

コンピュータコミュニケーションシステム設計における 初期要求知識獲得について

木下 哲男 沖電気工業(株)総合システム研究所
菅原 研次 千葉工業大学 電子工学科
白鳥 則郎 東北大学 電気通信研究所

コンピュータコミュニケーションシステム設計の要求仕様定義フェーズにおいて、利用者や設計依頼者から与えられる種々の要求(初期要求)に関する情報(知識)を獲得するための一手法について議論する。本稿で新たに提案する初期要求知識獲得方式は、コンピュータコミュニケーションシステムのための知識型設計方法論の枠組みが提供する知識モデルに基づいて初期要求知識を整理すると共に、要求仕様定義プロセスを知識獲得の手法を流用して支援するものである。本稿では、初期要求獲得のための知識モデル、及び知識モデルに基づいた初期要求知識表現言語を中心として、初期要求知識獲得方式の枠組みを提案する。更に、本方式を実現する初期要求知識獲得支援システムの概要についても述べる。

Original Requirements Knowledge Acquisition for Computer Communication Systems Design

Tetsuo Kinoshita OKI Electric Industry Co., Ltd.
Kenji Sugawara Chiba Institute of Technology
Norio Shiratori Tohoku University

11-22, Shibaura 4-Chome, Minato-ku, Tokyo 108, Japan

It is important to define formal requirements specifications in order to realize many systems including computer communication systems (CCS). This paper proposes a method for acquiring many requirements (Original Requirements Knowledge) given by the end users/clients in the requirements specification phase, upstream design phase of the computer communication systems. Proposed method not only uses a knowledge model defined by the knowledge-based design methodology for CCS, but also uses techniques of the knowledge acquisition of the knowledge engineering. The overview of the knowledge model, knowledge representation language for end users and the original requirements knowledge acquisition method are explained. Further, the basic framework of the support system based on this method is also explained.

1. はじめに

高次の設計フェーズを対象とする知識型設計システムに於ては、システム自身を駆動するための設計知識の獲得という問題に加えて、実際に設計を進めるための要求仕様や設計仕様を初めとする種々の設計情報の収集という問題がある。現状では、設計対象の機能や性能を規定する基本的設計情報である要求に関するこれらの情報を設計依頼者（利用者）から効率的に、また必要且つ十分なレベルで獲得することは、コンピュータコミュニケーションシステムの設計や大規模なソフトウェアシステム設計をはじめとする設計ドメインに於ける現状の大きな問題のひとつとなっている。こうした問題に対して、コンピュータコミュニケーションシステムの知識型方法論とその知識型設計支援システム⁽¹⁾の枠組みでは、こうした要求仕様定義フェーズに対して専用の設計支援エキスパートシステムを割り当て、知識型の問題解決アプローチによって要求仕様を定義することを提案している。

本稿では、知識型設計方法論に準拠して、コンピュータコミュニケーションシステム設計の要求仕様定義フェーズの第一ステージである、初期要求に関する知識の獲得方式とその実現手法について考察する。以降、2章では、コンピュータコミュニケーションシステムに対する知識型設計方法論の基本的枠組みと、要求仕様定義に於ける問題点について述べる。3章では、要求仕様定義フェーズの初期要求の獲得問題に焦点を絞り、知識型設計方法論に準拠した初期要求知識獲得方式の具体的な枠組みを提案する。4章では、提案された方式を実現する支援システムの概要を述べる。

2. 知識型設計方法論に於ける要求仕様定義

2.1 知識型設計方法論のコンセプト

知識型設計方法論は、従来の設計方法論の枠組みに対して知識工学に基づく手法を導入することにより、設計型タスクの高度化を目指したものであり^{(1), (2)}、次の2つのコンセプトに基づいて構成される。

○利用者要求指向アプローチ：利用者の持つ種々の要求を、システムの特性を決定する重要な要素としてシステム設計の初期の段階から取り入れ、システムの全体的仕様と設計対象に反映させる。

○知識型アプローチ：利用者要求や設計仕様に関する知識（問題依存型知識）と、熟練したシステム設計者の持つ経験的知識（領域依存型知識）を利用して、設計プロセスを組み立てる。

