59

Process Oriented Instruction for Software Engineers (POISE): Perspectives and Experiences

Srinivas Thummalapalli and Rajat Kumar Todi

Centre for Development of Telematics, 11/6B Shanti Chambers, Main Pusa Road, New Delhi - 110 005, India.

Abstract

This paper briefly describes the training model for imparting the process view of software development to the graduate engineer trainees in a Level 2 telecom research and development organization. The design of the approach was predicated on the concepts of ETVX model for software development. Its structure and features were designed to align the training objectives with the organizational goals.

It provides an insight into the operational aspects and their underlying objectives. It provides structured learning experience to the trainees and enable them to assimilate, appreciate and apply the concepts of software engineering to the software development.

It narrates the difficulties after implementing the approach and proposes requisite measures to make the approach responsive and vibrant. The paper also refers to similar such exercises in the academic arena and the relevance of some aspects in the organizational environment.

1. INTRODUCTION

Centre for Development of Telematics (C-DOT) is the telecommunications technology research centre of Government Of India. C-DOT's mission, its products, and its future perspective can be found from [9], [11]. C-DOT is a level 2 organization [6], [10]. Its software quality objective is to migrate to level 3 on SEI CMM [10] and to attain ISO 9000 certification for its software development process. It follows the spiral model [1], [6] for the development of its software products.

C-DOT recruits fresh engineering graduates (GETs) and engineers who have at most one or two year's of experience. GETs were from Electrical & Electronics Engineering (ECE) and Computer Science (CS) disciplines. Detail of first three batches of GETs are given in table 1.

It was felt that GETs require an introduction to software engineering, and to software development process model that is being followed in C-DOT. They should be exposed to C-DOT's software environment. Other desirable and essential skills are like communication, project   management, team-work and quality consciousness. The aforesaid needs are realized through Software Development Methodology workshop (SDM w/s).

Table 1
Attributes of GETs batches

	GET 1	GET 2	GET 3
Total Number	42	35	64
CS	35	24	38
ECE	7	11	26
No. of Teams	14	16	22
No. of Mentors	7	8	11
Trainees/Teams	3	2-3	2-3

2. SOFTWARE DEVELOPMENT METHODOLOGY WORKSHOP

The accent of the workshop is on making the trainee to appreciate:
- significance of process in the successful realization of the product.
- bearing of individual effectiveness on the quality software development.
- interplay of persons and processes to develop real-time software solutions.
- constraints and compulsions of real time software working environment.

The structure and features of SDM w/s with its underlying intent was also presented. For the detailed exposition refer to [3].

2.1. Structure

1. Task Force: A task force (TF) on SDM w/s was constituted to implement the approach. It consisted of technical group (TG) and process group (PG). Technical group consists of mentors who were engineers from switching software division and process group consists of members from technical training division.

2. Selection: Members of both the groups were nominated by their respective divisional managers. A typical mentor has a working experience of at least 1.5 years. The details of the earlier batches are presented in table 1.

3. Teams: A team (TR) consists of 2 to 3 trainees. It was formed by the process group by considering the following factors viz. educational qualifications (whether from CS or ECE), qualification level (undergraduate or graduate) and gender, disposition of individuals (whether an individual is extrovert or introvert).

It is a conscious attempt to re-transform existing groups into new associations. The experience with three batches of GETs (refer table 1) corroborates that the criteria for composition of successful teams in the arena of the academic world [8] are equally applicable in an industrial environment.

4. Roles: The roles and responsibilities of technical group, process group, and trainees were clearly defined and documented for easy reference [5], [14].

5. Communication: The nature of communication between different groups is depicted in figure 1. Brief about the different channels is given below.
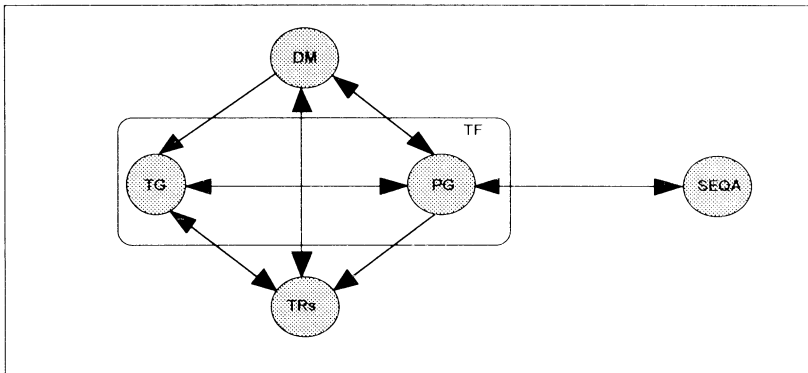
FIGURE 1: COMMUNICATION MODEL

Communication between:

- Technical Group and Process Group: to select assignments, to modulate the approach, and to track the progress of the trainees during the workshop.
- Process Group and Divisional Manager (DM): to update the status of progress, and to seek resource allocation.
- Technical Group and the teams: to help the trainees to solve their conceptual difficulties.
- Process Group and the teams: to give and receive the feedback for continuous process improvements, to make the trainees aware of process.
- Divisional managers and trainees: to convey the organizational vision and seek the first hand feedback from trainees.
- Process group and Software Engineering and Quality Assurance (SEQA): to identify the leverage areas for software process improvement.

