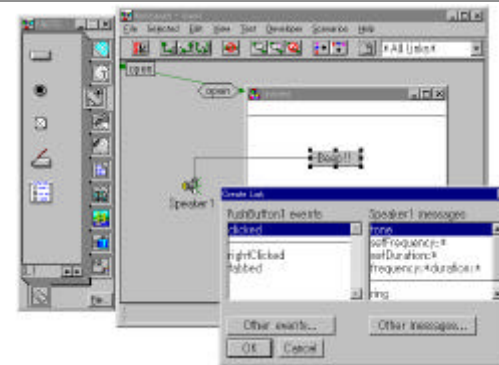


---

## ユースケースを用いる要求分析

2001年1月12日  
S Lagoon  
中谷多哉子

## コンポーネントのデモ



---

## 目次

- UML概要
- ソフトウェア開発にユースケースを適用する意義
  - Webアプリケーション
  - コンポーネントに基づく開発
- アクティビティ図の適用
- ユースケースの適用
- ユースケース構造による生産性向上
- ユースケース駆動型によるモデル化

---

## 表記法標準化に向けて

- ソフトウェア構築前にモデル化を行うための支援
- 開発者間の対話を円滑化
- Standard Modeling Languageへの中間点 (OMG)
- 標準方法論, 標準開発プロセスは提供しない

## 標準化への歴史（方法論乱立時代）

- Shlaer/Mellor (1989, 1991)
- Beck/Cunningham: Class – Responsibility - Collaboration: CRC (1989)
- Wirfs-Brock et al.: Responsibility-Driven Design (1990)
- Coad/Yourdon: OOA (1991)
- Rumbaugh et al.: OMT (1991)
- Coad/Nicola: OOD (1993)
- Booch: Booch Method (1994, 1995)
- Jacobson: Objectory (1987-1995)
- Odell/Martin (1994, 1996)
- Coad et al.: OOP (1995)
- >>>>>UML<<<<<<<<<<
- D'Souza/Willis: CATALYSIS (1998)
- Jacobson, Booch, Rumbaugh: Rational Unified Process (1998-)

## 標準化への歴史（標準化へ）

- 1994 Boochが標準化呼びかけ . Rumbaughが移籍
  - Booch/Rumbaugh 方法論勝利宣言
- 1995 Unified Method ver.0.8発表
  - Jacobson移籍, Objectory版權Rationalへ
- 1996 OMGにてOdell, Loomiseが方法論標準化活動
- 1997 OMG, UMLの意味論, UML ver.1.0を検討.
  - UML ver.1.1発表: <http://www.rational.co.jp>から入手可能
  - 現在 ver.1.3

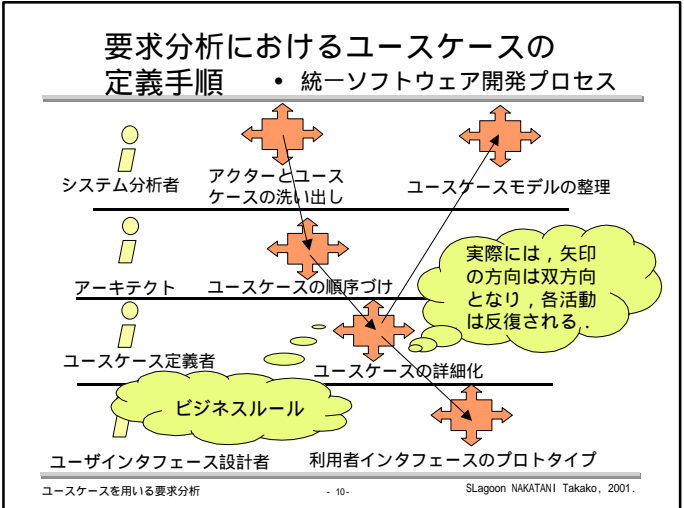
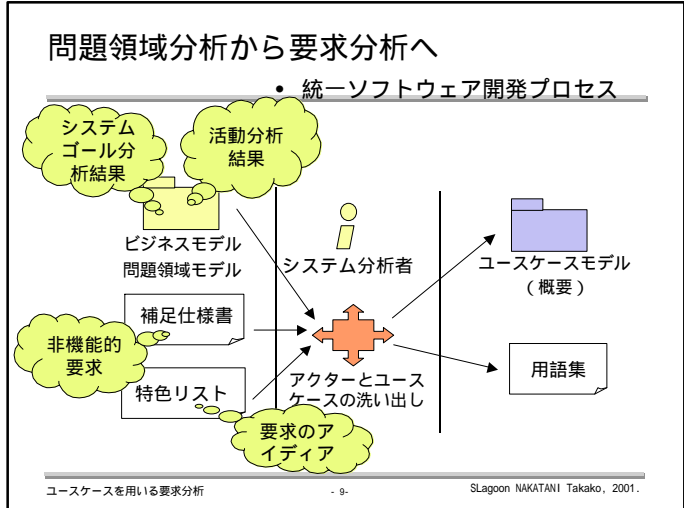
## UML(Unified Modeling Language)

