

ドメイン分析・モデリング概説

伊藤 潔, 田村恭久(上智大学)
杵嶋修三(山武ハネウエル)

ドメイン分析・モデリングは、対象システム自身が本来もつ各種の性質や開発上の多様な知識を十分に分析し認識して組織化し、システムの開発に有効な、共通の対象領域(ドメイン)に属する、用語、問題の捉え方、システムの構造、システムの作り方等の、固有な概念構造を得るプロセスである。この概念構造をドメインモデルと呼ぶ。このドメインモデルを用いて、複雑で大規模な実際のシステム開発での生産性の向上と再利用の促進を図ろうとしている。得られたドメインモデルは、そのドメインに属するシステムを繰り返して開発する際の分類モデル、ひな型モデル、標準モデル、ハンドブック等として使われる。本概説では、ドメインの意味、ドメイン分析・モデリングの目的とプロセス、ドメインモデルの構成要素・記述手段・種類を述べ、また、ソフトウェアの再利用、ドメイン工学、要求工学、設計、リバースエンジニアリング等の観点からソフトウェア工学との関係を述べる。

Survey on Domain Analysis and Modeling

Kiyoshi Itoh Yasuhisa Tamura
Sophia University

Kioi-cho 7-1, Chiyoda-ku,
Tokyo 102, Japan
{itohkiyo,ytamura}@sophia.ac.jp

Shuzo Kishima
Yamatake-Honeywell

Goudo-cho 134, Hodogaya-ku,
Yokohama 240, Japan
kishima@atc.yamatake.co.jp

Domain Analysis and Modeling is the process which obtains the domain specific domain model with the terminologies, the problem solving / system analysis / system design strategies, the target system structures, etc. by analyzing, identifying and organizing the properties of a family of target systems and the heuristic and reasonable knowledge for a family of practices of system development in order to improve the software productivity and reusability. The domain model can be used as a prototype model, a reference model, a handbook, etc. in order to develop many instances within identical domain repeatedly. This survey shows the meaning of "domain", the purpose and process of DAM, the components, representation styles and types of domain models, the relationships for software engineering technologies from the viewpoints of software reuse, domain engineering, requirements engineering, design activity, reverse engineering.

1. ドメイン分析の定義と効用

ソフトウェアやシステムの開発の基幹技術として、ドメイン分析・モデリング技術の重要性が認識されている。ドメイン分析・モデリング技術は、元々はソフトウェアの再利用を促進する手段として注目された。ここでは、既存のソフトウェア群をモデル化し、それを新システムの開発時に再利用しようとする。

現在の開発手法、ツール、ソフトウェア部品の多くは、標準的な作業規約に基づいて、一般的な業務処理のソフトウェアの正確な開発を目指して作られている。対象ドメインの認識が必ずしも十分ではなかったり、あまりにも汎用性を追求しようとして、適用の容易でないものもある。実際のシステムの開発現場では、これらの汎用的な手法やツールを、各々のドメインのシステム開発にやむを得ず適用しようとして、非効率さと不具合を生ずる問題がある。また、汎用的であるがため、各々の対象ドメイン向けの再利用部品群の同定が容易でないという問題もある。更に、これらの汎用的な手法やツールの適用を断念して、開発現場でこれまで長く採られてきた手作業をそのままの手順でコンピュータ化するにとどまっている場合も少なくない。

システムの開発では、単品として新たに開発されるものは高々2割程度といわれる。多くの場合、過去の同種システムの開発で蓄積された技術、経験、ソフトウェア資源を有効に再利用して、次のシステムの開発を図ろうとするのが現状である。この現状を踏まえた上で、ドメイン分析・モデリング技術は、同種のシステム全体の対象ドメインを明確に認識し、その対象ドメインに固有な性質や知識を包含したドメインモデルを構成し、このドメインモデルを基礎とした再利用部品群、記述言語、ツール等を提示しようとする。

Neighborsは、これをDRACOと呼ぶ再利用を主な目的とする開発環境上で実現した[Neighbors89]。Arangoは、これを分析-モデリング-再利用ループと呼ぶ[Arango92]。

Neighborsは、ドメイン分析を「あるドメインで、そのドメインの専門家が重要と考えるオブジェクト、操作、関係づけを同定しようとする試み」と定義した[Neighbors81]。また、Prieto-Diazは、「新システムを構築する際にこれまでの情報を再利用する目的で、ソフトウェアシステムが開発される際に使用した情報を同定し獲得し組織化する

