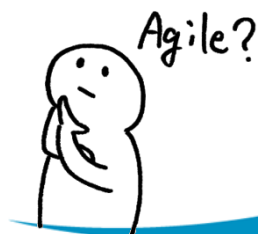


プロダクト、プロジェクトともに 価値をあげられるからアジャイル

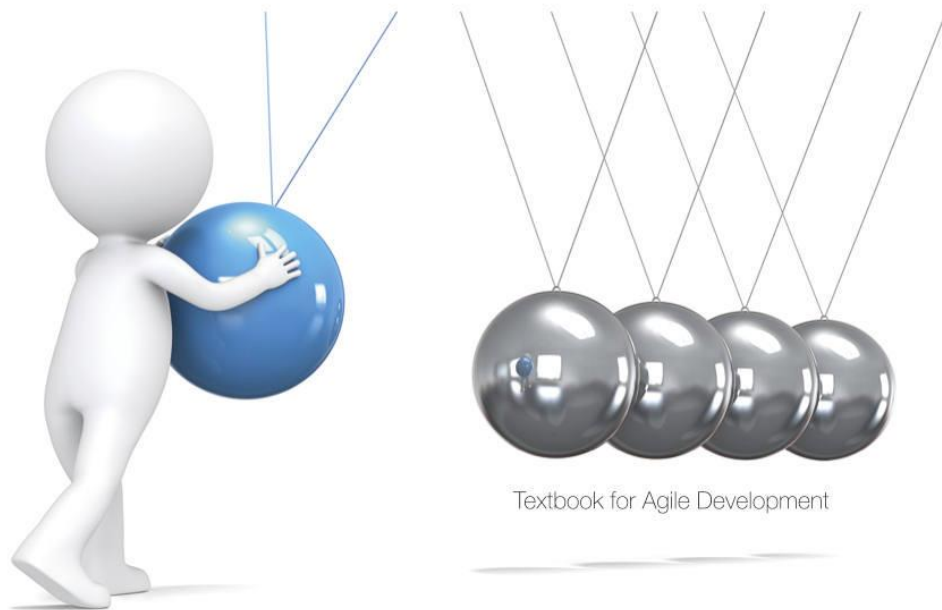
～ 価値をフィードバックし続けるアジャイルの本質 ～



October 9, 2015
株式会社日新システムズ
未来戦略室 室長
前川 直也

1. いまどきのプロマネ？
2. ソフトビジネスの変化
3. プロダクトの価値を創る
4. プロジェクトの価値を創る
5. アジャイルになる！





Textbook for Agile Development

わかりやすい **アジャイル**
開発の教科書

前川直也+西河誠+細谷泰夫 著

今日のベースは
これです！

<http://www.sbcr.jp/products/4797371284.html>

CM



— 壺 —

いまどきのプロマネ？

トップダウン or ボトムアップ？



トップダウン型マネジメント

- ☺ 経験・ノウハウが受け継がれる
- ☺ メンバがやるべきことがわかりやすい
- ☹ 経験がかえって邪魔になることも
- ☹ メンバが考えなくなってしまうリスク



ボトムアップ型マネジメント

- ☺ メンバ自身が考えて・工夫する
- ☺ 知識の共有が実現しやすい
- ☹ クローズ化してしまうと逆効果
- ☹ ビジョンの共有が不可欠

どちらも重要ですが、違いを使いこなしていますか？

ダグラス・マクレガーのX理論 Y理論

人間観・動機づけにかかわる2つの対立的な理論



【X理論】

人間は生来怠け者で、
できれば働きたくない
強制されたり命令されなければ
仕事をしない

or



【Y理論】

生まれながらに嫌いということはなく、
働くことは人間の本性
条件次第で責任を受け入れ、
自ら進んで責任を取ろうとする
問題解決のための創意工夫をこらす
能力は誰でも持っている

サーバント型リーダーシップ (支援型リーダーシップ)

メンバーのために「管理・命令」し、チームやプロジェクトを運営するスタンスのリーダーではなく、メンバーを「信頼・支援」し、チームやプロジェクトの「成功・成長」のために奉仕するリーダー。コミュニケーションを重視しながら、一人ひとりの思いと行動により目標を達成する。

ロバート・グリーンリーフ（米：1904～1990）が1970年に提唱



プロジェクトは何のためにあるの？



- What 何を作りたいのか？
- Who 誰が作るのか？誰と作るのか？
- When いつまでに作るのか？
- Where どのような環境で？
- How どうやって？
どんな技術？どんなプロセス？

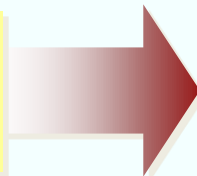


◆ Why

なぜ作らなければ
ならないのか？



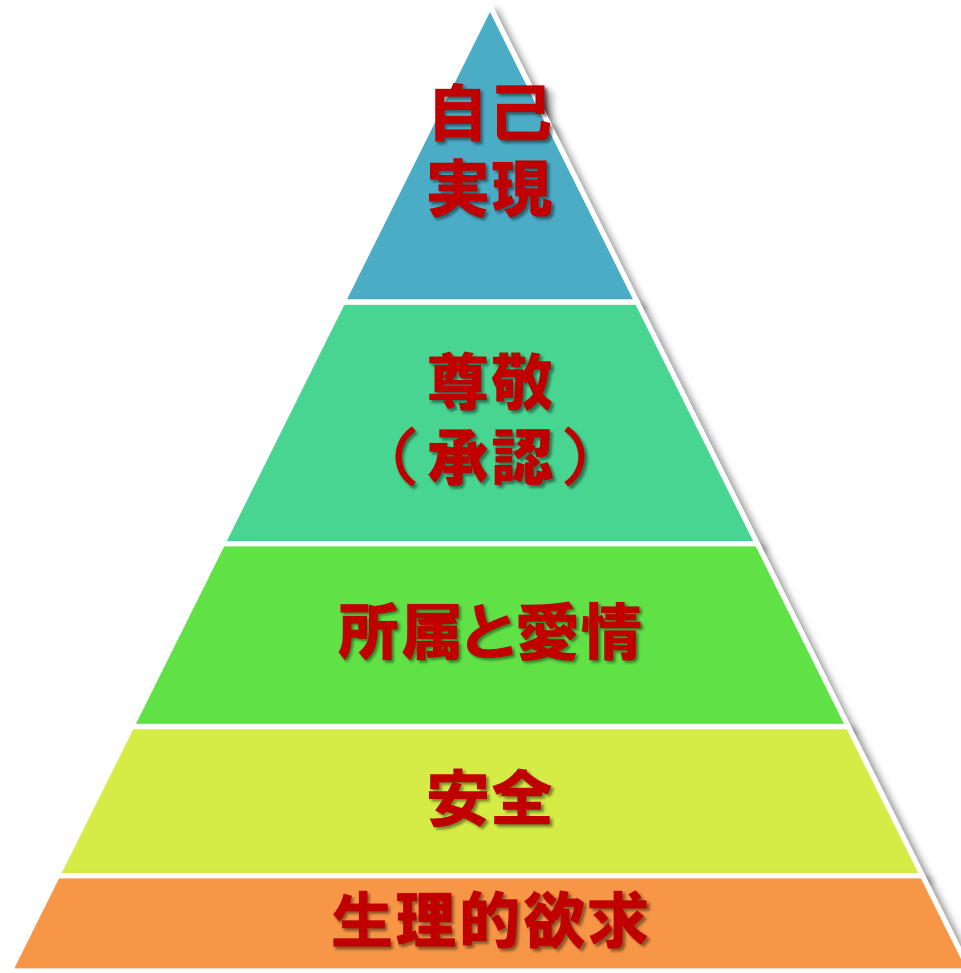
プロジェクトは**何のために存在し、**
なぜ自分たちがそこにいるのか



プロジェクトが**成功すれば**
未来にどんな価値が描ける？



マズローによる欲求階層



ほんとうの「Why」!?

企業・組織としての Why

- 利益をあげねば
- 競合他社に勝つ
- 生産性をあげる
- 高品質な商品



◆ Why
なぜ作らなければ
ならないのか？



個人としての Why

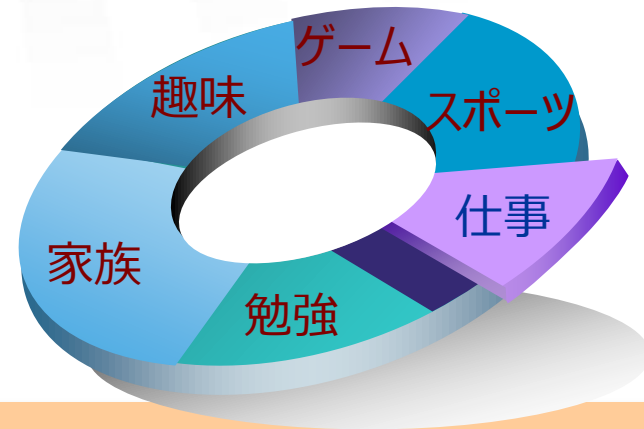
- リーダとしての成功
- 新しい技術を得る
- 自分の時間を作る
- 残業しても高収入

パワーを引き出す「素」は
結局のところ「欲求」につながる

いやなことはしたくない……



やりたい！につながる「Why」になってますか？



二つのプラスが マッチングしたとき 最高スペックに！

ビジネスで成功

業績アップ

企業・組織としての
Why

あせり・指示

Command Control



Why?



