

ソフトウェア品質保証 入門

高品質を実現する考え方とマネジメントの要点



安田 勝通



奈良隆正

紹介者：柏 祐太郎

構成

1. ソフトウェア品質保証の考え方
2. ソフトウェア開発と品質保証の方法
3. フィールド保証
4. 品質検証・評価の主要技術
5. 人材育成
6. いま求められている品質保証

本日のメニュー

1. ソフトウェア品質保証の考え方

1. ソフトウェアの品質
2. ソフトウェア品質保証の問題点
3. 品質保証の考え方
4. ソフトウェア検査のあり方
5. ソフトウェア品質保証の組織と体系

1.1 ソフトウェアの品質



品質とは

- 品質の定義はたくさんある
 - 品質保証の観点からの品質の定義：
『品質は**ユーザ**にとっての**価値**である』
 - 近代的な品質管理における品質の定義：
『品質は**ユーザ**にとっての**満足度**である』 ←日本企業で多用
 - ワインバーグの定義：
『品質は**誰か**にとっての**価値**である』 ←人によって変わる
 - ジェームズマーチンの定義：
『品質は**システムが本稼働する時**， **真のビジネスニーズ**にあっているかによる』 ←時期によって変わる

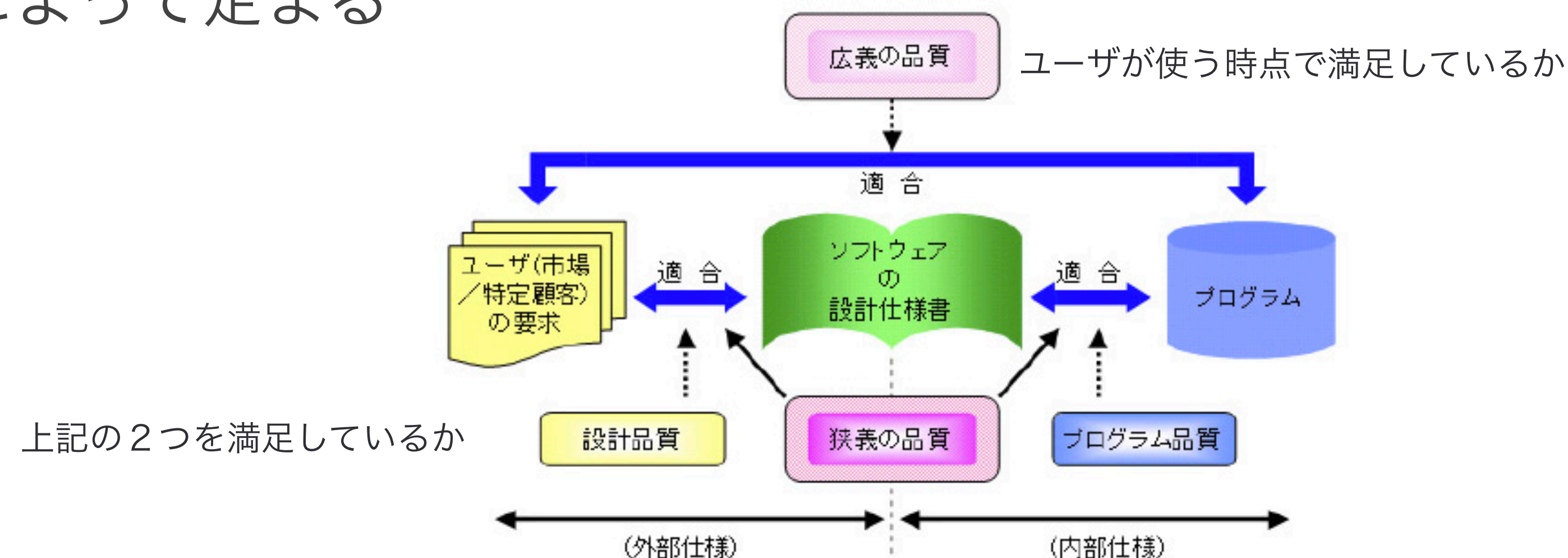
品質とは

- つまり . . .

