先送りポイント可視化がアジャイルチームに与える行動変容について

一般財団法人日本科学技術連盟

第36年度（2020年度）ソフトウェア品質管理研究会
成果発表会

研究コース4 アジャイルと品質チーム

2021年2月26日(金)

本日の発表内容

- 背景
- 提案
- 実験概要・手順
- 実験結果
- 考察
- まとめ
- 研究チームご紹介

アジャイル開発とメトリクス

プロジェクト原則

- プロジェクト進捗・品質状況の可視化は、ソフトウェア開発では不可欠

アジャイル開発では

- スプリントを繰り返すことにより継続的なプロジェクト改善が可能
- ウォーターフォール開発で使用するメトリクスを適用することが多い
- メトリクスは最終評価者の工程移行や出荷判断に活用することが多い

開発チームの継続的な改善を支援できるかという観点については未検討

＜NECアジャイルにて取得するデータ項目＞

	種別	メトリクス	単位
1	工数	設計・実装	人H
2		テスト	人H
3	ストーリー	完了	件
4	ベロシティ	全体ストーリーポイント	SP
5	バグ数	全体バグ数	件
6		うちデグレード	件
7		うち重大	件
8	コードメトリクス	カバレッジ	%

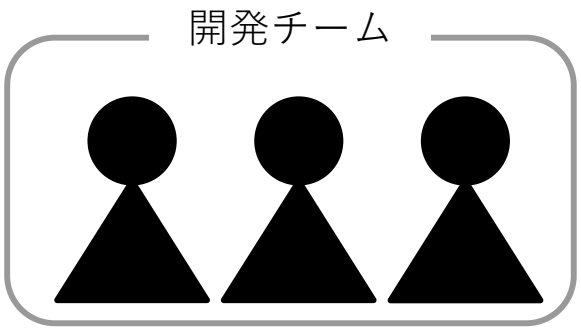
引用抜粋：アジャイルと品質会計－プロジェクトの高成功率を確保するハイブリッドアジャイルへの取り組み－，情報処理学会デジタルプラクティス Vol．7 No．3，2016.

本研究での課題

アジャイル開発では、短時間で動くものを作ることに集中するため、
メトリクス計測を負荷に感じやすく、データ取得に後ろ向きな傾向になりやすい

メトリクスのデータ取得を開発チームへ依頼すると・・・

・データ取得するのに手間がかかる
・データ取得よりもバグ修正を行いたい



本研究では・・・

以下の条件のもと、**開発チームの改善活動を促進するメトリクスを検討**

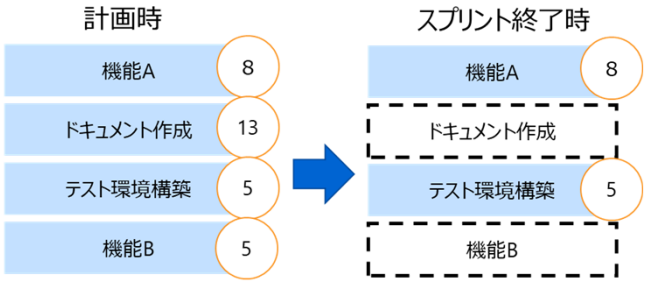
- ■アジャイル開発の短い単位で、データを収集するという特徴を活かす
- ■収集が容易で、開発メンバーが負担に感じない
- ■アジャイル開発のスピード感を損なわない

提案1:先送りポイント

開発チームの改善活動を促進するためのメトリクス、「**先送りポイント**」を提案
【先送りポイント ＝ 計画時に立てた見積もりポイント － 完了したポイント】

[特徴1]

