



CIM: Capability-Innovation-Motive Teaching Model for System Engineering Education – “Embedded Operating Systems” as an Example

Yu-Lun Huang^{1,*}, Chao-Yang Cheng², and Sunny S. J. Lin²

¹Institute of Electrical and Control Engineering, National Chiao Tung University, Taiwan

²Institute of Education, National Chiao Tung University, Taiwan

(Received 30 June 2015; Accepted 27 July 2015; Published on line 1 September 2015)

*Corresponding author email: ylihuang@cn.nctu.edu.tw

DOI: [10.5875/ausmt.v5i3.978](https://doi.org/10.5875/ausmt.v5i3.978)

Abstract: Traditional education models face great challenges from mobile devices, social networks and open courseware. Current professional knowledge and system design skills are inadequate for engineering-tracked students to thrive in a competitive job market and unpredictable professional contexts, which require them to develop unique ideas and innovations and know how to realize them. This study seeks to establish instructional arrangements in system engineering education to foster student creativity. A semester-long “embedded operating system (EOS)” course was offered as a trial system engineering course. In the paper, we explain the instructional challenges encountered and how these were addressed using the proposed Capability-Innovation-Motive (CIM) teaching model. Based on the CIM model, the EOS syllabus was redesigned to enhance student domain capabilities and build their innovative skills to help them better understand and manage abstract concepts discussed in the lectures. We then collected and analyzed student feedback by implementing Day Reconstruction Method (DRM). Analysis results show that the outcomes of both the best and worst final projects show a promising degree of creativity.

Keywords: System engineering education, innovative skills, SCAMPER, embedded operating system course

I. Introduction

Information technologies have evolved dramatically over the past few decades. In particular, the recent emergence and popularity of mobile devices, social networks and open courseware present significant challenges to traditional pedagogical approaches, especially for the instruction of system engineering. Today, instructional goals for teaching professional knowledge and system design skills must go beyond preparing engineering students to thrive in an unknown future, but must foster the ability to develop unique ideas and innovations, and to adapt and survive in a competitive market and a changing world.

To promote the reform of system engineering education, Taiwan government agencies are providing increased financial support to innovative projects to address the demands, challenges and trends presented by

new educational technologies for system engineering courses. Industry is also emerging as a strong proponent for innovative educational approaches. In response, universities are increasingly emphasizing embedded systems in system engineering courses, with a special focus on the development of hardware/software co-design skills, which are of particular importance in embedded systems. Another focus is OS design, which also plays a critical role in coordinating applications and tasks running on embedded systems.

Several years ago, the Institute of Electrical and Control Engineering at National Chiao Tung University, Taiwan started offering an “embedded operating systems (EOS)” course to bridge gaps between academic instruction and industrial demand. To cover many abstract concepts, the course adopts a “learning-by-doing” approach to supplement course lectures. These activities aim to introduce critical hands-on practices (see Fig. 1) and engage students in tasks similar to designing small-



scale real cases. The implementation requires integration of newly learned concepts and problem solving skills. In general, learning-by-doing deepens students' understanding of concepts covered in class, but students indicated they still experienced difficulty in applying what they learned in the lectures to real tasks.

In 2014, the EOS course was launched as a trial system engineering course. We collaborated with education experts to design a novel teaching model, and redesigned the syllabus to emphasize development of innovative skills. We expected this approach would help students to better understand and manage the abstract concepts covered in the lectures. In this paper, we describe our experience in redesigning the EOS course. Section II reviews the course outline, and Section III describes our teaching model (Capability-Innovation-Motive, abbreviated as CIM). We explain the teaching

challenges and how we address them in Section IV. In Section V, we detail how we adopted the CIM teaching model in our courses. Sections VI through VIII respectively evaluate improvements to student capability, innovation and motivation. Section IX draws conclusions.

II. EOS Course

Nowadays, embedded systems are nearly ubiquitous, appearing in consumer electronics, cars, airplanes, etc. Increasing complexity of such systems requires new design approaches as the emphasis shifts toward high-level tools and hardware/software tradeoffs, rather than just low-level assembly-language programming skills and logic design. Thus, in designing embedded system courses, we focus not only on low-level logic design, but also on the high-level embedded software design. Operating systems (OS) play an important mediating role between hardware and software. To improve system stability, students need to understand the interaction between software and OS, OS and hardware peripherals, etc. Embedded Operating Systems (EOS) is a fundamental course focusing on the basic concepts of OS services, such as scheduler, task synchronization, device drivers, kernel primitives and so on.

An embedded operating system is different from a general-purpose operating system in that it has to interact with various hardware-specific functions and customized features in an application-specific way. The EOS course objective is to bridge the gap between academic training and industrial skill requirements. Students need to understand the fundamental principles of operating systems and learn how to use the basic kernel primitives they provide. Hence, the topics covered in this course include (refer to Fig. 1):

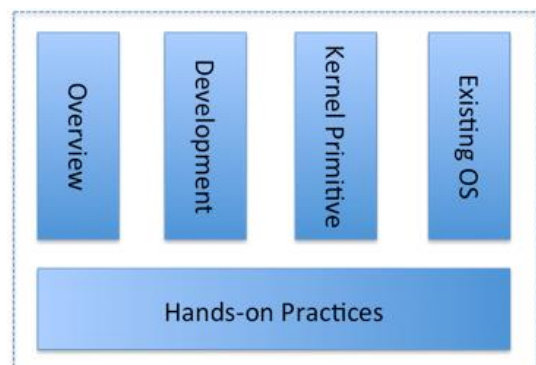


Figure 1. Original EOS syllabus

- Overview: explains OS architecture and the relationship between hardware, OS and application software.
- Cross Development Environments: introduces the

