Systems, Resources, and Systemic Development in TRIZ

Hans-Gert Gräbe, Leipzig University, Leipzig, Germany
graebe@infai.org

Version of June 12, 2022

This is a preprint of the following paper: Hans-Gert Gräbe, Systems, Resources, and Systemic Development in TRIZ, published in the *Proceedings of the TRIZ Future Conference 2020*. The final publication is available at Springer via https://link.springer.com/chapter/10.1007/978-3-031-17288-5_9.

Abstract

In TRIZ theory, resources play an important role when it comes to operate a systemic solution. It is only in this phase that the resource identified in the detailed solution plan as "any type of tangible or intangible matter that can be used to solve an inventive problem" must prove itself in practice. Conceptual distinctions such as "role definition" and "role occupation", which are central for the management of human resources, play only a subordinate role in the TRIZ resource conception.

In this paper, the close connection of the terms resource and component with systemic operating conditions is analysed in more detail and it is shown which influence, for example, the management and reproduction of scarce resources has on systemic development processes in a supersystem.

The resulting questions are compared with corresponding theoretical approaches from component software in order to work out the significance of higher-level abstraction concepts such as component models, component architectures or middleware.

It is proposed to bundle these overarching questions of the interplay of independent third parties providing resources in the huge real "world of technical systems" and thus constituting resource management structures in an new area *Resource Management Analysis* in the TRIZ theory corpus.

Keywords: systemic approach, resource, operating conditions, place and content, interfaces, component models.

1 The Aim of This Paper

The aim of this paper is to analyse the concept of a resource in the conceptual framework of TRIZ in more detail and in particular to analyse its relationship to systemic operating conditions. Further, the TRIZ resource concept is compared to resource concepts in the conceptual foundations of component systems in technical domains with a focus on Component Software.

Developed engineering disciplines are characterised by the extensive use of components that are developed, produced and offered by independent third parties, thus bringing the terms *resource* and *component* close together. It was only with the transition to software components that computer science developed from an "Art of Programming" into an engineering discipline and thus embarked on this path of a systemic development towards a mode of production based on the deep division of labour, which other engineering disciplines had already taken before. The theoretical concepts that accompanied this development towards Component Software are therefore of particular interest, as experiences from other engineering domains were taken up and further elaborated during this development.

The connection between a viable resource concept and the interrelation of systemic development processes of technical functionalities and their bundling in technical systems over longer periods of time is presented as a specific form of "organisation of material" in the sense of [14, p. 98], which institutionalises itself in patterns, norms, standards, component models and finally as "state of the art". In particular, it is shown the significance of *component models* and *component frameworks* in developed component architectures for a qualified development of systems of resource management. The explanations deepen the view on the conceptual foundations of TRIZ developed in [2].

In this paper, such approaches are discussed only in the scope of reuse of technical functionalities in cooperative action *within* larger companies. In another paper [3], this question is discussed for cross-company cooperation and the importance of open architectures is elaborated.

The results may remain unsatisfactory insofar as only *questions* of an appropriate conceptualisation are raised. It is suggested that TRIZ theory be extended to include the instrument of a *Resource Management Analysis*, which links classical and Business TRIZ and addresses the interrelation between problems of short-term operational resource provision and long-term development of resource availability.

2 The Concept of a Resource in TRIZ

In [21] Wessner collected some common definitions of the resource concept from various TRIZ scholls. All these definitions focus more on the availability than on the structure and material composition of resources.

In general, in TRIZ the resource concept is mostly used in an intuitive way, without making any effort to establish a precise conceptual foundation. In the *TRIZ Body of Knowledge* [6], for example, the word "resource" only appears in item 1.6 as "substance-field resource" and in the title of four publications [10], [11], [13], [22] listed there. In common TRIZ glossaries ([16], [19], [18], [5], [9]), only [16] and [9] explain the term *resource.* In [16] it is defined as "any type of tangible or intangible matter that can be used to solve an inventive problem: time, space, substances, fields, their properties and parameters, etc.". The object of *Resource Analysis* is derived from it as "examination of resources available in the technical system and its supersystem in order to compile a list of resources that can be used for solving a particular inventive problem" (ibid.).

