
実 践 演 習 教 材

静岡大学情報学部
組込みシステムアーキテクト研究所

目次 Contents

<第 I 部>

要求仕様の開発.....	1
プロジェクトマネジメント論ことはじめ.....	23
PMBOK の概説.....	27
WBS (Work Breakdown Structure).....	45
EMV と進捗報告.....	50
How to Use WBSEVM-Template.....	64
メトリクス測定：Source Monitor の使い方.....	70
附録：飛行船コマンド生成ソフトウェア要求仕様書.....	81
実践演習.....	102
オリエンテーション.....	104
システムのアーキテクト養成.....	112
実践演習中の取り組み方.....	117
実践演習の日程とお約束.....	123
要求.....	128
より実り多い実践演習のためにプロセス設計.....	134
プロセス設計.....	143

ねらいと各種制約の確認.....	150
モデリングとは.....	168
機能構造振る舞い.....	178
構造と機能の因果関係.....	184
システムの機能とソフトウェアの機能.....	186
概念化とモデル化.....	190

要求仕様の開発

ユースケースとソフトウェア要求仕様書

静岡大学 情報学部
野口靖浩, 松澤 芳昭

目次

- 議論の背景: 情報システムとV&V
- IEEEが推薦するソフトウェア要求仕様書
- ユースケースによる要求記述

要求仕様の方法論は数あれど

- ゴール指向要求分析法?
 - CAOS, i*
- UML? / SysML?
 - ユースケース, SysML要求図
- 形式仕様記述?
 - Z, Bメソッド
- 人間活動も含めた問題分析法?
 - 問題フレーム, SSM(Soft System Methodology)
- 超一般的問題解決法?
 - マインドマップ, KJ法, 目的一手段展開

やはり, 解決していない・・・



プロジェクト依頼者の考えていたこと



プロジェクト要求書に書かれていたこと



システム分析者が設計したもの



プログラマが作ったプログラム



利用者側に導入されたプログラム



利用者が要求していたもの

当面の解決案

- 考え, 未来を想像する努力をすること
 - 一般的な問題解決の手法は存在しないので
- 目的と手段を明確にする努力をすること
 - 問題の定義をし, 方向を見誤らないようにする
- スコープ(目標)を明確にする努力をすること
 - かつ, 変更への準備を怠らないことも重要

ところで, そもそも, 「情報システム」というスコープは明確か

情報システムの定義

- マネージャ, スタッフ, 顧客, 市民を含め, 情報の利用を望んでいる人々にとって, 手に入れやすく, 役に立つような形で, 組織体(または社会)に適切な情報をあつめ, 保管し, 処理し, 伝達するシステムである. 情報システムは人間活動の(社会的な)システムであって, コンピュータシステムを利用している, いなくてもよい

英国の情報システムカリキュラムで用いられているもの(「情報システム学へのいざない」から引用)

PMBOKの概説

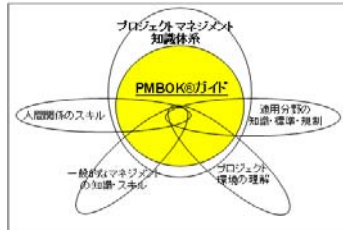
(PMBOK: Project Management Body of Knowledge)

松澤 芳昭

PMBOKの位置づけ

- 知識体系として、PM間の共通用語として使える。
- Full-specで扱うことはない。実際のプロセスに取り入れるかはPMの判断次第である。
 - 以下、p.77より引用
 - 経験のあるプロジェクトマネジメント実務者のほとんどは、プロジェクトをマネジメントする方法が一通りでないと認識している。望ましいプロジェクト・パフォーマンスを達成するためには、プロジェクトマネジメントの知識、スキル、プロセス等が様々な順序や厳格さの程度で適用される。
 - ただし、特定のプロセスが必要でないと考えたとしても、そのプロセスが適用されないということを意味するものではない。プロジェクトマネジャーおよびプロジェクトチームはすべてのプロセスを検討し、プロジェクトごとに各プロセスを実行する程度を決定しなければならない。

