

Teaching Systems Engineering to Undergraduates; Experiences and Considerations

Gerrit Muller

Buskerud University College

Gerrit.muller@hibu.no

G. Maarten Bonnema

University of Twente

Copyright © 2013 by Gerrit Muller. Published and used by INCOSE with permission

Abstract. Undergraduates need a teaching style that fits their limited experience. Especially in systems engineering this is an issue, since systems engineering connects to so many different stakeholders with so many different concerns while the students have experienced only thus far only a few of these concerns and met only few stakeholders. Students need to become aware of the inherent ambiguities, uncertainties, and unknowns in the systems world, in contrast to the focused world of mono-disciplinary engineering. There is a difference between the more traditional engineering disciplines (mechanical, electrical, etc.) and the upcoming and broader disciplines like industrial design engineering and systems engineering itself.

Introduction

In the systems engineering community there is a lot of debate when we should start to teach systems engineering. Can we effectively teach system engineering to undergraduates? We will discuss the differences between teaching systems engineering and teaching other engineering disciplines to undergraduates. Factors that play a role are (lack of) experience of students and learning styles.

We formulate goals of teaching systems engineering to undergraduates and propose an active learning paradigm to achieve them. As consequence, we see a different role than the conventional role for the teacher. We elaborate the structure of such course for system design, and we discuss how to select cases for student projects. Finally, we show examples and discuss our experiences of teaching in this way.

What makes Teaching Systems Engineering Special?

Didactics and Learning and Teaching Styles

Felder provides a good overview of learning styles (Felder 1996). He explains how teachers can accommodate students with different learning styles. Felder recommends among others:

- *Teach theoretical material by first presenting phenomena and problems that relate to the theory (sensing, inductive, global)*
- *Balance conceptual information (intuitive) with concrete information (sensing)*
- *Make extensive use of sketches, plots, schematics, vector diagrams, computer graphics, and physical demonstrations (visual) in addition to oral and written explanations and derivations (verbal) in lectures and readings.*

- To illustrate an abstract concept or problem-solving algorithm, use at least one numerical example (sensing) to supplement the usual algebraic example (intuitive)
- Provide class time for students to think about the material being presented (reflective) and for active student participation (active)

Bonwell and Eison explain what active learning is, why it is important, and what obstacles there are to deploy active learning (Bonwell 1991). Kolb and Schön describe the relevance of experience for learning (Kolb 2001, Schön 1983). Our target group for education is not yet experienced. However, it is worthwhile to read Kolb and Schön about the importance of experience and the use of reflection to learn. The approach described in this paper is to mimic this learning process at a very small and modest level. Smith provides a nice overview and discussion about experiential learning (Smith 2001). We follow an approach that is somewhat similar to (Bonnema 2005) with emphasis on *active learning* and *experiential learning*. In that learning setting, the systems engineering contents, have to be combined with engineering material (sensors and actuators) to design systems (e.g. a home climate system). This mimics a system architecting problem and stimulates students to apply what they learn.

The Student Perspective

Students that start working in a company enter a completely new world, where their original technical discipline is a small part of that context. Figure 1 illustrates the diverse context that the student enters. When teaching undergraduates, we need to be aware that most of them have not been in such context, and hence miss a frame of reference for most of the aspect shown in Figure 1. Bonnema shows that all these different aspects require different thinking approaches, see (Bonnema 2012, Boardman 2009).

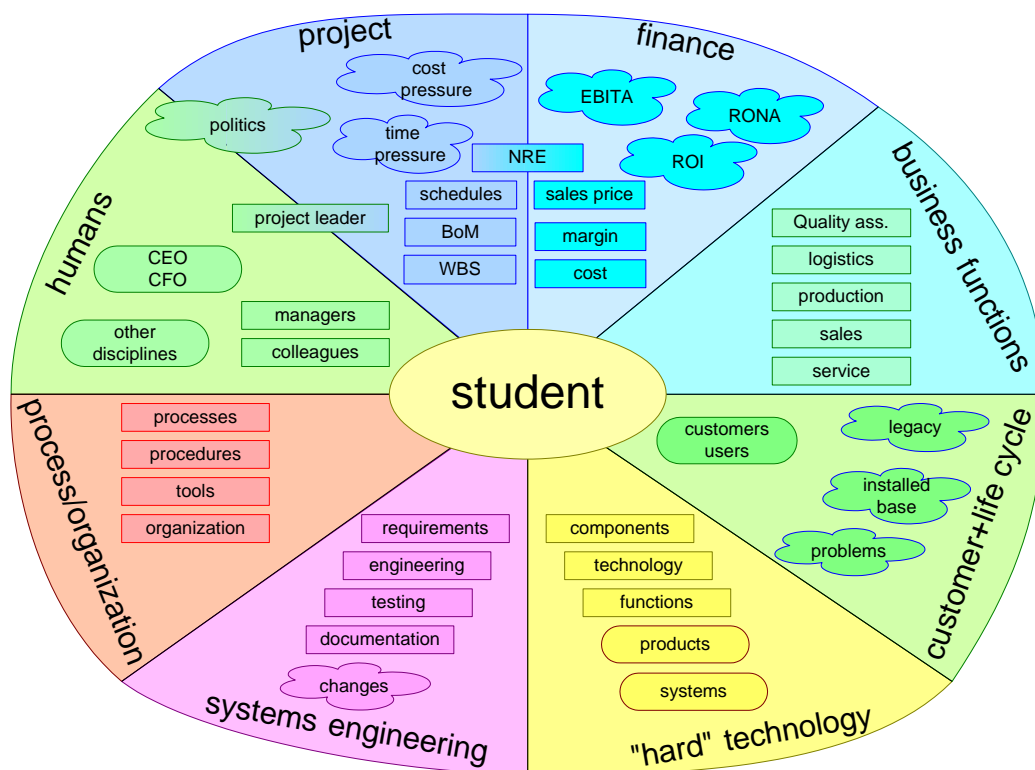


Figure 1. The context that students enter when they join an organization.

