Software Process Practicum: Lessons in Software Quality and Leadership

Judy Bamberger
Sequent Computer Systems Inc
Adjunct Professor, Oregon Graduate Institute

bamberg@sequent.com
+1-503-578-3028

Sequent Computer Systems Inc
15450 SW Koll Parkway COL2-861
Beaverton OR 97006-6063 USA

James Hook
Oregon Graduate Institute of Science & Technology

hook@cse.ogi.edu
+1-503-690-1169

Oregon Graduate Institute
PO Box 91000
Portland OR 97291-1000 USA

Abstract

This paper describes a Software Process Practicum covering Software Quality and Leadership topics taught at the Oregon Graduate Institute during the Fall term, 1994. The Practicum is a result of a partnership between industry and academia, quality practitioner and theoretically-oriented professor. It is structured to meet both short-term training and long-term education needs. The Software Process Practicum is a unique and highly effective course covering both the "hard" issues of software process (definition of quality, basic quality tools and problem solving methods, process definition techniques, quality frameworks, useful metrics, formal inspection, Quality Function Deployment) as well as the "soft" issues (social styles, team maturity, meeting management, group facilitation).

The students who took the first offering of the Software Process Practicum have used in their work environment the concepts and techniques we taught in the classroom. They are continuing to demonstrate objective evidence of improvement on personal and organizational processes, which is visible to many of their managers and Vice Presidents.

This paper covers:

(1) Background

(2) Basic premises behind the Software Process Practicum

(3) Course contents

(4) Experience to-date: student-based results and instructor-based results

(5) Next steps

A copy of the course syllabus, which contains bibliographic references, is provided as an Appendix.

1. Background

In 1993, Oregon Economic Development Department funded an effort to:

> ... create a Software Engineering Model that characterizes performance excellence (current state and future state), create the capability to assess software engineers against that model, and provide a means to assess individual and organizational progress toward excellence on an ongoing basis.[1]

The organizations participating in this effort were: Intel, Mentor Graphics, Sequent Computer Systems, and Tektronix.

In January 1994, this group made its first public presentation. Members of the four industry organizations, the higher education community, and professional societies in Oregon attended. One objective of this meeting was to inform the colleges, universities, and other institutions of higher and continuing education of the needs of these four companies. A second objective was to identify "next steps" to continue or initiate collaboration among all the stakeholders to provide education and training for meeting industry needs today and expectations for tomorrow's software engineering population.

At that meeting, Judy Bamberger (Sequent – a report author) and James Hook (OGI – an educator) met each other, began discussing the model, and soon found out they had many interests in common, including taking action to address the ideas in the model.

In April 1994, Judy and Jim committed to teach a new course: Software Process Practicum: Lessons in Software Quality and Leadership. Moreover, we agreed to leverage what we could from the *Report of Findings* to guide the structure and content of the course.[2]

2. Basic Premises behind the Software Process Practicum

We agreed upon a set of guidelines for our selection of topics, readings, in-class exercises, homework assignments. These were:

- Use the "adult learning paradigm" to make the learning experience and the materials relevant; leverage experiences that adults bring with them to the classroom setting and on homework assignments; use a variety of teaching methods (lecture, discussion, hands-on; visual, aural, tactile; etc).
- Ensure the topics are applied and reinforce each other: use in-class labs to practice new skills; take those skills and apply them in a real-world setting with a partner organization; take time in class to discuss and apply readings and ideas; teach, do, and discuss.
- Teach the "soft" stuff as well as the "hard" stuff: emphasize the inherent interdependency of the two; leverage the fact that there already are plenty of courses and materials that concentrate on basic quality tools, techniques, and frameworks; add value by integrating and applying them to software engineering.

---

[1]From the *Report of Findings: Joint Software Engineering Needs Analysis*, 18 February 1994, available from Oregon Economic Development Department, Salem OR.

[2]From this point on, "we," "us," and "our" refer to the authors.

- Balance education (concepts) and training (skills) needs; teach multiple techniques, methods, and frameworks; provide students with the ability to make decisions about what is appropriate under which circumstances.
- Build on a variety of materials: seminal papers (such as Fagan on inspections, Radice on software process definition), outrageous editorials (such as the Bach/Curtis debate on the Capability Maturity Model), classical writings (such as Crosby, Deming, Juran on quality), books taking a "system" view of software quality issues (such as Weinberg), current findings and issues, cartoons, real-world practice (such as interviewing practitioners in industry and working with a "partner organization" on a multipart project), and experiences in the student's own work environment.
- Use information from the *Report of Findings* to select or validate what and how we are teaching.
- Ensure the students have fun while learning, and that we, too, both have fun and learn.

3. Course Contents

3.1. Overview of the Software Process Practicum

The following course description appeared in the OGI Fall 1994 course catalogue:

> The Software Process Practicum is designed to immerse the working student in topics relevant to software process improvement and quality management, and to introduce them to the supporting theory. Topics include process management frameworks (capability maturity model, ISO 9000), measurement for process improvement, and key team skills necessary for effective collaborative software engineering efforts. At the end of the course the student will be able to demonstrate that the software development process can be managed and controlled, leading to increased software quality. In addition to lectures, the class will include one Saturday workshop.

We determined that the overall goal of the course was to provide the students with a total quality focus. Thus, the specific objectives of the course are:[3]

> After this class, the students will understand and have demonstrated that:
>
> - Software processes can be managed and controlled.
> - They understand that software engineering is a social process, too.
> - They have real skills they can apply today at work.
> - They have a framework on which to build their own educated decisions about applying software quality principles and tools to personal, project, and corporate software activities.
> - They have identified three things to improve at their own work place (or within their own personal process) and are working on them.

---

[3]From the course syllabus.

In order to fulfill the objectives, we used the framework shown in Figure 3.1-1 throughout our planning for the course.

