

**開発プロセスとの両輪としての
エンジニアリング活動
- エンジニアリング活動の10年を振り返る -**

2012年10月11日

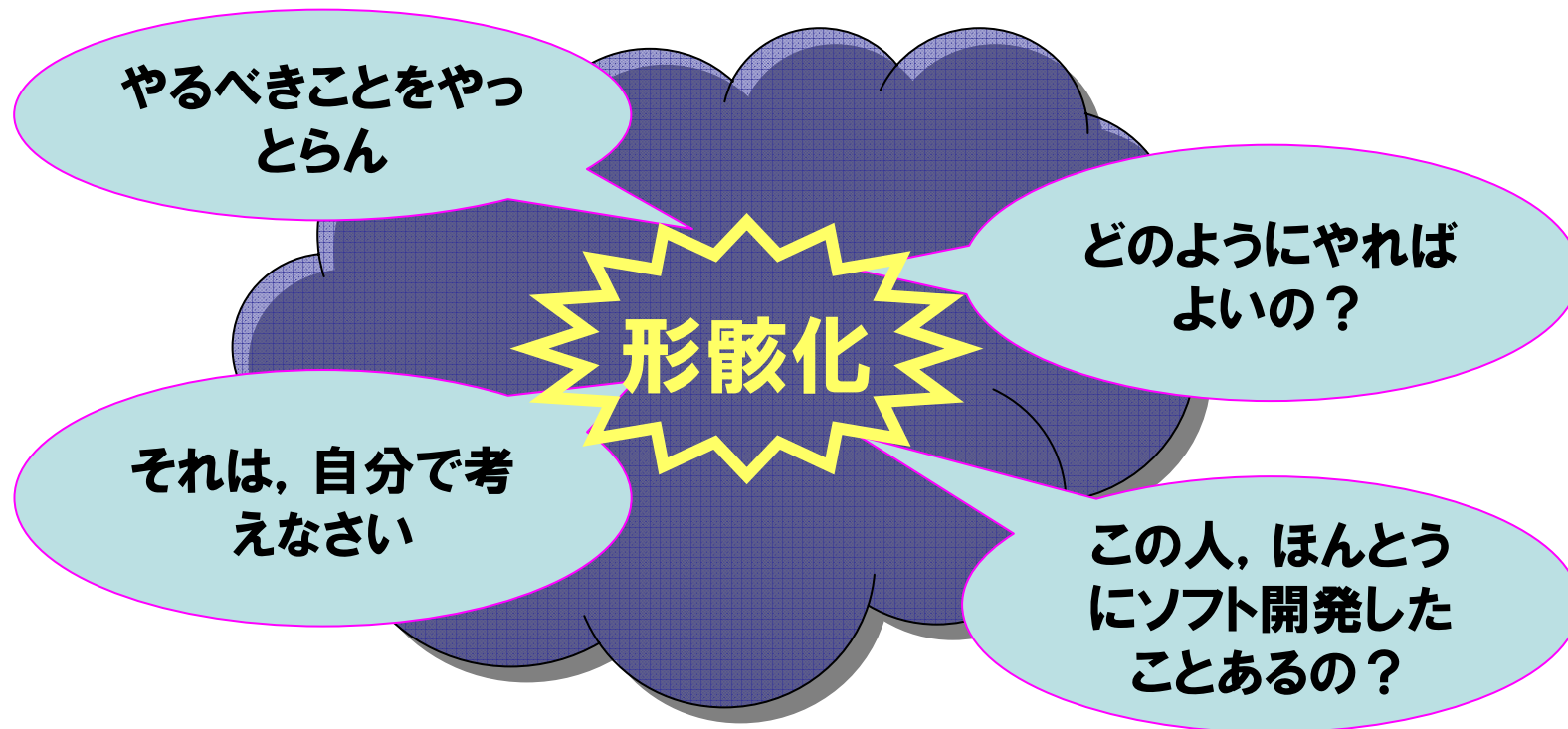
パナソニック株式会社

システムエンジニアリングセンター

