

ドメイン情報を利用した要求モデルの生成支援環境

塚本 明 小飼 敬 上田 賀一

茨城大学

〒316-8511 茨城県日立市中成沢町 4-12-1

本研究は、分類されたドメイン情報を利用した要求概念のモデル化により要求分析を支援する環境の提案とその実現について述べる。要求物を構成する要素とその関連からなる要求モデルを作成することで、ドメイン理解に有用な要求物の図式表現を得る。要求モデルの初期モデルは要求記述を形態素解析・構文解析することで生成される。ドメイン辞書などのドメインモデルと同様に要求モデルもドメイン情報として扱い、これらの情報はファセット分類されリポジトリに格納される。新たな要求モデル作成の際に利用することで、体系的な再利用を支援した。また作成した要求モデルの利用例として、リッチピクチャへの変換を行い要求獲得のための新たなビューを得ることを実現した。

A Support Environment for Generating Requirement Model by Using Domain Information

Akira Tsukamoto Kei Kogai Yoshikazu Ueda

Ibaraki University

4-12-1 Naka-Narusawa, Hitachi, Ibaraki, 316-8511 Japan

In this paper, authors propose a support environment of requirement analysis by modeling requirement concept using categorized domain information. User obtains the diagram representation for understanding the domain by creating a requirement model which consists of elements and their relationships for required system. The initial requirement is generated by morphological and syntactic analysis to requirement description. The requirement model is treated as domain information like domain model such as domain dictionary. It is categorized by facet scheme and stored to repository. In new requirement modeling process, it is reused systematically. The conversion from requirement model to rich-picture is shown as an example of use for requirement acquisition.

1 はじめに

ソフトウェア開発工程の中で最初でかつ最も重要である段階は要求分析である。要求物が何であるかを正確に理解しなければ、要求者の望むものができることはない。要求分析は一般的に要求文書の読解、要求者との話し合いにより進められていくが、その分析結果が分析者の個人的な知識としてしか残されなかった。そのため作成された要求

概念を構造化した情報は明示的に再利用されていないと言える。その分析結果が分析者の個人的な知識としてしか残されなかった。また、近年、要求されるシステムが大規模化・複雑化するにつれて、システムの生産性と信頼性を向上させることが困難になってきている。ソフトウェアも同様で、その生産性と信頼性を向上させる手段として再利用が行われてきた。その中の技術の一つであるドメ

インモデリングは、ソフトウェアをコンポーネント単位に分割して再利用するのではなく、ソフトウェア作成した対象世界(ドメイン)での使用に限定して分析・モデリングを行い再利用するという手法である。要求概念を構造化・図式化することは対象世界を定義することと等しく、ドメイン分析の一つとして考えられる。これらのことから、本研究はドメイン情報を利用した要求獲得支援環境を提案する。これは、要求モデルと呼ぶ要求物を構成する要素とその関連からモデルを作成し、新たなビューとして図式表現を得ることでの要求獲得を目指している。そのモデルをドメインモデルとしてリポジトリに蓄積し、新たな要求モデルを作成する際に対象に近いドメインと比較させることにより、要求モデルの再利用を実現する分析支援環境を構築した。分析以後の工程で得られた情報もドメイン情報としてリポジトリにフィードバックすることで、より多くの知識を得ることも可能である。作成した要求モデルはオブジェクトモデル等の開発工程で作られるモデルの基礎として位置付けられる。今回は例としてリッチピクチャへの変換をサポートし、さらなるドメインの理解支援を行った。この環境を使用することで、経験を積んだ開発者の知識であり、事例として蓄えられた開発物の情報を利用した要求分析が可能となる。以下、本論文ではこの環境について述べる。2章は環境を構築するにあたっての基盤技術について、3章では環境の概要を述べる。4章でドメイン開発法について触れ、5章で実現した環境の設計と実装について述べる。6章では利用例を出し、7章で関連研究との比較を行う。8章でまとめと今後の課題を述べる。