- The "hard stuff" – basic models, tools, and frameworks related to effective (software) process. Most of this would be provided initially via readings and class lectures.
- The "soft stuff" – the individual and team social concepts, models, and skills required to apply the hard stuff successfully. Most of this would be provided initially via readings and class lectures.
- Specific techniques – the integration of the frameworks and team skills in an applied setting. Most of this would be introduced via readings and class lectures and reinforced and practiced via hands-on labs during the class, the Saturday workshop, individual and team homework assignments. Moreover, team homework assignments would focus heavily on the integration of the hard and soft stuff in an applied, real-world environment. This would require the students to work in teams with a partner organization for an extended period.

Looking at the possible topics to teach, we recognized that there was a "hierarchy" of models, concepts, and skills: beginning with the simple and self-contained, and continuing to the more complex and more highly integrated.

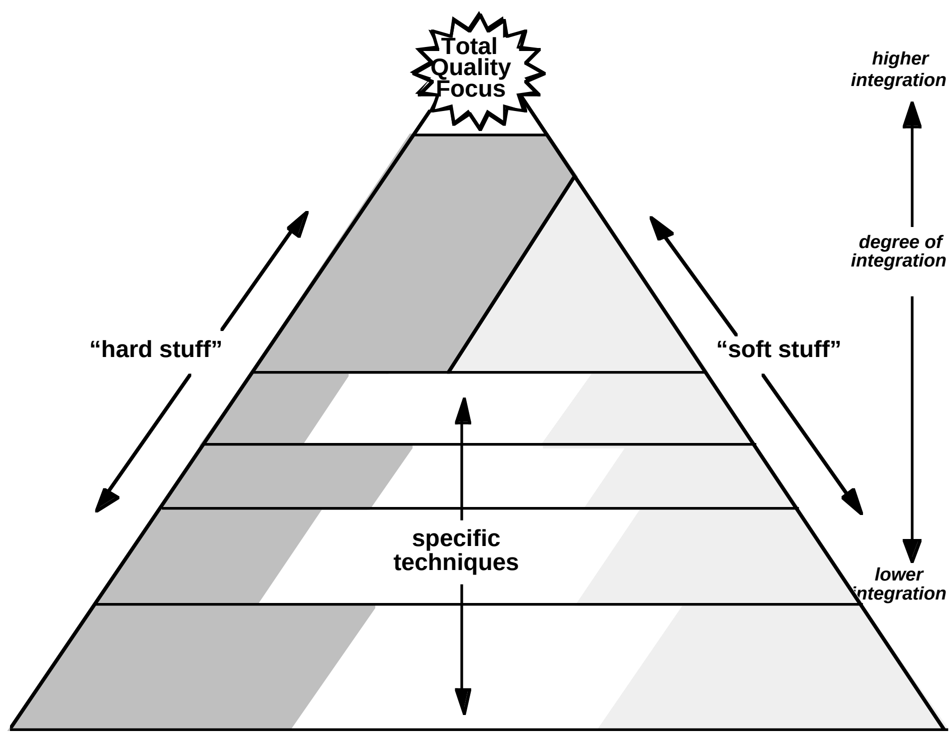


Figure 3.1-1. Framework for Software Process Practicum.

When we started to populate this framework with actual topics, we found we had identified many more topics than there was time to teach in 30 contact hours. To determine which topics we would teach in the fall of 1994, we used two primary sources for guidance: the competency model defined in the *Report of Findings*, and our knowledge of the local software engineering industry from which our students would come. The resulting framework populated with topics is shown in Figure 3.1-2.

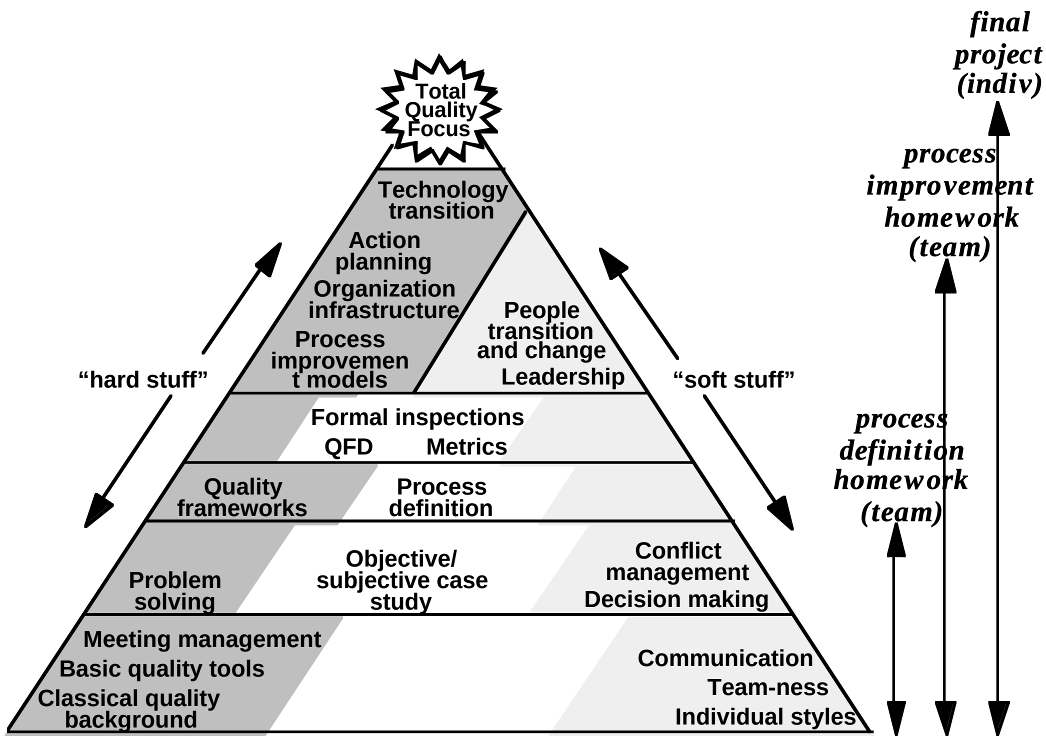


Figure 3.1-2. Topics Taught in Fall 1994 Software Process Practicum.

The Fall 1994 class sessions were ordered slightly differently than a literal interpretation of Figure 3.1-2 would indicate. (See the course syllabus in the Appendix.)

