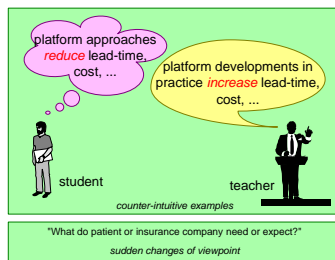# Didactic Recommendations for Education in Systems Engineering

-

Gerrit Muller

University of South-Eastern Norway-NISE

Hasbergsvei 36 P.O. Box 235, NO-3603 Kongsberg Norway

gaudisite@gmail.com

Abstract

Teaching systems engineering differs from teaching a mono-disciplinary course, because the focus is much more on skills and less on transferable facts. The teacher must trigger a learning process in the students that stimulates the student to become active with the subject in a perceptive, reflective, and explorative way. This paper provides a number of recommendations for interaction, illustration, soft skill development, the use of media and student feedback.

version: 0   status: concept   September 6, 2020

1 Introduction

The academic forum at INCOSE 2004 in Toulouse discussed the availability of systems engineering education. A few postgraduate programs are available; Stevens Institute, MIT and the University of South Australia were dominantly present at INCOSE, see the links section at the end for the URL's of these programs. The graduate programs were subject to a lot of discussion. Analysis of graduate programs as presented by Dagli[2] shows that most graduate programs are in fact conventional mono-disciplinary programs with about 120 hours of system subjects, 5% of the complete program.

The terms systems engineering and systems architecting are used as synonyms in this article. The function system architect in Philips corresponds more or less with the Systems Engineering function as promoted by INCOSE. We will use the architecting in the Philips context and engineering in the INCOSE context.

Bonwell[1] describes the benefits of active learning and the barriers to introduce this learning method. In our own systems architecting courses we have experimented with several forms of active learning. Figure 1 shows the Systems Architecting curriculum as proposed by Philips and the Embedded Systems Institute (ESI). The experience of the courses at Philips and ESI is that teaching Systems Architecting requires an active learning style, more active than normally applied in graduate engineering courses. The reason that an active learning style is important for architecting is that the development of interpersonal skills rather than know-how transfer is the goal of the course. We formulate didactic recommendations in this article to teach Systems Engineering or Architecting courses.

The curriculum by Philips and ESI relates the maturity of the system architect, as shown at the top of Figure 1, to the type of know-how and skills that is recommended as education. The conventional engineering disciplines are shown for completeness at the left-hand side. The more mature architects are, the more their educational needs shift to skills and non-technical know-how.

ESA (26 days) The Embedded Systems Architecting (ESA) course provides technical broadening to engineers with a few years of experience and the ambition towards an architecting job. It also provides in the module stakeholders a preview on the non-technical aspects that are important for an architect.

EXARCH (5 days) The Execution Architecture (EXARCH) course focuses on the performance and real-time aspects of embedded systems design.

CAFCR (5 days) The course Multi-Objective Embedded Systems Design, based on CAFCR provides multiple methods to translate customer objectives in design decisions. The CAFCR method is based on five views: Customer Objectives, Application, Functional, Conceptual, and Realization Views.
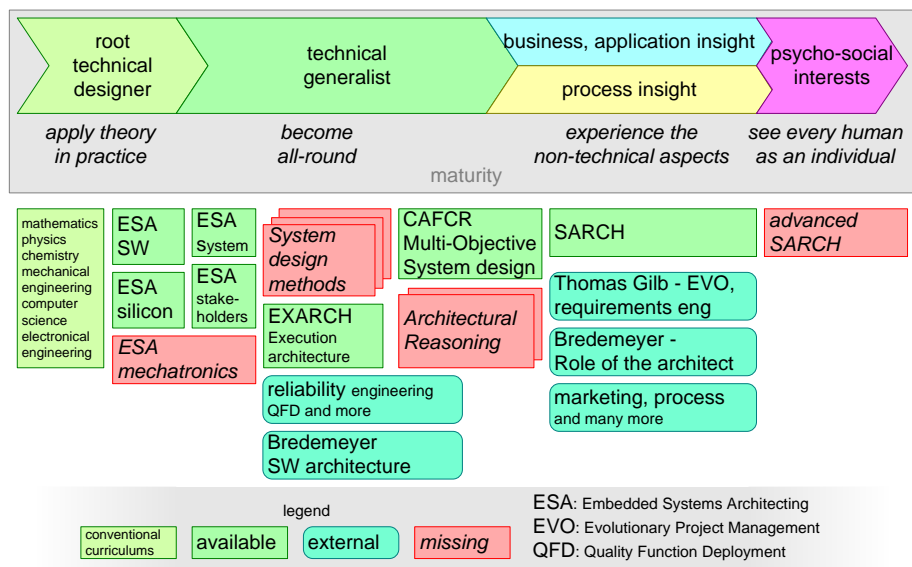
Figure 1: Systems Architecting Curriculum

