

TPS/リーン方式の基本原則とアジャイルソフトウェア開発への適用

2012/1/3 (社)持続可能なモノづくり人づくり支援協会(ESD21)

黒岩 恵 (skuro@esd21.jp)

1. TPS の概要

1.1 TPS/リーン方式とは何か

日本の自動車メーカーが小型車の品質とコストで米国車を凌駕し、80年代初に米国で生産を開始して以来、日本の製造業の強さは、米国製造業復権のテーマとしてMITなどで研究された。彼らは強さの源泉の一つとしてTPS(Toyota Production System=トヨタ生産方式)をリーン(贅肉のない)生産方式と命名し、90年代以降、世界中でTPS/リーン方式は、製造業の生産性向上の手法として広まった。

TPSは、トヨタが米国の自動車産業から学び、大野耐一ほか多くの先人により確立されたトヨタ独自のモノづくりの思想である。現在のトヨタでは、TPSは生産現場だけでなく、生産・物流、販売、製品開発、管理間接部門へと広く適用され、トヨタウェイの具体的展開における経営哲学となっている。90年代後半から、ICT(情報通信技術)の技術革新とグローバル化により、生産現場中心の伝統的TPSは、今では製造現場を超えた生産性向上の手法として進化し展開されてきた。海外では、米国を中心にTPS/リーン方式はビジネス活動の競争優位戦略として航空機、電子機器、家具、さらに製造業を超えて、流通業、ヘルスケア、情報サービス産業などに広く普及・展開されている。TPSを学び、USの研究者らがコンセプトや方法論を確立し、日本に導入された手法は多い。シックスシグマ、ゴールドドラットのTOC、米流通業のECR、そしてアジャイルソフト開発手法など。日本のIT関係者の多くは、「ITはアメリカ」としてUSから学ぶ。逆に世界の人たちは「ものづくりの強さの原点であるTPS」を今でも真剣に学んでいる。30年前に確立したTPSは、今ではUSの研究者やICTにより進化しており、筆者は伝統的なTPSとの違いを強調して、敢えてTPS/リーン方式と呼ぶ。

TPSは「お客様第一」を基本理念として、「ジャストインタイム」と「自動化」の二本の柱でなる。「自動化」とは、「機械に知能を与える」、「動きを働きに変える」、「品質は工程で造りこむ」、「自立/自律化」、の意である。トヨタの和製英語である「ジャストインタイム(Just-In-Time)」とは「必要なモノを必要な量、必要な時に造り、運ぶ」の意味であり、目的はリードタイム短縮という物づくりの重要なコンセプトである。

1.2 TPSのゴールは「あるべき姿に向けて改善しつづける人間集団」を創ること

トヨタはグローバル展開の拡大に対応するため、「トヨタウェイ2001」を策定した。これはトヨタに受け継がれた経営上の理念や価値観を体系的にまとめた英語・日本語

併記の小冊子で、世界中のトヨタで働く者の行動規範である。暗黙知として存在するトヨタ独自の思想・哲学をグローバルで情報の見える化、共有化を計り、価値観の異なる海外事業体のマネジメント層にも実践してもらおう意図である。トヨタウェイは、「知恵と改善」、「人間性尊重」の2つの柱とそれを支える5つの要素、Challenge、Kaizen、Genchi Genbutsu(現地現物)、Respect、Teamwork から成立つ。グローバルビジネスが拡大する中で、ビジョンやミッションの共有化、アクションプランの策定や活動に、全員参加による知恵の創出が期待され、その具体的展開がTPSである。TPSのゴールは「あるべき姿を求めて改善し続ける人間集団をつくる」とした、人的能力の向上にある。特に海外では、TPS実践の方法論やツールの導入をとおして、創意工夫や改善できる人材の育成を計り、TPSの導入とレベルアップが行われる。

