

4G-3

対話的知識処理システム開発支援の為の Smalltalk 環境

福島 学 小野 晴義 石川 敏 菅原 研次  
千葉工業大学工学部電子工学科

1. はじめに

情報処理システムの機能が高度化するに従ってその操作はますます複雑になってくる。利用者がこのようなシステムを使いこなすためには、ユーザインタフェースの果たす役割が重要になってくる。

本研究は高度ユーザインタフェースを有する対話的知識処理システムのための開発支援環境を実現する事を目的としている。本支援環境は Smalltalk-80 のソフトウェア開発環境に知識ベース機能とユーザインタフェース記述機能等を組み込む事により対話的知識処理システムの構築を支援する。

Smalltalk-80 にはオブジェクト指向プログラミングやユーザインタフェース機構等の特徴を持っているものの従来使われている知識記述言語は備わっていない。またユーザインタフェース機構を備えているがユーザインタフェースを効率的に記述するためのユーザインタフェース記述言語等は備わっていない。そこで、Smalltalk-80 の持つ様々な特徴と知識記述言語とを Smalltalk-80 の環境下で統合することにより、対話的知識処理システムの構築を支援する Smalltalk-80 環境(Smalltalk-80 Environment for Developing Interactive Knowledge system: SEDIK)を実現することにより先述した開発環境を実現する。

2. SEDIK の概要

知識がシステムに記述されているだけでは構築された知識処理システムを効率的に利用する事が難しい。ユーザフレンドリなシステムであれば、システムをより効率的に利用することが可能である。[1] ユーザフレンドリな対話的知識処理システムとするためにはユーザインタフェースが重要となる。しかし、ユーザインタフェースはそれ単体で知識処理システムとなる程複雑なものであるため、KE にとってユーザインタフェースの設計を行うのが知識を記述するのと同等の負担となる。このため、対話的知識処理システムの設計を支援する為には KE を支援する環境として知識記述環境の他にユーザインタフェース記述環境を提供する事が必要となる。(図1)

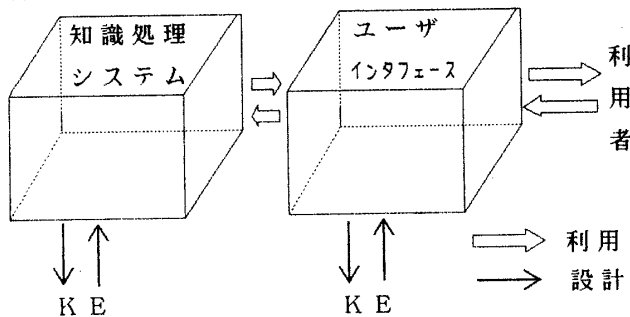


図1 対話型知識処理システム

Smalltalk にはユーザインタフェースに必要なウインドウ、メニュー等を記述するためのメソッドが用意されている。これによりウインドウ等の記述を行う際にそれらのメソッドを用いることにより、記述が容易に行える。また、Smalltalk にはこれらを制御するためのメソッドも用意されているため制御記述も容易に行える。この他、インヘリタンスが行えるために、一部変更が容易に行える。これらの機能により Smalltalk がユーザインタフェース記述に適している。

SEDIK は、従来の知識記述支援ツールの他にユーザインタフェース記述支援ツールを提供する。これにより対話的知識処理システムの構築を支援する。図2に示されるように SEDIK 環境は Smalltalk の環境に含まれており、Smalltalk におけるツールとして利用できる。KE は SEDIK において知識記述を行う際にそれぞれに提供されている KEView を用いることにより記述支援を受けられる。ユーザインタフェース記述においても同様に KEView を用いることにより記述支援を受けられる。ユーザインタフェース記述されたものは利用者用ユーザインタフェースとして生成され利用者の見える型になる。

3. 知識記述環境

知識を記述するためには知識記述言語・知識ベース・推論機構を提供することにより KE を支援することができる。

しかし、構築されたシステムの場合と同様に知識記述時もシステムがユーザフレンドリである方が記述効率が上がる。ユーザフレンドリなシステムであれば、KE は知識の本質にのみ時間をかけることができ、不必要な点で手間を取られないため、効率的に知識記述が行える。

そこで、SEDIK の知識記述環境として、先述した機構の他に KE の知識記述を支援する KEView を提供する。SEDIK ではルール、フレーム言語等が利用でき、各言語に適した KEView を提供する。[2]

この KEView とは知識記述環境における KE という利用者に対するユーザインタフェースである。すなわち、知識記述環境も知識処理システムであり、KE の知識記述

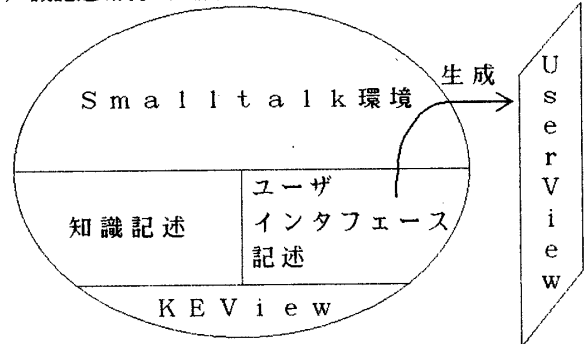


図2 SEDIK の概要

を支援するシステムである。そこで、知識記述環境に於いてもKEから知識獲得する為の知識、獲得支援の為の知識、KE用ユーザモデルが定義され、これを基にユーザインタフェースがKE Viewの種類等を決定していく。(図3) [3] すなわち、KEは知識を記述する際にKE Viewを用いる事により、KEが記述の際にミスを起こしても知識記述環境からの指摘によりミスに気付ける。これにより、KE Viewを用いた方が知識の記述が容易になるばかりでなく、ミスを未然に防ぐことにより知識のデバッグも効率的に行われる。

