

# 「事故前提社会」への対応力を高める、 ユーザ・ベンダ間での情報共有技術 ～ソフトウェアタグ、タグ利用シナリオ、ツール～

奈良先端科学技術大学院大学  
ソフトウェア工学講座・ソフトウェア設計学講座  
文部科学省StagEプロジェクト

松本健一，角田雅照，伏田享平

ホームページ : <http://www.stage-project.jp/>

E-mail : [stage-contact@is.naist.jp](mailto:stage-contact@is.naist.jp)

# StagEプロジェクト

文部科学省 次世代IT基盤構築のための研究開発

「ソフトウェア構築状況の可視化技術の普及」

エンピリカルデータに基づくソフトウェアタグ技術の開発と普及

Software traceability and accountability for global software Engineering

(2007年8月～2012年3月)

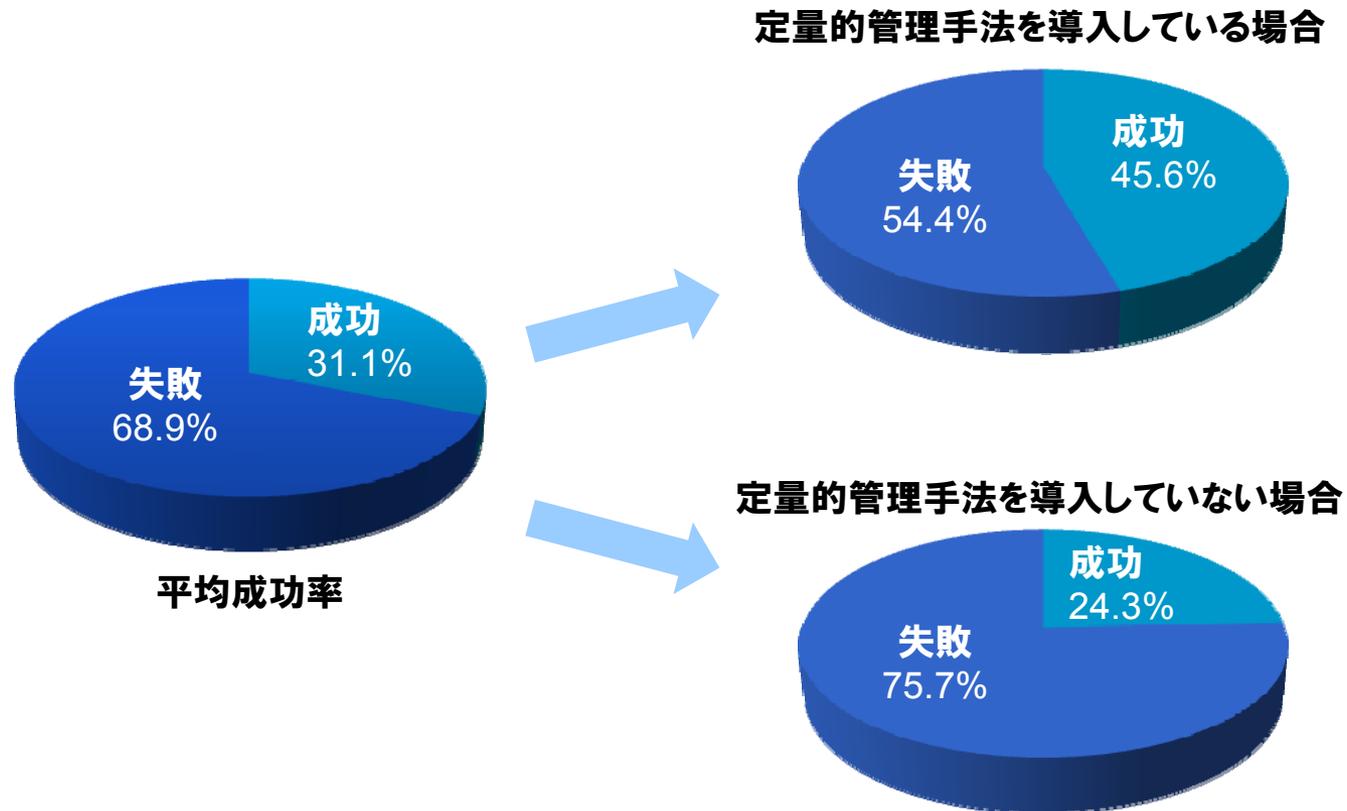
# プロジェクト成功率

観点 \ プロジェクト種別	ソフトウェア開発全般*	組込みソフトウェア開発**
品質	52%	71%
コスト	63%	43%
納期	55%	36%

\* “特集:プロジェクト成功率は33.1%”, 日経コンピュータ, 2008年12月1日号, pp.36-53, 2008.

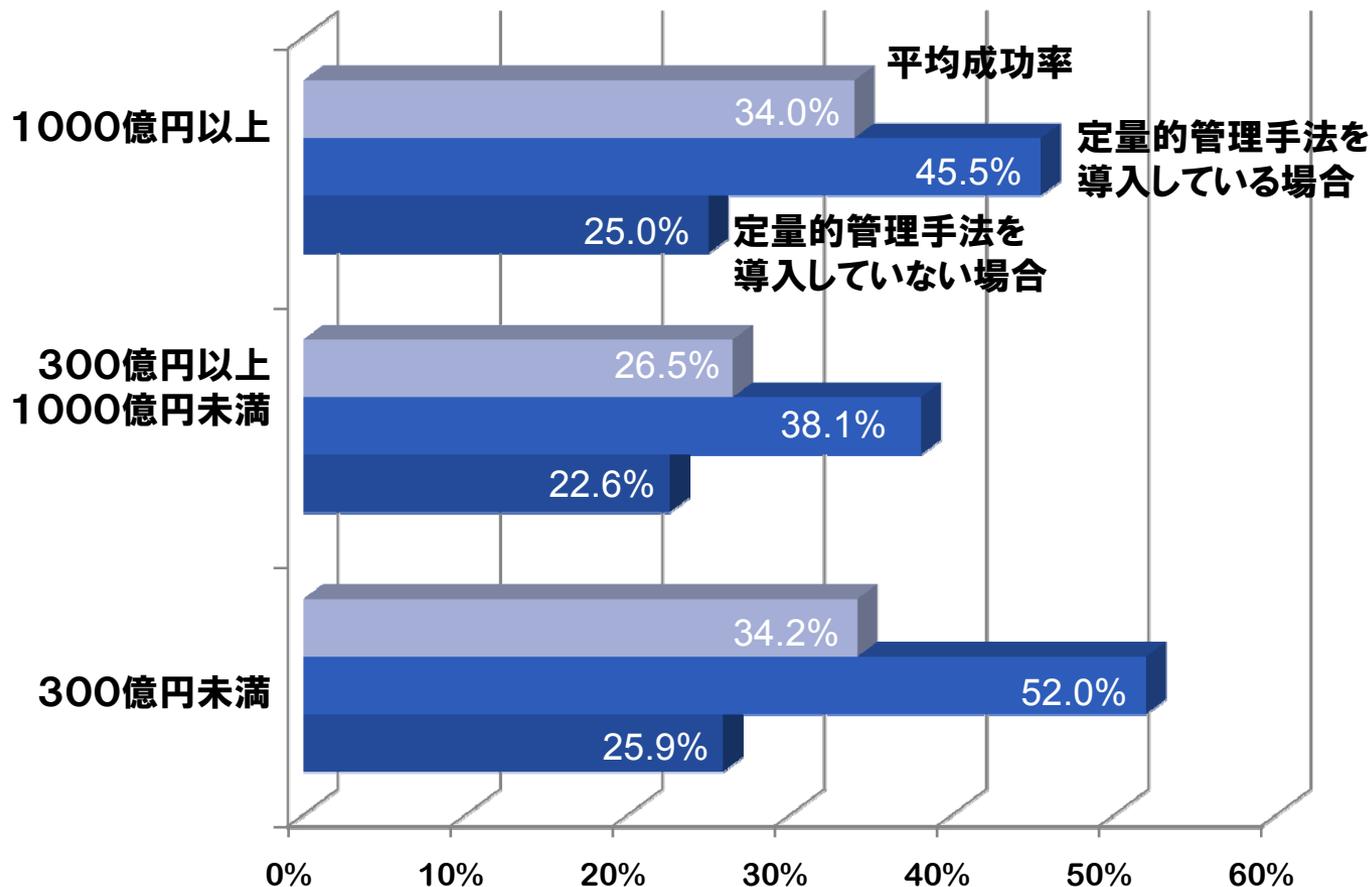
\*\* 2006年版組込みソフトウェア産業実態調査報告書, 経済産業省, 2006.

# プロジェクト成功率と定量的管理手法の相関



“特集:プロジェクト成功率は33.1%”, 日経コンピュータ, 2008年12月1日号, pp.36-53, 2008.

# 売上高別でみた成功率



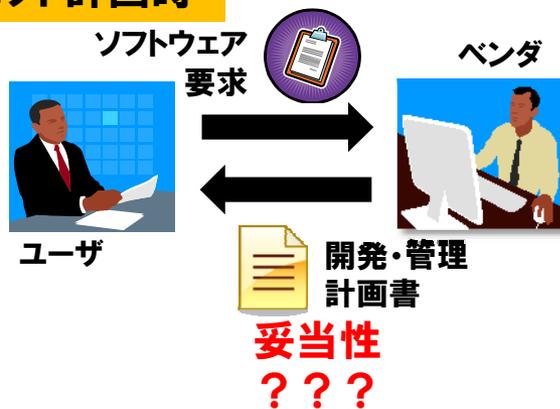
“特集:プロジェクト成功率は33.1%”, 日経コンピュータ, 2008年12月1日号, pp.36-53, 2008.

# 経済産業省ガイドライン

- 情報システムの利用者と供給者は、情報システムの信頼性・安全性の達成に向け、システムライフサイクルプロセス全体を通して実施体制、管理体制、仕組及びルール等を整備し、これらを活用しなければならない。
  - 経験則のみによらないプロジェクトマネジメントの導入
  - 定量データを活用した管理
  - 健全なプロジェクト運営に向けた活動の実施
  - 第三者によるレビュー及び監査の実施
  - 仕様変更の取扱いに関する利用者・供給者間での合意
  - 情報セキュリティ対策の実施

# コミュニケーション不足と開発リスク

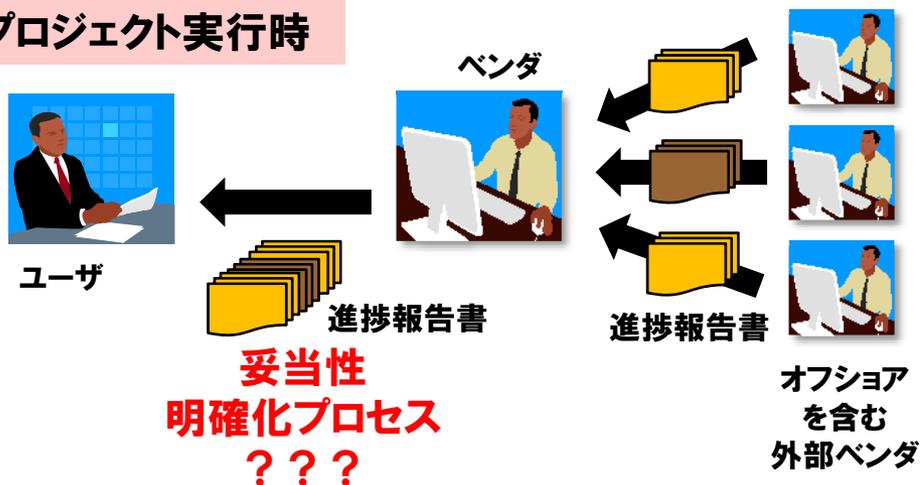
## プロジェクト計画時



- 「開発・管理計画」や「進捗報告」の妥当性を確認しない、できない。
- ソフトウェア要求が、プロジェクトの進捗とともに、どのように明確化され、実装、検証されてゆくか、ユーザは実感できない。



## プロジェクト実行時



- 開発リスクの増大
  - コスト・スケジュール超過
  - 品質低下(バグ多発)
  - モチベーション・モラル低下

# 膨大な社会的損失

- 米国では、ソフトウェアのバグによる年間損失額が595億ドル(国内総生産の0.6%) (米国商務省国立標準技術研究所(NIST)調査, 2002年)

日本にあてはめると**社会的損失約3兆円**

日本のGDP515.9兆円(2007年, 名目). 日米でソフトウェア開発プロジェクトの成功率はほぼ同じ.

