

ソフトウェア設計品質メトリクスの検証

— I M A P システムにおける品質保証 —

平山雅之 山田 淳

(株) 東芝 システム・ソフトウェア技術研究所

当社では、ソフトウェア生産の工業化一貫支援を目指すシステムとして、I M A P の開発を進めている。I M A P における品質保証技法として品質工程管理図 (Q C P) およびソフトウェア品質評価システム E S Q U T 研究・開発を実施している。Q C P は、ソフトウェア・ライフサイクルの各工程の標準作業・成果物を規定することで、作業品質の向上を目指すものである。E S Q U T は各工程での成果物を品質メトリクスにより定量評価するものである。本報告では、この Q C P, E S Q U T を実際のプロジェクトに適用することで、その試行評価および I M A P における品質保証技法について検討を加えることを目的としている。

Varification of software quality metrics

— Quality assurance methods in IMAP system —

Masayuki Hirayama Atsushi Yamada

Systems and Software Engineering Laboratory, Toshiba Corporation
70 Yanagi-cho Saiwai-ku Kawasaki, 210 Japan

We have been developed the integrated software management and production support system IMAP. In IMAP system, we are researching the QCP and ESQUT as a quality assurance methods.

QCP regulates the production and work of each work process in software lifecycle. QCP aims at improvement of work quality.

ESQUT is a software product quality measurement system, and is an effective measure for improvement of product quality.

In this report, we use QCP and ESQUT for real software project, and we estimate and investigate the software quality assurance methods in IMAP system.

1. 序言

近年のソフトウェア需要増大にともなうソフトウェア生産の分業化、工業化の動きに対応して、我々はソフトウェア生産一貫支援システムIMAP(#1)の研究を進めている。従来、ソフトウェアの品質保証については、テスト工程での試験項目数、テストカバレッジ、バグ発生件数等を主体としたアプローチがなされてきた。しかしながら、このようなソフトウェア作成後のテストに比重をおいた品質保証のみでは、品質作り込みの観点からは十分とは言えない。IMAPにおいては、工程単位での品質保証を前提として、ソフトウェア品質作り込み・作業標準化を目的とした技法QCP(#2)および、ソフトウェア品質評価支援システムESQUT(#3)を研究・開発している。

QCPでは、品質作り込みを目指した標準工程作業・成果物を定義し、また、DR(デザインレビュー)等のポイントを規定することで、ソフトウェア生産における工程作業そのものの品質を保証している。

また、ソフトウェア品質評価システムESQUTはソフトウェア・ライフサイクルに合せ、その各工程での品質メトリクスを導入した品質定量化手法・ツール群より構築されており、ソフトウェア成果物の品質の可視化・定量化による管理を目指すものである。

従来より、ソフトウェアの品質問題の原因の多くがソフトウェア・ライフサイクルの上流工程にあると言われているが、この領域での研究は、あまりなされていない。このため本報告では、QCPの概念をまとめ、ESQUTとの関わりについて述べるとともに、上流工程でのESQUTによる品質評価手法の確立を目的として、当社内での実プロジェクトの設計工程における事例を中心に検討を進め、ESQUTによる品質評価指針を示すものである。

- (#1) IMAP : Integrated software Management and Production support system
- (#2) QCP : Quality Control Process chart
- (#3) ESQUT : Evaluation of Software Quality from Usre's viewpoint

