

問題はどこにあるの？

【自己紹介】

飯泉 純子

iizumi.j@lcorp.co.jp

株式会社一（いち）

専任コンサルタント

<http://www.lcorp.co.jp>



一（いち）は、IT/ソフトウェアエンジニアリングをコアとするコンサルティングファーム。IT システム関連の調達・開発プロジェクトの第三者見積り評価、診断・改善のコンサルティングをコアビジネスとし、自ら開発はしないところが価値。

IPA/SEC 価値指向マネジメント WG メンバ、電子情報技術産業協会ソフトウェアエンジニアリング技術分科会幹事、アジャイルプロセス協議会副会長、同協議会/見積・契約 WG メンバ

【バックグラウンド】

日立製作所において、ソフトウェア開発支援ツール、要求定義・開発方法論の研究に従事。2001 年より IT システム関連の調達・開発プロジェクトの見積り評価、診断・改善のコンサルティング、ならびにソフトウェア技術者教育支援を行っている。

今回ペアで発表する弊社の大槻とは、風呂敷を広げる係（大槻）と畳む係（飯泉）の関係と称す。日立製作所時代は、ソフトウェアエンジニアリングの興味深い研究については、直属の上司よりも影響を受けた。

今回取り上げる「問題フレーム(Problem Frames)」や、同じく M. Jackson 氏がそれ以前に発表した JSD(Jackson System Development)についても、大槻からの推薦がそれらの方法論に触れるきっかけとなっている。

【宣教する方法論：問題フレーム】

(1) 実世界と問題

問題フレームは、ソフトウェアを使って解く「実世界にある問題」を分析する方法である。問題は、解決策である計算機やソフトウェアの外の世界、つまり、「実世

界＝ソフトウェアとは独立な世界」にある。システム／ソフトウェア開発において、随分前から「いかに (how)」よりも「何を(what)」を先に考える必要がある、「何を」が重要だ、と言われてきている。しかし、「いかに」と「何を」の区別は意外と難しい。問題が「どこに(where)」あるのかを区別することの方が、簡単で便利だ、というのが Jackson 氏の主張である。

(2) 「モデル」の意味

「モデル」という言葉には、いろいろな意味がある。混同されたり曖昧な使い方をされていることが多い。ここでは、ソフトウェア開発で主に使う 2 つのモデルを示しておく。

- (a) 分析モデル：モデルの対象に関する記述。例えば、経済モデル。さまざまな要素間の関係を記述する微分方程式の集合である。ソフトウェアのモデルは、外部からのイベント刺激を内部のソフトウェア状態の変化へ結びつける有限状態機械としての振る舞いの記述である。
- (b) 類推モデル：類似の特質を持った別の実体。例えば、関が原の合戦での両軍の動きを模擬するための地図や駒。類推モデルは実世界と対応させることにより、実世界の状態を把握することができる。

ソフトウェアにおいては、この 2 つのモデルの両方が必要であり、混同することは危険である。図書館の世界の問題をソフトウェアを使って解くとき、「会員と本のモデル化」が意味するのは、「(実世界の) 会員と本について記述する」ことなのか、「会員と本の類推モデルを形成するデータベースの仕様を記述する」ことなのか。この 2 つは全く違った活動である。

問題フレームでは、分析モデルを「記述」と呼び、類推モデルのみを「モデル」と呼ぶ。そして、「モデリング」活動については、一切言及しない方針を採っている。

(3) システムと機械

「システム」も、混同されやすい言葉である。「図書館システム」は、広い意味では、会員、図書館職員、書籍、書棚、提携図書館なども含む。狭い意味では、計算機（ハードウェア）とソフトウェアである。

問題フレームにおいては、「システム」という曖昧な言葉を避けるために、計算機とソフトウェアを示す言葉として「機械(machine)」を用いる。機械は、問題を解決するために、最終的に構築およびインストールしなければならないものである。

(4) ドメイン、機械、インタフェース、要求

解くべき問題の範囲のことを「コンテキスト（文脈）」と呼ぶ。問題フレームのコンテキスト図は、問題がどこにあるか(where)という観点で物理的ドメインを描く。

ドメイン(domain)は、ひとまとまりのものとして考えると都合がよかったり、ある程度世界の他の部分から分離して考えることができるものである。適用領域(application domain)と機械は区別すべきドメインである。