筆者はTPSの本質は、かんばん方式などトヨタやトヨタグループで適用されているTPSの方法論の導入ではない、と言う。その本質は、TPS実現のツールや技術、方法論の実践をとおして、改善力、現場力、すなわち人的能力の向上する人材育成にある。「かんばん」や「アンドン」、「ポカヨケ」などTPSで使われるツールや方法論は、多くの人に見える氷山の一角に過ぎない。TPSの本質を理解するためには、TPSの原理原則を追及することが重要。「かんばん」や「アンドン」などのツールの使い方、Know-Howよりも、ツールや方法論の依って立つ根拠、Know-Whyを理解頂きたい。ツールや方法論適用の意味や意義を知れば、それが、人のモチベーションを高め、人の能力を最大限に生かし、個人の能力だけでなく、チーム力、組織力を高めるための仕掛けやしぐみであることを理解できよう。TPSの人間性尊重を基本とする原理原則を理解すれば、TPSが生産現場、生産システムへの改善・改革手法だけではなく、流通業、ヘルスケアなどのサービス業、さらには自治体や学校経営などの公的経済主体の生産性向上にも有効であることを理解できよう。一方、TPSのゴールで言う「あるべき姿」とは何か。「あるべき姿」とは、お客様の注文(需要)に応じて(Pull)、仕事に流れを創る事、モノ(部品や車両など)の流れを創る事にある。鴨長明の方丈記の冒頭に「行く川のながれは絶えずして、しかも本の水にあらず。よどみに浮ぶうたかたは、かつ消えかつ結びて久しくとゞまることなし。」とある。仕事に流れを創る、モノの流れ、情報の流れ、設計図の流れ、決裁資料の流れを創る事、仕事の流れを構成する工程で、その流れを阻害し、滞留させることは、自然の流れに逆らっており、TPS流に言えば、ムダが新たなムダを生む、ことになる。

1.3 TPSの本質は、ムダ排除などの改善活動を通じた人材育成

TPSは、仕事の上流から下流へのPushでなく、お客様という下流からのPull、すなわち「お客様の引きに応じて仕事に流れを創る」活動である。生産・物流工程では、お客様の引きに応じたモノの流れの構築。オフィス業務においては、お客様の引きに応じた企画、設計、決裁などの情報の流れの構築である。モノや仕事に流れをつくり、

ムダを徹底的に排除して QCD(Quality, Cost, Delivery) を向上する。そのための第一歩は、5S(整理、整頓、清掃、清潔、躰)、そして「見える化」である。モノの流れ、仕事の流れを見える化し、チーム活動による知恵の創出、改善活動を可能にする。

TPS の二本の柱は「ジャストインタイム」と「自動化」である。TPS における改善の肝は「ムダを排除したリードタイム短縮」にある。リードタイムとはあるアクションを起こして終了するまでの時間を言う。「ジャストインタイム」の目的はリードタイム短縮にあり、リードタイム短縮に向けて、自律的に(=自動化)に「ムリ、ムラ、ムダ」を排除して改善を進める人材育成が TPS の狙いである。TPS におけるムダの定義は以下である。作業の中には、作業者の動きに着目して、①付加価値のないムダな作業、②付加価値はないが必要な作業、③付加価値作業、に分けられ、全作業に占める付加価値作業時間の割合は 5~15%程度と小さい。組立作業を例にとると、組立部品が前工程から流れてくるのを待つ作業は、手待ちのムダ。組立部品やビスを取りに行く作業は、ムダとは言わないが付加価値作業ではない。付加価値作業は組立作業により、モノに何らかの形状変化を与える作業ダケである。TPS では、ムダを 7 つに分類する。①造り過ぎのムダ、②手待ちのムダ、③運搬のムダ、④加工のムダ、⑤在庫のムダ、⑥動作のムダ、⑦手直しのムダ、である。非アジャイル型のソフトづくりで、仕様づくり、ソフト設計、プログラミングなど仕事の細分化、ロット化、多重下請け構造により生ずるムダの多さは、理解できるであろう。前工程で分厚い仕様やソフト設計のドキュメントはムダとは言わないが、ソフトづくりの顧客価値はオブジェクトプログラムだけである。モノづくりの TPS では、ムダを排除してリードタイム短縮するが、リードタイム全体の中に占める付加価値時間の割合は、ベストプラクティスでも 1/200~300、通常作業では 1/2000~3000 との過去のデータがある。ソフトづくりの納期短縮へアジャイル型の有効性、ウォーターフォール型ソフト開発の非生産性は理解できるであろう。

