

# モデルベースドテストによるテスト設計の自動化

組込みの事例を添えて

須原秀敏

1



## 自己紹介

⌘ 株式会社ベリサーブ所属

⌘ JTC1/SC7/WG26国内小委員会幹事

⌘ JSTQB技術委員

⌘ MBTは2020年ごろからR&Dとして本格的に取り組み開始

⌘ にしさんに助けられながら

⌘ 2017年ごろからテスト観点収集


2

# 本日お伝えしたいこと

- ⌘ モデルはこわくない
- ⌘ MBTは、普段のテスト作りをことばにすることで実現できる
- ⌘ MBTを使うことで、テスト作りの作業が楽になり・説明性が上がり・意思決定の質が上がる
- ⌘ 生成AIの話はしません
- ⌘ ターゲット想定：テスト設計はしているがMBTは知らない

3

## 目次

-  ⌘ MBTってどんなもの？
- ⌘ なんでMBTなの？
- ⌘ MBTのお作法
- ⌘ なんでMBTなの？（再）
- ⌘ （JaSST東海寄りの）事例
- ⌘ MBTに取り組むための第一歩

4

# JaSSTにおけるMBTの歴史

## 20年弱前からJaSSTに登場

- 2007年JaSST東京
- 2012年JaSST四国
- 2023年JaSST東京

## 今日は、JaSSTの歴史に敬意を表しつつ、引用します

- 『モデルベースドテスト入門』(西, 2007)  
『Model-Based Testing入門』(高木, 2012)  
『MBTで目指すこれからのテストづくり』(須原他, 2023)



# モデルベースドテスト(MBT)の定義

- JSTQB: “モデルに基づく、またはモデルを活用するテスト”[1]
- ISO/IEC/IEEE DIS 29119-8: “methodology to automatically generate and execute test cases from models of expected behaviour”[2]
- “テスト設計モデルを用いてテストケースを設計する技術の総称”[3]

[1][https://glossary.istqb.org/ja\\_JP/term/model-based-testing](https://glossary.istqb.org/ja_JP/term/model-based-testing)

[2]<https://www.iso.org/obp/ui/en/#iso:std:iso-iec-ieee:29119:-8:dis:ed-1:v1:en>

[3]『モデルベースドテスト入門』(西, 2007)

# つまり、テスト設計技法です

Ⓐ 本発表では「MBTモデルを用いてテストケースを設計する技術」を採用

- 本ワークショップにおけるMBTの定義：  
テスト対象ソフトウェアやその運用環境の期待される振舞いを形式的モデルによって抽象化し，これに基づいてテストケースを設計，実行する方法の総称．モデルの表記法，テスト基準，テストケース設計法などからなる．特にシステムレベルのテストにおいて効果が期待できる．

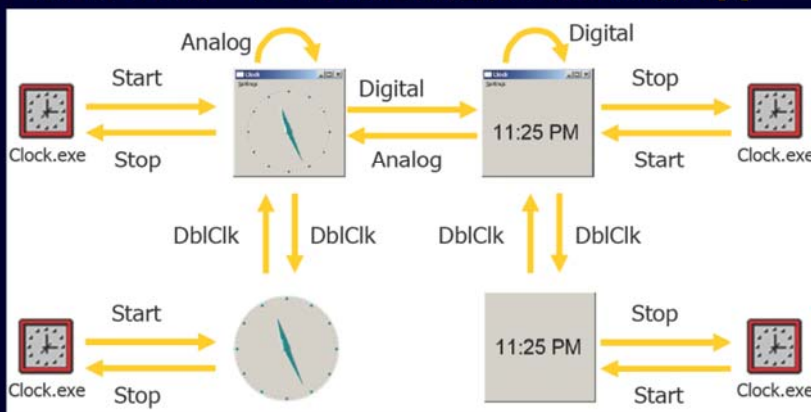
『Model-Based Testing入門』（高木, 2012）

7

## 例えば、MBTはこんなもの

### モデルを用いたテスト設計の例

- GUIアプリケーションの画面遷移テストの場合[1]



『モデルベースドテスト入門』（西, 2007）

8



# 例えば、MBTはこんなモデルを使う

## MBTで用いるモデルの種類 ([2]を修正)

- **フローモデル**
  - 制御フローグラフやデータフローダイアグラムなど
  - プログラムロジックや設計ロジックのテストなどに用いられる
- **状態遷移モデル(FSM)**
  - ステートチャートや状態遷移図、状態遷移マトリクスなど
  - GUI系のテストやリアクティブシステムのテストなどに用いられる
- **トレースベースドモデル**
  - シーケンス図やコラボレーション図
  - 通信系のテストなどに用いられる
- **組み合わせモデル**
  - 直交配列表やAll-pairなど
  - 組み合わせのテストに用いられる
- **統計モデル**
  - ユーザプロファイルモデルなど
    - » ユーザの操作頻度を確率として推測しテストを設計する
- **代数的モデル**
  - プロセス代数、ペトリネットなど
- **形式モデル**
  - 事前条件・事後条件モデル
  - Spec#, JML, B, UML with OCL, Z など

## こんなのもMBTです

### ⌘ EARS記法に従った自然言語記述

⌘ “<状態> の間、<システム> は <応答> しなければならない。”

⌘ ユーザが「タスク追加」ボタンを押したとき、アプリは新しいタスク入力フォームを表示しなければならない。[1]

