

Article

# Thinking Tracks for Multidisciplinary System Design

Gerrit Maarten Bonnema <sup>1,2,\*</sup> and Jan F. Broenink <sup>3,†</sup>

<sup>1</sup> Department of Design, Production and Management, Faculty of Engineering Technology, University of Twente, 7500 AE Enschede, The Netherlands

<sup>2</sup> Norwegian Institute for Systems Engineering (NISE), University College of Southeast Norway, 3603 Kongsberg, Norway

<sup>3</sup> Robotics and Mechatronics, Faculty of Electrical Engineering, Mathematics, and Computer Science, University of Twente, 7500 AE Enschede, The Netherlands; j.f.broenink.utwente.nl

\* Correspondence: g.m.bonnema@utwente.nl; Tel.: +31-53-489-2548

† These authors contributed equally to this work.

Academic Editor: Ockie Bosch

Received: 8 September 2016; Accepted: 2 November 2016; Published: 10 November 2016

**Abstract:** Systems engineering is, for a large part, a process description of how to bring new systems to existence. It is valuable as it directs the development effort. Tools exist that can be used in this process. System analysis investigates existing and/or desired situations. However, how to create a system that instantiates the desired situation depends significantly on human creativity and insight; the required human trait here is commonly called systems thinking. In literature, this trait is regularly used, but information on how to do systems thinking is scarce. Therefore, we have introduced earlier twelve thinking tracks that are concrete and help system designers to make an optimal fit between the system under design, the identified issue, the user, the environment and the rest of the world. The paper provides the scientific rationale for the thinking tracks based on literature. Secondly, the paper presents three cases of application, leading to the conclusion that the tracks are usable and effective.

**Keywords:** systems design; architecting; systems thinking; application

## 1. Introduction

“Systems Engineering is an interdisciplinary approach and means to enable the *realization of successful systems*” (italics by authors) [1] (p. 11). However, the emphasis in many articles and books is on the process for systems engineering, or on analysis of an existing system, or the problem at hand. How-tos for creation or realization—in short: design—of systems are scarce, due to the open nature of design problems. For creation, there is a need for a spark of invention. For dealing with such open problems in a systems context, we introduced [2,3] twelve thinking tracks earlier that act as a checklist; viewpoint changer; and creativity starters. The references mentioned contain a short [2], and a detailed [3] description of the thinking tracks.

An important aspect is that systems are set in an environment, and that there is mutual influence between the system and the environment. However, at design time, the system under design is still largely unknown, and, thus so, is its influence on the environment. We therefore have to build upon uncertain and incomplete information. The impact of uncertainty in decision making is well described in [4,5]. Madni and Jackson [6] describe this problem in the field of resilience engineering. The use of models, experience, estimations and imagination play a large role. However, system designers need aids to deal with this uncertainty and support in directing their imagination. We argue that our thinking tracks help in this respect.

First, in this paper, we add to the track descriptions in [2,3] the foundation of the tracks in systems thinking (ST) and systems engineering (SE) literature. Compared to concepts in Systems

Thinking literature, the proposed tracks make the systems thinking approaches concrete to help systems designers in their daily work.

Secondly, we illustrate the tracks’ workings via an example, and evaluate the tracks’ usability and effects in three cases. This way, we can draw conclusions on the usability and helpfulness of the tracks for the practitioner.

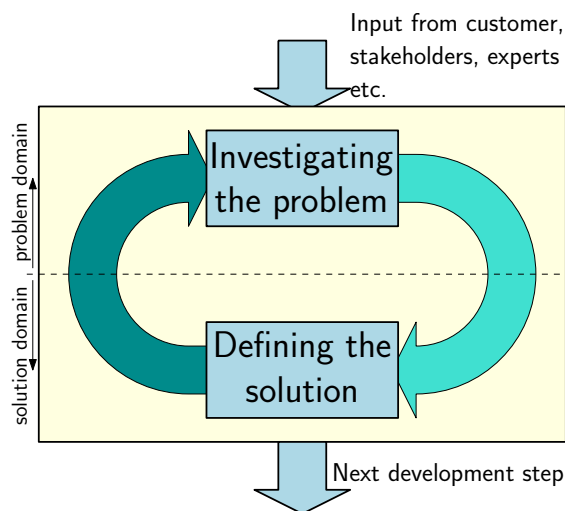
We focus on technical systems, most notably mechatronic systems, as that is our context and source of experience. Therefore, the reasoning in this paper will fit to this class of systems, allowing us to be specific. We do expect our reasoning to be valid when applied to the design of non-mechatronic systems, possibly after some tuning to those systems.

As a foundation, we start with exploring system design in Section 2, including the state-of-the-art in literature. An important trait of system designers is found to be the ability to think in and about systems: systems thinking. We look into that in Section 3. The thinking tracks are summarized in Section 4. In Section 5, we correlate the tracks presented in Section 4 to the literature in Sections 2 and 3, the first objective. The second goal is elaborated on in Section 6. Use of the tracks is illustrated in an example in Section 6.1, and in Section 6.2, we discuss three cases of application of the thinking tracks. The paper ends with conclusions and recommendations for future work in Section 7.

## 2. System Design

Designers create. Creating means bringing to existence, in other words: afterwards, there is something that was not there beforehand. This is, by definition, different from analysis (as in: an explanation of the nature and meaning of something [7]. Analysis interacts with something that is there but requires investigation. There is an intimate relationship between analysis and design, as a problem needs analysis before a resolution can be created [8]. In addition, the intended or created resolution has to be checked for its effectiveness; again, an analysis activity.

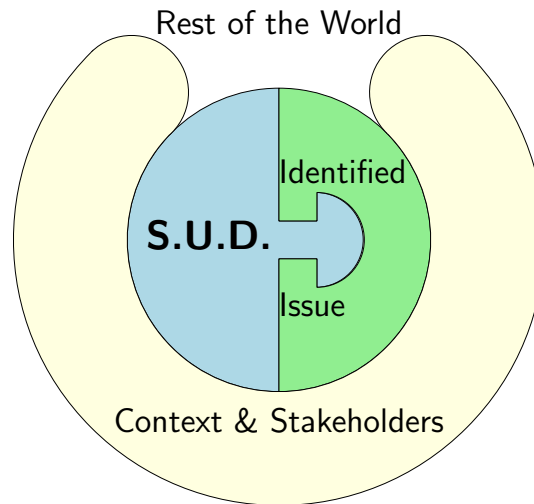
Figure 1 shows two domains: the problem domain at the top and the solution domain at the bottom. The result of investigating the problem should be an answer to “What should the system do?” and “What should the performance be?” In the solution domain, an answer is sought to the “How?” question, with the questions from the problem domain as reference.



