

# みんなdeScrum!!

細合 晋太郎

# 自己紹介

- 細合 晋太郎（ほそあい しんたろう）
- 九州大学大学院システム情報科学研究院
- 情報知能工学部門 社会情報システムコース
- enPiTプロジェクト専任研究員
  
- 組込み周りの教育がお仕事
- 組込み向けMDDが研究対象
  
- 趣味：ものづくり（日曜大工，電子工作，プログラミング等），お茶，カフェ巡り，SF読み

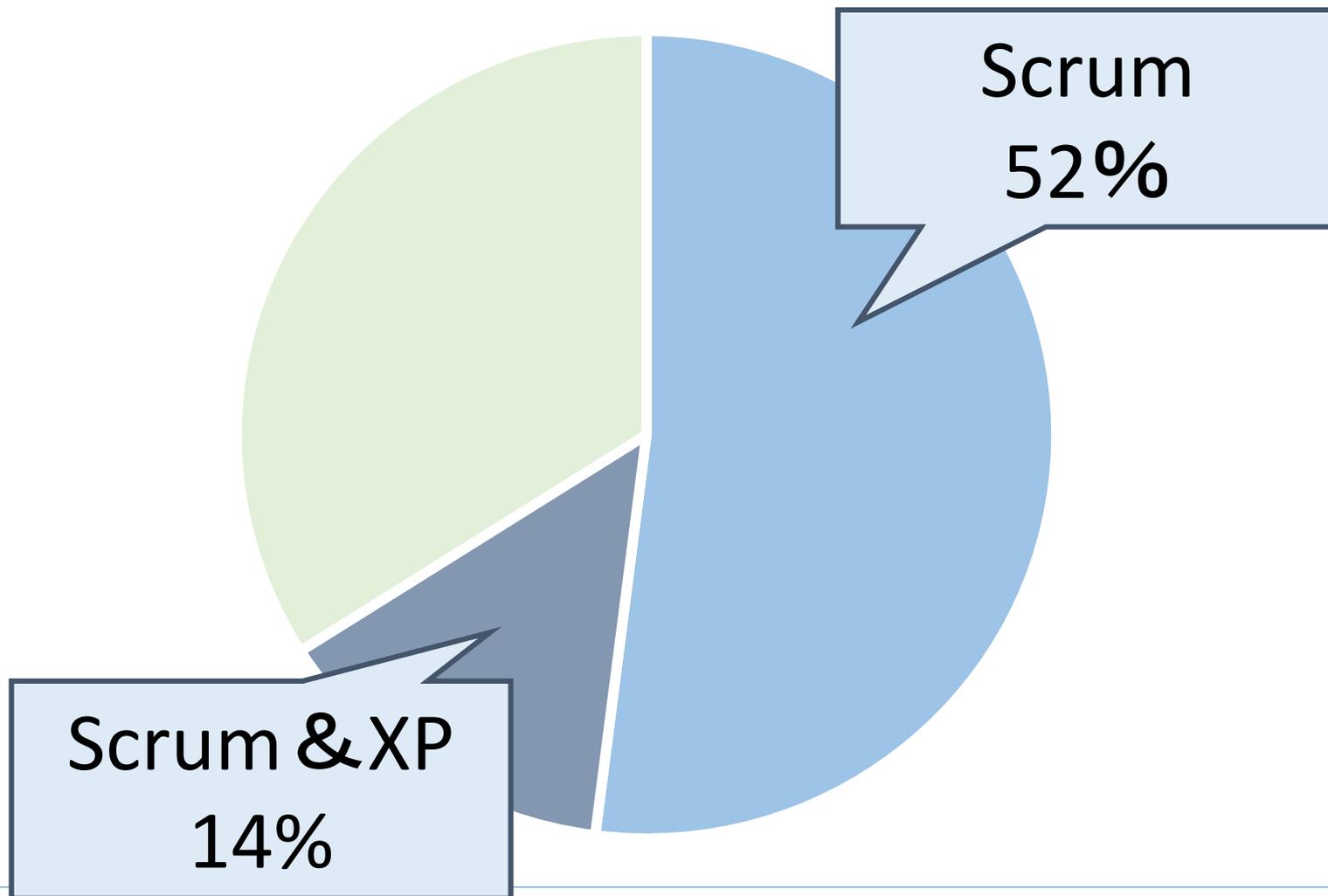
# Agile開発とは： アジャイルソフトウェア開発宣言

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だそうとしている。  
この活動を通じて、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。

# Agile ≙ Scrum



出典: Vision One社“State of Agile Development”  
conducted between July 22nd and November 1st, 2011.

# Scrumとは

- 複雑なプロダクト開発の管理を目的とした反復・漸増型のプロセス**フレームワーク**
  - どう実装するかはユーザに任されている
- 目的
  - 複雑で変化の激しい問題に対応する
  - 価値の高いプロダクトを生産的かつ創造的に届けることを目指す
- 特徴
  - 軽量
  - 理解が用意
  - 習得は非常に困難

