

実践！品質V字回復

～キーパーソンに頼らない
仕組み化によるチーム全体の品質向上～

株式会社オープトーン
金融ソリューション事業部
部長 畑中 貴之

自己紹介

■あなたは誰？

- 畑中 貴之 (@hatanaka_ot)
- 株式会社オープトーン
- 金融ソリューション事業部 部長

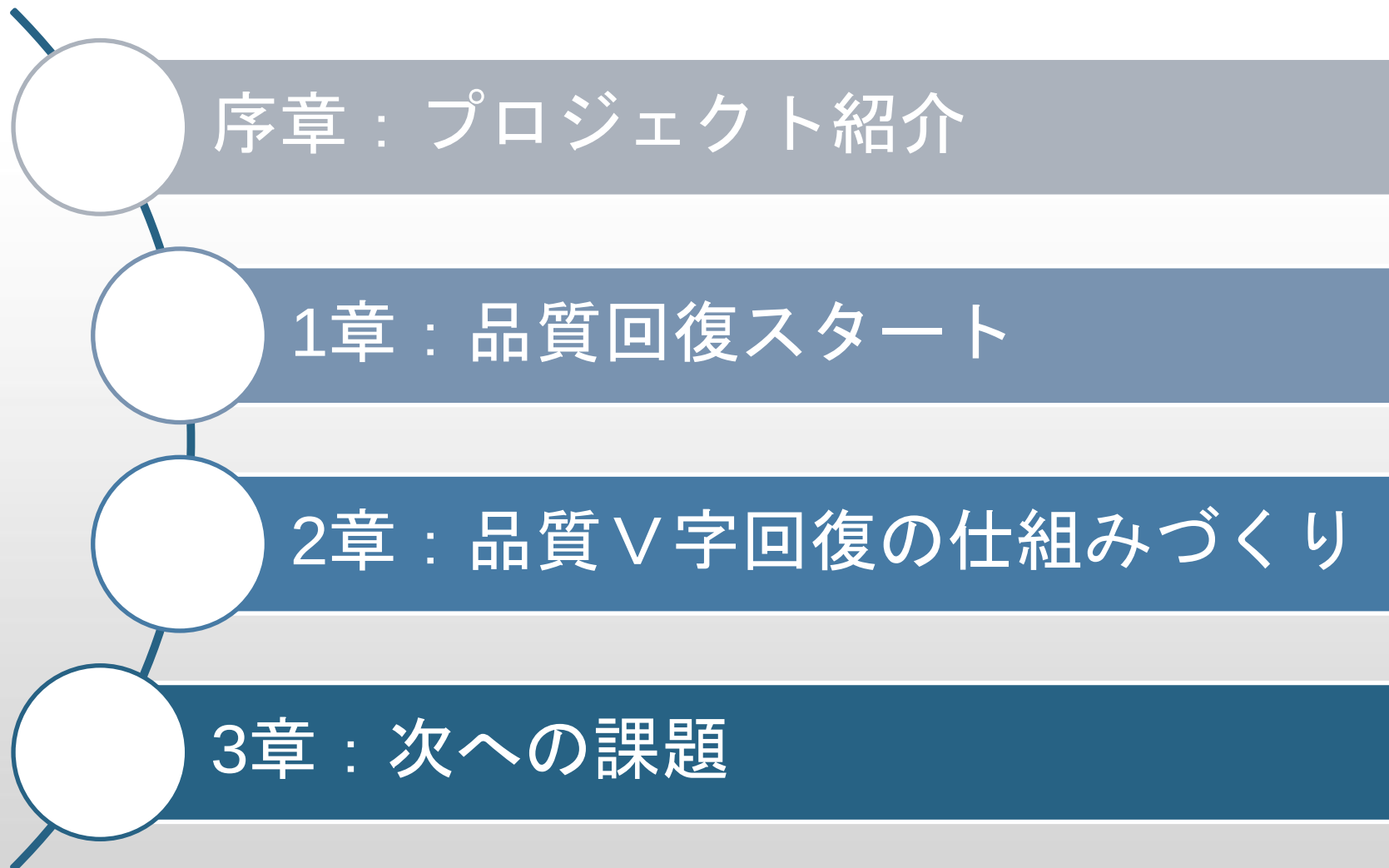
■最近の仕事は？

- ネットバンキングのシステム開発チーム（15名体制）のマネージャ

■最近の関心ごとは？

- EVMによるプロジェクト進捗管理
- 品質と効率のバランス
- 仕様変更とコストの関係

アジェンダ



プロジェクト紹介

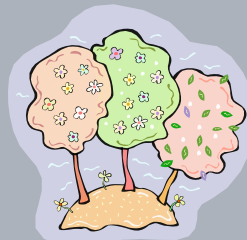
項目	内容
対象システム	ネットバンキングの口座開設、口座管理システム JavaのWebアプリケーション開発
開発プロセス	ウォーター・フォール型
チーム体制	15名を4-5名のサブチームに分割 1,2年目のエンジニアから10年以上のベテランエンジニアまで幅広く
特徴	複数のプロジェクトが同時進行。多い時には大小含めて10以上。 要求提示後もしばしば仕様追加・変更が発生
コスト・リソース	期初に計画、その範囲内で基本的に実施
リリース	3, 4か月に1回

Chapter 1

品質回復のスタート

きっかけとなった出来事

2009年 春リリース



オンライン・バッチ機能
障害発生



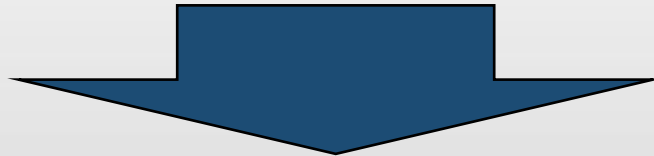
チームに品質に対する危機感