Yu-Lun Huang received the B.S., and Ph.D. degrees in Computer Science, and Information Engineering from the National Chiao-Tung University, Taiwan in 1995, and 2001, respectively. She has been a member of Phi Tau Phi Society since 1995. She is now an associate professor in the Department of Electrical & Computer Engineering of National Chiao-Tung University. She is now the Director of NCTU Center for Digital Content Production and Center for Continuing Education and Training. She has been serving the Secretary General of Taiwan Open Course Consortium since 2014. Her research interests include wireless security, virtualization security, embedded software, embedded operating systems, risk assessment, secure payment systems, VoIP, QoS and critical information infrastructure protection (CIIP), IoT Security, LTE Security, etc. She has been teaching courses of embedded systems for years and working on designing different hands-on experiments for the system engineering courses, like embedded operating systems, introduction of embedded systems, etc. She won NCTU's Excellent Teaching Award in 2015

Chao-Yang Cheng is a doctoral candidate at the Institute of Education at the National Chiao Tung University in Taiwan. His major is Educational Psychology. Flow theory, day reconstruction method, hierarchical linear models, creative teaching and learning outcome, and engineering education are central to his area of study. His current research focuses on the investigation on basic patterns of engineering students' engagement in the creativity process. Cheng is also a core member in the Project of System Engineering Education (SEE), a three-year (2014-2016) project funded by Taiwan Ministry of Science and Technology (MOST). He is responsible for developing research materials and statistical analysis. Because of his outstanding achievement, he has awarded a full scholarship for doctoral candidate from MOST.

Sunny S. J. Lin holds a Ph.D. degree in Counseling and Educational Psychology from the University of Southern California in 1995. Now she is distinguished professor of Education at the National Chiao Tung University. She teaches courses in Learning Psychology, Computer-Supported Cooperative Learning, Cognition and Instruction and Evidence-Based Instructional Design. She is director of Program of Information Education, Dept. International Cooperation & Science Education, Ministry of Science and Technology. Her research interest is in determining how adolescent learn (i.e., the science of learning) and how to help adolescents learn with the supports of interactive technology (i.e., the science of instruction). In year 2005, she received an outstanding research award, from National Science Council of Taiwan for her contribution to the advancement of interactive learning through technology. Recently, she has applied cognitive theories of learning and motivation to derive design principles of e-learning function and multimedia programs using research methods, such as longitudinal self-report and eye tracking techniques. Overall, she has more than 170 scholarly publications and a book of CSCL.



development environment in hosts and the tool chains for the target processors, including the cross-compiler, linker, loader, debugger, etc.

- Kernel Primitives of Embedded Operating Systems: introduces hardware/software co-design, basic kernel primitives, schedulers, scheduling priorities, tasks, processes, threads, task synchronization, inter-task communication, memory management, signals, timers, file systems, interrupts, interrupt service routines, I/O devices and drivers, etc.
- Existing Embedded Operating Systems: introduces the most popular embedded operating systems, such as WinCE, Windows Mobile, Embedded Linux (uClinux) [1], uC/OS-II [2], VxWorks [3], Android [4], Firefox OS [5] and Tizen [6].
- Hands-on Practices: apply lecture content to authentic tasks in our embedded systems teaching laboratory (EST Lab), as illustrated in Fig. 2.



Figure 2. Embedded System Teaching (EST) Lab.

In general, the instructor teaches the aforementioned lecture topics, and then explains the objective of each hands-on practice session. Two or three TAs then help students to address practical laboratory issues. Several quizzes and a final examination account for 70% of the final grade, while hands-on practice work accounts for the remaining 30%.

To induce creativity among EOS students, the authors designed a systematic teaching model, which integrates teaching elements including learning-by-doing, flip-flop teaching and innovative skills, and seeks to help students to better understand kernel primitives and create new opportunities for innovation.

III. The CIM Teaching Model

Based on Amabile's three critical elements [7], we propose a teaching model composed of domain Capability, Innovative skill and Motive building (CIM, Fig. 3) to

systematically integrate the aforementioned teaching elements. Based on our CIM model, we design a final project to encourage student innovation and creativity.

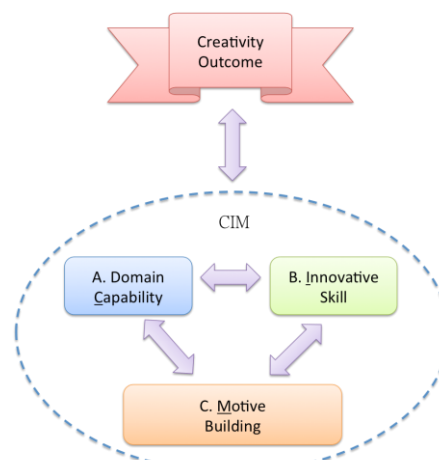


Figure 3. Capability-Innovative-Motive (CIM) teaching model.

A. Domain Capability

Capability means the professional domain knowledge that would serve as the basis to execute effective innovation processes, leading to innovations in new products, services and business models. Domain knowledge differs from course to course. We explain the domain capability of the EOS course in Section V.A.