2 基盤環境

2.1 リッチピクチャ

リッチピクチャとはソフトシステム方法論(SSM)のプロセスへの参加メンバによる状況の図的要約のことで、7ステージモデルのステージ2(発見)で利用される。分析や予想される結論を先取りしないように書く。要約であるから、主観的に重要だと思う情報が含まれていればよい。リッチピクチャは、SSMの実践プロセスへの参加者が状況につい

てゆるい共通認識をつくるための手段である。これによって、問題だと思われる状況の仮の範囲を設定することができる。したがって、リッチピクチャは、状況の変革に関連するシステムを抽出するための有力な情報基盤の一つとなる。手段であるから完成品をつくる必要はない。通常は、マンガ的な絵として書くが、テクニックが限定されているわけではない。関連システム抽出の基礎となるためには、状況の当事者のリストと各当事者の思惑を書き込むことが最低限必要である。要するに、「データ類から観察結果の解釈まで、状況の認識を絵図に描いて表現する」ことである。

2.2 ファセット分類

ファセット分類[1]とは、情報が持つ属性や性質をあらゆる観点から整理する方法である。一般的に言うと、源情報を何らかの方法で分類し、主類に分けたものに対して、ファセット分類するわけである。各主類は、固有のファセットを持つことになる。この分類法は、列挙型分類よりも記述的であり、柔軟性があるといえる。ファセット分類は、主にソフトウェアパーツの分類に適したものである。

2.3 ドメイン分析・モデリング

経験的再利用を体系化しようとする試みのひとつとして、提案されたドメイン分析・モデリング技術はどの分野でも再利用可能な形にソフトウェアを変換するのではなく、ある特定の対象世界(ドメイン)に限定して利用する方法である。ドメインに関わるプロセス全体をドメインエンジニアリングと呼び、これを行う技術者をドメインエンジニア(ドメイン開発者)と呼ぶ。

ドメイン分析の目的は、システムを開発する際に、過去に蓄積された技術や経験、ソフトウェア資源を有効に活用することで、再利用性を促進して、開発効率を高めることにある。また、要求獲得プロセスにおいて、発注者と受注者間の意図の相違を少なくして、発注者の望んでいるシステムを正確に開発するためにもドメイン分析は必要である。

ドメイン分析とは、この対象領域に属している問題群やソフトウェア群を入力として分析し、そ

れらに共通する要素を明らかにしてドメインの持つ性質を出力として獲得することである。こうして得られた出力をモデル化することでドメインモデル、ドメインに特化したモデルが生成される。システム開発者はこのドメインモデルを利用することでシステムの信頼性・生産性を高めることができる(図1)。

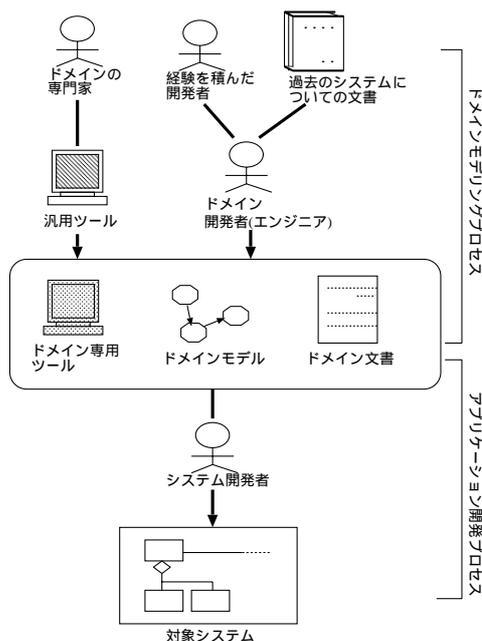


図 1: ドメインモデルを用いるシステム開発

3 本支援環境の概要

ここでは、本研究で行ったことについて、本研究室で以前に行われていた研究と比較しながら述べる。

3.1 以前の研究

以前の本研究室で同じく要求獲得を目的とする研究が行われた。これは、要求記述を形態素解析・構文解析し、品詞とその依存関係を決定することで、文書の静的構造を捉え、形態素からなる各要素とそれらを結ぶ関係をリッチピクチャとして自動生成するツールが開発された(図2)[2]。ユーザは生成されたリッチピクチャに編集を行い、ドメ

