Quality Assurance In Software Life Cycle

Paper on

In Which Way Software Process Improvement Together With BSC Can Help To Increase Process And Product Quality In Small Software Development Companies

by

Ganesan Vijayaraghavan
(ganesanvijay83@yahoo.de)

Date: 01/06/2009

Table of Contents

List of Figures

Abstract

Computer software companies have taken various steps and have come up with different tools to improve the quality of their product development processes and thereby increase their product quality and customer satisfaction. Software process improvement together with Balanced Scorecard is one such technique for improving quality which has gained wide acceptance among software developers. A combination of these two is proposed to boost quality in the software development process of these companies. Small software development companies are bound to benefit from this combination. This write-up highlights the usefulness of these techniques and their advantages to small software development companies.

Introduction

The success of any organization depends to a great extent on the importance given to 'Quality'. According to quality assurance glossary, quality is referred to as "The totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs"(CIMS, 2002)[13]. Quality includes both process quality and product quality. Software process can be defined as "a set of activities, methods, practices and transformations that people use to develop and maintain software products"(Richardson, 2000)[1].

If a company aims for a good quality software system, it must ensure that each of its parts, processes, and products is of high quality. The focus on process quality not only ensures production of high quality products but also improves productivity and eliminates defects.

The competitive position of a business organization is greatly affected by the quality of the software product developed by the company and the development process that goes into producing the product. The role that Information systems play within the organization also affects this position (Barnett, Raja, 1995)[2]. Software companies are under increased pressure to deliver operational systems and their other products on-time. Small software development companies face this pressure much more than bigger firms as they face a higher crunch of resources.

This article will examine the techniques that have been proposed in the literature for moving the quality focus from the software product to the software development process. Each of the techniques, namely, software process improvement and balanced scorecard will be examined in great detail for better understanding.

Need for process quality in small software development companies

Developing software systems on time, within budget, and with the agreed quality is definitely a difficult task. During the past years, the productivity of software engineers has increased considerably. Their educational level and that of their managers has certainly improved. New tools, methods, techniques, languages, and environments have been introduced with a rapid pace. However, software engineering projects still posses the same characteristics: they are often late with frequent schedule slips exceed budget, and the product does not satisfy customer's specification.

These are serious issues for small software development companies that can push them to the edge of the cliff. Small software development companies face strict budget of time and resources such as finance, manpower, etc. Poor quality systems consume additional resources and contribute to backlog of development projects that exists in most companies. The lack of advanced information technology support, prohibitive costs, financial crunch to invest in research and development efforts, and the lack of software process improvement knowledge degrades the competitive position of these small software development companies.

However, in order to compete in the global market, these software developers should concentrate on improving their productivity, time to market, and customer satisfaction. To establish themselves in the market and to grow higher, these firms need to be efficient in their processes and be able to deliver high quality products with minimal resources. These issues call for tools that can help these firms achieve their goal.

The purpose of this paper is to understand and analyze these available tools that benefit small software development companies.

Tools to increase process quality

Engineering-like discipline has long been proposed to be the most efficient process in the software development activity by software engineering and information engineering. One such method which uses automation for the improvement of the developer productivity is Integrated computer aided software engineering (I-CASE)(Barnett, Raja, 1995)[2].

"The software engineering community seems today to commonly agree that the focus should be shifted from single methods, tools and languages to efforts towards improving the entire development process including project management"(Börstler et.al,1999)[3]. Thus in order to increase process quality in a company, it has to be kept in mind that the company has to give special focus to software process improvement.

There are various tools available to companies to increase process quality. Some of the existing models are (Börstler, Johansson, Orci, 1999)[3]:

