

ハードウェアODC分析の試み

キヤノン株式会社

品質統括センター 製品品質保証部

○瀬能 芳幸

仁科 秀一

E-mail : seno.yoshiyuki@mail.canon

アジェンダ

1. はじめに
2. ODC分析方法論をハードウェア開発に適用
 1. ODC分析とは
 2. 適用において重要なこと
3. ハードウェアODCタイプ属性の設計と検証
4. ハードウェア欠陥タイプ属性のシグナチャー
 1. あるべき姿
 2. 重大な設計変更
 3. 難易度が高い
5. 考察
6. おわりに

1. はじめに

- ❑ 開発終盤、ハードウェアチームから設計変更依頼
 - ソフトウェア開発に大きな影響
 - でも、受けざるを得ない
- ❑ ソフトウェア開発はODC分析を活用している
- ❑ これをハードウェア開発に展開し、開発終盤の設計変更を減らせないか？

2.1 ODC分析とは

- ❑ 既存の欠陥分析 → 信頼度成長モデル、根本原因分析
 - 欠陥には原因がある → 因果関係を分析できる方法論がない

❑ ODCとは、Orthogonal Defect Classification



IBMワトソン研究所
Ram Chillarege氏

直交

欠陥

分類

- ❑ 開発プロセスの各作業段階に関する属性で欠陥票を分類



2.1 ODC分析とは - 欠陥タイプ属性

□ ソフトウェア欠陥タイプの定義

重要度	欠陥タイプ	解説
	代入/初期化 Assignment	変数/定数値の代入あるいは初期化の欠陥
	チェック/条件 Checking	条件文での条件値、判定式の欠陥
	アルゴリズム Algorithm	処理手順、アルゴリズムの欠陥
	タイミング Timing	順番制御/リソースアクセスへの排他制御/スレッドの同期処理の欠陥
	インターフェイス Interface	ブロック/コンポーネント/プロダクト間、ハードウェアとソフトウェア間のコミュニケーションに関する欠陥
	機能/クラス Function	要求仕様との不整合、機能性/グローバルデータ構造の欠陥
	ビルド Build	ビルドプロセス、ライブラリ又はバージョン管理に関連する問題
	ドキュメント Document	マニュアル、技術文書、表示されるメッセージの欠陥