プロセス」と定義した[Prieto-Diaz90]。田村・伊藤・杵嶋は、「対象システム自身が本来もつ各種の性質や開発上の多様な知識を十分に分析し認識して組織化し、システムの開発に有効な、共通の対象領域(ドメイン)に属する、用語、問題の捉え方、システムの構造、システムの作り方等の、固有な概念構造を得るプロセスである。この概念構造をドメインモデルと呼ぶ。このドメインモデルを用いて、複雑で大規模な実際のシステム開発での生産性の向上と再利用の促進を図ろうとしている」と定義した[田村94]。得られたドメインモデルは、そのドメインに属するシステムを繰り返して開発する際の分類モデル、ひな型モデル、標準モデル、ハンドブック等として使われ、ソフトウェア開発の生産性と再利用性を向上させる。

ドメイン分析・モデリングの効用は、(1)ドメイン毎に異なるモデルを採用するので、ドメインの特徴をより効率的に記述可能。(2)汎用のソフトウェアライブラリではなく、ドメイン毎にライブラリを蓄積するので、従来のソフトウェア再利用技術と比較して高い再利用率と品質が期待。(3)要求定義から保守に至る様々なフェーズで、そのドメインの観点から情報を再利用するため、従来に比べソフトウェアの生産性が向上。

(1)(2)については、既存のソフトウェアを部品化しても、その用途が汎用的であったり、逆に1つのアプリケーション固有な問題に過度に依存すると、部品の再利用性は低下する。このため、ドメイン分析・モデリングでは、同種のアプリケーション群から成る特定のドメイン毎に、共通に必要な機能を抽出し、その機能に沿って部品化を行い、部品の再利用の向上を促す。(3)については、ソフトウェアの部品の再利用だけではなく、要求工学等の他の開発フェーズでも、ドメイン分析・モデリングが注目されている。

2節ではドメインの意味を、3節ではドメイン分析・モデリングの目的とプロセスを、4節ではドメインモデルの構成要素・記述手段・種類を述べる。5節では、ドメイン工学、要求工学、設計、リバースエンジニアリング等の観点からソフトウェア工学との関係を述べる。6節で今後の課題をまとめる。

2. ドメイン

ドメイン分析でのドメインは、複数の類似した問題やソフトウェアが属する対象領域を指す。こ

これらの問題群やソフトウェア群とドメインの関係は、オブジェクト指向のインスタンス群とクラスの関係とはほぼ同一である。即ち、個々の問題やソフトウェアは、ドメインというクラスのインスタンスとみなせる。ドメインの性質や挙動は、問題群やソフトウェア群を分析し、それらに共通する要素を明確化する事で得られる。

例えば、A銀行・B銀行のオンライン処理等、「解くべき問題」が複数ある時、これらからオンライン処理問題のドメインを抽出できる。銀行には、顧客、口座、支店等の構成要素、また預金の預け入れや払い戻し、借入れ等の動作、更に口座番号や利率等の属性がある。これらの共通概念が「オンライン処理」問題のドメインを構成する。一方、顧客数、支店数、サービス内容、利率等は銀行毎に異なる。「オンライン処理」ソフトウェアのドメインは、これらを可変のパラメータとして扱う。あるドメインのソフトウェアを開発する場合には、個々に特有な構成要素を付加したり、パラメータに固有の値を与える。

ドメイン分析・モデリングでは、ソフトウェア開発での「解くべき問題」と「その問題を解決するアプリケーション」を区別する事が有効である。[Tracz94]では問題空間と解空間、[Schoen91]ではドメインモデルとタスクモデル、と呼んでいる。[Lubars91]や[Neighbors89]では、後者をアプリケーションドメイン、[Garlan93a]では、アプリケーションフレームワークと呼んでいる。これらを総合すると、問題ドメインとアプリケーションドメインという用語で区別するとよいと考える。問題ドメインは、類似した複数の問題から、共通した構成要素、関係、属性、振る舞い等から成り、いかにして解かれるべきかという情報を含まない。一方、類似した問題群に対しては、類似したソフトウェア群が開発される。アプリケーションドメインは、このソフトウェア群が共通してもつ構成要素、関係、属性、振る舞い、ソフトウェア部品等を抽出したものから成り、解くべき手順、方法、仕掛け等の、問題自体には含まれない情報を含む。

