

INTERVIEW

永和システムマネジメント 平鍋健児氏, オージス総研 藤井拓氏 インタビュー

進化し続けるアジャイル, IoT時代の ビジネス・エンジニアリングに向けて

インタビュアー 山下博之 (情報処理推進機構), 藤瀬哲朗 (三菱総合研究所)

日本におけるアジャイル開発の先駆者であり、長年国内へのアジャイル開発手法の普及に努めてこられた、(株)永和システムマネジメントおよび(株)チェンジビジョンの平鍋健児代表取締役社長と、長年にわたりアジャイル開発手法のソフトウェア開発現場への適用を推進するための研究と実践を続けてこられた、(株)オージス総研の藤井拓アジャイル開発センター長にお話を伺いました。豊富な経験と知見に基づいた、アジャイル手法の要諦とともに、その普及にかける思いを熱く語っていただきました。

■「アジャイル手法」の導入状況

・国内の導入状況

山下 本日は、お忙しいところをお集まりいただき、ありがとうございます。今回の特集のテーマであるアジャイル開発手法については、私の属する独立行政法人情報処理推進機構 (IPA) のソフトウェア・エンジニアリング・センター (SEC) ^{☆1}において、2009年から4年間、調査・検討を行いました。そのときは、平鍋さんにもご協力いただいたのですが、当時はまだ一部のWeb系企業を中心に導入されている程度であったと思います。しかしその後、現在に至るまで、国内でのアジャイル開発手法の導入が徐々に増えてきているという印象を持っています。そのあたりについて、どのようにお考えでしょうか？

藤井 そうですね、アジャイル開発の適用が増えているという感覚はあったのですが、最近いくつかのデータを見まして、それが裏付けられました。

1つは、弊社でアジャイル関係のWebページを作っているのですが、その中の「スクラム」のページのアクセスがここ2年ぐらいで7倍、10倍ぐらいに急激に増えているということがあります。昔はアジャイルというキーワードで検索して、アジャイルって何というレベルで見られていたのですが、今はもうスクラムというキーワードで探されて、そういうページを見られているという状況になってきていますね。そういう意味でスクラムの普及が結構進んでいるかなという気がします。

もう1つは、先日、エンタープライズアジャイルというテーマで100名ぐらいのイベントを開催したのですが、そのアンケートによれば、5割以上の方がパイロットプロジェクトで進められており、1割ぐらいの方が複数プロジェクトでアジャイルを適用しているという回答でした。

平鍋 僕の認識も、自分の会社の状況から見ると、ずいぶん増えていきますね。ここ5～6年、10人以下の小規模チームのスタートアップや小さいWeb系サービスを提供している企業の中に入って開発するというのをずっとやっていました。しかし最近では、全体でいうと大規模なプロジェクトを細かく分割して、その一部をアジャイル開発で行うというケースも出てきましたし、逆に大きな会社から、アジャイル開発をやりたいのだけれどもどうやっていいか分からないから1回ちょっと話にきてくれとか、あるいはコンサルをしてくれみたいなことが結構多くなっています。大手のSIerからもそういう話がきていますから、ずいぶん市場で認知が進んだのではないのでしょうか。

山下 そのようにアジャイル開発手法の導入が増えている中で、分野あるいは業種について、何か特徴的な傾向はありますか？

藤井 そうですね、すごくよく知られているのは、eコマースとか、ネット系ですよ。楽天さんやヤフーさんといった企業が先駆者になっていると思いますが、ゲーム会社でも増えています。そこでは、たとえば、デザイナー系の人たちはウォーターフォール的に開発をして、ソフトウェア系はスクラムが適用されていて、その

^{☆1} 現、ソフトウェア高信頼化センター (SEC)

両者を混合しながら製品を作るというような感じで開発をされているところも結構あります。このようなゲーム会社の中には、大規模アジャイルをもうすでに実践されているケースがあります。

また、最近特に興味が増している、関心が高まっているのは、製造企業です。

平鍋 僕もWebサービス系がやはりすごく多いと思います。あと、最近組込み系でも聞くようになりました。

山下 また、別の視点として、システムのすべての部分の開発をアジャイル手法で行うのではなく、適材適所というか、システムの中でもアジャイル開発がふさわしい部分に適用するという考えられます。冒頭で述べた、以前にIPA/SECで行った調査・検討の結果として、最初にきちんとシステムのベースを作っておき、その上のサービス開発にアジャイル手法を適用するというタイプが整理できました。そういった事例はご存じでしょうか？

平鍋 アジャイル開発の利点が出る場所を切り出すという感じですかね。東京海上日動システムズさんの「あんしん生命保険」事例があります^{☆2}。基幹系は基幹系でちゃんとあって、そこから実際にペーパーレスで保険の申し込みをiPadを使ってやるので、そのところはユーザビリティが最も重要ということになる。その完全な要求仕様を作っても始まらないので、まず実際にソフトを作って、使ってもらって、いい使い勝手のものに上げていった方がいいでしょう、という要求の出元からくるプロジェクト特性というのがこのアジャイルにマッチする場面が多くなってきているのではないですかね。

