

定量的マネジメントのための公開データ利用ガイド

付録

組織活動情報

平成 22 年 3 月

ソフトウェアメトリクス高度化プロジェクト
プロセスメトリクス WG

(空白ページ)

はじめに

メトリクスによる定量的マネジメントは、プロダクト（成果物）、プロジェクトの進捗、プロセスの質を可視化し、プロジェクトの工数、プロダクトの規模、プロダクトの品質の予測を可能にする。また、定量データを蓄積することで、非現実的な計画を立てるのではなく、実績に裏付けされた実現可能な計画を立案できるようになり、妥当性の高い改善施策を提案し、その組織内展開を推進することが期待できる。

公開データは、定量的マネジメントの経験が浅い組織では定量的マネジメントを理解、導入するうえでの参考情報となり、既に定量的マネジメントを実践している組織では業界内における自組織のパフォーマンスの把握・比較し、自組織の強みや弱みを把握し、重点的に改善すべき対象の特定に利用することができる。国内で利用可能な公開データは複数存在し、定量的マネジメントにとって海外でも類を見ない好環境にあるといっても過言ではない。しかしながら、それらの位置づけや内容を比較・整理した情報はなく、活用しづらい状況となっていたことは否めない。

こうした状況に対して、ソフトウェアメトリクス高度化プロジェクト プロセスメトリクス WG では、ソフトウェアを共通的に分析、評価できる環境の実現を目標とし、定量的マネジメントへの公開データの円滑な活用を目指し、国内で利用可能な公開データを整理するとともに、定量的マネジメントでの公開データの活用プロセスと、その際の留意点をまとめ、「定量的マネジメントのための公開データ利用ガイド」を作成した。

本付録は、本ガイドで取り上げた国内の主な公開データ提供者、及び定量的マネジメントに係る調査を行っている以下の組織活動内容について、情報を取り纏めたものである。

表 国内の主な公開データ提供者と定量的マネジメントに係る調査組織

公開データ提供者	社団法人 日本情報システム・ユーザー協会
	独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター
	財団法人 経済調査会
定量的マネジメントに係る調査組織	社団法人 情報サービス産業協会

目次

はじめに

基本的な用語の説明.....	1
記述内容.....	2
1. 社団法人 日本情報システム・ユーザー協会：ソフトウェアメトリックス調査.....	3
1.1 公開データ提供の背景と目的.....	3
1.2 ベンチマーキングプロセスの概要.....	5
1.3 他活動・標準との関係.....	14
2. 独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター：ソフトウェア開発データ白書.....	15
2.1 公開データ提供の背景と目的.....	15
2.2 ベンチマーキングプロセスの概要.....	17
2.3 他活動・標準との関係.....	23
3. 財団法人 経済調査会：ソフトウェア開発に関する調査票（受託者向け）集計結果その2.....	24
3.1 公開データ提供の背景と目的.....	24
3.2 ベンチマーキングプロセスの概要.....	26
3.3 他活動・標準との関係.....	30
4. 社団法人 情報サービス産業協会：定量的マネジメントに係る調査.....	31
4.1 活動概要.....	31
4.2 信頼性向上のベストプラクティスを実現する管理指標調査.....	31
4.3 情報システム信頼性向上のための管理指標活用の普及拡大調査.....	36
5. 付録.....	41
5.1 独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター：ソフトウェア開発データ白書.....	41

基本的な用語の説明

- 公開データ

ソフトウェアの開発、保守、運用プロジェクトに関するメトリクスの値を複数の組織から収集し、統計分析等の処理を経て公開された情報。

- 公開データ提供者

本ガイドでは、公開データを作成し、公開する者、組織。

- 公開データ利用者

公開データを利用する者、組織。

- 収集データ

公開データの作成のために複数の組織から収集されたメトリクスの値。

- データ提供者

公開データの作成のために公開データ提供者に対して、メトリクスの値を提供する者、組織。

- ベンチマーク

本付録では、公開データ提供者が公開データを作成するためにデータ提供者からデータを収集し、公開データとして統計分析等の処理をすることを示し、一連の手順をベンチマーキングプロセスと呼ぶ。また、ベンチマーキングプロセスで作成されたデータをベンチマーキングデータと呼ぶ。

注意：各章での組織活動情報は、当該組織で利用されている用語で記述している。このため、同音・同意味の語句でも、組織により異なる文字表記が付されていることがある。

記述内容

公開データ提供者（JUAS, IPA/SEC, ERA）に係る記述内容は、下表の構成にてそれぞれの組織で記載された内容となっている。

表 公開データ提供者の記述内容

章.節	項	内容
X.1 公開データ提供の背景と目的	X.1.1 背景	公開データ提供者が公開データを提供することとなった経緯等
	X.1.2 目的	公開データ提供者が公開データを提供する目的
	X.1.3 利用対象者	公開データ提供者が想定している公開データの利用対象者(公開データ利用者)
X.2 ベンチマーキングプロセスの概要	X.2.1 ベンチマーキングの基本概念(方針)と定義	公開データを提供するにあたり、データ提供者からデータを収集し、公開データを作成する上での方針
	X.2.2 ベンチマーキングの実施方法	方針のもとで構築、実施しているベンチマーキングプロセス(データ収集、分析、報告へ至るプロセス)
	X.2.3 データ収集、蓄積及び交換の方針	収集データをあつめる方法、収集期間、頻度
	X.2.4 ベンチマーキングデータの要件	収集データの精査(匿名化、データ品質の確保、データセットの偏り防止など)し、リポジトリ(参照用のプロジェクトデータの格納場所)、保管、管理方法
	X.2.5 結果の報告方法	公開データの提供形態(報告書、システム等)、頻度
X.3 他活動・標準との関係		他の活動、標準との関係

また、JISA では調査報告書別に概要を記載している。

1. 社団法人 日本情報システム・ユーザー協会：ソフトウェアメトリックス調査

1.1 公開データ提供の背景と目的

1.1.1 背景

信頼性の高いソフトウェアを効率的に実現するための手法、ツール、データベース等の提供と普及を目的とした組織として、ソフトウェア・エンジニアリング・センター（SEC）が、独立行政法人情報処理推進機構（IPA）に設置されたのが 2004 年 10 月。現在では、定量マネジメント、ビジネス・プロセス改善、要求・アーキテクチャ、高信頼ソフトウェア、それに、組込み系プロジェクトを含む各領域での調査研究活動がなされているが、発足当初は、そのベースとなるデータを収集し、ベンチマーキングを実現するためのデータ収集活動にも積極的だった。

社団法人日本情報システム・ユーザー協会（JUAS）は、そのソフトウェア品質の評価、保証のためのデータ収集と分析、蓄積、提供の方向性に賛同し、仮説の設定や調査票の設計にも参画し、特に情報システムユーザ企業を対象としたデータ収集と分析を担当した。

その実行部隊として、2004 年 6 月より「システム開発生産性評価プロジェクト」を立ち上げ、開発問題の解明を目指した調査を実施した。開発プロジェクトの工期・品質・生産性に焦点を当てた調査分析であったが、初年度に、予想以上の知見を得ることができた。

2 年目からは、「システム開発保守 QCD 向上プロジェクト」が継承し、システム保守についての管理指標もあわせて調査分析を行った。保守品質の確認のためには、変更要求書に対して一回で正解が得られることを理想とした評価尺度を設け、回答を求めた。6 年目を迎え、保守についても、仮説検証型の質問設定が可能になり、保守作業の実態と対策が解明されつつある。

2006 年からは、システム運用に関するデータも収集し、ソフトウェアライフサイクルにおいて網羅的なデータ収集、分析が実現し始めている。運用パートは、激変する技術革新の中で、運用担当者が成果を可視化できるための評価尺度を意識し、設定、収集している。この分野は、「システム運用研究会」が 2006 年から、アンケート作成、回答のレビューを担当してくれている。

これらの調査分析は、「ソフトウェアメトリックス調査」として毎年まとめて報告しているが、JUAS で、別途、実施している「企業 IT 動向調査（回答数：約 1000 社）」の中でも、メトリックスに関する設問を設けて併用し、合わせて評価できる形をとっている。

昨今は、非機能要求の明確化が強く求められるようになった。しかし、開発の段階で準備された非機能項目のほとんどは、運用フェーズにならないと評価できない。そこで、今では、開発、保守、運用の担当者間コミュニケーションの改善に向けた質問項目も検討し、全体的な定量管理と PDCA が回せる、言い換えると、採取データをベンチマーキングして改善アクションに結び付けられるような調査、及び、分析結果の提示を心掛けている。

それにより、調査に協力する意義、例えば、

- ・自社データを提示すれば他社比較ができるようになる
- ・回答データの統計的分析から有効なノウハウが抽出され確認できるようになる
- ・これから着手するプロジェクトには望む前から反映、活用できるノウハウが得られる

などの実感を回答者に得ていただき、成果への信頼性が高まるとともに、興味を持ってくれるように

なった。JUAS の質問と回答内容を参考に、各社別に素晴らしい分析を始めた企業も登場してきた。

今後は、データの背景となる属性やプロジェクトの性格を分類することで、データばらつきの根拠を確認し、パターン別(たとえば、重要インフラ系、基幹系、情報系等による差異など)に、評価すべきメトリックスと目安値を提示できることを検討していく。

1.1.2 目的

企業の情報システムは、企画、開発、保守、運用、利活用等の各フェーズにおいて多くの課題を抱えており、それぞれの課題も複雑に関連し合っている。例えば、JUAS「企業 IT 動向調査」では、システム開発において、工期や予算が守れている大型プロジェクトがおおよそ半分しかないという結果も出ている。

この改善に向け、JUAS では、経験則を体系化する活動に加えて、アクションに対する効果を測定・評価できる尺度の設定を目的に、定量的な管理を実現するための調査研究を行うことにした。

特に、品質については、結果品質とそれを作り込む工程での品質を、それぞれプロダクト品質、プロセス品質として捉え、前者の尺度や目標を実現するために後者がどう関与するかの観点から、仮説検証型で調査分析を行ってきた。

言い換えると、各フェーズで何をしたのかという工程での質の評価尺度を追及することを「(ソフトウェア開発における)プロセス志向」、成果物の評価尺度や品質目標を追求する方向性を「(プロセス志向に対して)プロダクト志向」とし、プロダクト志向あつてのプロセス志向をソフトウェア管理の基礎において考えた。

そのために、データで事実を語れるよう、開発、保守、運用の評価尺度(ソフトウェアメトリックス)の項目に何を採用し、どのようにデータを収集して分析すれば、ソフトウェアのプロファイルに応じた効果的な管理ができるか、というガイドの提供を目的とした。

本成果をより多くの方が活用し、稼働中または計画中のソフトウェア・プロジェクトに自社なりの尺度や目標値を設定することで、ソフトウェア品質が向上し、ひいては、事業品質の向上に寄与できる報告書を目指している。

1.1.3 利用対象者

ソフトウェア関連業務は、アウトソーシングしていても、ユーザの管理責任が大きい。

非機能要件の証明も、受け入れ後、運用フェーズにおいて初めて確認されるものが大半である。

また、運用環境は、システム構成の変化、データ量の増加を含め、開発時には想定しえなかった要因により障害が発生するという特徴もある。

システムの利用者、社会に対して最終責任を持つのはユーザ企業であり、それを開発しているベンダや運用を受け持っている業者ではない。JUAS のソフトウェアメトリックス分析は、この視点ありきの活動であり、その点に着目して成果を活用いただきたい。

【想定対象】

システムユーザ企業の開発責任者、同プロジェクトマネージャ、及びシステムベンダ企業のプロジェクトマネージャ

1.2 ベンチマーキングプロセスの概要

1.2.1 ベンチマーキングの基本概念（方針）と定義

（a）基本概念（方針）

ソフトウェアメトリックスのデータ収集と分析を始めるに当たり、いくつかの方針を示して協力者の了解を得た。

（a-1）分析に利用した指標

固定観念を捨てること

現在、流行している考え方のみを採用するのではなく、様々な角度から分析できるような測定項目を設定している。例えば、見積りに使う基準測定の単位であれば、FP だけでなく、LOC や人月、価額、データ項目数等の各評価要素を特性比較できるように考えた。

活用しやすい形に整理し、まとめること

分析結果は、可能な限り評価式にて表現すること。その式は対数を活用するようなものではなく、極力、単純な四則演算で答が得られるようなものが望ましいと考えた。

項目によっては、四則演算も使わない、知見を述べた定性的なメジャーも良しとした（Fact ベース）。データ数が少ない場合は信頼度が問題になるので、元の分析結果には信頼度を併記する。

仮説を持って設問を作成すること

実績に基づき経験を踏まえた仮説を、有識者及び実践者の知見により立案し、その仮説の証明に必要な設問を準備する。次にデータを集め、それを基に分析検証する。仮説が証明できなければまた別の仮説を立て検証を繰り返す。このようにして新たな知見を見出していく。

（a-2）開発調査分析方法について

目標値の設定

前年度の分析結果を目標値として活用してもらえよう各社に働きかけることで、各社においては効果性が高まり、調査データとしてはバラツキが減少することを狙っている。

仮説と設問

アンケートの設問の裏には仮説が存在しており、その証明、検証の意識を持ち、調査分析を行っている。一方で、仮説を重んじ過ぎて仮説を絶対視することを防ぐため、クロス分析も多用し、重要な要素を見失わない配慮をしている。

QCD 向上プロジェクトで、設問の過不足と構成、展開の妥当性を検証するため仮アンケートを実施し、本番アンケートに臨んだ。

分析方法

分析方法には、3つの考え方がある。

その1：特性基準値の精度向上を目指す方法

$A = b \cdot c^X$ などの仮説式を立てて係数を求める方法である。

例) 工期と投入工数の関係において 工期 = $2.4 \times (\text{人月})^{0.318}$ の0.318が妥当値かどうか、など

その2：分かりやすい特性基準値を元に、その活用方法を柔軟に求める方法

その1で求められた、何らかの分析結果を基準におき、各プロジェクトでは、その基準との差を意識して利用する方法である。「基準が無いよりは何かあれば一つの目安になる」との見解で基準を利用する方法である。

前出の式は、工期 = $2 \times (\text{人月})^{1/3}$ とすると使いやすくする。

「標準工期は投入工数の立方根の2倍」と覚えやすく、かつ、計算しやすくする。慣れれば暗算で行うことも出来る。

その3：特性を活用する方法 (Fact ベース)

因果関係を統計解析し原因と対策の関係を追求するだけでなく、基本的特性を見抜きその結果を利用する方法である。

上記工期の例でいえば、「当社では標準工期よりも50%短いプロジェクトは破綻するのでそのようなプロジェクトは実施しない」などと活用することである。大まかなデータ分析からでも、このような事実を発見できる。

(a-3) 保守調査分析方法について

保守作業の定義

システム開発を実施し本番に入ったところから、保守作業は始まる。

本調査では、保守作業の範囲を次の5つに定義した。

保守の問い合わせ、基盤整備、是正保守、適応保守、完全化保守
それぞれ作業については、さらに詳細な作業内容を定義し、指標を検討した。

保守理由

保守作業は何故発生するのか、その理由を7種類に整理した。

システムのバグから生じた保守作業

担当者からの要望から生じた保守作業

制度・ルールの変化から生じた保守作業

業務方法の変化から生じた保守作業

経営目標の変化から生じた保守作業

ユーザビリティの変化から生じた保守作業

その他の理由から生じた保守作業

この理由割合は、業種ごとに異なるのではないが、特にカットオーバー時の品質はシステム保守作業負荷に大きく影響するはずであるが果たしてどの程度の影響であろうか、などについて分析する。

保守作業管理

上記理由により発生する保守作業は要求通り実施されているのか。それとも予算や保守作業員の負荷の関係で調整あるいは制約を受けているのか。これには二通りの管理方法がある。

厳しく一件ごとに管理者が必要性を審議し、このシステム保守をしなくても大きな影響が無い場合は実施を制約しているプロジェクト
保守担当者の自主判断に任せているプロジェクト

特に担当者からの要望により生じたシステム保守要望には、無制限に実施できないような制約を設けた企業が多い。システム保守作業に SE をまわすか、新規システム開発要望に SE パワーを割くべきか判断し、目先の使用性には少し問題はあるが、経営の観点からは新規システムに大半の SE パワーを活用する方針を定めて開発に振り替えている企業もある。

システム保守契約形態

期間請負契約

「対象プロジェクトについて何人かを保守契約し問題対応させる場合」

システムの安定度、機能要求の程度、環境からの要請、プログラムの作成方法などの影響を受ける。プロジェクト規模に応じてどの程度の人数がアサインされるのが妥当かなどの分析を行う。

一件ごとの請負契約

「保守作業の要求書をもとに 1 件ごとに見積って作業契約する場合」

もしこの見積費用が高いならば中止もありうる。

上記の組み合わせ

「小規模の案件は期間請負契約内で対応するが、他の新システムが企画されたためにその影響でシステム保守をせざるをえず、かつ、相当な大負荷になることが予想される場合」

通常一件が 5 人日以上作業負荷になるものは、保守作業請負対象からはずして別途見積もっている企業もある。また今期のシステム保守作業を見積もった結果、基本契約で交わした保守作業以上に作業が発生することが予想されるので、今期に限って増員契約を交わすなどの方式を採用している企業もある。

保守作業結果の評価

実施された作業に対する結果評価の観点を列挙する。

依頼された工期は守れたか？

保守後の品質に問題はないか？

稼働率は目標を達したか？

作業工数は妥当であったか？

保守作業組織、指揮体制に問題はないか？

緊急時対応体制は準備されているか？

保守担当者のアサインは妥当であったか？

保守作業で採用している技術は適正なものか？
作業効率及び品質向上対策は存在するか？
予算管理は妥当なものか？
利用者との共同作業目標は守れたか？対策は？
例えば顧客迷惑度指数は確保されたか？
セキュリティ対策は完全か？
問題が生じた場合の報告、説明は妥当なものであったか？
人材育成は継続的に図られているか？