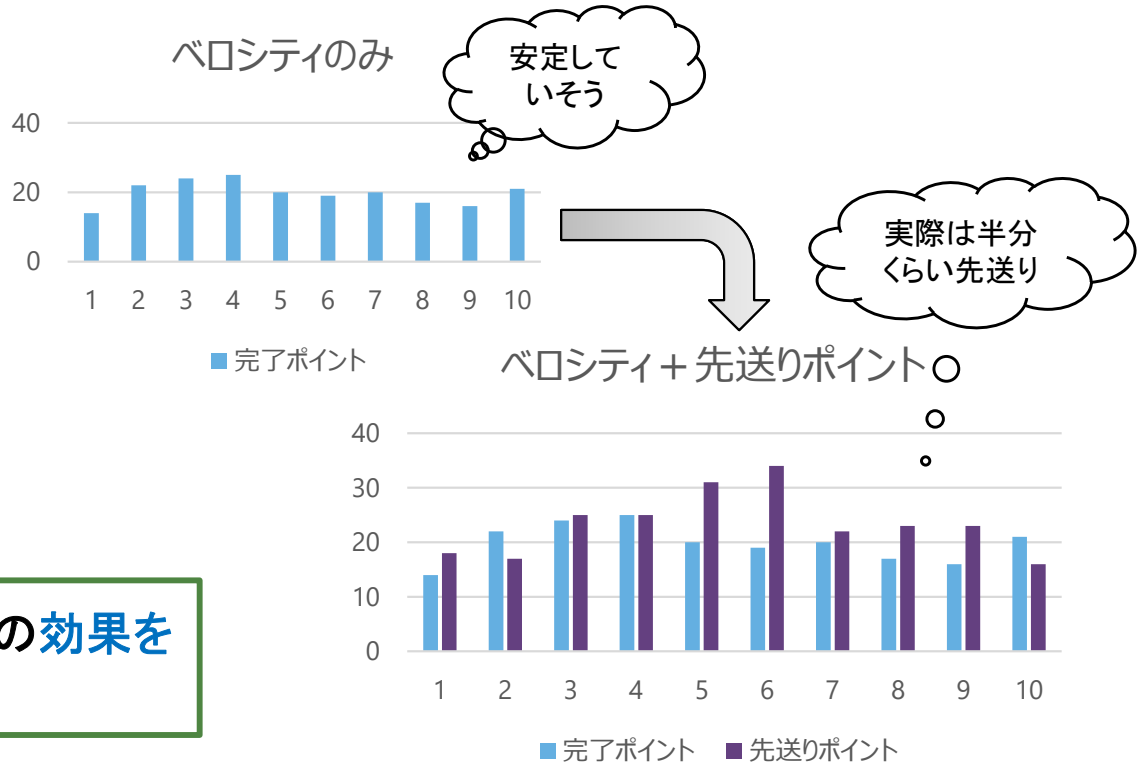
単純な差分で計測でき、導入コストが低い



	計画ポイント	完了したポイント	先送りポイント
スプリント1	31	13	18

[特徴2]

「完了しなかった」に着目することで、改善を誘発しやすい



[特徴3]

スプリントごとにトレンドが分かるため、改善の効果を確認しやすい

提案2：想定される原因と対策例

研究会メンバが各自の経験を持ち寄りブレインストーミング

- **想定される原因と対策の仮説を立てた**

想定される原因		対策例
割り込みが多い	品質が悪いため、バグ対応が発生	・割り込みの分析をして制御可能なものに対して対策する ・計画時に割り込みを予測し、バッファとして積む
	優先度のより高い業務が発生	
	大きな予定変更が発生	
	兼務をしているため、別PJの業務が圧迫	
過剰な計画をする	チームのベロシティを過信	・ベロシティを踏まえて計画をする ・既知の予定（個別のものも含む）を踏まえて計画をする
	POが無理な要求をする	
過少見積もりをする	ストーリーの受入条件が不明確	・ストーリーの詳細をPOに確認する（不明なままスタートしない） ・受け入れ基準／Doneの定義を明確にする ・リファインメント／計画会議を見直す
	計画を詳細に立てられるほど、ストーリーが詳細化されていない	
メンバーの負荷が偏る	業務の属人性が高い	・技術／知識共有の施策の検討をする（ペアプロ、モブプロ）
	スキルが偏っている	
待ちが多い	回答待ちなど、待ちとなる作業が多い	・他部署との調整が必要なものを明確にする