Challenges in Teaching Systems Engineering to Undergraduates

Systems engineering differs in a number of aspects from engineering disciplines:

- The scope is broader; systems engineers are working partially on technical problems and are facing many non-technical issues at the same time. Examples are the human factor and harsh natural conditions.
- Problems are often ill defined.
- Problem and solution space is full of unknowns, uncertainties, contradictions, and ambiguities.
- Problems and solutions are so heterogeneous that methods, techniques, and formalisms often have to be adapted to the situation. Systems engineers need the competence to select and adapt the right tool for a given problem.
- In the complex, many-dimensional field of stakeholders and concerns there is no single unique “best” answer.

For mono-disciplinary engineering, we can more or less invert the characteristics:

- The scope is limited; engineers tend to work on technical problems in a technical context.
- Problems are well defined. For example, is the bridge strong enough, or does the component keep cool enough?
- Problem and solution space are narrowed down so that there are few unknowns, limited uncertainties, and no or few ambiguities.
- Problems and solution are homogeneous allowing application of well-defined methods, techniques, and formalisms. Examples are the design flow for digital electronics with a complete tool chain and the use of Bode plots to reason about control characteristics.
- Many engineering problems that students face in their education are analytical and have one correct answer, such as the most efficient motor, or the most stable controller.

These differences have impact on the teaching style used by teachers. Teachers in engineering need to transfer knowledge and skills. Traditionally, teachers use lecturing (for knowledge transfer) and training (for skill development) dominantly. In engineering, students learn to be analytical and to apply a reductionist approach. The main challenge is to help students to open up for the different world in systems engineering, such that they can learn complementary ways of thinking and working, supported by the methods, techniques, and formalisms from the systems engineering world. Typical terms that are used in systems engineering are holistic (or “big picture”), synthesis, and soft systems.

The second challenge of teaching undergraduates is that undergraduates typically have little or no experience with working in larger organizations. They miss a frame of reference to position and appreciate methods and techniques that systems engineering offers. Students may perceive methods as common sense or motherhood statement. For example, a phased approach or the V-model is so logical that students perceive them as trivialities.

Proposed Approach to Teach Systems Engineering at Undergraduate Level

Goals of Teaching Systems Engineering at Undergraduate Level

As we have seen, students at undergraduate level have developed a mindset that in most aspects is the inverse of the systems engineering mindset. We propose that the initial ambition of teaching at undergraduate level is to create awareness of many system considerations and to provide insight in available systems engineering methods, techniques, and concepts. The system aspects to make students aware of are:

- the impact of working in a large organization, e.g. processes, organizations, roles, responsibilities, economy and financials

- the communication challenges between:
 - various technical disciplines
 - various less-technical stakeholders
- the ill-defined and multi-dimensional nature of system problems with uncertainties, unknowns, ambiguities, dynamics, and conflicting needs and goals
- the impact of external conditions on the system and its design, e.g. human behavior, and natural phenomena
- the system life cycle

Proposed Teaching Paradigm

The importance of experience in systems engineering education fits better to *action* and *experiential* learning styles. The unusual (for the students) nature of systems engineering requires a teaching format that is motivating for the students. Teaching systems engineering without an accompanying engineering (subject matter) background or experience is impossible. The students have to use that subject matter either by building upon previously acquired knowledge and competences, and/or by presenting new material in parallel to the systems engineering course. The combination should introduce them into the world of combining engineering with systems engineering. Once students have gone through the loop of appreciating means for problems they never saw before, we can repeat such a cycle several times.

We propose to immerse the students in real-life cases. One way is to use project education. Another way is to use role plays as a means to facilitate students to enter a new world and to start experiencing such new world. Project education is time consuming for the students, role-plays can be chosen if the study load is limited. We elaborate here the role-play, since in many undergraduate curriculums the available time for systems engineering is limited. We see basically two main types of role plays where a team of students play:

- a management team (leadership perspective)
- a specification and design team (subject matter perspective)

Teams typically contain 3 to 5 students; the risk of larger teams is that not all students participate actively. The teacher provides a case to work on, e.g. you manage a company that designs and delivers apple plucking robots. This context is provided with a few sentences only, such that it is a rather open assignment. The case has to appeal to the students by relating to their knowledge. The teacher facilitates a guided process alternating theory with applying it in the role play. After applying it, students briefly report and discuss this step.

In the management team role the focus is on process, organization, people, business, and financial aspects. The final assignment of this part of the course can for instance be a business plan to convince investors that your company is a good investment.

In the design team role, the focus is on the subject matter, e.g. understanding customers, determining requirements, selecting concepts, partitioning, defining interfaces, modelling and allocating functionality, managing qualities (e.g. performance, cost, reliability), and making technology choices. The final assignment for this part of the course can be a specification and design presentation for a managerial review; e.g. a 20 slide presentation outlining the core of the specification and design and its rationale to get permission to continue with the next development phase.

