First published in the proceedings of TRIZCON2005, the annual conference of the Altshuller Institute, in Brighton, MI USA, April 2005.

OVERALL DATAFLOW STRUCTURE FOR CREATIVE PROBLEM SOLVING IN TRIZ/USIT

Toru NAKAGAWA

(Osaka Gakuin University, Japan)

ABSTRACT

It has been a common understanding in TRIZ and many other scientific/technological problem solving procedures that instead of trying to go directly from one's specific problem toward specific solution(s) one should better go around through a generalized problem and its generalized solution in some standard model. This scheme, however, is often based on mapping by analogical thinking and has pitfalls of ambiguity in the model selection before analysis and of forcing different ways in analysis after selection.

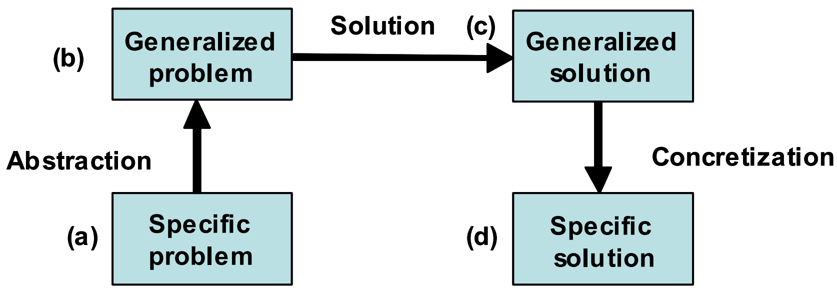
The overall structure of creative problem solving in USIT has been built into a six-box scheme in the dataflow representation. User's specific but vague problem should be converted into a well-defined user's problem, and analyzed to obtain understandings of the present system and of its ideal system, then transformed (by USIT operators for solution generation) into ideas of new system(s), and further built into conceptual solutions, and finally implemented into user's specific solutions.

It should be noted that the analogical mapping is eliminated. Problem solvers are guided all through the procedure in a logical and yet creative way. The difficulty in the multiple-path structure of TRIZ has been streamlined. All the methods and knowledge-bases developed in TRIZ have been reorganized, unified, and simplified in this new scheme.

1. INTRODUCTION

It has been a common understanding in **TRIZ** (Theory of Inventive Problem Solving) and many other scientific/technological problem solving procedures that instead of trying to go directly from one's specific problem toward specific solution(s) one should better go around through a generalized problem and its generalized solution in some standard model. The basic scheme may be presented in the Four-Box Scheme [1] as shown in Fig. 1.

Fig. 1 Four-Box Scheme of Problem Solving in General



In the **Four-Box Scheme** of problem solving, we first go 'upward' to the abstract level from user's specific problem (a) to a generalized problem (b), where the essence of the original problem (a) in some sense need to be extracted after discarding some other detailed or non-essential parts. Then the generalized problem (b) is to be solved into a generalized solution (c) in some standard way known as a theory, a model, a methodology, or whatever called. Finally, by using the generalized solution (c) as a clew, the problem solver need to find a specific solution (d) applicable to his/her own specific problem (a).

This scheme, however, just shows a philosophical outline whose contents are left to be specified for any methodology we are going to take. Individual methods in TRIZ, such as Altshuller's Contradiction Matrix Method and Su-field Modeling Method, have been understood in this scheme rather well (even though in an abstract way) [1]. But Overall Structure of TRIZ methodology has never been shown in the same scheme, until recently by the present author [2].

Overall Structure of TRIZ, both in its orthodox versions [3-6] and in the current modernized version by Darrell Mann [1], has been found unfortunately not simple but rather confusing [2]. This finding may not be a surprise because most TRIZ experts knew [1] that there are a lot of different ways and opposing opinions among them concerning Overall Procedure of problem solving with TRIZ. The point is that 'Overall Structure' represented in 'Data-Flow Diagram' like the Four-Box Scheme has demonstrated the origins of drawbacks in TRIZ much more clearly than 'Overall Procedure' represented in popular 'Flowcharts' or 'Process-Flow Diagrams' [2].

The present author has been recommending to use **USIT** (Unified Structured Inventive Thinking), developed by Ed Sickafus [7, 8] and extended by the present author [9-13], as a simple, unified, and effective Overall Procedure for applying TRIZ. Flowchart representation of Overall Procedure of USIT has been used for demonstrating its simple and streamlined nature since its early stage of development. 'Overall Structure' of USIT has lately been drawn by the present author in Ref. [2], and has been found instructive to understand the essence in the creative problem solving methodology.

In the present paper, 'Overall Structure' and 'Overall Procedure' of USIT are examined further in comparison with those of TRIZ. Implications of them are discussed in relation to the ways of application of USIT and in the context of general scheme in creative problem solving.

2. EXAMINATION OF OVERALL STRUCTURE AND OVERALL PROCEDURE OF TRIZ

2.1 TRIZ Knowledge Bases and Individual Methods

It is well known that TRIZ [1, 3-6] has developed a number of knowledge bases of facts and principles for supporting/leading creative problem solving. They include:

- Effects Database
- 40 Inventive Principles
- Contradiction Matrix
- 76 Inventive Standards
- Trends of Evolution of Technical Systems

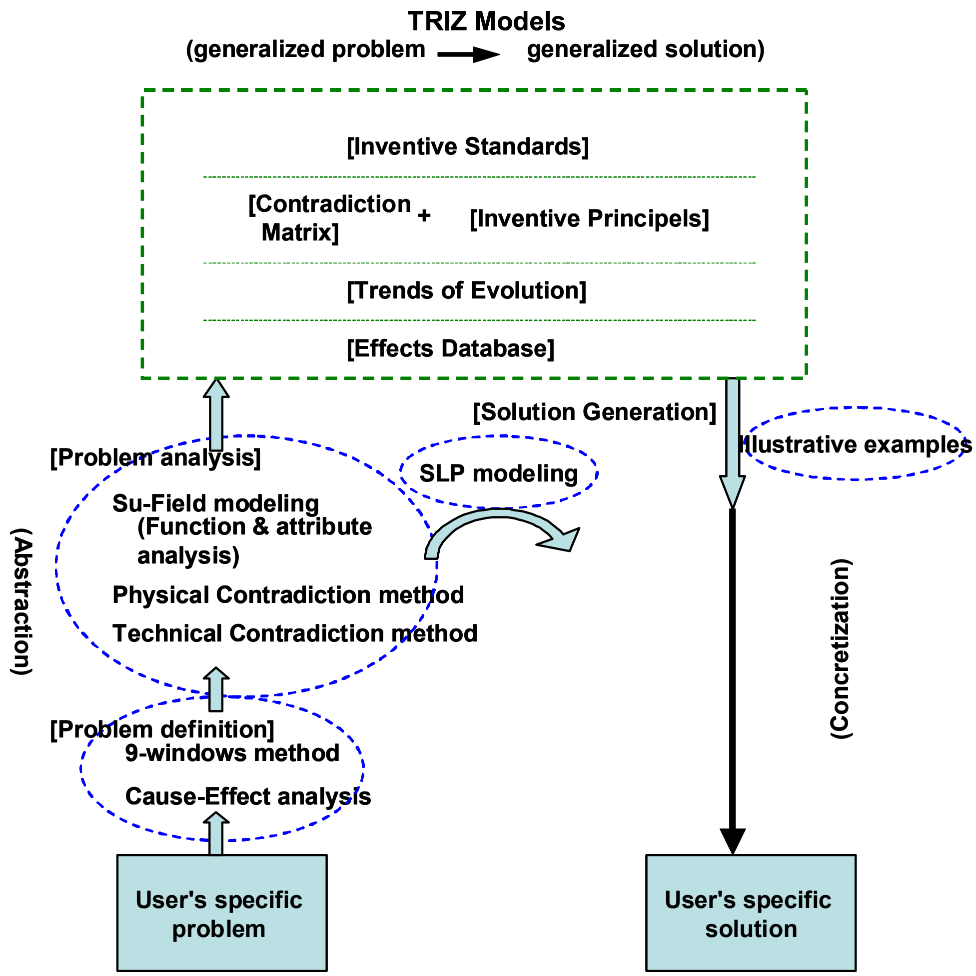
TRIZ has also developed a large number of individual methods and techniques for carrying out creative problem solving. How to use them individually is well understood among TRIZ experts. They include:

- 9-Windows Method (or System Operator Method)
- Su-Field Modeling
- Technical Contradiction Method
- Physical Contradiction Method
- Smart Little People's Modeling
- Cause-Effect Analysis
- Function Analysis; Function and Attribute Analysis

2.2 Overall Structure of Problem Solving in TRIZ

Overall Structure of Problem Solving in TRIZ has been represented as shown in Fig. 2 [2]. This figure is represented basically in the form of 'Data-Flow Diagram'. Information in problems, solutions, and knowledge bases are shown in boxes, while procedural methods with arrows accompanied by ovals. The roles and positions of major knowledge bases and procedural methods seem to have basic consensus among TRIZ specialists [1,3-6].

Fig. 2 Overall Structure of Problem Solving in TRIZ, Ref. [2]



This figure suggests the following characteristic points in Overall Structure of Problem Solving in TRIZ.

