1-20-2013

The Key Functions and Best Practices of Software Product Management

Goparaju Purna Sudhakar
purna24@hotmail.com

Follow this and additional works at: http://aisel.aisnet.org/sprouts_all

The Key Functions and Best Practices of Software Product Management

Goparaju Purna Sudhakar
INDIA

Abstract

This paper is basically an overview, practitioners approach and best practices of software product management. The basic functions of software product management, right from product concept, prototype development, requirements management, product planning, planning product lines, product development, product testing, product launch and deployment, product marketing, aligning product management with program management, the relationship of product management with individual project management and product withdrawal are discussed in this paper. This paper high lights some of the best practices of software product management and managing individual software projects. This paper gives a bird eye view of software product management with a block diagram, shows the relationship between product management and individual project management, product roadmap, different releases of software product and their relationships and flow of software product from one version to another version.

Keywords: Software Product Management, Project Management, product strategy, software product development.

INTRODUCTION

The global software product market has significant share in global IT industry. Global nature of software work has changed the way products are designed, developed, tested and delivered to the global customers (Prikladnicki, Audy and Evaristo, 2004). Few global MNCs have succeeded in this area to compare with product development and offshore providers from developing countries such as India, China, Korea, Philippines, and other Asian and Eastern European Countries. The software product market is dominated by many software MNCs from US, Western Europe, Japan and selected companies such as Huawei Technologies from China. Software product development requires investment of lot of time, effort and money and it is a long term investment and can return long term of returns if managed well. Availability of funding, human resources, strategy and long term commitment play key role in software product success. Even after this much effort and investment of time on software products, the life time of software product is hardly three to five years afterwards the product becomes obsolete or entirely redesigned or revamped product has to be launched to keep it alive. Also opening up of the global markets, competition from new and existing players and technological advancements are reducing the cycle time of the products (HBS, 1997).

Even with the lot of maturity of management profession, According to Royce (2008), software development is still unpredictable based on the statistics of software projects success rates across the world. According to the previous research, Seven out of 10 new products that are launched into the market failed (Tripathi, Guin, and De, 2012). Hence, there is importance for the study and make the software projects and products successful. The basic function of *product management* is to plan, monitor and support the software products and product lines (O'Brien, 1993).

Software products are basically two types. They are *custom built products* and *off-the-shelf products* (Buffa and Sarin, 2003). *Custom built* products are specifically developed for selected customers. They are highly customized according to the customer needs, wants and requirements. For example, developing a customized banking product for a specific multinational bank is a custom built product. Whereas, *off-the-shelf* or *readymade products*are developed for general purpose and targeted at wider audience. For example, developing an Anti-virus software product for a PC. However there are general purpose banking products are also available in the market such as Finnacle from Infosys (India). They do even customize the banking product based on customer requirements.

In this paper the entire software product development process, in addition to the role of support functions such as marketing, customer support and sales, and the need for cross functional teams in software product development are explained and highlighted. The basic functions of software product management are as shown in the Figure 1.

Product management starts with product concept (Kotler, 2000), development of a prototype (Pressman, 2010), product strategy, branding, product roadmap, thinking about a single product or product line, how to manage the technical issues, planning different releases, release management, product test management, product development, product maintenanceand product packaging. According to Choudaha (2004), *software product management* combines the

2

disciplines such as development which requires technical knowledge, marketing which requires business knowledge and sales which requires domain knowledge.
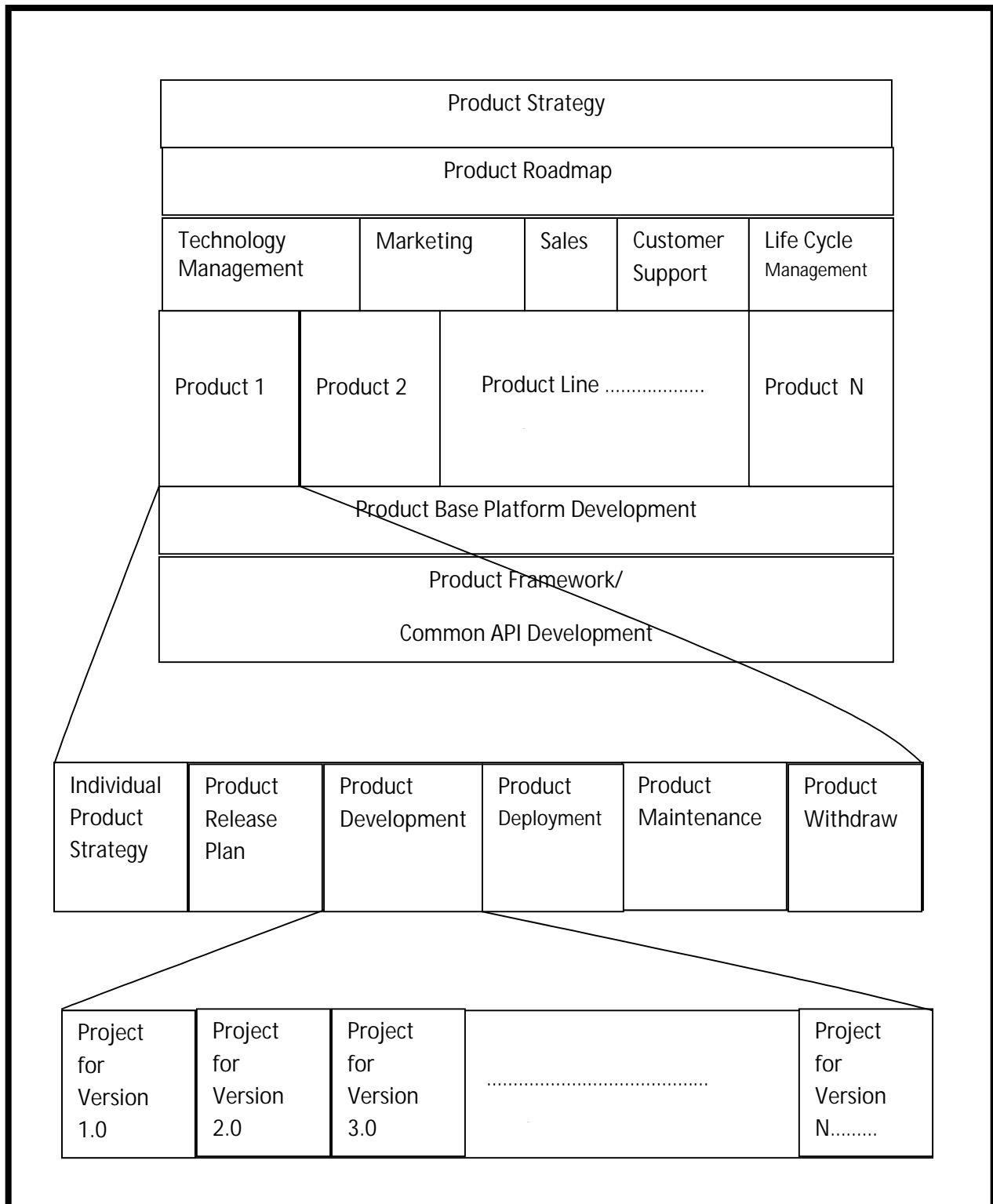
Product Strategy				
Product Roadmap				
Technology Management	Marketing	Sales	Customer Support	Life Cycle Management
Product 1	Product 2	Product Line		Product N
Product Base Platform Development				
Product Framework/ Common API Development				

Individual Product Strategy	Product Release Plan	Product Development	Product Deployment	Product Maintenance	Product Withdraw

Project for Version 1.0	Project for Version 2.0	Project for Version 3.0	....		Project for Version N.........

