

# TOSHIBA

Leading Innovation >>>

---

SPI Japan 2008

## 統計的手法による 不具合予測モデル構築

— 定量的プロジェクト管理の定着を目指して —

2008年11月26日

東芝デジタルメディアエンジニアリング株式会社

ソフトウェアセンター

竹内 朝一

# 本日の内容

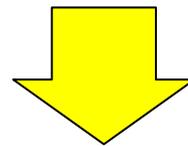
---

- SPI活動の取り組み
- 定量的管理の悩みと対策
- 不具合予測モデルの構築と活用事例
- “新”不具合予測モデルへ
- 成果と課題

# SPI活動の取り組み

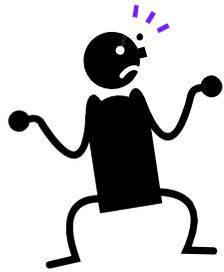
---

- 組込系ソフトウェア開発(主に携帯電話)を担当
- QCD(特に品質と納期)改善のために、SW-CMMを手本としたプロセス改善活動を推進
- 安定したプロジェクト管理のために、開発データを活用したプロセスの構築と展開、定着
- さらに...統計的手法を用いて、品質を制御する取り組みを継続中



**不具合予測モデル構築**

# 定量的管理の悩みと対策



- データの計測、集計が大変。時間がない
- データの精度が悪くて使えない
- データの見方、分析方法がわからない



## 【負荷軽減、精度向上】

- 計測、集計ツールによる自動化

## 【データの見える化】

- 予実差グラフによる分析支援
- Webモニタによる見える化

