

創造的問題解決の新しいパラダイム

- 類比思考に頼らない USIT の 6 箱方式 -

A New Paradigm for Creative Problem Solving:
Six-Box Scheme in USIT without Depending on Analogical Thinking

中川 徹 (nakagawa@utc.osaka-gu.ac.jp)

大阪学院大学情報学部： 564-8511 大阪府吹田市岸部南 2-36-1

要約： 問題解決のためには、抽象化して考えた後に具体化するという、「4 箱方式」が一般に受け入れられてきた。しかし、知識ベースを活用する類比思考、ヒントから本質を取り出そうとする等価変換理論、多数の技法を持つ故に混乱がある TRIZ などを検討しても、問題解決の基本方式はまだ内実を持っていない。この反省に基づき、USIT (統合的構造化発明思考法) の全体構造をデータフロー図に表現した結果、「USIT の 6 箱方式」を得た。現実世界における問題の定義、明確な分析方法 (抽象化)、「USIT オペレータ」によるアイデアの生成、アイデアを核にして解決策を構成し、さらに現実世界で実現する具体化の過程などを明瞭に記述している。これは、「4 箱方式」にずっと明確な内実を与えた「創造的問題解決の新しいパラダイム」を成すものである。

1. はじめに

科学・技術の広い範囲で、問題解決のための基本的な方式として、「問題を抽象化して考える」という「4 箱方式」(図 1) が受け入れられてきた。そこでは、具体的な問題を「一般化した問題」へと抽象化し、それに対する「一般化した解決策」を得て、それをヒントにして自分の問題に対する「具体的な解決策」を得る。

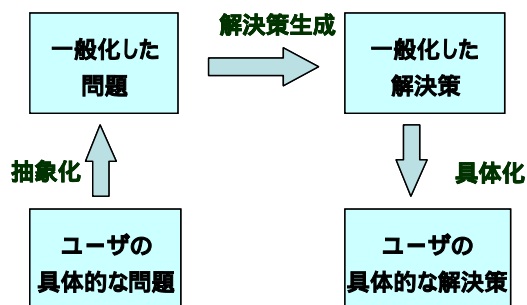


図 1. 問題解決の基本の 4 箱方式

本稿はこの基本方式の内実を吟味し直し、新たに、(技術分野における) 創造的問題解決のための基本方式として、「USIT の 6 箱方式」を提唱する。それは、各「箱」の内容をより明確に規定し、かつ、その間の変換の方法をも具体的に提示している。

この新提案に至るまでに吟味したのは、知識ベースの活用と類比思考による方法、市川亀久

彌の等価変換理論 [1]、Genrich S. Altshuller の TRIZ [2]、および Ed Sickafus の USIT [3] である。本提案は、筆者らによる USIT の改良 [4] を土台として、新しい「創造的問題解決の基本方式」の理解 [5] に至ったものである。

なお、このような基本方式を表現するにあたって、処理方法の表現 (例えばフローチャート) と、各段階で使う/生成する情報内容の表現 (例えばデータフロー図 (図 1 はこの形式)) とを、使い分けることの重要性に注意されたい。

2. 知識ベースの活用と類比思考

まず、従来の諸技法の基本部分を検討し、図 1 の基本方式の実際の状況を吟味しよう。

図 1 の方式で使うための、「一般化した問題」と「その解決策」は、理論、モデル、手本、事例などと呼ばれ、あらゆる分野で作られ、蓄積されてきた。専門分野が分化し、膨大な知識ベースが形成されるにつれて、問題解決の基本方式は、図 2 に示すものに变化した。

すなわち、知識ベース中には「一般化した問題とその解決策」の組 (すなわち、モデル) が多数存在するから、その中から適当な組を見出すことがこの方式の前提となる。

数学や物理のように理論的に確立された分野の限定的な問題なら、その問題領域で確立された方法で抽象化でき、ほぼ確実に「一般化した問題とその解決策」を経て、具体的な回答

