



連載

ビブリオ・トーク
—私のオズメー—

… 長野 徹 (日本アイ・ビー・エム (株))

アドレナリンジャンキー
プロジェクトの現在と未来を映す 86 パターントム・デマルコ, ピーター・フルシュカ, ティム・リスター, スティーブ・マクメナミン,
ジェームズ・ロバートソン, スザンヌ・ロバートソン 著, 伊豆原弓 訳
日経 BP 社 (2009), 280p., 2,200 円 + 税, ISBN : 978-4-8222-8401-5

アドレナリンジャンキー, とは聞きなれない言葉ですが, とにかく忙しくてアドレナリンが出ていないと気が済まないアドレナリン中毒の人を指します。計画や戦略を立てることなく, ただ忙しいことが(無意識に)好きな人のことです。今は昔ほど「ひたすら忙しく」働くことが推奨される時代ではなくなりましたが, 今でもそのようなタイプの人がいるかもしれません。この本は, このような, プロジェクトをうまくいかなくする「プロジェクトあるある・プロジェクトいるいる」を 86 個のパターンとして抽出したものです。「アドレナリンジャンキー」は本書のタイトルでもあり, この本の第 1 番目のパターンとしても紹介されています。

ソフトウェア関連で「パターン」といえばソフトウェア開発の設計ノウハウを集約した「デザインパターン(原著: Design Patterns : Elements of Reusable Object-Oriented Software)」が有名で, ファクトリ, アダプタ, シングルトン, など多くのプログラマが模範とすべき「良い」パターンが広く知られていますが, 本書で紹介されているパターンは, 筆者らのソフトウェア開発におけるプロジェクト管理の観点から見た「ひどい」例を抽出し, パターン化したものです。

筆者の 1 人, トム・デマルコ (Tom DeMarco) 氏は「ピープルウェア(原著: Peopleware)」「デッドライン (The Deadline)」「ゆとりの法則 (Slack)」「熊とワルツを (Waltzing with Bears)」といったソフトウェア開発に関する著書でよく知られています。本書「アドレナリンジャンキー (Adrenaline Junkies and Template Zombies)」はトム・デマルコ氏としては初めての 6 人の共著となっています。本書では, プロジェクト遂行の妨げとなるものは何か, プロジェクトのメンバをどのように理解すればよいか, 円滑に

プロジェクトを進めるにはどうすればよいか, 情報の共有をどうすればよいか, どのように時間を管理するのがよいか, どのようにメンバを評価すればよいか, といった事柄について示唆を与えてくれます。示唆を与えてくれます, と書きましたが, この本はあくまでもダメだった「あるある」「いるいる」パターンの集合で, どうすればよいかといったことはほとんど書かれていません。では, 本書ではどのような「あるある・いるいる」パターンが書かれているかを, いくつかご紹介します(以下の節のタイトル前にある数字は本書の先頭から「あるある」に振られている番号です。なお以下の説明は私の解釈によるもので必ずしも筆者の意図と一致しているとは限りません)。

45 ニュースの改良

チームリーダー「おそらく 1 月は無理です」

プロジェクトマネージャー「正直言って 1 月では不安です」

アプリケーションマネージャー「1 月というのは難題だと思えますが…」

CIO「1 月には間に合うと自信を持ってご報告できます」

皆嘘をつきたいのではないのですが, その役職に応じて当人の都合の良ように情報が変わっていくのは世の常です。まだ取り組んでいない内容だとしても, 苦し紛れに「今はじめたところです」「どう取り組むか考えています」「いまちょうど良いところです」「もうできる感じですよ」というように改良してしまうこともあるでしょう。

62 隠れた美

輻輳の概念とその処理方法は, 少なくとも私には予

想外だったが、すばらしくシンプルである。笑われるかもしれないが、私はイーサネットの仕様書を見て胸がいっぱいになった。

感動、とまではいなくても、よく見ると無駄のないソースコード、必要最小限の美しいサンプルコードなど、一度は出会ったことがある方も多いのではないのでしょうか。最近では git などを通じてソースコードを見る・見られる機会も増えてきたと思います。3カ月後に再度コードを見直す自分のためにもきれいなコードを書くに越したことはありません。

26 かかし

クライアントは、実物を見るまで、そして「これは違う」と思うまで、自分が何を欲しいのかわからない。

相手が何を考えているかわからない。何も考えていないのかもしれない、と感じることもあるでしょう。特にクライアントのような見る角度が違う人と話をする際に、話がかみ合うようになるまで時間のかかることがあります。相手の求めているものと違うものかもしれないけど、形のあるものを見せるとうまく情報を引き出せるかもしれません。

53 データエラーの真犯人

データベースソフトの品質より、処理するデータの品質のほうが劣っていることはめずらしくはないのだが、エンドユーザーの目から見れば、数少ないほうがシステムの品質低下の原因である。

入力データのフォーマットが正しくないためエラーが発生する、入力データの質が良くないため、期待した結果が得られない。どちらもツール（ソフトウェア）の問題ではないのですが、多くの場合、データの改善よりもツールの修正・調整に時間がとられることとなります。

77 パイクリングオン

新しいプロジェクトをダメにしようと思ったら、公然と対立するような危険なまねをする必要はなく、代わりに「プロジェクトがそのすばらしい約束を果たせるように」と称し、数十個の追加や改良を提案する。

はじめはシンプルで美しいアイデアでも、いろん

な人の意見を聞いてあれもこれもと追加しているうちに、かけ離れたものになってしまいます。

85 リーク

初期の業務から難しい作業がリークすると、あとになるほど難しい作業が増え、95パーセントまで完了したところで、5パーセントよりずっと長い時間立ち止まることになる。

「大体終わった」から「終わった」に至るまでの距離がいかにか長いかは皆様もよくご存知かと思います。「大体終わった」は大体半分くらいだと思っていた方がいいのではないのでしょうか。

どのパターンも実例とともに解説されており、読みやすくなっています。もちろんすべての「あるある」に共感できる人もそんなにいないと思いますが、プロジェクト経験の長い人ほど共感できるパターンが多いのではないのでしょうか。

本書では「はじめに」も読んでいただければと思います（ほんの3ページです）。大体「はじめに」を読まない方もいらっしゃると思います。工学書だと「近年のインターネットの発展により……」であったり、「本書をまとめ上げるにあたりお世話になった諸先生方の……」であったり、読み手にとっては、あまり意味のある内容がないことが多く、さっさと中身を読んでしまいがちなのですが、気づきの例を「あるある」としてパターン化＝抽象化することの重要性、について書かれています。

プロジェクトマネジメントに関する方にはもちろん、ビジネス書は固苦しくて苦手な方、ビジネス書は面白いけど、CEO といった遠い立場の人たちの著書はちょっと実践には難しく…、といった方にも、身近な内容で実践可能な示唆に富み、一読の価値はあるかと思います。実例はソフトウェア開発ですが、同じようにソフトウェアをまとめてモノを作り上げていく大学の先生方にもオススメしたい1冊です。

(2017年2月1日受付)

長野 徹 (正会員) TOHRU3@jp.ibm.com

日本アイ・ビー・エム(株)東京基礎研究所にて、自然言語処理および音声言語処理の研究に従事。博士(工学)。