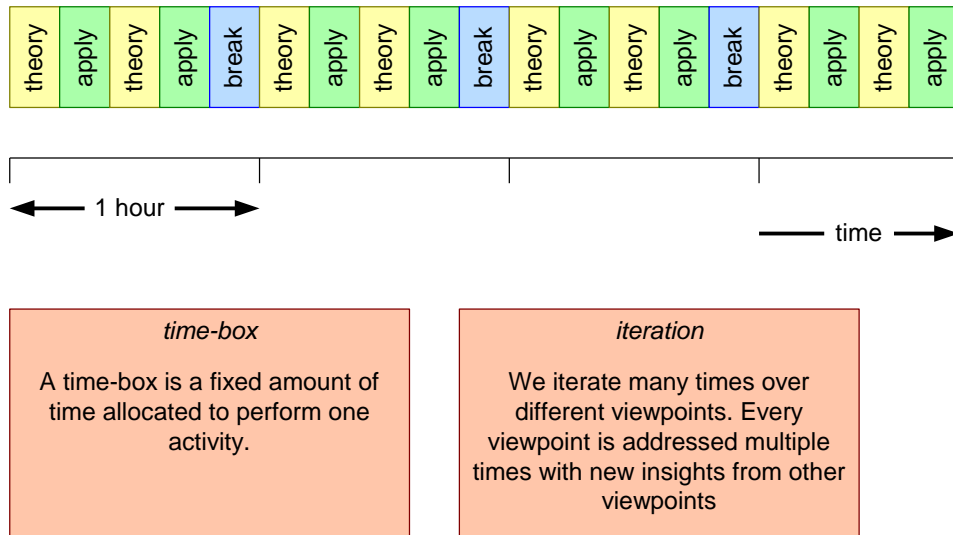


Figure 2. Didactic model when teams play management or design role.

Figure 2 illustrates the alternating theory and application. The duration of every box is limited, typically in the range 5 to 20 minutes. The limitation of theory lecturing is chosen such that students are kept active and engaged in the case. Students easily lose their attention, when the teacher explains theory. The teacher should only lecture theory according to the Just In Time (JIT) principle: theory that students can appreciate, because they have hit their heads, and the theory can be applied afterwards, so that it falls into place.

The limited duration to apply theory is based on the time-box and iteration principles. Time-boxing is a technique where a fixed amount of time is allocated for an activity. When the time has passed, then the next step is made even if the previous activity is not completely finished. The idea behind time-boxing is that the same topic is addressed multiple times; it is revisited in next iterations. When other viewpoints have been explored in separate time-boxes, then the re-iteration of a topic will be done with new insights gained from other viewpoints. We effectively guide students in small steps through an exploratory journey, somewhat similar to what happens in practice in real organizations. However, in the role play we have accelerated the time and we provide more guidance to achieve the desired learning outcomes.

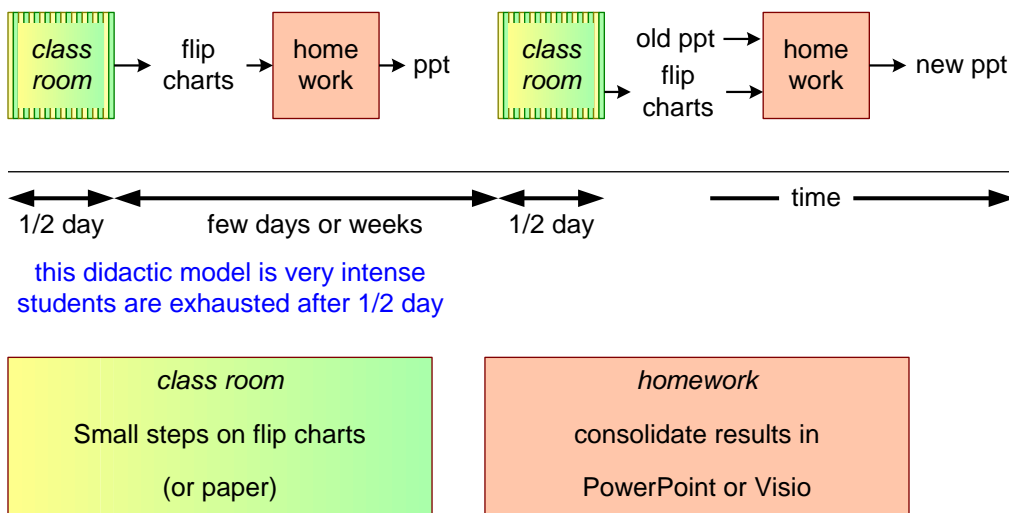


Figure 3. Didactic model over longer time period with homework.

After every time-box, we have a few teams presenting their results. They get a few minutes to present, followed by questions and a limited discussion. The duration of the time-boxes and the size of the assignment per time-box depend on the experience level and maturity of students. At

graduate level we often use time-boxes in the range of 30 to 45 minutes, with relatively broad assignments per step. However, undergraduates miss the frame of reference to cope with these broader assignments and need more guidance to progress. We have frequent breaks, to allow students to digest and refresh. In our experience, this teaching model is quite intensive. After half a day, students are saturated.

Between half day sessions students get homework that typically consists of a consolidation of manual drawings and notes into electronic form using, for instance, PowerPoint or Visio. We expect the students to include small enhancements based on increased insight in the homework version. The homework deliverable evolves over time into the final delivery. The frequency of the home work depends on the course schedule. For instance, if students have two sessions per week, we often have one homework assignment per week.

Role of the teacher

The teacher plays a crucial role in engaging the students and launching them on a steep learning curve. Every good teacher engages the students. However, the challenge for systems engineering is to help the students to step in this new unknown world. The contribution of the teacher is to

- Guide the students through a journey despite the lack of experience of the students.
- Stretch students one step at a time; if the teacher stretches too fast, then students may lose connection with case and theory.
- Regularly force students out of their comfort zone to help them to become aware of other thinking and working paradigms than they are used to.
- Provide feedback on their intermediate deliverables. The feedback should help students to develop systems thinking skills. Nevertheless, some feedback on the specifics of drawings and analysis is required too.
- Help students to reflect on their experience; what did we do, why? What can we learn from this?
- Provide theory JIT (Just In Time: appreciation and application). Most steps build logically on previous steps from didactic perspective. E.g. we need to understand customer needs to be able to agree on requirements.
- Illustrate theory with examples from practice. Some examples from the teacher's experience can illustrate theory. Keep in mind that most students do not have experience to relate to.
- Keep the pace high so that all students stay engaged, and to get short iteration cycles; shorter iteration cycles help to experience the benefit of iteration.
- Initiate frequent breaks (this approach costs lots of mental energy); at least one break per hour.
- Unfreeze students: let them sketch and stimulate creativity and imagination. Be aware that most education so far suppresses this behaviour.