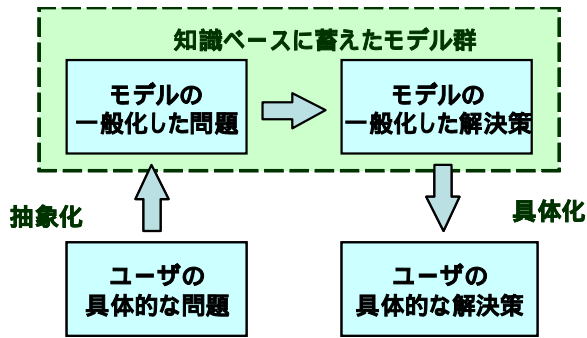


図 2. 知識ベースを使う問題解決の 4 箱方式

(解決策)を見出すことができる。

しかし、(本論で対象とする)技術の未解決問題のように、創造的な問題解決を必要とする場合には、どのモデルを使うとよいのかが自明でない。そして多くの場合に、モデルによって抽象化するべきことが異なる。通常のやり方をいくつか試みてもうまく行かないことが分かったときに、本当の問題にぶつかる。

そのようなときに、いろいろなモデルを(試行錯誤で/適当に)選んできて、それに合うように抽象化(すなわち、分析)することが行われる。何かをヒントとして探す、あるいはたまたま何かのきっかけでひらめきを得たというのが、このモデルの探索に関わっている。

このような形でモデルを探索し、それを使おうとするとき、典型的には「類比思考(Analogical Thinking)」が行われる。まず何らかのヒントとなるモデルを見つける(選ぶ)、そして自分の問題との共通点を見出し、ヒントとしたものの良い点を取り込んで自分の問題への解決策にしようとするわけである。

類比思考の問題点は、ヒントを見つけるための確実な指針がなく、抽象化および具体化のプロセスの明確な方法がないので、すべてが直感

的・感覚的な段階に止まっている。

3. 等価変換理論の吟味

この類比思考を脱して、創造的な問題解決のための方式を創り出そうとしたのが市川亀久彌の「等価変換理論」[1]であった。

その理論は、図 3 のように記号を使って表現され、各記号に多くの含蓄がある(言葉で補われている)。

この理論を、(門外漢であるが故に敢えて)より分かりやすく図的に表現し直してみると、図 4 のようである。これは、使う「情報」を中心に表現したもの(データフロー図)であり、処理の方法は(市川の「等価変換フローチャート」を参考にして)矢印と付随した番号で示している。

等価変換理論では、解決したい問題の視点 V_i と、目標とする機能 c を明確にした上で、その機能を持つ(別分野での)具体例をヒントとして探す。そのヒントの事例 A からその事例に特殊な条件を捨象し、本質的な条件(実行手段) c と実現機能 c を抽出する。この c は、「~を~によって(して)~する」と表現できるべきもので、図 1 の 4 箱方式での「一般化した解決策」(のエッセンス)に対応する。これを核にして、問題の技術分野での知識を導入して、当初の問題を解決できる具体的な解決策 B を構成する。

このように等価変換理論は、具体例 A を抽象化して、本質 c を明確にした上で、実際の解決策に具体化する指針を(考え方のレベルで)与えている。しかし、ヒントになる具体例を探し、それに依拠して考察する点は、類比思考の枠組みが残っているといえる。

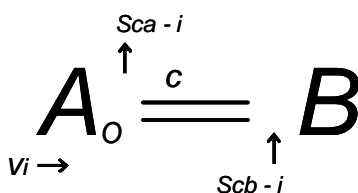


図 3. 等価変換方程式 [1]

