

DX時代におけるソフトウェア/ビジネス/社会のあり方を考える

1/27/2021

～平鍋健児～



平鍋健児

(株)永和システムマネジメント社長

(株)チェンジビジョンCTO

(株)Scrum Inc. Japan 取締役

福井で受託開発を続けながら、アジャイル開発を推進し、国内外で、モチベーション中心チームづくり、アジャイル開発の普及に努める。ソフトウェアづくりの現場をより生産的に、協調的に、創造的に、そしてなにより、楽しく変えたいと考えている。アジャイルジャパン初代実行委員長。



ソフトウェア工学についての後悔

Tom Demarco

- ソフトウェア工学、そのときは去った。

Ed Yourdon

- ソフトウェア工学に大切なことは？

Barry Boehm

- あの指数曲線は間違いだった。

Ivar Jacobson

- ソフトウェア業界は、ファッション業界のようだ。

Tom Gilb

- ソフトウェア工学は定義を間違った。

A close-up photograph of Tom DeMarco, an older man with white hair and glasses, wearing a blue shirt and a dark tie. He is speaking into a microphone, with his mouth open and a focused expression. The background is a wooden panel wall.

Tom DeMarco

Software Engineering:

An Idea Whose Time Has Come and Gone?

Tom DeMarco

IEEE SOFTWARE 2009

We're now just past the 40th anniversary of the NATO Conference on Software Engineering in Garmisch, Germany, where the discipline of software engineering was first proposed. Because some of my early work became part of that new discipline, this seems like an appropriate moment for reassessment.



My early metrics book, *Controlling Software Projects: Management, Measurement, and Estimation* (Prentice Hall/Yourdon Press, 1982), played a role in the way many budding software engineers quantified work and planned their projects. In my reflective mood, I'm wondering, was its advice correct at the time, is it still relevant, and do I

still believe that metrics are a must for any suc-

Compelled to Control

The book's most quoted line is its first sentence: "You can't control what you can't measure." This line contains a real truth, but I've become increasingly uncomfortable with my use of it. Implicit in the quote (and indeed in the book's title) is that control is an important aspect, maybe the most important, of any software project. But it isn't. Many projects have proceeded without much control but managed to produce wonderful products such as GoogleEarth or Wikipedia.

To understand control's real role, you need to distinguish between two drastically different kinds of projects:

- Project A will eventually cost about a million dollars and produce value of around \$1.1 million.
- Project B will eventually cost about a million dollars and produce value of more than \$50

ソフトウェア開発は、いつでもどこか実験的要素を含む。。。しかし、こここそが、本来、焦点を当てべき場所だったのだ。

-Tom DeMarco





Ed Yourdon

Top Ten Eleven Items

1. 計測できないものは制御できない
2. ピープルウェア (Peopleware)
3. インクリメンタル (Incrementalism)
4. 反復 (Iteration)
5. 欠陥が下流に漏れること、修正コストが増加する
6. トレードオフは、非線形
7. 再利用は重要
8. リスクマネジメントが鍵
9. 一貫性は才能+デスマーチに勝る
10. 車輪を再発明しない
11. 透明性を重視。何も隠さないこと



Barry Beohm

<http://www.flickr.com/photos/74987020@N00/131796142/>

Retrospectives on Peopleware

Steven Fraser (Impresario), QUALCOMM

Barry Boehm, Director, USC Center for Software Engineering

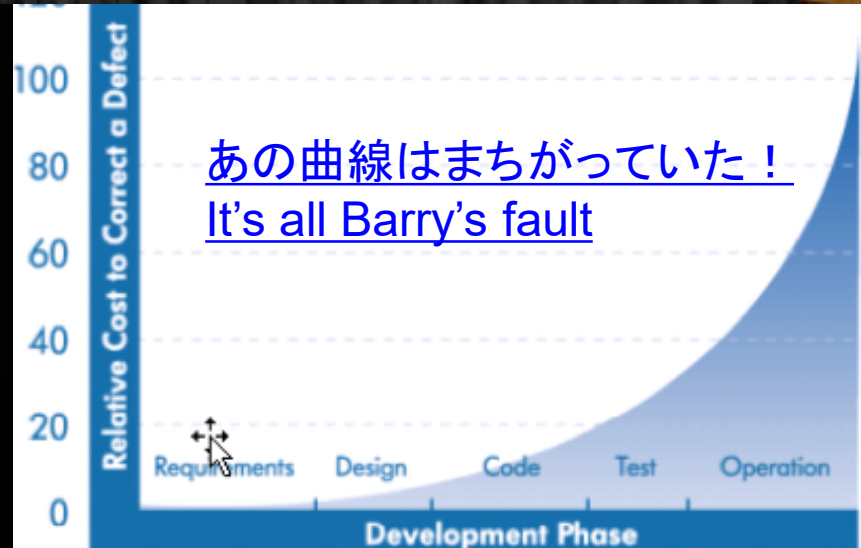
Fred Brooks Jr., Kenan Professor, University of North Carolina at Chapel Hill

