

要件定義における プロジェクトマネジメントフレームワークの提案

石井 信明

A Project Management Framework at System Requirements Definition Phase

Nobuaki Ishii

要 約

本論文では、システム開発プロジェクトの成功に大きく影響する要件定義について、要件定義の品質と生産性のバランスを考慮したプロジェクトマネジメントフレームワークを提案する。提案フレームワークでは、要件定義の品質に対応する指標として、要件が定義され、それを各ステークホルダーが共有化した状態を示す要件構造化率 (S) を導入する。 S は、各ステークホルダーの要件を仕様として定義する要件定義開発に要する工数 (Man-Hour)、および、各ステークホルダーの個別要件の獲得、調整、交渉、あるいは、変更などの要件管理に関する工数の割合として求める。提案フレームワークでは、 S および要件定義開発に要する累積工数 (CMH) の両面から要件定義計画 (S - CMH 計画) を立案し、 S - CMH 計画と実績との差異から要件定義遂行に関わる人的リソース体制を変更することで、要件定義の品質、および、生産性の向上を実現する。本論文では、要件定義におけるプロジェクトマネジメントの必要性、既存の研究、提案フレームワークの概要、および、フレームワークの課題と今後の研究課題について述べる。

キーワード：要件定義、システム分析、システム設計、要件構造化率、進捗管理、プロジェクトマネジメントフレームワーク

Abstract

The system requirements definition phase, which is the early phase in an information system development project, has a strong impact on the project's success. This paper proposes a project management framework, considering both quality and productivity of the system requirements definition, in order to make an information system development project succeed. The proposed framework introduces the rate of structured requirements (S), which indicates how well requirements are defined and are shared among stakeholders. The index S is defined based on the man-hours for requirements definition development tasks to define requirements as specifications and the man-hours for acquiring,

coordinating, and negotiating diversified requirements among stakeholders. The proposed framework makes a requirements definition plan (S-CMH plan) from the standpoints of *S* and *CMH*, which means cumulative man-hours for the system requirements definition phase. In addition, the proposed framework checks the deviation of *S* and *CMH* from the S-CMH plan, and modifies the human resources organization for the system requirements definition in order to improve quality of the deliverables and productivity of the system requirements definition phase. This paper describes the background of this research, surveys prior research, explains the proposed framework and discusses issues and further research area on the proposed framework.

Keywords & Phrases: System Requirements Definition, System Analysis, System Design, Rate of Structured Requirements, Project Management Framework

1. 緒言

情報システム開発（以下、システム開発）は、品質（Q）、コスト（C）、納期（D）の確保が困難なプロジェクトである。（本論文で述べるシステム開発とは、ソフトウェア開発を主体とした情報システムの開発を指す。）実際、システム開発プロジェクトのおよそ70%は、QCDのいずれかに問題が生じたプロジェクトである [1]。また、システム開発プロジェクトのおよそ18%は、プロジェクト完了後も使用されないか、または、プロジェクト完了前に中断したプロジェクトであるとの調査結果もある [2]。システム開発が困難なプロジェクトである理由として、たとえば、「システムはそれぞれの目的を担ったサブシステムが互いに関連を持ちながら機能する複雑な集合体である」、「システムへの要求は常に変化しており、システム開発プロジェクトの途上で多くの変更要求が発生する」[3] など、システムが持つ特性との関連が考えられる。さらに、システム開発プロジェクトが失敗する背景には、「ユーザー参画の欠如」、「経営者による支援の欠如」、「明確な目標の欠如」など、プロジェクトマネジメントに関わる様々な要因も考えられる [2]。

システム開発の方法に関しては、これまでに様々な開発方法論 [4] が提案されている。それらは通常、要件定義 [5]、設計、プログラミング、テスト、運用・保守の各フェーズに沿って実施する。（本論文では、要件定義と要求定義を同様の意味として使用する。また本論文では、システム開発における要件定義を、システムオーナー、システムユーザーなどの関係者（ステークホルダー）からシステム化目標、システム化要件を抽出し、それらを技術面、運用面、そして、経済面から実装可能性を検討した上でシステム設計用の仕様として表現し、それらを各ステークホルダーが共有化する作業と定義する。）

この内、システム開発の初期段階である要件定義は、ステークホルダーの曖昧な要件を技術面、運用面、そして、経済面から設計可能な内容を含んだ図書にまとめる役割を担っている。すなわち要件定義は、その後の設計から運用・保守フェーズにまでつながるシステムライフサイクルの評価を決定付ける重要なフェーズである。たとえば、システム開発の特性として先に述べた「変更要求」を抑止するには、要件定義の品質がポイントになる。また、先にあげたプロジェクトマネジメントに関わる様々な失敗要因についても、ステークホルダー間で異なる要件の調整方法など、要件定義の進め方に起因することが多い。

このようにシステム開発プロジェクトを成功に導くには、コスト、納期など様々な条件の下で要件定義成果物の品質を向上することが重要である。これまでに多くの研究者が、要件定義のプロセス、要件獲得の手法など、要件定義に関する個々の手法を中心に提案を行っている。しかしながら要件定義においても、コスト、納期など、プロジェクト遂行上の制約があり、その遂行にはマネジメントの視点が欠かせない。しかし、成果物の品質を保ちながら効率的に要件定義を行うことに視点を置いたプロジェクトマネジメントに関する研究は、現在までのところ限定的といえる。

本論文では、以上の観点から、システム開発プロジェクトの成功に大きな影響をもつ要件定義を対象に、要件定義の品質と生産性のバランスを考慮したプロジェクトマネジメントフレームワークを提案する。ここでプロジェクトマネジメントフレームワークとは、マネジメントの目的とそれを達成するための仕組み、および、要素を指すものとする。表1に本論文で提案するプロジェクトマネジメントフレームワークの概要を示す。(以下では、プロジェクトマネジメントフレームワークをフレームワークと呼ぶ。)

表1 プロジェクトマネジメントフレームワークの概要

目的	要件定義の品質と生産性のバランスを考慮したプロジェクトマネジメントの達成
仕組み	<ul style="list-style-type: none"> ・要件構造化率 (S) と要件定義開発に要する工数の両面から要件定義を計画 ・計画と実績の差異を定期的な管理サイクルで監視 ・差異の状況により、要件定義体制 (人的資源の編成, 投入工数) を変更
要素	<ul style="list-style-type: none"> ・進捗評価方法 ・要件定義計画立案 ・進捗差異改善方法

本論文で提案するフレームワークでは、各ステークホルダーの知識、要件の獲得、さらに、それらの成果物としての仕様化などの要件定義開発に要する工数 (Man-Hour)、および、要件に関する作業調整、交渉、あるいは、変更などの要件管理に関する工数を基に、対象システムへの要件が定義でき、それを各ステークホルダーが共有化した状態を示す指標として、要件構造化率 (S) を導入する。Sは、要件定義の品質に対応する指標と位置づける。さらに提案フレームワークでは、S、および、Sに対する要件定義開発の累積工数 (CMH) の両面から要件定義のプロジェクト計画 (S-CMH計画) を立案する。CMHは、要件定義の成果物作成に直接関係する工数であり、要件定義の生産性に対する指標と位置づける。そしてSとCMHの計画と実績との差異を一定の管理サイクルで定期的に監視し、必要に応じて要件定義の遂行方法を変更することで要件定義の品質、および、生産性の向上を実現する。