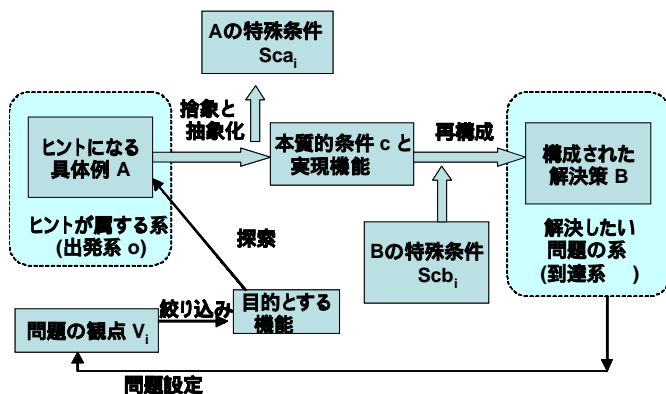


図 4. 等価変換方程式を解釈したデータフロー図 [中川]

4. TRIZ と USIT の基本方式の吟味

TRIZ は、旧ソ連で Genrich S. Altshuller とその弟子たちが開発した、創造的問題解決のための方法論である。彼らは科学技術の知識をさまざまな枠組みで整理しなおした「TRIZ 知識ベース」を作るとともに、多様な創造的問題解決の技法を編み出した [2]。

TRIZ は、発明原理、発明標準解、技術進化のトレンド、などの豊富で有能な知識ベースを持ち、それぞれに適した前処理 (問題の分析/抽象化) の方法をもつ。これらの知識ベースの内容と使い方や諸技法は、TRIZ 専門家たちに共通の理解があり、図 2 がその問題解決の基本方式であると共通に理解されている。

ところが、TRIZ での問題解決の「全体プロセス」(すなわち、全体のフローチャート表現) は、百家争鳴の状態にある。いくつもの「分析法と知識ベースの組」を逐次的に積み重ねていくやり方も、また、どの場合にはどの組を使えと分類していくやり方も、非常に複雑になる。それらはエキスパートが用いると強力でも、一般の技術者たちには容易に受け入れられない。

TRIZ での問題解決の方法論を「データフロー図」で表現しようとする、(図 2 の複数モデルを展開した複線の構造を持ち) すっきりした全体構造を持たないことが分かった。TRIZ の方法論の問題点の真因がここにあると、筆者は指摘した [5]。

TRIZ を消化・再編して、もっと分かりやすく有効な問題解決の方法論を作ろうとしたものの代表が、USIT (統合的構造化発明思考法) である。フォード社の Ed Sickafus が開発し [3]、その後筆者が改良を続けている [4、5]。

Sickafus は、問題解決の全体プロセスを、問題定義、問題分析、解決策の生成という明確な 3 段階にし、それぞれの段階に簡潔な技法を導入した。TRIZ の伝統であった知識ベースに依拠することを止め、グループの共同作業での考える手順をガイドすることによって、創造的な問題解決の方法論を作り上げようとした。

問題分析のために、「オブジェクト - 属性 (性質) - 機能」の概念を明確にして、現システムの機能分析と属性分析のやり方を明示している。また、空間と時間の特徴を考える。そして、理想の結果をまずイメージして、それを「魔法の Particles」に託して考えるという技法

(Particles 法) を導入している。

解決策生成の段階に対しては、(先行するイスラエルの SIT 法をベースにして) 簡潔な 5 種の方法を用いる。

筆者らが USIT を改良したのは、この解決策生成法に対してである [4]。TRIZ が蓄積した豊富な知識ベース中のさまざまな解法の要素をすべてばらばらにして、USIT の 5 種の解法に仕分けた。そして、豊富になった USIT の 5 種の解法の中をさらに分類して、階層的な体系を作った。それらは、5 種 32 サブ解法からなり、各サブ解法は簡潔な指針 (ガイドライン) を持つ (例えば、オブジェクト複数化法の中のサブ解法の一つでは、「オブジェクトを複数の部分に分割し、それらの属性を互いに变化させて、統合して用いよ」という)。これらの解法を、「USIT オペレータ (演算子)」と呼ぶ。

