TEST AND MEASUREMENTS PROJECT SUCCESS

WHAT LIES BEYOND LabVIEW AND TestStand SKILLS

Filipe Altoe

To the three women who bring real meaning to my life: my amazing wife, Riva, and my phenomenal daughters, Yasmin and Juliet.

ISBN-13: 978-1497455177
ISBN-10: 1497455170

It has become appallingly clear that our technology has surpassed our humanity.
—Albert Einstein

Contents

Preface

Along the course of over seventeen years involved in test and measurements project execution at various capacities, this author invested a considerable amount of energy in obtaining the root causes for the failing test and measurements projects as a way of learning how to improve upon the results of future deliverables.

When compiling the results from the numerous root-cause analyses performed for these failed projects, as well as considering what seems to have become the project management industry consensus around why technical projects fail, there is a strong bias toward the so-called lack of well-established requirements as the number one driver for why technical projects fail. Another cause that has statistical relevance is what will be called here poor project planning.

One may say that there is nothing novice and worthy of an entire book to be written based on this conclusion as one probably either came to the same conclusion based on experience or has made use of the vast project management (PM) literature available today in order to arrive at this realization in a very matter-of-fact way. This last sentence is exactly what motivated deeper research that culminated in the work you have in your hands now. The vast majority of the project managers and other members of project teams accept the lack of well-established requirements as their number one reason for why their projects failed, in a very matter-of-fact manner. The majority of the professionals seem to be complacent to the fact that their projects failed due to lack of well-defined requirements in much the same way a cancer patient accepts her fate in face of the incurable disease. It is not uncommon to see

project teams placing the blame on the end user for lack of well-written requirements.

They seem almost content when lack of requirements definitions could be assigned as the main root cause for their project failure, as it would be an accepted excuse by the community to a failed project. This seems to be a testament to their competency as there wasn't really much they could do to salvage their projects, as the requirements for implementation weren't well defined to begin with. Their great project management or other project-related skills were put to good use, but there just wasn't anything they could have done better, since, as stated by the literature and accepted by the technical project management community, if the requirements are not well defined, the project will most likely fail.

Thinking in more general terms, this would be similar to saying that the management of technical projects is determined by a component that is outside of the project manager's control, practically being a chance event. If the project team was lucky enough to work with end users who knew what they wanted at the beginning of the endeavor, then the project would succeed; otherwise, fate would take its course.

In the multiyear process of collecting data in search for a potential root cause for why technical projects fail, at project closeout, which is the time to probe deeper into the reasons why the project failed, the almost heretical question of why the requirements were poorly defined would be asked. After receiving a patronizing look from the person who just knows that is the way things go on test and measurements projects sometimes, the vast majority of the time the blame was always pushed to the end user. The end user just couldn't make up her mind in the functionality set and would keep remembering other functions along the course of project execution that would be showstoppers if not included in the initial release.

The real fact of the matter, though, is that regardless of the main root cause identified by the statistics of failed test and measurements projects and who is to blame for it, these projects were, and still are, failing. Millions and millions of dollars are still being wasted in product development initiatives that never see the market light of the day. Excellent ideas never come to implementation fruition due to failed project executions. Hundreds of service-based organizations still go out of business due to their inability to make profits from their project-based service offerings.

There is still an overall main problem to be solved in the industry, which is to change the current statistics of test and measurements (T&M) projects' outcomes in favor of successful execution.

The majority of technical projects and T&M projects nowadays involve a higher number of different technical disciplines than their counterparts of the past. These projects now require a much deeper level of technical skills by project team members in each one of those disciplines than before. Added to that, price pressure has reached an unprecedented high. Moreover, test and measurements projects are usually at the epicenter of new product introduction initiatives. As such, the number of stakeholders that are influenced or touched by a test and measurements project is much larger than it used to be. To make matters more difficult, these stakeholders have a multitude of different backgrounds and professional skills that are not necessarily the same as the project manager of a test and measurements project.

These facts serve as motivators for the project management community to focus growing attention to this ongoing problem. If this trend is maintained without a sound response from the test and measurements industry, the current statistics will tend even more to the side of project failure in the upcoming years for this class of projects.

