

ソフトウェア品質シンポジウム2022

ODC分析を導入する際のハードルを下げた取り組み

一般財団法人日本科学技術連盟 ODC分析研究会 研究テーマ2

- | | | | |
|--------------------|--------|--------------------|--------|
| ○ 京セラ株式会社 | 岡本 慎司 | ブラザー工業株式会社 | 牧野 里香 |
| ○ パナソニックITS株式会社 | 大山 一典 | 株式会社ベリサーブ | 白 宇飛 |
| キャノンITソリューションズ株式会社 | 吾妻 仙一郎 | 富士フィルムイノベーション株式会社 | 石田 敬太郎 |
| エプソンアヴァシス株式会社 | 小島 義也 | りらいあコミュニケーションズ株式会社 | 鈴木 慎太郎 |

- ODC分析手法とは
- 背景
- 導入障壁の調査結果
- 実施概要と結果①～③
- 課題の解決状況
- 今後の展望

従来の不具合分析手法

統計的 分析

不具合の全体を眺める
具体的な対策に直結しない。

原因分析 (RCA)

不具合“固有の原因”究明
時間とコストがかかる。

ODC分析 手法

不具合の要因を分析
要因の数量的な分析
要因毎にタグ(属性)を割り当てる。

個々の不具合の意味論を定量的に捉える。
プロセスの進歩と品質成熟への示唆を与える。

ソフトウェア品質の見える化の効果
が実証されている。

導入検討を進める組織は増えているが、
現在のところ導入事例は少ない。

ODC分析の導入事例を研究するチームを発足

ODC分析導入に課題を持っている開発者を募集、
導入の成功事例/失敗事例の研究を通して、ODC分析の導入障壁を
乗り越えるナレッジを獲得する。

ODC分析導入に関する課題

- ・不安や悩み
- ・導入障壁

ODC分析導入事例を分析

- ・成功事例
- ・失敗事例

ODC分析導入障壁を
乗り越えるナレッジを獲得する

未導入／課題あり



導入済み／課題あり



研究会の参加者へのヒアリング、課題を抽出

- 研究チーム(n=8名)へのヒアリング
- 過去の「ODC分析研究会」(第1期)のワークショップ結果(意見総数:109件)

ヒアリング結果から、以下の課題に取り組むこととした。

環境面の課題

不具合管理表に属性情報の入力場所がない

世の中の不具合管理表を調査、ODC分析手法を標準的にサポートしているものは無さそう。

スキル面の課題

属性情報の分類/分析方法が分からない

属性情報を正しく付与する知識が無い。
属性情報を付与した後の分析方法が分からない。

導入効果の課題

導入効果があるのか分からない

プロジェクトのグラフが実態を表しているか分からない。

課題解決に向けた取り組みを2つの企業で実施

- 企業A(組込系): 実施に向けた環境が準備できていない。
- 企業B(組込系): スキル不足から生じる現場の負担増による抵抗を懸念。
- 企業A,B共通: ODC分析導入による効果についての懸念。

環境面の課題



不具合管理表に属性情報の入力場所がない
世の中の不具合管理表を調査、ODC分析手法を標準的にサポートしているものは無さそう。

企業A 企業B



スキル面の課題



属性情報の分類/分析方法が分からない
属性情報を正しく付与する知識が無い。
属性情報を付与した後の分析方法が分からない。



導入効果の課題



導入効果があるのか分からない
プロジェクトのグラフが実態を表しているか分からない。



実施概要と結果①

企業A:環境面の課題解決への取り組み

属性情報入力の為に不具合管理表を改造したくない

不具合管理表



- | | |
|----------|-----------|
| ・タイトル | ・XXX(未使用) |
| ・担当 | ・原因工程 |
| ・カテゴリ | ・不具合原因 |
| ・発生バージョン | ・重要度 |
| ・対策バージョン | ・開始日 |
| ・優先度 | ・終了日 |
| ・・・・ | ・期限 |

タイプ属性

トリガー属性

インパクト属性

ソース属性





不具合管理表の項目を全部は使っていない。

不具合管理表




- | | |
|---|---|
| <ul style="list-style-type: none"> ・タイトル ・担当 ・カテゴリ ・発生バージョン ・対策バージョン ・優先度 ・・・・ | <ul style="list-style-type: none"> ・ XXX(未使用) ・原因工程 ・ 不具合原因 ・重要度 ・開始日 ・終了日 ・期限 |
|---|---|