PMBOKの適用範囲



http://www.pmaj.or.jp/online/0511/pmp_bukai.html

- 人間関係のスキル
 - コミュニケーション, リーダーシップ, ネゴシエーション, 問題解決力
- 一般的なマネジメントの知識・スキル
 - 定常業務の計画, 実行(会計, 業務計画, 組織構造, キャリアパス, 健康と安全, IT)
- プロジェクト環境の理解
 - 文化的・社会的環境, 国際的・政治的環境, 物理的環境
- 適用分野の知識・標準・規則
 - いわゆる"業務知識", 技術的要素, 産業の知識, 業務知識

PMBOKの構成

- 1章 序論
 - 基本的な用語の説明
- 2章 単一プロジェクトのプロジェクトマネジメント標準
 - 大まかな全体のマネジメントプロセスの説明
- 3章 プロジェクトマネジメント知識エリア
 - 8つの知識エリアの詳細な説明

WBS (Work Breakdown Structure)

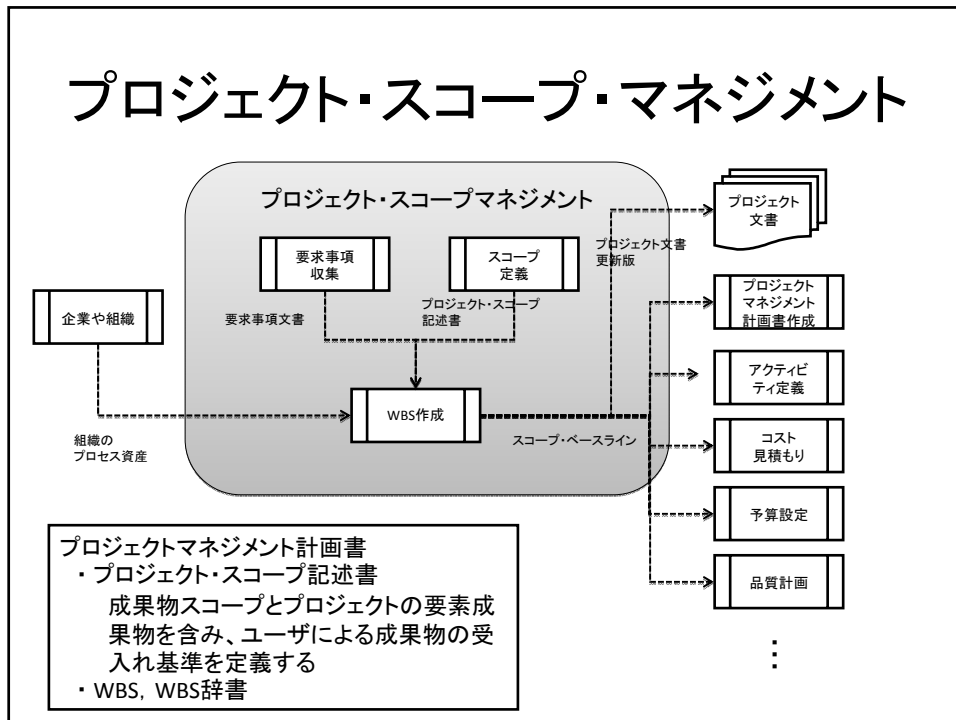
松澤 芳昭

WBS

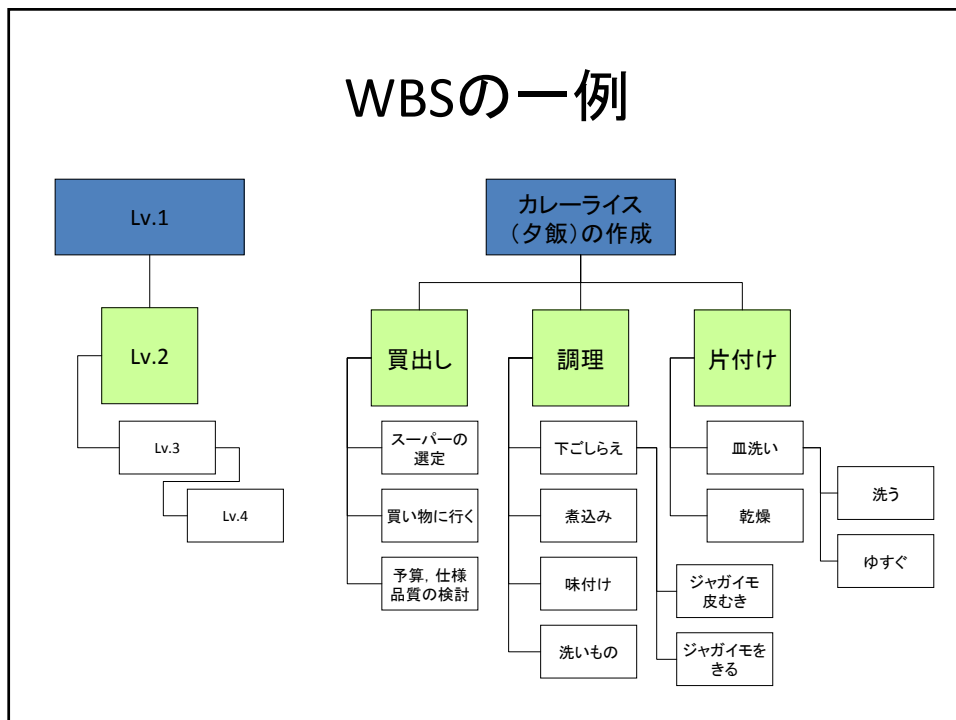
- Work Breakdown Structure
 - WBSはプロジェクト目標を達成し、必要な要素成果物を生成するために、プロジェクト・チームが実行する作業を、**要素成果物を主体に**階層的に要素分解したものである。WBSのレベルが一段下がるごとにプロジェクトの作業をより詳細に定義する。
 - WBSは、プロジェクトのスコープ全体を系統立ててまとめ、定義したものであり、承認されたプロジェクト・スコープ記述書に規定されている作業を表示するものである。

PMBOKガイド第4版より

プロジェクト・スコープ・マネジメント



WBSの一例



EVMと進捗報告

松澤 芳昭

進捗報告

- 進捗報告
 - プロジェクトの進み具合(進捗:しんちよく)を報告する
- 目的