2. IMAPにおける品質アプローチ

2-1 品質管理工程図~QCP

IMAPではソフトウェア生産の分業化・工業化を目指しているが、このためには工程作業の明確化、開発管理の

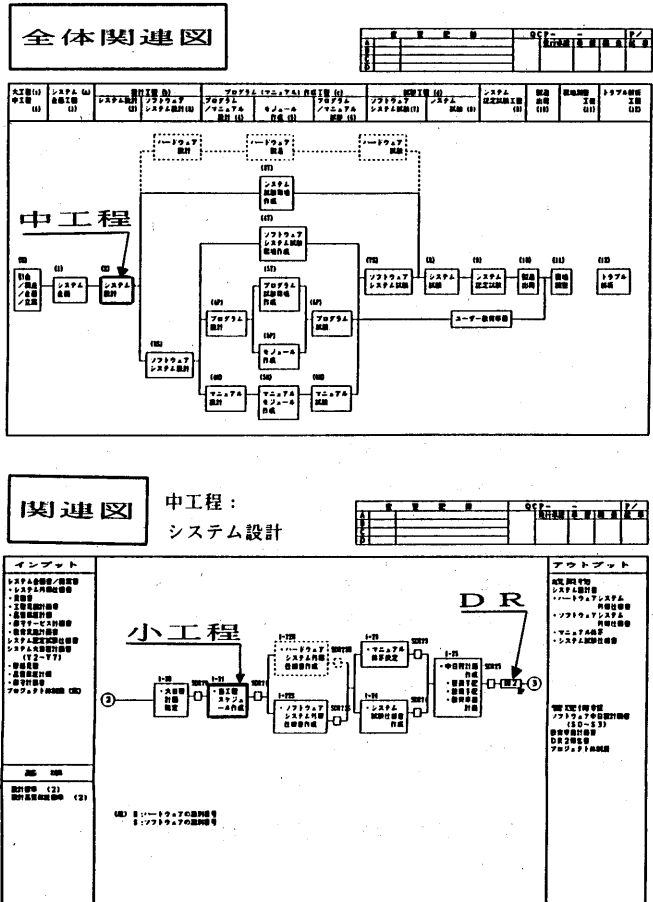


図1 QCP概要

分離、管理の定量化・可視化の推進が必須である。当社ではこの基盤として品質作り込みを意識した品質工程管理図QCPを規定している。QCPは、ウォークスルー・モデルをベースとし、以下の事項について定めることにより、工程内での品質作り込みを進めるとともに、工程の節目での品質管理の徹底を目指すものである。

- 工程作業・成果物を定義し、工程の明確化を図る
- 品質確認ルーチンとしての公式審査DR(デザインレビュー)、WT(ウォークスルー)を規定している。
- 開発組織・体制を規定する。

(注1) デザインレビュー

ここでは、生産工程の各マイルストーン毎に実施される品質の公式確認・承認作業をさす。

2・2 ソフトウェア品質評価システム～ESQUT

ソフトウェア品質評価システムESQUTは、図2に示すように、ソフトウェア・ライフサイクルを通して各工程での全ての成果物（プロダクト）および各工程の作業プロセスおよびテスト結果の品質を定量評価・支援するものであり以下のような機能・特徴をもつ。

- (1) 各工程の各成果物ごとに、その品質を測定・評価する手法・ツール群より構成される。
- (2) 各ツールは、測定する成果物に合せた品質メトリクスを採用し、品質の定量化・可視化をおこなう。
- (3) 各工程での作業プロセス自体の品質についても定量化・可視化の手法を導入する。また、テストについても同様に定量化・可視化の手法導入をはかる。
- (4) 開発中のソフトウェアに関しての適用が可能であり、品質の作り込みによる品質コントロール等が容易となる。

のESQUTシステムについては、現在、プログラム設計工程での設計書（モジュール内部仕様書）を測定・評価するESQUT-TFF、モジュール作成工程でのソースコード（C言語）を評価するESQUT-Cの2つの手法・ツールが実現されており、それぞれモジュール毎の品質定量化とともに、タスクレベルでの品質傾向・特性を把握することが可能である。

2・3 QCPにおけるESQUTの運用

ソフトウェア品質評価システムESQUTの運用は以下に示すようにQCPにそった運用を前提としている。

- (1) QCPで定義されている各工程での品質確認DRの際に、その工程での成果物・作業品質の指標値としてESQUTにより測定・評価した値を利用した工程別品質管理を行う。この場合には、ESQUTの利用対象者は品質管理部門などが考えられ、主としてタスク・レベル、システム・レベルでの品質評価が中心となる。
- (2) 成果物への品質作り込みをより積極的に行う立場からは、ソフトウェア作成担当者がその作業の随所でESQUTによる品質測定・評価を自ら実施し、自らの成果物の品質レベルを把握することで品質の作り込みを誘導する。この場合の品質評価は、モジュール・レベル、タスク・レベルでの実施が中心となる。

