Empirical Approach to Software Engineering

EASE PROJECT

見積と品質

～見積、外れたらどうする？ 外さないためにどうする？～

SEA Forum @ 全国情報サービス産業厚生年金会館 7 階会議室

大杉 直樹

国立大学法人奈良先端科学技術大学院大学

 2006年12月14日

見積とは

EASE PROJECT

- あらかじめ、大体の計算をすること。
  - Shin Meikai Kokugo Dictionary, 5th edition © Sanseido Co., Ltd. 1972,1974,1981,1989,1997.
  - ソフトウェア開発に係る見積の例
    - 工数見積：開発に要する工数（人月、コスト）を見積もる。
    - 工期見積：開発に要する時間を見積もる。
    - 品質見積：成果物の品質（潜在的な不具合の数、必要テストケース数）を見積もる。

工数（人月、コスト）

工期

品質（不具合数、テストケース数）

見積と品質の関係

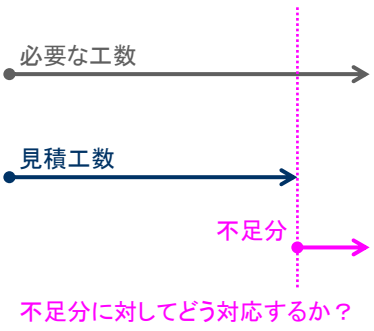
EASE PROJECT

- 見積が甘いと、品質が低下しやすい。
  - 工数が足りなかったので、止むを得ずテストの一部を省略した。
  - 納期を守るため、不具合を残したままとりあえず納品した。
  - どう考えても時間が足りないので、納品時に客先でテストした。

どうすればよいか

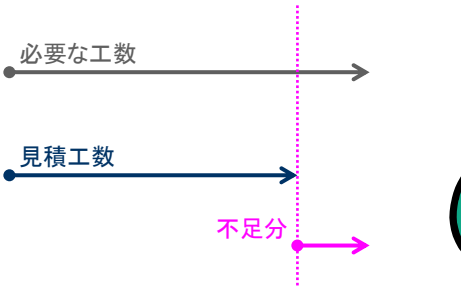
EASE PROJECT

- 見積が外れたときの対応策について予め検討しておく。
  - 雲行きが怪しくなりそうなら、対応策をステークホルダ間（例えば、ラインと QA 担当）で相談する。
- できるだけ正確に見積もる。



見積が外れたときの対応策を検討

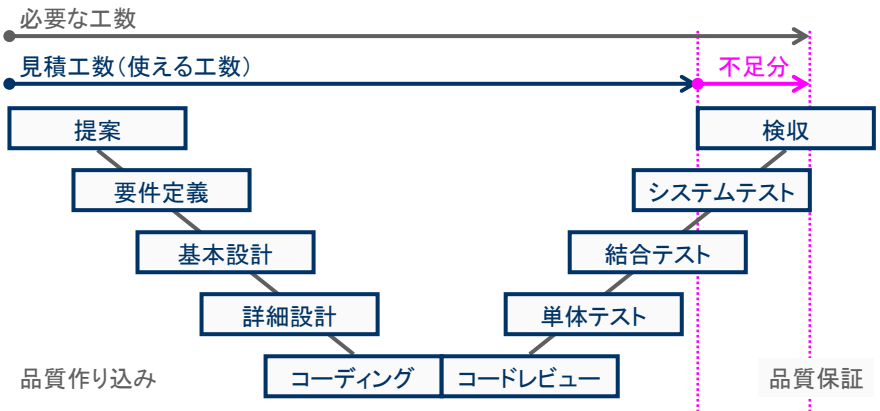
- 追加工数を入れる(発注者もち/受注者もち/従業員もち)。
- 開発のスコープを狭める。
- 品質の低下に目をつぶる。



てっぺんの星をつけないことにしよう
(発注者をうまく説得しないといけないが...)

