

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

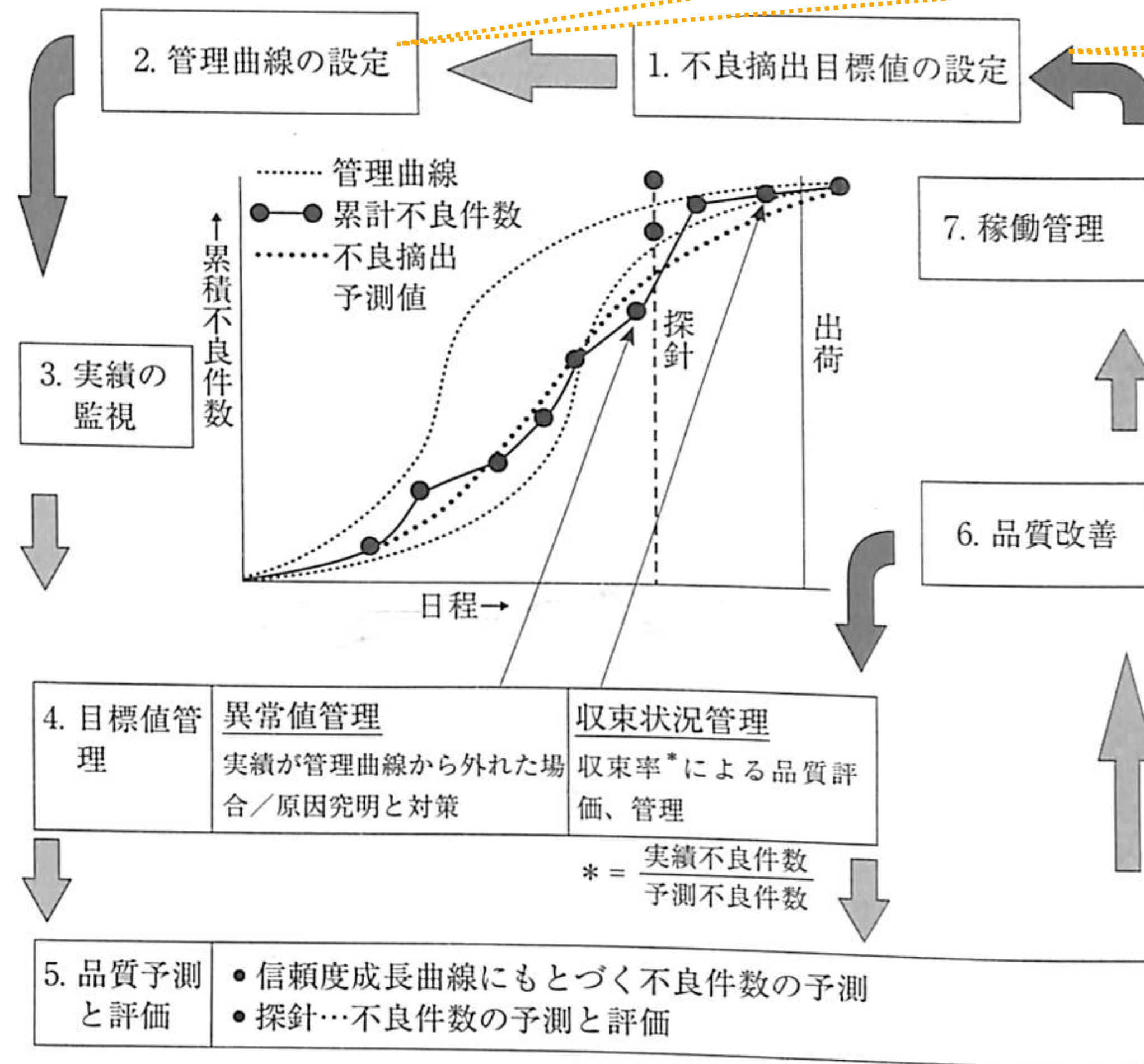
紹介者：柏 祐太郎

2. ソフトウェア開発と品質保証の方法

1. 開発基盤の整備
2. 開発計画レビューと検査計画
3. 高品質設計の要件
4. デザインレビューとドキュメント検査
- 5. テストと品質管理**
6. 製品検査の準備
7. 製品検査
8. システム検査

品質目標値の管理

3



テスト開始前に実施

不良摘出目標値を設定し，管理曲線を設定

- 不良摘出目標値は，過去の類似案件から設計・検査部門が設定

表 2.7 不良摘出目標値の設定モデル例

$$Y = \gamma_1 \cdot \gamma_2 \cdot \dots \cdot \gamma_n \cdot C \cdot X^\beta$$

Y: 作り込み不良件数

$\gamma_1 \cdot \gamma_2 \cdot \dots \cdot \gamma_n$: 品質要因の係数

C: 類似製品の平均不良件数(件/KS)