This can be extrapolated as to create a direct impact in the maintenance of the lifestyle that we have rapidly become used to. Product development organizations will require ever-growing product development budgets, which at some point will have to be rolled into the final end user cost of the off-the-shelf products. System integration organizations will have an even harder time remaining in business as they will either price themselves out of service opportunities due to the added risk costs that will have to be included in the final submitted proposal for technical project execution, or they will corrode their margins to the point of not being viable. It will become harder and harder to materialize into our daily lives the by-products of the latest technological advancements and discoveries. The point of the matter is that this is indeed a serious issue that deserves continuous research and effort.

If the main problems to be solved seem to be lack of well-defined requirements and poor project planning, what can be done to solve for those problems in test and measurements projects? Why don't the existing and, by now, mature frameworks of general project management and systems engineering work for T&M projects? Is it a matter of training of project managers? Is it just a matter of making sure the proper frameworks are being followed, or do we need something different than what is currently adopted by the community? Why doesn't outsourcing of T&M projects to system integrators, the experts in the field of building T&M systems, seem to fix this industry issue?

Answering these questions is what motivated the creation of the TMPM framework, a project execution framework that is tailored for test and measurements projects and that does indeed increase the odds of project success. The work you have in your hands details the process that was utilized for the root causes to be identified, which actually uncovered the real underlying issues that were driving those root causes. Once those issues were brought to life, the TMPM process was derived, focusing on addressing the real T&M project problems.

Introduction

Technological advancements introduced along the course of the last few years have changed the overall expectations of the end user for what can be considered to be a marketable product. As an example, nowadays, end users don't expect a cell phone to just allow them to wirelessly communicate with family and friends, but also to stream a huge amount of real-time data, function as a high-definition TV, and surf the web, all as a basic functionality set. And by the way, they expect to pay just a few hundred dollars for their unit. Price pressure is reaching an unprecedented level on a society that grew accustomed to not having to pay premiums for added functionality. This concept can be expanded beyond cell phones onto a multitude of other examples of the so-called daily life products of our modern society.

As a direct consequence, designing and producing a vast majority of products nowadays involve a higher number of different technical disciplines than their counterparts of the past. They now require a much deeper level of technical skills by product development team members on each one of those disciplines than before. Looking at this scenario from a test and measurements perspective, the complexity of test and measurements systems to test these products has been increasing at a much higher pace than the complexity of the products themselves. And not only is the number of technical disciplines that should now be part of the test engineering umbrella much higher. The aforementioned price pressure from the consumer base forces up the quality levels of the end user units to prevent costly recalls and loss of market share, and test times must be extremely reduced. Test time directly affects produc-

tion capacity and the capital investment on these test systems, which need to offer reduced cost of ownership.

These technical and price pressures force organizations to implement more and more complex T&M systems, which nowadays can be easily perceived as being part of what is known by the project management community as complex technical projects.

Studies performed by the community have highlighted a very alarming statistic: over two-thirds of all technical projects fail. In fact, this statistic encompasses technical projects of all sizes and industries. The experience gained through roughly twenty years involved with test and measurement systems of increased complexity indicates that this statistic is starting to become valid in the T&M industry also.

Ultimately, the professional project management discipline hasn't been totally adopted by the test and measurement industry, which still is in its infancy when compared with project management performed by technical projects executed in other areas such as construction and IT. Historically, experienced test engineers have been organically made project managers of complex T&M projects without the needed level of preparation in order for them to be successful project managers.

All the factors presented, when mixed together during the execution of complex T&M projects, invariably lead to the same outcome: test department schedules and budgets that are not met, missing test functionality requirements that lead to escapes of bad end user products to the market, inefficiencies in execution leading to raising fixed departmental costs, overall dissatisfaction of organizations toward their test departments…and the list goes on.

If you are reading this book, it means you are somehow involved in the statistics presented above. For the purposes of

the discussion presented in this book, a project is considered to have failed if either one or more of the following characteristics can be identified at project closeout: the project went over budget, the project failed to meet its planned schedule, or the project failed to deliver all stakeholders' expected results and/or quality standards, that is, the project failed to maximize the business value to the organization.

The project management community has developed and matured a body of knowledge for managing and executing projects along the course of the last ten years or so. Considerable research and advancements were made around two main frameworks that can be utilized in technical projects today: project management (PM) and systems engineering (SE). A third framework has been gaining momentum in the last few years, especially in the software development industry: agile methodology.