この方法論に基づいて、コンピュータコミュニケーションシステム（CCS）の設計過程は、以下の2つの

要素を用いた、種々の設計ステージと設計プロセスによって形式化される。即ち、

- (A) 知識モデル：設計の各ステージの設計情報を表現する
- (B) 写像：連続した2つの知識モデル間の変換プロセスを規定する

ここで、各知識モデルは前記の問題依存型知識、個々の写像は領域依存型知識によって定義される。この枠組みに基づいたCCSの知識型設計プロセスを図1に示す。全体の設計プロセスは3種類の設計フェーズから成り、その各々が幾つかの設計ステージによって構成される。本稿で議論する初期要求獲得ステージは、要求定義フェーズの第一ステージであり、利用者要求指向アプローチを達成する上で重要な設計ステージとなっている。

2.2 要求仕様定義フェーズに於ける問題点

設計対象の機能や構造に関する設計仕様を規定するために、利用者（要求知識提供者）から与えられる種々の要求を反映した要求仕様が明確に定義されなければならない。しかしながら、これは困難な課題となっており、現状では満足すべき解が得られていない。本節では、この課題に関する問題点を整理しておく。

いま、要求仕様定義の過程が4種類の処理、即ち、

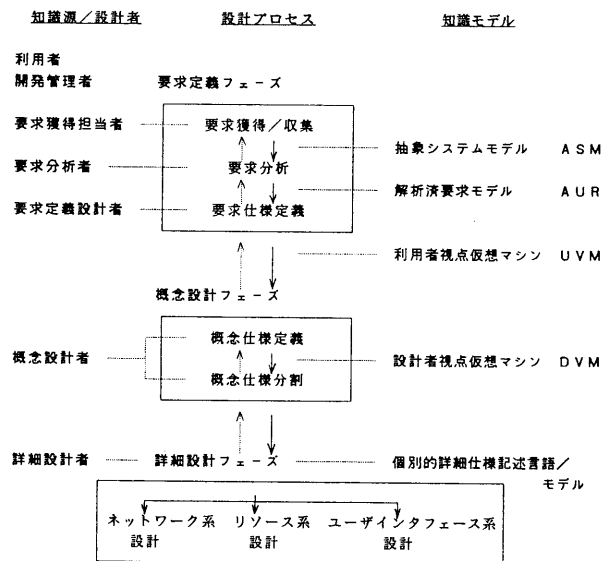


図1 CCSにおける知識型設計プロセス

1) 種々の要求事項（初期要求）の抽出（獲得）

2) 初期要求の分析と統合

3) 抽象化要求の生成

4) 形式化された要求仕様の作成

によって構成されるものとすれば、従来提案／利用されている種々の方式では、2)から4)までの処理を対象とし、1)は要求仕様設計者によるアドホックな作業に負っていることが多いと考えられる。従って、方法論的な観点からすれば、

P0) 要求仕様定義の上流工程を効果的に扱うための手法がほとんど得られていない

という問題が挙げられる。

これについて、要求知識を提供する利用者に関する観点から考えると、

P1) 利用者自身が自己の要求を正確に把握していることは稀である

P2) 利用者の要求には、暗黙的なものと明示的なものがあり、それらは状況の変化によって容易に変化する

P3) 利用者が自分自身で自己の要求を表現するための手段がない

という問題があり、また、要求仕様を作成する設計者や支援システム、或いは設計方法論に関する観点からみれば、

P4) システム設計者と利用者の間には要求に対する認識のギャップがある

P5) 利用者から効率的に要求情報を収集するための手段がない

P6) 初期要求を蓄積・管理するための効率的な手段が与えられていない

P7) 初期要求の獲得に対する枠組みを与える様な設計方法論がない

という問題が挙げられよう。本稿では、こうした問題を孕む初期要求の獲得フェーズに焦点を絞り、問題点を解決するための一手法について考察する。

3. 初期要求知識獲得方式：ORKAS/CCS

3.1 構成要素と初期要求知識獲得プロセス

初期要求知識の獲得方式の基本形態として、獲得対象となる初期要求知識は、それを提示する利用者自身によって自由に記述することとし、その後要求仕様設計者が種々の情報の補完や知識構造操作を行って、より洗練された要求仕様を定義することを前提とする。しかしながら、初期要求知識獲得ステージには前述した解決すべき幾つかの問題点が存在する。そこで、以下に示す5種類の方式要素を導入することにより、CCS設計のための初期要求知識獲得方式（Original Requirements Knowledge Acquisition Method for