## 【データの活用】

- 組織係数による見積支援
- 統計手法による不具合予測モデル

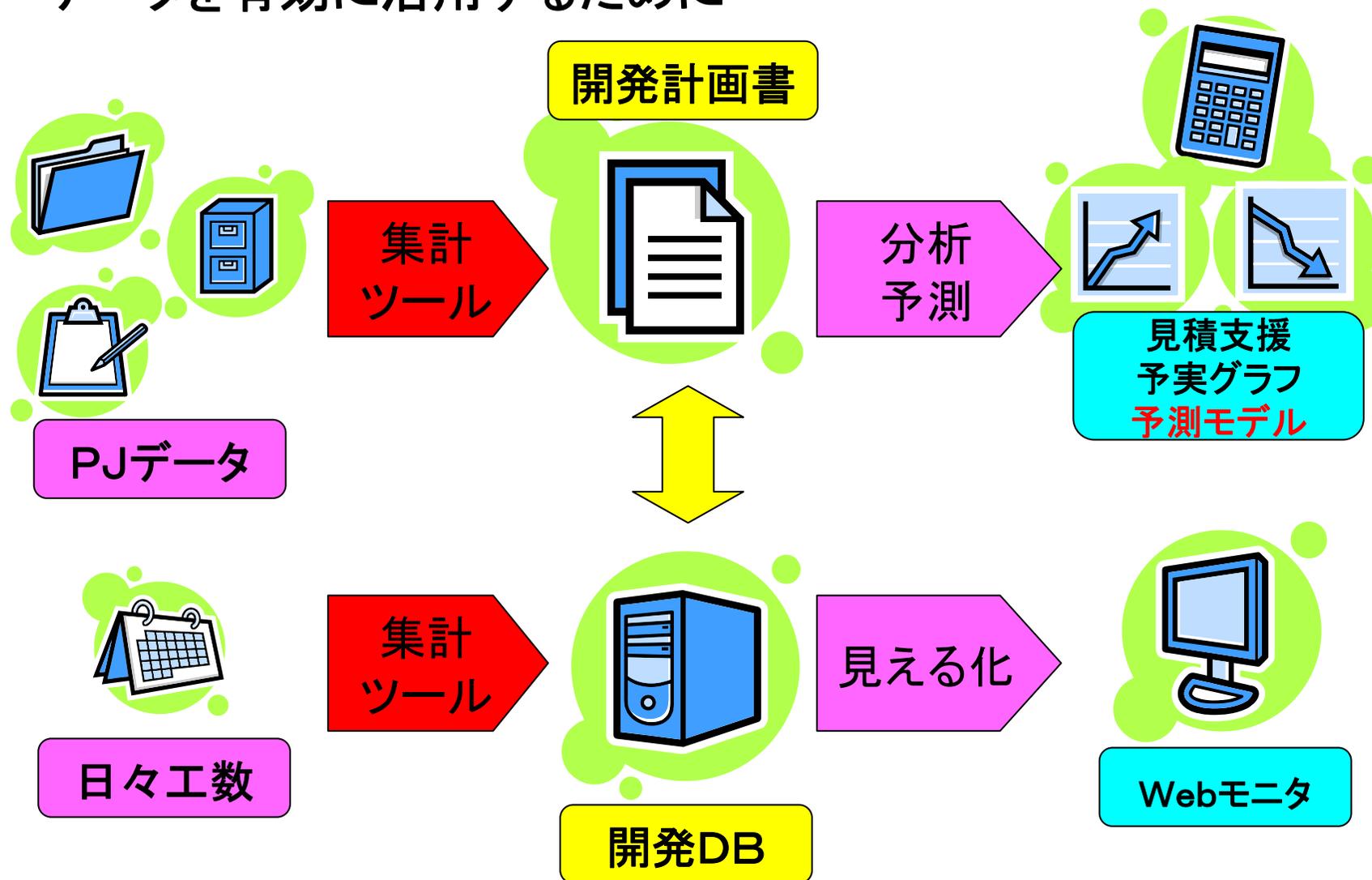
## 【モチベーションの向上】

- 問題が「見えたら」すぐに対策、改善



# プロジェクト管理環境

データを有効に活用するために…



# 不具合予測モデルの目的

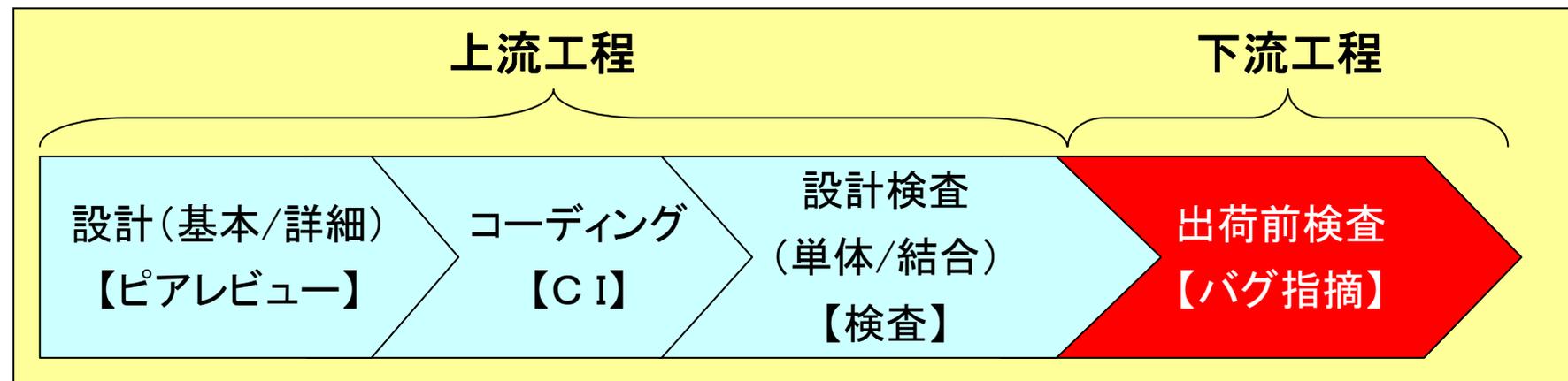
## 【品質目標：3つの指標】

「下流バグ密度(件/KS)」 [下流バグ件数(件) ÷ 開発規模(KStep)]

「上流バグ検出率(%)」 [上流バグ件数(件) ÷ 全バグ件数(件)]

「手戻り工数率(%)」 [手戻り工数(H) ÷ 全工数(H)]

- 下流工程で検出されるバグは、**QCDに影響が大きい**
- 出荷前検査でのバグ指摘件数が減少するよう**品質を制御**する
- 「**下流バグ密度**」に着目し、「不具合予測モデル」構築



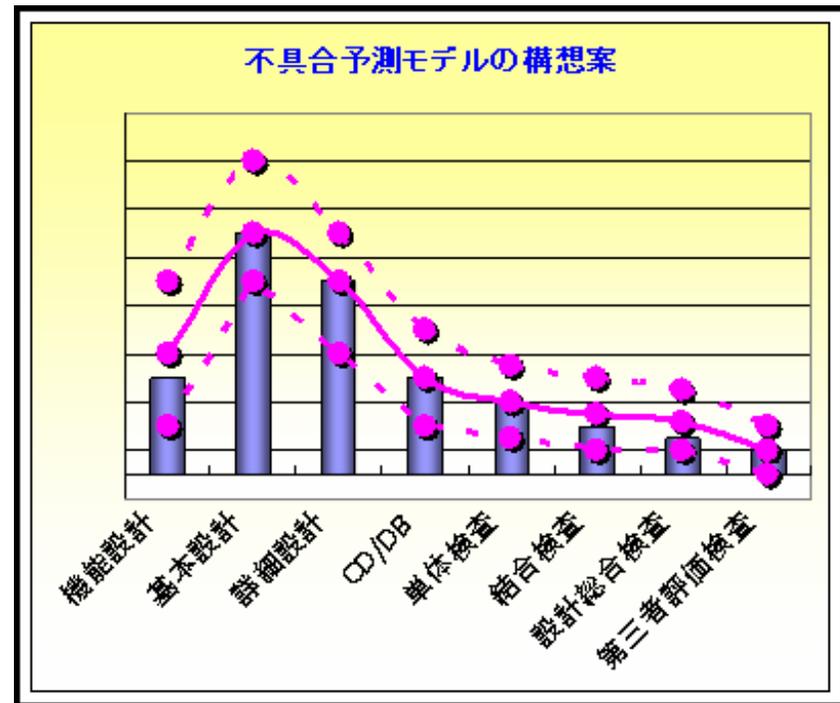
# 不具合予測モデルとは

## 【目的】

- ライフサイクル全体で検出するバグ(生涯バグ)件数を予測
- バグ検出の上流化、件数削減の対策を行う
- 品質を制御し、下流バグ件数を削減する

## 【構築ポイント】

- 蓄積したデータを活用し、統計的にアプローチ
- 進捗にあわせて品質を制御できるように、工程ごとに予測
- 予測値に上下限の幅を設け許容範囲からはずれたとき、問題ありと認識させる