The Project Management Institute, or PMI, is the organ that took upon itself the mission of improving the knowledge base of project management in general. Significant advancements were made toward what is now called the PMBOK®, or Project Management Body of Knowledge.

The PMBOK® provides, at the time this book is being written, a set of forty-two processes and tools that can be utilized by project managers in the activity of managing projects. PMI certainly recognizes the importance of requirements gathering and project planning. It does offer, through the PMBOK®, several high-level processes and tools for facilitating the activities that support the execution and management of projects in general. This addresses, to some extent, the two main causes for technical project failures as previously stated, lack of well-established requirements and poor planning. However, there are some shortcomings when these processes are applied to T&M projects. The main cause of T&M project failures, the lack

of well-established requirements, is not addressed in detail by the PMBOK® as an actual methodology to execute appropriate requirements elicitation. There are indeed some high-level scope definitions and requirements gathering processes that provide a somewhat all encompassing set of tools for all types of projects; however, a more comprehensive set of detailed practices has yet to be compiled around the methodologies for requirement gathering to be applied specifically on T&M projects.

The final conclusion is thus that the current project management framework by itself still leaves the door open for failure of technical projects, a fact that is reinforced by the statistics illustrating technical project results. That drives the conclusion for the root cause of this issue not to be a matter of lack of training of project managers alone on best PM practices to be applied to technical projects, but potentially something that needs to be changed and/or added to the PMBOK® framework itself, in order to provide a more specific set of processes and tools targeted to technical T&M projects.

One common action taken by the majority of the organizations in the attempts to mitigate this shortcoming is the promotion of a skilled technical professional into the project management ranks. The rationale is that if a technical resource who understands the idiosyncrasies of T&M projects could be trained to become a project manager, then the problem would be solved.

There are a few problems with that approach. First and foremost, there are some personality traits that are requirements for an individual who is looking to delve into project management, in order for her to be successful at it. In very much the same way a person who is not good under pressure cannot be made an explosives specialist for the military, only individuals who possesses a set of specific personality traits

can become a successful project manager. Personalities are formed in the early years of the individual's life, and they are definitely not something that can be easily changed through training during adult life. What that means is that the resource pool is now constrained not only by the level of technical ability the individual needs to posses, but also by specific personality traits that would qualify that individual to be a project manager candidate.

Experience has been showing that it is very difficult finding professionals who posses deep technical skills and also have the personality traits present in successful project managers. Generally speaking, the types of personalities that pursue advanced technical training usually have incompatible characteristics with the ones that would qualify her as a project manager candidate. In summary, finding a skilled technical resource with the personality of becoming a successful project manager is not a small feat.

That being the case, even if an organization is fortunate enough to find a couple of individuals that match those criteria and is able to implement this approach, the organization becomes very heavily coupled with its human resources. This model is not easily reproducible, expandable, or sustainable for that matter. A single turnover event by a resource who qualified as a technical project manager under this model can literately set the organization back several years on its progress, and instantaneously turn success into failure. This shows the overall instability of this approach and that therefore it cannot be considered as a valid solution.

The second framework covered under this analysis is the so-called systems engineering (SE) framework. Similarly to what PMI is for project management, INCOSE (International Council of Systems Engineering) develops and disseminates the interdisciplinary principles and practices that enable the

realization of systems. INCOSE also has a body of knowledge that presents a set of processes and tools to guide the implementation of technical projects, much like the PMBOK® for general project management. The point in favor of the systems engineering framework, from here on called SE framework, is that it is dedicated to the execution of technical projects. INCOSE does understand the overall challenges involved in the execution of technical systems composed of custom software and hardware, and it provides best practices as guidance for their successful implementation.

Though the SE framework has the advantage over the PMBOK® framework of being dedicated to technical project execution, providing a focused set of processes and tools for that purpose, there is a fundamental issue with its approach. This framework is heavily based on the so-called waterfall development model.