 - プロジェクトが無事に終わることをプロジェクトステークホルダー(上司, 組織, クライアント等)に示す)
 - もしくは, プロジェクトが無事に終わらないことを示す.

よくある進捗報告

- まずい例
 - ～をしました
- なぜまずいか
 - プロジェクトが無事に終わるかどうか評価できない
 - 作業を続けていればいつか終わるのか
 - やってないからできない(環境に問題?)/やってみるけどできない(方法に問題?計画に問題?)のかわからない

何を進捗報告しなければならないか

- 総合的なプロジェクトの状況
 - 青・黄・赤
- 計画に対する実績の状況
 - 何ができている予定であるか
 - どのくらいの時間(コスト)をかけてやったか
 - 実績はどのようであったか
- 質的な分析
 - リスク対策

メトリクス測定 SourceMonitorの使い方

2012-11-29

意図・目標

- モジュール設計の良し悪しを測る仕様
 - 結合度・凝集度(復習)
- ソースコードの良し悪しを測る指標
 - ソースコードメトリクス

ソフトウェアメトリクス

- ソフトウェアの開発過程を様々な視点で定量的に評価したもの
 - 開発工数, レビュー時間, テスト項目数, バグ数, …
- ソースコードメトリクス
 - 開発したソースコードを様々な視点で定量的に評価したもの

ソフトウェアの品質

- 性能
- 信頼性
- 可用性
- 安全性
- 全体としての顧客満足度
- 実装レベルでは主に以下の尺度がフォーカスされる
 - バグの少なさ
 - 機能修正や追加のしやすさ

実践演習

静岡大学 情報学部
組込みシステムアーキテクト研究所
2013年

1

実践演習資料 目次 Contents

1. オリエンテーション
実践演習の概要
2. システムのアーキテクト養成
アーキテクトとは？
3. 実践演習中の取り組み方
アーキテクトとしてのマインドとリーダーシップ概論
4. 実践演習の日程とお約束
いくつかの重要な事柄の確認