USIT では、その開発の初期から、全体プロセスをフローチャートで表現し、実地の問題解決においても活用してきた。

しかし長い間、Sickafus も筆者も、「USIT の問題解決の基本方式」という意識を持たなかった。前述のように、「TRIZ が簡潔な全体構造を持たないために、全体プロセスに混乱を生じている」ことを理解した後に、USIT の全体構造をデータフロー図に描いてみた。

そして、「USIT の全体構造図 (データフロー図)」が、新しい含蓄を持つことを理解するに至った。それが、本稿でいう「創造的問題解決の新しい基本方式 (新しいパラダイム)」という理解である。その内容を以下に示す。

5. USIT における問題解決の全体構造

まず USIT の全体プロセスをフローチャートの形式で図 5 に示す。

一つ一つの箱は処理のプロセスを表し、(ここで説明する余裕はないが) 具体的な技法で裏打ちされている。解決策生成の段階には、5 種の解法があり (3 種と 2 種とに分けて表現している)、それらを (特に順序を問題にせず) 繰り返し適用することによって、解決策のアイデア (の断片) が多数得られる。そして、そのアイデアをさらに技術的に考察することによって、解決策コンセプト (すなわち、概念レベルでの解決策) を得る。

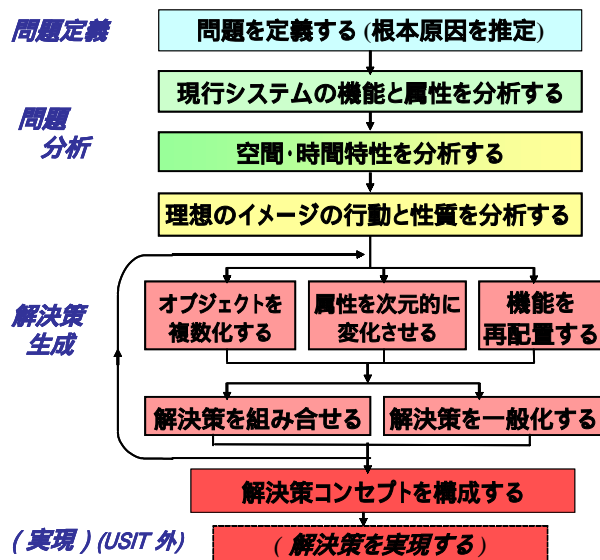


図 5. USIT の全体プロセス (フローチャート)

USIT の全体プロセスについて、その各過程で入出力として使う「情報」に注目して、データフロー図の形式で描いたものが、図 6 である。

図 6 で、箱内には「情報」を記入し、それらの間の矢印とそれに付随した楕円で「処理方法」を記述している。この図の 6 箱の内容をまず説明する。

(a) 第一の「ユーザの具体的問題」は、現実の世界で何か問題があり、解決する必要があると、ユーザが考え始め、そこで入手した (入手可能な) 詳細だが未整理な情報を表す。

(b) 「適切に定義された具体的問題」とは、USIT で扱えるように絞り込んだ一つの問題の

情報をいう。望ましくないこと (問題点) 一つ、解くべき課題の定義 (問題宣言文)、メカニズムを示す簡潔な図、推定される根本原因 (複数可)、問題を構成する最小限のオブジェクト (構成要素) の組、という情報から構成される。

(c) 第 3 の箱は、問題を分析した結果の情報である。まず、現在のシステム (問題対象のシステム、または現在技術で想定されるシステム) の機能分析の結果として、各構成要素の機能関係と全体のメカニズム (そしてその欠陥) の情報を得る。また、属性分析により、問題を起こすことを増大させる属性 (要因) と減少・抑制させる属性 (要因) とを知る。さらに、問題の状況を考察して、空間的な特徴、および時間的な特徴に関する情報を得る。またさらに、理想のイメージを分析し、本来どのようにあってほしいかの方向付けの情報を得る。(実現法は未知だが) システムの望ましい振る舞い (行動)、および持っているといだらうさまざまな望ましい性質に関する情報を含む。

