

高等学校「情報」科目の教科書案

—情報の視点からの「問題の把握」とは—

神沼靖子

前橋工科大学情報工学科

概要： 従来、「問題解決の過程」として、定式化、設計、実現、評価があげられてきた。しかし、ここで言う「問題解決とは何か」は、簡単に説明できる概念ではない。そこで、いわゆる問題解決を、情報システム開発における問題と対応づけて、「問題解決の過程」を議論する。情報システムの開発では、システムの「目標」を明確にして、それを達成する手段を考えるために、目標が達成されていない現行システムの状況を表現することから始める。これは「問題を把握する」ための行為である。したがってまず、「問題を把握する」という行為を分析し、その上で問題解決のプロセスを説明する。

The concept of 'grasp the meaning' in information processing of high school programs

Yasuko Kaminuma

Maebashi Institute of Technology

Abstract : It has been proposed that the process of problem-solving are modeling, designing, implementation and evaluation. However, it is not easy to express a problem situation with this process. In the information systems design, it is started from express a current systems in order to achieve the object. Therefore, I will compare the process of problem-solving and the design method of information systems development. This paper presents a concept of 'grasp the meaning' and the process of problem-solving.

1. はじめに

「問題解決」という言葉はよく使われているが、その概念を理解することは簡単ではない。それは、「問題解決とは何か」が、簡単に説明できる概念ではないからである。ここでは、「情報システムの開

発過程」に対応づけて、「問題解決の過程」を表現する方法を考える。

情報システムの開発では、システムの「目標」を明確にし、それを達成する手段を考えるために、目標が達成されていない現行のシステム状況を表現することから始める。はじめに現行状況を調査・分析して問題点や改善事項を抽出し、それらの解決方法を考えて目標とするシステムの仕様を設計し、これに基づいて実現、評価するという過程を経る。

個々の情報システムの開発目的は人間活動を支援するために遭遇する諸問題を解決することであるから、この過程はさまざまな問題の解決に適用できる。つまり、問題解決の出発点は「問題の把握」であると考えることができる。本稿では、「問題を把握する」という行為を分析することによって、この問題解決の意味を理解することを試みる。その上で、問題解決の方法を展開する。

2. 問題解決の過程

問題解決というとき、一般には「XXの解を求めなさい」というように解決目標を明確にしたところから始まっている。つまり、解決すべき目標が予め与えられていて、それを解く手段を考えることが問題解決であると考えられるケースである。たとえば、与えられた目標を達成する手続きを考えだし、それを設計して実現するというスタイルのソフトウェア開発や建築設計における問題解決方法がこれに該当する。

しかし、現実の世の中では、目標が明確になっていないことの方が多い。つまり、「何か問題がある」が、それが見えていないケースである。たとえば、「気分が悪い」というとき、どこがどのように悪いのか、何を治す（解決する）べきなのかが不明である。医者に診断をしてもらうにしても、頭が痛いのか、お腹が痛いのか、お腹のどこがどのように痛いのかなどを表現しなければならない。あるいは、頭が痛いのは、病気ではなくて、難題を抱えているからかもしれない。あるいは、学校に行きたくなくて頭痛という症状に問題がすり替わっているのかもしれない。

そこで、「問題は何か」を分析するために、問題のうちそとからいろいろな視点で観察し、あるいは自らが問題状況の中に入って調査して、「問題の状況」を説明することが重要なのである。つまり、問題解決の第一歩は「問題は何か」を分析し表現することであるといえよう。

情報システムの開発では、現行システム（現在稼働している仕組み）にどのような問題があるか、どのような不便があるのかを調査し分析する。それは、医者や診断行為に該当する。医者が診断結果に基づいて治療方法を判断するように、情報システムの開発者も調査分析結果に基づいて問題をモデル化し、解決方法を明確にしてその設計仕様を作成している。医者が治療計画にしたがって治療を施すように、情報システム開発者も仕様書にしたがってシステムを実現する。システムの目標を達成できたかを評価することは、病気が完治したかを見届けることと同じである。

ここで、ソフトウェア工学や建築学での問題解決スタイルをタイプⅠ、情報システム学や医学の問題解決スタイルをタイプⅡと呼ぶことにしよう。すると、タイプⅠの過程は〔定式化→設計→実現→評価〕とパターン化できる。また、タイプⅡの過程は〔状況把握→状況のモデル化→目標設定→定式化→設計→実現→評価〕と表示することができる。