入力項目は多いが、使っていない項目がある。

「トリガー属性」の入力項目として、項目名はそのままで再利用してみよう。

原因／対策のエリアには未使用項目はないが、フリーテキスト記述が可能。

「タイプ属性」の情報をフリー記述で入力すれば使えそう。



ODC分析手法をちゃんと理解しないうちに始めてみたら、人によって入力内容がバラバラに。

ODC分析手法に対する理解不足



属性付与のルールを決めていなかった
分析結果にも影響が生じた

▶ トレーニングはしていたが、一部の情報を見て理解した気になってしまった

相互作用

基本操作

負荷/ストレス

バラ
バラ

不具合管理表

- ・トリガー(未使用項目)
- ・原因工程





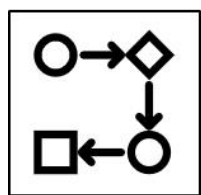
入カールールを統一し、しっかりトレーニングを実施しよう。

属性付与のルールを作成、精度の均一化

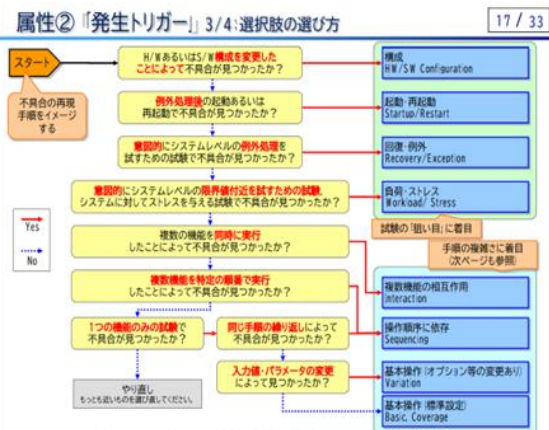
過去のODC分析研究会の資料を活用した

勉強会の開催

リーダークラスを対象に勉強会を開催した



トリガ属性付与フロー



トリガ属性付与フロー



不具合管理表



取り組み内容

- ・既存の不具合管理表の未使用項目を活用した。
- ・フリー記述が出来る記入欄を活用した。
- ・属性情報付与のフローチャートを組織内で共有した。
- ・勉強会を通じてODC分析の知識を習得した。

課題の解決状況

- ・メンテナンスをせずに、既存の不具合管理表を使用できた。
- ・入力する人に関わらず、属性付与の精度の均一化を図れた。
- ・異なるプロジェクト間で、同じ尺度で品質の共有ができた。
- ・スモールスタートでODC分析を部門内に一気に普及できた。

実施概要と結果②

企業B: スキル面の課題解決への取り組み

不具合管理表(既存)は入力項目が多数存在。
現場の入力負担が増え、属性情報を入力してもらえない。

不具合管理表



- | | |
|---|--|
| <ul style="list-style-type: none"> ・タイトル ・担当 ・カテゴリ ・発生バージョン ・対策バージョン ・優先度 | <ul style="list-style-type: none"> ・原因分類 ・原因工程 ・重要度 ・開始日 ・終了日 ・期限 ・.... |
|---|--|

30を超える入力項目

タイプ属性

トリガー属性

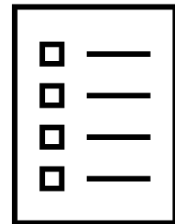
インパクト属性

ケース属性



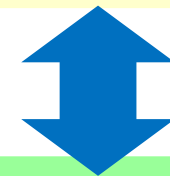
！ 既にある情報から属性情報が導出出来れば、
入力の手間を増やさずに済むのでは？

不具合管理表



- | | |
|----------|---------------|
| ・タイトル | ・ 原因分類 |
| ・担当 | ・原因工程 |
| ・カテゴリ | ・重要度 |
| ・発生バージョン | ・開始日 |
| ・対策バージョン | ・終了日 |
| ・優先度 | ・期限 |
| ・・・・ | ・・・・ |

「原因分類」: 不具合を作り込んだ理由



どちらも**不具合の原因を示す**情報

「タイプ属性」: 不具合の原因を示す属性



原因分類からタイプ属性を導出する
読み替えが出来そうだ。

原因分類を正しく入力する必要があるが、原因分類が多すぎて適切なものを選べない。

不具合管理表

- | | |
|---|---|
| <ul style="list-style-type: none"> ・タイトル ・担当 ・カテゴリ ・発生バージョン ・対策バージョン ・優先度 ・・・・ | <ul style="list-style-type: none"> ・原因分類 ・原因工程 ・重要度 ・開始日 ・終了日 ・期限 ・・・・ |
|---|---|

