

ETWest2007 コミュニティセッション、セッションNo. C-1、2007年6月6日(水)

# プロジェクトモニタリングの事例と 蓄積データを用いた見積り／傾向分析

森崎 修司

EASEプロジェクト

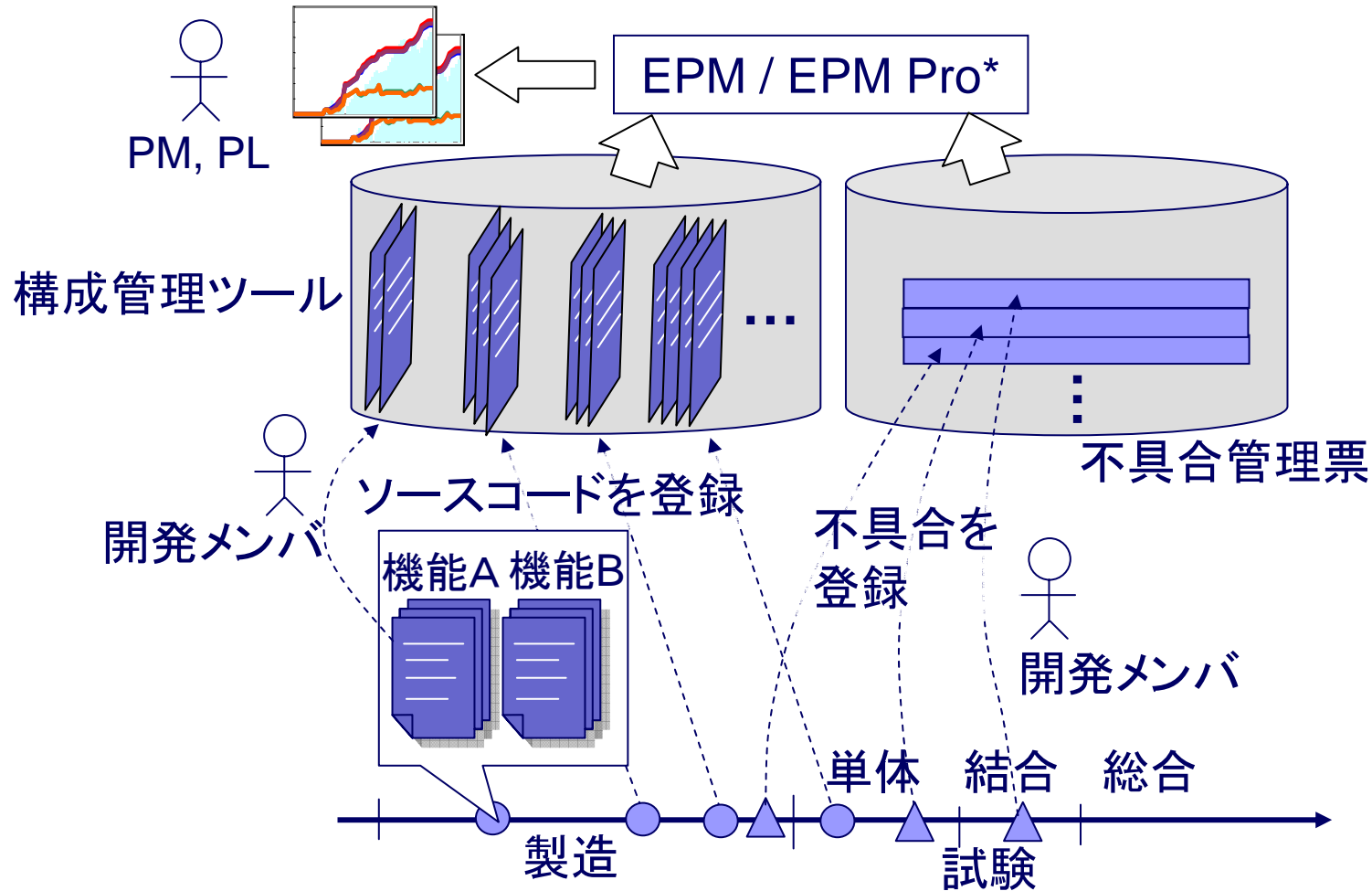
国立大学法人奈良先端科学技術大学院大学

# 概要

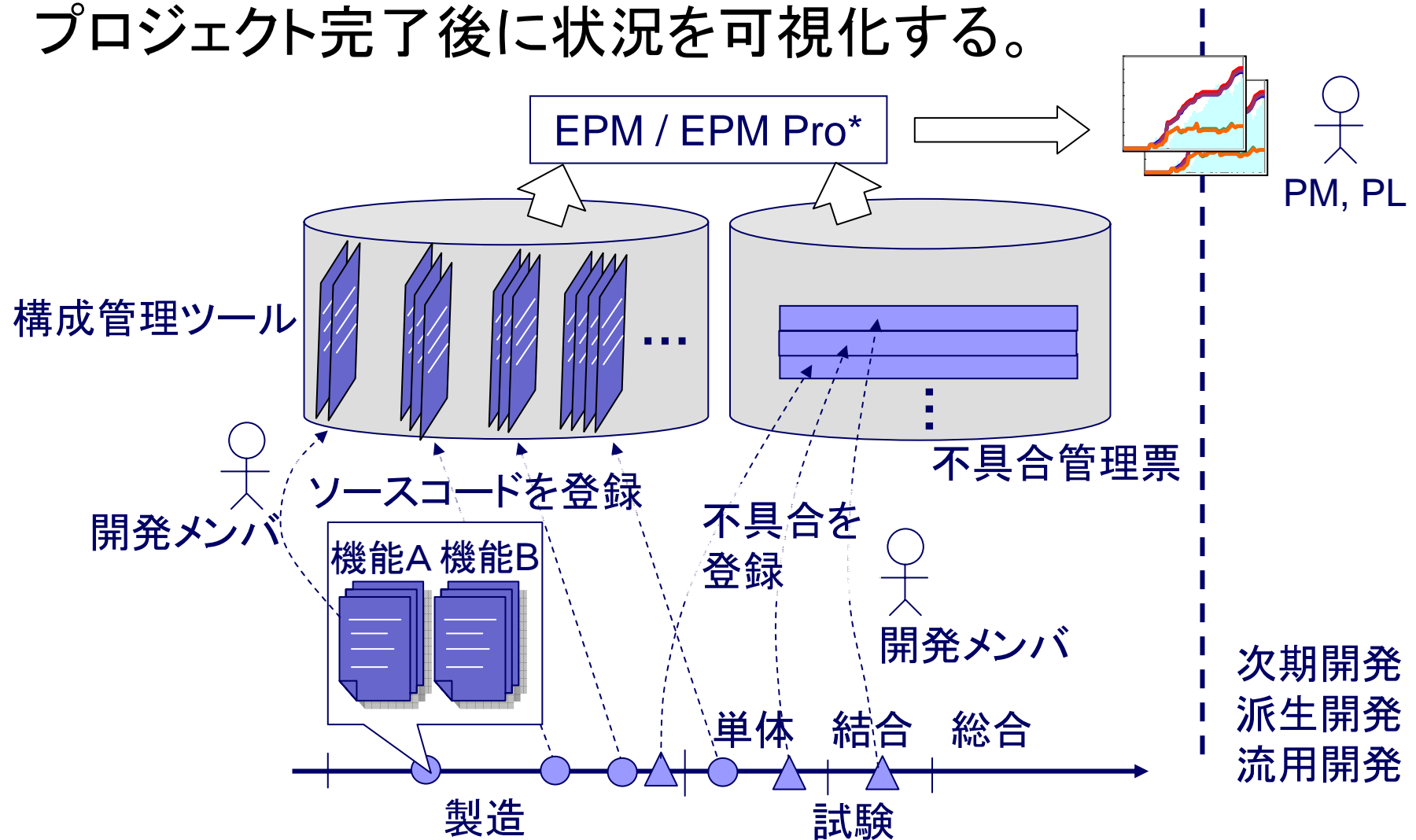
- プロジェクトモニタリング（インプロセス分析）
  - ソースコード構成管理システム
  - 不具合管理票（バグ票）
  - 適用事例（進行中、完了後）
- リポジトリを対象とした見積り/傾向分析
  - 過去のプロジェクトとの類似度による見積り（Magi: 協調フィルタリング見積り）
  - 相関ルール分析による傾向分析（NEEDLE）

