

自然語要求仕様からオブジェクト指向設計図を自動生成するシステム CAMEO

原 田 実^{†1} 野 村 佳 秀^{†2} 山 本 幸 二^{†2}
大 野 雅 志^{†3} 田 村 浩 樹^{†4} 高 橋 史 郎^{†5}

オブジェクト指向分析を支援するために、自然言語による要求文からオブジェクト図やイベントトレース図を自動生成するシステム CAMEO を開発した。CAMEO は、まず個々の要求文に字句解析・統語解析を行い、それを構成する語彙に対する一群の意味(格)フレームに分割する。この語彙に対する意味フレームには、品詞や EDR 電子化辞書による語彙の概念 ID や他の語彙との関係子が含まれている。次にオブジェクト指向分析部が、個々の意味フレームごとに分析ルールを適用し、クラス・属性・属性値・ロール・イベント・集約・継承・関連・計算・シナリオの 10 種のオブジェクトモデリング要素を RSL というリスト形式で抽出する。分析ルールの条件部は、語彙が表す概念の ID は何か、他の語彙とどんな関係子で関係しているか、語彙が表す概念の上位概念は何であるか、語彙が特定の表記を持つかなどを表す述語から構成される。最後に、抽出されたモデリング要素を遺伝アルゴリズムを用いて自動レイアウトし、オブジェクト図やイベントトレース図を生成する。

An Automatic Generation of Object Modeling Diagrams from Japanese Requirement Statements

MINORU HARADA,^{†1} YOSHIHIDE NOMURA,^{†2} KOUJI YAMAMOTO,^{†2}
MASASHI OONO,^{†3} HIROKI TAMURA^{†4} and SHIRO TAKAHASHI^{†5}

The purpose of our study is to develop a system CAMEO (Computer Automated Modeling Engine for Objects), which automates object oriented analysis and design activity. CAMEO inputs requirement statements written in Japanese. First, CAMEO parses and analyses each input sentence into the semantic frames of verb, noun, and adjective. Next, CAMEO extracts and classifies modeling elements from those semantic frames. These modeling elements are classes, attributes, attribute values, roles, associations, aggregations, inheritances, events, computations and scenarios. The condition parts of analysis rules consist of the predicates specifying the identifier of the concept represented by the frame, the type of the relationship with other frames, the upper-concept of the concept represented by the frame, and what specific word the frame contains. Finally, the extracted modeling elements are automatically layouted into object diagrams and event-trace diagrams.

1. 序 論

オブジェクト指向開発を行うには、通常対象世界を

記述した図 1 のような日本語要求文を受け取り、シユレーマラー法⁶⁾や OMT 法⁵⁾などの分析技法に従って、クラス、関連、属性、ロールなどからなるオブジェクト図や、クラス、イベント、動作などからなるイベントトレース図を、人手で作成している。しかし、分析の際に抽出するモデリング要素はかなりの数に上り、また分析ルールも経験を積まないとなかなか適用するには難しいものである。したがって、日本語要求文を受け付け、記述されている内容を意味的に理解し、分析ルールに従ってモデリング要素を自動抽出し、これを自動レイアウトして、たたき台となる設計図を自動生成することができれば、オブジェクト指向開発の生産性を向上できると考えられる。

†1 青山学院大学理工学部経営工学科

Department of Industrial Engineering, Faculty of Science and Engineering, Aoyama Gakuin University

†2 青山学院大学大学院理工学研究科経営工学専攻

Department of Industrial Engineering, Graduate School of Science and Engineering, Aoyama Gakuin University

†3 北陸電力システムサービス株式会社

Hokuden Information System Co., Inc.

†4 株式会社ソフトウェアリサーチアソシエイツ

Software Research Associates, Inc.

†5 東日本旅客鉄道株式会社

East Japan Railway Company

- (1) ビルは 10 階建てで、エレベータは 4 基備えられている。
- (2) 人は各フロアでランダムに発生する。
- (3) 人は行き先階を持つ。
- (4) 乗客の発生は乱数を用いる。
- (5) 行き先階が乗る階の上下に従って、上下行きボタンを押す。
- (6) 各エレベータ内には、行き先ボタンが 1 階から 10 階まである。
- (7) 人はエレベータに乗ると行き先階のボタンを押す。
- (8) ボタンは行き先階の指定を受けると点灯する。
- (9) エレベータは搭乗者の指定した階に止まり、外で待つ人の要求している階に止まる。
- (10) エレベータは止まるとドアが開き、先ず降りる人が降りる。
- (11) 次に乗る人が乗り、最後にドアが閉まる。
- (12) エレベータは次の要求階あるいは指定階へ移動する。
- (13) 1 階、10 階を除き、各フロアには 2 つの上下行きボタンがある。
- (14) 上下行きボタンは押すと点灯する。
- (15) 要求した人をエレベータが乗せるとその方向の上下行きボタンは消灯する。
- (16) 要求がない場合には、エレベータはドアを閉めて、最終到着フロアで停止し、次の要求を待つ。
- (17) 全てのフロアには等しく優先権が与えられていて、エレベータは最終的には全ての要求に応じる。
- (18) エレベータはエレベータ内からの行き先指定に対して、運転方向に従って、順序良く応じる。
- (19) エレベータには、10 名まで乗ることができる。
- (20) その他の詳細は実際のエレベータの動きを想定して考えよ。

図 1 エレベータ問題の要求文

Fig. 1 Requirement statements of the sample elevator problem.

