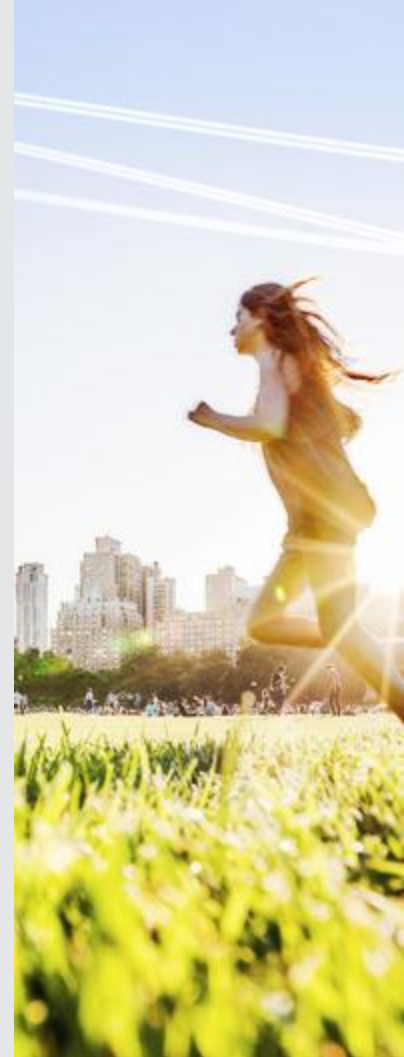


XDDP導入に躊躇する 開発チームへの アシストとは

中村 勝志

アンリツエンジニアリング株式会社
SPIRIT

派生開発カンファレンス2018



会社紹介

アンリツ株式会社

進化を続ける情報通信の分野で、各種通信システムやサービス・アプリケーションの開発、品質保証に欠かせない計測器を提供しています。



有線・無線通信



電子計測器

アンリツインフィビス株式会社

世界中の食品工場や薬品工場の生産、包装工程で異物混入検査のほか、多様な品質検査に対応し、食品や医薬品の安全と安心に貢献しています。



重量選別機



X線検査機

アンリツエンジニアリング株式会社

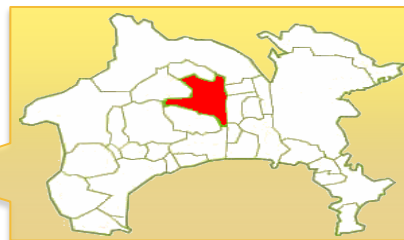
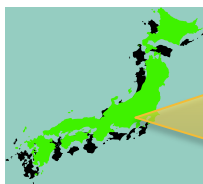
情報関連計測器の電子回路、組込みソフトウェア、アプリケーションおよびシステム開発によりアンリツグループの製品を通し社会に貢献します。



ソフトウェア・ハードウェア開発



所在地 神奈川県厚木市恩名



自己紹介

中村勝志（入社20年目、SE兼PL）

検査機の制御ソフトウェアの要求分析～実装～テストまで担当。機能追加・移植などの派生開発を多く経験。高品質なソフトウェアを提供するため、SPIRITに所属し活動中。
モットーは「新たなチャレンジで、仕事を楽しく！」

SPIRIT（Software Process Improvement/Reform and Innovation Team）

社内のソフトウェア開発プロセスの改善、および新規技術の導入・展開を推進し、アンリツグループ全体のソフトウェア開発力向上に貢献する。
「魂こめてやろうぜ！」の意。合言葉は「定時で帰って、ガッツリ稼ぐ！」

