ISSN: 2348 - 2117

International Journal of Engineering Technology and Computer Research (IJETCR)
Available Online at www.ijetcr.org
Volume 4; Issue 6; November-December; 2016; Page No. 09-17

SOFTWARE DEVELOPMENT PROCESS AND QUALITY MANAGEMENT METRICS

AARUSHI ARYA (13BCE0679) aarushia.95@gmail.com

NIKHIL SHANKER MATHUR (13BCE0280) mathur.nikhil7@gmail.com

AMAN AGARWAL (13BCE0508) aman9425307728@gmail.com

PROF. MANJULA R. (SOFTWARE PROJECT MANAGEMENT)

Abstract

Any software developed needs to possess superior quality, performance, reliability and durability in order to ensure smooth functioning for a long period of time.

Software quality is the degree to which a component, system or process meets the requirements, needs and expectations specified by the customer. The aim of this paper is to investigate the measures available to determine different quality factors. The identification of factors and as well as the metrics and measures was done on the basis of the literature survey by studying and analyzing various research papers. The results benefit software developers to easily identify the metrics used to measure the quality characteristics of the software. Furthermore, the work aimed at providing some suggestions, using the potential deficiencies detected as a basis. Also, comparisons will be made between the most common approaches of quality management, namely, Capability Maturity Model (CMM), The International Organisation of Standards(ISO) which sets certain standards and test that each product needs to pass in order to be able to sold in the market, Total Quality Management(TQM) and Six Sigma which further is divided into two sub-methodologies- DMAIC(define, measure, analyze, improve, control) and DMADV(define, measure, analyze, design, verify).Other common approaches will also be looked into and compared. These approaches have been in the industry of products and services development from a very long period of time and through their data-driven approach and methodologies to remove defects from the product, they have become the basic approaches to measure the quality of any software or product or service released in the market.

Introduction

Beginning with the evolution of the Capability Maturity Model (CMM), it was a development model initially created after performing a study on the data collected from various organisations that had contracts with the U.S. Department of Defence which was behind the funding of the research.

The term Maturity here has been used in the form of improvement of processes using various methods ranging from hit a trial to more formally defined steps to obtaining result metrics to finally optimum improvement of the processes.

The main aim of the model was to improve the software-development process, but in the later years it found its application in the development of various other processes too.

Moving on to the Capability Maturity Model Integration (CMMI), it was basically a development in the CMM model where the application of software development was facing a lot of problematic situations. The CMM model had various sub models whose implementation across one entire organisation without any integration turned out to be a huge issue. This was leading to cost ineffectiveness, many trainings and improvement activities. The CMMI was developed to sort the issue of using multiple models for the use of software development processes, hence in this aspect the CMMI model has dominated the CMM model even though the CMM model is used or referred to as the general theoretical process capability model used in the public sector.

Another quality management technique that has found its way in determining the quality is Total Quality Management (TQM). This is more business oriented than software development oriented but over time it has found its applications in the software development processes too. TQM is believed to be a highly systematic, participative approach to planning

Corresponding author: AARUSHI ARYA

and thereby implementing any process that can add business value to an organisation.

Another approach discussed in this paper is the Six Sigma approach. Now Six Sigma is a combination a set of innovative and adapted methodologies oriented towards the improvement of efficiency and effectiveness of a corporate process. In any corporate process, efficiency and effectiveness are very closely related. While one means doing right things the other means doing things the right way. Originally Six Sigma was introduced by Motorola Co. in the year 1986 and has since evolved to become one of the most successful business improvement strategies. Like TQM, Six Sigma was also developed as a business improvement methodology but over time it has found its application in the software development process too. The Six Sigma approach can be used to reduce the number of errors or defects occurring in the manufactured goods to less than 3.4 per 1 million units.

