

# SQuBOK V2設計開発領域について

わしざき ひろのり  
鷺崎 弘宜

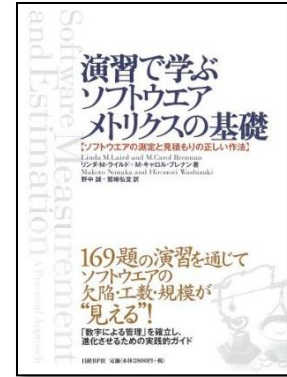
早稲田大学グローバルソフトウェアエンジニアリング研究所  
国立情報学研究所GRACEセンター

Twitter: @Hiro\_Washi [washizaki@waseda.jp](mailto:washizaki@waseda.jp)

<http://www.washi.cs.waseda.ac.jp/>



- 早稲田大学グローバルソフトウェアエンジニアリング研究所所長・准教授
- 国立情報学研究所客員准教授
- **SQuBOK策定委員会、開発領域世話役**
- SQiP研究会運営小委員会副委員長
- 情報規格調査会ISO/IEC SC7/WG20主査
- IEEE CS Japan Chapter Chair
- SEMAT Japan Chapter Chair
- SamurAI Coding 2014-15 Director
- 会議運営: ASE, SEKE, SPLC, APSEC, AsianPLOPほか
- プログラム委員多数: ASE, ICST, ...
- 論文多数: ICSE, ASE, ICST, ...



# 目次

- 設計開発領域の概要
- 工程に共通の品質技術
- 工程に個別な品質技術
- 専門的品質特性の技術
- 品質技術の整理

# あなた/あなたの組織は、 品質のプロか？

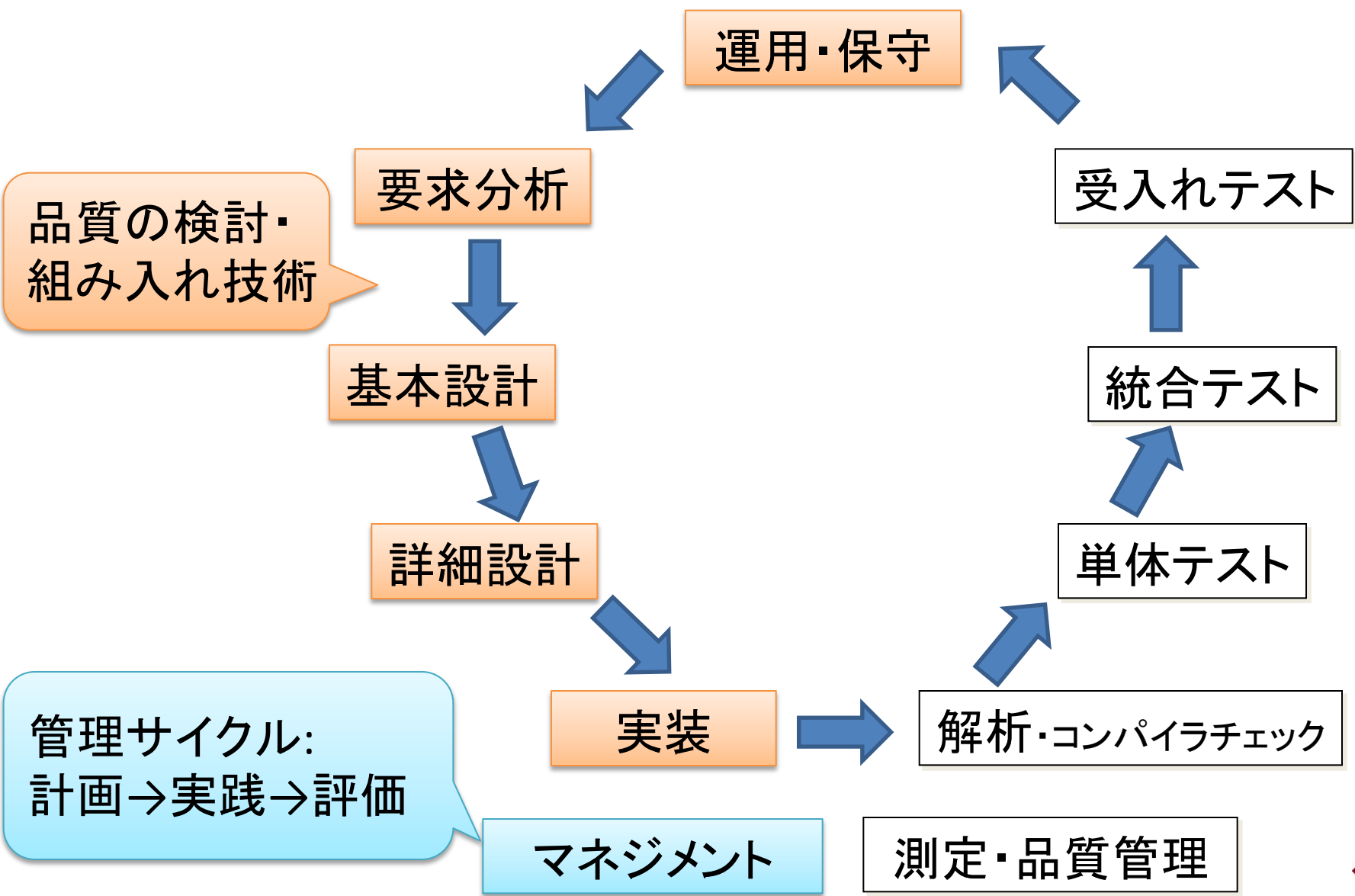
- テストやレビューの技術さえ収めれば良い「わけではない」
  - 知識体系にのっとった客観的証明と整理
- 品質特有の難しさ [鷺崎15]
  - ライフサイクルに対する横断性
  - 品質特性の多面性、トレードオフ
  - プラットフォーム特有の品質
  - 専門的な品質特性

