Modelling Sustainable Systems and Semantic Web

About the Notion of a Technical System

Lecture in the Module 10-202-2309
for Master Computer Science

Prof. Dr. Hans-Gert Gräbe
http://www.informatik.uni-leipzig.de/~graebe

April 2021

Technical Systems in TRIZ

„… a *number of components* combined to a system by establishing *specific interactions* between the components … assigned to *perform a controllable main useful function* … within a particular context.“ (Glossary V. Souchkov)

A system is a set of elements … that form a *unified whole* that has *properties* that only emerge from the *interaction* of the parts (*emergence* of the main useful function). (V. Petrov, 2020)

Significance of the system operator in TRIZ, non-triviality of the anti-system (N. Feygenson, 2020)

Basics

Despite its importance, the definition of the concept remains vague.

How can the concept of a *Technical System* (TS) be sharpened?

1. Which aspects should be considered?
2. Four dimensions of the concept of a TS.
3. Technical Systems as Black Boxes.

Aspects

Differentiation between design time and runtime

- During design time, the basic collaborative work is *planned*.
- During runtime this *plan is executed*.

This requires to distiguish between interpersonal

- forms of description that are communicated as *justified expectations*, and
- embodiments, which *experienced results* stand in a contradictory relation to the justified expectations.

Aspects

Aspect of Reuse

- This does not apply to most large TS – they are *unique specimen*, even if they are assembled from standard components.
- Most computer specialists also create such *unique specimen*, because the IT systems that control the operation of such large TS are also unique.
- The same applies to government agencies, organzations, etc.

Aspects

Clear distinction between the professions

- of Mechanical Engineering and Industrial Plant Engineering as well as
- the Supplier (specialist) and Master Builder (generalist) of such unique specimen.

Thesis 1:

The speciality of a technical system mainly lies in the *interaction of its components* in a world of technical systems.
Purposes embed this relationality in human practices.

First Approximation

The four dimensions of the notion of a *Technial System*

.

1. The real-world unique specimen.
2. The description of this real-world unique specimen.

For components that are manufactured in a larger number, additionally

3. The description of the design of the system template.
4. The description and functioning of delivery, assembly and operation of the real-world unique specimen that were produced according to this template (e.g. production plan, quality assurance plan, delivery plan, plans for operation, maintenance and service).

First Approximation

TS as Black Box

The basis of the concept is the *notion of an Open System* from the more general Theory of Dynamical Systems.

Existing TS are normatively characterized

- at the level of description through the *specification of its interfaces* and
- on the level of embodiment through *guaranteed functioning according to this specification*.

A TS consists of components, which in turn are TS, their specification-compliant functioning is assumed.

Role of the concept of a TS

The concept of a TS has an epistemic function of (functional) "reduction to the essential".

Human practice is inseparably built into the concept of TS because the concepts "essential", "guaranteed" and "functioning" can only be filled with meaning from these practices.

This makes the widespread in TRIZ distinction between technical and socio-technical systems problematic.

TS as White Box

1. Definition of the concept of a TS.
2. TS and the world of technical systems.

The core of a technical system is ...

... the description of concrete processes by reducing them to the essential with the goal of their practical application.

TS as White Box

The reduction to the essential ...

... focuses on the following three dimensions:

(1) Delimitation of the TS from the outside against an *environment*, reduction of this relationships to input/output relations and guaranteed throughput (Purpose and ability to work).

(2) Delimitation of the TS from the inside by grouping parts as *components*, reducing their functioning on a "behavior control" via their interfaces.

(3) Reduction of the relationships in the TS itself to *causally essential* ones.

TS as White Box

The TS in the World of Technical Systems

The description of a TS is only possible based on descriptions of other (explicitly or implicitly given) TS. The description is preceded ...

(1) ... by a vague idea of the (working) input/output characteristics of the environment.

(2) ... by a clear understanding how the components work beyond their pure specification.

(3) ... by a vague idea of cause and effect relationships in the system itself, that precedes the detailed modeling.