自己成長
チーム成長

自分の時間

個人としての
Why

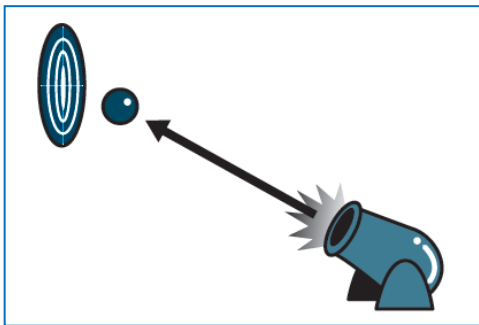
やらされ感

思考停止
成長停止



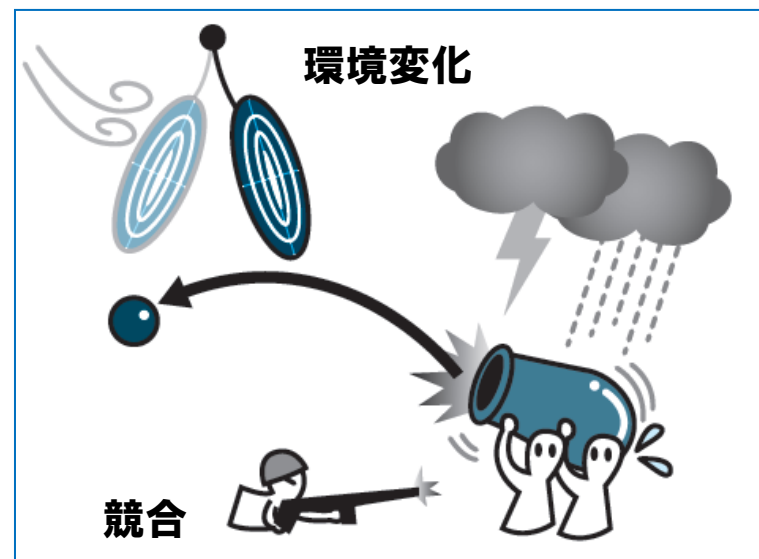
— 式 —
ソフトビジネスの変化

ソフト業界の変化



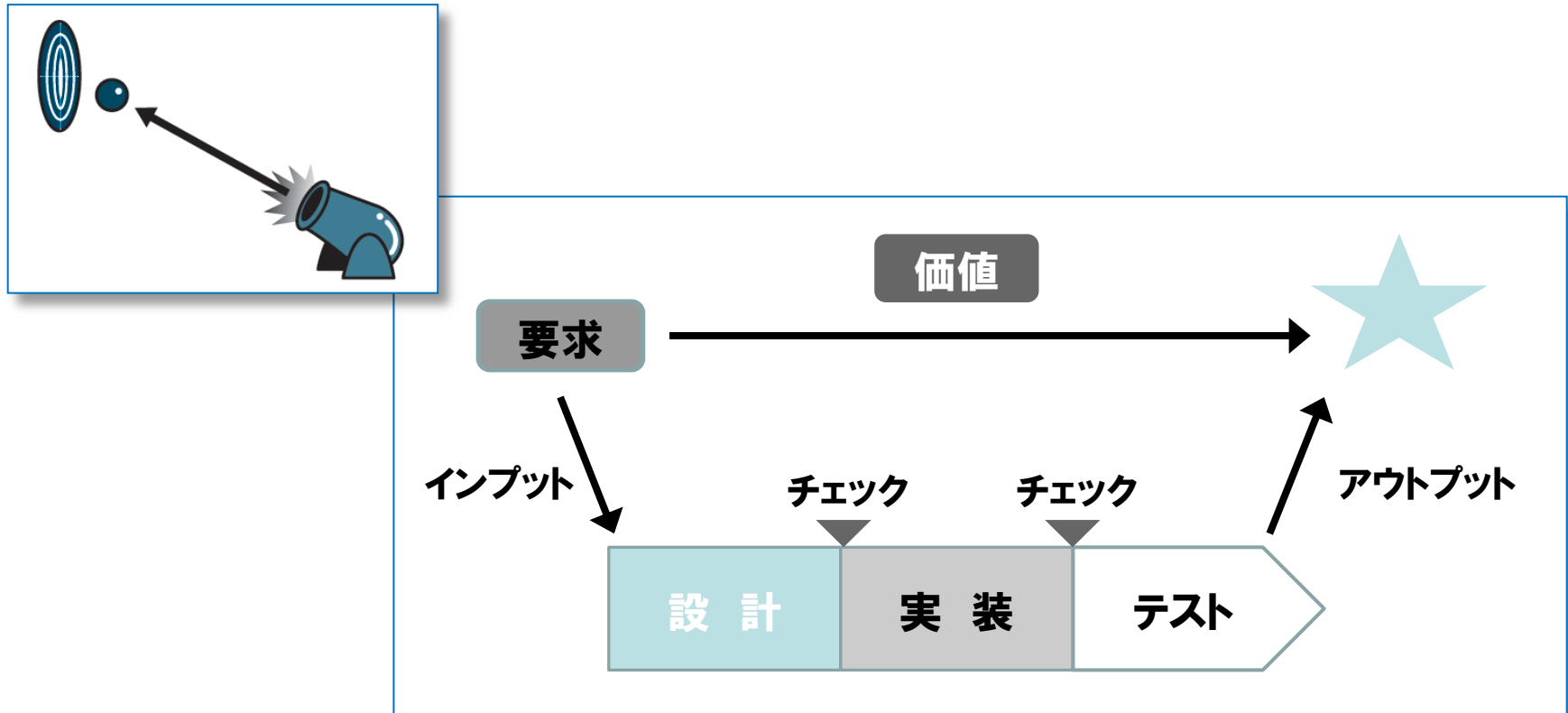
以前は、狙っていけば的中する確率が高かった

今のソフト業界では、
環境の変化
ユーザーニーズの多様化
競合他社との競争激化
などで、先の読めない状況・・・



『要求される価値』から『創り出す価値』の時代に突入！

これまでのソフトウェア開発

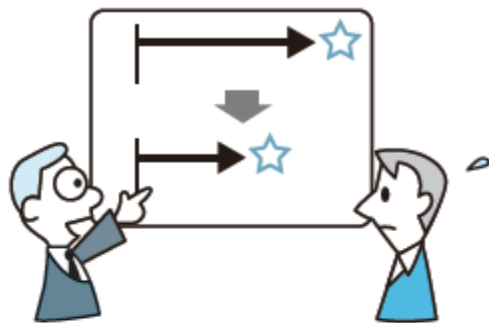


以前は、決められた機能を要件に落とし込み
計画をほぼ変えることなくものづくりができた時代

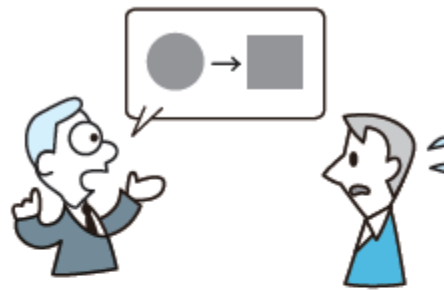


ソフトウェア開発に変化はつきもの

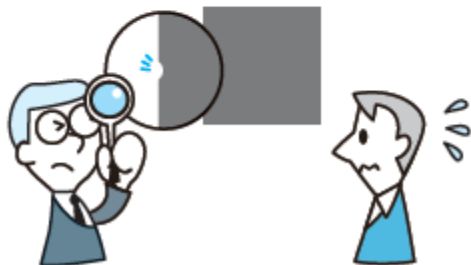
日程前倒し



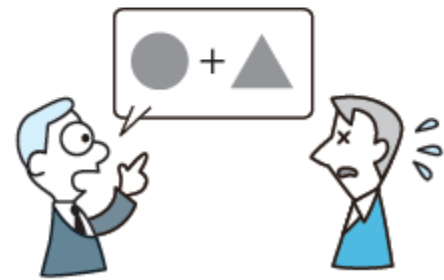
仕様変更



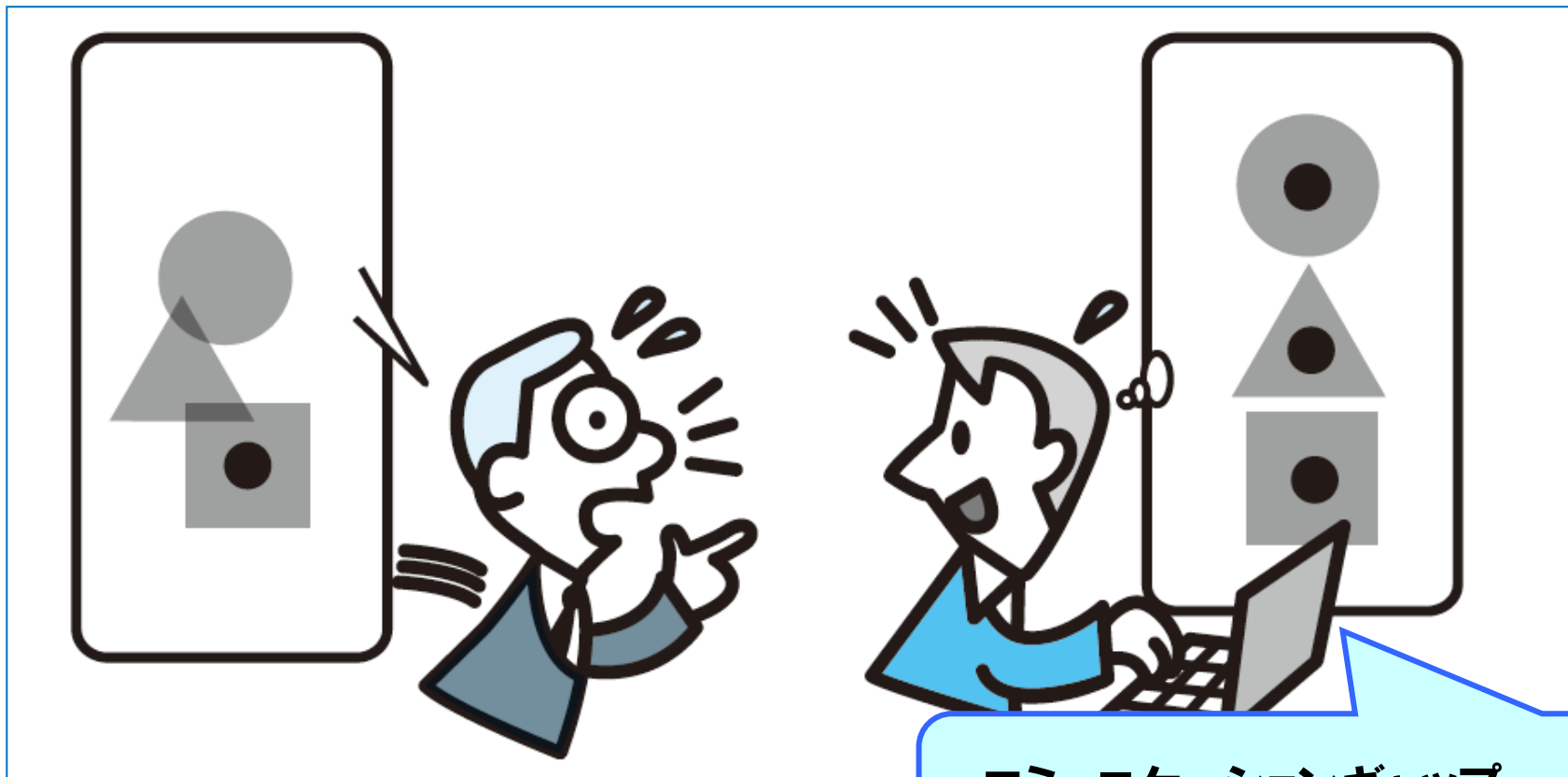
課題／バグ



仕様追加

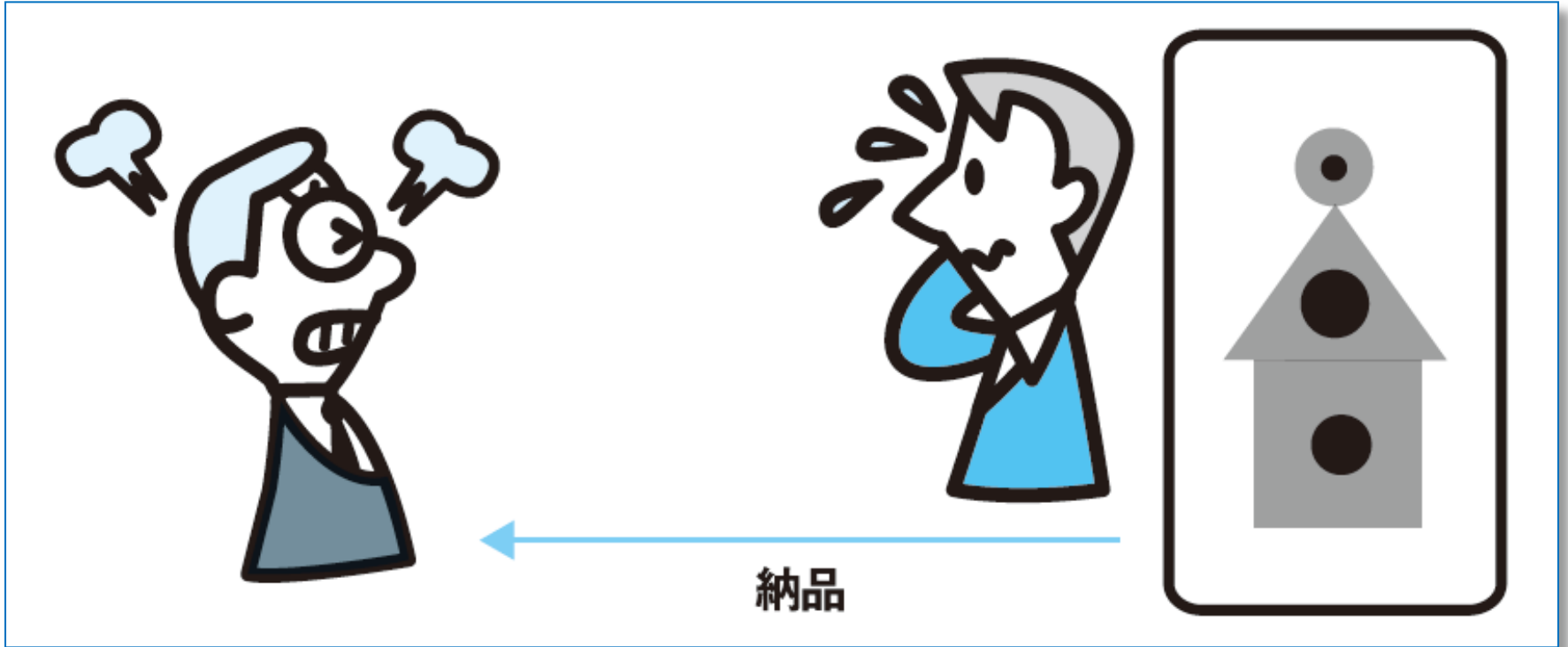


混沌としたソフト業界において、
ソフトウェアの変化が発生しないというのはいりえない
変化を前向きに受け入れていく必要があるのだが...

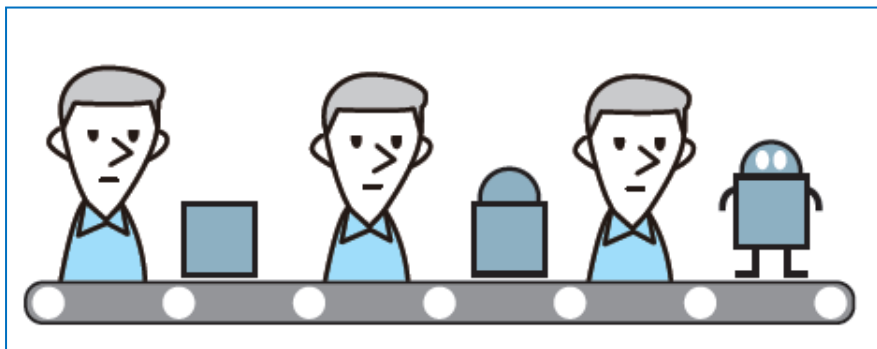


コミュニケーションギャップ
設計・実装上の都合・納期 等

納品時の結果

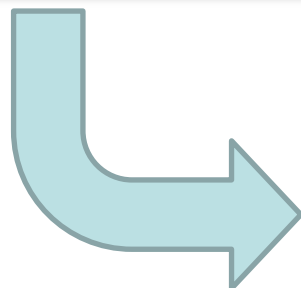


ソフト業界 約60年の歴史



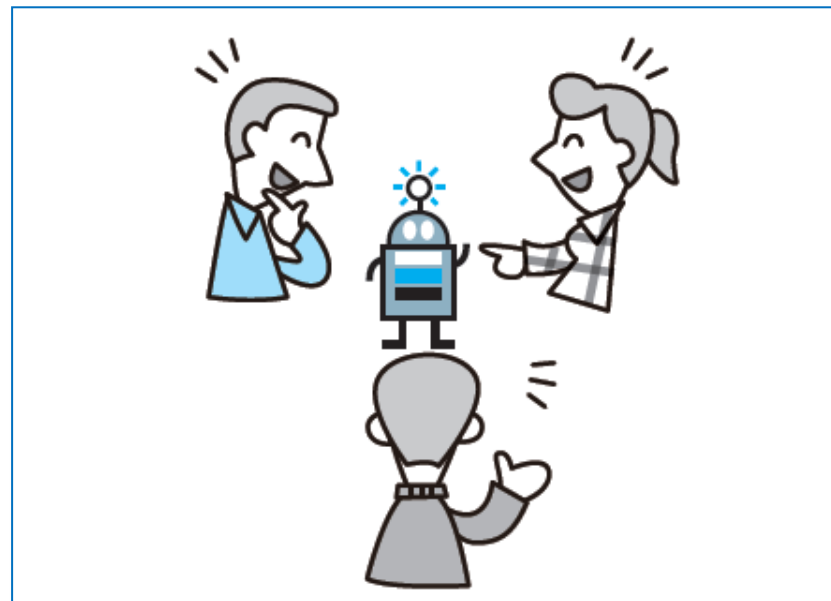
ソフトウェアが商業ベースになり
それにつれて、工学的にアプローチ

『誰でも同じように作れるソフトウェア』



2000年頃から、もう一度初心に戻り
新たなアプローチが始まる

『ソフトウェアは人が作るものである』



What

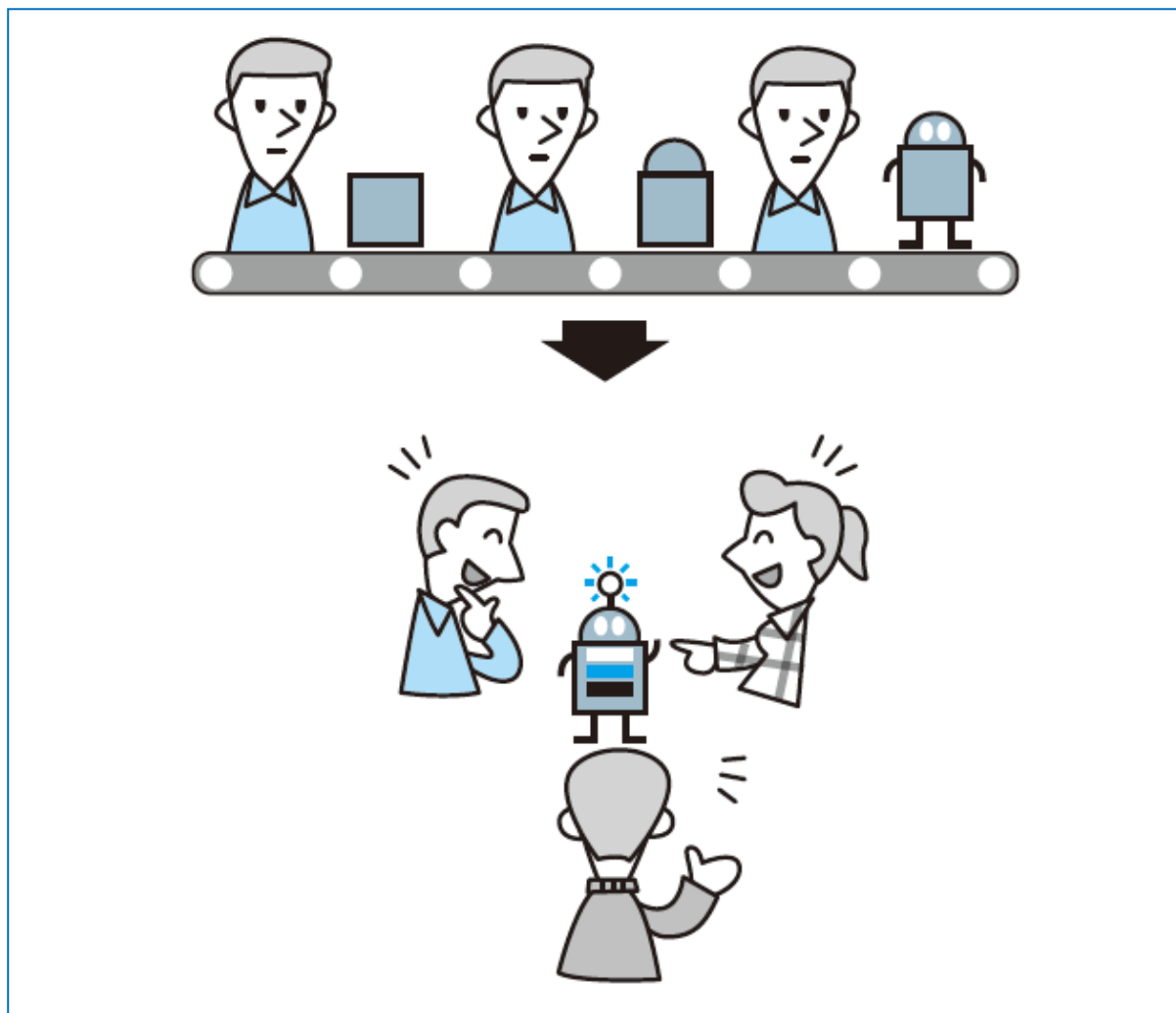
How



Who

Where / When

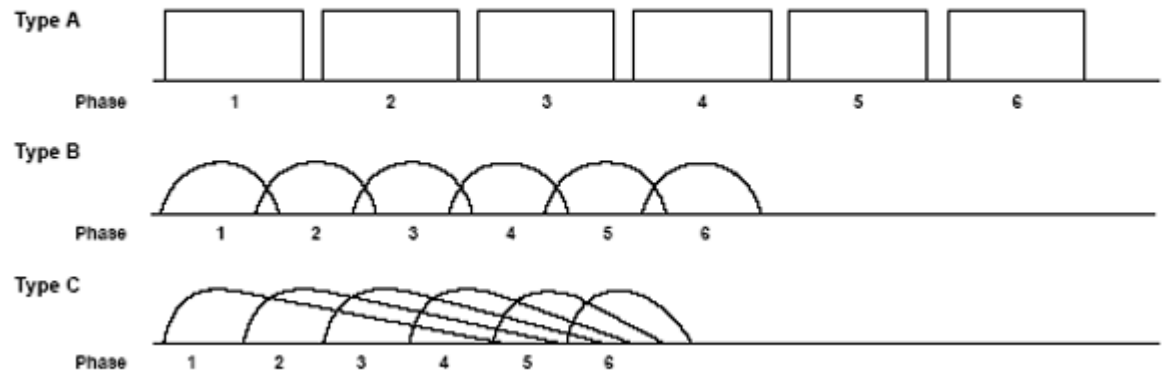
どちらがエンジニアを活かせますか？



スクラムが生まれたきっかけ

EXHIBIT 1

Sequential (A) vs. overlapping (B and C) phases of development



論文“The New New Product Development Game”
(竹内弘高・野中郁次郎『ハーバード・ビジネス・レビュー』1986年)

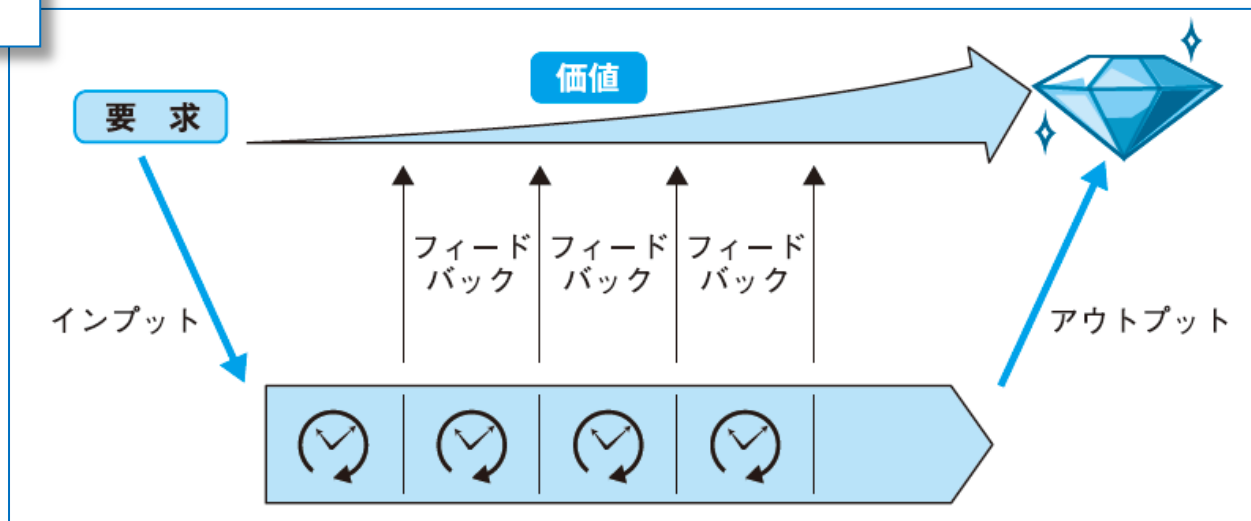
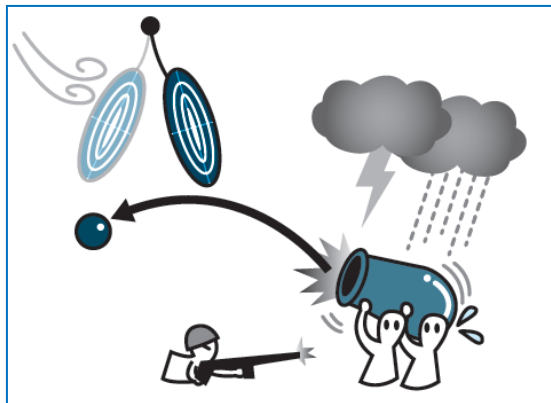
アジリティの考えは、従来の米国製造業が限界を認識し、それを越えようとしていた時期に生まれてきました。つまり、製造業はもはやハードのみの生産者でなく、ソフト、サービスを融合した価値を提供する、あらたな存在を目指すべきである、と。そうでない限りグローバルな競争環境において生存不能だ、という認識です。こうした認識にもとづいて、ハード中心だったコンピュータ業界がソリューションなどのソフト、サービス指向へと変身し、消費者向け製品でもマス・カスタム化（顧客の注文による生産。ワン・トゥ・ワン・マーケティングもこの一種）が重要な考え方となりました。