CCS：ORKAM/CCS）を構成する。即ち、

[方式要素1] 知識型設計方法論が与える知識モデルに基づいて要求知識を整理する；

知識型設計方法論が与える知識モデルである抽象システムモデル（Abstract System Model：ASM）を参照モデルとして利用者が初期要求知識を形式的に記述し、要求仕様設計者がASMを介してその初期要求の内容を抽出することにより、利用者と要求仕様設計者の認識のギャップを埋める。これによって、前述した問題点 P1)、P2)、P4)、P7)に対処する。

[方式要素2] 利用者向きの初期要求知識表現言語を提供する；

ASMは初期要求知識表現モデルであるが、それ自身知識表現言語とはなっていない。そこで、ASMに基づいて定義される初期要求知識表現言語（Graph-based Requirements Knowledge Representation Language：GRKRL）を利用者に提供する。これは利用者向きの視覚的知識表現言語であり、これによって問題点 P3)に対処する。

[方式要素3] 知識獲得の手法を組み込んだ獲得プロセスとする；

利用者から初期要求知識を獲得することは、知識工学に於ける知識獲得と同一の問題に帰着される。そこで、一般的な知識獲得の枠組み¹³⁾、例えば、知識提供者からの直接的な獲得方式や知識提供者に対するインタビュー機能などを積極的に導入し、問題点 P2)、P3)、P5)などに対処する。

[方式要素4] 利用者による初期要求知識記述作業の支援環境を提供する；

利用者が自分で直接初期要求知識を記述するために、前項の知識獲得の枠組みと一体化したインタラクティブなマンマシンインタフェースを備えた作業環境とする。これにより、問題点 P5)、P6)に対処する。

[方式要素5] 専用知識ベースにより要求知識の一貫した管理を行う；

ASMに従って表現された初期要求知識は、以降の設計フェーズで利用するために専用の知識ベース(ASM-KB)によって蓄積・管理する。こうした知識ベース管理により、問題点 P6)、P7)に対処する。

上記の方式要素を用いることにより、ORKAM/CCSの初期要求知識獲得は、

- (1) 利用者による要求知識の投入
- (2) 要求仕様設計者による要求知識の補完と洗練

という2段階のプロセスとして構成される。各プロセスは、ASMの構成要素に対する、a)実体化、b)構造化、c)詳細化、というGRKRL上の知識構造操作の組み合わせから成り、各操作は直接操作環境、インタビュー機能、専用の知識ベースを用いた知識管理によって支援される。

3. 2 抽象システムモデル：ASM

ASMは、利用者と要求仕様設計者の双方に対するCCSのメンタルモデルであると同時に、初期要求知識を表現する知識表現モデルでもある。メンタルモデルとしてのASMの構造を図2に示す。知識表現モデルとしてのASMは、ASM記述(ASMD)によって定義される。

```
ASMD = < ASMD_ID, INPUT_OBJ, OUTPUT_OBJ,
          RESOURCE, USER, RELATION >
```

ここで、ASMD_IDは個々の利用者要求の識別名、入力と出力の対象を表わすINPUT_OBJとOUTPUT_OBJ、要求される機能を表わすRESOURCE、利用者に関する種々の情報USER、要求知識間の関係RELATIONであり、それぞれ以下の様に定義される。

```
INPUT_OBJ = OBJECT_DESCRIPTION or
            COMMAND_DESCRIPTION
OBJECT_DESCRIPTION =
< OBJ_NAME, OBJ_CLASS, DATA_TYPE,
  [MAX_VOLUME], LOCATION, [REQUIRED_RESOURCE],
  [ACCESS_MODE], [IMPORTANCE_LEVEL],
  [SECURITY_LEVEL], OWNER >
COMMAND_DESCRIPTION =
< COM_NAME, COM_CLASS, [RESOURCE_CLASS],
  ARGUMENT_OBJ_CLASSES, [PRIORITY_LEVEL] >
```

リソースへの入力対象は、ファイルや入力データなどのオブジェクト、若しくはそのリソースのコマンドである。オブジェクトは、一意の識別名(OBJ_NAME)を持ち、それは既定のオブジェクトのクラス(OBJ_CLASS)に属する。また、コマンドも何らかのコマンドクラス

