

# 問題フレームについて

玉井 哲雄<sup>†</sup>

どのようなソフトウェアもそれが対象とする問題領域があり、開発されたソフトウェアはその問題領域に導入され、その環境で動作することによって初めて意味を持つ。したがってソフトウェアの開発を計画する際には、いきなり解法を検討するのではなく、まず何が問題か、問題を構成する領域は何で、領域間にどのような構造があるか、という考察から始めなければならない。

Michael Jackson によって提唱された「問題フレーム」は、そのような問題の構造と特徴を明らかにするための枠組みである。一般に問題は、複数の「問題領域」、それと相互作用をする「機械」(開発を想定しているソフトウェアの載ったコンピュータ)と「要求」とで構成される。問題フレームはそのような問題を分類し、典型的な形態をパターンとして表現したものである。

この稿では、このような問題フレームを要求定義技術との関連で概説する。

## On Problem Frames

TETSUO TAMAI<sup>†</sup>

### Michael Jackson が産み出した問題フレーム

Michael Jackson が問題フレームという概念を著述によってはっきりと世に問うたのは、1995 年に出版された「ソフトウェア要求と仕様 - 実践, 原理, 偏見の辞典」(以下では「辞典本」と呼ぶ<sup>1)</sup>) によってである。この本はいくつかの項目を選んでそれぞれに数ページのエッセイ風の文章を書くというスタイルで作られていたが、その項目の中に「問題フレーム」「単純制御フレーム」「単純 IS フレーム」「製造品フレーム」「JSP フレーム」「接続フレーム」「多フレーム問題」などがあった。

辞典本の記述は性格上断片的であったが、これを集大成した本格的な著作として「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」(以下では「問題フレーム本」と呼ぶ<sup>2)</sup>) が 2001 年に出された。この書は Jackson らしい綿密周到な作品で、約 400 ページに及ぶ紙幅を用いて十分に書きこまれている。問題フレームを理解するにはこの本を読むのが一番で、一言これを精読せよと言えば、本稿の役目は終わってしまう。それではこの特集号を企画した編集者が洪面を作るだろうから、以降で数ページを費やして、なんとか問題フレーム早わかりの解説となるよう努めよう。

### 世界と機械と要求

Jackson が問題フレームの着想を得たのは、AT&T の Pamela Zave と 10 年にわたって進めてきたソフトウェアの要求や仕様をテーマとする共同研究の中であつたようである。辞典本以前に Zave と共著で出した一連の論文(たとえば 4)の中で、問題領域における要求とシステムに対する仕様との関係が論じられている。問題領域は物理的な実体のある世界という意味で、しばしば実世界あるいは単に「世界」と呼ばれる。また、システムという語は多義的に使われあいまいなので、Jackson は好んでこれを「機械」と呼ぶ。実際、問題フレーム本では一貫して「機械」で通している。

図 1 が世界と機械と要求の関係を表す図である。ここでは読者に問題フレームに慣れてもらうため、記法も問題フレームの表現で使われるものを用いている。Zave との論文では別の描き方をしていた。図で世界と機械は長方形で表わされているが、一般に領域は長方形で示すことになっている。機械の方の左側に縦の 2 本線があるのは、この領域が設計すべき対象であることを示す。一般にこのような問題の図示では、縦棒 2 本がついた機械の領域はただ一つ現れる約束である。要求は領域ではなく記述であり、点線の楕円で表わす。要求から点線矢印が世界に向かって引かれているのは、要求が世界を制約することを意味する。世界と機械との間にも当然のことながら関係があり、両者の間の実

<sup>†</sup> 東京大学

The University of Tokyo

線はそれを示している。

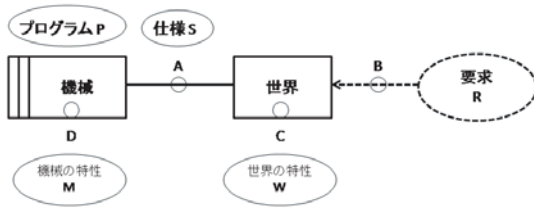


図 1 世界と機械と要求

ここで重要なのは機械と世界の線上にある A と、世界と要求の線上にある B が表すものである。A は機械と世界の間で共有される事象の集合であり、B は世界と要求の間で共有される事象の集合である。2 つの領域やプロセスの相互作用は、事象を共有することによって実現されるという考え方は、C. A. R. Hoare の CSP (Communicating Sequential Processes) に感化された Jackson が JSD (Jackson System Development) で展開した基本概念を継承するものである。