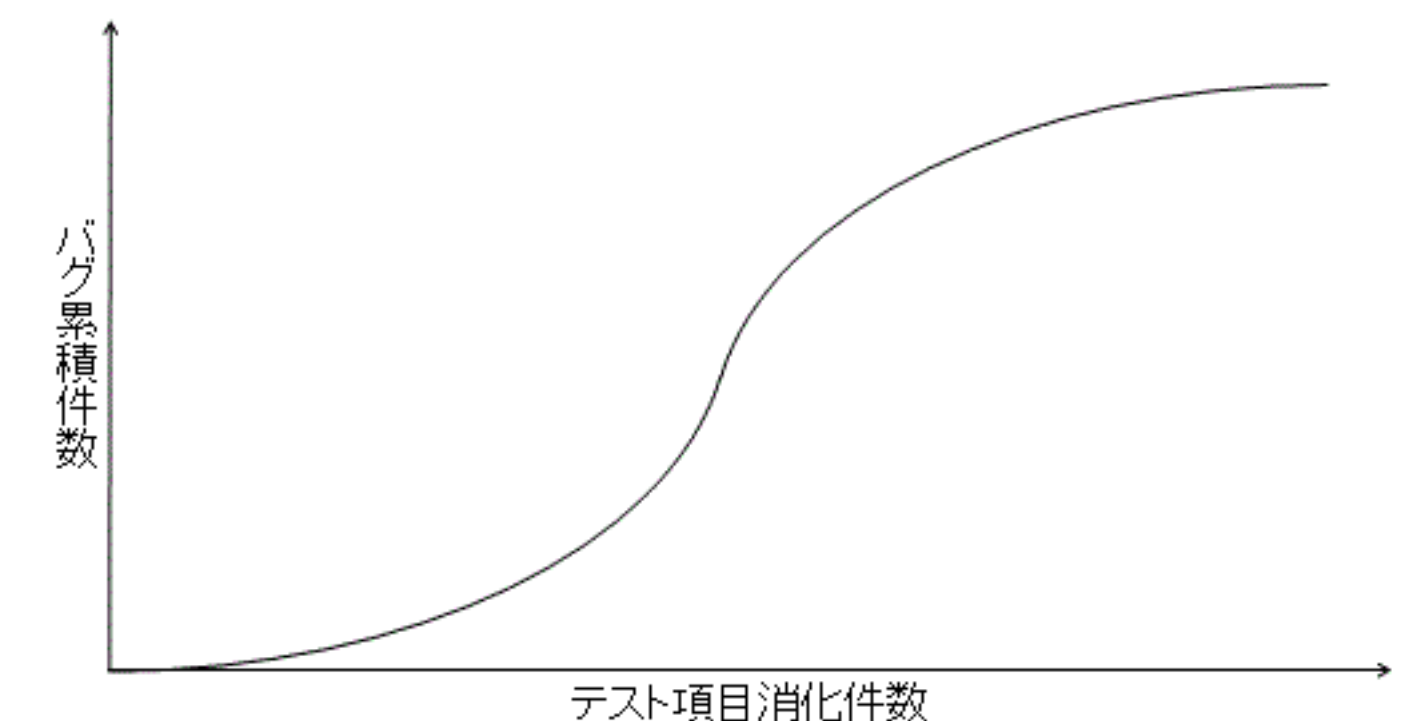
X: 開発規模(KS)

β : 定数

品質要因例

- 処理内容
- 処理形態
- 開発形態
- 設計工数
- 開発チーム能力

- 管理曲線は，信頼度成長曲線から求める



補足：信頼度成長曲線

- テスト段階や運用段階の実測データからソフトウェアの信頼度を推定する技法
 - ソフトウェア故障発生時間モデル
 - » ソフトウェア故障間隔（BTBF）を予測する手法
 - フォールト発見数モデル
 - » テスト時間と障害数の関係に着目し，数理モデルから総障害数や潜在障害数を予測する手法

