

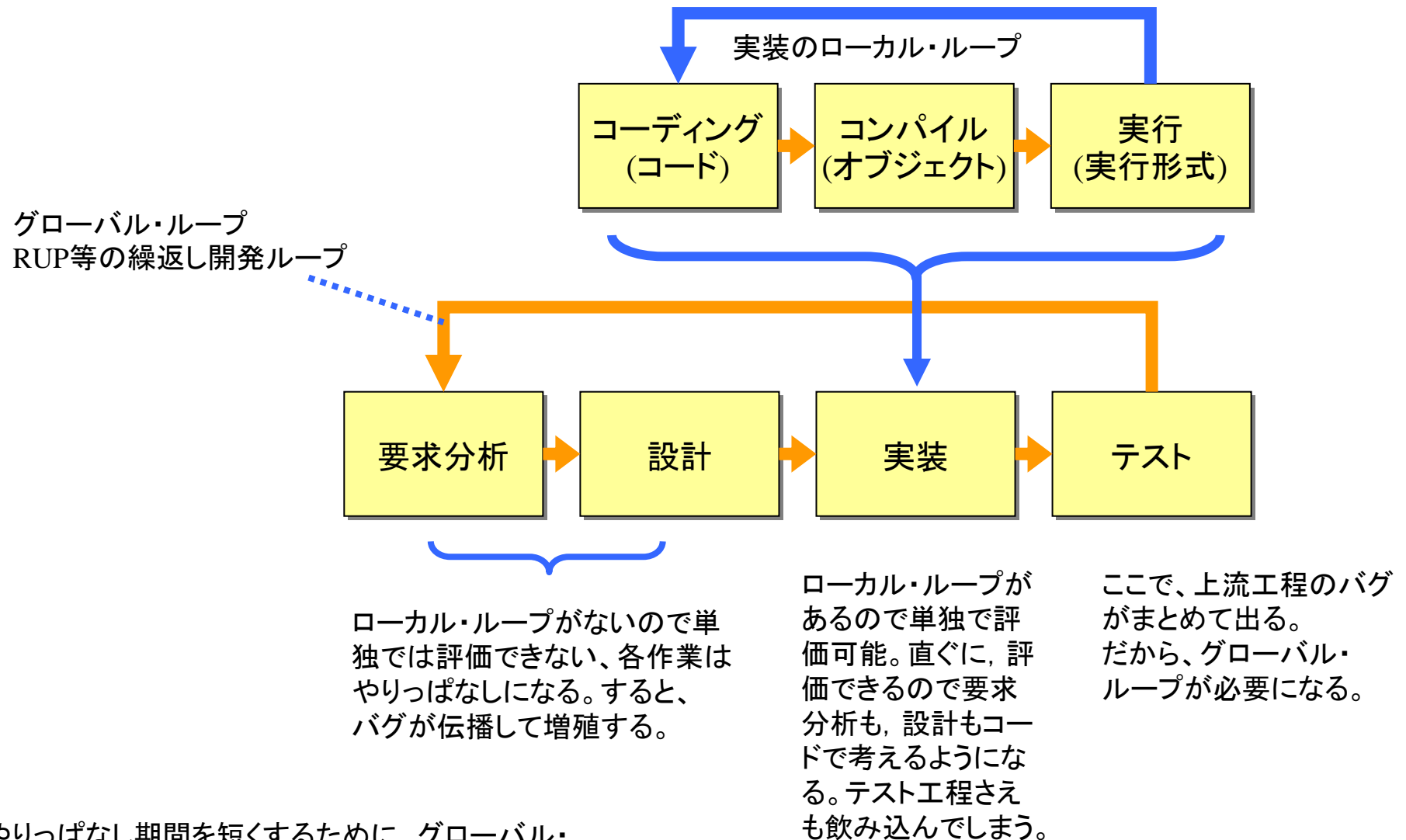
形式手法の概要と使い方 ソフトウェア品質の確保を目ざして

2006/11/17
藤倉俊幸

目次

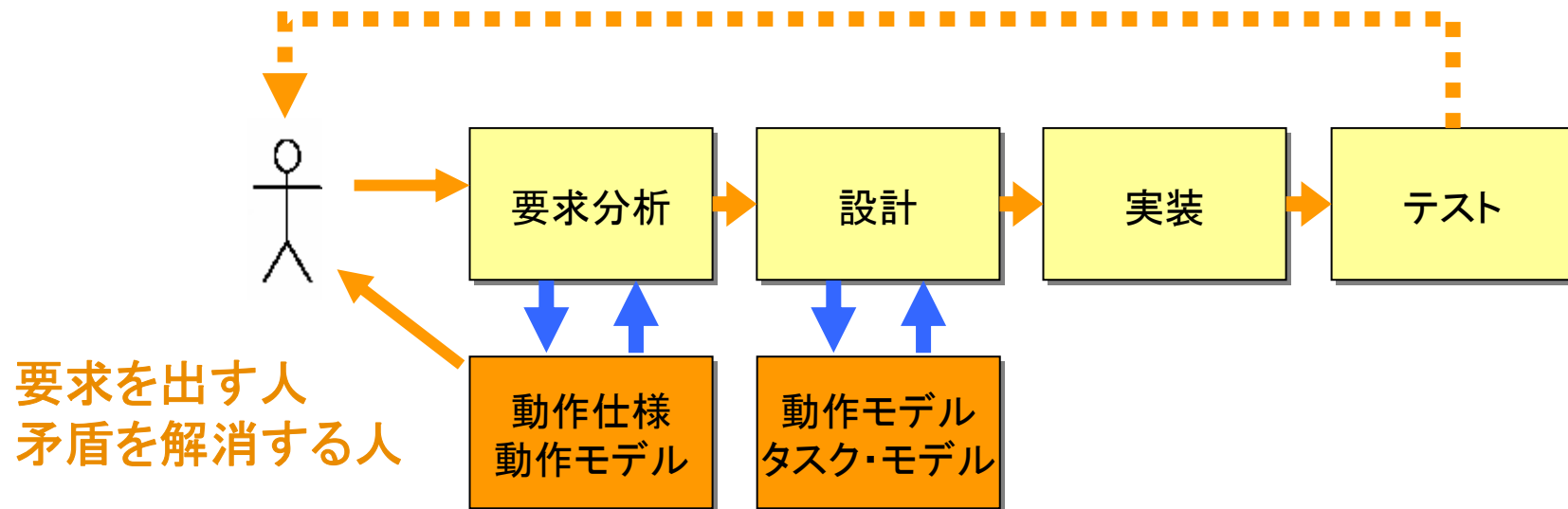
- イン트로ダクション
 - ❖ 形式手法とは何か
- 形式手法の使用前・使用後
 - ❖ 開発時
 - ❖ 開発後 (不具合対応等)
- どうすると使えるのか
 - ❖ 何が必要か
 - ❖ 本当に使えるのか
- まとめ

形式手法使用前の開発工程