そもそも、自然言語で記述された文章からモデリング要素やプログラム要素を抽出する自動化システムの研究は、Abbott¹⁾による英語で書かれたアルゴリズムの記述から変数名、データ型、演算子、制御構造などを抽出する研究に始まると考えられる。この研究ではこれらの要素を英文から抽出するルールは語対応であり非常に直接的なものであった。その後、このような研究は広く行われ、日本文からのものとしては、80 年代の機能モデルにおけるデータフロー図を生成する大野ら¹⁵⁾の研究や、我々のものと同目的のオブジェクト指向用のモデリング要素を抽出する稲葉ら⁷⁾や滝沢ら¹⁶⁾の研究がある。しかし、大野と稲葉の研究は自動化システムではない。また、ともに構文解析主体で、途中結果を格フレームに展開せず、また意味辞書も持たず意味解析の段階までは至っていない。我々のものに最も近い研究は滝沢のものである。しかし、この研究では構文が明確に決まった単文を入力することになっており、単文への変換は手作業で行うことになっている。また、意味辞書としては名詞辞書の中にその概念として、語ごとにクラス、インスタンス、属性のどれになるかが記述されている。これでは 1 つの語が状況によってクラスになったり属性になったりする

場合に分析できないという問題がある。また、オブジェクト指向固有の問題としては、関連の抽出ができない、操作と活動が分離されていない、イベントのセクタとレシーバの区別が明確でない、能動態表現と受動態表現による区別が明確でないなど、適用性に大きな問題があり、試作の域を出ていない。

我々は今回の研究目的を、通常の日本語で記述された図 1 のような要求記述からオブジェクト図とイベントトレース図を自動生成するシステム Computer Automated Modeling Engine for Objects (CAMEO) を開発することにした¹⁴⁾。開発にあたっては、実用にある程度耐えられるシステム作りを念頭に置いた。そこで、利用者に要求を記述する構文を制限したり、問題ごとに語彙の概念を与えてもらうなどの制約を設けることは避けた。具体的には、図 1 に示すような一般のドキュメントを入力とした。たとえこの内容のいくつかの語彙を間違ってもモデル化しても、たたき台となる設計図が生成されればその後設計者が自ら修正するような開発作業を想定した。またシステムの開発にあたっては日本文の形態素解析や統語解析に関しては理論も確立され¹⁷⁾解析ツール^{11),12)}や辞書^{8),13)}も提供されているのでそれらをそのまま利用することにし、研究の主眼を自動化に適したオブジェクト指向分析のモデル化においた。

以下では、2 章において本システムの全体構成を、3 章においてオブジェクト指向分析に適した意味フレームの形式を、4 章において自動化に必要なオブジェクト指向分析ルールの条件部を構成する述語要素を、5 章において遺伝アルゴリズムによる自動レイアウトを、最終の 6 章においてまとめを述べる。

2. CAMEO の全体構成

CAMEO のシステム構成を図 2 に示す。解析は 5 ステップからなる。CAMEO は、まず入力された日本語要求文を、形態素解析 (①)、統語解析 (②)、意味解析 (③) の順に処理し、結果を意味フレームのネットワークに展開する。次に、この個々の意味フレームに対してオブジェクト指向分析ルールを適用し、クラス、関連、属性等のモデリング要素を抽出・分類する (④)。最後に抽出されたモデリング要素を自動レイアウトし、設計図を生成する (⑤)。なお、①と②については奈良先端科学技術大学院大学の松本裕治氏が開発した JUMAN¹¹⁾と SAX¹²⁾を利用した。また、構文解析用の辞書としては情報処理開発協会の辞書¹³⁾を、意味フレームに埋め込む語彙の概念には EDR の電子化辞書⁸⁾を用いた。以下では我々の研究に固有の

