Inter-relationship of KPAs in Capability Maturity Models

Manzur Ashraf
King Fahd University of Petroleum & Minerals
Dhahra, KSA
Email: manzur@ccse.kfupm.edu.sa

***Abstract**: The heightened importance of the software process has created a need for process improvement, which requires methods for process analysis and assessment. The Capability Maturity Model (CMM) categorizes a software process maturity into five levels- from level 1 to level 5 with an objective of getting more automation in the software process. This paper quantitatively inter-relates the factors behind those transitions. Moreover the influence of Key Process Areas (KPAs) and sub-KPAs leading to different level-transitions will be presented with the aid of Bayesian Belief Networks. Context-specific tendency towards higher CMM level on the basis of different key factors (KPAs) are illustrated using quantitative assessments.*

***Keywords**: CMM, BBN, Software process Engineering, Software Engineering, Quantitative Software Engineering.*

1 Introduction:

The Capability Maturity Model (CMM) summarizes the best practices for the software development processes [1][2][3][4][10][11][12][13]. By outlining software development processes and activities and declaring the execution of repetitive activities, this model aims to efficiently executing organizational missions. Organizations with higher CMM levels are expected to operate using a more stable, sophisticated, and disciplined approach by making use of automated tools and the experiences gained from many past successes.

As the understanding of the repeatable processes increases, rules are constructed to avoid certain mistakes and to promote productivity. When the rules are composed as standard processes, then the automation grows more substantial as it is enlisted to reduce rework as process maturity increases. This is a type of software process improvement represented by CMM.

2 Maturity levels in CMM:

One of the objectives of CMM is to distinguish mature processes from immature or ad hoc processes [3][4]. In a software organization that follows immature software processes, projects are executed without appropriate guidelines of controlled and matured processes. However, with the aid of efficient capability of project team & leader. Consequently the project result is unpredictable & inefficient in most cases. On the contrary, if the processes of an organization are matured, the outcome of the project is less dependent on the team capability and more controlled by the processes. The result of the project is more predictable in the later case.

The path to higher maturity includes well-defined steps termed as *maturity levels* by CMM. Each level specifies certain characteristics for processes, with higher maturity levels having more advanced characteristics. The CMM framework describes key elements of software processes at different levels of maturity. The path includes five maturity levels.

In level 1, an organization executes a project in a manner that the team and the project manager are efficient. The repeatable level (level 2) is applied to an organization in which project management practices are well established, although organization-wide processes may not exist. Requirement management, project planning and tracking, contract & configuration management, quality-assurance are its key processes. In the defined level (level 3), periodic process assessments, necessary training programs, software projects are planned and managed using organization's standard software process and rules; integrated software management, inter-group coordination and peer reviews are achieved. In level 4 (managed level), plan for qualitative process management is taken; the performance of the project is controlled quantitatively. In level 5 (optimized level), plan for managing technology changes is adopted; a documented procedure for developing and maintaining plans for software process improvement is taken. Measures of defect prevention are also taken into consideration in this level.

3 Key Process Areas of CMM-levels:

We illustrate key process areas of different levels and highlight their goals according to [1][10][14]. Then we present our observation of inter-relationship among different goals of KPAs. The basis is subject analysis of goals and their sequential placement in terms of activities.

3.1 Level 1 (Initial):
The software process is characterized as a need-basis and sometimes even chaotic. This stage defines few processes and success depends on individual, mainly team leader's effort and pragmatics. There are no KPAs so far defined by SEI in this level.

3.2 Level 2 (Repeatable) KPAs
Project-management perspectives are established to measure cost, track schedule and functionality. The project's process is under the effective control of project management system, following realistic plans based on the performance of previous projects.

3.2.1 Requirements Management: There are two goals of this KPA.
The first goal focuses on requirements being documented and controlled, and the second focuses on maintaining of consistency of other documents (e.g., Plans, designs, etc.) with software requirements.
Therefore, goal 1 affects goal 2.

3.2.2 Software Project Planning. There are three goals of this KPA.
Goal 1 requires that estimates are documented, and activities supporting that require that estimates for size, effort, schedule, and critical resources are derived using a *documented procedure.*
Goal 2 requires that project activities are planned and documented. The activities require that the plan is controlled and managed, and identifies risks also.
Goal 3 requires that affected groups and individuals agree to their commitments and its activities require that the commitments are reviewed by senior management.
So, goal 3 is affected by both goal 1 and 2.

3.2.3 Project Tracking and Oversight. There are three goals in this KPA.
Goal 1 requires that actual performance is tracked against the plans. Activities require that effort, cost, schedule, and risks are tracked, and actual measurement data is recorded, and formal reviews are held at milestones.
Goal 2 requires that corrective actions are taken when actual performance deviates from planned.
Goal 3 requires that changes to plans and commitments are agreed and reviewed.
So goal 1 affects goal 2 which evidently affects goal 3.

