

# 情報システム開発の流れ

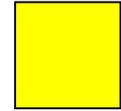
(情報システム開発論、第2回講義)

URL <http://homepage3.nifty.com/suetsuguf/>

Email [fwhy6454@mb.infoweb.ne.jp](mailto:fwhy6454@mb.infoweb.ne.jp)

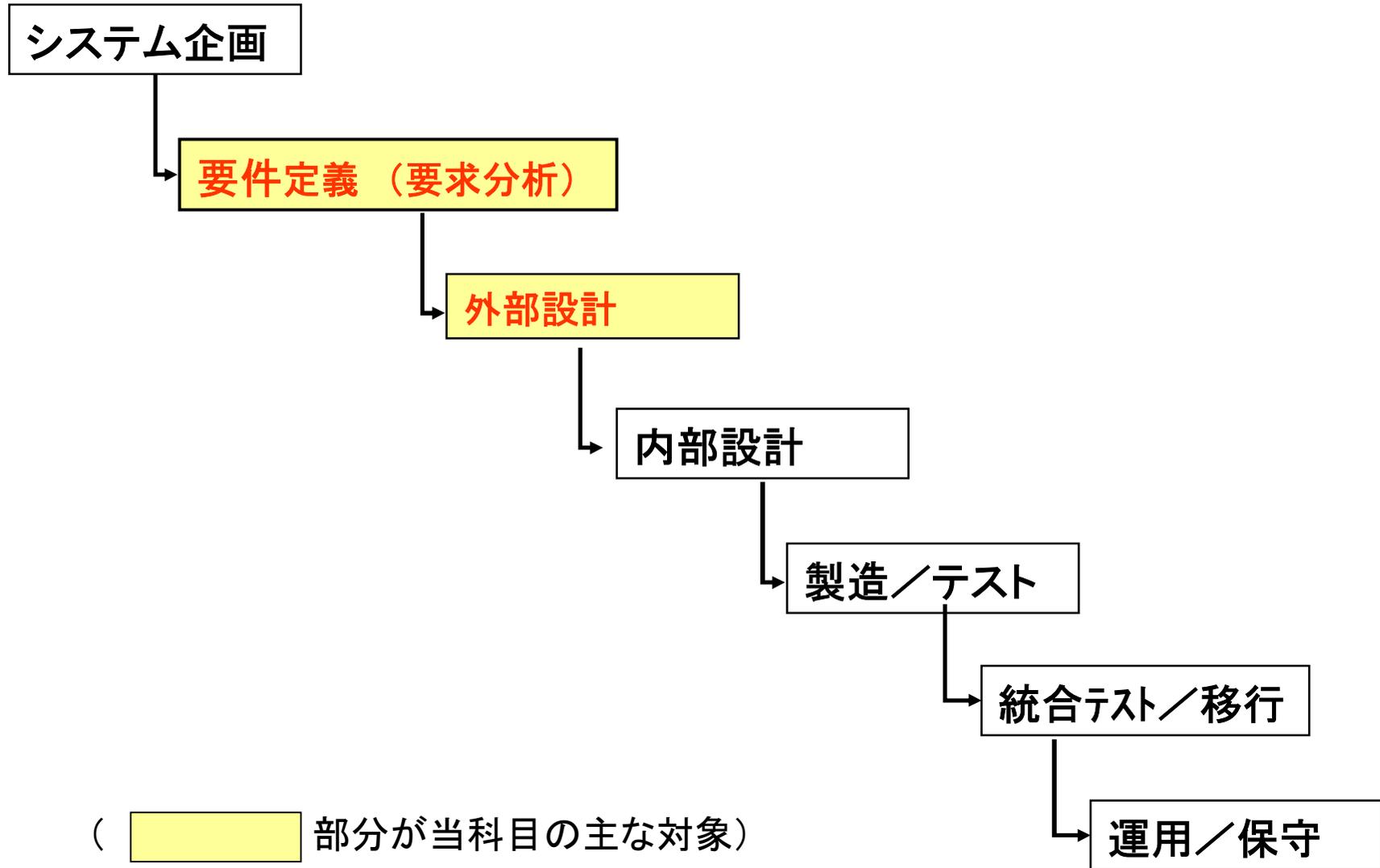
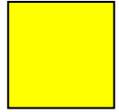
作成者 末次 文雄 ©

# 復習：情報システム開発とは



- システムが対象とするもの
  - 人、組織の活動-----当科目の対象
  - 機械の制御-----電子/機械工学との境界
- 人、組織の活動(例)
  - 主活動 (開発、購入、製造、物流、販売)
  - 支援業務(経営、財務、人事、システム)
- Make / Buy / アウトソーシング
- ユーザー部門との共同作業

# 復習：情報システム開発の全体プロセス



# 復習：情報システム開発に必要なスキル

## 技術・知識

- －ITスキル（情報処理技術、開発技術）
- －管理技法（分析技術、コミュニケーション技法）
- －業務知識
  - ・主活動（開発、購入、製造、物流、販売）
  - ・支援業務（経営、財務、人事、システム）

(株)アイテック

<https://cds01.knowledgewing.com/itec/exam/itec2004hs.asp?corporate=1130sch>

# 目次(情報システム開発の流れ)

1. 情報システムの位置付け
2. システム開発方法の種類
3. システム開発プロセス
4. プロジェクト管理
5. 必要なITテクノロジー
6. 必要な分析技法
7. まとめとレポート課題

# 1. 情報システムの位置付け

① 改革の目的、目標（市場創造、利益拡大、効率UP）

② 業務革新（新商品、ビジネスモデル、生産性向上）

③ 新システム  
（開発、購入、アウトソース）

④ 新業務  
（組織、異動、導入教育）

⑤ システム構築  
（構築技法、開発、PM）

⑥ 業務構築  
（役割分担、業務フロー）

⑦ 情報技術

⑧ 分析技法

⑨ 業務知識

## 2. 情報システム開発方法の種類

1. 設計手法に着目
2. 開発プロセスに着目
3. 実装時の配置に着目
4. システム規模に着目
5. 新規／改造に着目
6. 処理特性に着目
7. 企業統合に着目
8. 使用技術に着目

# 開発方法の違い

## 1. 設計手法に着目

データ 中心	<ul style="list-style-type: none"><li>・データの流れと分析に重点をおいて進める。(DFD、ER図、データ正規化、構造化設計)</li></ul>	主流。 DBは今後も主流
オブジェクト 指向	<ul style="list-style-type: none"><li>・属性と処理(=データと操作)を一体化して設計(カプセル化)</li><li>・ソフトの部品化を進め易い。</li></ul>	GUIから始まり、 適用拡大