TS as White Box

The concept is based on the availability of existing TS, which are present in (2) as components and in (3) as neighboring systems.

Engineering practices thus take place in a *World of Technical Systems*.

Other systems – components or neighboring systems – are present in the description of a TS only by their specifications.

A prerequisite for the smooth operation of a TS is therefore the guaranteed specification-compliant functioning of the corresponding infrastructure.

Components

1. The concept of a component according to Szyperski.
2. Core concern, cross cutting concern.
3. Components as functional connections.
4. Components as function-object relationships between independent third parties.
5. Components and infrastructure.
6. Norms and Standards.

The World of Components

The concept of a component according to Szyperski

What is a component?
Szyperski gives a simple answer: "Components are for composition".

TS are assembled from existing components. Components can be purchased from third parties or developed in-house.

The World of Components

core concern, cross cutting concerns

Szyperski divides the world of component manufacturing (i.e. TS) in two partial worlds – "design for component" and "design from component".

The first world is the world of component developers, that develop special component functions for business applications – "core concern", this corresponds to the MUF – as *core system function*.

The World of Components

In addition to this core function, the operation of the component requires *supporting functions* (logging, data security, access control, printer control, etc. – "cross cutting concerns") that are based on the use of *established concepts* (description dimension) and services from other, already *prefabricated components* (application dimension), that implement *other technical principles* in other systems.

Thesis 2:

In this sense, real-world components are always *bundles of functionality* that bundle procedural knowledge from *several* areas.

The World of Components

The *component developer* must master all such description forms of functions of supporting components, at least on the abstraction level of their specifications to build useful components.

The second world is the world of *component assemblers*. They assemble (following a pre-existing plan) the system from existing components, develop or modify additional support functions ("glue code"), integrate and test the complete system before releasing it to the customer.

Modularization and standardization

This approach of division of labor between component developers and component assemblers in the field of software engineering is also extensively used in other engineering areas.

Modular systems are widely used and allow the standardization of the design of the unique real-world technical systems.

Components and Frameworks

This requires to connect the *application logic* of the component as "core concern" with the *logic of infrastructural networking* as "cross cutting concerns".

Thesis 3:

The infrastructure logic is usually part of the *component framework* that can only be used effectively if it is *jointly owned by the actors of an entire area of technology*.

Standardization and Trends of TS Evolution

Application logic and infrastructure logic are orthogonal to each other, which means that the trends *4.2 of increasing completeness of a system* and *4.4 of migration to the supersystem* practically counteract to each other in the development of a TS.

Thesis 4:

An improvement in the understanding of the *infrastructure requirements* of interacting components (transition to the supersystem) as description form leads to a *reduction of the level of requirements on the completeness* of individual components.

Standardization and Economies of Scale

Standardization opens up the prospect of economies of scale for standard components. Economies of scale lead to lower costs per unit and thus shift the leading role of competition from competition for the *better technical solution* to the competition for its *cheaper economic manufacturing*.

This means that the S-curve switches at the top of mature technical solutions (including standardization) in the phase of general availability *to another mode* in which the reduction of the economic cost of the availability "state of the art" takes over as guiding function of further development.

Standardization and Economies of Scale

Theseis 5:

The technical "trend 4.1 of increasing (technical) value" changes on the third stage of the development on the S-curve to an economic "trend of decreasing (economic) value".

Or, in economic terms: a demand-driven market turns into a supply-driven market. The same (mature) use value has ever lower exchange value.

Conclusions

Thesis 6:

In the TRIZ theory of TS evolution a better distinction between young and mature technologies is required.

In mature technologies ...

- TS are *bundles of technical principles*,
- which in the descriptive form have the goal of *unity in diversity* (think globally),
- from which in the practice form the *diversity in real-world local application contexts* (act locally) has to be restored.

Conclusions

Thesis 7:

The directed graph of *realized purposes* is the core of relationality in the world of technical systems.

This graph is a global socio-technical artifact and is evolving in the contradictionality of description forms and execution forms.