# プロジェクトモニタリングの概要 - 進行中 -

プロジェクト進行中に随時状況を可視化する。



プロジェクト完了後に状況を可視化する。



## 対象となる構成管理ツールと不具合管理票

- 構成管理ツール

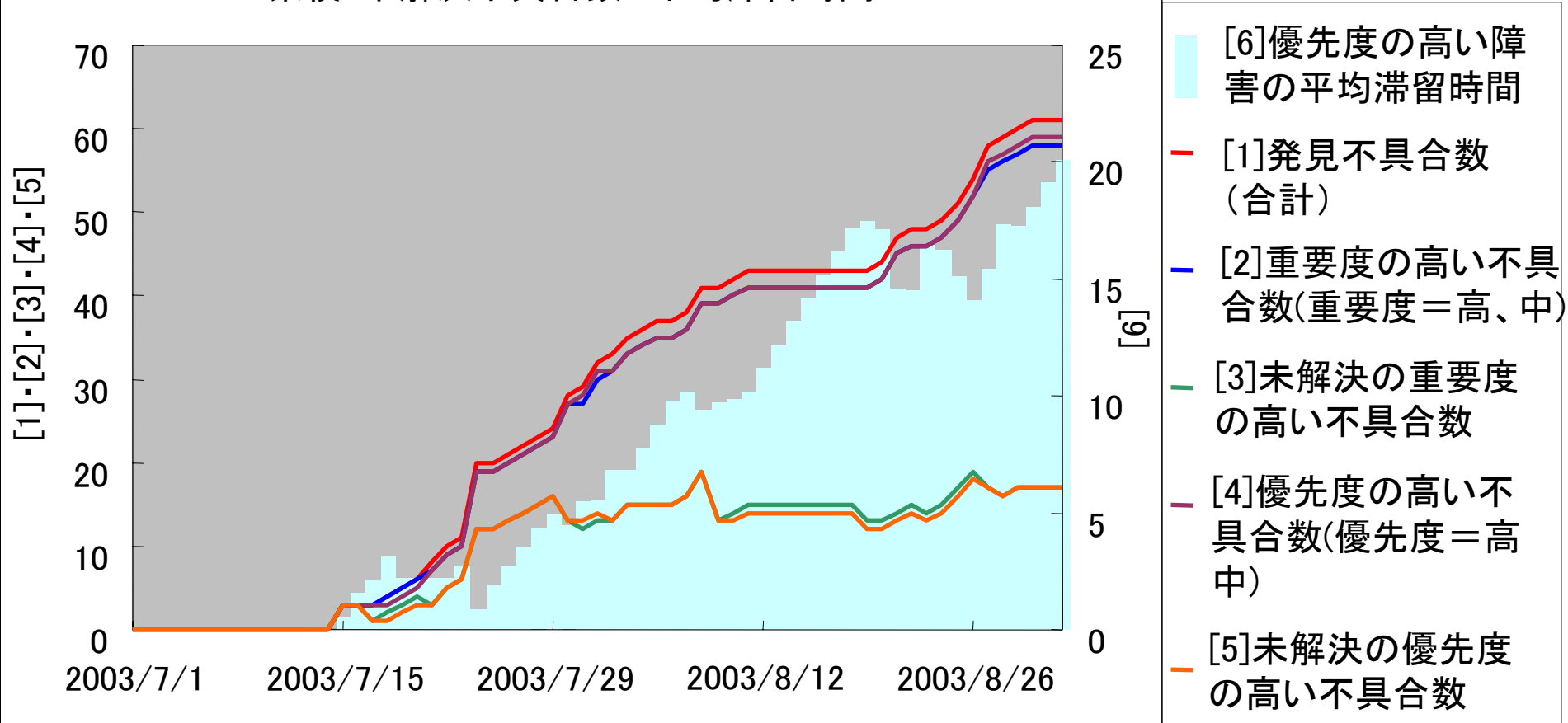
- 機能毎、機器毎のソースコードの行数遷移をコストをかけずに集計することが目的  
→ 行数を数えても原則的には同じことができる。
- 修正の波及範囲を調べるためには、ある程度の運用ルールが必要になる。

- 不具合管理票

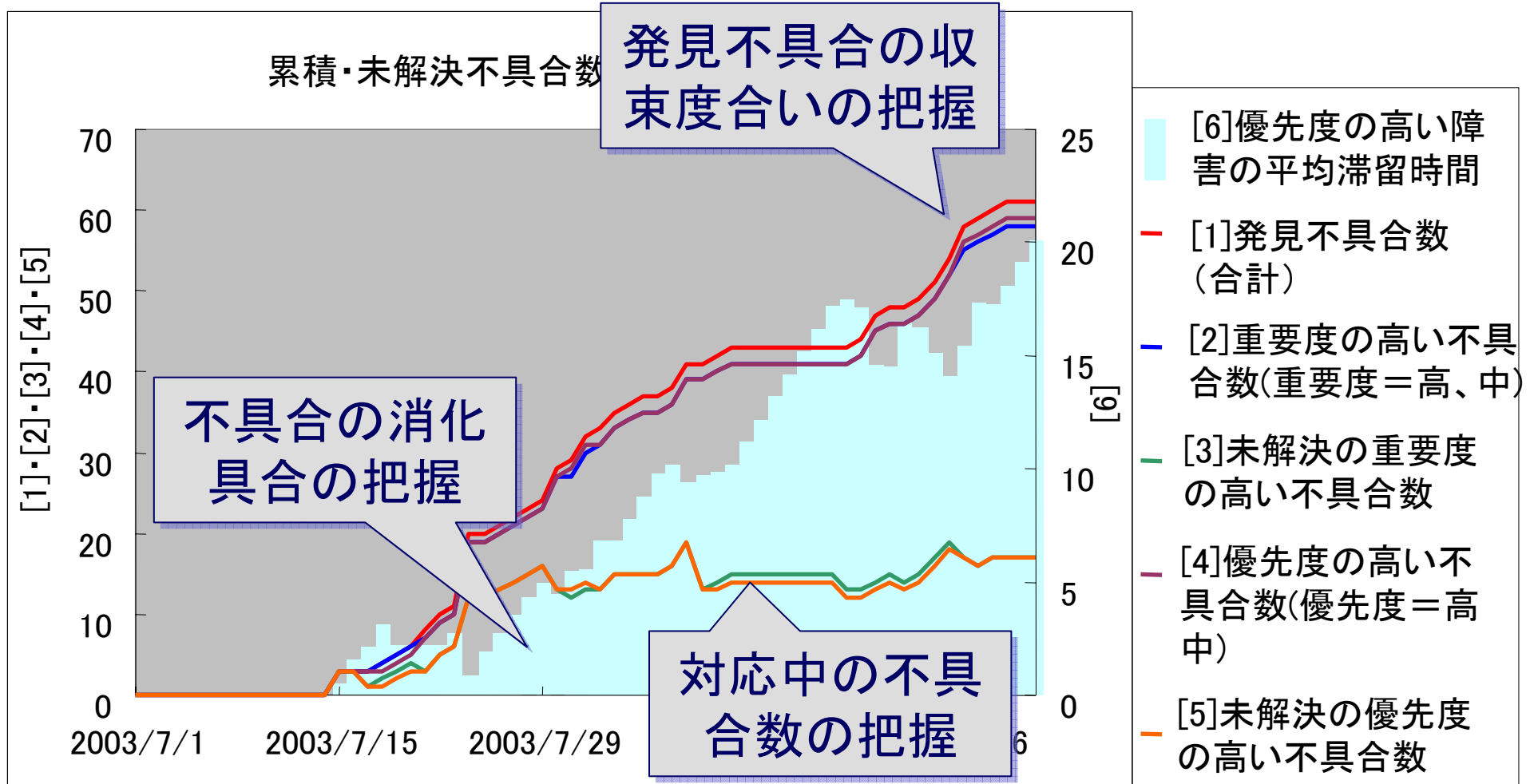
- 不具合の緊急度合いと発生、除去までの日数を得ることが目的
- バグ以外の項目（要望、差し戻し項目等）とは区別がつくものとする。

