慶應義塾大学学術情報リポジトリ
Keio Associated Repository of Academic resouces

Title	スクリプトにもとづく質問応答システム
Sub Title	A question answering system bosed on scriptal knowledge
Author	難波, 和明(Nanba, Kazuaki)
Publisher	慶應義塾大学大学院社会学研究科
Publication year	1983
Jtitle	慶應義塾大学大学院社会学研究科紀要 : 社会学心理学教育学 (Studies in sociology, psychology and education). No.23 (1983.) ,p.1- 30
JaLC DOI	
Abstract	
Notes	論文
Genre	Departmental Bulletin Paper
URL	https://koara.lib.keio.ac.jp/xoonips/modules/xoonips/detail.php?koara_id=AN0006957X-00000023-0001

慶應義塾大学学術情報リポジトリ(KOARA)に掲載されているコンテンツの著作権は、それぞれの著作者、学会または出版社/発行者に帰属し、その権利は著作権法によって保護されています。引用にあたっては、著作権法を遵守してご利用ください。

The copyrights of content available on the KeiO Associated Repository of Academic resources (KOARA) belong to the respective authors, academic societies, or publishers/issuers, and these rights are protected by the Japanese Copyright Act. When quoting the content, please follow the Japanese copyright act.

「スクリプトにもとづく質問応答システム」

A questiion answering system bosed on Scriptal Knowledge

難　波　和　明
Nanba Kazuaki

Question answering is the most important problem when we attempt to aralyze the mechanism of human understanding. We can specify what kind of knowledge we need to answer questions about a text through computer programming. This paper describes and illustrates a computer program which reads stories and answers questions.

質疑応答という行為は，単に理解されたどうかの判定材料として用いられるばかりでなく，ソクラテスの対話のように，理解を深めるために用いることができる。さらに佐伯[1]の Knowing how から Knowing that を導びきだす内的対話システムのように，自問自答という形で，深く人間の理解過程にかかわっている。この人間の理解過程の中で重要なはたらきをする質問応答を考えるための第一歩として，Lernert の質問応答システム[2]をもとにして開発した，スクリプトによる文章の理解と，それに対する質問応答システム Apple II 上の Apple LISP による Micro QA を解説する。

(1) Micro QA の概説

Micro QA は，テキスト文についての質問に対して答える。したがって，まずテキスト文の理解をしなければならない。そのために，まずテキスト文を，その意味をとりだした表現に変換する。この意味表現の形式には，さまざまなものがあるが，Micro QA では，Schank[3] の CD 表現（概念依存表現）を用いている。これは動詞を中心にして，主語，目的語などは，動詞の格として表現していこうというものである。また，動詞自体は，その基本的意味から，11～13の意味原素（基本アクト）によって表現されている。この格と，動詞の意味原素との組みあわせを軸として，CD 表現は，すべての文の意味を表現しようとしている。Micro QA においては，このテキスト文の CD 表現への変換をおこなう PARSER は，英語の文については，Micro ELI[4][1]をもちいる。また日本語については，北川，難波の Apple JUM[5]によっておこなう予定であるが，辞書部分だけが，まだ未完成である。ここでは，すでに動いている Micro ELI によるものだけを示す。

テキスト文から意味をとりだしただけでは，テキスト文を理解したことにはならない。テキスト文中で省略されている部分を，何らかの知識でおぎなってやらなければならず，さらに，階層的に構造化してやらなければならない。このテキスト文の省略をおぎない，階層的に構造化するために，ここでは，スクリプト[6]を用いた。スクリプトは，非常によく使われる手順的な知識の表現形式であり，スクリプトにあてはめることによって，省略された部分および階層的構造の情報がおぎなわれる。このスクリプトをテキスト文に応用する部分は，SCRIPT APPRIER Micro SAM[4][1]による。また，テキスト文が，単なる手順的知識だけでは理解できず，作文者の意図をさぐることが必要な場合には，PLAN APPRIER Micro PAM[4] を用いることができる。Micro PAM も Apple II 上ですでに動いており，Micro QA に接続することは容易である。

また質問文も，Micro ELI，および Apple JUM に

よって CD 表現に変換される。このテキスト文を CD 表現にし，さらにスクリプトによって省略をおぎない，階層構造化されたものと，質問文を表現にしたものとをつきあわせ，質問に応じた答をみつけだし，その CD 表現を返すのが，QUESTION ANSWERER Micro QA である。

Miro QA によって答として出された CD 表現を日本語の文になおすのが，日本語 GENERATOR である。これは，北川，難波によって，現在製作中であり，単文までしか可能ではないので，ここでは残念ながら述べることはできない。

以上，図を参照していただきたい。

（2） テキスト文と質問文

今回，Micro QA があつかったテキスト文と質問文は，Lernert[2] からとったもので，図2のようなものである。このテキスト文と質問文を Micro ELI によって CD 表現になおしたものが，図3，図4 である。また Micro ELI 実行例の一部を図10に示す。

（3） レストラン，スクリプト

図3のテキスト文の CD 表現は，この場合は，レストラン，スクリプトに，Micro SAM によってあてはめられる。この場合に用いたレストラン，スクリプトを図5に，レストラン，スクリプトによって省略をおぎなわれ，階層構造化されたテキスト文の CD 表現を図6に，その階層構造を図7に示す。

図5のレストラン，スクリプトは，コーヒー，ショップといわれるレスランにおいて，通常，客のまわりでおこるできごとを因果関係が連続するようにつないでいったものである。詳しくは引用文献（1）を参照していただきたい。

レストラン，スクリプトの階層構造は，図7のように，Script name—condition, scene—Maincons—個々の文という4段階になっている。entry および result condition は，スクリプトに入いる前の状態および，その結果の状態を示す文であることを示している。entering, ordering eating, exiting の4つのscene name は，レストラン，スクリプトが，この4つの場面で構成されていることを示し，それぞれの文が，どの場面に属しているかを示している。また，Maincons とは，各場面の中心的なできごとであることを示している。さらに一行右へずれているのは，スクリプト中に対応する記述がなく，テキスト中にのみ含まれている文であることを示している。何の記述もないのは，一般的なスクリプト中の文である。

Micro QA は，この階層化された知識ベースをもとにして，質問に対する答えと検索，生成する。この場合は，階層化のための情報のすべてを，レストラン，スクリプト内にもたせたが，これは，最も安易なやり方である。この他に，プランによる階層化，さらに，プランからスクリプトをつくりだし，プランとスクリプトをつかいわける MOPS[7] によるものなどがあり，また階層化をおこなうために質問応答をおこなうシステムも考えられる。いずれにしても，人間の理解をさぐるためには，この階層化をいかにおこなうかという部分の研究が最も重要である。

（4） Micro QA の動き

（4．1）質問文の識別

Micro QA の実行結果を図8に，プログラムを図9に示す Micro QA は，まず入力された質問文のタイプを識別する。入力された CD 文中に，leadto または cancanse という部分があれば，その内容をひきだし，それによってwhyではじまる質問文と，what ではじまる質問文は識別される。以上の2種類でない場合は，did で始まる疑問文か，who で始まる疑問文である。did ではじまる疑問文の場合は，CD 表現の最初の動詞の意味をあらわす部分（基本アクト）に *ATRANS* のように，* がつけられているのを手がかりにする。did ではじまる疑問文でない場合は，who ではじまる疑問文として処理される。この場合，who および，whom は，すべて一つの処理法で処理されるので，その間での識別は，必要ない。これらをまとめてみると図11のようになる。

なお今回は，how, where, which, which whose ではじまる疑問文，および，選択疑問文はとりあつかっていないが，これらをあつかうためのプログラムを書きかえが容易なように，Micro QA は設計されている。また各階層構造ごとに，それぞれ別のデータベースがつくられる。

（4．2）各質問文の処理

（4．2．1）did ではじまる疑問文の処理

did ではじまる疑問文の場合は，入力された省略をおぎない，階層化されたテキスト文（今後はストーリイ文とよぶ）中に，質問文の CD 表現と，関数 MATCH # によって一致しているとみなされた CD 表現がある場合には，その CD 表現を返す。ない場合には，質問文中の主

語が記録されている FORCUS の内容を変数化して，もう一度一致する CD 表現をさがす。一致する CD 表現があれば，そのCD表現を，なければ (I DONT KNOW) を返す。

たとえば，図8の4番目の<<<<<QUESTION IS の場合，図2のQ4に対応している。この“ウェイターは，キンタロウにメニューをわたしましたか”という文の場合，ストーリイ文中に一致する CD 表現がないため，これを“だれがキンタロウにメニューをわたしましたか”という CD 表現に書きかえ，次に述べる who ではじまる疑問文と同様に処理し，“ホステスがキンタロウにメニューをわたしました”と答えるのである。

（4．2．2）who ではじまる疑問文

who ではじまる疑問文の場合，図4の5番目の CD 表現のように，だれがにあたる部分に *#* がつけられている。したがって，*#* のついている部分以外の部分が，ストリー文中の CD 表現の中の対応する部分と一致すれば，その CD 表現を答として返す。それ以外は (I DONT KNOW) と返す。whom ではじまる文は，TO の格や FROM の格に *#* がついているだけなのでまったく同様に処理される。

（4．2．3）why ではじまる疑問文

why ではじまる疑問文の場合，まずその CD 表現と一致する CD 表現を，ストーリイ文中の中でさがしだす。そのさがしだした CD 表現が MAINCONS である場合には，その次の MAINCONS を返す。(例，図2の Q12，図8の12番目) ただし，一番最初の MAINCONS の場合には，一番目の (CONDITION (RESULT)) を返す。(例，図2の，Q1，図8の一番目) 一致した CD 表現が，特にサイズについて言及している場合には，そのサイズを結果として生じさせている CD 表現の前の CD 表現を返す。(例，図2の，Q10，図8の 10 番目) これは，サイズを結果としている文は，単にそのサイズを決定しているにすぎないので，因果関係的には，そのもう一つ前までさかのぼらなければならないためである。一致した CD 表現が，MAINCONS でなく，サイズに言及もしていない場合，その SCENE をしらべ，その SCENE の中の MAINCONS が，その文より後にある場合は，MAINCONS を(例，図2のQ6，図8の6番目)，前にある場合は，次の CD 表現を答える(例，図2のQ3，図8の3番目)。SCENE がない場合には，次の CD 表現を答える。

（4．2．4）what ではじまる疑問文

この場合は，図2の7番目だけであるので，what happened ではじまるものだけに対応している。一致した CD 表現の SCENE をしらべ，その SCENE 中の MAINCONS が，一致したCD表現よりも後にある場合は，一致した CD 表現の次のCD表現から，MAINCONS までのすべての CD 表現を，前にある場合には，その SCENE の残りの文全部を答として返す。

(5) む す び

実際に質問応答システムをつくってみると，質問に答えるためには，何を理解していなければならないかが，非常によくわかる。この Micro QA の場合，テキスト文の省略がおぎなわれ，さらに階層構造化されていなければ，理解しているとはいえないわけである。今回は，これらをすべてスクリプトに入れこんでしまったが，もちろん人間の理解のしかたは，そのような単純なものではない。今後は，プランにもとづく質問応答による階層構造化の実験，プロトコル分析，コンピューター・シミュレイションを幅広くとりいれた研究が重要であろう。

〔引 用 文 献〕

1) 安西，佐伯，難波「LISP で学ぶ認知心理学 2. 問題解決」東大出版会，1982.
2) Lehnert.「The process of question answering」Lawrence Erlbanm Associate, 1978.
3) Schank「Conceptual information processing」Noth'Holland, 1975.
4) Schank, Riesbeck「Inside computer uhderstanding」Lawrence Erlbaum Associate, 1981.
5) 北川，昭和53年度慶応大学管理工学科卒業論文（予定）
6) Schank & Abelsor,「Scripts, plans, goals and understanding: An inquiry into human knowledge struc ture」Lawrence Erlbaum Associate, 1977.
7) Schank,「Language and memory」Cognitive Science 4, 243-24.