・導入理由

山下 そのように、さまざまな企業がアジャイル開発手法を導入するようになったのは、どのような理由からと見ておられますか？

藤井 製造企業の例でいいますと、グローバル競争が激しくなり、海外製品や海外企業に対する競合が増えてきているときに、ニーズを聞いてそれをプロダクトに反映するまでのサイクルが従来のウォーターフォールだと遅すぎて、たぶん負けてしまうということから、開発手法を変えなくてはいけない、という危機意識が増えてきているような気がします。たとえば、海外のお客さんのニーズを仕様書にまとめて、それをウォーターフォールの

藤井 拓氏

(株)オーグス総研技術部アジャイル開発センター長。1984年京都大学理学研究科博士前期課程修了、2002年京都大学情報学研究科博士後期課程指導認定退学。1990年(株)オーグス情報システム総研(現(株)オーグス総研)に中途入社。ソフトウェア開発プロジェクトの測定、アジャイル開発を含む反復的な開発手法やモデリングの実践、研究、教育や普及に従事。エンタープライズアジャイル勉強会実行委員会委員長。認定スクラムマスター、SAFe Program Consultant、技術士(情報工學部門)、博士(情報学)。



に開発していたら、1年や2年かかるのではないですか、それを反映するまでに。それだともう、売上げがどんどん落ちていくという状況になるということだと思っております。Web系ですと国内でも競争が激しいので、そこで勝たなくてはいけないという状況があります。Web系以外では、グローバルの市場の中でそういうことが起きてきつつあるということだと思います。

平鍋 僕は二面あると考えています。すなわち、ビジネスの要請と、作る人のモチベーションが最終成果物にすごくかかわっているということ。

なんていうのかな、要求仕様書に書かれたものをそのまま作るみたいなどころより、本当は何がしたいんだっけといったような議論をしながら作る方が、作る側のモチベーションも上がるし、作る側が知っている最新技術のことも多くて、これだったらこういうふうにできますよという提案もできる。つまり、アジャイル開発では、「何を作るか」ということと「どう作るか」ということを、時間やフェーズを分けて、分断して考えるのではなく、会話をしながら、より良い問題発見と解決、問題提起と解決・提供をつなげられるという特性がある。その結果、働く人のモチベーションや工夫が、作り方や要求にフィードバックされていくことになる。作り方でなくて考え方がアジャイルに変わっていくこと、組織やプロジェクトマネジメントの変化が、すごく日本に必要なと思いますけれどもね。

このようなモチベーション、あるいは、良いチームワークのチームを作るという問題は昔からありましたが、ソフトウェア開発のところではアジャイル手法がすごくいい回答を持っているということが分かってきたのだと思います。

山下 藤井さんは数年前から「エンタープライズアジャイル勉強会」を運営されていますが、その趣旨はどのよ

^{☆2} アジャイル型開発で実現。業界初のペーパーレス「らくらく手続き」を開発。
<http://www.fujitsu.com/jp/group/fwest/resources/case-studies/2014/2014-04.html> (2016年5月25日現在)

うなことでしょうか？

藤井 「エンタープライズアジャイル勉強会」には、アジャイルをやると楽しいとか、自分たちの開発者冥利に尽きるような開発ができるとか、そういうところがアジャイルの魅力では確かにあると思うのですが、かたや、もう少しアジャイルが普及するためには、マネジメントの立場からアジャイルというものをきちんとと理解して使っていこう、自分たちのビジネスに活用していこうというふうに理解していくことが大事なのではないかと考えて取り組んでいます。ターゲットは中間管理職の方です。経営者の方は、イノベーションとか、元気がある人材を育成したいというような気持ちがあり、その主対象である若い人には、アジャイルのような新しいことをやりたいという意識があります。しかし、中間管理職の方は、アジャイルって何？ ということや、従来と違うのだけれどもその違いをどう吸収すればいいのかとか、どんなメリットがあるのかということをつかみ切れず、わりと抵抗したり反対したりすることが多いと思うので、そういう方々に情報を提供したいというのが勉強会の思いです。

突き詰めていうと、たとえば、中間管理職の方がリーダーシップを取っていくのは大事だと思いますが、その人たちにとってアジャイルをやるとするのはビジネス上の動機がないとおかしなことになってしまうので、それが第一です。会社をどうしたいという話はもちろんその先にあると思いますが、そこから入るとか、アジャイルをまずやらなくてはいけないというところから入るのではなく、ビジネス的に変化に対応するような開発をしなくてはいけない、この分野で競争に勝つためにやらなくてはいけないという意識から取り組む方がいいのではないかと思います。

