Disciplined Agility as a Strategic Enabler Capability for Facilitating Internal Organizational Changes Using the Synergy between SEI-TSP/PSP and Agile Methodologies

Ahmed S. Elsheikh[(1)] and Tamer Salah[(2)]

(1) Software Engineering Competence Center-SECC, ITIDA, Smart Village (Egypt)
E-mail: aselshikh@itida.gov.eg
(2) MiniMax Software, Ali Mobarak St., Mansoura (Egypt)
E-mail: tsalah@minimax-soft.com

ABSTRACT

The synergy between the disciplined (i.e. plan-driven) and agile (i.e. adaptive) approaches can achieve results that exceed any one of them separately in software development. In this paper, we focus on how introducing the disciplined agility approach called “SEI-Team Software Process” in MiniMax Software helped the company deals with the transitions between different stages in the internal learning and development change. Major stages, their discovered challenges, the needed methodological shifts and achieved breakthrough improvements in each stage will be explained. This development cycle can be summarized into three major stages. Summary of achievements per stage will be presented as well as a cross-stages analysis to illustrate the impact of these transitions and changes.

Keywords: Agile Methodologies, Disciplined Agility, Internal Organizational Change, SEI-Personal software Process (PSP), SEI-Team Software Process (TSP), Strategic Enabler Capability.

1- INTRODUCTION

Being agile means you are able to respond to the change or more accurately embracing the change. When we mention the change, the attention goes to the external changes which are usually the changes in the marketplace or customer’s preferences. However, the more important change is the internal change. We can consider any organization as a life body that has a lifecycle. During this lifecycle, many changes happen from the interaction between internal elements or form the interaction between organizational elements and the external environment. One of the most important change elements is the continuous development of the technical capabilities.

As an example, in the beginning, the work is conducted in an ad-hoc manner without guidance of any process or framework. This introduces the need to have a certain process or framework to improve the "effectiveness" of the work being done. This means doing the work correctly. Of course, this takes time to help individuals and team go through the typical learning curve. After

that the focus is shifted towards the need to improve the "efficiency". This means doing the work with less resources and lower levels of waste.

This highlights the need for an important skill, which is the ability to understand this learning cycle as well as detecting these changes when they happen and taking the required actions to help the transition happens smoothly. Each stage comes with its own benefits as well as its own challenges and problems. It's similar to the onion. It is a must to remove the outer layer to be able to touch the successive inner one. Hence, the capability of observing the changes in the internal capabilities is a critical success factors and more accurately as a strategic capability. Although skillful managers can detect these internal changes and take the suitable corrective actions intuitively, this can be a risky approach. These highlights the need for a comprehensive measurements system that can help the management take informed decisions. This paper explains the journey of improving the effectiveness and efficiency within MiniMax Software as a direct result of the provided SECC consultancy services to MiniMax during the past seven years. It highlights the major stages during company development in managing four major aspects in software development projects. These major aspects are:

- The ability to manage the deviations in the total expected release time (i.e. the Time to Market), which is the typical sum of the deviations in all interim iterations
- The ability to manage the process dynamics to improve both the effectiveness and the efficiency of the technical activities
- The quality of the final delivered product, in each release, in terms of post release defects and the underlining internal dynamics and costs to achieve these results. This includes the percentage of defects found before and during testing as well as efforts distributions
- Team dynamics will be discussed form two perspectives. The first one is how the top management were able to manage teams and the second one is the internal interaction between team members and its impact on the overall performance

It is not intended for this paper to either provide a full analysis of all metrics and indicators that explains the dynamics of these aspects or benchmark the performance of the company against any external data. The major contribution of this paper is to focus on the dynamics of internal changes that happened in the company and how these changes were handled including the challenges, pain areas and experienced learning curves. Furthermore, it is going to emphasize the role of the disciplined agility methodology, called SEI-Team Software Process, as a strategic enabler capability for discovering these challenges/pain areas as well as providing the required information to analyze the situation and supporting informed decisions.

2- SOFTWARE PROJECT MANAGEMENT METHODOLOGIES

Before diving deep into the details of this research and the presentation and discussion of the achieved results and the gained experience, it worth having a short discussion about the current known cornucopia of project management approaches and methodologies that can be used to manage software development projects. It is beyond the scope of this paper to explain any single method in details or to provide a detailed comparison between these methods. However, we are going to provide a brief description for the following few aspects:

- The shifted focus in software industry towards the focus on software project management approach away from the development technologies themselves
- A simple classification that can provide an easy to use positioning approach for each of these well-known methodologies
- The basic characteristics of disciplined (or plan-driven) approaches
- The basic characteristics of Agile (or adaptive) approaches
- The new wave of “Disciplined Agility” mindset
- The basics of the “SEI-Team Software Process” & “SEI-Personal Software Process” as two of the most famous and well-established disciplined agility methods

The following subsections provide this information in sequence:

2-1 THE IMPORTANCE OF A SUITABLE PROJECT MANAGEMENT METHODOLOGY

Around a decade ago, there was a study [1] that tried to understand the impact of the change in software development technologies on the project’s success rate.

The study examined data from 50,000 projects, which were:

- From all over the world
- Conducted across twelve years interval
- Have different sizes and complexity
- Conducted with different technologies

Fig. 1 presents the study results which indicate the percentage of the each category of projects (*successful projects in green, challenging projects in yellow and failed projects in red*). Results were grouped in intervals which equal to two years each.

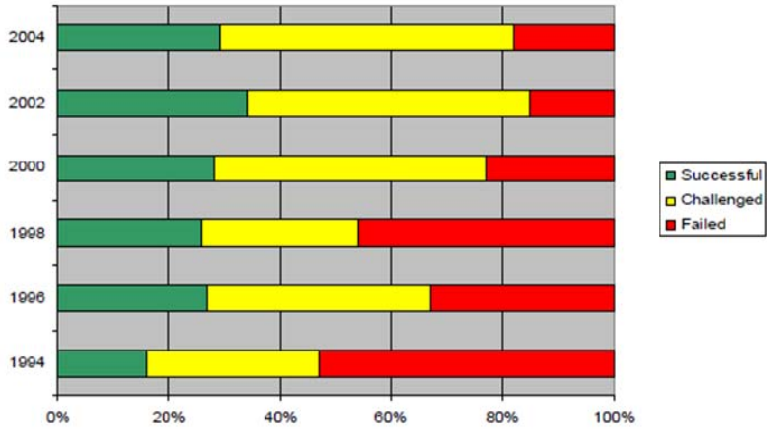

Figure 1. Typical Project Success/Failure Rates

The result was amazing. The success rate for the period of 2002 to 2004 (i.e. the last interval) was less than 30%, while the success rate for the period of 1992 to 1994 (i.e. the first interval) was less than 20%. This means only 10% increase in the percentage of successful projects from the overall sample size. This shows a clear shocking fact that the number of unsuccessful software development projects is still more than 70% even with the dramatic changes that happened to the software development technologies. This emphasized the urgency of shifting the focus from the development technologies themselves towards establishing good and success-proven methods for managing the software development projects. The critical success factors are in the management side more than in the technical one.

2-2 CLASSIFICATIONS OF COMMON APPROACHES

To manage different types of projects within different environments, there are a wide spectrum of methodologies from the extreme deterministic approaches that focus on a great deal of planning and control (i.e. extreme disciplined ones) to the other extreme that focuses on embracing changing, being fixable and depending on the emergent orders (i.e. extreme agile ones). A comprehensive list for these methodologies can be found in [2].