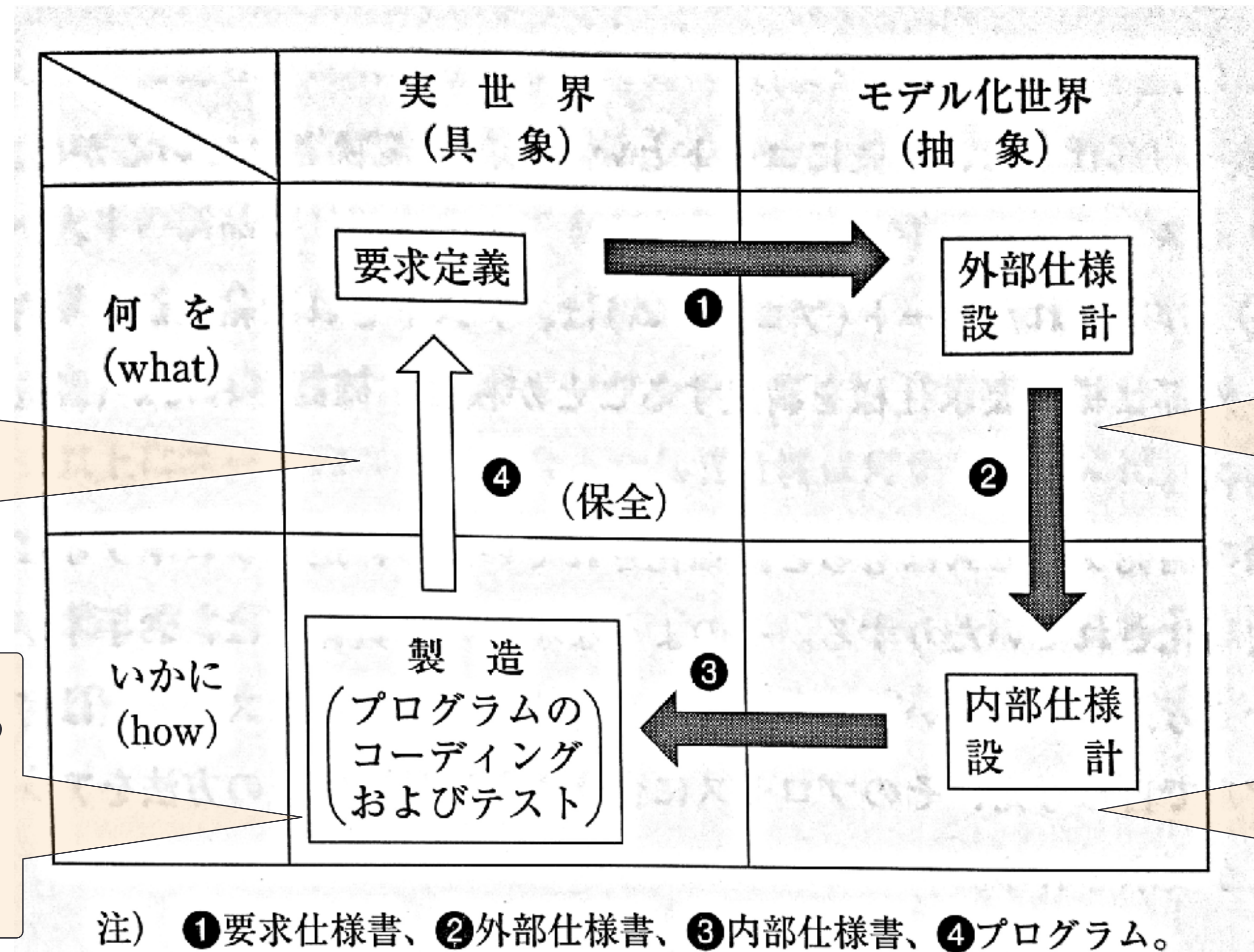
最終的にソフトウェアを**ユーザが使う時点**で、
ユーザのニーズを**どれだけ満足**しているか

