

Slerの中から見た  
**Agile開発**

2010年11月21日  
高橋

大切なことは  
最初に

ご清聴ありがとうございます  
ございます！

まとめ

Agile開発は、

価値  
スピード  
卓越した仕事

をもたらします

# 自己紹介

- 愛媛出身、愛大出身(ローカルな話)
- 在京のSlerの中の人
- Javaとか.NETとか、Rubyとかをやっています
- なんとか10年勤めました

- twitter: @walnut210
- mail: walnut.g@gmail.com

洒 值

「このシステムは高品質です」という  
キングファイル10冊の紙



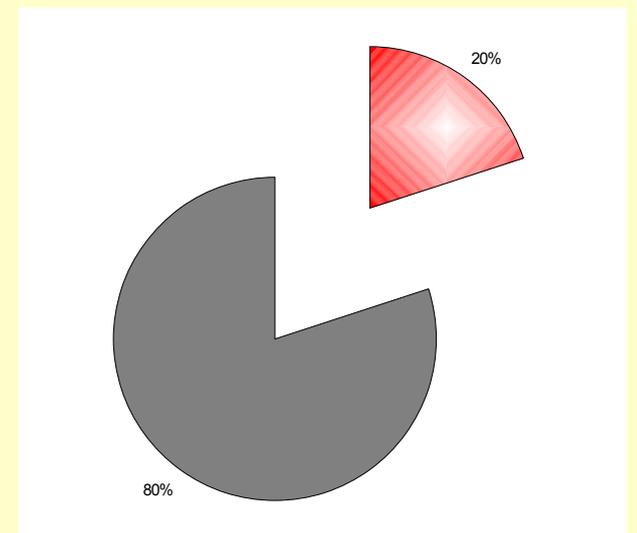
「このシステムはユーザーストーリーを満たしたテストをパスしています」という実際のプログラム



どちらが  
必要ですか？

優先度の低い要求  
はありますか？

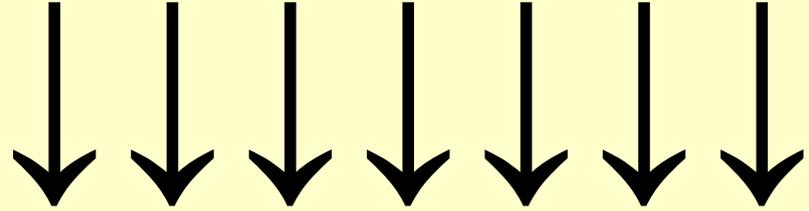
# 価値を生み出す2割に 集中します



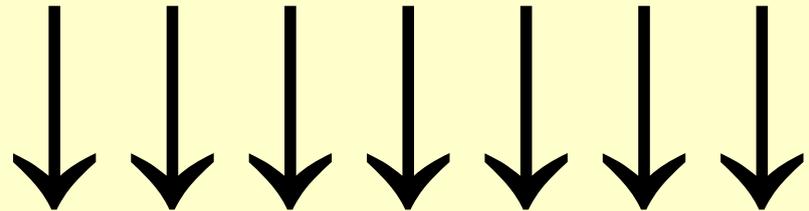
残る 8割を「今回の開  
発からはずす」には  
ユーザーさんの意思が  
不可欠です

スピード”





# 短納期化の圧力



「6ヶ月～1年」が...