## 2. 開発プロセスに着目

ウォーターフォール方式	<ul style="list-style-type: none"><li>・上流から下流までを順を追って開発する方式。(主流)</li></ul>	レビューが必須
プロトタイピング方式	<ul style="list-style-type: none"><li>・部分から試作し、実地検証しながら、繰り返す。(スパイラル)</li></ul>	工程内での適用

# その他の開発方法の特徴

3. 実装の配置	<ul style="list-style-type: none"><li>・ホスト中心</li><li>・2層、3層</li></ul>	(DB管理、処理、ユーザーインターフェース)
4. 規模	<ul style="list-style-type: none"><li>・一括開発(中小規模)</li><li>・段階開発(大規模)</li></ul>	(フェーズ分け)
5. 新規／改造	<ul style="list-style-type: none"><li>・新規開発</li><li>・改造型開発</li></ul>	
6. システム特性	<ul style="list-style-type: none"><li>・リアルタイム処理</li><li>・バッチ処理</li></ul>	
7. 企業統合	<ul style="list-style-type: none"><li>・一社のシステムへ統合</li><li>・ブリッジ方式</li><li>・新規システム開発</li></ul>	(S/W、H/W)
8. 使用技術	<ul style="list-style-type: none"><li>・低水準、高水準言語</li><li>・オブジェクト指向言語</li></ul>	

## 3. システム開発プロセス

- 3. 1 開発プロセスの標準化
- 3. 2 開発プロセス
- 3. 3 各工程の概要
- 3. 4 各工程の詳細
- 3. 5 開発プロセスの改善

## 3. 1 開発プロセスの標準化

### ①開発プロセス標準化以前の問題点

- ・システム仕様決定に期間がかかり過ぎる
- ・開発中に仕様変更が多く発生する
- ・開発工程上の手戻りが多く発生する
- ・種々のドキュメントが多すぎる
- ・システム仕様モレ、バグが残る
- ・予定以上の工数がかかる
- ・システムの完成が遅れがち
- ・システム保守が容易でない

## ②原因

- ・目標の周知徹底が不足
- ・機能定義および、その確認が不十分
- ・データ定義が徹底されていない
- ・プロジェクト参加者のコミュニケーションが不足
- ・作業手順、作業方法が不明確
- ・ドキュメンテーションルールがあいまい
- ・作業計画が十分明らかにされていない
- ・進捗管理、品質管理が不十分

### ③解決策

「システム開発工程、管理に標準ルールを取り入れる」

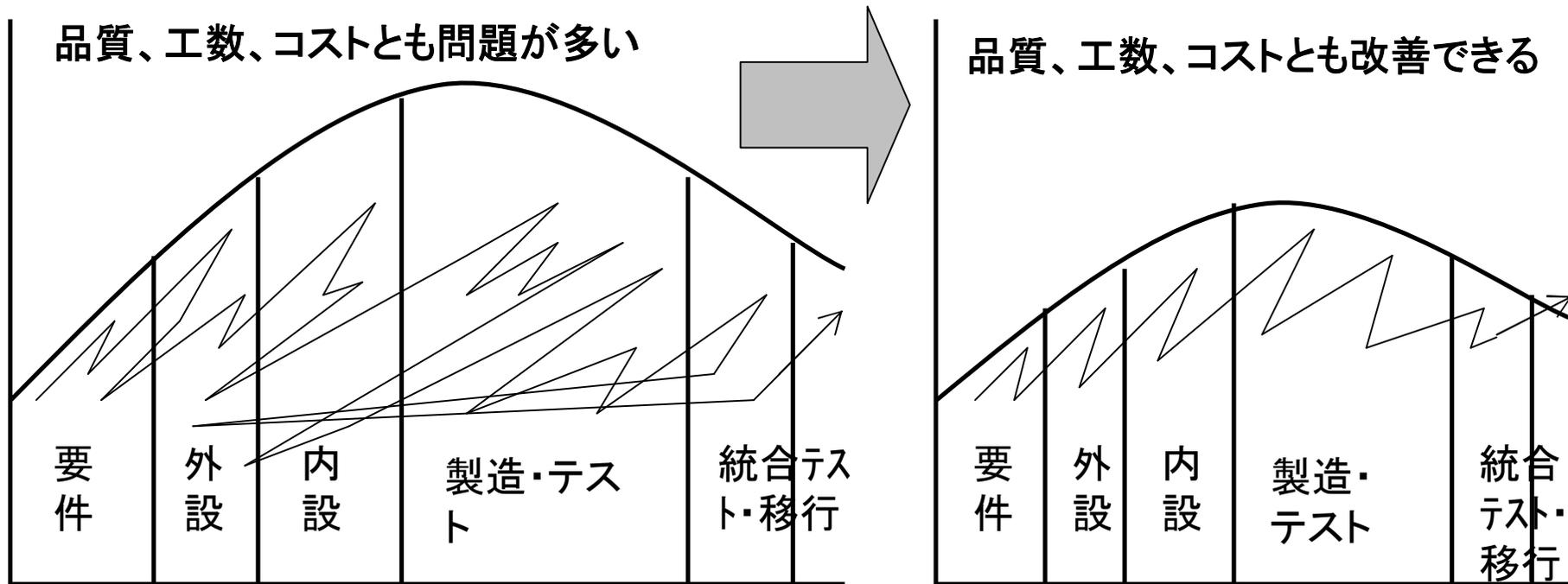
- ・確立した各種手法の取り入れ
  - －設計手法を決める  
(分析・設計は〇〇、データ分析はER手法とか)
  - －設計プロセスの各工程の内容・期間を決める
- ・プロジェクト管理手法を取り入れる
  - －納期・品質・コストの管理方法
- ・プロジェクト運営ルールを定める
  - －システム化の目的、目標を再確認
  - －役割分担、コミュニケーション方法
  - －ドキュメンテーション・ルール(=成果物の文書化)
  - －変更管理方法、ネーミングルール