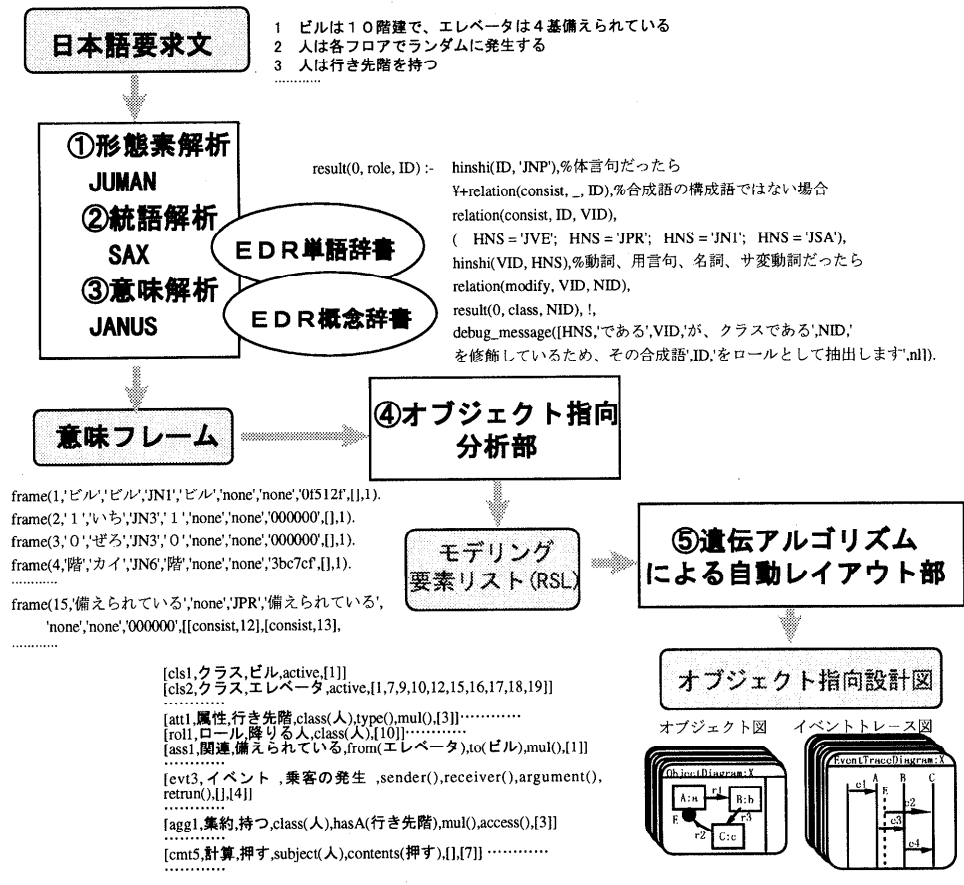


図2 CAMEO システム構成
Fig. 2 The architecture of CAMEO.

③, ④, ⑤ について詳しく論じる。なお、厳密には CAMEO は我々固有の部分 ③, ④, ⑤ を指すが、システム全体に対しても同じ呼び方をすることもある。

3. 意味 (格) フレームの形式と構成要素

CAMEO の意味解析部 JANUS (図 2 の ③) が生成した意味 (格) フレームは、図 3 に例を示すように、『frame(』から始まり以下の 10 種類の情報をこの順でカンマを挟んで並べ、最後に(』)からなるリスト形式を持つ。これらの 10 種類の情報は、語彙を単位にオブジェクト指向分析が自動的に行えるように選んだもので、語彙の情報と他の語彙との関係を表すものである。

1. 識別子 要求文全体での意味フレームの通し番号
2. 見出し語 要求文中での語彙の表記
3. 読み 語の読み
4. 品詞 EDR 単語辞書における品詞分類

5. 基本表記 語の基本形
 6. 活用型 語の活用型 (動詞に対して)
 7. 活用形 語の活用形 (動詞に対して)
 8. 概念 ID EDR 単語辞書を基本表記と読みをキーとして検索したとき得られる概念識別子
 9. 関係子 動詞の持つ格情報 (subject, at など), 語間の修飾関係 (modify), 句とその構成要素関係 (consist)
 10. 要求文番号 語の現れた要求文の番号
- 意味フレームの作成単位は、統語解析における構文木上の句を表す節以下の各節とし、これらをまとめて語彙と呼んでいる。名詞や動詞に対して意味フレームを生成するのは、EDR 電子化辞書には一部の例外を除いて、これらが単位として登録されているためである。さらに、その上位の体言句に対しても意味フレームを生成するのはロール (たとえば、『乗る』という動詞と『人』という名詞からなる合成語『乗る人』)

```

frame(1,'ビル','ビル','JN1','ビル','none','none','0f512f',[],1).
frame(2,'1','いち','JN3','1','none','none','000000',[],1).
frame(3,'0','ゼロ','JN3','0','none','none','000000',[],1).
frame(4,'階','カイ','JN6','階','none','none','3bc7cf',[],1).
frame(5,'建','ダテ','JB1','建','none','none','1f5a3d',[],1).
frame(6,'10階建','none','JNP','10階建','none','none','000000',[[consist,2],[consist,3],[consist,4],consist,5],1).
frame(7,'で','ダ','JJD','だ','判定詞','ダ列タ系連用テ形','2621ba',[],1).
frame(8,'10階建で','none','JPR','10階建で','none','none','000000',[[consist,6],[consist,7],[subject,1]],1).
frame(9,'エレベータ','エレベータ','JN1','エレベータ','none','none','0e68fe',[],1).
frame(10,'4','よん','JN3','4','none','none','000000',[],1).
frame(11,'基','キ','JN6','基','none','none','0eba98',[],1).
frame(12,'備え','ソナエ','JVE','備え','母音動詞','未然形','0f886e',[],1).
frame(13,'られて','られて','JJD','られる','none','none','000000',[],1).
frame(14,'いる','いる','JJD','いる','none','none','000000',[],1).
frame(15,'備えられている','none','JPR','備えられている','none','none','000000',[[consist,12],[consist,13],consist,14],[subject,9],
[range,10],[to,1]],1).
.....
frame(222,'想定して','ソウテイ','JSA','想定','サ変動詞','タ系連用テ形','3d0123',[[object,221]],20).
frame(223,'考えよ','カンガエ','JVE','考え','母音動詞','命令形','3cfa55',[[object,216]],20).
frame(224,'想定して考えよ','none','JPR','想定して考えよ','none','none','000000',[[consist,222],consist,223]],20).