3. ドメイン分析・モデリングの概観

3.1 ドメイン分析・モデリングの目的

筆者らはドメイン分析・モデリングの目的を次のように整理する。元来の目的は(a)であるが、明示はあまりされていないが、(b)(c)も重要な目的であると考ええる。

(a)ドメインに属する過去の生産物(ソフトウェア自体、ソフトウェア部品、開発フェーズでの仕様書等)の再利用、(b)ドメイン特有/固有の開発プロセス・方法・ツールの体系化/明確化/改良/創生、(c)組織内での開発事例/ノウハウ/経験/方法の収集/体系化/教育/継承。

3.2 分析プロセス

ドメイン分析のプロセスの方法として、Arangoの方法[Arango91]、Prieto-Diazの方法[Prieto-Diaz87,91][Frake92]、FODA[Kang90]、IDoA[Lubars91]、Traczの方法[Tracz93]、Fraserの方法[Fraser93]等がある。

現状では、広範囲のドメインに適用できるDAMプロセスは、未確立である。まず、ドメインに共通に利用できる汎用的であるが、かなり単純化した、ArangoのDAMプロセス[Arango91]を紹介する。そこでは、(1)ドメインの同定、(2)情報獲得、(3)ドメインモデルの表現、(4)進化、(5)評価・検証、から成る。即ち、対象ドメインを明確にし、当該ドメインに関する様々な情報を収集する。それらの情報をモデルとして表現する。一旦ドメインモデルを構成した後も、ドメイン自体の変化に追従させるためドメイン分析プロセスを繰り返す、ドメインモデルを進化させる。また、ドメイン分析プロセスの繰り返しや実システムへのフィードバックによりドメインモデルの正当性や有効性を検証する。

Arangoは次の様々な情報ソースからの知識をドメイン分析プロセスの入力とした。(1)技術文献:教科書、技術雑誌、マニュアル等、(2)既存のアプリケーション:ソースコード、設計資料、ユーザマニュアル、実装をリバー エンジニアリングで解析した結果等、(3)顧客・市場調査、(4)問題ドメインの専門家の知識、(5)対象ドメインのシステム設計の専門家の知識、(6)ドメインの修正や進化の記録。例えば次のドメインモデルが出力される。(1)問題とソフトウェアの仕様化に使われた概念の定義、(2)ソフトウェアの設計の代替案、トレードオフの定義、(3)ソフトウェアの実装の計画。

Prieto-Diazは、ドメイン情報の収集・ドメインの分析・再利用部品の生成から成るドメイン分析のプロセスをコンテキスト図とデータフロー図で表現した[Prieto-Diaz87]。また、Prieto-Diazは、ライブラリ科学の多面的分類スキームを利用したドメイン分析の手順を明確化した[Prieto-Di

az91][Frake92]。ここでは、再利用可能な部品の集まりを組織的に整理するために、(1)対象の集まりから代表的な項目を選ぶ、(2)それらの項目やその属性をグループ分けし、分類のためのいくつかの側面を定義する、(3)側面を順序づける、(4)各項目を側面毎に分類する、等の手順に従う。

FODA(Feature Oriented Domain Analysis)では、コンテキスト分析・ドメインモデリング・アーキテクチャモデリングとよぶ3つのステップで、ウィンドウ管理システムドメインを例に挙げて、ユーザに見える対象システム群の特徴(feature)を抽出するドメイン分析のプロセスを示した[Kang90][Cohen94]。

Traczのドメイン分析のプロセス[Tracz93]は、(1)ドメインの範囲の決定、(2)ドメイン固有の概念・要求を定義し明確化、(3)ドメイン固有の設計・実装に伴う制約を定義し明確化、(4)ドメインモデル(DSSA, 5.6節)を構成、(5)再利用部品を生成・集積である。

IDeAでは、同様な問題群の解の分析、同一ドメインとして解を特徴化、ドメインを抽象化/境界の限定という3つのステップから成るドメイン分析のプロセスを示した[Lubars91]。