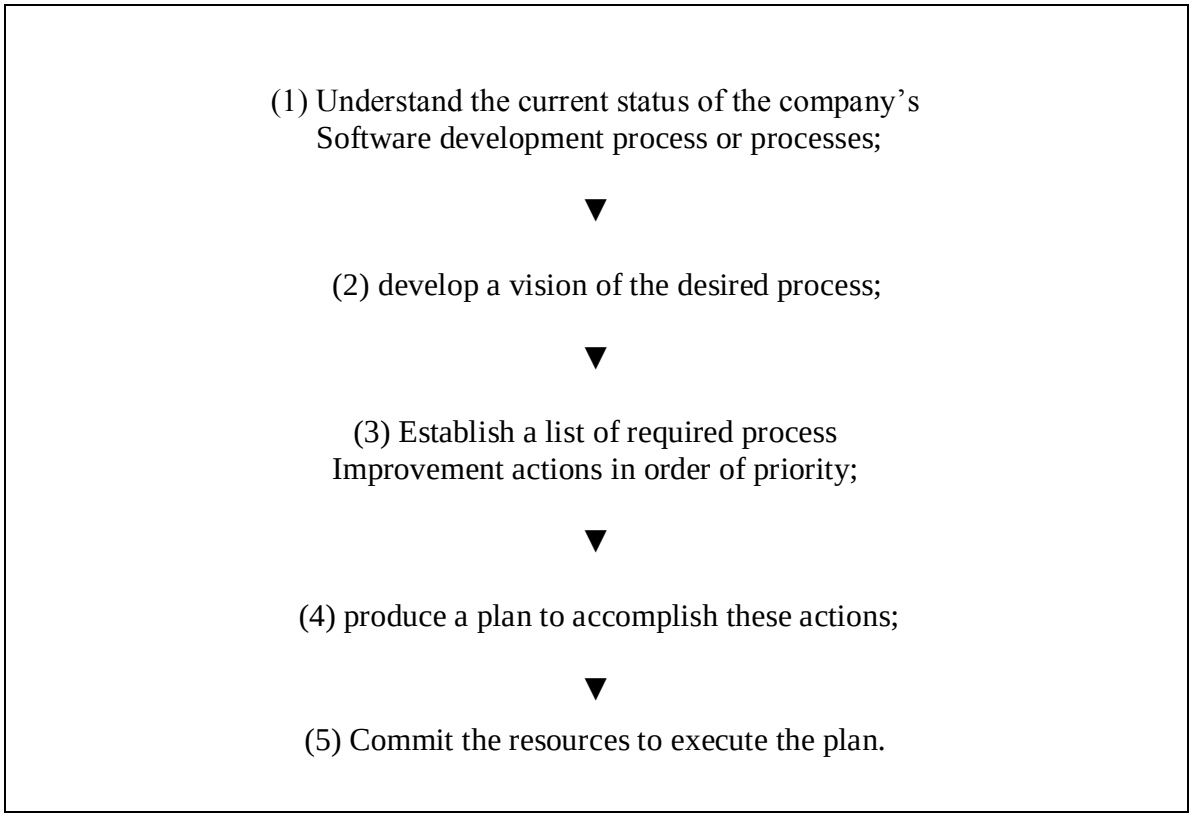
CMM – Capability Maturity Model
BOOTSTRAP
SPICE – Software Process Improvement and Capability Determination
Standards – (e.g.) ISO9000
PSP - Personal Software Process
TSP - Team Software Process
TQM – Total Quality Management
PIE – Process Improvement Experiment

The problem with the above mentioned models for process and people improvement is that they are more suitable for large and medium size companies and can prove to be costly and time consuming for small companies.

However one such technique that has proved effective in increasing process quality in organizations, especially small firms, is the 'Balanced Scorecard' (BSC) model. BSC in conjunction with the software process improvement tailored to small software development companies is an effective tool in increasing the process quality in such firms.

Software process improvement

In this era of rapid technological innovations, growth, and changes, the formula for success of a software development organization lies in the continual improvement of its development processes. The following five steps help in such process improvement:



Software Development Process Improvement (Humphrey, 1988)[4]

(1) Understand the current status of the company's processes

The first important move towards process improvement is, understanding the processes involved in the development of the product from start to finish. In other words, it is important to understand every process that goes into the Software Development Life Cycle (SDLC). It is essential to understand the process involved in the development, the software products produced in the company, the production environment, problems faced during the development process, and the characteristics of the process and product. Software process improvement has to take place at each level of this development cycle. Hence, we first examine the SDLC of software development companies.

Software Development Life Cycle (Stylusinc, 2004)[5]

All products including software products are produced for the customers based on their specifications. Even if products are developed using the best technology available, they'll not be accepted in the market if they are not customer or user friendly. They'll be taken off the market very early. Hence it is important to clearly understand the needs of the customers and develop a thorough process that can meet the requirements. The processes involved in software development:

Market Research

The first step in the development of a product is market research. First, potential customers' requirements are identified by conducting a market study. At this step of the process, customers' needs, problems faced by them with the exiting products, the scope for a new or improved product are studied carefully. This is the most important step in the development ladder as the study is done based on a lot of inputs and assumptions.

Although market research process can result in a lot of inputs and ideas, not all of them can result in a fully shaped product. Hence it is essential to clearly examine these inputs and look for ideas that can be developed into a product which will be beneficial to both the customers and the company. Eliminating unhealthy ideas is also very important, since such assumptions can waste a lot of time, effort, and money. This is especially important

for small development companies to thoroughly examine the ideas and come up with a system that will be more profitable and less resource consuming. This is done by the research and development department.

Research and Development

Once the Market study is made, the customers' need, the assumptions and the ideas are given to the Research and Development (R&D) department to come up with a cost-effective and user friendly system that would not only prove profitable for the company but also prove to be a better system than that offered by the competitors. The development of the software product begins after this research.

System Development Life Cycle Model (Stylusinc, 2004)[5]

This is also known as Classic Life Cycle Model (or) Linear Sequential Model (or) Waterfall Method. This has the following activities:

1. System/Information Engineering and Modeling
2. Software Requirements Analysis
3. Systems Analysis and Design
4. Code Generation
5. Testing
6. Maintenance

(1) System/Information Engineering and Modeling

The very first step in the development process is establishing requirements of the various systems involved in the development activity. The elements involved in the development process include hardware, people and other resources. Once the ideal system is established, the development team studies the software requirement for the system. An interface between these elements and the software product is then established.