本稿の立場は、タイプⅡを選択する。その理由は、自ら考える教育を重視し、広い視野で現実問題を見つめ判断する能力を養うことを目指したいからである。

3. 問題の把握

(1) 状況の把握

たとえば、「2の平方根を求める」という問題の理解とは何か。 $\sqrt{2}=1.41421356\dots$ は覚えているが、それが2次方程式 $x^2-2=0$ の解であることの対応づけができない（もちろん、それぞれの計算は理解している）。また、一般の2次方程式 $ax^2+bx+c=0$ を使って2の平方根を求めるプログラムを作成しようとするときに、係数 a 、 b 、 c の値に1、0、-2 を代入することができない（驚くべきことに、これが理工系の大学生に少なからずみられる現象なのである）。これらの状況は、断片的な経験知識を関連づける能力がないことを示唆している。それは、平方根の意味を理解していないというよりも、そこに問題を解くという視点がないことを示している。

情報システム設計者は、システムに関する知識を獲得するために問題の追跡を行っている。このとき、たとえばプロセスを記述したり、図面を描いたりする。これらの記述や図面が発する情報だけでは不十分な場合は、さらにヒアリングなどによって根拠を確認する。あらゆる情報は何らかの理由があって記されているが、必ずしも根拠が明確になっていないからである。

プロセスには人間の行為や周りの状況、あるいはシステムの振る舞いが含まれる。しかし、これだけでは状況を的確に把握することはできない。そこでそれぞれの知識の背景にある文脈を捉える視点も付加する必要がある。この視点の形成が、送り手と受け手の情報共有を可能とするのである。

たとえば、「送り手は自分であり、受け手はコンピュータである」と仮定すると、コンピュータに何らかの処理をさせるために、どのような手続きを与えればよいかを考えて指示を与えることになる。このときプロセスの視点をどこにおくかによって問題の受け取り方が異なる。

問題把握の訓練としては、自らの行為（プロセス）を再認識し、視点を変えてシステムを再構成する方法が容易である。具体的事例として、「買い物をするときに代金をどのように支払い、お釣をどのように受け取るか」という計算支援の問題を視点を変えて考える。あるいは、「通学バスに乗っていたが交通渋滞のために学校に遅刻してしまいそう。どうしたら遅刻しないですむか。」という問題を価値観を変えて考えてみる。またあるいは、「M市のAさんの家にはよく車が飛び込むようで、もう十数回も被害を被っているそうです。」というテレビ報道をもとにそれは何故か、どうしたら車が飛び込まなくなるか、という問題解決のために、どのような情報行動をとり得るかを考えてみる。このような問題は、価値観や世界観の違いによって、さまざまな解決案がもたらされるであろう。その中から、次項で示すように何が現実的に可能で、かつ効果的な案であるかを考えて一つを選択することになる。

一般に問題を理解するためには、問題状況がどうなっているかを把握するための何らかの行為または行動をする。その方法として、①問題に関係する資料を集める（資料調査）、②状況をよく知っている人に聞く（インタビュー）、③関係者が集まって検討する（ヒアリング、会議）、④問題の場所に行つて観察する（データ収集、観察）、⑤問題の環境に入って試してみる（アクションリサーチ、視点を変えて行動）、などの方法が考えられる。

前出の「Aさんの家に車が・・・」という問題を、この①から⑤で調査する方法を考えると、たとえば次のような案が浮かんでくる。

①問題を分析するために、近辺の地形図と事故のあった写真を入手して原因を調べる。②Aさんに状況を説明してもらい、③道路の管理者、家の所有者など関係者の意見を聞く。④事故が発生する道路で観察する。⑤実際に事故が発生する道路を車で走ってみる。以上のように異なる調査方法があるが、実際にこれらの全てを試すわけではない。このうちのどれか一つ選んで調査する人もいるだろうし、複数の方法を使用する人もいるであろう。調査の結果、Aさんの家は交差点の一角にあり、しかもその交差点は変則的で、Aさんの家の位置から道幅が急に狭くなっていることがわかった。つまり、広い道路か

ら見るとAさんの家は道路の真正面に見えるのであった。この先は改善のための設計問題になる。

問題解決にはこのような問題点の改善だけではなく、創造的活動を支援するケースもある。この場合には、アイデア生成のためのブレインストーミングなどが行われている。これも議論によって新たな視点を呼び起こす手段となろう。

(2) 状況のモデル化と目標の設定

状況を把握することと、それを他人に説明することはすこし違う。たとえば、自分の目で見たことを文章で他人に伝えようとする、非常に長い説明文を書いても言い尽くすことはできないであろう。文章だけでなく簡単な略図や記号を使って説明を補うことができれば、状況をより分かりやすく説明することができる。

