

ソフトウェア工学III

講義概要

ソフトウェア工学講座
門田暁人
akito-m@is.naist.jp
B303室, 内線5311

ソフトウェア工学IIIで学ぶこと

- データに基づくソフトウェア開発支援
 - プロジェクト特性データ
 - プロダクトデータ
 - プロセスデータ
 - ヒューマンファクタ

- ソフトウェアプロテクション
 - ソフトウェアと法律, ライセンス
 - 耐タンパー化技術
 - 盗用防止技術

講義予定

1. 12月 5日(火)1限: 講義概要
 2. 12月 8日(金)2限: 休講(COEフェスティバル)
 3. 12月12日(火)1限: プロジェクト特性データ1(可視化)
 4. 12月15日(金)2限: プロジェクト特性データ2(統計分析)
 5. 12月19日(火)1限: プロジェクト特性データ3(予測)
 6. 12月22日(金)2限: プロジェクト特性データ1(分析) レポート発表
-
7. 1 月 9日(火)1限: プロダクトデータ1(コードクローン)
 8. 1 月12日(金)2限: プロダクトデータ2(複雑さメトリクス)
 9. 1 月16日(火)1限: プロセスデータ, ヒューマンファクタ
 10. 1 月19日(金)2限: プロダクトデータ レポート発表
 11. 1 月23日(火)1限: ソフトウェアプロテクション1 耐タンパー化技術
 12. 1 月26日(金)2限: ソフトウェアプロテクション2 盗用防止技術
 13. 1 月30日(火)1限: ソフトウェアプロテクション3 実践
 14. 2 月 2日(金)2限: プロテクション レポート発表
 15. 2 月 6日(火)1限: 試験
 16. 2 月 9日(金)2限: (予備日)

導入 —— データに基づくソフトウェア開発支援

- ソフトウェア開発に関するデータにどのような種類ものがあり, それらをどう計測・分析・活用するか学ぶ.
- 「計測できないものは制御できない」
- データ活用方法の例
 - 現状把握, 危険の検出
 - スケジュール通りに開発が進んでいるか? 失敗の兆候は?
 - 予測
 - コスト予測, 工数予測, 信頼性予測, 失敗の予測
 - 傾向, 計画立案
 - 平均の生産性は?
 - 開発期間を半分にするには開発要員が何人必要か?

なぜデータを重視するのか？

- データ重視のソフトウェア工学
 - ソフトウェアの開発に関する仮説や疑問に答えるためには、個々のソフトウェアやその開発プロセスについて**基礎的なデータを収集、分析**し、定量的な情報に基づいて議論すべきである。
 - プロジェクトを成功へと導くためには、**プロジェクト内で起こっていることを定量的に評価**し、現状をまずは把握することが必要である。**データ収集、分析、改善のサイクル**
 - Controlled Experiment 比較的小規模でコントロールが容易な環境下での観察実験
 - 実プロジェクトからのデータ収集・分析

ソフトウェア開発に関するデータ

- 開発プロジェクト成果物
 - 提案依頼書, 要求仕様書, 概要設計書, 詳細設計書, ソースコード, テストケースなど
- プロジェクト管理のためのデータ
 - ガントチャート(作業計画・実績管理表), WBS(作業分解図), 設計レビュー報告書, 障害管理票, プログラム変更管理票, テストカバレッジなど
- 分析, 予測, 改善のためのデータ
 - プロジェクト特性表, ソースコードメトリクス, 構成管理データ, リスク分析表, 顧客満足度など

開発のたびに記録・蓄積され、将来の開発に役立てられる。

プロジェクト特性データの例

プロジェクトID	開発種別	業種	アーキテクチャ	開発言語(第1言語)	OS
1	a: 新規開発	a: 銀行	a: クライアントサーバ	d: VISUAL BASIC	g: WINDOWS NT
2	a: 新規開発	a: 銀行	b: スタンドアロン	f: PL/I	c: MVS
3	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
4	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
5	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
6	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
7	a: 新規開発	a: 銀行	b: スタンドアロン	c: COBOL	c: MVS
8	a: 新規開発	a: 銀行	c: 混合	d: VISUAL BASIC	c: MVS
...
73	b: 改修・保守	...	c: 混合	c: COBOL	...