3.2. Mapping between Software Process Practicum and Software Engineering Competency Model

The *Report of Findings* contains a matrix of technical skills and competencies. These skills and competencies were identified via a thorough and structured information gathering process, described in detail in the report, along with examples of products used and sample results.

This section explains, briefly, each component of the Software Engineering Skills and Competency Model and provides a mapping between it and the Software Process Practicum.

The four technical skill areas (requirements) that were identified are:

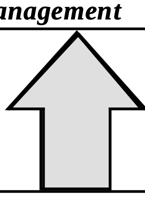
- Process and Management: software process, software review techniques, project management, business literacy, configuration management
- Design: object-oriented design/programming, data structure design, design methods/simplicity, formal analysis techniques, client-server architecture, human interfaces/graphical interfaces
- Implementation: software optimization, debugging techniques, understanding foreign code, technical writing, software testing techniques
- Specific Technical Skills: windows application programming, CAE related technical areas, communications and networking, natural data types, device drivers, real-time systems

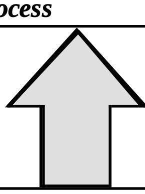
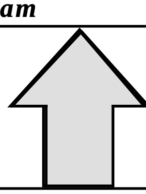
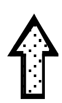
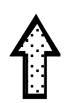
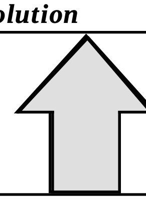
The identified competencies were organized into five groups:

- Concern for the Process (information gathering, efficiency, systematic thinking, discipline/rigor)
- Concern for the Team (collaboration, team building, technical leadership)
- Concern for Ideas (communication skills, influence/persuasion)
- Concern for the Company (risk management, results orientation, user orientation)
- Concern for the Solution (persistence, creativity, learning by doing, pattern matching, initiative)

The power of the software engineering competency model is demonstrated via the intersection of the technical skills and competencies. The Software Process Practicum satisfies the technical skills area of Process and Management very highly and three of the competency areas – Concern for the Process, Team, and Solution – very highly. These results are summarized in Table 3.2-1.

Table 3.2-1. Mapping of Software Process Practicum to the Skills/Competency Matrix

	Technical Skills			
	Process and Management	***Design***	***Implementation***	***Specific Technical Skills***
Software Process Practicum		None	None	None

	Competency Areas				
	Concern for the Process	***Concern for the Team***	***Concern for Ideas***	***Concern for the Company***	***Concern for the Solution***
Software Process Practicum					

The process and management technical skill area was the only one relevant to our course.[4] While all the competency areas were somewhat relevant, we needed to restrict our scope. As a result, we chose to focus very heavily on the Concerns for Process, Team, and Solution.

Each competency area summarizes several specific competencies, each of which in turn summarize multiple specific and observable behaviors identified via the in-depth interviewing and questionnaire steps described in the report. As will become clear in the following discussion, the original study did not go so far as to ensure each specific behavior, competency and competency area were completely distinct. Table 3.2-2 summarizes how the Software Process Practicum addresses each competency.

[4] OGI has issued a document mapping its courses to the Competency Model. This document is available from Oregon Graduate Institute, Computer Science and Engineering Department, Portland OR.

Table 3.2-2. Mapping of Software Process Practicum to the Competency Model

Key:	
R => Readings	
L => Class lecture	
A => Class activity	
H => Homework	

		Concern for the Process				Concern for the Team			Ideas		Concern for the Company			Concern for the Solution				
	Class, Topic	Information Gathering	Efficiency	Systematic Thinking	Discipline/Rigor	Collaboration	Team Building	Technical Leadership	Communication Skills	Influence/Persuasion	Risk Management	Results Orientation	User Orientation	Persistence	Creativity	Learning by Doing	Pattern Matching	Initiative
1		H		L	L				A H				L			H		
2		R L	R L						L					R	L A			
3	How does the "I" fit into "TEAM"?				L	R L		H				L				H		
4	Problem Solving Paradigms			R L H	H	L H	H	H	H						L H	H		
5-7	Process Definition Techniques		L A	L A	L	A	A	A	A				L			A		
8		R L		L								L						
9		R L A							A	A			R A			A	A	
10		R L	R L		R L													
11	Formal Inspection Workshop	A	L A		A	A	A	A	A	A						A		
12	Software Metrics			H					H			L H						
13	Process Improvement Models		R		L			L	H									
14		R L						R L			L			L				
15		R L						R L				L		R L				L
16		R L A		A		A	A									A		
		H		H	H	H	H	H					H	H	H	H		H
17		R A H	H	H	H	H	H	R H	A	A	A		H	H	H	H	H	H
18	Review and Summary	A							A									
	Final Project	H	H	H	H				H		H				H	H		H

The following paragraphs provide a detailed analysis of how the Software Process Practicum addresses each competency and many of the behaviors associated with each competency. This discussion is formatted as follows:

> ***3.2.x. Name of competency area***
>
> 3.2.x.y Name of specific competency (and description of that competency):
>
> WHAT THE CLASS SEES: The topics that are taught during class sessions or via readings.
>
> HOW THE STUDENTS USE IT: The ways in which the lectures, discussions, labs, and readings are used by the students to reinforce the subject matter.
>
> HOW THE INSTRUCTORS DO IT: The techniques and strategies we use to demonstrate how these same competencies apply to us, as instructors, as well.

3.2.1. CONCERN FOR THE PROCESS

3.2.1.1 Information Gathering (Being comprehensive in gathering information about technical and non-technical situations before making decisions or taking actions):

WHAT THE CLASS SEES: We teach information gathering techniques, such as brainstorming, force field analysis, and interviewing through class sessions and homework assignments. The problem solving model and process improvement models we teach later in the course leverage these techniques to gather data. We discuss the importance of validating the data, and only then making decisions. The importance of information gathering is discussed and demonstrated in class, and reinforced through the homework assignments.

HOW THE STUDENTS USE IT: The students are asked to demonstrate the information gathering competency throughout the course. Most of the course material is new to the students – so they are in information gathering mode throughout the term. Most importantly, all the homework assignments ask the students to *synthesize* the raw information and *apply* it in some way – to personal observations or to team projects, in simulated and real-world settings.

