

AUTOSAR OSに対するテスト ケースおよびテストプログラムの 自動生成

2012年1月25日

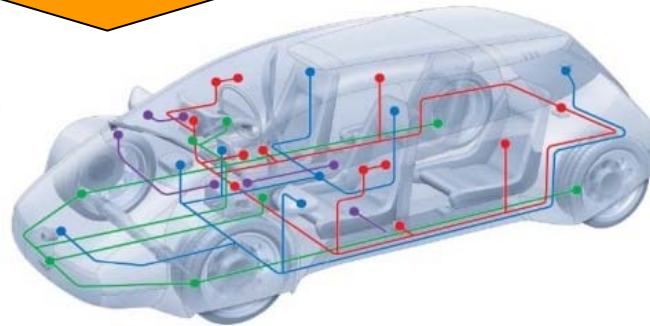
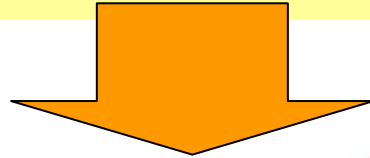
日本電気通信システム株式会社
風間 佳之

アジェンダ

1. はじめに
2. 既存のテストスイートの概要
3. テストスイート開発における課題
4. 組み合わせテストツールの導入検討
5. 組み合わせテストツールによる課題解決
6. 実施結果と評価
7. まとめ

背景

- ECU(Electronic Control Unit)数の増加
- ソフトウェア複雑性の増加
- 車両内通信の増加
- 省エネルギー化と安全性に対する要求の増加



車載ソフトウェア開発規模が増加...

HWに依存しない非競争領域を共通化したプラットフォームにする事により, 問題を解決する

AUTOSARとは?

AUTomotive Open System ARchitecture

車両ソフトウェアにおけるオープン
システムアーキテクチャ

AUTOSARとは、大規模化する自動車用ソフトウェア開発における様々な課題を解決するために生まれ、標準化されたアーキテクチャ

AUTOSAR Consortium

OEM

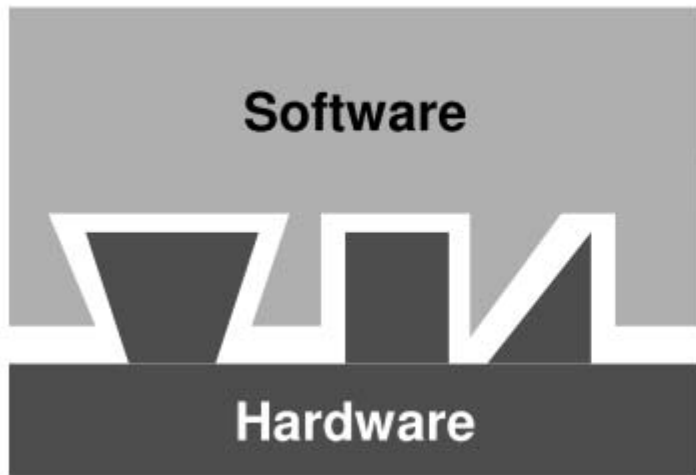
Tier 1
Supplier

Tool vender

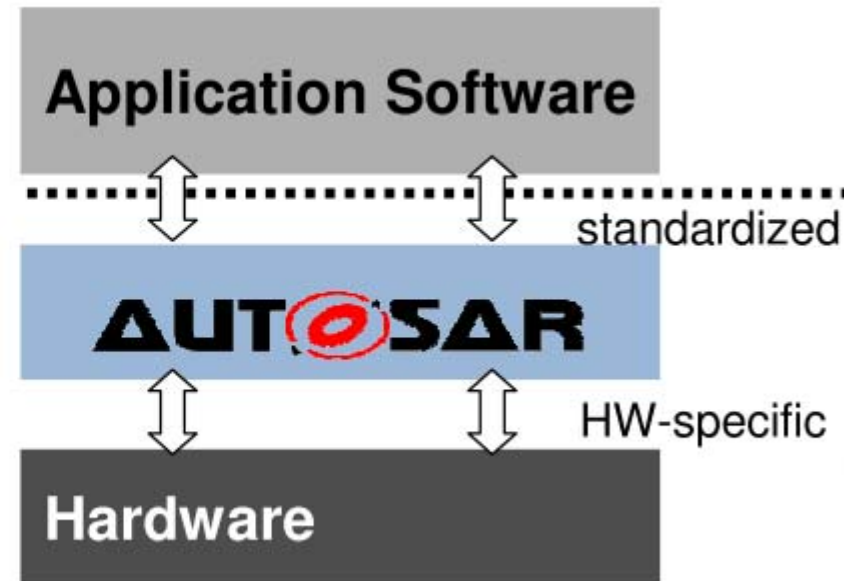
Software
developer

AUTOSARのイメージ

Non- AUTOSAR



AUTOSAR



研究の紹介

純国産AUTOSAR OS

TOPPERS/ATK2 (Automotive Kernel 2)

