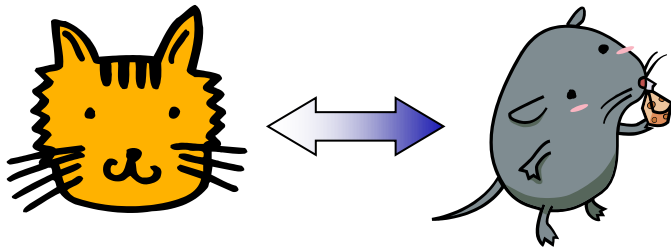


デジタル人材に求められる 設計の知識

2023年9月19日（火）

木野 泰伸



目次

1. 課題意識
2. 既存の体系
3. 設計手順の概要
4. 設計の対象

課題意識

DXの実施者は誰？

ずっと気になっていること

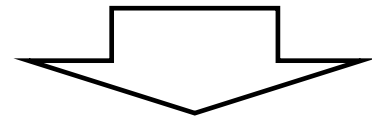
- IT技術者
- 現場業務の担当者
 - ITを設計・開発できる？
 - 業務設計できる？

中小企業様にインタビューに行くと

- どこかで**設計**を習ってきただろうか？

- できる人、いないよ。
- たまたま、いたんだよね。
- 専属の人が雇えるはずもない。

- これからの企業競争力は、**設計力**が大切。
- しかし、小学校から高校までの教育で、
設計について教わることは、ほとんどない。
大学は？ 工学部と芸術学部くらい？
- 現場で DX やれる？
技術者だけに任せるの？



誰が、設計するの？

文系の学生さんも

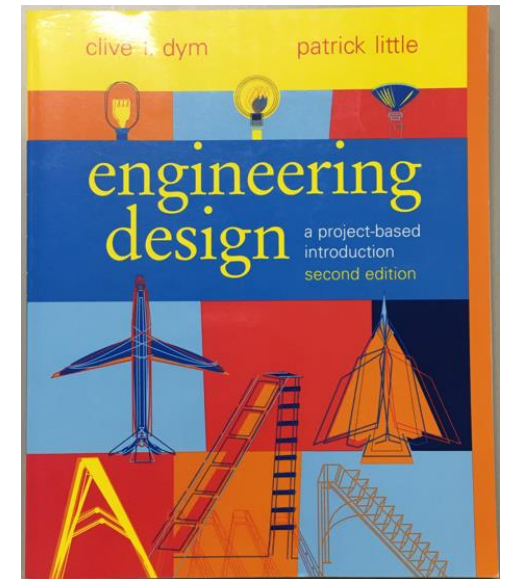
既存の体系

- 建築の設計
- インテリア設計
- 都市設計
- 機械設計
- 製品設計
- 情報システムの設計
- **仕事の設計**
- 組織の設計
- 行政の設計
など
- バウハウスの精神
ルートヴィヒ・ミース・ファン・デル・ローエ（3代目校長）
Less is More （より少ないことは、より豊かなこと）
- デザイン思考
Stanford d.school
- 一般設計学（吉川弘之）
吉川弘之, 設計とは何か 一般設計学の試み, 日本機械学会誌, 84 巻 749 号 p. 328-335, 1981.

Engineering Design

Contents

1. Engineering Design
2. The Design Process
3. Understanding the Client's Problem
4. Functions and Specification
5. Finding Answers to the Problem
6. Reporting the Outcome
7. Managing the Design Process
8. Design for ...
9. Ethics in Design



Clive I. Dyn, Patrick Little, Engineering Design -a project based introduction- 2nd Edition, 2004.



明確とは限らない

イノベーション・課題解決に向けた各種手法

データ

観察・記録

整理・分析

モデル化

モデル

設計・実装・生産

目的

フィールドワーク*

文章

質的分析

特徴を大切に (モデルとの差異)

インタビュー

文献
議事録
アンケート自由記入
インターネット

エスノグラフィ
KJ法
GTA, M-GTA
ケーススタディー
SSM, アクションリサーチ

分類・集合・階層化

時空間と要素関連による
モデル化

The world =
xyz + time with f

個性

目的, 意思ドリブン

V字モデル

Engineering Design

新しい創造 (イノベーション)
課題解決

(音、画像、映像)

