

1.3 アジャイルでの プロジェクト管理

1.3.1 計画づくりの難しさ

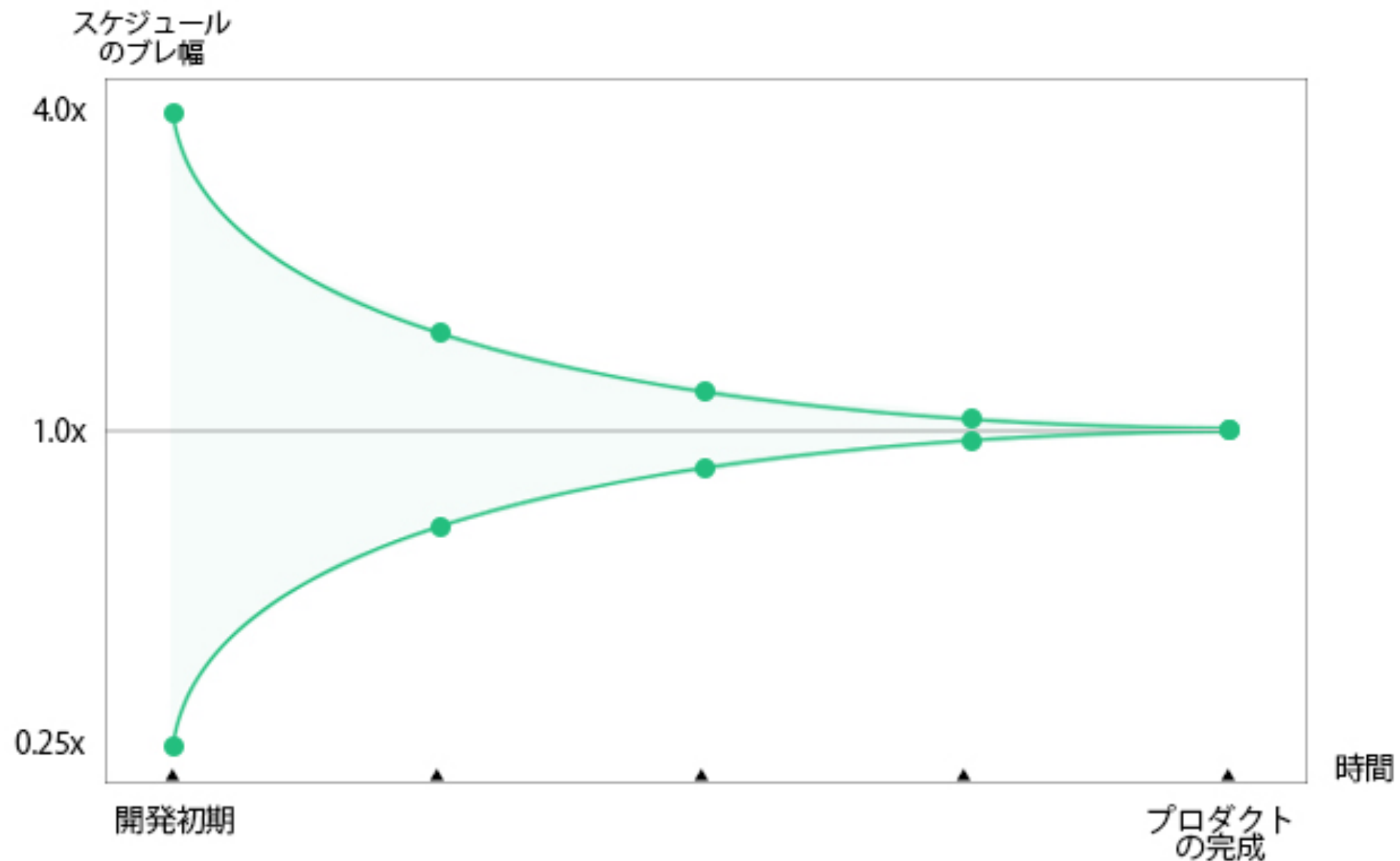
プロダクトを取り巻く状況によって
プロダクトへの要件は常に変化おり、
開発当初に全ての事象を予測することは非常に困難