Fraserは、ドメイン分析はチームによる作業である事が多い事から、グループウェアによるドメイン分析の手順や仕事分担を議論した[Fraser93]。

ドメイン分析の具体例としてビジネスアプリケーションに対するRaytheon社(例えば[Prieto-Diaz87]参照)、ミサイル用ソフトウェアに対するCAM Pプロジェクト(例えば[Arango91][Prieto-Diaz87]参照)、計測制御ソフトウェアに対するAsdreas S TD Triad[杵嶋93][Itoh94a][伊藤94]がある。

3.3 ドメイン分析の進化と知識獲得

ArangoとPrieto-Diazは、ドメイン分析プロセスを、知識や情報を獲得しながら進化するプロセスであると捉える[Arango91][Prieto-Diaz87]。ドメインは少しずつだが、概念の意味、仮定、実装技術等が常に変化したり置き換わる。このため、以前のドメイン分析プロセスで得られたドメインモデルは、徐々に更新/進化させていく必要がある。このドメインモデルの進化のプロセスを知識の集積と獲得と捉えられる。

Schoenは、ドメインの専門家の知識を計算機の知識ベースに変換し格納する枠組みを考察した[Schoen91]。

Devanbuは、あるドメインの大規模なソフトウ

ェアの隠れた構造を明らかにしていくプロセスのための知識表現と推論プロセスを考案した[Devanbu90]。

4. ドメインモデル

ドメインモデルは、個々の問題ドメインやアプリケーションドメインのモデル表現である。これは、個々のドメインの分析を行いモデリングを行うドメイン分析プロセスの出力結果である。

4.1 ドメインモデルの構成要素

以下をまとめて提示する文献は見あたらないが、特にソフトウェア部品の再利用のためのドメインモデルの構成要素として次の3つのものに着目する事が有効であると筆者らは考える。

(1)ドメイン言語:そのドメインで明らかにされたオブジェクト群や機能群、およびそれらの属性・振る舞いを関係づけて、ドメインの仕様を記述するために使われる言語。個々のドメインで、記述に用いる用語と文法が異なる。(例えば[Prieto-Diaz87])

(2)ライブラリ:仕様をプログラムコードに変換する時に用いる、ドメインに固有の標準的なライブラリ。(例えば[Simos91]のOrganon)

(3)ジェネレータ:言語で記述された問題ドメインをライブラリに適合させてアプリケーションドメインに自動あるいは半自動で変換する、ドメイン固有のジェネレータ。(例えば[Cleveland88])

例えば、航空機座席予約システムのドメインでは、用語として、座席、フライト、日付、空港、乗務員、フライトスケジュール、座席予約、乗務員割当等がある。ドメインモデルは、このドメインで許される全ての妥当なアクションを生成できる文法と意味をもった言語で記述される[Prieto-Diaz87]。

ビジネスアプリケーションに対するRaytheon社プロジェクトやミサイル用ソフトウェアに対するCAMPプロジェクトは、既存のプログラムを分析して上記のライブラリを得た事になる。

4.2 ドメインモデルの記述手段

ドメインモデルの記述手段は、対象とするドメインで、また同じ対象でも記述する側面で異なる。Arangoはドメインモデルが強調して記述する側面で、手段を次の通りに分類する[Tracz92]。

(1)宣言的/定義的モデル:Definitional Model

(構)成要素の抽出, 列挙, 分類を記述)…分類(taxonomy), クラス分け(class scheme), シソーラス(thesaurus)等

(2)構造的モデル: Structural Model(構成要素間の関係を明確にする)…概念グラフ(conceptual graph), ER図(E/R diagrams)等

(3)機能的モデル: Functional Model(システムや個々の構成要素の振る舞いを記述)…SADT, 状態遷移図(state transition diagrams), データフロー図(data flow diagrams)等

上記の分類には適合しないが, Bjornerは, リアルタイムシステムのドメインモデル記述手段に, VDMに基づく形式的仕様記述法を用いた[Bjorner92].

Raytheon社プロジェクトやCAMPプロジェクトは宣言的/定義的モデルに該当する。

杵嶋・伊藤は, 機能的モデルとして, リアクティブシステム向きに3つの部分図をもつ状態遷移図STD Triadを考案し, ドメインモデルを記述した[杵嶋93][Itoh94a][伊藤94].