(2) Software Requirements Analysis

This phase is also called as feasibility study. In this phase, the system analyst approaches the customer to study their requirements, analyze their system for collecting data, and looks for areas that could be automated. Specific recommendations for the software system are then furnished by the team as soon as the feasibility study is completed. The recommendation also includes the costs involved, allocation of personnel, responsibilities, project schedule, and target dates.

The essential purpose of this phase is to find the need for developing the software and to define the problem that needs to be solved. In order to come up with the software program to be built, the system analyst must clearly understand customer requirements and customer's existing system. The system analyst has to then clearly document all the findings and submit the document to the software engineer (Programmer) for developing the system design**(**Stylusinc, 2004)[5]

.

(3) System Analysis and Design

In this phase of the software development process, software engineers come up with the software's overall structure and design. "In terms of the client/server technology, the number of tiers needed for the package architecture, the database design, the data structure design etc are all defined in this phase" **(**Stylusinc, 2004**)**[5].

Analysis and Design is the most important step in the whole development cycle. Any hitch in the design phase could prove to be time consuming, expensive, and effort consuming to solve in the later stages of the software development. It is difficult to correct mistakes later in the development process and hence utmost care needs to be given for designing the best system to suit the customer. It is in this phase that logical system of the product is developed and hence given top priority and special attention.

(4) Code Generation

Once a software framework is designed, it must be translated into a machine-readable form. This process of converting the design into a machine-readable form is performed by the code generation process. Code generation can be achieved without much intricacy, if a detailed design is developed. This results in a smooth flow of the development process. Code generation process is the most time consuming of all the processes, since it is here that an idea transforms into a product.

The code can be generated by using some programming tools like Compilers, Interpreters and Debuggers. According to the type of application, the programming language is chosen for coding. Different high level programming languages like C, C++, VB, Visual Studio, and Java are used for coding**(**Stylusinc, 2004)[5].

(5) Testing

Testing of the software begins after the generation of the code. This is one of the crucial tasks in the development process which helps to solve the bugs that were committed during the previous phases. "Quality is built into the software product through a series of progressively more integrated testing" (Barnett, Raja, 1995)[2]:

* Unit testing;
* Integration testing;
* System testing;
* Acceptance testing

Some companies have their own development methods and hence they prefer to have their own testing tool. Due to this fact, they develop their own testing tool which performs the testing according to their own development methods**(**Stylusinc, 2004)[5].

(6) Maintenance

Software is bound to undergo changes after it is delivered to the customer. These changes could happen due to various reasons like some unexpected input values into the system, requirement of new applications, etc. The operations of the software are affected if there are some changes in the system. Hence, the software product should be developed in such a way that it can permit changes during the post implementation period.

Maintenance is also an important phase in the life cycle of the software product development because it is the nerve center that connects customers to the company even after the product is delivered. This improves the client-company relationship**(**Stylusinc, 2004)[5].

(2) Develop a vision of the desired process

In order to develop a clear vision of the desired process, the existing process and the problems faced with the system in these small software development companies needs to be first analyzed.

The most serious criticism of the current software development methods in these companies is that these methods are product focused rather than process focused. They also face difficulty in adapting to changing circumstances by changing their processes. They prefer to take the time tested traditional route rather than come up with new process improvements. Not much importance has been given to process improvement before a system break-down.

This is especially true with small software development companies which still fail to notice warning signals of system break-downs, as their priority is to deliver the product to the customer on time and end up with a profit. They are satisfied with the traditional processes as long as they deliver the product and make money. They wait till the last

rather than come up with process improvements since software process improvement needs continuous updating and coming up with innovative ideas.

Discovering quality problems by waiting for system failures is no longer acceptable. The critical role that software technology plays in embedded systems and other mission crucial applications does not give waiting room for system failures(Humphrey, 1988)[4].

Hence these risks cannot be reduced by addressing the problem after the software product is produced, but risk reduction must occur as part of the software development process at all the levels. "Proponents of the application of TQM techniques to software engineering hold that there is a need to shift from a product focus that introduces quality through inspection to a process focus that introduces quality into the software development activity"(Barnett, Raja, 1995)[2].

Especially, for small companies, the improvement efforts should become a part of life and culture of the organization before the processes becomes bigger and difficult to supervise and there is chaos. But, small companies fail to notice this and end up with lower quality software products(Humphrey, 1988)[4].

The importance given to process improvement efforts in companies on the European level has been acknowledged by the introduction of the ESSI (European Strategic Software Initiative) program (Barnett, Raja, 1995)[2]. The chief objective of the ESSI program is to support software improvement projects in small and medium size companies. Through this program, these companies can apply for financial support to the extent of 50% of the costs of undertaking a 'Process improvement experiment' (PIE) (Barnett, Raja, 1995)[2].