開発チームに対策のアイデアを発想するためのきっかけとしてもらう

実験概要

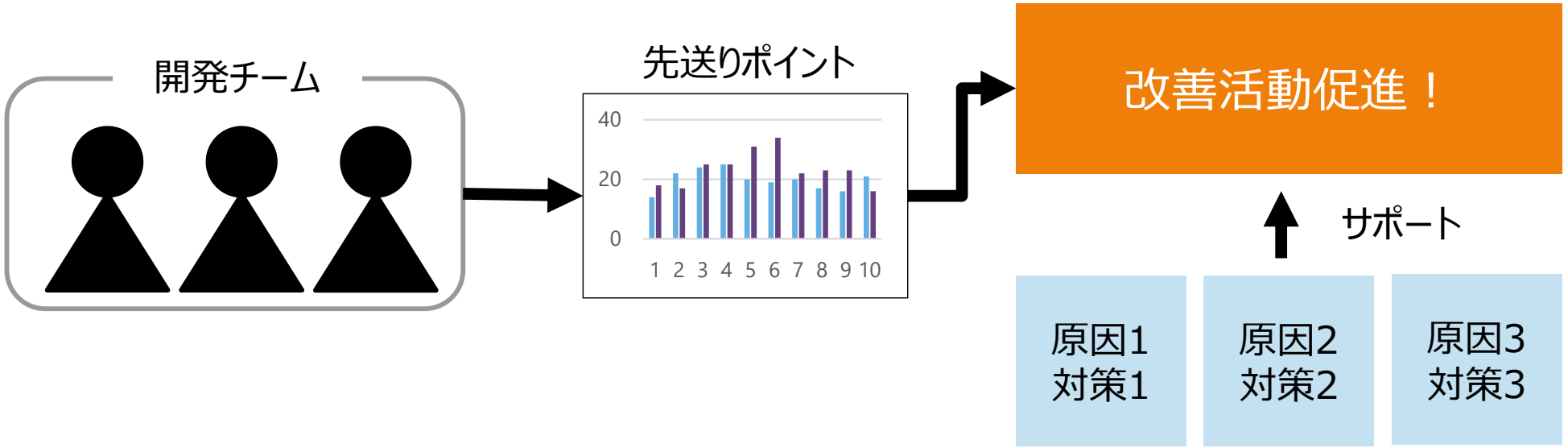
- **実験の目的**

「先送りポイント」がアジャイル開発チームに対して以下の効果があるメトリクスであることを確認する

（１）簡単に収集でき、継続的に取得できること

（２）開発チームの改善活動を促進すること

（３）原因と対策の例が実際の開発現場に即していること



実験対象プロジェクト

項目	説明	
	プロジェクトA	プロジェクトB
プロジェクトの説明	自社開発しているソフトウェア製品にて**自動テストを作成**するプロジェクト	主に**AI技術**を用いて製品の付加価値を提供するプロジェクト
人員構成	開発メンバ：3人 プロジェクトオーナー：1名 スクラムマスター：持ち回り制 スプリント期間：2週間	開発メンバ：4人 プロジェクトオーナー：1名 スクラムマスター：持ち回り制 スプリント期間：2週間
特長 (困りごと)	・自発的に改善に取り組んでいる 原因分析やその対策案には自信がない。 ・ベロシティとその推移グラフは作成しているが**改善活動には繋げられていない。** ・完了しなかったポイント、途中増減のポイントを測定しているが、グラフ化して**傾向分析はしていない。**	・完了ポイントはベロシティ測定のため計測しているが、グラフ化はしていない。 ・レトロスペクティブにて先送りになるポイントがあることを気にしていたが**原因分析や対策を具体的に打てていない。**