品質を上げるにはどうすればよい？

■ 品質を上げるためにはどうすれば・・・

- お金と時間をかければよい
- できる人を投入すればよい
- ツールを使って効率化すればよい
- エンジニア（開発チーム）の意識を上げればよい
- お客様の仕様変更は受け付けなければよい
- すべての作業を自動化して、コンピュータにやらせればよい



予算・時間・リソースに制限があり、
仕様変更もそれなりに発生する現場

対象を明確にする

ソフトウェア

- 開発成果物に対する品質

プロセス

- 開発手順に対する品質

組織・チーム

- 組織・チームの姿勢・文化

取り組み内容を決める

1.やり方を決める

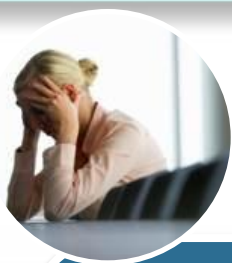
2.障害を埋め込まない

3.障害を検出する

4.ミスを防ぐ

5.常に改善する

どう変わったのか？（Before / After）



■Before

- ・ 品質は特定の人気づきと献身
- ・ リリース直前まで障害対応
- ・ リリースごとに障害？



チーム全体によるプロセス改善へ！！



■After

- ・ 品質は決められたプロセス、手順による担保
- ・ リリース前に最終品質確認
- ・ 長期間にわたって本番障害ゼロ！！

Chapter 2

品質V字回復の仕組みづくり

1.やり方を決める

■ プロセスの策定

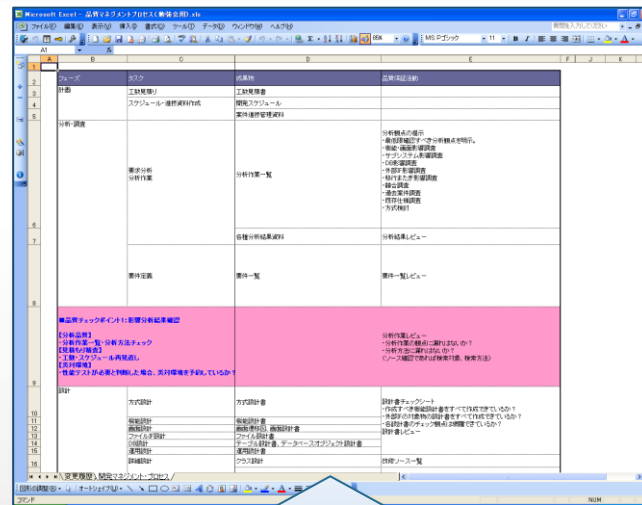
- フェーズごとのプロセス・手順を明記
- プロセス・手順で必要な成果物、品質活動を定義
- 「品質マネジメントプロセス」