2

実践演習資料 目次 Contents

5. 要求
実践演習の「要求」
6. より実り多い実践演習のために
コミュニケーション概論とチーム運営のためのTips
7. プロセスの設計
なにをいつどのように を決める
8. ねらいと各種制約の確認
いくつかの制約条件など
9. モデリングとは
上流から始めるためのTips
10. 機能構造振る舞い

3

第II部

教材の第II部は、次回お渡しします

4

オリエンテーション

実践演習の概要

5

実践演習へようこそ

- 実践演習では、漠然とした要求を受けた状況からはじめ・・・
 - ① 自分たちの知恵と想像力で新しい価値を設計し、
 - ② 実装して
 - ③ 検証して
 - ④ 送り出す
- 簡易ではあってもソフトウェア開発の全工程を体験していただけます

6

実践演習の目的1

- 自由で柔軟な発想の基に堅実なアウトプットを出すことができる「システムアーキテクト」を養成する

7

実践演習の目的2

- 異技術間のスムーズな統合ができるシステムアーキテクトを養成する
- 設計の本質を理解する（設計学）
- 機械、電気、ソフトウェアという、それぞれ異なる特徴を持つ対象を、統合的に扱う
- 「要求に対する手段を考える行為」の本質を捉える
- 大規模開発に対処できる、高抽象度設計と具体設計の両方を遂行できる能力を身につける
- 抽象的に構想し、要素技術を用いて具体的に設計

8

システムのアーキテクト養成

アーキテクトとは？

21

System Architect

- 賛否両論，否定派も擁護派も存在しますが，ここではまず，システムアーキテクトという概念について整理してみましよう
- 「職種をシステムアーキテクトというと，みんな目を見開いて，『一体どんな建物を作ってるんですか！？』と聞いてくる（とある，米国人system architectの紹介文より翻訳抜粋）

22

ArchitectとEngineer

- 大学の教育上は大きな差はない
- たとえば、アーキテクトもエンジニアも、設計図は描く（主流な図のタイプに違いはある模様）
- Architectは、建設業界の建築家と似ている
- 鉄筋コンクリートがどのように乾燥するのかについての知識は、アーキテクトに求められない
 - ただし、鉄筋コンクリートを用いる設計は責任を持って行う

23

Bigger Picture

- 敷いて、アーキテクトとエンジニアの違いを言うのであれば、「全体 vs 部分」
- アーキテクトは、開発対象に対して「全体（Bigger Picture;俯瞰的イメージ）を持つ
- この役割概念のそもそもの狙いは、ひとことで、「全体像が把握できている」人を指す
 - 分業化が進んだ企業ではなかなか難しい？
 - 養成にはかなりの経験（キャリア）が必要？

24

実践演習中の取り組み方

アーキテクトとしてのマインドと
リーダーシップ概論

31

なにはともあれTry & Failで

- 責任放棄するのではなく、まずはやって、挑戦してみましょう
- 仮説をたてて、その仮説を検証するような気持ちで挑戦してみましょう
- 目の前のモニタが20分も同じ状態でボーッとしてしまうような時は、気分転換しながら進めると良いでしょう（時間配分は各チームの裁量に任せています）
- Try & Failを繰り返してみてください

32

高度な思考法を身につけましょう

- まずは 創造的(creative) に
- つぎに 批判的(critical) に
- 発表時は 論理的(logical) に

- 抽象的な構想を具体的な設計に落としこめることが大切です
- 概念をしっかりと理解して作業することができることが大切です
- 構想は抽象的でも良いが、設計が抽象的なことはマズい

33

アーキテクトとしての「自律性」

- 技術リーダーとしてプロジェクトや組織を牽引できる、自律性と協調性が求められます

- 自律性
- 上位の要求を鑑みながら
- システムやソフトウェアの設計案、もしくは開発プロセスやスケジュールを案出し、
- 案のよさを合理的に説明するプロセスを経て、
- 最終的に決断を下すことができる能力

34

要 求

実践演習の「要求」

53

要求(requirements)とは

- A requirement specifies a capability or condition that must (or should) be satisfied. A requirement may specify a function that a system must perform or a performance condition a system must achieve. (OMG SysML v1.2仕様書)

達成されなければならない（もしくは達成されるべき）能力や条件を指定するもの。システムが実現しなければならない機能や、達成しなければならない性能条件を示す。

54

要求(requirements)とは

- より簡潔な定義（吉川・富山, 2000）では, 次のように定義される

これから設計される人工物に要求される属性, 挙動, 機能などの情報.