『知識経営のすすめ -ナレッジマネジメントとその時代』野中 郁次郎／紺野 登1999年12月20日 第1版

富士ゼロックス
キャノン
ホンダ
NEC
の事例から分析

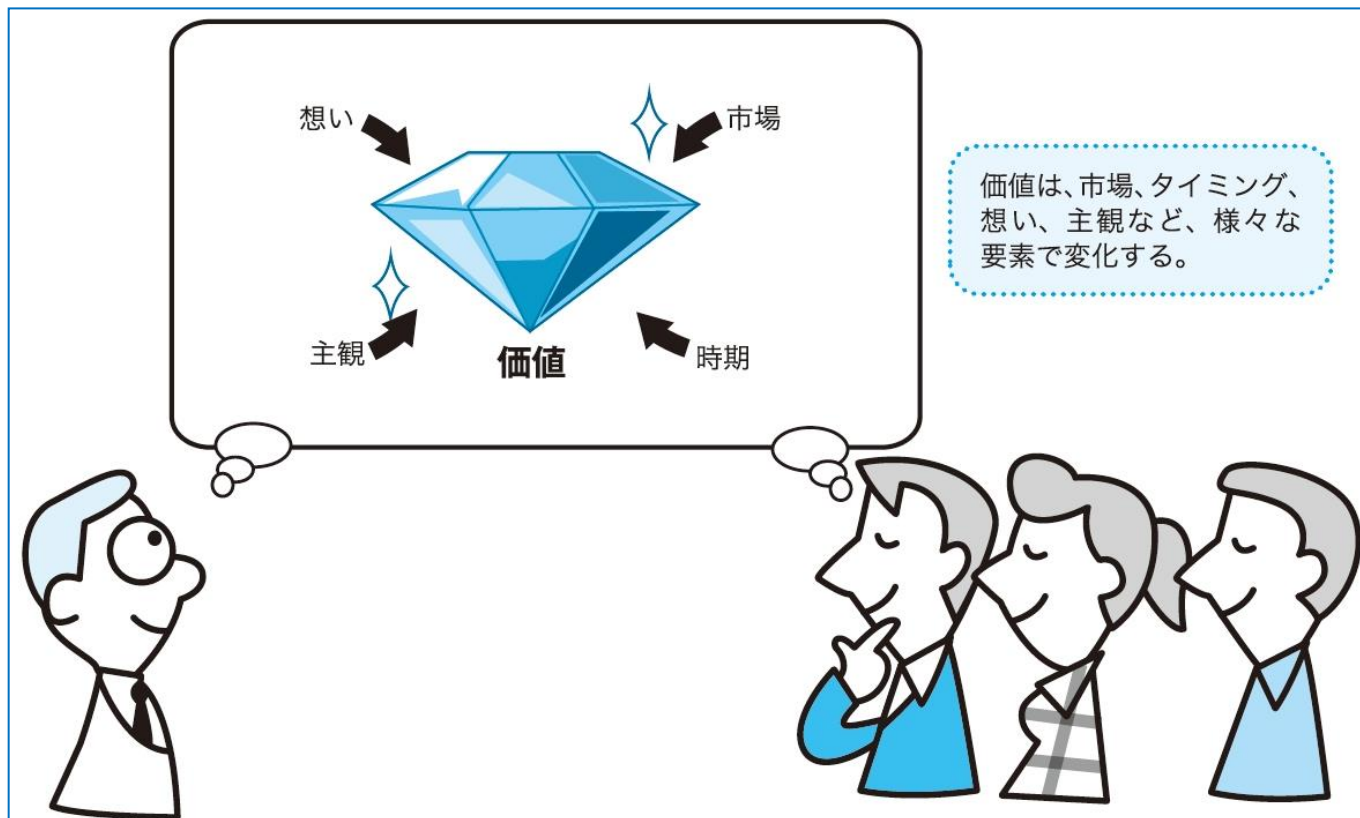


今求められているソフトウェア開発



短い『タイムボックス』で回しながら、
細かくフィードバックし、価値を膨らませていく開発スタイル

システム・製品の価値の最大化を考える



システムや製品の価値を決めている主要な部門はどこですか？



より良い
ものを提供

**変化を味方につけ
お客様のビジネス価値を最大化する**



— 式 —

プロダクトの価値を創る

アジャイルソフトウェア開発宣言

私たちは、ソフトウェア開発の実践
あるいは実践を手助けをする活動を通じて、
よりよい開発方法を見つけたそうとしている。
この活動を通して、私たちは以下の価値に至った。

プロセスやツールよりも**個人と対話**を、
包括的なドキュメントよりも**動くソフトウェア**を、
契約交渉よりも**顧客との協調**を、
計画に従うことよりも**変化への対応**を、

価値とする。すなわち、左記のことがらに価値があることを
認めながらも、私たちは右記のことがらにより価値をおく。

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

<http://agilemanifesto.org/>

アジャイル開発はあたりまえになりつつあります

ただし、なぜアジャイルなのか？（目的）

どんな価値を出すのか？（目標）をより明確にしなければ・・・

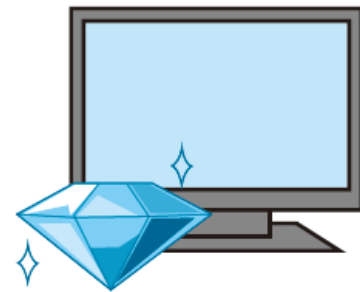
アジャイルソフトウェア宣言の背後にある原則

原則	補足説明
顧客満足を最優先する	価値のあるソフトウェアを継続的に提供する
要求の変更を歓迎する	開発の終盤でも変更を受け付ける。変化を味方につけることでお客様の競争力を引き上げる
動くソフトウェアを継続的にリリースする	動くソフトウェアを2-3週間から2-3ヶ月というできるだけ短い時間間隔でリリースする
ビジネス側の人と開発者が一緒に働く	ビジネス側の人と開発者はプロジェクトを通じて日々、一緒に働く
意欲のある人でプロジェクトを構成する	意欲のある人に環境と支援を与え、仕事が無事終わるまで彼らを信頼する
フェイス・トゥ・フェイスで対話をする	情報を伝える最も効率的な方法は、直接話をする
進捗の尺度は動くソフトウェア	動くソフトウェアこそが進捗の最も重要な尺度になる

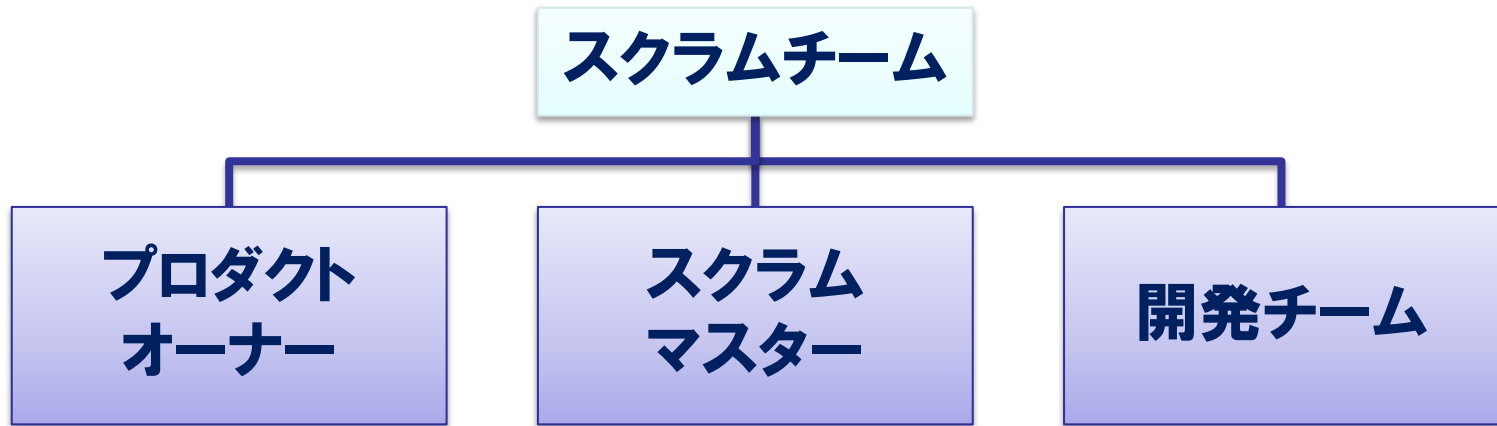
アジャイルとは？



顧客のシステム・製品の
ビジネス価値を最大化



アジャイルとは、
お客様のビジネス価値を最大化するための
「考え方」 や **「姿勢」** のこと



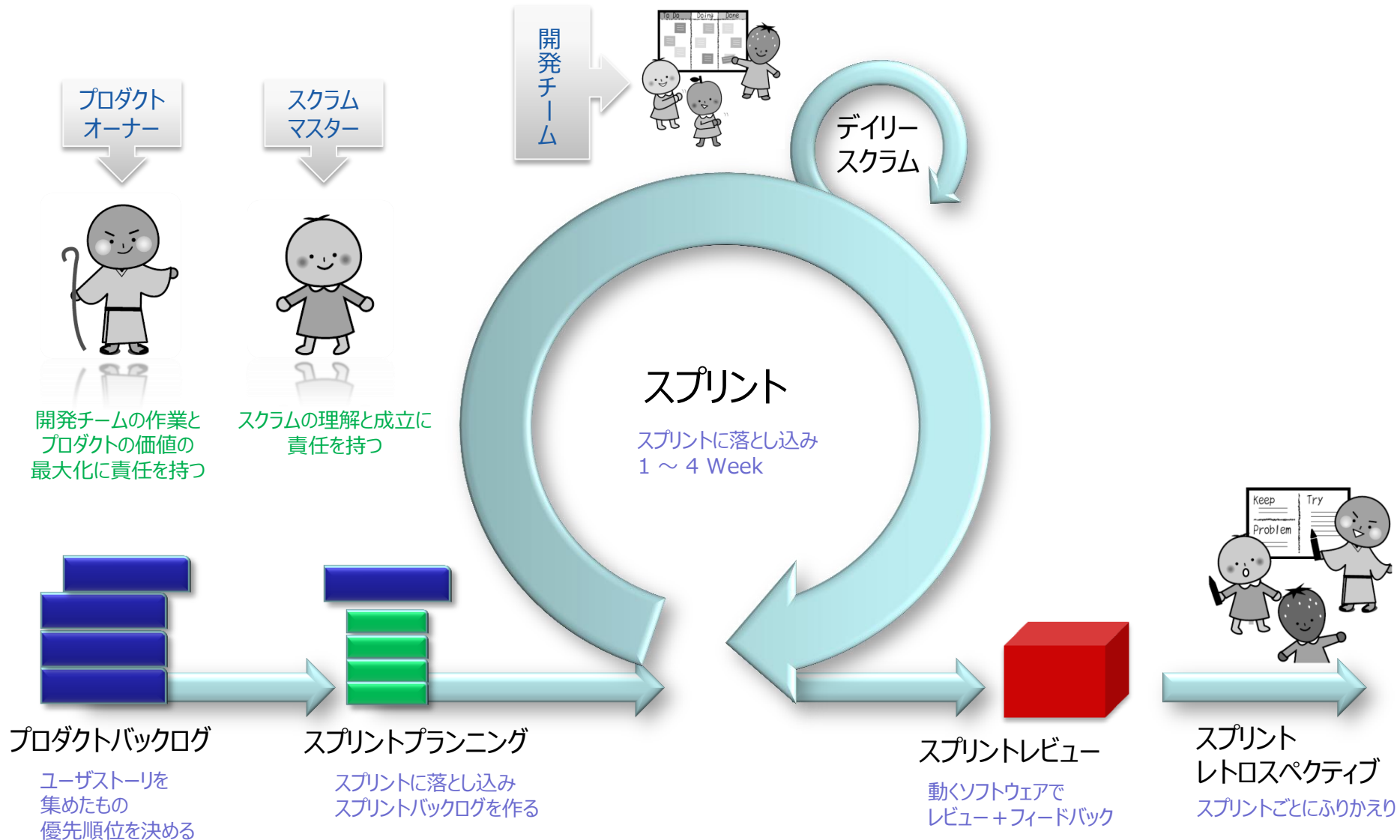
自己組織化チームは、作業を成し遂げるための最善の策を、
チーム外からの指示ではなく、自らが選択する

「スクラムガイド」より

© 1991-2013 Ken Schwaber and Jeff Sutherland, All Rights Reserved

<https://www.scrum.org/Portals/0/Documents/Scrum%20Guides/2013/Scrum-Guide-JA.pdf>

スクラムの流れ



アジャイルの3つの要素とスクラムでのロール

Agile



ゴール

何のために何をつくるか
ゴールを描き共有する

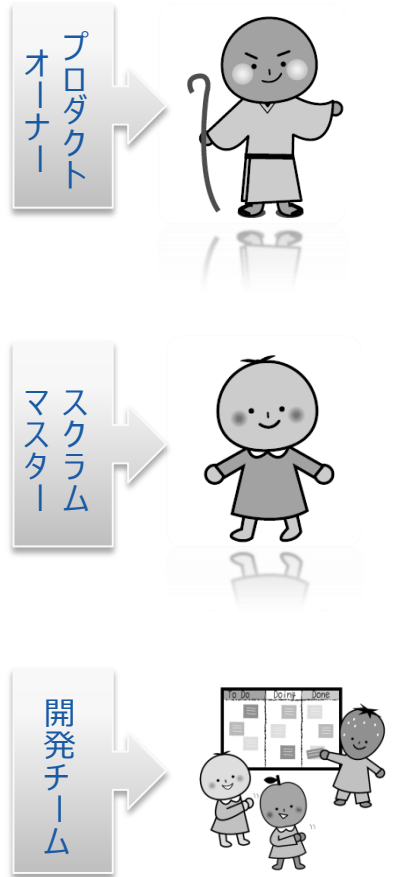
リズム

リズムを共有させて
チームの動きを一つにし
相互作用を引き出す

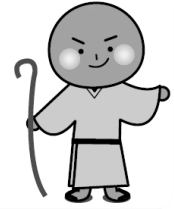
愛

自分たちが作るもの
自分たち自身に愛着があり
そのために力を注げる

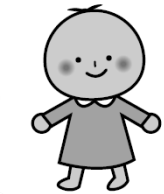
一つのチームとして連携



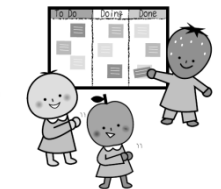
プログラクト
オーナー



スクラム
マスター



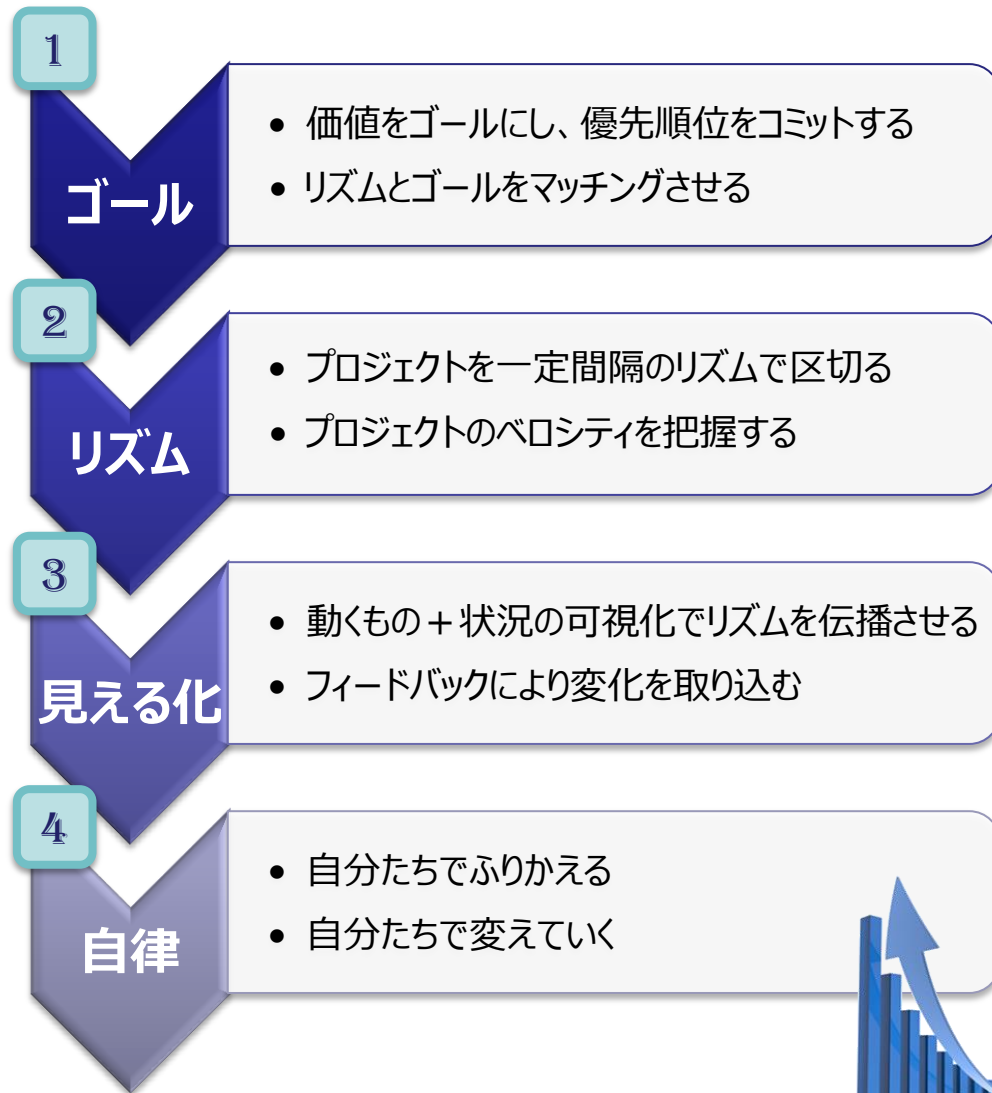
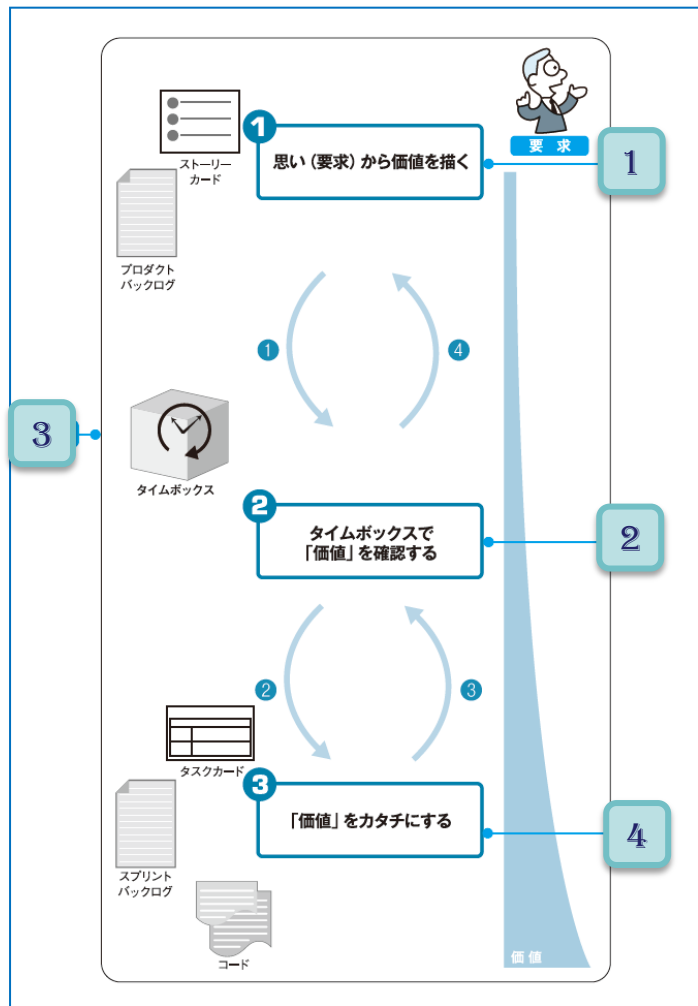
開発
チーム



[チェックポイント]

- ✓ プロジェクトにエンジニアリング & プロセスの基本スキルはどのぐらい？
- ✓ メンバが風土を変えるぐらいの改善意欲を持っていますか？
- ✓ 自分たちが作っているものに愛着を持っていますか？
 - お客様が何を求めているのか考えてる？
- ✓ 改善指標を数値だけで判断していませんか？
 - コミュニケーションは活発？
- ✓ メンバとゴールを共有できていますか？