- **社会的損失の大きな26事例**(日経コンピュータ, 別冊特別編集版, 2009年 他)

- 三菱UFJのATM停止, 人事院システムの開発費倍増, PASMO運賃二重課金, . . .
- 13事例は, ユーザ・ベンダ間のコミュニケーションが良好であれば防げた可能性が高い.
- 医療における「インフォームドコンセント」のようなしくみがソフトウェア開発にもあれば. . .



# 国際競争力の喪失

- **日本主要ICTベンダの営業利益率**(総務省 情報通信白書, 2008年)
  - 日本国内で **6.4%**, **海外で 1.8%**  
(米国ベンダ 米国内で 15.6%, 海外で 13.7%)
  - **ユーザとの関係が確立されていない海外市場で苦戦.**
    - 日本企業の**オフショア開発能力レベル**の平均は,「オフショア活用方針・基準」,「計画・契約」,「実行管理」,「評価」すべてにおいて, **米国企業を下回る**. (JEITA 海外・国内企業におけるソフトウェアのオフショア開発についての調査・分析と提言, 2006年)
    - ソフトウェア性能に関してユーザとの事前合意がなされていない場合, 設計レビューにおける**バグの見逃しは3倍になる**.
  - **ユーザ・ベンダ間のコミュニケーションが良好でないと, 開発リスクは増大する.**

# 内閣官房 第2次情報セキュリティ基本計画

- 情報セキュリティ対策が埋め込まれた、**安心安全な機器の実現や利用者環境の提供**
- 「事故前提社会」への対応力強化
  - 冷静で迅速な対応、**説明責任の明確化**
  - **利用者にとっての安全・安心の確保**
  - ...

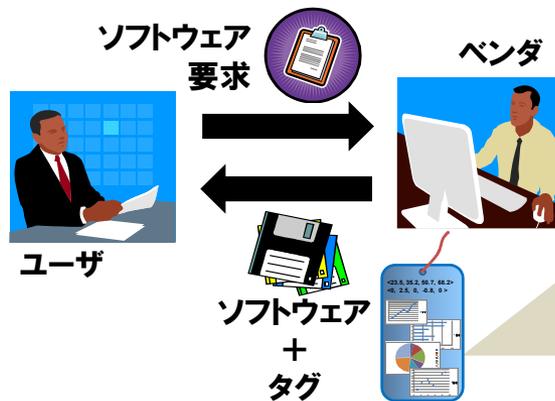


- 重要インフラ分野においては、**IT障害の発生は隠すべきものではなく、関係者間で共有すべき。**

StagEプロジェクトが実施したアンケート調査(対象者46,500名, 回答者302名)でも、「**ユーザの95%, ベンダの92%が, 開発データ共有の有用性を強く認識している**」との結果が得られている。

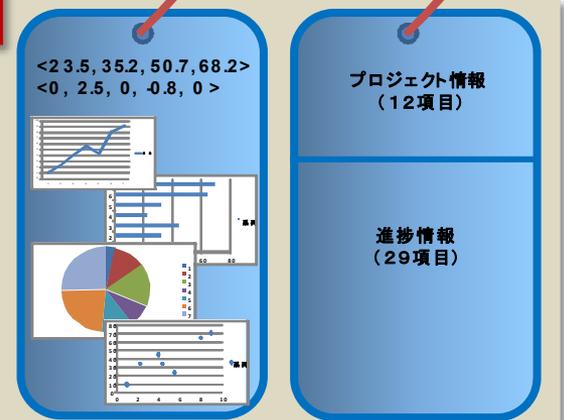
# 研究開発の目的

- ソフトウェア開発が適正な手順で行われたかどうかを表す実証データを「**ソフトウェアタグ**」としてソフトウェア製品に添付し、**ユーザ／ベンダ間等で共有**する技術を世界に先駆けて開発する。
  - ソフトウェアに対するトレーサビリティの概念を普及させる。
  - 世界最高水準の安心・安全なIT社会を実現する。



## ソフトウェアタグとは？

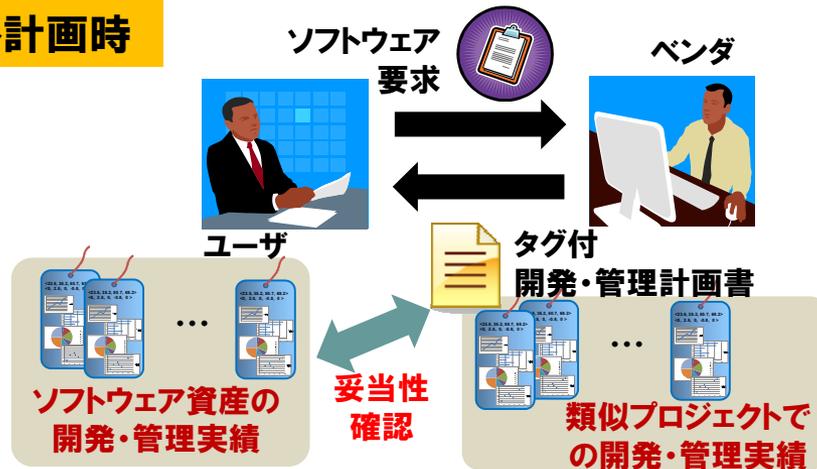
ソフトウェア開発組織のプロファイルや開発プロジェクトから収集した様々なデータを一定の形式で整理し、ソフトウェア製品に添付できるようにしたもの。



## プロジェクト概要:

# タグが実現する安心・安全なソフトウェア開発

### プロジェクト計画時



- ベンダもユーザも、「開発・管理計画」や「進捗報告」の**妥当性を確認**し、**開発リスク増大の原因を究明**することができる。
- ソフトウェア要求が、プロジェクトの進捗とともに、どのように**明確化**され、**実装**、**検証**されてゆくか、ユーザは**実感**できる。



- ソフトウェア開発における**説明責任**、**コンプライアンス**の**基盤技術の確立**

### プロジェクト実行時



研究開発活動:

# ① ソフトウェアタグ技術

研究開発項目(1-1)  
ソフトウェアタグの規格化  
井上克郎(大阪大学)

タグ規格

プロジェクトの総合的推進  
成果の積極的公表,  
産業界の現状・ニーズ把握

研究開発項目(1-2)  
ソフトウェアタグの収集技術  
楠本真二(大阪大学)

利用シナリオ

ツール

研究開発項目(2-2)  
ソフトウェアタグの実装と適用  
松本健一(奈良先端大)

運用・法的  
ガイドライン

研究開発項目(2-1)  
ソフトウェアタグに基づく  
可視化と評価技術  
飯田元(奈良先端大)

研究開発項目(3)  
ソフトウェア構築可視化に伴う法的諸問題  
久保浩三(奈良先端大)

# ソフトウェアタグの規格化

## ● ソフトウェアタグ規格第1.0版の開発

- 2008.10.14に**ソフトウェアタグ規格第1.0版**(プロジェクト情報12項目, 進捗情報29項目の合計41項目から構成される)を作成, 公開.
- 研究会等において, ベンダ・ユーザ企業は項目として**概ね妥当であると評価**.

ソフトウェアタグ規格第1版 (一部)

## ● 第2.0版への改訂方針

- **タグ利用シナリオ**の作成(適用ドメイン, 適用プロセス等毎に)
- 適用実験を通じた**タグ項目の評価**.
- **標準化**への取組.

分類	項番	タグ項目	具体化例	予定・実績の要否
要件定義	13	ユーザヒアリング情報	ユーザヒアリング実施件数(回) ユーザヒアリング項目数(件), ユーザヒアリング回答率(ユーザヒアリング回答数÷ユーザヒアリング項目数) など	○
	14	規模[推移]	画面、機能項目、ユースケース、アクター、顧客要件、機能、FPなど	○
	15	変更[推移]	規模の計測単位に依存	
設計	16	規模[推移]	機能設計(ページ数・帳票数・画面数・ファイル数・項目数・UML図の数、クラス数、パッチプログラム数、重要な機能数など) 構造設計(データ項目数、DFDデータ数、DFDプロセス数、DBテーブル数など) など	○
	17	変更[推移]	規模の計測単位に依存	
	18	要件の網羅率	設計に取り入れられた要件数÷要件数	

# ソフトウェア構築可視化に伴う法的諸問題

ソフトウェアタグ＝ソフトウェア開発の過程を**見える化**し、その情報をタグとしてユーザ、ベンダが**共有**するもの

紛争分析による法的な問題の分類	法的責任	紛争を避けるためのソフトウェアタグの意義	法的問題委員会の要請により今まで研究してきたこと	法的問題委員会の要請によりこれから研究すること
1. ソフトウェアが <b>完成しなかった</b> か又は完成したが欠陥があり使用していない場合	ユーザ又はベンダの債務不履行責任	<ul style="list-style-type: none"> <li>・ユーザに進捗を見せ、<b>相互に管理</b>を行う。</li> <li>・開発が<b>順調</b>に進んでいることをユーザに知らせる。</li> <li>・開発が<b>順調</b>に進んでいない場合の<b>軌道修正</b>を容易にする。</li> </ul>	<ul style="list-style-type: none"> <li>・ユーザの要望(<b>要求定義</b>)がベンダに伝わっているかどうかを明らかにすること。</li> <li>・実績が予定どおり進んでいること(<b>予実管理</b>)を行えるようにすること。</li> <li>・最初の異変(<b>ファーストクラッシュ</b>)を明らかにすること。</li> </ul>	<ul style="list-style-type: none"> <li>・実証実験により、<b>基準値、標準曲線</b>を作成し、ユーザは最低限それだけチェックしていれば、大丈夫という内容を定めること。(つまり人間ドックにおける<b>基準値を定める</b>こと。)</li> </ul>
2. ソフトウェアが <b>完成し</b> 使用しているが、バグ又は欠陥により、障害が発生した場合	ベンダの製造物責任・不法行為責任・債務不履行責任	<ul style="list-style-type: none"> <li>・事故があった場合の<b>トレース</b>を容易にする。</li> <li>・<b>早期復旧</b>を容易にする。</li> </ul>	<ul style="list-style-type: none"> <li>・システムがダウンする前に、どこに<b>バグ又は欠陥</b>があるかのおおよその<b>当たり</b>をつけること。</li> </ul>	<ul style="list-style-type: none"> <li>・システムがダウンした場合の原因、修正履歴をタグとしてソフトウェア開発に盛り込む手法を開発し、<b>障害の発生率を大きく下げる</b>ことができるようにすること。</li> </ul>

特色:

# 関連組織・プロジェクトとの比較

関連した研究を行っている組織・プロジェクト	本プロジェクト	
ISGSG (オーストラリア)	ベンチマーキングデータの提供。 データ収集ツールが提供されておらず、組織間でのデータの整合性に問題あり。	タグ規格準拠ツールを開発中。 <b>組織間でのデータ共有が目的。</b>
NASA Metrics Data Program (米国)	プログラムコードレベルのメトリクス、障害データ、要求に関するデータが蓄積、公開されている。 開発プロセスを把握できるデータは含まれていない。	プロジェクトのプロファイルや <b>進捗状況に関するデータも対象。</b>
Fraunhofer IESE (ドイツ)	ドイツ最大の研究機関の一研究所で、産学連携を通じたエンピリカルソフトウェア工学を推進している。 データや分析結果が広く公開されることはない。	研究開発の成果を、 <b>ソフトウェアタグ規格、ツール、利用シナリオ等として公開。</b>
HackyStat (米国), IPA/SEC, JUAS (日本), ...	ベンダ、もしくは、ユーザの視点でデータが収集、蓄積されている。	<b>ベンダ・ユーザ双方の視点での、ソフトウェア開発の見える化。法的観点からも議論。</b>
経済産業省 (日本)	情報システム・モデル取引・契約書などにより、ユーザ・ベンダ間でやり取りすべきドキュメントや手順を規定している。	ユーザ・ベンダ間でやり取りされるドキュメントの評価や組織間比較を可能に(左記とは <b>補完関係</b> )。