### ⌘ 作られるテストケース

⌘ ユーザが「タスク追加」ボタンを押したとき、アプリは新しいタスク入力フォームを表示する

⌘ ユーザが「タスク追加」ボタンを押さないとき、アプリは新しいタスク入力フォームを表示しない


[1]<https://qiita.com/NaoikiIshimura/items/490faa689a44c161070f>

# つまりMBTとは

MBTモデル（テスト対象の振る舞いなどをテストケースの生成に**必要な抽象度で抽象化**したもの）をもとに、ルールベース（テストケースづくりにおける一般的な知識や組織・プロジェクトの**経験を形式知化**したもの）でテストケースを機械的に生成する技術

11

## 目次

- ⌘ MBTってどんなもの？
-  ⌘ なんてMBTなの？
- ⌘ MBTのお作法
- ⌘ なんてMBTなの？（再）
- ⌘ （JaSST東海寄りの）事例
- ⌘ MBTに取り組むための第一歩

12

# なぜMBTなのか？

## ⌘ 須原の原体験

- ⌘ 十数年前刈谷にて、現場のテスト設計担当者
- ⌘ テスト設計の方針が自然言語で定義されており、ある程度決まったルールにてテスト設計が行われていた
- ⌘ 日々テストベースを読み解き、テストケースを作る。「比較的単純な作業だなあ」と思い、エクセルマクロで機械化を図る
- ⌘ つまり、テスト設計はルールベースにできる、と気づいた
- ⌘ テスト設計技法の存在理由そのもの

13

# そこから月日が経ち、改めてMBTのメリットを考えた

- ⌘ 現場エンジニアにとって…
  - ⌘ 作業が楽（手間、質）
- ⌘ テストマネージャーにとって…
  - ⌘ 説明が楽
  - ⌘ 改善が楽
- ⌘ マネジメント層にとって…
  - ⌘ 判断が楽

14

# 目次

- ・ MBTってどんなもの？
- ・ なんてMBTなの？
- 👉 ・ MBTのお作法
- ・ なんてMBTなの？（再）
- ・ （JaSST東海寄りの）事例
- ・ MBTに取り組むための第一歩

15

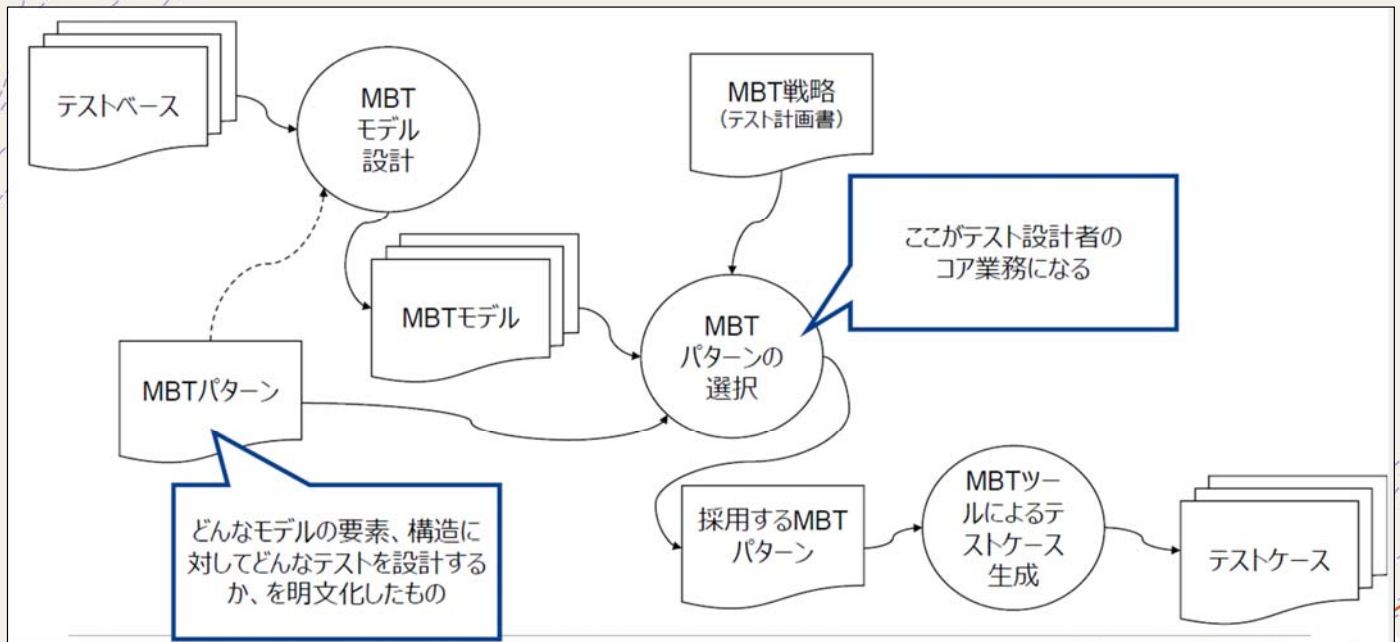
# MBTのプロセス

- ・ MBTモデリング活動
  - ・ MBTアプローチと選択基準に基づいたテストウェアの生成
- 『モデルベースドテスト担当者(CT-MBT)シラバスV1.1.J01』（JSTQB）