ただし、**ユーザが誰か**でその価値が変わる

- ソフトウェアの製品規格を満足するために必要なこと
 - 設計仕様がユーザの要求に整合していること
 - **設計品質**と呼ぶ
 - 外部仕様[ユーザから見た仕様]によって定まる
 - プログラムが設計仕様通りに正しく動作すること
 - **プログラム品質**と呼ぶ
 - 内部仕様[開発者から見た仕様]によって定まる



モデル化した開発プロセス



整理されていない
ユーザのニーズを整理

要求仕様を
ソフトウェア外部仕様
として厳密に定義

内部仕様をコードという
実行可能な形に変換
およびテストで検証

外部仕様を
内部仕様に変換

モデル化した開発プロセス

- 以下の5つを検証することにより高品質を実現

【要求定義プロセス】

1. ユーザニーズが要求仕様に正しく反映されているか

【外部仕様プロセス】

2. 要求仕様は厳密に外部仕様として展開されているか

【内部仕様プロセス】

3. 外部仕様は正しく矛盾なく内部仕様にマッピングされているか

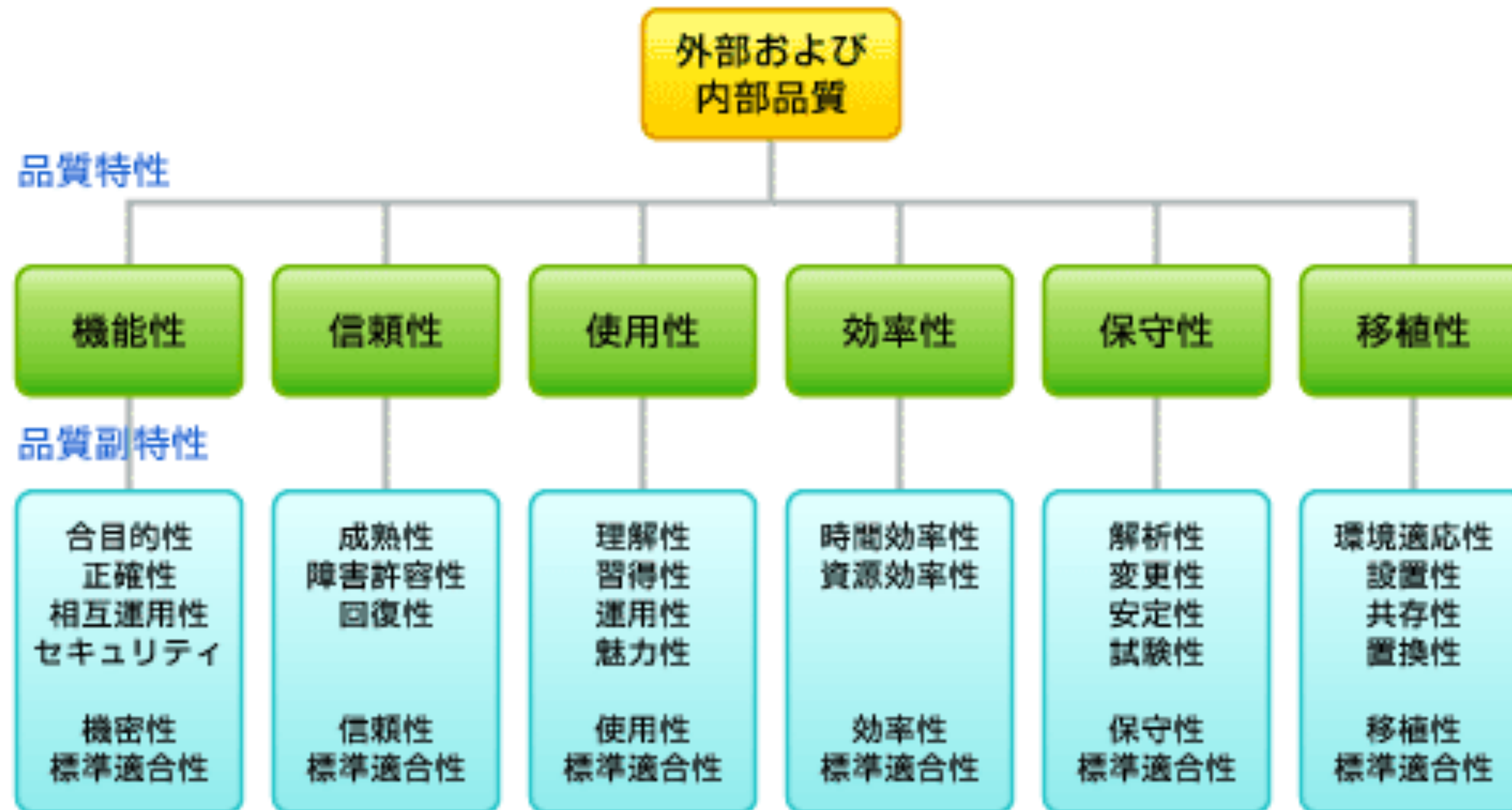
【製造プロセス】

4. 内部仕様は、正確にコードという形態に変換されているか

5. 作成されたコードは、テストされた結果がそれぞれ内部仕様、外部仕様、要求仕様を満足することが確認されたか

品質特性とは

- ISO/IEC 9126 ソフトウェア品質特性



- 機能とその特性に影響する特性群
 - 合目的性 (suitability)
 - » ユーザ改良要求件数, 仕様の改訂率
 - 正確性 (accuracy)
 - » ユーザから要求された数値の精度の実現率, 説明書と実動作の合致率
 - 相互運用性 (interoperability)
 - » データ形式の合致度, インターフェースとプロトコルの合致度
 - セキュリティ (security)
 - » 暗号化率, アクセス履歴保有率
 - 標準適合性 (compliance)
 - » 関連するすべての規格のうち, 遵守している項目の割合

- ある状況がある時間続いたときに、ソフトウェアがどの程度機能するかに影響する特性群
 - 成熟性 (maturity)
 - » 平均故障間隔 (MTBF), 平均故障寿命 (MTTF)
 - 障害許容性 (fault tolerance)
 - » 誤入力・誤操作検出力, 単位故障数あたりのダウン回数
 - 回復性 (recoverability)