To facilitate the task of selecting a suitable methodology for each project, there were many attempts to classify each methodology according to certain factors. Fig. 2 shows a good example for these classifications based on the suitability to embrace high rate of change and the iterative/participative nature [3].

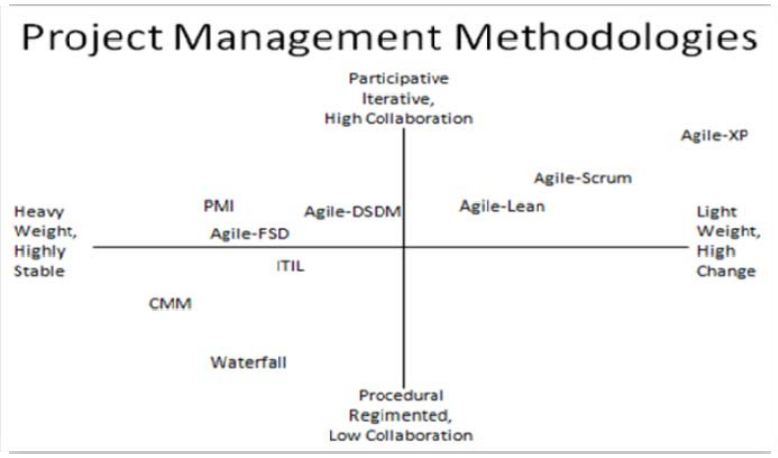

Figure 2. Project Management Methodologies Landscape

According to the previous classification, and when considering the year of origin of each one of the above methods and approaches, you can see that the need to move towards more agile approaches increased due to the faster pace of market dynamics and the need for higher level of collaboration between team members to support innovation.

2-3 DISCIPLINED APPROACHES

Among the well-established disciplined approaches are the PMI project management methodology [4] and any project management approaches that were inspired by the original mindset of the SW-CMM [5] and CMMI for Development [6]. In more IT oriented contexts, the ITIL [7] inspired quality management systems fall in the same category. The “Waterfall” is not a standard, framework or method in itself. More or less, it is just a way of organizing the project or what we call the project life cycle [8].

In this category, you can find a great emphasize on the detailed planning for the whole projects in advance. These approaches depend on a basic assumption that the next project will have similar conditions such as the previous ones.

Furthermore, they require a great amount of historical data and long experience from the project manager to be able to plan and control the execution of project and the team in charge. Intuitively, although this can be suitable for projects in stable environments such as civil engineering, but it cannot be the same for highly dynamic environments and innovation-driven environments like software development projects. Large amounts of historical data and long experience in similar conditions are no longer critical factors.

2-4 AGILE APPROACHES

Usually, agile methodologies like Scrum [9] focus on team dynamics, embracing the changes to the last minute and delivering in short iterations (called sprints in Scrum). Delivering in short sprints will facilitate learning from the feedback, while embracing changes to last minute is a typical practice in development projects as new knowledge about the product is acquired and more understanding of what can be achieved happens. Fig. 2 showed other methods as well as many combinations of two methods.

2-5 THE DISPLINED AGILITY WAVE

The synergy between the disciplined approaches and the agile ones is not a new concept. It was discussed comprehensively more than a decade ago by Boehm & Turner [10]. The idea behind this “Disciplined Agility” approaches is simple. Instead of the competition between these two schools of thinking, why not cooperating together? By combining the two mindsets, the advantages of both can be achieved and the synergy between them can produce further levels of value proposition. Furthermore, this synergy aims to reduce the impact of the disadvantages of both. Boehm & Turner stated, in the time of their publication, some important observations about the future of these approaches that can be summarized in the following points:

- There is no silver bullet(s)
- Agile and plan-driven approaches are designed originally assuming extreme conditions, which rarely appear in real projects
- Both schools will be needed in the future
- Balanced methods will emerge in the future
- It is recommended to build your own method instead of tailoring the already existing ones
- The focus should be more on the people and the value proposition instead of following any method blindly

Four years ago, there was a clearer trigger that raised the awareness about the urgent need for balancing the agility with the discipline. A worldwide “Agile Retrospective” event [11] was conducted by the original authors of the agile manifesto [12] to evaluate a whole decade of agile practicing. The result was a list of ten impediments towards achieving agility excellence [13]. Five out of the ten were identified as the most critical from the practical point of view [14], which are:

- Lack of a ready backlog, which cause delays (usually 10% only of user stories are ready for implementation)
- Lack of being “done” at the end of each iteration (some items are deferred or skipped near the end to close the iteration)

- Lack of teamwork efficiency (problems are under the surface till discovered accidently or lately)
- Tolerating defects (usually near the end of each iteration and deferring them to the backlog of the upcoming iterations as technical debt, which make the quality and the cost of obtaining it totally uncontrollable)
- Finally, lack of good design (either for the whole product or for each piece of code or a single user story)

These impediments can be addressed by adopting a well-established disciplined agility methodology [11, 14] called SEI-TSP/PSP [15, 16]. The next subsection gives overview for it.

2-6 SEI-TSP/PSP OVERVIEW

The SEI-Team Software Process (TSP) [15] and SEI-Personal Software Process (PSP) [16] are well-established and known methodologies designed and managed by the SEI-Software Engineering Institute [17]. To fully understand how these methodologies are different from the other similarly well-known methodologies in the industry, it is helpful to look in the following classification [18] showed in Fig. 3. It shows the logical and technical position of the TSP/PSP (mentioned shortly as TSP in the graph). This article classified all methodologies into three sub-domains of improvements, which are the “What To Improve”, the “How To Improve” and the “How Well Is The Improvement”.

TSP/PSP is in the “How” domain as it provides detailed operational descriptions of how to conduct the technical activities such as software inspection. More importantly is the close relations that appear between the TSP and CMMI-DEV and Agile approaches as shown through the triangle of arrows. This is exactly the heart of this study as MiniMax was already CMMI ML2 accredited and was trying to improve its performance dynamics and quality with TSP as well as raising its agility.

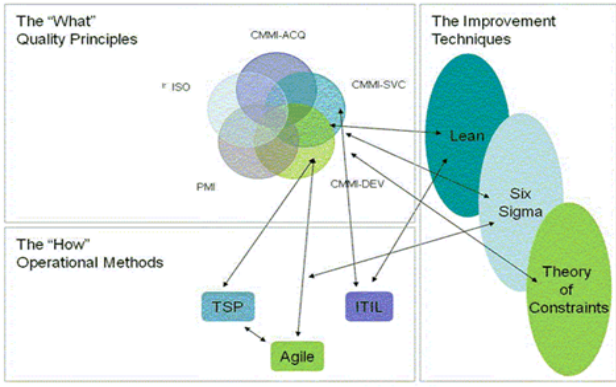

Figure 3. TSP Position in the Quality Approaches

Previously conducted researches on the resulted of TSP/PSP adoption had showed breakthrough improvements in both the process dependability [19] and delivered product quality [20] as shown in Fig. 4. The values are the average number of defects per KLOC in the typical software development projects in companies that are accredited in certain CMMI maturity level. On the extreme right of the graph, you can see how TSP can achieve a lower number of defects than ML5 (i.e. a higher level of quality).