派生開発カンファレンス2012

『現場からの障壁克服

～XDDPを導入する際の障壁と その克服に向けたアプローチ～』



派生開発カンファレンス2014

『「現場からの障壁克服」その後 ～ 生えた根を太くする ～』



派生開発カンファレンス2018

『XDDP導入に躊躇する開発チームへのアシストとは』

目次

1. 仮説→実行→失敗、
そして「ふりかえり」
2. 再チャレンジ
3. 結果は成功
4. まとめ

「アシスト」とは？

サポーター



アシスト



開発チームをヒーローに
応援ではなく、ゴールをお膳立て

仮説→実行→失敗、
そして「ふりかえり」

「改善に躊躇」なぜ？

開発チームはつらいよ！

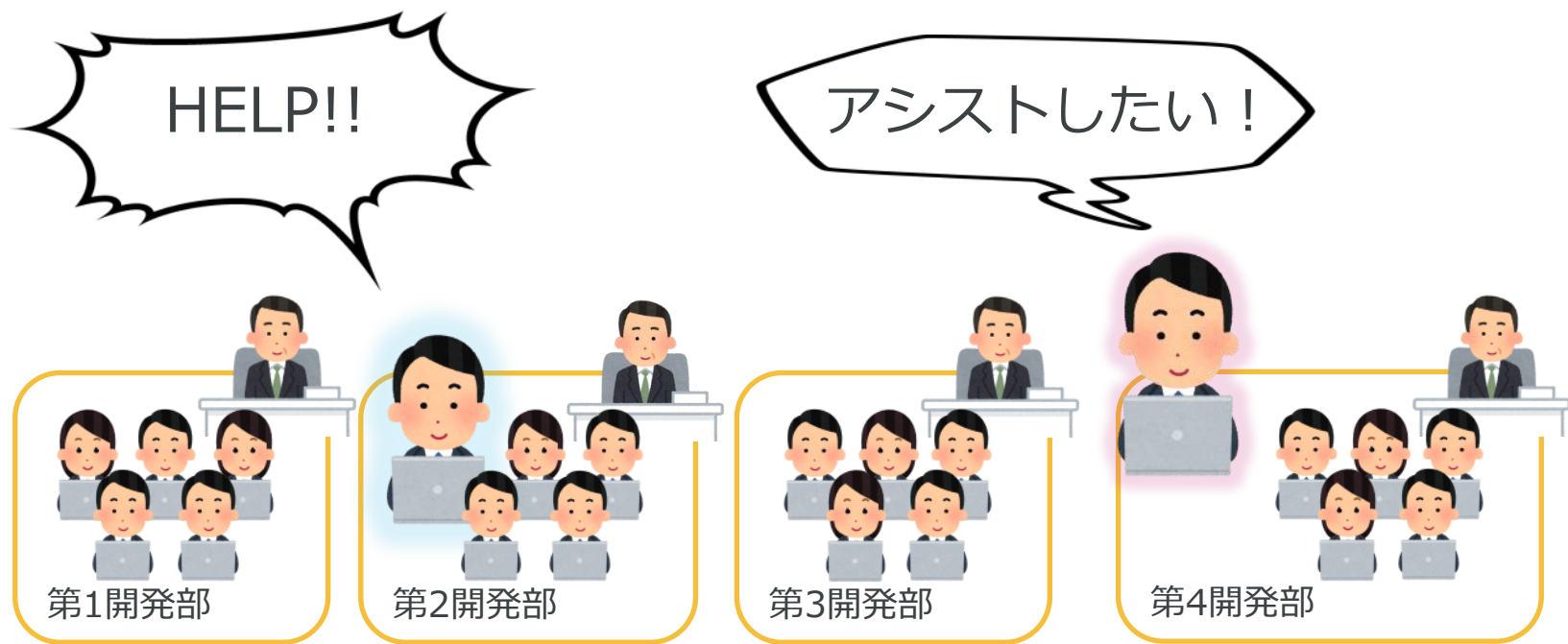