3. ソフトウェア設計工程での品質評価

3・1 ソフトウェア設計の標準化

IMAPでは、部品化・再利用を促進し、設計の理解性・保守性を高めるためソフトウェア設計手法として設計記述法 TFF (*4) を採用している。TFFは、モジュール内部仕様・外部仕様、データ仕様等の作成・設計の標準化を目指すものである。TFFはその支援ツールとして TFFED(*5) が用意されている。図3はこのTFFEDにより作成したモジュール内部仕様書の例である。

(*4) TFF : Technical Formula for
Fifty steps/module design

(*5) TFFED : TFF EDITER

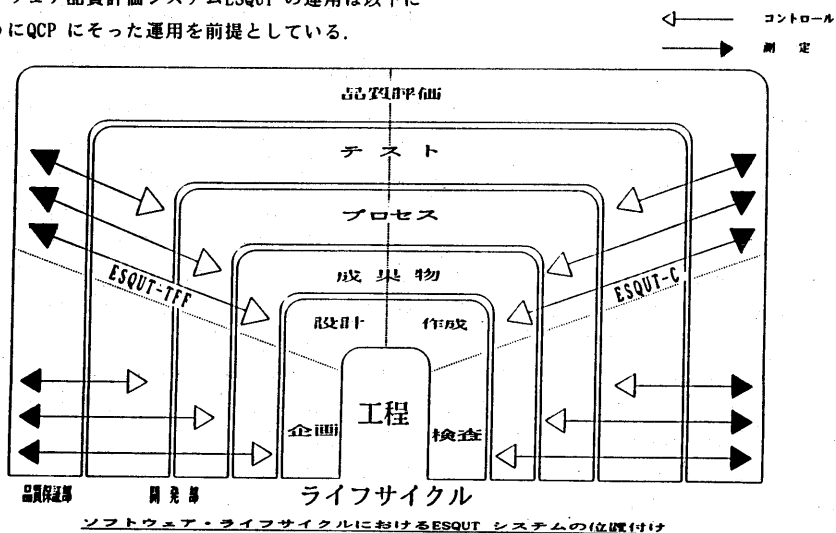


図2 ソフトウェア品質評価システム～ESQUT概要

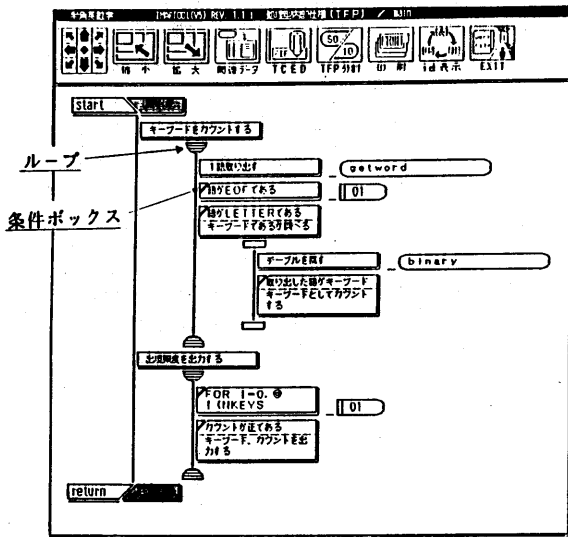


図3 TFF-Cシート

3・2 ソフトウェア設計品質評価システム～ESQUT-TFF

ESQUT-TFF は、TFPED により作成されたモジュール内部仕様書の品質を、品質メトリクスにより定量化するツールである。

ESQUT-TFF では、品質メトリクスとして、ソフトウェア設計の理解性・保守性の観点から特に、モジュールの独立性・単純性等に着目し、TFF による内部仕様書に記載されている図形言語から形式的に判断のできる、以下の6つを採用している。

- 条件ボックス数
- ループ数
- 条件ボックス・ネストレベル
- ループ・ネストレベル
- ループ内の条件ボックス
- ボックス数

従来、モジュールの制御構造に関係する品質指標としては、Halstead, McCabe 等の品質指標が提唱されているが、これらは、いずれもソースコードに関するものである。このESQUT-TFF では設計段階でモジュールの制御構造等に関する品質を評価することができる。これまでの研究により設計にTFFを用いた場合には、モジュールの制御構造等については、そのソースコードとの対応が比較的良好とれることがわかっており、設計段階でソースコードの品質が予測できる意義は大きい。ESQUT-TFF の機能を以下に列挙する。