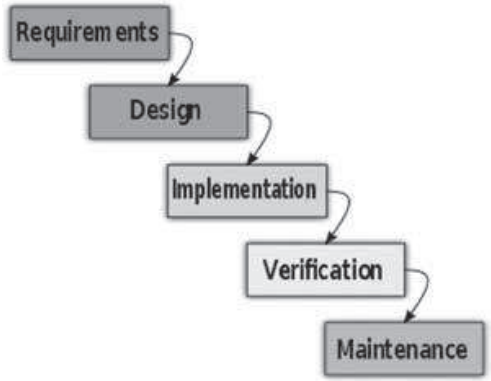

Figure 1. Waterfall Development Model

It is probably worthwhile including a brief high-level description of the basic idea of the waterfall development model in support of the argument that is about to be made. The

whole concept of waterfall development is in the idea that the overall project can be broken down into different phases and that work needs to be brought to completion in one phase in order for work on the next phase to be initiated. For instance, all requirements need to be 100 percent defined and set in stone, or, using SE's terminology, baselined, before system design can start. The main argument is that design done on a system with fluid requirements leads to wasted effort and potential rework. There are several instances in the literature that compare the cost of fixing a defect in early stages of the project life cycle versus the cost of fixing the same defect at final stages, the latter being orders of magnitude higher than the former.

Waterfall development thus emphasizes more time spent up front in the project life cycle, highlighting the importance of requirements and design documentation, which certainly carries merit. Other benefits can be perceived when a project is managed under this model, such as:

- It is easier to ramp up new team members in the event of team turnover due to the extensive level of documentation.
- It is simpler to manage since the milestones often include very tangible work products that culminate in the end of a phase.
- It provides very clear and marketable milestones.

In theory, this would be a great answer for the number one reason of why complex technical projects fail; however, in practice, the story is much different than that. For the readers who are familiar with T&M projects and product development efforts, when was the last time the initial set of requirements never changed after the design phase started?

There are several reasons why it is impossible to predict and dictate, with 100 percent certainty, the nature of

the entire requirements set of a T&M project. Uncertainties range all the way from changing market conditions, through unforeseen technological roadblocks, and even the human nature of our minds that prevents one to think through every possible scenario and exception case that can be applicable to a given complex system. This leads to the popular phrase "analysis paralysis," whereby an excessive amount of time is spent up front in the requirements gathering phase that ends up generating missed opportunity windows for the implementation of the T&M system.

This is especially applicable to highly regulated industries, such as medical device, pharmaceutical, and government-contracted projects. The fear that stems from project managers of the regulatory bodies responsible for approval of projects within these domains sometimes forces a final product to be released many years after its requirements started being elicited, causing premature obsolescence of the solution in the world of today's fast-paced technology advancements.

Another obvious example of a missed opportunity is in the scenario where the test and measurements system needs to support an organization's NPI, or new product introduction. During an NPI phase, in order for the organization to shorten the new product time to market, the test and measurements system needs to be started well before the device under test, the product to be launched, is fully developed. This causes a typical catch-22 if a typical waterfall methodology is driving the execution of the T&M system.

Under a rigorous waterfall methodology, even when needed requirements changes are identified prior to the final product release, if they come about after requirements baseline, usually the project manager brute-forces the project through completion based off of the obsolete requirements set in order to comply with the waterfall paradigm.

The immediate consequence of this is that the organization may not maximize the full business value that was intended when the T&M project was funded. In this situation, the "success" of the project takes precedence over the success of the organization. As the reader probably recalls, the very definition of project failure presented by this book includes the scenario where the organization fails to maximize the business value of the project.

This also indicates that the SE framework by itself doesn't provide the answers to the proposed questions, and neither does the lack of training and/or monitoring of the SE framework implementation during the execution of technical projects.

The "analysis paralysis" feature of the waterfall development mode drove the creation of what is now called the agile methodology. Agile has gained a good level of popularity among the software development project management community, and it is now making some strides into the overall technical project arena, where the project encompasses multiple technical disciplines and not only software engineering.

The basic ideas of the agile methodology are such that verbal communication is preferred over written documents and working products are the main measure of progress. Also, agile breaks tasks into small increments called sprints. At the end of every sprint, it is expected that the project team has a functioning working product that will get fed back onto the requirements set and will aid the product owner to make more informed decisions about how the system requirements will be prioritized. The learning that comes about with the sprint exercise is used in the tweaking of the requirements set if necessary. In theory, this seems like a silver bullet for the main problem of waterfall development, as one can make the case that it is much more efficient to document something that is already completed. Also, one can argue that it becomes much

easier to adapt to the inherent characteristics of complex technical projects, which is the changing nature of its requirements set, as more is learned about the system. With this model, potentially, less effort can be spent on a collection of short-term tasks than on sometimes multiyear-long ones until a workable product is seen. An overall more adaptable process unfolds based on the constant feedback from the many working products that are generated throughout the overall project life cycle.