# 不具合予測モデル構築 ①

## 【組織のバグ発生件数の能力ベースラインを求める】

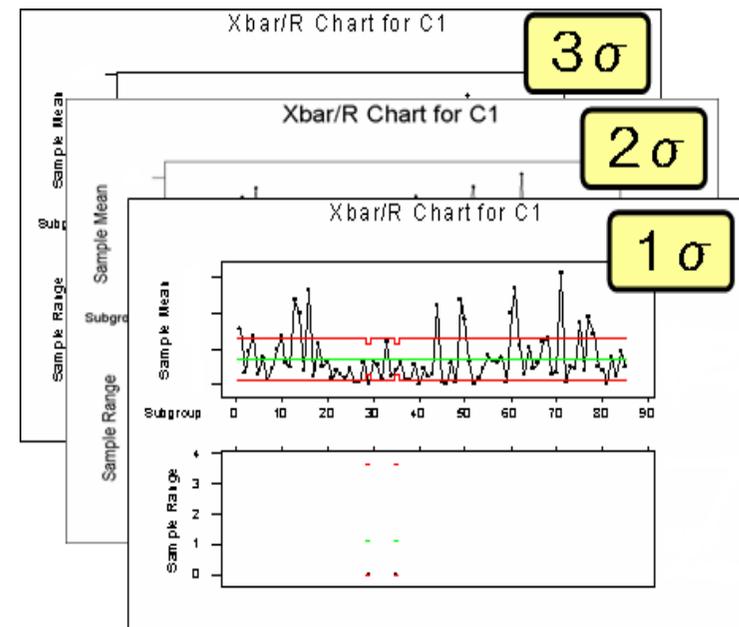
- 組織の**標準的な品質**（下流バグ密度）のPJを選出する
- 管理図を用いて、 $3\sigma$ 、 $2\sigma$ 、 $1\sigma$ と管理領域を変えながら、管理外のPJを**分析**し、ハズレ値として**除外**する

## 【分析結果】

- $1\sigma$  圏外PJは、**特殊要因**あり
- $1\sigma$  圏内PJは、下流バグ密度の平均値が品質目標の**目標値**を**クリア**している（=成功PJ）