要求の記述は世界と共有する事象 B を用いて表現される。要求は世界に所属する利害関係者が出すものだからである。一方、機械に課される仕様 (S) の記述は、機械と世界とが共有する事象 A を用いて表現される。仕様は機械が満足すべき制約条件を表すものだからである。

話を具体的にするために、缶コーヒーの自動販売機という世界を考えてみよう。ここでは硬貨の投入口、押しボタン、表示ランプ、排出口などを備えた販売機のハードウェアを世界とし、それを制御するコンピュータを機械と考える。B に入る事象は、「硬貨が投入される」「表示ランプがつく」「ボタンが押される」「缶コーヒーが排出される」などである。一方、A に入る事象は、「硬貨が投入されたことを感知しその種類を判定する」「ランプ表示の信号を送る」「ボタンが押されたことを感知する」「缶コーヒー排出命令を出す」などであろう。

世界の内部で起こる事象で要求にも仕様にも使われないものがある。図ではそれを C としている。たとえば「缶コーヒーを収めたラックから 1 本の缶コーヒーを外に送る」などの事象がそれに当たる。また、機械の内部で起こる事象で世界とは共有されないものを D としている。たとえばメソッドの呼び出しや変数への値の代入などがその例である。

世界が関与する事象は、 $A+B+C$  ということになるが、これらの事象が満たすべき性質が世界の特性を定める。図ではこれを W と表している。機械が関与

する事象は  $A+D$  であり、これらの事象が満たすべき機械の特性を、M とする。W と M を厳密に表せば形式的な論理式になる。そこで Jackson は、要求 R に対し次の論理的な関係が満たされるべきであるとする。

$$W, S \vdash R \quad (1)$$

つまり仕様が満足されることと世界に関する性質 W が成り立つことを仮定すると、要求が成立するという意味である。これは逆にみると、与えられた要求に対して、式 1 が成り立つように仕様 S を定める必要があるということである。

問題フレーム本では、問題を扱い解法は扱わないということを強調するので、図 1 に示したプログラム P は出てこない。プログラムは機械の中にあるもので機械に一体化されて考えられている。しかし、仕様が出てきたのでそれを満たすプログラムという関係を考えてみると、次のような論理的関係が満たされなければならないことになる。

$$M, P \vdash S \quad (2)$$

つまりプログラムの機能と機械の特性を仮定すると、仕様が満たされるという意味である。要求の場合と同じようにこれを逆にみると、与えられた仕様に対して、式 2 が成り立つようにプログラム P を作る必要があるということである。この 2 つの並行する関係は、Zave & Jackson の論文<sup>4)</sup>で導入されている。

### 複数の領域を扱う問題

さて、図 1 が問題の基本的な形だが、実際の問題はより複雑であり、一般に複数の領域を含む。世界を表す領域が 2 つになる場合の基本形は、図 2 と図 3 である。

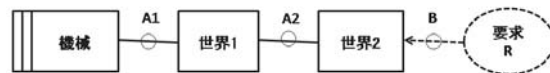


図 2 直列型の問題領域

図 2 の直列型の場合は、要求と機械の間に 2 つの世界が順につながっている。要求は世界 2 に向けられその事象で記述される。世界 1 はより機械に近い世界で、仕様はこの世界 1 の事象で記述される。世界 1 と世界 2 を結合する A2 は、この 2 つの世界が共有する事象の集合からなるインターフェースである。

図 3 の並列型の場合は、機械が 2 つの世界と並列に結合している。典型的な例は世界 2 が発する事象を機

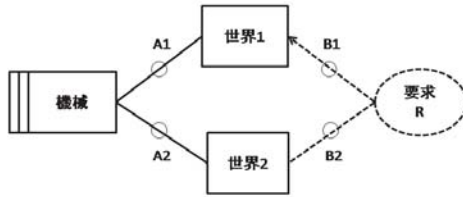


図 3 並列型の問題領域

械が受け、それに基づいて世界 1 を制御する事象を発信するという形態である。この場合は機械から見て、世界 2 は入力、世界 1 は出力という位置づけとなる。要求から世界 1 に出る点線には矢印がつけられてこの世界を制約することを示しているのに対し、世界 2 と結ぶ点線には矢がないのは、要求がこの世界の事象を参照するが制約はしないことを示している。

