

バルテス株式会社  
石原 一宏

テストプロ直伝！

長く使い続けられる  
『テスト基盤』の構築方法

# ご挨拶と今日のテーマ

# バルテスとは

|      |                                                                       |
|------|-----------------------------------------------------------------------|
| 商号   | バルテス株式会社                                                              |
| 所在地  | 大阪本社、東京本部、名古屋オフィス、福岡オフィス                                              |
| 資本金  | 9,000万円                                                               |
| 設立   | 2004年4月19日                                                            |
| 社員数  | 285名（2015年6月現在 ※契約社員含む）                                               |
| 事業内容 | ソフトウェアテストサービス<br>コンテンツテストサービス<br>テストコンサルティング                          |
| 関連会社 | バルテス・モバイルテクノロジー株式会社<br>VALTES Advanced Technology, Inc. (Philippines) |



# バルテスの提供するサービス

## ソフトウェアテスト



## 改善・教育

## アプリ開発

モバイル・Web  
アプリ開発



## オフショア



## セキュリティ



さまざまな角度から、**ソフトウェアの品質向上**を支援しています



石原 一宏 (いしはら かずひろ)

バルテス株式会社 R&C部部長 兼 上席研究員

ソフトウェアテスト技術の研究開発、および教育セミナー、  
コンサルティングサービスを担当

大阪大学 土屋達弘教授とテストツール『Qumias』を開発

年間300名を超える開発エンジニアに、**テスト・品質**を教えるセミナー講師

## 主な著書



「ソフトウェアテストの教科書」  
ソフトバンク・クリエイティブ (2012年)



「いちばんやさしいソフトウェアテストの本」  
技術評論社 (2009年)

## 今回のテーマは

テストのプロ直伝！

# 長く使い続けられる 『テスト基盤』の構築方法

こんな**悩み**に向けたセミナーです。

**リリース後も、不具合対応**に  
追われてしまっている。

既存のテストケースでは、  
**不具合が出ない**。

仕様変更で、  
どのテストケースを  
直していいか**わからない**



# 『テスト基盤』構築の実践

# テスト基盤構築のためのアプローチ

## Theme 1

**何を**テストするのかを考える

## Theme 2

**どう**テストするのかを考える

## Theme 3

テストの**全体像**を考える

## Approach 1

**要求分析**

## Approach 2

**F V 表**

## Approach 3

**テストマップ**

# Approach 1 要求分析

## Theme 1

**何を**テストするのかを考える



Approach 1  
**要求分析**

## Theme 2

どうテストするのかを考える

Approach 2  
F V表

## Theme 3

テストの全体像を考える

Approach 3  
テストマップ

# ソフトウェアによるトラブル事例

## ▶ JAL チェックインシステムダウン

2009年6月3日 午前5時45分ごろ、日本航空（JAL）の羽田空港にあるチェックインシステムがダウンし、搭乗手続きが一時できなくなった。バックアップシステムを使って約1時間後に復旧。



## ▶ 三菱UFJ証券 インターネット取引が利用できない

2008年4月28日、追加した13の新機能のうち、3つの機能がシステムに重大な負荷をかけたため、インターネット取引サービスが利用できなくなった。3つの機能を停止することで翌日に復旧した。



## ▶ タカシマヤ 衣料品ECサイトが開始できず

2008年9月21日 開始予定の「タカシマヤ ファッションモール」のサービス開始が5日遅れた。原因は既存ショッピングサイトと会員情報の連携がうまくいかなかったため。