B. Innovative Skills

Innovation brings evolution and may possibly change the way we live. As we seek to ramp up our students' innovation capabilities, we renew our focus on training and education. Since students may have difficulty coming up with totally new ideas, we encourage students to think about how to improve existing ones through a training technique for lateral thinking, SCAMPER. Defined by Eberle [9], SCAMPER is an acronym for seven thinking tools: Substitute, Combine, Adapt, Modify, Put to other uses, Eliminate and Rearrange. Students are asked to apply some of these active verbs in their feasibility study to come up with new ideas and improve their original design.

C. Motive Building

Motive building in this class focuses on three perspectives: social, intrinsic and extrinsic [10]. Social motive is built explicitly and implicitly. The instructor explicitly asks students to work together in a group, to discuss (generate) ideas, and interact with each other. The enjoyable and informal atmosphere of social media is viewed as an implicit way to promote social interaction. Intrinsic motive building is conducted through the improvement of course design, adding fun to learning-by-

doing projects and redesigning teaching environments, etc. Extrinsic motivation comes from external factors, such as rewards, prizes, etc.

IV. Teaching Challenges and Plans

When teaching embedded operating systems to students with electrical engineering (EE) backgrounds, course instructors face many challenges in the classroom, including:

[C1] Abstract Concepts: OS composition requires explaining many abstract concepts which can be difficult for students to understand and manage.

[C2] Limited Information Literacy: Though students can understand the principles of kernel primitives explained in the class, they have no idea how to take advantage of these primitives and improve the stability of their applications. They do not know how to locate the required primitives and use them correctly.

[C3] Lack of Innovation and Enthusiasm: Even though students can make good use of the kernel primitives taught in the class, they lack enthusiasm to generate innovative applications in their final projects.

[C4] Lack of Interaction and Cooperation: Lacking time for further discussion, students may be forced to limit the scale of their final projects, and may also miss opportunities to interact and cooperate with other students on larger projects.

Many of these teaching challenges can be handled by effective planning. Based on the principles of the CIM teaching model, Plan 1 (P1) to Plan 5 (P5) below are used to address these challenges.

[P1] Learn by Doing: Practice is the key to putting knowledge to effective use. Our hands-on practices can help students better understand the abstract concepts taught in the lectures. Individual hands-on practice sessions are divided into smaller assignments, enabling students to better manage operating system concepts. Plan P1 aims to improve on challenge C1 as faced by classroom instructors, and is also expected to strengthen the domain capability of the CIM model.

[P2] Group Discussion: We create several cases for students to discuss in groups in the classroom. For each case, each group needs to present the appropriate kernel primitives and their reasons for selecting them. Participation credits are given individually for each 10-minute group discussion. Instead of searching on the Internet, plan P2 encourages students to exchange experiences and study further each kernel primitive learned in the

class. This aims to improve on challenge C2, and enhance the domain capability of CIM and increase student motivation.

[P3] Innovation Methodology: Experts are invited to introduce innovation methodologies to empower students to adopt novel perspectives or breaking out of habitual mindsets [7]. Course instructors are urged to encourage student motivation. Plan P3 improves on challenge C3, and is expected to enhance the innovative skills in CIM.

[P4] Cooperation Enhancing Atmosphere: Atmosphere is an important factor in encouraging Asian students to interact with each other. We design a teaching laboratory allowing students to move their computers and desks to create new small groups for further discussion. Figure 4 shows the new EST laboratory after remodeling. Plan P4 improves on challenges C2 and C4 and is expected to enhance the domain capability and innovative skills in CIM.

[P5] Self-Learning speaks louder: Traditional closed-book examinations are abandoned in favor of open-book quizzes held at the beginning of every class. Students are encouraged to preview the course materials before each lecture. Plan P5 helps students develop their reading habits, which may help increase understanding and enthusiasm towards the subject material. P5 hence improves on challenge C3 and is expected to enhance students motivation and domain capability in CIM.



Figure 4. Remodeled EST Lab.

V. CIM in EOS 2014

To realize the above plans (P1, P2, P3, P4, P5), we redesigned the EOS 2014 syllabus based on the CIM teaching model. The revised syllabus is divided into three phases: Fundamental Concepts, Hands-on Practices and



Final Project, as illustrated in Fig. 5.

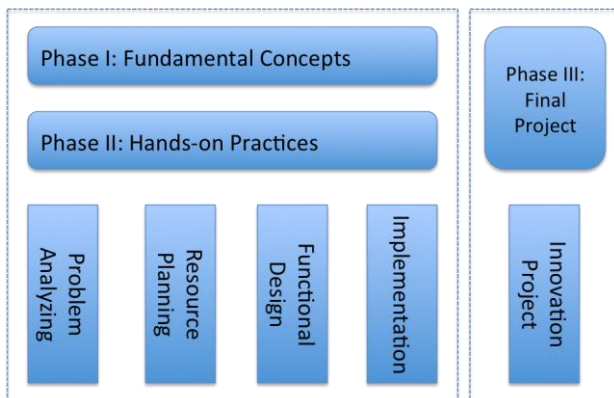


Figure 5. 3-Phase Design of EOS 2014.

Phase I emphasizes lecture content while Phase II stresses the importance of learn-by-doing. Topics in Phase I and II are staggered with students practicing assignments related to the kernel primitives covered in the previous lecture class. This arrangement allows students to develop a better understanding of the principles involved and makes the abstract concepts given in the lectures more concrete. In Phase III, we introduce students to project and innovation management practices to teach them how to prioritize the development processes and to create better solutions that can meet the existing market needs or new requirements. The following paragraphs explain how we adopt the CIM teaching model in the EOS course. Fig. 7 shows how we adopt the CIM teaching model in the EOS course.