# Scrumの価値

## 自己組織化

チーム全員でゴールを共有し、各自がゴールに向けてベストを尽くす

## 検査と適応(フィードバック)

プロジェクトがゴールに向かっているかを検査し、適応する。継続的な改善

## 透明性の確保

ゴールや進捗, 解決すべき問題を可視化出来る

# Scrumのルール

Scrum Master

スクラムマスター



Teamに指示や命令を  
する役割**ではない**

Teamの自己組織的な振  
舞いを促す

Teamが守るべきルールを  
守らせる. 調整役

Scrumのインストラクタ,  
縁の下の力持ち

Product Owner

プロダクトオーナー



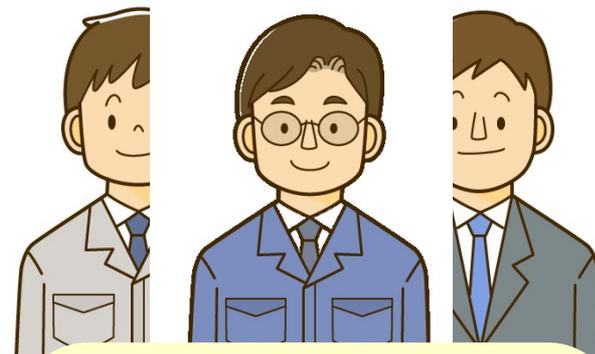
開発するプロダクトの責  
任を持つ

