

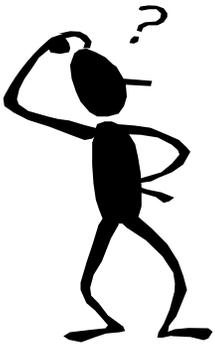
Taming a Monster

●ソフトウェア品質問題を管理するために●

2007年1月31日

山浦恒央 (YAMAURA, Tsuneo)

yamaura@keyaki.cc.u-tokai.ac.jp



ソフトウェア開発と品質



●理想の自動車のジレンマ

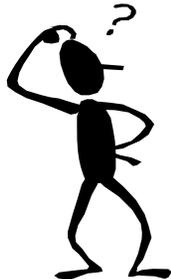
ガソリン1リットルで30km走り、6人がゆったり座れて大型ボストンバッグも6個積める。最高速度は250kmで、操作性、安全性抜群。スタイリッシュで、最新の電子機器を搭載しているのに、故障知らずで環境汚染もしない。しかも、1台50万円。

「理想の自動車」は作れないことは誰でも理解しているのに、ソフトウェアでは可能と考える顧客は多い。

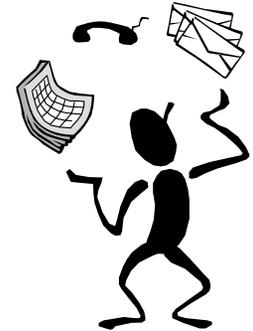
ソフトウェア開発と品質

●なぜ、「理想の自動車」が可能と思うのか？

- (1) ソフトウェアは何でもできる「魔法の力」があるとの幻想を抱く人が多い。
- (2) ソフトウェアは目に見えず、複雑さが実感できない。
- (3) 工学の3要素、「安い(コスト)」、「早い(スケジュール)」、「美味しい(品質)」の中で、コストとスケジュールは定量的計測が可能だが、品質は難しい。



ソフトウェア開発と品質



●「現実的な自動車」を開発するには？

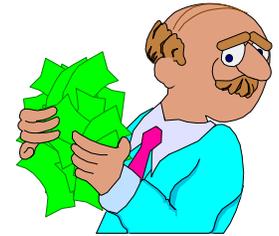
→ 「工学的な」妥協が必要

- ・「おいしい(品質)」、「早い(納期)」、「安い(コスト)」間の妥協
- ・品質特性(機能性、信頼性、使用性、…)間での妥協

→ Triage的な考えの導入

- ・限られた予算と日数をどこに投入するか判断

ソフトウェア開発と品質



- 開発費と開発日を早期に食いつぶさないための具体策
(何もしなくても、コストと日数は消費するが、品質はできない)
 - (1) 開発のライフサイクル全部を1つの契約にするのではなく、「要求仕様作成」、「設計書作成」、「コーディング」等、短い単位で見積もり、契約する。
 - (2) 「開発バランスシート」を可視化し、毎日更新する。これにより、現在、いくら開発費が残っているかを開発側と発注側の両方で見えるようにする。
 - (3) 見積もりにおいて、「工数」、「開発期間」、「機能総量」、「生産性」、「品質」の関係式[L. Putnam]を利用し、発注側を説得する。
→次スライド参照

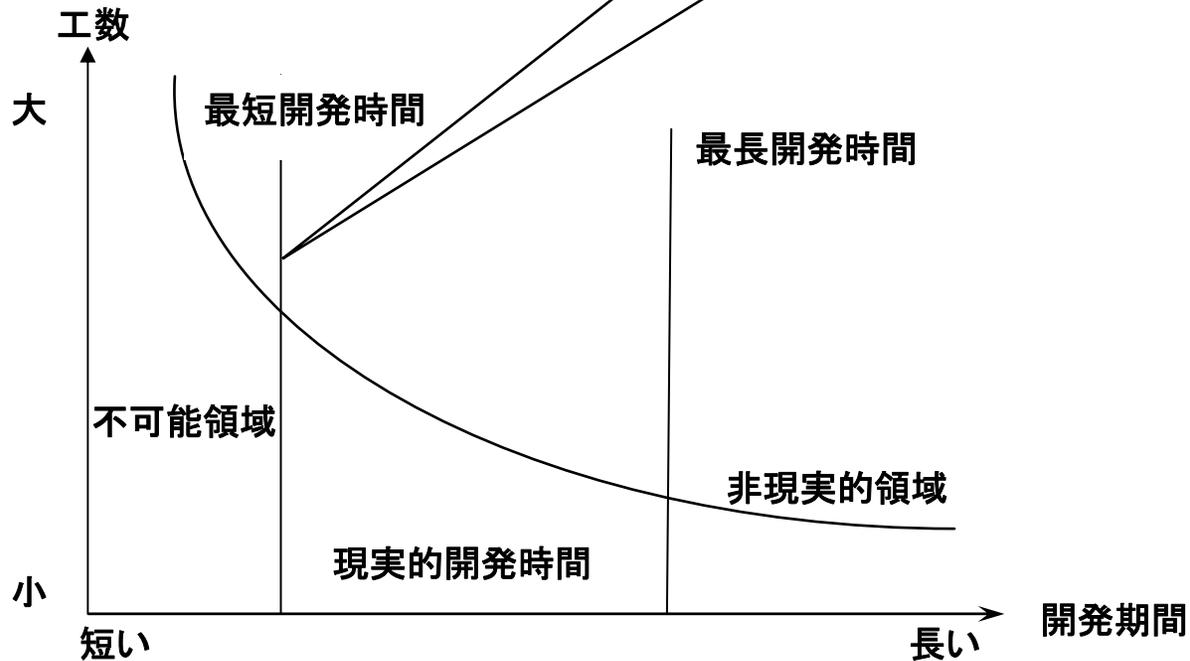
ソフトウェア開発での工数、開発期間、機能総量、生産性、品質の関係式

[Lawrence Putnam]

$$\left(\frac{\text{工数}}{\beta}\right)^{\frac{1}{3}} \times (\text{開発時間})^{\frac{4}{3}} = \frac{\text{機能の総量}}{\text{生産性}}$$

$\beta = \text{定数}$

無限に開発要員やテスト技術者をどれだけ増やしても、これより短くならない「最短開発時間」が存在する。



品質確保の具体策



(1) 品質開発マネージャの設置

QA部門の技術者を1人、開発プロジェクトに配置する。

- ・品質管理のプロであり、品質の最高責任者となる。
- ・開発担当者を実働部隊として、品質を向上させる。

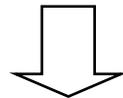
(2) 「人間仕様書」の設置

・仕様の細部と発注側の業務を何でも知っている開発技術者を設置する。

・実際の開発作業はせず、仕様の管理のみを担当する。

(3) プロジェクト・マネージャの役割

・コスト管理とスケジュール管理を担当する。



「品質開発マネージャ」、「人間仕様書」、「プロジェクト・マネージャ」が協調的にソフトウェアを開発する。