(a-4) 運用調査分析方法について

予備調査から本格調査へ

初年度の運用部分の調査は、予備調査とし、JUAS 運用研究部会を中心とする企業の方々に質問作成の協力を依頼して、質問にも答えていただく限定調査とした。

次年度以降、本格調査とし、回答しやすく、課題を浮き彫りできる質問に変更して実施した。

運用調査の難しさ

その1：データが取られているか？

運用調査の基本になる各種データを収集、蓄積している企業が少ないのが実態のため、まずは、項目の提示と運用実態の把握促進の段階からのスタートであった。

その2：問題の複雑さ

運用問題のベースは稼働率になるが、システムのプロファイルや、システム構成、推進体制などによる差異も考慮する必要がある。

質問内容

運用品質に関する運用調査の質問内容は以下のように分類した。

運用のITガバナンスに関する質問

経営者や社会の要請にシステム運用は応えることが出来ているのか、環境の厳しさに運用組織や活動が追随しているのかなどのITガバナンスをまず確認する。

運用品質

高信頼性システムが望まれているが、その評価尺度は稼働率と稼働品質率である。

稼働率 = (稼働すべき時間 - 障害が発生しシステム停止した時間) / 稼働すべき時間

稼働品質率 = 利用者迷惑をかけた件数 / 稼働資産 (STEP数、残存簿価金額、運用費総額など)
従来の「動いていればよい稼働率志向」から「顧客に正しい情報を提供できたかどうか」を問う稼働品質率志向へ変えてゆかないと利用者からの評価が得られない時代になった。障害対策の評価を複眼でみることになる。

運用価額

運用費は何で決まるのか、自社の運用費用は高いのか安いのかを経営者は質問してくる。しかし各

社で所持資産は異なる、システム運用環境も異なる、システムの安定性への期待度も異なる、そのような多くの変数の中を模索して何とか運用費用総額のモデル作成ができないか、に挑戦したい。

運用作業の内容

最近のシステム運用環境はセキュリティや高信頼性確保のために、多くの知識と経験を必要とする。そのような環境にあって運用作業の中身はどのように変わっているのか、その現代的作業にあわせた人材育成が出来ているのか、身の丈にあったシステム運用が出来ているのか、などの問題も抉り出してみたい。

運用調査の仮説と検証

設定した仮説と検証項目は次表の通り。

仮説	検証
1.SLA が厳しくなると運用費用は高くなる。	SLA の厳しさと運用費用の関係
2.運用費用額と資格取得には関係がある	各種資格と運用費、運用要員の関係
3.会社規模が大きいと教育予算も高い	従業員数、売上高と教育予算の関係
4.運行要員よりも運用管理者の育成には投資が必要	運用、運行、開発別教育予算
5.運用には一定の費用がかかる	運用予算 / 人年、(ハードウェア + 基盤ソフトウェア) / 人年、開発予算 / 人年、売上高 / 人年(業種別)、運用予算 / (社内 + 社外利用端末数)
6.オンライン入力にも相当な費用はかかる	運用予算 / トランザクション数 (社内外)
7.運行要員の費用には相場がある	印刷、サービスディスク、オペレーションの費用 / 人年
8.高稼働率、高速レスポンスタイムの実績を持っているところは、運用管理費用の割合が高い	運用管理費 / 運用費と高稼働率、高速レスポンスタイムの状況
9.一定金額以上の運用費を持っているところは、企画、運用のスタッフ費用も高い	横軸運用費、縦軸スタッフ費用
10.高稼働率を持っているところは、回復時間の目標時間も短い	横軸稼働率、縦軸回復時間 (社内、社外向け)
11.レスポンスタイムの SLA を持っているところは、レスポンスタイムの実績も短い	SLA の内容種類別レスポンスタイム
12.稼働ジョブ数と運行オペレーション要員数には関係がある	横軸オペレーション人数、縦軸稼働ジョブ数
13.サーバの台数が増えると運行要員の数も増える	サーバ台数、ジョブ数 (横軸)、運用及び運行要員数
14.外部委託管理と直営体制の関係は稼働率に現れる	稼働率と直営グループ、外注グループの稼働比較
15.高稼働品質率を維持できているセンターは、費用もかけている	稼働品質率と運用費用基準
16.前提条件を示せば一定の標準運用金額が求まる	運用の要素と価額の比較

(b) 用語の定義

分類	名称	定義
工程	開発工程	JUAS では、開発工程は以下の 5 工程で表している。 (b-1) . 企画 (b-2) . 要件定義 (b-3) . 設計 (b-4) . 実装 (b-5) . テスト テストは単体テスト、結合テスト、システムテスト、受入テスト、運用テストなどのフェーズに分けて実施される。 JUAS では、設計からシステムテストまではこの開発を受託したベンダが行うもの、企画、要件定義と受入テスト以降のフェーズはユーザが実施するものとしている。
工期	標準工期	Barry Boehm 氏は、ソフトウェア開発に必要な期間はそのソフトウェア開発に要する工数(人月)の 3 乗根に比例する、という経験則を明らかにした。JUAS では、JUAS が集めた開発実績からこの工期の統計を求めており、その平均を標準工期と呼んでいる。
	工期乖離度	実際のプロジェクトの期間と前記標準工期との乖離を工期乖離度と呼ぶ。これによりプロジェクトを適正工期、短工期、長工期に分類している。
品質	目標・欠陥率	JUAS がいう「欠陥」とは、発注者であるユーザがベンダから開発済のソフトウェアの納入を受けて、受入テストを開始してから安定稼働に至るまでの間にユーザ側で発見した欠陥をいう。したがって欠陥率とはソフトウェアの規模(KLOC、FP、あるいは発注金額 500 万円)あたりのこの欠陥の割合であり、「目標・欠陥率」はこのソフトウェアの企画段階で立てた欠陥率の目標を、「欠陥率」はその実績をいう。
	欠陥率	
	テストケース密度	ここでいうテストケースは、ユーザが実施する受入テストでのテストケースであり、ベンダが実施するシステムテストのものではない。テストケース密度はこのテストケース数をソフトウェアの規模で割ったものだが、ソフトウェアの規模には KLOC のものと FP のものの 2 種類がある。
	レビュー比率	レビュー比率は、総開発工数に占めるレビュー工数の割合である。レビューには要求仕様のレビューからプログラム・インスペクションまで多くのものが含まれ、発注者であるユーザの作業ばかりではない。しかしレビューはソフトウェアの品質向上のために非常に大切な作業であるので、ユーザが行ったレビューとベンダが行ったものを合算してこの指標を求めることにしている。
	レビュー指摘率	レビュー指摘率とは、前記レビューで指摘した欠陥数の、総開発工数の割合である。
	年間障害発生頻度(役員が認識した件数)	運用時に発生する障害には、いろんなレベルのものがある。あるものは影響範囲が少なくすぐに復旧し、あるものは影響する範囲が広く、甚大な被害を及ぼす。この一定以上のレベルの障害件数を把握しようとするものが、この 2 つの指標である。「役員が認識した障害」とは、あるレベル以上の影響があって IT 部門内だけでなくその担当役員にまで報告したケース、「事業が中断した障害」とは前記役員への報告に加えて、その障害によってその会社の事業推進に何らかの影響が出たケースをいう。厳密にいうと各企業により、あるいは業種により、このレベルは一定していない。しかしここでは、その相違には目をつぶっている。
	年間障害発生頻度(事業が中断した件数)	
評価	ユーザ満足度(全体)	IT 投資の結果の評価には、いくつかの指標を用いることができる。例えば ROI や KPI などは、そのために用いられる指標である。しかし最も優れているものの 1 つに、「ユーザ満足度」がある。これはアンケートやインタビューなどによってユーザから直接評価を得る方法で、5 段階、あるいは 3 段階で結果を取得する。情

分類	名称	定義
		報システムの開発全体に関わる評価に加えて、工程、品質、コスト、マナー、機能、ユーザビリティなどについて、それぞれユーザ満足度を得ることもできる。
保守	保守費対初期開発費比率	この指標は、あるソフトウェアの当初の開発にかけた総費用に対する、各年の保守のためにかけた費用の割合を示す。
運用	稼働率	この指標は、情報システムの稼働すべき総時間に対する、実際に稼働した時間の割合を示す。稼働率は、%で表すことが多い。稼働率と年間に停止してもかまわない時間の間には、以下の関係がある。 稼働率 年間の停止時間 99.999% 5分 99.99% 50分 99.9% 8.6時間 99% 86時間
	運用業務費	年間の、情報システムの保守と運用に要する費用の合計。

1.2.2 ベンチマーキングの実施方法

JUAS では、多くの研究会やプロジェクトが活動している。その中の2つ、つまり「システム開発保守 QCD プロジェクト」と「システム運用研究会」が、このベンチマーキングを行うに当たって積極的に関与している。

JUAS におけるベンチマーキングの実施方法は、表 1-1、及び図 1-1 に示す手順に従っている。

表 1-1 ベンチマーキングの実施手順(表)

手順	input	プロセス	output
1 .質問表の事務局原案の作成	<ul style="list-style-type: none"> 昨年度の評価結果、及び今年度の方針 	<ul style="list-style-type: none"> 昨年度の総合分析時の評価結果と今年度の方針に基づいて、JUAS の事務局で質問表の原案(開発・保守)、及び(運用)別に)を作成する。 	<ul style="list-style-type: none"> 質問表(事務局原案・(開発・保守)、及び(運用))
2 .質問表(事務局原案)のレビュー	<ul style="list-style-type: none"> 質問表(事務局原案・開発・保守)、及び(運用)) 	<ul style="list-style-type: none"> 「システム開発保守 QCD プロジェクト」(開発・保守)と、「システム運用研究会」(運用)によるレビューを受ける。 	<ul style="list-style-type: none"> レビュー済の質問表((開発・保守)、及び(運用))
3 .アンケートの回答要請	<ul style="list-style-type: none"> レビュー済の質問表(開発・保守)、及び(運用)) 	<ul style="list-style-type: none"> JUAS の会員、及び一部会員外の企業に、質問表を送付して回答を要請する。 	<ul style="list-style-type: none"> 回答済の質問表(開発・保守)、及び(運用))
4 .データの精査と分析	<ul style="list-style-type: none"> 回答済の質問表(開発・保守)、及び(運用)) 	<ul style="list-style-type: none"> 開発・保守と運用別に、以下の作業を行う。 <ul style="list-style-type: none"> 匿名性を確保するための処理を行う。 データを精査する。 各種の分析を行う。 	<ul style="list-style-type: none"> 統計値、式、表、図(公開データ)
5 .成果物作成、公表	<ul style="list-style-type: none"> 統計値、式、表、図(公開データ)、分析結果のコメント 	<ul style="list-style-type: none"> 結果を IPA へ報告する。 ソフトウェアメトリクス調査報告書として作成し、会員企業などに配布する。 分析の結果を評価し、来年度の方針を決める。 	<ul style="list-style-type: none"> 成果物、公開データ

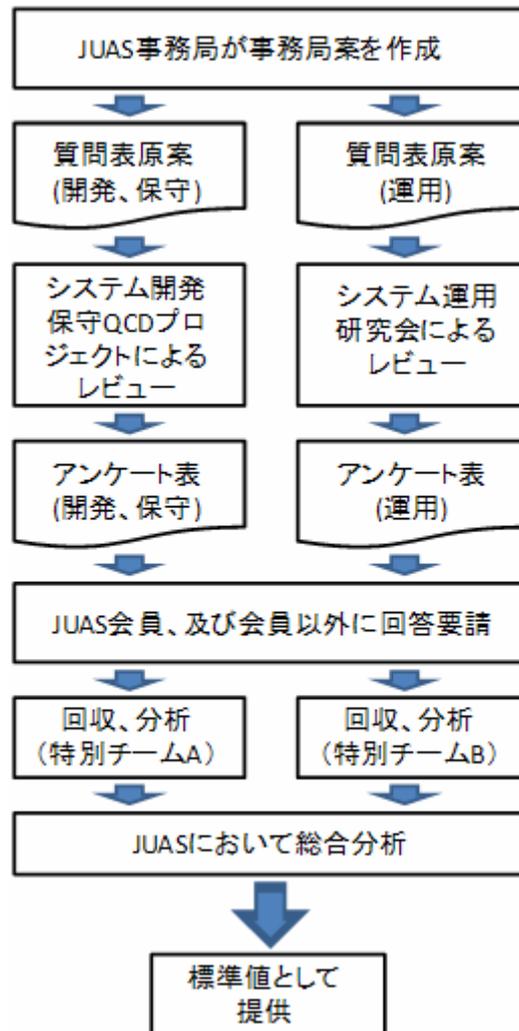


図 1-1 ベンチマーキングの実施手順(図)

なおソフトウェアメトリクス調査以外に JUAS では、毎年企業 IT 動向調査も実施しており、IT 部門と経営企画部門に分けてのアンケート調査とインタビュー調査を行っている。この報告書は、ソフトウェアメトリクス調査とは別の形で報告書を作成している。

1.2.3 データ収集、蓄積及び交換の方針

(a) データ収集対象

データ収集対象は、開発・保守、運用とも、基本的に JUAS の会員企業である。(一部例外的に、JUAS の会員外企業からも回答を得ることがある。)

2008 年秋に実施した調査では、開発は 118 社 / 438 プロジェクト、保守は 105 社 / 231 プロジェクト、運用は 49 社 / 50 のコンピュータ・センターから、それぞれ回答を得た。

(b) データ収集方法

質問票の設問数は、それぞれ、開発：47 問、保守：30 問、運用：84 問である(2008 年度実績)。

データの収集は、既に述べたように調査票を各企業に配布し、それに記入してもらったものを回収するという、アンケート方式を採用している。質問表は毎年秋口頃から配布し、翌年年末に回収している。

また、分析のために掘り下げたい項目やより明確化したり背景を確認したい項目については、ヒアリングも併用し、データの精度や正確性を高めている。

また、前述の QCD プロジェクトやシステム運用研究会のメンバーである有識者及び実践者の皆様に、効率的なデータ収集の仕組みを検討いただき、JUAS の多くの他のプロジェクト参画者にも回答率向上の支援を願う。

集められたデータは情報保護に注意しながら、統計分析の専門家を含めての分析に移る。分析結果の評価も、上述の 2 つの組織のメンバーにレビューしてもらっている。

1.2.4 ベンチマーキングデータの要件

収集するデータには、一定の条件を付けている。

開発関係のデータの条件は、以下の通りである。

- 過去 2 年間に開発を終了したもの
- 開発コストが 500 万円以上のももの
- 新規開発、または改修のプロジェクト

保守関係のデータの条件は、保守発注の責任者が「保守案件と認識したもの」がその条件である。

運用関係には、特別の条件はない。なお運用関係はコンピュータ・センター毎の回答になり、ほとんどが 1 社 1 回答となっている。

また、開発、保守はプロジェクトごとのデータとして、運用は会社ごと(実際は計算センターごと)のデータとして収集する。

1.2.5 結果の報告方法

調査結果の公表(標準値としての提供)の方法は、毎年報告書を印刷物として作成し、JUAS の会員企業各社に配布しているほか、市販も行っている。

昨年度(2008 年度)の報告書は、以下のものである。

経済産業省情報処理振興課、社団法人日本情報システム・ユーザー協会、「ユーザー企業ソフトウェアメトリクス調査 2009 ソフトウェアの開発・保守・運用の評価指標」、社団法人日本情報システム・ユーザー協会、2009年7月。

1.3 他活動・標準との関係

JUAS は毎年、前述の企業 IT 動向調査も実施している。

この調査の内容は、ソフトウェアメトリクス調査ほど具体的に個々の開発や保守などには立ち入らず、企業の IT 部門としての方針、動向、考え方など、より企業経営に近いテーマを対象にしている。

調査の方法もアンケートに加えて、一部の企業にはインタビューも実施しており、報告書は両方の結果を併せて記載している。この報告書も、JUAS の会員企業各社に配布しているほか、市販も行っている。

この昨年度（2008 年度）の報告書は、以下のものである。

経済産業省、社団法人日本情報システム・ユーザー協会、「2008 年度版 企業 IT 動向調査 2009」、社団法人日本情報システム・ユーザー協会、2009 年 5 月。

2. 独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター：ソフトウェア開発データ白書

2.1 公開データ提供の背景と目的

2.1.1 背景

近年の科学技術や産業の目ざましい発展には、IT が大きく寄与してきた。今日では、IT はあらゆる分野に浸透し、経済活動や我々の日常生活に不可欠の存在となっている。市場競争が激しくなるにつれ、短期間で多機能で高性能な製品開発が求められるようになったこともあり、システムへの要求が増大し続けている。そうした状況下で、品質・コスト・納期に関するソフトウェアエンジニアリング手法の確立・普及は、緊急の課題と言える。

特に、最近の情報システムにおけるトラブルの多発により、その安全・安心の確保の要請はますます増大している。システム開発の各工程 / 段階に科学的・工学的に信頼のおけるマネジメントを実行し、システムトラブルなどの発生を最低限抑止することが必要である。

また、プロジェクトのスケジュール遅延の結果、ソフトウェアエンジニアリングとは程遠い人海戦術的な対処方法で対応する場合もいまだに多く残っている。その一因として、見積り方法の問題や見積りの根拠となる生産性の見誤りなど、定量データに裏付けされたマネジメントが十分に実施されていないことがあげられる。

2.1.2 目的

このような背景のもと、IPA/SEC（独立行政法人情報処理推進機構 / ソフトウェア・エンジニアリング・センター）ではソフトウェア開発の課題を改善するため、製造業など他の産業では常識となっているプロセスの標準化、定量的アプローチによる科学的マネジメントなどの普及拡大を進めている。特に、定量化に関しては、IPA/SEC 発足の 2004 年 10 月以来、定量データの定義をはじめ、データの収集、精査、分析を継続して行っている。その結果として、ソフトウェア開発関係者間で共有できるデータの整備が進んできている。

取り組みの目的のひとつは、複数の企業から様々なプロジェクトデータを新たに毎年収集することで、「定型化された統計分析を毎年継続し、モノサシとしての精度を高めていくこと」、もうひとつは「特有の課題やテーマに応じて分析の対象を拡張し、新たなモノサシや課題抽出の切り口を提案すること」であり、ベンチマークとそのデータを提供している。IPA/SEC がこのような取り組みを続けることで、定量データに基づく科学的マネジメントの普及・拡大に少しでも貢献できることを願っている。さらに、継続的にある一定の視点からの観察を続けているため、以前と比較してその差異を分析することが可能で、世の中のトレンドを把握するベースになると考えている。

なお、定量データの開示に際しては、データが持つ様々な背景を考慮して慎重に結論を出す必要が

あるため、ただちに基準値を示すことはしていない。

2.1.3 利用対象者

企業の経営層（ユーザ企業、ベンダ企業）の方々

ユーザとベンダ間でソフトウェア開発に関わる事象の共通認識の形成のために、規模、工期、工数、信頼性などの基礎情報を提供する。ユーザ企業では、ソフトウェア開発プロジェクトにおいて使用する経営資源に関わる基礎情報として、本書のデータを参照していただければと思われる。また、ベンダ企業では、プロジェクトを成功に導くための基礎情報としての利用をお勧めする。