HOW THE INSTRUCTORS DO IT: We provide a wide variety of readings and leverage the knowledge brought by the students themselves, as described in the premises described in Section 2.

3.2.1.2. Efficiency (Finding ways to solve problems or complete tasks faster or with the fewest bugs):

WHAT THE CLASS SEES: We focus a series of lectures, class activities, and homework assignments on software process definition and improvement. We also teach two specific quality techniques: formal inspections and quality function deployment (QFD). In formal inspections, the objective is to build software work products right the first time (by following rule sheets, guidelines, etc), and then verify that the software work products are correct, complete, and consistent according to the same rule sheets, guidelines, and predecessor products. In QFD, the objective is to capture the voice of the customer by eliciting requirements, tracing them to possible design/implementation schemes, and establishing relative weightings for them.

HOW THE STUDENTS USE IT: The Week 5 homework assignment, to be completed as a team, is to define a process in a partner organization with the intent of creating an improvement plan during the Week 9 homework assignment. This involves using experts early (to define the process), and leveraging those same experts to identify potential improvements. We hold a formal inspection lab where we walk through the entire formal inspection process, produce lists of defects and metrics, and discuss their use in improvement. The hands-on exercise in the QFD course allows the students to begin understanding how important it is to capture the voice of the customer clearly and early in the project.

HOW THE INSTRUCTORS DO IT: We established module templates from the beginning; we reused and tailored course modules from other offerings; we used tried-and-true published samples of plans, standards, guidelines, etc instead of creating our own.

3.2.1.3. Systematic Thinking (Taking a well-ordered and logical approach to analyzing problems, organizing work and planning technical solutions):

WHAT THE CLASS SEES: We teach one canonical problem solving model; we provide several different techniques for making team decisions – from voting schemes to selection grids. We teach basic meeting organization and management skills. When we teach process improvement and action planning, we frame it as an orderly process – just like any other project. We provide students with the tools to be able to apply these skills and models.

HOW THE STUDENTS USE IT: The meeting management, problem solving, and decision making skills are used in the first team exercise – applied to a real-world decision-making activity. They are then reinforced in later assignments. All team assignments have a problem to solve; all of them require effective planning, problem solving, and decision making skills. All of them require the ability for a greatly varied group of students to come together as a team and present a cohesive and coherent solution.

HOW THE INSTRUCTORS DO IT: We structure all our course modules in a clear and consistent manner: asserting objectives at the beginning; using those to drive the lecture and lab materials, as well as the related readings and homework assignments; and summarizing the material presented or discussed and how it relates to the module objectives. We drive this from the top – beginning with course-wide objectives and vision; our last class session is devoted to reviewing how the parts related to the whole of the course.

Fall 1994: Our students indicated we need to improve in this area; they did not see the "big picture" that was in our heads. We will introduce Figure 3.1-2 in the first class session and we will use it in each class to focus students on where the current topic fits into the overall course structure.

3.2.1.4. Discipline/Rigor (Making sure that technical work is done correctly, completely and to the highest standards):

WHAT THE CLASS SEES: The modules that we teach on problem solving, formal inspections, QFD, meeting management, and action planning address many of the key behaviors covered in this competency. As part of the software process definition module, we teach the importance of ensuring that current-step products are consistent with their predecessor products.

HOW THE STUDENTS USE IT: In all team assignments, the students are asked to provide a description of team goals, plans, meetings. They are also to provide a discussion of how the team achieved its goals, and how effectively the team worked. Most team projects are structured so the work can be shared; it requires planning and discipline for all pieces to come together in a single package.

HOW THE INSTRUCTORS DO IT: Our goals are to ensure reasonable and consistent structure, organization, and relationships among our materials, lessons, objectives, and homework assignments. When we make commitments to ourselves or to the students, we keep them.

3.2.2. CONCERN FOR THE TEAM

3.2.2.1. Collaboration (Cooperating with others to achieve a common purpose):

WHAT THE CLASS SEES: One course objective states that "software engineering is a social process, too," and we emphasize this aspect. We teach one module specifically focused on individual social styles, team process, team maturity, and team leadership. We weave these themes throughout the course; we highlight them explicitly in all team exercises, and we often discuss them following in-class, team exercises. Slightly more than half the student's grade comes from team assignments; all members of a team receive the same grade on those assignments. About half the small homework assignments are to be done as a team (accounting for 10% of the student's grade). Both the Week 5 and Week 9 homework assignments are team assignments, accounting for another 50% of the student's grade (20% and 30% respectively).

HOW THE STUDENTS USE IT: Week 5 and Week 9 homework assignments are extremely difficult to do without *effective* team collaboration. In addition to working together on the assignments outside of the classroom, students are required to present the results of both Week 5 and Week 9 homework assignments to the entire class. This is to be done as a team.

Fall 1994: We obtained much insight into the degree and effectiveness of this collaboration by reading the minutes of the team meetings (often submitted with the Week 5 and Week 9 homework assignments).

HOW THE INSTRUCTORS DO IT: We taught collaboration by demonstrating it on all aspects of course development and delivery.

Fall 1994: Students were brought into this collaboration as well; at a few points throughout the term, we elicited input and discussed student reactions to the content and pacing of the course, and we made adjustments accordingly. In fact, at the first session, we asked the students for their background and their expectations from this course, and we tailored some information accordingly. We will incorporate many of their suggestions before the next offering of the course.

3.2.2.2. Team Building (Demonstrating leadership to create an environment which sustains highly motivated technical groups working to achieve common objectives):

WHAT THE CLASS SEES: We teach modules on teams, meeting management, organizational infrastructure for successful process improvement, and managing change to emphasize this theme.

HOW THE STUDENTS USE IT: Students are asked to describe and explain the maturity of their teams, the roles and responsibilities of each team member, how well each student fulfilled her/his role, and the impact of her/his social style on team activities.

