チームスピリットのアジャイル開発における品質保証
~チームの形成から現在まで~

2019年 8月 30日

株式会社チームスピリット　生井 龍聖

自己紹介

名前：生井 龍聖

所属：株式会社チームスピリット

ScrumMaster / QA Lead / SET

経歴：業界8年目

2015年よりスクラム開発のQAに

2017年にチームスピリットに入社

資格

JSTQB AL TA

認定スクラムマスター（LSM）

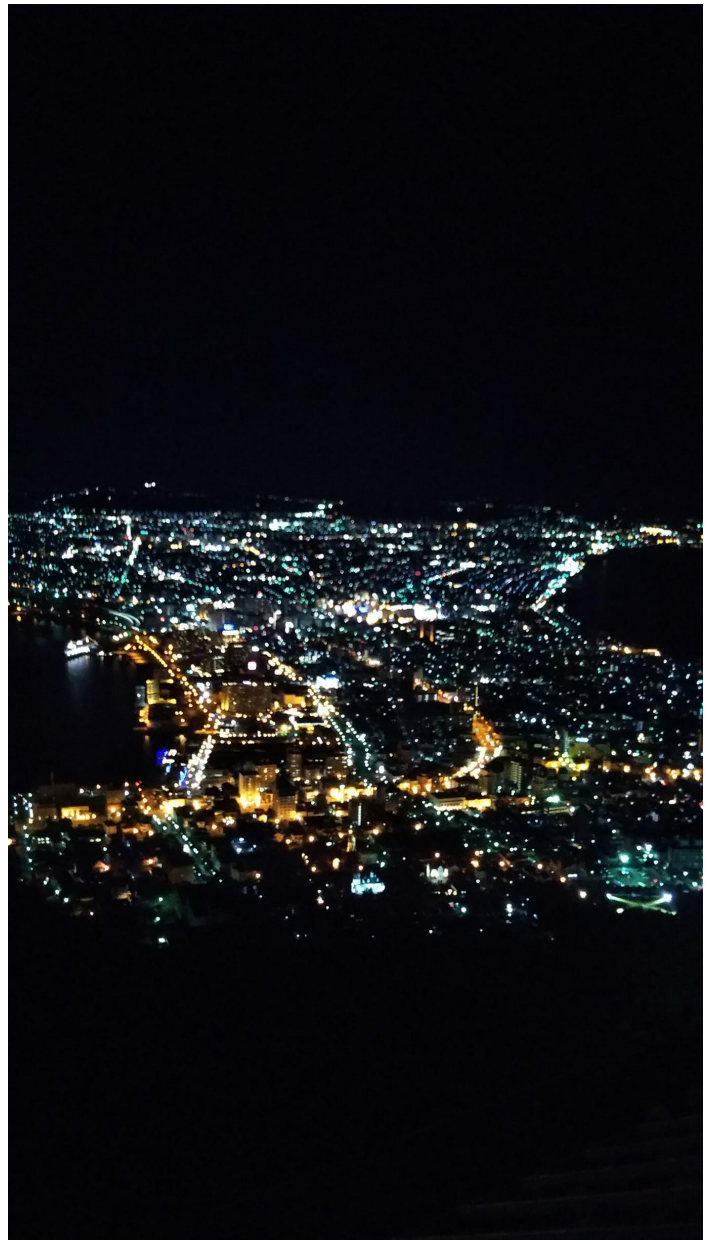
趣味

フットサル・スニーカー

北海道

休暇を取って満喫しています！

月曜日～水曜日：函館
水曜日～土曜日：札幌



目次

はじめに

はじめに

チームスピリットはスクラムでの開発を行っており、
品質の作り込みについてはスクラム一丸となって取り組んでいる。
チームの人数を形成期、統一期と2つの段階に分け、現在に至るまでの取り組みを紹介したい。

解決したい課題

チームの規模・状態によって異なる課題が見られた

形成期	**アジャイル開発における品質保証体制が未整備** 立ち上げ当初のチームはQAエンジニアがおらず、アジャイル開発の経験も少ないメンバーでチームを形成していた。 そのため全体的な品質保証体制の構築が必要だった。
統一期	**人数増加に伴う仕組みづくり** 人数の増加にともない、これまで解決できていた問題が徐々に対応しきれなくなってきた。 個人での品質保証体制からチームでの品質保証体制に移り変わる必要があった。