わかりやすい アジャイルの『ツボ』



① 価値をゴールに設定し、コミットする

ゴール

- 価値をゴールにし、優先順位をコミットする
- リズムとゴールをマッチングさせる

リズム

- プロジェクトを一定間隔のリズムで区切る
- プロジェクトのベロシティを把握する

見える化

- 動くもの + 状況の可視化でリズムを伝播させる
- フィードバックにより変化を取り込む

自律

- 自分たちでふりかえる
- 自分たちで変えていく



プロダクト
オーナー



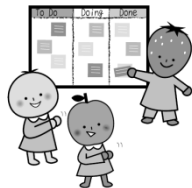
開発チームの作業と
プロダクトの価値の
最大化に責任を持つ

スクラム
マスター



スクラムの理解と成立に
責任を持つ

開発
チーム



デイリー
スクラム

スプリント

スプリントに落とし込み
1 ~ 4 Week

プロダクトバックログ

ユーザストーリーを
集めたもの
優先順位を決める

スプリントプランニング

スプリントに落とし込み
スプリントバックログを作る

スプリントレビュー

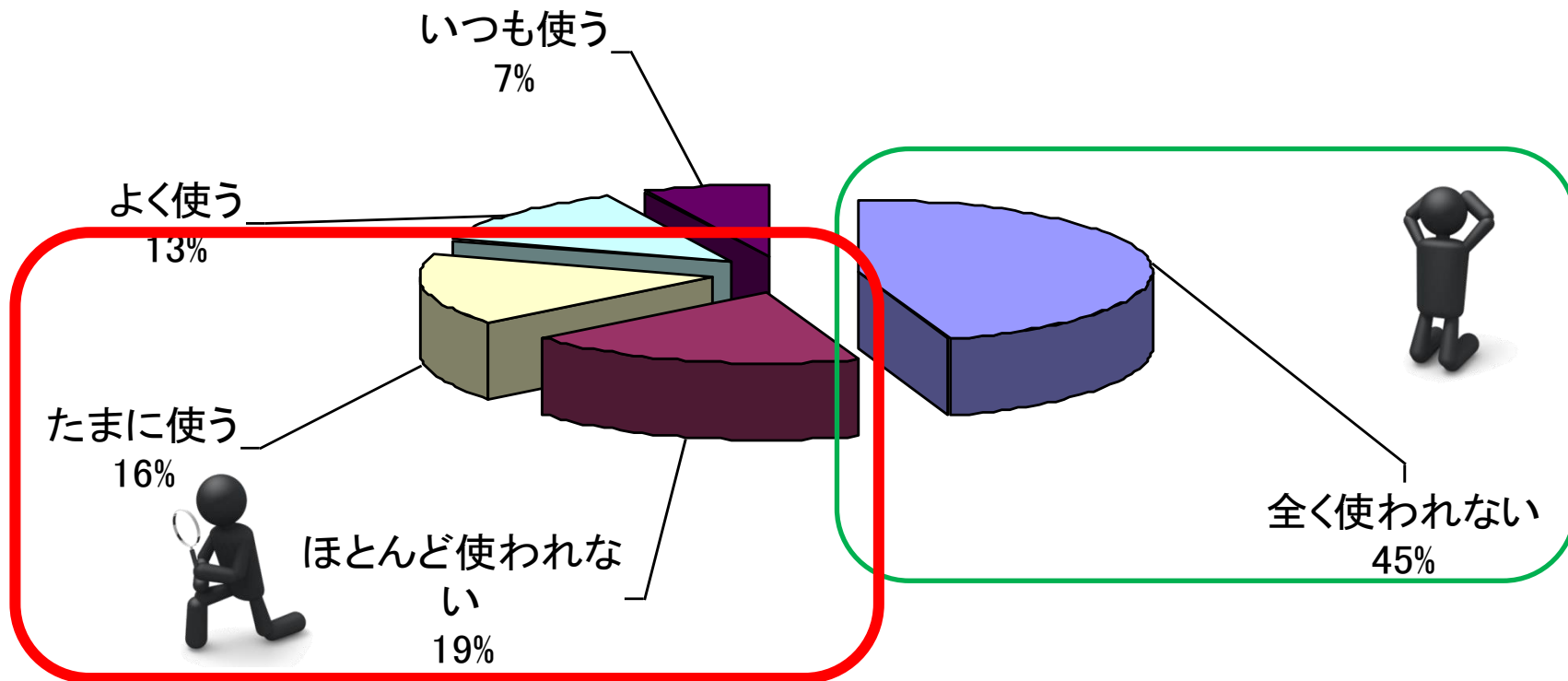
動くソフトウェアで
レビュー+フィードバック

スプリント
レトロスペクティブ

スプリントごとにふりかえり

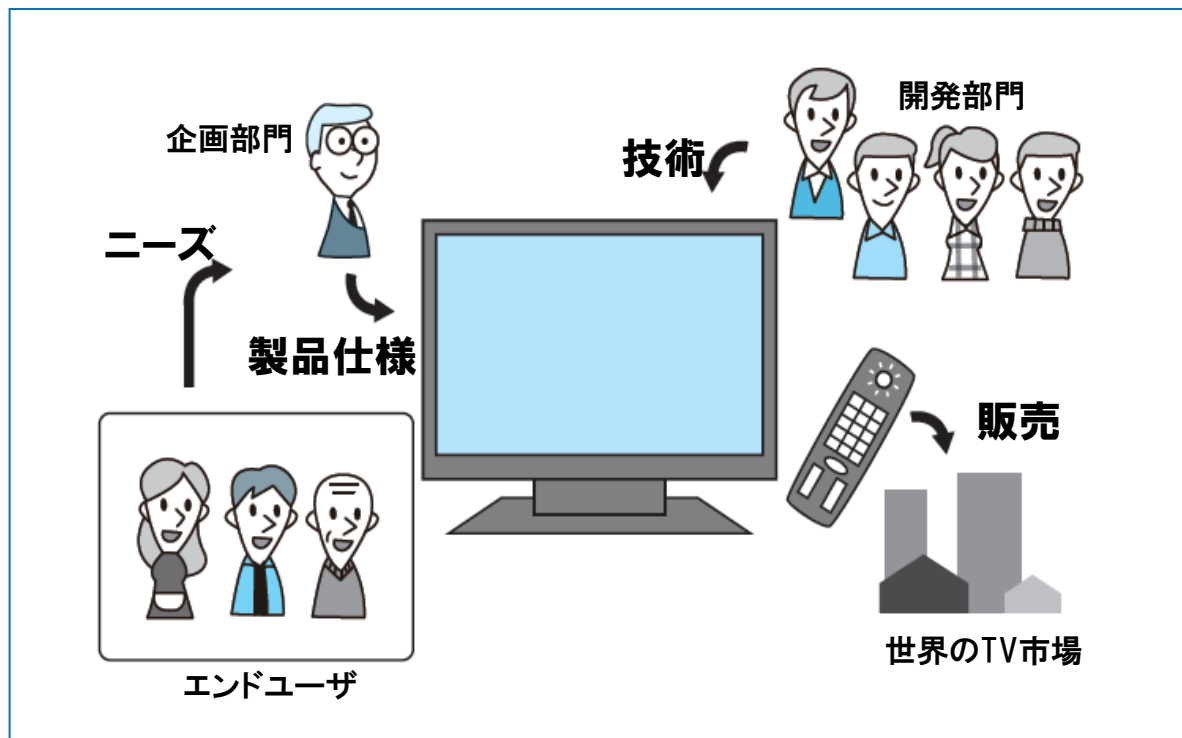
「たまに使う」に意味があったりしませんか？

システムの機能の利用度



Standish group study report in 2000 chaos report

製品の価値の最大化を考えるには？



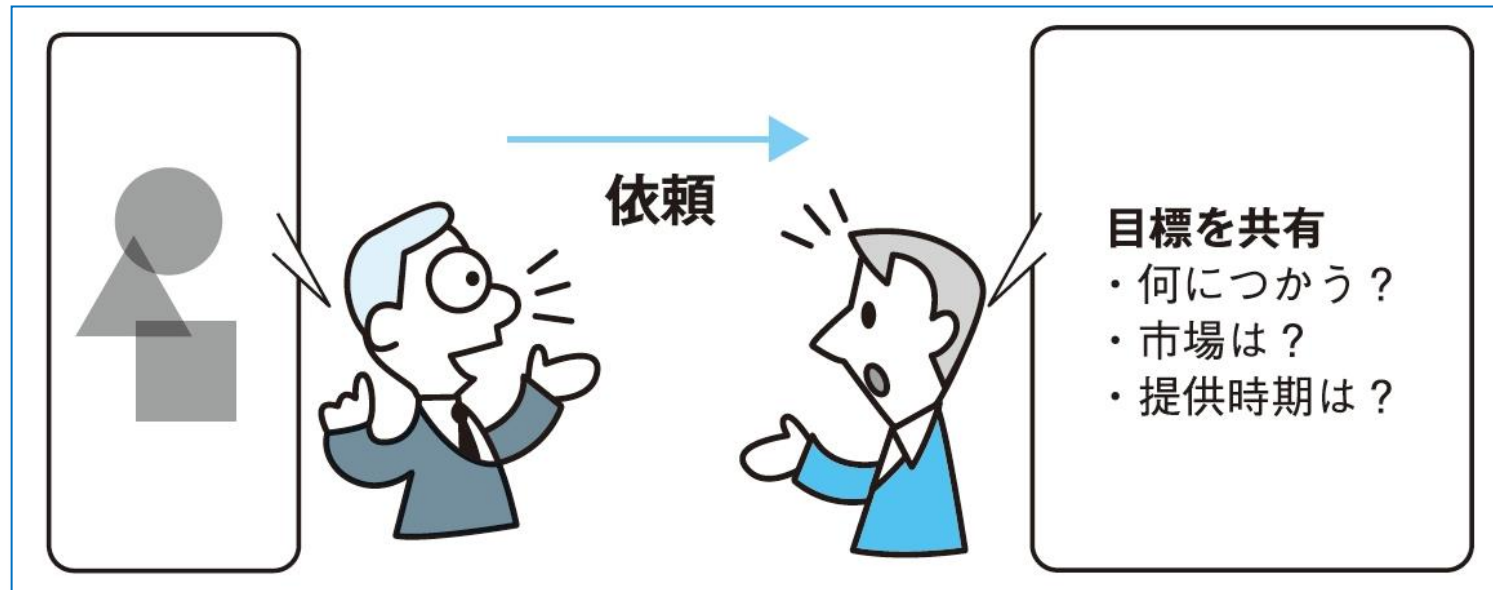
製品の価値を最大化するために
作り手側に何が求められているのか？

これだどう思いますか？



1. 世界初 4Kミラーレス一眼
2. 他社に負けないAFスピード
3. 動画プロフェッショナル対応

※実際のコンセプトとは違います

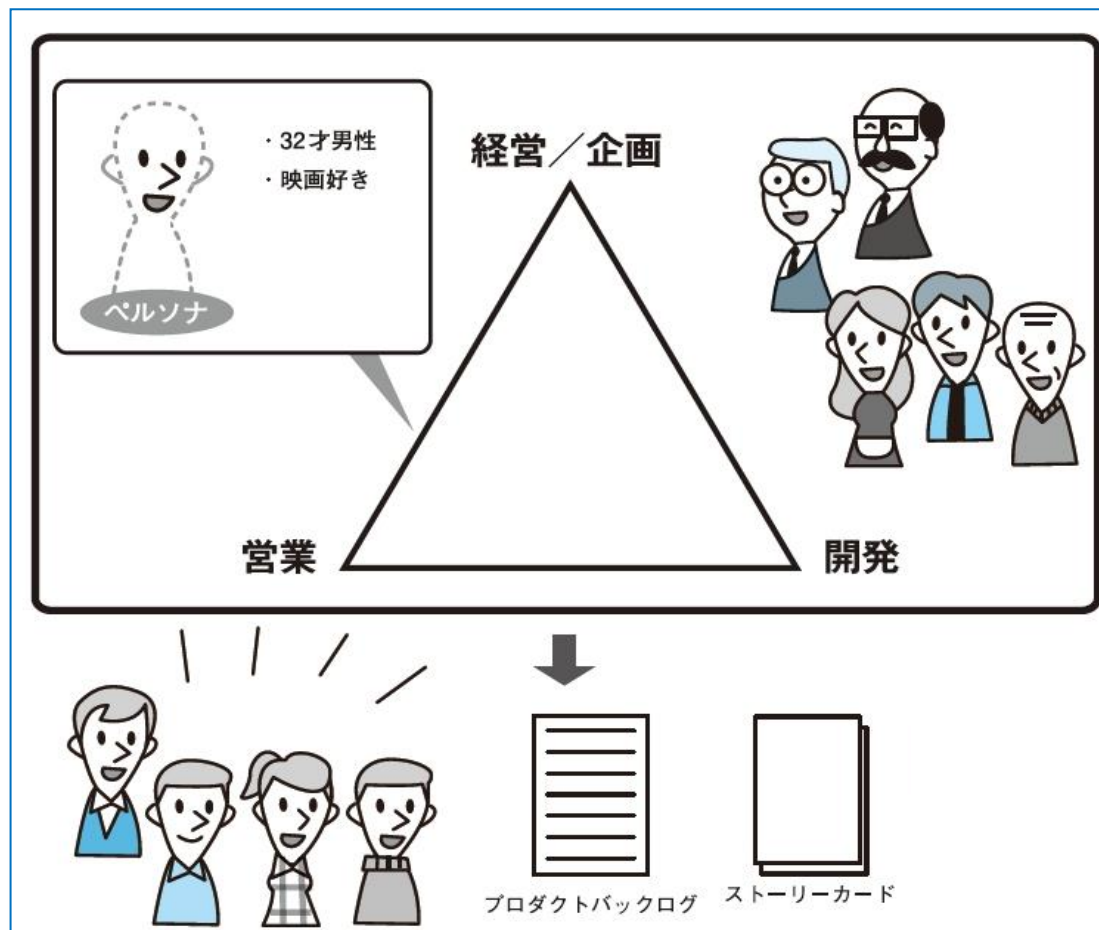


お客様の価値の最大化を考える
変化は当然（必要）にとらえ、
すばやく変化を取り入れられるように進める

変化を味方につける（開発側から）

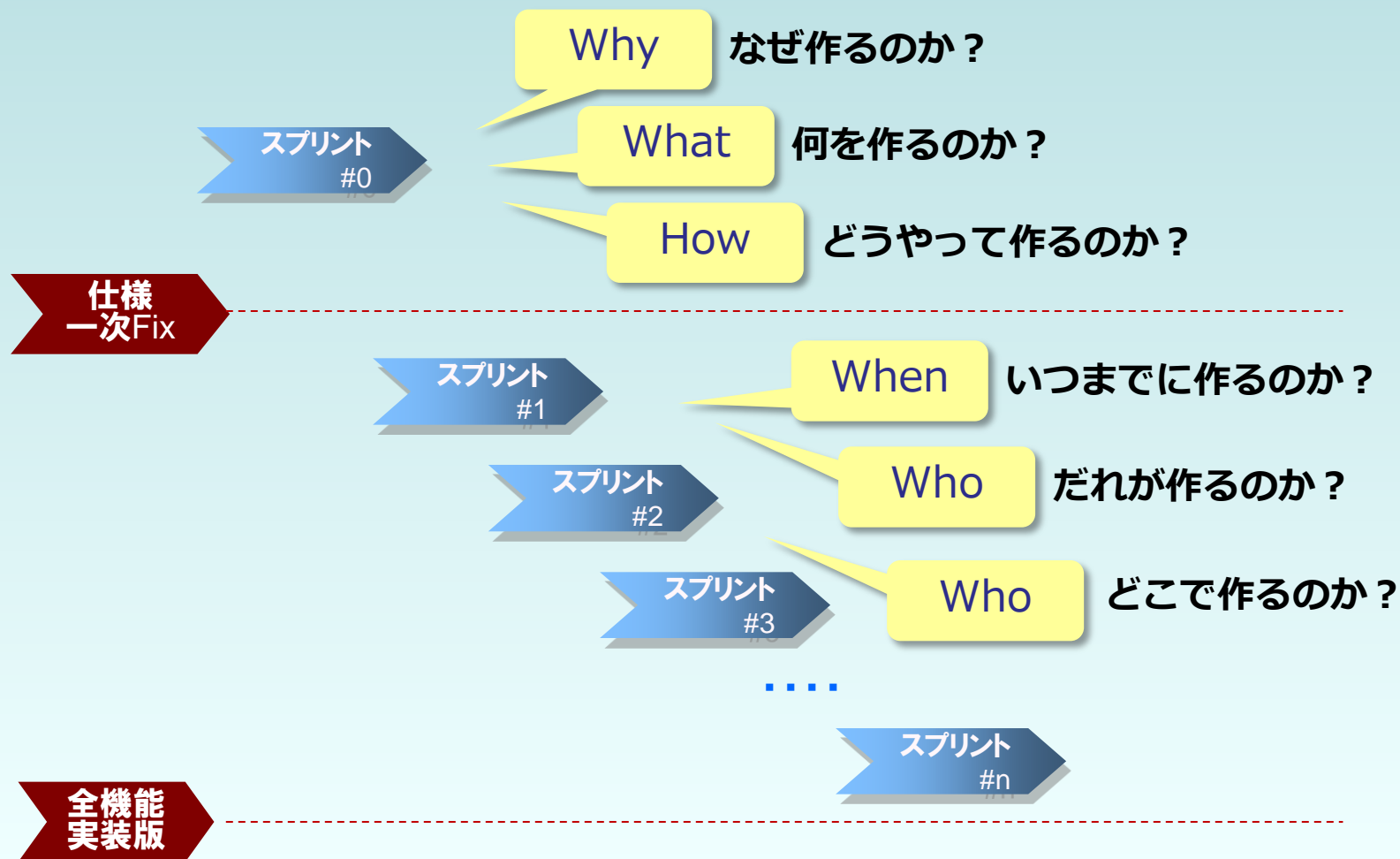


価値を最大化するには？



部門の壁を越えて、システムや製品の価値を共有するには一緒に描き、一緒に作り、一緒に確認していく必要がある

5W1Hで考えるのはアジャイルでも同じ



設計基本のV字モデルをキープしながら
いつまでに、何を、どう作るのか
ゴールを明確にして、一定期間のサイクルで回す



ゴールを分割して、プロセスの流れを作る

要求分析

システムテスト

基本設計

結合・機能テスト

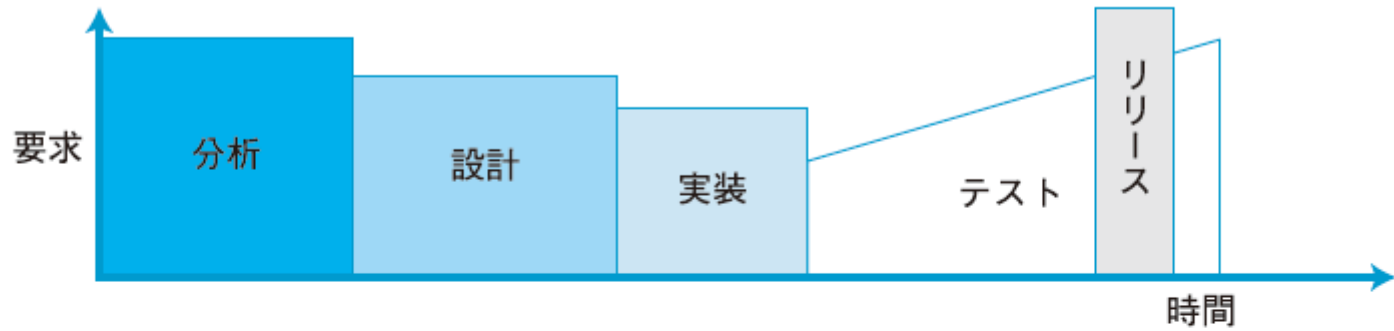
詳細設計

単体テスト

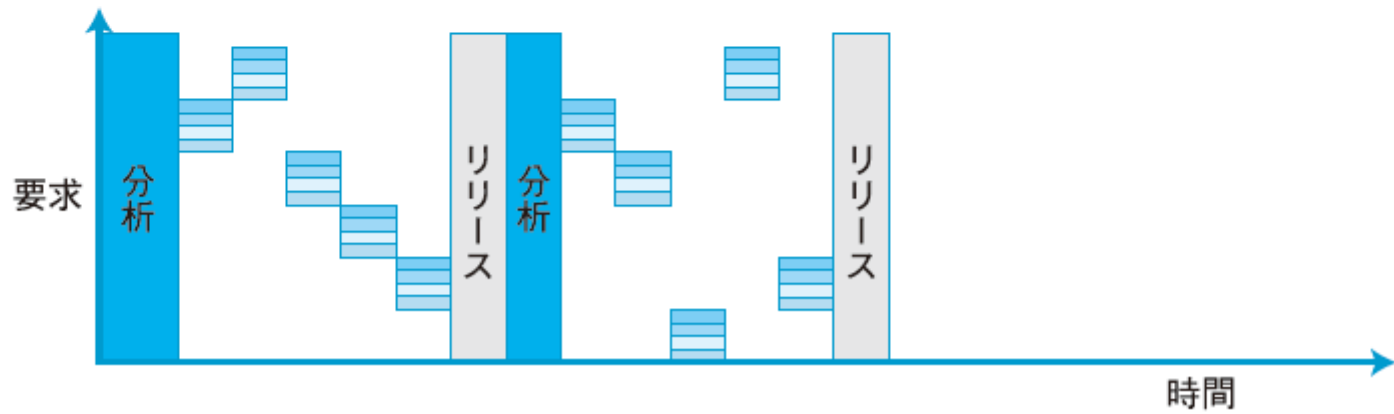
実装

従来のプロセスとの違い

●従来のプロセス



●アジャイルプロセス



ゴールへのコミットメント形成

➤ ゴールをシミュレーションし、自分たちがコミットする

設計着手

仕様一次Fix

機器仕様
検討

関連部署とのコミットメント

プロジェクトでのコミットメント

プロダクト詳細シート

プロダクト
検討

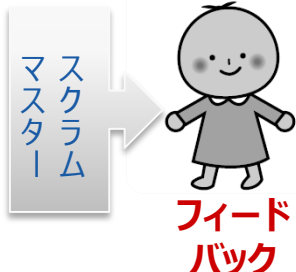
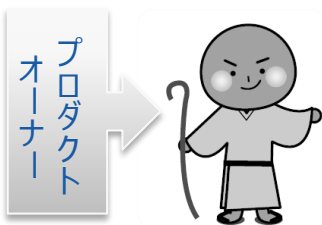
ゼブラパターンを表示する