## ④開発プロセス標準化の効用

- 誰が担当しても、一定水準のシステムを開発することが出来る。

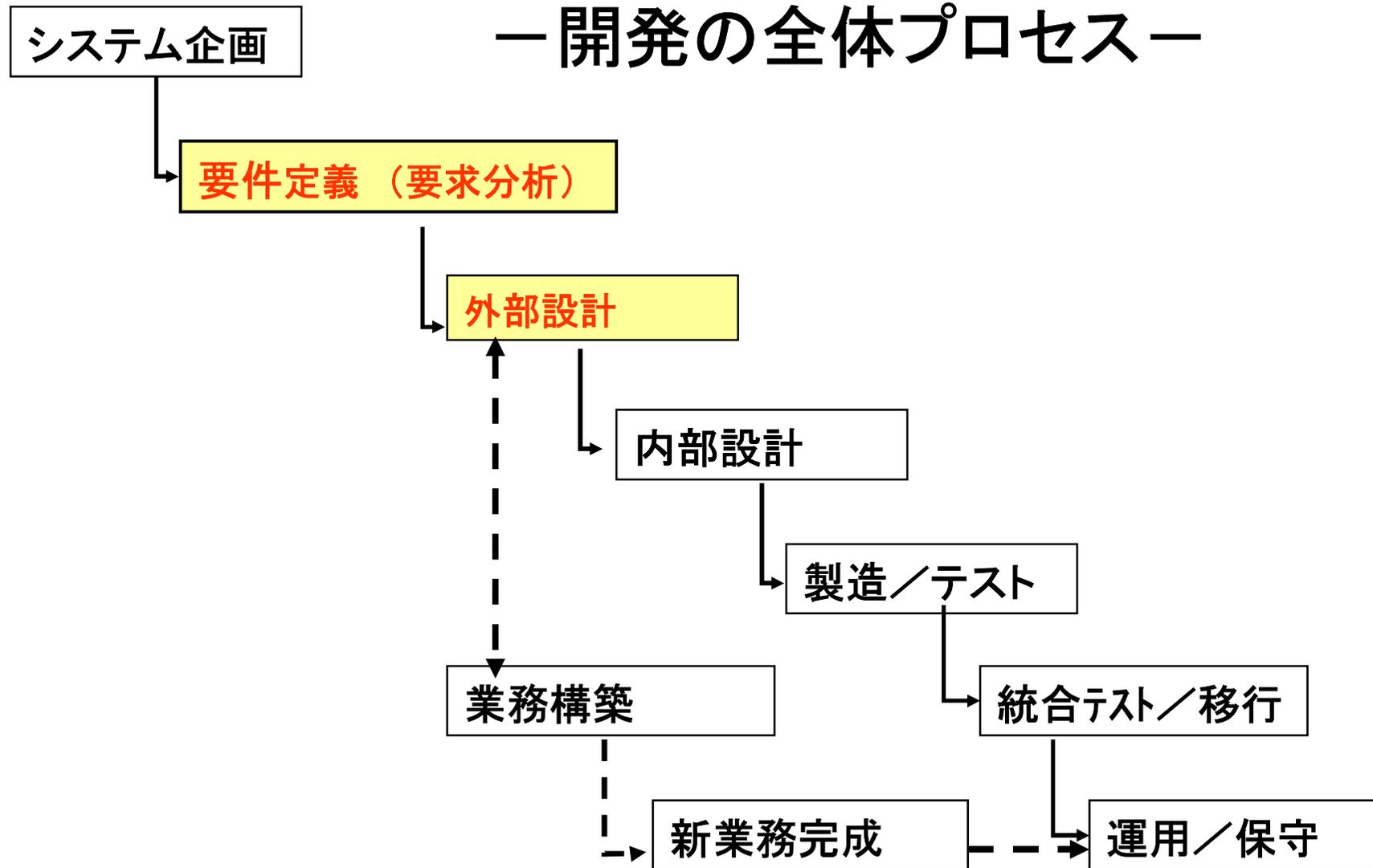
ームラ、ムダ、ムリが少なくなり、

漏れ、誤解が減少し品質が上がる



# 3.2 開発プロセス

—開発の全体プロセス—



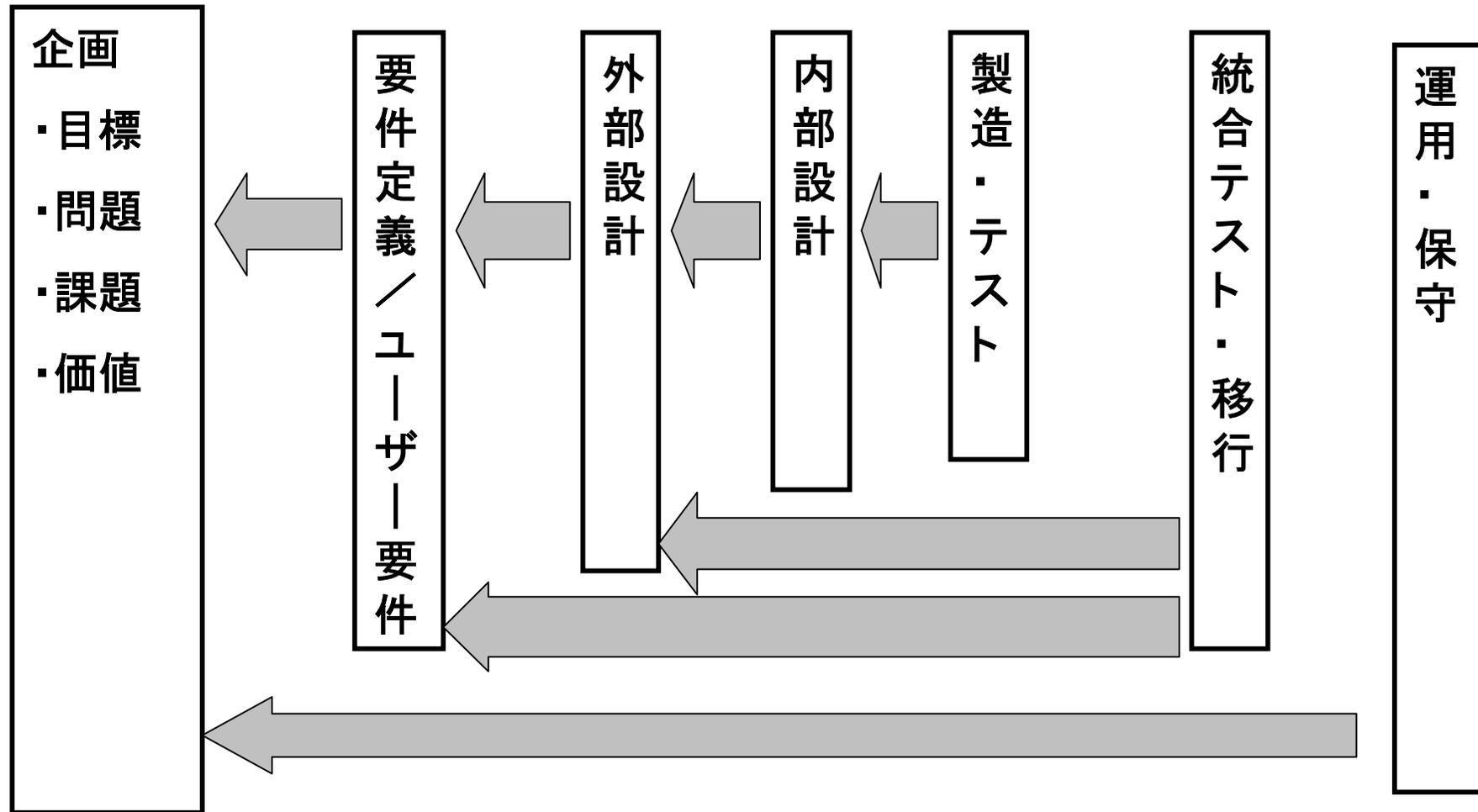
(  部分が当科目の主な対象)