春名 修介

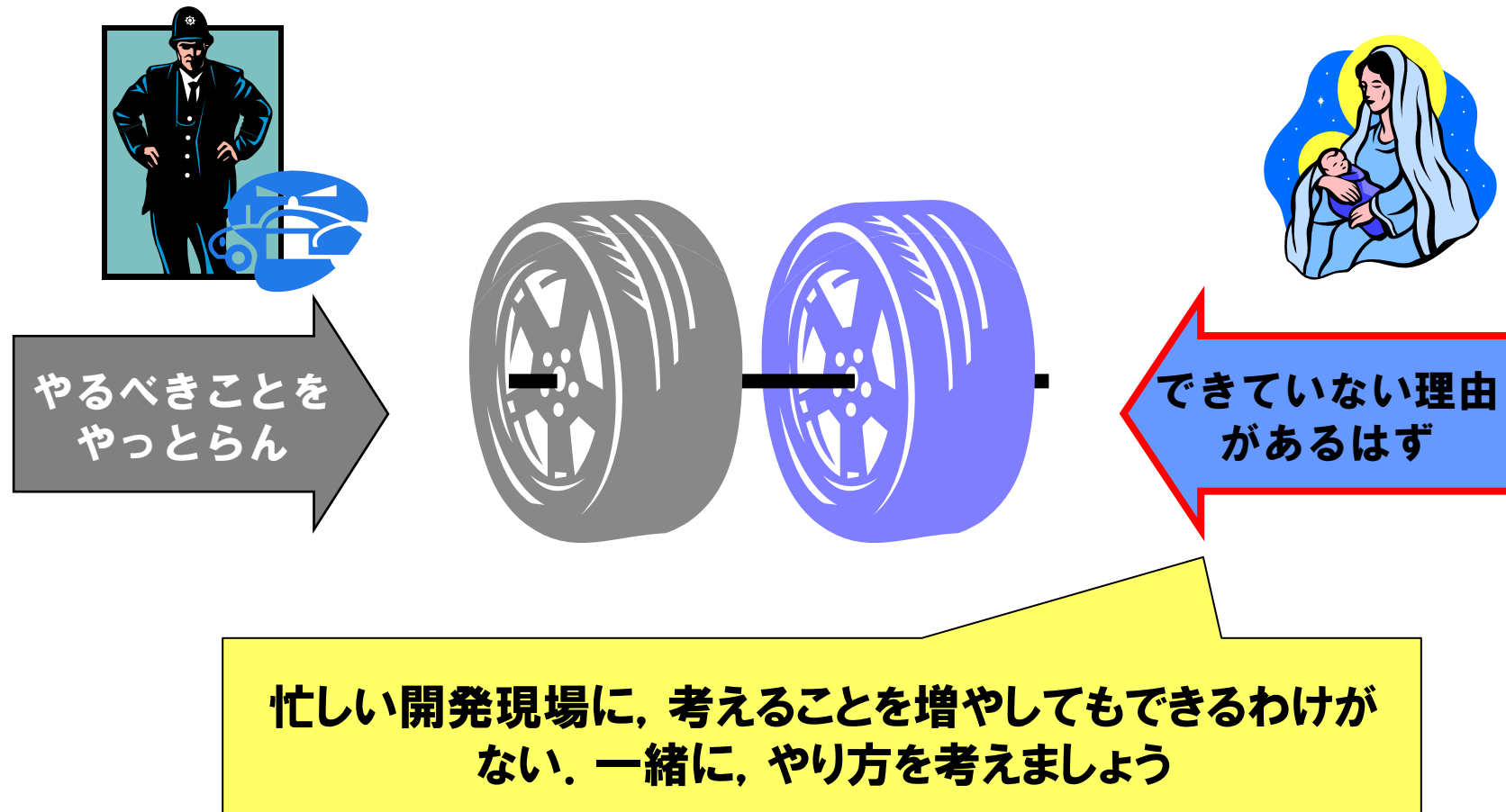
よくあるやいと

- 形(プロセス), 実態(開発力)の両立がなければ, 真の効果は出ない. ← 当たり前
 - 当たり前のことが開発組織の末端まで行き届いているでしょうか?



