

ソフトウェア工学

03: 要求分析／定義と設計 (I) - 構造化分析／設計

理工学部 経営システム工学科
庄司 裕子



今回のテーマ

- ⇒ 要求分析／定義と設計 (I)
 - 構造化分析／設計
 - ❖ 開発プロセスにおける位置づけと目的
 - ❖ 構造化分析
 - 👉 構造化設計 → 次回
 - ❖ 支援ツール／環境

2

要求分析／定義と設計 (I) - 構造化分析／設計

- ⇒ 開発プロセスにおける位置づけと目的
- ❖ 構造化分析
- 👉 構造化設計
- ❖ 支援ツール／環境

3

開発プロセスにおける位置づけ

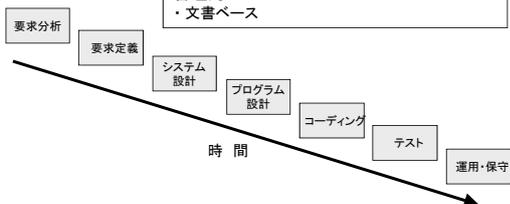
要求分析／定義と設計は以下の作業に相当：

- ❖ ウォーターフォール型開発モデルの
 - ❖ 分析フェーズ (要求分析、要求定義)
 - ❖ 設計フェーズ (システム設計、プログラム設計)
- ❖ 反復型開発モデルの1反復サイクルの
 - ❖ 要求分析／定義タスク
 - ❖ 設計タスク

4

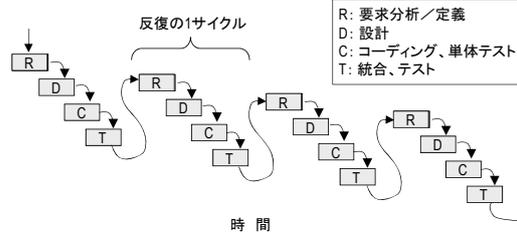
ウォーターフォール型開発モデル

- ・ トップダウン方式の問題解決手法
- ・ 上流から下流へとリニアなタスクの流れ
- ・ 上流で仕様が完全かつ明確に記述される必要がある
- ・ 仕様が最初に明確化できる場合には、最も合理的
- ・ 文書ベース



5

反復型開発モデル



出典：フィリップ・クルーシェン著『ラショナル統一プロセス入門』

6

要求分析／定義の目的

- ❖ アナリスト(または上級SE)が
- ❖ ユーザの要求を分析して理解し
- ❖ システム機能仕様書(あるいは単に機能仕様書)にまとめる

☞ どのようなシステム(what)を作るかを考える

7

設計の目的(Ⅰ)

❖ システム設計

- ❖ 設計者(またはSE)が
- ❖ 機能仕様書を基にシステムのモジュール構成、個々のモジュールの機能、モジュール間のインタフェースを決定し
- ❖ システム設計仕様書(モジュール構成図、モジュール機能仕様書、モジュールインタフェース仕様書)にまとめる

☞ 目的とするシステムをどう(how)作るかを考える

8

設計の目的(Ⅱ)

❖ プログラム設計

- ❖ プログラマが
- ❖ 個々のモジュールに使用するアルゴリズム(ロジック)とデータ構造を決定し
- ❖ プログラム設計仕様書あるいはロジック設計仕様書にまとめる

☞ 目的とするシステムをどう(how)作るかを考える

9

要求分析／定義と設計の手法

☞ 構造化分析／設計

- ❖ 構造化分析(Structured Analysis: SA)／構造化設計(Structured Design: SD)の総称
- ❖ システム化の対象をさまざまな単位に分割し段階的に詳細化
- ❖ ERモデリング
 - ❖ 実世界を実体(Entity)と関連(Relationship)で表現
- ❖ オブジェクト指向分析／設計とUML
 - ❖ データとその処理／機能をオブジェクトとして一体化。UMLはさまざまなOOADを統一したもの

