

問題解決の身近な事例 と USIT 法のまとめ

目標: いままで学んだ問題解決の方法を概観する。身近な問題に対して USIT を適用した事例を取り上げ、その上で USIT 法による問題解決の方法をまとめる。

前回: 解決策の生成法 (3) 解決策を生成する方法の体系 (USIT)

目標: TRIZ をやさしく・使いやすくすることを目指した USIT 法における解決策の生成法を学ぶ。特に、USIT の「解決策生成技法の体系」を学び、その考えかたと使い方を例を通して学ぶ。単なる「発想法」でないアイデア生成の考え方の本質を理解する。

要点: USIT 法による解決策生成の例として、「額縁掛けの問題」を例に、その思考過程を示した。特に、一つの良い解決策に至るのにいろいろな(標準的な)解決策生成法が使えることを示した。

TRIZ の各種の解決策生成法 (特に、発明原理、発明標準解、進化のトレンドなど、膨大な知識ベースをなしているもの) をばらして、USIT の解決策生成法の体系に整理し直して、新たに得た「USIT の解決策生成法の体系」を示した。

USIT の解決策生成法を、額縁掛けの問題に一つ一つ適用した例を示し、その考え方を例示した。

1. 問題解決の全体のプロセスと、解決策の生成法の種々のやり方 (まとめ)

1.1 問題解決の全体プロセスと 解決策生成の過程

いままで説明してきたように、問題解決は基本的にはつぎの3段階で行われる。

- (1) 問題を定義する
- (2) 問題を分析する
- (3) 解決策を生成する

これらの段階で、何をやる必要があるか、特にどんな情報を獲得する必要があるか? これらをまとめるとおおよそつぎのようである。

(1) 問題を定義する:

何が困ることかをはっきりさせ、選択する。
困ることが起こる根本原因は何かをはっきりさせる。
結局、何を解決したいのか (すなわち、課題) を明確に述べる。

(2) 問題を分析する:

現在のシステムはどんなメカニズムで働いているのか (機能分析)、
困ることを起こしているのはどのような性質 (属性) か、また
それを抑制するのに関係するのはどのような性質 (属性) か?
空間的な特徴はどんなことか?

時間的な特徴 (変化のしかたなど) はどんなことか?
問題に潜む根本の矛盾 (根本の困難点) は何か?
理想の結果はどうあって欲しいのか?
理想のシステムはどのような振る舞いが望ましいのか?

(3) 解決策を生成する:

問題を解決した新しいシステムの核になるアイデアを見つける。
そのアイデアが問題を解決できるように、現在のシステムの修正・改造法を考える。
新しいシステムの解決策のコンセプトをつくりあげる。
新しい解決策を実地に設計し、実験し、製作する。
新しい解決策を実際の製品やプロセスとして実現する。
さらに新しい解決策を、商品やサービスにし、社会的に普及させる。[技術革新]

これらのうちで、(2) 問題を分析する段階の方法は比較的明確で、一般的になってきている。

一方、(3) 解決策を生成する段階では、

アイデアが出た後の過程は、工学の各専門分野でよく確立されているけれども、
最初の「アイデアを見つける」過程がいままで不明確であった。
(基本的には「ひらめき」と呼ばれ、偶然のきっかけとヒントを頼りにしていた。)

つきつめると、「発明の難しさ」、「問題解決の難しさ」、「技術革新の難しさ」は、
この「解決策の核になるアイデアを見つける」ことの難しさであるといえる。

[注: 近年、技術革新(イノベーション、Innovation) の最重要課題は、「新しい技術を創ること
(= 発明)」ではなく、「社会のニーズに答えるもの(技術やサービス)を創り、普及させる
こと」であると、認識されてきている。
この観点からは、社会のニーズを的確に捉えることが最重要の課題である。]

1.2 「解決策の核になるアイデアを見つける方法」の流れ

この難しさは、昔からの懸案である。

この過程を乗り越えるために、おおまかにつぎのような歴史的な流れがある。

(a) 「ひらめき」を待つ。

環境を整え、心理的なリラックスを図るなど
==> 本講義の 第 4 回参照

(b) 「知識ベース」を蓄積し、「ヒント」として提供する/使う。

科学技術全体で、膨大な知識を蓄積して、それを使おうとしてきた。
TRIZ は、科学技術知識をより使いやすくした知識ベースを作って、使おうとした。
==> 本講義の 第 10 回参照