A. Domain Capability for EOS

We intend to establish fundamental EOS capabilities through the first two phases. These lectures explain the fundamental abstract concepts (Scheduler, Communication, Synchronization, Interrupt, etc.), with each lecture followed by a hands-on practice (Task, IO, IPC, Signal, etc.) in Phase II. We further divide one practice into several smaller assignments. The assignments and practices then become the key to acquiring domain knowledge.

Since EE students are new to the fundamental concepts given in EOS, each class is previewed through handouts. An open-book assessment is used to measure how well-prepared students are before the class. These assessments are given biweekly to help establish robust study habits.

Faced with unfamiliar technical jargon and concepts, students usually have difficulty immediately mastering the information in the handouts. Asian students are generally shy to ask questions in class, and another method is needed to determine and address knowledge gaps.

Students today are highly proficient and willing to communicate in online social networks, and these venues have an additional benefit in reducing the formality of interaction with instructors.

We created a Facebook group to promote faster interaction for the fall 2014 EOS classes. In the group, the teacher and TAs can quickly respond to student questions about the handouts, practice and projects. The group was also used to share news and stories to stimulate student innovation and motivation. Figure 6 shows an example screen of our Facebook discussion group.



Figure 6. EOS Facebook discussion group.

B. Innovative Skills for EOS

The final project mentioned in Phase III (Fig. 5) wraps up the previous phases. Since the final project emphasizes innovation, instead of a specific topic, a loose project theme should be used when guiding students to come up with new ideas. In addition, one or two lectures are used to introduce SCAMPER and instruct students on generating new ideas by conducting a feasibility study for their final project, as well as reviewing peer comments for

further improvement.

B1. Loose Project Theme

The teacher could impose a project theme to guide student idea generation, but this needs to be done in such a way that it does not place additional constraints on student innovation [8]. For example, in our previous exploration, we found that suggesting a theme can result in students design products in a narrowed/limited scope closely associated with surface concepts of the given theme (e.g., schedule calendars or tour information search engines in response to a prompt to design something “for a trip”). Therefore, we recommended teachers offer a “less concrete”, “loosely-structured” theme, such as a product to “touch a traveler’s heart,” thus allowing a freer exploration of innovative ideas.

B2. Idea Generation

Initially, student groups need to discuss, propose and present to the class more than one ideas for the final project, using the kernel primitives and peripherals provided in the experimental embedded system. Comments are given by peers and the teacher to encourage ideas of higher divergence, originality and uniqueness. The teacher needs to emphasize and reemphasize the value of innovation and the prominent

before they have any value, a feasibility study is required after presenting their ideas.

B3. Dishabituation

In applying SCAMPER to their proposal, students have opportunities to redesign the functional blocks and prioritize the development details. Using SCAMPER to try out various ways of changing their initial proposal gives them a valuable chance to further avoid habituated ideas or break away from mind traps. Students can then start to implement the functional blocks and realize their proposals. In the EOS course given in fall semester of 2014, the final project ended with a demonstration and a peer assessment, fostering an atmosphere conducive to interaction and discussion.

C. Motive Building for EOS

We intend to build students’ social, intrinsic and extrinsic motives to learn EOS and to take a creative approach to the final project.

C1. Social Motivation

Social motivation is established through the instructor’s requests, such as asking students to work together in a group, to discuss while idea generation, to present ideas to other groups, etc. The enjoyable and

