



Title	開発透明性や追跡性の向上を目指したソフトウェアタグの提案
Author(s)	井上, 克郎
Citation	
Version Type	VoR
URL	<a href="https://hdl.handle.net/11094/51432">https://hdl.handle.net/11094/51432</a>
rights	
Note	

*The University of Osaka Institutional Knowledge Archive : OUKA*

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

# Using Software Tag for Traceability and Transparency in Global Software Engineering

開発透明性や追跡性の向上を目指したソフトウェアタグの提案

Katsuro Inoue

井上克郎

Osaka University

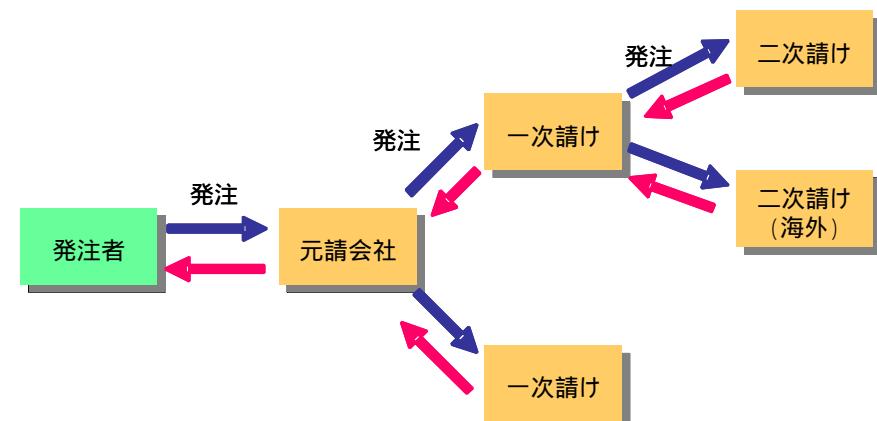
大阪大学

# ソフトウェアの重要性と開発の複雑化

- ソフトウェア品質に起因する問題の重大化と大規模化
  - 重大な社会インフラが停止する
    - 銀行や通信システム
    - ユーザ・ベンダに莫大な経済的損失を与える
    - 人命に関する危険を引き起こす
      - 航空管制システム, 自動車安全制御システム
- ソフトウェアの大規模化と開発期間の短縮
  - コストの低減や生産性の向上が要求される

# 開発情報の不透明化

- 開発体制の複雑化
  - 受注ベンダから2次請け, 3次請けという多重下請け構造
  - オフショア開発(海外へのアウトソース)の拡大
- ソフトウェア部品の流用・再利用の拡大
  - COTS(Commercial off-the-shelf), OSS(Open Source Software)など



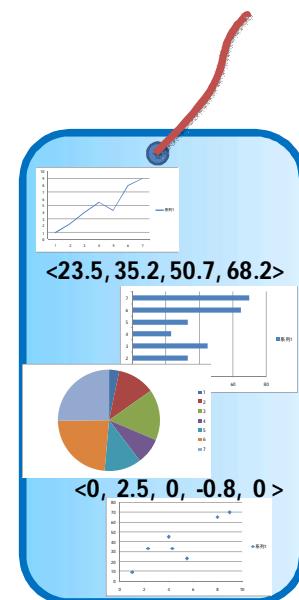
3

# 不透明化に伴う問題

- 発注者が納品物の品質を検証できない
  - 開発者任せになっている
- ユーザは要望に合ったソフトウェア商品を選択することができない
  - 機能と値段などの情報は入手できるが、信頼性や保守性などについては検証する手段がない
- 問題が発生した場合、原因や責任の所在を突き止めることが困難である
  - 迅速な障害対応が困難
  - 法的な係争に発展した場合、長期化する