(c) 経験的な「発明のノウハウ」を蓄積し、使う、トレーニングする。

多様なノウハウ、「逆転の発想」「頭を柔軟にする方法」など。

(d) 発明の事例の分析から、発明の原理、発明の方法を体系的に作り出す。

TRIZ は、「発明原理」、「発明標準解」、「物理的矛盾を解く分離原理」などを作った。
==> 本講義の 第 11 回参照

USIT は、TRIZ の諸方法を総合して

「解決策生成法の体系 (USIT オペレータ)」を作った。

==> 本講義の 第 12 回参照。

この USIT の解決策生成法の体系 (USIT オペレータ) は、

解決策生成法に関して、全体を総合し、かつすっきりとまとめたものである。

それが有効なのは、問題定義と問題分析の段階とよく対応しているからである。

2. 身近な問題での問題解決の事例

以下には、諸君の先輩たちの卒業研究での問題解決の事例を紹介する。

身近な問題でも、しっかり深く考えることにより、いろいろな新しい考え方ができることを示す。

事例 (1) 「裁縫で針より短くなった糸を止める方法」

2006 年 2 月 下田 翼 大阪学院大学 情報学部 卒業研究

『インターラボ』2007 年 1 月号、中川 徹

『TRIZ ホームページ』掲載、2007 年 1 月。

事例 (2) 「ホッチキスの針をむしやげなくする方法」

2004 年 2 月 神谷 和明 大阪学院大学 情報学部 卒業研究

『インターラボ』2006 年 5 月号、中川 徹

『TRIZ ホームページ』掲載、2006 年 5 月。

事例 (3) 「書店で万引きを防ぐ方法」

2004 年 2 月 林 尚也、大阪学院大学 情報学部 卒業研究

『インターラボ』2006 年 6 月号、中川 徹

『TRIZ ホームページ』掲載、2006 年 6 月。

事例 (4) 「オートロックドア方式のマンションで不審者の侵入を防止する方法」

2007 年 2 月 藤田 新、大阪学院大学 情報学部 卒業研究

2007 年 8 月 中川 徹、藤田 新、第 3 回 TRIZ シンポジウム 発表

『TRIZ ホームページ』掲載、2007 年 9 月。

『TRIZ ホームページ』 www.osaka-gu.ac.jp/php/nakagawa/TRIZ/

学生による学生のための TRIZ ホームページ: www.osaka-gu.ac.jp/php/nakagawa/TRIZ/TRIZ-st/

1. USIT (統合的構造化発明思考法) の全体像

USIT (Unified Structured Inventive Thinking, 統合的構造化発明思考法)

1995年にFord社(米国)のEd Sickafusが開発したもの。
彼は1993年にイスラエルのSIT法(Systematic Inventive Thinking)のセミナーをたまたま聴いた。
そして、彼らをフォードに招いて1週間のセミナーをした。
その後さらに調査し、改良してFord社の研修プログラムを1995年に始めた。
イスラエルのSIT法はTRIZを簡略化したもの(1980年代)。
USITはSITにいろいろな改良をし、新しいアイデアを加えて統合している。

また、1999年に中川がSickafusのUSITトレーニングセミナーでUSITを習得し、その後日本においてUSITを改良・発展・普及させた。
現在は、日本がUSITの中心。

参考文献:

中川 徹: 「連載: USIT 入門: 創造的な問題解決のやさしい方法」, 機械設計 (日刊工業新聞社刊)、2007年8月号~12月号; 『TRIZ ホームページ』掲載。

また、USITに関する中川の論文が『TRIZ ホームページ』に掲載してある。

中川 徹 (大阪学院大学)、古謝秀明、三原祐治 (富士写真フィルム)、「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」、ETRIA国際会議: TRIZ Future 2002, ストラスブール (フランス), 2002年11月6~8日;
和訳: 『TRIZ ホームページ』, 2002年9月掲載。

中川 徹: 「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 -」, 日本創造学会第27回研究大会、2005年10月29-30日、学術総合センター (東京)

2. USITの全体構造 (データフロー表現)

現在のUSITの特長は、
その全体構造がデータフロー表現で明確に示されていることである。

「データフロー表現」とは、情報科学 (ソフトウェア工学) で明確になった概念で、
入力として必要なデータ (情報)、中間に得られるデータ (情報)、そして、
処理の結果出力として得られるデータ (情報) を明確に記述することによって、
その処理の全体を記述することを特徴とする。

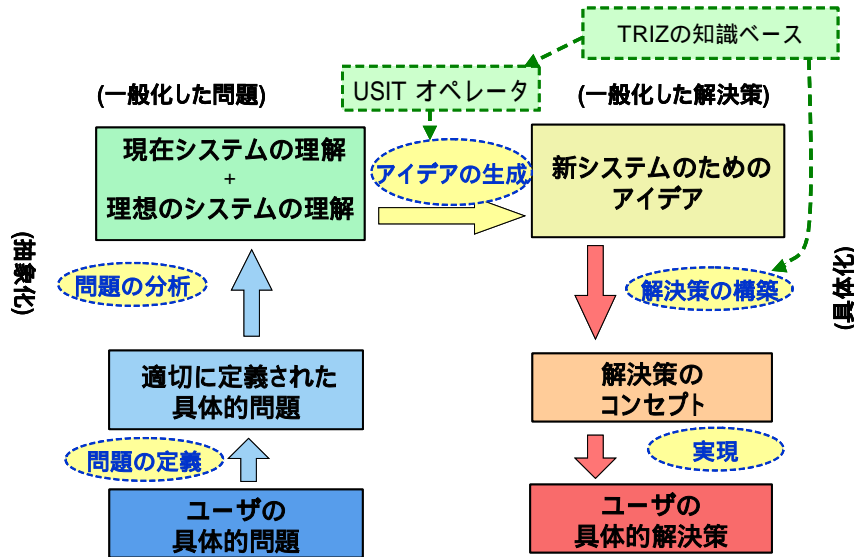
注: 一方、一般的にしばしば用いられる「フローチャート」は、
一つ一つの処理の要素と、その実行の順番や実行の条件を明確にすることにより処理の全体を表現しようとする。

