

国際規格に基づく総合的なソフトウェア品質評価の枠組みとその実製品への適用による品質ベンチマーク



鷲崎 弘宜^{*1*}^{*4}



津田 直彦^{*1}



本田 澄^{*1}



中井 秀矩^{*1}



深澤 良彰^{*1}



東 基衛^{*1}



込山 俊博^{*2}



中野 正^{*3}



鈴木 啓紹^{*3}

ソフトウェア製品の品質を業界の中で定量的に明らかとし改善可能とするため、多面的な品質を測定評価する共通の枠組みが必要である。我々は国際規格シリーズ SQuaRE (Systems and software Quality Requirements and Evaluation) の具体化により、製品によらず共通に製品品質と利用時品質を総合的に測定評価可能な枠組みを実現した。更に枠組みを 21 製品に実適用し、品質特性別の傾向、品質特性間の関係、利用時品質・製品品質間の関係、及び製品コンテキストと品質特性の関係の一端を、適用した範囲において明らかとし、品質ベンチマークとして公開した。

Software Quality Evaluation Framework based on International Standards and Benchmark obtained by applying the Framework

Hironori Washizaki^{*1*}^{*4}, Naohiko Tsuda^{*1}, Kiyoshi Honda^{*1}, Hidenori Nakai^{*1}, Yoshiaki Fukazawa^{*1}, Motoei Azuma^{*1}, Toshihiro Komiyama^{*2}, Tadashi Nakano^{*3}, Hirotsugu Suzuki^{*3}

※1 早稲田大学 ※2 日本電気株式会社 ※3 一般社団法人コンピュータソフトウェア協会
 ※4 国立情報学研究所, 株式会社システム情報, 株式会社エクスマーシオン

We established a comprehensive software quality evaluation framework based on ISO/IEC 25000 series SQuaRE (Systems and software Quality Requirements and Evaluation). Our framework successfully realized many product measures and quality in use measures originally defined in the SQuaRE. By applying the framework to 21 commercial ready-to-use software products, we revealed the current status of software product quality and opened it as a benchmark to the public. The benchmark includes the trends of quality measurement values, relationships among quality characteristics, relationship between quality in use and product quality, and relationship between quality characteristics and product contexts within the limits of the application.

1 はじめに

ソフトウェア品質の向上は現代社会の最重要課題の一つであり [東 15], その実現に向けて利用者のほか多様な利害関係者への影響に着目したソフトウェアの利用時の品質の把握, 開発者視点の製品品質の把握, 並びにその間の関係の把握が欠かせない. しかしながら従来のソフトウェア製品の品質評価の取り組みは, 機能適合性や信頼性などの特定の品質特性に集中しており, 様々な品質特性を多面的かつ具体的に捉える取り組みを欠く. 結果として, 製品や組織を超えた測定データが限られており, 品質の実態, 更には品質特性間や利用時の品質と製品品質間の関係は不明であり, 効果的な品質向上の取り組みの検討と実施の妨げとなっている.

製品や組織を超えた多面的, 客観的かつ標準的な品質測定評価に役立てられることが期待される枠組みとして, ISO/IEC 25000 シリーズ SQuaRE [ISO25000] が挙げられる. SQuaRE では, 特定のドメインや製品によらずに, 一般的に重要と考えられる品質特性, 品質測定量及び評価方法をまとめている. しかしながらそれらの定義は国際規格という性質上, 汎用的かつ抽象的なものにとどまり, 利用にあたって具体化が必要である.

そこで我々は, IPA から 2015 年度ソフトウェア工学分野の先導的研究支援事業 (RISE) の委託を受けて, 第三者が利用可能な形で提供されるパッケージソフトウェアやクラウドアプリケーションなどのソフトウェア製品を対象に, 製品や組織を超えて共通に適用できるように SQuaRE における品質測定量を品質測定法として具体化及び拡張し, 得られる測定値を多面的な品質特性群の単位でまとめ上げる総合的な品質評価の枠組みを実現した. 更に同枠組みを 21 の実ソフトウェア製品群に共通に適用することで, 品質特性別の傾向, 品質特性間の関係, 利用時品質・製品品質間の関係, 及び製品コンテキストと品質特性の関係の一端を, 製品や組織を超えて明らかとした. 我々はそれらの結果を世界初の総合的な品質ベンチマーク WSQB17: Waseda Software Quality Benchmark[WSQB] として公開した. 企業

において枠組みと品質ベンチマークの活用が始まっている [小島 17].

本稿では 2 節で関連研究を説明する. 3 節で品質評価枠組みを提案し, 4 節で製品群への適用による調査の結果を報告すると共に業界などへの提言を述べる. 最後に 5 節で成果や展望をまとめる.

2 関連研究

上流工程で検証される静的な内部品質が, 下流工程で検証される動的な外部品質に影響し, 外部品質が運用時に評価される利用時の品質に影響を与えることがうたわれている [ISO25000]. その前提のもとで, 開発の上流工程では設計や実装において内部品質を作り込み, テストにおいて外部品質を確認及び補正する. しかし, その影響関係は限定された範囲でのみ明らかであり, 結果として総合的及び様々な立場における品質評価に向けて品質測定法や測定結果は十分に活用されていない [鷺崎 07]. 例えば設計モデルやソースコードを測定して内部品質を評価する枠組みは QMOOD[Bansiya02], Ortega らのスイート [Ortega03], 我々の過去のスイート [鷺崎 07][鷺崎 10] など数多く存在するが, 最終的な顧客満足度を含む利用時の品質までを網羅するものではなく, それらの関係も明らかではない.

SQuaRE では利用時の品質と製品品質の代表的な品質特性及びサブカテゴリーとしての品質副特性を 2 層構造でモデル化した ISO/IEC 25010:2011 [ISO25010] (JIS X 25010:2013), 同品質モデルに基づき利用時の品質測定量をまとめた ISO/IEC 25022:2016 [ISO25022], 製品品質測定量をまとめた ISO/IEC 25023:2016 [ISO25023] (JIS X 25023:2018) が策定されている. SQuaRE (及び前身にあたる ISO/IEC 9126 シリーズ) におけるそれらの定義は汎用性を重視し抽象的であり [Arban03][Heidrich14], 直ちに利用可能な具体的なソリューションには至っていない [Biscoglio14]. 加えて, 測定量間の関係も実証されていない.

パッケージ製品やクラウドアプリケーションの品質要求として国際規格 ISO/IEC 25051:2014 [ISO25051] (JIS

X 25051:2016) があり、同規格及び IPA のソフトウェア品質説明のための制度ガイドライン [IPA14] に基づき、コンピュータソフトウェア協会 (CSAJ) により JIS X 25051:2016 に準拠した PSQ 認証制度 [PSQ] が 2013 年から開始されている。PSQ は業界初のソフトウェア品質認証制度であり、一定水準のテスト結果に基づき、ソフトウェア製品が品質要求を満足することを認証するものである。本稿の品質評価枠組みはこれらに対して、同様の製品を対象に品質特性単位の詳細な測定と品質特性間の関係分析を通じて補う関係にある。

開発組織を超えたソフトウェア開発の実態調査としては IPA データ白書 [IPA15] や Cusumano らによる調査 [Cusumano03], Jones による調査 [Jones09] などがあるが、いずれも製品品質と利用時の品質の一部の関係分析にとどまり、品質特性について網羅性を欠く。

3 総合的な品質評価枠組み

本研究が実現する枠組みの目的を、国際規格に基づくことで製品や組織を超えて共通に実適用可能とし、更に、多面的かつ具体的に様々な品質特性を捉えられることとする。本研究の結果として得られる枠組みは、製品の開発側や運用側において、開発・保守・運用中あるいは運用検討中のソフトウェア製品の品質を、客観的、定量的かつ総合的に評価し、開発・保守における品質要求定義や品質改善、取捨選択の判断材料に役立てられることが期待される。実際に、枠組みを企業において製品に適用して品質特性単位の相対的なポジションを明らかにし、以降の品質改善に役立てる活用が始まっている [小島 17]。