のコンソーシアム型共同研究/開発

具体的な活動

- ・AUTOSAR OS仕様を明確化, 修正
→自由に使える日本語版の仕様書の作成
- ・オープンソース実装
- ・**検証スイート開発**



AUTOSAR OSの日本における標準を策定する

コンソーシアム型共同研究



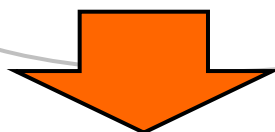
産学連携による技術課題の解決

2011年度コンソーシアム型共同研究参加企業一覧

企業名(50音順)
株式会社ヴィッツ
株式会社 OTSL
株式会社サニー技研
株式会社デンソー
株式会社東芝
トヨタ自動車株式会社
日本電気通信システム株式会社
パナソニック アドバンステクノロジー株式会社
富士ソフト株式会社
富士通 VLSI 株式会社
ルネサスエレクトロニクス株式会社

AUTOSAR OSの検証における背景

過去のコンソーシアム型共同研究で実績のあったテストスイートをAUTOSAR OSの検証に活用したい…



μ ITRON仕様のOSであるTOPPERS/ASP/FMPを対象にしたテストスイートであるTTSP (TOPPERS Test Suite Package)をベースにAKTSP (Automotive Kernel Test Suite Package)を開発

補足: TOPPERS新世代カーネル

- ・ μ ITRON仕様をベースとして, 信頼性, 安全性, ソフトウェアポータビリティを向上させるための改良・拡張
 - シングルプロセッサ向けRTOS: TOPPERS/ASPカーネル
 - マルチプロセッサ向けRTOS : TOPPERS/FMPカーネル



-
1. はじめに
 2. 既存のテストスイートの概要
 3. テストスイート開発における課題
 4. 組み合わせテストツールの導入検討
 5. 組み合わせテストツールによる課題解決
 6. 実施結果と評価
 7. まとめ

システムサービステストの概要

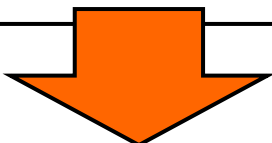
外部仕様書に基づいてシステムサービス発行前後のシステム状態の変化を確認する

ActivateTaskの仕様抜粋

ActivateTask タスクの起動

【機能】

ActivateTask は、TaskID で指定されたタスクの状態を、休止状態から実行可能状態へ遷移させる



仕様の振舞いを確認するテストを実施する

- ・自タスク以外のタスクを指定して呼び出す。
- ・休止状態のタスクを指定する。
- ・対象タスクの優先度が、実行状態のタスクより低い場合、対象タスクが実行可能状態になること。

仕様, 実装の両面からテストを実施

前状態



(実行中)
優先度:高

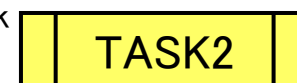


(休止中)
優先度:中

処理



ActivateTask



後状態



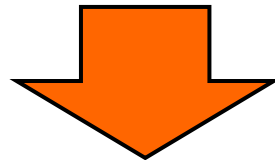
(実行中)



(実行可能)

ハンドコーディングでのテストプログラム開発の問題点

1. テストプログラムの可読性, 保守性の低下
テストケースの実現方法が複数存在するため, 開発担当者が異なることによるばらつき発生
2. テストの開発工数
ASPカーネルのAPIは121個存在し, 抽出したテストケースは1,669件
3. 異なるRTOSへの流用不可
同一テストシナリオに対するRTOS毎の重複開発

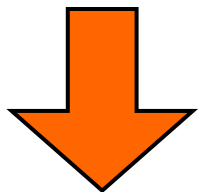


テストプログラムを生成するツールを開発

テストシナリオからのテストプログラム生成

テストケース
(ブラックボックス)

テストケース
(ホワイトボックス)



テストケースを
前状態・処理・後状態
へ具体化する

形式化したテストシナリオ

前状態
<システムサービス発行前のシステム状態>

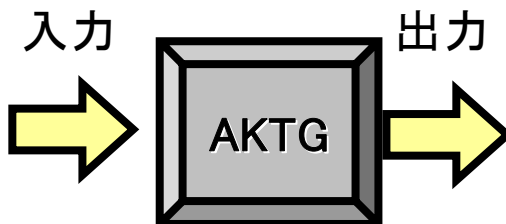
処理
<システムサービス発行処理>

後状態
<システムサービス発行後のシステム状態>



TESRY記法

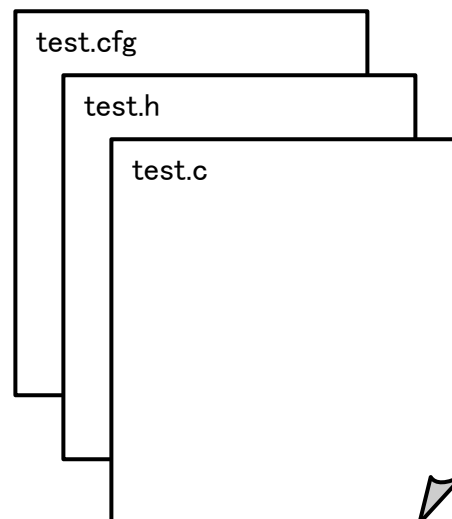
(TEst Scenario for Rtos by Yaml)