**Figure 1.** The design process as an iterative loop of alternating analysis (**top**, in the problem domain) and creation (**bottom**, in the solution domain). Picture courtesy of CRC press [3].

The result of the design process should be a system (the System Under Design, S.U.D.) that fits the identified issue, the context and stakeholders, and the rest of the world, as shown in Figure 2. Important intermediate results are the system requirements that direct the design effort. However, requirements mostly relate to the identified issue and context and stakeholders (Figure 2).

These may change during the design process, and the rest of the world may have relevant, yet unforeseen, interactions with the system, for which no requirements have been formulated.



**Figure 2.** The System Under Design (S.U.D.) has to fit the identified issue, the context, the stakeholders and the rest of the world. Picture courtesy of CRC press [3].

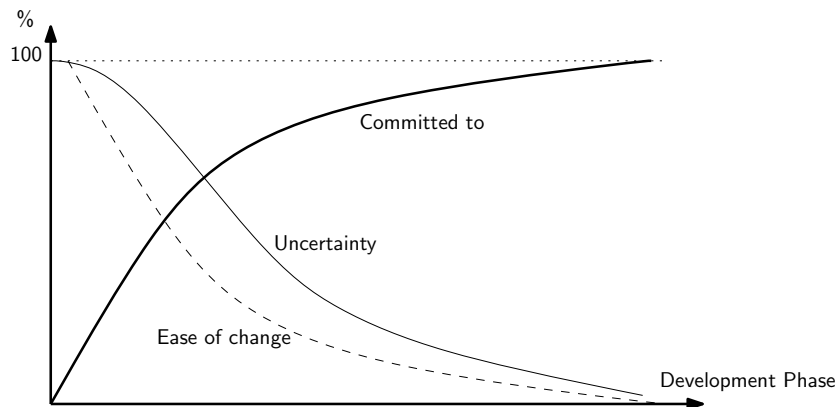
Note that systems engineers create systems where they are unlikely to actually use the result by themselves (not many system engineers of fighter jets are actual fighter pilots, nor are system engineers of wafer steppers actual chip manufacturers). In addition, there are many disciplines involved in the further development process. The people involved are spread over countries, sometimes even globally. This can lead to a large gap between the problem domain and the solution domain.

To verify the appropriateness of the solution to the problem (the upward arrow in Figure 1), the analytical principles of, for instance, axiomatic design [9] are useful. However, for the solution definition process (the downward arrow), uncertainty, creativity and search are discerning characteristics [10]:

“A designer makes things. Often, the thing initially is a representation, a plan, a program, or an image to be constructed by other people. Many of the relevant variables cannot be represented in a model; this limitation makes the design process inherently complex. A system is complex in the specific sense that, whenever I make a move, I get results that are not just the ones that I intend. That is, I cannot make a move that has only the consequences that I intend. Any move has side effects.

This unpredictability is a central attribute of design—it is not necessarily the defining one, but it is important”.

This attribute of design is also shown in Figure 3. At the start of the design process, the outcome is unknown, while the freedom to make decisions is infinite. In that situation, the architecture of the system has to be defined, which shapes the overall system. This is a typical situation that describes the core of the system designer’s job: making important, often far-reaching decisions, based on little and uncertain information.



**Figure 3.** In the early phase of development, many decisions have to be made, and technologies have to be committed to. However, a significant uncertainty (or: lack of knowledge) exists at that point in time. The catch is that also the ease of change decreases rapidly in the progressing of the project. (based on, and adapted from, [11].)

The documented system design methods and approaches [1,11–16] tend to bridge the gap by a (combination of) focus on the systems engineering process; and/or focus on (a set of) systems engineering tools. Variations exist for the process, like the Vee-model, spiral model etc. Tools that exist include  $N^2$  diagrams, functional block diagrams, requirement engineering tools, etc. However, while the process emphasizes the managerial side of systems development, the tools relate to problem and system analysis. There is a lack of the creation aspect. In addition to the two pillars process and tools, a third pillar—systems thinking—is required to provide a stable system design approach.

Before looking at ways that help, we will analyze the Systems Thinking state-of-the-art and characteristics of good systems engineers.

### 3. Systems Thinking

From the above, we conclude that following the SE process and using tools will increase knowledge about the current situation and the problem, yet it will not produce a new system design. This requires creational power, and thus human input. As systems are composed of many parts, and span several disciplines, it is important to see the whole. Senge [17,18] calls Systems Thinking the discipline for seeing the whole.

Cabrera [19] has produced an extensive study on systems thinking using literature from diverse fields, and interviews, linking knowledge about systems and conceptual systems thinking. This study is too extensive to be treated here in detail, as it starts from the early mentions of systems thinking up and to the turn of the century, referring to works by Ludwig von Bertalanffy, Peter Checkland, Midgley, Capra, Senge and many others.

Cabrera identifies two fields in the systems literature: the *knowledge about systems* and conceptual and epistemological *systems thinking*. Noteworthy is that he identifies multiple times systems thinking as conceptual or as “habits of mind”. Cabrera identifies a lack of training in systems thinking, and a difficulty in training due to absence of a clear theory of systems thinking. His thesis is a step in the direction of such a theory. Aspects that have to be included are different levels: one man’s systems thinking is another man’s detail thinking ([19], p. 61); and the need for boundaries or enclosures to make systems thinking practical ([19], p. 63).

“Systems thinking is defined as thinking that is informed by knowledge-about-systems (of all kinds). In other words, systems thinking is not one kind of thinking, but rather is thinking that utilizes an understanding of many types of systems.”([19], p. 51)

Cabrera also discusses the relation between systems thinking and the use of system dynamics as introduced by Forrester in the 1950s [20]. While system dynamics, including the use of causal loop

modeling and modeling of stocks and flows, play an important role in systems thinking, it provides just one view on systems.

