

ソフトウェア工学III

プロセスデータ・資源データと プロジェクト管理

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

ソフトウェア開発に関するデータ

- プロジェクト特性データ
 - プロジェクト種別, 利用局面, システム特性, ユーザ要求管理, 要員スキル, システム規模, 工数, 品質など
→将来の開発の工数予測や計画立案などに役立つ.
- プロダクトデータ
 - 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
→現在の開発・保守や次期バージョン開発に役立つ.
- プロセスデータ, 資源データ
 - 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, 開発者の能力, プロダクトデータを時系列に並べたもの
→プロジェクト管理(マネージメント)に役立つ.

本講義の目的

- プロセスデータ, 資源データにはどのようなものがあり, プロジェクト管理にどう役立てられるかを学ぶ.
- データの例
 - WBS
 - PERT
 - ガントチャート
 - 設計書, ソースコードの履歴
 - テスト管理表
 -

プロジェクト管理とは

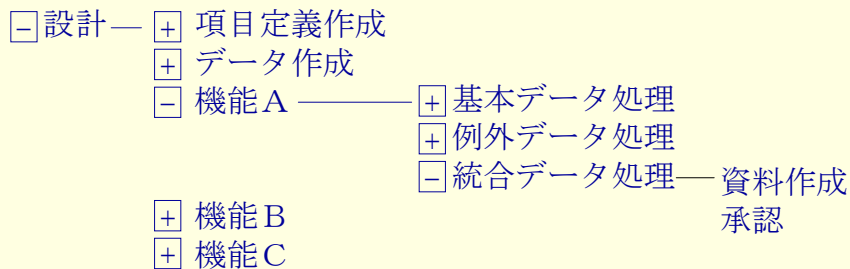
- プロジェクトの現状を把握する.
- PCDA (Plan Do Check Action) サイクルをまわす.
 - 計画, 実施, 確認, アクション
- プロジェクト関係者間の調整を行う.
 - 要求, 利害, 意見, 対話, 報告, 連絡
- 変更の管理
 - 仕様変更, メンバー変更, スケジュール変更
- マイルストーンの管理
 - 設計の終了日, コーディングの終了日, . . .

プロジェクト管理の3大技術 (1)WBS

■ WBS (Work Breakdown Structure)

- プロジェクト全体を詳細な作業に分割する手法
 - 作業=成果物がある.
 - 作成すべき成果物(要求仕様書, 設計書, ソースコード, レビュー報告書など)を, より詳細な成果物に分割する.
 - 木構造

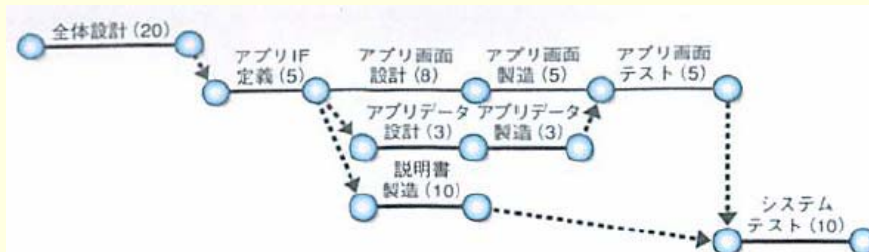
■ 例



プロジェクト管理の3大技術 (2)PERT

■ PERT (Program Evaluation and Review Technique)

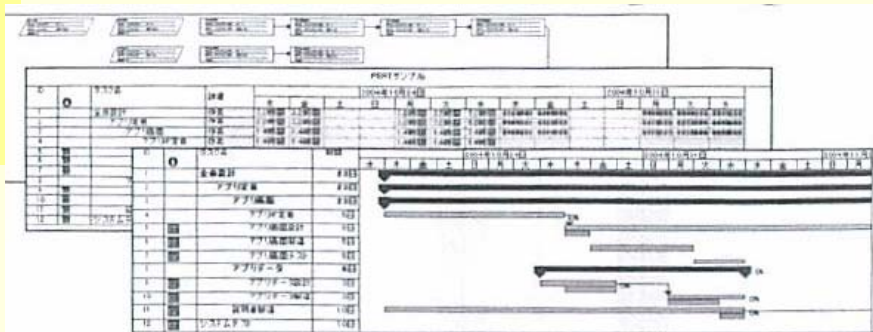
- WBSで洗い出された作業を組み合わせ, プロジェクトを最短で終了させるための計画を立案する手法
- ネットワークロジック+見積もり値+クリティカルパス
- 例



出典: IPA-SEC編, 組み込みソフトウェア開発におけるプロジェクトマネジメント導入の進め, SEC BOOKS

プロジェクト管理の3大技術 (3)ガントチャート

- Gantt chart(ガント図, 工程管理図, 線表)
 - WBS+PERT+リソース(人間)
 - 作業, 人, スケジュールを横型棒グラフで示したもの
 - 各作業はいつ始まり, いつ終わるのか
 - 誰が各作業を担当するのか



出典: IPA-SEC編, 組み込みソフトウェア開発におけるプロジェクトマネジメント導入の進め, SEC BOOKS

ガントチャートの例

- 2種類のガントチャート
 - タスク・ガントチャート
 - リソース・ガントチャート

(MS Projectを用いた例)