「失敗したら誰の責任？」

成功して当たり前、失敗したら開発チームのせい

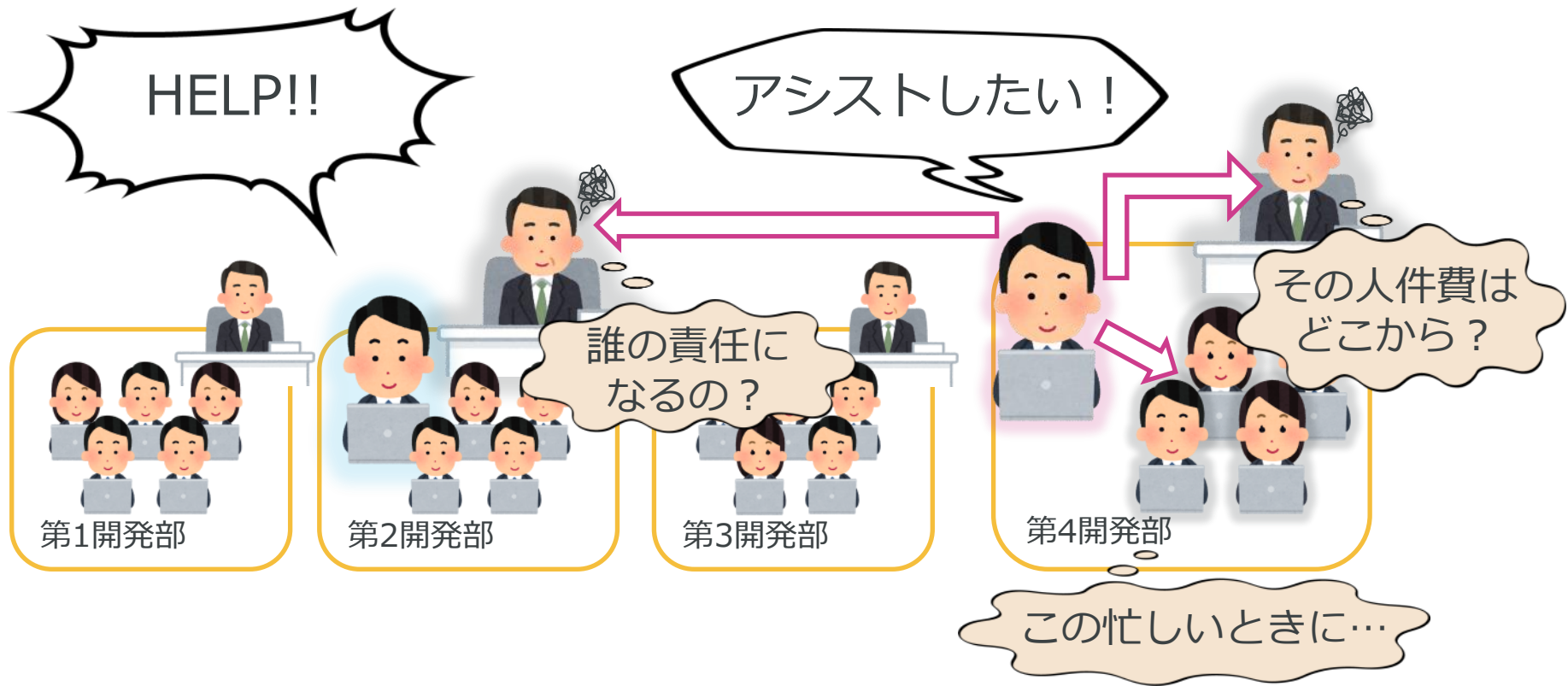
組織の壁

関係者を個別に説得するのは大変



組織の壁

関係者を個別に説得するのは大変



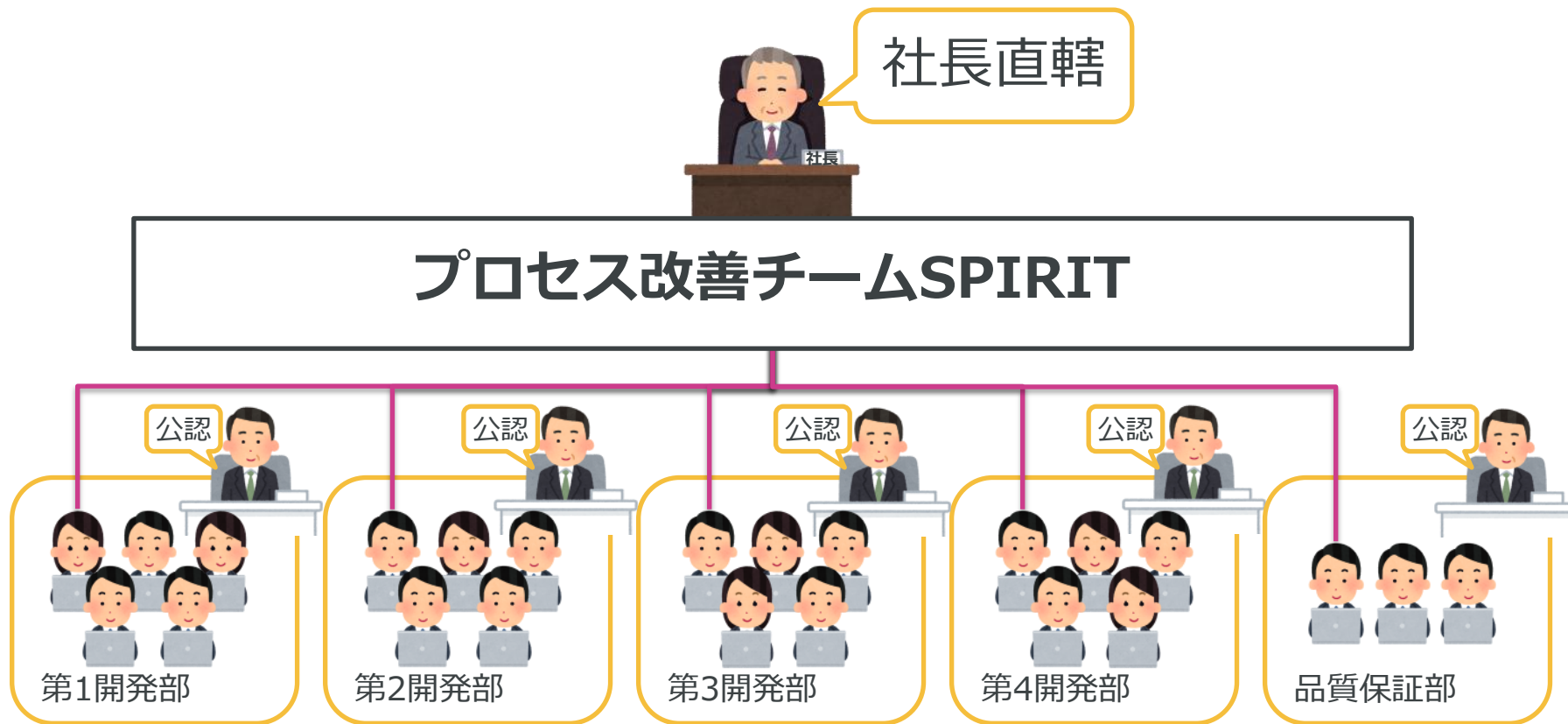
先ずは「安全・安心」

~~開発チーム~~管理職はつらいよ



兎にも角にも**安全を確保** 気分はアゲアゲで

安全を「仕組み」で確保



夕テヨコを巻き込んだ組織を設立

アシスト開始



「アシスト」失敗

話を聞いてもらえなかった



こうすれば
良いのに

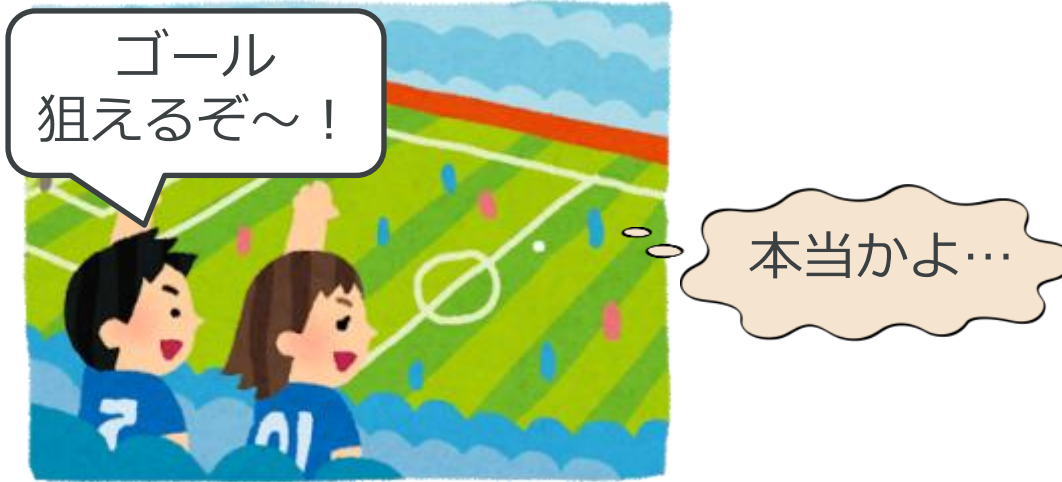


名ばかり「アシスト」