顧客の要求を開発者に  
伝える

プロダクトバックログの管  
理

Developer

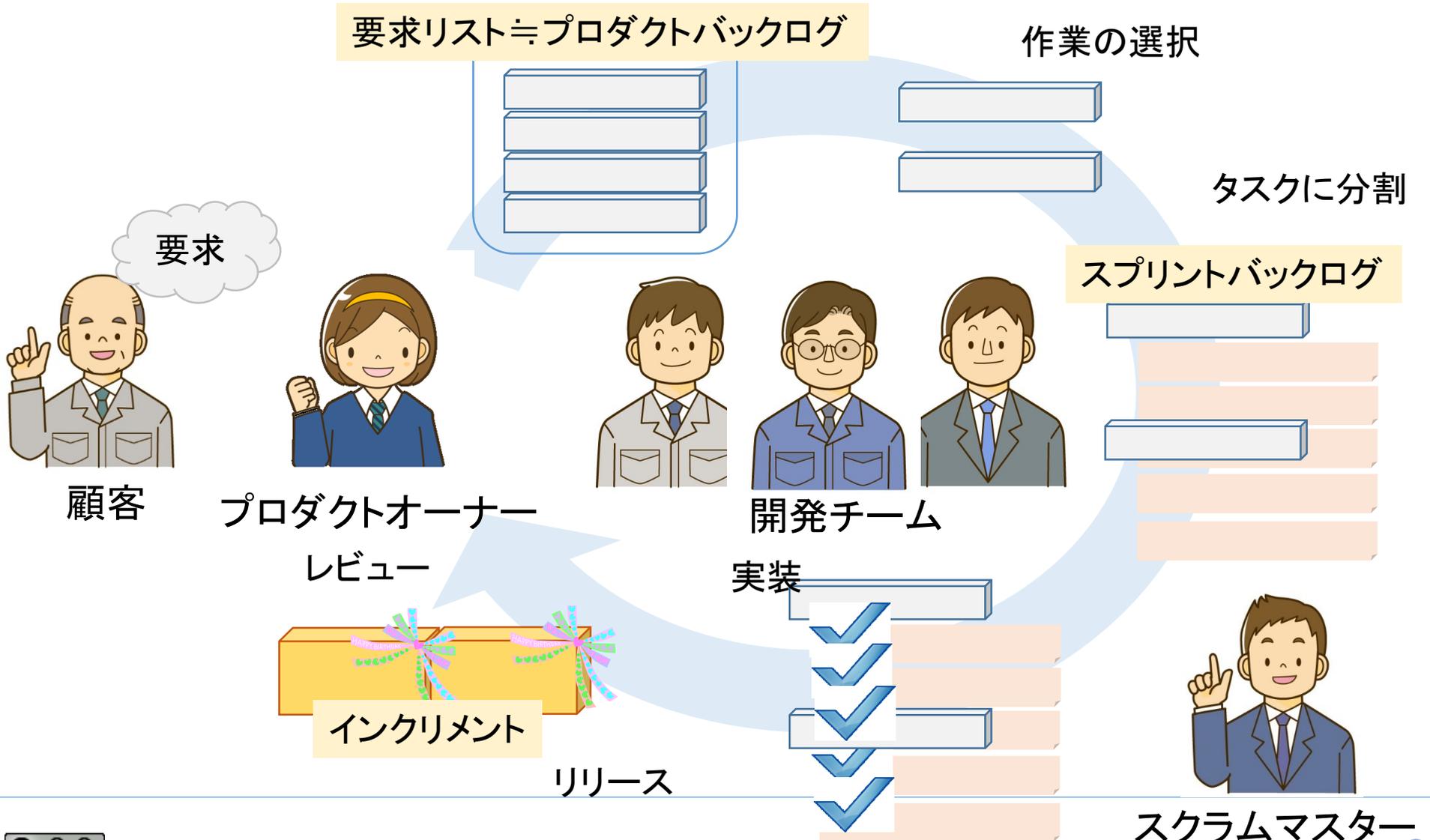
開発チーム



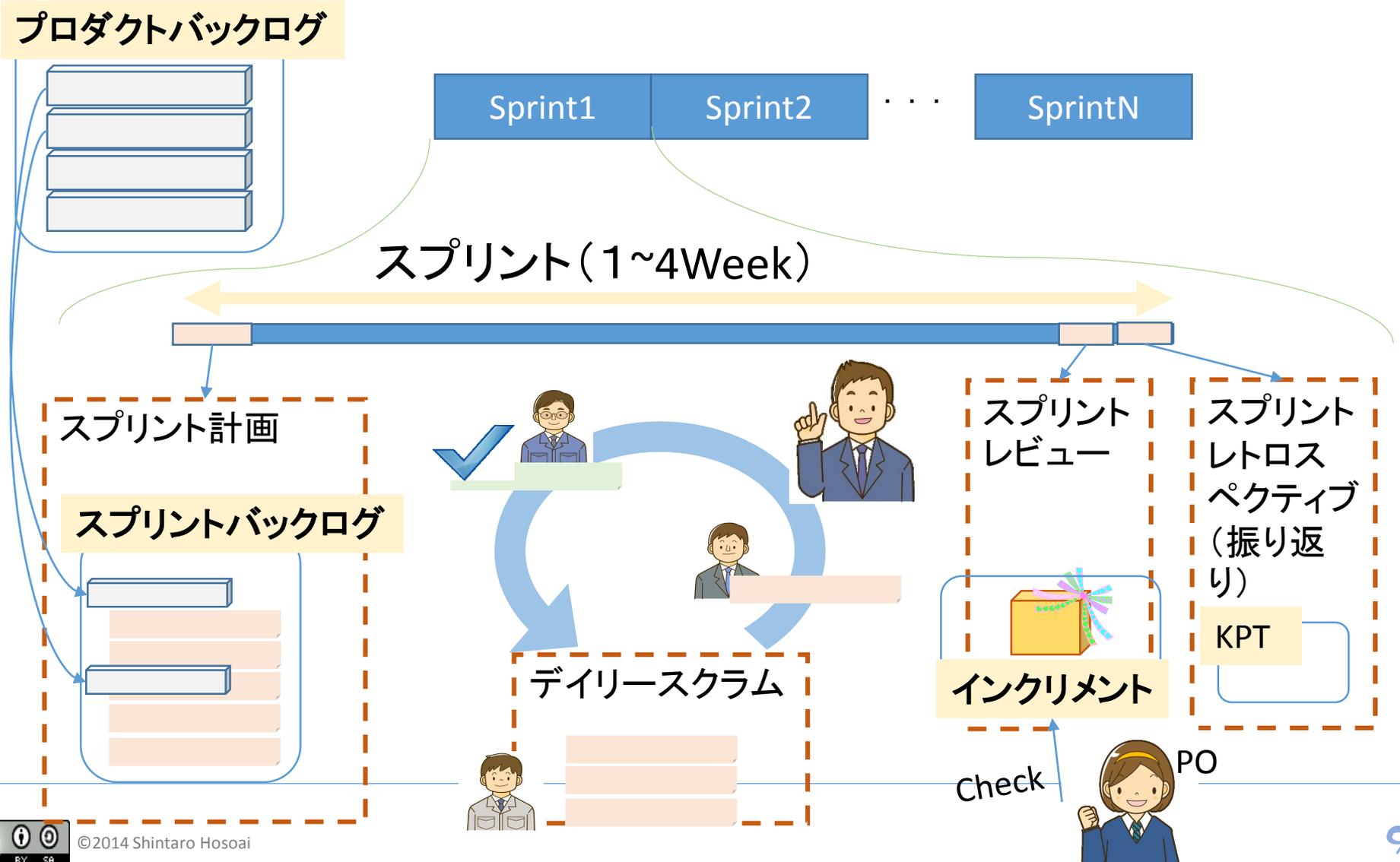
開発の実働隊

常に自己改善, チーム改  
善を念頭におく

# Scrumフレームワーク (1)



# Scrumフレームワーク (2)



# Scrumフレームワーク：まとめ

## ルール

- プロダクトオーナー
- スクラムマスター
- 開発チーム

## 成果物

- プロダクトバックログ
- スプリントバックログ
- インクリメント

## イベント

- スプリント計画ミーティング
- デイリースクラム
- スプリントレビュー
- 振り返り

# スプリント

- 「完了」した，動作するプロダクトインクリメントを作るための期間
  - 反復の単位
  - 一ヶ月以内
    - 長すぎると期間中の定義変更などが起こりうる
    - 失敗のリスクも1スプリントに留める
- 開発チームの編成やスプリントゴールに影響する変更を加えない
- 中止の権限はプロダクトオーナーが持つ

# タイムボックス

- Scrumにおけるあらゆるイベントは固定的（上限のある）な時間枠（タイムボックス）を持つ
  - 一度決定したタイムボックスは変更しない
    - スプリント（2Week）
    - スプリント計画ミーティング（1時間）
    - デイリースクラム（一人5分）
    - スプリントレビュー（1時間）
    - 振り返り（30分）
- 括弧内の時間は例。チームに応じて固定する。