(Automotive Kernel Test Generator)

- ・Rubyで開発
- ・約22,000行

テストシナリオを
実現するテストプログラム



AKTGはTTSPで開発したTTG(TOPPERS
Test Generator)をベースに開発する

TESRY記法

階層型データ形式言語であるYAML形式を用いて
全カーネルオブジェクトの属性/状態の記述方法を定めた
(TESRY記法で記述したデータファイルを**TESRYデータ**と呼ぶ)

(例)ActivateTaskのテストケース

前状態

優先度中のTASK1が実行状態
優先度高のTASK2が休止状態

処理

TASK1がActivateTask(TASK2)を
発行してE_OKが返る

後状態

TASK1が実行可能状態となる
TASK2が実行状態となる

pre_condition:

```
TASK1:
  type : TASK
  tskpri : TSK_PRI_MID
  tskstat: running
TASK2:
  type : TASK
  tskpri : TSK_PRI_HIGH
  tskstat: waiting
  wobjid : SLEEP
```

do:

```
id : TASK1
syscall: ActivateTask(TASK2)
ercd : E_OK
```

post_condition:

```
TASK1:
  tskstat: ready
TASK2:
  tskstat: running
```

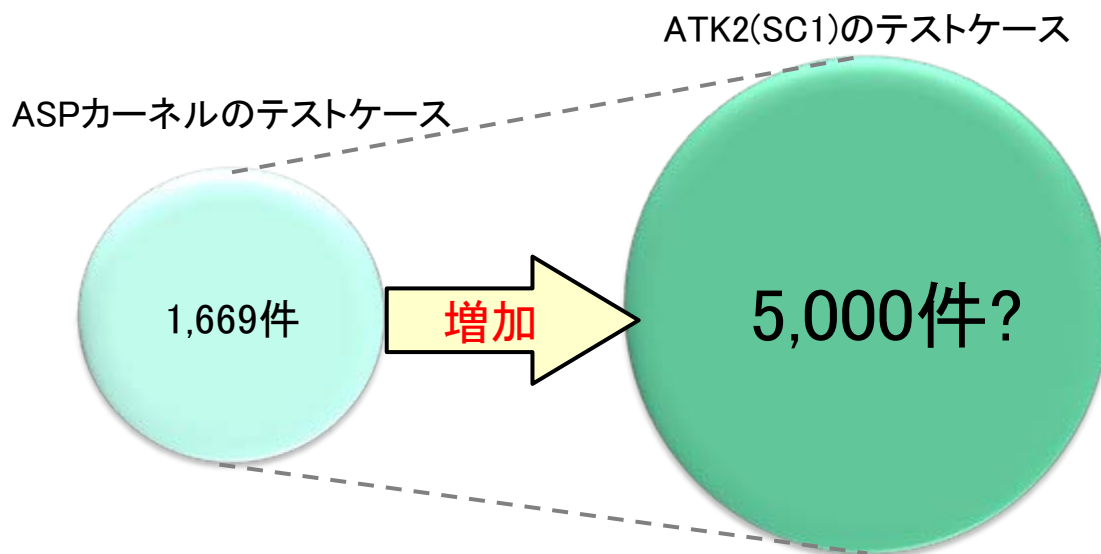
-
1. はじめに
 2. 既存のテストスイートの概要
 3. テストスイート開発における課題
 4. 組み合わせテストツールの導入検討
 5. 組み合わせテストツールによる課題解決
 6. 実施結果と評価
 7. まとめ

課題:カーネル仕様の差異による組み合わせパターン増加

検査対象OSが μ ITRON仕様からAUTOSAR仕様に変更

OSが提供するAPIの数は減少したが…

仕様の複雑化によりテストとして考慮すべき組み合わせパターンが増大
(タスクならば前状態だけでASPの4倍の組み合わせになる)



従来型のテスト開発プロセスで行っていた,
組み合わせテストケースの手動作成が困難に!