Selecting a case

A good case has multi-disciplinary aspects. However, the original discipline of the students should be present clearly, to ensure engagement of the students at the start. The application must be chosen such that students have affinity with the topic. For instance, for a group of mechanical engineering students in Norway, we took a tree cutting robot as case. Mechanical aspects are dominantly present; however, vision and control require electronics and software as disciplines too. Wood harvesting is common in Norway with a countryside full of woods. The case was introduced as follows:

background: Less young people are willing to work in the wild and mountainous areas in Norway, Canada, or USA to cut trees for wood production.

product: Robot that supports the cutting and processing of trees so that less people are needed.

Structure of the System Design part of the course.

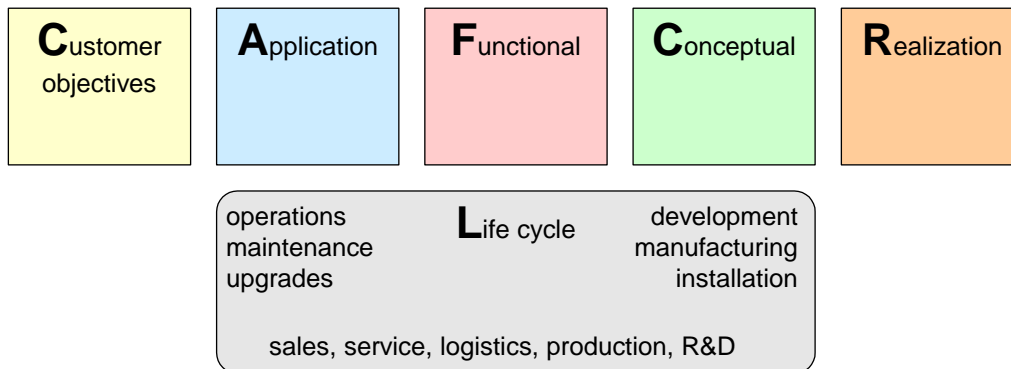


Figure 4. CAFCR+ model with six views as used in the system design.

The system design part of the course is based on the so-called CAFCR-model (Muller 2011). All course material is available on-line at <http://www.gaudisite.nl/BachelorSDallSlides.pdf>. The CAFCR model proposes 5 views on a system: The Customer objectives and the Application view describe what and how of the customer context of the system. The Functional view describes the system as a black box, and the Conceptual and Realization views describe the internals of the system. The CAFCR+ model, shown in Figure 4, adds a life cycle view covering all aspects from conception to decommissioning.

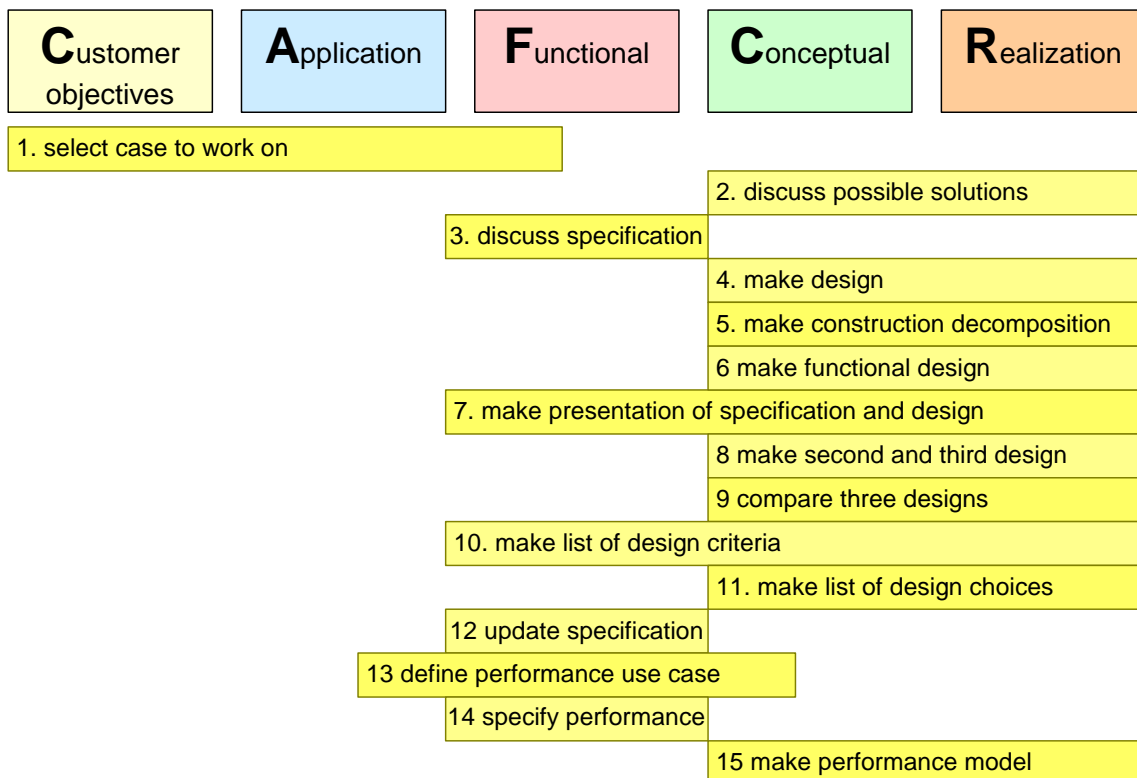


Figure 5. The first 15 exercises during sessions 1 and 2.