## 【結論】

- $1\sigma$  圏内PJを当部門の**成功PJ**と位置づけ、能力値を算出



# 不具合予測モデル構築 ②

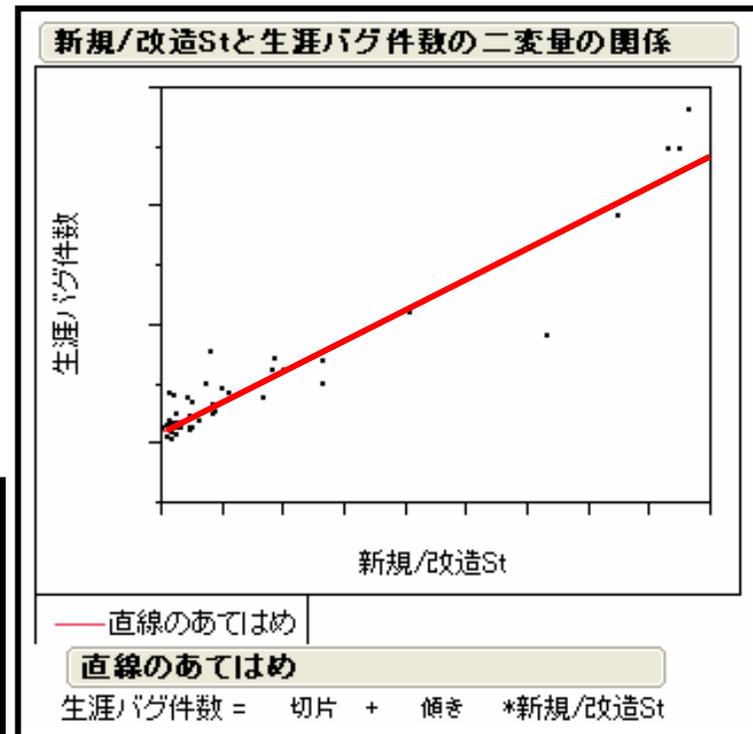
## 【生涯バグ件数を予測する単回帰式を求める】

- 生涯バグ件数(目的変数)の説明変数を求め**相関の強さ**を確認
- 説明変数から生涯バグ件数を算出できる**単回帰式**を求める

## 【分析結果】

- 「**生涯バグ件数**」と「**開発規模**(ステップ数)」に強い相関がある
- 単回帰式により生涯バグ件数が**予測可能**である

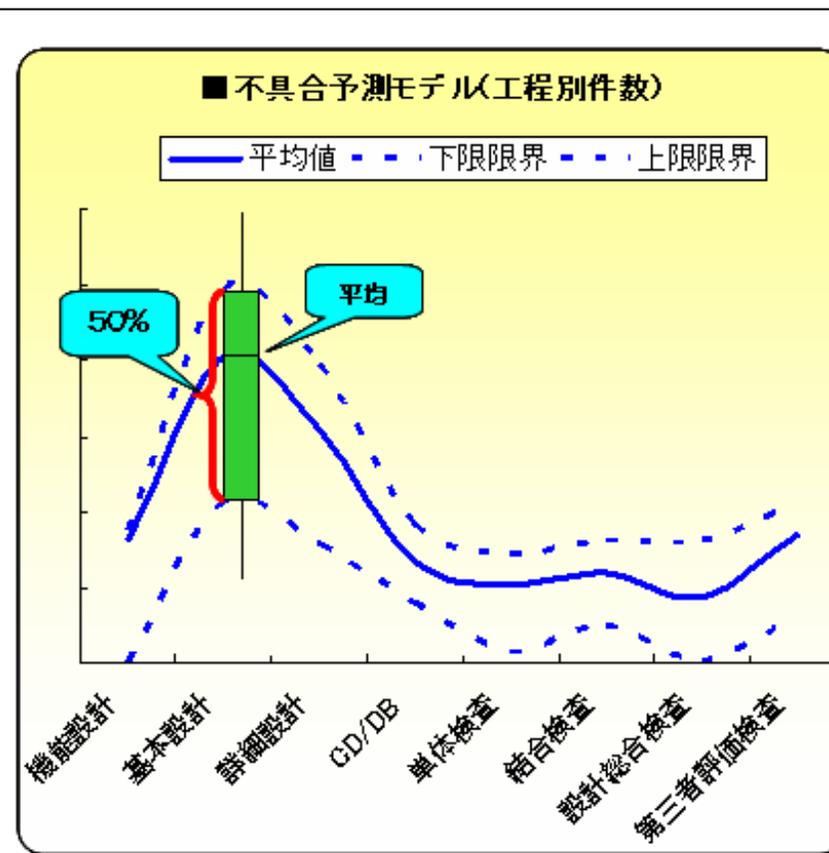
|                           |          |
|---------------------------|----------|
| 生涯バグ件数 = 切片 + 傾き *新規/改造St |          |
| あてはめの要約                   |          |
| R2乗                       | 0.909626 |
| 自由度調整R2乗                  | 0.907367 |



# 不具合予測モデル構築 ③

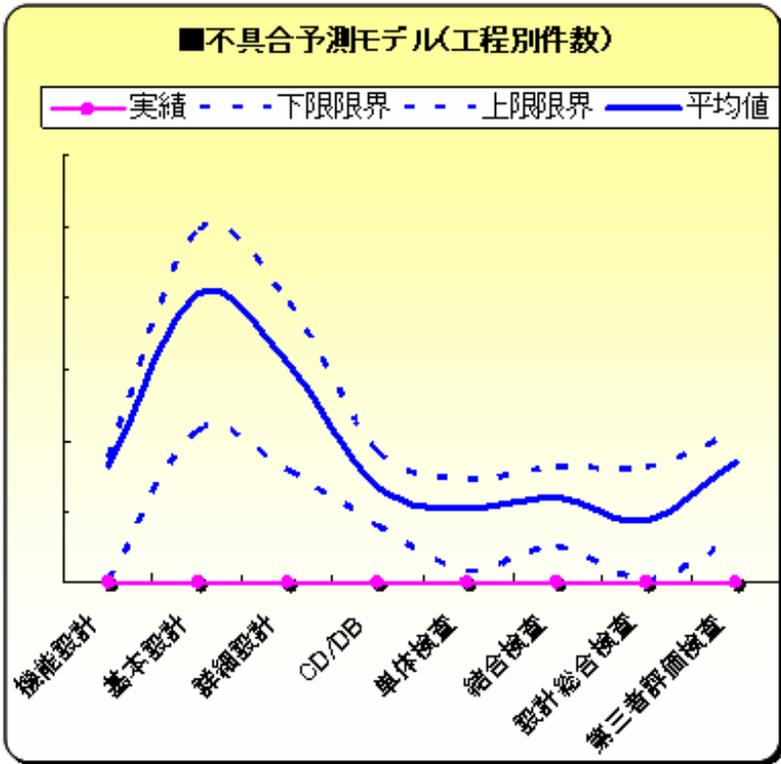
## 【構築ポイントをクリアする】

- 進捗にあわせて品質を制御できるように、**工程ごとに予測**
  - 各工程の**平均バグ発見率**で、「生涯バグ件数」を割り振る
- 予測値に**上下限の幅**を設け許容範囲からはずれたとき、問題ありと認識させる
  - 各工程のバグ検出率を四分位数で分割し
    - 上限を75%の観測点**
    - 下限を25%の観測点**
    - を許容幅とする



