2L-08

ドメイン駆動設計における制約条件記述を含むドメインモデルの準形式化手法の提案

A Proposal for Semi-Formalizing Domain Models, Including Constraint Descriptions, in Domain-Driven Design

水野 駆 †　　**和崎 克己** ††
Kakeru Mizuno　Katsumi Wasaki

1　はじめに

ソフトウェア工学において，様々なアプローチによるリファクタリングの手法が提案されている [1]．ソフトウェアの品質を高めるうえで，ドメイン駆動設計は近年注目されている開発手法の一つである．優れたドメインモデルを作成することは，非常に価値のあることであるが，容易に作成できるものではない．現在ではドメインモデル抽出法は多種多様な状況にあり，準形式的な記述も含めてドメイン駆動設計を成功させるためのフレームワーク造りが重要であると考える．本研究では，ドメインモデルのエンティティや値オブジェクト等に対する制約条件記述について着目し，ツール支援も含めた検討を行っている．

2　ドメイン駆動設計とは

ドメイン駆動設計（Domain-Driven Design，略称: DDD）は，ソフトウェア開発における設計手法の一つであり，エリックエヴァンス氏によって提唱された．ソフトウェアシステムを「ある領域や業務に関連する情報やプロセス」のことであるビジネスドメインに焦点を当て，そのドメインの情報をモデル化した「ドメインモデル」を育てていく手法のことである [2]．

2.1　ドメインモデル

ドメインモデルとは，特定の図のことではなく図が伝えようとしている考え方のことである．ドメインに対しての知識が厳密に構成され，選び抜かれて抽象化されたものである [3]．制作物としては，図や自然言語のドキュメントを作成する．

2.2　UML を用いた表現例

DDD において，ドメインモデルを抽出する際に，よく使われる UML（Unified Modeling Language）図がドメインモデル図とユースケース図である．ドメインモデル図はクラス図を簡易的にして，各クラス（エンティティ）に対して吹き出しを付け，その中に制約条件を記述するスタイルが一般的である．UML 図は自由度が高すぎるため，DDD で上位記述した成果を他者とレビューしたり，下流工程での実装の半自動コード生成等にそのまま利用するすることは，困難さが生じる．

† 信州大学大学院総合理工学研究科, Graduate School of Science and Technology, Shinshu University
†† 信州大学工学部電子情報システム工学科, Department of Electrical and Computer Engineering, Faculty of Engineering, Shinshu University

3　ドメインモデルの準形式化に対するアプローチ

3.1　PlantUML について

PlantUML（Plant Unified Modeling Language）[4] は，テキストベースの図示言語およびツールで，構造化されたデータやプロセスのモデリングに使用される．PlantUML は UML 図などを生成するために設計されている．DDD において，ドメイン仕様の変更は頻繁に起こりうる．ドメインモデル図やユースケース図等を作成する際に図として管理するとバージョンを管理するのが困難である．PlantUML ではテキストベースで記述していくので，バージョン管理システムでコードと一緒に管理することができる．また，テキストベースであるので準形式化へ取り組みやすいと考え，本ツールを採択した．

3.2　具体的な予約システムを対象とした PlantUML 記述

具体的にホテル宿泊予約システムに対して制約を考えたのが以下の通りである．

1. 重複する利用日に同じ部屋に対するアクティブな予約は作成できない．
2. 当日予約はできない．
3. 顧客の連絡先は有効な電話番号形式でなければならない．

図 2 はホテル宿泊予約システムのドメインモデル図作成のための PlantUML 記述の例を一部抜粋したものである．ドメインモデルに対する制約は一つにまとめて JSON 形式で記述を行うようにした．constraints と紐づく配列に制約を書き込むように設定した．上記の自然文で書かれた制約が content に内包していることが分かる．また，relation にはこの制約がどのエンティティもしくは値オブジェクトに結びつくかを指定している．meta には図には表示させない別途記述しておくべきことを記述している．

3.3　ドメインモデル図の自動生成

図 2 の PlantUML 記述から自動生成したものが，図 1 のドメインモデル図である．relation で記述したエンティティ（値オブジェクト）に対して各制約が関連づいていることが分かる．独自で記述した createConstraintsFromJson 関数によって，JSON 型を強制できるので図に表現したくない制約としたほうがいい制約の取捨選択をできるようにするなどの拡張が見込まれる．

