

ソフトウェア工程管理システム IKKS (1)

3R-1

— システム化へのアプローチ —

寺門浩之¹ 原田勝利² 樋山淳雄² 岩崎新一² 岡田雄一郎² 東出正裕²¹日本電気技術情報システム開発株式会社 ²日本電気株式会社

1 はじめに

今日ソフトウェア開発量の激増とユーザ要求の多様化により、ソフトウェア開発部門の管理者にとって開発作業を効率よく計画・管理するためにはかなりの経験と努力が必要となってきた。また、ソフトウェアの開発管理を行なう管理者が不足あるいはその作業に集中することができない状況が顕著になってきている。

このために、プロジェクト開始時の計画が十分練成されずに作業の見積りが甘くなったり、作業の遅れがなかなか報告されずに処置が手遅れとなり納期の遅れとなってしまうといったケースが目立ってきている。

このような状況におけるソフトウェア開発管理の問題点を簡潔にまとめると次の様になる。

1. 開発計画が変更され易い
2. 進捗の把握が難しい
3. 迅速な対応がとられていない

これらの問題点を持ったソフトウェア開発管理作業は開発作業の進行に伴い計画の修正・変更が数多く発生する。しかも、対象となるプロジェクトが大きくなればなるほど変更量は多く、かつ複雑になるため、それらにうまく対処し管理するためには、ツールによるサポートが必須になってくる。

そこで本論文では、ソフトウェア開発管理の進め方および開発管理のツール化を行なうに当たっての考慮事項を検討し、ソフトウェア工程管理を総合的に支援するシステムの基本概念について提案する。

2 ソフトウェア開発管理の進め方

ソフトウェア開発管理の進め方を工程管理のそれぞれの局面ごとに考察する。

2.1 計画立案局面

開発計画に基づいて作成されるのがスケジュールであるが、計画通りに作業を進めるためには、まず実行可能

なスケジュールを作成することが必要である。このためには、

- 作業決定…… 作業項目とその内容を決める
- 作業構成…… 作業の関連(順序、優先度)を決める
- 見積り…… 工数、規模、費用を見積もる
- 日程計画…… 作業のスケジュールを作成する

などを十分考慮することが大切だと考えられる。

2.2 進捗把握局面

実際の作業の開始後は、常にスケジュール通り作業が進んでいるかその進捗を、感覚的でなく具体的に把握しておかなければならない。そのためには進捗の計測方法、評価方法、分析方法等を設定し、進捗の計測を客観的に、具体的にこなすことが必要だと考えられる。

2.3 対応局面

進捗を把握し、万一スケジュールが遅れていた場合にはその対応を速やかにとる必要がある。進捗の遅れを分析した結果をもとにとるべき方策を決め、

- 原因説明…… 計画とのずれの原因を追求する
- 方策決定…… とるべき対応方法を具体的に決める
- 対応…… 開発活動に対して実際に対応する

などについて具体的な対応をとることが大切だと考えられる。

3 システム化に当たっての考慮事項

ソフトウェア開発管理は、開発対象が大規模になるほど計画立案・進捗把握・対応が複雑になり、これらを効率良く行なうことが困難となる。

このため、ツールによる工程管理のサポートが望まれるが、この工程管理を支援するシステムは、単にスケジュール表を作成・修正するだけのものではなく、管理者の意思決定を促せるようなものでなければならない。

ここでは、実際に工程管理を支援するシステムを開発する上で必要と思われる基本的な考え方について考察する。

A Software Development Control System IKKS (1)

— Approach to Systemization —

Hiroyuki Terakado,¹ Katsutoshi Harada,² Atsuo Hazeyama,² Shin'ichi Iwasaki,² Yuichiro Okada,² Masahiro Higashide,²¹NEC Scientific Information System Development, Ltd.²NEC Corporation

3.1 スケジュールデータのモデル化

まず、何を管理するのかを明確にするためスケジュールをモデル化する必要がある [1]。

ここでは1つのスケジュールを、「どのソフトウェアについて(作業対象)」、「どのような作業を(作業項目)」、「誰が(作業)」、「いつからいつまで(作業時間)」の4つの要素の関連として考える。