55

要求 Requirements

- ①障害物が設置された規定のコースを
- ②障害物を検知し, 避けながら走行*する
- ③（ロボット）カーを開発してください

*ライントレースは用いません

56

プロセスの設計

なにをいつどのように を決める

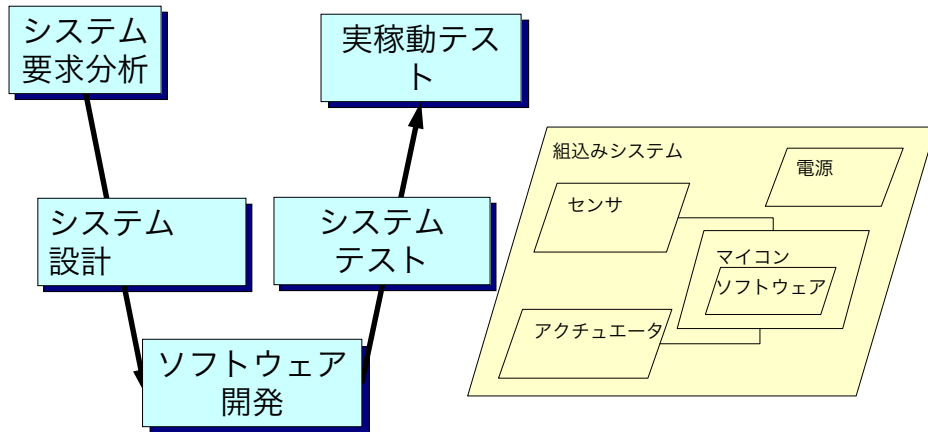
83

開発プロセスとは？

- ここでは
「成果物（設計情報）と作業，及び
それらの関係を記述したもの」とする
 - 組織内標準のような抽象的な開発プロセス
 - 個々のシステムのアーキテクチャに即した
具体的な開発プロセス
- 開発目標に対する「手段」
 - 唯一の正解はない
- 成果物には，システムの構成要素の要求，機能
／振る舞い，構造に関する情報が含まれる