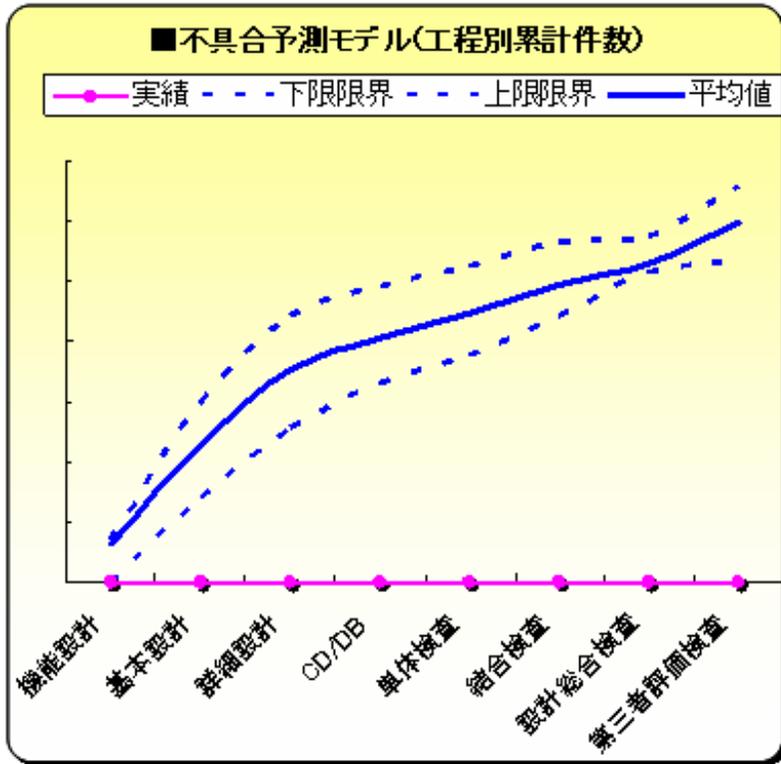
# 不具合予測モデル構築 ④

## 工程バグ件数予測モデル



各工程に発見すべきバグ件数

## 生涯バグ件数予測モデル

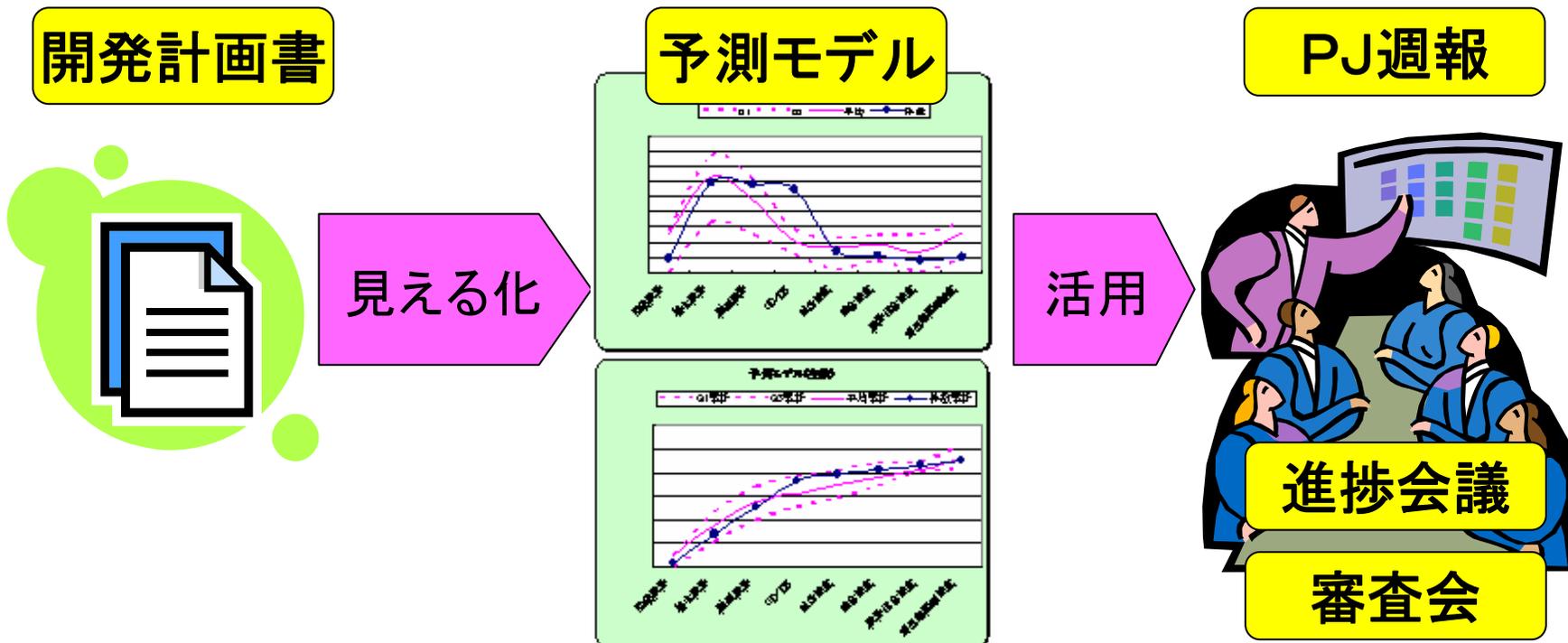


各工程までに発見すべきバグ件数の累計

# 不具合予測モデル構築 ⑤

## 【予測モデルの活用定着のために】

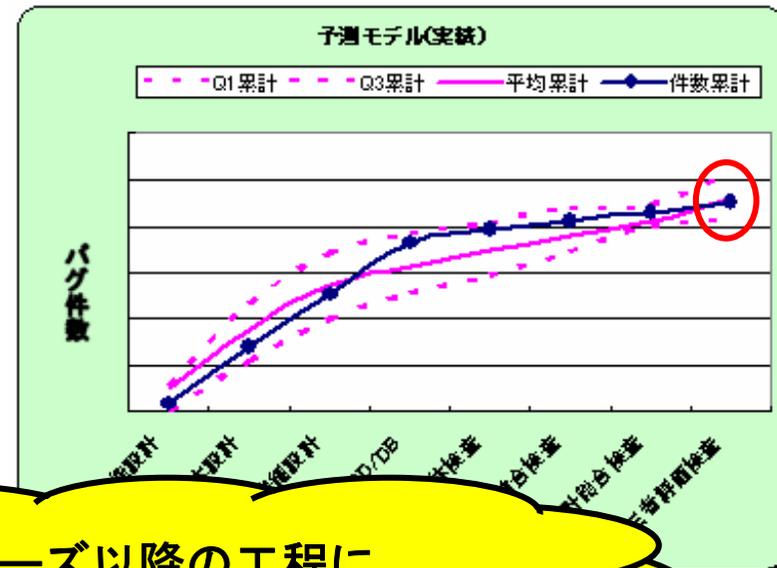
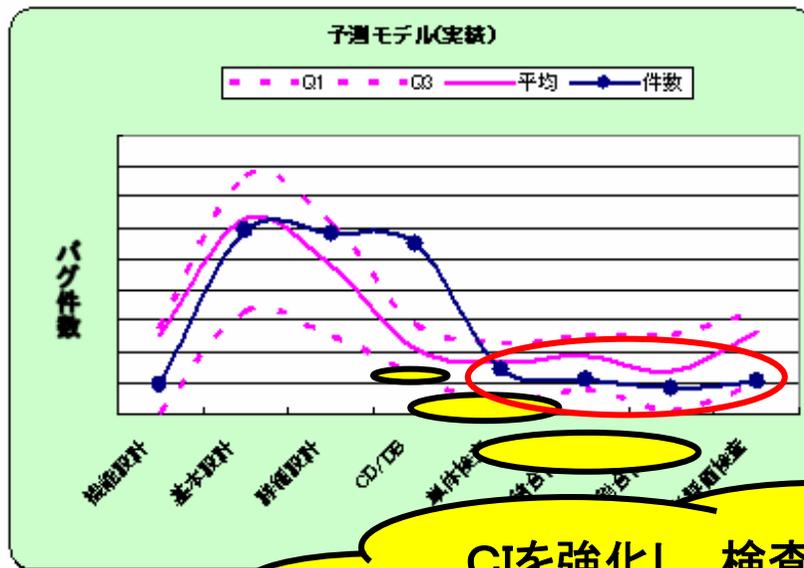