補助リソース

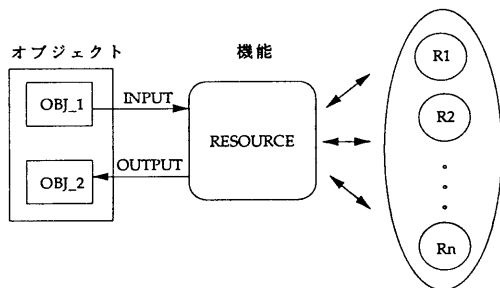


図2 抽象システムモデルASM

の要素となっている。上記の定義に於て、スクエアブラケットで囲まれたものは任意属性と呼ばれ、デフォルト値を利用することが可能であるが、それ以外のもは必須属性と呼ばれ、必ず何らかの値が割り付けられねばならない。

```
OUTPUT_OBJ = OBJECT_DESCRIPTION or
              RESPONSE_DESCRIPTION
RESPONSE_DESCRIPTION =
< RESP_NAME, RESP_CLASS, [RESOURCE_CLASS],
  [ARGUMENT_OBJ_CLASSES], [PRIORITY_LEVEL] >
```

リソースの出力は、入力と同様のオブジェクト、及び特定のクラスに属するリソース(RESOURCE_CLASS)に対する引数(ARGUMENT_OBJ_CLASSES)を伴った応答によって定義される。

```
RESOURCE = FUNCIONAL_RESOURCE or
            SYSTEM_RESOURCE
FUNCIONAL_RESOURCE =
< RES_NAME, RESOURCE_CLASS, "F_RES",
  REQUIRED_INPUT_OBJ, REQUIRED_OUTPUT_OBJ,
  [PROCESSING_TIME_MARGIN], PROCESS_TYPE,
  [AUXILIARY_RESOURCE_CLASSES] >
SYSTEM_RESOURCE =
< RES_NAME, RESOURCE_CLASS, "S_RES",
  REQUIRED_INPUT_OBJ, REQUIRED_OUTPUT_OBJ,
  [PROCESSING_TIME_MARGIN], PROCESS_TYPE,
  [LOCATION], AUXILIARY_RESOURCE_CLASSES >
```

入力/出力オブジェクトの処理を行うリソースは、機能リソースとシステムリソースに大別され、それぞれあるリソースクラスの要素である。機能リソースは既定のタイプ" F_RES"を持つ機能名(RES_NAME)によって指定されるリソースで、その実現形態が未だ定まっていないもの、システムリソースはタイプ" S_RES"を持つシステム名(RES_NAME)によって指定されるリソースである。これは、実現形態が何らかのソフトウェアアシスタントとなり、原則としてその稼働場所(LOCATION)を指定することができる。






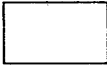


```
USER = < USER_NAME, USER_ID, USER_TYPE,
         USER_GROUP_ID, LOCATION, [TERMINAL_TYPE],
         USER_MODEL_ID, OWN_ASMD_ID >
```

利用者に関する情報は、利用者名やその所属、場所などに関する情報の他に、使用する端末装置のタイプ(TERMINAL_TYPE)、システムが管理・利用するユーザモデル(USER_MODEL_ID)、個々の利用者が定義した要求知識へのポイント(OWN_ASMD_ID)などを持つ。

```
RELATION = < RELATION_TYPE, RELATIONL_ASMD_ID >
```