図 2 と図 3 は問題の形を示しただけの抽象的なものだが、より具体的な問題例を見てみよう。図 4 は問題フレーム本の最初に出てくる集中治療室の医療機器監視システムの例である。

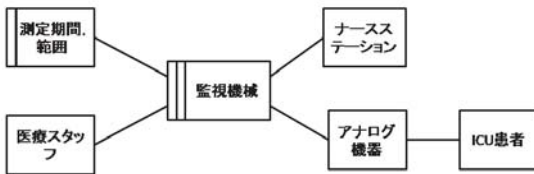


図 4 集中治療室監視システム

集中治療室 (ICU) にいる患者は、種々のアナログ計測器に常時つながれ、データが観測されている。データは随時監視機械に送られる。患者ごとに取るべき計測データの種類と、計測期間、および適正な値の範囲が定められている。その範囲を外れた値が計測されると、ただちに警告がナースステーションに送られる。データの計測期間や範囲は 1 つの領域を作るが、その内容を設定するのは医療スタッフである。

図 4 は、このような複数の領域と機械との関係を描いている。このような問題領域の結合関係だけを表した図は、一般に文脈図と呼ばれる。図を簡素化するため要求や共有事象の表示は省いている。また「計測期間、範囲」の領域の左方に縦の 1 本棒があるのは、この領域が開発者の設計すべき領域であることを示す。開発対象の機械は問題図に 1 つだけ現れるが、そのほかに設計の対象となる領域がこのように登場することができる。

## 問題フレーム

実はここまで図として掲げたのはいずれも「問題」であって、「問題フレーム」ではない。問題フレームとは、よく現れる問題をパターン化したものである。だから図 4 より抽象化されているが、図 1-3 のように形式だけ示したのではなく、問題の種類に応じた固有の意味が与えられている。

問題フレーム本では、5 つの問題フレームが挙げられ、それぞれ例題を使って詳しく説明されている。辞典本でも 5 つの問題フレームが紹介されていた。それらはほぼ引き継がれているが、一部は名称が変わり意味も若干変化している。問題フレーム本の 5 つとは、以下である。

- 動作制御フレーム：物理的実体のある世界の中に、その動作をある条件を満たすように制御すべきものがある場合に、その制御を実現する機械を構築するという問題のフレームである。
- 操作命令フレーム：物理的実体のある世界の中に、その動作を操作者が与える命令に従って制御すべきものがある場合に、操作者の命令を受け対象を制御する機械を構築するという問題のフレームである。
- 情報表示フレーム：物理的実体のある世界の中に、その状態や振舞いについての情報を継続的に知る必要のあるものがある場合に、世界からその情報を収集し、それを定められた場所に定められた形式で表示する機械を構築するという問題のフレームである。
- 単純編集フレーム：コンピュータで処理可能な文書、図画像、その他の構造を生成、編集し、生成された対象を複写、印刷、分析などの用に供するためのツールが必要な場合に、そのツールとして使える機械を構築するという問題のフレームである。
- 変換フレーム：コンピュータが読み取り可能な入力ファイルに対して、そのデータを変換し要求されるような出力ファイルを作る必要がある場合に、形式の定められた入力から要求される形式の出力を生成する機械を構築するという問題のフレームである。

この小解説ではこれら 5 つの問題フレームをすべて紹介するだけのスペースがないので、代表例として動作制御フレームと操作命令フレームを取り上げて説明することにする。

動作制御フレーム

図5は、動作制御フレームの一般形を示すものである。物理的な被制御領域があり、そこで起こる現象を与えられた要求に従うように制御するのが制御機械の役目である。

ここで新たな記法として CM!C1 や CD!C2 が登場している。C1 と C2 は、これまでと同様、2つの領域で共有される事象の集合を表す。この場合は、C1 と C2 のいずれも制御機械と被制御領域の間の共有事象である。両者の違いは事象を発生させる主体で、C1 は制御機械 (CM) が起こす事象という意味で CM! という修飾が頭につき、C2 は被制御領域 (CD) が起こす事象という意味で CD! という修飾が頭についている。

