

# システム開発の現場

(情報と社会 第8回)

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 ©

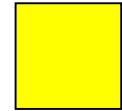
# 復習：インターネット発想の原点

ねらい：戦時下で通信回路の一部が切断されても、  
全体の通信は支障なく動作すること。

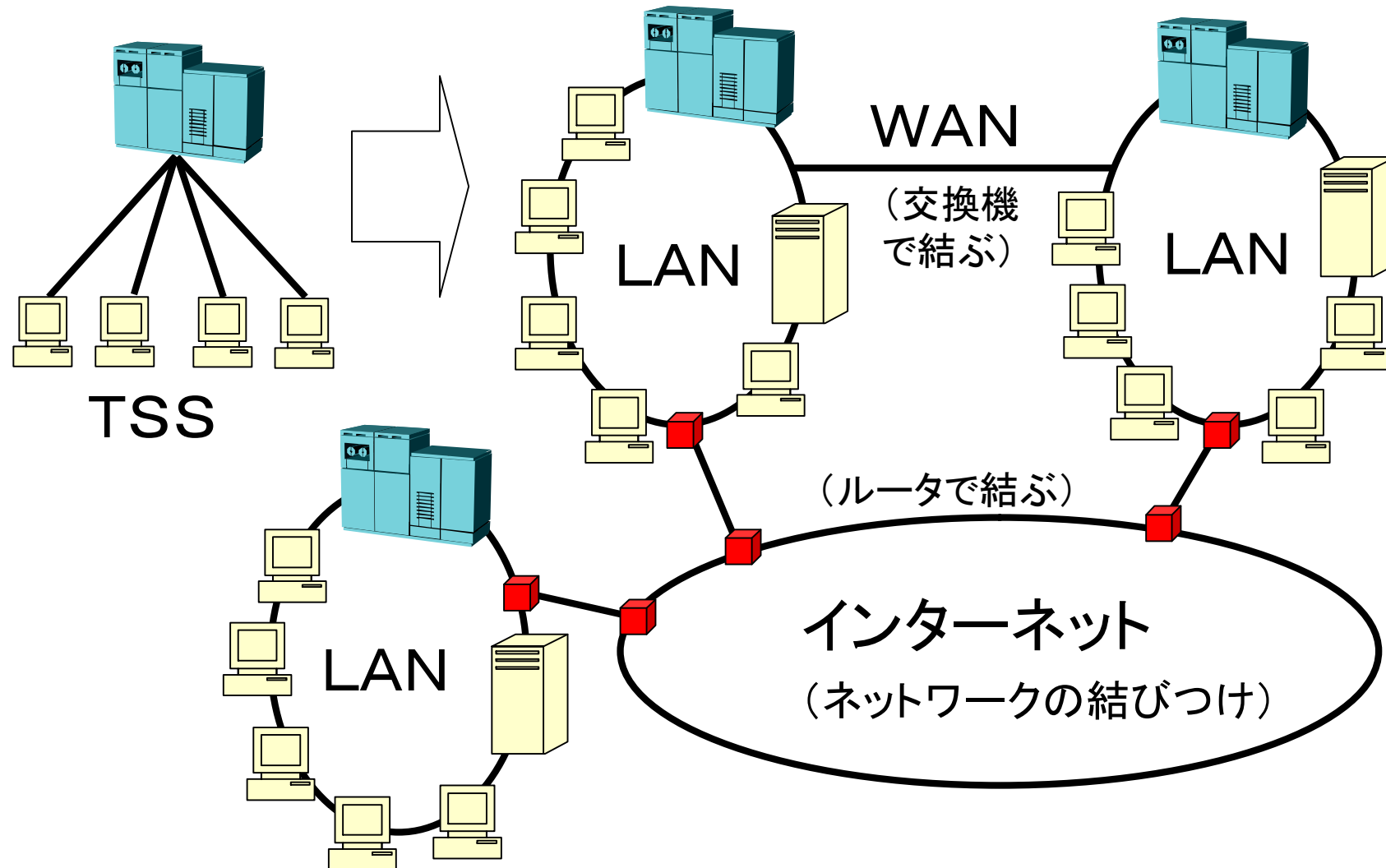
そのためには：

- ・通信路が切れた場合は、別の通信路を使う。  
(ルーティング機能＝通信経路選択)
- ・データ全部を一度に送らず、細切れで送る。  
(パケット通信機能＝効率、機密上で有利)

# 復習： インターネットの定義



- ・世界的規模で、通信ネットワーク同士を結んだもの
- ・通信ネットワークのネットワークとも言う(通信網のひとつ)



# 復習： インターネットの良さ

- 場所が自由 ———— どこにいても利用できる
  - タイミング自由 ———— 好きな時間に利用できる
  - 国境が無い ———— どこへでも行ける
- 
- 網羅性 ———— 幅広く情報を集めることができる  
(専門資料、興味)  
(個人が企業並みにできる)  
(ショッピング、趣味)
  - 速報性 ———— 新聞、テレビ、書物より早い

# 復習： インターネットでできること

## 情報の送付

- ① ファイル転送

## サーバーの利用

- ② 端末からサーバーを使用

## コミュニケーション

- ③ 電子メール
- ④ 電子掲示板
- ⑤ メールングリスト
- ⑥ 電子会議室(ニュースグループ)
- ⑦ チャット
- ⑧ IP電話

## 情報の利用

- ⑨ WWWの利用 (ホームページ、ブログ(Weblog))

# 復習：インターネットの問題点

## ① 確実性

- ・通信障害で止まる(停電、負荷ピーク、故障)
- ・情報伝達が、確実でない(**best effort**)

## ② 安全性、信頼性

- ・道路に面した家と同じで、外部からの侵入がある
- ・システムの破壊、個人情報流出の危険性

## ③ デジタルデバイド

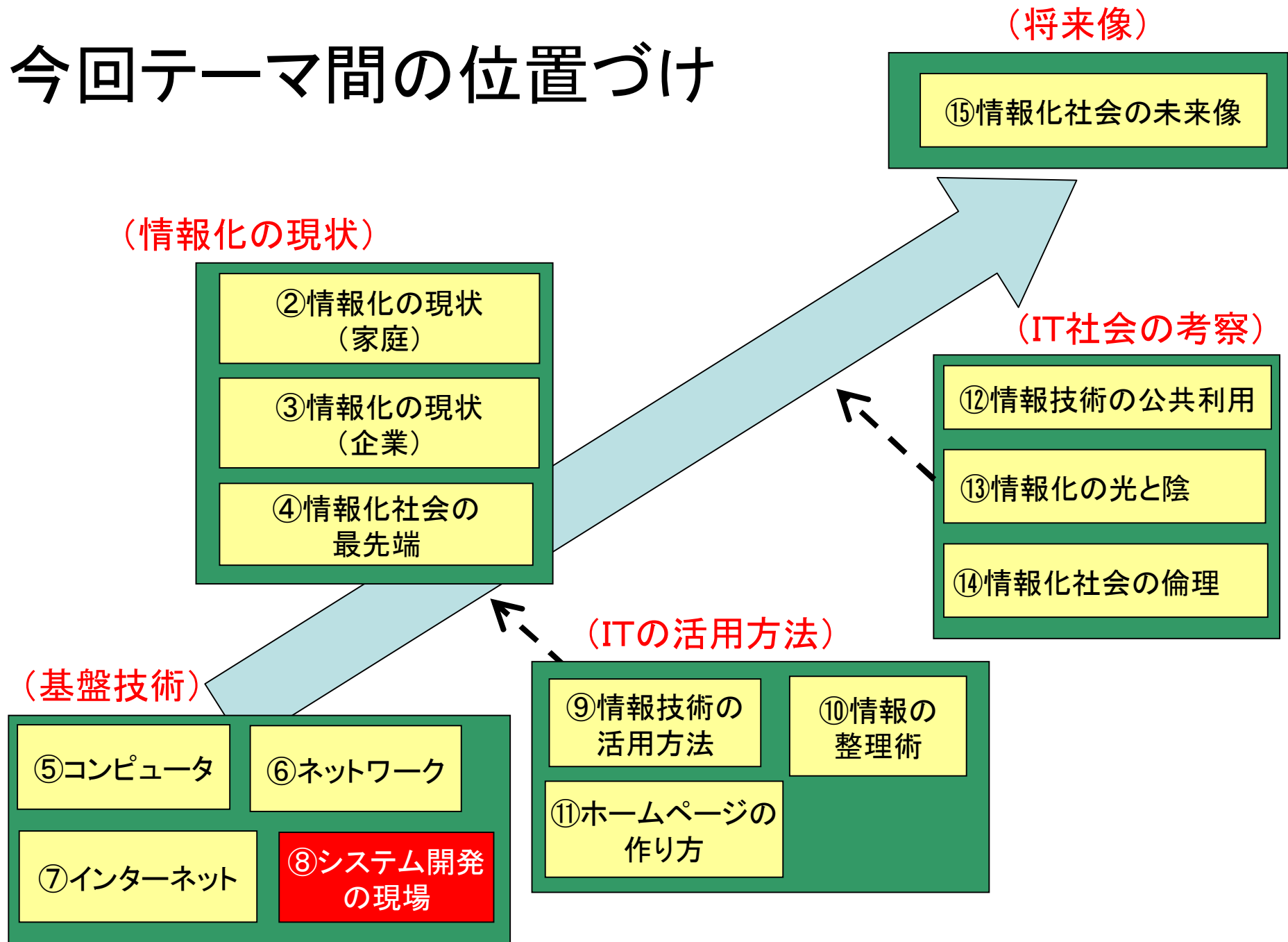
- ・使える人と使えない人の差が大きい
- ・お金がかかる、操作が面倒、英語力が要る