ここで要件定義の品質とは、要件定義がシステム開発プロジェクトの最終的な成否、すなわち、プロジェクト全体のQCD目標に与える影響の視点から考える。たとえば、要件定義以降のシステム開発フェーズにおいて要件定義にさかのぼる仕様変更の回数などがあてはまる。システム開発プロジェク

トフェーズ全体を通して見た場合、要件定義の品質への評価は要件定義終了後に実施するフェーズで明らかになる。SおよびCMHによる要件定義の評価は、システム開発プロジェクトを通じた要件定義への評価を事前に要件定義フェーズにおいて行うことを意味している。

以下では、問題の背景と研究の意義、既存の研究、および、提案フレームワークの概要について説明する。最後に、提案フレームワークの課題、および、今後の研究を要する事項について言及する。

2. 問題の背景

2. 1 システム開発と要件定義の重要性

システム開発プロジェクトの最終的な成否と要件定義との関係については、これまでに多くの調査、研究が報告されている。

たとえばBlanchard [6] は、システムライフサイクルにおける初期段階の意思決定がライフサイクルコスト(LCC)の70%~80%を決定付けることを報告している。またBoehm [7] は、システム開発プロジェクトの実績データ分析から、プロジェクトが進行するに従い、プロジェクト初期フェーズに遡る修正または変更(手戻り)費用が増大することを検証している。さらに、Wieggers [5] は、システム開発における「手戻り」が全システム開発費の30~50%を占めており、さらに、要件定義の誤りによる手戻りコストがその内の70~85%を占めていることを紹介している。システム開発プロジェクトの実態調査 [1], [2] においても、要件定義を含むシステム開発の上流フェーズの内容が、システム開発全体のQCDに影響する点を報告している。また、要件定義の成果物の品質とシステム開発プロジェクトの最終的な成否との関係の説明を試みる研究も行われている [8]。

また石井 [9] は、実際のシステム開発プロジェクトについて、プロジェクト過程における各ステークホルダーのプロジェクトへの認識の変化を分析し、プロジェクト経験豊富なプロジェクトマネージャーは、要件定義の進捗状況からそのプロジェクトの最終的な成否を想定し、随時必要な是正措置を講じていることを報告している。

すなわち、要件定義の良否は、そのままシステム開発プロジェクトの最終的な成否につながるものであり、要件定義段階のプロジェクトマネジメントの失敗は、そのまま失敗プロジェクトの評価を受けることにつながるといえる。

2. 2 要件定義の困難性

要件定義では、ユーザー、オーナー、システムアナリストなどの関係者(ステークホルダー)が、あいまいなシステム化要件に基づいて対象システムを調査、分析し、明確なシステム化目標、要件にまとめていくことになる。要件定義の初期には、通常、システム化の範囲、必要な作業項目、時間、リソースがあいまいであり、それらは要件定義を進めるに従って次第に明らかになる。すなわち、要件定義の作業内容とプロセスには不確実性が存在し、適用する方法論の選定を含め、確度の高いプロジェクト計画を立案することは困難である。その結果、要件定義段階の正確な進捗把握とマネジメントがおろそかになりやすい。また、要件定義にはそれぞれ異なる背景を持ったステークホルダーが参画しており、要件定義の進捗あるいは品質面の完成度を各ステークホルダーに正しく伝え、共有化することが難しい。そのため、要件定義の成果、および、進捗状況への認識がステークホルダー間で一致することなく、工数実績、発行済み要件定義書数、あるいは、仮に定めた要件定義終了期日などを

基に、要件定義を終了としてしまうことが多い。さらに、要件定義を順調に進めるには、対象領域全般の知識を持ち、各ステークホルダーの要件獲得と、要件の調整・交渉の出来る人材の起用が好ましいが、実際のところそれらを全て満たした人材を探すことは難しい。

要件定義の困難性に関する要因を、表2に、「作業内容の不確実性」、「プロセスの不確実性」、「コミュニケーションの困難性」、および、「的確な人的資源確保の困難性」の観点から示す。

表2 要件定義の困難性の要因とその概要

困難性の要因	概要
作業内容の不確実性	あいまいなシステム化目標、問題構造、システム構造、入出力情報による作業を求められ、適切なプロジェクト計画の立案が困難
プロセスの不確実性	要件定義の開始当初はその作業内容があいまいであり、要件定義プロセスの見通しと進捗管理が困難
コミュニケーションの困難性	各ステークホルダーの様々な個別要件、プロジェクトへの認識、思いが交錯し、それらの共有化が進まないままに要件定義を終了とする場合が多い
的確な人的資源確保の困難性	対象領域全般の知識を持ち、要件獲得およびステークホルダーとの調整・交渉の出来る人材の確保が困難

2.3 要件定義とマネジメント技術

開発プロジェクトに関するマネジメント技術については、プロジェクトマネジメント [10] の研究対象として、知識体系の整理に加え、理論面および実務家を含めた実践面からの研究、および、ツールの開発が進んでいる。プロジェクトマネジメントでは、プロジェクトのスコップ、コスト、時間、品質を計画し、様々なツールを活用しながら、状況に応じた人的リソース、コミュニケーション、リスクなどのマネジメントを実践し、プロジェクトを成功に導いていく。

これまでのプロジェクトマネジメントでは、スコップのあいまい性が少なく、プロジェクト計画が立案できる時点から後のフェーズをマネジメントの主な対象としてとらえている。たとえば、プロジェクトマネジメントの中核の一つをなすWBS (Work Breakdown Structure) [10] は、プロジェクトのタスクが明確でないと作成できない。

しかしながら図1に示すように、要件定義開始時点では、通常、システム化目標、範囲、機能などはあいまいであり、要件定義を進めることでこれらが徐々に明らかになっていく。各ステークホルダーの個別要件も、要件定義開始当初は整合性が取れていないことが多い。そのため、要件定義では大まかなWBSを設定した上で、およそのシステム規模と仕様から、経験的に要件定義の内容、期間、および、総工数を設定することが通常であり、通常のプロジェクトマネジメントの知識、手法をそのままでは利用できない。

すなわち要件定義においては、要件定義の進捗状況を随時把握し、それを要件定義以降のシステム

開発プロジェクトのQCDに与える影響を考慮しながら要件定義の進め方を柔軟に変更し、与えられた期間と総工数の範囲内で効果的に要件定義を進めるプロジェクトマネジメント技術が必要といえる。

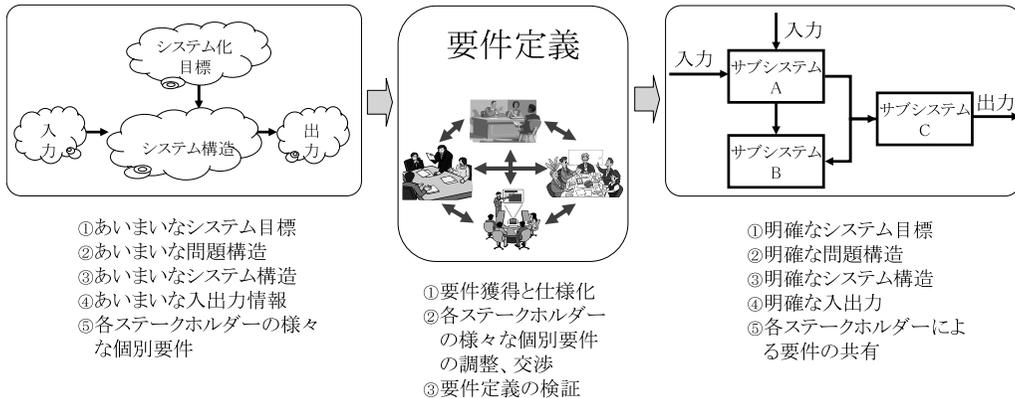


図1 要件定義の概要

3. 既往の研究

これまで多くの研究者が、システム開発における要件定義プロセス、要件獲得手法、そして、システム開発のプロジェクトマネジメントについて研究を行っている。

(1) 要件定義プロセス

要件定義プロセスについては、多くの研究者が、会議、インタビューなど、ステークホルダー間のコミュニケーション、調整、交渉などを通じた要件獲得を支援する観点から研究を行っている。