それで先ほど申し上げた、あの製造企業などに、まさにそのような意識が芽生えつつあるのです。

平鍋 エンタープライズアジャイルに関しては、僕自身は一概には賛成できないですが、アジャイルであっても、エンタープライズアジャイルであっても、さらに言えばウォーターフォールであっても、それが今作っているものにいい方向に働けば、それはいいことだと割り切っています。つまり出自で判断せずに、それが現在のやり方より良い方向に向かうか、ということだけを問うべき。そういう意味で、エンタープライズアジャイルがうまくいけば、それはいいことだと思います。つまり、今まで小さいチームでやっていたアジャイルはうまくいきましたが、それが会社の中でほかの部署、たとえば、マーケ

ティングの部署とどうかかわりますか、運用の部署とはどうですか、ということになります。製品を作っている会社であれば、ポートフォリオ管理とはどうかかわりますかとか、ユーザ企業側であれば、ITの投資対効果を測っているはずですから、そことどうかかわりますかとか、そういう視点は必ず必要になります。そういう意味で、アジャイルチームとリアルな企業の現状の部署との接合点がうまく見つかるというか、解決できることはいいことだと思います。

・導入に際しての工夫

山下 さて、アジャイル開発を導入する企業にとっては、その文化というか風土というか、具体的には、その企業の従来の開発や管理のやり方、たとえば、プロジェクトの進捗やプロダクトの品質等の管理方法、経営層への説明方法等との整合が課題の1つになると思われま

平鍋 はい、特に大企業でアジャイルをやるときには、今までの企業のルールの中に入れていかなければいけません。たとえば、年次の予算や評価制度、アウトソースやオフショアはどうするのかという話、品質管理など、今まで企業がやってきたことの枠組みには、アジャイルのやり方は簡単にははまらないので、誰かが苦勞してその隙間を埋めなければいけないのです。既存の仕組みとの接合点が必要であり、そこでみんな苦勞しています。

藤井 その1つとして出てくるのが、契約の問題です。ただ、それは結局、やる必要性が弱い、やる／やらないという話になっているので、形式的なところで引っかかっているのだと思います。実質的にやらなくてはいけないのであれば、そこはマネジメント担当が中心になって変えていく、ガイドラインなどの必要なところを変えればいいだけの話だと思います。

実際、機能規模で開発量を定義し、実際の機能は規模の範囲内で変更可能という形で契約して開発した例があります。たとえば、開発量を100ファンクションポイントとして契約し、その100の中身はどんどん開発途上で変えていってもいいという形です。ただ、こういう事例もあるのですが、なかなかそこまでやるというお客さんの決断がないケースが多いです。したがって結局、実際には準委任で契約されているお客さんが多いと思います。

なお、米国で契約の話聞いたことがありますが、3カ月期間ごとに契約を更新していく形が多いようです。最近、米国政府も契約形態をかなり柔軟に変えてきているということも聞きます。

山下 私としては、契約は問題なのですが、ビジネスを

うまくやりたいと思ったら、ビジネス成功の優先順位が高いならば、契約の話は工夫により何とでもなると、極端かもしれませんがそう考えています。

平鍋 というのは僕の昔の立場ですね。でも、まあ、契約は重要ですから（笑）。

もし、ソフトウェアが、仕様を書いて、発注して、同じ納期で一番安いものを選ぶように「調達」できるものであれば、おそらく一括請負発注のように、これを作ってくださいということはたぶんできるのでしょね。この場合、発注時点で全体を定義しなければいけないので、要求を止めないといけないことになります。

しかし、そういうタイプではない、つまりコモディティ化していないソフトウェアというのもどんどん増えてきているのは間違いないと思っています。つまり、一括発注がしにくいソフトウェアが増えていることになります。ということはやはりサービスとして発注するしかなく、契約の種類でいうと準委任というのが日本では一番やりやすいことになり、実際、弊社のアジャイル開発の契約はかなり準委任契約でやっています。それに追加して、弊社独自の、価値創造契約という、Win-Winを狙う契約もあります。

山下 発注側のユーザ企業に準委任への抵抗がなければ、その方がいいと思います。

平鍋 そうですね。ところがユーザ企業にはシステム開発の知識がほとんどないということが多くあるので、それが難しいですね。舵取りを誰がするかということが問題となります。弊社でやっているアジャイル開発では、通常発注元にプロダクトオーナーになってもらいます。

山下 技術的な面では、今回の特集号の論文の中にも、それぞれの企業の状況に応じた工夫の事例がいくつか含まれています。たとえば、ハイブリッド型のプロセス、プラクティスの取捨選択とカスタマイズ、などです。具体的には、スクラムのイテレーションの回数を固定的に決めているという事例はありますし、最終的なリリース確認のためのイテレーションを最後にもう1回設定するという事例もあります。

平鍋 それはよく聞きますね。そのように、企業ごとに、その既存の枠組みとうまく整合するような工夫をしながら、アジャイル開発手法をうまく導入されてきているものと思います。