Fall 1994: By the end of the term, all students were describing their teams as fairly mature teams, and were able to describe their role(s) within their team clearly. One team, in fact, had adopted the technique of spending the first 5 - 10 minutes of each meeting reviewing the effectiveness of the previous meeting, and then would make adjustments accordingly. During their Week 5 and Week 9 homework presentations, these students shared their belief that these 5 - 10 minutes were extremely well spent – very few issues or feelings were left to simmer across meetings, and the team could move forward. In fact, the "team stuff" was cited most often as one of the most valuable skills learned, and one that they could find most useful in their work environments. Many students were already using the techniques we taught long before the term ended.

HOW THE INSTRUCTORS DO IT: As discussed just above, the students saw Jim and Judy taking leadership roles in those areas where each was more expert; we operated as a "tag team." Moreover, we empowered the students to be effective and self-managing by providing the knowledge base (class time, readings, hands-on practice) and any additional assistance needed.

3.2.2.3. Technical Leadership (Brings technical expertise to solve difficult problems and demonstrates confidence in one's own technical capability and skills):

WHAT THE CLASS SEES: We discuss team leadership when we discuss teams. We also discuss the role of the leader in meeting management. We revisit the topic of leadership in a more global, organizational sense, toward the end of the course in the modules on organizational infrastructure and technology transition. In that context, we discuss leadership characteristics from positional, influential, and expertise leverage points.

HOW THE STUDENTS USE IT: Each student has the opportunity to take the lead during team assignments.

Fall 1994: Some teams had more stable structure (i.e., certain members gravitate to leadership roles and maintain them, naturally, for the duration of the term); other teams had a more dynamic structure (i.e., different members step into the lead, identify that, and maintain it just for the particular activity). This was reflected in the team analysis assignments. The students were conscious of who was leading what and when, and discussed in their homework assignments how effective this was.

HOW THE INSTRUCTORS DO IT: Nothing to add to the above.

3.2.3. CONCERN FOR IDEAS

3.2.3.1. Communication Skills (Providing information so that others can understand complex and detailed technical ideas or processes):

WHAT THE CLASS SEES: This is one area that fell below the "cut line" when we prioritized topics to teach.

HOW THE STUDENTS USE IT: While we didn't teach communication skills directly, the students have the opportunity to demonstrate their abilities at a variety of techniques. There are written, individual homework assignments; oral presentations by the team; general discussions during class.

HOW THE INSTRUCTORS DO IT: We recognize that each student, with her/his own style, may prefer one communication format over another. As a result, the course uses several means of communication: informal, in-class discussions (analysis, sharing of ideas, asking questions); team presentations of team homework assignments; and individual and team writing assignments. We use words and pictures, discuss our own experiences, cite experts, provide quantitative and qualitative data, and more. As instructors, we try our best to set good examples.

Fall 1994: We did not correct grammar (written or verbal); we did not focus on presentation skills explicitly. The students varied greatly in their ability to communicate. We tried to provide positive feedback for excellence, and objective comments where opportunities for improvement were identified.

3.2.3.2. Influence/Persuasion (Establishing personal technical credibility as a means of influencing others or gaining acceptance of one's ideas):

WHAT THE CLASS SEES: This is another area that fell below the "cut line" when we prioritized topics to teach.

HOW THE STUDENTS USE IT: Week 5 and Week 9 homework assignments require working with a partner organization; the key to success is the ability to influence that partner to seeing the benefits of that collaboration.

Fall 1994: During student presentations, there was clear demonstration of using these techniques with their organizational partners in order to get the initial buy-in to the collaborative effort. Several of the students also discussed significant acceptance of the results – one student was starting to brief a series of Vice Presidents in her company about the process definition and process improvement plan her team created. And already some of her team's work was being integrated into an on-going process improvement effort.

HOW THE INSTRUCTORS DO IT: Nothing to add to the above.

3.2.4. CONCERN FOR THE COMPANY

3.2.4.1. Risk Management (Foreseeing and taking action to deal with possible future events, problems and opportunities):

WHAT THE CLASS SEES: This was not discussed explicitly as part of the course.

HOW THE STUDENTS USE IT: The Week 9 homework (team) and the final project (individual) ask the students to define a process improvement plan using a technique that (among other things): (1) produces a schedule of who, what, and when; (2) examines enablers and barriers and their relative strengths (via force field analysis); and (3) focuses on building the sponsorship chain needed for successful institutionalization of the process improvement. These are all geared toward making schedule, requirements, trade-offs, and status visible throughout the effort.

HOW THE INSTRUCTORS DO IT: In our case, as instructors, our risks are in the area of developing, maintaining, and delivering a high-quality course. We provide the complete syllabus at the beginning of the course, and revisions in a timely manner as needed (often incorporating suggestions of the students).

Fall 1994: Given this was the first offering of the course and that many of the materials were new, we tried to reuse what we could, and allocated time to stay ahead of deadlines. We did not accommodate machine crashes very well in our schedules, and on occasion, over-committed ourselves.

3.2.4.2. Results Orientation (Working to achieve measurable (i.e. quantitative) success on organizational or team objectives):

WHAT THE CLASS SEES: We teach a module on metrics. Many of the readings we provide describe process changes and how quantitative data are used to demonstrate the overall improvements. One of the (individual) assignments is for the students to produce a detailed impact case study and subjective impact study, applying the readings from the Weinberg text.

HOW THE STUDENTS USE IT: In order to fulfill the enumerated requirements of the Week 5 and Week 9 homework assignments, students have to stay focused on those tasks. They need to spend time up front reviewing the goal/objective stated in the assignment; planning what is to be done, when and by whom; and how to produce the write-up and presentation.

Fall 1994: The student teams whose meetings had clear objectives and agendas appeared to make more effective use of their time together, as stated by the students themselves. Making and using agendas had the benefit of the teams focusing on stating the goal/result desired, and structuring the activities to achieve those results. The team that used the first 5-10 minutes to review their previous meeting observed that by taking that time, they ensured closure on topics and team issues, which allowed the team to move forward toward their goals more effectively.

In addition, many of the student projects defined a set of metrics needed to measure the effectiveness of the process being analyzed, providing a quantitative base for success or failure.

HOW THE INSTRUCTORS DO IT: We focus intensely on ensuring the students had all the materials (readings, handouts, other supporting materials) they need at the committed times.