数量化 形態素解析
テキスト分析



量的解析

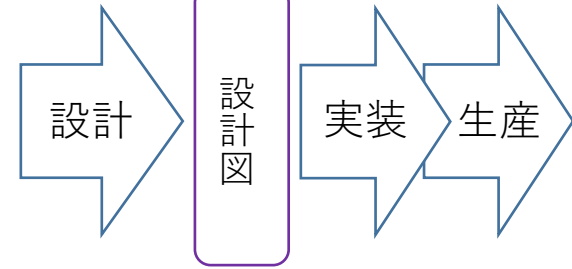
微分積分
代数
幾何

フローチャート
ER図
状態遷移図
構造化 (階層化)



モデル・構造・法則

※図や数式で表記されることが多く、普遍性を持つもの



制御 (統制型、自律分散型)
モジュール分割
QFD, TRIZ,
デザイン思考

数値

計測
実績値
アンケート

相関分析、重回帰
主成分分析、因子分析
SEM (構造方程式モデル)
検定

文法
5W1H

機械学習 (ランダムフォレスト、SVM 等)

ニューラルネット/
ディープラーニング (モデルを自動生成、
構造は見えない)

*全てのデータタイプに共通

プロジェクトマネジメント, IPD
品質、安全、信頼性、リスク

※成果を出すには、
ここまで通して実施。

設計手順の概要（順番は前後することあり）

1. 目的、ニーズの明確化（思想も大切）
2. 機能要件の明確化
非機能要件の整理（目標性能、製品寿命など）
3. 現状の把握（存在しない場合あり）
4. 抽象的設計（現状が存在する場合は以下の(1)～(3)）
（1）概念と物体の分離、（2）時空等の制約解除、（3）重複、順序の見直し
5. 時空を考慮した物理設計
6. 開発すべき技術の整理、アイデアの創出
7. 安全性、信頼性、品質、リスクによる見直し
8. 量産、販売、修理、回収、廃棄の計画による見直し
9. モジュール分割インターフェースとパラメータ設計
10. 詳細設計へ

1. 目的、ニーズの明確化 (思想も大切)

- 使用者は誰か？
- 何のために作られるのか？
- 設計者の思想 人生観、環境への理解
- ニーズの把握と整理
機能的なニーズ、性能的ニーズ、官能的ニーズ
意識されているニーズ、無意識のニーズ
ブレインストーミング、KJ法、GTA、QFD、テキスト分析 など

設計手順の概要（順番は前後することあり）

1. 目的、ニーズの明確化（思想も大切）
2. 機能要件の明確化
非機能要件の整理（目標性能、製品寿命など）
インタビュー（打合せ）
整理方法 表にしたり
概念の抽出
3. 現状の把握（存在しない場合あり）
4. 抽象的設計（現状が存在する場合は以下の(1)～(3)）
（1）概念と物体の分離、（2）時空等の制約解除、（3）重複、順序の見直し
5. 時空を考慮した物理設計
6. 開発すべき技術の整理、アイデアの創出
7. 安全性、信頼性、品質、リスクによる見直し
8. 量産、販売、修理、回収、廃棄の計画による見直し
9. モジュール分割インターフェースとパラメータ設計
10. 詳細設計へ