原因工程

要件定義

基本設計

詳細設計

コーディング

原因分類

原因分類

原因分類

原因分類

原因分類

原因分類

原因分類

原因分類

原因分類

原因分類

原因分類

原因分類

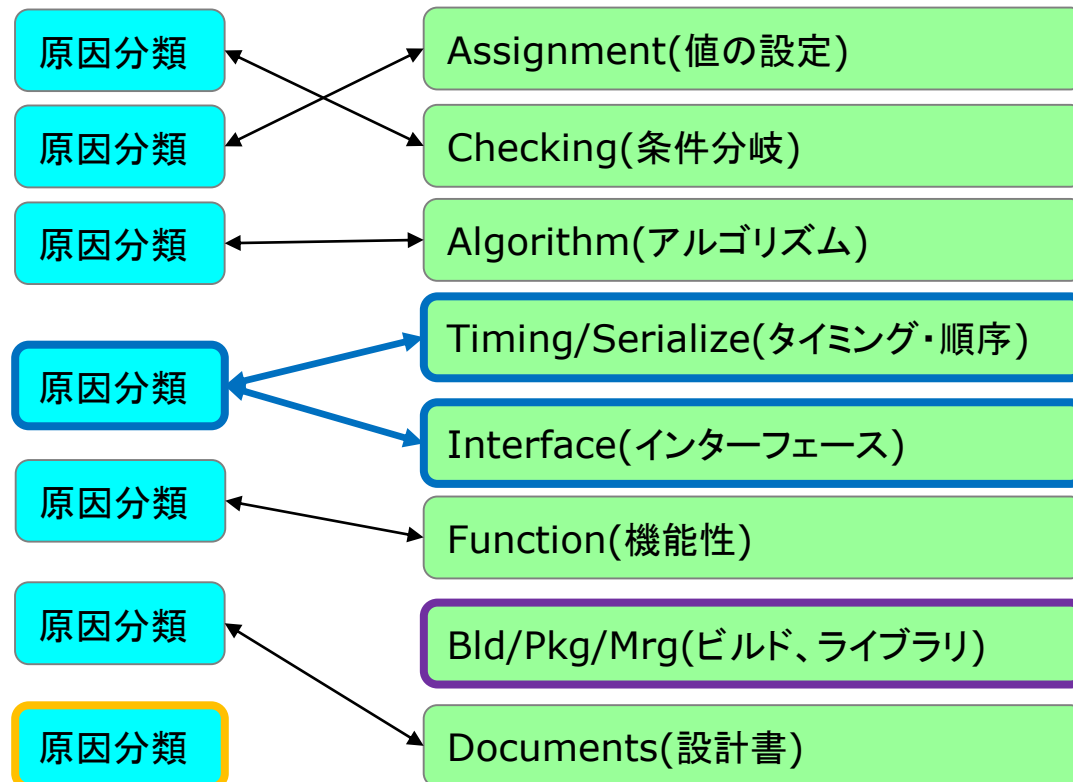
原因分類

原因分類だけで100項目以上存在。
適切な原因分類を選択するのは困難！



ODC分析の観点で原因工程・原因分類を整理したら、原因分類を正しく選択出来るのでは？

タイプ属性



原因分類とタイプ属性を紐づけてみる。

- 原因分類が複数の属性と紐づいた
→ 分類の粒度を見直し
- 原因分類が属性と紐づかない
→ 原因分類の網羅性を見直す or 原因分類の必要性を見直す

取り組み内容

- ・原因分類/工程を整理、タイプ属性を導出可能にした。
- ・整理した情報をBTS(不具合管理システム)に反映した。

課題の解決状況


開発者は整理済みの原因分類から適切な原因を選択、BTS上でタイプ属性情報が付与出来るようになった。

| 原因分類 | 原因工程 | | | | タイプ属性 |
|-------------|------|------|------|--------|----------------------------|
| | 要件定義 | 基本設計 | 詳細設計 | コーディング | |
| データ設定誤り | — | — | ○ | ○ | Assignment(値の設定) |
| 判定条件の誤り | — | — | ○ | ○ | Checking(条件分岐) |
| 状態遷移処理の誤り | — | ○ | ○ | ○ | Assignment(値の設定) |
| 状態管理の誤り | — | ○ | ○ | — | Algorithm(アルゴリズム) |
| 処理ロジックの誤り | — | ○ | ○ | ○ | Timing/Serialize(タイミング・順序) |
| シーケンス設計の誤り | — | ○ | ○ | — | Interface(インターフェース) |
| ユースケース設計の誤り | — | ○ | — | — | Function(機能性) |
| 文書記述の誤り | ○ | ○ | ○ | — | Documents(設計書) |