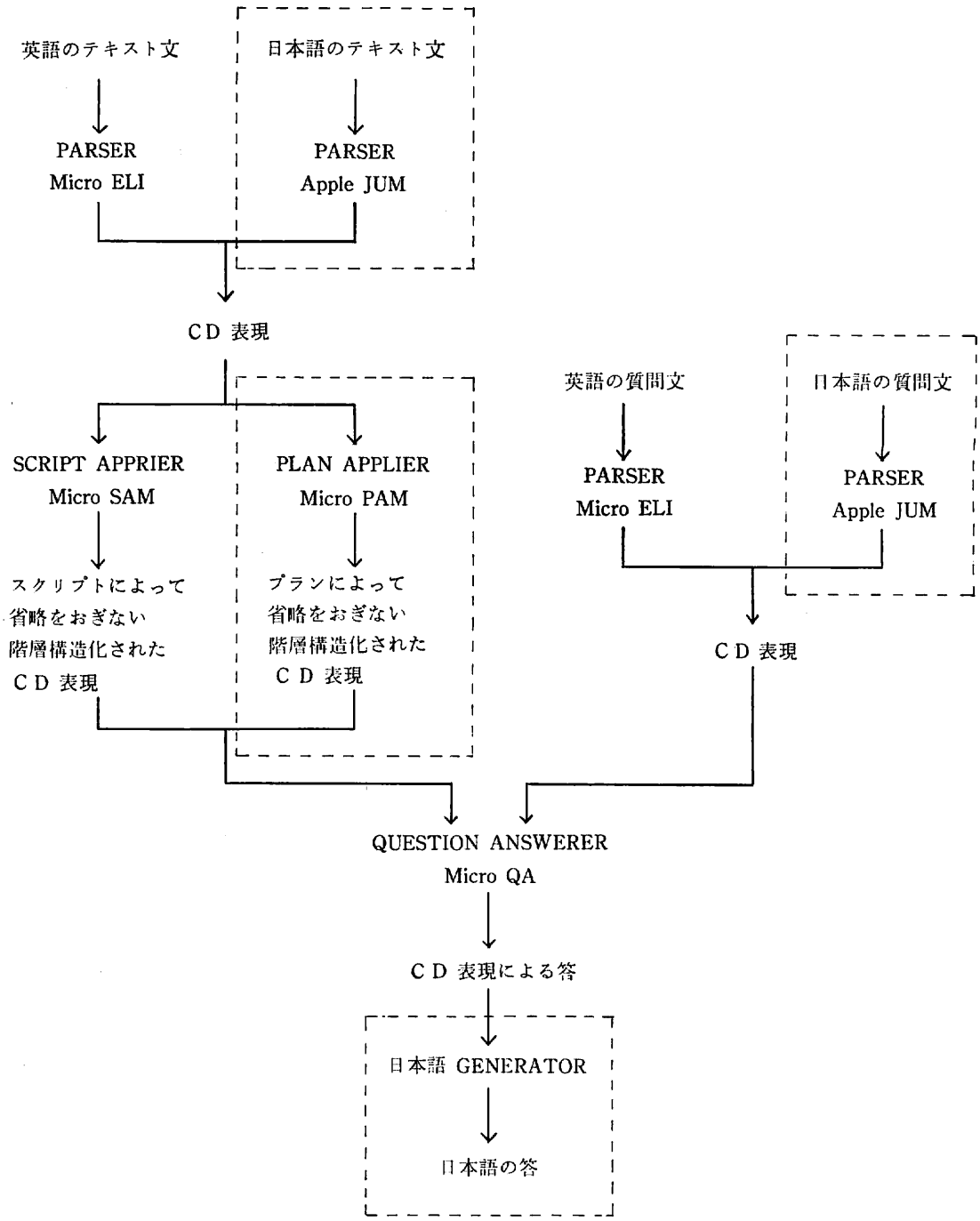

図1 Micro QA の概観（この論文であつかったのは破線で囲んだ部分以外の部分）

テキスト文

1. (KINTARO WENT TO A RESTAURANT)
2. (THE HOSTESS SEATED KINTARO)
3. (THE HOSTESS GAVE KINTARO A MENU)
4. (THE WAITER CAME TO THE TABLE)
5. (KINTARO ORDERED A LOBSTER)
6. (HE WAS SERVED QUICKLY)
7. (HE LEFT A LARGE TIP)
8. (HE LEFT THE RESTAURANT))

質 問 文

Q 1. (WHY DID KINTARO GO TO THE RESTAURANT)
Q 2. (DID KINTARO GO TO A TABLE)
Q 3. (WHY DID KINTARO GO TO A TABLE)
Q 4. (DID THE WAITER GIVE KINTARO A MENU)
Q 5. (WHO GAVE KINTARO A MENU)
Q 6. (WHY DID THE HOSTESS GIVE KINTARO A MENU)
Q 7. (WHAT HAPPENED WHEN THE HOSTESS GAVE KINTARO A MENU)
Q 8. (DID KINTARO ENJOY THE LOBSTER)
Q 9. (DID THE WAITER GIVE KINTARO THE CHECK)
Q 10. (WHY DID KINTARO LEAVE THE WAITER A LARGE TIP)
Q 11. (DID KINTARO PAY THE BILL)
Q 12. (WHY DID KINTARO PAY THE CHECK))))

図 2　Micro QA のテキスト文と質問文

```
 ((PTRANS (ACTOR (PERSON (NAME (KINTARO))))
          (OBJECT (PERSON (NAME (KINTARO))))
          (TO (PLACE (TYPE (RESTAURANT)))))
  (MTRANS (ACTOR (PERSON (ROLE (HOSTESS))))
          (OBJECT (FACT (RESULT (MOVE (ACTOR (PERSON (NAME (KINTARO))))
                                      (TO (CHAIR))
                                      (INSTRUMENT (PTRANS (ACTOR (PERSON (NAME (
KINTARO))))
                                                          (OBJECT (PERSON (NAME
(KINTARO))))
                                                          (TO (TABLE))
                                                          (INSTRUMENT (MBUILD (A
CTOR (PERSON (NAME (KINTARO))))
                                                                              (O
BJECT (LOC (ENABLE (MOVE (ACTOR (PERSON (NAME (KINTARO))))

                                 (TO (CHAIR))))))))))))))
          (TO (PERSON (NAME (KINTARO))))
          (FROM (PERSON (ROLE (HOSTESS)))))
  (ATRANS (ACTOR (PERSON (ROLE (HOSTESS))))
          (OBJECT (BOOK (TYPE (FOODLIST))))
          (TO (PERSON (NAME (KINTARO))))
          (FROM (PERSON (ROLE (HOSTESS)))))
  (PTRANS (ACTOR (PERSON (ROLE (WAITER))))
          (OBJECT (PERSON (ROLE (WAITER))))
          (TO (TABLE)))
  (MTRANS (ACTOR (PERSON (NAME (KINTARO))))
          (OBJECT (FACT (RESULT (ATRANS (OBJECT (FOOD (TYPE (LOBSTER))))
                                        (TO (PERSON (NAME (KINTARO)))))))))
          (FROM (PERSON (NAME (KINTARO)))))
  (MOVE (OBJECT (ACTION (RESULT (IS (ACTOR (PERSON (SEX (MALE))))
                                    (STATE (PLEASE (VAL (10))))))))))
  (ATRANS (ACTOR (PERSON (SEX (MALE))))
          (OBJECT (TIP (SIZE (LARGE))))
          (FROM (PERSON (SEX (MALE)))))
  (PTRANS (ACTOR (PERSON (SEX (MALE))))
          (OBJECT (PERSON (SEX (MALE))))
          (FROM (PLACE (TYPE (RESTAURANT))))))
```

図 3　テキスト文の CD 表現

```
((PTRANS (ACTOR (PERSON (NAME (KINTARO))))
         (OBJECT (PERSON (NAME (KINTARO))))
         (TO (PLACE (TYPE (RESTAURANT))))
         (FOCUS (PERSON (NAME (KINTARO))))
         (LEADTO (*## WHYVAR)))
 (*PTRANS* (ACTOR (PERSON (NAME (KINTARO))))
           (OBJECT (PERSON (NAME (KINTARO))))
           (TO (TABLE))
           (FOCUS (PERSON (NAME (KINTARO)))))
 (PTRANS (ACTOR (PERSON (NAME (KINTARO))))
         (OBJECT (PERSON (NAME (KINTARO))))
         (TO (TABLE))
         (FOCUS (PERSON (NAME (KINTARO))))
         (LEADTO (*## WHYVAR)))
 (*ATRANS* (ACTOR (PERSON (ROLE (WAITER))))
           (OBJECT (BOOK (TYPE (FOODLIST))))
           (TO (PERSON (NAME (KINTARO))))
           (FROM (PERSON (ROLE (WAITER))))
           (FOCUS (PERSON (ROLE (WAITER)))))
 (ATRANS (ACTOR (*## WHOVAR))
         (OBJECT (BOOK (TYPE (FOODLIST))))
         (TO (PERSON (NAME (KINTARO))))
         (FROM (*## WHOVAR))
         (FOCUS (*## WHOVAR)))
 (ATRANS (ACTOR (PERSON (ROLE (HOSTESS))))
         (OBJECT (BOOK (TYPE (FOODLIST))))
         (TO (PERSON (NAME (KINTARO))))
         (FROM (PERSON (ROLE (HOSTESS))))
         (FOCUS (PERSON (ROLE (HOSTESS))))
         (LEADTO (*## WHYVAR)))
 (ATRANS (ACTOR (PERSON (ROLE (HOSTESS))))
         (OBJECT (BOOK (TYPE (FOODLIST))))
         (TO (PERSON (NAME (KINTARO))))
         (FROM (PERSON (ROLE (HOSTESS))))
         (FOCUS (PERSON (ROLE (HOSTESS))))
         (LEADTO (*## WHATVAR)))
 (*IS* (ACTOR (PERSON (NAME (KINTARO))))
       (STATE (ENJOY (VAL (5))))
       (FOCUS (PERSON (NAME (KINTARO))))
       (INSTRUMENT (INGEST (ACTOR (PERSON (NAME (KINTARO))))
                           (OBJECT (FOOD (TYPE (LOBSTER))))))
 (*ATRANS* (ACTOR (PERSON (ROLE (WAITER))))
           (OBJECT (CHECK))
           (TO (PERSON (NAME (KINTARO))))
           (FROM (PERSON (ROLE (WAITER))))
           (FOCUS (PERSON (ROLE (WAITER)))))
 (ATRANS (ACTOR (PERSON (NAME (KINTARO))))
         (OBJECT (TIP (SIZE (LARGE))))
         (TO (PERSON (ROLE (WAITER))))
         (FROM (PERSON (NAME (KINTARO))))
         (FOCUS (PERSON (NAME (KINTARO))))
         (LEADTO (*## WHYVAR)))
 (*ATRANS* (ACTOR (PERSON (NAME (KINTARO))))
           (OBJECT (MONEY))
           (FROM (PERSON (NAME (KINTARO))))
           (FOCUS (PERSON (NAME (KINTARO))))
           (INSTRUMENT (MTRANS (ACTOR (PERSON (NAME (KINTARO))))
                               (OBJECT (BILL))
                               (TO (CP (PART (PERSON (NAME (KINTARO))))))))
 (ATRANS (ACTOR (PERSON (NAME (KINTARO))))
         (OBJECT (MONEY))
         (FROM (PERSON (NAME (KINTARO))))
         (FOCUS (PERSON (NAME (KINTARO))))
         (LEADTO (*## WHYVAR))))
```