Figure 1. Software Product Management Block diagram

3

PRODUCT REQUIREMENTS MANAGEMENT

Any software product development starts with collecting business requirements. Usually the business requirements of the product come from the marketing teams or from standards bodies such as ISO, ANSI, OMG and IFPUG. Collecting product requirements is the first step for product development. The requirements collected should be *unambiguous*, *clear*, *certain*, *feasible* and *understandable*. The inherent complexity of software products usually makes the requirements keep changing. Here the product manager has to play a vital role in freezing the requirements for a specific release.

Product requirements management includes the steps such as requirements elicitation, requirements analysis, requirements documentation and requirements review (Kishore and Naik, 2001). During requirements elicitation, techniques such as interviews, brainstorming, literature review, documents review, competing product reviews, trade publications and expert opinions can be used. Once requirements elicitation is over, the requirements are to be analyzed so that they are feasible or not. What is the impact of those requirements on the existing platforms, frameworks and common APIs has to be assessed. Based on the analysis of the requirements, the selected requirements are to be documented and numbered. There should be a common requirements specification language throughout the product line. According to Kishore and Naik (2001), a software product consists of both functional and non-functional requirements.

The documented requirements are to be reviewed by the selected customers, vendors (if any), product development teams, product test teams, product manager and project managers. Once all the stakeholders agree for the requirements, the product requirements are to be freezed. Further modifications to the requirements are to be done by the change management process with the help of CCB (Change Control Board) in the organization by maintaining different versions of the requirements document. According to Kuloor and Eberlein (2002), the requirements management should ensure that the requirements are available, traceable, and requirements change should be manageable.

PRODUCT STRATEGY

Product strategy decides about the kind of product to be launched, what kind of product is it, what is the domain of the product, who are main customers to the product, in which geography it is to be marketed, what can be the price range of the product, who are main competing products for this product and what are their features, what kind of life cycle the product should have, what development life cycle to be followed, what is the entry criteria, what is the exit criteria, when is the time to market, what kind of marketing strategy to be followed, What is the product pricing strategy, what are the discounts and service offerings along with the product, and how long the product has to be maintained, what project development methodologies/process models and project management methodologies to be followed.

Every product, not only software product, but also any other product, under goes certain life cycle. The *product life cyclehas stages such as introduction, growth, maturity and decline* (Kotler, 2000; Porter, 1980). Every product after introduction, its sales grow over a period of

4

time, which is the growth stage, after sometime, the sales remain stable, which is a maturity stage. After some more time, the sales go down, which is a decline state leading to the withdrawal of the product from the market.

The product strategy has to be in line with the business and organizational strategy. The product outcomes should be beneficial to the organizational business with return on investment, profit growth, sales growth, and improve customer satisfaction. Microsoft and IBM product strategies are the best examples of successful and proven product strategies.

Product strategy thinks about what is to be done and how it is to be done from business and high level technical and architectural perspectives. It is about long term thinking and planning for the product and product line.

Product line is nothing but the related family of products being developed based on common platform or framework.

The products in the product line share common APIs (Application Programming Interfaces), common libraries, components, frameworks, and overall based technology platforms such as operating systems and hardware. MS Office product line consists of individual products such as the MS word, MS Excel, MS Power Point, MS Access, and MS Publisher etc.

Software product strategy includes both *technology strategy* and *marketing strategy* for the product. *Technology strategy* thinks and plans about possible operating systems, programming languages, compilers, IDEs, frameworks, libraries, third party components to be used, technology acquisition, bug tracking tools to be used, version control systems to be used, APIs to be developed, interfaces to be given to the external users of the system, user interface, whether stand along application or distributed system, architectural strategies, design strategies, implementation, deployment and test strategies.

Product *marketing strategy* thinks and plans about when to enter into the market, what kind of pricing strategy to be follow, one time license pricing or software as service (SAAS) to be offered, which geography to enter first, who are the target customers, kind of customer service can be offered, how long to market, what is the exit criteria for the marketing, what is the exit criteria for the product, and how much earning targets can be kept and timelines.

Once software product strategy is in place, it is time to have product roadmap.

PRODUCT ROADMAP

The product roadmap gives a blue print of the successive releases planned for the product with timelines. In this stage the set of product requirements can be split and assigned to specific releases and have target dates for respective releases. Each version of the product comes with specific features at specified time. For getting this, each version of the product is treated as a separate project with separate sub-set of requirements. In Scrum project management methodology, the set of product requirements is known as *product log.*

