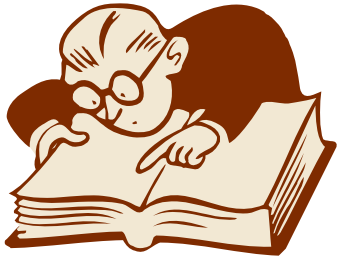
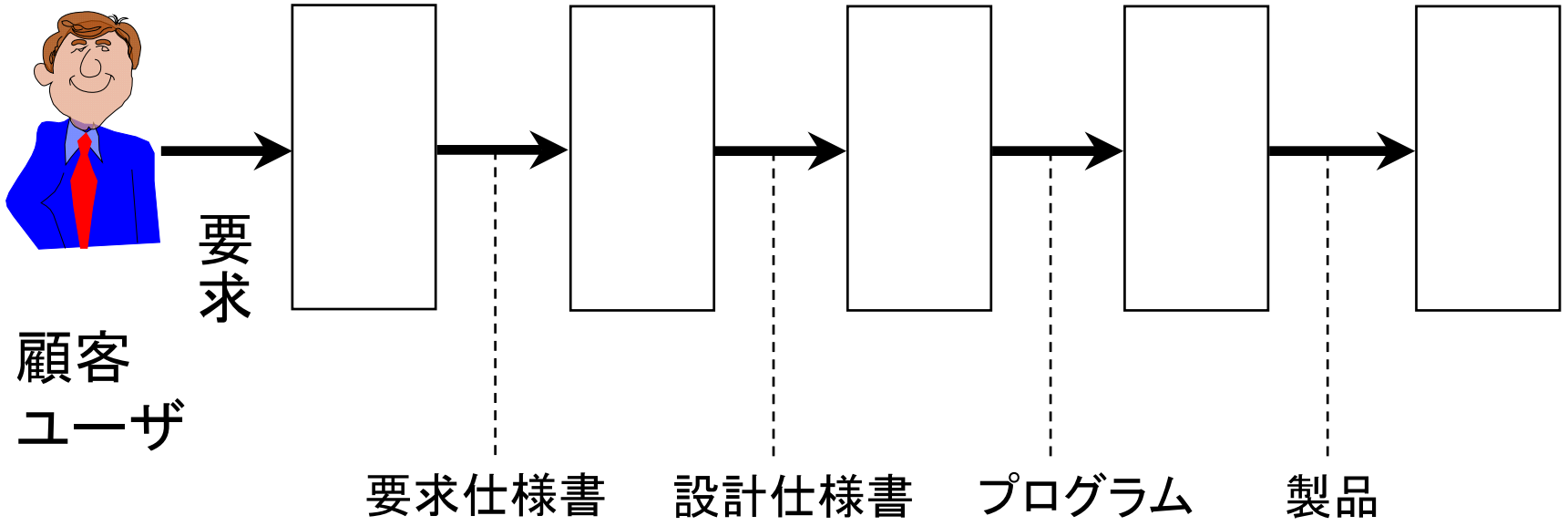
これまでならってきたこと

- オブジェクト指向設計法
  - クラス図の書き方
  - ソースコードへの変換
  - デザインパターン
- では，どうやってクラス図を書く？
  - クラス，オブジェクト等の抽出は？
- そもそもどんなソフトを作るのか？

