

ソフトウェア開発体制の戦略的構築に関する一提案

A strategy construction of software development organization

桂田 忠知†
Tadatomo Katsurada

大原 茂之‡
Shigeyuki Ohara

1. はじめに

近年ソフトウェア開発に対する要求が厳しくなっている。ソフトウェアの品質を維持したまま、開発費用の削減や開発期間の短縮を求められている。しかし、既存のソフトウェア開発方法論では厳しくなる要求を必ずしも満たせていない。

ソフトウェア開発方法論には多くのツールがあるが、開発体制を導入する必要がある。個別的に開発ツール等を導入しても全体的な管理運営ができなければ高品質かつ短期間での大規模開発は望めない。システム開発後に保守運用を効率よく行うことを含めて、コンポーネントベース開発が注目を浴びている。品質の高いコンポーネントを組み合わせることで、ソフトウェアの品質を維持できる。また、コーディングの手間を省けるので開発期間の短縮にもなる。しかし、コンポーネントベース開発にも問題がある。再利用しやすいコンポーネントを開発しようとする、余計な手間がかかり開発費用の増加や開発期間の長期化に陥る。そのため、要求を満たすために長期的な利益に繋がるコンポーネントの開発を後回しにしている。[1]よって、コンポーネントの数が増えず有効にコンポーネントベース開発を行えない。

そこで、本報告ではソフトウェア開発体制を戦略的に構築し、コンポーネントベース開発を行うことで、開発期間の短縮を行える方法を提案する。

2. ソフトウェア開発体制の戦略的構築

2.1 ソフトウェア開発体制

ソフトウェア開発体制とは、ソフトウェア開発及び開発方法論を含めた組織の在り方である。Fig.1 に戦略的に構築したソフトウェア開発体制を示す。

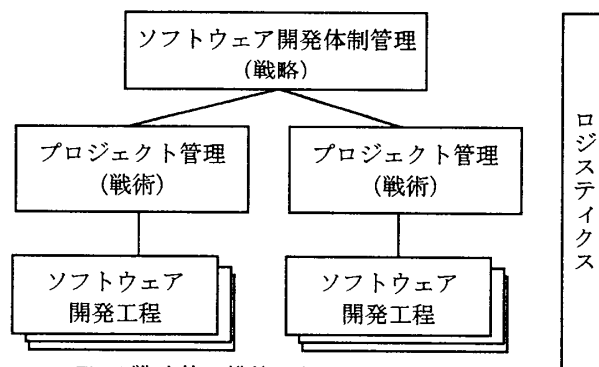


Fig.1 戦略的に構築したソフトウェア開発体制

戦略的に構築したソフトウェア開発体制は、ソフトウェア開発体制管理、プロジェクト管理、ソフトウェア開発工程、ロジスティクスで構成される。ソフトウェア開発の作業を目的毎に分けることで、コンポーネントベース開発を行う際の問題を解決できる。

2.2 ロジスティクスの分離

ロジスティクスとは、ソフトウェア開発体制の一要素であり、ソフトウェア開発体制管理以下の支援を行う。ロジスティクスのリソースには、開発者、開発環境、ドキュメント、情報、コンポーネント、開発資金がある。ロジスティクスの作業としては、ソフトウェア開発体制管理以下に対するリソースの割り当て、コンポーネントの開発、教育、開発環境の構築がある。

ここで言うロジスティクスは、ソフトウェア開発工程の中で行われていたコンポーネントの開発や教育などを分離したものである。このように分離した理由として、ソフトウェアを開発するという観点から見たとき、ロジスティクスはオーバーヘッドと考えられる。従って、ロジスティクスをソフトウェア開発から分離することでソフトウェア開発中の無駄を省くことができ、開発期間の短縮が可能になる。

ソフトウェア開発の時に作成されるコンポーネントは、ソフトウェアに合わせている。そのため、資源としては使い易いコンポーネントになっていない。そこで、コンポーネントを他のソフトウェアで使えるように、ロジスティクスの中で使いやすいものとして準備する。ロジスティクスでは、コンポーネント化やコンポーネントの新規開発、コンポーネントの入手がある。

ソフトウェア開発で行われている OJT も資源のコンポーネント化やコンポーネントの新規開発に伴いロジスティクスへ分離する。教育の一環としてコンポーネント開発を行うことで、ソフトウェア開発における教育にかける手間を省くことができる。

2.3 戦略的視点と戦術的視点

ソフトウェア開発体制に戦略と戦術の2つの視点を取り入れる。戦略的視点では全体を見て長期的に考える。戦術的視点では一部を見て短期的に考える。

戦略的視点から、ソフトウェア開発体制管理を行う。ソフトウェア開発体制管理では、組織全体としての体制管理を行う。複数のプロジェクト間の調整やロジスティクスの強化項目の決定を行いソフトウェア開発体制の最適化を図る。戦術的視点から、プロジェクト管理を行う。プロジェクト管理では、個々のプロジェクトを管理する。ソフトウェア開発工程の進捗管理などを行いプロジェクトの最適化を図る。ソフトウェア開発工程では要求定義、設計、プログラミング、テストを行い、ソフトウェアを開発する。ロ