In product roadmap, multiple versions of the product are planned with specific features. Each version is treated as separate project and assigned to a project team lead by the project manager. This specific project team work towards the specific version of the product. For example, one project team works for Ver 2.0, other project team works for Ver 2.5, and another project team works for Ver 3.0 (Figure 2). Sometimes some of the intermediate releases such as Ver 2.5, ver 2.9 are merge with main Ver 3.0.
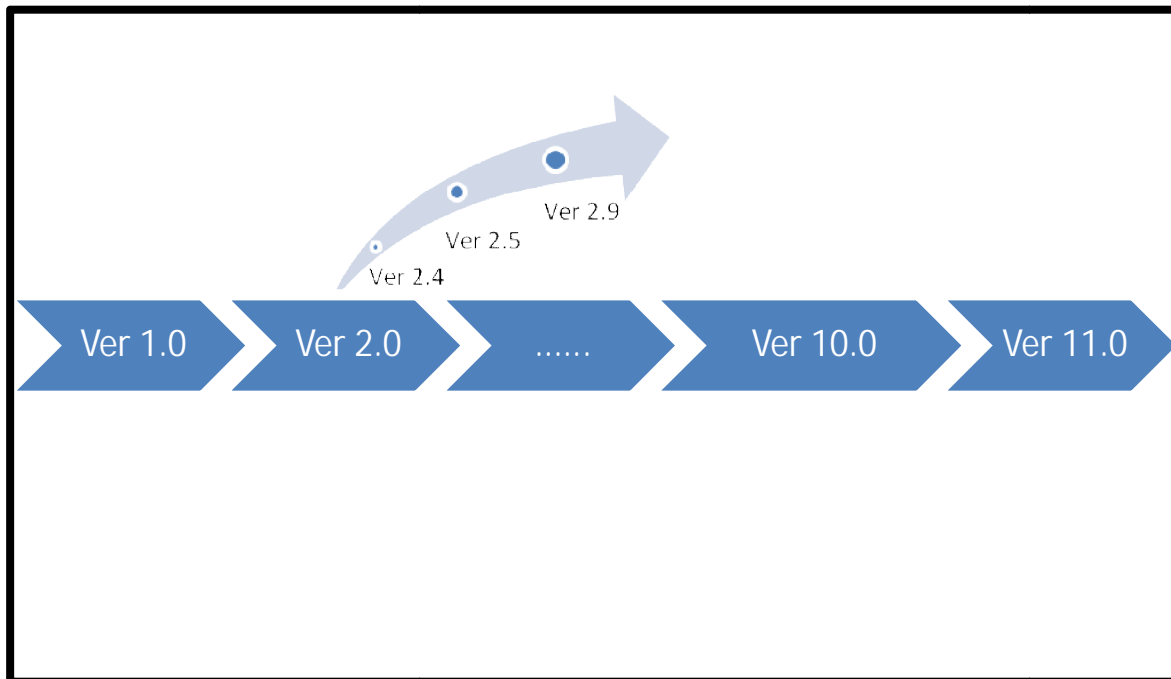

Figure 2. Product Roadmap

Product roadmap also thinks about the multiple releases, branching, merging and labeling strategies for the product in the version control system. Minor releases are branched out from main versions and later merged with main stream. This is the usual practice when dealing with multiple versions of the product. Each project team works dedicatedly towards the specific version release of the product. Which features go into which release are decided in product roadmap meetings. Are there any interim bug fix or hot fix releases or customer specific releases, if so, plan for those specific releases is also done in product roadmap discussion.

DEVELOPING THE BASE PLATFORM/FRAMEWORK

The first step in product line development is the development of common technology platform needed for the family of products. Microsoft Dot Net, MFC, COM/DCOM, OLE are examples of frameworks, components and libraries useful for the development of product lines. Base technology platform is the interface between different products and operating system. If an application in the product wants to interact with the operating system, the communication has to go through the common platform (Figure 3).
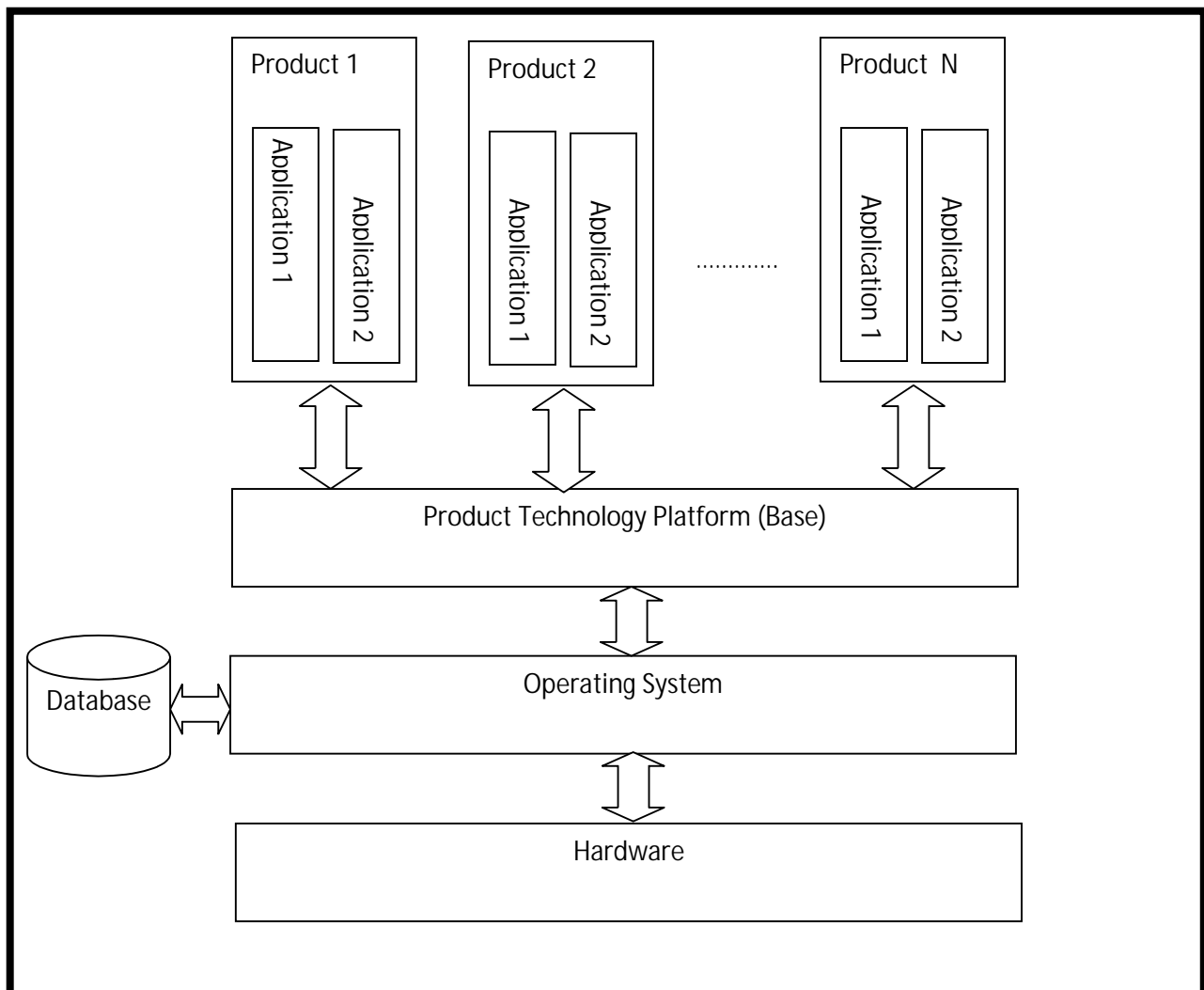
6

Figure 3. Product Base Platform and Its Interfaces

For example, in distributed systems development, ORBs (Object Request Brokers) work as common platforms for the interaction between different applications. In case of standalone applications, these base platforms reside on top of Dot Net framework or JVM (Java Virtual Machine). The common functionality in all products or the needed common functionality of the product line goes into the base technology platform. The APIs, components, and libraries provided by the common platform can be used in development of individual products. This increases the reusability of libraries, components and APIs throughout product line development. It also saves the lot of development and test time for the products. The common APIs can also be given for users/customer programmers to interact with or to customize their applications in the product.