この際、各処理要素で必要とする入力/出力のデータ (情報) は、
言及されることはあっても明確には表示されない。

情報科学でよく知られていることは、

データフロー表現の方が表現としてより安定であること、である。
 (すなわち、同じ情報を得るのに、方法は複数あることがある。
 大事なのは、どんな情報を得たいかである。)

創造的問題解決の新しい方式 (USITの「6箱方式」)

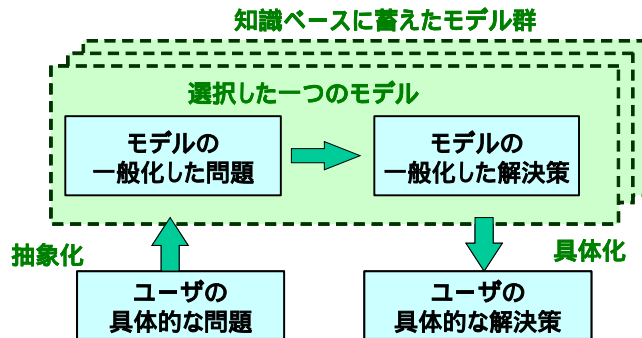


この「6箱方式」は、非常に重要な概念である。

従来、一般的に知られていたのは、次のような「4箱方式」であった。

問題解決の基本的な方法 (パラダイム)

(伝統的)TRIZの基本的な方法 = 科学技術の基本的な方法



多数のモデル => どう選択するのか? どう抽象化するのか?

この「4箱方式」では、「一般化した問題」は、「モデル」に属していて、その「モデル」ごとにいろいろな表現を取る。

ユーザの具体的な問題からの「抽象化」も、実はその「モデル」へのマッピング (写像) であった。

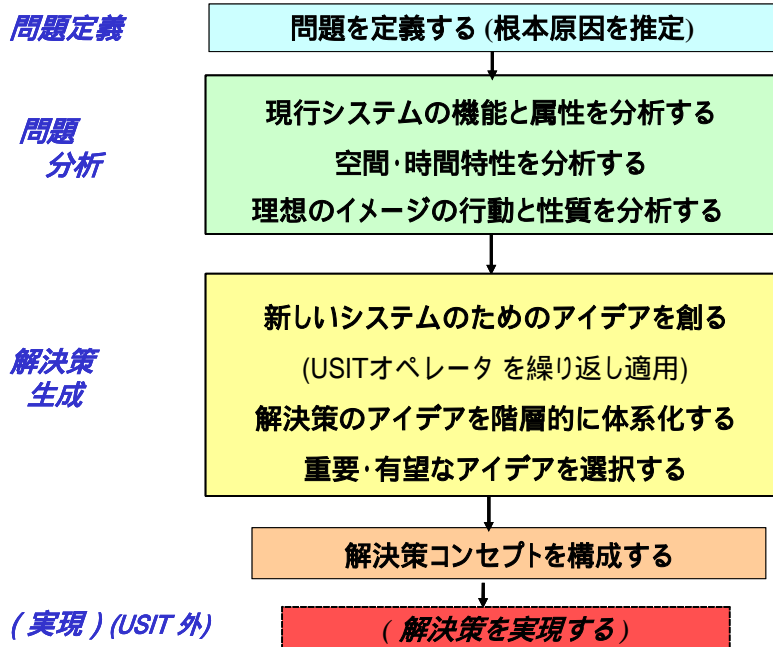
「6箱方式」は、具体的な問題を標準的な方法できちんと分析することにより得る。現在のシステムの理解のしかた (用語や概念) は明確になっている。

3. USIT法全体のフローチャート

USIT法の全プロセスはつぎのフローチャートで表わされる。

USITの全体プロセス (フローチャート)

改良: 中川
2007.12



全プロセスは、3段階 (問題定義、問題分析、解決策生成) からなり、それぞれに明確でわかりやすいガイドラインが与えられている。

4. USITの問題解決のプロセスの 問題定義段階

(本講義 第3回 §7, 第6回 §1.2, §4.1-2 参照)

望ましくない効果: 困ること (望ましくないこと) を一つに絞って採り上げる。

問題(課題)定義文: 解くべき課題を1-2行の文で定義する。

図解: 問題状況を理解するための簡明な概念図

根本原因: 問題 (困ること) を生じている根本の原因と考えられるもの (複数可)。

オブジェクト群: 問題のシステムを構成するオブジェクトを列挙する。

例: (再掲: 本講義 第3回 §7)