(20×20)

図4 質問文の CD 表現

```
(PUT (QUOTE *RESTAURANT)
     (QUOTE EVENTS)
     (QUOTE ((IS (ACTOR (*VAR* CUSTOMER))
                 (STATE (HUNGER (VAL (5))))
                 (CONDITION (ENTRY)))
             (POSS (ACTOR (*VAR* CUSTOMER))
                   (OBJECT (MONEY))
                   (CONDITION (ENTRY)))
             (MBUILD (ACTOR (*VAR* CUSTOMER))
                     (OBJECT (FACT (RESULT (PTRANS (ACTOR (*VAR* CUSTOMER))
                                                   (OBJECT (*VAR* CUSTOMER))
                                                   (TO (RESTAURANT))))))
                     (SCENE (ENTERING)))
             (PTRANS (ACTOR (*VAR* CUSTOMER))
                     (OBJECT (*VAR* CUSTOMER))
                     (TO (*VAR* RESTAURANT))
                     (SCENE (ENTERING))
                     (SENTENCE (MAINCONS)))
             (ATTEND (ACTOR (*VAR* CUSTOMER))
                     (OBJECT (EYES))
                     (TO (TABLE))
                     (SCENE (ENTERING)))
             (MBUILD (ACTOR (*VAR* CUSTOMER))
                     (OBJECT (LOC (ENABLE (MOVE (ACTOR (*VAR* CUSTOMER))
                                                (TO (CHAIR))))))
                     (SCENE (ENTERING)))
             (PTRANS (ACTOR (*VAR* CUSTOMER))
                     (OBJECT (*VAR* CUSTOMER))
                     (TO (TABLE))
                     (SCENE (ENTERING)))
             (MOVE (ACTOR (*VAR* CUSTOMER))
                   (TO (CHAIR))
                   (SCENE (ENTERING)))
             (PTRANS (ACTOR (*VAR* *A*WAITER))
                     (OBJECT (*VAR* *A*WAITER))
                     (TO (TABLE))
                     (SCENE (ORDERING)))
             (ATRANS (ACTOR (*VAR* *B*WAITER))
                     (OBJECT (BOOK (TYPE (FOODLIST))))
                     (TO (*VAR* CUSTOMER))
                     (FROM (*VAR* *B*WAITER))
                     (SCENE (ORDERING)))
             (PTRANS (ACTOR (*VAR* CUSTOMER))
                     (OBJECT (BOOK (TYPE (FOODLIST))))
                     (TO (*VAR* CUSTOMER))
                     (SCENE (ORDERING)))
             (MTRANS (ACTOR (*VAR* CUSTOMER))
                     (OBJECT (BOOK (TYPE (FOODLIST))))
                     (TO (CP (PART (*VAR* CUSTOMER))))
                     (SCENE (ORDERING)))
             (MBUILD (ACTOR (*VAR* CUSTOMER))
                     (OBJECT (FACT (RESULT (ATRANS (ACTOR (*VAR* *C*WAITER))
                                                   (OBJECT (*VAR* FOOD))
                                                   (TO (*VAR* CUSTOMER))
                                                   (FROM (*VAR* *C*WAITER)))))
)
                     (SCENE (ORDERING)))
             (MTRANS (ACTOR (*VAR* CUSTOMER))
                     (OBJECT (SIGNAL))
                     (TO (*VAR* *D*WAITER))
                     (FROM (*VAR* CUSTOMER))
                     (SCENE (ORDERING)))
             (PTRANS (ACTOR (*VAR* *E*WAITER))
                     (OBJECT (*VAR* *E*WAITER))
                     (TO (TABLE))
                     (SCENE (ORDERING)))
```

図5　レストラン、スクリプト（その1）

```
                (MTRANS (ACTOR (*VAR* CUSTOMER))
                        (OBJECT (FACT (RESULT (ATRANS (ACTOR (*VAR* *F*WAITER))
                                                      (OBJECT (*VAR* FOOD))
                                                      (TO (*VAR* CUSTOMER))
                                                      (FROM (*VAR* *F*WAITER)))))
)
                        (TO (*VAR* *F*WAITER))
                        (FROM (*VAR* CUSTOMER))
                        (SCENE (ORDERING))
                        (SENTENCE (MAINCONS)))
                (PTRANS (ACTOR (*VAR* *G*WAITER))
                        (OBJECT (FACT (RESULT (ATRANS (ACTOR (*VAR* CUSTOMER))
                                                      (OBJECT (*VAR* FOOD))
                                                      (TO (*VAR* CUSTOMER))
                                                      (FORM (*VAR* *G*WAITER)))))
)
                        (TO (CHEF))
                        (FROM (*VAR* *G*WAITER))
                        (SCENE (ORDERING)))
                (DO (ACTOR (CHEF))
                    (OBJECT (DO*COOKING))
                    (SCENE (ORDERING)))
                (ATRANS (ACTOR (CHEF))
                        (OBJECT (*VAR* FOOD))
                        (TO (*VAR* *H*WAITER))
                        (FROM (CHEF))
                        (SCENE (EATING)))
                (ATRANS (ACTOR (*VAR* *I*WAITER))
                        (OBJECT (*VAR* FOOD))
                        (TO (*VAR* CUSTOMER))
                        (FROM (*VAR* *I*WAITER))
                        (SCENE (EATING)))
                (INGEST (ACTOR (*VAR* CUSTOMER))
                        (OBJECT (*VAR* FOOD))
                        (SCENE (EATING))
                        (SENTENCE (MAINCONS)))
                (MTRANS (ACTOR (*VAR* CUSTOMER))
                        (OBJECT (FACT (RESULT (ATRANS (ACTOR (*VAR* *J*WAITER))
                                                      (OBJECT (CHECK))
                                                      (TO (*VAR* CUSTOMER))
                                                      (FROM (*VAR* *J*WAITER)))))
)
                        (TO (*VAR* *J*WAITER))
                        (FROM (*VAR* CUSTOMER))
                        (SCENE (EXITING)))
                (MOVE (ACTOR (*VAR* *K*WAITER))
                      (OBJECT (HAND (PART (*VAR* *K*WAITER))))
                      (TO (CHECK))
                      (SCENE (EXITING)))
                (PTRANS (ACTOR (*VAR* *L*WAITER))
                        (OBJECT (*VAR* *L*WAITER))
                        (TO (*VAR* CUSTOMER))
                        (SCENE (EXITING)))
                (ATRANS (ACTOR (*VAR* *M*WAITER))
                        (OBJECT (CHECK))
                        (TO (*VAR* CUSTOMER))
                        (FROM (*VAR* *M*WAITER))
                        (SCENE (EXITING)))
                (MOVE (ACTOR (*VAR* *N*WAITER))
                      (OBJECT (ACTION (RESULT (IS (ACTOR (*VAR* CUSTOMER))
                                                  (STATE (PLEASE (VAL (*VAR* PLEA
SEVAL)))))))
                      (SCENE (EXITING)))
                (MBUILD (ACTOR (*VAR* CUSTOMER))
                        (OBJECT (FACT (RESULT (ATRAN (ACTOR (*VAR* CUSTOMER))
                                                     (OBJECT (TIP (SIZE (*VAR* TI
```

図5 レストラン、スクリプト（その2）

```
PSIZE)))))))))
                (ATRANS (ACTOR (*VAR* CUSTOMER))
                        (OBJECT (TIP (SIZE (*VAR* TIPSIZE))))
                        (TO (*VAR* *O*WAITER))
                        (FROM (*VAR* CUSTOMER))
                        (SCENE (EXITING)))
                (PTRANS (ACTOR (*VAR* CUSTOMER))
                        (OBJECT (*VAR* CUSTOMER))
                        (TO (*VAR* MANAGER))
                        (FROM (TABLE))
                        (SCENE (EXITING)))
                (ATRANS (ACTOR (*VAR* CUSTOMER))
                        (OBJECT (MONEY))
                        (TO (*VAR* MANAGER))
                        (FROM (*VAR* CUSTOMER))
                        (SCENE (EXITING))
                        (SENTENCE (MAINCONS)))
                (PTRANS (ACTOR (*VAR* CUSTOMER))
                        (OBJECT (*VAR* CUSTOMER))
                        (TO (ANYWHERE))
                        (FROM (*VAR* RESTAURANT))
                        (SCENE (EXITING))
                        (SENTENCE (MAINCONS)))
                (IS (ACTOR (*VAR* CUSTOMER))
                    (STATE (HUNGER (VAL (0))))
                    (CONDITION (RESULT)))
                (POSS (ACTOR (*VAR* CUSTOMER))
                      (OBJECT (MONEY (SIZE (LESS))))
                      (CONDITION (RESULT)))))))
```