The International Organisation of Standards (ISO) has defined certain set of standards that are used in the development of quality management system in all organisations. ISO 9000 standards have been known to be applied to a variety of organisations ranging from manufacturing to service oriented industries.. The most general ISO 9001 standard has been used to be applied on the organisations that design, maintain and develop their own quality processes. The ISO 9000-3 has been used to interpret ISO 9001 for software development process.

The ISO 9001 is't specific to just the software development process but its principles can be applied various projects of software development.

SOFTWARE PROCESS FRAMEWORK CMM

Each organisation has certain Key Process Areas. This framework has been designed to guide anyone who wants to assess the performance of any organisation or product with respect to these Key Process Areas. For each maturity level in the CMM there are five checklist types:

Table 1:

Type	Description
Policy	Describes the policy contents and KPA goals recommended by the Key Process Areas.
Standard	Describes the recommended content of select work products described in the Key Process Areas.
Process	Describes the process information content recommended by the Key Process Areas. These are refined into checklists for: Roles, entry criteria, inputs, activities, outputs, exit criteria, reviews and audits, work products managed and controlled, measurements, documented procedures, training, and tools
Procedure	Describes the recommended content of documented procedures described in the Key Process Areas.
Level overview	Provides an overview of an entire maturity level. These are further refined into checklists for: Key Process Areas purposes, goals, policies, and standards; process descriptions; procedures; training; tools; reviews and audits; work products; measurements

Top Ten Principles of Conventional (Waterfall) Software Management

1. It is imperative that the requirement gathering is done at the very initial step. The requirements of the project should be very carefully decided and later precisely implemented. Any change in the requirements at a later stage could affect the functioning of the system drastically. It could result in code not being implemented properly and the test cases would also fail. Also an effort should be made to make the requirements as unambiguous as

possible and only once the requirements re set should the major investments be made in design and other development processes and activities.

2. Another thing that needs to be ensured is that a design needs to be developed before beginning with the coding process. The design can greatly influence the coding of the project and a simple change in the design could bring significant change in the code of the project. Hence, it is always recommended to materialise the design before beginning to code the project.

3. Modern day has seen the development of some very high-order programming languages. These languages are very easy to understand and make the concepts very easy to implement. Such languages also ensure that very complex concepts can now be implemented in few lines of code. These languages use advanced data typing, interface separation and other packaging and programming constructs.

4. There's a big difference between the design implementation process and the testing process. While the design focuses on implementing the broader concepts first followed by designing the units within the concepts, the testing part focuses on checking the functioning of each unit first and then combing or integrating and testing the model as a whole. It is always easier to find errors in smaller codes. This attempt is made to remove all the minute errors or small-scale errors that might have come up during the coding process and if not worked on at the unit level then they might result in substantial rework at the whole level. Hence, always complete the unit testing before integration.

5. When a system is designed, a lot of new scenarios that haven't occurred before are obtained and all of them are carefully analyzed to develop a proper design followed by code for the same. These new scenarios or investigations or experiments are termed as artefacts and with artefacts comes the need to maintain traceability among them. This traceability can ensure that the program is consistent at each stage of the project development. The requirement artefacts need to be traced to the level of design and test artefacts. This helps in determining the impact that any change on one level could have on other levels of the project.

6. The design created needs to be carefully documented. For this purpose the use of SDS (Software Design Specifications) is made. Documentation is considered to be so important for design that in the initial stages the documentation is considered as the design. It is only in the later stages that the code becomes the main artefact and the design artefacts must be updated in order to ensure completeness and consistency. It is believed that in the initial stages, it is the document which needs to be considered as the design and later on, once the code becomes the main artefact, the dean artefacts are made and developed to ensure consistency and to provide a basis for any change that needs to be brought to the project.

7. It is highly essential to assess the quality of the project. The teams which have planned, coded, designed, tested and analyzed the project would not be able to assess the quality of the project as much as a team assigned specifically for this purpose. To know that the final product and the processes is in accordance to all the standards and is of the best quality, there needs to be a quality assessment team.