- 静的モデル
  - クラス図
  - パッケージ図
  - ユースケース図
  - コンポーネント図
  - 配置図
- 動的モデル
  - 相互作用図 (シーケンス図, コラボレーション図)
  - ステートチャート図
- その他の表記
  - 実装図

これらとは別に自然言語  
で表記するユースケース  
が定義されている

## UMLの方針

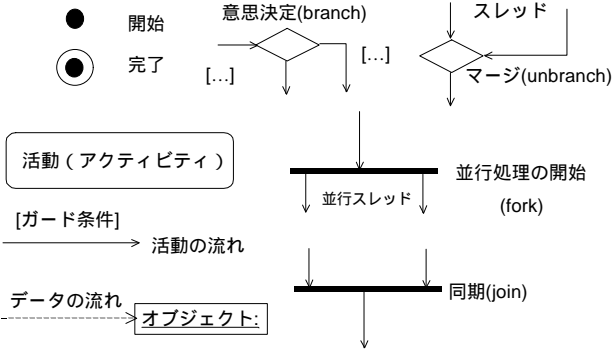
- 表記の意味論を定義
- オブジェクト制約言語 (OCL) の導入
- 方法論には言及しない
  - 新しい方法論が提案され始めた
  - ユースケースの使い方についても言及しない
  - ユースケースの詳細な記述方法や構造についても定義しない



- ### Webアプリケーションの特徴
- ・ 電子商取引の場合
    - 非対面式である
    - リアルタイムの問い合わせは不可
      - ・ 銀行のATM仕様はOK?
    - 直感的に使えるインターフェースが必要
      - ・ 顧客囲い込み手段のひとつ
      - ・ 他社よりも使いやすく
- ユーザーケースを用いる要求分析
- 11 - SLagoon NAKATANI Takako, 2001.

- ### 使いやすさを追求するには
- ・ 仮想ユーザの活動分析が必要
    - アクティビティ図
  - ・ 仮想ユーザの抽出
    - ユーケース図
  - ・ 仮想ユーザとWebアプリケーションとの対話分析が必要
    - ユーケース
    - 利用者インターフェースのプロトタイプ
- ユーザーケースを用いる要求分析
- 12 - SLagoon NAKATANI Takako, 2001.

## アクティビティ図の紹介 (UML)

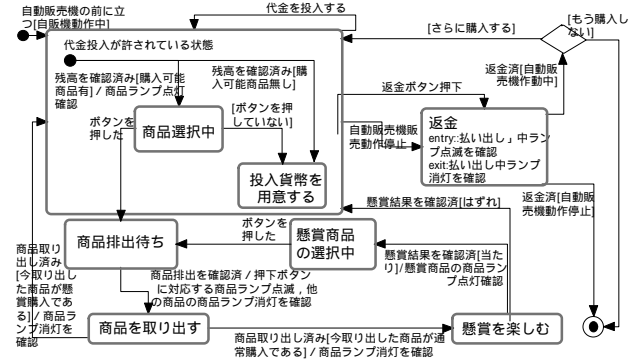


ユースケースを用いる要求分析

- 13 -

SLagoon NAKATANI Takako, 2001.