# 不具合管理票からの品質リスク推定

累積・未解決不具合数と平均滞留時間

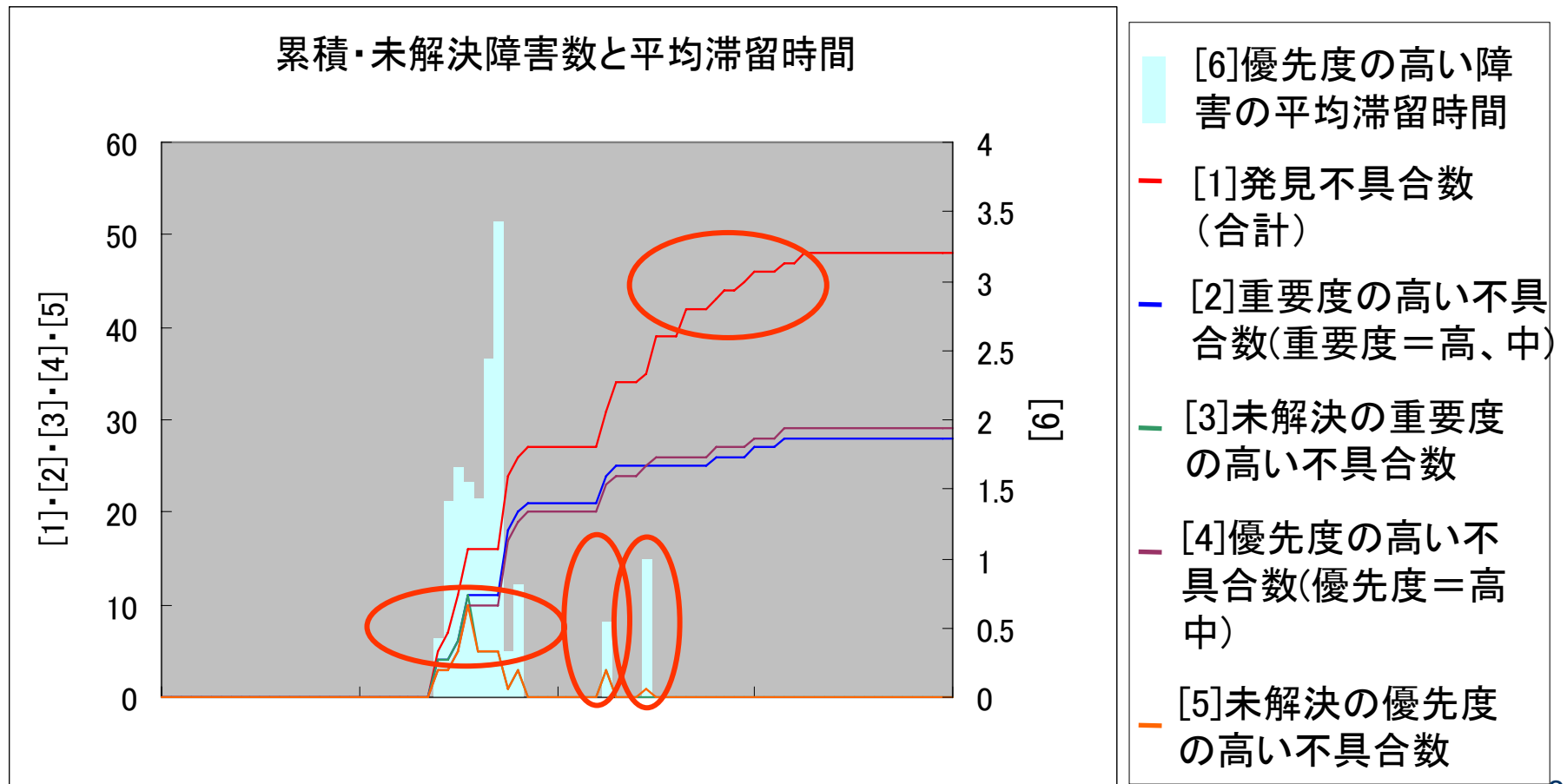


# 不具合管理票からの品質リスク推定



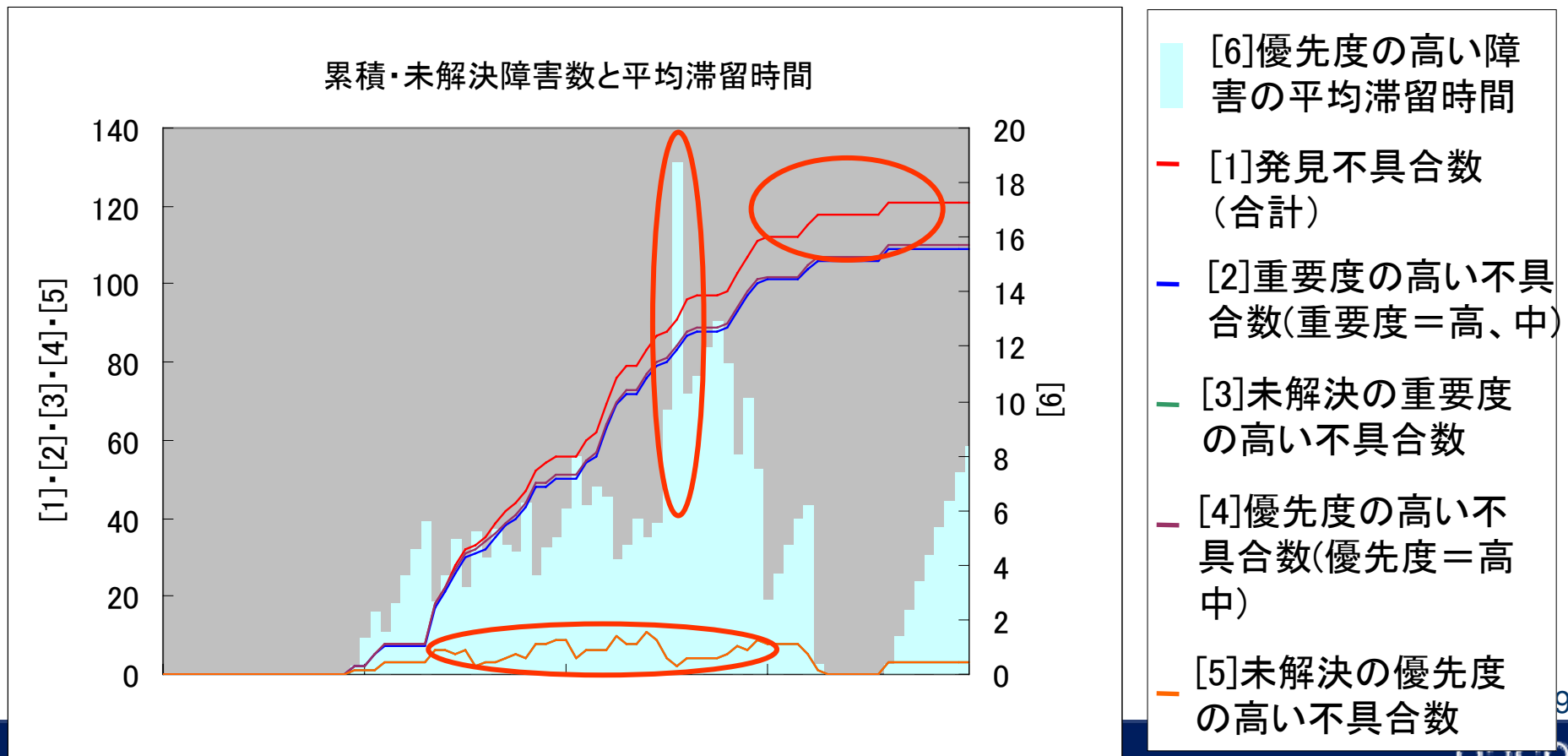
# 不具合管理票からの品質リスク推定

- 順調な例: 滞留時間/未対応不具合数が少ない。



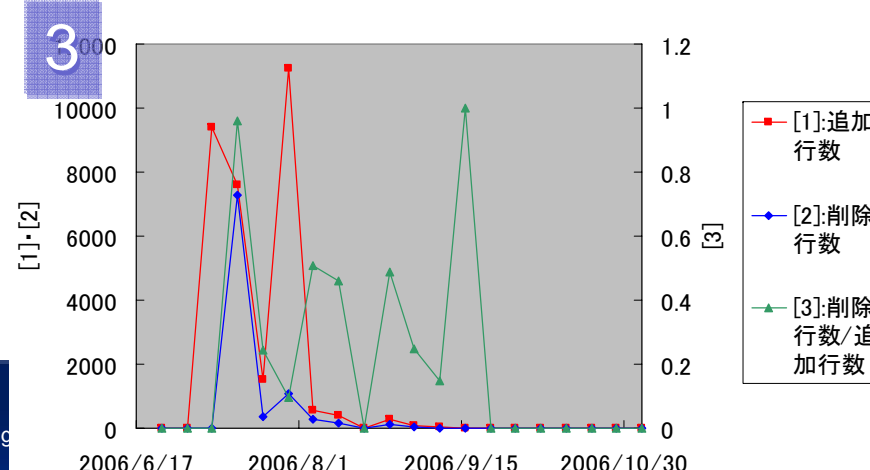
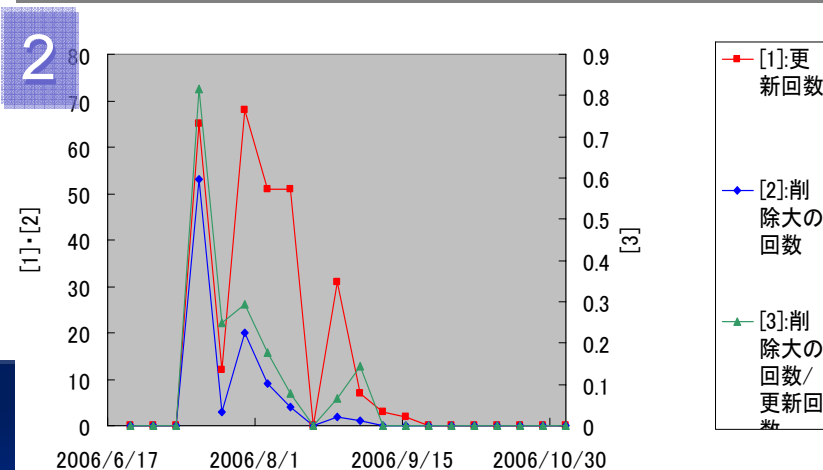
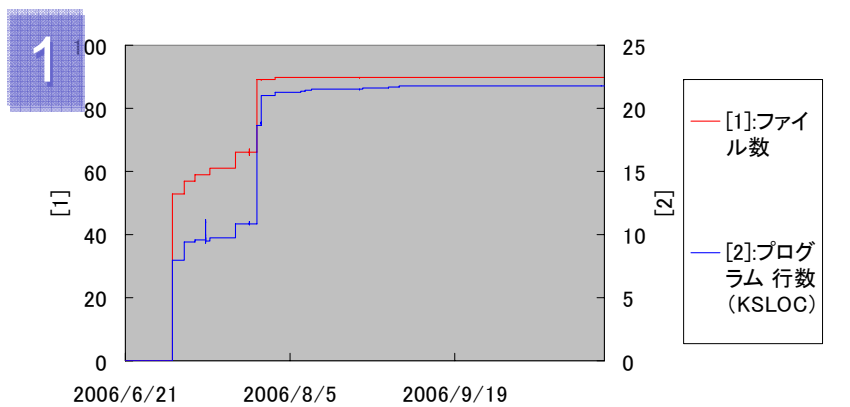
# 不具合管理票からの品質リスク推定