In section 4.2 of the textbook [4], Resource Analysis is discussed on 13 pages and with material, field-like, spatial, temporal, informational and functional resources six types of resources are distinguished. Specific *qualitative* determinations of such "substances and fields" as resources play almost no role in the classification proposed in [4, p. 51-52] according to *value* (free, not expensive, expensive), *quality* (harmful, neutral, useful), *quantity* (unrestricted, sufficient, insufficient) and *readiness for use* (ready, to be modified, to be developed). Such qualitative determinations in the sense of the fulfilment of *specifications* are, however, essential in more complex technical contexts in order to ensure the *operation* of a specific functional property, which is to be provided by the systemic context.

Matvienko estimates in [9] the current state of conceptual penetration of the notion *resource* as follows:

> A resource is in general a set of systemic properties of an object not previously used to solve some inventive problem. It is not defined in TRIZ in any way, although there are numerous methodologies for finding resources, resource tables, resource lists, etc.
>
> Despite the obvious abundance of methodical literature on the subject, the search and use of resources in specific conditions of practice always remains subjective, because no problem solver can ever reliably know whether or not a given system property has been used before to solve this inventive problem.
>
> As a rule, in one and the same technical system, acting under the same problem conditions using the same methods, different solvers find completely different resources, often even of different epistemological level.

The aim of a systemic modelling of a problematic situation in TRIZ is not only, and not so much, to develop a functionality that solves the problem *potentially*, but to develop the solution up to its practical operational use. For such a practical operation, however, *operating conditions* must exist or be established, which include the use of prestructured resources, which "exist or can be easily produced" only to the extent as this is provided by a developed market structure.

Such a *use* of resources both in artefact form (as objects) and functional resources (components) is an essential point of the *implementation* of a solution plan and the subsequent operation of the solution, i.e. of the "system to be", in the interplay with the operation of both the components and the neighbouring systems. This qualitatively and quantitatively determined availability of substance, energy and information is closely connected to the concept of resources, but requires much more structure than just the (better) exploitation of "any available type of tangible or intangible matter".

3 On the Systemic Approach

The systemic approach is one of the central methodological elements of TRIZ theory. As a problem-solving methodology it unfolds its advantages if it is possible to work out a *contradiction* within the requirements, to delimit this contradiction spatiotemporally in an *operative zone*, to demarcate it from an *environment* in a systemic way and to analyse the problem more precisely within such a well delimited system.

Through such a threefold delimitation the horizon of consideration is focussed – by demarcation from the outside against an *environment*, by internal delimitation against *components* and by limiting the *relations* between these components to be considered to essential ones, see [2] for details.

Typically, TRIZ methodology is limited to describing a solution *plan* for transforming a "system as is" into a "system to be" and does little focus on the implementation of that plan.

The four phases Define, Select, Generate and Evaluate, into which the TRIZ solution process is divided in [8], end with such a taylored solution plan only.

In this section, the relationship between such a solution plan and its implementation is examined in more detail. It will become apparent that the TRIZ concept of *ideality* corresponds to the concept of *pure function*, whose "viability" (a term from [14]) only emerges and can emerge in the course of practical implementation through connection with a "viable environment".

3.1 Systems and Emergent Functions

Systems are characterised by the fact that they realise *emergent functions* which cannot be reduced to individual parts of the system, but result from the interaction of these parts [12, p. 17]. For a system considered from the outside as a Black Box, such a *main useful function* as *main parameter of value* is in the foreground. Usefulness, expediency and purposefulness embed the (technical) system into larger socio-cultural contexts and justify the existence of the system itself.

On the respective system level, therefore, the appropriate arrangement and interplay of these *relations* play a leading role, whereby a distinction is to be made between the dimensions of structural and processual organisation. In the interrelationship of both dimensions the fundamental contradiction of every systemic approach does manifest itself – the contradiction between decomposability and unity in the categorical part-whole relationship.