Thus the vision for these small software development firms should be to come up with process improvements that can provide long lasting solutions to the problems encountered by these development companies in their various processes. This vision should be clearly communicated to all the employees of the firm. For example, if a

company faces a high defect rate and as a result faces a lot of customer problems, the vision for process improvement should be to come up with solutions that can minimize this problem and eventually end up with higher customer satisfaction.

Vision should also include the goal the company wishes to achieve through continuous process improvements. For example, if the software company wants to go global, it should try to come up with process improvements efforts that can improve the efficiency and effectiveness of the process, reduce the cycle time, eliminate defects and thereby improve productivity(Humphrey, 1988)[4].

(3) Establish a list of required process improvement actions in order of priority

It is a wishful thinking to do an unplanned process improvement. Process improvement can be understood as the activity of elevating the performance of a process with regard to its goal. Software process improvement involves the study, understanding, improvement, and control of how software is produced. This is a continuous process that needs frequent updating and coming up with new ideas. It is also an important iterative process that should be agreed upon by all the employees, teams, and the management.

'Software Process Improvement' means learning how to work better (both efficiently and effectively) with the available resources and adapting to changing times. The idea here is to create a process or an environment where people can do a better job with greater efficiency with minimal defects. The desired goal for a company is to be able to develop products that are less expensive with better quality, and that takes less time do deliver. Such process improvements should create a short learning curve for employees.

The first important detail that organizations should take care of before making process improvement decisions is to highlight the importance of good communication among the

employees. Good communication includes good articulation, clear and undisrupted communication channels, and active listening. This is the most important attribute of success in any process improvement exercise(Humphrey, 1988)[4].

For example, software developers who are assigned to the development of a product should first have a discussion about the requirements, resources, and responsibilities at the beginning of the product cycle. It is very important that they designate someone to ensure that the discussions actually take place. An appropriate process is agreeing on the conclusion of such discussions, ensuring that no one is excluded from the discussion, and clearly assigning responsibilities and target dates to the engineers. This should also be communicated to all the parties involved in the development process. This meeting should establish the following(PS&J Software Six Sigma,2005)[6]

- clearly defined roles and responsibilities of the engineers involved
- tasks to be completed and the target dates
- required resources and facilities
- anticipated costs
- Realistic budget of resources
- expected returns
- cost/benefit analysis
- detailed plan
- Risk identification
- Improvement targets for the quarter

This avoids future confusions and differences and improves the communication channel. Since such a discussion takes place at the very beginning of the software development process, ambiguity in the assumptions can be avoided. This also avoids conflicts regarding responsibilities. Since resource requirements are clearly evaluated at this stage,

future scarcity of resources can be avoided. Thus unnecessary delays are effectively reduced by introducing such a process improvement at the very beginning of the project.

It is also important that such small software development companies consider their processes as a goal- and risk-driven process. Understanding goals to be achieved and the risks involved goes a long way towards building a good process that can balance the need and reward situation. This can effectively direct difficult work to success.

For example, a group rushing to develop a new product in order to obtain funding from venture capitalists will have a different set of goals and risks different from the group which develops version two of an existing software. Once such goals are clearly set at the beginning of the product development life cycle, software developers can easily direct their efforts towards meeting the goals within the scheduled target dates.

The next important task in the development process is making sure various departments in the company share their plans and proposals. For example, personnel from various departments namely, marketing, professional services, engineering, finance and so forth should share their project plan with each other to ensure that everyone in the company understands what the company is trying to accomplish with upcoming releases and get themselves familiarized with the product and target dates. This helps to orient employee and company goals.

With the above primary features established, the company should start focusing their efforts on process improvements addressing problems and goals in the order of priority. For example, Small software development companies face a lot of problems in the development process that results in risks, such as, schedule slip, late delivery of products, inadequate product performance, cost overruns, a short product life, and failure to continue post implementation maintenance of the product.

These management risks must be controlled in order for them to sustain in the market. It

is also essential for the companies to eliminate these risks along with continuing 'the fitness for use' (Barnett, Raja, 1995)[2] concept to achieve quality software. "Fitness for use" was the traditional definition of software quality. A software product is considered to be fit for use by customers if its performance is up to their expectation to some extent, and is user friendly.

The above mentioned risks could be caused at various stages in the software development. For example, the software analyst could take more time in understanding customer requirements or the software engineer could spend hours together for coding the software which could all result in schedule slips. The other risk of inadequate product performance or software bugs could be a result of inadequate testing. Since small companies cannot afford more testing staff or more resources for rigorous testing, the serious problem of defects could arise. Thus all these problems result in customer dissatisfaction. Hence the process improvement effort should prioritize these issues and address them in the order of precedence.

(4) Produce a plan to accomplish these actions

Once areas in the development process that needs improvement is identified and the risks and goals sorted based on their priority, a plan has to be drafted to get the process improvement program started and going. For example, if the company faces a lot of defects which results in various other problems like late delivery of the product to the customer, customer complaints, and customer dissatisfaction, it is a high priority issue that can plunge the company deep down the hill. This needs immediate attention. The process improvement effort should address this issue first.