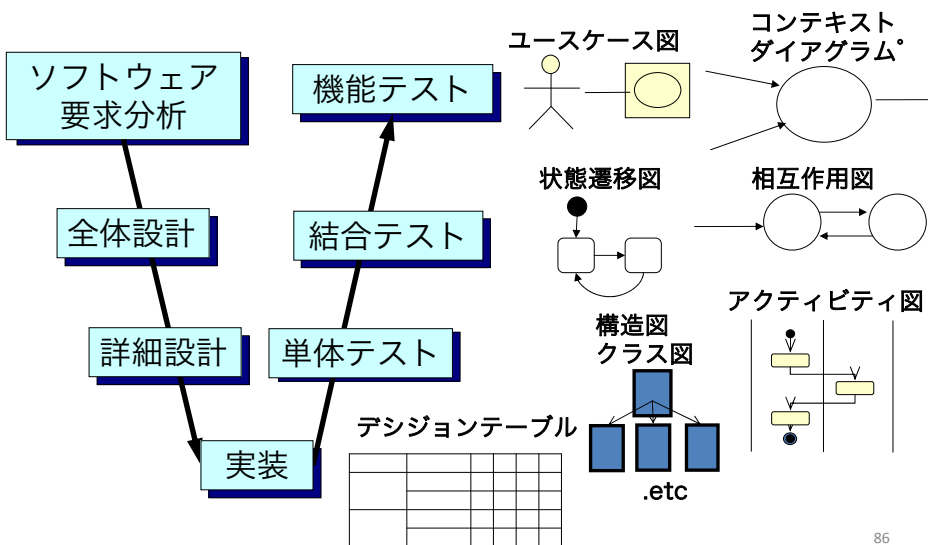
84

組込みシステム開発のV字モデル



85

ソフトウェア開発のV字モデル



86

ねらいと各種制約の確認

いくつかの制約条件など

97

演習の内容と目標

ロボットの構造解析, 機構解析およびソフトウェアによる制御



目標

複数の技術領域, 部門を横断的に
俯瞰し統合できる技術リーダーの育成

さまざまな要求を考慮した設計を
遂行できる技術者

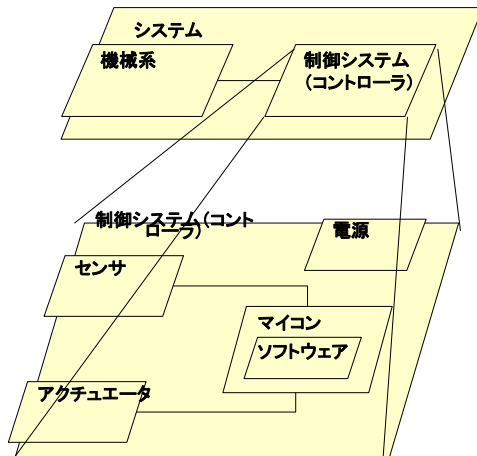


ソフトウェア技術部門から複合部門の
中核になれる人材

- 「理論」「モデリング」「実際のものづくり」を通じて, 異技術間のスムーズな統合を促進する能力を身につける

98

期待される技術：開発技術



粒度	プロセス中のアクティビティ	
システム	要求分析	適格性確認 テスト
	設計	キャリブレーション
制御システム	要求分析	結合
	設計	
ソフトウェア	要求分析	適格性確認 テスト
	方式設計	ソフトウェア 結合
	詳細設計	
	コード作成とテスト	

99

期待される技術：管理技術

- 対象システムの性質に応じて、成果物の定義や、分析・設計プロセスを定義するスキル



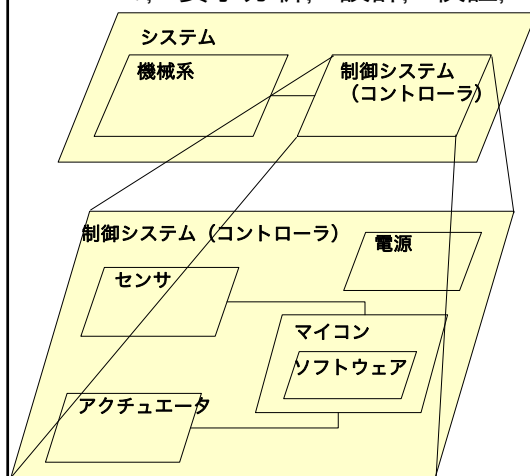
モデリングとは

上流から始めるためのTips

133

スムーズな統合のための「理論」(2)

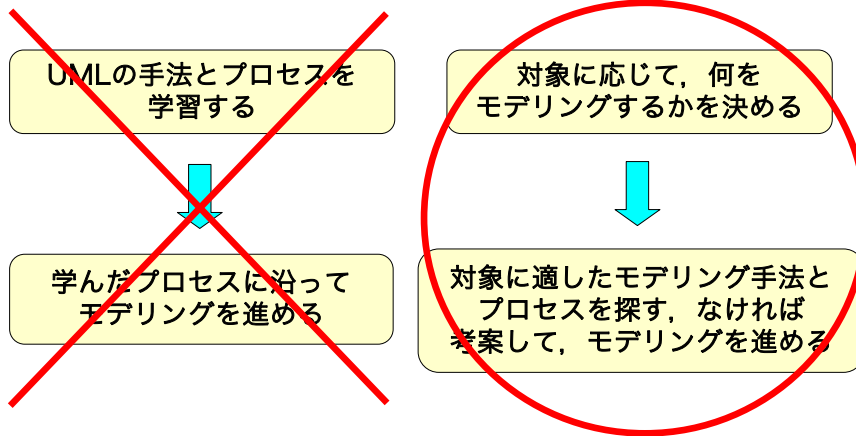
- システム、制御システム、ソフトウェアの粒度を意識して、要求分析、設計、検証、管理を遂行する