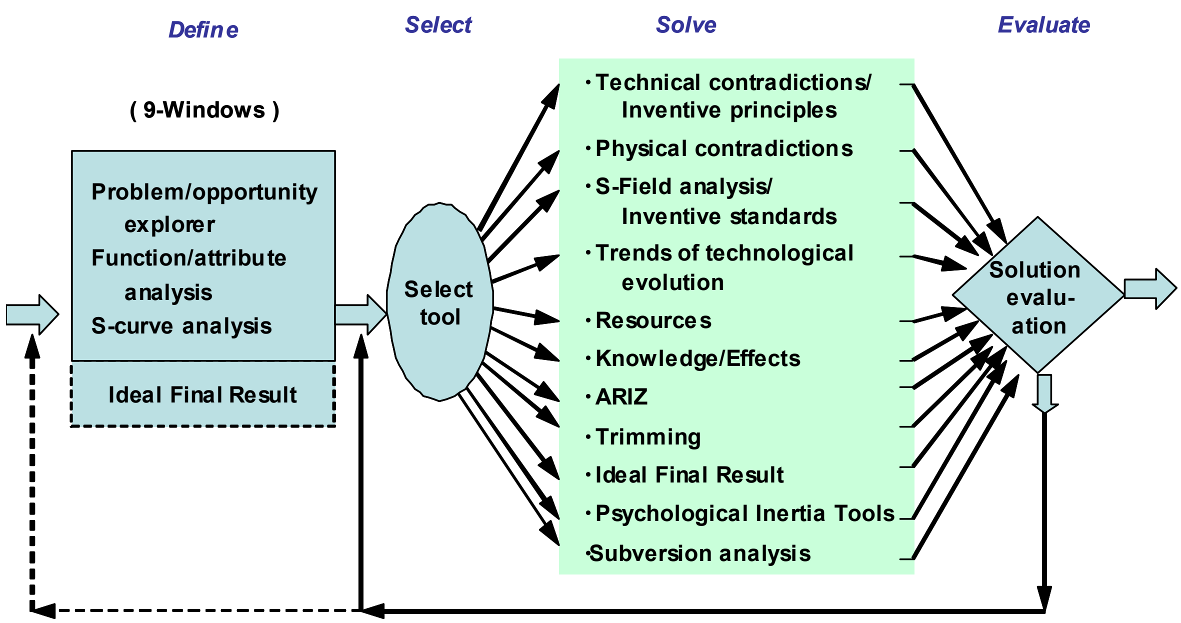
- Knowledge bases in TRIZ may be placed at the positions of generalized problems and generalized solutions. In some cases (e.g., Inventive Standards and Inventive Principles) generalized problems are separated from generalized solutions, while in other cases (e.g., Trends of Evolution and Effect Database) generalized problems are not well separated from generalized solutions.
- Major TRIZ knowledge bases are placed in parallel, because they do not have any sequential order in their logics. They are more or less independent of one another, but have some overlapping [1].
- Major methods in Problem Analysis Stage are connected to particular solution generation tools. Thus the problem analysis is usually carried out with one method at a time, forming only partial understanding of the problem [13].
- Illustrative examples are the only support at the stage of concretization from generalized solution to user's specific solution.

2.3 Overall Procedure of Problem Solving in TRIZ

Overall Procedure of problem solving in TRIZ means some standard procedure for using all the above-mentioned knowledge bases and individual methods in an effective way. This has been a most controversial issue among TRIZ specialists [1]. Genrich Altshuller developed these knowledge bases and individual methods little by little sometimes in parallel and sometimes in sequence in his history for searching more powerful and more effective components and overall procedures. There are a number of versions of overall procedure developed by Altshuller (in the name of ARIZ) [3] and proposed later by a number of his followers such as Boris Zlotin and Alla Zusman [5], and Yuri Salamatov [6].

Recently Darrell Mann [1] has established an updated version of TRIZ in the name of 'Systematic Innovation'. Overall Procedure of his methodology can be shown in Fig. 3. The methods shown in this figure are based on the chapter titles of his textbook [1].

Fig. 3 Overall Procedure of TRIZ (Darrell Mann's 'Systematic Innovation' [1])



Principal stages of Overall Procedure [1] in Fig. 3 may be summarized as:

(1) Problem Definition Stage: Three methods (i.e. Problem explorer, Function/attribute analysis, and S-curve analysis) are mandatory, and one (i.e., Ideal Final Result) is highly recommended. The 9-Windows Method is most relevant to this stage but works all through the procedures.

(2) Tool Selection Stage: Corresponding to the problem situations, a special table recommends the user a few appropriate solution-generation tools. For each of 19 cases of situations, up-to-four tools are recommended with priority order.

(3) Solution Generation Stage: There are 11 individual tools for solution generation. Mann advises to learn these tools one by one when they become necessary to apply to your own case.

(4) Solution Evaluation Stage: Determine which is the best among the generated solutions and decide

whether the solution is good enough. If not satisfactory, go back to the tool selection stage (2) or to the top of the whole procedure for re-defining the problem.

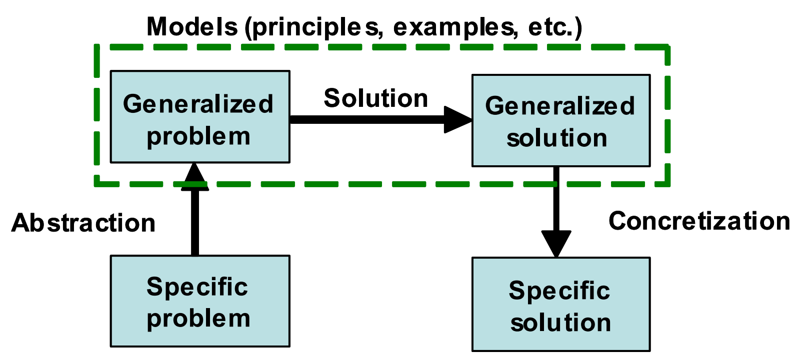
Even though Mann has described all these methods clearly, his Overall Procedure of TRIZ (in its new generation) contains the following problems:

- The phase of analyzing the problem and its system is not articulated, but is separately included in the Problem Definition Stage (e.g., Function/attribute analysis) and in the Solution Generation Stage (e.g., Contradiction Matrix Method and Su-field Analysis).
- The table of tool selection is too large and complicated to remember.
- There are so many tools for Solution Generation Stage; they are more or less independent of one another and yet have some overlaps without any clear philosophy.
- Most of the tools in Solution Generation Stage contain or request their own methods of analysis, each of which gives partial understanding of the problem situation.

2.4 Basic Scheme of Using Models with Enforced Analogy

The underlying assumption of using Models (or Knowledge Bases) in TRIZ, as in many scientific and technical methodologies, is illustrated in Fig. 4.

Fig. 4 Four-Box Scheme of Problem Solving with Models of Problem-Solution Pairs



This scheme is modified from the basic Four-Box Scheme (Fig. 1) only slightly in the point that Generalized Problem is paired with Generalized Solution, as enclosed by a broken-line box. This is quite natural because a large number of models are already built, where a model is typically given by a problem-solution pair. Such models are everywhere in the textbooks, in patent databases, in software tools, etc. It does not make sense to mix them up and choose a solution belonging to a different model.

This slight modification, however, makes a significant difference in the problem solving procedure. This is because different models usually request different ways of abstraction. A Generalized Problem in a model is the target of abstraction. Thus even with a specific problem at hand, the problem solver has to make abstraction in different ways as requested by different models.

Thus solving a specific problem in this scheme should be as follows [2]:

- Select a model first,
- Then analyze (or make abstraction or mapping of) the specific problem in the way requested by the model to get a generalized problem,
- Further solve it following the method of the model to get a generalized solution,
- Finally try to concretize it into a specific solution.
- Evaluate the resultant specific solution and repeat the procedure by selecting different models one after another until you find a satisfactory solution or exhaust the models.

This procedure is critically based on a mapping from the specific problem onto a model problem which is chosen arbitrarily beforehand. There are cases that the models have some analysis method to support the mapping process in a logical way. However, in the context of inventions, the models are often just preceding examples and do not provide any clear method for guiding the mapping process. In the latter cases the procedure may be called 'Enhanced Analogy'. Intuitive thinking and enlightenment are main sources of problem solving in such cases.

In the scheme of Fig. 4, there occurs a paradox in the effectiveness of problem solving:

- **Paradox of More Models with Less Effectiveness**: Suppose individual models in the set of methodology have partial applicability in solving problems, then the more models there are in the methodology, the more powerful but the less effective the methodology becomes as a whole.

In this situation, we want to have more models because we know that no model so far is almighty. By selecting a model we may go through the procedure in Fig. 4 to derive a specific solution with some satisfaction. But we want to select another model because there may be a possibility of deriving a different specific solution with better satisfaction. Only when we exhaust the models we can say with confidence that we have found the best possible solution in the methodology. By increasing the models, the best solution obtainable with the whole set of models (i.e. the Power of the methodology) may get better or stay the same. However, the overall effectiveness of the methodology in the sense of cost-performance may be reduced (or sometimes increased).

The current situation related to Overall Procedure in conventional TRIZ seems to be caught by such a paradox. Parallel positions of major models (including Inventive Principles, Inventive Standards, and Trends of Evolution) and also of a large number of their sub-principles may have caused the Paradox of More Models with Less Effectiveness.