## ④ 悪徳商法

- ・詐欺、掲示板での誹謗中傷、迷惑メール、なりすまし

(詳細は、第12回、「情報化の光と陰」を参照のこと)

# 今回テーマ間の位置づけ



# 目次(システム開発の現場)

1. 誰が何を開発しているのか
2. システム開発のプロセス
3. プロジェクト運営
4. システム開発の問題点
5. SEに必要な素養
6. 分析技法の紹介
7. レポート課題
8. 参考書、参照Webサイト



# 1. 誰が何を開発しているのか

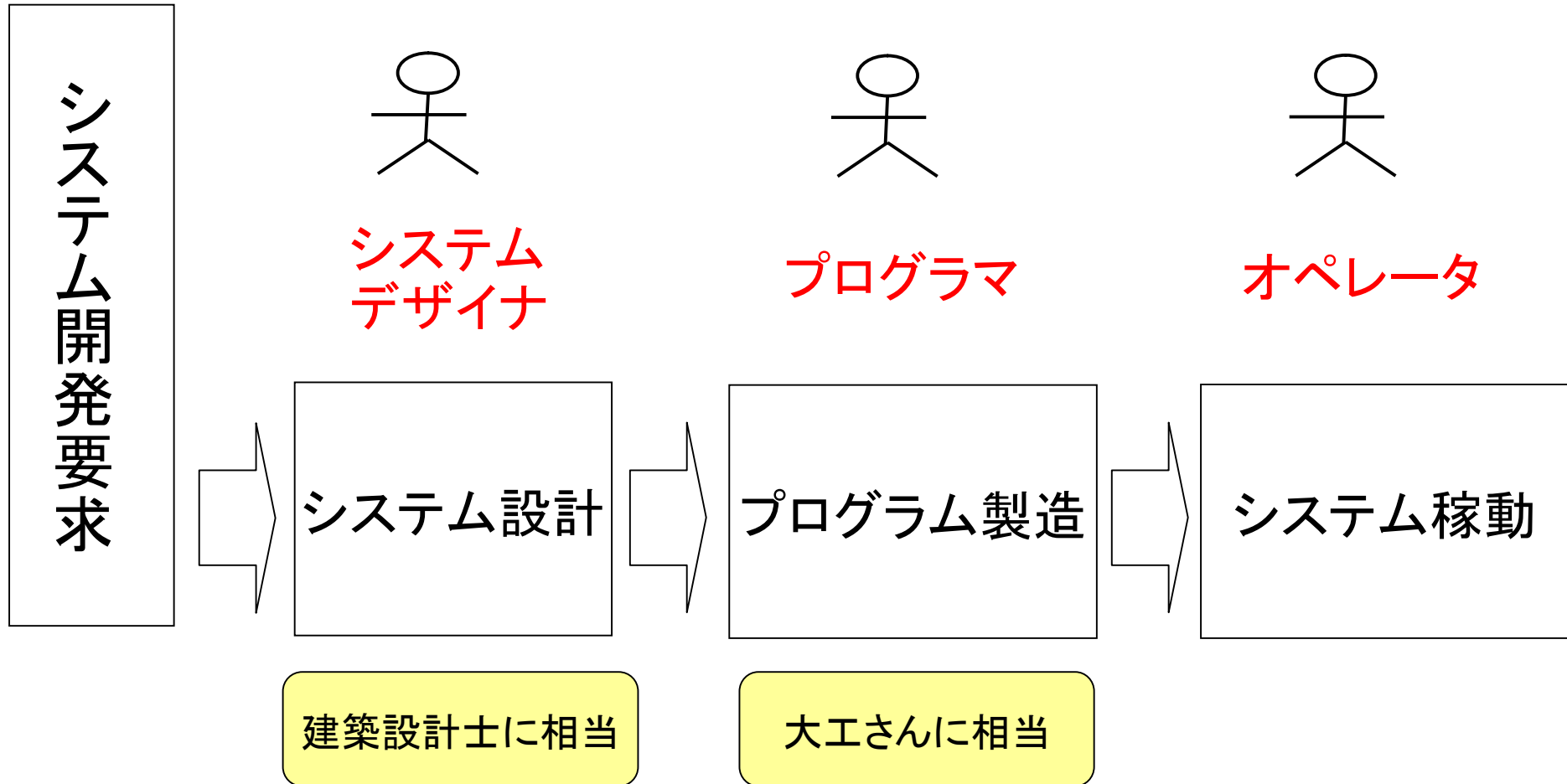
1. 1 誰が開発しているのか

1. 2 システム開発の例示

1. 3 システム開発のアウトプット

全て、人が考えたことを基にして、  
人手によりシステムを製造している。

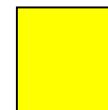
# 1.1 誰が開発しているのか



- ・全て、人が考えたことを基にして、
- ・人手によりシステムを製造している。

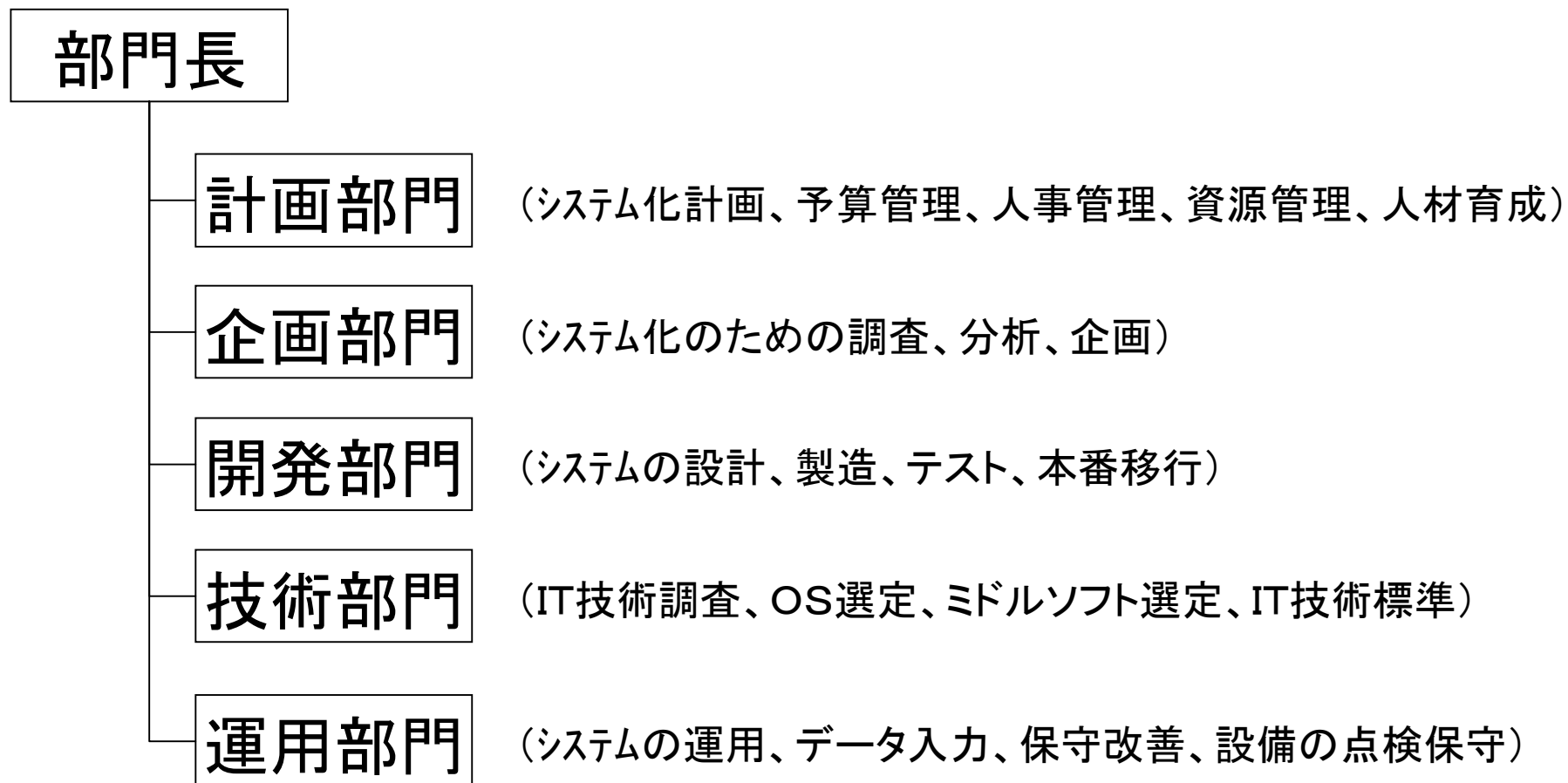
(デザイナー+プログラマ+オペレータ=システムエンジニア=SE)

