

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

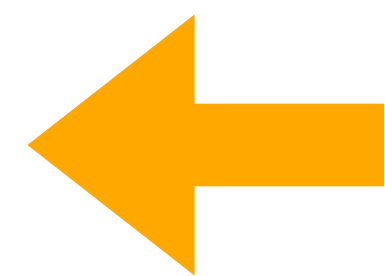


奈良隆正

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

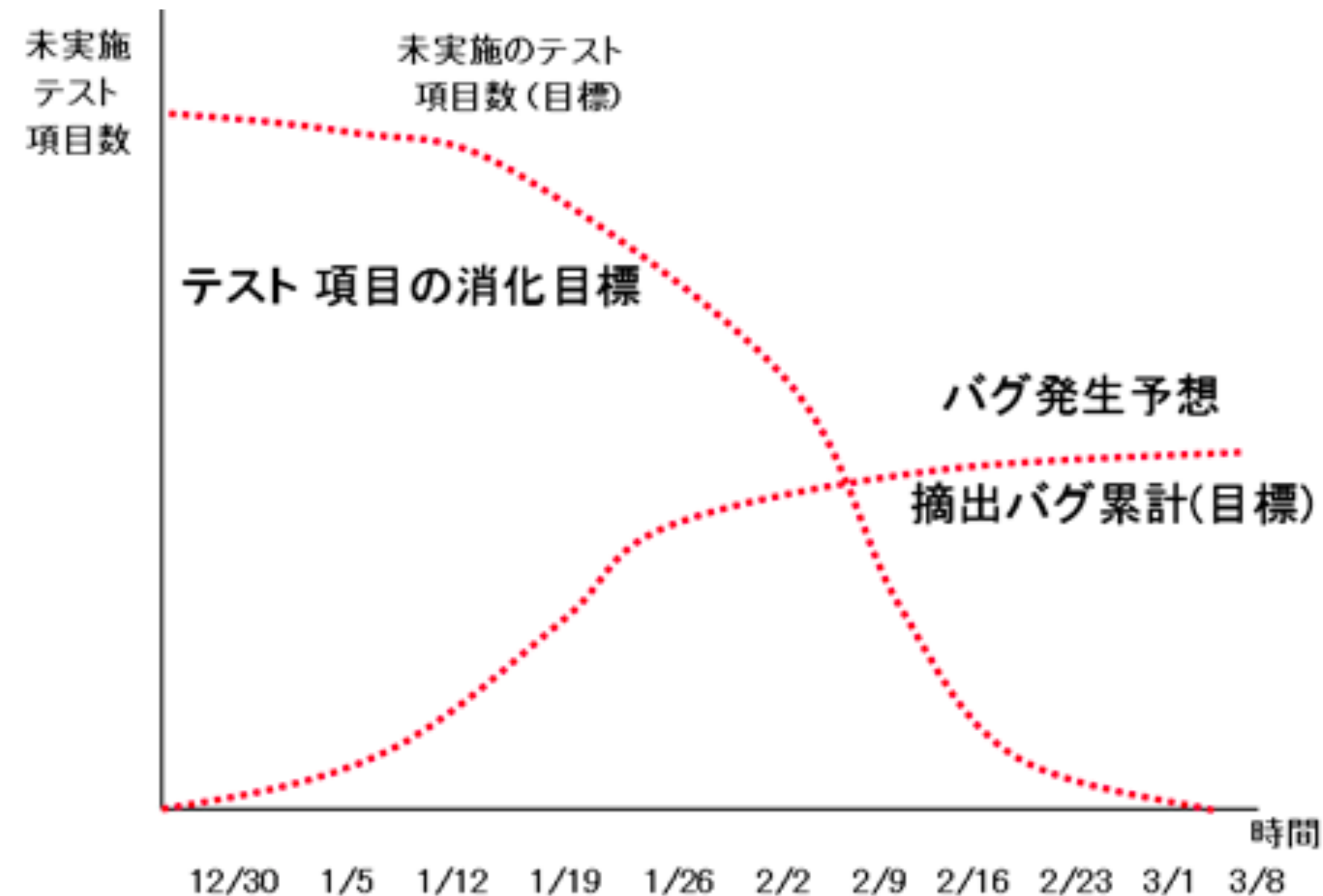


番外編

バグ曲線

バグ曲線

- 時間経過やテスト項目の消化に伴うバグの発生件数をグラフにまとめたもの
 - バグ発生曲線やバグ収束曲線とも言う（左肩下がり）
 - 似たものにテスト消化曲線がある（右肩下がり）