(d) ついで、「新システムのためのアイデア」というのは、新しい解決策のためのアイデアの断片であり、組み込み方や実現可能性などはまだ吟味していない段階のものである。そのようなアイデアが孤立して存在しているのではなく、(c) で得たすべての情報の中に位置付けられ、「新しい変化の核」として現れたことに意義がある。USIT では、多様な「USIT オペレータ」を作用させることにより、これらのアイデアを多数生成できる。

(e) 「解決策のコンセプト」は、上記のアイデアを核にして、ひとまとまりの新しい解決策として概念レベルで構成したものである。この構成には問題の分野に関するそれなりの技術知識が必要である。ただし、定性的な概念レベルでの解決策であり、定量的な検討や詳細の設計などはこの後のことである。

(f) 「ユーザの具体的解決策」は、実地の問題に対して実施するレベルの解決策であり、新しい製品やその部品、実際の製造装置や製造プロセスなどに組み込まれる解決策である。ただし、これは、「創造的問題解決の技法」としての USIT の範囲を越えたものであり、USIT にとっての最終成果は (e) の段階である。

この図 6 が USIT の全体構造をデータフロー図の形式で表現したものであり、これを「USIT の 6 箱方式」と呼ぶ。その特徴は、各箱で示さ

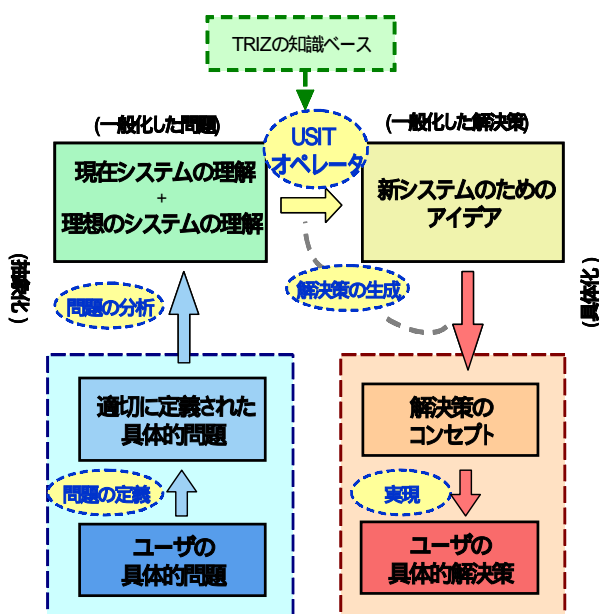


図 6. USIT の問題解決のデータフロー図 = 創造的問題解決の 6 箱方式

れている「情報」の内容が(図1のものに比べてはるかに)しっかりと規定されていることである。また、これらの情報を確実に獲得していく方法を、USITはきちんと示している。

6. 「USITの6箱方式」が意味すること

USITにおける問題解決の全体構造を図6のように書き出した結果、それが新たにさまざまなことを教えてくれていることが分かった。それらの理解を以下にまとめて述べる。

第一の理解は、一般に、処理方法を中心に記述したもの(フローチャートなど)よりも、入出力に使う情報を中心に記述したもの(データフロー図など)の方が、根源的で重要だということである。

入出力の情報の規定は、処理法に対する制約条件と目標仕様を規定する。それらの規定を満たせば、処理の方法はいろいろな代案があってもかまわない。処理方法をいろいろ試して改良していくと、フローチャートは変化するけれども、データフロー図は一貫しているということが起こる。だから、大きな段階を指定し、各段階ごとに必要とする「情報」を規定することが最も重要であり、それを(全体構造、あるいは)「基本方式」と呼ぶ。