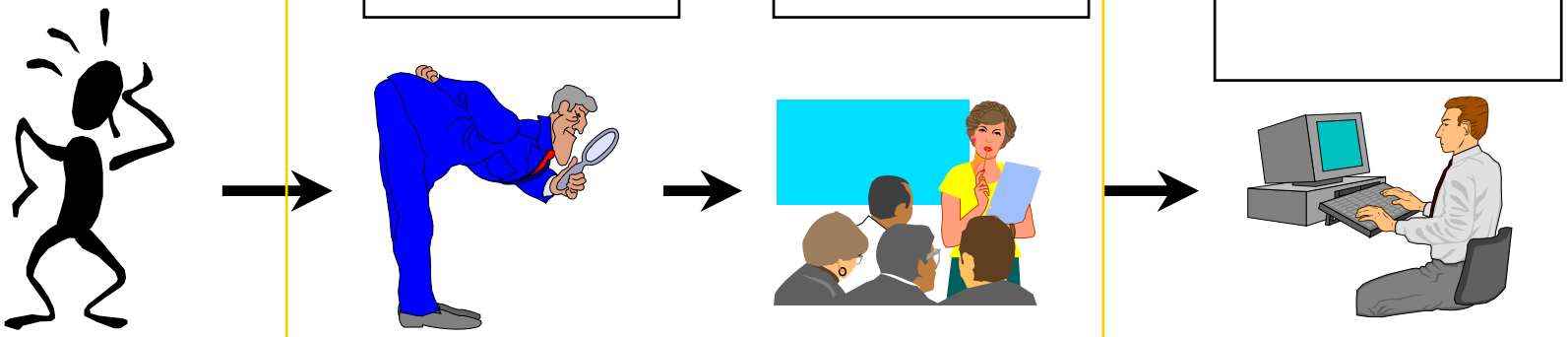


ソフトウェア開発プロセス

ソフトウェアのライフサイクルモデル



要求分析



困っている
顧客・ユーザ

市場調査
資料の収集
インタビュー
アンケート

発想法

機能，データ
振る舞い
構成要素
性能，信頼性
納期，予算

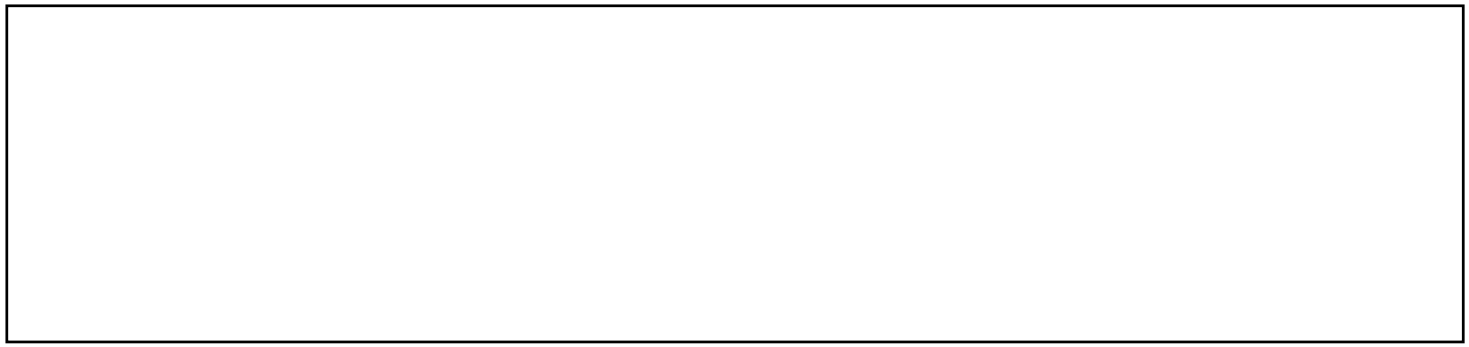
要求分析の例（1）

昔の銀行

問題点

1）お金をおろしたり(Withdraw),
預けたり(Deposit)する際，自分の銀行の
窓口に行かなければならない.

2）銀行の窓口も照会が面倒.