要件の明確化

- しかし、**担当者やユーザーが、自分たちの課題やニーズを明確に把握しているわけではない。**
- そのため、要求/要件定義は難航する。

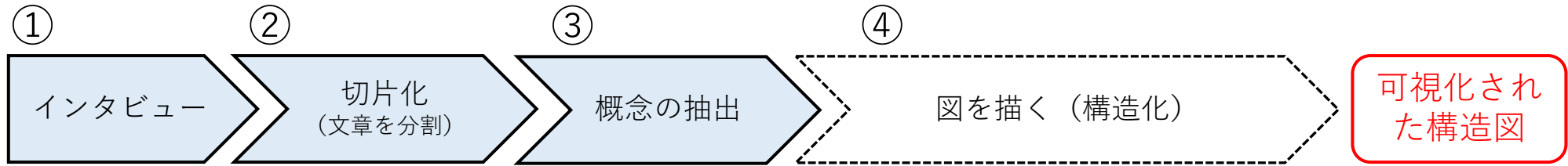


- **中途半端な要求/要件定義**で実施されたプロジェクトの成果物は、ユーザーの**ニーズを満たさなくなる**可能性が高い。



エスノグラフィー、GTA (Grounded Theory Approach)、KJ法といった**質的研究法を利用**する動きがある。

質的研究法によるニーズの把握の例



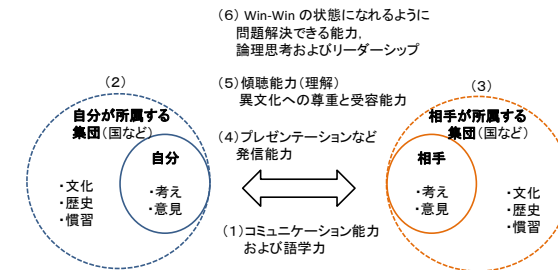
文化人類学、看護学、心理学など、
人文社会系の分野で発展

エスノグラフィー

GTA (Grounded Theory Approach), Modified-GTA

KJ 法

テキストマイニング など



GTA(Grounded Theory Approach)

- Grounded Theory Approach
1967年, グレーザー and ストラウス
The Discovery of Grounded Theory
- GTAの課題意識
Grand Theory \leftrightarrow Grounded on Data
データ (書き起こし文章) に密着した分析 質的研究を科学に
- 切片化 => オープン・コーディング => 選択的コーディング
- ただし、初版本では手順において不明な点が多い。
このことから、のちに様々なバージョンが現れる。

設計手順の概要（順番は前後することあり）

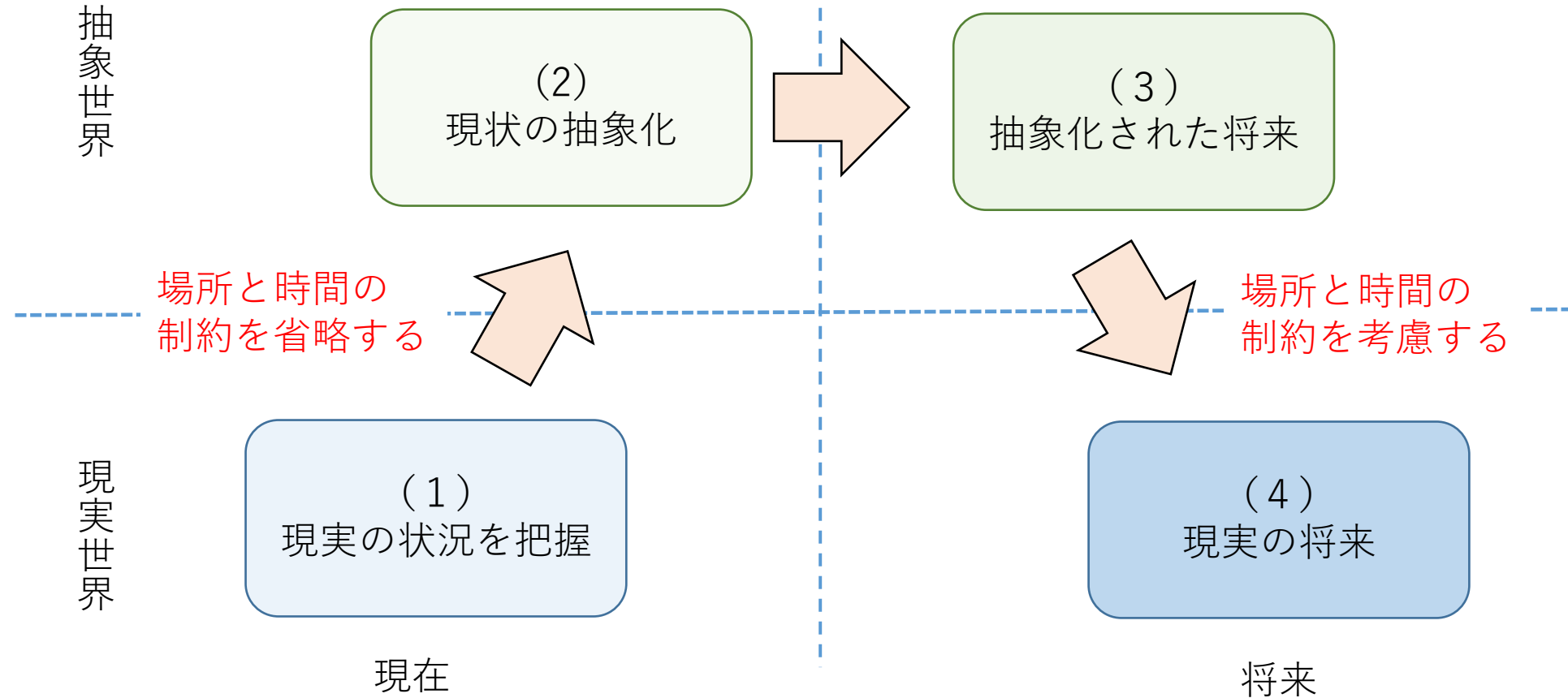
1. 目的、ニーズの明確化（思想も大切）
2. 機能要件の明確化
非機能要件の整理（目標性能、製品寿命など）
3. 現状の把握（存在しない場合あり）
4. 抽象的設計（現状が存在する場合は以下の(1)～(3)）
（1）概念と物体の分離、（2）時空等の制約解除、（3）重複、順序の見直し
5. 時空を考慮した物理設計
6. 開発すべき技術の整理、アイデアの創出
7. 安全性、信頼性、品質、リスクによる見直し
8. 量産、販売、修理、回収、廃棄の計画による見直し
9. モジュール分割インターフェースとパラメータ設計
10. 詳細設計へ