イン辞書から提供される各要素の画像を張り付けることで最終的なリッチピクチャを得るという形であった。

この研究は要求記述という文書だけでなく、リッチピクチャという直観的なビューを作成することで、クライアントとベンダ間の意志の疎通を円滑に行うことで要求獲得支援を目指していた。また部分的ではあるが、ドメイン情報を利用した例でもある。

しかし、生成されたリッチピクチャの再利用性については考えられておらず、一度生成したことのある文書でも生成工程を初めから行う必要があり、解析、編集によって得られた静的構造の情報が無駄になっていた。

また、リッチピクチャ生成というビューを得ることのみを目的としていたため、生成物をその後の過程に利用することが難しかったと言える。

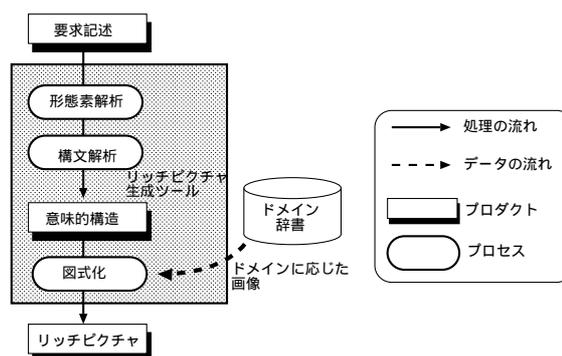


図 2: 以前の研究

3.2 概要

本研究で目的とするのも同じく要求獲得を含めた分析支援である。

以前の研究で問題となった点を以下にまとめる。

1. 生成したリッチピクチャは保存対象となっておらず、変更があると最初の段階から実行する必要があり、効率が悪い
2. リッチピクチャというビュー表現から他のモデルへの変換は再利用性が低い
3. 分析を支援するためのドメイン情報の利用割合が低い

これらの問題点を解決するために以下のことを行った。

1. 単一のビュー表現ではない他のモデルへの変換を考慮にいった基本モデルへ作成対象を変更
2. 作成した基本モデルを含むドメイン情報を利用した再利用性の向上

以上のことから本研究は作成対象をリッチピクチャに限定するのではなくそれらの要求物の構成要素とその関連を表した基本モデルである要求モデル(対応するドメイン情報を付加することでリッチピクチャのようなビューやUML図のような設計段階でのモデルへの変換を行うことができる)とし、また、作成した要求モデルはリポジトリに蓄積され新たなモデル作成時にその情報を利用することによって再利用性の向上を目指した分析支援環境を構築した。また分析段階だけでなく、ドメイン分析・モデリングに基づいた方法論でソフトウェア開発工程全体を支援することも考慮にされる。今回構築した環境の概観を図3に示す。