やりっぱなし期間を短くするために、グローバル・ループの管理が必要になる。

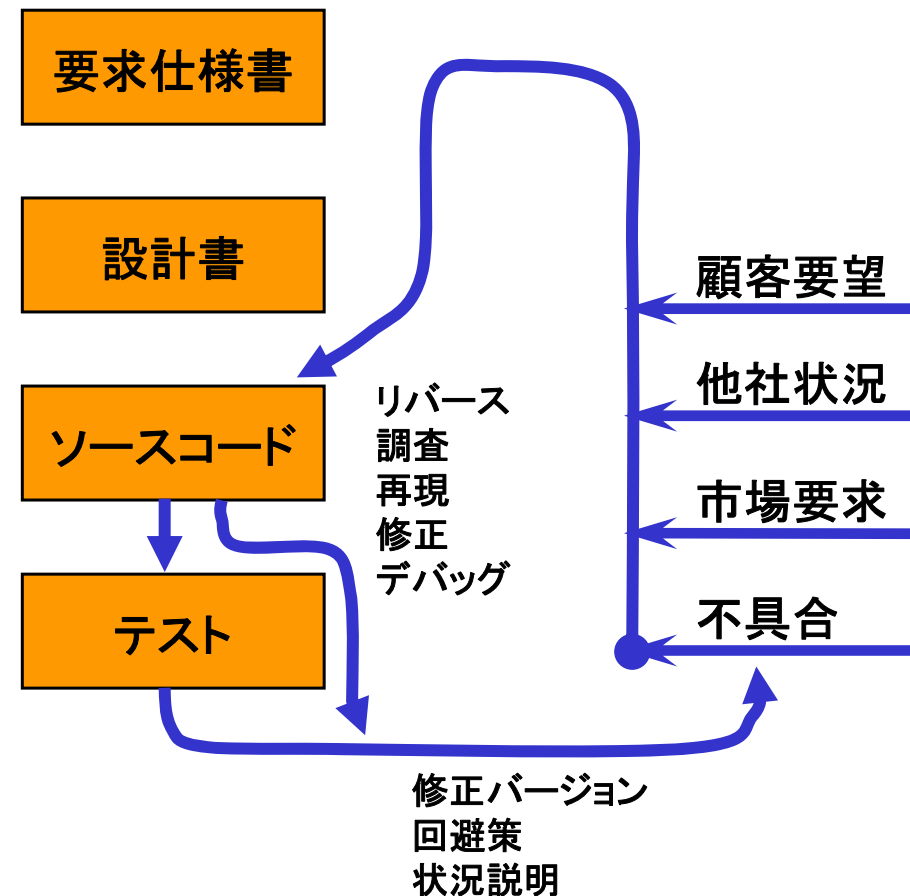
形式手法使用後の開発工程



- 形式手法を導入することによりフィードバックがかかるようになる
 - ❖ 繰り返し回数を少なく、繰り返し期間を長く取ることが可能になる
 - ❖ 上位からの入力(仕様追加)に対して独自に作業を開始できる
 - ❖ 技術として閉じるので、それぞれの作業の専門家が育つ
 - ❖ 作業見積もり精度の向上
- 要求を出す人に直ぐフィードバックできる
 - ❖ 要求自身の開発, 要求開発に通じる

