Nikolay Shpakovsky
PhD, TRIZ specialist
Company "TRIZ-profi", Moscow

Abstract of book "Evolution Trees. Analysis of technical information and generation of new ideas"

About the book

The book presented by this abstract is an expanded and enlarged version of the electronic book "Processing of Technical and Patent Information by Using Evolution Trees" developed jointly with the company Mitsubishi Research System and published in 2003 in Japan.

The peculiarity of the approach presented in the book consists in using the so-called "Evolution Tree" – an organized set of technical system evolution patterns - to structure technical and patent information. Using the Evolution Tree provides tangible advantages at all stages of dealing with information – searching for, analyzing and producing of new information. Search enables us to perform preliminary marking out of the "information field", which immediately increases the found information relevance. The Evolution Tree is easy enough to analyze due to the demonstrativeness and objectiveness of the presented data. Acting in accordance with the special method described in the book and illustrated by examples, we may fill blanks on the Evolution Tree at the stage of analysis. The Evolution Tree also helps obtain new ideas and technical solutions, for example, by using our method of structural analogy, as well as other innovative methods.

Special consideration was given to two main cases of practical application of the Evolution Tree – avoiding a competing patent and forecasting a technical system evolution which were illustrated by examples, in particular, by analyzing the bypassing of the patent for a device for feeding a conditioning solution into a washing machine. The book comprises many examples based on practical experience.

The text of the book is presented on 250 pages and comprises 350 illustrations and 270 examples, part of which is presented as the display Evolution Tree and the remaining part is an expanded description of transformations of specific systems.

Introduction

To meet competition, each company exerts every effort to improve the product quality. In the calmness of laboratories as well as at testing benches, spadework is being done which results in the generation of new ideas, new knowledge laying the foundation of a new promising product. Huge amounts of patent and technical information are collected and processed for examining all versions of a particular type of product available on the market, develop and manufacture competitive goods.

This requires some classification structure – a map which would allow visualizing all existing versions of a product. It would be wonderful if such a map could tell us as much as possible about still unrealized versions. We have developed the so-called "Evolution Tree" which may perform the role of this map. This book is devoted to the methods for constructing and creating such a tree. The methods are based on the TRIZ tools (TRIZ stand for the Theory of Inventive Problem Solving created by H. Altshuller). This theory has proved effective in solving complicated technical problems. We think that the main theses of this theory are effective for describing and analyzing different versions of a technical system. The fact is that one has to deal with scattered, incomplete information. The methods for processing such information presented in this book and based on the use of the Evolution Tree allow carrying out an effective analysis of large volumes of obtained information as well as finding interesting innovative ideas.

With patent and technical information properly organized within the Evolution Tree, one can make a correct forecast of the company's product evolution, which allows reasonable business decisions to be taken. Using the evolution trees also offers good opportunities for bypassing the competing companies' patents and protecting one's own technical solutions, including creating of so-called "patent umbrellas".

Generally speaking, the Evolution Tree is effective where it is necessary to see the entire picture of the technical system evolution, all its basic versions, including the most unique and promising ones. This is particularly important, for example, for strategic planning of production, innovations, and business. Evolution Trees underlie the effective methods for patent and technical information processing which have tangible advantages over conventional approaches.

This book is meant for people of different professions who are interested in innovations. These are first of all designers and engineers, as well as students of technical universities interested in inventive work. The material presented in the book will also be interesting and useful to heads of enterprises, businessmen, people engaged in promotion of new goods and interested in making them competitive.

The author of the book is grateful to everyone who participated in the development of this approach, as well as in discussing and updating the book text, first of all to the colleagues from the "Invention Machine" project, companies SAMSUNG and TRIZ-profi.

A special gratitude is to Viktor N. Baturin who assembled a team of creative people, a group of TRIZ-profi and organized work on interesting projects which are important for Russia. The atmosphere of cooperation, open creative discussion characteristic of our team was very useful for fruitful work on the book.

Content of Chapter 1. "Structuring of information field"

In forecasting the evolution of technical systems and searching for new ideas, we deal with information units - descriptions of versions of a system of interest. To forecast, it is convenient to imagine that all these embodiment versions are arranged within some limited region, a kind of "prognostic field". It is much easier and more convenient to travel across such a field, when the system's versions are known and arranged in a convenient order so that we can see all of them together – both existing ones and those which may appear in future.

In this situation, we can construct an effective information organization model, a kind of information structure to be used in future to find new concepts. This effective information structure should itself point to missing versions of the system and suggest which new versions should be sought for in the first instance.

When solving a prognostic problem, search for a new concept may be illustrated by the diagram, presented in Fig. 1. First, the initial situation is analyzed, then, a new concept, a new knowledge is synthesized. The analysis stage is accompanied by a patent search. The maximum amount of information is collected on the technical problem of interest, the prototype of the system to be improved. The peculiar feature of this stage is that it is accompanied by the extension of our knowledge of the prototype and the design problem.

The researcher structures the obtained information according to his own classification criteria. Then the analysis itself of the produced structure is carried out and new ideas and technical solutions are generated. The higher the quality of the information that fills the structure, the more logical its classification and the higher the quality of the analysis, the more efficient is the second information processing stage – *synthesis*, construction of new concepts.

The main problem consists in the absence of an efficient method for the organizing and classifying of technical and patent information. The problem is that information obtained by patent search is specific and comprises data about different versions of the *same* technical system. All the embodiments have similar designation, produce the same product, and perform the same useful function. The development of an effective classification method for the information obtained through patent search is a key problem on the way toward the improvement of technical information processing. The most effective would be an information structure capable of presenting all known basic versions of a given technical system at once. In addition, of great importance is

the ability of such a system to provide information about possible, but non-existing versions of the technical system. With a full and logical information structure available and the knowledge of the modern state of art, one could reasonably determine the initial point of analysis in making a forecast. After determining which performance parameters have already reached their potential, we can determine, with a high degree of accuracy, expected technical system versions, i.e., to generate prognostic technical solutions.
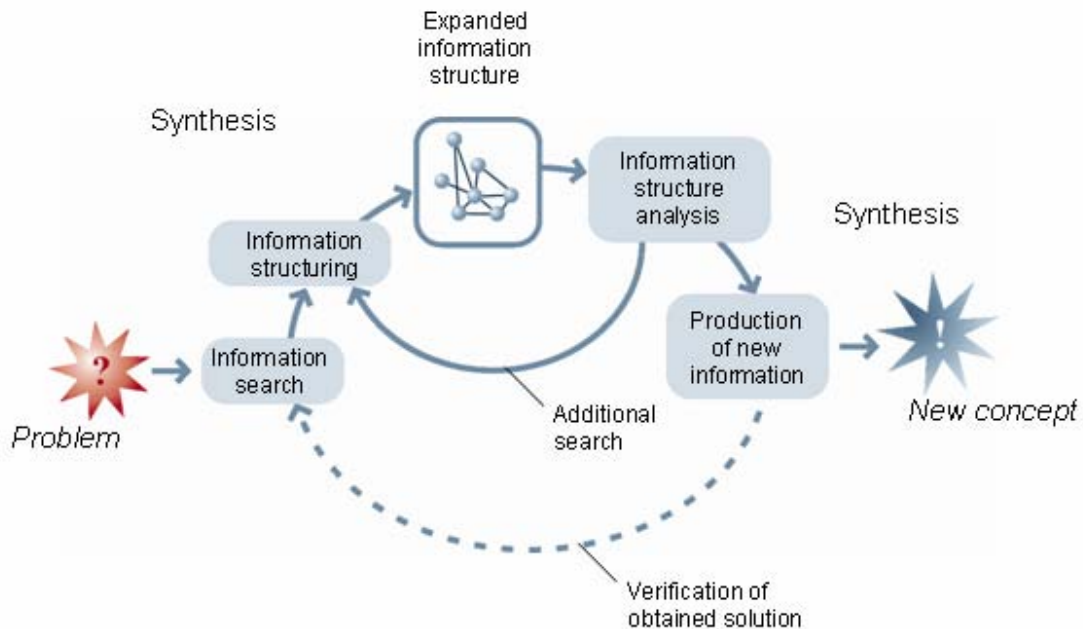

Fig. 1. Searching for a new concept

The problem of effective information classification is also urgent and actively studied in the patent area. There exist and are actively developed special patent information classification systems, where patents are arranged in a definite order, i.e., an array of information has a definite *structure*. Such structures are quite logical at the level of sections, presenting *different technical systems*, each of which is designed for performing its own useful function. However, inside the sections, which host *different versions of one and the same technical system*, patent is structured in a random, sometimes chaotic manner. The thing is that the functional criterion is unsuitable inside the sections, because all versions of the technical system are designed for performing the same useful function; therefore, the results of the using of many, often unrelated classification criteria, may be traced here. These may be both design features and the operation peculiarities of a system described in a patent. As a result, information organization and search in the patent base is difficult enough.

What is the best method for classifying the technical system transformation versions?
Arranging them in a non-structured list or a single-level table practically does not add any new information in addition to that obtained by means of search. To organize and present structured information, a tree-like structure may be used where transformation versions are arranged according to some hierarchic principle. Detected variants may be grouped by patenting companies or by filing year. Such a classification will provide new information – about the patenting dynamics, patent priority, leading companies in the production of a given product. The value of the initial information grows in this case. One of the effective patent information classification methods is classification by the principle of operation or some characteristic feature. This is just the method employed in WIPO patent classification subclasses. It is more effective then other methods but still has a serious disadvantage as it is not easy to determine whether all possible versions of the system were considered or something was lost.

Some fuzziness of patent classification only leads to some difficulty in finding a needed patent, whereas search for innovation concepts requires a more harmonic and logical classification structure. Thus, optimal information structure is a key to new concepts and technical solutions.

What should a structure for the grading of technical and patent information be like?
To answer this question, let us formulate a kind of "design specification" which is *a list of requirements* for the classification method being developed.

- Classification should be based on objective criteria (*objectivity requirement)*.
- It should take into account all system transformation versions differing significantly from each other (*completeness* requirement).
- The classification method should be general for all technical systems and at the same time it should be suitable for describing the transformations of a specific system (requirement of *generality* and *specificity*).
- The information presentation method should provide maximum demonstrativeness and show the presence of gaps, lacunas in the data base obtained by means of patent search (*demonstrativeness* requirement).
- The structure should contain data about missing information units, sufficient for obtaining concepts for their practical realization (*informativeness* requirement).

Out task is building an information structure capable of satisfying all these requirements.

Conclusion: Chapter 1 formulates requirements for an effective information structure.

Content of Chapter 2. "Objective evolution patterns"

- **Concept of evolution patterns**

To satisfy the objectiveness requirement, it is necessary to find a method for arranging the technical system versions obtained by means of patent search in certain sequences aligned in accordance with objective criteria. This problem is considered by the Theory of Inventive Problem Solving (TRIZ) which is based on the objective laws of technical system evolution, the action of which is illustrated by the "technical system evolution patterns". *The evolution pattern is a series of sequential transformation versions of a technical system or its element, arranged by a change in a certain parameter of this system.*
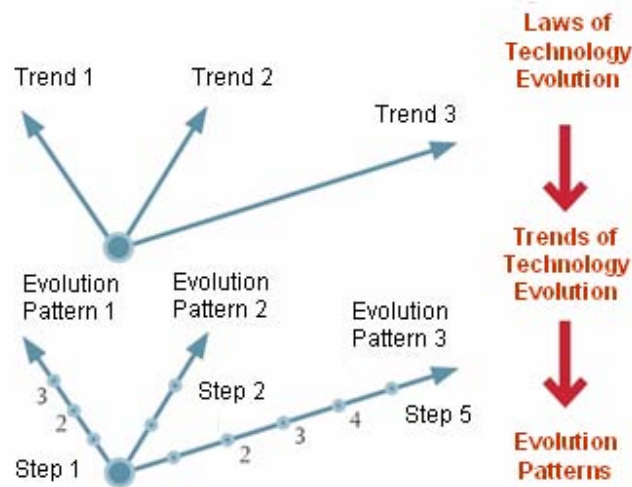
Fig. 2. Conventional design a) evolution trends, b) evolution patterns.