16



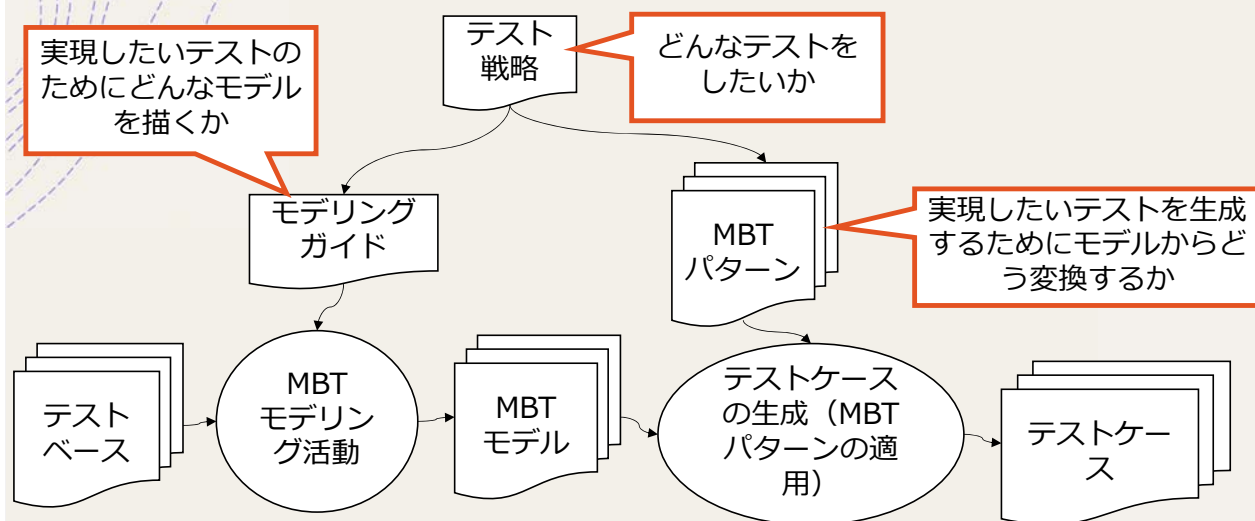
# MBTのプロセス



『MBTで目指すこれからのテストづくり』(須原他, 2023)

17

# MBTのプロセス



18<sup>18</sup>

# MBTパターン

・ MBTモデルの構造や特性を基にテストケースを生成するロジックのこと  
≡テスト設計技法

・ MBTパターンは汎用的なテスト条件とも紐づけられる

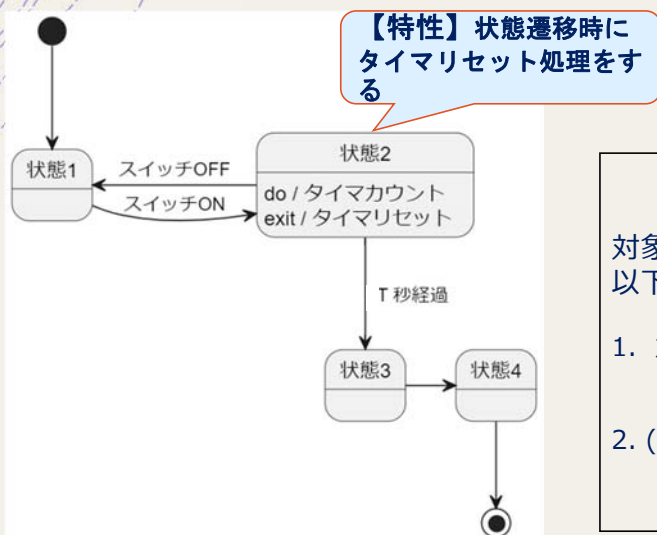
・ 例

- ・ 同値分割法のモデル構造：ある特定のパーティションのすべての要素がテスト対象によって同等に処理されることを想定して、データをパーティションに分割する[1]
- ・ 同値分割法のテストケース生成ロジック：イーチチョイスカバレッジは、テストケースが各パーティションセットから各パーティションを少なくとも1回は通す[1]

[1] 『Foundation LevelシラバスVersion 2023V4.0.J01』 (JSTQB)