■ ルールの作成

- プロセス・手順に関するルールをWikiに集約
- 現場の開発・業務ルールも集約
- 品質を上げるための独自のルールも作成・集約

■ 主なルール（他にもいっぱいあるよ）

- ・ プロセス・手順
- ・ 進捗基準
- ・ 課題管理、障害対応
- ・ 役割分担表
- ・ 業務ルール



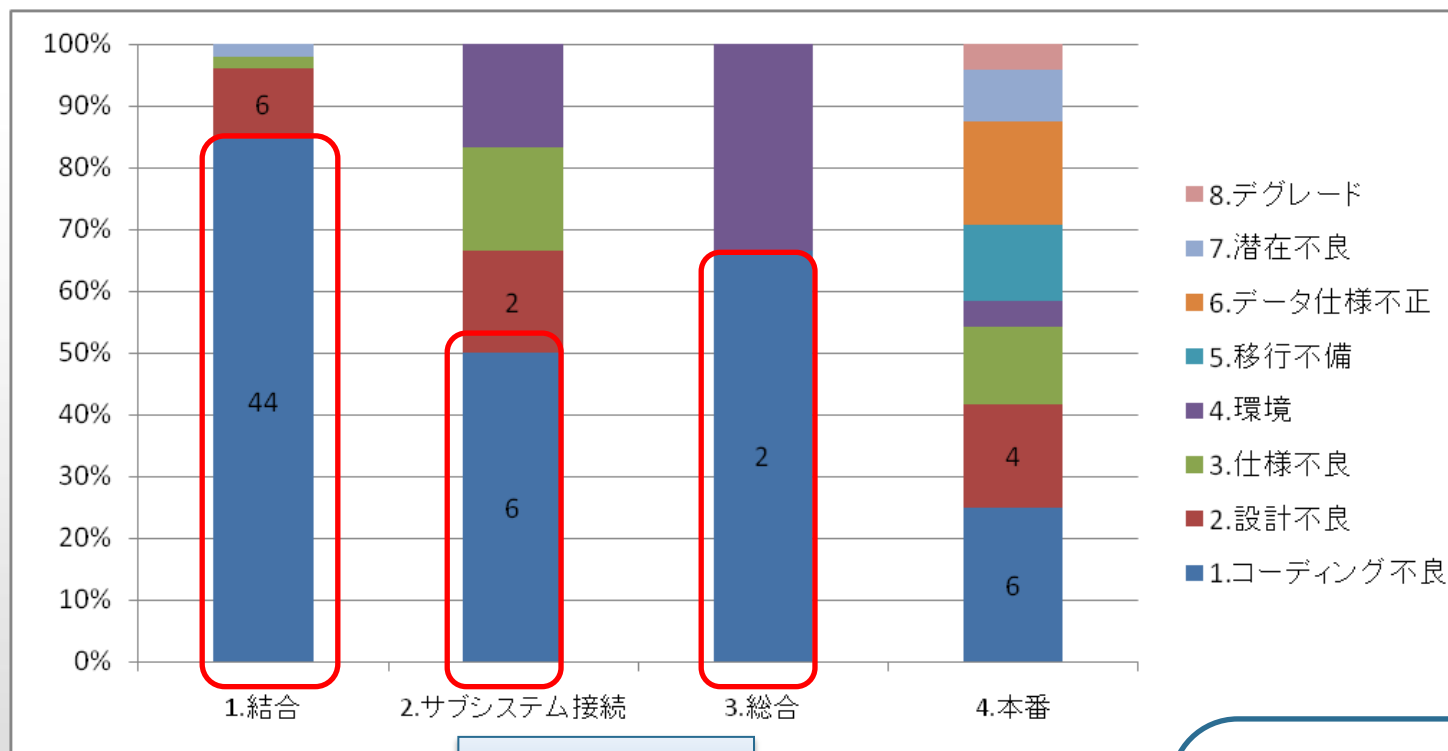
フェーズ	プロセス	品質活動
1. 企画	企画書作成	企画書の作成・承認
2. 分析	分析・調査	分析・調査の実施・報告
3. 設計	設計書作成	設計書の作成・承認
4. 開発	開発作業	開発作業の実施・報告
5. 検証	検証作業	検証作業の実施・報告
6. 納品	納品作業	納品作業の実施・報告