# あなた/あなたの組織は、 品質のプロか？

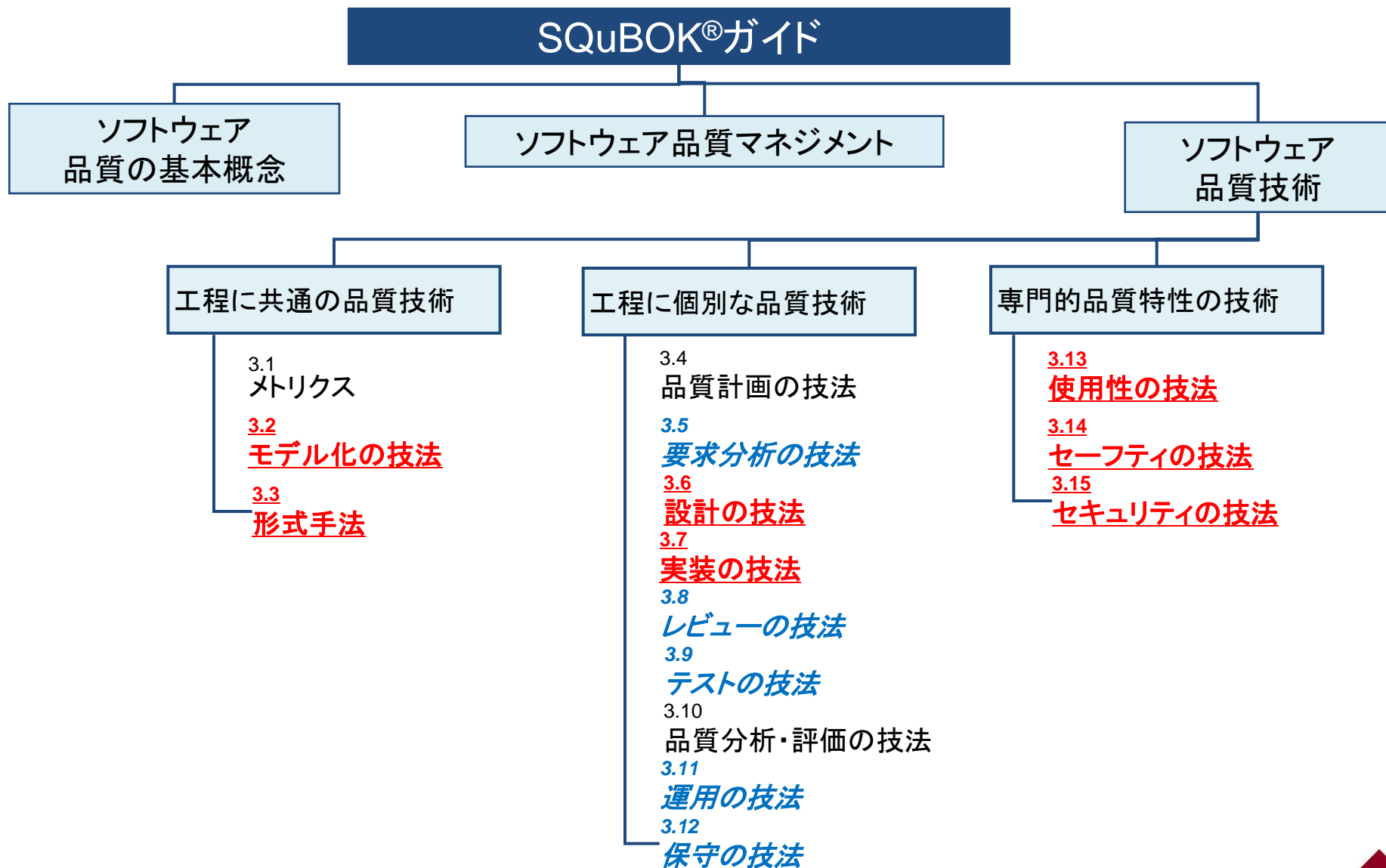
- テストやレビューの技術さえ収めれば良い「わけではない」
  - 知識体系にのっとった客観的証明と整理
- 品質特有の難しさ [鷺崎15]
  - ライフサイクルに対する横断性
  - 品質特性の多面性、トレードオフ
  - プラットフォーム特有の品質
  - 専門的な品質特性

# SQuBOK V2.0 設計開発領域 拡充

## 「すべての人に品質の意識を！」



# SQuBOKガイド V2.0: 品質技術



# 例: アーキテクチャ方法論(トピック)

(主にソフトウェアに関する)品質要求の獲得から、アーキテクチャを設計し、評価するまでの方法論の総称

- **【目的】** 品質要求を満たしたアーキテクチャを獲得
- **【方法】**
  - 品質要求を明らかに: QAW(Quality Attribute Workshop)
  - 品質要求に基づくアーキテクチャ設計: ADD(Attribute Driven Design)
  - アーキテクチャの品質要求満足分析: ATAM(Architecture Trade-off Analysis Method)
  - アーキテクチャの経済的評価: CBAM(Cost Benefit Analysis Method)
- **【効果】** 品質要求の不備を避けて...
- **【参考文献】** 『実践ソフトウェアアーキテクチャ』
- **【関連トピック】** アーキテクチャパターン, ...