業務部門、情報システム部門の責任者の方々

ソフトウェア開発現場で、データ収集・定量的管理・精度向上等の取り組みの啓発、プロジェクトを成功させるための基礎情報として活用することをお勧めする。

プロジェクトマネージャ、プロジェクトリーダーの方々

ソフトウェア開発プロジェクトを成功させるためには、定量データを用いたプロジェクト管理の推進が重要である。それには、プロジェクトの特性条件を定量的に把握し、規模、工数、工期、品質を見積りまたは予測し、プロジェクト中の工程ごとに定量的に把握して制御や予測を行うことが望ましい。その際、自プロジェクトデータとの比較や参照を行うなどの形で、本書の情報を活用いただきたい。

プロジェクトマネジメントオフィス、品質保証部門の方々

本白書の情報を参考に自社プロジェクトの定量データベースの構築や、自社プロジェクトのベンチマーキングを行っていただければと思う。IPA/SEC の収集データ項目及び定義（付録 A、B）をフルに活用していただきたい。

2.2 ベンチマーキングプロセスの概要

2.2.1 ベンチマーキングの基本概念（方針）と定義

（a）基本概念（方針）

提供されているデータはその組織の力量を示すものであり、また、各プロジェクトは発注者であるユーザのシステムを明確にする可能性を含んでいる。そのため、以下の規則の元、統計的手法により処理したデータのみを公表して匿名性を確保し、かつ、分析の品質を確保している。

- ・分析結果は有識者を交えて、レビューを行い、公表データを精査している。
- ・SEC 内に機密室を用意し、収集データを管理、アクセス者を制限して、公表データ以外はベンチマーキングの利用に提供しない。
- ・基本的基準（2.2.4 に示す）を満足しない分析は公表しない。
- ・分析結果のコメントは定めた基準（2.2.4 に示す）を基に評価する。

（b）用語の定義

提供データ項目を組み合わせて定義した導出指標の名称と定義を以下に示す。なお、提供データ項目は付録参照。

「導出指標」は JIS X0141：2004 ソフトウェア測定プロセスでは「導出測定量」といわれている。

分類	名称	定義
規模	実効 SLOC 実績値	<p>コメント行、空行を除いた SLOC 値。 すなわち、SLOC 値(5004_SLOC 実績値_SLOC、改良開発の場合は以下定義) から、コメント行比率 (10086_SLOC 実績値_ コメント行比率)、空行比率 (10087_SLOC 実績値_ 空行比率) をもとに算出した行数を除いた値。 なお、本書で使用している SLOC、実効 SLOC 値も同意。 KSLOC は実効 SLOC 実績値を 1,000 行単位で表現したもの。 改良開発の場合の SLOC 値 開発プロジェクトの種別が b : 改修・保守または d : 拡張で、母体を含まない SLOC 値。具体的には下記の条件で算出する。 (1) 11003_SLOC 実績値 (母体) + 11004_SLOC 実績値 (追加・新規) + 11005_SLOC 実績値 (変更) + 11006_SLOC 実績値 (削除) > 0 の場合は、 SLOC 規模_ 改良開発 = 11004_SLOC 実績値 (追加・新規) + 11003_SLOC 実績値 (変更) + 11004_SLOC 実績値 (削除) (2) 11003_SLOC 実績値 (母体)、11004_SLOC 実績値 (追加・新規)、11005_SLOC 実績値 (変更)、11006_SLOC 実績値 (削除) のデータが 1 つも無かつ、11017_SLOC 母体包含有無 = 1 の場合は、 SLOC 規模_ 改良開発 = 5004_SLOC 実績値_SLOC (注意) 11003 ~ 11006 の詳細値が無く、11017_SLOC 母体包含有無 = 0 または 2 で母体含む可能性がある場合は算出の対象とならない。</p>
	データファンクション	IFPUG 手法で計測された 5057_ILF 実績値_FP + 5065_EIF 実績値_FP の値
	トランザクションファンクション	IFPUG 手法で計測された 5053_EI 実績値_FP + 5041_EO 実績値_FP + 5049_EQ 実績値の値
工期	実績月数_プロジェクト全体	5167_ プロジェクト全体工期 (実績) のデータ。 ただし、5167_ プロジェクト全体工期 (実績) が無い場合は、10128_ 月数 (実績) _ プロジェクト全体 (各社提出値) のデータを使用。
	実績月数(開発 5 工程)	開発 5 工程の開始日と終了日の間の日数を 30 日を一月として月数に換算した値。すなわち、5165_ 終了日 (実績) 総合テスト (ベンダ確認) と 5152_ 開始日 (実績) 基本設計から計算した月数。
工数	実績工数 (開発 5 工程)	基本設計 ~ 総合テスト (ベンダ確認) の各工程、ならびに工程配分不可の工数を合計した値 (単位は人時)。表下 (p.282) の 1 を参照。 開発 5 工程が全て実施されたプロジェクトのみを対象に算出。 なお、工数には社員工数 (開発工数、管理工数、その他工数、作業配分不可工数) と外部委託工数を含む。
	実績工数 (プロジェクト全体)	システム化計画 ~ 総合テスト (ユーザ確認) の各工程、ならびに工程配分不可の工数を合計した値 (単位は人時)。 なお、工数には社員工数 (開発工数、管理工数、その他工数、作業配分不可工数) と外部委託工数を含む。
	外部委託比率	外部委託工数比率 (次項を参照) のデータ。ただし、外部委託工数比率が算出できない場合は、5204_ 外注実績 (金額比率) のデータを使用。
	外部委託工数比率	基本設計 ~ 総合テスト (ベンダ確認) の各工程、及び工程配分不可の外部委託工数の合計値を、実績工数 (開発 5 工程) で割った値。 外部委託工数 ÷ 実績工数 (開発 5 工程) で算出。 なお、外部委託工数を明示的に "0" で回答しているものは "0%" とする。
	基本設計工数率	実績工数 (開発 5 工程) に対して、基本設計工数が占める割合。 基本設計工数 ÷ 実績工数 (開発 5 工程) で算出。
生産性	FP 生産性	人時あたりの FP 数。 5001_FP 実績値_ 調整前 ÷ 実績工数 (開発 5 工程) で算出。
	SLOC 生産性	人時あたりの SLOC 数。 実効 SLOC 実績値 ÷ 実績工数 (開発 5 工程) で算出。
信頼性	発生不具合数	次項に示す発生不具合数 (原因数) を使用。ただし、発生不具合数 (原因数) が無い場合は、発生不具合数 (現象数) を使用。

分類	名称	定義
	発生不具合数(原因数)	稼働後の発生不具合原因数。 以下のデータで回答があるもののうち、期間が最長のものを使用。 ・ 10112_発生不具合原因数(合計)_1ヶ月 ・ 10113_発生不具合原因数(合計)_3ヶ月 ・ 10114_発生不具合原因数(合計)_6ヶ月
	発生不具合数(現象数)	稼働後の発生不具合現象数。 以下のデータで回答があるもののうち、期間が最長のものを使用。 ・ 5267_発生不具合現象数(合計)_1ヶ月 ・ 5268_発生不具合現象数(合計)_3ヶ月 ・ 5269_発生不具合現象数(合計)_6ヶ月
	FP 発生不具合密度	FP あたりの発生不具合数。 発生不具合数 ÷ 5001_FP 実績値_調整前で算出。
	SLOC 発生不具合密度	KSLOC あたりの発生不具合数。 発生不具合数 ÷ 実効 SLOC 実績値 × 1,000 で算出。
体制	月あたりの要員数	実績工数(開発5工程) ÷ 実績月数(開発5工程) ÷ 人時換算係数で算出。人時換算係数は、901_工数の単位が「b:人月」ならば 902_人時換算係数_人時/人月を使用。「a:人時」ならば、人月人時換算係数として 160 を使用して算出する。
層別の 主要な カテゴリ	開発対象プラットフォームのグループ	309_開発対象プラットフォーム_1/2/3 の内容で、Windows 系と Unix 系のいずれかに分類。 「Windows 系」は、次のいずれかのデータのもの。 a: Windows95/98/Me 系, b: WindowsNT/2000/XP 系, c: Windows Server 2003 「Unix 系」は、次のいずれかのデータのもの。 d: HP UX, e: HI UX, f: AIX, g: Solaris, h: Redhat Linux, i: SUSE Linux, j: Miracle Linux, k: Turbo Linux, l: その他 Linux, m: Linux, n: その他 UNIX 系 「その他」は、選択肢が a ~ n ではないもの。
	主開発言語グループ (312_ 主開発言語_1/2/3 が指定した言語の種類の内いずれかであるもの)	312_ 主開発言語_1/2/3 の内容が、指定した言語の種類の内いずれかと一致するものをグループとしたデータセット。本書の場合は、4 種類 (b: COBOL、g: C、h: VB、q: Java) を指定した。 指定した種類の言語を、312_ 主開発言語_1/2/3 の 1 2 3 の順に調べて、指定言語の内いずれかに合致すれば対象となる。例えば 312_主開発言語_1 で合致した場合は、312_ 主開発言語_2/3 については調べない。 (例1) 次のような場合は、"q: Java" でグループになる。 312_ 主開発言語_1 が "a: アセンブラ" 312_ 主開発言語_2 が "c: PL/I" 312_ 主開発言語_3 が "q: Java" (例2) 次のような場合は、"h: VB" でグループになる。 312_ 主開発言語_2 以降は検査しない。 312_ 主開発言語_1 が "h: VB" 312_ 主開発言語_2 が "g: C" 312_ 主開発言語_3 が "a: アセンブラ" (例3) 次のような場合は、グループに含めない。 312_ 主開発言語_1 が "c: PL/I" 312_ 主開発言語_2 が "m: PL/SQL" 312_ 主開発言語_3 が "a: アセンブラ"
FP 種別 カテゴリ	FP 計測手法混在	ファンクションポイント (FP) の計測手法において、IFPUG 法、SPR 法、NESMA 概算法、その他手法 (企業独自の手法) を区別していない場合、FP 計測手法混在と表記する。
	IFPUG グループ	FP 実績値の算出に使用した計測手法のうち、a: IFPUG, b: SPR, c: NESMA 概算 を総称として表現したもの。

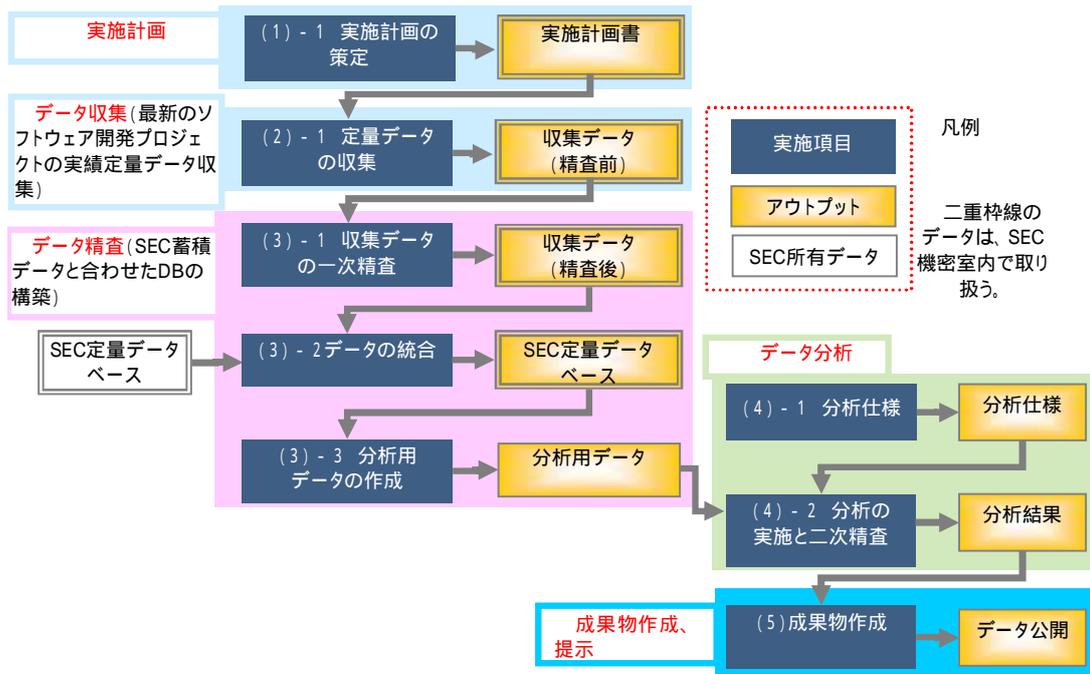
1 実績工数（開発 5 工程）の図解

基本設計～総合テスト（ベンダ確認）の 5 工程が全て実施されたプロジェクトに対して、下表の薄黄色セルの工数を合算し、さらに人時へ換算した値を範囲とする。

工数内訳		開発 5 工程							総合テスト (ユーザ確認)	工程配分 不可
		システム 化計画	要件 定義	基本設計	詳細設 計	製作	結合 テスト	総合テスト (ベンダ確 認)		
社内 実績 工数	開発									
	管理									
	その他									
	作業配 分不可									
外部 委託 工数	開発工 数									

薄い黄色は合算する対象の工数を示している。

2.2.2 ベンチマーキングの実施方法



手順	input	プロセス	output
1．実施計画	<ul style="list-style-type: none"> • 新年度方針 	<ul style="list-style-type: none"> • 新方針に沿って、収集データ、成果物の提示方法等の実施計画をたてる。データの収集は継続性を考慮しているため、収集対象、収集項目、データの取り扱い、成果物の表示方法は変更のあったもののみ検討する。 • 分析の精度向上等、特に重点的に収集するプロジェクトデータとして収集項目を検討する。 	<ul style="list-style-type: none"> • データ提供者対象 • データ項目 • 収集方法 • リポジトリ管理方法 • 分析方法 • 提示方法などを含む実施計画書
2．データ収集	<ul style="list-style-type: none"> • 収集対象 • データ項目 • 収集方法 	<ul style="list-style-type: none"> • 収集方法に従い、収集対象へデータ項目の収集を行う。 • 過去提供データも変更が在れば収集の対象にする 	<ul style="list-style-type: none"> • 収集データ
3．データ精査	<ul style="list-style-type: none"> • 提供データ • リポジトリ管理方法 	<ul style="list-style-type: none"> • 管理方法に従い、提供データの矛盾チェックや極端な値のデータ精査を実施する • 精査済みのデータをリポジトリ（SEC 定量データベース）に追加する • 分析のため匿名性を確保するための処理を行い、分析用データを作る。 	<ul style="list-style-type: none"> • 精査済みのプロジェクトデータ（リポジトリ）分析用データ
4．データ分析	<ul style="list-style-type: none"> • 分析方法 • 匿名化された分析用データ 	<ul style="list-style-type: none"> • 分析方法に従い各種分析を行う • 分析の結果を踏まえ、二次精査する • 分析結果は識者を集めた部会にてレビューする。 	<ul style="list-style-type: none"> • 統計値、式、表、図（公開用元データ）
5．成果物作成、提示	<ul style="list-style-type: none"> • 統計値、式、表、図（公開用元データ） • 提示方法 	<ul style="list-style-type: none"> • 提示方法に従い、成果物を作成し、提示する 	<ul style="list-style-type: none"> • 成果物、公開データ

2, 3, 4は機密データを扱っているため、収集中はデータを暗号化し、データの処理は機密が保障できる環境で行う。

2.2.3 データ収集、蓄積及び交換の方針

(a) データ収集対象

データ提供者は提供を申し出た組織を対象としている。提供にあたり、組織によっては秘密保持契約が必要なところとは契約を結び、収集を行っている。

基本的収集対象プロジェクト内容は以下に示す。なお、年度により、強化したい分析項目に沿って収集強化項目を定めている場合もある。

< 収集対象プロジェクト内容 >

1．重点収集項目について、欠損が極力ない状態のもの

- 工期、工数、規模、要員数、フェーズ(工程)の有無、稼働後不具合、開発言語（複数の場合は大体の比率も）、アーキテクチャ、OS、外部委託比率、業種、等

- ・ [必須] と [条件付必須] が極力埋まるもの

2. 工程

- ・ 基本設計～総合テスト(ベンダ確認)をカバーしているプロジェクトを優先
- ・ 工程ごとの区切りの情報(工期、工数)をできるだけ記入できているものを優先
- ・ 分割発注/受注などで工程の途中で案件が分割されているプロジェクトデータは、1本に纏めて工程の抜けがない状態にしてある

3. 新たな収集件数は、各組織 10 件～20 件とする

4. プロジェクト時期は、近年 3 年間終了のもの

< 基本的重点収集項目 >

- ・ 開発種別： 新規開発、改修・保守、拡張、再開発
改良開発(改修・保守、拡張)は、昨年度に続き、件数の充実を図る。
- ・ アーキテクチャ： イン트라ネット/インターネット、2 階層 C/S、3 階層 C/S
- ・ 業種： 製造業、情報通信業、卸売・小売業、金融・保険業、公務 など
- ・ 業務： (該当するものを記入下さい)
- ・ 開発言語： Java、VB、C、COBOL、C++、Visual C++、C++、C#、など
- ・ プラットフォーム： Windows 系、Unix 系、Linux 系
- ・ 規模の指標： FP、SLOC のいずれかで計測されているもの
FP 規模の場合： 計測手法名が明確
SLOC 規模の場合： 言語名が明確
改修・保守，拡張，再開発の場合は、FP/SLOC 母体包含識別フラグを必須入力。さらに、母体規模を極力記入。
- ・ 稼働後の不具合数が記録されているもの
- ・ 工期、工数、規模の実績データが把握できているもの
プロジェクトの全体
工程別の工期、工数
- ・ 工程
基本設計、詳細設計、製造、結合テスト、総合テスト(ベンダ確認)の開発 5 工程を含む
基本設計、詳細設計、製造、結合テスト、総合テスト(ベンダ確認)のうちの 3 つ以上を含む
- ・ 開発プロジェクトの概要： 必須
- ・ 業務の種類： 必須
- ・ システムの用途： 必須

(b) データ収集方法

データの収集は毎年、対象組織に以下のように依頼している。

- ・ 前年度までに収集した対象組織のデータを SEC 蓄積データより抽出、収集ツールと共に各組織に送付して、収集を依頼する。
- ・ 収集期間は年度により若干異なるが、概ね 8～10 月に行っている。