7

PLANNING THE PRODUCT LINE

Usually product development organizations split the features into different similar products because of several reasons. The reasons can be development cost, product price, heavy functionality in single product, maintainability and testability, ease of operation for the user, to simplify the product complexity, time takes to develop the entire product line, and competition in the market. For example an investment bank product line can have different trading products such as Swaps, Options, Cap/Floor, Bonds, FRA, shares and mutual funds as separate individual products. Similarly a banking product line can have personal banking and corporate banking products separately.

Usually all the products in the product line belong to the same business domain. They all use the same base technology platform and libraries in development. Architecting product lines can be done at enterprise architecture and domain architecture levels. For example, IBM has a product line of MQ Series and MQ Series Integrator. Similarly, Autonomy, an HP Company earlier known as Interwoven has product line with little distinct products such as WorkSite, TeamSite, LiveSite, etc.

It is best practice to design and implement a product in the product line successfully and then proceed to develop the other products in the product line so that the needed common APIs will be developed during the course of a product implementation. Every product in the product line has to be planned, have documented requirements, architected and designed, implemented, and tested independently and also together as product line on the same platform and on different platforms. Microsoft Project Professional, which is part of Microsoft Enterprise Project Managementsolution and CA's Clarity can be used for planning product's projects and portfolio of projects.

PROTOTYPE DEVELOPMENT

Once documented requirements for a product are available, to have the proof of concept it is best practice to develop a software product prototype. Usually prototypes are of two types. They are 1) Evolutionary prototypes and 2) Throwaway prototypes (Pressman, 2010). *Evolutionary prototype* development starts by taking few important critical requirements of the product and implement it. Once these critical requirements are implementable, keep adding some more requirements on an incremental basis periodically and finally this prototype becomes the fully functional and fully fledged product. The other type of prototypes is *throwaway prototype*. Throwaway prototypes are developed using few critical requirements. Once proof of concept is done, these prototypes are thrown away and not used any more. The entire product from scratch with all requirements with proper life cycle phases is developed.

For software prototype development many Rapid Application Development (RAD) tools are available in the market such as Microsoft Visual Studio, JBoss, IBM Rational Rose (both design and code generator). Using Microsoft Visual studio one can design the GUI (Graphic User Interface) very easily. Prototypes guarantee that the planned product or product line is feasible and implementable without any technical risks. They assure both technical and economic feasibility in product development.

The selection of an individual project from the list of projects has to be based on its alignment with organizational strategy, business strategy, product strategy, profitability, probability of technical and commercial successes, and competitive advantage (Cooper, 2007).

Next section deals with product development.

PRODUCT DEVELOPMENT

Once proof of concept is done with the development of software prototype, it is time to develop the individual product. The product development involves execution of multiple projects spanned at different times over the life cycle of product. Each project which is spanned in the life time of product has to be initiated, planned, executed, monitored, controlled and closed (Figure 4). Here comes the professional *project management* into *product management*. The advancements in software design methodologies from structured to object oriented have improved the productivity of product design process and also improved the quality of product design (Fox and Koskela, 2008).
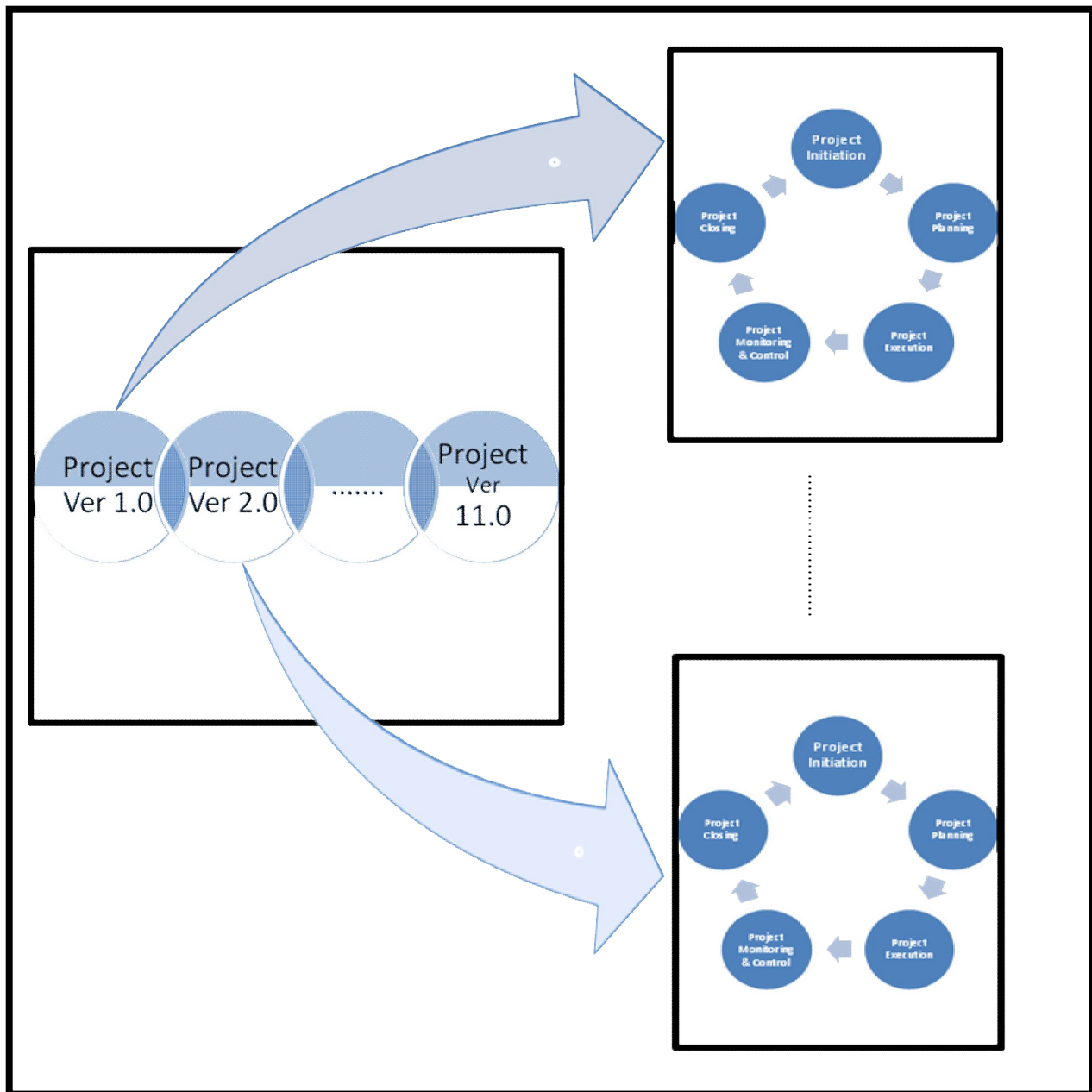
Figure 4. Product Development Vs Individual Project Management

In each of the project execution stage, certain product deliverables such as design documents, source code and test cases come out. With the help of multiple projects, each project making a specific release of the product finally brings the stable product into the market. Each release may consist of additional new features of the product or it may be the bug fixes of earlier versions. The bug fixes and additional features finally make the product error free and fully functional and stable. The customer needs to install the upgrades of the new versions of the product.