実現した総合的な品質評価枠組みの全体像を図 1 に示す。ソフトウェアの利用時の品質と製品品質（外部品質及び内部品質）のすべてについて、ISO/IEC 25051:2014 や PSQ 認証制度を参考に、ISO/IEC 25010:2011 における品質モデルの品質特性を網羅的に扱う。更に、製品の別を問わずにおおむね共通に得られる属性を用いることで ISO/IEC 25022:2016 及び ISO/IEC 25023:2016 における測定量について測定の方法を具体化して測定法としてまとめる。具体化にあたり、我々が実績を有する信頼性評価の仕組みやソースコード解析の仕組みを組み入れる。

測定値を後述の方法でスコアとして正規化し、品質特性単位で集約可能とする。品質特性単位のスコアを用いて、品質特性間の関係及び利用時の品質と製品品質間の関係を分析し、その結果を品質ベンチマークとしてまとめる。また製品ごとに品質評価結果、品質ベンチマークにおける製品のポジション、並びに、品質向上に向けた人手による助

言を品質診断レポートとしてまとめて製品開発元に提供した。レポートの例については [WSQB] を参照されたい。

測定の詳細を以下に説明する。

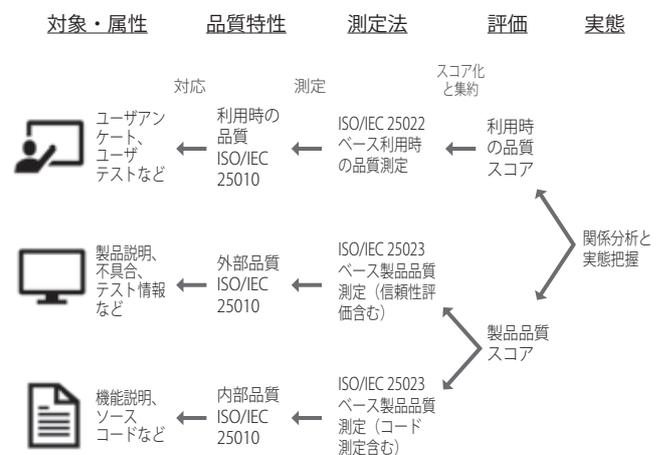


図 1 品質評価枠組みの全体

3.1 製品品質の測定

品質測定法の具体的な定義にあたり、測定可能な事柄や測定法ありきで検討すると、本来の目的を見失う可能性がある [鷲崎 16][Washizaki17]。そこで、目標指向の枠組みである Goal-Question-Metric (GQM) 法 [Basili94] を適用し、製品品質の品質副特性を網羅するように合計 66 の品質測定法を定義した。適用にあたり、JIS X 25051:2016 を参考とし、ISO/IEC 25010:2011 の品質特性を明確に評価することを目標とした。また、ISO/IEC 25023:2016 における製品品質測定量を、測定法として具体化する際の初期候補とした。測定法の一覧を付録 A.1 に示す。

測定法の定義の例を図 2 に示す。図ではセキュリティの副特性である否認防止性について経路のデジタル署名に着目し、目標達成可否を判断するための種々の側面を網羅するように質問を設定し、最終的には測定結果の分かりやすさを重視して種々の側面を総合する形で一つの測定法へまとめあげている。

ISO/IEC 25023:2016 の解釈や測定データの現実的な利用可能性について SQuaRE 及びその JIS 化を担当した国内委員会のメンバからの助言を得た。そして測定に必要なデータを入力し測定値を得るための記入様式を策定し公開した [WSQB]。例えば否認防止性については様式において、経路数及びそのうちの署名経路数の記入を求める。

21 製品に適用した結果、57 の測定法について 1 製品以上で測定値が得られた (表 1)。大多数について、測定値が得られたことから、もともと抽象的な測定法を具体化して実際の製品に適用し、客観的かつ多面的に品質を測定でき

たことが分かる。また測定できた割合は 66 測定法× 21 製品のうちで 34% であった。未測定の原因の多くは根拠となるデータの開発中の未記録である。

測定法導出にあたり目標に基づき、我々が実績を有する信頼性評価及びソースコード測定法を組み入れた。

- テスト時の発見欠陥数の記録に対して信頼性成長モデル [Honda17] を適用して、①欠陥の発見状況を安定したと考えられるタイプ (安定タイプ)、②漸増的に増加しつつあるタイプ (漸増タイプ)、③爆発的に増加しつつあるタイプ (爆発タイプ) の 3 種に大別する。
- ソースコードの静的解析ツールとして Understand と CheckStyle を用いて、保守性の副特性であるモジュール性と再利用性の測定法 (複雑度, 結合度, コーディング規約違反数など) [鷲崎 07][鷲崎 10] を組み入れた。

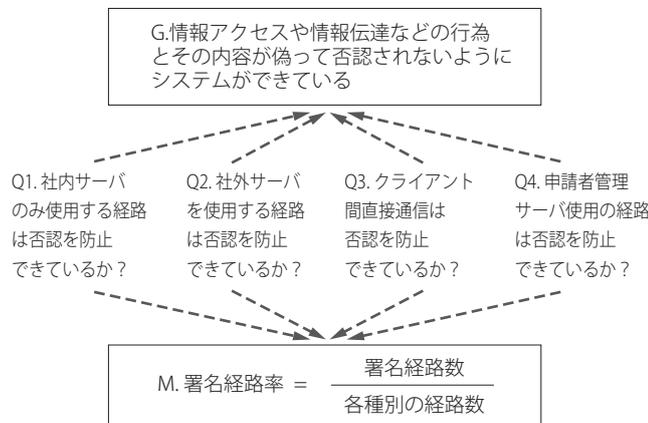


図 2 GQM 法適用例 (G: 目標, Q: 質問, M: 測定法)

表 1 製品品質の測定法の定義数

(あり:測定値が得られたもの, なし:測定値が得られなかったもの)

特性	機能適合性	性能効率性	互換性	使用性	信頼性	セキュリティ	保守性	移植性
あり	4	10	2	9	13	6	8	5
なし	0	1	0	0	3	0	4	1

3.2 利用時の品質の測定

GQM 法を適用し、利用時の品質の品質副特性を網羅するように合計 17 の品質測定法を定義した。適用にあたり、JIS X 25051:2016 を参考とし、ISO/IEC 25010:2011 の品質特性を明確に評価することを目標とした。また、ISO/IEC 25022:2016 における利用時の品質測定量を、測定法として具体化する際の初期候補とした。測定法の一覧を付録 A.2 に示す。そして測定に必要なデータを入力して測定値を得るための記入様式、ユーザアンケート、及び、ユーザテストの実施方法を策定した。

- 満足性、リスク回避性及び利用状況網羅性の測定法の多くは利用者の実感に基づくものであるため、そ

の測定のためのユーザアンケートを作成し公開すると共に、製品開発元から実利用者へ配布及び回収した [WSQB]。

- 有効性と効率性の測定法の多くは利用時の具体的な操作に基づくものであるため、ユーザテストの実施により測定可能な形で定義した。限られた時間内で効率よく主要な機能の正常系及び異常系テストを実施するため、製品開発元が機能一覧及び正常系テスト項目を用意し、その参照により外部評価機関の支援を受けて対象領域やテスト項目への理解を深めながら我々が異常系テスト項目を定義し、我々が実利用者により代替テストした。

21 製品に適用した結果、全 17 の測定法について 1 製品以上で測定値が得られた。すべての測定法について、測定値が得られたことから、もともと抽象的な測定法を具体化して製品へ実適用し、客観的かつ多面的に品質を測定できたことが分かった (表 2)。また測定率は 17 測定法× 21 製品のうちで 24.4% にとどまった。未測定の原因の多くは、製品開発元における都合などの理由によりユーザテスト及びユーザアンケートを実施できなかったことにある。