Eventually, he arrives at a Minimal Concept Theory of Systems Thinking (MCT/ST): [19] (p. 176):

“Systems thinking is a conceptual framework, derived from patterns in systems science concepts, theories and methods, in which a concept about a phenomenon evolves by recursively applying rules to each construct and thus changes or eliminates existing constructs or creates new ones until an internally consistent conclusion is reached. The rules are:

- Distinction making** differentiating between a concept’s identity (what it is) and the other (what it is not), between what is internal and what is external to the boundaries of the concept or system of concepts
- Interrelating** inter-linking one concept to another by identifying reciprocal causes and effects;
- Organizing Systems** lumping or splitting concepts into larger wholes or smaller parts; and,
- Perspective taking** reorienting a system of concepts by determining the focal point from which observation occurs by attributing to a point in the system a view of the other objects in the system (e.g., a point of view).”

The four rules that are part of the definition are of particular importance in the present paper.

Frank has conducted a series of research into systems thinking and what distinguishes someone with a capacity for systems thinking from another person [18,21]. He states a system thinker is able to:

1. See the big picture;
2. Understand connections and closed loops;
3. Understand synergy;
4. Understand the system from multiple perspectives;
5. Think creatively;
6. Step over details and handle uncertainty and ambiguity;
7. See implications of changes;
8. Understand a system when he/she sees it;
9. Understand analogies;
10. Understand limits to growth.

Davidz and Nightingale [22] conclude from 205 interviews that componential; relational; contextual; dynamic; and modal elements are part of systems thinking. Their resulting definition is: “Systems thinking is utilizing modal elements to consider the componential, relational, contextual, and dynamic elements of the system of interest.”

Richmond [23], from an educational point of view and based on system dynamics, defined seven thinking skills as critical for the 1990s and beyond: dynamic, closed loop, generic, structural, operational, continuum and scientific thinking. In fact six of the thinking tracks that we present are based on these skills by Richmond as they provide a means to connect the system to the context (see Figure 2).

Hitchins’ five-layers [24,25] can serve as a reference for a system in its context:

- Level 1 Product Level. Many products (can) make a system. Many engineers and their institutions consider this to be the only “real” systems engineering.
- Level 2 Project or System Level. Many projects make a Business. Western engineer-managers operate at this level, principally making complex artifacts.
- Level 3 Business Systems Engineering—many businesses make an industry. At this level, systems engineering seeks to optimize performance somewhat independent of other businesses.

Level 4 Industrial Systems Engineering, or engineering of complete supply chains/circles. Many industries make a socio-economic system. Japan seems to operate most effectively at this level.

Layer 5 Socio-Economic, legal and political influences, government regulation and control.

Finally, Checkland [26] is an important author in the field of systems thinking. His book proposes a soft systems methodology. On p. 102, he defines systems thinking: “An observer giving account of the world or a part of it, ...”. The approach by Checkland is elaborated upon and is put into perspective by Boardman and Sauser [27].

While the above describes what is required to be successful in systems thinking (descriptive), it does not help in how to do good systems design: we have to make the step from investigation (as clearly described by Checkland) towards creation. On the one hand, this implies limiting and directing the thinking effort (Checkland does treat the difference between creation-oriented engineering and analysis-oriented scientists). On the other hand, the descriptive treatment of systems thinking as summarized in this section has to be expanded to make it applicable to the systems engineering field. The thinking tracks that we present in the next section are an attempt of doing that.

#### 4. Systems Thinking Tracks

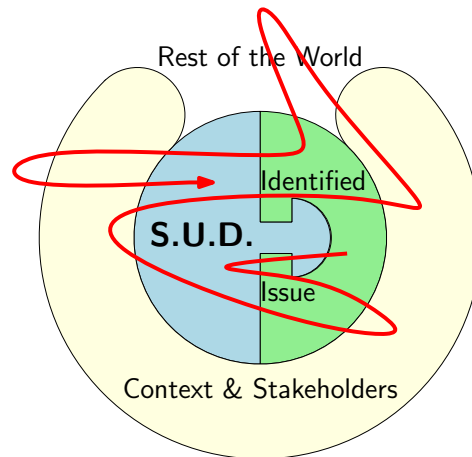
We already saw that design is not finding the answer to a closed problem, where there is only one solution. Moreover, the problem can be described in several ways [26]. Therefore, we have to look for inspiration on the one hand and constantly evaluate possible solutions on the other hand (see Figure 1). In literature, several approaches are presented. Some are more in the line of creativity [28,29], others use lines of reasoning or (systems) thinking tracks [14,23,30]. Finally, there are descriptions of how to look at systems [25,26,31,32]. Combining these with experience from the authors, we summarize in this section twelve systems thinking tracks that can be used as:

- checklist;
- viewpoint changer to understand different aspects of the problem;
- means to avoid mental inertia;
- creativity starter; and
- means to transfer experience to novice system designers.

It is important to realize that the system design methods mentioned in Section 2 are quite precise and result in a description of the system to be designed that will fit the identified issue, the context and the stakeholders. However, how it will fit the unknown rest of the world is not treated. The thinking tracks deliberately pull the thinking of the system designer out of the identified issue—System Under Design (S.U.D.)—context-stakeholders path, as shown by the chaotic red path in Figure 4. The system designer has to sample the whole design field, using the thinking tracks and aided by tools. Only the essential and risky parts will then be given more attention in the early part of the process.

The thinking tracks help in uncovering information and knowledge, or sometimes only making the knowledge explicit and (more) readily available. Compare this to finding a victim of an avalanche. The snow cover has to be probed such that the victim is found quickly. In system design, the number of “victims” (important or crucial issues) is, however, unknown.

In the following, we summarize the twelve thinking tracks. Elaborate descriptions, including examples and applicable tools, are presented in our earlier publications [2,3]. The short descriptions given in the following sub-sections merely introduce the tracks so that the rest of the argument can be followed.



**Figure 4.** The S.U.D., the identified issue, the context, the stakeholders and the rest of the world and one of many thinking routes (in red) for the system designer. This path clearly shows the somewhat chaotic reasoning applied by a system designer. Picture courtesy of CRC press [3].

