

Gaining Competitive Advantage through Reducing Project Lead Times
Duncan Patrick, CMS Montera Inc.
Jack Warchalowski, CMS Montera Inc.

Introduction

We all know that any successfully completed project has to achieve three simultaneous objectives for scope (or content), schedule, and cost. While this is well known, it is hard to achieve in reality. Often, companies find themselves forced to sacrifice one (or even two) of the objectives to achieve any one of them. Many of us have heard about situations where companies go over budget in order to deliver a project on time or even worse, ship a project to a customer (to achieve schedule) and finish it in the field. There are many, many stories and anecdotes that people share during conferences and training sessions about projects that have gone ‘wrong.’

While all three objectives of projects (scope, schedule and cost) are important, this paper will narrow its focus to primarily schedule. It is not because scope and cost are less important. They are very important. The narrowed focus is because we believe that the proper focus on schedule can have a very dramatic impact on cost control a scope achievement as well. In fact, we believe that if companies can better manage schedule, they will be able to dramatically lower their overall project costs and deliver better (higher quality) content.

Consider a company that designs, builds, and installs automation equipment. Like many environments the market is highly price competitive and lead time sensitive. Quality is an assumed given. Most projects are awarded to the supplier with the most reliable design and lowest price. What is the best way for a company like this to gain a competitive advantage?

The Sooner we Complete the Projects, the Sooner we Realize the Benefits, the More Capacity we Expose

Companies undertake projects to achieve a benefit: develop software to sell to the market; build an apartment building to rent to tenants; implement software to improve productivity; develop a new piece of automation equipment to deliver to a customer. Assuming that these companies have done a fair assessment of the business case for the project (i.e., the benefits will outweigh the costs); it is fair to say that the sooner the company completes the project, the sooner the company will achieve the intended benefits. In other words, the sooner the project is complete, the sooner the benefits are realized (Kendall & Austin, 2013).

If companies achieve the project benefits sooner by completing the project sooner, shouldn't these companies try to reduce the overall project lead time as much as possible? Doesn't this make achievement of the schedule the most important objective? Or even more, shouldn't companies try to deliver more projects in less time in order to achieve the benefits from more projects sooner? In other words, shouldn't companies try to constantly reduce project lead times?

An important characteristic of project environments is that shortening project lead times allows companies to deliver more projects with the same resources. For example, if it normally takes a company 9 months to develop a new product and bring it to market and they change the way they execute projects in order to do the same in 6 months, what are the benefits? Besides the fact that they are receiving revenue from the sales of the new product 3 months sooner, the product development resources have 3 months (months 7, 8, and 9) available to start work on the next project. In other words, completing a 9-month project in 6 months exposes 50% more project capacity.

So the answer is YES, companies should constantly try to reduce project lead times as long as the following is true:

1. Companies can find a way to reduce project lead times without significantly increasing project costs and/or sacrificing project content; and
2. There are clear benefits to completing projects sooner.

Does every organization realize benefits from completing projects sooner? We claim yes. We claim that every organization can achieve a competitive advantage by completing projects sooner. Sometimes this competitive advantage is obvious to see: getting products to market faster, quoting shorter lead times than competitors on projects, and realizing internal benefits sooner. Sometimes this competitive advantage is harder to see: delivering

more projects with the same resources lowers the overall labor cost per project. Lower labor costs lead to either higher net profit or greater price competitiveness. Either way – a competitive advantage.

So back to the automation company, not only are shorter lead times more attractive to their customers (sometimes even allowing a price premium), but shorter lead times also allow the automation company to complete more projects with the same resources (lower costs per project), accelerate revenue collection, and provide the option to lower prices to win more business.

The unavoidable conclusion is that reducing project lead times always provides companies with a competitive advantage. This competitive advantage comes from potentially two sources:

1. The project benefits are realized sooner; and
2. Additional project capacity is made available to complete even more projects with the same resources.