特色:

# 国の戦略的取組への波及

- 「高度情報化社会における情報システム・ソフトウェアの信頼性及びセキュリティに関する研究会の中間報告書(案)」(経済産業省 商務情報政策局情報処理振興課)において、本プロジェクトの重要性が言及されている。
  - 「複雑化する情報システム・ソフトウェアの信頼性及びセキュリティ水準を高度化するためには、ソフトウェアタグの高度化と現場への適用を検討する必要がある。」(82～83ページ)

<http://search.e-gov.go.jp/servlet/Public?CLASSNAME=Pcm1010&BID=595209006&OBJCD=&GROUP=>

特色:

# 機能安全性の保証

- **トレーサビリティの視点から、現実的な機能安全性を保証する仕組みとして、ソフトウェアタグが機能すると期待される。**
  - 組込みソフトウェアでは、機能安全性の保証問題が課題になりつつあり、IEC61508が制定されている。車載組込みソフトウェア向けにはISO26262の制定作業も進んでいる。
  - 急速に大規模化している組込みソフトウェアにおいては、1つの組織だけでの開発は困難になりつつあり、複数の開発組織の連携が必須となりつつある。

(日本が主導権を取って国際規格化した情報関連の提案はいくつか存在するが、日本で開発したアイデアを骨格としたソフトウェア規格を提案したものはない。)

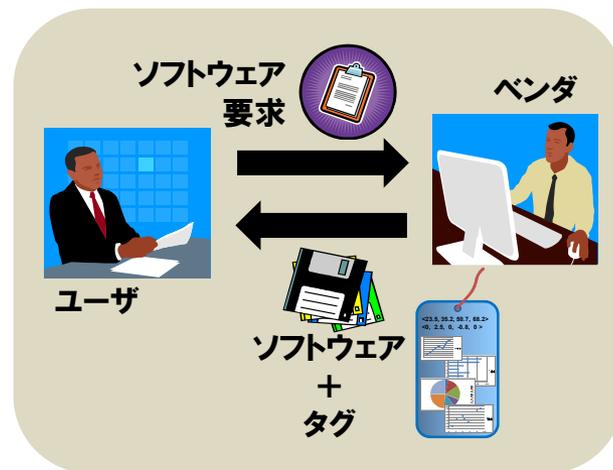
特色:

# 実用化・普及に向けて

- **ソフトウェアタグ規格やツールの標準化**
  - 多くのユーザが、開発データの共有の実用化・普及の方策として、「データやその共有方法の標準化」を挙げている。
    - ソフトウェアタグ規格: JIS化, ISO化
    - ツール: デファクトスタンダード
- **ソフトウェアタグ実証実験の推進**
  - 多くのベンダが、開発データの共有の実用化・普及の方策として、「ユーザへの開示のメリットの明確化」を挙げている。
  - ユーザ・ベンダニーズに基づくソフトウェアタグ利用シナリオ案を作成し、実際のソフトウェア開発プロジェクトに適用する。
- **国際連携の強化**
  - アジア・太平洋圏ソフトウェア工学研究ネットワークAPSERNを核とした情報交換と人的交流を進める。

# まとめ

- ソフトウェア開発全般. 特に, オフショアを含むマルチベンダ開発, 組込みソフトウェア開発を対象として, ソフトウェア構築状況を表す**実証データを「ソフトウェアタグ」としてソフトウェア製品に添付し, ユーザ／ベンダ間等で共有する技術**を世界に先駆けて開発する.
  - ソフトウェア開発における**トレーサビリティ**, 更には, **説明責任, コンプライアンスの基盤技術の確立.**
  - **タグ規格の国内／世界標準化.**
  - **タグ準拠ツール・利用シナリオの整備**
  - **法的視点での検討.**
  - **社会的損失の回避と国際競争力の強化に資する人材に育成.**



# タグ利用シナリオの概要

---

# タグ利用シナリオの位置付け

研究開発項目(1-1)  
ソフトウェアタグの規格化  
井上克郎(大阪大学)

タグ規格

プロジェクトの総合的推進  
成果の積極的公表,  
産業界の現状・ニーズ把握

研究開発項目(1-2)  
ソフトウェアタグの収集技術  
楠本真二(大阪大学)

利用シナリオ

ツール

研究開発項目(2-2)  
ソフトウェアタグの実装と適用  
松本健一(奈良先端大)

研究開発項目(2-1)  
ソフトウェアタグに基づく  
可視化と評価技術  
飯田元(奈良先端大)

運用・法的  
ガイドライン

研究開発項目(3)  
ソフトウェア構築可視化に伴う法的諸問題  
久保浩三(奈良先端大)

# ソフトウェアタグ利用時の課題

- ソフトウェアタグを用いたプロジェクト(プロセス/プロダクト)の評価方法が明確になっていない。
  - どのタグ項目に基づき、どうやって評価するのか。



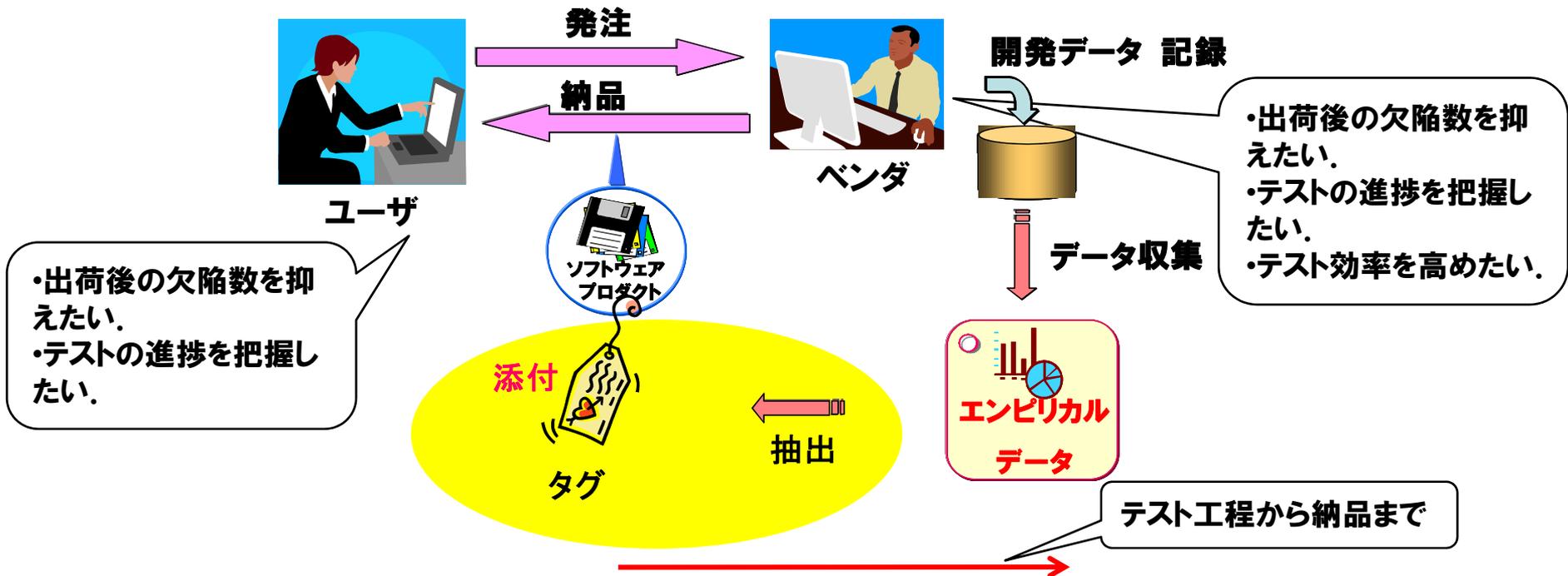
# タグ利用シナリオ

- **目的**
  - ソフトウェアタグを用いたプロジェクトの評価方法を示す。
- **要件**
  1. **ゴール(評価の目的)を明示するとともに, ゴールとタグ項目/評価方法の関係を明確にする。**
  2. **プロジェクト完了時だけでなく, 進行中にもプロジェクトを評価し, 早期に対処する方法を明確にする。**
- **アプローチ**
  1. **測定情報モデルとGQMパラダイムに基づくモデルを提案する。**
  2. **KGI(重要目標達成指標)とKPI(重要業績評価指標)の概念を導入し, 問題発生時の対応策を明示する。**

# タグ利用シナリオの構成要素

- **アウトライン**
  - シナリオの概要を示す.
- **管理指標(KGI/KPI), 解釈モデル, 分析モデル, 測定関数**
  - プロジェクトの評価方法, タグ項目の導出方法を示す.
- **ゴール**
  - ユーザ/ベンダのプロジェクトを評価する目的を示す.
- **利用タイミング**
- **開始時条件, 終了時条件**
- **対応策**

# 利用シナリオの例



企画プロセス			開発プロセス			運用プロセス		保守プロセス
システム化の方向性	システム化計画	要件定義	システム設計	ソフトウェア開発 プログラミング ソフトウェアテスト	システムテスト	運用テスト	運用	保守

# アウトライン(1/5)

- 「アウトライン」では、タグ利用シナリオの適用場面、ユーザ/ベンダの要求、タグを用いたプロジェクトの評価手順が記述される。

## 例

- 場面
  - ソフトウェア開発のテスト工程から納品時
- 要求
  - ユーザ, ベンダはソフトウェアの出荷後の欠陥数を抑えたい。
  - ユーザ, ベンダはテスト工程の進捗を把握したい。
  - ベンダはテスト効率を高めたい。

# アウトライン(2/5)

- **評価手順(テスト工程の進捗把握)**
  - **ベンダは、テスト工程において、実証データを毎週計測する。**
    - **テスト項目数の予実、発見欠陥数と修正欠陥数**
  - **ベンダは、収集したデータに基づき、以下の項目をソフトウェアタグに含める。**
    - **テスト項目消化率、欠陥消化率**
  - **ベンダは、ソフトウェアタグを毎週ユーザに渡す。**
  - **ユーザ/ベンダは、それぞれの項目の増加傾向を、過去のデータと比較し、進捗が順調かどうかを判断する。**
  - **テストに遅れが見られると判断された場合、ユーザ/ベンダはテスト計画の変更などの対応策を協議する。**

※ 説明のためのサンプルです。正式なタグ利用シナリオではありません<sup>28</sup>

# アウトライン(3/5)

- 評価手順(ソフトウェアの出荷後の欠陥数抑制)
  - ベンダは、単体テスト中に実証データを計測する。
    - 開発規模, 発見欠陥数, 実施テスト項目数, コードカバレッジ, 要件に対応するテストケース数, 要件の数,
  - ベンダは、収集したデータに基づき、以下の項目をソフトウェアタグに含める。
    - 欠陥密度, コードカバレッジ, 要件カバレッジ
  - ベンダは、「欠陥密度」が含まれたタグを、単体テスト終了時にユーザに渡す。
  - ユーザ/ベンダは、各項目が規定範囲内の場合、テストが適切であったと判断する。
  - ユーザ/ベンダは、信頼度成長曲線を作成し、グラフが収束している場合、テストが適切であったと判断する。

※ 説明のためのサンプルです。正式なタグ利用シナリオではありません<sup>29</sup>

# アウトライン(4/5)

- **評価手順(ソフトウェアの出荷後の欠陥数抑制) (続き)**
  - テストが適切でないと判断された場合, その原因を調査し, コードの再レビュー, テスト項目追加などの対応策を協議する.
  - 結合テスト, システムテストにおいて, ユーザ, ベンダは同様の作業を行う.
  - 出荷後の欠陥数を計測する.
  - 出荷6カ月後に, 出荷後欠陥密度を計算し, 規定値以下の場合, ユーザ/ベンダは欠陥数が抑制されたと判断する.

