



Software project management – reaping the benefits of an installed software quality management system (SQMS)

P.W. Chan

*Information Communication Institute of
Singapore, 1 Hillcrest Road, #08-00 Tower Block
2, Singapore Telecommunication Academy,
Singapore 1128, Republic of Singapore*

ABSTRACT

According to Burrill and Ellsworth [1], “Increased size, complexity, involvement of many areas, and the demand for quality will all generate an increased need for effective project management.” This paper outlines a framework which project managers can employ to develop their own effective game plan for managing software projects. The framework has several ingredients. The first ingredient is about laying the foundation for tackling software projects. The second ingredient is a three-phased software project management lifecycle model - assessment, planning, and control. The paper will discuss what the purpose, the tasks, and the outputs of each phase are, as well as the need for repeating the cycle. The third ingredient is the three project management work aspects - process, product, and people management. Those aspects help managers to identify tasks which must be done in their own specific project. The paper will also illustrate how the work aspects can combine with the lifecycle model to a good effect. An example will be given to demonstrate the usefulness of the proposed framework.

1. INTRODUCTION

Quality, or the lack of it, is a costly business. It is therefore important for software organisations to take bold steps towards improving the quality of its software processes and products. An essential step would be to

688 Software Quality Management

develop and implement a software quality management system (SQMS) which covers the entire software development lifecycle, including maintenance and user support [2]. However, an organisation's software quality management system does not actually produce quality software product. It instils the discipline for such product to be produced. It is project work and the management of it will produce a quality software product. Therefore, to serve its purpose, the installed SQMS has to be applied to specific projects. Application is the necessary test for the suitability and applicability of the SQMS to software development. It is no more appropriate than to see the potential benefits of the installed SQMS realised through its use in the context of software project management.

In managing a project, every manager faces constraints. In software projects they are cost, schedule, scope, and resources. The art of software project management is about balancing these constraints and yet achieving the project goals. To do that, a proper game plan is needed. To "cook up" this game plan, three ingredients are necessary. First and foremost is a firm foundation whereupon project managers can build their management activities. What important to this foundation are a useful definition of what a project is, a company-wide project policy which will direct the effort of developing measurable goals suitable for planning and control purposes, and finally a strategy to tackle projects of any sizes. Second ingredient is a project management lifecycle model applicable to software development. The model proposed in this paper consists of three phases - assessment, planning, and control. Those typical tasks in each of the phases are discussed, as well as the need for these three phases to be cyclical in order to effectively manage a software project. The final ingredient is a mechanism to help project managers to identify work specific to their project requirements. This is provided through the three aspects of management work - process management, product management, and people management.

2. LAYING THE FOUNDATION

The first step to effectively tackle a software project is to have a useful game plan that will allow project managers to have the project under control at all time. The first and foremost important in the process of deriving the game plan would be to lay a foundation. Three important pillars would have profound influence on the foundation - a proper understanding of what a project is; a policy-driven measurable goal at which the manager can aim for and from which the performance and the result of the project can be measured; and an effective management strategy. Each of these will be discussed in the followings.



Many a time, project work effort has been treated, though perhaps unconsciously, as a continuing effort. It is no wonder that those misguided projects would go on forever. According to Demarco [3], 15% of all software projects never deliver anything and overrun of 100% to 200% are common to software projects. To avoid such situations, software organisations should adopt a project definition which in turn can be used to derive some powerful rules governing the lifespan of a software project. Burrill and Ellsworth [1] have offered such a useful definition. They define project as a formally established, one-time effort with the following characteristics:

- An established beginning or start date,
- A well-defined purpose and scope,
- A formally established, documented cost-benefits basis,
- A well defined project product, including product performance criteria,
- An established endpoint or completion date.

Under the definition, it is clear that project managers must seriously view their project as a definable entity right at the start. It encourages them to draw a definite boundary around the project by clearly establish its purpose, scope, justification, and product. The definition does not imply that setting the start and end points or any other characteristics is an one off activity. They could be altered as long as each alternation is formally established, that is, approved and baselined, throughout the lifetime of the project. Those characteristics will also help project team members to recognise if they are in the project, what they are trying to achieve, why they they are doing what they are doing, as well as to define the process to achieve the desired results.