被制御領域の右下に C とあるのは、この領域が因果的 (causal) 領域であることを表す。Jackson は領域に3つの種類があるとする。その一つが因果的領域で、そこで起こる事象に予測可能な因果関係があるものである。たとえば電気機械的な現象が起こるエレベータ設備は、因果的な領域となる。他の二つは、指示可能 (biddable) 領域と、文法的 (lexical) 領域である。指示可能領域とは主として人により構成される領域で、そこに指示を出すことはできるがそれが必ず実行されるとは限らず、その意味で必ずしも予測可能ではない領域である。この領域は右下に B と示すことで表わされる。文法的領域はデータ表現で作られる領域で、そこで起こる現象は記号的なものである。この領域は右下の L をもって表す。

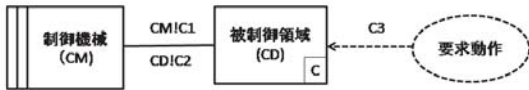
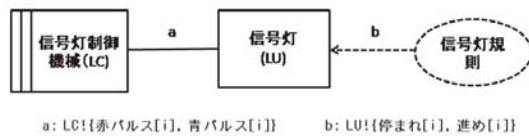


図5 動作制御フレーム

この問題フレームの具体的な事例として、信号機制御問題をとり上げる。



a: LC!{赤パルス[i], 青パルス[i]}      b: LU!{停まれ[i], 進め[i]}

図6 信号機制御問題

工事中で1車線の規制をしている区間の両端にそれぞれ信号灯があるでしょう。信号灯には赤か青の光が点く。制御機械は赤を指令するパルスが青を指令する

パルスを信号灯に送る。それを問題として図示したのが、図6である。その基本的な構造は、図5を引き継いでいる。図で a と表されている事象集合は、信号灯制御機械から信号灯に向けて発せられるもので、赤パルス [i] と青パルス [i] からなる。信号灯は区間の両端にあるので、添字 i によってどちらの信号灯に向けられるのかを区別している。同様に要求としての信号灯規則が被制御領域の信号灯を制約するときに用いられる現象、停まれ [i] と進め [i]、も2つの信号灯を区別するために添字付きとなっている。

操作命令フレーム

図7は、操作命令フレームの一般形を示すものである。この問題フレームは、制御機械と被制御領域が存在するという意味では制御動作フレームと共通点があるが、さらに操作者という領域が加わっているところに特徴がある。この操作者領域は典型的な指示可能領域である。被制御領域の制御は、操作者が行う操作に従って行われる。制御機械は与えられた操作命令に応じて被制御領域を制御する。しかし、操作者は柔軟だがときに誤った操作をすることもある。したがって制御機械は誤った操作に対して安全に機能するように設計されなければならない。

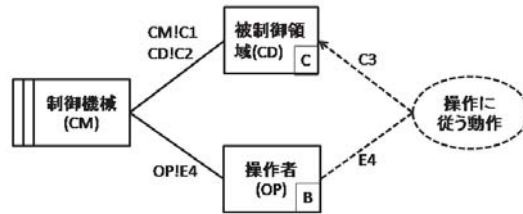
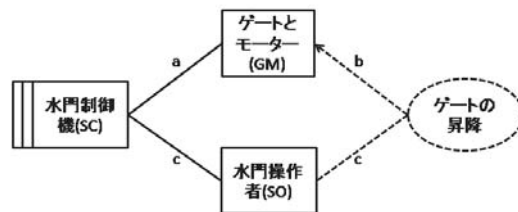


図7 操作命令フレーム

この問題フレームの例として、水門制御問題をとり上げる。



a: SC!{時計回り, 反時計回り, オン, オフ}      b: GM!{開, 閉, 上昇中, 下降中}  
GM!{最上位, 最下位}      c: SO!{上げる, 下げる, 止める}

図8 水門制御問題



水門は上下に昇降するゲートを持ち、水量に応じてゲートの高さが調節されるが、その操作を行う水門操作者がある。操作者が発する命令はゲートを上げる、下げる、止めるのいずれかである。ここで水門操作者が水門制御機に出す命令の現象  $c$  は、要求としての「ゲートの昇降」の記述でもそのまま使われるので、操作者と要求を結ぶ線の上に同じ  $c$  が置かれている。ただし、この  $c$  の内容は操作者の領域を制約するものではないので、間を結ぶ点線には矢印が付けられていない。