Petrov [12] emphasises that for analytical purposes, the system *must* be disassembled, but it can be operated only in assembled state. This inherent contradiction between decomposability and wholeness does not end at the boundary of the system: the operation of the assembled system in turn requires a qualitatively and quantitatively determined *throughput* of substance, energy and information. Even if the decomposition of a system into its parts provides important insight into its functioning, only in assembled state the system can be operated and thus unfold its specific functionality. In this sense every systemic approach reduces in a certain way to a *conditional mind game*.

In the TRIZ notion of a *minimal technical system*, a *tool* acts on an *object* (workpiece) to be processed in order to transform it into a *useful product*. The concept of the *ideal system* [4, p. 40] considers the tool as a purely functional property, the effect of which to intentionally change the state of the workpiece to a useful product is achieved without any additional efforts and any wear of the tool. In other words, it is not the *real* tool but the *imagination of the tool* that creates the required action in such an *ideal machine*.

3.2 Systems and Their Operating Conditions

This is, of course, only an ideal picture, since in addition to the structural design the *operation* of the system and thus a *throughput* of substance, energy and information through the system is required in a qualitatively and quantitatively determined form. This aspect is somewhat underexposed in TRIZ, as the usefulness of a system is primarily defined in terms of its *main useful function* [4, p. 40], i.e. in its *potential* usefulness.

For the *real* usefulness, the mentioned three types of throughput must be organised, i.e. the system must have *resources* at its disposal for its operation. In the classical understanding of a *complete technical system* [7, 4.2], [19, p. 9] the energy throughput is centered on the tool, the throughput of substance transports the workpieces and the throughput of information is directed to the control of the action. Thus, in any case, the concept of a resource is understood in [4, p. 51] and also [19, p. 7] as “means that can be used to solve a problem.”

The understanding of the relationship of action conveyed here is asymmetrical. An active tool has a state-changing effect on a passive workpiece, while retaining its own functionality and – ideally – without undergoing a state change itself. In substance-field models this understanding is replaced by a more symmetrical model of a field-mediated action between two substances. At the same time, in the systemic abstraction, the materiality of the tool is pushed back further from the tool to the action of a field, and a component concept is prepared as proposed by Szyperski [17] for Component Software. There, *components* are basically conceptualised as *stateless* with all resulting consequences. In contrast to this, *objects* are conceptualised as state-bearing units of instantiation to maintain a certain standardisation of workpieces required for a repeated application of a function within a production process.

Such an approach also corresponds well with the widespread organisation of production processes, where a distinction is made between operating and maintenance mode. In the operating mode, the focus is on the functional properties of the tool, while in the maintenance mode its material properties are focused. As an independent technical system in a narrower sense, only the operating mode is modelled as the target of a “problem solution”. The maintenance mode is part of the supersystem, which is concerned with the *reproduction* of the tools as *resources* used in the operating mode. In the (classical) operating mode the focus is on the use of tools and the material throughput of workpieces, which are thereby transformed into useful products, in many cases *technical artefacts*, which are either further processed as semi-finished products in a following technical system or enter into such contexts as tools themselves. In both cases the useful product is a *resource* for further systemic processes.

This roughly outlines what must be conveyed by the concept of a resource in a systemic context. As already explained above there exists a whole variety of resource concepts proposed by different TRIZ schools. Let us take a closer look at Souchkov’s definition in [4, p. 51], where a resource is understood as “a means, a tool to carry out an action or to make a process take place” and equipment, money funds, raw material, energy or even people (human resources) are mentioned as examples of resources. Souchkov also sees *Resource Analysis* as an essential component of TRIZ with two goals:

- Analysis of the resources that are to be *treated or consumed* in the course of a process,
- and analysis of the resources that can be *used* to carry out the process or to solve the problem,

i.e. he distinguishes resources of the first kind, which undergo state-changing transformations as *workpieces* and resources of the second kind, which are used as tools to *mediate* these state changes.

3.3 Systemic Development and Problem Solving

While the focus of our considerations so far has been on the operating conditions of a *given* technical system, TRIZ is about problem solving and thus it is concerned with the design of viable technical systems in a *systemic development process*. For this purpose the role of Resource Analysis is defined more precisely in [4, p. 51]:

> A technical system has different resources at its disposal for the completion of its function. A function can only be completed using suitable resources. Resources are therefore elementary building blocks of a problem solution. The skilful use of resources distinguishes an efficient from an inefficient system.