- 品質リスクの可能性がある例:  
滞留時間が長い/未対応不具合数が減らない。



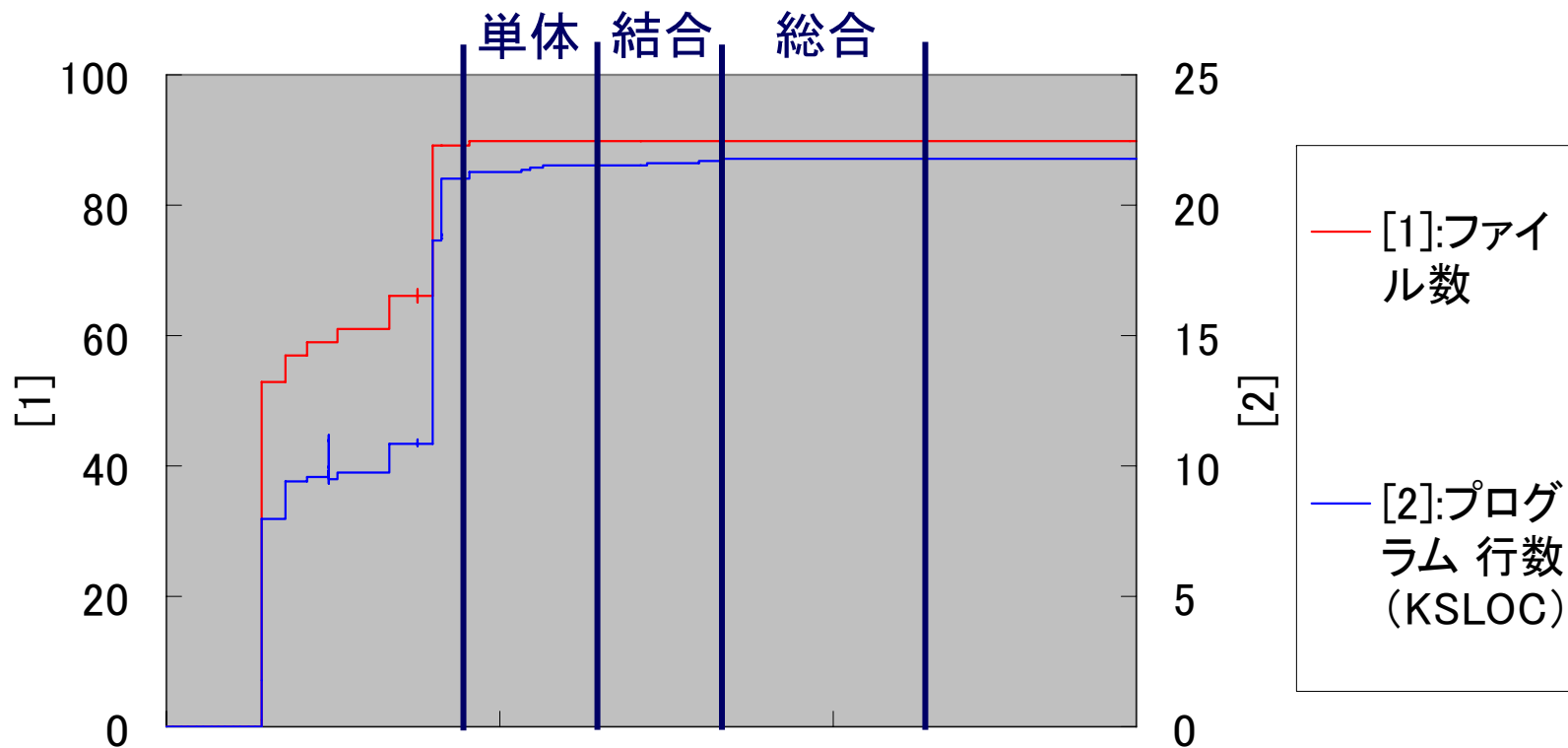
## 構成管理ツールからの品質リスク推定

- 一般に下流の開発管理に使われている規模遷移と同様に納期遅延/品質リスクを把握する。
- 1→3の順で段階的に規模遷移が詳細化されている。



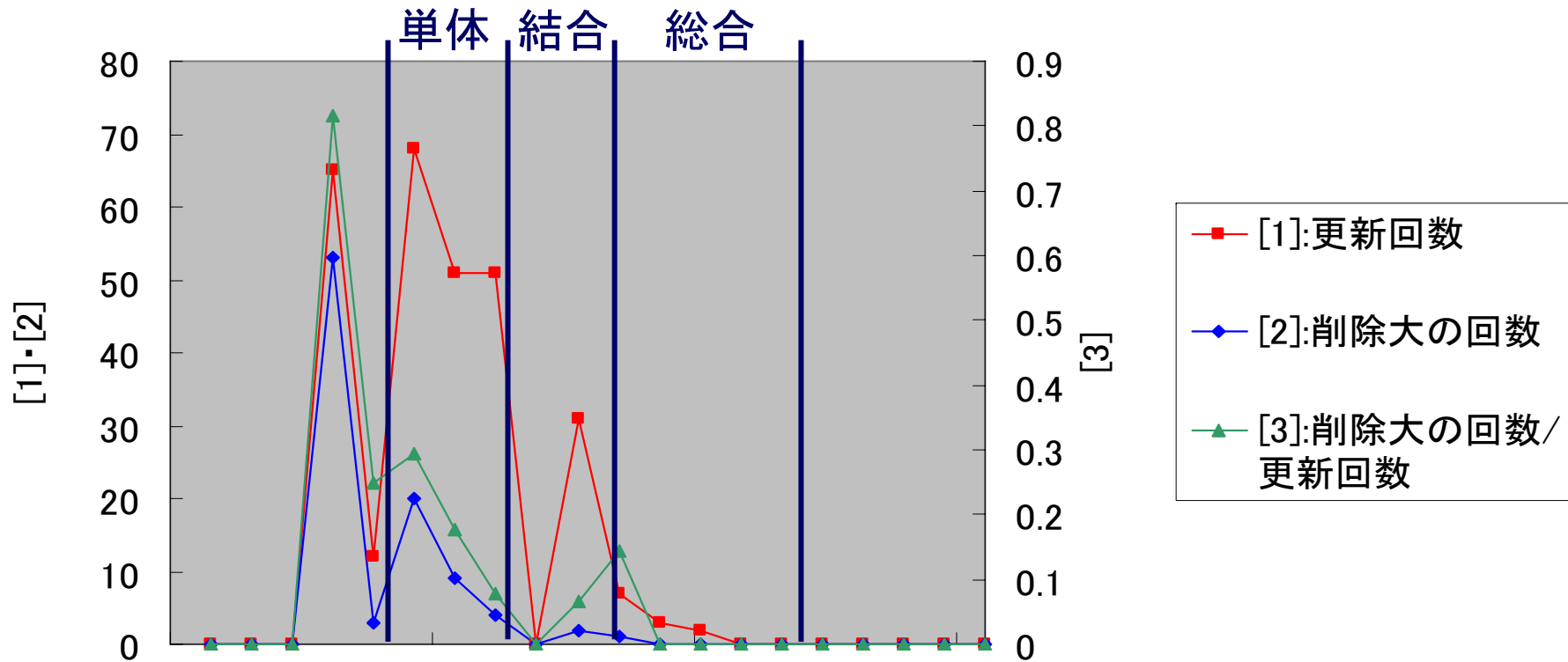
# 構成管理ツールからの品質リスク推定 (1/3)