8. Testing and Inspection are two very different approaches. While testing means coming up with various scenarios and test cases that can occur to assess the deliverance and performance of the product, inspection on the other hand means checking the functionalities and the performance delivered by the product in real time scenarios and situations. It means looking at the actual functionalities of the product and the processes. Inspection should cover all the artefacts o the product beginning form the planning to code followed by design and test artefacts.

9. Planning is the key to a successful product. The planning should be started out keeping the broad picture in mind and should branch out to the most core and critical decisions. This is necessary to identify critical paths and manage the risks that might occur and also evaluate e various programmatic changes.

10. It is also necessary to have a control over the source code baselines meticulously. Once the artefacts move into the coding phase, careful configuration management becomes imperative to maintain the baseline control of formal releases in the test process and to develop the product with absolutely no errors making it suitable for the release.

The most desired Ten Principles of Modern (Iterative) Software Management

1. Focus the procedure on the architecture first. A demonstrable balance among the driving requirements, architecturally significant design

decisions, and lifecycle plans before the organisation reserves the required resources for full-scale deployment is needed.

2. Deal with risks well in time with an iterative lifecycle. An iterative procedure is required to refine comprehension of the issue, and to shape a viable arrangement and a plan that guarantees adjusted treatment of all partner objectives. Real risks should be tended to right on time to build consistency and avoid costly scrap and modify later on.

3. Stress on component-based advancement. To diminish the measure of human-produced source code and custom improvement, extend groups must move from a line-of-code mindset to a component based attitude inside a current structural system. A component is a durable arrangement of prior lines of code, either in source or executable configuration, with a characterised interface and behaviour.

4. Build up a change administration environment. The elements of iterative improvement incorporate simultaneous work processes, as various groups use shared artefacts. This calls for objectively controlled baselines that all members can access.

5. Upgrade change flexibility with devices for round-trip engineering. Round-trip building gives the environment support important to mechanise and synchronise engineering data in various configurations (e.g., requirement specifications, plan models, source code, and executable code). Without significant utilisation of automation, it is hard to decrease iteration cycles to manageable time spans that permit and encourage change. Option to change artefacts is a need in an iterative procedure, as it evacuates one of the transcendent sources of friction perceived by the teams.

6. Utilise thorough, model-based notation. A model-based approach (e.g., UML) underpins the advancement of semantically rich graphical and textual outline documentations. Visual demonstrating with thorough documentations and a formal, machine-processable language allows more target evaluation than the conventional human audit and assessment of ad-hoc plan representations in paper records.

7. Instrument the procedure for target quality control. Lifecycle evaluations of both the procedure and every intermediate module should be firmly incorporated into the procedure, utilising characterised measures derived from the advancing engineering artefacts and coordinated into all tasks and teams.

8. Utilise demonstration-based appraisal of artefacts. Transitioning the present, state of-the-product artefacts (whether an early model, a benchmark engineering, or a beta version) into an executable exhibit of important use cases invigorates prior union on incorporation, more unmistakable comprehension of plan tradeoffs, and prior disposal of structural imperfections.

9. Plan releases with advancing levels of detail. It is vital that the software administration prepare drive ahead of schedule and constant trials within the operational setting of the framework, in particular its use cases. Every venture augmentation and exhibition should reflect current levels of detail for both necessities and design. Use cases are the essential component for arranging prerequisites, characterising iteration content, surveying executions, and arranging acceptance testing.

10. Set up a versatile, configurable process. No single procedure is reasonable for all product advancement ventures. To be businesslike, a procedure system should be configurable to a wide range of tasks. To guarantee economies of scale and rate of profitability, the association must ingrain a common procedure "spirit," so that all tasks acquire a typical arrangement of best practices, particularly for project administration and setting autonomous work processes, checkpoints, measurements, and artefacts. It should likewise permit specialisation so that every venture can optimise the procedure usage for the particular setting of the venture.