「じゃあ3ヶ月で」  
になった

開発完了後→本番環境  
開発中は本番環境相当に載らない

引き渡してからクリティカルな問題が出る



工程管理が建前に

実際、工程を終えました報告した翌日

「要求変わりました」

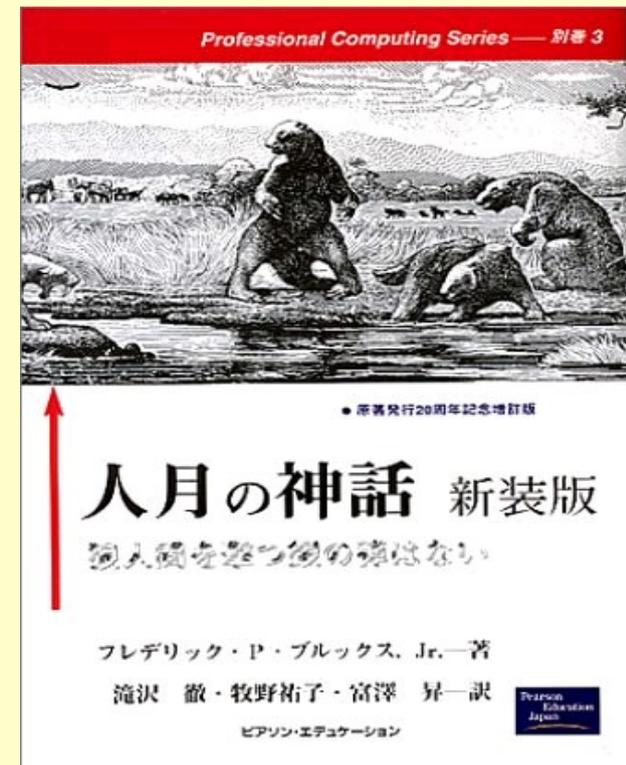
で、やり直し。  
しかも納期は  
「変わらない。」

「手戻りなのでリスケです」というと揉める

~~よるしいならば戦年だ~~

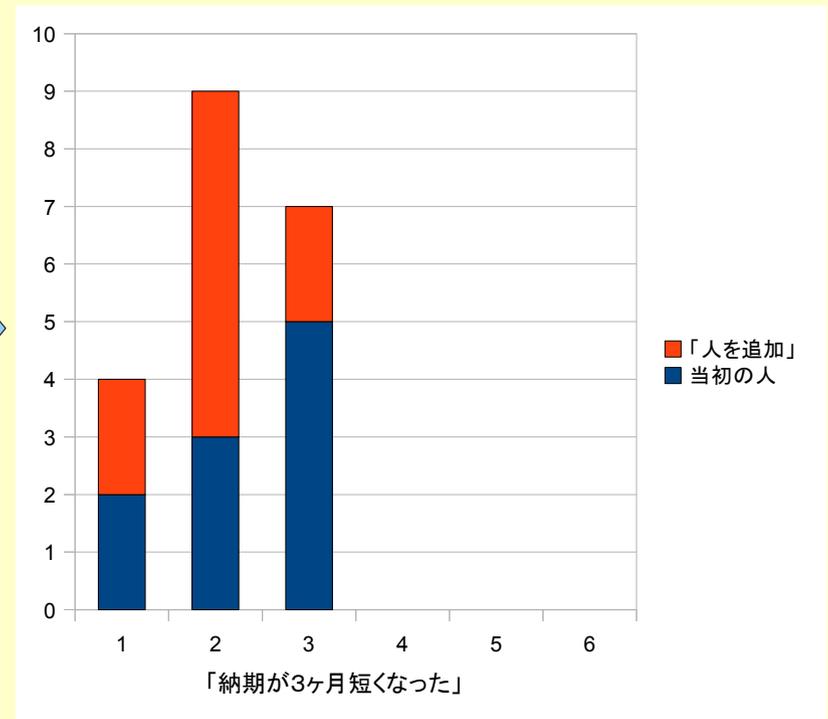
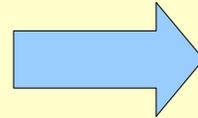
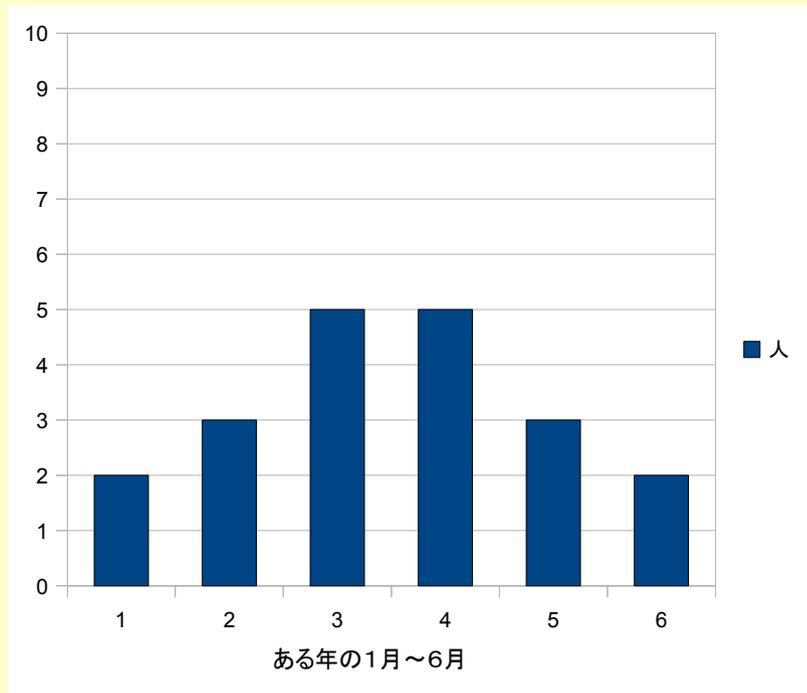
揉めて、3ヶ月の内の1週間停めるともうツライ