実験手順

実験手順

①　完了ポイントと先送りポイントの推移グラフを作成

- 対象プロジェクトへその結果を報告
- 継続的に先送りポイントを取得するかを観察

②　改善活動に繋がる事例があるかを確認

- 継続的にレトロスペクティブに参加し、改善活動の取り組みがあるか観察

③　プロジェクトの原因分析結果と仮説の原因・対策案を比較

- 各プロジェクトメンバと、先送りになったストーリーの原因分析を実施
- ただし、研究会で立てた仮説については提示しない
- 実際のプロジェクトでの原因分析結果と研究会での原因パターンを比較
  →原因パターンの妥当性を確認する

グラフの作成などは積極的なサポートを行ったが、先送りポイントの活用法については直接的な誘導を行わないよう心掛けた

実験結果

実験の目的である３点の確認ポイントに対し、以下の確認ができた

ポイント1

（１）簡単に収集でき、継続的に取得したか？

➡ **2つのプロジェクトとも、継続的に「先送りポイント」を収集し始めた**

※自動的に集計できるよう改良加えることも始めた

ポイント2

（２）開発チームの改善活動を促進したか？

➡ **先送りの原因を探るために別のメトリクスを収集した**

プロジェクトA: 見積ブレ/割りこみの2種類のメトリクスに分けて記録

プロジェクトB: 割り込みを計測始めた

完了しなかったストーリーについて予実差分を取り始めた