TPS 導入の成功要件は、トップのコミットメントを前提として、①理想(あるべき姿)の追求、②現地現物(現地に行って現物で確認)、③5 回のなぜによる真因の追求、④まずは行動すること、⑤設備導入や IT 化の前に仕組み改善、である。生産工程においては、ジャストインタイム、自動化のほかに、かんばん、平準化、アンドン、ポカヨケ、一個流し、多能工、シングル段取り、リードタイム短縮、タクトタイム、小ロットサイズ、セル生産、U 字ライン、工程短縮、自己完結作業など、TPS で引用される多数のキーワード、ツールや方法論で実践される。

1.4 TPS は、ICT の技術革新と米研究者の貢献で、TPS/リーン方式へ

TPS の基本原則は今から 30 年前の 1980 年前後に確立した。その後トヨタが GM との合併により米国で自動車の生産をスタートした 1984 年の NUMMI から、米国を中心に広まった。当時、日本を含めた優良製造業の強さを調査研究し、景気低迷していた米製造業の復権を期待してまとめられた著書「Made in America」によれば、優

良企業の活動パターンは、①QCD(品質、コスト、納期)の同時改善、②顧客との密着、③サプライヤーとの密接な関係、④階層、部門数の少ない組織、⑤革新的な人材育成方針(継続的学習、チームワーク、参加意識、学習する組織)が上げられている。さらに James Womack らによる“The Machine That Changed the World (1990)”、“Lean Thinking (1996)”、さらに 2003 年に Jeffrey Liker による”THE TOYOTA WAY”の出版などを契機に、世界中でTPS/リーン方式は、競争力向上のデファクトスタンダードとして知られるようになった。30 年前に確立した伝統的 TPS が米国の著書の影響を受け、さらに ICT(情報通信技術)で進化していった。筆者は米国の研究者の TPS 進化への貢献に敬意を表して、TPS/リーン方式と表現している点を理解頂きたい。トヨタの創業からの先人の実践を通してできあがったモノづくりの思想哲学、TPS の原理原則と具現化のツールや方法論を学び、US の研究者がリーン方式と命名。暗黙知の多い TPS を明示知化し、TPS を競争優位のビジネスモデルとして展開可能にした US の研究者やコンサルタントの貢献は評価したい。

2. TPS の基本「人間性尊重」は、ソフト開発の「人のモチベーション向上」の原点

2.1 アジャイル生産方式から、ソフトウェアのアジャイル開発プロセスへ

80 年代に日本のものづくりの強さ、TPS を米国の研究者がリーン方式と命名し、その後、アイアコッカ研究所でポスト・リーン方式として、90 年代初に Agile (俊敏な、迅速な)生産方式が提唱された。しかし、Robust、Virtual corporation、Reconfiguration、Mass Customization などのアジャイル生産方式のコンセプトは、リーン方式に代わる顕著な成果もなく、生産の分野でアジャイル方式の名は死語になった。一方、人間性尊重を基本とする TPS/リーン方式の思想哲学、方法論は、米国におけるソフトウェア開発に取り入れられ、XP、Scrum など、現在のソフトウェア開発における非ウォーターフォール型の開発手法がいくつか提案され、以下、2001 年にアジャイル宣言 (Manifesto)としてまとめられた。

1. プロセスやツールより、個人との対話を重視する
2. ドキュメントより、動くソフトウェアを作ることに専念する
3. 契約交渉より、顧客との協調に力を入れる
4. 計画を厳守することより、変化への対応を優先する

2.2 ウォーターフォール型は人のモチベーション、ソフトの QCD 向上を阻害する

筆者らは、「人はいかなる時に仕事に意欲、満足を感じるか」について、生産現場の作業員に対してトヨタ社内でのアンケートおよび日米で類似調査を実施してきた。その結果、生産現場で働く作業員のモチベーションは、①自律化、②責任感、③達成感、④適正、⑤仕事そのもの、⑥向上心、⑦評価、が欧米の同様な調査でも共通にあげられる。生産システムにおいてもソフト開発プロセスにおいても主役は人。人のモ