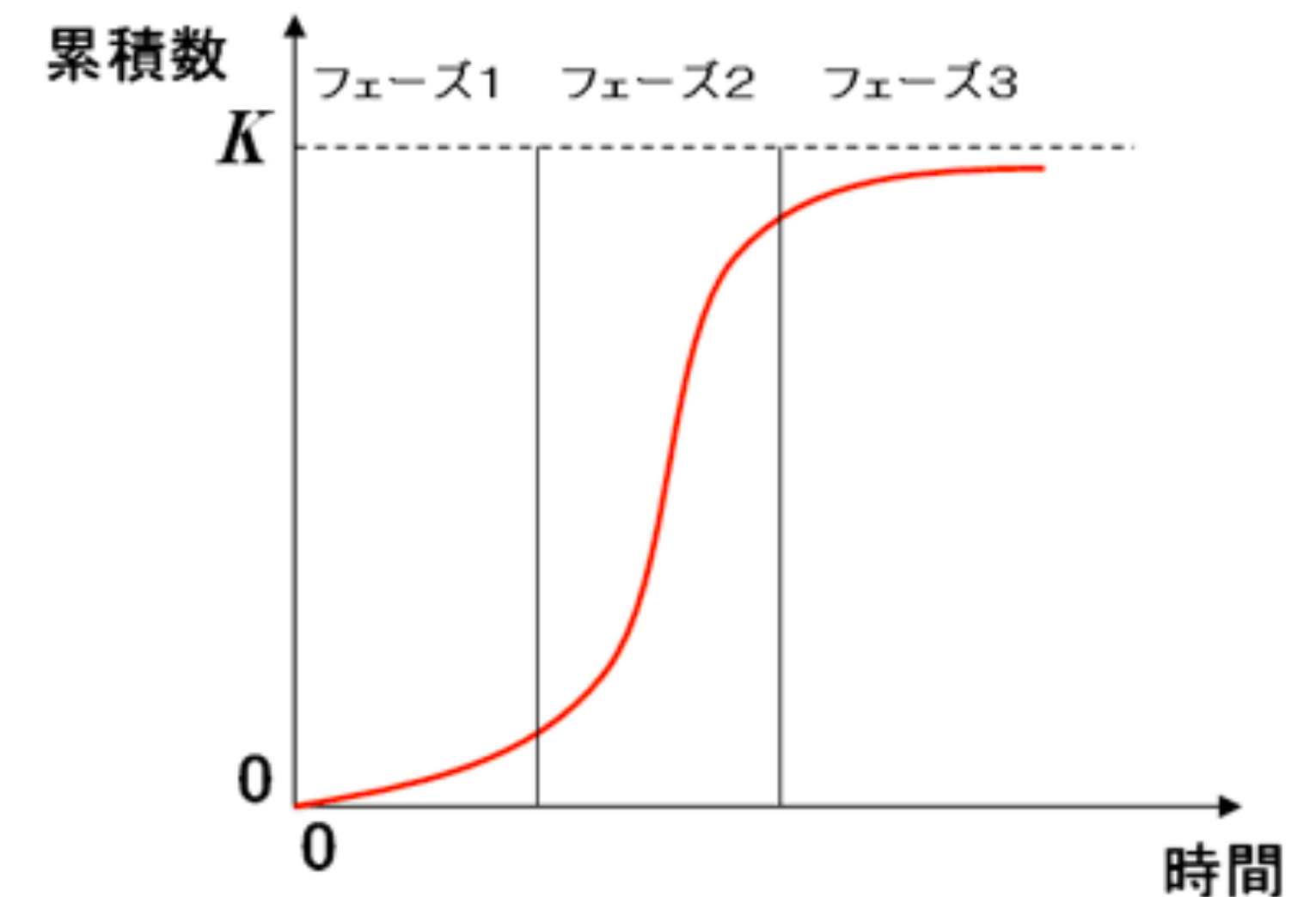


どう描く（予測する）の？

• 成長曲線を利用

- 代表格：Gompertz Curve（ゴンペルツ曲線）

- » フェイズ1：胎動期（準備段階）
 - * 正常系の不具合が見つかりやすい
- » フェイズ2：成長期（急激に成長する段階）
 - * 異常条件や境界条件，限界条件に関する不具合が見つかりやすい
- » フェイズ3：停滞期（成長が止まり、漸近線に近づく段階）
 - * 「特殊条件」「複合条件」（例えば、タイミング）や性能関連の不具合が見つかりやすい