- (1) 設計品質を形の上から測定することが可能
- (2) 設計品質の低い、危険度の高いモジュールを探索することが可能
- (3) 設計段階での品質改善・品質作り込みを誘導することが可能
- (4) モジュール設計品質計測の結果をもとにタスク全体の設計品質についても、その傾向・特性把握が可能である。

4. ESQUT 試行概要

4・1 試行概要・目的

プログラム設計工程での品質評価技法の研究の一環として、当社内のソフトウェア・プロジェクトに関し品質評価システムESQUTの試行を実施した。このプロジェクトのプログラム設計工程で作成されたソフトウェア内部仕様書(TFF-Cシート)をESQUT-TFFにより品質計測し、これにより収集された各メトリクス値について以下のような分析・検討を実施した。

- (1) ESQUT-TFF メトリクスの特性・評価
- (2) ESQUT-TFF メトリクスの標準値検討
- (3) 総合評価メトリクスの導出・検討

4・2 試行対象

今回の試行は、比較的規模の大きい開発プロジェクトを対象とした。ソースステップ数は約50万ステップ(C言語90%,アセンブラ10%)であり設計は基本的にTFFを用いており、対象としたのはこのうちの7つのタスクである。表1にこの対象タスクのモジュール数その他を示す。

ESQUT-TFFを用いてこの7つのタスクの品質計測を実施し、このデータをもとに各メトリクスの平均値・分散標準偏差、分布パターン、相関・回帰特性等のメトリクス特性を分析、これよりメトリクス標準値の検討を行った。今回の試行対象プロジェクトで得られたデータの検証のため、今回の試行対象とは異なり、当研究所内で作成された中規模ソフトウェアに関するデータを対比データとして利用した。

また、ESQUT-TFFで用いている6つのメトリクスを組合わせた総合評価メトリクスの検討を行った。この検討については、試行プロジェクトにおける設計の修正・変更回数に対比データとして利用した。

表1 試行対象プロジェクト概要

タスク名	解析対象 モジュール数	総ステップ数 K step
Z11	59	2.9
Z12	54	2.3
Z31	45	2.3
Z32	65	2.7
Z34	73	3.0
Z35	14	0.4
Z36	66	2.5

表2 ESQUT-TFFメトリクス基本統計量

	平均値	分散	標準偏差
条件ボックス数	2.89	11.13	3.29
ループ数	0.33	0.40	0.63
条件ボックスネストレベル	1.05	0.98	0.99
ループネストレベル	0.28	0.27	0.52
ループ中の条件ボックス数	0.91	3.58	1.88
ボックス数	10.62	56.41	7.18

表3 ESQUT-TFFメトリクス基本統計量

	平均値	分散	標準偏差
条件ボックス数	3.48	8.29	2.88
ループ数	0.50	0.69	0.83
条件ボックスネストレベル	0.75	0.39	0.63
ループネストレベル	0.32	0.26	0.51
ループ中の条件ボックス数	1.25	3.96	1.99
ボックス数	10.27	38.19	6.18

(当社 システム・ソフトウェア技術研究所・実験データより)

5. ESQUT 試行結果

5.1 ESQUT-TFF メトリクス特性

(A) 基本統計量

分析結果

ESQUT-TFF で採用している6つのメトリクスについてその平均値・分散・標準偏差を表2に示す。この値は7タスクそれぞれに求められた値を平均したものであるが、各タスク間での差異はあまり見られなかった。また、今回の試行プロジェクト以外のデータ(表3)と比べても、多少の差は見られるが、極端な違いは見られなかった。これより、各メトリクスの平均値・分散・標準偏差は、メトリクス固有の特性値と考えられ、これはTFFによる設計の特性ともいえる。

品質管理への利用

・モジュール評価

ここで得られた表2に示した各メトリクスの平均値・分散・標準偏差の値を基準値として、この値と極端に異なるメトリクス値をもつモジュールは設計品質の観点で基準からはずれた、何らかの問題をもつものであると推測することができる。