19

# MBTパターン



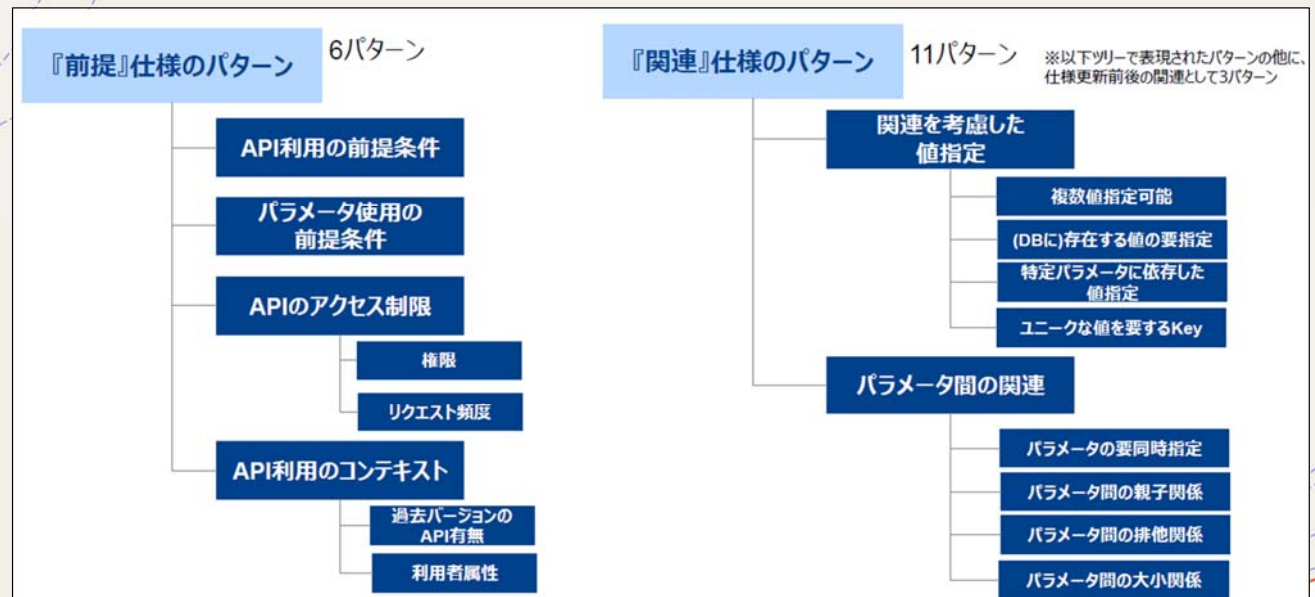
## テストケース生成ロジック

対象状態にタイマリセットがある場合、  
以下を確認するテストケースを生成する

1. 対象状態に遷移後、別状態でT秒経過しても遷移先に遷移しないこと
  - ・ 状態2に遷移→T秒経過前に状態1に戻る→T秒経過しても状態3に遷移しないこと
2. (上記後)対象状態に遷移後、T秒経過したら遷移先に遷移すること
  - ・ 状態2に遷移→T秒経過→状態3に遷移
  - ・ T秒経過前に状態3に遷移しないこと

20

# MBTパターンは構造化できる



『MBTで目指すこれからのテストづくり』(須原他, 2023)

21

## モデリングガイド

⌘ データに対して同値分割法を適用したい

⌘ モデリングガイド：データは、パーティションのすべての要素がテスト対象によって同等に処理されることが想定されるのであれば一つのパーティションとして表現する

⌘ 状態遷移内のタイマリセット処理のMBTパターンを適用したい

⌘ モデリングガイド：状態遷移のStateの処理としてタイマリセット処理が含まれる場合、状態遷移図に特性として記載する

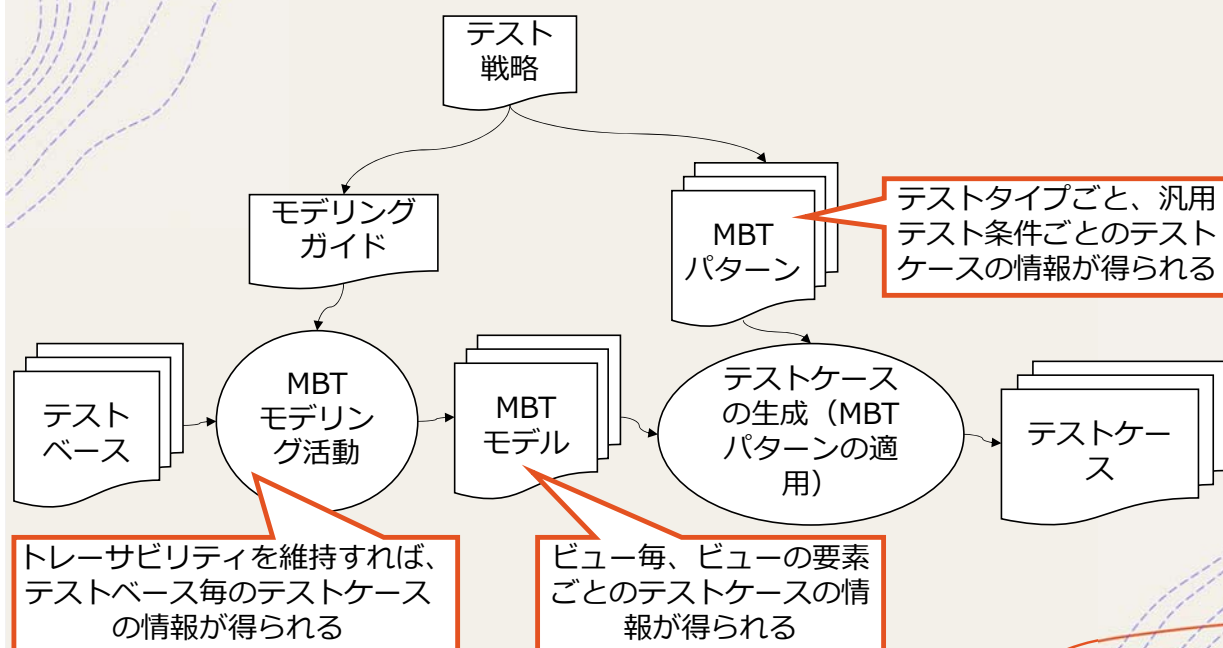
22

# モデリングガイドの例

状態要素	記述ルール	記述例	解説
状態名	状態遷移図内でユニークな名称であること *アクションで変数計算をしないこと	停止中状態、 動作中状態	・ 状態遷移図内で状態名が重複することなくユニークな状態名を記述する
entry処理	-	entry/ モード A = ON	・ 状態に遷移した時に1度だけ実行される処理を記述する ・ 処理が複数ある場合は、カンマ区切りで記述する
do処理	-	do/LED = RED	・ その状態にある間に継続して処理される実行を記述する ・ 処理が複数ある場合は、カンマ区切りで記述する