チベーションを上げることがマネージャーの大きな責務である。ソフトづくりの生産性は個人のスキルと“やる気”に左右される。アジャイル型では、数人のチーム活動による2週間~2ヵ月の短いスパンでのユーザ要件定義からソフトづくり、リリースまで実施する。この間での朝会、チーム内コミュニケーションの活性化、ペアプログラミングなど、適度の緊張感と相互研鑽、アジャイルプロセス実践の各種の手法(例えば XP の 12 のプラクティス)は、ウォーターフォール型に比べて格段に人間性尊重を強調する TPS の原理原則にマッチしている。

誰がソフトづくりを工業化と称して、ソフトウェア工場などと命名し、現在の多重下請け構造のソフト開発プロセスにしたのか、を非難するつもりはない。しかしソフトづくりは、同じ人工物とは言え、建物やプラントのインフラ事業と同じ構築手法を導入するには無理がある。システムづくりがトップダウンかつヒエラルキー構造で、仕様の凍結を期待すれば、必然的にウォーターフォール型になる。その結果、システムができあがる時には環境が変わり、仕様の手戻りは多く、プロジェクトの70%以上は失敗、という過去の調査データは理解できる。ソフト開発に、PM手法としての米国流のPMBOKやScrum、XPなどのアジャイルプロセスの手法もさることながら、日本のものづくりの強みである車や電子機器などの商品開発、TQMなど日本のマネジメント手法をソフト業界の人達は学ぶべきである。とは言え、日本のものづくりのノウハウからできたPM手法が公開されていない現在、エンジニアリング業界のPM手法からできたPMBOKや日本のP2MIに学び、「守破離」という言葉がある様に日本のソフト業界が独自のアジャイルソフトとそのPM手法の確立を期待したい。

現在のICT社会進展の二つの中核技術である、マイコン(μ CPU)と光ファイバの発明者は日本人であることを知る人はほとんどいない。トヨタで70年代からロボット開発、大規模な制御情報システム構築でミニコン/マイコンと光ファイバを多用してきた筆者は、ビジネスは確かにアメリカに学ぶ点が多いが、技術的な面で「ITはアメリカ」と思った事は日本のIT業界の人ほどない。しかし、コンピュータで世界をリードしてきたIBMを筆頭とする米IT業界が、トップダウンかつヒエラルキー構造のシステムアーキテクチャと開発スタイルを情報システムの構築に普及したことは事実である。悪く言えば、狩猟文化、略奪文化の欧米の民族性を継承し、トップダウンかつヒエラルキー構造のビジネススタイルと、皆で助け合うという農耕文化の日本の民族性、TPSのボトムアップかつフラット構造を基本にした日本的経営の違いを理解しなければならない。テクノロジーの導入は難しくはない。しかし、欧米文化が持つ支配者・被支配者の構図、ヒエラルキーな下請け構造に安住している業界構造の中で、社団法人ESD21は、日本でTPS/アジャイル方式を主流(メインストリーム)にする役割を担って活動しているが、改革には道遠し、と言えよう。「改革に反対する者は、せめて横で黙っていてくれ」との元経団連会長の奥田氏(90年代初のトヨタ社長時の語録)の言葉を思い出す。

ウォーターフォール型では、顧客の要件定義から完成したソフトのリリースまでのリー

ドタイムが長く、多くはソフト開発プロセスの一部しか担当しないことから、自律化、責任感、向上心、達成感など個人と組織のモチベーションを高めることは難しい。アジャイル開発プロセスにより、小グループで分割されたモジュールを仕様づくりからソフトのリリースまで一貫して開発する。それにより、ソフトづくりのリードタイムは短く、PDCAサイクルは速められ、個人の達成感や向上心を高め、顧客満足、QCDの同時改善を可能にできる。社会経済学者、E.F.シューマッハーの”Small is Beautiful”は、ソフト開発においてもTPS/アジャイル方式の妥当性を示唆している。

2.3 全体システムの俯瞰・鳥瞰と「見える化」は人の改善力を向上させる

TPS/リーン方式における生産・物流システムの構築では、システム全体の俯瞰・鳥瞰を重視する。トヨタの「見える化」ツールとして定着している方法として、A3一枚文化がある。これは、ある事業計画を提案する時に、たとえ数十億円の案件でも、上司含めて全ての関係者に、質、量、コスト、タイミングをA3用紙一枚で説明する書類作成技法であり、多くの人はA3用紙での説明訓練ができています。全ての関係者が新しいプロジェクトを推進するときに、A3一枚の資料でシステム全体の概要を情報共有し、各人の役割分担による詳細な業務執行モデルに基づいてプロジェクトは推進される。

