

組み込み設計を 効率化する UMLの実践



電子ポットの温度制御システム開発を例に

「できる技術者」のクラス抽出は こんなに違う！

山田大介
久我雅人

ここでは、「電子ポットの温度制御」という例題を使って、キャリアの異なる3人の技術者の開発経過を紹介する。オブジェクト指向開発の最初にして最大の難関である、クラス抽出からクラス図の作成にかけての思考過程を探る。また、オブジェクト指向技術を始めたばかりの組み込み技術者が陥りがちなパターンや「定石」とも言うべき方法についても紹介する。

(筆者)

UMLを利用してオブジェクト指向開発を進める際に最初にぶつかる壁は、クラス抽出です(また、これはオブジェクト指向の最大の難関でもある)。特に、オブジェクト指向の経験がある人となない人とは、作成するクラス図がまったく違ってきます。

ここでは、電子ポットの温度制御という例題により、キャリアの異なる3人の技術者がどうクラス抽出を行い、クラス図を作成するかを疑似開発事例として紹介します。

なお、この疑似開発事例は、「組み込みソフトウェア管理者・技術者育成研究会」(SESSAME; Society of Embedded Software Skill Acquisition for Managers and Engineers)の、ワーキング・グループ(WG2)が作成したものです。

●プロローグ：要件定義とユース・ケース

今回設計する電子ポットは、ユーザに以下の機能を提供する家電製品です。

- ポット内の水を沸騰・保温する機能
- ポット内の水をユーザに供する(給湯する)機能
- ユーザが指定した時間が来たら、ブザーを鳴らして知らせるタイマ機能

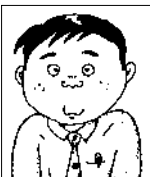
ヒータのON/OFFによって水温を制御します。その制御は、ヒータの操作量(%)に基づいて行われます。操作量を決定する方式としては、PID制御方式、温度制御テーブル方式、目標温度ON/OFF方式の3種類があるものとします。各制御方式の詳細については、今回は重要ではありません。3方式のうちのいずれかが、場面によって選択される、ということだけ知っていれば十分です。

さて、このポットの主となるユース・ケースは何でしょうか？

—— 定石その1：ユーザの心 ——

組み込みシステムでは、適切なアクタを見つけるのが難しく、またユース・ケースの粒度についても、詳細レベルを書いてしまう傾向にあるようです。それを回避する一つの方法として、ユーザの心、すなわち「～したい」

登場人物



●若竹さん

入社1年目。オブジェクト指向技術の習得に日々励んでいる。文科系出身。組み込みシステムの開発は初めて。



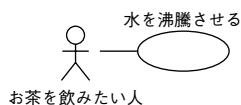
●西林さん

C言語による開発経験10年。組み込みシステム開発も10年。開発チームの中ではリーダ格で、今まで数多くのプロダクトを手がけてきた。ここ数年、実務に追われて、世の中のソフトウェア開発の動向にうとくなくなってしまっている。もちろんオブジェクト指向は初めてで、張りきっている。



●木森さん

オブジェクト指向の開発経験6年。いろいろな失敗プロジェクトを体験し、最近オブジェクト指向を使いこなせるようになったと自覚し始めている。



ユース・ケース名：水を沸騰させる

概要：ポットの中に入っている水を沸騰するまで加熱する

アクタ：お茶を飲みたい人

事前条件：ポットに水が入っていること、ポットのふたが閉まっていること

事後条件：ポットの水が沸騰していること

基本系列：

- 1) アクタは、ポットの沸騰ボタンを押下する。
- 2) システムは、温度制御可能な水位かどうかをチェックし、水位が不十分であればこのユース・ケースを終了する。
- 3) システムは、沸騰ランプを点灯する。
- 4) システムは、ヒータの操作量の算出を行い、ヒータで加熱する。
- 5) システムは、サーミスタをチェックし、現在の温度を表示する。
- 6) システムは、加熱時のエラー検知アルゴリズムにより、エラー検知を行う。
- 7) システムは、現在温度が100℃に到達するまで、4)～6)を繰り返す。
- 8) システムは、現在温度が100℃に到達したら、3分間加熱を続ける。
- 9) システムは、3分間加熱を行ったら、沸騰ランプを消してブザーを鳴らす。
- 10) アクタは、ブザーを聞いて、水が沸騰したことを知る。