23

# MBTのプロセス



24<sup>24</sup>

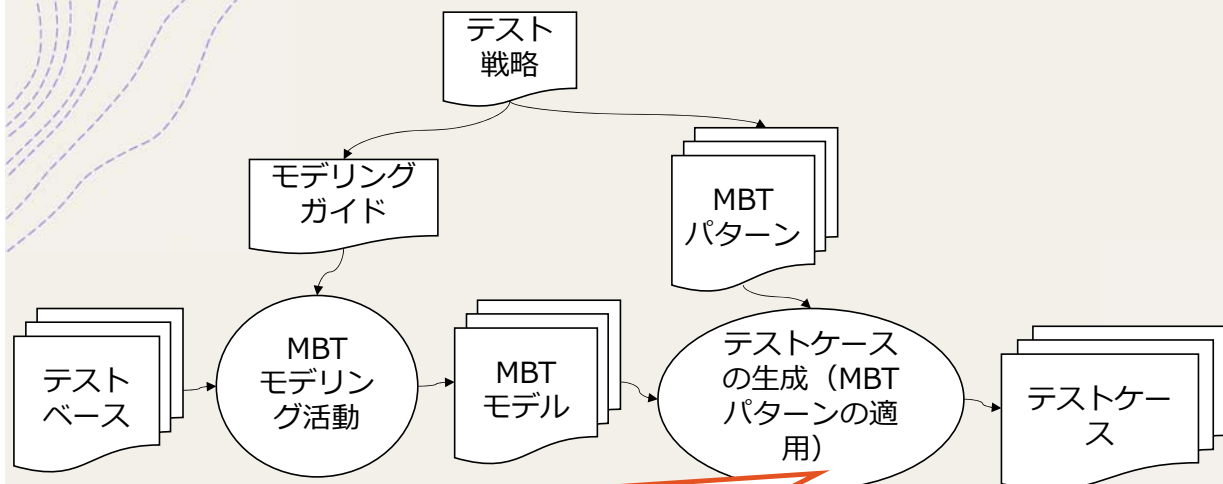


# 目次

- ・MBTってどんなもの？
- ・なんでMBTなの？
- ・MBTのお作法
- ・なんでMBTなの？（再）
- ・（JaSST東海寄りの）事例
- ・MBTに取り組むための第一歩