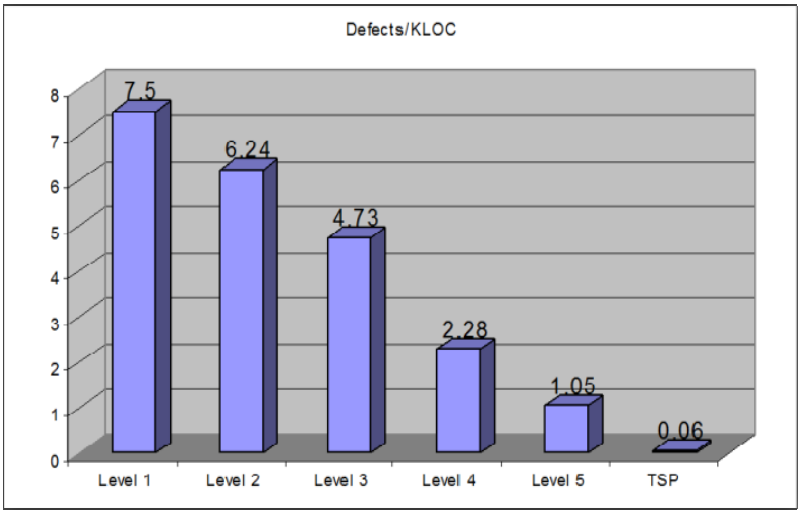

Figure 4. TSP Quality Levels Exceeds CMMI ML5

3- THE EXPREEMNTAL CONTEXT

Minimax is software house that has been established in 2005 in Egypt. Its business sector can be classified vertically by the specialization in developing Web Applications and Web Portals and horizontally by using the portfolio of Microsoft platforms/technologies. According the classification of the Egyptian market place, it is a small sized company with a technical team of that varies from 18 to 25 engineers, other than the support staff.

For about 10 years, MiniMax applied most recent software development technologies and proven software quality models and methodology in order to provide superior levels of software services to its clients. As a result, currently, its customer base had been expanded to cover Egypt and the Gulf area.

From the perspective of this paper, the technical journey in the field of process improvement and quality started in 2008 and was supported by SECC consultations. In 2011, Minimax archived CMMI maturity level two. Continuing its process improvement program, Minimax decided to apply TSP/PSP techniques in 2012 to resolve project management, measurement and product quality challenges. Results of TSP/PSP encouraged senior management to continue in the journey to achieve higher performance levels.

3-1 OVERVIEW

For more than seven years, it is very hard to claim that the journey was planned or that its end results were a part of clear vision from day one. However, it is fair to assume that the theme of the presented results as an output from the whole journey is exactly the same approach followed during the journey.

In other words, what we are trying to say is that there was a balanced mix between disciplined periods (i.e. when short term plans were in place to guide simple improvement initiatives) and the agile/adaptive periods (i.e. when retrospectives were conducted to reflect on what had been achieved, analyze the situation and determine the next required actions to overcome the latest challenges) that helped us adjust the long term overall direction. Hence, the followed approach used to introduce the “Disciplined Agility” capability into the organization was the “Disciplined Agility” approach itself.

The results presented and discussed in this paper cover a long journey that was started in 2008 and cover up to 2014. The following subsections gives overview for the followed techniques and more importantly the faced constrains that affected the available choices and the motivations behind the whole journey. Changes in these areas are mentioned too.

3-2 MOTIVATIONS & CONSTRAINS

This followed approach was a direct result of the surrounding constrains as well as the existing motivations during the journey. On the constrains part, the most powerful constrains was the knowledge and experience of SECC consultant himself, the available funding and finally the company purchasing power even towards SECC subsidized services.

SECC consultancy services were totally oriented towards the disciplined plan driven approach which was the mainstream direction in the beginning. Later one, the knowledge and experience were available regarding agile approaches as well as the synergy between these disciplined and agile approaches.

On the other hand, the cost of receiving the services related to the disciplined agility approaches was too high for a small company like MiniMax even after this cost was subsidized by around 90%. Due to some innovative approaches1, the cost of acquiring the service was reduced dramatically which enabled the company to request the service.

4- ACHIEVED RESULTS SUMMARY

To summarize the achieved results, we need to summarize the major stages

[1] Outside the scope of this paper

or periods as well as the sub-stages or major phases. There were three major stages during the company development lifecycle, which are the “Ad-hoc Stage”, the “Disciplined Stage” and the “Disciplined Agility Stage”. Briefly, these stages can be explained as:

- The ad-hoc stage that represents the period of time when the company was developing software without following any predefined approach or process. Of course, and although there were many delivered software systems during this stage but there was no clear idea about the success/failure levels. However, qualitatively, there was a dominating feeling that this approach cannot be used forever. At least it doesn’t support scalability to higher number of projects
- The disciplined stage represents the time period that the company was able to build and follow a quality management system that utilizes a plan-driven approach with different lifecycles which were iterative in nature. CMMI for Development [6] and the SECC-Software Process Improvement Guide [21] were the basis for this quality management system
- The disciplined agility stage was the final one and the one that contained the highest number of radical changes at the same time. These radical changes happened as sequence of consecutive changes that each one was a foundation to the successive one. These changes included the employing of the “Empowered Self-Directed Team”, “Performance Analysis to Support Agility Through Informed Decisions” and introducing the concepts of “Software Inspection and the SEI-Personal Software Process (PSP)” training [22]. Fig. 5 presents these stages and the included sub-stages or major phases:

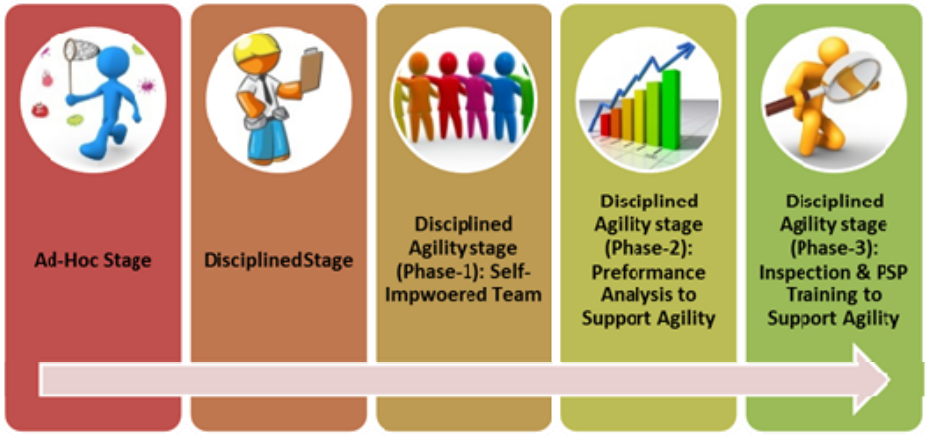

Figure 5. Development Stages and Phases (The Internal Change Cycle)

However, some important notes should be highlighted. Although the widths of all stages/phases are equal in the graph, this does not mean that the time periods are equal two. The detailed information about the time scale, the break-

through achievements and results as well as the major challenges/pain areas that appeared in each stage can be found in the following list. The challenges/pain areas are the challenges that could not be solved during this stage/phase and required the introduction of new skills or approaches to overcome the barriers and pushing the organization towards the next stage/phase to release its full potential.