Since small software development companies face serious time crunch, there is always hurry in doing things. This results in high defect rate. However, such defects can be identified to a great extent by performing thorough testing of the software. The management has to make sure that engineers at each level of the process do their job efficiently and effectively. An audit of the job performance becomes necessary.

Hence it is essential that the company has more resources to the testing of the product before it reaches the customer. Testing of the code by the software engineer who writes the code before it goes to the testing team should also be insisted, so that errors can be found out at a much earlier stage without much wastage of time and effort. Code review by the peers should become a process by itself. Thus, process improvement effort should have a plan of action for each of the problems and goals of the company(Humphrey, 1988)[4].

(5) Commit the resources to execute the plan

The last important step in the process improvement program is to commit the required resources to execute the plan of action. In the above example, the company should hire more testing staff to do rigorous testing of the software. The company should also insist that all the engineers document their process. This is especially important for the testing engineers who need to document their testing queries and tools. This can reduce future time spent on coming up with new testing queries and it also proves to be a good log of already encountered errors or bugs(Humphrey, 1988)[4].

Process improvement and quality

Process improvement and quality are related to each other. Process improvement is one of the means for creating products with high quality. It helps in generating new improvements to the existing processes so that the final output meets the specification of the customer during changing times. Quality is essentially a relation between the customer and a product or service. Does the product or service fulfill the requirements and specifications from the customer and is the quality of the product or service satisfying enough to keep the customer coming back? Process improvement on the other hand wants to establish processes where the employees can do a better job with the existing resources.

In other words, process improvements enable employees to develop products with sufficient quality with the help of quality processes. It can be rightly said that, the quality of a software product depends to a great extent on the quality of the process used to develop and maintain it.

Thus, the relationship between the development process and that of the process outcome can be described as:

Quality (Process) ► Quality (Outcome)

(Leads to)

However, it should be kept in mind that the same degree of change to a process does not always lead to a similar percentage change in outcome. The relationship depends on various factors like experience, expertise, fad, environmental instability, organizational size etc.

Measuring results of process improvement

Once process improvement effort begins, it is essential to measure the impact of these process changes on the products generated, most importantly on the quality of the products produced. It is important to determine which technologies or process improvement tools are beneficial and appropriate to the particular environment and, more importantly, how these technologies (or processes) must be refined to best match the development process of the company.

Thus performance measurement of such process improvements is very important to help these small software companies to evaluate and improve their performance. It is essential that such performance measures increase the scope of the measurement focus beyond financial reporting to less monetary benefits like focusing on quality, service and speed, creating a value, and bringing about organizational learning, growth and innovation.

One useful tool that helps companies, especially small and medium based companies, to measure the impact of process improvement on the success of such companies is the balanced scorecard, developed by Robert Kaplan and David Norton. According to them, "The balanced scorecard provides executives with a comprehensive framework that translates a company's vision and strategy into a coherent set of performance measures." (Willcocks and Graeser, 2005) [7]

Balanced Scorecard - a process improvement measuring tool

"The balanced scorecard is a management system (not only a measurement system) that enables organizations to clarify their vision and strategy and translate them into action. It provides feedback about both the internal business processes and external outcomes in order to continuously improve strategic performance and results. When fully deployed, the balanced scorecard transforms strategic planning from an academic exercise into the nerve center of an enterprise"(Arveson, 1998)[8]

The scorecard is a useful tool that helps in translating a company's goals, visions, and high-level business strategies into concise, specific, and quantifiable goals. The analysis takes into account financial and non-financial measures, internal processes, past performance outcomes based on customer satisfaction and customer retention, and current requirements as indications of future performance. Any activity undertaken by the company should be part of a link of cause-and- effect relationships, which forms a loop between the key perspectives (explained below) ending in financial outcomes that ultimately affects the growth of the organization.

It is an analytical tool for monitoring and measuring performance against the company goals. "The methodology breaks high-level strategies into objectives, measurements, targets and initiatives" (Anthes, 2003)[9]. Another advantage of this balanced scorecard approach is that it retains thousands of detailed information for frontline supervisors but gives top management only a snap short of information relating to key drivers.

The Balanced Scorecard Approach

Balanced Scorecard technique was developed by Robert S. Kaplan and David P. Norton to measure performance. This approach combined financial measures, which were originally the performance evaluation criteria, together with intangible (non-financial) drivers to provide more specific and relevant information to managers about the tasks they are managing (Epstein, Rejc, 2005)[10].

The balanced scorecard technique helps management to communicate, execute, and monitor strategy using four perspectives: the learning and growth perspective, financial perspective, internal processes perspective, and customer perspective (Epstein, Rejc, 2005)[10]. The idea here is to strike a balance among these key perspectives in strategic planning and in selecting a plan of action to accomplish the goals of the organization.