SARCH (5 days) The Systems Architecting (SARCH) course provides an overview of all non-technical aspects of the architecting job. Nine different viewpoints, ranging from process and role to requirements and roadmapping, are used to put the systems architecting job in a framework.

The target audience of this type of courses consists of highly educated, professional engineers. They seek education to mature from mono-disciplinary engineers into multi-disciplinary systems engineers. The conventional graduate learning style as used for mono-disciplinary subjects is one-way lecturing and individual training, mostly mathematical tasks. Such a conventional learning style does not fit well with the audience and the multi-disciplinary subject. The multi-disciplinary nature of architecting requires good interpersonal skills, in order to get all stakeholders involved. The training style for these courses should support this. In contrast many mono-disciplinary skills can be trained individually. Straightforward classroom teaching might be sufficient for mono-disciplinary skills.

This paper has been published in the proceedings of the INCOSE 2005 in Rochester. However, in this version the figures have been added that have been used during the presentation at the conference itself.

2 Combining Passive and Active Learning

The purpose of the systems architecting courses is to provide students with insight in methods. This requires some theoretical background of the subject to be transferred. A very effective way to support the transfer of know-how is to show examples. The theoretical part provides overview and reflection; the example brings the subject alive. However, providing theory and examples is still insufficient for the students to completely digest the information. Actively working with the theory develops the required practical skills. Felder gives a clear description of learning styles in[3].
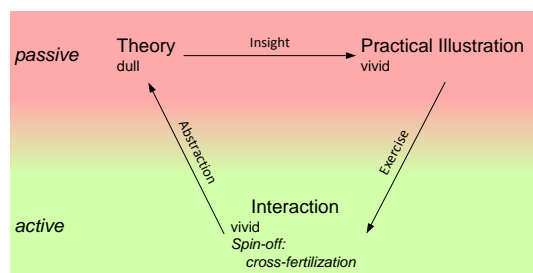

Figure 2: Active vs Passive
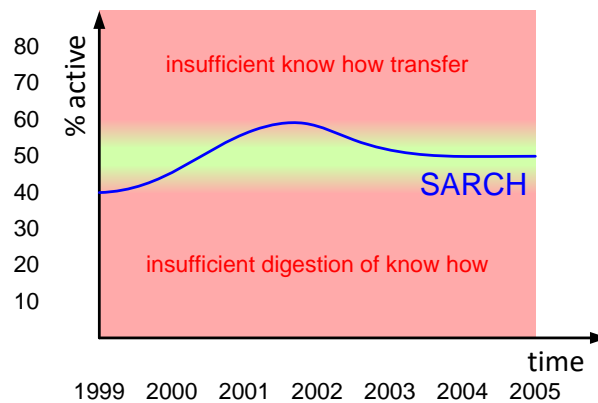
Figure 3: Finding the Balance Active-Passive for the SARCH course

The course setting dictates a strict time limit. As teachers we have to find a balance between the passive, theory and illustration, and more active, exercise, parts of the course, see Figure 2. When the 5-day Systems Architecting (SARCH) course at Philips started a passive/active ratio of 60/40 was used, see Figure 3. In other words a lot of information was provided, somewhat less time was scheduled

to actively use the material. It soon became clear that active use of the material is crucial for the appreciation. In the next courses the ratio was gradually shifted to 40/60: less theory and more exercise. At this point the students started to complain that they learned insufficient new theory. At this moment the ratio is about 50/50. The SARCH course is a high-density course, providing material over a wide variety of subjects. In more focused courses, such as the Execution Architecture (EXARCH) course or the Multi-Objective System Design (CAFCR) course the ratio shifts to 25/75. The exercises in these courses need intensive coaching by the teacher, because the exercise itself becomes the learning vehicle.