図5　レストラン、スクリプト（その3）

```
***DATA BASE CONTAINS***
((IS (ACTOR (PERSON (NAME (KINTARO))))
     (STATE (HUNGER (VAL (5))))
     (CONDITION (ENTRY)))
 (POSS (ACTOR (PERSON (NAME (KINTARO))))
       (OBJECT (MONEY))
       (CONDITION (ENTRY)))
 (MBUILD (ACTOR (PERSON (NAME (KINTARO))))
         (OBJECT (FACT (RESULT (PTRANS (ACTOR (PERSON (NAME (KINTARO))))
                                       (OBJECT (PERSON (NAME (KINTARO))))
                                       (TO (RESTAURANT))))))
         (SCENE (ENTERING)))
 (PTRANS (ACTOR (PERSON (NAME (KINTARO))))
         (OBJECT (PERSON (NAME (KINTARO))))
         (TO (PLACE (TYPE (RESTAURANT))))
         (SCENE (ENTERING))
         (SENTENCE (MAINCONS)))
 (MTRANS (ACTOR (PERSON (ROLE (HOSTESS))))
         (OBJECT (FACT (RESULT (MOVE (ACTOR (PERSON (NAME (KINTARO))))
                                     (TO (CHAIR))))))
         (TO (PERSON (NAME (KINTARO))))
         (FROM (PERSON (ROLE (HOSTESS)))))
 (ATTEND (ACTOR (PERSON (NAME (KINTARO))))
         (OBJECT (EYES))
         (TO (TABLE))
         (SCENE (ENTERING)))
 (MBUILD (ACTOR (PERSON (NAME (KINTARO))))
         (OBJECT (LOC (ENABLE (MOVE (ACTOR (PERSON (NAME (KINTARO))))
                                    (TO (CHAIR))))))
```

図6　スクリプトによって省略をおぎなわれ、階層構造化されたテキスト文（その1）

```
          (SCENE (ENTERING)))
(PTRANS (ACTOR (PERSON (NAME (KINTARO))))
          (OBJECT (PERSON (NAME (KINTARO))))
          (TO (TABLE))
          (SCENE (ENTERING)))
(MOVE (ACTOR (PERSON (NAME (KINTARO))))
      (TO (CHAIR))
      (SCENE (ENTERING)))
(PTRANS (ACTOR (PERSON (ROLE (HOSTESS))))
          (OBJECT (PERSON (ROLE (HOSTESS))))
          (TO (TABLE))
          (SCENE (ORDERING)))
(ATRANS (ACTOR (PERSON (ROLE (HOSTESS))))
          (OBJECT (BOOK (TYPE (FOODLIST))))
          (TO (PERSON (NAME (KINTARO))))
          (FROM (PERSON (ROLE (HOSTESS))))
          (SCENE (ORDERING)))
(PTRANS (ACTOR (PERSON (NAME (KINTARO))))
          (OBJECT (BOOK (TYPE (FOODLIST))))
          (TO (PERSON (NAME (KINTARO))))
          (SCENE (ORDERING)))
(MTRANS (ACTOR (PERSON (NAME (KINTARO))))
          (OBJECT (BOOK (TYPE (FOODLIST))))
          (TO (CP (PART (PERSON (NAME (KINTARO))))))
          (SCENE (ORDERING)))
(MBUILD (ACTOR (PERSON (NAME (KINTARO))))
          (OBJECT (FACT (RESULT (ATRANS (ACTOR (PERSON (ROLE (WAITER))))
                                        (OBJECT NIL)
                                        (TO (PERSON (NAME (KINTARO))))
                                        (FROM (PERSON (ROLE (WAITER))))))))
          (SCENE (ORDERING)))
(MTRANS (ACTOR (PERSON (NAME (KINTARO))))
          (OBJECT (SIGNAL))
          (TO (PERSON (ROLE (WAITER))))
          (FROM (PERSON (NAME (KINTARO))))
          (SCENE (ORDERING)))
(PTRANS (ACTOR (PERSON (ROLE (WAITER))))
          (OBJECT (PERSON (ROLE (WAITER))))
          (TO (TABLE))
          (SCENE (ORDERING)))
(MTRANS (ACTOR (PERSON (NAME (KINTARO))))
          (OBJECT (FACT (RESULT (ATRANS (ACTOR (PERSON (ROLE (WAITER))))
                                        (OBJECT (FOOD (TYPE (LOBSTER))))
                                        (TO (PERSON (NAME (KINTARO))))
                                        (FROM (PERSON (ROLE (WAITER))))))))
          (TO (PERSON (ROLE (WAITER))))
          (FROM (PERSON (NAME (KINTARO))))
          (SCENE (ORDERING))
          (SENTENCE (MAINCONS)))
(PTRANS (ACTOR (PERSON (ROLE (WAITER))))
          (OBJECT (FACT (RESULT (ATRANS (ACTOR (PERSON (NAME (KINTARO))))
                                        (OBJECT (FOOD (TYPE (LOBSTER))))
                                        (TO (PERSON (NAME (KINTARO))))
                                        (FORM (PERSON (ROLE (WAITER))))))))
          (TO (CHEF))
          (FROM (PERSON (ROLE (WAITER))))
          (SCENE (ORDERING)))
(DO (ACTOR (CHEF))
    (OBJECT (DO*COOKING))
    (SCENE (ORDERING)))
(ATRANS (ACTOR (CHEF))
          (OBJECT (FOOD (TYPE (LOBSTER))))
          (TO (PERSON (ROLE (WAITER))))
          (FROM (CHEF))
          (SCENE (EATING)))
(ATRANS (ACTOR (PERSON (ROLE (WAITER))))
```

図6　スクリプトによって省略をおぎなわれ、階層構造化されたテキスト文（その2）

```
        (OBJECT (FOOD) (TYPE (LOBSTER))))
        (TO (PERSON (NAME (KINTARO))))
        (FROM (PERSON (ROLE (WAITER))))
        (SCENE (EATING)))
(INGEST (ACTOR (PERSON (NAME (KINTARO))))
        (OBJECT (FOOD (TYPE (LOBSTER))))
        (SCENE (EATING))
        (SENTENCE (MAINCONS)))
(MTRANS (ACTOR (PERSON (NAME (KINTARO))))
        (OBJECT (FACT (RESULT (ATRANS (ACTOR (PERSON (ROLE (WAITER))))
                                      (OBJECT (CHECK))
                                      (TO (PERSON (NAME (KINTARO))))
                                      (FROM (PERSON (ROLE (WAITER))))))))
        (TO (PERSON (ROLE (WAITER))))
        (FROM (PERSON (NAME (KINTARO))))
        (SCENE (EXITING)))
(MOVE (ACTOR (PERSON (ROLE (WAITER))))
      (OBJECT (HAND (PART (PERSON (ROLE (WAITER))))))
      (TO (CHECK))
      (SCENE (EXITING)))
(PTRANS (ACTOR (PERSON (ROLE (WAITER))))
        (OBJECT (PERSON (ROLE (WAITER))))
        (TO (PERSON (NAME (KINTARO))))
        (SCENE (EXITING)))
(ATRANS (ACTOR (PERSON (ROLE (WAITER))))
        (OBJECT (CHECK))
        (TO (PERSON (NAME (KINTARO))))
        (FROM (PERSON (ROLE (WAITER))))
        (SCENE (EXITING)))
(MOVE (ACTOR (PERSON (ROLE (WAITER))))
      (OBJECT (ACTION (RESULT (IS (ACTOR (PERSON (NAME (KINTARO))))
                                  (STATE (PLEASE (VAL (10))))))))
      (SCENE (EXITING)))
(MBUILD (ACTOR (PERSON (NAME (KINTARO))))
        (OBJECT (FACT (RESULT (ATRAN (ACTOR (PERSON (NAME (KINTARO))))
                                     (OBJECT (TIP (SIZE (LARGE))))))))
(ATRANS (ACTOR (PERSON (NAME (KINTARO))))
        (OBJECT (TIP (SIZE (LARGE))))
        (TO (PERSON (ROLE (WAITER))))
        (FROM (PERSON (NAME (KINTARO))))
        (SCENE (EXITING)))
(PTRANS (ACTOR (PERSON (NAME (KINTARO))))
        (OBJECT (PERSON (NAME (KINTARO))))
        (TO NIL)
        (FROM (TABLE))
        (SCENE (EXITING)))
(ATRANS (ACTOR (PERSON (NAME (KINTARO))))
        (OBJECT (MONEY))
        (TO NIL)
        (FROM (PERSON (NAME (KINTARO))))
        (SCENE (EXITING))
        (SENTENCE (MAINCONS)))
(PTRANS (ACTOR (PERSON (NAME (KINTARO))))
        (OBJECT (PERSON (NAME (KINTARO))))
        (TO (ANYWHERE))
        (FROM (PLACE (TYPE (RESTAURANT))))
        (SCENE (EXITING))
        (SENTENCE (MAINCONS)))
(IS (ACTOR (PERSON (NAME (KINTARO))))
    (STATE (HUNGER (VAL (0))))
    (CONDITION (RESULT)))
(POSS (ACTOR (PERSON (NAME (KINTARO))))
      (OBJECT (MONEY (SIZE (LESS))))
      (CONDITION (RESULT)))
```

図6　スクリプトによって省略をおぎなわれ、階層構造化されたテキスト文（その3）

```
(*RESTAURANT (CUSTOMER (PERSON (NAME (KINTARO))))
             (RESTAURANT (PLACE (TYPE (RESTAURANT))))
             (*B*WAITER (PERSON (ROLE (HOSTESS))))
             (*E*WAITER (PERSON (ROLE (WAITER))))
             (FOOD (FOOD (TYPE (LOBSTER))))
             (PLEASEVAL (10))
             (TIPSIZE (LARGE))))
```