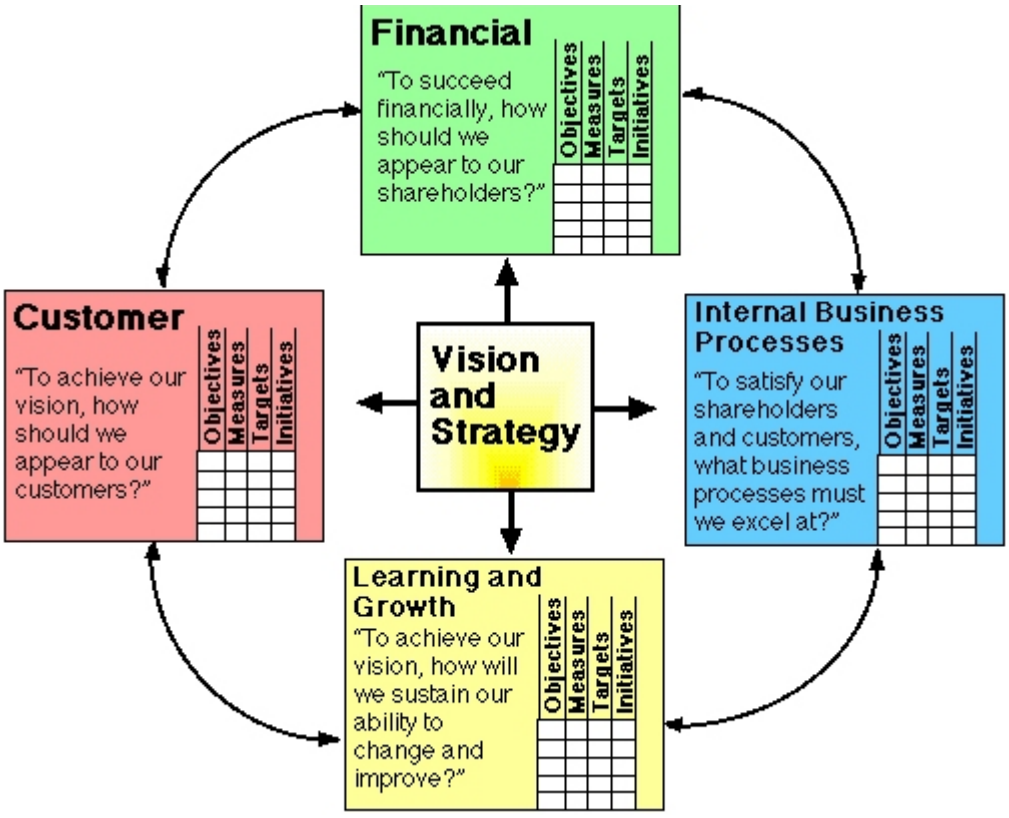


Figure 1 view of four key perspectives of Balanced Scorecard(Arveson, 1998)[8]

The four key perspectives

· **Financial perspective**: relates to profitability – measured by return on investment, profitability index etc

· **Customer perspective**: measures successful outcomes of company strategies resulting in customer satisfaction, customer retention, and increased market share. Customer includes both internal (employees, staff, and others) and external (clients, venture capitalists, and others) parties involved with the company.

· **Internal processes**: includes processes followed within the organization which aids in the development of a product(Arveson, 1998)[8].

· **Learning and growth**: relates to the continuous learning through research and development efforts that results in long-term growth and improvement for the firm.

Objectives, Measures, Targets, and Initiatives under the approach

For each of the key perspectives under the balanced scorecard approach, the following four things are monitored (scored) [12]:

- **Objectives**: major goals or objectives to be achieved (for e.g. improving process and product quality, profitable growth, entering new business line, etc.)

- **Measures**: the observable factor which is used for measuring progress made towards accomplishing the objective. For example, the objective to improve process quality can be measured by measuring the decrease in cycle time, decrease in defects rate etc.

- **Targets**: the precise target values for the measures, for e.g. 5% reduction in defect rate annually.

- **Initiatives**: projects or tasks to be initiated to accomplish the objective.

Objectives and drivers under Balanced Scorecard

The first step under Balanced scorecard approach is to set specific objectives for each of the four perspectives (Epstein, Rejc, 2005)[10]. These objectives need to address important issues in the software development process and should be aligned with the overall business strategy. For example, if a small software development company's overall business strategy is to improve process quality and thereby increase product quality, each of the four perspectives should concentrate on process quality improvement as their primary objective.

Once these objectives are identified for all the four perspectives, the company must determine the drivers of success. Determination of the critical drivers is the key to success. Drivers specify the actions to be taken by managers to improve the success of the development activities which will ultimately result in the overall success of the organization.

"A well-defined scorecard should contain a good mix of outcome measures (or long-term targets) along with performance drivers to track the progress in the short term" (Sarkar, 2003)[11]. Outcome measures like employee productivity, customer satisfaction, customer retention, etc which are lagging indicators are general to all companies. In contrast, performance drivers are company-specific (Sarkar, 2003)[11]. These drivers measure the effectiveness of the company's overall business strategy.