- ・ツールには精査機能がついており、提供段階である程度精査されたデータとなっている。

収集ツールのフォーマットは付録に示す。

2.2.4 ベンチマーキングデータの要件

分析結果は有識者を交えて、レビューを行い、分析方法・内容に誤りがないか、誤った解釈をされないかなど精査して、公表データとする。

個別プロジェクトのプロファイルの内、プロジェクトの判別ができる項目は固有名詞を一般名などに変更する。

分析対象の標本が 10 件以上であること。

分析に用いるデータに偏りがないよう、分析対象の標本が 3 組織以上かつ 1 組織が占める割合が 70%以下であること。

統計量の代表値は分布の非対称性が大きいと見られた場合、平均値より中央値を採用している。

| 相関係数 | 0.85 のとき強い関係、 $0.85 >$ | 相関係数 | 0.70 のときやや強い関係として表している。

2.2.5 結果の報告方法

毎年「ソフトウェア開発データ白書」にて公表

「定量データに基づくプロジェクト診断支援ツール」に白書の結果を反映

2.3 他活動・標準との関係

ISBSG を参考に必要と思われる項目を追加、データ収集が可能なように実情に合った定義に編集し策定した。

3. 財団法人 経済調査会：ソフトウェア開発に関する調査票（受託者向け）集計結果その2

3.1 公開データ提供の背景と目的

3.1.1 背景

財団法人経済調査会は、昭和21年9月の設立以来、主に公共事業、そのなかでも建設工事の費用算定の根拠となるコスト情報を提供してきた。具体的には、建設工事のなかの土木工事の場合、工事費用は直接工事費（材料費、労務費、機械経費等）と一般管理費などの諸経費で構成されるが、それらの構成要素及び歩掛りといわれる施工の生産性について調査を行い、また、刊行物（弊会発行月刊「積算資料」など）、講習会などで情報提供を行ってきた。

昭和の終わりころから、コンピュータを活用した業務の割合が増加するのに伴い、ソフトウェアの開発費がクローズアップされてきた。ただ、ソフトウェアの分野については、建設工事と比べ費用算定の考え方が未成熟であり、そのため、多くの公共機関、民間企業の情報システム発注に関わる担当者からソフトウェア開発費の根拠情報及び妥当性の高い費用算定の考え方などを求める声が多かった。

当時、そのような弊会刊行物ユーザの声を受けて、また、ソフトウェア開発費に関する公表された情報が見当たらなかったこともあり、ソフトウェアの分野について調査を開始した。その結果、平成2年に弊会が発刊した「ソフト・サービス料金」（現在休刊中）に公表できたのは、システムエンジニアなど技術者料金を中心とする一部の情報のみであったが、ある程度ソフトウェア開発費用算定の参考情報を提供できたものと考えている。

しかし、ソフトウェア開発費全体について妥当性の高い費用算定の根拠を求める声に対しては、不十分であったため、平成9年から本格的にソフトウェア開発費算定の考え方を確立すべく活動を開始した。

まず、平成10年、外部の専門家に多数参画いただき、弊会内に「ソフトウェア開発費積算研究委員会」を設置し、同時に、ソフトウェア開発に関する調査もスタートさせた。当初、同委員会は約2ヶ月に1回の頻度で開催し、対象範囲、開発費算定方法、開発工程区分、開発技術者区分などベースとなる部分について多くの議論を行った。

その成果として、平成13年度には費用算定の基本的な考え方をまとめ公表した。調査はほぼ毎年実施し、平成15年度から日本ファンクションポイントユーザ会FP法利用検討委員会（JFPUG / FPSMSG）、平成18年度からは奈良先端科学技術大学院大学情報科学研究科にも参画いただき、現在は3者の共同調査として実施している。

ソフトウェア開発の調査開始当初はソフトウェア開発費のコスト中心で考えていたが、QCD（品質・コスト・工期）すべてを考慮する必要があるという観点から、品質に関する項目を追加などの結果、現在の調査項目は多岐にわたっている。調査にあたっては、継続性を重視し、基本項目は原則として変化させない、時代に即応した設問項目を取り入れるというスタンスで実施してきたものであるが、調査を重ねるごとに少しずつ調査項目数が増えてきたことは否めない。

また、平成18年度以降はソフトウェア開発の他、システム運用、ソフトウェア保守の分野についても調査を実施し、調査範囲を拡大させている。

これらの調査データは、単年度ごとの集計・分析結果として冊子にとりまとめ調査協力者に配布すると同時に、弊社発行月刊「積算資料」などに公表してきた。

今後は、蓄積データの集計・分析結果を公表、Webサイトにベンチマークのツールとして提供していく予定である。

3.1.2 目的

コンピュータシステムの開発環境が急速に変化している現在では、ソフトウェア開発において、多くの場合、受発注者ともに開発費の見積りが困難な状況になっている。

そのため、弊社ではかねてより、ソフトウェア規模の計測手法として、ファンクションポイント法に注目してきた。適用にあたっての課題は多いものの、定量的にソフトウェアの規模を計測することができる、比較的開発の早い段階に規模を把握できるといった利点をみると、有効な手段と考えられるからである。

また、弊社では、ソフトウェア開発においては標準化が特に重要であると考えている。平成10年から本格的にソフトウェア開発の費用積算について研究を開始すると、まず、最初に着手した研究テーマは、ソフトウェア開発工程の標準化であり、開発技術者区分の標準化であった。プロジェクトを定量的に管理するためには同じ条件の開発環境のデータを収集し、その統計値と比較することが必須だからである。社内標準によりデータをそろえることはもちろんであるが、できれば業界標準にそろえることが望ましいと考えている。お手本として、建設業界では長い歴史を重ねて施工体系などの標準化が進み、共通の積算基準が整備されていった事例がある。

弊社がソフトウェア開発の調査・研究を開始した目的は、妥当性、客観性の高いソフトウェア開発費算定のための標準的な数値を提供することであるが、そのためには同じ条件の開発環境のデータを収集する必要があると考えている。そのため、直接的なコスト要素はもちろん品質、工期といった要素もコストに影響するため、幅広く情報を収集し、データ分析を行ってきた。

現在の収集データの分析結果は、ゆるい条件でそろえたものであることから、当面はプロジェクトのさまざまな目安となる情報を提供することを目的としている。例えば、ソフトウェア開発プロジェクトの発注者にとっては、ソフトウェア開発工数や費用を見積する目安となる情報であり、受注者にとっては、自社のプロジェクトデータを評価する目安となる情報を提供することを目的としている。

3.1.3 利用対象者

想定している成果物の主な利用対象者(メトリクスユーザ)とは、ソフトウェア開発プロジェクトの発注者・受注者それぞれのプロジェクトマネージャなどである。

3.2 ベンチマーキングプロセスの概要

3.2.1 ベンチマーキングの基本概念（方針）と定義

（a）基本概念（方針）

基本分析（回帰分析）の方針

- ・ 2項目間の相関をみる場合は、分析対象データ全体をカバーする散布図（全体散布図）を表示する。データ値域が大きい場合は、対数軸とする。
 - ・ 散布図を表示したものについては回帰分析を行い、基本的に回帰式及び回帰曲線を表示する。
- 層別分析の方針

- ・ 層別化をおこなう場合は、基本的に層別された分析対象データ全体をカバーする散布図（層別全体散布図）を表示する。データ値域が大きい場合は、対数軸とする。
- ・ 分析の種類によっては、層間の違いを明確にするために、層ごとの箱ひげ図を表示し、統計的有意差を確認する。

（b）用語の定義

名称	定義
開発工程	下記に弊会で定義している開発工程と共通フレーム 2007 の開発プロセスの比較表を示す。
開発 6 工程	弊会で定義している開発工程の内、2. 基本設計(A) ~ 7. 総合テスト(ベンダ確認)の範囲を指す。
FP 規模	ファンクションポイントの単位で表す規模数
FP 見積	FP 規模数の見積値
FP 実績	FP 規模数の実績値
FP 生産性	人月あたりの FP 規模数
標準 FP 生産性	弊会のメトリクス測定式を利用して求めた FP 生産性
工期(月数)	ソフトウェア開発の期間(月数)
見積工期(月数)	工期(月数)の見積値
実績工期(月数)	工期(月数)の実績値
標準工期(工数と工期)	弊会のメトリクス測定式を利用して求めた工期(月数)
見積・実績差の比率	弊会のメトリクス測定式を利用して求めた見積・実績差の比率
工数(人月)	ソフトウェア開発の工数(人月)
見積工数(人月)	工数(人月)の見積値
実績工数(人月)	工数(人月)の実績値
月あたりの開発工数	工数(人月)を工期(月数)で除算した値
工程別工数(人月)	弊会で定義している開発工程別の工数(人月)
製造工程の工数(人月)	弊会で定義している開発工程別の プログラム設計・製造における工数(人月)

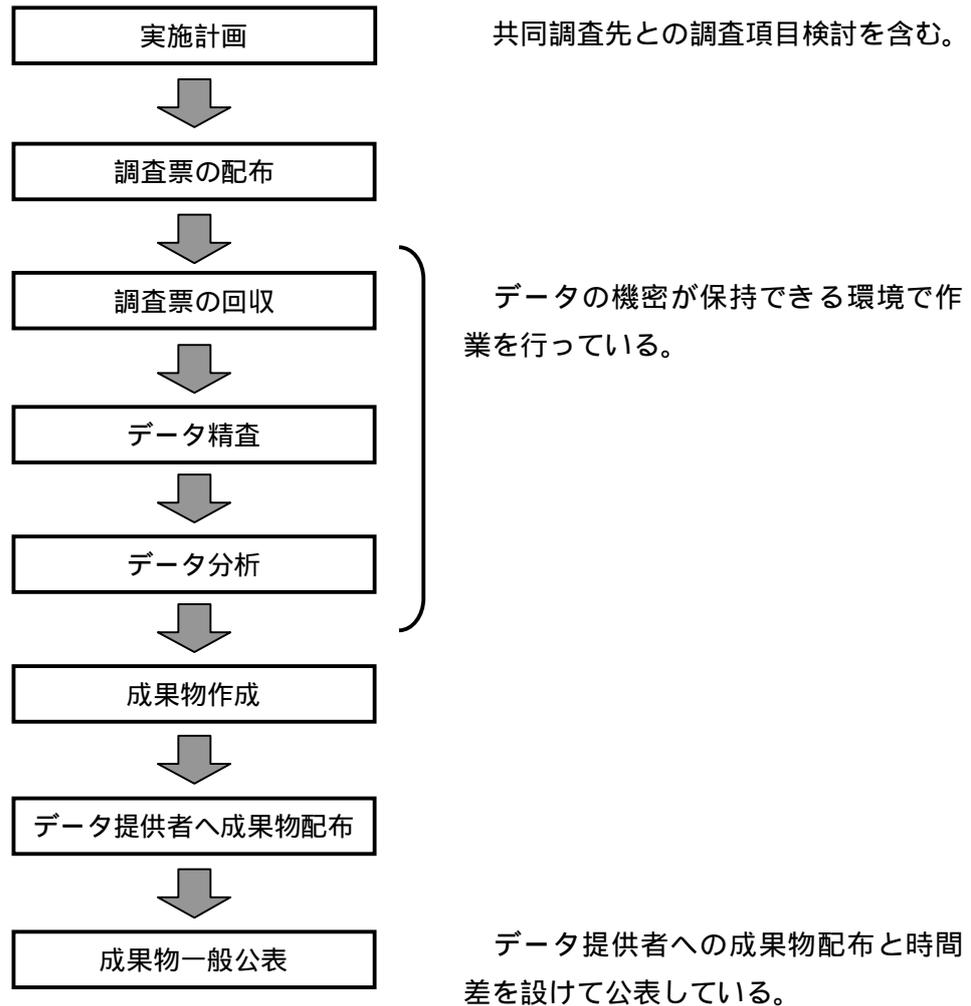
名称		定義
設計工程の工数(人月)		弊会で定義している開発工程別の 基本設計(A)と 基本設計(B)における工数(人月)
工程別工数比率		弊会のメトリクス測定式を利用して求めた工程別工数比率
発生バグ件数		システム稼動後3ヶ月間のバグ件数
発生バグ密度		弊会のメトリクス測定式を利用して求めた1000FP当りの発生バグ件数
生産性 変動要因	機能性	機能性とは、下記の要件を指す。 <ul style="list-style-type: none"> ・ 難易度の高い機能(合目的性) ・ 精密性の高い計算(正確性) ・ 他システムとの接続(相互運用性) ・ 税法、OSI(Open Systems Interconnection)規格等の公的規則や公的標準、社内規則や社内標準等(標準適合性) ・ 機密保護やアクセス管理(セキュリティ)
	信頼性	信頼性とは、下記の要件を指す。 <ul style="list-style-type: none"> ・ 故障発生率(成熟性) ・ システムダウン(障害許容性) ・ システムダウンからの再開・回復時間(回復性)
	プラットフォーム	プラットフォームとは、ハードウェアとソフトウェア(OSやDBMSなど)の複合体であり、その適合性(ニーズ、性能など)を指す。
	開発スケジュール要求	開発スケジュール要求とは、開発スケジュールの制約度合を指す。
	発注要件の明確度と安定度	発注要件の明確度とは、発注時点における発注仕様書の明確度合を指す。 発注要件の安定度とは、ソフトウェア開発期間中に発注者の要因により仕様変更が発生する度合を指す。
	ユーザの参画割合	ユーザの参画割合とは、発注者の関与度合を指す。

開発工程

弊会のソフトウェア開発工程	共通フレーム 2007 の開発プロセスのアクティビティ
1. 開発プロセス開始の準備	1.6.1 プロセス開始の準備
2. 基本設計(A)	1.6.2 システム要件定義
	1.6.4 ソフトウェア要件定義
3. 基本設計(B)	1.6.3 システム方式設計
	1.6.5 ソフトウェア方式設計
4. 詳細設計	1.6.6 ソフトウェア詳細設計
5. プログラム設計・製造	1.6.7 ソフトウェアコード作成及びテスト
6. 結合テスト	1.6.8 ソフトウェア結合
	1.6.10 システム結合
7. 総合テスト(ベンダ確認)	1.6.9 ソフトウェア適格性確認テスト
	1.6.11 システム適格性確認テスト
8. 総合テスト(ユーザ確認)	1.6.12 ソフトウェア導入
	1.6.13 ソフトウェア受入れ支援

3.2.2 ベンチマーキングの実施方法

手順	input	プロセス	output
1. 実施計画	<ul style="list-style-type: none"> • 新年度方針 	<ul style="list-style-type: none"> • 方針に沿って、調査票の調査項目(前年度と比較して追加・削除など)、データの取扱い、成果物の提示方法等の実施計画をたてる 	<ul style="list-style-type: none"> • 調査対象先 • 調査方法 • 調査票 • 調査項目などを含む実施計画書
2. 調査票の配布と回収	<ul style="list-style-type: none"> • 調査票 • 配布・回収方法 	<ul style="list-style-type: none"> • 調査対象先への調査票の配布・回収方法に従い、収集データを収集する 	<ul style="list-style-type: none"> • 収集データ
3. データ精査	<ul style="list-style-type: none"> • 提供データ • データ精査方法 	<ul style="list-style-type: none"> • 匿名性を確保するためデータの加工を行う • 調査項目間のデータの整合性を確認し、収集データを精査する • 必要に応じて追跡調査を行う 	<ul style="list-style-type: none"> • 匿名化、精査済みのプロジェクトデータ(リポジトリ)
4. データ分析	<ul style="list-style-type: none"> • 匿名化、精査済みのプロジェクトデータ(リポジトリ) 	<ul style="list-style-type: none"> • 既存の成果物で採用しているデータ分析方法に従い各種分析を行う 	<ul style="list-style-type: none"> • 統計値、式、表、図(公開データ)
5. 成果物作成、提示	<ul style="list-style-type: none"> • 統計値、式、表、図(公開データ) 	<ul style="list-style-type: none"> • 既存の成果物の様式に従い、成果物を作成し、提示する 	<ul style="list-style-type: none"> • 成果物、公開データ



3.2.3 データ収集、蓄積及び交換の方針

(a) データ収集対象

日本ファンクションポイントユーザ会（JFPUG）に登録されている企業を中心とした全国の情報処理サービス企業など。

平成 20 年度の調査では 589 社に配布し、88 社 / 119 プロジェクトの回答を得た。（うち JFPUG 会員 60 社 / 85 プロジェクト）

(b) データ収集方法

データ収集方法

紙ベースの調査票による調査を原則としている。

データ収集期間

基本的には、11 月～翌年 2 月に掛けて実施している。

データ収集頻度

基本的には、年に 1 回の頻度で実施している。

3.2.4 ベンチマーキングデータの要件

(a) 収集データの精査方法

匿名化

収集したデータから企業を特定する調査項目が含まれている調査票 を分離している。

データ品質の確保

データの精査において、各調査項目に異常値と考えられるような極端なプロジェクトデータについては、データ提供者に確認を行い必要に応じて補正している。

データセットの偏り防止

ソフトウェア開発に関する調査（受託者向け）については、特定企業のデータによる影響を抑えるため、中小企業から大企業まで幅広い層を調査対象としている。（平成20年度のソフトウェア開発の調査回収結果では、ソフトウェア開発技術者数100人未満の企業の回答比率は27.1%、100人以上300人未満の企業の同率は31.8%、300人以上1,000人未満の企業の同率は23.5%、1,000人以上の企業の同率は17.6%）

(b) リポジトリ（参照用のプロジェクトデータの格納場所）の保管、管理方法

データベースとして保管している。

単年度のデータと複数年度のデータを管理している。

3.2.5 結果の報告方法

成果物は、ダイジェスト版として報告書を作成し、データ提供者に配布している。

3.3 他活動・標準との関係

ソフトウェア開発に関する調査（受託者向け）については、日本ファンクションポイントユーザ会 FP 法利用検討会（JFPUG / FPSMSG）及び国立大学法人奈良先端科学技術大学院大学との共同調査である。

調査結果の分析ケースについては、IPA/SEC のソフトウェア開発データ白書を参考としている。

今回公開データの対象としたソフトウェア開発に関する調査（受託者向け）の集計結果の他に、委託者側に調査を実施したソフトウェア開発に関する調査（委託者向け）の集計結果及びソフトウェア保守に関する調査の集計結果（受託者向け・委託者向け）等も公表している。

2010年6月ころには調査協力先に2001年以降のソフトウェア開発に関する調査の蓄積データの分析結果をまとめ（A4 約200頁）配布する予定である。

4. 社団法人 情報サービス産業協会：定量的マネジメントに係る調査

4.1 活動概要

JISA「情報システム信頼性向上委員会」では、平成19年度に業界に対して具体的な信頼性向上活動の提示・普及させること、特に業界における信頼性向上の取組みの見える化、測れる化としての定量的なマネジメントとしてライフサイクルプロセスにおける管理指標とその活用方法を提案することを目的に、「信頼性向上のベストプラクティスを実現する管理指標調査」(以下、“管理指標調査報告書”)を実施した。本調査は、会員企業を中心に開発、保守、運用の各々について、現場で実際に使われている信頼性の管理指標を集約・提示するとともに、その効果、活用方法等をプラクティスとして示し、かつ、経済産業省の信頼性ガイドラインへの対応関係を提示した。これより、信頼性向上に向けた具体的な活動の示唆を与えている。

