

# テスト要求分析チュートリアル

テスト設計コンテスト実行委員





# お話すること



これまでの  
プロセス

テスト設計 ※またはテスト計画、テスト準備など表現は様々

テスト  
実施

JSTQBの  
テスト開発  
プロセス

テスト分析

テスト設計

テスト実装

テスト  
実行

本講での  
テスト開発  
プロセス

テスト  
要求分析

テスト  
アーキテクチャ  
設計

テスト  
詳細設計

テスト  
実装

テスト  
実施

テスト要求の  
獲得と整理/  
テスト要求  
モデリング

テスト  
アーキテクチャ  
モデリング

テスト技法の  
適用による  
テストケースの  
列挙

手動/自動化  
テストスクリプト  
(テスト手順)の  
記述

テスト要求分析の一部

開発プロセスの図は以下より：

[http://aster.or.jp/business/contest/doc/2019\\_U-30\\_V1.0.0%20.pdf](http://aster.or.jp/business/contest/doc/2019_U-30_V1.0.0%20.pdf)



# お話すること



- テスト開発プロセス（テストプロセス）のうち  
テスト要求分析 に焦点を当てます。
  - テスト設計の良し悪しは  
テスト設計成果物のみでは語れません。
- やり方は多種多様。  
「なぜテスト要求分析を行うか？」考え、  
自分たちに合うやり方を考えるヒントにして  
いただければ幸いです。



今回の範囲

集める(準備含む)



分ける(分解、分類)



整理する(構造化)



観点を出す



テスト計画に反映



## 1. テストの課題

2. テスト要求分析とは

3. テスト要求分析のやり方

4. 過去の応募作の審査で  
気になった点に対して



# テスト要求分析とは



本講での  
テスト開発  
プロセス

**テスト  
要求分析**

テスト要求の  
獲得と整理/  
テスト要求  
モデリング

**テスト  
アーキテクチャ  
設計**

テスト  
アーキテクチャ  
モデリング

**テスト  
詳細設計**

テスト技法の  
適用による  
テストケースの  
列挙

**テスト  
実装**

手動/自動化  
テストスクリプト  
(テスト手順)の  
記述

**テスト  
実施**

テスト開発プロセスの図は以下より：

[http://aster.or.jp/business/contest/doc/2019\\_U-30\\_V1.0.0%20.pdf](http://aster.or.jp/business/contest/doc/2019_U-30_V1.0.0%20.pdf)