テスト手法の適用による効率的な組み合わせテストケース削減の検討

組み合わせ増加に対し、テスト手法による効率的な組み合わせテストケースの削減をすべきでは？

しかし…

組み合わせ削減によりRTOSの品質に与える影響が不明

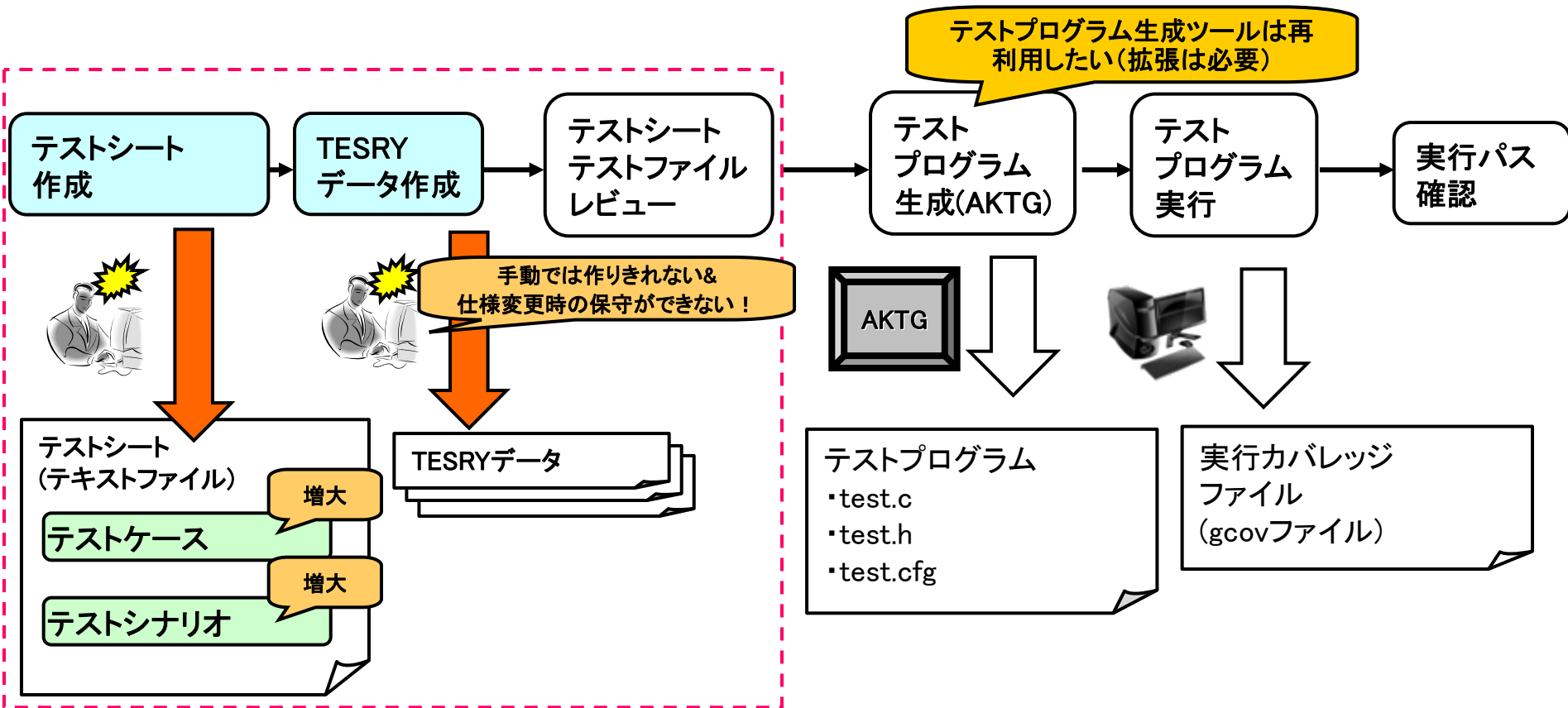
例：

特定の因子の組み合わせのみ網羅に注力して組み合わせ削減
→どこに注力すべきか、データの蓄積が無いのでポリシーが作れない

できることなら、考慮しうる全組み合わせでのテストを実施したい…



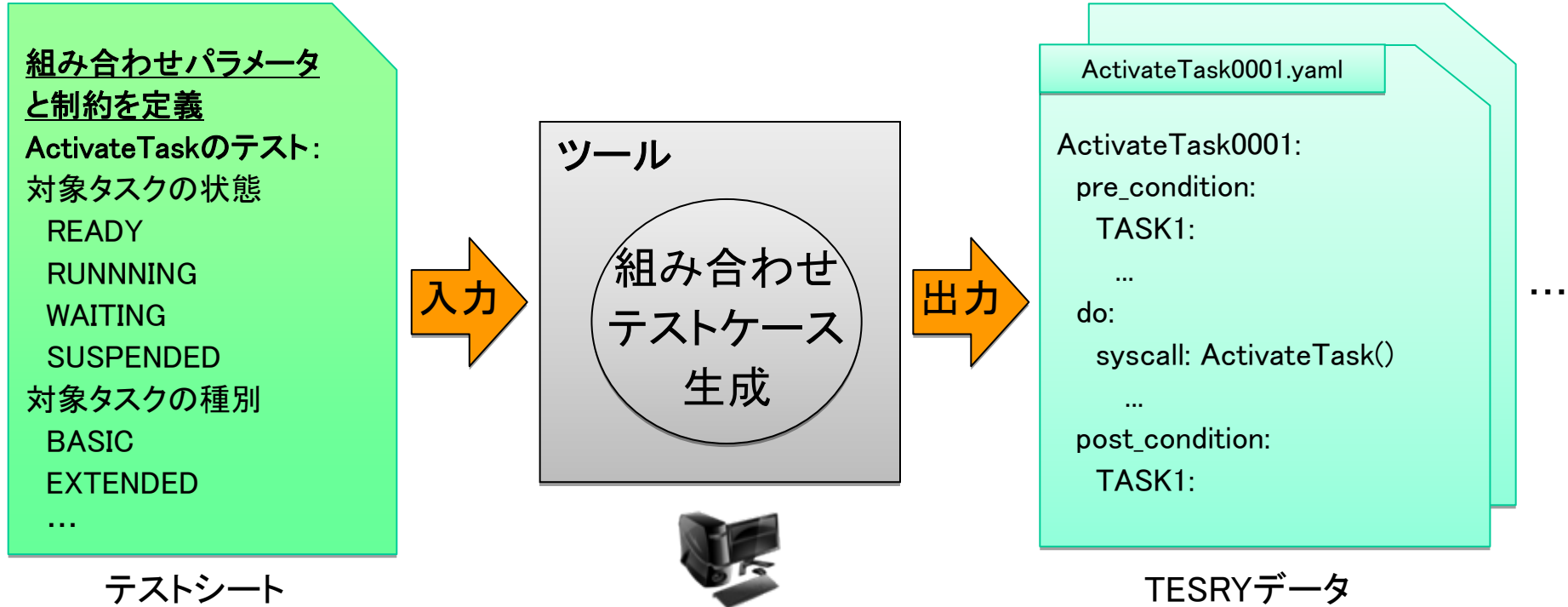
従来型プロセスの課題と改善ポイント



組み合わせテストケース作成+TESRYデータ生成を
自動化すれば, テストプログラム生成, 実行まで自動で実施可能になりそう
→組み合わせテストケース作成ツールの導入へ

-
1. はじめに
 2. 既存のテストスイートの概要
 3. テストスイート開発における課題
 4. 組み合わせテストツールの導入検討
 5. 組み合わせテストツールによる課題解決
 6. 実施結果と評価
 7. まとめ

組み合わせテストツールの導入検討



ツール要件1: 出力フォーマットはYAML形式が望ましい

→ソースコードが公開され、拡張性があること

ツール要件2: 組み合わせに対して制約条件を指定したい

→仕様によって組み合わせできないパラメータを排除できること

組み合わせテストツールの選定

候補

ALLPAIERS

QICT

HEXAWISE

PictMaster

選 定

PictMasterとは？

岩通ソフトシステム社が開発した組み合わせテストケース生成ツール。

PICTというMicrosoft社が無償で公開しているCUIベースの組み合わせテストツールを、VBAを使ってGUI(エクセル)上で扱える。