# 目次

- 設計開発領域の概要
- 工程に共通の品質技術
- 工程に個別な品質技術
- 専門的品質特性の技術
- 品質技術の整理

# 再掲: あなた/あなたの組織は、 品質のプロか？

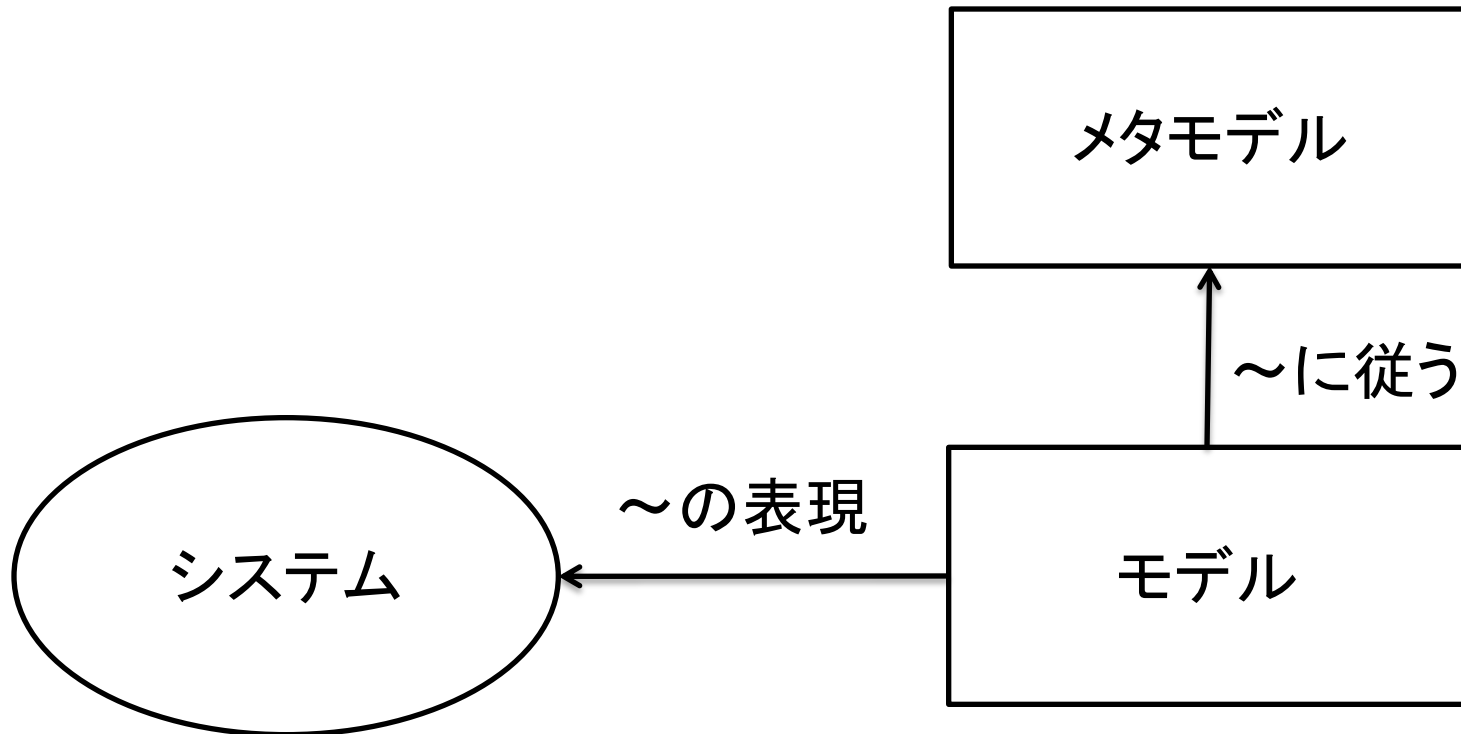
- テストやレビューの技術さえ収めれば良い「わけではない」
  - 知識体系にのっとった客観的証明と整理
- 品質特有の難しさ [鷺崎15]
  - ライフサイクルに対する横断性
  - 品質特性の多面性、トレードオフ
  - プラットフォーム特有の品質
  - 専門的な品質特性

# 工程に共通の品質技術

- モデル化の技法
  - 離散系のモデル化技法: UML, SysML, 構造化チャート PAD
  - 連続系のモデル化技法: Matlab/Simulink
  - ドメイン特化言語
- 形式手法
  - 形式仕様の技法
  - 形式検証の技法

# モデルとは

「現実世界の何か」に対する「認知上の目的」  
のためのコスト上効率的な「代用物」



# システム・ソフトウェア開発のモデル

- システム: SysML, ADL
- アプリケーション: UML
- 制御: ブロック線図、Simulink
- 機械、電気: CAD、HDL、製図



```

window.onload = setEventHandler;
function setEventHandler() {
    $("reg_type").onchange = calcPrice;
    ...
    $("reg_addcart").onclick = addCart;
};

function calcPrice() { ... };
function addCart() {
    if(isValidInput()) {
        reqRunTrans();
    } else {
        alert("Invalid user inputs");
    }
};

function reqRunTrans() {
    new Ajax.Request("runTrans.php", {
        method: "GET", parameters: getParams(),
        onSuccess: succeeded });
};

function succeeded() { disableAll();
    jumpToConfirm();
};

```

Price: \$500

Type

All days ▾

Attendee

Regular ▾

Payment

Early ▾

Quantity:

Add to Cart

# 目次

- 設計開発領域の概要
- 工程に共通の品質技術
- 工程に個別な品質技術
- 専門的品質特性の技術
- 品質技術の整理



# 再掲: あなた/あなたの組織は、 品質のプロか？

- テストやレビューの技術さえ収めれば良い「わけではない」
  - 知識体系にのっとった客観的証明と整理
- 品質特有の難しさ [鷺崎15]
  - ライフサイクルに対する横断性
  - 品質特性の多面性、トレードオフ
  - プラットフォーム特有の品質
  - 専門的な品質特性



# 工程に個別な品質技術(前半)

- 要求分析の技法
  - 要求抽出: ステークホルダー識別, 要求開発・Openthology
  - 要求分析: 機能要求分析, 非機能要求分析, 品質機能展開, 要求可変性分析
  - 要求定義: USDM
  - 要求の妥当性確認
- 設計の技法
  - 方式設計の技法: 部品化の技法, アーキテクチャパターン, アーキテクチャ方法論(品質駆動設計・評価ほか), DSM, フレームワーク
  - 詳細設計の技法: テスト駆動開発, デザインパターン, 設計原則