図6　スクリプトによって省略をおぎなわれ、階層構造化されたテキスト文（その4）

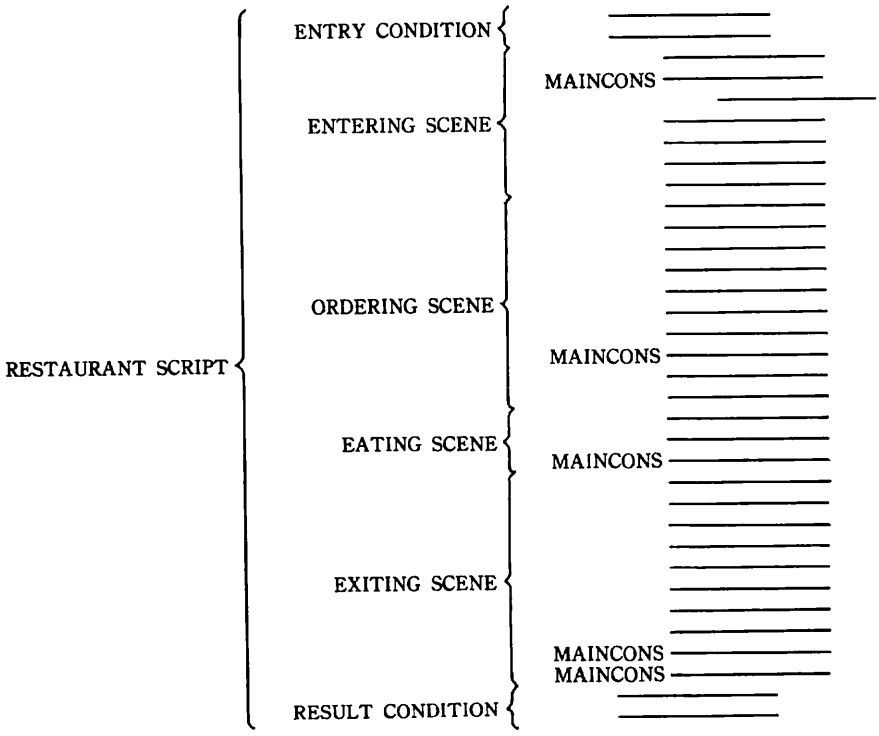

図7　図6の階層構造

```
(PROCESSQAS QASTORY LKINTAROSTORY)

>>>>>> QUESTION IS
(PTRANS (ACTOR (PERSON (NAME (KINTARO))))
            (OBJECT (PERSON (NAME (KINTARO))))
            (TO (PLACE (TYPE (RESTAURANT))))
            (FOCUS (PERSON (NAME (KINTARO))))
            (LEADTO (*## WHYVAR)))

    MATCHING CD IS
      (PTRANS (ACTOR (PERSON (NAME (KINTARO))))
            (OBJECT (PERSON (NAME (KINTARO))))
            (TO (PLACE (TYPE (RESTAURANT))))
            (SCENE (ENTERING))
            (SENTENCE (MAINCONS)))

    ANSWER IS
      (DO (ACTOR (PERSON (NAME (KINTARO))))
        (CONJUNCTION (BECAUSE))
        (ENABLE (IS (ACTOR (PERSON (NAME (KINTARO))))
                    (STATE (HUNGER (VAL (0))))
                    (CONDITION (RESULT)))))

>>>>>> QUESTION IS
(*PTRANS* (ACTOR (PERSON (NAME (KINTARO))))
               (OBJECT (PERSON (NAME (KINTARO))))
               (TO (TABLE))
               (FOCUS (PERSON (NAME (KINTARO)))))

    MATCHING CD IS
      (PTRANS (ACTOR (PERSON (NAME (KINTARO))))
            (OBJECT (PERSON (NAME (KINTARO))))
            (TO (TABLE))
            (SCENE (ENTERING)))

    ANSWER IS
      (PTRANS (ACTOR (PERSON (NAME (KINTARO))))
            (OBJECT (PERSON (NAME (KINTARO))))
            (TO (TABLE))
            (SCENE (ENTERING)))

>>>>>> QUESTION IS
(PTRANS (ACTOR (PERSON (NAME (KINTARO))))
            (OBJECT (PERSON (NAME (KINTARO))))
            (TO (TABLE))
            (FOCUS (PERSON (NAME (KINTARO))))
            (LEADTO (*## WHYVAR)))

    MATCHING CD IS
      (PTRANS (ACTOR (PERSON (NAME (KINTARO))))
            (OBJECT (PERSON (NAME (KINTARO))))
            (TO (TABLE))
            (SCENE (ENTERING)))

    ANSWER IS
      (DO (ACTOR (PERSON (NAME (KINTARO))))
        (CONJUNCTION (BECAUSE))
        (ENABLE (MOVE (ACTOR (PERSON (NAME (KINTARO))))
                      (TO (CHAIR))
                      (SCENE (ENTERING)))))
>>>>>> QUESTION IS
(*ATRANS* (ACTOR (PERSON (ROLE (WAITER))))
               (OBJECT (BOOK (TYPE (FOODLIST))))
               (TO (PERSON (NAME (KINTARO))))
               (FROM (PERSON (ROLE (WAITER))))
               (FOCUS (PERSON (ROLE (WAITER)))))
```

図8　Micro ＱＡの実行例（その１）

```
      MATCHING CD IS
        (ATRANS (ACTOR (PERSON (ROLE (HOSTESS))))
                (OBJECT (BOOK (TYPE (FOODLIST))))
                (TO (PERSON (NAME (KINTARO))))
                (FROM (PERSON (ROLE (HOSTESS))))
                (SCENE (ORDERING)))

      ANSWER IS
        (ATRANS (ACTOR (PERSON (ROLE (HOSTESS))))
                (OBJECT (BOOK (TYPE (FOODLIST))))
                (TO (PERSON (NAME (KINTARO))))
                (FROM (PERSON (ROLE (HOSTESS))))
                (SCENE (ORDERING)))

  >>>>>> QUESTION IS
  (ATRANS (ACTOR (*** WHOVAR))
                (OBJECT (BOOK (TYPE (FOODLIST))))
                (TO (PERSON (NAME (KINTARO))))
                (FROM (*** WHOVAR))
                (FOCUS (*** WHOVAR)))

      MATCHING CD IS
        (ATRANS (ACTOR (PERSON (ROLE (HOSTESS))))
                (OBJECT (BOOK (TYPE (FOODLIST))))
                (TO (PERSON (NAME (KINTARO))))
                (FROM (PERSON (ROLE (HOSTESS))))
                (SCENE (ORDERING)))

      ANSWER IS
        (ATRANS (ACTOR (PERSON (ROLE (HOSTESS))))
                (OBJECT (BOOK (TYPE (FOODLIST))))
                (TO (PERSON (NAME (KINTARO))))
                (FROM (PERSON (ROLE (HOSTESS))))
                (SCENE (ORDERING)))

  >>>>>> QUESTION IS
  (ATRANS (ACTOR (PERSON (ROLE (HOSTESS))))
                (OBJECT (BOOK (TYPE (FOODLIST))))
                (TO (PERSON (NAME (KINTARO))))
                (FROM (PERSON (ROLE (HOSTESS))))
                (FOCUS (PERSON (ROLE (HOSTESS))))
                (LEADTO (*** WHYVAR)))

      MATCHING CD IS
        (ATRANS (ACTOR (PERSON (ROLE (HOSTESS))))
                (OBJECT (BOOK (TYPE (FOODLIST))))
                (TO (PERSON (NAME (KINTARO))))
                (FROM (PERSON (ROLE (HOSTESS))))
                (SCENE (ORDERING)))

      ANSWER IS
        (DO (ACTOR (PERSON (NAME (KINTARO))))
          (CONJUNCTION (BECAUSE))
          (ENABLE (MTRANS (ACTOR (PERSON (NAME (KINTARO))))
                          (OBJECT (FACT (RESULT (ATRANS (ACTOR (PERSON (ROLE (WAI
TER))))
                                                        (OBJECT (FOOD (TYPE (LOBS
TER))))
                                                        (TO (PERSON (NAME (KINTAR
O))))
                                                        (FROM (PERSON (ROLE (WAIT
ER)))))))
                          (TO (PERSON (ROLE (WAITER))))
                          (FROM (PERSON (NAME (KINTARO))))
                          (SCENE (ORDERING))
                          (SENTENCE (MAINCONS)))))
```

図8 Micro QAの実行例(その2)

```
>>>>>> QUESTION IS
(ATRANS (ACTOR (PERSON (ROLE (HOSTESS))))
        (OBJECT (BOOK (TYPE (FOODLIST))))
        (TO (PERSON (NAME (KINTARO))))
        (FROM (PERSON (ROLE (HOSTESS))))
        (FOCUS (PERSON (ROLE (HOSTESS))))
        (LEADTO (*** WHATVAR)))

   MATCHING CD IS
     (ATRANS (ACTOR (PERSON (ROLE (HOSTESS))))
        (OBJECT (BOOK (TYPE (FOODLIST))))
        (TO (PERSON (NAME (KINTARO))))
        (FROM (PERSON (ROLE (HOSTESS))))
        (SCENE (ORDERING)))

   ANSWER IS
     ((PTRANS (ACTOR (PERSON (NAME (KINTARO))))
              (OBJECT (BOOK (TYPE (FOODLIST))))
              (TO (PERSON (NAME (KINTARO))))
              (SCENE (ORDERING)))
      (MTRANS (ACTOR (PERSON (NAME (KINTARO))))
              (OBJECT (BOOK (TYPE (FOODLIST))))
              (TO (CP (PART (PERSON (NAME (KINTARO))))))
              (SCENE (ORDERING)))
      (MBUILD (ACTOR (PERSON (NAME (KINTARO))))
              (OBJECT (FACT (RESULT (ATRANS (ACTOR (PERSON (ROLE (WAITER))))
                                            (OBJECT NIL)
                                            (TO (PERSON (NAME (KINTARO))))
                                            (FROM (PERSON (ROLE (WAITER))))))))
              (SCENE (ORDERING)))
      (MTRANS (ACTOR (PERSON (NAME (KINTARO))))
              (OBJECT (SIGNAL))
              (TO (PERSON (ROLE (WAITER))))
              (FROM (PERSON (NAME (KINTARO))))
              (SCENE (ORDERING)))
      (PTRANS (ACTOR (PERSON (ROLE (WAITER))))
              (OBJECT (PERSON (ROLE (WAITER))))
              (TO (TABLE))
              (SCENE (ORDERING)))
      (MTRANS (ACTOR (PERSON (NAME (KINTARO))))
              (OBJECT (FACT (RESULT (ATRANS (ACTOR (PERSON (ROLE (WAITER))))
                                            (OBJECT (FOOD (TYPE (LOBSTER))))
                                            (TO (PERSON (NAME (KINTARO))))
                                            (FROM (PERSON (ROLE (WAITER))))))))
              (TO (PERSON (ROLE (WAITER))))
              (FROM (PERSON (NAME (KINTARO))))
              (SCENE (ORDERING))
              (SENTENCE (MAINCONS))))

>>>>>> QUESTION IS
(*IS* (ACTOR (PERSON (NAME (KINTARO))))
      (STATE (ENJOY (VAL (5))))
      (FOCUS (PERSON (NAME (KINTARO))))
      (INSTRUMENT (INGEST (ACTOR (PERSON (NAME (KINTARO))))
                          (OBJECT (FOOD (TYPE (LOBSTER)))))))

   ANSWER IS
     (I DONT KNOW)

>>>>>> QUESTION IS
(*ATRANS* (ACTOR (PERSON (ROLE (WAITER))))
          (OBJECT (CHECK))
          (TO (PERSON (NAME (KINTARO))))
          (FROM (PERSON (ROLE (WAITER))))
          (FOCUS (PERSON (ROLE (WAITER)))))
```