The evolution pattern is used in TRIZ as a tool for obtaining a technical solution by analogy with other technical systems. In addition, the organizing of information in evolution patterns looks very promising for describing the evolution of technical systems and elements thereof.

Let us imagine the hierarchy of notions in TRIZ (Fig.2):

- *Laws of technical system evolution*, which describe in general relations between phenomena. The laws are difficult to use as tools for problem solving due to their generality; therefore, the action of the laws is described through system evolution trends and patterns.
- *Trends* show general directions of the system elements evolution in accordance with the objective laws of technical systems evolution. A trend may be graphically presented as a vector.
- *Evolution pattern* is a specification of some evolution trend of a given object or process. It is not just an evolution direction, but a detailed "route" with the indication of the characteristic transformation versions of a technical system or its element.

Today, numerous evolution patterns are known in TRIZ. They are the result of study of real technical systems. For example, the "Mono-bi-poly" pattern revealed by H. Altshuller by studying and interpreting the law of transfer of a technical system to a super-system (Fig. 3).
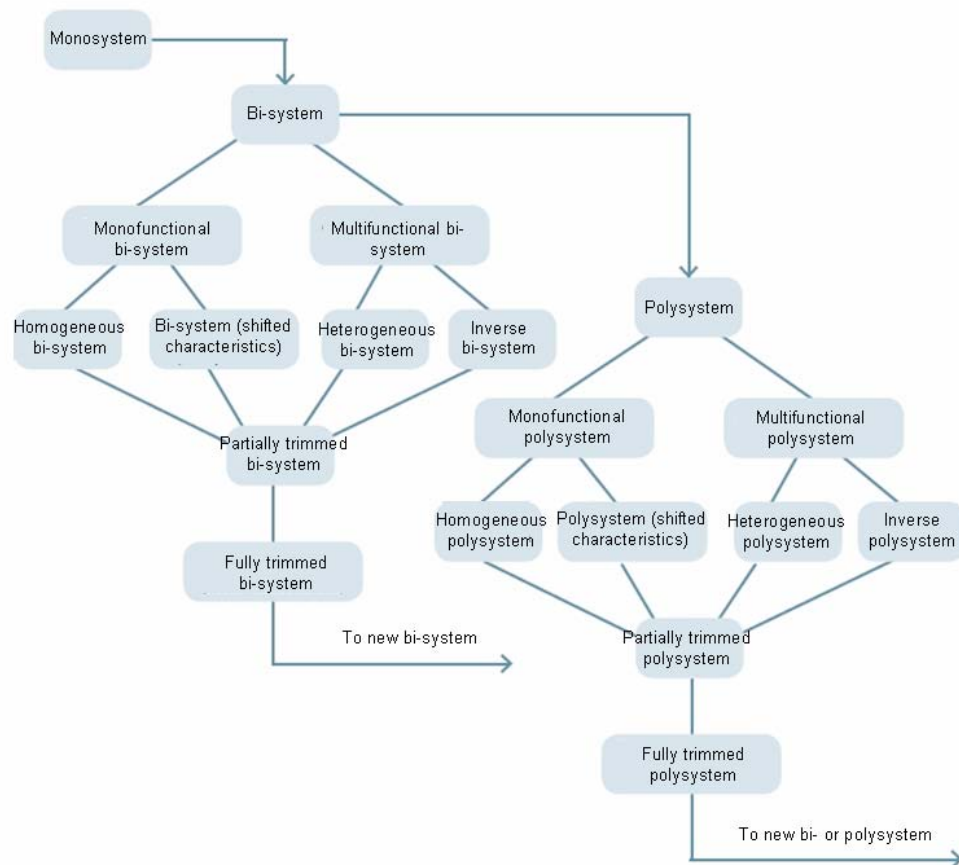
5

Fig. 3. "Mono-bi-poly" pattern according to H.Altshuller.

The interpretation of this evolution pattern is as follows. During its evolution, a system depletes its resource and combines with another system, thereby forming a more complicated structure – a bi-system. Several initially independent systems may unite to form a polysystem. The basic condition for transfer to bi- and polysystems is the necessity to improve the operational characteristics of the initial system and to introduce new functions that can be performed by a combined system.

Bi- and polysystems may be monofunctional and polyfunctional. Monofunctional bi- and polysystems consist of similar or different systems which can perform the same functions. Multilfunctional structures may comprise heterogeneous systems which perform different functions and, as a particular case, inverse systems, performing contrary functions. Generally, after a system has combined with other systems, all elements of the produced polysystem start combining into a higher level monosystem.

Such well-developed strong prognostic patterns as "Mono-bi-poly" are very effective for the analysis of the technical system evolution. Many other interesting evolution patterns revealed by B. Zlotin, Yu. Salamatov, V. Petrov and other researchers may be given as an example, but the following should be mentioned. Evolution patterns are usually given as sequential sets of versions of technical objects or technological operations which may be used to get a prompt for problem solving. To this end, it is necessary to analyze the evolution pattern of some known similar technical system and to transform the system of interest by analogy with this system. Using the evolution patterns in TRIZ may be made even more effective by developing effective mechanisms.

In addition, to correctly describe information field, single patterns, even if they are based on objective laws, are not enough. The problem consists in that evolution of some object or system is not described by one pattern

6

only. Normally several patterns may be traced in the evolution of a real system. There arises a question how to separate one pattern from another.

The book demonstrates this idea by giving examples of the evolution of two systems, a simple and a complex one: a ruler and an aircraft wing.

Using the evolution patterns for getting a prompt, searching for solutions to real problems by analogy with already known technical solutions will not caused any difficulties. But if our task is getting a clear information structure necessary for making a high-quality system evolution forecast, it is necessary to have more strict definitions of every evolution pattern, to separate one pattern from another and to select the one that describes transformation of the technical system elements in a most adequate manner.

To achieve this, it is necessary:

- to determine the technical system model corresponding to the existing system and to make a list of the main elements of this model;
- to understand how the system described by this model evolves and to determine the character and sequence of actions which change its elements when transferring from one system version to another.

- **Model of a functioning system**

To analyze a system, we should build a model of the technical system, i.e., to identify its parts and relations between these parts. Which model will be the most adequate to a real technical system?

To perfect machinery, one has to deal with the notion of a "technical system" illustrated by the model suggested by H. Altshuller. The model includes the following parts: an engine, a drive, a tool and a control system (Fig. 4). Although the notion "technical system" is described by one model, it has many definitions, and even more interpretations of these definitions. The inconsistency of this notion is in that, on the one hand, TRIZ treats the technical system as a "technical object" and, on the other hand, the structure of the technical system is clearly defined and should include all the four above-mentioned components. For most technical objects, this is impossible just like it is impossible to find an engine in a hammer.

It is necessary to develop a more logical, self-consistent model of a system. It should be understood that for correct comparison of different technical systems, these should be considered in similar conditions which characterize a system in the fullest and most objective manner. Some information about the system may be obtained by analyzing it at each lifecycle stage. However, those features of the system are important to us which manifest themselves during operation, when the system performs its useful function, manufactures a product for the production of which it was created. The most important moment for everyone participating in the creation of this system takes place on the proving ground. Only the system operation and operation results allow a trustworthy identification of the composition and structure of the system, peculiarities of interaction between the system elements and provides information necessary for further perfection of the system. Hence, it is the model of an *operating system* that we have to build.
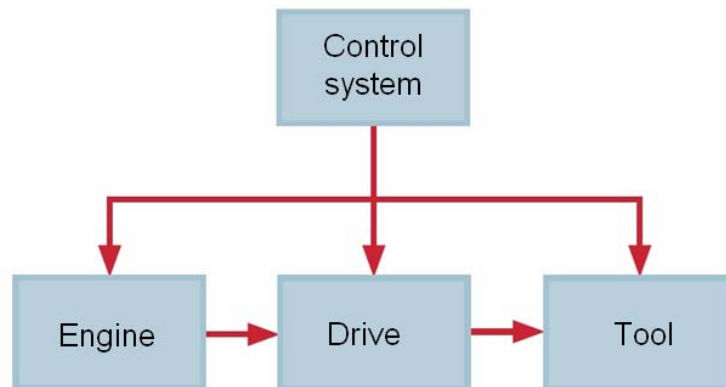
Fig. 4. H.Altshuller's model of the technical system.

Thus, the term "technical system" is more suitable for denoting an organized set of technical objects. As to an operating system, let us introduce the notion of a "functioning technical system".

The functioning technical system is a system that combines all elements which are needed to fulfill a required function and that is considered and analyzed during its operation.

The composition and structure of a functioning technical system are shown in Fig. 5.
The functioning system described by this model exists at two levels: an *information* level and an *object* level, which interact through an operator or controls.
At the information level, an operator or control should be informed about a sequence of performing a process aimed at the realization of the system's function. In addition, the operator needs knowledge and skills of controlling the object part of the system and in case a special control organ is used, it is necessary to specify the algorithm of its work.
At the object level, the model of a functioning technical system may comprise the same parts as those included in the technical system (see Fig. 4): an engine, a drive, and tool acting on the object being worked. However, the functioning system may use any technical object, for example, a hammer in a nailing system. To provide a functioning system, natural objects may be used, for example, a stone may be used instead of a hammer in a nailing system. Sometimes, a functioning system may only include an operator. An electrician replacing a burn-out light bulb may serve as an example. Here, the man performs not only the function of an engine and a drive, but his hand also serves as a tool.

It would be helpful to introduce an object to be processed into a functioning system and to consider its dynamics: starting from a blank, then at different levels of processing and, finally, the finished product. There is one more significant element of a functioning system – substances, fields and additional systems necessary for the functioning of an operator or a machine. For a car, these may be fuel, lubricants, air, heat and many others, even a road. For a biro, these are ink, heat, force of gravity. For a copying machine, these are electricity, powder, documents being copied. An operator needs air, heat, food and water, clothing, footwear, goggles, etc. In addition, when considering the composition of the functioning system, the products of processing the supplied substances, fields and systems should be taken into account as well as the possibility of their use or removal, i.e., exhaust gases of the car should be removed from the driver and the heat produced during the engine operation may be used to heat the cabin.
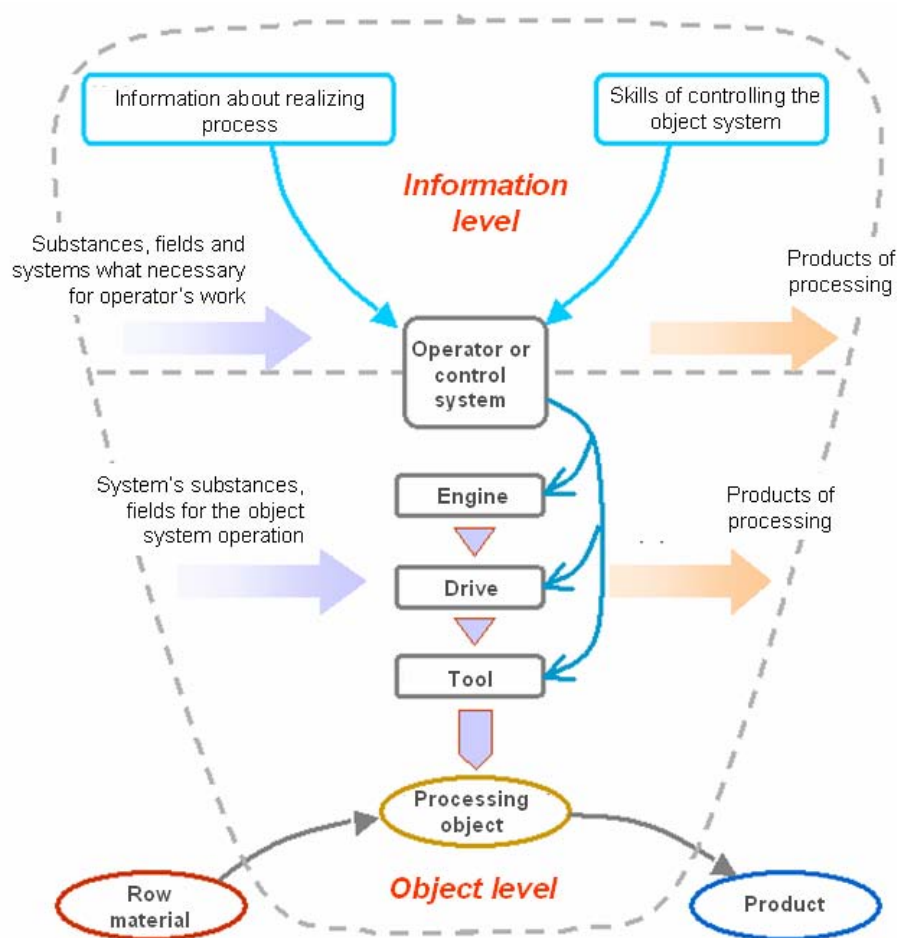
8