#### 4.1. Dynamic Thinking

The sort of systems designed by system designers and system engineers are not monolithic invariant systems. They are systems that interact with the environment and are often highly dynamic (Definition of dynamic: always active or changing; [7]). Looking at the system from a dynamic perspective is thus essential:

- How does the system change over time?
- How does the environment change over time?
- When a change in input/output occurs, what are the effects (or: should be)?
- Does it (have to) behave linearly, or non-linearly or even chaotic?

#### 4.2. Feedback Thinking

Closely related to the fact that systems have time-related behavior is the question whether there is feedback: comparing the actual output of a system with the desired output, and adjusting the available controls so that the requested output is achieved. While this represents an engineering view on systems, it is useful in a wide range of applications, from engineering systems to politics.

#### 4.3. Specific-Generic Thinking

This track is about determining the scale of magnitude of the problem, and the estimated investment for the proposed solution. The moment numbers are introduced in the discussion—even if they are off by a factor of two—will already filter out many “shooting a mosquito with a canon” situations.

#### 4.4. Operational Thinking

In operational thinking, it is investigated how something is done in the real world. As this thinking track relates to how the actual system works, it is not possible to make a generic list that works for all systems. However, the use of stories and scenarios is advised. Involve experts and operators of such systems in this process. In particular, pay attention to start-up and shut-down sequences, and abnormal situations.

#### 4.5. Scales Thinking

In scales thinking, we reason about the scales used to describe the problem and the solution. In particular, the resolution and validity of these scales are looked at. Computer simulations are

regularly based on abstracted models, thus their predictive value is restricted to those situations where the assumptions of the model hold. The purpose of the model, i.e., what design questions we want to get answered/supported using this model, dictates the level of detail of the model.

Another way of using scales thinking is the difference between yes/no or black/white and shades of grey.

#### 4.6. Scientific Thinking

It is important to always use scientific principles when designing a system:

1. formulate a question;
2. formulate a hypothesis;
3. create a model that can be used to predict behaviour;
4. verify the hypothesis and thus the model, by means of simulations, experiments, consulting literatur, or consulting experts;
5. and analyse the results and adjust the model if necessary.

#### 4.7. Decomposition–Composition Thinking

In many cases, systems engineering is presented as a way to determine how the system can be decomposed in subsystems, assemblies, parts, components, etc. to make the development process manageable. In this track, we reason about how all of these can be integrated—composed—into a well-functioning system. In addition, how can, in the whole process, the integrity be safeguarded?

#### 4.8. Hierarchical Thinking

The systems engineering process is often represented in a hierarchical organization (hierarchy: a graded or ranked series ([7]) from system to subsystem to assembly etcetera. This structure is also how many systems are organized. However, this organization does not necessarily dictate the way the system’s control functions. In hierarchical thinking, the system designer has to reason about ranking authority, facilities and priorities of the system’s parts.

#### 4.9. Organization Thinking

Most of what is said about the system under design is valid for the organization that creates the SUD; therefore, most thinking tracks can—and should—be applied to the organizational structure as well. There is an intricate relation between this structure and the architecture of the system [33]. An extreme form is tagged “Conway’s Law” [34].

#### 4.10. Life-Cycle Thinking

Life-cycle thinking can be understood in three distinctive, but equally important, ways that will be treated in separate subsections.

##### 4.10.1. Product Life-Cycle Thinking

Any product and system goes through different phases in its life: Need Identification, Design, Production, Distribution, and Use and Retirement. Consideration of all product life-cycle phases results in trade-offs. For instance, between making a system more durable (and thus more expensive), or implementing more frequent maintenance, resulting in a lower purchasing cost.

##### 4.10.2. Resource Life-Cycle Thinking

The systems engineer should be aware of the basic principles of Life-Cycle Analysis (LCA), as, for instance, outlined in [35], and Cradle to Cradle [36], and be able to think about the resource use. Trade-offs will occur between using more resources in the development and production phase versus using more resources in the actual use or retirement phase.



#### 4.10.3. Organization Life-Cycle Thinking

In order to create the SUD, an organization has to be designed, created, put into use, used, maintained and retired. This shows that the organization also has a life-cycle. This track focuses on the long life-cycle of the organization.

#### 4.11. Safety Thinking

Product safety is paramount. The systems engineer has to reason about how the system can be used, and whether that is safe; and in what ways the system can be used unsafely. Note that, depending on the type of user (persons with specific training and responsibilities, or general public), the safety measures, including handling abnormal behavior, can take a large amount of the development effort.

#### 4.12. Risk Thinking

Designing innovative systems comes with risks. Avoiding risks will automatically lead to non-discerning results, and thus to less profit or even loss. It is, therefore, a matter of managing the risks in the project such that the impact can be controlled and contained. Risk thinking considers different types of risks (technical, cost, planning and external [1]), reasons about the likelihood and impact, and investigates possible mitigation scenarios.

### 5. Thinking Tracks Verified

The thinking tracks summarized above condensate years of experience and results from theory and scientific literature in different engineering disciplines.

To check whether this set of thinking tracks is adequate and not too extensive in light of the overview of systems thinking given in Section 3, we correlate the thinking tracks to the aspects of systems thinking as listed by Cabrera, Frank, Davidz and Nightingale, Richmond, and Hitchins.

While the result, Table 1, might look exact, some caution is required. In the first place, the descriptions of the authors mentioned in Section 3 are useful, but often prone to different interpretations. This is inherently part of the soft side of systems thinking. Secondly, the thinking tracks described in Section 4 have large areas of application and serve as creativity starters. Therefore, a wide interpretation of the tracks is likely (and even encouraged). This results in not as clear-cut borders of the markings in Table 1 as one may interpret.

Looking at Table 1, and keeping the caution above in mind, we see that (1) nearly all thinking tracks (the columns in the table) correlate to at least one of the aspects of every systems thinking author (the exceptions will be treated below); and (2) all of the systems thinking aspects (the rows in Table 1) map to at least one, often more, thinking tracks. Further observation reveals that none of the thinking tracks have equal patterns of correlation.