4.3 ドメインモデルの種類

筆者らは次の種類のドメインモデルが存在すると思われる。カバーする範囲には重複がある。全てドメイン毎に存在するモデルである。

(a)分類モデル, (b)ひな形モデル, (c)パラメータ化モデル, (d)参照モデル, (e)標準モデル, (f)分析/設計/実装ハンドブック, (g)分析/設計/実装チェックリスト, (h)成功/失敗事例集

筆者らは, モデル化の対象によりドメインモデルを次の3つに分類する事が有効であると思われる。(1)解くべき問題群(問題ドメイン)を表すためのドメインモデル(Domain Problem Model), (2)解群(アプリケーションドメイン)を表すためのドメインモデル(Domain Product Model), (3)前者から後者を導出するプロセスを表すドメインモデル(Domain Process Model)

筆者らはこれをTriadic Domain Modelと呼ぶ[Itoh94ab][伊藤94]。ドメイン固有のプロセスモデルの認識は, 2節の[Schoen91]でのドメインモデルとタスクモデルをつなぐ問題解決モデルの認識と同様である。

5. ソフトウェア工学との関係

5.1 ドメイン工学

[Arango91]等の文献を総合すると, ドメイン工学は, 問題の明確化やソフトウェアの開発プロセスをドメインを意識して行う活動全般であり, 問題ドメインとアプリケーションドメインの間にある概念のギャップを埋める技術である。ドメイン工学では, (1)ドメイン分析, (2)インフラストラクチャの仕様化, (3)インフラストラクチャの実装, の3領域が扱われる。これは, ソフトウェア工学の(a)要求分析, (b)システムの仕様化, (c)設計と実装, に対応するが, 後者では, 個々のシステムの一品生産を念頭に置いている事に対して, ドメイン工学では, ドメインに属するシステムの集まりの認識とその再利用のためのインフラストラクチャの構成を念頭に置いている。

知識工学のドメインの専門家と知識エンジニアと同様に, ドメイン工学では, 問題ドメインの専門家とドメインアナリスト(ドメインエンジニア)の役割分担が重要である, と言われている。

5.2 要求工学との関係

ドメイン分析・モデリング技術はソフトウェアの再利用を補完するものとして生まれたが, ソフトウェア群を共通のドメインとして捉える観点で要求分析プロセスも効率化すると期待される。そこでの主張は以下の通りである[Ghezzi92].

・アプリケーションの要求仕様をドメイン毎に分類して再利用すべき。

・ドメイン知識は, 要求仕様を抽出したり要求と設計のギャップを埋める手助けになる。

Fickasは, 要求仕様をドメインモデルと照合して仕様デバッグを行う手法を提案した[Fickas88].

GreenSpan[Greenspan82]は, 実世界ドメインを考慮したRMP(Requirement Models Framework)を提案した。用語としてオブジェクト, 活動, 断定を定義し, それらに3種類の操作(クラス化, 集合化, 一般化)を施す。

Easterbrook[Easterbrook93]は, 例えば, 要求分析に参加する人達の, 社会的・組織的なビューを組み込んだ概念モデルを表すために, 要求工学でもドメインモデルを用いるべきであると述べた。ビューの階層的な記述を導入し, 相反したり矛盾するビューを1つのドメインモデルの中に組み込む。

Kramer[Kramer93]は, (1)ドメインモデルを単なる抽象化として捉えるのではなく, システムを仕様化・開発した経験からドメインモデルを抽出しなければならない事, (2)システムの仕様化の

経験をドメインエンジニアに正しく伝達する仕組みを作らなければならない事、を挙げた。

Barstow[Barstow93]は、ドメインモデルに基づいて個々のシステムを開発する利点は、(1)システムを構成する物理的/計算的な現象の要素の概念、属性、関係をドメインモデルとして明確化するので、個々のシステムの開発が容易になる、(2)システムのカスタム化や進化をドメインモデル上で把握し易い、を挙げた。

Jackson[Jackson93]は、要求仕様はドメインを明確に記述すべきであり、個々のシステムに共通なドメインの性質と、個々のインスタンスの性質を明確に区別すべきであると、述べた。

5.3 設計との関係

Arangoはあるドメインの分析と設計の統合を図るプロセスをドメイン分析ととらえて、そのドメインの技術ブックを得る方法を考案した[Arango93]。このブックはドメインの技術・設計情報から成るドメインモデルとみなす事ができ、このブックのインスタンスとして、個々のソフトウェアの開発を行い分析や設計の再利用を行うととらえられる。これはハンドブック形式のドメインモデルの設計への適用の一例である。