# そんなに困っていないけど、、、



「困っているのはテスト要求分析じゃないよ」…？

なんとなくやれてしまうこともある

- 長年のつながりで、なんとなくできてしまう
- プロセス上それなりの要求分析は行うようになっている

「テストアーキテクチャに不整合がたくさん」…？  
「テストケースが分かりにくいと言われる」…？

テスト要求分析で失敗していたのかも



# テスト要求分析の理解レベル

---



- テスト要求分析の説明はすごく簡単に聞こえる
- 受け手によって深さがマチマチ





## 1. テストの課題

## 2. テスト要求分析とは

## 3. テスト要求分析のやり方

## 4. 過去の応募作の審査で 気になった点に対して



# 用語の定義



本講での  
テスト開発  
プロセス

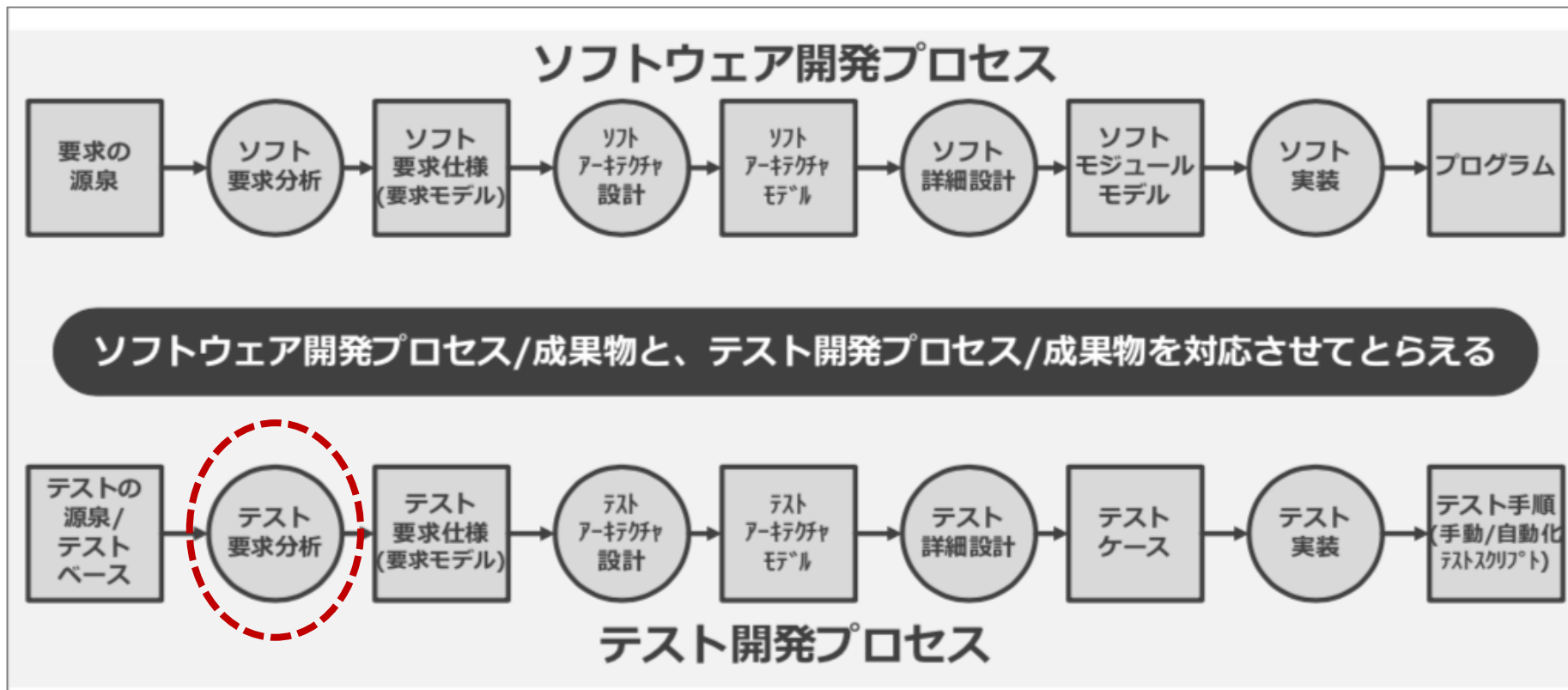


テスト開発プロセスの図は以下より：

[http://aster.or.jp/business/contest/doc/2019\\_U-30\\_V1.0.0%20.pdf](http://aster.or.jp/business/contest/doc/2019_U-30_V1.0.0%20.pdf)



# 本書での用語の意味



以下より（点線の丸を除く）：

[http://aster.or.jp/business/contest/doc/2019\\_U-30\\_V1.0.0%20.pdf](http://aster.or.jp/business/contest/doc/2019_U-30_V1.0.0%20.pdf)

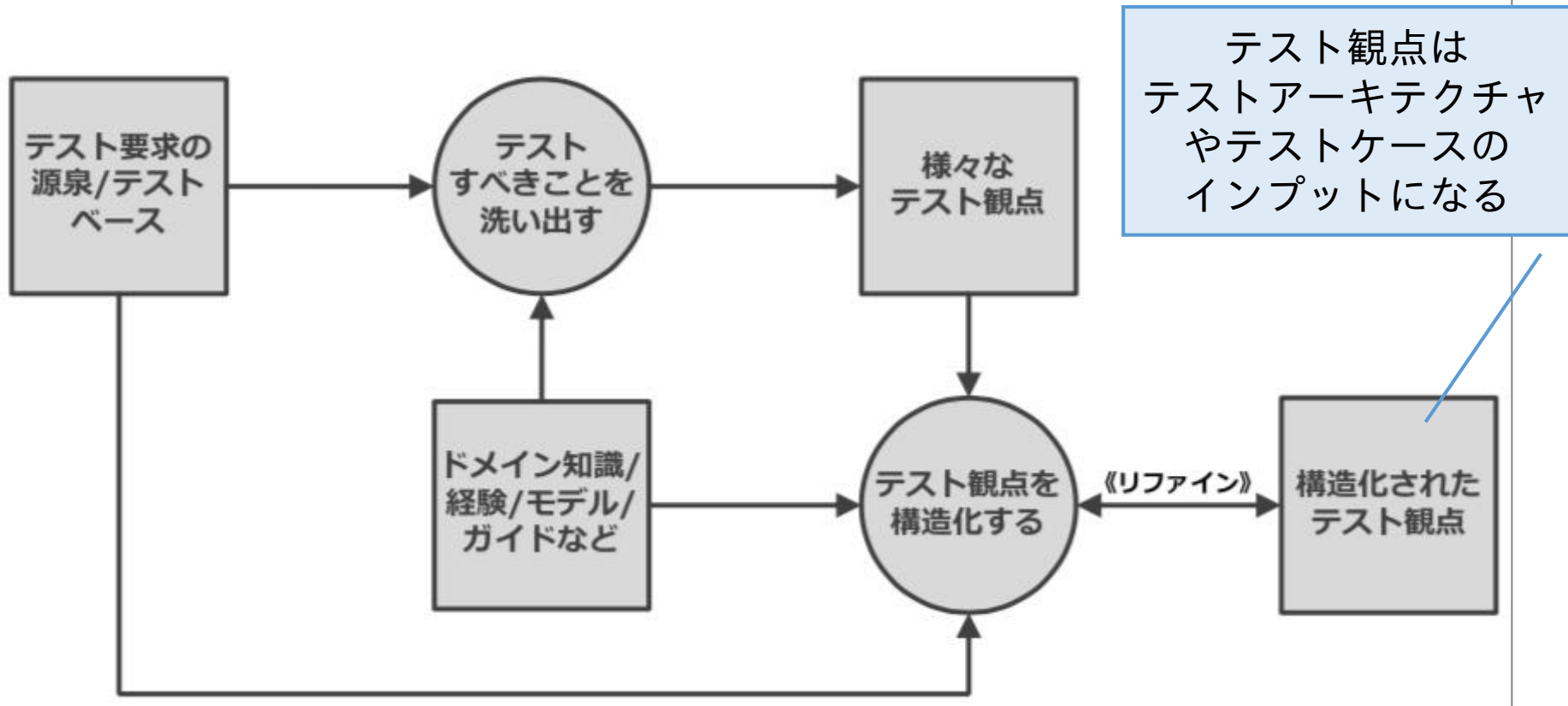


# 本書での用語の定義



## テスト要求分析の一例

※テスト要求分析を定義したものではなく、具体的にイメージするためにテスト要求分析をプロセス設計した一例



以下より(吹き出しを除く) :