# リダイレクトシステム開発の例 [繁在家08]

## • 要求

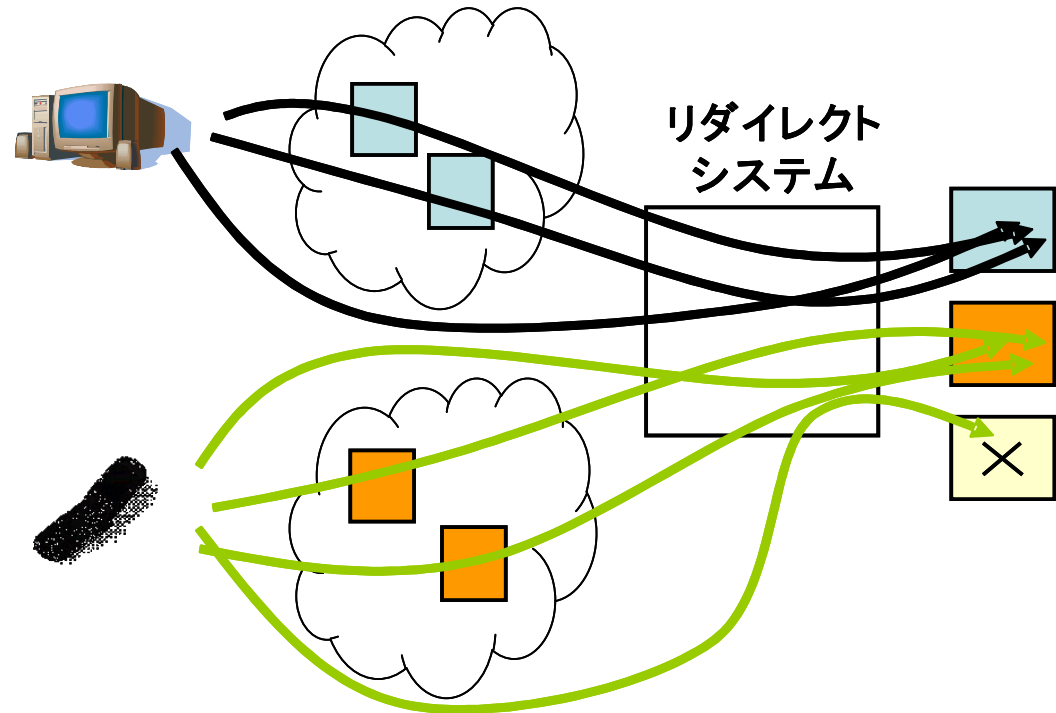
品質要求を明確化できるか？

- シナリオ
- ゴール指向分析

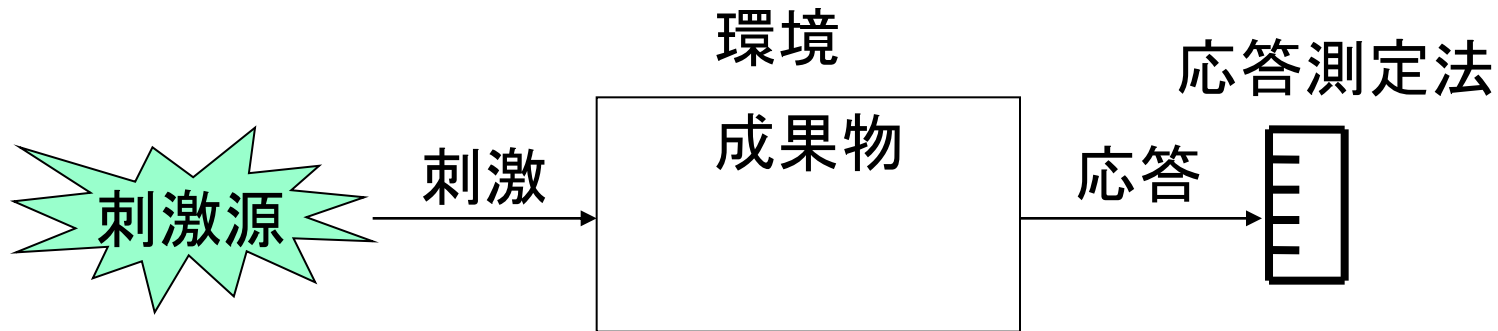
## • 設計

品質を着実に組み入れられるか？

- パターン
- 品質駆動設計・評価



# 要求: 品質シナリオ



A7

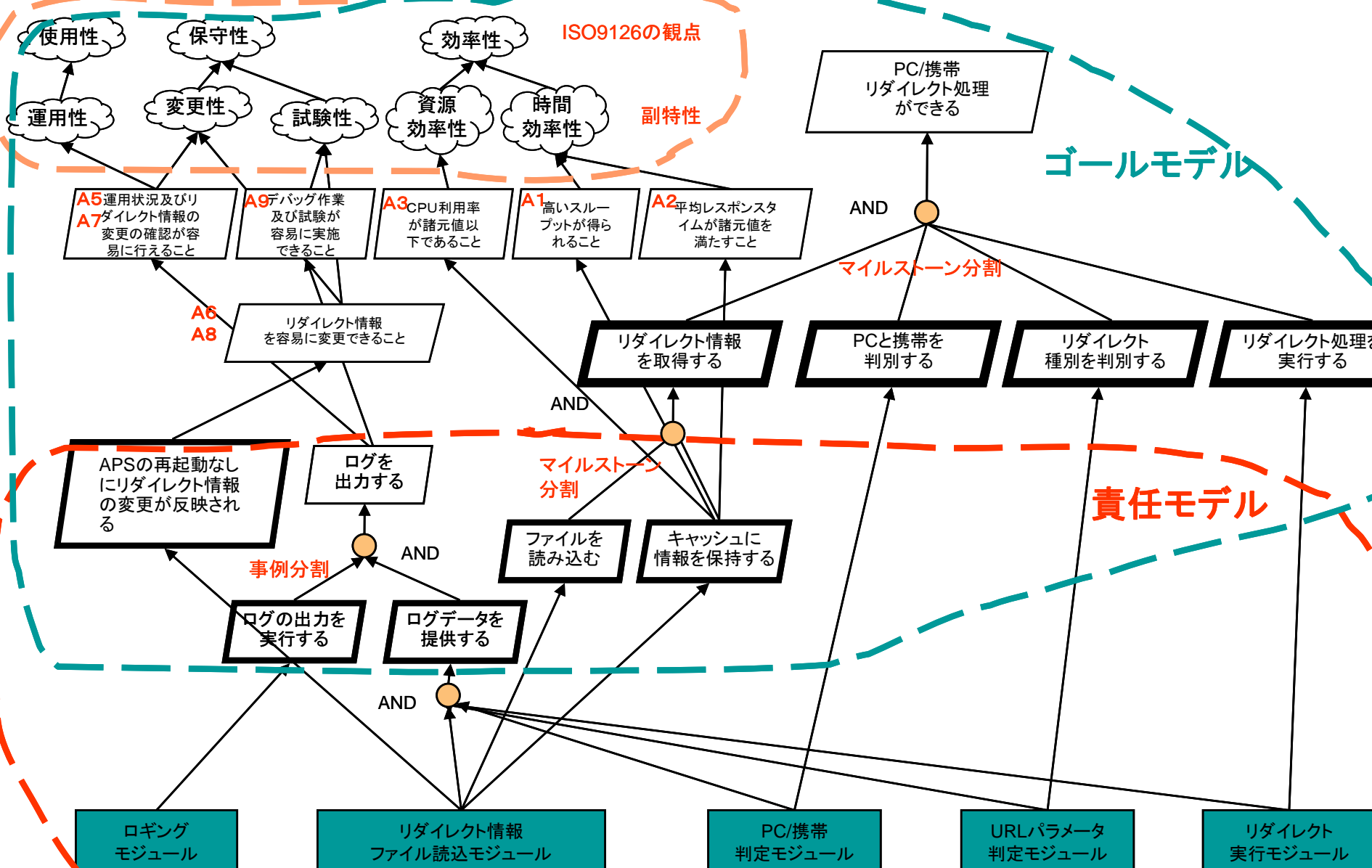
(環境) 通常稼働時に  
(刺激) リダイレクト情報に変更があった場合 (応答) 運用を止めずに反映できる。