Every individual project undergoes the cycle of requirements, analysis, design, coding and testing during development. The product development teams face vey dynamic environment in their day to day work, which can impact the project task or the project work products (Gluesing

10

and Gibson, 2003). Hence, in each project, during development, reviews such as requirements reviews, design document reviews, test plan reviews, quality reviews, code reviews, deployment reviews, estimations reviews and training plans reviews should take place (Jones, 2004). This is very much required to ensure the quality of the project product. The individual project management play vital role in overall product management. The role of project management in product management is discusses in the next section.

ROLE OF PROJECT MANAGEMENT IN PRODUCT DEVELOPMENT

The individual project management can be done by initiating, planning, executing, monitoring and controlling, and closing the project (Sudhakar, 2010). During *Project initiation* step,project kickoff meeting happens and all the stakeholders are informed about the project scope, objectives, purpose, initial timelines, budget and risks. The requirements for that project are also collected during project initiation. Once documented requirements are available, next step is *project planning*.

In *project planning*, the project work breakdown structure is created, project tasks are identified and decomposed, task dependencies are identified, critical path is found out in the network diagram, schedules are made, time estimates, cost estimates, resource estimates such as hardware, software and human resources estimates are made. Usually project management tools such as Microsoft project, Primavera and CA Clarity are used for project planning and scheduling purposes. For effort estimation, Function Point (FP) analysis and/or KLOCs (Kilo Lines of Code) techniques are used. The estimates follow the following path (Figure 5).
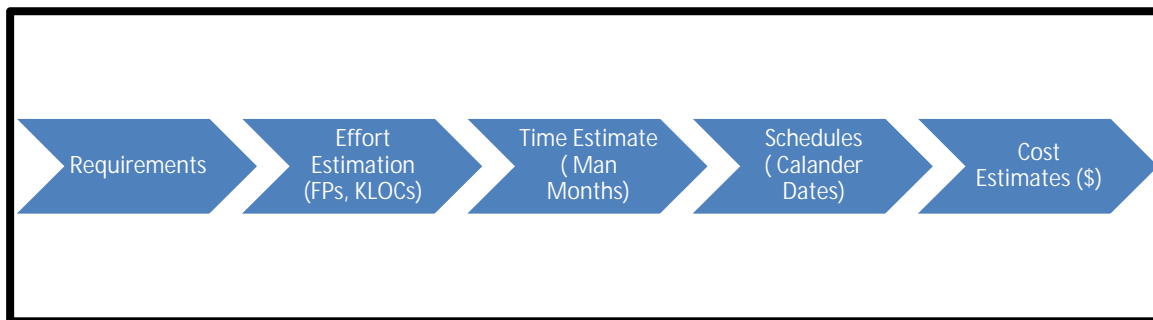

Figure 5. Different types of Estimates

Based on the estimates, the project manager makes the time and cost estimates, schedules and timelines for the project. Then, the project manager has to execute the project according to the project plan. In *project execution*, the planned tasks are executed by the project team members. During this step, the project manager facilitates and provides needs resources for the project. He or she also makes the team full fledged and the team consists of the maximum team size for this duration of the project.

Next step would be to monitor the progress and control project and manage human resources. For controlling project, different project control tools such as traceability matrix, earn value analysis, balancing scope, time, quality and cost need to be done. Scope management, change management needs to be done using this step. The programmers performance needs to be

11

evaluated againstalready set objectives and corrective actions need to be taken. Corrective and preventive actions are needed not only to people but also to projects. Based on the schedule delays, cost delays or project risks the project manager has to take corrective actions in addition to regular risk management. All these things happen in *monitoring and controlling* step of the project management.

Once the project product, that is, specific version of the product is developed and tested, it is time to close the project. That is *project closure* has to happen. The product has to be handed over to the customer support and marketing teams. The project postmortem has to happen, lessons learned are to be documented and the project best practices have to be propagated across the organization or the division. Contracts, if any, have to be closed properly with customers, vendors and suppliers.

All these things are to be repeated for each project of the product. Then the product manager moves his concentration to other projects and upcoming releases of the product.

SOFTWARE PRODUCT TESTING

The main objective of product testing is to deliver the bug free product to the customer. Each project product has to be tested before marketing or delivering to the customers. The project product has to be tested against the requirements implemented. Here different types of testing such as functional testing, stress testing, performance testing, usability testing, peak load testing, boundary testing, failure testing can be done to the product. The different types of testing which can be done during the product development are unit level testing, integration testing, system testing, and interface testing. According to McGregor (2001), software product testing should be done at static level, dynamic level and with regression test strategies. Static level testing includes the reviews, inspection, architectural and design evaluations. Dynamic level testing includes unit, integration and system testing.
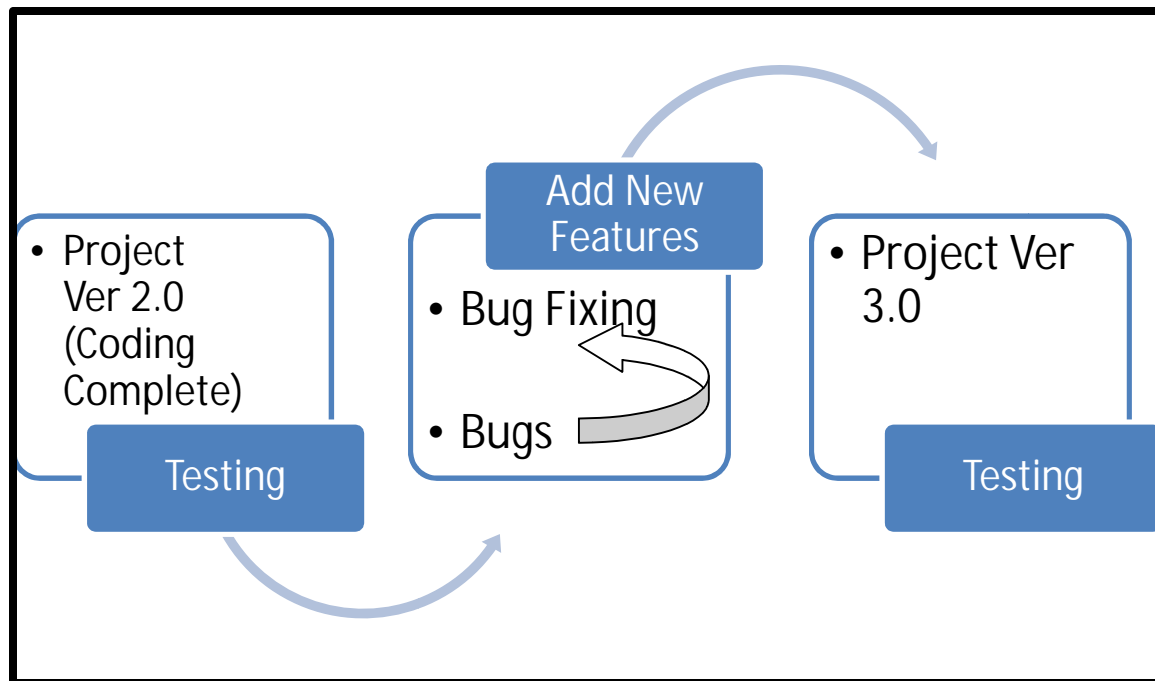
Figure 6. Testing Different Versions of the Product