# ー開発プロセスのポイントー

	工程のポイント
システム企画	・何のためにどういうシステムが必要かを提案し承認を得る
要件定義（要求分析）	・具体的に何がやりたいかをまとめて、かつ実現可能性を検証する
外部設計	・ユーザーの立場に立って、必要な仕様を決める（＝ユーザーマニュアルの完成に等しい）
内部設計	・実装レベルの仕様を全て決定
製造／テスト	・上記に基づいて、実装する
統合テスト／移行	・本番並みのテスト、ユーザー承認
運用／保守	・運転し、かつシステム育成

# — 各工程の対象 —

各工程は、矢印で示す上流工程の各要件を、満足している必要がある。



# 補足：建築との対比

<システム開発>	<住宅の建築>
企画	プラン(目的、希望、問題解決、敷地、資金…)
要件定義	施主の要求まとめ(用途、部屋数、内装、予算…)
外部設計	基本設計(外観、間取り、平面図、模型、材質、収納、設備、照明、コンセント位置…)、見積り
内部設計	実施設計(強度設計、施行設計、配線、配管…)
製造・テスト	施工(基礎、柱立て、内装、配線…)、設計監理
統合テスト、移行	検査申請、建築確認、引越し
運用・保守	点検、クレーム、補修、増改築

### 3.3 各工程の概要

	仕事の内容	OO方式
システム企画	・目的、方針、ビジネスモデル、機能、構造、効果、予算	同じ
要件定義 (要求分析)	・要求の調査／分析、範囲、要求仕様まとめ(機能・DB・品質)、実現性、費用、計画作成	要求モデル
外部設計	・機能、DB、I/O、構造を決定	分析モデル
内部設計	・上記の物理モデル(実装レベル)	設計モデル
製造／テスト	・プログラム製造、テスト	同じ
統合テスト／移行	・本番並みのテスト、ユーザー承認、本番移行	同じ
運用／保守	・運転、監視、保守(改善・バグ修正・トラブル予防)	同じ

## 3.3 各工程の詳細

### ①システム企画

- 内容 : 情報化戦略に基き、重点となる個別のシステム化を計画する。  
: システム化の目的、目標、対象範囲、システム構造、主要機能、C/O時期、予算、効果を検討。
- 成果 : システム化企画書
- その他 : システム計画ともいう

## ②要件定義

- 内容：現状を調査し問題点、課題を分析する。  
：その解決策の検討、および、システム  
要求を検討。  
：次いでシステム化要件を分析し、優先順位  
付けをし、目標・期待効果・費用・リスクを勘案して  
システム化範囲、フェーズ分けを決定。  
：機能・データ・品質に関し要求仕様としてまとめる。  
：要員、技術、費用、納期の実現可能性を検証。
- 成果：機能一覧、DB概念図、使用技術、開発計画
- その他：要求分析、システム分析ともいう

### ③外部設計

- ・内容：システム機能構造を具体化する。  
（ユーザー視点で下方展開）
  - ：下方展開された機能を概念DB・データ項目に  
関連付けて、処理機能をデータ利用面から  
明確にする。
  - ：I/O画面、帳票を決定する。
  - ：DBのデータ項目の追加、補強。
  - ：内部設計移行の開発計画の見直し。
- ・成果：外部機能一覧、論理DB図、業務フロー、  
I/O画面・帳票仕様、移行・運用要件、  
（上記は、ユーザーマニュアルのもとネタとなる）
- ・その他：概要設計、基本設計、機能設計ともいう

## ④内部設計

- ・内容
  - : 外部機能を投影した形で、どのような内部機能構造で実現するかについて検討。
  - : 最下位レベルであるプログラム単位にまで機能展開する。
  - : さらにプログラム機能を、単機能にまでモジュール分割し、モジュール機能を決定する。
  - : 処理効率を検討してDB、ファイル物理構造を決定。
  - : 製造工程の開発計画の見直し。
- ・成果
  - : プログラム／モジュール機能一覧、物理DB図、ファイル構造、**内部仕様書**
- その他
  - : 詳細設計、技術設計、プログラム構造設計ともいう

## ⑤製造～運用・保守

	内容	成果物
製造・テスト	<ul style="list-style-type: none"><li>・プログラム、画面の作成</li><li>・単体テスト</li><li>・モジュール結合テスト</li><li>・H/W、ミドルソフト導入</li></ul>	<ul style="list-style-type: none"><li>・プログラム</li></ul>
統合テスト・移行	<ul style="list-style-type: none"><li>・システムテスト</li><li>・ユーザー承認テスト</li><li>・移行(システム、データ、ドキュメント)</li></ul>	<ul style="list-style-type: none"><li>・テスト結果報告</li><li>・移行結果報告</li></ul>
運用・保守	<ul style="list-style-type: none"><li>・運転計画、運転、監視</li><li>・保守(予防、改善、バグ)</li></ul>	<ul style="list-style-type: none"><li>・運転監視記録</li></ul>

# 補足： システム化のポイント

- 目的が明解：あいまいさを残さない
- 段階分け：一番必要とされる機能から開発する。
  - ：対象領域は、面で捉える。
  - （現状のやり方を残さない）
- 他社と差別化：同じことをやっているには負ける
  - ・クロネコヤマト、セブンイレブン
- 有益である：利用者が明らかに利益を得る
  - ・オンライン証券会社
  - ・ネット・オークション

## 3.5 開発プロセスの改善

### ①RAD方式 (Rapid Application Development)

－短納期、迅速開発

- －特徴
- ・ユーザー責任者が常時参加し、すばやく仕様決定を行う
  - ・少数精鋭(2～4名)、高度スキル
  - ・段階的で、追加型開発(6ヶ月内)
  - ・ツールの活用(自動プログラミング等)

(ユーザーインターフェース部分が多いWebシステムが適している。但し、要件定義、外部設計、DB設計を省略することはない。)