Fig. 5. Model of a functioning technical system

It is necessary to distinguish between a functioning system and a system of technical objects, a machine. The functioning system performs a certain function at a given moment. The machine is originally designed for performing some main function, the designation of this function being subjective to a certain extent, i.e., the machine may be part of a system that performs the same function as the main function of the machine, but not necessarily. For example, a motor vehicle is designed and manufactured for carriage of loads and passengers. If it is used in a functioning system having the same designation, the tool of the car will be the truck body or the car cabin and the object being processed will be cargo or passengers. If someone wants to use a motor vehicle for compacting garden paths, the functioning system will be quite different. Its function will be "soil compaction", its tool will be the vehicle's wheels and the object being processed will be a garden path. The motor vehicle may also be used for other purposes: in blockbusters, the car's bumper (tool) is used to smash down gates (object being processed). And, finally, a driver reposing in the shade of the truck will be an object being processed and the truck body will be a shadow-producing tool. All functioning systems will be different in this case, though they include the same system of technical objects – a motor vehicle.

Because we are dealing with technical and patent information, it is important that the concept of the functioning system be matched with the models used in the patent legislation. The comparison proved that the composition and structure of the proposed model is well-matched with the list of patented objects specified in the patent legislation. Because the patent legislation is based on one of the most well-developed information organization methods, we may conclude that the functioning system model is correct.

9

Accordingly, we may assume that the most adequate and effective will be those evolution patterns which describe transformations of the main elements of the functioning system model.

They are:
- sets of material objects,
- description of their interaction process,
- substances, fields and systems required for work of the object part of a system and an operator,
- substances, fields and systems as products of processing,
- algorithms and programs for automatic control systems,
- training programs and courses.

- **Evolution of the functioning system**

Evolutions of biological and technical systems have much in common, but they also have one fundamental distinction. Biological systems have an inherent transformation mechanism; to trigger it, it is necessary to provide an initial impulse and maintain favorable evolution conditions. All the rest will be done by a plant or animal itself. In technical systems, each transfer from one version to another is only performed at the expense of the external interference of a man, a subject. In most cases, without this interference a system can only evolve towards the destruction of constructions, gradual loss of nuts and other small parts.

How does a functioning system evolve?
TRIZ has discovered the laws of technical systems evolution which are conventionally united in three groups – Static, Kinematics, and Dynamics (Table 1).

Table 1

1.	The law of system completeness	
2.	The law of "energy conductivity" in a system	Static
3.	The law harmonization	
4.	The law of increasing ideality	
5.	The law of irregularity of system's parts evolution	Kinematics
6.	The law of transition to a super-system	
7.	The law of transition to micro-level	
8.	The law of increasing the degree of substance-field interactions	Dynamics
9.	The law of dynamics growth, controllability and displacing a man	

Some researchers, first of all H. Altshuller, think that the three laws of the "Statics" group (Table 1) are the *criteria or conditions* required for the occurrence and existence of any technical system. In this context, we may say that a functioning system may be in two qualitatively different states:
- at the stage of creation or transformation (in a "workshop") – at this stage, laws 1-3 of the Table 2.1 are in force;
- at the stage of performing its function and evaluation of performance parameters (on a "testing ground"). Here, we can trace the action of laws 4-9.

These two stages are fundamentally different and have considerable distinctions of kind. When a system is disassembled and reworked in a "workshop", some operations necessary for preparing it for work are performed on this system. This may be accompanied by a change in its composition (set of elements of which it is composed) and by a change in its structure (arrangement of these elements and character of relations between them). Thus, at the creation and transformation stage, we may only speak about some set of parts of a functioning system, but the system itself only appears after performing all work aimed at its completion. On the test ground, a fully completed functioning system operates and its parameters only change within negligible limits.
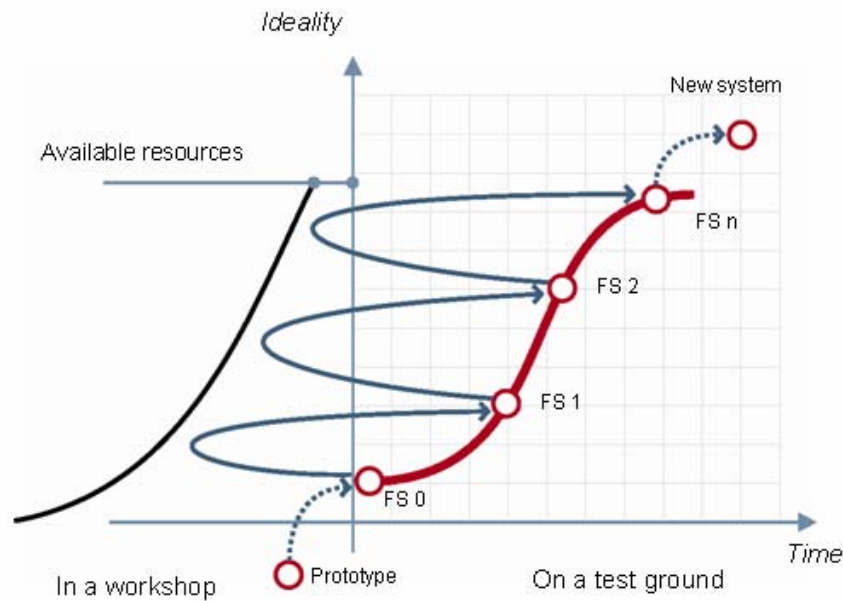
10

Fig. 6. Stepwise evolution of the system.

The system transformation results in the coordination of the parameters and operation modes of all its elements, first of all, the tool and the object being processed. Only after such coordination, the system may be considered completely ready for transition to a new qualitative state, from just a "set of elements" to a "functioning technical system". Now it is possible to test the new version of a fully prepared and completed system and to evaluate its performance characteristics. The designers' efforts are focused on the process of the creating of new versions of the technical system, and the system evolution illustrated by S-curve is but a visible result of this process. Thus, the continuous system evolution process is realized step by step, through a sequence of discrete structure transformations (Fig. 6). As the system evolves, the resources for its improvement are gradually depleted, evolution decelerates and the system inevitably transfers to a new level and is replaced with a fundamentally new version.

It is obvious that transforming a functioning system requires performing some operations on its elements. The best known lists of operations performed on the elements of a technical system during its transformation are principles of resolving technical contradictions as well as the set of standard solutions to inventive problems by H. Altshuller and the "Interindustrial fund of heuristic principles of object transformation" compiled by A.I. Polovinkin. Many of these operations with slight changes double each other but, nevertheless, about 150 significantly differing operations may be selected for changing the elements of a technical system. To use such an ample data base, it is necessary either to reduce it strongly, or to structure it trying to single out several basic transformations and using the rest of them as auxiliary ones. Let us see which of the element-transforming operations are basic, obligatory and which of them may be considered auxiliary ones. To answer this question, let us examine in more detail the process of transition of a system from one of its version to another.

- **Three-stage system transformation algorithm**

As proved by H. Altshuller, each time you transform a system into a new, improved version, it is necessary to fulfill three conditions required for the existence of a complete, absolutely operable functioning system:

1. *providing the "correspondence of the system's composition to the function it performs",*
2. *establishing of links between the system elements,*

3. *coordination the parameters and operation modes of subsystems of a particular system.*

At each of these stages, there occurs accumulation and organization of resources required for the realization of the next stage. As the system existence conditions are satisfied, the *coordination* of the parameters and operation modes of the system's interacting parts also increases. This eventually makes possible a more complete coordination of the tool and the object being processed, as well as the coordination of the system and the environment.

Satisfying these three existence conditions of the functioning system may be presented in the form of a three-stage algorithm (Fig. 7).
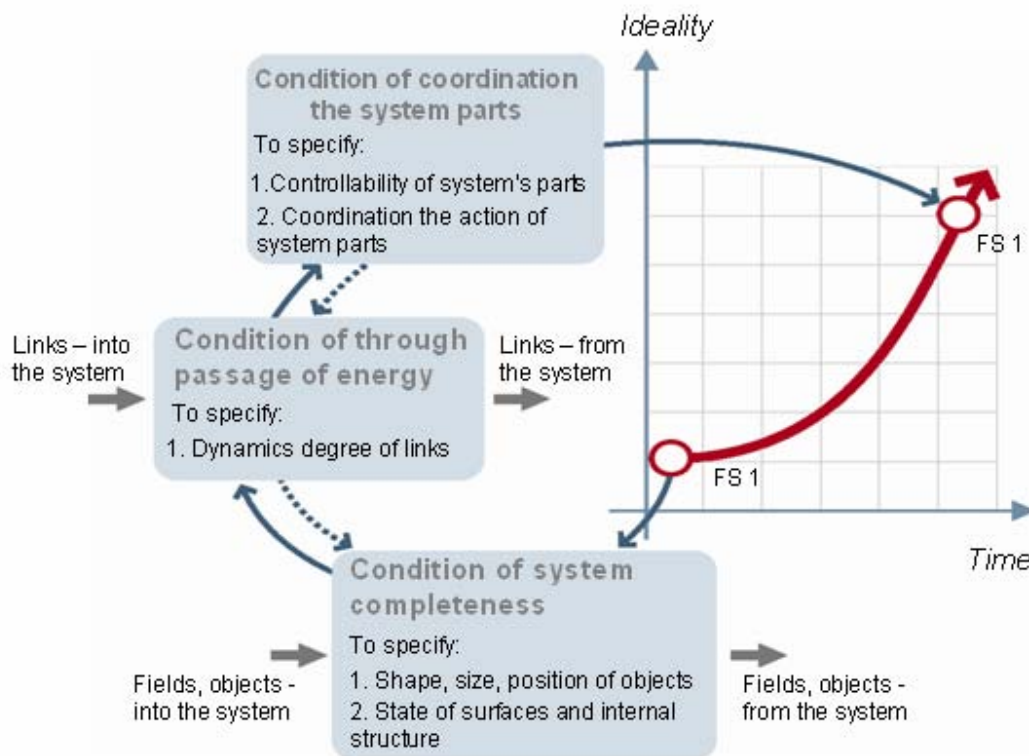

Fig. 7. A three-stage system transformation algorithm

Stage 1.

At this stage, the main existence condition of the system is satisfied – *the system completeness in accordance with the function performed.* When designing a new version of the system, we may introduce some new elements into it - objects, fields, processes, or remove some of the existing elements from the system. These operations may be performed simultaneously, by replacing one object with another. A special case of producing a new object is separating an original object into several parts – segmentation.

At the fist stage of work, the system gets most substance and field resources. After coordination the amount and type of added elements, it is necessary to additionally match their shape, size and mutual position; then to specify the properties of the materials from which they are manufactured, as well as the properties of their internal structure and the surface parameters of the system's elements. Position, size and shape of spread as well as parameters should be matched for fields.

Hence within the first stage, we may speak about the following operations:
- introduction of elements into a system,
- removal of elements from a system,

12

- segmentation of system's elements,
- changing the shape and size of elements,
- complicating the internal structure of elements,
- changing the surface state of elements.

Stage 2.

The next condition to be satisfied is *through passage of energy between the functioning system elements.* This condition is satisfied by organizing internal links within the system. Operations performed to establish links are similar to those for providing the system completeness: we may introduce or remove one or several links or replace one link with another.