TPSにおける生産現場の工程改善などでは「モノと情報の流れ図」をトヨタでは長年採用されてきた。USで提唱されたリーン方式では、仕事の流れの見える化ツールとしてVSM (Value Stream Map)が多用され、海外の改善現場のハウスでは壁一面に貼られたVSMを見る機会が多い。トヨタでは90年代以降に、生産・物流分野の見える化ツールとして、TLSC(Total Link System Chart)という技法が開発された。大規模システムでは、畳3~4枚の巻紙に、工程管理、工程図、制御図、情報システム図を一枚に描く。一枚の巻紙で全ての関係者で情報共有でき、システム全体の俯瞰・鳥瞰が可能。TLSCは多くの人の知恵の創出、チーム活動による改善を可能にする。

現在、ソフト開発におけるアジャイルプロセスの失敗談を聞くと、多くの場合、ソフト開発における全体システムの計画、全体のソフトウェア・アーキテクチャ設計の正確さの欠如の場合が多い。その意味で、ウォーターフォール型での設計者の全体システムの構想力、システムアーキテクチャの設計経験は極めて重要である。アジャイル方式とは言え、モジュール分割含めた基本計画は重視したい。悪い計画(Push)に基づく実践(Pull)はナンセンスである。TPSのかんばん方式はPushを基本にしたPull方式である。システムの全体構造、アーキテクチャ設計など計画(Planning)は、顧客の要件に合わせてできるだけきちんとやり、実行(Execution)段階では、変化にフレキシブルに対応する「Push & Pull」の考え方は、全てに通じる。

2.4 TPSは、お客様の計画(Push)に基づいた実需(Pull)による生産活動

ソフト開発はお客様の最初の要望(Push)に対する実際に必要な機能の実現(Pull)

のギャップが大きな問題を生む。お客様の期待する機能の40%以上のソフトは一度も使われないソフト、というUSでの調査データは多く引用される。顧客にとっても作成者にとっても使われない多くの無駄なソフトを創る、という現実是不幸である。ソフト設計においても当初計画、すなわち要求仕様(Push)と本当に必要な仕様(Pull)は合致しない。全体のソフトの基本構造(プラットフォーム、フレームワーク、アーキテクチャ)が決定され、必要なソフトの機能モジュールの製作は、お客様をソフト開発のプロセスに巻き込み、共同作業でソフトづくりを行う。必要な機能に優先度をつけて、お客様のPullによるソフト制作、リリースに徹するべきである。それによりソフト品質の不具合、機能の過不足はタイムリーに改善できる。

ウォーターフォール型のソフト開発における数ヵ月～1、2年後のフィードバック(不具合や機能変更)すなわちPDCAの遅れは、手直しのムダを生む。ソフトづくりが、お客様の業務内容を表す言語をコンピュータ言語への通訳作業と仮定すれば、短い文章を逐次通訳した方が誤解を生まないのは言うまでもない。厚いドキュメントはTPSにおける7つのムダの一つとは言わないが、「顧客にとっての唯一の価値は、動くソフト、オブジェクトプログラム」であり、分厚い設計図書ではない。ソフトは建物の様に外部から視解化可能な構造物、ハードではなく、創作者の知的資産であり、生き物である。「厚いドキュメントより動くソフト」、「計画を厳守するのではなく変化を受入れる」、「ソフトづくりは顧客との共同作業」は、アジャイル宣言の重要なコンセプトである。

2.5 「見える化」は、管理者のための道具ではなく、改善のための道具

TPS/リーン方式の基本哲学は人間性尊重である。そのためにはコミュニケーションが第一であり、コミュニケーションにより問題点を共有し、仲間意識を醸成し、改善意識を高揚する。コミュニケーションを活性化し、チーム活動による改善を進める第一歩は「見える化」である。TPS/リーン方式では生産現場でも設計現場でも「見える化」を強調する。「見える化」の例は前述の「モノと情報の流れ図」やVSM、TLSCなどが典型例である。生産現場では、職場の方針、工程図、作業標準、品質問題、改善提案、スキルマップなど様々の方法で「見える化」され、生産現場の見やすい場所に掲示される。生産現場の重要品質不具合の再発防止例を上げる。「さらし首」という表現で長年言われてきたが、品質不良部品を関連職場で全員が見える様に展示する。重要なことは品質不良を起こした人は絶対に責めずに、重要不具合を発生した不良品の展示で、再発防止を計る意識付けの徹底である。「見える化」は、管理者自身や彼らの上司に報告するための管理の道具ではなく、関係者による改善の道具である。