■ アジャイル手法のためのメトリクス

山下 既存の枠組み、管理方法との整合ということの1つとして、通常、ソフトウェア開発の管理のために、さまざま

なメトリクスが使われます。アジャイル開発の場合には、どのようなメトリクスが使われるのでしょうか？

藤井 アジャイル開発手法自身にメトリクスが組み込まれているので、従来手法よりも手軽に定量化しているという側面はあると思います。たとえば、私どもでは、バックログ項目の開発規模を見積もるために「ストーリーポイント」というメトリクスを使っています。また、開発スピードを「ベロシティ」というメトリクスで測っています。さらに、継続的インテグレーション（CI）をやりながら、欠陥の数や密度、テストのカバレッジといったデータを測定しています。これらは、アジャイルのフィードバックループ、PDCA（Plan, Do, Check, Act）のサイクルをまわすときに結構大事になっています。そのようなことが一般的に行われていると思います。

平鍋 僕たちも、ベロシティなどはプロジェクト管理のために測ってはいます。しかし、ソフトウェア開発プロジェクトは1件1件違うので、そもそもすべて同じ工程を切ってやろうという考え方はまったくないので、このお客さんと話をして重要な品質指標を決めて、それを見ていきましょうということです。だから、メトリクスや定量的管理について画一的にあるいは全体的に何かやっているということはありません。ベロシティも簡単に2倍、3倍になり、プロジェクトで全然違います。

山下 IPA/SECでは、多様で幅広い業種・業務から収集した多数のソフトウェア開発に関するプロジェクトデータを収集し、生産性や信頼性の観点で整理・分析した「ソフトウェア開発データ白書」を定期的に発行するなど、ソフトウェア開発の定量的管理の普及に取り組んでいます。データ白書は、ベンチマーキングなどに活用されていますが、現状では、ウォーターフォール型開発のデータしかありません。アジャイル開発が結構普及してきているので、アジャイル開発のデータについても収



平鍋健児氏

1989年東京大学精密機械工学科卒業、同年NKK入社後、エクサにて3次元CAD、Design Spinnaker開発に携わる。1995年、福井にUターンを決意、福井の（株）永和システムマネジメントに入社。田舎での受託開発を続けながら、オブジェクト指向開発、アジャイル開発を推進し、UMLエディタ astah*（旧JUDE）を開発。2006年、astah*を世界マーケットに展開すべく、（株）チェンジビジョン設立。代表取締役社長就任。2015年（株）永和システムマネジメント代表取締役社長。著書『アジャイル開発とスクラム～顧客・技術・経営をつなぐ協調的ソフトウェア開発マネジメント』等多数。

集・分析し、その結果を公開できれば役立つかもしれない、と考えています。そこで、アジャイル開発にはどういったメトリクスがあるのだろうか、議論を始めたところでした。

藤井 私どものところでは従来から、一般のプロジェクトでは、さまざまなデータを収集し、それらの比較も行っています。しかし、アジャイル開発の場合、先ほどお話ししたようなベロシティについていえば、その対象ストーリーポイント（開発規模）を揃えるような仕組みはないので、各プロジェクト内での一貫性はあるものの、プロジェクト間での比較はできないのですよね。

そのため私どもでは、機能規模を測定するなど、プロジェクト間で比較できるような工夫を行っています。そうしてみると、生産性についていえば、従来手法とそんなに変わらないかなという感想です。むしろ、作る中身、たとえば、より有効な機能を作ることによって同じエネルギーが違った形で、違ったものを生み出すわけではないですか。そこまで考えないと本当の評価をしたことにならないと考えています。

したがって、アジャイル開発の場合は、基本的にある程度定量化されている状況もあり、プラスアルファで何か計るということに関しては、ちょっとオーバーヘッドと思われるところがあって、難しさはあると思います。

平鍋 先ほども述べた通り、one size fits allの考え方ではできないと思いますね。できないという発見が僕はアジャイルだと思っているので。

うちの会社では、プロジェクトが成功するか失敗するかが最重要であり、それは生産性や信頼性というより、たぶんテストがちゃんと通っているかということと、それからお客さんがそれをちゃんと見てくれているかということによって左右されるので、ほかにどんなメトリクスを使ったらお客さんを納得させることができますかということですけれどもね。

実際、弊社ではプロジェクトの失敗は減少しています。アジャイル開発によって、失敗しにくくなっているといえます。それは生産性や品質管理に秘密があるのではなく、顧客との信頼関係ですね。それから、顧客側が舵取りをしてくれているかとか、顧客の参画度合いとか、そういったことのほうがより効きます。

■ アジャイル手法のためのツール類

山下 アジャイル開発をうまく（成功裏に、効率的に）進めるためには、ライフサイクルを通した管理（ALM）とそのためのツールの効果的活用が必須と考えますが、

日本国内では、ウォーターフォールでもアジャイルでも、ツール活用についてあまり積極的ではないように、個人的には感じています。アジャイル開発におけるツールの整備状況、活用状況などについて、ご紹介いただければと思います。