・タスク評価

タスク全体として、そのタスクに含まれる全てのモジュールの各メトリクス値をもとに、各メトリクスの平均値・分散・標準偏差をもとめ、この基準値との差を見ることで、タスクの品質傾向を知ることができる。

(B) メトリクス分布パターン

分析結果

図4は、表2の平均値・標準偏差をもとに描いたメトリクスの分布パターンである(正規分布)。1タスク100モジュールとして、横軸がメトリクス値、縦軸にモジュール数をとったヒストグラムである。この図では、100モジュールのうち条件ボックス数が0~3のモジュールが約35、3~6のモジュールが40、6~9のモジュールが20あることがわかる。

これに対し、実際のメトリクス分布を比較しても、ごく近い分布パターンをもっており、メトリクス値分布は正規分布に近いものと考えられる。また、これは今回の試行プロジェクト以外のメトリクス分布パターンとも一致するものである。

図4では、メトリクスとして条件ボックスを例にとったが、これ以外のメトリクスについても、それぞれ固有の分布パターンを持つことがわかった。

品質管理への利用

・タスク評価

あるタスクについてメトリクス分布ヒストグラムを

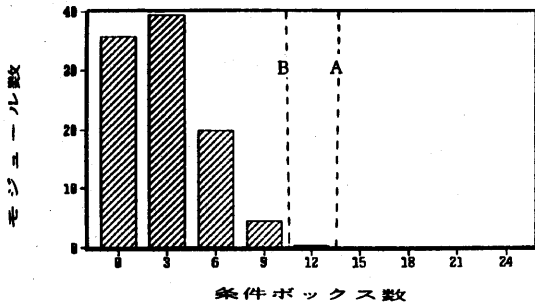


図4 メトリクス値分布パターン
(条件ボックス数)

作成し、その値分布形態が上記とは異なると判定された場合にはそのタスクの設計について品質面で他とは異なる傾向があるとの判定が可能となり、タスク全体の異常を検知することができる。

・モジュール評価

分布が正規分布に従う場合には、その管理限界値として (平均値) + (標準偏差) × 3 が一般的なものとして利用されている。図中の線Aがこの値に相当する。また、線Bは (標準偏差) × 2 の値に相当するこれより、線A、Bで示された値を管理限界値として利用することにより、各メトリクスの観点から設計に問題があると考えられる危険モジュールを指摘・管理することが可能である。

(C) メトリクス相関

分析結果

表4はESQUT-TFFの6つのメトリクス間の相関係数を示したものである。この表では対象7タスクについて、それぞれメトリクス相互の相関をもとめ相関係数が0.7以上のものについて示したものである。例えば、条件ボックス数とボックス数の関係をみると、7タスクのうちこの組み合わせについて相関係数が0.9以上のタスクが1タスク、0.8~0.9のものが2タスク、0.7~0.8のものが3タスクあったことを示しており、7つのタスクのうち6つまでがこのメトリクスの組み合わせについて相関が高いことがわかる。表3より、以下のメトリクスの組み合わせについては、そ

の相関が高いことがわかる。

- 条件ボックス数 ~ 条件ボックス最大ネストレベル
- 条件ボックス数 ~ ボックス数
- ループ数 ~ ループネスト最大レベル
- ループ数 ~ ループ中の条件ボックス数
- ループネスト ~ ループ中の条件ボックス数
最大レベル

ここに示したメトリクス相互の相関についても、このメトリクスのもつ特性の1つであり、今回の試行プロジェクト以外のプロジェクトに関しても同様の傾向があることが確認されている。

品質管理への利用

・タスク評価

各メトリクスの相関をとった場合、上記で指摘したメトリクスの組み合わせについて実際に相関が高くなっているかどうかを判定することで設計の問題点を検討することができる。すなわち、このメトリクス同士の相関については、メトリクスの特性であるとともに、TFFを標準的に用いた場合には、その特性として必然的に現れるものと考えられ、この相関関係が成立しないということは、TFFによる設計それ自身に問題があるケースが多い。

(D) メトリクス回帰分析

分析結果

この相関の高いメトリクスの組み合わせについて回帰分析を行った結果が表5である。表5では、各タスク毎に求めた結果を平均して表示してある。各タスクの平均値からの値のずれはどのメトリクスの組み合わせについてもごく僅かであり、また、今回の試行プロジェクト以外の

