

# 第8章 ソフトウェア エンジニアリングの展開

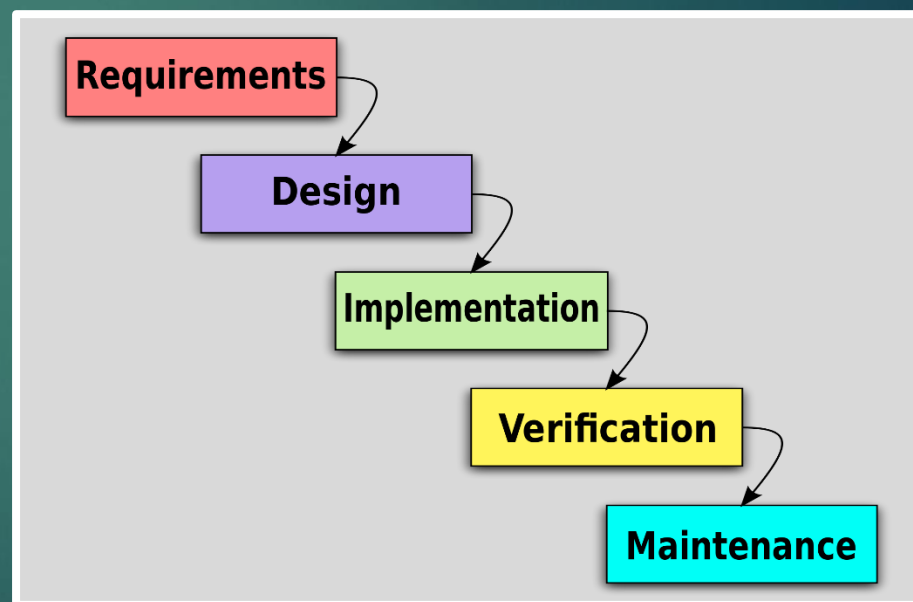
資料：図解でわかるソフトウェア開発のすべて・第8章ISBN-534-03109-2  
日本実業出版社

著者：MINT(経営情報研究会)・浜崎正和(第8章担当)

# ソフトウェア エンジニアリング

2

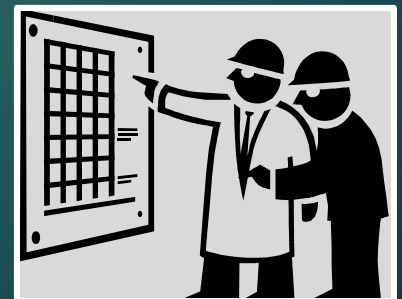
- ▶ 開発
  - ▶ 部品化と再利用
- ▶ 運用
  - ▶ システム運転
  - ▶ 監視
- ▶ 保守
  - ▶ 修正
  - ▶ 改訂
  - ▶ 予防



# ソフトウェア開発の現状

3

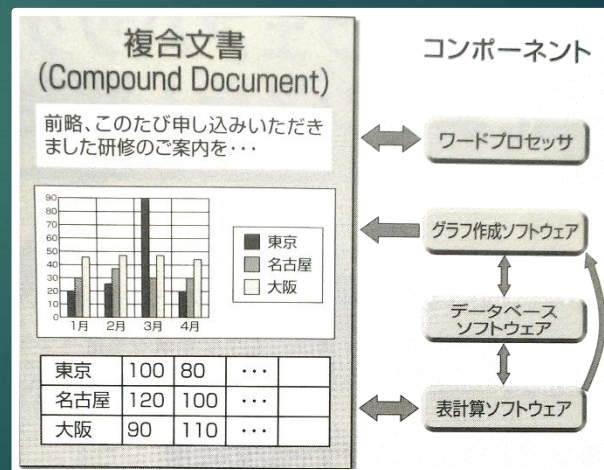
- ▶ あえて理論を無視した開発
  - ▶ 正規化しない
  - ▶ モジュール混合
- ▶ "開発"しないソフトウェア開発
  - ▶ 部品化
  - ▶ 再利用



# コンポーネントによる部品化

4

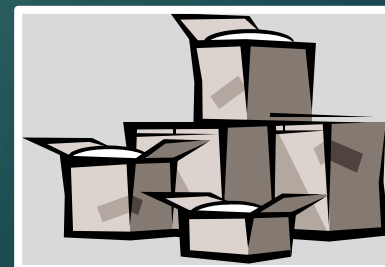
- ▶ プラグ&プレイ型部品
  - ▶ プログラム
  - ▶ データベース
- ▶ 動的にコンポーネントを組み込み
  - ▶ コンポーネントベース開発



# パッケージを使った再利用

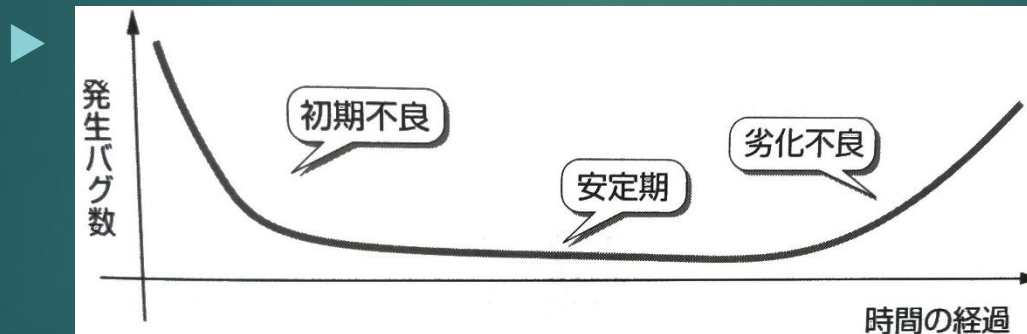
5

- ▶ パッケージを業務に適用
  - ▶ FIT：変更せず適用
    - ▶ 早い
    - ▶ 安い
    - ▶ 安定した品質
  - ▶ GAP：変更して適用
    - ▶ 高い
    - ▶ 品質は不確定
    - ▶ 保証なし



- ▶ 構造化手法（4章参照）
  - ▶ 各開発工程を構造化
  - ▶ 事務作業のシステム化に適切
  - ▶ ウォーターフォール型開発
- ▶ オブジェクト指向手法（5章参照）
  - ▶ リアルタイム制御系への適用に期待
  - ▶ 技法・目的への理解が必要

- ▶ 保守とは？
  - ▶ システムを長期的に運用する段階
    - ▶ 修正
    - ▶ 改訂
    - ▶ 予防
- ▶ バスタブ曲線



- ▶ 保守支援ツールの活用
  - ▶ ソーススキャンツール
  - ▶ 文書化ツール

# 保守を意識した設計・開発

8

- ▶ 保守のしやすさ
  - ▶ 他人が手を加えやすい
  - ▶ 時間が経っていても問題ない
- ▶ 留意する点
  - ▶ モジュール分割
    - ▶ 規模を小さく
  - ▶ コーディング規格
    - ▶ 全体で共有
  - ▶ コメント
    - ▶ 理解しやすく
  - ▶ ドキュメント化
    - ▶ 陳腐化を避ける

# まとめ

10

- ▶ 最初からの開発はしない
  - ▶ パッケージ
  - ▶ コンポーネント
- ▶ 長期的な保守
  - ▶ 開発段階から保守を視野に入れる