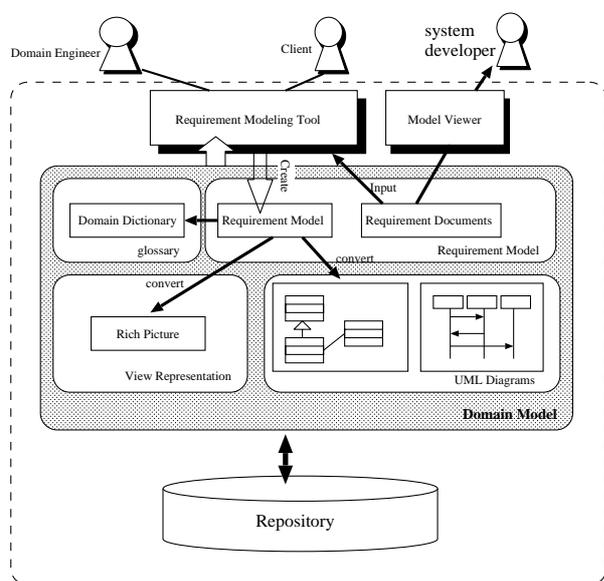


図 3: 本環境の概念図

3.3 ユーザ

本研究で開発した環境の対象となるのはソフトウェアの開発を行いながら、研究もしくは開発対

象世界(ドメイン)を発展させる開発現場である。その中でも対象となるユーザは主としてドメインエンジニア(開発者)である。経験を積んだドメインエンジニアやドメインの専門家によってドメイン情報が蓄えられた場合、経験の浅い開発者が本環境を利用することで、要求分析・モデリングを学習していくことも可能であると考えられる。またクライアントである要求者での使用も想定している。そのため、要求モデル作成のインターフェースは直観的でわかりやすいものとなっている。

3.4 リポジトリ

要求モデルを始めとした様々なドメイン情報を格納する。テキストデータ以外にもリッチピクチャやUML図といった図表現もドメインモデルとして保存対象としている。リポジトリの情報は、ドメインというカテゴリで分類されている。ファセット情報を付加することで各ドメインを特徴付けることが可能である。この概念を取り入れたことで、より柔軟な検索ができるようになっている。

このリポジトリに含まれる情報を分類すると以下ようになる。

- 構造情報
主に要求モデルにおけるモデル中の要素とその間の関係の情報。
- 意味情報
各ドメイン、各要素、各関連の意味を表す情報。ドメイン辞書として扱われることが多い。
- 画像情報
リッチピクチャやリッチピクチャを構成する要素の画像情報。直観的な理解に役立つ。
- 分類情報
各ドメインモデルを分類するための情報。ドメインをファセットや継承関係に基づく階層情報によりカテゴリ分けする。

4 要求モデル生成支援ツール

4.1 要求モデル

要求物を構成する要素とその関連を表現したモデルを要求モデルと呼ぶ。言い換えれば、要求モ

デルは要求概念を可視化・図式化したものであり、ユーザの要求をそれ以降のソフトウェア開発工程へ伝搬する役割を担うモデルである。本研究の要求モデルは要素、関連、サブドメインの3つのオブジェクトで構成する。要求モデルは要素、サブドメインをノードとし、関連をアークとしたNAモデルで表わされる。要求モデルを作成することで、対象世界のデータ構造や構成要素の性質や機能を表現することができる。

要求を可視化することは、対象世界を分析することに他ならない。このことから、作成した要求モデルはドメイン分析のプロダクトと考えて良い。本研究ではこの要求モデルをドメインモデルの一部として利用する。

図4は図書館管理システムにおける要求モデルの例である。

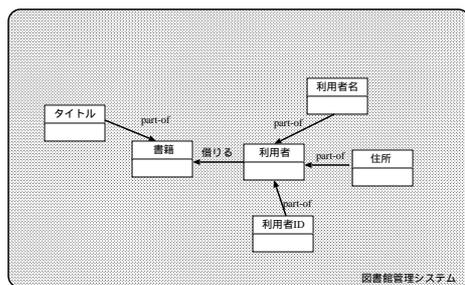


図4: 要求モデルの例

この図の例より「利用者」というデータ構造は「利用者名」、「利用者ID」、「住所」という属性で構成されることがわかる。また「利用者」は「書籍」に対して「借りる」という関連があることが表されている。

以下で要求モデルを構成するオブジェクト(図5)について記す。

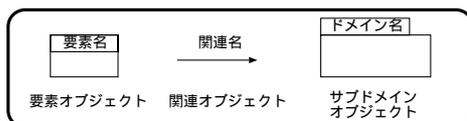


図5: 要求モデルのオブジェクト

4.1.1 要素

各ドメインに存在する要素(実体)を表すオブジェクトである。要求モデル中では矩形の中に要素名を書くことで表す(図5左)。各要素は要素名をもち、所属するドメインに応じた意味やそのドメインにおける重要度を示す重みももつ。要求記述中に現われた名詞は、要求モデルにおける要素として表現される可能性が高いといえる。