Also testing techniques such as black box testing, which tests only the input and output and white box testing which tests the internal data and control flow of the system are widely used for product testing purposes. Usage of test automation tools such as WinRunner, Silk Test, Load Runner, HP Quick TestPro (QTP,), HP-Quality Center and JUnit can increase the productivity and reduce the product development time. There are bug tracking tools as well to track individual bug and its progress/status. Similarly Alpha testing is done at the development organization and Beta testing is done at the customer sites. Usually before the product launch into the market, the product development organizations take feedback from selected customers by installing their beta version of the product at client site. This helps in making the product more stable and increasing the quality of the product. Also the users accustom to the product and they are trained in the product before they actually purchase the product.

As the product is maturing the number of bugs detected per release should reduce. It indicates that the product is becoming stable (Figure 7).
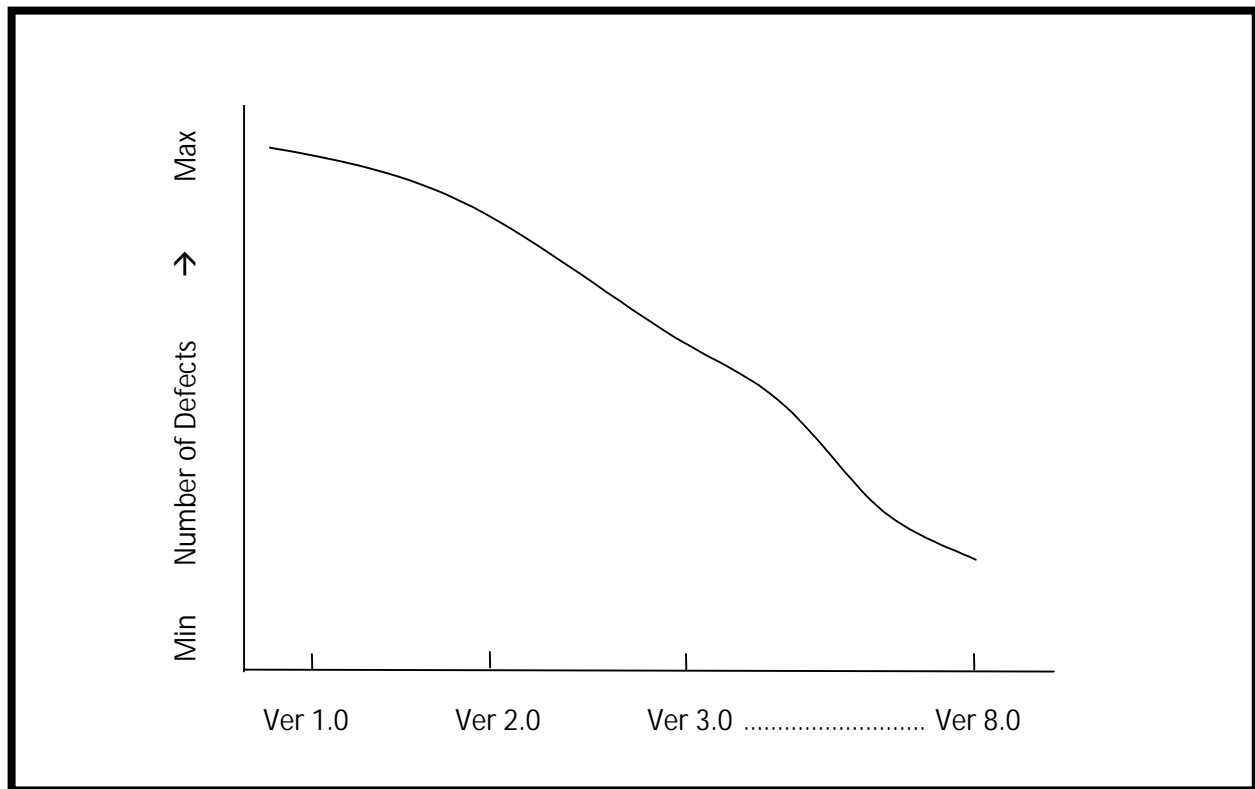
13

Figure 7. Product Releases Vs Number of Defects

One should make sure that the coding bugs are found at unit test level itself, otherwise finding bugs and fixing them is costly in later stages of testing the product. Similarly, the bugs found in initial phases of project such as requirements and design are easy and less expensive to fix compare with fixing them in later phases of project development. That is, it is costly to fix the defect once the product is delivered to customer or once it is in acceptance test phase to compare with fixing the defect in design phase or coding phase. Research indicates that, it is 60 to 100 times more costly to fix a bug after product release to customers than to fix it in design phase (Keith, 2002).

It is easy to maintain the product, if the occurrence of defects is less and also the development processes of the product are well defined. There is relationship between defined processes and the quality of software product. Hence, the quality emphasis should be there in overall product development time. Some of the product development organizations have product verification and process verification. Product verification tests its functionality and make sure that it meets the requirements. Whereas process verification is checking the development processes used to develop the product. Quality process audits also check the maturity of development and test processes used to develop the product. Majority of the product develop setups have dedicated QA/Testing teams. Testing teams operate at the project or product level and the QA team operates at the organization level verifying the development processes followed by different teams in the organization.

14

PRODUCT DEPLOYMENT AND LAUNCH

Once the product is beta tested, it is time to deploy the product at customer sites. For deploying the product at customer site, customer has to have the needed computational environment such as hardware, software and tools. The product support team sets up the environment at customer site, installs and configures the product. They make the trail run of the product at customer production environment. They even provide the support at production environment of the customer by fixing any bugs or showstoppers.

The support for the product can be provided at 3 to 4 levels based on the criticality of the customer problem or issue. The levels differ in the time they take to respond, the severity of defect, and the depth of technical work involved. If the customer issue is not solved at lower levels it gets propagated to the higher level support teams. This issue can become a bug in the product or a suggestion from the customer. Product installations and support is the responsibility of technical support/customer support teams.