# スプリント計画ミーティング

- 何をこのスプリントで開発するか（第一部）
  - プロダクトバックログ
  
- どのように開発するか（第二部）
  - スプリントバックログ

# プロダクトバックログ（1）

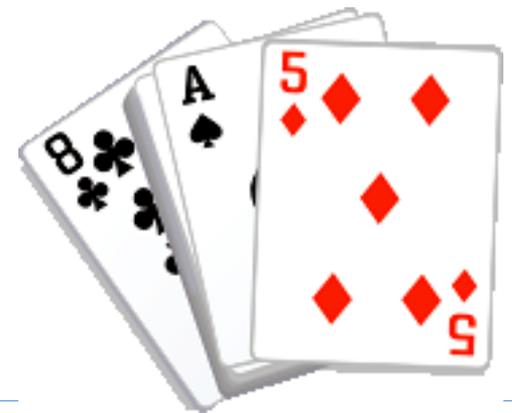
- プロダクトに必要なものが全順序付きで一覧になったもの
  - プロダクトの機能，顧客要求，修正等を含む
  - 顧客に対して価値を与えるもの
- 個々の要素は合意のとれた相対見積りポイントを持つ
  - → プランニングポーカー参照
  - 1スプリントで開発可能なプロダクトバックログ項目の相対見積り  
ポイント数 = ベロシティ

# 相対見積もりとプランニングポーカー

- 初回から正確な見積もりなど出来ない

- プランニングポーカー

- 各自が1,2,3,5,8,13といったカードを持ち、提示されたプロダクトバックログ項目やタスクに対してカードを出す。
- 数字が揃えば適用、合わない場合は一番大きい/小さい人が話し、再度全員でカードを出す。合わない場合は、予め決めたルール（一番大きい数字、平均など）を適用



# スプリントバックログ

- 選択されたプロダクトバックログの項目を「完了」するために必要な作業（タスク）群
  - スプリントゴール達成に必要なタスクがスプリントバックログとして見える化される
  - 見積りは時間単位で小さく見積もる(大きすぎる場合は分割)