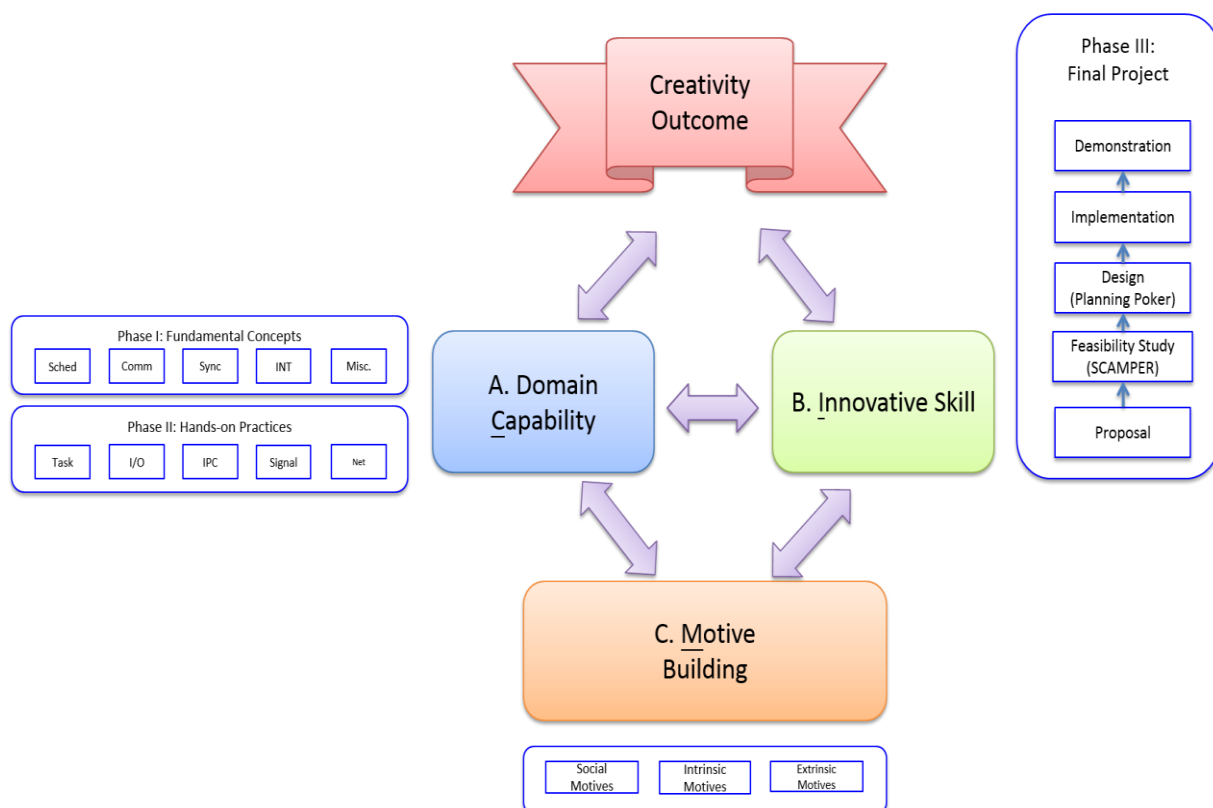


Figure 7. Applying CIM into EOS 2014.

role of unique ideas in the creativity process. Because ideas must be evaluated, refined, polished and marketed

informal atmosphere of social media is also used to promote social motive. Social motivation is relatively easy

to promote because most college students enjoy interacting informally with peers, but this type of interaction can also be extended to open-minded and friendly teachers.

C2. Intrinsic Motivation

The key to establishing intrinsic motivation is to elevate students' interest, enjoyment, satisfaction, and challenge while studying EOS and conducting the final project. Intrinsic motivation is established through the improvements to course design, adding fun in learning-by-doing projects and remodeling the teaching environment. In the class, the teacher emphasizes the value of EOS and innovation, explicitly praising effortful learning, good ideas and willingness to take on challenges.

C3. Extrinsic Motivation

To establish extrinsic motivation, creativity is explicitly given a significant weight in the final project in terms of: (1) uniqueness (unusualness, originality), and (2) breaking away from constraints (divergence, surprise), along with (3) professional appropriateness and (4) high product quality. In addition, the teacher announces that the best and the most creative product in the final project will be rewarded with 3 points for the students' final score.

VI. Evaluation on Domain Capability

EOS 2014 featured five open-book quizzes and a closed-book final examination. Each quiz was given in the beginning of the lecture class. Both correctness and hand-in order (response speed) were considered in scoring the quizzes (90 points and 10 points, respectively). In other words, the student who correctly answered the most questions in the shortest time obtained the highest score.

Fig. 8 summarizes quiz results for EOS 2014, with averages improving over the course from 48.91 to 66.83, as standard deviations fall from 21.13 to 8.68. These results and class interaction suggest that students quickly get into the habit of previewing lecture notes, allowing them to perform better on the quizzes, and thus improving their domain capability. Table 1 shows the averages and standard deviations of the final examinations of EOS 2013 and 2014.

	EOS 2013	EOS 2014
Average	63.00	69.04
Standard Deviation	17.87	13.27

Table 1. The statistics of final examinations.

The average final examination score for EOS 2014 was 69.04, which is 6.04 points higher than that of EOS 2013. The standard deviation is 13.27, which is about 5 points smaller than that of EOS 2013. Note that, due to

the limited number of experimental boards, the maximum EOS class size is limited to 27. EOS 2013 had 27 students, while EOS 2014 had 25 students.

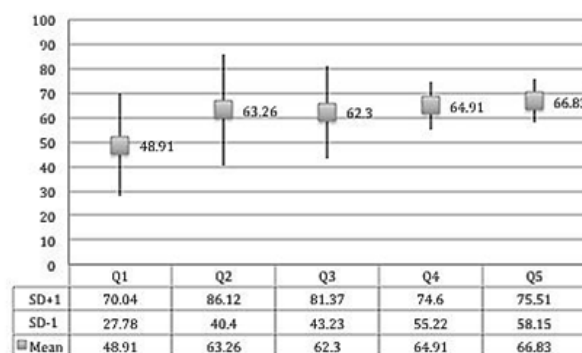


Figure 8. Quiz statistics of EOS 2014.

As mentioned above, EOS explains many kernel primitives, concepts which are not easily understood by the first-time learners. Through hands-on practice, students gain a better and more systematic understanding of the axioms involved and to approach problems from alternative perspectives.

Starting from 2014, the course introduced open-book quizzes at the beginning of lecture classes to encourage students preview the lecture notes. Compared to EOS 2013, students in EOS 2014 were more assertive in asking questions, either in person or via emails, requesting detailed explanations for how the primitives work in different cases, and seeking to better understand the primitives in lecture classes. Posing similar questions in the final examination showed students of EOS 2014 had a more comprehensive grasp of the material. The average final examination grade in EOS 2014 was about 10 points higher than in the previous session, with a considerably smaller standard deviation.

VII. Evaluation on Innovation Outcomes

In the EOS 2014 final project, students were assigned to design an assistive tool that can "touch a traveler's heart". The tool should be realized on a specified embedded platform, Microtime Creator-XScale-PXA270, equipped with an LCD, eight LEDs, four 7-Segment LEDs and sixteen keypads. The class broke into 8 groups (2-3 students per group), with each group presenting a proposal, feasibility study, implementation and the commercial value of the final outcome. A demonstration was scheduled in the end of the semester.