たとえば土井ら [11] は、会議を通して顧客の要求を洗練し、開発者のために要件を網羅的に獲得する要件獲得方法論として、USP (User-oriented System Planning method)を開発している。USPでは、要件を洗練する方法として、陳述、提案、議論、質問、交渉などインタラクティブ性の高い「会議」をコミュニケーション・ツールとし、会議を効果的に進めるための支援法（オンライン法）と会議分析法（オフライン法）を展開している。そして、このオンライン法とオフライン法を繰り返すことで、要件の質と量を洗練することを提案している。また、オフライン法として、会議で述べられている静的な面を取り出す「機能分析」、動的な情報を取り出す「シナリオ分析」、機能の品質情報を取り出す「品質分析」、そして、会議参加者には直接わからない関心事項の情報を取り出す「関心事分析」からなる方法を提案している。

長田ら [12] は、要件間の相互関係を考慮することで矛盾のない要件定義（仕様書作成）を行うために、要件のドメイン特性とその相関を表すメタモデル、および、既存システムの仕様書を用いたドメインのモデル構築手法について研究を行っている。研究では、モデル構築方法を示すとともに、ケーススタディによりその有用性と改善を要する点を示している。

また、品質管理の分野で主に新製品開発に使われているQFD（品質機能展開）の考え方を要件定義プロセスに応用し、プロジェクト計画作成、情報共有、知識蓄積のための手法を開発する研究 [13] が進みつつある。

Kirsch, et al. [14] は、多くのローカルなユーザーグループの要件を満足する共通（グローバル）要件を定義するためのプロセスについて研究を行い、ケーススタディに基づいた要件定義プロセスモデルを提案している。提案プロセスモデルでは、各ステークホルダーが、ローカルな問題領域の理解と潜在的な要件の明確化をおこなう「知識獲得」、および、共通要件の合意形成のための「交渉」の2段階のサイクルを繰り返しながら、共通要件の定義に加え、ローカルユーザーの共通要件への理解を促進することを仮定している。そして、知識獲得は、新たなシステムへのグローバルな要件を各ステークホルダーが共有した上で、ローカルな要件を獲得する構造化された合理的な活動としている。それに対し交渉は、異なるステークホルダーがそれぞれの利害により行動する、非構造かつ政策的なものであることを示している。さらに、「知識獲得」と「交渉」のサイクルからなる要件定義プロセスを効率よく遂行するには、各ステークホルダーの間に入ってそのプロセスをコントロールするファシリテーター(facilitator)の役割とテクニックが重要であることを述べている。

（２）要件獲得手法

要件獲得手法については、多くの研究者が、ステークホルダー間のコミュニケーションの視点、および、コミュニケーションの主体である自然言語記述の視点から研究をおこなっている。

たとえばMaier *et al.* [15] は、コミュニケーションが要件定義にとどまらず多くの設計プロセスの基本要素であり、プロジェクトの成否の決定要因であることを述べている。そして事例研究を通して、コミュニケーションの向上には、CSCW (Computer Supported Cooperative Work) ツール、KMS (Knowledge Management System) などの情報技術の導入以上に、マネジメントの役割が重要であることを指摘している。さらに、コミュニケーションを向上するためのポイントを診断する方法として、maturity grid-inspired approach を提案している。

Sawyer *et al.* [16] は、要件獲得における要件の自然言語記述に注目した研究として、特に問題のドメイン領域の理解に関わる要件定義の初期フェーズに注目した研究を、language engineering [17] の応用として行っている。研究では、ステークホルダーへのインタビュー記録、あるいは、会議の会話記録などの生のテキスト情報からドメイン領域の主要概念の特定を支援する方法として、いくつかのstatistical language engineering 手法を組み合わせたCorpus-based statistical language engineering techniquesを提案している。

（３）システム開発マネジメント

システム開発プロジェクトに関するマネジメント手法については、プロジェクトマネジメントの対象分野として研究が進んでいる。

なかでもプロジェクトの時間と費用に関するマネジメントとして、EVM (Earned Value Management) [18] に関する研究が進み、それら成果の実務での活用が広がっている。たとえば、Liu, et al. [19] は、EVMを活用し、プロジェクトの進捗を定量的に管理する手法を提案している。また、従来からプロジェクトマネジメントの対象であったスコープ、費用、時間、品質に加え、リスクマネジメント、コミュニケーションマネジメントなど、プロジェクトを成功に導くために、より幅広い視野からの研究が行われている。

さらに、システム開発におけるプロジェクトマネジメントの情報活用に関する研究も進んでいる。たとえば大平ら [20] は、システム開発にともなうデータを収集・分析することでプロジェクトの問題点を迅速に発見し、的確な管理を行うためのプロジェクト支援システムを提案している。

このように、要件定義プロセス、および、要件定義獲得手法については、これまでに多くの研究者が様々な視点から研究を行っている。しかし、システム開発のプロジェクトマネジメントに関する既存の研究の多くは、システム要件の多くが明確になり、WBS、マイルストーンなど、プロジェクトマネジメントに必要な情報を入手可能な設計フェーズ以降を対象にしたものであり、表2および図1に示した特徴を持つ要件定義を対象としたプロジェクトマネジメントに関する研究は限られている。

あいまいな要件から具体的なシステムの姿を導き出すことが役割である要件定義においては、投入する人的リソースによる要件定義の生産性への影響に加え、要件定義過程におけるステークホルダーの要件への動的な認識の変化から生じる変更要求により、プロジェクト計画立案、および、進捗管理などに必要なマネジメント指標の設定自体が難しい。すなわち、要件定義の品質確保に加え、それを費用、納期を考慮しながら達成するには、要件定義を対象としたプロジェクトマネジメントに関する研究を進める必要がある。なかでも、要件定義のプロセスとその特性を考慮したプロジェクトマネジメントフレームワークの研究と、フレームワークに必要な進捗の評価指標、評価指標に対する要件定義計画の方法、進捗把握の方法、計画と実績の差異への対応方法に関する研究を進める必要がある。

4. 要件定義におけるプロジェクトマネジメントフレームワークの概要

4. 1 フレームワーク研究の方法

本論文では、要件定義におけるプロジェクトマネジメントフレームワークについて、次の視点から提案を行う。

(1) 要件定義のモデル化

要件定義の活動内容とそれらの関連をモデル化し、マネジメントにおける計画対象、管理対象、そして、コントロール対象を明らかにする。

(2) 要件定義の進捗評価方法

要件定義の進捗評価方法について、要件定義モデルを基に、その進捗を、品質、時間、費用の面から評価する指標を導き出す。

(3) 要件定義計画立案と進捗差異改善方法

要件定義モデルを基に、要件定義における時間と費用を考慮しながらその品質を確保するための対応方法を導き出す。

(4) フレームワークの開発

要件定義のマネジメントの要素とそれらの関連を明らかにする。すなわち、要件定義の進捗評価方法、進捗差異改善方法の研究結果を統合し、要件定義の特徴を考慮したプロジェクトマネジメントフレームワークを組み立てる。