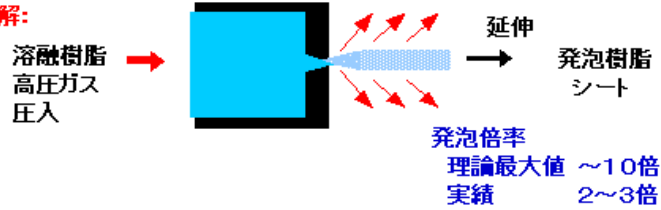
USIT法の適用例(2) 樹脂シートの発泡倍率の増大

1999.3.12 USIT研修にて 中川

USIT法の適用プロセス(1)

問題設定: 高圧でガスを溶解したポリマーから、発泡樹脂シートを押し出し成形する際の、発泡倍率を向上する

図解:



根本原因: ガスが表面から逃げる。
発泡セルが大きくならない。

オブジェクト: 発泡樹脂, 溶融樹脂, ガス, ノズル, (空気)

5. USIT の問題分析段階

(本講義 第8回 §4, 第9回 §2.3, §4)

(a) 閉世界法: 現システムを「オブジェクト-属性-機能」の概念をベースに分析する。

オブジェクト: システムを構成する「物」(実体) (「情報」もオブジェクト)

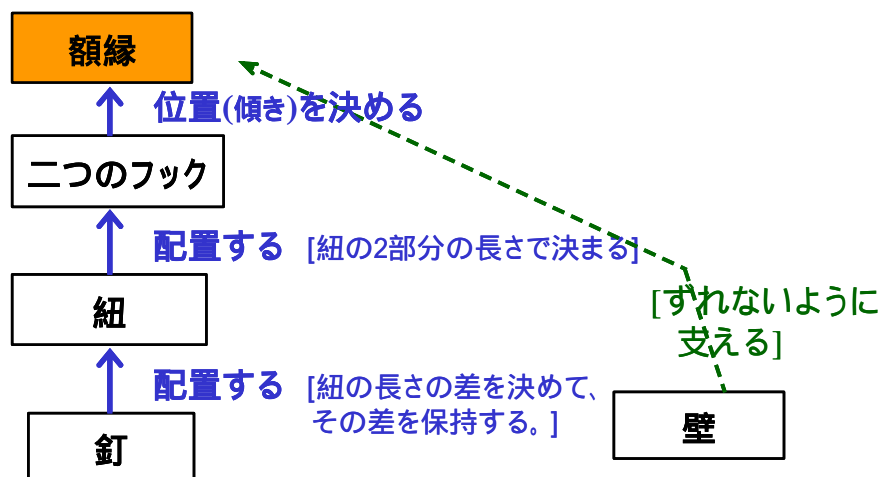
属性: オブジェクトの「性質」のカテゴリ (性質の「値」ではない)

機能: オブジェクトとオブジェクトの間に働く「作用」であり、
オブジェクトの属性を変える/不変に保つ。

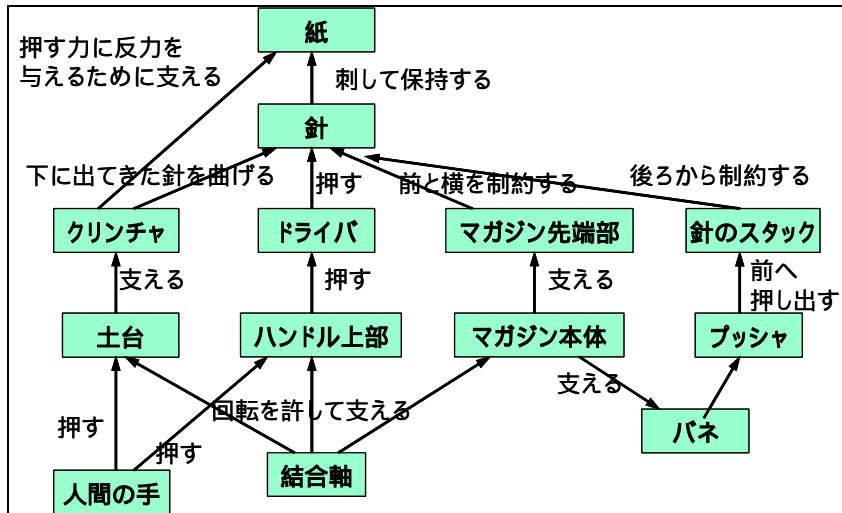
(a1) 閉世界ダイアグラムを構築する:

現システムの本来のデザインを機能分析する。

例: 「額縁掛けの問題」の機能分析 (閉世界ダイアグラム) (再掲: 第8回 §4.2)