ソフト開発現場においても、全体システム、顧客の要求、機能優先度、作業工程と進捗、ムダと付加価値作業の分類、品質不具合、スキルマップ、改善提案などを見る化する。「見える化」する方法を全員で考案し、全員参加で改善する。TPSの基本は、お客様第一を基本とした全員参加による継続的改善である。全員参加で見える化

のツールを考案し、見える化により現状と問題点を共有し、チーム活動による問題解決する。TPS/リーン方式は人間中心システムであり、個人個人の能力を100%引出すための仕掛けづくりであり、チーム活動のベクトル合わせによる組織能力の向上を計る仕組みである。見える化は改善の道具であり、管理者の自己満足のツールではない点は強調したい。

2.6 ソフト開発プロセスに、TPS 実現の前提である多能工化を重視しよう

現在の伝統的なソフト開発におけるウォーターフォール型では、全体のソフト開発プロセスのある部分だけを担当し、その部分に特化した能力を期待する。人はいろいろな仕事を担当することで成長し、仕事の達成感と仕事の結果の良し悪しの反省により成長する。ウォーターフォール型を中心にしたソフト開発は人間性尊重に反し、TPS/リーン方式で定義している多くのムダを生む。すなわち、仕事を細分化して前後工程を、別の人、部門、会社に引き渡す際の仕事の切替え、スイッチングロス、伝言ゲームにより発生するムダは多数。アジャイル方式で同じチーム内で全てやれば、設計コストが数分の一、数十分の一で済むことを多く人は理解できるであろう。TPS では、生産性向上のために多能工を重視する。例えば機械工場では、作業者は旋盤、フライス盤、研削盤、NC 機など多くの工作機を操作できる訓練をスキルマップに基づき学ぶ。セル生産工程では一人が百数十の作業を覚えて組立作業をする。規模の大きなソフトとは言え、いくつかの疎結合の独立なソフトモジュールとして分割できる。分割されたソフトモジュールをいくつかのサブモジュール(タスク)に分割して、それを一人から数人のチームでシステムの要件定義からソフト制作、プログラムデバック、完成したソフトモジュールのリリースまでやる。TPS/リーン方式における一個流し、工程短縮、仕事の平準化、多能工化は、そのままソフト開発に適用できる。TPS/リーン方式を学んだ US の多くのアジャイル手法提唱者が、TPS のモノづくりの思想・哲学や方法論をアジャイル実践手法として体系化した事に敬意を表したい。

40 数年前コンピュータが企業に導入されはじめた時代を考えて欲しい。ある問題に対して、一人で問題解決のための要件定義やプログラム開発をやってきた。複雑な情報化社会になりソフトは大規模になったとはいえ、大規模システムはいくつかの小規模のシステムの集合体である。ウォーターフォール型の開発における基本計画を順守しながら、全体ソフトを機能分割し、お客様の引き(Pull)による優先度でソフトをつくる。ウォーターフォール型から徐々にアジャイル方式へ移行し、多重階層の下請け構造を少しずつフラット化する。これがソフト業界の構造改革への第一歩と筆者は信じている。筆者らは 80 年代に TPS の求めるシステム像を「調和型自律分散システム (Harmonized Autonomous Decentralized System)」と命名した。そしてそのモデルはオーケストラと説明した。指揮者が全体の流れを決め、全ての奏者が日ごろ鍛えた腕前を披露して最高のシンフォニーを奏でる。ソフトづくりを、仕様まとめ、ソフト設計、