図8　Micro ＱＡの実行例（その３）

```
    MATCHING CD IS
      (ATRANS (ACTOR (PERSON (ROLE (WAITER))))
              (OBJECT (CHECK))
              (TO (PERSON (NAME (KINTARO))))
              (FROM (PERSON (ROLE (WAITER))))
              (SCENE (EXITING)))

    ANSWER IS
      (ATRANS (ACTOR (PERSON (ROLE (WAITER))))
              (OBJECT (CHECK))
              (TO (PERSON (NAME (KINTARO))))
              (FROM (PERSON (ROLE (WAITER))))
              (SCENE (EXITING)))

>>>>>> QUESTION IS
(ATRANS (ACTOR (PERSON (NAME (KINTARO))))
              (OBJECT (TIP (SIZE (LARGE))))
              (TO (PERSON (ROLE (WAITER))))
              (FROM (PERSON (NAME (KINTARO))))
              (FOCUS (PERSON (NAME (KINTARO))))
              (LEADTO (*## WHYVAR)))

    MATCHING CD IS
      (ATRANS (ACTOR (PERSON (NAME (KINTARO))))
              (OBJECT (TIP (SIZE (LARGE))))
              (TO (PERSON (ROLE (WAITER))))
              (FROM (PERSON (NAME (KINTARO))))
              (SCENE (EXITING)))

    ANSWER IS
      (DO (ACTOR (PERSON (ROLE (WAITER))))
        (CONJUNCTION (BECAUSE PROBABLY))
        (ENABLE (MOVE (ACTOR (PERSON (ROLE (WAITER))))
                      (OBJECT (ACTION (RESULT (IS (ACTOR (PERSON (NAME (KINTARO
))))
                                                  (STATE (PLEASE (VAL (10))))))
))
                      (SCENE (EXITING)))))
>>>>>> QUESTION IS
(*ATRANS* (ACTOR (PERSON (NAME (KINTARO))))
              (OBJECT (MONEY))
              (FROM (PERSON (NAME (KINTARO))))
              (FOCUS (PERSON (NAME (KINTARO))))
              (INSTRUMENT (MTRANS (ACTOR (PERSON (NAME (KINTARO))))
                                  (OBJECT (BILL))
                                  (TO (CP (PART (PERSON (NAME (KINTARO))))))))
    MATCHING CD IS
      (ATRANS (ACTOR (PERSON (NAME (KINTARO))))
              (OBJECT (MONEY))
              (TO NIL)
              (FROM (PERSON (NAME (KINTARO))))
              (SCENE (EXITING))
              (SENTENCE (MAINCONS)))

    ANSWER IS
      (ATRANS (ACTOR (PERSON (NAME (KINTARO))))
              (OBJECT (MONEY))
              (TO NIL)
              (FROM (PERSON (NAME (KINTARO))))
              (SCENE (EXITING))
              (SENTENCE (MAINCONS)))

>>>>>> QUESTION IS
(ATRANS (ACTOR (PERSON (NAME (KINTARO))))
              (OBJECT (MONEY))
```

図8 Micro Q Aの実行例（その4）

```
            (FROM (PERSON (NAME (KINTARO))))
            (FOCUS (PERSON (NAME (KINTARO))))
            (LEADTO (*** WHYVAR)))

    MATCHING CD IS
      (ATRANS (ACTOR (PERSON (NAME (KINTARO))))
            (OBJECT (MONEY))
            (TO NIL)
            (FROM (PERSON (NAME (KINTARO))))
            (SCENE (EXITING))
            (SENTENCE (MAINCONS)))

    ANSWER IS
      (DO (ACTOR (PERSON (NAME (KINTARO))))
        (CONJUNCTION (BECAUSE))
        (ENABLE (PTRANS (ACTOR (PERSON (NAME (KINTARO))))
                        (OBJECT (PERSON (NAME (KINTARO))))
                        (TO (ANYWHERE))
                        (FROM (PLACE (TYPE (RESTAURANT))))
                        (SCENE (EXITING))
                        (SENTENCE (MAINCONS)))))


### QA DONE ###

!!! VALUE OF EVAL !!!
(### PROCESSING IS OVER)
```

図8　Micro ＱＡの実行例（その５）

```
 (PROCESSQAS
    (LAMBDA (STORY DATA)
       (PROG (QA)
          (CLEARQA)
          (CSETQ *DATA* DATA)
          (SHEADER)
          LOOP (COND
                  ((NULL STORY)
                    (MSG T T (QUOTE ###)
                         1 (QUOTE QA)
                         1 (QUOTE DONE)
                         1 (QUOTE ###)
                         T)
                    (RETURN (QUOTE (### PROCESSING IS OVER)))))
          (SETQ QA (CAR STORY))
          (SETQ STORY (CDR STORY))
          (MSG T (QUOTE >>>>>>)
               1 (QUOTE QUESTION)
               1 (QUOTE IS)
               T)
          (PP2 QA 4)
          (MSG T)
          (PROCESSQA QA)
          (GO LOOP)))
```

図9　Micro ＱＡのプログラム（その１）

```
(PROCESSQA
   (LAMBDA (QA)
      (COND
         ((ISLEADTOWHY QA)
           (DOLEADTOWHY1 QA))
         ((ISLEADTOWHAT QA)
           (DOLEADTOWHAT QA))
         ((NULL (DIDQA QA))
           (REDIDQA QA))))

(DIDQA
   (LAMBDA (QA)
      (PROG (QA1 QADATAS QADATA)
         (COND
            ((NULL (SETQ QA1 (ISDID QA)))
              (DOWHO QA)
              (RETURN T)))
         (SETQ QADATAS *DATA*)
         LOOP (COND
                 ((NULL QADATAS)
                   (RETURN)))
         (SETQ QADATA (CAR QADATAS))
         (SETQ QADATAS (CDR QADATAS))
         (COND
            ((MATCH# QA1 QADATA NIL)
              (PRINTMATCH# QADATA)
              (ANSWER QADATA)
              (RETURN T))
            (T
              (GO LOOP)))))
(ISDID
   (LAMBDA (QA)
      (PROG (X Y Z)
         (COND
            ((IS*#* QA)
              (RETURN)))
         (SETQ X (UNPACK (CAR QA)))
         (SETQ Z (LIST Z))
         (COND
            ((NOT (EQUAL (QUOTE *)
                         (CAR X)))
              (RETURN)))
         (SETQ Y (CDR X))
         LOOP (COND
                 ((NULL Y)
                   (RETURN))
                 ((EQUAL (CAR Y)
                         (QUOTE *))
                   (RETURN (CONS (MAKNAM (CDR Z))
                                 (CDR QA)))))
         (SETQ Z (CONSEND Z (CAR Y)))
         (SETQ Y (CDR Y))
         (GO LOOP)))

(IS*#*
   (LAMBDA (X)
      (AND (CONSP X)
           (EQUAL (CAR X)
                  (QUOTE *#*))))

(CLEARQA
   (LAMBDA NIL
      (CSETQ *DATABASE* NIL)
      (CSETQ *SHEADER* NIL)
      (CSETQ *MAINROLE* NIL)
```

図9 Micro QAのプログラム(その2)

```
      (CSETQ *CONDITION* NIL)
      (CSETQ *MAINCONS* NIL)
      (CSETQ *RESULT* NIL)
      (CSETQ *ENTRY* NIL))

(FILLERROLE
   (LAMBDA (ROLE CD)
      (PROG (PAIR)
         (SETQ PAIR (ASSOC ROLE (CDR CD)))
         (RETURN (COND
                     ((AND PAIR (CADR PAIR))
                       (CADR PAIR))
                     (T
                       NIL)))))

(CONSEND
   (LAMBDA (L X)
      (APPEND L (LIST X)))

(CARIF
   (LAMBDA (X)
      (COND
         (X
           (CAR X))
         (T
           NIL)))

(ASSOC
   (LAMBDA (A B)
      (SASSOC A B NIL))
(ANSWERWHY2
   (LAMBDA (X)
      (PROG (ACTORVAR ANS)
         (SETQ ACTORVAR (FILLERROLE (QUOTE ACTOR)
                                    X))
         (SETQ ANS (CONSEND (CONSEND (CONSEND (QUOTE (DO))
                                              (CONSEND (QUOTE (ACTOR))
                                                       ACTORVAR))
                                    (QUOTE (CONJUNCTION (BECAUSE PROBABLY))))
                           (CONSEND (QUOTE (ENABLE))
                                    X)))
         (ANSWER ANS)))

(ANSWERWHY
   (LAMBDA (X)
      (PROG (ACTORVAR ANS)
         (SETQ ACTORVAR (FILLERROLE (QUOTE ACTOR)
                                    X))
         (SETQ ANS (CONSEND (CONSEND (CONSEND (QUOTE (DO))
                                              (CONSEND (QUOTE (ACTOR))
                                                       ACTORVAR))
                                    (QUOTE (CONJUNCTION (BECAUSE))))
                           (CONSEND (QUOTE (ENABLE))
                                    X)))
         (ANSWER ANS)))

(NEXTMAINCONS
   (LAMBDA (QADATA)
      (PROG (MAINS MAIN1 MAIN2)
         (SETQ MAINS *MAINCONS*)
         LOOP (COND
                 ((NULL MAINS)
                   (RETURN)))
         (SETQ MAIN1 (CAR MAINS))
         (SETQ MAIN2 (CADR MAINS))
```

図9　Micro QAのプログラム（その3）

```
          (COND
             ((EQUAL MAIN1 QADATA)
               (RETURN MAIN2))
             (T
               (SETQ MAINS (CDR MAINS))
               (GO LOOP)))))

(ISLEADTOWHAT
   (LAMBDA (QA)
      (COND
         ((EQUAL (CADR (FILLERROLE (QUOTE LEADTO)
                                   QA))
                 (QUOTE WHATVAR))
           T)
         (T
           NIL)))

(DOLEADTOWHAT
   (LAMBDA (QA)
      (PROG (QADATAS QADATA)
         (SETQ QADATAS *DATA*)
         LOOP (COND
                 ((NULL QADATAS)
                   (RETURN)))
         (SETQ QADATA (CAR QADATAS))
         (SETQ QADATAS (CDR QADATAS))
         (COND
            ((MATCH# QA QADATA NIL)
              (PRINTMATCH# QADATA)
              (RETURN (DOLEADTOWHAT2 QADATA QADATAS)))
            (T
              (GO LOOP)))))

(DOLEADTOWHAT2
   (LAMBDA (QADATA QADATAS)
      (PROG (QADATA1 SCENE1 SCENE2)
         (COND
            ((NULL QADATA)
              (RETURN (IDONT))))
         (SETQ SCENE1 (FILLERROLE (QUOTE SCENE)
                                  QADATA))
         (SETQ QADATA1 (CAR QADATAS))
         (COND
            ((NULL SCENE1)
              (SETQ SCENE2 (SCENES (CAR (FILLERROLE (QUOTE SCENE)
                                                    QADATA1))
                                   QADATAS))
              (EQUAL SCENE1 (FILLERROLE (QUOTE SCENE)
                                        QADATA1))
              (SETQ SCENE2 (SCENES (CAR SCENE1)
                                   QADATAS)))
            (T
              (SETQ SCENE2 (SCENES (CAR (FILLERROLE (QUOTE SCENE)
                                                    QADATA1))
                                   QADATAS))))
         (COND
            ((NULL SCENE2)
              (RETURN (ANSWER (CAR *RESULT*))))
            (T
              (ANSWER SCENE2)))))

(DOWHO
   (LAMBDA (QA)
      (PROG (QADATAS QADATA)
         (SETQ QADATAS *DATA*)
```