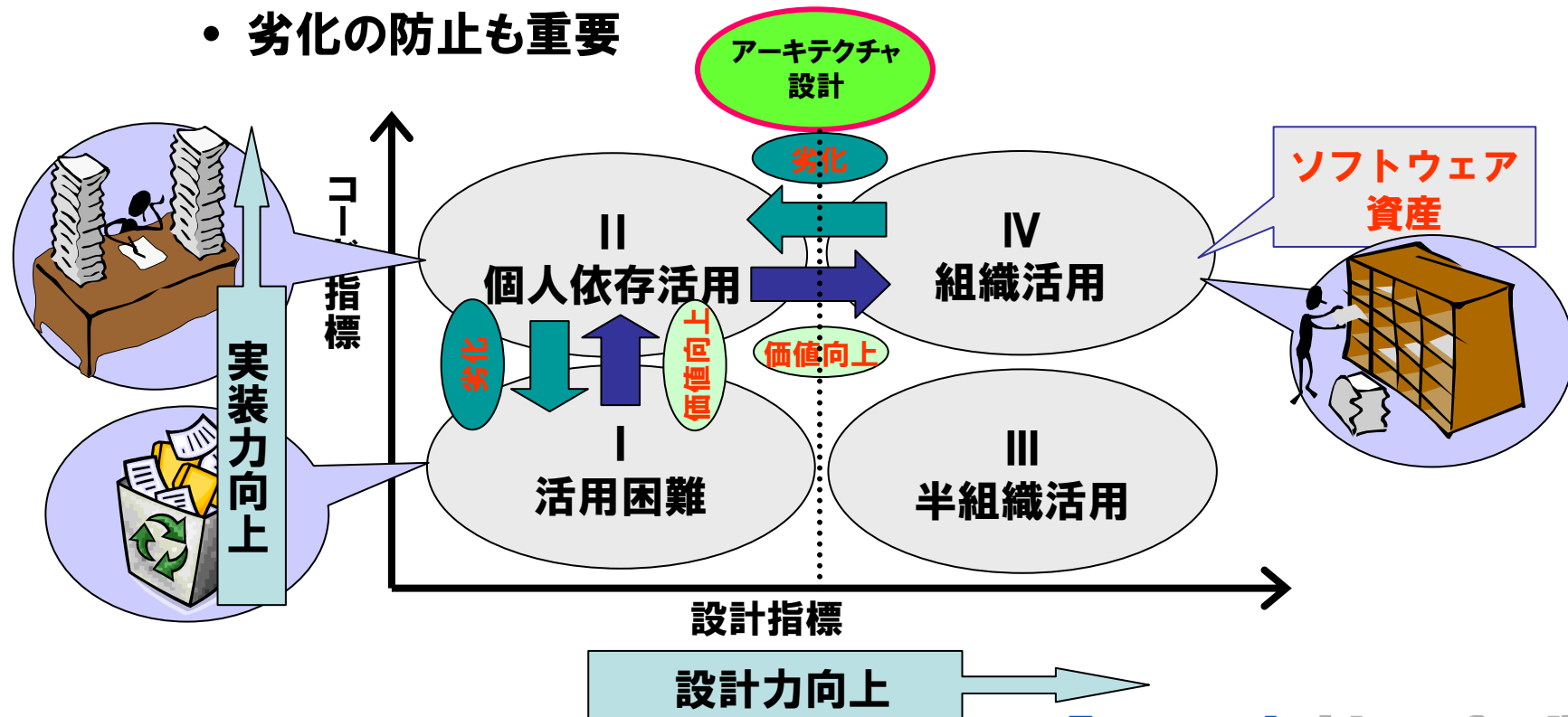
開発プロセスとの両輪としての開発力

- 開発プロセスに加えて、開発の実態に踏み込んだ開発力(開発方法)に関する支援が必須



ソフトウェアの資産価値から見た開発力

- ソフトウェア資産とは、組織活用可能なソフトウェア
 - 一度作ったソフトウェアを開発者以外が簡単に利用・活用できるソフトウェアのこと
 - 実装力・設計力双方の完成度を高めて、達成が可能
 - 劣化の防止も重要



実装力向上施策:2001~2006

• 全社コーディング規約の策定

- 全社の実装起因の不具合事例の収集