形成期

～「透明性」を根付かせる～

形成期の課題と分析

■アジャイル開発における品質保証体制が未整備

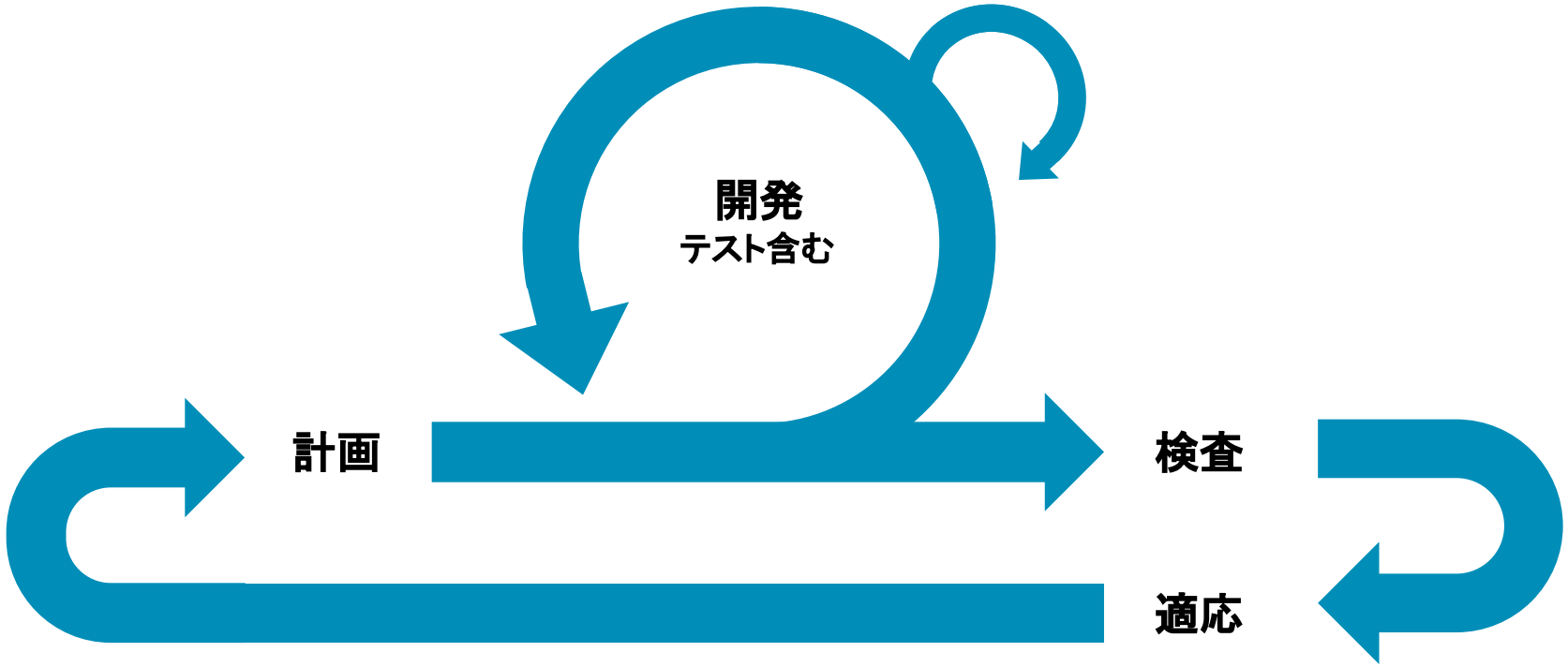
- 少人数の機能別チームに分かれており、十分な情報共有が出来ておらずパフォーマンスの良し悪しが顕著に分かれていた
- 機能ごとのテストはあったものの、機能を跨る品質保証施策が整っていなかった
- プロダクト全体での品質について判断できる状態ではなかった