表 2 利用時の品質の測定法の定義数

特性	有効性	効率性	満足性	リスク回避性	利用状況網羅性
あり	4	2	6	3	2
なし	0	0	0	0	0

3.3 測定値に基づく品質評価

測定値の閾値を、人手を介さずに測定値の分布の違いによらず妥当に決定する方法として、パーセンタイルの利用が知られている [Alves10]。そこで我々の枠組みにおいて、値域の異なる測定法の違いによらず妥当に正規化し集約可能とするために、図 3 に示すように多くの製品群から得られた測定値の集合におけるパーセンタイルのランク値を測定法のスコアとする。測定法のスコアの平均を副特性のスコアとし、その平均を品質特性のスコアとする。ランク値は 0 ~ 1 の値を取るが、{0,1,5,5} のように集合中で最大の

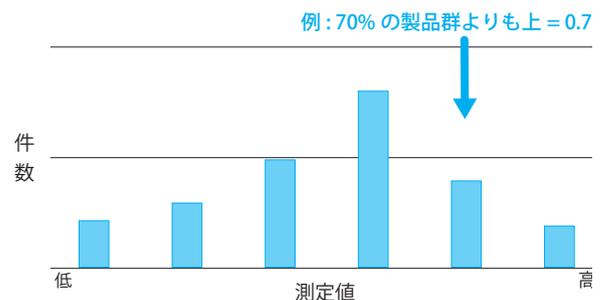


図 3 パーセンタイルによるスコア化

値（ここでは5）が複数ある場合、ランク値の最大値は1にならない（ここでは0.5）。

測定法によっては、特定の品質特性の観点から値が大きいほうが望ましいものと、値が小さいほうが望ましいものがある。例えばセキュリティの観点から、通信経路がデジタル署名によって改ざんから保護されている割合を示す「ネットワーク経路のデジタル署名対応率」は大きいほうが望ましい。一方、保守性の観点から、関数の制御フローグラフにおける線型独立な経路数を示す「関数のサイクロマティック複雑度」は、同じ要求を満足する限り複雑さが抑

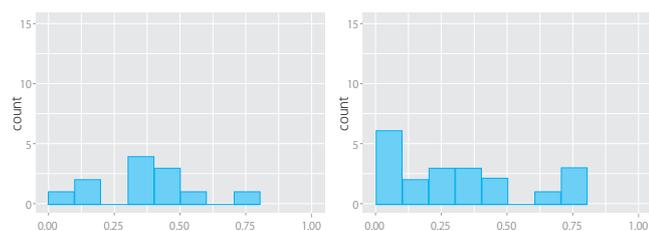


図4 機能適合性（左）、性能効率性（右）の傾向

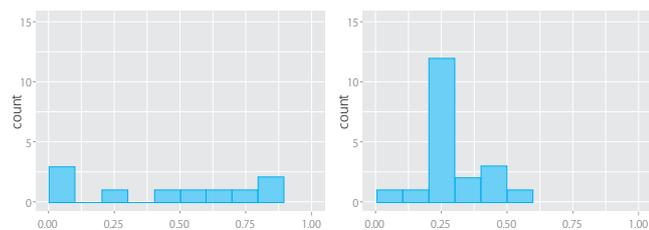


図5 互換性（左）、使用性（右）の傾向

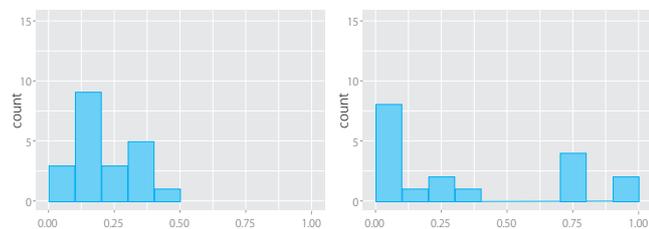


図6 信頼性（左）、セキュリティ（右）の傾向

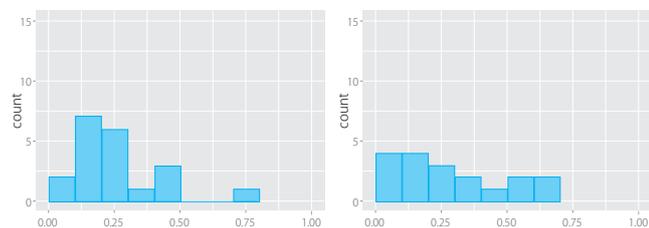


図7 保守性（左）、移植性（右）の傾向

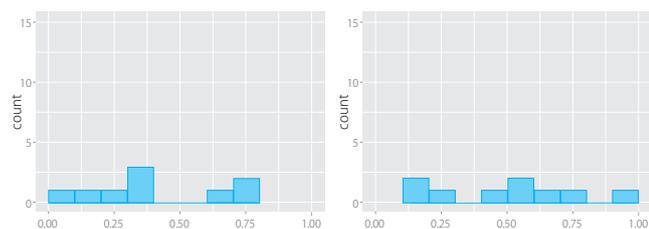


図8 有効性（左）、効率性（右）の傾向

えられていることが望ましいため、値が小さいほうが望ましい。そこで、前者の場合は上位30%のランク値が0.7となるように計算し、後者の場合は逆に下位30%のランク値が0.7となるよう計算する。この方法は、測定値が適当な範囲の値を取ることが望ましい場合を考慮しておらず、今後の課題である。

4 枠組み適用による調査結果

製品や組織を超えた品質の実態が不明であることが、効果的な品質向上の取り組みの検討と実施の妨げとなっている。

そこで我々はCSAJの協力を経て様々な組織が開発した21製品に枠組みを適用し、品質特性別の傾向、品質特性間の関係、利用時品質・製品品質間の関係、及び製品コンテキストと品質特性の関係を測定評価できた範囲において明らかとし、品質ベンチマークとしてまとめた。ただし製品ごとに、提出データに基づき測定評価できた品質特性は異なる。以下に結果を報告する。加えて、品質向上及び測定法拡充の観点から考察する。

4.1 品質特性別の傾向

品質特性ごとに、スコアの製品数分布を図4～図8に示し、それぞれの結果と考察を以下に述べる。最初に製品品質について以下に述べる。

(1) 機能適合性、性能効率性、移植性：

- 結果：分布としてなだらかに広がり、製品によりそれらの品質の程度が異なることが分かる。
- 考察：製品ごとに異なる品質要求の程度に応じた結果と推測され、自然なものとして捉えられる。

(2) 互換性：

- 結果：2極化している。データ交換などの互換性に通じる仕組みを一部の製品において考慮していないことが原因として挙げられる。また、SQuaREにおける定義から具体化できた測定法が2つにとどまったため、1つの測定値がスコアへ多大な影響を及ぼすこととなった。
- 結果：互換性は、品質モデルがISO/IEC 9126-1:2001からISO/IEC 25010:2011へ改訂される際に品質特性として格上げされたものである。国際規格側で、具体化して実適用可能な測定量の拡充が望まれる。

(3) 使用性：

- 結果：得られたスコアの範囲内で低いほうに製品が集中している。

	性能 効率性	互換性	使用性	信頼性	セキュリ ティ	保守性	移植性	有効性	効率性	満足性	リスク回避 性	利用状況網 羅性
機能適合性	0.31	0.19	-0.72	0.37	-0.05	0.50	0.31	-0.14	0.52	1.00	1.00	1.00
性能効率性		0.44	0.24	0.36	-0.17	0.37	0.32	0.32	-0.10	-0.50	-0.50	-0.50
互換性			0.04	0.17	-0.06	0.36	-0.04	-0.14	0.05	-0.50	-0.50	-0.50
使用性				0.17	-0.21	0.11	0.44	-0.09	-0.20	-1.00	-1.00	-1.00
信頼性					0.30	0.41	0.45	-0.08	0.11	1.00	1.00	1.00
セキュリティ						-0.06	0.19	0.64	-0.34	0.50	0.50	0.50
保守性							0.26	-0.29	0.01	1.00	1.00	1.00
移植性								-0.21	0.67	0.50	0.50	0.50
有効性									0.03	-1.00	-1.00	-1.00
効率性										1.00	1.00	1.00
満足性											1.00	1.00
リスク回避性												1.00