藤井 実装レベルのツールやプロジェクト管理の



山下博之

ためのツールはよく使われていますよね、CI、テストフレーム、配置の自動化ツールとか。ツールは、昔に比べてかなり使われるようになってきていると思います。特に、オープンソースのものが結構多いと思います。

ただ、テストに関しては、特にシステムテスト、自動化というところにおいては、もっと投資しないといけないという状況にあると思います。

山下 そうなのですよね。（一社）日本情報システム・ユーザー協会（JUAS）の調査によれば、アジャイル開発手法ではウォーターフォールに比べ、工期は短いものの、工数とコストがかなり大きいという結果になっています。この原因をいろいろ分析してみると、テストにツールがあまり使われていないということが分かりました。要するに、各リリースの前に必ず全体のテストをひと通りすると思いますが、それを人手でやっていたら大変ですよと。そこをうまく自動化することによって工数をかなり削減できるはずですよ。

藤井 テストのツールが入りにくい理由の1つは、プロジェクト単位で収支を考えるからだだと思います。テストの自動化にはプラスアルファのコストがかかりますが、1つのプロジェクトでそのコストを取り戻すのはかなり難しいので、保守まで含めて先行投資みたいな形で費用を捉えないと、踏み切るのは難しいと思います。楽天さんがテスト自動化をうまく進めているのは、そのサービスが継続して存在するという状況の中で、そのサービスをどんどん発展させなくてはいけない、かなりスピーディにやっていかなくてはいけないということから、テスト自動化が必須だということが理由だと思います。

このように、クラウドまわりとか、そういうところではテストの自動化はかなり必要なのですが、そうではなくて、オンプレミスというか、従来のエンタープライズ

システムについては、テストのためのツールが入るのは難しいというのが現実だと思います。

平鍋 ツールというとちょっと語弊があるのですがけれども、テストの自動化は必須ということにはすぐ同意しますね。

最近、VersionOne社のアジャイル開発に関する調査、この調査は10年続いているものですが^{☆3}、レポートを見てすごくびっくりしたのは同じことです。テストングの普及率が案外低いのですよね。アジャイル開発のプラクティスに「ユニットテスト」というのがありますが、この採用率がなんと63%となっています。こんなあり得ないだろうと思って、何でなのかな、というのが僕はすごく気になりました。これはちょっと低すぎますよね。この調査の回答者が、昔はピュアにアジャイルをやっている人ばかりだったのが、最近では大企業の中の回答者が増えてきていることから、このような結果が出ているのではないかなと思っています。

藤井 かつて少人数のチームで小さなアジャイル開発を行っていたときには、プロジェクト管理用のツールが必要といった意識はなかったと思います。最近になって、人数としては40人以上とかのチームで開発することも出てきたときに、そのプログラム全体の状況を見るにはそれなりのツールが必要です。現状は、一部ではそのレベルまでアジャイル開発がスケールアップしているものの、その規模のアジャイル開発のプロジェクトがまだ比較的少ないので、全体的には見えてきていないということはあると思います。

私は、スケールド・アジャイル・フレームワーク(SAFe)についてずっとやってきています。これは、企画を承認するポートフォリオレベル、複数チームで連携してプロダクトを作るプログラムレベル、普通のスクラムとかを実行していくような1つのチーム単位のチームレベルの3層で、アジャイル手法によってプロダクトを展開していくための大きな組織の動きを表現しているフレームワークです。2012年に1.0版が導入されて以降、年々バージョンアップし現在では4.0版になっています。4.0版でシステムズエンジニアリングがサポートされ、ポートフォリオレベルとプログラムレベルの間にバリューストリームレベルという、複数のバリューストリームを統合して開発してプロダクトを作るための層が追加されました。海外ではSAFeはかなり使われている

ようですが、日本ではほとんど使われていません。楽天さんくらいでしょうか。また、日米に分散する開発部門を有するある日本企業が、SAFeを両方で適用して、かなり大きなプロダクトを作っているというようなケースもあります。この例では、米国側が主導して実践されています。

山下 私は、ツール活用の未成熟さが、日本の労働生産性の低さ、ソフトウェア開発における生産性の低さの一要因となっていると考えています。このあたりについて、どう思われますでしょうか？

藤井 私たちの感覚からすると、実際に測定をやっているので、生産性は低くはないのではないかと思います。ただ、価値を生み出すという観点ではちょっと低いところはあるかなという気はします。たとえば、仕様書通りにモノを作るという点では生産性はそこそこあるのですが、本当に現実に市場や業務の中で必要になっているものが効率的に開発されているかといえば、そうでない部分があり、そういう意味での効率性の悪さというのはあるのではないかと思います。仕様書通りに100のモノを作っても、その製品やサービスを市場に投入したときに売れなかったら、結局無駄になって、価値を生み出していないという状態になってしまうわけですよ。