特に着目した点

- ・ 組み合わせテストがエクセルベースでモデル化されるので扱いが容易
- ・ 日本語マニュアルが整備されておりツールの習得に時間がかからない
- ・ オープンソースになっており、拡張性が高い

PictMasterの入力

Microsoft Excel - PictMaster.xls

ファイル(F) 編集(E) 表示(V) 挿入(I) 書式(O) ツール(T) データ(D) ウィンドウ(W) ヘルプ(H)

MS ゴシック 9 B I U

N67 欠

A B C D E F G H I J K L M N O P Q R S T U V W

1 PictMaster v4.3.2 2011/7/19

2	大項目No.	大項目名			
3	小項目No.	小項目名			
8	パラメータ	値の並び			
9	参加者	一般,小学生			
10	参加申し込み単位	個人,団体			
11	県内在住	YES,NO,-			
12	年齢	6歳未満,6~12歳,13~64歳,65歳以上			
63	制約表				
64	パラメータ	制約1	制約2	制約3	制約4
66	参加者	小学生	一般		
67	参加申し込み単位				
68	県内在住				
69	年齢	6~12歳	#6~12歳		
116	結果表				
117	結果内容	参加者	参加申し込み単位	県内在住	年齢
119	1200円	一般	個人		6~12歳,13~64歳
120	1000円	一般	団体		6~12歳,13~64歳
121	800円	小学生	個人	NO	6~12歳
122	500円	小学生	団体	NO	6~12歳
123	無料				6歳未満,65歳以上
124	無料	小学生		YES	6~12歳
125					