表4 ESQUT-TFFメトリクス間の相関

	条件ボックス数	ループ数	条件ボックス ネストレベル	ループ ネストレベル	ループ中の 条件ボックス数	ボックス数
条件ボックス数						
ループ数	△					
条件ボックス ネストレベル	○○○○ ○△△					
ループ ネストレベル	○	●●●● ●○○				
ループ中の 条件ボックス数	○	●●○ △△		●●○○ △△		
ボックス数	●○○ △△△	△	○	○	△	

● 0.9 < r ○ 0.8 < r < 0.9 △ 0.7 < r < 0.8

プロジェクトに関しても同様の傾向があり、この値をメトリクス相関～回帰特性基準値として考えることができる。

品質管理への利用

・タスク評価

タスクレベルで求まる各メトリクス間の回帰分析・特性値とこの回帰特性基準値と比較することで、そのタスクの品質傾向を把握することができる。

・モジュール評価

図5は試行プロジェクトのあるタスクに関してボックス数～条件ボックス数の組合せについての散布図を描き、合せて回帰直線（線A）を描いたものである。また、線Bはボックス数と条件ボックス数の等しい点を結んだものである。この図で示した領域C（線A、Bに囲まれる領域）の範囲に属するモジュールはボックス数に対し条件ボックスの数が多くより複雑な設計がなされていることがわかる。この組合せ以外についても散布図に回帰直線を描いた場合に、その回帰直線から極端に離れるモジュールについては、何らかの問題があるものと考えられる。

表4 回帰分析結果

メトリクス1 X	メトリクス2 Y	X係数(α)	Y係数(β)
条件ボックス数	条件ボックス 最大ネストレベル	0.238	0.377
ループ数	ループ 最大ネストレベル	0.777	-0.031
ループ数	ループ中の 条件ボックス数	2.820	0.014
ループ 最大ネストレベル	ループ中の 条件ボックス数	3.520	-0.049

回帰式: $Y = \alpha X + \beta$

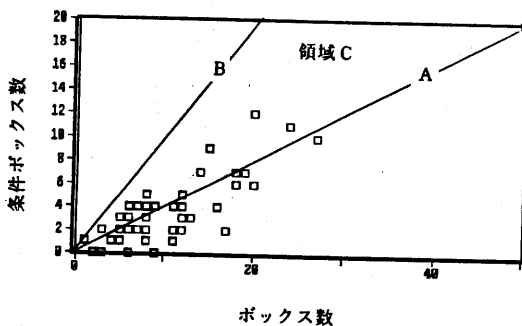


図5 メトリクス相関図
(ボックス数～条件ボックス数)

5・2 総合評価指標

ESQUT-TFF で採用している6つのメトリクスを組合せてモジュールのもつ複雑さ・規模を評価する総合評価指標を考える。このような総合指標については、その値のもつ意味およびその数値から指標の意味する特性がイメージできることが重要である。ここでは、これらを踏まえて以下のような総合評価指標を提案する。

総合評価指標の定義

本指標は以下のようにして定義される。

- ① ESQUT-TFF の6つのメトリクスについて、その平均値・標準偏差をもとに偏差値をもとめる。
- ② この偏差値をもとにXY座標系に以下の4点をとり、四角形を描く。

- 第1点 (条件ボックス数, ボックス数)
- 第2点 (ループ数, ボックス数)
- 第3点 (ループ数, ループネスト最大レベル)
- 第4点 (条件ボックス数, 条件ボックスネスト最大レベル)

但し、ループ数、ループ・条件のネストレベルは、ともに各軸マイナス方向にとるものとする。

- ③ この図形が囲む面積(X1)を計算し、この値を指標値とする。指標の定義より、各メトリクスが全て平均値であるモジュールはX1の値は10000となる。

評価例

図6は、比較的複雑と考えられるモジュール(ボックス数12, 条件ボックス数4, ループ数3, 条件ボックス・ネスト最大レベル2, ループ・ネスト最大レベル2)について、この指標を示したものである。X1の値は17949となっており、標準値の10000よりはかなり大きくなっている。また、四角形の形も、標準の形(正方形)と比較してかなりいびつになっており、バランスが良くないことがわかる。