図9 品質特性単位の品質スコア間の相関係数

- 考察: 原因としては、使用性を十分に考慮できていないこと、あるいは、エンドユーザ対象ではないといったことから意図的に考慮していない製品が多かった可能性がある。スコアの低い製品について、品質要求との対応関係を確認し、意図通りでない場合は使用性向上のための機能の作り込みといった取り組みが必要と考えられる。

(4) 信頼性:

- 結果: 分布として狭く、全体的に同程度のスコアを取っている。
- 考察: 狭義の品質として高い信頼性を作り込んだ上で製品をリリースしていることがうかがえ、日本らしいソフトウェア製品作りと捉えられる。

(5) セキュリティ:

- 結果: 2 極化している。一部の製品で、暗号化や破損防止などの高セキュリティ化に通じる仕組みを考慮していないことが原因である。
- 考察: すべてがつながる IoT/IoE 時代において、企画時にほかとの接続利用を想定していない製品についても保守や派生の中でほかと接続が求められる可能性がある。従って、製品に必要なセキュリティについて慎重な検討が必要と考えられる。

(6) 保守性:

- 結果: 得られたスコアの比較的広い範囲内で低いほうに製品が集中している。
- 考察: 全体的に保守性を十分に考慮できていない製品が多いと考えられる。製品寿命や品質要求との対応関係を確認し、意図通りでない場合は設計・実装上の複雑さを抑えるといった保守性向上の取り組みが必要と考えられる。

続いて利用時の品質について以下に述べる。

(7) 有効性, 効率性:

- 結果: 2 極化している。ユーザテスト実施時にタスク実行に難がありタスクを達成しにくい製品が一部見られたためである。
- 考察: 製品の価値を判断する立場は本来利用者であり、これらの利用時の品質が低い製品についてユーザ中心の製品デザインといった利用者視点のタスク実行のしやすさの考慮が必要と考えられる。

(8) 満足性, リスク回避性, 利用状況網羅性:

- 結果: 測定評価できた製品数が 3 製品と極めて限られ、意味のある傾向を得られなかった。
- 考察: 多くの製品について開発者視点の製品品質を重視するあまり、利用者視点の利用時の品質の把握と改善を軽視していた可能性がある。これらの品質特性の測定評価のためにユーザアンケートのような利用者の声・実感を直接的に把握する取り組みが不可欠である。

4.2 品質間の関係分析

21 製品のうち測定評価できたものについて品質特性スコア間のスピアマンの順位相関係数を図9に示す。図9に記載した数値は相関係数の大きさである。また、相関係数について p 値が 0.10 未満であるものは偶然に高い相関係数が算出された可能性が低い（つまり統計的に有意）と考えられるため、点線の枠で印を付けた。統計的に有意と見られた関係のそれぞれの結果と考察を以下に述べる。

(1) 信頼性と保守性, 移植性の関係:

- 結果: 信頼性が高いほど保守性や移植性が高い傾向にあった。
- 考察: 高信頼が求められる製品において、併せて、長期間における保守や様々な環境への適合並びに移植が求められる場合が考えられる。そのために正の相関が見られた可能性がある。

(2) 移植性と使用性, 信頼性, 効率性の関係:

- 結果: 移植性が高いほど, 使用性や信頼性及び効率性が高い傾向にあった.
- 考察: 様々な環境に対する移植のしやすさを製品へ作り込む過程において, 併せて, 移植検討先の様々な環境の都合から当該製品の品質を多面的に再検討する場が考えられる. そのために結果として, 高い信頼性などの製品への作り込みにつながった可能性がある.

(3) 機能適合性と使用性の関係:

- 結果: 機能適合性が高いほど, 使用性が低い傾向にあった.
- 考察: 原因として幾つかの可能性が考えられる. 製品の当初の企画通りの機能仕様の満足を最重視した結果, 利用者にとって本来重要な使いやすさを損なってしまったという副作用の可能性もある. また逆に, 使いやすさを重視した結果として, 一部の機能を満足で

きなかったという可能性がある. あるいは品質要求として, そもそもエンドユーザー向けの製品ではないといった理由から, 使いやすさを重視していなかった可能性がある.

(4) セキュリティと有効性の関係:

- 結果: セキュリティが高いほど, 有効性が高い傾向にあった.
- 考察: これは製品によってはセキュリティ関連の機能(ログインなど)が一定割合を占め, それらをユーザーテストにおいて正確に実行できたことが影響している可能性がある.

4.3 欠陥発見状況タイプと品質特性

時系列の欠陥票を得られた9製品について信頼性成長モデルを適用した結果, 欠陥発見状況は安定タイプが3製品, 漸増タイプが3製品, 爆発タイプが3製品であった. 図

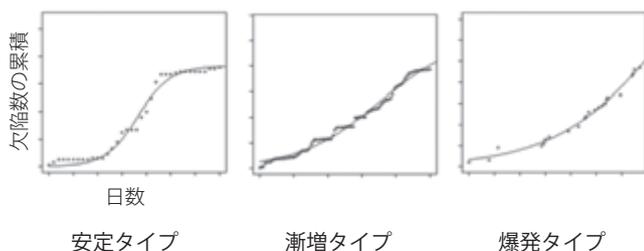


図 10 欠陥発見状況と欠陥数予測 (横軸: 日数, 縦軸: 欠陥数の累積)