A1

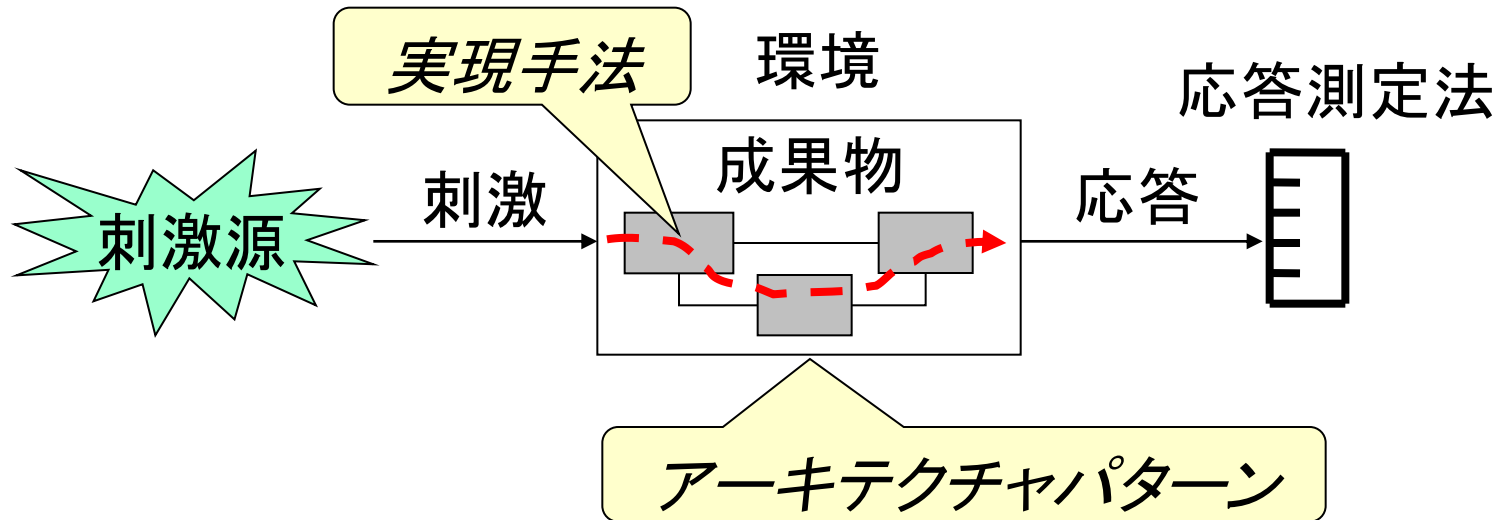
(環境) 通常稼働時に  
(刺激) JMeterで高負荷をかけた場合  
(応答) HTTPリクエストを秒間1000トランザクション処理できる。