図9 Micro QAのプログラム(その4)

```
        LOOP (COND
                ((NULL QADATAS)
                  (RETURN (IDONT))))
        (SETQ QADATA (CAR QADATAS))
        (SETQ QADATAS (CDR QADATAS))
        (COND
           ((MATCH# QA QADATA NIL)
             (PRINTMATCH# QADATA)
             (RETURN (ANSWER QADATA)))
           (T
             (GO LOOP)))))

(RMATCH
   (LAMBDA (QA)
      (PROG (QADATAS QADATA1 QADATA2)
         (SETQ QADATAS *DATA*)
         LOOP (COND
                 ((NULL QADATAS)
                   (RETURN)))
         (SETQ QADATA1 (CAR QADATAS))
         (SETQ QADATA2 (CADR QADATAS))
         (COND
            ((MATCH# QA QADATA2 NIL)
              (RETURN QADATA1))
            (T
              (SETQ QADATAS (CDR QADATAS))
              (GO LOOP)))))
(SCENES
   (LAMBDA (SCENE1 QADATAS)
      (PROG (SCENE2 QADATA)
         LOOP (COND
                 ((NULL QADATAS)
                   (RETURN SCENE2)))
         (SETQ QADATA (CAR QADATAS))
         (SETQ QADATAS (CDR QADATAS))
         (COND
            ((AND (EQUAL SCENE1 (CAR (FILLERROLE (QUOTE SCENE)
                                                 QADATA)))
                  (EQUAL (QUOTE (MAINCONS))
                         (FILLERROLE (QUOTE SENTENCE)
                                     QADATA)))
              (RETURN (CONSEND SCENE2 QADATA)))
            ((EQUAL SCENE1 (CAR (FILLERROLE (QUOTE SCENE)
                                            QADATA)))
              (SETQ SCENE2 (CONSEND SCENE2 QADATA))
              (GO LOOP))
            (T
              (GO LOOP)))))

(ISLEADTOWHY
   (LAMBDA (QA)
      (COND
         ((EQUAL (CADR (FILLERROLE (QUOTE LEADTO)
                                   QA))
                 (QUOTE WHYVAR))
           T)
         (T
           NIL)))

(ANSWER
   (LAMBDA (X)
      (MSG T 4 (QUOTE ANSWER)
           1 (QUOTE IS)
           T 6)
      (PP2 X 4)
      (MSG T))
```

図9　Micro ＱＡのプログラム（その５）

```
(PRINTMATCH#
   (LAMBDA (X)
      (MSG T 4 (QUOTE MATCHING)
           1 (QUOTE CD)
           1 (QUOTE IS)
           T 6)
      (PP2 X 4)
      (MSG T))

(DOLEADTOWHY2
   (LAMBDA (QADATA QADATAS)
      (PROG (SCENE1 QADATAS1 QADATA1 RMAT)
         (COND
            ((NULL QADATA)
              (RETURN (IDONT)))
            ((EQUAL (FILLERROLE (QUOTE SENTENCE)
                                QADATA)
                    (QUOTE (MAINCONS)))
              (COND
                 ((EQUAL (CAR *MAINCONS*)
                         QADATA)
                   (RETURN (ANSWERWHY (CAR *RESULT*))))
                 (T
                   (RETURN (ANSWERWHY (NEXTMAINCONS QADATA))))))
            ((FILLERROLE (QUOTE SIZE)
                         (FILLERROLE (QUOTE OBJECT)
                                     QADATA))
              (COND
                 ((NULL (FILLERROLE (QUOTE RESULT)
                                    (FILLERROLE (QUOTE OBJECT)
                                                (SETQ RMAT
                                                      (RMATCH QADATA)))))
                   (RETURN (ANSWERWHY2 RMAT)))
                 (T
                   (RETURN (ANSWERWHY2 (RMATCH RMAT)))))))
         (SETQ QADATAS1 QADATAS)
         (SETQ SCENE1 (FILLERROLE (QUOTE SCENE)
                                  QADATA))
         LOOP (COND
                 ((NULL QADATAS1)
                   (RETURN (ANSWERWHY (CAR QADATAS)))))
         (SETQ QADATA1 (CAR QADATAS1))
         (COND
            ((NULL SCENE1)
              (RETURN (ANSWERWHY (RMATCH QADATA))))
            ((AND (EQUAL (FILLERROLE (QUOTE SENTENCE)
                                     QADATA1)
                         (QUOTE (MAINCONS)))
                  (EQUAL (FILLERROLE (QUOTE SCENE)
                                     QADATA1)
                         SCENE1))
              (RETURN (ANSWERWHY QADATA1)))
            (T
              (SETQ QADATAS1 (CDR QADATAS1))
              (GO LOOP)))))

(DOLEADTOWHY1
   (LAMBDA (QA)
      (PROG (QADATAS QADATA)
         (SETQ QADATAS *DATA*)
         LOOP (COND
                 ((NULL QADATAS)
                   (RETURN)))
         (SETQ QADATA (CAR QADATAS))
         (SETQ QADATAS (CDR QADATAS))
```

図9 Micro QAのプログラム(その6)

```
            (COND
               ((MATCH# QA QADATA NIL)
                 (PRINTMATCH# QADATA)
                 (RETURN (DOLEADTOWHY? QADATA QADATAS)))
               (T
                 (GO LOOP)))))

(MATCHARGS#
   (LAMBDA (PATARGS CONST BINDINGFORM)
      (PROG (CONSTVAL PATARG)
         (SETQ PATARG NIL)
         (SETQ CONSTVAL NIL)
         LOOP (SETQ PATARG (CAR|F PATARGS))
         (SETQ PATARGS (CDR PATARGS))
         (COND
            ((NULL PATARG)
              (RETURN BINDINGFORM))
            (T
              (SETQ CONSTVAL (FILLERROLE (CAR PATARG)
                                         CONST))))
         (COND
            ((SETQ BINDINGFORM (MATCH# (CADR PATARG)
                                       CONSTVAL BINDINGFORM))
              (GO LOOP))
            (T
              (RETURN))))))

(MATCHVAR#
   (LAMBDA (PAT CONST BINDINGFORM)
      (PROG (VARVALUE)
         (SETQ VARVALUE (FILLERROLE (NAMEVAR PAT)
                                    BINDINGFORM))
         (RETURN (COND
                    (VARVALUE
                      (MATCH# VARVALUE CONST BINDINGFORM))
                    (T
                      (CONSEND BINDINGFORM (LIST (NAMEVAR PAT)
                                                 CONST)))))))

(MATCH#
   (LAMBDA (PAT CONST BINDINGS)
      (PROG (BINDINGFORM)
         (SETQ BINDINGFORM (COND
                              ((NULL BINDINGS)
                                (LIST T))
                              (T
                                BINDINGS)))
         (RETURN (COND
                    ((OR (NULL CONST)
                         (EQUAL PAT CONST))
                      BINDINGFORM)
                    ((IS*#* PAT)
                      (MATCHVAR# PAT CONST BINDINGFORM))
                    ((OR (ATOM CONST)
                         (ATOM PAT))
                      NIL)
                    ((EQUAL (CAR PAT)
                            (CAR CONST))
                      (MATCHARGS# (CDR PAT)
                                  CONST BINDINGFORM)))))))

(TAKEROLE
   (LAMBDA (R1 QA)
      (PROG (TOP QA1)
         (SETQ TOP (LIST (CAR QA)))
         (SETQ QA1 QA)
```

図9　Micro ＱＡのプログラム（その７）

```
          LOOP (COND
                  ((NULL (SETQ QA1 (CDR QA1)))
                    (RETURN TOP))
                  ((MEMBER (CAAR QA1)
                           R1)
                    (SETQ TOP (CONSEND TOP (CONSEND TOP
                                                    (CONSEND (LIST (CAAR QA1))
                                                             (LIST (QUOTE ***)
                                                                   (QUOTE XXX)))
)))
                    (SETQ QA1 (CDR QA1))
                    (GO LOOP))
                  (T
                    (SETQ TOP (CONSEND TOP (CAR QA1)))
                    (SETQ QA1 (CDR QA1))
                    (GO LOOP)))))

(IDONT
   (LAMBDA NIL
      (MSG T 4 (QUOTE ANSWER)
           1 (QUOTE IS)
           T 6 (QUOTE (I DONT KNOW))
           T))

(REDIDQA
   (LAMBDA (QA)
      (PROG (FOC R1 QA1)
         (MAINROLE)
         (COND
            ((NULL (SETQ FOC (FILLERROLE (QUOTE FOCUS)
                                         QA)))
              (RETURN (IDONT)))
            ((MEMBER FOC *MAINROLE*)
              (RETURN (IDONT))))
         (SETQ R1 (SROLE FOC QA))
         (SETQ QA1 (TAKEROLE R1 QA))
         (COND
            ((NULL (DIDQA QA1))
              (RETURN (IDONT)))
            (T
              (RETURN)))))

(SROLE
   (LAMBDA (FILLER DATA)
      (COND
         ((NULL DATA)
           NIL)
         ((EQUAL FILLER (CAR (CDADR DATA)))
           (CONS (CAADR DATA)
                 (SROLE FILLER (CDR DATA))))
         (T
           (SROLE FILLER (CDR DATA)))))

(MAINROLE
   (LAMBDA NIL
      (PROG (SHEADERS SHEADER)
         (SETQ SHEADERS *SHEADER*)
         LOOP (COND
                 ((NULL SHEADERS)
                   (RETURN)))
         (SETQ SHEADER (CAR SHEADERS))
         (CSETQ *MAINROLE* (CONSEND *MAINROLE* (CADR (CADR SHEADER))))
         (SETQ SHEADERS (CDR SHEADERS))
         (GO LOOP)))
```

図9 Micro QAのプログラム(その8)

```
(SHEADER
   (LAMBDA NIL
      (PROG (QADATAS QADATA)
         (SETQ QADATAS *DATA*)
         LOOP (COND
                 ((NULL QADATAS)
                   (RETURN)))
         (SETQ QADATA (CAR QADATAS))
         (SETQ QADATAS (CDR QADATAS))
         (COND
            ((EQUAL (QUOTE *)
                    (CAR (UNPACK (CAR QADATA))))
              (CSETQ *SHEADER* (CONSEND *SHEADER* QADATA))
              (GO LOOP))
            ((EQUAL (FILLERROLE (QUOTE CONDITION)
                                QADATA)
                    (QUOTE (ENTRY)))
              (CSETQ *ENTRY* (CONSEND *ENTRY* QADATA))
              (GO LOOP))
            ((EQUAL (FILLERROLE (QUOTE SENTENCE)
                                QADATA)
                    (QUOTE (MAINCONS)))
              (CSETQ *MAINCONS* (CONSEND *MAINCONS* QADATA))
              (GO LOOP))
            ((EQUAL (FILLERROLE (QUOTE CONDITION)
                                QADATA)
                    (QUOTE (RESULT)))
              (CSETQ *RESULT* (CONSEND *RESULT* QADATA))
              (GO LOOP))
            (T
              (GO LOOP)))))

(FILLERROLE*N*
   (LAMBDA (ROLE CD)
      (PROG (VARROLE1 NCDS NCD VARROLE2 VARROLES)
         (COND
            ((NULL (SETQ VARROLE1 (IS*N* ROLE)))
              (RETURN)))
         (SETQ NCDS (CDR CD))
         LOOP (COND
                 ((NULL NCDS)
                   (GO LOOP2)))
         (SETQ NCD (CAR NCDS))
         (COND
            ((NULL (SETQ VARROLE2 (IS*N* (CAR NCD))))
              (SETQ NCDS (CDR NCDS))
              (GO LOOP))
            (T
              (SETQ VARROLES (CONS VARROLE2 VARROLES))
              (SETQ NCDS (CDR NCDS))
              (GO LOOP)))
         LOOP2 (SETQ VARROLE2 (CAR VARROLES))
         (COND
            ((NULL VARROLES)
              (RETURN))
            ((EQUAL (CDR VARROLE1)
                    (CDR VARROLE2))
              (SETQ VARROLE2 (APPEND (UNPACK (CAR VARROLE2))
                                     (UNPACK (CADR VARROLE2))))
              (RETURN (FILLERROLE (MAKNAM VARROLE2)
                                  BINDINGS)))
            (T
              (SETQ VARROLE2 (CDR VARROLES))
              (GO LOOP2)))))
```