Figure 5 shows the exercises of the first two sessions related to the CAFCR-model. Only the very first step touches the customer context: what kind of system will we be working on. Most

of the following exercises invite the students to think about possible solutions. In this way, we let students work in the relatively well-known territory of technology. We start free format to stimulate students to think creatively about the solution, and gradually offer them some typical systems engineering means, such as partitioning (make construction decomposition), and functional design. Regularly, we stretch the students to think in black box terms: what should the specification of the system be?

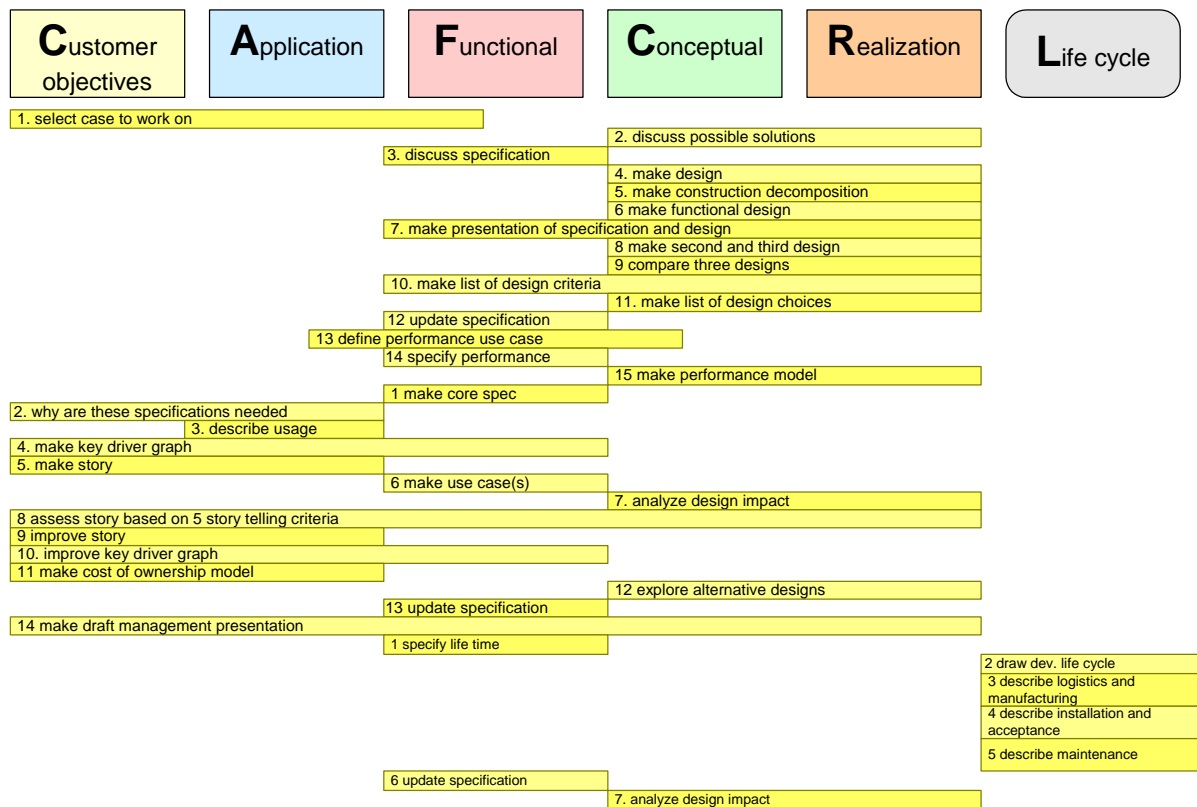


Figure 6. All exercises of the systems design part of the course mapped on CAFCR+.

Figure 6 shows all the exercises mapped on the CAFCR+ model. In 36 steps we gradually broaden students from the technical world into the customer and life cycle worlds. These 36 steps require 5 to 6 sessions of about half a day each.

Figure 7 elaborates the main learning objectives in this process:

- Multi-disciplinary: be aware of other disciplines with their particular methods, techniques, formalisms, and models.
- Multiple concepts: the need to explore multiple alternatives and to avoid that designers fixate on one solution.
- Requirements: understand that requirements are specific, measurable, and verifiable, and describe “what?” rather than “how?” (e.g. describing the systems as “black-box”).
- Keep iterating: the need to understand design decisions in relation to its context and vice versa.
- Customers: be aware of the customer perspective; what are their needs and concerns?
- Humans, nature, and “reality”: be aware of the broad variation in humans and natural circumstances, ranging from emotions to natural disasters.
- Life cycle: Understand what happens from idea conception to commissioning of a system.