There are four exceptions:

1. Scientific thinking is not connected to any of Cabrera's MCT/ST rules. However, Cabrera's rules are based on scientific reasoning, and thus the scientific approach is represented in all of the rules.
2. Thinking creatively (from Frank's competences) does not *directly* correlate to a specific thinking track. However, as stated earlier, the tracks are intended to promote creativity in the systems thinker. We have, therefore, given the entire row a (lighter) shade.
3. Safety thinking does not map to any of Frank's competences. Safety thinking may be too specific for Frank's classification. It is, however, our opinion that every system engineer should practice this form of systems thinking in every project.
4. Richmond's thinking skills only relate to a limited set of thinking tracks. As stated earlier, Richmond's skills are the basis for thinking tracks 1 to 6.

Finally, the thinking tracks can be reviewed against Hitchins' five layers. Here, it is more a question of which thinking track is applicable to which layer in Hitchins' five-layer model. It is relatively easy

to understand that at the first two levels—Product and Project or System Level—the thinking tracks are all applicable. On the higher levels, Safety and Decomposition–composition thinking may be harder to apply, but all the others are relevant. In feedback thinking, a relevant example at level 5 is the introduction of tax exemptions for *green* cars in the Netherlands. This was a great success such that other tax increases, unanticipated limitation of the rule or other measures were necessary in order to ensure sufficient government income. Some form of feedback thinking while creating the rule would have prevented or limited the nationwide discussions.

**Table 1.** Correlating the thinking tracks to (a) the rules that constitute Cabrera’s definition of systems thinking [19], (b) Frank’s competences [21], (c) Davidz and Nightingale’s elements of systems thinking [22], and (d) Richmond’s thinking skills [23].

	Thinking Track											
	D	F	SG	Op	Sc	Sf	DC	H	O	LC	Saf	R
<b>(a) Cabrera’s MCT/ST rules</b>												
Distinction making												
Interrelating												
Organizing Systems												
Perspective taking												
<b>(b) Frank’s competences</b>												
See the big picture												
Understand connections & closed loops												
Understand synergy												
Understand system from multiple perspectives												
Think creatively												
Step over details; handle uncertainty & ambiguity												
See implications of changes												
Understand a system when seeing it												
Understand analogies												
Understand the limits of growth												
<b>(c) Davidz &amp; Nightingale’s elements</b>												
Componential												
Relational												
Contextual												
Dynamic												
Modal												
<b>(d) Richmond’s thinking skills</b>												
Dynamic												
Closed-loop												
Generic												
Structural												
Operational												
Continuum												
Scientific												

Legend: D: Dynamic, F: Feedback, SG: Specific-Generic, Op: Operational, Sc: Scales, Sf: Scientific, DC: Decomposition-Composition, H: Hierarchical, O: Organizational, LC: Life-cycle, Saf: Safety, R: Risk. MCT/ST: Minimal Concept Theory of Systems Thinking.

What we derive from this reasoning is that the thinking tracks span the systems thinking universe as identified by Cabrera, Frank, Davidz and Nightingale, Richmond, and Hitchins. At the same time, there is no thinking track that can be omitted, as the patterns in the correlation table differ.

Summarizing, the thinking tracks cover the required system thinking capabilities identified by Frank, Davidz and Nightingale, and Richmond [21–23], and they cover the five levels of systems identified by Hitchins. In addition, they conform to the definition of systems thinking that is based on an extensive literature review given by Cabrera [19].

Their applicability in representative cases is treated next.

## 6. Examples and Application

In this Section, we investigate the useability and usefulness of the thinking tracks. First (Section 6.1), we show how they can be used in an example application developed by the authors.

Secondly (Section 6.2), we introduced the tracks to others (students and PhD researchers), to see how they can work with them, and whether they find the tracks of use.

### 6.1. Example: A Solar Racer

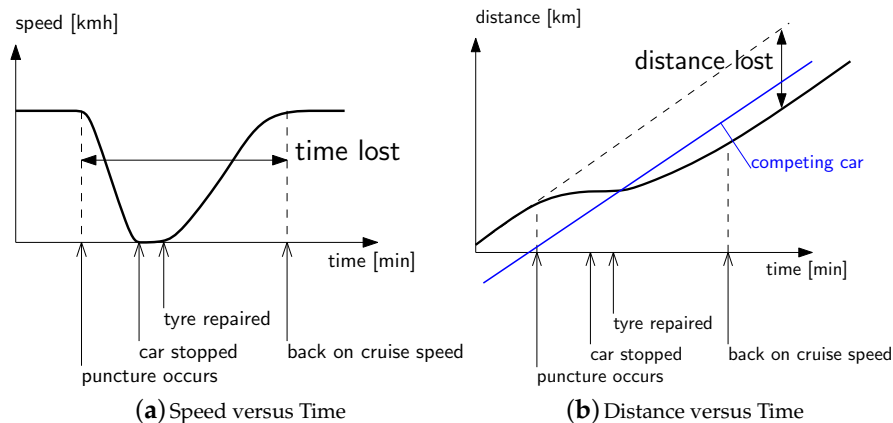
In [3], we use the design of the University of Twente solar race car that can compete in the World Solar Challenge [37] across Australia, as a running example of the use of several tracks. University of Twente Student teams of ca. 18 people participated in 2005, 2007, 2009, 2011, 2013 and 2015. In this paper, we summarize the example.

#### 6.1.1. Dynamic Thinking

For the solar racer from the Twente Solar Team interesting time scales are:

- seconds: How does the car react to vibrations?
- minutes: How does the weather change? How to react to a puncture?
- hours: driver behavior;
- days: overall strategy and race planning;
- weeks: project planning and manufacturing;
- months: financial planning, and overall project planning.

Modeling, for instance, the impact of a puncture on the total racing time, can be done for velocity as a function of time (see Figure 5a). From this, conclusions can be made about the time lost. An alternate representation of data present in the model shows distance as a function of time (Figure 5b). Now, the trajectory of a competing car that has no puncture can be easily incorporated (see the blue line). From these two diagrams, design consequences can be derived.



**Figure 5.** Two representations for the dynamic, time-related, behavior of a solar car when confronted with a puncture. In (a), the time lost can be found; in (b), the position relative to a competing car can be seen. Picture courtesy of CRC press [3].