- ***Ad-Hoc Stage (2005 to 2008):***
  - Major Challenges/Pain Areas:
    1. Typical 100% to 200% total variance in the release time
    2. Uncontrolled product quality (*average 100 critical functional defects per release delivered to customer*)
    3. Ad-hoc team interaction

- ***Disciplined Stage (2008 to 2011):***
  - Breakthrough Achievements:
    1. Typical 30% to 50% total variance in the release time
    2. Controlled product quality (*average 40 critical functional defects per release delivered to customer*)
    3. CMMI ML2 accreditation achieved

  - Major Challenges/Pain Areas:
    1. The difficulties of centralized planning, measuring the performance and managing the product quality
    2. Relative high effort and schedule planning variance especially with shared human resources which complicates project plans
    3. Effective and efficient collecting and analyzing measures with high impact to project and product success is a pain. It needs considerable effort to collect and employing several tools to analyze
    4. Storing and managing performance historical data in an organized repository in order to be employed in future projects planning
    5. Relative uncontrolled product quality (*more than 60% of critical functional defects are identified on system test, which highly increase the cost of defect fixing*)
    6. Effective management of both the teams and the individuals in order to increase performance and improve the quality
    7. Low team motivation, commitment and adaptability

- ***Disciplined Agility Stage-Phase 1 (Q4-2012 to Q1-2013):***
  - Breakthrough Achievements:

1. Improved team motivation, commitment and adaptability
2. Increased visibility of product vision, project status and product quality
3. Utilization of a comprehensive measurements system (*with 40+ metrics/indicators*)

- Major Challenges/Pain Areas:
  1. The need to correctly interpret the available metrics and take the corrective decisions

- ***Disciplined Agility Stage-Phase 2 (Q1-2013 to Q2-2013):***
  - Breakthrough Achievements:
    1. Fully understanding of the dynamics of the daily challenges and performance
    1. Ability to take suitable decisions to adapt to the changes
    2. Effective improve in the ability to recover from the negative consequences of the changes
    3. Typical 7% total variance in the release time

  - Major Challenges/Pain Areas:
    1. Many logical defects escape to system test or customers side that originate of requirements and high level design
    2. Effort distribution per tasks indicates that the time of the design was too small in comparison to the code and test, which reduces the final product quality

- ***Disciplined Agility Stage-Phase 3 (Q2-2013 to Q2-2014):***
  - Breakthrough Achievements:
    1. Enhanced product quality (*average 3 critical defects per release*)
    2. Reduced cost of poor quality by improving process yield (*from 40% to 85%*)
    3. Reduced percentage of rework in testing and bug fixing and focusing on improving the design to raise the overall product quality (*to around 15%*)

5- CROSS-STAGES ANALYSIS

In order to make the previously mentioned breakthrough achievements more appealing, the changes of the quantitative values will be compared between the three different stages. Furthermore, a major qualitative aspect will be discussed at the end, which is related to the team interactions. The analysis will be limited to cover the firstly stated aspects in the introduction section, which are summarized as follows:

- Controlling the time to market
- Improvements in process yield & dynamics
- Improvements in delivered product quality
- Improved team dynamics and collaboration

5-1 RATIONALE BEHIND DISCUSSED METRICS

Whatever the followed project management methodology, all project have some ultimate objectives to achieve, which are usually concerned with delivering the required scope on time, higher delivered product quality and distributing the effort/cost wisely during the project lifecycle [25, 26]. However, all these objectives should not come with the cost of destroying team dynamics and individual morals [30, 35].

Hence, although there are a lot of well know metrics that can used to evaluate the performance of software development process, project and product [15, 16, 27, 28], it seems logical to focus on the previously mentioned small set of metrics to evaluate the improved performance across the different stages.

This does not mean that the other metrics are not important, but it means that this limited set should be at the heart of the analysis efforts. Other metrics can help you to dig deeper to gain more insight if needed. Comprehensive lists of the available metrics in the SEI-TSP/PSP and Agile methodologies can be found in [29], [30], [32] & [33].

All metrics that will be discussed in the following sections were collected using the SEI-TSP/PSP individuals and team excel tool (A simplified version of the tool can be obtained from SEI website [34]) in the early beginning and then the Software Process Dashboard tool [23] after it has been introduced. Data collection and validation approach followed the SEI-TSP/PSP procedures and guidelines [15, 16, 29, 30] that were explained during the coaching and training sessions to the teams in MiniMax.

In brief, data collection focused on recording size, efforts, schedule and defects data. While data validation activities were concerned with the correctness and competence of the recorded data. Finally, during metrics analysis the coaching approach focused not only on the technical meaning of each metric/indicator, but more practically on the following aspects:

- Evaluating the process, project and product status and situation from the qualitative point of view to collect unstructured open feedback from project team about the situation in hand
- Explaining the importance of the metrics under the focus of the analysis and emphasizing its role in the overall methodology/approach
- Explaining the current situation and linking it to previously discussed qualitative feedback to indicate how the metrics is able to reflect the real situation
- Compare and contrast the current value with other obtained values to gain better insight about the progress. As an example, the current value can be compared to previous weeks or other projects/modules
- Focus on transferring the obtained insight into decisions that aim to improve the performance and enhance the situation
- Analyze the previously taken decisions to understand their effect on the performance and understand if these decisions were correct or not

Hence, as a summary, the simple followed logic says that you need to reduce the variance in the delivery time (section 5.2), while maintaining the quality of the delivered product (section 5.4) proactivity by managing the defects early during the life cycle (section 5.3). Of course, achieving this should be through a highly dynamic and motivated team (section 5.5).

5-2 CONTROLLED “TIME-TO-MARKET”

Being able to respond to changes and being fixable and adaptive are critical success factors in the nowadays turbulent marketplace. These themes are fully supported by pure agile methodologies. However, these flexibility and adaptability should not overcome the tendency to minimizing the total time to market (i.e. the total time to release) or at least to control it.

This is where the mix between the disciplined approaches and the agile one can add value. Utilizing SEI-Team Software Process metrics provides the required level of support to either control this time or at least knowing the impact of the daily changes on the overall release time [11, 14]. Fig. 6 shows the ability of the company to control the deviations in the time to market from stage to another.

Although disciplined approaches were able to reduce the variance dramatically, introducing the disciplined agility provided lower levels of variance. The variation of the time to market was reduced from the range of 100% to 200% in the ad-hoc stage to within the range of 7% to 10% at the disciplined agility stage. This increased the dependability of the process. You can trust team when they promise to deliver on time.

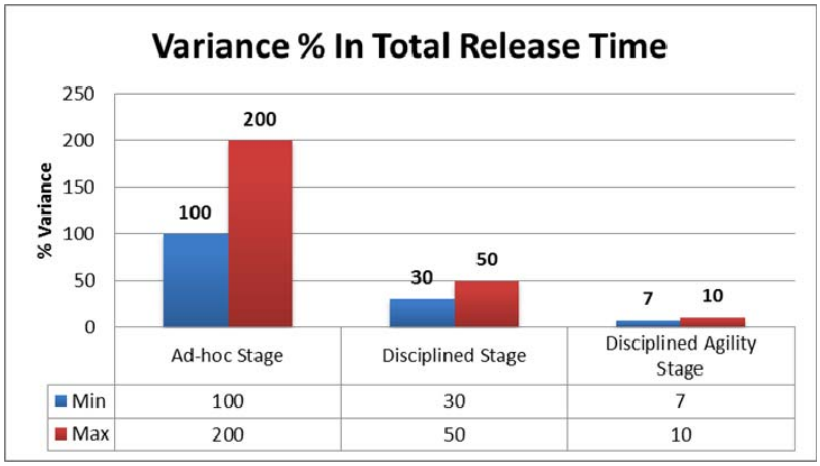

Figure 6. Reduction in Release Time Variance

5-3 IMPROVED PROCESS YIELD & DYNAMICS

Being able to deliver on time and achieving a shorter time to market are critical success factors. However, the quality of the delivered product and the cost of achieving/controlling this quality is more, or at least equally, important. Controlling and forecasting the quality of the delivered product is not one of the fully covered topics in agile methodologies [11, 13 & 14].

