Didactic Recommendations for Education in Systems Engineering

by Gerrit Muller       University of South-Eastern Norway-NISE
                        e-mail: gaudisite@gmail.com
                        www.gaudisite.nl

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.
Experience in SE education
"effective transfer of know-how requires an active attitude from the audience"
Experiences of Teaching Systems Architecting, Gerrit Muller at INCOSE 2004
Example Postgraduate Programs Systems Engineering

Stevens Institute Systems Engineering and Engineering Management
http://www.soe.stevens.edu/seem/

MIT System Design and Management
http://lfmsdm.mit.edu/sdm/index.html

University of South Australia
http://www.unisa.edu.au/seec/
<table>
<thead>
<tr>
<th>BS program at:</th>
<th>credit hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>University of Arizona</td>
<td>128</td>
</tr>
<tr>
<td>University of Arkansas at Little Rock</td>
<td>130</td>
</tr>
<tr>
<td>University of Pennsylvania</td>
<td>120</td>
</tr>
<tr>
<td>University of Virginia</td>
<td>128</td>
</tr>
<tr>
<td>U.S. Naval Academy</td>
<td>143</td>
</tr>
<tr>
<td>Washington University</td>
<td>120</td>
</tr>
</tbody>
</table>

- Credit hours for BS programs varies between 120 – 143
- All BS programs build on basic engineering and science courses.
- Programs differ in their emphasis areas from university to university although the systems engineering fundamental courses remain the same.
- Some universities offer considerable amount of flexibility in their BS programs by creating emphasis areas.

source: Professor Cihan H Dagli, PhD at INCOSE 2004, Toulouse
Undergraduate Education in Systems Engineering in USA
Didactic Recommendations for Education in Systems Engineering

version: 0
September 6, 2020
SXcurriculumSA

Gerrit Muller
Active vs Passive

- **Theory**
  - passive: dull
  - active: vivid

- **Practical Illustration**
  - passive: vivid
  - active: dull

- **Interaction**
  - passive: vivid
  - active: dull
  - **Spin-off:** cross-fertilization

- **Insight**
  - passive: dull
  - active: vivid
Finding the Balance Active-Passive

Didactic Recommendations for Education in Systems Engineering

Gerrit Muller
Interaction

+ Pose questions to the students
+ Keep the communication open in all directions
+ Keep the students alert
+ Maintain a consistent mindset
Example questions

Provocative:
"What is the most important process in your company?"
differentiate between important or core processes and less important supporting processes.

Explorative:
"What are the deliverables of an architect?"
followed by f.i. "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?"
Keep the Communication Open

+ Allow or even stimulate discussion
+ Managing two-way communication, the parking flip
+ Creating an open and safe learning environment, rules:
  · Argue in a constructive way, no heat seeking missiles allowed!
  · Stupid questions don't exist
Keep the students alert

platform approaches reduce lead-time, cost, ...

platform developments in practice increase lead-time, cost, ...

"What do patient or insurance company need or expect?"

sudden changes of viewpoint

student

teacher

counter-intuitive examples
Maintain a consistent mindset

- 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
Example maintain mindset by keeping alert

"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?"

We cannot do this, because the amount of software is way too large

student

teacher
+ presenting
+ teamwork
+ self-reflection
+ providing balanced feedback
course material
+ slides
+ reader

low-tech support
+ flips
+ yellow notes
Exercise instruction:
short, asking for illustration and specifics

Team size:
4 is optimal; 3 or 5 members is acceptable

Duration
40 minutes