要求仕様_明確度合	開発期間(月数)	ピーク要員数	FP計測手法	規模(FP)	開発工数(人時)
c: ややあいまい	15	15	a: IFPUG	556	24690
	8	6	a: IFPUG	80	825
	6	1	a: IFPUG	77	758
a: 非常に明確	4	6	a: IFPUG	255	2119
	6	0	a: IFPUG	349	2741
d: 非常にあいまい	1	3	a: IFPUG	69	1090
b: かなり明確	4	11	a: IFPUG	375	1855
	6		a: IFPUG	271	1747
b: かなり明確	12	4	a: IFPUG	439	2007

プロセスデータの例: 障害の平均滞留時間の推移

プロセスデータの例: 仕様変更密度の推移

プロダクトデータの例

```
AFG::AFG(JaObject* obj) {  
    objname = "afg";  
    object = obj;  
}  
AFG::~~AFG() {  
    for(unsigned int i = 0; i < children  
        if(children[i] != NULL)  
            delete children[i];  
    for(unsigned int i = 0; i < nodes.si  
        if(nodes[i] != NULL)  
            delete nodes[i];  
}
```

- ・LOC, ノード数, Halsteadメトリクス
- ・オペレータ数/種類数, オペラ
ンド数/種類数
- ・代入変数数, 参照変数数, 外部
変数数
- ・ジャンプ数, ループ数, ネスト数,
Cyclomatic数
- ・FANIN数, FANOUT数, 関数呼
び出し数/定義数
- ・比較演算子数, 論理演算子数,
算術演算子数 ...など

ヒューマンファクタ(スキルデータ)の例

- 専門分野
 - マーケティング
 - セールス
 - コンサルタント
 - プロジェクトマネジメント
 - ソフトウェアディベロップメント
 -
- レベル
 - 1～7

出典:ITスキル標準, IPA

ヒューマンファクタ(スキルデータ)の例

- 専門分野=プロジェクトマネジメント
 - システム開発/アプリケーション開発/システムインテグレーション
- 総合マネジメント
 - レベル5:
ピーク時の要員数10人以上50人未満, または年間契約金額1億円以上のプロジェクト責任者として, プロジェクト計画策定, 計画実施, 変更管理を行い, プロジェクト成功裡に遂行することができる.

出典:ITスキル標準, IPA

導入 —— ソフトウェアプロテクション

- ソフトウェアの法的側面
 - 著作権, 特許, 不正競争防止法
- 契約
 - ソフトウェア開発委託契約, 使用許諾契約, 保守契約
- 犯罪・攻撃に対するソフトウェアの防御
 - 耐タンパー化技術
 - 難読化, 暗号化, 改ざん防止
 - 盗用防止技術
 - 電子透かし, バースマーク

導入 —— ソフトウェアプロテクション

- ソフトウェアの法的側面
 - 著作権, 特許, 不正競争防止法
- ライセンス契約, 受注開発での著作権の取り扱い
- 犯罪・攻撃に対するソフトウェアの防御
 - 耐タンパー化技術
 - 難読化, 暗号化, 改ざん防止
 - 盗用防止技術
 - 電子透かし, バースマーク

ソフトウェアプロテクションの目的

- ソフトウェア内部の秘密(データ, アルゴリズム)を隠す.
- ソフトウェアの改ざんを防ぐ.
- ソフトウェアの再利用(盗用)を防ぐ.
- 要素技術
 - 暗号化, 難読化, 多様化(Software Diversity)
 - アクセス制御
 - アンチデバッグ, アンチ逆アセンブラ
 - 電子透かし, バースマーク
- 法律
 - 著作権, 特許, 不正競争防止法

ソフトウェア特許の現状

- ソフトウェアは特許となる.
 - 無数の出願があり, 特許として認められたものも数多い.
- 訴訟が頻発している.
- ハードウェア製造と同様, ソフトウェア開発においても特許を意識する必要がある.

ソフトウェアは著作物である.

著作権法における「プログラム」の定義

第二条 この法律において、次の各号に掲げる用語の意義は、当該各号に定めるところによる。

十の二 **プログラム** 電子計算機を機能させて一の結果を得ることができるようにこれに対する指令を組み合わせたものとして表現したものをいう。

ソースプログラム, アセンブリプログラム, バイナリプログラム,
1つのモジュールなど

プロジェクト特性からみた ソフトウェア開発

ソフトウェア工学講座
門田暁人
akito-m@is.naist.jp
B303室, 内線5311

プロジェクト特性からみたソフトウェア開発

- 題材：
 - 情報処理推進機構(IPA)ソフトウェアエンジニアリングセンターにおいて収集されたプロジェクト特性データ
 - 1419プロジェクト×約400変数
 - 日本のソフトウェアベンダー19社から収集したもの