3.2.4 Software Subcontract Management..
Goal 1 requires that a qualified subcontractor is selected.
Goal 2 requires that there is an agreement with the subcontractor.
Goal 3 says that there is a regular communication with the subcontractor.
Goal 4 requires that the performance of the subcontractor is tracked against its commitments.
So goal 1 and 2 individually affects goal 3 which then affects goal 4.

3.2.5 Software Quality Assurance:
Goal 1 says that Software quality assurance activities are planned
Goal 2 says that adherence of software products and activities to the applicable environment are verified objectively.
Goal 3 says that affected groups and individuals are informed of the quality assurance activities and results are audited as well as preserved.
Goal 4 states that senior management addresses Noncompliance issues that cannot be resolved within the project.
Here goal 1 affects goal 2 and goal 2 affects both goal 3 and 4.

3.2.6 Software Configuration Management.
Goal 1 defines that software configuration management activities are planned.
Goal 2 says that selected software work products are identified & controlled.
Goal 3 says that affected groups and individuals are informed of the status and content of software baselines.
Goal 1 to 2, 2 to 3 sequentially affects each other.

3.3 Level 3 (Defined) KPAs

The software process for both management and engineering activities is documented, standardized and integrated into a standard software process for the organization. All projects use an approved version of the organization's standard software process for developing and maintaining software.

3.3.1 Organization Process Focus. There are three goals in this KPA.
Goal 1 states that software process activities are coordinated across the organization. The activities require that a group exists for this coordination, a process database is there, tools and processes in limited use are evaluated for possible use in the rest of the organization (i.e. learning), and training is provided.
Goal 2 requires that the process is assessed periodically and action plans developed based on the report.
Goal 3 requires that organization level process development and improvement activities are planned.
Goal 1 affects 2 and 2 affects 3.

3.3.2 Organization Process Definition. This has two goals.
Goal 1 requires that a standard process for the organization is developed. The activities require that process is developed according to a documented procedure, and is documented using some standards, and guidelines and criteria for tailoring of the process for projects exist.
Goal 2 requires that information about process use is collected. Activities require that a process database be established, and a library of process related assets be maintained.
Goal 1 affects goal 2.

3.3.3 Training Program. The goal 1 requires that training activities are planned.
Goal 2 is that training is provided. The activities for this require that procedures for conducting training are needed and standards for courses are needed.
Goal 3 requires that individuals receive training, and an activity requires that there should be a waiver procedure for required training.

Here Goal 1 affects 2 and 2 affects 3.

3.3.4 Integrated Software Management. It requires that the process being used on the project is a tailored version of the organization process. Project is planned and managed according to the project's process. The activities require that lessons are learned from projects and documented, a documented procedure for risk management exists, projects effort and critical resources are managed, thresholds for deviation from planned are established for effort, schedule, critical resources, and reviews of the project are periodically done.

3.3.5 Software Product Engineering.

Goal 1 is that software engineering tasks are defined and performed consistently. CMM requires that rationale for tool selection is documented.
Goal 2 requires that consistency is maintained across different work products.
Goal 1 affects goal 2.
3.3.6 Inter-group Coordination. The Goal of Inter-group Coordination is to establish a means for the software engineering group to participate actively with the other
engineering groups so the project is better able to satisfy the customer's
needs effectively and efficiently.

3.3.7 Peer Reviews. The Goal of Peer Reviews is to remove defects from the software work products early and efficiently. Additionally trying to develop a better understanding of the software work products so that defects can be prevented. The peer review is an important and effective engineering method that can be implemented via inspections, structured walkthroughs, or a number of other collegial review methods.

3.4 Level 4 (Managed) KPAs

Detailed measures of software process and product quality are collected. Both the process and products are quantitatively understood and controlled.

3.4.1 Quantitative Process Management. This KPA requires that goals for the performance of the process on a project are established, based on the past performance of the process, and measurements are taken along the process execution and data used to quantitatively control the project.

3.4.2 Software Quality Management. This KPA requires that quantitative quality goals are set for the software products, plans are made to achieve the goals, and that the actual progress is monitored and corrections made, if needed, to ensure that goals are met.

3.5 Level 5 (Optimizing) KPAs

Continuous process improvement is enabled by quantitative feedback from the process and from directing innovative ideas and technologies. Although not explicitly mentioned in the goals, the fact that level 5 comes after level 4 implies that effects of defect prevention, technology, and process change be measured quantitatively at level 5. [1]

3.5.1 Defect prevention: The purpose is to identify the causes of defects and prevent them from recurring. The software team analyzes errors, defects, identifies reasons of defects and modifies is defined processes.

3.5.2 Technology Change Management: The objective is to identify useful new technologies (i.e., tools, methods, and processes) and transfer them into the organization in an orderly manner. The focus of Technology Change Management is on performing innovation efficiently in an ever-changing world.

3.5.3 Process Change Management: Its purpose is to continually improve the software processes used in the organization with the intent of improving software quality, increasing productivity, and decreasing the cycle time for product development.