After the links have been introduced into necessary locations within the system, their character should be specified. Links may be both rigid and flexible, with different degrees of freedom, for example, different types of hinge joints. Material objects may be connected by fields. In the first instance, it is necessary to provide the main direction of energy passage in the system: from an engine through a drive to a tool. Also, it is important to establish rigid and flexible links between the system's parts and the control. The establishing of rigid and flexible links between the system parts results in a kind of logically connected frame, the system base where flexible, dynamized elements of the system may be fixed by means of flexible links.

Within the second stage, the following actions may be performed:
- introduction of linking elements into the a system,
- removal of linking elements from a system,
- providing the link mobility,
- providing the change of field parameters.

Stage 3.

The system is checked for coordination of its functioning with the performance conditions of the main operation – the action of the tool on the object being processed. Because the composition, structure and character of interaction of the system's elements have already been determined in general, the check optimizes the system operation and specifies the parameters of its elements.

Here, it is necessary to check the system's elements for adequate coordination of system elements' parameters while meeting the first two conditions. Final coordination of the interaction of the system's elements in accordance with the technological process of its operation should be performed, if necessary. Most system's elements have been matched at the previous stages and have rigidly set parameters which provide the system operation. The parameters of some elements should be coordinated during operation by regularly eliminating their deviations from the optimum values. Part of the system's elements requires *operational control* – permanent change in accordance with the changing operation conditions.

When a system has a high degree of initial coordination, it is comparatively easy to control and almost does not require any real-time control. For example, coaster cars are well matched with the motion conditions and only require acceleration at the beginning and stopping at the end of the path. If coordination is mostly achieved through real-time control, the operator's job is not easy. An example of minimal coordination is a Buggy-Rollin device for skating from asphalt, which consists of multiple rollers attached directly to a human body: knees, feet, hands, elbows, etc. Rollers are not connected with each other and are only combined into a system by the efforts of roller-skaters. Buggy-Rollin skating is not easy because one has not only to follow the turns of the path, but also to hold each roller in a required position.

Checking the system's elements for the degree of coordination and finally coordination all the parameters and actions of its elements are the final design stage at which the following transformations may be performed:
- providing an operational control,
- introducing a new operation into a process,
- removal of operations from a process,
- segmentation of an operation into smaller ones,

- combining of several operations.

Thus, the degree of coordination of parameters and operation modes of the system's elements with the process it performs is increased as the three-step algorithm is performed:

- *Initial coordination* of the system's parameters with some averaged operation conditions carried out at the design stage. This coordination may be static – a car is stable both when it stands and when it moves. The example of dynamic coordination is a bicycle; it is only stable in motion.
- *Periodic coordination* which is performed from time to time during system operation by means of compensating adjustments.
- *Final coordination* of the system's parameter with the changing operation conditions, which needs a possibility of operational control, i.e., fast and comparatively simple change of the system's parameters with a change in its functioning parameters.

How can we get a list of the main evolution patterns?

It is necessary to combine all of them in a single list of operations performed at each transformation stage of the elements of the system's object part. As a result, we obtained a list of ten main operations (left column of Table 2).

Then, having collected a data base of technical system evolution patterns known from TRIZ, we selected those ones, which could be obtained as a result of each operation. For example, performing the "Segmentation" operation on some element several times on end will result in a set of versions of this element arranged along the "Segmentation of objects or substances" pattern line.

The main evolution patterns are given in the right column of Table 2.

Table 2

	Operations performed for system transformation	*Corresponding evolution patterns*
1.	*Introduction of elements and links into a system*	*Transfer from a monosystem to bi- and polysystem (Mono-bi-poly)*
2.	*Removal of elements and links from a system*	*Trimming a system*
3.	*Replacing of one elements and links with other ones*	*Expansion-trimming of a system*
4.	*Segmentation of a system*	*Segmentation of objects and substances*
5.	*Changing the shape and size of system's elements*	*Geometrical evolution of objects*
6.	*Changing the internal structure of a system*	*Evolution of the object structure*
7.	*Changing the surface state of the system's elements*	*Evolution of the object surface mircorelief*
8.	*Providing the link mobility between the elements and possibility of changing other parameters of the system*	*Dynamization*
9.	*Providing and facilitating of operational control*	*Increasing the controllability of the system's elements*

10.	Verification and improvement of the coordination of functioning of the system's elements	Increasing the coordination of functioning of the system's elements

Thus, constructing a correct model of a system and analyzing the order of actions aimed at its transformation we obtained a list of evolution patterns which describe, in the most adequate manner, transformations of a technical system and its elements.

Going back to the functioning system model (See Fig.5) will allow seeing that the produced evolution patterns first of all describe evolution of the material elements of the system – substances, fields, and forces. Evolution patterns for processes are beyond the scope of this book.

Let us consider the main evolution patterns of the system's elements (*detailed description illustrated by examples is given in the book):*

"Mono-bi-poly transfer"

Transition from a monosystem to bi- and polysystems which is usually referred to as "Mono-bi-poly" demonstrates expansion of an object by means of combining with similar objects. The initial "Mono-bi-poly" version is some single object or system.
>   The next steps may be:
> - introduction of one additional object into a system;
> - introduction of several additional objects,
> - transition to a monosystem of a higher level.

Trimming

Trimming a system means combining the system's functions in some element in order to remove the freed elements. From the technological evolution viewpoint, trimming is treated more generally, as simplification of the system. The initial version of the "Trimming" pattern is some original structure of a technical system to be simplified by removing some elements.
>   The pattern may include the following steps:
> - removing one object from a system,
> - removing several objects from a system,
> - transition to a maximally trimmed system,
> - using an ideal system.

Expanding-trimming a system

This pattern occurs as a result of a sequential execution of two operations described in relation to the previous patterns: at the initial stage, new elements are introduced into a system, thereby expanding the system (the "Mono-bi-poly" pattern) and at the last stage, elements are removed from the system (the "Trimming" pattern). The "Expansion-Trimming" pattern starts with some single object of a system and may include the following steps:

	• Forming a function nucleus of a system and providing the minimum operability of a system
Expansion	• Introducing one additional object into a system

	• Introducing several additional objects
	• Forming a complete system and providing its satisfactory operability
	• Removing one object from a system
	• Removing several objects from a system
Trimming	• Transition to a maximally trimmed system
	• Transition to an ideal object

Segmentation of objects and substances

This pattern is a series of object transformations at the expense of sequential segmentation of the object into increasingly small parts, down to atoms and molecules, with transition to the level of fields and vacuum. It starts from a single one-piece object and may include the following steps:

- o transition to an object segmented into two parts,
- o transition to an object segmented into many parts,
- o transition to granules,
- o transition to powder.
- Segmenting an object to the molecular level:
  - o transition to a paste-like substance,
  - o transition to a fluid,
  - o transition to foam,
  - o transition to fog,
  - o transition to gas.

- Segmenting an object to the level of parts of atom;
  - o transition to plasma.
- Segmenting an object to the field level
  - o transition to field interaction.
- Segmenting an object to vacuum:
  - o transition to vacuum
- Using an ideal system

Evolution of surface properties

This pattern is a series of objects having different states of surface. It starts from a smooth surface object and may include the following steps:
- formation of projections and cavities,
- formation of fine profile surface,
- use of a surface having special properties.

Evolution of internal structure

It demonstrates versions of the object's inner space state. It starts from the object version having a continuous inner structure and may include the following steps:
- introduction of void,
- formation of several volumes,
- separation of space into multiple volumes,
- introduction of fields and forces.

Geometrical evolution

Geometrical evolution is a series of objects of different geometrical shape. It starts from a single geometrical element (point) and may include the following steps:

	• Point
Transition from point to line	• straight line • line curved in one direction • line curved in two directions • complex line
Transition from line to surface	• flat surface • cylindrical surface • spheroidal surface • combined surface
Transition from surface to volume	• prism • cylinder • sphere • complex 3D structure

Dynamization

Dynamization is a series of objects arranged in the order of increasing mobility, dynamicity, primarily at the expense of changing their basic parameters. The first step of this pattern corresponds the system version where system's parts are rigidly connected to each other. The pattern may include the following steps:
- transition to a system which is movable in one direction,
- increasing the degree of freedom of the system's elements,
- transition to flexible links,
- transition to a system having field-linked parts,
- transition to a system with separated parts.

Increasing of controllability

This pattern demonstrates simplification of the interaction between an operator or a control device and the object part of the system. The pattern starts from the system's version where its elements only have preliminary coordination, but do not have real-time control by an operator or a control program. The pattern may include the following steps:
- manual control,
- transition to semi-automatic control,
- transition to automatic control.

Increasing of coordination

The coordination of all the parameters, characteristics and operations of the system's elements with the performance peculiarities of its main function – interaction with an object being processed – is the object of all

the designer's operations. Coordination is verified at all system transformation stages. The system coordination possibilities are better seen when the system has already been designed and is ready for work.

The system should be checked for sufficient coordination by different parameters:

- coordination the *functions* performed by the system's elements with the main useful function of the system;
- coordination the *composition and structure* of the system is aimed at leaving only the system's elements necessary for the performance of its function and at organizing them in an optimal structure;
- coordination of *different parameters of the system's parts* - shape, size, surface state and internal structure of the interacting elements of the system - both with each other and with the environment;
- coordination the *rhythmic,* operation sequence of the system's parts allows tuning the system operation to a single rhythm, amplify the action of the system's parts due to the operation in the resonance mode;
- coordination the *materials* used for the manufacture of the system's parts with the complexity of these parts allows choosing the optimal manufacturing and functioning technology of the system.

The initial version of the "Increasing of coordination" pattern corresponds to a system with non-matched or poorly matched elements. The parameters of the system's parts are matched mostly at the expense of preliminary and regular coordination while the final coordination is absent. The pattern may include the following steps:
- transition to a stepwise change of parameters,
- transition to a gradual change of parameters,
- recuperation (repeated use) of energy.

- **Evolution pattern construction rules**

When constructing and using the evolution patterns, it is important to obey the following rules:

1. the rule of the unity of the object's property being transformed and the transformation type,
2. the rule of completeness of action and transformation hierarchy,
3. the rule of checking for coordination,
4. the rule of optimal generalization of information.

When *obeying the first rule,* it is necessary to take into account, that a correctly constructed pattern describes the results of sequential use of one operation only. This operation is aimed at changing only one property of the object being transformed.

> *If it is necessary to transform a surface, a pattern should describe sequential change of one of the surface properties, for example, shape. Another pattern may describe evolution of surface properties or dynamic properties of the surface, etc.*

According to *the second rule*, when transforming a system, all operations with its elements are performed in a certain order. Each subsequent transforming action may only be performed after creating conditions and providing resources by means of performing the previous action. Hence, there exists a certain hierarchy of actions performed during the system transformation, as well as the hierarchy of evolution patterns formed as a result of each of these actions. This is how we understand this hierarchy (Fig. 8):

1. introduction of new or segmentation of existing objects, processes, fields and forces,
2. coordination the shape, size, and properties of the surfaces with the internal structure of the system's elements, process parameters, fields, and forces,
3. dynamization of sets of objects, processes, fields, and forces,
4. providing controllability of the system's elements,
5. and as a result, coordination the action of the system's elements.