例: ホッチキスの機能分析の図

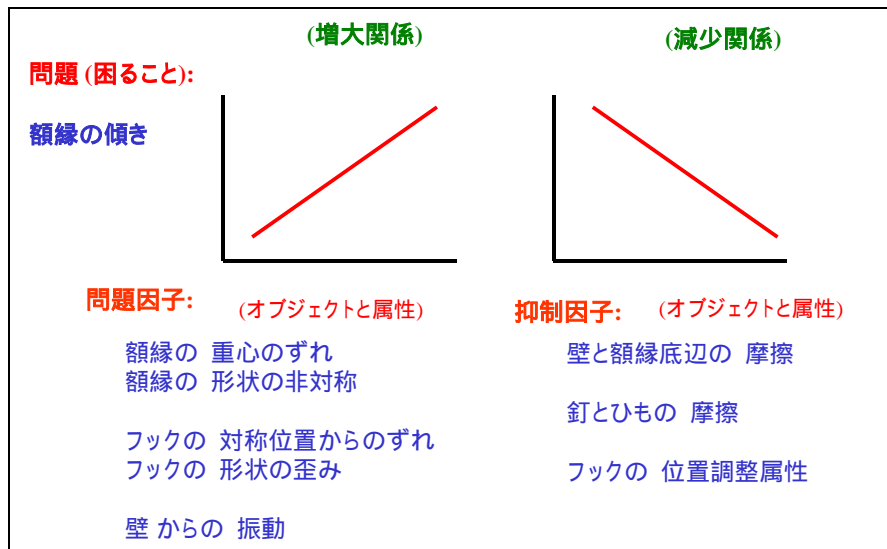


(a2) 定性変化グラフを構築する:

問題 (困ること) に関わる属性を分析する。

問題 (困ること) を増大させる属性と、抑制する属性を列挙する。

例: 額縁掛けの問題の定性変化グラフ (再掲: 第 8 回 § 4.3)



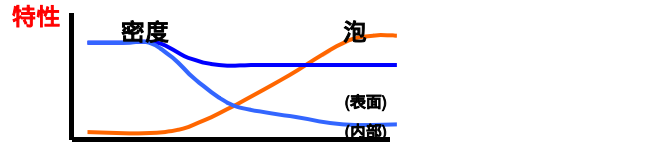
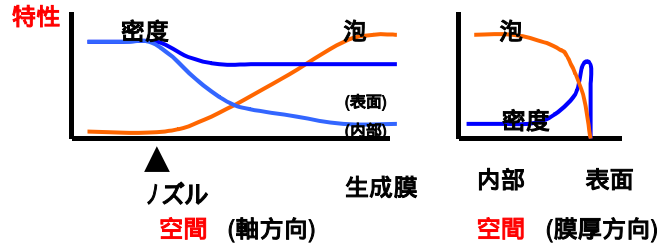
(b) 空間・時間特性の分析: 空間および時間的な特徴・特異性を見出す。(第 9 回 2.3)

例: 「樹脂シートの発泡倍率の増大」の問題

問題状況

<略>

理想解の特性 (定性的な概念図)



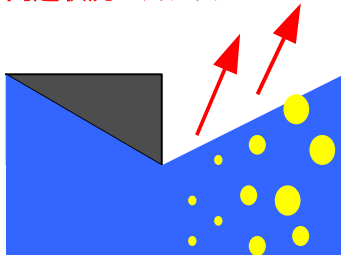
時間 (この場合は軸方向空間特性と定性的に同じ)

(c) Particles 法: まず理想解をイメージして, 望ましい行動と性質を考える。

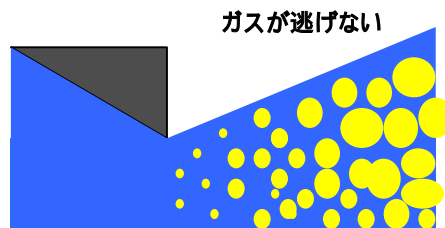
(c1) 現状のスケッチと「理想」をイメージしたスケッチを描く。
違いのある部分に x 印を書き, 魔法の Particles だと考える。

例: 「樹脂シートの発泡倍率の増大」の問題 (再掲: 第 9 回 § 4)

問題状況のスケッチ:

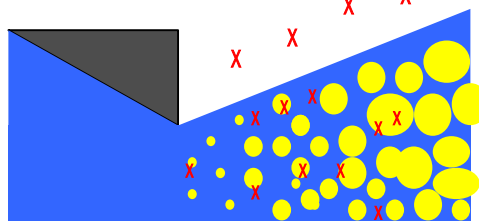


理想解のスケッチ:



泡が多く, 大きくなる

Particlesの適用:

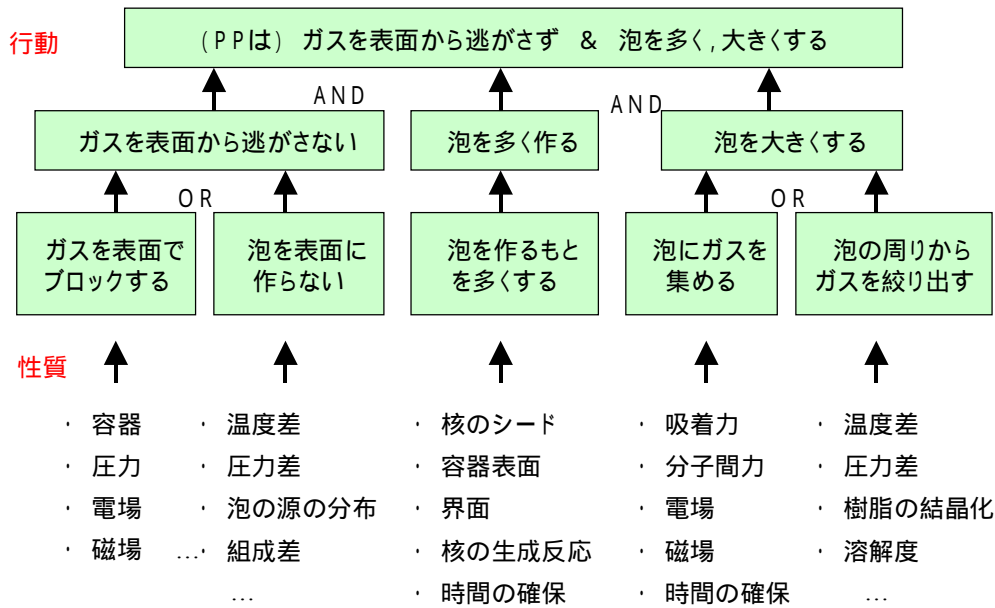


XX "Particles":
任意の性質を持ち,
任意の行動ができる
魔法の物質/場

樹脂の内部, 泡の中と周辺
および樹脂の外部に Particlesを配置した

(c2) 行動-性質ダイアグラムを構築する:
Particles に託したい「行動」を考え, 詳細化する。
Particle の望ましい「性質」の候補を列挙する。

例： 「樹脂シートの発泡倍率の増大」の問題（再掲：第9回 §4.3）



6. USITの 解決策生成段階

(本講義 第11回 §4, 第12回 §2 - 4)

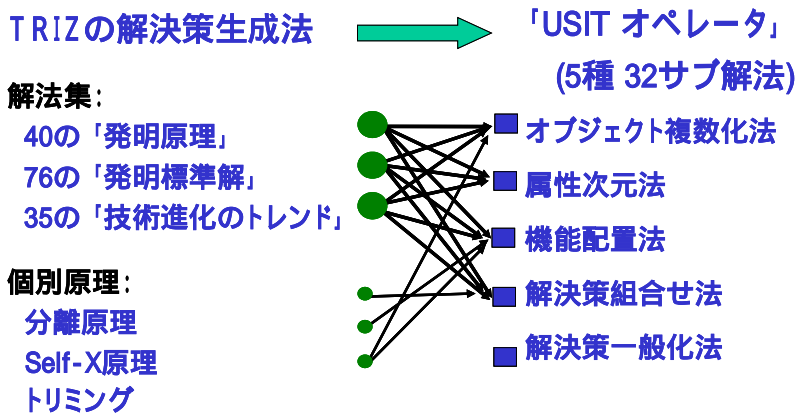
- (a) オブジェクト複数化法: オブジェクトを「複数化」する。
- (b) 属性次元法: オブジェクトの属性の「次元」を変える。
- (c) 機能配置法: 機能をオブジェクト間で再配置する。
- (d) 解決策組合せ法: 複数の解決策をさまざまな面から組み合わせる。
- (e) 解決策一般化法: 解決策の用語・概念を一般化し、体系化する。

TRIZの解決策生成技法のすべてを再整理して、USITの5解法に統合し直した。

USITの解決策生成法の体系と使い方

TRIZのすべての解法をばらして、再編成した

中川徹・古謝秀明・三原祐治 (ETRIA 2002)



USIT 解決策生成法 一覧表

1) オブジェクト複数化法

- 消去する
- 多数 (2, 3, ..., 個) に
- 分割 ($1/2, 1/3, \dots 1/n$ ずつ)
- 複数をまとめて一つに
- 新規導入/変容
- 環境から導入
- 固体から, 粉体, 液体, 気体 へ

2) 属性次元法

- 有害属性を使わない
- 有用な属性を使う
- 有用を強調, 有害を抑制
- 空間属性を導入,
属性(値)を空間変化
- 時間属性を導入,
属性(値)を時間変化
- 相を変える, 内部構造を変える
- マイクロレベルの属性
- システム全体の性質・機能

3) 機能配置法

- 機能を別オブジェクトに
- 複合機能を分割, 分担
- 二つの機能を統合
- 新機能を導入
- 機能を空間的变化, 移動/振動
- 機能を時間的に変化
- 検出・測定 of 機能
- 適応・調整・制御 of 機能
- 別の物理原理で