4 Exploration of Inter-relationship of KPAs

4.1 The Impact of P-CMM on SW-CMM Implementation:

The main goal of SW-CMM is to create a direction to rise above the chaos arising out of dependence on individuals. To establish good software processes is not sufficient to take away this dependence. People need as much attention & to be focused as well as to motivate them to rise above themselves and their projects, thus contributing to organizational enrichment.

While implementing SW-CMM for any organization, existing people processes can be directed to advantage. People processes help to obtain the commitment across the organization on innovative ideas. Moreover the investment into SW-CMM becomes optimal also.

4.1.1 People Capability Maturity Model (P-CMM):

The People Capability Maturity Model (People CMM) [15] is a framework that helps organizations successfully address their critical people issues. Based on the best current practices in fields such as human resources, knowledge management (KM), and organizational development, the People CMM directs organizations in improving their processes for managing and developing their workforces. The People CMM helps organizations characterize the maturity of their workforce practices, establish a program of continuous workforce development, set priorities for improvement actions, integrate workforce development with process improvement, and establish a culture of excellence. [15] Reports the inter-relationship or mapping of P-CMM KPAs into CMM KPAs. It is named as 'Influence Matrix' described in next section. But in this report we term it as 'Relationship matrix', since we name derived 'probabilistic table' as 'Influence matrix'.

4.1.2 Relationship Matrix:

The Relationship Matrix represents which KPA of P-CMM impacts which SW-CMM KPA strongly. While implementing SW-CMM, organizations would benefit from analyzing what gaps the P-CMM KPA has, and address it appropriately by either closing the gaps, or improvising the SW-CMM process itself.

Table 1: People CMM to SW-CMM mapping

	RM	SPP	SPTO	SSM	SQA	SCM	OPF	OPD	TP	ISM	SPE	IC	PR	QPM	SQM	DP	TCM	PCM

Work Environment		√	√	√			√		√									
Communication	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Staffing		√	√	√	√		√											
Performance Management		√	√	√	√					√		√	√	√	√	√		
Training	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Compensation		√	√		√									√	√	√		
Knowledge & skills Analysis		√	√		√				√								√	√
Workforce Planning		√	√	√						√								
Competency Development	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Career Development	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Competency-Based Practices	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Participatory Culture	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Mentoring		√	√							√								
Team Building		√	√	√	√		√	√	√			√	√				√	√
Team-Based Practices		√	√	√	√		√	√	√			√	√				√	√
Organizational Competency Management	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Organizational Performance Alignment	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Personal Competency Development	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
Coaching		√	√		√							√						√
Continuous Workforce Innovation														√	√	√	√	√

4.2 The Impact of ISO 9001 on SW-CMM Implementation

4.2.1 Influence Matrix for ISO 9001:

ISO 9001 is a standard that has 20 clauses, which are meant for service organizations [17]. This standard has been used for a software organization in ISO 9000-3 [18], and further guidelines and illustration has been given in TickIT guidelines [19]. Differences have been done between the ISO9001 model and the CMM [20][21]. However, these tend to be comparison between the models and tend to be clause-by-clause or KPA-by-KPA (key practice area) analysis like section 4.1.

There is an inter-relationship or mapping of ISO key processes with CMM KPAs. [3] reports that inter-relationship in table 2.

Table 2 ISO 9001 to SW-CMM mapping

	RM	SPP	SPTO	SSM	SQA	SCM	OPF	OPD	TP	ISM	SPE	IC	PR	QPM	SQM	DP	TCM	PCM
Management responsibility		√	√		√										√			
Quality System		√			√			√			√							
Contract review	√	√													√			
Design control		√	√			√					√				√			
Document and data control						√					√							
Purchasing				√														
Control of customer-supplied product				√														
Product identification and traceability						√					√							
Process control		√			√						√			√			√	
Inspection and testing											√		√					
Control of inspection, measurement & test equipment											√							
Inspection and test status						√					√							
Control of conforming product						√					√							
Corrective and preventive action					√	√										√		
Handling, storage, packaging, preservation and delivery						√					√							
Control of quality records						√					√		√	√				

Internal quality audit					√													
Training									√									
Servicing																		
Statistical technique								√							√			

5 Our direction:
According to mark C.Paulk, SEI, Carnegie Mellon University, there is a clear understanding of the relationship between the KPA in the CMM and its goals. [2]. This relationship is subjective. He did not provide quantitative aspects of that relationship. Our objective of investigation is to that direction. We illustrate the subjective relationship of KPAs according to ISO-CMM mapping in Part 1, People CMM-CMM mapping in Part 2 and finally our own subjective viewpoint in Part 3.

Part 1: According to commonality of goals of KPAs and mapping those to ISO key processes we can create relationship using a directed graph. Our convention A→ B means that the ‘B’ affects ‘A’. We can use Bayesian Belief Net (Discussed in section 5.1) to calculate the approximate (‘low’,’ high’) influence of ‘B’ on ‘A’. We term it as ‘inter-dependency’ and our goal is to calculate that influence provided that historic data are available in this context. The figure 1 is based on dependency according to ISO-CMM mapping.