注：上記のタイプ属性に対して「誤り Incorrect」あるいは「漏れ Missing」という限定詞が付与される。

2.1 ODC分析とは - 欠陥タイプ属性

- ❑ 欠陥タイプは開発プロセスの各作業段階に関する

機能/クラス 漏れ

要件定義

インターフェイス 漏れ

タイミング 漏れ

機能 誤り

基本設計

インターフェイス 誤り

タイミング 誤り

アルゴリズム 漏れ

詳細設計

チェック/条件 漏れ

アルゴリズム 誤り

代入/初期化 漏れ

実装

代入/初期化 誤り

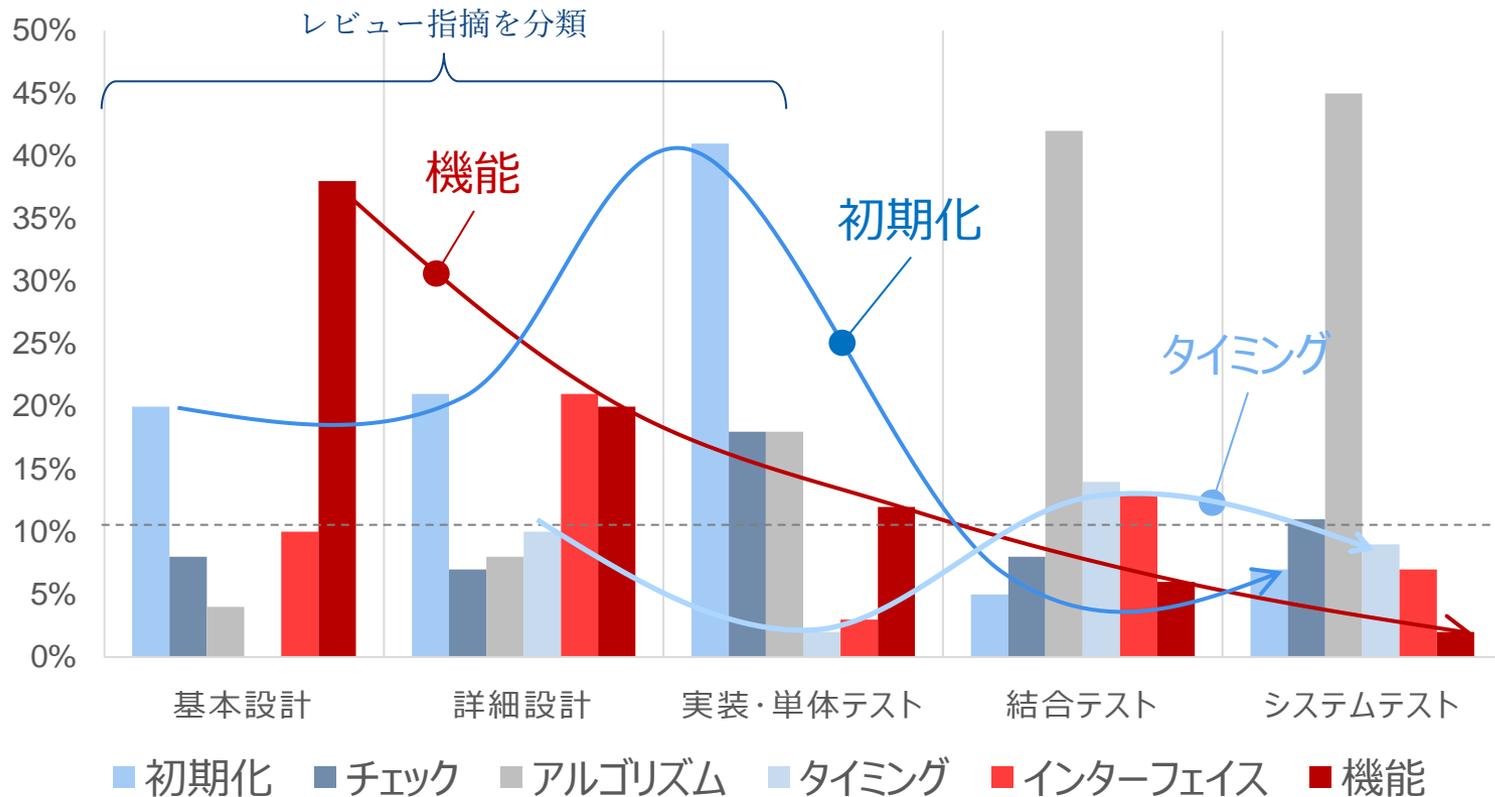
チェック/条件 誤り

2.1 ODC分析とは - シグナチャー

シグナチャーとは時系列変化

欠陥タイプシグナチャー

- 「機能」は最重要タイプ属性。基本設計で多く指摘され徐々に減少すべき
- 「初期化」は実装・単体テストで多く指摘され、結合テストは減少する。システムテストで若干増える
- 「タイミング」は、詳細設計と結合テスト・システムテストで検出され出荷前には収束する



2.2 適用において重要なこと

□ ODC分析のコンセプト

- 欠陥が生じてしまった事には、意味がある → Semantic Classification
- 欠陥タイプは 開発プロセスと密接な関係がある

□ 適用時に気を付けること

1. 属性値が重なり合わない
開発プロセスのある作業に関連しているので、属性値は重ならない
2. 欠陥タイプの属性値は開発プロセスのタスクと関連し一貫している
欠陥タイプ属性はプロセスにまたがって一貫性がある
3. 製品間で共通である
他の製品に適応できないと方法論として展開できない

3. ハードウェアODC欠陥タイプ属性の設計(1)