4. 2 要件定義のモデル化

要件定義は、様々な背景を持つステークホルダーがそれぞれの問題領域を理解し、システム化の目的、業務要件、機能要件を明らかにするとともに、それらをステークホルダー間で共有化する活動 [14] といえる。その活動は、人間と人間とのかかりから成るものであり、要件定義の良否、生産性は、要件定義に参画する人間の経験、スキル、背景となる知識、および、それぞれが代表する利益に依存する。

図2は、要件定義に関するケーススタディ [21] におけるステークホルダーのかかり方のモデル、および、問題点を示している。すなわち要件定義において、システム開発プロジェクトへの投資者であるオーナー、および、利用者であるユーザーなどのステークホルダーは、インタビュー、会議など様々な方法によるコミュニケーションを介して、新システムに関するそれぞれの知識、および、要件を獲得するとともに、それらのギャップの調整、妥協を行う。SE（システムエンジニア）は、ステークホルダー間のコミュニケーションを円滑化する役割を持つとともに、獲得した要件を成果物である要件定義書にまとめる。プロジェクトマネジャー（PM）は、要件定義のQCD全般に責任を持つ。すなわちPMは、要件定義の品質を評価するとともに、要件定義全般の進行を監視し、状況に応じて分野ごとの要件獲得の深さの変更、あるいは、SEの増減などの対応を行う。

また図2では、ケーススタディ [21] から導いた要件定義の課題として、システム化対象分野の知識、および、要件獲得が不十分な状態において、ステークホルダー間の要件の調整、妥協が行われないうちに要件定義書が成果物として完成し、それが次のシステム開発段階である設計フェーズの基になる点を指摘している。すなわち、成果物として要件定義書が発行されても、その内容は要件を満足しているとは限らず、システム開発における設計以降のフェーズにおいて、要件定義に波及する変更が生じる可能性を持ったものであることを示している。また、要件定義がコンセプトレベルに止まり、分野毎の個別要件にまで踏み込んでいない場合も、同様に設計以降のフェーズで要件定義への変更が生じる可能性を持っていることを示している。

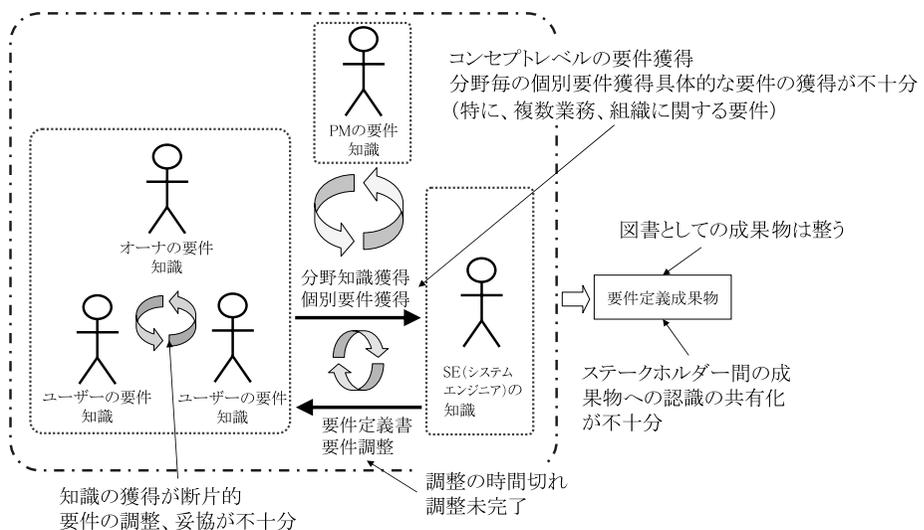


図2 要件定義におけるステークホルダーの関与モデルと要件定義の問題点

本論文では、上記に示した要件定義におけるステークホルダーの関係と課題を基に、要件定義の工数計画と進捗把握に加え、要件定義成果物に対する各ステークホルダーの共有化状態を要件定義の品質を示すものとして定量化し、要件定義の品質と生産性の両面を要件定義におけるプロジェクトマネジメントに活用するフレームワークを提案する。

フレームワークの開発に当たり、本論文では図3に示すように、その基となる要件定義基本モデルを提案する。提案する要件定義基本モデルでは、要件定義を、成果物として要件定義書を作成し、それを解釈する活動（要件定義開発）と、各ステークホルダーがコミュニケーションを介してそれぞれの知識および要件のギャップを認識した上で、そのギャップを調整・妥協しながら改善し、その結果を要件定義の改定として反映する一連の活動の繰り返しと考える。（池田 [22] は、2者間のコミュニケーションの基本モデルとして、前提の共有を基にしたコミュニケーションモデルを提案している。本研究では、この基本的なコミュニケーションモデルを参考とし、要件定義固有の特徴を考慮したモデルを提案する。）

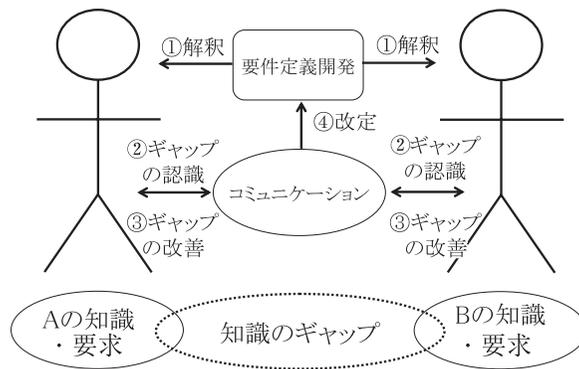


図3 要件定義の基本モデル

4. 3 要件定義の進捗把握手法

本論文では、要件定義の基本モデルに示したように、要件定義において各ステークホルダーは要件定義の成果物となる要件定義書の作成、および、コミュニケーションを介したそれぞれの知識、要件のギャップの認識とギャップ改善のための調整・妥協を行っていると考えられる。すなわち、要件定義の進捗は、要件定義書など、形として認識できる成果物作成の進捗と、各ステークホルダー間のコミュニケーションを介したギャップの改善および調整・妥協による要件定義成果物の共有化の進捗の両面を評価する必要がある。

本論文ではこの両面を、表3に示すように、要件定義書作成作業である「要件定義開発」、および、要件の獲得、ギャップの認識、調整・妥協により各ステークホルダーが要件定義成果物を共有化する作業を「要件管理」として分類する。そして、「要件定義開発」と「要件管理」の両面から要件定義におけるプロジェクト計画を立案し、その進捗を把握する方法を提案する。すなわち、「要件定義開発」は、要件定義基本モデルの「要件定義開発」に関わる部分に対応し、「要件管理」は、「コミュニケーション」を中心とした活動に関わる部分に対応する。

以下では、「要件定義開発」および「要件管理」の進捗把握の方法について述べる。

表3 要件定義作業の分類

要件定義作業	主な内容
要件定義開発	<ul style="list-style-type: none"> ・要件の分析 ・要件定義成果物（要件定義書）作成 ・要件定義成果物の妥当性確認 ・ステークホルダーへの要件定義の説明
要件管理	<ul style="list-style-type: none"> ・システム化目標設定，要件獲得 ・ステークホルダー間の知識，要件のギャップ認識の促進 ・ステークホルダー間の知識，要件の調整・妥協の促進 ・要件定義開発における作業調整，変更管理