一方、信頼性向上には、供給者であるベンダ企業での指標の活用だけでなく、発注者であるユーザ企業においてもシステム開発の状況を把握し、コントロールすることが望まれることから、平成20年度には「情報システム信頼性向上のための管理指標活用の普及拡大調査」を実施し、管理指標調査報告書で示された管理指標を中心に、委員会のメンバ企業とユーザ企業に対するヒアリング調査や、当該企業が一堂に会しての意見交換会を行い、ユーザ企業とベンダ企業の双方で共有することによりシステムの信頼性向上に高い効果をもたらすと考えられる指標を取り纏め、発注者と供給者間で活用できる管理指標リスト(以下、共有管理指標)を発表した。

次章以降では、平成19年度に実施した「信頼性向上のベストプラクティスを実現する管理指標調査」、及び平成20年度に実施した「情報システム信頼性向上のための管理指標活用の普及拡大調査」について概要を述べる。

4.2 信頼性向上のベストプラクティスを実現する管理指標調査

4.2.1 信頼性向上のための管理指標活用の背景

情報システムのライフサイクル(開発、保守、運用)での管理指標活用の背景には、情報システムの開発や運用を取り巻く外部環境の変化(システム技術の転換)やプロジェクトの特性の変化(短期開発、高品質要求)の下での情報システムの品質確保に対する問題がある。この問題に対応するには、まずシステムそのものの品質や開発・保守・運用のプロセスがどのような状況であるのかを把握する必要がある。この行為は基本的に事後対策となる。続いて、問題が生じる前に対策を打つことを目的に品質や状況の傾向分析などの事前の予測や対策が行われるようになり、この両フェーズで管理指標が利用される。システムやプロセスの状況を測定(見える化、測れる化)し、その数値を定量的にマネジメントすることで、状況の良し悪しの判断、原因究明、状況の予測、コントロールなどが可能

となり、問題解決、問題の未然防止及びユーザを含む関係者が客観的に議論することが出来るようになる。

4.2.2 管理指標の導入・活用へのアプローチ

情報システムの品質確保に対する問題には、様々な課題や課題に対する目標（ゴール）がある。管理指標は、それら課題の対処方法を検討するにあたり、状況を把握しコントロールするためには何を測定しなければならないかを検討した上で設定される。さらに、実際に当該指標で現状を把握し、目標との乖離を把握し、コントロールする活動（PDCA サイクル）へと繋がられる。

管理指標の導入、活用ステップにおいては、まずは個々のプロジェクトで管理指標のデータを取得することで現象（不具合等）を把握し、要因を分析し不具合の発生防止へ向けた対策を講じる。組織全体でデータが蓄積された後には、当該データを元に組織的な計画値を定め、プロジェクト業務での予実のコントロール（リアルタイムコントロール）を行い、組織的な改善活動へ繋げていくことが重要である。このため、管理指標活用の PDCA サイクルは、個々のプロジェクトでのサイクルと、組織的（全社的）活動でのサイクルの 2 通りがある。

プロジェクトでの PDCA サイクル

プロジェクトを担当するプロジェクトマネージャや PMO が主体となり目標（基準）値を定め（P）、プロジェクト遂行で各種指標を測定（D）し、プロジェクト特性にあった目標値に対する乖離を把握（C）し、原因を探り、コントロールを随時実施する（A）。

組織活動での PDCA サイクル

組織の品質保証部門や全社的な管理を行う PMO が主体となり、組織目標を定め（P）組織で実施しているプロジェクトでの共通現象（主に問題）を把握（D）し、その組織的現象の状況分析・要因解析を行い（C）、改善へ向けた管理指標活用のさらなる対策立案、実施（A）を行う。

4.2.3 システムのライフサイクル全体にわたる指標活用と効果

開発により構築されたシステムは、運用を通じてサービスが提供されるとともに状況が監視され、不具合の発見や外部環境の変化の把握に応じた是正・改良を保守として実施し、改良されたシステムを継続して運用することが繰り返される。

開発時には運用のイメージが確実に反映されなければならない、開発時のつくりは保守における作業効率に大きな影響を及ぼし、さらにシステムの信頼性の確保・向上に大きな影響を与える。信頼性は、開発時の高信頼性の実現をベースとして、運用を通じて、高信頼

性の維持・向上が次第にはかられることとなる。

開発、運用、保守は、システムのライフサイクルを構成する主要な活動であり、各活動の管理指標は三位一体となりシステムのライフサイクルの中で信頼性向上に寄与する。例えば、運用時の管理指標によりシステムの問題点や改良点が定量的に把握されれば、重要性から優先順位をつけて保守または開発を実施するなどの活動の最適化を図ることが出来る。(図 4-1)

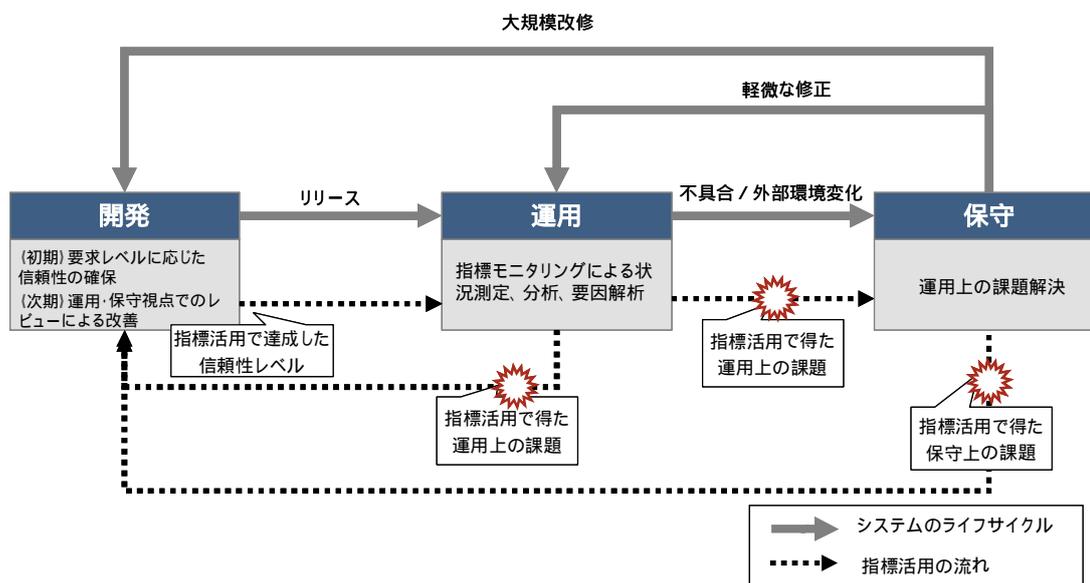


図 4-1 システムのライフサイクル全体にわたる管理指標活用

4.2.4 ITベンダ企業によるプロセス毎の管理指標活用と効果

開発・保守プロセス

開発では、プロセス、プロダクトの評価に管理指標を用いるほかに、さらに発注者、受注者双方の視点での評価を加え、全体にわたって管理指標で評価する利用方法がある。また、開発量や生産性に影響を与える品質特性、ソフト・ハードウェアの条件、変更量、受発注者間のコミュニケーションなどを指標で管理し、価格体系の中に入れて利用する方法がある。一方では、仕様変更量の管理、予実差異分析、プロジェクト評価へ管理指標を活用する方法もある。

なお、開発プロセスで共通的に利用されている管理指標には、設計時のレビュー、テスト時のバグ摘出件数、納期に係る管理指標がある。(表 4-1)

表 4-1 開発・保守プロセスで共通的に活用されている管理指標

	管理指標	定義例
レビュー	レビュー指摘率	指摘件数 / レビュー対象規模 × 100
	レビュー回数率	実施件数 / レビュー実施予定数 × 100
	レビュー速度	レビュー対象規模 / レビュー時間
	レビュー前倒指摘率	レビュー時指摘バグ数 / レビュー及びテスト時指摘バグ数合計 × 100
	レビュー効率	指摘件数 / レビューにかけた工数 × 100
テスト	テストケース密度	テストケース件数 / 実ソースコード規模
	テスト網羅性(率)	テスト実行済分岐数 / 全分岐数 × 100
	机上デバッグ抽出バグ密度	机上デバッグ抽出バグ件数 / 実ソースコード規模 × 100
	工程毎バグ抽出件数	検出バグ数
	リリース後バグ件数(密度)	出荷後一定期間以内のリリース規模に対する障害件数(の割合)
納期	納期達成率	一定期間での、納期達成件数 / 全完了件数 × 100
	納期遅延率	(契約納入日 - 納入日) / 工期日数 × 100

保守プロセスでは、開発プロセスでの管理指標に加え、問題把握及び要因分析を目的とした管理指標が活用されている。

運用プロセス

運用プロセスでは、ユーザと運用のマネジメントレベルについて正確な共通認識とコミュニケーションを行うために、ユーザとの間で締結した SLA を最終目標としてその基礎となる各種指標を月例で収集、分析した結果をまとめたレポートをユーザへフィードバックする利用方法がある。ここで扱う管理指標のレベルは、システムの重要度に応じて定めることとなる。一方、アウトソーサーのサービス提供状況を評価するために利用する方法もある。

障害発生状況、移管、稼働、性能、セキュリティに係る管理指標が共通的に活用されている傾向にある。(表 4-2)

表 4-2 運用プロセスで共通的に活用されている管理指標

	管理指標	定義例
障害発生状況	オンライン、バッチ、デリバリー別障害発生件数	件数、割合の経時(月次)変化
移管管理	作業登録件数、割合の経時変化(ジョブ登録、監視登録、プログラム登録)	同上
稼働管理	オンライン開局状況(開局達成率、特別作業状況)	同上
	オンライン利用状況(トランザクション件数、対前年比)	同上
	バッチジョブ稼働状況(稼働ジョブ数、特別作業状況)	同上
	サービスデリバリ実施状況(印刷件数・整備件数・発送件数)	同上
性能管理	オンライン稼働状況(リソース使用状況、トランザクション滞留)	同上
	バッチジョブ稼働状況(予定時刻内の完了率)	同上
セキュリティ管理	ID管理(ホスト、サーバ)	同上
	入退館管理	同上

管理指標を活用することによる効果には、障害、トラブル、人的ミスの経年的減少による「信頼性の向上」のほか、リスクや障害の早期発見、プロジェクトの状況を分析して障害発生を未然に防止することによる「プロジェクトマネジメントの高度化」、経年的な状況把握で組織の特徴を理解し改善することや、内部統制への対応が容易になるなどの「組織マネジメントの高度化」が確認されている。

4.2.5 今後の課題

今後は情報サービスに対するユーザのニーズがより一層高度化することにより、情報システムは大規模、複雑化が進み、社会活動への影響度、重要度が高くなる可能性がある。こうしたシステムを国内外の企業・組織での分業によって実現し維持していく上で、信頼性を確実に確保しかつ向上させていくためには、個々の現場での実践的な信頼性に対する取組みが不可欠である。よって、経済産業省の信頼性ガイドライン、重要インフラの情報セキュリティ対策に係る行動計画、非機能グレード検討会や JISA の管理指標調査などの信頼性向上に係る各種取組み内容は、公表するだけでなく、今後は実践に向けて、業界全体にわたり普及啓発していかなければならない。

また、システムの検収には、ユーザとベンダとの間で客観的な妥当性により、お互いが納得することが望ましい。しかし実際は曖昧な契約や仕様により、システムをどの状況で承認するかの判断が明確になっていない状況が多い。このため、曖昧な契約、仕様を払拭し、ユーザとベンダとの間で齟齬のない、円滑なコミュニケーションをはかるための仕組みづくりと定着が必要である。

JISA では、我が国におけるソフトウェア分野における信頼性向上に向けた取組み等を世界各国に紹介し、有識者と学際的な議論を構築することを目的とした、経済協力開発機構

(OECD)、経済産業省、独立行政法人経済産業研究所主催のカンファレンス¹で調査成果の講演を行った。また、業界各社における管理指標活用の導入を促進するため、シンポジウムを開催し、管理指標調査内にある各プラクティスを紹介している。さらに管理指標調査を元に、ユーザとベンダの双方が納得できる定量的な管理指標の活用へ向けた取組みを行い、情報システムの役割に応じた信頼性に係る管理指標の策定を進めている。今後は、その成果をユーザとベンダ双方に普及させ、ユーザ・ベンダが一体となった上での情報システムの信頼性・安全性向上への活動を活性化、推進していくことが重要である。

4.3 情報システム信頼性向上のための管理指標活用の普及拡大調査

4.3.1 発注者と供給者間で活用できる共有管理指標

「情報システム信頼性向上のための管理指標活用の普及拡大調査」では、次のような「共有管理指標（案）」を定めた（表 4-3）。これら指標は、『管理指標調査報告書』に示される項目に含まれているものである。実際の利用では、システムに求められる要件（信頼性要件等）に応じて、必要と考えられる管理指標を検討、選択し、活用することとなる。

¹ OECD-METI-RIETI カンファレンス「ソフトウェア分野におけるイノベーション」- 最新トレンドと産業競争力への示唆 - 「ソフトウェア産業の構造的変化がもたらす課題と新たな可能性」のセッション内（平成 20 年 10 月 6 日）

表 4-3 発注者と供給者間で活用できる共有管理指標

カテゴリ	名称	定義	管理目的
レビュー	レビュー密度 レビュー指摘密度	レビュー密度(1ページあたりの時間) レビュー指摘密度(1ページあたりの指摘件数)	(レビュー密度)レビューの妥当性確認 (レビュー指摘密度)プロダクトの品質確認
状況確認(仕様)	仕様変更量	開発量 = 当初量 - 変更棄却対象量 + 変更追加量 + 変更正味棄却量 最終実現規模 = 当初量 - 変更棄却対象量 + 変更追加量 変更棄却対象量 = 当初量 × 変更棄却対象率 変更追加量 = 当初量 × 変更追加率 変更正味棄却量 = 当初量 × 変更棄却対象率 × 完成率	仕様変更に伴う開発量の状況把握
状況確認(進捗)	WBS(進捗)	WBSの消化状況	開発の進捗状況の把握
テスト	バグ(不具合検出)密度 テストケース(試験)密度	バグ(不具合検出)密度: 規模あたりのバグ数 テストケース(試験)密度: 規模あたりのテストケース数	テストの十分性とプロダクトの品質確認
テスト	累積バグ(不具合検出)数	実施試験項目数に対する不具合検出数	プロダクトの品質予測とテストの十分性確認
テスト	バグ(不具合検出)残存数	解決すべき発生不具合の数(または、発生不具合総数に対する割合)	プロダクトの品質状況の把握
テスト	実施テストケース数	実施したテストケースの累積数(または、全テストケース数に対する実施割合)	テスト活動の進捗状況の把握
テスト	テストカバレッジ	C0 指標: テスト確認したコードの割合 C1 指標: テスト確認した分岐処理の割合	テストの十分性確認
テスト	バグ(不具合検出)密度とテストケース(試験)密度との比較	バグ(不具合検出)密度: 規模あたりのバグ数 テストケース(試験)密度: 規模あたりのテストケース数	テストの十分性とプロダクトの品質確認
テスト	結合テスト工程とシステムテスト工程のバグ(不具合検出)密度との比較	バグ(不具合検出)密度: 規模あたりのバグ数	プロダクトの品質予測

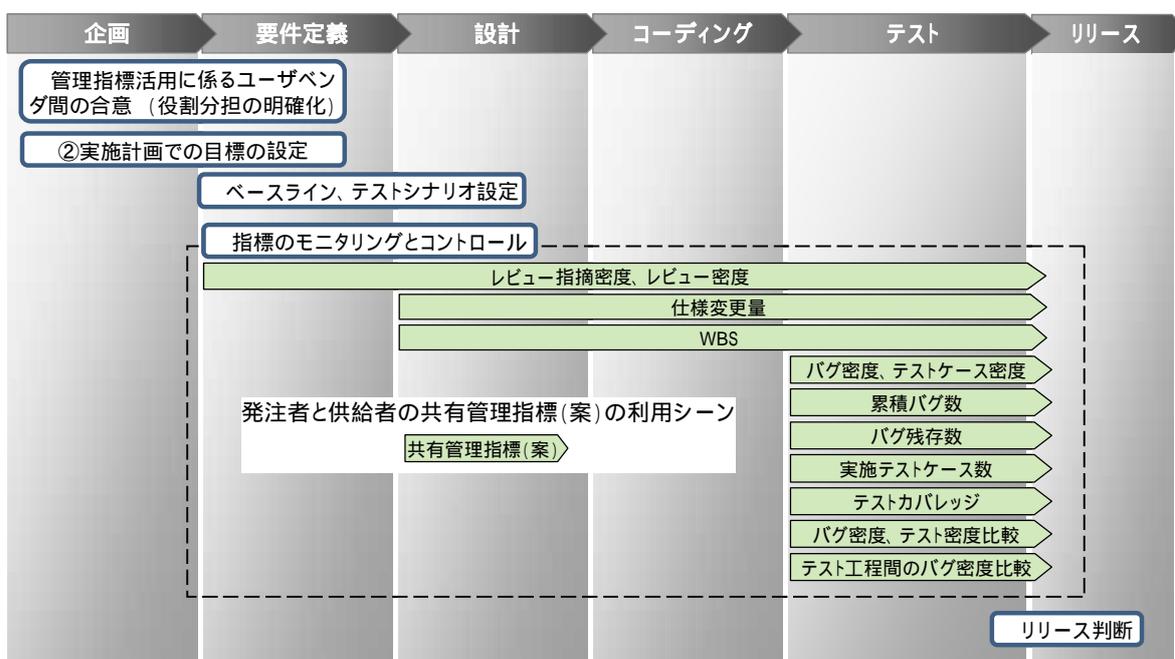


図 4-2 ユーザ・ベンダ間の共有管理指標(案)の活用ステップ

4.3.2 発注者と供給者間の共有管理指標（案）活用の前提条件

発注者と供給者の役割分担の明確化

システム開発での信頼性確保の前提として、発注者と供給者の役割分担を明確にし、かつ、両者がその役割を果たすことが必要である。

発注者は、システム開発の目的（ビジネスの目的）を明らかにし、システムに対する要件（非機能要求を含む）を決定する責任がある。また、システム開発プロジェクトを自らのプロジェクトであると意識して参画する必要がある。供給者は、システム開発の目的を理解し、確実に要件を実現する役割を果たす責任がある。また、発注者にシステム開発の過程の状況がわかりやすいように「見える化」を果たす必要がある。