[http://aster.or.jp/business/contest/doc/2019\\_U-30\\_V1.0.0%20.pdf](http://aster.or.jp/business/contest/doc/2019_U-30_V1.0.0%20.pdf)



# テスト要求分析では



- 以下を決める
  - どういった目的でテストするか
  - どこまで、どんなテストをするか

この解釈に大きな壁が存在する



# どういった目的でテストするか



## よく言われること

- バグを見つけること
- ちゃんと動くことを保証すること

## もっと深く考えてみる

- なぜそこは厚くテストするのか
- なぜその部分を先にテストしないといけないのか
- そのテストで何の課題を解決したいのか



# どこまで、どんなテストをするか



## よく言われること

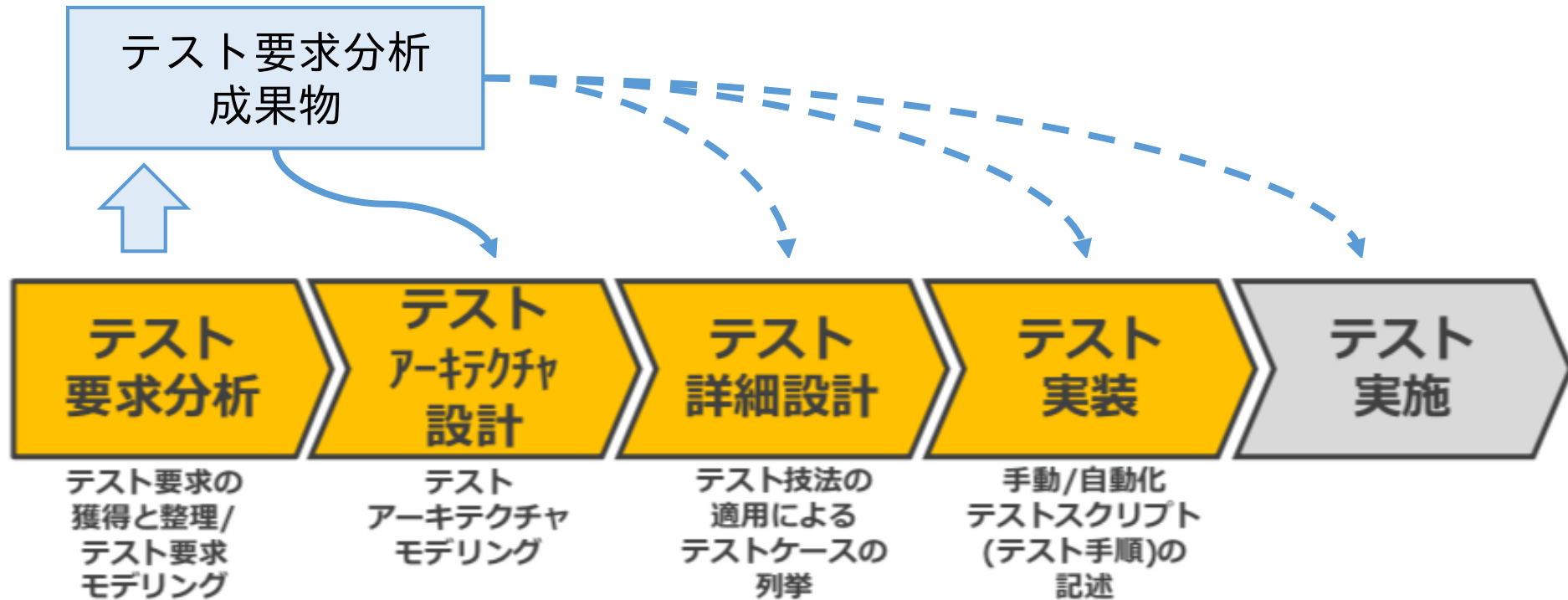
- テストの範囲を明確にすること
  - インターフェースのある箇所
  - 他社製のプログラムを使っている箇所
- どんなテストをするか明確にすること
  - 機能テスト、非機能テスト

## もっと深く考えてみる

- なぜテスト範囲がそこでいいのか
- なぜ非機能テストが必要なのか
- ××ごと（例えば「機能ごと」）に分けるだけでいいのか



# テスト要求分析の影響箇所



テスト開発プロセスの図は以下より：

[http://aster.or.jp/business/contest/doc/2019\\_U-30\\_V1.0.0%20.pdf](http://aster.or.jp/business/contest/doc/2019_U-30_V1.0.0%20.pdf)





# テスト要求分析の役割



- テスト要求分析結果は、他のプロセスの根拠や背景になる
  - テスト要求分析の成果物は他の成果物と整合するべき
  - 他のプロセスと行き来しフィードバックを取り込みながら進める