4. 3. 1 要件定義開発の進捗把握

要件定義開発は、表3に示すように、要件を要件定義書などの形として認識できる成果物として作成する作業である。すなわち通常のプロジェクトマネジメントでは、WBSを作成した上で、その作業単位で作業工数、あるいは、成果物の数量などを設定し、それらの計画と実績値を基に進捗を把握することになる。しかし、「2. 3 要件定義とマネジメント技術」で示したように、要件定義の詳細な作業内容は、要件定義を進める過程で徐々に明らかになる。このことから、本論文で提案するフレームワークでは、これまでのプロジェクト規模などから経験的に、要件定義開発と要件管理からなる要件定義に必要な総工数、および、要件定義期間を設定し、それらと工数実績との比較により要件定義開発の進捗を把握するものとする。なお、提案フレームワークでは、総工数、および、要件定義期間について、実際の進捗により必要に応じた修正をおこなう仕組みを設ける。

4. 3. 2 要件管理の進捗把握

要件管理は、要件定義基本モデルに示したように、各ステークホルダーが互いにコミュニケーションを介しながら、それぞれの持つ知識、要件のギャップを認識し、それらを改善するための調整・妥協を通して要件定義を共有化する、非構造的なプロセスといえる。そのため、要件定義開発の成果物である要件定義書の出来高、または、工数実績のような一般的な指標では、要件管理の達成目標、および、進捗を表すことは難しい。通常は、ケーススタディ [21] のように、プロジェクトマネジャーの経験、各ステークホルダーの個別認識により進捗を把握することになり、それぞれの進捗への認識の差異がプロジェクト遂行の混乱につながる場合がある。あるいは、プロジェクトマネジャーは、要件管理の進捗は考慮せず、要件定義開発の進捗を考えて設定した要件定義スケジュールを根拠に、ステークホルダーによる要件定義書の共有化が不十分な状態で要件定義を終了とし、問題を設計フェーズ以降に先送りすることになる。

要件定義開発の内容について各ステークホルダーがそれぞれの個別要件とのギャップを認識し、それらを調整・妥協し、最終的に共有化する要件管理のプロセスは、その後の開発フェーズでの修正、手戻りを少なくし、システム開発プロジェクトを成功につなげるものである。すなわちシステム開発プロジェクトの成功には、要件管理の進捗を反映したプロジェクトマネジメントの実現が必要である。

本論文では、要件管理が進むに従い要件の構造化が行われると仮定し、要件構造化のプロセスを考

慮した要件構造化率（S）により要件管理の進捗を把握することを提案する。ただし、要件構造化[21]とは、次の状態とする。

<要件構造化の定義>

- ① システムの目的と範囲，サブシステムの構成，および，システム間入出力データの関係が要件定義の成果物として発行された状態。
- ② 各ステークホルダーが，①の要件定義成果物に関する認識を共有化している状態。

また各ステークホルダーは，要件定義基本モデルに示すように，要件定義において次のプロセスにより要件の構造化を行うものと仮定する。

<要件構造化プロセス>

- ① 各ステークホルダーは，要件定義の内容をそれぞれの知識と要件の前提に基づいた情報交換を行うことで，互いのギャップを認識する。
- ② 明らかになったギャップに関して，各ステークホルダーはコミュニケーションを介しながら，要件定義の修正，または，調整・妥協を行う。
- ③ ①におけるギャップの認識と②におけるギャップの修正，または，調整・妥協の過程を繰り返しながら，ギャップの縮小と要件の共有化を行い，要件定義の構造化を達成する。

本論文では，上記に示した要件定義構造化の定義，および，要件構造化プロセスを前提に，表3に示した要件定義開発と要件管理に要する工数の割合が要件定義の進捗との関係において図4のように推移すると仮定し，要件構造化率を求める。図4（1）は，要件定義開発への工数実績が増加するに従い要件定義が進行するものとし，その進行にともなう要件定義開発と要件管理への工数割合の変化をモデル化したものである。実際には，要件定義途中で新たな要件が生じることなどによる要件の調整，要件定義書の修正など，図4（2）に示すように，要件管理の工数割合が要件定義の進捗に対して単調に減少するとは限らない。しかし要件定義が終了に近づく段階では，新たな要件の出現，ステークホルダー間での要件の調整などの要件管理に関わる工数は少なくなり，要件定義書の妥当性確認など，要件定義開発に関わる工数が多くを占めるものと考えられる。換言すると，要件管理の工数が多くを占める段階では，依然として基本的な要件の獲得，または，ステークホルダー間の要件への共有化が不十分である状態といえる。さらに，初期段階で要件管理への工数が少ないことは，要件獲得，ギャップ認識など，要件への理解が不足している可能性がある。

すなわち，要件構造化プロセスにおいて，要件定義開始当初は，要件の確認，ギャップの把握，調整・妥協など，要件管理に要する時間の割合が多い。それに対し，要件の内容が明確になり，その内容についてステークホルダー間の共有化が進むに従い，要件定義の作成，妥当性確認に比較して，調整・妥協などの要件管理に要する時間の割合が縮小するものと仮定する。

この仮定を基に，要件構造化の進捗を，次式に示す要件構造化率（S）により把握する。すなわち，ある期間（ i ）に要件定義開発を行う各ステークホルダーにおいて，ステークホルダー間の作業調整会議など，表3に示す要件管理に要した要件管理工数（ MMH_i ），および，要件定義開発工数（ TMH_i ）から， i 期の要件構造化率（ S_i ）を次の様に求める。

$$S_i = TMH_i / (TMH_i + MMH_i)$$

この要件構造化率は、要件定義開発に必要な条件の明確さを表す指標ともいえる。要件構造化率100%とは、要件定義の内容が明確になり、要件定義開発の実施に関わる一連の要件管理、すなわち、図4に示すステークホルダー間のコミュニケーションを介した要件定義へのギャップの認識と、それに基づいた要件定義成果物の改定作業が不要な状態といえる。

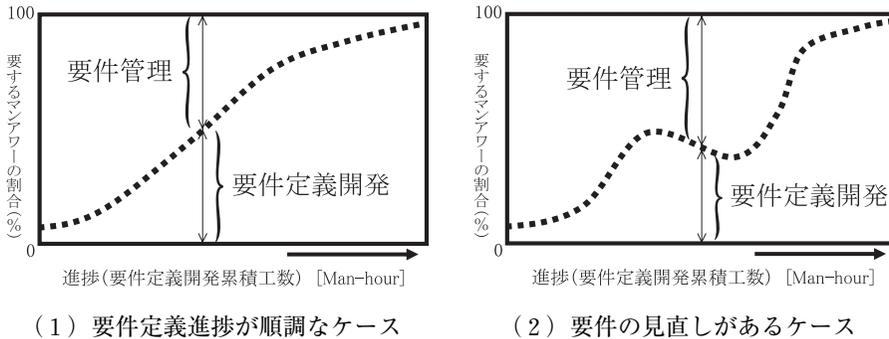


図4 要件定義の進捗と要件定義開発および要件管理にかかわる工数の割合の関係

4. 4 要件定義計画立案と進捗差異改善方法

要件定義の遂行は人間が主体であり、その計画は人的リソース計画（要件定義体制）といえる。本論文では、人的リソース計画に対し、「4. 3 要件定義の進捗把握手法」で示した2つの進捗把握指標、すなわち、要件定義開発の工数、および、要件構造化率（S）に対し、それらの進捗と人的リソース投入の関係を予測し、人的リソース計画を立案するものとする。すなわち、要件定義の進捗計画を立案し、さらに実績との差異の修正を行うには、人的リソースと要件定義の生産性との関係を明らかにした上で、人的リソースの投入と要件定義の成果を予測する必要がある。