ガントチャートの計画と実績

- 計画の例

ガントチャートの計画と実績

- 実績の例

テスト管理表

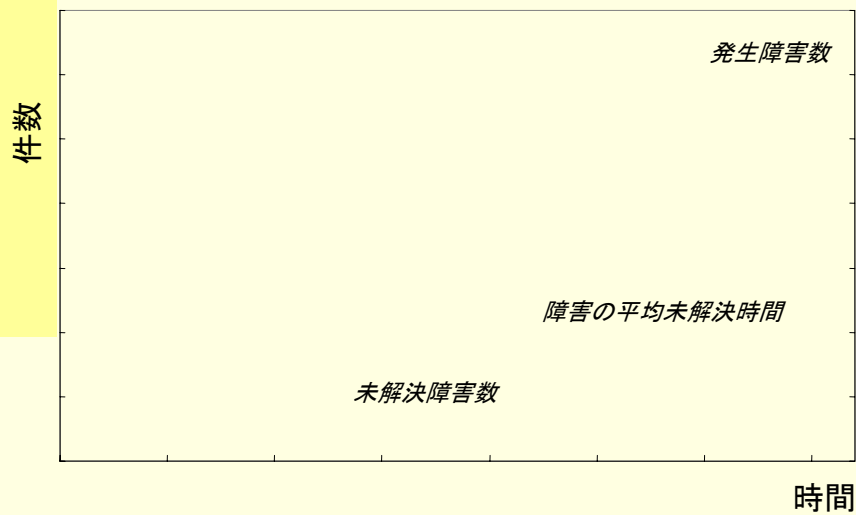
- ガントチャートに準じる
 - ガントチャート+テストケース数+障害情報

No.	機能大分類	機能小分類	テスト項目番号	テスト項目名	開始時刻 (Start)	テスト開始時刻	テスト終了時刻	実時間 (El)	正常率 (%)	テスト実施状況											
										発生件数	検出件数	対応完了日	未対応件数	対応予定日	仕様変更	仕様変更	仕様変更	CD 0%1	CD 0%2		
1	外部連携機能	外部連携処理	検査	S/1	S/1	8	8	0	100	8											
2		外部連携処理	検査	S/1	S/1	2	2	0	100	2											
3		外部連携処理	検査	S/1	S/1	10	10	0	100	7	3	3	S/11			1					2
4		外部連携処理	検査	S/1	S/1	4	4	0	100	3	1	1	S/11								1
5	検索機能	検索機能	検査	S/8	S/8	8	8	0	100	8											
6		検索機能	検査	S/8	S/8	4	4	0	100	4											
7		検索機能	検査	S/8	S/8	9	9	0	100	9											
8		検索機能	検査	S/8	S/8	1	1	0	100	1											
9	連携機能	連携機能	検査	S/8	S/8	24	24	0	100	15	9	9	S/17			1					4
10		連携機能	検査	S/8	S/16	10	10	0	100	5	5	5	S/10								3
11		連携機能	検査	S/8	S/8	3	3	0	100	3											
12		連携機能	検査	S/8	S/8	8	8	0	100	7	1	1	S/8								1
13	メール機能	メール機能	検査	S/8	S/8	5	5	0	100	5											
14		メール機能	検査	S/8	S/8	9	9	0	100	8	1	1	S/8								1
15		メール機能	検査	S/8	S/8	5	5	0	100	4	1	1	S/8								1
16		メール機能	検査	S/8	S/8	5	5	0	100	4	1	1	S/8								1
17	メール機能	メール機能	検査	4/27	4/27	8	8	0	100	8											
18		メール機能	検査	4/27	4/27	2	2	0	100	2											
19		メール機能	検査	4/27	4/28	3	3	0	100	3											
20		メール機能	検査	4/28	4/28	5	5	0	100	3	2	2	S/11			2					1
21			検査	4/27	4/28	15	15	0	100	14	1	1	4/28								

障害管理表(1障害あたりの記録)

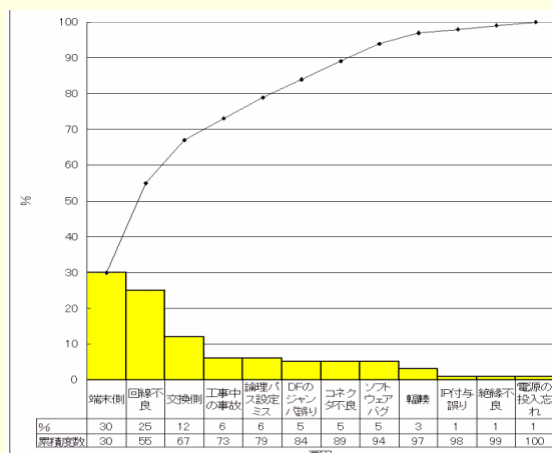
- 障害番号
- 対応状況
- 起票分類
- 起票日
- 起票者
- 責任者
- 問題概要
- 発見日
- テスト対象
- 発見箇所
- 発見工程
- 発見作業
- 発見バージョン
- 問題処理機能
- 問題内容
- 再現度
- 再現方法
- 重要度
- 優先度
- 問題原因
- 混入工程
- 混入対象・バージョン
- 本来発見すべき工程
- 発見できなかった要因
- 修正工程
- 修正対象・バージョン
- 修正工数(人時)
- 修正内容
- 確認方法
- 最終変更日
- 完了日