The iteration between theory and illustration is much more fine-grained. Most theoretical models need illustration to be appreciated. The teacher has to use own experience to find illustrative examples. In some cases the participants themselves have nice examples available. See the section about Interaction for more hints. The course material covers the theoretical parts, because this is generic information. The illustrations are often not available in an easily accessible format, due to the confidentiality. Examples are most effective if someone who has been involved explains them. If somebody else has to tell the illustrative story then a lot of the authenticity is lost, reducing the actual value of the example.

3 Interaction

+ Pose questions to the students

+ Keep the communication open in all directions

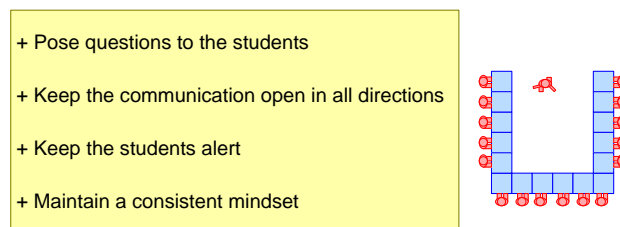+ Keep the students alert

+ Maintain a consistent mindset

Figure 4: Interaction

Actively involved students absorb and digest more material. The teacher should stimulate student activity as much as possible. The teacher must be interested in the students and curious in their interests and concerns. This is very stimulating. In particular for students that are reserved or even unwilling at first. Probably they have a valid issue or contribution. One way of achieving this relationship between students and teacher is by making the lecture highly interactive.

In the following sections we will discuss a variety of interaction measures:

- Posing questions to the students:
  - Provocative

- Explorative

- Inviting experiences

- Keep the communication open in all directions:

  - Allow or even stimulate discussion

  - Managing two-way communication, the parking flip

  - Creating an open and safe learning environment

- Keep the students alert by changes of viewpoint and counter-intuitive examples

- Maintain a consistent mindset throughout the course

3.1 Posing questions to the students

Every module starts by posing a question to the audience. Examples of questions:

Provocative: "What is the most important process in your company?" The provocation is that the "most important" in this context is meaningless. Nevertheless posing the question this way forces the students to differentiate between important or core processes and less important supporting processes.

Explorative: "What are the deliverables of an architect?" The students have to be active themselves by coming up with the deliverables. In the slipstream of this question a number of related discussions are held, such as: "What are deliverables?"

Inviting experiences: "Who has seen a roadmap?" followed by the question "What was the contents of this roadmap?" or "What is the value of this roadmap for the organization?" The answers to these questions already illustrate a significant part of the theory. The theory is then shown quickly to provide the overview.

3.2 Keep the communication open in all directions

Allow or even stimulate discussion The presentation and question must invite the students to bring their own experiences forward. Discussing the own experience and comparing it with someone's other experience helps the students to enliven the know-how. It will also trigger discussion between the students, enriching the insights even more.

Managing two-way communication, the parking flip The course setting must create a mood where the students feel invited to participate actively. This means that asking questions must be possible anytime. Sometimes these questions will

trigger a discussion between the participants. The teacher has an important task here: at the one-hand to nourish this mood, but at the other hand to look after the course program and the related time limits. The teacher needs facilitation skills to play this balancing game. One of the useful facilitator tools is the Parking flip. The Parking flip is a flip chart that is attached to the wall of the classroom, clearly marked as being the Parking flip. Students may park questions or comments any time on this flip. The teacher can suggest to park questions or discussion items, when the question or discussion deviates too much from the current module objectives. At the beginning and at the end of every module some time is reserved to address parked issues.