※詳細は次回，調査・説明します！

探針

- 製品全体から一部を抜き取って、その不良の割合から全体の品質を予測すること

- 抜き取り検査のアナロジー

- 効果

- » 残障害数を把握できる

- » 探針での障害内容からソフトウェアの弱点を把握できる

① 探針項目として、検査項目の **10%～20%** をサンプリングする。

サンプリング方法は層別抽出方法である。

② 探針項目から出た不良に対して不良率を次の式で求める。

$$\text{標本不良率: } \rho = \frac{r}{n} \cdots (1)$$

n : 探針項目数 r : 不良件数

③ 二項確率紙または次の近似式により母不良率の信頼限界の上限値と下限値を求める。

(近似式は不良5件以上の時に適用)

$$\text{上限値: } P_U = \rho + u(\alpha) \sqrt{\frac{\rho(1-\rho)}{n}} \cdots (2)$$

$$\text{下限値: } P_L = \rho - u(\alpha) \sqrt{\frac{\rho(1-\rho)}{n}} \cdots (3)$$

(注) 危険率: α は通常 5% としている。

補足: 総検査数を上限値に乗算すると残障害数わかる

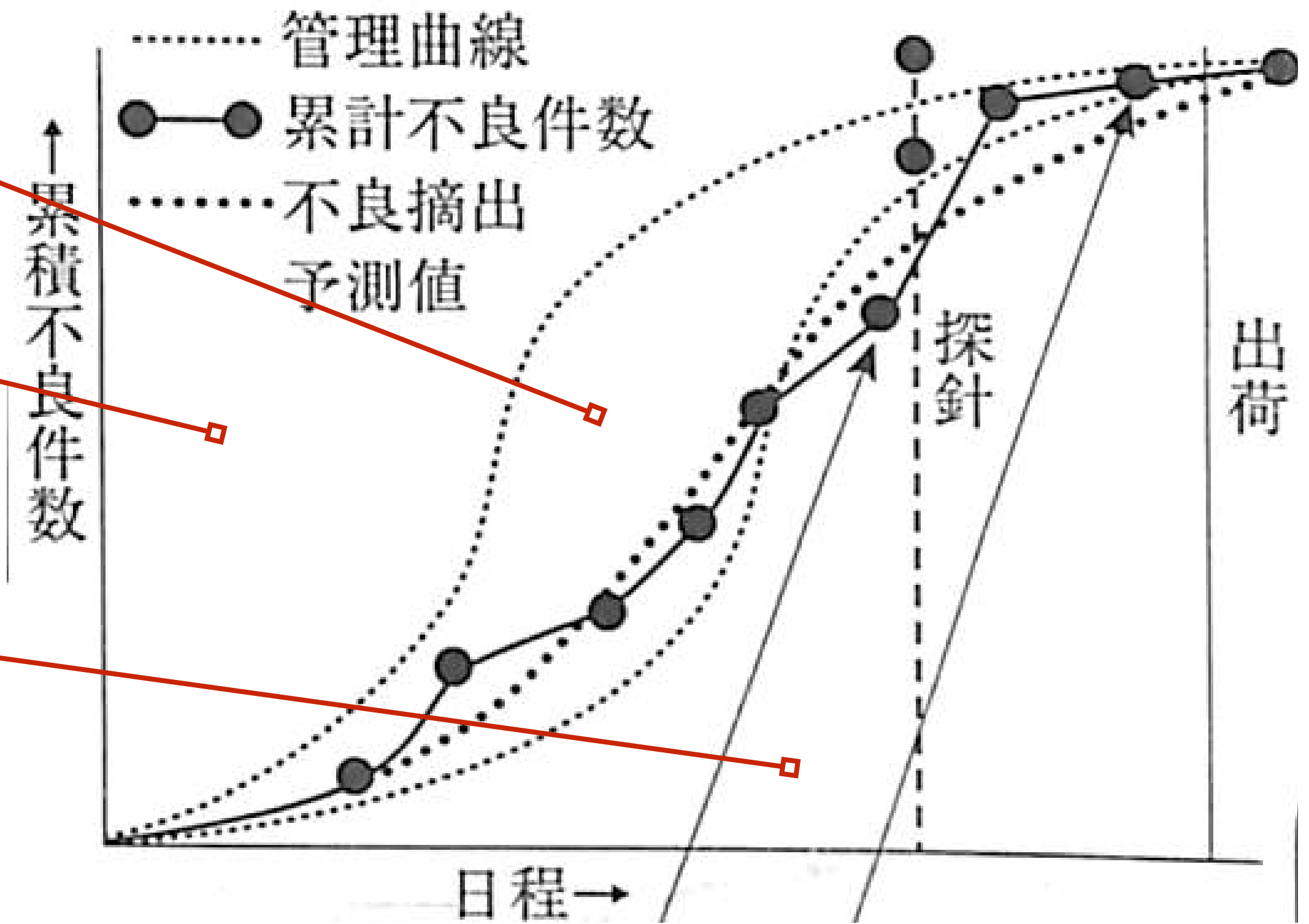
不良摘出累計実績と信頼度成長曲線の関係

管理曲線の上限・下限の間：
⇒問題なし

管理曲線の上限より上：
⇒ソフトウェアの品質が悪い
(or テスト能力が向上した)