## ②見積りの精度向上

### ー従来からの見積り方法

- ・システム規模、生産性をもとに、  
開発期間、費用を見積る  
(ベースになるのは、経験から割り出したプログラム本数・サイズ、1本当り工数)

### ーFP法---Function Point(機能数)

- ・ユーザーに提供するシステム機能をカウントし、  
その合計を求めるやりかた
- ・カウント対象は、入力数・出力数・データ  
項目数など
- ・システムの特長、処理複雑さ度合いなどを  
加味する

(採用企業が増えれば、標準になる可能性がある)

# ③その他の改善項目

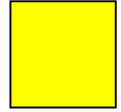
## ー技術標準の整備

- ・新技術のための開発標準の作成
- ・ソフト部品の品揃え(サブルーチンなど)
- ・技術教育の充実

## ー開発プロセスの改善

- ・ドキュメントの種類削減、様式の簡素化
- ・開発に必要な情報の共有化
- ・ツールの導入(自動プログラミング、テスト支援・・・)
- ・プロジェクトレビュー用資料の削減、兼用化 など

# 補足：仕様書の構成記号



- ドキュメント(書類)の構成では、章立ての
- 記号を統一しておく必要がある

例示：

第 I 部 (ローマ数字)

1. (章)

1. 1. (章の構成)

1)

①

イ)

a)

# 4. プロジェクト管理

## 4.1 プロジェクト管理の概要

### ①プロジェクト管理の対象

- ・開発資源（組織、要員、外注先、  
設備、情報、時間・期間、コスト）
- ・開発の方法（開発工程、変更履歴）
- ・成果物（仕様書、プログラムコード、機能、  
性能、品質）

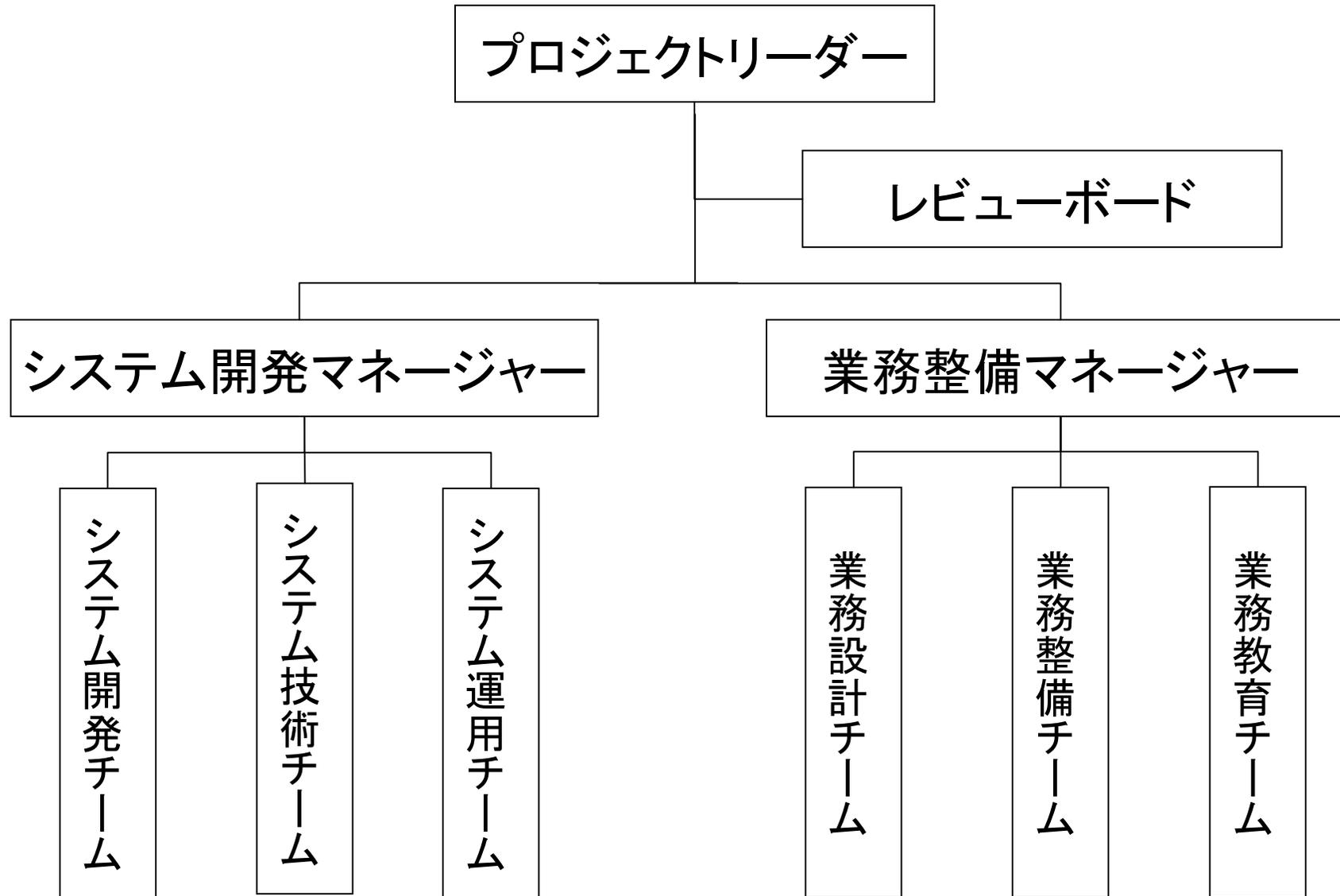
### ②プロジェクト管理の方法

- ・進捗（計画との差、その原因、解決策は）
- ・体制（レビュー体制、レビュー時点）
- ・システム内容（重要点はシステム仕様検証）

## 4.2 経験則

- ・納期を守ることが出来れば、品質・コストは、後からついて来る。
  - 納期厳守に集中
- ・プロジェクト・メンバーが出来ないと考えることは、当たっている。
  - 障害物を除く
- ・いくら早くプログラミングを始めても、システムは早く完成しない。
  - 上流工程が肝要
- ・大半のシステム欠陥は、上流工程で生まれる(要件定義、外部設計、内部設計)
  - 上流工程が肝要
- ・大規模システムは一度に開発しない
  - 分割が有効
- ・ツールの効用は絶大である
  - 広くツールを探し、活用する

# 4.3 プロジェクト体制



# 補足：企業の情報システム開発組織

部門長

計画部門

(システム化計画、予算管理、人事管理、資源管理、人材育成)

企画部門

(システム化のための調査、分析、企画)

開発部門

(システムの設計、製造、テスト、本番移行)

技術部門

(IT技術調査、OS選定、ミドルソフト選定、IT技術標準)