# ソフトウェアタグ

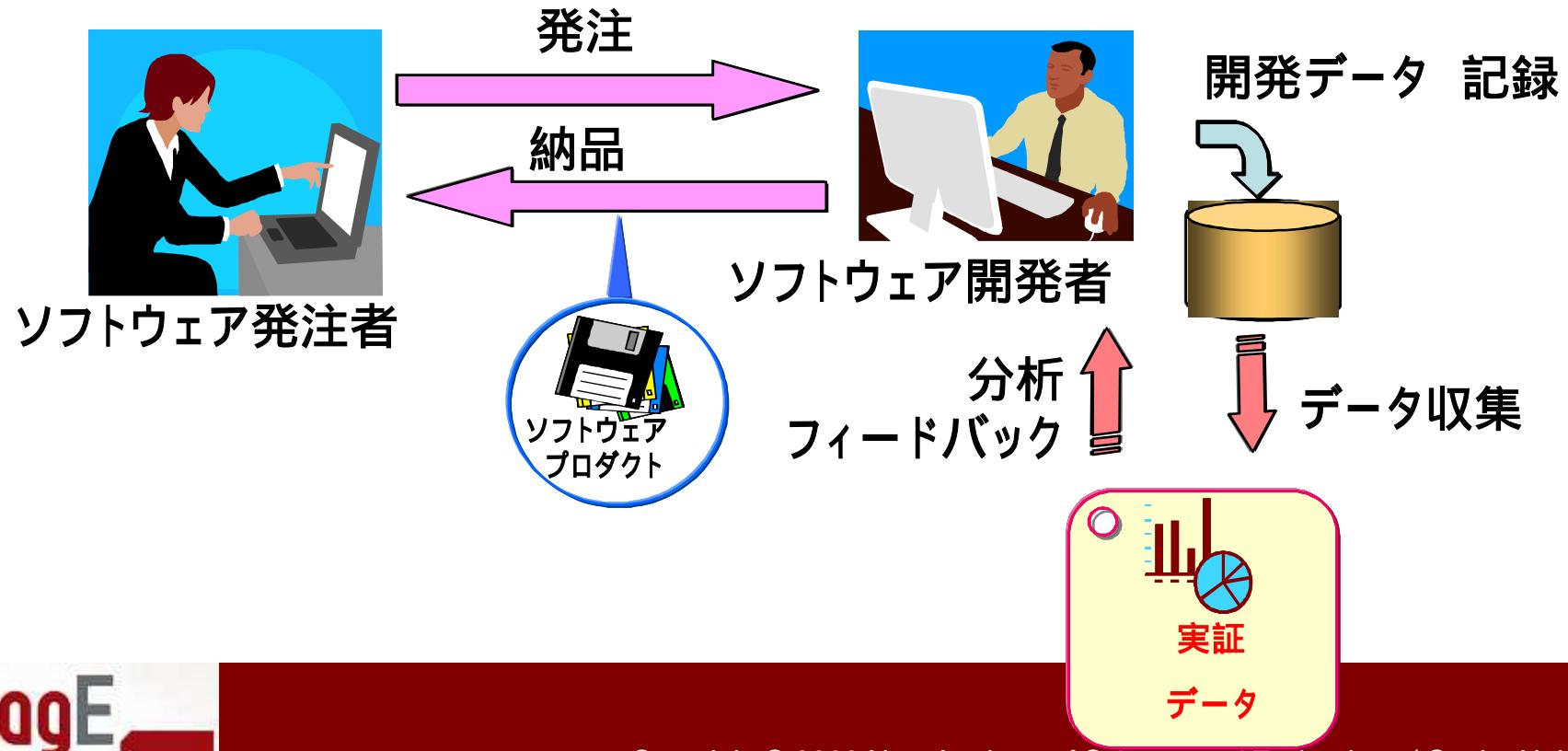
- ユーザが納品された、あるいは購入・流用したソフトウェアを安心して安全に用いるために、ソフトウェアの開発プロセスや成果物に関する情報を共有する仕組み
  - 開発時に得られる種々の実証データをユーザに提供
  - 2者間の取引で発注者のみに開示(一般公開なし)
  - 目標タグ、途中タグ、最終タグ
- 期待される効果
  - 発注者によるソフトウェアの品質の検証
  - ユーザによる適正なソフトウェア製品の選択の促進
  - 問題発生時の対応の迅速化
  - 透明性の拡大による法的問題の発生の予防と早期の公正な解決の促進