意図的に説明できるようにしておかないと、なかなかできない

なぜそう考えたか？説明でき  
ステークホルダーと合意できる内容であること



1. テストの課題

2. テスト要求分析とは

3. テスト要求分析のやり方

4. 過去の応募作の審査で  
気になった点に対して



今回の範囲

集める(準備含む)



分ける(分解、分類)



整理する(構造化)



観点を出す



テスト計画に反映



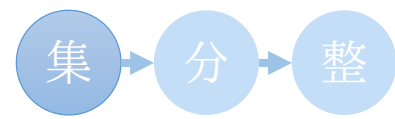
# どこから集める？



- テスト要求の集め先は様々
  - テスト対象やプロジェクトの状況にも依存する
- まず、集め先を特定する



# 集め先の特定



- 製品やサービス

- テスト対象分析をする

- テスト対象の使われ方などテスト対象を理解するための分析をする
    - ≡仕様

製品や  
サービス



- 製品やサービス

- テスト対象分析に必要なもの
  - 実際の物
  - 開発時に作成された文書
    - 設計書
    - ユーザーズマニュアル
  - 想定された使われ方(≡仕様)
  - 問題が起きやすいところ、考えられるリスク



# 陥りがちなテスト要求分析

集

分

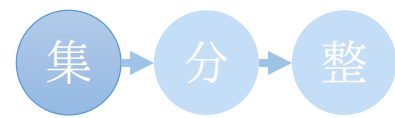
整



- テスト対象以外から得られるテスト要求を忘れる
  - テスト対象分析ができていればテスト設計は行える
    - 頭の片隅で対象以外の分析も本当は行っている  
→共有しないと、もったいない
  - テスト対象分析だけで終わってしまうと？
    - テスト対象が改修された経緯がわからず、  
変更弱いテスト設計になってしまう かもしれない
    - テストを厚くする箇所が的確でなくなる かもしれない



# 集め先の特定



製品やサービスの周辺には人(ステークホルダー)がいる



どんなステークホルダーがいるか？考える





# 集め先の特定

集

分

整



## • どんなステークホルダーがいるだろうか？

### • 手法

#### • 製品やサービスのライフサイクルから考える

- いつ、どこで、どんな人がどんな関わり方をするか？

#### • ステークホルダー識別

- 「ソフトウェア及びシステムに直接関与する一次ステークホルダー」

「直接的には関与しない二次ステークホルダー」

- 「要求とステークホルダーの関係を明らかにするために、ステークホルダーを影響度と重要度の特性で分析する。」

(「」内はソフトウェア品質知識体系ガイド -SQuBOK Guide-(第2版), SQuBOK策定部会 編, オーム社, 2014/11 より)

- 本書では、  
使用者、生産者、開発メンバー、テストメンバー  
の例で考える



- 使用者の要求

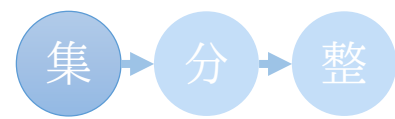
- 使用者の特定と、期待や実使用感の確認
  - 製品やサービスを利用する人は思った以上に多い。
    - 実ユーザー
    - 利益を得ている人
    - メンテナンスする人
  - 立場により、期待や実使用感は異なる。



- 生産者の要求
  - この製品にかける思惑
    - プロダクト戦略
      - フィーチャー
    - 品質についてどう考えているか
- 開発メンバーの要求
  - 開発時の制限や組織的な思惑
    - 開発方針
    - 体制
    - 予算
- テストメンバーの要求
  - テスト時の制限や組織的な思惑



# 集めるときの注意点



- 必要な情報を多面的に集められるようにする
  - 集めた情報を分けて俯瞰する
  - ロジックツリー
- 深掘りの方向
  - 何を必要と思うか、は考えやすい
    - なぜそれが必要だと思ったの？
  - 状況変化の情報を集め忘れないように
    - 過去プロジェクトからの変化、プロジェクト内の変化
    - なぜ開発メンバーが休みがちに？ なぜハードを変更？
- MECE（漏れもだぶりもない）か？  
深掘りが不足しているところはないか？



- 想像する
  - 集められなかったすべての立場について想像する
- 想像できても忘れてはいけないこと
  - できれば、事実を集めたほうが良い
  - あるいは、事実をインプットに想像する
    - 想像に頼りすぎると、説明しにくくなる
    - 想像を、うまく使う
      - 思考の幅を広げるきっかけにする