The question of systemic operating conditions is thus reversed – it is not about what conditions are *required* for the operation of a particular system, but what kind of system under *given* operating conditions promises an efficient problem solution. The focus thus shifts from the operating conditions of an existing system to the question of a systemic development under given conditions. This systemic development can cover a complete genesis of a system from the scratch, when vague technical solution concepts have to be detailed and developed into a full size practical solution. In most cases, however a working technical system already exists, in which deficiencies have to be overcome, often resulting from changes in operating conditions. Such a conception of the development of a "system as is" to a "system to be" is the core of the TRIZ ontology project [18], which aims to further sharpen TRIZ conceptualisations.

In both approaches, a *sustainable* problem solution requires the *sustainable* availability of the necessary resources in the "environment" to operate the system. Hence in the next section the structure of this "environment" is detailed in which these resources are to be found.

4 The World of Technical Systems

The operational demand of a technical system is fixed in the form of *specifications* as requirements to the "environment", which must be fulfilled for the *operation* of the (assembled) system. The "reduction to the essentials" that characterises the systemic approach is, as already stated above, only a *conditional mind game* that presupposes a sufficiently powerful *environment* as given, in which the necessary resources can be allocated to fulfil the operating conditions.

However, this environment consists of similarly structured systems. Hence the *coupling* of these specifications comes into focus. Technically these specifications are transformed into *interface definitions*, and the specifications are divided into input and output specifications in order to differentiate which resources a system requires for operation and which it produces and makes available to other systems. Those interface definitions are a moment of decomposition of the unity, because it affects *two* systems that are evolving separately. In the simplest case the *agreement* on the interface definition takes place in a *supersystem* which covers both systems. Altshuller's development laws of "'energetic conductivity' of a system", of "coordination of the rhythms of the parts of a system", of "transition to a supersystem" and to a certain extent also of "transition from the macro-level to the micro-level" [1, p. 72-74] address different aspects of this problem of coordination of interfaces.

4.1 Components, Interfaces, Component Models

Sommerville [15, ch. 6.4] emphasises the importance of such interface specifications for the development of software systems that "need to interoperate with other systems that have already been developed and installed in the environment" (ibid). The same perspective is significant when large systems are to be created in a cooperative development process and for this a decomposition into subsystems is required that are to be developed independently of each other [15, ch. 10.2].

Such component-based development scenarios are of growing importance over the last 20 years and developed to an established approach in Software Engineering, even if no reusable components from third parties are used [15, p. 477]. Systemic development manifests itself as a concurrent process of parallel in time developments and unfolding of subsystems, which is controlled by a socio-technical supersystem of project coordination.

In the V-Modell XT [20], for example, a process model of software development widely used in Germany, the requirements elicitation and system specification are carried out in this supersystem in cooperation between the client and contractor. It concludes with the requirements specification as a detailed (legally binding) agreement between both sides. This part of the process is similar to part 1 of ARIZ-85C. It is followed by the definition and development of the *architecture* and the *design* of the system including the *component specifications* as a prerequisite and reference for the parallel development of the individual components. At the end of the development process, these pieces are separately tested in *component tests* and based on an appropriate *integration strategy* assembled into the overall system. The behaviour of the whole system with regard to the functional and non-functional requirements is validated in various *system tests*.

Sommerville [15, p. 477] emphasises that this development process in turn requires a more extensive socio-technical infrastructure with

1. *independent components* that can be fully configured via their interfaces,
2. *standards for components* that simplify their integration,
3. a *middleware*, which supports the component integration with software
4. and a *development process* that is designed for component-based software engineering.

Components are thus conceptually integrated into an overarching *component model*, which essentially ensures the technical interoperability of different components beyond concrete interface specifications and thus forms a moment of unity in the diversity of the components. However, this unity extends not only to the model, but also to the operating conditions of the components (as "viable" functional property provided by the middleware) as well as to their socio-technical development conditions (as a partial formalisation of the development process). This frame constitutes as *component framework* [17, ch. 9] a socio-technical supersystem as an "environment" of components that were created according to the specifications of that component model. At that supersystem level a subdivision of functional properties to be used or to be developed into *core concerns* and *cross cutting concerns* allows for further synergetic effects of a division of labour also on higher levels of abstraction, such as the *CORBA services*, which themselves have component character, but are provided by the CORBA platform as *services* (i.e. as "living components") [17, ch. 13.2].