- 有識者(品質部門, コンパイラ開発者, 教育部門など)のもと, コーディング規約の策定
 - 開発実態に即し, 活用性を考慮し, 90ルールに限定
- 代表的な静的解析ツールのためのフィルタの開発. 全社配布
 - C言語を使う高品質プログラミングの教育コースへ展開

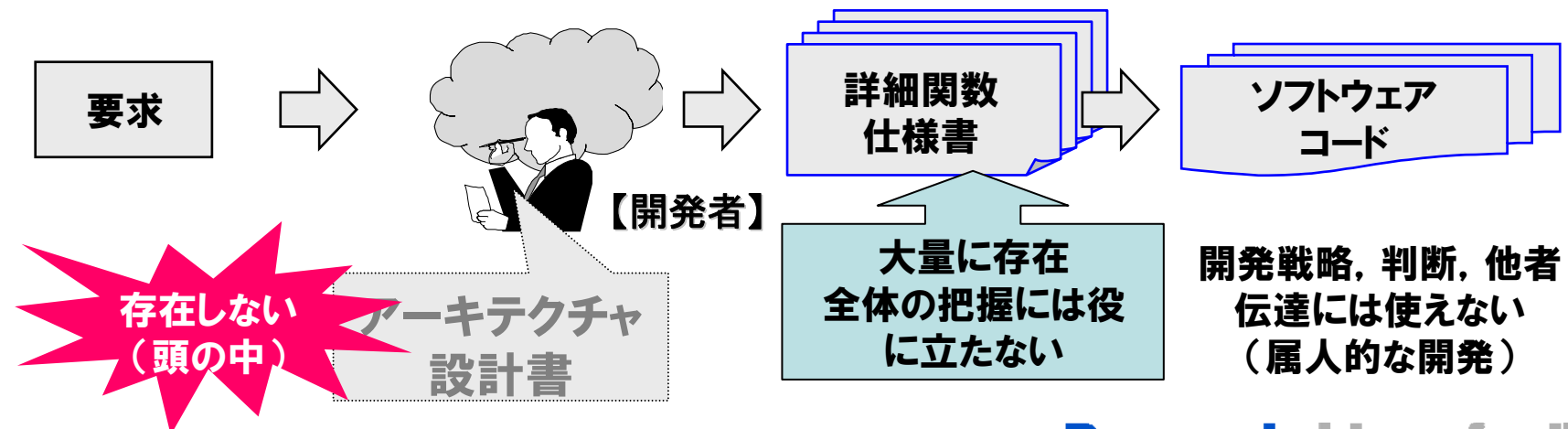
• コードリファクタリング方法の規定

- 複雑度, 関数行数, 外部変数などコード指標の規定
- アーキテクチャ・リファクタリング方法のガイド化
 - 開発現場の協力の元, リファクタリング事例作り
 - 例:RTOS無しのループ構造から, RTOSの導入方法の定式化 など