#### 6.1.2. Feedback Thinking

In the solar racer, speed is controlled using a cruise control system. However, the solar racer is subject to more signals and disturbances than speed alone; both long-term and short-term: wind, solar energy input, road inclination, energy needed for overtaking vehicles, etc. If these can be measured or even predicted, and the controller can be taught to cope with that, the overall behavior can be optimised to parameters like speed and energy use.

### 6.1.3. Specific-Generic Thinking

Although the solar racer has to drive on public roads, and thus has to show its road worthiness, it does not have to comply with the Road Traffic Act completely, but only to the basic rules. Instead, two vehicles that do fully comply, drive in front and directly behind the solar racer.

### 6.1.4. Scientific Thinking

In the development of the solar racer, the total drag coefficient is determined by roll resistance and air drag, plus internal resistance. These are unknown at the beginning of the project as the number of wheels, their size, the shape of the body, etc. are open. Information is created during the project, for instance, because data from preceding years can be used and simulations are run. As soon as a mock-up is ready, data on roll resistance can be measured. Air drag can be measured with a model in a wind tunnel. As soon as the real body is ready, real drag resistance data can be acquired at different speeds and different angles to the wind to be used in tuning the cruise control of Section 6.1.2. This does require a well-thought out measurement protocol (requiring operational thinking).

### 6.1.5. Product Life-Cycle Thinking

In the design of the 2011 Twente solar racer, it was decided to use a monocoque construction, as this provides a light and stiff construction (advantages in the use phase). However, in the design and construction phases, consequences are that the monocoque frame cannot be designed before the aerodynamic shell design is finished. However, the monocoque frame is the first part needed in the construction phase. A solution was found in making a simple welded steel frame, that would act as a test set-up for most other parts.

## 6.2. Application

The above shows an example to illustrate the workings of the tracks. To verify their applicability and usefulness, we have executed three application cases with students:

1. Use, at will, by two generations of industrial design engineering students;
2. Application on the Denver airport vignette from the systems engineering BKCase [38] by three students; and
3. Application by PhD students in the International Spring School on Systems Engineering (IS3E) [39].

Please note that in reporting such case-based research, it is impossible to report the entire case and all details involved. However, by documenting the steps taken in a transparent manner, we believe we have circumvented this significantly. In addition, note that it is inherently difficult to do comparative studies in (system) design research (see [40,41]).

### 6.2.1. Case 1: SE Course

Two generations of third-year undergraduate Industrial Design Engineering students have been introduced in the thinking tracks. A course reader (preliminary versions of [3]) was handed to the students, containing the more extensive track descriptions. In the lectures, regular reference was made to the thinking tracks. Parallel to the course, the students worked on a design project (the first year an automated supermarket delivery system, the second year a smart hospital bed), in groups of 12–14 persons. The application of the thinking tracks was voluntarily, and was not measured explicitly.

As the students had to write an essay about the application of systems engineering, it was possible to find their experiences. The two-page essays have been merged into one pdf file, and by searching on the terms “track” and “thinking” possible mentioning of the thinking tracks was sought. Upon hitting one of the terms, the context was read to see whether, and, if so, in what manner a thinking track was treated. A small number of students delivered a word-version of the essay. These were treated separately but analogously. The results are then combined: from the first year, nine (out of 104)

students explicitly reported the use of one or more thinking tracks, and 10 (out of 97) in the second year, as summarized in Table 2, columns EssY1 and EssY2.

**Table 2.** Number of mentioned thinking tracks in the Systems Engineering Essays (EssY1 and EssY2), the Denver Airport Vignette (Vignette) and the Spring School (Spring).

Thinking Track	EssY1	EssY2	Vignette	Spring
Dynamic	3	1	1	5
Feedback	2	2	1	
Specific-generic	1	2	1	
Operational	4	1	3	3
Scales	1			3
Scientific		1	1	
Decomposition-composition	1	2	3	1
Hierarchical			1	
Organizational		1		
Life-cycle		1	1	1
Safety	1		2	
Risk	1	2	1	2
Unspecified	9	8		

The tracks are mentioned in two ways. One was that the particular track was helpful (often clarified with an example). The other was the observation that, in the project team systems thinking, or a particular type of thinking, was lacking or insufficient. The students that applied one (or more) track(s), discussed in their essays how a track contributed to the improvement of the system design, mostly in combination with one or more system design tools (functional block diagrams, system budgets, etc.). One student explicitly mentioned the use as creativity starter.

#### 6.2.2. Case 2: Denver Airport Vignette

Three students (again, Industrial Design Engineering students in their third year) received an extra assignment in Systems Engineering. The assignment was to review the twelve thinking tracks and assume the role of a chief systems engineer in the Denver Baggage Handler case (described in the vignette included in the Systems Engineering Body of Knowledge (SeBOK, part of [38])). Then, they had to apply five or more thinking tracks, support the thinking with SE tools and evaluate the effectiveness.

As using at least five thinking tracks was required, the three students mentioned fifteen tracks in total, as shown in Table 2, Vignette. Here, the information on usefulness was clearly available in the required evaluation of the thinking tracks. One of the students even gave each of the tracks he used (Dynamic, Feedback, Operational, Decomposition-Composition, and Life-Cycle Thinking) a (highly subjective) “success rate”. Here, the highest score (9 out of 10) was given to dynamic thinking. Life-Cycle Thinking got 5 out of 10 for this specific case.

While the information in the case description is limited, it shows from the reports that the thinking tracks helped the students to investigate the case, and structure the assignment. Moreover, the selection of system design tools was guided by the thinking tracks. For instance, safety thinking led to the creation of scenarios with non-normal behavior. Decomposition-composition thinking provided overview and initial breakdown. One student applied specific-generic thinking to investigate the problem. While this, at first, did not reveal a lot of information, the FunKey architecting method [42] helped this student in showing the main key drivers and most important functions. One student used dynamic thinking to investigate the difference between large baggage carts, or multiple smaller, faster ones. Another student evaluated dynamic thinking as “This is a great thinking track. It forces you

to look at the system on a different way as you would do normally. At this way, new problems or functions are discovered ...”.