Unfortunately, theory and practice don't usually go hand in hand in the case of agile development. The lack of documentation in favor of working products more often than not ends up creating too casual of an approach to project execution by the team members, who find in this feature of agile the perfect excuse to avoid less glamorous tasks such as documenting one's work. This creates tribal knowledge and consequently a heavy dependency on individual team members who possess knowledge snippets of the overall project.

Another issue that is somewhat obvious to see is the difficulty of determining an overall project budget and schedule up front. Due to the dynamic nature of the project requirements set in an agile environment, sometimes just a very high-level wish list is created as opposed to an initial requirements set, which makes developing an accurate forecast of schedule, budget, and resources a close to impossible task. This information are fundamental inputs to the organization in its efforts of manage its projects portfolio, as well as setting up an NPI budget. Imagine an executive having the task of prioritizing product A versus B without having access to an analytical set of financial data concerning those projects. Not only that, but since under this model the requirements are so fluid at the beginning, usually system integrators can't really submit a proposal for the execution of a T&M system if this framework is in place.

The last sentence above mentioned the system integrators, and they will be another focus of attention of this book, chiefly around their utilization by clients for the execution of T&M projects. System integrators are herein being defined as professional organizations whose business it is to execute the requirements elicitation, project management, design, implementation, verification, and deployment of T&M systems. As mentioned in the paragraphs above, T&M systems complexity has been increasing exponentially, and as such, it sometimes become difficult for organizations whose main business is not to implement T&M systems to have the needed level of expertise on staff to execute such projects in the time frame its business demands. These organizations usually hire system integrators as the experts in the T&M field in order to shorten the implementation time for the needed systems as well as to reduce the overall deployment risks. Opportunity cost is the name of the game, and as such, hiring an organization whose main business is the execution of T&M systems to run the project sounds like the most reasonable choice.

Contrary to what many may think, this also brings about challenges to the successful execution of the project. Regardless of the level of expertise the hired system integrator brings to the table, as this book will show, there is a set of problems that need to be addressed to foster success of this relationship. The fact that the organization has an expert system integrator in charge of the project is not, in itself, an immediate guarantee of success; however, it is a good first step for sure.

There are five scenarios very common in the relationship between clients and system integrator companies that are important drivers for project failure. Each one of these scenarios will be described in detail in a later chapter.

The fact of the matter is that it is ultimately the client's responsibility to make sure the business value for the T&M sys-

tem being outsourced to an integrator is realized. Even though the integrator may share some of the same interests, at the end of the day, the integrator is a for-profit business, and its behavior will be highly influenced by what maximizes its success. What this book proposes is a methodology whereby the client places itself in the driver's seat and makes sure there is a clear alignment between the integrator's best interest and the maximum business value for the T&M system being delivered. This also adds substantial value to integrators, as a successful engagement is the number one driver of repetitive business from a client. It is also in the best interest of the integrator that its client maximizes the business value for the system it is contracted to deliver.

Since the two frameworks and the agile methodology have positive processes, tools, and methods, those being somewhat complementary when applied to T&M projects, this provided a good indication that the answer we were looking for would be in some form of a hybrid framework, composed of an adaptation and combination of the relevant processes and tools of PMBOK® and SE frameworks and the agile methodology, into one that is targeted to T&M projects. Moreover, since the involvement of a system integrator is not a guarantee of success for the project either, this hybrid framework needs to address the issues that are usually present when the organization trusts a system integrator with the T&M project implementation.