- 機能やモジュール単位で規模推移をグラフ化する。

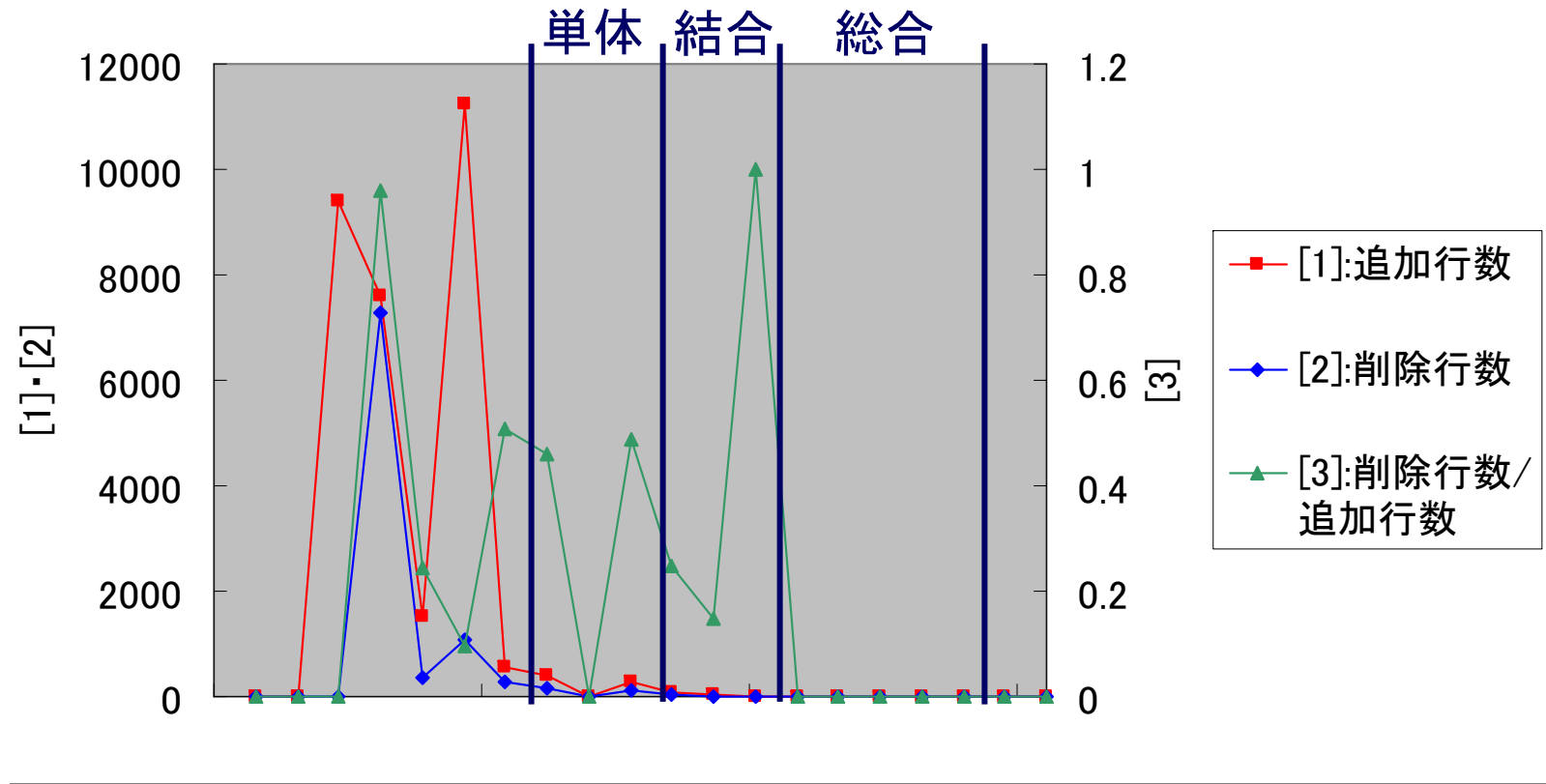


# 構成管理ツールからの品質リスク推定 (2/3)

- 更新回数から収束度合いを推測する。
- 「1回の修正につき1コミット」という運用ルールが前提

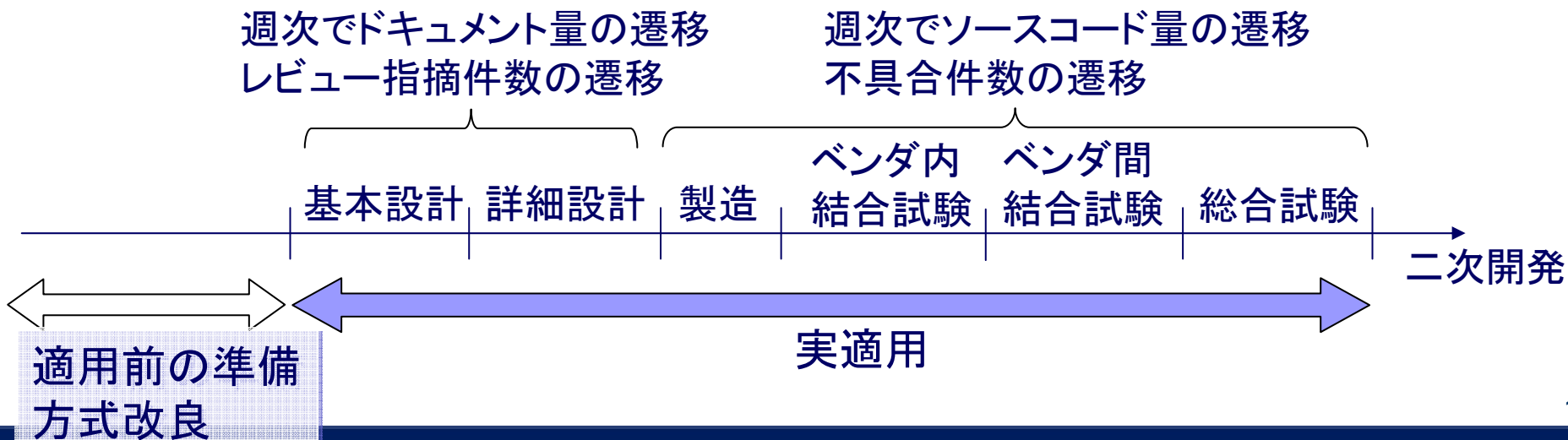
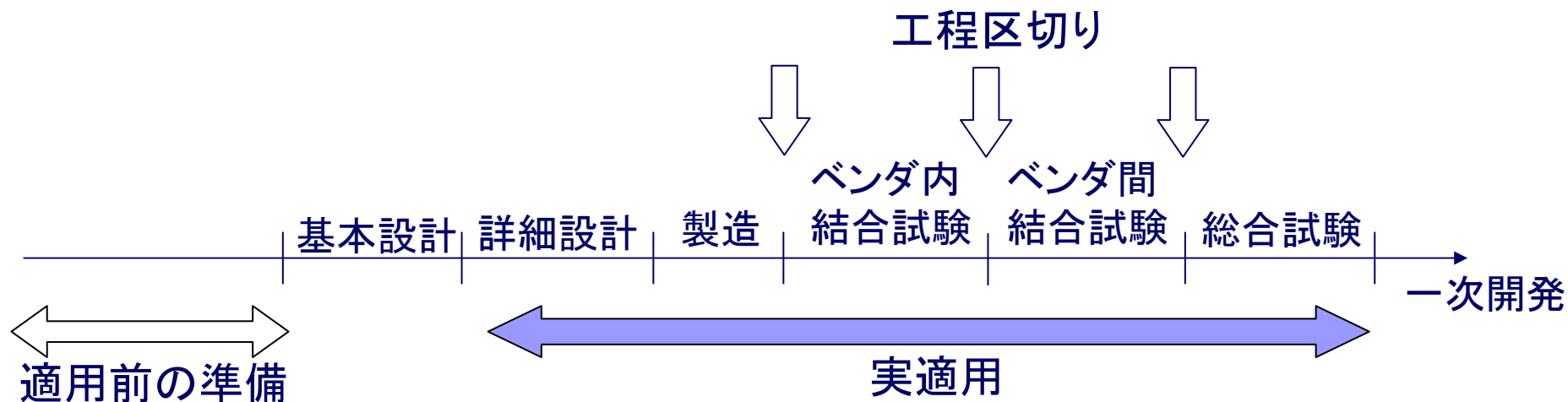


- 変更規模が大きい場合には、さらにソースコードファイルのリストを参照し、回帰テスト実施範囲等を検討する。



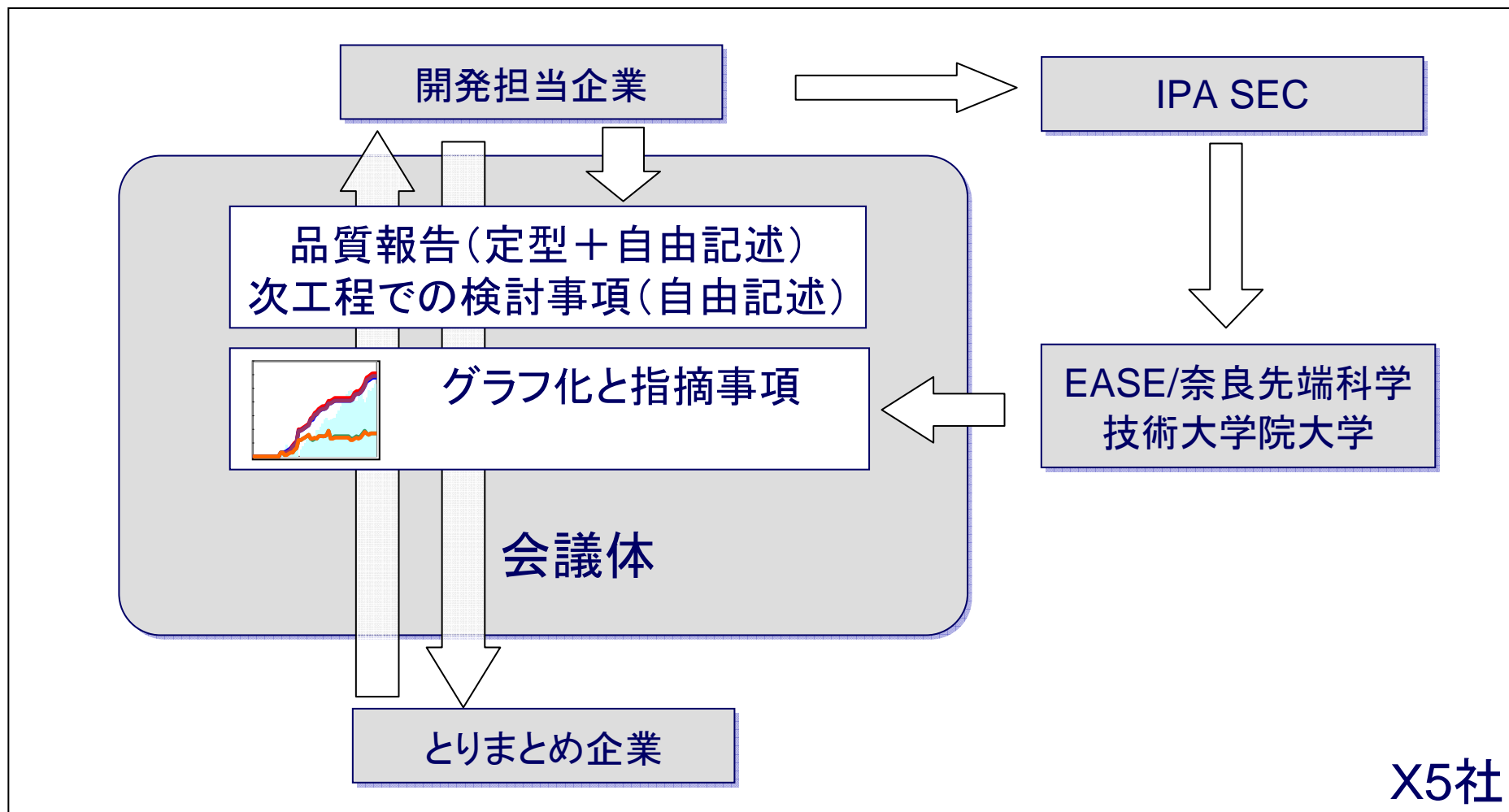
- ソフトウェアエンジニアリング技術研究組合での開発が対象
- 開発ドメイン
  - プローブ情報(センサーネットワーク)システム
  - エンタープライズ系
- 規模、プロセス
  - 複数ベンダ(とりまとめ1社、開発5社)による中規模開発
  - ウォータフォール

## 適用フェーズ



インプロセス分析 適用事例(進行中)

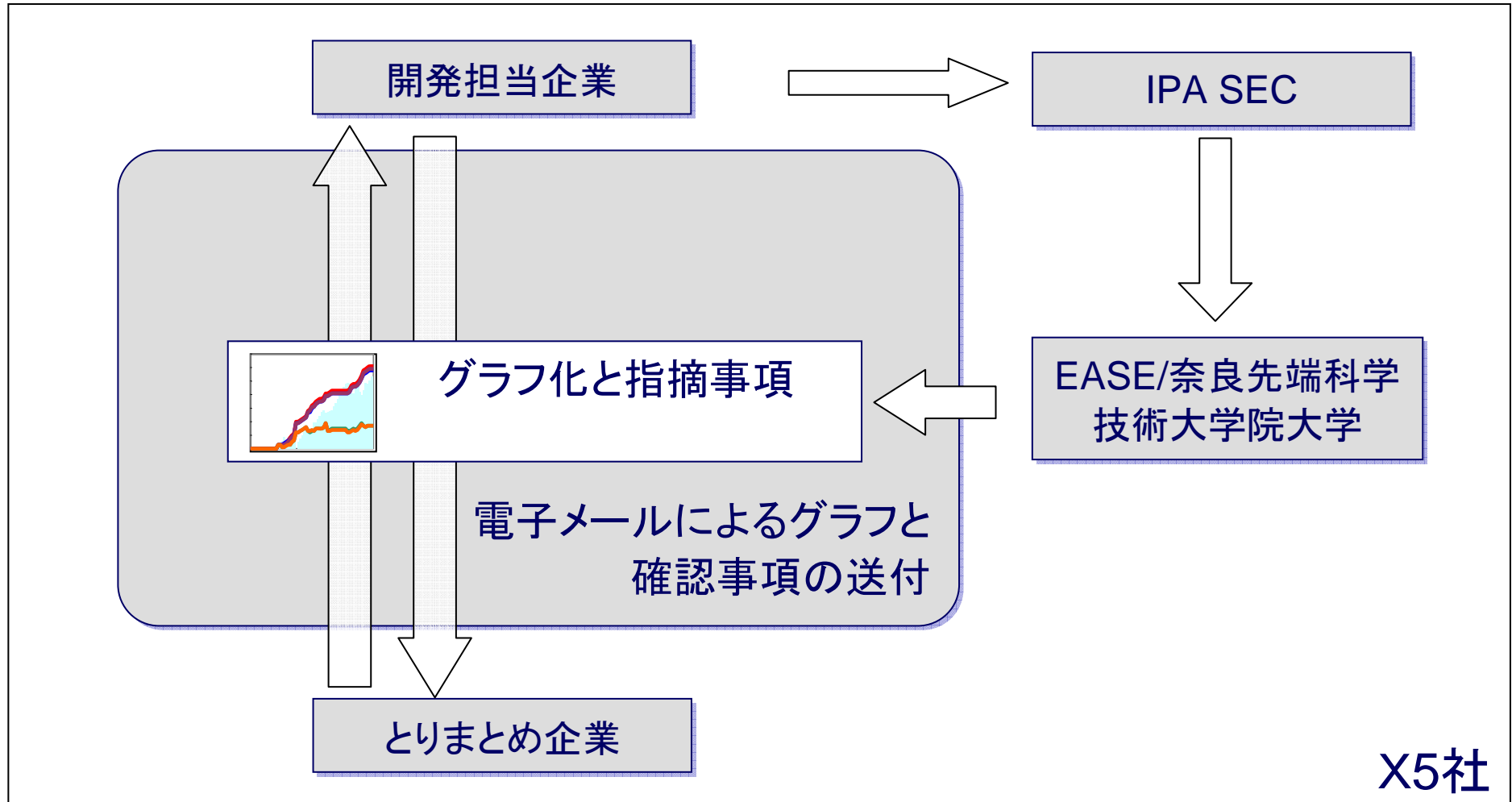
# 工程区切り会議(単体、社内結合、社間結合、総合)