4.2 Functional and Attributive Properties

The explanations show that systemic development processes even within a single company working on component-based foundations are interweaved in many ways and cannot be described solely on the level of lines of development of individual technical systems. Szyperski

[17] shows clearly that the component approach is an approach of reuse that is not limited to the (possibly modified) abstract reuse of the technical functionality of a problem solution, but always reuses components together with their operating conditions as *services* and thus not detached from their environment.

For this, Shchedrovitsky's distinction between functional and attributive properties in the categorial relation of part and whole, as well as the distinction between the notions of *part* and *element* are essential. This cannot be elaborated here in more detail due to lack of space and is reduced to the quotation of essential points in the words of Shchedrovitsky himself.

> *Elements* are what a unity is made up of, so an element is a part inside the whole, which functions inside the unity, without as it were being torn out of it. A simple body, a part, is what we have when everything has been disassembled and is laid out separately. But elements only exist within the structure of *connections*. So an element implies two principally different types of properties: its properties as material, and its functional property derived from connections.
>
> In other words, an element is not a part. A part exists when we mechanically divide something up, so that each part exists on its own as a simple body. An element is what exists in connections within the structure of the whole and functions there. [...]
>
> *Functional properties* belong to an element to the extent that it belongs to the structure with connections, while other properties belong to the element itself. If I take out this piece of material, it preserves its *attributive properties*. They do not depend on whether I take it out of the system or put it into the system. But functional properties depend on whether or not there are connections. They belong to the element, but they are created by a connection; they are brought to the element by connections. [14, p. 93-94]

4.3 Functional Properties and Ideality

In the TRIZ methodology of the genesis of a system, these functional properties as "usefulness for others" are in the foreground. An engine as itself is not interesting, but only as an engine that drives a vehicle and is therefore "useful". The terms *usefulness* and *harmfulness* play an important role in TRIZ alongside the objectives of profitability and efficiency as socio-cultural guiding principles. With the concepts of *Ideality* and *Ideal Machine* [4, ch. 4.1] a mental construct of anticipation of the functional properties of a system stands at the beginning of its genesis. "The ideal machine is a solution in which the maximum utility is achieved but the machine itself does not exist." [4, p. 40]. The ideal machine is therefore pure functionality, pure "connection" in the sense of Shchedrovitsky, without any resource-related underpinning. Nonetheless, that fictitious idea, reminiscent of the fairy tale of Cockaigne, is central to TRIZ, for it develops a strong orientation towards the intended usefulness and thus has a socio-cultural guiding effect.

4.4 Place and Content

In the further system genesis, this conceptual frame of functional properties has to be filled with suitable resources [4, ch 4.2]. The systemic concept turns out to be a kind of magnifying glass, under which the combination of the functional properties, filling the "connections" with resources can be followed. To describe this composition process Shchedrovitsky distinguishes the concepts *place* and *content*.

> An *element* is a unity of a place and its content – the unity of a functional place, or a place in the structure, and what fills this place.
>
> A *place* is something that possesses functional properties. If we take away the content, take it out of the structure, the place will remain in the structure (assuming that the structure has a conservative and rigid nature), held there by connections. The place bears the totality of functional properties.
>
> The *content* by contrast is something that has attributive functions. Attributive functions are those that are retained by the content of a place, when this content is taken out of the given structure. We never know whether these are its properties from another system or not. Now we might take something out as content, but it is in fact tied to another system, which, as it were, extends through this place. [14, p. 94]

The search for resources as "content" is constitutive for the process of confinement in the course of the implementation of the system that is to be developed from the pure functionality of the ideal machine. This corresponds to Altshuller's first law of development of "completeness of the parts of a system": "The necessary condition for the viability of a technical system is the existence of the main parts of the system *and* their minimal functionality (i.e. viability – HGG)." [1, p. 72]

However, the thing viewed with the magnifying glass as a connection of place and content remains a "dead body", because "a living being has no parts" [14, p. 91]. Beyond the connection of place and content an operational process dimension is essential for a living system. It is not enough to insert the plug ("place") of an electrical appliance into the socket ("content") to bring the appliance "to life". The fit of plug and socket guarantee a certain minimum compliance, but to operate the device, the socket itself must be "alive" and make electrical energy available in precisely specified quality and quantity. The resource plugged in as "content" at this "place" requires an at least rudimentary system of resource "lifecycle" management.