## アクティビティ図の例



ユースケースを用いる要求分析

- 14 -

SLagoon NAKATANI Takako, 2001.

www.askul.co.jp



ユースケースを用いる要求分析

- 15 -

SLagoon NAKATANI Takako, 2001.

www.rakuten.co.jp

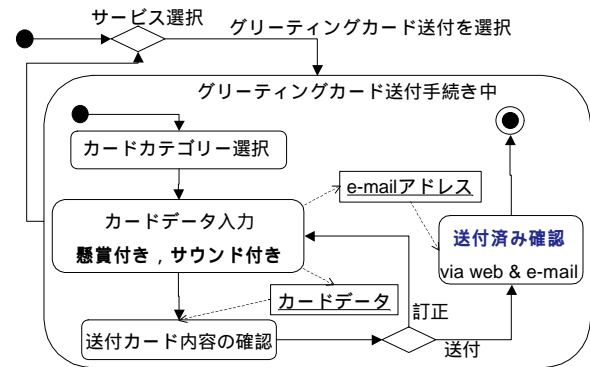


ユースケースを用いる要求分析

- 16 -

SLagoon NAKATANI Takako, 2001.

## アクティビティ分析：“楽天”年賀状送付



## 活動分析はなぜ必要か

- システム利用者の広範な活動を分析する
  - 開発するシステムの境界を決定できる
- システムが活動を支援する必要性を明らかにできる
  - ユースケース抽出の手段である
- 他社のシステムとの差異を明確に把握できる
- 利用者のニーズの変化を察知する手段である

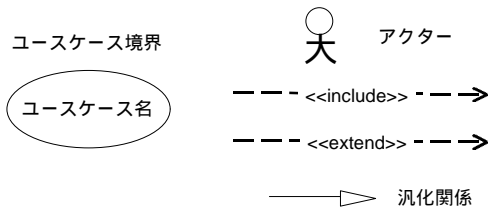
www.kuronekoyamato.co.jp



## アクティビティ図とユースケース

- 最大粒度のユースケースについて
  - 1 アクティビティ 1 ユースケース
  - 1 アクティビティの事前/事後条件
    - 1 ユースケースの事前/事後条件
- ユースケースの詳細化
  - アクティビティ図の再検討
  - 理想世界の利用者のアクティビティから、現実世界のアクティビティへと変化する過程

## ユースケース図 (UML)

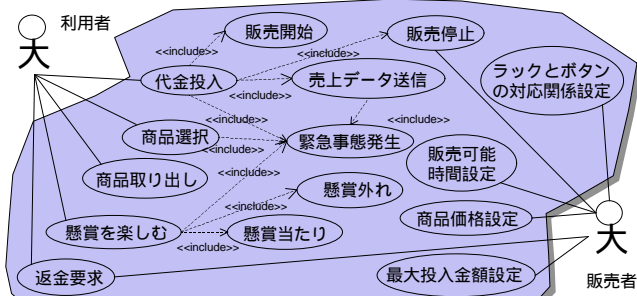


## 抽象 / 具象 / 拡張 / 実ユースケース

- 抽象ユースケース：複数のユースケースで共有される再利用可能な振る舞いを記述する
- 具象ユースケース：他の具象ユースケースと共有する振る舞いは記述しない
- 拡張ユースケース：他のユースケースに追加する振る舞いを記述する
- 実ユースケース：要求から抽出したユースケース
  - 具象ユースケース, 抽象ユースケース, 拡張ユースケースを汎化, 拡張関係, 包含関係で結合して定義する

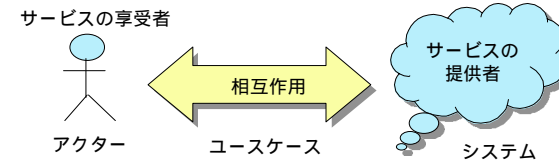
## ユースケース図の例

- 要求の妥当性を確認する対象者を一覧できる

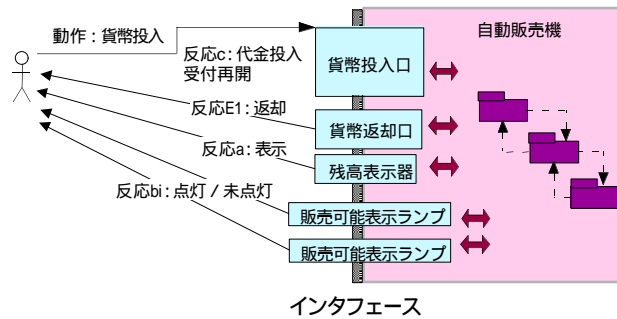


## ユースケース

- アクターと呼ばれるシステムからサービスを受けるオブジェクトと、サービスを提供するシステムとの間の相互作用
- 自然言語で記述した要求仕様の記述方法



## 例：ユーザとシステムとの相互作用



## ユースケースの分類

- ユースケースの詳細化レベル
  - 概要ユースケース
  - 主要タスクユースケース
  - 従属機能ユースケース
- ユースケースの優先順位
  - 主要ユースケース
  - 従属ユースケース
  - 選択ユースケース