X5社

▶ 16

# インプロセス分析 適用事例(進行中) メールによるフィードバック

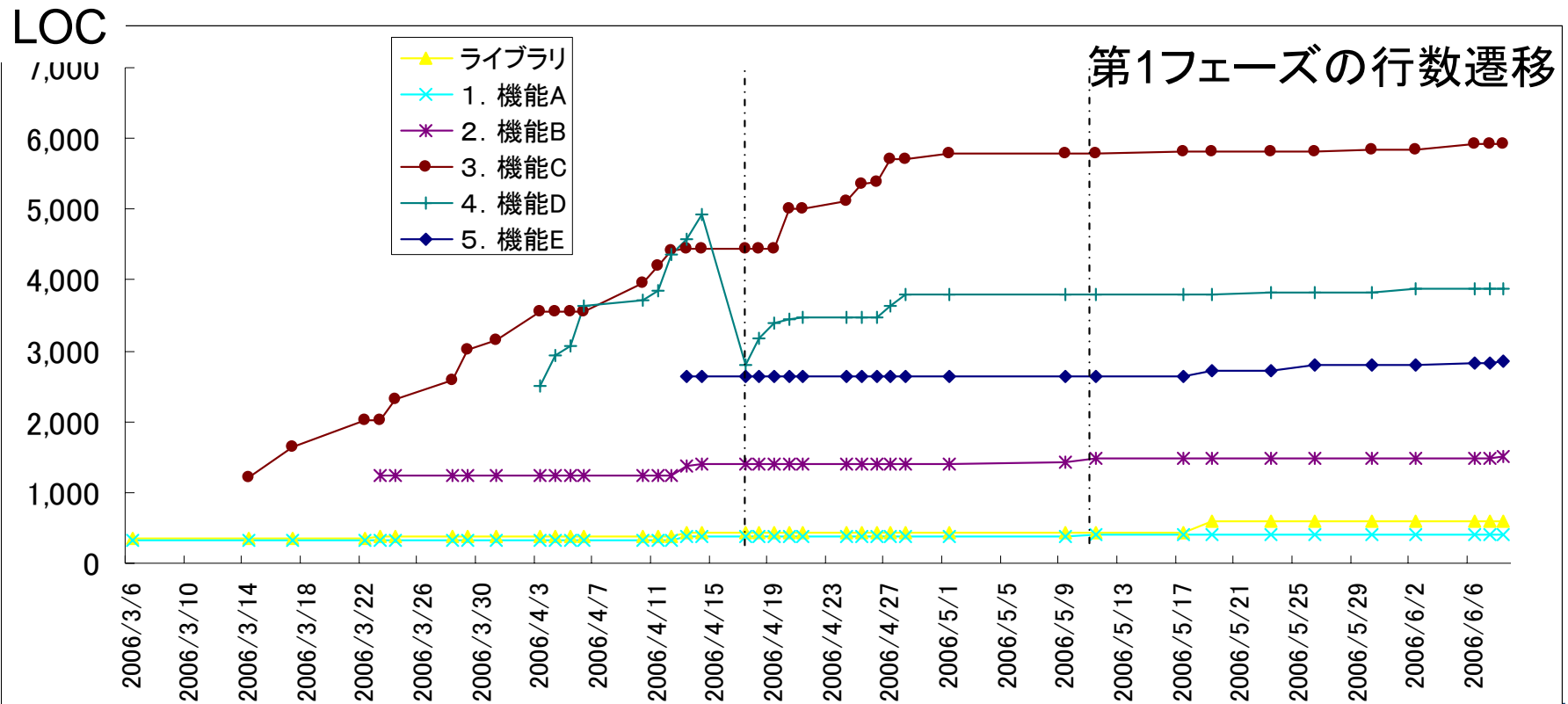


X5社

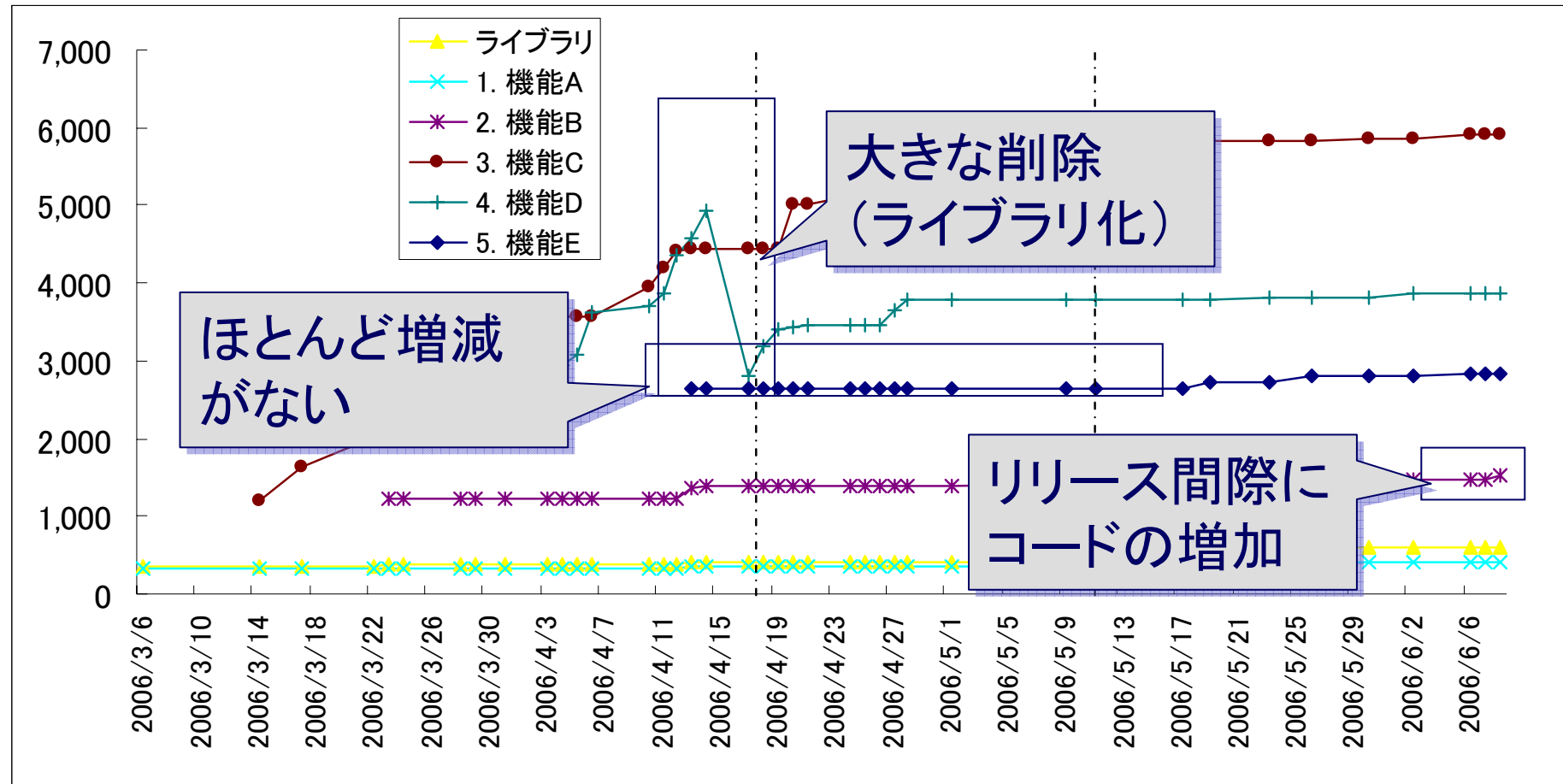
17

# 拡張開発の計画に母体の規模遷移を利用

- 第1フェーズの機能毎の行数遷移をもとに第2フェーズ開始前に機能毎の品質推定

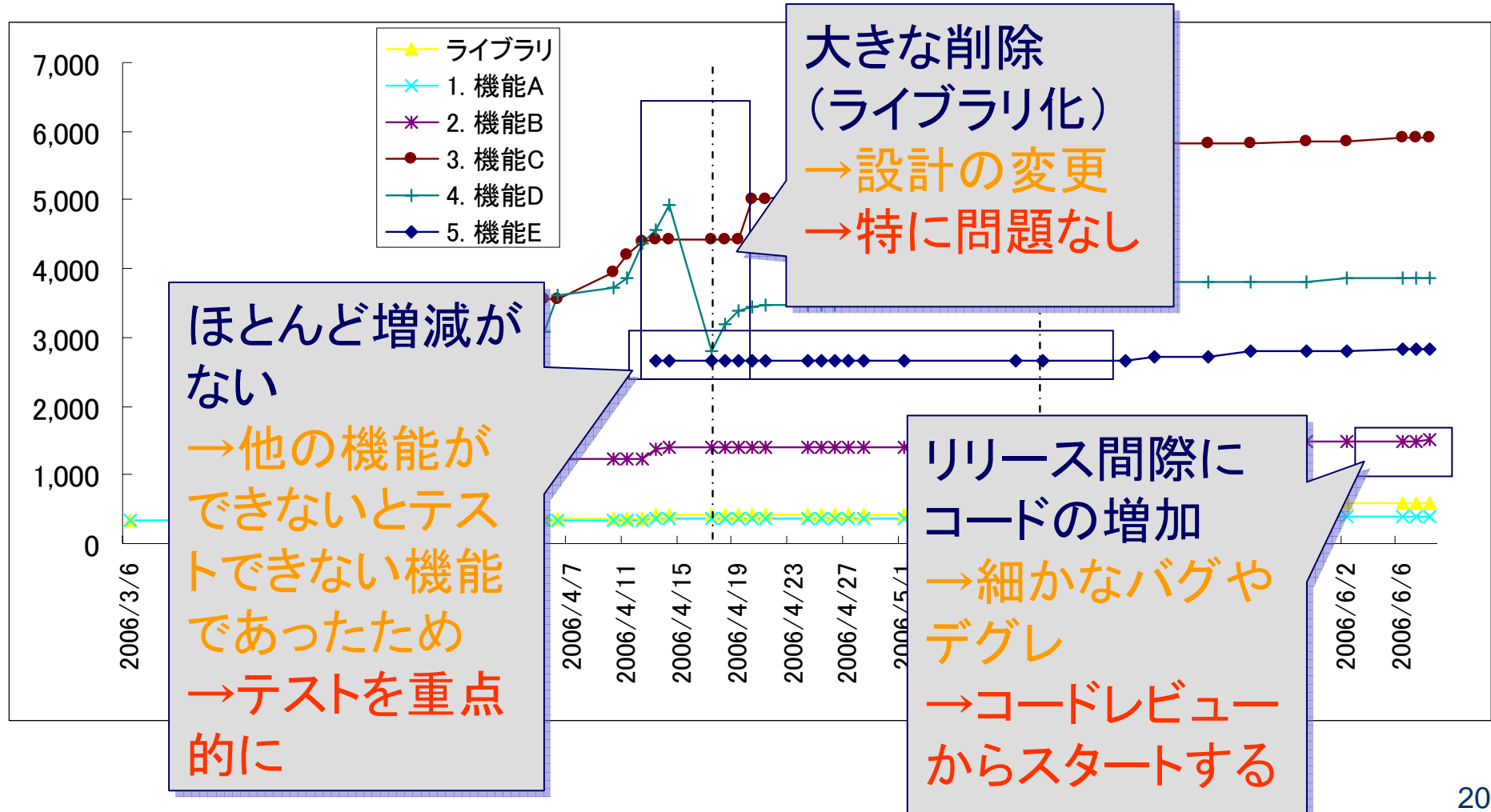


# 拡張開発の計画に母体の規模遷移を利用



# インプロセス分析 適用事例(完了後)

## 拡張開発の計画に母体の規模遷移を利用



- 進行中プロジェクトへの適用
  - 会議時間の短縮
  - 品質低下の可能性が高い部分の指摘
- 完了後プロジェクトへの適用
  - 安定している部分、より多くのテストが必要な部分の切り分け
- 他のプロジェクトへの適用可能性
  - 「リスク把握や品質推定に関して他のプロジェクトに適用した場合でも役立つだろう」
  - 「収集データの客観性により透明性が上がり、品質に対して前向きになれる」

# インプロセス分析 参考文献

一部は <http://se.naist.jp/cgi-bin/pman/pman.cgi> より入手できます

- [1] 松村知子, 門田暁人, 森崎修司, 松本健一, “マルチベンダ情報システム開発における障害修正工数の要因分析,” 情報処理学会論文誌, Vol.48, No.5, pp.1926-1935, 2007.
- [2] 大平 雅雄, 横森 励士, 阪井 誠, 岩村 聡, 小野 英治, 新海 平, 横川 智教, “ソフトウェア開発プロジェクトのリアルタイム管理を目的とした支援システム,” 電子情報通信学会論文誌D-I, Vol.J88-D-I, No.2, pp.228-239, 2005.
- [3] 松村知子, 森崎修司, 玉田春昭, 大杉直樹, 門田暁人, 松本健一, “設計書の再利用を考慮したレビュー効率の比較方法の提案と事例紹介,” Information Science Technical Report, NAIST-TR2007007, 奈良先端科学技術大学院大学, 2007.3.
- [4] 松村知子, 森崎修司, 玉田春昭, 大杉直樹, 門田暁人, 松本健一, “プロセス改善を目的とするODCを用いた欠陥修正工数分析,” Information Science Technical Report, NAIST-TR2007006, 奈良先端科学技術大学院大学, 2007.3.
- [5] 松村知子, 森崎修司, 玉田春昭, 大杉直樹, 門田暁人, 松本健一, “GQMモデルに基づく設計工程完成度計測手法の提案,” Information Science Technical Report, NAIST-TR2007005, 奈良先端科学技術大学院大学, 2007.3.
- [6] 玉田春昭, 松村知子, 森崎修司, 松本健一, “プロジェクト遅延リスク検出を目的とするソフトウェア開発プロセス可視化ツールProStar,” Information Science Technical Report, NAIST-TR2007002, 奈良先端科学技術大学院大学, 2007.2.
- [7] 松村知子, 勝又敏次, 森崎修司, 玉田春昭, 大杉直樹, 門田暁人, 楠本真二, 松本健一, “自動データ収集・可視化ツールを用いたリアルタイムフィードバックシステムの構築と試行,” Information Science Technical Report, NAIST-TR2007001, 奈良先端科学技術大学院大学, 2007.2.
- [8] 松村 知子, 横森 励士, 大杉 直樹, 川口 真司, 松下 誠, “ファイルの同時変更パターンと変更差分の分析による論理的結合関係の自動抽出,” ソフトウェアシンポジウム2005, June 2005. (富山国際会議場)

# アジェンダ

- プロジェクトモニタリング（インプロセス分析）
  - ソースコード構成管理システム
  - 不具合管理票（バグ票）
  - 適用事例（進行中、完了後）
- リポジトリを対象とした見積り/傾向分析
  - 過去の事例との類似度による見積り（協調フィルタリング見積り）
  - 相関ルール分析による傾向分析（NEEDLE）

# リポジトリを対象とした見積り/傾向分析

- 過去のプロジェクトのサマリデータ
  - プロファイル、規模、品質、コストの項目を含む。

| ID  | プロファイル   |      |     | 規模   |     | 品質        |     | コスト |     |
|-----|----------|------|-----|------|-----|-----------|-----|-----|-----|