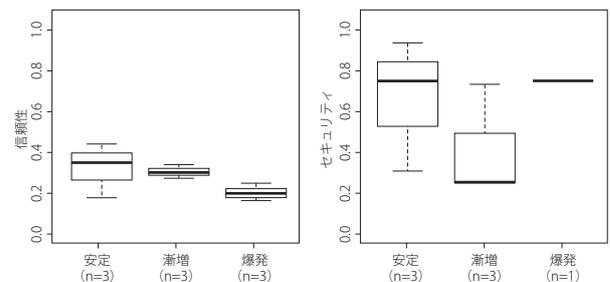


図 13 欠陥発見状況タイプ (左から安定, 漸増, 爆発) と信頼性 (左), セキュリティ (右) の関係

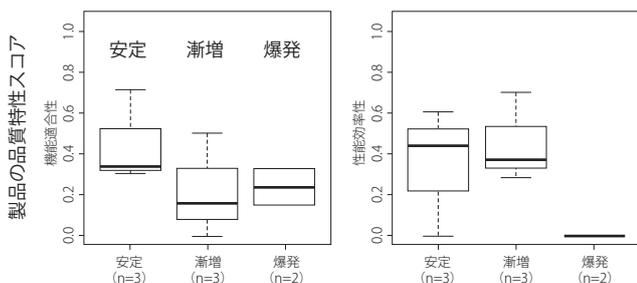


図 11 欠陥発見状況タイプ (左から安定, 漸増, 爆発) と機能適合性 (左), 性能効率性 (右) の関係

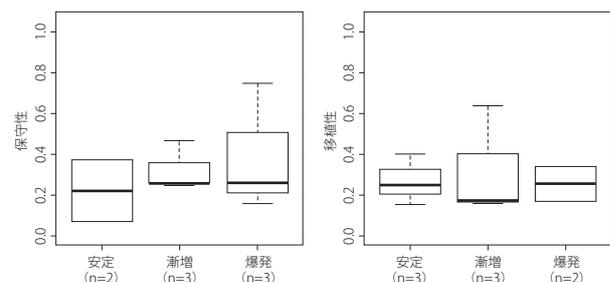


図 14 欠陥発見状況タイプ (左から安定, 漸増, 爆発) と保守性 (左), 移植性 (右) の関係

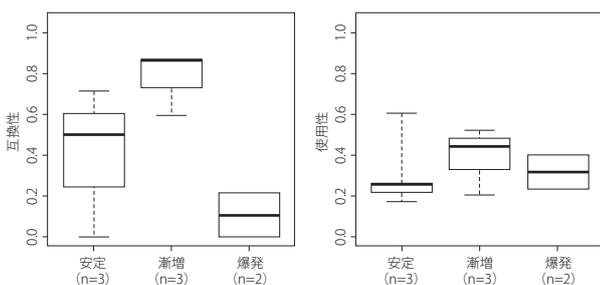


図 12 欠陥発見状況タイプ (左から安定, 漸増, 爆発) と互換性 (左), 使用性 (右) の関係

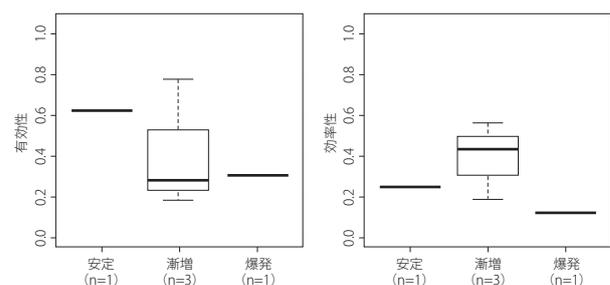


図 15 欠陥発見状況タイプ (左から安定, 漸増, 爆発) と有効性 (左), 効率性 (右) の関係

10に各タイプの欠陥発見状況及び欠陥数予測の例を示す。安定タイプについては十分な欠陥を発見できていると考えられる。漸増タイプについては、テストを実施するたびに欠陥が徐々に発見されており、リリース後も欠陥が発見される可能性があると考えられる。爆発タイプについては今後も多くの欠陥が発見されると考えられる。

欠陥発見状況のタイプ別の品質特性のスコアの分布を図11～図15に示す。ただし利用状況網羅性、リスク回避性、満足性については、測定評価できた製品数が限られたため省略した。品質特性別に結果と考察を以下に述べる。

(1) 機能適合性, 信頼性, 有効性:

- 結果: 機能適合性, 信頼性, 有効性について安定タイプにおいて高品質であった。
- 考察: これは, 機能適合性, 信頼性, 有効性が高い製品について, 十分にテストされ欠陥を発見していると考えられる。

(2) 性能効率性, 互換性:

- 結果: 性能効率性, 互換性については爆発タイプにおいて低品質であることが分かった。
- 考察: これは, 性能効率性, 互換性が低いソフトウェア製品については, テストで十分に欠陥を発見できず, 今後も欠陥が発見される可能性が高いと考えられる。

(3) ほかの品質特性:

- 結果: ほかの特性については, 顕著な違いは見られなかった。
- 考察: これは対象とした製品数の数が少ないことも原因と考えられる。

4.4 プロジェクトコンテキストと品質特性

製品コンテキストとしてドメイン, 対象開発期間, 開発形態, 提供種別, 機能数, プログラムソースコード行数の各情報を製品開発元から得た。製品によっては一部の情報が得られていない。コンテキストの一部と平均スコアの関係を図16, 図17に示す。コンテキストの種別のうちで顕著であった結果と考察を以下に述べる。

(1) ドメイン別の傾向:

- 結果: 互換性及びセキュリティについてドメインにより平均スコアに顕著な差が見られた。エンドユーザ向けサービス製品においてセキュリティは極めて高く, 数値計算シミュレーション製品において低い結果となった。
- 考察: 重視される品質特性の違いに起因すると考えら

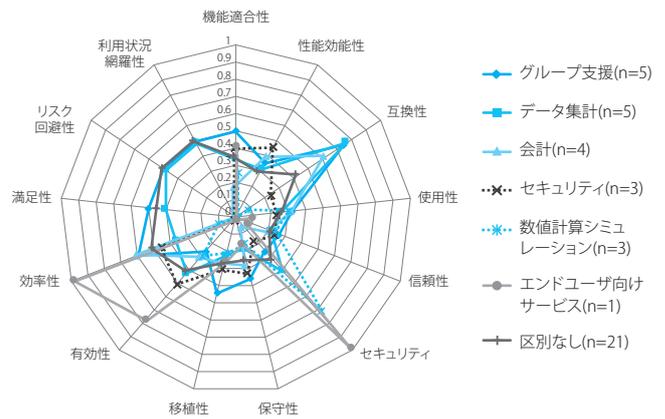


図16 ドメイン別の品質評価スコア平均

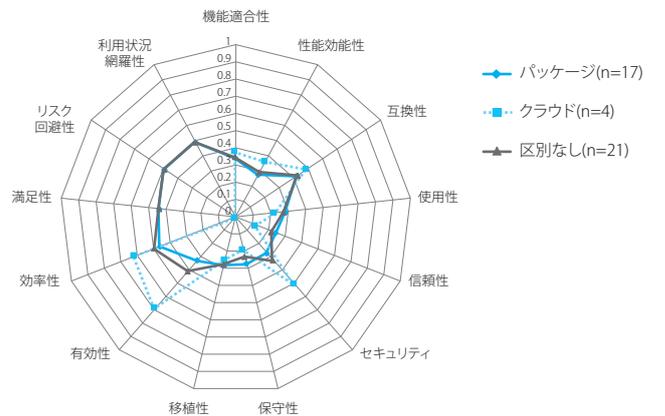


図17 提供種別に基づく品質評価スコア平均

れる。もう一つの理由としては, 具体化できた測定法が限られていたこともあげられる。

(2) 提供種別・パッケージ製品の傾向:

- 結果: パッケージ製品はクラウド製品に比べてセキュリティが著しく低い結果となった。
- 考察: ネットワーク接続や利用環境の変化を考慮する場合にセキュリティの強化が課題とすることができる。

(3) 提供種別・クラウド製品の傾向:

- 結果: クラウド製品についてはパッケージ製品に比べて保守性や移植性がやや低い。
- 考察: 原因として, クラウド製品におけるソフトウェアの保守及び移植においてはDevOpsに代表される運用体制と連携した機能拡張や, 仮想化システムの標準への対応といったクラウド環境特有の考慮が必要となり, パッケージ製品とはやや異なる点が測定評価に用いた国際規格シリーズSQuAREにおいて考慮されていないと推測できる。また, 信頼性についてもクラウド製品はパッケージ製品に比べて低い, これはSQuAREにおいて測定量の定義にあたりクラウド環境の考慮が不足している可能性がある。SQuAREにおけ

る品質測定方法の多くは1990年代までのソフトウェア製品の形態及び開発方法を念頭においており、アジャイル開発やプラットフォームとしてのクラウドに対する考慮を幾らか欠いており、今後の対応強化が期待される。

なお規模、期間、開発形態のそれぞれにおいては、品質に顕著な傾向の違いが見られなかった。

4.5 調査結果のまとめと提言

実現した枠組みを21製品へ実適用して得られた結果を以下にまとめると共に、産業界へ以下を提言する。なおこれらの提言は、調査対象の製品提供元へも伝達済みである。

- 提言 1. すべてがつながる IoT/IoE 時代に最重要となるはずの品質であるセキュリティ及び互換性が一部低いパッケージ製品が見られた。ソフトウェアの開発プロセスや方法、取り巻く環境が変革されつつある中で、ソフトウェア品質に対する意識の変革も必要である。
- 提言 2. 機能適合性の作り込みの結果として、本来の利用者にとって重要な使用性が損なわれている可能性がある。関連して、利用時の品質である効率性と有効性について2極化の傾向が見られた。製品の価値を判断する立場は、本来は利用者であり、様々な品質特性を多面的に考慮するユーザ中心の取り組み（例えばユーザ中心の設計など）が今後は求められる。同様に、開発者視点の製品品質を作り込む中で、常に利用者視点の利用時の品質も併せて意識し、それらの測定評価及び改善の取り組みが今後はより一層重要である。
- 提言 3. 製品品質と利用時の品質の間の関係としては、信頼性と有効性及び移植性と効率性の間に正の相関が見られた。加えて、信頼性成長モデルにより特定される欠陥発見状況が安定タイプである場合に有効性が高く、逆に漸増及び爆発タイプであると有効性が低い傾向にあった。有効性を高めるためには、開発中において確実に摘出欠陥数を収束させることが望ましい。
- 提言 4. 21 製品の協力を得たが、品質特性単位で見ると実際に測定評価できた製品数は半数に満たないものがほとんどであった。この原因としては、各製品開発元において、様々な品質特性の観点から評価可能とするための根拠となるデータを記録していないこと、及び、SQuaRE においてもともと想定している品質測定の方法において必要な目標値がほとんど設定されていないことが挙げられる。多面的な品質測定評価を可能とするためのデータの記録、並びに、ベンチマークを参照した上での目標値設定が重要である。