※ 説明のためのサンプルです. 正式なタグ利用シナリオではありません<sup>30</sup>

# アウトライン(5/5)

- 評価手順(テスト効率を高める)
  - ベンダは, テスト工程において実証データを毎週計測する.
    - 開発規模, テスト工数, 発見欠陥数, 実施テスト項目数
  - ベンダは, 収集したデータに基づき, 以下の項目をソフトウェアタグに含める.
    - 生産性, 欠陥指摘率
  - ベンダは, ソフトウェアタグを毎週ユーザに渡す.
  - ベンダは, 欠陥指摘率が規定範囲外の場合, テスト要員にヒアリングを行い, 必要に応じて指導を行う.
  - ベンダは, テスト完了後, 生産性を計算し, 規定値以上の場合, 効率が高まったと判断する.

※ 説明のためのサンプルです. 正式なタグ利用シナリオではありません<sup>31</sup>

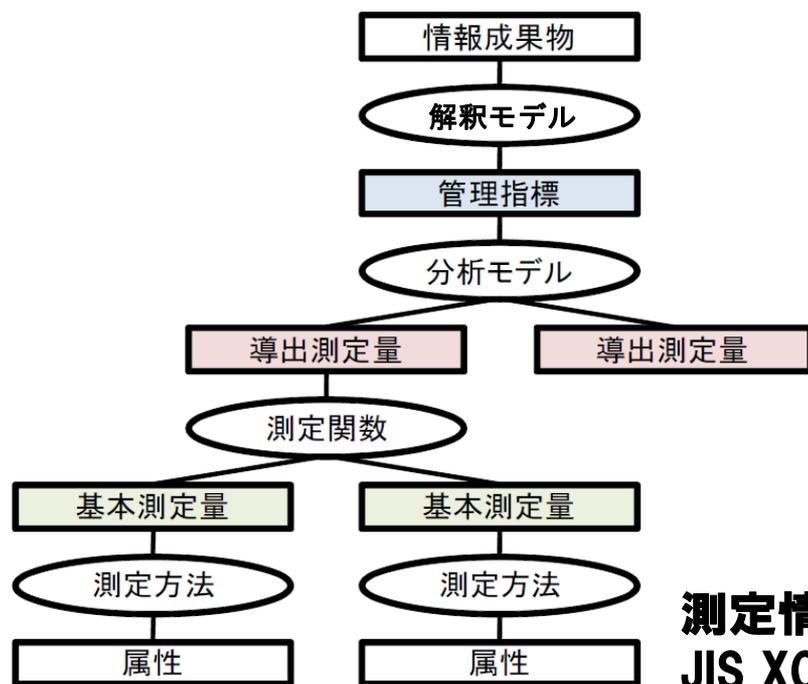
# ゴール, 管理指標

- **ゴール**
  - **ユーザゴール**
    - プロジェクトの評価を通じて, ユーザが達成したい目標.
  - **ベンダゴール**
    - プロジェクトの評価を通じて, ベンダが達成したい目標.
- **KGI(Key Goal Indicator; 重要目標達成指標)**
  - **ゴールを達成したか否かを判断するための定量的指標.**
- **KPI(Key Performance Indicator; 重要業績評価指標)**
  - **プロジェクト進行中にプロセスを評価するための定量的指標.**
  - **KPIが目標値をクリアするようにプロジェクトを遂行することにより, KGIも目標値をクリアできるような関係となる.**

# 管理指標, 解釈モデル, 測定関数(1/3)

## 測定情報モデル

- **測定結果**から, 意志決定の基礎となる**情報成果物**がどのような過程で得られるかを, 階層構造により表したモデル.



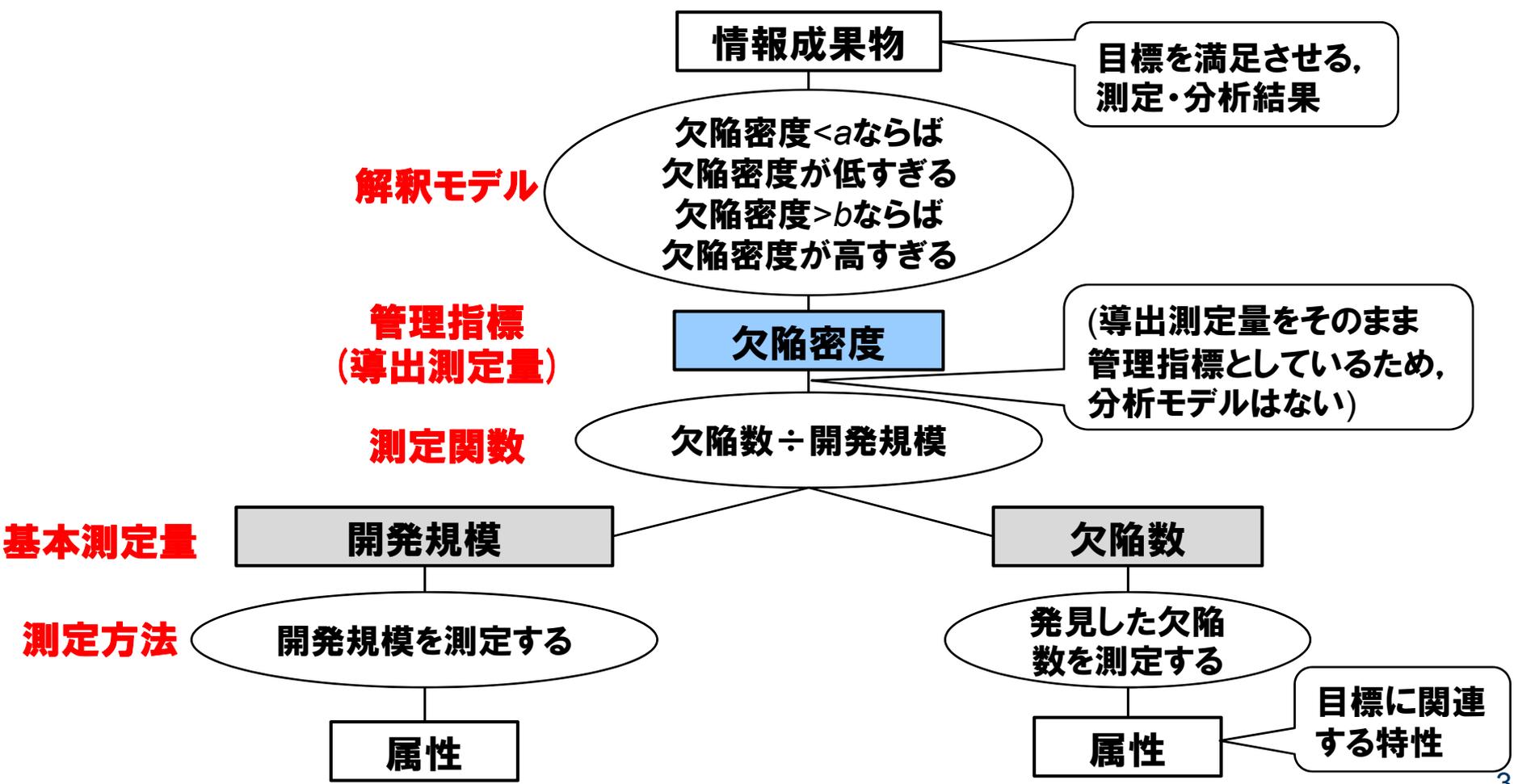
- **基本測定量**: プロジェクト中に存在する属性を, 定められた測定方法に基づいて定量化した数値.
- **導出測定量**: 測定関数に基本測定量を与えることにより求められる数値.
- **管理指標**: 分析モデルに導出測定量を与えることにより求められる数値.

### 測定情報モデル

JIS X0141:2004の図A.1を基に作成

# 管理指標, 解釈モデル, 測定関数(2/3)

## 測定情報モデルの例(欠陥密度)

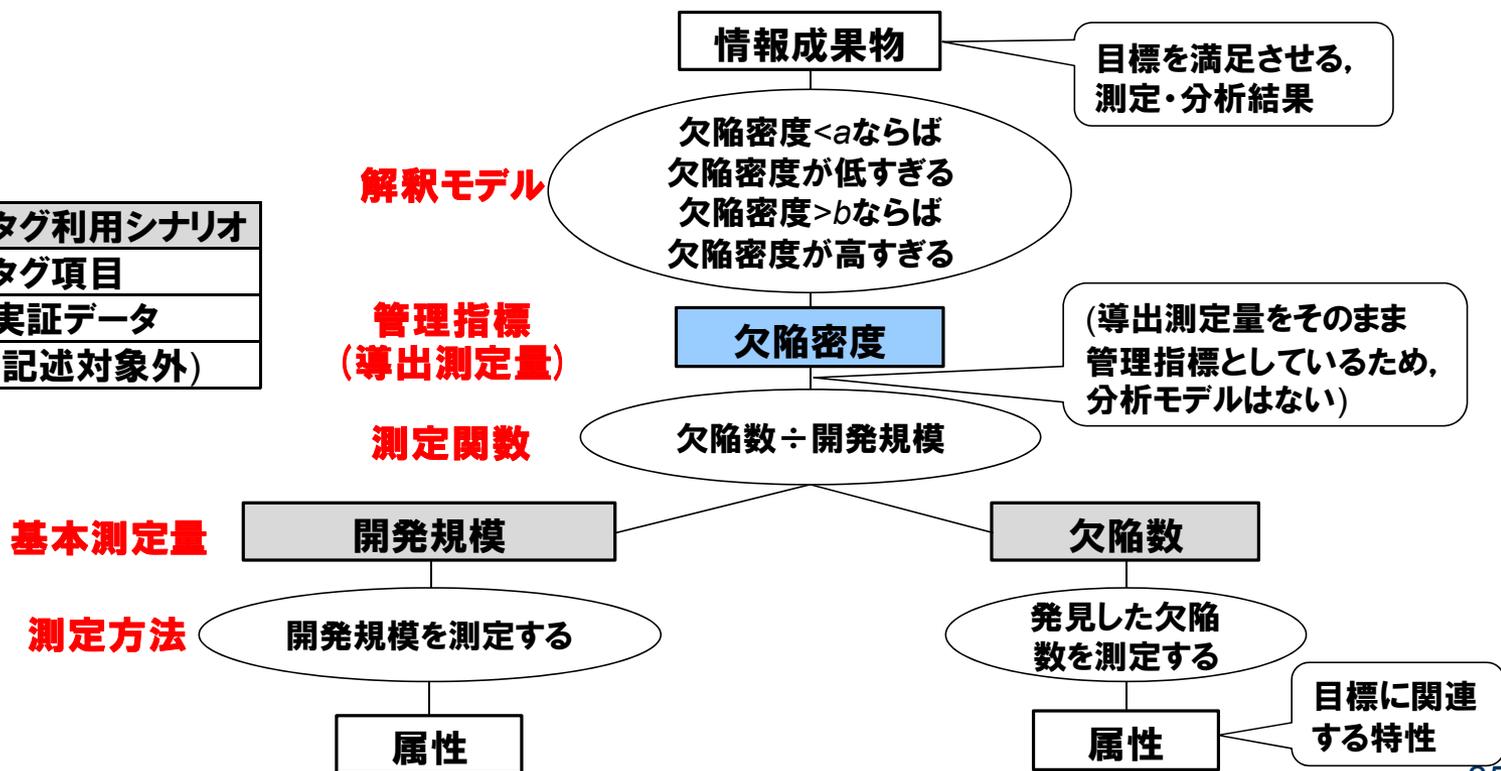


# 管理指標, 解釈モデル, 測定関数(3/3)

## 測定情報モデルに不足している概念

- 測定情報モデルでは, ゴール, KGI, KPIの概念が明確に存在しない.