計画を狂わせる要因

- ・途中で要件や仕様が変わる
- ・開発開始後に新たな課題が見つかる
- ・より良い実装手段が見つかった
- ・メンバーのスキルアップ
- ・など

1.3.2 不確実性コーン

開発当初に立てた計画は誤差が大きく、
開発が進むにつれて精度が高まっていく



1.3.3 アジャイルによる計画づくり

状況の変化に対応するためのプラクティス

- ・ プロダクトバックログ
- ・ スプリントバックログ
- ・ ストーリーポイントによる見積もり
- ・ カンバン
- ・ バーンダウンチャート
- ・ ベロシティ計測

1.3.4 プロダクトバックログ

プロダクトに必要な機能・要件・要望・修正事項をリスト化したもの

- ・顧客に提供できる価値の大きさやコストなどを勘案して一列に順序づけ
- ・ステークホルダーと調整して決定する
- ・開発チームは順位の高いものから着手する

① やってみよう

ボウリングのスコア計算プログラムの
プロダクトバックログを作成

1. プログラムに必要な要件を洗い出す
 - この段階では優先度は考えない
 - 必要だと思われるものは全て書きだす
2. 要件に順位をつけ 1 列に並べる
 - 早く実現したいものや重要なものは上に
 - 余力があれば実現するというものは下に

1.3.5 スプリントバックログ

プロダクトバックログのうち

今回のスプリントで開発するストーリーを取り出し

そのストーリーを完成させるために必要な

すべての作業項目を洗い出したもの

- ・ 作業量は大きすぎず小さすぎず
- ・ 第三者が完了したか判断できる粒度

ストーリーポイントによる相対見積り



Q1. Aのビルは何メートルですか？

→絶対的な値による見積りは外れやすく時間もかかる

Q2. AのビルはBのビルの何倍ですか？

→相対的な見積りは簡単で正確

1.3.7 プランニングポーカー

0, 1, 2, 3, 5, 8, 13, …の数字が書かれたカードを使用して素早く見積もる



- ・時間をかけすぎない
- ・みんなで見積もる
- ・認識を合わせる
- ・大きな数字の
小さな差は気にしない













① やってみよう

プロダクトバックログを見積もる

1. 基準となるプロダクトバックログを決める
 - ・ 適当なバックログを選び2ptまたは3ptとする
2. 他のプロダクトバックログを見積もる
 - ・ 見積もるプロダクトバックログを1つ選ぶ
 - ・ 基準となるプロダクトバックログと比べて何倍ぐらいの作業量になるかを見積り、その数値が書かれたカードを選ぶ
 - ・ 全員が選び終わったら一斉にカードを表に向ける
 - ・ 数値が一致している場合は見積り終了
 - ・ 数値が違っていたら理由を聞いてから再度見積もる。再見積りは2回までで終わる。