This is where the mix between the disciplined approaches and the agile ones can add value. Utilizing SEI-Team Software Process metrics provides the required level of support to either control and forecast this quality or at least know the impact of the daily changes on the overall release time [11, 14].

One of the major concepts in TSP/PSP is the overall process yield [15, 16]. Shortly, it is the ability of your process to capture the defects before reaching the testing activities which is considered as a cost of poor quality [29].

This means that finding and removing these defects before testing reduce both the cost of poor quality and the overall project cost, which improves the profit margin [36].

Fig. 7 shows the process yield in the disciplined stage, while Fig. 8 does the same for the disciplined agility stage2. Process yield is more than doubled while the total number of defects found during testing was reduced by 62.5%. We can fairly assume the same percentage reduction in the cost in dollar. No need to mention the qualitative impacts of these results. The team and individuals self-confidence were improved dramatically.

[2] No similar data available from the ad-hoc stage

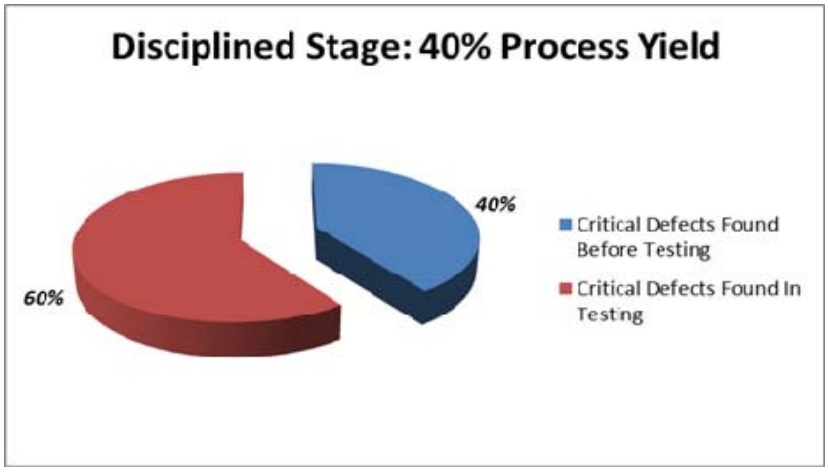

Figure 7. Disciplined Stage Process Yield

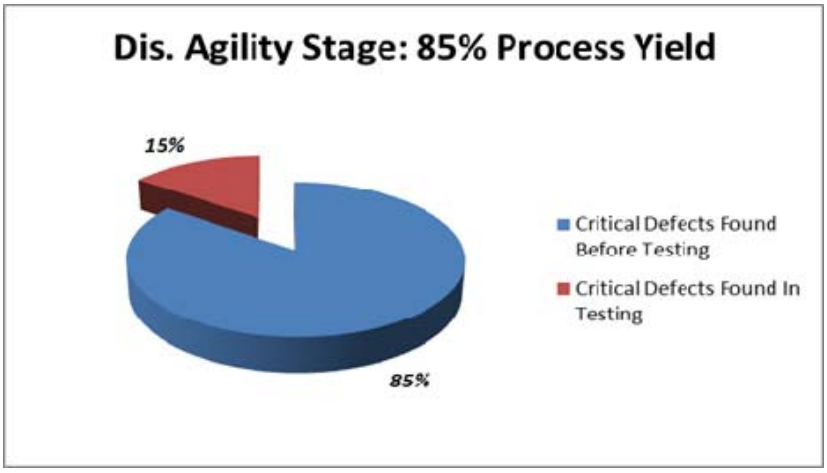

Figure 8. Disciplined Agility Stage Process Yield

To understand how these results were achieved, we can have a look on the effort distribution by activities types and where the defects were found and removed. Fig.9 shows the effort distribution by activities types in the disciplined stage before introducing the concept of inspection, while Fig. 10 does the same for the disciplined agility stage after introducing the inspection.

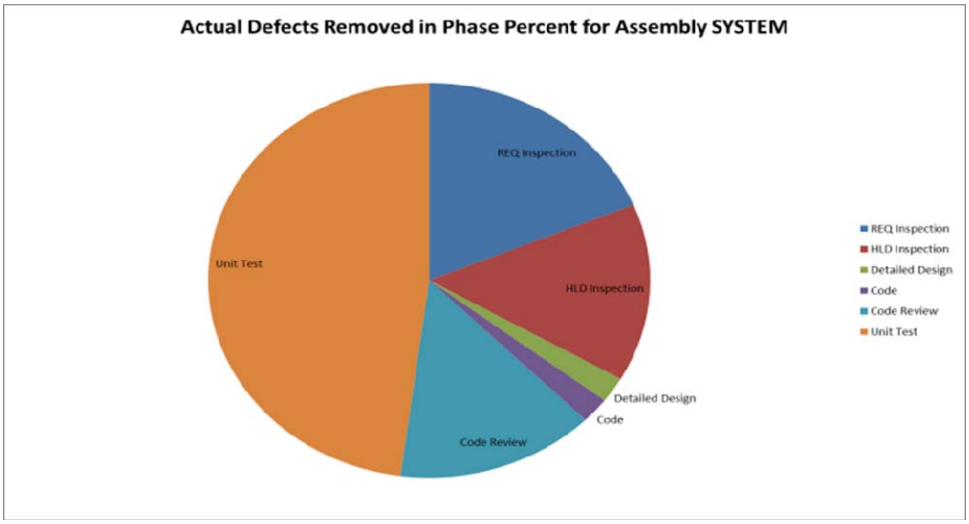

Figure 9. Effort Distribution/Release before Inspection

Before inspection, the code was taking the majority of the effort of all of the technical activities. This means that all of the other “Doing” tasks and related “Reviewing/Inspecting” tasks were done in hurry. As an example, producing the high level design and architecting was done almost ad-hoc with no quality gates that can discover the produced defects before propagating in the system.

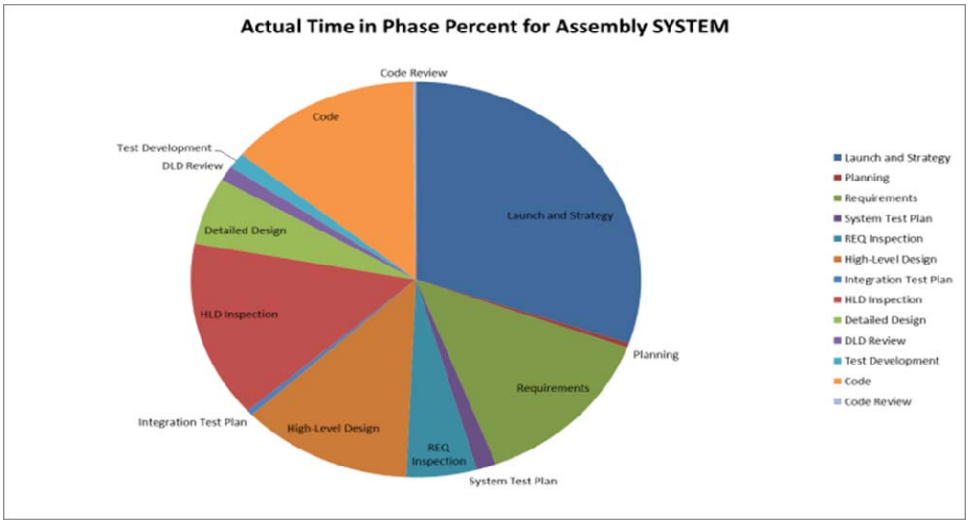

Figure 10. Effort Distribution/Release after Inspection

Introducing the inspection strengthen the quality gates and raised the awareness of the quality of each individual which encourages him to focus more on the quality of the job in hand before sending to inspection. This also reduced the amount of effort required in coding as shown in Fig. 10. Finally, understanding where the defects were detected and removed can explain both the changes in process yield and effort distribution. Fig. 11 shows defects removal distribution by activities types in the disciplined stage before introducing the concept of inspection, while Fig. 12 does the same for the disciplined agility stage after introducing the inspection.