4) 解決策組み合わせ法

- 機能的に 組み合わせる
- 空間的に
- 時間的に
- 構造的に
- 原理レベルで
- スーパーシステムに移行

5) 解決策一般化法

- 用語の一般化と具体化
- 解決策の階層的な体系

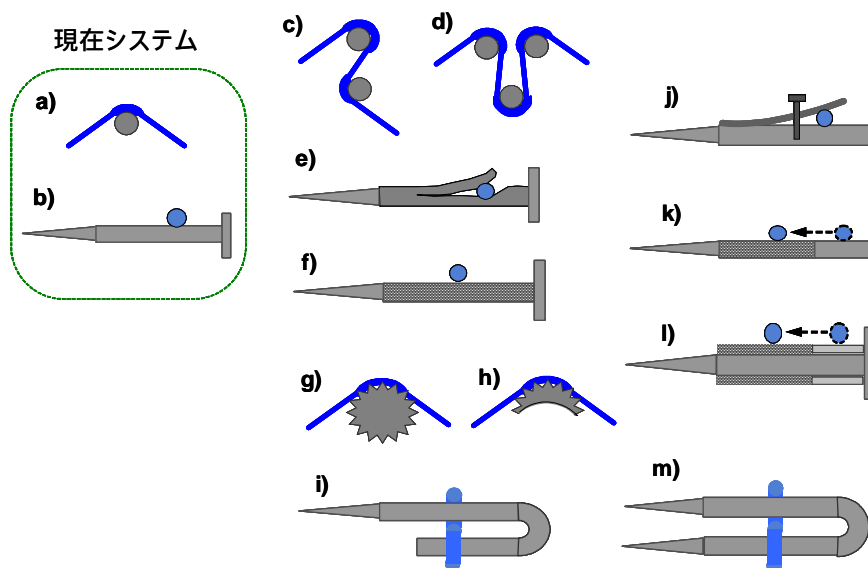
USITの解決策生成法の一覧表 (再掲: 第12回 §4)

USITの解決策生成法はTRIZの解法の全体を受け継ぎ, 簡潔で強力な体系。
これらの解決策生成法を「オペレータ」として順次作用させる。

例: 「額縁掛けの問題」: さまざまなオペレータを適用した例 (再掲: 第12回 §4)

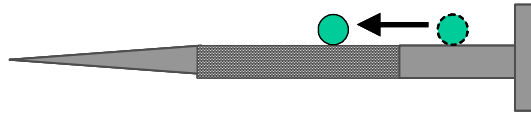
USITの解決策生成オペレータを作用させた例 (部分)

「額縁掛けの問題」で、「釘」にオブジェクト複数化法と属性次元法を作用させた。



(USIT法では) 一つの解決策を複数の方法で同様に生成できる。

例: 「額縁掛けの問題」での「Sickafusの釘」 (再掲: 第12回 §3.1)