1.3.8 カンバン（タスクボード）

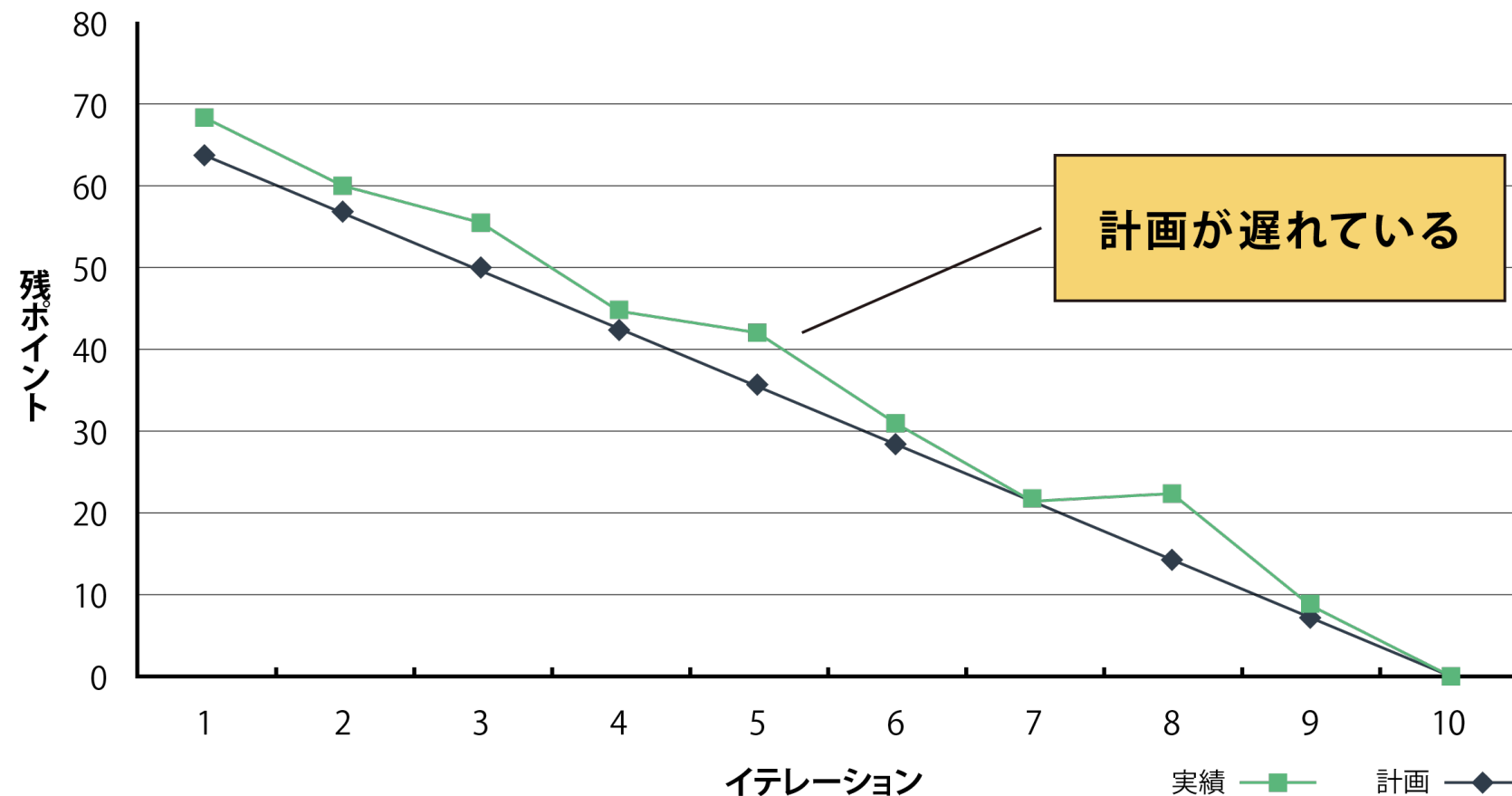
バックログを付箋などに書き出して、
作業を見える化するためのツール

ToDo	Doing	Done
    	  	   

カードには、期日・残作業時間・作業責任者・進捗インジケータなどを補足情報として書く場合もある

1.3.9 バーンダウンチャート

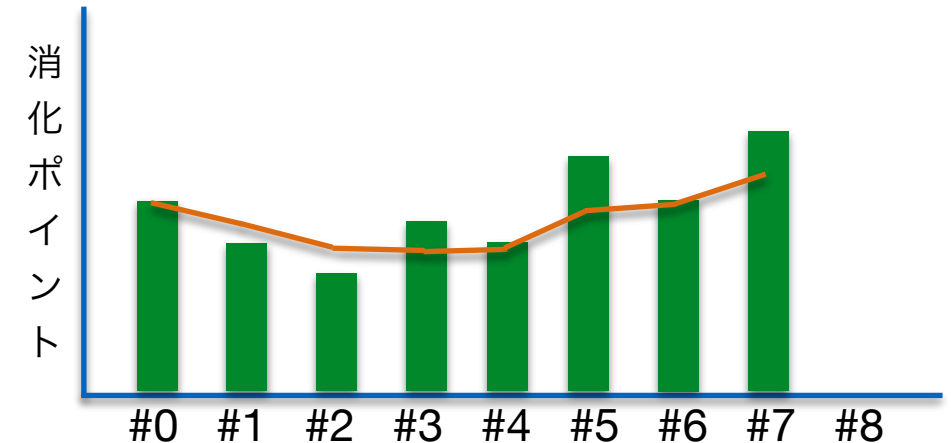
ポイントや残作業時間、残タスク数を使って
バーンダウンチャートを描くことで
プロジェクトの進捗を見える化する



計画と実績が合わない場合は、プロジェクトの期間か
スコープ（実現するバックログ）を調整する

1.3.10 ベロシティ計測

スプリントで完成したバックログのポイントの合計をベロシティと呼びます。



ベロシティを計測することで、
残りのバックログを消化するために
あと何スプリント必要かがわかります。

スプリントのベロシティにはバラつきがあるため
通常は数回分のスプリントの平均ベロシティを
使います。