要求分析の例(2)

現状調査(お金の引き出し)

行わなければならないこと

1. 引き出し用の書類に氏名等を記入し，印鑑を捺す
2. 通帳とともに銀行の窓口に提出
3. 窓口ではその人の口座を記録した帳簿をさがし，チェック
4. OKなら現金を用意し，通帳を書き換える．帳簿も書き換える
5. 本人に現金と通帳を渡す．

要求分析の例(3)

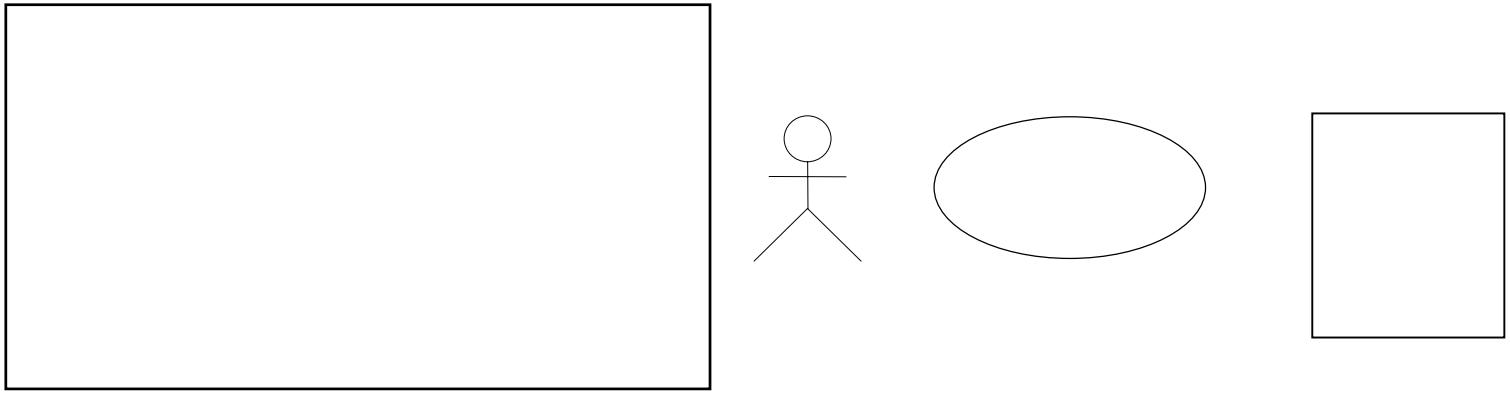
要求の獲得

- データベース化
  - 入力・管理が面倒くさくないように
  - ただちにチェックができる
- ネットワーク化
  - 接続された銀行の支店でもOK
- 通帳の代替案
  - 電子化：磁気カード，ICカードなど
- 印鑑の代替案
  - 暗証番号，指紋，網膜紋など