モデル化技法
設計図面、
UML
業務フローチャート

設計図（モデル図）の進化

理想的な新しいフローを設計する

- ・ 例えば、重複するデータの流れや処理を省略する
- ・ 順序を検討する



設計手順の概要（順番は前後することあり）

1. 目的、ニーズの明確化（思想も大切）
2. 機能要件の明確化
非機能要件の整理（目標性能、製品寿命など）
3. 現状の把握（存在しない場合あり）
4. 抽象的設計（現状が存在する場合は以下の(1)～(3)）
（1）概念と物体の分離、（2）時空等の制約解除、（3）重複、順序の見直し
5. 時空を考慮した物理設計
6. 開発すべき技術の整理、アイデアの創出
7. 安全性、信頼性、品質、リスクによる見直し
8. 量産、販売、修理、回収、廃棄の計画による見直し
9. モジュール分割インターフェースとパラメータ設計
10. 詳細設計へ

四原因説を念頭においた整理

• ソクラテス・プラトン・**アリストテレス** 「自然学」

• 現象について4種類の原因を検討

• 質量因 (Causa materialis)

木

• 形相因 (Causa formalis)

形、デザイン

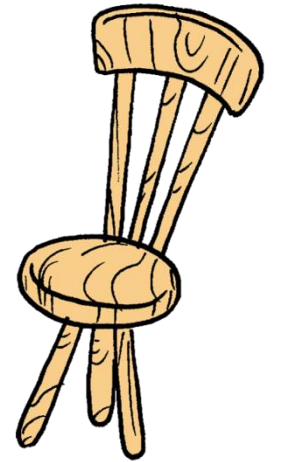
• 作用因 (Causa efficiens)

のみで彫る (大工仕事)

• 目的因 (Causa finalis)

座る

運動変化の原因



設計手順の概要（順番は前後することあり）

1. 目的、ニーズの明確化（思想も大切）
2. 機能要件の明確化
非機能要件の整理（目標性能、製品寿命など）
3. 現状の把握（存在しない場合あり）
4. 抽象的設計（現状が存在する場合は以下の(1)～(3)）
（1）概念と物体の分離、（2）時空等の制約解除、（3）重複、順序の見直し
5. 時空を考慮した物理設計
6. 開発すべき技術の整理、アイデアの創出 **デザイン思考、TRIZ（トリーズ）など**
7. 安全性、信頼性、品質、リスクによる見直し
8. 量産、販売、修理、回収、廃棄の計画による見直し
9. モジュール分割インターフェースとパラメータ設計
10. 詳細設計へ