管理指標による「見える化」とコントロール

開発するシステムの発注者と供給者で、システムの「見える化」のために必要な管理指標（種類、目標）を合意し共有する。これは、契約（実施計画書を含む）として明確にすることが望ましい。また、管理指標に基づいてシステムの状況を共有し、必要に応じて、供給者及び発注者のそれぞれでコントロールできることを実施する。

なお、管理指標の目標、基準値（数値）は、状況をコントロールのために活用することが基本であり、達成すること自体を目的とするものではない。目標と実態とのギャップに基づいて対策を検討することが目的である。また、最終的にはシステムの現実の実現状況を見るべきであり、指標が達成したからといって完了とすることができない場合もあることに留意する必要がある²⁾。最初に決めた目標、基準値が絶対ではなく、状況に応じて適切な見直しも十分に念頭におくべきである。一方、なし崩し的な目標、基準値の変更は最も強く戒められるものであり、指標を活用したコントロールでは厳格さと柔軟性のバランスを常に意識する必要がある。

4.3.3 活用手順

管理指標に関する合意

システムの目的や特性等に応じた管理指標を選択し、供給者と発注者で合意する。

実施計画での目標の設定

実施計画において、当該プロジェクトと背景・特性（コンテキスト）を十分に考慮して、管理指標の目標値（基本的には過去の実績に基づく）を設定する。そのため、過去の実績についても背景・特性（コンテキスト）の情報を記録しておく。

²⁾ このあたりの事情は、現状のソフトウェアエンジニアリングが他の産業（例：建設業界）のように品質確認方法が定量的な「業界標準的」手法として実現できていないことが背景にある。ただし、そのため管理指標が活用できないということにはならない。個別のケースに応じた管理指標の設定は可能である。

要件定義でのベースライン、テストシナリオの設定

変更量を管理するため、要件のベースラインを設定・共有する。供給者は場合によりテストシナリオを設定（設定は後工程の場合もある）する。

管理指標を用いた開発状況のモニタリング・コントロール

実施計画に基づき、供給者と発注者間で開発状況を共有し、目標との乖離があれば、供給者側での対策及び発注者側での対策を共同で検討する。

リリース

最終的な品質確認のための管理指標を用い、発注者側でリリースの意思決定を行う。

4.3.4 信頼性を実現するための管理指標活用の普及と今後の課題

(1) 管理指標活用の高度化

ユーザ企業とベンダ企業で管理指標の目標を設定するにあたり、目標の妥当性の判断が難しい、特にユーザ企業では報告された値の承認判断が難しいという現状がみられる。これには、ベンダ企業のみならず、ユーザ企業側においても、自ら必要な管理指標を定めて、データの蓄積・分析をするなど主体的な管理指標の活用の経験を重ねることが肝要である。

また、上流工程では「レビュー」、下流工程では「テスト」、全体を通しては「仕様の変更状況」及び「開発の進捗状況」を主要なモニタリング対象としている現状に対し「上流での定量化」や「さらに確実に精度が良い品質の予測」のニーズがある。今後、管理指標活用には、カバーする工程の拡大や、精緻さ・確実性等の高度化が求められている(表 4-4)。

表 4-4 管理指標活用の高度化として望まれる事項

項目	内容
目標値の設定	<ul style="list-style-type: none">定量的な目標の妥当性の判断
上流工程	<ul style="list-style-type: none">要件網羅性の確認、評価方法上流工程の進捗管理の見える化
レビュー	<ul style="list-style-type: none">レビュー指摘分類の整理（レビュー指摘事項の標準化等）レビューの質のばらつきを抑える方法（レビューを実施するメンバーの評価等）
完了基準	<ul style="list-style-type: none">建築業界等で実現している、業界で共有できる妥当な基準の整備前工程と後工程（特に要件定義、設計工程間）で正しく継承されているかの確認方法テストの十分性（例：テスト網羅性）を保証する仕組みの整備など、検査完了指標
メトリクスの定義等の標準化	<ul style="list-style-type: none">メトリクス利用上の前提条件の標準化（例：規模の単位（ドキュメント1ページ、コード1行といっても定義は複数）不具合の数え方等）
その他	<ul style="list-style-type: none">コスト、スケジュールについてユーザ企業・ベンダ企業双方でモニタリングすべき管理指標信頼性以外の品質特性への取組み

(2) 業界の基本リテラシとしての管理指標活用の普及

システム開発において信頼性確保・向上のために目指すべきことの一つは、ユーザ企業・ベンダ企業の双方が定量的なマネジメントを当たり前として活用すること、即ち、ソフトウェア開発を定量的に把握し、コントロールしていくことが業界の基本リテラシとなることである。基本リテラシとしての管理指標活用は、管理指標の高度化にも重要な活動基盤となる。このため JISA 信頼性部会では、ユーザ企業とベンダ企業間で実際に活用されている管理指標を提供するなど、信頼性向上を目的とした管理指標活用の普及促進を図っている。

5. 付録

5.1 独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター：ソフトウェア開発データ白書

5.1.1 工程の呼称と SLCP マッピング

次の表に、本データ項目定義で使用されているソフトウェア開発工程の名称と、SLCP JCF2007 との対応関係を示す。「工程」列には、収集したデータ項目の定義及び本白書で使用している工程名称を示している。「SLCP プロセス/ アクティビティ」と「SLCP の定義」列で SLCP との対応で工程の定義を示している。なお、ここでの工程は、おおむねベンダ側でのプロセスを示しているが、ユーザの参画が不可欠なところはユーザプロセスも多少含まれる。

工程	SLCP プロセス/アクティビティ	SLCP の定義
システム化計画	システム化計画の立案	企画者は、システム計画の基本要件の確認を行い、対象業務の確認、システムが実現している機能等の確認と整理により、システム課題を定義、業務機能をモデル化する。モデルからシステム化機能の整理、システム方式とシステム選定方針の策定、付帯機能・設備やサービスレベル及び品質の基本方針を明確化、プロジェクトの目標設定、実現可能性の検討、全体開発スケジュールの作成、費用とシステム投資効果の予測を行い、具体化したシステムに対する前提条件を整理し、システム化計画として文書化し承認する。複数プロジェクトがある場合はプロジェクト計画の作成と承認を得る。
要件定義	システム要件定義 ソフトウェア要件定義	開発者は、システム及びソフトウェアに関する要件について技術的に実現可能であるかを検証し、システム設計が可能な技術要件に変換し、システム要求仕様と確立したソフトウェア要件を文書化する。また、設定した基準を考慮して、システム要件、ソフトウェア要件を評価し文書化する。さらに、共同レビューを行う。
基本設計	システム方式設計 ソフトウェア方式設計	開発者は、ハードウェア構成品目、ソフトウェア構成品目及び手作業を明確にし、システム方式及び各品目に割り振ったシステム要件を文書化する。また、ソフトウェア品目に対する要件をソフトウェア方式に変換する。最上位レベルの構造とソフトウェアコンポーネントを明らかにし、データベースの最上位レベルでの設計、利用者文書の暫定版の作成、ソフトウェア結合のための暫定的なテスト要求事項及び予定等を明らかにする。方式設計の評価と共同レビューを実施する。
詳細設計	ソフトウェア詳細設計	開発者は、ソフトウェア品目の各ソフトウェアコンポーネントに対して詳細設計を行う。ソフトウェアコンポーネントは、コーディング、コンパイル及びテストを実施するユニットレベルに詳細化する。また、ソフトウェアインターフェイス、データベースの詳細設計、必要に応じて利用者文書を更新、ユニットテスト、結合テストのためのテスト要求事項及び予定を定義する。評価及び共同レビューを実施する。

工程	SLCP プロセス/アクティビティ	SLCP の定義
製作	ソフトウェアコード作成 及びテスト	開発者は、ソフトウェアユニット及びデータベースを開発する。また、それらのためのテスト手順及びデータを設定する。さらに、テストを実施し、要求事項を満足することを確認する。これらに基づいて、必要に応じて利用者文書等の更新を行う。また、ソフトウェアコード及びテスト結果を評価する。
結合テスト	ソフトウェア結合 システム結合	開発者は、ソフトウェアユニット及びソフトウェアコンポーネントを結合して、ソフトウェア品目にするための計画を作成し、ソフトウェア品目を完成させる。また、結合及びテストを行う。完成したソフトウェア品目と合わせてハードウェア品目、手作業や他システム等とあわせてシステムに結合、要件を満たしているかをテスト、システム適格性確認テスト実施可能状態であることを確認する。必要に応じて利用者文書等の更新を行う。テストの評価と共同レビューを実施する。
工程	SLCP プロセス/アクティビティ	SLCP の定義
総合テスト (ベンダ確認)	ソフトウェア適格性確認テスト システム適格性確認テスト	開発者は、ソフトウェア品目の適格性確認要求事項及びシステムに関して指定された適格性確認要求事項に従って、適格性確認テスト及び評価を行う。必要に応じて利用者文書等の更新を行う。また、監査の実施と支援をする。
総合テスト (ユーザ確認)	ソフトウェア導入支援 ソフトウェア受け入れ支援	開発者は、契約の中で指定された実環境にソフトウェア製品を導入するための計画を作成し、導入する。 開発者は、取得者によるソフトウェア製品の受け入れレビュー及びテストを支援する。また、契約で指定するとおりに、取得者に対し初期の継続的な教育訓練及び支援を提供する。
フォロー (運用)	運用プロセス	ソフトウェア製品の運用及び利用者に対する運用支援を行う。運用者は、このプロセスを管理するために具体化した管理プロセスに従って、運用プロセスの基盤となる環境を確立する、など。

5.1.2 データ項目定義

この節では、本白書で使用しているデータ項目の定義を示す。本書で扱ったプロジェクトデータは、この定義に従って収集し、分析を行った。

表の「データ名称」列は、データ項目の名称を表す。名称は“項番_名前”という書式となっている。「定義」列は、データ項目の定義の説明である。「回答内容、選択肢」列は、付録 B に掲載するデータ収集フォームでの回答方法（質問内容）を表しており、選択式の場合は選択肢の一覧を、自由記入の場合は（ ）と記載している。また、自由記入の場合の回答例や補足説明を記載したものもある。

(1)開発プロジェクト全般

データ名称	定義	回答内容、選択肢
10084_各社採番のプロジェクトID	各社にてプロジェクトを識別するためのID。 サブシステムの識別にも利用。 サブシステム単位でデータを捕捉できている場合に、それらを集約しないこと。	1、2、3、・・・：全体システムの場合 1-1、1-2、・・・：全体システム1のサブシステムの場合
11001_全体システム・サブシステム識別フラグ	全体システムかサブシステムかを識別するフラグ。	a：全体システム, b：サブシステム
11002_グルーピングID	グルーピングできるプロジェクト群を識別するグループIDを振る。 1 11001 の選択内容に拘らず記入する。 2 正の整数(1, 2, 3, ...)で入力する。 3 提出データセットの中にグルーピングするプロジェクトがない場合は空白。	() 例1.全体システムに”1”、サブシステム2つに”1”を入れる。 例2.サブシステム2つに”2”を入れる。 同じ数字が入っているものは分析時に集約することを検討する場合もある。
10085_本データの信頼性	当該プロジェクトデータの信頼度を右欄の4段階(A~D)で評価した値を記述する。	A：データに合理性があり、完全に整合していると認められる。 B：基本的には合理性があると認められるが、データの整合性に影響を及ぼす要因が幾つか存在する。 C：重要なデータが提出されていないため、データの整合性を評価できない。 D：データの信頼性に乏しいと判断できる要因が1つもしくは複数見受けられる。

データ名称	定義	回答内容、選択肢
103_開発プロジェクトの種別	開発プロジェクトの種別(新規か改良か)。	a: 新規開発 : ベースとなるシステムが存在せず、新規の開発を行うもの。ただし、ベースとなるシステムが存在する場合でも、新規開発部分が本プロジェクトの開発部分の約 90%以上の場合は、新規開発として扱う。 b: 改修・保守 : リリース後のシステムの運用フェーズで、ベースとなるシステムが存在し、機能追加など改修を伴う開発を行う(新規開発部分は約 10%未満である)。 c: 再開発 : 既存システムが存在し、機能仕様を殆ど変更する事無く、作り直す場合。(いわゆるリプレース) d: 拡張 : ベースとなるシステムが存在し、機能追加など改修を伴う開発を行う。(新規開発部分は約 10 ~ 90%である)
104_母体システムの安定度	103 が「改修・保守」、「拡張」、「再開発」の場合、母体システムの安定度。	a: システムは安定している, b: システムは安定化傾向にある, c: システムは不安定である, d: 母体の安定度を把握していない
105_開発プロジェクトの形態	開発プロジェクトの形態。	a: 商用パッケージ開発, b: 受託開発, c: インハウスクラス, d: 実験研究試作, e: その他(具体的名称)
106_受託開発の場合の作業場所	105 が「受託開発」の場合、その開発作業場所。	a: 顧客先, b: 自社, c: その他(具体的記述)
107_開発プロジェクトの概要	開発プロジェクトの作業概要として含んでいるものを指定する(複数選択可)。	a: ソフトウェア開発 b: インフラ構築 c: 運用構築 d: 移行 e: 保守 f: 業務支援 g: コンサルティング h: プロジェクト管理 i: 品質保証 j: 現地(本番システム)の環境構築・調整 k: 顧客教育 l: その他(具体的名称)
108_新規の顧客か否か	新規の顧客か否か。	a: 新規顧客, b: 既存顧客
109_新規の業種・業務か否か	新規の業種・業務か否か。	a: 新規業種・業務, b: 既存業種・業務
118_外部委託先情報	外部委託が有る場合に、外部委託先の情報を主要なものから 3 つまで選択する。 系列 = 資本関係有りの企業	a: 日本企業(グループ内/系列) b: 日本企業(グループ外/系列外) c: 海外企業(グループ内/系列) d: 海外企業(グループ外/系列外) e: 外部委託なし
119_外部委託先国名	118 が「c: 海外企業(グループ内/系列)」、「d: 海外企業(グループ外/系列外)」の場合に、国名を記述する(複数記入可)。 例. 中国、インド	()

データ名称	定義	回答内容、選択肢
110_新規協力会社か否か	新規の協力会社を使ったか否か。 118 が「e：外部委託なし」以外の場合に記述。	a：初回利用の協力会社, b：2回以上利用の協力会社
111_新技術を利用する開発か否か	新しい技術を利用する開発か否か。	a：新技術を利用, b：新技術を利用していない
112_開発プロジェクトチーム内での役割分担・責任所在の明確さ	開発プロジェクトチーム内での役割分担・責任所在の明確度合い。	a：非常に明確, b：概ね明確, c：やや不明確, d：不明確
113_達成目標と優先度の明確さ	納期・品質・技術開発等の達成目標と優先度の明確度合い。	a：非常に明確, b：概ね明確, c：やや不明確, d：不明確
114_作業スペース	プロジェクト遂行環境における作業スペースの状況。	a：個々人に十分広く閉じられた個人スペースあり b：個々人のスペースは普通の広さながら、集中した思考にかなり適した環境 c：やや狭くオープンスペース, 思考の集中は持続しにくい環境 d：明らかに狭くオープンスペース, 資料や計算機の設置場所もない
115_プロジェクト環境(騒音)	プロジェクト遂行環境における雑音・騒音の状況。	a：騒音は全く無く、電話による作業中断も最低限 b：騒音はほとんど気にならない。電話による作業中断は時々ある c：時としてかなりの騒音があり、電話も作業を度々中断する d：騒音がひどく、必要な集中力が維持できない。電話による作業中断も一時間毎以上の頻度である
116_プロジェクト成否に対する自己評価	当該プロジェクトのQCD観点からの成否に関する総合的な自己評価。 成功：適切な計画を立て、それを達成した場合。未計画の場合は、終了状態が良好であるといえる場合。	a：QCD 全て成功 b：QCD のうち2つは成功 c：QCD のうち1つだけ成功 d：QCD のうち成功が0
117_顧客満足度に対する主観評価	顧客が当該プロジェクトの成果に対して満足しているか否かについての回答者の主観。	a：十分に満足している b：概ね満足している c：やや不満な点がある d：不満足である
120_計画の評価(コスト)	コスト計画の妥当性を評価する。	a：コスト算定の根拠が明確で実行可能性を検討済み b：コスト算定の根拠が不明確、または実行可能性を未検討 c：計画なし
121_計画の評価(品質)	稼働後品質の目標の妥当性を評価する。	a：品質目標が明確で実行可能性を検討済み b：品質目標が不明確、または実行可能性を未検討 c：計画なし
122_計画の評価(工期)	工期計画の妥当性を評価する。	a：工期計画の根拠が明確で実行可能性を検討済み b：工期計画の根拠が不明確、または実行可能性を未検討 c：計画なし

データ名称	定義	回答内容、選択肢
123_実績の評価 (コスト)	コスト計画に対する実績の評価。	a: 計画より 10%以上少ないコストで達成 b: 計画通り(±10%未満) c: 計画の 30%以内の超過 d: 計画の 50%以内の超過 e: 計画の 50%を超える超過
124_実績の評価 (品質)	品質計画(稼働後品質の目標)に対する実績の評価。	稼働後不具合数が a: 計画値より 20%以上少ない b: 計画値以下 c: 計画値の 50%以内の超過 d: 計画値の 100%以内の超過 e: 計画値の 100%を超える超過
125_実績の評価 (工期)	工期計画に対する実績の評価。定めたまたは顧客と合意した納期に対する遅延状況で評価する。	a: 納期より前倒し b: 納期通り c: 納期を 10 日未満遅延 d: 納期を 30 日未満遅延 e: 納期を 30 日以上遅延
126_QCD の計画未達の場合の理由	コスト、品質、工期(納期)の計画が未達の場合(例えば 123 が c、d、e の場合)、その理由を主要なものから 3 つまで選択する。 (注) v1.0 の「803_予実差(遅延/前倒し)の理由」を廃止し、本項目 126 番に統合した。	a: システム化目的不適當 b: RFP 内容不適當 c: 要求仕様の決定遅れ d: 要求分析作業不十分 e: 自社内のメンバーの人選不適當 f: 発注会社選択ミス g: 構築チーム能力不足 h: テスト計画不十分 i: 受入検査不十分 j: 総合テストの不足 k: プロジェクトマネージャの管理不足 l: その他(具体的記述)
1012_総括コメント	提供データについて、分析時に考慮すべき点や IPA/SEC への連絡事項など。 例 1. 外部委託があるが、比率が分からず、記入していない。 例 2. 社内の開発工数にインフラ構築対応作業を約 3 割含む。	() 全角 256 文字まで。