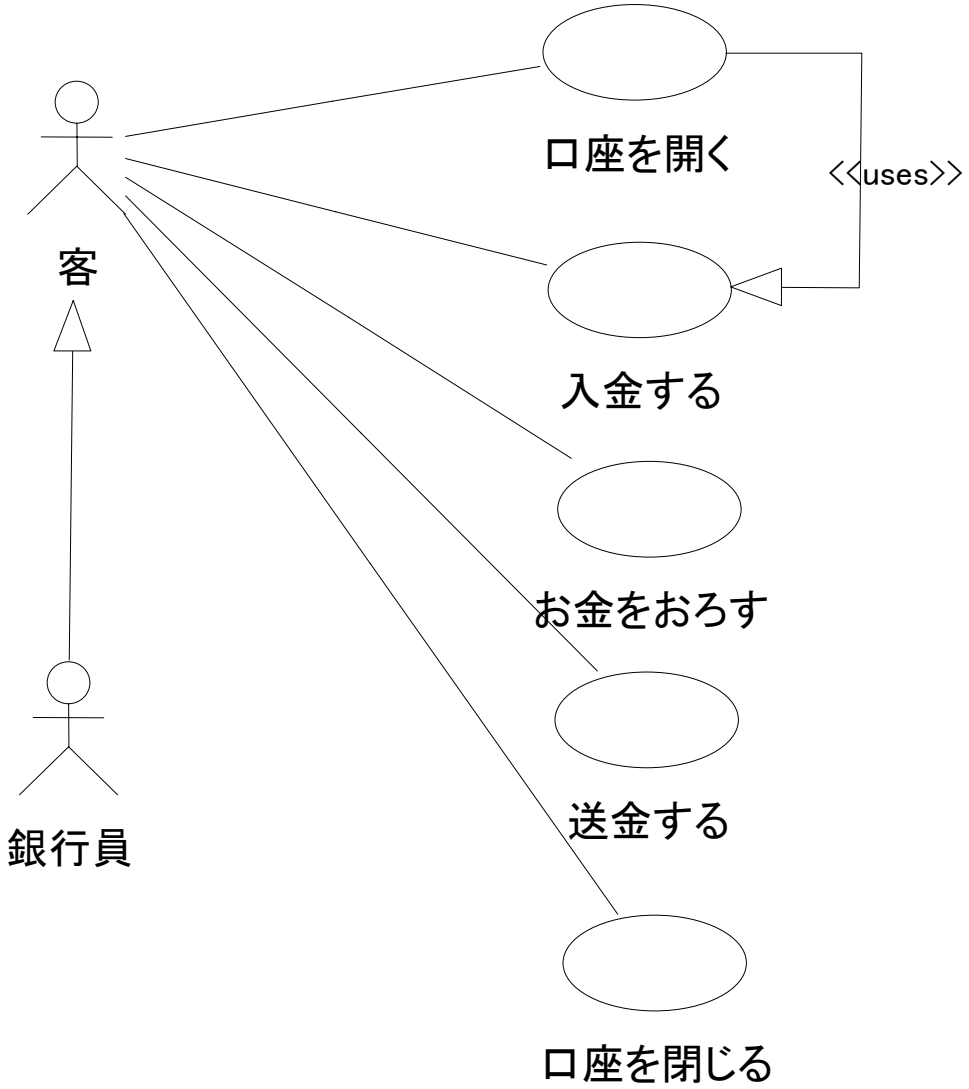
分析結果を書く:ユースケース図

ユースケース図を使用

- システムが行わなければ機能
- 外部(アクター)から見る
- １つの機能を１つのユースケース
-
-
-



ユースケース図の例

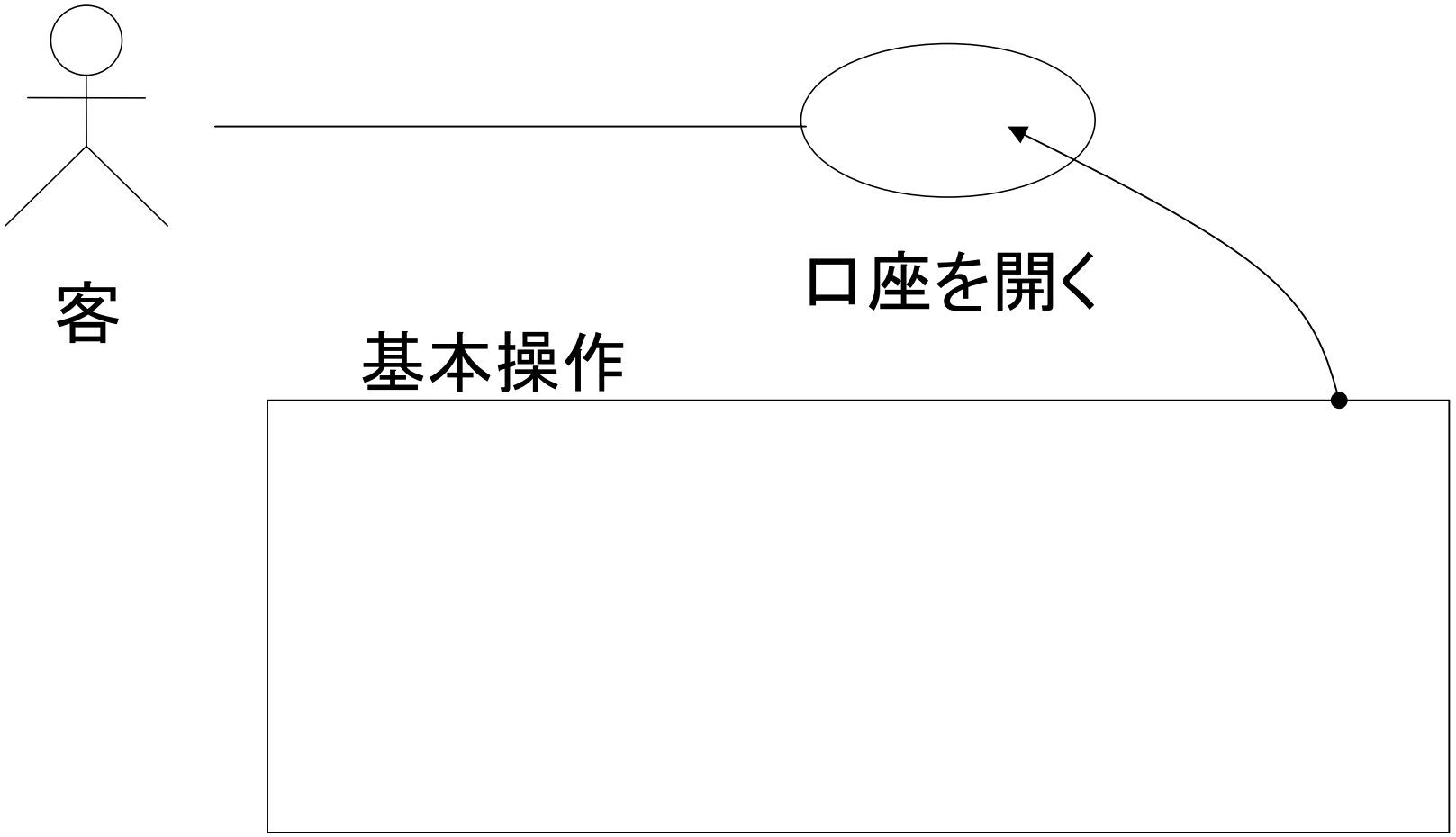


ユースケースの記述（1）

- ユースケースの目的
- ユースケースの始動法
- アクターとユースケース間のメッセージ交換
- エラー処理や例外処理（代替フロー）
- ユースケースの終了法

中身は通常は自然言語で書く

ユースケースの記述（2）



ユースケース名：　口座を開く

アクター：　口座を開設しようとする人

目的：　口座を開設し，入金処理を行う

事前条件：　アクターは実在する人でなければならない