プログラミング、検査、顧客へのリリースまで全てを小集団の人達、いくつかの小集団によるフラット構造で開発する。人間性尊重の本来の姿に近づける活動が、ウォーターフォール型からアジャイル型へのソフトウェア業界の構造改革のアプローチである。前述した狩猟文化、略奪文化、支配者と被支配者の層別というヒエラルキー構造の業界構造から脱皮して、人は皆平等、皆で助け合う農耕文化の日本的なソフト開発スタイルこそ TPS を原点とするアジャイルソフト開発手法である。

3. 最後に. …ソフトづくりの構造改革。しかし、アジャイル方式に甘んじるな

TPS は本来、生産性向上の手法と言うより、人的能力を向上し、その能力を最大限に生かすための、人と組織細胞の活性化が本質である。ものづくりとソフト開発の違いを議論する人も多いが、筆者は、全てのビジネス活動は「人間・機械系 (ICT 含む)」で成り立ち、生産性の差異は技術や方法論より、それを使いこなす人に依存する、と言う。TPS の本質は、「改善しつづける人間集団を創る」、とする人的能力であり、そのしくみづくりである。ものづくりはソフトづくりに比較して、CAD/CAM/CAE による設計の自動化、生産分野ではロボット化、FA 化で、機械系の進歩により生産性は向上した。一方、ソフト開発プロセスでは、人的資源に依存する割合は生産現場よりはるかに大きい。しかし、その生産性の向上はものづくりに比して決して高くはない。それは人間性尊重に反する仕事の細分化、ウォーターフォール型を前提にした多重下請け構造にある、と筆者は理解する。

ソフト開発現場は 3K(キツイ、キビシイ、帰れない)どころか 7K 職場と言う。明るい職場づくりがマネージャー最大の責務である。製造業が、T 型フォードという同じ型式の自動車を一月中造っていた時代と同様に、ソフトづくりにおけるウォーターフォール型開発は、100 年前の車づくりのフォード方式と同じく過去の遺物である。「要件定義」、「システム仕様」、「ソフト設計」、「プログラミング」、「検査」などを分業して同じ仕事ばかりをやり、成果物は“最終顧客の価値ではない”ドキュメントと言う。自分がやったシステムの結果がどうなっていくかは分からない。PDCA による反省の機会は少なく、反省と改善によるスキル向上機会は少ない。人間性尊重を基本にモチベーションを上げて人の能力を最大限に生かす、アジャイルソフト開発プロセスへの転換は歴史的必然と言える。しかし、アジャイルソフトは、TPS における、人のモチベーションを上げる方法論に過ぎない。ビジネス活動は全て「人間・機械系 (ICT 含む)」で実現されるがソフト開発活動も全く同じ。人間系(人間コンピュータ)により、顧客価値であるオブジェクトコード生成の生産性向上がアジャイルソフト開発の肝ではあるが、機械系(コンピュータ)を駆使したソフトウェア工学に立脚した自動化手法により、従来法より格段に生産性向上を計れるソフト開発システムの開発・普及を筆者は期待している。

例として、日本発の「ユニケージ法」とイスラエルの「Sapiens」を概説する。前者は當仲氏(USP 研究所)が開発した、Unix 上のコマンドとシェルスクリプトによりシステ

ム開発手法である。古い話ではあるが、筆者には以下を想起させる。マイコン(ワンチップ CPU)が 1971 年に世に出て、70 年代後半にラジオシャック、コモドール、アップルなど US 勢の PC に続き、NEC が PC8801 でコンピュータを大衆のモノにした。その後、1980 年にソード(後に東芝が買収)が PIPS という表計算システムを世に出し、原価計算、給与計算などを PC 上で可能にし、一世を風靡(Windows 対応不可でフリーソフトに)した。ユニケージ法は Linux 上でコマンド文と定義文(シェルスクリプト)で、従来の 1/10 程度の短いプログラムで、短期間のソフトづくりを可能にし、30 年前の PIPS と同じ。もう一方の Sapiens は、ウォータフォール型のソフト開発方法から脱皮したアジャイル型で、かつプログラムレスのソフト開発を可能にする DOA(Data Oriented Approach)による設計手法。筆者は「なぜこのようなシステムが IT 業界に普及しないのか」と不思議でならない。ユーザもベンダーも人月稼業の古い商慣習に満足するのなら、日本のものづくりの衰退に繋がる。多重下請け構造によるビジネス慣行のそれぞれの立場で、受注元ではなく最終の顧客満足、顧客価値を提供しているか、ムダを排除してリードタイムを 1/2~1/3 短縮できないのか、真剣に検討する時期にきていると思う。本論の主題であるウォータフォール型からアジャイル型への転換だけに甘んじるのではなく、もっと積極的にシステム開発のコンピュータ化、自動化に取り組むべき、との思いは、トヨタでコンピュータの発展の歴史と共に歩み、国の IT 化施策など IT 化推進役を 40 年以上経験してきた筆者の思いである。