4.6 制限

製品への実適用を通じて提案する枠組みを具体的に利用できることを確認した。また上述のように、企業において枠組みと品質ベンチマークの活用が始まっている [小島 17]。しかしながらそれらをもって枠組みの妥当性のすべてを明らかとしたこととはならない。枠組みにより得られる評価結果と、各製品の品質要求との対比や、枠組みが直接に扱わない製品リリース後の特徴量や実績値（例えば保守性であればリリース後の保守工数）との対比を通じて、枠組みの妥当性をより明らかとすることが今後の課題である。

測定法や品質特性によっては対象製品数や取得できたデータが少なく統計的に母数として十分な数量とは言い難いものもあることに留意が必要である。上述の分析結果はあくまでも今回取得したデータ規模における推計である。また、今回の研究はソフトウェアのすべての領域にわたっているわけではなく、パッケージなどの供給側の立場からの品質評価が主になっている点にも留意が必要である。

ユーザテストについて、上述のように実利用者に代わり、我々が異常系テスト項目を定義し、正常系テスト項目と併せて我々がテストした。対象領域やテスト項目への理解を深めながら進めており、少なくとも初心者相当の利用者の視点を確保できたと考えられる。しかしながら、現実の業務活動の環境下において実施したものではないため、幅広い利用者の視点からの様々な利用をテストできていない可能性がある。また、操作の熟練度において実利用者の平均的なものよりも劣っている可能性があり、これらは有効性及び効率性の測定評価における妥当性への脅威である。その緩和や克服に向けては、各製品の実利用者もしくは一定時間をかけて熟練した被験者を準備することが対策として考えられる。

5 おわりに

得られた枠組みと品質ベンチマークは、ソフトウェア製品の開発側や運用側において、開発・保守・運用中あるいは運用検討中のソフトウェア製品の品質を、客観的、定量的かつ総合的に評価可能とし、評価結果を開発・保守における品質要求定義や品質改善、取捨選択の判断材料に役立てられることが期待できる。実際に、企業において枠組みと品質ベンチマークの活用が始まっている [小島 17]。また我々は、具体的な測定データに基づいて産業界への提言を示した。

得られた調査結果を品質実態の調査結果として捉えようとする場合に、品質実態の限られた一端を明らかとするこ

とつながったと考えられるが、品質特性によっては記録の不足や目標値未設定といった理由からデータが得られた製品数が限られている。そこで更なる品質実態の調査に向けて、今後データ数を増やし、継続的に関係分析結果を更新する予定である。また、データ数が一定数に達した時点で、より精緻な関係モデルの導出を試みる予定である。

A 付録

品質特性ごとに一覧を次頁以降に示す。表中の“SQuaRE”は、SQuaREにおいて対応する測定量を示す。対応する測定量がない場合は空欄である。“ID”は、我々が定義した各測定法に一意に付与した識別子である。

A.1 製品品質の測定法一覧

(1) 機能適合性

(副)特性	SQuaRE	ID	名称	定義	詳細
機能完全性	FCp-1	FCp.1.1	要求実装率	$X=A/B$	A= 見送らず、着手・実現した要求の数 B= 対象期間内で挙った要求の数
機能正確性	FCr-1	FCr.1.1	深刻不具合除去率	$X=1-A/B$	A= そのうち、残存している不具合の数 B= 対象期間内で発見した深刻な不具合の数
機能適切性	FAp-1	FAp.1.1	システム試験数目標達成率	$X=A/B$	A= 対象期間内でのテストケース実施数 B= 対象期間内でのテストケースの実施目標数
機能適切性	FAp-1	FAp.1.2	ユーザの意図に則す度合い	$X=X$ のユーザ群での平均	X= 意図に即す度合のユーザアンケート回答

(2) 性能効率性

(副)特性	SQuaRE	ID	名称	定義	詳細
時間効率性		PTb.0.1	時間効率性の試験実績	$X=A/B$	A= 応答時間、ターンアラウンドタイム、スループットのうち、実施しているものの数 B=3
時間効率性	PTb-1	PTb.1.1	応答時間 平均	X' = タスク群での X 平均値	X= ある性能試験タスクでの実測値
時間効率性	PTb-2	PTb.2.1	応答時間 実測対目標	X' = タスク群での X 平均値 $X=A/B$	A= ある性能試験タスクでの実測値 B= その性能試験タスクでの目標値
時間効率性	PTb-3	PTb.3.1	ターンアラウンドタイム 平均	X' = タスク群での X 平均値	X= ある性能試験タスクでの実測値
時間効率性	PTb-4	PTb.4.1	ターンアラウンドタイム 実測対目標	X' = タスク群での X 平均値 $X=A/B$	A= ある性能試験タスクでの実測値 B= その性能試験タスクでの目標値
時間効率性	PTb-5	PTb.5.1	スループット 目標達成率	X' = タスク群での X 平均値 $X=A/B$	A= ある性能試験タスクでの実測値 B= その性能試験タスクでの目標値
資源効率性		PRu.0.1	資源効率性の試験実績	$X=A/B$	A=2 種のうち、実施しているものの数 B=2 (CPU 使用率系試験、メモリ使用率試験)
資源効率性	PRu-1	PRu.1.1	CPU 使用率 最大値	X' = タスク群での X 平均値	X= ある性能試験タスク実施中の CPU 使用率 最大値
資源効率性	PRu-2	PRu.2.1	メモリ 使用率 最大値	X' = タスク群での X 平均値	X= ある性能試験タスク実施中のメモリ使用率 最大値
容量満足性		PCa.0.1	容量満足性の試験実績	$X=0$ または 1	X= ユーザ同時アクセス数系の試験を実施していれば 1、さもなくば 0
容量満足性	PCa-2	PCa.2.1	ユーザ同時アクセス可能数 目標達成率	X' = タスク群での X 平均値 $X=A/B$	A= ある性能試験タスクでの実測値 B= その性能試験タスクでの目標値

(3) 互換性

(副)特性	SQuaRE	ID	名称	定義	詳細
共存性	CCo-1	CCo.1.1	他製品共存試験実績	X'=サーバマシン, クライアントマシンの X 平均 X=A/B	A= 試験環境で意図的に共存させているソフトウェア種別 (セキュリティソフトウェア, 対象製品) B=2
相互運用性	Cln-1	Cln.1.1	ファイル形式のインポート/エクスポート両対応率	X=A/B	A= インポート・エクスポート両対応のもの数 B= 取り扱っているファイル拡張子の数