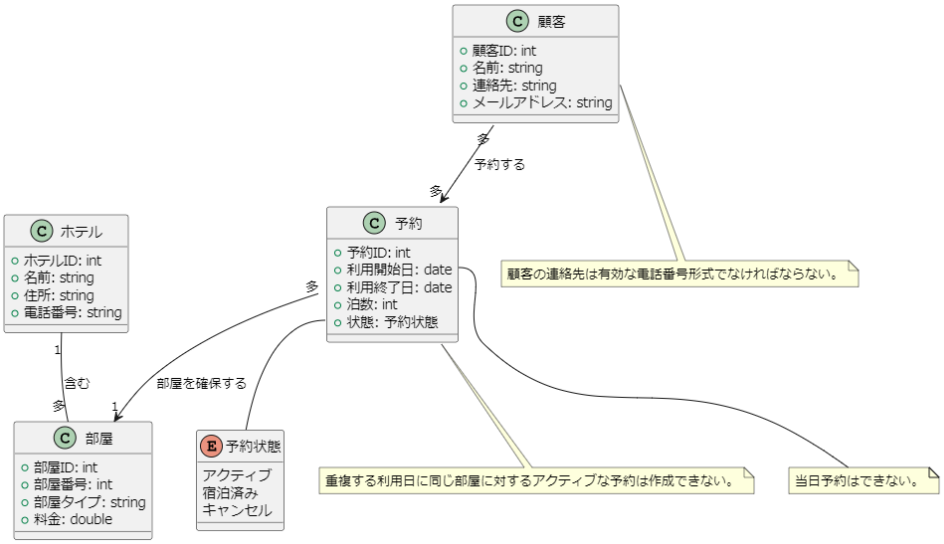

図 1　ホテル宿泊予約システムのドメインモデル図

```
class 予約 {
  + 予約ID: int
  + 利用開始日: date
  + 利用終了日: date
  + 泊数: int
  + 状態: 予約状態
}

enum 予約状態 {
  アクティブ
  宿泊済み
  キャンセル
}

!$c = {
  "constraints": [
    {
      "id": "C1",
      "content": "重複する利用日に同じ部屋に対するアクティブな予約は作成できない。",
      "relation": "予約",
      "meta": "重複する利用日に同じ部屋に対する予約がどのような状態か検証する必要"
    },
    {
      "id": "C2",
      "content": "当日予約はできない。",
      "relation": "予約::利用開始日",
      "meta": "予約の利用開始日は予約作成日以降でなければならない。"
    },
    {
      "id": "C3",
      "content": "顧客の連絡先は有効な電話番号形式でなければならない。",
      "relation": "顧客::連絡先",
      "meta": "電話番号形式は国や地域によって異なるため、国や地域ごとに検証が必要"
    }
  ]
}
$createConstraintsFromJson($c)

ホテル "1" -- "多" 部屋 : 含む
予約 "多" --> "1" 部屋 : 部屋を確保する
予約::状態 -- 予約状態
顧客 "多" --> "多" 予約 : 予約する
```

図 2　ホテル宿泊予約システムの PlantUML 記述

4　ドメインモデルの準形式化の応用

準形式化を行うことで，一つのモデルに対して一つの JSON 記述の制約条件をドメインモデル図に反映させることができた．ここからの応用として 2 つ考えていることがある．

4.1　制約条件の自動改善

ChatGPT を利用して，ドメインモデルの制約条件の吟味・提案を自動で行うシステムである．DDD ではドメインモデルをさまざまな人とのコミュニケーションを通して育てていくことが重要である．ChatGPT であれば，一般的なビジネスロジックを知っているので，その一端を担えるのではないかと考えている．本研究では，ChatGPT の API を使用しプロンプトエンジニアリングをしながら，制約条件の改善を提案してくれるようなシステムを考えている．

4.2　ドメインモデルの検証

従来の DDD では，ドメインモデルを作成したらすぐに実装に移るのが一般的である．そこで実装に取り掛かる前にドメインモデルの整合性を確認し，ドメインモデルの曖昧性を排除することで，さらに良いドメインモデルの作成に役立つのではないかと考える．本研究では，モデル規範型形式手法である VDM との連携を考えている．形式仕様記述言語 VDM++ を採択することで，オブジェクト指向性を持たせたドメインモデルの記述を行うことで実装に近い形で検証することができる．テキストベースでドメインモデル図が記述されるので，そこから VDM++ の記述で生成・検証を行えるのではないかという試みである．

5　まとめと今後の課題

本研究でドメインモデルの準形式化手法の提案とそれに対する応用を考えた．本稿の時点では，ドメインモデルの準形式化の応用については構想段階である．今後は，ドメインモデルの準形式化を進めるとともに，順次ドメインモデルの抽出・練り上げの部分をサポートするツール・手法の開発に取り組むつもりである．

参考文献

[1] Misbhauddin, M. and Alshayeb, M., An integrated metamodel-based approach to software model refactoring, Software and Systems Modeling, 18, pp.2013-2050, 2019
[2] 成瀬允宣, ドメイン駆動設計入門, 株式会社翔泳社
[3] エリック・エヴァンス（著）, エリック・エヴァンスのドメイン駆動設計, 株式会社翔泳社
[4] PlantUML, https://plantuml.com/ja/