出典:ソフトウェア開発データ白書2006, 日経BP

既存システム利用の度合い

- 開発プロジェクト種別
 - **新規開発(59.6%)**
 - ベースとなるシステムが存在しない(存在する場合でも10%未満の場合)
 - **改修・保守(8.7%)**
 - 運用中のシステムに機能追加を行う(追加機能が10%未満)
 - **再開発(4.9%)**
 - 既存システムをほぼ完全に作り直す
 - **拡張(26.8%)**
 - ベースとなるシステムが存在し, 機能追加や変更を行う(10~90%)

誰のためのシステムか？

- 開発プロジェクトの形態
 - 商用パッケージ開発(6.4%)
 - 受託開発(91.8%)
 - インハウスユース(0.1%)
 - 実験研究試作(1.0%)
 - その他(0.7%)

受託開発では、顧客との交渉、要件定義が重要となる。

開発の作業場所

- 「受託開発」の場合の作業場所
 - 顧客先(74件)
 - 自社(372件)
 - その他(19件)

自社内で開発するとは限らない。顧客先で開発することもある。

開発以外の作業

- 開発プロジェクトの作業概要(重複選択可)
 - ソフトウェア開発(1412件)
 - インフラ構築(94件)
 - 運用構築(52件)
 - 移行(179件)
 - 保守(85件)
 - 業務支援(6件)
 - コンサルティング(8件)
 - 品質保証(60件)
 - 現地(本番システム)の環境構築・調整(52件)
 - 顧客教育(40件)

開発以外の作業も多い。

仕事を新規開拓するか否か

- 新規顧客か否か
 - 新規顧客(16.9%)
 - 既存顧客(83.1%)
- 新規業種か否か
 - 新規業種・業務(20.3%)
 - 既存業種・業務(79.7%)

新規顧客、業種は、プロジェクト失敗のリスクを高める場合がある。
その一方で、新規顧客、業種を開拓しないと、生き残っていけない。

外部委託先の決定

- 外部委託先
 - 日本企業(グループ内) (76件)
 - 日本企業(グループ外) (102件)
 - 海外企業(グループ内) (5件)
 - 海外企業(グループ外) (12件)
- 新規の外部委託先か否か
 - 初回利用の会社(25件)
 - 2回以上利用の会社(241件)

グループ内企業への外部委託(外注)がそれなりに多い。
最近では海外企業への外部委託が増えている。(中国, インド)

プロジェクトの成否

- プロジェクト成否自己評価
 - QCD全て成功(81件) QCDのうち2つ成功(39件)
 - QCDのうち1つだけ成功(10件) QCDのうち成功が0(5件)
- 実績の評価(工期)
 - 納期より前倒し(5件), 納期通り(338件), 納期を10日未満遅延(20件), 納期を30日未満遅延(16件), 納期を30日以上遅延(41件)
- 実績の評価(コスト)
 - 計画より20%以上すくない(55件), 計画値以下(308件), 計画値の50%以内の超過(31件), 計画値の100%以内の超過(6件), 計画値の100%を超える超過(22件)
- 実績の評価(品質)
 - 計画より20%以上すくない(30件), 計画値以下(113件), 計画値の50%以内の超過(43件), 計画値の100%以内の超過(2件), 計画値の100%を超える超過(12件)

プロジェクト失敗の原因

- QCD未到達の場合の理由
 - システム化目的不適當
 - RFP内容不適當
 - 要求仕様の決定遅れ
 - 要求分析作業不十分
 - 自社内メンバーの人選不適當
 - 発注会社選択ミス
 - 構築チーム能力不足
 - テスト計画不十分
 - 受入検査不十分
 - 総合テストの不足
 - プロジェクトマネージャの管理不足

利用局面からみたシステムの種類

- 業種(システムがサポートするビジネス分野)
 - 農業, 林業, 漁業, 鉱業, 建設業, 製造業, 電気ガス水道, 情報通信業, 運輸業, 卸売り, 小売業, 金融・保険業, 医療福祉, 公務など
- 業務の種類(システムの対象業務)
 - 経営・企画, 会計・経理, 営業・販売, 生産・物流, 人事・厚生, 在庫, 物流管理, 顧客管理, 商品管理など
- システムの用途
 - ワークフロー支援&管理システム, ネットワーク管理システム, 金融取引処理システム, Webポータルサイト, 文書管理, 3Dモデリング／アニメーション, 画像, ビデオ, 音声処理, OSなど
- 利用拠点数

システム特性 (1/2)