Cohenは、ドメイン工学とアプリケーション工学から成るモデルベースソフトウェア工学(MBSE)を提唱した。前者で得られた情報モデルと特徴モデルから成るドメインモデルをベースに、後者で要求分析し、前者で得られた設計モデルをベースとして後者で設計を行う。その後、アプリケーションジェネレータでライブラリ群を統合する[Cohen94]。ひな形形式のドメインモデルの設計への適用の一例である。

5.4 リバースエンジニアリングとの関係

実装済みのソフトウェアから設計の仕様を抽出して、類似のソフトウェア開発時にそれらの情報を利用するリバースエンジニアリングも、ソフトウェア再利用の一種である。このリバースエンジニアリングも、ドメイン分析・モデリングの手法により更に効果的なものになる。リバースエンジニアリングで得られる情報は、設計時と異なり、変更、バグフィックス、バージョンアップ等で変更/追加された情報も含むため、以前の仕様を推定する事は必ずしも容易ではない。対象のソフトウェアが属するドメインを意識する事により、それに属するソフトウェアの機能や構造を特定しや

すくなると、Harrisは主張する[Tracz92]。

5.5 その他のフェーズとの関係

ドメインモデルを保守、教育、リスク分析等にも利用可能であるとする意見もある[Tracz92]。

5.6 ドメイン指向ソフトウェアアーキテクチャ

ソフトウェアアーキテクチャ(例えば[Garlan93b])とは、ソフトウェアのもつ機能、構造、マンマシンインタフェース等の、比較的抽象的な構成要素、相互の関係、属性等を総称したものである。通常、ソフトウェアアーキテクチャには、ウォーターフォールモデルでの上流のフェーズ(要求分析や概要設計)で決定される項目が含まれる。Traczが提唱するドメイン指向ソフトウェアアーキテクチャ(DSSA)は、各ドメインに固有のソフトウェアアーキテクチャを指す[Tracz93]。

BarstowはDSSAで給油所とコンビニの販売管理の事例を考察した[Barstow94]。DSSAである参照モデルを構成した後の個々のカスタム化の方式も検討した。

5.7 複数記述手段の活用

筆者らは、同一のシステムに対しても複数の側面を記述する手段が必要であると考え[伊藤93]。例えば、本稿の4.2節で述べた記述手段には、構造等の静的な側面の記述に優れたもの、状態遷移等の動的な側面の記述に優れたものがある。

オブジェクト指向のシステム開発技法では、要求分析から設計・実装に至るまで、一貫して対象システムを複数オブジェクトとそれらの相互作用としてモデル化する。ドメイン分析・モデリングの観点からは、ソフトウェア開発の全工程に統一したモデルを適用する事で、工程間のセマンティクスギャップを最小化できる。例えばJohnsonは、オブジェクト指向で再利用が促進される、と主張する[Johnson88]。また、Batoryは、オブジェクト指向で階層的にシステムをとらえる、GenVocaと呼ぶドメインモデリングの方式を提案した[Batory92]。de Cimaはオブジェクト指向設計でのドメインモデリングの方式を提案した[de Cima94]。

オブジェクト指向に限らず1つの手段のみで対象システムをモデル化しようとする見方が一面的になり、ドメインモデルがもつ様々な側面を記述し損なう可能性もある事に留意したい。

6. 今後の課題

ソフトウェア工学で注目されているドメイン分析・モデリング技術を概観した。ドメイン分析・モデリング技術は、元々はソフトウェアの再利用を促進する手段として注目された。そこでは、既存のソフトウェア群をモデル化し、それを新システムの開発時に再利用しようとする。本稿で述べたように、ドメイン分析・モデリングは、ソフトウェア部品の再利用だけではなく、要求工学等の他の開発フェーズでの役割も注目されている。

この技術は、従来の開発作業をそのままCASEとして実現するために分析するものではない。複数のソフトウェアをドメインという枠組みでとらえ、得られたドメインモデルを基礎として、ソフトウェアの再利用を促進し、あるいは開発の方法論や手法を構成する。ドメインモデルに基づくシステム開発は、実際の開発作業からは遊離せず、システム開発での様々な問題をドメインという枠組みで見直す。これにより、実際の開発作業が効率的になり、システム製品の品質が向上すると期待できる。