# 資料： 情報処理資格試験



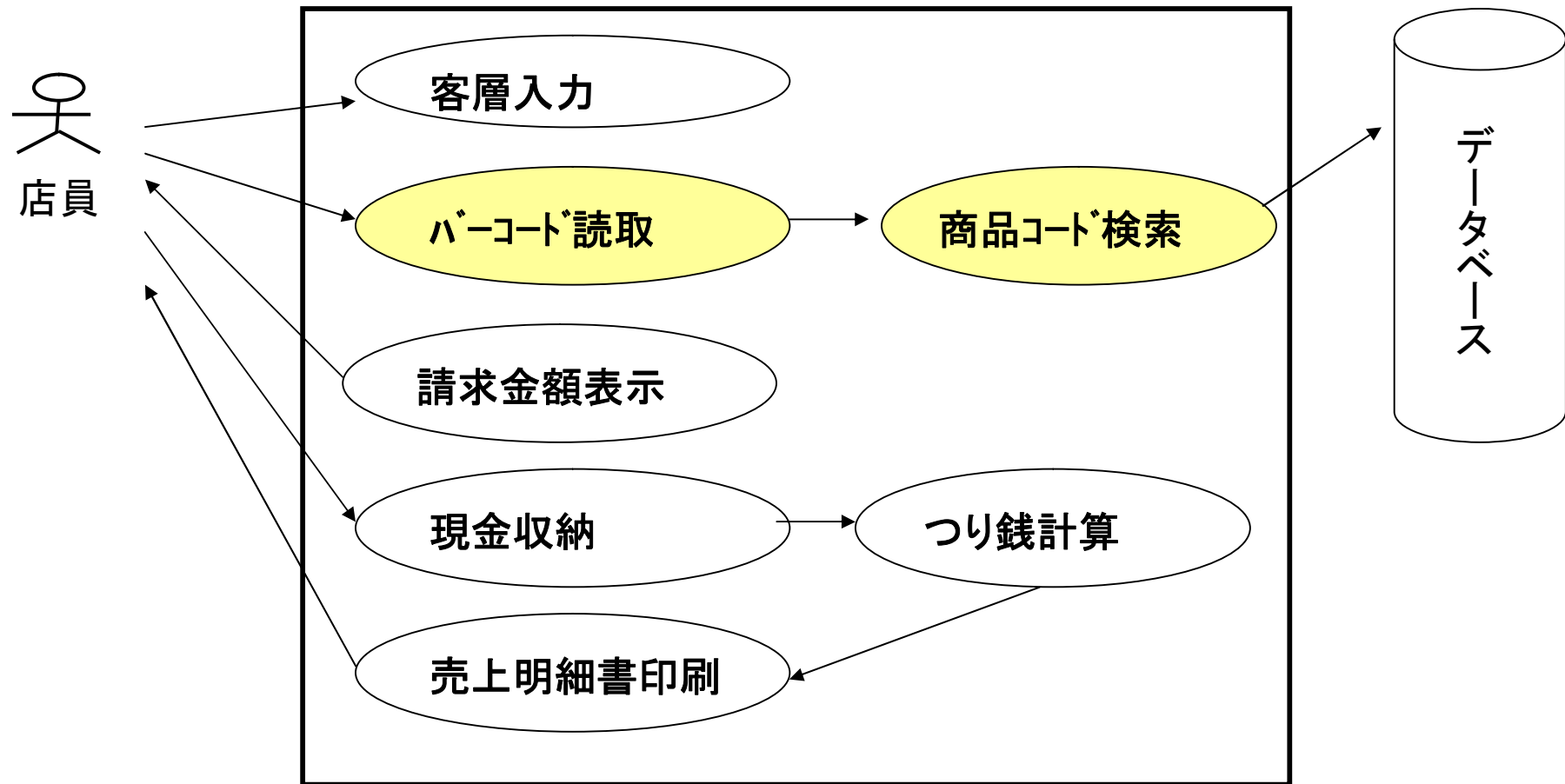
	資格	該当する開発プロセス
アプリケーションSE	システムアナリスト	システム企画
	プロジェクトマネージャー	外設～移行
	アプリケーションエンジニア	要件定義
	ソフトウェア開発技術者	外設、内設、製造・テスト、 統合テスト・移行、運用・保守
	基本情報技術者	製造
テクニカルSE	テクニカルエンジニア	NW、DB、システム、機械組 込みシステム
ユーザー	システムアドミニストレータ	全般、改善、セキュリティ
第三者	システム監査技術者	内部統制、システム監査、報告

# 資料：企業の情報システム部門の組織



# 1.2 システム開発の例示

## ① スーパーのレジのシステム化要求

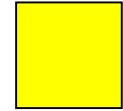


## ② スーパーのレジのシステム設計

例示:バーコード読取部分

1. バーコードを読み取る。
2. バーコードをキーにして、商品データベースを見にゆき、商品名と価格を取り出す。
  - ・日時により価格が変わる場合があるので、有効な価格を取り出すこと。
  - ・商品データベースに見当たらなければ、レジスター画面にエラー表示。
3. 買い物かごから全ての商品が無くなるまで、1、2を繰り返す。
4. 買い物かごから全ての商品が無くなったら、終了キーを押す。
5. 終了キーが押されたら、お買い上げ商品の合計額を計算して、レジスターに合計額を表示する。

### ③ プログラミング例



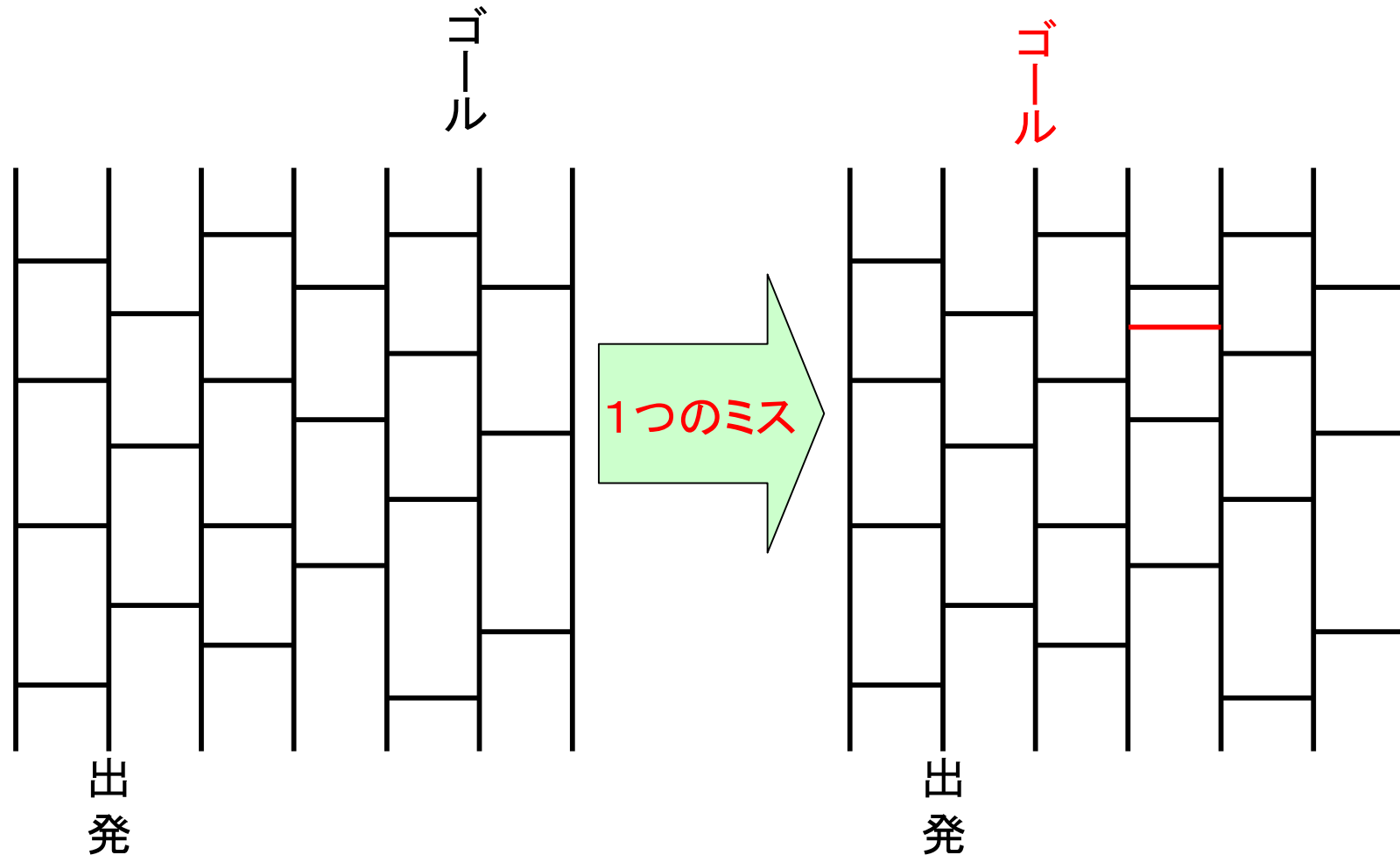
例示: バーコード読取部分 (以下は、説明用に仮のプログラミング言語を使用)

```
start .
  if end-sw = 1 go to end-shori,
    else move 0 end-sw .
  if count = 0 move 1 to sfx of uriage-tbl,
    else add 1 to sfx of uriage-tbl.
  read label-file.
  move shouhincode of label-file to shouhincode-area.
  get time from clock
  exec sql select hinmei, kakaku
           from shouhin-tbl
           where shouhin-tbl-key = shouhincode-area
                 and ( shouhin-tbl-end-time = space
                       or shouhin-tbl-end-time > time ) ;
  if db-error display 'ショウヒンコード ナシ' go to return
  else move shouhincode-area to shouhincode-t(sfx),
        move hinmei to hinmei-t(sfx),
        move kakaku to kakaku-t(sfx) go to start.

end-shori.
(省略)
return.
end.
```