## 概要ユースケースの例

- 利用者がコイン、紙幣をシステムに投入するとシステムは、投入された金額に応じた販売可能な商品を利用者に知らせる。
- 利用者がシステムに返金要求をすると、システムは、残金を返金する。
- 利用者が購入したい商品をシステムに知らせると、システムは代金を受け取り、商品を利用者に引き渡す。

## 主要ユースケースの例

- 代金投入ユースケース
- アクタ: 代金投入する購入者
- 目的: システムが正当な貨幣を受け付けて、受け付けた貨幣の合計金額によって商品を販売できるようにする。
- 事前条件: 自動販売機作動中、商品送出し中でない、および商品取出し済み、懸賞中でない、返金中でない
  - 注) これらの条件は、代金投入可能であることを表わす。
- 事後条件: 販売可能な商品の商品ランプ点灯

## 続：主要ユースケースの例

- 基本系列
  - 1.アクターは紙幣投入口、硬貨投入口、プリペードカード読み取り機に手で貨幣を投入する。
  - 2.システムは残高表示器に表示していた額に、新たに投入された金額を合計して表示する。
  - 3.システムは投入された額に応じて購買可能な商品の商品ランプを点灯する。

## 続々：主要ユースケースの例

- 例：代替系列
  - 1.基本系列2において、硬貨および紙幣が投入されて、合計金額が最大投入金額を超えた場合は、新たに投入された貨幣をそれぞれの返金口から返却する。
  - 2.基本系列2において、千円札と認識できた紙幣以外の不良貨幣が投入されたときは、それらを投入口から返却する。
  - 3.基本系列2において、500円、100円、50円、10円硬貨と認識されなかった異物は釣銭取り出し口から返却する。
  - 4.基本系列2において、有効期限切れ、残高のないプリペードカードは投入口から返却する。
  - 5.このユースケースを実行中、アクターが商品を選択しなくなるときは返金ユースケースを起動する...

## ユースケースの基本構造

- ユースケース名
- ゴール、目的
- アクター
- 事前条件
- 事後条件
- 基本系列
- 代替系列

## 品質向上のための構造の検討

- ユースケースの名前と管理用ID
- 概要(Summary)
- アクター(Actors)
- 前提条件(Preconditions)
- 処理(Descriptions)
- 例外事項(Exceptions)
  - 入力の誤り、入力データ間の矛盾、
  - オブジェクトの属性値自体の問題、属性値間の矛盾、
  - 入力データと属性値間の矛盾に起因する事項
- 後条件(Postconditions)

富士通:吉田ら



## ユースケーステンプレート

- ユースケース名：
- ゴール：
- スコープ：システム設計で考慮するブラックボックスの範囲
  - システム，サブシステム，コンポーネントなど
- レベル：概要，主要タスク，従属関数
- 事前条件：
- 成功時の事後条件：
- 例外終了時の事後条件：

Cockburn

## ユースケーステンプレート（続）

- プライマリアクター：役割名と説明
- トリガー：ユースケースを起動させる動作
- 基本系列（Main Success Scenario）
  - <系列番号><動作の説明>
- 代替系列（Extensions）
  - <例外事象が発生する系列番号><条件>:<動作，起動される他のユースケース>
- 類似ユースケースの多様性項目
  - <識別子><多様性項目のリスト>

## 多様性項目の例

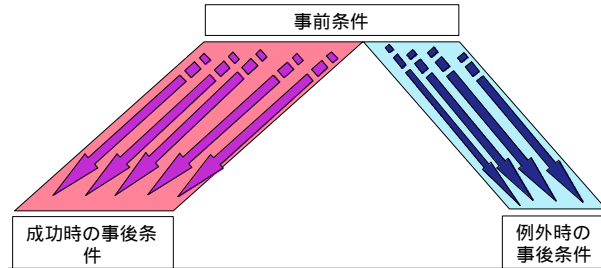
- （自動販売機）投入可能な貨幣は以下のとおり
  - 硬貨，1000円札，プリペイドカード
- この項目があると，各貨幣の多様性に対して別のユースケースを定義する必要がなくなると同時に，多様性に変化がおきたときには，この部分を変更すれば済む．
- 多様性項目は，シナリオでは分岐動作に対応する．
- ユースケースの数の爆発を回避する手段である．