海外の企業では価値を生み出す効率が高い開発をされているのではないかと思います。生産性ということについては、そういう観点で考えたらいいのではないかと。

山下 なるほど。プロダクトの量ではなくて質に対するコストというか。

藤井 はい。だからかなり無駄になってしまっているのではないかなと、日本の開発は。

あとは、たとえば、分業が行き過ぎているという面もあるかもしれないですね。どうしても、段階的に開発、それぞれの工程ごとにきっちり作業するというような風潮になり、作業をきっちりやるのが目的になってしまうことから、それぞれにコストがかかってしまい、同じ費用なのにアウトプットがあまり出てこない、というように。

それとちょっとずれるかもしれないのですが、日本では、実際の開発が親切すぎるということもあるような気がしています。過剰に作っているというか、お客さんからの要求があるときに、何でもかんでもその要求を受け止めて作ろうとする。この点、海外ではわりと、本当に必要なのですかとドライに判断して、本当に必要なものだけを切り出していくという、まあ、先に述べた価値のところとつながるのですが、そういう意味でいうと、海

^{☆3} アジャイルの採用状況レポート2016の要約、
<http://qiita.com/kenjihiranabe/items/b9540696b3a0198fcec>
(2016年5月25日現在)

外では費用対効果ということをしビアに考えて、日本ほど手厚くシステムを作らないと思うのです。

山下 それは、アジャイルの考え方というか、価値志向というか、本当に役立つものを優先的に作っていくとか、そういうことですよ、ソフトウェア開発に限らず。

藤井 そうですね。シンプルなものからまず作って、本当に絶対に必要なものを作る。ミニマムマーケットブルフィーチャ (MMF) とか、そういう言葉で言いますが、最低限必要なものをまず作って、そこから必要に応じて機能を増やしていくというように開発すると、より無駄がなくなってくる、価値が高くなる。

■ アジャイル開発向きのソフトウェア・アーキテクチャ

山下 さて、このようにさまざまな対象への導入が進むアジャイル開発手法ですが、システム開発にアジャイル手法を適用するにあたり、システムのアーキテクチャ上の工夫も重要になってくるのではないかと考えます。個人的には、リファクタリングだけでは、限界があるのではないかと思います。また、最近では、「マイクロサービス」というアーキテクチャが注目されているようです。

この点については、さまざまな考え方があると思いますが、アジャイル向けのアーキテクチャについて、ご意見や動向、事例などをご紹介いただければと思います。

藤井 小規模のシステムであれば創発的に設計していくというスタイルもありますし、スタートアップからいきなりアーキテクチャをゴリゴリと作っていても、ビジネスが成り立たない、売上げが立たないのにそこだけ投資していくわけにはいかないという状況だと創発的にやらざるを得ないのではないかと思います。一方、ある程度大規模なシステムを作るのであれば、やはりアーキテクチャをしっかりと作っていかないと、修正コストとか、後でアーキテクチャ上の問題があったときのペナルティが大きいので、考えるべきだと思います。そういう前提で、既存のいろんな機能を組み合わせて作るというケースも多いと思うので、そういう意味ではマイクロサービスとか、そういうようなスタイルも使われていると思います。

山下 規模が小さいところからアジャイル開発手法が適用されてきたので、昔はアーキテクチャにあまり注目されていなかった、そんなには言われていなかったということでしょうか？

藤井 まあ、そうですね、あまり言われていなかったようなところはありますね。しかし、アジャイル開発の専門家

の間でも、アーキテクチャに対する考え方は多様です。

エクストリーム・プログラミング (XP) が創発的な設計ということで、アーキテクチャは考えなくてもいい、アーキテクチャは実装しながら徐々に形成されるものだというような考え方が一方であります。他方、アジャイル・アーキテクチャとか、そういう考え方できちんとアーキテクチャを考えるべし、という考え方もあります。ただ、アーキテクチャというどうしても難しい感じがして、誰か特別な専門家が作らないといけないという雰囲気がするので、チームレベルで小さい規模の開発をやっているときにわざわざそういう人を置いて考えるのですか、という話があり、前者の考え方のほうが、従来、多く受け入れられてきたのではないかと思います。

しかし、最近になって、企業が新しいビジネスを展開するときも、実際に既存のシステムを、いろんな組合せをして作ることが必要になり、たとえば、業務システムに新しい機能を追加するときもその既存システムとの連携がどうしても必要になってくるので、そういう意味でサービス志向とか、マイクロサービスとか、そういう考え方がとられるようになってきたものと思います。1つのアプリケーション機能が走るときにどんな相互作用がシステム間で起きるかとか、1つのターゲットの開発システムの中でどんな機能があるかというレベルでまず考えていくということが、論理的なアーキテクチャ設計のレベルだと思います。