リリース
スプリント

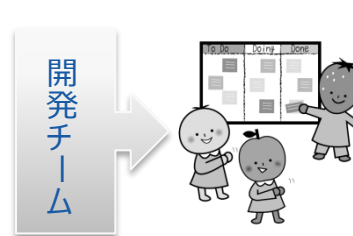
要件番号 14A4079
企画優先度 A
主担当U REC
推進L
難易度 B

どんな人が?	プロの動画カメラマン(静止画カメラマンの一部も含む)
どんな目的 (どんな気持ちで)	撮影対象の中に、白トビが発生していないかをリアルに確認する ライブビュー、撮影中も確認できる 再生中は表示されない
こうなる機能	設定した値以上の白が表示されている部分が、ライブビュー出力中に白黒縞々のパターンで マーキングされる(LCD、LVF、HDMI、外部モニターなどすべて)

事例



フィード
バック



チームメンバー全員で
ゴールへのコミットメント

Smiling Adventure 「お客様は誰？」



お客様と価値を共有するためには、その前にプロジェクトのメンバーで、お客様をしっかりとイメージして、全員で共有していくことが第一歩となります。

Smiling Adventure を使って「私たちにとってのお客様は誰？」のイメージを共有してみましょう。

★目的

誰にどんな価値を届けるのかをチームで共有する

★フォーマット

お客様は誰ですか？	どんな人ですか？	求めているものはどんなもの？

★進め方

- ①フォーマットに合わせて、みんなでディスカッションしてみましょう
- ②「お客様は誰ですか？」は、名前など特定できればOK
- ③「どんな人ですか？」「求めているものはどんなもの？」で具体化していきましょう
- ④1人だけとは限りません。いろいろなパターンを考えてみましょう。

★ヒント

- ・チーム全員でやるのがポイントです
- ・参加している全員が発言しましょう

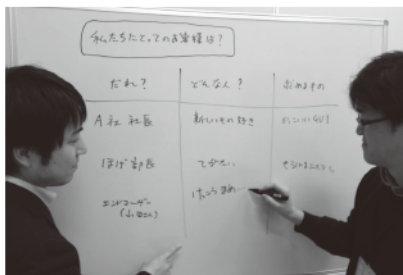


図3-10: ホワイトボードを使って実践してみよう

【お客様は誰？】

- ① フォーマットに合わせて、みんなでディスカッションをしてみましょう
- ② 「お客様は誰ですか？」は、名前など特定できればOK
- ③ 「どんな人ですか？」「求めているものはどんなもの？」を具体化していきましょう
- ④ 1人だけとは限りません
いろいろなパターンを考えてみましょう

Smiling Adventure 「お客様のハッピー」

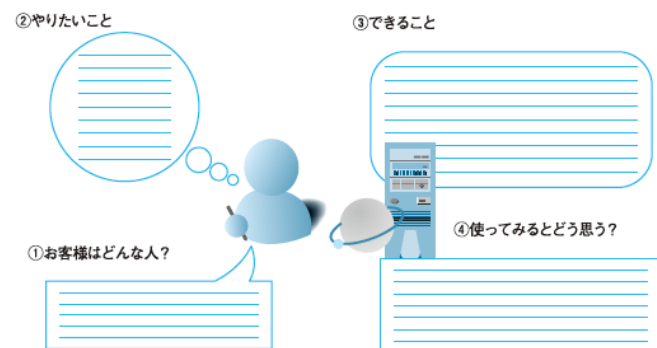


「お客様は誰？」のSmiling Adventureでお客様のイメージが共有できたら、そのイメージでお客様が本当にハッピーになるかどうかを再確認してみましょう。

★目的

「お客様のやりたいこと」と「開発者・システム・ソフトウェアができること」を比較し、最終的にお客様がどのようなハッピー（満足）が得られるのか？ をフォーマットで考えてみる

★フォーマット



★進め方

- ①「お客様は誰？」の結果を書きましょう
- ②お客様はどんなことをやってみたいと思っていますか？ 求めているものは？
- ③みなさんがつくり出すもので、どんなことが実現できますか？
- ④実際にお客様が使ってみると、どんな気持ちになりますか？

★ヒント

- ・チーム全員でやるのがポイントです
- ・できる限りシンプルにまとめて、共有しやすくしましょう
- ・この結果がプロジェクトのゴールになります

【お客様のハッピー】

- ① 「お客様は誰？」の結果を書きましょう
- ② お客様はどんなことをやってみたいと思っていますか？
求めているものは？
- ③ みなさんがつくり出すもので、どんなことが実現できますか？
- ④ 実際にお客様が使ってみると、どんな気持ちになりますか？

Smiling Adventure 「ストーリーテラー」



「お客様はハッピー」のSmiling Adventureはシステム全体での価値を共有しましたが、それぞれのストーリーを描く際には、まずはシンプルにまとめてみることから始めましょう。

★目的

それぞれのストーリーを1人で考えるのではなく、メンバーと一緒に書きながら語り合ってみましょう。

★フォーマット

このストーリーは

誰

が

目的

のために

アクション

するものです

★進め方

- ① フォーマットに合わせて、シンプルに書いてみましょう。
- ② 結果をもとに具体的なディスカッションにつなげてみましょう。

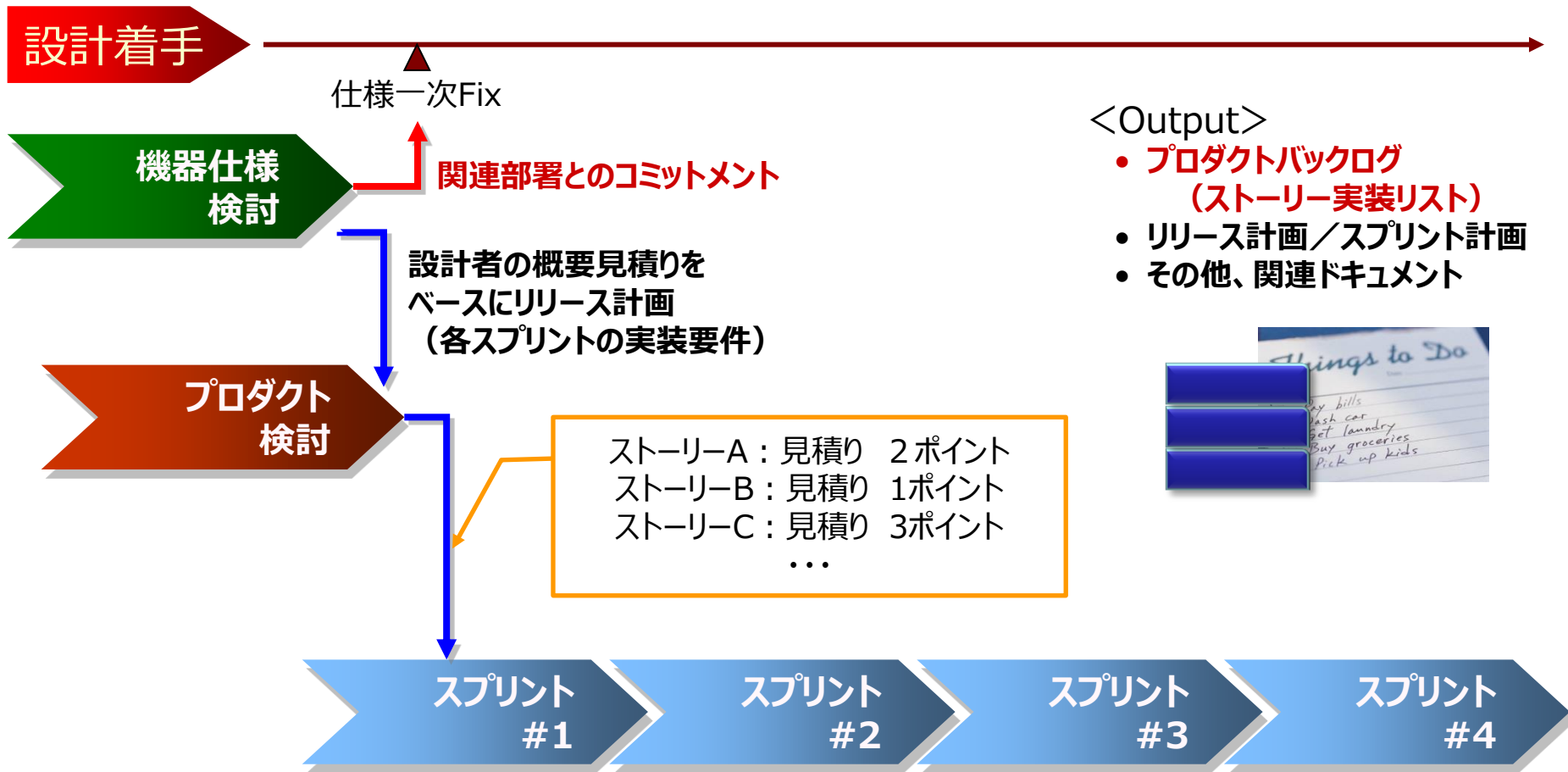
★ヒント

- ・できあがったものは、それぞれストーリーカードに展開できます。
- ・Smiling Adventure「お客様は誰?」「お客様のハッピー」を先にやっておくと、システム全体の価値が明確になって効果的です。

【ストーリーテラー】

- ① フォーマットに合わせて、シンプルに書いてみましょう
- ② 結果をもとに具体的なディスカッションにつなげてみましょう

スプリントとゴールのマッチング（見積）



各ストーリーはチーム全員で見積を実施し
実現可能かどうかを見極める

②一定間隔をリズムを継続させる

ゴール

- 価値をゴールにし、優先順位をコミットする
- リズムとゴールをマッチングさせる

リズム

- プロジェクトを一定間隔のリズムで区切る
- プロジェクトのベロシティを把握する

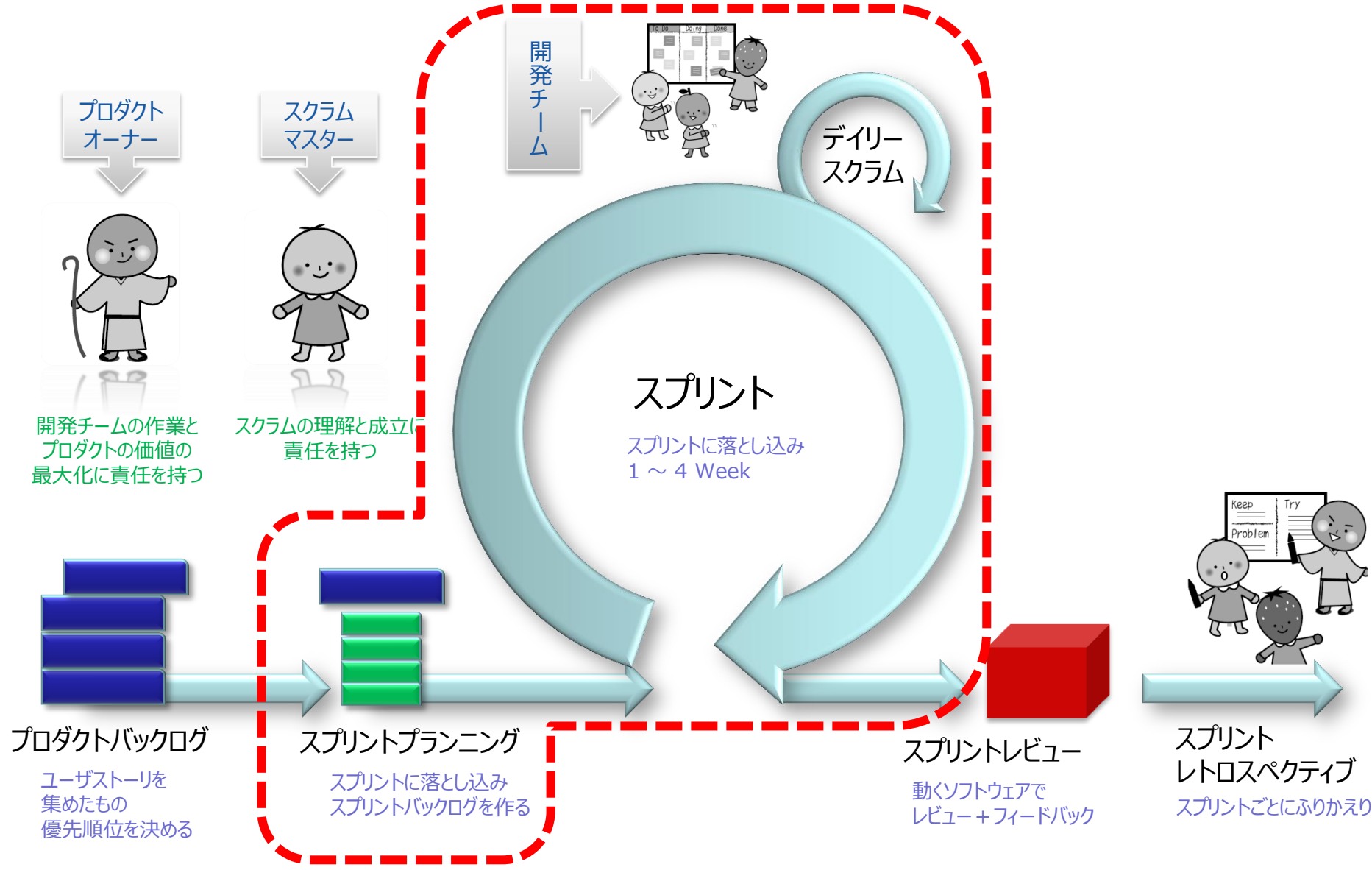
見える化

- 動くもの + 状況の可視化でリズムを伝播させる
- フィードバックにより変化を取り込む

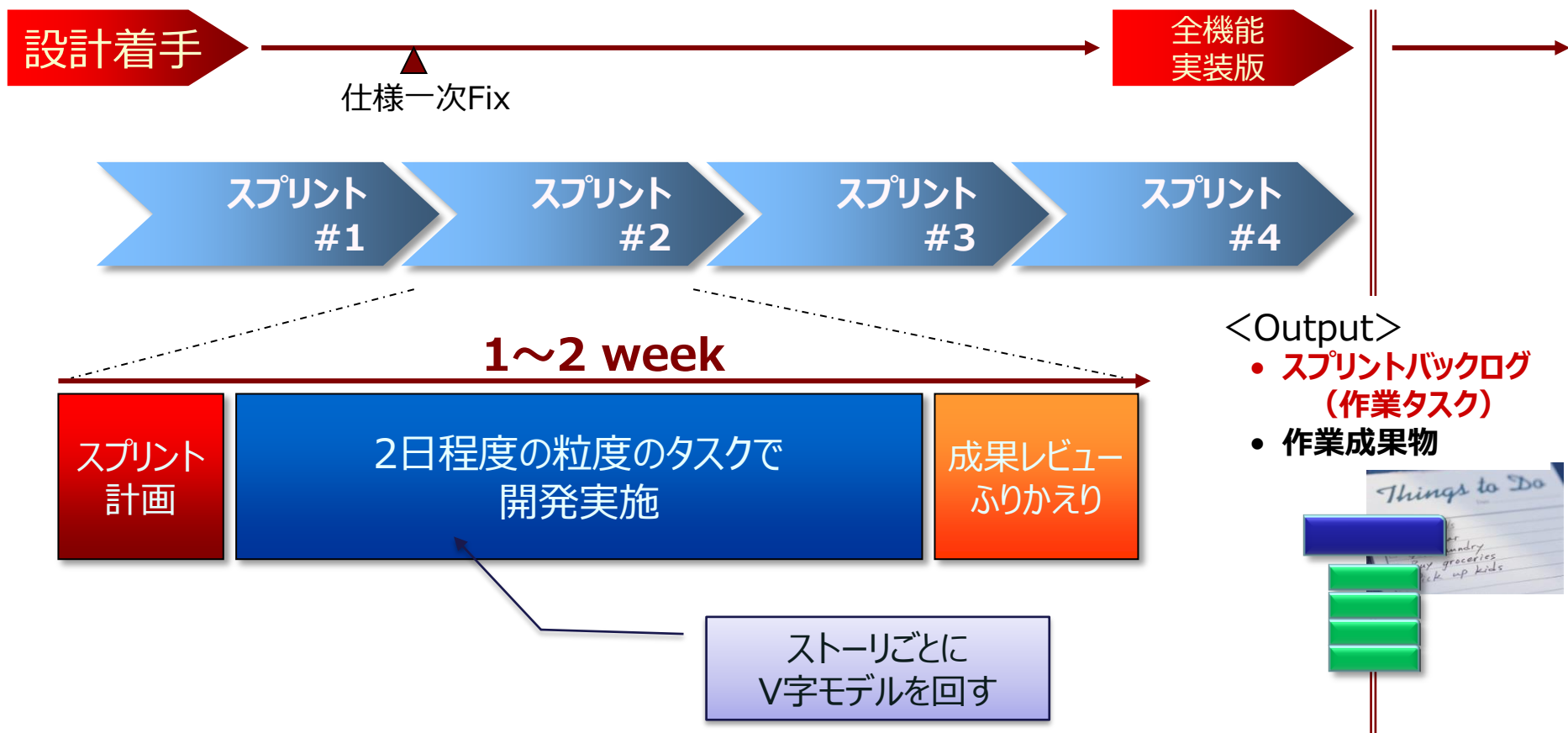
自律

- 自分たちでふりかえる
- 自分たちで変えていく





ストーリーをさらに細かいタスクに細分化する



プロダクトバックログをスプリントバックログに細分化することは基本的にWBSと似ているが、リズムが一定間隔なことが重要



テスト駆動開発 (Test Driven Development)