Tom DeMarco, Atlantic Systems Guild

Tim Lister, Atlantic Systems Guild

Linda Rising, Independent Consultant

Ed Yourdon, NODRUIJOY Inc.



A photograph of Ivar Jacobson, a man with glasses and a dark jacket, smiling and speaking into a microphone. He is holding a small card in his left hand. The background is a plain, light-colored wall.

Ivar Jacobson

SEMAT Call for Action(1/2)

ソフトウェア工学は未成熟なプラクティス(immature practices)によって、重大な阻害(gravely hampered)を今日受けている。例えば、具体的には以下のように:

- 言葉の流行が、工学の一分野というよりファッション業界のようだ。
- しっかりした広く受け入れられた、理論的基礎の欠如。
- 非常に多くの方法論(methods)とその派生。またそれらの違いがほとんど理解されずに作為的に強調されている。
- 信頼できる実験的評価(experimental evaluation)と妥当性確認(validation)の欠如。
- 産業界の実践(industry practice)と学界の研究(academic research)の乖離。



Tom Gilb

ソフトウェアシンポジウム第一回

ソフトウェア工学とは、ウェアシステムの「価値」をバランスしながら、バランスしたステークホルダーに提供する、ソフト工学の1分野である。

– *Tom Gilb*

ソフトウェア工学についての後悔

Tom Demarco

・ ソフトウェア工学、そのときは去った。

Ed Yourdon

・ ソフトウェア工学に大切なことは？

Barry Boehm

・ あの指数曲線は間違いだった。

Ivar Jacobson

・ ソフトウェア業界は、ファッション業界のようだ。

Tom Gilb

・ ソフトウェア工学は定義を間違った。

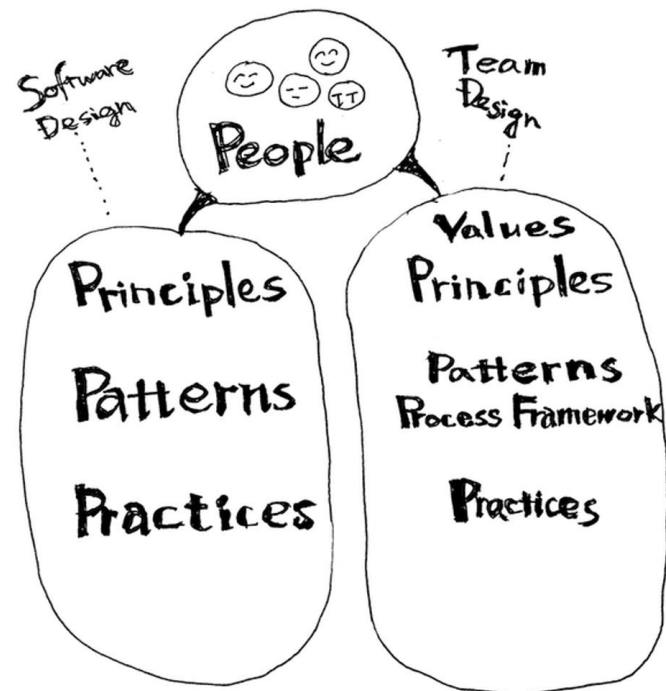
ソフトウェア工学とアジャイル

- ソフトウェア工学は、ソフトウェアのデザインからチームデザインへと移してアジャイルへと広がった。
- アジャイルは「ビジネス」と「ピープル」を文脈に強く持って、ソフトウェア開発を席卷した。
- それだけでなく「組織論」として発展。
- 「考える」ことと「行う」こと、を分けない。やっていない人が論じない（全体性の回復）。