Another pillar in the foundation is a company-wide policy on project. This will be a statement about senior management commitment to successful project implementation and how they intend to support that commitment. The policy is best translated into a management goal which allows project managers to have a basis to measure the effectiveness of his project process and at the same time form a yardstick to balance the constraints that they face throughout the project lifetime. Generically, the goal of software project management is "to use resources, especially human resources, effectively and efficiently, in order to produce a software product on time, within budget and that the product meets the requirements specified for it". The goal focuses project managers to four dimensions of measurements - cost, time, scope and quality. Each measurement dimension must be as clearly specified as possible and yet

690 Software Quality Management

bearing in mind that they have impact on each other. For example, the more functionality required in a piece of software, the more time and cost will incur to build it. On the other hand, if time and/or cost exceeds the threshold limit, then negotiation could take place between the customer and the supplier to reduce the number of functionalities in the software product. The specification of those targets must be set and set right at the beginning of the project and maintained throughout. The project management goal tie up very well with the project definition. To illustrate their close relationship, for instance it is not possible to set those targets on cost and duration without first establishing the start and end points of a project. In the same token, a well defined product requires specifying clearly and precisely the software functionality. So it is clear that the goal setting works complimentarily with the definition.

Both the project definition and the management goal are absolutely essential to the manager to define and scope the work effort required. However, one other pillar is needed in laying the foundation - an effective strategy to tackle project work. The strategy follows the good old principle on how to eat an elephant - one piece at a time. It is well recognised from experience that managing a software project is a complex matter with many ever changing variables. It is in the project manager's own peril in trying to handle all these variables all at once throughout the project life span. A more effective way would be to tackle the complexity by following the strategy of divide and conquer. To simplify project planning and control, project work can best be divided into a hierarchy of smaller pieces using the work breakdown technique [1,4]. However, dealing with a piece of complex work such as software development needs more than just breaking it down into pieces. The work breakdown structure will only be useful as an effective weapon to divide and conquer software development work when each piece of work in the structure is treated as a stand-a-lone project and applied with the same discipline of goal setting and project definition as discussed previously.

One technique which can energise the combined power of these three pillars is the input-process-output (IPO) model. Work is a function that transforms certain inputs into outputs [1]. The collection of these transformation steps are usually referred to as a process. For example, a software development process is about transforming customer's requirements into a software product comprising of a set of operational programs together with their related documents. The IPO model, as applied to project work, forces managers to draw a clear boundary around every piece of work in the project by the fact that managers must define the required inputs and output as well as the process needed to transform



the inputs into outputs. Each IPO model, by its definition, is a self-contained, stand-a-lone entity that can be subjected to the rigorous discipline of project definition and goal setting. In summary, effective project work requires first applying the IPO model to the project as a whole - the formation of a global picture to direct the project. This will ensure that the project is well defined having specific start and end dates, well defined purpose and scope, a justifiable basis, specific product, and a set of related project management targets. The project is then broken down into several manageable hierarchical pieces with each piece represents a stand-a-lone project and be managed as such - the formation of a detailed picture to substantiate the global variables. As seen here, the IPO model can be applied to any granularity of work, whether it be the entire project, a phase, a task, or even a work unit which can be assigned to an individual. In the following Sections on project management, the word project refers to not only the entire project but also each of these stand-a-lone pieces.

Taking seriously the IPO model, project definition, management goal setting, and work analysis and breakdown, project managers have taken the first step to a disciplined approach to tread the path of project management. An approach which allows planning from a top down viewpoint of subdivision and control from a bottom-up viewpoint of aggregation. Having taken the first important step, project managers must now tackle the operational aspects of project planning and control.

3. SOFTWARE PROJECT MANAGEMENT LIFECYCLE MODEL

Like technical development, software project management has its own life cycle during software production. There are three phases in the software project management lifecycle model, namely, assessment, planning, and control (Figure 1). The purpose, tasks, and output of each of these three phases will be examined in the followings.