Although the students were inexperienced as systems engineers, they managed to identify several key issues using the thinking tracks. While the resulting system designs should be discussed with actual stakeholders in the real design case instead of this exercise, the usefulness of the thinking tracks combined with system design tools can be seen. It should be noted that verification of the designs produced was outside the scope of the assignments.

### 6.2.3. Case 3: Spring School

Eighteen PhD-level students in five groups were introduced to the thinking tracks in a workshop at the First International Spring School on Systems Engineering, organized jointly by Technische University München (Institute of Product Development) and Fraunhofer Institute for Production Technology IPT—Project Group Mechatronic Systems Design (supported by University of Paderborn (Heinz Nixdorf Institute), Stevens Institute of Technology, École Centrale Paris, University of Twente and the International Council of Systems Engineering (INCOSE) (German Chapter)). After presenting the tracks, a productivity case of ASML wafer steppers was introduced. This real-life case concerned the impending semiconductor industry shift from 300 to 450 mm diameter silicon wafers. The industry aims at 30% cost reduction. The question was to identify the main challenges for ASML wafer scanners in making this shift. There was limited time available for making this assignment (4–6 h), and the students were not experienced in the field of semiconductor manufacturing.

The tracks used, as reported in the presentations (both the slides used and the oral presentation), have been tallied. Table 2, Spring, summarizes the tracks used. The high occurrence of dynamic thinking, followed by operational and scales thinking, can be explained by the type of problem. What is more interesting but harder to present in this paper is the fact that the teams, despite their novice state in the field and the limited time, did manage to identify some of the key issues that relate to stage performance (relating to increasing mass and dynamics), and lack of available personnel due to the other continuous developments in the semiconductor industry: device shrink. One group created key figures for machine cost comparing a 450 mm machine to a 300 mm one using scales thinking. Others used scales thinking to find opportunities for the aimed cost reduction.

### 6.3. Discussion of Case Studies

The example illustrates the way of working with the thinking tracks. The three application cases show that the thinking tracks are easy to learn and apply. While content matter knowledge is needed for high quality designs, the thinking tracks provide direction and inspiration for exploring the design space. As seen in the spring school application, even in a short time, key issues can be uncovered by application of the tracks.

From the SE course case, we conclude that the variation of thinking tracks is important. While in the first year operational thinking and dynamic thinking were mentioned most often, the other tracks and all tracks mentioned in the second year are evenly distributed. Most of the twelve tracks introduced have been used.

In the Denver airport case, it was shown that the tracks are useful in exploring both the width and depth of a design case. In particular, they helped in connecting to system design tools, and exploring abnormal behavior.

In the Spring School and in the Denver case, the issue of sequence of application was addressed. In fact, the thinking tracks represent integrated thinking. A very experienced systems designer performs this kind of thinking automatically and largely simultaneously as “habits of mind” [19]. For less experienced systems designers, the thinking tracks are there to perform the thinking explicitly or in a team. It is a question of wandering on as many tracks as needed in parallel. It may occur that a few tracks dominate for the problem at hand.

Because of the variety in application and the results that are in line with each other, we can state that the tracks are helpful in systems design.

Note that from the application cases presented, the quality of results cannot be judged objectively. For that, further research is needed.

## 7. Conclusions

Systems engineering gives directions for what to do in what order [11]. Systems analysis provides means to investigate existing systems and the environment in which new systems have to operate [11,26]. However, the design or architecture of new systems rely heavily on human skills. Diverse sources have identified the importance of specific “habits of mind” [19].

In this paper, we have summarized twelve thinking tracks that are helpful in investigating a system under design in order to create a better system that fits not only the identified issue but also unidentified parts of the issue. The tracks deliberately help the designer to wander outside the paved path of the well-known systems engineering process. At the same times, the tracks provide guidance in applying systems thinking in practice.

Correlating the thinking tracks to literature on systems thinking shows that these twelve tracks span the systems thinking universe, and that there is no redundancy in this set of tracks (Table 1). Moreover, the tracks make important aspects of systems thinking (as, for instance, presented in [21]) concrete to help the (novice) system designer.

Three preliminary experiments in the form of case studies corroborate the experience of the authors that led to the thinking tracks. These experiments show that a varied combination of thinking tracks helps to quickly dive into a problem and expose the essential characteristics. Even if the content matter knowledge is limited, key factors can be found in a limited time.

In effect, we can state that, next to the well-known pillars process and tools, a third pillar is needed for effective systems design: systems thinking.

Further research is needed to rate the effectiveness of the thinking tracks. We plan to do this in a game setting and in real industry cases. Such a game may also be helpful for novices to learn how to use the tracks. In addition, people have requested aids in selecting and organizing the thinking tracks. Combination with the CAFCR model (CAFCR stands for the Customer objectives, Application, Functional, Conceptual, and Realization views) [30] seems appropriate for this. Another thread for future research relates to the applicability of the thinking tracks outside the engineering domain.

**Acknowledgments:** The authors would like to thank Karel Th. Veenliet for his help and advice in formulating the twelve thinking tracks.

**Author Contributions:** Gerrit Maarten Bonnema and Jan F. Broenink developed and formulated the thinking tracks, where Karel Veenliet was instrumental. The examples and experiments have been performed by Gerrit Maarten Bonnema. The paper is largely written by Gerrit Maarten Bonnema. Jan F. Broenink and Gerrit Maarten Bonnema have performed the final editing of the paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. INCOSE Systems Engineering Handbook Working Group. *Systems Engineering Handbook*, 4th ed.; INCOSE: San Diego, CA, USA, 2015.
2. Bonnema, G.M. Thinking Tracks for Integrated Systems Design. In Proceedings of the 1st Joint International Symposium on System-Integrated Intelligence, Hannover, Germany, 27–29 June 2012; pp. 32–35.
3. Bonnema, G.M.; Veenliet, K.T.; Broenink, J.F. *Systems Design and Engineering: Facilitating Multidisciplinary Development Projects*; CRC Press: Boca Raton, FL, USA, 2016.
4. Osman, M.; Glass, B.D.; Hola, Z. Approaches to Learning to Control Dynamic Uncertainty. *Systems* **2015**, *3*, 211–236.
5. Suh, N.P. Complexity in Engineering. *Ann. CIRP* **2005**, *2*, 581–598.