第二の理解は、図6の下部で点線で囲んだようにして、全体で4箱にして考えることである。左下の点線の箱は、問題解決の前提段階として「ユーザの具体的問題」という段階の漠然とした情報から、もっと絞り込んで解くべき問題を明確に定義することの必要性を示す。また、右下の点線の箱は、創造的問題解決の直接の成果物としての解決策は、まだ概念レベルであり、それを実際の解決策に実現する後続プロセスの必要性を示している。

これらの点線の箱で考えると、図6は「4箱の基本方式」(図1)によく対応する。図1では非常に一般的な語で漠然と述べているのに対して、図6では、箱内の情報についても、矢印の処理方法についても、しっかりした内実をもって記述されている。すなわち、図6は、「創造的問題解決のための基本方式」の一つをはじめ明確に記述したといえる。

第三の理解は、図1での「一般化した問題」という情報が、図6ではユーザの具体的な問題を標準的な方法で分析した結果として得られることである。分析法は一般的だから(USIT

の)教科書で学ぶことができ、一方「一般化した問題」の中身は、自分の問題から個別に得られる。それは、図2のような知識ベース(教科書など)中のモデルではないし、図4のようにヒントにした例から抽出するものでもない。

第四の理解は、図1での「一般化した解決策」が、図6では、必ずしもまとまった(模範的な)解決策ではなく、まだアイデアの断片(一つのひらめきのエッセンス)の段階にあることである。ここで注目すべきは、図2の「モデルの一般化した解決策」は、等価変換理論においては「ヒントになる具体例A₀」とみなすべきものである。そして、ヒントから抽出した「本質的条件と実現機能c」に対応するのが、USITの図6の「新システムのためのアイデア」であり、「USITオペレータ」によってそれが「裸で出てくる」のだといえる。

アイデアの段階から、「解決策のコンセプト」の段階への変換を、等価変換理論では、「再構成」として記述し、「Bの特殊条件」を使うのだと表現している。図6での考え方もほぼ同様であるが、この変換に必要な情報の大部分は、USITでは分析結果(図6の左上の箱)としてすでに用意できているのである。

第五の理解は、図7に示すものである。

6箱のうちの下4箱は、社会、ビジネス、技術などを含めた「現実の世界」に属すべきものである。現実にはいろいろな問題がある中で、どの問題の解決に取りかかるかは、(技法の世

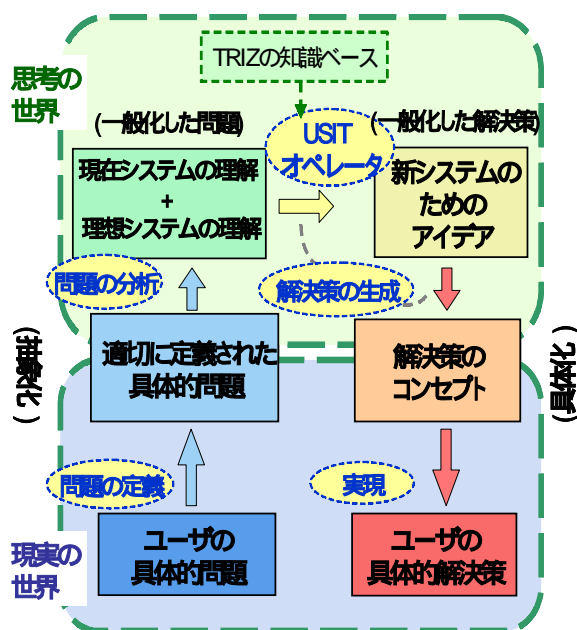


図7. 現実の世界と思考の世界の役割分担

界ではなく) 現実の世界で判断して絞り込むべきである。ただ、それが往々にして不十分だから、USIT では最初に「問題定義」の段階を設けて絞り込みを行うとともに、問題解決グループの意志統一を図っている。また、解決策のコンセプトは、社会・ビジネス・技術の諸条件をクリアするように、現実世界での実現の過程が必要になる。