障害の推移



パレート図(累積度数分布図)

- 障害の各種類について、数の多い順に棒グラフと累積百分率を示したもの



出典: 浅瀬野, QC7つ道具と新QC7つ道具, <http://asaseno.cool.ne.jp/study/qc1.html>

設計書の推移

ソースコード行数の推移

事例

- 事例1: 開発作業履歴を用いたマイクロプロセス分析
 - 障害管理ツール(GNATS), 構成管理ツール(CVS)の履歴
- 事例2: GQM(Goal-Question-Metric)モデルに基づくプロジェクトのモニタリング

事例1: マイクロプロセス分析

- 目的
 - ソフトウェア製品の品質を評価するのと同様に, ソフトウェアプロセスの品質を評価したい.
- アプローチ
 - 作業記録からプロセスに関する詳細な(マイクロな)情報(作業の順序, 時間, 作業量)を得る.
 - 作業手順を定めるプロセスモデルを定義する.
 - プロセスモデル通りの作業が行われているかどうかを評価する.

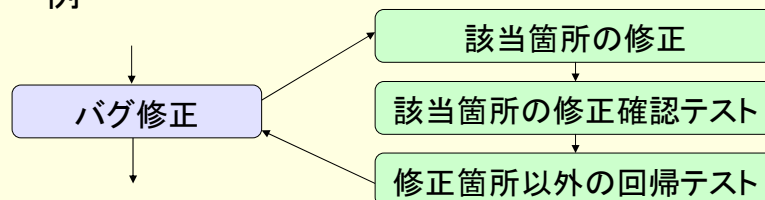
作業記録の例

- 障害管理ツールの操作履歴
 - 障害発生時刻, 障害修正時刻,
- 構成管理ツールの操作履歴
 - ソースコードのチェックイン, チェックアウト

日時	操作	ファイル名	version	コメント
2003-08-07 10:5...	CHECKOUT	EmpiriPri		
2003-08-07 11:5...	CHECKOUT	EmpiriPri		
2003-08-07 12:0...	CHECKOUT	EmpiriPri		
2003-08-07 12:0...	CHECKOUT	EmpiriPri		
2003-08-07 12:5...	CHECKOUT	EmpiriPri		
2003-08-07 13:1...	CHECKOUT	EmpiriPri		
2003-08-07 13:1...	CHECKOUT	EmpiriPri		
2003-08-07 13:1...	ADD	EmpiriPri/XML/onTaira	1.1	実装作業の前に、Antを利...
2003-08-07 13:1...	ADD	EmpiriPri/XML/build.xml	1.1	実装作業の前に、Antを利...
2003-08-07 13:1...	ADD	EmpiriPri/XML/build.sh	1.1	実装作業の前に、Antを利...
2003-08-07 13:2...	CHECKOUT	EmpiriPri		
2003-08-07 14:4...	ADD	EmpiriPri/XML/docs/Empi...	1.1	XMLを解析して、データ...
2003-08-07 15:3...	CHECKOUT	EmpiriPri		
2003-08-07 15:4...	CHECKOUT	EmpiriPri		
2003-08-07 15:4...	CHECKOUT	EmpiriPri		
2003-08-07 15:4...	CHECKOUT	EmpiriPri		
2003-08-07 15:4...	ADD	EmpiriPri/translator cvs/C...	1.1	初期登録 : Translator (cv...
2003-08-07 15:5...	CHECKOUT	EmpiriPri/translator cvs		

ソフトウェアプロセスモデル

- ソフトウェア開発における作業間の順序関係や包含関係を記述したもの
例



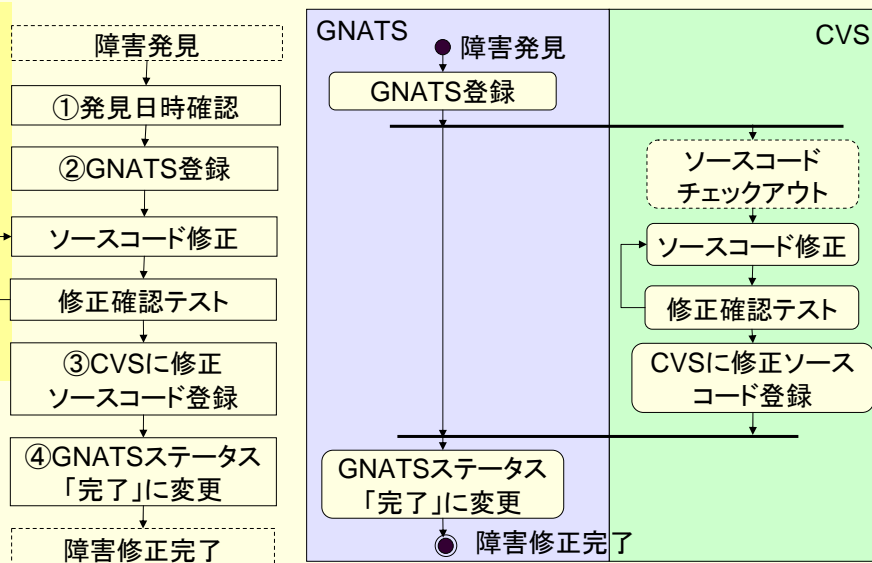
- アクティビティ図 (UML) などを用いて表現できる.