The final project was assessed by the instructor (accounting for 30% of the final grade), TAs (30%) and peer groups (40%). We assessed each project in terms of the domain capability and the innovation (uniqueness and divergence) displayed. In the final presentation, each

group should explain the design, demonstrate their project and promote their ideas to the peer groups, TAs and instructor. The winners were awarded an extra 3 points on their final scores. Group G4 won the best work award for its demonstration of a multifunctional trip assistance tool featuring a path planner, a language translator, a guide to local attractions and an alarm. Group G2's effort was ranked last among the eight groups for a trip advisor featuring a tour recommendation engine, a pre-tour reminder, and an achievement subsystem.

For the domain capability, G4 produced the best work by adopting many kernel primitives taught in the lectures, including multithreads, semaphore, queue, etc. G4 also integrated modern programming skills and a model-view-controller (MVC) to separate the data, display and computing functions. The characteristics of MVC programming made it easy for the members of G2 to collaborate. G2 only used the keypads and LCD to respectively enter parameters and display results, and thus only realized part of their design in a personal computer, instead of the given embedded platform.

In terms of innovation, G4's path planner was designed by planning paths between the source and destination points at beginning. Later, however, G4 ADAPTEd (using the 3rd thinking tool, Adapt, in SCAMPER) their path planner to additionally consider multiple immediate points. This adaptation gives travelers additional flexibility in planning a trip with multiple destinations. Since the hardware spec for the final project does not provide for audio peripherals, students MODIFIED (the 4th thinking tool, Modify, in SCAMPER) their audio translator as a text translator (using LCDs). Students translated sentences from English to Japanese using English transliteration; e.g., "How are you?" is translated to "O genki desuka?", as illustrated in Fig. 9.

G2's design displayed some innovation. Originally, they wanted to incorporate a recording subsystem. But, after applying SCAMPER, they ADAPTEd (the 3rd thinking tool, Adapt, in SCAMPER) the recording subsystem as an achievement subsystem. In addition to recording the

traveler's itinerary, the achievement subsystem also encourages a traveler to visit as-yet-unvisited attractions. However, they failed to complete their design on the required hardware, which was one of the mandatory requirements for the final assessment, and thus significantly hurt their final grade.

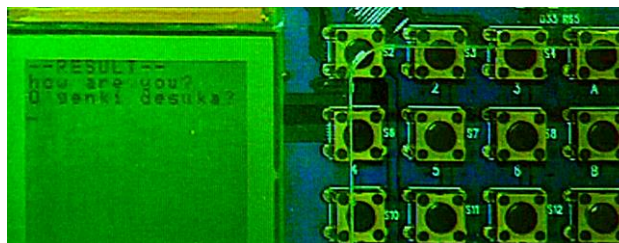


Figure 9. Demonstration of G4's work.

VIII. Evaluation on Motive

Compared to emotional data, momentary data regarding motives, confidence of domain capability and engagement in innovation were collected to evaluate student motivation in the 2014 fall EOS course.

A. Day Reconstruction Method to Collect Motive Data

The Day Reconstruction Method (DRM) [11] is implemented to collect immediate and momentary evidence of students' motivation, confidence of domain capability and engagement in innovation. DRM is a rather new technique that asks participants to recreate the previous day's activities and records the cognitive and affects experiences associated with those activities using a questionnaire. Data collected by DRM has timestamp and situation-stamp so as it could connect the situations (EOS, other classes, computer use, on-campus activities etc.) with the feelings that the respondents experience in the situations.

In this study, students' momentary self-reports were repeatedly collected over five selected weeks of various types of EOS classes (e.g., I/O lecture, SCAMPER

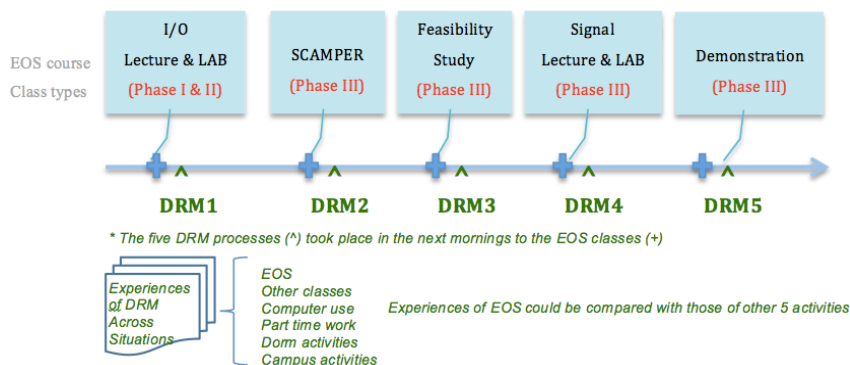


Figure 10. DRM momentary self-reports: repeatedly collected over five selected weeks of various types of EOS classes.

introduction, working on final project, course test, and final presentation). The five time-points of DRM data collection are illustrated in Fig. 10. The 1st and 4th DRM data were collected after hands-on practice sessions (Phase II in Fig. 5) following EOS lectures (Phase I in Fig. 5). The 2nd, 3rd and 5th DRM data points were respectively collected after the teacher introduced SCAMPER, the feasibility study (project design phase) and the final demonstration of the final project (Phase III in Fig. 5).