パラメータ, 値入力表

制約表

結果表

組み合わせテストをエクセルにモデル設計

PictMasterの出力

	A	B	C	D	E	F
1	No.	参加者	参加申し込み単位	県内在住	年齢	結果内容
2	1	一般	個人	NO	13~64歳	1200円
3	2	一般	個人	NO	6歳未満	無料
4	3	一般	個人	NO	65歳以上	無料
5	4	一般	個人	YES	6歳未満	無料
6	5	一般	個人	YES	13~64歳	1200円
7	6	一般	個人	YES	65歳以上	無料
8	7	一般	団体	NO	65歳以上	無料
9	8	一般	団体	NO	6歳未満	無料
10	9	一般	団体	NO	13~64歳	1000円
11	10	一般	団体	YES	13~64歳	1000円
12	11	一般	団体	YES	6歳未満	無料
13	12	一般	団体	YES	65歳以上	無料
14	13	小学生	個人	NO	6~12歳	600円
15	14	小学生	個人	YES	6~12歳	無料
16	15	小学生	団体	NO	6~12歳	500円
17	16	小学生	団体	YES	6~12歳	無料
18						

組み合わせパラメータと期待結果が**形式化されて**出力される

-
1. はじめに
 2. 既存のテストスイートの概要
 3. テストスイート開発における課題
 4. 組み合わせテストツールの導入検討
 5. 組み合わせテストツールによる課題解決
 6. 実施結果と評価
 7. まとめ

PictMasterによる組み合わせテストケース, TESRYデータの自動生成

TESRYデータ

赤字: 組み合わせ対象パラメータ

pre_condition:

TASK1:

type: TASK

tstat: RUNNING

spolicy: AK_SCHFULL

TASK2:

type: TASK

tstat: SUSPENDED

inipri: TSK_PRI_HIGH

do:

id: TASK1

sysrv:

ActivateTask(TASK2)

rettype: StatusType

retval: E_OK

post_condition:

TASK1:

tstat: READY

TASK2:

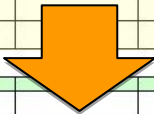
tstat: RUNNING

PictMaster for ATK2						
大項目名		ActivateTask				
小項目名		任意条件テストケース				
パラメータ						
call_origin	TASK_C2ISR	組み合わせ				
other_tstat	READY, SUSPENDED, -					
other_curpri	TSK_PRI_LOW, TSK_PRI_MID, TSK_PRI_HIGH, -					
other_ttype	AK_BASIC, AK_EXTENDED, -					
other_ext_flg	ANY_EVT, NON_EVT, -					
org_spoliory	AK_SCHFULL, AK_SCHMON, AK_SCHFULL/GROUP_TASK					
other_sotont	0, 1, ANY_ACTION, NOT_MAX_I, -	ANY_BASIC, MAXCNT				
param_task_id	ORG_TASK, OTHER_TASK					
os_intstat	DIS_ALL_INT, SUS_ALL_INT, SUS_OS_INT					
org_ttype	AK_BASIC, -	AK_EXTENDED				
org_sotont	1, ANY_ACTION, NOT_MAX_I, -					
other_spoliory	AK_SCHFULL, AK_SCHMON, AK_SCHFULL/GROUP_TASK, NON_GROUP_TASK					
制約表						
パラメータ	変更	制約1	制約2	制約3	制約4	制約5
call_origin						
other_tstat	READY	READY		READY	SUSPENDED	READY
other_curpri		TSK_PRI_HIGH				
other_ttype	AK_EXTENDED		AK_BASIC	AK_EXTENDED		
other_ext_flg			NON_EVT			
org_spoliory	AK_SCHFULL					
other_sotont				1	0	80
param_task_id						ORG_TASK
os_intstat						\$-
org_ttype						\$-
org_sotont						\$-
other_spoliory						
TESRY生成表						
TESRY_param	call_origin	other_tstat	other_ttype	other_sotont	other_spoliory	other_ext_flg
\$COPC2	TASK_C2ISR					
\$E01	TASK_C2ISR					
\$E02			AK_BASIC			0
\$E03			AK_BASIC			0, 1, ANY_ACTION
\$E04						1
\$E05						0
\$E06						0, 1, ANY_ACTION
\$E07						1
\$E08	TASK_C2ISR					
\$E09	TASK_C2ISR					
\$E10	TASK_C2ISR					
\$E11	TASK_C2ISR					
\$E12	TASK_C2ISR					
\$E13	TASK_C2ISR					
\$E14	TASK_C2ISR					
\$E15	TASK_C2ISR					
\$E16	TASK_C2ISR					
\$E17	TASK_C2ISR					
\$E18	TASK_C2ISR					
\$E19	TASK_C2ISR					
\$E20	TASK_C2ISR					

パラメータ, 値入力表

制約表

結果表



PictMasterによる組み合わせテストケース, TESRYデータの自動生成

外部仕様書

テスト設計

テストシート

入力

PictMaster

組み合わせ
テストケース
+TESRY生成

出力

VBA修正量を考え, 出力形式は従来型のCSV形式のままにした

TESRYデータ
(CSV形式)