粒度	プロセス中のアクティビティ	
システム	要求分析	適格性確認 テスト
	設計	キャリブレーション
制御システム	要求分析	結合
	設計	
ソフトウェア	要求分析	適格性確認 テスト
	方式設計	ソフトウェア 結合
	詳細設計	
	コード作成とテスト	

134

モデリングの本質(1)



135

温故知新:「図面」の特徴

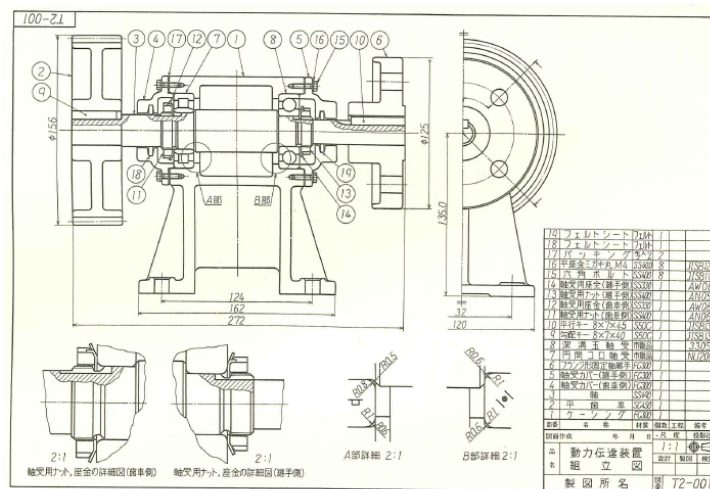


図 7.1 組立図の例 (大西清著「要説機械製図」理工学社より転載)