# DoD : Definition of Done 完了の定義

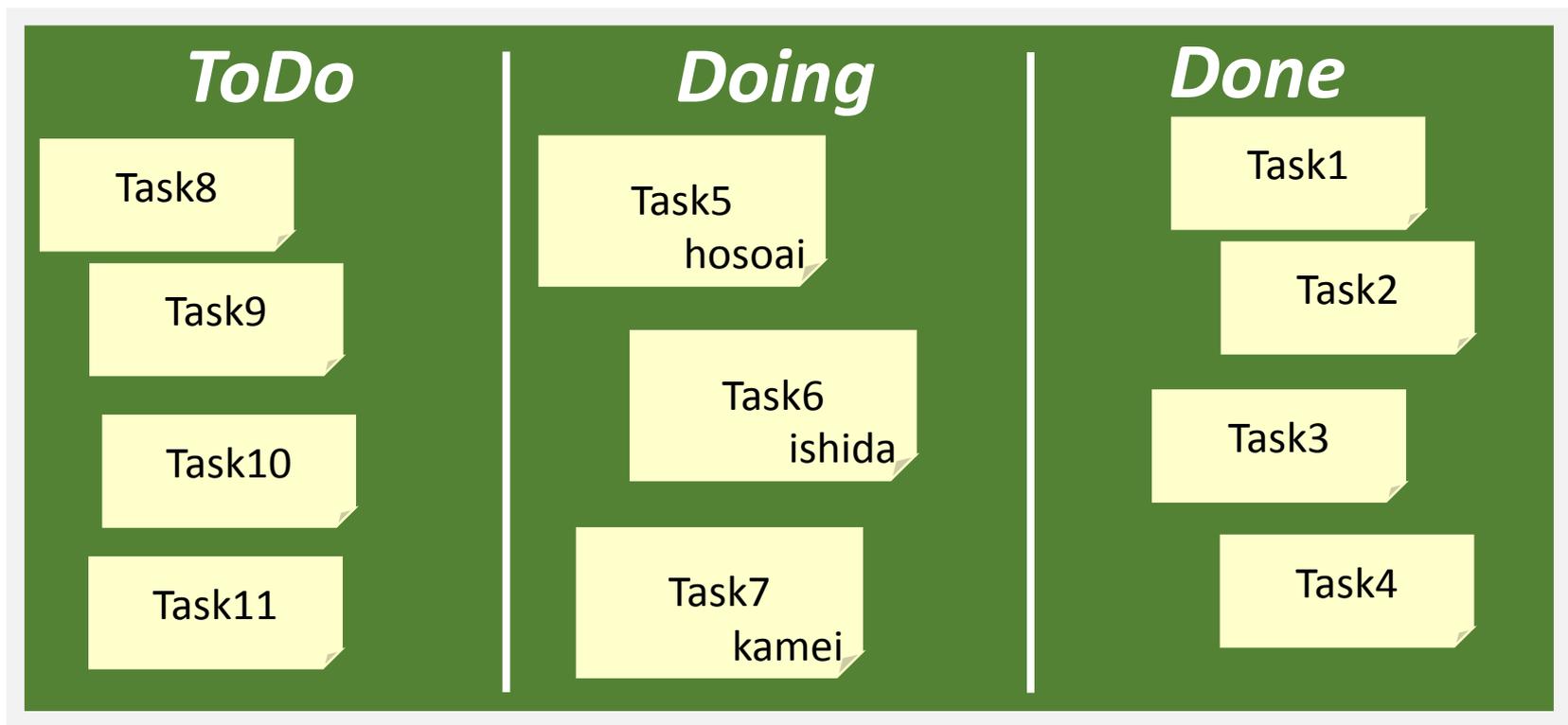
- 全員がスプリントバックログにおけるタスクの“完了”に共通認識をもたなければならない
  - 何をすればそのタスクが“完了”なのか
- Doneの定義は必須
  - “完了”に関する意見統一が取られていなければ、チーム開発は不可能
- 例えば
  - コンパイル可能
  - 指定したテストをすべて通過
  - リポジトリにコミット

# デイリースクラム

- 開発チームが状況を確認し、次の24時間の計画を作る
- スタンドアップで極短時間で終わらせる
- 全員が3つの質問に答える
  - 前回のデイリースクラムから行ったこと
  - 次のデイリースクラムまでに行うこと
  - 問題点
    - デイリースクラムでは問題の洗い出しのみ。対策の議論等は別途行う

# タスクボード (かんばん)

- 何が残っていて, 何が終わったか
- 誰が今何をしているか



# チケットシステム

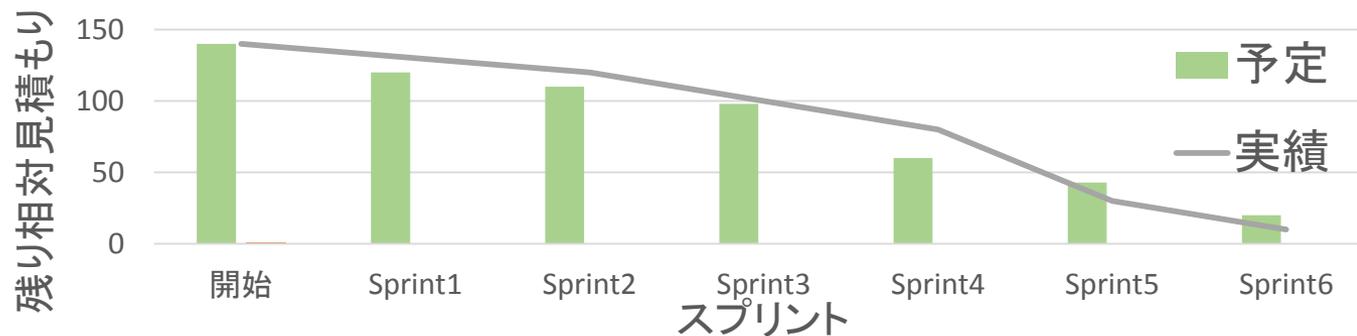
- スプリントバックログのタスクをデータとして入力し, どのタスクが残っており, 何が終わったのかを可視化しやすくする
- 成果物 (ソースコード) とタスクの紐づけ

