

4F-5 SKETCHシステムにおける 要求仕様記述形式

田口安男、宮下洋一
(情報処理振興事業協会 技術センター)

1. はじめに

要求分析を円滑に行なうため、要求仕様を含むソフトウェア環境を知識情報ととらえる研究が、いくつか行なわれている(例えば、S.Greenspanら[1]のRMLなど)。現在、著者らが開発している要求分析システムSKETCH[2]もそうとらえており、要求分析工程で現われるいろいろなモデル(現在の業務形態モデル、ソフトウェア実現モデルなど、以下こうしたモデルを**対象モデル**と呼ぶ)の記述が、重要な機能の1つである。

本報告では、著者らの経験[3,4]などをもとに考案した、SKETCH内での知識表現指向の対象モデル記述形式について述べ、対象モデルの記述を支援する要求分析ガイド系の基本的機能についても述べる。

2. 対象モデル記述形式

対象モデルの記述単位をオブジェクトと総称するが、次の3つのカテゴリーに整理される。

- ①entity ...物を表わす。
ex. ファイル、ウィンドウ、テキストなど
- ②activity ...entityに対する行為や仕事を表わす。
ex. エディティング作業、ウィンドウ表示など
- ③assertion ...entityに関する条件や関係を表わす。
ex. カーソルのテキスト内での位置に関する条件、ファイルの初期値に関する条件など

記述形式には、フレーム表現方式を採用し(図1)、スロットとしては、単にオブジェクトをいくつかの部分に分割するばかりでなく、オブジェクト間の相互関係や事前事後条件も表現できるようなものを用意し、要求分析ガイド系が容易にアドバイス・ガイド項目を決定し、記述内容を分析できるようにした。例えば、したため、entityの時間的変遷が扱える。また、対象entityへの基本オペレーション(全てのactivityは最とするentityを操作別にわけたため、activity間の情報の授受を分析できる。

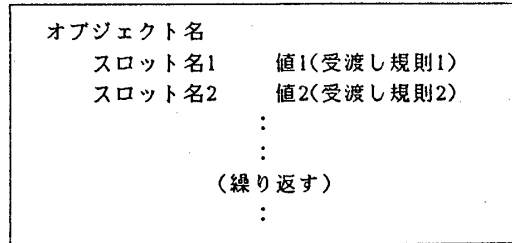


図1: オブジェクトの記述

(説明)

スロット名: 値がオブジェクト内ではたす役割
値: 他のオブジェクト名
受渡し規則: 値に書かれたオブジェクトとの情報(entity)の受渡し方

表1: entityで使えるスロット

スロット名	意味	値域
USR_DEFINE	構成するentity	en
always	常に成立する条件	as
initially	生成時に成立すべき条件	as
finally	消滅時に成立すべき条件	as
producer	生成させる基本オペレーション	ac
consumer	消滅させる基本オペレーション	ac
modifier	修正する基本オペレーション	ac
assertor	entityに関連する条件	as

(注) USR_DEFINEは記述時に定義できる名称を表わす。

表2: activityで使えるスロット

スロット名	意味	値域
refer	参照するentity	en
modify	修正するentity	en
produce	生成するentity	en
consume	消費するentity	en
argu	一時的に使われるentity	en
seq(i)	i番目に行なうactivity	ac
one_of(i)	i番目に選択するactivity	ac
start_cond	activityを開始させる条件	as
stop_cond	activityを終了させる条件	as
initially	開始時に成立すべき条件	as
finally	終了時に成立すべき条件	as

表3: assertion で使えるスロット

スロット名	意味	値域
obj clause	対象とするentity 構成する条件や関係	en as

表1,表2,表3 には、カテゴリごとに使われるスロット名をまとめた。(表中の値域欄では、en=entity, ac=activity, as=assertionを表わす)

3. 対象モデル記述例

2.で述べた記述形式で画面エディタの基本的機能が表現できた。以下はその一部である。(図2,図3,図4)