## ユースケーステンプレート（続々）

- 関連情報：関係するユースケース名，このユースケースを呼び出す／呼び出されるユースケース
- 優先順位：システムにおける当該ユースケースの重要度
- 性能：ユースケース実行時間の制約
- 頻度：起動される頻度
- アクターとの対話経路：対話型，静的ファイル，データベース．．
- 備考：検討事項など
- 発行年月日
- 管理情報：

## トラウザモデル

- ユースケースからシナリオ（起こりうる相互作用の一つの流れ）を抽出する



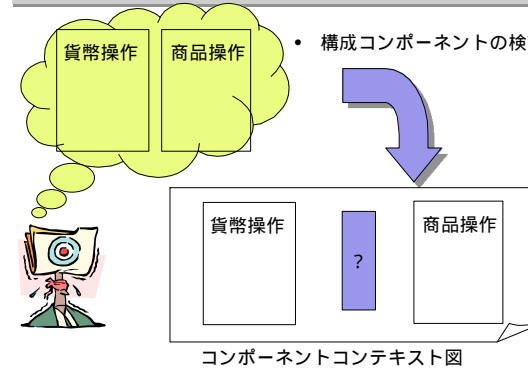
ユースケースを用いる要求分析

- 37 -

SLagoon NAKATANI Takako, 2001.

## 振る舞いファブリック

- 構成コンポーネントの検討手段



ユースケースを用いる要求分析

- 38 -

SLagoon NAKATANI Takako, 2001.

## ユースケースマップ

- 振る舞いファブリック（織物：糸の相互作用と布の柔軟性と具体性）を導入
- 抽象度が高く、ひとつのユースケースマップから様々な組み合わせのコンポーネントを抽出 / 検討できる
  - より適切なコンポーネント抽出の議論が可能
  - UMLの相互作用図は、コンポーネントの分割（アーキテクチャ）まで踏み込みすぎ、抽象度の高い議論をしづらい。

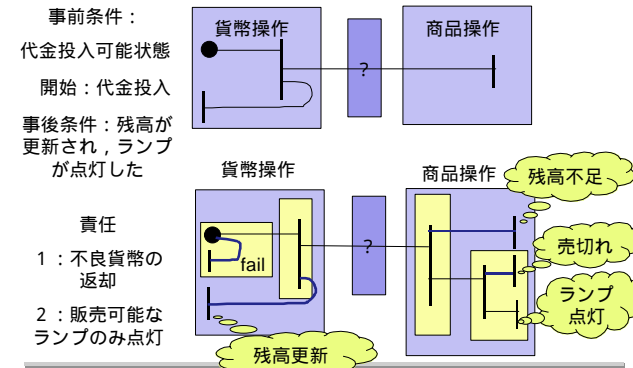
Buhr and Casselman

ユースケースを用いる要求分析

- 39 -

SLagoon NAKATANI Takako, 2001.

## 振る舞いパターンの検討



ユースケースを用いる要求分析

- 40 -

SLagoon NAKATANI Takako, 2001.

## ユースケースフレームワーク構築の試み

- ドメイン固有の留意点を規定の構造として取り込みたい
- より厳密な要求定義の枠組み
- より適切な表現, 品質の安定
- ユースケース間で共有されるデータの存在が見えない
- ユースケース記述の生産性を向上させたい
- 定型的な処理の流れによるソフトウェアの使い勝手を向上させたい
  - 再利用可能なユースケースを抽出 / 管理できない
  - ユーザインタフェースとユースケースの関係を管理できない

中谷, 浦井他

## ユースケースフレームワークの検討

- ユースケース名 : **商品登録**ユースケース
- アクター : **商品登録希望者**
- 目的 : **商品**を登録する
- 事前条件 : アクターが**商品登録に必要な情報**を得ている
- 正常終了における事後条件 :
  - **商品**が登録済みとなったという結果をアクターが得ている
- 例外終了における事後条件 :
  - **商品**の登録が中止され, 副作用が残っていない