外野からあれこれ言うだけのサポーターだった

開発チームと「同じ立場」で考える

サポーターの言葉は届きにくい



同じフィールドにいる仲間なら
安心できる



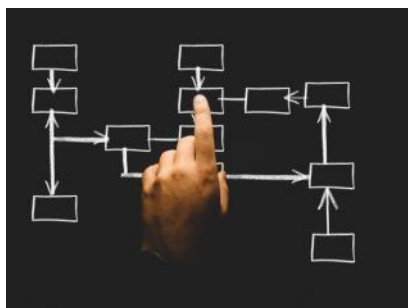
「アシストの行動」とは



- 開発チームの上長への説明と協力の確約を得る



- 開発チームのやる気を向上させる
 - ✓ 欠点を見つけてもまずは肯定する
 - ✓ 詰問ではなく質問する
 - ✓ 最後は提案し合意する



- 開発チームの困っているところを見つける
 - ✓ 現場の様子を知る
 - ✓ 現物の出来栄を見る
 - ✓ 現実を映し出す

行動を褒める

再チャレンジ

発表事例の背景①

どんなプロジェクトか

- 基本機種を改造する「派生開発」
- 要求管理が原因で
実機テストで**重大な**不具合が複数検出されている



事前の作戦

- 従来の要件管理方法にUSDMの考え方を取り入れ
要求漏れをなくす
- 要求確定以降の開発はXDDPの標準的な方法を
順次取り入れていく

発表事例の背景②

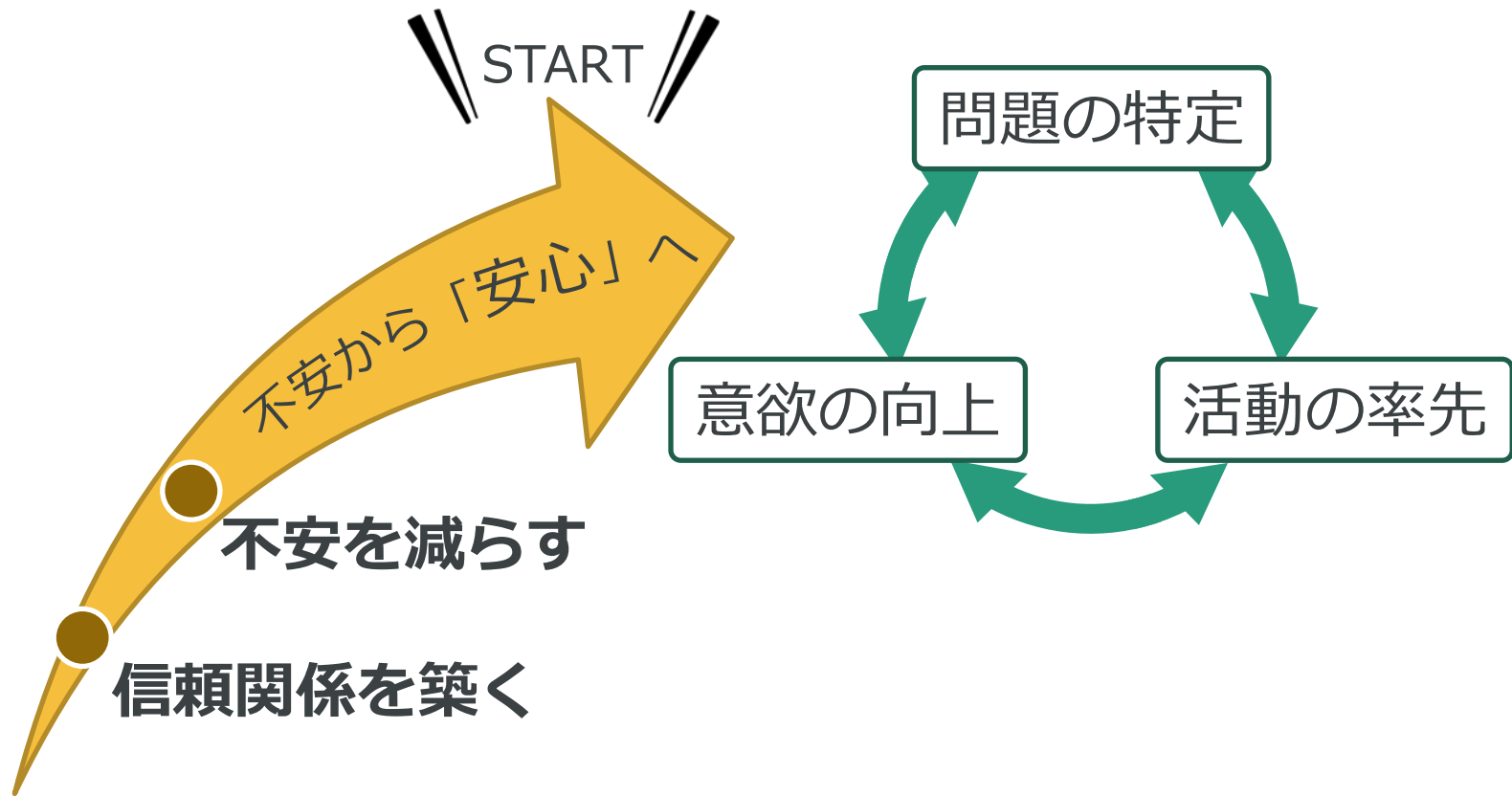
アシストはチームで実施



三人寄れば文殊の知恵

即応できる体制で開発チームも安心

やったこと



先ず、信頼関係を築く



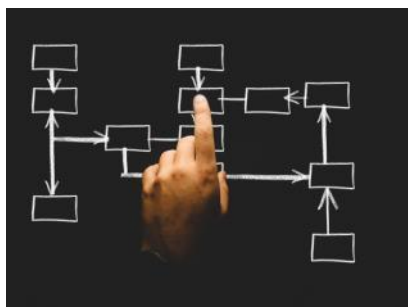
● 全員参加でキックオフ

- ✓ 開発チームの上長
- ✓ 技術リーダー
- ✓ 開発メンバー



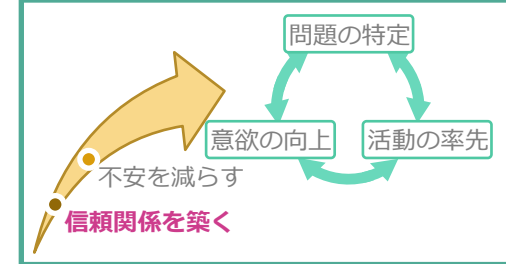