```

図3 意味解析部が生成した意味フレーム

Fig. 3 Semantic frames generated by the semantics analysis part.

やイベントを取り出すためである。同様に用言句に対して意味フレームを生成するのは関連などを取り出すためである（たとえば、『備えられている』を語幹、『備え』で品詞や概念を、語尾『いる』で態を取り出して関連に分類する）。

概念ID(8)は、EDRの日本語単語辞書(25万語)を基本表記(5)で引いた結果得られる概念の識別子である。EDRには、このほか概念体系辞書(40万語)があり、この中で概念間の上位下位関係を元に概念が木構造で体系化されている。また、関係子(9)としては、実験の結果、オブジェクト指向分析を自動化するためには図4に示すようなものが適切であることが分かった。

図2のエレベータ問題をJANUSによって解析した結果の意味フレームの一部を図3に示す。エレベータ問題の要求文から、全部でおよそ224個の意味フレームが得られた。

4. 意味フレームからのオブジェクト指向分析

オブジェクト指向分析の手法は多くの書籍や論文に発表されている^{2)~7),18)}。しかしこのどれもが、人手による分析作業のための手順と若干の選別基準を述べるにとどまっている。したがって、人手で分析する際にも経験や常識に頼る部分が多く、本位田ら¹⁸⁾が指摘しているように知的CASE化するための詳細なモデル化が切望されている。

我々の目標である分析の自動化にあたっては、いっそうの精密なモデル化が必要なことは自明である。し

関係子	具体例
Subject (主体)	「～は(働く)」「～が」
Object (対象)	「～を(送る)」「～のを」
Sender (与え手)	「～から(送る)」
Receiver (受け手)	「～へ(送る)」「～に」
At (場所)	「～に(建てる)」「～へ」「～を」「～で」
From (場所・始点)	「～から(送る)」
To (場所・終点)	「～へ(送る)」「～に」「～まで」
Start (始状態)	「～から(～になる)」
Goal (終状態)	「(～から)～に(なる)」
Cause (原因・理由)	「～のため(失敗した)」「～に」「～から」「～で」
Tool (手段・道具)	「～で(書く)」「～を」「～に」
Component (構成)	「～から(構成される)」「～で」
Condition (条件)	「～ならば(働く)」「～で」「～なら」「～ったら」「～には」「～ごとに」「～から～まで(搬入する)」「～だけ」「～ばかり」「～のみ」「～しか」「～こそ」「～くらい」
Assert (判定)	「～だ」
Predictive (陳述)	「～と(言っている)」「～ものと」
Manner (様式)	「～に(欠ける)」「～で」
Topic (話題)	「～とは(～することだ)」「～って」「～も」「～など」「～なんか」「～なんて」
Degree (度合)	「～ほど(～する)」「～さえ」「～でも」「～すら」「～だって」
Viewpoint (観点)	「(～の観点)から(見直す)」
Modify (修飾関係)	「連体修飾などの語の修飾先を示す」
Consist (構成語)	「合成語の構成要素である構成語を示す」
Owner (所有者)	「～の～」

図4 関係子

Fig. 4 Relational operators.

かも、これまでの分析ルールがクラスや関連などのモデリング要素ごとに整理されているのに比べ、自然語文からの自動分析を行うには、語彙ごとにその意味フ

レームの格や概念を中心に、分析ルールを体系化する必要がある。したがって我々は、前章で述べた意味フレームから、クラス、属性、属性値、ロール、関連、集約、イベント、計算、シナリオの10種のモデリング要素を抽出・分類する独自の分析ルールを作成した。