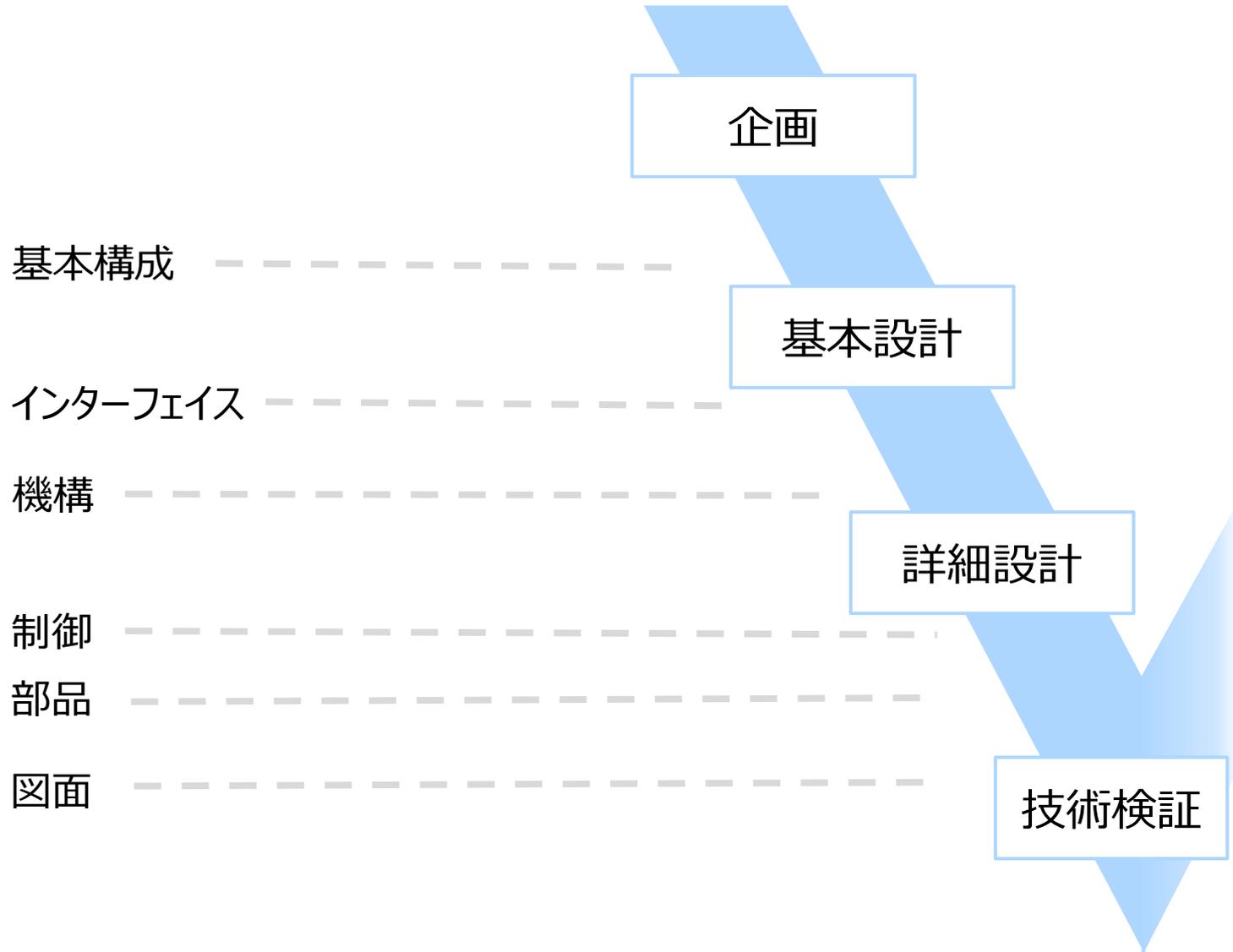
- ハードウェア開発プロセスを手掛かりに欠陥タイプを定義した。
- 有効性 プロセス一貫性 製品間の共通性を過去製品の欠陥票を用いて検証した。

欠陥タイプ	解説
図面 Drawing	部品図面化の際の配慮不足、ミス
部品 Parts	部品の強度・材料・形状の問題
制御 Control	ユニット、コンポーネントで発生する機能不良でパラメータ変更・条件等の制御などハードウェア変更なくファームで修正した問題
機構 Mechanism	ユニット、コンポーネントで発生する機構不良で強度・剛性などハードウェア修正を要する問題
インターフェイス Interface	ユニット、コンポーネント間で発生する機能不良で、複数ユニットにまたがるコミュニケーションやインターフェイス仕様変更を要する問題
基本構成 Fundamentals	設計思想・製品基本構成に影響する問題、仕様の漏れ