4.1.2 関連

各ドメインにおいて、要素間、サブドメイン間の関連を表すオブジェクトである。要求モデル中では有向線に関連名を書くことで描かれる(図5中)。矢印の向きは、関連元と関連先を表している。各関連は関連名をもち、所属するドメインに応じた意味やそのドメインにおける重要度を示す重みももつ。主に要求記述上に現われる動詞が対応し、図4の例では全体-部分構造を表す part-of 関連や「借りる」という対象-被対象間の関係を表すものがある。

4.1.3 サブドメイン

要求モデルを使って定義するドメインにおいて、ある機能のまとまりをサブドメイン(システム)として表すためのオブジェクトである。これにより、階層的な定義も可能になっている。要求モデル中では、矩形2つを組み合わせた形で表される(図5右)。通常のドメインと同様に、サブドメインも一つのドメインとしてリポジトリに格納する。要求空間における既存のシステムやそれ単体で一つのシステムと考えることのできるものをサブドメインとして表すことで、階層的に要求を定義することができ、より柔軟な理解支援が行える。

銀行業務ドメインにおけるATMシステムなどがこのサブドメインとして表される例である。

4.2 機能

本ツールでサポートするのは、要求モデル生成、要求モデル編集、要求モデル評価、要求モデル保存の各段階である。

以降に4つの各機能について説明する。

4.3 要求モデル生成

この段階の入力は自然日本語で書かれた要求記述であり，出力は自動生成された要求モデルである．ただし，この要求記述文は単文で構成されているものとする．入力された要求記述を形態素解析・構文解析し，抽出した名詞，動詞をそれぞれ要求モデルの要素，関連として初期モデルを生成する．この段階で完全なモデルを作成することは目的ではなく，以後の段階で編集されることを考えており，要求モデル生成支援の立場をとっている．

4.4 要求モデル編集

生成された要求モデルを変更し，より完全な要求モデルに近づけることがこの段階の目的である．この段階で主に行われる編集操作を以下にまとめる．

- 新たな要素の生成，既存の要素の変更，削除
- 新たな関連の生成，既存の関連の変更，削除
- 各要素，関連に対する重要度の変更
- 各要素，関連に対する定義（意味）の追加

4.5 要求モデル評価

この段階では入力である要求モデルを既存の要求モデルと比較し，評価を行うことで，モデルの作成支援することを目的としている．要求モデルが経験をつんだ開発者やドメインの専門家によって作成され，事例ベースとして蓄積されている場合，この比較は有効に働くといえる．

比較対象となるドメインモデルをリポジトリから検索するための方法は以下の3つである．

- 要求モデルの名前，構成要素に対するキーワード検索
- 要求モデルを生成したもとなった要求記述に対するキーワード検索
- 要求モデルの属するドメインに対するファセットに対する検索

検索により適用するモデルが決定されると，以下のことに関して比較を行い，ユーザに通知する．

- 重要度の高い要素が欠如していないか
- 各要素，関連の定義（意味）が異なっていないか
- 同じ意味の異なる表現の要素（類義語）が含まれていないか

また比較だけでなく一般的なモデル評価も行う．実際の要求物の構成要素であれば，必ず他の要素と関連を持つことから，要求モデル作成で関連の張られていない要素オブジェクトが存在することは望ましくない．この場合，その旨を表示することで要求モデルの生成支援を行う．

4.6 要求モデル保存

この段階で作成された要求モデルをリポジトリに登録する．この要求モデルはドメインモデルとして扱われるため，どのドメインとして保存するかが問題となる．このために，分類情報として，対象要求モデルの特性であるファセットを同時に登録する．保存フォーマットをXMLとしているため，データの交換性と可読性が向上している．