※実際の開発現場では、「テスト項目の総数」と「テストに割り当てた日数」から適当に曲線を引いている場合が多い

$$E = K \cdot a^{b^t}$$

E : 累積バグ数

K : $t \rightarrow \infty$ での E の収束値

a, b : 1より大きいパラメータ

} 測定値から推定

どう使う？



進捗管理・品質管理



出荷判断

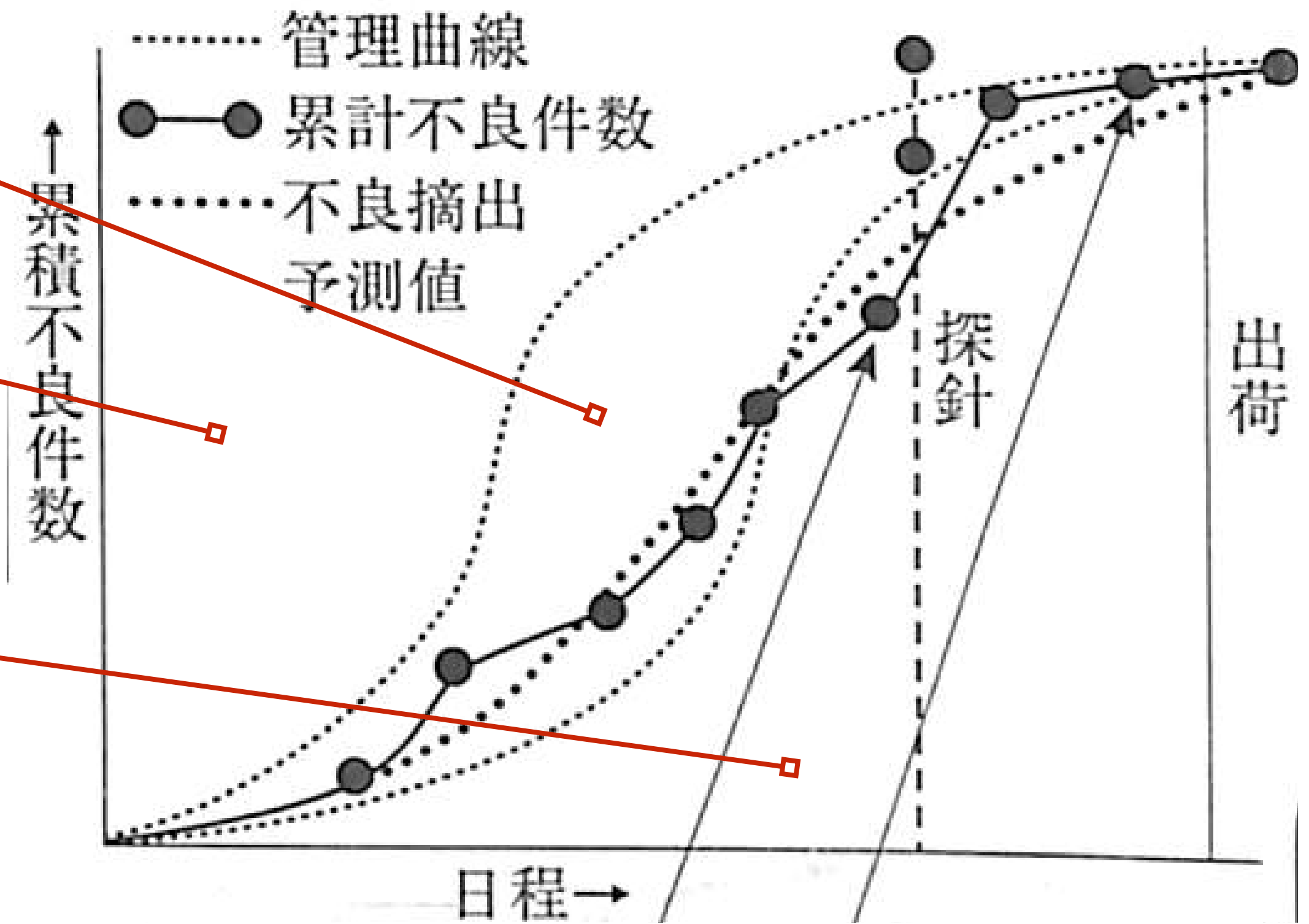
【復習】 管理図に基づく品質評価

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)