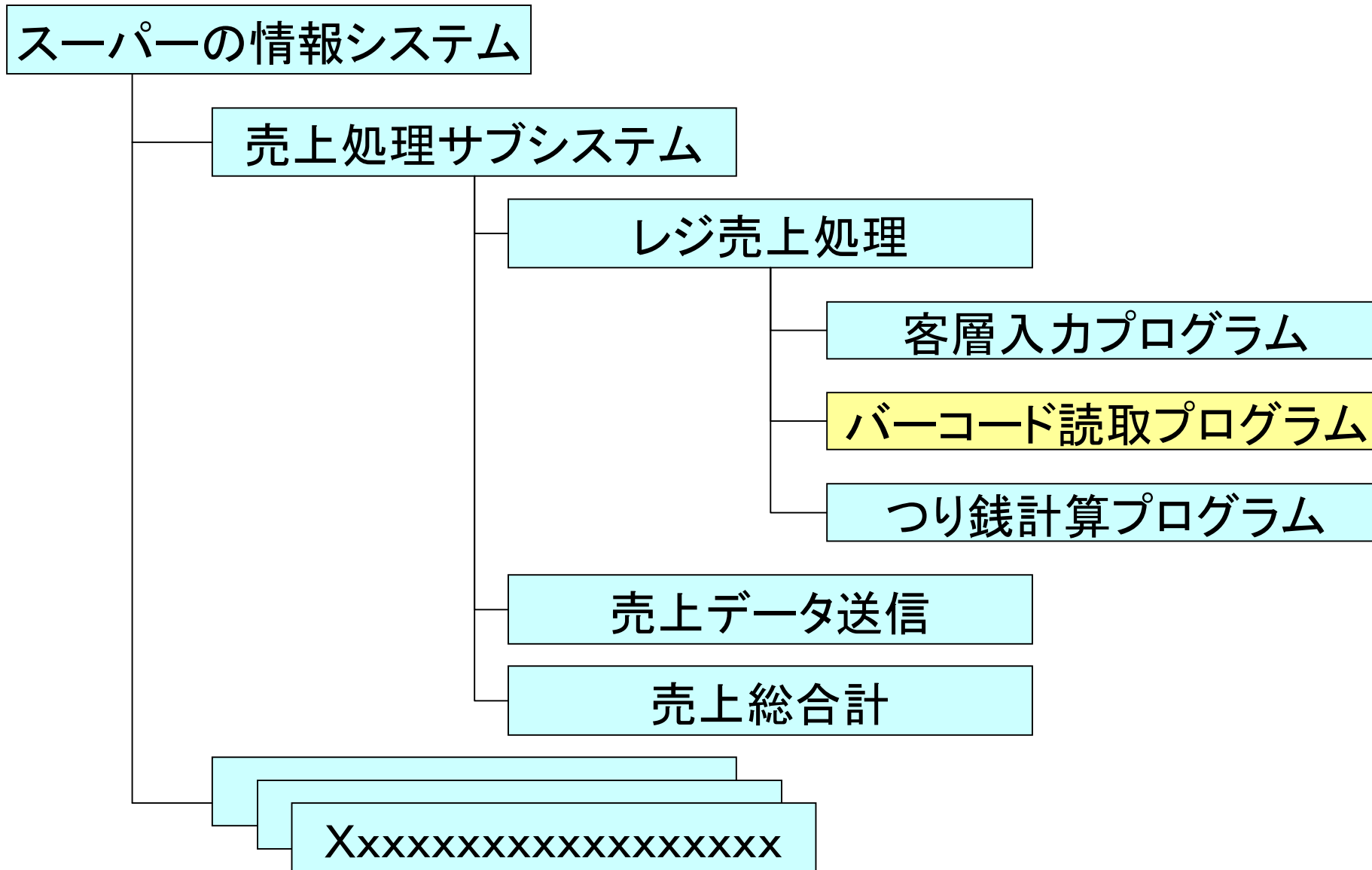
# 参考：情報システムはアミダ

プログラム上のわずかなミスが命取り（1つのミスできない世界）





# ④ 情報システム構成例



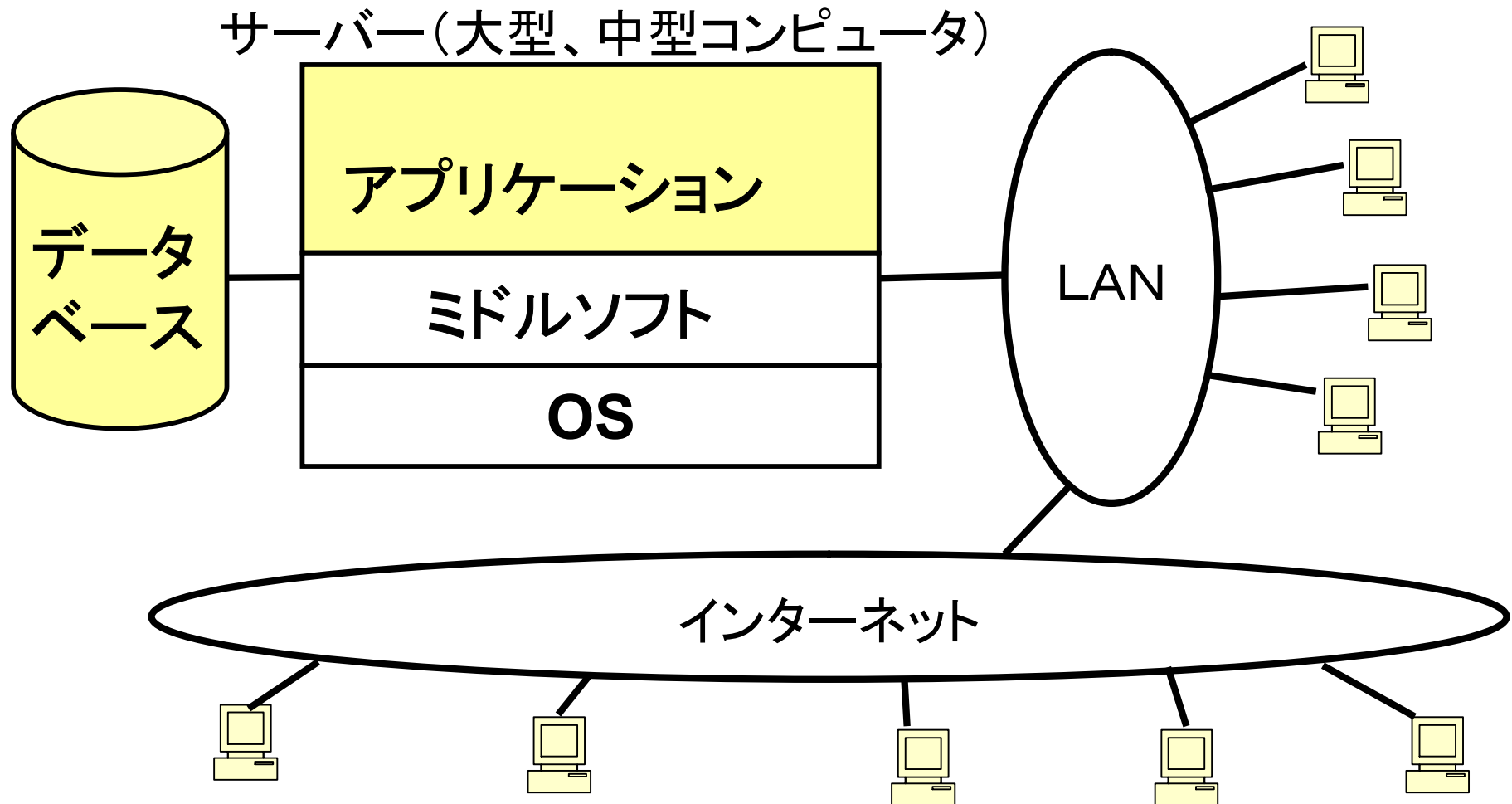
# 1. 3 システム開発のアウトプット

- 企業、公共システム
  - 企業システム (生産管理、販売管理、経理…)
  - 公共システム (住民票管理、ビザ申請…)
- ソフトパッケージ
  - 統合ソフト (企業の業務用、生産管理用…)
  - 汎用ソフト (文書作成、表作成、メールソフト…)
  - 専用ソフト (CAD、経理、給与計算…)
- 基本ソフト
  - OS (UNIX、Windows、Mac、Linux…)
  - ミドルウェア (DB管理、NW管理、ドライバ…)
  - (ブラウザ、動画処理、音楽処理…)
- その他
  - ゲーム (各種機器向けのゲーム…)
  - 機器制御 (携帯、電子機器、設備、ロボット…)

# 補足： 情報システムの構造例

情報処理の大半は、サーバーのアプリケーションプログラムが実行。

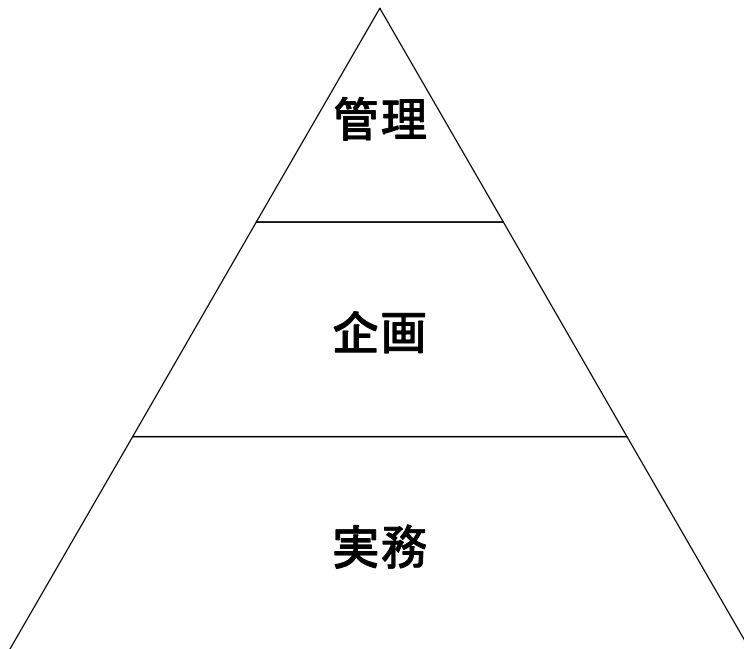
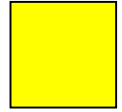
パソコンでの処理は、一部の処理のみ(データの形式変換、表示機能、文書作成)