- 使用者を分解して想像する

- 手法

- ステークホルダー識別
    - ステークホルダーを洗い出し、特徴や使う場面を踏まえて、それぞれの気持ちや反応を考える

- より具体的に想像する

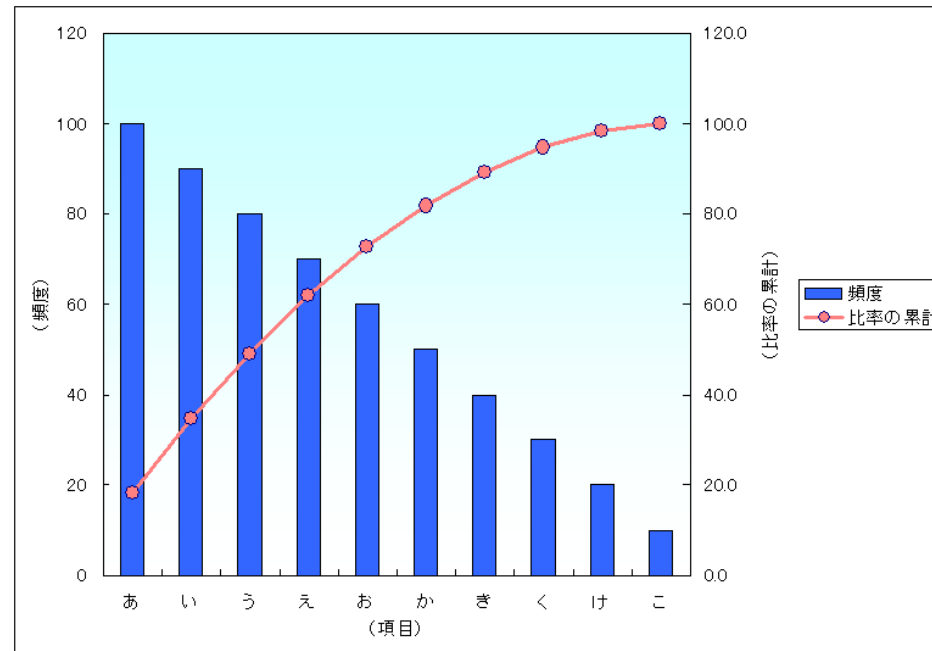
- 手法

- カスタマージャーニーマップ
    - ペルソナ
  - ユーザーが体験するであろうことをきちんと情報収拾する
  - ペルソナに優先度をつける
    - 製品やサービスのターゲットを確認し、重み付けをする
    - パレート分析などで、重み付けをする



## • パレート分析

- 出現頻度を確認し、  
どれを対象にすれば効果が大きいか、導き出す
  - 例：どのグループを対象にすると  
利用者のうち8割をカバーできる？

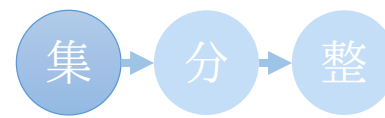


グラフは以下(最終更新 2017年4月11日 (火) 07:42 )より

<https://ja.wikipedia.org/wiki/%E3%83%91%E3%83%AC%E3%83%BC%E3%83%88%E5%9B%B3>



# 生産者の要求の想像



- 製品の強み弱みを知る

- 手法

- 3C分析

- 顧客 (Customer)、競合 (Competitor)、自社 (Company)

- SWOT分析

- 強み (Strengths)、弱み (Weaknesses)、機会 (Opportunities)、脅威 (Threats) の4象限を考える。

	プラス要因	マイナス要因
内部環境	強み	弱み
	...	...
外部環境	機会	脅威

- 生産者がどちらに行きたいかを想像する

- 強みを伸ばす？ 弱みをなくす？





# 開発(テスト)メンバーの要求の想像

集

分

整

- 組織の強み弱みを知る
  - 手法
    - SWOT分析
- 自組織がどうなっていきたいかを想像する
  - 行動指針
  - 目標
  - KGI (Key Goal Indicator)
  - KPI (Key Performance Indicator)



- 品質を区分して考える

- 品質特性

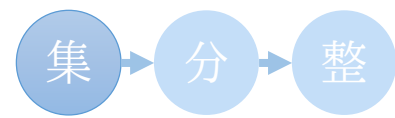
- ISO/IEC 25010などのSQuaREシリーズ

- 「当り前品質」 「魅力的品質」 「一元的品質」 …

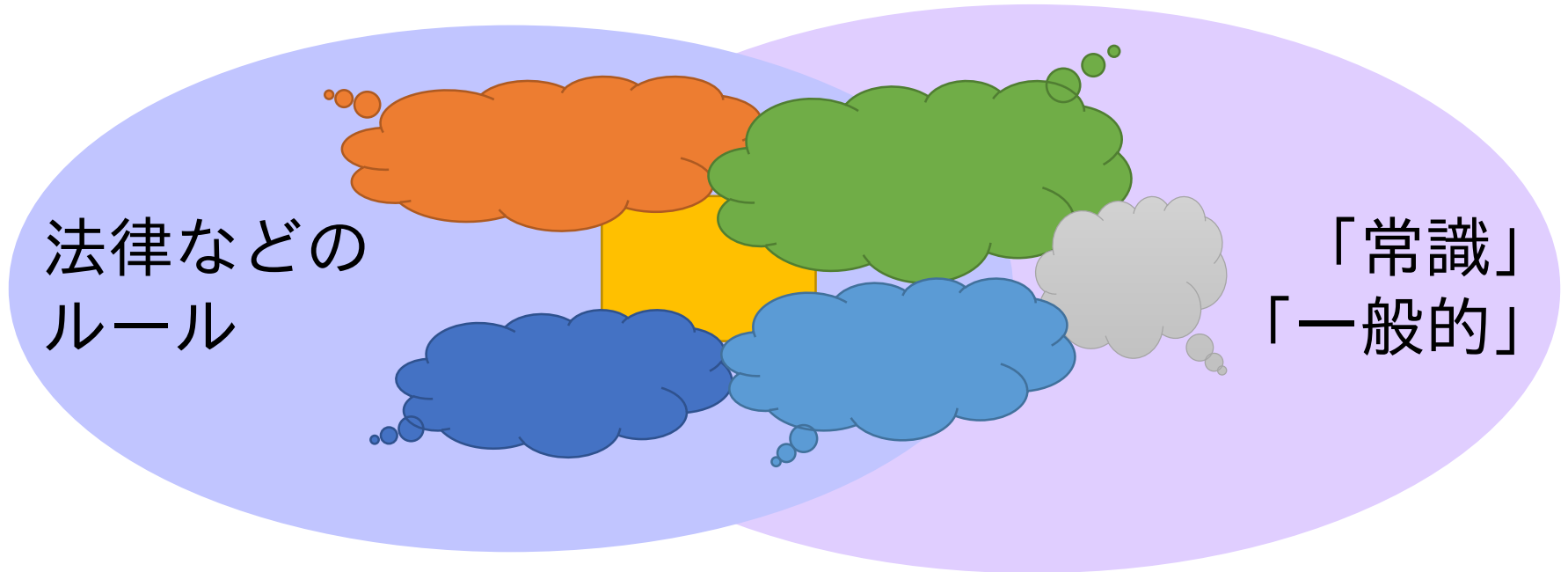
- 既存の区分けは便利だが、  
やりたいこととの相性の確認や  
内容の取捨選択については、検討すべき