運用部門

(システムの運用、データ入力、保守改善、設備の点検保守)

## 5. 必要なITテクノロジー (例示)

- B2C (Business to Customer)
  - ーセキュリティ (Fire Wall、VPN、SSL、電子証明書・・・)
  - ーユーザーインターフェース (i-mode、ブラウザー・・・)
  - ーサーバー (Webサーバー、アプリケーションサーバー、DBMS、Java、JavaBeans、OS・・・)
- B2B (Business to Business)
  - ーセキュリティ (VPN、SSL、電子証明書・・・)
  - ーデータレベルの連携 (XML、XSL、XSQL・・・)
  - ーインターフェース (FTP、HTTPS、SMTP、メッセージキューイング・・・)
- コラボレーション
  - ーワーフローサーバー、ネットミーティング、メールサーバー
  - ーデータ共有 (DBMS)
- その他要素
  - ーOS (UNIX、Windows-NT、LINUX)
  - ー運営 (リモート障害監視、サーバー集中監視・・・)

## 6. 必要な分析技法 (例示)

### ① アイデアを出す

#### ー アイデア発想法

- つめ込む(ネタ、ルート、媒体)
- 絞り込む(重点、重要)
- 体系化 (関連付け、構造化)
- 追加する(話す、人から)

#### ー 5W3Hで発想する

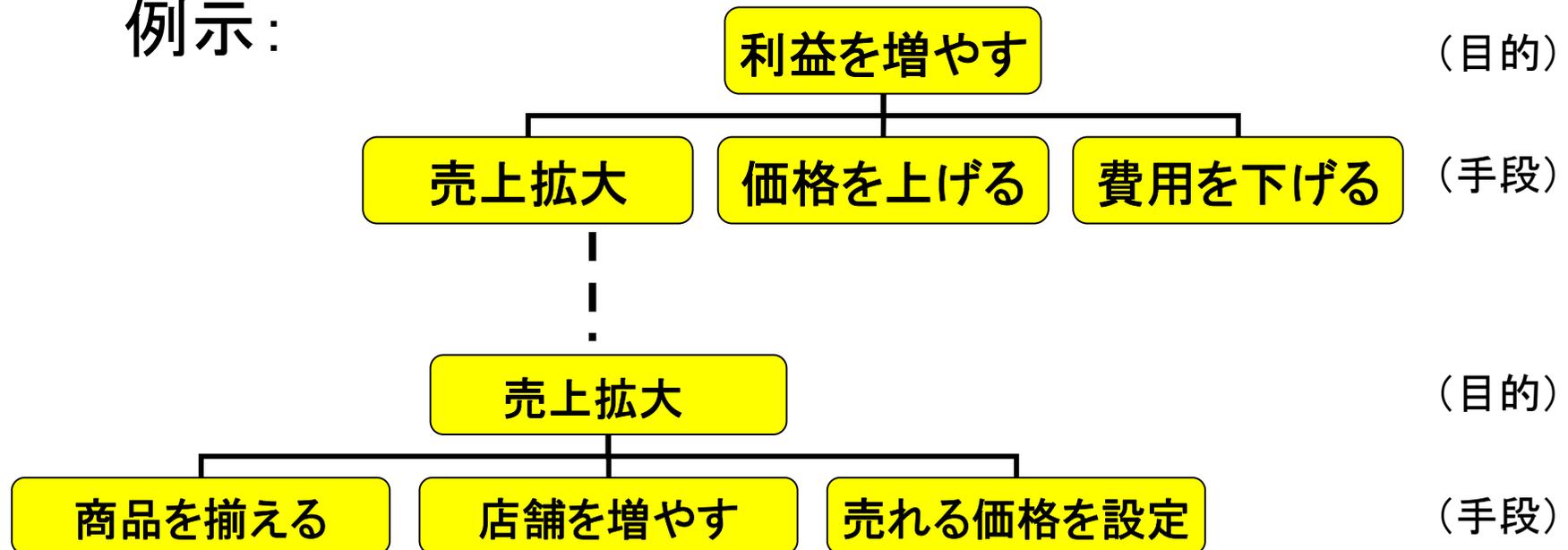
- What、Why、When、Who、Where
- How to
- How much
- How many

## ② 目的発想法----(村上 哲大氏)

- ・目的を決めて、その達成手段を考える。(機能分析)
- ・問題点を見つけて、その原因を探る。(因果分析)

(カードに書く、WHYを繰り返す、原点に帰る)

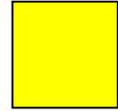
例示:



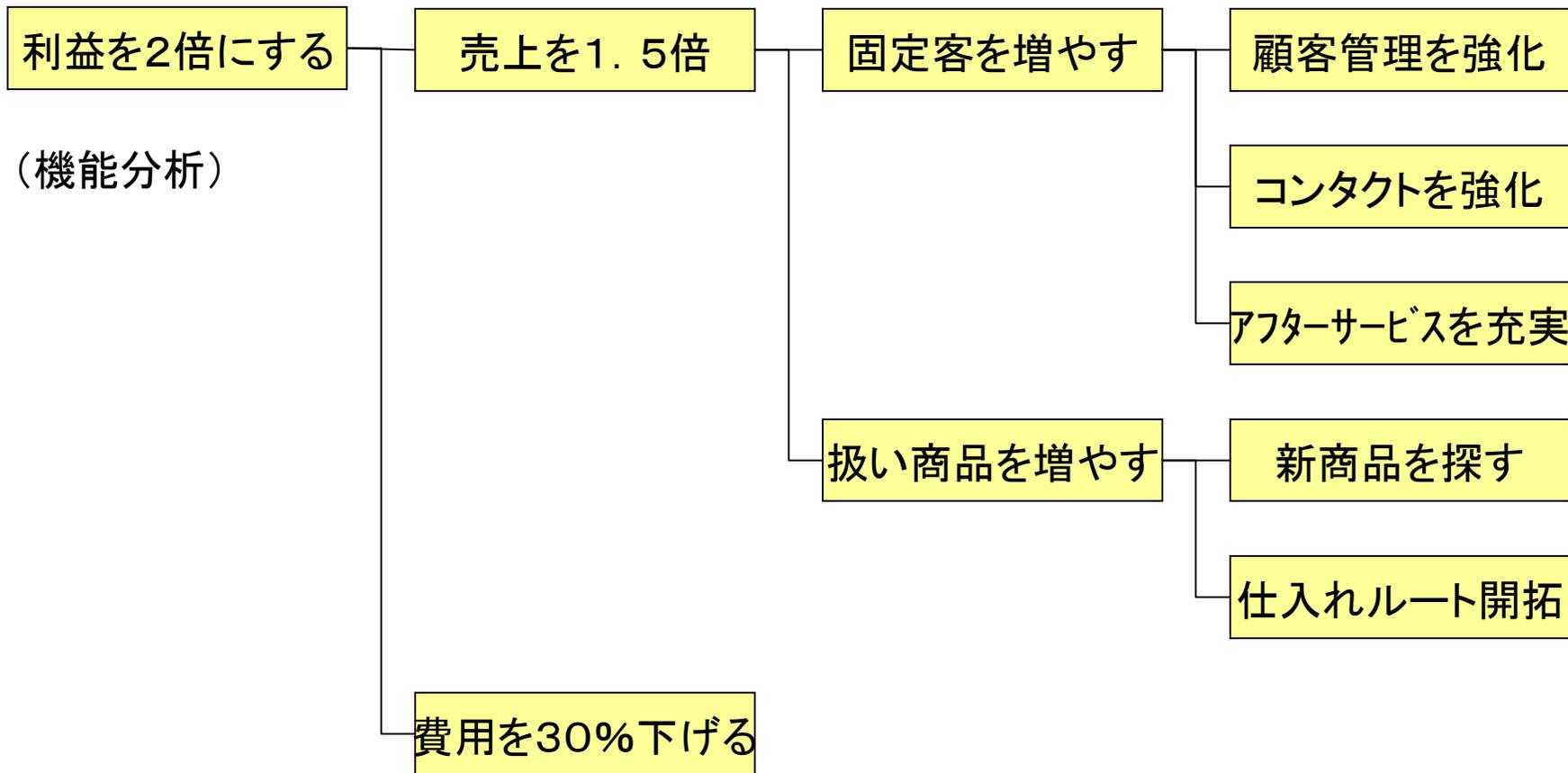
### ③ 下位への展開方法

- 機能に着目（つまり役割に着目）
  - ・XXXが必要、正確さ、迅速さ、安く
- 業務プロセスに着目（順序、手順）
- 資源に着目（人、モノ、金、情報、時間）
- 管理サイクルに着目（Plan-Do-See）
- 現場に着目（数量、品質、コスト、納期・・・）
- 内容の説明型（下位で上位の説明をする）

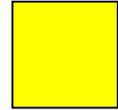
# 例示：機能に着目した下位展開方法



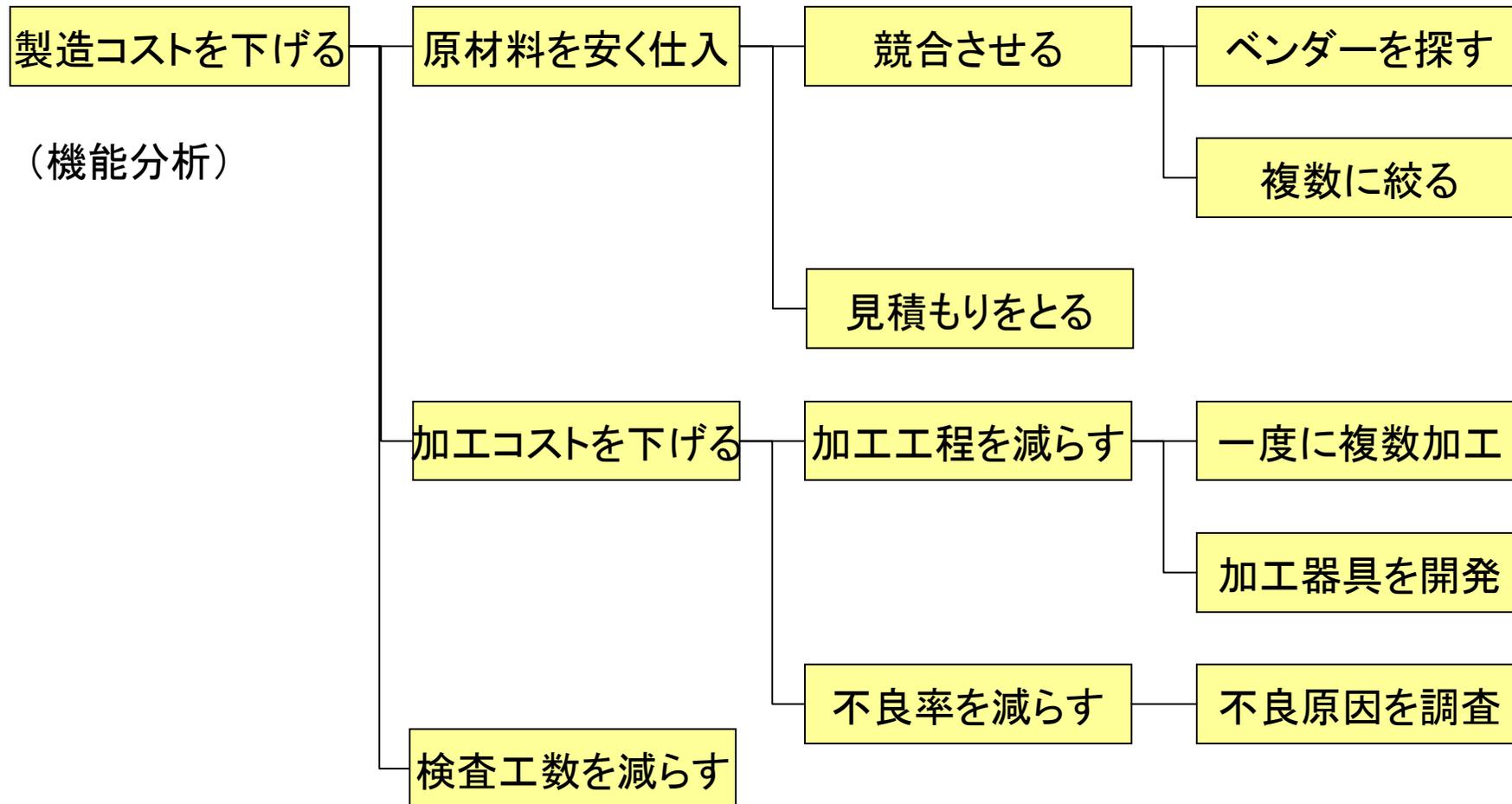
- ・対象とする目的や問題を機能面から分析
- ・機能面では、働き、役割に注目する