4. ユーザインタフェース記述環境

ユーザインタフェースとは、利用者とシステムを結び付けるものである。手法としてはマルチウィンドウ、ポップアップメニュー、プルダウンメニュー、アイコンなどが挙げられる。対話的知識処理システムに於いてはユーザインタフェースによって利用者のシステム利用効率を上げることが目的である。このためにユーザインタフェースにおいては先述した手法をいかにうまく利用するか、どのように利用すれば利用者にとって使い易くなるかという事が重要となる。

ユーザインタフェースを記述するためには、以下に挙げる各言語が必要である。

- View記述言語
- 対話制御記述言語
- ユーザモデル記述言語

画面に表示されるウィンドウ等を記述するView記述言語、システムからの発話、利用者からの発話、システム受話等の制御を記述する対話制御記述言語、ユーザがどのようなタイプであるか、対話をどのように進めていくべきか等を記述するユーザモデル記述言語である。これらにより、KEは対話的知識処理システムの為のユーザインタフェースを記述し構築する事ができる。

しかし、先述した知識記述でもそうであったようにこのユーザインタフェースに於いてもそれぞれがユーザフレンドリであるほうが記述効率上がる。そこでSEDIKにおけるユーザインタフェース記述環境として先述した言語、記述されたものを管理保管するベース機構、記述されているものを利用するための機構の他に、知識記述環境と同様にKEの記述を支援するKE Viewを提供する。先述したように記述言語が複数あるので、知識記述環境と同様に、それぞれに適したKE Viewを提供する。KE Viewの仕様については知識記述環境と同様の仕様をとる。すなわち、このユーザインタフェース記述環境に於けるKE Viewもそれ自身が知識処理システムとなっている。

対話制御記述言語においてユーザとシステムとの対話型を図4に示すように定義した。さらに、応答要求型のう

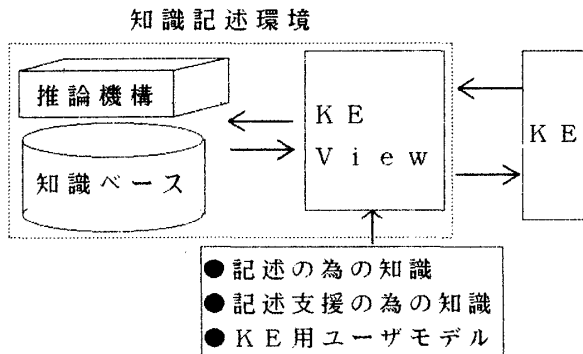


図3 知識記述環境

ちシステムからの発話は、即答要求型とそれ以外のものの2タイプに分けられる。また、宣言型のうちシステムからの発話は、利用者からの発話の対象となるものと、発話の対象にならないものの2タイプに分けられる。

対話制御記述言語において定義された対話型に対応して、View記述言語においても図5に示すようにウィンドウ型を定義した。それぞれのウィンドウには利用可能な対話型と不可能な対話型がある。

これらによりKEはユーザとシステムとの対話のシナリオ、対話に必要なウィンドウ記述を行う事が出来る。

5. おわりに

SEDIKは対話的知識処理システムの構築を支援する為の知識記述環境、およびユーザインタフェース記述環境を提供する。また、それぞれの記述の際に、それ自身が記述を支援する為の知識処理システムとなっている高度な対話的KE Viewを提供する。

本システムはSmalltalk-80のプログラミング環境の中の知識工学ツール(クラス)として実現されている。本システムはSmalltalk-80プログラマにより自由に利用できるが、SEDIKで提供される、それ自身が記述支援の為の知識処理システムとなっている高度な対話的KE Viewを利用することにより、Smalltalk-80を知らないKEでも知識入力や利用が可能である。

これによりSEDIKに於いてそれぞれの記述言語に提供されているKE Viewを用いる事により、知識処理システムの構築、さらにユーザインタフェースを構築することが出来るため、対話的知識処理システムの構築を効率的に支援することが可能になる。

参考文献

- [1] 福島 学等「対話型演習支援システムのためのユーザインタフェース」、情報処理学会第37回全国大会、1988
- [2] 石川 敏等「Smalltalk-80環境における知識ベースシステム」、電子情報通信学会学生会研究発表会、1988
- [3] 福島 学等「C<sup>2</sup>S設計の為の利用者要求獲得システム」、電子情報通信学会学生会研究発表会、1988

タイプ	システム	発話方向	ユーザ
宣言型	資料の表示等	→	表示をみる
	宣言を受ける	←	宣言
応答要求型	質問等	↔	質問に答える
	質問に答える	↔	質問等

図4 対話型

タイプ		宣言型		応答要求型	
		対象	以外	即答	以外
表示	推移変化表示型	○	○		○
	表示型	○	○		○
選択	項目選択型	○	○	○	○
	二項選択型	○		○	○
	単項選択型	○		○	○
入力	複数項一括入力型			○	○
	単項要求入力型			○	○

図5 ウィンドウ型