(4) 使用性

(副)特性	SQuaRE	ID	名称	定義	詳細
適切度 認識性	UAp-2	UAp.2.1	機能の動画説明対応率	X=A/B	A= 説明動画が公開されている機能の数 B= 機能の数
習得性	ULe-1	ULe.1.1	機能の説明カタログ記載率	X=A/B	A= 機能のうち, カタログに載っている数 B= 機能の数
習得性	ULe-1	ULe.1.2	機能の説明マニュアル記載率	X=A/B	A= 機能のうち, マニュアルに載っている数 B= 機能の数
運用操作性	UOp-6	UOp.6.1	機能の Undo 対応率	X=A/B	A= 元に戻せる機能の数 B=Undo が必要な機能の数
ユーザエラー 防止性	UEp-1	UEp.1.1	機能での入力内容チェック対応率	X=A/B	A=Bのうち, エラーメッセージや警告文が出る機能の数 B= ユーザインプットを求めている機能の数
ユーザインターフェース 快美性	Uln-1	Uln.1.1	UI の使いやすさ度合い	X'=X のユーザ群での 平均	X=UI の使いやすさのユーザアンケート回答
アクセシビリティ	UAc-3	UAc.3.1	機能の聴覚ハンディキャップ配慮率	X=A/B	A= 機能のうち, 聴覚ハンディキャップへの配慮があるものの数 B= 機能の数
アクセシビリティ	UAc-4	UAc.4.1	機能の視覚ハンディキャップ配慮率	X=A/B	A= 機能のうち, 聴覚ハンディキャップへの配慮があるものの数 B= 機能の数
アクセシビリティ	UAc-5	UAc.5.1	言語の対応度合い	X=A/B	A= 各言語の対応度重み総和 B= 対応言語の数

(5) 信頼性

(副)特性	SQuaRE	ID	名称	定義	詳細
成熟性	RMa-1	RMa.1.1	不具合除去率 (単体試験)	$X=A/B$	A=Bのうち修正されたものの数 B=対象期間内に発見した欠陥数 (fault)
成熟性	RMa-1	RMa.1.2	不具合除去率 (結合試験)	$X=A/B$	A=Bのうち修正されたものの数 B=対象期間内に発見した欠陥数 (fault)
成熟性	RMa-1	RMa.1.3	不具合除去率 (システム試験)	$X=A/B$	A=Bのうち修正されたものの数 B=対象期間内に発見した欠陥数 (fault)
成熟性	RMa-2	RMa.2.1	MTBF 目標 達成率	$X=C/D$	A=稼働時間 B=故障発生回数 C=MTBF 実測値 =A/B D=MTBF 目標値
成熟性	RMa-3	RMa.3.1	不具合発見率 (単体試験)	$X=A/B$	A=対象期間内での不具合発見数 (実測) B=対象期間内での不具合発見数の目標値
成熟性	RMa-3	RMa.3.2	不具合発見率 (結合試験)	$X=A/B$	A=対象期間内での不具合発見数 (実測) B=対象期間内での不具合発見数の目標値
成熟性	RMa-3	RMa.3.3	不具合発見率 (システム試験)	$X=A/B$	A=対象期間内での不具合発見数 (実測) B=対象期間内での不具合発見数の目標値
成熟性	RMa-3	RMa.3.4	不具合発見率 (チケットベース)	$X=B/(B-abs(B-A))$	A=対象期間内での不具合発見数 (実測) B=対象期間内での不具合発見数の予測値 abs=絶対値関数 ※ Bを信頼性曲線を元に算出
成熟性	RMa-4	RMa.4.2	試験実施率 (システム試験)	$X=A/B$	A=対象期間内でのテストケース実施数 B=対象期間内でのテストケースの実施目標数
可用性		RAv.0.1	運用試験実績	$X=0$ または 1	運用試験を実施していれば 1, さもなくば 0
可用性	RAv-1	RAv.1.1	運用実時間 対 規定時間	$X=A/B$	A=継続運用時, 実際に製品が正常稼働できた時間 B=継続運用時, 製品が正常稼働し続けられる時間の予想値
可用性	RAv-2	RAv.2.1	システムダウン 時間 実際対 目標	$X=(A/B)/C$	A=システムダウンしていた時間合計 B=システムダウン回数 C=システムダウン時間平均の目標値
障害許容性	RFt-1	RFt.1.1	fault-pattern テスト ケース (結合試験)	$X=A/B$	A=Bのうち, 成功した数 B=対象期間内での fault Pattern テストケース数
回復性	RRe-1	RRe.1.1	システムダウン 回復時間 実際 対目標	$X=(A/B)/C$	A=システムダウン回復にかかった時間合計 B=システムダウン回数 C=システム回復時間平均の目標値

(6) セキュリティ

(副)特性	SQuaRE	ID	名称	定義	詳細
機密性	SCo-1	SCo.1.1	データのアクセス権限管理対応率	$X=A/B$	A=Bのうち、アクセス権限管理が可能なものの数 B= データ種別の数
機密性	SCo-2	SCo.2.1	データの暗号化対応率	$X=A/B$	A=Bのうち、暗号化されているものの数 B= データ種別の数
インテグリティ	SIn-2	SIn.2.1	データの破損防止策対応率	$X=A/B$	A=Bのうち、破損防止機能のあるものの数 B= データ種別の数
否認防止性	SNo-1	SNo.1.1	ネットワーク経路のデジタル署名対応率	$X=A/B$	A=Bのうち、デジタル署名有効なもの数 B= 利用している通信経路の種類数
責任追跡性	SAC-1	SAC.1.1	データのアクセスログ対応率	$X=A/B$	A=Bのうち、アクセス履歴がログに残るものの数 B= データ種別の数
真正性	SAu-1	SAu.1.1	ログイン機能での認証方式対応率	X' = ログイン機能についてのX $X=A/B$	A= 製品でサポートしている認証方式の種類数 B=6種：固定パスワード、ワンタイムパスワード、期限付きパスワード、物理的トークン、生体認証、解読型

(7) 保守性

(副)特性	SQuaRE	ID	名称	定義	詳細
モジュール性	MMo-1	MMo.1.1	クラスの結合度	X' = クラス群でのX平均	X= クラス結合度 (クラス単位)
モジュール性	MMo-2	MMo.2.1	関数のサイクロマティック複雑度	X' = 関数群でのX平均	X= サイクロマティック複雑度 (関数単位)
再利用性	MRe-1	MRe.1.1	クラスの凝集性の欠如	X' = クラス群でのX平均	X=100-LCOM2 (クラス単位) LCOM2= 集性の欠如の定義 ver2 ※ここでは LCOM2 が 0 ~ 100 の値
解析性	MAn-1	MAn.1.1	データのアクセスログ 対応率	$X=A/B$	A=Bのうち、アクセス履歴がログに残るものの数 B= データ種別の数
修正性	MMd-3	MMd.3.1	不具合除去率 (単体試験で発見分)	$X=A/B$	A= 修正済み不具合数 B= 発見済み不具合数
修正性	MMd-3	MMd.3.2	不具合除去率 (結合試験で発見分)	$X=A/B$	A= 修正済み不具合数 B= 発見済み不具合数
修正性	MMd-3	MMd.3.3	不具合除去率 (システム試験で発見分)	$X=A/B$	A= 修正済み不具合数 B= 発見済み不具合数
試験性	MTe-1	MTe.1.1	モジュールの単体試験 実施率	$X=A/B$	A= 単体試験実施済みモジュール数 B= 全モジュール数