※ 限定詞は設けていない。

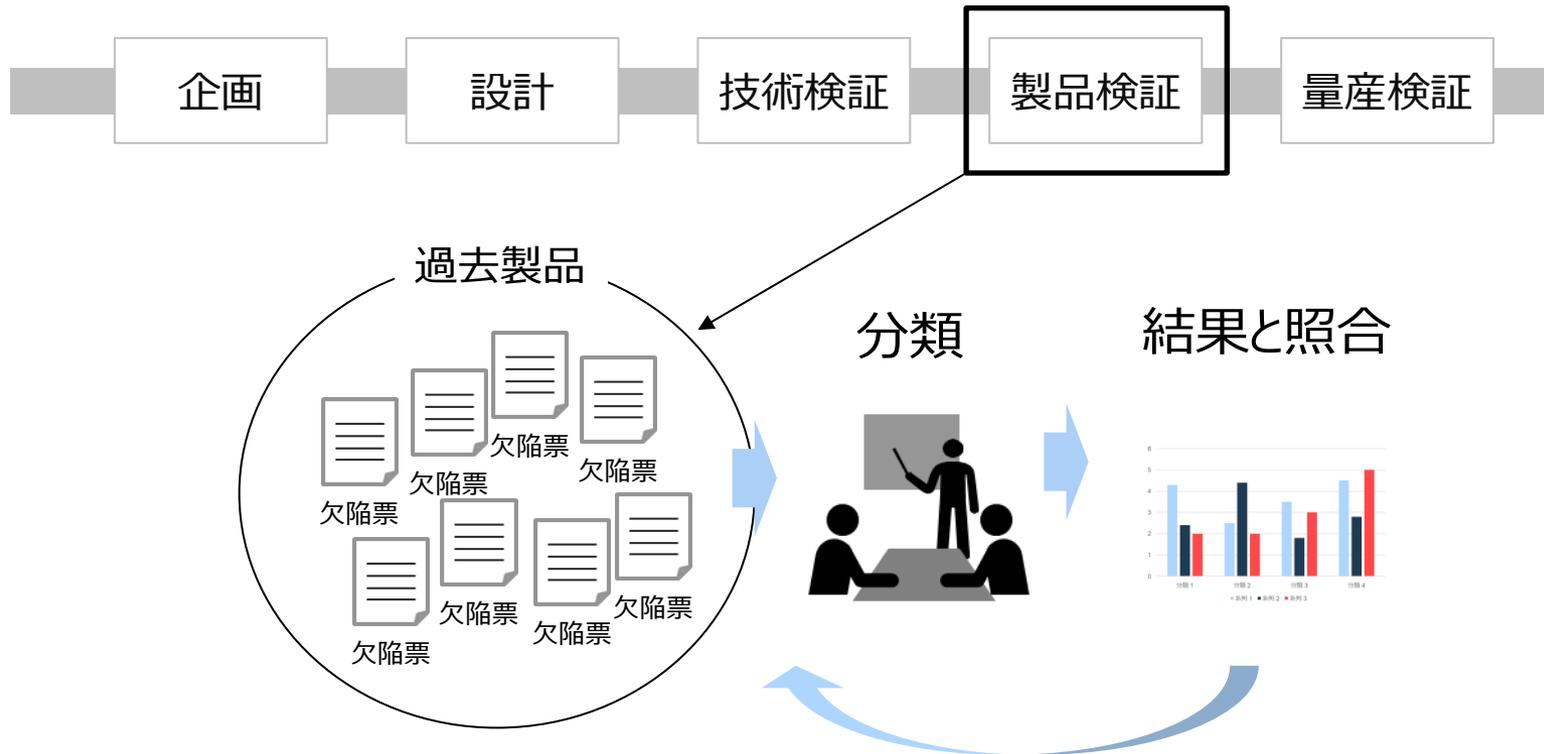
3. ハードウェアODC欠陥タイプ属性の設計(2)