5 Systemic Development Processes in a Modern Society

This is a typical phenomenon of a modern society, in which the electricity comes from the socket and the milk from the shop. The division of labour in such a modern mode of production leads to the emergent phenomenon of social unity and stratification of the reproduction of infrastructural conditions.

In a developed country, one can rely on electricity coming out of the socket and can use it at any time for devices that run on electrical power, provided that the technical standards such as operating voltage and power consumption are adhered to and a suitable plug-socket combination is used. The existence, reliability and robustness (resilience) of such an infrastructure has a significant influence on the way people organise their daily lives. Even in a less developed country where a continuous supply of electrical power is not guaranteed, it is still possible to use electrical devices. However, a coordination effort is required to match the availability of electrical power and the working processes in which the electrical equipment is used. Altshuller's "Law of coordinating the rhythm of the parts of a system" [1, p. 73] is thereby seemingly reversed into its opposite – the more perfect the infrastructure, the less there is a need for coordination with that black box of power supply. Nevertheless the law is not invalidated, because the stable availability of electricity as a resource requires a sophisticated management *inside* the power supply system.

These requirements of coordination grow even more markedly in the transition from classical electricity supply systems with clearly defined base loads and unidirectional power distribution to modern systems of decentralised power generation based on "renewable energies". The cascade of trends from coordination, controllability and dynamisation [7, p. 6] is becoming increasingly effective and, with smart meter concepts, also reaches the end consumer, who is thus raised to a more comfortable level of rhythmic coordination.

These developments in the electricity supply system, however, are in turn dependent on a digital infrastructure, in which machine-readable descriptions of control information circulate. Evolutionary technological development in the web as one area of technology leads to disruptive changes in this power supply system as another area of technology. The future will show whether those reserves of control potential beyond the (present) limits of the power supply system will be used or whether the *systemic decoupling* associated with an unconditional stable power supply as *anti-trend* to increasing coordination has a socially higher value.

6 Summary

In modern component architectures, the concepts of resource and component move closer together. In a "world of technical systems", artificial artefacts combine functional and material properties that link their usefulness for a certain purpose in a structural system design of a more complex unit with the guarantee of operation, if the necessary operating conditions are provided in the "living" operating environment. However, this fundamental capability of a socially provided resource infrastructure, which is also legally fixed in the concept of the "state of the art", requires an actice reproduction. The management of scarce resources and the preemptive development of resource pools are essential forms of collective action that extend beyond the narrow horizon of individual companies.

In the term *Resource Management System*, the concepts of resource and component are equally present. However, socio-technical abstractions on a higher process level such as component model, middleware, component architecture, etc. are required to describe corresponding operating conditions of such a "system of systems".

In concrete technical domains, such conceptual worlds have long been developed and are waiting to be included and generalised in the methodological toolkit of TRIZ. "Components are for composition" [17, ch. 1.1] is a short definition by Szyperski and those *rules of composition* in turn constitute a diversity of socio-technical development processes corresponding to the diversity of component models, which provide different environments of systemic development processes of concrete components. Szyperski, for his part, analyses in [17] this diversity of compatibilities and incompatibilities of different component models and identifies different levels of abstraction for the reuse of concepts that go beyond the use of prefabricated components. In his 20-year-old book he already emphasises a diversity of conceptual notations as

> the growing importance of component deployment, and the relationship between components and services, the distinction of deployable components (or just components) from deployed components (and, where important, the latter again from installed components). Component instances are always the result of instantiating an installed component – even if installed on the fly. Services are different from components in that they require a service provider. [17, p. xvii]

In our modern "world of technical systems" the question of resources to be used in a systemic problem-solving context has to cover the condition that resources are both offered and required in a highly pre-structured form. These pre-structures rely on *standards*, are the basis for *component models*, and are supported in that "world of technical systems" by "living" *technical infrastructures*.