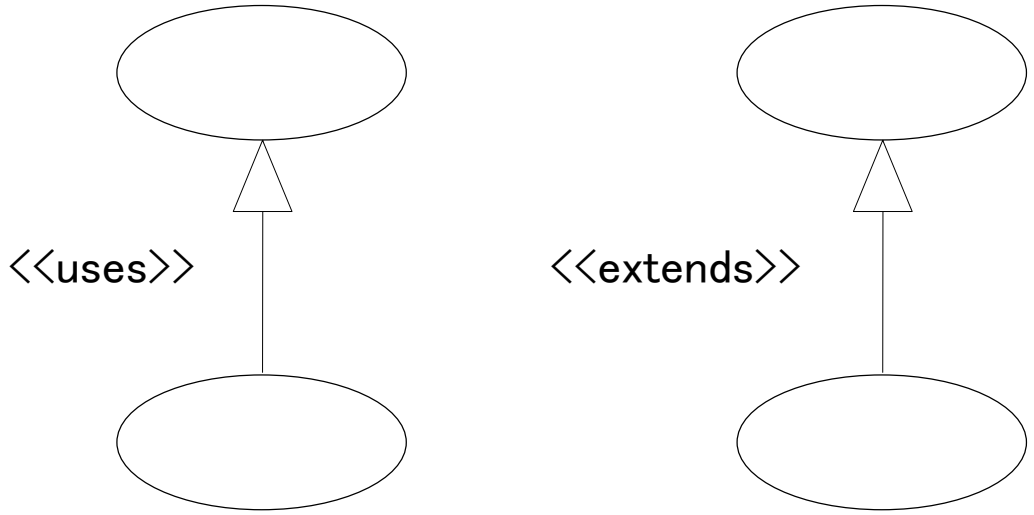
基本系列：

1. アクターは申込書を書き，印鑑を添えて提出する
2. システムは本人であるかどうかを確認する
3. システムは通帳を作成する
4. アクターは入金を行う
5. システムは印鑑，通帳をアクターへ返却する

事後条件：アクターのもとには印鑑と，入金が記載された通帳があり，入金された口座がある．

代替系列：

ユースケース間の関係



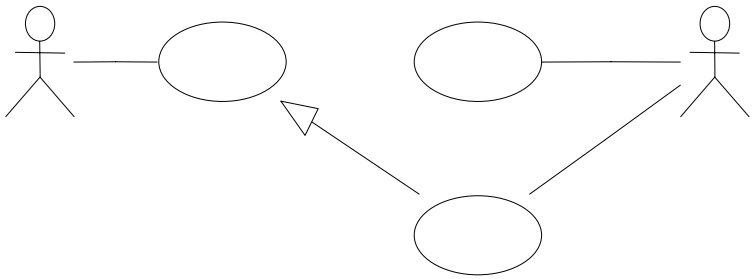
uses：他のユースケースが使用する
extends：他のユースケースの振舞い(の一部)を含む
再利用関係

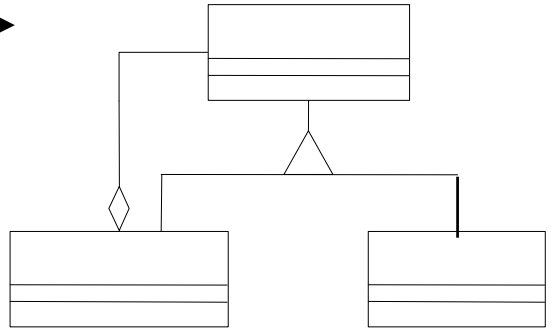
ユースケースに基づく開発プロセス

要求分析

設計