問題を解決する以前の状況がどうなっているのか、図や説明を使って、抽象化しないありのままの姿（情報システムでいう現行物理モデル）を表現すると分かりやすい。有効な図として、データフローダイアグラム、実体関連図、実体履歴図がある。データフローダイアグラムは、データとプロセスの関係を時間軸に関係なく示す図であり、実体関連図はデータどうしの関連を静的に示すものである。また、実態履歴図とはデータの動的な見方、つまり、データへの介入によってデータが変化する状態を示すものである。現行状況をこの3つの視点で独自に表現することによって、それらは互いに比較することが可能になり矛盾を見つけやすくなる。矛盾について、さらに吟味することで問題を明確に捉えることが容易になる。

状況を表現する文章の書き方についても工夫が必要である。「XXがYYされている」、「XXをYYする」、「XXをYYするために、ZZを行う」というように、主語と述語、あるいは目的との関係を明確に整理することによって、情報処理の手順を引き出すことが可能となる。ここで、XXは処理対象のデータであり、YYとZZはプロセスである。

問題の捉え方や視点の違いについて、前出の交通渋滞の話題を再登場させよう。たとえば、次のような解決案が列挙されたとしよう。

- ・遅刻したら学校を休む（学校に行くことに魅力を感じていない生徒の視点：a）
- ・遅刻するのは交通渋滞が原因だから何も気にすることはない（自分が考える問題ではないという生徒の視点：b）
- ・通学バスのダイヤを変更しよう（生徒が遅刻しなければよいという先生の視点：c）
- ・バスダイヤを変えるか授業開始時間を変えるか考えるべきだ（生徒会の視点：d）
- ・交通渋滞を緩和するために道路幅の拡大や信号機の制御変更が可能か（学校関係者の視点：e）
- ・バスで通学するのを止めよう（生徒の視点：f）
- ・家を早く出ればよい（父兄の視点：g）

これらのばらばらな視点は、問題解決の方向づけとしてグループ化ができる。それらは、個人の行動を変える立場（a, b, f, g）、仕組みの改善を検討するケース（c, d, e）に分けられる。個人の問題とする立場は、この先システムの改善を検討する必要がないのでここでは扱わない。そこで、2番目のケースをさらに分析することにする。どのように改善するかを検討する前に、何故交通渋滞になるかを調査する必要がある。そこで、XX, YY, ZZを使って問題状況を表現する方法を利用する。

このとき、先の①から⑤を活用して問題を調査・分析し解決すべき目標を定めることができる。

4. 問題の解決

(1) 定式化と設計

定式化の方法として文脈を捉えるモデリングがある。たとえば、プロセスを記述する要素（対象、行為、制約、機能など）を記号化してプロセスを構造的に示す。ここで、制約は、問題解決の視点と選んだ根拠に関する情報を示し、機能とは、既存の標準機能やアプリケーションなど選択できる資源を指している。

システム設計では、モデリングにあたってシミュレーションや過去のプロセスの追跡などを行っている。この方法は背後にある視点を理解するのに有効である。また、既存のアプリケーションや過去に作成した標準的な機能を使って、簡単な実験を行うこともある。あるいはまた、可視化によって、情報の収集や編集を分かりやすく表示するのもよい。

問題の解決目標を示すのに図や絵で補足すると効果があることを既に述べたが、これらを生成したり解釈する思考プロセスの背景には、物理的な知識の他、何らかの経験的な知識の断片が利用される。経験が少ない場合には、先行事例の知識や概念を流用することもある。したがって、これらの知識や概念を関連づけるために、問題のある周辺情報を収集し分析することが重要となる。シミュレーションは問題の文脈を理解し、客観的に問題を表現するのに役立つ。

情報システムのように自然科学のルールに依存しないものは、定式化だけで事象を表現することが困難であるが、起こりうるすべての事象を検討することはできない。そこで、無数の事象の中から問題状況に関係するものを選んで複数の仮説を立てている。その中から合理的な手段（関係者の合意など）で一つを選ぶことになる。それは何らかの制約や理由があって選択される訳であるから、何故選ばれたのか、何故選ばれなかったかを明確に記述しておかなければならないし、また実現の可能性も保証できなければならない。ここに経験的な知識が必要であるが、この知識の獲得には実践が重視されている。

たとえば交通渋滞の問題では、何故渋滞するかを知るために、道路の利用状況を一定期間にわたって継続的に調査することが考えられる。調査で得られたデータを統計的な手法で分析して、仮説と比較しながらモデルを形成する。このような作業では、表作成のアプリケーションを使用することも可能である。さて、モデルを参照して、d、e、fの視点のいずれを目標にするのかを設定しなければならないが、実現可能性の条件には技術のほかに費用と期間が大きな制約になるので、ここではさらにバスダイヤの変更（c、dの一部）と信号機の制御（eの一部）の2案に絞ることにしよう。