- 予測値、計画値、実績値がグラフに反映される→「見える化」
- 予測モデルを活用した品質確認、対策検討 →「活用」



# 不具合予測モデル活用例

## 【PJ計画から推進状況、振り返りまで】

- 予測モデルで、PJの品質計画（レビュー、検査）
- PJ推進中の品質確認と後工程の計画見直し
- PJ終了後の振り返り、改善分析



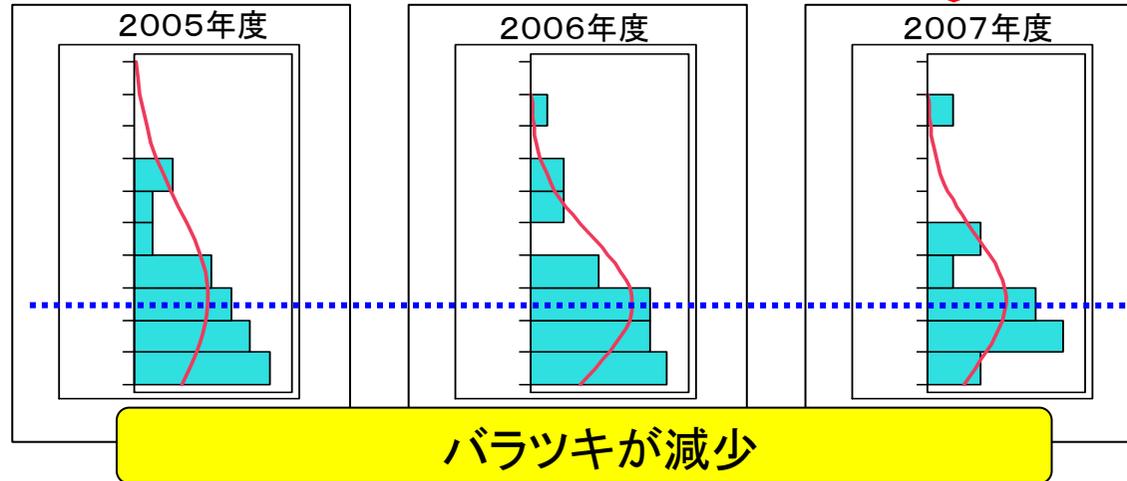
CIを強化し、検査フェーズ以降の工程に  
不具合を持ち越さない計画でPJを推進