連携



2.障害を埋め込まない

■ 障害の原因分析⇒原因発生を防止する



設計・実装工程の品質改善

- 設計
 - ・ 自己チェックレビュー
- 実装
 - ・ 自己チェックレビュー

3.障害を検出する

■ 検出するためのテストケースの洗い出し方は？

- 設計書からパターン抽出
- 過去の類似機能のテストから作成

■ どんなテストケースが考えられるか？

- 入力チェック、入力値変換
- 検索条件の仕様
- 検索件数
- 検索結果の表示内容
- 最大表示件数（ページング）
- 文字化け、セキュリティ、二度押し制御

■ テストの観点マスタ作成

- 「必要なのは何か？」ではなく、「どれをするか？」
- 設計時も参照

The screenshot shows a web browser window with the address bar displaying 'file:///C:/Users/thatanaka/Desktop/検索画面.html'. The page content is as follows:

検索画面

検索条件

日付: 2011/5/20
ID:
氏名(漢字): はたなか
氏名(カナ): ハタナカ
住所: 千住1-1-1

検索結果

日付	ID	氏名(漢字)	氏名(カナ)	住所
2011/5/20	111	畑中 貴之	ハタナカ タカユキ	東京都足立区千住1-1-1

4.ミスを防ぐ

品質チェックポイント

- フェーズとフェーズの合間でチェック
- 前工程の作業内容のミスや漏れがなかったか確認

No.	カテゴリ	目的・背景	チェック内容
1	デグレード	品質チェックポイント③以降に発生したリリースの取込漏れがないことを確認する。 実装時に発生したモックとの差分を吸収するため	マスタリポジトリと差分を比較し、直前のリリースの取込み漏れがないことを確認 ※品質チェック③以降、新規に追加もしくは改修が発生したソース(バグ)に対して実施 モック不具合一覧にてクローズされていない事項が無いことを確認する。クローズ事項がある場合はモック担当者へ修正依頼を出す。
2	テスト不測	テスト漏れがないか、また障害内容がすべて確認できているかを確認する。	テスト仕様書を確認し、テスト漏れがないかどうか確認する。 リリース対象の機能に対して、テストのエビデンス検証がすべて完了していることを確認する。
3	作業完了確認	案件の作業の完了条件を満たしていることを確認する。	作業完了報告書は作成されているか？ 設計品質チェックシートは作成かつ捺印されているか？ 製造品質チェックシートは作成かつ捺印されているか？
4	リリースアクション確認	本書時の移行ミスや作業漏れをなくすために実施する。	開発管理票を確認し、リリース前に実施すべきアクションが漏れていないことを確認 (例)運用部への依頼事項、ユーザーへの確認事項

課題票の埋もれ防止

- 課題は最後のアクションまで実施して完全にクローズ
- クローズ済みかはチェックポイントで確認

対応・回答	対応・回答日	対応・回答者	完了日	設計書反映	最終アクション
5/20 佐藤 ユーザの検証テスト中に確認できるよう6月前半までには手配する。	2011/5/20	佐藤	2011/5/20	5/20 畑中	6月10日にリンク確認