TEXT	
before.cursor	ASCII_STRING
after.cursor	ASCII_STRING
initially	before.cursor =NULL after.cursor =NULL
modifier	INSERT
modifier	DELETE
modifier	MOVE
assertor	ON_WORD
assertor	BETWEEN_WORD

(説明) TEXTは、アスキー文字列のbefore.cursor,after.cursor からなり、生成時には、どちらも空である。基本オペレーションとしては、INSERT,DELETE,MOVEがある。関連するassertionとしてON_WORD,BETWEEN_WORDがある。

図2: entityの記述例

EDITING	
modify	FILE_SYSTEM
arg1	TEXT
arg2	BUFFER
seq(1)	READ_TEXT(refer.PAR = modify.SELF, produce.PAR = arg1.SELF)
seq(2)	PROCESSING(modify1.PAR = arg1.SELF, modify2.PAR = arg2.SELF)
seq(3)	WRITE_TEXT(modify.PAR = arg1.SELF)

(説明) EDITING は、FILE_SYSTEMを修正する。その時TEXT, BUFFERが使われる。READ_TEXT,PROCESSING,WRITE_TEXT を連続的に行なう。

図3: activityの記述例

ON_WORD	
obj	TEXT
clause1	BETWEEN_WORD(obj.PAR = obj.SELF)
clause2	('sp ≠ head after.cursor of obj.SELF) or ('cr ≠ head after.cursor of obj.SELF)

(説明) ON_WORD は、TEXTに関する条件で、clause1,clause2 からなっている。assertion では、clause2 のように=、論理結合子(and,or,not,...)を使った簡単な記述を許される。

図4: assertion の記述例

4. 要求分析ガイド系

要求分析ガイド系の支援機能は、次の2つに分かれる。(図5)

①基本ガイド機能

- ・利用者が、オブジェクトごとに必要なスロット値を決定することを対話的に支援する。
- ・未定義オブジェクトを検出する(完全性)。スロットを利用しオブジェクトの相互関係が正しいか、事前事後条件が満たされるかなどを検証する(無矛盾性)。

②知的ガイド機能

対象分野に応じた対象モデルの枠組みやよく使われるオブジェクトを知識ベースに蓄え、利用者が関心を持つべき項目の定義を促すなどのガイドする。例えば、エディタの分野ならば、テキストの論理的構造や表示形式などがあげられる。

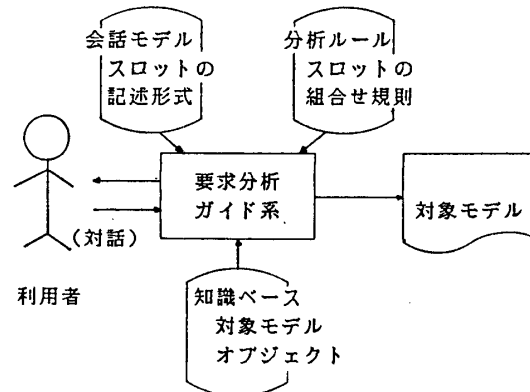


図5: 要求分析ガイド系の概念図

5. おわりに

要求分析における対象モデルの記述が3つのカテゴリにわかれ、必要なスロットが整理できることを画面エディタの記述例から確かめた。現在、KEE^{*)}のフレーム表現を利用して要求分析ガイド系の基本機能を構築中で、知的ガイド機能に関してもガイド方式や検証項目を検討している所である。

[参考文献]

- [1]Sol J. Greenspan, et al: Capturing More World Knowledge in The Requirements Specification, Proc. 8th ICSE(1982), pp.225-234.
- [2] 宮下 他: SKETCHシステムによる要求分析支援、情報処理学会第33回全国大会 4F-4
- [3] 田口、宮下: "Prologによる形式的仕様記述-画面エディタの場合"、情報処理振興事業協会 技術センター 第4回発表回論文集(1985), pp.44-51
- [4] 田口、宮下: 形式的仕様作成における知的対話機能の実験、情報処理学会第32回全国大会 4J-10

*)KEE はIntelliCorp 社の登録商標である。