実験

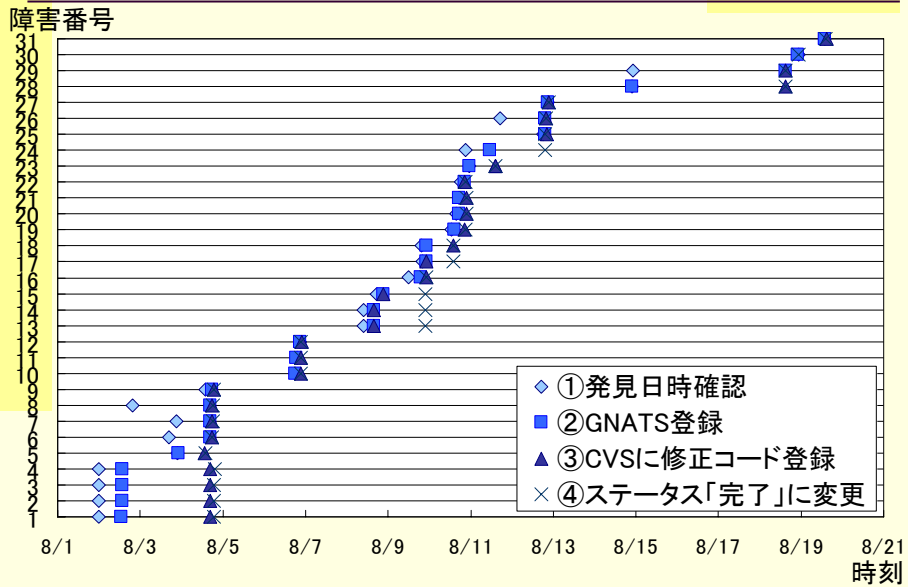
- 目的: 商用プロジェクトのプロセスの品質を評価する.
 - プロセスモデルに準拠した作業が行われているか.
- 対象プロジェクト
 - プローブ情報システム開発
 - マルチベンダによるウォーターフォール型中規模開発
- 分析の対象
 - 工程: コーディング工程後半～組織内結合試験工程
 - 作業記録: CVSとGNATS(障害管理ツール)
 - CVS記録: 518レコード, バグ数: 31件

対象プロセスモデル

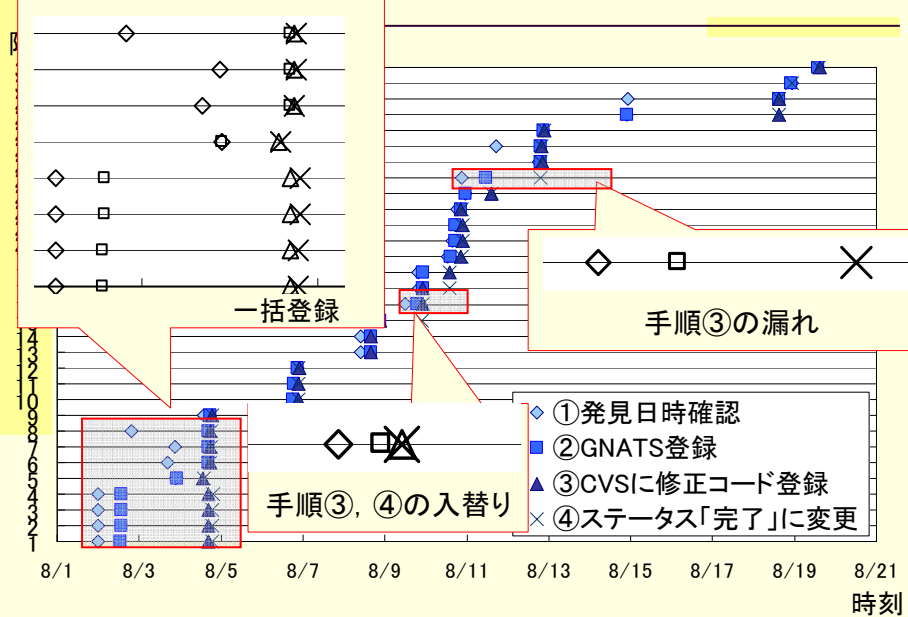
開発メンバにお願いした手順 ———— 想定プロセスモデル(アクティビティ図)



結果:作業履歴



結果:作業履歴



結果:手順漏れ, 不遵守, 登録情報漏れ一覧

項目	該当件数	割合
手順漏れ(②)	0	0%
手順漏れ(③)	2	6.5%
手順漏れ(④)	3	9.7%
手順の不遵守(①→②)	7	22.6%
手順の不遵守(②→③)	0	0%
手順の不遵守(③→④)	7	22.6%
登録情報漏れ, 誤り(②)	7	22.6%
登録情報漏れ, 誤り(③)	0	0%