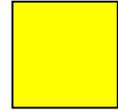
# 例示：業務プロセスに着目した下位展開方法



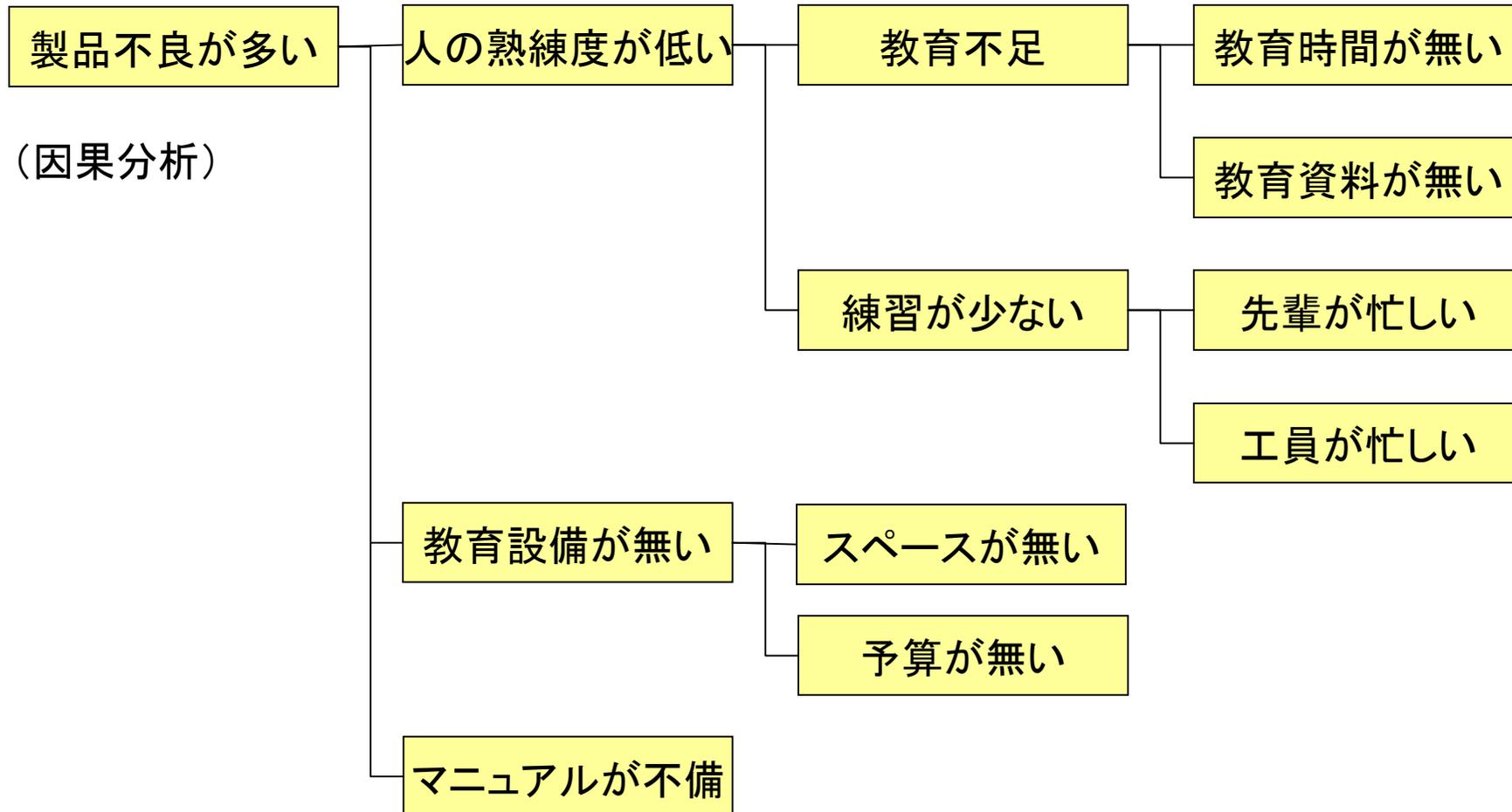
- ・対象とする目的や問題をプロセス面から分析
- ・業務の実施順序、手順に着目する



# 例示：資源に着目した下位展開方法



- ・対象とする目的や問題を資源面から分析
- ・資源面では、人、モノ、金、情報、時間に注目する



# 参考：リテラシ(素養)

## 従来のリテラシ(素養)

- ・読み、書き、そろばん、話す、教養

## 情報技術リテラシ(ITリテラシ)

<狭義> **コンピュータを操作できる**

- <広義> ・情報ネットワークを活用して、
- ・必要な情報を収集・整理・加工・分析し、
  - ・本質をつかんで情報を発信できる能力。
  - ・ITを技術として管理、評価できる能力。

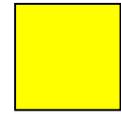
## 仕事のリテラシ

- ・一時には、**一つのことに集中する**
- ・立場を変えて考えてみる力(相手、お客)
- ・人との協調、チームメンバーシップ
- ・考えを紙に書いて整理する
- ・問題解決能力(問題点→原因究明→解決策)
- ・目標達成能力(目的→手段の連鎖発想)

## ベースのリテラシ

- ・**自分の考えを持つ(本を読む＝考える)**
- ・自分は、何がしたいかを考える力
- ・文章表現の能力(構想力、構成、文書化)
- ・うそを見抜く力
- ・他人の権利を侵害しない態度
- ・技術を評価して、何に使うべきかを考える力

# 補足：SEの能力



- ・アプリケーションSEは、上流のシステム設計能力が主
- ・プログラミング能力は無くてもSEが務まる

<p>システム設計 (業務知識、IT専門知識)</p> <p>65%</p>	<p>プログラミング 能力</p> <p>5%</p>
<p>基礎能力 (好奇心、集中力、分析力、概念構成力、人間関係)</p> <p>30%</p>	

# 7. まとめとレポート課題

- 重要項目

- ーシステム開発の、大きな二つの方法(DOA、OO)
- ーシステム開発プロセスと、各工程の内容
- ーシステム開発の5要素  
(IT処理技術、システム開発の技術、プロジェクト管理、  
業務知識、分析技法)

- レポート課題(A4x1、2枚)

- ①システム開発の流れを説明してください。
- ②上記①の中で自分が最も興味がある工程について重要ポイントをあげてください。

期限      次回の授業開始時点

提出      レポート用紙またはメール

# 参考書等

- 参考書 : 中桐 紀幸著  
「即戦UMLモデリング 業種・業務別サンプル集」  
(リックテレコム社、3100円)
- : 布川 薫ほか著  
「SEの基礎知識、アプリケーション開発技術」  
(リックテレコム社、3000円)
- : 河村 一樹著  
「情報システム設計・開発技術」  
(近代科学社、2900円)
- : 村上 哲大著  
「仕事を生かす目的発想法」  
(溪水社、1300円)