

Developers Summit 2012 参加レポート

技術部アジャイル開発センター 張嵐
ビジネスイノベーションセンター 水間 丈博

2月17日と18日、目黒雅叙園で開催された Developers Summit 2012 (通称「デブサミ」) に参加しました。デブサミは開発者向けの2日間の無料セミナーで、今年は「10年後も世界で通じるエンジニアであるために」というスローガンを掲げ、「Special」、「10th Special」、「DeNA Special」、「Global+Smartphone by GREE」、「開発プロセス」、「Web Technology」、「Mobile Technology」、「Cloud Development」「これからのアーキテクチャ」という9テーマ68セッション、「Open Jam」及びスポンサーの展示、コミュニティの展示を実施しました。ここでは、筆者が参加した「開発プロセス」テーマの中からいくつかのセッションについてレポートをします。

■目次

●17日 (1日目)

- [岩切氏の登壇](#)
- [\[16-B-1\]漆原茂「極上のSI戦略」](#)
- [\[16-B-2\]長岡実、高橋一貴「Yahoo! アジャイルクロニクル～コーチとエンジニアの体験記～」](#)
- [\[16-B-3\]和智右桂「教科書と現場のあいだ～学びを活かすために～」](#)
- [\[16-B-4\]平鍋健児「アジャイル開発の10年と今後を語ろう」](#)
- [\[16-B-5\]藤原大「アジャイルリーダーシップと組織改革～楽天のアジャイル開発というリアル～」](#)

●18日 (2日目)

- [\[17-B-5\]角谷信太郎「アジャイルマニフェストディケイド」](#)

■岩切氏の登壇

岩切氏はデブサミの産みの親ともいえる人物です。デブサミの10年間の歴史の中で、氏は800名も及ぶ登壇者を送り出しました。今回は岩切氏自らが登壇し、初めて参加者の前で発表しました。少々緊張気味ながらも、氏は自分が10年間続けてやっていることと、それができた理由について淡々と話しました。氏の日本の開発者数の減少に対する危機感、デブサミや出版を通じて、高いスキルを持つ開発者を日本で育てようという使命感、そして10年間一つのことを途切らせることなく継続する情熱に筆者は感服しました。

去年、筆者は初めてデブサミに参加しました。会場内に参加者であふれかえっていると、

その時一人の女性が椅子を一脚また一脚と運び入れ、参加者に提供していました。その優しい女性がまさに岩切氏ご本人であったことを、今回初めて知りました。小さいことでもコツコツと積み重ねれば、大きなものへと積み上がることを学んだ思いです。

■[16-B-1]極上のSI戦略

漆原茂（ウルシステムズ株式会社）

発表者の漆原茂氏はウルシステムズ株式会社の代表取締役社長です。氏はこれまでデブサミに複数回を渡って登壇しました。創業早期の泥臭い事例、それから会社の発展期を代表する格好良い事例をそれぞれ2回ずつ発表し、そして今回は自分の悟りとして、起業経験、経営経験から「極上のSI戦略」をまとめ、講演しました。

日本のSIerの受注開発モデルについて、最近批判の声があちこちで聞こえるようになっていきます。漆原氏も「人月の膨張＝会社の成長」、「無理受注」、「多重請負」、「稼働優先」、「利益優先」のビジネスモデルに問題があると指摘しました。経営者の覚悟として、案件の「取り方」、開発の「やり方」、会社の「人事評価」を変えれば、当初は赤字でも、最後にきっと良い結果につながると言います。以下漆原氏の論点を概要的にまとめます。

1. 案件の提案時、無理に取りに行かない

取ると決めたら、プリセールスに十分な時間をかけ、また提案内容で妥協しません。時間を惜しまずかけ、顧客の考えを超えるような提案をします。

プライム以外の案件、正しいメンバーがアサインできない案件、理不尽に値切られる案件、惜しいと感じても手が出すべきではありません。

2. 案件の実施時、プロパー純血主義でアサインし、先端技術にチャレンジし、生産性に拘り、顧客と一蓮托生となる

一つのプロジェクトにプロパーを3割以上薄めないという方針だそうです。品質と技術に拘り、効率の良いコードを求め、しっかりテストを行います。言われたことそのままの開発をせず、また、無意味な作業は全力で阻止します。

3. 採用は少数精鋭、人事制度で社員のチャレンジと成長をサポートする

評価は、顧客からの評価に加え、技術スキルと人間力を重視します。受け身な姿勢の人、チームの足を引っ張る人、成長を志さない人、チャレンジしない人については評価しません。