Trends of increasing coordination, controllability and dynamisation [7] refer not only to system-internal development lines, but also to the coordination *between* systems which are developed, offered and operated by independent third parties. The systemic development of such infrastructural frameworks, for example, of the power supply system, as supersystem has to take into account the relations of *mutual interdependency* of such independent third parties in a modern industrial mode of production and thus forces of socio-cultural self-organisation on the inter-company level of such a supersystem as target of a forthcoming TRIZ concept of a *Resource Management System*.

References

[1] Altshuller, G. S.: Creativity as an Exact Science. Quoted from the Russian edition. Moscow, Sov. Radio (1979).

[2] Gräbe, H.-G.: Technical Systems and Their Purposes. In: Mayer, O. (ed.): Proceedings TRIZ-Anwendertag 2020. pp. 1-13. Springer Nature (2021).

[3] Gräbe, H.-G.: Components as Resources and Cooperative Action. Accepted for Publication in the Proceedings TRIZ-Anwendertag 2022.

[4] Koltze, K., Souchkov, V.: Systematic Innovation Methods (in German). Hanser (2017).

[5] Lippert, K., Cloutier, R.: TRIZ for Digital Systems Engineering: New Characteristics and Principles Redefined. Systems 7, 39 (2019). doi: 10.3390/systems7030039

[6] Litvin, S., Petrov, V., Rubin, M., Fey, V.: TRIZ Body of Knowledge. MATRIZ Website (2012).

[7] Lyubomirskiy, A., Litvin, S., Ikovenko, S., Thurnes, C. M., Adunka, R.: Trends of Engineering System Evolution (TESE). TRIZ Consulting Group (2018).

[8] Mann, D.: Hands-On Systematic Innovation for Business and Management. IFR Press (2007).

[9] Matvienko, N.N.: TRIZ Encyclopedia (in Russian). `https://triz.org.ua/works/ws72.html`

[10] Petrov, V.M.: Principles of the Theory of Resource Utilization. Leningrad (1985).

[11] Petrov, V.M.: A Technology of Resource Utilization. – Theory and practice of teaching engineering creativity. Abstracts of scientific papers. Chelyabinsk: UDNTP (1988), pp. 55-56.

[12] Petrov, V.: Laws and patterns of systems development (in Russian). Independent Publishing (2020).

[13] Royzen, Z.: Specific Features of Resources Utilization for Problem Solving and Improving Obtained Solutions. Kishinev (1986).

[14] Shchedrovitsky, G. P.: Selected Works. A Guide to the Methodology of Organisation, Leadership and Management. In: Khristenko, V.B., Reus, A.G., Zinchenko, A.P. et al.: Methodological School of Management. Bloomsbury Publishing (2014).

[15] Sommerville, I.: Software Engineering. Citations based on the 8th German edition. Pearson Studium (2007).

[16] Souchkov, V.: Glossary of TRIZ and TRIZ-Related Terms. The International TRIZ Association, MATRIZ. 2018 (2014, 1st ed.)
`http://www.xtriz.com/publications/glossary.htm`

[17] Szyperski, C.: Component Software. 2nd edition. ACM Press (2002).

[18] TRIZ Glossary of the TRIZ Ontology Project.
`https://triz-summit.ru/onto_triz/100/`

[19] VDI: Norm 4521, Blatt 1. Inventive Problem Solving with TRIZ. Fundamentals, Terms and Definitions. September 2021.

[20] Weit e.V.: V-Modell XT. Release 2.3 (in German). 2020.
`http://weit-verein.de/`

[21] Wessner. J.: Resource-Oriented Search. In: Mayer, O. (ed.). Proceedings TRIZ-Anwendertag 2020. pp. 93-105 and 106-113. Springer Nature (2021).

[22] Zlotin, B.L., Vishnepolskaya, S.V.: Use of Resources in Search for New Engineering Solutions. Kishinev (1985).