5

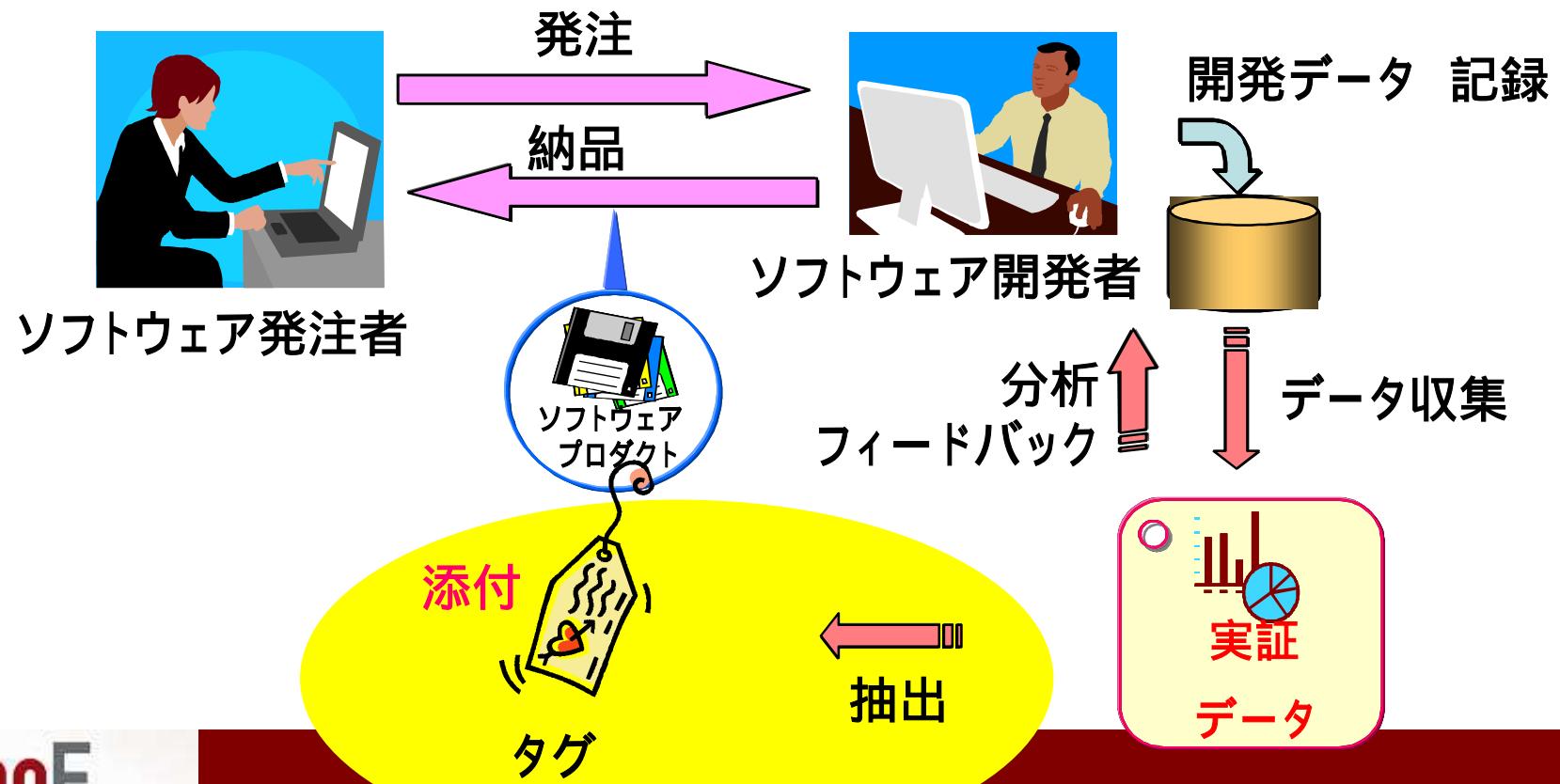
# 開発現場におけるデータ計測

- 実証データの収集
  - 品質管理, 進捗管理, リスク管理, コスト管理



# ソフトウェアタグと開発プロセス

- 実証データを選択、抽象化してタグにする



# タグ項目

- **Project 情報**(12項目)  
開発プロジェクト及びシステムの基本情報
- **Progress 情報**(29項目)  
開発プロセス(進捗、軌跡)情報



# プロジェクト情報の例

分類	項目名	説明	具体化例	実証データ例
基本情報	1 プロジェクト名	プロジェクトを一意に決定するための識別名		プロジェクト計画書など
	2 開発組織の情報	当該プロジェクトの開発を担当する組織の情報。一般には、受注者となる組織情報となる。	開発組織名	発注仕様書
			ISO15504(SPICE), CMMIなどのアセスメントモデルによる認証レベル	CMMI認定情報 体制表 キャリアシートなど
			対象業種の経験	
			...	
	3 開発プロジェクト情報	開発プロジェクトの特徴や当該タグデータの対象とするプロジェクトの種類を示す情報。タグデータの解釈や分析時に必要なデータ。	開発プロジェクト種別:新規・改造・保守・運用・流用など	要件定義書 プロジェクト計画書
			開発プロジェクト形態:商用パッケージ、受託開発など	品質計画書 設計ドキュメント
			利用パッケージ	ソースコード テスト計画書など
			設計書再利用率・ソースコード再利用率・コンポーネント再利用率など	
			...	
	4 顧客情報	当該システムのユーザ、もしくは第1発注者となる組織の情報。	顧客名 新規顧客か否か ...	顧客との議事録 プロジェクト計画書など
システム情報	5 システム構成	開発システム構成の特徴や当該タグデータの対象とするシステムの種類を示す情報。タグデータの解釈や分析時に必要なデータ。	利用したサブシステムの安定度:利用ハードウェア、ライブラリ、OS等、調達したサブシステムの安定度合い(初回、不安定等)	基本設計書 プロジェクト計画書 工程管理表 勤務実績データなど
			サブシステムの検証期間/工数:利用を検討したサブシステムの検証期間と工数	
			...	
	6 システム規模	開発システムの規模、計画値と最終実績値とする。進捗情報に同じ情報が含まれる場合は、省略可。	受注者側の想定する顧客要件の数、実装した要件数など	基本設計書 ソースコード 要件定義書など
			工程終了時のソースコード行数、機能、FP、処理データ量、処理データ数など	
			...	



# プロジェクト情報の構成

- 基本情報 (4)
  - プロジェクト名、開発組織の情報、開発プロジェクト情報、顧客情報
- システム情報 (2)
  - システム構成、システム規模
- 開発情報 (3)
  - 開発手法、開発体制、プロジェクト期間
- プロジェクトの階層構造情報 (2)
  - 親プロジェクト情報、子プロジェクト情報
- その他 (1)
  - 特記事項

# 進捗情報の例

分類	項目番	項目名	説明	具体化例	実証データ例
要件定義	1	ユーザヒアリング情報	要件に関してユーザに行ったヒアリングに関する情報	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 .....	ユーザヒアリング議事録 ユーザヒアリング質問票など
	2	規模[推移]	開発側で作成した要件数	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	要件定義書など
	3	変更[推移]	変更された要件数	規模の計測単位に依存	要件定義書 要件定義書の変更履歴など
設計	4	規模[推移]	設計成果物の規模 新規・改造・再利用(流用)毎に計測する	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 .....	基本設計書 機能設計書 構造設計書 詳細設計書など
	5	変更[推移]	変更された設計成果物の数、もしくは変更量	規模の計測単位に依存	基本設計書 機能設計書 構造設計書 詳細設計書 各設計書の変更履歴など
	6	要件の網羅率	要件定義で作成された要件の実装率	設計に取り入れられた要件数 ÷ 要件数	要件定義書 基本設計書 機能設計書 構造設計書 詳細設計書など