この指標では、その定義よりモジュールの規模・複雑さが増大するに従い、四角形が大きくなり指標値X1の値も大きくなることがわかる。また、設計レベルが平均的なものは形が正方形に近くなるが、バランスの悪いモジュールについては、四角形がいびつになることがわかる。このようにこの指標はモジュールのもつ複雑さ・規模に関して、指標値X1および描かれる四角形を合せて

判断することで、この指標値の意味する特性をとらえることが可能である。

プロジェクトへの適用

表6は、4・2で示した対象プロジェクトのうちの1タスクにこの総合指標を適用した結果である。この表中での修正回数は、設計の変更・修正された回数を示し設計工程でのバグ数とも考えられる。例えば、X1が6000~10000の範囲のモジュールでは、以下のことがわかる。

全モジュール数	33
修正を受けていないモジュール (比率)	55 %
何らかの修正を受けたモジュール (比率)	45 %

表6より、X1の値が10000を越えると徐々に修正を受けるモジュールの比率が高くなるのがわかる。これは、X1の基準値である10000を越えた場合、モジュールの複雑さ・規模の両面から設計に問題を生じる可能性が高いことを示している。以上のようにこの総合指標X1は、設計品質を考えるうえで有効であると言える。

また、総合評価指標X1の図形の形状評価では、図形がいびつでバランスの悪いものほど、修正をうける度合いが高いことがわかった。

6. 結言

品質評価システムESQUTの実プロジェクトへの適用事例を踏まえて、IMAPにおける品質管理へのアプローチについて検討を行った。今回の事例では、特定プロジェクトの設計工程に限ったものであるため、ソフトウェア・ライフサイクル全体にわたるESQUTシステム、QCP全体の検証には必ずしも十分とは言えない。しかし、実プロジェクトに適用し、それをもとに品質指標値の検討を進めたことで今後、今回得られた品質指標値および品質指標値の利用手法等をもとに、ESQUT品質評価メトリクス標準値、並びにESQUTによる品質評価手法の確立さらにQCPで規定するDRWT時の品質評価へのESQUT導入についての指針が得られたと言える。

また、今後の課題として以下の事項を挙げるができる。

(1) ESQUT,QCPについて設計工程以外での試行・適用により、ソフトウェア・ライフサイクル全体にわたる品質管理の仕組みを確立する。

(2) 複数プロジェクトに対して試行を行い、ESQUTメトリクス標準値を始めとする品質指標標準値およびそれによる評価手法を確立する。

(3) 設計段階での品質評価の観点からは、現在のESQUT-TFFでは、モジュール内部仕様書の計測が中心となっているが、今後はモジュール関連図、モジュール外部仕様等の評価手法を実現し、タスク・システムレベルでの品質評価を可能にする。

[参考文献]

- (1) 平山他 “モジュール設計段階における品質評価の一手法”
情報処理学会第36回全国大会 5L-5 1988
- (2) 松村他 “IMAPシステム(2)”
情報処理学会第31回全国大会 4F-2 1985
- (3) 綾 他 “ソフトウェア開発の管理手法とその評価”
第8回 ソフトウェア生産における品質管理シンポジウム
A-2 1988 日科技連

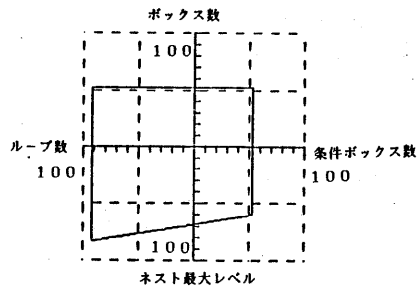


図6 総合評価指標値～X1

表6 総合評価指標 X1と修正回数との関係

X1	モジュール数	修正回数					修正モジュール数	モジュール修正比率 %
		0	1	2	3	4		
0-3600	0	0	0	0	0	0	—	
3600-6400	2	0	1	1	0	0	3	
6000-10000	33	18	10	5	0	0	20	
10000-14400	13	8	5	2	0	0	9	
14400-19600	6	1	0	3	1	1	0	
19600-25600	1	0	1	0	0	0	1	
25600-32400	0	0	0	0	0	0	—	