|     | プラットフォーム | 開発種別 | ... | KLOC | ... | 不具合密度     | ... | 総工数 | ... |
| 1   | Windows  | 拡張   | ... | 150  | ... | 0.12/KLOC | ... | 250 | ... |
| 2   | Windows  | 拡張   | ... | 200  | ... | 0.14/KLOC | ... | 350 | ... |
| 3   | HP-UX    | 拡張   | ... | 250  | ... | 0.21/KLOC | ... | 420 | ... |
| 4   | LINUX    | 保守   | ... | 300  | ... | 0.12/KLOC | ... | 300 | ... |
| ... | ...      | ...  |     | ...  |     | ...       |     | ... |     |

# 協調フィルタリング見積りの特徴

- 収集データに欠損がある場合でも適用できる。

|       | 項目 1     | 項目 2  | 項目 3  | ...   | 項目 122 |         |
|-------|----------|-------|-------|-------|--------|---------|
| 製品群 A | プロジェクト 1 | 値 1-1 | 値 1-2 | 欠損値   | ...    | 欠損値     |
|       | プロジェクト 2 | 値 2-1 | 値 2-2 | 欠損値   | ...    | 欠損値     |
|       | プロジェクト 3 | 値 3-1 | 値 3-2 | 欠損値   | ...    | 欠損値     |
| 製品群 B | プロジェクト 4 | 欠損値   | 値 4-2 | 値 4-3 | ...    | 欠損値     |
|       | プロジェクト 5 | 欠損値   | 値 5-2 | 値 5-3 | ...    | 欠損値     |
|       | プロジェクト 6 | 欠損値   | 値 6-2 | 値 6-3 | ...    | 欠損値     |
| 製品群 C | プロジェクト 7 | 欠損値   | 欠損値   | 値 7-3 | ...    | 値 7-122 |
|       | プロジェクト 8 | 欠損値   | 欠損値   | 値 8-3 | ...    | 値 8-122 |
|       | プロジェクト 9 | 欠損値   | 欠損値   | 値 9-3 | ...    | 値 9-122 |

# 協調フィルタリング見積り例(工数)

- 現行プロジェクトと類似度の高い過去プロジェクトを選ぶ。
- 類似プロジェクトの工数を加重平均し、現行プロジェクトの値(工数)を予測する。

|                       | プラットフォーム | 開発種別 | 概算 FP | 要員数 | 開発総工数 |
|-----------------------|----------|------|-------|-----|-------|
| 現行プロジェクト X            | Windows  | 新規   | 1500  | 10  | 欠損値   |
| 類似度: +1.0<br>プロジェクト 1 | Windows  | 新規   | 欠損値   | 8   | 400   |
| 類似度: +0.9<br>プロジェクト 2 | HP-UX    | 欠損値  | 2500  | 6   | 350   |
| 類似度: -1.0<br>プロジェクト 3 | 欠損値      | 保守   | 3000  | 20  | 2500  |

# 協調フィルタリング見積り例(工数)

- 現行プロジェクトと類似度の高い過去プロジェクトを選ぶ。
- 類似プロジェクトの工数を加重平均し、現行プロジェクトの値(工数)を予測する。

|                       | プラットフォーム | 開発種別 | 概算 FP | 要員数 | 開発総工数      |
|-----------------------|----------|------|-------|-----|------------|
| 現行プロジェクト X            | Windows  | 新規   | 1500  | 10  | 予測値: 402.5 |
| 類似度: +1.0<br>プロジェクト 1 | Windows  | 新規   | 欠損値   | 8   | 400        |
| 類似度: +0.9<br>プロジェクト 2 | HP-UX    | 欠損値  | 2500  | 6   | 350        |
| 類似度: -1.0<br>プロジェクト 3 | 欠損値      | 保守   | 3000  | 20  | 2500       |