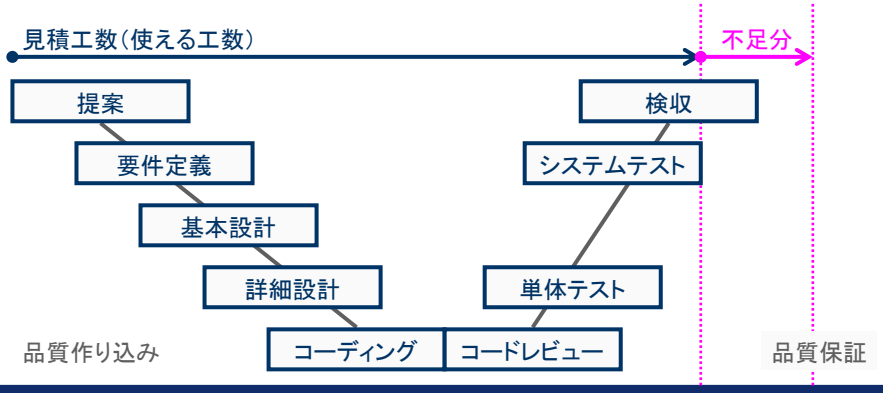
どうやって品質を低下させるか

- 品質保証の一部を省略する。
- 品質保証に費やす工数を圧縮する。
- とりあえず渡してしまう。



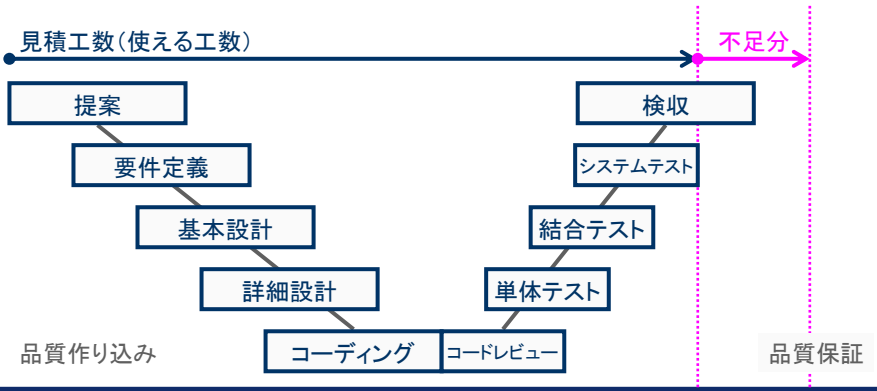
省略(例)
結合テストを省略

- 利点:結合テストの一部をシステムテストで代替できる。
- 欠点:不具合除去に必要な時間が増える。
  - とりわけ原因箇所の調査に要する時間が増える。



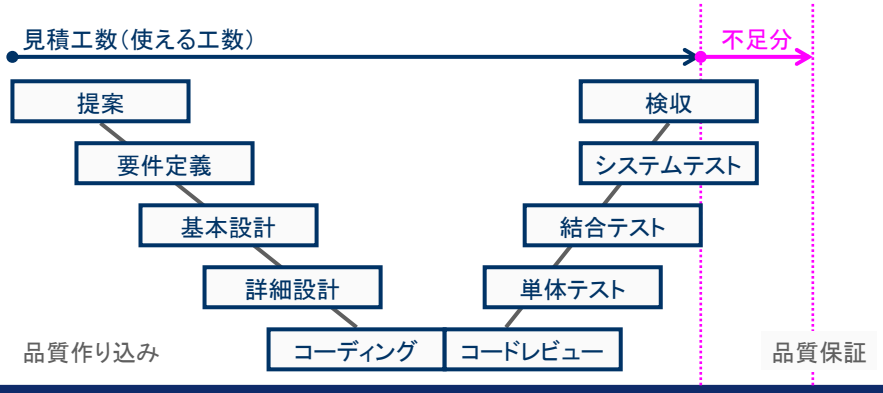
工数圧縮(例)
システムテストまでを全体的に圧縮

- 利点:テストを効率化できれば品質低下を抑えられる。
- 欠点:効率化に失敗すると多くの/重大な不具合が残る。
  - 効率化の戦略を熟考する必要がある。



とりあえず渡してしまう(例)
検収の裏でシステムテスト

- 利点:最終的には品質低下を抑えられる。
- 欠点:受注者からヒンシュクを買う可能性がある。
  - 検収で多くの/重大な不具合が出てしまう。

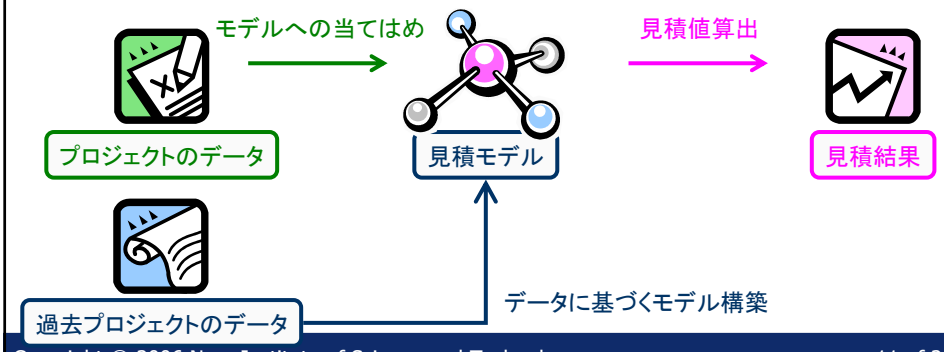


できるだけ正確に見積もるには

- 複数のエキスパートが見積もる。
  - (例)3名以上で個別に見積を出し、最終的な見積やリスク(見積のばらつき)を検討する枠組みをつくる。
- 見積をレビューする。
  - (例)リスク管理の一環として、見積や計画の妥当性を検証する会議を設置する。
- 定量的データに基づいて見積もる。
  - 次頁以降で説明。