The purpose of assessment is to provide management with information about the worthiness of the project. Assessment is an attempt to answer the question: "Is the project worth doing or continuing?". A project is not worth investing into it if it is not technically feasible or cost justifiable. It will also be a grave mistake to assume that once a project has been started the team must complete it at all cost. The organisation should regularly ask that question and satisfy itself that the project would yield the benefits that it has intended for. Regular intermediate assessments are just as important, if not more important, as the initial assessment because of the

692 Software Quality Management

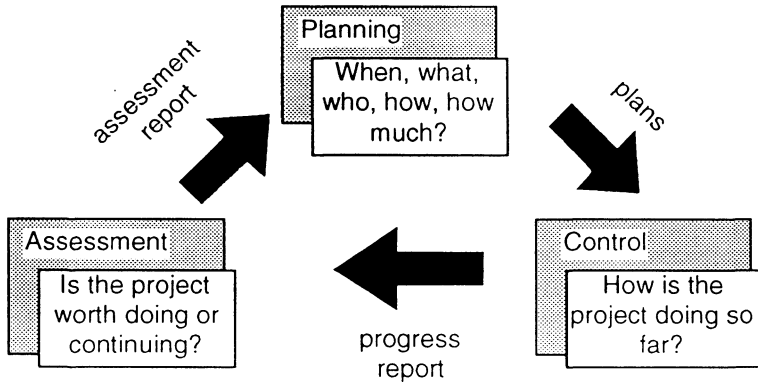


Figure 1 - Project Management Cycle

many factors affecting the worthiness of the project will change over the project life span. Indeed, the longer the project, the more likely will the changes to those factors happen. The assessment interval depends on the contract requirements and the project process. For instance, an assessment can take place at the end of each technical development lifecycle phase. Assessment will not only help project team to assess past performance but also to anticipate the occurrence of any adverse events in the future and plan to tackle them if and when they do occur. Indeed, assessment forms the basis of management commitment to the funding of the project. Assessment phase requires the following tasks:

- **Risk Assessment.** All projects inevitably face risk of one kind or another. Cost overrun, being late, key staff leaving, product does not meet customer's expectation are but a few risks facing the project team. Risks must be analysed, measured and controlled. The principle is to highlight any potential problems rather than hiding them and face the dire consequences [5].
- **Cost/benefit assessment.** All projects must justify its existence. No organisation should commit valuable resources on a project which does not yield sufficient benefits in return. "Cost/benefit analysis is an attempt to measure the relationship between the cost of resources and the value of benefits, and to use this measure to decide if an investment is worthwhile" [1].
- **Technical capability assessment.** The project team must assure itself that it has what it takes to successfully accomplish the project. This is



an assessment of the organisation's engineering methods, procedures, standards, and tools.

- Process capability assessment. Project managers must earnestly ask themselves if their software organisation is ready to take on the challenges of the project at hand. Does it have the necessary management methods (project management, quality assurance, configuration management, etc.) as well as other standards and procedure to accomplish the project [6].
- Productivity assessment. Finally, project managers must determine how to increase the efficiency of their most valuable asset - people. The issues include determining the skill and qualification requirements and then hiring the right person for the job, mapping out the training requirements, and provision of productivity tools such as CASE tool and methodologies.

The outcome of the assessment phase is usually a feasibility study, a project proposal, a tender document, or simply an assessment report. Its intention is to solicit management commitment to initiate or continue with the project. The assessment phase product should contain sufficient information that would allow management to make an informed decision with respect to the worthiness of the project.

If the outcome of the assessment is positive, then project planning phase begins. Again, Burrill and Ellsworth offer a useful definition on planning. It is defined as "a process that starts with a mission, fixes goals and objectives to be reached, develops plans and procedures for reaching them, establishes an organisation for archiving this, and assigns responsibility and accountability to see that it is done". In this phase, project managers make decision on such things as cost, schedule, scope of the project, and resource requirements. As mentioned earlier, these four items are at times in competition. The purpose of the planning phase is therefore to resolve any conflicts imposed by these competing requirements. At this stage, project managers will attempt to answer the question: "when should what happen by whom, how, how much". Part of the answer to this question will inevitably lead to the determination of how much should the project accomplish, how long will the project last, what is the cost of the project, what and when equipments are needed, and who should be doing what job. If the assessment phase is to help management to make a decision on whether to commit company's valuable resources to the project, then planning phase will allow management to have a clearer picture on what they are committing