設計力向上施策:2007~

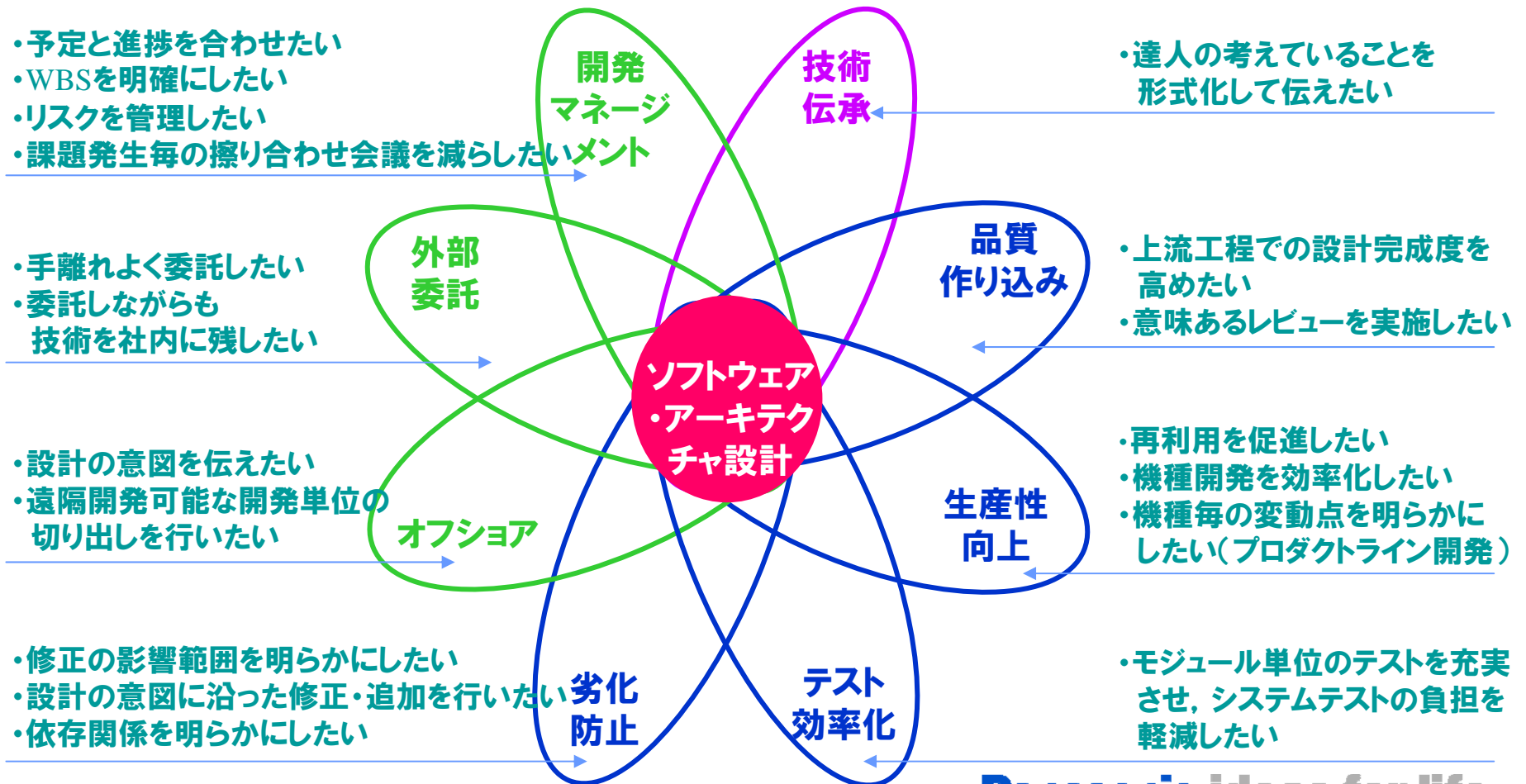
• 開発現場の設計課題の分析

- 流用開発の繰り返しにより、全体設計ができない技術者の増加
- アーキテクチャ設計書が存在しない開発部門が多い
 - 開発プロセスでは、方式設計工程の成果物は、アーキテクチャ設計書になっているが...
 - アーキテクチャ設計自体の意味が理解できていない人が大変多い
 - できる人も表現することができない(表現手段がコードになっている)



