

連載 プロマネの現場から

第8回 タイム・マネジメントとコミットメント

蒼海憲治（大手 SI 企業・金融系プロジェクトマネージャ）

ここ数年、システム開発の現場においても他の職場と同様に、深夜残業・休日出勤の原則禁止や早帰りデーの設定等という時短の職場が増えてきています。その一方、プロジェクトの立上げ局面や、バースト気味のプロジェクトにおいては、「原則」の規定を越えて、依然として勤務時間のピークが続くことが往々にしてあります。特に、プロジェクトの立上げ局面においては、立上げ局面として求められる成果物の作成はもちろんですが、その後のプロジェクトの円滑な推進のための前準備である体制構築、標準策定、開発環境の構築などの段取りが必要となります。ところが、各事項について合意すべき対象である顧客や自社・協力会社などのステークホルダーの理解がなかなか得られず、時間のみが過ぎてしまうということが往々にしてあります。「時間切れ」は「負け!」と言い聞かせて叱咤激励するのですが、その頑張りどころのベースとなるのが、マイル・ストーンを踏まえたスケジュールであり、それに基づくタイム・マネジメントとなります。とりわけ、先の見通しが見えにくい段階こそ必要だと痛感しています。

そこで今回は、プロジェクト管理における様々な側面の中で、タイム・マネジメントについて考えてみたいと思います。

スケジュールやそれに基づく進捗管理を適切に行うためのプロジェクト・マネジメントの知識エリアとして、PMBOKにおいては、プロジェクト・タイム・マネジメントがあります。プロジェクト・タイム・マネジメントでは、プロジェクトを所定の時期に確実に完了させるために必要なプロセスとして、6つのプロセスを規定しています。

1. アクティビティ定義・プロジェクトにおける成果物を生み出すのに必要なアクティビティを洗い出す
2. アクティビティ順序設定・各アクティビティ間の相互依存関係を明確にし、文書化する
3. アクティビティ資源見積り・各アクティビティ作業にどのような資源（リソース）がどの程度必要かを見積もる
4. アクティビティ所要期間見積り・個々のアクティビティを完了するために必要な作業時間を見積もる
5. スケジュール作成・アクティビティ定義、アクティビティ順序設定、及び資源に対する要求事項を分析し、プロジェクト・スケジュールを作成する
6. スケジュール・コントロール・プロジェクト・スケジュールの変更をコントロール

対象とする成果物を作り出すための作業を、アクティビティとして洗い出します。最下層のアクティビティの粒度は、普通の人々が1週間働けば、進捗が目に見えるようにするため、最大4日間以内とするように分解します。複数のアクティビティ間の作業順序を明確にし、クリティカル・パスやボトルネックを把握します。過去の実績値やプロトタイプによる試行によって得た作業時間を踏まえて、アクティビティ毎の作業時間を見積もります。そして、スケジュールを作成します。表記法としては、ガント・チャートやマイルストーン・チャート、ネットワーク図等により記述します。

このスケジュールを基に、現場ベースでは、毎日、進捗把握を行い、遅れ・進みの状況をウォッチします。遅れのリカバリーに対して、プロジェクト・メンバーの増強やリ

スケジュールを必要とする場合、週次または月次ベースでの社内上長や顧客等へのエスカレーション等を図ります。

このタイム・マネジメントの実効性をあげるための個人レベルの活動を2つ紹介したいと思います。

1つ目は、作業時間の「記録」をつけること。

つい最近、ドラッカー氏の本を手にとった際、次の言葉が目飛び込んできました。

「成果をあげる者は、仕事からスタートしない。時間からスタートする。計画からもスタートしない。何に時間がとられているかを明らかにすることからスタートする。」
(* 1)

作業時間の実績を把握することの重要性・何にどれだけ時間を使っていることがわかって初めて、時間のリストラも、新しい業務にその時間を当てることもできるということだと思えます。

この作業時間の実績把握について、個人のプラクティスに着目して、ソフトウェア開発作業を行うための指針として、パーソナルソフトウェアプロセス (P S P : Personal Software Process) (* 2) というものがあります。