分析結果は、後工程の自動レイアウトシステムに送るために、モデリング要素ごとに図5に示すような固有のスロットを持つRSL (Requirement Specification List language) と呼ばれるリスト形式で出力される。紙数の都合でRSL表記法の詳しいことは述べないが、基本的には、第1要素には抽出されたモデリ

```
[cls1, クラス, ビル, active, [1]]
[cls2, クラス, エレベータ, active, [1,7,9,10,12,15,16,17,18,19]]
[cls3, クラス, 人, active, [2,3,7,10,11,15]]
*[cls4, クラス, フロア, active, [2,13,16,17]]
[cls6, クラス, 階, active, [3,5,7,8,9,12]]
*[cls7, クラス, 乱数, active, 4]
.....
[att1, 属性, 行き先階, class(人), type(), mul(), [3]]
*[att2, 属性, 上下, class(乗る階), type(), mul(), [5]]
[att3, 属性, 行き先ボタン, class(エレベータ), type(), mul(), [6]]
*[att4, 属性, ボタン, class(行き先階), type(), mul(), [7]]
[att5, 属性, 2つの上下行きボタン, class(フロア), type(), mul(), [13]]
.....
[rol1, ロール, 行き先階, class(階), [3,5]]
*[rol3, ロール, 行き先ボタン, class(ボタン), [6]]
[rol4, ロール, 降りる人, class(人), [10]]
[rol5, ロール, 乗る人, class(人), [11]]
*[rol6, ロール, 上下行きボタン, class(ボタン), [14]]
.....
[ass1, 関連, 備えられている, from(エレベータ), to(ビル), mul(), [1]]
*[evt1, イベント, 備えられている, sender(エレベータ), receiver(ビル), argument(), retron(), [], [1]]
[evt2, イベント, 発生する, sender(人), receiver(), argument(), retron(), [], [2]]
[evt3, イベント, 乗客の発生, sender(), receiver(), argument(), retron(), [], [4]]
.....
[agg1, 集約, 持つ, class(人), hasA(行き先階), mul(), access(), [3]]
.....
[cmt1, 計算, 発生する, subject(人), contents(発生する), [], [2]]
*[cmt2, 計算, 乗客の発生, subject(), contents(乗客の発生), [], [4]]
*[cmt3, 計算, 用いる, subject(), contents(用いる), [], [4]]
.....
[cmt9, 計算, 点灯する, subject(ボタン), contents(点灯する), [], [8]]
.....
[sce1, シナリオ, type(定義), event(乗る), receiver(), return(), argument(), source(押す), [7]]
.....
[non22, 対象外, 従って, 5]
.....
```

図5 エレベータ問題の分析結果 (RSL形式)

Fig. 5 The analysis result in RSL format for the elevator problem.

ング要素のIDが、第2要素にはモデリング要素の種類が、第3要素には原文中の表記が、第4要素以降には属性やロールや集約についてはそれを持つクラスが、具体的にはイベントについてはセンダクラスとレシーバクラスが、計算についてはそれを行う主体となるクラスが、関連については始点クラスと終点クラスが、それぞれ配置され、最後の要素には原文の文番号のリストが配置されている。たとえば、第1行のモデル要素 cls1 は原文1中の『ビル』が active なクラスとして抽出されたことを示している。また第7行のモデル要素 att1 は原文3中の『行き先階』が『人』クラスの属性として抽出されたことを示している。なおここで、「対象外」は与えられた語彙がオブジェクト指向によるモデリング要素にならないときの分類先である。

CAMEO/Aのオブジェクト指向分析部(図2の④)の分析ルールは、意味フレーム中の各スロット値がどんな値かによって、先に示した10種類のモデリング要素を抽出・分類する。具体的にはこれらのルールは prolog で記述したホーン節集合から成る。分析ルールを構成するホーン節のヘッド部は、図6に示すように result(0, モデリング要素名, ID) の形式をしており、モデリング要素名に11種類の要素名(対象外が含まれるので 10+1=11になる)が入る。ボディ部には、各モデリング要素に分類する条件を表す述語が列挙される。この条件を構成する述語は大きく次の3つの種類に分けることができる。

- 語彙の品詞による述語 **hinshi**
例：語彙が名詞 'JN1' や用言句 'JPR' であるかどうかを表す条件。
- EDR 電子化辞書の概念体系辞書から得られる上位概念を表す述語 **is_upconcept_c**
例：図6の(1)に示すように、その語が名詞で、体言句の末尾で、「人間または人間と似た振舞いをする主体 '3aa911'」、「具体的あるいは抽象的存在物 '444e3f'」、「もの '444d86'」のどれかを上位概念に持ち、「物や物に対する指示的な呼称 '444e3f'」や「事物の属性 '444db1'」を上位概念に持たないならば、その語をクラスとして抽出する。
- 他の語との関係(関係子)を表す述語 **relation**
例：図6の(2)に示すように、動詞(例：乗る)がクラスである名詞(例：人)を修飾しているとき(modify格で関係している)なら、その合成語である名詞句(例：乗る人)をロールとして抽出する。

```

(1)
result(0, class, ID) :-
  hinshi(ID, 'JN1'),%その語が名詞であるとき
  relation(consist, ID1, ID),%合成語の構成語の場合
  hinshi(ID1, 'JNP'),%その合成語が体言句だったら
  relation_tail(consist, ID1, ID),%その合成語の末尾語であれば
  debug_message([ID,'は体言句',ID1,'の末尾語',nl]),
  include_concept(ID, CID),
  (UPCID = '3aa911' % 人間または人間と似た振舞いをする主体
  ; UPCID = '444c3f' % 具体的あるいは抽象的存在物
  ; UPCID = '444d86' % もの
  ),
  is_upconcept_c(CID, UPCID),
  \+is_upconcept_c(CID, '444c3f'), % 「物や物に対する指示的呼称」ではない
  \+is_upconcept_c(CID, '444db1'), !, % 「事物の属性」ではない
  debug_message([ID, 'の上位概念に', UPCID, 'があるため, クラスとして抽出します', nl]).

(2)
result(0, role, ID) :-
  hinshi(ID, 'JNP'),%体言句だったら
  \+relation(consist, -, ID),%合成語の構成語ではない場合
  relation(consist, ID, VID),
  ( HNS = 'JVE'; HNS = 'JPR'; HNS = 'JN1'; HNS = 'JSA'),
  hinshi(VID, HNS),%動詞, 用言句, 名詞, サ変動詞だったら
  relation(modify, VID, NID),
  result(0, class, NID), !,
  debug_message([HNS, 'である', VID, 'が, クラスである', NID, 'を修飾しているため, その合成語', ID, 'をロールと
  して抽出します', nl]).

(3)
result(0, association, ID) :-
  hinshi(ID, 'JPR'),%用言句だったら
  relation_tail(consist, ID, GOBIID), %合成語の語尾
  midashi(GOBIID, GOBI),
  ( GOBI = 'いる'; GOBI = 'できる'),
  debug_message([ID, 'の語尾に', GOBI, 'がありました', nl]),
  include_concept(ID, CID),
  (UPCID = '444dd8'%対象行為
  ;UPCID = '30f7e5'% 現象
  ;UPCID = '30f801' %移動;
  UPCID = '3f9856' %変化),
  is_upconcept_c(CID, UPCID), !,
  ( CASE = object; CASE = to;
  CASE = at; CASE = secondary ),
  relation(CASE, ID, NID),
  result(0, class, NID), !,
  debug_message(['用言句',ID,'の語幹が', GOBI, 'であり, さらに上位概念に', UPCID, 'があり, さらに', CASE, '格
  がクラスなので, 関連として抽出します', nl]).