## ユースケースフレームワークの例

- ユースケースのドメイン依存の定型的な枠組みを提供
  - データの登録 / 検索 / 更新
  - 一覧表作成
  - 書類作成 / 書類送付
- 基本ユースケース
  - 単一の目的を達成するために定義されるユースケース
- 複合ユースケース
  - 基本ユースケースを利用して生成

抽象ユースケースと具象ユースケースとは異なる。これは記述の枠組み。半完成ユースケース

## ユースケースフレームワークの例

- ユースケース名 : {**登録対象オブジェクト**}登録ユースケース
- アクター : {**オブジェクト登録希望者**}
- 目的 : **登録対象オブジェクト**を登録する
- 事前条件 : アクターが{**登録に必要な情報**}を得ている
- 正常終了における事後条件 :
  - **登録対象オブジェクト**が登録済みとなったという結果をアクターが得ている
- 例外終了における事後条件 :
  - **登録対象オブジェクト**の登録が中止され, 副作用が残っていない

## 続ユースケースフレームワークの例

- ユースケース起動動作 :
  - アクターはシステムに**登録対象オブジェクト**の登録希望を{**サービス要求インターフェース**}を介して通知する
- ユースケース全体にかかる代替系列 :
  - アクターが登録希望を取りやめたときは、登録対象オブジェクトの登録を中止し、副作用を残さない

## 続々ユースケースフレームワークの例

- 基本系列 :
  - 1. システムはアクターに**登録対象オブジェクト**を登録するための{**登録インターフェース**}を提示する .
    - :::代替系列1-1: **登録インターフェース**が提示されない
  - 2. アクターが入手した**登録インターフェース**を介してシステムに登録対象オブジェクトの**登録に必要な情報**を渡すと、システムは渡されたデータが{**データ制約**}を満たし、かつ既に同じデータが登録されていないことを確認し、登録対象オブジェクトを{**永続化**}する .
    - :::代替系列2-1: データが正当である条件を満たしていなかった場合は次のことを行う .
    - ::::2-1-1. システムはデータが不当である理由を明らかにする .
    - . . . . .

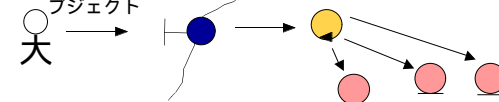
## 定型的なユースケースの適用範囲

- 情報処理学会ソフトウェア工学研究会の要求工学ワーキンググループで採用されている共通例題「プログラム委員長業務」のユースケースを記述

複合ユースケース数	2 9
基本ユースケース数	5 6
フレームワークが適用できる基本ユースケース数	5 1
ユースケースフレームワーク数	6

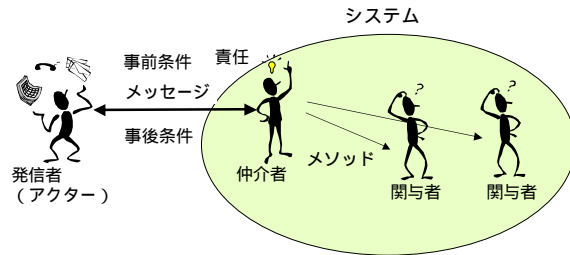
## ユースケース駆動型によるモデル化

- インタフェースオブジェクト
  - アクターとシステムの境界上に存在し、アクターからメッセージを受け取るオブジェクト
- エンティティオブジェクト
  - システムの状態を保持するオブジェクト
- コントロールオブジェクト
  - エンティティオブジェクトへのメッセージの配信を行うオブジェクト



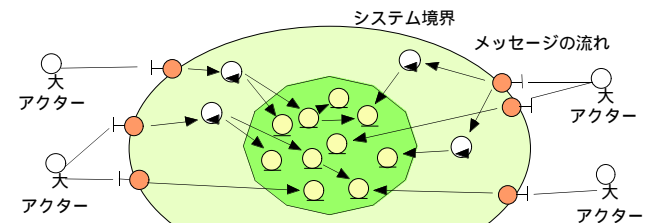
## オブジェクトの世界

- メソッドや関与者は情報隠蔽されており、メッセージの発信者からは見えない



## ユースケースの世界観

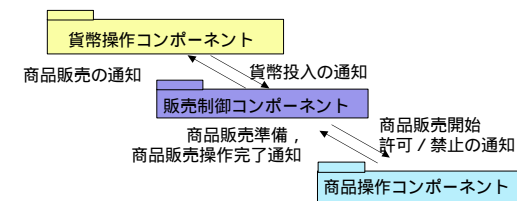
- 協調するオブジェクト
  - インタフェースオブジェクト
  - コントロールオブジェクト
  - エンティティオブジェクト