## ソフトウェアの品質が良いとは

『品質は「要求を満たすこと」である』（クロスビー）

『品質は「誰か」にとっての価値である』（ワインバーグ）

「誰か」⇒メーカーが最も大切にすべきは「顧客」

▶ 欠陥が少ない

▶ 予算範囲内

▶ 要求条件を満たす

▶ 使い勝手がよい

例えばこれらは「顧客」にとって品質が良いということである



『品質とは顧客の満足度である』

# 品質と顧客満足の関係

## ▶ 3つの要求

### 基本要求

ユーザーが、これだけは絶対に満たしてほしいと考える要因、特性、パフォーマンス基準。

### 変動要求

この要求を満たすことができれば、ユーザーの評価は上がり、満たすことができないければ評価が落ちるという要因、特性、パフォーマンス基準。

### 潜在要求

ユーザーの期待を上回る要因、特性、パフォーマンス基準。

## ▶ 3つの品質

### 当たり前品質

ユーザーの基本要求を満たすもの。最低限持っておくべき品質。

### 一元的品質

充足されていると満足し、不充足の場合は不満に感じるもの。

### 魅力的品質

充足されていると満足し、不充足の場合は「仕方がない」と感じるもの。

# 品質と顧客の満足の関係

## ▶ 当たり前品質と魅力的品質

### 当たり前品質



#### 1. 「基本要求」

顧客が、これだけは絶対に満たしてほしいと考える要因、特性、パフォーマンス基準。

### 魅力的品質



#### 2. 「変動要求」

これらの要求を満たすことができれば、顧客の評価は上がり、満たすことができないければ評価が落ちるという要因、特性、パフォーマンス基準。

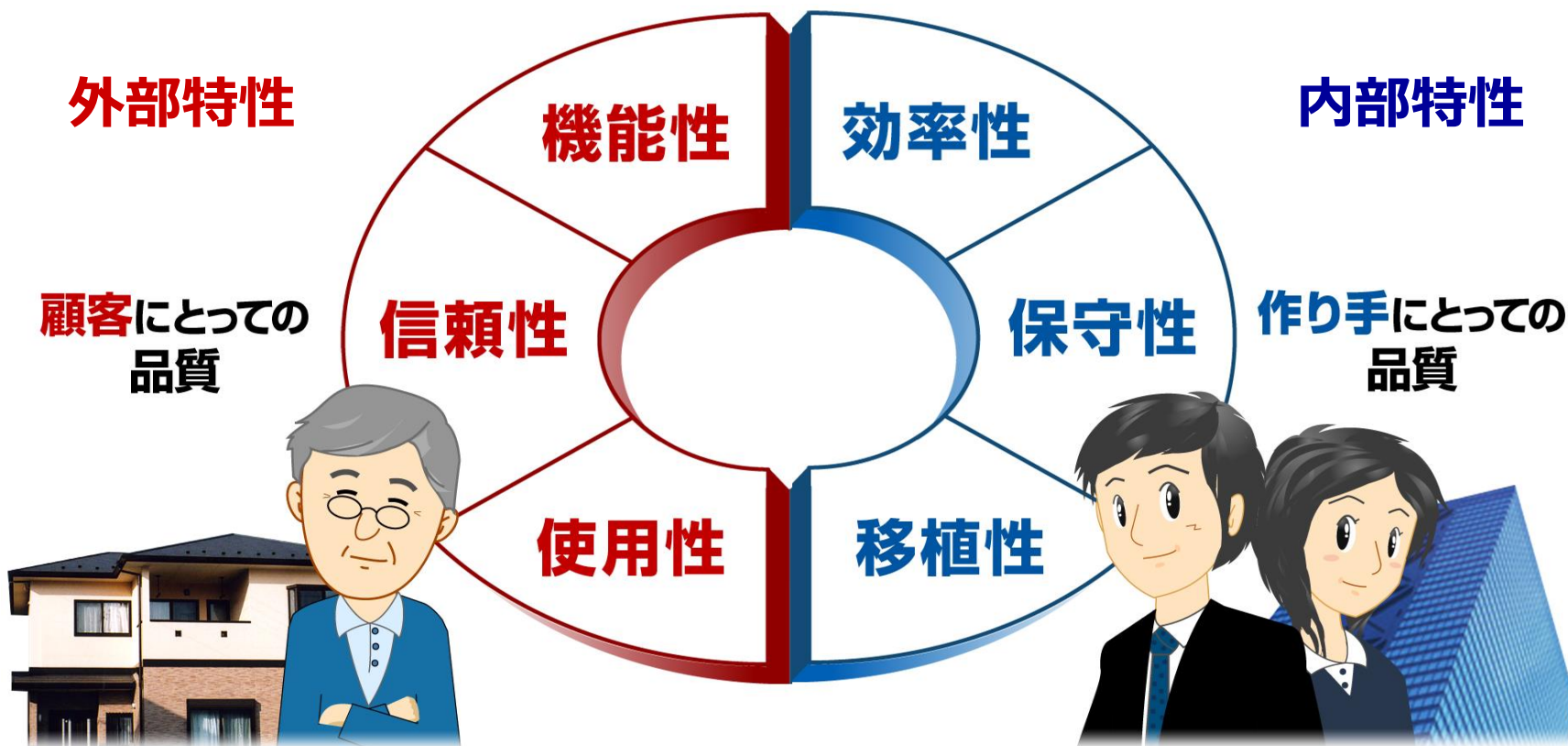


#### 3. 「潜在要求」

顧客の期待を上回る要因、特性、パフォーマンス基準。

# ソフトウェアの品質の観点①

## ISO9126品質特性



# Approach 1 のまとめ

何をテストするのかを考える



要求分析を活用

Never Must Wantは何か

テストすべき対象を抽出する

# Approach 2 F V表

Theme 1

何をテストするのかを考える

Approach 1

要求分析

Theme 2

**どう**テストするのかを考える



Approach 2

**F V表**

Theme 3

テストの全体像を考える

Approach 3

テストマップ

## 「どうテストするのか」の抽出

「要求分析」を使って、「何をテストするのか」を抽出



「何をテストするのか」 = 「テスト対象（機能）」



テスト対象を「どうテストするのか」を洗い出したい

⇒ 『F V表』が有効