図9　Micro ＱＡのプログラム（その９）

```
(ISVAR
   (LAMBDA (X)
      (AND (CONSP X)
           (EQUAL (CAR X)
                  (QUOTE *VAR*))))

(PP2
   (LAMBDA (L POS)
      (COND
         ((ATOM L)
           (PRIN1 L))
         ((ATOM (CAR L))
           (PRIN1 LPAR)
           (PRIN1 (CAR L))
           (PPCDR (CDR L)
                  (PLUS (ADD1 POS)
                        (LENGTH (UNPACK (CAR L)))))
           (PRIN1 RPAR))
         (T
           (PRIN1 LPAR)
           (PP2 (CAR L)
                (ADD1 POS))
           (COND
              ((CDR L)
                (TERPRI)
                (SPACES POS)
                (PPCDR (CDR L)
                       POS))
              (T
                NIL))
           (PRIN1 RPAR))))

(PPCDR
   (LAMBDA (L POS)
      (COND
         ((NULL L)
           NIL)
         ((NULL (CDR L))
           (SPACES 1)
           (PP2 (CAR L)
                (ADD1 POS)))
         (T
           (SPACES 1)
           (PP2 (CAR L)
                (ADD1 POS))
           (TERPRI)
           (SPACES POS)
           (PPCDR (CDR L)
                  POS))))

(SPACES
   (LAMBDA (N)
      (WHILE (GREATERP N 0)
             (PRIN1 BLANK)
             (SETQ N (SUB1 N))))

(NAMEVAR
   (LAMBDA (X)
      (COND
         ((AND (CONSP X)
               (CONSP (CDR X)))
           (CADR X))
         (T
           NIL)))
```

図9 Micro QAのプログラム（その10）

```
(SCHANKFNS
   (LAMBDA NIL
      (QUOTE (FILLERROLE*N* ISVAR PP2 PPCDR SPACES NAMEVAR
                            SCHANKFNS SOME INSTANTIATE
                            INSTANTIATE2 MSG CONSEND CONSP
(SOME                       EVALA MAKNAM)))
   (LAMBDA (L FNS)
      (COND
         ((NULL L)
           NIL)
         ((FNS (CAR L))
           T)
         (T
           (SOME (CDR L)
                 FNS))))
(INSTANTIATE
   (LAMBDA (CDFORM BINDINGS)
      (COND
         ((ATOM CDFORM)
           CDFORM)
         ((ISVAR CDFORM)
           (PROG (VAR)
              (SETQ VAR (FILLERROLE (NAMEVAR CDFORM)
                                    BINDINGS))
              (COND
                 ((NULL VAR)
                   (RETURN (INSTANTIATE (FILLERROLE*N* (NAMEVAR CDFORM)
                                                       BINDINGS)
                                        BINDINGS)))
                 (T
                   (RETURN (INSTANTIATE VAR BINDINGS))))))
         (T
           (CONS (CAR CDFORM)
                 (INSTANTIATE2 (CDR CDFORM))))))

(INSTANTIATE2
   (LAMBDA (PAIRS)
      (COND
         ((NULL PAIRS)
           NIL)
         (T
           (CONS (LIST (CAAR PAIRS)
                       (INSTANTIATE (CADR (CAR PAIRS))
                                    BINDINGS))
                 (INSTANTIATE2 (CDR PAIRS))))))
(CONSEND
   (LAMBDA (L X)
      (APPEND L (LIST X)))

(CONSP
   (LAMBDA (X)
      (NOT (ATOM X)))

(EVALA
   NIL

(MAKNAM
   (LAMBDA (X)
      (PROG NIL
         (CLEARBUFF)
         L (COND
              ((NULL X)
                (RETURN (MKNAME))))
         (PACK (CAR X))
         (SETQ X (CDR X))
         (GO L)))
```

図9　Micro ＱＡのプログラム（その11）

```
((PUT (QUOTE EVALA)
      (QUOTE FEXPR)
      (QUOTE (LAMBDA (SEXP ALIST)
                 (EVAL (EVAL (CAR SEXP)
                             ALIST)
                       ALIST))))
 (PUT (QUOTE MSG)
      (QUOTE FEXPR)
      (QUOTE (LAMBDA (L A)
                 (PROG (L1)
                    (WHILE L (SETQ L1 (CAR L))
                           (COND
                              ((EQ (QUOTE T)
                                   L1)
                               (TERPRI))
                              ((NUMBERP L1)
                               (SPACES L1))
                              (T
                               (PRIN1 (EVAL L1 A))))
                           (SETQ L (CDR L)))
                    (RETURN))))))
```

図9 Micro QAのプログラム(その12)

```
(PROCESSTEXT KINTAROQA)

>>>>>> INPUT IS (WHY DID KINTARO GO TO THE RESTAURANT)

PROCESSING *START*

PROCESSING WHY

    *PARTOFSPEECH* = INTERROGATIVE

    *ROGATIVE* = (LEADTO (*## WHYVAR))

PROCESSING DID

    *PARTOFSPEECH* = AUXVERB

PROCESSING KINTARO

    *CDFORM* = (PERSON (NAME (KINTARO)))

    *PARTOFSPEECH* = NOUNPHRASE

    *SUBJECT* = (PERSON (NAME (KINTARO)))

PROCESSING GO

    *PARTOFSPEECH* = VERB

    *CDFORM* = (PTRANS (ACTOR (*VAR* GOVAR1))
                       (OBJECT (*VAR* GOVAR1))
                       (TO (*VAR* GOVAR2))
                       (FROM (*VAR* GOVAR3))
                       (FOCUS (*VAR* GOVAR1)))
```

図10 Micro ELIの実行例の一部(その1)

```
    *QCDFORM* = (*PTRANS* (ACTOR (*VAR* GOVAR1))
                          (OBJECT (*VAR* GOVAR1))
                          (TO (*VAR* GOVAR2))
                          (FROM (*VAR* GOVAR3))
                          (FOCUS (*VAR* GOVAR1)))

    GOVAR1 = (PERSON (NAME (KINTARO)))

    *CONCEPT* = (PTRANS (ACTOR (*VAR* GOVAR1))
                        (OBJECT (*VAR* GOVAR1))
                        (TO (*VAR* GOVAR2))
                        (FROM (*VAR* GOVAR3))
                        (FOCUS (*VAR* GOVAR1))
                        (LEADTO (*## WHYVAR)))


PROCESSING TO
### NOT IN THE DICTIONARY ###

PROCESSING THE


PROCESSING RESTAURANT

    *PARTOFSPEECH* = NOUN

    *CDFORM* = (PLACE (TYPE (RESTAURANT)))

    *PARTOFSPEECH* = NOUNPHRASE

    *CDFORM* = (PLACE (TYPE (RESTAURANT)))

    GOVAR2 = (PLACE (TYPE (RESTAURANT)))

CD FORM IS
  (PTRANS (ACTOR (PERSON (NAME (KINTARO))))
          (OBJECT (PERSON (NAME (KINTARO))))
          (TO (PLACE (TYPE (RESTAURANT))))
          (FOCUS (PERSON (NAME (KINTARO))))
          (LEADTO (*## WHYVAR)))


>>>>>> INPUT IS (DID KINTARO GO TO A TABLE)


PROCESSING *START*


PROCESSING DID

    *PARTOFSPEECH* = AUXVERB


PROCESSING KINTARO

    *CDFORM* = (PERSON (NAME (KINTARO)))

    *PARTOFSPEECH* = NOUNPHRASE

    *SUBJECT* = (PERSON (NAME (KINTARO)))


PROCESSING GO

    *PARTOFSPEECH* = VERB
```

図10　Micro ELIの実行例の一部（その２）

```
    *CDFORM* = (PTRANS (ACTOR (*VAR* GOVAR1))
                       (OBJECT (*VAR* GOVAR1))
                       (TO (*VAR* GOVAR2))
                       (FROM (*VAR* GOVAR3))
                       (FOCUS (*VAR* GOVAR1)))

    *OCDFORM* = (*PTRANS* (ACTOR (*VAR* GOVAR1))
                          (OBJECT (*VAR* GOVAR1))
                          (TO (*VAR* GOVAR2))
                          (FROM (*VAR* GOVAR3))
                          (FOCUS (*VAR* GOVAR1)))

    GOVAR1 = (PERSON (NAME (KINTARO)))

    *CONCEPT* = (*PTRANS* (ACTOR (*VAR* GOVAR1))
                          (OBJECT (*VAR* GOVAR1))
                          (TO (*VAR* GOVAR2))
                          (FROM (*VAR* GOVAR3))
                          (FOCUS (*VAR* GOVAR1)))

PROCESSING TO
### NOT IN THE DICTIONARY ###

PROCESSING A


PROCESSING TABLE

    *PARTOFSPEECH* = NOUN

    *CDFORM* = (TABLE)

    *PARTOFSPEECH* = NOUNPHRASE

    *CDFORM* = (TABLE)

    GOVAR2 = (TABLE)

CD FORM IS
  (*PTRANS* (ACTOR (PERSON (NAME (KINTARO))))
            (OBJECT (PERSON (NAME (KINTARO))))
            (TO (TABLE))
            (FOCUS (PERSON (NAME (KINTARO)))))
```

図10 Micro ELIの実行例の一部(その3)

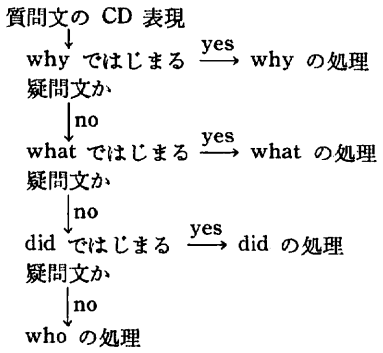

図. 11 Micro QA の質問文の CD 表現の識別