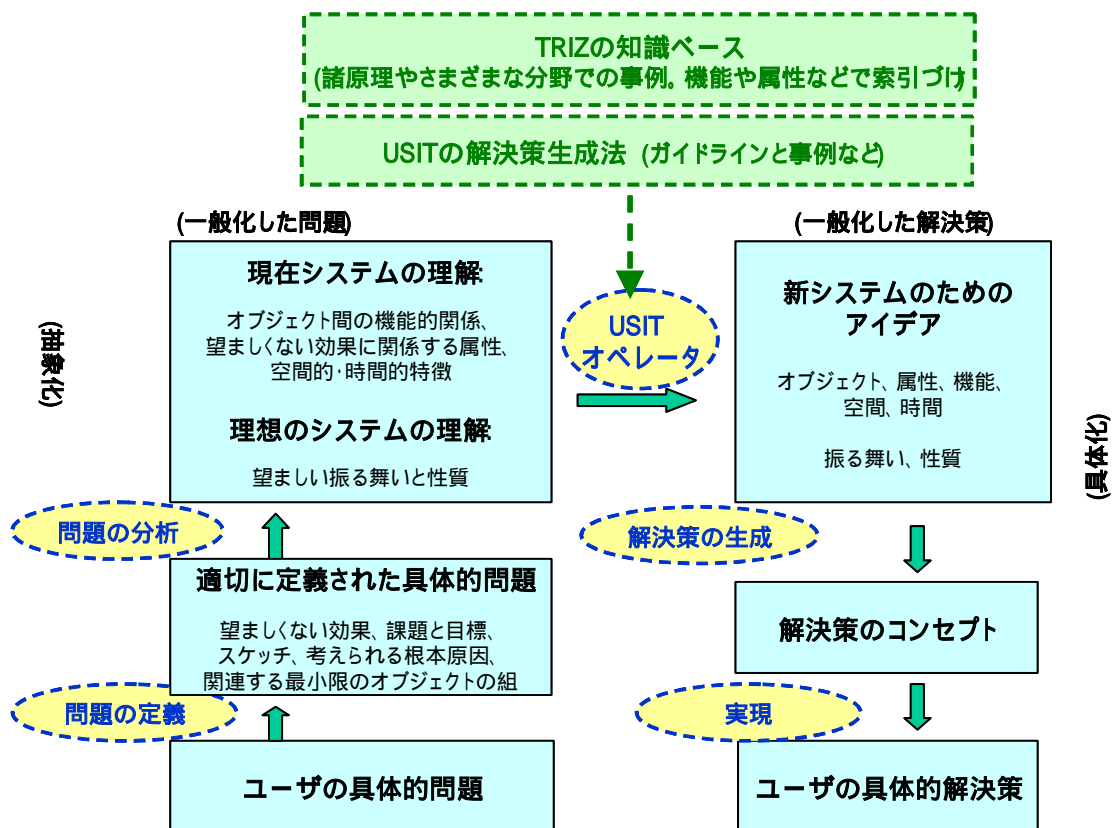


- (a) 「釘」オブジェクトを半分ずつにして、性質を変えて統合。
- (b) 釘の表面の「滑らかさ」の属性の値を、釘の部分によって変えた。
- (c) 釘の「調節」機能と「保持」機能を分離し、釘の部分毎に担当させた。
- (d) 釘を滑らかにして調節しやすくする解決策と、釘の表面を粗くして、傾きにくくさせる解決策とを、釘の部分分割して組み合わせた。(空間による組合せ)
- (e) 上記(d)の二つの解決策を、時間的に分離して組み合わせた。(時間による組合せ)

USITの「解決策一般化法」は、解決策を体系化し網羅するのに有効である。

7. USITの問題解決の6箱モデル

USITの問題解決のプロセスを、「データフロー図」で詳しく表すと下記のようなものである。



8. USIT法の企業への導入法

USIT法は企業における「技術革新」のための非常に有効な方法に育ってきた。

- ・ USIT開発者の Sickafus による Ford 社での実績 (1995年~2000年)

5年間で 同社の技術者の希望者 1000 名に 3日間トレーニングセミナーを実施。
多数のプロジェクトでの実地適用事例を持つ。適用分野は、技術開発全般。

- ・ 日本では、中川が 1999 年より継続して導入・普及させている。

中川による U S I T 2 日間トレーニングセミナーは下記の要領で行う。

==> 中川 徹:「USIT 2日間実践トレーニングセミナーのやり方 (やさしいTRIZ
の普及のために)」、『TRIZ ホームページ』参照、(2005 年 7月)

この特徴は以下の点である。

- ・ 一つの企業内の技術者 10~25 名に対する 2 日間のトレーニング。
- ・ 第 1 日午前が全体的な講義。以後 1 日半は実地問題でのグループ演習。
- ・ 参加者が担当している実地問題 3 件を持ち込んで研修する。関係者と非関係者。
- ・ 演習は U S I T のプロセスに従い、6 セッションで行う。
- ・ 各セッションは、講師説明、グループ別の演習、全体での発表と指導よりなる。
- ・ 各人は、一つの問題をグループで解決すると同時に、全問題の解決過程も学ぶ。

USIT トレーニングセミナー (2日間)

2006. 3

		9:00	(L4)問題分析(Particles法)
			問題分析2 (Ex 3) グループ演習
		11:15	(D3) 発表・討論
		12:00	(L5) 解決策生成
			昼食
		13:00	解決策生成1 (Ex 4) グループ演習
			(D4) 発表・討論
		14:30	解決策生成2 (Ex 5) グループ演習
			(D5) 発表・討論
		16:15	解決策生成3 (Ex 6) グループ演習
			(D6) 発表・討論
		17:30	(L7) 企業への導入法
		18:00	(D7) 総合討論
10:00	(L0) 導入		
	(L1) TRIZ/USIT の概要		
12:30	昼食		
13:30	問題の概要説明		
14:15	(L2) 問題定義		
	問題定義 (Ex 1) グループ演習		
	(D1) 発表・討論		
16:30			
16:45	(L3) 問題分析(閉世界法)		
	問題分析1 (Ex 2) グループ演習		
	(D2) 発表・討論		
19:00			

注： 全く同じ形式・内容で、公募制、複数企業からの技術者達で行うことあり。
この場合には、守秘義務誓約を行う。セミナー成果は問題提案者に帰属させる。

U S I T の評価は以下のようである。(日本における評価)

- ・ U S I T は T R I Z よりもはるかに学びやすい。
- ・ U S I T はグループの共同作業に適している。
- ・ U S I T は実地の技術問題に適用できる。
- ・ U S I T はコンセプトレベルの解決策を得るのにスムーズで強力である。

USITはTRIZから発展した一つの問題解決法（プロセス）であり、
TRIZのエッセンスを適切に継承・発展させている。
USITは（新しい世代の）TRIZの一部であると位置づけられる。

「レポート」提出について

テーマは本講義（または情報学部分野）に関連したテーマを自分で選択。
「感想文」でない正式「レポート」（研究・調査・学習の報告書）
注意：他者（インターネット資料など）の文をそのまま／抜き書きして
自分が書いたかのように記述しているものは「盗作」であり、不合格とする。
ワープロ打ち、A4サイズ、40字×40行程度、5頁以上15頁以内（超過は認める）

締切： 1月28日(金) 18:00
提出先： OGU-Caddie 「期末レポート」
MS Word