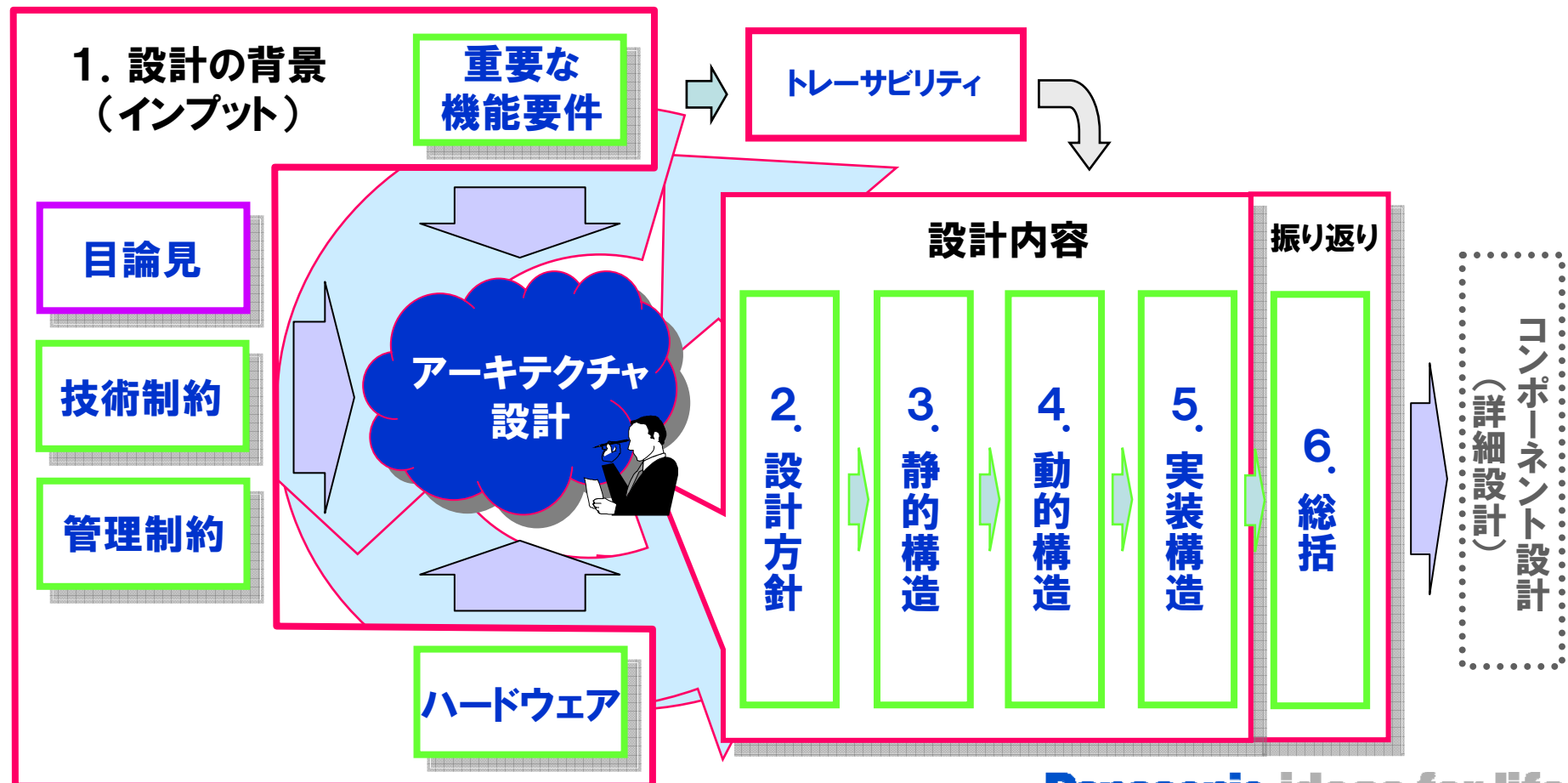
アーキテクチャ設計の位置付け

- 様々な施策が成功するためには、ソフトウェアの全体把握(アーキテクチャ設計)技術が根幹として必要



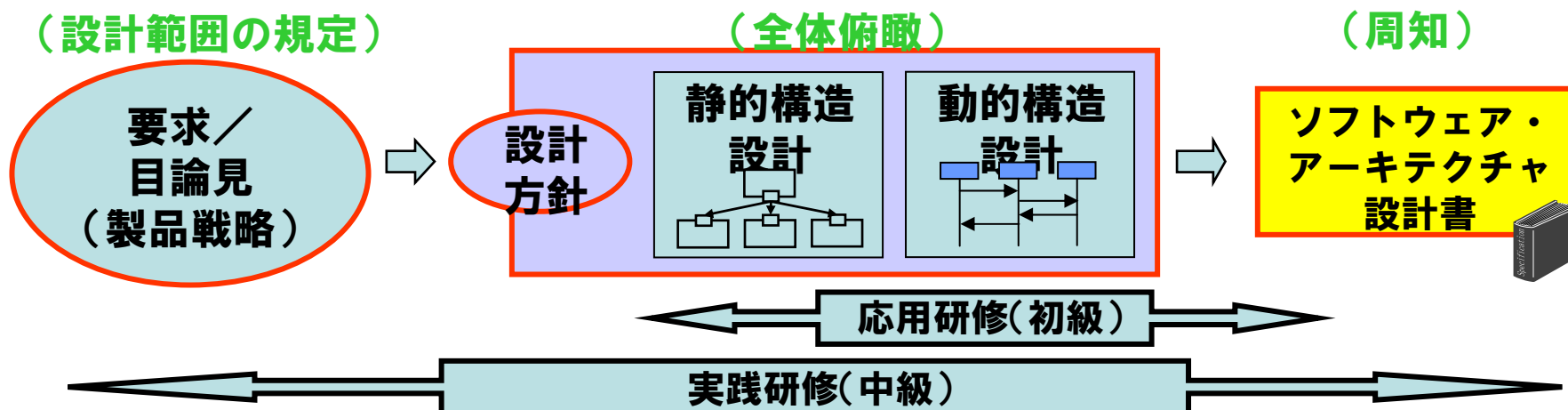
アーキテクチャ設計書への記載項目の規定

- 組込みソフトウェア設計の実態(動的設計の比重が高い)を踏まえた設計項目を規定
 - 設計ガイドライン, テンプレート, サンプル, ソフト構造の表記法の整備



アーキテクチャ設計研修への展開

- **ソフトウェア・アーキテクト育成応用研修(初級)**
 - － 研修目標：個々の設計手法を組み合わせて、全体構造が設計できる
 - 構造図（静的構造，タスク構造，状態遷移）として表現できる
- **ソフトウェア・アーキテクト育成実践研修(中級)**
 - － 研修目標：設計意図を他者に伝えることができる
 - 設計内容に加えて、設計上重要な決定事項(根拠，方針など)を明確に説明でき、実業務の中でアーキテクチャ設計書を策定することができる
- **研修成果物のレベルによりスキル認定を全社施策として実施**



今後の取組み

- **組込みソフトウェア開発に関する実装力向上・設計力向上施策については、一応の目処**
- **アーキテクチャ設計が浸透している部署から、モデル駆動開発(MDD)、プロダクトライン開発(SPL)導入を支援**
- **クラウド、ネットワークを含めた統合系ソフトウェア開発における上流工程の明確化**
 - **具体的な実施項目・実施方法を明らかにしたい**
 - **要件獲得、要求仕様記述、統合系アーキテクチャ設計など**

ご清聴ありがとうございました

Panasonic
ideas for life