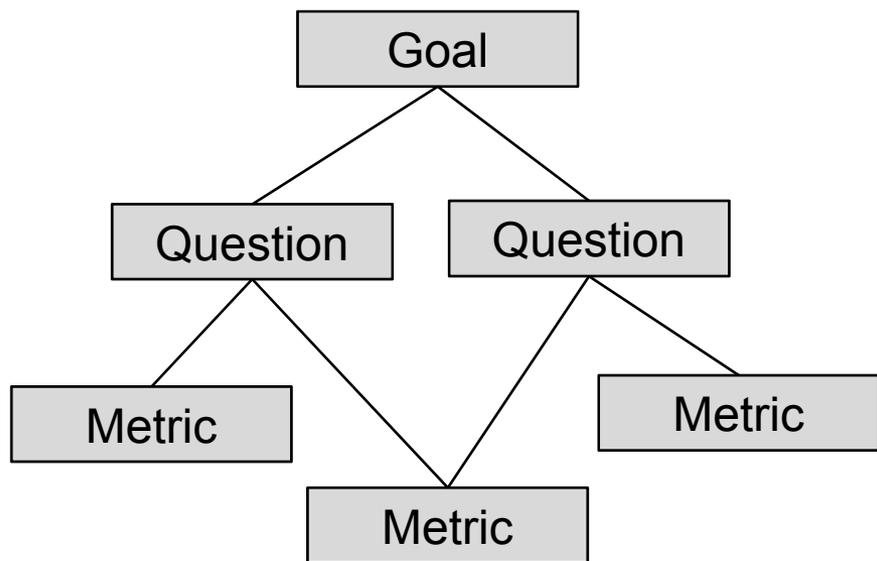
測定情報モデル	タグ利用シナリオ
導出測定量	タグ項目
基本測定量	実証データ
測定方法	(記述対象外)



# 測定情報モデルへのゴールの追加(1/2)

## GQMパラダイム

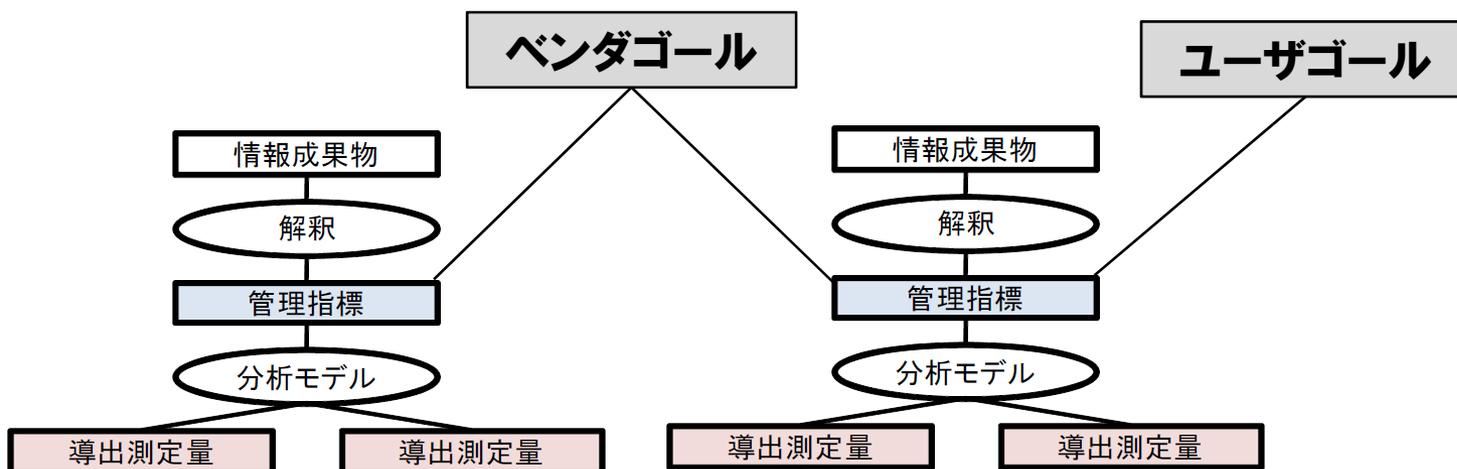
- **総合的なソフトウェア評価(計測)の枠組み.**
  - **評価の目標とメトリクスの対応関係を明確にすることができる.**



- **Goal: 計測の目標を, 計測対象, 計測理由などに基づいて明確にした文.**
- **Question: Goalを評価, あるいは達成する方法を明確にした文.**
- **Metric: Questionに定量的に答えるためのデータの集合.**

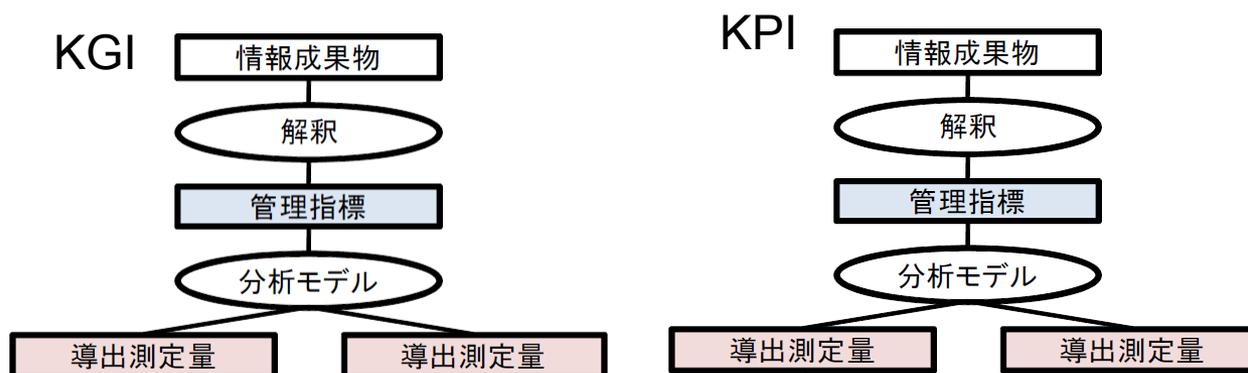
# 測定情報モデルへのゴールの追加(2/2)

- GQMパラダイムのゴールによって紐付けされた、測定情報モデルの管理指標の集合が、タグ利用シナリオの枠組みとなる。
- ゴールを**ユーザゴール**と**ベンダゴール**に分ける。
  - ユーザゴール、ベンダゴールはシナリオに複数存在してもよい。



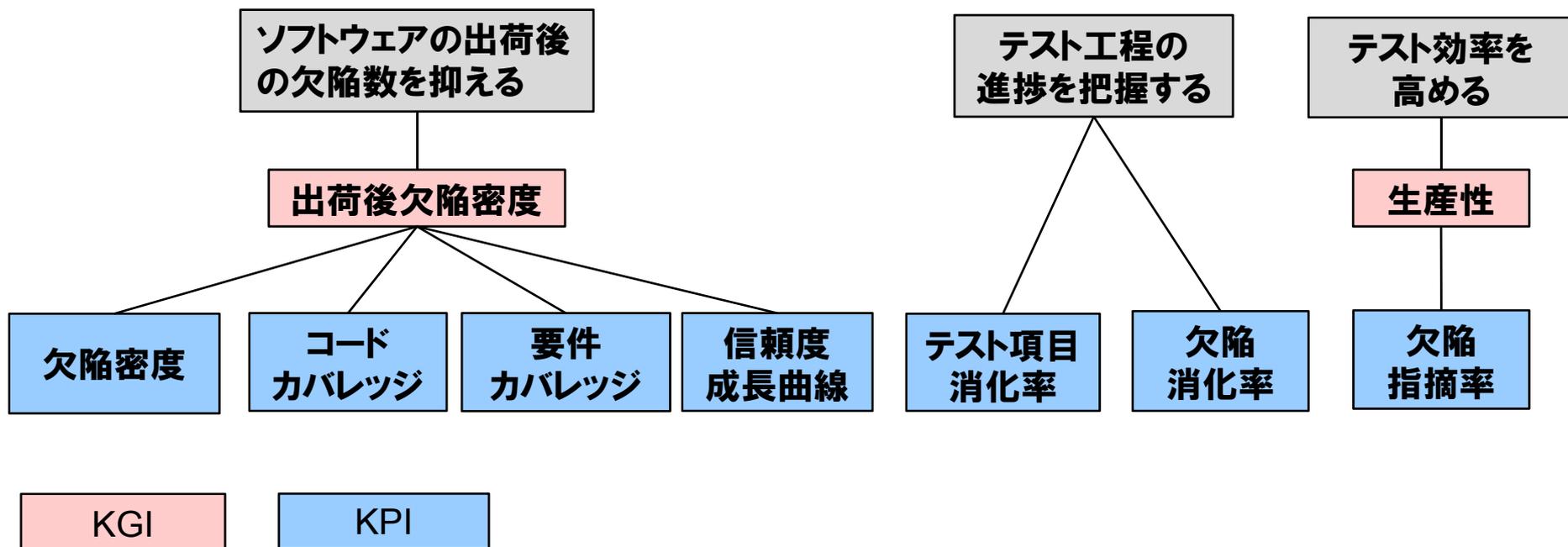
# 測定情報モデルへのKGI, KPIの追加

- 測定情報モデルに、ビジネスマネジメントの分野で用いられている、**KGI**, **KPI**の概念を導入する。
  - KGI: ゴールを達成したか否かを判断するための定量的指標。
  - KPI: プロジェクト進行中にプロセスを評価するための定量的指標。
- KGIとKPIに分類して、管理指標を記述する。



# ゴールとKGI, KPIの関係

- ベンダゴール, ユーザゴールごとにKGI, KPIを設定する.



※ 説明のためのサンプルです。正式なタグ利用シナリオではありません<sup>39</sup>

# 利用タイミング

- 実証データの収集, ソフトウェアタグの受け渡し, タグを用いた評価のタイミングを示す.

企画プロセス			開発プロセス				運用プロセス	保守プロセス
システム化の方向性	システム化計画	要件定義	システム設計	ソフトウェア開発 プログラミング ソフトウェアテスト	システムテスト	運用テスト	運用	保守

ベンダ

**実証データ収集, KPIに基づく評価**  
 欠陥密度, コードカバレッジ, 要件カバレッジ  
 信頼度成長曲線

**KGIに基づく評価**  
 出荷後欠陥密度

ユーザ

(単体, 結合, システムテスト終了時)  
**タグ受取, KPIに基づく評価,**  
**ベンダへの確認**  
 欠陥密度, コードカバレッジ  
 要件カバレッジ, 信頼度成長曲線

(出荷後6カ月)  
**KGIに基づく評価**  
 出荷後欠陥密度

※ 説明のためのサンプルです. 正式なタグ利用シナリオではありません

# 開始時条件, 終了時条件

- **開始時条件**

- タグ利用シナリオを適用するために、プロジェクトがあらかじめ満たしておくべき条件。
  - 例)要員のスキルが一定以上である, オフショア開発である, …

- **終了時条件**

- タグ利用シナリオ適用後に、プロジェクトの評価に誤りが含まれないことが保証される条件。
  - 例) 要件の頻繁な変更が発生しなかった, 納期の短縮が発生しなかった, …