# 進捗情報の構成

- 要件定義 (3)
  - ユーザヒアリング情報、規模、変更
- 設計 (3)
  - 規模、変更、要件の網羅率
- プログラミング (3)
  - 規模、変更、複雑度
- テスト (4)
  - 規模、変更、密度、消化
- 品質 (8)
  - レビュー状況、レビュー作業密度、レビュー指摘率、欠陥件数、欠陥対応件数、欠陥密度、欠陥指摘率、静的チェックの結果
- 工数 (2)
  - 作業工数、生産性
- 計画・管理 (4)
  - プロセス管理情報、会議実施状況、累積リスク項目数、リスク項目滞留時間
- その他の成果物 (2)
  - 規模、変更

# 利用モデル

---

## タグの利用モデル

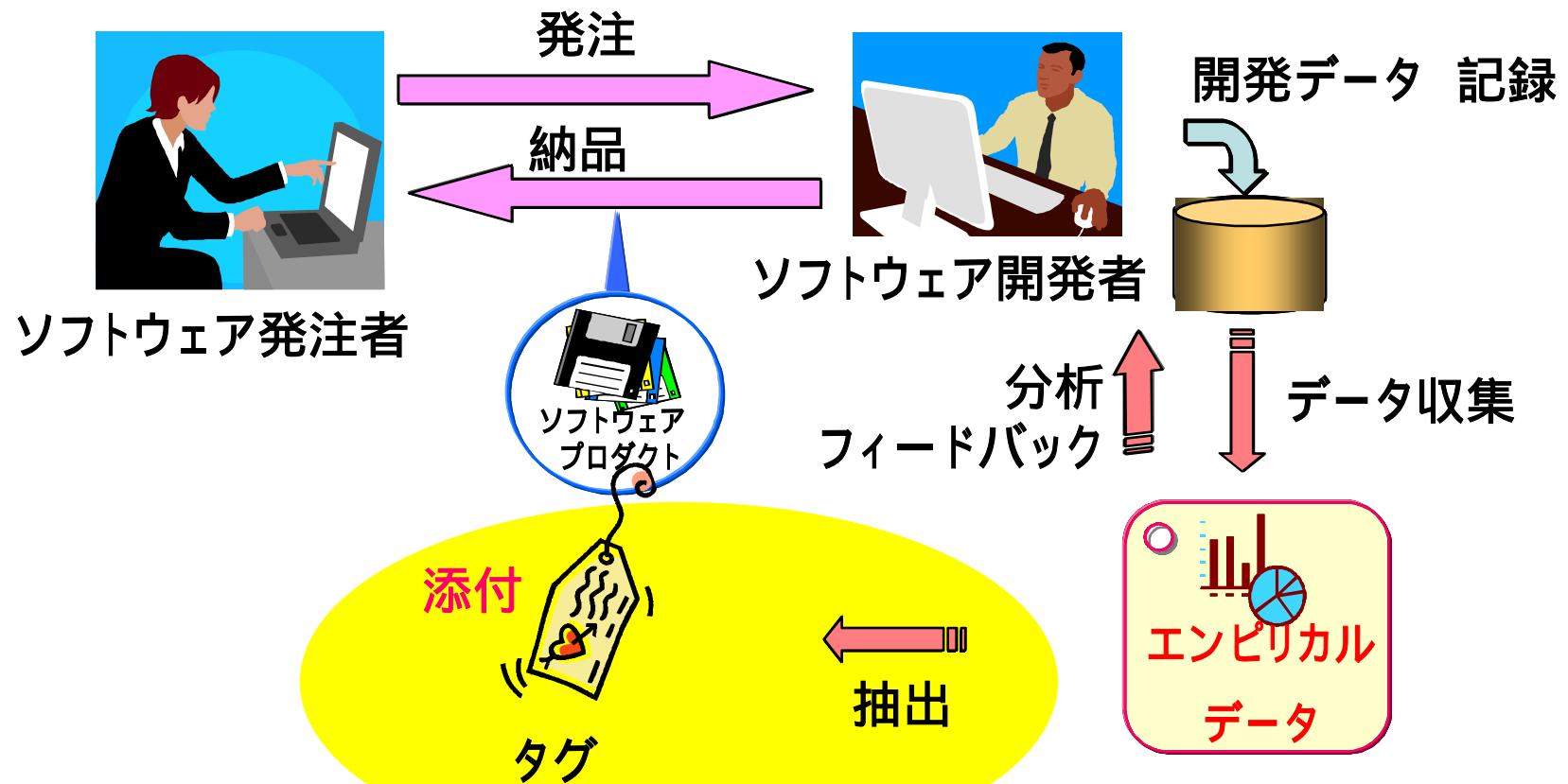
ソフトウェアの開発形態、利用形態、ユーザなどに依存

### 典型例

1. 委託開発時、ユーザが開発状況を知るため
2. 重大問題発生時の原因究明や法的紛争時に、第三者による評価を行うため
3. ソフトウェア部品等の評価を行うため

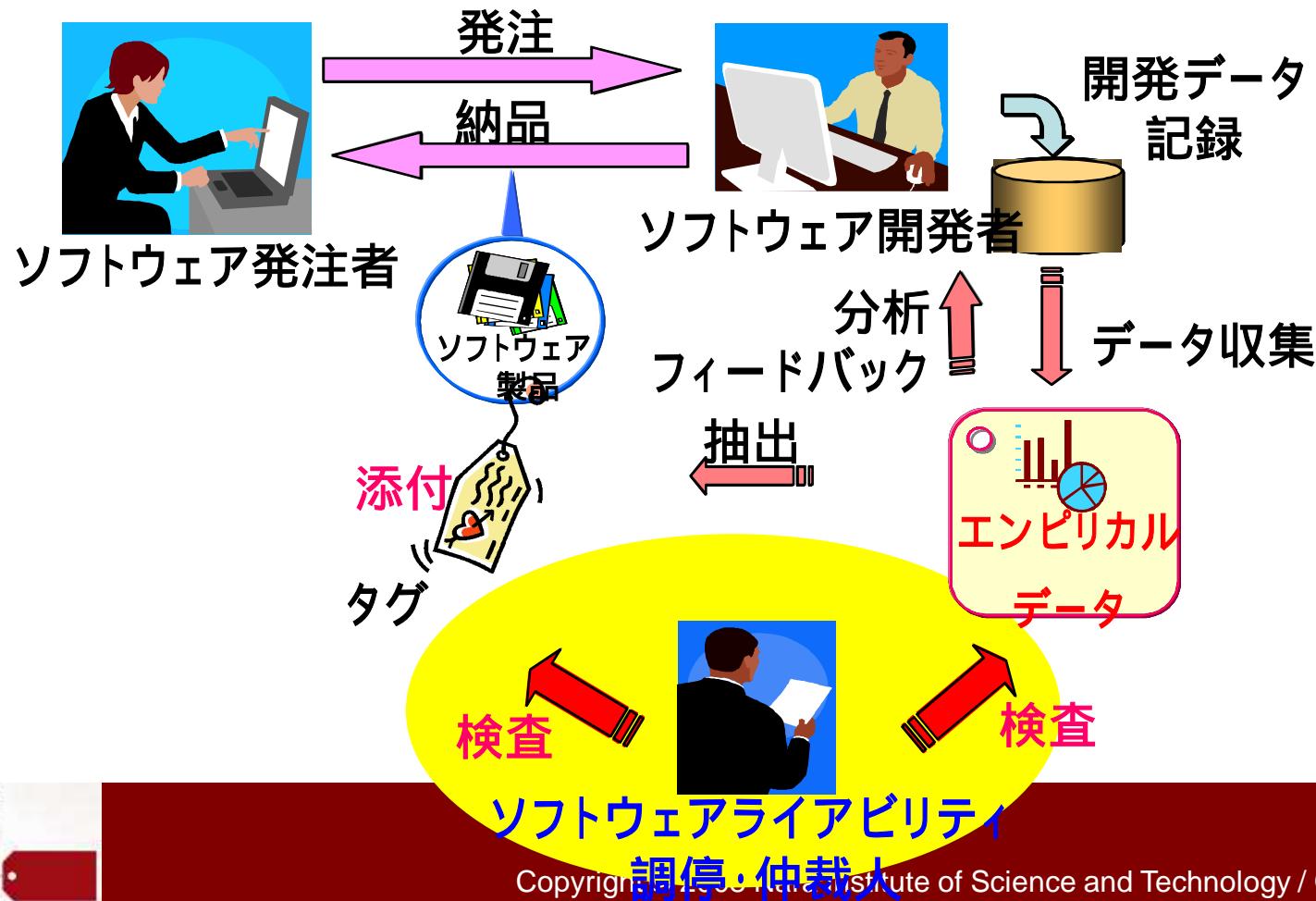