Fall 1994: When the 1.5-hour class session was insufficient for presenting the entire topic, we tailored. The criteria that we used were determined by the course and module objectives, and by the need to support successful completion of homework assignments. Other times, we asked the students if they were willing to stay an additional fifteen minutes to finish covering the section, to demonstrate negotiation techniques.

3.2.4.3. User Orientation (Immersing self in customer's/user's environment to understand their needs):

WHAT THE CLASS SEES: The basic premises of quality focus heavily on customer/user focus and involvement. The process definition technique we teach is a collaborative process between the team and those who enact the process. Many of the quality frameworks we explore have an explicit section on customer focus (e.g., Malcolm Baldrige). In fact, this is one comparison dimension we use during the class discussion of the frameworks. The session on QFD focuses on listening to the voice of the customer, and how that is driven down into the various engineering solutions that are possible to satisfy the customer.

HOW THE STUDENTS USE IT: The students must get the "customer" – the partner organization – involved in Week 5 and Week 9 homework assignments.

Fall 1994: We saw many instances of user orientation demonstrated, for example: involvement in the process definition exercise and feedback/validation sessions; prototyping improvement-oriented metrics with real data to validate the model; testing of assertions with the customer throughout the process improvement plan definition.

HOW THE INSTRUCTORS DO IT: At the start of the course, we emphasize our desire to practice what we preach – focus on continuous improvement. We believe that the students are our customers/users, and we are sincerely interested in their ideas for improvement throughout the course.

Fall 1994: We explicitly set aside two times to get feedback from students, and made improvements from their suggestions – those they could see during this offering of the course, and others for the longer term.

3.2.5. CONCERN FOR THE SOLUTION

3.2.5.1. Persistence (Persevering in the face of obstacles and challenges):

WHAT THE CLASS SEES: We discuss this in the context of making lasting process improvements, in the sessions on action planning; organizational structure and position, expertise, and influence power; leadership; technology transition and managing change. We discuss typical and actual barriers to process improvement, and how to structure a process improvement effort for success.

HOW THE STUDENTS USE IT: Week 5 and Week 9 homework assignments are important in allowing students to demonstrate this capability in two key ways. First is the very premise of having a team of students from different companies (different geographies, work schedules, positions, backgrounds, etc) come together, identify common meeting times, work with a partner organization. This persistence continues with the partner organization in defining a process, identifying opportunities for improvement, and creating a plan to implement that improvement.

Second, the process improvement plans that are written as part of Week 9 homework and the final project contain a section identifying enablers and barriers (presented as force field analysis).

Fall 1994: Some individuals and teams also provided a discussion of using the enablers to address the barriers. Others identified the sponsorship structure they intended to build in order to address – or even circumvent – the obstacles and challenges facing them.

HOW THE INSTRUCTORS DO IT: Our biggest obstacle may be that each of us also works a "real" full-time job, so it is a challenge for us to meet and to get materials ready in a timely manner!

3.2.5.2. Creativity (Challenging existing assumptions, ground rules, constraints, etc. to propose and act on alternative strategies or ways to frame a problem):

WHAT THE CLASS SEES: We introduce the topic of creativity at the beginning of the course, via brainstorming. We discuss it again as a key step in the problem solving process, which is then used throughout the rest of the course.

HOW THE STUDENTS USE IT: As stated above, the challenges faced by the student teams in doing the Week 5 and Week 9 homework assignments require significant creativity – creativity in identifying feasible and sale-able process improvement activities and strategies; creativity in overall team problem solving – including when, where, and how to meet.

HOW THE INSTRUCTORS DO IT: On the surface, our choice of topics may appear traditional; however, our teaching methods, homework assignments, and reading selections are creative and designed to encourage creativity by the students.

Fall 1994: At the time we began, we were unable to locate any information about courses offered anywhere that integrated the concepts and techniques we identified, met the goals and objectives we derived from the local community, or that addressed the needs identified in the *Report of Findings*.

3.2.5.3. Learning by Doing (Having a deliberate and well-thought-out strategy about learning how to learn (i.e., meta-thinking)):

WHAT THE CLASS SEES: As stated many other places, we provide numerous opportunities to try, test, demonstrate, and synthesize the information taught and discussed, provided as readings, and shared during class and team sessions.

HOW THE STUDENTS USE IT: Every homework assignment and class activity provides the opportunity to apply what is being taught and shared.

HOW THE INSTRUCTORS DO IT: This is a basic premise of the class, whether it is "doing" during class time (hands-on, group exercises, discussions) or as homework (more than half the assignments are team-oriented, working with a "partner organization" on real-world issues). Even individual assignments often ask the students to address how the readings or classroom topics apply to them in their work situations.

Fall 1994: Our statements of what this course has accomplished is based on individual and class discussions and on the homework assignments. We observed compelling written and oral demonstrations that the knowledge we intended to impart was, in fact, integrated into each student's working knowledge of the field – to varying degrees across students.

The competency model calls out learning by doing as key to effective learning in all skill areas. Our experience and observations during this class confirms this yet again.

3.2.5.4. Pattern Matching (Using prior experience or mental models to make informed choices or decisions on new situations):

WHAT THE CLASS SEES: We use this technique in discussions and homework assignments. For example, each student has to read information about three of seven quality frameworks. During class, the students discuss the applicability and relative strengths and weaknesses of each model. We seed the discussion with a few questions we know elicit differences and discussions; the students use their knowledge and experiences to carry the discussion forward.

HOW THE STUDENTS USE IT: The students are provided with many basic tools and models throughout the course that they are expected to apply. As we present this information, we discuss when each is applicable, and we emphasize that there is no "one right set for all occasions."

Fall 1994: One team stated they spent some (frustrating) time trying to determine which methods and tools were most appropriate for their activity (this was also documented in their minutes!). They suggested that, perhaps next time, we propose a preferred set to apply. When we probed further, we discovered that this struggle alone had provided some invaluable learning that we intended to occur – determining what is appropriate when, with whom, under what circumstances, etc. This was not something the students had considered when we began discussing this issue. This was later confirmed in a student comment submitted with the final project.