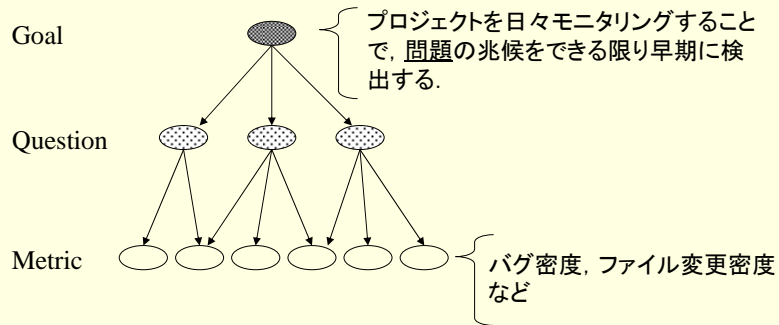
- ①発見日時確認, ②GNATS登録,
③構成管理ツールに修正ソースコード登録
④GNATSステータス「完了」に変更

まとめ

- プロセスの品質を評価するためのマイクロプロセス分析を行った.
 - 障害修正のプロセスが定義された商用開発プロジェクトのデータに適用した.
 - 5個の手順漏れ
 - 14個の手順の不遵守
 - 7個の登録情報漏れ・誤り
- を発見することができた.

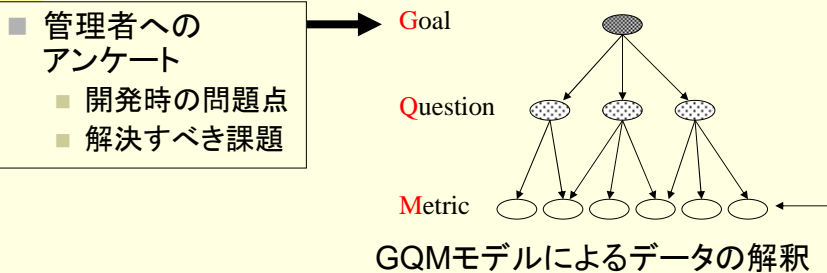
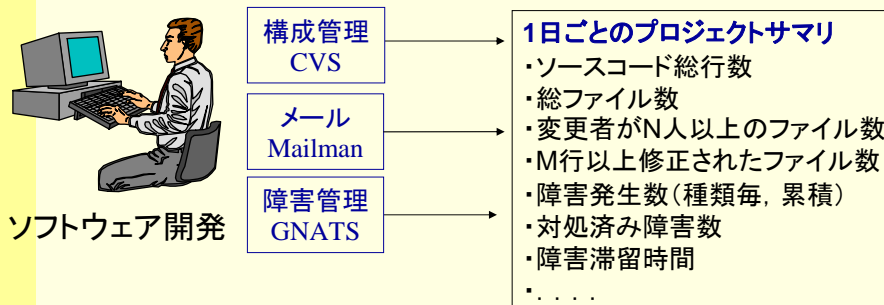
事例2: GQMモデルに基づくプロジェクトのモニタリング

- GQMパラダイム: ゴール指向の分析アプローチ
 - 1984年にBasiliとWeissによって提唱された.
 - 目的を明確化し, 収集データやメトリクスと**対応付ける**.



[1] V.R. Basili and D.M. Weiss, "A methodology for collecting valid software engineering data," *IEEE Trans. on Software Engineering*, Vol.SE-10, No.6, pp.728-838, 1984.

GQMに基づくプロジェクトモニタリングの概要



遅れが生じた主な原因

- 上流工程の設計品質が悪い。単体テスト・結合テスト段階のプロジェクト管理不良。
- 要求仕様の矛盾。期待した性能が出ない。ライブラリ、パッケージソフトウェアの品質の問題。
- 要求が大幅に膨れ上がった。仕様変更の発生。
- 設計途中で見積り精度の悪さが発覚。システムテストで致命的な設計不備が見つかった。
- 要件が確定しない。本来単体テストで出るべきバグが後工程で出現。
- 詳細設計フェーズで規模見積り過少が判明。結合テスト工程で単体テストの不備が判明。

遅れが生じた主な原因

- 現物あるいは設計書を（顧客に）見せると変更要求が多発する。
- 顧客との関係。技術不足。マネジメント（金・人）不足
- 結合テスト時にインタフェースの不整合が発生した。仕様の理解不足。
- 要求仕様の顧客との合意もれ。詳細仕様のあいまいさによるブレークダウン不能。仕様変更の頻発
- 新規プロジェクトのため、業務知識の欠如。新技術の導入による開発ノウハウの欠如。