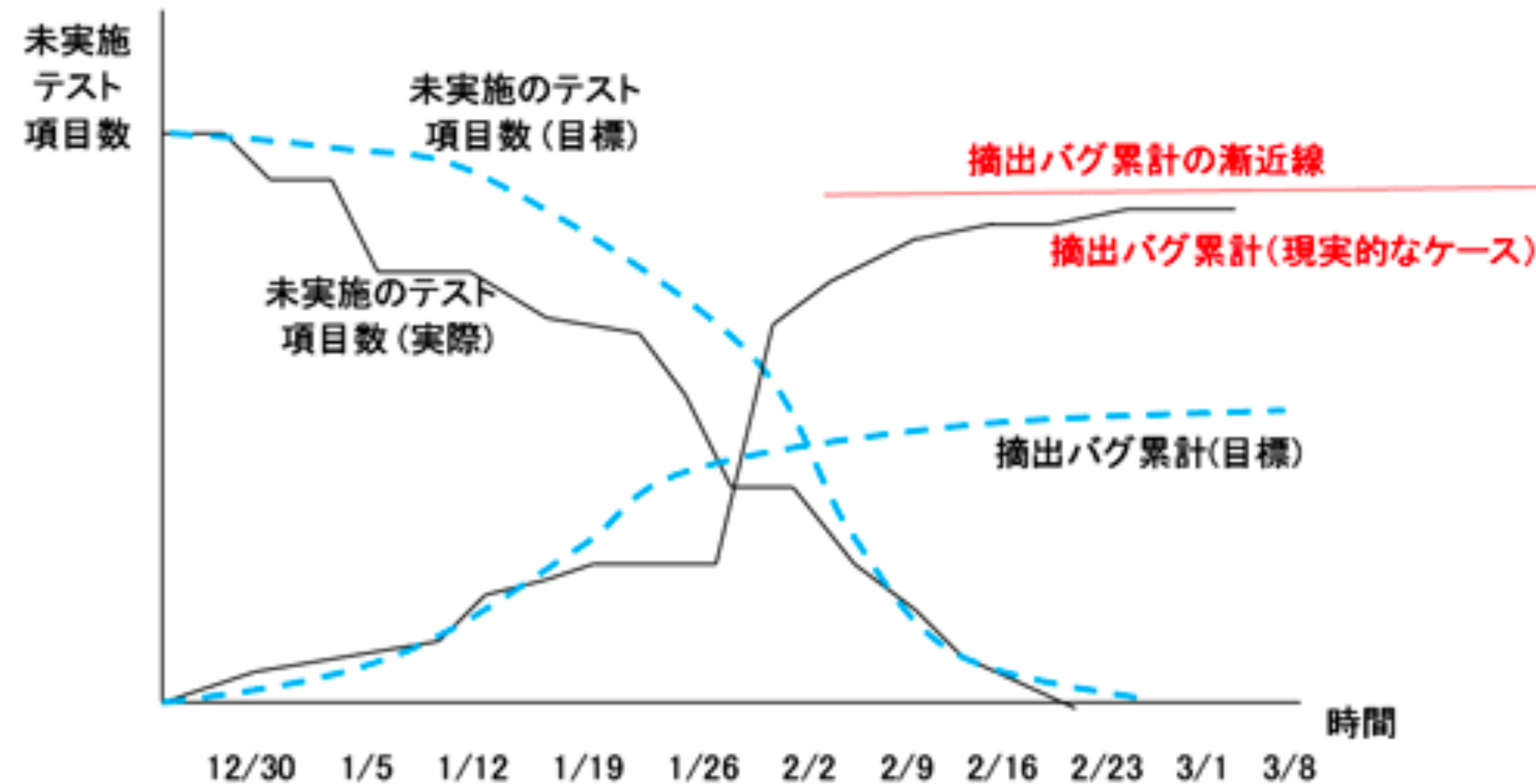


※テストに掛ける人数によって大きく変わる

出荷基準

- 全機能をテストした
 - » ブラックボックス／ホワイトボックス・テストで同値分割を実施した
- 境界条件をテストした
 - » ブラックボックス／ホワイトボックス・テストの境界値分析を実施した
- 未実行コードがない
 - » ホワイトボックス・テストのC0パス網羅を満足した
- エラー・ゲッシング（Error Guessing）を実施した
 - » バグを想定し、それを摘出するためのテスト項目を設計・実施した
- 長時間耐久テスト、過負荷テストを実施した
 - » 48時間の連続運転や、過負荷状態での稼働テストを実施した
- バグの発生が頭打ちになった