欠陥タイプは開発プロセスの各作業段階に係る様に定義した



3. ハードウェアODC欠陥タイプ属性の検証

- ❑ 定義したハードウェア欠陥タイプが、実際の製品開発から出た欠陥票に当てはめ分類しシグナチャーが適切に働くかを検証した
- ❑ 本活動ではハードウェア開発プロセスの製品検証フェーズの欠陥票を対象とした



4.1 あるべきシグナチャー

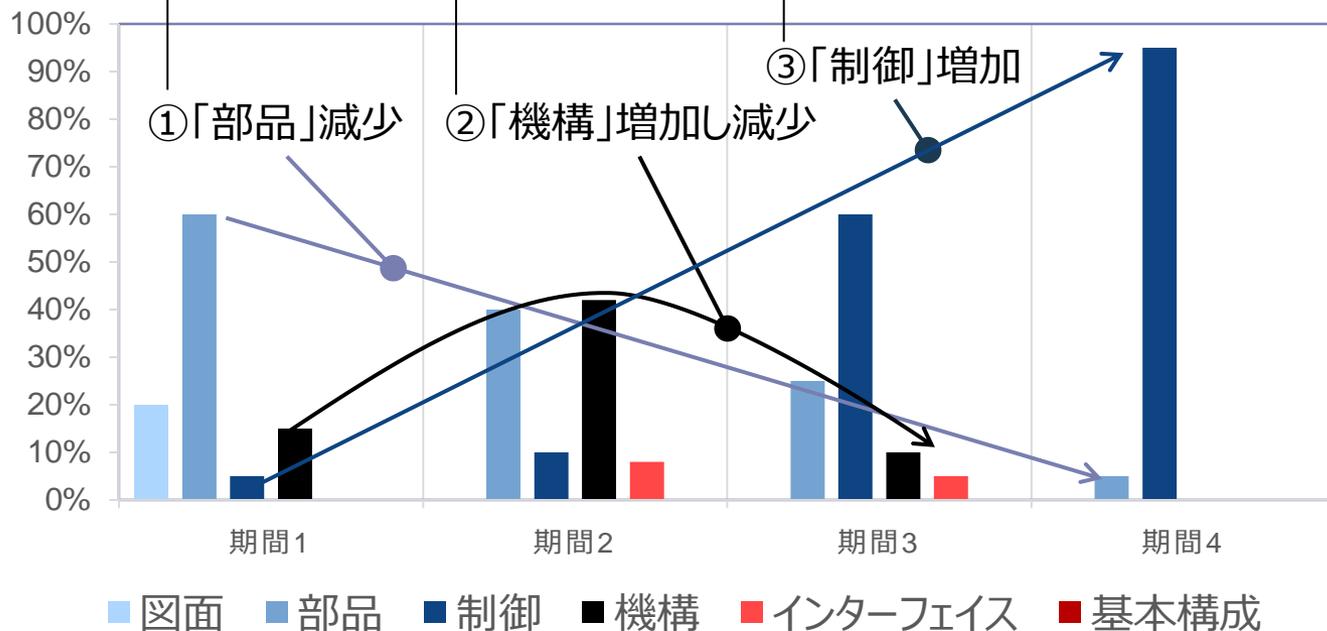
製品検証フェーズに入る段階で機能検証が十分であった製品が示すシグナチャー

- ① 「部品」がはじめ高い割合を占め、徐々に減少し 検証フェーズ終盤には収束する。
- ② 「機構」は検証が進むと割合が増加し、後半には収束する。
- ③ 「制御」は終盤に向け増加する。製品検証フェーズの終盤では「制御」における設定値や条件の修正が多くを占める。