# 1. 委託開発時の利用モデル

- 納品物の品質について検証する



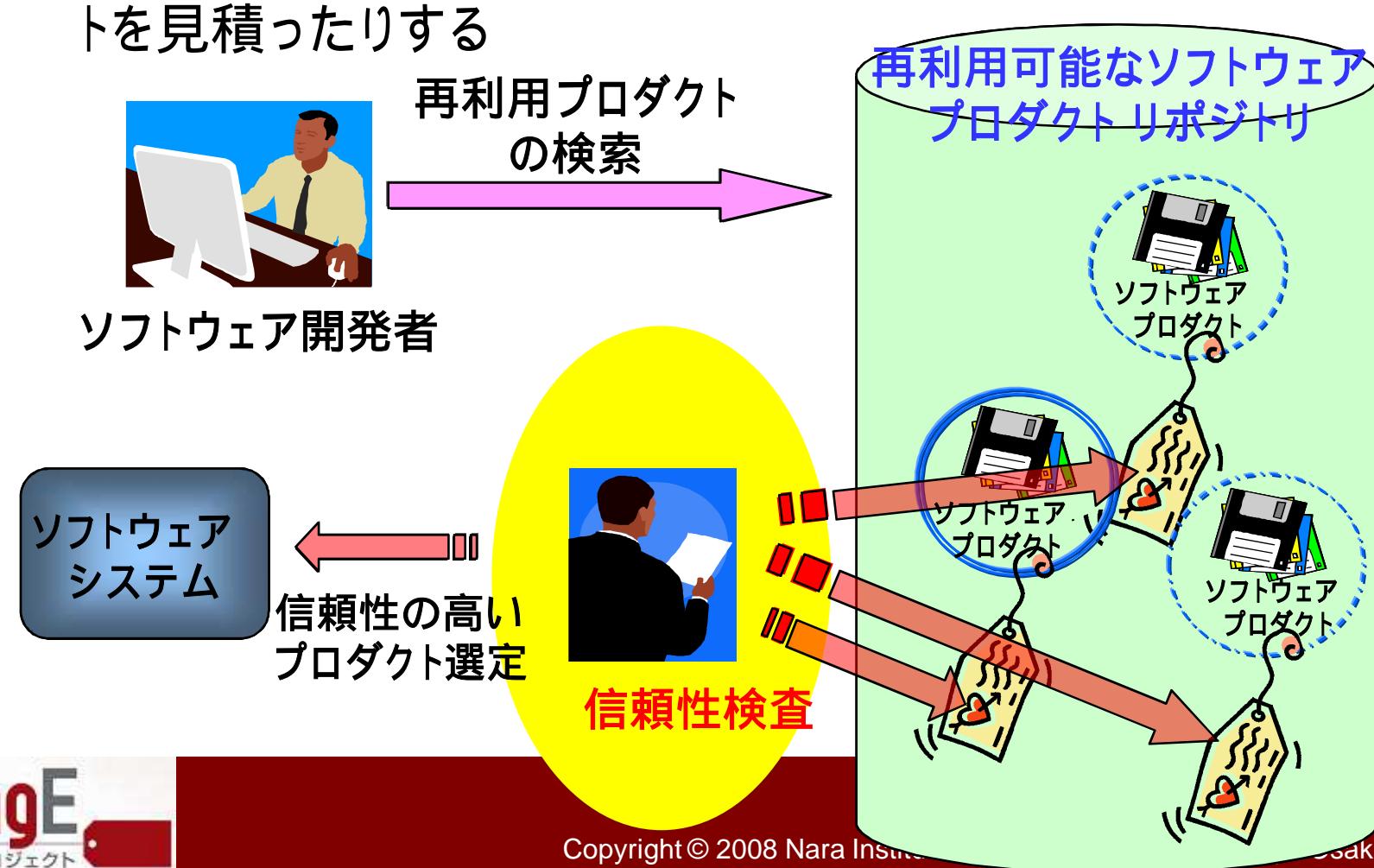
## 2.法的紛争時の利用モデル

- 問題の原因や責任の所在を精査する



### 3.ソフトウェア部品等の評価の利用モデル

- 再利用プロダクトの品質を検査し、選定したり、リスクやコストを見積ったりする



# 実証データ

---

- タグの証拠となるデータ
  - 対象ソフトウェアに関する成果物や開発中に付随的に発生する種々のデータ
    - ソースコード、設計書、障害票、進捗表など生データ
  - 暗号化して安全な場所に保管
  - 開示には一定の条件が必要
    - 重大な不具合発生時の原因調査、法的紛争など

# タグの規格の策定方法

- ソフトウェアタグ規格技術委員会及びWGでの議論
  - 第1回(2007年7月9日) ~ 第11回(2008年7月28日)
  - タグ項目洗練のためのWG(2008年5月1日)
  - 構成員 14組織 27人
- SWEBOK、CMMI、ISO/IEC15939、SECデータ白書等を調査
- プロセス
  1. メンバーからの必要なメトリクスの提案
  2. プロジェクト、プロセスの大分類とその下の中分類の導入
  3. 種々の規格との整合性チェック
  4. 利用シーンの検討
  5. 項目の整理、統合

# ソフトウェアタグ規格技術委員会

---

- 富士通研究所 (Fujitsu Lab)
- 日立製作所 (Hitachi)
- NEC
- シャープ (SHARP)
- SRA先端技術研究所 (SRA Key-Tech Lab)
- 東京証券取引所 (Tokyo Stock Exchange)
- 東芝 (Toshiba)
- JAXA (Japan Aerospace Exploration Agency)
- デンソー (DENSO)
- NTTデータ (NTT Data)
- IPA (Information Technology Promotion Agency, Ministry of Economy, Trade and Industry, Japan)
- 奈良先端科学技術大学院大学 (Nara Institute of Science and Technology)
- 大阪大学 (Osaka University)