# 「人月」という単位



「人」と「月」は交換可能ではない

# よくある間違い



# 増員して月を減らす条件

- 文脈に依存しない作業
  - 機械にまかせられる
- お隣さんじゃないサブシステム
  - 文脈を共有していると何かしら影響はある
- 経験者を増やす
  - コミュニケーションが浸透するまでのオーバーヘッド

# 期間を延ばして人を減らす条件

- 期間を無限大に持って行くと0人でできる??
  - 探索型のプロジェクトは、期間も延びるし、人もそれなりにかかる。
- 四半期や半期に製品が更新されるので、期間を延ばすこと自体が難しい
- 企画と予算折衝で知らないうちに時間がとられる
  - フロントローディング
    - 開発者も踏み込んでいく。
    - 早く始める。

※業界全般のお話です。念のため。

卓越した  
仕事



「ゴールを奪う」のが仕事だとすると



Agile開発は、「ゴールを奪う」ことに集中します

現状は、「ゴールを奪う」ために、

企画書を作成し、  
計画書を作成、  
詳細な計画書を作成し、  
計画書に沿ってゴールできることをチェックし、  
企画どおりゴールできることをチェックし、  
それぞれの課程が完璧であることチェックし、  
偉い人の承認を受けてからやっている。

と言えます。  
その間に、

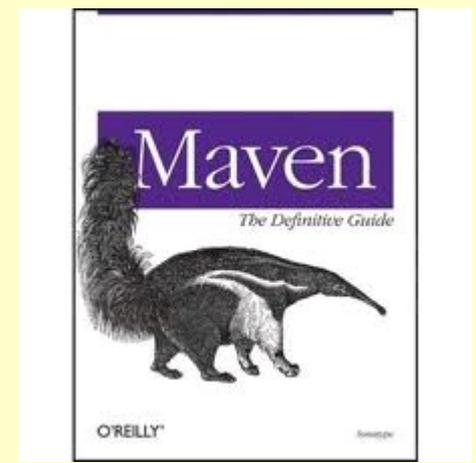
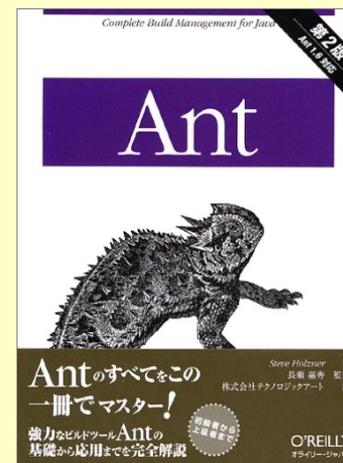
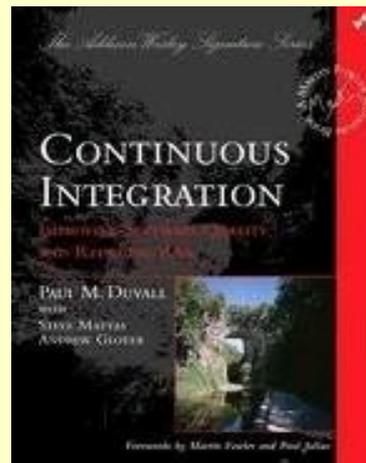
「ゴールが移動」します。