プロジェクト遅延の兆候

- プロジェクト遅延が発覚する前に、何らかの兆候はありましたか？

はい:17人 いいえ:3人 無回答:1人

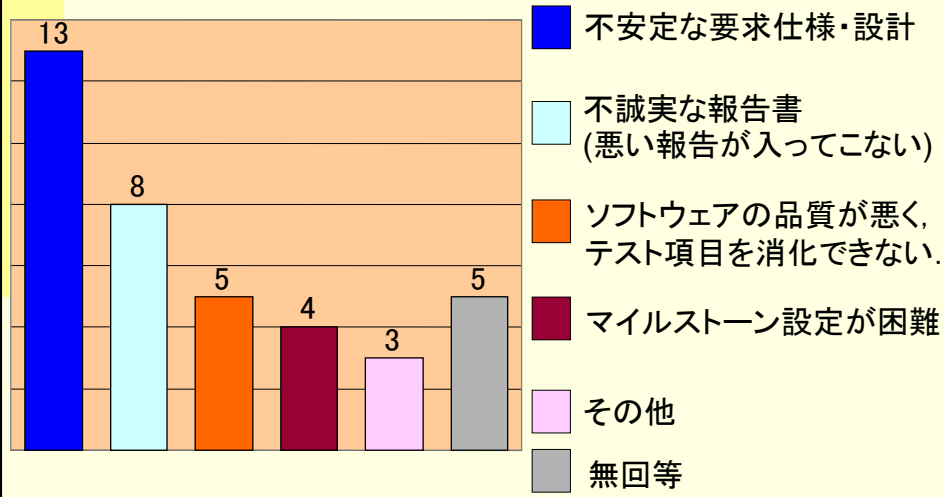
- 考慮もれのチェック・レビュー等がなかった。ライブラリ、パッケージソフトの検証が十分でない
- 完成が少しずつ遅れだす
- 「完了」との進捗報告に具体的な根拠がない
- 単体テストの進捗遅れ
- 仕様変更が多発。節目での見積り規模の増加
- 工数の増大。不具合の増大

プロジェクト遅延の兆候

- 命令系統のみだれ
- 設計仕様書の頁数が少ない
- 不安定な仕様。別企業の開発チームとの調整の難航
- 仕様の確認の打ち合わせが少なく不安だったが、納期が短く、進めるしか選択がなかった。
- 不具合の解決が進まない。成果物の品質が悪い。
- 打合せの工数は大きいですが、成果物ができていない
- システムテストでバグが想定以上に発生していた
- 兆候はあったが、マネジメントされていなかったため、何をどうしたらよいかわからない状態だった

プロジェクト進捗管理における課題

- プロジェクト進捗管理における課題をお持ちでしたらご回答ください。(複数回答)



GQMモデリングの材料

- Goal
 - プロジェクトの問題: 不安定な要求・不完全な設計・劣悪なソースコード品質など
 - 目的: プロジェクトの現状理解, 問題の検出
 - プロジェクト管理者の立場
- Question
 - ?
- Metric
 - 構成管理システム(CVS): 変更したファイル, 変更者, 日時, 追加行数, 削除行数など
 - メール管理システム(Mailman): 送信者, 宛先, 表題, リファレンス, 本文など
 - 障害管理システム(GNATS): 対象ファイル, 報告者, 日時, 種別, 状態(未解決/解決済), 優先順位(高/中/低), 重要度(高/低)など

GQMモデルにおけるGoal

- プロジェクトをモニタリングすることで、問題の兆候をできる限り早期に検出する。

- ・要求が不安定
- ・設計が不完全
- ・リソース配置が不適切
- ・コーディング品質が悪い

これら4つの問題の兆候を検出するためのGQMモデルをそれぞれ作成した。

Model 1 ... 要求の不安定さの評価 (1/2)

Goal

構成管理データ(CVS)と障害管理データ(GNATS)からプログラム変更、仕様変更パターンを解析し、企業における特定のプロジェクトにおいて、プロジェクトマネージャの視点から、要求の不安定さについて評価する

Question

プログラム変更
頻度は？

プログラム変更
規模は？

プログラム変更
範囲は？

仕様変更数の増
加傾向は？

Hypothesis
(仮説)