筆者らは対象ドメイン毎に固有の用語、概念、要求の獲得方法、分析・仕様化の方法や手順等があると考える。このため、ドメイン分析・モデリングの研究では、今後次の研究を行う必要があると考える。

- ・対象ドメイン毎の分析手順
- ・対象ドメイン毎のドメインモデルの集積
- ・ドメインモデルに基づいた対象ドメイン向けの再利用部品群
- ・ドメインモデルに基づいた、一般的な開発手法の特定のドメイン向け特化方法
- ・ドメインモデルに基づいた個々のドメインに固有な開発手法、ツール

参考文献

- [Arango89]Arango, G.: Domain Analysis - From Art Form to Engineering Discipline, 5th IWSSD, pp.152-159, ('89).
- [Arango91]Arango, G., R. Prieto-Diaz: Introduction and Overview: Domain Analysis Concepts and Research Directions, pp.9-26, Domain Analysis and Software Systems Modeling (R. Prieto-Diaz and G. Arango ed.), IEEE, ('91).
- [Arango92]Arango, G., B. I. Blum: Applications of Domain Modeling to Software Construction EDITORIAL, Int. J. SE&KE, Vol. 2, No. 3, pp. 321-323, ('92).
- [Arango93]G. Arango, E. Schoen, R. Pettengill: Design as Evolution and Reuse, IEEE IWSR'93, pp.9-18, ('93).
- [Barstow93]D. R. Barstow: Should We Specify Systems or Domain, IEEE RE'93, pp. 79, (Jan., '93).
- [Barstow94]D. Barstow, N. Erol: Domain-Specific Architectures and Reuse through Customization: A Case Study, ICSR'94, Invited Paper, (Nov., '94).
- [Batory92]D. Batory, V. Singhal, M. Sirkin: Implementing a Domain Model for Data Structures, Int. J. SE&KE, Vol. 2, No. 3, pp. 375-402, (Sept., '92).
- [Bjorner92]D. Bjorner: Trusted Computing Systems: The ProCos Experience, ICSE'92, pp. 15-34 (May, '92).
- [Cleaveland88]J. C. Cleaveland.: Building Application Generators, IEEE Softw., Vol. 5, No. 4, pp. 25-33, ('88).
- [Cohen94]S. Cohen: Feature-Oriented Domain Analysis: Domain Modeling, ICSR'94, Tutorial, (Nov., '94).
- [de Cima94]A. M. de Cima, C. M. L. Werner, A. A. C. Cerqueira: The Design of Object-Oriented Software with Domain Architecture Reuse, ICSR'94, pp. 178-187, (Nov., '94).
- [Devanbu90]P. Devanbu, R. J. Brachman, P. G. Selfridge, B. W. Ballard: LaSSIE: A Knowledge-Based Software Information System, ICSE'90, ('90).
- [Easterbrook93]S. Easterbrook: Domain Modelling with Hierarchies of Alternative Viewpoints, IEEE RE'93, pp. 65-72, (Jan., '93).
- [Fickas88]S. Fickas, P. Nagarajan: Critiquing Software Specifications, IEEE Software, Vol. 5, No. 6, pp. 37-47, (Nov., '88).
- [Frakes92]B. Frakes, R. Prieto-Diaz: Introduction to Software Reuse, Video Course, Software Engineering Guild and Reuse Inc., ('92)
- [Fraser93]S. D. Fraser, C. S. Saunders: Enhanced Reuse with Group Decision Support Systems, IEEE 2nd IWSR, pp. 168-175, ('93).
- [Garlan93a]D. Garlan: Domain Specifications