For increasing the effectiveness of Overall Procedure in such a paradoxical situation, we may take different **strategies**, some theoretical while others practical, as follows:

- Make models produce highly satisfactory (generalized or specific) solutions for some ranges of problems. -- E.g., Separation Principle for Physical Contradiction type problems [3].
- Make individual models highly applicable to some areas/types of problems and make their applicability readily seeable. -- E.g., Trends of Evolution [1].
- Evaluate the power and effectiveness of individual models beforehand, by experiences or by benchmarks, and use the individual methods in the priority order as determined by the evaluation results. -- E.g., the Tool Selection Table by Mann [1].
- Eliminate less powerful and less effective individual models. -- E.g., Israeli SIT and ASIT [14].
- Arrange the models in such an order that the information obtained in the preceding models help use the latter models. -- E.g., ARIZ (Technical Contradiction and then Physical Contradiction) [3].
- Establish a set of models for which a common method of abstraction (or analysis) is applicable. -- E.g., Functional Analysis as a basis for many tools [1].
- Introduce some common framework which can unify the relevant models into a simpler, powerful, and effective new model, and eliminate the old ones. -- E.g., USIT [7] (See Section 3).

3. OVERALL STRUCTURE AND OVERALL PROCEDURE OF USIT

3.1 Basics of USIT (Unified Structured Inventive Thinking)

USIT was developed by Ed Sickafus [15] in 1995 and intensively used at Ford Motor Co. At first he adopted Israeli SIT (Systematic Inventive Thinking) [14], which was a much simplified version of TRIZ, and then he introduced a new framework and built USIT. The present author have introduced USIT in Japan since 1999 [9] and have extended it further by reorganizing TRIZ methods of solution generation into the USIT framework [10]. USIT, especially in its current form in Japan, has the following general features [2]:

- USIT is a unified and structured methodology for creative problem solving principally targeted to technological fields.
- USIT is based on TRIZ especially in its philosophy, and contains many methods derived from TRIZ, and hence it may be regarded as a new way of applying TRIZ or a new-generation of TRIZ.[1]
- USIT intends to be applied to technical problems at any stage of lifecycle of a product for obtaining creative solutions at the concept level.
- USIT guides the problem solvers especially in their steps/ways of thinking.
- USIT has a clearly defined Overall Procedure with three stages (i.e., Problem Definition, Problem Analysis, and Solution Generation stages).
- USIT also has a clear Overall Structure, where the requirements of information input/output in each stage are clearly defined.

---

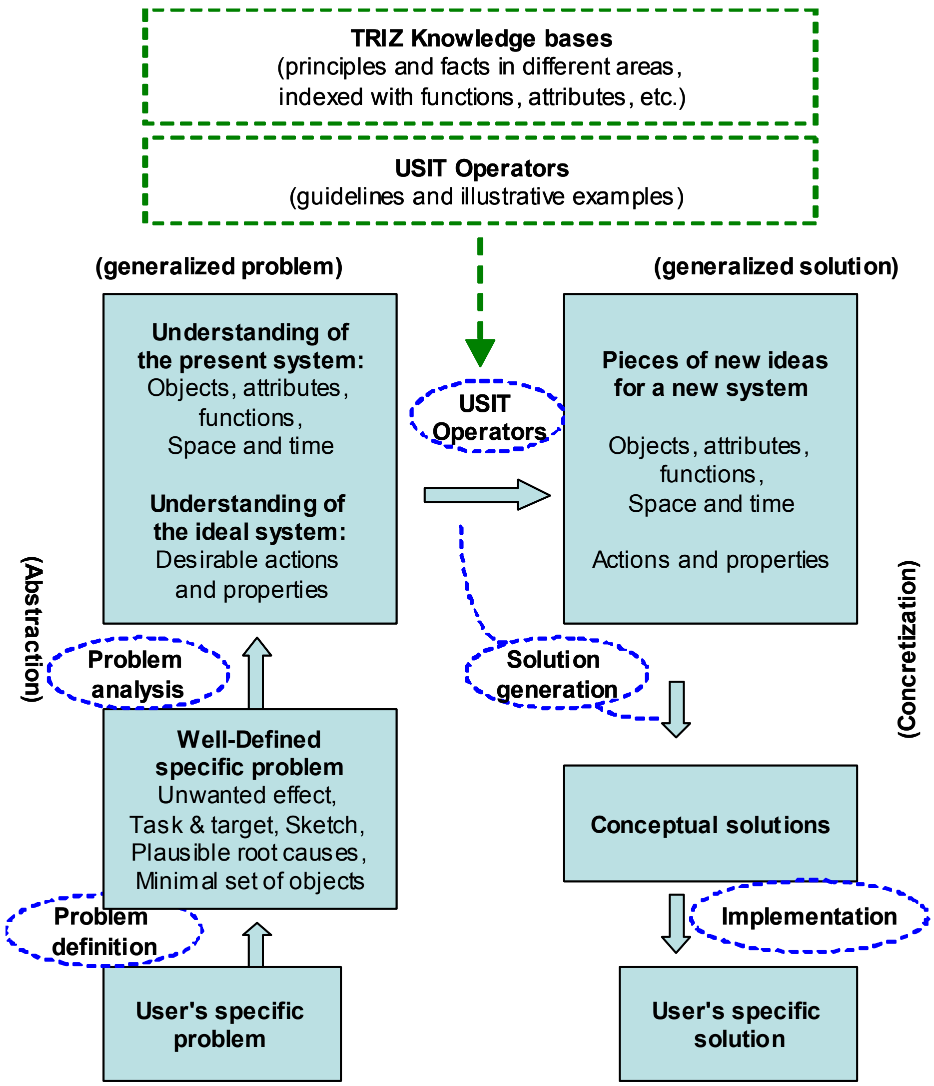
[1] Ed Sickafus never says USIT is a part or a form of TRIZ. (Private communication with Sickafus)

- USIT uses the concept of Objects-Attributes-Functions in a unified way as a new framework of methodology [7]
- USIT does not depend on knowledge bases, handbooks, nor software tools, in contrast to conventional TRIZ.

3.2 Overall Structure of USIT

The Overall Structure of creative problem solving in USIT is shown in Fig. 5 [2].

Fig. 5 Overall Structure of Problem Solving in USIT, Ref. [2]



The six boxes in Fig. 5 shows the information at each stage of problems solving. They are briefly summarized as:

(a) **User's specific problem**: A problem is recognized by the user(s), sometimes vaguely and some other times sharply, often in a complex and confusing situation.

(b) **Well-defined specific problem**: The result of the problem definition stage, where one problem is determined to be solved in the project. The problem must have the following information [7]:

- Unwanted effect: An effect which urges us to eliminate/prevent/reduce/etc.
- Problem statement: A statement of 1-2 lines clarifying the task and target of the project.
- Sketch: A simple drawing of the problem situation showing its mechanism.
- Plausible root causes: Statements of root causes of the unwanted effect. It is desirable to have them identified with experiments, but acceptable to have them speculated through the best-effort analysis of the mechanism. Multiple causes may be listed.
- Minimal set of objects: Minimal set of relevant objects (or components) which contain the problem.

(c) **Understanding of the present system and of the ideal system**: This information forms the generalized problem resulting from the problem analysis. The present system in the problem (or a possible base system in the current technology) is represented in the following terms [7]:

- **Objects-Attributes-Functions**: Functional relationships among objects are represented in Functional Analysis Diagram, especially for revealing the intended mechanism of the original design. Attributes (i.e., categories (and not values) of properties) of objects relevant to the unwanted effect are listed up, for confirming the plausible root causes and revealing possibilities of changing the unwanted effect.
- **Space and Time**: Characteristics of the system and problem situation are revealed in terms of space and time. This information provides background clews for solving Physical Contradictions.

An ideal system for the problem is also identified and analyzed in the following terms [7]:

- **Desirable actions**: Behavior of the ideal system is imagined and shown in a hierarchical structure of desirable actions by using plain, non-technical terms.
- **Desirable properties**: For supporting the desirable actions in the ideal system, desirable properties are listed up as they come up with.

(d) **Pieces of new ideas for a new system**: Generalized Solutions in its initial form, obtained as pieces of new ideas of modifying the elements of the present system or of building up some part of new system(s). They are expressed in the term of and with the background of information (c).

(e) **Conceptual solutions**: New concept-level solutions built up for fitting to the requirements of the Well-defined specific problem (b). They may still be qualitative and not designed yet in engineering. This is the goal of solution generation in USIT as the creative problem solving.

(f) **User's specific solution**: Concrete solution(s) which fulfill user's requirements and desires in (a) and are obtained only after actual implementation work onto the conceptual solution(s). Final solution need to be provided as a new product, or installed in the equipment, etc., after the examination with technological, business, and social criteria. The whole process of problem solving may be said successful only when the solutions (f) are accepted successfully in the engineering, business, and social circumstances.

3.3 Overall Procedure in USIT

The Overall Structure of creative problem solving described in Fig. 5 is realized in USIT by the use of Overall Procedure as shown in a Flowchart in Fig. 6. The Flowchart of USIT has been used since the initial stage of USIT development [7] with minor refinement from time to time [9, 2]. It is remarkable that the USIT Flowchart has a streamlined sequential structure of three principal stages: Problem Definition Stage, Problem Analysis Stage, and Solution Generation Stage. They are briefly described below (together with an additional Implementation Stage which follows the USIT procedure):

(1) Problem Definition Stage:
The process for converting user's specific problem (a) into Well-defined specific problem (b) [7]. This process is usually carried out through a group discussion in the USIT project team (See Section 5.1). The requirements for a Well-defined specific problem (b) are shown to the group, and no particular procedural methods are provided in USIT.