■分析方法

すべてのスクラムイベントに参加し、情報収集から始めた

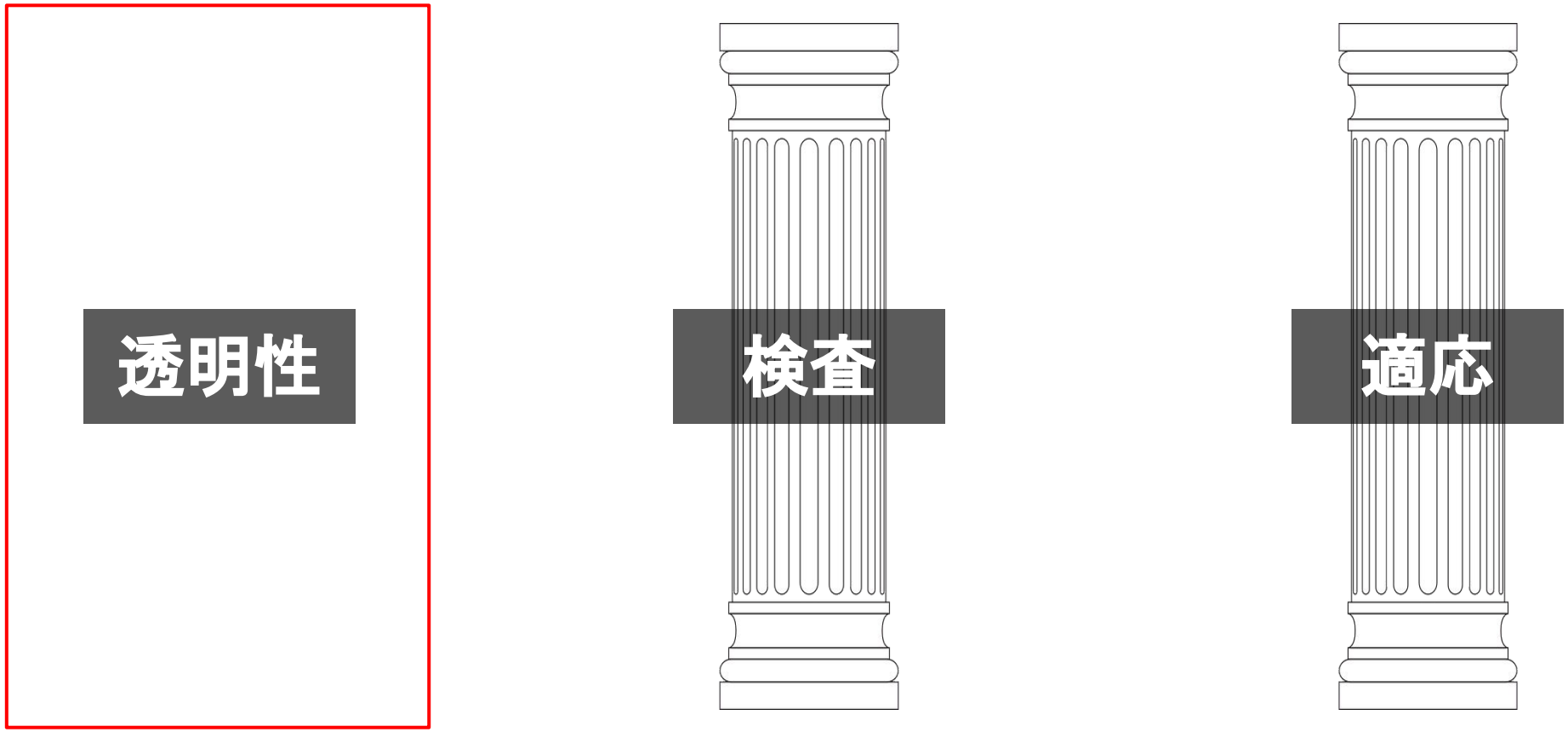
形成期の課題と分析

経験的プロセス制御の実現は、
「透明性」・「検査」・「適応」の 3 本柱に支えられている。



形成期の課題と分析

3つの柱の「透明性」に課題が見えてきた



形成期の施策

QAエンジニア
×
Scrum Master

QAとスクラムマスターの兼務

Scrum Master	**スクラムを根付かせる** ● チームでの経験を大切にする ● 開発チームをコーチする ● チームで品質を意識する
Quality Assurance	**プロダクトの「透明性」を確保する** ● インシデント分析の整備・報告 ● テスト計画 ● テスト実施