システム開発プロジェクトの生産性と人的リソースの関係については、これまで多くの研究者が、システムの規模と工数の関係に視点をおいた報告、研究を行っている。

Banker *et al.* [23] は、システム開発において、システム規模の拡大に従いプロジェクトの固定的オーバーヘッド減少により開発効率が向上するケース（規模の経済）と、システム規模の拡大に従いマネジメントの困難度増加により開発効率が悪化するケース（規模の不経済）があることを、実際のプロジェクト記録から分析した。その上で、システム開発における規模の経済と規模の不経済との関係を反映したソフトウェア開発モデルを作成し、両者の関係を分析した。その結果、システム開発においては、システム規模が拡大するにつれて規模の経済と規模の不経済が転換する点（MPSS: Most Productive Scale Size）があることを明らかにし、MPSSをシステム開発プロジェクトのマネジメント指標にすることを提案した。

Stensrud *et al.* [24] は、ERPプロジェクトの生産性について、48件の実績データをDEA (Data Envelopment Analysis)を適用して分析し、システム規模を表すFP (Function Points) [25] が大きくなると規模の不経済が生じる場合のあること示した。

これらの研究成果を含め、システム規模が拡大した場合にシステム開発の生産性が低下する要因として、次の点が考えられる。（なお以下では、システム開発プロジェクトの遂行形態を考慮し、シス

テム開発を実施する人的リソースをグループの単位で考える。個々のグループは、SE、アナリストなどからなり、一つのステークホルダーを形成する。）

＜システム規模拡大にともなう生産性低下の要因＞

- ・ チームメンバー間の相互関連複雑化によるコミュニケーションオーバーヘッド（COH）の増加 [3], [7]
- ・ 計画・実績管理、ドキュメント管理などの管理業務の増加
- ・ プロジェクトスケジュールへの過剰な余裕の見積

すなわち要件があいまいな状況では、要件定義開発を行うグループ内、および、グループ間でシステム化目標、範囲、作業内容確認、作業分担確認、そして、変更管理などの要件管理のためにCOHに多くの工数を必要とし、場合によりCOHが保有工数をオーバーしてしまうことになる。（保有工数とは、ある期間に人的リソースが提供できる工数の合計を示す。たとえば、1ヶ月間の保有工数（Man-hour）で現せば、「人員数×1ヶ月間の一人当たり稼動時間」で求まる。）たとえば、3グループで要件定義を行う場合、仮にグループ間のCOHに工数の60%を必要とすると、各グループはCOHに120%の工数を要することになり、その期間の保有工数を超えることになる。この状態は、要件管理の作業が十分に行えず、プロジェクトが混乱している状況といえる。

すなわち図5に示す様に、要件定義初期における、システム化要件があいまいで要件管理に多くの工数を多く必要とする段階（要件構造化率（S）の低い段階）で人的リソースを投入しても、グループ間の要件調整・妥協作業によりCOHが増加し、必ずしも要件定義の生産性は向上しない。むしろ、システム開発の生産性に関する既存の研究結果が示すように、多くの人的リソースを投入すると生産性はかえって低下し、本来の要件定義開発の進行を阻害する可能性がある。ただし、限定した人数で作業を継続したのでは、システム開発全体のスケジュールを圧迫しかねない。（図5の実線は、要件定義を実施するグループ間のコミュニケーションの状況を示している。すなわち要件定義が進んでいない状況では、要件のギャップの確認、調整・妥協などにグループ間の組み合わせによりCOHが発生する。これに対し要件定義（要件構造化）が進んだ段階では、各グループ内で要件定義開発を行えば良い状態となり、グループ間のCOHは減少するものと想定する。）

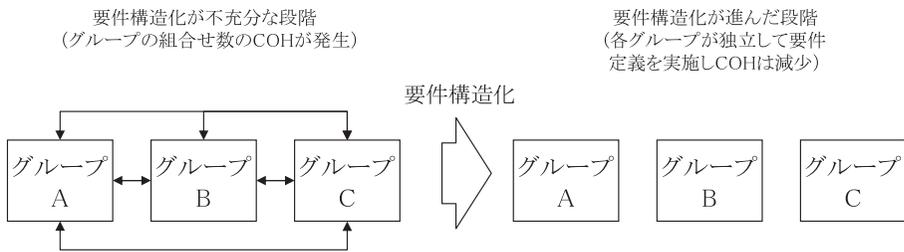


図5 要件の構造化とコミュニケーションオーバーヘッド（COH）の関係

すなわち、要件定義を限られた工数の中で効率的に進めるには、図6に示すように、要件構造化率(S)と投入工数の状況からの確かな人的リソース計画を立案し、Sの変化に応じて適時その人的リソース計画を見直すことが要件定義の生産性向上につながる。

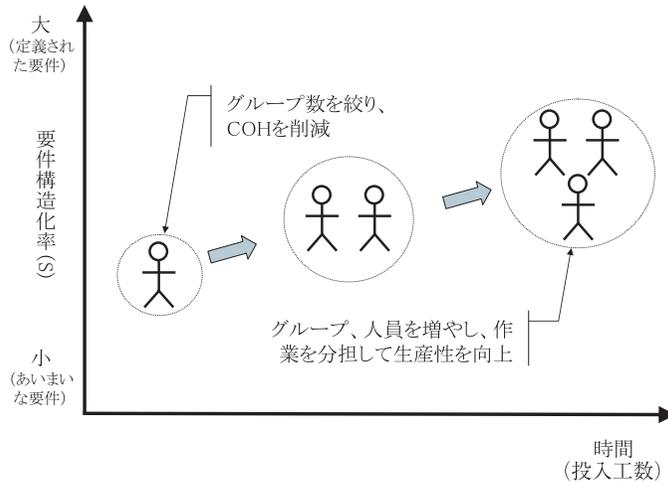


図6 要件定義進捗と人的リソース計画の例

4.5 要件定義におけるプロジェクトマネジメントフレームワークの概要

4.5.1 基本アプローチ

本論文では、「4.3 要件定義の進捗把握手法」、「4.4 要件定義計画立案と進捗差異改善方法」に示した事項を踏まえ、次の基本アプローチにより要件定義におけるプロジェクトマネジメントフレームワークを組み立てる。

- (1) 要件定義の計画、進捗を定期的な管理サイクル(1~n期)ごとに計測し、計画と実績との差異を人的リソース投入の変更により修正する
- (2) 要件定義進捗を、管理サイクル*i*に対して次の2つの側面から監視および評価する
 - ・期間*i*における要件構造化率(S_i)
 - ・期間*i*における要件定義開発累積投入工数(CMH_i)
- (3) 要件定義の達成目標を次の項目により設定する
 - ・要件構造化率達成目標(GS)
 - ・要件定義開発工数目標($GCMH$)
 - ・要件定義期間目標(GT)

なお、要件管理の工数については、 GS 、 $GCMH$ 、 GT 、および、人的リソース計画から求まるものとし、明示的な目標は設定しない。

- (4) 要件定義計画をS、および、 CMH の側面から各計画サイクルの人的リソース計画として立案する

なお、提案フレームワークは、次に示す事項を前提条件としている。

<提案フレームワークの前提条件>

- ・各人的資源の能力は同一（単位工数当たりの生産性の個人差は無い）
- ・システム開発プロジェクトにおけるコミュニケーションマネジメント [10] が有効に機能