Approaches in TQM:

Some of the approaches that have been implemented by well known organisations and have been found to be successful are:

- Knowledge ,adaptability ,flexibility, change
- Work force skill , motivation ,innovation
- Improvement of means
- Quality ,productivity, process performance
- System optimisation
- Proactive, dynamic, continuous change
- Truth, beauty and god
- To build competence
- Process oriented
- Rational networking

Functions of TQM:

The major functions of TQM are:

- Directing and Controlling
- Organising

- Planning
- Staffing

Benefits of TQM:

TQM has endless benefits which help an organisation in getting better results. Some of them are:

- Improved employee morale
- Increased customer loyalty and retention
- Higher profitability
- Elimination of defects and waste
- Higher productivity
- Strengthened position in comparison to competitors
- Adaptability to changing or emerging market conditions and to environmental and other government regulations
- Enhanced market image
- Reduced costs and better cost management
- Improved customer focus and satisfaction
- Increased job security
- Improved and innovative processes
- Enhanced shareholder and stakeholder value

Advantages of TQM:

TQM has been implemented in various industries and organisations and has been successful irrespective of their size, area, sophistication and level of production, they had the following advantages:

- Better control on process
- Improved cost effectiveness ,reduce unit cost
- Improved competitiveness. quality is profitable
- Lowered costs
- Improved Reputation
- Better profitability, better returns reduction in operating cost
- Reduction of rejection, scrap and wastage-reworking

ISO

ISO 21500:2012, Guidance on Project Management, is a worldwide standard created by the International Organisation for Standardisation, or ISO beginning in 2007 and released in 2012. It was proposed to give generic direction, clarify centre standards and what constitutes great practice in venture management.[1] The ISO specialised council managing venture administration, ISO/PC 236 was held by the American National Standards Institute (ANSI) which had endorsed four benchmarks that utilised PMI materials, one of which was ANSI/PMI 99-001-2008, A Guide to the Project Management Body of Knowledge - fourth Edition.

ISO plans for this standard (21500) to be the first in a group of project administration norms. ISO likewise composed this standard to adjust with other, related standards, for example, ISO 10006:2003, Quality management systems – Guidelines for configuration management, ISO 31000:2009, Risk administration – Principles and rules. Since ISO 21500:2012 is a direction record, it is not proposed to be utilised for certification/registration purposes.

The ISO 9001 standard does not characterise the quality procedures that ought to be utilised as a part of the advancement procedure. Associations can build up their own quality procedures and they can in any case be ISO 9000 consistent organisations. The ISO 9000 standard just requires the definition of procedures to be utilised as a part of an organisation and it is not worried with guaranteeing that these procedures give best practices and high caliber of items. Hence, the ISO 9000 affirmation doesn't imply precisely that the nature of the product delivered by an ISO 9000 confirmed organisations will be superior to that product from an uncertified organisation.

SIX SIGMA APPROACH can be further divided into two techniques, DMAIC and DMADV, as focal points to inspect and address correlative parts of business procedures. The DMAIC and the DMADV qualifications are aimed at reviewing distinctive divisions of a business all the while however tending to them independently. In spite of one of a kind qualifications, the strategies cover amid the examination procedure and have similar true objective - improvement of business procedures.

Every procedure has its own particular arrangement of rules and objectives focused at enhancing business processes using information accumulation and statistical tools. While the systems are intended to accomplish similar thing, there are important contrasts between the two that ought to be considered by experts in administration roles or in business situations with an extensive variety of authoritative settings.