設計手順の概要（順番は前後することあり）

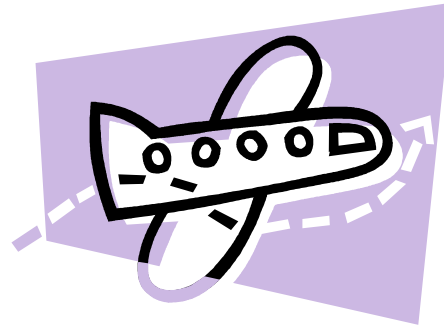
1. 目的、ニーズの明確化（思想も大切）
2. 機能要件の明確化
非機能要件の整理（目標性能、製品寿命など）
3. 現状の把握（存在しない場合あり）
4. 抽象的設計（現状が存在する場合は以下の(1)～(3)）
（1）概念と物体の分離、（2）時空等の制約解除、（3）重複、順序の見直し
5. 時空を考慮した物理設計
6. 開発すべき技術の整理、アイデアの創出
7. 安全性、信頼性、品質、リスクによる見直し
8. 量産、販売、修理、回収、廃棄の計画による見直し
9. モジュール分割インターフェースとパラメータ設計
10. 詳細設計へ

フェイル・セーフ

エレベーター



飛行機



ハドソン川の奇跡 (2009年)

「わたしは、やるべきことをやったまでだ。訓練を怠らないこと、乗客を守る
こと、

すべてはパイロットの義務である」

「誰がなんと言おうと、あれはチームプレイだった」 サリー機長

中国の高速鉄道事故 (信号故障の場合 [YouTube](#)) 閉塞区間

原子炉を止めるためにはエネルギーが必要

フール・プルーフ (バカよけ)

- 人はときどき、おろかなミスを犯す。
それは、ごく自然なあたりまえのこと。

例：病院のベッド

酸素と吸引のプラグ

例：鉄道（単線の線路）

タブレット交換

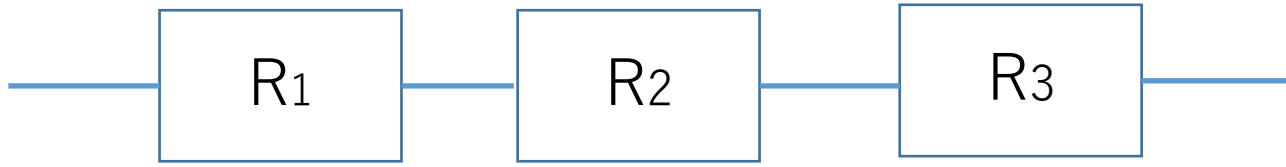
ばかな行動をしても、それが致命的失敗にならないように設計する。

システムの信頼度

- 機械をはじめ、多くのものは、複数の部品から出来ている。
- タイヤが脱落すると走れない。
- でも、ラジオが壊れても、走行に影響はない



- タイヤ 4 本
- タイヤ 1 本の信頼度は、98%である場合
- 車全体での信頼度は、いくらになるか？

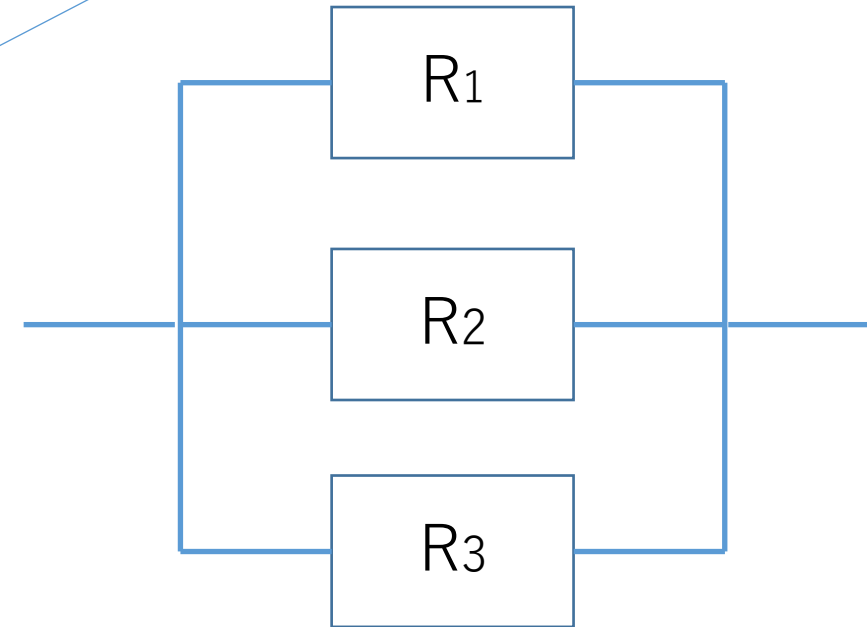


$$R_s = R_1 \times R_2 \times R_3$$

S : series

直列

並列



$$R_p = 1 - (1 - R_1) \times (1 - R_2) \times (1 - R_3)$$

P : parallel

FTA (Fault Tree Analysis)