It is best practice to install the product initially at few customer locations, take the feedback and later have wide customer installations. The products like Microsoft Office have millions of installations across the globe. It is the engineering team's responsibility to package the software and make an installer for the product. This makes the job easy for customer support teams in installing the product.

PRODUCT MARKETING

If the marketing starts along with product conception, the marketing costs increase. Hence, the product management has to start the marketing at right time by observing the time to deliver the product. Marketing can be done geographically, mass marketing, or viral marketing. This has to be decided based on the nature of the product and target customers. The objective of marketing is to spread about the product, achieve competitive advantage, and to increase sales. According to Kotler (2000), *marketing management* is about conception, product promotion, distribution of products and services in meeting individual, departmental and organizational goals. According to Chandran (2007), the changing face of marketing is changing the way organizations are functioning, organizational strategies and the product strategies.

In some organizations, marketing is also responsibility of product management team. For example in Organizations such as Siemens Medical Systems, product marketing involves lot of technical marketing. The people with technical background are moved to the marketing or product management roles. This is because of the complexity of the products. Marketing involves advertising, participating in trade shows, product launch events, symposia, making presentations in seminars and conferences, etc. The effective marketing shows the increase in sales. That is the marketing function should show the return on investment.

Current day software product marketing involves knowing the in depth technical details of the product, architecture, features, domain and technologies. Also the marketing engineer should know the competing products in the market, what are their features, price, and how his or her product is different from the competitors' products. He should be able to show the distinction of his product. *The main objective of marketing is to get customer satisfaction*. According to

15

Steinhardt (2004), the product going to be launched should be market driven, that is, the development organization should listen to the customers and find out their wants and needs. Also the main principles of product marketing include finding the needs, wants and expectations of the customer (Kotler and Armstrong, 1996). Once the team gets the sales order for the product, they pass the order to customer support team for the installation of the product at the customer site.

PRODUCT MAINTENANCE

The product maintenance is done by the engineering teams. The main duty of maintenance team is to fix the defects, add new features to the product and make new releases or specific releases to selected customers. Usually product maintenance teams make intermediary releases, bug fix releases, minor releases or hot fixes to the product. Making the major releases still remains as the responsibility of the development or engineering teams. Or within engineering team, some members will be working on maintenance projects and some will be working on major release projects.

A product reaches the maintenance stage once it gets deployed at the customer sites. Well defined processes (Sriram, 1997), proper team climate while development, review mechanisms, higher team productivity and early detection of defects make sure that the maintenance costs are at minimal. It is easy to maintain a good quality product rather than a bug full product. Otherwise maintenance costs zoom up. Usually product development organizations such as IBM, Microsoft, and ORACLE outsource product maintenance activities or have vendors to do this work. The product has to be maintained and supported till it gets withdrawn from the market. Usually product development organizations give some grace time to customers to upgrade their product to the latest versions and they mention the time limits they are going to support the older versions in the release notes of the new versions. Sometimes they even deprecate some of the features of older versions in new versions. This is communicated using product releases notes. The product release notes should consists of all these details in addition to highlighting the new features, mentioning known defects, deprecated features, time limits to support older versions and contacts in case customer needs support.

ALIGNING PRODUCT MANAGEMENT WITH PROGRAM MANAGEMENT

Program Management is nothing but executing the related projects together in a beneficial way so that the benefits might have not come when these projects are executed separately. It is very tricky to interlink the program management with product management because all the projects under the product are any way interrelated. One way to distinguish these is, combine some of the interrelated projects such as Ver 2.0, Ver 2.5, Ver 2.8 and make it as a program under product management. Similarly another program for Ver 3.0 and its intermediary releases.

In some organizations the program manager, himself owns the entire product development team and he will be supervising all the projects of the product and managing multiple project managers under him. Basically this distinction varies from organization to organization. The main objective of program management is to get financial, organizational, cost, schedule and

16

resource benefits and efficiency by executing projects together. Also the risks and dependencies can be managed well in programs.

The program management has the processes such as capacity management, competency management, problem management, logistics management, PMO (sometimes), change management, supervise the work and quality management (Rai and Subramanian, 2007). The objective of program management is to integrate and manage the mutually dependent projects to deliver the overall software product (Martinelli and Waddell, 2007).

PRODUCT WITHDRAWAL

After few years of existence, every product, not only software product but also any other product in the world gets withdrawn from the market. It ceases its existence. The drastic reduction in sales, customer dissatisfaction, poor quality of product and outdated technologies can lead to the product withdrawal. The organizations can use the lessons learned, customer base, knowledge, skills, and experience gained for the future products of the organization.

Product withdrawal leads to releasing team members, closing projects, closing contracts and closing administrative activities. Sometimes it may even lead to shutting down the organizational operations. To avoid this, the organization has to get competitive advantage (Porter, 1980) by making different regular versions of the products, variety and portfolio of products, different pricing and discount strategies and product distinction.

CONCLUSION

Product management plays vital role in software organization. This is evident from organizations such as Cisco, IBM, HP, DELL, ORACLE and Microsoft. It is yet to see the organizations from developing countries succeeded as their western counter parts in products area. These developing countries mastered in providing low cost, high quality offshore services with the advantage of reduced product development time for their customers across the world.

It is best practice to distinguish between product management and individual project management and to have separate managers to handle these responsibilities. However, they are mutually interdependent functions. Product management has got bigger scope to compare with individual project management.

As explained product development and product management is very much cross functional and makes the product development teams more innovative. That is the reason the product development should bring in people from different streams and business functions to get advantages of product distinction, innovation, productivity and customer satisfaction. *Software product development involves lot of creativity, innovationand teamwork.*