HOW THE INSTRUCTORS DO IT: We structured the course to build on basic tools, and then to have the students develop criteria for selecting which tool, technique, model, framework was appropriate. This is the thinking behind teaching the "low integration" concepts first and the "high integration" concepts at the end.

3.2.5.5. Initiative (Taking independent action and going beyond what the job or situation requires):

WHAT THE CLASS SEES: This was not discussed explicitly as part of the course.

HOW THE STUDENTS USE IT: The students are provided with a set of criteria for each homework assignment. They see the "what"; by their individual and team initiative, they provide the rest.

Fall 1994: Overall, the students significantly exceeded our expectations, which were enumerated as requirements in the assignment descriptions. Week 5 and Week 9 homework results were all excellent to superlative (we ended up creating a new grade to accommodate the results we saw). Many of the students' final projects also ranked that way – demonstrating significant integration of the various topics that were heard, discussed, read, and practiced throughout the entire course.

HOW THE INSTRUCTORS DO IT: It was unclear what type of follow-up was to occur from that initial meeting in January 1994 where we first met. The industry group that produced the *Report of Findings* had asked for a coordinated response from academia. In the sense that industry had provided a consolidated report, and based on the number of institutions and individuals involved, it made sense to request a coordinated response at the "policy" level. We did not wait for that to occur; we saw a need, a possibility, an opportunity; we had the knowledge, connections, and experience; we had the burning desire to take action. And we did – we scheduled, developed, and taught this course.

We also take the initiative to solicit input from students for improving the course. We do this at planned, key points throughout the term and we are open to receiving them on a continuing basis. We are set up to react to many of those "in real time," and have a mechanism to carry forward the others to improve future offerings of the course.

Fall 1994: Based on the results we have seen to-date, and the increasing interest as our students spread the word and demonstrate success, and as we distribute our syllabus, we will teach this course next fall. Before we offer this course again, we will address those areas our students identified as the weakest.

4. Experience To-Date

Ten students enrolled in the Fall 1994 offering of the Software Process Practicum, an acceptable enrollment for a first offering of a course like this, based on past history. Of the ten, four were full-time students at OGI; the other six from industry. Six students are pursuing Masters or Doctorate degrees. The industry students are from various sized companies, and have a variety of job responsibilities, including software engineering, customer support, and training. Very few students had any prior software engineering education beyond unstructured "on the job" training.

The class was scheduled to meet twice each week for one and one-half hours; several classes ran over by up to 15 minutes. The class met for 10 weeks, with one all-day, Saturday lab. Class time was mostly lecture, along with significant time for discussion and hands-on classroom activities.

Readings were assigned each night; individual homework assignments were assigned intermittently; there were two major team homework assignments. As soon as the students indicated the load was just too much, we reduced the amount of reading and eliminated many of the remaining individual homework assignments.

The two centerpiece homework assignments were to be done in teams and with the participation of a partner organization. These were to define a real-world process (Week 5 homework) and to create a process improvement plan (Week 9 homework). Both these assignments required significant effort on the part of the students - including identification and enlistment of the partner organization, interviewing and working with its staff, and providing information back to various levels of the organization. Students were required to give presentations of these two major homework assignments.

As it happened, the teams stayed constant throughout the term. For some of the in-class activities, we specifically selected teams to ensure the students had an opportunity to work with a variety of people.

The Saturday lab was a hands-on formal inspection workshop, concluding with the presentation of the process definition projects (Week 5 homework). We provided coffee, breakfast, and snacks. The students presented their process improvement projects (Week 9 homework) at the penultimate class session. The final session was a review and synthesis session in preparation for the final project. In addition, we provided Consensus Pizza to celebrate the ending of the course and the success of the students.[5]

The students had a final project to complete, to be done individually although peer review was allowed. This project was to produce a process improvement plan for a process in which the student was involved - a personal process or a larger, organization-wide process. We proposed a cap of eight hours to be spent on this final project; some students chose to spend more time than this.

Grades were computed as follows, with about half the grade based on individual performance and half the grade based on team performance:

- Homework - individual/team 20%
- Week 5 homework - Process Definition - team 20%
- Week 9 homework - Process Improvement - team 30%
- Final project - individual 30%
- Class participation (individual) subjective

4.1. Student-Based Results

Nearly all of our students cited meeting management as the most important single topic we taught, and one that was immediately applicable in their work environments. Many students also cited the process definition representations and techniques as something they are using already at work. We know that they are being used in three of the companies. One student shared with us an observation from the process definition and improvement homework assignments done with her partner organization. The organization's staff stated that they finally saw the relevance of and tangible results from many quality techniques in which they had been

[5]Started almost absent-mindedly and just in fun, this became a powerful activity where the students, in trying to decide on one and only one type of pizza to be consumed by all of us, demonstrated exceptional skills in meeting management, facilitation, consensus building, and problem solving. The students self-selected a leader, facilitator, recorder, and timekeeper. The leader and facilitator ensured everyone participated. Creative persuading and negotiating techniques were used to arrive at the final decision. And in case the reader is curious: whole wheat crust, thick, lots of cheese (mozzarella, parmesan, romano), mushrooms, green peppers in large pieces, roasted garlic, artichoke hearts, sun-dried tomatoes. It was delicious!

trained, but had not used to that point. Two students shared with us that their industry managers told them that the topics and methods being taught in our course are of significant importance to them.

Two students are introducing formal inspections into their workplaces. For example,

> A project at ADC Kentrox scheduled a three week period specifically for code inspections, and they are already benefiting from the results of preparing for and conducting three code review sessions. While not as formally conducted as were the sessions taught in class which used uniform codes for defect categorizations and which also focused on accumulating defect metrics, the time spent has widened the scope of understanding of the software by each team member and has resulted in positive, identifiable improvements to the software.[6]

Another student, in her final project, prepared a three-year plan for the introduction of formal inspections across her organization. Her plan includes:

- A force field analysis (supporting and opposing forces) and a set of diagnostic questions to help assess risks and barriers to success as well as strengths and opportunities to leverage.
- A list of sponsors and other influential people and situations to initiate and maintain momentum for this effort. Her plan leverages position, influence, and expertise power.
- An action plan - a detailed list of steps with a responsible person and due date. This builds on the previous two items above.
- A set of metrics to measure the effectiveness of the effort. These metrics will be used to evaluate the results from a pilot project. Based on those results, the introduction of formal inspections will be evaluated and rescoped if necessary before continuing.
- A training and roll-out plan, focused to ensure organization-wide institutionalization of the formal inspection method.