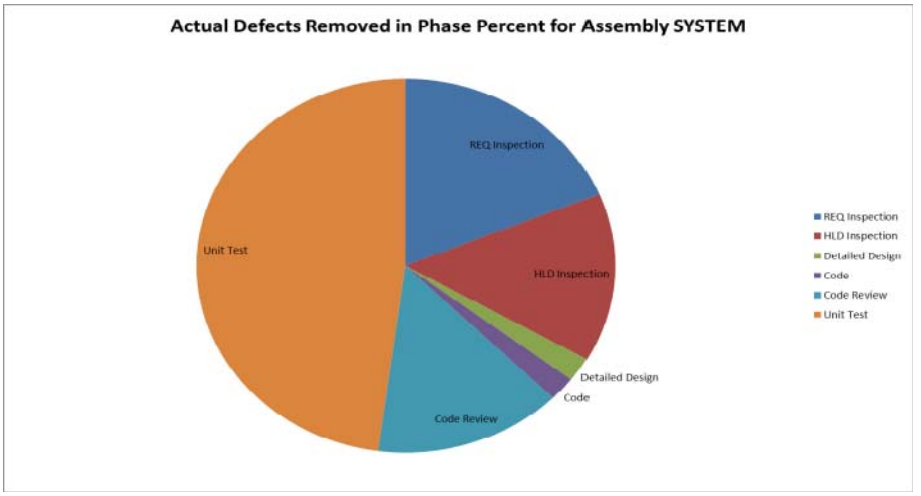

Figure 11. Defect Removal Distribution before Inspection

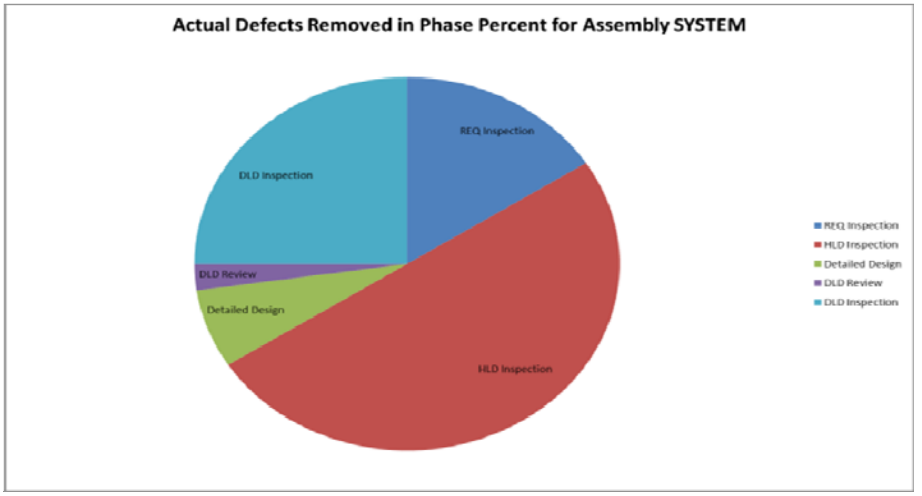

Figure 12. Defect Removal Distribution after Inspection

By "Introducing Inspection", in this paper, we mean the detailed design inspection on team level (i.e. in the TSP) and detailed design reviews (i.e. In the PSP). The requirements inspection and high level design (i.e. HLD) inspection were already known practices from the disciplined stage. This can be noticed in Fig. 11.

Before inspection, almost half of the defects was found in testing (i.e. a failure cost), which raises the cost of bug fixing and the total cost of the project. Introducing the inspection worked as the quality gates and raised the awareness of the quality of each individual which let him focus more on the quality of the job in hand before sending to inspection. The result was almost zero defects in testing (at least in the interim builds). The average number per the whole release was around only three defects as explained in the next section.

5-4 IMPROVED DELIVERED PRODUCT QUALITY

The value of the improved process yield and dynamics can be summarized in the quality of the finally delivered product. The customer is usually shielded from all these details, except in the outsourced projects. Hence, the total number of defects per the final release can be the solo metric of quality. Sometimes these defects are called the “Post-release Defects”. Fig. 13 shows the dramatic reduction in the number of the post-release defects per stage. The impact of introducing the disciplined agility approach limits this number to around three per release. No need to mention the huge number of good results based on this achievement.

As an example, the total cost of maintenance was reduced dramatically. The amount of the maintenance fees in the contract became almost a net profit. Furthermore, customer satisfaction and trust boosted dramatically too. This induced higher level of customer loyalty and changed the market position of the company from the producer of “Low-Price/Low Quality” products to producer of “Low-Price/High Quality” products. It was a vivid strategic shift.

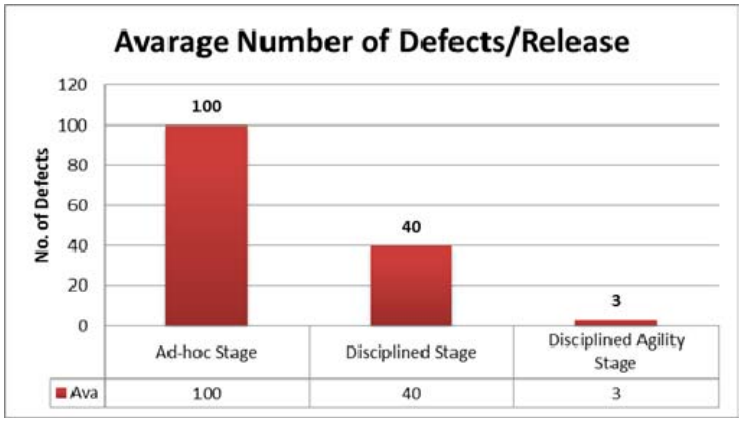

Figure 13. Average Number of Post-release Defects

5-5 IMPROVED TEAM DYNAMICS

From another perspective, all these breakthrough improvements should not come in the cost of team moral, engagement, commitments, and welling to innovate. Of course, ad-hoc communications cause frustrations. Centralized communications becomes a bottleneck and reduce team moral, engagement, commitments, and welling to innovate. A self-directed team [15] is the only well-known team structure that boosts all of the above mentioned factors. Fig. 14 shows how the team dynamics and structure changed across stages till reaching the final required distention of the self-directed team.