# 補足：企業システムの開発方法

- システムが対象とするもの
  - 人、組織の活動-----各業務の遂行
  - 機械の制御-----電子機器/機械/設備
- 人、組織の活動
  - 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）
- 導入方法
  - Make / Buy / アウトソーシング
- 利用部門との共同作業
  - プロジェクトを組織する

# 補足：私が目指して来たシステム



## 管理レベル

- ・うまく回っていない業務は、その事実、原因がわかり有効な手が打てるような情報提供。

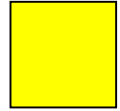
## 企画レベル

- ・開発、製造、販売現場の生の情報をタイムリーに提供する。

## 実務レベル

- ・システムが指示した通りにやれば業務がうまく回る。

# 資料：システム開発をする企業



## • OS

- ・マイクロソフト(MS)、アップル、坂村教授(TRON)、
- ・IBM、Linux供給企業(無料OS)、サンマイクロ(SUN)...

## ▪ ミドルソフト

- ・データベース管理(オラクル、IBM、MS...)
- ・ネットワーク管理(IBM、MS、アップル、ノベル、SUN...)
- ・インターネット(IBM、MS、アップル、シスコ...)
- ・画像(MS、アップル、IBM、Adobe.....)
- ・音声、音楽(MS、アップル、REAL.....)
- ・暗号化技術(ベリサイン、ロータス...)
- ・自動翻訳(富士通、マジカルゲート、)
- ・音声認識(ドラゴンシステムズ、IBM、東芝、NEC...)

## ▪ アプリケーションソフト

- ・利用企業、一般のソフト開発会社、コンピュータメーカー

# 資料：ソフトウェア産業の売上

2003年度

ソフトウェア開発	5.4兆円
ソフトウェア製品	1.2兆円
情報処理受託	0.6兆円
システム運営	1.0兆円
その他(DBサービスなど)	0.8兆円

計

9.0兆円

(出典：経済産業省、「特定サービス産業動態統計・情報サービス業」統計、

<http://www.jisa.or.jp/static/dotai/2003-j.pdf> )

# 資料：企業のITレベル実態



(調査対象：5303社、平均社員数 886名)

## ①システム開発状況

既存システムを運用	50%
新システム構築	44%
今年是新システム無し	6%

## ②ネットワーク導入状況

LAN導入	93%
メール導入	90%

## ④IT部門の要員数

SE	11.8名
プログラマ	6.0名
NW	1.5名
管理、企画	2.8名
その他	2.3名
計	24.4名

## ③保有コンピュータ台数

ホスト、オフコン	9.4台
ワークステーション	36.4台
パソコン	242.2台
端末	18.4台

(出典：通産省、2001年、情報処理実態調査速報)



## 2. システム開発のプロセス

- 2. 1 開発工程
- 2. 2 開発プロセスの重要点
- 2. 3 システム開発で使う技法
- 2. 4 システム開発業務の改善点

# 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

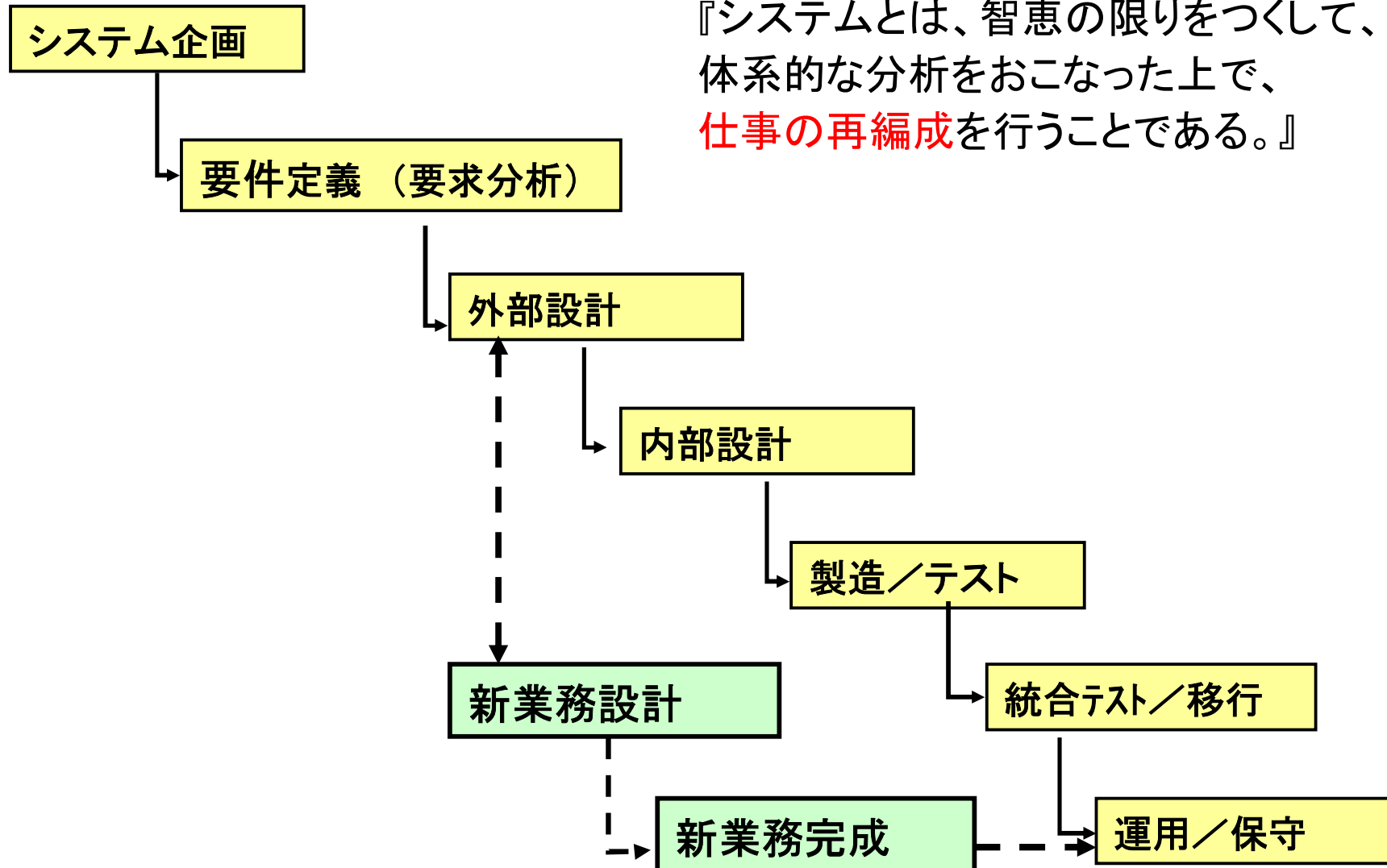
⑨ 業務知識

# 2.1 開発工程

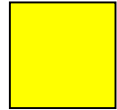
## ① 開発の全体プロセス

P. ドラッカー

『システムとは、**智恵の限りをつくして、体系的な分析をおこなった上で、仕事の再編成**を行うことである。』



## ② 開発プロセスの内容



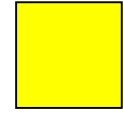
	仕事の内容
システム企画	・目的、方針、ビジネスモデル、機能、構造、効果、予算
要件定義 (要求分析)	・要求の調査／分析、範囲、要求仕様まとめ(機能・DB・品質)、実現性、費用、計画作成
外部設計	・機能、DB、I/O、構造を決定
内部設計	・上記の物理モデル(実装レベル)
製造／テスト	・プログラム製造、テスト
統合テスト／移行	・本番並みのテスト、本番移行
運用／保守	・運転、監視、保守(改善・バグ修正・トラブル予防)

## 2.2 開発プロセスの重要点



	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

# 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

# 補足： システム開発のポイント

- **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

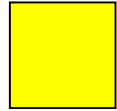
## 2.3 システム開発で使う技法

### ① 業務の現状調査、分析

- 問題点、課題を分析する。
  - 関係者へのインタビュー
- 目標、問題の原因分析技法
  - 問題－原因－解決策(因果分析技法)
  - 目的－手段の連鎖分析(構造化分析)
- 新システムへの要求を分析
  - 要求の構造化
  - プロトタイピング(試作)

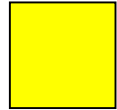


## ② システム設計技法



- ユーザー視点でシステム機能を考察
  - システム機能の下方展開
  - データベースの設計
  - I/O画面、帳票を設計
- ユーザーマニュアルの作成
- 内部処理方法を考察
  - 最下位のプログラム単位まで下方展開
  - 単機能まで分解(モジュール設計)
  - アルゴリズム(コンピュータでの処理方法)
  - デシジョンチャート(ケース分析法)

