

連載 プロマネの現場から 第109回 トラブルプロジェクト再考

蒼海憲治 (大手 SI 企業・金融系プロジェクトマネージャ)

「遅れているソフトウェアプロジェクトへの要員追加は、プロジェクトをさらに遅らせる」というのは、『人月の神話』のフレデリック・ブルックスによる「ブルックスの法則」の一つです。その心は・・・単に人を増やすだけだと、既存のプロジェクトメンバーと増やした人数の合計の二乗分、コミュニケーションパス・コストとも増大する。また、新規メンバーが既存メンバー並みにキャッチアップするためには、学習時間が必要であり、既存メンバーからの教育が必要となる。ただでさえ忙しい既存メンバーの時間は奪われる。さらに、新規メンバーが習熟するまでの間は、作業品質は悪く、新たな不具合を作り込む恐れもあります。

しかし、そんなことは百も承知の上で、今日も、トラブルプロジェクトへの人の投入は行われているのだと思います。ここ数年、システム投資という面では景気が良かったためか、新しい案件が順調に立ち上がっているように思います。しかし、その一方、突然、プロジェクトがトラブルに見舞われる、バーストするというよくない現象も、周りでは増えてきているように感じています。

最近、大規模な増員にもかかわらず、プロジェクトの続行が困難となったケースをいくつか見聞きしました。今回は、単純な要員投入では解決しない、トラブルプロジェクトが構造的な問題を抱えた事例を紹介したいと思います。

トラブルプロジェクトについては、以前、『第50回 リカバリー・マネジメント』で、トラブルプロジェクトの実態把握とそれに基づいたリカバリー方針と実施手順について書いたことがあります。その際、人を増やすことで、既存のメンバーを支援し、立て直すことができるのであれば、トラブルレベルとしては軽傷のうちであり、ぜひ早期に増員等の手当てをすべきです。しかしながら、重症のプロジェクトであれば、単なる増員では問題は解決しないし、さらに悪化する恐れもあります。実際には、新しいプロマネをアサインし、リカバリー専用チームを組成し、別スケジュール・別予算で実行しなければ立て直せない、といたしました。

ところで、トラブルの予兆は、進捗遅れ・品質悪化懸念に現れます。

表面化した問題に対して、対策を講じます。過勤務・休日出勤対応、増員、そしてスケジュールの度重なる延伸になります。

でも、いつまで経っても、問題は沈静化せず、許されるだけコストが増大し、スケジュールが遅延します。どうしようもなくなってから、プロジェクトはバースト認定されるこ

とになります。この時点で、お金で解決するなら、実は一番簡単なのかもしれませんが。多くのプロジェクトは、ここで単純に人を増員しても、またスケジュールを延ばしても、解決しません。

以下、最近、見聞きした、海外システムの大規模再構築プロジェクトを、振り返ってみたいと思います。

・プロジェクトは、海外にある10拠点以上のシステムの共通システムの構築を目的としたものでした。

そのため、全海外拠点の要望を盛り込んだ共通システムを構築した上で、海外拠点へ順次展開していくという方針をとりました。

・この共通システムそのものの規模が大きいため、大きく2段階に分けて開発することになります。すなわち、共通システム構築の先行プロジェクトと本格プロジェクトの二段階で推進することとなりました。

・海外複数拠点に分かれた大規模システムを再構築することそのものの難易度が高い。しかも、それを共通化するという、一見して、非常に難易度の高いプロジェクトでした。

実際にプロジェクトを推進してわかった主要な課題は、以下のとおりです。

1. 業務要件定義が、いつまで経っても収束しない。

各国の拠点の業務責任者の発言が強く、既存の機能にプラスアルファの新規要件の拡充の要望が多発し、要件が肥大化した。

また、各国毎に要望事項は異なったが、拠点側の発言が強く、拠点全体を束ねる部署が要件集約の統制をきかせられなかった。

2. 共通システムが肥大化する。

業務要件を踏まえて、共通システムに盛り込むべき機能と、拠点個別機能とすべき機能の切り分けを行う必要があった。しかしながら、各国の拠点の業務責任者の発言が強く、共通システムに持ち込むべき機能が膨れてしまった。軽くスタートしようとした所期の目論見が外れ、先行プロジェクトそのものも、フェーズ分けする必要性がでてくる事態となる。