真に価値のあるコードを書きます

手順を創ります

手順を守ったり、記録したりするのは、  
まずコンピューターにさせることです



人が手順を守るということも大事

# 飛行機や鉄道の運行



プロトコルにまず従う

時々、例外が発生する

例外対応もプロトコルがある  
しかも人間の判断重要

Agile開発では、

手順は

プロが自分達を律するためにあります

なぜその手順なのか、  
いつ使うべきか、  
いつ変えるべきか、  
変えるならばどう変えるか、  
を知っています

何故、  
Agile開発なのか

「ウォーターフォールでうまくいってる」

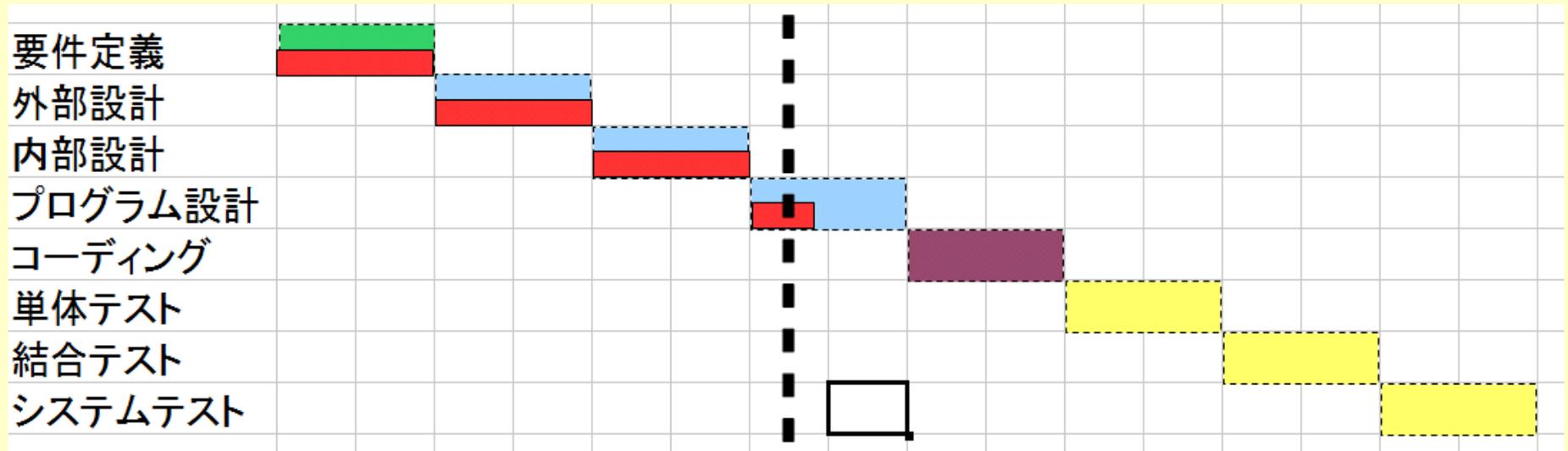
可能性1

騙されてます

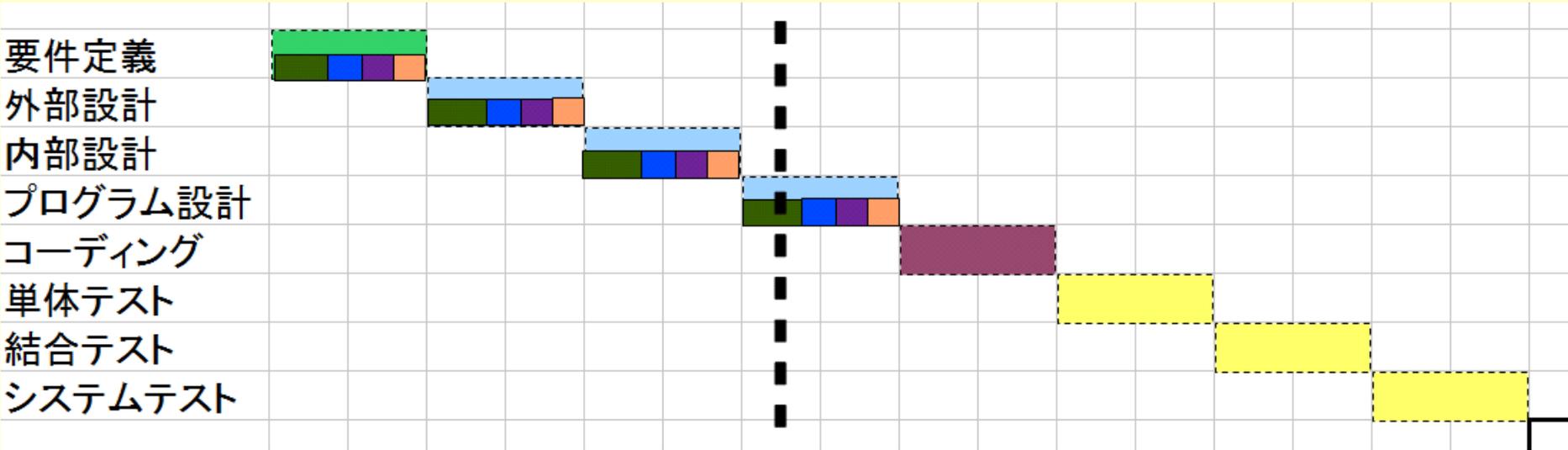
可能性2

実は  
Agile開発をやっている

# 報告書上はこう



# 実際はこう



「実際は・・・」は、もともとは行われていた

「プロトタイピング」  
「先行開発」