```

図6 クラスに対する分析ルール (一部)

Fig.6 Analysis rule for class element.

- 語彙が特定のものであることを表す述語 *midashi*
例：図6の(3)に示すように、対象行為、現象、移動、変化などを表す用言句が、語尾に「いる」や「できる」を持ち、クラスを表す語彙を *object*, *to*, *at*, *secondary* 格に持つ場合、関連に分類する。
このような述語を用いて分析ルールを、クラス用に2個、属性用に2個、属性値用に1個、ロール用に1個、関連用に2個、イベント用に3個、継承用に1個、集約用に2個、計算用に3個、シナリオ用に1個、それぞれ用意した。ここで個々のモデリング要素ごとに複数個のルールがあるのは、語彙が合成語かその構成語であるかによって、ボディ部の条件が大きく異なることが多いからである。

5. 遺伝アルゴリズムによる自動レイアウト

CAMEO が生成する設計図は構造化オブジェクトモデリング技法 SOMM¹⁰⁾に基づいたものである。SOMM の特長はクラス間の3種の関係である、関連、継承、集約を、異なる方法で視覚化したことにある。具体的には、継承は派生図に、集約はクラスを表すオブジェクト図の階層構造に、関連はオブジェクト図内のクラス間の弧として視覚化する。このため、CAMEO はまずモデリング要素をこの3種の設計図に分配する。その後、各設計図内でのモデリング要素の適切な配置を求めるために、配置の見やすさに関する多くの評価基準に柔軟に対応できる遺伝アルゴリズムを用いる⁹⁾。具体的には、位置情報を遺伝子に持ち、1個体が1配置例である仮想生物集団を生成し、この仮

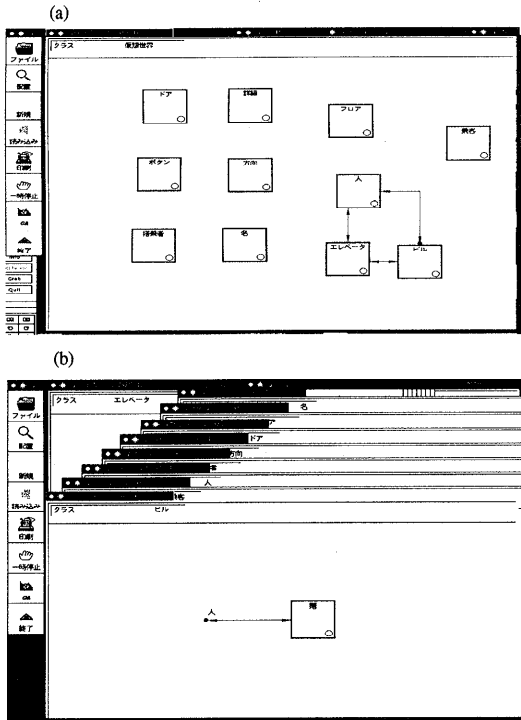


図7 CAMEO/Dが作成したオブジェクト図

Fig.7 Object diagram generated by CAMEO/D.