Creating an open and safe learning environment The rules of interaction during the course are set at the beginning of the course. To ensure two-way communication with all students an open and safe learning environment must be created. The main rules to achieve this are:

- Argue in a constructive way, no heat seeking missiles allowed!

- Stupid questions don't exist

In reality "stupid" questions turn out to be questions from a different viewpoint that offer new insight. Some questions have insufficient value for the whole group in that case the questions are answered off-line, for instance during the break. Questions that are rather diverging must be taken off-line to prevent confusion of the students.

The teacher must also facilitate a balanced contribution of all participants. Silent participants must be stimulated to contribute, while monopolization by outspoken students must be prevented.

3.3 Keep the students alert by changes of viewpoint and counter-intuitive examples

The classroom setting may unconsciously seduce the students to listen in a lean-back mode. Besides a lively presentation style it helps to wake the student by sudden changes of viewpoint, or by providing counter-intuitive examples. The students should react to these events. If they stay inactive then this is a clear sign that the teacher has lost the audience. The teacher can either wake them up explicitly, or announce a break.

During the presentation of a platform based product line approach a sheet with provocative disadvantages, such as higher development cost and longer time-to-market, most often triggers a lively discussion.
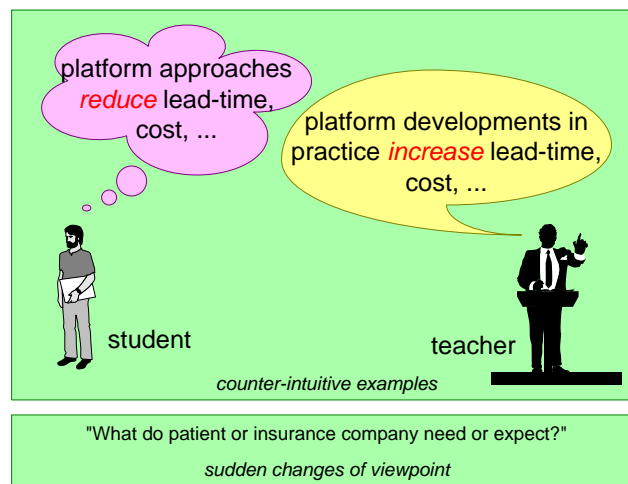
Figure 5: Keeping the students alert

3.4 Be aware of cultural differences

The cultural background of the students influences their ability to go along with interactive lectures. This subject deserves quite more attention than the limited space available in this paper. We limit the recommendations here to be aware of cultural differences in hierarchy and in individualism versus collectivism. Dutch students, for instance, are much less sensitive for hierarchy than German students. The hierarchical sensitivity can be addressed by inviting the student to step into the managerial viewpoint. Students is Asian countries tend to have a more collective mindset, than the highly individual European and American students. Good listening skills and coaching skills help in bridging the individual and collective mindset.

3.5 Maintain a consistent mindset throughout the course

The job of a system architect requires a specific mindset. This mindset is woven through the entire course in a limited set of threads:

- Be customer, market, and result oriented

- Use common sense

- Use multiple viewpoints

- Be constructively critical

- Maintain your integrity and credibility as an architect

- Use facts, be specific

- Communicate clearly and to the point, provide overview

All the interaction aspects mentioned (posing questions, open communication, and keeping them alert) are used to stimulate this mindset. For instance, a statement of a student: "We cannot do this, because the amount of software is way too large" can be countered with questions such as:

- "Why do we need this amount of software?"

- "How much work is required with this amount of software?"

- "If the customer really needs this, how can we serve the customer anyhow?"

Multiple viewpoints support a better description of the inherent complexity of the system and its context. Most system descriptions get too complex due to the one-dimensional approach. Concerns that don't fit in the dominant description dimension are scattered through the entire description, making it impossible to maintain a consistent view of the system.

4 Soft Skill Development

Skill development is an important goal of all SE-oriented courses. In the following sections we will discuss how the soft skills presenting, teamwork, and self-reflection are continuously trained as a spin-off of the exercises. The last section discusses the importance of providing the students with balanced feedback.