6. Madni, A.M.; Jackson, S. Towards a Conceptual Framework for Resilience Engineering. *IEEE Syst. J.* **2009**, *3*, 181–191.
7. Merriam–Webster Online Dictionary. Available online: <http://www.merriam-webster.com> (accessed on 8 November 2016).
8. French, M.J. *Conceptual Design for Engineers*; Springer: London, UK, 1985.
9. Suh, N.P. Axiomatic Design Theory for Systems. *Res. Eng. Des.* **1998**, *10*, 189–209.
10. Schön, D.; Bennet, J. Reflective Conversation with Materials. In *Bringing Design to Software*; Winograd, T., Ed.; ACM Press: New York, NY, USA, 1996; pp. 171–184.
11. Blanchard, B.S.; Fabrycky, W.J. *Systems Engineering and Analysis*, 5th ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2011.
12. Kapurch, S.J.; Rainwater, N.E. *Nasa Systems Engineering Handbook Sp/-2007-6105 Rev1*; NASA: Washington, DC, USA, 2007.
13. Maier, M.W.; Rechtin, E. *The Art of Systems Architecting*, 2nd ed.; CRC Press: Boca Raton, FL, USA, 2000.
14. Muller, G. *Systems Architecting: A Business Perspective*; CRC Press: Boca Raton, FL, USA, 2011.
15. Kossiakoff, A.; Sweet, W. *Systems Engineering Principles and Practice*; Wiley Series in Systems Engineering and Management; John Wiley & Sons: Hoboken, NJ, USA, 2003.
16. Sage, A.P.; Armstrong, J.E., Jr. *Introduction to System Engineering*; John Wiley and Sons, Inc.: New York, NY, USA, 2000.
17. Senge, P. *The Fifth Discipline: The Art and Practice of the Learning Organization*; Currency: New York, NY, USA, 1990.
18. Frank, M.; Kasser, J. Assessing the Capacity for Engineering Systems Thinking (Cest) and Other Competencies of Systems Engineers. In *Systems Engineering—Practice and Theory*; Cogan, P.B., Ed.; InTech: Rijeka, Croatia, 2012.
19. Cabrera, D.A. Systems Thinking. Ph.D. Thesis, Cornell University, Ithaca, NY, USA, 2006.
20. Radzicki, M.J.; Taylor, R.A. Introduction to System Dynamics; U.S. Office of Policy and International Affairs. Available online: <http://www.systemdynamics.org/DL-IntroSysDyn/> (accessed on 8 November 2016).
21. Frank, M. Knowledge, Abilities, Cognitive Characteristics and Behavioral Competences of Engineers with High Capacity for Engineering Systems Thinking (Cest). *Syst. Eng.* **2006**, *9*, 91–103.
22. Davidz, H.L.; Nightingale, D.J. Enabling Systems Thinking to Accelerate the Development of Senior Systems Engineers. *Syst. Eng.* **2008**, *11*, 1–14.
23. Richmond, B. Systems Thinking: Critical Thinking Skills for the 1990s and Beyond. *Syst. Dyn. Rev.* **1993**, *9*, 113–133.
24. Hitchins, D.K. *Advanced Systems Thinking, Engineering, and Management*; Artech House: Norwood, MA, USA, 2003.
25. Hitchins, D.K. World Class Systems Engineering. *Eng. Manag. J.* **1994**, *4*, 81–88.
26. Checkland, P. *Systems Thinking, Systems Practice*; 2nd ed.; John Wiley and Sons: Malden, MA, USA, 1981.
27. Boardman, J.; Sauser, B. *Systems Thinking—Coping with 21st Century Problems*; CRC Press: Boca Raton, FL, USA, 2008.
28. Gelb, M.J. *How to Think Like Leonardo Da Vinci: Seven Steps to Genius Everyday*; Delacorte Press: New York, NY, USA, 1998.
29. Eger, A.; Bonnema, G.M.; Lutters, E.; Voort, M.C.V.D. *Product Design*; Eleven International Publishing: The Hague, The Netherlands, 2013.
30. Muller, G.J. CAFCR: A Multi-View Method for Embedded Systems Architecting. Ph.D. Thesis, Delft University of Technology, Delft, The Netherlands, 2004.
31. Martin, J.N. The Seven Samurai of Systems Engineering: Dealing with the Complexity of 7 Interrelated Systems. In Proceedings of the Fourteenth Annual International Symposium of the International Council on Systems Engineering (INCOSE), Toulouse, France, 20–24 June 2004.
32. Boardman, J.; Sauser, B.; John, L.; Edson, R. The Conceptagon: A Framework for Systems Thinking and Systems Practice. In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, San Antonio, TX, USA, 11–14 October 2009; pp. 3299–3304.
33. Gulatti, R.K.; Eppinger, S.D. *The Coupling of Product Architecture and Organizational Structure Decisions*; Working Paper 3906; MIT Soal School of Management: Cambridge, MA, USA, 1996.
34. Conway, M.E. How Do Committes Invent? *Datamation* **1968**, *14*, 28–31.



35. Baumann, H.; Tillman, A. *The Hitch Hiker's Guide to LCA: An Orientation in Life Cycle Assessment Methodology and Application*; Studentlitteratur AB: Lund, Sweden, 2004.
36. Braungart, M.; McDonough, W. *Cradle to Cradle*; Random House: London, UK, 2009.
37. World Solar Challenge. Available online: <http://www.worldsolarchallenge.org/> (accessed on 8 November 2016).
38. Body of Knowledge and Curriculum to Advance Systems Engineering (BKCASE). Available online: <http://www.bkcase.org> (accessed 4 November 2014).
39. International Spring School on Systems Engineering, is3e. Available online: <http://is3e.eu> (accessed on 8 November 2016).
40. Martin, J.N.; Davidz, H.L. Systems Engineering Case Study Development. In Proceedings of the 5th Annual Conference on Systems Engineering Research 2007 (CSER2007), Hoboken, NJ, USA, 14–16 March 2007.
41. Muller, G. Systems Engineering Research Methods. *Proced. Comput. Sci.* **2013**, *16*, 1092–1101.
42. Bonnema, G.M. Insight, Innovation, and the Big Picture in System Design. *Syst. Eng.* **2011**, *14*, 223–238.



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).