ソフトウェアデザイン



チームデザイン



アジャイルを 日本で広げたい

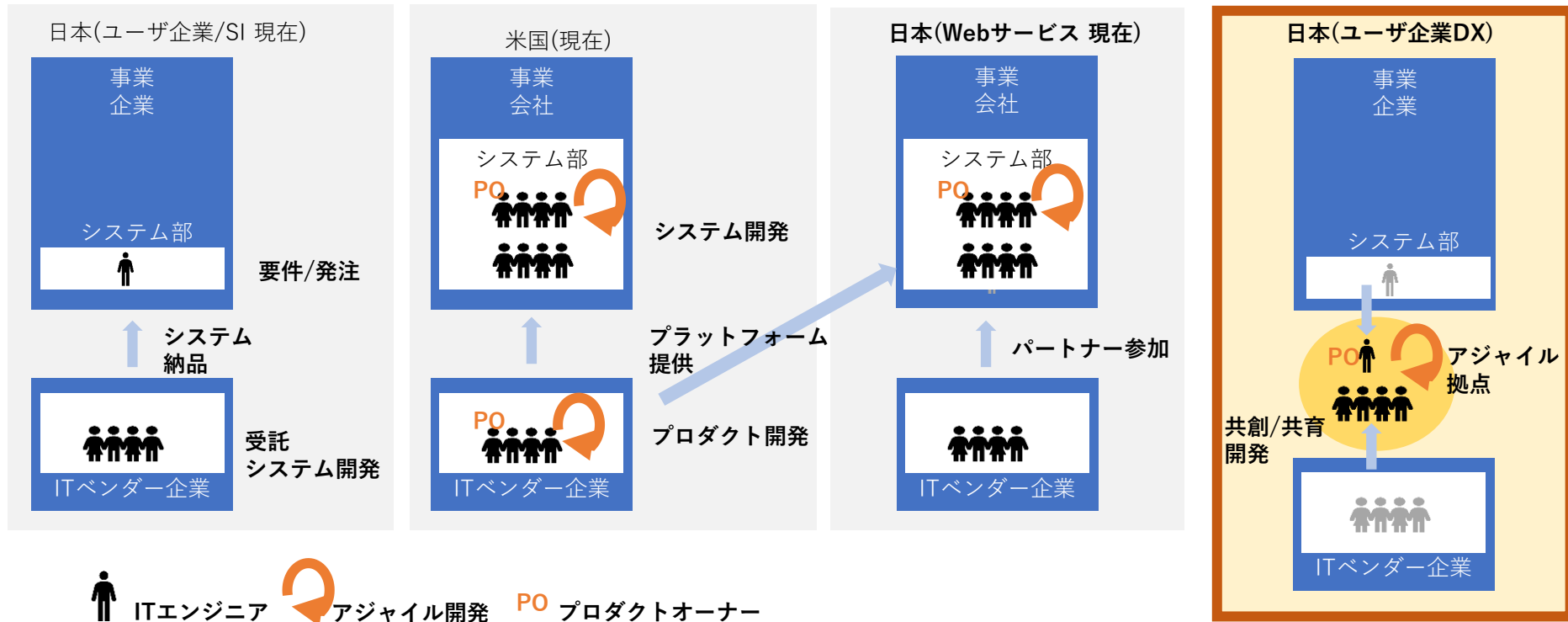
- 首都圏、地方を問わず、優秀なエンジニアは多い。
- エンジニアが直にビジネス、社会に影響を与えられる時代。
- ソフトウェアはもはや、調達、ではない。イノベーションの核。
- もともとチームワークは、日本は得意なはず。
- 人と関わること、の大切さ。
- わくわく、ドキドキ、やった！ の感覚。
- エンジニアとしての、倫理観を持って、社会と関わる。



という ことで、 私の講演は…

- 「アジャイル入門」 + 「事例」です。
- スクラムの基礎と、技術プラクティス、チーム作りについて。
- アジャイルとDXの潮流、日本のソフトウェア産業構造とDXについて、も言及します。
- 時間があれば「経営/組織理論」（野中郁次郎先生）とスクラムについても。

アジャイル開発の形（受託から共創/共育へ）



第二版



企業のリーダー層に向けた **SE**
「アジャイル」と「スクラム」の解説書
「組織や社会を考えるリーダーシップが、
トップにも、現場にも、そして スクラムの普及 - 一橋大学名誉教授
ミドルマネジメントにこそ必要だ。」野中郁次郎
リクルート・富士通・楽天・ANA...日本を代表する企業の開発事例を収録!



 kenji.hiranabe@change-vision.com
 kenji.hiranabe  @hiranabe
 hiranabe