組立図の例 (吉川弘之「設計学」 p.82より抜粋)³⁶

機能構造振る舞い

Function Structure
Behaviourの確認

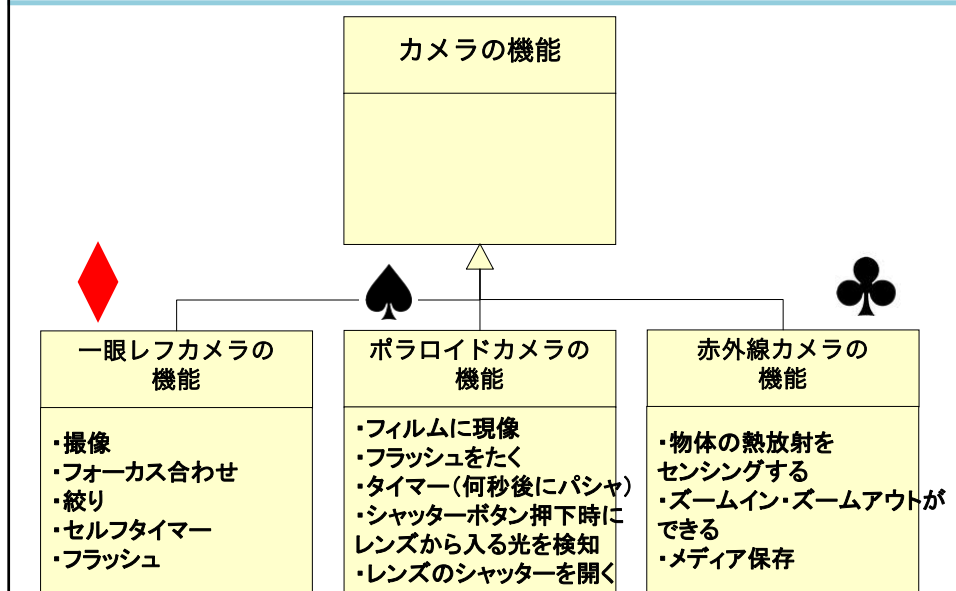
153

「機能」の意味

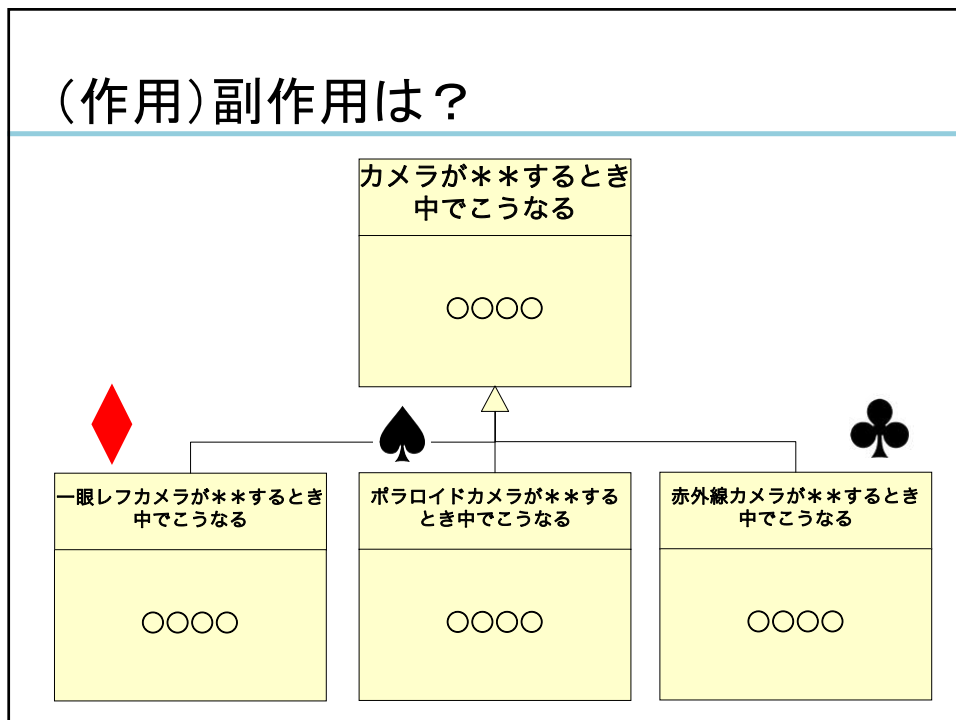
- 何らかの人工物を題材に議論を進め、
普段われわれが「機能」という言葉を
どういう意味で使っているかを見直しましょう
 - まずは、議論の題材を決めましょう
 - ホワイトボードに意見を書いてまとめましょう
 - 題材として選んだ人工物に関して
以下を列挙、調査しましょう
 - 機能
 - 機能が現れるとき、起きている振る舞い
 - (作用,) 副作用
 - 皆さんの結果をみんなで議論し、
「機能」という言葉の意味を考察してみましょう。

154

「機能」には何がある？



(作用)副作用は？



概念化とモデル化

様々なインスタンスを集めて
考える

177

一般論としてのモデリングの効用

- 知識共有
- 合意形成手段の提供
 - モデル化によりお互いの考えを可視化
 - 違いを認識し、その原因を同定
 - 討論し、修正を行う
- 重要な概念の表出化
- 不要な概念の捨象
- 再利用

オントロジーと知識処理（溝口理一郎）

<http://www.ei.sanken.osaka-u.ac.jp/pub/miz/bit99.pdf>

178

構成概念

- 構成概念 (construct) とは
 - 理論的仮説において用いられる、意識的に厳密に定義された、抽象的な概念
 - その存在をとりあえず仮定することによって、複雑に込み入った現象を比較的単純に理解することを目的として構成した概念
 - 構成概念妥当性(construct validity)と呼ばれる視点から妥当性の検証が可能

179

(参考)構成概念妥当性

- 大規模なシステムなどでは、ひとつの概念だけではなく、いくつもの概念が関連しあってひとつの解を出すことがしばしば
- その際、理論的に正しい概念と概念を使うこと（想定すること）ができていないかを検証するための指針
- A) 諸概念間の理論的關係が明確に述べられる
- B) 諸概念の測定間の経験的な關係が吟味される
- C) 経験的証拠がどのようにその構成概念を明らかにしているかを示す

180