社員にチャンスを与えます。例えば、メディアの連載や書籍の執筆を薦めます。グローバルなベンチャーとの交流、コミュニティ活動への参加、勉強会への参加についても支援します。

ウルシステムズでは、こうした戦略によって、大規模の案件でも少数、精鋭で遂行できています。氏の会社は100人月の開発について、生産性は業界平均比2倍以上で、500人月の開発では、生産性は業界平均の5倍近くを達成できたと言います。また、経営上の数字として、売上的大幅の上昇がなくとも、利益率が向上でき、資金繰りもよくなります。

何より、顧客とパートナーが幸せになります。漆原氏は自社の利益の推移を見せながら、「極上のSI」への道は確かにあると断言しました。

■ [16-B-2] Yahoo! アジャイルクロニクル～コーチとエンジニアの体験記～ 長岡実、高橋一貴 (Yahoo! Japan)

「クロニクル」は年代記の意味です。アジャイルコーチの高橋一貴とエンジニアの長岡実氏はそれぞれの立場から、自社 Yahoo! のアジャイル開発の経験を披露しました。

「コーチ編」では、高橋氏はコーチングのやり方に触れ、社内アジャイル開発普及の条件、成功するためのやり方、そしてその結果について述べました。

1. アジャイル開発が普及できる条件：「人+環境」

アジャイル開発ができる人がすでにいるという状況下で、ウォーターフォール開発からアジャイル開発への移行を会社が推奨することで、アジャイル開発の導入と拡大が成功しやすくなるそうです。

アジャイル開発人材の用意について、いろいろな手段がありますが、Yahoo! の場合、アジャイル開発を得意とするベンチャー企業を買収することによって、必要な人材を獲得しました。

また、社内のアジャイル開発を推進するに当たっては、関連部署と積極的にコンタクトを取り、理解、共感、支持を得る必要があります。

2. 成功の小さな火を灯していく：「セミナー⇒成功事例⇒ルール化」

アジャイル開発の推進に当たっては、会社の文化や開発の文化の変化が必要とよく言われます。高橋氏は変化には時間がかかるということを理解した上で、まず社内セミナーを開催し、次にテストプロジェクトを探し、そして、成功に向け、プロジェクトの支援としてコーチとなる人を提供しました。テストプロジェクト終了後、その結果を社内向けに一般化しました。

3. コーチングのやり方：「教示⇒質問⇒委任」

高橋氏はプロジェクトとの関わり方について、教示⇒質問⇒委任という三つの段階を紹介しました。チームとして、コーチの「教示」に従う「守」の段階⇒コーチの「質問」によって考えて、自分なりに工夫する「破」の段階⇒チームとしてのやり方を固定し、コーチと「離」の段階を踏まえ、成熟していきます。

アジャイルコーチとして、高橋氏は社内に広くノウハウを普及させる仕組み、つまり問い合わせ体制を確立することで、プロジェクトとアジャイル開発を推進する部署が接触する機会を拡大することが大事だと言います。また、上長やエンジニアをセミナーやコミュニティ活動に巻き込んで、会社の外の環境とインタラクションすることも有効だと紹介しました。

4. 結果

テストプロジェクトを実施し、よい結果を得られました。プロジェクト生産性はスク

ラムの取り組み開始前と比べ77%アップしました。スクラムに起因する重大な障害の発生は0件です。

アジャイル版開発ワークフローを定義し、サポート体制の継続と資料の充実に力を入れました。社内では、ウォーターフォールとアジャイル開発を両立できるようになりました。

「エンジニア編」ではプロジェクトリーダーの長岡氏がアジャイル開発導入の理由とコーチの作用について紹介しました。

1. アジャイル開発導入の理由：「課題解決への期待+コーチのサポート」

長岡氏は「Yahoo News Topics」の5人からなる開発チームのリーダーです。この開発は、主に3人月規模の機能の改善と追加の作業から成ります。アジャイル開発の導入以前、個々人が黙々と開発していて、お互いに周りが何をやっているか分からないというコミュニケーションの問題があり、要求したものが期日に間に合わないのではという不安を常に抱えていました。ある時、高橋氏のアジャイルセミナーに参加し、「アジャイルコーチがサポートしてくれるならやってみよう」とトライし始めました。