定量的データに基づく見積

- 定義済モデルに基づく見積
  - COCOMO、COCOMO II、Agile COCOMO…
- 過去プロジェクトのデータに基づく見積
  - モデルベース手法（重回帰分析、機械学習、CoBRA、…）
  - メモリベース手法（事例ベース推論、EASE:CF法、OSR、…）



見積に用いるデータ

- ソフトウェア開発の特徴を表すデータ項目を、各プロジェクトについて収集したもの。
  - データ項目の例
    - 終了した工程に費やした工数、工期、要員数、要員のスキル、用いた技術／言語、プロセスモデル、用いたツール、ファンクションポイント、コード行数、レビュー指摘件数、テスト項目数、母体システムのファンクションポイント／品質、顧客の知識／協力姿勢、要件の安定性、…

	プログラム言語	開発種別	概算 FP	…	データ項目 *n*
プロジェクト 1	Java	新規	1500	…	値 1-*n*
プロジェクト 2	Java	新規	2500	…	値 2-*n*
…	…	…	…	…	…
プロジェクト *m*	COBOL	保守	3000	…	値 *m*-*n*

モデルベース手法：重回帰分析、機械学習、CoBRA、…

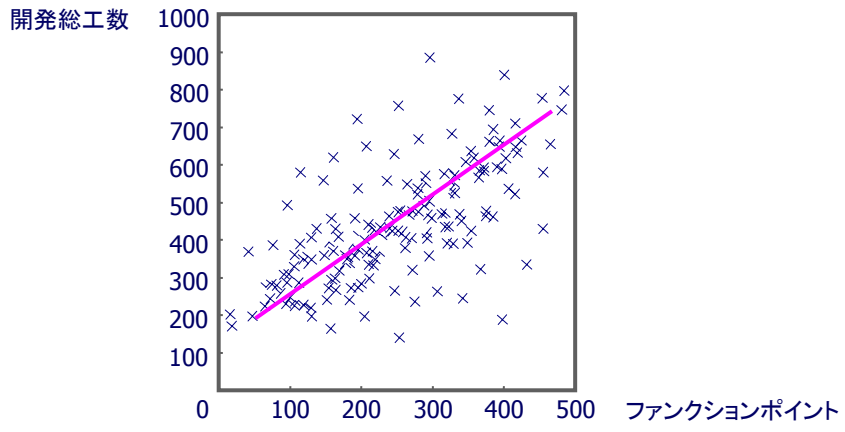
- 過去のデータから重回帰分析によって見積モデル式を構築する。
- 現行プロジェクトの実績を式に代入して工数を見積もる。

見積モデル式: 開発総工数 = 177 × 開発言語 + 0.012 × 概算 FP + 23

	開発言語	開発種別	概算 FP	要員数	開発総工数
現行プロジェクト X	Java	新規	1500	10	見積値：402.5
過去プロジェクト 1	Java	新規	欠損値	8	400
過去プロジェクト 2	Java	欠損値	2500	6	350
過去プロジェクト 3	欠損値	保守	3000	20	2500

モデルベース手法の問題（1）

- 多様なソフトウェア開発プロジェクトを、ひとつのモデルで表すことが難しい。



モデルベース手法の問題(2)

EASE PROJECT

- 欠損値(未記録の値)に弱い手法が多い。
  - 部署間で収集するデータが異なるため、欠損値が多くなりがち。
  - 開発過程で得られるデータの多くは、取り直しがきかない。

		項目 1	項目 2	項目 3	・・・	項目 490
部署 A	部署 A プロジェクト 1	値 1-1	値 1-2	欠損値	・・・	欠損値
	部署 A プロジェクト 2	値 2-1	値 2-2	欠損値	・・・	欠損値
	部署 A プロジェクト 3	値 3-1	値 3-2	欠損値	・・・	欠損値
部署 B	部署 B プロジェクト 4	欠損値	値 4-2	値 4-3	・・・	欠損値
	部署 B プロジェクト 5	欠損値	値 5-2	値 5-3	・・・	欠損値
	部署 B プロジェクト 6	欠損値	値 6-2	値 6-3	・・・	欠損値
部署 C	部署 C プロジェクト 7	欠損値	欠損値	値 7-3	・・・	値 7-490
	部署 C プロジェクト 8	欠損値	欠損値	値 8-3	・・・	値 8-490
	部署 Cプロジェクト 9	欠損値	欠損値	値 9-3	・・・	値 9-490