# 集め先の特定、集める物



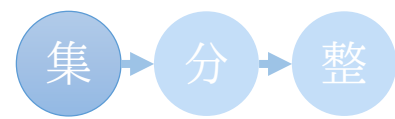
誰からも出てこなくても気する必要があること



- ルールは、調べる
- 類似製品や経験から考える
  - 例 NGの理由が読み取れない入力エラーメッセージ



# なぜこんなに集めるの？



テスト対象以外の情報を、なぜそんなに集めるの？

- すべてのテストをすることはできない
  - テストの厚みをコントロールする
- 一歩先を見たテストができるように
- 変化に強くする



# 「集める」のおまけ



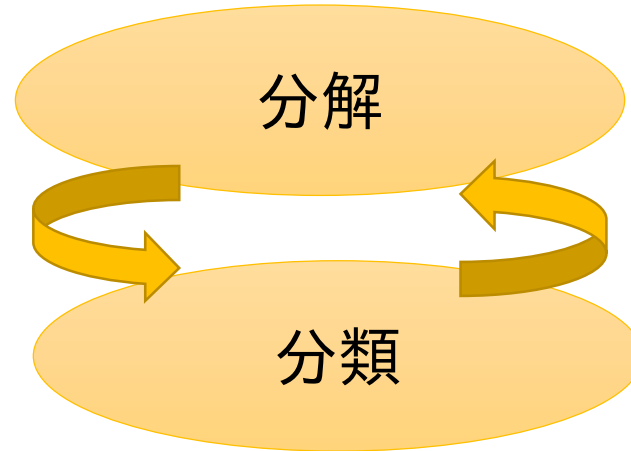
- 様々なところから集める
  - テスト対象
  - 人
  - テスト対象や人を取り巻くルールや常識
  - ...

Q : テストエンジニアが、そこまでやらなきゃならないの？

A : 組織でどこまでやれるか、やるべきか、決める必要あり

✗ 「チュートリアルで聞いたから、必要かわからないが、やる」

- 重要なことは、理由や問題を理解して行うこと
  - なぜその分析が必要なのか
  - 自分たちのテスト設計を良くする為に必要な点は何か  
→結局はテスト要求分析と一緒に



- まず分解
  - 複数の要求の集合になっていないか？
  - 文章を図表にしてみる



- 次に分類

- トップダウン、ボトムアップ両方から
  - ボトムアップ …「要求がどう分類できるか？」
- かぶることもある
  - 理解できる識別がされていることが重要
- テストを厚くする要求、厚くしない要求
- バグを見つけない要求、動きを保証したい要求
- テストケースを導くエンジニアリング的テスト要求、テストケースを導かないマネジメント的テスト要求

([http://aster.or.jp/business/contest/doc/2019\\_OPEN\\_V1.0.0.pdf](http://aster.or.jp/business/contest/doc/2019_OPEN_V1.0.0.pdf) より)



- エンジニアリング的テスト要求：  
システムの完成像とテスト対象の途中経過
  - システムの完成像への要求の例
    - システム要求、ソフトウェア要求、機能要求、非機能要求、理想的な使い方、差別化要因、目的機能
  - テスト対象の途中経過に関する情報の例
    - 良さに関する知識
      - テスト対象のアーキテクチャや詳細設計、実装、自信があるところ
    - 悪さに関する知識：バグが多そうなところ
      - 構造上問題が起きそうなところ
      - 前工程までの検証作業（レビューやテスト）が足りなかったり滞ったところ
      - 類似製品や母体系製品の過去バグ、顧客クレームから分析した知識
      - スキルの足りないエンジニアが担当したところ、設計中に不安が感じられたところ
      - 進捗が滞ったりエンジニアが大きく入れ替わったりしたところ

以下より：

[http://aster.or.jp/business/contest/doc/2019\\_OPEN\\_V1.0.0.pdf](http://aster.or.jp/business/contest/doc/2019_OPEN_V1.0.0.pdf)