694 Software Quality Management

themselves to. Planning phase requires the following tasks:

- **Work analysis.** First amongst the key tasks for project managers to compile a useful project plan is the analysis of work based on such factors as project purpose, contract requirements, output required, etc. The outcome of this task is usually a work breakdown structure (WBS) depicting the various pieces of work required in the project and their hierarchical relationship.
- **Sizing and Estimation.** Each piece of work in the WBS should be sized properly in order to provide a baseline for the estimation of other variables. To date, the two most popular methods used for measuring size of a software product are Line-of-Code and Function Point [7]. Once the size has been established, other estimates such as effort, duration and resources requirements can be determined. Some estimating techniques such as COCOMO provides a straight forward derivation from size to manpower effort, followed by duration, resource requirement and other variables.
- **Scheduling.** Having identified, sized, and estimated the effort required for every piece of work in the project, it will be necessary for project managers to schedule them in a calendar and determine the end date of the project. It can be done using a bar chart or a network diagram [8]. Some factors must also be taken into consideration when compiling a schedule. The dependency relationship, the number of people available to the managers at any given time, the optimum number of personnel to carry out a particular task, and the availability of skills.
- **Costing and budgeting.** When all these tasks have been scheduled, project managers would have sufficient information to work out the total cost and prepare a budget. The budget should indicate both the specific and summary costs.
- **Work assignment.** Work assignment represents the first step to execute the plan. It is the time when individuals are given the responsibility and accountability for the accomplishment of a piece of work. It also, for the same token, signals the start of the control phase. Assignments have to fit both the person and the need of the situation. All the planning tasks up till now deal with objective, impersonal work. However, work assignment is by far the most difficult task because it begins to involve human being.

Managing Quality Systems 695

Since planning is about making decisions, it is hard work. Furthermore the many variables involved in the planning process rely on the sizing and effort estimation exercises. With the ever increasing complexity of today's software, planning work is not going to get easier. At the same time, the work analysis plays also an important role in the planning process since it is that task that determines the work needed. Obviously, the better the analysis and breakdown, the easier the sizing and estimation exercises would become. The key planning documents are the project plan, the quality plan and the resource allocation plan. These plans and their subsequent updates must be baselined so that they can become the basis for measuring progress and deviation.

Plans must be executed. If planning is about making certain decisions as to how to accomplish a project, control is the execution of those decisions. In the control phase, project managers attempt to answer the question; "How are we doing so far? Do we need to make any adjustment? Should management be informed of the present situation?" Since project managers constantly face the competing targets on cost, schedule, scope, and resources, the control part of the management cycle will concentrate on these four also. Control phase therefore requires the following tasks:

- **Quality control.** Quality is monitored through appropriate reviews and inspections that have already been built into the work schedule. It is worth noting that quality monitoring is largely qualitative and rely heavily on the experience of the reviewers.
- **Cost control.** To control cost, project managers must first decide a cost monitoring method which would include cash flow projection, cost data collection, and variance calculation. As the project proceeds through development, project managers soon find out the accuracy of their cost estimation [9].
- **Schedule control.** This is very similar to cost control. However, one thing worth noting is that work monitoring is best based on completed work. That is why a proper work analysis and breakdown is so important [1,9].
- **Performance monitoring.** Project managers may find it useful to focus their attention on two key questions "What would help the team members most to do their job?" and "What would hinder them most from doing their job?". Then it will be a matter of putting the helpful in place and getting rid of the hindrances.

696 Software Quality Management

One final point about the proposed three-phased project management lifecycle model is that the cycle repeat itself throughout software development. The number of repetitions depends on a number of factors such as organisation's control structure, contract requirements, project methodology, financial approval procedure, etc. The cycle is not one off because the project variables, due to changes in circumstances, will change too. To cope with the increasing complexity of software and those ever changing project variables, a better strategy to tackle project work would be to divide work into smaller pieces and manage each as if it were a project. The cycle would then be applicable to each such project. In fact in Section Five, we will discuss a possible methodology to deliberately plan for such repetition of the cycle - a proactive rather than a reactive approach. However, before looking at such a methodology, it is important to examine project management work aspects which will help managers identify work suitable for their specific project.