(2)利用局面

データ名称	定義	回答内容、選択肢
201_業種	当該情報システムがサポートするビジネス分野。例えば顧客企業のビジネス分野。	付録 A.3 の「産業分類」の中項目の項番 01 ~ 99。
202_業務の種類	開発した情報システムの対象とする業務の種類。	a : 経営・企画, b : 会計・経理, c : 営業・販売, d : 生産・物流, e : 人事・厚生, f : 管理一般, g : 総務・一般事務, h : 研究・開発, i : 技術・制御, j : マスター管理, k : 受注・発注・在庫, l : 物流管理, m : 外部業者管理, n : 約定・受渡, o : 顧客管理, p : 商品計画(管理する対象商品別), q : 商品管理(管理する対象商品別), r : 施設・設備(店舗), s : 情報分析, t : その他(具体的名称)
203_システムの用途	開発した情報システムの用途。	a : ワークフロー支援&管理システム, b : ネットワーク管理システム, c : ジョブ管理・監視システム, d : プロセス制御システム, e : セキュリティシステム, f : 金融取引処理システム, g : レポートینگ, h : オンライン解析&レポートینگ, i : データ管理/マイニングシステム, j : Web ポータルサイト, k : ERP, l : SCM, m : CRM/CTI, n : 文書管理, o : ナレッジマネジメントシステム, p : カタログ処理・管理システム, q : 数学モデリング(金融/工学), r : 3D モデリング/アニメーション, s : 地理/位置/空間情報システム, t : グラフィクス&出版ツール/システム, u : 画像, v : ビデオ, w : 音声処理システム, x : 組み込みソフトウェア(for 機械制御), y : デバイスドライバ/インタフェースドライバ, z : OS/ソフトウェアユーティリティ, A : ソフトウェア開発ツール, B : 個人向け製品(ワープロ, 表計算ソフトなど), C:EDI, D : EAI, E : エミュレータ, F : ファイル転送, G : その他(具体的名称)
204_利用形態	開発した情報システムの利用形態(特定ユーザの利用か、不特定ユーザの利用か)。	a : 特定ユーザの利用, b : 不特定ユーザの利用
205_利用者数	204 が「特定ユーザの利用」の場合、情報システムを利用するユーザ数。	約()人

データ名称	定義	回答内容、選択肢
206_利用拠点数	開発した情報システムの設置拠点数(サーバ設置場所数など)。	()ヶ所
207_同時最大利用ユーザ数	開発した情報システムを同時に利用するユーザ数の最大値。	()人

(3)システム特性

データ名称	定義	回答内容、選択肢
301_システムの種別	開発した情報システムの種別。	a : アプリケーションソフト, b : システムソフト(ミドルウェア、OS), c : ツール類, d : 開発環境ソフト, e : その他(具体的名称)
302_業務パッケージ利用の有無	当該プロジェクトにおける業務パッケージソフトの利用の有無。 自社開発したパッケージソフトは除く。	a : 有り, b : 無し
303_業務パッケージの初回利用か否か	302 が「有り」の場合、その業務パッケージを初めて利用するの否か。	a : 初回利用, b : 過去に経験有り, c : 経験度がわからない
304_業務パッケージの名称	302 が「有り」の場合、パッケージの名称。例 . SAP、Oracle Applications。	()
305_パッケージの機能規模の比率	302 が「有り」の場合、システム全体の機能規模に対するパッケージの機能規模の概算比率 (感覚的な値で良い)。	約()%
306_パッケージのカスタマイズの度合い	302 が「有り」の場合、カスタマイズ金額÷パッケージの金額。	()%
307_処理形態	開発した情報システムの処理形態。	a : バッチ処理, b : 対話処理, c : オンライントランザクション処理, d : その他(具体的名称)
308_アーキテクチャ	アーキテクチャの種類。 複数ある場合は、開発規模の大きい順に3つまで選択。	a : スタンドアロン, b : メインフレーム, c : 2階層クライアント/サーバ, d : 3階層クライアント/サーバ, e : イン트라ネット/インターネット, f : その他(具体的名称)
309_開発対象プラットフォーム	主たる開発対象プラットフォーム。	a : Windows95/98/Me 系, b : WindowsNT/2000/XP 系, c : Windows Server 2003, d : HP-UX, e : HI-UX, f : AIX, g : Solaris, h : Redhat Linux, i : SUSE Linux, j : Miracle Linux, k : Turbo Linux, l:その他 Linux, m:Linux, n:その他 UNIX系, o : MVS, p : IMS, q : TRON, r : オフコン, s : その他 OS (具体的名称)
310_Web 技術の利用	Web 技術の利用状況。	a : HTML, b : XML, c : Java Script, d : ASP, e : JSP, f : J2EE, g : Apache, h : IIS, i : Tomcat, j : JBOSS , k : OracleAS , l : WebLogic , m : WebSphere , n : Coldfusion , o : WebService , p : その他 (具体的名称) , q : 無し
311_オンライントランザクション処理	オンライントランザクション処理。	a : TUXEDO, b : CICS, c : OPENTP1, d : その他(具体的名称), e : 無し

データ名称	定義	回答内容、選択肢
312_主開発言語	<p>主たる開発言語。 規模の大きい順に 5 つまで選択。 Web 系の CGI、Java アプレット、EJB など、選択肢にないものは、「w:その他」を選び、具体的名称を記述すること。</p>	<p>a : アセンブラ, b : COBOL, c : PL/I, d : Pro*C, e : C++, f : Visual C++, g : C, h : VB, i : Excel (VBA), (j, k : 欠番), l : InputMan, m : PL/SQL, n : ABAP, o : C#, p : Visual Basic.NET, q : Java, r : Perl, s : Shell スクリプト, t : Delphi, u : HTML, v : XML, w : その他言語(具体的名称)</p>
313_DBMS の利用	<p>当該プロジェクトにおいてDBMSを使用したか否か。</p>	<p>a : Oracle, b : SQL Server, c : PostgreSQL, d : MySQL, e : Sybase, f : Informix, g : ISAM, h : DB2, i : Access, j : HiRDB, k : IMS, l : その他 DB(具体的名称), m : 無し</p>

(4)開発の進め方

データ名称	定義	回答内容、選択肢
401_開発ライフサイクルモデル	開発ライフサイクルモデル。	a : ウォーターフォール, b : 反復型, c : その他(具体的名称)
402_運用ツールの利用	開発において利用した運用ツール。	a : JP1, b : SystemWalker, c : 千手, d : A-Auto, e : その他(具体的名称), f : 無し
403_類似プロジェクトの参照の有無	システム化計画時に過去に実施した類似プロジェクトを参照したか否か。 類似プロジェクトは存在したが、参照できなかった場合は「無し」とする。	a : 有り, b : 無し
404_プロジェクト管理ツールの利用	開発におけるプロジェクト管理ツールの利用の有無。	a : 有り, b : 無し
405_構成管理ツールの利用	開発における構成管理ツールの利用の有無。 構成管理ツールの例, ClearCase、CVS、Subversion、PVCS、SCCS、VSS など。	a : 有り(具体的名称), b : 無し
406_設計支援ツールの利用	開発における設計支援ツールの利用の有無。	a : 有り(具体的名称), b : 無し
407_ドキュメント作成ツールの利用	開発におけるドキュメント作成ツールの利用の有無。	a : 有り(具体的名称), b : 無し
408_デバッグ/テストツールの利用	開発におけるデバッグ/テストツールの利用の有無。	a : 有り(具体的名称), b : 無し
409_CASE ツールの利用	上流/統合 CASE ツールの利用の有無。 v1.0 の「410_統合 CASE ツールの利用」は廃止し、409 に統合。	a : 有り(具体的名称), b : 無し
411_コードジェネレータの利用	コードジェネレータの利用の有無。 社内製ツールで具体的名称を明記できない場合は、“社内開発ツール”も可。	a : 有り(具体的名称), b : 無し
412_開発方法論の利用	開発方法論の利用状況。	a : 構造化分析設計, b : オブジェクト指向分析設計, c : データ中心アプローチ(DOA), d : その他(具体的名称), e : 無し
413_システム化計画書再利用率	システム化計画書の再利用したページ数 ÷ 全ページ数。	()%
414_要件定義書再利用率	要件定義書の再利用したページ数 ÷ 全ページ数。	()%
415_基本設計書再利用率	基本設計書の再利用したページ数 ÷ 全ページ数。	()%
416_詳細設計書再利用率	詳細設計書の再利用したページ数 ÷ 全ページ数。	()%
417_ソースコード再利用率	再利用した SLOC ÷ 全 SLOC。	()%
418_コンポーネント再利用率	ソフトウェアコンポーネント(ライブラリ等)の再利用率(概数)。 再利用した機能規模 ÷ システム全体の機能規模。	約()%
419_テストケース再利用率_結合テスト	結合テストにおいて再利用したテストケース数 ÷ 全テストケース数。	()%

データ名称	定義	回答内容、選択肢
420_テストケース再利用率_総合テスト(ベンダ確認)	総合テスト(ベンダ確認)において再利用したテストケース数÷全テストケース数。	()%
421_テストケース再利用率_総合テスト(ユーザ確認)	総合テスト(ユーザ確認)において再利用したテストケース数÷全テストケース数。	()%
422_開発フレームワークの利用	開発フレームワークの利用の有無。 例 . Struts、.Net、JBOSS、J2EE など。	a : 有り(具体的名称), b : 無し

(5)ユーザ要求管理

データ名称	定義	回答内容、選択肢
501_要求仕様の明確さ	基本設計フェーズ開始時点での要求仕様の明確さ。	a：非常に明確, b：かなり明確, c：ややあいまい, d：非常にあいまい
502_ユーザ担当者の要求仕様関与	ユーザ担当者の要求仕様定義への関与度合い。	a：十分に関与, b：概ね関与, c：関与が不十分, d：未関与 例. a：ユーザが全て作成 b：ベースはユーザが作成し、細部はベンダが作成 c：ラフなものをユーザが作成し、残りはベンダが作成 d：ベンダが全て作成
503_ユーザ担当者のシステム経験	ユーザ担当者のシステム経験の度合い。	a：十分に経験, b：概ね経験, c：経験が不十分, d：未経験 例. システムの説明に対して a：ストレス無く話が通じる b：概ね話が通じる c：多くの点で説明を要する d：全てを説明する必要がある
504_ユーザ担当者の業務経験	ユーザ担当者の対象業務に関する経験の度合い。	a：十分に経験, b：概ね経験, c：経験が不十分, d：未経験 例.(対象業務に関する質問に対して) a：レスポンス良く正確な返答 b：レスポンスは落ちるが正確な返答 c：レスポンス悪く回答に曖昧さがある d：回答できない
505_ユーザとの役割分担・責任所在の明確さ	ユーザ・ベンダ間の役割分担・責任所在の明確度。	a：非常に明確, b：概ね明確, c：やや不明確, d：不明確
506_要求仕様に対するユーザ承認の有無	要求仕様に対するユーザ担当者の承認の有無。	a：有り, b：無し
507_ユーザ担当者の設計内容の理解度	ユーザ担当者の設計内容に対する理解度。	a：十分に理解, b：概ね理解, c：理解が不十分, d：全く理解していない
508_設計内容に対するユーザ承認の有無	設計内容に対するユーザ担当者の承認の有無。	a：有り, b：無し
509_ユーザ担当者の受け入れ試験関与	ユーザ担当者が主体的に「総合テスト(ユーザ確認)」に関与したか否か。	a：十分に関与, b：概ね関与, c：関与が不十分, d：全く関与していない
5114~5121_要求仕様変更の発生状況(フェーズ別)	各フェーズ(工程)での仕様変更の発生有無、及び工数への影響度合い。	フェーズ別に以下を記入。 a：変更なし, b：軽微な変更が発生, c：大きな変更が発生, d：重大な変更が発生
511_要件決定者の人数	実質的なキーマン(要件決定者)の人数。	()人
512_要求レベル(信頼性)	システムの故障の頻度、故障状態からの回復時間・影響を受けたデータの修復などに関する、要求の厳しさ。	a：極めて高い, b：高い, c：中位, d：低い
513_要求レベル(使用性)	利用者にとってソフトウェアが理解しやすいか、適用法を習得しやすい	a：極めて高い, b：高い, c：中位, d：低い

データ名称	定義	回答内容、選択肢
	か、運用管理しやすいか、またグラフィカル・デザインなど魅力的であるかなどに関する、要求の厳しさ。	
514_要求レベル (性能・効率性)	システムを実行する際の応答時間・処理時間・処理能力、及びディスク・メモリのハードウェア・その他の資源の使用量などに関する、要求の厳しさ。	a : 極めて高い, b : 高い, c : 中位, d : 低い
515_要求レベル (保守性)	ソフトウェアの修整に関して、故障箇所・原因の特定のしやすさ、変更作業のしやすさ、修正の際の予期せぬ影響の防止、修正の妥当性の確認のしやすさなどに関する、要求の厳しさ。	a : 極めて高い, b : 高い, c : 中位, d : 低い
516_要求レベル (移植性)	ソフトウェアをある環境から他の環境に移す際の、新環境への順応のさせやすさ、設置のしやすさ、他のソフトウェアとの共存のさせやすさ、他のソフトウェアからの置き換えのしやすさなどに関する、要求の厳しさ。	a : 極めて高い, b : 高い, c : 中位, d : 低い
517_要求レベル (ランニングコスト要求)	システムのランニングコストに関する要求の厳しさ。	a : 極めて高い, b : 高い, c : 中位, d : 低い
518_要求レベル (セキュリティ)	システムのセキュリティに関する要求の厳しさ。	a : 極めて高い, b : 高い, c : 中位, d : 低い
519_法的規制の有無	法的規制の有無。	a : 業法レベルの規制あり, b : 一般法レベルの規制あり, c : 規制なし 業法の例 : 銀行業法、証券取引法

(6)要員等スキル

データ名称	定義	回答内容、選択肢
601_PM スキル	プロジェクトマネージャ(PM)のスキル。 ITスキル標準(V2.0以降)の職種「プロジェクトマネジメント」で評価する。 レベルの達成度指標、スキル熟達度については、「ITスキル標準(V2.0以降)プロジェクトマネジメント」(http://www.ipa.go.jp/jinzai/itss)を参照のこと。	a: レベル6、レベル7 b: レベル5 c: レベル4 d: レベル3
602_要員スキル_業務分野の経験	開発する情報システムの対象業務に関するプロジェクトメンバの経験の度合い。	a: 全員が十分な経験 b: 半数が十分な経験、残り半数はいくらかの経験 c: 半数がいくつかの経験、残り半数は経験なし d: 全員が経験なし
603_要員スキル_分析・設計経験	プロジェクトメンバの分析・設計の経験の状況。	a: 全員が十分な経験 b: 半数が十分な経験、残り半数はいくらかの経験 c: 半数がいくつかの経験、残り半数は経験なし d: 全員が経験なし
604_要員スキル_言語・ツール利用経験	プロジェクトメンバの言語・ツールの経験の状況。	a: 全員が十分な経験 b: 半数が十分な経験、残り半数はいくらかの経験 c: 半数がいくつかの経験、残り半数は経験なし d: 全員が経験なし
605_要員スキル_開発プラットフォームの使用経験	プロジェクトメンバの開発プラットフォームの使用経験の状況。	a: 全員が十分な経験 b: 半数が十分な経験、残り半数はいくらかの経験 c: 半数がいくつかの経験、残り半数は経験なし d: 全員が経験なし

(参考) 601_PM スキル に関して、白書のデータ定義 v1.0 の選択肢と、データ定義 v2.0 以降の選択肢の対応付けは、次の表のようになっている。

IPA/SEC データ項目の旧定義 (v1.0 まで)	IPA/SEC データ項目の定義(v2.0 以降)で IT スキル標準の定義の対応	専門分野「システム開発/アプリケーション開発/システムインテグレーション」におけるサイズ指標
a : 多数の中・大規模で複雑なプロジェクトの管理を経験	a:レベル 6、レベル 7	管理する要員数がピーク時 500 人以上、または年間契約金額 10 億円以上。
	a:レベル 6、レベル 7	管理する要員数がピーク時 50 人以上 500 人未満、または年間契約金額 5 億円以上。
b : 少数の中・大規模で複雑なプロジェクトの管理を経験	b:レベル 5	管理する要員数がピーク時 10 人以上 50 人未満、または年間契約金額 1 億円以上 5 億円未満。
c : 小・中規模プロジェクトの管理しか経験していない	c:レベル 4	管理する要員数がピーク時 10 人未満、または年間契約金額 1 億円未満。
d : プロジェクト管理の経験なし	d:レベル 3	特定せず。

(7)システム規模

データ名称	定義	回答内容、選択肢
701_FP 実績値の計測手法	FP 実績値の算出に使用した計測手法。 ただし、ユースケースポイントは含まない。	a : IFPUG, b : SPR, c : NESMA 試算, d : NESMA 概算, e : COSMIC-FFP, f : その他(具体的名称)
10124_FP 実績値の計測手法の純度, 10125_同具体名称	FP 実績値の算出に使用した計測手法の計測ルールへの準拠度。 a : 計測ルール (ISO や JIS など標準のルール) に準拠 b : 自社でルールをカスタマイズ	a : オリジナル版 b : カスタマイズ版 (具体的名称があれば記述)
702_FP 計測の支援技術	FP 計測ツールの利用の有無(もしくはFP 計測専任者の有無)。	a : 有り(ツール利用 or 計測専任者), b : 無し
11018_FP 母体包含	103_ 開発プロジェクトの種別が、b : 改修・保守、または、d : 拡張の場合、5001_FP 実績値 (調整前) に母体規模の包含を示す識別フラグ。	0 : 不明 , 1 : 含まない , 2 : 含む
FP 計画値の推移と計画値の計測手法名		
5082_調整前FP値_システム化計画後, 10116_同手法, 10117_同具体名称	システム化計画後の調整前 FP 値。 及び、FP 値の計測手法名(その他の場合は具体名称)。	()FP ()手法
5083_調整前FP値_要件定義後, 10118_同手法, 10119_同具体名称	要件定義後の調整前 FP 値。 及び、FP 値の計測手法名(その他の場合は具体名称)。	()FP ()手法
5084_調整前FP値_基本設計後, 10120_同手法, 10121_同具体名称	基本設計後の調整前 FP 値。 及び、FP 値の計測手法名(その他の場合は具体名称)。	()FP ()手法
5085_調整前FP値_詳細設計後, 10122_同手法, 10123_同具体名称	詳細設計後の調整前 FP 値。 及び、FP 値の計測手法名(その他の場合は具体名称)。	()FP ()手法
FP 実績値		
5001_FP 実績値 (調整前)	総合テスト(ベンダ確認)完了時の調整係数適用前の FP 値。	()FP
5002_FP 実績値 (調整後)	総合テスト(ベンダ確認)完了時の調整係数適用後の FP 値。	()FP
5003_調整係数	FP の調整係数。	()
706_調整前 FP 値の信頼性	調整前 FP 値の信頼度を4段階 (A ~ D) で評価した値。 データ収集の事務局が客観的に評価して記入する。	A : 調整前 FP 値に合理性があり、完全に整合していると認められる。 B : 調整前 FP 値に合理性が認められるが、調整後 FP 値と調整係数の片方のみ提出されているため、その整合性を評価できない。 C : 調整前 FP 値、もしくはFP 詳細値が提出されていないため、調整前 FP 値を算出できない。 D : 調整前 FP 値の信頼性に乏しいと判断できる要因が1つもしくは複数認められる。
FP 詳細値 (IFPUG の場合)		