So the question now turns to: How do we significantly reduce project lead time without increasing cost or sacrificing project content? However, we prefer to make the question even more challenging. ***How do we significantly reduce project lead time while decreasing cost and improving project quality?***

The Root Causes as to Why Projects are often Late, Over Budget, and/or Under Scope

Project environments are characterized by high degrees of uncertainty. Uncertainty in content (for many projects the work has never been done before and the customer may not have fully determined what it is they want). Uncertainty in vendor performance and uncertainty in internal skill sets. Because of this uncertainty, the common complaints related to successfully managing projects include:

- Original due dates are not met
- Too many changes to scope or timing
- Resources (internal and external) are not available when needed
- Necessary things are not available on time – material, information, specifications, approvals, etc.
- Disagreements about priorities among projects
- Budget overruns
- Too much rework

In our experience, when we show the above list of complaints to companies from a variety of industries, the common comments are: “we have them all” or “that’s just the way project environments are,” or “yep, always been that way.”

Fortunately for us all, Eliyahu Goldratt was not satisfied with the above explanations and so he published his ground breaking book, *Critical Chain*, in 1997. In this book, Goldratt identifies the primary root causes underlying all of the above common complaints and presents his solution to dramatically improving project performance — Critical Chain Project Management (Goldratt, 1997). Central to Goldratt’s examination of the problem and his proposed solution is that it is not the uncertainty in projects that must be reduced (as projects are by definition uncertain) but it is the way we manage the uncertainty that must change.

So what is the root cause as to why projects are often late, over budget, and/or under scope?

For a single project environment, the root cause is that companies don’t ensure that task gains offset task delays. In other words, companies do not structure and manage their projects to take advantage of gains in task performance (tasks that are completed in less time than planned) in order to ensure they cancel out the many delays in task performance (tasks that take longer than planned). In fact, most companies are so unaware of this that they actually waste task gains.

For a multi-project environment, the root cause (in addition to the root cause related to single projects) is that resources are forced to multitask among tasks and among projects so much that productivity is lost, quality suffers, delays go unresolved, queues of work build up, and lead times extend greatly.

We will devote the remainder of this paper to an explanation of Goldratt's description of these root causes and his Critical Chain approach and then connect this solution to the competitive advantages gained by reducing project lead times.

Root Cause #1 – Task Gains Don't Offset Task Delays (Single Projects)

Most companies structure and manage projects according to the following generic project planning approach: break the project down into a series of logical tasks, identify the important dependencies (which tasks must be complete before the next can start, etc.), determine task durations and assign expected start and end dates to each, assign resources to tasks, insert key milestones that must be met, and then execute the plan to achieve the task and milestone dates in order to complete the project on time/budget.

As previously stated, projects are uncertain. This means that task durations are estimates (some better than others depending on the level of uncertainty). Because task durations are based on estimates, the actual time to complete them may either be more than the estimate (a task delay) or less (a task gain). Many tasks take longer than expected due to a variety of reasons including: waiting on vendors, materials, information, approvals, specifications, resources, etc. Companies do an increasingly better job managing these and other issues that cause delays. But the reality is that due to the uncertainty inherent in projects, these delays will always occur.

Likewise, many tasks take shorter than expected. Sometimes things go better than planned and good project management techniques are helping companies do a better job at managing tasks.

In order for a project to be done on time, the task delays must be offset by the task gains. It is simple math. So if projects are often not complete on time (or equally require extra budget to be done on time) something must be happening to the task gains. Goldratt (Goldratt, 1997) determined that Task Gains are wasted because of companies' inability to properly manage the following:

- Student Syndrome – people put off work until later (they usually don't start right away)
- Parkinson's Law – work expands to fill the time available
- Integration Points – if two tasks must come together before a third can start, both must be early in order to take advantage of the gains (but only one must be late to pass on the delay)
- Resources with multiple tasks – if Resource A completes his or her task early and passes it on to Resource B, the gain is passed on only if Resource B can start immediately (but usually Resource B is busy working on other tasks)

Because of the above four factors, most task gains are wasted and therefore task delays accumulate causing projects to be late and/or over budget.

Root Cause #2 – Bad Multitasking – (Multiprojects)

In many project environments, the same resource works on more than one project at a time. This often results in resources being forced to multitask among different tasks and among different projects. In this case, by multitasking we mean starting a new task before an existing task is complete. This happens for a variety of reasons. One such reason being that a resource (Resource A) is working on a task and then requires the input from Resource B. Resource B is busy working on something else, so Resource A must wait for Resource B to respond. In the waiting period Resource A naturally works on a different task (could be the same or a different project) until resource B responds. The net result is that the initial task that Resource A was working on is delayed. Even worse, when Resource B finally responds and provides his or her input back to Resource A, Resource A may need to re-review what he or she was doing and why in order to get re-oriented to the task (all dependent upon the amount of time that has passed).