Part 2: In terms of People CMM, the KPA dependency is all-to-all mapped. It means that all KPAs of level n affect all KPAs of level (n+1) and higher. So we can conclude that in terms of ‘People’-effect on CMM, all lower level KPAs affect all upper level KPAs.

Part 3: Our own subjective analysis based on goals [1] reveals the dependency graph of Figure 4. First we make assumption of our parameters:
List 1[Parameter chart]:

Level 5:
Process change management C1
Technology Change Management C2
Defect prevention C3
Level 4:
Software Quality Management C4
Quantitative Process Management C5
Level 3:
Organizational process focus C6
Organizational process definition C7
Training program C8
Integrated S/W Engineering C9
S/W Product Engineering C10
Inter-group communication C11
Peer reviews C12
Level 2:
Requirement Mgmt C13
S/W Project planning C14
S/W project tracking & oversight C15
Subcontract management C16
Quality assurance C17
Configuration management C18

5.1 Bayesian Belief Networks:

Bayesian Belief Networks are a network-based framework for representing and analyzing models involving uncertainty. BBN comes from the cross disciplines of probability, artificial intelligence, and decision analysis. Bayesian Belief Networks have established themselves as an effective and principled framework for knowledge representation and reasoning under uncertainty. Bayesian Belief Networks exploit conditional independence relationships to create natural and compact domain models, thereby supporting useful reasoning patterns, and providing effective probabilistic inference and learning algorithms. Because of the development of propagation algorithms follows by availability of easy to use commercial software and growing number of creative applications, BBN catches the sudden interest of research in different research fields since early 90's. After ten years researches, BBNs have succeeded creating models in areas of intelligent decision, safety assessment, medical diagnosis, pattern recognition, and computer network diagnosis.

The key feature of BBN is that they enable us to model and reason about uncertainty. Typically in BBN modeling, we assign a Bayesian belief value to each uncertain event, such as the belief value of "It is cloudy" is assigned as 0.55, the belief value of "John is late" is 0.2, etc. All these probabilities for the uncertain events come from people's subjective judgments, which are determined by collecting empirical, historical or statistical data. When we are dealing engineering projects, these probabilities can be decided by domain experts who can possibly be project managers, lead designers, senior programmers, and test researchers. In BBN modeling, the probability representations of uncertainties are assigned as the prior belief values to each node in a certain BBN.

5.2 The Formal Definition of Bayesian Belief Networks:

A Bayesian Belief Network consists of the following:
i) A set of variables and a set of directed edges between variables;
ii) Each variable has a finite set of mutually exclusive states;
iii) The variable together with the directed edges from a directed acyclic graph (DAG);
iv) To each variable A with parents B1,….,Bn, there is attached the conditional table
P(A| B1,…..,Bn).

In the topology of a particular BBN, nodes depict variables; arcs between variables represent probabilistic dependency; the Conditional Probability Table associated to each node encode the strength of the dependencies. In Figure 2 and 3, we describe that dependency table (Influence matrix) particularly for our case-study.