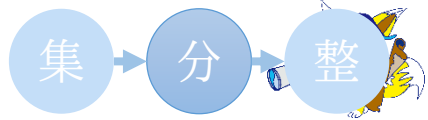
- マネジメント的テスト要求：品質リスクなどに反映させる
  - 工数、人数、スキル分布、作業場所、オフショアか否か、契約形態など
  - 機材利用可否（シミュレータや試作機など）、ツール利用可否（ツールの種類とライセンス、保有スキル）など
  - 目標残存バグ数、信頼度成長曲線など
  - テストスイートの派生可能性や保守性など

以下より：

[http://aster.or.jp/business/contest/doc/2019\\_OPEN\\_V1.0.0.pdf](http://aster.or.jp/business/contest/doc/2019_OPEN_V1.0.0.pdf)



# テスト対象分析結果を分けるとき



- まず仕様書の内容を分解する
  - 1箇所に書かれていても分解した方が分かりやすくなる場合がある
  - 散らばっていても集約した方が分かりやすくなる場合がある
    - 複数の章やファイルに散らばっていることもある
- 分類する、仕様書の行間を埋める
  - 手法
    - 3色ボールペン法
    - 6W2H



# テスト対象分析結果の識別

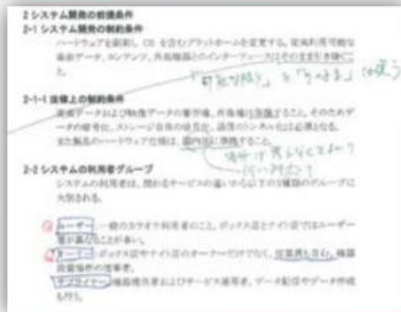


## • 3色ボールペン法

- 仕様書に意味の違う色をつけていき識別する
  - 赤：とても重要
  - 青：まあ重要
  - 緑：疑問やおかしい点
- 赤ばかりにになりがちなので自分たちで意味付けすることも大事

## • 拡張した例

テストベースを3色ボールペンでチェック



**！ポイント**

目的、チームメンバーの状況を考慮し、**3色ボールペンの各色を独自に定義！**

**赤：仕様不備**

**青：観点やテスト条件**

**緑：不明点、気になった点**

テスト設計コンテスト'16 SASADAN Go

<http://aster.or.jp/business/contest/contest2016.html>



# テスト対象分析の識別



## • 6W2H

- When, Where, Who, What, Why, Whom, How, How Much をガイドワードに網羅する

### ◆ 成果物



テスト設計コンテスト'13 Shelly

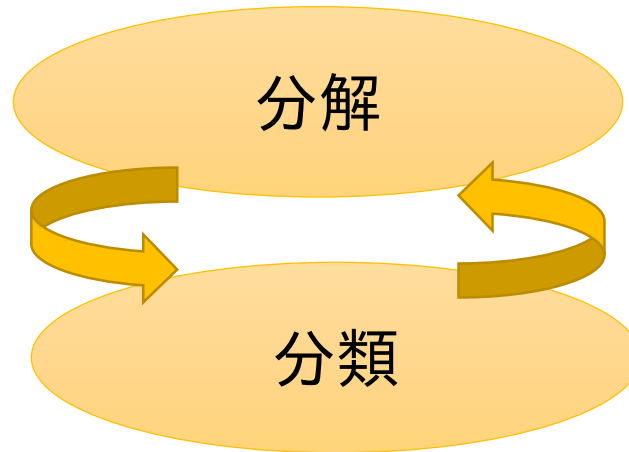
<http://aster.or.jp/business/contest/contest2013.html>