欠陥タイプ シグナチャー

あるべき姿

製品検証フェーズ



4.2 重大な設計変更

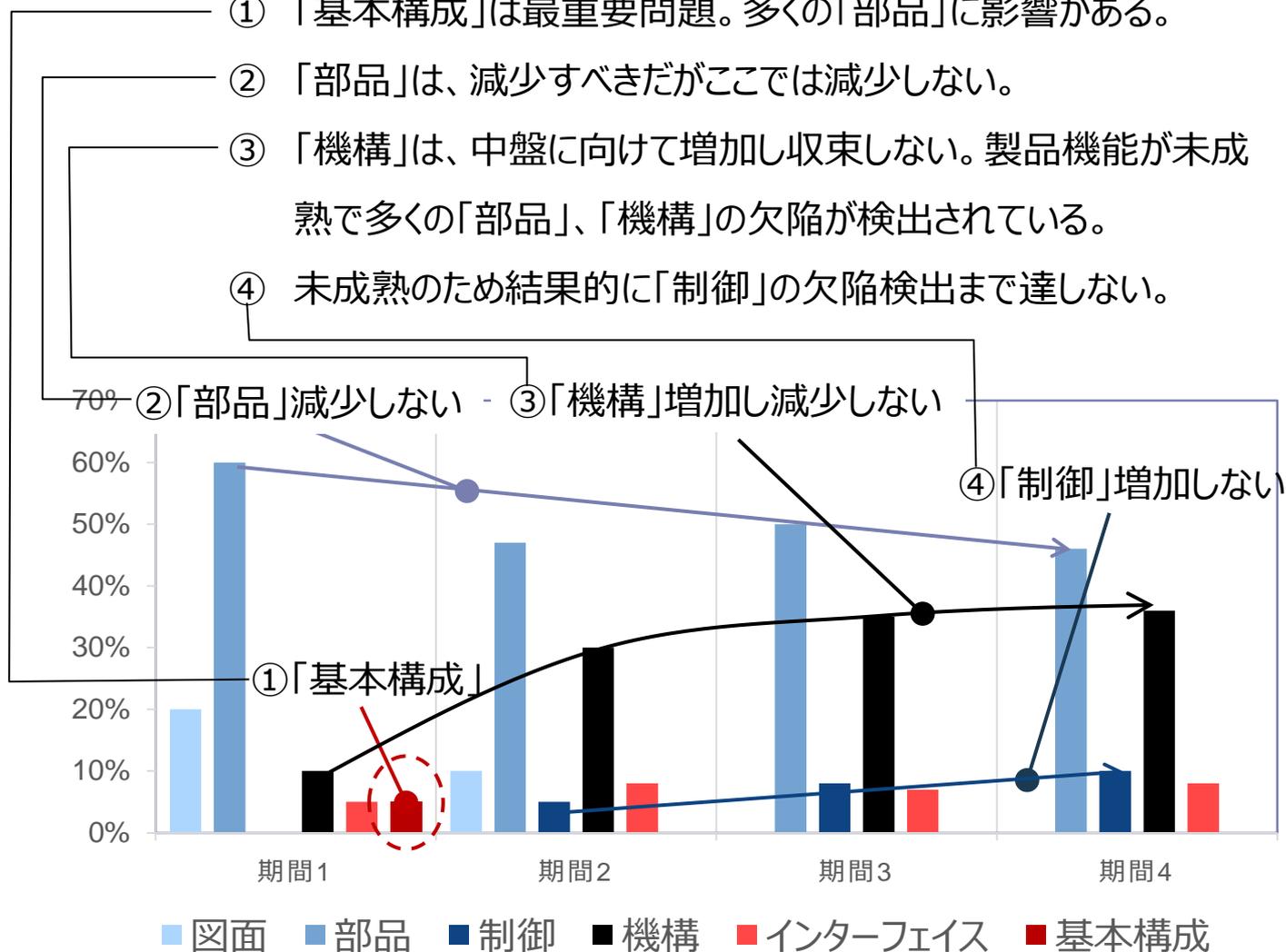
重大な設計変更が入ってしまった事態のシグナチャー

- ① 「基本構成」は最重要問題。多くの「部品」に影響がある。
- ② 「部品」は、減少すべきだがここでは減少しない。
- ③ 「機構」は、中盤に向けて増加し収束しない。製品機能が未成熟で多くの「部品」、「機構」の欠陥が検出されている。
- ④ 未成熟のため結果的に「制御」の欠陥検出まで達しない。