DMAIC

The arrangement of Six Sigma techniques that is most applicable to the assembling or generation side of an item or service, DMAIC incorporates these venture stages:

◦Define - address the identification of particular procedures to be analyzed
◦Measure - record information and utilise measurements to track viability and assess efficiencies

◦Analyze - use basic speculation aptitudes to survey information and elucidate objectives
◦Improve - make changes in business forms designed for development and better arrangement with corporate objectives

◦Control - construct an arrangement of checks and alterations for progressing change underway procedures

DMADV

The corresponding arrangement of Six Sigma forms that is most material to looking at and enhancing the client relations side of an organisation, DMADV incorporates these venture stages:
◦Define - address client needs in connection to an item or administration
◦Measure - include the utilisation of electronic information accumulation to quantify client needs, reaction to item, or review of services
◦Analyze - use measurements to assess ranges where item or administration can be better adjusted to client objectives and requirements
◦Design - cover the change of business procedures that streamline corporate objectives to best meet customer and client needs
◦Verify - fabricate an arrangement of tests and models to monitor that client details are being met through on-going enhancements

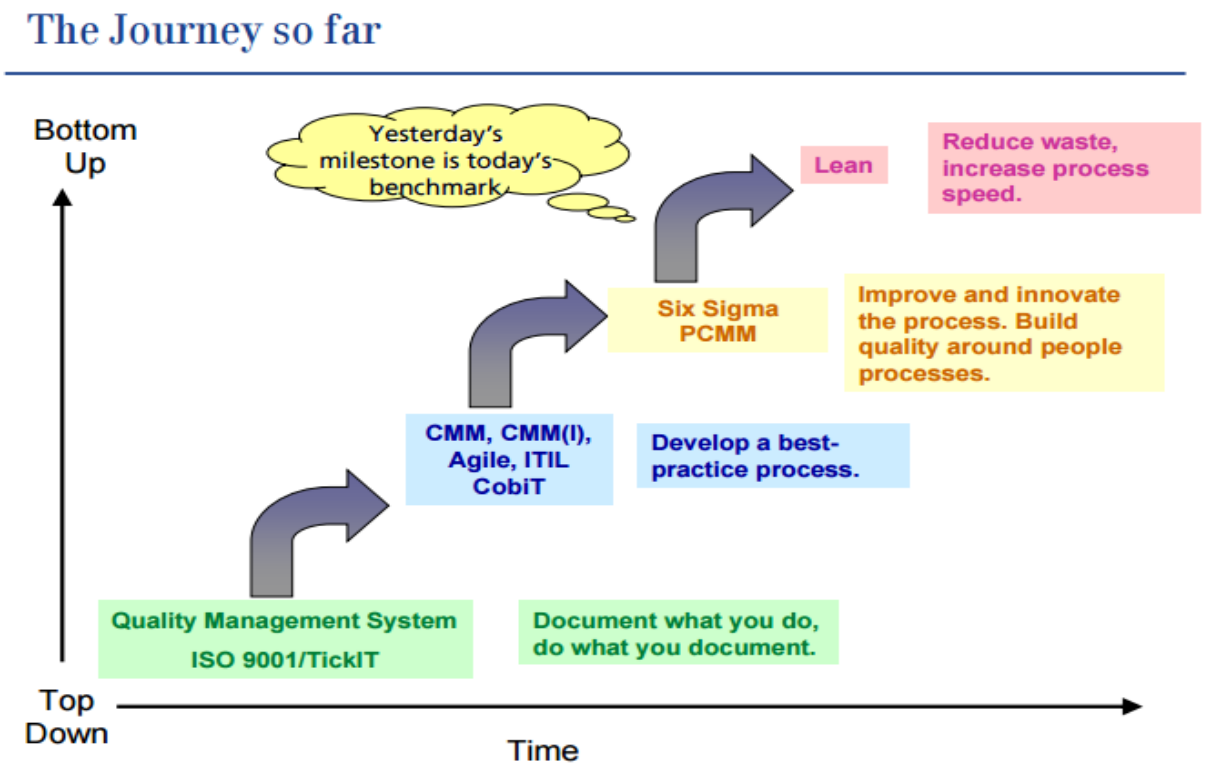

Figure 1: A photo showing the journey so far of concerned methods

Comparisons

We can compare CMM and TQM as follows:
Capability Maturity Model (CMM) and the Total quality management (TQM) were analyzed as organisational change paradigms. This study considered six characteristics of TQM and CMM: initiator of change; issues addressed; basic conception of organisations; change technology; solution offered; and inherent problems. TQM deals with the work environment and customer satisfaction, while the CMM deals with the development issues and the software systems. Under TQM solutions we need new ways for using tools like statistical process control (SPC) and management. While the solutions presented by CMM are focused