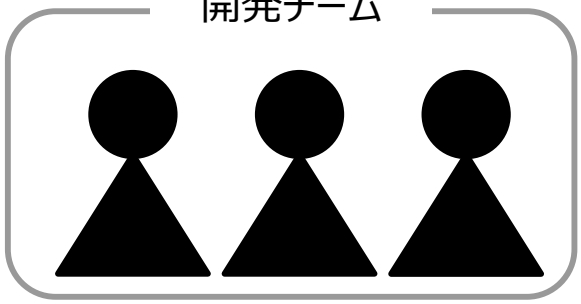
ポイント3

（３）原因と対策の例が実際の開発現場に即しているか？

➡ **5つの原因パターンを準備したが、そのうち4つに該当した**

結果①「先送りポイント」は計測簡単！

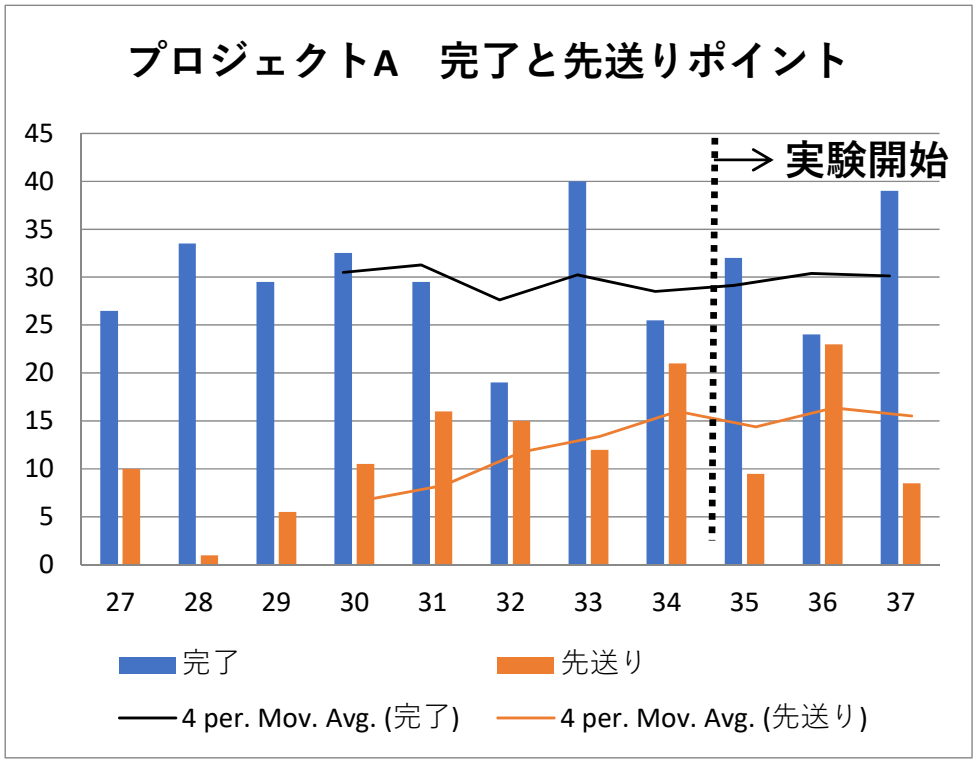
2つのプロジェクトとも、継続的に「先送りポイント」を収集し始めた

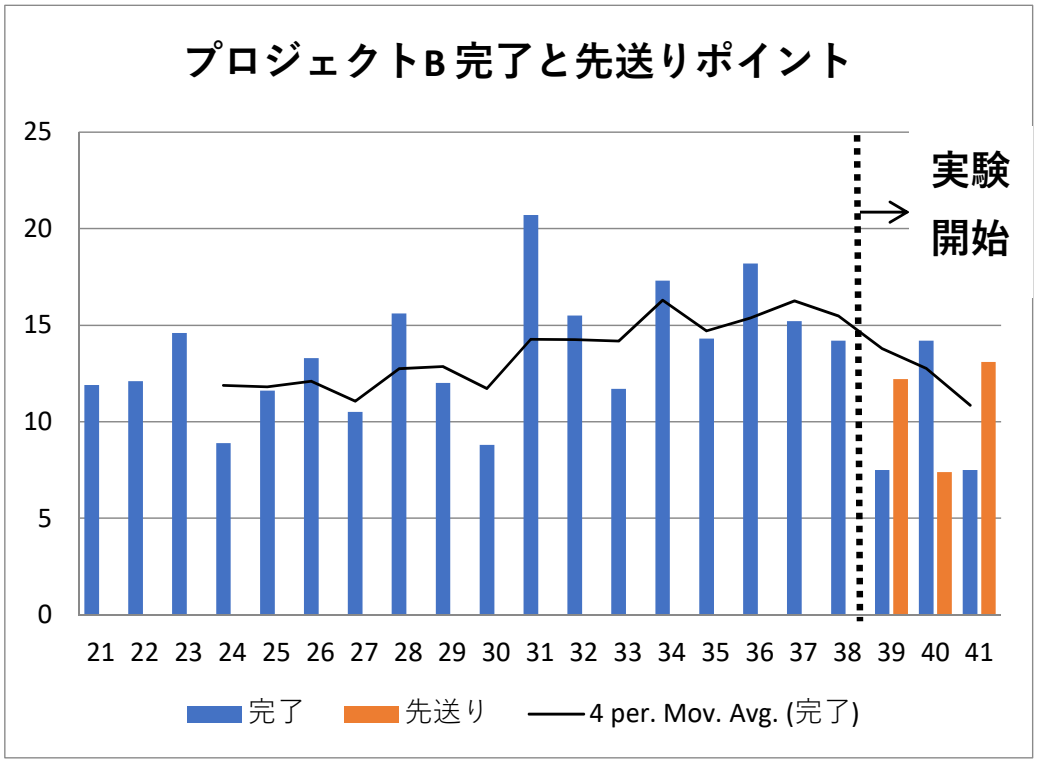


こんなグラフになるなんて！

新しくデータとる必要ないよね！

自動的にとれるよね！





結果②「別のメトリクス」を探し始める！

プロジェクトA 途中増減の詳細を計測し始めた

実験開始前

Sprint No.	期間	計画	途中増減	完了
27	7/3-7/17	32	4.5	26.5
28	7/17-8/4	24.5	11	33.5
29	8/4-8/24	30	5	29.5
30	8/24-9/7	29.5	12	32.5
31	9/7-9/18	29	16.5	29.5
32	9/18-10/2	24.5	9.5	19
33	10/2-10/1	25.5	26.5	40
34	10/16-10/	34.5	12	25.5
35	10/29-11/	37	4.5	32