4. 5. 2 提案フレームワークの概要

要件定義におけるプロジェクトマネジメントのフレームワークの構造を、図7に示す。

提案フレームワークでは、要件構造化率達成目標（GS）、要件定義開発工数目標（GCMH）、要件定義期間目標（GT）、1単位期間の時間（DUT）に基づき、納期（ $GT \times DUT$ ）目標に合わせた要件定義の進め方を、SおよびCMHの各管理サイクル期間（ i ）の予測として計画（S-CMH計画）する。そのS-CMH計画に基づき、要件定義体制（要件定義を行うグループ構成などの人的リソース編成、および、投入工数）を決定し、要件定義を行う。要件定義の進捗状況は、各期間（ i ）における要件管理工数（ MMH_i ）、および、要件定義開発工数（ TMH_i ）の実績値から算出する S_i 、および、 CMH_i とS-CMH計画とを比較することで把握する。その結果から、 i 期以降の要件定義体制の見直しをおこなう。S-CMH計画と実績が大きく異なる場合には、S-CMH計画の変更、または、目標値の変更を行う。

下記に、提案フレームワークにおける個々の要素の概要を示す。なお、フレームワークの個々の要素の実現については、いくつかの方法が考えられる。（たとえば石井 [26] は、S-CMH計画、S-CMH分析に関する手法を提案している。）本論文ではフレームワークの構造に関する説明を中心とし、フレームワークに必要な個別の要素については、その概要を述べるにとどめる。

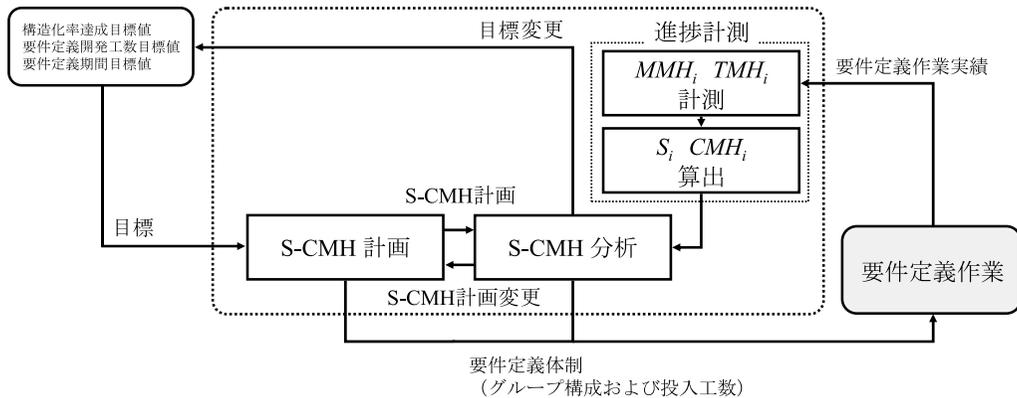


図7 要件定義におけるプロジェクトマネジメントフレームワークの構造

(1) S-CMH計画

S-CMH計画では、目標として与えられたGS, GCMH,および, GTに基づき, 要件定義開発の進め方について計画を立案する. 計画立案の方法は, 経験に基づく方法, 要件定義モデルを作成し, シミュレーション手法により検討する方法などが考えられる. S-CMH計画で立案する計画の内容は, 次の事項である.

- ・要件定義への投入工数と要件構造化率に関する計画
- ・計画を実現するための要件定義体制計画 (人的リソース編成, 各グループの人員数の計画)

(2) 進捗計測

(a) MMHおよびTMHの計測

期間 i において, 要件定義の実施に関わる各グループが要件管理に要した工数を作業報告書などから集計し, MMH_i を算出する. 同様に, 期間 i において要件定義開発に要した工数 (TMH_i) を作業報告書などから集計する. (成果物を工数に換算する方法も考えられる.)

(b) SおよびCMH算出

期間 i における要件構造化率 (S_i) を計測した TMH_i および MMH_i に基づき, 下記の式から算出する.

$$S_i = TMH_i / (TMH_i + MMH_i)$$

また, 期間 i までの要件定義開発累積工数 (CMH_i) を下記の式から算出する.

$$CMH_i = CMH_{i-1} + TMH_i \quad \text{ただし, } CMH_0 = 0$$

(3) S-CMH分析

S-CMH分析では, 図8に示すように, 各期間 (i 期) の終了時点で, S_i および CMH_i の計画値と実績値を比較し, $i + 1$ 期以降の要件定義を行う要件定義体制の見直しを実施する. 図8では, 最終の目標期間GTに対し, 1から5までの期間 (管理サイクル) の計画と実績値を例として示している. (図8中の円内の数値は, 各期の値を示す. また, 実線は計画, 破線は実績を表す.) 各期について, 計画と実績に差がある場合には, 要件定義体制の見直しを検討する.

要件定義体制見直しは, たとえば, 表4に示すような対応により, グループ数の増減, 人員の増減を計画する. 図8の例の場合, 第1期では投入工数実績に対してSの実績が計画値よりも小さいため, 少ないCOHによる効率的な要件定義の実施のために, 要件定義グループ簡素化, あるいは, 投入人員削減を検討する. これに対し, たとえば, 第3期では, 計画値よりもSが優れているため, グループ数の増加, および, 投入人員の増加による要件定義開発の促進を検討する.

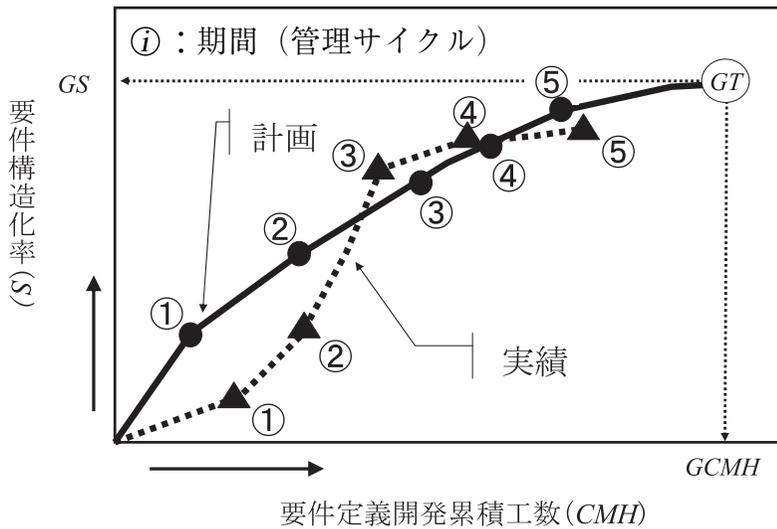


図8 S-CMH計画と実績の比較

表4 S-CMH計画と実績比較による要件定義体制（人的リソース計画）見直し基本方針の例

ケース	Sの値	CMHの値	見直し方針
A	計画値 \geq 実績値	計画値 \geq 実績値	要件定義体制簡素化，投入人員削減
B	計画値 \geq 実績値	計画値 $<$ 実績値	
C	計画値 $<$ 実績値	計画値 \geq 実績値	要件定義実施グループ数および投入人員増員
D	計画値 $<$ 実績値	計画値 $<$ 実績値	現状の体制を継続

(4) S-CMH計画見直し

計画と実績値の差がGT，または，GCMHの達成の観点から許容できない場合は，S-CMH計画を新たに立案する．すなわち，現時点の管理サイクル期間でのSおよびCMHの実績値を初期値とし，新規のS-CMH計画，および，要件定義体制を計画する．さらに，GT，または，GCMHの達成が，S-CMH計画の変更では対応できない場合は，目標の設定値を変更する．