水門制御機がゲートとモーター領域に出す信号は、時計回り、反時計回り、オン、オフの 4 種類である。一方、要求の記述で使われるゲートとモーター領域の現象は、開、閉、上昇中、下降中の 4 種類である。この記述はゲートとモーター領域を制約するので、矢印付きの点線が使われている。

### 問題フレームの関心事

典型的な問題フレームを類別しその基本的な構造を示しただけでは、問題フレームのありがたみは大したものではない、といえるかもしれない。Jackson は問題フレーム本で、各問題フレームが持つ固有の関心事 (concern) を明示し、実際にフレームを用いてシステムを開発する際に、どのような点に注意を配って問題を具体化し、解法に向けて展開していけばよいかについて、かなりのページ数を割いて説明している。

ここでいう関心事とは、問題構造の領域や要求の間のインターフェースとして現われる共有事象による記述が持つべき性質と、それらの関係である。その記述は機械に対しては仕様として、また他の領域に対してはそれらに仮定される性質ないし制約として、意味づけられる。

という説明では抽象的で何のことかさっぱり分らないという声が聞こえてきそうなので、すでに出した動作制御フレームの具体的な問題である信号機制御問題と、操作命令フレームの具体的な問題である水門制御問題について、その関心事がどんなものか図解してみよう。

図 9 を見ると、これが図 1 の領域間の共有現象 A や B を具体化して描いたものであることに気がつくだろう。ここでは 1, 2, 3 と番号が付けられた 3 種類の記述がある。それらは順に、仕様、領域の記述、要求である。これらがこの問題の関心事であり、また図で 1, 2, 3 の流れを矢で示しているように、これらの関心事の関係をこのような流れの中に位置づけて、問題分析をすることが勧められている。

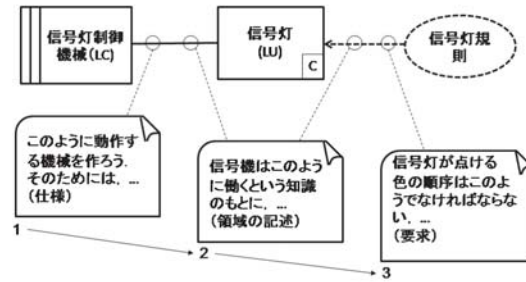


図 9 信号機制御問題の関心事

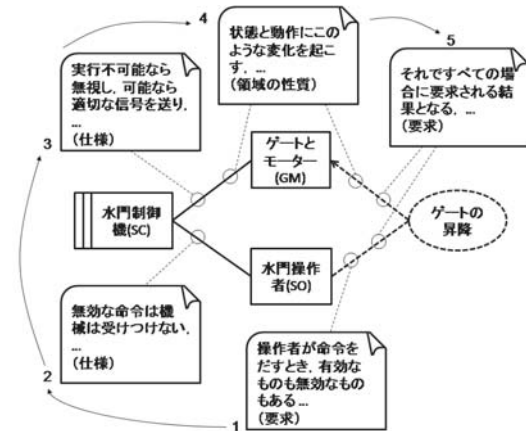


図 10 水門制御問題の関心事

図 10 も基本的に同じように読めるが、ここでは関心事が 5 つの記述に分かれていて、その分複雑である。しかし、図の矢印に沿って 1 から 5 まで見ていけば、この問題で何をとくに考えなければならないかが分かるはずである。

図 9 や図 10 の関心事の書き方は、一部を例示したものにしてもあいまいな感がある。問題フレーム本ではこれを形式化して厳密に記述する方法も書かれている。とくに信号灯やゲートとモーターのような問題領域の性質を記述するには、状態遷移モデルが使われている。実際、事象の集合を語彙として領域の状態や振舞いを記述するには、状態遷移モデルは好適であろう。しかし、関心事の記述方法は状態遷移に限定されるものではない。Jackson も要求や仕様の記述については述語論理表現を併用しながらも、自然言語による表現を中心においている。また文法的領域に対しては、Jackson 得意の Jackson 構造図も使われる。