6 Experiments and findings:

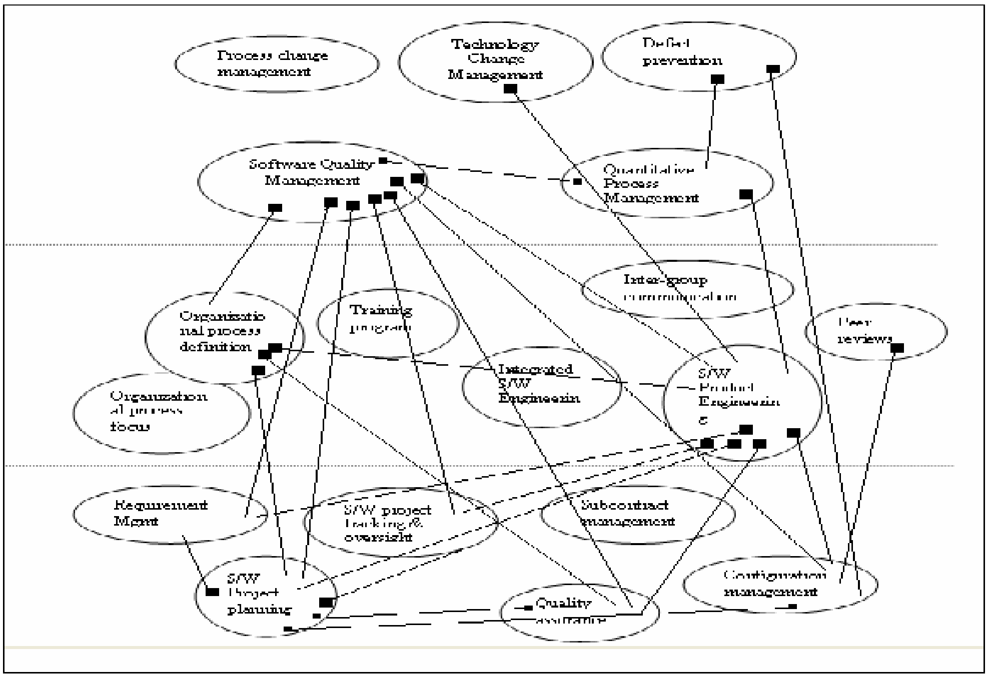

Figure 1 Dependency graph of KPAs according to CMM-ISO mapping

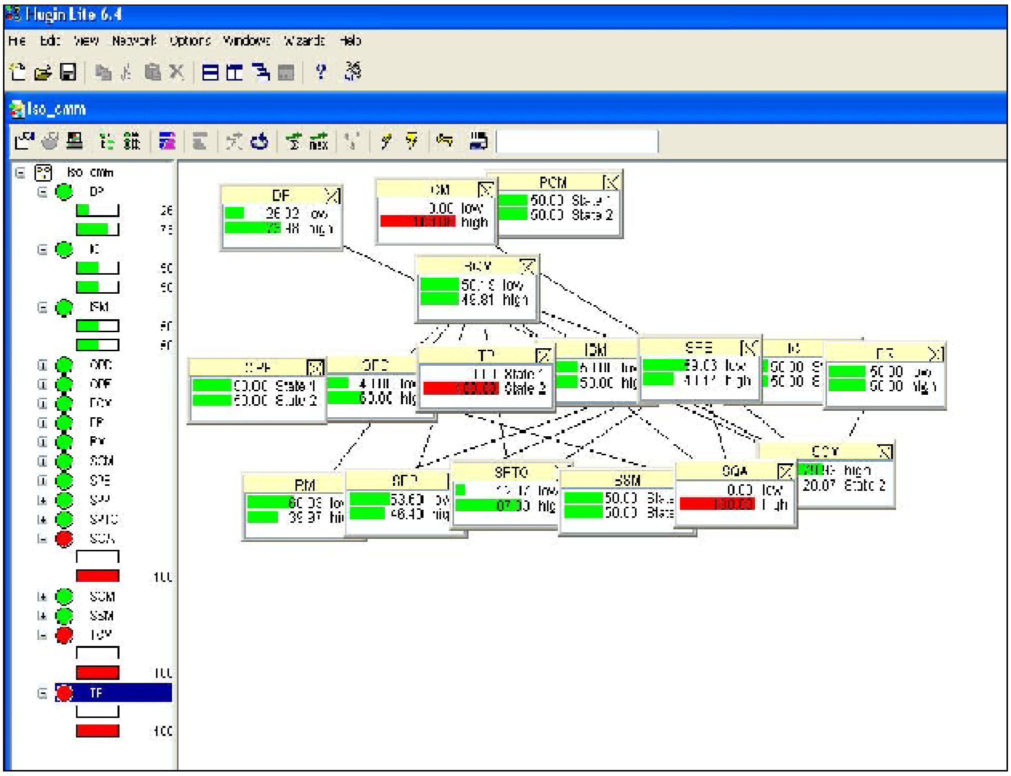

Figure 2 Influence matrix of KPA-dependency according to CMM-ISO mapping

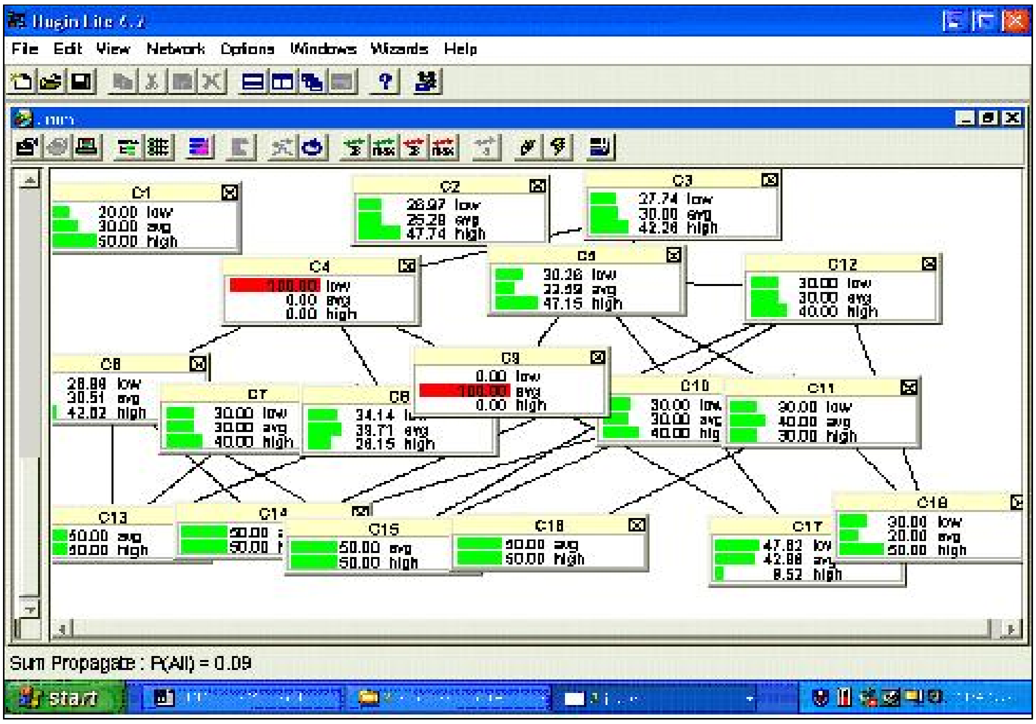

Figure 3: Influence matrix of dependency of KPAs based on [1]