# 要求:ゴール指向分析

品質特性群



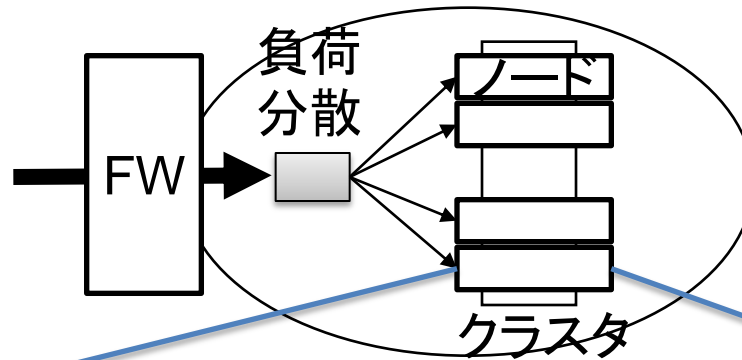
# 設計: パターンと品質駆動設計



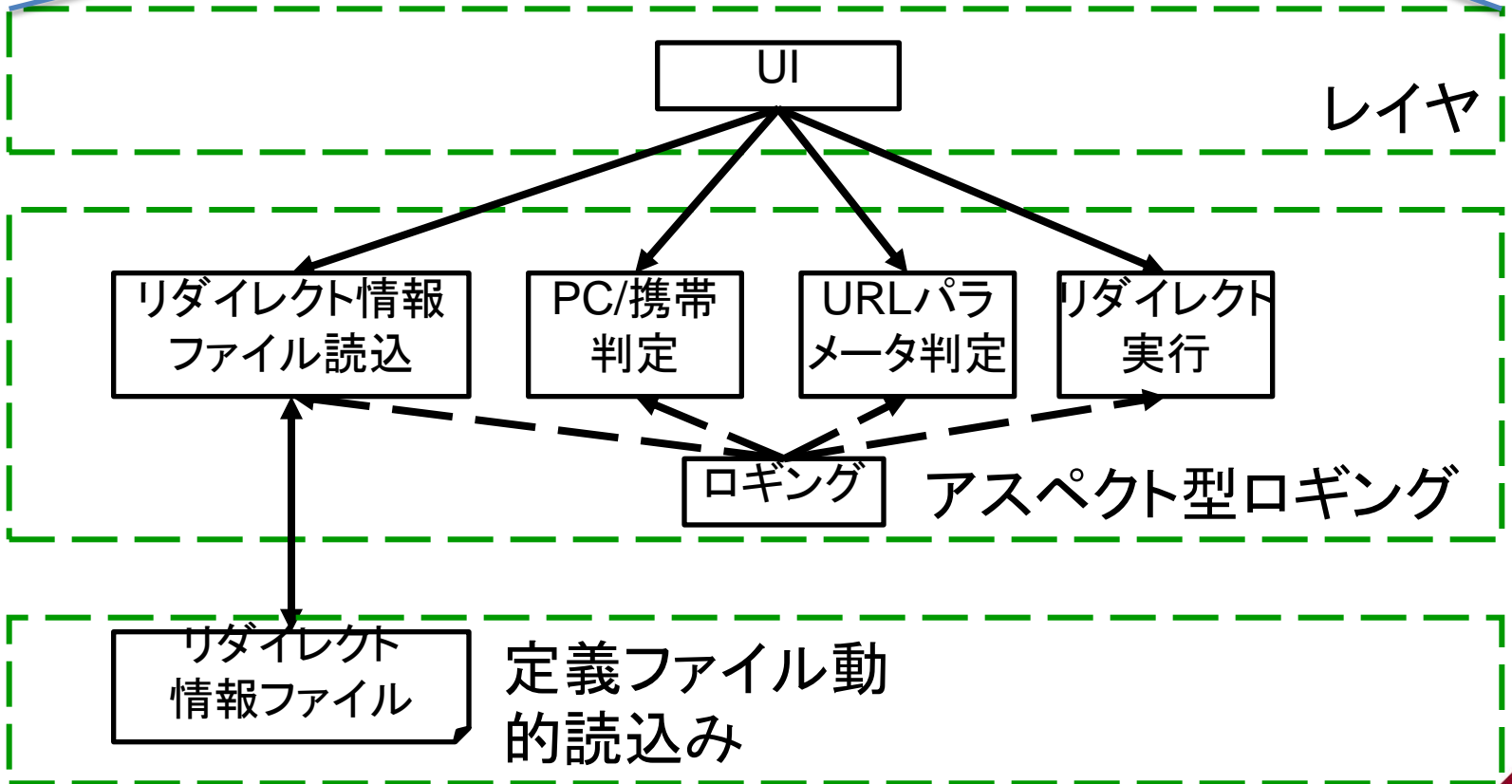
アーキテクチャパターン

	定義ファイル 動的読み込み	多重化・ 負荷分散
A7. リダイレクト情報の 変更を停止無しで反映	↑	
A1. HTTPリクエストを秒間 1000トランザクション処理	↓	↑

システム・基盤



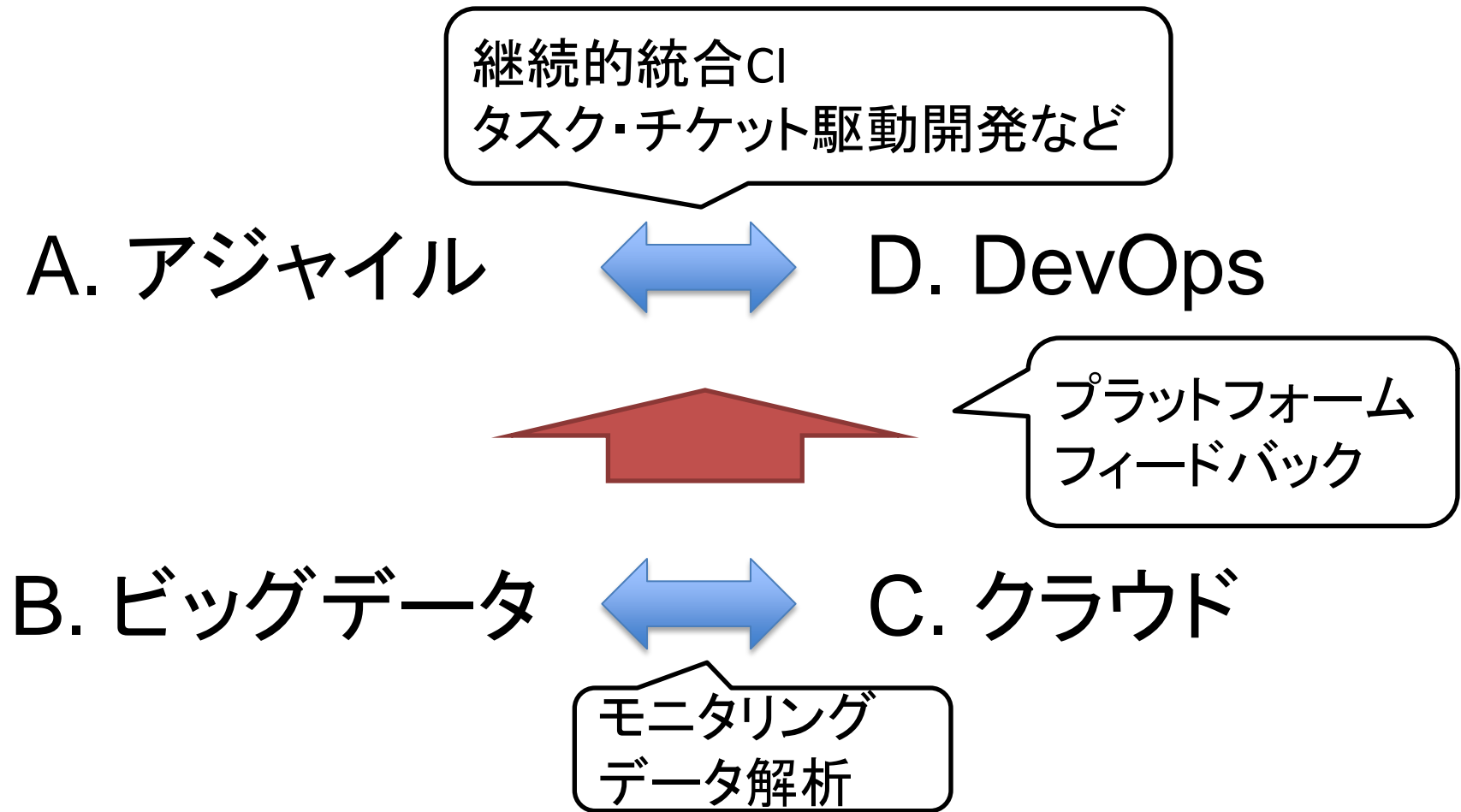
アプリケーション



# 工程に個別な品質技術(後半)

- 実装の技法
  - コーディング規約, リファクタリング, 防御的プログラミング・契約による設計, 統合開発環境, **継続的統合 CI**
- 運用の技法
  - データ品質, 仮想化, **クラウド**, ソフトウェア若化
- 保守の技法
  - プログラム理解, リエンジニアリング, リバーズエンジニアリング, コードクローン分析