- コーディングミスが発生しにくい
- コーディングでの誤り混入から発見までの時間が短い
(数分～数10分程度)
- コードの可読性が向上する

デイリー
スクラム

スプリント

ペアプログラミング

- コードがきれいになる
(レビュー率100%)
- システムやコード、ツールなどの知識を共有できる
- ペアの組み方によっては、トレーニングの効果を得られる
- コミュニケーションが活発になり一体感を生み出す
- 1人で悩んで停止してしまうことが少なくなり作業が着実に進む

コードの共同所有

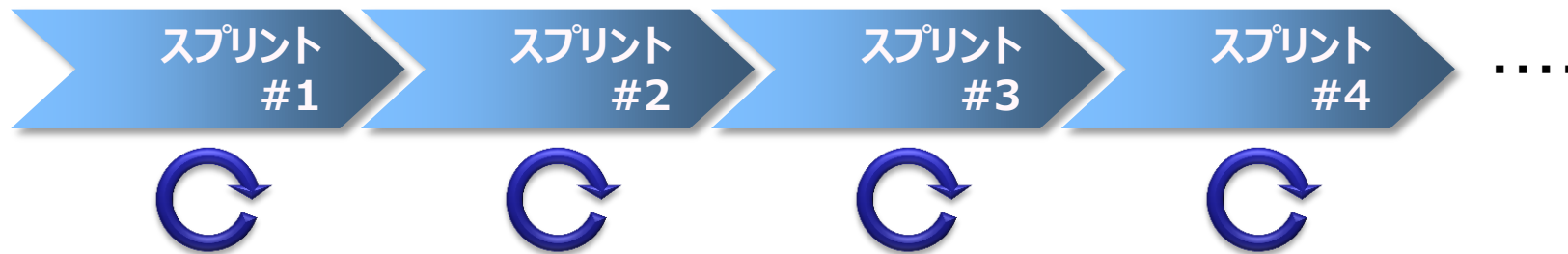
サインアップ



これ、私が
やりまーす!

**その他にもノウハウはたくさん
作るもの、プロジェクトに合わせ
自分たちでプラクティスを活用していく**

プロジェクトのベロシティを把握する



一つのスプリントで
実現できる作業量（ベロシティ）を
自分たちで把握することができる



一つのストーリー（要件でもあり、ファンクションでもある「実現すべき価値」）を
日程や規模、ページ数などではなく

設計、レビュー、テストなど
全ての作業を合わせた
『ストーリーポイント』として見積る

ストーリーポイントは開発にかかわるチーム全員で見積る
カードを使った「プランニングポーカー」を活用し
全員でコミュニケーションしながら見積を実施する場合もある



(画像提供：楽天株式会社さま)

③常に状況を見えるようにして変化へ対応する

ゴール

- 価値をゴールにし、優先順位をコミットする
- リズムとゴールをマッチングさせる

リズム

- プロジェクトを一定間隔のリズムで区切る
- プロジェクトのベロシティを把握する

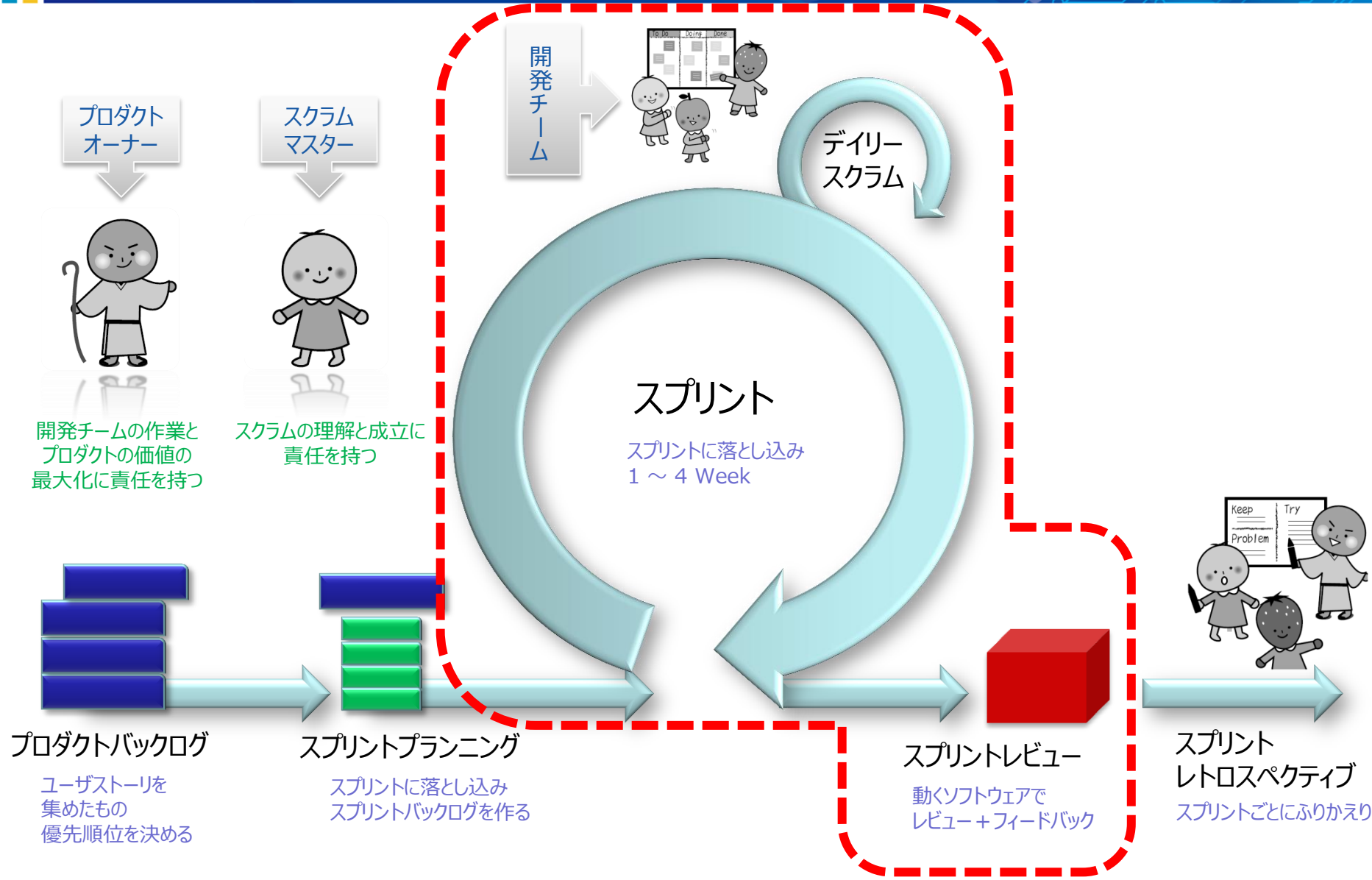
見える化

- 動くもの + 状況の可視化でリズムを伝播させる
- フィードバックにより変化を取り込む

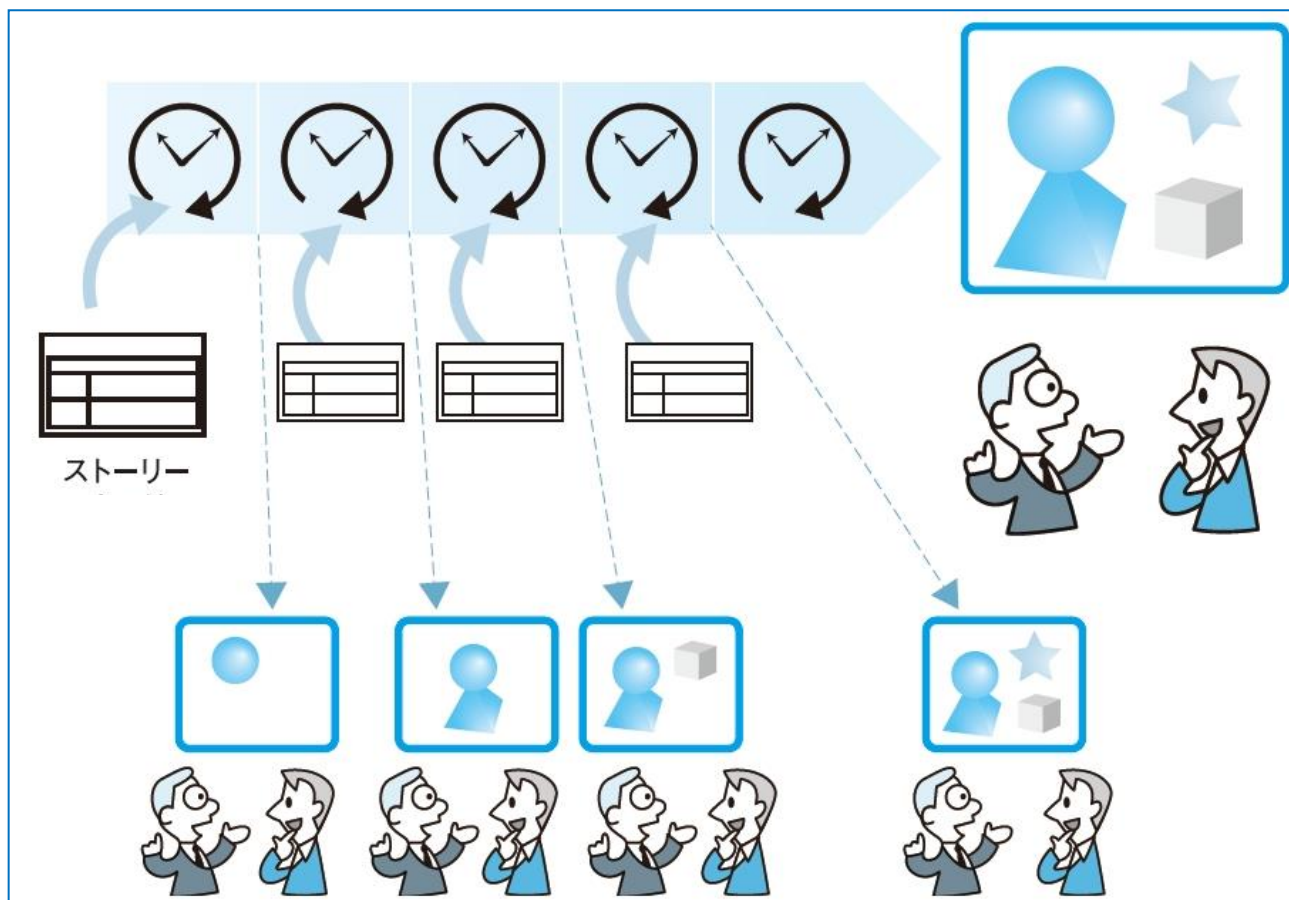
自律

- 自分たちでふりかえる
- 自分たちで変えていく





価値の最大化するためのフィードバック

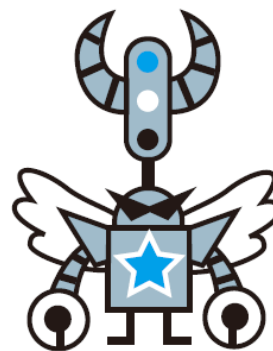
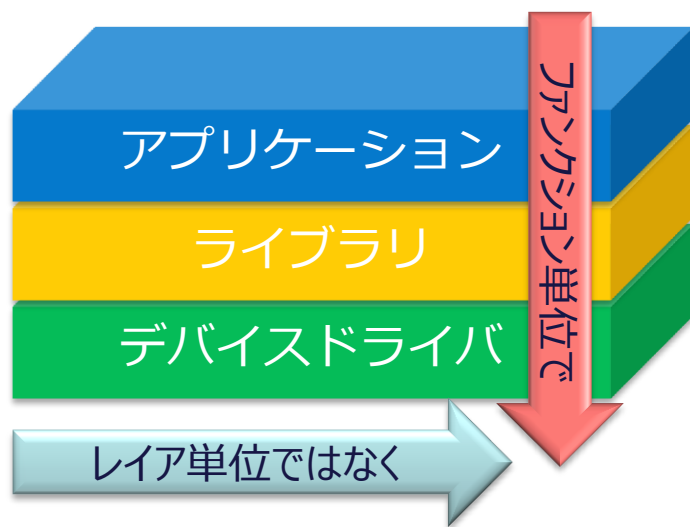


**アジャイルは単に早い・高品質なだけではなく
フィードバックにより価値を最大化していかなければ効果がない**

実際にうごくもので価値を確認する

➤ スプリントごとに価値をフィードバックするには？

- ☺ 価値が確認できるレベルに動作していること
- ☺ シンプルに設計し、リファクタリングができること
- ☺ 今必要としていないものをムダに作りこまないこと

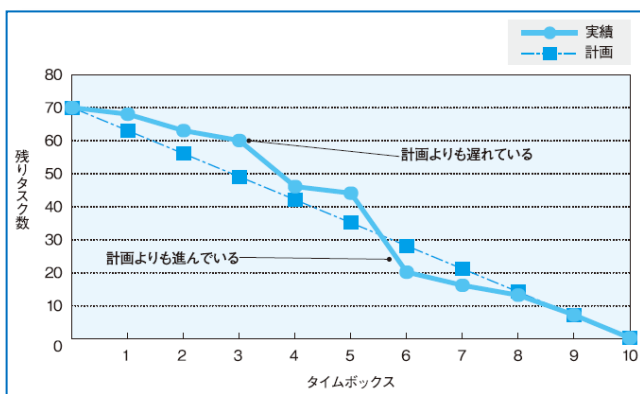


様々な見える化により開発の透明性を実現する



かんばん

	ToDo 未実施	Doing 実施中	Done 完了
Aさん	■ ■	■	■ ■
Bさん		■	■ ■ ■
Cさん	■ ■	■	



バーンダウンチャート

ニコニコカレンダー (2/18-23)

	18	19	20	21	22	23(土)
細谷	😊	😊	😞	😊	😊	
西川	😊	😊	😞	😊	😊	😞
前川	😊	😞	😊	休	😊	
鈴木	😊	休	😊	😞	😊	

ニコニコカレンダー

プロジェクトの見える化

Redmineでのチケット駆動開発

進捗確認



ゴールごとのチケット
完了状況の確認

リアルタイムにバーンダウンチャートを見
ることで進捗状況の共有化を図る

各チケットの状況で
どの作業が残っている
か簡単に把握できる

概要 活動 **ロードマップ** チケット 新しいチケット ガントチャート カレンダー

ロードマップ

S3_機能A

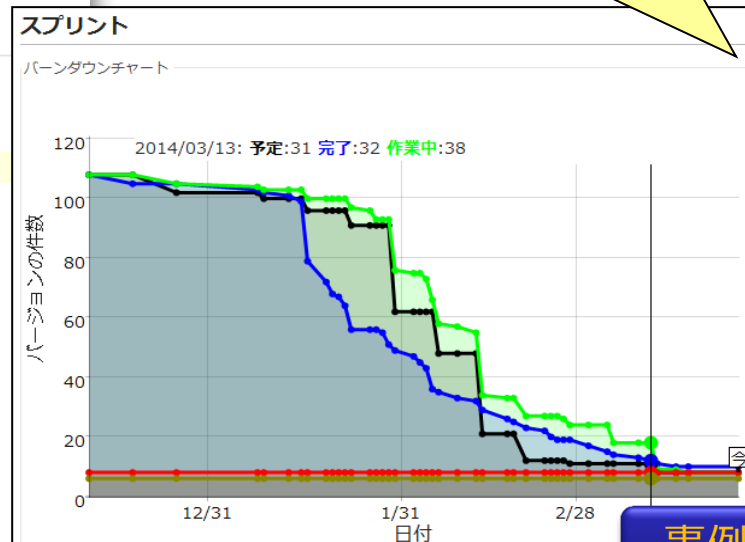
3日 遅れ (2011/09/09)

72%

11件完了 (69%) 5件未完了 (31%)

関連するチケット

- コンポーネントA - 設計・実装 #42614: 機能A-ソフト仕様書作成
- コンポーネントA - 設計・実装 #42635: 機能A-結合テスト
- コンポーネントA - 設計・実装 #42646: 機能A-ソフト仕様検討
- コンポーネントA - 設計・実装 #42647: 機能A-詳細設計
- コンポーネントA - 設計・実装 #42648: 機能A-実装
- コンポーネントA - 設計・実装 #42649: 機能A-単体テスト
- コンポーネントB - 設計・実装 #42667: 機能A-ソフト仕様書作成
- コンポーネントB - 設計・実装 #42622: 機能A-結合テスト
- コンポーネントB - 設計・実装 #42640: 機能A-詳細設計
- コンポーネントB - 設計・実装 #42641: 機能A-実装
- コンポーネントB - 設計・実装 #42642: 機能A-単体テスト
- コンポーネントC - 設計・実装 #42627: 機能A-実装
- コンポーネントC - 設計・実装 #42628: 機能A-単体テスト
- コンポーネントC - 設計・実装 #42629: 機能A-結合テスト
- コンポーネントC - 設計・実装 #42644: 機能A-ソフト仕様
- コンポーネントC - 設計・実装 #42645: 機能A-詳細設計