- 利用した業務パッケージ名
 - SAP, Oracle Applications, . . .
- 業務パッケージの初回利用か否か
- パッケージの機能規模の比率
- アーキテクチャ
 - スタンドアロン, メインフレーム, 2層クライアントサーバ, 3層クライアントサーバ, イン트라ネット/インターネット
- 開発対象プラットフォーム
 - Windows95/98/Me系, WindowsNT/2000/XP系, HP-UX, AIX, Solaris, Redhat Linux, SUSE Linux, TRONなど
- 開発言語

複数の開発言語が使われることが多い。

システム特性 (2/2)

- Web技術の利用
 - HTML, XML, Java Script, ASP, Apache, Tomcat, WebLogic, WebSphereなど
- DBMSの利用
 - Oracle, SQL Server, PostgreSQL, MySQL, Sybase, Accessなど

開発の進め方 (1/2)

- 開発ライフサイクルモデル
 - ウォーターフォール, 反復型, その他
- 運用ツールの有無
 - JP1, SystemWalker, 千手, A-Auto, その他, 無し
- 類似プロジェクト参照の有無
- 構成管理ツールの利用
 - ClearCase, CVS, PVCS, SCCS, VSS
- 設計支援ツールの利用
- ドキュメント作成ツールの利用
- デバッグ/テストツールの利用
- CASEツールの利用
- コードジェネレータの利用

開発の進め方 (2/2)

- 開発方法論の利用
 - 構造化分析設計, オブジェクト指向分析設計, データ中心アプローチ(DOA), その他, 無し
- 各種再利用率
 - システム化計画再利用率
 - 要件定義書再利用率
 - 基本設計書再利用率
 - 詳細設計書再利用率
 - ソースコード再利用率
 - テストケース再利用率
- 開発フレームワークの利用
 - Struts, .NETフレームワーク, JBOSS, J2EEなど

ユーザ要求管理(抜粋)

- 要求仕様の明確さ
- ユーザ担当者の要求仕様関与
 - ユーザが全て作成
 - ベースはユーザが作成し, 細部はベンダが作成
 - ラフなものをユーザが作成し, 残りはベンダが作成
 - ベンダが全て作成
- ユーザ担当者のシステム経験
 - ストレス無く話が通じる
 - 概ね話が通じる
 - 多くの点で説明を要する
 - 全てを説明する必要がある
- 要求仕様の変更発生状況, 要求レベル(信頼性, 保守性, 移植性, セキュリティ. . .)

要員スキル(抜粋)

- ITスキル標準の「プロジェクトマネジメント」での評価
 - レベル3~7
- 要員スキル
 - 業務分野の経験
 - 分析・設計経験
 - 言語・ツール利用経験
 - 開発プラットフォームの使用経験

システム規模

- ファンクションポイント計測手法
 - IFPUG, SPR, NESMA, COSMIC-FFPなど
- ファンクションポイント
- SLOC (Source Lines of Code)
- 文書ページ数
 - システム化計画書
 - 要件定義書
 - 基本設計書
 - 詳細設計書
- 他指標
 - DFDデータ数, プロセス数
 - 画面数
 - ユースケース数

工期

- 工程別工期(計画, 実績)
 - システム化計画
 - 要件定義
 - 基本設計
 - 詳細設計
 - 製作
 - 結合テスト
 - 総合テスト
- アイドリング期間
 - 顧客のサイン待ち, テストデータの受領待ちなど

工数

- 人時換算係数
 - 例:1人月=160人時
- プロジェクト総工数
- プロジェクト総工数に含まれる工程
- 社内実績工数
- レビュー実績工数, 実施回数, 指摘件数
- 外部委託工数
- 社内平均要員数, ピーク要員数
- 外部委託平均要員数, ピーク要員数
- プロジェクト開発工数計画値(基本設計開始時)
- プロジェクト開発工数計画値(詳細設計開始時)

品質

- 発生不具合数
 - システム稼動後1ヶ月:現象数, 原因数
 - システム稼動後3ヶ月:現象数, 原因数
 - システム稼動後6ヶ月:現象数, 原因数
- 発生不具合数(重大度別内訳)
 - 重大〇件
 - 中度〇件
 - 軽微〇件
- テストケース数
 - 結合テスト
 - 総合テスト
- 品質保証体制
- テスト体制

おわりに

- データから見てくるソフトウェア開発
- プロジェクト特性データに含まれる項目は、それぞれ意味がある。
- 大分類
 - 開発プロジェクト全般
 - 利用局面
 - システム特性
 - 開発の進め方
 - ユーザ要求管理
 - 要員スキル
 - システム規模
 - 工数(コスト)
 - 品質