「IS○を記録的に満たせばよい」

となって、

「実際は」から

日本的な、フィードバックの無いウォーターフォール  
になった

混沌に光を当てた効果はあった

光が強すぎて、実態が見えない  
「消失効果」



可能性3

本当にうまく行っている

Agile開発の必要ない領域

Where?

課題だと  
思っていること

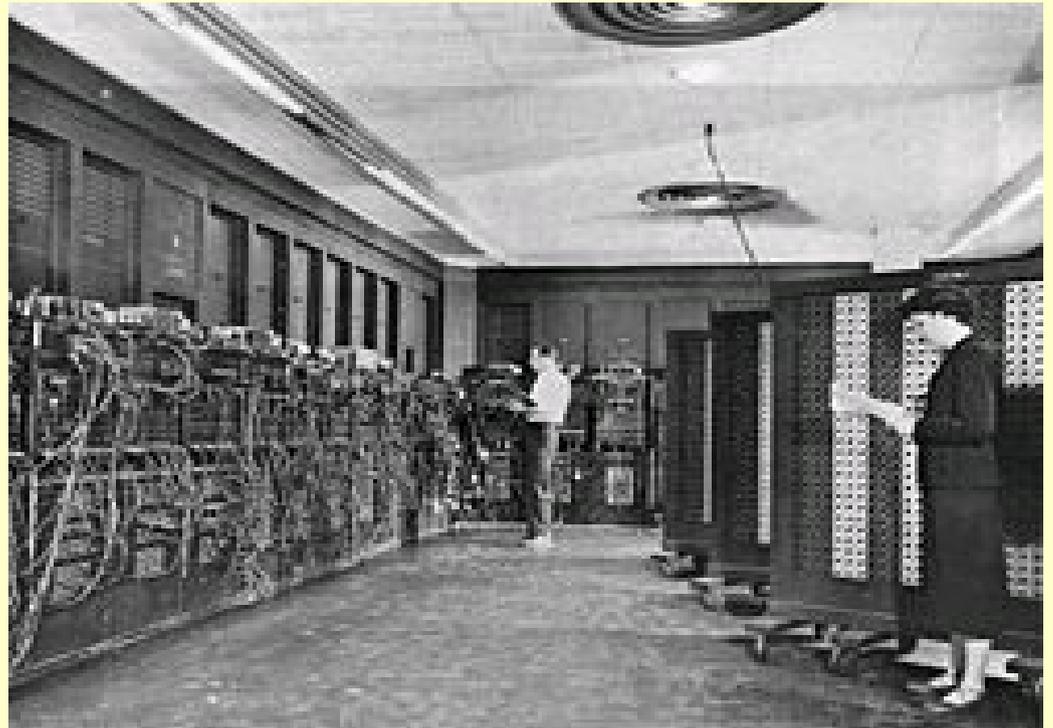
# 大規模システム

- アーキテクチャで縛る必要がある
  - アーキテクチャとアジャイルの融合
- 文脈に依存しない作業を作れる
- 



# レガシーシステム

- 一度に全ては変えられない
  - アジャイルでないコードやシステムと付き合う必要がある



# ビジネス慣習

- 要件は固められる→常に変わる
- 工程を管理できる→動くシステムをWatchする

まとめ

Agile開発は、

価値  
スピード  
卓越した仕事

をもたらします

Q&Aは懇親会で

ありがとう  
ございました！

「ものづくり」の向こうの「Agile開発」を目指して