図7の上半分の4箱は、(USITによる) 問題解決のための「思考の世界」である。そこでは、種々の制約を一旦横に置いて、できるだけ自由に柔軟に考えるとよい。中段の2箱が、両世界での情報を受け渡すインタフェースである。

第六の理解を、図8に示す。左側の3箱が、自分の問題を抽象化(分析)する過程を示し、その結果として、現在と理想のシステムの(機能、属性、空間、時間などの用語による)理解が生成される。この過程は技法がリードするが、問題の技術的な理解が必須である。右側の3箱は、アイデアの核を得て、それを解決策のコンセプトにし、実用の解決策にまで具体化する過程である。ここでは、問題解決技法によるリードはごくわずかであり、技術的な知識・力量の役割が主要部を占める。

現在のシステムの理解から、新しいシステムのアイデアに移行(飛躍)するのを助けているのは、「USITオペレータ」である。その全体はしっかりした体系を成しているが、実際に核になる一つのアイデアを生み出すのは、ごく簡単

な一つのオペレータ(例えば、オブジェクトの分割、時間による組合せなど)である。大発明でも、その核は小さなアイデアである例は多々知られている。USIT オペレータの適用が、技術者たちに新しいシステムの核となるアイデアをつぎつぎとひらめかせる。しっかりと分析した後で、このような飛躍を誘発していることが、この基本方式の神髄である。

7. 結論:創造的問題解決の新パラダイム

以上6節で述べた含意を含めて、図6に示す「6箱方式」が、「創造的問題解決の基本方式」の一つを示しており、本論文はこれを「新しいパラダイム」として提唱する。この方式を実施する具体的な方法が、USITである。

参考文献:

- [1] 市川亀久彌、『創造性の科学 - 図解・等価変換理論入門』、日本放送出版協会、1970年。
 - [2] 『TRIZ 実践と効用 (1) 体系的技術革新』、Darrell Mann 著、中川徹監訳、創造開発イニシアチブ刊、2004年6月。
 - [3] 『USITの概要 (USIT eBook)』、Ed Sickafus 著、川面恵司・越水重信・中川徹訳、TRIZ ホームページ、2004年9月掲載。
 - [4] 「TRIZの解決策生成諸技法を整理してUSITの5解法に単純化する」、中川徹・古謝秀明・三原祐治、ETRIA 国際会議、ストラスブール、2002年11月; TRIZホームページ、2002年9月。
 - [5] 「TRIZ/USITにおける創造的問題解決のためのデータフローの全体構造」、中川徹、TRIZCON2005、デトロイト、2005年4月; TRIZホームページ、2004年5月掲載。
- 注: 『TRIZ ホームページ』、中川徹 編集、URL: <http://www.osaka-gu.ac.jp/php/nakagawa/TRIZ/>

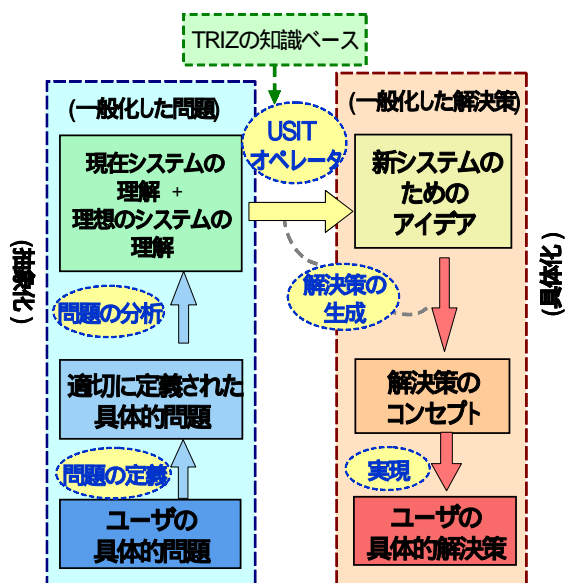


図8. 抽象化、アイデアの生成、具体化