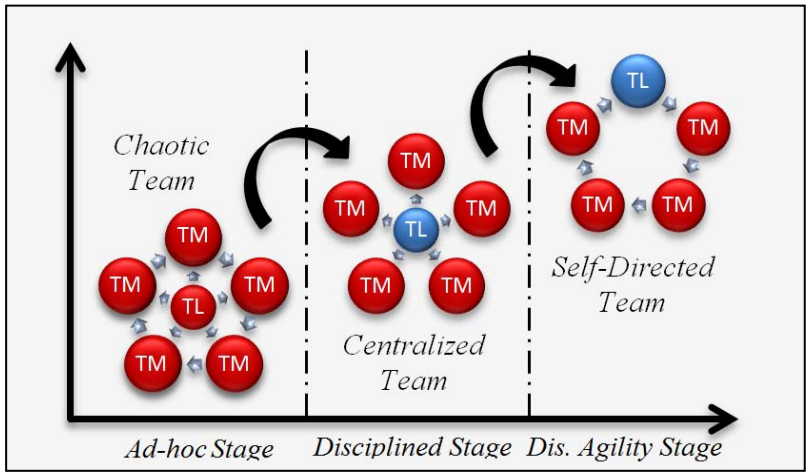

Figure 14. Team Dynamics across Stages

During the first stage, all communications channels were possible without any guidance from the team leader for team members on how to communicate or interact to accomplish the mission in hand. Of course, this came with the cost of communication overhead and waste as well as some forms of frustration, low morale and low commitment.

In the second stage, the communication channels and interaction mechanisms were reformulated in favor of centralization. The role of the team leader became clearer and had higher impact on team direction and performance. Although this added some benefits and clarity to the daily work, these benefits vanished so fast. With the increasing in team size, project duration and product complexity, this team leader became overwhelmed and represented a clear bottleneck.

Furthermore, this induced the passive and helpless behavior between team members as they felt that they are neither responsible for the project performance nor product quality. “There is a Boss” culture dominated the teams.

Finally, introducing the TSP helped the team to understand the details of both the mission in hand and the product vision as well. This impacted team moral, engagement, commitments, and welling to innovate positively as well as made them responsible to adapt to changes in a constructive and productive manner. This doesn't mean that the role of the team leader role in the project became less important; however, his role was totally changed. He became a facilitator more than a controller, which suites the dynamic nature of software development projects.

6- THE DISCIPLINED STAGE

The focus of discussion in the disciplined stage will be on the introduced concepts, skills and approaches as well as the achieved breakthrough improvements. To keep the logical flow of discussion, the challenges that appeared in each phase/part of the stage will be highlighted as they were the reason behind the successive radical changes.

The first step was to introduce the concept of the process based on SECC-SPIG [21] and CMMI-DEV [6] and builds the first quality management system. This helped the company think in the development projects as a system that includes many phases (such as initiation and planning) and many disciplines (such as project and configuration management). This was the stage of transferring the ad-hoc system into a disciplined one (i.e. the stage of introducing the effectiveness inside the system). MiniMax team was able to reduce the total variance in the planned release time from between 100% to 200% to the range of 30%-50% and limit the number of post-release critical defects to the range of 40 per release. Although this solved many problems, but highlighted many others such as:

- **The Difficulty of The Centralized Planning**

  It was a very challenging task for project manager to plan for a long term project in advance and achieve a good level of resource utilization. No need to mention the difficulty of responding to changes in project plan or product requirements. More importantly, the low level of involvement and commitment from team members were great barriers towards success.

- **The Difficulty of Measuring Performance**

  Collecting sufficient and accurate data about the performance of individuals, team, project status, and product quality was almost an unattainable goal. The effort and complexity of collecting two to three metrics were huge and there was no trust of the accuracy of the obtained numbers. Furthermore, all metrics were lagged metrics which means that numbers were obtained at the end of the project to indicate that the project had failed. There is no room to recover from the failure.

- **The Difficulty of Managing Product Quality**

  The only available metric to measure product quality was the defects that are discovered at customer side after product delivery. This means not only a reactive behavior, but a totally passive too. There is nothing to be done to improve neither this delivered product nor the upcoming ones.

7- THE DISCIPLINED AGILITY STAGE

By introducing SEI-Team Software Process [15] that achieves a good balance between the plan-driven approaches and the dynamic agility, MiniMax development team, with support of SECC TSP Coach, was able to solve these problems gradually. This was the stage of transferring the disciplined system into a more agile and adaptive system that is able to reflect on its previous performance to improve its future conduct of the development practices (i.e. the stage of improving the efficiency of the system). This happened through the following sub-stages:

- **Introducing The Concept Of Empowered Self-Directed Team**

  Through the typical TSP launch workshop [15], each project team was enabled to engage in a full workshop for understanding the mission of the project and plan for product vision through successive releases as well as producing detailed plan for the first iteration. Team engagement in the launch workshop improved their understanding of the mission in hand and the overall vision of the product. Furthermore, it enhanced their ability to adapt to changes that affect either a single iteration goal or the overall release plan. In addition, during the same launch workshop, SECC TSP Coach, helped the team install, understand and utilize a comprehensive measurements system using the SEI-TSP Excel Workbook in the beginning and then the Software Process Dashboard [23]. This was the initial step toward solving the second problem (*which is the difficulty of measuring the performance*). After the workshop, the team was guided by the TSP coach to focus on recording time, defects and schedule data accurately as possible without the focus on discussing or analyzing the results.

- **Introducing The Concept Of Performance Analysis To Support Agility**

  After collecting the data for around three weeks, SECC TSP Coach helped the team to read and interpret different performance indicators on both individuals and team level with the focus on analyzing the project status and product quality. Indicators [23] include release and iteration burn-down charts, agile earned value charts, direct task hours, defects profile, quality profiles and many more (*around 40+ metrics/indicators*). The most powerful characteristics of these indicators is that they are updated instantaneously with the closure of each individual task or detecting/resolving each defect as well as their availability for individuals and team all the time through the intranet.

Knowing how to interpret these indicators helped MiniMax teams understand the dynamics of their daily challenges and performance. In return, they were able to take suitable decisions to adapt to the changes they were facing and effectively improve their ability recover from the negative consequences of the changes. As an example, they were able to achieve around 7% variance in the total release time. However, this comprehensive measurements system was able to highlight some of the chronic problems that the team was not able to react to. There were no obvious decisions that the team can take in the time horizon of any single iteration or even a whole release.

For example, logging the efforts and defects attributes highlighted that there were many defects of the logical types that may escape to system test or customers that originate of requirements and high level design. Furthermore, effort distribution per tasks in each single piece of code highlighted that the time of the design was too small in comparison to the code and test, which reduces the final product quality. There were no obvious options that can be done, which was an indicator that the team is in a learning stage that needs the introduction of additional skills.

- **Introducing Inspection and PSP Training**

  Introducing inspection [15, 24] on the team/project level (detailed design inspection) and the personal reviews (detailed design review) on the personal level using PSP [16] solved many problems. On the team level, inspection helped the team catch defects early and reduces the number of defects that escape to the customers at the end with lower cost of quality. Furthermore, this helped the team adjust the effort distribution across each iteration and the whole release (*i.e. reduce the percentage of rework in testing and bug fixing and focusing on improving the design to raise the overall product quality*). On the individual level, the same effects happened. The total number of defects that appears from individual work in the system test was reduced and the effort distributions on individual tasks were shifted towards the design to improve the final product quality. At this stage the total number of defects per release was reduced from around 40+ critical defects to only three defects pre- release.