平鍋 マイクロサービスは最近流行っていますね。これは、すごくアジャイルと合いますよね、この粒度で、もしチームが組めれば。そうしたら、大規模開発の必要がないのですものね。小さなサービス開発をやるということにできます。

コーンウェイの法則というのがあり、これは、製品のアーキテクチャはチームの組織構造と一致するというものです。たとえば、4チームでコンパイラを作ったら、4パスコンパイラになるという話ですけれども (笑)、つまり人のコミュニケーション情報流というものがアーキテクチャに非常に影響するという話なのです。ソフトウェアは結局、人間が作っているので、その人間の組織をまたぐコミュニケーションの流れというのがソフトウェアに非常に色濃く反映するし、それとまったく違う構造でやってしまうと、必ずうまくいかなくなるという話ですね。

ウォーターフォールは、要求を定義する人、設計する人と分かれているから、その組織構造と作り出すソフトウェアの構造がマッチするのです。だからコーンウェイ

の法則はまだ生きています。

最近、逆コーンウェイ戦略という作戦（笑）があります^{☆4}。製品のあるべきアーキテクチャに合わせて組織を再定義する、逆流だと思えます。分かりやすく言えば、DevOpsをやりたいなら、DevとOpsの組織を1つにすること、となります。

この文脈で、マイクロサービスというものは、大きな機能を小さなサービスに分ける、というソフトウェアの構造と、大きなチームを小さなアジャイルチームに分ける、という組織の構造が一致しているので、もしかしたら大きなブレークスルーになるかもしれません。

■ アジャイル手法の課題

山下 今後さらにさまざまな分野やより大規模なソフトウェア開発にアジャイル手法の適用が進められると思われそうですが、そこで課題となることがあるとすれば、それはどのようなことと考えられていますか？

藤井 技術的な話ではないですが、日本でアジャイル開発を普及させるという意味では、そういうことが本当に必要なビジネスモデルがあることがまず大事だと思います。次は、顧客と開発者との信頼関係ですね。産業構造の面で、ユーザ企業とSIベンダのように分離していると、長年、相互の信頼関係があまりない状態になっていると思います。そのために、契約とか、仕様書をきっちり書いて、要求が満たされているかいないかという形の開発の進め方をせざるを得ないということがあるのですが、そこも何かアジャイルを通じて信頼関係を少しずつ良くしていくことが大事なというふうに思います。

現実問題として、アジャイル開発をやるというときに、チームが組めないといけないので、その中にアジャイル開発の技術者が1人、2人くらい入っても、なかなか実践しにくいし、大きなものを作りにくいので、そういう意味では、やはり外部的な助けが必要になるのです。そのときに、本当にお客さんのニーズに則した開発をやるというような関係が、パートナー同士の信頼が、自分たちの業務を、分厚い仕様書がなくても割と軽量に伝えて、それで作ってもらうというような関係をこれから築いていくのが大事だと思います。

したがって、アジャイル開発が得意な、アジャイル開発にコミットしているような開発組織を社内に作り、お客さんのニーズにより応えた開発を実績として蓄えてい

くことによって、信頼関係という実績ができて、それが広がっていくのではないかと思います。アジャイル開発への注目が高まっているのであれば、SIerとしてはそういう組織を立ち上げて考えていく必要があるのではないかと思います。ビジネスですぐに利益が上がるかどうか、売上げが上がるかどうか、どうやって考えるかということは、難しいところがあるのですが。

山下 大手のSIerは、そういうようなアジャイル対応の部署を作っていると聞いています。あるいは、アジャイル開発向けのそういうルール、よくいわれる会社標準、社内標準みたいなものを一応整備してやってきていると聞いています。その一端が、今回の特集号に掲載されています。

平鍋 ソフトウェア開発ではなくて、アジャイルのチーム作りの方を、教育やハードウェア設計などに使っている人たちがいます。ゴールを決めて、そこに向かってイテラティブにいろんなスキルを持った人が集まって進んでいくという、まあ、イノベーションの手法としてというのがいいのかな。たとえば、ワイナリーが新しいシャンパンを作るということをスクラムでやっているのですよ（笑）。アジャイルジャパンで基調講演していただきますが^{☆5}、面白いなと思って、イノベーションを創り出すチームのマネジメント手法としてということ。

山下 その方がシステムティックであり、何か出てきそうな気がしますね、やり方として。

平鍋 そうなのです。だって、大きな要求を書いても仕方がないし、イノベーションを起こすときに先に要求を書きなさいといってもね。野中郁次郎先生によるスクラムに関する最初の論文がぴったりくるのはそこでしょうね。新製品開発の手法ですものね。世の中のないもの（新製品）、イノベーションをどう起こすかということですよものね。