4.7 実装

この要求モデル生成支援ツールの実装は JAV A で記述されたモデリングツールである Argo/UML[3] をベースに要求モデルのためのライブラリを作成することで実現した．これにより，本ツールはプラットフォームを選ばずに実行することが可能である．リポジトリにはフリーのデータベースである PostgreSQL を用い，ツールとDB間のインタフェースには JDBC を利用した．前述のとおり，形態素解析/構文解析ツールには，それぞれ JUMAN/KNP[4][5] を用いた．ツールの実行画面を図 6 に示す．

5 関連研究

この章では本研究に関連した研究について触れ，比較した結果を述べる．

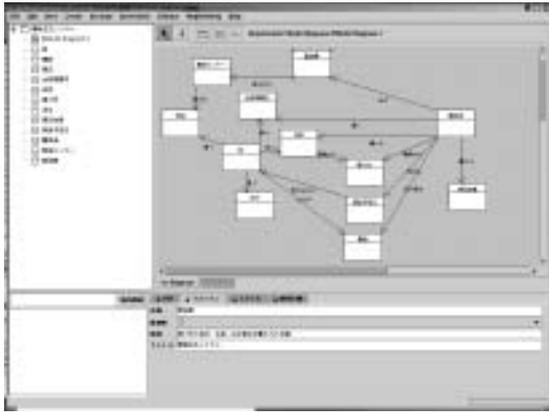


図 6: 実行画面

5.1 Odyssey

Regina M. M. Braga らは Odyssey[6] と呼ばれるドメインモデルに基づいた再利用環境を開発した。Odyssey はあるドメイン内のコンポーネントベースソフトウェア開発を支援する。具体的に言えば、Odyssey はドメインモデルを元に対応するコンポーネント群を結びつけ、その組合せによって対象のシステムを作り上げようという方法をとっている。Odyssey はドメインモデルをその抽象度によって概念モデル、構造モデル、実装モデルの3つに分類している。

概念的にはドメインエンジニア/ドメインエキスパートが作成したドメインモデルを利用し、ソフトウェア開発者が実際のシステムを組み上げることから本研究に似た研究であると言える。

しかし、Odyssey ではドメインモデルの利用に重きが置かれ、ドメインモデルの作成、およびリポジトリへの蓄積についてはほとんど触れられていない。

またドメインモデルを利用するにあたり最も重要となるどのドメインモデルを適用するか、つまり分析段階での支援が行われていない。

これに対し、本研究では要求記述からのモデル生成支援、ファセット情報を利用したことによるドメイン分析支援を行うことで、ユーザの分析に関わるコストを軽減していると言える。

5.2 知識処理を利用した要求モデル構築支援

角の行ったこの研究は要求者が自分自身の要求を認識する作業と、その結果をソフトウェア設計に利用可能な要求モデルに変換する作業を行き来しながら進めることが可能な要求モデル構築支援システムを開発することを目的としている[7]。この研究では発想支援システムを構築し、これを利用することで構築された要求空間上に認識された要求概念をソフトウェア設計に反映するために要求モデルとして構造化をしている。また要求モデル生成の自動化のために知識処理アプローチをとり、事例知識からの構造抽出や新規モデル合成における推論を利用できる。要求概念を可視化・構造化し、要求獲得、要求分析を行おうという考え方は本研究と類似するところである。しかし、この研究ではドメイン分析についてほとんど触れられていない。要求モデルの表現がドメイン指向的な考え方であることは述べているが、ドメイン概念を用いて要求モデルを分類していない。適応できるドメインモデルをファセットを用いて検索可能な本研究はより柔軟なドメイン決定支援が行えると言える。

また、この研究では対象が要求定義段階のみであり、作成された要求モデルの具体的な利用については述べていない。それに対し、本研究ではドメインに基づいて分類されたその後の開発工程のモデルへの変換も考慮に入れた環境作りをしている。このことから本研究はより有効なドメイン情報の再利用を行っていると言える。