A more holistic approach to T&M projects is needed, one that would apply not a single person functioning at a capacity of project manager and systems engineer, but a multirole management team. On this new framework, one role would function as the project manager for the project execution itself, bringing all the offered benefits of the PMBOK® framework. Another role would function as the systems engineer, responsible for the technical leadership of the project and bringing with her

all the offered benefits of the SE framework. A third role would function as the person making sure the organization business value is being maximized by providing the project manager and systems engineer with all the needed direction, information, and tools in order for them to be successful at their roles, a business analyst of sorts, but focused on T&M. This third role, as it will be seen along the course of this book, will make sure that project success is not just the completion of the initial project scope on schedule and on budget or to make sure the contractual obligations of a system integrator have been fulfilled, if an integration is being used to implement the T&M system. This person will make sure that once the project is completed, it maximizes its business value to the organization. Furthermore, this hybrid methodology should maximize the benefits of the PMBOK® framework, SE framework, and agile methodology, and try to minimize and/or eliminate their shortcomings.

This book has three main high-level goals:

1) To explore the root causes for T&M project failure and to determine the real reasons why these project fail
2) To explore the root causes for failure in engaging a system integrator company
3) To provide a modified framework that facilitates both the successful management of T&M projects as well as the engagement of system integrator companies

This book will present a high-level analysis of the PM and SE frameworks and agile methodology, focusing on their respective gaps and strengths when applied to T&M projects. The same analysis will be performed to cross-reference the aforementioned frameworks and agile with the known issues in engaging system integrators.

This discussion will motivate the introduction of the TMPM framework, which will be presented in detail. The

TMPM proposes a new organizational structure along with a collection of processes, tools, and best practices that address the gaps identified in the conventional PMBOK® and SE frameworks as well as the agile methodology for the successful execution of T&M projects. The TMPM also includes an organizational structure model and life cycle implementation that facilitates the engagement of system integrators for the execution of T&M projects.

This book is targeted to test engineers; professional LabVIEW and National Instruments consultants; project managers of test and measurements projects; test managers and any other functional managers that are involved in T&M project execution; engineering and product development executives of service, technology, and product development organizations; and any organization that is faced with the challenging business of implementing and managing T&M projects.

The book is broken down into two main sections. The first section describes the problems that surround T&M project implementation in detail. The second section proposes a new hybrid framework, named TMPM, to address the problem statement specified in section 1.

Chapter 1 presents a root-cause analysis for the two main drivers of project failure, a detailed account for the reasons why T&M projects fail. This analysis is basically the application of a typical problem-solving technique, the goal being to break a large, difficult-to-grasp high-level problem down into smaller, easier-to-answer questions. With the problem at hand, the first step was to perform a thorough data-driven root-cause analysis of the problem in order to find underlying issues.

Chapter 2 presents a detailed analysis of the main issues that drive T&M project failure when system integrators are being utilized to execute projects for an organization. The aforementioned five main issues with this engagement are detailed and complemented with illustrative use cases.

Chapter 3 focuses on a detailed presentation of the current frameworks utilized to manage T&M projects, the ones that are typically used by organizations in the execution of these projects. It also presents a gap analysis of these frameworks, highlighting their strengths and weaknesses when specifically applied to T&M projects.

Chapter 4 repeats the analysis of the current frameworks, but with the focus of cross-referencing the challenges identified in the engagement of system integrators. A similar gap analysis to the one presented in chapter 3 will be determined, but now in relation to the system integrator's engagement challenges.

Chapter 5 summarizes the problem statement presented in section 1 of the book. This problem statement is used as a foundation for section 2.

The second section focuses on presenting the TMPM framework. Chapter 6 introduces the framework. Through a project example, it presents the typical T&M project team structure and the various stakeholders that are usually involved in such projects.

Chapter 7 presents the organizational model that best fits the proposed framework and a description for the multiple roles and interactions between these roles. It also includes a suggested role to make sure the engagement of a system integration company is best aligned with the organization's business value for the T&M system. This chapter will focus on the people element of the typical people, process, and tools combo that defines a given framework.

Chapter 8 focuses on the tools element of the framework. It provides a summarized introduction to UML, Universal Markup Language, and how the UML available diagrams can be utilized to address the root issues for T&M project failure

via system modeling. It also suggests a methodology for better manage the T&M project stakeholders.

Chapter 9 brings it all together by demonstrating how the organizational structure and tools presented in the two sections can be put together in a process that is tailored to T&M projects.

Chapter 10 presents an overall summary for the entire book. It ties the problem statement derived in section 1 to the TMPM presented in section 2.