25

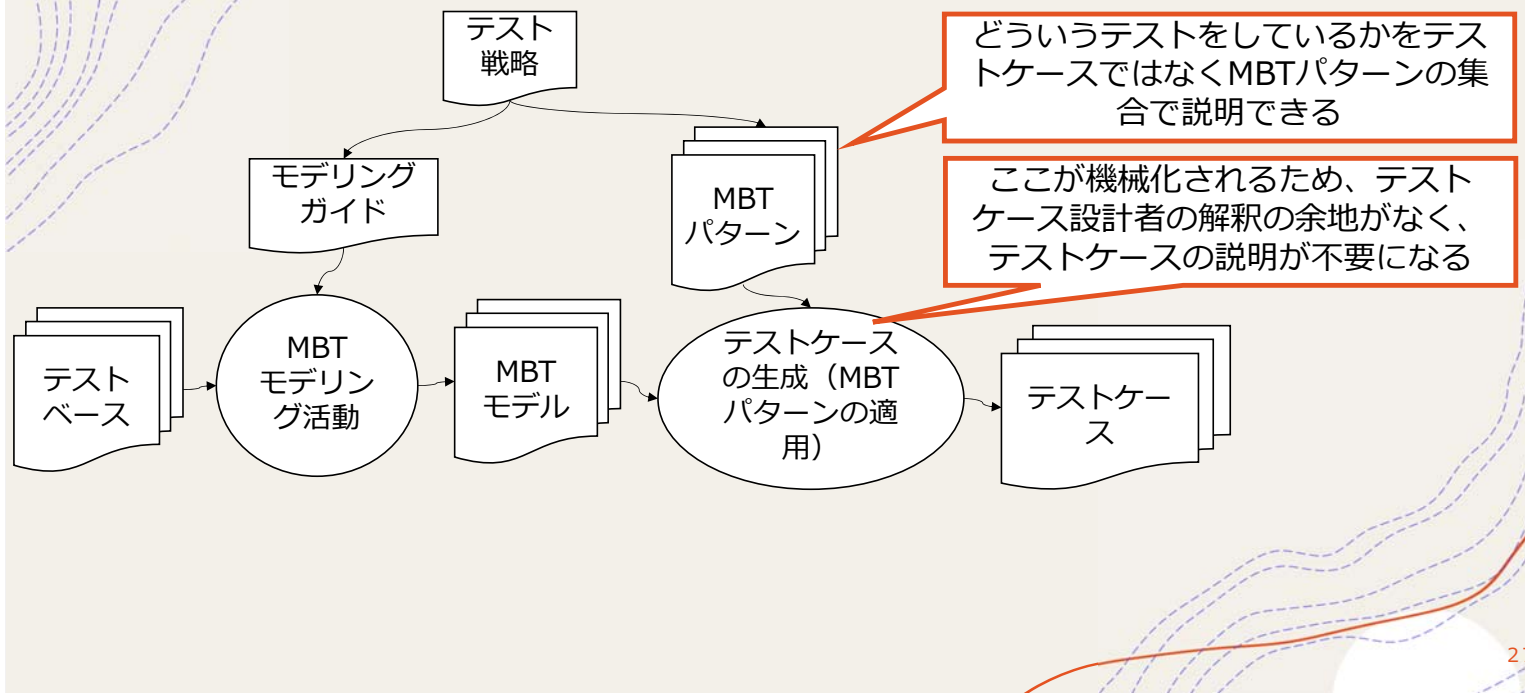
## 現場エンジニアにとって作業が楽



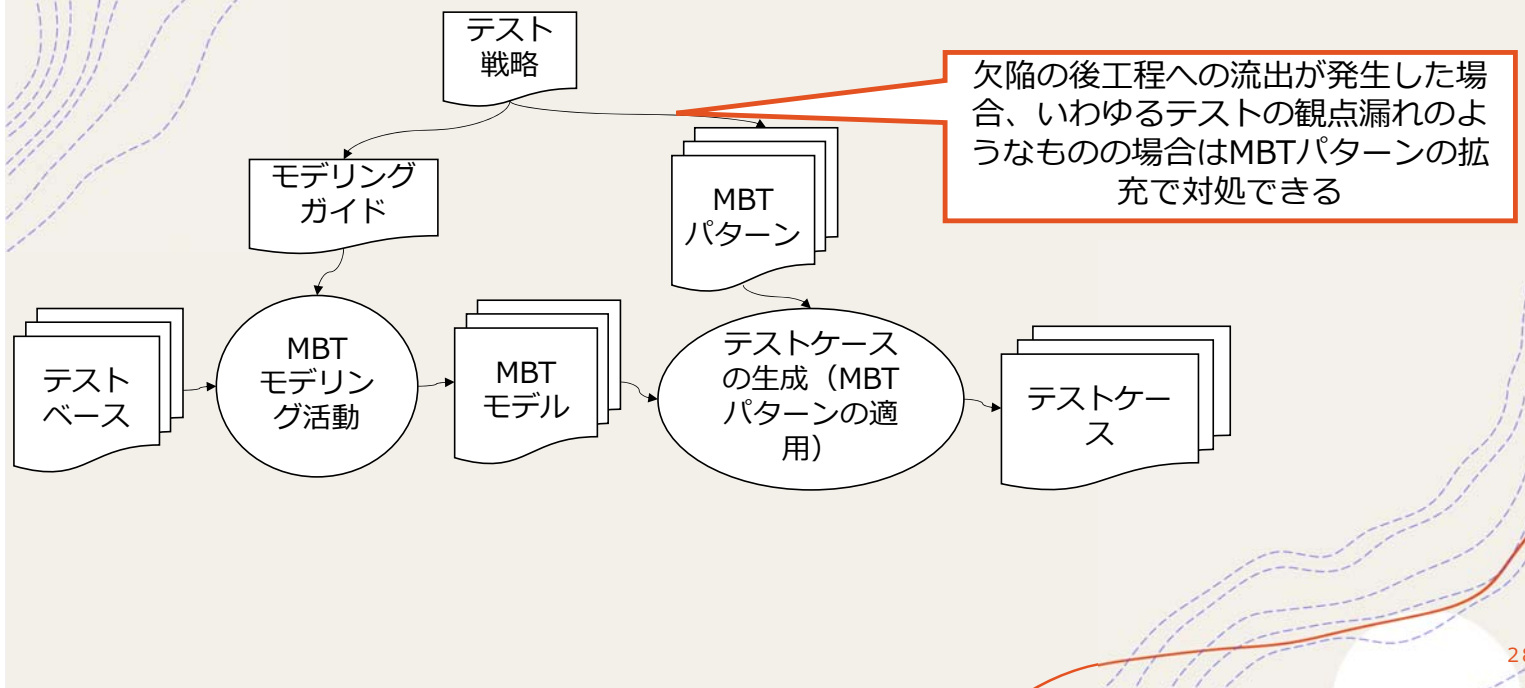
ここが機械化されるため、MBTモデルさえ作ればよく、テストケースの作成コストが0に近くなる。また、更新時もMBTモデルの更新に集中できる。さらに、レビュー時もテストケースではなく、MBTモデル（とMBTパターン）のレビューに集中できる