欠陥タイプ シグナチャー

重大な設計変更

製品検証フェーズ



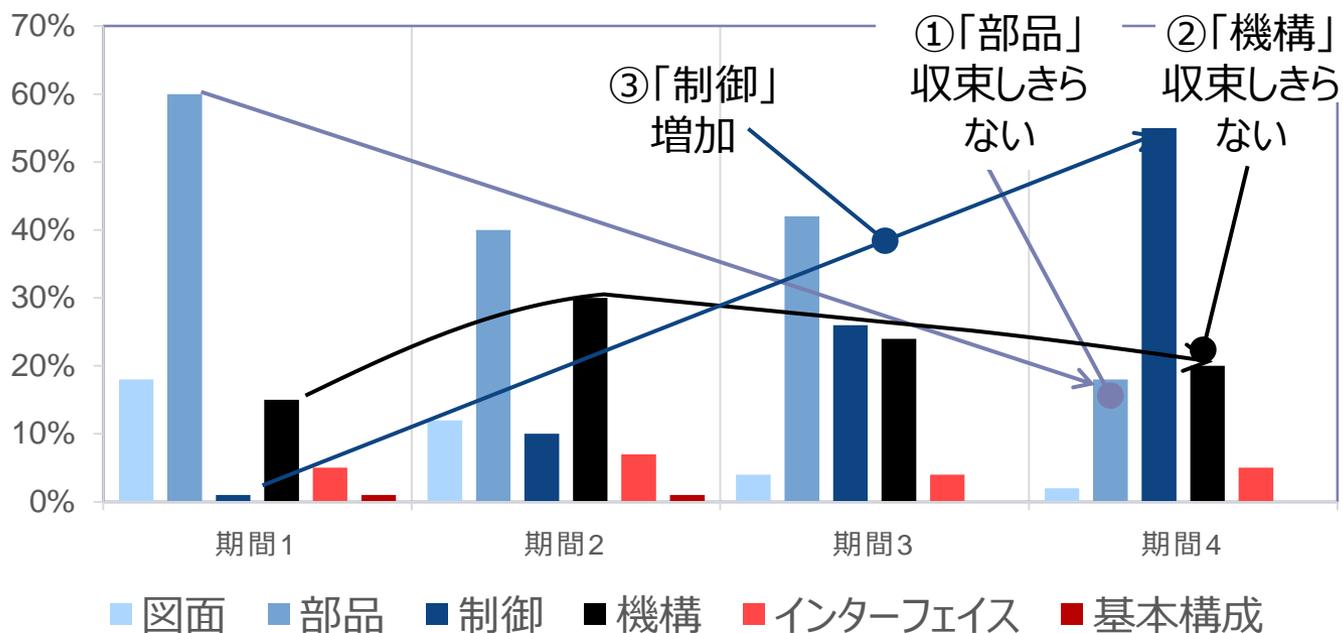
4.3 難易度が高い

新規性・難易度が高い製品は欠陥数が多く、検証フェーズに入る段階で課題が残る場合が多い。製品検証フェーズ終盤にあるべき姿と乖離する

- ① 「部品」は減少するが、その後収束しきらない。
- ② 「機構」は一旦増加し、その後収束しきらない。
- ③ 「制御」は終盤に向け増加する。難易度高によりハードウェア修正が入り、終盤においても「部品」「機構」の欠陥タイプが残ってしまう。