10

要求分析／定義と設計(Ⅰ) － 構造化分析／設計

- ✓ 開発プロセスにおける位置づけと目的
- ⇒ 構造化分析
- ☞ 構造化設計
- ❖ 支援ツール／環境

11

システム機能仕様書の記述方法

- ❖ 自然言語(日常言語)
 - ❖ 厳密な記述ができないという欠点がある
- ❖ 形式言語
 - ☞ データフローダイアグラム(DFD)記法
 - ❖ 拡張DFD記法
 - ❖ 制御システム向け(Ward)
 - ❖ 実時間システム制御向け(Hatley)
 - ❖ JSD(ジャクソンシステム開発法)
 - ❖ …

12

構造化分析とは

- ❖ システム機能仕様書を以下の道具で記述
 - ❖ データフローダイアグラム (DFD)
 - ❖ データディクショナリ (DD)
 - ❖ ミニ仕様書
- ❖ SA (Structured Analysis) と略称される
- ❖ 提唱者 Tom DeMarco の名前を取って、デマルコ (DeMarco) 法とも呼ばれる

13

DFD (データフローダイアグラム)

- ❖ システム全体の情報の流れをグラフ状の図面で記述
 - ❖ 制御データを書けない (書いてはいけない)
- ❖ 個々の情報の処理手順を逐次的なプロセスとして記述

14

DFDで扱われるデータ

- ❖ システム内を定常的に流れるものに限定
 - ❖ 制御情報は除外
 - ❖ たとえば、データ処理に関して発生するイベント
- ⇒ データの処理方法 (how) ではなく、データが流れる状態 (what) に着目する

15

DFDを構成する基本要素

- ❖ 矢印: データフロー 
 - ⇒ 定常的なデータの流れ
- ❖ 丸印: プロセス (バブル) 
 - ⇒ データの処理
- ❖ 二重線: データストア (ファイル) 
 - ⇒ データの一時的な格納場所
- ❖ 四角形: ソース/シンク (エンティティ) 
 - ⇒ 現在の記述対象システムの外部にある組織や人 (データの発生/吸収源)

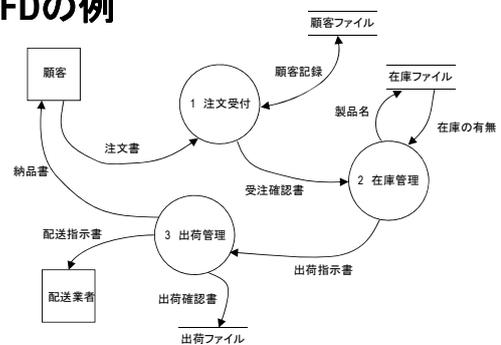
16

DFD記法

- ❖ 4種類の記号のみ使用
 - ❖ シンタクスとセマンティクスを明確に分離
 - > 記号の組み合わせ: シンタクス (構文)
 - > 記号の名前の持つ情報: セマンティクス (意味論)
 - ❖ 特別の器具がなくても容易に作図できる
- ⇒ 分析の使いやすい道具になる

17

DFDの例



※ 日経ソフトウェア (編) 「ゼロから学ぶソフトウェア設計」所収の図を一部変更

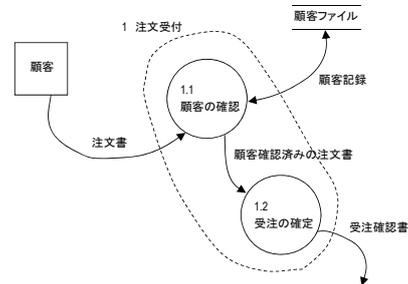
18

DFDの階層化による詳細化

- ❖ トップダウン方式
 - ❖ 概要を先に、詳細は後に
 - ❖ 全体図に登場する各プロセスの詳細DFDを用意
- ❖ 1枚のDFD内のプロセス数は7個前後が目安
 - ❖ 心理学で言う「マジックナンバー7」
 - ❖ それより多くなりそうなら、少し抽象度の高いプロセスにまとめて記述し、それぞれのプロセスの詳細を下位DFDとして1枚ずつ用意する
- ❖ プロセスへのデータフローの出入りは詳細化の前後で一致する必要がある