B. DRM Measures and Data Analyses

The five DRM processes took place the next morning following EOS classes. The investigators first asked volunteer participants to write a whole-day diary about the activities in the previous day and then determine the number of time-episodes per day. Then, for each episode students were asked to answer a short list of questions regarding their cognitive experiences (e.g., I paid full attention, I was involved in creative work) and affect experiences (I felt happy, I was anxious) during activity participation.

A total of 996 episodes were collected from 17 students who completed five DRM sessions (on average, students reported 11.72 episodes in a given day). Of these, 274 experiences were regarded as EOS class activities and learning (on average, students reported 3.22 episodes in a given EOS class). Descriptive statistics were calculated and pie charts were drawn. ANOVA tests were conducted to compare means (Confidence of domain capability, Engagement in innovation and motives) across activities in various situations and EOS classes.

Fig. 11 summarizes student responses to the question: “Why are you taking the EOS course?” The response of “I personally wanted to take EOS” shows higher degree of autonomy compared with the response of “I had to because of department requirements” or “I had nothing else to do.” Although EOS is a required course, 39% students replied that they elected to take the course (high autonomy) while 61% took the course because of external requirements or with no clear reason (low autonomy).

C. Student Perception on General EOS Situation

In Fig. 12, most responses from students in the modified EOS class indicate they saw the teachers (45%) and peers (31%) as learning companions, while 20% indicated they felt they were learning in relative isolation, suggesting the class was largely successful in overcoming passive learning habits, and that this interaction played an important role in the learning process

D. Confidence of Domain Capability, Engagement in

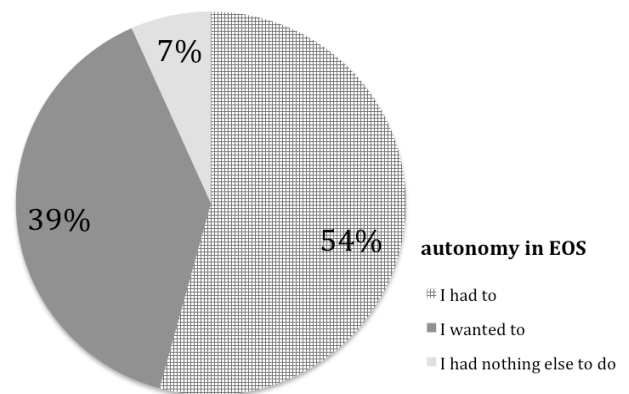


Figure 11. Why are you taking EOS?

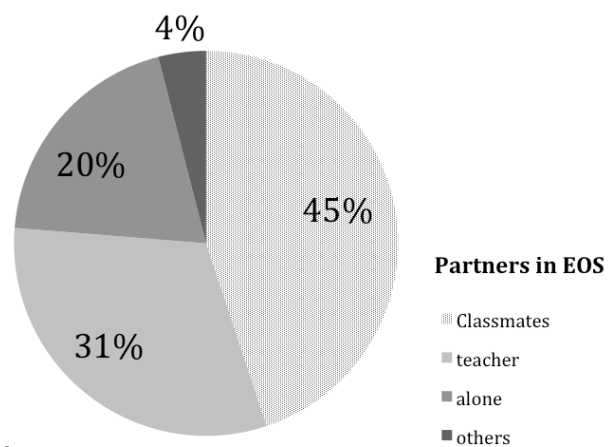


Figure 12. Who do you interact with in EOS classes?

Innovation and Motivation

Table 2 shows the Means, Standard Deviations, and ANOVA tests to compare the means of <A. confidence of domain capability>, <B. engagement in innovation> and <C. motivation>.

When students were asked to rate factors contributing to their confidence in their domain (System Engineering) capability, they indicated that EOS class activities were found to be highly effective promoted, as opposed to in-dorm and on-campus activities, which were given the lower rates. This implies that students considered that they can barely perceive the domain capability in dormitory and campus activities.

EOS activities were found to be more conducive to fostering creativity than other activities, such as using computers, studying in other classes, engaged in in-dorm activities, at work and on campus. Students reported feeling highly engaged in creativity process and idea generation in EOS classes. Teachers offered clear goals for creative products and thus created a sense of high challenge.

Results also showed that EOS classes were conducive to establishing strong motivation. Students