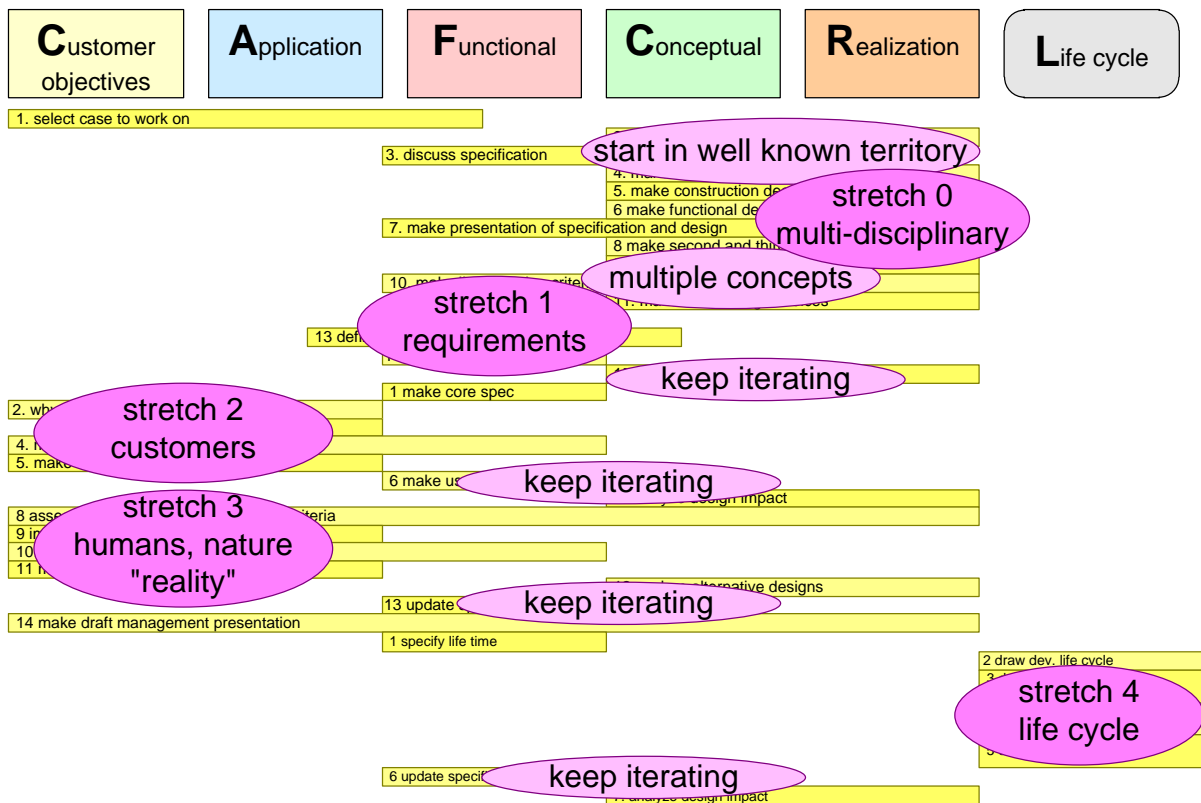


Figure 7. The program annotated with the didactic objectives

Experiences of Teaching Systems Engineering in this way

Classes where we applied this approach

We have applied the approach as discussed in this paper at Buskerud University College in Kongsberg, Norway, for third year Mechanical Engineering bachelor students from 2008 until 2011. These students follow a 5 ECTS course in Systems Engineering, where we teach 1/3 as described in this paper. The rest of the course is conventional weekly lecturing.

We have used the same approach in Eindhoven, the Netherlands, as 2 day workshop “Cars in context” with bachelor and master students in Automotive Engineering, and as 3 day workshop in Ålesund, Norway, for master students in Maritime Engineering. Typical class size is between 15 and 30 students. These students had little or no working experience.

In all these classes, the number of contact hours is only 2 to 3 days. The total study load of this course is only 4 to 8 days, including contact hours. Compared to the complete study of 2 or 3 years full-time, this course is a small part, about 1%, of the complete study. In most cases, the systems engineering course is a mandatory part of the curriculum. Driving force behind the mandatory nature is vision from the study leaders in combination with demand from local industry.

Some examples from class rooms

In this section, we show examples from class rooms to illustrate the kind of cases and the way that students approach them. We hope that the illustration show that students who get the same input go in entirely different directions. The illustrations show also that students are quite capable of creative and artful work; as teachers we have to unfreeze students and to stimulate them to express themselves creatively.