形成期の施策

スクラムマスターとしてスクラムを根付かせる

チームでの経験を大切にする

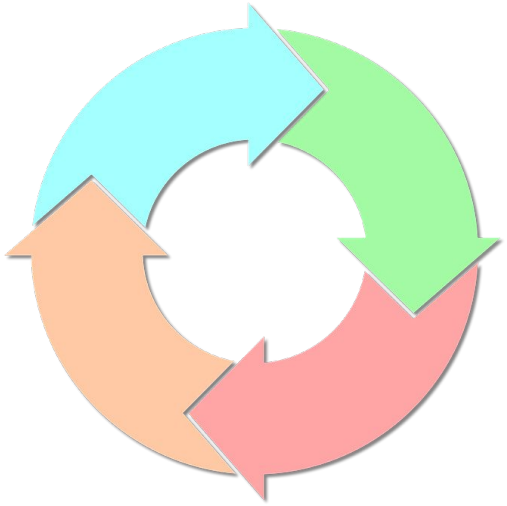
複数チームを統一し、
助け合い・チームでの成果を意識する

開発チームをコーチする

スプリントの過ごし方、ベロシティの見える化、自己組織化を促す

チームで品質を意識する

開発・テストで分けて考えない
QAも開発プロセスの改善提案する

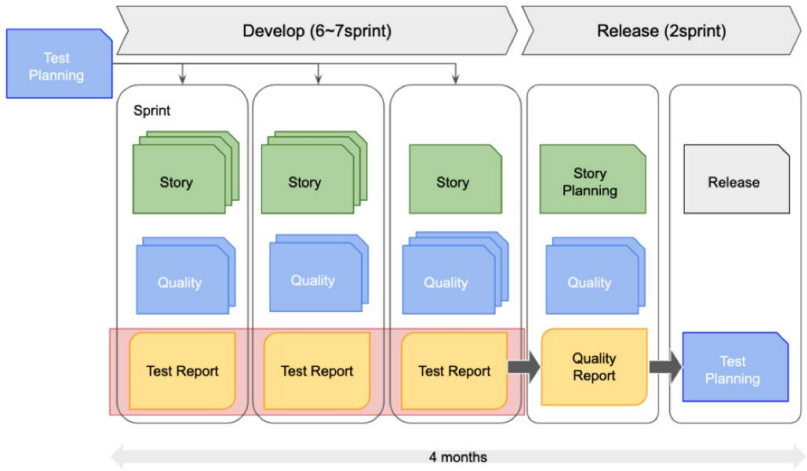


形成期の施策

QAとしてプロダクトの「透明性」を確保する

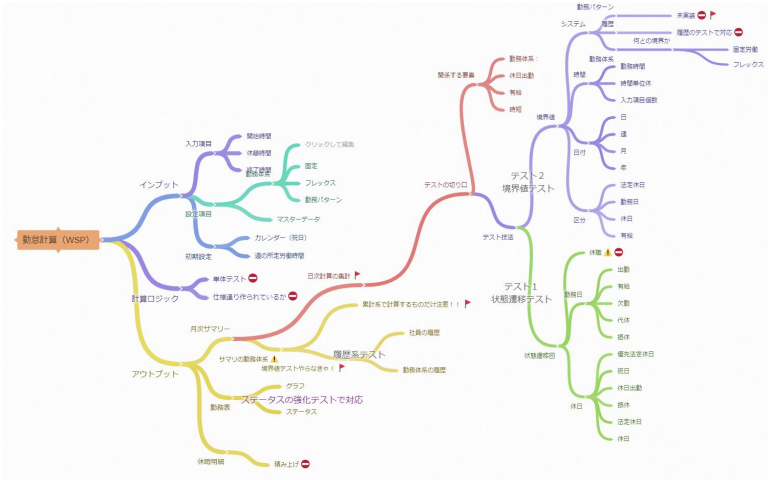
テスト計画

開発サイクルで実施する内容を決める



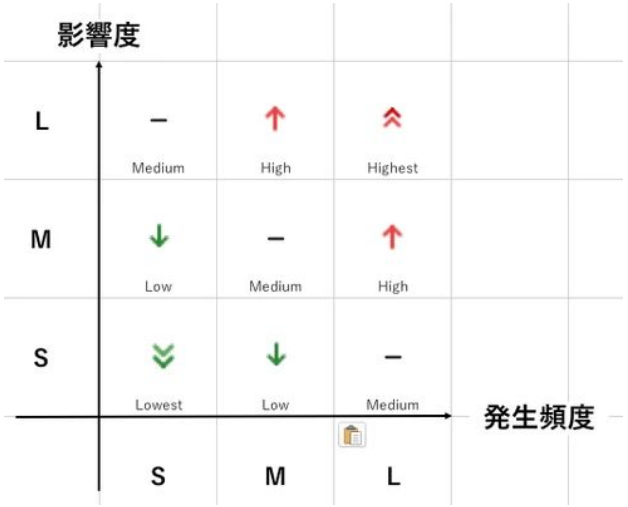
テスト実施

テストが一番得意なエンジニアとしてチームに貢献する



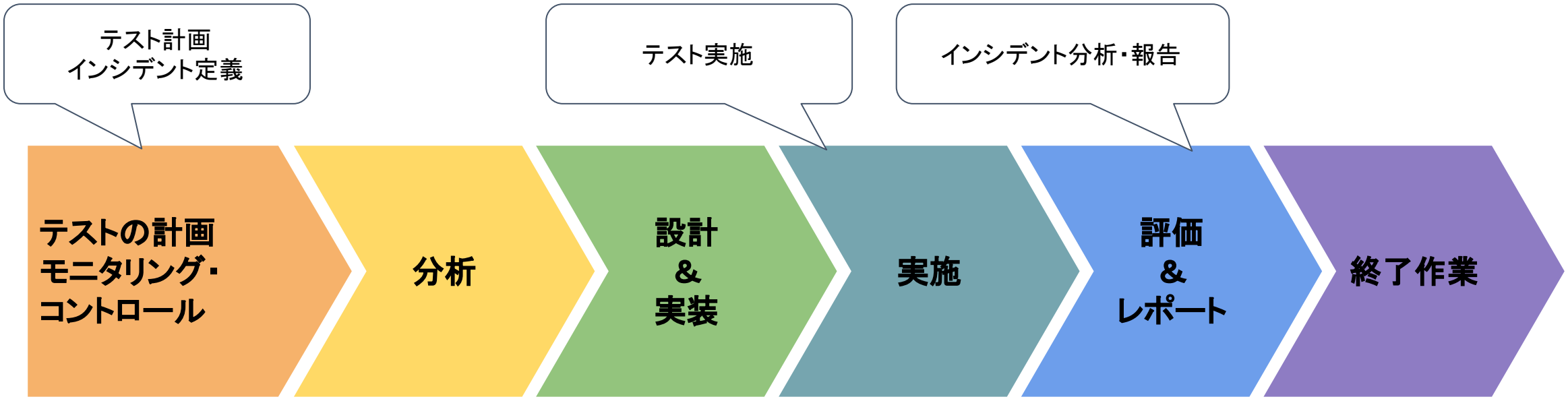
インシデント分析・報告

プロダクトに問題がないか明らかにする



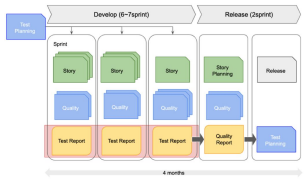
形成期の施策

テストプロセスに基づいて施策を用意した

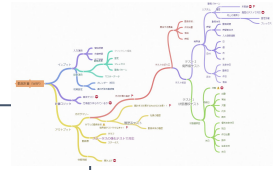


施策の結果（アジャイル開発における品質保証体制）