4.1 Presenting

Communication skills are important. In these courses the participants are trained all the time in communication and presenting, as spin-off of the exercises. The result of every exercise must be presented in 5 minutes, using 1 or in some cases 2 flip charts. This format forces the students to report the essentials. There is no time, nor space on the flip charts, to tell the complete story. In real life system engineers have to be able to present their case on the spot in less than 5 minutes. The presenter's role is rotating; so all students get a few turns.

4.2 Teamwork

System engineers always operate as member of larger teams: project team, spec team, design team, et cetera. Developing the skill to work in teams is also a spin-off of the exercises. All exercises are done in teams of about four people. The teams are kept relatively small to ensure the active participation of everyone. In larger teams there is more room to silently lean back. The power of a team is that the members learn to cross borders in multiple dimensions. For instance, hardware

and software engineers cooperating across the discipline boundary, and relating functional analysis to performance analysis. Members of a team have to learn that making errors is inevitable. Explicit communication in a team helps to resolve errors and improves the progress.

4.3   Self-reflection

The development of skills is effectively triggered by self-reflection. In the discussion of the results the students are often invited to take a step in the "meta"-direction: "What did I/we do?", "What are the results?", "Are there alternatives?". The final module of the course may invite the student to cumulate the insights in a personal plan for the near and longer term future.

4.4   Providing Balanced Feedback

The teacher must provide feedback to the students about exercise results and the process towards the results. It is important that this feedback is balanced, and that the strong points are specific and explicit. The points to improve must be mentioned also specifically and explicitly. Suggestions must be given how to improve, preferably building on the strong points. Many students have to build up the self-confidence to give this kind of presentation or to operate in an open and vulnerable fashion in a team. The positive feedback is needed to enforce this self-confidence.
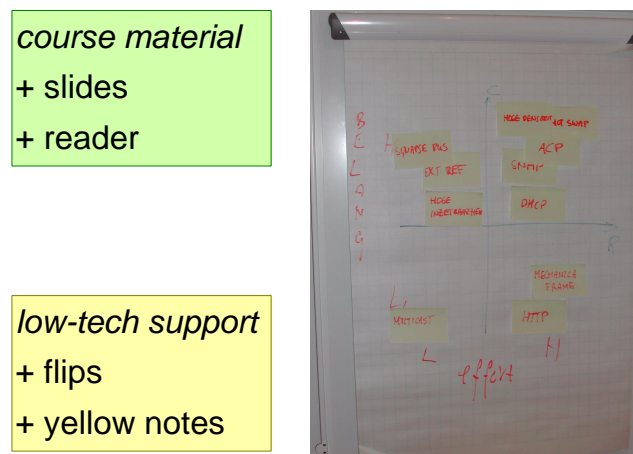
5   The Use of Media

Figure 6: The use of media

Gerrit Muller   University of South-Eastern Norway-NISE
Didactic Recommendations for Education in Systems Engineering
September 6, 2020   version: 0   page: 9

A rich collection of media can be used: slides, animation, lectures, flip charts, sticky notes, laptops with all kinds of interactive programs and much more, see also Figure 6. From the didactics viewpoint media are means and not goals. It is recommended to select media that are supportive and fit in a natural way. Avoid complex tools and blazing animations, because the risk that they do not work in the right way is significant. Even if they operate as planned a dominant medium is often more distracting than supporting. A picture or diagram is a powerful means of communication. A good picture explains more than can be said in a thousand words. As an example we discuss our way of working below.

At Philips and ESI the courses are given as block courses in a classroom. The blocks have lengths between 2 and 5 days. The course material is available on the Internet: `http://www.gaudisite.nl/CourseDescriptions.html`. The students get the URL a few weeks before the course, with the message that they may prepare themselves, but this is not a prerequisite. The availability of the material is a service to those students that feel more comfortable after preparing themselves. The slides and the reader are also provided as handouts at the start of the course.

During the lecture itself the slides are projected and the contents is discussed. To facilitate faster browsing by simple forward/backward key presses, this material does not use any animation. The exact slides shown during a lecture depends on the interaction. No two courses follow identical sequences of slides. The combination of flexibility and very simple browsing works effectively.