2. コーチの存在がキー

アジャイル開発の導入により、全体状況が把握でき、残業も減るといった効果はすぐ得られましたが、1スプリントを1週間でこなすので、チームメンバーからは「しんどい」という声も寄せられていました。1週間という制約の中で余裕がなくなり、作業を完了できなかつたり、タスク漏れによるリリースミスが起きたり、などの問題も起こりました。この時、コーチからは「スプリントを2週間にすべき」というアドバイスが与えられ、その時初めて、問題の原因がスプリントの短さにあることに気がつきました。また、メンバーは新しいやり方に慣れ始めると、モチベーションも低下し始めました。この時、コーチと相談し、チーム主導で新サービスを考えるという対策を取りました。チームが突っ走っているなかで、単純なことにも気がつかなかったことがあり、それをアジャイルコーチが一步引いた目で見ながら、適宜ツッコミ、助言をしてくれたり、相談に乗ってくれたりするので、コーチの存在はとてものありがたいものを感じられたとのことです。

■ [16-B-3] 教科書と現場のあいだ～学びを活かすために～

和智右桂氏（グロースエクスパートナーズ株式会社）

グロースエクスパートナーズ株式会社のITアーキテクトである和智右桂氏は「エリック・エヴァンスのドメイン駆動設計」の翻訳者として知られています。

氏は文系大学院出身で、本の翻訳を経験し、また、開発の経験もあり、現場の現実も見てきました。そして、自分自身の体験で、学んだことを現場で活かすためにはどうすればよいのか、その模索の過程を踏まえて講演を行いました。以下、氏の話の内容の趣旨を簡単にまとめます。

1. 名著と呼ばれている本はやはり読む価値があるので、精読しよう。本を読んで、知っていることは重要です。また、機会があれば、原書の著者と接するチャンスも増やしましょう。
2. 知っていることとそれを実践することのギャップがありますが、原則を理解すれば、実践に幅が生まれます。例えば、おしゃれなプラクティスを試したい、その思いが大局観を失わせます。実践のため、原則を守りつつ、プラクティスの取捨選択を行わなければなりません。
3. 受託開発では顧客によって方法を変える必要があります、常に新しいコンテキストが待っています。JUnit、課題管理ツール、ワンクリックデプロイなど、実践性の高いやり方やツールを重視し、価値が高いと自分が信じているものを相手に伝える能力を培います。

氏の講演後に、質疑のやり取りの中で、「守・破・離」の話が出ました。和智氏は剣道を学んだ経験があり、その経験を生かした素晴らしい回答で会場の皆さんをうならせました。以下氏の回答の概要です。

守：師匠の「型」を覚え、その通りにやる。

破：師匠の「型」通りにはやらなくなる。

「型」通りにやっても、どうしてもできないところが出てくる。それは、自分の身体的特徴などが師匠とは違うから。経験を積み重ね、自分が師匠と違う部分を活かせるようになれば、それなりにいいところまで行ける。

離：自分の「型」を完成させ、いつの間にか師匠とは別の流派を作り上げる。

■ [16-B-4] アジャイル開発の10年と今後を語ろう

平鍋健児（永和システムマネジメント）

アジャイル開発宣言の誕生は11年です。永和システムマネジメントの副社長、株式会社チェンジビジョン代表平鍋健児氏は海外でもよく知られている日本のアジャイル開発の代表的な人物です。氏が海外でアジャイル関連の話をするとき、日本人だと分かたら、次の質問は「Do you know KENJI?」だったという逸話があります。

日本のアジャイル開発を牽引してきた達人として、平鍋氏の話は俯瞰的、かつ分かりやすいです。アジャイル開発の各プロセスの誕生時期、各プロセスの関係と進化について、氏は一枚の図を用いて時間軸に沿ってまとめました。また、名著（XP、リファクタリング、Lean、Kanban 関連）との出会い、アジャイル開発分野の著名人（Kent Beck、Mary Poppendieck、野中郁次郎、Jeff Sutherland）との関わり、本のタイトルや写真を見せながら、アジャイル開発がこの10年間歩んできた歴史を振りかえました。

今後について、氏の考えとしては、お客さまに価値のあるもの以外は全部無駄であって、これからは方法論よりはプラクティス重視とすべきとのこと。また、今後、ソフトウェア開発に求める理想な人材像として、知的体育系、実践知のリーダー、身体化された心

などと挙げました。

最後に、氏は悟りと禅の話をしました。福井県の永平寺に書かれた以下の言葉を引用し、悟りが目的でその手段が座禅であるという風に考えてはいけない、やることと考えることは分けてはいけないことを説明しました。

「知る」ということと 「わかる」こととは違うのです
知ってはいても 実行されなければ分かったことにはなりません
薬の効能書を読んだだけでは 病気は治りません
禅も実行して初めて分かることなのです