Importance of the key perspectives

Balanced scorecard serves as an important tool for improving process quality through its measurement metrics which are based on the four perspectives that inter connect to form a loop. A software development company's success, especially that of a small software company, is dependent to a great extent on the following learning and growth elements, namely appropriate and adequate resources which includes finance, people etc, suitable and timely corporate activities like training, information sharing, performance measurement, and incentive systems, and organizational culture and environment.

Information technology learning and growth improves internal business processes by bringing about new and improved changes to these processes. "IT learning and growth affects IT internal processes, such as standardization; integration and consolidation; security; and overall quality of IT processes, products, and services" (Epstein, Rejc, 2005)[10]. These learning and growth elements help the organization to adapt itself to the changing environment. With new research and development activities, these companies make the necessary changes in their development processes so as to be able to overcome bottlenecks and improve process quality.

These improved internal business processes perk up customer (both internal and external customers) satisfaction, which ultimately leads to improved financial performance. Internal customers are the employees of the firm. Internal customers' satisfaction is reflected in their increased productivity, coming up with creative and innovative ideas, and brings about increase in quality of work performed. External customers are third parties associated with the company. External customers' satisfaction will get reflected in higher loyalty, customer retention, new customer acquisitions, and more sales.

Thus customer satisfaction and improved development process leads to higher revenue and cost reduction for these small software development companies. The financial drivers in the balanced scorecard approach keeps score of the progress/outcome of the

improvement techniques undertaken in the company by measuring revenue growth, return on investment etc. These drivers help companies to improve their process and product quality by setting specific and accurate objectives for the key drivers.

These perspectives thus allow the monitoring of present performance and also collect and analyze information about how well the organization is positioned to perform in the future.

Measuring performance under BSC approach

Strategic feedback and learning process are important measuring tools under the BSC approach. These help organizations in evaluating the success of the new initiatives in the organization (Sarkar, 2003)[11]. It helps management to understand the effects of new initiatives on the overall objectives of the company and helps organizations to adapt themselves to emerging conditions in the business environment. A properly developed and implemented performance measurement system improves productivity by focusing attention on the most important issues, tasks, and objectives of the organization.

Utility of Balanced scorecard to small software development companies

Small development companies can use the 'Balanced scorecard' approach to evaluate the impact of the company's new activities (e.g., making a process change, entering a new market) on the existing goals of the company. These companies can also use this approach to track performance of the key perspectives with the introduction of the new strategies.

For example, if the software development department in the company moves its existing platform to a different operation system, it can use scorecard to analyze the effect of this

change and document the same. For instance, such a change could affect the budget requirement which is taken care of by the financial perspective. It could increase training requirements of the personnel working on the system which gets reflected in the learning and growth perspective.

Such a change could also bring about great improvements in the software development process which gets evaluated by the internal process perspective. Finally if this improvement results in an introduction of a user friendly system, it will result in customer satisfaction, it will be tracked by the customer perspective. Ultimately financial perspective analyzes the resultant profit compared with the cost involved for the company.

Thus the benefits of the Balanced Scorecard are[12]:

- Focusing the activities of the entire organization towards the few key things that can result in breakthrough performance.
- Helps to coordinates all the corporate programs such as initiating process improvement measures, improving quality, re-engineering, and customer service measures.
- Breaking down complex strategic goals into lower level objectives in order to help lower and middle management understand their role in the organization and the responsibility at their level to achieve excellent overall performance.

Balanced scorecard and quality

The advantage of using the 'Balanced scorecard' approach is that, if firms record and measure software development processes based on the key perspectives, which measure resources allocated and the outcome achieved, they will have the necessary information and data to identify the processes that actually increase both financial and non-financial outcomes and those processes that do not. This will help the firm to identify avenues that require increased investment and those to receive disinvestment.

Balanced scorecard can help companies to evaluate proposals by looking into ideas that balance the four perspectives. It helps them to identify the idea that is beneficial to the company. For example, by using the balanced scorecard approach, the company can evaluate the various investment avenues in the company.

If under the approach, it is best for the company to invest its financial resources in new, improved, and sophisticated technology, skills, and knowledge, rather than investing in a new business line, the company shall benefit by getting an edge over its competitors by coming up with new products and services with high quality. This will lead to opening doors for new market segments, acquiring new customers**,** retaining and satisfying the existing ones, and, consequently, increasing sales, revenue, and profits.

This also results in establishing fair incentive schemes for employees which increases employee satisfaction. This reduces attrition rates and ultimately increases quality of the services and products. Finally it leads to improved software development process and reduced costs (Epstein, Rejc, 2005)[10].

Combination of Software process improvement and BSC – aid to increase process quality

As it can be seen from the above discussions, software process improvement and balanced scorecard go in the same direction to ultimately aid a company increase its process quality and thereby its product / service quality.

One of the missions under the balanced scorecard approach is to deliver high quality products and services with the help of high quality internal processes. This mission is approached by establishing accurate objectives, such as (Sarkar, 2003)[11]:

- High quality software development process to be established
- Reduce development life cycle time involved
- Introduce continuous process improvement programs, such as improving the quality of inputs to the system, improving communication channels, etc.
- "Introduce reengineering and transformation programs such as introduction of new software packages, major application redesign, etc" (Sarkar, 2003)[11].