入力

Rubyスクリプト
(160行程度)

CSV形式
→YAML形式
変換

出力

ActivateTask.yaml

```
ActivateTask0001:  
  pre_condition:  
    TASK1:  
    ...  
  do:  
    syscall: ActivateTask()  
    ...  
  post_condition:  
    TASK1:
```

TESRYデータ

行毎に出力される組み合わせパターンに付随してCSV形式のTESRYデータを出力するように, PictMasterのVBAを60行程度修正, 拡張した

PictMasterとRubyスクリプトを連携させることで,
TESRYデータを自動生成

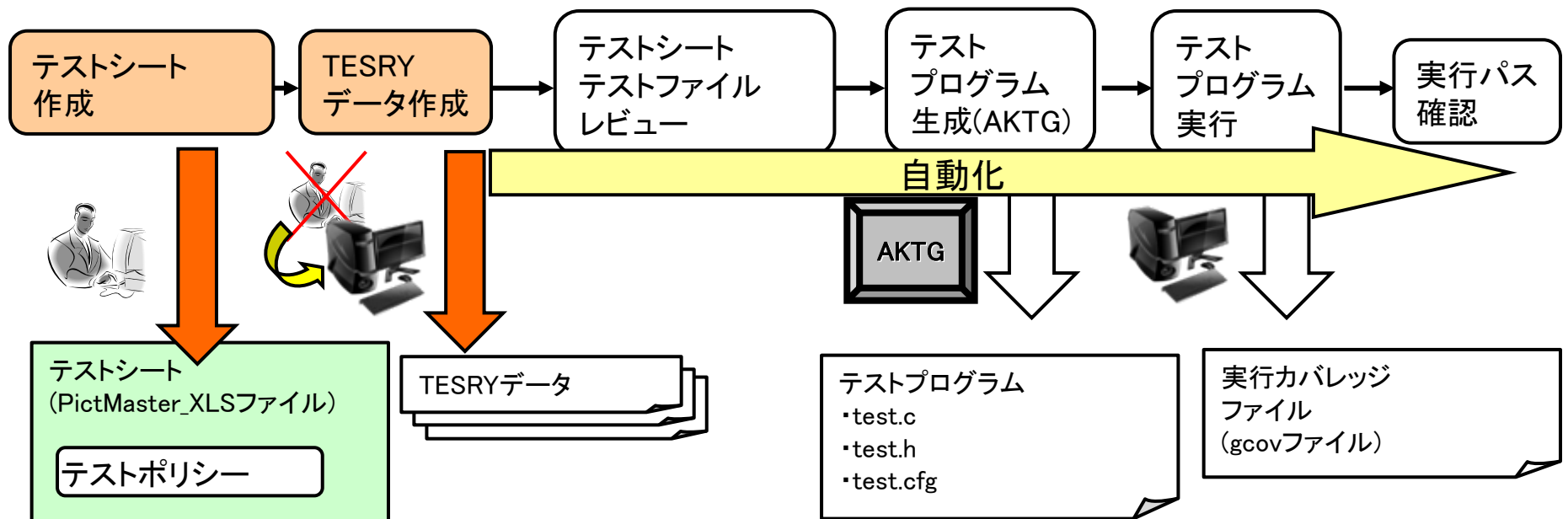
-
1. はじめに
 2. 既存のテストスイートの概要
 3. テストスイート開発における課題
 4. 組み合わせテストツールの導入検討
 5. 組み合わせテストツールによる課題解決
 6. 実施結果と評価
 7. まとめ

実施結果

- 2011/12月末時点で6,107件のTESRYデータを自動生成し、テストを実施した。
 - ・ TESRYデータ総行数 : 20万行以上
 - ・ PictMaster改造量 : 約50行
 - ・ Rubyスクリプト作成量 : 約160行
- 発見したバグ