# タグの運用方法

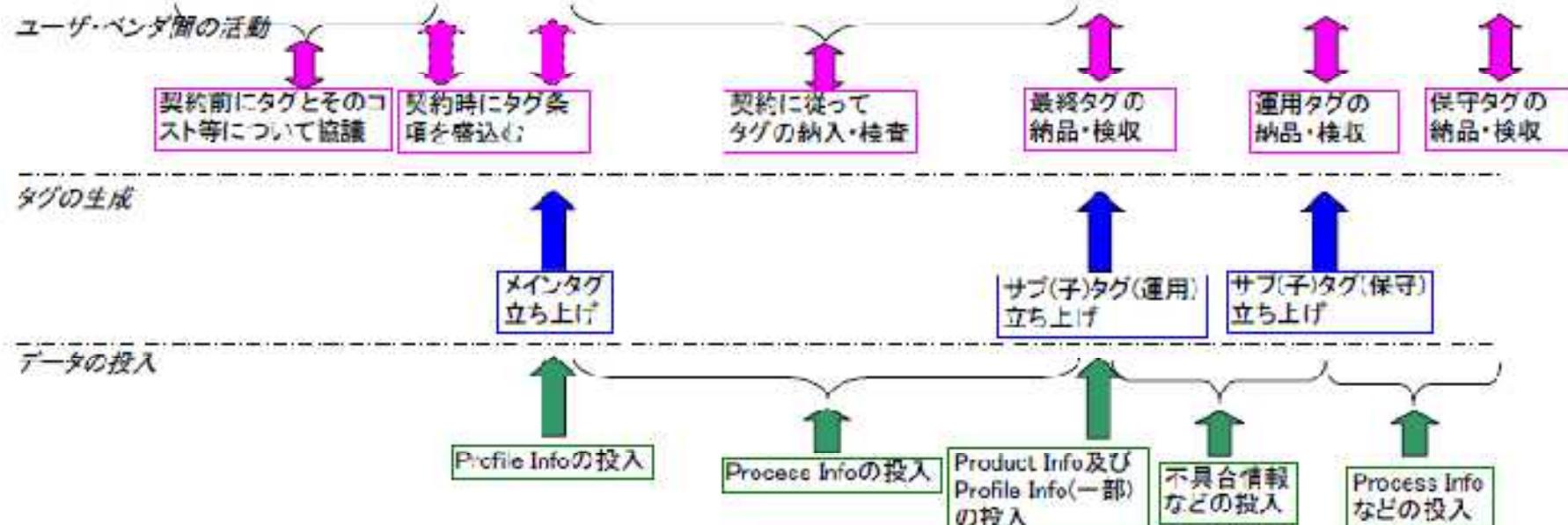
- すべての項目を網羅的にタグ化されることを求めない
- 収集データの対象物、対象範囲、粒度、収集期間(工程)  
受発注者間で取り決める
- 他の項目から算出可能な数値は、項目から除外する(欠陥  
密度など一般的なものは1項目とする)
- 収集データのタグ化のタイミング、もしくは受注者への開示  
タイミングは、受発注者間の協議により決定される  
納品時、工程毎、一定時間ごと(週次、日次)などが考えられる

# タグのライフサイクル例

## プロジェクト・ライフサイクル

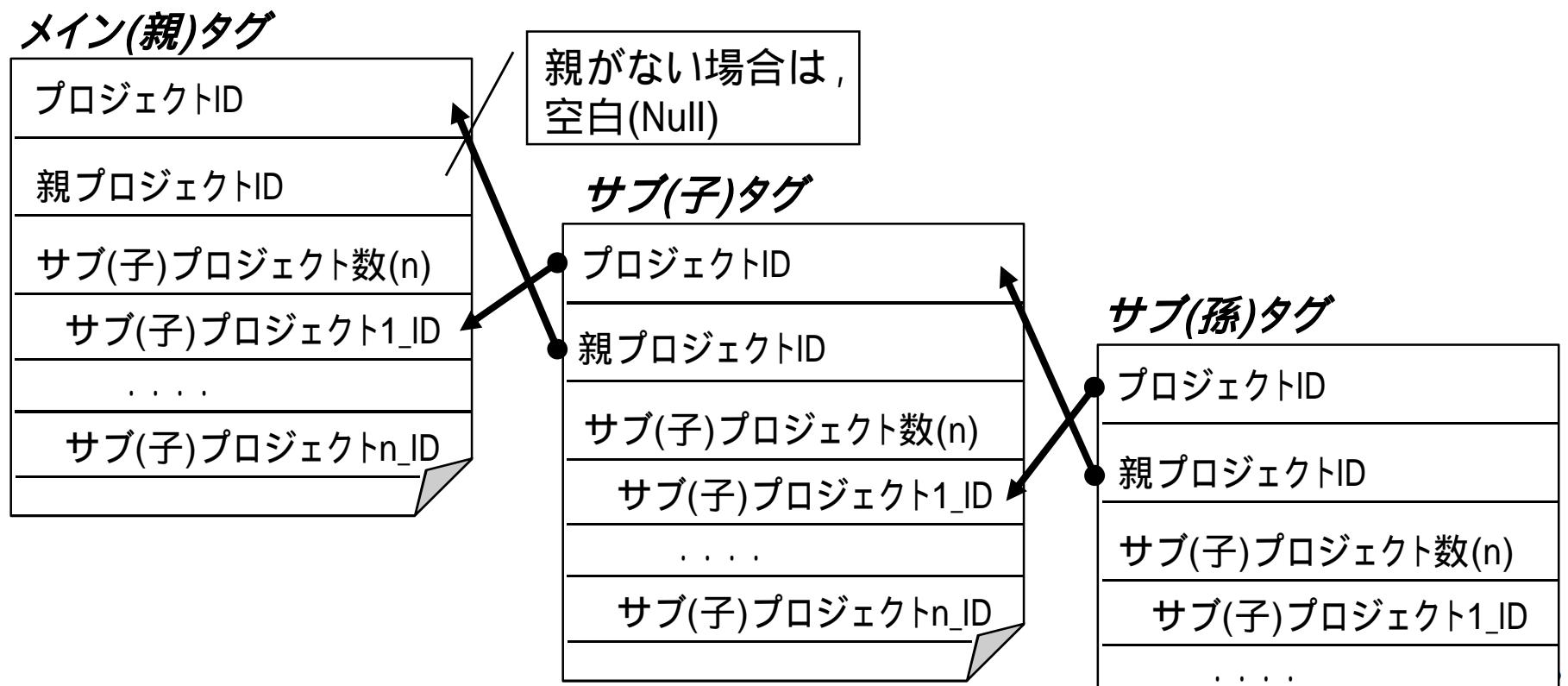
SI CP-2007	企画プロセス	開発プロセス	運用プロセス	保守プロセス	
モデル契約書による工程	システム化の方向性 システム化計画	要件定義 システム設計	ソフトウェア開発 プログラミング ソフトウェアテスト	システムテスト 運用テスト	運用 保守