†東海大学大学院工学研究科電子工学専攻

‡東海大学電子情報学部情報メディア学科教授

ジスティクスで準備したコンポーネントをソフトウェア開発工程で利用することでコンポーネントベース開発を実現する。

2.4 コンポーネントデータベース

コンポーネントの粒度や種類を増やすことで、ソフトウェア開発におけるコンポーネントの利用率を高めることができる。そこで、多種多様なコンポーネントを開発し、コンポーネントデータベースに蓄える。

コンポーネントデータベースはコンポーネントを用途や粒度で分類して蓄積する。コンポーネントを利用するために検索する時は、コンポーネント名と入出力、使用例、評価を表示する。コンポーネントを開発するために検索する時は、コンポーネント全体の評価と実装済みコンポーネント、未実装のコンポーネントを表示する。実装済みのコンポーネントにはコンポーネント名と入出力、使用例、評価、利用率を表示する。未実装のコンポーネントにはコンポーネント名と入出力、使用例、開発優先順位、開発する場合の費用、入手する場合の費用を表示する。

コンポーネントの需要があると未実装のコンポーネントとしてコンポーネントデータベースに追加する。コンポーネント全体の評価や費用、開発に回せる人員を考慮して開発優先順位を付け、順番に開発する。開発する場合の費用より入手する場合の費用の方が安い時もあるため、コンポーネントを開発せずに入手する時もある。

3. ロジスティクスの運用

3.1 コンポーネントの準備

コンポーネントベース開発では、コンポーネントの品質や種類が開発力に直結する。コンポーネントの準備には開発と入手の2通りの方法がある。Fig.2 にコンポーネントと情報の流れの関係を示す。

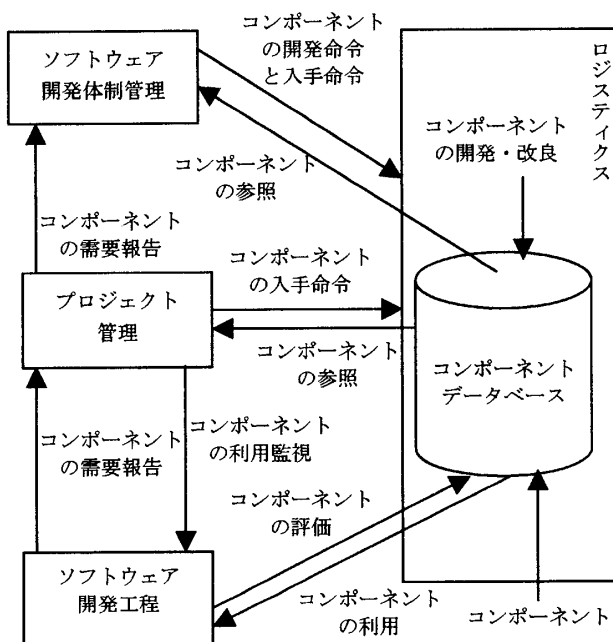


Fig.2 コンポーネントと情報の流れの関係

開発力強化の観点からロジスティクスでコンポーネントの開発やコンポーネントの入手を行う。このとき、組織全体の開発力強化に当たるため、ソフトウェア開発体制管理からロジスティクスへコンポーネントの開発命令やコンポーネントの入手命令が出る。

3.2 コンポーネントの利用

コンポーネントはソフトウェア開発工程の中で利用する。このとき、プロジェクト管理でコンポーネントの利用を監視する。コンポーネントデータベースにないコンポーネントを使いたいときは、ソフトウェア開発工程からプロジェクト管理へコンポーネントの需要報告を行う。プロジェクト管理でそのコンポーネントを入手するか判断し、そのプロジェクトで使うのであればコンポーネントの入手命令をロジスティクスへ出す。また、コンポーネントの利用監視中にコンポーネントのリファクタリングが必要だと判断した場合、プロジェクト管理からソフトウェア開発体制管理へコンポーネントの需要報告を出す。

3.3 コンポーネントの評価

コンポーネントの評価は、単体の評価と全体の評価を行う。単体の評価はコンポーネントの品質を評価する。全体の評価はコンポーネントの分類と開発力を強化したい分野との一致を評価する。

ソフトウェア開発工程でコンポーネントを利用した場合、コンポーネントの単体の評価を行う。評価は数字評価と記述評価を行う。数字評価が低いコンポーネントから記述評価を参考にしてリファクタリングを行う。特に、利用率が高く評価が低いコンポーネントを優先する必要がある。ロジスティクスでコンポーネントの準備を繰り返すことで開発力を強化する。

ソフトウェア開発体制管理ではコンポーネント全体の評価を行う。コンポーネントの粒度に係数を割り当て、コンポーネントの種類を乗じることでコンポーネント開発力を求める。戦略的視点から強化したい開発力を決め、目標を設定する。ロジスティクスでは、その目標に向かってコンポーネントの準備を行う。

コンポーネントベース開発では、コンポーネント開発力を向上することで、組織の開発力強化とプロジェクトの成功を両立できる。

4. おわりに

本報告では、ソフトウェア開発体制の戦略的構築とロジスティクスの運用について提案した。今後の予定として、コンポーネントが利用できない作業への対応の検討を行う。また、コンポーネントデータベースの実装を行い、ソフトウェア開発体制の戦略的構築とロジスティクスの運用の評価を行う。

参考文献

[1] 瀬川弘司 編：プロジェクトを成功に導くソフトウェア開発の必修スキル，日経 BP 社