petard

One fascinating aspect of CPM is that the design and language of the technique has contributed to its problems. *Total float*, one of the key outputs of critical path calculations is a prime example. Take a look at Figure 1, on the following page.

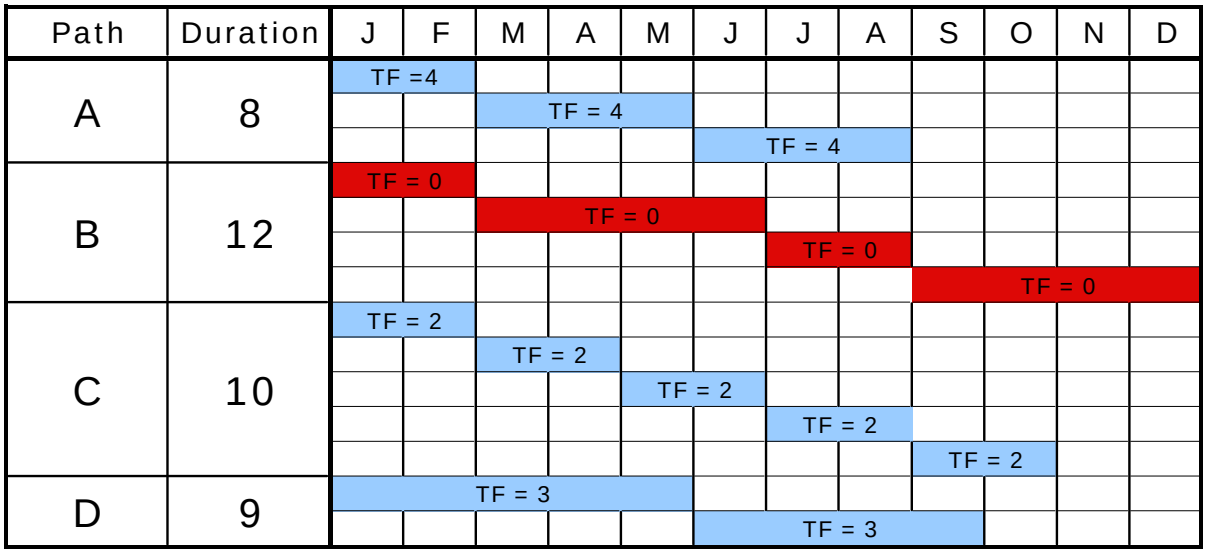

Figure 1: Project Schedule in Gantt Chart Format

This is a simple project with fourteen activities arranged along four independent paths. Path B, with a duration of twelve months, is the critical path. But path B has zero float. We all understand that zero is "nothing" and thus we are led to believe that there is "nothing" we can do about the critical path.

At the same time, we are lulled by the fact that every non-critical path activity has at least two months of total float. It appears that the first activity along Path C can slip by two months without causing any trouble. Yet total float is another linguistic short cut. It really means "total float along the longest path that this activity is a part of." If the first activity on Path C slips by just one month, the total float for every other activity along that path gets reduced from two months to one month. A one month slip causes us to lose four months of (total) float!

The language of CPA inadvertently causes many schedulers and project managers to focus on avoiding delays on non-critical path activities rather than trying to shorten the critical path so that the project can get finished sooner and start providing business value sooner as well.

Compressing the Schedule

One of us, Steve, has developed a metric to focus attention on the critical path. It's called DRAG and it stands for Devaux's Removed Activity Gauge. It is a simple concept: how much can we shorten this critical path activity before some other path becomes the critical path? Figure 2, below, shows the same schedule with DRAG highlighted instead of the rather misleading total float.