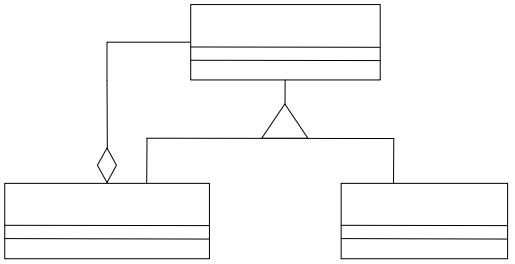
クラス・オブジェクト
の発見



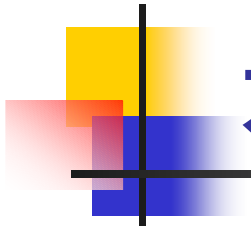


ユースケース図

クラス図，．．．

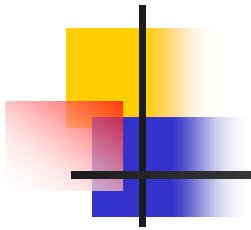


オブジェクト・クラスを
抽出するには？



オブジェクト・クラスの抽出

- 名詞に注目
- シナリオ
- ユースケース
- CRC法(Class-Responsibility-Collaboration)
- ．．．



ユースケースによる クラス･オブジェクトの抽出

- ユースケース記述中のステップやアクションを，クラス，クラスの操作，クラス間関係に割り当てる
- ユースケース記述中の名詞に注目
  - 客，銀行員，口座，お金，通帳，印鑑，申込書，．．．

クラス図の例