- 特徴 トップダウンに考える
FMEA は、ボトムアップに考える。
予防すべき故障事象が分かっている場合に、その発生頻度を計算する方法。
- 1961年 ミニットマンミサイルの発射制御システムの開発に活用 (ベル研究所の協力で)
- 1974年 スリーマイル島 原子力発電所事故
FTAを利用した詳細な安全性解析
- 分析の過程で、発生頻度を下げる工夫も合わせて行わなければ意味がない。

設計手順の概要（順番は前後することあり）

1. 目的、ニーズの明確化（思想も大切）
2. 機能要件の明確化
非機能要件の整理（目標性能、製品寿命など）
3. 現状の把握（存在しない場合あり）
4. 抽象的設計（現状が存在する場合は以下の(1)～(3)）
（1）概念と物体の分離、（2）時空等の制約解除、（3）重複、順序の見直し
5. 時空を考慮した物理設計
6. 開発すべき技術の整理、アイデアの創出
7. 安全性、信頼性、品質、リスクによる見直し
8. 量産、販売、修理、回収、廃棄の計画による見直し
9. モジュール分割インターフェースとパラメータ設計
10. 詳細設計へ

企業活動

顧客・利用者
(組織の外部)

ユーザー


⑤ 使用する

企業・生産者
(組織内部)

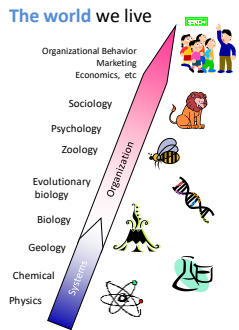
① 観察する
(ニーズの把握)

③ **設計**する

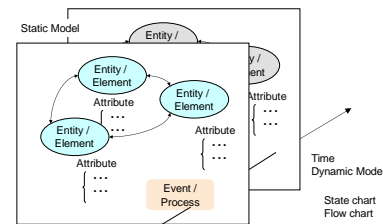
④ 実現する

⑥ 再利用する 

⑦ 評価する



② 分析・モデル化する



- ◇ 単品生産と大量生産
※設計に基づいて1個しか作らない、
個別・単品生産は、プロジェクト
- ◇ 質量有りの場合と
質量無しの場合、その混合

設計手順の概要（順番は前後することあり）

1. 目的、ニーズの明確化（思想も大切）
2. 機能要件の明確化
非機能要件の整理（目標性能、製品寿命など）
3. 現状の把握（存在しない場合あり）
4. 抽象的設計（現状が存在する場合は以下の(1)～(3)）
（1）概念と物体の分離、（2）時空等の制約解除、（3）重複、順序の見直し
5. 時空を考慮した物理設計
6. 開発すべき技術の整理、アイデアの創出
7. 安全性、信頼性、品質、リスクによる見直し
8. 量産、販売、修理、回収、廃棄の計画による見直し
9. モジュール分割、インターフェースとパラメータ設計
10. 詳細設計へ

モジュール間結合度

1. 内容結合
2. 共通結合
3. 外部結合
4. 制御結合
5. データ結合

設計手順の概要（順番は前後することあり）

1. 目的、ニーズの明確化（思想も大切）
2. 機能要件の明確化
非機能要件の整理（目標性能、製品寿命など）
3. 現状の把握（存在しない場合あり）
4. 抽象的設計（現状が存在する場合は以下の(1)～(3)）
（1）概念と物体の分離、（2）時空等の制約解除、（3）重複、順序の見直し
5. 時空を考慮した物理設計
6. 開発すべき技術の整理、アイデアの創出
7. 安全性、信頼性、品質、リスクによる見直し
8. 量産、販売、修理、回収、廃棄の計画による見直し
9. モジュール分割インターフェースとパラメータ設計
10. 詳細設計へ

設計の対象

- 設計手順の概要 (How to)
- 次に、何を設計するのか？ (What)

業務（行政の業務） すなわち、業務知識が必須

そのために、
Zachman フレームワーク, Enterprise Architecture

Zachman, John A. (1987). "A Framework for Information Systems Architecture". IBM Systems Journal. 26 (3). IBM Publication G321-5298.