■ [16-B-5] アジャイルリーダーシップと組織改革～楽天のアジャイル開発というリアル～

藤原大（楽天）

前出のYahoo!の高橋氏の話が「メンタルコーチング」についてだとすると、藤原氏の話は「スキルコーチング」についてと言えます。

講演者の藤原大氏は楽天の標準化チームに所属しています。自分がコーチとして実践したこと、そして、ぶつかった壁とそれをどう乗り越えたかについて、時間軸にそって紹介しました。

藤原氏は2009年、自分が担当する3人のプロジェクトにXPを適用しました。その成功経験を社内に公開するとともに、継続的なインテグレーション（CI）や自動化といったアジャイルプラクティスの導入を推奨しました。

自動化の広がりには最初が早かったのですが、徐々に速度を失いました。そのプラクティスのどこかに特別難しい部分があるのかどうかを検証するために、新人チームに適用させても見ましたが、とくに問題点は見つかりませんでした。

次に、現場が忙しく、自動化に手が回らないのではないかと推測し、他の部署と連携し、CIのプロセスに品質管理担当（QA）を参画させながら、プロセスの改善や人材の育成を推進しました。しかしそれでも、どこか一体感が欠如していました。よくよく調べてみると、原因は組織構造にあることが分かりました。つまり、報告先が違いますので、担当者はどうしても目の前の作業を最優先しがちだったのです。そこで、自分のやっていることを整理して考え直し、新たな方向性を考えました。

まず、楽天の開発は工場的な要素が強かったので、ソフトウェア開発をあるべき「創造」の方に引っ張っていかねばなりません。

また、社内には、勘違いアジャイルや失敗アジャイルも色々ありました。軽い気持ちで導入し、失敗して「アジャイルは使えない」という結論が広まる前に、成功事例を作らないといけません。

まず、コーチとして現場に潜り込み、現場の悩みを理解しました。現場は新人や若手が多く、教育が行き届いていなかったのです。一方で、レガシーコードは山盛り状態です。

そこで、現場と話し合い、いくつかの対策を施しました。

1. 新しい技術「アジャイル+クラウド」を導入してみました。
2. 「朝礼+見える化」を実施しました。ルールを決めて朝礼を実施し、見える化によって自分に関係ありそうな所に耳を傾けるようになりました。
3. イベントに積極的に参加させ、その時は上司も連れていくようにします。外に出て行って、会社とのギャップをどんどんフィードバックさせました。

このように日々地道な活動を行った結果、開発のスピードが上って行きました。やっけて楽しく、問題があったらみなで解決できて、一体感があるといったポジティブな意見が聞かれるようになりました。また、プロジェクトに対する定量的な評価を行ったところ、良いデータが上がってきました。従来は、作業の優先順位付けがきちんと行われておらず、突発的な作業に振り回されることが多かったですが、アジャイル開発の導入後は、プロジェクトに集中することができるようになり、開発作業にかかる時間がプロジェクトスタート時点の6倍に増え、ストーリーの完成数は倍になりました。カバレッジ98%、バグは0.02%に成りました。

しかし、継続的な改善は簡単ではありません。コーチが抜けると、元に戻ってしまったケースもありました。そこで今度は、現場に足を運びながら、アジャイル開発を組織全体に広げるための仲間づくりにチャレンジしました。

藤原氏はアジャイルコーチとしては、ファシリテーションスキルが重要だと言います。現場の改善の熱意は人を変え、人を育てます。氏は、以下のアジャイルリーダーシップモデルについて、自分の思いを述べました。

- Servant Leadership 召使いのリーダー
- Self Organizing 自己組織化のチーム
- Flat フラット
- Empowered チームを勇気づける
- Accountable 責任がある
- Self Directed 自己管理
- Merit Based メリットベース

最後に、藤原氏は横須賀にある戦艦三笠のマストの写真をもって、チームを勇気づけるためのZ旗について説明しました。会場の参加者に「各員一層奮励努力せよ」というエールを送りました。そして、「アジャイルをやっていると社内にも仲間ができるが、それだけでなく、社外にも、海外にも仲間ができる」として締めくくりました。

■[17-B-5]アジャイルマニフェストディケイド

角谷信太郎（永和システムマネジメント）

講演者の角谷信太郎氏はアジャイルの世界では有名なエバンジェリストの一人で、永和システムマネジメントシステムに所属されつつ日本 Ruby の会でもご活躍されています。『アジャイ