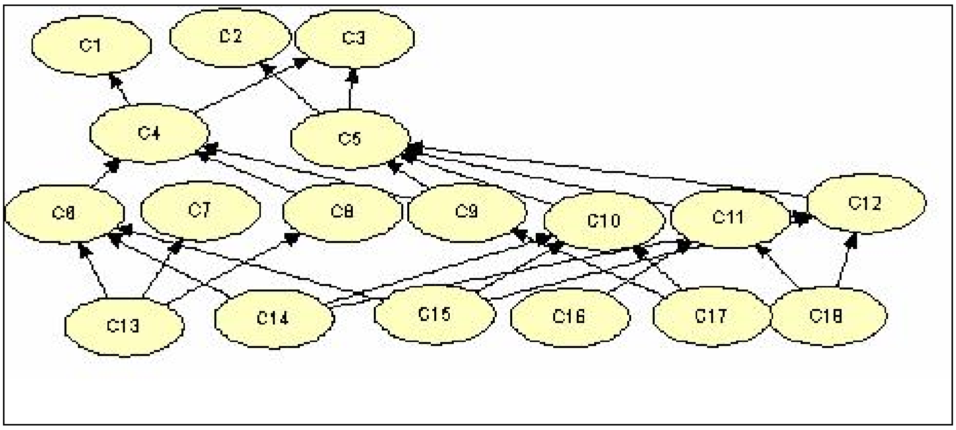

Figure 4 Dependency graph of KPA s of CMM levels

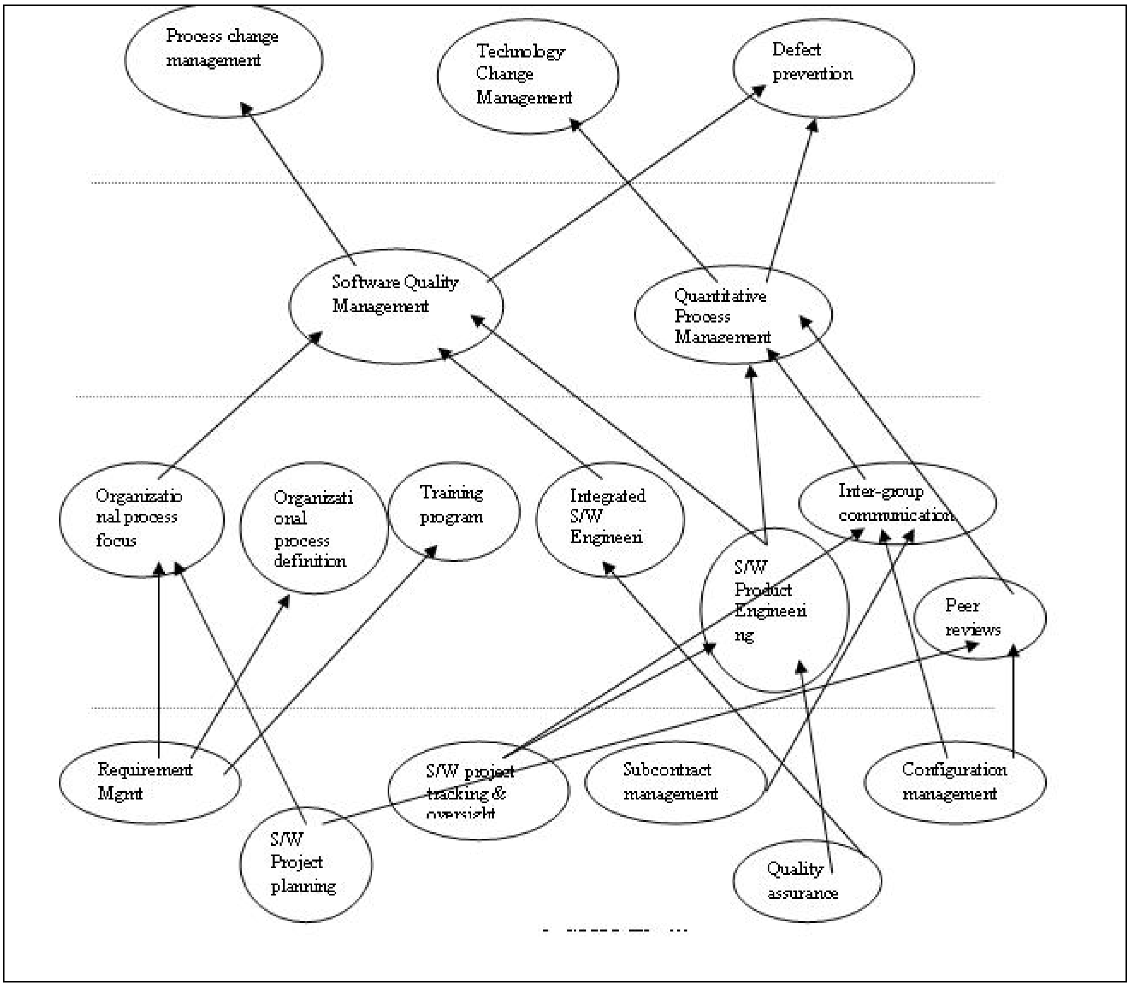

Figure 5 Influence graph of KPA s of CMM levels

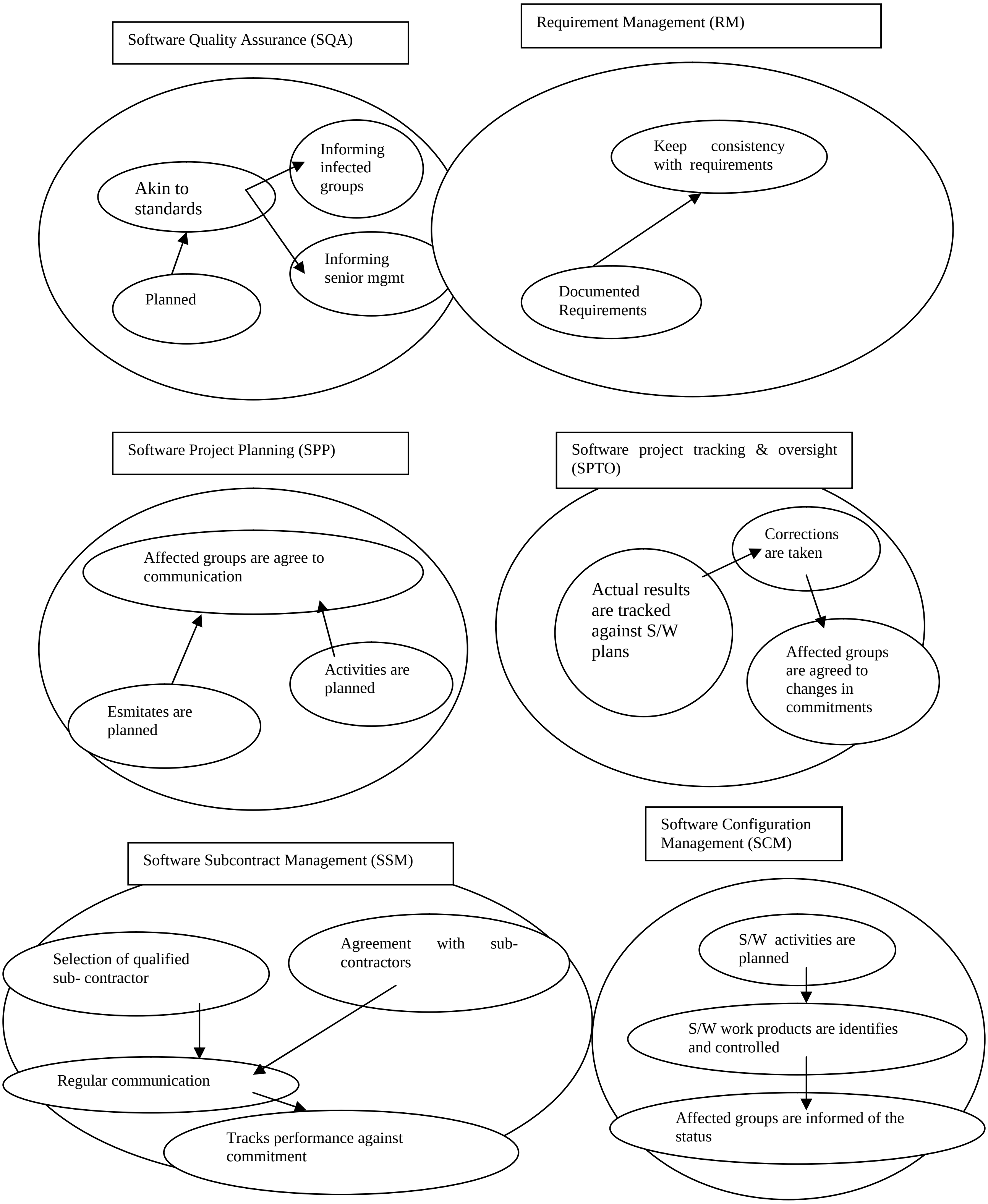

Figure 6: Intra-KPA dependency of Level 2

While performing observation and experimentation, we illustrate dependency graph or influence graph of KPAs relationship based on CMM-ISO 9001 in Figure 1. In Figure 2, we show influence matrix based on it. The simulation is done by HUGIN tool [16]. The influence table shows the quantitative measures of KPAs dependencies based on historic data. Moreover, if we want to figure out the {'low','high'} patterns of any KPA based on definite prediction (like 100% ‘high’ of any KPA), HUGIN also supports corresponding calculations. For example, in Figure 2, we have made TCM, TP and SQA’s influence to 100% of ‘high’ projection. We figure out that the probability of SPTO’s influence to become ‘high’ is now 87%, SPE’s influence to become ‘low’ is now 60%, and so on.

Our subjective analysis of KPAs’ relationship is based on [1] and is depicted in Figure 3 to 5. In Figure 3, we show the influence matrix performed by HUGIN. In Figure 4, we highlight the dependency graph and in Figure 5, the influence graph of abbreviate symbols of KPAs is shown. In Figure 6, we preview the INTRA-KPA dependency, i.e., the goal-dependency of Level-2 based on subjective analysis of section 3. Similar justification may be applied on other levels.

7. Significance of the work & Conclusion:

The significance of this work is any software company can predict the quantitative measurement or projection of its achieved CMM level and related KPAs. With the introduction of ‘probabilistic approximation’, the company can predict how much (%) of any KPA has been achieved and how much (%) does it require further to go up to the next level. It will open up new possibility of Quantitative Capability Maturity model (QCMM) that will assess the standing of process engineering maturity rigorously or mathematically.

References:

[1] Pankaj Jalote, CMM in Practice Processes for Executing Software Projects at Infosys, Pearson Education, India
[2] Mark C. Paulk, SEI ,Questions and answers on CMM, Software Engineering Institute issue1, 5 April, 1994
[3] Nancy Eickelmann, An Insider’s View of CMM Level 5, IEEE Computer Society, 2003, 0704-7459/03
[4] Frank McGarry, Bill Decker, Attaining Level 5 in CMM Process Maturity, IEEE Computer Society, 2002, 0704-7459/02
[5] Donald J. Reifer, XP and the CMM, IEEE computer society, 0740-7459/03, 2003
[6] Shihong huang, cott Tilley, Towards a Documentation maturity Model, ACM 1-58113-696-X/2003
[7] Lisandra V. Manzoni and Roberto T. Price, Identifying Extensions Required by RUP (Rational Unified Process) to Comply with CMM (Capability Maturity Model) Levels 2 and 3, IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 29, NO. 2, FEBRUARY 2003
[8] Christ Vriens, Certifying for CMM Level 2 and ISO9001 with XP@Scrum, In the Proceedings of the Agile Development Conference (ADC ’03) 0-7695-2013-8/03 ,2003 IEEE
[9] Mala Murugappan and Gargi Keeni, Blending CMM and ix Sigma to Meet Business Goals, 0740-7459/03, 2003,