● うまくいっているプロセスはいじらない

- ✓ 設計ミスとプログラミングミスは少ないので
TMと変更設計書は導入しない
- ✓ テストも今までどおり



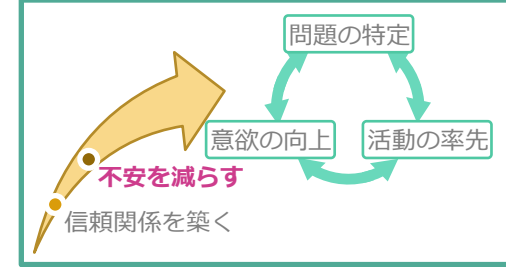
● 開発チームの不安に耳を傾ける

- ✓ 答えでなく不安を探す
- ✓ 感情移入はしない



チームに信頼されてこそそのアシスト

次に、不安を減らす



どうすればいいか、わからない…

➤ PFDを使った問題の特定方法を提案



開発だけで、もう手一杯だよ…

➤ アシストが率先して改善活動を駆動



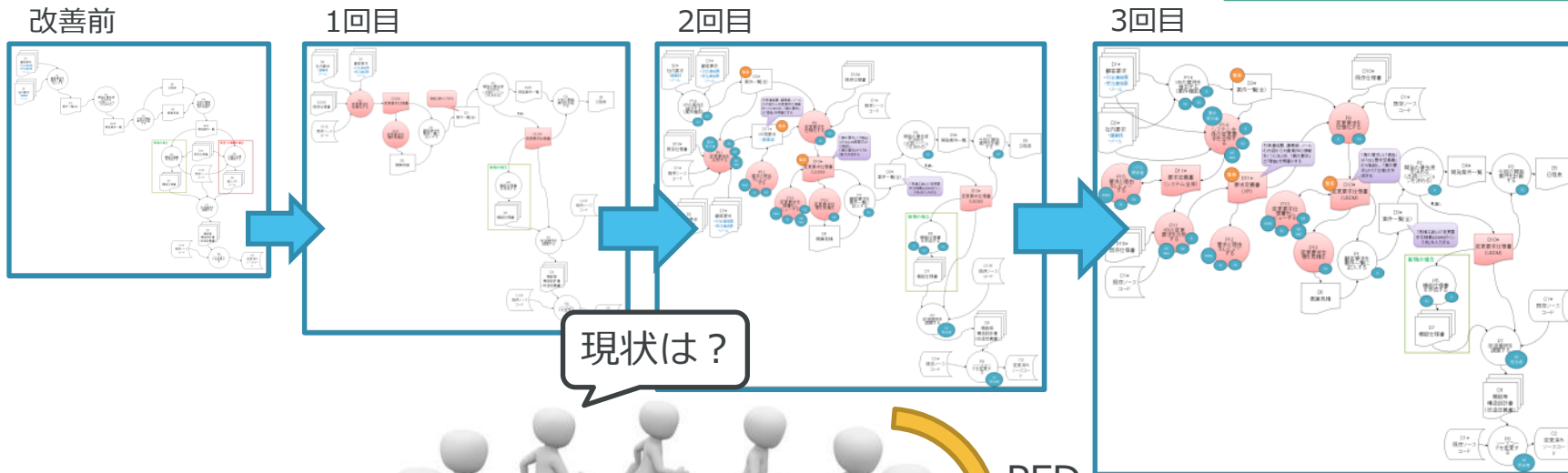
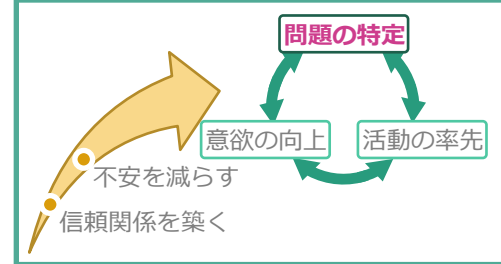
新しい方法なんか、やりたくないよ…