## ソフトウェアタグ・ライフサイクル



# 階層的なタグの定義

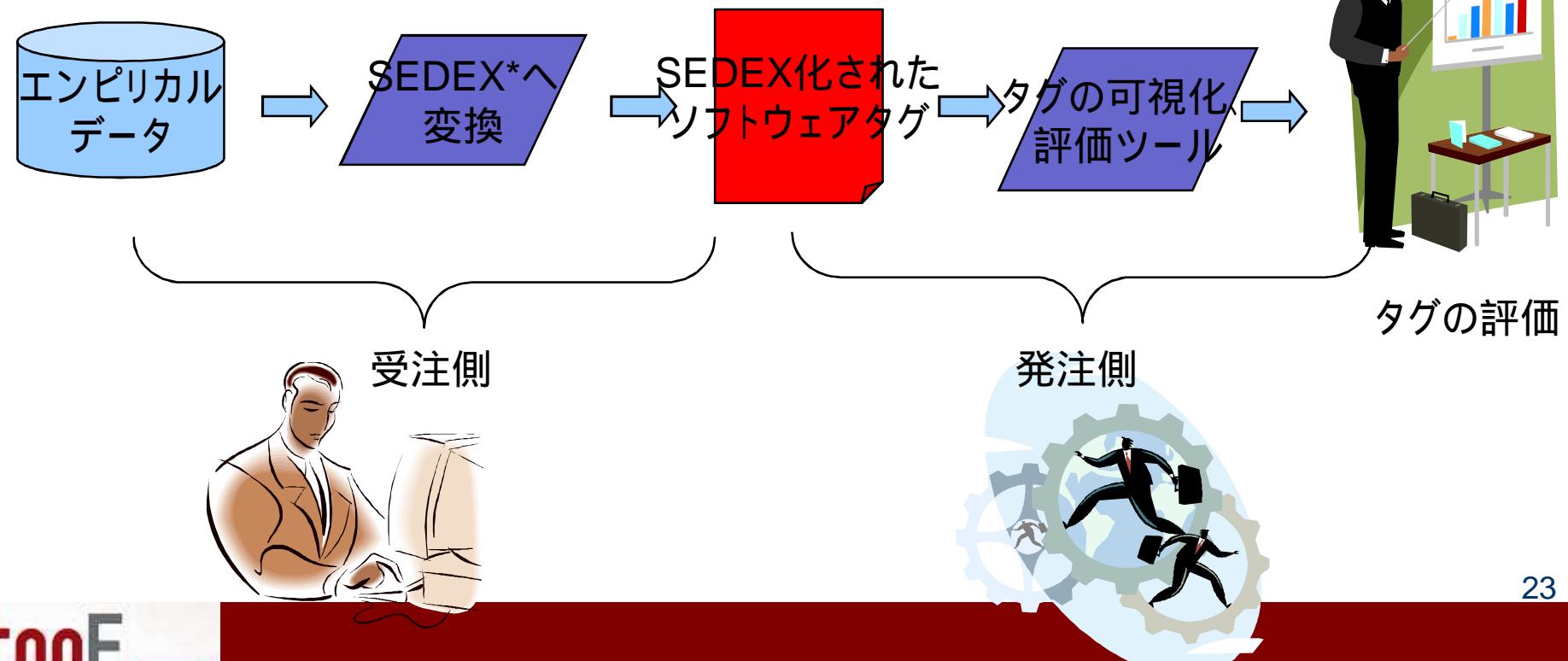
- ・ プロジェクトの構造 ≡ 木 (親子関係)
- ・ 親プロジェクトは子プロジェクトへのリンク情報を持つ(その逆も)



# タグフォーマットの標準化

- 多様なユーザへの開示・流通を目的として、タグ情報の標準フォーマット化をXMLベースで実現する

**SEDEX\*** : Software Engineering Data Exchange language



# StagEプロジェクト

- 文部科学省の「次世代ITのための研究開発」
  - 2007年より5年間、約1億／年
  - 奈良先端科学技術大学院大学と大阪大学
    - 松本健一教授が代表
- ソフトウェアタグ技術の開発普及
  - ソフトウェアタグの規格化
  - ソフトウェアタグデータの収集方式
  - ソフトウェアタグに基づく可視化と評価技術
  - ソフトウェアタグの実装と運用
  - ソフトウェア構築可視化に伴う法的諸問題

# 今後の課題

---

- タグ利用シナリオの構築
  - タグの具体化例の充実
- タグ活用技術の研究・開発
  - タグデータの収集、分析、可視化の手法とツール等の開発
- 実証実験による実用性評価

# リンク

---

- ホームページ: <http://www.stage-project.jp>
- E-mail : [stage-contact@is.naist.jp](mailto:stage-contact@is.naist.jp)