# 不具合予測モデルの効果!?

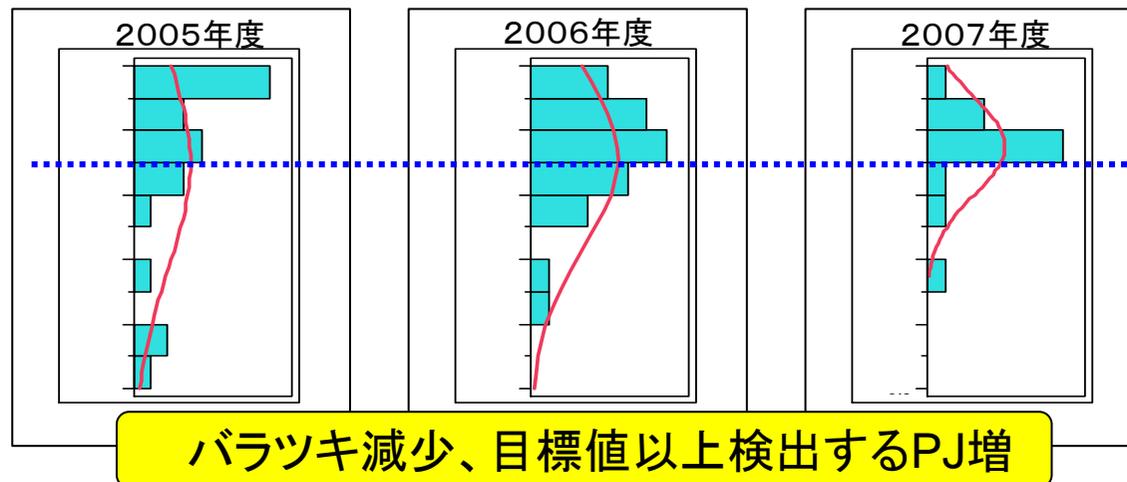
予測モデル導入

( ..... は、部門目標値)

下流バグ密度



上流バグ検出率



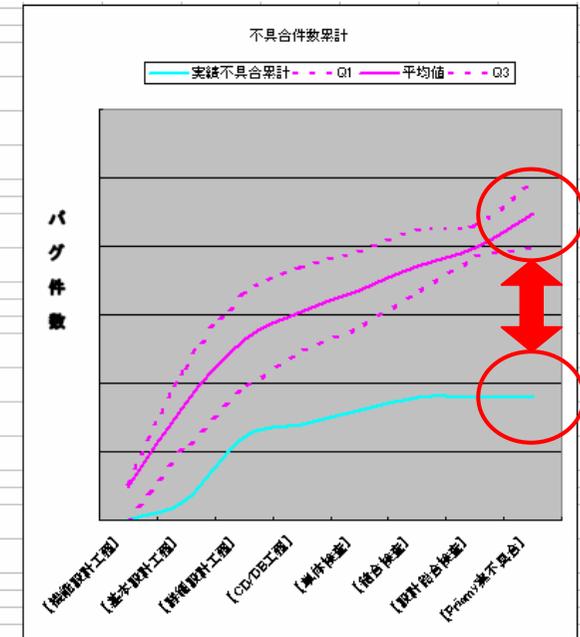
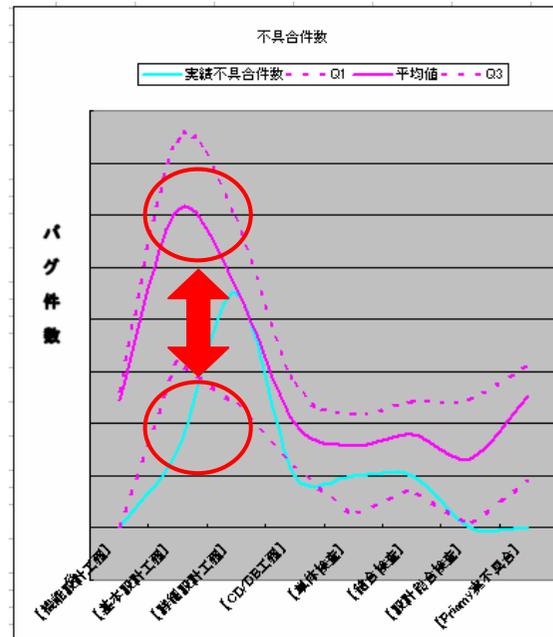
# 不具合予測モデルの問題点

## 【組織の全PJのデータを対象にしたので...】

- 小規模または、新規大規模開発など、予測モデルが**感覚的にあわず**、使いづらい場合がある
- 過去三年分のデータを元にしており、トレンドの開発スタイルと**ずれてきている**(ようだ)

## 【問題例】

- 小規模開発なのに、予測件数が多すぎる
- 差分開発のため基本設計量が少ないのに...