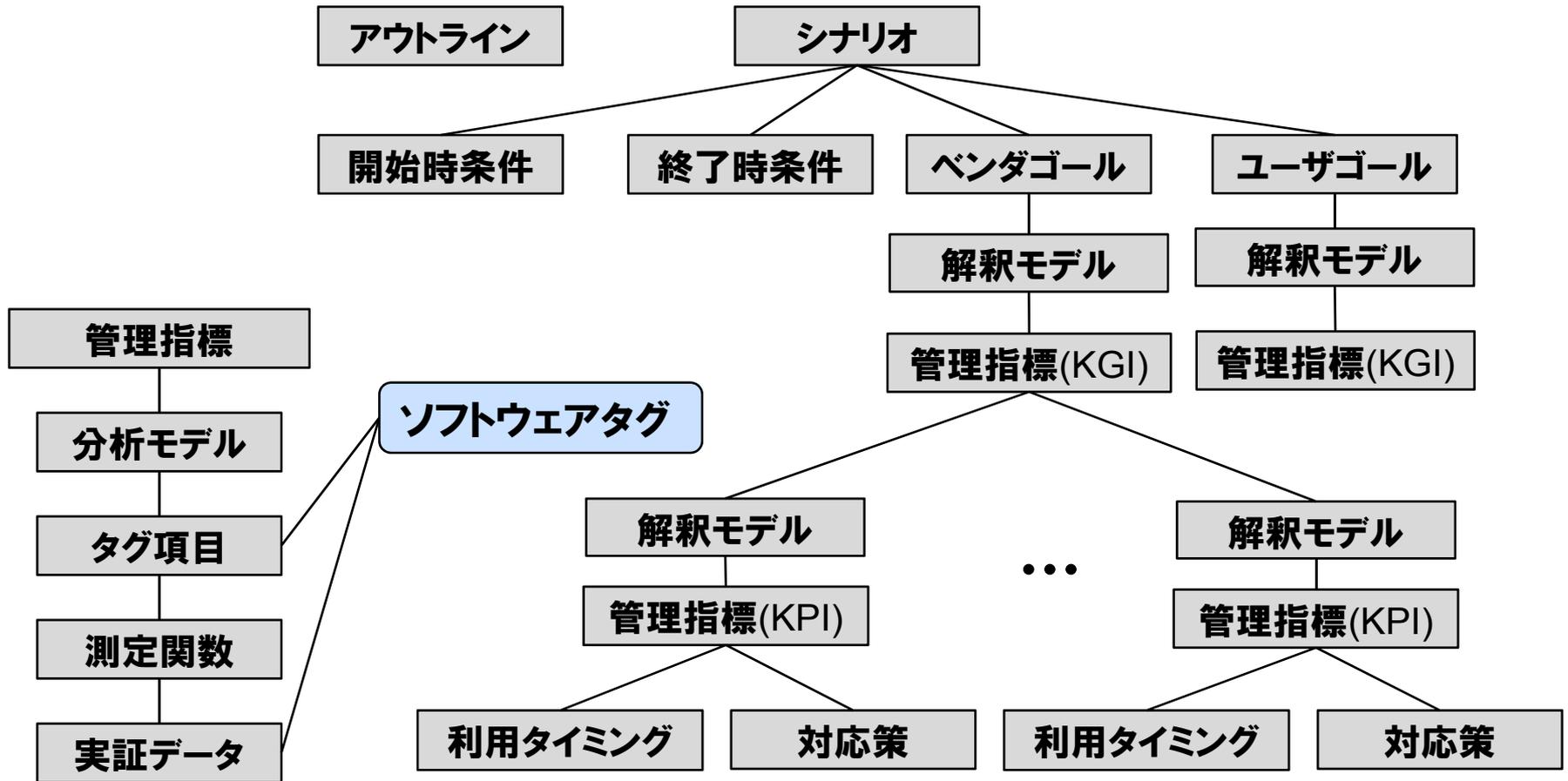
# 対応策

- プロジェクトの評価に基づいて、どのような対応を取ればよいのかを記述する。
  - 例) 欠陥密度が標準よりも高かった場合
    - 前工程(現在の工程が結合テストの場合, 単体テスト)でのテストが適切かどうか確認する.
    - コードが適切であるか, 再レビューを行う.
  - 例) 欠陥密度が標準よりも低かった場合
    - テストケースが適切であるかどうか確認する.
- 対応策は, KPIとKGIの組み合わせごとに記述する.

※ 説明のためのサンプルです。正式なタグ利用シナリオではありません

42

# タグ利用シナリオに含む情報の関係



# まとめと今後の予定

## ● まとめ

- ユーザ/ベンダが、ソフトウェアタグを用いて開発プロセスやプロダクトを評価する方法を説明した、タグ利用シナリオを定義した。
  - ゴール(評価の目的)を明示するとともに、ゴールとタグ項目/評価方法の関係を示す。
  - プロジェクト完了時だけでなく、進行中にもプロジェクトを評価し、早期に対処する方法を示す。
- タグ利用シナリオの構成要素
  - アウトライン, 管理指標(KGI/KPI), 解釈モデル, 分析モデル, 測定関数, ゴール(ユーザ/ベンダ), 利用タイミング, 開始時条件, 終了時条件, 対応策

## ● 今後の予定

- タグ利用シナリオ案を実プロジェクトに適用し、有効性を評価する。

# ソフトウェアタグ収集・分析のための ツール群

---

# ソフトウェアタグ収集・分析ツール群の 位置付け

研究開発項目(1-1)  
ソフトウェアタグの規格化  
井上克郎(大阪大学)

プロジェクトの総合的推進  
成果の積極的公表,  
産業界の現状・ニーズ把握

タグ規格

研究開発項目(1-2)  
ソフトウェアタグの収集技術  
楠本真二(大阪大学)

利用シナリオ

ツール

研究開発項目(2-2)  
ソフトウェアタグの実装と適用  
松本健一(奈良先端大)

研究開発項目(2-1)  
ソフトウェアタグに基づく  
可視化と評価技術  
飯田元(奈良先端大)

運用・法的  
ガイドライン

研究開発項目(3)  
ソフトウェア構築可視化に伴う法的諸問題  
久保浩三(奈良先端大)

# タグデータの収集・分析に関する課題

- ソフトウェアタグに記録するデータ(タグデータ)の収集は容易ではない
  - データの収集に多大なコストがかかる
  - 様々な種類や単位のデータを一元的に収集する必要がある
  - タグデータを用いてソフトウェア構築状況の分析・評価を行うために、分析しやすい形でデータを蓄積する必要がある
- タグデータをどのように分析・評価すれば良いかが明らかになっていない
  - ソフトウェアタグには膨大なデータが記録されており、俯瞰することだけでも難しい
  - どのような観点からデータを分析・評価していけば良いかわからない

# ソフトウェアタグ収集・分析ツール群の目的

- **ソフトウェアタグに記録するデータ(タグデータ)の収集・蓄積作業支援**
  - タグ規格に含まれているタグデータを収集する
  - 低コストでタグデータを収集する
  - タグデータ収集後に得られたソフトウェアタグを分析しやすい形で蓄積する
- **タグデータを利用した, ソフトウェア構築状況の分析・評価作業支援**
  - ソフトウェアタグに基づいたプロジェクトデータを提示する
  - タグデータをもとに開発プロセスを評価する

# ソフトウェアタグ収集・分析ツール群

- タグデータの**収集・蓄積を支援**するツール
  - AQUAMarine/TagPlanner: タグ利用計画立案支援システム
  - タグデータ収集システム
  - パーソナルタグデータ自動計測システム
- タグデータの**可視化・評価を支援**するツール
  - TagReplayer: プロジェクト再現システム
  - TagSimulator: プロジェクトシミュレーションシステム
  - Lap-MAP: プロジェクト階層内可視化フレームワーク

# ソフトウェアタグライフサイクルにおける ツール群の位置づけ

- ソフトウェアタグツール群は、タグデータの策定、収集、分析を支援する

SLCP-2007

	企画プロセス			開発プロセス			運用プロセス	保守プロセス	
モデル契約書 による工程	システム化の 方向性	システム 化計画	要件定義	システム 設計	ソフトウェア開発 プログラミング ソフトウェアテスト	システム テスト	運用テスト	運用	保守

## ソフトウェアタグ・ライフサイクル

ユーザ・ベンダ間の活動

契約前にタグとそのコスト等について協議

契約時にタグ条項を盛り込む

契約に従ってタグの納入・検査

最終タグの納品・検収

運用タグの納品・検収

保守タグの納品・検収

収集するタグ項目の策定を支援

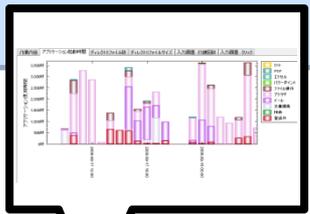
タグデータの収集を支援

タグデータに基づく分析・評価を支援

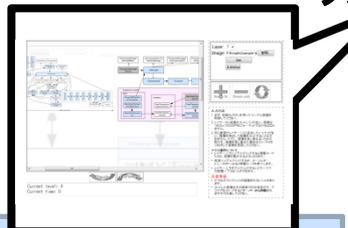
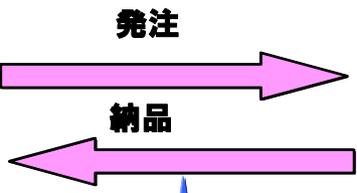
# ソフトウェアタグ収集分析ツール群の位置づけ

タグデータ収集システム  
 パーソナルタグデータ自動計測システム  
 タグデータの収集・蓄積を支援。

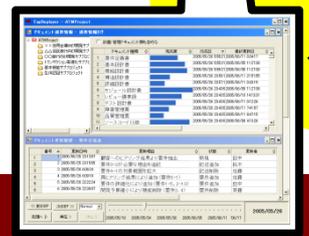
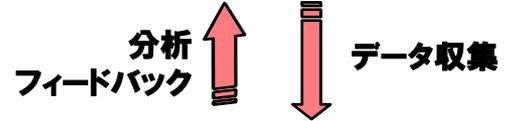
TagPlanner  
 プロジェクト管理計画を可視化。  
 収集するタグデータの策定を支援。



TagSimulator  
 プロジェクト進行をシミュレート・可視化。  
 プロジェクト進行の予測を支援。



LaP-MAP  
 プロジェクトの構造を可視化。  
 開発プロセス全体の俯瞰が可能。



TagReplayer  
 タグデータを解析してプロジェクト進行を可視化。  
 プロジェクト進行の確認を支援。

# タグデータ収集・蓄積ツール

- **タグ規格で定義されているデータを収集・蓄積する必要がある**
  - 収集されたデータは、分析・可視化ツールを用いて、ソフトウェアの構築状況把握に役立てる
  - 紛争時には、蓄積されたデータを利用して問題解決に当たる
- **タグ収集ツールには、以下の要件が求められる**
  - タグ規格に沿った形で、タグデータを収集する
  - 低コストでタグデータを収集する
  - タグデータ収集後に得られたソフトウェアタグを分析しやすい形で蓄積する
- **今回は、タグデータ収集を支援する2つのツールを紹介する**

# AQUAMarine / TagPlanner

- **タグデータの収集計画立案を支援するツール**
  - 対象プロジェクトで利用するタグデータの構造や定義作成を支援し、可視化して表示する
  - プロジェクトマネージャなどが、データ定義の閲覧者収集計画の作成、調整を行うことを容易にする
- AQUAMarine/TagPlannerで作成した収集計画は、ユーザー・ベンダ間でタグ収集項目を検討し、合意をとる際に用いることができる。

# AQUAMarine / TagPlanner の機能

- 開発管理計画のオーサリング・テーラリング支援機能
- その他の支援機能
  - 定量データと開発プロセスの関係を表すGUIの提供
  - 外部プロセス定義のインポートと検索機能の提供



# AQUAMarine / TagPlanner

プロジェクト全体の構造を表示

個々のプロセスや収集するタグデータの定義を修正可能

タグデータ項目を一覧表示

要素の... WBSのインポート 保存 別名で保存 IDの変更 削除

IS1 BSC01 DS1 SYP1 SYP2

概要  
ID: BSC01  
名称: 計画時点要件数  
説明:

定義  
尺度カテゴリ1: 数値  
尺度カテゴリ2: 比例尺度  
利用カテゴリ: プロダクトサイズ  
測定単位: 項目数  
測定タイプ: 見積  
対象成果物:

測定  
測定ツール:  
収集者:  
名前  
レビューア対象成果物作成者  
追加  
編集  
削除

測定方法:  
プロジェクト受注時、プロジェクト計画書のレビュー時、レビュー時に要件定義書から収集する。

ID	名称	尺度カテゴリ1	尺度カテゴリ2	利用カテゴリ	測定単位	測定タイプ	対象成果物	測定者
BSC01	計画時点要件数	数値	比例尺度	プロダクトサイズ	項目数	見積	成果物毎	レビューア対象...
BSC02	発生要件数	数値	比例尺度	プロダクトサイズ	項目数	実測	成果物毎	レビューア対象...
BSC03	確定要件数	数値	比例尺度	プロダクトサイズ	項目数	実測	成果物毎	レビューア対象...
BSC04	重要度別発生要件...	数値	比例尺度	プロダクトサイズ	項目数	実測	成果物毎	レビューア対象...
BSC05	重要度別確定要件...	数値	比例尺度	プロダクトサイズ	項目数	実測	成果物毎	レビューア対象...