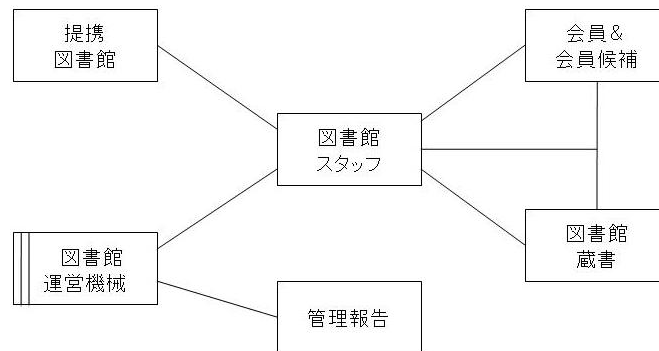


図 1 図書館問題のコンテキスト図

ドメイン間で作用を及ぼすもの、すなわちインタフェースを持つものを線で結んで表す。インタフェースとはドメイン間の共有現象である。適用領域と機械のインタフェースは仕様である。

要求とは、実世界のあるべき現象に関する記述であり、機械が及ぼす効果によって適用領域において達成されるべき条件である。

(5) 問題フレームとは

問題フレームとは、一種のパターンである。いくつかの有効な考え方や、ソフトウェアによって解くという前提での問題領域でのものの見方をまとめたものである。非常に単純で、それゆえに繰り返し現れる、少ない数のパターンに分類することを意図している。基本的な問題フレームとして、以下の 5 つがある。

- (a) 動作制御フレーム(Required Behavior Problem Frame)
- (b) 操作命令フレーム(Commanded Behavior Problem Frame)
- (c) 情報表示フレーム(Information Display Problem Frame)
- (d) 単純編集フレーム(Simple Workpieces Problem Frame)

(e) 変換フレーム(Transformation Problem Frame)

【問題：片側交互通行用の信号機制御】

狭い橋や道路の補修区間等で、両側に信号機をつけて、車輛を一定時間ごとに一方ずつ通す。この両端にある2台の信号機を制御する信号制御機械を構築することが目的である。

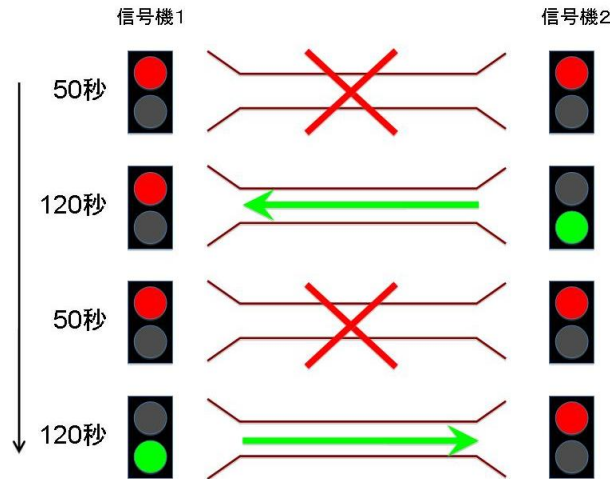


図 2 片側交互通行用の信号機

この問題は動作制御フレームに適合している。プレゼンテーションでは、問題フレームを使った結果、ドメイン、機械、要求がどのようなになるかを示す。

【おわりに】

「問題の解き方よりも、まずは問題そのものを把握することに注力せよ」という考え方には非常に共感している。しかし、実のところ、私自身が問題フレームを完全に理解し、使いこなしているかというと、そうでもない。問題フレームをある企業の技術者教育の題材として教えながら、どのようにして使えばいいかを探っている状態である。ワーキング・グループ参加者の皆さんのご意見も参考にさせていただきたいと考えている。

【参考図書】

Michael Jackson, “Problem Frames: Analyzing and structuring software development problems”, Addison-Wesley, 2001

マイケル・ジャクソン, 「プロブレムフレーム：ソフトウェア開発問題の分析と構造化」, 翔泳社, 2006年5月