5. 実務への適用と今後の研究課題

本論文では，要件定義におけるプロジェクトマネジメントフレームワークを提案した．提案フレームワークを実務に適用するには，今後，下記の課題の考慮と研究が必要である．

- ・提案フレームワークの個別機能として示した，S-CMH計画，S-CMH分析について，実務での適用を考慮した手法の研究が必要である．
- ・プロジェクトマネジャーは，実際の要件定義体制見直しの場面において，体制の変更による業務

引継ぎ作業が発生し一時的に生産性が低下することを考慮している。またプロジェクトマネジャーは、実際の人的リソースの能力は画一的ではないため、それぞれのもつ知識、経験領域を考慮した見直しを行っている。すなわち、提案フレームワークを実践するには、要件定義体制見直しにおける各人的リソースの個別性、および、体制見直し直後の生産性低下の影響を考慮する仕組みが必要である。たとえば、要件定義体制変更を効果的に行うために、体制変更の要件定義への影響モデルとシミュレーションによる予測手法の研究が必要である。

- ・要件構造化率（S）を計測するには、要件定義における各作業時間を計測する必要がある。時間の計測を効率化するための仕組みが必要である。
- ・要件構造化率（S）を正しく計測するには、要件管理において知識、要件の獲得、要件の調整・妥協が確実に行われる必要がある。そのためには、要件定義におけるアナリスト、あるいは、プロジェクトマネジャーがファシリテーター [14] として機能する必要がある。

6. 結言

本論文では、システム開発プロジェクトの最終的な成否に大きく影響する要件定義を対象に、要件定義の品質と生産性のバランスを考慮したプロジェクトマネジメントフレームワークを提案した。また、提案フレームワークを実務で使用するための現状の課題、および、今後の研究課題について触れた。

要件定義では、その内容があいまいな状況からプロジェクトが始まるため、通常、経験に基づき設定する期間と総工数に基づいたマネジメントが行われる。そのためプロジェクトマネジャーは、詳細な要件獲得にまで至らない状態、あるいは、要件定義の内容についてステークホルダー間の共有化が進まない状態で、当初設定した期間、または、総工数を根拠に要件定義を終了としてしまうことが多い。要件定義における意思決定の先送りは、それが部分的なものであっても、要件定義に続くシステム開発フェーズの作業を混乱させることにつながり、システム開発プロジェクトの最終的な成否に影響する。しかし、要件定義フェーズにおいて無制限に期間と工数を費やすことも、システム開発プロジェクトの遅れ、コスト超過などの混乱を引き起こす。すなわち要件定義においては、本論文で示したように、あらかじめ設定した期間と工数を制約としながらその品質確保を行なうために、要件定義の特性を考慮したプロジェクトマネジメントを導入する必要がある。

参 考 文 献

- [1] 日経コンピュータ, “2003年情報化実態調査”, 11月17日号, pp.50-71 (2003).
- [2] The Standish Group International, 2004 Third Quarter Research Report, ([http:// www.standishgroup.com](http://www.standishgroup.com)).
- [3] Metzger, P. and Boddie, J., *Managing a Programming Project*, Third Edition. Prentice-Hall, New Jersey (1996).
- [4] Whitten, J.L., Bentley, L.D., and Dittman, K.C., *Systems Analysis and Design Methods*, Fifth Edition. McGraw-Hill Irwin, New York (2001).
- [5] Wiegers, K. E., *Software Requirements*, Microsoft Press, Washington (2003).

- [6] Blanchard, B.S., *System Engineering Management*. John Wiley & Sons, New York (1991).
- [7] Boehm, B.W., *Software Engineering Economics*. Prentice-Hall, New Jersey (1981).
- [8] 鎌田真由美, 細川宣啓, 「成果物から見た要求定義」, 情報処理学会研究報告 (情報システムと社会環境研究報告), pp. 9-13 (2005).
- [9] 石井信明, プロジェクト過程のステークホルダー意識変化とマネジメント, 第1回研究発表大会, 情報システム学会 (東京国際大学) (2005).
- [10] Project Management Institute, *A Guide to the Project Management Body of Knowledge* (2000).
- [11] 土井晃一, 蓬萊尚幸, 渡部 勇, 片山佳則, 園部正幸, 「要求獲得会議を分析することによるユーザー指向要求獲得法」, 情報処理学会論文誌, Vol. 44, No. 1, pp. 48-58 (2003).
- [12] 長田晃, 小澤大伍, 海谷治彦, 海尻賢二, 「特定分野のソフトウェアに関する特性の相関を用いた要求獲得法」, 電子情報通信学会技術研究報告 (ソフトウェアサイエンス), pp.1-6 (2005).
- [13] プロジェクトマネジメント学会研究委員会, 「2005年度 第2回 研究委員会フォーラム資料」, プロジェクトマネジメント学会 (2006).
- [14] Kirsch, L. J., and Haney, M. H. Requirements Determination for Common Systems: Turning a Global Vision into a Local Reality, *Journal of Strategic Information Systems*, Vol.15, No.2, pp.79-104 (2006).
- [15] Maier, A. M., Eckert, C. M., and Clarkson, P. J., Identifying Requirements for Communication Support: A maturity Grid-inspired Approach, *Expert Systems with Applications*, (2006).
- [16] Sawyer, P., Rayson, P., and Cosh, K., Shallow Knowledge as an Aid to Deep Understanding in Early Phase Requirements Engineering, *IEEE Trans. Software Eng.* Vol. 31, No.11, pp.969-981 (2005).
- [17] Cunningham, H., A Definition and Short History of Language Engineering, *Natural Language Engineering*, Vol. 5, No.1, pp. 1-16 (1999).
- [18] クウォンティン・フレミング, ジョエル・コッペルマン, 「アーンド・バリューによるプロジェクトマネジメント」, 日本能率協会マネジメントセンター (2004).
- [19] Liu, G., Akiyama, T., and Yokoyama, S., Proposal to Utilize for Quantitative Project Progress Management, *Journal of the Society of Project Management*, Vol. 6, No. 6, pp. 4-9 (2004).
- [20] 大平 雅雄, 横森 励士, 阪井 誠, 岩村 聡, 小野 英治, 新海 平, 横川 智教: “ソフトウェア開発プロジェクトのリアルタイム管理を目的とした支援システム”, 電子情報通信学会論文誌D-I, Vol. J88-D-I, No.2, pp.228-239 (2005).
- [21] 石井信明, “要件定義段階における進捗把握に関する考察”, プロジェクトマネジメント学会, 2006年度春季研究発表大会予稿集, pp.113-118 (2006).
- [22] 池田謙一, 「コミュニケーション」, 東京大学出版会 (2000).
- [23] Banker, R.D. and Kemerer, C.F., Scale Economics in New Software Development, *IEEE Trans. Software Eng.*, Vol. 15, No. 10, pp. 1199-1205 (1989).
- [24] Stensrud, E. and Myrtveit, I., Identifying High Performance ERP Projects, *IEEE Trans. Software Eng.*, Vol. 29, No. 5, pp. 398-416 (2003).
- [25] Stutzke, R.D., *Estimating Software-Intensive Systems*. Pearson Education, New Jersey (2005).
- [26] 石井信明, “システム要件定義における人的資源を中心としたマネジメント手法の提案”, プロジェクトマネジメント学会, 2005年度秋季研究発表大会予稿集, pp. 223-228 (2005).