関連する要求知識(RELATIONAL_ASMD_ID)を既定の種類の関係(RELATIONAL_TYPE)によって関係付けるためのものである。

表 1 初期要求知識表現言語：GRKRL

Name	Surface Icon	Internal Attributes	Domain
Normal Object		Object_Name Object_Class Data_Type Max_Volume Location	Finite String [1,30] <Object Class> Binary/Numeric/Text Integer [0,1000] <KByte> Address_List
Terminal Object		Required_Resource Access_Mode Importance_Level Security_Mode	<Resource Class> R/W/RW High/Middle/Low High/Middle/Low
Group Object		Owner Requirement_Group Embedding_Level	<Owner Class > <R_Group Class> Integer [0,5] <>
Command Object		Command_Name Command_Class Resource_Class Argument_Obj_Class Priority_Level Requirement_Group Embedding_Level	Finite String [1,30] <Command Class> <Resource Class> <Object Class> Integer [1,10] <> <R_Group Class> Integer [0,5] <>
Response Object		Resp_Name Resp_Class Resource_Class Argument_Obj_Class Priority_Level Requirement_Group Embedding_Level	Finite String [1,30] <Response Class> <Resource Class> <Object Class> Integer [1,10] <> <R_Group Class> Integer [0,5] <>
Functional Resource		Resource_Name Resource_Class Resource_Type Required_Input_Obj Required_Output_Obj Proc_Time_Margin Proc_Type	Finite String [1,30] <Resource Class> "F_RES" or "S_RES" Object_Property_List Object_Property_List Real_Number_list RT/OL/BT
System Resource		Location Auxiliary_Res_Class Requirement_Group Embedding_Level	Address_List <Resource Class> <R_Group Class> Integer [0,5] <>
Link		Link_Name Source_Node Destination_Node Activation_Condition Embedding_Level Requirement_Group	Finite String [1,30] <Object> or <Resource> <Object> or <Resource> Condition_Description Integer [0,5] <> <R_Group Class>

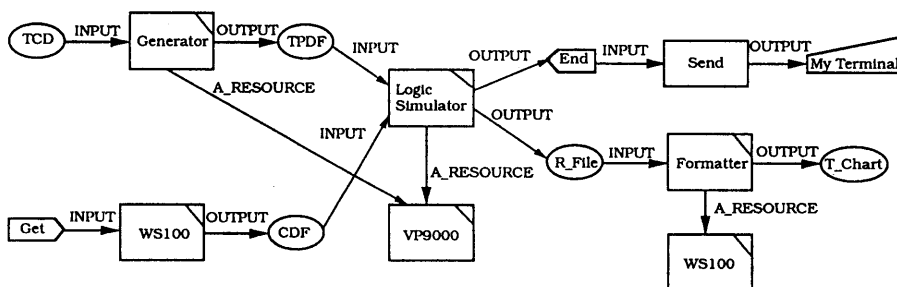


图 3 初期要求知識記述例

3.3 初期要求知識表現言語：GRKRL

ASMに基づいて定義される初期要求知識表現言語GRKRLは、ノードとリンクを表現プリミティブとするグラフ構造によって初期要求知識を表現するための言語系である。

ノードは、ASMの入力/出力のオブジェクト/リソースに対応し、表1に示す様に、7つのタイプに分類される。これらのノードは、それぞれASMDの定義で与えられるノードの内部属性と共に、グラフ構造として表現される要求知識を管理するための属性があり、各属性は固有の定義域を持つ。例えば、一般的なオブジェクトNormal Objectは、OBJECT_DESCRIPTIONで与えられる10個の属性を持ち、例えば、OBJ_CLASSはObject_Classという属性名を持ち、その属性値の定義域はオブジェクトクラス<Object Class>のクラス階層に属するオブジェクトの識別名となる様に定義されている。

一方、リンクはノード間の種々の関係を表わすもので、任意の関係名を持つ名前付き有向リンクとして定義される。通常、上記の種々のノードの内部属性を記述するために用いられ、この場合、対応する属性名をリンク名として用いる。図3に記述例を示す。

更に、規定のノードとリンクを組み合わせることで、特定の要求知識の抽象化された構造を表わす表現テンプレートなども定義できる。

4. 初期要求知識獲得支援システム：ORKAS/CCS

4.1 基本構成

ORKAS/CCSに基づいて、初期要求知識獲得支援システム(Original Requirements Knowledge Acquisition support System for CCS:ORKAS/CCS)が構成される。その概念構成を図4に示す。ORKAS/CCSは、利用者や要求仕様設計者がGRKRLを用いてインタラクティブに要求知識獲得作業を行うインタフェースモジュールIF、IF上での種々の知識獲得操作をモニタして支援機能モジュールを起動する監視モジュールMM、IMを介して与えられる種々の知識操作を処理する知識操作モジュールHM、記述された要求知識に基づいて助言や質問を生成するインタビューモジュールIM、及び獲得された要求知識を蓄積・管理する専用知識ベースシステムモジュールKBMから構成される。

こうしたORKAS/CCSを用いて、利用者と要求仕様設計者による2段階のプロセスによって初期要求知識の獲得が行われる。各プロセスは、3.1で述べた諸操作を実行する4段階の処理ステップ、即ち、