テストプロセスをスクラムに落とし込む



テスト計画
インシデント定義



テスト実施
インシデント抽出

開発
テスト含む

計画

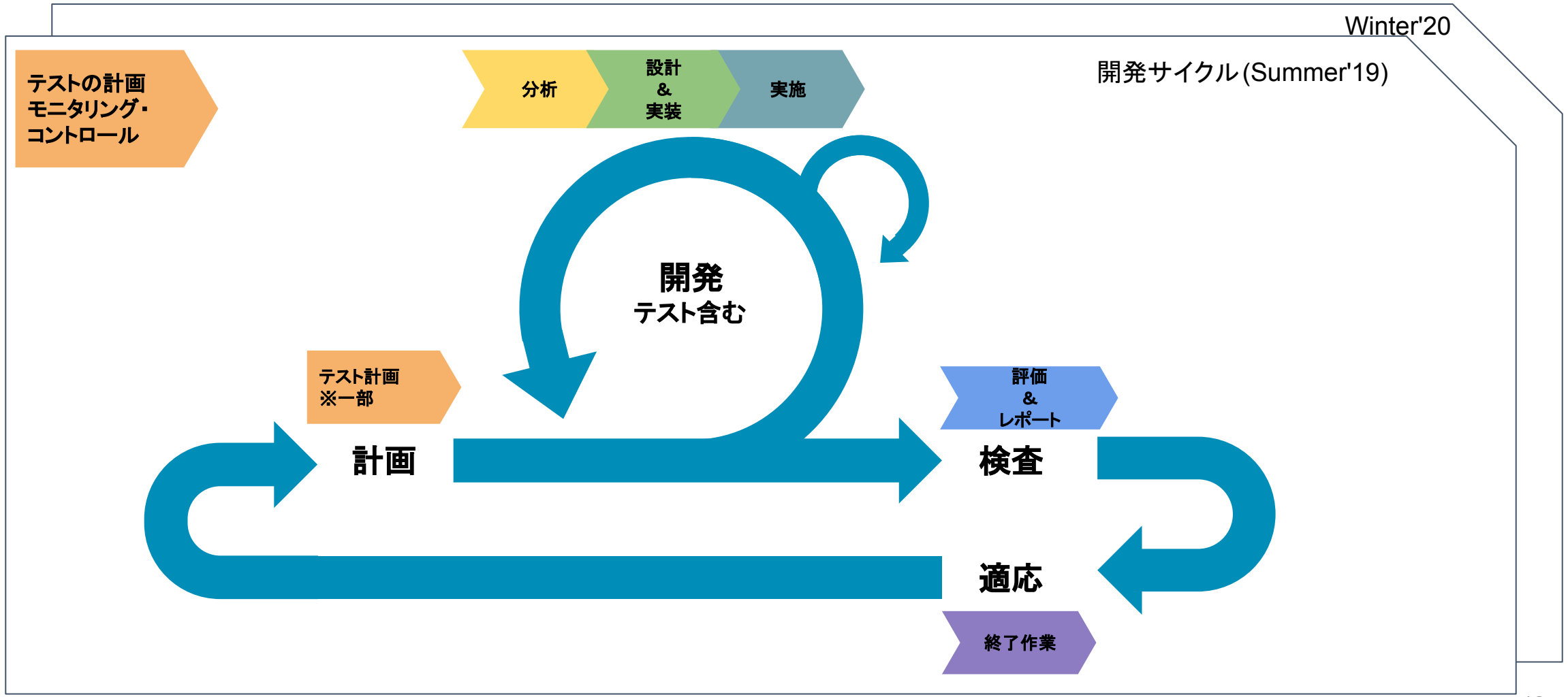
検査

適応

インシデント分析 /報告

施策の結果（アジャイル開発における品質保証体制）

テストプロセスをスクラムに落とし込む



施策の結果（アジャイル開発における品質保証体制）

スクラムチームの品質保証体制を構築し「透明性」を得た

メリット	デメリット
親和性があり、「透明性」を高める仕事である QA ● プロダクトの品質をリードするので、必然的に全体的なチケットの流れを追いかける ● テストを通してプロダクトの品質を明らかにする Scrum Master ● 開発チームのベロシティを高めるために、障害になりえる事象が発生していないかチケットの流れを確認する ● プロセスを通して仕事や無駄の可視化をする **「開発とテスト」の境界を無くす推進力になる** ● メンバーの経験からどうしても境界が見え隠れする ● 理想は「スウォーミング」	**仕事量の調整が大変** ● 任せる・協力する必要があり、苦手の克服になった ● QAがボトルネックにならないように注意が必要 **（間違えると）リリースを妨げる力になる** ● 開発チームのミッションとしてより良いプロダクトを世に出すこと、またその一員であることを意識する