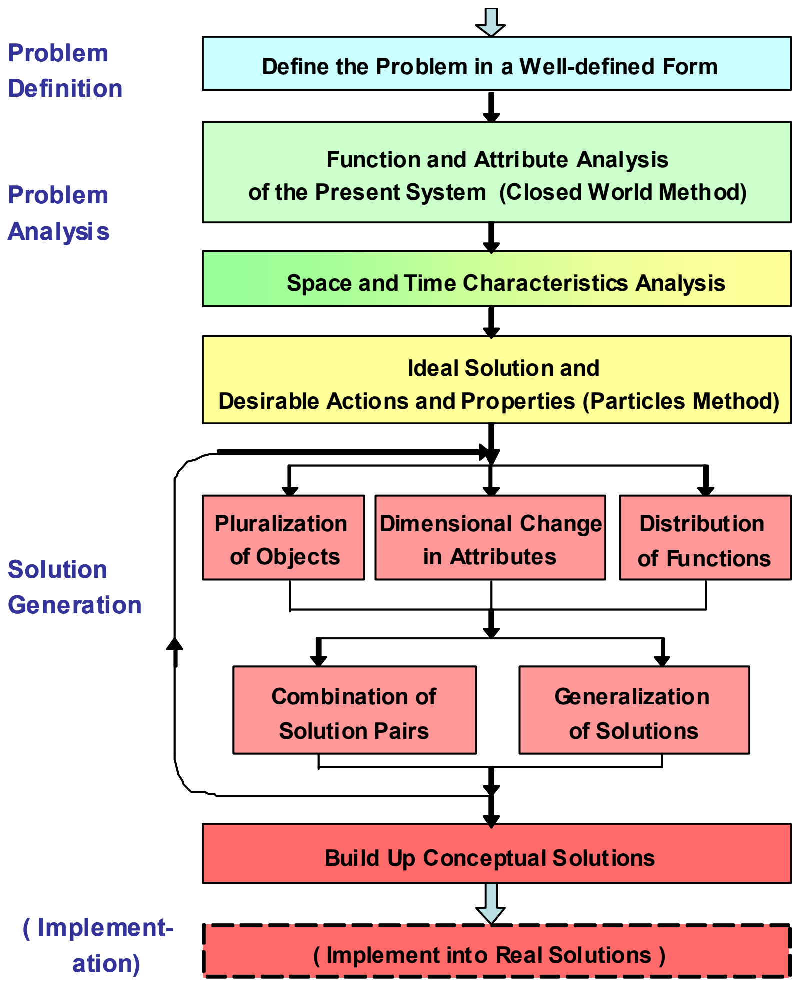
(2) Problem Analysis Stage:
In order to obtain the understanding of the present and ideal systems (c) in the Well-defined specific problem (b), the following three analysis methods are carried out in sequence [7, 8, 9]:

(2-1) **Function and Attribute Analysis of the Present System** ('Closed-World Method'): The present system in the problem is analyzed with the following two methods in sequence:

- **Functional Analysis** ('Closed-World Diagram Method'): Functional relationships of the original design in the present system are drawn. The most important Object in the system is placed at the top, and then other Objects are drawn downward basically in sequence of the 'functionally beneficial relationship' (or useful function) for supporting the functions of the upper objects.
- **Attribute Analysis** ('Qualitative-Change Graph Method'): A parameter representative of the unwanted effect (or else a performance parameter) is chosen, and then any relevant attribute of the objects in the system are listed up after distinguishing positive or negative correlation with the unwanted-effect parameter.

(2-2) **Space and Time Characteristics Analysis**: Draw some diagrams for revealing characteristic features of the system/problem with respect to space and to time. Typically, some parameter which represents the system behavior is plotted against a characteristic spatial coordinate, and

Fig. 6 Overall Procedure of Problem Solving in USIT (Flowchart)



against a time coordinate [9]. Sometimes it is useful to simply itemize the functions in action along characteristic space and time axes.

(2-3) **Particles Method**: A process for identifying Ideal Solution and imagining desirable actions and desirable properties is carried out in the following sequence [7, 9]:

- Sketch the present system: especially to show the mechanism of the problem.
- Sketch an Ideal System: Imagine an ideal result and draw it, without trying to draw any means to achieve the result.
- Draw 'Particles' with x marks: Draw x marks at the positions where the sketch of the Ideal System differs from that of the present system. The x marks are now regarded as 'Particles', which are imagined to be magical substances or 'Fields' (in the TRIZ sense) capable of performing any desirable action and having any desirable property.

- Imagine Desirable Actions and draw them in a hierarchical diagram: Ask the magical Particles to achieve the goal of the ideal solution, and then break down the actions that the Particles are imagined to be doing for us. The desirable actions are drawn in a logical AND/OR tree diagram, generating a prototype of a hierarchical presentation of solutions.
- List up desirable properties: List up desirable properties to support each of the desirable actions. Broadening the idea space is more important at this stage than selecting ideas with feasibility criteria.

(3) Solution Generation Stage:

Process to generate many pieces of new ideas (d) and then to build up conceptual solutions (e) for new system(s). Five solution generation methods in the form of **USIT Operators** are applied repeatedly without any fixed order [7, 10-12, 16]:

- **Pluralization of Objects**: 'Plural' here means any number except 1; thus 0, 2, 3, ... infinity, 1/2, 1/3, ... 1/infinity, 0.9, 1.1, etc. An Object is trimmed (0), multiplied (2, 3, ... infinity), divided (1/2, 1/3, ...1/infinity), modified (0.9, 1.1, etc.), newly introduced, and so on.
- **Dimensional Change in Attributes**: Each Object has a wide range of Attributes, namely a large number of dimensions of properties. Thus USIT Operators provide dimensional changes in Attributes, which mean using (or activating) a new Attribute, eliminating (or de-activating) an Attribute, introducing space or time dependence of an Attribute, and so on.
- **Distribution of Functions**: Functions in the system are re-distributed among the Objects and in the space and time, so that useful functions are carried out better (or introduced newly) and harmful functions are prevented or reduced.
- **Combination of Solution Pairs**: Pairs of Solutions (or solution elements) are combined in various ways in space, in time, in functional relationships, in structural relationships, etc., so as to make better solutions by overcoming weak points in both sides of the pairs.
- **Generalization of Solutions**: Solutions are expanded and enhanced by using generalization (or abstraction) and specification (or concretization) repeatedly, and thus a system of solutions is built into a hierarchical structure.

These five USIT Operators are always applicable to the elements of the present and solution systems expressed in terms of Objects-Attributes-Functions, and to the known and newly-generated solutions. Thus in USIT, the process from (c) to (d) for obtaining pieces of ideas of a new system and the process from (d) to (e) for building conceptual solutions are carried out in an inseparable repetition process.

(4) Implementation Stage (just after USIT):

Conceptual solutions are to be evaluated, experimented, prototyped, designed, manufactured, installed, tested, etc. so as to make them real into successful specific solutions for users. This stage needs technical and business capabilities and decisions, and hence is out of the range of USIT as a creative problem solving methodology.

4. TRIZ HAS BEEN REORGANIZED INTO USIT

4.1 History and Strategy for Reorganizing TRIZ into USIT

This section describes how the methods, principles, and philosophy in TRIZ have already been reorganized into those in USIT. Historically, the reorganization was achieved in three major steps:

(i) Genady Filkovski much simplified TRIZ into SIT in Israel in early 1980s. See Ref. [14].

(ii) Ed Sickafus adopted and enhanced SIT and built a new framework of USIT in USA in the later half of 1990s [15].

(iii) Toru Nakagawa enriched USIT solution-generation methods by reorganizing all the TRIZ methods into USIT Operators in Japan during these few years [10].

It should be noted that for converting a methodology into another, various types of transformation (and their combinations) are theoretically possible. They include: no-change, elimination, division, taking-away, modification, merging, making flexible, introducing another dimension, addition, changing parameters, enhancing, combination, standardization, trimming, simplification, classification, regrouping, unification, introducing a new framework, reorganization, and so on.

In the case of transition from TRIZ into USIT, because TRIZ is already so big, rich, and complicated, the most meaningful strategy has been **reorganization into a new framework and simplification by unification**. (See the comments at the end of Section 2.4, and note the contrast to many other strategies trying to keep TRIZ powerful without touching it [17], to make TRIZ more powerful by enhancing some methods or by introducing additional methods, and to make TRIZ easy to apply by using only some parts of it.)

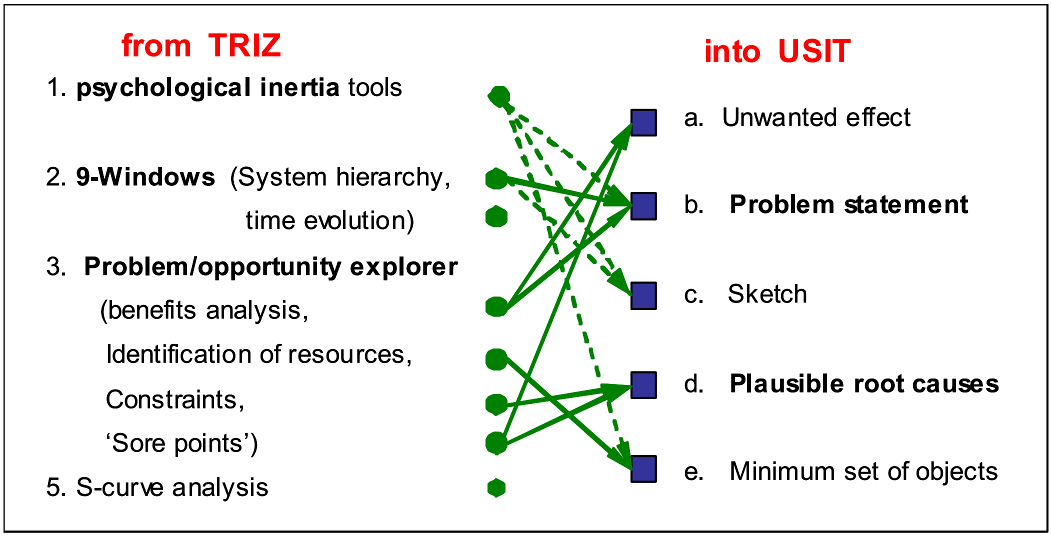
The results of the reorganization are described below by showing the relationships between the current components of USIT in Japan [2] and those of TRIZ in its current form edited by Darrell Mann [1].