6. Medium: Through e-mail, personal dialogue and through internal courier.

7. Assignments: Problems were either designed or selected from the problem bank. Process group provides the criteria of selection. Normally the criteria [1] are: degree of complexity of the problem should be such that it enables the implementation of processual aspects such as reviews, documentation, etc. and it provides practical insight into product development cycle.

8. Allocation: The assignments were allocated to the trainees after the presentation of the problem statements by the mentors. During the presentation, mentors state the problem, their expectations and requisite understanding that is essential to carry out the assignment. Each mentor guides 2 teams.

9. Duration: 2 months

10. Resources: Standard learning resources on Software Engineering were made available. Systems laboratory has one VAX and two VME systems with 18 terminals.

2.2. Features

1. INTERFACE: To create a sense of expectancy about the workshop, and to build the requisite tempo to practice software engineering, a series of thematic outlines entitled INTERFACE on software engineering was released. The details of the themes can be found in [3], [7], [12].

2. PACT: To simulate the concept of contractual obligation between the user and the developer on the process level, Process Adoption and Conformance Treaty was made to be executed between the teams and the respective mentors. It is also to clarify the roles, expectations and responsibilities of the mentors and trainees. The text was adapted from [10].

3. ETVX Model: The concepts of ETVX model [5], [13] have been applied to the development process. A notion of the same was applied to the workshop operations. Precise entry criterion (E), task descriptions (T), validation procedures (V), and exit criteria (X) has been outlined both for the development process and the various phases of the workshop. The details of the model are depicted in figure 2.
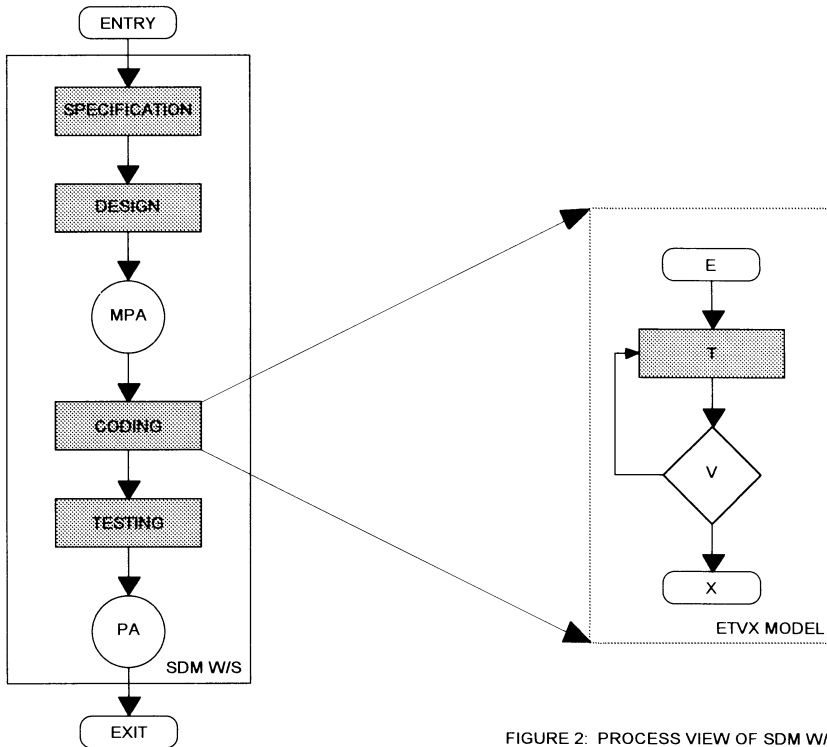

FIGURE 2: PROCESS VIEW OF SDM W/

4. Plans: The salient features of approach are strategic, operational and   software   quality plans [11]. They explain how the training objectives are aligned with the organizational goals.

5. Focus   areas: Error   prevention   and   early   error detection   were given due significance. Reviews, simultaneous documentation and strict adherence to schedules were identified as focus areas to achieve the needed behavioral change in the trainee.

6. Schedules: The schedule for the SDM w/s was designed based upon [12], with an explicit mention of reviews and milestones. Trainees were introduced to software project planning aspects.

7. Orientation: Process group introduces the trainees to approach and development process model of SDM w/s. During the SDM w/s, a variant of waterfall model was followed.

8. Lectures: GETs were exposed to the language module and digital switching software (DSS) architecture module prior to the SDM w/s. During a given phase of SDM w/s, generic, specific and

practice lectures were delivered. Lectures were also presented on review process and inspection cycle. Inspection cycle consists of problem data resolution, inspection, causal analysis, and fixing of errors.