データ名称	定義	回答内容、選択肢
5026 ~ 5033_EI	External Inputs。 計画値があれば記入する。	・機能数： 大 (), 中 (), 小 () ・FP 数：()
5034 ~ 5041_EO	External Outputs。 計画値があれば記入する。	・機能数： 大 (), 中 (), 小 () ・FP 数：()
5042 ~ 5049_EQ	External Enquiries。 計画値があれば記入する。	・機能数： 大 (), 中 (), 小 () ・FP 数：()
5050 ~ 5057_ILF	Internal Logical Files。 計画値があれば記入する。	・機能数： 大 (), 中 (), 小 () ・FP 数：()
5058 ~ 5065_EIF	External Interface Files。 計画値があれば記入する。	・機能数： 大 (), 中 (), 小 () ・FP 数：()
FP 詳細値 (IFPUG 以外の場合)		
5066 ~ 5069_ト ランザクションフ ァンクション	IFPUG の場合の、EI、EO、EQ に相 当。 計画値があれば記入する。	機能数(), FP 数()
5070 ~ 5073_デー タファンクション	IFPUG の場合の、ILF、EIF に相当。 計画値があれば記入する。	機能数(), FP 数()
改修 FP 実績値 (5022 ~ 5025)	改修プロジェクトの場合、以下に示す 4 つの FP 詳細値。 ・母体 FP(5022) ・追加 FP(5023) ・変更 FP(5024) ・削除 FP(5025)	母体：()FP 追加：()FP 変更：()FP 削除：()FP
改修 FP 計画値 (11007 ~ 11010)	母体、追加、変更、削除の各 FP 計画 値。 ・母体 FP(11007) ・追加 FP(11008) ・変更 FP(11009) ・削除 FP(11010) 対応する FP 実績値(5022 ~ 5025) の値がある場合は必須。	母体：()FP 追加：()FP 変更：()FP 削除：()FP
COSMIC-FFP の詳細値		
5074_トリガーイ ベント数	COSMIC-FFP のトリガーイベント 数。	()
5075_機能プロセ ス数	COSMIC-FFP の機能プロセス数。	()
5076_データグル ープ数	COSMIC-FFP のデータグループ数。	()
5077_Entry	COSMIC-FFP の Entry 値。	()
5078_Exit	COSMIC-FFP の Exit 値。	()
5079_Read	COSMIC-FFP の Read 値。	()
5080_Write	COSMIC-FFP の Write 値。	()
5081_Cfsu	COSMIC-FFP の Cfsu 値。	()
SLOC 計画値の推移		
5086_システム化 計画後	システム化計画終了後の SLOC 計画 値。	()SLOC
5087_要件定義後	要件定義終了後の SLOC 計画値。	()SLOC

データ名称	定義	回答内容、選択肢
5088_基本設計後	基本設計終了後の SLOC 計画値。	()SLOC
5089_詳細設計後	詳細設計終了後の SLOC 計画値。	()SLOC
SLOC 実績値		
SLOC 実績値 (5004, 5005, 5006, 10086, 10087)	総合テスト(ベンダ確認)完了時の ・ SLOC 値(5004) ・ コメント行取り扱い(5005)、同比率(10086) ・ 空行取り扱い(5006)、同比率(10087)。 1 FP 値がない場合は必須。 FP 値がある場合も SLOC 値が計測できていれば記述。 2 SLOC の単位は Line (KiloLine ではない)。	()SLOC コメント行： a：含む, b：含まず a：含む場合、コメント行比率を 5%刻みで記述(例 . 25%) 空行： a：含む, b：含まず a：含む場合、空行比率を 5%刻みで記述(例 . 25%)
11003_SLOC 実績値(母体)	5004 の規模の値がある場合、その母体 SLOC 値を記述。 SLOC の単位は Line (KiloLine ではない)。	()SLOC
11004_SLOC 実績値(追加・新規)	5004 の規模の値がある場合、その追加・新規 SLOC 値を記述。 SLOC の単位は Line (KiloLine ではない)。	()SLOC
11005_SLOC 実績値(変更)	5004 の規模の値がある場合、その変更 SLOC 値を記述。 SLOC の単位は Line (KiloLine ではない)。	()SLOC
11006_SLOC 実績値(削除)	5004 の規模の値がある場合、その削除 SLOC 値を記述。 SLOC の単位は Line (KiloLine ではない)。	()SLOC
11011_SLOC 計画値(母体)	5004 の規模の値がある場合、その計画段階の母体 SLOC 値を記述。 SLOC の単位は Line (KiloLine ではない)。	()SLOC
11012_SLOC 計画値(追加・新規)	5004 の規模の値がある場合、その計画段階の追加・新規 SLOC 値を記述。 SLOC の単位は Line (KiloLine ではない)。	()SLOC
11013_SLOC 計画値(変更)	5004 の規模の値がある場合、その計画段階の変更 SLOC 値を記述。 SLOC の単位は Line (KiloLine ではない)。	()SLOC
11014_SLOC 計画値(削除)	5004 の規模の値がある場合、その計画段階の削除 SLOC 値を記述。 SLOC の単位は Line (KiloLine ではない)。	()SLOC

データ名称	定義	回答内容、選択肢
5007 ~ 5021, 10001 ~ 10005, 10088 ~ 10097_言 語別 SLOC 実績値	開発言語が複数言語の場合、言語別に 上位 5 言語について次の項目。プロジ ェクト内で使用言語の規模の多いもの から順に記載する。 ・言語名称 (10001 ~ 10005) ・SLOC 値(5007 ,5010 ,5013 ,5016 , 5019) ・コメント行取り扱い (5008 , 5011 , 5014 , 5017 , 5020) ・コメント行比率 (10088 , 10090 , 10092 , 10094 , 10096) ・空行取り扱い (5009、5012、5015、 5018、5021) ・空行比率 (10089 , 10091 , 10093 , 10095 , 10097)	a : 言語(), ()SLOC b : 言語(), ()SLOC c : 言語(), ()SLOC d : 言語(), ()SLOC e : 言語(), ()SLOC 各々について、以下から選択。 ・コメント行： a : 含む, b : 含まず a : 含む場合、コメント行比率を 5%刻みで記述(例 . 25%) ・空行： a : 含む, b : 含まず a : 含む場合、空行比率を 5%刻みで記述(例 . 25%)
11017_SLOC 母体 包含	103_開発プロジェクトの種別が、 b: 改修・保守、または d : 拡張の場合、 5004_SLOC 実績値に母体規模の包含 を示すフラグ。	0 : 不明 , 1 : 含まない , 2 : 含む
設計書の文書量(実績値)		
5090_システム化 計画書	システム化計画書の実測ページ数。	()ページ ・ページの定義は各社の定義に従う
5091_要件定義書	要件定義書の実測ページ数。	()ページ ・ページの定義は各社の定義に従う
5092_基本設計書	基本設計書の実測ページ数。	()ページ ・ページの定義は各社の定義に従う
5093_詳細設計書	詳細設計書の実測ページ数。	()ページ ・ページの定義は各社の定義に従う
その他規模指標		
5094_DFD データ 数	DFD (データフロー・ダイアグラム) のデータ数。	()
5095_DFD プロセ ス数	DFD のプロセス数。	()
5096_DB テー ブル数	DB (データベース) のテーブル数。	()
5097_画面数	画面数。	()
5098_帳票数	帳票数。	()
5099_バッチ本数	バッチプログラムの本数。	()
5100 ~ 5102_ユー スケース数	ユースケース数。単純(5100)、平均的 (5101)、複雑(5102)の 3 段階で記述。	単純 : () 平均 : () 複雑 : ()
5103 ~ 5105_アク ター数	アクター数。単純 (5103)、平均的 (5104)、複雑(5105)の 3 段階で記述。	単純 : () 平均 : () 複雑 : ()

(8)工期

データ名称	定義	回答内容、選択肢
5123 ~ 5148_工 程別工期(計画)	工程別開始年月[日](計画)、終了年月[日](計画)。「工程別終了年月[日](計画) - 工程別開始年月[日](計画)」で計算した月数(小数点第一位まで)でも可。工程配分不可の月数は、工程区分が不明な場合の総月数(計画)。	以下のいずれかで回答。yyyy/mm/dd。 ddは省略可能。 ・開始年月[日]、 終了年月[日] ・月数()ヶ月
5150 ~ 5175_工 程別工期(実績)	工程別開始年月[日](実績)、終了年月[日](実績)。「工程別終了年月[日](実績) - 工程別開始年月[日](実績)」で計算した月数(小数点第一位まで)でも可。工程配分不可の月数は、工程区分が不明な場合の総月数(実績)。	以下のいずれかで回答。yyyy/mm/dd。 ddは省略可能。 ・開始年月[日]、 終了年月[日] ・月数()ヶ月
5122, 5131, 5140_プロジェ クト全体工期(計画)	開始年月[日](計画)、終了年月[日](計画)。 月数は「プロジェクト終了年月[日](計画) - プロジェクト開始年月[日](計画)」で自動計算される。 開始日 = 工数が発生する日 終了日 = 工数が発生する最後の日	以下のいずれかで回答。yyyy/mm/dd。 ddは省略可能。 ・開始年月[日]、 終了年月[日] ・月数()ヶ月
5149, 5158, 5167_プロジェ クト全体工期(実績)	開始年月[日](実績)、終了年月[日](実績)。 月数は「プロジェクト終了年月[日](実績) - プロジェクト開始年月[日](実績) - アイドリング期間」で自動計算される。 開始日 = 工数が発生した日 終了日 = 工数が発生した最後の日。 例：発注者の検収が完了した日、納品日。	以下のいずれかで回答。yyyy/mm/dd。 ddは省略可能。 ・開始年月[日]、 終了年月[日] ・月数()ヶ月
806_アイドリ ング期間	プロジェクトの非活動期間月数 (例：顧客のサイン待ち、テストデータの受領待ち)。この月数をプロジェクトの総工期から引くと、プロジェクトの活動期間が算出される。	()ヶ月

(9)工数(コスト)

データ名称	定義	回答内容、選択肢
901_工数の単位	工数の単位を人時、人月から選択する。	a：人時, b：人月
902_人時換算係数	工数の単位が人月の場合の人時への換算係数。 例．1人月 = 160人時	・ 1人月 = ()人時
5106 ~ 5113_プロジェクト総工数に含まれるフェーズ	開発プロジェクトに「システム化計画」～「総合テスト(ユーザ確認)」までの各フェーズが含まれているか否か。 該当フェーズに相当する作業の有無を記述。 【回答は次の定義から選択】 ○：作業があり、工数等のデータをこのフェーズの欄に記入する場合 ×：作業が無い場合 △：作業があるが、当該フェーズに相当する作業工数等のデータは、他フェーズの欄に合算して記入する場合。 複数フェーズの作業をまとめて一フェーズとして管理する場合や、データが合計でのみ把握できる場合、まとめた工数データは、後の方の工程の欄に両方の作業の合計工数を記録する。 例．基本設計・詳細設計・製作のデータを合計で記入する場合は、基本設計は 、詳細設計は 、製作に を記入する。	・ システム化計画 () ・ 要件定義 () ・ 基本設計 () ・ 詳細設計 () ・ 製作 () ・ 結合テスト () ・ 総合テスト(ベンダ確認) () ・ 総合テスト(ユーザ確認) ()
社内実績工数	社員(社員と一緒に作業する派遣社員を含む)の実績工数 (a)開発：開発作業工数(5177～5184, 10130) (b)管理：管理作業工数(5186～5193, 10131) (c)その他：開発、管理に分類されない実績工数。(10007～10014, 10132) 例．テスト環境構築、インフラ構築、運用構築、移行、業務支援、コンサルティングなど (d)作業配分不可：開発、管理、その他に分類されない実績工数。(10133～10141) フェーズ別の値のみ入力し、プロジェクト全体は自動入力。 フェーズ別の「工程配分不可」には、工程(フェーズ)区分が不明の実績工数を記述。	・ 開発 () ・ 管理 () ・ その他 () ・ 作業配分不可 ()
レビュー実績工数	社内のレビュー実績工数(社内工数の内数)。 フェーズ別：5206～5213, 10146 プロジェクト全体(5205)は自動入力。	()
レビュー実績回数	レビュー回数。 フェーズ別：5215～5222, 10147 プロジェクト全体(5214)は自動入力。	()回
レビュー指摘件数	レビュー指摘数。 フェーズ別：5249, 5250, 10078～10083, 10150 プロジェクト全体(10077)は自動入力。	()件
外部委託工数	外部委託の開発工数(社内工数の外数)。 フェーズ別：5196～5203, 10145 プロジェクト全体(5195)は自動入力。	()

外部委託作業有無 (10033 ~ 10040 , 10144)	開発作業の外部委託の有無。 外部委託工数を入力すると が自動入力される。	<自動入力>
5204_外部委託金額比率	外部委託工数が不明の場合に、全体金額に対する外部委託金額比率を記述。	()%
データ名称	定義	回答内容、選択肢
社内平均要員数 (5223 ~ 5231)	社内の平均要員数。	()人
社内ピーク要員数 (5232 ~ 5240)	社内のピーク要員数。	()人
外部委託平均要員数 (10059 ~ 10067)	外部委託の平均要員数。	()人
外部委託ピーク要員数 (10068 ~ 10076)	外部委託のピーク要員数。	()人
11015_プロジェクト開発工数計画値(基本設計開始時点)	プロジェクト全体の開発工数(社内及び外部委託)の基本設計開始時点の計画値。	()
11016_プロジェクト開発工数計画値(詳細設計開始時点)	プロジェクト全体の開発工数(社内及び外部委託)の詳細設計開始時点の計画値。	()

(10)品質

データ名称	定義	回答内容、選択肢
稼働後の不具合		
5267 ~ 5270 , 10112 ~ 10115_ 発生不具合総数	システム稼働後(サービスイン後)に報告された不具合の総数。現象数と原因数に分ける。 それぞれの数は一定期間経過時点の累計で表す。つまり 1 ヶ月経過時点の合計値、3 ヶ月経過時点の累計値、6 ヶ月経過時点の累計値で表す。 1 例として、稼働後 5 ヶ月しか経過していない場合は、1 ヶ月、3 ヶ月の値のみ記入する。 2 サービスイン日が不明な場合は記入しない。	稼働から次の期間の累計 ・1 ヶ月：現象数：()、原因数：() ・3 ヶ月：現象数：()、原因数：() ・6 ヶ月：現象数：()、原因数：() 複数記入可
5255 ~ 5266 , 10100 ~ 10111_ 発生不具合数(重大性別内訳)	上記値の不具合重大度(重大、中度、軽微)別の内数。現象数と原因数に分ける。 【重大性の定義】 ・重大：顧客へ損害を与え、緊急対応を要する ・中度：顧客への損害はないが、緊急対応を要する ・軽微：顧客への損害はなく、緊急対応も不要 それぞれの数は一定期間経過時点の累計で表す。つまり 1 ヶ月経過時点の合計値、3 ヶ月経過時点の累計値、6 ヶ月経過時点の累計値で表す。 1 例として、稼働後 5 ヶ月しか経過していない場合は、1 ヶ月、3 ヶ月の値のみ記入する。 2 サービスイン日が不明な場合は記入しない。	稼働から次の期間の累計 ・1 ヶ月：重大、中度、軽微の現象数：()、原因数：() ・3 ヶ月：重大、中度、軽微の現象数：()、原因数：() ・6 ヶ月：重大、中度、軽微の現象数：()、原因数：() 複数記入可
テストフェーズ別テストケース数		
5251, 1005_結合テスト	結合テストケース数(5251)、結合テストケース数定義(1005)	・テストケース数：() ・テストケース(数)の定義について補足(任意回答)。
5252, 1005_総合テスト(ベンダ確認)	総合テストケース数(5252)、総合テストケース数定義(1005)	・テストケース数：() ・テストケース(数)の定義について補足(任意回答)。
テストフェーズ別検出バグ数		
5253, 10098, 1007_結合テスト	検出バグ現象数(5253)、検出バグ原因数(10098)、バグ数定義(1007)	・検出バグ数： 現象数：()、原因数：() ・バグ(数)の定義について補足(任意回答)。
5254, 10099, 1007_総合テスト(ベンダ確認)	検出バグ現象数(5254)、検出バグ原因数(10099)、バグ数定義(1007)	・検出バグ数： 現象数：()、原因数：() ・バグ(数)の定義について補足(任意回答)。
5241_品質保証体制	開発中の品質保証の体制。 定義 v1.0 のフェーズ別設問 5242 ~ 5248 は v2.0 以降は廃止した。	a：プロジェクトメンバが実施。 b：品質保証の専任スタッフが実施。 c：実施していない。

データ名称	定義	回答内容、選択肢
1010_テスト体制	テスト体制。	a：スキル、員数ともに十分 b：スキルは十分、員数は不足 c：スキルは不足、員数は十分 d：スキル、員数ともに不足
1011_定量的な出荷品質基準の有無	対象プロジェクトにおいて定量的な出荷品質基準が設定されていたか否か。	a：有り(具体的に記述), b：無し
1013_第三者レビューの有無	第三者レビューを実施しているか否か。 第三者：プロジェクトに関係しない人員。 例．品質保証部門、PMO。	a：有り、b：無し