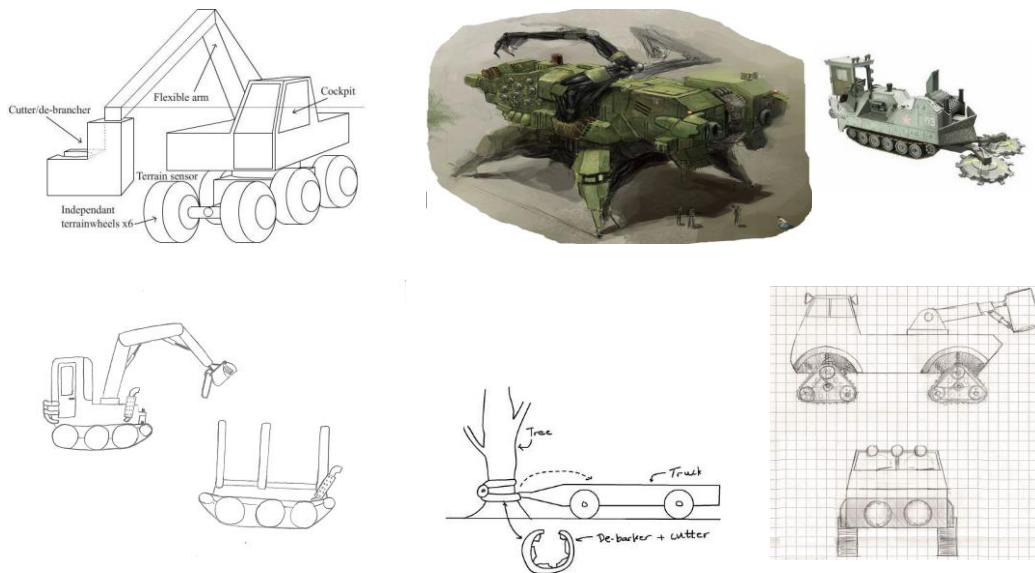


Figure 8. Examples of tree cutting robots

Figure 8 shows some of the visualizations that students made for the tree-cutting robot at different stages in the process. The three bottom figures are drawn manually on paper. The top left drawing is made with computer assistance. The drawing in the middle, kind of mechanical monster on legs, and the picture at the top-right hand side come from Internet. Searching on the web is good, since it can help students to quickly see new alternatives and to get new ideas. However, care should be taken that these students do not stay too superficial; an exercise in copy-paste-modify does not have sufficient didactic value. Often, having students make order of magnitude calculations will eliminate the “fantastic” solutions. The monster on legs illustrates another problem that we can hit in the class room. These students started with a design that was too far from reality and the teacher did not manage to guide them back to realistic solutions. Such unrealistic solution keeps hitting the students in the later analysis.

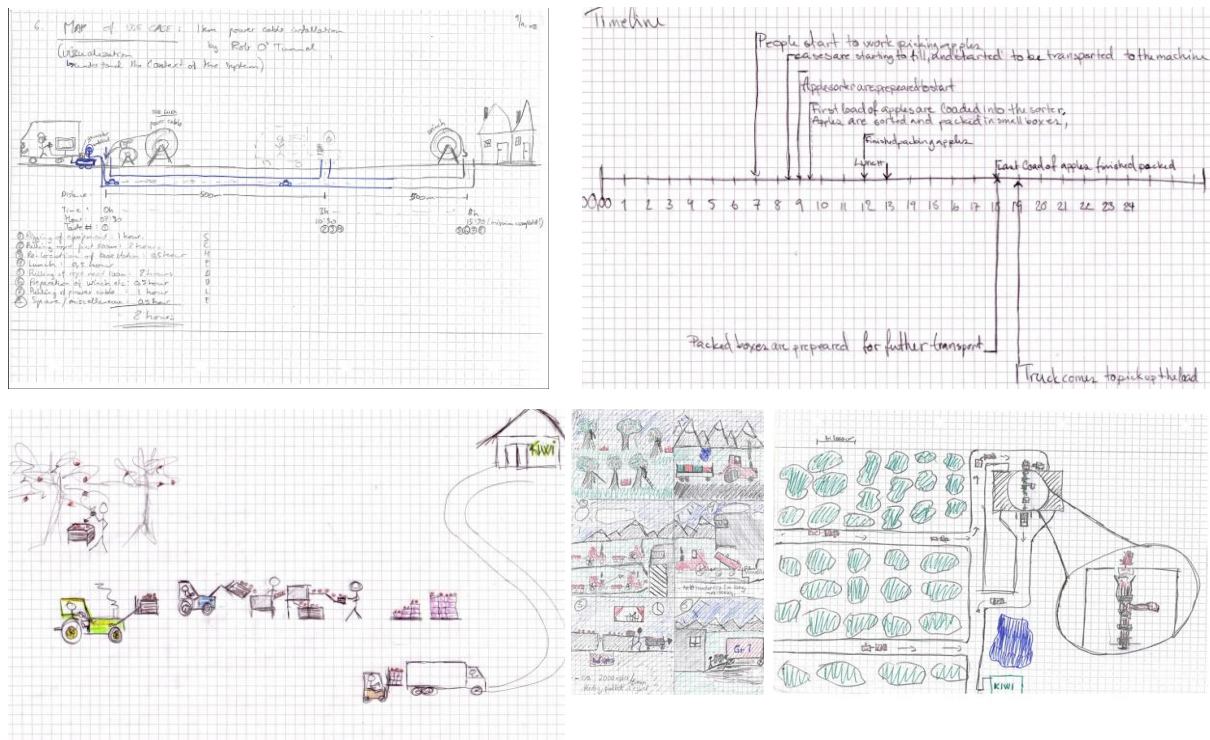


Figure 9. Examples of story-telling applied on cable robot (top-left) and apple-plucking robots.

One of the steps in the process is that the students have to create a user story to force them into a customer perspective, e.g. operational thinking (Bonnema 2012). Figure 9 shows several user stories, and their use for further analysis. Top-left is a cable-laying robot. The drawing shows among others distances, practical hurdles, and the operational steps. Top-right is an example of a time-line, when does what happen in the user story? Bottom-left is a farm with an apple plucking robot in operation. Bottom-middle is a cartoon-like user story. Finally, bottom-right shows a map of an apple farm to show where is happening what.

Figure 10 shows flip charts on a wall in a class room where we were teaching “cars in context”. The case for the students is to design an inner-city electric cargo van. These flips show the variation of assignments that students go through during the system design. We see here drive cycles, a map, sketches of the van, a functional model, a block diagram, a (visual) user story, user cost price estimation, and a concept selection matrix.

Feedback from the Students

We have applied a “light weight” evaluation method: asking all students to write free-format benefits and concerns per course day. Benefits that are commonly mentioned are:

- It is fun, entertaining, and inspirational (~50% of the respondents)
- Seeing multiple system perspectives, e.g. CAFCR (~30% of the respondents)

Other aspects that are frequently mentioned are time-boxing and iteration, understanding the customer’s point of view, and team work. Main concerns that were mentioned are:

- Sometimes confusing or chaotic (~30% of respondents)
- Struggles with team and home work (~30% of respondents)
- Time-boxes are too short (~20% of respondents)

Some students had an idea of how to use the CAFCR model, but had the feeling not yet to understand the model.



Figure 10. Example of flip charts of a class room that gets completely hung by flip-charts during the sessions. The case in this example is an inner city electric cargo van.

Pitfalls

Students can come up with too funny concepts or stories. This distracts from the specification and design objectives. Some fun is OK (see benefits). However, when it gets too funny then guide the team back into realistic concepts or stories.