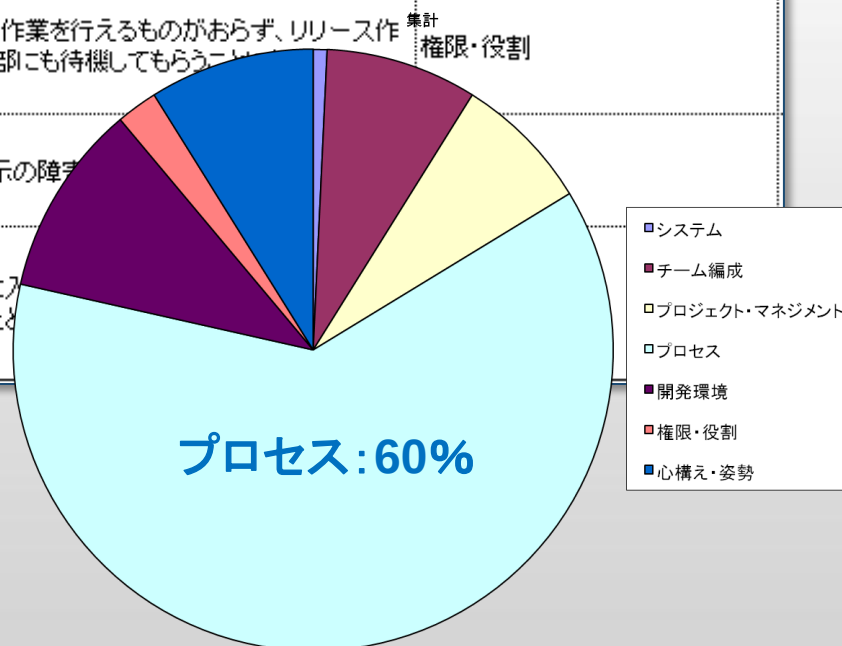
他チームの品質をチェックする

5.常に改善する～その1～

■振り返りポイント

- プロジェクト改善としてリリースごとに振り返る
- 良かったこと、反省点、困ったこと、障害事項

No	種類	内容	対象
1	反省点	〇〇のチェックをすべきところ実施が抜けてしまい、結合テストの障害として検出されてしまった。	プロセス
2	困ったこと	リリース時の作業のときに、構成管理作業を行えるものがおらず、リリース作業が滞ってしまった。結果として運用部にも待機してもらった。	集計 権限・役割
3	障害	シナリオテストフェーズにて、画面表示の障害発生。	
4	良かったこと	フロントシステムの有識者がチームにコードレビュー、結合テストまで困った。	



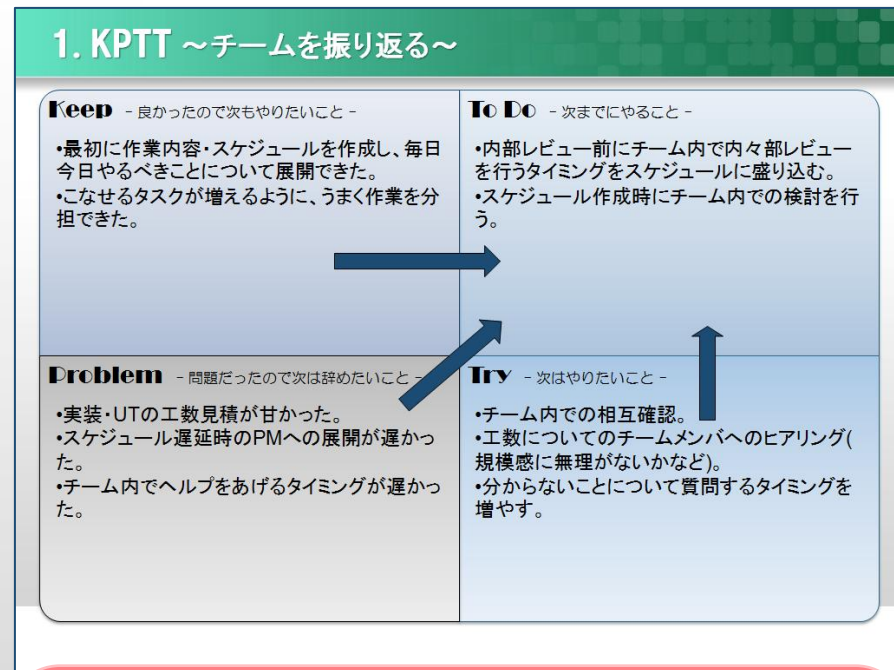
5.常に改善する～その2～

■改善策の発表

- プロジェクト改善MTGの実施、個人 / チームごとに発表
- 発表が参加意識、問題意識を育てる
- KPTTによる対策案の策定

■プロジェクト改善対策の実施

- 対策事項の実施状況をフォロー
- 次回振り返り時に効果測定



■KPT / KPTT (ケプトツ)

- Keep : 次回も実践したい
- Problem : 改善したい問題点
- Try : 次回は挑戦したい
- To DO : 次回までに行うこと

取り組み内容まとめ

1.やり方を決める

- プロセス・手順を明記
- ルールをWikiに集約

2.障害を埋め込まない

- 設計書、コーディングの自己チェック

3.障害を検出する

- テスト観点マスタからテストケースを作成

4.ミスを防止する

- フェーズ間でチェックする
- 課題票で最終アクションを管理する
- 他チームの品質をチェックする

5.常に改善する

- プロジェクトを振り返る
- 改善事項を実施する

それで効果はどうだったのか？

年度	PJ数	規模 (Step)	総コスト (人日)	障害 (接続)	障害 (総合)
2009	39	69,359	2,278	20	24
2010	44	105,227 +30%	1,864 Δ20%	6 Δ70%	10 Δ60%