40項目程度まで削減できた。

実施概要と結果③

企業A,B: 導入効果の課題解決への取り組み



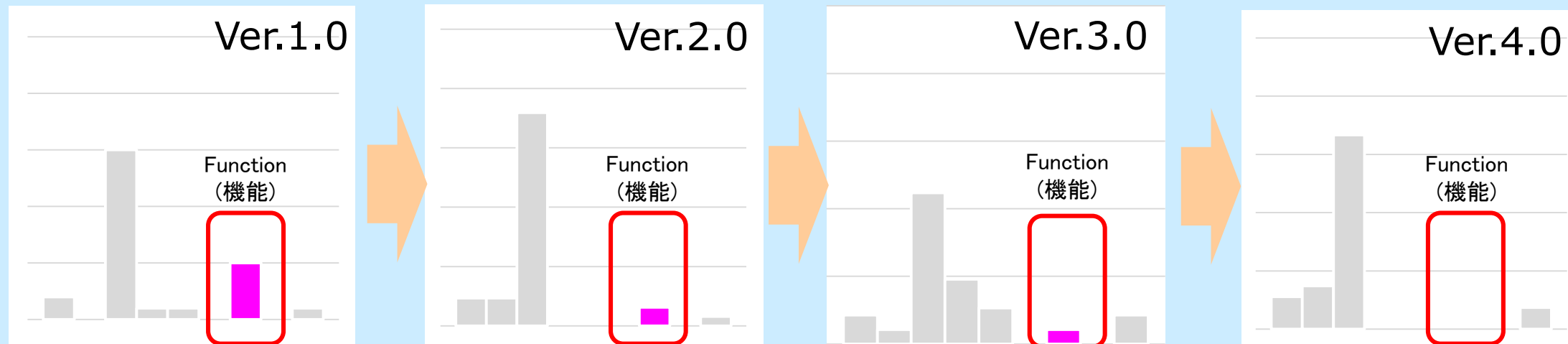
ODC分析を導入したらどんな効果があるのか分からない。



！ ODC分析結果を使うと、どういう効果があるのか、実際にODC分析結果でPRJと会話をしてみよう。

企業Aの導入効果

タイプ属性の経時観察



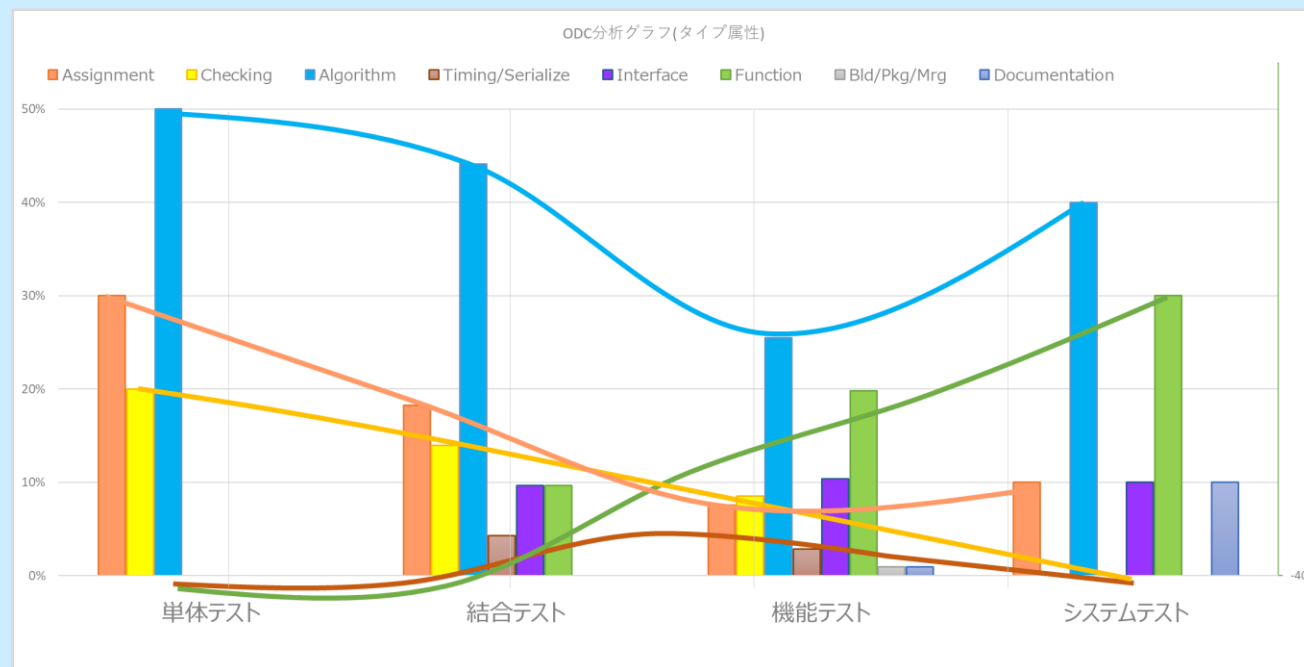
同一製品のバージョン毎にタイプ属性の分析結果を比較観察。
⇒バージョンが上がる毎にタイプ属性「Function(機能)」の障害が減少。

現在

- ・バージョン毎に分析結果を振り返り、次の開発/評価のインプットにした。
- ・複数バージョンで比較を繰り返したことでPRJの弱点の傾向が見えた。
- ・現在も複数バージョンでの比較を継続中。

企業Bの導入効果

実際にデータを使いグラフを作成、PLと認識合わせを実施した。



- ・テスト工程の不具合データにタイプ属性を付与、グラフ化。
- ・テスト欠陥の分布状況についてPLと認識合わせを行った。

その結果、「プロジェクトの状況」が表せていることが確認出来た。

現在

- ・後継PRJではこのグラフをシグネチャー(比較対象)として使用予定。

課題解決の状況

環境面の課題

不具合管理表に属性情報の入力場所がない
世の中の不具合管理表を調査、ODC分析手法を標準化してサポートしているものは無さそう。

解決!

スキル面の課題

属性情報の分類/分析方法が分からない
属性情報を正しく付与する知識が無い。
属性情報を付与した後の分析方法が分からない。

解決!