4.2 Reorganization from TRIZ into USIT for Problem Definition

USIT methods in the Problem Definition Stage have been organized by Sickafus [7]. The relationships between TRIZ methods in Mann [1] and USIT methods may be summarized as shown in Fig. 7 [11].

USIT in this stage is rather simple and just shows the requirements of information to be obtained by the discussion among the USIT project members. Emphasis is placed on the Problem statement (i.e., statement of the goal of problem solving) and on the Plausible root causes (i.e., understanding of the origins of the problem to be solved).

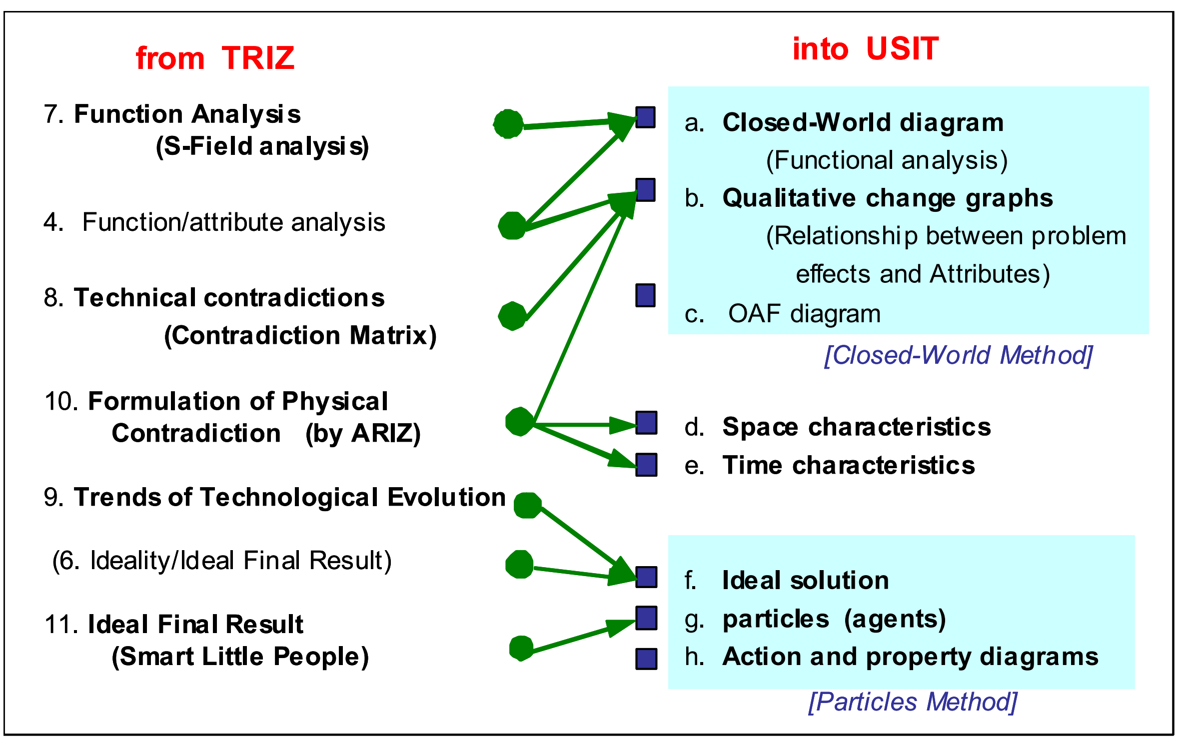
Fig. 7 Reorganization from TRIZ into USIT in the Problem Definition Stage, Ref. [11]



4.3 Reorganization from TRIZ into USIT for Problem Analysis

The relationships between current TRIZ methods and current USIT methods for problem analysis may be summarized in Fig. 8 [11]. Sickafus [7] adopted the Closed-World Method from Israeli SIT and Smart Little People's Modeling from Altshuller, but he refined and enhanced them further.

Fig. 8 Reorganization from TRIZ into USIT in Problem Analysis Stage, Ref. [11]



Introduction by Sickafus [7] of the basic concept of Objects-Attributes-Functions is important here as a new framework in USIT for understanding the systems. Functional Analysis in USIT represents the intended useful Functions among the minimal set of Objects in a concise and disciplined way, while Attributes relevant to the unwanted effect are listed up in the Qualitative-Change Graphs. Characteristics of the system in Space and in Time are also analyzed with simple graphic representations in USIT. All these USIT methods are designed to be simple, thorough, and effective to understand the present system. Su-Field Modeling has been replaced with Functional Analysis.

TRIZ processes for formulating **Technical Contradictions and Physical Contradictions** [1] are intently eliminated in USIT [7]. Instead of formulating Technical Contradictions with Matrix Parameters, the relationships of various Attributes to the unwanted effects are examined so as to reveal the problem situation better. And Contradiction Matrix is not used in USIT because the selection of a limited number of solution-generation operators is not necessary. In USIT, system characteristics are always examined in Space and Time, as the preparation for solving any Physical Contradiction. (With this preparation, Physical Contradictions can be solved in the Solution Generation Stage without explicit formulation.) In this manner, the two most difficult parts in TRIZ for novices (i.e., the formulation of Contradictions in Contradiction Matrix and in ARIZ) are trimmed in USIT.

Altshuller's SLP Modeling [4], which uses agents for stimulating unrestricted imagination, is enhanced into the Particles Method in USIT [7]. Particles Method is unified with imagining Ideal Systems in terms of desirable actions and desirable properties and also with preparation for structural thinking of a hierarchy of solution concepts.

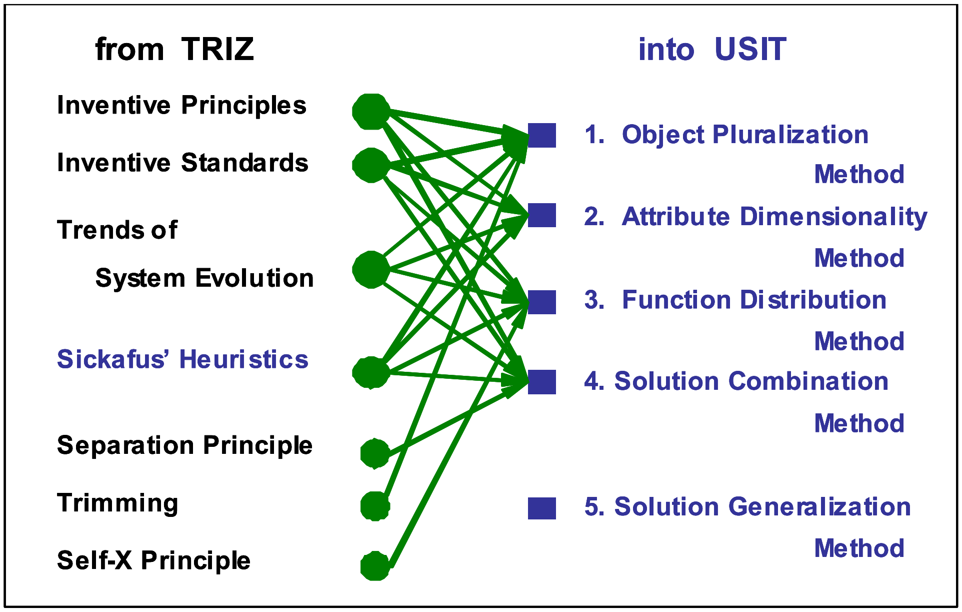
4.4 Reorganization from TRIZ into USIT for Solution Generation

Solution Generation methods of TRIZ were much simplified into only 4 methods in Israeli SIT [14] and then refined into 5 methods by Sickafus [7, 8]. Then the present author reorganized the whole set of TRIZ Solution Generation methods into the framework of USIT Operators, as shown in Fig. 9 [10].

Thus USIT has a hierarchical system of Solution-Generation Operators with 5 principal operators (and 32 sub-operators in total) [10, 16], as explained in Section 3.3. This reorganization is a case of re-classification, unification, and systematization. The whole system of Solution Generation Methods is much simplified and made more effective without loosing power. The following points are remarkable:

- USIT Operators are readily applicable to various possible operands, such as Objects, Attributes, Functions, and Solutions in the present and new systems. Thus the mapping process is straight forward [11].

Fig. 9 Reorganization from TRIZ into USIT for Solution Generation, Ref. [10]



- Resolving contradictions, especially Physical Contradictions, is achieved easily, sometimes even without recognizing them, by use of the USIT Operators of Solution Combination, which correspond to the critical third phase of applying the Separation Principle.
- The USIT Operators of Solution Generalization do not have origins in TRIZ. They are very useful for systematizing possible conceptual solutions.

4.5 Paradigm Shift in the Four-Box Scheme of Creative Problem Solving

The Overall Structure of USIT shown in Fig. 5 may be drawn in a simpler form in Fig. 10 [2]. This figure is simple but yet found significant in many ways, especially urging us a paradigm shift in the Creative Problem Solving in general. Its three main implications are discussed in this section.