開発とテストを分けない理由

Scrum Masterとして

・「スウォーミング」と「T字型の個人」を目指している

・開発とテストでチームを分けて、「未テスト」の在庫を抱えない

・すべての工程で品質は作りこむ

QAとして

・開発プロセスに積極的に関わる

・フロントローディングの実現

統一期

～変化に適応し、再現する～

統一期の課題と分析

■人数増加に伴う仕組みづくり

人数が多くなるにつれて別の問題が表面化してきた

- 新メンバーのパフォーマンスがあがらない
- チームを分割して新しいチームを立ち上げたい

  →新しいQAも参画し、個人からチームでの品質保証体制に移り変わる必要があった

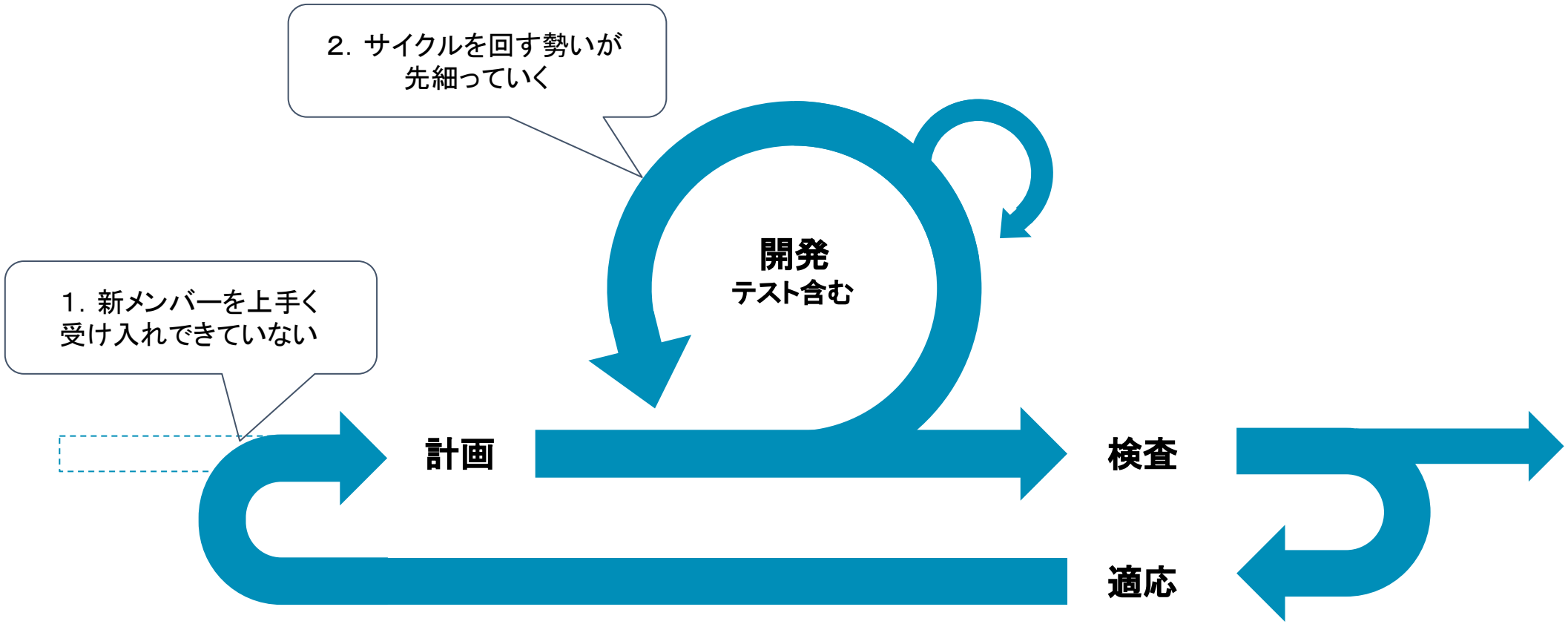
■分析方法

ふりかえりを通して課題が分かるようになったため、

課題に対して改善を行った

統一期の課題と分析

属人化してしまっており、「再現性」がなかった



統一期の施策

スクラムチームの仕組み化

- スクラムイベントの整理
- コーディング規約・WorkingAgreementの更新
- オンボーディング

品質保証系

- 役割定義、キャリアパス作成
- テスト計画書のアップデート