- Require First Class Connectors, IEEE RE' 93, pp. 78, ('93).
- [Garlan93b]D. Garlan, Mary Shaw: Architectures for Software Systems, Tutorial on ACM SigSoft' 93, (Dec. 93).
- [Ghezzi92]C. Ghezzi, G. C. Roman: Succceedings of 6th Int. Workshop on Softw. Specification and Design, ACM SIGSOFT Softw. Engi. Notes, Vol. 17, No. 1, pp. 44-51, (January, '92).
- [Greenspan82]S. J. Greenspan, J. Mylopoulos, A. Borgida: Capturing More World Knowledge in the Requirements Specification, 6th ICSE, pp. 25-234, ('82).
- [伊藤93]伊藤潔, 杵嶋修三, リアルタイムシステムにおけるネット指向開発技術の適用, 情報処理, Vol. 34, No. 6, pp. 747-760 (June '93).
- [Itoh94a]K. Itoh, S. Kishima, Y. Tamura: Systems Integration on Specification, Design and Generation of Reactive System - Triadic Domain Model-Based Approach -, IEEE ICSI' 94, (August, '94).
- [Itoh94b]K. Itoh, Y. Tamura, S. Kishima: Triadic Domain Model-Based Development of Software Systems, IEEE ICSR' 94, (Nov., '94).
- [伊藤94]伊藤潔, 田村恭久, 杵嶋修三: Triadic Domain Modelに基づくシステムの分析・設計, FOSE' 94, 日本ソフトウェア科学会, (Dec., '93).
- [Jackson93]M. Jackson, P. Zave: Domain Description, IEEE RE' 93, pp. 56-64, (Jan., '93).
- [Kramer93]J. Kramer: Generalizations are False, IEEE RE' 93, pp. 80, (Jan., '93).
- [Johnson88]R. E. Johnson, B. Foote: Designing Reusable Classes, J. of Object-Oriented Programming, Vol. 1, No. 2, pp. 22-35, (June /July, '88).
- [Kang90]K. Kang, S. Cohen, J. Hess, W. Novak, A. S. Peterson: Feature-Oriented Domain Analysis (FODA) Feasibility Study, Carnegie Mellon Univ. -Softw. Engi. Institute, CMU/SEI-90-TR-21, ('90).
- [杵嶋93]杵嶋修三, 伊藤潔, リアクティブシステムの分析・設計向きドメインモデル: Asdreas STD Triad, 情報処理学会論文誌, Vol. 34, No. 9, pp. 2025-2036 (Sept., '93).
- [Lubars91]M. D. Lubars: Domain Analysis and Domain Engineering in IDeA, Domain Analysis and Software Systems Modeling (R. Prieto-Diaz and G. Arango cd.), IEEE, ('91).
- [Neighbors81]J. M. Neighbors: Software Constructing Using Components, Ph.D. Thesis, Univ. Calif. Irvine ('81).
- [Neighbors89]J. M. Neighbors: DRACO: A Method for Engineering Reusable Software Systems, Softw. Reusability, Vol. 1, ACM, pp. 295-320, ('89).
- [Prieto-Diaz87]R. Prieto-Diaz: Domain Analysis for Reusability, COMPSAC' 87, pp. 23-29, ('87).
- [Prieto-Diaz90]R. Prieto-Diaz: Domain Analysis: An Introduction, ACM Sigsoft Softw. Engi. Notes, Vol. 15, No. 2, pp. 47-54, ('90).
- [Prieto-Diaz91]R. Prieto-Diaz: Implementeing Faceted Classification for Software Reuse, CACM, Vol. 34, No. 5, pp. 88-97, (May, '91).
- [Schoen91]E. Schoen: Active assistance for domain modeling, 6th IEEE Annual Knowledge-Based Softw. Engi. Conf., pp. 26-35, ('91).
- [Simos91]M. A. Simos: The Growing of an Organon: A Hybrid Knowledge-Based Technology and Methodology for Software Reuse, Domain Analysis and Software Systems Modeling (R. Prieto-Diaz and G. Arango cd.), pp. 204-223, IEEE, ('91).
- [田村94]田村恭久, 伊藤潔, 杵嶋修三: ドメイン分析・モデリング技術の現状と課題, 情報処理, ('94).
- [Tracz92]W. Tracz: Domain Analysis Working Group Report-First Int. Workshop on Software Reusability, ACM SIGSOFT SE. Notes, Vol. 17, No. 3, pp. 27-34, ('92).
- [Tracz93]W. Tracz, L. Coglianese, P. Young: A Domain-Specific Software Architecture Engineering Process Outline, ACM SIGSOFT Softw. Engi. Notes, Vol. 18, No. 2, pp. 40-49, ('93).
- [Tracz94]W. Tracz, J. Poukin: A Domain Specific Software Architecture Tutorial, 3rd ICSR, ('94).