想生物集団に対して下記の評価基準の達成度の和を個体の環境適応度として、仮想生物集団の進化をシミュレーションすることで配置の最適化を行う。

評価基準は以下の5つである。

- (1) 弧の交差が少ない。
- (2) 頂点どうしの重なりが少ない。
- (3) 分布面積が狭い。
- (4) 頂点間の間隔が指定値 X ドットに近い。
- (5) 多くの頂点が x, y 軸上に並べて表示されている。

この結果、図5のRSLで与えられるモデリング要素が再配置され図7に示すような階層化された設計図が生成された。図7の(a)は問題世界を表すオブジェクト図であり、ここには図5のRSLで抽出されたクラス候補である、「ビル」、「エレベータ」、「人」などが描かれている。また「エレベータ」と「ビル」間には関連が正しく描かれている。さらに図7の(b)では、図7の(a)に描かれた各クラスに対し、その集約関係上の子であるクラスが描かれている。実際「ビル」には「階」が描かれている。

6. 評価

CAMEOはオブジェクト指向分析の自動化システ

ムのプロトタイプではあるが、実際に図1に示されたような文章群から図7に示すような設計図を生成することができた。生成された設計図は設計工程において十分にたたき台に使えるものであった。この結果は酒倉庫問題などの他の数個の事例でも同様であった。

実際、別グループでCAMEOの効果を測定した。3つの問題において、問題文を理解し手作業でモデリング要素を取り出しオブジェクト図を作成するに要する時間は、CAMEOが生成したオブジェクト図を修正して正しいものにするのに要する時間の約3倍かかることが分かった。

一方で以下のような課題がこの実験で明らかになった。個々に指摘したようにそのいくつかに対しては具体的な解決策をあげることができ、その実装は今後の課題である。

- 1) 余計なモデリング要素が抽出されている。

図1のエレベータ問題を解析した結果図5にその一部を示したようなRSLが生成されている。全分析結果には、クラスが8(4)個、属性が8(3)個、ロールが8(3)個、関連が1個、イベントが9個、集約が1個、計算が33(9)個、シナリオが5個、対象外が130個抽出されていた。ここで、()内の数は、人手で分析したときには現れない、言い換えれば抽出すべきでなかったモデル要素(図5中でその前に*のついている分析結果)の数である。このような誤った分析は以下のような理由による。

①『乱数』など実装の概念がクラスとして、また『次の要求』や『運転方向』などの抽象名詞が計算として、それぞれ誤って抽出されている。これらは、概念IDによる分類知識が粗いため、ルールを精密化することで対応可能である。

②『上下行きボタン』などの合成語や『降りる人』などの連体修飾付きクラスをロールとして抽出しているが、これらを関連のロールとして明示的にオブジェクト図に描く必要があることはまれで、ほとんどは単に文脈中でどのクラスを指しているのかを明示化するための修飾語にすぎない。この区別は表層的な意味から判断することは難しく、関連を基準にしてロールを抽出すべきである。ただし、関連は与えられた文に明示的に指定されていることはまれで、多くは問題文に出現したクラスに関する常識から抽出することが多い。したがって、常識やドメイン知識を機械的に表現し利用することが必要である。

③『の』の解釈が難しい。実際、『乗る階の上下に従って』という文内の『の』を『所有』の『の』として、また『上下』を『乗る階』の属性として、それぞ

れ誤って抽出している。この誤りは、『の』の意味解釈を表層格の意味素を用いて『所有』、『限定修飾』、『動作主格』の3つの格フレームに区分して照合することで解決可能である。

④ 同意語や類義語の同一化が行われていない。たとえば、『階』と『フロア』はこの問題においては同じ対象を表しているのに、モデル化においてはどちらかに統一すべきである。これは辞書の充実で対応可能である。

⑤ 『計算』や『イベント』に関しては総じてかなり正しく分類された。しかし、『用いる』、『要求』、『想定して考える』、『乗ることができる』など問題のモデル化の対象外の動詞や連体修飾が誤って抽出されることが少しあった。問題文にはモデル化の対象外も含まれることがよくあるが、これを識別することは一般に難しい。

2) イベントトレース図では、イベントの送受信者の格や計算の主体格をより正確にとらえる必要がある。

⑥ receiver 格や subject 格が未定のことよくある。これは問題文中の個々の単文には明示的に与えられていないからである。これを克服するには、文脈解析を行い、欠けている主格や対象格を推定する必要がある。

⑦ 文7の『行き先階のボタンを押す』と文8の『行き先階の指定を受けると』は、同じ計算を表している。シナリオを正確に分析するには、このような同じイベントや計算の異なる表現を同一視する必要がある。

⑧ 『運転方向に従って順序よく』、『先ず』、『上下に従って』など、処理の条件や範囲などを表す情報は現在無視されて捨てられている。この情報を適切に処理しシナリオにおけるイベントのタイミング情報として抽出しなければならない。

謝辞 この研究の一部はEAGL事業推進機構の助成金を使って行われた。JUMANとSAXを提供してくださった奈良先端科学技術大学院大学の松本裕治教授や電子辞書を提供してくださった情報処理振興事業協会の桑畑和佳子さんらには、使い方などでお世話になった。感謝いたします。

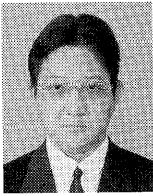
参 考 文 献

- 1) Abbott, R.: Program Design by Informal English Descriptions, *Comm. ACM*, Vol.26, No.11, pp.882-894 (1983).
- 2) Booch, G.: *Object-oriented Design With Applications*, Benjamin/Cummings (1990).
- 3) Coad, P. and Yourdon, E.: *Object-oriented Analysis*, Prentice Hall (1991).