導入効果の課題

導入効果があるのか分からない
プロジェクトのグラフが実態を表しているか分からない

解決!

- 異なるプロジェクトであっても、同じ尺度でプロセスの品質を共有できた。
- 分析結果を用いて、設計/評価プロセスの課題の見える化ができた。
- ODC分析結果を用いて、プロジェクトの実態を表せることが分かった。

ODC分析研究会

研究活動

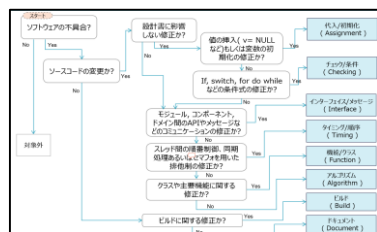
書籍



セミナー



ナレッジ/HowTo



施策 施策 施策 施策

ODC分析を導入したい組織

課題

課題

解決

解決

解決

ODC分析手法を導入するには、「導入課題を解決する実践的なやり方」を組織に合わせた形で提供することが重要。

過去の研究会の活動にもこの実践的なやり方が活動成果として多く残っている。

実践的なやり方をパッケージ化し、多様な組織でODC分析を手軽に始められる

「ODC分析スターターキット」

が作れるのではないかと考えた。

参考書籍

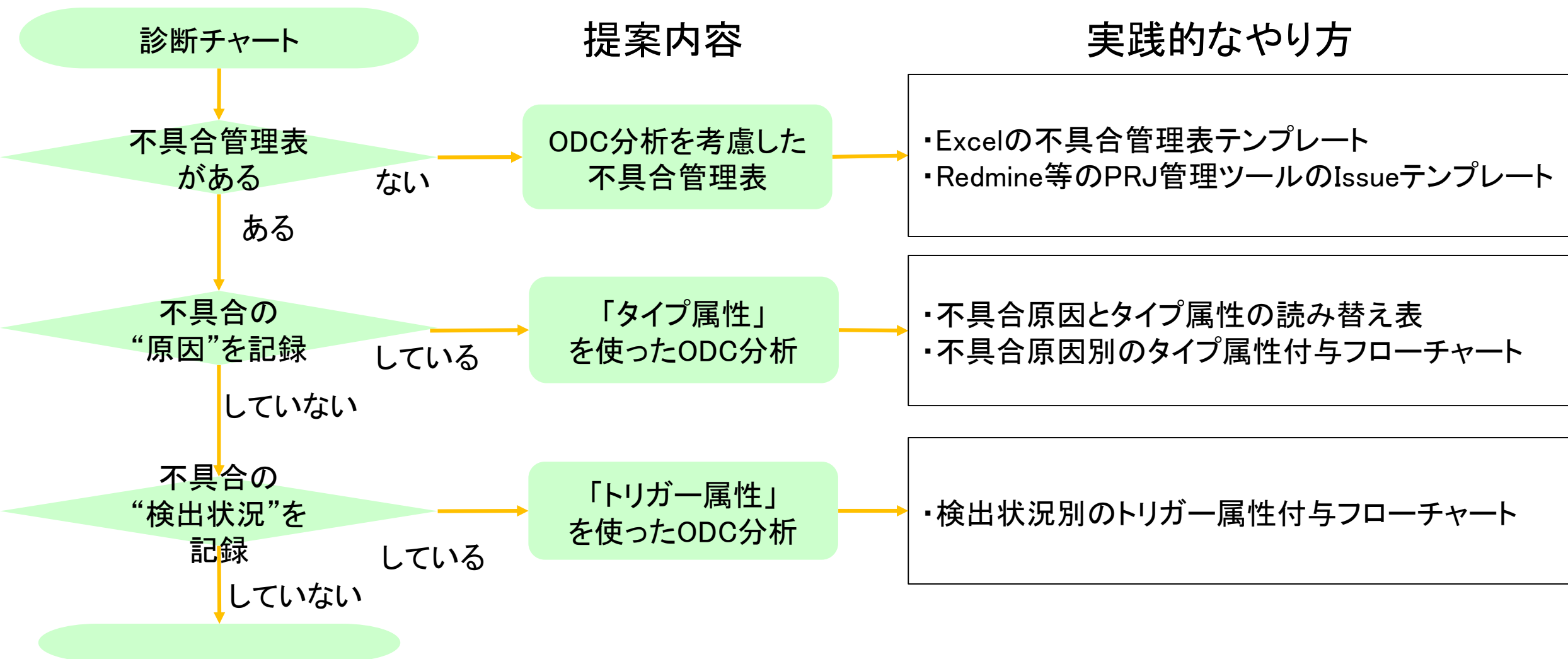
出版社名: 日科技連

著者名: 杉崎真弘・佐々木 方規

書籍名: ソフトウェア不具合改善手法 ODC分析 工程の「質」を可視化する

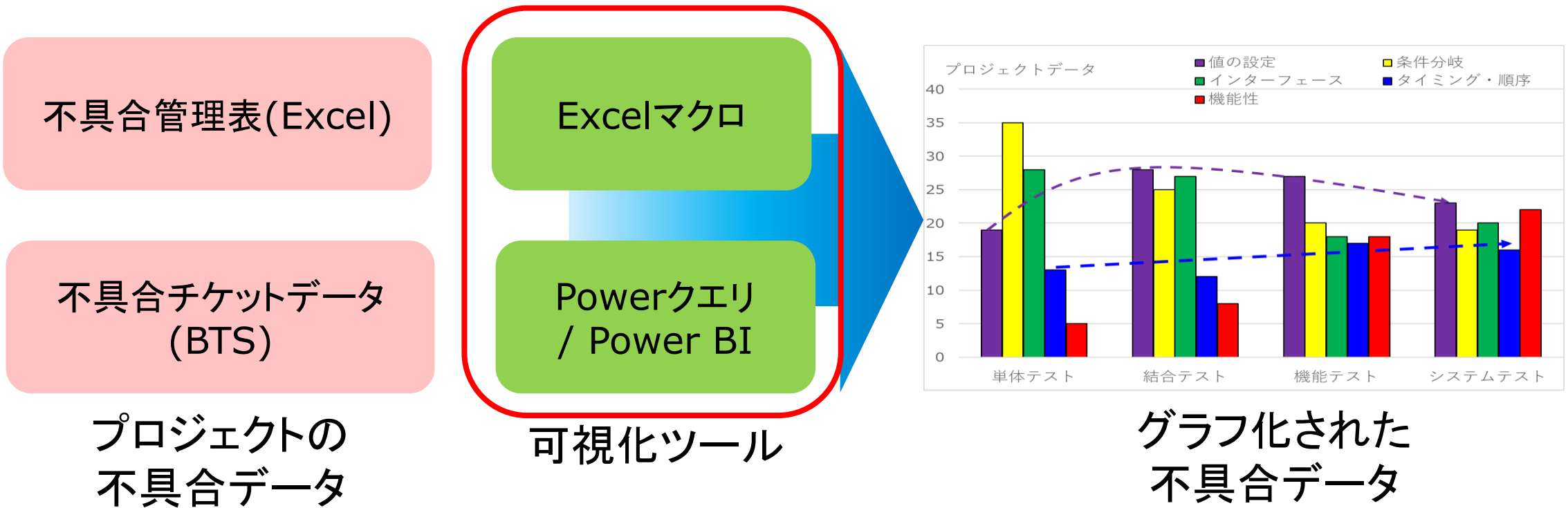
スターターキットの構成要素について検討してみた。

ODC分析導入の診断チャート。状況に応じた導入の提案/実践的なやり方を提示。



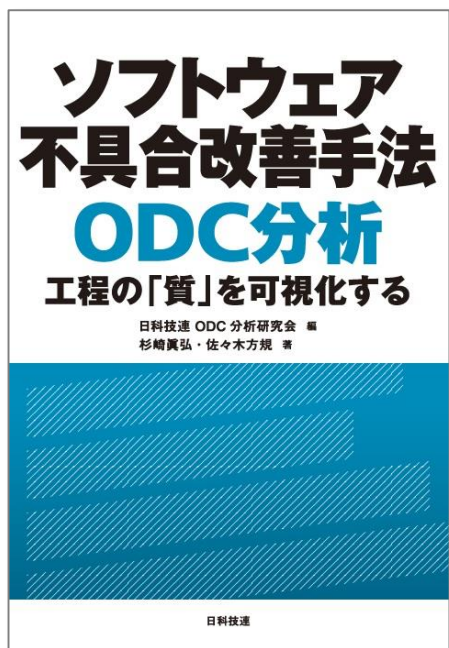
可視化手段の提供

プロセスの品質を可視化するには、プロジェクトデータのグラフ化が必要。
不具合管理表からプロジェクトの現状を可視化するExcelマクロや、
Powerクエリ/Power BIを提供し、可視化/分析の手段を提供する。



比較対象(プロセス・シグネチャー)の提供

ODC分析には、比較対象となるプロセス・シグネチャーが必要。
未導入の組織には実績データが無い為、ODC分析の書籍を
基にしたプロセス・シグネチャーをデータとグラフとして提供。