このP S P、元々は学生や新人を対象としているところがあるのですが、ソフトウェア技術者自身が、徹底して自分の実施した作業に関する作業時間の「記録」を取ることを求めている点が参考になります。

当初の計画で設定した見積りの値は妥当だったのか否か。自分の作業効率が良かったのか、悪かったのか。悪かった場合、どこがまずかったのか・等々、これらの分析や反省を行うためには「記録」が必要となります。また、作業にかかった時間とともに、「中断時間」を計測することも薦められています。作業と作業の間のロスや待ち時間がわかれば、作業の品質と効率を上げることができるようになります。

時間をおいて、「記録」を眺めてみると、「忙しい」と思っていた状態が、段取りが不十分であったため対応が後手に回った「気ぜわしい」ことであることがわかること。

また、「記録」の開始直後としばらく経過後では、自分自身の対象作業への習熟度合いが高まり、また段取り = プロセスが良くなるため、品質も効率も良くなっていきます。そして、しばらくすると、生産性が安定するという過程を通して、タイム・マネジメントへの感度が上がっていくようになります。

プロジェクト成熟度モデルにおいてレベル1未達の組織においては、実プロジェクト全体に適用することはちょっと無理があるように感じる方がいるかもしれませんが、作業工程毎にパイロット・チームのメンバーに意識して作業をしてもらうだけでも、効果的だと思います。

というのも、以前初めてプロジェクトの進捗管理に、E V M (Earned Value Management) を適用した際に、実績記録を取り、その結果を分析して驚いたことがあるからです。

それは、各工程の計画段階において詳細にアクティビティをブレイクダウンしW B S を策定していたと思っていたにもかかわらず、プロジェクトの在籍要員のトータル作業時間を足し合わせたトータル所要作業時間と、W B S のアクティビティのトータル作業実績時間との間に、2割から3割以上の開きがあったことでした。

つまり、2割から3割以上の時間が、WBSに記載されておらず、プロジェクト・メンバーは、WBSにないアクティビティを一生懸命実行しているということになるのです。

その主な原因は、

1. プロジェクトマネージャやチームリーダーのマネジメント工数の抜け・漏れ
2. 実作業員以外による成果物のレビューアの作業工数の抜け・漏れ
3. 他チームや顧客間の調整時間、また調整による待ち時間やロス

が、WBSとして管理されていないことでした。・・・そして、恥ずかしながら、これらはEVM以前の問題です。

ただ、この事実がわかったことにより、結果として、マネジメントのアクティビティやレビューのアクティビティのWBSへの反映、待ち時間やロス等の発生事由の実績への記述とリカバリー対策のフォロー等のアクションへつなげる、という工夫につながりました。

2つ目は、コミットメントの必要性です。

コミットメントとは「約束、誓約、関与」・・・目標に対する決意表明です。

責任を引き受けると、何か起こった時の対応が早くなります。人のせい、環境のせいにする習慣がついていると、それは誰かがやるはずだ・・・と、対応がワンテンポ遅れ、即応していればボヤですんでいたことが火事になってしまうことも起こりえます。

プロジェクトの推進上においても、顧客や上司等のステークホルダーに対する様々な約束・・・コミットメントが要求されます。

PSPにおいては、このコミットメントの中でも、ソフトウェア技術者自身が、自分に対して行うコミットメントが一番重要である、といます。ソフトウェア技術者自身が、「何を行うか」「それが行われたことを判定するための基準」「誰が行うか」「いつまでに行うか」「見返りとして与えられる報酬またはその他の対価」「誰がこの報酬または対価を提供するか」を明確にした上で、スケジュール案やその代替案を考え、コミットすべき相手に対して正式に合意します。

プロジェクトにおいて、ソフトウェア技術者自身は様々なコミットメントをすることになります。コミットメントに合意し、責任を引き受けた上で、そのコミットメントを適切に管理するマインドを持つことで、プロジェクト・メンバー一人一人のタイム・マネジメントへの感度が上がっていくと思います。

(* 1) ピーター・ドラッカー「経営者の条件」

(* 2) ワッツ・ハンフリー「パーソナルソフトウェアプロセス入門」