➤ 小さな実感の積み重ねで意欲を向上

「もやもや+やりたいこと」を聞き出す

問題の特定

ボトルネックの解消は段階的に



現状は？

PFD

やってみよう！

ここが辛い～

シミュレーション

これならできる？

現実を映し出す → 不足が見つかる

活動の率先

プロセス改善を駆動する

■面倒な管理作業 を引き受ける。

メトリクス の測定、
議事録 の作成、
ステークホルダー への報告、 などなど…

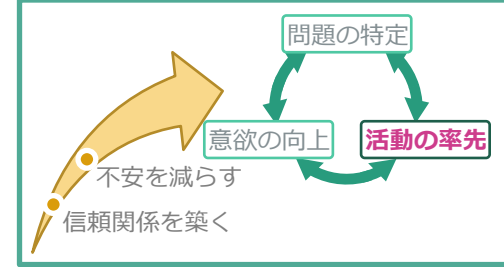
開発チームは
開発業務に集中！

開発チームを
守る壁！



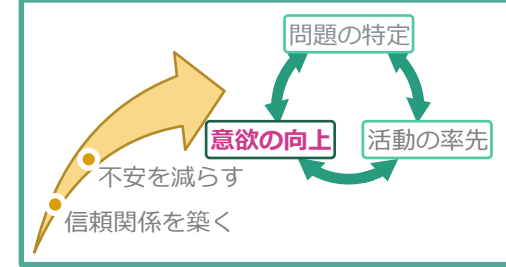
CC BY 2.0 (c) Ronnie Macdonald 2011.