管理曲線の下限より下：
⇒テスト能力が悪い
(or ソフトウェアの品質が良い)

※括弧内と判断は避けるべき
まずは、後述する不良分析を実施すべき



- データ収集
 - プログラム品質の評価
 - » 不良は十分に摘出されているか
 - テスト内容（品質）の評価
 - » 漏れなくテストされているか
 - 不良の作り込み
 - » どの工程で、どの程度の不良（件数、密度）が作り込まれているか
 - » どのような種類の不良（技術的、動機的原因）が作り込まれているか
 - » どのような部分（機能、モジュール）で不良が作り込まれているか
 - » どの部門（誰）が不良を作り込んだか
 - 不良の摘出
 - » 本来どのテスト工程や作業で摘出すべき不良か
 - » どのようなテスト技術で摘出すべき不良か

テスト工程の不良分析

- 実践的品質指標による評価
 - モジュール単位の規模あたりの不良件数
 - » 値の高すぎるモジュールは「でき」が悪く、いつまでも収束しない
⇒モジュールの質を見極めるコーディングレビューなどが必要
 - » 値の低すぎるモジュールは、テストが不十分の可能性がある
⇒テストカバレッジ率を調べ、テスト項目をレビューする必要
 - 重要不良の発生頻度推移

重要不良とは、実行不可になる不良やコンピュータリソース破壊、基本機能不良、機能劣化のこと。

 - » テスト工程の後半になっても、重要不良が散発
⇒不良1件ごとに、作り込み要因と摘出遅延要因を分析する
 - 不良収束率
 - » テスト工程の終盤になっても、不良収束率が高くない場合
⇒まだ品質が安定していないと考え、テスト項目の追加が必要

品質目標値の管理

テスト開始前に実施
不良摘出目標値を設定し、管理曲線を設定

- 不良摘出目標値は、過去の類似案件から設計・検査部門が設定

表 2.7 不良摘出目標値の設定モデル例

$$Y = y_1 + y_2 + \dots + y_n + C \cdot X^p$$

Y: 作り込み不良件数
 $y_1, y_2, \dots, y_n$: 品質要因の係数
 C: 類似製品の平均不良件数(件/KS)
 X: 開発規模(KS)
 p: 定数

品質要因例
 ● 処理内容
 ● 処理形態
 ● 実装形態
 ● 設計上級
 ● 開発チーム能力

管理曲線は、信頼度成長曲線から求める

1. 不良摘出目標値の設定
 2. 管理曲線の設定
 3. 実績の監視
 4. 目標値管理
 5. 品質予測と評価
 6. 品質改善
 7. 移転管理

品質目標値管理
 実績が管理曲線から外れた場合、原因究明と対策

収束不良管理
 収束中による品質目標値管理

※ 実績不良件数、予測不良件数

管理図に基づく品質評価

- 不良摘出累計実績と信頼度成長曲線の関係

管理曲線の上限・下限の間：
⇒問題なし

管理曲線の上限より上：
⇒ソフトウェアの品質が悪い
(or テスト能力が向上した)

管理曲線の下限より下：
⇒テスト能力が悪い
(or ソフトウェアの品質が良い)

※括弧内と判断は避けるべき
まずは、後述する不良分析を実施すべき

テスト工程の不良分析

- 実践的品質指標による評価
 - モジュール単位の規模あたりの不良件数
 - 値の高すぎるモジュールは「でき」が悪く、いつまでも収束しない
⇒モジュールの質を見極めるコーディングレビューなどが必要
 - 値の低すぎるモジュールは、テストが不十分の可能性があり
⇒テストカバレッジ率を調べ、テスト項目をレビューする必要
 - 重要不良の発生頻度推移
重要不良とは、実行不可になる不良やコンピュータリソース破壊、基本機能不良、機能劣化のこと。
 - テスト工程の後半になっても、重要不良が散発
⇒不良1件ごとに、作り込み要因と摘出遅延要因を分析する
 - 不良収束率
 - テスト工程の終盤になっても、不良収束率が高くない場合
⇒まだ品質が安定していないと考え、テスト項目の追加が必要

- 本当の品質管理表

- 今日のスライドで紹介したものは簡素化したもの
- 本来は、図のような管理表で一元管理すべき