### ③ 製造、テスト、運用の技法



- ・プログラム部品的设计、製造
- ・テストケースの分析
- ・リスク分析(システム不具合と悪影響)
- ・システム保守マニュアル作成
  - ・システム設計のノウハウ集
  - ・異常時の記録の整理保存
- ・システム運用マニュアル作成
  - ・運用時の例外ケース分析
  - ・運用時の事前の異常発見
  - ・運用記録の保存(データ量、処理時間推移)

## 2.4 システム開発業務の改善点

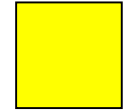
### ① RAD方式 (Rapid Application Development)

- ・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)

### ② 見積りの精度向上

- ・従来からの見積り方法
  - ・経験から割り出したプログラム1本当りの工数もとに開発期間、費用を見積る
- ・Function Point(機能数)法
  - ・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

## 2.4 (つづき)



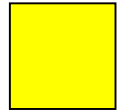
### ③ IT技術標準の整備

- ・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

### ④ 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 など

# 3. プロジェクト運営



## 3.1 プロジェクト管理の概要

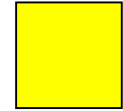
### ①プロジェクト管理の対象

- ・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）

## 3.2 経験則



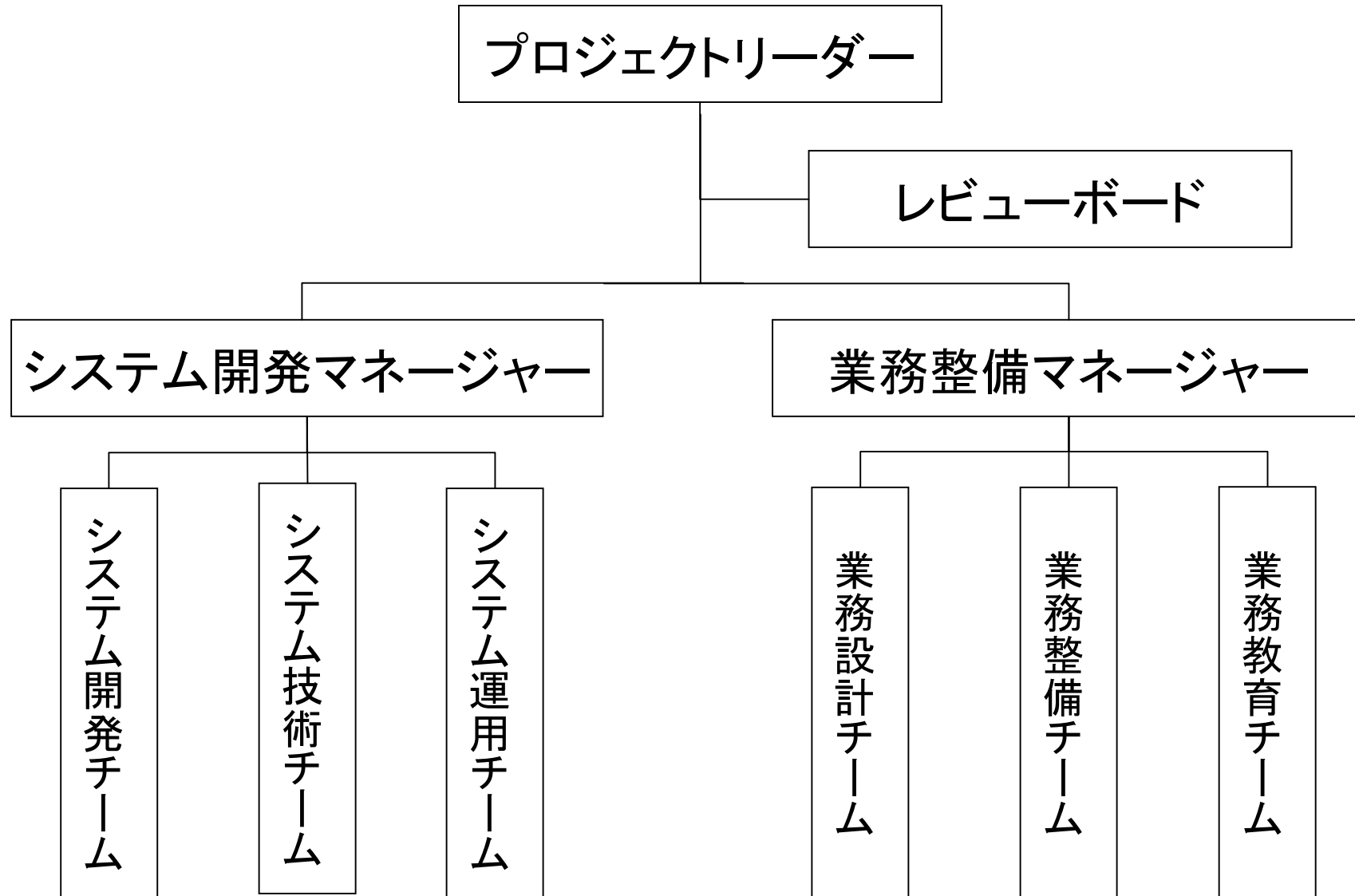
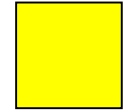
- ・納期を守ることが出来れば、品質・コストは、後からついて来る。
  - 納期厳守に集中
- ・プロジェクト・メンバーが出来ないと考えることは、当たっている。
  - 障害物を除く
- ・いくら早くプログラミングを始めても、システムは早く完成しない。
  - 上流工程が肝要
- ・大半のシステム欠陥は、上流工程で生まれる(要件定義、外部設計、内部設計)
  - 上流工程が肝要
- ・大規模システムは一度に開発しない
  - 分割が有効
- ・ツールの効用は絶大である
  - 広くツールを探し、活用する

# 担当部門、プロジェクト管理、見積

	ユーザー	SE	PM	レビュー	見積
システム企画	○	◎	実施	実施	実施
要件定義（要求分析）	○	◎	実施	実施	実施
外部設計	○	◎	実施	実施	実施
内部設計	—	◎	実施	実施	（実施）
製造／テスト	—	◎	（実施）	（実施）	—
統合テスト／移行	○	◎	実施	実施	—
運用／保守	○	◎	—	実施	—

◎主担当、○共同、—関係ない

# 資料：プロジェクト体制





## 4. システム開発の問題点

- ① 完成したシステムに不具合が潜在する
  - ・人手で開発しており、100%の保証は無い
- ② システム開発の期間が長い
  - ・小規模でも2、3ヶ月かかる
  - ・大規模だと2、3年かかる
- ③ IT技術の進歩が早く、修得したスキルが陳腐化
  - ・IT要員のスキル修得が追いつかない
  - ・IT要員の教育負担が多大(30%の工数)
- ④ 他人が開発したシステムの保守が難しい
  - ・システム変更が多く、保守マニュアルが古い
  - ・要員の交代が難しい

# 5. SEに必要な素養

5. 1 仕事のリテラシ

5. 2 熟慮断行

5. 3 SEに必要なスキル

5. 4 分析技法の例示

## 5. 1 仕事のリテラシ

- 一時には、一つのこと集中する
- 考えを紙に書いて整理する
- 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
- 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)

上記があってはじめて組織での仕事ができる。

## 5.2 熟慮断行

- 熟慮
  - 平素からの対象の専門研究
  - 現状、課題調査と分析
  - 新IT技術の適用検討
  - 解決策のまとめと提案
- 断行
  - 新IT技術の適用の決定
  - 新システムの効果をアピール
  - 新システムの開発、導入

## 5.3 SEに必要なスキル

### ① 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)

## ② SEの心構え

### • SEの仕事の進め方

- 何事も計画を立ててから進める
- 書き出してみる(リスト、関連図)
- 自分の考えを話し、人からアイデアをもらう
- 大きな課題からかたずける
- 決まらないことはユーザとトップと直談判
- 5W3H(What, Why, Who、When、  
Where, How to, How much, How many)

### • SEの心構え

- 好奇心→もっと知る→対象業務を好きになる
- 忍耐力、持久力、スタートしなければゴールに着かない
- プラス思考(入って来る全ての情報を「快」と捉えるクセ)