REFERENCES

17

1. Buffa, E.S. and Sarin, R.K. (2003)*Modern Production/Operations Management,*Eighth Edition, John Wiley & Sons, New York.
2. Chandran, M.P. (2007) "Marketing: The Changing Face",*ISB Insights,*September 2007, pp 4-5.
3. Choudaha, R. (2004) "Software Product Management". A White paper available online at http://articles.softwaremarketingresource.com. (accessed on 08-Oct-2012).
4. Cooper, R.G. (2007) "Managing Technology Development Project",*IEEE Engineering Management Review,*35(1), pp. 67-76.
5. Fox, S. and Koskela, L. (2008) "Improving the reliability and validity of results from multinational research projects",*Project Perspectives,*XXIX(2008), pp. 72-76.
6. Gluesing, J.C. and Gibson, C.B. (2003)"Designing and Forming Global Teams". In Maznevski, M. L., Mendenhall, M. and McNett, J. (eds.). *Handbook of Global Management: A Guide to Managing Complexity,* Blackwell, USA.
7. HBS. (1997)*Project Management Manual,*A manual of Harvard Business School, Boston, MA, Doc# 9-697-034, October 6, 1997, pp. 1-42.
8. Jones, C. (2004) "Software Project Management Practices: Failure Versus Success",*The Journal of Defense Software Engineering,*17(10), pp. 5-9.
9. Keith, E.R. (2002) "Agile Software Development Processes: A Different Approach to Software Design". A White paper available online at http://cf.agilealliance.org/articles/system/article/file/1099/file.pdf. (Accessed on 08-Oct-2012).
10. Kishore, S. and Naik, R. (2001)*Software Requirements and Estimation.* Tata McGraw-Hill, New Delhi.
11. Kotler, P. (2000)*Marketing Management: The Millennium Edition,* Prentice-Hall of India Private Limited, New Delhi.
12. Kotler, P. and Armstrong, G. (1996)*Principles of Marketing,* Prentice-Hall, UpperSaddle River, New Jersey.
13. Kuloor, C. and Eberlein, A. (2002) "Requirements Engineering for Software Product Lines",*Proceedings of the 15th International Conference on Software & Systems Engineering and their Applications (ICSSEA, 2002),*December 3-5, 2002,Paris, France.
14. Martinelli, R. and Waddell, J. (2007) "Program Management: It's about the Business!",*Project Management Practice,*2007(1), pp. 12-15.
15. McGregor, J.D. (2001)*Testing a Software Product Line,*Technical Report of Software Engineering Institute, Carnegie Mellon University, Report# CMU/SEI-2001-TR-022. pp. 1-68.
16. O'Brien, J.A. (1993)*Management Information Systems: A Managerial End User Perspective,*Second Edition, IRWIN, Boston, MA.
17. Porter, M.E. (1980)*Competitive Strategy: Techniques for Analyzing Industries and Competitors,* The Free Press, New York.
18. Pressman, R.S. (2010)*Software Engineering: A Practitioner's Approach,* McGraw-Hill, New York.
19. Prikladnicki, Audy, R., Jorge L. N. and Evaristo, R. (2004)"An Empirical Study on Global Software Development: Offshore Insourcing of IT Projects",*Proceedings of the 3rd International Workshop on Global Software Development,*May 24, 2004,Edinburgh, Scotland, pp. 1-6.

20. Rai, V.K. and Subramanian, K. (2007) "Program Management Design: A Viable System Perspective",*A White Paper of Tata Consultancy Services (TCS) Limited*, India, pp. 1-13.

21. Royce, W. (2008)*Software Project Management: A Unified Framework,*Pearson Education, Inc, USA.

22. Sriram, J. (1997) "Software Engineering, An Operational Perspective". *Proceedings of the 31ˢᵗ Annual Conference of the Computer Society of India,* 30 October – 3ʳᵈ November 1996, Bangalore, India, Tata McGraw-Hill, New Delhi, pp. 395-402.

23. Steinhardt, G. (2004) "Who's Driving your Company?",*productmarketing.com*, Mar/April, 2004, pp 17-20.

24. Sudhakar, G.P. (2010)*Elements of Software Project Management,* Prentice-Hall of India Private Limited, New Delhi.

25. Tripathi, S.S., Guin, K.K. and De, S.K. (2012) "Critical Success Factors for Marketing a new Product: An Empirical Investigation". *The Indian Journal of Management,*5(2), pp 3-15.

About the Author

Goparaju Purna Sudhakar, PMP with over a decade of experience in IT industry is currently working as Consultant and Visiting Faculty based in Hyderabad, India. Earlier worked as **Sr. Faculty and Head (In-Charge),** IT Division at **Engineering Staff College of India**, Hyderabad and **Consulting Editor** at **ICFAI Research Center**, The ICFAI University, Hyderabad, India. He worked in **USA, UK, Ireland, Finland and India** as **Software Professional**. He has **B.Sc., MCA, M.Tech. and Executive MBA**. He is Project Management Professional (**PMP®**) certified from Project Management Institute (PMI), USA. He has submitted his **PhD** thesisin Business Administration to Aligarh University (AMU), India. He is member of All India Management Association (AIMA). He is also visiting faculty to several institutions in India.

He worked as employee or consultant to companies such as IBM, Siemens, Interwoven, Wipro Technologies, Citicorp, Nokia, Salomon Smith Barney, SIAC, DSET Corporation, IONA Technologies, Birla-Horizons International, and PCL Mindware. He held both managerial and technical roles in IT industry. He worked as Project Manager at Citicorp Overseas Software Ltd(COSL), Hyderabad during 2001-03. He worked as Systems Manager at Wipro Technologies during 1999-01. The reuse domain software product he managed, *Scorpus* was identified as one among the top 100 IT innovations by NASSCOM (2007).

He worked in Banking, Financial Services and Insurance (BFSI), Telecom, E-Commerce, Enterprise Content Management, Enterprise Software Asset Management (ESAM), Health Care and HR domains. He has extensive experience in project management, program management, general management, quality management, business process re-engineering, HRM, and human aspects of project management. He has wide experience in management consulting.

He has **60**published articles. His articles were published in The Hindu, Indian Management, Businessgyan, Business & Management, PMresearch (IPMA, Switzerland), *Team Performance Management* (Emerald, UK), *Acta Universitatis Danubius. Economica***, Scientific Annals** *of the*

*"Alexandru Ioan Cuza" Univesity of Iasi: Economic Sciences***,** *The annals of the Stefan cel Mare University of Suceava* (Romania), *The Serbian Journal of Management*, The Global Educator, Express Computer, Projects and Profits, www.pmhub.net, Computers Today and CSI Communications. He has authored or edited 14 books. He authored ***Business Essentials for Software Professionals* (Excel Books, 2008),** ***Project Management: Training Manual* (Akansha Publishing House, 2010),** ***Elements of Software Project Management* (Prentice-Hall, 2010),** ***Global Organizational Behavior* (**LAP Lambert Academic Publishing, Germany, 2012**)**and***Project Management FAQ* (Galgotia, 2012)**and **edited 10 other books**. He has written about Project Management, Human Aspects of Project Management, General Management and IT industry.

He is on the Editorial Board of International Journal *Acta Universitatis Danubius. Economica* (Romania) and International Journal *Revista de Gestão & Projetos – GEP,* a Brazilian Journal in Management and Projects. He is also an article reviewer for peer reviewed International Journal *Team Performance Management* (Emerald, UK, 2011).

He worked as Executive Vice President and President of VSPGS alumni association. He is recipient of Government of India Scholarship for 18 months during his M.Tech (Computer Science) at Devi Ahilya University, Indore, India. He has done an academic project at National Informatics Center (NIC), Hyderabad during 1993-94.

In his spare time, He spends with his family. He is married with two daughters. His hobbies include photography, gardening, reading, writing, and seeing places.

20