管理指標一覧 導出測定量一覧 基本測定量一覧

# AQUAMarine / TagPlanner

個々の作業と、その作業で収集するタグデータを可視化して表示

The screenshot displays the AQUAMarine software interface. On the left is a project tree with folders like 'SYP1.1: システム要求仕様書の作成' and 'SYP2: システム・アーキテクチャ設計'. The main area shows a hierarchical view of tasks and associated tags (BSC, BTM). A right-hand pane shows a detailed view for 'BSC02: 発生要件数' with fields for '説明', '測定量分類', '利用分類', etc. A bottom table lists usage metrics for various IDs.

利用	ID	名称	関連導出測定量
<input checked="" type="checkbox"/>	IS1	未確定要件数の推移	DS1
<input checked="" type="checkbox"/>	IS2	未確定要件数の推移	DS2
<input checked="" type="checkbox"/>	IS3	未確定要件数の推移	DS3
<input checked="" type="checkbox"/>	IS4	未対応数の推移	DS4
<input checked="" type="checkbox"/>	IS5	未確定機能数の推移	DS5

タグデータに関する詳細な情報を表示

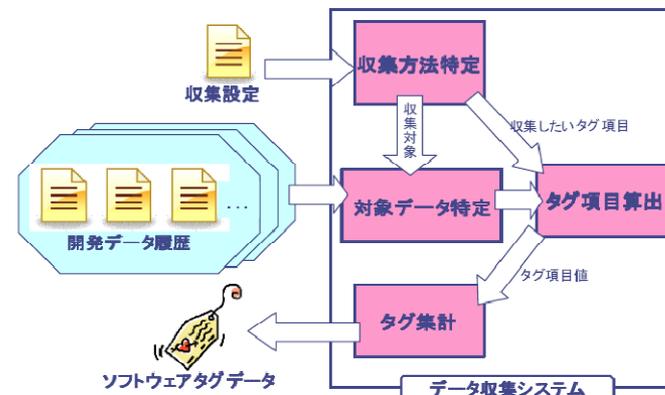
# タグデータ収集システム

- **タグデータ収集・計測システムの  
プロトタイプ実装と評価**

- **ソフトウェアタグ規格第1.0版準拠.**
- **品質・プロジェクト特徴の把握, 開発状況の分析における, タグ利用可能性を確認.**

- **システムの改良方針**

- **対応タグ項目の充実.**
- **タグ項目に対する基本メトリクスの設定とユーザビリティの向上.**



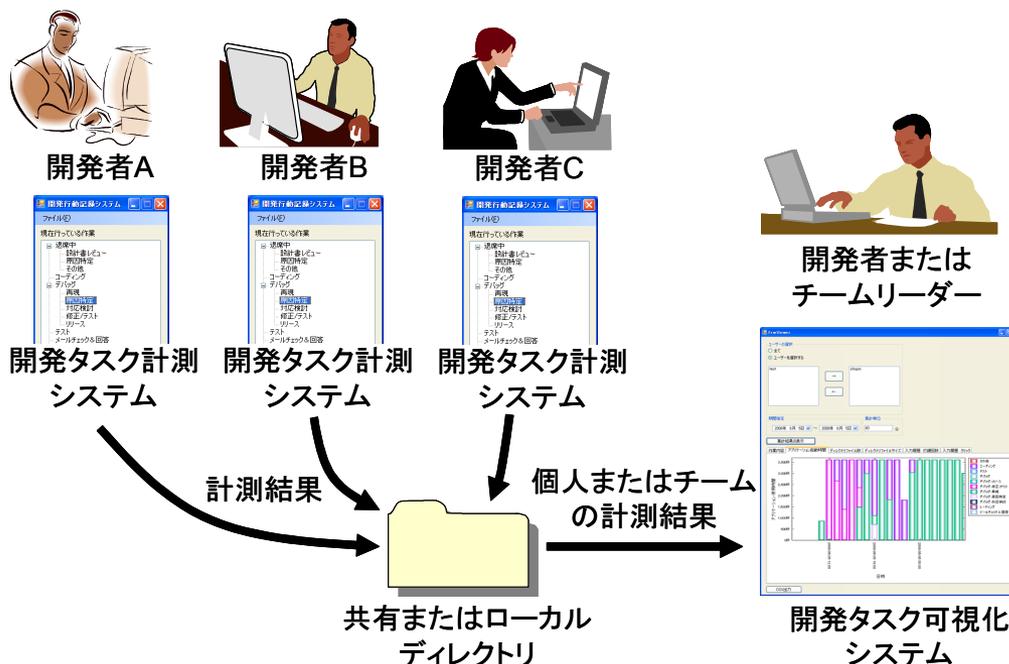
システム概要

プロトタイプで実装したタグ項目 (一部)

分類	項番	メトリクス(レベル1)	実測値
要件定義	13	ヒアリング回数	4
	14	ユースケース関数	12
	15	ユースケース関数	48
設計	16	UML関数	128
	17	UML関数	434
	18		
プログラミング	19	行数(全体)	26033
	20	変更量(追加+削除行数)	88841
	21	WMC	6.277551
		LCOM	10.955102
		NOC	0.7387755
		DIT	2.8081632
		CBO	10.43
	21	RFC	12.995918

# パーソナルタグデータ自動計測システム

- タグデータのうち、特に計算機を使って行う作業に関するタグの自動計測をおこなう
  - 設計書作成, プログラミング, テストなどの作業工数
  - 成果物の規模推移と変更量, 生産性

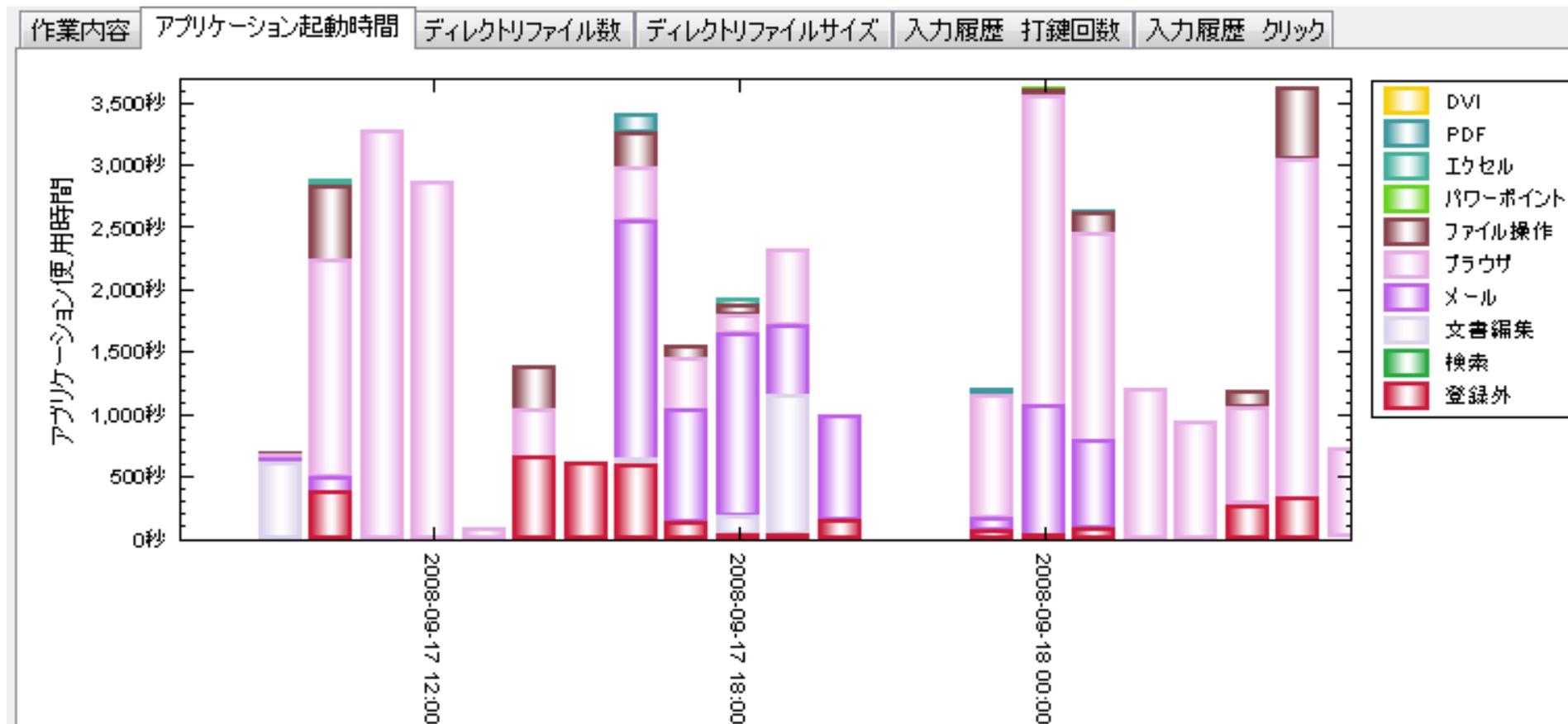


# パーソナルタグデータ自動計測システムの特徴

- 特定のアプリケーションに依存した計測を行わないため、多くの環境で利用可能
- 特定の開発プロセス、開発計画・予定に依存しない
  - タスクを特定のウィンドウ名を持つ1つ以上のアプリケーション上での作業とみなして、タスクごとの作業時間、打鍵回数、マウスクリック回数を計測する
- 計測データを可視化(グラフ化)する機能を提供する

# パーソナルタグデータ自動計測システム 動作例

## ● 可視化システム(1時間単位の表示)



# 既存ツールとの比較

- Subversion, CVSなどの版管理システムや, BugzillaやGNUTSなどの障害管理システムと比較して, タグに必要な情報を収集することができる
- EPM(エンピリカルプロジェクトモニター:ソフトウェア開発時の様々なデータを自動的に収集するシステム)と比較して, 必要なタグデータを選択し, 分析しやすい形で収集を行うことができる

# タグデータ可視化・分析ツール

- **目的**

- ソフトウェア発注者がソフトウェアタグに記録された情報からソフトウェア製品の由来を簡便に知る
- ソフトウェア製品やその開発プロセスの評価をおこなう

- **現在開発を進めている3つの可視化・分析ツール**

- TagReplayer: プロジェクト再現システム
- TagSimulator: プロジェクトシミュレーションシステム
- Lap-MAP: プロジェクト階層内可視化フレームワーク

# TagReplayer

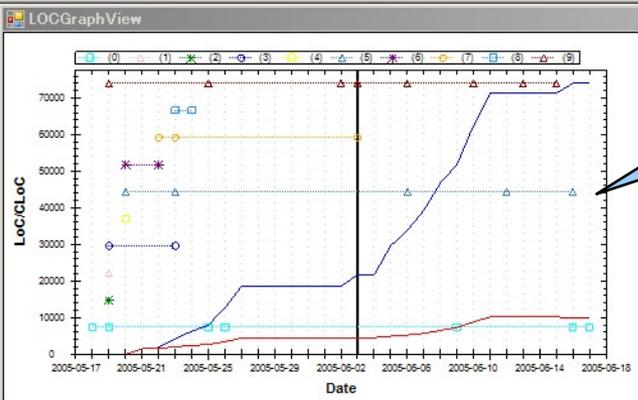
- **タグデータをもとに、プロジェクトを様々な表示形式で再現するツール**
  - “ある時点”でのプロジェクトの状態を、録画したビデオを再生するように提示する
  - プロジェクトのレビュー(事後分析)の際に活用できる
- **TagReplayerの機能**
  - プロジェクトにおいて発生したイベントを時系列に沿って網羅的に表示する
  - プロジェクトの時系列に沿った、各種メトリクスの変遷をグラフとしてプロットする
  - プロジェクトで発生したイベントを詳細に分析するために、開発者間の対話や障害報告などのテキスト情報をマイニングし、表示する

# TadReplayer

どのようなイベント  
が発生したか

表示

番号	操作	日時	詳細	オーナー
1	Select	2005/06/03 01:11	Mail Cluster--	Kyohei FUJII
2	Select	2005/06/03 01:11	Mail--	kyohei-f@i
3	Select	2005/06/03 9:00	Mail Cluster--	滝進也 <st
4	Select	2005/06/03 9:00	Mail--	滝進也
5	Select	2005/06/03 13:13	FrameworkTest/GameSceneTyping.cpp	akihir-t
6	Select	2005/06/03 13:13	FrameworkTest	akihir-t
7	Select	2005/06/03 13:13	FrameworkTest/GameSceneClick.cpp	akihir-t
8	Select	2005/06/03 14:14	FrameworkTest	shinva-t