## 6. 分析技法の紹介

### ① アイデアを出す

#### ・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)

#### ・5W3Hで発想する

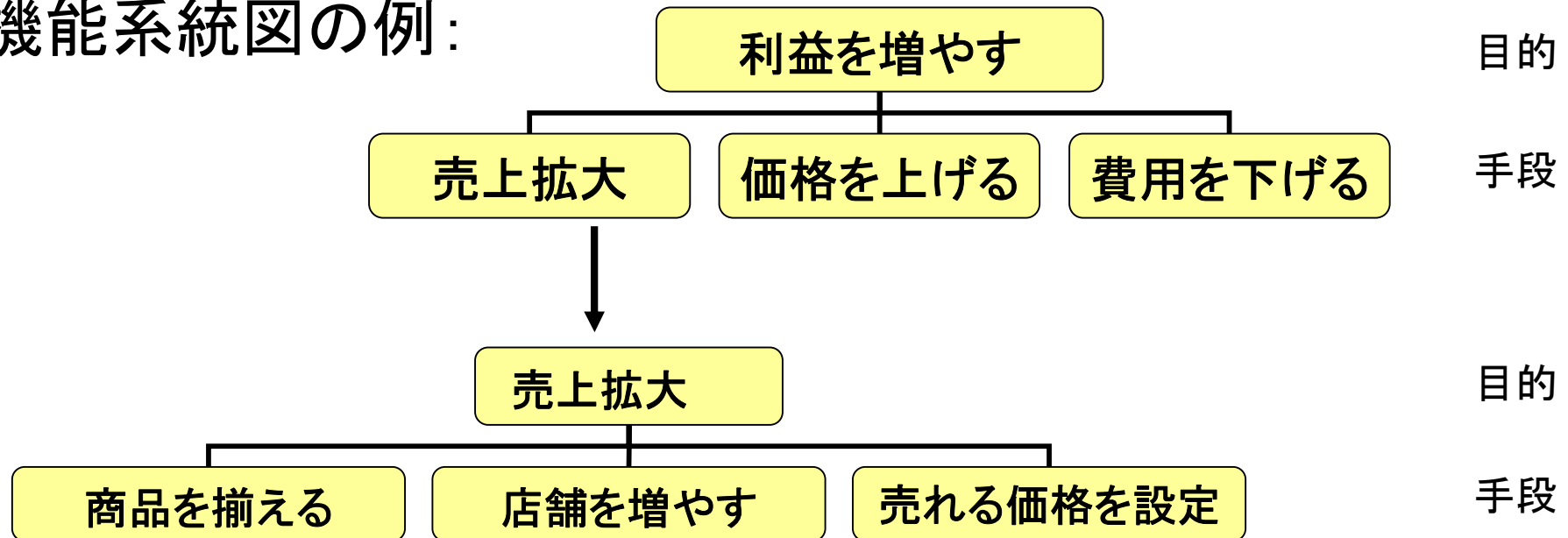
- Why、What
- When、Who、Where
- How to
- How much
- How many

## 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)

機能系統図の例:



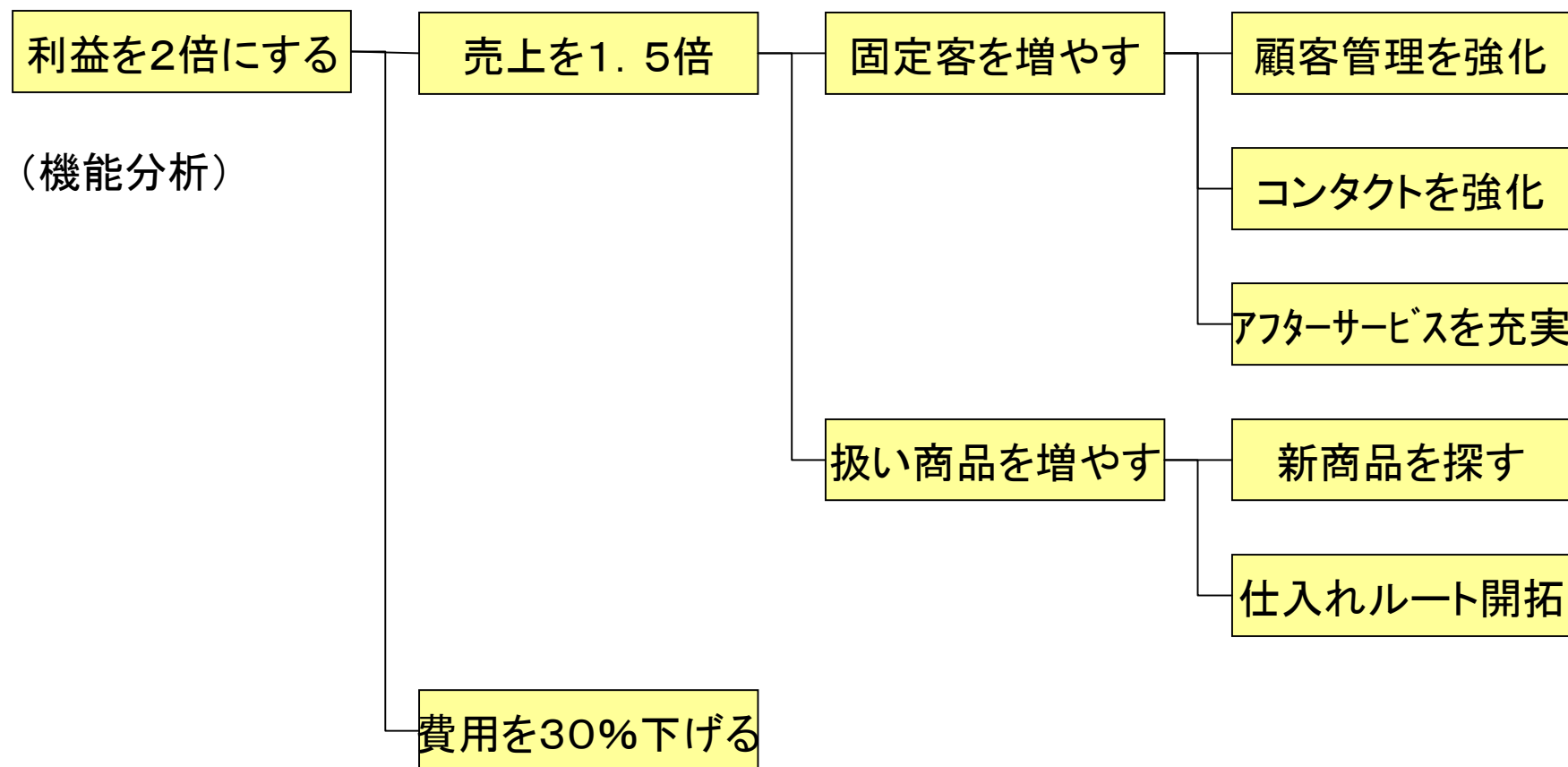


### ③ 下方への展開方法(着眼点)

- 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)