26

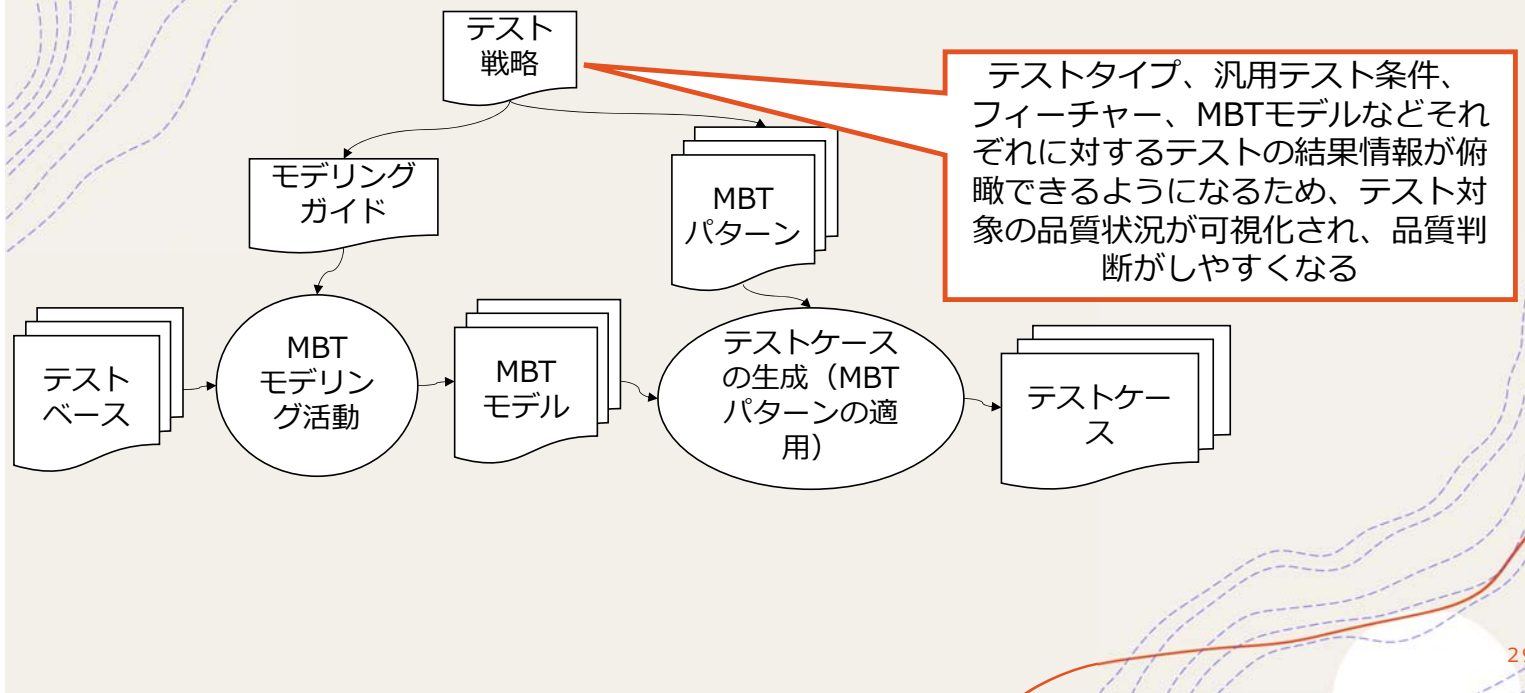
# テストマネージャーにとって説明が楽



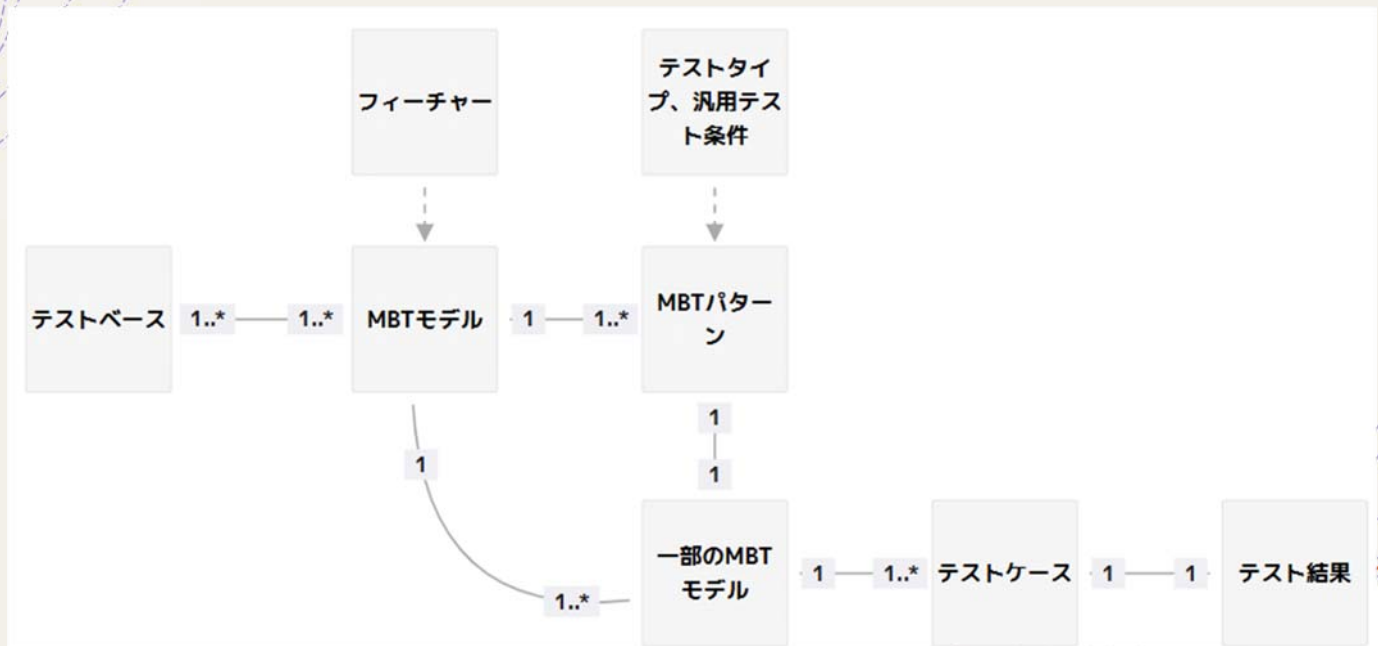
# テストマネージャーにとって改善が楽



# マネジメント層にとって判断が楽

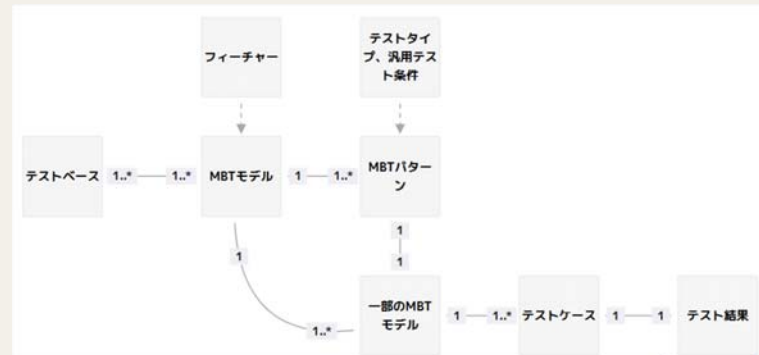


# MBTにおける各データ



# マネジメント層の関心事

- ・フィーチャー毎のテストのカバレッジやPass率が知りたい
- ・テストタイプ「信頼性」に関するテストの量が知りたい
- ・欠陥率が高いMBTパターングループ「アクセス制限」についての追加テストを検討したい