4. SOFTWARE PROJECT MANAGEMENT WORK ASPECTS

Central to the activities of all three project management phases, especially in the planning phase, is the determination of work requirements. To determine what activities which must be carried out to manage software projects, managers may find it useful to focus their attention on the three aspects of work - process management, product management, and people/performance management. Process is defined as that set of tasks when properly performed, produce the desired result/product [6]. Process management is about managing the production process which is central to all software development. It is about the determination and organisation of the various analysis, design, programming, testing and implementation methods needed to develop the software product. The primarily concern with product management is quality. Its emphasis is on putting a mechanism in place to ensure that a right product has been specified and that the final software product meets that specification. In other words, product management runs along side with the production process to make sure that it is being carried out correctly. The last and probably feature least prominently in project management work is the management of people. This is about making the best out of our scarcest resource - people. Despite heavy push for automation, software industry is still labour intensive. It is therefore hard to imagine an organisation can produce on-time, within-budget, quality product without treating its people right. The need for leadership, team building, choosing the right people for the job, performance planning and monitoring are some of the key issues in people management. Paying adequate attention on all three aspects will help project managers to

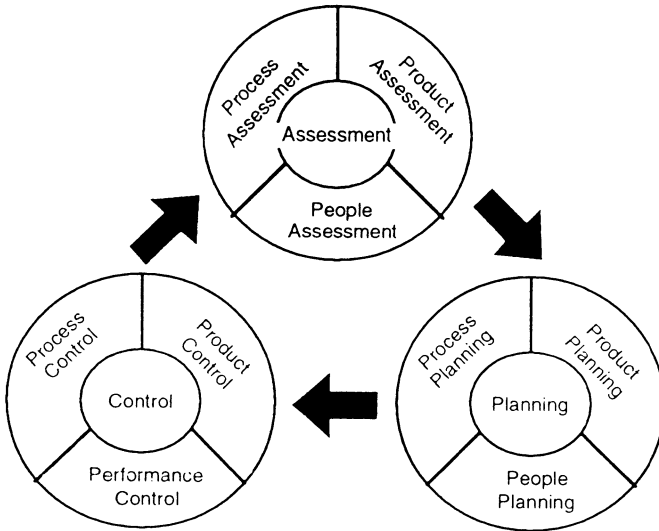


Figure 2 - Combining Management Cycle With Work Aspects

identify the necessary management tasks and ensure successful project implementation. It also ensures that, under pressure, they will not just concentrate on one and compromise on the other. As project managers face the competition on cost, schedule, resources and scope, they will need to balance the work in these three aspects to ensure that adequate planning have gone into each of these. Therefore, in order to help project managers to focus their minds on his assessment, planning, and control phases, it is thus helpful to consider the aspects separately and then together in a coordinated manner.

Combining the project management lifecycle phases and the three work aspects (Figure 2), the project managers do have a useful tool to tackle project work. For example, to consider the risk assessment task within the assessment phase, project managers could take into account of

- process risk - schedule miss, cost overrun, equipment down time
- product risk - the response time is too slow, the operational cost is too high, product outdated
- people/performance risk - high staff turnaround, low morale, lack of skills



698 Software Quality Management

So as the managers go through their project management phases, the three work aspects will focus their mind to ponder over the work required. Ultimately, all these tasks form part of the work breakdown structure and hence impact the cost and duration outcomes of the project.

5. SOFTWARE PROJECT MANAGEMENT IN ACTION

To see how the framework of project management lifecycle model, management work aspect, and a disciplined approach to the treatment of a project could help project managers, an example is given below (Figure 3). The example is based on the phased approach promoted by Burrill and Ellsworth [1].