実験開始後

Sprint No.	期間	計画	途中増減		締め	
			見積ブレ	割り込み	完了	先送り
27	7/3-7/17	32	4.5	-	26.5	10
28	7/17-8/4	24.5	11	-	33.5	1
29	8/4-8/24	30	5	-	29.5	5.5
30	8/24-9/7	29.5	12	-	32.5	10.5
31	9/7-9/18	29	16.5	-	29.5	16
32	9/18-10/2	24.5	9.5	-	19	15
33	10/2-10/16	25.5	26.5	-	40	12
34	10/16-10/29	34.5	12	-	25.5	21
35	10/29-11/12	37	4.5	-	32	9.5
36	11/12-11/27	34	13	-	24	23
37	11/27-12/10	40	5	2.5	39	8.5

プロジェクトB 割込み/タスクの予実差分を取り始めた

Sprint No.	期間	完了	割込	先送り
34	8/7 - 8/28	17.3		
35	8/31 - 9/14	14.3		
36	9/15 - 10/1	18.2		
37	10/2 - 10/16	15.2		
38	10/19 - 11/2	14.2		
39	11/4 - 11/18	7.5		12.2
40	11/19 - 12/4	14.2	3.2	7.4
41	12/7 - 12/21	8.5	0.5	12.1

←実験開始　先送りポイント取得

←割込にフラグ追加

Story	実績工数(時間)	初期見積(時間)	予実差分
Story 001	21.25	9	12.25
Story 002	4.5	6	1.5
Story 003	4.4	33	28.6
Story 004	2.5	8	5.5