19

DFDの階層化による詳細化(例)



※ 日経ソフトウェア(編)「ゼロから学ぶソフトウェア設計」所収の図を一部変更

20

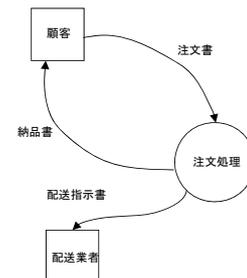
作業の手順

- ① 最上位DFDである「コンテキストダイアグラム」を描く
 - システム全体を1つの丸印で表し、システムと外界(ソース/シンク)の間でやり取りされるデータの流れを描く
- ② 上位DFD内のプロセスを詳細化
- ③ プロセスの機能を1ページ以内のミニ仕様書で完全に記述できるまで詳細化を繰り返す

21

コンテキストダイアグラム

- ❖ 1個の丸印 (プロセス)
- ❖ 複数の四角形 (ソース/シンク)
- ❖ 複数の矢印 (データフロー)



22

ミニ仕様書

- ❖ 最下位プロセス1つ1つの逐次的な処理手順を記述する
- ❖ 記述方法(書式)
 - ❖ 構造化言語
 - ❖ フローチャート
 - ❖ デシジョンテーブル(デシジョンツリー)
 - ❖ ...

23

データディクショナリ

- ❖ DFDで記述したデータフローやプロセスに関して、より詳細なデータ項目を定義したもの
- ❖ 表記法(デマルコの記法)
 - ❖ =: 等価
 - ❖ +: 接続(~と~)
 - ❖ [: 選言(または)
 - ❖ {: 反復
 - ❖ (: 任意の選択
- ❖ (例): 商品リスト=(商品番号+商品名)

24

要求分析／定義と設計(Ⅰ) － 構造化分析／設計

- ✓ 開発プロセスにおける位置づけと目的
- ✓ 構造化分析
- ☞ 構造化設計
- ⇒ 支援ツール／環境

25

支援ツール／環境

- ❖ 統合CASEツール
 - ❖ CASE (Computer-Assisted Software Engineering)
 - ❖ 分析/設計から保守まで一貫支援
 - ❖ ソースとドキュメントを一元管理
 - ❖ UMLベースのIDEも統合CASEツールの一種
- ❖ 個々の工程専用のCASEツール
 - ❖ 上流CASEツール
 - ❖ 下流CASEツール
 - ❖ …

26

まとめ: 構造化分析(Ⅰ)

- ❖ システム機能仕様書を以下の道具で記述
 - ❖ データフローダイアグラム (DFD)
 - ❖ データディクショナリ (DD)
 - ❖ ミニ仕様書
- ❖ システム全体の情報の流れをグラフ状の図面で記述
- ❖ 個々の情報の処理手順を逐次的なプロセスとして記述

27

まとめ: 構造化分析(Ⅱ)

- ❖ 扱う情報はシステム内を定常的に流れるものに限定(制御情報は除外)
- ❖ データの処理方法 (how) ではなく、データが流れる状態 (what) に着目する
- ❖ DFDの基本要素
 - ❖ データフロー、プロセス、ストア、ソース/シンク
- ❖ 作業手順
 - ❖ コンテキスト→DFDの詳細化→ミニ仕様書の作成

28

参考文献

- ❖ 有沢誠:「岩波コンピュータサイエンス ソフトウェア工学」, 岩波書店
- ❖ 日経ソフトウェア(編):「ゼロから学ぶソフトウェア設計」, 日経BP
- ❖ 鈴木正人:「ソフトウェア工学－プロセス・開発方法論・UML－」, サイエンス社

29

考えてみよう

- ❖ Web上のオンラインショッピングサイトでよく見かける個人情報の管理(登録/確認/変更)の処理をDFDで記述してみなさい。

30