

4L-7

CafeOBJ による
プロセス記述ライブラリの作成*本間毅寛^{†,1} 中川中^{†,‡} 谷津弘一^{†,2} 二木厚吉[§][†]情報処理振興事業協会 (IPA) 技術センター[‡](株)SRA ソフトウェア工学研究所[§]北陸先端科学技術大学院大学

1 はじめに

情報処理振興事業協会 (IPA) の実行可能な形式仕様言語の研究開発プロジェクトでは、実用的な形式仕様言語システムの基幹言語として、CafeOBJ[3, 5] を開発中である。CafeOBJ は代表的な代数仕様言語 OBJ[1] に

- 動的なシステムの進化を記述するための書き換え規則
- 属性が変化するオブジェクトを宣言するためのクラス構文

を導入し、動的なシステムの記述を行なえるようにした言語である。

従来の代数仕様言語の普及の障害として、ライブラリの不備と記述例の不足が挙げられている。CafeOBJ の基となっている OBJ のライブラリは基本的なデータ型のみからなり、現実のソフトウェアの仕様記述を効率的に行なうには不十分といえる。CafeOBJ の普及のためには、充実したライブラリと豊富な記述例は必要不可欠である。

ライブラリに関しては、現在までのところ、

*CafeOBJ library for Processes Description: Takehiro Honma, Ataru Nakagawa, Hirokazu Yatsu, Kokichi Futatsugi

[†]Information-Technology Promotion Agency, Japan (IPA) Software Technology Center

[‡]Software Research Associates, Inc. Software Engineering Laboratory

[§]Japan Advanced Institute of Science and Technology, Hokuriku

¹(株)SRA より出向中

²日本ユニシス (株) より出向中

OBJ、Z[7]、Larch[6] といった形式仕様言語や ML 等のプログラミング言語のライブラリを参考にし、CafeOBJ ライブラリを作成してきた。現在、ライブラリには、Z の数学的ツールキット相当のモジュールや、リストや待ち行列などの一般的なデータ構造を表現する基本的なモジュールがそろっている。今後は仕様記述実験を通して、ドメインに特化したライブラリを用意していく予定である。

本稿では、ドメインに特化したライブラリの例として、逐次交信プロセスの記述のためライブラリを紹介する。

2 CafeOBJ における逐次交信プロセス

CSP[2] は、Communicating Sequential Process の略で、互いに通信している複数の逐次交信プロセスのシステムを記述するための理論である。数学的に厳密であること、システムの上位レベルを記述できることなどが、大きな特長となっている。以下に、CSP のプロセスを表すモジュールを示す。

```
module PROCESS is
  sorts Event Process ProcessName Action .
  subsort ProcessName < Process .
  using SET[ Event ]*(sort Set to Alphabet) .
  using LIST[ Event ]*(sort List to EventList) .
  op badinput : -> Event .
  op stop : -> Process .
  op bleep : -> Process .
  op _->_ : Event Process -> Process .
  op _|_ : Process Process -> Process .
  op _[_] : Process Event -> Process .
  op menu(,_) : Alphabet Process -> Alphabet .
  op action(,_) : Alphabet Action -> Action .
```

```

op nilaction : -> Action .
op interact(.,.,.) :
  Alphabet Process EventList -> Action .
op concurrent(.,.,.,.) :
  Process Alphabet Alphabet Process -> Process .
vars E E1 E2 : Event .
var EL : EventList .
vars A A1 B : Alphabet .
vars P Q : Process .
var PN : ProcessName .
eq (E1->P)[E2] = if E1==E2 then P else bleep fi .
eq (E1->(P|Q))[E2] =
  if E1==E2 then P else Q[E2] fi .
eq menu(empty,P) = empty .
eq menu({E}+A,P) =
  if P[E]==bleep then menu(A,P)
  else {E}+menu(A,P) fi .
eq interact(A,P,nil) =
  action(menu(A,P),nilaction) .
eq interact(A,P,(E,EL)) =
  if P[E]==bleep
  then action(menu(A,P),
    action({badinput},interact(A,P,EL)))
  else action(menu(A,P),interact(A,P[E],EL))
  fi .
eq concurrent(P,A,B,Q)[E] =
  if P==bleep or Q==bleep
  then bleep else
  if E in A and E in B
  then concurrent(P[E],A,B,Q[E]) else
  if E in A
  then concurrent(P[E],A,B,Q) else
  if E in B then concurrent(P,A,B,Q[E])
  else bleep fi fi fi .
...
endm

```

menu(A,P)は、プロセスPの最初のイベントとして起きる可能性のあるAのすべてのイベントの集合を表す。interact(A,P,EL)は、プロセスの振る舞いを詳しく調べるために定義されている。発生するイベントの列ELを、順にPに適用し、そのつどPのメニューを返す。concurrent(P,A,B,Q)は、同期をとり、相互作用するプロセス(CSPでは“||”で表される)である。A、Bに共通するイベントは、P、Q同時に参加することによってのみ発生する。Pのアルファベット(すなわちA)に属しているが、Qのアルファベット(すなわちB)には属していないイベントは、Qとは無関係に発生する。同様にQのアルファベットに属しているが、Pのアルファベット(すなわちB)には属していないイベントは、Pとは無関係である。

先の記述を利用して、次のような簡単なベンディングマシンを記述することができる。

```

module VMS is
  using PROCESS .
  op Vms : -> Processname .
  op coin : -> Event .
  op choc : -> Event .
  var E : Event .
  eq Vms[E] = (coin->(choc->Vms))[E] .
endm

```

上のモジュールは、

$$VMS = \text{coin} \rightarrow \text{choc} \rightarrow VMS$$

というプロセスを表す。これは、硬貨を入れるとチョコレートを出す自動販売機の動作の記述である。

CafeOBJでは、項の書き換えが発散してしまうので、プロセスを直接等式で定義することはできない。上のように、任意のイベントが起こった後のプロセスが等しいという、いわば観測同値的にプロセスを定義する。

謝辞

本稿で報告した試みは情報処理振興事業協会技術センターにおける実行可能な形式仕様言語システムの研究開発プロジェクトで行われた。

参考文献

- [1] J.Goguen, T.Winkler, J.Meseguer, K.Futatsugi and J.-P.Jouannaud, Introducing OBJ, Technical Report SRI-CSL-92-03, Computer Science Laboratory, SRI International, 1992. To appear In J.A.Goguen editor, Application of Algebraic Specification using OBJ, Cambridge University Press.
- [2] C.A.R.Hoare, Communicating Sequential Processes, Prentice-Hall, 1985.
- [3] 谷津、本間、中川、二木, 実行可能な形式仕様言語 CafeOBJ の概要, IPA 第12回技術発表会論文集, 1993.
- [4] 本間、中川、澤田、谷津、二木, CafeOBJによるオブジェクト指向システムの仕様記述ライブラリの記述, 第4回情報処理学会全国大会講演論文集, 1994.
- [5] CafeOBJ 言語仕様書, 1993.
- [6] J.J.Horning, J.V.Guttag, Larch: Languages and Tools for Formal Specification, Springer-Verlag, 1993.
- [7] J.M.Spivey, The Z Notation: A Reference Manual 2nd ed., Prentice Hall, 1992.