GMOペパボ

やってみよう！TQC

品質管理の基礎講座

柴田博志 / GMO PEPABO inc.

2016.07.26 新卒研修座学

GMOペパボ



柴田博志　@hsbt

チーフエンジニア

技術部技術基盤チーム

https://www.hsbt.org

self.introduce

```
=>
{
  github: “@hsbt”,
  commit_bits: [
    “ruby”, “rake”, “rubygems”, “rdoc”,
    “psych”, “syck”, “ruby-build”, “jenkins.rb”…
  ],
  sites: [
    “hsbt.org”, “ruby-lang.org”, “rubyci.org”,
    “railsgirls.com”, “railsgirls.jp”
  ],
}
```



品質(Quality)って何だと思いますか?
(自由記述: 2分)

システム工学

・事象をインプット、系(システム)、アウトプットの三つでモデル化する

・インプットと系から生まれたアウトプットが期待値通りであるかどうかが品質の高低を決める



品質というのは絶対値ではなく相対値

品質工学とは何か?

・“開発・設計段階の品質工学”による定義

(品質)=(機能のばらつきによる損失)+(使用コスト)+(弊害項目による損失)

・理想機能のばらつきによる損失

人による価値基準は人によって異なるため、品質というものを工学として扱う時には、価値基準に照らすのではなく、作成物の理想機能と実際に有する機能とのばらつきを基準とする

品質工学の基本的な考え方

・開発設計段階で品質を作り込む

→ 製造段階で作り込むから機能のばらつきが発生する

・よくある誤解

「テスト・試験で品質向上」

・テストでわかることは作成物の品質がどの程度かであり、作成物そのものの品質が変化するわけではない

企画段階の品質管理

“品質”はユーザーの期待する機能と実際に提供している機能との乖離である

高品質

・機能と理想機能との乖離のばらつきが小さいこと

・プラスの乖離が高いことだけが高品質とは呼ばない(例: 過剰品質)