more on evolutionary improvement through the five maturity levels. The originator of change in TQM is the top management along with the contribution of all employees. All the changes via CMM are initiated by the software engineering process groups and the specified project team members. While in TQM the change technology comprises of the adoption of TQM philosophy; running of specific plan-do-check-act cycles and implementation of statistical process control. The CMM approach relies on the IDEAL model of initiating, diagnosing, establishing, acting, and learning. In TQM the problems we encounter are its origination in the manufacturing sector and the specific expertise required for its implementation, while the problems in CMM include its origination in the defence sector and the greater emphasis on software engineering. Indeed CMM would greatly benefit from adopting some TQM characteristics and principles, including dealing with people issues.

Table 2: Comparison between Traditional and Six Sigma

Traditional	Six Sigma
Solves only quality tools	It can solve key business pains. Also solve problems beyond the domain of quality
Mainly used for manufacturing Only	Used for both manufacturing and service
For gut - based solutions	For variables affecting a process
Quality for mostly domain levels	Helped quality professionals to add directly to the top line/ bottom line. Can be used in any part of the business. Quality from all levels,
Regarded as quality program	Regarded as Business strategy
Only, the professionals in the quality control department will specify customer satisfactions measures.	Here individuals define customer satisfaction measures and using teams continuous improvements are achieved at all levels of the enterprise. It also uses customer focused goals and measurements.

TQM compared to ISO 9001

ISO 9001 is a standard for system quality management. TQM is a philosophy of perpetual improvement. The ISO quality standard sets in place for a system to deploy policy and verifiable objectives. We know that an ISO implementation is a basis for a TQM implementation. If we have an ISO system about 75% of the steps are already in place for TQM .In short, implementing TQM is being practical concerning the quality rather than reactive.

The principles of TQM are as follows:

- everyone has a customer and is a supplier
- processes, not people are the problem
- every employee is responsible for quality
- quality must be measured
- quality improvements must be continuous
- goals are based on requirements, not negotiated
- life cycle costs, not front-end costs
- problems must be prevented, not just fixed
- management must be involved and lead
- the quality standard is defect free
- plan and organise for quality improvement.
- quality can and must be managed

TQM mainly focuses on quality assurance function, review, quality plan, inspection and internal audit.

The strengths of ISO 9001 are described as follows:

- document configuration control
- defect tracking and status report
- training activities and plans.
- project management process

Comparison of CMM and ISO 9001 models

Both of the CMM and the ISO 9001 standards were developed to improve the organisational processes. Paul and Rozman have done comparison of the model where there is a strong correlation between ISO 9001 and CMM, although there are some issues in CMM which are not covered in ISO 9001 and vice versa. Only the minimum criteria for an acceptable quality system are satisfied by the ISO 9001. Another important difference is that CMM focuses strictly on software, whereas the ISO 9001 has a very broader scope that encompasses processed materials and services, system and software. The ISO 9001 documentation should contain instructions or guidance on what has to be done or how it has to be done. CMM emphasises on processes which are well documented and practised as documented. Now, comparison between ISO 9001 and CMM is showed in Table 4. Table 5 and is concerned with users who are currently using CMM and are also interested in ISO 9001:2000 compliance or towards users who are simultaneously addressing CMM process improvement and also ISO 9001 compliance.

Table 3: Comparison of ISO 9001 and CMM.