- 4) Coleman, D., Arnold, P., et al.: *Object-oriented Development - The FUSION METHOD*, Prentice Hall (1994).
- 5) Runbaugh, J., Blaha, M., Premerlani, W., Eddy, F. and Lorenzen, W.: *Object-oriented Modeling and Design*, Prentice-Hall (1991).
- 6) Shlaer, S. and Mellor, S.: *Object-oriented System Analysis Modeling the World in Data*, Prentice-Hall (1988).
- 7) 稲葉稔智, 箕輪行真, 石原平太郎: 構文解析を用いたオブジェクト指向分析手法, 第41回情報処理学会全国大会論文集, No.5, pp.171-172 (1990).
- 8) EDR 電子化辞書仕様説明書(第2版), 日本語電子辞書研究所(1995).
- 9) 原田 実, 高橋史郎: オブジェクト指向分析によるモデル要素集合からの遺伝アルゴリズムを用いた設計図作成支援システムの開発研究, 人工知能学会第9回全国大会論文集, pp.363-366 (1995).
- 10) 原田 実, 澤田隆史, 藤沢照忠: 構造化オブジェクトモデリング技法 SOMM とその構築環境 SOME, 電子情報通信学会論文誌(D-I), Vol.J79-D-I, No.3, pp.158-171 (1996).
- 11) 松本裕治, 黒橋禎夫, 宇津呂武仁, 妙木 裕, 長尾 真: 日本語形態素解析システム JUMAN 使用説明書 version 1.0, 京都大学工学部長尾研究室(1993).
- 12) 松本裕治, 伝 康晴, 宇津呂武仁: 構文解析システム SAX 使用説明書 version 2.0, 京都大学工学部長尾研究室/奈良先端科学技術大学院大学松本研究室(1993).
- 13) 計算機用日本語基本動詞辞書 IPAL (Basic Verbs) —解説編, 情報処理振興事業協会(1987).
- 14) 大野雅志, 原田 実: オブジェクト指向分析支援システム CAMEO—日本語文章記述からの設計要素の自動抽出, 情報処理学会ソフトウェア工学研報, Vol.94-SE-99, pp.105-112 (1994).
- 15) 大野浩史, 菊野 亨, 鳥居宏次: プログラム設計のための構造表現モデルの提案と複合設計への適用, 情報処理学会ソフトウェア工学研報, Vol.89-SE-65, pp.1-7 (1989).
- 16) 滝沢陽三, 上田賢一: オブジェクト指向に基づく要求記述からの形式的仕様の導出手法, 情報処理学会ソフトウェア工学研報, Vol.93-SE-94, pp.57-64 (1994).
- 17) 田中穂積: 自然言語解析の基礎, 産業図書(1988).
- 18) 本位田真一, 山城明宏: オブジェクト指向分析・設計, 情報処理, Vol.35, No.5, pp.392-401 (1994).

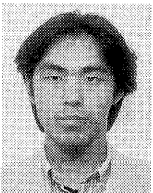
(平成8年12月26日受付)

(平成9年7月1日採録)



原田 実 (正会員)

1975年東京大学理学部物理学科卒業。1980年東京大学理学系大学院博士課程修了。理学博士。同年(財)電力中央研究所担当研究員。1989年より青山学院大学理工学部経営工学科助教授。自動プログラミングシステム、ソフトウェアの要求理解や設計の自動化、オブジェクト指向CASE、株式投資エキスパートシステム、自律ロボットシステムなどの研究を行う。1986年(財)電力中央研究所経済研究所所長賞受賞。1992年人工知能学会第6回全国大会優秀論文賞。訳書「ソフトウェアの構造化設計法」(日本コンピュータ協会)、編著書「自動プログラミングハンドブック」(オーム社)、監修書「CASEのすべて」(オーム社)など。IEEE, ACM, AAAI, 人工知能学会, 日本ロボット学会, OR学会, 経営工学会各会員。



野村 佳秀 (学生会員)

1996年青山学院大学理工学部経営工学科卒業。1998年同大学大学院修士課程修了予定。



山本 幸二

1996年青山学院大学理工学部経営工学科卒業。1998年同大学大学院修士課程修了予定。



大野 雅志

1992年青山学院大学理工学部経営工学科卒業。1994年同大学大学院修士課程修了。同年北陸電力システムサービス(株)入社。



田村 浩樹

1994年青山学院大学理工学部経営工学科卒業。1996年同大学大学院修士課程修了。同年(株)ソフトウェアリサーチアソシエイツ入社。



高橋 史郎

1993年青山学院大学理工学部経営工学科卒業。1995年同大学大学院修士課程修了。同年東日本旅客鉄道(株)入社。