メモリベース手法(例):EASE:CF法(協調フィルタリング)

EASE PROJECT

- Amazon 社の書籍推薦システム(www.amazon.com)
  - 各ユーザが、読み終えた書籍を 5(好き)～1(嫌い)の 5段階で評価する。
  - システムが、好みの傾向が似たユーザを探し出し、そのユーザが高く評価した書籍を推薦する。

Amazon 社のオススメ書籍予測

- ステップ1: 類似度計算
  - 推薦対象ユーザと他ユーザ間の類似度を計算する。
  - 類似度の高い k (例えば $k = 2$)人のユーザを選ぶ。
- ステップ2: 予測値計算
  - 類似ユーザの評価を加重平均し、推薦対象ユーザの評価を予測する。

	書籍 1	書籍 2	書籍 3	書籍 4	書籍 5
推薦対象ユーザ X	5:大好き	4:好き	2:嫌い	1:大嫌い	予測値:4.53
類似度:+1.0 1	4:好き	4:好き	欠損値	1:大嫌い	5:大好き
類似度:+0.9 2	5:大好き	欠損値	2:嫌い	1:大嫌い	4:好き
類似度:−1.0 3	欠損値	1:大嫌い	4:好き	5:大好き	1:大嫌い

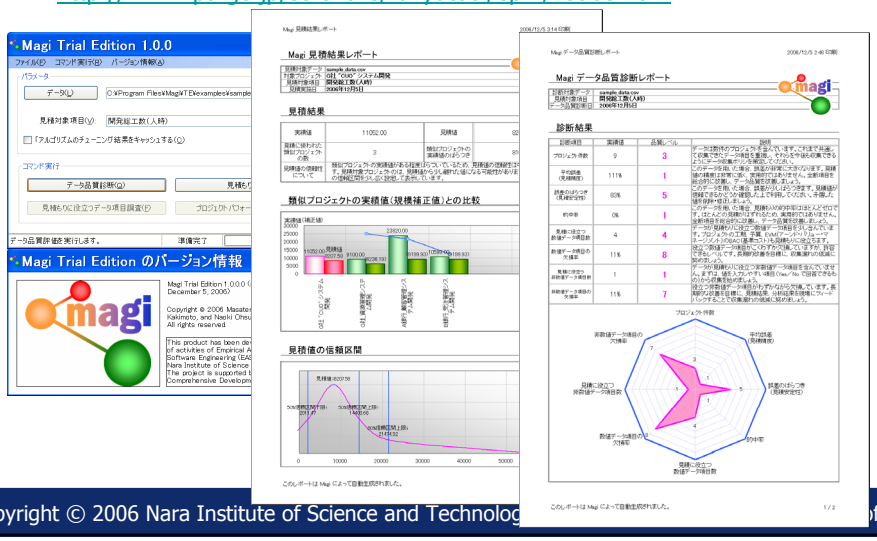
協調フィルタリングによる工数見積

- ステップ1: 類似度計算
  - 現行プロジェクトと過去プロジェクト間の類似度を計算する。
  - 類似度の高い k (例えば $k = 2$)個のプロジェクトを選ぶ。
- ステップ2: 予測値計算
  - 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工数を予測する。

	開発言語	開発種別	概算 FP	要員数	開発総工数
現行プロジェクト X	Java	新規	1500	10	予測値:402.5
類似度:+1.0 エクト 1	Java	新規	欠損値	8	400
類似度:+0.9 エクト 2	Java	欠損値	2500	6	350
類似度:−1.0 エクト 3	欠損値	保守	3000	20	2500

Magi Trial Edition：ワンクリック見積もり&データ品質診断ツール

- EPM ツール（下記）に同梱の形で配布予定です。
  - http://www.ipa.go.jp/software/fukyutool/epm/koubo.html



まとめ

- 見積、外れたらどうする？
  - 追加工数を入れる。
  - 開発のスコープを狭める。
  - 品質の低下に目をつぶる。
    - どうやって品質を低下させるかについても検討しておく。
- 外れないためにどうする？
  - 複数のエキスパートが見積もる。
  - 見積をレビューする。
  - 定量的データに基づいて見積もる。
    - モデルベース手法、メモリベース手法（Magi Trial Edition もよろしくお願いします）
- 本日の発表内容について，ご意見，ご質問ございましたら，下記までご遠慮なくお問い合わせください．
  - 大杉，naoki-o@is.naist.jp，0743-72-5318（研究室代表）