[10] M.C. Paulk, B. Curtis, M.B. Chrissis, et al, Capability Maturity Model for Software, Software Engineering Institute, CMU/SEI-91-TR-24, ADA240603, August 1991.
[11] M.C. Paulk, B. Curtis, M.B. Chrissis, and C.V. Weber, Capability Maturity Model for Software, Version 1.1, Software Engineering Institute, CMU/SEI-93-TR-24, February 1993.
[12] M.C. Paulk, C.V. Weber, S. Garcia, M.B. Chrissis, and M. Bush, Key Practices of the Capability Maturity Model, Version 1.1, Software Engineering Institute, CMU/SEI-93- TR-25, February 1993.
[13] C.V. Weber, M.C. Paulk, C.J. Wise, and J.V. Withey, Key Practices of the Capability Maturity Model, Software Engineering Institute, CMU/SEI-91-TR-25, ADA240604, August 1991
[14] Mark C. Paulk, SEI, “How ISO 9001 compares with the CMM”, 0740-7459/IEEE (January 1995)
[15] People CMM materials, http://www.sei.cmu.edu/activities/p-cmm
[16] HUGIN Expert, www.hugin.com
17] ISO9001, Quality Systems – Model for Quality Assurance in Design/Development, Production, Installation, and Services, Intl. Standards Organization, Geneva, 1987.
[18] ISO9000-3: Guidelines for the application of ISO9001 to the development, supply and maintenance of software, International Standard, 1991.
[19] TickIT: A Guide to Software Quality Management System Construction and Certification Using EN29001, UK Dept. of Trade and Industry and British Computer Society, 1992.
[20] R. C. Bamford and W. J. Deibler II, “Comparing, contrasting ISO 9001 and the SEI capability maturity model”, IEEE Computer, Oct 1993, pp. 68-70.
[21] M. C. Paulk, “Comparing ISO 9001 and the capability maturity model for software”, Software Quality Journal, 2, 1993, pp. 245-256.