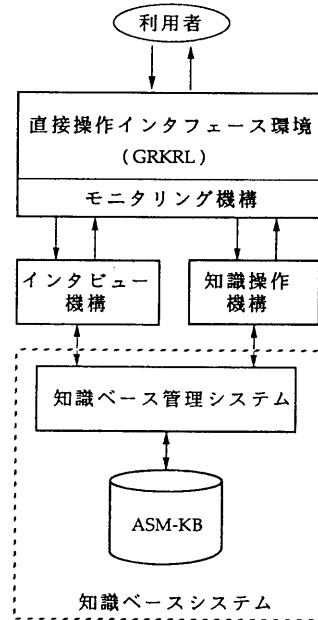


図4 初期要求知識獲得支援システム ORKAS/CCS

- 1) 要求要素選択（要求プリミティブ選択操作）；
要求知識の内容に基づく、表現プリミティブ/表現テンプレートの選択
- 2) 要求知識要素の記述（要求プリミティブ実体化操作）；
ノード/リンクへの命名と部分構造の決定
ノード/リンクの内部属性値の割り付け
- 3) 要求知識構造の操作（要求知識構造化操作）；
要求知識表現（ASMD）の結合・関係付け
ASMD表現上の知識構造変換
- 4) 要求知識の洗練（要求知識詳細化操作）；
要求知識の補完・詳細化
要求知識構造の更新

によって構成され、利用者は1)~4)のステップを反復的に利用して自己の要求が十分表現されるまでインタラクティブに獲得作業を行う。原則として、これらのステップは利用者主導で駆動されるが、必要に応じてインタビュー機構を用いたガイダンスによって暗黙的に制御することも可能である。

4.2 支援機構

ORKAS/CCSの知識獲得作業を支援するための機能要素の概要を述べる。

直接操作環境（IF）：GRKRLの表現プリミティブ

ブを直接操作して初期要求知識を記述するためのマルチウインドウ環境である。表現プリミティブ/テンプレートは、ウインドウ上のメニュー、或いはポップアップメニューなどから直接選択し、リンク付けやメニューに割り当てられた操作コマンドによって初期要求知識の記述を行う。図5に、IFのモックアップモデルによる環境のイメージを示す。

モニタリング機構 (MM) : 利用者が行うインタラクティブな要求知識の記述作業をバックグラウンドでモニタし、利用者がどの様な対象にどの様な操作を施したかという情報を得る。更に、利用者によって使用された表現プリミティブの内部属性や表現テンプレートの構造に関する知識の定義を参照することにより、入力値の適合性や、他の要求知識の内容との相互関係を調べ、検査情報を生成する。

インタビュー機構 (IM) : MMの情報に基づいて、利用者に対するフィードバックとして、ASMDのノードの必須属性に対する誤情報の訂正、不足情報の要求、任意属性に対する属性値選択などに関する情報要求メッセージを提示し、要求知識の補完を促す。また、利用者から発せられる質問 (HELP要求) に対する応答

なども行うことによって、インタビュー機構はCCSの利用者の要求知識記述作業の支援とガイドを行う。この機構は、一般的な知識型システムの知識ベース構築で利用される知識獲得システムに装備されるインタビュー機構を要求知識獲得ドメインに特化させたものとなっている。

知識構造操作機構 (HM) : GRKRLを用いたインタラクティブな知識記述過程をサポートするために、表現プリミティブ/テンプレートなどの表現要素の検索・複写、表現要素に対する種々の値の割り付けによる実体化、実体化された表現要素を他の表現要素と関連付けて要求知識の構造を作成するためのリンク付けなどを行う基本的なグラフ構造操作機能、更には、幾つかの表現要素から成る要求知識の集合に対して行われるグルーピング操作、埋め込み (マクロノード化) 操作、ノード分割操作などの知識表現変換操作などを提供する。また、利用者によって任意に定義されるノードや名前付き有向リンクなどに関する標準的な知識構造変換操作を容易にするため、基本的な操作を組み合わせた知識表現変換規則を作成し、それを適宜利用できる機能も付加する。