# 開発

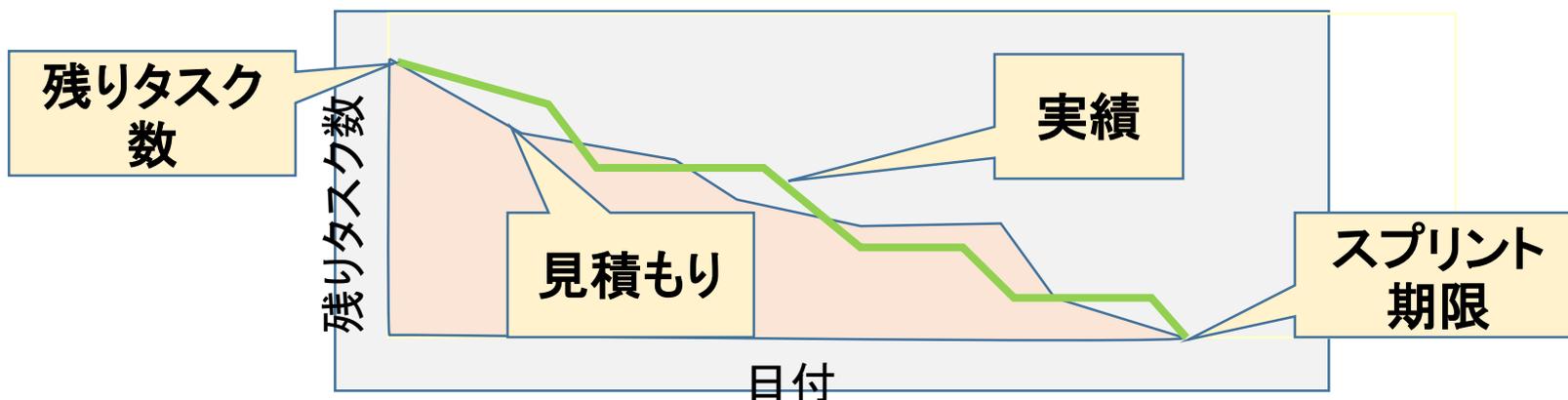
- インクリメントの開発“完了”を目指す
  - スプリントで開発が完了した“動作する”ソフトウェア（紙芝居ではダメ）
- 進捗管理，タスク割り当てはScrum Masterの支援のもと全メンバーが自己組織的に行う
  - 利用するツール，イベント
    - デイリースクラム
    - タスクボード
    - チケットシステム
    - スプリントバーンダウンチャート