また、共通システムに盛り込む機能が増えたため、プロジェクト推進途上においても、

その機能に対する変更要件が増大し、修正作業負荷が極めて大きくなった。

3. 既存システムの業務要件・仕様が整理できない。

現行システムは、30年以上かけて、順次拡張されてきており、業務要件・仕様とも、習熟しているメンバーはほとんどいない。

それに加えて、現行のレガシーシステムには、現在稼働中のプログラムはあるものの、それに対応する要件定義書や基本設計書が整備されていない。

また、プログラムやプログラム仕様書はあるが、それを解析したとしても、システム仕様はわかるものの、業務要件の整理・理解には足りないものだった。

4. 本格プロジェクトチームを立ち上げようとしたものの、体制が組成できない。

当初、先行プロジェクトが開発工程に入ったタイミングで、先行プロジェクトの要件定義・基本設計を担当していたメンバーを、本格プロジェクトの要件定義にスライドさせることを想定していた。

しかし、前述の先行プロジェクトの要件定義が収束しないことにより、別要員を中心として体制を組成することとなる。しかしながら、既存システムも先行プロジェクトも知らない要員が中心のため、一向に進まなかった。

5. 多段階リリース及び移行システムの負荷が増大する。

当初、先行プロジェクトと本格プロジェクトの二段階のみでのリリースを想定していた。

しかし、先行プロジェクトそのものが機能増大にともない2ステップとなるとともに、1ステップだけをとっても、10拠点同時にリリースした場合、リリース直後に海外拠点に張り付き、即時にトラブル対応できる要員・体制が組成できないことが想定された。

そのため、2～3拠点毎の拠点グループ毎のリリースが必要となる。この多段階のリリースに伴い、現行システムとの整合性を図るための移行システムが膨れることとなった。

これらの課題に対して、実プロジェクトとしては、解決策として、100名以上の追加要員の投入がなされました。特に、プロジェクト内PMOの体制強化がなされ、進捗・品質等の「見える化」は、大幅に改善しました。

しかし、プロジェクト状況の本質的な改善がされず、厳しい状況が続きました。残念ながら、最終的には、先行プロジェクトを残し、並走させていた後続プロジェクトはいった

ん中断となりました。

トラブルプロジェクトをリカバリーするにあたっては、いったん中断し、計画を策定し直すことが、トータルで見ると、コスト面・スケジュール面ともに賢明なのだと思います。

しかし、社運を賭けた大規模プロジェクトであるほど、プロジェクトを継続しながら、並行して、トラブルのリカバリーを推進する必要があります。このリカバリーの主体が、別部隊であればよいのですが、既存チームに増員する場合、冒頭のブルックスの法則の世界が出現してしまいます。

今回のプロジェクトは、一時、プロジェクトを継続しながらのリカバリー活動でしたが、その後、中断して、計画を整理し直すという判断は正しかったと思っています。

ところで、実は、このパターンの失敗事例を見聞きするのは、初めてではなく何度目かになります。

トラブルプロジェクトは、大なり小なり構造問題を抱えています。その構造にメスを入れ、その原因に対応する要員を投入しない場合、人海は、塵芥と化してしまいます。

トラブルプロジェクトの場合、その実態把握そのものが非常に難しい作業になります。実態を押さえ、その原因を作りだしている構造を捉える。その過程は、仮説・検証の繰り返しであり、その結果を踏まえて、初めて有効な改善策を打つことができます。

今回のプロジェクトにおいては、たとえば、以下のような方針転換が必要だったのでは、と思っています。

- ・大上段に共通システム化を唱えず、1拠点のみのコンパクトなシステムを構築する。小さく始めて、大きく育てる、という発想を崩さない。
- ・懸案の収束しない要件定義を確定させるためには、声の大きい各国の拠点の業務責任者を、中央の要件とりまとめチームの一員にする。

私自身、現在、大規模プロジェクトのプロマネとして、日々発生し、累積していく不良を前にすると、増員をかけたい誘惑に駆られるのですが、問題の所在と、プロジェクトの構造を把握した上で、適切な手を打っているか、そして、それらは頭の中だけではなく、計画書に記載されているか、と自問自答することが大切だと思っています。