なお、Jackson は以上のようなフレームに直接関わる関心事のほか、個々の状況に応じて生じる固有の関心事についても触れている。その例には、時間的な

不整合にかかわる関心事（領域のインターフェースで速度の不整合が起こる場合）、初期化に関する関心事、信頼性に関する関心事、完全性に関わる関心事などである。これらは要求工学の分野でひとくくりに「非機能的な要求」とまとめて議論される要求項目と重なる部分が多いが、Jackson 流の問題分析に沿って独自の議論が展開されているので、問題フレーム本のこの部分をじっくり読むとよいだろう。

### 問題の分解と合成

問題フレームは分かりやすいがそれぞれがあまりに単純で、実用的な問題分析の役に立つのだろうか、という疑問が生じることが想像される。もちろん、実際の問題が上にあげたような簡潔な問題フレームの一つに帰着されて、インターフェースが明快であるような場合は限られるだろう。当然のことながら問題をより小さいものに分解し、分解された小問題か、あるいは小問題をさらに分解した小小問題が問題フレームに対応する、ということになるはずである。では、対象とする問題をどのように分解すればよいだろうか。

Jackson によれば最悪な分解法は下降型 (top-down) の機能分解だという。それを印象づけるために挙げている悪例が面白いので、紹介しておこう。100 万までの素数を表示する、という問題が与えられたとする。それを機能分解の精神に従って、次のように展開したとする。

```
p = 2;
while (p<=1000000) {
  print p;
  p = nextPrime(p);
}
```

もとの問題はより小さな  $\text{nextPrime}(p)$  という、素数を引数として与えられ次の素数を返す関数を作るという問題に分解された。ところが最悪なことに、この  $\text{nextPrime}$  の計算はもとの問題より難しいのである。

すでに述べたように、問題フレームで問題領域間で事象が共有されるという発想は JSD にさかのぼり、さらに Hoare の CSP に行きつく。領域はそれぞれ並行的に振舞い、それらの相互作用は共有事象を介してのみ生じるという考え方である。したがって問題の分解も、小問題が互いに他と重なる部分がないように平面的に分割するというよりは、事象の共有を通して並行的に合成されるような形に分けることになる。これを Jackson は分割でなく射影だと言っている。それを模式的に示したのが図 11 である。

図の左は射影で、A, B, C, D は 4 つの小問題を概

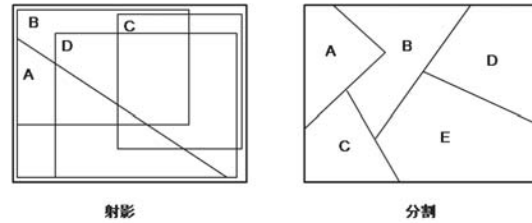


図 11 射影と分割

念的に表している。もとの問題全体がこれで覆われているが、それぞれの間にも重なりがある。一方、分割の場合は小問題間に重なりがない。

複雑な問題をこのような射影的な小問題に分けるのには、ある程度の訓練がいるだろう。そのためにもとよりあえず 5 つある問題フレームに慣れ親しんでおくと、与えられた問題の部分部分に適合したフレームが自ずから見えてくるようになるだろう。

逆に小問題を合成するのは、重ね合わせである。これは本質的に並行性を含んだ概念だから、やはり手強いところがある。しかし、現実の世界で起こる事柄はやはり本質的に並行的である。だからこの方法は現実の問題を扱うのに向いているはずである。

### 問題フレーム研究と実践の最近の動向

最後に、問題フレーム本以降の研究と実践の動向をざっと見てみよう。

問題フレームを対象とした論文はかなりの数が発表されているが、その主な発表の場は要求工学国際会議 (International Requirements Engineering Conference, 通称 RE) である。Jackson は相変わらず孤高なコンサルタントではあるが、5-6 年前から英国の Open University にも籍を置いている。Open University には要求工学の分野で長く活躍している Bashar Nuseibeh のグループがあり、Jackson は Nuseibeh に招かれて彼のグループと共同研究を行ってきている。RE で発表されてきた問題フレーム関連の論文も Nuseibeh 等のものがもっとも多い。その代表的な論文として、問題フレームとアーキテクチャーとの関係を扱ったもの<sup>3)</sup>をあげておこう。

他で目につくのは MIT の Daniel Jackson のグループである。Daniel は Michael の子息で、一階の関係論理言語 Alloy とその検査系の開発などのめざましい研究実績でその名が知られる。その Alloy を駆使して、世界の要求から機械への仕様に変換していく過程、前々節の言葉を使えば関心事の変換を形式的に扱う手法などを提案している。Daniel と Michael 父子は研