We received many positive comments about the exposure to the wide variety of quality frameworks we used, for example:

> ... exposure to the Trillium Model (which integrates and adds to aspects of SEI's CMM [Software Engineering Institute's Capability Maturity Model], ISO 9000, Malcom Baldrige, and IEEE concepts, and targets these concepts to the Telecommunications Software industry), has led to intense interest in making use of the model for identifying specific areas for potential improvement at ADC Kentrox.[7]

One of the student process definition and improvement projects has been presented to two Vice Presidents. The instrumentation (metrics) requirements identified by the students will be included in the partner organization's metrics. These metrics will help the partner organization measure how well it is meeting its specific goals in the area of time commitments to customers. The proposed metrics are defined precisely and tied to specific process steps. Other process

---

[6]Personal correspondence from Gary Hanson.

[7]Personal correspondence from Gary Hanson.

improvements were also proposed and will be adopted. These include better control of certain software products, and taking five, piece-meal, information-gathering process steps and merging them into a single step done once at the beginning.

4.2. Instructor-Based Results

Our basic job is going to entail figuring out what we did right and repeating that, and improving areas where we were weak. We asked for, and received, feedback throughout the course. The most significant piece of information we received was "too much too fast!" To address that, we will increase the credit hours earned for this course from three to four, and run the class for two two-hour sessions each week. We will not add more material; we will slow down and spend more time making explicit ties with the readings and individual homework assignments, another issue that was raised.

The independently administered end-of-term course evaluations yielded no surprises. We received very strong ratings in the following areas overall:

- Instructors knowledgeable
- Lectures well prepared, organized; presentations made for easy note-taking; topics appropriate
- Good communication skills
- Concerned about students; opportunity to ask questions; students treated fairly
- Course material challenging
- Increased my interest
- Built/augmented student competency

And, as expected, we received lower ratings in the following areas:

- Related course concepts in systematic manner
- Pace was not too fast
- Textbook and supplementary materials valuable

Other comments we received included:

- "Keep it as a pair of instructors." coupled with "Jim provided a good 'sounding board' and 'plant' for Judy's lectures. Jim's PacSoft work was good background."
- "As [a] professor, [Jim] gives a strong testimonial for the material's importance."
- "More on metrics." (We will add one more session on metrics and remove something else.)
- "Course content very important for OGI and community. Get more industry involvement."
- "[Aspects you would like to stay the same] Team projects. Software processes. Subject matter. Instructors excellent - retain and reward. Saturday labs very good."

We are planning on teaching this course again in the fall 1995 term. That gives us sufficient time to execute our own process improvement cycle, especially addressing those areas where the students gave us lower ratings.

5. Next Steps

We have been maintaining sporadic contact with our students. The improvement efforts that our students began are continuing - albeit slower than anticipated. They are making progress, and they know we are resources willing and able to assist them. We believe strongly that it is not the classroom performance that will prove the benefit of our course, but rather how that information is taken back into the workplace and applied successfully. We plan on doing a more formal follow-up with our students six months after the course.

In the mean time, we have been pursuing additional liaisons with local industry. Through the Oregon Graduate Institute Computer Science and Engineering Department, we targeted the four companies that participated in creating the software competency model - Intel, Mentor Graphics, Sequent, and Tektronix. Our goal is to improve collaboration between industry and academia. Industry has stated it's needs; it is now academia's opportunity to put a program together - like we did for the Software Process Practicum. And then it is back to industry to provide the opportunities and encouragement to its employees to participate in the educational opportunities that have been created.

We have also created a series of short courses that are available to industry. Many of these courses are being presented as "workshops" at national and international conferences.

And as we ourselves do not have all the answers, we would welcome any comments, observations, suggestions, and participation from readers of this report.

Our conclusions are straightforward:

- We have demonstrated that it is possible to have a working, effective, collaborative partnership between industry and academia wherein everybody wins - especially the students.
- We have demonstrated that via this collaboration, academia (the Oregon Graduate Institute) can respond to industry needs effectively, once those needs are articulated (as they are in the *Report of Findings: Joint Software Engineering Needs Analysis*).
- Our students tell us that they are using what we taught them, that it is working for them, and that they are making improvements and changes in their personal processes, their projects, their organizations.
- We have demonstrated that students can be taught that "software quality stuff" in a way that enables them to begin to use it immediately and to demonstrate objective and quantitative results.
- We believe as strongly as ever that the "soft" stuff is just as important as the "hard" stuff. In fact, it is the interpersonal and team techniques and concepts that underlie and enable effective process definition and improvement techniques.

- Learning by doing – again – and again – and again – is a very powerful teaching technique. The academic environment provides a "safe haven" to experiment with new concepts and methods. By tying our learning-by-doing assignments to real-world issues, we have provided a safe and relevant environment that fosters effective learning.

6. Acknowledgements

We appreciate the efforts of the State of Oregon and the team from Intel, Mentor Graphics, Sequent, and Tektronix for their efforts in creating an outstanding model that proved very useful to us. Thank you to the support staff of OGI who went beyond the call of duty to support this initial teaching of the Software Process Practicum, notably Phyllis Raymore and Kerri Burke. Most important thanks go to our students, who took the material we offered and made brilliant use of it in their own work environments: Jim Bindas (Intel), Debbie Blanchard (Consolidated Freightways), Michael P Gerlek (OGI), Gary Hanson (ADC Kentrox), Allyn Jackson (ADP Dealer Services), Celeste Johnson (Mentor Graphics), Alexei Kotov (OGI), Christos Mandalides (OGI), Thom Parker (Intel), and William Trost (OGI). Our course, and this paper, reflects many of their suggestions for improvement. Our appreciation also to those partner companies who provided a real-world laboratory for our students, and who benefited from the results.

Appendix: Software Process Practicum (CSE 582) Syllabus from Fall 1994