日本では茶道の世界からできた「守・破・離」という言葉がある。最初は先人に学び真似をする。次にその作法を改善し、新たな作法・方法論を確立する、という意味である。日本で生まれた伝統的 TPS は人間系としての TPS/リーン方式、TPS/アジャイル方式に進化してきた。ソフト開発の抜本的改革には、人間系に加えて IT によるソフトの機械化・自動作成のシナジーにより、従来法の 10 倍~100 倍の生産性向上を計れる新しい TPS/アジャイルソフト開発手法の展開を期待したい。 (以上)

<黒岩恵 略歴>

- ・ 1969 年、九州大学修士(電気・制御)卒、トヨタ自動車(株)入社。
- ・ トヨタで、ロボット含む生産技術開発、工場建設と新車開発に伴う大規模な制御情報システム(PA、LA)の開発。FA(Factory Automation)、トヨタ生産方式の IT 化、物流改善に関わる。
- ・ 2000 年より 5 年間、トヨタのトップを補佐して電子商取引推進協議会(ECOM)、経団連などで「e-Japan 戦略」、IT/EC 関連の非営利団体活動。
- ・ トヨタ退社後は名工大客員教授、国連大学「中部 ESD 拠点」などで NPO 活動に従事。
- ・ 現在、(社)持続可能なモノづくり・人づくり支援協会会長、九工大客員教授。NPO 法人「ものづくり APS 推進機構」理事長、(社)TPS 検定協会理事、COXEC 理事、中部 ESD 拠点推進会議顧問など非営利団体の活動に従事。



Agile2011 で「TPS の基本とアジャイルソフトへの適用」のテーマで招待講演、国内から 10 数名参加。US の皆さんが日本のモノづくり、リーン方式 (TPS) を学び、XP、SCRUM、Kanban などのアジャイル手法を提唱しているが、日本のソフト関係者が TPS を真剣に学び、日本的経営環境にマッチした TPS/Agile 方式の開発を期待したい。



Agile2011
(8月ソルトレークシティ)



Agile2011
(8月ソルトレークシティ)

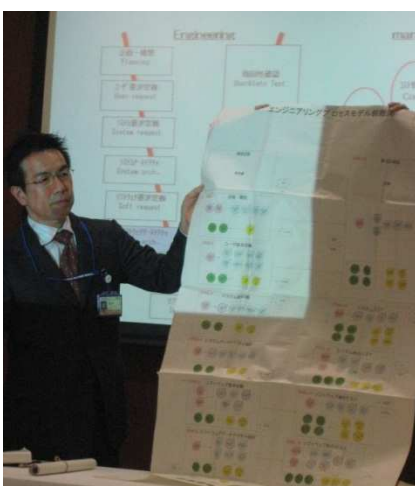


Agile2011
(8月ソルトレークシティ)

Kanban board and daily standup meeting were introduced in early February 2007 making the urgency visible

- More personal responsibility and accountability
- Resulted in better visual control
- Enabled more self-organization
- Less management supervision
- Better productivity
- Visual board and physical presence at standup generated better team collaboration
- Spontaneous quality circles and frequent Kaizen events

A
M
A
G
I
N
A
L
E
G
E
M
E
N
T



Toyota uses "waterfall" for software



2008 年、平鍋さん & ポップエンディーク 夫妻とデンマークのアジャイラー 10 名がトヨタ組込み系開発担当の BR グループ石井主査訪問。トヨタのソフト開発もウォーターフォール型。トヨタが、アジャイル方式の原点である TPS の本家本元のソフト開発プロセスは？との思いで訪問した皆さんが「がっかりしたかどうか」は平鍋さんだけが知る？