■ 組込みシステムとアジャイル

藤瀬 さて、これまではエンタープライズ系システムを中心とするお話をしてきましたが、最初の方で少し話の出た組込みシステムへのアジャイル開発手法の導入状況について、もう少しお伺いしたいと思います。以前は、組込みシステム開発でのアジャイル開発手法の活用はあまり見られませんでした。最近はいかがでしょう？

IoTの時代、AIやビッグデータ解析等のキーワードが

☆4 逆コーンウェイ戦略とDEVOPS, MICROSERVICES, AGILE <https://anagileway.wordpress.com/2016/05/25/inverse-conway-maneuver-and-devops-microservices-and-agile/> (2016年5月25日現在)

☆5 Joe Justice 「スクラムがイノベーションを加速する～ソフトウェア以外にも適用されはじめたアジャイル～」 <http://www.agilejapan.org/program.html#keynote1> (2016年5月25日現在)



藤瀬哲朗

出てくる案件について、2～3年もかけて開発しても時代や環境も変化してしまっており、役に立つものになるようには思えません。組込みシステム開発でも高速開発技法が必要とされているのではないのでしょうか。

藤井 先ほどもお話ししましたが、最近製造

企業がアジャイル開発に非常に興味を持ってきているようです。IoTの時代、グローバル競争が激しくなり、海外では当然製造企業同士で競合します。現場のニーズを聞いてからプロダクトに反映するまでの対応サイクルが従来型のウォーターフォール開発では遅すぎ、競争に負けてしまいます。製造企業では、開発手法を変えていかなくてはならない、という危機意識が増加している気があります。組込みということよりは、付加的なサービスに差別化の価値があるともいえますが、

エンタープライズアジャイルの勉強会を主催していますが、参加者の内訳は大雑把に、3割がユーザ系企業ですが、同数程度の製造企業の方がいます。組込みシステム開発でも、アジャイル開発手法のフレームワークに興味を持っていただいている、チーム単位でまずスクラムの実践等も行われています。

平鍋 組込みシステムではテストの際、どうしても実機を必要とすることが多いため、アジャイル開発手法の適用が難しかったのですが、AUTOSAR^{☆6}のようなプラットフォームの登場により、モデルを活用した開発が可能になり、結果的にアジャイル開発の利用が進んでいるようです。フロントローディングとって良いのか分かりませんが、設計のリスクをこの段階で発見できるというのは、モデルによる開発ではよく聞く話です。

アジャイルとモデルは、元々はそんなに近いものではないです。アジャイル宣言に「動くソフトウェア(Working Software)を大事にする」という一文があります。ここは元々、Working “Code” だったものをSteve Mellorが“Software”と書き直したと聞いています。つまり、動くモデルも含めてWorking Softwareと

いうわけです。組込みシステム開発の実践の場では、どうしても実機を扱うことになります。ですから、ハードウェアが直接動かなくても、モデルでも(動けば)よい。こう捉えると、組込みシステムのアジャイルではシミュレーションがとても重要になります。

■ IoT時代に向けたアジャイル

山下 そろそろまとめたいと思います。かつてWeb系ビジネスの黎明期において(現在もそうかもしれませんが)、競争優位を保つためにアジャイル開発手法が採用されました。近い将来のIoT時代においては、アジャイル手法はどのような役割を担うと思われますか?

藤井 冒頭の方で述べたように、ビジネスでは、この時期にこれを出さないと競争に負けてしまいますよ、というような意識で考えます。今後、IoTが1つのブームとなって関連ビジネスの競争が激しくなったときに、どういうスタイルでシステムやサービスの開発を進めていくかということは、非常に重要になってくると思います。そういう意味で、アジャイル手法は、今後非常に重要な要素の1つになっていくでしょう。

平鍋 最初から要求を作れるという場面で、たとえば、大きな金融系システムのマイグレーションであれば、すでに稼働しているものがあるので、動かない要求というのが結構定義できる、変化しない要求です。

しかしながら、顧客の勧誘のためにこの新しいデバイスを使って、こんな場を作って、そこでお客さんにまず触ってもらおうみたいなことを考えると、要求を固めることは非常に難しく、最初は当然プロトタイプから固めるのでしようけれども、固めて作ってもほとんど意味がない。そうではなくて、第1版の動くシステムを早く作って、それをもとに議論しながら要求を変化させていった方が、実際に面白いもの、顧客のエンゲージメントが進むものができるケースが多いのではないですかね。それは、そのシステム・オブ・レコード(SoR)の外のシステム・オブ・エンゲージメント(SoE)に多く今後出てくると思っていて、その部分でアジャイル手法が決定的に効くのではないかと。

山下 IoT時代の関連ビジネスの黎明期において、アジャイル開発手法の適用は必須ということですね。そして、アジャイル開発手法をうまく適用するための工夫が重要ということでしょうか。

本日は、どうもありがとうございました。

^{☆6} AUTomotive Open System ARchitecture. 欧州の自動車メーカーや部品メーカーが中心となって立ち上げた組織であり、車載ソフトウェアの標準化に取り組む。