The way we often explain this concept to different executive teams is to ask them to perform a short exercise. We ask them to complete two tasks. Task 1 is to write out the word MULTITASKING. Task 2 is to write down the numbers 1 through 12. When they are complete both tasks, we ask them to record how long it took to complete both. Then we change the rules a little to get them to do both tasks at the same time. In other words, write the letter M, then write the number 1, then write U, then write 2, and so on until they are complete. Again we ask them to record how long it took to complete both tasks. Typically, the second scenario takes on average 40% to 50% longer to complete. (Scenario 1 – both done in approximately 12 seconds, Scenario 2 – both done in approximately 18 seconds).

Similarly, in his book, *The Billion Dollar Solution* (Newbold, 2008), Robert Newbold describes the Confetti Game where he asks executives to tear two different color pieces of paper into 30 smaller pieces – one color followed by the second color. After recording their time they perform the task a second time, only now they must alternate between the two colors every three moves. Again the time is recorded. In our experience running this exercise with executives, the second scenario often takes groups more than 50% longer than the first. (Scenario 1 – both done in approximately 90 seconds, Scenario 2 – both done in approximately 135 seconds). One additional measure taken in the Confetti Game is the counting of the number of torn pieces. Inevitably, the number of players ending up with 30 pieces of each color in Scenario 2 is usually 50% less than in Scenario 1.

In both exercises, the participants quickly realize that the poorer performance in Scenario 2 is due to their own multitasking. What they often don't realize is the implications behind the results. The implications are as follows:

1. A 50% longer lead time to complete two projects when multitasking means that it is possible to complete three projects of equal size in the same time when not multitasking. A 50% capacity increase with the same resources.
2. Quality always suffers when multitasking is present versus when it is not. When quality suffers, rework often occurs, which not only adds to cost, it also extends the lead time and reduces capacity. In practice, software developers and design engineers often complain how interruptions can greatly slow down their progress (sometimes to the point of requiring them to start over) as well as not ending up with the optimal code or design.
3. When multitasking, the first project completes very close to the completion time of the second project. When not multitasking, the first project completes often 50% sooner.

From a practical perspective, multitasking often creates the following problems in organizations (Kendall & Austin, 2013):

- Management and support group attention is harder to get – causing task to be delayed
- Problems take longer to discover – wasting time until intervention occurs
- Resource managers are very busy and do not take the time to clarify tasks before assigning them to people or making sure tasks are actually ready to begin

In summary, multitasking in project environments often leads to longer project lead times, more rework, lower organizational capacity and as a result, increases project cost and reduces competitiveness.

The Solution is to Reduce Multitasking and Ensure Task Gains Offset Task Delays

So far we have made the case for reducing project lead times. Shorter project lead times create a competitive advantage for most organizations because it allows them to realize the project benefits sooner as well as deliver more projects with the same resources. In addition, we explained the primary root causes for why projects are late and/or over budget — task delays don't offset task gains and resources are engaged in too much multitasking. To be clear, if companies can solve these two root causes, they will also succeed in reducing their project lead times.

Goldratt's Critical Chain Approach (Goldratt, 1997) provides a way to solve both of these root causes. We summarize this solution here with the following three steps:

1. **Stagger** the release of projects into execution in order to limit the number of active projects in the pipeline
2. **Buffer** project plans in order to better protect against uncertainty
3. **Prioritize** task execution based on project completion versus buffer consumption

Stagger

Resources usually multitask for two reasons:

1. They don't have what they need to finish their current task, and
2. They have 'other available tasks' to work on.

So, they stop working on one task (send off a request for whatever is missing) and pick up another task – and so on. The more 'other available tasks' they have, the larger the queue of started (but not finished) tasks. In other words, multitasking creates multitasking. We call it a vicious cycle. The best way to break this vicious cycle is to limit the number of active projects in execution in accordance with the organization capacity to quickly flow projects to completion (Kendall & Austin, 2013).