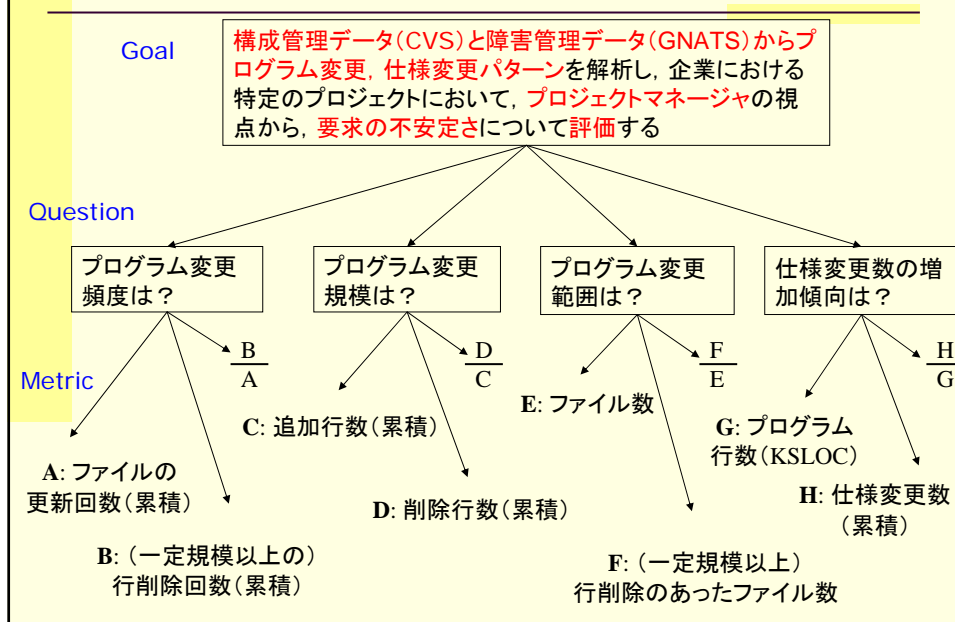
要求仕様が不安定ならば
プログラム変更が頻繁に
起きる。

要求仕様が不安定ならば
規模の大きな変更が発生
する。

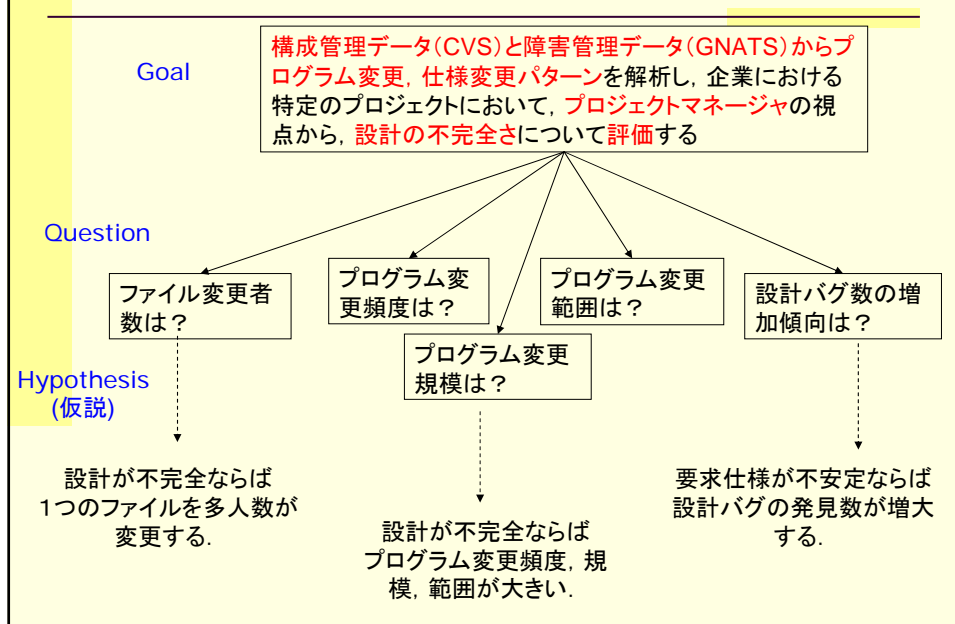
要求仕様が不安定ならば
範囲の広い(多数のファイ
ルにまたがる)変更が発
生する。

要求仕様が不安定ならば
仕様変更数が著しく増大
する。

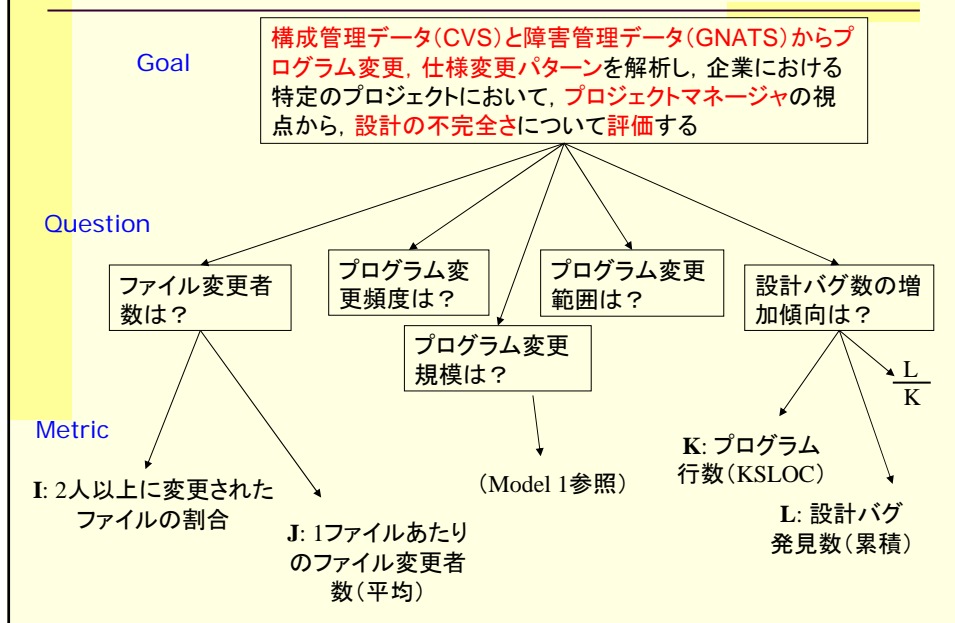
Model 1 ... 要求の不安定さの評価 (2/2)