Usually, a project is initiated when the company is responding either externally to a request to tender or internally to an initial idea. In either case, it forms the input to the first project management phase - assessment. It is important to start off the assessment phase bearing in mind that this initial assessment is a project on its own right. That is, it will have its own specific start and end dates, well defined purpose and scope, a justifiable basis, and specific product. Also, it will have its specific project management goals in terms of resources, cost, time, and scope. The next important activity would be to determine the tasks required for the initial assessment. The tasks listed under assessment phase in the Section Three of this paper provides the first level of work breakdown. Armed with the assessment purpose and required product, project manager can examine each task in turn, gives due consideration to the three aspects of project management work, and decides what exactly are required to do (the second and subsequent levels of work breakdown). He or she will eventually form a plan for the initial assessment phase. Once the plan is approved, work will commence. One would immediately notice that even within the initial assessment phase, the manager follows the same pattern of assessment, planning, and control cycle. The output of this phase is normally a tender document or a project proposal or a feasibility study report.

Following the approval of the initial assessment recommendation, project manager moves onto the second phase of project management - planning. Needless to say, the planning phase will be treated as a project in the same way as the initial assessment phase. Obviously, this initial planning phase is important in that it sets the scene for the rest of the project. One thing worth highlighting in this initial planning phase is that it will include two separate stages of work. They are the overall project planning and the detail phase planning for the next phase of work. The

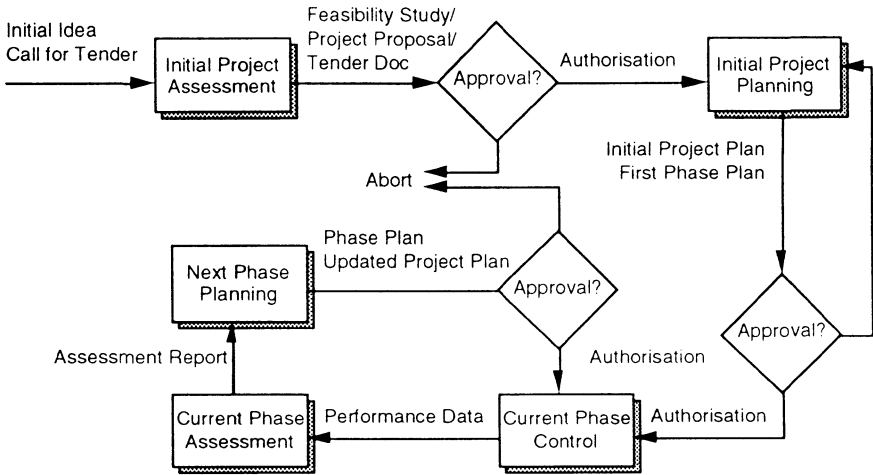


Figure 3 - Project Management Framework in Action

project manager will first carry out the tasks as listed out in this paper for the planning phase, targeting them at the overall project level. Again, project manager must take all three aspects of work into consideration especially in the work analysis task, since it will affect the variables such as project cost and duration tremendously. Those variables will be the initial basis upon which management will commit, plan, and track the progress of the project. However, as pointed out earlier, a useful strategy to tackle a sizable project would be to divide it into smaller pieces and manage it one piece at a time. The work analysis task carried out in the overall project planning part of work would give the project manager the necessary start to determine the next phase work. However, the data in the overall project plan is not detail enough to give him the comfort to manage it. In other words, planning tasks carried out for the overall project plan will have to be performed again at this first phase planning stage. This time, the variables will be at a lower level. A level detail enough for senior management to actually commit funds to the work of this phase. The outcome of the initial planning phase is therefore an overall project plan and the first phase plan.

Upon approval given, the first phase work begins and the manager moves onto control mode. Now the project manager is in the micro level of phase management. Each phase again is a project and must be treated



700 Software Quality Management

religiously as such. Each phase would go through the usually assessment, planning, and control cycle of project management. At the end of the first phase, the actual will be compared with the planned data and an assessment will be carried out to examine if it is worth continuing the project. To make such decision, senior management requires at least three pieces of information. An assessment report on how the first (or subsequent phase depending on where the project is) phase has performed and what can be learned from it. A next phase plan to specify what will be the next step of commitment. An updated project plan to give senior management an idea of the expected cost and duration to completion. In other words, current phase assessment and next phase planning should be carried out one straight after another. However, it must be pointed out that it would be legitimate to abort the project if phase assessment suggests that it is no point to carry on.