# 分解と分類



分解と分類を行き来しながら、  
集めた要求を欲しいアウトプットに近づける



- アウトプットをイメージし、  
どんな切り口の情報が必要か考えながら行う
  - テスト設計などのインプットとして  
欲しい情報は何か？



# 整理のやり方

集

分

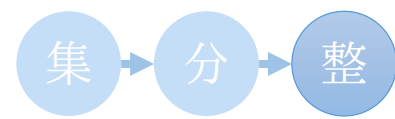
整



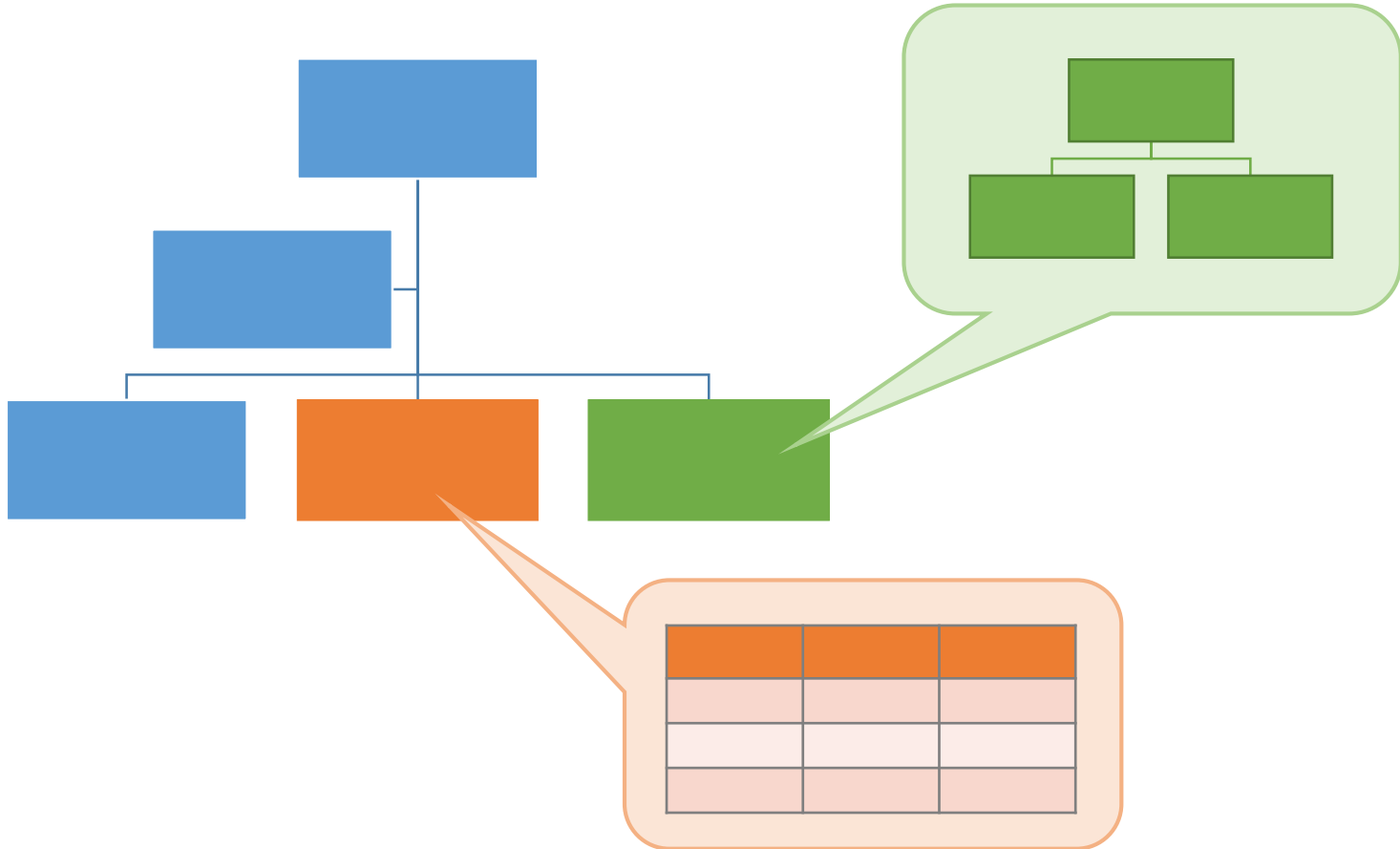
- 要求の全体像が見通せるようにする
  - 要求同士がどう分類されたか明確か？見直す
  - 要求同士の関係性を明確にする
    - 親子、相反、順序
  - トップダウン、ボトムアップ両面から繰り返しておこなっていく
  
- 表記は何でもよい
  - 図
    - マインドマップ
  - 表
    - USDM(Universal Specification Describing Manner)
      - 「派生開発の変更を含む要求仕様の表現方法」  
([http://swquality.jp/temp/nagasakiqdg15\\_usdm.pdf](http://swquality.jp/temp/nagasakiqdg15_usdm.pdf) より)



# 整理の重要ポイント



- 全体を俯瞰できることが大切
  - 図と表などを組み合わせてもよい





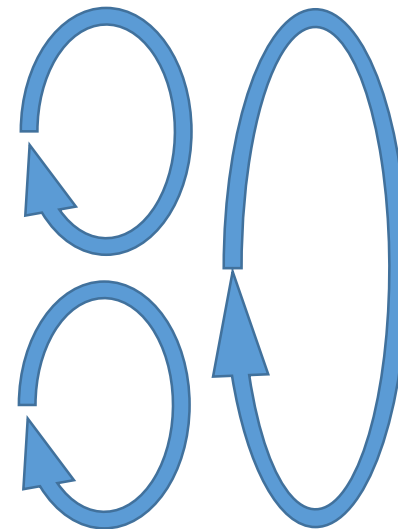
集める(準備含む)



分ける(分解、分類)



整理する(構造化)







# テスト要求分析のプロセス



- テストアーキテクチャ、  
テストケースなどの成果物が、  
テスト要求分析結果をもとに作成される
- 後に繋げやすくなるように、  
テスト要求分析に取り組む
  - 繋がっていることが  
ステークホルダーに説明しやすくなるように  
取り組む



# 持ち帰っていただきたいこと



なぜテスト要求分析を行うか？というところ…

- テスト要求分析の結果は  
テストアーキテクチャ設計やテスト詳細設計の  
品質を左右する
  - 低品質のテスト要求分析結果は、  
テストアーキテクチャやテストケースの品質低下  
を招く

また、気をつける必要があることは…

- テスト要求分析結果について  
なぜそう考えたか？説明でき  
ステークホルダーと合意できるようにする
  - テスト要求を多面的に集め、  
根拠をもって分け、整理しないと  
理解されやすく一貫したテストは非常に困難



1. テストの課題

2. テスト要求分析とは

3. テスト要求分析のやり方

4. 過去の応募作の審査で  
気になった点に対して