Rather than trying to limit the number of projects at each phase of a project or limit the number of projects to each resource group, the organization should pick a Staggering Point around which to make this decision. The Staggering Point is either the key Project Phase that often limits the number of projects an organization can quickly complete or a Resource Group that is often in very high demand. It differs by organization.

A good way to determine the location of the Staggering Point is to answer the question “what are tasks waiting on the most?” (Kendall & Austin, 2013):

- If the answer is one type of resource group that is usually specified in project plans, then the organization should stagger the release of projects around this resource group in accordance with their capacity.
- If the answer is varying resource groups that are often not specified in project plans (i.e., management and support resources), then the organization should stagger the release of projects around a Project Phase (typically some kind of Integration Phase).

In parallel with implementing the Stagger step, organizations also need to implement a Full Kitting process, where a detailed process of what is required to start a specific task, phase, or chain of a project is established and adhered to. The basic rule of Full Kitting is to not start on work until adequate preparation is complete. (Adequate preparation means the task, phase, or chain can be fully completed once started.)

Once successfully implemented, Staggering and Full Kitting greatly reduce the bad Multitasking that occurs in most project companies.

Buffer

In most project plans, tasks have both expected start and end dates (in accordance with their estimated durations). These durations are based on either historical actuals or forecasted projections and have embedded in them a certain amount of waste and safety. The waste comes from the organizations' past experiences full of too much multitasking and task delays. The safety comes from people's admirable desire to be good and reliable people. People are not going to provide task estimates that have only a 50% chance of success. Instead they will choose an estimate with a much higher likelihood of success in order to be seen as a reliable professional.

To ensure that companies have a better chance of taking advantage of task gains, project plans should be developed in a way that removes the task safety from each task and accumulates it at the end of the project so that all tasks can have a chance of using it. This way all of the safety is used to protect the entire project, not each individual task. The accumulation of the task safety at the end of the project is called a Project Buffer. This is accomplished by asking resources for aggressive durations for each task, assuming that “everything goes right” during project execution. This process takes a while to perfect and requires a degree of trust between management and the resources.

Then the notion of identifying the Critical Path should be extended to include not only the longest chain of task dependencies, but also resource dependencies. Resource contention should be resolved at this time. This newly identified longest chain of both task and resource dependencies is called the Critical Chain. Once the Critical Chain has been identified, the duration of the Project Buffer can be determined, which is usually half the duration of the Critical Chain.

It is important to communicate and reinforce that the Project Buffer is meant to be consumed during the execution of the project. In other words, the project lead time is expected to be the sum of the Critical Chain duration plus the Project Buffer duration.

Finally, chains of tasks that enter into the Critical Chain also need to have their own buffers (Feeding Buffers) inserted at each integration point to ensure that these chains start early enough to not pass on unnecessary delays to the Critical Chain.

Prioritize

Since the duration of the Critical Chain plus the duration of the Project Buffer determines the project lead time, the way to determine the progress of the project is to determine how much of the Critical Chain is complete and compare it to how much of the Project Buffer (PB) has been consumed. Furthermore, since the duration of the Project Buffer typically is established to be half of the Critical Chain, it is expected that 0.5 day of PB be consumed for each 1 day of CC completion. This ratio, known as the Flow Index divides the CC complete by the PB consumed. The target is 1. Any value greater than 1 means the project is completing CC faster than it is consuming PB (a good thing). Any value less than 1 means the project is completing CC slower than it is consuming PB (a warning of potential trouble).

Also, the same ratio of Flow Index is used to determine the number of Chain Delay Days a given project is currently experiencing. For example, Project A has an aggressive CC duration of 100 days and a PB of 50 days (total project lead time is expected to be 150 days). If 50 days of the CC are complete and 30 days of the PB are consumed (i.e., 80 days have elapsed since the start of the project), then the Flow Index is .833 (.5/.6) and the Chain Delay Days are 5 days late (as the project should have only consumed 25 days of the PB). Expressed as a formula, the Chain Delay Days = (Flow Index Target – Current Flow Index) x PB Consumed (in this example Chain Delay Days = (1 - .833) x 30 = 5 days)

Therefore, in order to drive project and task priorities during execution, the Flow Index and Chain Delay Days are used at both the Project and Task levels to determine overall priorities across projects and within a project. At the Project Level the project with the lowest Flow Index has the highest execution priority requiring the most focus. The same can be said at the task level based on the task with the highest Chain Delay Days. By calculating these measures for every active task, the relative priority of every task across all projects is known and used to drive task execution decisions and resource allocation decisions.