8-CONCLUSION

To help development teams reach the effective and efficient way of work, a close guidance is needed. A close coaching and mentoring is needed to help the teams transform their approach from the ad-hoc manner to the disciplined agile approach. During SECC engagement with MiniMax team, SECC TSP Coach/Consultant was able to help the team move from the “Ad-hoc” to the “Effective Disciplined State” and from the effective disciplined state to the “Efficient Disciplined Agile State”. Transition across these stages was managed by introducing a set of new skills when the team reaches its full potential in his

current stage with the support of a comprehensive measurements system that was able to highlight the important areas of improvements.

ACKNOWLEDGMENT

This work is a direct result of the Software Engineering Competence Center-SECC's fund provided in the form of subsidized consultancy services (by around 90% of the original cost) delivered to MiniMax Software from 2008 to 2014. All consultancy services were provided through the first author (*who is a SECC Senior Consultant, SEI-Certified TSP Coach & SEI-Authorized PSP Instructor*) and managed internally within MiniMax by the second author (*who is MinMax CTO*).

REFERENCES

[1] Dwolatzky B., "South Africa's proposed TSP Pilot Program," JCSE at Wits University. Based on: Standish Group Report, 2009.

[2] Wikipedia: Project Management, 2013. [Online]. (https://en.wikipedia.org/wiki/Project_management). (Accessed: September 11, 2014)

[3] Project Leadership Waterloo, Week 2b. Project Management Methodologies, 2013 [Online]. (http://projectleadershipwaterloo.wordpress.com/2011/05/16/project-management-methodologies/). (Accessed: September 11, 2014)

[4] Phillips J., "PMP: Project Management Professional Study Guide," McGraw-Hill Professional, ISBN 0-07-223062-2 p.354, 2003.

[5] Paulk M., Curtis B., Chrissis M., and Weber C., "Capability Maturity Model for Software," Version 1.1. Technical Report, CMU/SEI-93-TR-024, 1993.

[6] CMMI Product Team, "CMMI® for Development," Version 1.3. Improving processes for developing better products and services. TECHNICAL REPORT, CMU/SEI-2010-TR-033, 2010.

[7] Information Technology Infrastructure Library (ITIL), 2014 [Online]. (http://www.itil-officialsite.com/home/home.aspx). (Accessed: September 11, 2014)

[8] Wikipedia: Waterfall Model, 2014 [Online]. (http://en.wikipedia.org/wiki/Waterfall_model). (Accessed: September 11, 2014)

[9] Schwaber K. and Sutherland J., "The Scrum Guide: The Definitive Guide

to Scrum & The Rules of the Game," 2011. [Online]. (http://www.scrum.org/portals/0/documents/scrum guides/scrum_guide.pdf). (Accessed: September 11, 2014)

[10] Boehm B. and Turner R., "Balancing Agility and Discipline: A Guide for the Perplexed Paperback," Addison-Wesley/Pearson Education; 1st edition, 2003.

[11] Schmidt D., "Balancing Agility and Discipline at Scale," SEI Blog Series on Team Software Process, 2012.

[12] Agile Manifesto. 2001. [Online]. (http://agilemanifesto.org/). (Accessed: September 11, 2014)

[13] Microsoft Developers Network, "Ten Year Agile Retrospective: How We Can Improve In The Next Ten Years," 2011. [Online]. (http://msdn.microsoft.com/en-us/library/hh350860.aspx). (Accessed: September 11, 2014)

[14] Over J. "Views on Balancing Agility and Discipline," SEI Agile Research Forum. Part of SEI Webinar Series. May 2012.

[15] Humphrey W. "The Team Software Process (TSP)," TECHNICAL REPORT. CMU/SEI-2000-TR-023, 2000.

[16] Humphrey W., "The Personal Software Process (TSP)," TECHNICAL REPORT. CMU/SEI-2000-TR-022, 2000.

[17] Software Engineering Institute (SEI), 2014 [Online]. (http://www.sei.cmu.edu/). (Accessed: September 11, 2014)

[18] Phillips M., "CMMI in Focus: CMMI with Agile, Lean, Six Sigma, and Everything Else," news@sei, columns, January 2008.

[19] McAndrews D., "The TSP: An Overview and Preliminary Results of Using Disciplined Practices," CMU/SEI-2000-TR-015, 2000.

[20] Davis N. and Mullaney J., "The Team Software Process (TSP), in Practice: A Summary of Recent Results," CMU/SEI-2003-TR-014, 2003.

[21] SECC-Software Process Improvement Guide (SPIG), 2014. [Online]. (http://www.secc.org.eg/SECC_SPI.asp). (Accessed: September 11, 2014)

[22] Personal Software Process (PSP) Fundamentals, 2014. [Online]. (http://www.sei.cmu.edu/training/p18b.cfm). (Accessed: September 11, 2014)

[23] Software Process Dashboard, 2014. [Online]. (http://www.processdash.com/). (Accessed: September 11, 2014)

[24] Fagan M., "Design and Code inspections to reduce errors in program development,". IBM Systems Journal 15, 1976.

[25] MSDN Blog, New book: Successful Project Management: Applying Best Practices and Real-World Techniques with Microsoft Project, 2011. [Online]. (http://blogs.msdn.com/b/microsoft_press/archive/2011/03/28/new-book-successful-project-management-applying-best-practices-and-real-world-techniques-with-microsoft-project.aspx). (Accessed: June 16, 2015)

[26] University of Glasgow: Fast, cheap, and good: The cost-quality tradeoff triangle /quadrilateral, 2014. [Online]. (http://www.psy.gla.ac.uk/~steve/best/tri.html). (Accessed: June 16, 2015)

[27] Grady B., “Software metrics: establishing a company-wide program,” Prentice-Hall, Inc. Upper Saddle River, NJ, USA, ISBN: 0-13-821844-7, 1987.

[28] Fenton E. and Lawrence S., “Software Metrics: A Rigorous and Practical Approach”, Third Edition, ISBN-13: 978-1439838228, 1998.

[29] Humphrey W., "PSP: A Self-Improvement Process for Software Engineers," Addison Wesley. ISBN-13: 978-0321305497, 2005.

[30] Humphrey W., "TSP: Leading a Development Team (The SEI Series in Software Engineering)," Rearson Education. ISBN-13: 978-0321349620, 2005.

[31] Hayes W. in SEI Blog, Agile Metrics: Seven Categories, 2014. [Online]. (https://blog.sei.cmu.edu/post.cfm/agile-metrics-seven-categories-264). (Accessed: June 15, 2015)

[32] Hayes W. in SEI Blog, Agile Metrics: Seven Categories, 2014. [Online]. (https://blog.sei.cmu.edu/post.cfm/agile-metrics-seven-categories-264). (Accessed: June 15, 2015)

[33] Pillai S. in Scrum Alliance Community Blog: Agile Project Reporting and Metrics, 2013. [Online]. (https://www.scrumalliance.org/community/articles/2013/july/agile-project-reporting-and-metrics). (Accessed: June 15, 2015)

[34] SEI Website: Introductory Team Software Process (TSPi), 2015. [Online]. (http://www.sei.cmu.edu/tsp/tools/tspi/). (Accessed: June 15, 2015)

[35] Humphrey W. "TSP: Coaching Development Teams,". Addison-Wesley. SBN: 0201731134, 2006.

[36] Humphrey W., “Winning with Software: An Executive Strategy”, Addison-Wesley, Boston, ISBN 0201776391, 2002.