- テスト設計チェックシートの作成
- E2Eテストの作成

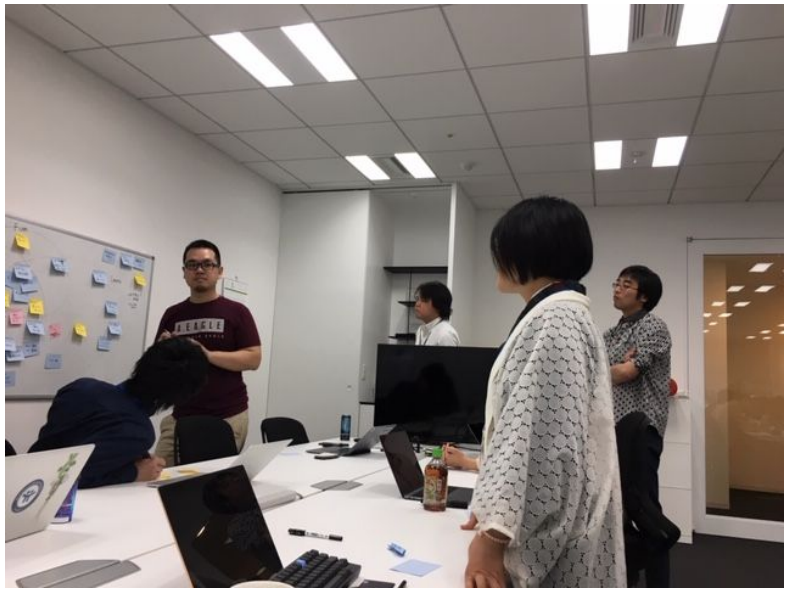
スクラムチームの仕組み化

スクラムイベントの整理・WorkingAgreementの更新

イベントの目的・やることを改めて再整理

明文化することで、新しいメンバーのキャッチアップと

なぜこの形になっているのか変遷が分かるようにした



コーディング規約作成・導入

コードの可視性・可読性がよくなることで、コーディング方法が各人で違うために既存のコードを理解するのに時間をかけてしまい、修正に時間がかかるなど避ける

また、ルールに対して静的解析ツールを導入することで検知ができるようにする。

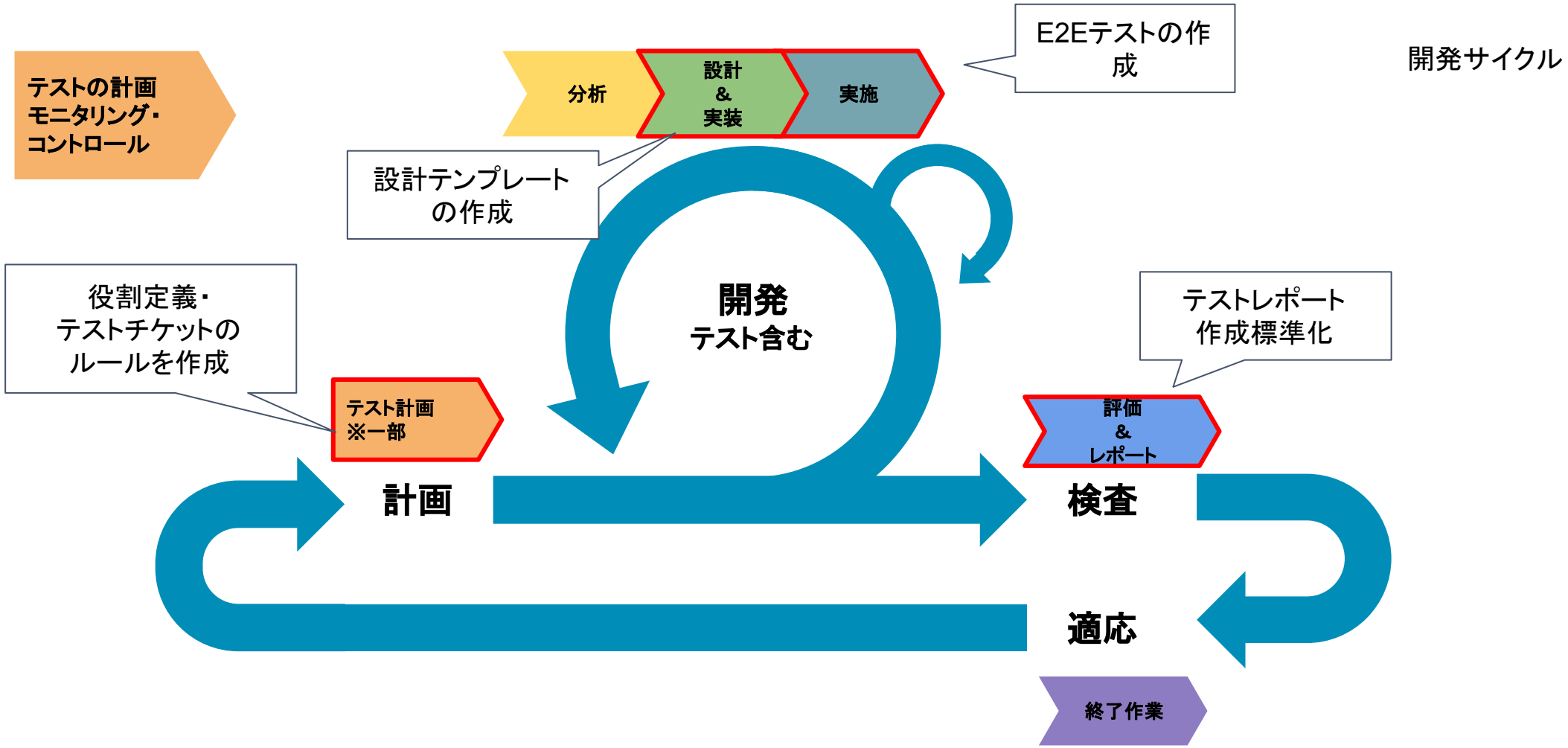
役割定義・キャリアパス

新しいメンバーのレベル感と期待することを決めることで
チームで標準化する優先順位を決めた

役割	期待すること	その他
アシスタント QA	テスト実施 テストレポート作成	要サポート
QAエンジニア	テスト設計 Qualityチケットの消化	
QA リード	テスト計画作成 品質評価レポート作成 リリース判定	
SET	自動テスト作成・整備 CI環境 構築・整備	兼務