# 協調フィルタリング見積り例(工数)

- 現行プロジェクトと類似度の高い過去プロジェクトを選ぶ。
- 類似プロジェクトの工数を加重平均し、現行プロジェクトの値(工数)を予測する。
- 見積り根拠となる類似したプロジェクトが提示される。

|                       | プラットフォーム | 開発種別 | 概算 FP | 要員数 | 開発総工数      |
|-----------------------|----------|------|-------|-----|------------|
| 現行プロジェクト X            | Windows  | 新規   | 1500  | 10  | 予測値: 402.5 |
| 類似度: +1.0<br>プロジェクト 1 | Windows  | 新規   | 欠損値   | 8   | 400        |
| 類似度: +0.9<br>プロジェクト 2 | HP-UX    | 欠損値  | 2500  | 6   | 350        |
| 類似度: -1.0<br>プロジェクト 3 | 欠損値      | 保守   | 3000  | 20  | 2500       |

# リポジトリ分析 実績

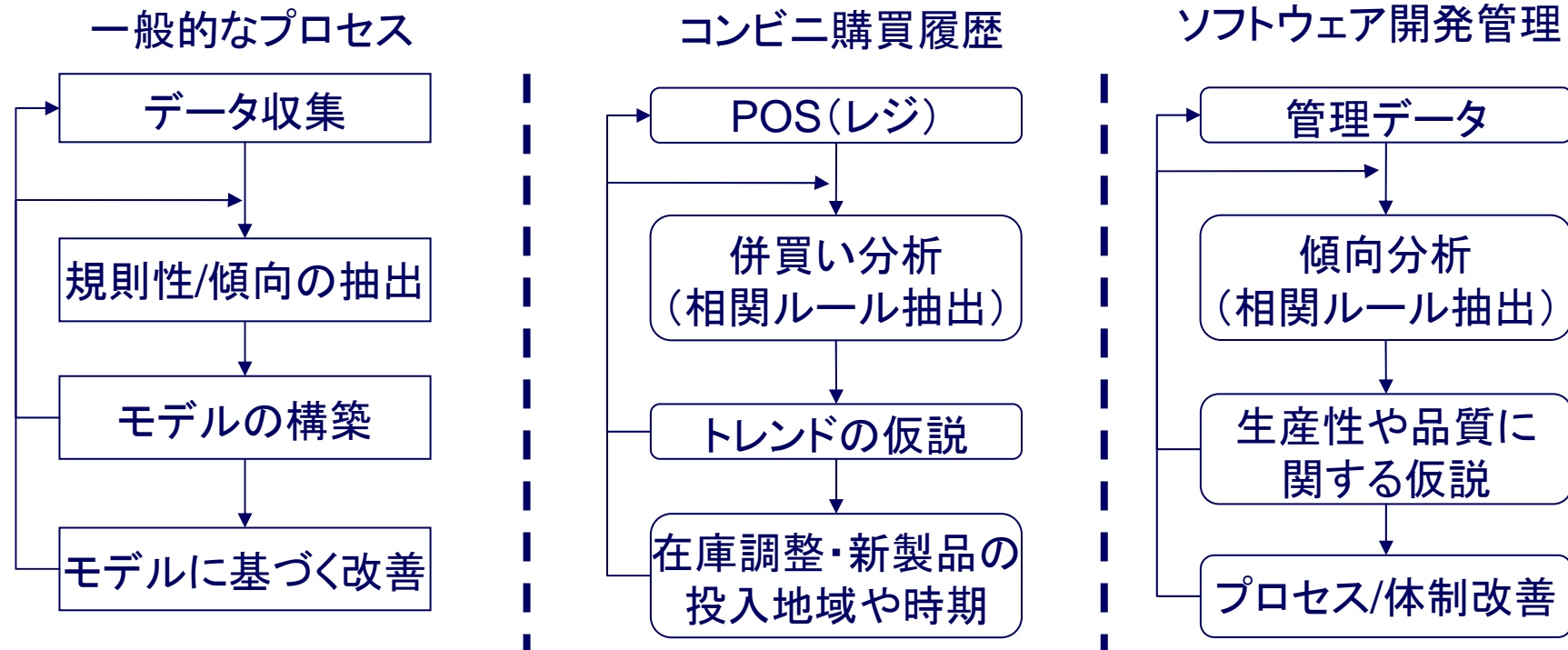
|                 | 目的                                | データ  | 結果                                  | 組織                     |
|-----------------|-----------------------------------|--|-------------------------------------|------------------------|
| [1]<br>-<br>[2] | 開発総工数見<br>積もり                     | 特性変数: 13 種類<br>プロジェクト: 1081 件<br>データ欠損率: 60 %              | 平均誤差: 79%<br>的中率: 37%               | NTTデータ様                |
| [3]             | 企業横断的<br>データを用いた<br>開発総工数見<br>積もり | 特性変数: 97 種類<br>プロジェクト: 378 件<br>(15 社から収集)<br>データ欠損率: 67 % | 平均誤差: 64%<br>的中率: 30%               | IPA SEC様               |
| [4]             | プロジェクト早<br>期のシステム<br>規模見積もり       | 特性変数: 20 種類<br>プロジェクト: 85 件<br>データ欠損率: 7 %                 | 平均誤差: 28%<br>的中率: 56%               | 日立システム<br>アンドサービ<br>ス様 |
| [5]             | コスト超過リス<br>クの予測(判別<br>予測)         | 特性変数: 199 種類<br>プロジェクト: 45 件<br>データ欠損率: 42 %               | 適合率: 0.73<br>再現率: 1.00<br>F1値: 0.84 | 日立製作所様                 |

一部は <http://se.naist.jp/cgi-bin/pman/pman.cgi> より入手できます

- [1] N. Ohsugi, M. Tsunoda, A. Monden, and K. Matsumoto, "Effort estimation based on collaborative filtering," In *Proc. of 5th International Conference on Product Focused Software Process Improvement (Profes2004)*, Lecture Notes in Computer Science, Vol.3009, pp.274-286 (2004).
- [2] 角田雅照, 大杉直樹, 門田暁人, 松本健一, 佐藤慎一, "協調フィルタリングを用いたソフトウェア開発工数予測方法," *情報処理学会論文誌*, vol.46, no.5, pp.1155-1164 (2005).
- [3] 大杉直樹, 角田雅照, 門田暁人, 松村知子, 松本健一, 菊地奈穂美, "企業横断的収集データに基づくソフトウェア開発プロジェクトの工数見積もり," *SEC journal*, No.5, pp.16-25 (2006).
- [4] 大杉直樹, 松本健一, 津田道夫, 中屋広樹, 十九川博幸, "協調フィルタリング技術によるソフトウェア規模の予測," *日立システムジャーナル* (2006).
- [5] 本村拓也, 柿元健, 角田雅照, 大杉直樹, 門田暁人, 松本健一, "協調フィルタリングを用いたプロジェクトコスト超過の予測," *信学術報*, SS2005-39, pp.35-40 (2005).
- [6] 柿元健, 角田雅照, 大杉直樹, 門田暁人, 松本健一, "協調フィルタリングに基づく工数見積もりのロバスト性評価," *日本ソフトウェア科学会FOSE2004, ソフトウェア工学の基礎XI*, pp.73-84, (2004).
- [7] 柿元健, 角田雅照, 大杉直樹, 門田暁人, 松本健一, "協調フィルタリングによる工数見積もり手法におけるデータ数と見積もり精度の関係の分析," *日本ソフトウェア科学会FOSE2005, ソフトウェア工学の基礎XI*, pp.77-86 (2005).

# プロジェクトデータ/品質データの相関ルール分析

- コンビニでは在庫調整の業務プロセス中に組み込まれている。



## 小売店の販売履歴の相関ルール分析例

- 対象データに含まれる「AならばB」という規則(相関ルール)を全て列挙する。
- 列挙されたルールから解釈を与えることができるルールを人手により探し、役立てる。
- コンビニの購買履歴から得た相関ルールの例  
休日に「レジャーシート」を買う顧客は「おにぎり」と「お茶」も同時に買っている。  
「(曜日=土日) and おにぎり and お茶 ⇒ レジャーシート」  
→ 休日には、レジャーシートの配置をおにぎりかお茶に近づけ、発見率、併せ買い率を上げる。

- 表記 :  $A \Rightarrow B$  (平均、標準偏差)、支持度、基準化平均、基準化標準偏差
  - 基準化平均 (全体平均に対する倍率)  
前提Aの有無による平均の比
  - 基準化標準偏差 (全体標準偏差に対する倍率)  
基準化平均と同様
- ルール例  
(開発種別 = 流用)かつ(母体安定度 = 低)  
 $\Rightarrow$  テスト工数比率 (平均: 32%、標準偏差 0.21)

一部は <http://se.naist.jp/cgi-bin/pman/pman.cgi>, <http://se.naist.jp/~morisaki/> より入手できます

[1] バグ票への適用 Shuji Morisaki, Akito Monden, Tomoko Matsumura, Haruaki Tamada, and Ken-ichi Matsumoto, "Defect Data Analysis Based on Extended Association Rule Mining," In Fourth International Workshop on Mining Software Repositories (MSR 2007), May 2007

[2] プロジェクトデータへの適用 Shuji Morisaki, Akito Monden, Haruaki Tamada, Tomoko Matsumura, and Ken-ichi Matsumoto, "Mining Quantitative Rules in A Software Project Data Set," 情報処理学会論文誌 2007/8掲載予定.

[3] 品質データ、プロジェクトデータへの適用 十九川 博幸: NEEDLEの適用事例の紹介  
[http://www.empirical.jp/research/publicdata/10th\\_kenkyukai/2-2.pdf](http://www.empirical.jp/research/publicdata/10th_kenkyukai/2-2.pdf)

[4] NEEDLEの紹介 森崎 修司: [http://www.empirical.jp/research/publicdata/10th\\_kenkyukai/2-1.pdf](http://www.empirical.jp/research/publicdata/10th_kenkyukai/2-1.pdf)

[5] バグ票への適用と例外ルール: 森崎修司, 松村知子, 玉田春昭, 大杉直樹, 門田暁人, 松本健一: 相関ルール分析を用いた障害対応データの特徴分析 <http://library.naist.jp/cgi-bin/limedio/limewwwopac/av1?sessionLang=jpn;sessionCode=utf8;bibid=96866>

# 実績 (NEEDLE: 相関ルール抽出ツール)

- ご検討企業様 (利用中含む)
  - NTTデータ様
  - キヤノンシステムソリューションズ様
  - 東芝様
  - デンソークリエイイト様
  - 日本ユニシス様
  - パナソニックMSE様
  - 日立システムアンドサービス様
  - 富士フイルム様
  - ほかに12社様