(1) Six-Box Scheme as a Refinement of the Basic Four-Box Scheme

The first main implication of the Six-Box Scheme is illustrated in Fig. 11. We may enclose boxes (a) and (b) together into a box of 'Specific Problem', by regarding the Problem Definition process as a refinement inside the statement of the specific problem. We may also enclose boxes (e) and (f) together into a box of 'Specific Solution', because the Implementation process may be regarded as a refinement inside the specific solution. Thus our Six-Box Scheme in USIT (Fig. 10) is interpretable as a form of realization of the basic Four-Box Scheme in Fig. 1.

In relation to Fig. 11, the following points are remarkable:

- For creative problem solving, the specific problem should be prepared as being well-defined, or focused.

Fig. 10 Six-Box Scheme of Creative Problem Solving in USIT, Ref. [2]

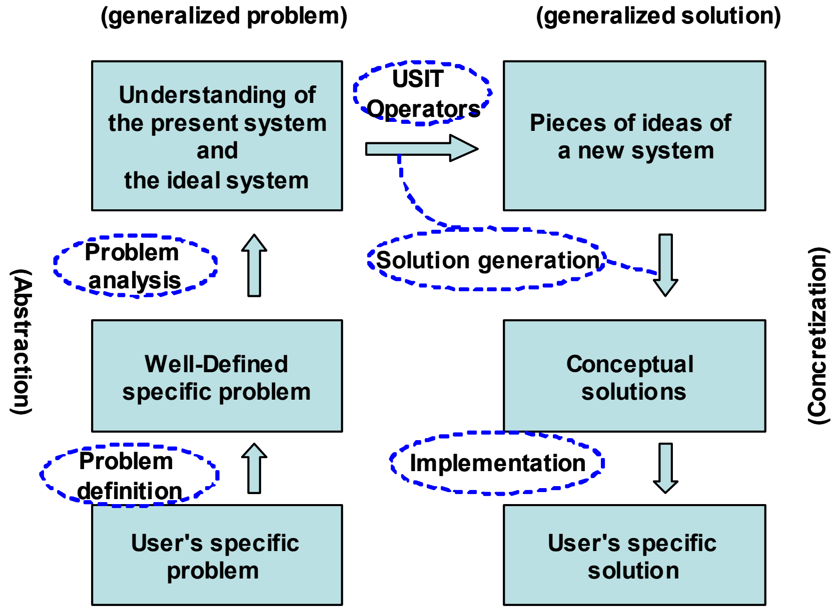
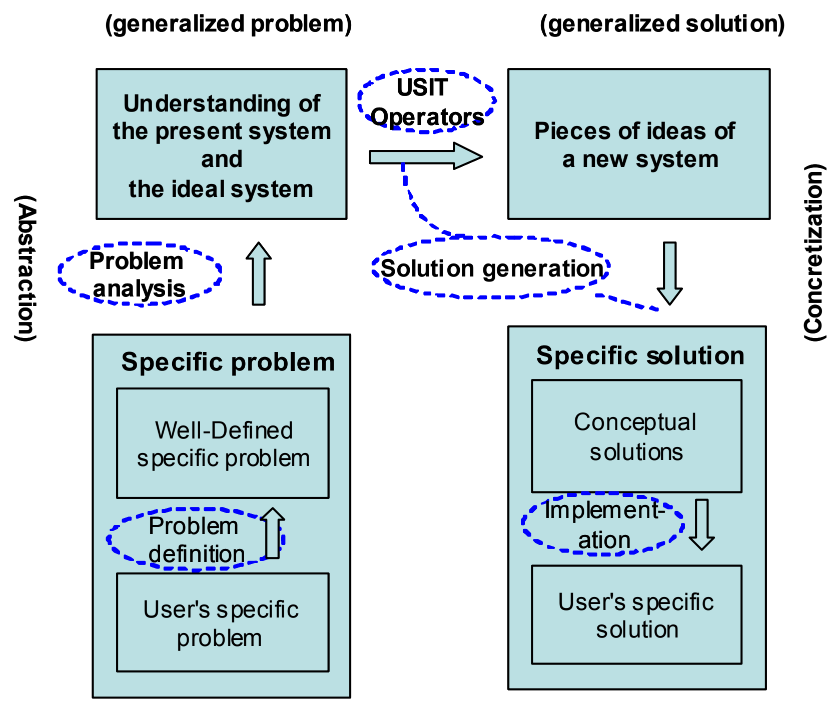


Fig. 11 Six-Box Scheme regarded as a refinement of the basic Four-Box Scheme



- Abstraction should be performed by some standard analysis methods for obtaining good understanding of the present system and the ideal system.
- Understanding of the present system and of the ideal system corresponding to it in an abstract way forms the generalized problem. The framework for understanding a system is

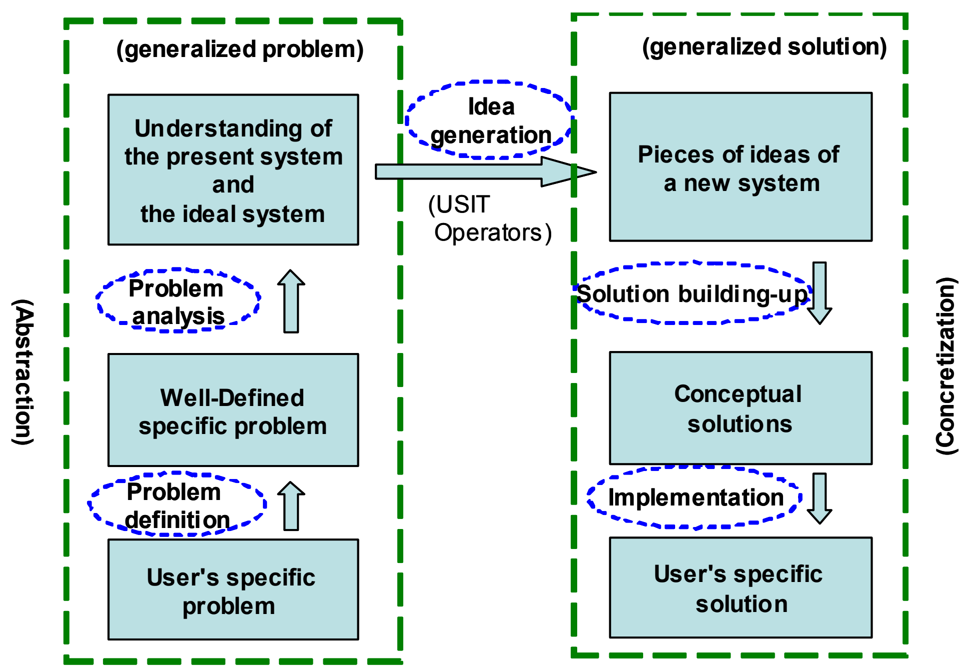
Objects-Attributes-Functions and Space and Time. One should note that the generalized problem does not come from any model in textbooks. Only the framework is given and the contents of the generalized problem come solely from your specific problem.

- A generalized solution is initially obtained as a core piece of new idea in the abstract understanding of a system. Such an idea may be formed by some simple operation(s) on the elements of a system/solution in the abstract world. USIT Operators provide a systematic set of transformation operations in this process.
- Such core pieces of new ideas need to be built into a conceptual solution, which may still be qualitative but worthy of being tried in engineering. This is the initial part of concretization process and needs creative capability in engineering. Conceptual solutions are the goals for creative problem solving methodologies.
- Conceptual solutions need to be implemented and realized into specific solutions in the real world of users. This process may be regarded as refinement of user's specific solutions.

(2) Simple Idea Generation Makes a Jump from Abstraction to Concretization

The second main implication of the Six-Box Scheme is illustrated in Fig. 12. Three boxes in the left column and three other boxes in the right column are grouped, respectively. This figure suggest the following implications:

Fig. 12 Six-Box Scheme where Simple Idea Generation Makes a Jump from the Abstraction Sequence to the Concretization Sequence



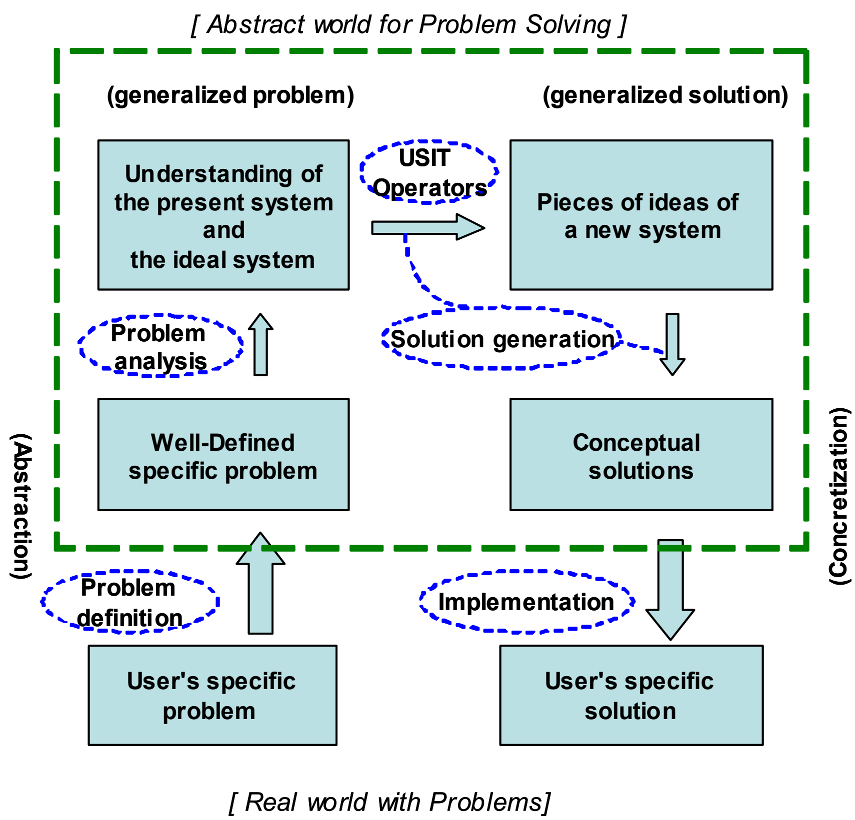
- Boxes in the left column are the sequence of abstraction of the user's problem. The problem should be defined first and then analyzed in a standard way by using the framework provided in USIT. This abstraction process need to be guided by a methodology for creative problem solving, like USIT.
- Boxes in the right column are the sequence of realization (or concretization) of a new idea into real solutions specifically applicable in user's world. An idea need to be built into a conceptual solution and then need to be implemented into real solutions. This realization process should be guided by some creative engineering methodology, depending on the field of application.
- The most critical process is the generation of a useful piece of new core idea. This generation can be effectively supported by USIT Operators. Each operation of the USIT Operators may be small and simple, and the core ideas may also be small and simple (even for great inventions).
- Note that the models of generalized problem-solution pairs in Fig. 4 have disappeared in Fig. 12 (and in Figs. 10 and 11). This means that the essence of Six-Box Scheme of creative problem solving (in USIT) is no longer 'Enforced Analogy'. We no longer need loops of model selection and model-dependent abstraction-concretizations trials.