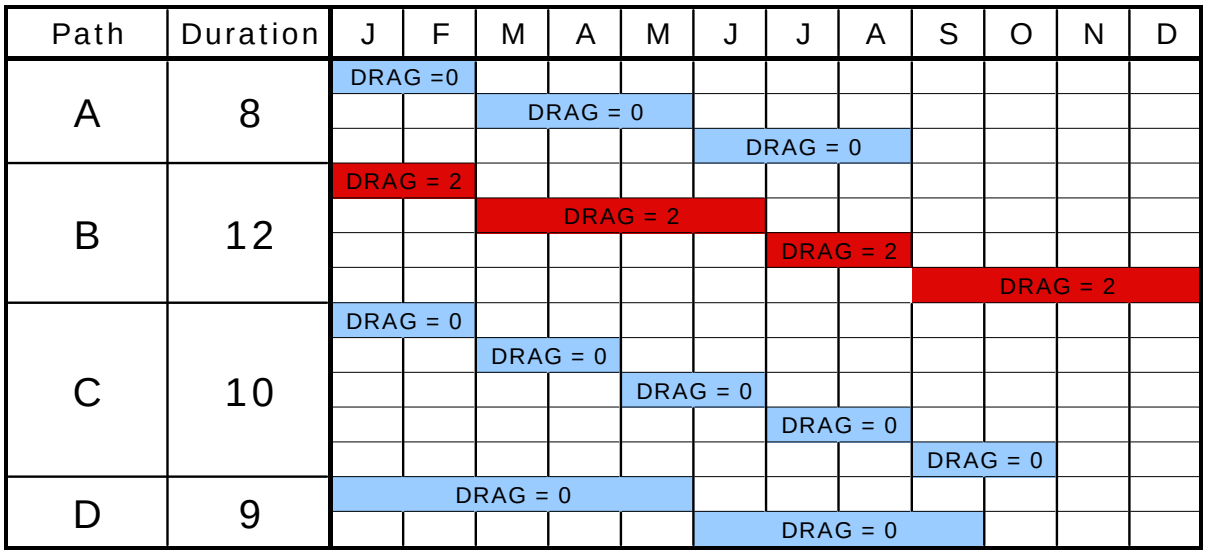

Figure 2: A Schedule Showing DRAG

Now we've got we want — everyone's attention is focused on the critical path. It's the non-critical path activities that have zero values, so those are the ones we'll ignore. Look at the second activity on Path B. Its DRAG is two, so that means that we can reduce its duration by two months before another path becomes the critical path. Reducing its duration by three months will only shorten the project by two months (its DRAG) because Path C then becomes the critical path.

If this project were the CRM project mentioned above with a monthly cost of delay of $1,000,000, it is now easy to see that we should be willing to spend a *lot* of money to shorten the duration of Path B.

Keep in mind that we're dealing with a *very* simple project — computing DRAG on a real project with complex dependencies and intersecting paths is much more difficult. We've also made a crucial assumption — that you are skilled enough in techniques such as crashing, fast-tracking, and resource management so that you can shorten activity durations without creating excessive risk.

Some project management software vendors have begun to include DRAG calculations in their products. But even without software support, you can get a rough estimate of DRAG by reducing the duration estimate for any critical path activity to see how much can be cut before another path becomes critical.

Summary

Shorter schedules are good. And shorter schedules come from focusing attention on the critical path.

As a project manager, remember that any schedule you produce is based on an *estimated* critical path. As the project progresses, the critical path will also change in response to scope modifications, resource constraints, and a host of other factors. Some project managers will use these inevitable changes as an excuse for not analyzing the critical path, but remember our absolute: every project has a critical path. You can work to understand it, or you can ignore it. But don't delude yourself into thinking that it isn't there.

######

William R. Duncan is a principal of Project Management Partners, a project management consulting and training firm headquartered in Lexington, MA USA. He was the primary author of the original version of *A Guide to the Project Management Body of Knowledge*, and his "process model" of project management was used to organize ISO 10006, *Guidelines for Quality in Project Management*. He currently heads the certification program for the American Society for the Advancement of Project Management. He speaks widely throughout the world on topics such as Organizational Performance Improvement, Project Dynamics: the Law of Unintended Consequences, and Project Portfolio Management.

Stephen A. Devaux is an Adjunct Professor of Project Management at Brandeis University and Adjunct Lecturer in Project Management at the University of the West Indies at Cave Hill, Barbados. He is also President of Analytic Project Management of Bedford, MA. He is the creator of the Total Project Control (TPC) methodology, and author of Total Project Control: A Manager's Guide to Integrated Project Planning, Measuring and Tracking, John Wiley & Sons, May 1999. A former project manager of corporate training, Steve has spent 18 years training and consulting with Fortune 500 clients in industries ranging from aerospace and pharmaceuticals to software development and telecom. A frequent presenter at project management conferences, he has published numerous articles on both instructional design and project management implementation.