Some teams get stuck in an unrealistic proposal or stick to the initial solution, for example the 4-leg monster in Figure 8. The same team proposed an on-board nuclear power generator. The idea behind iteration is that a design team can afford to explore exotic solutions for a limited amount of time. However, for the learning experience the teacher has to guide the team to a realistic solution in one of the early iterations. Be aware that most people, including our students, show mental inertia; e.g. they tend to continue in a chosen direction. Part of the learning is that adaptations of the direction are a fact of life and necessary. Creativity and facilitation techniques are helpful in assisting here.

In several occasions we had students that missed sessions. This teaching model requires mandatory participation. The class will be able to cope with a small amount of missing students, e.g. because of illness.

This teaching model is quite dynamic. It forces students to follow a quick succession of viewpoint changes. Not all students have the mental agility to follow. For some students this teaching model does not fit their personal learning style.

Considerations

We have been using time-boxes from 5 to 20 minutes at undergraduate level. If time-boxes get too long, then the process gets too slow. Nearly always students will complain that they need more time. This is one of the many small learning aspects: in practice, designers always need more time. When we apply a similar approach at graduate level, then we use larger assignments with time-boxes up to 40 minutes.

We have run this course a few times with full days for teaching. However, undergraduate students are saturated and exhausted after half a day. At graduate level, using longer time-boxes we are able to teach a full day with this model.

The timing of the course and all steps will vary, depending on the students and the case. We use the program with flexibility; sometimes a few steps are skipped.

Benefits

The sessions can be a lot of fun for students and teacher; we consider this a clear benefit, since learning that is fun is more motivating.

In the open atmosphere, some interesting concepts popped-up. For example, in a case for robot supported showering for elderly people, we got a wide variety of concepts. The variation of solutions is an indication for the perceptiveness of the students.

In several courses some nice visualizations or animations were shown. For example, a complete animation of the tree cutting robot that cuts and peels trees. Such visualizations support communication and learning.

We observe an increase of awareness while teaching. For example, when we make the step from the requirement specification to human and nature context, students discover problems like uneven ground, rocks, rivers, swamps, rain, storm, and many more practical complications.

Conclusions

We have described an approach to teach systems engineering at undergraduate level that aims at making students aware of the broadness of “systems”. The proposed learning model is quite active. Challenge for the teacher is to engage the students and to help them discover aspects that are outside their current frame of reference. Teaching in this way can be rewarding. However, it is demanding for students and teachers the like, at the same time.

References

- Boardman, J., B. Sauser, L. John and R. Edson (2009). The conceptagon: A framework for systems thinking and systems practice. International Conference on Systems, Man and Cybernetics, 2009. SMC 2009. IEEE.
- Bonnema, G. M., I. F. Lutters-Weustink and F. J.A.M. van Houten (2005) *Introducing Systems Engineering to Industrial Design Engineering Students with hands-on experience* Proceedings of the 18th International Conference on Systems Engineering (ISCEng'05)
- Bonnema, G. M. (2012) *Thinking Tracks for Integrated Systems Design*, 1st Joint International Symposium on System-Integrated Intelligence 2012
- Bonwell, C. C. and Eison, J.A., *Active Learning: Creating Excitement in the Classroom*, In ERIC Digest September 1991, online available at <http://www.ntlf.com/html/lib/bib/91-9dig.htm> retrieved 3 August 2012
- Felder, R.M., *Matters of Style*, ASEE Prism, 6(4), 18-23, December 1996, available online at <http://www4.ncsu.edu/unity/lockers/users/f/felder/public/Papers/LS-Prism.htm>

Kolb, A. and Kolb D. A. (2001) *Experiential Learning Theory Bibliography 1971-2001*, Boston, Ma.: McBer and Co

Muller, G. (2011), *Systems Architecting: a Business Perspective*. CRC Press.

Schön, D. (1983) *The Reflective Practitioner*, New York: Basic Books

Smith, M. K. (2001). 'David A. Kolb on experiential learning', *the encyclopedia of informal education*. Retrieved [3 August 2012] from <http://www.infed.org/b-explrn.htm>.

Biography



Gerrit Muller.

Gerrit Muller, originally from the Netherlands, received his Master's degree in physics from the University of Amsterdam in 1979. He worked from 1980 until 1997 at Philips Medical Systems as a system architect, followed by two years at ASML as a manager of systems engineering, returning to Philips (Research) in 1999. Since 2003 he has worked as a senior research fellow at the Embedded Systems Institute in Eindhoven, focusing on developing system architecture methods and the education of new system architects, receiving his doctorate in 2004. In January 2008, he became a full professor of systems engineering at Buskerud University College in Kongsberg, Norway. He continues to work as a senior research fellow at the Embedded Systems Institute in Eindhoven in a part-time position. All information (System Architecture articles, course material, curriculum vitae) can be found at: Gaudí systems architecting <http://www.gaudisite.nl/>



Maarten Bonnema

G. Maarten Bonnema is an associate professor in multidisciplinary systems design at the Laboratory of Design, Production and Management of the Faculty of Engineering Technology at the University of Twente. He studied Electronic Engineering, and did a two year designer course on Technical Systems. He has worked as a Systems Engineer at ASML. He received his PhD in 2008 from the University of Twente. In 2006 and 2007 he was involved in the design of a wafer stepper at MAPPER lithography (part time). In the young multidisciplinary systems design group, the various tracks of research aim at supporting system designers, conceptual design and mechatronic design by improving multidisciplinary communication. Also systems thinking is researched. He has a broad teaching expertise spanning design in general, industrial design, and systems engineering. He is involved in various student projects as a tutor.