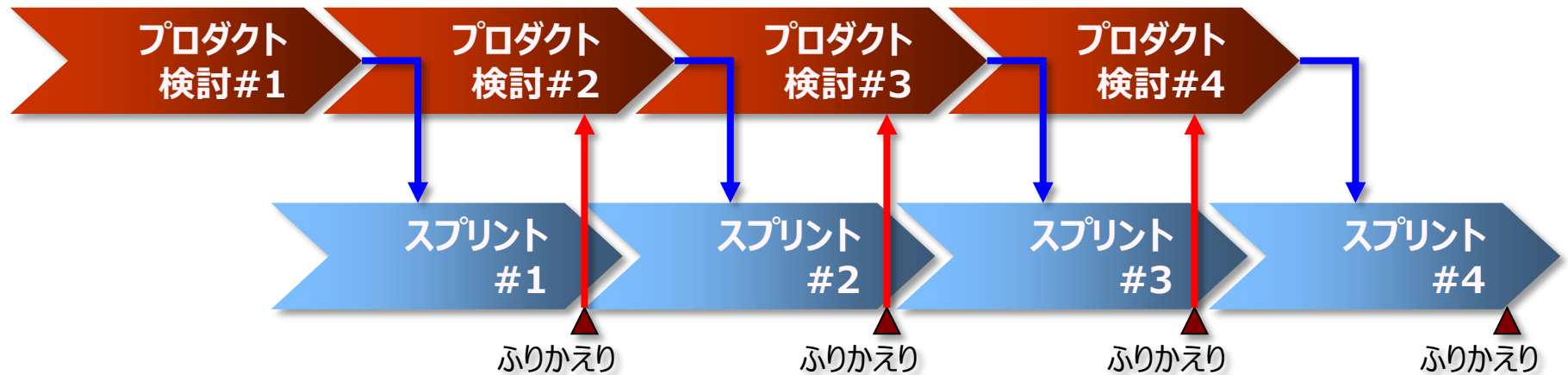


事例

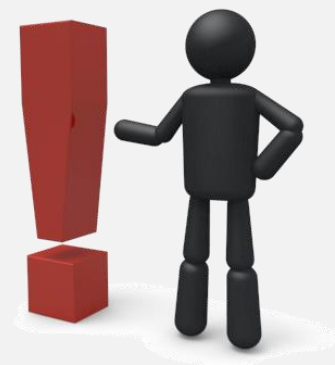
ゴールは固定されるのではなく継続的にふりかえる

設計着手

仕様一次Fix



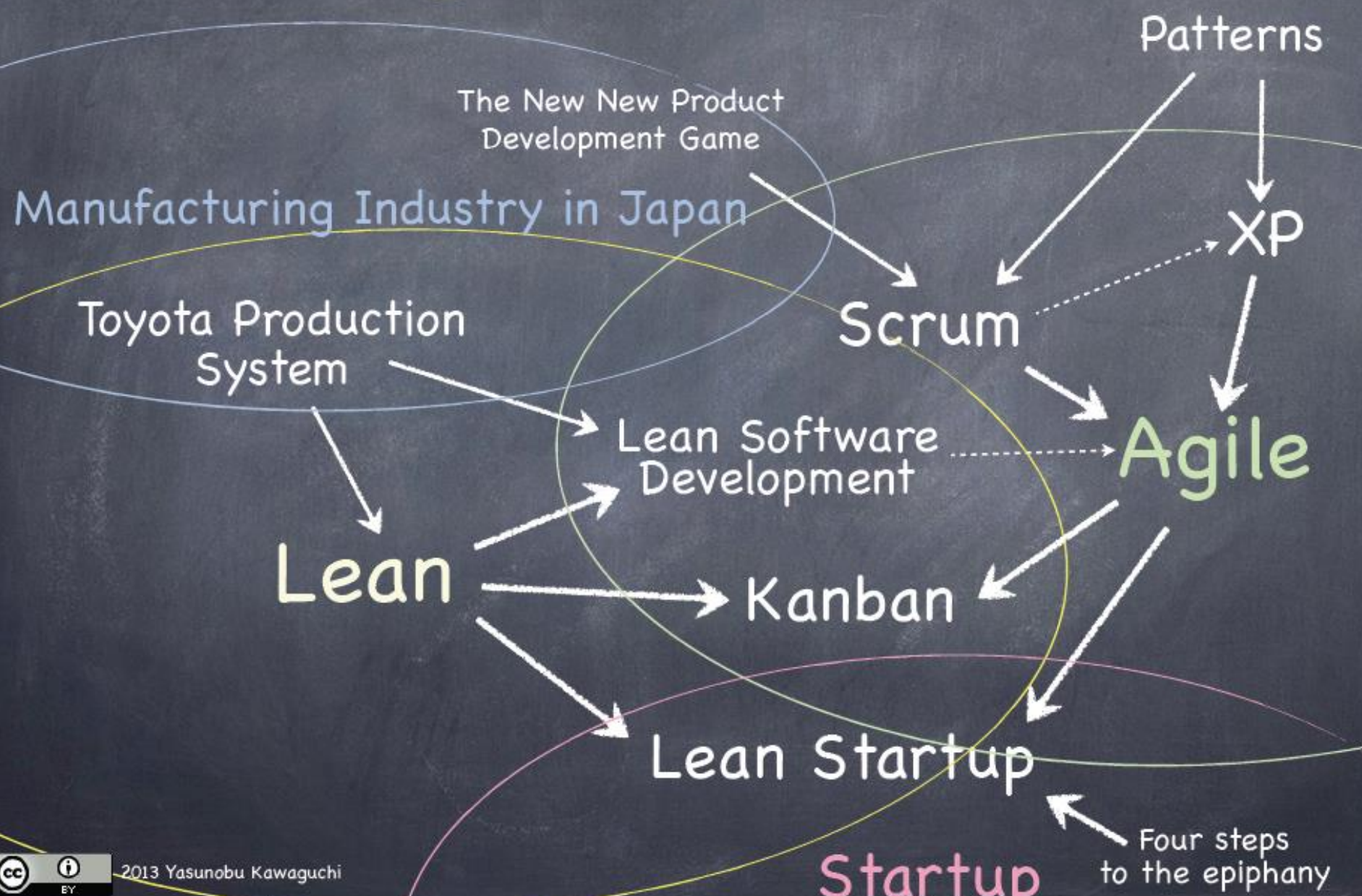
動くもの（見えるもの）を作り上げ、価値を確認することで次のスプリントにフィードバックさせていく



— 参 —

プロジェクトの価値を創る

Agile and Lean



野中先生提唱のSECIモデル

知識創造は暗黙知と形式知の相互変換運動である

暗黙知 (Tacit Knowledge)

言語・文章で表現するのが難しい
主観的・身体的な経験知

特定の文脈ごとの経験の反覆によって
体化される
思考スキル(思い・メンタル・モデル)や
行動スキル(熟練・ノウハウ)

形式知 (Explicit Knowledge)

言語・文章で表現できる
客観的・理性的な言語知

特定の文脈に依存しない一般的な
概念や論理(理論・問題解決手法・
マニュアル・データベース)

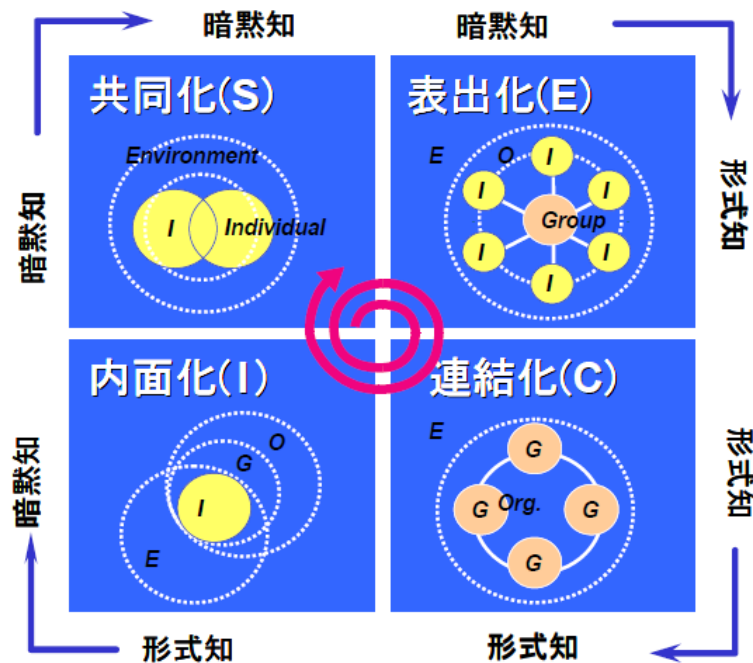
相互作用の スパイラルアップ

アナログ知ーデジタル知の動的総合

© Nonaka I.

身体・五感を駆使、
直接経験を通じた
暗黙知の共有、創出

対話・思慮による概念・
デザインの創造
(暗黙知の形式知化)



© Nonaka I. & H. Takeuchi

形式知を行動・実践の
レベルで伝達、新たな
暗黙知として理解・学習

形式知の組み合わせによる
新たな知識の創造
(情報の活用)

- I = 個人
- G = 集団
- O = 組織
- E = 環境



スクラムは、経験的プロセス制御の理論（経験主義）を基本にしている。経験主義とは、実際の経験や既知に基づく判断によって知識が獲得できるというものである。スクラムでは、反復的で漸進的な手法を用いて、予測可能性の最適化とリスクの管理を行う。

「スクラムガイド」より

© 1991-2013 Ken Schwaber and Jeff Sutherland, All Rights Reserved

<https://www.scrum.org/Portals/0/Documents/Scrum%20Guides/2013/Scrum-Guide-JA.pdf>

◆ スクラムチーム

スクラムチームは、プロダクトオーナー・開発チーム・スクラムマスターで構成される。スクラムチームは自己組織化されており、機能横断的である。**自己組織化チームは、作業を成し遂げるための最善の策を、チーム外からの指示ではなく、自らが選択する。**機能横断的チームは、チーム外に頼らずに作業を成し遂げる能力を持っている。スクラムにおけるチームのモデルは、柔軟性・創造性・生産性に最適化されたものとなっている。

「スクラムガイド」より

© 1991-2013 Ken Schwaber and Jeff Sutherland, All Rights Reserved

<https://www.scrum.org/Portals/0/Documents/Scrum%20Guides/2013/Scrum-Guide-JA.pdf>

④自分たち自身の価値も向上させる

ゴール

- 価値をゴールにし、優先順位をコミットする
- リズムとゴールをマッチングさせる

リズム

- プロジェクトを一定間隔のリズムで区切る
- プロジェクトのベロシティを把握する

見える化

- 動くもの+状況の可視化でリズムを伝播させる
- フィードバックにより変化を取り込む

自律

- 自分たちでふりかえる
- 自分たちで変えていく



プロダクト
オーナー



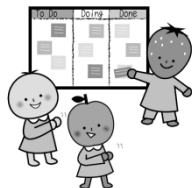
開発チームの作業と
プロダクトの価値の
最大化に責任を持つ

スクラム
マスター



スクラムの理解と成立に
責任を持つ

開発
チーム



デイリー
スクラム

スプリント

スプリントに落とし込み
1 ~ 4 Week

プロダクトバックログ

ユーザストーリーを
集めたもの
優先順位を決める

スプリントプランニング

スプリントに落とし込み
スプリントバックログを作る

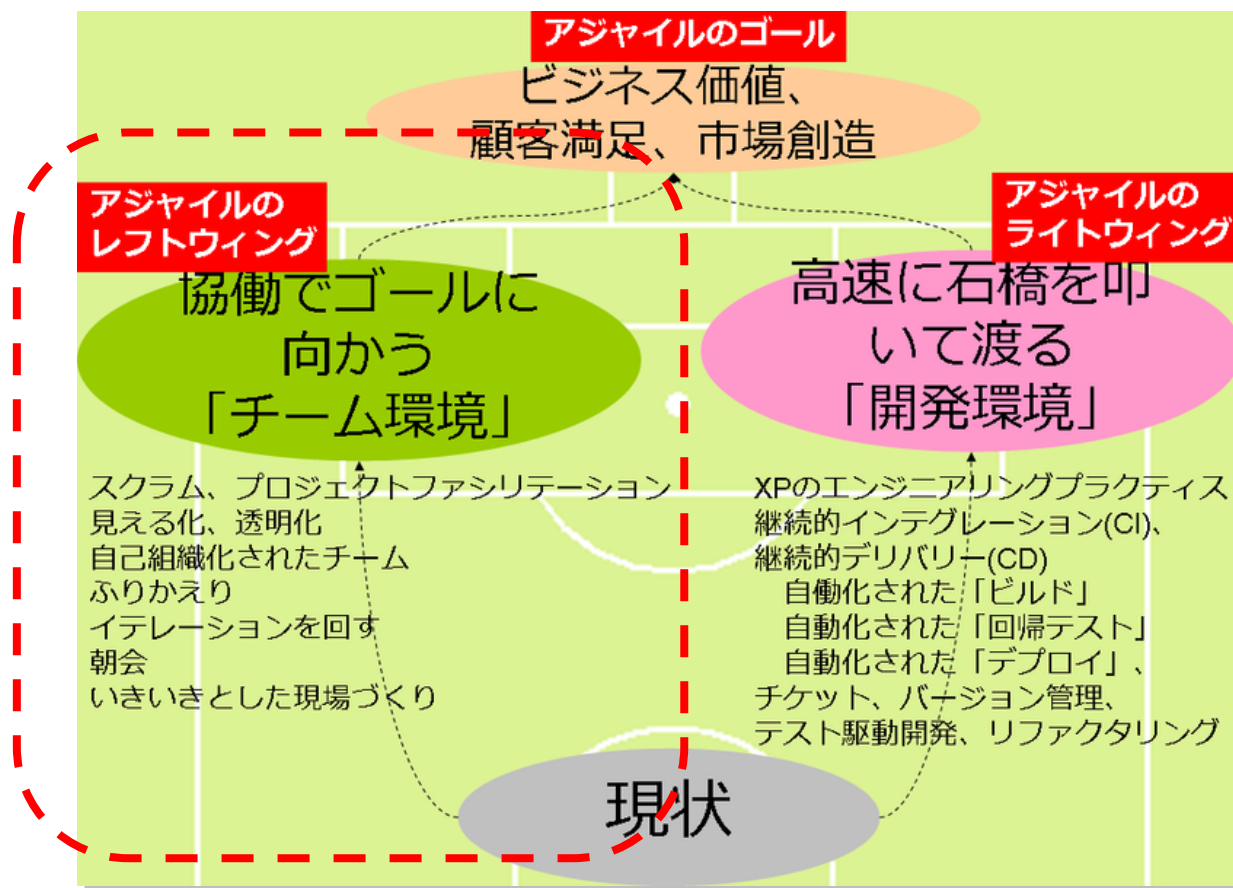
スプリントレビュー

動くソフトウェアで
レビュー+フィードバック

スプリント
レトロスペクティブ

スプリントごとにふりかえり

アジャイルのレフトウィング



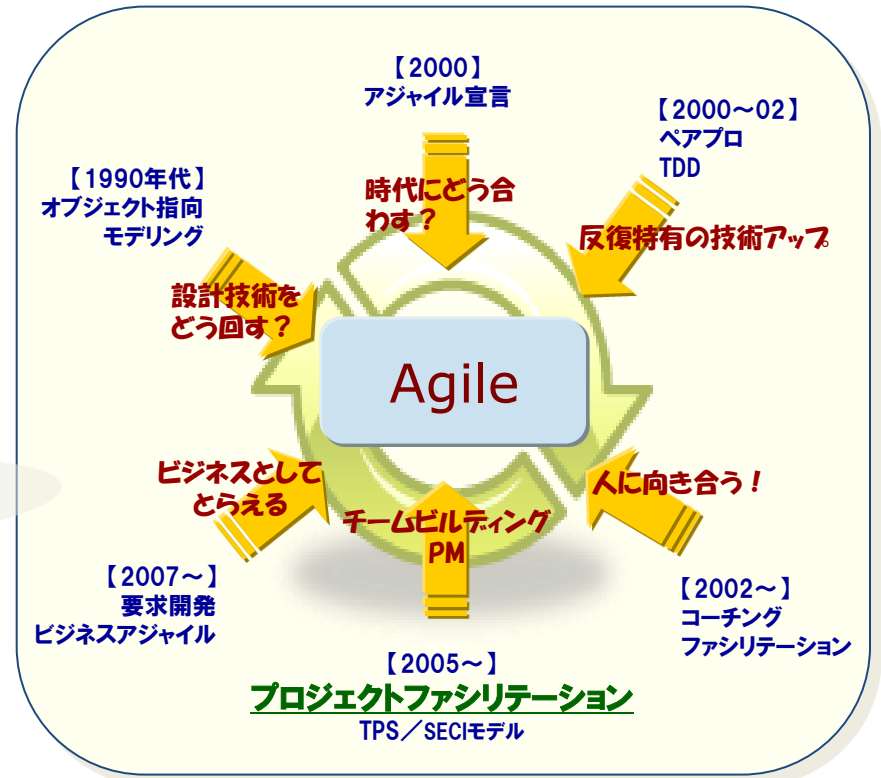
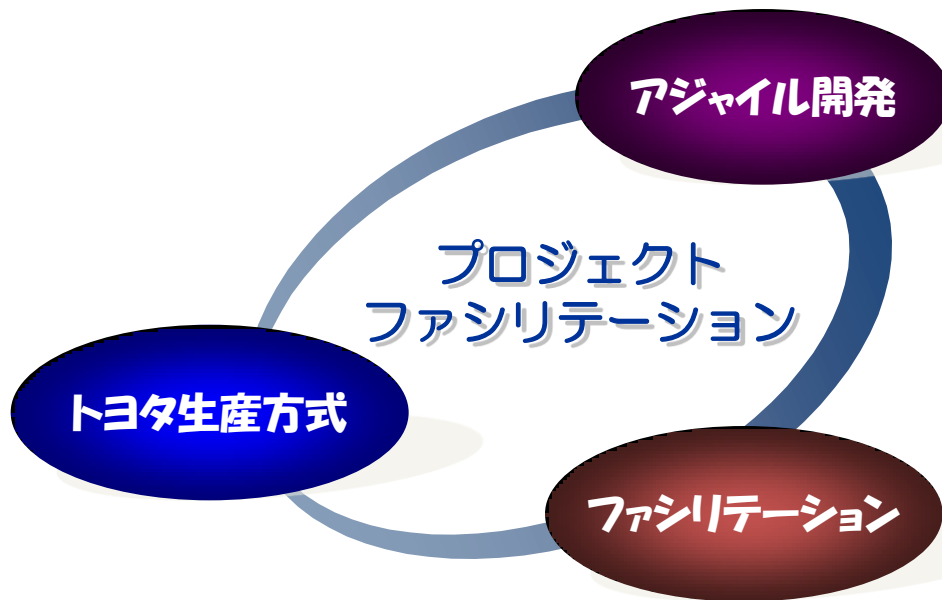
平鍋さんブログ「An Agile Way」より
<http://blogs.itmedia.co.jp/hiranabe/2012/09/rightwing-and-leftwing-of-agile.html>

プロジェクトファシリテーションとは？

【プロジェクトファシリテーション】とは？

プロジェクトマネジメント(PM)が重要であることは昨今強く言われています。PMが「**計画達成のマネジメント**」に重点を置くのに対してPFは「**参加者の協調の場作り**」に重点を置いています。PMは、計画の立案と実行、差異に注目した管理が中心で、どちらかと言うと「**コマンド・コントロール型**」のマネジメントスタイルが背後にあります。これに対してPFは、**その場その場の変化に対応し、チームが協力し合って創発的に成果を出していく、「リーダーシップ・コラボレーション型」**の新しいチーム作りの形です。

(オブラブ公式Webサイトより <http://www.objectclub.jp/community/pf/>)