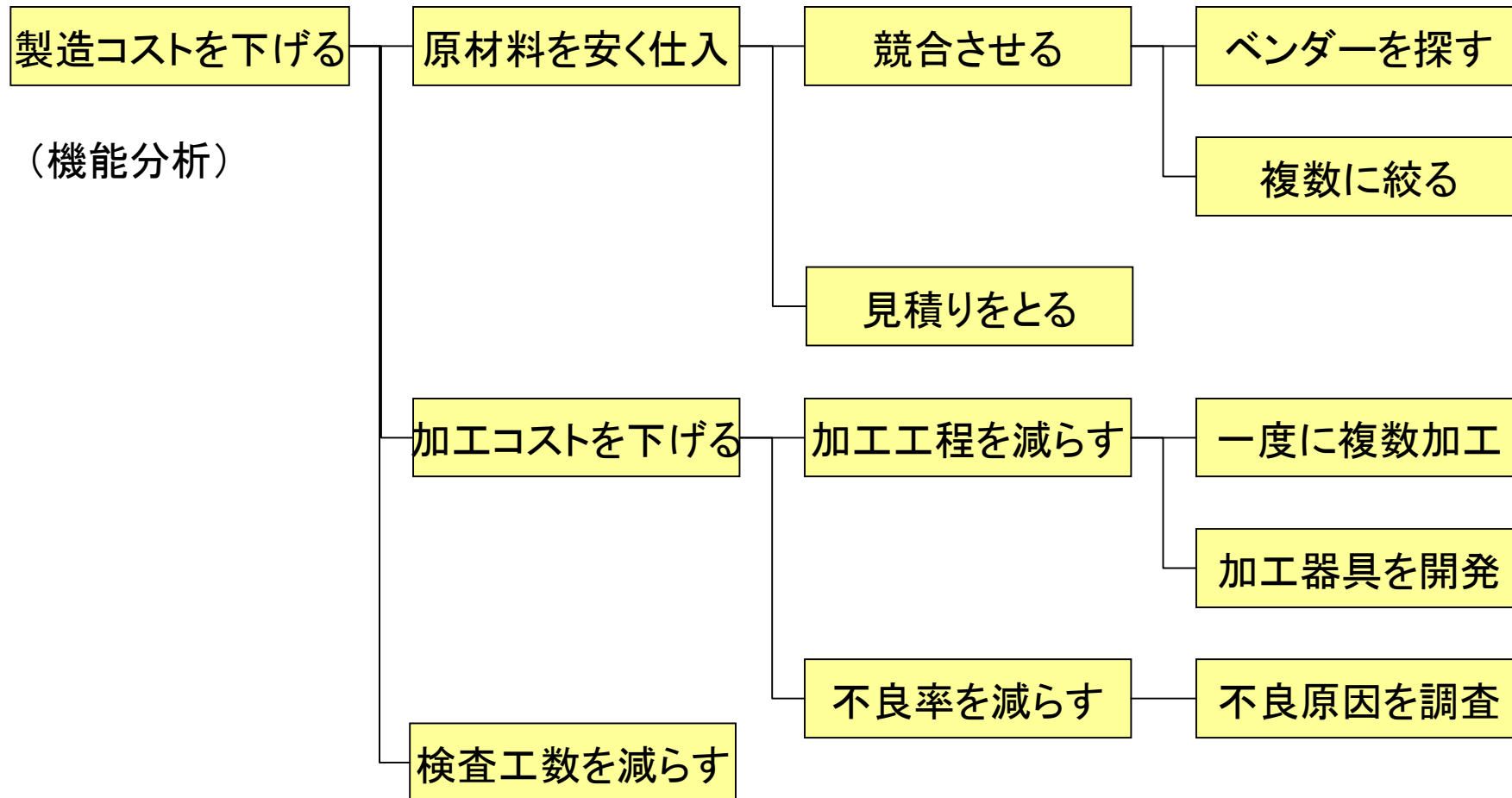
## 例示：機能に着目した下位展開方法

- ・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



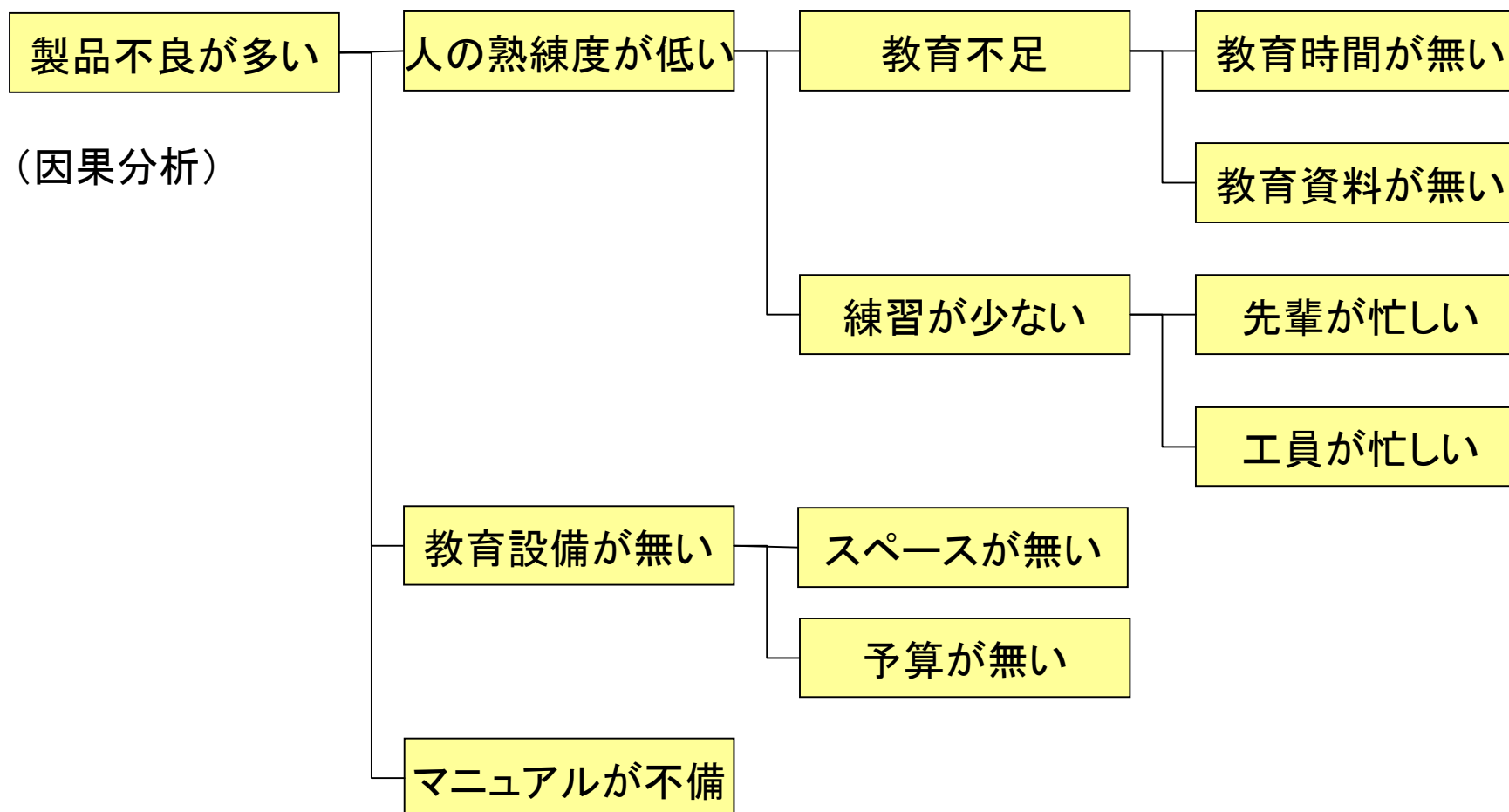
# 例示：業務プロセスに着目した下位展開方法

- ・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

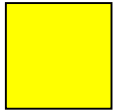


## 例示：資源に着目した下位展開方法

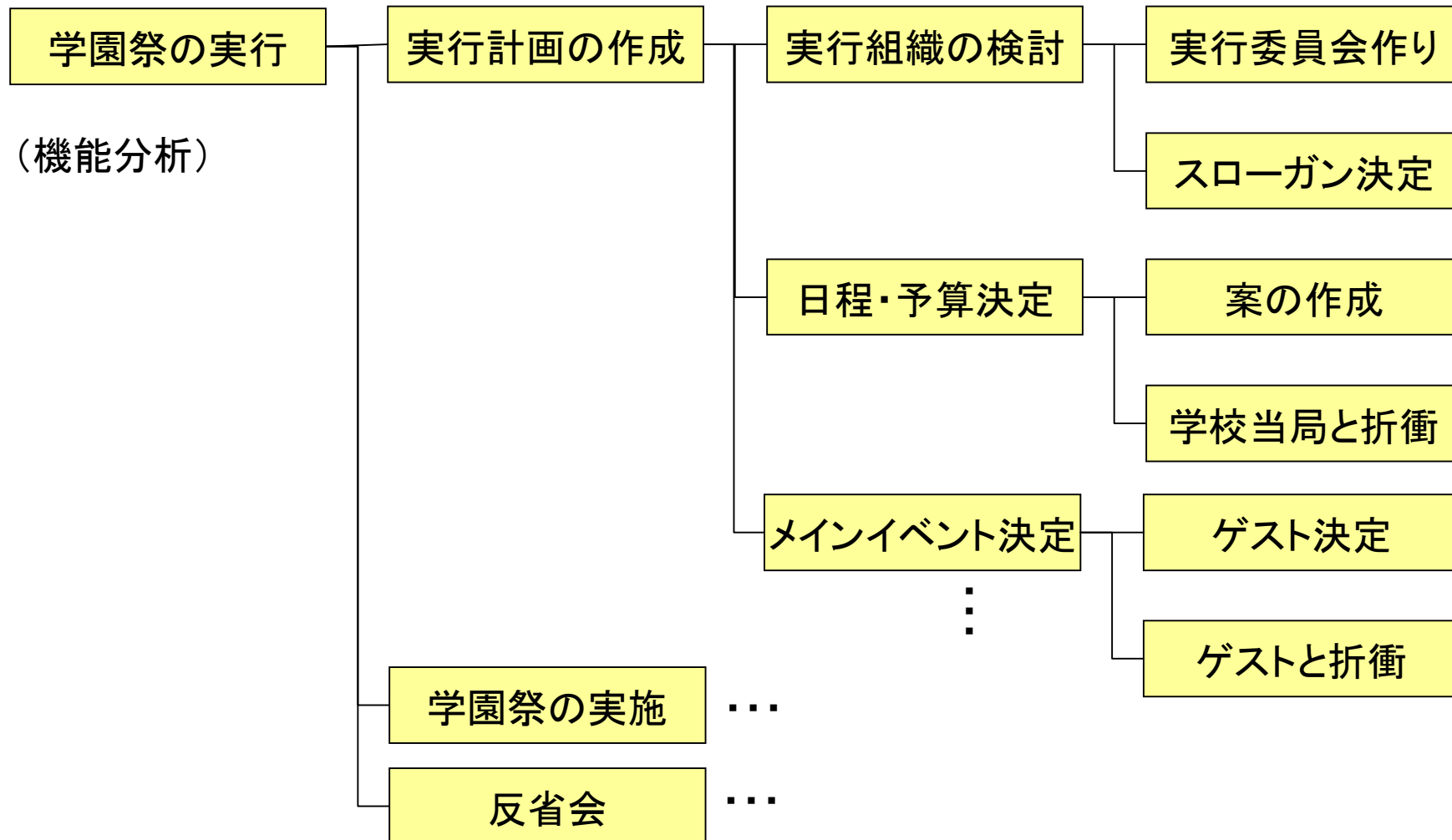
- ・対象とする目的や問題を資源面から分析
- ・資源面では、人、モノ、金、情報、時間に注目する



# 例示：検討例として、「学園祭」の実行を考える



- ・目的を・目標を決めて、その実現手段を考える
- ・実現手段ごとの着眼点を考え、解決手段を挙げる



# IT社会は、SEが支える

- 2000年問題
- サーバーテロ
- 通信障害
- オンライン停止
- デジタルデバイド



- 情報漏洩
- ウィルス
- フィッシング
- スパイウェア
- 破壊・改竄



システム・エンジニア

# 7. レポート課題

- 以下の点について、まとめをレポートで提出
- 内容 ①自分の目標について、
  - ・目標を具体的に定め、
  - ・目標実現のためのアイデアを出し、
  - ・実現方法、手段を考え、
  - ・機能系統図にまとめよ。

(具体的に実行できる手段がわかるレベルまで、目標－手段を下方展開すること。)
- 形式 A4、1枚以内 (学科、学生番号、氏名明記)
- 提出方法 メール(添付も可)、またはペーパー  
TO: [fwhy6454@mb.infoweb.ne.jp](mailto:fwhy6454@mb.infoweb.ne.jp)
- 提出期限 次回の授業開始までに。

## 8. 参考書、参照Webサイト

- 戸田忠良「上級SEになるための50のポイント」 共立出版
- アラン M.デービス著、松原友夫訳  
「ソフトウェア開発201の鉄則」 日経BP社
- 村上哲大「仕事を生かす目的発想法」 溪水社
- 情報処理ハンドブック 情報処理学会 オーム社
- パソコン用語事典 岡本茂ほか 技術評論社