9. Documentation: Specific deliverables were identified. Guidelines on documentation were provided to the GETs.

10. Self Learning Instruments: At the end of each phase questionnaire from [12] were compiled and released to trainees. It is to reinforce the conceptual understanding specific to a phase.

11. Accomplishment Review: Presentations were conducted for teams after the completion of all the exit criterion for all phases. Trainees are expected to *reflect* on *how* aspects of the development of the solution and discuss with the other teams.

12. Feedback: Trainees and mentors gave feedback on speakers' performance, on the workshop, and on the resource utilization. This was analyzed and necessary measures were incorporated in the design of the next SDM w/s.

13. Metrics: Seven process measurements were collected from the teams to gain the visibility into the process and fine-tune the approach for the future batches of GETs. The type of measurements can be found in [3].

14. Assessments: The performance appraisal form in C-DOT [3], [4] was adapted to meet the needs of SDM w/s. The scheme has one mid-term appraisal (MPA) and final performance appraisal (PA). They are carried out by the mentors.

15. Recognition Awards: Peer to peer recognition program (PEPUP) was instituted basing on the essence of [2]. The details of awards and nominations are in [3].

16. Newsletter: A newsletter named *STRIDES: for excellence* is brought out exclusively for sharing of experiences, information and insights during the workshop amongst the constituent members.

17. Meetings: The meeting process consists of agenda distribution, preparation, meeting and release of minutes. It was found that meetings were 80% effective. Criteria of effectiveness are the extent to which agenda was discussed and action items were finalized. On average 1.75 hours were spent on each of 6 meetings.

3. DIFFICULTIES

The salient difficulties that were encountered during SDM w/s were:
- Teams felt that the checklists and guidelines were generic.
- The deadlines were common to all the teams irrespective of the complexity of the assignment. It does not reflect the inherent complexities and in turn the need for different deadlines for different assignments.
- The propensity to realize the product at the expense of process has its bearing.
- GETs felt that documentation hampers the product development though they agree to the philosophy of documentation.
- Process group could not provide timely analysis of the inspection summary's data owing to high volume.
- Mentors had difficulties in assessing team members' involvement for the interactions were limited to review and consultative meetings.

4. FUTURE DIRECTIONS

The following measures are to be incorporated in the approach for the future SDM w/s. They are based on the feedback from the constituents of the communication model.
- Formalization of the meeting process.

- Introducing the prospective mentors to consulting skills.
- Conducting formal practice sessions on reviews.
- Increasing the variety of assignments.
- Strength of the team to be decided   basing on the man-hours' requirement for a given assignment.
- Team specific deadlines to be decided.
- Providing the checklists with examples.
- Project management practices and configuration management procedures to be incorporated.

REFERENCES

1. Bruegge, Bernd: "Teaching   an   Industry-oriented Software Engineering Course", *Software Engineering Education: SEI conference 1992 proceedings.* Springer-Verlag, LNCS 640, 1992, pages 64-79.
2. Carder, Brooks and Clark, James D.: "The Theory and Practice of Employee Recognition", Quality Progress, December 1992, pages 25-30.
3. C-DOT. Approach to Software Development Methodology Workshop - A primer. Centre For Development of Telematics, N. Delhi, December 1993.
4. C-DOT. C-DOT Performance Appraisal Manual. Centre For Development of Telematics, N. Delhi, March 1993.
5. Fowler P.and Rifkin, P: "Software Engineering Process Group Guide", Technical Report CMU/SEI-90-TR-24 ESD-90-TR-225. Carnegie Mellon University, Software Engineering Institute, February 1993.
6. Ghosh, H.: "Current Software Engineering Practices in C-DOT", Technical Report, C-DOT TR 001-1993, Centre For Development of Telematics, N. Delhi, June 1993.
7. Gibbs et. al., N. E., "The Carnegie Mellon University Master of Software Engineering Training Program", *Software Engineering Education: SEI Conference Proceedings*, Pittsburgh, Pennsylvania, July 1989, pages 152-154.
8. Kantipudi et. al., M.: "Software Engineering Course Projects: Failures and Recommendations", *Software Engineering Education: SEI conference 1992 proceedings*, Springer-Verlag, LNCS 640, 1992, pages 324-337.
9. G. B. Meemamsi. C-DOT story. Kedar Publications, N. Delhi, 1993.
10. M. C. Paulk et al. Capacity Maturity Model for Software, Version 1.1. Technical Report CMU/SEI-93-TR-24 ESC-TR-93-177. Carnegie Mellon University, Software Engineering Institute, February 1993.
11. B. D. Pradhan. Telecommunication: At least a phone in every village. IEEE Spectrum, March 1994, pages 43-47.
12. R. S. Pressman. Making Software Engineering Happen. A Guide for Instituting the Technology. Prentice-Hall, 1988.
13. R. A. Radice et. al. A programming process architecture. IBM Systems Journal vol. 24, no. 2, 1985, pages 79-95.
14. T. Srinivas. Role of Mentors. Process Document of Technical Training Division, Centre For Development of Telematics, N. Delhi, December 1992.