低品質

・機能と理想機能との乖離のばらつきが大きいこと

・マイナスの乖離が大きければユーザーの満足感を満たすことはできない

当たり前品質と魅力的品質

狩野(魅力的品質と当たり前品質)による定義

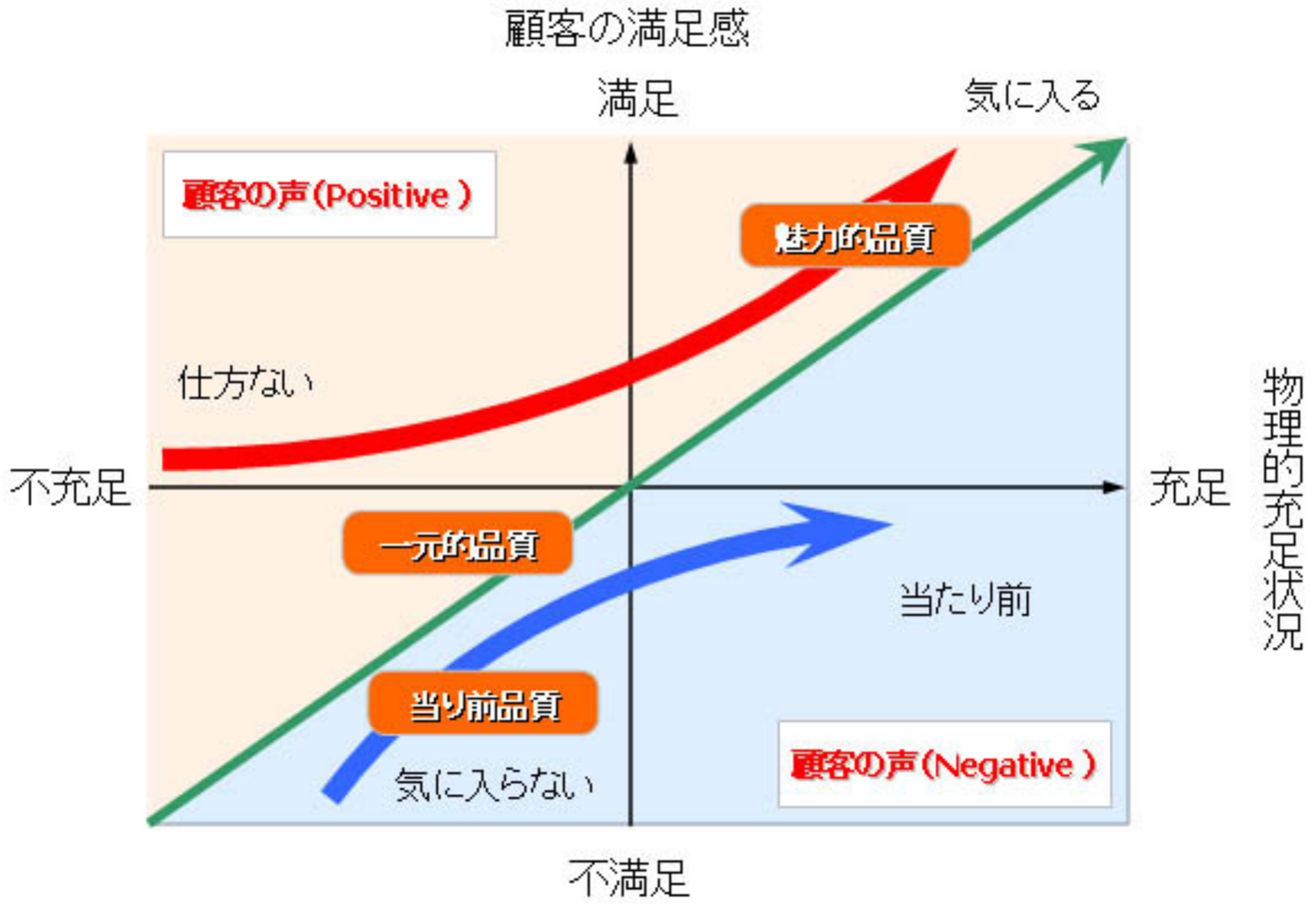
当たり前品質

・品質基準を満たさない場合、ユーザーの満足度を低下させる要因となるが、満たしていても満足度の向上は見込めない

魅力的品質

・品質基準を満たした場合、ユーザーの満足度を向上させる要因となるが、満たしてない場合でも満足度には影響しない

当たり前品質と魅力的品質



https://sites.google.com/site/techdmba/kanomodel

スマートフォンの
当たり前品質と魅力的品質を考えよう
(自由記述: 3分)

twitter の
当たり前品質と魅力的品質を考えよう
(自由記述: 3分)

品質管理と歴史

- 品質(Quality) は 管理(Control)する
- 1940年代: 「安かろう悪かろう」
- 1950: デミング博士を日本に招き品質管理に取り組む
- 1951: デミング賞の設立
- 1960年代: QC活動と呼ばれる現場からグループ単位での品質改善活動(ばらつきを抑える)が盛んになる
- 品質といえば “Made in Japan” の時代へ



TQC(Total Quality Control)

・QC活動は現場を中心とした小集団によるハードウェアの改善活動

・TQCは人々のコミュニケーションに目を向けたソフトウェア(方針管理、日常管理)の改善活動

ソフトウェアと品質管理の歴史(1)

・1960-70年代: 色々破綻して”ソフトウェアの危機”と呼ばれる

・1968年: ソフトウェア工学という言葉が生まれる

・1990年代: ISO/IEC 9126 の策定(JIS X 0129)



JIS X 0129 に定めるソフトウェア品質特性

- 機能性（functionality）
- 信頼性（reliability）
- 使用性（usability）
- 効率性（efficiency）
- 保守性（maintainability）
- 移植性（portability）



ソフトウェアと品質管理の歴史(2)

・2001年: アジャイルソフトウェア開発宣言

(なんだかんだあって15年経過)

・2016年現代: 今もなおソフトウェア開発は難しい



ソフトウェア開発が難しいのはなぜか?
(自由記述: 5分)

“ソフト”ウェア

“私は、ソフトウェア構築において困難な部分は、この概念構造体の仕様作成とデザインおよびテストにあって、それを表現する仕事やその表現に忠実か否かをテストする仕事ではないと考えている。”

『人月の神話【新装版】』丸善出版、2014年、p.170

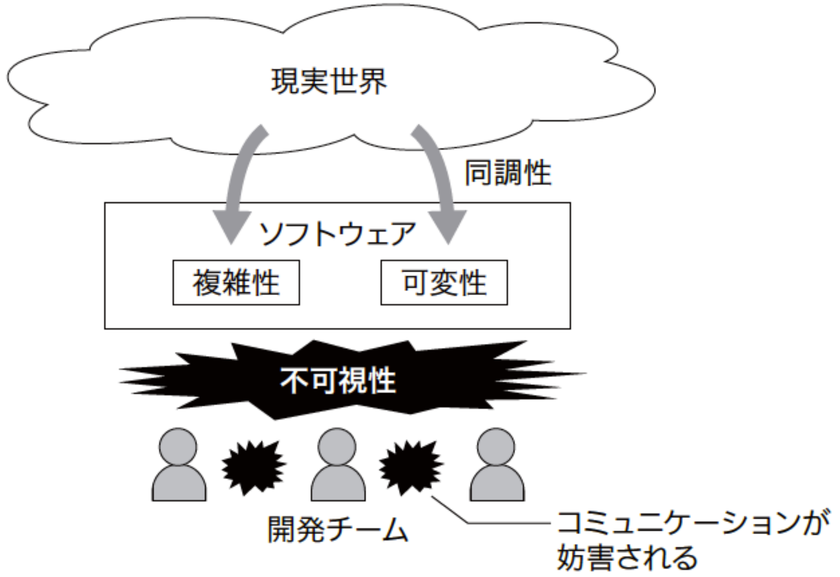


ソフトウェアはどこにあるのか

・ソフトウェアは人々の心の中にある

・ソフトウェアは人々のコミュニケーションの中にある

『スクラム実践入門』技術評論社、2016年、p.10



ソフトウェア開発の困難さに立ち向かう

・ソフトウェアパラダイム

→ 構造化プログラミング、オブジェクト指向プログラミング、GC、型推論…

・ソフトウェアアーキテクチャ

→ GoFのデザインパターン、MVC、MVVM、Micro Services…

・ソフトウェア開発プロセス

→ ウォータフォール、スパイラルモデル、XP、Scrum…



ソフトウェア開発と TQC

- 2016年現在、ソフトウェア開発は品質工学の一分野として語れるほどにまで工学として昇華していない
- 未だに人に強く依存し、産業としても未成熟な業界
- 人と人とのコミュニケーションに目を向けて困難さを乗り越えるという部分は TQC に通じるものがある(と hsbt は考えている)



まとめ

・品質は絶対値ではなくて相対値

・品質はテストによって得られるものではなく自分たちが作り込むもの

・自分が作っているものの品質は何かを意識する(当たり前品質と魅力的品質)

・そもそもソフトウェア開発は難しい

・難しいのを理解した上でより良いソフトウェアを生み出すために頑張りましょう