究者としても互いに敬意を抱いているようで、それぞれの著書で相手に心のこもった謝意を述べているところもほほえましい。

問題フレームをテーマとするワークショップとしては、「問題フレームの応用と進歩に関する国際ワークショップ (IWAAPF)」が、2004 年以降隔年に開催されている。2008 年 5 月には国際ソフトウェア工学会議 (ICSE) の併設として、第 3 回の IWAAPF がドイツのライプツィヒで開かれる予定である。

産業界における問題フレームを用いた実践例は、まだそれほど多いとは言えない。しかし、過去 2 回のワークショップ IWAAPF では、経験論文がいくつか発表されており、今年のワークショップでもそのような事例報告が多くなることが期待される。

Michael Jackson は昨年 (2007 年) 12 月に名古屋で開かれたアジア太平洋ソフトウェア工学会議 (APSEC'07) に基調講演者として来日し、問題フレームについて基調講演とチュートリアルを行った。いずれの講演の資料も Web 上に掲載されているので、関心のある読者は入手されるとよい。

毎年秋に開催される RE の会議は、今年 (2008 年) は 9 月 8-12 日にスペインのバルセロナで開催される。筆者はそのプログラム委員長を務めているが、基調講演者の一人としてやはり Michael Jackson を招いている。実は Jackson は 1995 年のこの会議でも基調講演を行っているのだが、その時代は RE の開催主体がまだ 2 つの組織に分れていて、年ごとに交互に開くという変則的な形をとっていた。2002 年から両者が完全に一体化して運営されるようになったので、この統合 RE としては初めての基調講演となる。1995 年に Jackson が基調講演をしたときは、ちょうど辞典本が出た年だった。今年はそれから 13 年経っているので、この間の経緯を踏まえた話をしてほしいと頼んでいるところである。

#### 参 考 文 献

- 1) M. Jackson. *Software Requirements & Specifications: a lexicon of practice, principles and prejudice*. Addison-Wesley, 1995. 邦訳 . 玉井哲

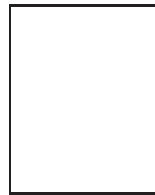
雄, 酒匂寛 訳: ソフトウェア要求と仕様 — 実践, 原理, 偏見の辞典, 新紀元社, 2004 .

- 2) M. Jackson. *Problem Frames: Analyzing and structuring software development problems*. Addison-Wesley, 2001. 邦訳, 榊原彰, 牧野祐子 訳: 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 .
- 3) L. Rapanotti, J. G. Hall, M. Jackson, and B. Nuseibeh. Architecture-driven problem decomposition. In *Proceedings of the International Requirements Engineering Conference (RE'04)*, pages 80– 89, Kyoto, Japan, Sept. 2004. IEEE.
- 4) P. Zave and M. Jackson. Four dark corners of requirements engineering. *ACM Transactions on Software Engineering and Methodology*, 6(1):1–30, 1996.

(平成 年 月 日受付)

(平成 年 月 日採録)

玉井 哲雄 (正会員)



1948 年生 . 1970 年東京大学工学部計数工学科卒業 . 1972 年同大学院工学系研究科計数工学専攻修士課程修了 . 同年 (株) 三菱総合研究所入社 . 1985 年同社人工知能開発室

室長 . 1989 年筑波大学大学院経営システム科学専攻助教授 . 1994 年東京大学教養学部教授, 1996 年東京大学大学院総合文化研究科教授, 2000 年同大学院情報学環教授, 2003 年同大学院総合文化研究科教授, 現在に至る . 工学博士 . ソフトウェア要求技術, 検証技術, モデル化技術, 進化プロセスの分析, 協調計算モデルの開発, 等の研究及びそれらの技術の実践的な問題への適用に従事 . 著書に「ソフトウェア工学の基礎」(岩波書店, 2004, 大川出版賞受賞)、「ソフトウェアのテスト技法」(共立出版, 1988) など, 訳書に「ソフトウェア要求と仕様 – 実践, 原理, 偏見の辞典」(新紀元社, 2004) などがある . 日本ソフトウェア科学会, 情報処理学会, 日本オペレーションズリサーチ学会, 人工知能学会, ACM, IEEE 各会員 .