The flip chart is heavily used for the interaction. For instance an exploratory question, such as "What are the deliverables of an architect?" results in a brainstorm-like list, where all the answers are written on a flip. In a short discussion the answers are annotated ("Is a schedule really a deliverable of an architect? No, it is a project leader deliverable."). These flip charts are then attached to the walls of the classroom. During the course the discussed material and its illustrations are present on the walls, forming a collective memory. This helps when discussing related issues. For instance, when discussing the activities of the architect, the teacher can refer back to the deliverables: many activities never show up in one of the deliverables!

Another low-tech means are yellow sticky notes. These can be used to collect individual inputs, for brainstorms, and to create easily adaptable maps. The notes are typically put on flip charts that are also attached to the classroom wall. For brainstorms it is useful to first use these yellow notes, because one captures the unbiased inputs, including inputs from the more silent students.

6 Exercises

The exercises invite the students to compare the provided theory with their own experiences. The reality of the day-to-day work is always more troublesome than the provided theory.

6.1 Exercise instruction

The instruction for the exercises must ask for specific information, e.g. show the operational organization where you are operating, mention the names of the people involved explicitly. The specificity of the exercise is important, because the general superficial answers avoid the learning points. In the above example nameless organization diagrams look nice, while assigning names suddenly triggers lots of discussions amongst the students.

The description of the exercises must be short and open. Highly detailed exercise descriptions bias the students into a narrow field of answers. Providing an approach of a few steps can support some exercises. For instance, start with an individual brainstorm to determine the most critical timing requirements, and then create collectively a control diagram and a timing diagram.

6.2 Team size

Teams must be sufficiently large to create some experience of the group dynamics. At the other hand the size is limited to ensure the active participation of all team members, including the more quiet team members. The experience is that the above two requirements are best met with teams of four persons. A team of three persons works also, although the group dynamics is already much less. In exceptional cases teams of 5 people are used, although in many cases the most quiet team member stays then too much in the background. The number of teams is also limited, because it is difficult to keep all teams involved with too many teams. Four teams is about the maximum where all teams keep paying attention to the work of other teams.

6.3 Duration

The duration of the exercises is about 40 minutes. If more time is provided then not much more progress is made. To get an exercise really a step further significant more time must be allocated, several hours. The reporting of the four teams together in the classroom takes about half an hour, up to an hour if many discussions are triggered.

Figure 7: Exercises

7 Student Feedback

The students in these courses have rather varied backgrounds in education and domain. This makes every course unique. It is recommended to explicitly ask for feedback every day. We use the benefit and concern method: every student writes at the end of the day a yellow note with benefits and a yellow note with concerns. This method forces the students also to give balanced feedback.

The teacher reads all the feedback. The benefits provide feedback how much of the course objectives have been met. The concerns must be screened for issues that require immediate action. Examples of concerns and immediate action are:

- "The room is too hot" requires an adjustment of the climate control

- "Discussions are too long" means that more use of the parking flip is needed

- "I got lost in all information" is counteracted by providing a short overview

The teacher has to show that the student feedback is read and used. This can be realized by starting the next day addressing a few concerns of the previous day.

References

[1] Charles C. Bonwell and James A. Eison. Active learning: Creating excitement in the classroom. http://www.ntlf.com/html/lib/bib/91-9dig.htm, ERIC Clearinghouse on Higher Education, Washington, D.C., September 1991.

[2] Cihan H. Dagli. Undergraduate education in systems engineering in usa. Presentation at the Academic Forum INCOSE 2004 in Toulouse, June 2004.

[3] Richard M. Felder. Matters of style. `http://www.ncsu.edu/felder-public/Papers/LS-Prism.htm`, ASEE Prism, 6(4), 18-23, December 1996.

[4] Gerrit Muller. The system architecture homepage. `http://www.gaudisite.nl/index.html`, 1999.

History
Version: 0, date: June 17, 2005 changed by: Gerrit Muller
- Created, no changelog yet