プログラミングは**萌え** マネジメントは**萎え**



意欲の向上

小さな成功体験の積み重ね



- やりたいと思ったところから逐次
- できたかどうかすぐに確認

不安...



安心から不安に戻さない



安心!

開発チームの実感を重視

数字より気持ち

今度は成功

プロセス改善の結果

- 要求管理の方法を変更
 - ✓ USDMを改造した要求定義書で一元管理
 - ✓ PFD上でステークホルダーを特定・周知



要求の抜け漏れ・不明点の洗い出しが早い段階で可能になった



数字もついてきた

実機テストの重大不具合が減少



	変更コード行数 の比率	テスト&デバッグ 工程の工数の比率	潜在欠陥率 の比率
改善前	100	100	100
アシスト時	82	14	15
アシスト終了後	213	52	20



すごい!

潜在欠陥率80%
改善しました



社長

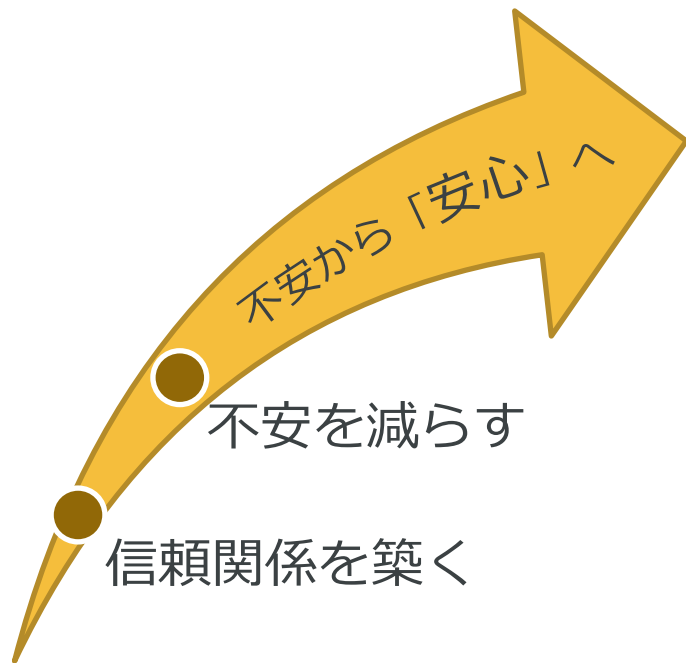
上長がアシストし
開発チームが改善活動を継続

まとめ

まとめ

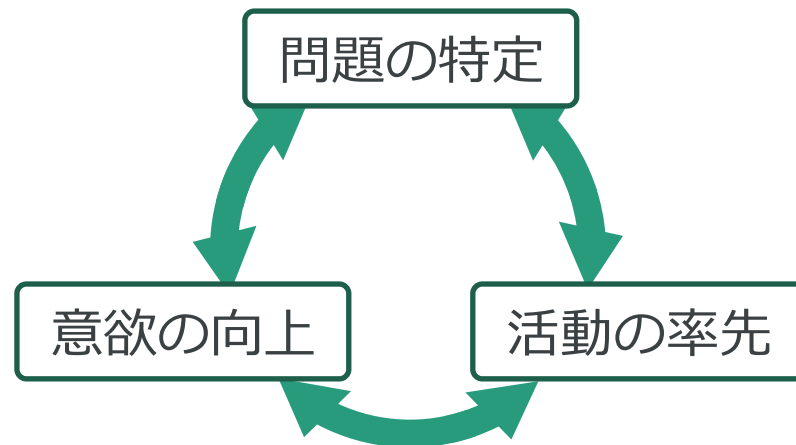
Step1.

開発チームの話を聞く



Step2.

開発チームのアイデアを形にする



アシストは「黒子」に徹することが重要

Anritsu
envision : ensure