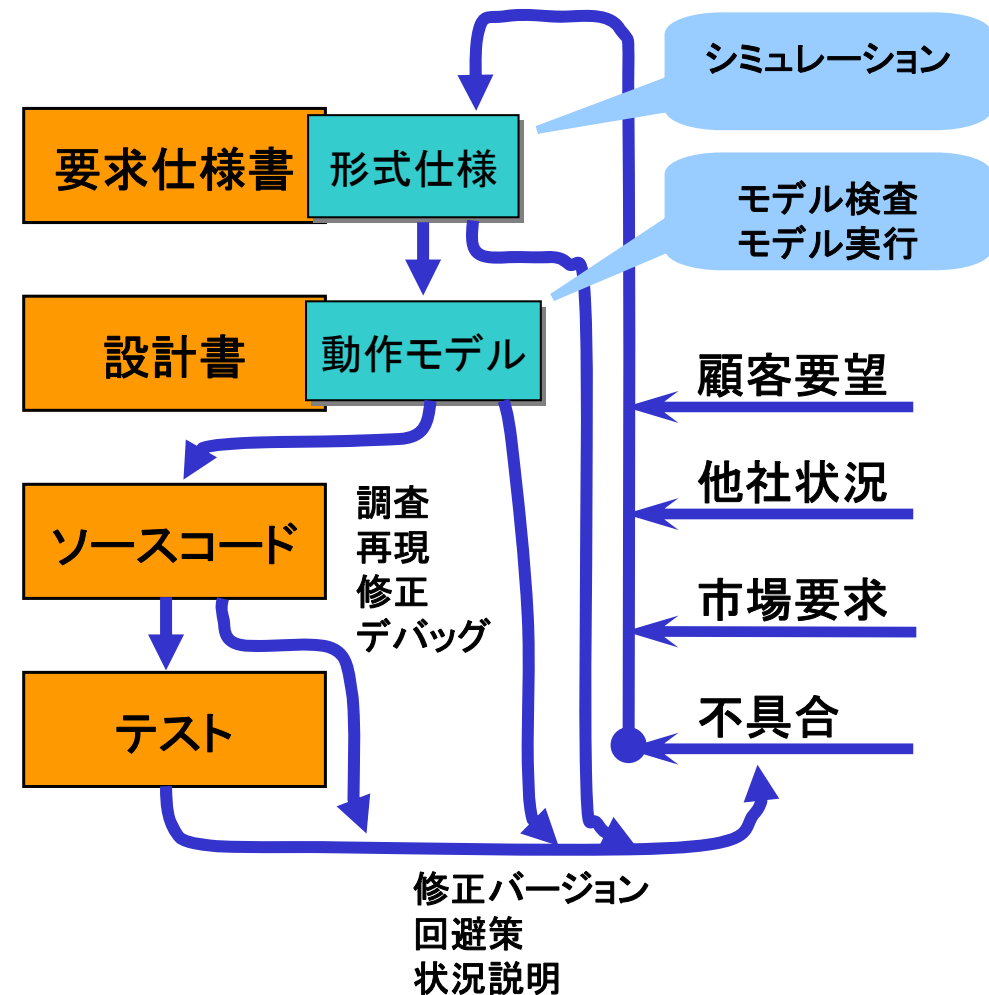
形式手法使用前の変更対応

- 開発終了後の変更要求に対して上流成果物は無力
 - ❖ 仕様変更なのか、設計変更なのか、バグなのか全てはソースコードで判断する
 - ▶ 自然言語の解釈は曖昧
 - ▶ メンテナンスされていない
- 実装担当者の負担は増え続けるだけ
- 実装担当者が何時までも居るとは限らない
- 実行環境が何時までもあるとは限らない



形式手法使用後の変更対応

- 上流成果物を有効に利用できる
 - ❖ 動く上流成果物
 - ❖ メンテナンスは単なるコストではなくなる
- 仕様の問題か、設計の問題か、判断することができる
 - ❖ 早いレスポンス
 - ❖ 見積もり精度の向上
 - ❖ 影響範囲の分析精度の向上





イーソル株式会社

Research & Consultation services

CONTACT:

Office : 03-5365-1560 FAX : 03-5365-1561

E-Mail : Consult@eSOL.co.jp

URL: <http://www.esol.co.jp/solution/consult/index.html>