# 開発と運用にみる次の”ABCD” [青山14]





# 目次

- 設計開発領域の概要
- 工程に共通の品質技術
- 工程に個別な品質技術
- 専門的品質特性の技術
- 品質技術の整理



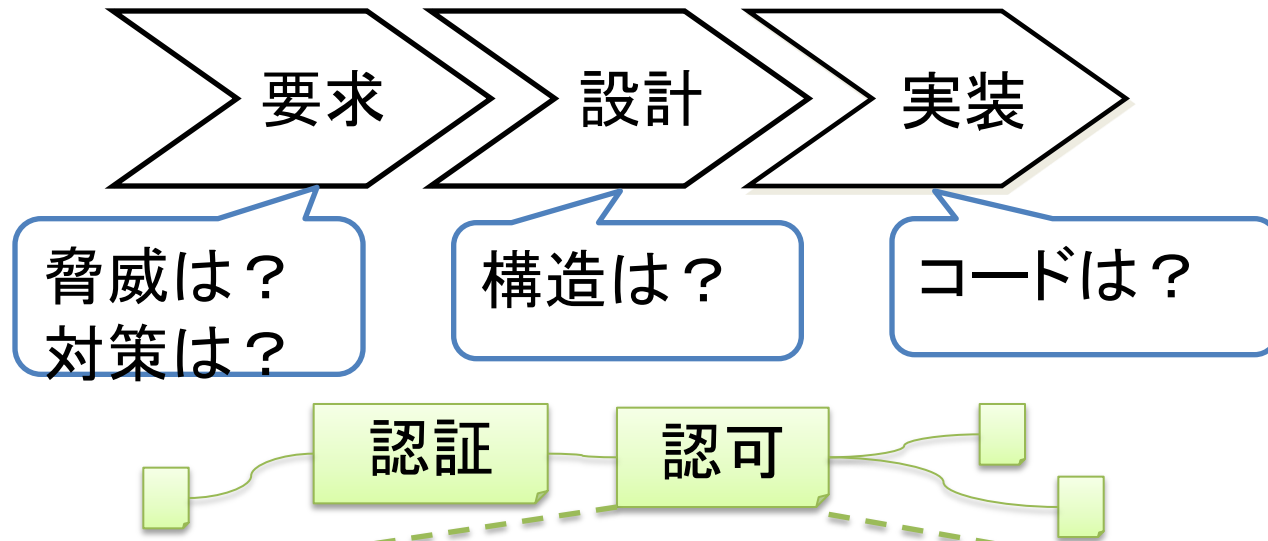
# 再掲: あなた/あなたの組織は、 品質のプロか？

- テストやレビューの技術さえ収めれば良い「わけではない」
  - 知識体系にのっとった客観的証明と整理
- 品質特有の難しさ [鷺崎15]
  - ライフサイクルに対する横断性
  - 品質特性の多面性、トレードオフ
  - プラットフォーム特有の品質
  - 専門的な品質特性

# 専門的品質特性の品質技術

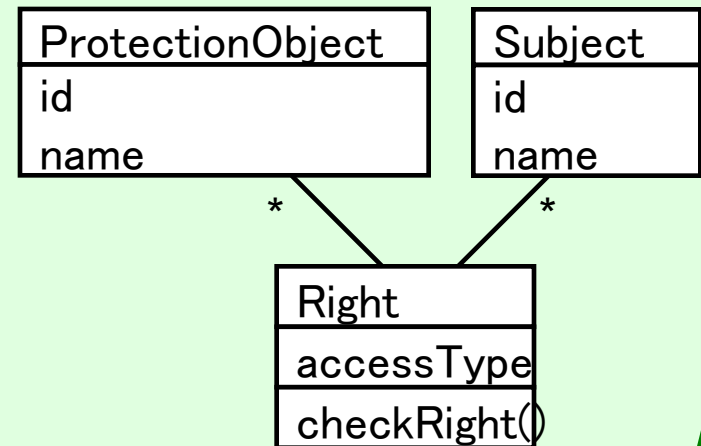
- ユーザビリティの技法
  - ユーザビリティテスト
- セーフティー(安全性)の技法
  - リスクアセスメント技法(ISO/IEC 31010), リスク低減法
- セキュリティの技法
  - セキュリティ要求分析, セキュリティ設計(SDL、脅威分析・モデリングなど), **セキュリティパターン**, セキュアコーディング, セキュリティテスト