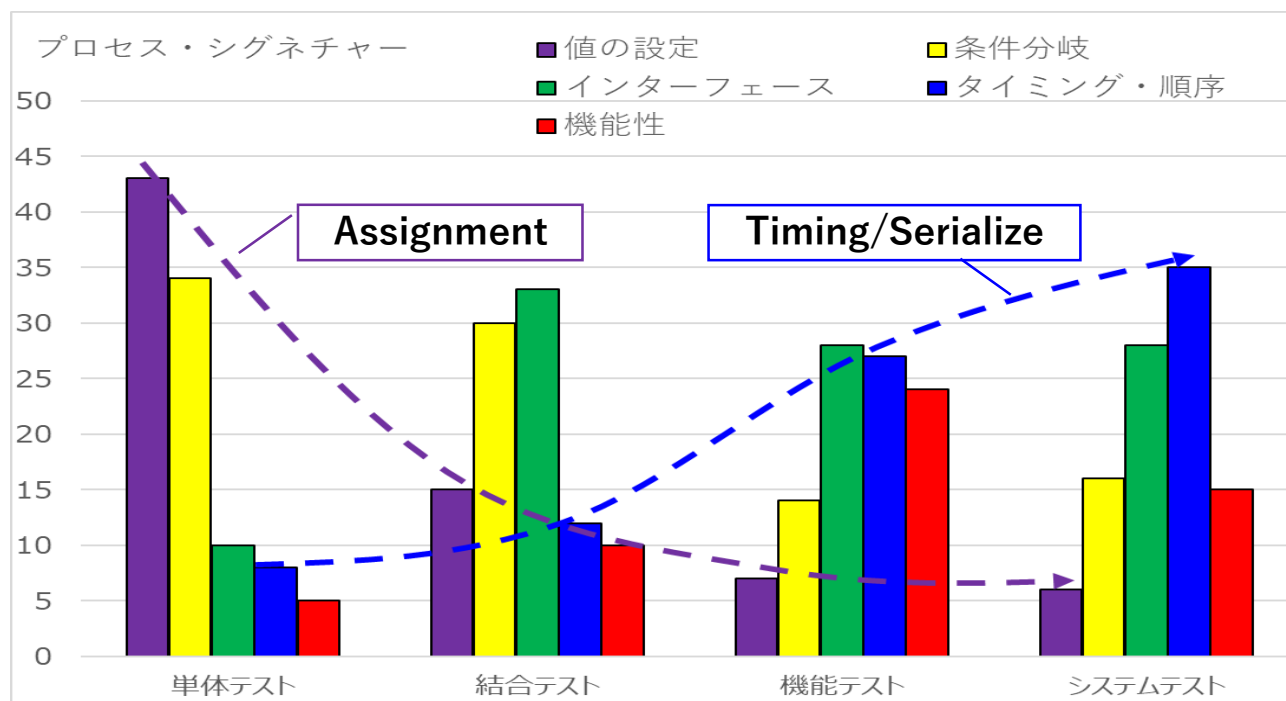


参考書籍

出版社名: 日科技連

著者名: 杉崎真弘・佐々木 方規

書籍名: ソフトウェア不具合改善手法 ODC分析 工程の「質」を可視化する



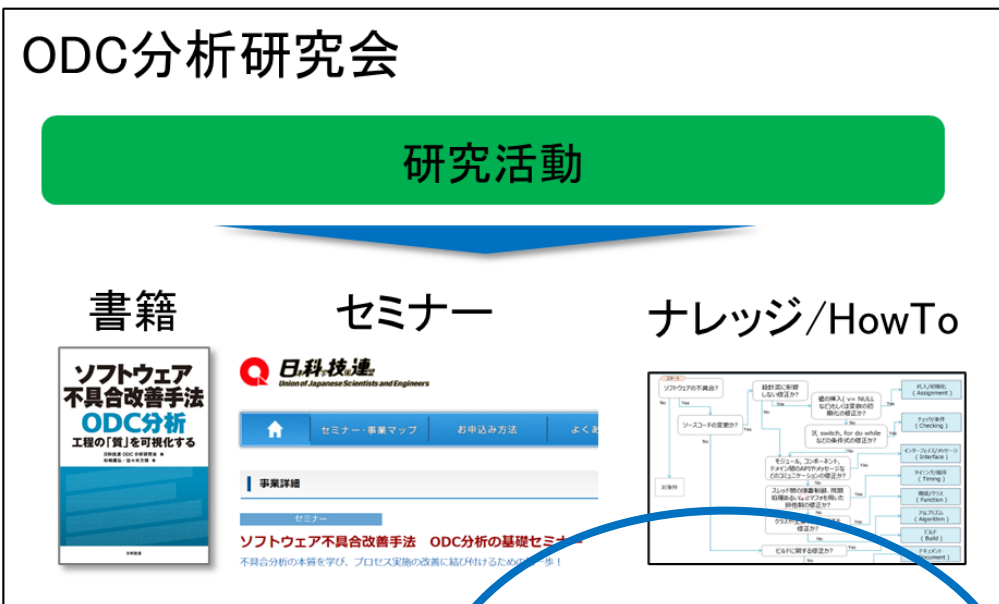
ステークホルダーへの情報提供

ODC分析結果がステークホルダーに対し、どのような情報提供が可能かについて、BMC(ビジネスモデル・キャンバス)形式で整理した一覧を提供する。

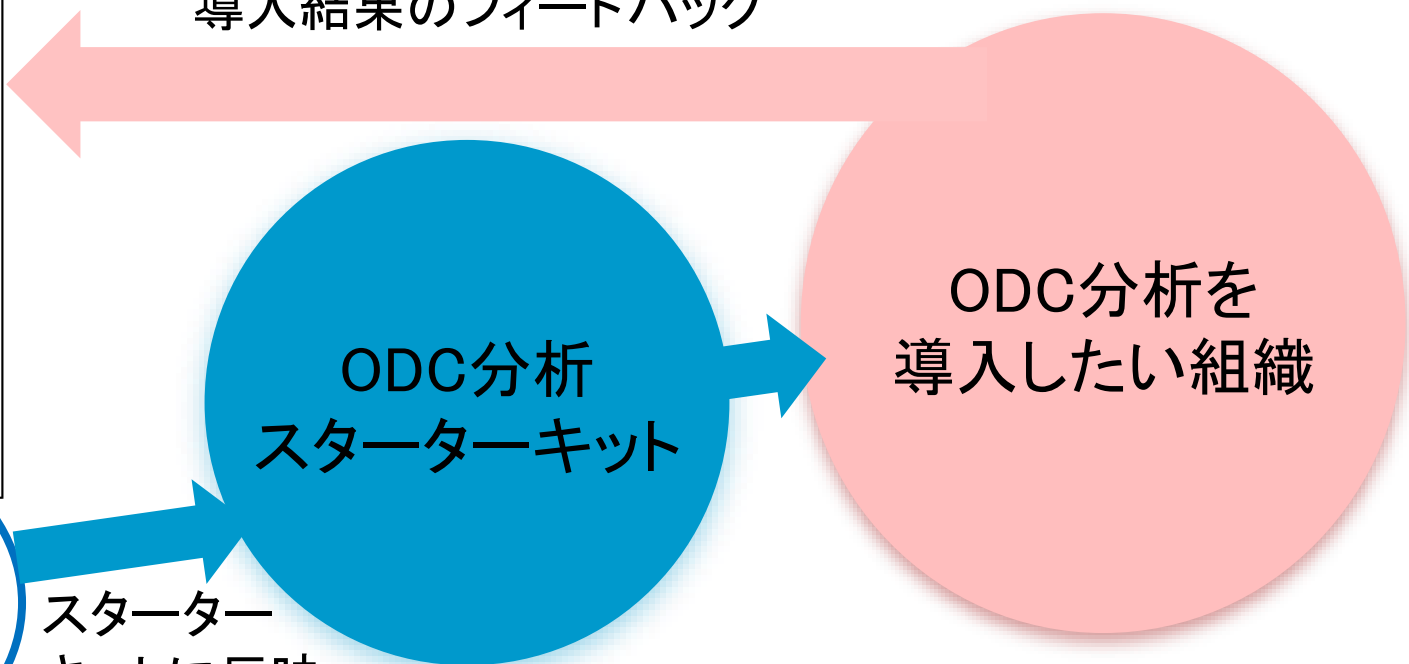
| (主体者) | 主要活動 | 価値提案 | 顧客との関係性 | 顧客セグメント |
|--------------|-------------------------|-----------------------------------|----------------|--------------|
| 検査者 品質管理者 | お客様満足度向上には、非機能要件の満足度が重要 | お客様の側に立って機能要件と非機能要件を分析し、実現価値を提供する | ソフトウェア製品の開発依頼先 | 外部顧客 オーナー |