統一期の施策

複数のQAで働けるように適応する



自己組織化と標準化

スクラムでは武道から来た守破離の考え方が役に立つ

守：とにかく型に従う、身体で覚える

破：工夫する・質問する

離：同じ結果にたどり着く

2倍のベロシティがでるまでは「守」の状態。
自己組織化と標準化は逆行しない

現在・まとめ

解決したい課題

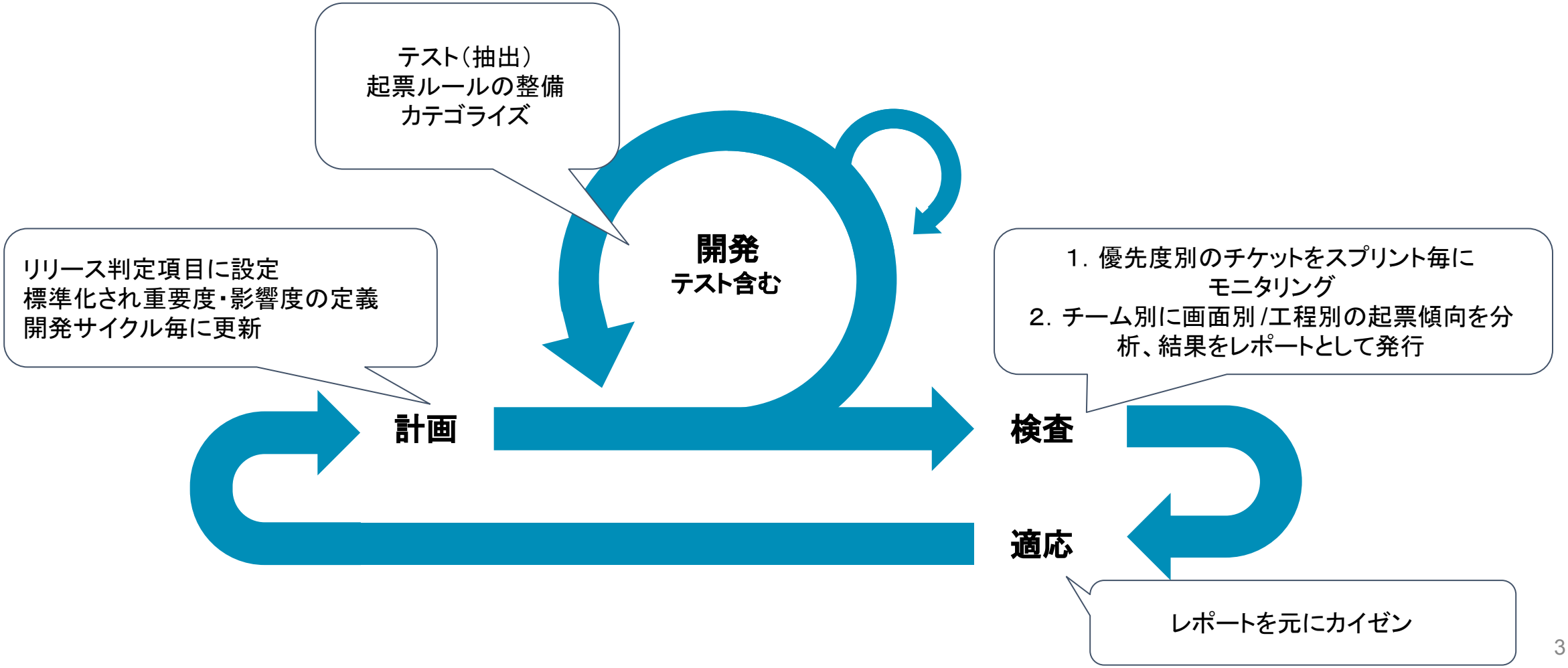
例としてインシデント分析を通して差分を振り返る

形成期	**アジャイル開発における品質保証体制が未整備** 立ち上げ当初のチームはQAエンジニアがおらず、アジャイル開発の経験も少ないメンバーでチームを形成していた。 そのため全体的な品質保証体制の構築が必要だった。
統一期	**人数増加に伴う仕組みづくり** 人数の増加にともない、これまで解決できていた問題が徐々に対応しきれなくなってきた。 個人での品質保証体制からチームでの品質保証体制に移り変わる必要があった。

現在の品質保証体制

インシデント管理の体制が構築できた

2019年4月には優先度最高の障害を出すことなくリリースができた



現在のスクラムチーム

2拠点4チームのスクラムチームで開発中

WSP Unit

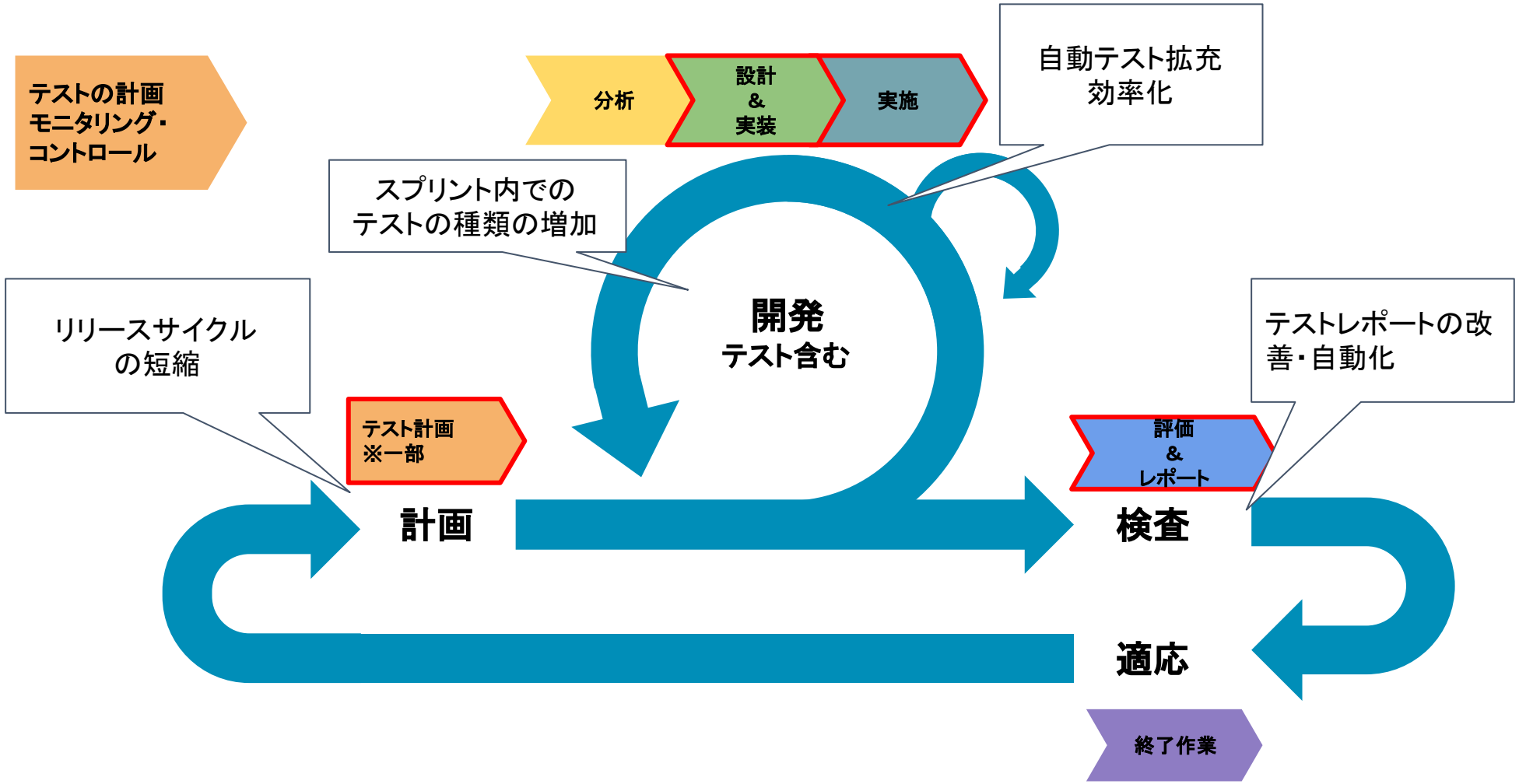
- Team A（勤怠 / Attendance）
- Team B（工数・プランナー / Time Tracking・Planner）
- Team C（経費 / Expense）
- Team D（PSA）

現在のスクラムチーム

ベテランだけでなく若いエンジニアも活躍できるようなチームになってきた

現在の課題・施策の予定

開発規模がスケールできるように仕組みを磨き込んでいく



https://www.teamspirit.com/