Chapter 3

次への課題

マネジメント

- 進捗管理
- メンバー育成
- 情報共有

メンバー

- 資料・課題
- レビュー・会議

取り組み状況は「おぶとんらぼ」で

What is おぶとんらぼ？

株式会社オープントーンが
システム開発の現場より皆さんへ発信していく情報サイトです

実際に開発現場で培ったノウハウを

社名の「オープントーン」の名の如く **オープン** に公開していきます！！

できる範囲でね！



Process



Document



Testing Suite



Twitter : @opentonelabs
Facebook: Opentone Labs
<http://labs.opentone.co.jp/>

V字回復を支えたツールたち

■ TestingSuite

人間は単調な反復作業を続けると意識が低下しエラーを起こしやすくなる・・・

- 画面キャプチャツール
 - ブラウザのキャプチャを検証に合わせたファイル名で自動保存
- HTMLフォームの自動入力、改竄ツール
 - 何度も入力するフォームの値を記録しておいて復元
- DBデータ投入ツール
 - EXCELで作成したデータをテストケース毎に投入するツール
 - 事前にデータを作成しておくことで、検証の正当性の確認も可能
- DBエビデンス取得ツール
 - 複数テーブルのデータを一度に取得し、証跡として一括保存できるツール



**テストに必要な反復作業をツールがアシストすることで、
テスト内容に意識の大半を使える！！**

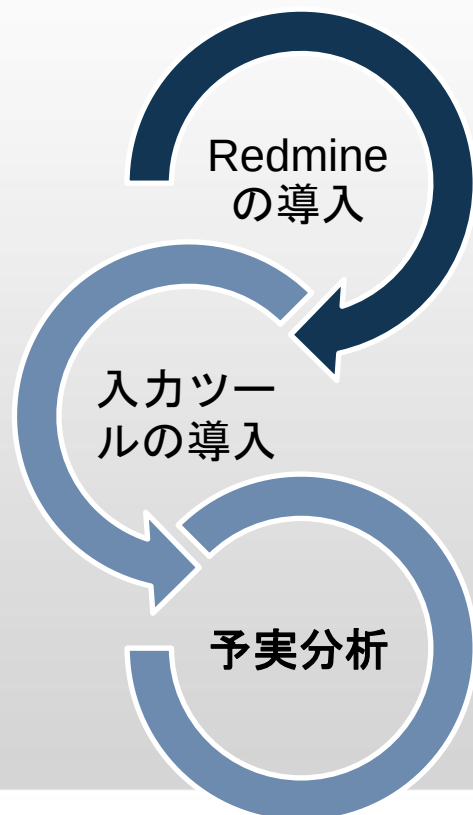


V字回復の計算を可能にしたツールたち

RMC

Redmineの実績入力に特化したWindowsクライアントツールです
タスク管理による入力支援とプロジェクト改善に活躍しています

～ History ～



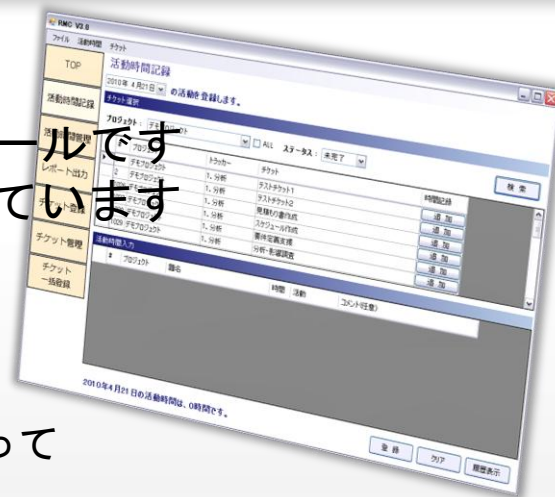
- フェーズごとのタスクを割り出し、予定と実績についてRedmineを使って管理をはじめる

But

入力が面倒なので入力率が悪かった

- 実績入力に特化したクライアント(RMC)を作成
- And**
入力だけに意識を向けることができ、入力率UP↑

- 集めたデータを活用した予実分析が可能に
- That is**
経験や勘に頼らない事実に基づいた進捗管理へ



最後に～品質向上の処方箋～

1. 当たり前のことを当たり前に継続する
2. 全体の流れの中でいつ品質担保する仕組みを入れるか？
3. ツール、自動化、新しい手法はうまく活用しよう！
でもそれは「銀の弾丸」ではない
4. 経営・企画、運用・インフラなど全社一丸で取り組むことで効果は倍増
5. 「品質のために自分にできることは何か？」を考える