(3) Abstract World of Problem Solving Placed in the Real World with Problems

The third main implication of the Six-Box Scheme is illustrated in Fig. 13. The upper four boxes are grouped to show that they belong to an abstract world of problem solving in USIT, while the bottom two boxes belong to the real world. This implies as follows:

- Creative problem solving needs to be done in an abstract world where thinking is the powerful and efficient driving force.
- In the user's real world, user's specific problem need to be converted into a Well-defined specific problem (or a focused problem) so as to be handed to the abstract world. This process need to be done in the technical, business, and social context (and evaluated under such criteria) of the real world.
- The upper four boxes are in the abstract world, where the Well-defined specific problem is requested to be converted (finally) into some conceptual solution(s). The resultant conceptual solutions should be creative and applicable in the context of the real world.
- The conceptual solutions need to be implemented into user's specific problems by using available resources and methods in the real world.
- When people want to learn 'success stories' of applying a problem solving methodology, they often want to see implemented specific solutions which have succeeded in real business context.
- The effectiveness or value of a creative problem solving methodology should be evaluated on the basis of conceptual solutions as its output, whereas the effectiveness or value of a case of problem solving is naturally evaluated on the basis of implemented solutions in the real world context.

Fig. 13 Six-Box Scheme places the Abstract World of Problem Solving in the Real World of Problems



- Cooperation between experts in the real world (e.g., engineers, IP people, business managers, etc.) and experts in the abstract world of creative problem solving (e.g. USIT experts) is important for successful problem solving not only at the real-world level but also at the methodology level.
- This point also urges us the combined use of creative problem solving methodology with some other methodologies, especially for supporting the problem definition (e.g., QFD, Mind mapping, etc.) and the solution implementation (e.g., Robust design, CAD/CAM, Digital engineering, etc.).

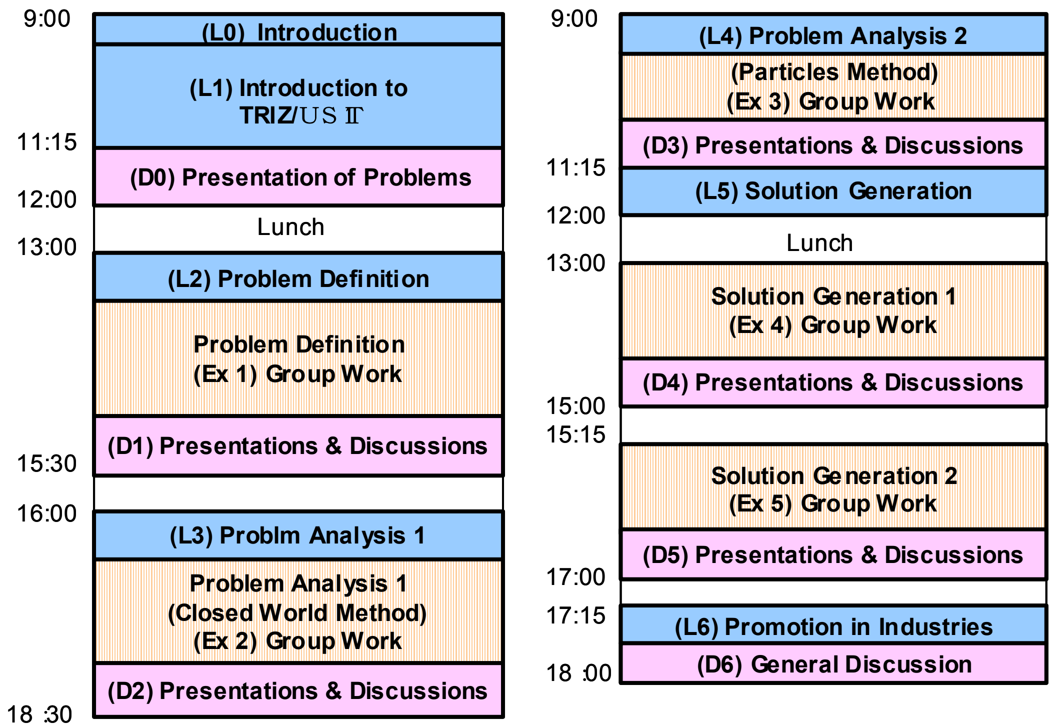
5. PRACTICES OF PROBLEM SOLVING WITH USIT

5.1 Training of USIT in 2-Day Seminars

Most effective way of mastering/training USIT is 2-day Training Seminars instructed by the present author [9]. Typically, an organizer arranges the Training Seminar inside a company, finding 3 real industrial problems to be solved and 15-25 members. The problems should be real, unsolved, and important to solve. Participants should include the engineers responsible for the problems, inside and around the projects, having different background, and managing intellectual properties, and managers, etc. It is desirable to have a few members having some experiences in TRIZ/USIT, but almost all the members may be novices in USIT.

The Training Seminar is carried out with the schedule shown in Fig. 14. After an introductory lecture on TRIZ and USIT for 2 hrs, three real problems are tried to solve in 5 sessions of parallel group practice by following the USIT procedure. Each session contains 3 sub-sessions; they are a short lecture on the process, a parallel group practice, and a meeting for group presentations and discussions.

Fig. 14 Program of 2-Day USIT Training Seminar



In the training seminar, usually 3 to 10 new conceptual solutions are obtained for each real problem. Every participant obtains the experience of really solving a problem with USIT and understands how to apply USIT in the three different cases. Handling with real problems, instead of textbook cases, is important for the participants to be highly motivated, to have real experience of thinking and solving for themselves with USIT, and to realize that the methodology helped them on the problems whose solutions the instructor did not know.

5.2 Case Studies of Promotion and Penetration of USIT in Japan

Since 1999, the present author has been promoting TRIZ/USIT in Japan, by writing many introductory and conference papers, by giving many short lectures and one-day seminars, and 17 times of 3-day or 2-day USIT Training Seminars. The current stage of penetration of TRIZ/USIT in Japan in industries and in academia was reported in Ref. [13]. Very recently, in a public TRIZ/USIT Seminar, five companies (i.e., Hitachi, Japan Railway - East Japan, Fuji Photo Film, NISSAN Motors, and Matsushita Electric Works) presented their experiences of promotion and application of TRIZ and USIT.

In NISSAN Motors, the present author was invited by its IP division to give a 3-day USIT Training Seminar for its Laboratory people in Mar. 2002, and another 2-day Training in Feb. 2003. IP division has been the promoter of TRIZ/USIT in NISSAN as reported in [13]. Then he was invited by an engineering division of NISSAN in Sept. 2003 to give 2-day USIT Training Seminar. In Feb. 2005, Kimihiko Nishimura (for Masahiko Iizumi) of the engineering division publicly reported [18] as: "In our engineering division, 90 % of about 200 engineers already received half-day TRIZ/USIT seminars given by ourselves. TRIZ/USIT has been used regularly in our division for generating ideas and enhancing solutions in the group meetings for IP generation. After defining a problem (by using Mann's Problem Definition Tools), relevant patents are surveyed and summarized in a form of a road map. Then in a month or so, one-day Workshop of the project is held to decide focus areas of the problem on the map and to generate a large number of conceptual solutions to the problems in parallel group discussions using TRIZ/USIT solution-generation methods. Each conceptual solution is written down briefly in a simple format similar to patent proposal. As the result, our division have recently generated patents much more than before in number, in quality, and in strength."

Matsushita Electric Works [19] reported as follows: "Our company tried to use TRIZ with its software tools in a Corporate Laboratory during 2000-2002, but failed. In Dec. 2003, we invited Prof. Nakagawa for a lecture on TRIZ/USIT at a meeting in IP division. Then we have decided to do trials of USIT. Three corporate engineering management divisions (i.e., IP, Technology Management, and R&D Planning divisions) formed the USIT promotion team, and started to organize USIT training at its three Corporate Laboratories. We held 2-day USIT Training Seminars instructed by Prof. Nakagawa three times from Mar. 2004 to Feb. 2005 with mostly laboratory people. Typically, 5-10 conceptual solutions were obtained for each real problem. Evaluation by the participants just after the training may be summarized in Table 1. With these good evaluation, we are going forward to introduce USIT for actual use. We need to bring up our own in-house instructors and to find what types/areas of problems are suitable (or unsuitable) for applying USIT."