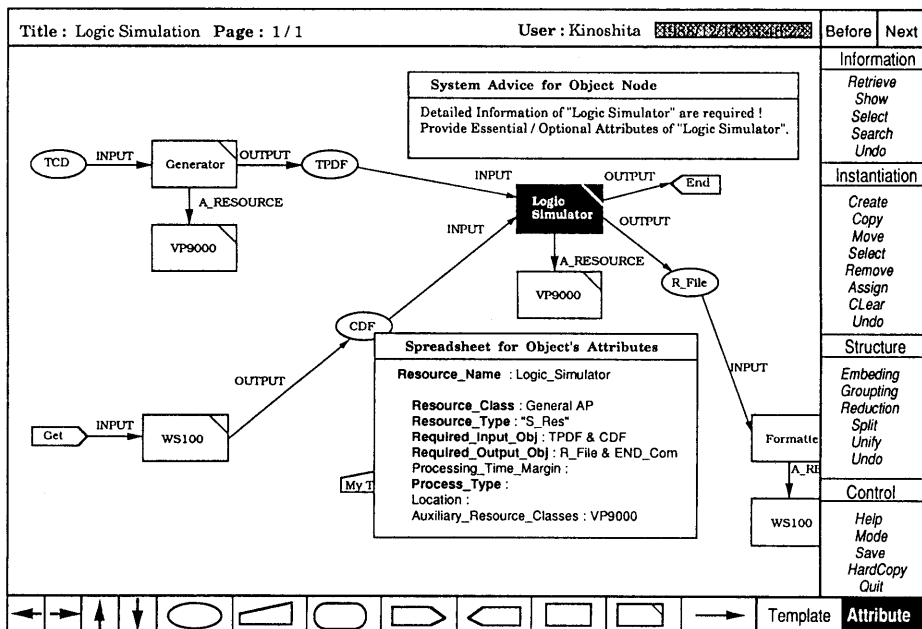


図5 ORKAS/CCSインタフェース環境の例

知識ベース管理 (KBM) : GRKRLのグラフ構造として獲得された表層表現は、形式的な処理が可能な内部表現に変換され、初期要求知識ベース(ASM-KB)に格納される。これにより、初期要求知識の一貫した管理を行うと共に、知識型設計方法論の次段の設計フェーズ^{(11)・(14)}に対する設計情報を供給する。

4. 3 実現上の課題

現在、フレーム型の知識表現、及びその知識ベースシステムに基づいて、ORKAS/CCSの基本設計を進めている。こうしたORKAS/CCSを実現する上で、

(D1) 表層のグラフ表現主体の知識記述形式は、記述言語形式に比べて記述規模が大きくなるので、その表示方法や管理方法

(D2) 設計ドメインの深い知識に基づいたインタビュー戦略の抽出と組み込み方法

(D3) 獲得された要求知識のより強力な(自動的な)検証機構との組み合わせ方法

など、幾つかの問題が残されている。これらの課題は、バックエンドのフレーム型知識ベースシステム上で取り扱いたいと考えている。

5. おわりに

本稿では、知識工学の枠組みを援用したCCSの設計タスクに於ける利用者の初期要求知識獲得方式の基本的な枠組みを提案した。その主要な特徴は、CCSの知識型設計方法論で与えられる知識モデルに基づく初期要求知識表現言語、及びインタビュー機構などの知識工学の技法を適用したインタラクティブな要求知識獲得プロセスである。要求仕様定義の問題に対して十分強力な手法が得られていないのが現状であるが、知識工学的手法を設計ドメインの枠組みと組み合わせることで効果的に利用することは、より高度な要求仕様定義を行うためのひとつの有力な手段と見做すことが出来るであろう。

文献

- [1] T. Kinoshita, et al., "Knowledge-based Design Support System for Computer Communication System", IEEE Journal SAC, Vol. 6, No. 5, pp. 850-861, 1988.
- [2] 木下, 他, " コンピュータコミュニケーションシステムの知識型設計支援について", マルチメディア通信と分散処理研究会資料 37-9, 情報処理学会, 1988.
- [3] 木下, 他, " 多階層知識獲得モデル", '人工知能の枠組み' シンポジウム予稿集, 情報処理学会, pp. 141-150, 1987.
- [4] 菅原, 他, " 分散処理システムの機能要求定義", マルチメディア通信と分散処理研究会資料 41-5, 情報処理学会, 1989.