At every transformation of the system, it is necessary to execute the entire sequence of actions provided by the three-stage algorithm (see Fig. 7). The set of actions should result in the production of a new version of the functioning system.
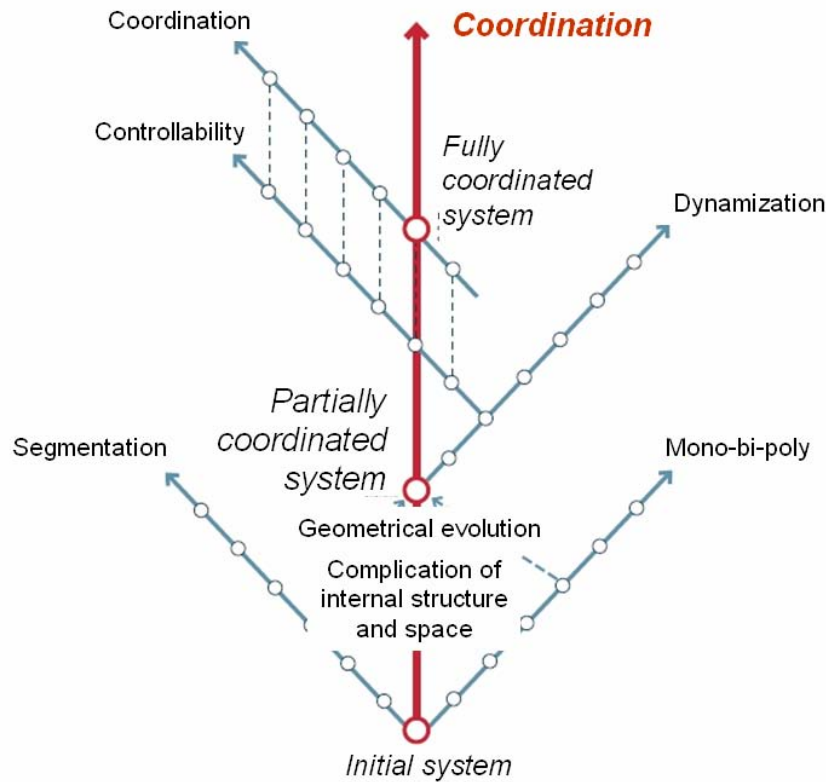

Fig. 8. Hierarchy of actions aimed at the transformation of the system's elements.

The *third rule* follows from the second one.

After each transformation cycle, the system is checked for coordination. This is performed in the order that is reverse to the execution order of operations in accordance with rule two, i.e., to match the operations of the system's elements, these elements should be controllable. For the elements to be controllable, it is necessary to have a possibility to change the parameters of these elements during operation; hence, they should be dynamized. It is very difficult to dynamize a rigid monolith object or a process with accurately preset parameters. Consequently, it is necessary to supplement the system with some objects (or processes) having different parameters or to segment the available objects (or processes) and to *match the parameters of the obtained parts* with each other.

> For example, to provide a controllable motion of an airplane, it is necessary to modify the wing shape. The wing shape should be controlled from the pilot's cabin. To provide controllability, it is necessary that the wing be dynamized, i.e., the entire wing or its parts be movable. Dynamization resources may be obtained by separating the wing rear part from the wing body. Also, additional rudders may be added to the rear edge of the wing, their shape may be matched and links, for example, hinge mechanisms be introduced.

The forth rule determines the information *generalization* degree. Depending on the analysis goal, it is necessary to find the optimal degree of difference between the transformations in each particular case, because excessive specification increases the number of insignificantly differing versions and impedes the analysis while excessive generalization is not instrumental. It is important that transformation have *qualitative* difference.

Conclusion: Chapter 2 determines ten basic evolution patterns of technical elements on the basis on the analysis of a functioning system and its evolution peculiarities.

Content of Chapter 3: "Constructing an Evolution Tree"

- **Elementary function: how can it be described?**

The model of a system for performing a required function comprises: a tool (element that performs an action), object being processed (system's element subjected to the action) and interaction between these two elements. This is the minimal set of acting element; their number may be increased.

Fig. 9. Function performance

To describe the performance of an elementary function, a morphological box composed of two axes may be used. One axis hosts tool versions while the other one shows the versions of the object being worked (Fig. 10) The cells of the morphological box may be filled with all possible versions of interacting elements by using the evolution pattern where specific embodiment variants of these elements are located.

There occurs a question: What is the best way of arranging the tool evolution patterns on the morphological box axis?
The morphological box shown in Fig. 10 would satisfy the problem solution if all the versions of the elements were described by one evolution pattern only. However, several patterns are needed. Therefore, it is necessary to determine their rational arrangement on the morphological box axes.
Evolution patterns may be arranged on the morphological box axes sequentially, one after another, or may start from the same point (Fig. 11). However, as shown above, evolution patterns demonstrate the results of actions aimed at system transformation which are performed in a certain sequence: each consecutive action is performed with account of the results of the previous one. This determines a certain pattern arrangement hierarchy (see Fig. 8) according to which a new pattern may start from any point, any transformation version. Therefore, arrangement of patterns in the form of a tree-like structure conveys more adequately the essence of actions aimed at the transformation of a system as a whole and its elements.
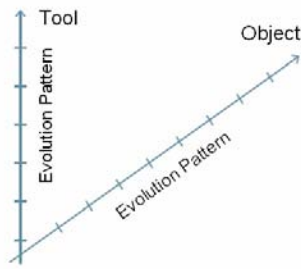
Fig. 10
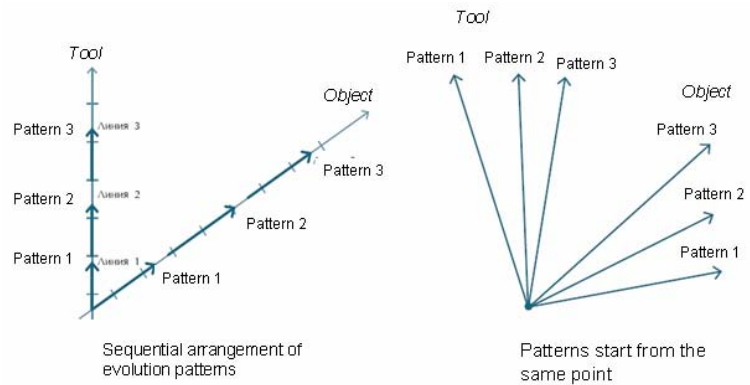A simple morphological box for the tool and the object.

Fig. 11. Arrangement of evolution patterns on the morphological box axes

We call the structure, where the system evolution is observed through a change of some of its parameters, the *Technological Evolution Tree.*

The structure of a real Evolution Tree is given in Fig. 12. Each branch of the Tree is an evolution pattern of one of the elements of a system of interest and corresponds to the objective evolution trends. There is always the main, principle pattern which starts directly from the initial version of a technical object. The object versions arranged at each point of the main pattern may serve as initial ones for the branch patterns of the second order. Each consecutive level of the Tree hierarchy is represented by patterns built on the previous branch patterns. A set of thus arranged patterns forms the simplest structure of the Tree. In the constructing of real Evolution Trees, it is necessary to observe certain sequence of actions and take into account a series of rules considered below.
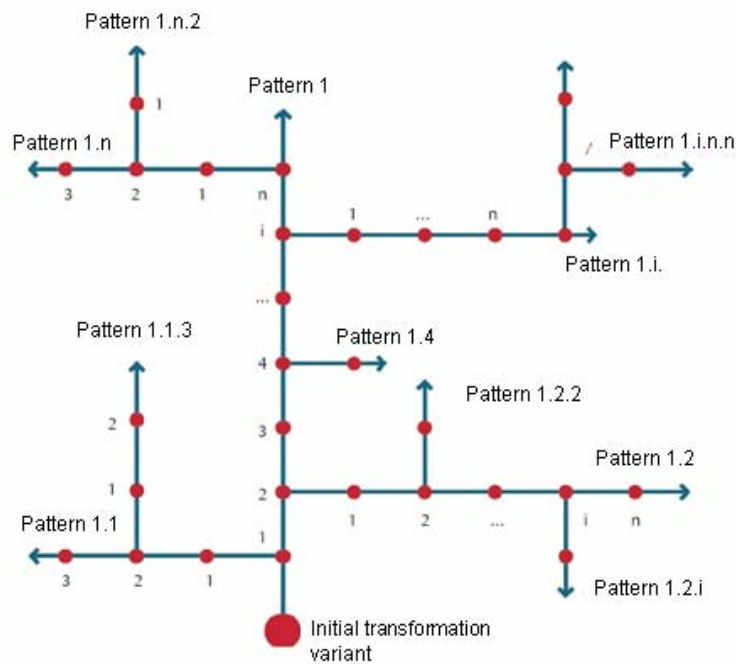

Fig. 12. The Evolution Tree structure

In principle, there are no limitations, either concerning the number of the hierarchy levels, or concerning the number of patterns used for building a pattern tree, i.e., the Evolution Tree of any technical object may be continued infinitely both broadwise, according to the number of the evolution patterns of one hierarchic level,

21

and depthward, according to the number of these levels. It may be said that the Evolution Tree has some properties of infinite fractal structure. Fractal is a self-similar set, i.e., a set having the same structure in large as it has in small. The theory of fractals is well-elaborated, which creates objective premises for mathematicization of information processing using the Evolution Tree.

Using a tree-like structure helps visualize possible transformation versions of the system's elements and considerably simplifies navigation of the diversity of these versions. In this case, each axis of our morphological box for the tool and the object will be not a set of several, often random transformation versions of the elements, but an Evolution Tree where all basic embodiment versions are arranged in a certain order.
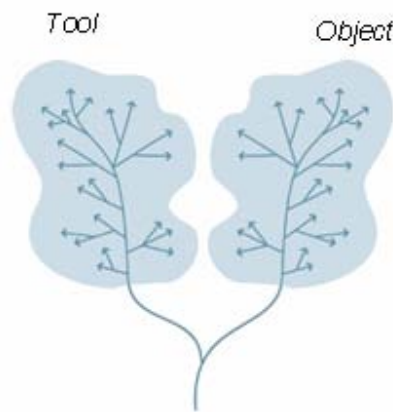

Fig. 13. A morphological box having axes in the form of tree-like structures

Considering the "tool-object" pair by consecutively selecting each component of the pair on the Evolution Tree of these elements allows reckoning all main methods for performing a function of interest. This provides a quality analysis of information for obtaining prognostic technical solutions.


- **Basic and specific Evolution Tree**

Each evolution pattern exists as if at two levels – at the abstract level where sequences of generalized transitions making the sense of a transformation pattern and similar for multiple evolving technical objects, and at the level where description is given for specific technical objects with the explanation of these abstract transformations.
According to this division of the pattern into an abstract level and a specific level, we can construct an Evolution tree at two levels. An Evolution Tree having branches in the form of abstract descriptions of evolution patterns will be called basic, and those constructed for a real technical object will be referred to as specific.

The Evolution Tree for a specific object will be unique due to the peculiarities of the problem being solved, availability of information, research task definiteness, etc.

- *By way of illustration, the book gives a detailed description of the display Evolution Tree, which demonstrates the technological evolution of the display starting from a simple optical element through cinematography, cathode-ray tube TV set, flat display to an ideal display which is a device that induces image directly in the user's brain.*

22
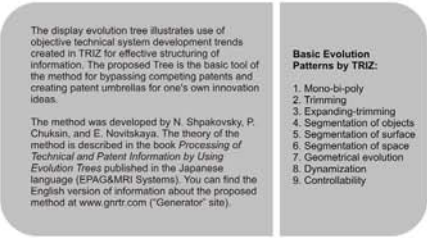
Fig. 14. Display Evolution Tree

Classifying the technical system transformations in the form of an Evolution tree substantially satisfies the requirements formulated in Chapter 1:

1. To organize information, we have selected a tree-like structure which provides visualization of the descriptions of *all basic* known versions of an object under study.

2. The Evolution Tree is an organized set of evolution patterns created by analyzing the evolution of numerous technical systems. Hence, constructing an Evolution Tree implies using an *objective classification criterion.*

3. Each evolution pattern includes a *set of generalized descriptions* of transformation versions and transitions between these; it may also be illustrated by an example of a specific technical object transformation. Thus, the *requirement of generality and specifity* is satisfied.

4. Information presentation in the form of a tree-like structure allows a designer to simultaneously see the basic transformation versions and clearly *trace their structure.*

5. The basic Evolution Tree provides all *essential transformations* even when information about the versions of a system under analysis is insufficient, fragmentary.

Conclusion: Chapter 3 proves that the Technology Evolution Tree satisfies all the requirements for the classification structure formulated in Chapter 1.

Content of Chapter 4. "Use of the Evolution Tree"

- **Search by marking the information field**

In traditional information search methods, an information field is homogeneous and the information units it contains are equivalent. It is only after a thorough study of the information field structure that the most promising zones with the most valuable information are determined. Using the basic Evolution Tree allows determining the valuable information concentration zones prior to search. These zones are located in the areas of the best known versions of a system and are specified with the help of versions suggested by the basic Evolution Tree.