# “新”不具合予測モデル構築の取り組み

## 【開発規模毎にモデルを使い分ける】

- 開発規模毎に異なる相関を持つグループ分けを行う。  
(JMP7※のパーティション機能を用いた)
- グループ毎に生涯バグ件数を予測する単回帰／重回帰式を求め「不具合予測モデル」を構築する

| グループ | 開発規模(Step) | 相関の見られる説明変数         |
|------|------------|---------------------|
| A    | < 853      | コード流用率              |
| B    | < 1373     | PR工数、総工数            |
| C    | < 2447     | PR工数、総工数            |
| D    | < 2851     | 総工数、ソースファイル流用率      |
| E    | < 7108     | PR工数、ソースファイル流用率     |
| F    | >=7108     | PR工数、総工数、ソースファイル流用率 |

※JMP7 : SAS Institute Japan株式会社 <http://www.jmp.com/japan/>

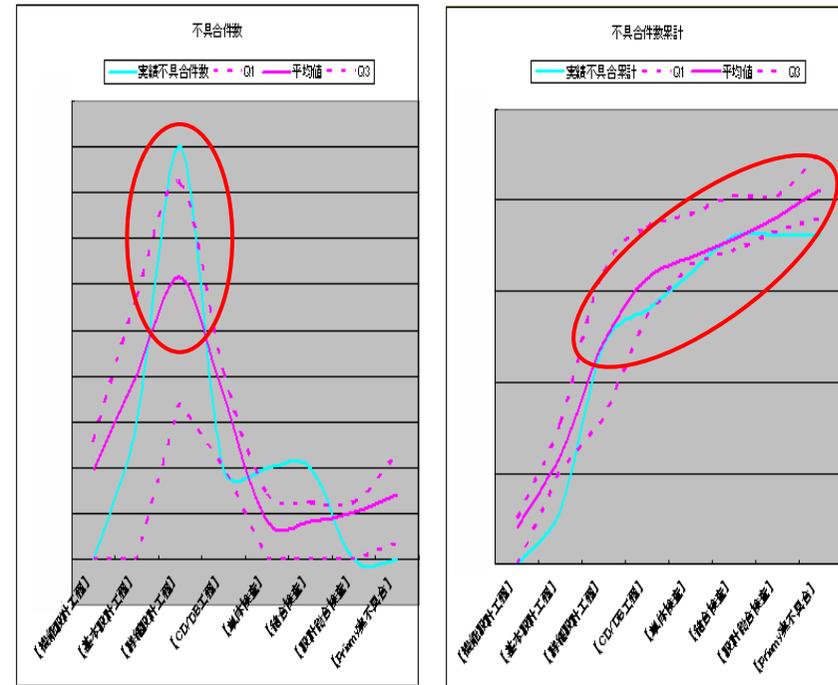
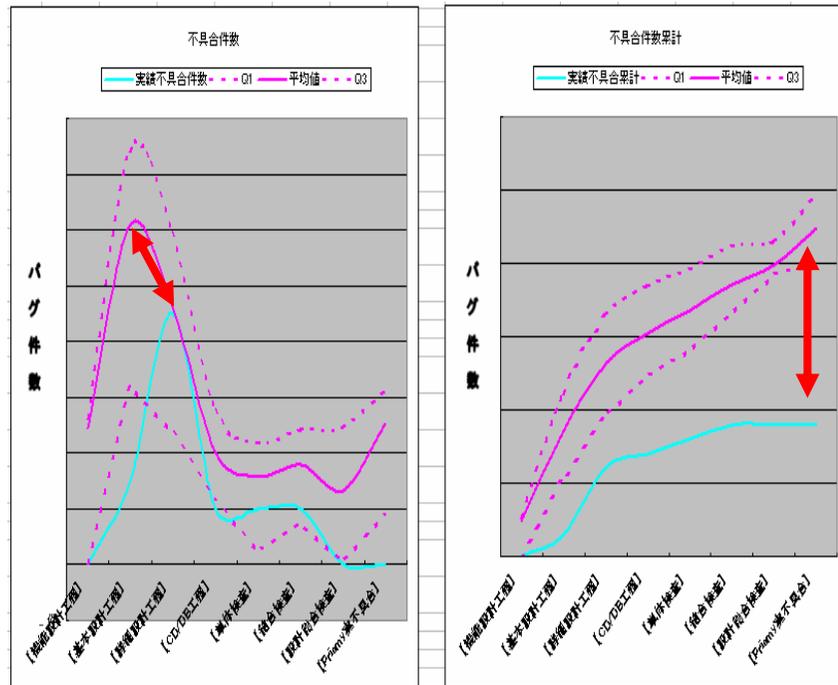
# “新”不具合予測モデルのあてはめ例

## 【Aグループ】

- 開発規模 : 853 Step未済
- 説明変数 : コード流用率

旧モデル

新モデル



# 成果と課題

## 【成果】

- 工程毎に実績を確認し、下流品質を予測することで、**事前対策**がとれるようになった
- 統計的手法を用いて、トレンドにあわせたモデルを容易に構築する**ノウハウを蓄積**することができた
- 不具合予測モデルなど、計測したデータの活用方法を提供することにより、定量的プロジェクト管理が**定着**した

## 【課題】

- 新不具合予測モデルの説明変数の多くは**工数**であり、予測モデルとして適切とはいえない。新モデルのさらなる改善が必要
- 高精度な予測モデルとするため、**多重相関**を考慮して発展させていきたい

**TOSHIBA**

**Leading Innovation >>>**