In order to capitalize on the real advantages offered by the Buffer and Prioritize steps, the organization needs to make a change in the way it manages and schedules resources. The movement towards building project plans using aggressive durations, resolving resource contentions at the time of designating the Critical Chain, and prioritizing using Flow Index and Chain Delay Days requires the organization to schedule tasks not resources. Scheduling tasks means assigning the optimal number of resources to tasks as to get the tasks done as fast as possible. Even if this is not the most efficient way to utilize the resources. The objective is to keep the tasks busy and moving, not to keep the resources busy.

Similarly, projects should be managed more like a relay race than a train schedule. In a relay race, the next runner starts as soon as the preceding runner is complete. Likewise, under the Critical Chain approach, the next task starts as soon as the previous task is done (the next resource must be ready even if it means the next resource might occasionally stand idle waiting for the preceding task to be completed). You don't want the next runner in a relay race to miss the baton hand off because he was busy getting ready for another race. Under the train schedule approach, if a task is finished early, the next task often does not start until its prescribed scheduled start date because the next resource is not ready to start until then.

Once successfully implemented, the Buffer and Prioritize steps allow for many more task gains to offset task delays.

The Solution Leads to Results

Based on our implementation experience of working with many different clients, we have seen companies realize surprisingly great results in a short period of time. Many of the companies we have worked with successfully reduced their lead times and lowered their project costs between 25% and 50%. The automation company that we introduced at the start of this paper reduced their average project lead time by greater than 25% and improved their project profit margin by 24% on the first three projects they managed using the Critical Chain approach.

In Kendall and Austin's book, *Advanced Multi-Project Management* (Kendall & Austin 2013), Appendix B provides a list of approximately 75 case studies of companies realizing benefits from implementing the approaches described in this paper. Benefits include:

- lead time reduction of 50% or more;
- on time delivery of 99%+;
- up to 50% more projects delivered; and
- cost savings in the millions of dollars

Conclusion

Reducing project lead times always leads to providing companies with a competitive advantage. Sometimes this competitive advantage is obvious to see: getting products to market faster, quoting shorter lead times than competitors on projects, and realizing internal benefits sooner. Sometimes this competitive advantage is harder to see: delivering more projects with the same resources lowers the overall labor cost per project

Project lead times can be significantly reduced while simultaneously reducing project costs and improving project quality by implementing the three rules of Critical Chain:

1. **Stagger** the release of projects into execution in order to limit the number of active projects in the pipeline
2. **Buffer** project plans in order to better protect against unknowns
3. **Prioritize** task execution based on project completion vs. buffer consumption

Implementing these three rules in a relatively short period of time (within 4 to 6 months) often requires software support, a proven implementation methodology and a solid understanding of the principles of Critical Chain.

About the Authors

Duncan Patrick, MBA, CMC – Duncan works with clients to assist them design and implement Theory of Constraints (TOC) based solutions focused on accelerating project execution. Duncan holds an MBA degree from Western University, Canada and a Bachelor of Commerce degree from The University of Calgary, Canada. Duncan is certified by the TOC International Certification Organization.

Jack Warchalowski, P. Eng., MBA, CMC – Jack focuses on strategy development and implementation for a variety of engineering, manufacturing and project management clients. Jack is a Certified Management Consultant and a Professional Engineer. He holds an MBA degree from the Wilfrid Laurier University, Canada and a Bachelor of Applied Science in Mechanical Engineering from the University of Waterloo, Canada. Jack is certified by the Theory of Constraints International Certification Organization.

CMS Montera provides management solutions and software to Accelerate Projects and Optimize Operations for clients in Engineering, R&D, Manufacturing and Project Management.

References

Goldratt, E. M. (1997). *Critical Chain*. Great Barrington, MA: North River Press.

Kendall, G. I. & Austin, K. M. (2013). *Advanced Multi Project Management*. Boca Raton, FL: J. Ross Publishing.

Newbold, R. C. (2008). *The Billion Dollar Solution*. Lake Ridge, VA: ProChain Press.