In addition, such structuring allows determining the "entry points" – the most typical system transformation versions, best suited for starting a search. Search starts at these points, the "area of the known" gradually expands to new areas of the information field. Links are established between separate search zones, some of the zones join. A kind of logic frame of the information field is formed on the basis of already studied key zones.

When finding (or generating) a new information unit (a version of a system under investigation), we can easily determine its place in the Evolution Tree structure. This is often accompanied by the appearance of new "entry points", which allows the search direction to be corrected. This kind of search is called *structural* (Fig. 15)
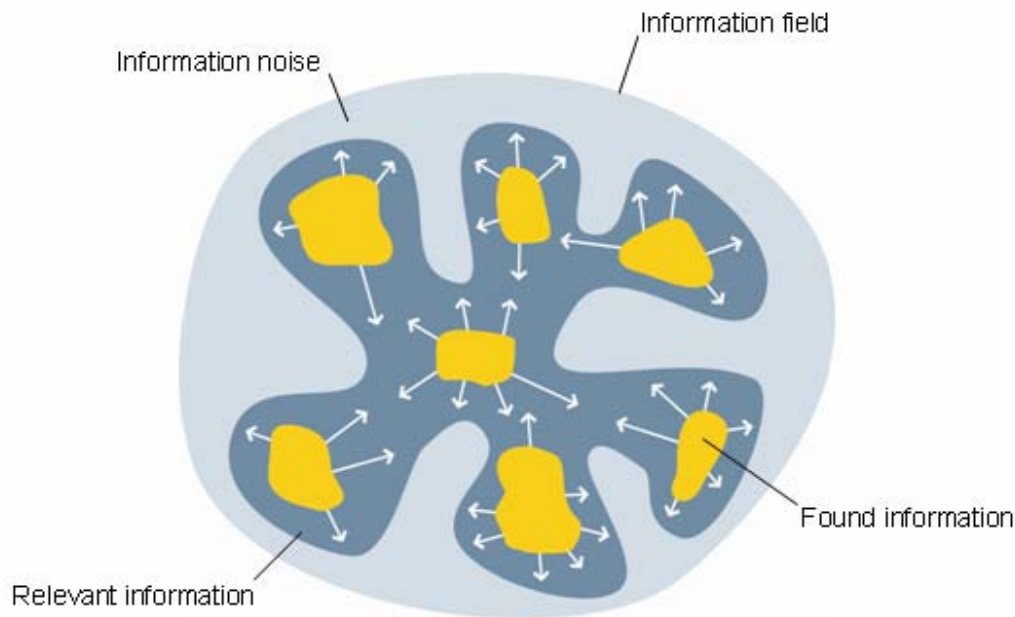This type of search may be called structural (Fig. 15).

Fig. 15. Information search with pre-marking of the information field. The arrows show the search direction from the information concentration centers

The Basic Evolution Tree allows a designer to get sets of key words required for search. Each set comprises two parts – an object name and a transformation name. A characteristic definition which describes the gist of a technical system transformation in the Basic Evolution Tree added to the name of the object may considerably simplify information search.

This query form corresponds to the set-up of patent search systems where it is necessary to indicate the object name and give definition of its important characteristic.

Description of the search for Bridgestone nanopowder displays is given as an instance.

- **Analysis of structured information**

Information structure is analyzed by performing the following operations:
  - ○ *Selecting or specifying a prototype*
  - ○ *Selecting a method for improving the prototype*
  - ○ *Preliminarily filling the empty places of the Tree and finishing the patterns*

Let us expand on the search of missed versions of a system. To analyze the Evolution Tree of a specific system for completeness, it is necessary to compare it with the Basic Evolution Tree. Two situations may arise in this case.

- *Some transformation versions are missed in a specific pattern (Fig. 16, a).* After the empty places have been revealed, it is necessary to repeat search. If no results have been obtained, it is quite probable that this is just the area where unknown, unpatented versions of the object under analysis are located. Such lacunas are the most convenient places for attacking a competing patent and searching for new prognostic solutions.

25

- *It is also important to reveal unfinished evolution patterns.* It often happens that the end portions of patterns are not filled (Fig. 16, b). This is just where the most promising transformation versions are located which could become a basis for new, more perfect technical solutions.
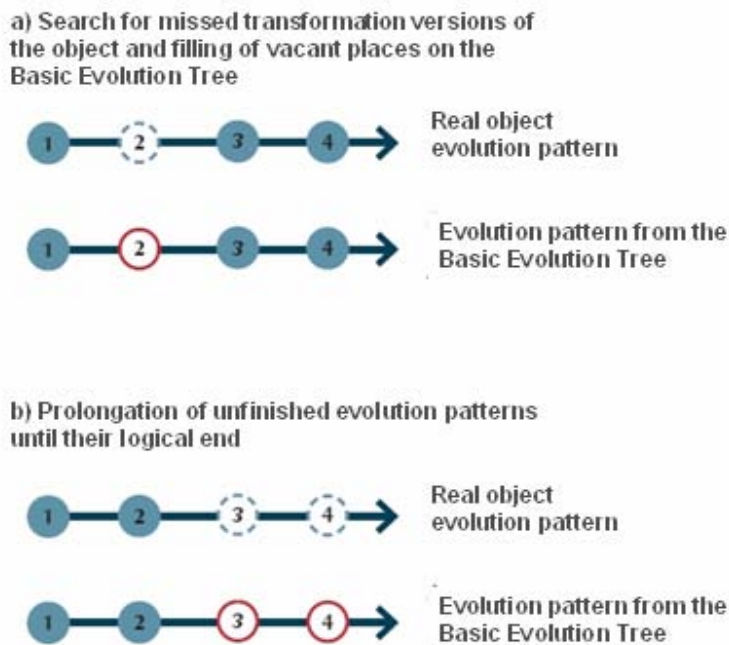

Fig. 16. Comparing the Basic Evolution Tree and the real object Evolution Tree

The peculiarity of our approach consists in that it offers an opportunity of getting a description of a missed variant built by analogy with a corresponding transformation version of the basic pattern. It may be said that comparing the real object Evolution Tree with the Basic Evolution Tree offers a kind of "prompt machine" or "concept generator" which allows producing "portraits" of missed versions of real system transformations. Having such portraits-descriptions, it is much easier to find way to real constructions while generating new technical solutions.

The conception is illustrated by the description of express-analysis of evolution patterns of an aircraft propeller (training example) and the Evolution Tree of the jet printer head (real example).

- **Production of new information**

Generation of new ideas and concepts is the key moment in the work with the Evolution Tree. At this stage, a new knowledge of a system under analysis is synthesized. All previous operations are analytical and preparatory for the production of new information. The goal is not only to obtain some new idea, but the obtained idea should be more effective and more applicable to the problem of interest than the prototype as well as more patentable.

In TRIZ, a system is evaluated by its degree of ideality. The concept of *Ideality* is interpreted as a ratio of a complex parameter which characterizes function performance by the system to the system operation cost:

I = F/C, where:

> **I** is the technical system ideality;
>
> **F** is the complex parameter characterizing the performance of the useful function;
>
> **C** is the cost of performing this function.

When improving a system in accordance with the above formula, there may occur three types of problem situations:
1. the number of the system's functions and the function performance quality are satisfactory, by the cost of performing these functions does not suit us;
2. the number of functions performed by the system does not suit us; it is necessary to provide performance of additional functions;
3. the number of functions performed by the system is satisfactory, but the function performance quality does not suit us.

The first type of problem situation occurs when the denominator value (cost) of the Ideality formula is unacceptably high. The second type of situation is due to a too low numerator (functionality). The third problem situation takes place when both parts of the Ideality formula are not satisfactory – the technical system productivity is insufficient and its operation cost is too high. Depending on the situation type, we can determine three ways of improving the system:

1. Trimming the system composition, removing one or several elements.
2. Expanding the system, when new systems, carriers of required functions, are introduced into its structure.
3. Optimizing the system composition, which is a complex action: both introduction of correcting systems improving the performance quality of the main useful function, and trimming these systems, transferring their functions to the elements already available in the system.

A special technical system perfection method is transformation *by analogy.* Our *method of structural analogy* demonstrated high efficiency when it was used with the Evolution Trees. Let us consider it in the first instance.

- **Structural analogy method**

Analogy may be defined as correspondence between objects or processes which gives ground for transfer of information, describing one object, to another object, having similar essential properties.

A special case of analogy is the so-called *"structural similarity",* the principal idea of which is that *it is always more convenient to compare not separate objects but similar sets of objects consisting of two main elements – a tool and an object worked by this tool.*

The structural similarity may be regarded as evolution of the method of focal objects developed by E.Kuntze and C.Whiting and providing the transfer of the properties of objects selected at random to a technical system being perfected. This may result in unexpected combinations which are further developed by means of free associations.

The method of structural similarity is described in the article by E. Novitskaya. The book describes structural similarity of five technical systems which are referred to the same structural scheme – a "shell with filler": an automobile tire, a parachute, a vacuum bottle, a boat, and a car. The table in Fig. 17 shows that well-grounded versions of similar transformations were found for each technical system.

Fig. 17. The joint table of transformations of various systems

We also succeeded in constructing a similar table after analyzing the display Evolution Tree. It was revealed that a transition to a microlevel is traced in the evolution of four main types of display – cinematograph, black-and-white TV, color TV, and 3D display. Mechanical scanning of light, color or position of an active pixel used in initial display versions was replaced with electronic scanning.

- **Trimming a system**

Trimming is always aimed at simplification of a system and reduction of its cost, the number and quality of the functions being preserved. This is achieved by removing from the system of the elements which are unnecessary according to the results of the *value-engineering analysis*. The functions performed by the removed elements are transferred to the remaining elements of the system.

The trimming technique includes the following steps.
First the composition, structure and functions of a system are determined. This is done by performing structural and functional analysis and dividing all the functions into main, auxiliary and unnecessary ones. Then a structural-elemental scheme of the technological process realized by this system is built and the system's elements are identified, each of which is considered as a tool for performing the system's function. After that, the summary value of each element is determined. It depends on the importance of the function performed by this element, the manufacturing and operational cost associated with this element. Elements having a high summary value are the first candidates for removal from the system.

The system analysis is followed by a creative stage – trimming the technological process itself. The trimming purpose is removing all elements - carriers of unnecessary and auxiliary functions and, if possible, also the

elements-carriers of main functions. Auxiliary and unnecessary functions should be eliminated and the main functions should be transferred to the remaining elements of the system.

A standard formulation is used here: *"It is possible not to perform a function, if this function may be realized a) at the expense of the previous operations, b) at the expense of the following operations".*
After removing some elements, a functionally ideal model of the process is built. The next step is creating a real design of the improved product based on this model.
In trimming the elements of a technological process, of great importance is creative thinking, absence of psychological inertia. The more decisive the actions of the team working on the problem, the more radical are the changes suffered by the system. The occurring problems usually contain aggravated contradictions which may be resolved by using TRIZ tools and other methods.
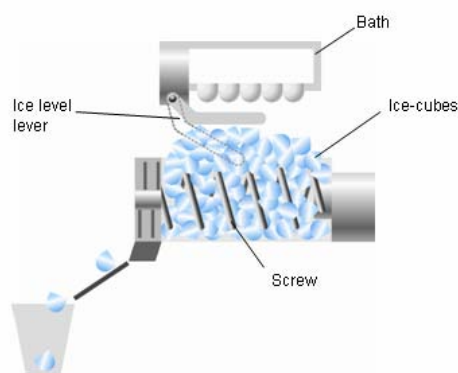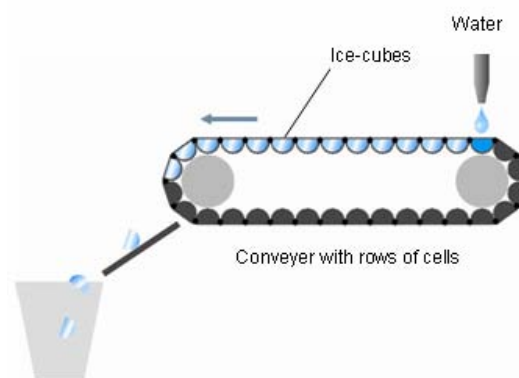

Fig. 18. "Ice-maker" before VEA

Fig. 19. A conveyer-type "ice-maker" (after VEA)

The work resulted in transforming a complex construction which occupied one third of a freezer (Fig. 18) into a wide conveyer provided with cells, located directly under the freezer ceiling and occupying almost no space (Fig. 19).