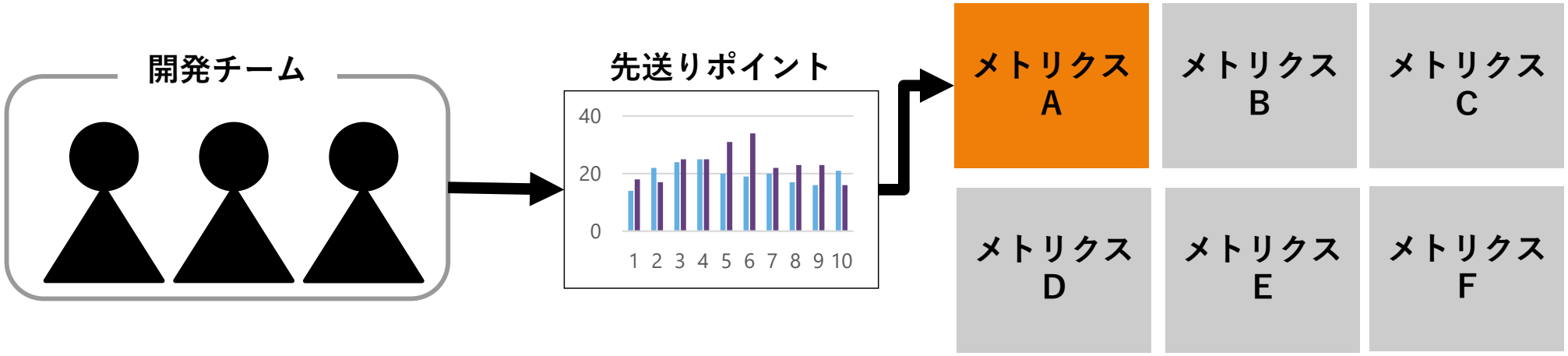
結果③ 原因パターンの妥当性を確認！

5つの原因パターンを準備したが，そのうち4つに該当した

想定される原因		対策例
割り込みが多い	品質が悪いため、バグ対応が発生	・割り込みの分析をして制御可能なものに対して対策する ・計画時に割り込みを予測し、バッファとして積む
	優先度のより高い業務が発生	
	大きな予定変更が発生	
	兼務をしているため、別PJの業務が圧迫	
過剰な計画をする	チームのベロシティを過信	・ベロシティを踏まえて計画をする ・既知の予定（個別のものも含む）を踏まえて計画をする
	POが無理な要求をする	
過少見積もりをする	ストーリーの受入条件が不明確	・ストーリーの詳細をPOに確認する（不明なままスタートしない） ・受け入れ基準 Doneの定義を明確にする ・リファインメント／計画会議を見直す
	計画を詳細に立てられるほど、ストーリーが詳細化されていない	
メンバーの負荷が偏る	業務の属人性が高い	・技術／知識共有の施策の検討をする（ペアプロ、モブプロ）
	スキルが偏っている	
待ちが多い	回答待ちなど、待ちとなる作業が多い	・他部署との調整が必要なものを明確にする

考察

先送りポイントは、適切なメトリクスを見つけるきっかけ



考察

- **先送りの改善を切り口に、適切なメトリクスを考えるきっかけを与える**
- **ただし、先送りポイントの効果が低い開発チームが存在する**
  - **例. 先送りの発生があらかじめ想定されていた場合**

先送りポイントのさらなる活用に向けて

- 先送りポイントの最適な可視化の方法を検討する
  - 先送り原因ごとの円グラフなど複数の候補あり
- 先送りポイントとソフトウェア品質との関係を明らかにする

まとめ

「**先送りポイント**」の導入により、開発チームの自発的な改善活動を促進

【先送りポイント ＝ 計画時に立てた見積もりポイント － 完了したポイント】

[特徴1]

単純な差分で計測でき、導入コストが低い

[特徴2]

「完了しなかった」に着目することで、改善を誘発しやすい

[特徴3]

スプリントごとにトレンドが分かるため、改善の効果を確認しやすい

「**先送りの原因と対策例**」で対策の検討をサポート可能

想定される原因		対策例
割り込みが多い	品質が悪いため、バグ対応が発生	・割り込みの分析をして制御可能なものに対して対策する ・計画時に割り込みを予測し、バッファとして積む
	優先度のより高い業務が発生	
	大きな予定変更が発生	
	兼務をしているため、別PJの業務が圧迫	
過剰な計画をする	チームのベロシティを過信	・ベロシティを踏まえて計画をする ・既知の予定（個別のものも含む）を踏まえて計画をする
	POが無理な要求をする	
過少見積もりをする	ストーリーの受入条件が不明確	・ストーリーの詳細をPOに確認する（不明なままスタートしない） ・受け入れ基準／Doneの定義を明確にする ・リファインメント／計画会議を見直す
	計画を詳細に立てられるほど、ストーリーが詳細化されていない	
メンバーの負荷が偏る	業務の属人性が高い	・技術／知識共有の施策の検討をする（ペアプロ、モブプロ）
	スキルが偏っている	
待ちが多い	回答待ちなど、待ちとなる作業が多い	・他部署との調整が必要なものを明確にする

アジャイルと品質 チームメンバご紹介

<研究員>

星野 友基（ビー・ユー・ジーDMG森精機株式会社）

中野 雄（キヤノン株式会社）

夏目 珠規子（株式会社東芝）

松崎 紀子（TIS株式会社）

<主査>

永田 敦 (サイボウズ株式会社)

<副主査>

山口 鉄平（freee株式会社/一般社団法人アジャイルチームを支える会）

荻野 恒太郎（楽天株式会社）

<アドバイザー>

細谷 泰夫（三菱電機株式会社）

ご傾聴ありがとうございました。