アジャイルの変遷とプロジェクトファシリテーション

価値

コミュニケーション
行動
気づき
信頼関係
笑顔

原則

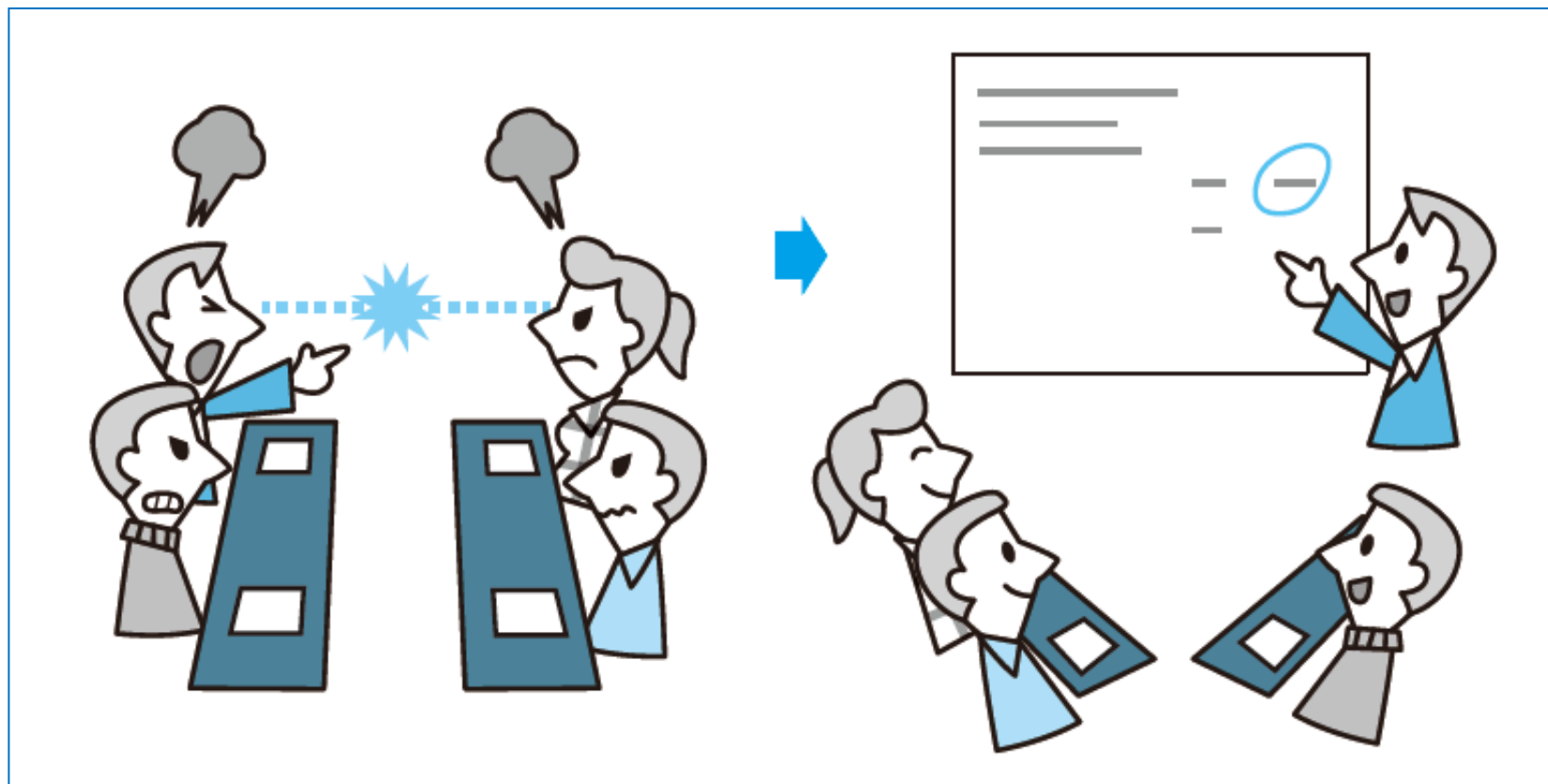
見える化
リズム
名前付け
問題vs.私たちの構図
カイゼン

フラクティス

朝会
かんばん
KPT
ペアボード
ニコカレ
アイスブレイク
偏愛マップ
MindMap

etc.

対立構図から「問題対私たち」へ

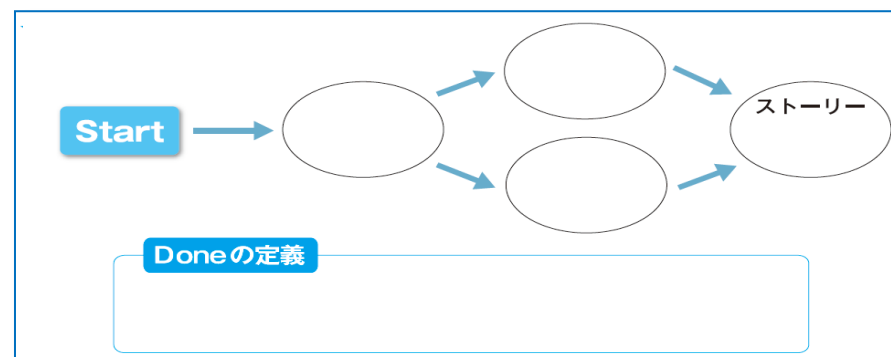






- ストーリへの「完了条件を定義していない」「共有していないこと」がズレの始まりに・・・

- ☺ どんなテストを完了していますか？
- ☺ レビューは完了していますか？
- ☺ お客様視点で動作できますか？



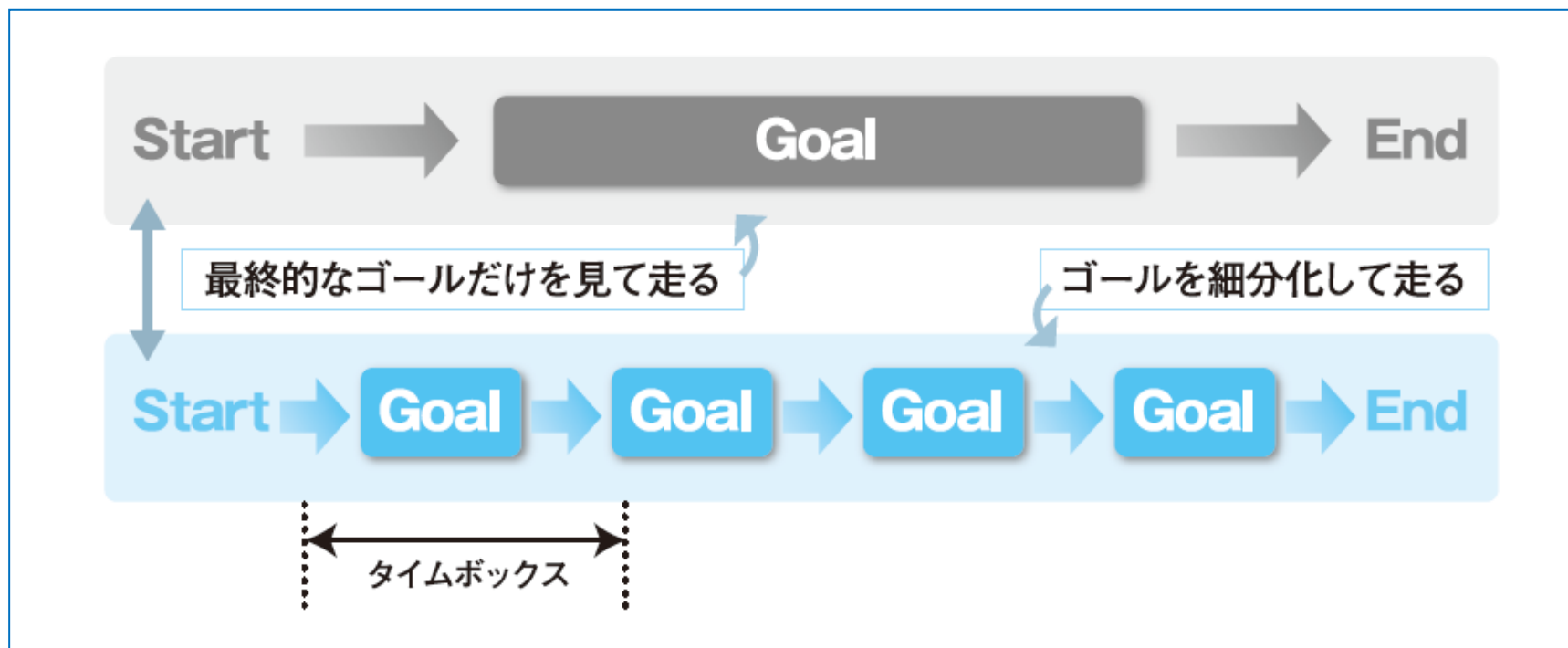
一つひとつのゴールもずれていないか
きちんと定義して共有する

- ▶ コミットメントは
個人の目的を、チームの目標達成に融合させて
最大のパフォーマンスを発揮するためのもの



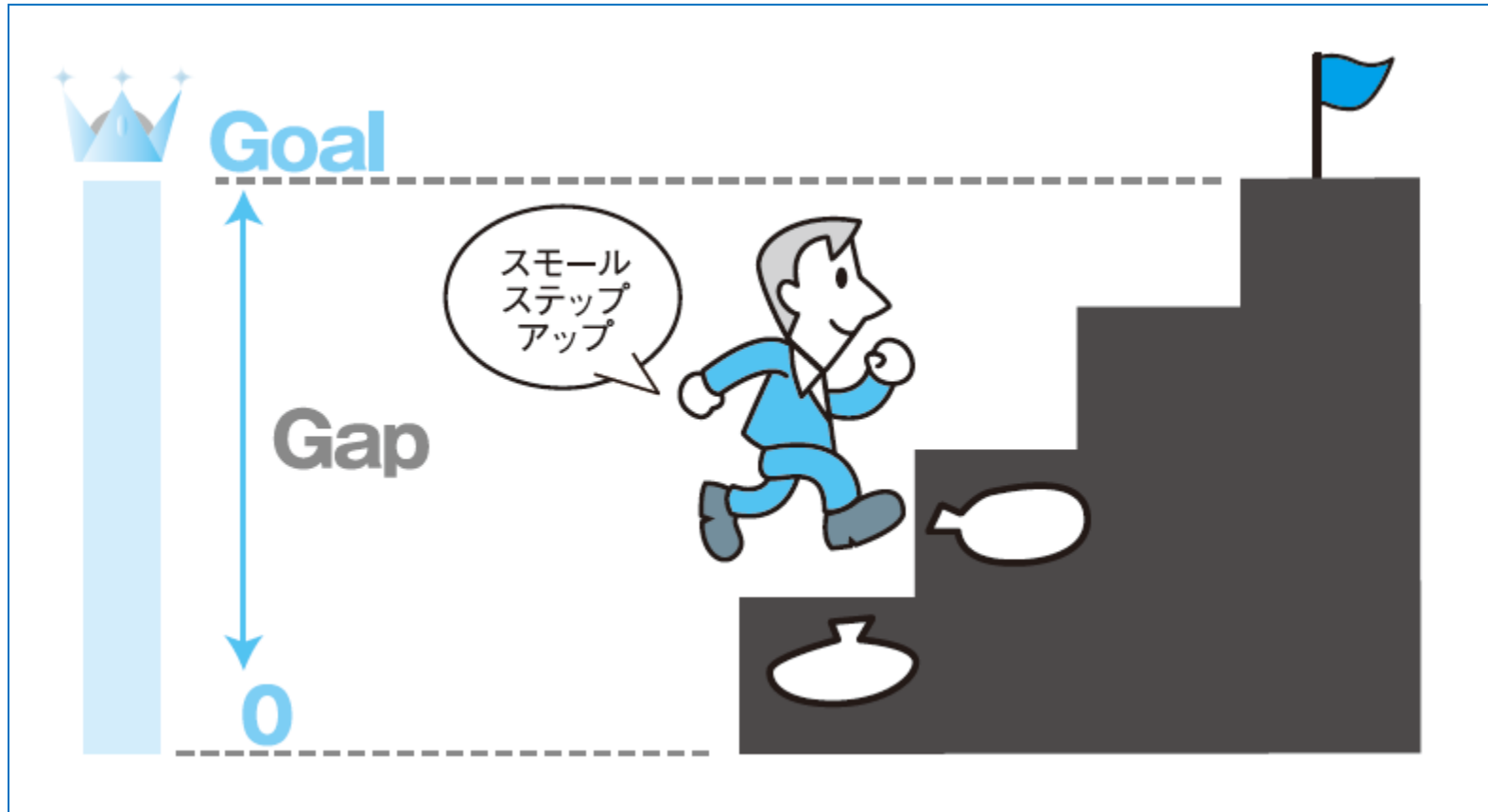
- ▶ **メンバー全員が**
 - ▶ **共通のゴールに向けて進んでいる**
 - ▶ **効果的な協同作業を行うためのプロセスをつくるか選び、活用する**
 - ▶ **成果物やプロセスは素早いフィードバックで常に改善していく**

タイムボックスというリズムの効果

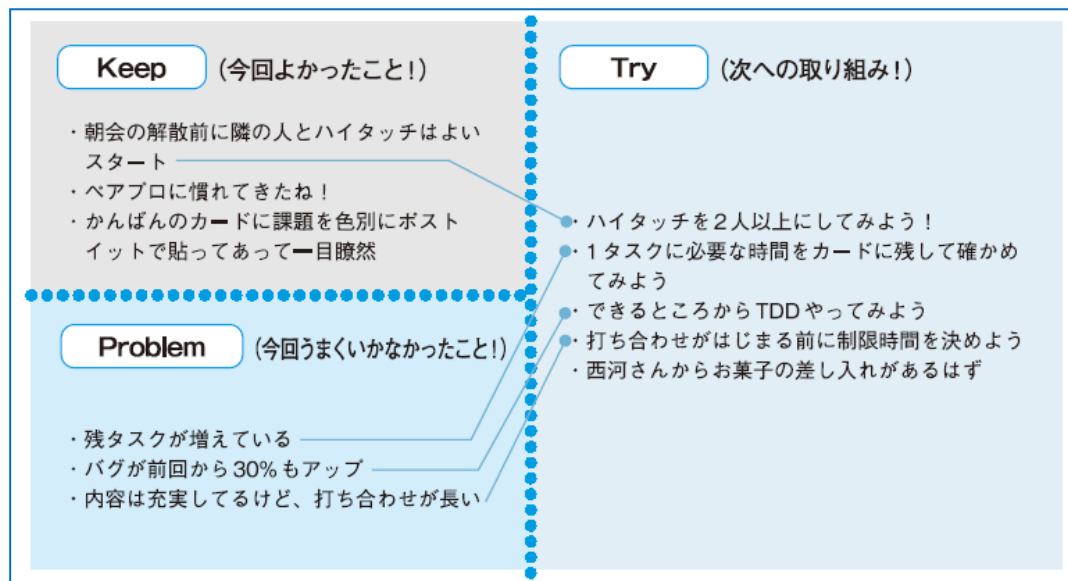


「変わらない時間の測定基準」を使って
プロジェクトのパワーを測り、引き出していく

ジャンプして届くゴールの繰り返し



KPTは成長を促すツール



😊 Keepで、チームとしての共感を得る

✦ よかったことを褒めあうと、チームを信頼できる

😞 Problemを共有する

✦ 問題を誰かのせいにするのではなく、チームの問題であると認識する

👍 そして、みんなでTry!

✦ 誰かがやってくれる、のではなく、自分が、そして誰もがやる!



定期的に成長するしくみ

- 一定期間の短いリズムでふりかえる
- 習慣にして継続させる



「歩み」を実感しつづける

- 成長していることは、自信につながる





— 四 —

アジャイルになる！

トップダウンか ボトムアップか？

トップダウン



ボトムアップ



サインアップ

リファクタリング

ペアプロ

KPT

TDD

朝会

ニコカレ

チームの活動を
引き出す

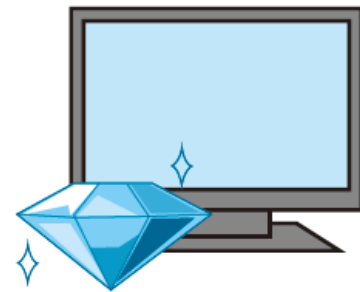
見える化と改善

プランニングポーカー

チケット駆動開発



顧客のシステム・製品の
ビジネス価値を最大化

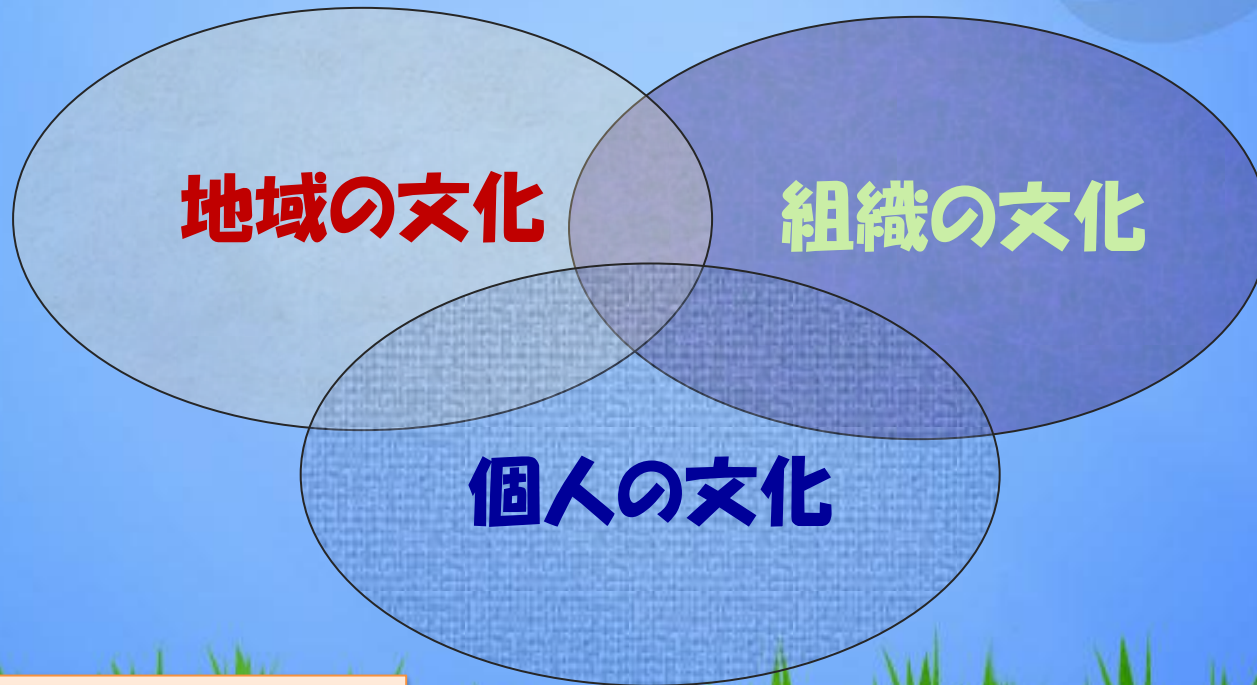


アジャイルとは、
お客様のビジネス価値を最大化するための
「考え方」や「姿勢」のこと

「みんな違ってみんないい」

金子みすず 「わたしと小鳥と鈴と」より

一人ひとりが、
いろいろな「文化」を背負っています



一人ひとり違ったメンバを
信頼できていますか？

自分を開いて伝えよう！

相手を知りながら
自分自身を知ってみよう！



成長を続けることで プロジェクトの価値をアップ

- 😊 自分たちを成長させていく
- 😊 成長を感じることでより活発化していく
- 😊 プロジェクト自体が「自律」しはじめる！
- 😊 自律しながら「自分たちの道」を探す



プロジェクト価値をアップすることが
アジャイルの最大メリット



~~Do Agile~~



Be Agile

キーワードは「自己組織化」

変化を受け入れるのではなく 社会に対して変化を生み出していく