ルサムライ –達人開発者への道- (オーム社)』や『アジャイルプラクティス (オライリー)』などの翻訳・監訳でも著名な方です。

講演は『アジャイルマニフェストディケイド』と題され、ちょうどアジャイルマニフェストが2001年2月に発表されて以降の10年を振り返り、吟味しつつさらに次代のアジャイル開発を俯瞰するといった内容でしたが、そのプレゼンの展開は余人を寄せ付けないユニークさに富み、斬新かつ深いものでした。

ウォーターフォール型開発が“人工のソフトウェア”とすれば、ハードウェアや要素技術が進化した現在、“自然のソフトウェア”とは何か？これが実践されていないとすればそれは“不正義”ではないか？と言う問いから始まりました。そこへ建築家 Christopher Alexander に源を発する『パターン、WIKI、XP 時を超えた創造の原則 (江渡浩一郎著)』の紹介が。アレグザンダーのパターンによる建築プロセスの特性

- ・少しずつ成長させる
- ・人間の感性に深く基づく
- ・繰り返し実施される

これは、ソフトウェアの「正義」を実現する考え方ではないかと。

そこへ「仮面ライダーディケイド」(3年前-2009-に放映されていましたが、あなたはご存知でしたか?)になぞらえた“マニフェスト17人衆”が仮面ライダーの恰好で表現されて、会場大爆笑。ここで“The Nature of Software”の意味が深掘りされます。

「ソフトウェアの本質とは何だろうか？」と。その答えは

- ・人とソフトウェアの間に価値があること
- ・“システム”全体を構成すること
- ・変更に対応することが求められていること

これはそれぞれ

- ・人がソフトウェアを使って初めてその価値に気付くこと
- ・ハード、ソフト、文書、運用が重要であること
- ・育てることと技術的負債を認識する必要があること

に対応する、と説いていることが分かります。

最も力点が置かれていたのは、“人とソフトウェアのあいだに価値がある”ことでした。そしてプログラミングと言う仕事の価値を再認識させる主張が続きました。プログラムの創造は“スキルを備え世界観を構築してその世界を具体的に作り出す創造的な仕事であり、単純なお仕事ではない！”と。コードを作る人々にプライドと自信を与えてくれるメッセージです。併せて、“良いコードはコード自体がドキュメントであることを十分認識することこそが、コードをクリアにするために重要なのだ”、と言う Martin Fowler の言葉が紹介されました。角谷さんのメッセージは「コードにしたものとコードにしなかったことがプログラミング」であるというものでした。

そしてまた、「人工のソフトウェア」と「自然なソフトウェア」に立ち返ります。「自然

なソフトウェア」は「自由なソフトウェア」なのではないか？それをどのように創るのか？アジャイルソフトウェア開発の“開発”に目を向けると、「そこには実行する主体と実行することが偏在しており、そこにはプロセスが存在する、プロセスとは創ることであり、細胞分裂のように合成・組立・分化が行われている」ことが示されます。James Coplien の「パターンに機能と型を取り戻す」が紹介され、アレグザンダーのパターンが再び想起されます。

では、“アジャイル”とは何か。それは「名詞じゃなくて形容詞なんだ、身軽に気軽に行動するってことなんだ」と悟らせてくれます。結局“アジャイル”とは、「アジャイルであることを示す度合いに過ぎないのだ」と。角谷さんは“アジャイルさとはプロセスがどれだけ生き活きしているかを示す度合いである”と言う独自の定義を示します。そこには色々なフレーバー（XP、SCRUM、FDD、Lean・・・）があるが、特に SCRUM は“習得は非常に困難！”と言う Jeff Sutherland と Ken Schwaber の言葉も示されます。これって、SCRUM の否定？

いや、そうではなく、『手法や名前にとらわれてはいけない！！』と主張していることに気付かされます。つまり大事なことは何か？アジャイルマニフェストの一節が再度示されます。

“顧客満足を最優先し価値のあるソフトウェアを早く継続的に提供します”

開発をアジャイルにできない理由はいくらでも存在するけれど、それを目指して開発することが自然なソフトウェアを創ることなんじゃないの？ってことなんでしょうか。そのための支援材料としてなのか、もう一度「組織パターンとプロセスパターン」が紹介されます。

そして最後に、「次の 10 年を走って行け！傷だらけでもこの世界の未来は君の手にある！」と勇気を与えられました。

180 枚に及ぶプレゼンで聴衆を楽しませ、飽きさせず、驚かせ、勇気づける、そして深いアジャイルソフトウェア開発の本質に切り込んだ見事な講演でした。

皆さんも是非「角谷さんワールド」(http://codezine.jp/devsumi/2012/session_data)を堪能してみてください。