〔図1〕今回のメイン・ユース・ケース

という思考で考えてみる「マインド・ステート」という方法があります。

さて、ポットの場合は…。ユーザの心は、「お茶を飲みたい」ではないでしょうか。そうすると、メイン・ユース・ケースは図1のように書くことができます。

今回は、このユース・ケースに基づいて、オブジェクト指向開発を進めたいと思います。

●第1幕：クラスの抽出と初版クラス図

それではいいよ、クラスの抽出です。オブジェクト指向の教科書では「ユース・ケースから名詞句を見つけましょう」とあります。それでは、この水を沸騰させるユース・ケースからクラスを見つけて、クラス図を作ってみましょう。

1) 初心者「もの」を抽出

若竹さんは、教科書どおり名詞句を抽出し始めたようですが、すぐに手が止まってしまいました。

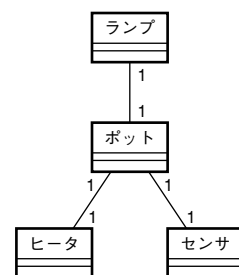
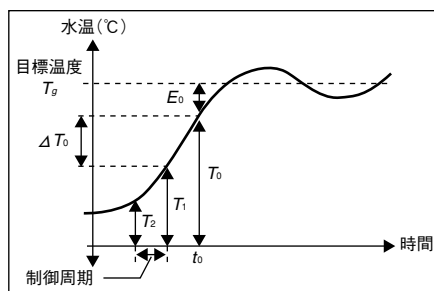
若竹「電子ポットか。ふだん使っているけれど、何をソフトウェアで制御しているのか皆目見当がつかないぞ。クラス候補になりそうなものは、ポットでしょ、あたためヒータでしょ、あとはセンサ、ランプ、ボタン、ブザーがあるみたいだな(図2)。

でも、それらがどのような関係にあるんだろう？ポットがヒータやセンサから構成されているんだよね？でも、どのように動くのか、ぜんぜん想像できないや」

どうやら、若竹さんは組み込みソフトウェアの開発経験

備考：操作量の算出方法の一つであるPID制御方式
この方式では、水の温度特性によって決定される比例係数 K_p 、微分係数 K_d 、積分係数 K_i を使って、以下の式で時間 t_0 における操作量 $M(\%)$ を計算する。

$$M(E_0) = K_p E_0 + K_i \int E_0 dt + K_d dE_0/dt$$



〔図2〕若竹さんのクラス図は目に見える「もの」ばかり

若竹さんがクラスとして抽出したのは、物理的な実体のあるものばかりである。このままでは、「ポット」がすべてを知っていることになる中央集権的なモデルになってしまいそうである。

がないため、物理的な目に見える実体しかクラス候補として見つからなかったようすです。

このままでは、ポット・クラスがすべてを知っていることになる、中央集権的なモデルになってしまいそうです。

2) ベテランは制御の流れからクラスを抽出

一方、西林さんは、ユース・ケースを読んで、なにやらほほえんでいます。もう、先が見えたのでしょうか？

西林「なんだ、これならたいしたことはしてなさそうだから、すぐ動くぞ。しかし、これだけじゃ作れないな。細かいところが抜けだらけじゃないか。ガツンと一発で作ってみるか」

おやおや、西林さんは、ウォータ・フォール・プロセスの癖が抜けきれていないようです。最初から詳細仕様を考えているのでしょうか？

西林「教科書どおりに、まずはユース・ケース記述から名詞句をピックアップしてみよう。

意外とたくさん出てきたな。まずは名詞のグループを作ってみよう(図3)。いろいろ視点を変えるとモデルが変わるな。どうするといちばん動きがわかるだろう？」

処理の流れを追いかけてようとしているのでしょうか？

西林「まずはトリガとして、沸騰ボタンからのイベントが来るから……(図4)」

西林さんは、やはり処理シーケンスとしての動きを追いかけているようですね。もっと、「そのクラスが何をするのか」という視点(責務^{注1}の視点)で考えなければなりません。

注1：「責務」とは、クラスが持つ役割(クラスが何をするのか)のことであり、「機能」とはまったく別のものである。「機能」はシステム内部の処理であり、「責務」は外部から見たミッション(=役割)であると言える。