Table 1. Evaluation of USIT 2-Day Training Seminars by Participants

Date	Main area	Understanding of USIT	Usefulness to the theme	Recommend USIT to others
Mar. 2004	Mechanical	3.8 points (in 5)	3.1 points (in 5)	71 % of participants
Sept. 2004	IT systems	3.4 points (in 5)	3.0 points (in 5)	69 % of participants
Feb. 2005	Materials	3.5 points (in 5)	2.6 points (in 5)	64 % of participants

5.3 Requirements for Further Development and Penetration of USIT

For further development and penetration of USIT, the following points are currently required:

- Make many real application cases, and publish them as case studies.
- Conduct public USIT Training Seminars, and bring up voluntary engineers in industries and in academia into USIT practitioners.
- Establish practical methods of applying USIT to various types and areas of problems.
- Prepare a collection of instructive examples of applying USIT Solution Generation methods.
- Clarify practical ways of using USIT in combination with TRIZ knowledge-base software tools.
- Develop practical ways of promoting/applying USIT in industries, and spread them widely.
- Develop the capabilities/organizations of consulting with USIT, as well as training with USIT.
- Penetrate USIT into SMEs (small and medium-sized enterprises), and make successful cases of creative problem solving.

As the key trial for fulfilling a number of these requirements at the same time, public USIT Training Seminars are going to be held starting in Apr. 2005. Such public seminars were held by Mitsubishi Research Institute five times from Jan. 2001 to Sept. 2002 [9], but have not been held for these 2 and half years mostly because of the difficulty in obtaining participants who are willing to bring in their own real problems. An arrangement of non-disclosure agreement for 1 year as reported in Ref. [9] seems to be thought still risky for big businesses. Now we are going to hold public Training Seminars under similar non-disclosure arrangement by inviting engineers (mostly from SMEs) with real problem proposals and those (mostly from big businesses) wishing to master USIT without problem proposals.

6. CONCLUDING REMARKS

Overall Structure of creative problem solving in USIT has been built into a Six-Box Scheme in the dataflow representation. User's specific but vague problem (a) are converted into a well-defined user's problem (b), and analyzed to obtain understandings of the present system and of its ideal system (c), then transformed by USIT Operators for Solution Generation into pieces of new ideas for new systems (d), and further built into conceptual solutions (e), and finally implemented into user's specific solutions (f).

Problem solvers are guided all through the USIT procedure in a logical and yet creative way. The difficulty in the multiple-path structure of TRIZ and the ambiguity in the (enforced) analogy have been eliminated. All the methods and knowledge-bases developed in TRIZ have been reorganized, unified, and simplified in this new scheme of USIT. Six-Box Scheme in USIT seems to urge a paradigm shift in creative problem solving. Effectiveness and easiness of USIT have been demonstrated in recent practices in Japanese industries.

REFERENCES 1)

[1] Darrell Mann: "Hands-On Systematic Innovation", CREAX Press, Ieper, Belgium, 2002 (E); Japanese edition, translated by Toru Nakagawa et al., SKI, Tokyo, 2004 (J).

[2] Toru Nakagawa: 'USIT Operators for Solution Generation in TRIZ: Clearer Guide to Solution Paths', presented at ETRIA TRIZ Future 2004 Conference, held at Florence, Italy, on Nov. 3-5, 2004 (E); TRIZ HP Japan, Oct. 2004 (J), Nov. 2004 (R); TRIZ Journal, Mar. 2005 (E).

[3] Genrich Altshuller: "Creativity as an Exact Science", Gordon &Breach, 1984 (E).

[4] Genrich S. Altshuller: "The Innovation Algorithm", Technical Innovation Center, Worchester, MA, USA (1999) (E); Japanese translation from Russian 1st edition, Nikkei BP (1997) (J).

[5] Boris Zlotin and Alla Zusman: "Tools of Classical TRIZ", Ideation International Inc., Southfield, MI, USA (1999) (E); Japanese translation, Nikkei BP (2000) (J).

[6] Yuri Salamatov: "TRIZ: The Right Solution at The Right Time", Insytec, The Netherlands (1999) (E); translated by Toru Nakagawa et al., Nikkei BP (2000) (J).

[7] Ed. N. Sickafus: "Unified Structured Inventive Thinking: How to Invent", NTELLECK, Grosse Ile, MI, USA (1997).

[8] Ed Sickafus: "Unified Structured Inventive Thinking -- An Overview", eBook, URL: http://www.u-sit.net/, (2003) (E); translated by Keishi Kawamo, Shigeomi Koshimizu, and Toru Nakagawa, TRIZ HP Japan, Oct. 18, 2004 (J).

[9] Toru Nakagawa: 'Experiences of Teaching and Applying the Essence of TRIZ with Easier USIT Procedure', TRIZCON2002: Fourth Annual Altshuller Institute for TRIZ Studies International Conference, Apr. 30- May 2, 2002, St. Louis, MO, USA; TRIZ HP Japan, Jan. 2002 (J); May 2002 (E).

[10] Toru Nakagawa, Hideaki Kosha, and Yuji Mihara: 'Reorganizing TRIZ Solution Generation Methods into Simple Five in USIT', ETRIA World Conference "TRIZ Future 2002" held at Strasbourg, France, on Nov. 6-8, 2002; TRIZ HP Japan, Sept. 2002 (J); Nov. 2002 (E).

[11] Toru Nakagawa, Hideaki Kosha, and Yuji Mihara: 'Usage of USIT Solution Generation Methods: A Simple and Unified System of TRIZ', TRIZCON2003, held at Philadelphia, USA, on Mar. 16-18, 2003; TRIZ HP Japan, Jan. 2003 (J); Apr. 2003(E).

[12] Toru Nakagawa, 'USIT Approach in Japan for Simpler and Powerful Process of Creative Problem Solving in TRIZ', ETRIA World Conference "TRIZ Future 2003" held at Aachen, Germany, on Nov. 12-14, 2003; TRIZ HP Japan, Dec. 2003 (E).

[13] Toru Nakagawa: 'Practices of Applying TRIZ/USIT in Japan', TRIZCON2004, held at Seattle, USA, on Apr. 25-27, 2004; TRIZ HP Japan, May 2004 (E); Aug. 2004 (J).

[14] Roni Horowitz: 'From TRIZ to ASIT in 4 Steps', TRIZ Journal, Aug. 2001 (E); TRIZ HP Japan, Sept. 2001 (J). URL: http://www.start2think.com/ (E).

[15] Ed Sickafus: 'A Rationale for Adopting SIT into a Corporate Training Program', TRIZCON99 held by Altshuller Institute, March 7-9, 1999 at Novi, Michigan (E); TRIZ HP Japan, May 1999 (J).

[16] Toru Nakagawa, Hideaki Kosha, and Yuji Mihara: 'USIT Solution Generation Methods: Simplified

System by the Reorganization of TRIZ Solution Generation Methods', Appendix to Ref. [10], ETRIA World Conference "TRIZ Future 2002" held at Strasbourg, France, on Nov. 6-8, 2002; TRIZ HP Japan, Sept. 2002 (J); Nov. 2002 (E).

[17] Victor Fey: 'Why Does TRIZ Fly But Not Soar?', presented at ETRIA TRIZ Future 2004 Conference, held at Florence, Italy, on Nov. 3-5, 2004 (E).

[18] Kimihiko Nishimura: 'Effective Construction of Patent Networks', presented at 'TRIZ/USIT Seminar for Industrial Practices', held by Technology Information Institute, at Tokyo, on Feb. 24-25, 2005.

[19] Kouji Tsuji and Jiro Hashizume: 'Introducing USIT at Matsushita Electric Works', presented at 'TRIZ/USIT Seminar for Industrial Practices', held by Technology Information Institute, at Tokyo, on Feb. 24-25, 2005.

Note: "TRIZ HP Japan" represents: "TRIZ Home Page in Japan", Editor: Toru Nakagawa.
http://www. osaka-gu.ac.jp/php/nakagawa/TRIZ/eTRIZ/ (in English),
http://www.osaka-gu.ac.jp/php/nakagawa/TRIZ/ (in Japanese).

Note: (E): written in English, and (J): written in Japanese.

About the author:

Toru NAKAGAWA: Professor of Informatics at Osaka Gakuin University. Since he was first exposed to TRIZ in May 1997, he endeavored to introduce it into Fujitsu Labs for which he was working. After moving to the University in April 1998, he has been working for introducing TRIZ into Japanese industries and academia. In November 1998 he founded the public WWW site "TRIZ Home Page in Japan" and serves as the Editor. He is currently working to present TRIZ in a simple, unified and yet powerful way for solving real industrial problems and for teaching students. -- He graduated the University of Tokyo in chemistry in 1963, studied at its doctoral course (receiving D. Sc. degree in 1969), became Assistant in Department of Chemistry, the University of Tokyo in 1967; he did research in physical chemistry, particularly experiments and analyses in the field of high-resolution molecular spectroscopy. He joined Fujitsu Limited in 1980 as a researcher in information science at IIAS-SIS and worked for quality improvement of software development. Later he served as a managing staff in IIAS-SIS and then in R&D Planning and Coordination Office in Fujitsu Labs. -- E-mail: nakagawa@utc.osaka-gu.ac.jp