6 まとめ

本論文は、ソフトウェアの要求モデルの構築支援による分析支援環境の開発について述べた。

ソフトウェア開発工程で重要である要求定義工程に着目し、ユーザが要求者の求める対象世界の概念を可視化することで設計に利用可能な要求モデルとして表現する作業を支援するシステムの構築を目的とした。

ユーザの要求をモデル化するために、要素オブジェクトと関連オブジェクトによって簡潔なルールで対象物を定義できるように要求モデル生成支

援ツールを開発した。要求物を2次元の図として表すことで、ユーザ自身の理解が曖昧であった箇所を明確にし、より深い要求獲得を行うことができる。さらに自然日本語で書かれた要求記述を形態素解析・構文解析を行い、要素とその関連を抽出することで、要求モデル生成を半自動化し、モデル生成のコストを軽減することにも成功した。

また過去に開発されたソフトウェアの要求物やドメイン情報を新規の要求モデルとを比較することで、抽象的で未定義な部分の発見を促すことで要求分析の支援を行うこともできた。

本研究は、要求概念のモデル化による要求獲得・要求分析の方法論を提案し、その実装を行った。要求モデルを含めたドメイン情報を蓄積し、管理するリポジトリを開発することで、ソフトウェア資源の再利用性を向上させた。

7 今後の課題

● 要求記述の形式化

本研究で入力となる要求記述は限定条件として単文で表現されているため、条件文や時系列に沿った流れを表すには不向きである。そこで、要求記述を一度それらの形式化された記述に変換し、その後要求モデルを生成することでより多くの情報を含んだモデルが作られると考える。その形式には、要素に対する条件や連続した流れを表す動的な構造、参照関係などが含まれる必要がある。

● 他のドメインモデルへの変換ツールの作成

本研究では要求モデルの変換例としてリッチピクチャを生成したが、要求モデルを別の形へ変換することも可能であると考えられる。具体的には、設計段階で有用なクラス図、シーケンス図などがあげられる。そのような変換を行う際、不足する情報をドメイン情報としてあらかじめ蓄えておくことで実現できると考える。要求モデル及びドメインモデルの利用の幅を広げることでソフトウェア開発工程の全体の支援が可能であると考えられる。

参考文献

- [1] Chung-Hong Lung, Joseph E. Urban : “An Approach to Classification of Domain Models in Support of Analogical Reuse”, ACM SIGSOFT Software Engineering Notes(Aug, 1995)
- [2] 黒田 剛, 塚本 明, 上田 賀一: “要求記述の静的側面の可視化によるコンピュータ支援共同作業の実現”, 茨城大学大学院理工学研究科情報工学専攻修士論文 (2000, 2)
- [3] Argo/UML : <http://argouml.tigris.org/>
- [4] Sadao Kurohashi, Toshihisa Nakamura, Makoto Nagao, Yuji Mastumoto : “日本語形態素解析システム JUMAN の改善”, International Workshop on Sharable Natural Language Resources, [http://cl.aist-nara.ac.jp/lab/events/SNLR/snlr/html\(1994\)](http://cl.aist-nara.ac.jp/lab/events/SNLR/snlr/html(1994))
- [5] Sadao Kurohashi, Makoto Nagao : “KN Parser : Japanese Dependency / Case Structure Analyzer”, International Workshop on Sharable Natural Language Resources, [http://cl.aist-nara.ac.jp/lab/events/SNLR/snlr/html\(1994\)](http://cl.aist-nara.ac.jp/lab/events/SNLR/snlr/html(1994))
- [6] Regina M. M. Braga Claudia M.L.Werner Marta Mattoso : “Odyssey : A Reuse Environment based on Domain Models”, Proceeding of IEEE symposium on Application-Specific System and Software Engineering Technology(ASSET'99), 1999, March
- [7] 角 康之: “ソフトウェアの要求モデル構築支援に関する研究-要求獲得における発想支援と要求モデルの構造化支援-”, 東京大学工学系研究科 情報工学専攻 博士学位論文, (1995, 2).