# バーンダウンチャート

## • リリースバーンダウンチャート



## • スプリントバーンダウンチャート

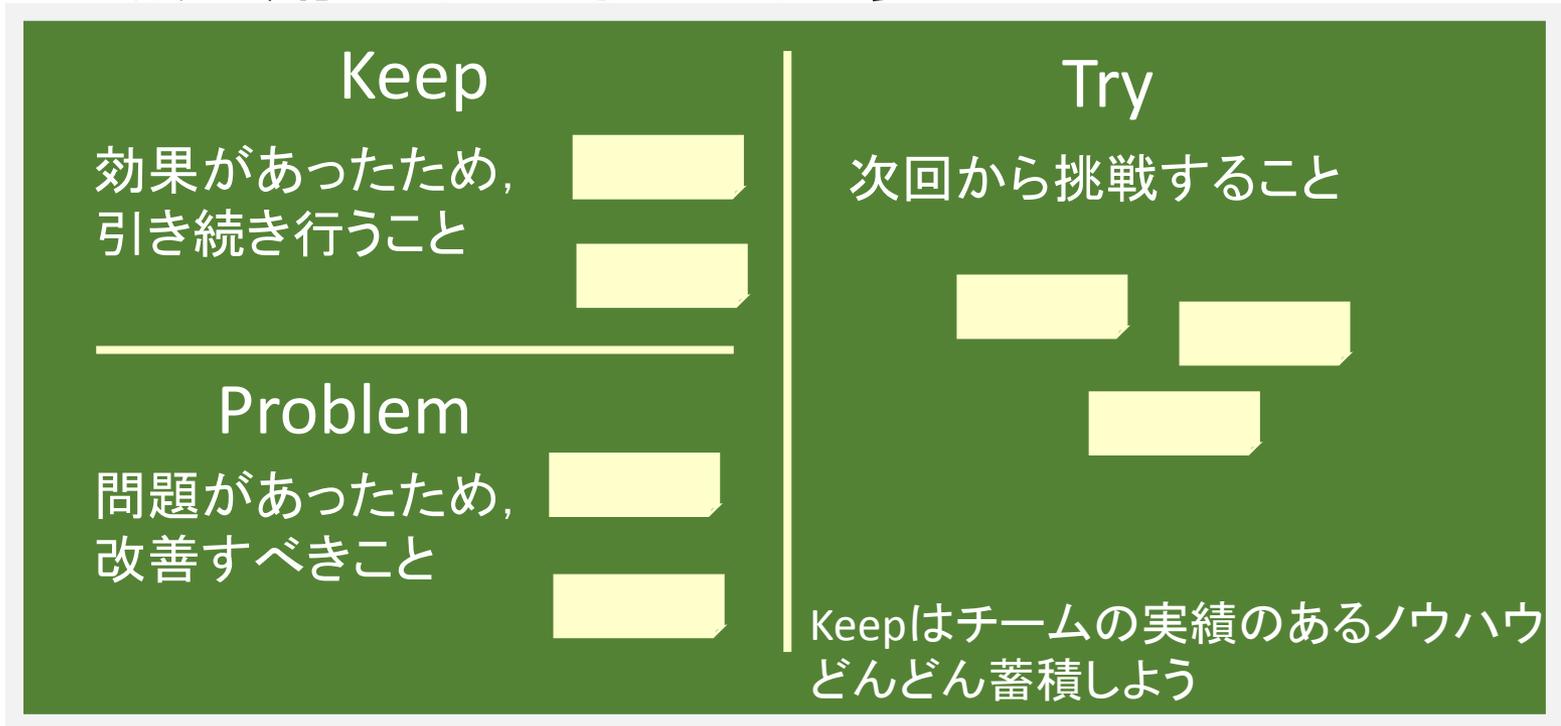


# スプリントレビュー

- プロダクトオーナーは何が“完了”して何が完了していないのかを特定する
- 開発チームはインクリメントのデモを行う
- デモを通じてフィードバックを引き出し，必要に応じてプロダクトバックログを修正する

# 振り返り

- 人・関係・プロセス・ツールの観点からスプリントを検査し，改善実施計画を作成する
- KPT法が用いられることが多い



# 振り返りの注意

- Keepは資産なので大事にする
  - 自分のプロジェクトにおける守るべきルール群になる
- 個人攻撃の場ではない
  - タスクをすすめるスピードは人によって異なって当たり前
  - 個人差を考慮した見積りや計画を考慮できないか考える
- サポート体制を含めたチームでの課題/改善案を考える
  - A君の実装スピードが遅いのでがんばれ, などは改善案ではない

# まとめ