 - » 「バグ摘出曲線」がフラットになった
- 未修正のバグが残っていない
 - » 摘出したバグは全件修正した

バグの発生が頭打ち



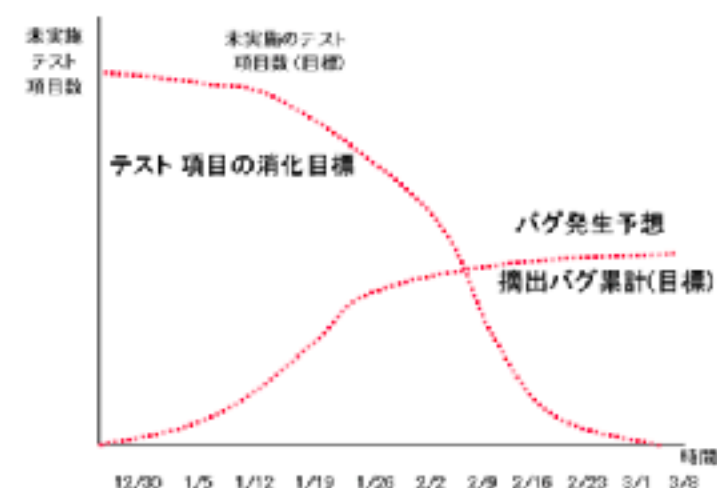
- 条件

- » テストが実施していること
- » テストの品質が低くないこと
- » テスト項目が抜けていないこと
- » 重大な不具合が発生していないこと (クラッシュなど)

バグ曲線

3

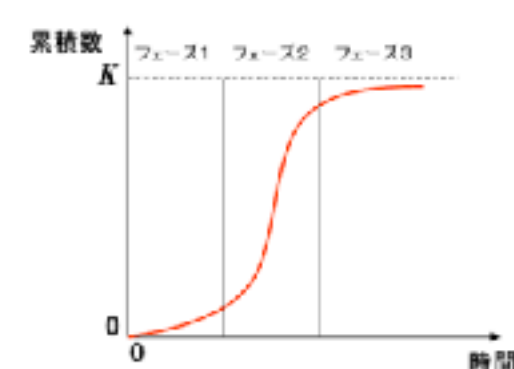
- 時間経過やテスト項目の消化に伴うバグの発生件数をグラフにまとめたもの
 - バグ発生曲線やバグ収束曲線とも言う（左肩下がり）
 - 似たものにテスト消化曲線がある（右肩下がり）