プロジェクト全体の  
概要をグラフ化

- (0) 起動中村実験プロジェクト表示
- (1) テスト送信メール
- (2) 直し書ききた
- (3) メール駄目コミットマン
- (4) 中村利用方法カスタマイズ対処
- (5) 対処利用方法カスタマイズ中村岩村
- (6) ファイル成果注意
- (7) ミーティング部分明日コマワーク
- (8) 明日ミーティングコマ田中
- (9) ミーティング亀井明日クリックキックオフ
- (10) 取得
- (11) リソース上司顧客アニメーションコマ
- (12) リポードワーク明日
- (13) 統括ミーティング開催キャンセル田中

この人が  
何をしたか

MemberView

伏田享平

日時	イベント
2005/06/03 0:16:31	MAIL_CLUSTER [opencampus:134] Re: ミーティングについて-> 開
2005/05/27 11:07:42	MAIL [opencampus:119] opencampus2005-commit
2005/05/27 11:01:46	CVS MODIFY FrameworkTest/TestScene.cpp
2005/05/27 11:01:46	CVS MODIFY FrameworkTest/EasyDM.cpp
2005/05/27 11:01:46	CVS MODIFY FrameworkTest/GameScene.cpp

亀井裕高

日時	イベント
2005/06/02 21:21:51	MAIL [opencampus:132] ミーティングについて -> 開発者
2005/06/02 17:11:25	MAIL_CLUSTER [opencampus:132] ミーティングについて -> 開
2005/05/27 12:02:42	MAIL [opencampus:125] opencampus2005-commit
2005/05/27 11:56:28	CVS MODIFY FrameworkTest/BackGroundList.h
2005/05/27 11:56:28	CVS MODIFY FrameworkTest/BackGroundList.cpp

どのようなメール  
がやりとりされて  
いるか

BugView

件名	起票日	状態	報告者
1 背景を管理するためのクラスを作成	2005/05/25 23:13:57	open	亀井
2 読み込んだ画像の左右反転機能が欲しい	2005/05/27 0:24:52	open	後藤
3 マウスのキーイベントを取得できない	2005/05/25 23:15:55	closed	yasa
4 フォントのオブジェクトがない	2005/05/26 6:06:24	open	Kyo
5 テクスチャにも透過機能をつける	2005/05/26 22:22:24	open	後藤
6 テクスチャオブジェクトへのアニメーシ	2005/05/26 6:09:18	closed	Kyo
7 GetAnimeNumを追加する	2005/05/26 22:30:07	closed	後藤

プログラムの大きさ  
がどれくらい  
変化しているか

FileProgressView

ファイル名	LoC	完成度	Last Modify
1 cvsignore	11/23	<div style="width: 100%;"></div>	2005/05/21 10:01
BackGroundList.h	160/202	<div style="width: 80%;"></div>	2005/05/27 18:00
EasyDM.cpp	86/92	<div style="width: 90%;"></div>	2005/06/03 17:20
GameScene.cpp	0/50	<div style="width: 0%;"></div>	
GameSceneClick.cpp	0/22	<div style="width: 0%;"></div>	
GameSceneTyping.cpp	0/36	<div style="width: 0%;"></div>	
TestScene.cpp	0/48	<div style="width: 0%;"></div>	
TestScene.h	0/216	<div style="width: 0%;"></div>	

任意の時間に  
巻き戻すための  
タイムバー

StagEプロ

再生コントロール: 再生 > 停止 ||

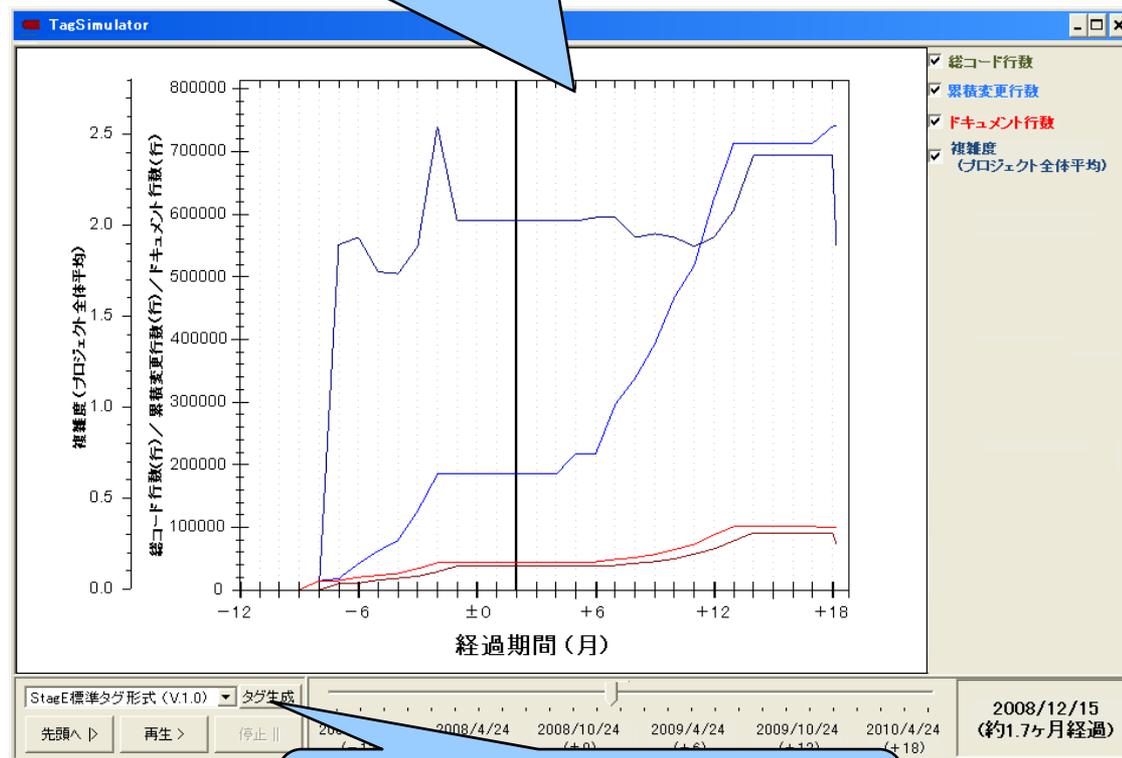
タイムライン: 2005/05/18, 2005/05/24, 2005/05/30, 2005/06/05, 2005/06/11, 2005/06/17, 2005/06/03

# TagSimulator

- **タグデータをもとにプロジェクトの動向をシミュレーションするツール**
  - プロジェクト進行時に、ある時点までの進捗から今後のプロジェクトの動向を予測する
  - 予測結果を検討し、必要であれば改善策を講じる
- **TagSimulatorの目指す機能**
  - プロジェクト計画時に、PMがプロジェクト完了時に目標となるタグデータが得られるよう、計画に関する様々な意志決定を支援する
  - ある時点でのタグデータをもとに、今後生成されるコードやドキュメント行数の推移をシミュレートし、発注者によるプロジェクト進行の予測を支援する

# TagSimulator イメージ図

将来のプロジェクト進行を可視化



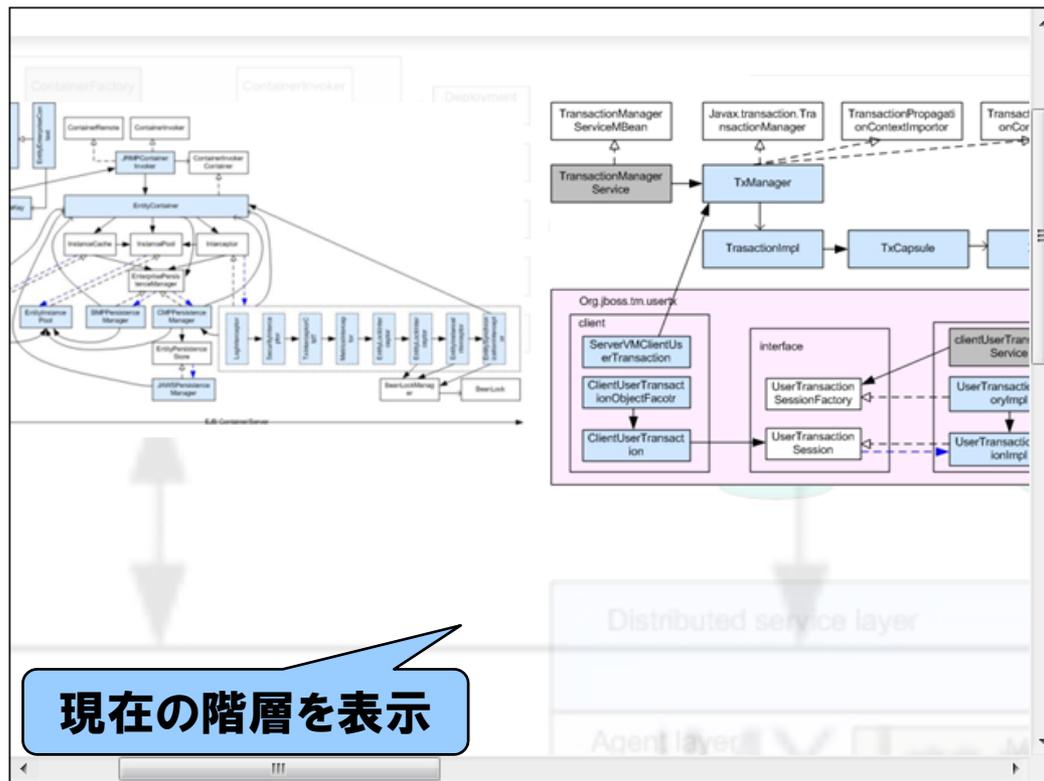
シミュレーション結果をソフトウェアタグ形式で出力

※ 画面は検討中のツールの完成イメージです.

# LaP-MAP

- **ソフトウェア開発プロジェクトの階層を考慮した可視化ツール**
  - マルチベンダ開発やオフショア開発では、複数の開発組織が階層的に関わるようなプロジェクトでのソフトウェアタグ情報の可視化を行う
- **Lap-MAPの特徴**
  - プロジェクト階層間をインタラクティブに移動することができる
  - 時系列に沿って、プロジェクト構造を閲覧することができる
  - 開発プロセス全体の俯瞰と、各組織ごとの情報の精査を素早く容易に行うことができる

# タグデータ可視化・分析ツール: LaP-MAP



現在の階層を表示



Current level: 0  
Current time: 0

時間軸移動ボタン

Layer: 1 ▾  
Image: F:\tmp\n2sample-jt 参照...  
Set  
Initialize

+ Zoom In   - Zoom out   ↻ Reload

入力方法

階層移動ボタン

ません。  
3. 同じ要領でレイヤー1,2と追加していただきます。階層を飛ばして画像をセットすることはできません。ただし、画像を差し替えることは可能です。画像を差し替えた場合はリロードボタンを押して画像を更新してください。

マウス操作について  
 • レイヤー上でシングルクリックすると移動モードになり、画像を動かせるようになります。  
 • 再度シングルクリックするか、カーソルがビューの外へ出ると移動モードを終了します。  
 • レイヤー上でダブルクリックすることで一つの階層へ下りることができます。

注意事項

- IEではダブルクリックが認識されないことがあります。
- セットした画像はその画面でのみ有効です。ブラウザをリロードするとサーバーから削除されますのでご注意ください。

# まとめ

## ソフトウェアタグの利用を支援するツールを紹介した

- **ソフトウェアタグに記録するデータ(タグデータ)の収集・蓄積  
作業支援ツール**
  - タグデータ収集システム
  - パーソナルタグデータ自動計測システム
- **タグデータを利用した, ソフトウェア構築状況の分析・評価  
作業支援**
  - AQUAMarine/TagPlanner: タグ利用計画立案支援システム
  - TagReplayer: プロジェクト再現システム
  - TagSimulator: プロジェクトシミュレーションシステム
  - Lap-MAP: プロジェクト階層内可視化フレームワーク