目標を「ダイヤの変更」とする場合には、時間固定方式、シーズンと連動した時間可変方式、他の交通機関と連動した時間可変方式などの中から一つを選択して設計することになる。目標を「信号機の制御」とする場合には、混雑状態に連動して制御するシステムを設計することになる。

システムの設計では、さらに詳細仕様（対象や行為の要素の記号化、関数による機能の表示、論理式による制約の表示などの論理的な詳細設計）を決定したり、実現仕様（データベースやアプリケーションの導入方法、プログラムの仕様などを決める物理的な設計）を決めたりする。このとき、意志決定や目標設定、視点の形成、プロダクト情報や内部仕様の形成に必要な能力が問われる。これらの知識を獲得するには抽象化された文脈も役に立つ。たとえば設計では、メタファー（直観的に意味をイメージする手段）やアナロジー（構造的・機能的に類推する手段）が積極的に用いられている。

(2) 実現と評価

新しい目標が定まり、その実現モデルが描かれ仕様が確定すると、仕様に基づいて実現化が行われる。その問題の性質によって、既存のアプリケーションソフトを活用して実現できるものもあれば、仕組みそのもののプログラムを開発しなければならないものもある。また、いろいろな手段を組み合わせることもある。どのような資源を使って実現するかは仕様に定められているために、この仕様に忠実で且つ技術を正しく活用してバグのないプロダクトを生成できることが実現で問われる能力となる。

結果の評価は、大きく分けて2つある。一つは実現したシステムが正しく機能するかのテストであり、もう一つはシステムが要求を満足しているかの評価である。機能のテストは実現者と出題者がそれぞれ独自のテストデータを作成して行う。この時あらゆる状況のデータを組み合わせたテストが望ましいが、それは現実問題では不可能である。そこで、経験的な知識を活用して、重要なケースについて重点的なテストを行うことになる。

要求の満足度に関する評価はさらに難しい。既に仕様を確定する段階で関係者の合意を得ているはずであるが、その時点で意味理解のずれがあったにも拘わらず認識されていなかったり、文脈に変化があったりして要求の内容が違ってしまうことがあるからである。また、経験的な知識はシステムを開発している間にも蓄積されているために、要求そのものが少しずつ変化してしまうという問題も発生する。

というわけで、システムが使用されている限り、評価活動は続けられることになる。

5. おわりに

情報システムの開発技法を参照しながら、問題解決とは何か、問題把握とは何かについて述べてきた。我が国の教育では、与えられた目標を如何に効率よく解決するかという視点で、定式化、設計、実現を行う狭義の問題解決が主流である。問題の本質やその原因を探ることが重要であるにも関わらず、時間がかかること、必要な知識の範囲が広いことなどがネックになって、避ける傾向がみられる。

しかし、経験的な知識が少なくても、それなりに活用することによって問題の本質を見つけることは可能であり、これによって「考える」という習慣を身につけることができる。それは、どの教科でも必要なことであるが、身近な問題を扱うことができる「情報」という教科ではこれが特に適している。つまり、無意識に行われている行為や行動を顕在化することによって、問題解決への糸口を見つける訓練をしようということである。このとき、プロセスの背景にある文脈を捉える行動を付加すれば、問題発見の視点はさらに広がり、より確かなものへと近づくことができると考える。

一方、人間は経験的な知識を何時でも思い出せる訳ではないし、知識として知っていてもそれを生かせないこともしばしばある。また情報処理的な問題解決は自然科学のルールだけで満たされる訳ではなく、多義的な解釈が可能である。そこで、経験したこと（経験事例の解を得るまでの思考過程）を知識として蓄積しておき、後の問題解決に役立たせることを考えれば、問題の整理方法やデータベースなどへの蓄積技術も修得させることができる。

人工物を生み出すことは人間が主体であって、人間のみが可能である。実際には、問題把握の能力の育成を正面から議論しなくても（つまりそれを目的としなくても）、教育実践の手段として常に使い続けることにより自ずと身につくものであると考えている。

参考文献

- [1] Checkland, P.: Systems Thinking, Systems Practice, John Wiley & Sons, 1981

- [2] 神沼靖子, 佐藤敬 : アクションリサーチとソフトシステム方法論, 情報処理, Vol.36, No.10, pp.941-946, 1995
- [3] 浦昭二, 神沼靖子, 槻木公一, 松谷泰行共訳 : 情報システムの分析と設計—SSADM とその実践—, 培風館, 1995
- [4] 浦昭二, 細野公男, 神沼靖子, 宮川裕之編著 : 情報システム学へのいざない, 培風館, 1998