(8) 移植性

(副)特性	SQuaRE	ID	名称	定義	詳細
適応性		PAd.0.1	複数環境の試験実現	$X1=A/B$ $X2=C/B$	A= 複数環境で試験済みの主機能数 B= 主機能数 C= 複数環境で試験成功した主機能数
設置性		Pln.0.1	インストールの試験実現	$X=0$ または 1	インストール試験を実施していれば1、さもなければ0
設置性	Pln-1	Pln.1.1	インストール時間 平均	X' = タスク群での X 平均 $X=A/B$ (インストール時間)	A= あるタスクでの実測値 B= あるタスクでの目標値
設置性	Pln-2	Pln.2.1	インストーラ提供形態 対応率	$X=A/B$	A=3 種の内, 対応している製品提供形態 B=3 ※種別: Web, CD, セットアップ代行
設置性	Pln-2	Pln.2.2	インストールオプション 対応率	X' = サーバソフトでの X $X=A/B$	A= 以下の8種のうち, 対応しているインストールオプション B=8 ・アンインストール時に, あらゆる設定情報を消去できるか ・アンインストール時に, 利用者が希望すれば設定情報をレジストリに残せるか ・アンインストール時に, 利用者が希望すれば設定情報をレジストリ以外の場所に残せるか ・サーバについて複数台構成にできるか ・サーバについて一台編成にできるか ・root フォルダのインストール先フォルダの変更可能か ・インストーラを途中で中断できるか (最初からやり直しにせずに) ・事前にインストールしておくべきソフトや環境がない状態でインストールしようとしたとき, それ以上先に進めなかったり警告したりしてくれるか
置換性	Pre-1	Pre.1.1	追加学習必要度合い	X' = X のユーザ群での平均	X = 満足度のユーザアンケート回答

A.2 利用時の品質の測定法一覧

(副)特性	SQuaRE	ID	名称	定義	詳細
有効性	Ef-1	Ef.1.1	タスク完了率	$X=A/B$	A=完了したタスク数 B=全タスク数
有効性	Ef-3	Ef.3.1	タスク当たりエラー数	$X=A/B$	A=全エラー数 B=全タスク数
有効性	Ef-4	Ef.4.1	エラーが発生したタスクの率	$X=A/B$	A=エラーがあったタスク数 B=全タスク数
有効性	Ef-5	Ef.5.1	エラーを起こした被験者の率	$X=A/B$	A=エラーを起こした人数 B=被験者人数
効率性	Ey-1	Ey.1.1	タスクにかかった時間の平均	X' =タスク群でのX平均	A=タスクの開始時刻 B=タスクの終了時刻
効率性	Ey-5	Ey.5.1	タスク中の総アクションの無駄でないアクションの率	X' =タスク群でのX平均	A=必要アクション数 B=全アクション数
実用性	SUs-1	SUs.1.1	製品に対する満足度	X' =Xのユーザ群での平均	X=満足度のユーザアンケート回答
実用性	SUs-1	SUs.1.2	ネットプロモータースコア	X' =Xのユーザ群での平均	X=ネットプロモータースコアのユーザアンケート回答
実用性	SUs-2	SUs.2.1	機能に対する満足度	X' =Xのユーザ群での平均	X=各機能満足度のユーザアンケート回答結果のユーザにおける平均
信用性	STr-1	STr.1.1	信用度合い	X' =Xのユーザ群での平均	X=信用度のユーザアンケート回答
快感性	SPI-1	SPI.1.1	快感度合い	X' =Xのユーザ群での平均	X=ストレスのない利用に関するユーザアンケート回答
快適性	SPo-1	SCo.1.1	快適度合い	X' =Xのユーザ群での平均	X=快適度のユーザアンケート回答
経済リスク緩和性		REc.0.1	経済的損失の無さ	X' =Xのユーザ群での平均	X=経済的な影響の感じなさのユーザアンケート回答
健康・安全リスク緩和		RHe.0.1	健康や人命への影響の無さ	X' =Xのユーザ群での平均	X=健康や人命への影響の感じなさのユーザアンケート回答
環境リスク緩和性		REn.0.1	環境への影響の無さ	X' =Xのユーザ群での平均	X=自然・社会環境への影響の感じなさのユーザアンケート回答
利用状況完全性		CCm.0.1	主要な目的以外での製品利用	X' =Xのユーザ群での平均	X=主要な目的以外で製品を利用することのユーザアンケート回答
柔軟性		CFI.0.1	非主要目的での製品利用時タスク達成度合い	X' =Xのユーザ群での平均	X=主要な目的以外の利用時の目的達成のユーザアンケート回答

謝辞：本研究は独立行政法人情報処理推進機構技術本部ソフトウェア高信頼化センター（SEC：Software Reliability Enhancement Center）が実施した「2015年度ソフトウェア

ア工学分野の先導的研究支援事業」の支援のもと行われた。調査対象製品開発元、評価機関、CSAJほか関係各位に謝意を記す。

【参考文献】

- [東 15] 東基衛, "システム・ソフトウェア品質標準 SQuaRE シリーズの歴史と概要", SEC Journal, 10(5), 2015.
- [ISO25000] ISO/IEC 25000:2014 Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- Guide to SQuaRE
- [WSQB] WSQB17: Waseda Software Quality Benchmark, http://www.washi.cs.waseda.ac.jp/?page_id=3479
- [小島 17] 小島嘉津江, 森田純恵, 廣瀬竹男, 若本雅晶, 菊池慎司, 鷺崎弘宜, "ソフトウェア品質技術が品質特性に与える効果の見える化と活用の一考察", ソフトウェア品質シンポジウム, 2017.
- [鷺崎 07] 鷺崎弘宜ほか, "プログラムソースコードのための実用的な品質評価枠組み", 情報処理学会論文誌, 48(8), 2007.
- [Bansiya02] J. Bansiya and C.G. Davis, "A Hierarchical Model for Object-Oriented Design Quality Assessment," IEEE Transactions on Software Engineering, 28(1), 2002.
- [Ortega03] M. Ortega, M. Perez and T. Rojas, "Construction of A Systematic Quality Model for Evaluating A Software Product," Software Quality Journal, 11(3), 2003.
- [鷺崎 10] 鷺崎弘宜, 田邊浩之, 小池利和, "ソースコード解析による品質評価の仕組み", 日経エレクトロニクス, 2010年1月25日号
- [ISO25010] ISO/IEC 25010:2011 Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models
- [ISO25022] ISO/IEC 25022:2016 Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- Measurement of quality in use
- [ISO25023] ISO/IEC 25023:2016 Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- Measurement of system and software product quality
- [Arban03] A. Abran, et al., "Usability meanings and interpretations in ISO standards." Software Quality Journal, 11(4), 2003.
- [Heidrich14] J. Heidrich, et al., "Model-based quality management of software development projects," Software Project Management in a Changing World, 2014.
- [Biscoglio14] I. Biscoglio and E. Marchetti, "Definition of Software Quality Evaluation and Measurement Plans: A Reported Experience Inside the Audio-Visual Preservation Context," 9th International Joint Conference on Software Technologies (ICSOFTE), 2014.
- [ISO25051] ISO/IEC 25051:2014 Software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - Requirements for quality of Ready to Use Software Product (RUSP) and instructions for testing
- [IPA14] IPA, "製品・システムにおけるソフトウェアの信頼性・安全性等に関する品質説明力強化のための制度構築ガイドライン", 2014.
- [PSQ] CSAJ, PSQ 認証制度, <http://www.psq-japan.com/>
- [IPA15] IPA/SEC, "ソフトウェア開発データ白書 2014-2015", 2015.
- [Cusumano03] M. Cusumano, et al., "Software Development Worldwide: The State of the Practice," IEEE Software, 20(6), 2003.
- [Jones09] C. Jones, "Software Engineering Best Practices: Lessons from Successful Projects in the Top Companies," McGraw-Hill, 2009.
- [鷺崎 16] 鷺崎弘宜, "実践的ソフトウェア品質測定評価のための4つの「落とし穴」と7つの「コツ」", 品質, 46(3), 2016.
- [Washizaki17] H. Washizaki, "Pitfalls and Countermeasures in Software Quality Measurements and Evaluations," Advances in Computers, 106, Elsevier, 2017.
- [Basili94] V. Basili, et al., "Goal, Question, Metric Paradigm," Encyclopedia of Software Engineering, 1, 1994.
- [Honda17] K. Honda, et al., "Generalized Software Reliability Model Considering Uncertainty and Dynamics: Model and Applications," Int'l J. Software Engineering and Knowledge Engineering, 27(6), 2017.
- [Alves10] T. L. Alves, et al., "Deriving metric thresholds from benchmark data," IEEE International Conference on Software Maintenance (ICSM), 2010.