バグ種別	発見件数
仕様バグ	8件
実装バグ	7件

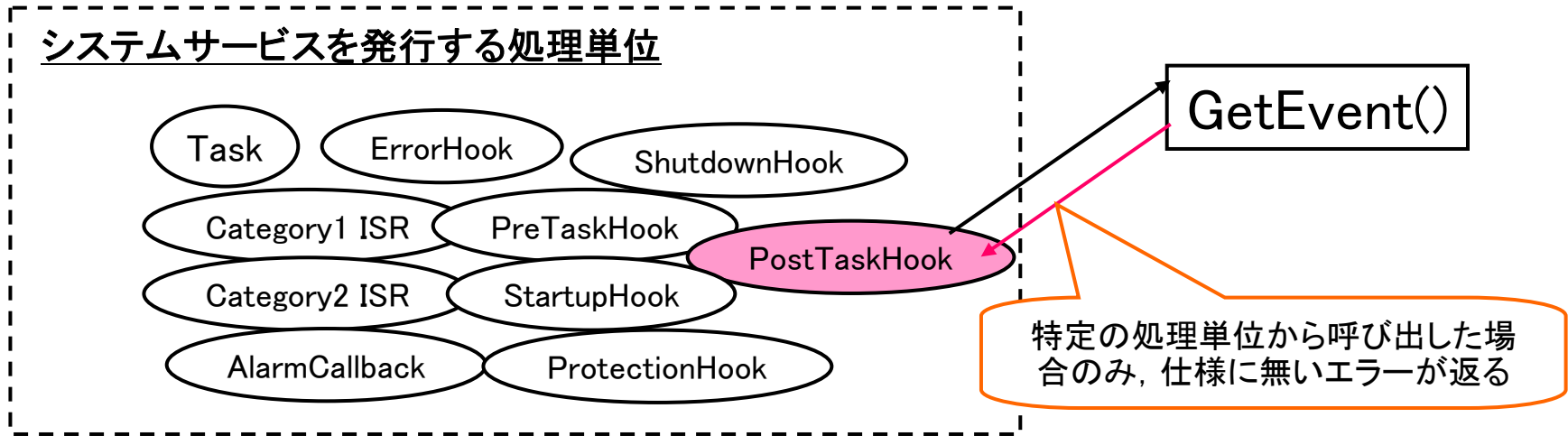
評価 ～テストプロセス改善結果～



得られた効果

- ・ **生産性が向上**
 - TESRYデータを手動→自動生成
 - テスト設計における組み合わせ漏れレビューが容易に
- ・ **保守性が向上**
 - 仕様変更に対する保守性が向上した
 - テストケースの属人性を排除

評価 ～実装バグ分析～

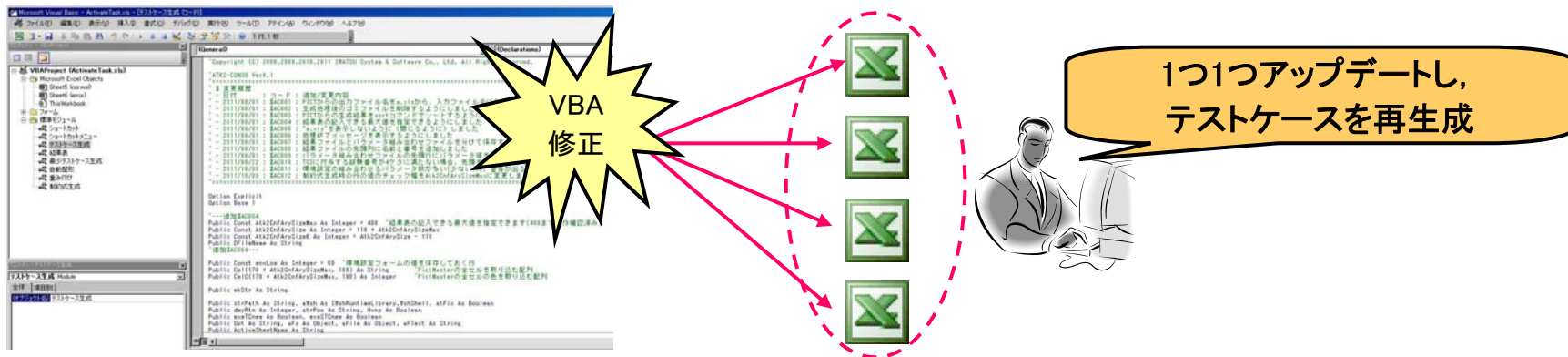


複数ある処理単位のテストなど、
手動ならばポリシーで部分網羅しかしなかった組み合わせを
全網羅することでバグを発見できた

課題

PictMasterの品質維持に関する課題

- ・ ソースコードを直接バージョン管理できない
- ・ ブックが複数ある場合、VBAのアップデートは各ブック毎に実施する必要あり



課題解決の為、新たなツールの開発/導入を検討

-
1. はじめに
 2. 既存のテストスイートの概要
 3. テストスイート開発における課題
 4. 組み合わせテストツールの導入検討
 5. 組み合わせテストツールによる課題解決
 6. 実施結果と評価
 7. まとめ

まとめ

- ・ 名古屋大学を中心としたATK2カーネルの開発/検証に関するコンソーシアム型研究組織は、2009年度に開発したASPカーネルのテストスイート(TTSP)を拡張する形で、2011年度、ATK2のテストスイート(AKTSP)を開発しテストをシステムサービステストを実施した
- ・ PictMasterの導入により、組み合わせテストケースとTESRYデータ(形式言語によるテストスクリプトファイル)を自動生成し、生産性、保守性の改善を実現した
- ・ ATK2の実装、仕様の不具合を合計15件検出した