ISO 9001	CMM
Minimum requirement to become certification with implied continuous improvement	Five levels of certification with continuous improvement
Designed to apply to all industry development	Designed specifically for the software industry
Outwardly focused	Inwardly focused
Yearly recertification	No yearly recertification process

Table 5 Mapping ISO 9001:2000 to CMM

ISO 9001 clause	CMM process areas and practices
4.1: General requirements	Outsourcing, supply agreement management, quality assurance and management, process definition and process improvement
4.2: Documentation requirement	requirement Integrated enterprise management: evaluation, quality assurance and management: configuration management, information management
5.1: Management commitment	Integrated enterprise management
5.2: Customer focus	Needs, Requirements
5.3: Quality policy	Integrated enterprise management
5.4: Planning	Integrated enterprise management, needs, quality assurance and management
5.5: Responsibility, authority, and communication	Integrated enterprise management
5.6: Management review	Integrated enterprise management Project management

- **Difference between Six Sigma and CMMI**

Six Sigma and CMMI are two of the finest enhancement-oriented initiatives, with a number of overlaps. However a major fundamental difference between these two is that CMMI is specifically domain specific whereas Six Sigma is not.

The indispensable difference between Six Sigma and CMMI pertains to the scope of application. CMMI aims at process improvement in specific disciplines or process areas. The basic model of CMMI usually caters to 22 of such disciplines or “process areas". On the contrary Six Sigma usually aims at solving various process or product related issues for the improvement of the overall organisational processes. So we see that CMMI is a highly domain specific improvement engine whereas Six Sigma caters to a much broader application, used as both a tactical improvement engine and an enterprise governance model.

- **Relationship between CMMI and Six Sigma**

Six Sigma and CMMI must not be in competition and if in an organisation these concepts are implemented simultaneously then it helps in successfully achieving the company goals quickly in a better and cheaper way.

Many companies use CMMI and Six Sigma simultaneously using integrated process architecture. Such integration could take four approaches:

1. Implementation of CMMI to the most elevated amount to set up frameworks and procedures in view of set up best practices. At that point, these practices are trailed by usage of Six Sigma to tweak such procedures in view of client desires and to proceed with required enhancements.

2. First implementing Six Sigma as the governance model and then using CMMI to close the various problematic gaps in process infrastructure.

3. Simultaneous execution of CMMI and Six Sigma by utilising CMMI to recognise basic process figures and conveying Six Sigma structures in such basic variables, or utilising utilising Six Sigma to distinguish territories where CMMI is required.

4. CMMI Level 3 is implemented to establish defined processes and then successfully implementing Six Sigma to move to higher CMMI maturity levels. Six Sigma either helps in the flourishing implementation of CMMI or helps in the successful transition from one CMMI level to another. The appropriate approach to be selected greatly depends on the concerned organisation’s circumstances.

An examination of CMMI versus Six Sigma scatters numerous myths with respect to these ideas going up against each other. An essential distinction in approach in any case, CMMI and Six Sigma compliment each other while disregarding the conceivable cooperative energy resultant from utilisingutilyizing these ideas as a part of pair leads, which keeps an organisation from copying endeavours.

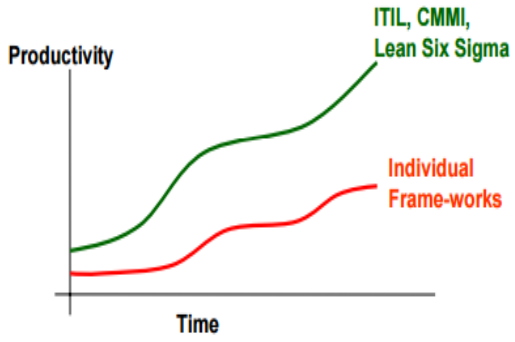

Figure 2: A photo showing the varying productivity over time of the concerned processes

References

1. www.google.com
2. https://www.wikipedia.org
3. http://www.brighthubpm.com/six-sigma/69025-relationship-between-cmmi-vs-six-sigma/
4. http://resources.sei.cmu.edu/asset_files/Presentation/2008_017_001_22517.pdf