31

## 目次

- ・MBTってどんなもの？
- ・なんでMBTなの？
- ・MBTのお作法
- ・なんでMBTなの？（再）
- ・（JaSST東海寄りの）事例
- ・MBTに取り組むための第一歩



32

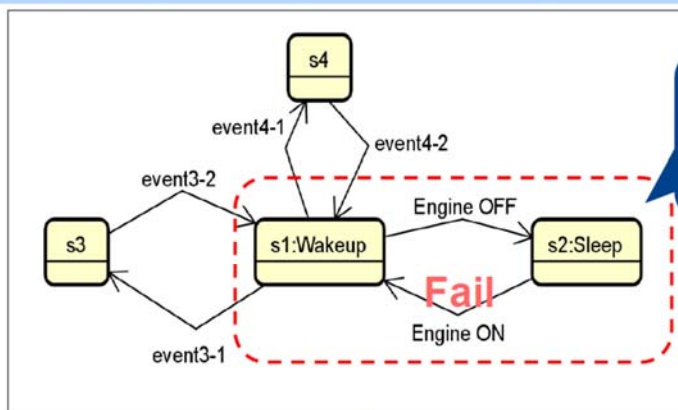


# 事例

- 1つの状態遷移モデルから、各状態に遷移できることを確認するテストケースを生成したが、以下の不具合はそのテストケースからは検出できなかった

## Summary

“Sleep” から“Wakeup”への状態遷移に失敗することがある



遷移時に  
**不揮発性メモリ**  
に読み書き処理をする

MBTモデル: 状態遷移モデル  
(SUT: Automotive system)

\* SUT : System under test

『MBTで目指すこれからのテストづくり』(須原他, 2023)

33

# 事例

- 不揮発性メモリの信頼性の問題でデータが欠損した
- データ読み込み処理に失敗し遷移不可になった

s1とs2の遷移を複数回確認するテストケースが無かった

## 全ての遷移パスを網羅

#	from	to
1	s1:Wakeup	s2:Sleep
2	s2:Sleep	s1:Wakeup
3	s1:Wakeup	s4
...	...	...

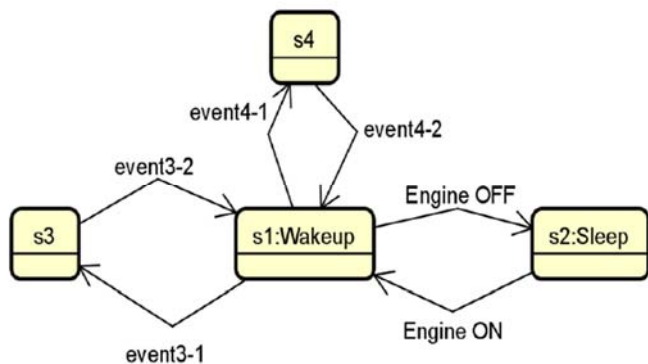
不具合の原因

MBTモデルから作られたテストケース

『MBTで目指すこれからのテストづくり』(須原他, 2023)

34

# 事例



状態遷移モデル

統合

(不揮発性メモリに由来する)  
エラー確率モデル

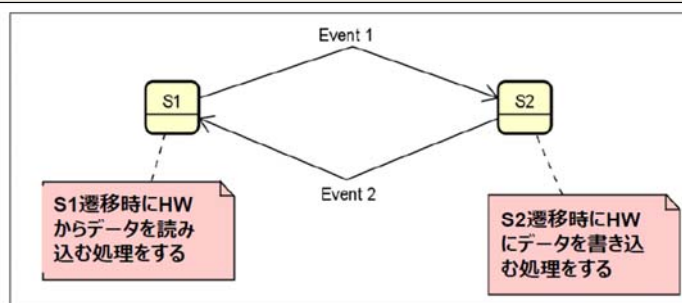
テストケース生成

"Sleep"と"Wakeup"の状態遷移を確率モデルを考慮して複数回試行する

『MBTで目指すこれからのテストづくり』(須原他, 2023)

35

# 事例



もし、状態遷移モデルが上記の特性を持っているなら、以下のテストケースを生成する

テストステップ	期待動作	試行回数
S1 -> S2 -> S1	S1に遷移し、処理が開始されること	N (>1)

Nは、確率モデルのPの値に基づいて決定する

『MBTで目指すこれからのテストづくり』(須原他, 2023)

36

# 目次

- ⌘ MBTってどんなもの？
- ⌘ なんでMBTなの？
- ⌘ MBTのお作法
- ⌘ なんでMBTなの？（再）
- ⌘ （JaSST東海寄りの）事例
- 👉 ⌘ MBTに取り組むための第一歩

37

## どこから始めるか（ペインポイント）

- ⌘ 現場目線
  - ⌘ 単純作業が多い、メンテがきつい
- ⌘ テストマネージャー目線
  - ⌘ レビューきつい、改善回したい
- ⌘ マネジメント目線
  - ⌘ 自信をもって判断したい

38

# 本日お伝えしたかったこと

- ・モデルはこわくない
- ・MBTは、普段のテスト作りをことばにすることで実現できる
- ・MBTを使うことで、テスト作りの作業が楽になり・説明性が上がり・意思決定の質が上がる

ありがとうございました！