- **Expanding a technical system**

Technical system expansion implies introduction of new elements into the system's structure. It is obvious that such a complication of the system is only justified if its degree of ideality increases, i.e., if the functional possibilities of the system grow faster than the expenses necessary for their realization.

When synthesizing a new expanded technical system, one or several systems performing *additional functions advantageous for users* should be added to the prototype. As a result, the united system has several tools each of which works objects having different properties. The example of such a structure is a personal computer *which was equipped with additional peripheral devices in the process of evolution.*
Expanding a system may be accompanied by the introduction of new elements – carriers of additional functions improving the performance quality of the system's main function. Systems become more complicated and more effective, their reliability, durability and safety improve. This may be illustrated by the automobile braking system.

The combining of competing or alternative systems, i.e. systems performing the same function in different manners, may be regarded as a particular case of expanding the functions of a system when optimizing its

composition. All transformation versions described by the Evolution Tree are competing systems; this is just why this method is especially valuable for the processing of information obtained by means of patent search.

The main idea of the combining of alternative systems consists not in their mechanical connection, but in *transfer of the properties* of a more advanced system to the prototype being perfected. It is necessary to take one system as a prototype, impart necessary properties of one or several alternative system to it and resolve the arising contradictions. Generally, a less expensive system version is selected as a prototype.

The examples are combinations of alternative systems – a cistern and a truck as well as washing machines with a horizontal and vertical axis of rotation.

- **Resolving of contradictions by using TRIZ tools**

The most effective method for revealing and resolving the contradictions occurring during technical system perfection is TRIZ. This method implies an active use of inventive algorithms and psychological operators, which develop a special thinking style.

TRIZ foundations were laid in 1950-1980 by numerous enthusiasts-inventors headed by H. Altshuller. TRIZ generalizes the experience of creating millions of inventions, because the entire human history is the history of inventions. Trying to survive our ancestors had to solve many inventive problems. The multi-century history of technology evolution often demonstrates surprising examples of solutions to inventive problems, i.e. problems based on an aggravated conflict.

Erection of the Pyramids, construction of vessels and war machines required extraordinary wit and tenacity. Let us consider, for example, the process of manufacture, carriage and erection of the monument to Peter the First in Petersburg. Having both neither track-tractors nor powerful vessels, engineers delivered the 1600 t stone to a distance of 22 km, 13 of which it was carried by water.

Unfortunately, the accumulated experience is not preserved in full: skills were transferred from a craftsman to a apprentice on the quiet, part of information was lost for ever. Inventors themselves could not explain how the process of the appearance of new ideas ran. Though certain information about the process of the appearance of new ideas was gradually accumulated but the appearance of interesting solutions continued to be attributed to intuition, insight and the like.

H. Altshuller was among the first who defined and solved the problem of *how to find a method for quickly entering the area of effective solutions avoiding multiple trials and errors.* The solution to this problem was sought for in three directions:
- Analyzing a great many of patents which cover interesting inventive solutions and identifying the main contradiction-resolving principles used by inventors.
- Studying the philosophy of the resolving of contradictions in nature and society, first of all dialectic materialism.
- Because neither of the methods works outside the human brain, the third direction was studying the creativity psychology. It was not for nothing that the first article by H. Altshuller and R. Shapiro, dedicated to the control of thinking in inventive problems solving appeared in the "Voprosy Psihologii" (Psychology Problems) journal.

The result of that work was building a reliable and convenient bridge across the gulf separating the high sciences – philosophy that knows answers to all questions and psychology that understands how to organize the human brain for making it more efficient – from everyday hopes of an engineer, designer, scientist who tend to obtain and materialize new ideas.

The TRIZ training system is aimed at teaching:

30

1. the problem-solving technology which allows generating relevant associative prompts helpful in finding a solution idea,
2. the ability to see and amplify these prompts which sometimes float in one's mind unnoticed without special training.

TRIZ is most effectively used to solve contradictions occurring during the perfection of a prototype selected from the Evolution Tree. TRIZ is based on the evolution laws of technical systems, one of the derivatives of which – evolution patterns – was used in building the Evolution Tree.

In addition to the postulate about the law-governed evolution of technology and the requirement of ideality, TRIZ includes the following basic approaches:

1. identification and solving of contradictions,
2. operation with models,
3. consideration of psychological peculiarities of mind,
4. using an information base.

We are not going to dwell on the description of these approaches and refer the reader either to the full text of the book, or to the TRIZ materials. In addition to the books indicated in the list of reference, we may recommend the following Russian language web sites:

Generator	www.gnrtr.com
TRIZland	www.trizland.ru
TRIZ-pro	www.trizprofy.com
Metodolog	www.metodolog.ru
TRIZ Encyclopedia	www.triz.port5.com
Site "Altshuller foundation"	www.altshuller.ru

- **New possibilities for bypassing of competing patents**

To bypass a competing patent as well as to obtain a patent for a new invention, efforts of two key persons are needed. These are an inventor who finds some idea and a patent agent who prepares a patent application in accordance with the patent laws and files it with a patent office. It is just their work that sometimes causes a tremendous success. It is just their errors that may be too expensive for a company.

If an inventor and a patent agent act each in his own direction while bypassing a patent, two main bypassing methods may be identified.

- The first and most important is the *juristic* method.

This method implies taking advantage of the imperfections of the patent law and errors in patent descriptions. This does not entail any change in the object of invention itself. This situation presented in the form of a diagram shows that using the juristic method of avoiding a patent, the patent agent tends to cover the object of the invention being bypassed without introducing any change into it. To legally avoid a patent, it is necessary to find inconformity of the features of a real object, method or substance with the claims texts and the description of these features in a competing patent. Then it is necessary to describe an existing object of invention using other terms and reformulating the description in accordance with certain rules the content of which is the "know-how" of every experienced patent attorney.

The main range mark in the work is the existing patent law. This work often results in a possibility to avoid a patent by proving that it covers a previously known invention or to prepare a new application and obtain an alternative patent for the existing invention.

With a certain skill and sufficient competence of the patent agent, the legal method of avoiding a patent may prove very effective. Correctly drafting the claims and description capable of fully and reliably cover an obtained solution requires careful and thorough work. It may be suggested that such work had not been done thoroughly while preparing the competing patent. Preparing and examining an application and obtaining a patent is a very complicated and expensive procedure which requires participation of dozens of people. Any mistake made at some stage may cause insufficient protection of the invention. As a result, the patent may be avoided in future. In addition, no absolutely perfect patent law exists in nature and the level of patent protection may differ significantly in different countries. Sometimes this leads to funny situations. The example is the story which happened to the Australian patent agent John Keyo, who managed to get a patent for a wheel from the Australian patent agency.

- There is also an *inventive* method for obtaining a new patent.

This method implies more or less significant transformation of the structure of a device or a process - the subject of invention. Here the main role belongs to an inventor.
This is the most radical way out of a problem situation: find a better solution concept than that of your competitor and patent it, i.e., a new technical solution to the problem, a substantial change of the subject of invention is meant.

- We propose a "*juristic-inventive*" method for avoiding a patent.

It often happens that a variant patented by a competing company fully satisfies us and does not need any considerable change.

In this situation, there arises the following contradiction:
- It is necessary to change the subject of invention in order to obtain an alternative patent, surpassing the competing patent,
- Changes should not touch the principle of action of the technical system – subject of invention.

This contradiction can be solved by combining the juristic method and the inventive method in one "*juristic-inventive*" method which may be formulated as "change without changing". We mean a minimal technological change of the subject of invention. However, such a change often enables a patent attorney to build an alternative technical solution having legally significant distinctions from the patent being bypassed. It should be noted that the composition and structure of any system always implies some variability; therefore, it is always possible to introduce slight changes or sets of changes into the system's elements. The presence of the slightly differing elements in the system composition do not change the principle of action of the system but may be sufficient for obtaining a significative change of some patent feature, thereby giving the patent attorney additional possibilities of legal protection of the alternative patent.

In addition, study of possible changes in the system elements and patent features of an invention under investigation may also be used for effective patent protection of your company's developments. This is achieved by performing a reverse operation – considering a technical solution patented by your company as a competing patent and performing a patent search for basic alternative versions of the elements of the system described by this invention. Constructing several new models of the system which include changed elements will help find possible bypasses for your own patent in future. Alternative models of the system should be patented providing coverage by the principle of the so-called "patent umbrella" which blocks all possible basic bypasses of your patents by competing companies.
To solve such problems, it is advisable to use the Evolution Tree for the technical system described in the patent to be avoided. The more significative alternative solutions we obtain, the easier is orientating ourselves in this

field and finding more effective solutions helpful in avoiding of patents. Then it remains for us to find unpatented system versions on the Evolution Tree which are satisfactory from the technological point of view and we are ready for attacking the competing patent.

What operations should be performed to identify alternative versions of a competing invention? It is necessary:

- to determine a prototype, i.e. a patent to be avoided;
- to determine the function, composition and structure of a system protected by the patent;
- to determine patent features to be changed;
- to make patent search, find basic alternative versions of this system;
- to build the Evolution Tree of the system of interest;
- to find out which transformation versions are not covered by patents comparing the basic and real Evolution Trees (see Fig.16);
- to evaluate possibility of using these versions in the system, select the most suitable ones;
- to propose technical solutions based on these versions.
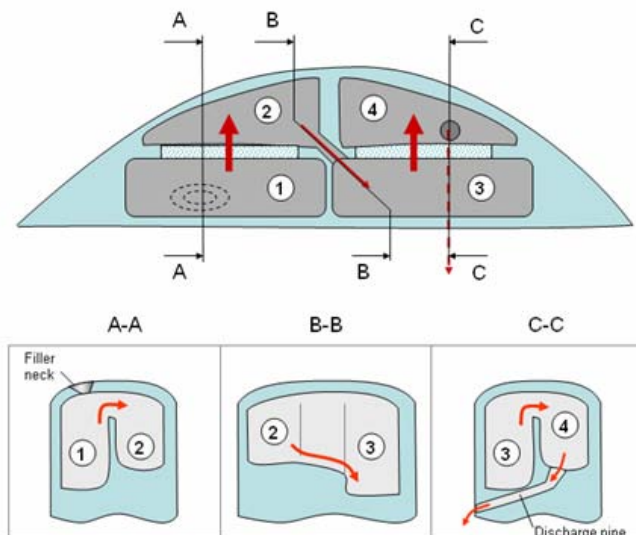- to provide legal protection of these solutions.

Fig. 20. A conditioning solution feeding
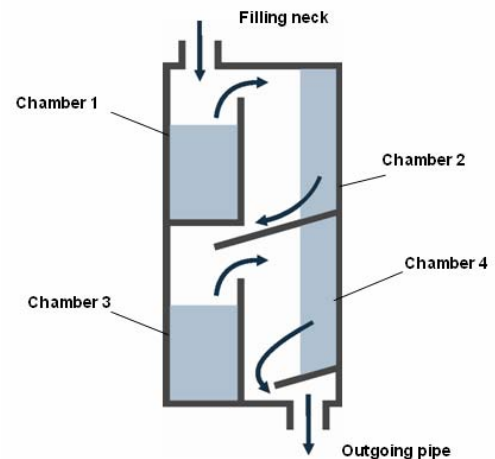device of a competing company

Fig. 21. An alternative design found by using the
Evolution Tree

The method for avoiding of patents is illustrated by describing the operations performed to bypass a real patent for a washing machine. The purpose was to find a new design for the conditioning solution feeding device operating on the principle of inertia – as the drum rotates quickly, the solution flows from one chamber to another and is discharged into the tank before rinsing (Fig. 20). After the device has been analyzed by using the Evolution Tree, there were found six alternative unpatented schemes of such a device. One of them is presented in Fig. 21.