Accepting the project manager's recommendation, the project once again moves onto the next phase control and the cycle repeats until the project is successfully implemented.

6. SOFTWARE PROJECT MANAGEMENT AND SOFTWARE QUALITY MANAGEMENT SYSTEM

Installing a software quality management system (SQMS), such as the one based on and possibly accredited to the International Standards Organisation (ISO) 9001/9000-3, would set an umbrella for overall software development activities. That would certainly include, amongst other things, technical development, project management, configuration management, and project quality assurance. However, the Standard itself specify only the requirements for such an SQMS, not its implementation. The framework proposed in this paper will contribute towards helping software organisations to implement their project management process within the umbrella of SQMS. Unfortunately a full description on how the project management process could integrate into the SQMS documentation would carry the scope and the length of this paper way beyond its limit. In the followings, however, a few suggestions on such integration are given.

It is generally recognised [2] that an installed SQMS would have a quality manual (specifying the "what") and a procedure manual (specifying the "how"). In a way, the quality manual is the organisation's quality plan stating exactly how it has committed herself to produce quality software on time and within budget. The quality manual contains largely policy statements and refer the "how to" to a procedure manual. When needed

to apply to individual projects, the quality manual will be used to develop the specific project plan and quality plan. The project management process must therefore be integrated as part of the quality manual as well as the procedure manual.

- Incorporate a definition of what a project is into the definition section of the quality manual. A definition which is similar to the one discussed in this paper could be used.
- Incorporate a project policy statement stating categorically what an organisation's project management goal is and how its senior management intends to support the achievement of such goal. The statement could be published in the same section as the organisation's quality policy.
- Formalise a project methodology specifying the organisation's project management process. The methodology will specify, amongst other things, the project management phases and, for each of the phases, the purpose, tasks, and outputs. This will require adopting the proposed framework. The methodology could form the third base document for the installed SQMS, together with the quality manual and the procedure manual.
- A procedure in the procedure manual outlining a guideline on how to classify projects into different sizes and how to customise the project methodology to suit these different sizes.
- Document all the management methods, procedures, standards, and tools used by the project management tasks as discussed in Section Three of this paper in the procedure manual
- Provide checklists for the consideration of the three management work aspects such that project managers can them use to design exactly what must be done in management their project.

Since it is a certification requirement that the supplier must demonstrate to its purchasers that it has the capability and the capacity to producing quality product. This demonstration is through use - the use of SQMS on project. Nothing is more central in this demonstration process than the project management work carried out by the project team. If quality improvement is a journey, then successful project implementations are milestones. The more milestones an organisation achieves, the more mature they are in this quest for quality. Project management is a crucial



702 Software Quality Management

vehicle to lead the organisation from one milestone to another.

REFERENCES

- [1] Burrill, Claude W. and Leon W. Ellsworth.
Modern Project Management - Foundations for Quality and Productivity.
The Data Processing Handbook Series, 1986.
- [2] Chan, Ping W.
"Installing An ISO9001 Accredited Software Quality Management System" Proceeding of the First International Conference on Software Quality Management, Southampton, UK, March 1993, pp13-26.
- [3] DeMarco, Tom.
Controlling Software Project. Prentice Hall, 1982.
- [4] Tausworthe, Robert C.
"The Work Breakdown Structure in Software Project Management"
The Journal of System and Software, Volume 1, 1980, pg181-186.
- [5] Charette, Robert N.
Software Engineering Risk Analysis and Management. McGraw-Hill Book, 1989.
- [6] Humphrey, Watts S.
Management the Software Process. Addison-Wesley, 1990.
- [7] Albrecht, A.J.
"Measuring Application Development Productivity" Proceeding of the Joint SHARE/GUIDE/IBM Application Development Symposium, October 1979, pp83-92.
- [8] Cori, Kent A.
"Fundamentals of Master Scheduling for the Project Manager"
Project Management Journal, June 1985, pg 78-89.
- [9] Kezsbom, Deborah S., Donald L. Schilling, Katherine A Edward
Dynamic Project Management. John Wiley & Sons, 1989.