# 例えばセキュリティパターン [Shiroma10]



## 認可パターン

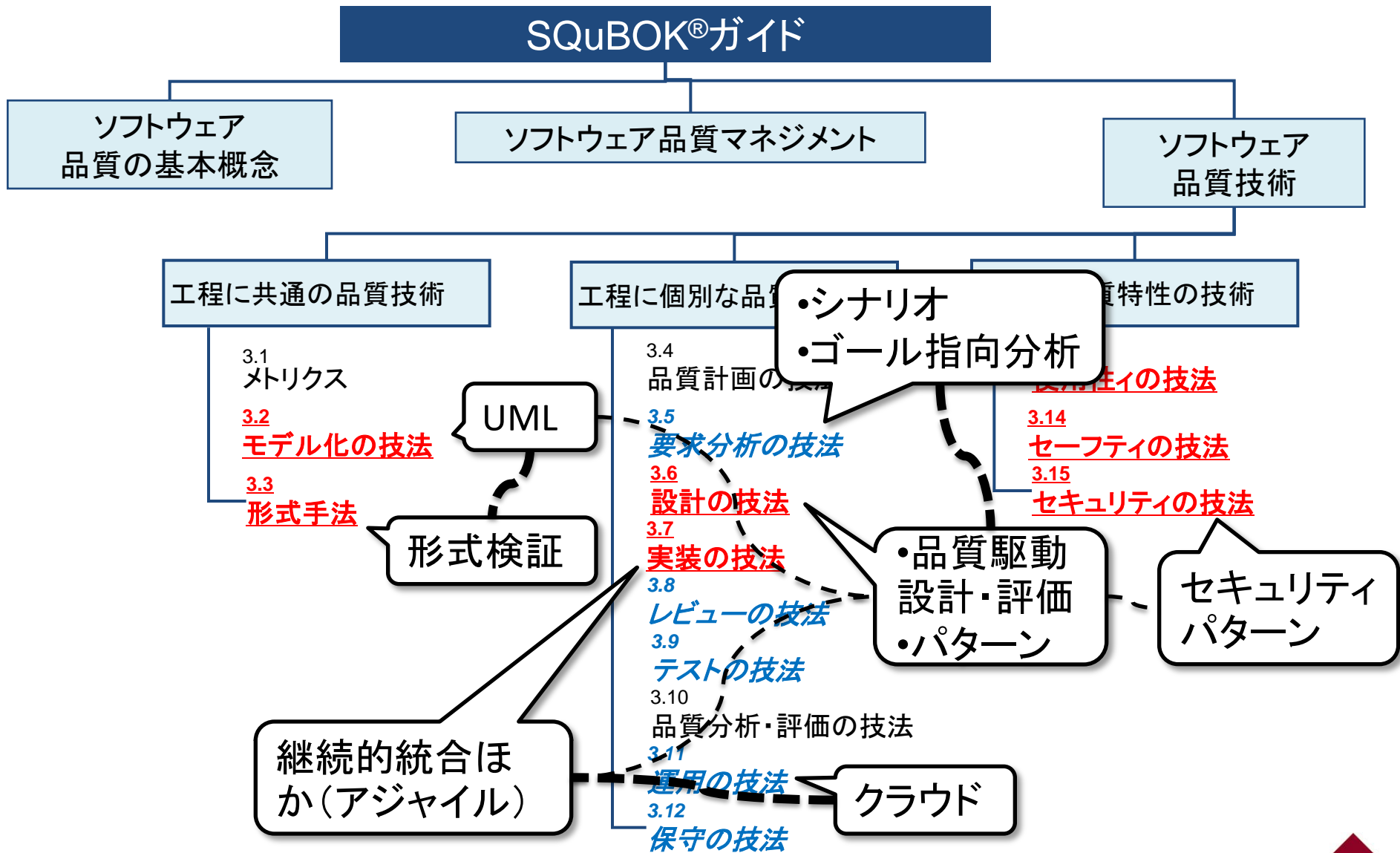
問題: 認可の構造が資源と独立・・・  
解決: 資源にアクセスしようとする・・・



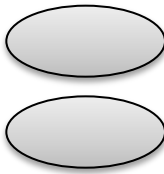
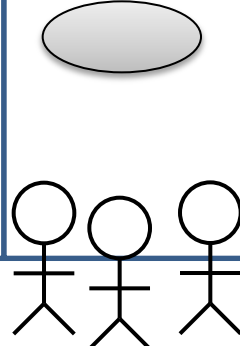
# 目次

- 設計開発領域の概要
- 工程に共通の品質技術
- 工程に個別な品質技術
- 専門的品質特性の技術
- 品質技術の整理

# 取り上げた技術のSQuBOK上のつながり

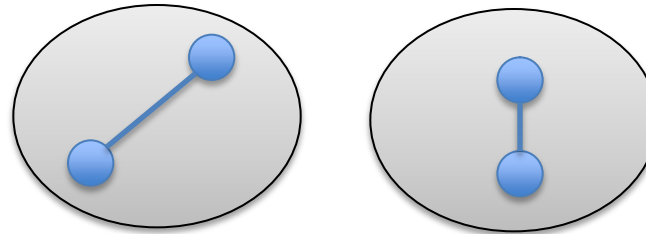


活動

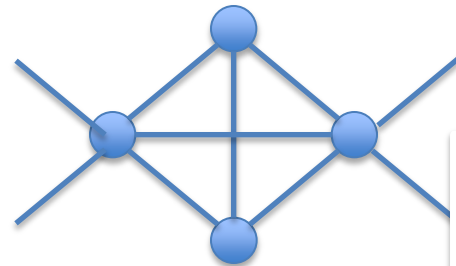
To Do	Doing	Done
		

組立実践

パターン、  
プラクティス、手法

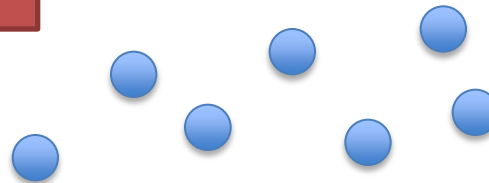


知識の体系



フィードバック

知識の島々



<div style="text-align: center;">主に扱う品質</div> <div style="text-align: center;">知識・技術</div>	機能性	信頼性	使用性	効率性	保守性	移植性	セキュリティ
離散系モデリング・UML	X				X	X	
形式検証	X	X		X			X
品質シナリオ	X	X	X	X	X	X	X
ゴール指向要求分析	X	X	X	X	X	X	X
品質駆動設計・評価	X	X	X	X	X	X	X
アーキテクチャ、デザイン、セキュリティパターン	X				X	X	X
継続的統合CIほか	X	X			X		
クラウド	X	X		X	X	X	





# 個人・組織が品質プロであるために

- ライフサイクルに対する横断性
  - 保証のみならず、設計開発における組み入れ重要
  - 工程を超えた知識の整理、組み立て・プラクティス化、蓄積、洗練
  - 組織外の場: SQiP研究会、SQuBOKユーザ会ほか
- 品質特性の多面性、トレードオフ
  - 知識・技術×品質マトリクスで整理しよう
  - トレードオフへの考慮、それに適した技術把握
- プラットフォーム特有の品質
  - ABCDなど
- 専門的な品質特性
  - 使用性、セーフティ、セキュリティの一通りを概観