- **"Effective forecasting of systems**

The traditional forecast based on Delphi method predicts change of some or other mechanisms of a system, but seldom determines the ways for realizing this change. Predictions made by using TRIZ offer specific technical solutions and allow formulating an integral and reasoned conceptual model of an article or a process of a new generation. Thus, a prognostic solution may be obtained by using all the TRIZ tools; moreover, any solution obtained at the invention level is prognostic. A patent application is neither more nor less than a description of a

new system version having certain advantages over the prototype. At the moment of filing as well as during the development stage of a new technical solution, it is unclear whether the system will evolve along this way or the obtained patent will remain unclaimed as a failed forecast try. Accordingly, increasing the forecast reliability of technology evolution is a matter of especial importance.

In the final analysis, forecasting the evolution of some system means building a chronological sequence of its versions. Doing this along the entire time axis – both in the past and in the future – would present the entire evolution of the system as if it is spread before the eyes. Placing the already existing versions of the system on the time axis is easy enough whereas understanding which version will be next tomorrow, in a week or in a month is not easy.

Suppose, the versions of a system under analysis are arranged in order of appearance (Fig. 22). Moving along the axis we reach the present day and cannot know which version is going next. This is similar to moving through a corridor train. When the doors between the carriages are open, you may go from carriage to carriage, enter any of them and examine everything inside it. But suddenly we come up a locked door (the present day) and do not know what is behind it. Another carriage or something else? To understand this, it is necessary "to detrain", i.e., to exit the system evolution time model and analyze all possible versions of the system by using some other models.

A good forecast may be defined as finding an optimal balance between the needs of a man and technological possibilities for satisfying these needs.
The idea that a man needs not objects and things themselves but the *functions* they perform and the *products* obtained as a result of performing these functions has become very popular in the past decade. We buy not goods but the ability of these goods to provide some consumer's effect, to give a product meeting some of our needs. The commodity itself is just a means for obtaining such a product.

There exist numerous motivation theories which analyze and structure the needs of a man making him act somehow or other. The authors of many of them (Herzberg, Maslow et al.) proceed from the assumption that a man needs what he misses in terms of physiology or psychology. Accordingly the human needs are different: primary (animal) and secondary (psychological). All the human needs are met by using some objects which are the operation products of some systems. For example, the need for intercommunication is met inter alia by means of a sheet of paper with letters written on it.

The minimal composition of a system which produces some product is a set of a tool and an object to be worked. Of course, to provide interaction of these elements, it is necessary to introduce some additional components into the system, but it is just this pair of elements that performs the system's function.
As follows from the diagram presented in Fig. 22, a system will be efficient and competitive if the product it gives is located at the point of meeting of two vectors. One of the vectors symbolizes the consumer needs and the other one describes the operation of a system which gives a product for meeting these needs. The more ideal the system, the lower the cost and the higher the quality of the product it gives, the more needs will it encompass in a given market niche.

Let us assume that the minimal composition of a system includes a tool and an object worked by this tool. Then a new version of the system can be obtained by changing one of these elements. To make a forecast, it is necessary to imagine basic promising versions of this system. Generally, the object to be worked for meeting the needs is already determined. Hence, forecasting consists in replacing the existing variants of the tool with a more promising one.

In accordance with the diagram presented in Fig. 22, the following algorithm of obtaining a promising prognostic version of the system is taking shape:
- determining the user's needs,
- determining an object to be processed by the system
- determining a product to be produced by acting on the object

34

- transforming the object (if necessary),
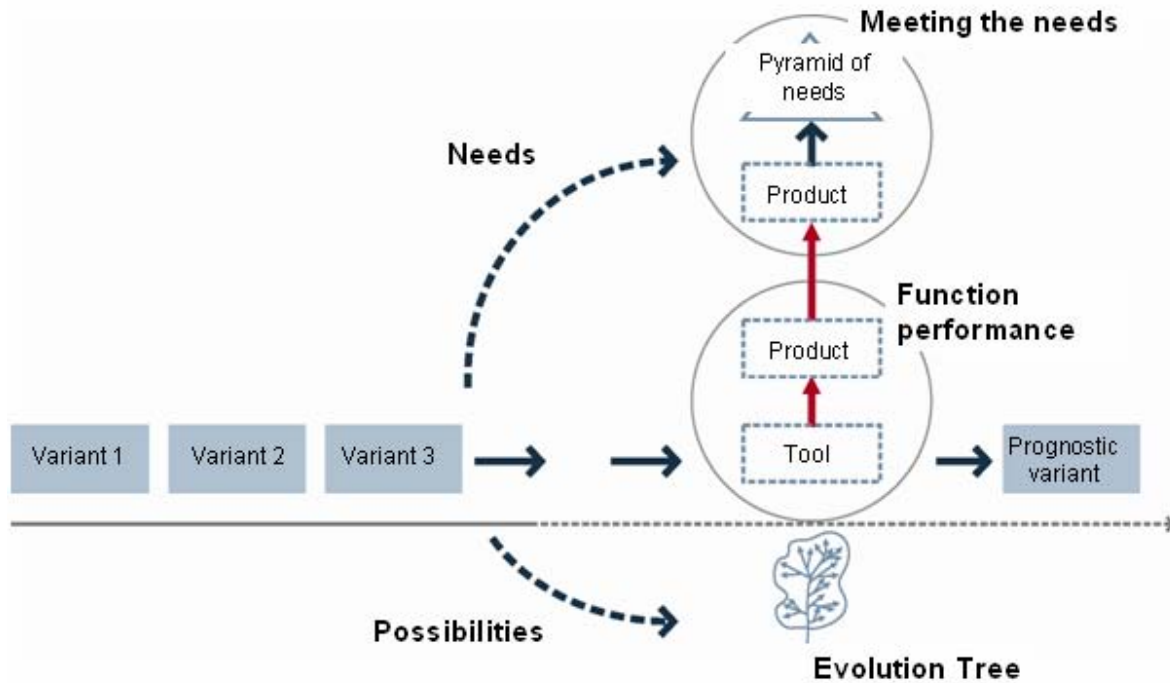- transforming the tool.

Fig. 22. Diagram of meeting the needs

For effective transformation of a tool and an object, it is important that there be a sufficient number of their versions within eyeshot and, ideally, all basic embodiments of these elements of the system. The Evolution Tree hits off perfectly with the role of such a visualizer because it presents all existing embodiments of an object under investigation as they appeared in accordance with the evolution laws. In addition, there is an opportunity to obtain significative information about missing, probable versions. New embodiments of the object may be found by comparing the real Evolution Tree we built with the evolution patterns of the Basic Evolution Tree.

Forecast is always an equation in many unknowns. Using the Evolution Tree for visualization of probable system versions makes it possible to determine some of these unknowns. This will eventually simplify forecasting and make it more complete and accurate. In a "marked out" information field, a researcher sees a clear picture presenting all basic versions of the system. Having this information, the researcher can completely concentrate on solving the reverse problem and study the system's versions trying to find an answer to the following question:

- What for may one or another version of a system be needed?
- Which parameters of a new system will change and how will they change?
- What are the advantages of each version?
- What are the drawbacks of each version?
- How will the quality of the product produced by the system change?
- Will the new version be more expensive or less expensive?
- How will the supersystem of the system under investigation change?

Thus, we are speaking about an ordinary research work, study of the model of a new hypothetic system.
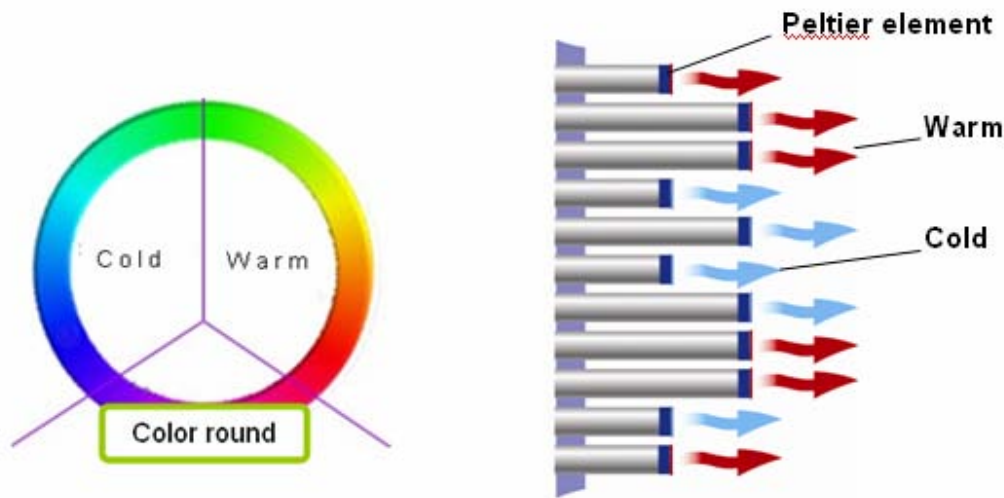
Fig. 23. «Color» display for blind peple

The book describes some prognostic solutions obtained after analyzing the display Evolution Tree, for instance, some proposals aimed at increasing the display informativity for blind and starblind people. One of the most interesting solutions is an original "color" needle display. The points of the needles, creating a raised picture, have miniature Peltier elements. Activating these elements can make the needles cold of hot, thereby imitating colors perceived as "cold" and "warm" by a man (Fig. 23).
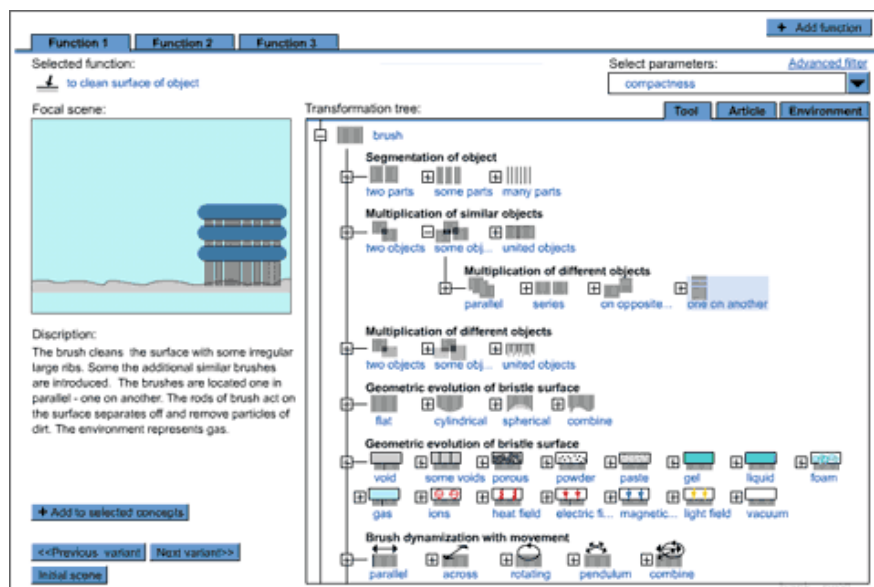

Fig. 24. Concept Generator prototype
Part of the Evolution Tree for a tool

The afterward shows how using a computer can considerably increase the effectiveness of the prognostic analysis based on the Evolution Tree. Using the Evolution Tree, we have developed a concept and a prototype of a computer program which helps a designer quickly obtain a great number of function performance ideas which differ from each other in quality. Each idea gives an elementary solution concept while a combination of these concepts forms a complex concept of how to build a technical system. Having obtained a series of

concepts, a designer can start analyzing them or selecting the most promising ones according to recommended criteria.

To generate concepts, a technical system is disintegrated to see the elementary parts performing a certain function. Evolution patterns are created for the key elements in accordance with the structure proposed by the Basic Evolution Tree (Fig 24).

The approach based on the Evolution Tree makes it possible to build a model of any technical system. This model may be used to forecast the system evolution, bypass competing patents and provide strategic planning of business. Systematization of evolutionary models built for different technical systems allows a designer to create a virtual technical world – the world of function performance models with all its relations and variety where the most different systems and processes can be simulated.