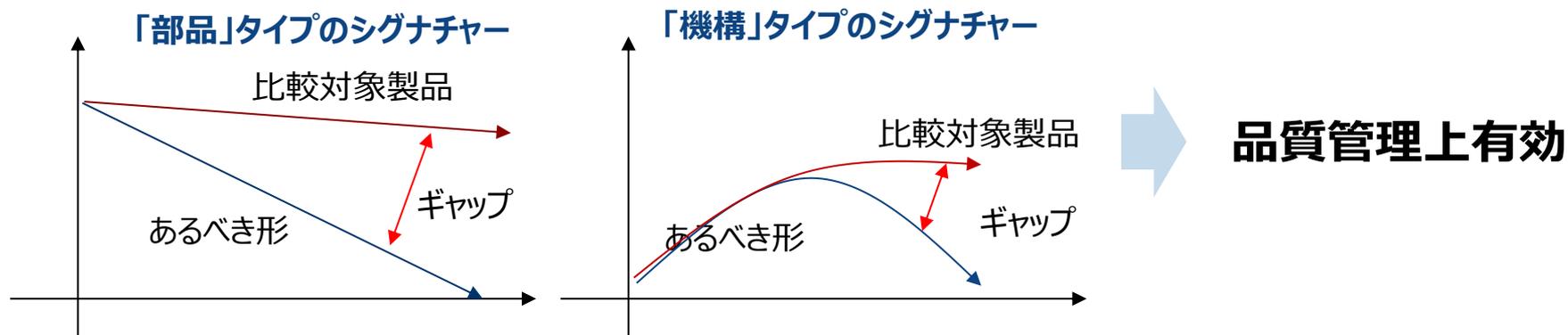
欠陥タイプ シグナチャー

難易度が高い
製品検証フェーズ

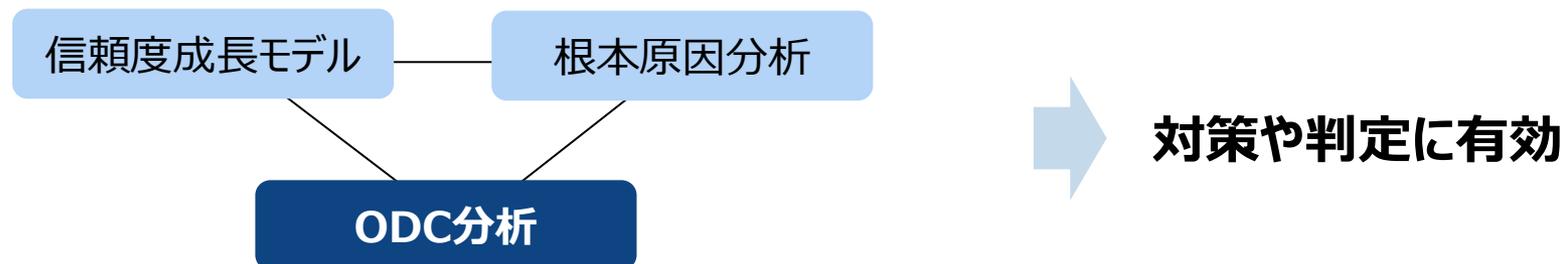


考察

- ❑ ハードウェア欠陥タイプのシグナチャーは、あるべき姿に対してギャップを示し、特徴ある動きを読み取ることができるので、ハードウェア開発における欠陥タイプを正しく定義できたと考えられる。これを活用すれば品質管理上で有効である。



- ❑ 欠陥タイプの重要度を考慮すると質的な把握ができる。従って、従来の信頼度成長曲線、根本原因分析に加え、ODC分析を行えば質的考察が可能となるので、対策や判定に有効に利用できる。



振り返って

- ❑ ここで示した欠陥タイプ定義に至るために実際の多くの欠陥票で分類し属性定義の**検証とチューニング**を数回**繰り返す**必要があった。
- ❑ ソフトウェア技術者にとって**ハードウェア欠陥票の記述は難解**であり、製品によって**記載レベルがさまざま**異なることが分かった。その上**解決までの期間が長く**、修正も様々な**変遷**があり最終的な修正でどれが有効であったか**判定が難しく**分類に手間取った。
- ❑ **ハードウェア技術者の協力**が無ければ、ハードウェア用の欠陥タイプを定義することは難しかった。

今後の課題

- ❑ **設計、技術検証そして、量産検証**についての欠陥に対する考察が残っており、早いフェーズからODC分析を適応した場合の検証を行うことが必要である
- ❑ この活動で定義したハードウェア開発用の欠陥タイプ属性定義は、引き続きより**広い製品を対象として調整し精度**を上げるべきである
- ❑ **ハードウェア開発プロセス**の質的な改善ができれば、開発の終盤でのソフトウェアに対する仕様変更が**質的に改善できる**ことを検証すべきである
- ❑ **ハードウェア技術者とソフトウェア技術者の協力が重要**である

ご清聴ありがとうございました