スケジュールの各要素「作業対象」、「作業項目」、「作業時間」は単なるデータの集合としてではなく、それぞれを階層データとしてとらえる。これら階層化された各データを組み合わせスケジュールに設定することにより、プロジェクト全体のスケジュールから個人の詳細なスケジュールまでの様々なビューを得ることが可能となる(図1)。

更に、正しいスケジュールを作成するためには、スケジュール間に内含関係(親子関係)、前後関係等のルールを規定することも必要と考えられる。

また、実際に、開発管理作業を行なうためには、上の4要素から構成される情報だけではなく、「費用情報」、「工数情報」、「進捗情報」、「設備情報」、「品質情報」、等の関連情報も必要であると考えられる。

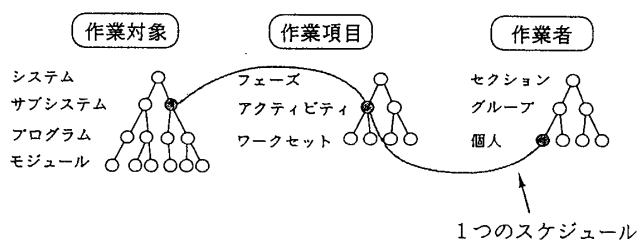


図1 スケジュールデータの階層化

3.2 管理の階層化と標準化

(1) 標準化

2.1 項の計画立案局面での作業項目の決定、見積り作業等をより正確に行なうためには、作業項目、作業対象の定義、作業時間情報などが標準化されていることが要求される。

また、標準化は計画立案局面だけでなく、進捗把握局面においても有効である。すなわち、進捗の計測基準が統一され、より正確な進捗把握が行なえるようになる。

標準化は単一のプロジェクトだけでなく組織における複数のプロジェクト間で適用し、管理情報の標準ファイル化を行ない、リソースの共有化と管理作業の標準化を図ることが望ましい。

(2) 階層化

スケジュールに全体線表から詳細線表があるように、開発管理を行なう管理者にもいくつかのレベルがある。すなわち、

1. 個人の工程の管理
2. プロジェクトの工程の管理
3. 複数のプロジェクトの工程の管理

そこで、工程管理を行なう管理者を階層化し、それぞれのレベルに合わせた情報の提供が必要と考えられる。

3.3 履歴情報の提供

開発管理を遂行する上で、どの段階での計画がうまくいかなかったかを検討することは大切である。このためにプロジェクトの初期計画から終了までの各段階での計画情報、予実のずれを保存しそれを履歴情報として提供することも有効と考えられる。

3.4 高度な UI の提供

ユーザに使い易さを提供するための高度なインターフェースの提供が必要である。また、同時にユーザがインターフェースをカスタマイズする機能も提供することが望ましい。

3.5 多様なユーザビューの提供

1つのスケジュールでも、その見方はガントチャート、フローチャート、アローチャート、等々である。ユーザが見たいデータを見たい形で見せるため、各種管理情報に合わせて、その情報にふさわしい形式(グラフ、図、表等)で提供することが望ましい。

4 おわりに

以上のような考え方に基づき、我々はソフトウェア工程管理システム IKKS を開発した [2, 3]。開発に当たっては、上記で述べてきた点を考慮し、次の3点を主目的とした。

1. ソフトウェア開発作業の管理者が必要とする情報を効率良く入力/編集し、管理する。
2. 入力された情報を管理者に分かりやすくかつ、効果的に提供(表示/出力)する。
3. 複数のプロジェクトに対しての共通リソース管理と管理作業の標準化を支援する。

参考文献

- [1] 岡田：ソフトウェア開発におけるスケジュールデータのモデル化について、情報処理学会第38回全国大会, 3M-7 (1989).
- [2] 樋山他：ソフトウェア工程管理システム IKKS(2) — 概要と特徴 —, 情報処理学会第39回全国大会, (1989).
- [3] 原田他：ソフトウェア工程管理システム IKKS(3) — 管理の標準化と階層化 —, 情報処理学会第39回全国大会, (1989).