 - » 平均修復時間 (MTTR), 平均ダウンタイム (MDT)
 - 標準適合性 (compliance)
 - » 関連するすべての規格のうち, 遵守している項目の割合

- 利用するのにかかる手間、
個人の努力などに影響する特性群
 - 理解性 (understandability)
 - » デモが用意されている機能の数
 - 習得性 (learnability)
 - » マニュアル装備率, 学習機能装備率
 - 運用性 (operability)
 - » メッセージ的確度, キータッチ数, レスポンス時間
 - 魅力性 (attractiveness)
 - » 利用者にとって魅力的である機能の数
 - 標準適合性 (compliance)

- ソフトウェアの性能や
それに要するリソース量に影響する特性群
 - 時間効率性 (time behavior)
 - » 処理速度 (CPU実行時間, 入出力処理時間など),
処理能力 (トランザクション処理件数など)
 - 資源効率性 (resource behavior)
 - » 資源使用量 (CPU使用量, メモリ使用量など),
資源使用率 (単位時間あたりのCPU使用率, メモリ使用率など)
 - 標準適合性 (compliance)
 - » 関連するすべての規格のうち, 遵守している項目の割合

- 何らかの変更を加えるのにかかる
手間に影響する特性群
 - 解析性 (analyzability)
 - » 誤り箇所識別率, 診断機能の装備率
 - 変更性 (changeability)
 - » 1000ラインあたりの修正実施時間, 障害 1 件あたりの修正実施時間
 - 安定性 (stability)
 - » モジュール結合度
 - 試験性 (testablity)
 - » 1000ラインあたりのテスト実施時間, 障害 1 件あたりのテスト実施時間
 - 標準適合性 (compliance)

- 別の環境にソフトウェアを移行させる
可能性に影響する特性群
 - 環境適応性 (adaptability)
 - » 適用可能機種・OS数, 変更せずに使用できるデータの割合
 - 設置性 (installability)
 - » ソースコードの変更率, 移植ツールの適用できる割合
 - 共存性 (co-existence)
 - » 共通資源を他のソフトウェアと共有できる割合
 - 置換性 (replaceability)
 - » 置換前のソフトウェアとくらべて, 実現できる機能や性能の割合
 - 標準適合性 (compliance)