Situations	EOS		Computer		Other		Dorm		Part-time		Campus		ANOVA F (5, 862)
	Activities		Use		Classes		Activities		Work		Activities		
DRM data	M	SD	M	SD	M	SD	M	SD	M	SD	M	SD	
<i>A. Confidence of Domain Capability</i>													
Domain capability	6.38^a	2.13	4.56	2.69	4.59	2.39	<u>2.84^b</u>	2.30	4.41	2.88	<u>2.74^b</u>	2.33	55.19***
Career Importance	6.61	2.01	6.32	2.10	5.78	2.28	5.90	2.56	6.08	2.14	6.19	2.51	2.95*
<i>B. Engagement in Innovation</i>													
Creative	4.82^a	2.41	4.36	2.34	4.08	2.27	<u>2.99^b</u>	2.04	4.15	2.36	3.26	2.24	14.11***
Idea generation	6.24^a	2.02	5.58	1.88	5.64	2.03	<u>4.13^b</u>	2.35	5.55	1.95	<u>4.59^b</u>	2.18	22.13***
Clear Goal	6.02^a	2.06	5.63	2.00	5.86	2.24	<u>5.11^b</u>	2.57	5.43	1.98	5.52	2.31	3.49**
Challenge	6.35^a	2.04	4.84	2.77	4.89	2.51	<u>2.73^b</u>	2.26	4.42	2.84	2.75	2.27	56.71***
<i>C. Motives</i>													
Positive Feedback	6.25^a	2.15	5.26	2.25	5.80^a	2.25	<u>3.83^b</u>	2.55	5.28	2.20	4.64	2.36	22.148***
Concentration	6.71^a	1.98	6.18^a	2.18	5.55	2.37	<u>4.42^b</u>	2.68	5.92^a	2.32	4.94	2.52	21.62***
Time past fast	5.64^a	2.46	5.72^a	2.41	4.72	2.85	<u>3.92^b</u>	2.46	5.31	2.34	4.38	2.65	12.40***
Involvement	6.25^a	2.09	6.20	2.05	5.60	2.11	5.62	2.37	<u>5.30^b</u>	2.19	6.23	2.39	4.72***
Active	5.78	2.26	6.24	2.03	<u>5.26^b</u>	2.38	6.08	2.28	5.92	2.04	6.79^a	2.07	5.82***
Happy	<u>4.72^b</u>	2.25	5.32	2.32	4.56	2.21	5.31	2.58	4.92	2.27	6.11^a	2.72	6.93***
Demanding	6.20^a	2.04	4.48	2.56	4.38	2.52	<u>3.01^b</u>	2.29	4.14	2.66	<u>2.78^b</u>	2.27	50.97***
Anxious	3.96^a	2.53	3.39	2.43	3.08	2.39	<u>1.89^b</u>	1.72	3.16	2.49	2.20	1.97	17.51***

Note. *p < .05. **p < .01. ***p < .001.

^a (higher) > ^b (lower): means with mark ^a is significantly higher than means with mark ^b.

Table 2. The statistics of DRM momentary data.

reported that the classes provided positive experiences in terms of positive feedback, concentrated learning, a sense of time passing quickly, and high involvement. Nonetheless, students felt the activities in EOS were demanding, creating a certain level of anxiety, and their continued enthusiasm for the class despite such anxiety can be taken as an indication of strong motivation.

IX. Conclusions

For years, engineering education has placed a strong emphasis on establishing and improving students' domain capabilities and problem-solving skills. In the paper, we present a the redesign of an EOS course syllabus to foster creativity in system engineering education. We first examined the challenges we faced in such a course. Second, we designed a methodology considering the relationships between the domain capability, innovation skills, motivation and innovative outcomes. We redesigned the syllabus and applied SCAMPER in the final project. Then, we collected and analyzed student feedback by implementing the Day Reconstruction Method. The analysis shows that:

- 1) instructors and peers both played important roles in the class
- 2) students were highly engaged in the creativity process and idea generation
- 3) students were highly concentrated in the EOS class activities

- 4) students felt time passed quickly in the EOS class and
- 5) the course was challenging and the students felt a heightened level of anxiety in the classroom.

At the end of the semester, a final demonstration was given to present the outcomes of final project along the theme of a product that can “touch a traveler’s heart”. The idea should be practical enough to be realized in a given embedded platform using the domain capabilities taught in the class. After applying SCAMPER, we found that all project outcomes showed a certain degree of creativity.

X. Acknowledgment

The authors would like to thank Professor Bing-Fei Wu, Professor Yon-Ping Chen and Yi-Kai Wang for sharing their experiences in teaching system engineering courses. Their contribution provided the authors with a deeper understanding of students’ learning status.

This work was financially supported in part by Ministry of Science and Technology, Taiwan, under grant numbers MOST 102-2511-S-009-010-MY3, 102-2511-S-009-013-MY3 and 102-2511-S-009-012-MY3.

References

[1] Embedded Linux/Microcontroller Project. [Online]. Available: <http://www.uclinux.org/>



- [2] J. J. Labrosse, *Microc/OS-II The Real-Time Kernel*, R & D Books, 1998.
- [3] Wind River VxWorks. [Online].
Available: <http://www.windriver.com/>
- [4] Android. [Online].
Available: <https://www.android.com/>
- [5] FirefoxOS. [Online].
Available: https://en.wikipedia.org/wiki/Firefox_OS
- [6] Tizen. [Online].
Available: <https://en.wikipedia.org/wiki/Tizen>
- [7] T. M. Amabile, *Creativity in Context: Update to the Social Psychology of Creativity*, Boulder, Colorado: Westview Press, 1996.
- [8] T. B. Ward, M. J. Patterson, and C. M. Sifonis, "The Role of Specificity and Abstraction in Creative Idea Generation," *Creativity Research Journal*, vol. 16, no. 1, pp. 1–9, 2004.
doi: [10.1207/s15326934crj1601_1](https://doi.org/10.1207/s15326934crj1601_1)
- [9] R. F. Eberle, "Developing Imagination Through Scamper," *Journal of Creative Behavior*, vol. 6, no. 3, pp. 199–203, 1972.
doi: [10.1002/j.2162-6057.1972.tb00929.x](https://doi.org/10.1002/j.2162-6057.1972.tb00929.x)
- [10] M. A. Collins and T. M. Amabile, *Handbook of creativity*, Cambridge, England: Cambridge University Press, 1999.
- [11] D. Kahneman, A. Krueger, D. A. Schkade, N. Schwarz, and A. A. Stone, "A Survey Method for Characterizing Daily Life Experience: The Day Reconstruction Method," *Science*, vol. 306, no. 5702, pp. 1776–1780, 2004.
doi: [10.1126/science.1103572](https://doi.org/10.1126/science.1103572)