| | キー・リソース(ODC) | | (対応策) | |
| | インパクト | | 機能要件と非機能要件の検証 | |
| コスト | | 収益 | | |
| 非機能要件の検査費用 | | 顧客への提供価値の増大 | | |

上記は、「ODC提供情報」としてまとめた16件の内のひとつ、**「お客様満足度向上には、非機能要件の満足度が重要」**を「ビジネスモデル・キャンバス」で表現したものの。今後、この抽象的な情報を具体的な用例に落とし込むために整理する予定である。

第4期では、ODC分析導入課題の解決に向けた取り組みを実施し、導入事例をスターターキットとして整理・提供する道筋を立てることが出来た。

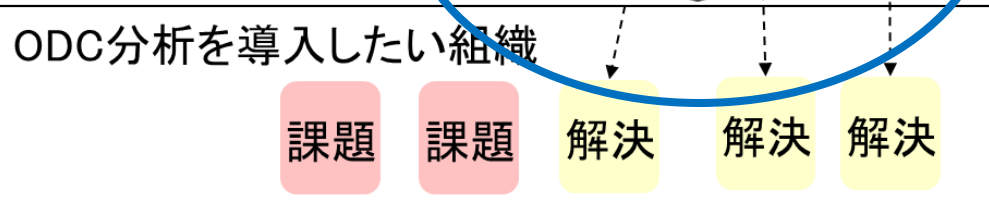


導入結果のフィードバック



施策 施策 施策 施策

スターターキットに反映



第5期も引き続き、ODC分析導入課題の解決に取り組む、実践例をスターターキット化して提供。導入結果のフィードバックを得て、スターターキットそのものの改善を実施していく予定。

ご清聴ありがとうございました。