## ソフトウェアアーキテクチャ決定後の適用

- アーキテクチャ
  - コンポーネントの役割分担を規定
- 役割分担に基づき、コンポーネント間の相互作用をユースケースで定義
- ユースケースに基づき、コンポーネントのインタフェースを定義
  - 個々のコンポーネントを独立に開発

## コンポーネントとユースケース



## ユースケースの様々な種類

- ビジネス活動の記述：ビジネスユースケース
- システムのユーザの大まかな活動を捉えるための記述
- 要求抽出のために定義されるユーザとシステムとの間の相互作用の概略の記述
- 機能仕様書として定義される詳細な要求定義
- オブジェクト，コンポーネント間の相互作用の記述
  - など．いずれも繰り返し，詳細化と内容の検討が行われる

## 戦略的システム開発への応用

- 高品質のシステムを短期間で開発
  - 他社よりも早く市場投入
  - コンポーネントに基づく設計
  - プロダクトラインに基づく開発戦略

## ユースケースが注目される理由

- ユーザとシステムの相互作用を重視するシステムが主流
  - インターネットなど
  - ワンクリック（ビジネス特許）
- システムがユーザに提供する機能のみの記述だけで要求を定義できなくなってきた
- より柔軟／斬新なユーザインタフェース，相互作用シナリオの開発が必要

## www.amazon.co.jp



www.honyasan.co.jp



ユースケースを用いる要求分析

- 57 -

SLagoon NAKATANI Takako, 2001.

## 参考文献

- Buhr, R. and Casselens: Use Case Maps for Object-Oriented Systems, Prentice-Hall, 1996.
  - (金沢典子他訳, ユースケースマップ, トッパン(1999))
- Cockburn, A.: "Basic Use Case Template," <http://members.aol.com/accoburn/papers/uctemplate.htm>
- Constantine, L.: "Essential Modeling, Use Cases for User Interfaces," Interactions, Vol.2, No.2, pp.34-46(1995 April).
- Fowler, M. and Scott, K.: UML Distilled, Addison-Wesley (1997)(UMLモデリングのエッセンス)
- Jacobson, I., Christerson, M., Jonsson, P. and Overgaard, G.: Object-Oriented Software Engineering: A Use Case Driven Approach, Addison-Wesley (1992).
  - (西岡利博他訳, オブジェクト指向ソフトウェア工学OOSE, トッパン (1995))
- Jacobson, I., Booch, G. and Rumbaugh, J.: The Unified Software Development Process, Addison-Wesley, 1996.
  - (日本ラショナルソフトウェア, UMLによる統一ソフトウェア開発プロセス, 翔泳社)
- 中谷多哉子, 玉井哲雄: "ユースケースフレームワークの検討", ソフトウェアシンポジウム論文集, pp.259-216 (2000).
- 中谷多哉子, 玉井哲雄: "ユースケース記述のためのフレームワークとメタモデル", オブジェクト指向2000シンポジウム論文集, pp.141-148 (2000).
- 山浦眞人, 安達隆, 栗田健司, 中谷多哉子: "上流工程における再利用部品としてのユースケース", 情報処理学会第60回全国大会論文集, 2000, pp.1-203-1-204.
- Rumbaugh, J., Jacobson, I. and Booch, G.: The Unified Modeling Language Reference Manual, Addison-Wesley, 1999.
- Schneider, G. and Winters, J. P.: Applying Use Cases - A Practical Guide-, Addison-Wesley, 1998.
- UML, <http://www.rational.co.jp>
- 吉田裕之, 山本重枝子, 上原忠弘, 田中達雄: UMLによるオブジェクト指向開発実践ガイド, 技術評論社, 1999.

ユースケースを用いる要求分析

- 58 -

SLagoon NAKATANI Takako, 2001.