## F V 表とは

ユーザーが何をしたいのか（要求）を探っていき、テストの十分性を確保するテクニック。

- ✓ 必要なテストが洗い出せる
- ✓ 要求から見たテストなので、ブレがなくなる
- ✓ テストの優先順位がつけられる
- ✓ 無駄なテストが減る



## Approach 2 のまとめ

どうテストするのかを考える



F V表を活用

機能の目的は何か？

機能目的から検証方法を考える

# Approach 3 テストマップ

Theme 1

何をテストするのかを考える

Theme 2

どうテストするのかを考える

Theme 3

テストの**全体像**を考える



Approach 1

要求分析

Approach 2

F V 表

Approach 3

テストマップ

## 「テストの全体像」を考える

「要求分析」を使って、  
「何をテストするのか（テスト対象機能）」を抽出



「F V表」を使って  
「どうテストするのか（テスト観点）」を抽出



テスト対象機能とテスト観点の全体像を把握したい

⇒ 『テストマップ』が有効

# テストマップとは

『テストのプロ』の定石

バージョンアップや派生開発に強いアプローチ

- ✓ **テストの抜け漏れが激減!**
- ✓ **最短でメンテナンスができる!**
- ✓ **テストと仕様がトレースできる!**
- ✓ **無駄なテストが減る!**
- ✓ **テスト全体が見える化!**



# テストマップのイメージ



テストマップで、仕様書とテストケースを**確実に**紐づける

# テストマップとは

「機能」と「テストの観点」をマッピングしたもの

## これにより

- ▶ 変更時に、修正するテストケースがトレースできる
- ▶ 確実にトレースできるので、運用し続けられる
- ▶ 抜け漏れなく、俯瞰できる

## Approach 3 のまとめ

テストの全体像を考える



テストマップを活用

テスト対象機能は何か？テスト観点は何か？

両者をマトリクスにして、テストの全体像を俯瞰する

# 『テスト基盤』の考え方

# テストマップを中心に据えたテスト基盤の考え方

## テストマップで「仕様」と「テストケース」を紐づけ



### テストケース作成時に

➡ 作成するテストケースがわかる

### 仕様変更時に

➡ 修正するテストケースがわかる

### 追加テスト時に

➡ プライオリティがつけられる

安定したテストの運用には、**「テスト基盤」**



- 変更時に修正するテストケースが、トレースできる
- 確実にトレースできるので、運用し続けられる
- 抜け漏れなく、俯瞰できる

一歩ずつ、確実に  
品質を向上させましょう

御清聴ありがとうございました

---