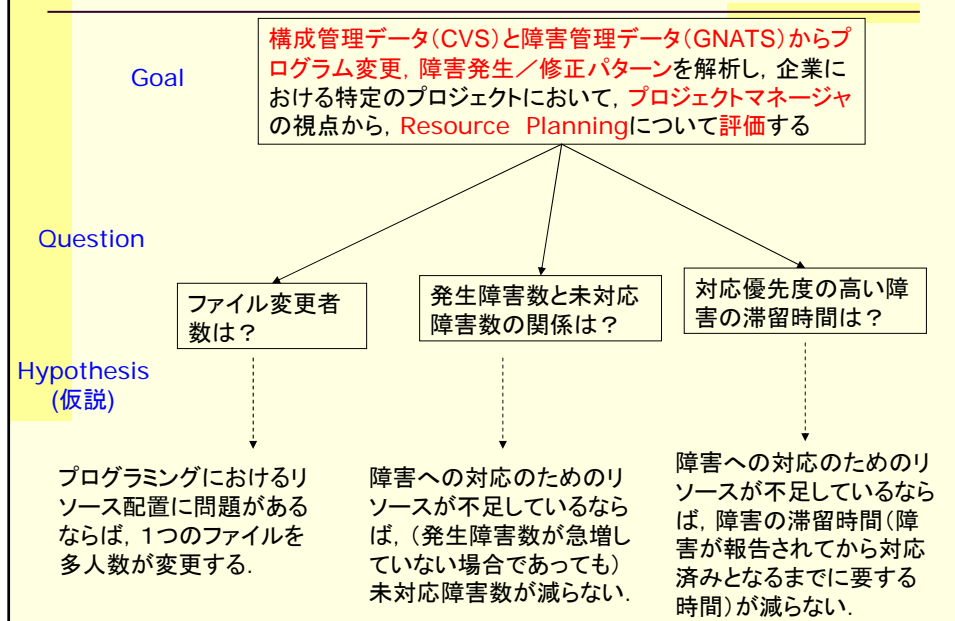
Model 2 ... 設計の不完全さの評価 (1/2)



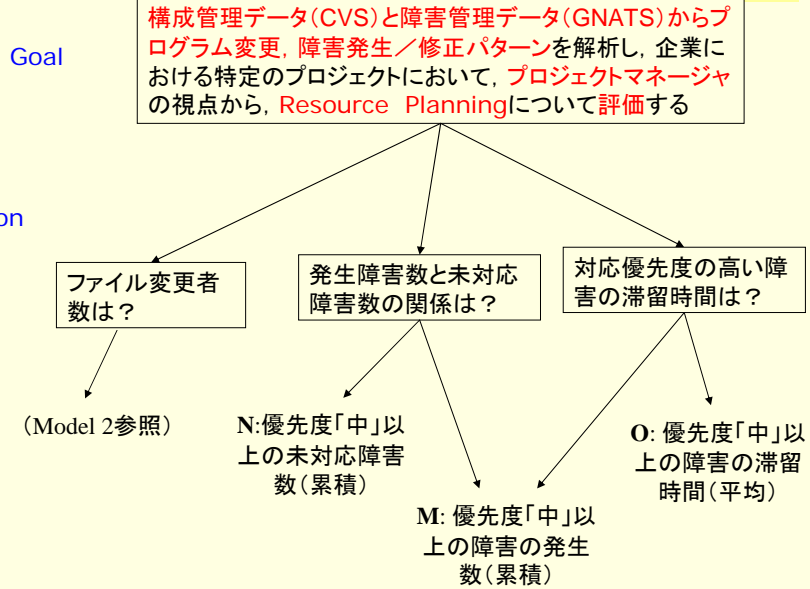
Model 2 ... 設計の不完全さの評価 (2/2)



Model 3 ... リソース配置の評価 (1/2)



Model 3 ... リソース配置の評価 (2/2)



分析例1

分析例2

分析例3

分析例4

分析例5

まとめ

- GQMモデルによるプロジェクトモニタリング事例の紹介
 - 事例1:不安定な仕様
 - 事例2:頻繁なソースコード変更
 - 事例3:人手不足
 - 事例4:要追加テスト
 - 事例5:開発要員の変更

本講義のまとめ

- プロジェクト管理に役立つプロセスデータ・資源データ
 - WBS … 作業の分解
 - PERT … 作業間の関係, 作業量
 - ガントチャート … WBS + PERT + スケジュール + 人間
 - テスト管理表 … ガントチャート + テスト・障害情報
 - 設計書, ソースコード, 障害の履歴 … プロジェクト状況把握
- 2つの事例
 - 事例1:作業履歴を用いたマイクロプロセス分析
 - 事例2:GQMモデルに基づくプロジェクトのモニタリング

次回 1月19日(金)2限 ― レポート提出 & 発表

- レポート課題
 - コードクローン分析 or プロダクトデータ分析
- 希望者はレポート課題の内容を発表すること
 - レポートの内容について説明する.
 - レポート用紙をスクリーンに映す, もしくは, パワーポイント等のプレゼン資料を用いる.
- 時間が余れば, 提出されたレポートの中からいくつかを講義中に(門田が)紹介します.

- 連絡先
 - 門田暁人 akito-m@is.naist.jp
 - B303室, 内線5311