どう描く（予測する）の？

4

- 成長曲線を利用
 - 代表格：Gompertz Curve（ゴンペルツ曲線）
 - フェーズ1：胎動期（準備段階）
 - 正常系の不具合が見つかりやすい
 - フェーズ2：成長期（急激に成長する段階）
 - 異常条件や境界条件、限界条件に関する不具合が見つかりやすい
 - フェーズ3：停滞期（成長が止まり、漸近線に近づく段階）
 - 「特殊条件」「複合条件」（例えば、タイミング）や性能関連の不具合が見つかりやすい



※実際の開発現場では、「テスト項目の総数」と「テストに割り当てた日数」から適当に曲線を引いていることが多い

$$E = K \cdot a^{b \cdot t}$$

E : 累積バグ数
 K : $t \rightarrow \infty$ での E の収束値
 a, b : 1より大きいパラメータ

測定値から推定

【復習】管理図に基づく品質評価

6

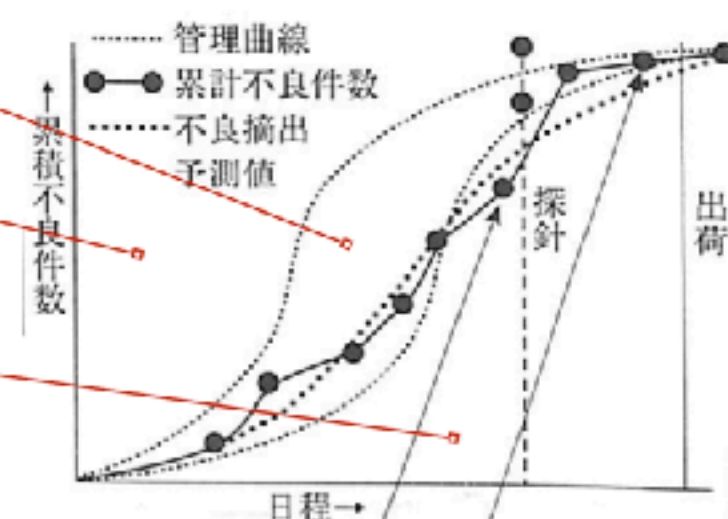
- 不良摘出累計実績と信頼度成長曲線の関係

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)

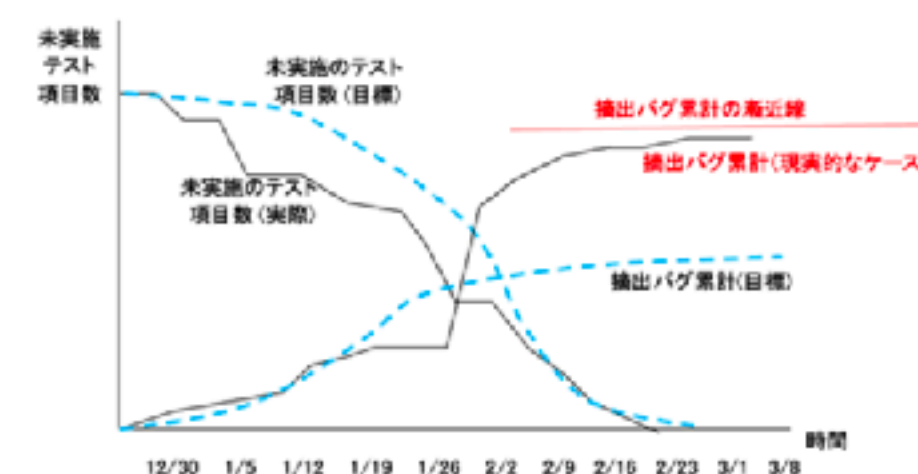
管理曲線の下限より下：
⇒テスト能力が悪い
(or ソフトウェアの品質が良い)

※テストに掛ける人数によって大きく変わる



バグの発生が頭打ち

8



- 条件
 - テストが実施していること
 - テストの品質が低いこと
 - テスト項目が抜けていないこと
 - 重大な不具合が発生していないこと（クラッシュなど）