These objectives under the BSC approach are especially important for small software development companies as availability of resources to meet the demand is the primary concern of such companies. By accomplishing the objectives set, such small companies can deliver high quality products with the existing available resources.

It is to be noted here that small firms can easily implement the 'Balanced scorecard' approach for increasing process quality. With respect to understanding the key drivers in the customer perspective, small firms are at a more advantageous position compared to bigger firms. Employees of small firms have direct contact with customers and hence it becomes easy for them to understand customer requirements.

Larger firms might take more time in collecting data regarding customer requirements. Information collected could also be lost in the process of getting communicated from the

customer to the engineer who works in the development process. On the other hand, customer requirements are directly collected by engineers working on the product development in small firms. This helps these firms to easily bring new and improved changes to the existing processes in order to satisfy the changing needs of customers and also deliver quality products to their customers.

Thus it can be clearly seen that BSC and process improvement activities go hand in hand for increasing process quality in such small software development companies. The implementation of the 'Balanced scorecard' approach thus focuses mainly on continuous process improvements with importance given to feedback and learning mechanisms and customer satisfaction surveys. It provides specific metrics that can be established as benchmarks and this can be used as selling points in contract negotiations when approaching new customers.

Conclusion

The consequence of poor software quality has been felt tremendously in the present times due to the increased importance given to information technology in the operations of modern organization. Hence, it is no longer acceptable to evaluate the quality of the product at the very end of the software development life cycle. The solution proposed is to introduce quality into the process of developing software at every stage of the development process. This helps in identifying flaws at the beginning of the development process rather than wait until the last moment.

In order to sustain quality in the development process, the company needs to frequently make process improvement efforts. In order for these efforts to be successful, a clear and accurate measuring tool needs to be established. The Balanced score card is one such tool that helps organizations, especially small firms, establish metrics that can accurately evaluate activities taking place in the organization.

This paper has thrown light into the usefulness of tools like continuous process improvement and balanced scorecard to small software development companies. It has been clearly felt that software process improvement together with balanced scorecard benefits small software development companies to a great extent by increasing process quality and ultimately improving of quality software products developed by these companies.

References

1. Dr. Richardson Ita, "Software process improvement in small indigenous software development companies", 2000 available on internet http://www.irishscientist.ie/2000/contents.asp?contentxml=230as.xml&contentxsl=insight3.xsl, accessed on 16th December at 8 p.m.

2. Barnett D. William and M.K Raja, "Application of QFD to the software development process. (Quality function deployment)(Quality Function Deployment: Theory and Practice)", International Journal of Quality & Reliability Management, June 1995, accessed on 17th December at 8 a.m.

3. Börstler Jürgen, Johansson Olof, Orci Terttu, "Quality Management for Small Enterprises", 1999 available on internet http://www.cs.umu.se/~jubo/Projects/QMSE/application.html, accessed on 18th December at 9 p.m.

4. W.S. Humphrey, "Characterizing the Software Process: a maturity framework", available on internet www.cs.ucl.ac.uk/staff/A.Finkelstein/advmsc/8.pdf, accessed on 20th December at 10 a.m.

5. Stylusinc, "Software Development Life Cycle (SDLC)", 2004 available on internet http://stylusinc.com/Common/Concerns/SoftwareDevtPhilosophy.php accessed on 22nd December at 8.30 p.m.

6.PS&J Software Six Sigma, "Measurable, Managed Process Improvement Structured to Meet Business Goals", 2005 available on internet http://www.softwaresixsigma.com/ImpPlan_Main_Workshop.htm accessed on 24th December at 9 p.m.

7. Willcocks Leslie and Graeser Valerie, 2005 “Delivering IT and e-business value”, Computer Weekly, May 17, p30.

8. Arveson Paul, "What is the Balanced Scorecard?” 1988 available on internet http://www.balancedscorecard.org/basics/bsc1.html, accessed on 24th December at 10 p.m.

9. Anthes H. Gary, “ROI Guide: Balanced Scorecard”, Computer world, February 17, 2003 accessed on 25th December at 11 p.m.

10. Epstein J. Marc, Rejc Adriana, “How to measure and improve the value of IT: a balanced scorecard geared toward information technology issues can help you start the process. (Performance Measurement)”, Strategic Finance, 2005 p35 (7).

11. Sarkar Pushpak, "Applying the Balanced Scorecard in the IT Organization", DM Review Magazine, 2003 available on internet http://www.dmreview.com/article_sub.cfm?articleId=7762 accessed on 26th December at 10 a.m.

12. 2manage, “Balanced Scorecard”, 2006 available on internet http://www.12manage.com/methods_balancedscorecard.html accessed on 28th December at 8 p.m.

13. CIMS, Quality Assurance, 2002, available on internet http://www.chesapeakebay.net/info/qa_glossary.cfm accessed on 29th December at 9 p.m.